Adaptive Design for Global Fit of Non-stationary Surfaces in Computer Experiments

Marian L. Frazier* William I. Notz†

Abstract

Computer experiments are used to study processes that are too difficult or dangerous to experiment with in the physical world. Complex computer code that simulates these experiments often results in an extremely long run time. The design points at which to run the simulations must be chosen carefully and intelligently; thus, sequential designs that allow users to focus their attentions on interesting areas of the response are a logical choice. We propose a family of sequential design methods that are developed for use when it is believed that the response surface may exhibit non-stationary behavior. These criteria, which were inspired by an expected improvement criterion, focus on the search for areas with large changes in slope, with the idea that sudden changes in slope are an indication of non-stationary "breaks" in the response. While seeking out these boundary points, our methods still result in an effective fit of the entire response surface using a small number of design points. The merits of these methods are exhibited in a two-dimensional example, including comparisons to existing sequential design methods.

Key Words: computer experiments, Bayesian treed Gaussian processes, sequential design

1. Introduction

Computer experiments are used to study physical processes that are too costly, difficult, or dangerous to experiment with in the physical world. Engineers and computer scientists produce the simulator (computer code) that they feel is a relatively close approximation of the complex reality that they wish to investigate. The computer code is a "black box": values of the input (predictor) variables go in, and values of the output (response) variable(s) come out. It is the job of statisticians to handle what goes in (the design) and what comes out (the analysis of the responses).

Of course, we wish to understand the input-output relationship. Since the computer code that simulates these physical experiments is often very complex, it is not an efficient use of time to run the simulator over the entire input space. Instead, we build an inexpensive (fast) approximation to the unknown response surface, known as a "surrogate model" or "emulator." In order to build this surrogate, we require some data from the simulator. Since the simulator is assumed to have a long run time, each data point is considered to be costly. Hence, the design points must be chosen carefully by an efficient design method that can investigate the response surface in a small number of samples. An especially efficient design method is sequential (or adaptive) design. In a sequential design, a small initial space-filling sample is taken, and an estimated model for the response surface is built based on this sample. Based on this estimated model, the statistician decides where the next sampled point should be. Sequential designs are a natural choice for computer experiments, where obtaining each data point is costly, because users can focus their attentions on interesting areas of the response surface, as described by features of the predicted model from already-sampled points. In this way, we can get more relevant information in a smaller number of samples than if we were to take one large batch of randomly-scattered design points. The choice of the next sampled point will depend on the experimental goals: local

*Gustavus Adolphus College, 800 W. College Ave., Saint Peter, MN 56082

†The Ohio State University, 1958 Neil Ave., Columbus, OH 43210

or global optimization, global fit of the response surface, calibration to some physical experiment, sensitivity analysis, or prediction/estimation of a particular subset of the response are all popular choices. Different sequential designs have been developed for all of these applications. (See Santner et al. (2003) for an overview, or Lam (2008) for a comparison study of several methods.)

Historically, the outputs of computer experiments have been modeled using a Gaussian stochastic process model (GP), which assumes stationarity in the response surface (see Sacks et al. 1989b; Santner et al. 2003). However, work in the last ten years by Gramacy and Lee (see Gramacy et al. 2004; Gramacy 2005; Gramacy and Lee 2008a,b) relaxes that assumption and allows us to accurately build surrogate models for non-stationary processes using their treed Gaussian process model (TGP). Most existing experimental design criteria for computer experiments are built on the assumption of stationarity and the use of the stationary GP model. Thus, there exists opportunity for new contributions to design in situations where we believe there may be non-stationarity behavior in the response surface.

In Frazier and Notz (2014), the authors introduced the Expected Difference of Slopes (EΔM), a new sequential design method for the purpose of obtaining a good global fit of the entire response surface when the response is non-stationary. They found that when non-stationary behavior was present – the true response surface has well-defined boundary points, or when the "interesting" areas are highly localized – EΔM was successful at achieving an efficient fit. But in cases where the "interesting" areas are more spread out over the input space, or when there are not distinct boundaries between regions, EΔM was less successful; its search was too localized.

The focus of this work is modification of the EΔM criterion, with the goal of achieving good global fit on a wider array of surfaces. Section 2 introduce the details of the Gaussian stochastic process (GP) and treed Gaussian process (TGP) models, along with the prediction and estimation of these models. Section 3 reviews the Expected Difference of Slopes, along with two competing methods, and introduces our modification. An empirical study comparing these methods on a synthetic 2-dimensional function is carried out in Section 4. Finally, Section 5 discusses the advantages and disadvantages of the four design criteria, and proposes several areas for future research.

2. Stationary and Non-stationary Gaussian Processes

In a computer experiment, the output $z(\cdot)$, evaluated at a $m_X \times 1$ vector of inputs $\boldsymbol{x} \in \boldsymbol{\mathcal{X}} \subset \mathbb{R}^{m_X}$, is thought of as a realization of the random process

$$Z(\boldsymbol{x}) = \sum_{j=1}^{p} \beta_j f_j(\boldsymbol{x}) + Y(\boldsymbol{x}) = \boldsymbol{f}^T(\boldsymbol{x})\,\boldsymbol{\beta} + Y(\boldsymbol{x}), \tag{1}$$

where $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x})), f_2(\boldsymbol{x})), \ldots, f_p(\boldsymbol{x}))^T$ is a $p \times 1$ vector of known regression functions of $\boldsymbol{x}$, $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)^T$ is a $p \times 1$ vector of regression coefficients, and $Y(\boldsymbol{x})$ is a mean-zero random process with covariance given by

$$\mathrm{Cov}[Y(\boldsymbol{x}_1), Y(\boldsymbol{x}_2)] = \sigma_Y^2 R(\boldsymbol{x}_1, \boldsymbol{x}_2). \tag{2}$$

If we further assume that $Y(\boldsymbol{x})$ is a Gaussian stochastic process (Gaussian random function), model (1) is called a Gaussian stochastic process (GP) model (Sacks et al. 1989b,a; Santner et al. 2003).

Traditionally (Santner et al. 2003), the stochastic process $Y(\cdot)$ is assumed to be stationary, so that the correlation between two points depends only on the distance between two

points, not on the location of those points:

$$R(\boldsymbol{x}_i, \boldsymbol{x}_j) = R(\boldsymbol{x}_i - \boldsymbol{x}_j).$$

We want to predict the output $Z(\boldsymbol{x}_0)$ at a single untried $\boldsymbol{x}_0 \in \boldsymbol{\mathcal{X}}$ based on n training points $D = \{\boldsymbol{X}, \boldsymbol{Z}^n\}$. The distribution of outputs at $\boldsymbol{x}_0$ is Gaussian with mean and prediction variance

$$\hat{z}(\boldsymbol{x}_0) = \boldsymbol{f}^T(\boldsymbol{x}_0)\hat{\boldsymbol{\beta}} + \boldsymbol{r}^T(\boldsymbol{x}_0)\,\boldsymbol{R}^{-1}(\boldsymbol{Z}^n - \boldsymbol{F}\,\hat{\boldsymbol{\beta}}), \tag{3}$$

$$\hat{\sigma}^2(\boldsymbol{x}_0) = \sigma_Y^2\left[1 - \boldsymbol{r}_0^T\,\boldsymbol{R}^{-1}\,\boldsymbol{r}_0 + (\boldsymbol{f}_0^T - \boldsymbol{r}_0^T\,\boldsymbol{R}^{-1}\,\boldsymbol{F})(\boldsymbol{F}^T\,\boldsymbol{R}^{-1}\,\boldsymbol{F})^{-1}(\boldsymbol{f}_0^T - \boldsymbol{r}_0^T\,\boldsymbol{R}^{-1}\,\boldsymbol{F})^T\right], \tag{4}$$

where

$$\begin{aligned}
\boldsymbol{F} &= (\boldsymbol{f}(\boldsymbol{x}_1), \ldots, \boldsymbol{f}(\boldsymbol{x}_n))^T \\
\boldsymbol{f}_0 &= \boldsymbol{f}(\boldsymbol{x}_0) \\
\boldsymbol{R} &= [R(\boldsymbol{x}_i, \boldsymbol{x}_j)] \quad \text{for} \quad i, j \in 1, \ldots, n \\
\boldsymbol{r}_0 &= \boldsymbol{r}(\boldsymbol{x}_0) = (R(\boldsymbol{x}_1, \boldsymbol{x}_0), \ldots, R(\boldsymbol{x}_n, \boldsymbol{x}_0))^T \\
\hat{\boldsymbol{\beta}} &= (\boldsymbol{F}^T\,\boldsymbol{R}^{-1}\,\boldsymbol{F})^{-1}\,\boldsymbol{F}^T\,\boldsymbol{R}^{-1}\,\boldsymbol{Z}^n\,.
\end{aligned}$$

2.1 Bayesian Treed GPs

The sequential design method proposed in Section 3 is specifically developed for use when it is believed that the response surface may exhibit non-stationary behavior. Thus, use of the stationary Gaussian process model described above may not be adequate. To this end, surrogate modeling in this work will be performed using the Bayesian treed Gaussian process model (TGP) developed by Gramacy and Lee (Gramacy et al. 2004; Gramacy 2005; Gramacy and Lee 2008a,b, 2009, 2010).

TGP models build on classification and regression tree (CART) models popularized by Breiman et al. (1984), and the Bayesian versions developed Chipman et al. (1998) and Chipman et al. (2002). The TGP consists of splitting the input space into $\mathcal{N}$ non-overlapping regions, then fitting independent stationary Gaussian process models to each region.

Each region b_ν $(\nu = 1, \ldots, \mathcal{N})$ contains data $D_\nu = \{\boldsymbol{X}_\nu, \boldsymbol{Z}_\nu^{n_\nu}\}$ from already-sampled points. The TGP model fits an independent stationary GP with linear trend (Equation 1) to the data D_ν within each region. For a particular region ν, the hierarchical model is

$$\begin{aligned}
\boldsymbol{Z}_\nu \,|\, \boldsymbol{\beta}_\nu, \sigma_\nu^2, \boldsymbol{R}_\nu &\sim N_{n_\nu}(\boldsymbol{F}_\nu\,\boldsymbol{\beta}_\nu, \sigma_\nu^2\,\boldsymbol{R}_\nu) \\
\boldsymbol{\beta}_\nu \,|\, \sigma_\nu^2, \tau_\nu^2, \boldsymbol{W}, \boldsymbol{\beta}_0 &\sim N_{m_X}(\boldsymbol{\beta}_0, \sigma_\nu^2\tau_\nu^2\,\boldsymbol{W}) \\
\boldsymbol{\beta}_0 &\sim N_{m_X}(\boldsymbol{\mu}, \boldsymbol{B}) \\
\sigma_\nu^2 &\sim IG(\alpha_\sigma/2, q_\sigma/2) \\
\tau_\nu^2 &\sim IG(\alpha_\tau/2, q_\tau/2) \\
\boldsymbol{W} &\sim W((\rho\,\boldsymbol{V})^{-1}, \rho)
\end{aligned} \tag{5}$$

where $\boldsymbol{F}_\nu = (\boldsymbol{1}, \boldsymbol{X}_\nu)$ is the design matrix of $\boldsymbol{Z}_\nu$, and $\boldsymbol{W}$ is a $(m_X + 1) \times (m_X + 1)$ matrix. The hyperparameters $\boldsymbol{\mu}, \boldsymbol{B}, \boldsymbol{V}, \rho, \alpha_\sigma, q_\sigma, \alpha_\tau$, and q_τ are constants and are treated as known. IG is the Inverse Gamma distribution and W is the Wishart distribution. The correlation matrix $\boldsymbol{R}_\nu$ is specific to each region, but it is assumed that all $\boldsymbol{R}_\nu$ $(\nu = 1, \ldots, \mathcal{N})$ come from the same family of correlation functions, the separable Gaussian correlation with scale

parameters $d_{i,\nu}\,(i = 1, \ldots, m_X; \nu = 1, \ldots, \mathcal{N})$, plus a nugget parameter g_ν (Gramacy and Lee 2010):

$$R_\nu(\boldsymbol{x}_j, \boldsymbol{x}_k) = R^*_\nu(\boldsymbol{x}_j, \boldsymbol{x}_k) + g_\nu \delta_{j,k} \tag{6}$$

$$= \exp\left\{ -\sum_{i=1}^{m_X} \frac{|x_{ij} - x_{ik}|^2}{d_{i,\nu}} \right\} + g_\nu \delta_{j,k} \tag{7}$$

where $\delta_{\cdot,\cdot}$ is the Kronecker delta function.

Using this hierarchical GP model, the predicted value for $Z(\boldsymbol{x}_0)$, where $\boldsymbol{x}_0$ is in region b_ν, is normally distributed with mean and variance

$$\hat{z}(\boldsymbol{x}_0) = E(\boldsymbol{Z}(\boldsymbol{x}_0)|(\boldsymbol{x}_0, \boldsymbol{Z}^{n_\nu}) \in b_\nu) = \boldsymbol{f}^T(\boldsymbol{x}_0)\tilde{\boldsymbol{\beta}}_\nu + \boldsymbol{r}_\nu^T(\boldsymbol{x}_0)\, \boldsymbol{R}_\nu^{-1}(\boldsymbol{Z}_\nu^{n_\nu} - \boldsymbol{F}_\nu \tilde{\boldsymbol{\beta}}_\nu), \tag{8}$$

$$\hat{\sigma}^2(\boldsymbol{x}_0) = \mathrm{Var}(\boldsymbol{Z}(\boldsymbol{x}_0)|(\boldsymbol{x}_0, \boldsymbol{Z}^{n_\nu}) \in b_\nu) = \sigma_\nu^2 \left[\kappa(\boldsymbol{x}_0, \boldsymbol{x}_0) - \boldsymbol{q}_\nu^T(\boldsymbol{x}_0)\boldsymbol{C}_\nu^{-1}\boldsymbol{q}_\nu(\boldsymbol{x}_0)\right], \tag{9}$$

where

$$\tilde{\boldsymbol{\beta}}_\nu = \boldsymbol{V}_{\tilde{\beta}_\nu}(\boldsymbol{F}_\nu^T \boldsymbol{R}_\nu^{-1} \boldsymbol{Z}_\nu^{n_\nu} + \boldsymbol{W}^{-1} \boldsymbol{\beta}_0 / \tau_\nu^2) \tag{10}$$

$$\boldsymbol{V}_{\tilde{\beta}_\nu} = (\boldsymbol{F}_\nu^T \boldsymbol{R}_\nu^{-1} \boldsymbol{F}_\nu + \boldsymbol{W}^{-1} / \tau_\nu^2)^{-1},$$

$$\boldsymbol{C}_\nu^{-1} = \left(\boldsymbol{R}_\nu + \tau_\nu^2 \boldsymbol{F}_\nu \boldsymbol{W} \boldsymbol{F}_\nu^T\right)^{-1},$$

$$\boldsymbol{q}_\nu(\boldsymbol{x}_0) = \boldsymbol{r}_\nu(\boldsymbol{x}_0) + \tau_\nu^2 \boldsymbol{F}_\nu \boldsymbol{W} \boldsymbol{f}(\boldsymbol{x}_0), \tag{11}$$

$$\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2) = R_\nu(\boldsymbol{x}_1, \boldsymbol{x}_2) + \tau_\nu^2 \boldsymbol{f}^T(\boldsymbol{x}_1) \boldsymbol{W} \boldsymbol{f}(\boldsymbol{x}_2).$$

Within a region b_ν, there are $(3 + m_X + p)$ parameters $\boldsymbol{\theta}_\nu = \{\sigma_\nu^2, \tau_\nu^2, g_\nu, \boldsymbol{d}_\nu, \boldsymbol{\beta}_\nu\}$ that must be estimated using the data $D_\nu = \{\boldsymbol{X}_\nu, \boldsymbol{Z}_\nu^{n_\nu}\}$. For derivations, explanations of the estimation of these parameters, and details on tree structure, see Gramacy (2005).

In Gramacy (2005) and Gramacy and Lee (2008b), the authors suggest the treed Gaussian process with jumps to the limiting linear model (TGP LLM) as a way to reduce computational cost when the surface within a region is approximately linear. The linear model is as given in (5), except the conditional distribution of $\boldsymbol{Z}_\nu^{n_\nu}$ is no longer dependent on a correlation function $\boldsymbol{R}$. Instead,

$$\boldsymbol{Z}_\nu^{n_\nu} \,|\, \boldsymbol{\beta}_\nu, \sigma_\nu^2 \sim N_{n_\nu}(\boldsymbol{F}_\nu \boldsymbol{\beta}_\nu, \sigma_\nu^2 \boldsymbol{I}_{n_\nu}), \tag{12}$$

where $\boldsymbol{I}_{n_\nu}$ is the $n_\nu \times n_\nu$ identity matrix.

Under the LLM, the predictive distribution (8) and (9) is simplified; within a region b_ν the predicted value for $Y(\boldsymbol{x}_0)$, is normally distributed with mean and variance

$$\hat{z}(\boldsymbol{x}_0) = \boldsymbol{f}^T(\boldsymbol{x}_0)\tilde{\boldsymbol{\beta}} \tag{13}$$

$$\hat{\sigma}^2(\boldsymbol{x}_0) = \sigma^2 \left[1 + \boldsymbol{f}^T(\boldsymbol{x}_0)\, \boldsymbol{V}_{\tilde{\beta}}\, \boldsymbol{f}(\boldsymbol{x}_0)\right], \tag{14}$$

where $\tilde{\boldsymbol{\beta}}$ is given in (10) and $\boldsymbol{V}_{\tilde{\beta}} = (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F} + \boldsymbol{W}^{-1} / \tau^2)^{-1}$.

All surface modeling in this paper will be done using this model, the treed Gaussian process (TGP) or TGP with jumps to the limiting linear model (TGP LLM), as implemented by Gramacy and Taddy (2008) in their R package `tgp`. [1]

[1] Available on CRAN at www.cran.r-project.org/web/packages/tgp/index.html.

3. Sequential Design for Non-stationary GPs

3.1 Bayesian Adaptive Sampling

Gramacy and his colleagues (Gramacy 2005; Gramacy et al. 2005; Gramacy and Lee 2009) argue that some of the more traditional sequential design methods are not useful in the context of a Bayesian Treed model, given the assumption of stationarity behind these design methods. Instead, they develop a two-stage sequential design method for the situations in which TGP models are used, termed "Bayesian adaptive sampling" (BAS).

The first stage of BAS is selection of candidate points from a sequential treed maximum entropy design. This consists of starting with a relatively dense grid of original candidate points, then building a sequential maximum entropy design within each region $\hat{b}_\nu$ proposed by the current estimated tree structure $\hat{\mathcal{T}}$ ($\nu = 1, \ldots, \hat{\mathcal{N}}$) (Gramacy and Lee 2009). Once the candidate points from each region are selected, these points are fed to the Cohn active learning (ALC) algorithm. The goal in Cohn (1996) is to minimize the expected mean squared error averaged over the entire input space. When a new point $\tilde{\boldsymbol{x}}$ in region ν is selected to sample, there is an associated reduction in predictive variance at the other points in that region (the reduction in prediction variance at points in regions other than ν is assumed to be zero). We want to select $\tilde{\boldsymbol{x}}$ such that the resulting prediction variance is minimized; or rather, such that the global change in prediction variance is maximized. (See Gramacy (2005); Gramacy and Lee (2009) for the computational details, and Seo et al. (2000) for more explanation of the ALC algorithm.)

Gramacy (2005) notes that the ALC is quite costly to compute, especially if the original grid of candidate points is dense and the problem is high-dimensional. This is the thrust of the two-stage BAS algorithm: by reducing the number of candidate points for which the ALC quantity must be computed, we reduce the computing costs while still taking advantage of ALC's searching capabilities.

3.2 Expected Improvement for Global Fit

Lam (2008) proposes a sequential design method that he argues is much simpler computationally than BAS while achieving a global surface fit more efficiently. The expected improvement for global fit (EIGF) criterion is that which chooses the point $\boldsymbol{x}_0$ that maximizes

$$E_{GF}(I) = \left(\hat{z}(\boldsymbol{x}_0) - z(\boldsymbol{x}_*)\right)^2 + \mathrm{Var}(\hat{Z}(\boldsymbol{x}_0)), \tag{15}$$

where $z(\boldsymbol{x}_*)$ is the observed response at the sampled point $\boldsymbol{x}_*$ that is closest in Euclidean distance to $\boldsymbol{x}_0$. Notice this has both a local component and a global component: the first part of (15) will be large when the predicted response has a large absolute increase over its closest sampled neighbor; the second part will be large when there is a large uncertainty associated with $\boldsymbol{x}_0$.

In two dimensions, if the response surface is smooth, the EIGF criterion has the property of picking new points close to the midpoint (in Euclidean distance) of two existing design points. This makes the design nicely space-filling, even in higher dimensions. However, the presence of the variance component keeps the design points from getting stuck in areas with steep gradients.

3.3 Expected Difference in Slopes

The EIGF locates the candidate point $\boldsymbol{x}_0$ that has the largest expected squared vertical distance between the candidate and its closest neighbor $\boldsymbol{x}_*$. Thus, EIGF seeks out areas of the predicted surface with large slope, at least in the local portion of the search (the global

variance component of the EIGF tempers this by searching for areas of high uncertainty). However, in a non-stationary situation, we believe that a high priority should be locating the boundary points of the regions. One way to locate these boundaries is to focus on areas where the slope is *changing rapidly*. A sudden change in slope disagrees with the assumption of a smooth function, and thus would be an indication of a possible non-stationary "break." With this in mind, we developed the Expected Difference of Slopes (EΔM) criterion (Frazier and Notz 2014), which is based on the difference of slopes between three points.

Consider three points $(\boldsymbol{x}_1, Z(\boldsymbol{x}_1)), (\boldsymbol{x}_2, Z(\boldsymbol{x}_2)), (\boldsymbol{x}_3, z_3)$, such that $\boldsymbol{x}_3$ has already been sampled ($(\boldsymbol{x}_3, z_3) \in D$) but $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \boldsymbol{\mathcal{X}}$ have not yet been sampled. Similarly to the EIGF, the choice of the three points $\boldsymbol{x}_1, \boldsymbol{x}_2$ is determined by their Euclidean distance to $\boldsymbol{x}_3$. The difference of the slopes between these three points is

$$\left| \frac{z_3 - Z(\boldsymbol{x}_2)}{\Delta x_{32}} - \frac{Z(\boldsymbol{x}_2) - Z(\boldsymbol{x}_1)}{\Delta x_{21}} \right|, \tag{16}$$

where $\Delta x_{ij} = \| \boldsymbol{x}_i - \boldsymbol{x}_j \|$ is the Euclidean distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. We would like to choose a new point that resides in the region where the square of this quantity is expected to be the largest; thus, we must take the expected value of the square of (16).

$$E[(\text{slope diff})^2] = \frac{1}{\Delta x_{32}^2 \Delta x_{21}^2} \times \Big\{ \left[(\Delta x_{31} \hat{z}(\boldsymbol{x}_2) - \Delta x_{32} \hat{z}(\boldsymbol{x}_1)) - \Delta x_{21} z_3 \right]^2 + \text{Var}\left[\Delta x_{31} Z(\boldsymbol{x}_2) - \Delta x_{32} Z(\boldsymbol{x}_1) \right] \Big\}. \tag{17}$$

The prediction equations for $\hat{z}(\boldsymbol{x}_i)$ and $\hat{\sigma}^2(\boldsymbol{x}_i)$ under the TGP model are given in (8) and (9). $\text{Cov}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ will be estimated by the covariance matrix $\hat{C}$, which is of course based on the estimate of the correlation matrix $\boldsymbol{R}$ and its scale and nugget parameters $(\boldsymbol{d}, g)$. All of these estimates are calculated as part the R package `tgp`.

Equation 17 makes explicit the connection between the EΔM criterion and the EIGF criterion in Section 3.2. The EIGF criterion (15) is made up of a local component and a global component: the global piece is the variance of the predicted response at an untried $\boldsymbol{x}$ value; the local piece is the squared distance between the predicted response at that unsampled $\boldsymbol{x}$ and the known response at its nearest sampled neighbor. The EΔM criterion has a similar form: the global component is the variance of the difference between predicted responses at two untried $\boldsymbol{x}$ values; the local component measures the squared distance between the difference between predicted responses at two untried $\boldsymbol{x}$ values and the known response at the nearest sampled neighbor. Thus, EΔM should balance between exploring locally and globally in its search for boundary points.

3.3.1 Modification to the Expected Difference in Slopes

Although the EΔM is built to search both globally and locally, in practice we find that it tends much more towards a local search. This works well when the true response surface has well-defined boundary points, or when the "interesting" areas of are highly localized (Frazier and Notz 2014). In this way, EΔM is effective when the response surface is truly non-stationary. However, it is less successful when the "interesting" areas are more spread out over the input space, or when there are not distinct boundaries between regions. Empirical investigation indicates that this is mainly due to the coefficient of $E[(\text{slope diff})^2]$, $(\Delta x_{32}^2 \Delta x_{21}^2)^{-1}$. This coefficient tends to overwhelm the global component of (17), and always results in the criterion choosing new points close to already-sampled points. Not surprisingly, this is especially pronounced when the grid of candidate points is fine, and thus the distances between $\boldsymbol{x}_1, \boldsymbol{x}_2$, and $\boldsymbol{x}_3$ are very small.

Thus, we propose a modification of to the EΔM criterion. This modification is called the EΔM.nocoef method, and uses the following criterion in place of Equation (17):

$$\Delta x_{32}^2 \Delta x_{21}^2 E[(\text{slope diff})^2]. \tag{18}$$

Implementation of the modified EΔM method is the same as implementation of the EΔM criterion outlined in Frazier and Notz (2014); see Appendix A for an outline.

4. Simulation Study and Results

Below we will investigate the performance of the four design criteria – BAS, EIGF, EΔM, and EΔM.nocoef – on a 2-dimensional example. Comparing the performance of the BAS to the EIGF in previous works (Lam 2008; Gramacy and Lee 2009) is nearly impossible due to the many differences in how they implemented the simulations, unrelated to their competing design schemes. These differences include different starting designs and sample sizes, different sets of candidate points, and different surrogate models. Perhaps most importantly, Lam uses the traditional computer experiments scenario of deterministic output and an interpolating predictor (3). Gramacy and Lee do include random noise in their data; their predictor (8) is not interpolating and their modeling uses a nugget term in the correlation structure (see Equation 6). The addition of noise to a response function will obviously make it more difficult to model the surface accurately. Below we attempt to compare the BAS, EIGF, EΔM, and EΔM.nocoef methods on a level playing field.

As in Lam (2008), Gramacy and Lee (2009), and Frazier and Notz (2014), the performance of the various criteria will be judged based on empirical root mean square prediction error (ERMSPE) over the grid of N' candidate points $\boldsymbol{XX}$.

$$\text{EMSPE} = \frac{1}{N'} \sum_{i=1}^{N'} \left(\hat{z}(\boldsymbol{x}_i) - z(\boldsymbol{x}_i)\right)^2 \tag{19}$$

$$\text{ERMSPE} = \sqrt{\text{EMSPE}} \tag{20}$$

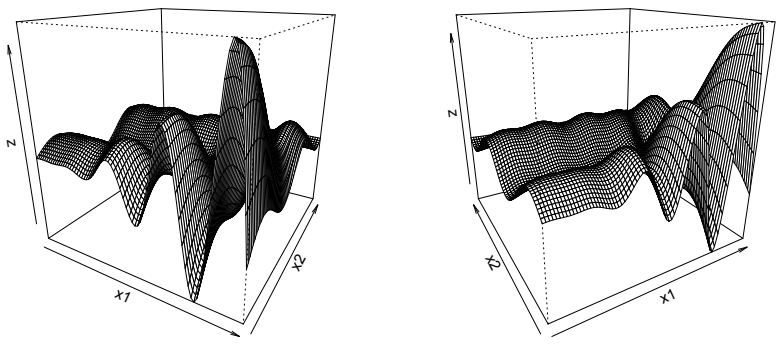

Figure 1: Two views of the 2-dimensional L1 (Lehman) function.

For a two-dimensional test function we turn to a class used in Lehman (2002):

$$z(\boldsymbol{x}) = (x_1 - \theta_1)(x_2 - \theta_2)(x_1 - x_2)\cos(\theta_3 x_1) + 0.1\sin(\theta_3 x_2/2), \tag{21}$$

where $x_1, x_2 \in [0, 1]$; θ_1, θ_2 are independent and uniformly distributed on $(-1, 1)$; and θ_3 is independent of θ_1, θ_2 and uniformly distributed on $(0, 8\pi)$. Lehman reports that this is

a very flexible class of functions, which can have several local optima, and can be smooth or wavy. Consider the Lehman function (21) with a randomly-drawn $\boldsymbol{\theta}$ vector of (0.1350, 0.9710, 23.8251) in Figure 1. This realization, which we will call function L1, has two distinct regions: the gently wavy portion over most of the space; and the higher-amplitude area in $(x_1, x_2) \in [0.5, 1] \times [0, 0.5]$. Performance of the EIGF, BAS, and EΔM sampling methods on this function was investigated in Frazier and Notz (2014).

4.1 Comparison of design methods under Gramacy and Lee's conditions

For consistency with Gramacy et al. (2004), the starting design was a $N_0 = 10$-point Latin hypercube sample chosen from the full list of candidate points $\boldsymbol{XX}$, which was an evenly spaced grid of $N' = 30 \times 30 = 900$ points in $[0, 1] \times [0, 1]$. All the predictions are done using the Bayesian TGP LLM, using a linear mean function, and the separable Gaussian correlation function (6). $N(0, \sigma = 0.001)$ noise was added to the realizations of the response.

The ERMSPE (calculated on the grid of 900 points) from $N = 10$ until $N = 65$ for all four adaptive sampling methods is in Figure 2. In Frazier and Notz (2014), the authors found that between the EIGF, BAS, and EΔM, the EIGF leads to the lowest and most consistent ERMPSE up to $N = 35$, after which the ERMSPE achieved by EΔM drops significantly. After $N = 45$, BAS is comparable to EΔM; after $N = 55$, these three methods are essentially comparable. With the inclusion of the new modified method, EΔM.nocoef, it is clear that EΔM.nocoef leads to the best fit (for all N shown), as it achieves a lower and more consistent ERMSPE than any of the other methods.

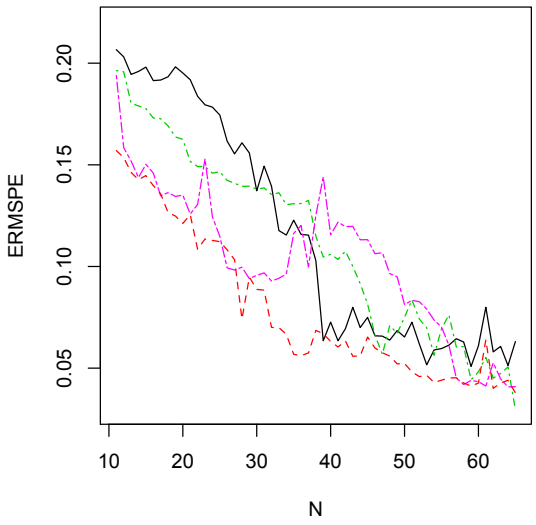

Figure 2: ERMSPE on the 2-d L1 function (noise included) over an evenly-spaced grid of 900 locations, using adaptive sampling methods: BAS (shorter dash-dot green line), EIGF (longer dash-dot violet line), EΔM (solid black line), and EΔM.nocoef (dashed red line).

The adaptively-chosen points selected by $N = 40$ are in Figure 5 (in Appendix B), which give the reader a "snapshot" of the searching behavior of all four methods. BAS starts with a more global search, then eventually focuses where $x_2 < 0.6$. EIGF concentrates very heavily in the southeast quadrant ($x_1 > 0.5, x_2 < 0.5$), and it emulates that area well but as a result fit over the rest of the input space is poor until $N = 45$. EΔM tends to search only around existing design points (after $N = 40$ it concentrates in the southeast quadrant). As

discussed in Section 3.3.1, this method seems to be getting stuck in "bunches" around the input space, sampling many points in three or four distinct areas, while ignoring the rest of the input space. The EΔM.nocoef modification (18) seems to solve the issue in this case; it explores the full input space throughout and only takes a slightly larger sample in that area of interest.

4.2 Comparison of design methods under Lam's conditions

Now we compare these three methods using the same conditions as described in Lam (2008). As in that work, the starting design is now a maximin Latin hypercube sample (Morris and Mitchell 1995) of size $N_0 = 10$, and the candidate points are an evenly spaced grid of $30^2 = 900$ points. Consistent with Lam's modeling conditions, there was no noise was added to the responses from Equation 21 and no nugget was included in the correlation function. We used the TGP (without jumps to the LLM[2]) for surrogate modeling of the response surface, but forced it to be an interpolator (consistent with Lam's GP predictor) However, it should be noted that this allows the predicted response surface to be "treed", unlike in Lam (2008). To be consistent with Lam (2008), we will compare methods once $N = 40$ total samples are reached.

The ERMSPE values calculated on the 900-point grid for $N_0 = 10$ to $N = 40$ are in Figure 3, and Figure 6 (in Appendix B) contains the adaptively-sampled points at $N = 30$ for all four methods. As we saw in the previous section, BAS searches globally, EΔM.nocoef and EIGF take slightly more points in the southeast quadrant while still searching globally, and EΔM concentrates almost exclusively in this quadrant. BAS achieves an extremely poor fit; in fact, the ERMSPE actually increases from $N = 11$ to $N = 40$. It is difficult to ascertain which of the other methods performs best; EΔM has a consistently low ERMSPE, although EIGF and EΔM.nocoef achieve smaller values by $N = 30$.

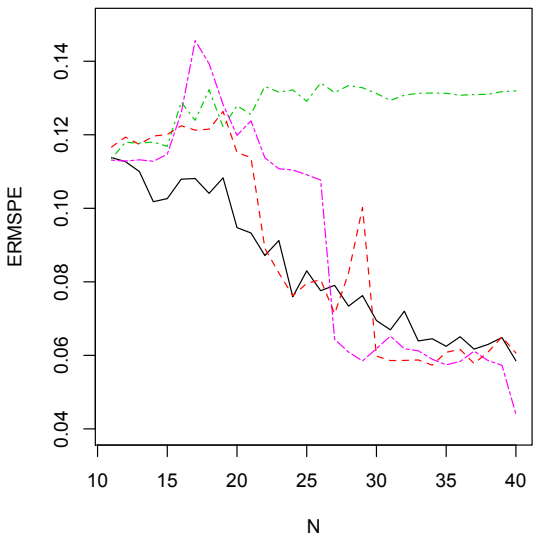

Figure 3: ERMSPE on the 2-d L1 function (no noise) over an evenly-spaced grid of 900 locations, using adaptive sampling methods: BAS (shorter dash-dot green line), EIGF (longer dash-dot violet line), EΔM (solid black line), and EΔM.nocoef (dashed red line).

[2]Due to numerical issues, the limiting linear model does not work when no nugget is included and the predictor is an interpolator

5. Conclusion

Consider the three design schemes: Bayesian adaptive sampling, expected improvement for global fit, and expected difference of slopes investigated in Frazier and Notz (2014). The authors concluded that in general BAS is the most global search; this is perhaps not surprising since it is based in part on entropy, which favors "space-fillingness." EIGF balances between global and local, focusing in "interesting" areas, but not solely these areas. The EΔM method that they introduce in that work is extremely localized. As such, the most successful design scheme for a certain set of data depends on whether searching globally or locally is desired. In that work, the authors found that the EΔM adaptive sampling scheme was generally successful at achieving an efficient global fit of a response surface, as long as the true surface exhibits non-stationary behavior. For surfaces with less localized "interesting" areas, like the L1 function, EΔM did not perform as well because it is *too* localized. The method was not accomplishing the global-local balance that the authors desired when developing it.

Herein, we have proposed a modification to the EΔM criterion that results in a global-local search balance that is more successful than its competitors at efficiently fitting response surfaces with somewhat localized areas of interest, like the L1 surface. It is notable that it is successful under both the traditional (Lam) conditions and the non-deterministic (Gramacy and Lee) conditions. Other work (Frazier 2014) includes many examples showing this modification, EΔM.nocoef, is more efficient than EΔM, BAS, or EIGF on many different types of surfaces, under varying starting designs and modeling conditions.

Planned future work includes a sampling scheme that incorporates both the EΔM and the EIGF criteria, to take advantage of the strengths of both methods. Alternatively, a weighted version of the EΔM, to encourage a more local or global search, is a logical next step.

References

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984), *Classification and Regression Trees*, Wadsworth.

Chipman, H. A., George, E. I., and McCulloch, R. E. (1998), "Bayesian CART model search," *Journal of the American Statistical Association*, 93, 935–948.

— (2002), "Bayesian treed models," *Machine Learning*, 48, 299–320.

Cohn, D. A. (1996), "Neural network exploration using optimal experimental design," *Advances in Neural Information Processing Systems*, 6, 679–686.

Frazier, M. L. (2014), "Adaptive Design for Global Fit of Non-stationary Surfaces," Ph.D. thesis, The Ohio State University.

Frazier, M. L. and Notz, W. I. (2014), "Adaptive design for non-stationary surfaces using changes in slope," Submitted.

Gramacy, R. B. (2005), "Bayesian Treed Gaussian Process Models," Ph.D. thesis, University of California, Santa Cruz.

Gramacy, R. B. and Lee, H. K. H. (2008a), "Bayesian treed Gaussian process models with an application to computer modeling," *Journal of the American Statistical Association*, 103, 1119–1130.

— (2008b), "Gaussian processes and limiting linear models," *Computational Statistics*, 53, 123–136.

— (2009), "Adaptive design and analysis of supercomputer experiments," *Technometrics*, 51, 130–145.

— (2010), "Cases for the nugget in modeling computer experiments," *Statistics and Computing*, 22, 713–722.

Gramacy, R. B., Lee, H. K. H., and Macready, W. G. (2004), "Parameter space exploration with Gaussian process trees," in *Proceedings of the 21st International Conference on Machine Learning*, Omnipress and ACM Digital Library, pp. 353–360.

— (2005), "Adaptive exploration of computer experiment parameter spaces," in *ISBA Bulletin, Applications*, vol. 11, pp. 3 – 6.

Gramacy, R. B. and Taddy, M. A. (2008), "`tgp`: Bayesian treed Gaussian process models," `R` package version 2.1-2.

Lam, C. Q. (2008), "Sequential Adaptive Designs in Computer Experiments for Response Surface Model Fit," Ph.D. thesis, The Ohio State University.

Lehman, J. S. (2002), "Sequential Designs of Computer Experiments for Robust Parameter Design," Ph.D. thesis, The Ohio State University.

Morris, M. and Mitchell, T. J. (1995), "Exploratory designs for computational experiments," *Journal of Statistical Planning and Inference*, 43, 381–402.

Sacks, J., Schiller, S. B., and Welch, W. J. (1989a), "Designs for computer experiments," *Technometrics*, 31, 41–47.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989b), "Design and analysis of computer experiments," *Statistical Science*, 4, 409–423.

Santner, T. J., Williams, B. J., and Notz, W. I. (2003), *The Design and Analysis of Computer Experiments*, Springer-Verlag.

Seo, S., Wallat, M., Graepel, T., and Obermayer, K. (2000), "Gaussian process regression: active data selection and test point rejection," in *Proceedings of the International Joint Conference on Neural Networks*, IEEE, vol. III, pp. 241–146.

A. Implementation of the Modified Expected Difference in Slopes

Implementation of the modified EΔM method is the same as implementation of the EΔM criterion outlined in Frazier and Notz (2014). A relatively dense grid of N' candidate points $\boldsymbol{XX}$ is chosen ahead of time. The prediction model (TGP or TGP LLM) is fit to an initial set of N_0 training points, shown as black circles in Figure 4. A predicted response surface is fit using this initial sample. To find the next point satisfying the EΔM.nocoef criterion, follow this algorithm:

1. Pick a candidate point, $\boldsymbol{x}_1$, from $\boldsymbol{XX}$.

2. Determine which c points in $\boldsymbol{X}$ (the existing design points) are closest in Euclidean distance to the candidate point. (By default, c is twice the dimension of the input space m_X.) So there will be c $\boldsymbol{x}_3$ values.

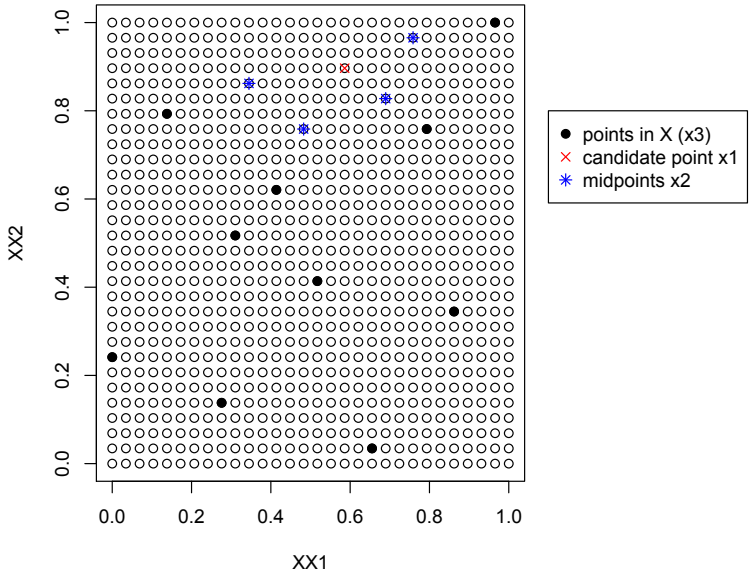

Figure 4: Implementation of the EΔM.coef method with $c = 4$ in two dimensions for a sample candidate point $\boldsymbol{x}_1$ (red x).

3. For each of the c points, find the point in $\boldsymbol{XX}$ that is halfway between this point and $\boldsymbol{x}_1$. These are called the "midpoints," $\boldsymbol{x}_2$.

4. We now have c sets of three points: $\boldsymbol{x}_1$ (which is in $\boldsymbol{XX}$), the closest point $\boldsymbol{x}_3$ (from $\boldsymbol{X}$), and the midpoint between them $\boldsymbol{x}_2$ (also in $\boldsymbol{XX}$). (A visual representation can be seen in Figure 4.)

5. For each of the c sets, calculate the EΔM.nocoef criterion value (using (18) with the predicted surface) for that set.

6. Determine which $\Delta x_{32}^2 \Delta x_{21}^2 E[(\text{slope diff})^2]$ value is largest among the c points; this tells us which direction has largest expected numerical second derivative for that candidate $\boldsymbol{x}_1$.

7. Return to Step 1 and repeat for all $\boldsymbol{x}_1$ in $\boldsymbol{XX}$ that are not already part of the sample $\boldsymbol{X}$.

8. We now have $N' - N$ values of $\max\left\{\Delta x_{32}^2 \Delta x_{21}^2 E[(\text{slope diff})^2]\right\}$. The new sampled point is in the set with the largest $\max\left\{\Delta x_{32}^2 \Delta x_{21}^2 E[(\text{slope diff})^2]\right\}$. Specifically, the new point is $\boldsymbol{x}_2$, the midpoint of that set.

A few points of clarification are needed here. The midpoint $\boldsymbol{x}_2$ is chosen as the new sample point (rather than $\boldsymbol{x}_1$) because the idea is that we want to move *in the direction* of the highest $\Delta x_{32}^2 \Delta x_{21}^2 E[(\text{slope diff})^2]$, and the midpoint is in that direction. As described in Step 3, a midpoint is that which is halfway (in Euclidean distance) between $\boldsymbol{x}_3$ and $\boldsymbol{x}_1$, but it must be in $\boldsymbol{XX}$. Thus, usually $\boldsymbol{x}_2$ is not exactly half the distance, but is slightly closer to either $\boldsymbol{x}_3$ or $\boldsymbol{x}_1$. Finally, existing design points $\boldsymbol{x} \in \boldsymbol{X}$ are barred from being chosen again (even in cases when noise has been added to the responses, so they are not strictly deterministic).

B. Fitted surfaces and point selection for examples in Section 4

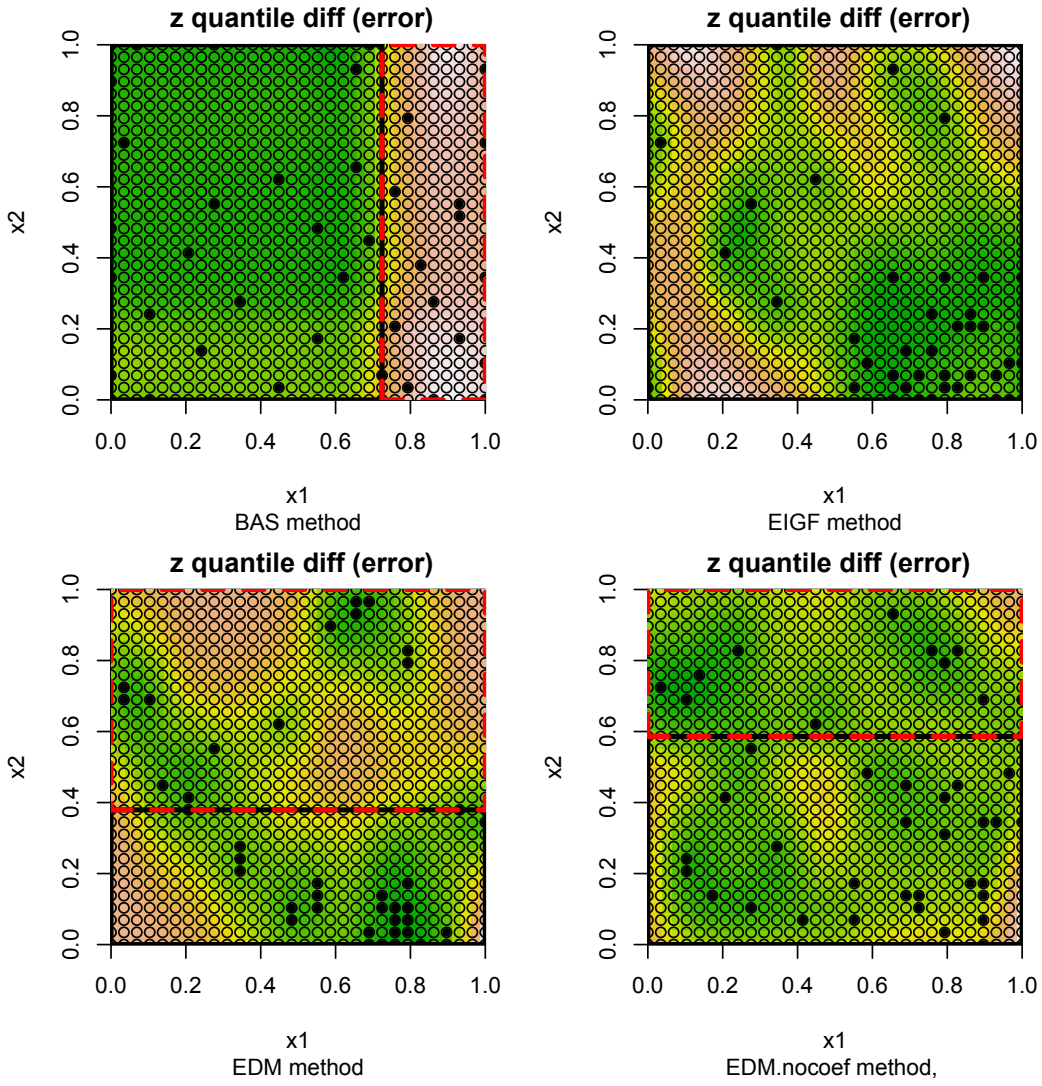

Figure 5: Point selection by $N = 40$ for the four design methods under Gramacy and Lee's conditions for the 2-d L1 data (noise included in responses). Plots are of posterior predictive variance (white = highest variance, through green = lowest variance), with tree $\hat{\mathcal{T}}$ (boxes), sampled points $\boldsymbol{X}$ (dots), and candidate points $\boldsymbol{XX}$ (circles).

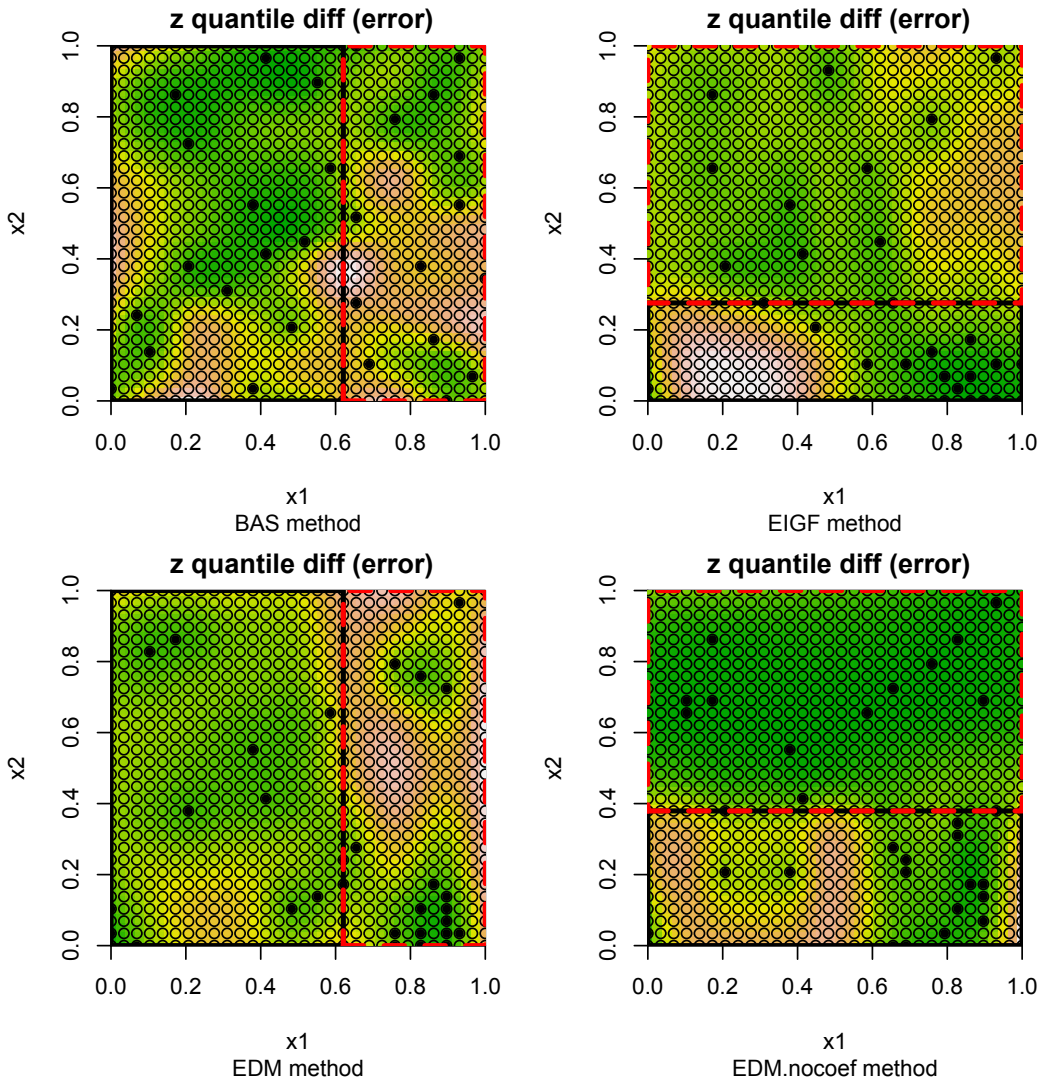

Figure 6: Point selection by $N = 30$ for the four design methods under Lam's conditions for the 2-d L1 data (no noise in responses). Plots are of posterior predictive variance (white = highest variance, through green = lowest variance), with tree $\hat{\mathcal{T}}$ (boxes), sampled points $\boldsymbol{X}$ (dots), and candidate points $\boldsymbol{XX}$ (circles).