

Submitted to *Operations Research*
manuscript (Please, provide the manuscript number!)

Authors are encouraged to submit new papers to INFORMS journals by means of a style file template, which includes the journal title. However, use of a template does not certify that the paper has been accepted for publication in the named journal. INFORMS journal templates are for the exclusive purpose of submitting to an INFORMS journal and should not be used to distribute the papers in print or online or to submit the papers to another publication.

Gaussian Markov Random Fields for Discrete Optimization via Simulation: Framework and Algorithms

Peter Salemi

The MITRE Corporation, McLean, VA 22102, psalemi@mitre.org

Eunhye Song

Department of Industrial and Manufacturing Engineering, The Pennsylvania State University, University Park, PA 16802,
eus358@psu.edu

Barry L. Nelson, Jeremy Staum

Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL 60208,
nelsonb@northwestern.edu, j-staum@northwestern.edu

We consider optimizing the expected value of some performance measure of a dynamic stochastic simulation with a statistical guarantee for optimality when the decision variables are *discrete*, in particular integer-ordered, the number of feasible solutions is large, and the model execution is too slow to simulate even a substantial fraction of them. Our goal is to create algorithms that stop searching when they can provide inference about the remaining optimality gap similar to the correct-selection guarantee of ranking and selection when it simulates all solutions. Further, our algorithm remains competitive with fixed-budget algorithms that search efficiently but do not provide such inference. To accomplish this we learn and exploit spatial relationships among the decision variables and objective function values using a Gaussian Markov random field (GMRF). Gaussian random fields on continuous domains are already used in deterministic and stochastic optimization because they facilitate the computation of measures like expected improvement that balance exploration and exploitation. We show that GMRFs are particularly well suited to the discrete-decision-variable problem, both from a modeling and a computational perspective. Specifically, GMRFs permit the definition of a sensible neighborhood structure, and they are defined by their precision matrices which can be constructed to be sparse. Using this framework we create both single and multiresolution algorithms, prove the asymptotic convergence of both, and evaluate their finite-time performance empirically.

Key words: large-scale discrete optimization via simulation, inferential optimization, Gaussian Markov Random Fields

History: First submitted on May 2016; revisions submitted on March 2017, July 2017, October 2017.

1. Introduction

Optimization plays a central role in operations research. For many practical stochastic optimization problems the objective function cannot be evaluated exactly and instead must be estimated using simulation. Such problems require optimization via simulation (OvS). OvS problems in which the decision variables can only assume discrete values are called discrete optimization via simulation (DOvS) problems, and they present particular challenges. In this paper, we focus on DOvS with integer-ordered decision variables.

Due to its practical importance, DOvS has been an active area of research for many years. Remarkable theoretical and practical success has been achieved using exhaustive search algorithms known collectively as ranking and selection (R&S). R&S simulates all feasible solutions and terminates with either a guaranteed probability of correct selection (frequentist) or posterior assessment of the relative quality of the selected solution (Bayesian); see Kim (2013) and Frazier (2012), respectively, for surveys. These guarantees are comforting because even when all feasible solutions are simulated it is not possible to promise optimality in finite time except in very special cases.

When the number of feasible solutions is too large to exhaust, then many DOvS algorithms employ adaptive random search (ARS). These algorithms attempt to leverage, in an informal way, anticipated spatial structure such as “good feasible solutions tend to be clustered.” ARS algorithms are usually proven to be either globally or locally convergent as the simulation effort increases without bound. See, for example, the survey in Nelson (2010).

In this paper we provide a DOvS framework to formally learn spatial relationships, to balance exploration and exploitation, and to facilitate a stopping criterion that considers the uncertainty at feasible solutions that have and have not been simulated for large-scale problems. Loosely speaking, we want efficient search with R&S-like conclusions when we stop. We will represent the objective-function response surface as a Gaussian random field (GRF), because GRFs support assessments of the benefit of expending simulation effort in various ways, and statistical inference on the potential of unseen solutions. GRF-based optimization methods were introduced for deterministic computer experiments in Jones et al. (1998), followed by Huang et al. (2006) which considered computer experiments with noisy output. Recently, Xie et al. (2016) and Quan et al. (2013) created GRF-based optimization methods for DOvS. All of these methods use a GRF with a continuous domain and then project the problem into the discrete setting. Unfortunately, the standard covariance functions for continuous domains lead to a dense covariance matrix, and the inverse of this matrix is used for fitting the GRF and generating inference that guides the search. Therefore, methods based on a continuous covariance function can encounter difficulties for large-scale problems with many feasible solutions due to slow and ill-conditioned numerical computations, and (more importantly for this paper) can give misleading inference upon stopping.

A central contribution of this paper is creating a GRF that is appropriate for the DOvS problem. Specifically, we model the objective function values at the feasible solutions as a realization of a discrete Gaussian Markov random field (GMRF) (Rue and Held 2005). The Markov structure of GMRFs is intuitive for problems in operations research (Salemi et al. 2013). For example, if we are interested in predicting the value of the objective function at a feasible solution, then the values of the objective function at the feasible solutions in a neighborhood of it would often be sufficient; other feasible solutions would provide little additional information. This type of structure can be represented in a GMRF, and the neighborhood is user selected. Further, GMRFs can be defined on a lattice so the use of GMRFs in DOvS problems is more natural than using a GRF with a continuous domain. Most importantly, the Markov structure lends itself to efficient and numerically stable calculations for large-scale problems. The dependence in a GMRF is defined by its precision matrix, which is the inverse of the covariance matrix. Using the Markov structure of GMRFs, the precision matrix can be constructed to be sparse. Thus, we can use sparse-matrix methods to calculate expressions that involve the precision matrix.

Although our GMRF framework facilitates the use of sparse-matrix methods that support algorithms for large numbers of feasible solutions, there will nevertheless be a computational limit on the size of problem we can handle directly. *A second contribution of this paper is to extend the GMRF approach to a multiresolution framework that can be used to solve DOvS problems with vast numbers of solutions.* The multiresolution framework exploits a region-level GMRF to learn about the quality of disjoint regions of the solution space, and solution-level GMRFs to learn about the quality of individual feasible solutions within each region.

A third contribution is to demonstrate that expected improvement (EI) combined with our framework can provide effective inference for terminating the search when the estimated optimality gap is small enough. Jones et al. (1998) first introduced EI for optimization problems with deterministic computer experiments. In their setting the objective function can be observed without noise, so their EI criterion does not incorporate the uncertainty in the output from a stochastic simulation. We show how to correctly calculate EI for a GMRF+noise, which turns out to be similar to Williams et al. (2000). To distinguish our result from EI for deterministic experiments, we call it complete expected improvement (CEI); it is “complete” in the sense that it incorporates the uncertainty in stochastic simulation by treating the value at the current sample-best solution as a random variable, just like the values of the feasible solutions being considered to improve it.

The paper presents both a GMRF framework within which algorithms can be designed, and specific solution-level and multiresolution DOvS algorithms. Both algorithms are shown to converge to a globally optimal solution as simulation effort increases under very mild conditions (essentially finite variance). More importantly, however, these algorithms can self-terminate well short of infinite

effort with statistical assurance about the optimality gap; this inference, which depends on the GMRF assumptions, is a central feature of our work.

In the next section we present the GMRF framework, which includes defining GMRFs and explaining how they can represent a DOvS problem. Section 3 details a specific solution-level DOvS algorithm, including parameter estimation and sparse-matrix calculation techniques. For problems with vast numbers of feasible solutions, we extend the framework to a multiresolution algorithm in Section 4. Results of numerical experiments are found in Section 5.

2. A Framework for DOvS using GMRFs

A GMRF is a non-degenerate multivariate Gaussian random vector $\mathbb{Y} = (\mathbb{Y}_1, \mathbb{Y}_2, \dots, \mathbb{Y}_n)^\top$ that is associated with an undirected and labeled graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where \mathcal{V} denotes the set of nodes and \mathcal{E} denotes the set of edges; see Rue and Held (2005). Each node in \mathcal{V} is associated with a unique element of \mathbb{Y} . Two nodes in the graph are called neighbors if they are connected by an edge. If we denote the mean vector and precision matrix of \mathbb{Y} by $\boldsymbol{\mu}$ and \mathbf{Q} , respectively, then we can write the probability density function of the GMRF as

$$f(\mathbf{y} \mid \boldsymbol{\mu}, \mathbf{Q}) = (2\pi)^{-n/2} |\mathbf{Q}|^{1/2} \exp\left(-\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^\top \mathbf{Q}(\mathbf{y} - \boldsymbol{\mu})\right),$$

where the positive-definite precision matrix \mathbf{Q} is the inverse of the covariance matrix. The diagonal entries of the precision matrix are the conditional precisions $\text{Prec}(\mathbb{Y}_i \mid \mathbb{Y}_{\mathcal{V} \setminus \{i\}}) = Q_{ii}$, where $\mathbb{Y}_{\mathcal{V} \setminus \{i\}}$ is the vector of values of the GMRF observed only at the nodes $\mathcal{V} \setminus \{i\}$. The scalar precision Q_{ii} is the reciprocal of the conditional variance. The off-diagonal elements are proportional to conditional correlations; specifically $\text{Corr}(\mathbb{Y}_i, \mathbb{Y}_j \mid \mathbb{Y}_{\mathcal{V} \setminus \{i,j\}}) = -Q_{ij} / \sqrt{Q_{ii}Q_{jj}}$, where $\mathbb{Y}_{\mathcal{V} \setminus \{i,j\}}$ is the vector of values of the GMRF observed only at the nodes $\mathcal{V} \setminus \{i,j\}$. The graph \mathcal{G} determines the non-zero pattern of the precision matrix \mathbf{Q} , and vice versa, since for a GMRF $Q_{ij} \neq 0$ if and only if $\{i,j\} \in \mathcal{E}$. Thus, the precision matrix will be sparse if the set of edges is small, and vice versa. GMRFs are ‘‘Markov’’ because they possess the local Markov property:

$$\mathbb{Y}_i \perp \mathbb{Y}_{\mathcal{V} \setminus \{i, \mathcal{N}(i)\}} \mid \mathbb{Y}_{\mathcal{N}(i)} \quad \text{for every } i \in \mathcal{V},$$

where $\mathcal{N}(i)$ is the set of neighbors of node i in \mathcal{G} ; that is, $\mathcal{N}(i) = \{j : \{i,j\} \in \mathcal{E}\}$.

To better understand the local Markov property, consider the left-hand graph \mathcal{G} in Figure 1, which represents a portion of a feasible region where the nodes correspond to solutions $\mathbf{x}^\top = (x_1, x_2)$ with constraint $x_1 < x_2$. If we observe the values of the GMRF at $\mathbf{x} = (0, 3)$, $(1, 4)$, $(2, 3)$ and $(1, 2)$, then our prediction of the value of the GMRF at $(1, 3)$, conditional on that information, does not depend on the GMRF at any other lattice point in the graph. The Markov property does *not* imply that nodes far away from one another are independent, but rather that if we know the value of the

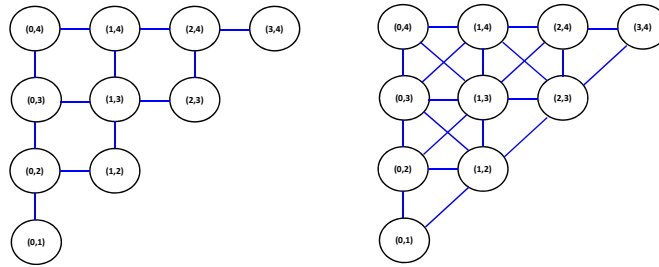


Figure 1 A portion of two graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ corresponding to a subset of the integer lattice \mathbb{Z}^2 with the constraint $x_1 < x_2$ and illustrating the neighborhoods $\mathcal{N}(\mathbf{x})$ (left) and $\mathcal{N}^+(\mathbf{x})$ (right).

GMRF at nodes close by, then we can ignore nodes farther away conditional on those values. To completely specify a GMRF we only need to specify the mean $\boldsymbol{\mu}$ and the precision matrix \mathbf{Q} , whose non-zero pattern is associated with the structure of the graph \mathcal{G} . *A problem-appropriate GRF is important since our stopping inference will be with respect to it.*

2.1. A Precision Matrix for DOvS

In a DOvS problem with integer-ordered decision variables, the feasible region \mathcal{X} is a finite subset of the d -dimensional integer lattice \mathbb{Z}^d . Thus, the straightforward construction of the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ starts with defining the nodes of the graph \mathcal{V} to be \mathcal{X} . To finish the construction of \mathcal{G} , we must specify the neighborhood structure for any given node/solution.

Two sensible neighborhoods for a solution $\mathbf{x} \in \mathcal{X}$ are $\mathcal{N}(\mathbf{x}) = \{\mathbf{x}' \in \mathcal{X} : \|\mathbf{x} - \mathbf{x}'\|_2 = 1\}$, which has up to $2d$ neighbors in d dimensions, and $\mathcal{N}^+(\mathbf{x}) = \{\mathbf{x}' \in \mathcal{X} : \|\mathbf{x} - \mathbf{x}'\|_\infty = 1\}$, which has up to $3^d - 1$ neighbors in d dimensions. Both are illustrated in Figure 1 for $d = 2$.

Since we are particularly interested in DOvS problems with large feasible regions, and because (roughly speaking) the more sparse \mathbf{Q} is the larger the DOvS problems we can solve, our algorithm will use the neighborhood structure $\mathcal{N}(\mathbf{x})$. For this neighborhood the fraction of non-zero entries in the precision matrix \mathbf{Q} is bounded above by $(2d + 1)/|\mathcal{X}|$, which is very small for large problems. Although we assume this specific neighborhood for efficient computations, our GMRF framework allows any neighborhood structure and our convergence proofs do not depend on it.

Defining the graph \mathcal{G} , and thus the non-zero pattern of the precision matrix, does not specify the values of the non-zero entries of \mathbf{Q} . A standard approach is to have the entries of \mathbf{Q} given by a function $p(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector of parameters; i.e., $Q_{ij} \triangleq p(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta})$. For the neighborhood $\mathcal{N}(\mathbf{x})$, we propose $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \dots, \theta_d)^\top$ and

$$p(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}) = \begin{cases} \theta_0, & \text{if } \mathbf{x} = \mathbf{x}' \\ -\theta_0 \theta_j, & \text{if } |\mathbf{x} - \mathbf{x}'| = \mathbf{e}_j \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

for $\mathbf{x}, \mathbf{x}' \in \mathbb{Z}^d$, where \mathbf{e}_j is the j th standard basis vector. Recall that Q_{ii} is the conditional precision of solution i , and $-Q_{ij}/\sqrt{Q_{ii}Q_{jj}}$ is the conditional correlation between solutions i and j . Thus, θ_0 is the conditional precision of each solution, and θ_j is the conditional correlation between solutions that differ by 1 in the j th coordinate direction. That the conditional correlations can depend on the coordinate direction is important to allow for response surfaces that change more rapidly in one direction as compared to another.

Since the conditional precisions must be positive, it follows that $\theta_0 > 0$. We also want neighbors to have non-negative conditional correlations, so we restrict the values of $\theta_1, \theta_2, \dots, \theta_d$ to be non-negative as well. Furthermore, we need $\theta_j \leq 1$ for $j = 1, 2, \dots, d$, since the conditional correlations must be less than one. Lastly, \mathbf{Q} should be positive-definite. Exploiting these restrictions, we calculate the maximum likelihood estimates of the parameters; see EC.2 in the e-companion to this paper. With this parameterization and these restrictions, \mathbf{Q} is a non-singular M -matrix so its inverse is nonnegative, i.e., $[\mathbf{Q}^{-1}]_{ij} \geq 0$ for all i and j . In other words, there are no negative unconditional correlations among nodes in the GMRF, which is a property that makes sense in many DOvS problems.

In summary, this parameterization is compact ($d+1$ parameters), captures key features we expect in DOvS problems, and all but θ_0 are upper and lower bounded which facilitates estimation. Notice that even though we construct the precision matrix $\mathbf{Q}(\boldsymbol{\theta})$ to be sparse, the covariance matrix it implies, $\mathbf{Q}(\boldsymbol{\theta})^{-1}$, will typically be dense, as it should be.

2.2. A GMRF Representation of DOvS

We consider the DOvS problem, minimize $y(\mathbf{x}) \triangleq \mathbb{E}[Y(\mathbf{x})]$ subject to $\mathbf{x} \in \mathcal{X}$, where \mathcal{X} is a finite subset of the d -dimensional integer lattice \mathbb{Z}^d and $n \triangleq |\mathcal{X}|$. The distribution of the random variable $Y(\mathbf{x})$, as a function of the feasible solution \mathbf{x} is unknown. However, the expectation $y(\mathbf{x}) \triangleq \mathbb{E}[Y(\mathbf{x})]$ can be estimated via stochastic simulation. Formally, we are able to observe

$$Y_j(\mathbf{x}) = y(\mathbf{x}) + \epsilon_j(\mathbf{x}) \quad (2)$$

for any feasible solution \mathbf{x} on replication $j = 1, 2, \dots$, where $\{\epsilon_j(\mathbf{x})\}$ are i.i.d. with mean 0 and finite variance that may depend on \mathbf{x} . Below we will also assume that the $\epsilon_j(\mathbf{x})$ are normally distributed. While we assume the ability to make replications in this paper, a single-run steady-state simulation setting could be accommodated with, say, batch means playing the role of $Y_j(\mathbf{x})$.

Let \mathbf{y} denote the vector of objective function values $(y(\mathbf{x}_1), y(\mathbf{x}_2), \dots, y(\mathbf{x}_n))^\top$. Of course \mathbf{y} is unknown, so we model it as a realization of the GMRF

$$\mathbb{Y} \triangleq (\mathbb{Y}(\mathbf{x}_1), \mathbb{Y}(\mathbf{x}_2), \dots, \mathbb{Y}(\mathbf{x}_n))^\top \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})^{-1}) \quad (3)$$

with mean vector $\boldsymbol{\mu}$ and precision matrix $\mathbf{Q}(\boldsymbol{\theta})$, as defined above.

Similar to virtually all methods for optimization based on GRFs, the essence of our approach is to use the conditional distribution of the GMRF after having simulated some feasible solutions as the guidance and inference engine for our search. Throughout the paper we use the term “design point” to refer to a feasible solution \mathbf{x} that has been simulated for any number of replications, and we use the terms “point” and “feasible solution” interchangeably. Therefore, we are interested in the conditional distribution of the GMRF given simulation output at the design points.

Let $\Xi_2 \subseteq \mathcal{X}$ denote the current set of design points and partition \mathcal{X} into the two disjoint sets Ξ_2 and $\Xi_1 = \mathcal{X} \setminus \Xi_2$. Thus, Ξ_2 is the set of feasible solutions that have been simulated, and Ξ_1 is the set of feasible solutions that have not. For simplicity, we use “1” as a subscript to denote quantities associated with the set Ξ_1 and “2” as a subscript to denote quantities associated with the set Ξ_2 . For instance, $n_1 = |\Xi_1|$ and $n_2 = |\Xi_2|$ are the numbers of solutions in each set.

Using these disjoint sets, we can partition the vectors $\mathbf{y}, \mathbb{Y}, \boldsymbol{\mu}$, and the precision matrix $\mathbf{Q}(\boldsymbol{\theta})$, and rewrite expression (3) as

$$\begin{pmatrix} \mathbb{Y}_1 \\ \mathbb{Y}_2 \end{pmatrix} \sim \mathcal{N} \left(\begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \begin{pmatrix} \mathbf{Q}_{11}(\boldsymbol{\theta}) & \mathbf{Q}_{12}(\boldsymbol{\theta}) \\ \mathbf{Q}_{12}(\boldsymbol{\theta})^\top & \mathbf{Q}_{22}(\boldsymbol{\theta}) \end{pmatrix}^{-1} \right).$$

Let $\bar{\mathbb{Y}}_2$ be the vector of sample means of the simulation output at the design points. Consistent with the output model (2), we represent $\bar{\mathbb{Y}}_2$ as a realization of the GMRF $\mathbb{Y}_2^\epsilon = \mathbb{Y}_2 + \boldsymbol{\epsilon}$, with \mathbb{Y}_2 and $\boldsymbol{\epsilon}$ independent, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\vec{\mathbf{0}}_{n_2 \times 1}, \mathbf{Q}_\epsilon^{-1})$, where \mathbf{Q}_ϵ is the intrinsic precision matrix of the noise inherent to the stochastic simulation output $\bar{\mathbb{Y}}_2$. When we simulate design points independently, \mathbf{Q}_ϵ is a diagonal matrix, whereas when we simulate with common random numbers (CRN), \mathbf{Q}_ϵ is a dense matrix. The values in \mathbf{Q}_ϵ also depend on how many replications have been averaged, which need not be the same at all design points. In EC.1 we prove the following:

THEOREM 1. The conditional distribution of \mathbb{Y} given $\mathbb{Y}_2^\epsilon = \bar{\mathbb{Y}}_2$ is

$$\mathcal{N} \left(\begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix} + \bar{\mathbf{Q}}(\boldsymbol{\theta})^{-1} \begin{pmatrix} \vec{\mathbf{0}}_{n_1 \times 1} \\ \mathbf{Q}_\epsilon(\bar{\mathbb{Y}}_2 - \boldsymbol{\mu}_2) \end{pmatrix}, \bar{\mathbf{Q}}(\boldsymbol{\theta})^{-1} \right), \quad (4)$$

where

$$\bar{\mathbf{Q}}(\boldsymbol{\theta}) \triangleq \begin{pmatrix} \mathbf{Q}_{11}(\boldsymbol{\theta}) & \mathbf{Q}_{12}(\boldsymbol{\theta}) \\ \mathbf{Q}_{12}(\boldsymbol{\theta})^\top & \mathbf{Q}_{22}(\boldsymbol{\theta}) \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{n_1 \times n_1} & \mathbf{0}_{n_1 \times n_2} \\ \mathbf{0}_{n_1 \times n_2}^\top & \mathbf{Q}_\epsilon \end{pmatrix}$$

is the conditional precision matrix, and $\mathbf{0}_{n_i \times n_j}$ is the $n_i \times n_j$ matrix of zeros.

Remark: The sparsity of $\bar{\mathbf{Q}}(\boldsymbol{\theta})$ is inherited from the sparsity of $\mathbf{Q}(\boldsymbol{\theta})$ and \mathbf{Q}_ϵ . Furthermore, as will be discussed in Section 3.2, we can avoid direct inversion of $\bar{\mathbf{Q}}(\boldsymbol{\theta})$ using sparse-matrix methods.

2.3. Indifference-Zone R&S and Expected Improvement

Our goal is to create a framework for DOvS algorithms that are self-stopping with inference similar in spirit to that of indifference-zone (IZ) R&S but only simulating a small fraction of the feasible solutions. This is clearly impossible without a useful model of the unknown objective function, and we propose a GMRF—with appropriate neighborhood structure and parameterization—that is observed with noise.

Let \mathbf{x}^* denote the unknown best of the n feasible solutions, which we assume for simplicity of exposition is unique: $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x}} y(\mathbf{x})$. Let $\tilde{\mathbf{x}}$ be the selected solution by whatever method. Two standard objectives in IZ R&S are attaining a desired probability of correct selection (PCS) $\Pr \{ \tilde{\mathbf{x}} = \mathbf{x}^* \mid \min_{\mathbf{x} \neq \mathbf{x}^*} y(\mathbf{x}) - y(\mathbf{x}^*) \geq \delta \} \geq 1 - \alpha$, and attaining a desired probability of a good selection (PGS) $\Pr \{ y(\tilde{\mathbf{x}}) - y(\mathbf{x}^*) \leq \delta \} \geq 1 - \alpha$. Both statements refer to properties of the R&S procedure averaged over the stochastic (intrinsic) simulation noise. Our approach is more akin to PGS than PCS in that we make no assumption about the gap between the best and next-best feasible solution, but we do require the user to provide a smallest practically significant difference δ measured in the same units as the response $y(\mathbf{x})$.

Using our notation, the standard definition of “EI” for a feasible solution \mathbf{x} is

$$\text{EI}^t(\mathbf{x}) = \text{E} [\max\{0, \mathbb{Y}(\tilde{\mathbf{x}}^t) - \mathbb{Y}(\mathbf{x})\} | \bar{\mathcal{Y}}_2^t] \quad (5)$$

where we have appended a superscript t to indicate quantities available at the end of the t^{th} iteration of an algorithm. In the original EI criterion for deterministic computer experiments of Jones et al. (1998), $\mathbb{Y}(\tilde{\mathbf{x}}^t) = y(\tilde{\mathbf{x}}^t)$, the true value of the best solution simulated through iteration t (which is unambiguously known); and the only uncertainty in $\mathbb{Y}(\mathbf{x})$ is due to the conditional distribution of the GRF $\mathbb{Y}(\mathbf{x})$. Clearly this is inappropriate when the output is from a stochastic simulation, so there have been attempts to account for noise, most prominently MNEI (Quan et al. 2013) and Augmented EI (Huang et al. 2006). However, these attempts are incomplete in the sense that do not average over all sources of uncertainty.

To distinguish it from these other versions, we use the term *complete expected improvement* (CEI) for the mathematically correct evaluation of (5) under our GMRF+noise model. CEI averages over both the full conditional joint distribution of the GMRF $\mathbb{Y}(\mathbf{x})$, and the simulation noise, given the observed simulation outputs up through the current iteration. It is, in this sense, “complete,” but it is still EI. Because we intend to use EI for stopping inference it is important to evaluate it correctly. See EC.8 for further analysis of how CEI is different from deterministic EI.

Given the simulation output $\bar{\mathcal{Y}}_2^t$, the conditional joint distribution of $\mathbb{Y}(\tilde{\mathbf{x}}^t)$ and $\mathbb{Y}(\mathbf{x})$ is bivariate normal with parameters from the rows and columns of Equation (4) corresponding to $\tilde{\mathbf{x}}^t$ and \mathbf{x} .

Denote the conditional means by $M^t(\tilde{\mathbf{x}}^t)$ and $M^t(\mathbf{x})$, the conditional variances by $V^t(\tilde{\mathbf{x}}^t)$ and $V^t(\mathbf{x})$, and the conditional correlation by $\rho^t(\tilde{\mathbf{x}}^t, \mathbf{x})$. Let

$$V^t(\tilde{\mathbf{x}}^t, \mathbf{x}) \triangleq V^t(\tilde{\mathbf{x}}^t) + V^t(\mathbf{x}) - 2\rho^t(\tilde{\mathbf{x}}^t, \mathbf{x})\sqrt{V^t(\tilde{\mathbf{x}}^t)V^t(\mathbf{x})}$$

be the conditional variance of the difference $\mathbb{Y}(\tilde{\mathbf{x}}^t) - \mathbb{Y}(\mathbf{x})$. Then the CEI of solution \mathbf{x} , $\text{CEI}(\mathbf{x})$, is

$$\text{CEI}^t(\mathbf{x}) = (M^t(\tilde{\mathbf{x}}^t) - M^t(\mathbf{x})) \Phi\left(\frac{M^t(\tilde{\mathbf{x}}^t) - M^t(\mathbf{x})}{\sqrt{V^t(\tilde{\mathbf{x}}^t, \mathbf{x})}}\right) + \sqrt{V^t(\tilde{\mathbf{x}}^t, \mathbf{x})} \phi\left(\frac{M^t(\tilde{\mathbf{x}}^t) - M^t(\mathbf{x})}{\sqrt{V^t(\tilde{\mathbf{x}}^t, \mathbf{x})}}\right), \quad (6)$$

where ϕ and Φ are the density and cumulative distribution function, respectively, of a standard normal random variable. We used the expression for EI in Jones et al. (1998) to derive (6) by noting that conditional on \bar{y}_2^t the difference $\mathbb{Y}(\tilde{\mathbf{x}}^t) - \mathbb{Y}(\mathbf{x})$ is a Gaussian random variable with mean $M^t(\tilde{\mathbf{x}}^t) - M^t(\mathbf{x})$ and variance $V^t(\tilde{\mathbf{x}}^t, \mathbf{x})$.

We propose stopping when

$$\max_{\mathbf{x} \neq \tilde{\mathbf{x}}^t} \text{CEI}^t(\mathbf{x}) \leq \delta \quad (7)$$

where $\tilde{\mathbf{x}}^t$ is the sample best solution (smallest sample mean) through iteration t , and simulating the feasible solution with the largest $\text{CEI}(\mathbf{x})$ when it is not. This stopping criterion has been suggested in other EI-based methods, such as Huang et al. (2006). Here is why we adopt it:

- We chose this definition of $\tilde{\mathbf{x}}^t$ because it is the solution we would actually select if we stopped; $\tilde{\mathbf{x}}^t = \min_{\mathbf{x}} M^t(\mathbf{x})$ is another viable choice, but we consider $\tilde{\mathbf{x}}^t = \min_{\mathbf{x}} \bar{Y}(\mathbf{x})$ more reliable in that our GMRF, like any GRF, is a useful model, but $\bar{Y}(\mathbf{x})$ is a direct estimator free of the estimated parameters of GMRF.
- PCS/PGS provide guarantees averaged over the stochastic simulation noise; $\text{CEI}^t(\mathbf{x})$ goes further by averaging over the remaining uncertainty about the values of $y(\mathbf{x})$, both simulated and not simulated, as represented by the conditional distribution of the GMRF. Both relate to the optimality gap between the selected and best solution. Thus, EI is a logical extension when not all n solutions have been simulated, as they are in R&S.
- Since EI evaluates improvement of each solution \mathbf{x} , we take the maximum of EIs to decide whether to stop. One can choose to stop when improvement from the smallest possible response value given by the GMRF is less than δ , i.e, $E[\max\{0, \mathbb{Y}(\tilde{\mathbf{x}}^t) - \min_{\mathbf{x} \in \mathcal{X}} \mathbb{Y}(\mathbf{x})\} | \bar{y}_2^t] \leq \delta$. However, this criterion can be overly conservative for large \mathcal{X} .

While EI is consistent with IZ R&S, other measures could be employed. For instance, the knowledge gradient (KG) chooses the next solution to simulate by looking forward to the impact on the estimated value of the optimal solution via the predictive distribution obtained from allocating one

or more observations to \mathbf{x} . In our notation, assuming a fixed number of replications per iteration, and formulated so that the KG is positive like EI,

$$\text{KG}^t(\mathbf{x}) = \min_{\mathbf{x}'} \mathbb{E} [\mathbb{Y}(\mathbf{x}') | \mathcal{Y}_2^t] - \mathbb{E} \left[\min_{\mathbf{x}'} \mathbb{Y}(\mathbf{x}') | \mathcal{Y}_2^t, \mathbf{x}^{t+1} = \mathbf{x} \right]. \quad (8)$$

In words, *EI* estimates the optimality gap given the current data, while *KG* seeks the steepest descent direction to improve the optimal solution in the next iteration. One could stop a search when all of the *KG*'s are small implying that little additional improvement is possible, i.e., $\max_{\mathbf{x}} \text{KG}(\mathbf{x}) \leq \epsilon$ for some small ϵ . However, it is not appropriate to choose $\epsilon = \delta$ as the improvement from a single iteration is small, although the optimality gap of the current optimal solution is bigger than δ . *KG*-based stopping is more natural when there is a cost to continued simulation and one wants to know if it is worth the effort.

3. A Solution-Level Algorithm

In this section we present an instance of our framework, the Gaussian Markov improvement algorithm (GMIA). We refer to it as a “solution-level algorithm” because the nodes of the GMRF correspond to individual feasible solutions \mathbf{x} , and we employ the neighborhood structure $\mathcal{N}(\mathbf{x})$ described in Section 2.1. An outline is given in Section 3.1. We then describe efficiently calculating the conditional distribution (4) using sparse matrix techniques in Section 3.2. GMIA represents specific choices that we have made and found effective, but additional enhancements, other neighborhood structures, and fine-tuning using known properties of a specific problem, are certainly possible. Even using sparse matrix techniques, $|\mathcal{X}|$ can be so large that direct application of GMIA is computationally impossible. To address such cases we describe a multiresolution algorithm in Section 4 that exploits a region-level GMRF as well as solution-level GMRFs; to the best of our knowledge this is the first such algorithm.

3.1. GMIA Outline

GMIA utilizes CEI to search for solutions to simulate, and to stop when it drops below a specified threshold δ .

GMIA

0. Generate a set of k_s design points. Simulate r replications for each design point and use the simulation output to calculate the maximum likelihood estimates (MLEs) for the GMRF parameters using the method in EC.2.
1. Let $\tilde{\mathbf{x}}$, the current sample-best solution, be the design point with the smallest sample mean.
2. Calculate the CEI with respect to $\tilde{\mathbf{x}}$, defined in Section 2.3, for each candidate feasible solution. If $\max_{\mathbf{x} \neq \tilde{\mathbf{x}}} \text{CEI}(\mathbf{x}) \leq \delta$, then go to Step 5. Otherwise, go to Step 3.

3. Simulate r replications at both $\tilde{\mathbf{x}}$ and the candidate feasible solution $\mathbf{x}_{\text{CEI}}^*$ that maximizes the CEI over the set of all candidate feasible solutions.
4. Update the simulation output at $\tilde{\mathbf{x}}$ with the new replications. If $\mathbf{x}_{\text{CEI}}^*$ is a design point then update the simulation output at $\mathbf{x}_{\text{CEI}}^*$ and go to Step 1. If $\mathbf{x}_{\text{CEI}}^*$ is not a design point, then add $\mathbf{x}_{\text{CEI}}^*$ to the set of design points, add the simulation output obtained at $\mathbf{x}_{\text{CEI}}^*$ to the collection of simulation output, and go to Step 1.
5. Return $\tilde{\mathbf{x}}$ as the estimated optimal solution.

The inputs to the algorithm are the initial number of design points k_s , the number of replications to be allocated each time a solution is simulated r (which can be adaptive provided $r > 0$), and the smallest practically important optimality gap δ .

3.2. Calculating the Conditional Distribution and CEI

Substituting the plug-in estimate $\hat{\mathbf{Q}}_\epsilon$ for the intrinsic precision matrix, and the MLEs $\hat{\boldsymbol{\theta}}$ and $\hat{\beta}_0$ for the parameters of the GMRF (see EC.2), into the conditional distribution (4), we obtain the estimated conditional distribution

$$\mathcal{N}\left(\hat{\beta}_0 \mathbf{1}_{n \times 1} + \bar{\mathbf{Q}}(\hat{\boldsymbol{\theta}})^{-1} \begin{pmatrix} \vec{\mathbf{0}}_{n_1 \times 1} \\ \hat{\mathbf{Q}}_\epsilon(\bar{\mathbf{y}}_2 - \hat{\beta}_0 \mathbf{1}_{n_2 \times 1}) \end{pmatrix}, \bar{\mathbf{Q}}(\hat{\boldsymbol{\theta}})^{-1}\right) \quad (9)$$

where

$$\bar{\mathbf{Q}}(\hat{\boldsymbol{\theta}}) \triangleq \begin{pmatrix} \mathbf{Q}_{11}(\hat{\boldsymbol{\theta}}) & \mathbf{Q}_{12}(\hat{\boldsymbol{\theta}}) \\ \mathbf{Q}_{12}(\hat{\boldsymbol{\theta}})^\top & \mathbf{Q}_{22}(\hat{\boldsymbol{\theta}}) \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{n_1 \times n_1} & \mathbf{0}_{n_1 \times n_2} \\ \mathbf{0}_{n_1 \times n_2}^\top & \hat{\mathbf{Q}}_\epsilon \end{pmatrix}.$$

In this conditional distribution, we ignore the variability that is added from estimating the mean β_0 , so the conditional covariance matrix is still $\bar{\mathbf{Q}}(\hat{\boldsymbol{\theta}})^{-1}$.

The computationally expensive calculation required to compute the conditional distribution is inversion of the matrix $\bar{\mathbf{Q}}(\hat{\boldsymbol{\theta}})$. Inverting a dense $n \times n$ matrix is, in general, $O(n^3)$. However, our matrix is sparse, and with our proposed neighborhood structure the fraction of non-zero elements is bounded above by $(2d+1)/n$. To compute all $n-1$ CEI's we only need the n diagonal elements of $\bar{\mathbf{Q}}^{-1}(\hat{\boldsymbol{\theta}})$, and the $n-1$ off-diagonal elements of one column. Methods for computing selected elements of diagonal-dominant sparse matrices can be used to get the diagonal elements and very few others. Most such methods are based on the pioneering paper by Takahashi et al. (1973), with more recent variations and applications found in Erisman and Tinney (1975), Niessner and Reichert (1983) and Vanhatalo and Vehtari (2012), and implemented in packages such as PARDISO (<http://www.pardiso-project.org/>). The one column we need can be obtained by a direct back-solve. Further, using sparse-matrix format the storage required to retain $\bar{\mathbf{Q}}$ is linear in the (small) number of non-zero elements. In EC.7 we provide some additional insight into how our specific neighborhood structure can be exploited. For our proof-of-concept GMIA and MR-GMIA (below) we used the general-purpose sparse matrix methods in Matlab, but custom code can fully exploit the techniques described above.

3.3. Asymptotic Convergence of GMIA

GMIA exploits the GMRF framework to terminate when the expected improvement from additional searching is small. Under very mild conditions GMIA, without any stopping condition, converges with probability 1 to a globally optimal solution. *This result does not depend in any way on the validity of the GMRF framework, but only on the following assumptions:* (A) $y(\mathbf{x}) > -\infty$ for all $\mathbf{x} \in \mathcal{X}$; (B) $0 < \text{Var}[Y(\mathbf{x})] < +\infty$ for all $\mathbf{x} \in \mathcal{X}$; and (C) the initial estimated precision matrix $\mathbf{Q}(\hat{\boldsymbol{\theta}})$ is positive definite and is not updated thereafter. In EC.3 we prove the following theorem:

THEOREM 2. The GMIA algorithm without a stopping condition simulates each solution $\mathbf{x} \in \mathcal{X}$ infinitely often with probability 1 as the number of iterations goes to infinity.

As a consequence of Theorem 2, the fact that the number of feasible solutions n is finite, and that the estimated optimal solution is the simulated solution with the smallest sample mean, the asymptotic convergence of GMIA follows by the strong law of large numbers.

Theorem 2 does not depend on our chosen neighborhood structure being “correct,” nor does it depend on having “good” parameter estimates for the GMRF. In fact, the result will hold with any neighborhood and GMRF parameterization that satisfies Assumption C. The efficiency of the CEI computations, and the validity of the CEI inference, will be improved by good choices and estimates, however. Further, we can relax Assumption C and allow $\hat{\boldsymbol{\mu}}$ and $\hat{\boldsymbol{\theta}}$, and therefore $\mathbf{Q}(\hat{\boldsymbol{\theta}})$, to be updated a finite number of times.

4. The Multiresolution Framework and Algorithm

The core problem in DOvS is balancing exploration and exploitation. Gaussian-process-based optimization obtains this balance by incorporating measures like EI to guide the search. Although a local search can be used to find a point that maximizes EI or KG locally (Xie et al. 2016), to fully benefit from this approach we need to compute an EI or KG measure for every feasible solution on every iteration. There will always be feasible regions \mathcal{X} that are too large for this to be possible. What constitutes “too large” depends on a number of implementation-specific and problem-specific factors.

In this section we show that our GMRF framework lends itself to a powerful multiresolution extension: We first partition \mathcal{X} into m disjoint regions $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_m$, where each region is a subset of \mathcal{X} that is connected in the graph \mathcal{G} . Then we let the nodes of a region-level GMRF represent a measure of the overall solution quality within each region, with a neighborhood structure again defined by proximity (i.e., adjacent regions). The region-level GMRF provides global guidance by facilitating a CEI comparison among regions. As in the previous section, the quality of individual solutions within a region \mathcal{P}_j is represented by a solution-level GMRF, except that the GMRFs for \mathcal{P}_j and \mathcal{P}_ℓ for $j \neq \ell$ are assumed to be independent.

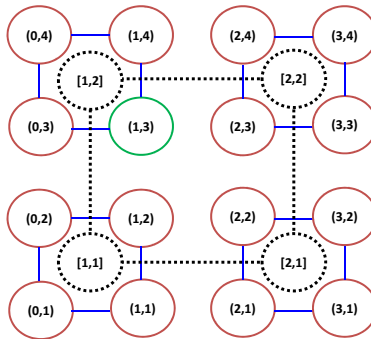


Figure 2 Illustration of a region-level GMRF and four solution-level GMRFs in two dimensions.

In this framework the solutions across regions are connected by the region-level GMRF, rather than by their neighboring individual solutions, and the solutions within a region are connected by their solution-level GMRF. We want the regions to have a large number of feasible solutions, to make the independence approximation reasonable, but not so many that we cannot efficiently calculate the CEI for each individual solution within a region. Further, it is beneficial (but not required) that the regions be of similar shape and contain a similar number of feasible solutions. Figure 2 shows a (very small) region-level and four solution-level GMRFs of a rectangular feasible region for (x_1, x_2) . Conceptually, we are not limited to just two resolutions: a GMRF for *regions of regions* could also be defined. Fortunately, region-level and solution-level GMRFs will be sufficient for many problems of practical interest.

In the multiresolution framework the region-level GMRF describing the quality and spatial structure of regions has its own mean and precision matrix, as described below. The solution-level and the region-level GMRFs work together to balance exploration and exploitation, and terminate when the joint CEIs fall below a threshold. The specific multiresolution algorithm described in this section is a first step toward realizing this general framework, and it is particularly well-suited to low-dimension problems with vast number of solutions. In Section 5.3, we demonstrate a variation that facilitates solving higher dimensional problems.

The response associated with the region \mathcal{P}_ℓ is

$$z(\mathcal{P}_\ell) = |\mathcal{P}_\ell|^{-1} \sum_{\mathbf{x} \in \mathcal{P}_\ell} y(\mathbf{x}).$$

That is, the measure of region-level quality $z(\mathcal{P}_\ell)$ is the *average* of the objective-function values of all solutions in \mathcal{P}_ℓ . Similar to the solution-level GMRF, we model $\mathbf{z} = (z(\mathcal{P}_1), z(\mathcal{P}_2), \dots, z(\mathcal{P}_m))^\top$ as a realization of a region-level GMRF $\mathbb{Z} \triangleq (\mathbb{Z}(\mathcal{P}_1), \mathbb{Z}(\mathcal{P}_2), \dots, \mathbb{Z}(\mathcal{P}_m))^\top \sim \mathcal{N}(\boldsymbol{\eta}, \mathbf{T}(\boldsymbol{\tau})^{-1})$ with mean vector $\boldsymbol{\eta}$ and precision matrix $\mathbf{T}(\boldsymbol{\tau})$ as a function of parameters $\boldsymbol{\tau}$, which are analogous to $\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})$, and $\boldsymbol{\theta}$ at the solution level. Similar to the solution-level GMRF, our main interest is to find the conditional distribution of \mathbb{Z} given observations from design regions.

Let $\mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_m\}$ denote the set of region-level “feasible solutions.” A “design region” \mathcal{P}_ℓ is any region for which at least one of its feasible solutions \mathbf{x} has been simulated; that is, there exists $\mathbf{x} \in \mathcal{P}_\ell$ such that $\mathbf{x} \in \Xi_2$. The set of design regions is denoted by Π_2 , and $\Pi_1 = \mathcal{P} \setminus \Pi_2$. We can partition $\mathbf{z}, \mathbb{Z}, \boldsymbol{\eta}$, and $\mathbf{T}(\boldsymbol{\tau})$ using Π_1 and Π_2 as we did at the solution level. Within design region \mathcal{P}_ℓ , let $\Xi_2(\mathcal{P}_\ell) \subseteq \mathcal{P}_\ell$ be the set of feasible solutions in \mathcal{P}_ℓ that have been simulated, and let $\Xi_1(\mathcal{P}_\ell) = \mathcal{P}_\ell \setminus \Xi_2(\mathcal{P}_\ell)$. See EC.4 for a justification and complete specification of the region-level GMRF.

For the solution-level GMRFs, we may choose to fit a different set of parameters for each region— $\boldsymbol{\mu}_\ell$ and $\boldsymbol{\theta}_\ell$ for \mathcal{P}_ℓ —or assume all regions share the same solution-level parameters $\boldsymbol{\mu}$ and $\boldsymbol{\theta}$, which substantially reduces the computation for parameter estimation. We employ the latter approach in experiments.

In EC.4, we present a detailed version of MR-GMIA. On each iteration of the algorithm, we select three *regions* in which to run simulations: (i) the design region \mathcal{P}_{\min} that contains the solution with the smallest sample mean among all simulated solutions in \mathcal{X} , (ii) the design region $\tilde{\mathcal{P}}$ that has the smallest region-level estimator $\tilde{\mathcal{Z}}(\mathcal{P}_\ell)$ among $\mathcal{P}_\ell \in \Pi_2$ (see EC.4), and (iii) the region $\mathcal{P}_{\text{CEI}}^*$ that has the largest region-level CEI among all regions in \mathcal{P} . Since it is possible that \mathcal{P}_{\min} is the same as $\tilde{\mathcal{P}}$ or $\mathcal{P}_{\text{CEI}}^*$, we may only select two regions on some iterations. Notice that $\mathcal{P}_{\text{CEI}}^*$ may be a region in which no solution has yet been simulated, in which case we simulate k_s initial design points from $\mathcal{P}_{\text{CEI}}^*$. For \mathcal{P}_{\min} and $\tilde{\mathcal{P}}$, and for $\mathcal{P}_{\text{CEI}}^*$ if it already is a design region, we apply the same method as in the solution-level GMIA algorithm: at each iteration, we simulate two solutions, the solution with the smallest sample mean and the solution with the largest solution-level CEI in that region. *The algorithm stops when both largest region-level CEI and solution-level CEI fall below δ .*

In EC.4 we prove the following theorem:

THEOREM 3. The MR-GMIA algorithm in EC.4 without a stopping condition simulates each solution $\mathbf{x} \in \mathcal{X}$ infinitely often with probability 1 as the number of iterations goes to infinity.

Therefore, the MR-GMIA finds the global optimal solution with probability 1 as the simulation budget increases. More generally, such global convergence result can be obtained for any MR-GMIA that guarantees a “solution” in a GMRF at each resolution is simulated infinitely often. In Section 5.3, we define a three-resolution GMIA to solve a 15-dimensional problem whose global convergence can be shown by a simple extension of the proof of Theorem 3.

5. Numerical Experiments

In this section we present three examples to demonstrate the performance of GMIA and MR-GMIA. The main goals of this section are to show (i) GMIA combined with expected improvement

provides valid inference on the remaining optimality gap at termination while achieving good finite-budget performance for a DOvS problem, and (ii) MR-GMIA can solve a large-scale DOvS problem efficiently and effectively. We also provide a proof-of-concept demonstration of applying MR-GMIA to a fifteen-dimensional problem by using a three-resolution GMRF model.

The first example is an (s, S) inventory optimization problem that exhibits behavior like a practical DOvS problem. We use this example to demonstrate (i) above by comparing performance of GMIA and a continuous GRF-based procedure with Gaussian correlation function. We also compare three sampling criteria, EI, CEI, and KG, combined with GMIA and with the continuous GRF-based procedure. For EI and CEI, the stopping conditions $EI/CEI \leq \delta$ were tested. Because $KG \leq \delta$ is interpreted differently from $EI/CEI \leq \delta$ as discussed in Section 2.3, we did not test KG as a stopping criterion. Instead, we examined the finite-budget performance of GMIA and GRF-based procedures to compare EI, CEI, and KG for search guidance. We also apply the MR-GMIA presented in EC.4 to the same (s, S) inventory problem to show its improvement in efficiency compared to GMIA while still achieving the target optimality gap.

To achieve goal (ii) we tested the MR-GMIA on the second test problem with two different numbers of feasible solutions, 10,000 and 1,000,000, to examine the impact of the problem size on the performance of the MR-GMIA. This problem is created by adding stochastic noise to the two-dimensional Griewank function, a well-known test function for a global optimization problem with many local minima, but evaluated only on a lattice to test the performance of MR-GMIA on a difficult DOvS problem.

Finally, we devise a three-resolution GMIA to demonstrate that we can add a higher resolution to the two-resolution GMRF structure presented in EC.4 to solve a high-dimensional problem with a large number of solutions. The response function for this test case is a fifteen-dimensional inverted Gaussian density function, which we converted to a DOvS problem in the same way as the Griewank function, on an integer lattice with $4^{15} = 1,073,741,824$ solutions.

In all examples in this section, MLEs are computed only once at the beginning of each algorithm.

5.1. (s, S) Inventory Problem

The objective function for the (s, S) inventory problem suggested in Koenig and Law (1985) is the expected average cost per period of the inventory system over 30 periods. We assume the demand for inventory in each period is a sequence of i.i.d. Poisson random variables with a common mean of 25. To obtain a simple rectangular feasible region as in Figure 2, we chose our decision variables to be s and $S - s$, where the feasible region is defined by constraints, $1 \leq s \leq 100$ and $1 \leq S - s \leq 100$, which leads to 10,000 feasible solutions. The optimal solution to this DOvS problem is $s = 17$ and $S - s = 36$ with an expected average cost per period of \$106.12 based on 1 million replications

of Monte Carlo simulation at each feasible solution. We assume the minimum difference in the objective function that is worth detecting is $\delta = \$1$, which is used for the stopping criterion in the algorithms.

If one simulation run of the inventory problem takes a long time, 10,000 feasible solutions can be considered large. On the other hand, if each run takes a short time, then we may apply an IZ R&S procedure that simulates all solutions in the feasible region. A main takeaway from this example is that GMIA provides similar inference on the optimality gap as the IZ R&S procedure by simulating many fewer feasible solutions when combined with EI or CEI.

5.1.1. Comparison of GMIA and Continuous GRF-based Algorithm To compare with GMIA, we constructed a continuous GRF-based procedure by changing the metamodel of the response surface in GMIA from GMRF to continuous GRF with a Gaussian correlation function. We used the Matlab `fitrgp` library to fit the GRF regression. For both algorithms, we tested EI, CEI, and KG for search guidance, which we computed for all 10,000 feasible solutions at each iteration. To compute KG for the feasible solutions, we used the `MatlabKG` library provided by Frazier (2009–2010), where the exact algorithm is presented in Frazier et al. (2009). Note that we assume the cost of sampling a solution is constant for all solutions. For the stopping criterion, we tested $EI/CEI \leq \delta$ for both the GMIA and GRF-based procedures. Because $KG \leq \delta$ has a different interpretation than $EI/CEI \leq \delta$, performance of KG as a stopping criterion was not tested. Instead, we compared the finite-budget performances of GMIA + EI/CEI/KG and GRF + EI/CEI/KG. Although we claim the strength of GMIA is its inference about the optimality gap at stopping, good fixed-budget performance of GMIA will reassure users.

For all six algorithms, $k_s = 20$ initial design points in the feasible region were selected by Latin hypercube sampling and $r = 10$ replications were simulated for each sampled design point. For GMIA, the solution with the smallest sample mean among the simulated solutions was chosen as the current best solution at each iteration when EI or CEI is used as a criterion. On the other hand, the solution with the best conditional mean was selected as the current best when KG is used to be consistent with the definition of KG. For the GRF-based procedure, a solution with the best conditional mean was selected as the current best regardless of the sampling criterion.

We first present the finite-budget performance comparisons. Figure 3 shows boxplots of optimality gaps from 400 runs of all six algorithms stopped after 100 iterations, where for each run the same random number seed was used for all six algorithms. Below the boxplots, the sample means and their standard errors of the optimality gap are presented. Notice that all algorithms based on GMRF have lower medians and means than those based on the continuous GRF. When combined with GMIA, EI, CEI, and KG show very similar performance. On the other hand, GMIA + CEI

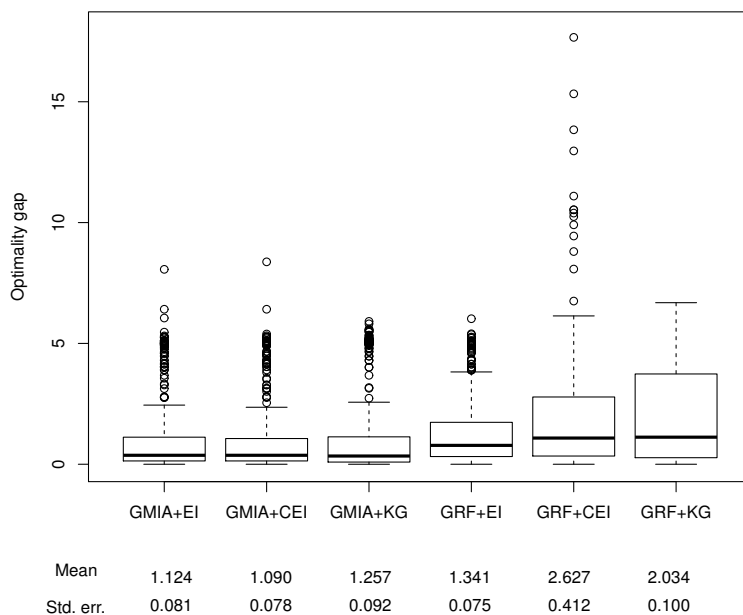


Figure 3 Boxplots and the sample averages and their standard errors of the optimality gaps of six algorithms applied to the (s, S) inventory problem stopped after 100 iterations. For each algorithm, 400 runs were made. The whiskers of boxplots are extended to the observations within 1.5 times the inter-quartile range. We omitted three observations (two with optimality gap 112.26 and one with 24.51) from the boxplot of GRF + CEI for better scaling of other boxplots.

has statistically significantly smaller average optimality gap than GRF + KG, which is a popular current method to solve a DOvS problem. The performance of GRF + EI is comparable to GMIA’s, whereas, GRF + CEI has the largest average optimality gap among all six algorithms and several of its runs have high optimality gaps.

We also present the progress each algorithm made during the first 100 iterations averaged over 400 runs in Figure 4. The trajectory of GMIA + EI is very similar to that of GMIA + CEI, and therefore was omitted from Figure 4 for better comparisons to the other the algorithms. GRF-based procedures exhibit faster convergence at the beginning but slow down substantially as the algorithms progresses, especially for GRF + CEI and GRF + KG, whereas GMIA + CEI/KG show more steady progress.

We observed a few runs of GMIA and the GRF-based procedure to see which solutions are selected to sample at each iteration and discovered that GMIA tends to sample solutions closer to the current optimal solution, whereas the GRF-based procedure tends to sample solutions farther away from the current optimal. This difference in behaviors makes the GRF-based algorithm converge faster initially as it explores the feasible solution space more broadly, but slow down as

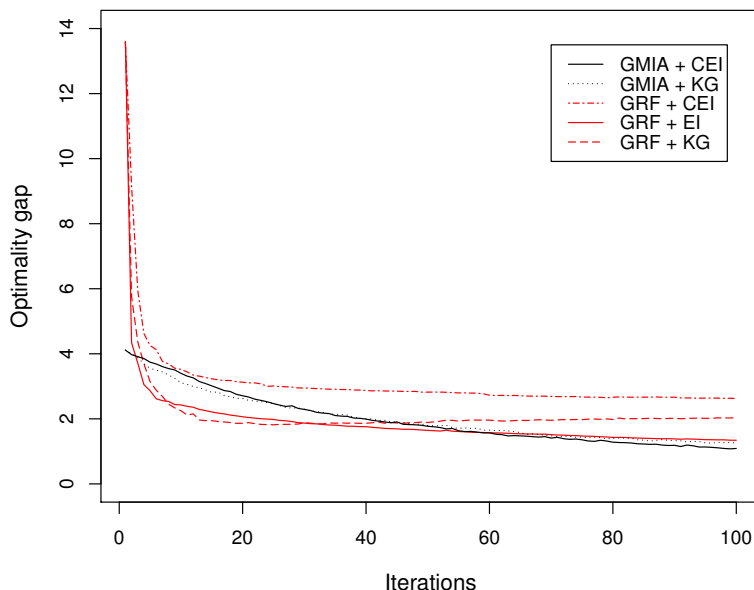


Figure 4 The trajectory of optimality gap at each iteration of each algorithm included in Figure 3 (except for GMIA + EI) averaged over 400 runs. GMIA + EI is omitted since it has a very similar trajectory as GMIA + CEI.

the algorithm proceeds since it does not exploit the neighborhood containing the current optimal solution as much. GMIA, on the other hand, makes slower progress at the beginning as it explores less globally compared to the GRF-based procedure, however, it improves steadily by making local improvements from the current optimal solution.

Now we move to examine whether GMIA and the GRF-base procedures make valid inference on the remaining optimality gap at termination when EI and CEI are used as stopping criteria. From the definition of EI (CEI), $EI (CEI) \leq \delta$ indicates that the expected optimality gap of the current best solution is less than δ . For the benchmark, we applied the KN R&S procedure (Kim and Nelson 2001) to the same (s, S) inventory problem; it simulates *all* feasible solutions at the beginning and sequentially eliminates solutions from contention. We adopted CRN for the KN procedure, set the number of first-stage replications of each feasible solution to $n_0 = 10$, $1 - \alpha = 0.95$ for the PCS and $\delta = \$1$. Table 1 shows the average and maximum optimality gaps at termination from 50 runs of all five procedures including KN. The average number of solutions that each algorithm visited as well as the average number of replications spent until termination are also presented. All standard errors are in the parentheses.

Notice that both average and maximum optimality gaps of GMIA + EI and GMIA + CEI are less than the desired optimality gap $\delta = \$1$. Moreover, their average optimality gaps are statistically indistinguishable from the optimality gap of KN, while spending about a half of the simulation

Table 1 Average and maximum of optimality gaps at termination of 50 runs of GMIA + EI/CEI, GRF + EI/CEI, and KN given $\delta = \$1$. The average number of sampled solutions and the average number of replications spent by each algorithm until termination are also presented. The standard errors are presented in parentheses.

	GMIA + EI	GMIA + CEI	GRF + EI	GRF + CEI	KN
Average optimality gap	0.089 (0.009)	0.096 (0.012)	3.283 (0.346)	3.401 (0.343)	0.097 (0.009)
Max optimality gap	0.271	0.348	8.545	8.545	0.271
Average number of sampled solutions	2,775 (82)	2,750 (76)	20.9 (0.2)	20.3 (0.1)	10,000 -
Average number of replications	55,314 (1,633)	54,854 (1,526)	218 (4)	207 (2)	108,111 (185)

budget KN spent and without simulating all feasible solutions. On the other hand, GRF + EI and GRF + CEI stopped prematurely when the optimality gap is still much larger than δ after sampling a few solutions. These results indicate that the GMRF metamodel lets GMIA correctly assess the remaining optimality gap via EI/CEI, whereas the continuous GRF underestimates it. This shows that a GMRF is a better model to represent the responses of the discrete feasible solutions of this DOvS problem.

In Figure EC.1 of EC.5, we present heatmaps of the variance-covariance matrices of the underlying GMRF and the continuous GRF for this problem, which provides an intuitive explanation for such a performance difference. Although the initial variance-covariance matrices of the two Gaussian processes are based on the MLEs computed from the same set of initial observations, the continuous GRF has a much thicker diagonal band with high variances and covariances relative to GMRF. However, the continuous GRF’s variances and covariances decrease significantly after updating the distribution of the GRF conditional on the initial observations because the high correlations among nearby solutions makes the variances and covariances of the solutions near the initial design points greatly reduced. This causes the GRF to quickly decide that there is not much variability left in the random field resulting in small EI or CEI of solutions. On the other hand, the Markovian assumption of GMRF combined with our choice of neighborhood structure makes the precision matrix $\mathbf{Q}(\boldsymbol{\theta})$ very sparse and as a result, the off-diagonal elements of the variance-covariance matrix of the GMRF (inverse of $\mathbf{Q}(\boldsymbol{\theta})$) decay faster than the continuous GRF’s. Hence, the variance-covariance matrix of GMRF updated conditional on the initial observations still has higher variances and covariances compared to the continuous GRF, which helps us make better inference on the remaining optimality gap.

Note that `fitrgp` fits the parameters of GRF’s prior distribution using maximum likelihood estimation. One may consider taking a fully Bayesian approach using hyperparameters, which will increase uncertainty about the response surface and as a result, may delay the stopping decision for

the continuous GRF. Also recall that we used a Gaussian correlation function for the continuous GRF, which models an infinitely differentiable response surface. Thus, the modeled response surface may be too smooth to represent a discrete surface. A Matérn correlation function can be an alternative to a Gaussian correlation function as it lets one control the differentiability of the surface. Nevertheless, it still models a continuous surface.

The results presented in this section show that GMIA combined with EI or CEI provides good finite-budget performance and valid inference on the remaining optimality gap at termination when EI or CEI is used as a stopping criterion. As the *no free lunch* theorem in Wolpert and Macready (1997) proves, we may find an instance of a DOvS problem on which the continuous GRF-based procedure has better finite-budget performance than GMIA. However, the valid stopping decision of GMIA is a valuable feature of GMIA independently from its finite-budget performance. Although CEI is the expected improvement that correctly averages over all remaining sources of uncertainty, the empirical performance of deterministic EI and CEI in this example were statistically indistinguishable.

5.1.2. Comparison of GMIA and MR-GMIA We also ran the version of MR-GMIA in EC.4 combined with CEI and EI on the same (s, S) -inventory problem with region-level neighborhood structure as depicted in Figure 5b. We divided the same feasible region with 10,000 solutions into 25 square regions including 400 feasible solutions numbered bottom to top starting from the lower-left corner. The true optimal solution $(17, 36)$ is contained in region \mathcal{P}_2 , while \mathcal{P}_3 has the smallest region-level response.

We have two precision matrices: region-level precision matrix $\mathbf{T}(\boldsymbol{\tau})$ and solution-level precision matrix $\mathbf{Q}(\boldsymbol{\theta})$. The former, in this example, is a 25 by 25 matrix and the latter is 400×400 , both much smaller than the $10,000 \times 10,000$ precision matrix used in the GMIA run. As Cholesky decomposition of a sparse $n \times n$ matrix is $O(n^2)$, the smaller dimensions of the precision matrices significantly reduces the computation time involved in both MLE and CEI calculations.

We chose $k_r = 5$ initial design regions by Latin hypercube sampling. To select the design points within each region, we used the same Latin hypercube sample of size $k_s = 10$ for all design regions; for this reason some of the regions in Figure 5b show the same light grey pattern. From each selected design point we simulated 10 replications and used the result to compute the MLEs. At each iteration, region-level CEIs are calculated for all regions. The solution-level CEIs are calculated for all solutions within the currently sampled region at the iteration if the region was simulated previously. The algorithm stops if the largest region-level CEI and the largest solution-level CEIs of the regions that are sampled at the current iteration drop below $\delta = \$1$.

Figure 5 compares the simulation replications allocated across the feasible solutions by GMIA and MR-GMIA from one sample path of both algorithms when CEI is used as the sampling and

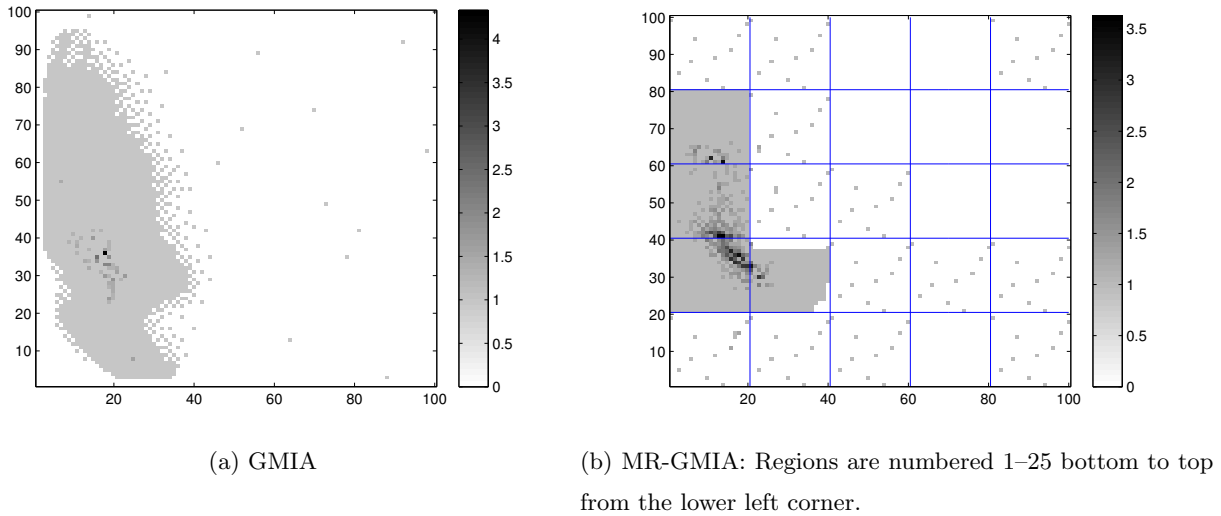


Figure 5 Allocations of simulation replications at feasible solutions for the (s, S) inventory problem in base-10 log scale.

stopping criterion ($CEI \leq \$1$). The large light grey area in Figure 5a indicates the solutions that were sampled only once, i.e., one time allocation of 10 replications. Points with darker shades were selected as the current sample best for one or more iterations and simulated repeatedly. The darkest point $(18, 36)$ is the solution that GMIA chose as optimal on this run. Notice that the design points are clustered with each other except for a few initial design points obtained from Latin hypercube sampling. The algorithm did not simulate any solutions near the upper right and the lower left corners of the feasible region where the objective function increases steeply. This indicates that the estimated GMRF structure guides the algorithm to rule out inferior solutions. In EC.6, we present the plots of the conditional means of the GMRF at different iterations to show how the GMRF updates its belief about the response surface as the algorithm proceeds.

Figure 5b shows that MR-GMIA allocated most of the simulation replications within $\mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4$, and \mathcal{P}_7 . The surrounding regions of these four regions were selected once or a few more times before the algorithm stopped. After the initial sampling, MR-GMIA quickly finds the region that is likely to contain the global optimal solution using the region-level CEI. Notice that \mathcal{P}_{25} at the upper right corner whose response $z(\mathcal{P}_{25})$ is the largest among all regions was simulated only once and never selected again. Also, the neighboring regions of \mathcal{P}_{25} were not simulated. This shows that the region-level GMRF informed the algorithm correctly that the regions close to \mathcal{P}_{25} are not likely to contain the optimal solution. Compared to the simulation replication allocation of GMIA in Figure 5a, the biggest difference is that MR-GMIA first chooses regions to simulate then selects the solutions within those regions; therefore, sampling solutions is restricted within the boundaries

Table 2 Average and maximum of optimality gaps at termination of 50 runs of MR-GMIA + EI/CEI. The average number of sampled solutions and the average number of replications spent by each algorithm until termination are also presented. The standard errors are presented in parentheses.

	MR-GMIA + EI	MR-GMIA + CEI
Average optimality gap	0.505 (0.065)	0.321 (0.054)
Max optimality gap	1.886	2.218
Average number of sampled solutions	303 (36)	525 (73)
Average number of replications	6,071 (901)	13,214 (2,188)

of selected regions. This, in fact, helps MR-GMIA concentrate on the regions that are likely to contain the global optimal solution.

Table 2 shows the average and maximum optimality gaps of 50 runs of MR-GMIA when EI and CEI are used as stopping criteria. Compared to the performance of GMIA + EI/CEI in Table 1, MR-GMIA + EI/CEI have larger average and maximum optimality gaps at termination, however, their average optimality gaps are less than δ , which means that MR-GMIA still achieves the target optimality gap on average. Moreover, the average numbers of replications MR-GMIA + EI/CEI spent are much lower than those of GMIA + EI/CEI (2,775/2,750) and KN (108,111) in Table 1. Also, the average numbers of solutions that MR-GMIA + EI/CEI visited are also much lower than that of GMIA + EI/CEI (55,314/54,854), which confirms the behavior observed in Figure 5b; MR-GMIA concentrates its effort on the regions likely to contain the optimal. MR-GMIA sampled a larger number of solutions and spent more replications when CEI is used as a stopping criterion than EI. As a result, we have smaller average optimality gap for CEI.

Another advantage of MR-GMIA over GMIA is the matrix computation time. For the GMIA runs of this problem, the Cholesky decomposition of the $10,000 \times 10,000$ precision matrix, the additional multiplications to invert it, and computing the conditional mean of all feasible solutions, took 13.61 (standard error = 0.23), 78.00 (0.92) and 6.9×10^{-5} (2×10^{-6}) seconds, respectively. For the MR-GMIA runs, the same calculation times for the region-level GMRF took 2.6×10^{-5} (2×10^{-6}), 5.3×10^{-5} (3×10^{-6}), and 1.2×10^{-4} (8×10^{-5}) seconds, respectively, and those for the solution-level GMRF took 2.4×10^{-3} (1×10^{-4}), 9.0×10^{-3} (2×10^{-4}), and 6.3×10^{-4} (3×10^{-5}) seconds, respectively. Even after considering that MR-GMIA performs the solution-level matrix calculations for up to three regions at each iteration, the total time MR-GMIA spends on matrix computation is far less than that of GMIA.

5.2. Griewank Function

The Griewank function (<http://www.sfu.ca/~ssurjano/griewank.html>) has many local minima, which makes it a challenging test function for optimization algorithms. Figure 6 shows the two-dimensional Griewank function on the domain $[-5, 5] \times [-5, 5]$. Within this domain the range

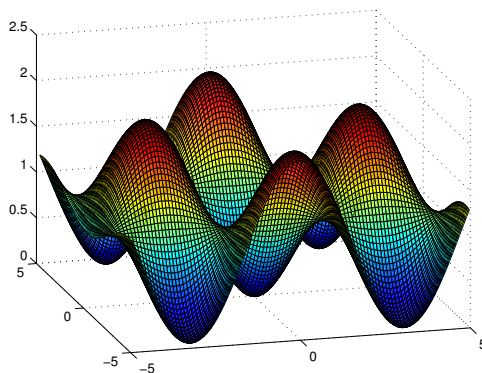


Figure 6 Two-dimensional Griewank function on $[-5, 5] \times [-5, 5]$.

of the function is $[0, 2.0044]$, and the global minimum is at $(0, 0)$. Notice that there are four local minima near the four corners of the domain and their response values are 0.0086.

To create a DOvS problem, we take as feasible solutions the 100×100 lattice on $[-5, 5] \times [-5, 5]$ where the response at each solution is given by the Griewank function in Figure 6. Normally distributed simulation noise with mean 0 and variance σ^2 is added to the response function to make it stochastic. As depicted in Figure 7a, we divided the feasible solutions into 25 square regions, each of which contains 400 solutions. To analyze the impact of δ on the performance of the algorithm, we tested three values of δ : 0.005, 0.01, and 0.02. We chose these values relative to 0.0086, the difference between the responses at the local minima and the global minimum, so that we can test the effect of δ on the solution quality of the algorithm. We adjusted the standard error of the sample mean at each design point when it is first visited to be relative to δ . Given $r = 16$, the number of replications at each solution, and δ , we tested three values of σ/\sqrt{r} : $\delta/2$, δ , and 2δ . For each run, we used Latin hypercube samples of $k_r = 5$ and $k_s = 20$ to select the initial design regions and design points, respectively.

Figure 7a shows how simulation replications were allocated across the feasible region in one sample path of MR-GMIA. Notice that all regions were visited at least once, unlike the (s, S) inventory problem in Figure 5b. Compared to the (s, S) inventory problem, the response surface of the Griewank function is harder to predict as the region-level response $z(\mathcal{P}_\ell)$ for all 25 regions do not vary much. Therefore, the region-level CEI did not rule out any region before it was visited at least once. However, once the regions were visited, the algorithm correctly found the regions that either contain the global and local minima or are close to them and concentrated the simulation effort there. Notice that the algorithm simulated solutions in the ellipsoidal area that contains the global optimal solution most frequently.

Table 3 presents the results of 50 runs of MR-GMIA with the chosen δ and σ values. For each run of all nine cases of (δ, σ) in Table 3, common random numbers were adopted. Notice that our

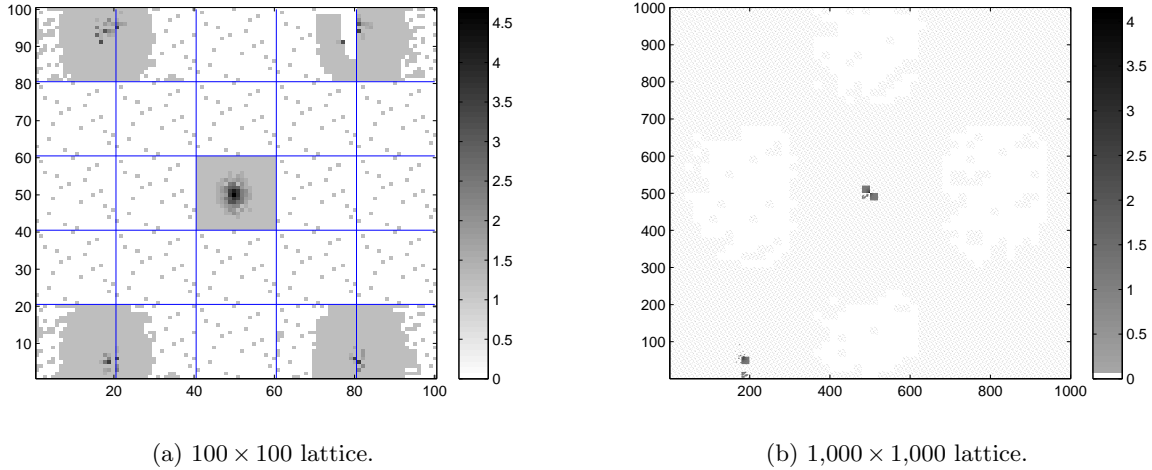


Figure 7 Allocations of simulation replications at feasible solutions of the Griewank function in Figure 6 in base-10 log scale.

algorithm returned the global optimal solution in all 50 runs of all settings except for $(\delta, \sigma) = (0.02, 0.08)$, and $(0.02, 0.16)$. Even in these two cases, the average response at the solution is within δ from 0. As expected, both the number of simulated solutions and the number of replications spent tend to increase as δ decreases and as σ increases. However, MR-GMIA is not very sensitive to the choice of δ for smaller σ .

To test the impact of the number of feasible solutions on the performance of the algorithm, we ran MR-GMIA with $(\delta, \sigma) = (0.005, 0.01)$ but using a finer lattice, as depicted in Figure 7b: a $1,000 \times 1,000$ grid instead of 100×100 , which results in 1,000,000 feasible solutions. These solutions are divided into 2,500 regions, each of which contains 400 feasible solutions. Figure 7b is a map of replications spent in the feasible region in one run of MR-GMIA. Notice that the areas in white, which indicate that they were never simulated, correspond to the peaks of the Griewank function in Figure 6. Unlike in the 100×100 case, the region-level responses actually differ in this case as we have a finer grid. The region-level CEI effectively rules out the regions that are not likely to contain the optimal solution using the region-level GMRF structure. Most of the simulation effort is concentrated in the regions near the global minimum and a local minimum in the lower-left corner.

The average optimality gap of 50 runs of MR-GMIA applied to the $1,000 \times 1,000$ Griewank problem is 0.0002 (standard error = 0.00004) and the maximum response was 0.0016, which is smaller than $\delta = 0.005$. The average number of simulated solutions is 24,696 (1307), which is only 2.5% of the 1 million feasible solutions. Compared to the 100×100 case, which simulated 19% of the feasible solutions on average, we can confirm that the algorithm works efficiently even if we have a large number of feasible solutions. The average number of replications spent is 89,079 (2775),

Table 3 Results from 50 runs of MR-GMIA applied to the Griewank functions with different δ and σ . Standard errors are provided in parentheses.

δ	0.005			0.01			0.02		
σ	0.01	0.02	0.04	0.02	0.04	0.08	0.04	0.08	0.16
Average response at optimal solution	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002 (0.0001)	0.0015 (0.0006)
Max response at optimal solution	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0025	0.0219
Average number of simulated solutions	1,927 (26)	1,945 (25)	2065 (27)	1,945 (25)	2065 (27)	2444 (29)	2065 (27)	2509 (28)	2837 (81)
Average number of replications	108,955 (2417)	111,229 (2377)	122,036 (2877)	111,229 (2377)	121,974 (2872)	155,254 (2742)	121,974 (2872)	157,207 (3007)	184,147 (6,634)

which is less than 8 times larger than that of the 100×100 case in Table 3. These results show the multiresolution algorithm scales well as the number of feasible solutions increases.

5.3. Inverted Multivariate Normal Density Function

Lastly, we present a proof-of-concept study that applies MR-GMIA based on a three-resolution GMRF model to solve a high-dimensional DOvS problem. The response function is the inverted Gaussian density function suggested in Xu et al. (2010):

$$f(x_1, x_2, \dots, x_d) = -\beta \exp \left\{ -\gamma \sum_{j=1}^d j x_j^2 \right\}, \quad \mathbf{x}^\top = (x_1, x_2, \dots, x_d). \quad (10)$$

The function is minimized at $\mathbf{x}^\top = (0, 0, \dots, 0)$ and does not have any local minima. We chose $\beta = 1000$ and $\gamma = 0.001$, which makes the optimal objective function value -1000 . Normally distributed noise with mean 0 and variance 16 is added to the response function to make it stochastic. We chose $d = 15$ and considered solutions within the integer lattice in $[0, 3]^{15}$ as the feasible solution set containing $4^{15} = 1,073,741,824$ solutions within which the range of (10) is $[-1000, -861.29]$.

In Section 5.2, we designed the regions of the MR-GMIA to have the same dimension as the feasible solution space. However, in this example, the smallest hyper-rectangular region with the same dimension has $2^{15} = 32,768$ solutions and a $32,768 \times 32,768$ solution-level precision matrix. Even with the sparse-matrix method in Matlab, this is a large matrix. Another problem for such a region is that all solutions within the region are corner solutions, and hence, we cannot fully take advantage of the inference from their neighbors.

We overcome the first issue by adding an additional resolution; instead of a two-resolution GMRF we discussed in Section 4, we have a three-resolution GMRF consisting of *super-region-level*, *region-level*, and *solution-level* GMRFs. A super-region-level “solution” represents a region of regions just as a region-level “solution” represents a region of solutions. This extension reduces

the number of “solutions” within the GMRF at each level. Moreover, we reduced the dimension of higher-level GMRFs by projection. For instance, a “solution” of the super-region-level GMRF represents an average quality of regions of solutions whose first five coordinates in $[0, 3]^{15}$ are $[1, 0, 1, 3, 1]$ and among these regions, a region represents solutions whose first ten coordinates are $[1, 0, 1, 3, 1, 1, 0, 0, 0, 2]$. Therefore, we have 4^5 regions of regions in the super-region-level GMRF, 4^5 regions in each of 4^5 region-level GMRFs, and 4^5 solutions in each of 4^{10} solution-level GMRFs. There are two advantages of designing the GMRFs in this way. Firstly, it reduces the number of corner solutions, and hence lets us better utilize inference from the neighboring solutions. Secondly, we have more flexibility in choosing the number of layers of GMRFs versus how many solutions to include in each GMRF. For instance, we may decide to have five layers of GMRFs instead of three and it is easier to design which “solutions” to include in each layer of GMRF by projection rather than preserving all 15 dimensions in each layer.

We applied the MR-GMIA described above to this problem and observed 10,000 iterations of 200 sample paths. For the initial design, 5 solutions per region, 5 regions per super-region, and 5 super-regions were sampled. In total, 125 solutions are sampled initially, which is only $1.16 \times 10^{-6}\%$ of the feasible solutions. Each time a solution is selected we simulated $r = 4$ replications and CEI was used as the sampling guidance. Similar to the two-resolution GMIA in EC.4, we selected three super-regions at each iteration; the super-region with the smallest super-region-level mean, the super-region that contains the current optimal solution, and the super-region with the largest super-region-level CEI. In each selected super-region, two regions were sampled; the region with the largest region-level mean and the region with the largest region-level CEI. Similarly, two solutions were sampled in each selected region.

The average optimality gap of 200 runs after 10,000 iterations is 1.27 (standard error = 0.12), and the average number of solutions that the algorithm visited is 73,344 (1,046), which is only 0.0068% of the feasible solution space. We also tested the stopping condition of $\text{CEI} \leq \delta$ for $\delta = 1$ and let the algorithm stop either when all super-region-level, region-level, and solution-level CEIs fall below δ or after 10,000 iterations. Among the 200 runs, 58 runs stopped by the stopping criterion and their average optimality gap is 1.23 (0.24), which shows good performance.

This experiment explores a different way to construct multi-resolution GMRFs to solve a high-dimensional problems. There are various design questions including, but not limited to the number of resolutions to use and which projection function to define “solutions” in each level of GMRF. We defer the answers to these questions to future research, but the potential is clear.

Acknowledgments

This article is based upon work supported by the National Science Foundation under Grant No. CMMI-0900354. We would also like to thank Håvard Rue, the Area Editor, Associate Editor

and two conscientious referees for feedback that improved the paper. Portions of this work were previously published in Salemi et al. (2014).

Author Biographies

Peter Salemi is a Modeling and Simulation Engineer/Data Scientist in the Operations Research Department at The MITRE Corporation. His research interests are simulation optimization and simulation metamodeling. He received his Ph.D. in Industrial Engineering and Management Sciences from Northwestern University, his M.S. in Operations Research from the University of California, Berkeley, and his B.S. in Mathematics and Finance from the University of Massachusetts, Amherst.

Eunhye Song is the Harold and Inge Marcus Early Career Assistant Professor in the Department of Industrial and Manufacturing Engineering at Penn State University. Her research area is simulation analysis methodology with emphasis on large-scale optimization via simulation, simulation uncertainty quantification and optimization via simulation under input model risk. Her website is www.eunhyesong.info.

Barry L. Nelson is the Walter P. Murphy Professor of the Department of Industrial Engineering and Management Sciences at Northwestern University and a Distinguished Visiting Scholar at Lancaster University in England. His research focus is on the design and analysis of computer simulation experiments on models of discrete-event, stochastic systems, including methodology for simulation optimization, quantifying and reducing model risk, variance reduction, output analysis, metamodeling and multivariate input modeling. He has published numerous papers and three books, including *Foundations and Methods of Stochastic Simulation: A First Course* (Springer, 2013). Nelson is a Fellow of INFORMS and IIE. Further information can be found at www.iems.northwestern.edu/~nelsonb/.

Jeremy Staum is a former Associate Professor of Industrial Engineering and Management Sciences at Northwestern University. His research interests include simulation metamodeling, optimization via simulation, and risk management. His website is users.iems.northwestern.edu/staum.

References

- Amemiya T (1985) *Advanced Econometrics* (Cambridge, MA: Harvard University Press).
- Ankenman B, Nelson BL, Staum J (2010) Stochastic kriging for simulation metamodeling. *Operations Research* 58(2):371–382.
- Benassi R, Bect J, Vazquez E (2011) Robust gaussian process-based global optimization using a fully bayesian expected improvement criterion. *Learning and Intelligent Optimization*, 176–190 (Springer).

- Chen K (2005) *Matrix Preconditioning Techniques and Applications* (Cambridge, UK: Cambridge University Press).
- Chen Y, Davis TA, Hager WW (2008) Algorithm 887: CHOLMOD, supernodal sparse cholesky factorization and update/downdate. *ACM TOMS* 35(3):1–14.
- Cochran W (1977) *Sampling Techniques* (New York: Wiley).
- Diggle PJ, Menezes R, Su T (2010) Geostatistical inference under preferential sampling. *Journal of the Royal Statistical Society: Series C* 59(2):191–232.
- Erismann A, Tinney W (1975) On computing certain elements of the inverse of a sparse matrix. *Communications of the ACM* 18(3):177–179.
- Frazier P (2012) Tutorial: Optimization via simulation with Bayesian statistics and dynamic programming. Laroque C, Himmelspach J, Pasupathy R, Rose O, Uhrmacher AM, eds., *Proceedings of the 2012 Winter Simulation Conference*, 1–16 (Piscataway, NJ: IEEE).
- Frazier P, Powell W, Dayanik S (2009) The knowledge-gradient policy for correlated normal beliefs. *INFORMS Journal on Computing* 21(4):599–613.
- Frazier PI (2009–2010) URL <https://people.orie.cornell.edu/pfrazier/src.html>.
- Huang D, Allen TT, Notz WI, Zeng N (2006) Global optimization of stochastic black-box systems via sequential kriging metamodels. *Journal of Global Optimization* 34(3):441–466.
- Jalali H, Van Nieuwenhuysse I, Picheny V (2015) Comparison of Kriging-based methods for simulation optimization with heterogeneous noise. Technical report, KU Leuven.
- Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* 13(4):455–492.
- Kim S (2013) Statistical ranking and selection. *Encyclopedia of Operations Research and Management Science*, 1459–1469 (New York: Springer).
- Kim S, Nelson BL (2001) A fully sequential procedure for indifference-zone selection in simulation. *ACM TOMACS* 11(3):251–273.
- Koenig LW, Law AM (1985) A procedure for selecting a subset of size m containing the l best of k independent normal populations, with applications to simulation. *Communications in Statistics* B14(3):719–734.
- Miller KS (1981) On the inverse of the sum of matrices. *Mathematics Magazine* 54(2):67–72.
- Nelson BL (2010) Optimization via simulation over discrete decision variables. *TutORials in Operations Research* 7:193–207.
- Niessner H, Reichert K (1983) On computing the inverse of a sparse matrix. *International journal for numerical methods in engineering* 19(10):1513–1526.
- Quan N, Yin J, Ng SH, Lee LH (2013) Simulation optimization via kriging: A sequential search using expected improvement with computing budget constraints. *IIE Transactions* 45(7):763–780.

- Rue H, Held L (2005) *Gaussian Markov Random Fields: Theory and Applications* (New York: Chapman and Hall/CRC).
- Saad Y (2003) *Iterative Methods for Sparse Linear Systems* (Philadelphia: Society for Industrial and Applied Mathematics (SIAM)), second edition.
- Salemi P, Nelson BL, Staum J (2014) Discrete optimization via simulation using Gaussian Markov random fields. Tolk A, Diallo SY, I O Ryzhov LY, Buckley S, Miller JA, eds., *Proceedings of the 2014 Winter Simulation Conference*, 3809–3820 (Piscataway, NJ: IEEE).
- Salemi P, Staum J, Nelson BL (2013) Generalized integrated Brownian fields for simulation metamodeling. Pasupathy R, Kim SH, Tolk A, Hill R, Kuhl ME, eds., *Proceedings of the 2013 Winter Simulation Conference*, 543–554 (Piscataway, NJ: IEEE).
- Santner TJ, Williams BJ, Notz W (2003) *The Design and Analysis of Computer Experiments* (New York: Springer).
- Sasena MJ, Papalambros P, Goovaerts P (2002) Exploration of metamodeling sampling criteria for constrained global optimization. *Engineering optimization* 34(3):263–278.
- Sóbester A, Leary SJ, Keane AJ (2005) On the design of optimization strategies based on global response surface approximation models. *Journal of Global Optimization* 33(1):31–59.
- Takahashi K, Fagan J, Chin MS (1973) Formation of a sparse bus impedance matrix and its application to short circuit study. *8th PICA Conference Proceedings*, 16–29 (IEEE Power Engineering Society).
- Vanhatalo J, Vehtari A (2012) Modelling local and global phenomena with sparse gaussian processes. *arXiv preprint arXiv:1206.3290* .
- Williams BJ, Santner TJ, Notz WI (2000) Sequential design of computer experiments to minimize integrated response functions. *Statistica Sinica* 10:1133–1152.
- Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. *Trans. Evol. Comp* 1(1):67–82, ISSN 1089-778X.
- Xie J, Frazier PI, Chick SE (2016) Bayesian optimization via simulation with pairwise sampling and correlated prior beliefs. *Operations Research* 64(2):542–559.
- Xu J, Nelson BL, Hong LJ (2010) Industrial strength COMPASS: A comprehensive algorithm and software for optimization via simulation. *ACM TOMACS* 20(1):3:1–3:29.