AIChE

A Centroid-Based Sampling Strategy for Kriging Global Modeling and Optimization

Eddie Davis and Marianthi Ierapetritou
Dept. of Chemical and Biochemical Engineering, Rutgers-The State University of New Jersey, Piscataway, NJ 08854

DOI 10.1002/aic.11881
Published online October 6, 2009 in Wiley InterScience (www.interscience.wiley.com)

A new sampling strategy is presented for kriging-based global modeling. The strategy is used within a kriging/response surface (RSM) algorithm for solving NLP containing black-box models. Black-box models describe systems lacking the closed-form equations necessary for conventional gradient-based optimization. System optima can be alternatively found by building iteratively updated kriging models, and then refining local solutions using RSM. The application of the new sampling strategy results in accurate global model generation at lower sampling expense relative to a strategy using randomized and heuristic-based sampling for initial and subsequent model construction, respectively. The new strategy relies on construction of an initial kriging model built using sampling data obtained at the feasible region's convex polytope vertices and centroid. Updated models are constructed using additional sampling information obtained at Delaunay triangulation centroids. The new sampling algorithm is applied within the kriging-RSM framework to several numerical examples and case studies to demonstrate proof of concept. © 2009 American Institute of Chemical Engineers *AIChE J,* 56: 220–240, 2010
Keywords: optimization, mathematical modeling, numerical solutions, design, simulation

Introduction

Process design problems are difficult to solve when model equations are unavailable, and noisy input–output data are instead the only known information. Model equations may be unknown because first-principles models have not yet been developed, as in the case of emergent technologies. Alternatively, process models may be inaccessible if they are embedded within legacy codes. For both of these cases in which closed-form equations are absent, the system behavior can be described as being "black-box." Surrogate models can alternatively be generated in place of the black-box models, and although model reliability can be improved using additional sampling information, resource constraints can limit the number of additional experiments allowed. Many substitute models may need to be constructed in the case of process design problems requiring the optimization of a high number of input variables. Since it may not be possible to a priori estimate the problem cost in terms of the number of experiments required, there is a need for strategies targeted at the generation of sufficiently accurate surrogate models at low resource cost.

Two important cases motivate the need to address problems having noise and black-box characteristics: (1) resource allocation during the early product life cycle for funding the most promising research projects, and (2) process train optimization for systems that have been retrofitted, or no longer exhibit the same behavior expected by current operating policies. Problems in which the equations are available, yet mathematically intractable, such as CFD problems, constitute a third possible case in which the problem can be considered

Additional Supporting Information may be found in the online version of this article.

Correspondence concerning this article should be addressed to M. Ierapetritou at marianth@soemail.rutgers.edu

black-box. For these problems, direct equation-solving may be too expensive relative to the optimization of a model generated from input–output data.

Algorithms which address problems having noise and black-box characteristics can also be applied as equation-solving methods. For this problem class, process equations are available, but if multiple solutions exist, equation solvers such as the GAMS solver CONOPT may fail to find the complete solution set. When it is not known a priori how many solutions exist, the application of a solver such as CONOPT is not effective for guaranteeing that all solutions have been found.

A specific diffusion-reaction problem is taken from Lucia and Gattupalli[1] and described in more detail in Diffusion and Reaction in a Catalyst Pellet section. The model equations are given by heat and mass PDEs of a catalyst pellet whose reaction rate can be described by multiple physically meaningful solutions. Equation-solving routines may require specification of additional input conditions, in the form of warm-start iterates or a reduced subregion for search, for the entire solution set to be found. The discovery of all solutions is motivated by a practical issue in which a metric quantifies one solution as the best. For this problem, the solution set consists of a set of possible reaction rates. If only a subset of the solutions are found, and the best possible reaction rate remains undiscovered, this reaction system may be discarded from further study beyond the conceptual design level.

Several modeling and optimization challenges exist when black-box models are present and the only system information is in the form of noisy sampling data. Because of the noise, there is some uncertainty associated with both the surrogate models and the point estimate obtained for the optimum. In the former case, an accurate description of system behavior is not guaranteed, and in the latter case, global optimality cannot be guaranteed. If the surrogate models interpolate the noisy data, the application of derivative-based techniques can lead to the identification of poor search directions because the model has captured the noise instead of the true behavior. However, construction of noninterpolating models can require additional sampling information, thereby increasing the resource costs needed for problem-solving.

To increase the chances of finding the globally optimal solution, a global model should be constructed over the entire feasible region because the optimization of a local model may only lead to a local solution. Global model information can also lead to more rapid identification of all optima, in contrast to local models built over disjoint subregions. The global modeling sampling expense is generally higher than that required for local modeling because the system behavior over the entire feasible region is usually more complex, in terms of mathematical geometry, in contrast to behavior described over a localized area. A quadratic polynomial describing the system response in terms of continuous input variables can sufficiently model system behavior near an optimum. However, this same polynomial may not accurately describe the locations of system ridges, valleys, and stationary points. To accurately capture all these features, a more detailed model is needed which requires additional sampling information. It may not be a priori known how many sampling points are needed for accurate model construction, or where their locations should be. Therefore, the technique of building an initial model using a low number of sampling points, then iteratively refining it using additional sampling information, can be used to control sampling expense. This practice minimizes resource costs attributed to sampling at locations where the contribution to model development is low.

In previous work,[2–4] this strategy has been used whereby accurate global models are iteratively constructed using kriging. At each iteration, the previous global model is updated using additional sampling information collected at points of interest identified from the earlier predictor, such as where the model predictions are lowest, or alternatively where the model uncertainty is highest. The initial model is built from a set of randomly selected feasible points dispersed throughout the feasible region. Different nominal models are generated from different initial sampling sets. However, the initial sampling data may fail to contain the information needed for the kriging-based global modeling algorithm to accurately identify important system behavior in the form of ridges, valleys, optima, or stationary points. In this case, the initial model will be poor, and a higher sampling expense may be required to obtain an accurate model.

As a result, based on the current sampling strategy of performing (1) random sampling for initial modeling, and (2) heuristic-based sampling at model-based points of interest, there exists an opportunity for reducing the global modeling sampling expense. The goal of reducing sampling expense is motivated further by the need to eliminate random sampling as a method for initial modeling. The attractiveness of a surrogate model-building technique is increased if it can be known a priori how many sampling experiments are needed before an accurate model can be reasonably expected. Because random sampling is inherently uncertain, the number of sampling experiments required for accurate estimator construction can vary over a wide range. A technique independent of random sampling eliminates the question of how much additional sampling expense was required due to the initial set having been chosen under uncertainty.

The contribution of this article is the presentation of a sampling technique that relies on sampling at Delaunay triangle centroids. The new technique has been successfully used in attaining accurate global kriging models, and the main advantages of the method are that it is possible for (1) a complete set of local and global optima to be found for NLP containing black-box models and/or noisy variables, although this cannot be guaranteed, and (2) a complete set of optima to be obtained at lower sampling expense relative to a randomized/heuristic-based sampling strategy. This centroid-based sampling technique is proposed as a novel sampling approach for other sampling-based modeling methods. The main features of the new sampling technique involve (1) sampling at the vertices and centroid of the feasible region's convex polytope for initial modeling, and (2) sampling at the Delaunay triangle centroids for model refinement.

Literature Review

In the field of global optimization, Floudas et al.[5] present a review of the state-of-the-art techniques. As the majority of the gradient-based techniques assume knowledge of

closed-form models, they cannot be directly applied to the problem class addressed in this article because the formulation contains black-box models. A similar problem has been addressed by Meyer et al.[6] for NLP containing nonanalytical constraints that are both known and differentiable. For this problem class, the solution is obtained using a global model whose form is that of a blending function from which over- and under-estimators can be generated to provide ε-guarantee of global optimality.

Zero-order techniques such as Nelder-Mead,[7] Divided Rectangles,[8] multilevel coordinate search,[9] and differential evolution,[10] can be used to find NLP solutions of problems containing black-box models and noise. Unfortunately, convergence to the optimum can be slow, motivating the use of gradient-based techniques classified as first- or second-order algorithms depending upon whether first- or second-order derivative information is used to find optima. However, the limitations of first- and second-order techniques may be either asymptotic convergence to true optima or premature termination at artificial noise-induced optima. The latter problem has been addressed using modified gradient-based techniques whereby finite differences of large enough interval length are used to filter out noise.[11–13] At the same time, model-based optimization techniques can overcome the limited understanding of system behavior that results when nonmodel-based optimization methods such as the above-named zero-order methods are used. RSM belongs to a class of local methods in which gradient techniques are sequentially applied to inexpensively fitted local polynomial models known as response surfaces.[14] Each response surface is a low-order model that is minimized to find the next candidate solution. A new response surface centered at the new iterate is also built and minimized. This procedure is repeated until a convergence criterion such as a prespecified decrease in the objective function has been attained. Sampling expense is the primary source of the optimization cost since the response surfaces can be built cheaply using least-square fitting. In our recent work, an algorithm using (1) adaptive experimental designs to retain feasibility, and (2) projection of the n-dimensional response surface onto constraints, has been successfully applied toward identifying NLP solutions in higher dimensions described by arbitrary convex feasible regions.[4] In addition to local optimization algorithms, global response surface methods have also been studied for box-constrained problems, with a larger basis function set being used for response surface construction.[15–18]

Unfortunately, local techniques such as RSM ensure a global optimal solution only under convexity conditions, which cannot be known in advance for problems containing black-box functions. As a result, these approaches are inefficient for solving nonconvex NLP as only local optima may be found. This problem can be addressed by first building global models in order to identify the best areas to apply the local methods. Kriging is one popular global modeling technique that comes from geostatistics. In kriging, the black-box model outputs are treated as realizations of normally distributed random function models. The technique is interpolation-based whereby a prediction and its variance at a test point are obtained according to a weighted sum of the nearby sampling data. The methodology has seen increasing use because an uncertainty mapping can be obtained in addition to the global model. Kriging originated in 1951[19] and was initially used to describe the spatial distribution of mineral deposits in mining applications based on physical sampling data. Kriging has also been used to generate computer experiment models when simulation is used to describe complex processes.[20] The main limitation of kriging is that model building costs are higher compared to response surface methods since a more detailed description of process behavior is obtained, due to the generation of predictions for a larger set of feasible vectors. However, once a global model is determined to be accurate, local optimization of the best kriging solutions using RSM can lead to discovery of the complete solution set.

The field of sampling-based techniques focuses on optimizing both the test set at which predictions are to be obtained in addition to the number and spatial arrangement of sampling points. The designation "sampling techniques" can refer to either (1) an algorithm specifically targeted at identifying where field/computational experiments should be performed, as is the goal of the proposed method in this article, or alternatively, (2) a method for identifying the locations where model predictions should be generated. Considering the first class of sampling techniques, four main algorithmic subclasses exist for choosing any n_p samples from a set of k_{Test} feasible vectors for modeling: (1) random, (2) systematic, (3) stratified, and (4) cluster. Random sampling is easily implementable but has the limitation that the information obtained from a sampling set poorly representing the feasible region may fail to result in accurate model development. Systematic sampling relies on a set of heuristics for identifying sampling vectors, such as every seventh feasible point from the set of ordered k_{Test} points. A key limitation of this technique is that a poorly chosen value for n_p can result in too few sampling experiments being performed, resulting in the generation of an inaccurate model. Conversely, too many sampling experiments may be conducted in the sense that redundant system information is identified from sampling points located close to one another. The centroid-based sampling technique used in kriging modeling is an example of a systematic sampling procedure. For this method, the sampling expense is controlled by sampling at a single point within any Delaunay triangle, and terminating the algorithm after the third or fourth iteratively improved global model has been built.

Stratified sampling is the application of any other sampling algorithm to each one of a subset of the k_{Test} points separated by some stratification rule. The advantage of this technique is that a more uniform sampling arrangement is generated relative to random sampling. Cluster sampling requires sampling to be performed for all sampling vectors in a given strata of k_{Test} points at the exclusion of sampling being performed in other strata. The advantage of cluster sampling is that extensive system information can be obtained over a localized subregion of interest, such as when neighborhoods containing potential optima are identified and additional sampling information is needed to refine the current solution. The sampling templates used in RSM, given in Figure 9, can be considered as the result of a cluster sampling algorithm because no sampling is performed outside the subregions of interest.

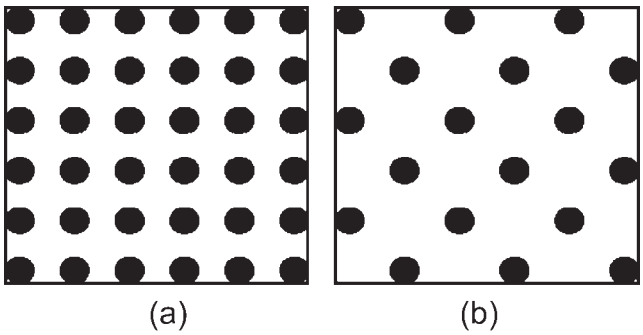

Figure 1. Generation of a test point set based on discretization (a) or the application of a Latinized/Hammersley/Halton algorithm (b).

For the class of methods directed at the identification of the locations of all k_{Test} points requiring model generation, the generation of model estimates at discretized feasible region locations as shown in Figure 1a is the most intuitive and easily implementable algorithm. However, this method becomes impractical as the problem dimensionality increases because the number of test points is a multiplicative function of the number of grid points generated for each dimension. Since one goal of modeling techniques is that the modeling algorithm itself be as computationally inexpensive as possible, alternative pseudo-uniform sampling schemes such as Latinized, Hammersley, and Halton sampling as shown in Figure 1b have been used instead. These algorithms were initially applied to problems whose feasible regions were predominantly hypercuboidal, but have since been extended to other symmetrical regions.

The Centroidal Voronoi tessellation techniques applied by Romero et al.[21] are used to generate pseudo-uniform sets using modified Voronoi diagrams when the problem dimensionality increases, a limitation of the aforementioned sampling techniques. The Voronoi diagram is considered to be the dual of a Delaunay triangulation. However, in contrast to the Romero method, the proposed sampling method in this article relies on sampling at Delaunay simplex centroids instead of the corresponding Voronoi simplex centroids. The number of simplex vertices in a Delaunay triangle is fixed at $(n + 1)$, where n is the problem dimensionality, whereas for a Voronoi simplex it is often greater than $(n + 1)$. Some of the Voronoi simplices will have a small hypervolume, while others will have a much larger one as shown in Figures 2a–c. The volume of the simplices formed by applying Delaunay triangulation, on the other hand, have approximately equal volume, as shown in Figures 2d–f. Therefore, the application of a Voronoi-based sampling technique requiring samples to be obtained at the centroids of iteratively updated Voronoi diagrams can result in redundant sampling information being obtained. This occurs as a result of sampling points being close to one another between neighboring Voronoi simplices as shown in Figure 2c. As mentioned previously, in the new sampling method, this problem is addressed by terminating

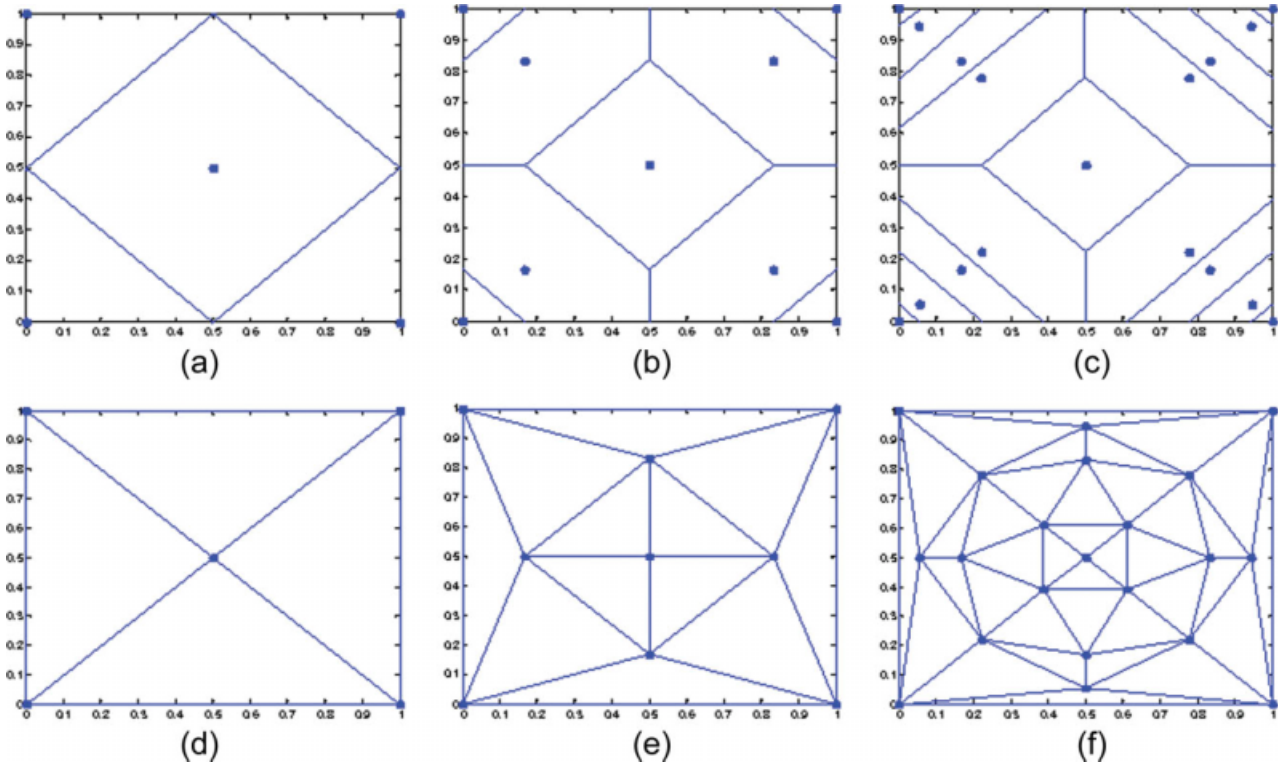

Figure 2. Generation of sampling points concentrated along feasible region diagonals when sampling is performed at the simplex centroids for a set of iteratively updated Voronoi diagrams (a–c), a nonuniform sampling arrangement, in contrast to the more uniform sampling arrangement obtained from sampling at simplex centroids of iteratively updated Delaunay triangulations (d–f).

the kriging algorithm after the initial model has been refined only three or four times.

One main advantage of using the pseudo-uniform sampling schemes is that an approximately uniform sampling scheme can be generated for an arbitrary number of samples N. As an example, the Aszkenazy implementation of Hammersley sampling generates N sampling points according to a univariate function $F(Pr(i))$ of the problem dimensionality p as shown by Giunta et al.[21] If the variable i represents the ith-dimensional vector component of any one of the N sampling points, the variable $Pr(i)$ is a number that is an offset of the prime number sequence. For $i = 1$, $Pr(i)$ is zero, and for $i > 1$, $Pr(i)$ is the $(i\text{th} - 1)$ prime number, starting with two. If the problem dimension is five, $Pr = \{0, 2, 3, 5, 7\}$. All N points along the ith dimension will be generated as a function of $F(Pr(i))$. For $i = 2$, the N points corresponding to the x_2-component of the vector $\{x_1, x_2, x_3, x_4, x_5\}$ are obtained according to $F(2)$. Similarly, for $i = 4$, the N points corresponding to the x_4-component are obtained according to $F(5)$. The details of the closed-form function $F(Pr(i))$ are found in Giunta et al.[22]

The flexibility gained from generating a pseudo-uniform sampling scheme for any number of sampling points is an advantage over the fixed number of sampling points obtained using the centroid-based sampling techniques. For the centroidal sampling methods, the number of sampling points is fixed by the number of Delaunay triangles or Voronoi tessellation regions obtained with respect to the previous sampling points within the feasible region. At the same time, for black-box systems, the sampling number flexibility of pseudo-uniform sampling schemes is limited when it may not be known a priori how many samples are required to generate an accurate model.

For the algorithm presented in this article, sampling data are collected at a specific number of points and used to improve a model at each iteration. The number of sampling points generated for model improvement increases for each iteration as the number of Delaunay triangles increases as a function of the number of previously sampled points. As an example, for a 2D box-constrained problem, five sampling points are used for initial modeling, four to improve the first model, 12 to improve the second model, and so on. For the test cases presented in the Examples section, it is assumed that all samples can be obtained on demand.

If resource restrictions only allow sampling data to be collected for a subset of the specified number of points, the algorithm's efficiency is expected to decrease. A maximin sampling scheme might prove more effective at generating the same pseudo-uniform sampling scheme attained by sampling at Delaunay triangle centroids, although this is not investigated within the scope of this article. One advantage of our algorithm is that it is designed to work for generalized convex feasible regions, whereas the corresponding Hammersley/Halton/Latinized sampling methods have been mainly applied to hyper-rectangular or hyper-spheroidal regions. The application of these latter methods to generalized convex feasible regions requires a function that maps the set of N points generated for a p-dimensional hypercube or hyperspheroid to the current feasible region. This mapping can be particularly useful in the case of higher-D problems when the model must be constructed from a limited number of vectors to avoid the prohibitive CPU modeling expense associated with generating estimates obtained at vectors corresponding to fine discretization.

Other sampling designs optimize some other estimator property, such as minimization of estimator variance at a given subset of points of interest, maximization of the estimator confidence region, or a user-designed merit function created for comparison of multiple design alternatives. Modified central composite designs, squared-error designs, orthogonal arrays, and minimax designs are a few of the designs reviewed in more detail in Simpson et al.[23] In the next section, the targeted NLP problem is formulated. The details of the centroid-based sampling method, as implemented within a kriging-RSM for solving this NLP follow.

Solution Approach

The problem addressed in this work can be expressed in the following form:

$$\begin{aligned} \min \quad & F(x, z_1, z_2) \\ \text{s.t.} \quad & g(x, z_1, z_2) \le 0 \\ & h(x, z_1) = 0 \\ & z_2(x) = (1 + N(\mu, \sigma^2))\Gamma(x) \\ & x \in \Re^n, 2 \le n \le 5 \end{aligned} \tag{1}$$

Based on this formulation, the vector of continuous variables is given by x. The objective function is represented by F, and the deterministic variables z_1 describe outputs whose modeling equations $h(x, z_1)$ are known. The vector of stochastic output variables z_2 exists when the input–output functionality $\Gamma(x)$ is black-box. Each z_2 variable is modeled as having a mean corresponding deterministic output value perturbed by an additive noise component. One sample model for the noise is that it behaves according to a normal distribution, and in this case, an estimate of the parameters μ and σ^2 can be obtained by conducting replicate experiments for a given sampling vector. The modeled values of μ and σ^2 may need to assume a range of values if it is known that the noise is spatially variant with respect to x. Design and operating equations are given by $g(x, z_1, z_2)$, where the feasible space for x may be further constrained based on the feasible space defined by $z_2(x)$. In our previous work,[2] a Branch-and-Bound Kriging-RSM-Direct Search algorithm has been applied to process synthesis problems described by integer variables appearing both inside and outside the black-box function. The formulation of this problem class is the analog of (1) which would now contain two integer variable sets (y_1, y_2) as additional dependent variables for F, g, h, and z_2. Both integer variable sets are required to separately designate the integer variables residing inside or outside of Γ. Even though the sampling strategy presented in this article is applied to problems containing only continuous variables, a future work would be the extension of our new method to this more general problem class.

The optimization strategy used to obtain the solution of the problem described by Eq. 1 relies on the generation of an iterative sequence of kriging global models in order to identify promising warm-start iterates for local optimization. At each stage of the kriging algorithm, new sampling data

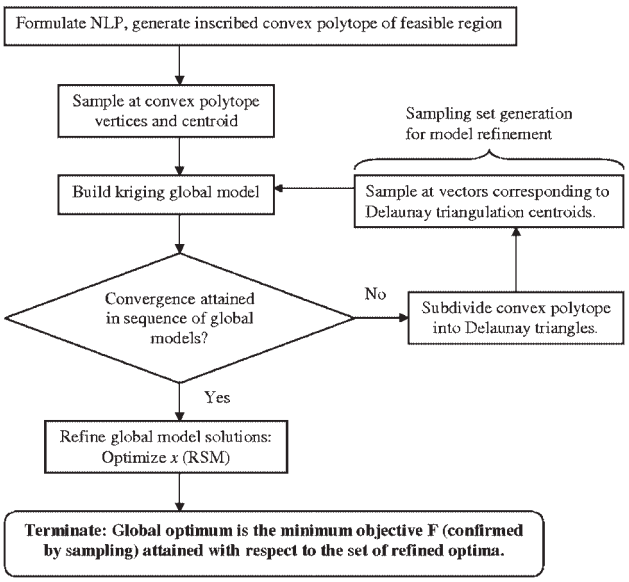

Figure 3. Flowchart of the centroid-based sampling strategy embedded within a kriging-RSM modeling/optimization algorithm; this technique is designated as the KC-RSM method.

are used to update the previous model. Over the course of applying kriging, a sequence of global models is generated, with the late models closely resembling one another. Once the system predictor has converged, the global model is considered accurate and the best local solutions are then identified for further local optimization via RSM. The application of RSM to a subset of, if not all, the kriging model optima improves the chances of both finding a global optimum, and also establishing confidence in classifying it as a global optimum due to the identification of additional local optima having inferior objective values F. The converged kriging model is intended to accurately reproduce the shape of the actual function with all its minima. However, since no theoretical guarantees can be made that the converged kriging model always captures the true function, another global optimization algorithm having global convergence properties can be applied to help ensure that all optima have been identified. When centroid-based sampling is used for global model construction and refinement, the kriging-RSM algorithm is denoted as the KC-RSM algorithm. A flowchart is shown in Figure 3.

The basic steps for building and refining global models, based on the incorporation of sampling information obtained at Delaunay triangle centroids, are as follows: (1) sample at the feasible region's convex polytope vertices and its centroid, (2) build the global model, (3) sample at Delaunay triangle centroids, and (4) repeat steps (2) and (3) until the sequence of global models has converged. In Figure 4, sampling templates are presented which correspond to the construction and refinement of a 2D global model whose output variable z_2 is a function of two variables x_1 and x_2, for a box-constrained feasible region.

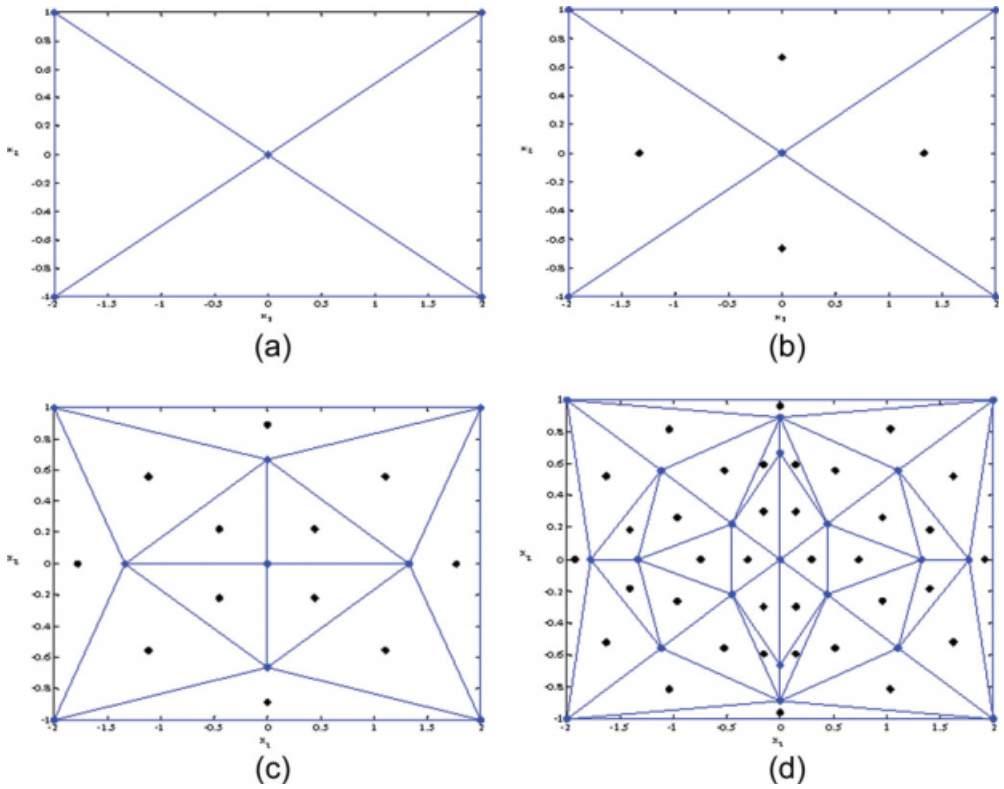

Figure 4. Generation of sampling templates used to build kriging models for a 2D box-constrained problem, in which five sampling points are used for initial model construction (a), and four (b), twelve (c), and thirty-six (d) additional sampling points are used in subsequent model-building.

In Figure 4a, the Delaunay triangulation of the initial sampling set is shown. It is supposed that once the initial model has been built, it is not yet considered to be an accurate estimator, and therefore requires refinement using new sampling information. The new sampling set is comprised of vectors corresponding to the Delaunay triangle centroid locations, designated by the interior points as shown in Figure 4b. If the second model, built using the set of nine sampling points, is still considered inaccurate, another Delaunay triangulation is performed. The centroids of the Delaunay simplices are shown in Figure 4c, and the number of sampling points in the new sampling set is 12. If the third model, constructed from 21 sampling points, still requires refinement, Delaunay triangulation is performed yet again. The 36 Delaunay triangle centroids are shown in Figure 4d. After the fourth iteration, it is supposed that for this hypothetical system the sequence of global models has converged. At this point, the set of local kriging solutions are identified, and further local optimization using RSM is initiated.

Locally optimal kriging solutions are defined as the vectors for which the objective function value F is lower than the corresponding objectives for the set of nearest neighbors. For all the presented examples, the test point set is generated using discretization. The generation of a gridded test point set can result in a very high number of test points for dimension $n \geq 10$, increasing the modeling/optimization computational expense. In addition, at each iteration of the centroid-based kriging algorithm, a call is made to the Matlab 2008b implementation of the Delaunay triangulation algorithm. This particular version of the Delaunay triangulation algorithm relies on convex hull construction using qHull,[24] whose computational complexity increases rapidly as a function of problem dimension and therefore is nontrivial when the problem dimensionality is greater than 10. If n_p is defined as the total number of sampling points, Barber et al.[24] conjecture that the computational time required to obtain the convex hull of r points having a maximum number of facets f_r is $O(n_p \log r)$ for problems of dimensionality equal to or less than three, and $O(n_p f_r/r)$ when the problem dimensionality is greater than four. The computational complexity required to determine the number of facets based on r points for a problem of dimension n is determined according to the following equation:

$$f_r = O\left(\frac{r^{\lfloor n/2 \rfloor}}{\lfloor n/2 \rfloor !}\right) \tag{2}$$

A more efficient Delaunay triangulation algorithm is required to alleviate this problem, and therefore, the current version of the KC-RSM algorithm has so far been efficient only when applied for problems of low dimensionality (less than 10 input variables). Further details of the centroid-based sampling strategy, kriging, and RSM will now be presented.

Centroid-Based Methodology

The initial kriging model is built using sampling data obtained at the vertices of the feasible region's convex polytope vertices and centroid. For an n-dimensional problem, the set of M convex polytope boundaries can be given in terms of a matrix formulation:

$$Ax \leq b$$

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix} \qquad m = 1 \ldots M \tag{3}$$

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \qquad m = 1 \ldots M$$

A vector x is a convex polytope vertex if it satisfies Eq. 4 for any n constraints and is also feasible with respect to the others.

$$\tilde{A}x = \tilde{b} \tag{4}$$

where $\tilde{A}$ is an arbitrary $(n \times n)$ submatrix of A and $\tilde{b}$ is the corresponding $(n \times 1)$ submatrix of b. In Figure 5, the convex polytope's vertices are shown as the vertices of a diamond-shaped feasible region.

Once all convex polytope vertices have been obtained, the next step is to generate the centroid of these points. Let the set of N_V convex polytope vertices be denoted as V, and let the ith vertex be denoted as $x_{V,i}$. The corresponding convex polytope centroid can then be obtained as the arithmetic mean of the vertex points as given in the following equation:

$$x_C = \frac{\sum_{i=1}^{N_V} x_{V,i}}{N_V} \tag{5}$$

For a convex polytope, its centroid will always reside within the feasible region. The strategy of including the convex polytope centroid as a sampling point for initial modeling is motivated by the idea that the system information obtained from sampling at this point is equivalent to the information obtained from sampling at a sequence of points converging to the convex polytope centroid. Once x_c has been determined, the sampling set S^1 used to build the initial kriging-based global model is defined as follows:

$$S^1 = \{V, x_c\} = \left\{ x_{V,i}\big|_{i=1 \ldots N_V}, x_c \right\} \tag{6}$$

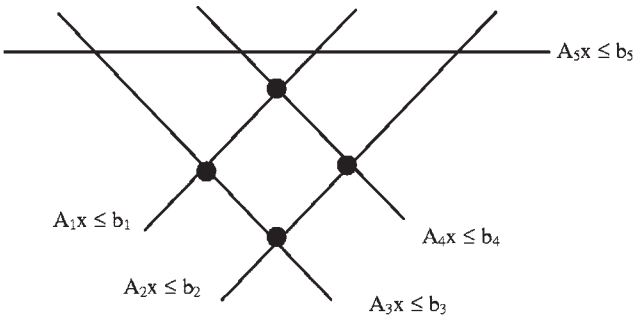

Figure 5. Application of a feasibility check to identify the set of sampling points located at the convex polytope vertices.

Table 1. Differences Between the Two Sampling Strategies for Building Iteratively Updated Global Kriging Models

Model Creation	Old Sampling Strategy Sample Randomly	New Sampling Strategy Sample at Convex Polytope and Centroid
Model Refinement	Sample at locations where there is: 1) Minimum model prediction 2) Maximum model uncertainty 3) Highest current/previous model discrepancy	Sample at each one of the N_D Delaunay triangulation centroids

Once the initial kriging model is generated, it may not accurately describe the system behavior within the feasible region. Model inaccuracy occurs in part because kriging is an interpolatory method whereby predictions are generated as a weighted sum of nearby sampled function values. Kriging model behavior resembles the behavior exhibited by models built using inverse distance weighting methods, in that model accuracy generally decreases as a function of increasing distance between a test point and the nearby sampling points. The centroid location is likely to be far away from the convex polytope vertices, and model inaccuracy will be highest at test points approximately midway between the centroid and the vertices. To attain significant model improvement, additional sampling is needed at locations where the uncertainty is highest.

To address this problem, a Delaunay triangulation is applied to the current sampling set S^m, where S^m refers to the additional sampling data used together with the S^{m-1} dataset to build the mth kriging model as presented in the kriging algorithm flowchart shown in Figure 8. The Delaunay triangulation algorithm subdivides the feasible region into a set of adjacent, nonoverlapping simplicial subregions having as close to uniform volume as possible. The Delaunay triangulation of a convex polytope having k possible subdivisions will be the one that contains the highest minimum simplicial angle. Each simplicial subregion contains $(n + 1)$ vertices serving as the vectors used in the corresponding centroid calculation given by Eq. 5. The new sampling set S^{m+1} is then generated as the set of all Delaunay triangulation centroids.

In Table 1, the differences between the current and new sampling algorithms are presented. Either one of the sampling algorithms can be applied within the framework of the kriging method, presented in the next subsection, to generate iteratively refined global models.

One feature of the proposed algorithm is that the number of sampling points required at each iteration is strictly identified as the number of Delaunay triangle centroids. This sampling strategy provides a suitable sampling framework for iterative modeling of black-box systems whereby models are improved using sampling data obtained at new Delaunay triangle centroids. For black-box systems, it is not known a priori how many samples are needed before an accurate model will be generated. Therefore, the strategy of building intermediate models limits the amount of sampling used, while also leaving open the possibility that an accurate model may be obtained using fewer points relative to an arbitrary maximum number allowed. In addition, a comparison of the previous and current models can be used to identify local regions where the model has changed significantly. Kriging models built from sampling at these locations, in addition to the Delaunay triangulation centroids, can result in the generation of accurate predictors more rapidly as opposed to when sampling information comes solely from data obtained at the centroids.

The goal of finding all the problem's minima using kriging relies on the kriging model accurately reproducing the true functionality. The model is considered accurate if the average predictor value matches, within a given tolerance $\mathrm{Tol}_{\mathrm{Krig}}$, the corresponding average value for the previous model. However, it is possible that this test may be satisfied even when the current model still fails to capture the deterministic behavior. This can occur when an initial sampling set is poorly chosen, such as via random sampling, and a subsequent sampling set yields redundant information leading to negligible model change.

Kriging Methodology

The kriging model consists of predictions obtained at all test points x_k, in which each kriging estimate is a weighted sum of nearby sampled function values. The kriging model is built with respect to the NLP objective because it is the objective function that is being optimized instead of the process outputs. The steps for obtaining a prediction at x_k are as follows: (1) determination of covariance function coefficients based on sampling data; (2) calculation of the covariance $\mathrm{Cov}(d_{i,k})$ between the test point and each nearby sampling point; (3) generation of weighting values λ for each sampling point x_i close to x_k after solving the linear system $C\lambda = D$, where the elements of C and D are $\{x_i, x_j\}$ and $\{x_i, x_k\}$ covariance values, respectively; and (4) estimation of the kriging predictor. Additional details for each step follow.

From the sampling data contained in the current sampling set S^m, where m denotes the iteration index, squared output value differences $F_{i,j}$ are first calculated as given by Eq. 7, and then plotted relative to sampling-pair distances as given by Eq. 8:

$$F_{i,j} = \left[F(x_i) - F(x_j)\right]^2 \quad i,j = 1\ldots\mathrm{card}(S^m), \quad i \neq j \tag{7}$$

$$d_{i,j} = \|x_i - x_j\|_2 \tag{8}$$

From a scatterplot of $F_{i,j}$ as a function of $d_{i,j}$, a semivariance function is then fitted. Because of the plot complexity as shown in Figure 6a, the best fit to one of the established semivariance models in the literature[19] may not be immediately apparent.

To alleviate this problem, data smoothing is applied, and the semivariance function is then fitted to the reduced set of

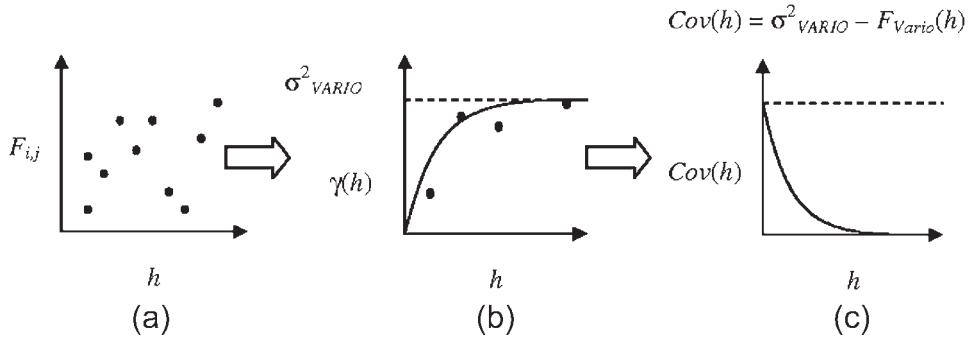

Figure 6. Data smoothing applied to squared function differences $F_{i,j}$ (a) in order to obtain a semivariogram fit (b) and covariance function fit (c).

scatterpoints known as semivariances γ as shown in Figure 6b. A set of P equidistant intervals are defined between zero and the maximum $d_{i,j}$ distance. The pth interval midpoint is denoted by h_p, and the semivariance corresponding to the pth interval, $\gamma(h_p)$, is obtained by averaging the set of squared function differences falling inside this interval as given by Eq. 9:

$$\gamma(h_p) = \frac{1}{2N(h_p)} \sum_{r=1}^{N(h_p)} F_{i,j} \quad p = 1 \ldots P, \;\; i \neq j \tag{9}$$

where $N(h_p)$ is the number of sampling pairs $\{x_i, x_j\}$ whose separation distance $d_{i,j}$ lies inside the pth interval. The semivariance function behavior typically rises from zero to an asymptotic maximum known as the sill σ^2_{VARIO}. When the semivariance function is reflected between the x-axis and the sill, the corresponding covariance function is obtained as displayed in Figure 6c. The covariance between any two x_i-x_j or x_i-x_k vectors can now be determined by substituting $d_{i,j}$ or $d_{i,k}$ into the covariance function. The kriging weights are then obtained as the solution of a linear system of equations in which the LHS consists of a matrix of $\{x_i, x_j\}$ covariances, and the RHS consists of a vector of $\{x_i, x_k\}$ covariances between k_{Cluster} nearest-neighbor sampling points x_i and x_k. If the weights are forced to sum to unity, the linear system can be recast in a Lagrangian formulation as given by Eq. 10:

$$\begin{bmatrix} \lambda(x_k) \\ \lambda'(x_k) \end{bmatrix} = \begin{bmatrix} \text{Cov}(d_{i,j}) & 1 \\ 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} \text{Cov}(d_{i,k}) \\ 1 \end{bmatrix} \quad i,j = 1 \ldots k_{\text{Cluster}}, \quad i \neq j \tag{10}$$

where $\lambda(x_k)$ and $\lambda'(x_k)$ represent the weight vector and the Lagrange multiplier, respectively. Once the weights are obtained, the kriging prediction $F(x_k)$ and its expected variance $\sigma^2_k(x_k)$ are obtained according to Eqs. 11 and 12, respectively:

$$F(x_k) = \sum_{i=1}^{k_{\text{Cluster}}} F(x_i)\lambda(x_i) \tag{11}$$

$$\sigma^2_k(x_k) = \sigma^2_{\text{VARIO}} - \sum_{i=1}^{k_{\text{Cluster}}} \lambda(x_i)\text{Cov}(d_{i,k}) - \lambda'(x_k) \tag{12}$$

The methodology is then applied to another test point until objective function estimates have been obtained for all test points. If additional sampling data are obtained, a new covariance function can be generated. From this new function, a refined global model can be created once new kriging estimates have been obtained for all k_{Test} test points. For each global model, its corresponding average predictor value μ is compared against its counterpart based on the previous model. Once convergence has been achieved in μ, further refinement is terminated.

Let the iteration index m refer to any property based on the mth kriging model, and let Tol_{Krig} be a percentage stopping tolerance. A sample range for Tol_{Krig} would be any value between 1 and 10%. The mth average prediction value μ_m is defined as the average of the set of kriging predictions and sampled function values as given by Eq. 13:

$$\mu_{\text{m}} = \frac{1}{\left(\text{card}(S^m) + k_{\text{Test,m}}\right)} \left(\sum_{i=1}^{\text{card}(S^m)} F_i(x_i) + \sum_{r=1}^{k_{\text{Test,m}}} F_r(x_k) \right) \tag{13}$$

where $\text{card}(S^m)$ refers to the number of sampled function values in the current sampling set S^m and $k_{\text{Test},m}$ refers to the number of test points used in constructing the mth global model. A nominal value μ_0 is obtained by averaging the sampled function values from S^1. Once the mth global model has been constructed, μ_m/μ_{m-1} is evaluated. If this ratio falls inside the interval $(1 \pm \text{Tol}_{\text{Krig}})$, the mth global model is considered accurate, and no additional updating occurs. Specifically, the convergence criterion is that both of the conditions $\mu_m/\mu_{m-1} > 1 - \text{Tol}_{\text{Krig}}$ and $\mu_m/\mu_{m-1} < 1 + \text{Tol}_{\text{Krig}}$ must be satisfied to terminate further model updating. If these conditions are not met, another model is built using additional sampling data. The locations of new candidate sampling points are obtained using either one of the two methods described in Table 1. When the noncentroid-based sampling method is used, global model improvement is enforced by requiring that all new sampling points reside some minimum distance apart from one another. This practice ensures that the new sampling set will not consist of clustered points located at a single high-variance region, thereby de-emphasizing local model refinement. The global models presented in Figure 7 show the kriging predictor stabilizing after iterative refinement.

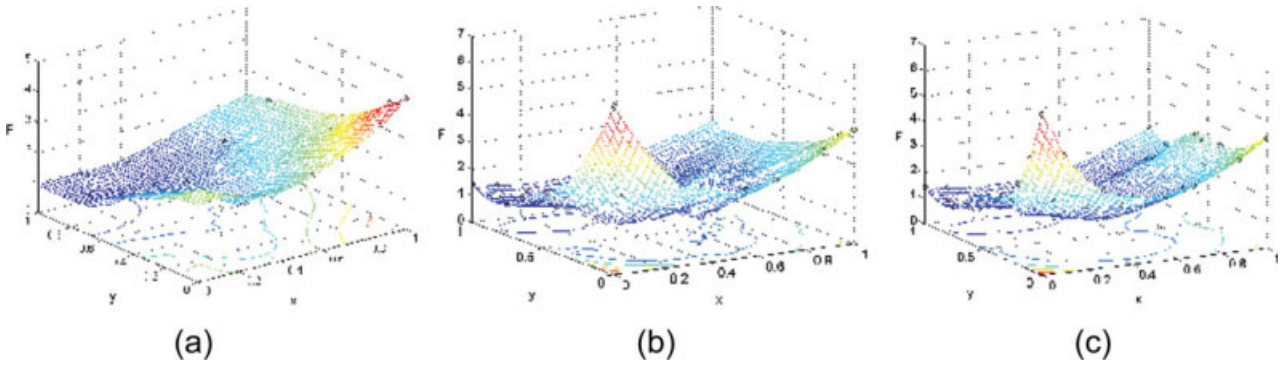

Figure 7. Kriging model generated at initial (a), intermediate (b), and final (c) stages.

[Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

The procedure for obtaining a prediction at x_k, referred to as the kriging algorithm, is now summarized. First, the feasible region is characterized and iteration index m is initialized at unity. A nominal sampling set S^1 is specified using one of the two techniques presented in Table 1. The location x_k is specified and k_{Cluster} nearest-neighbor sampling points are chosen from S^1 that are nearest to x_k as given by Eq. 8. If the centroid-based sampling technique is used, the value of k_{Cluster} is set at $(n + 1)$, otherwise for the noncentroid-based method, referred to as the KNC-RSM algorithm, it is a user-determined parameter set at seven for the presented examples. Semivariances are then generated from all sampling data. The best semivariance model is fitted using least squares, and the complementary covariance function is then obtained. The matrices on the RHS of Eq. 10 are then constructed from submatrices h_{Cluster}, h_0, C, and D, as given by Eqs. 14–17. The matrices C and D are augmented in order to remain consistent with the Lagrangian formulation given in Eq. 10.

$$h_{\text{Cluster}} = \begin{bmatrix} 0 & d_{1,2} & \cdots & d_{1,k_{\text{Cluster}}} \\ d_{2,1} & 0 & \cdots & d_{2,k_{\text{Cluster}}} \\ \vdots & \vdots & \ddots & \vdots \\ d_{k_{\text{Cluster}},1} & d_{k_{\text{Cluster}},2} & \cdots & 0 \end{bmatrix} = [d_{i,j}]$$
$$i,j = 1 \ldots k_{\text{Cluster}} \quad (14)$$

$$h_0 = \begin{bmatrix} d_{1,k} \\ d_{2,k} \\ \vdots \\ d_{k_{\text{Cluster}},k} \end{bmatrix} = [d_{i,k}] \quad i = 1 \ldots k_{\text{Cluster}} \quad (15)$$

$$C = \begin{bmatrix} \text{Cov}(h_{\text{Cluster}}) & 1 \\ 1 & 0 \end{bmatrix} \quad (16)$$

$$D = \begin{bmatrix} \text{Cov}(h_0) \\ 1 \end{bmatrix} \quad (17)$$

Once the kriging weights λ have been obtained from solving the system of equations given in (10), the prediction $F(x_k)$ and its variance $\sigma_k^2(x_k)$ are determined. The weights are then recalculated for each one of the remaining k_{Test} sampling vectors to generate corresponding estimates for $F(x_k)$. Once the global mapping has been constructed, the value of μ_m is determined from Eq. 13 and compared against μ_{m-1}. If convergence is not achieved, the iteration index is advanced by unity and additional sampling is performed based on application of the sampling rules as given in Table 1. A new covariance function is generated, new kriging estimates $F(x_k)$ are generated, and an updated mapping is built. The procedure is terminated once convergence has been achieved in μ_m. The best kriging solutions are then locally optimized using RSM. A flowchart of the kriging algorithm is presented in Figure 8.

Local Optimization Using RSM

The response surfaces used in this article are quadratic polynomials containing bilinear terms which are easily

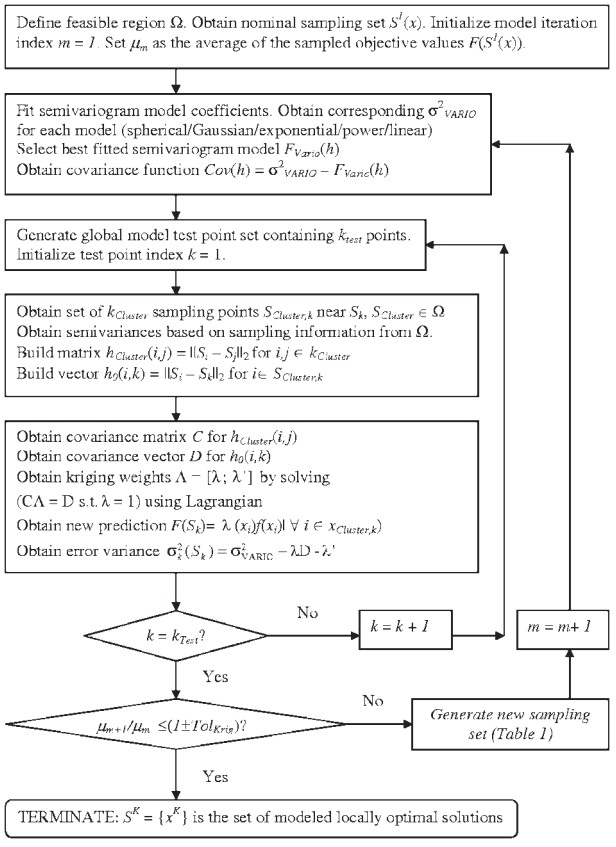

Figure 8. Kriging algorithm flowchart for building/refining a data-driven global model.

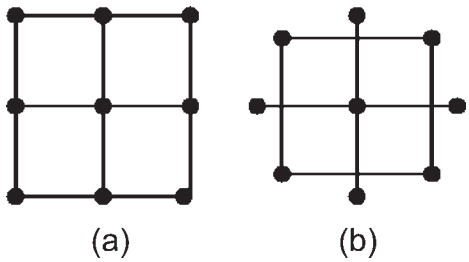

Figure 9. Factorial (a) and central composite design (b) for 2D response surface generation.

optimized using standard gradient techniques. If additional sampling at the response surface optimum yields a better solution, a new model is then built which is centered around this iterate, and the process continues until the objective has converged. The set of input points for building the response surface conforms to a stencil arrangement known as an experimental design that is centered about an iterate.[25] When RSM is applied after an accurate kriging model has been built, the iterate is any one of the best kriging solutions.[2–4] The factorial design for a 2D problem requires a^b points, where a is the number of factors and b is the number of continuous input variables. For problems whose feasible regions are described by linear constraints, a factorial design, shown in Figure 9a for a 2D problem, enables system behavior along a linear boundary to be effectively described.

An alternative to the factorial design is the central composite design (CCD), shown in Figure 9b for a problem in two dimensions, which requires $(1 + 2n + 2^n)$ sampling points for a problem of dimension n. Response surfaces built according to the factorial design are defined over an n-D hyper-rectangular region. In contrast, response surfaces built according to the CCD are instead defined over an n-D spherical region. When iterates are located along a linear boundary, the system behavior is captured more effectively if the response surface is built according to the factorial design. However, the CCD design is associated with a lower sampling expense relative to a factorial design since data are not obtained at every factor-level combination.

At the start of the algorithm, the iteration index w is initialized at a value of unity. A response surface is built around a local kriging solution S^K by fitting sampling data obtained from a collocation set $S_{\text{coll},w}$. The vectors comprising $S_{\text{coll},w}$ are determined by applying either one of the factorial or CC design templates for a predetermined initial model radius b_w. For all the presented examples, the nominal value of b_w is set at 10% of each variable's operability range as defined by the difference in corresponding lower and upper bounds. The vector S^K and its corresponding objective value F^K comprise the nominal solution set $\{S_{\text{opt},w}, F_{\text{opt},w}\}$. Once the response surface has been created, the optimum $S_{\text{opt},w+1}$ having corresponding value $F_{\text{opt},w+1}$ is determined using gradient methods. Sampling is performed at the model optimum vector to confirm objective value improvement. If the difference between the current and previous optimum $|F_{\text{opt},w+1} - F_{\text{opt},w}|$ falls below a prespecified criterion Tol_{RSM}, the algorithm terminates with $[S_{\text{opt},w+1}, F_{\text{opt},w+1}]$ established as the RSM solution. Otherwise, the iteration index is advanced by unity and another response surface having a new bound radius b_w is constructed at the new $x_{\text{opt},w}$. At any iteration w, the value of b_{w+1} is different from b_w only if the Euclidean distance between the current and previous solution vectors is lower than the current radius b_w. During the later stages of the algorithm, $S_{\text{opt},w+1}$ will be near $S_{\text{opt},w}$, signifying that the basin containing the RSM optimum has been found. At this point, a more accurate description of the system behavior near the optimum can be attained using more localized response surfaces. Whenever iterates are close to the boundaries, lower-D response surfaces are created by projecting the model onto constraints so as to prevent model generation based on an asymmetrical arrangement of the feasible sampling data.[2] A flowchart of the RSM algorithm is presented in Figure 10.

The RSM-optimal solution vector is denoted as S^R and has corresponding objective value $F(S^R)$. Once its value has been attained, the remaining locally optimal kriging solutions are also optimized using RSM.

Examples

In this section, the performance of the centroid-based sampling algorithm, as implemented within the kriging-RSM optimization method (KC-RSM), is evaluated based on its application to two global optimization test functions and two industrial case studies. In the first case study, the objective is to determine an optimal set of reaction conditions, given in terms of two species concentrations. This problem is initially solved with the only constraints being simple lower and upper bounds for each of the two reaction species. The problem is then solved when additional constraints are present to demonstrate the applicability of the centroid-based sampling strategy to problems defined by nonbox-constrained feasible regions. In the second case study, the objective is to identify the maximum reaction rate possible for simultaneous diffusion-reaction occurring in a catalyst pellet. For coupled heat and mass transfer, multiple possible reaction rates exist, and therefore it is important that application of the kriging-RSM algorithm leads to identification of all reaction rates. The physical first-principles equations are recast in a discretized form using orthogonal collocation, and the optimization

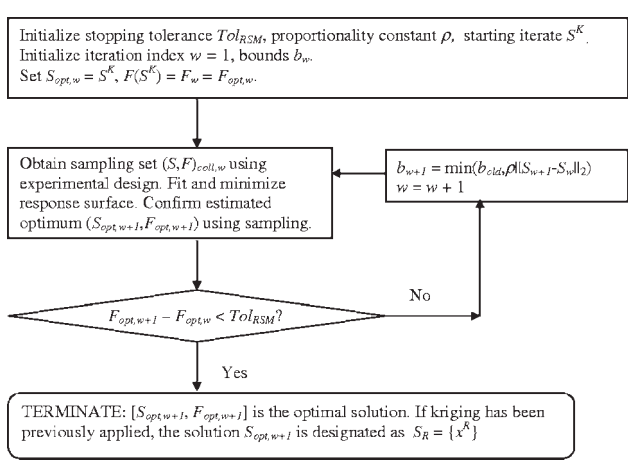

Figure 10. Flowchart of the RSM algorithm.

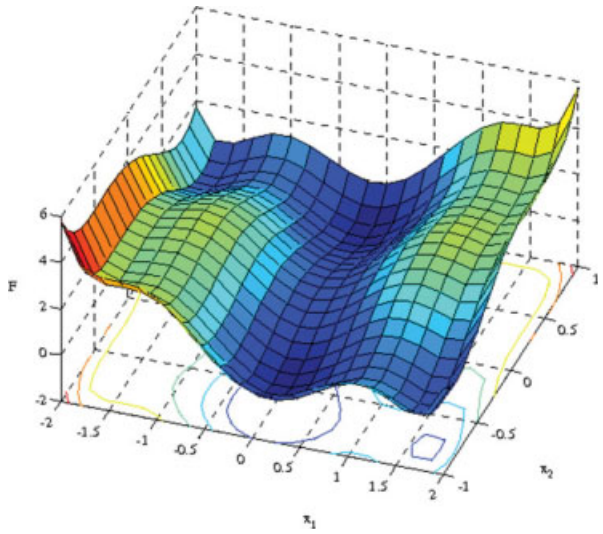

Figure 11. Plot of the objective function given in Problem (18).

[Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

problem involves a least-squares objective. For this case study, the kriging-RSM method is applied as an equation-solving algorithm.

The KC-RSM and KNC-RSM acronyms are used to refer to application of the kriging-RSM algorithm which does, and does not, use the centroid-based sampling strategy for kriging model updating, respectively. KNC refers to the kriging algorithm in which the initial global model is built from 10 dispersed sampling points, and in which subsequent models are updated based on data collected at regions of highest model variance, lowest prediction, and where the current and previous model predictions differ the most. Three sampling points are chosen for each of the three heuristic criteria, for a total of nine sampling points to use in model updating. For each example, a table of solution information is provided based on application of each of the KC-RSM and KNC-RSM algorithms. All results are obtained using an HP dv8000 CTO Notebook PC with a 1.8 GHz AMD Turion 64 processor.

Six-Hump Camel Back Function

The six-hump camel back function is a 2D box-constrained global optimization test function. Introducing black-box complications into this example, the output z_2 is a function of two continuous variables x_1 and x_2. The problem is formulated as shown below in problem (18) and a deterministic plot of $z_2(x_1, x_2)$ is presented in Figure 11:

$$\begin{aligned} &\min \quad z_2 \\ &\text{s.t.} \quad z_2 = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 - \left(-4 + 4x_2^2\right)x_2^2 \\ &\qquad -3 \le x_1 \le 3 \\ &\qquad -2 \le x_2 \le 2 \end{aligned} \tag{18}$$

The NLP given by problem (18) implies that only the single global optimum is to be found. However, this problem contains four local optima in addition to two global optima, and the goal of applying the KC-RSM and KNC-RSM algorithms is to identify the complete set of optima instead of simply the global one. Solution information obtained from application of both methods is presented in Tables 2 and 3, respectively.

From the information presented in Table 2, all six optima are identified when the centroid-based sampling strategy is used to build the global models. In contrast, all optima are obtained in only 16% of the cases when randomized/heuristic sampling is used for global model building/refinement. In addition, for the KC-RSM method, 13% fewer function calls are required relative to the 256 required by the KNC-RSM algorithm. The final kriging model, obtained after four iterations of the KC-RSM algorithm, is shown in Figure 12.

Schwefel Function

The 2D Schwefel test function is also a global optimization test problem. This box-constrained problem has been modified to include both a linear and a nonlinear constraint as shown in problem (19). The black-box variable z_2 is a sinusoidal function of x_1 and x_2 and therefore its nonconvex deterministic behavior resembles that of a noisy function. The problem is formulated as shown below and a 3D plot is shown in Figure 13. Because of the function nonconvexity induced by the additive sinusoidal terms, a contour plot is also provided in Figure 14.

$$\begin{aligned} &\min \quad z_2 \\ &\text{s.t.} \quad z_2 = \sum_{i=1}^{2} -x_i \sin\sqrt{|x_i|} \\ &\qquad -2x_1 - x_2 \le 800 \\ &\qquad 0.004x_1^2 - x_1 + x_2 \le 500 \\ &\qquad -500 \le x_1, x_2 \le 500 \end{aligned} \tag{19}$$

Table 2. Optimization Results Obtained for Problem (18) Based on Application of the KC-RSM Algorithm

No. of Iterations		No. of Function Calls			Optima Found			CPU Time		
KC	R	KC	R	Total = KC + $\sum(R)$	x_1	x_2	F	KC	R	Total
4	3	57	27	223	−0.0899	0.71098	−1.03161	1.609	1.047	2.656
N/A	3	0	27		0.0899	−0.71098	−1.03161	N/A		
	5	0	26		−1.7011	0.79460	−0.21538			
	5	0	26		1.7011	−0.79460	−0.21538			
	2	0	18		−1.6060	−0.56716	2.10427			
	2	0	18		1.6060	0.56716	2.10427			

Table 3. Optimization Results Obtained for Problem (18) Based on Application of the KNC-RSM Algorithm

No. of Iterations		No. of Function Calls			No. of Optima Found (N_{opt})	No. of Cases in Which Only N_{Opt} Optima Are Found/Total No. of Cases	CPU Time		
KNC	R	KNC	R	Total = KNC + $\sum(R)$			KNC	R	Total
4	13	30	59	89	2	2/100	1.211	0.352	1.563
5	15	43	89	132	3	11/100	2.003	0.911	2.913
6	20	48	121	168	4	38/100	2.090	1.097	3.187
9	21	71	135	206	5	33/100	3.158	1.317	4.474
12	24	103	153	256	6	16/100	4.885	1.450	6.335

This function contains a number of local optima and one global optimum located at (420.97, −302.525) which has an objective value of −719.53. The global optimum is located in a corner of the feasible region, surrounded by local optima. Because of the number of inferior local optima, the KC-RSM and KNC-RSM algorithms have been applied to find the five dispersed optima having objective values lower than −500. This problem was selected to demonstrate the application of the centroid-based sampling technique when a nonlinear constraint exists. For the initial modeling, the sampling set is obtained at locations corresponding to the vertices of the convex polytope inscribed within the nonlinear feasible region. For this problem, even if a nominal noise term were additively applied to z_2, the underlying geometry would still pose the main complication affecting successful convergence to the global optimum. Optimization results based on application of the KC-RSM and KNC-RSM algorithms are presented in Tables 4 and 5.

When the KNC-RSM method is applied, the five optima are found in 86 of 100 trials. The total number of samples required for kriging modeling is 469, just over twice the 230 required by the KC-RSM method. Because of the sinusoidal function behavior, no function "settling" is observed as the model is refined. The objective value varies over a wide range, between ±800, and many shallow/deep local optima are distributed over both small and large subregions. As the objective function value can vary over a wide range, when additional sampling is performed at locations where, (1) the model predictions are smallest, (2) the model variance is highest, and (3) the model has changed most between iterations, the average model value for the kriging predictor can change significantly between iterations. For the KNC-RSM algorithm, 32 iteratively refined global models are required before local optimization is initiated. As so many global models are created, this suggests that the termination criterion of requiring model convergence may be effective for successful identification of coarse local optima, but ineffective with respect to minimizing the number of samples needed for accurate global modeling.

Kinetics Case Study

For this example, the KC-RSM and KNC-RSM methods are applied to a kinetics case study.[26] There are five reaction species, two of which are system inputs. The reactions occur in an ideal CSTR and the reaction network considered is a modification of the Fuguitt and Hawkins mechanism given by: $A \rightarrow E$, $A \rightarrow D$, $B \leftrightarrow D$, $C \leftrightarrow 2D$, and $2A \rightarrow C$. Only A and C enter the reactor, and their respective concentrations C_A^0, C_C^0 are given in units of mol/m^3. The dynamic behavior of the system is described by problem (20).

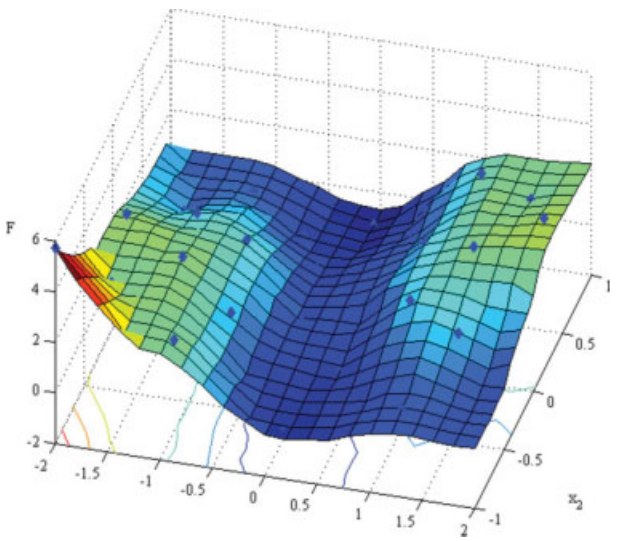

Figure 12. Refined kriging model of the objective function given in Problem (18), based on the application of the KC-RSM algorithm.

[Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

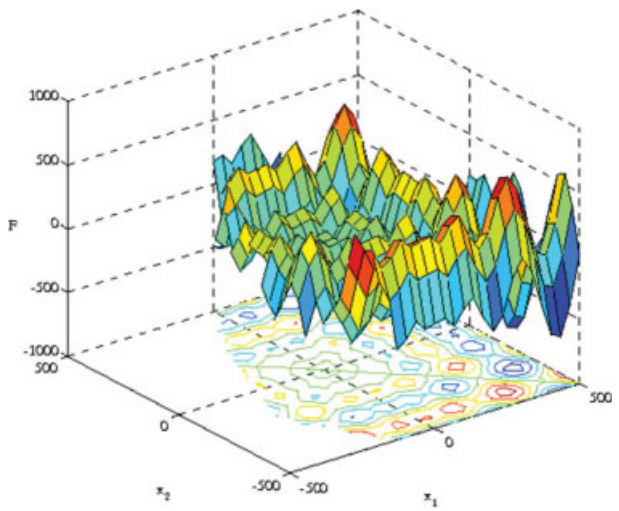

Figure 13. 3D plot of the deterministic objective function given in Problem (19).

[Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

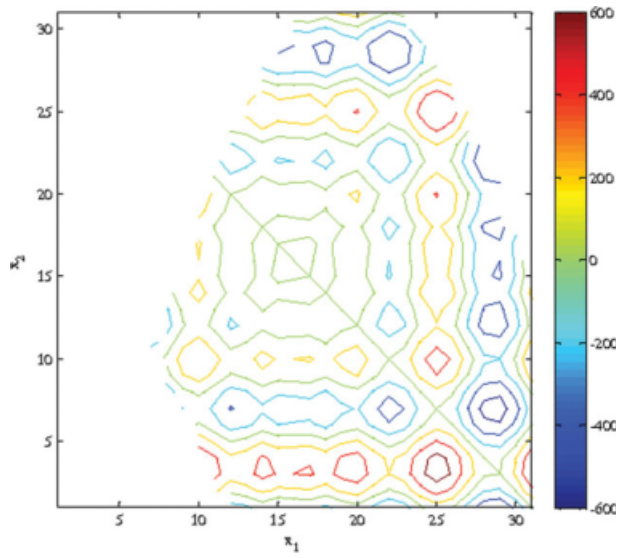

Figure 14. Contour plot of the deterministic objective function given in Problem (19).

[Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

$$\begin{aligned}
&\min F = 4(X-0.6)^2+4(Y-0.4)^2+\sin^3(\pi X)+0.4\\
&X = 0.1428C_C^{SS}-0.357C_D^{SS}\\
&Y = -0.1428C_C^{SS}+2.857C_D^{SS}+1.0\\
&\frac{dC_A}{dt}=\frac{F_R}{V}\left(C_A^0-C_A\right)-k_1^fC_A-k_2^fC_A-k_5^fC_A^2\\
&\frac{dC_B}{dt}=\frac{F_R}{V}(-C_B)-k_3^fC_B+k_3^rC_D\\
&\frac{dC_C}{dt}=\frac{F_R}{V}\left(C_C^0-C_C\right)-k_4^fC_C+0.5k_4^rC_D^2+0.5k_5^fC_A^2\\
&\frac{dC_D}{dt}=\frac{F_R}{V}(-C_D)+k_2^fC_A+k_3^fC_B-k_3^rC_D+2k_4^rC_C\\
&\qquad -k_4^rC_D^2\\
&\frac{dC_E}{dt}=\frac{F_R}{V}(-C_E)+k_1^fC_A\\
&\left[k_1^f,k_2^f,k_3^f,k_3^r\right]=[3.33384,0.26687,0.29425,0.14940]\\
&\left[k_4^f,k_4^r,k_5^f,F_R,V\right]=\left[0.011932,1.8957\mathrm{e}^{-3},9.9598\mathrm{e}^{-6},0.008,0.1\right]\\
&3\mathrm{e}^3\le C_A^0\le 3\mathrm{e}^4\\
&0\le C_C^0\le 10\,\mathrm{e}^4
\end{aligned}\tag{20}$$

The variables C_C^{SS} and C_D^{SS} represent the steady-state values of C_C and C_D, respectively. The rate constants, input flow rate, and reactor volume are similarly given by k, F_R, and V. A plot of the objective F as a function of ${C_A}^0$ and ${C_C}^0$ is shown in Figure 15. In the following two subsections, this problem is solved as a box-constrained and then nonbox-constrained NLP.

Box-Constrained Problem with Noise

The deterministic solution of problem (20) yields a global optimum of $F = 0.7422$ at $[{C_A}^0, {C_C}^0] = [10.117, 8.378]$ and a local optimum of $F = 1.2364$ at [13.202, 3.163]. Black-box complications are introduced by assuming that the rate equations are unknown. In the absence of the rate equations, the system is simulated using a microscopic model represented by a lattice containing N molecules. The microscopic model is generated by first translating concentrations to molecular variables, evolving the microscopic system using the Gillespie algorithm,[27] and then mapping the final number of particles for each species back to concentrations. The noise in the output concentrations arises as a function of the microscopic model coarseness, given as N. Steady-state solution vectors are obtained from an initial point by running the microscale simulations for a long time horizon, after which the objective function is evaluated. The optimization problem is formulated in problem (21), in which the steady-state concentrations ${C_C}^{SS}$ and ${C_D}^{SS}$ are treated as the output variables for this black-box system:

$$\begin{aligned}
\min\ \ & F = 4(X-0.6)^2+4(Y-0.4)^2+\sin^3(\pi X)+0.4\\
& X = 0.1428C_C^{SS}-0.357C_D^{SS}\\
& Y = -0.1428C_C^{SS}+2.857C_D^{SS}+1.0\\
&\left[z_{2,1},z_{2,2}\right]=\left[C_C^{SS},C_D^{SS},\right]=\Gamma\begin{pmatrix}C_A^0,C_C^0,k_1^f,k_2^f,k_3^f,k_3^r,\\ k_4^f,k_4^r,k_5^f,F_R,V,N\end{pmatrix}\\
&3\mathrm{e}^3\le C_A^0\le 3\mathrm{e}^4\\
&0\le C_C^0\le 10\,\mathrm{e}^4
\end{aligned}\tag{21}$$

The variance of the objective is evaluated over k replicate simulations at a nominal species concentration vector $x = [{C_A}^0, {C_C}^0]$ as:

$$\sigma^2=\mathrm{Var}(\varepsilon)=\mathrm{Var}\{F_i(x,N)|i=1\ldots j\}\tag{22}$$

To solve problem (22) both deterministically and stochastically, noise is added to F in the form of a normally

Table 4. Optimization Results Obtained for Problem (19) Based on Application of the KC-RSM Algorithm

No. of Iterations		No. of Function Calls			Optima Found			CPU Time (s)		
KC	R	KC	R	Total = KC + $\sum(R)$	x_1	x_2	F	KC	R	Total
4	2	73	31	230	420.775	−302.028	−719.491	3.391	3.766	7.156
N/A	1	0	30		203.783	420.958	−620.826	N/A		
	11	0	20		416.871	202.563	−618.513			
	3	0	40		420.734	−123.123	−541.478			
	7	0	36		202.105	−301.280	−501.820			

Table 5. Optimization Results Obtained for Problem (19) Based on Application of the KNC-RSM Algorithm

No. of Iterations		No. of Function Calls			No. of Optima Found (N_{Opt})	No. of Cases in which N_{Opt} Optima Are Found/Total No. of Cases	CPU Time (s)		
KNC	R	KNC	R	Total = KNC + $\sum(R)$			KNC	R	Total
32	29	283	186	469	5	86/100	26.337	4.638	30.974

distributed error having standard deviations $\sigma = 0$ and $\sigma = 0.011$. The corresponding microscale system sizes are $N =$ 1e6 and $N = 100{,}000$, respectively. Problem (21) is now restated as an optimization problem containing a noisy objective, given by problem (23):

$$\begin{aligned}
&\min F = 4(X-0.6)^2+4(Y-0.4)^2+\sin^3(\pi X)+0.4+N(0,\sigma^2)\\
&\quad X = 0.1428C_C^{SS} - 0.357C_D^{SS}\\
&\quad Y = -0.1428C_C^{SS} + 2.857C_D^{SS} + 1.0\\
&\left[z_{2,1}, z_{2,2}\right] = \left[C_C^{SS}, C_D^{SS},\right] = \Gamma\begin{pmatrix} C_A^0, C_C^0, k_1^f, k_2^f, k_3^f, k_3^r, \\ k_4^f, k_4^r, k_5^f, F_R, V \end{pmatrix}\\
&\quad 3e^3 \le C_A^0 \le 3e^4\\
&\quad 0 \le C_C^0 \le 10\,e^4
\end{aligned} \tag{23}$$

For the local optimization using RSM, the value of Tol_{RSM} is 0.01, meaning that the difference between the current and previous optimum objective values, as given by $|F_{opt,w+1} - F_{opt,w}|$, must fall below 0.01 before the RSM algorithm is terminated. Once this termination criteria is met, another response surface is built and optimized using a design radius b_w whose value is half that of the previous one, to further improve the optimum. Tables 6 and 7 present results when problem(23) is solved deterministically while Tables 8 and 9 present complementary results when problem (23) is solved stochastically. For all tables, the reported CPU times refer to the computational expense required for modeling and optimization, and exclude the time required for each function call in the form of a microscopic model simulation.

When problem (23) is solved deterministically, the number of function calls required to obtain both optima is identical when either one of the KC-RSM or KNC-RSM methods is used. However, even though the global optimum is found in all cases when the KNC-RSM method is used, both optima are identified in only 53% of the cases.

When problem (23) is solved stochastically, at least one of the optima are obtained in (17 + 25 + 39) = 81% or (9 + 48 + 22) = 79% of the cases when the KC-RSM and KNC-RSM methods are applied, respectively. The global optimum alone is found in (25 + 39) = 64 cases when the KC-RSM algorithm is applied, and in (48 + 22) = 70 cases when the KNC-RSM method is used instead. However, both optima are identified in 39 cases when the KC-RSM method is used, nearly double the number of cases (22) when using the KNC-RSM method.

Nonbox-Constrained Problem

To demonstrate the applicability of centroid-based sampling to a nonbox-constrained feasible region, problem (21) has been modified to include three additional linear constraints as shown in problem (24). A plot of the function is shown in Figure 16.

$$\begin{aligned}
&\min F = 4(X-0.6)^2+4(Y-0.4)^2+\sin^3(\pi X)+0.4\\
&\quad X = 0.1428C_C^{SS} - 0.357C_D^{SS}\\
&\quad Y = -0.1428C_C^{SS} + 2.857C_D^{SS} + 1.0\\
&z_2 = \left[C_C^{SS}, C_D^{SS}\right] = \Gamma\begin{pmatrix} C_A^0, C_C^0, k_1^f, k_2^f, k_3^f, k_3^r, \\ k_4^f, k_4^r, k_5^f, F_R, V, N \end{pmatrix}\\
&\quad C_A^0 - 11.5681C_C^0 \le -69.4928\\
&\quad C_A^0 + 1.6615C_C^0 \le 13.8\\
&\quad C_A^0 + 1.8C_C^0 \le 33.6\\
&\quad 3e^3 \le C_A^0 \le 3e^4\\
&\quad 0 \le C_C^0 \le 10\,e^4
\end{aligned} \tag{24}$$

The additional constraints alter the original box-shaped region to that of a pentagon. The nominal sampling set obtained using the KC-RSM algorithm is now comprised of (1) the five vertices of the reduced feasible region, rather than the four used for the box-constrained region, and (2) the centroid location determined with respect to these five vertices.

The solution set of problem (24) consists of (1) a constrained global optimum at $[C_A^0, C_C^0] = [12.6140, 7.0977]$

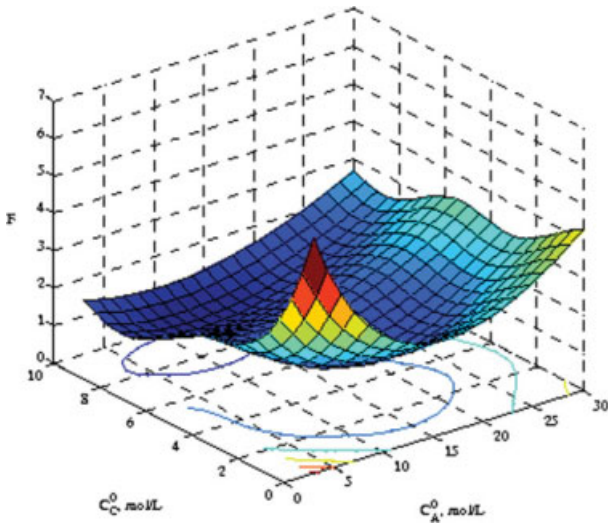

Figure 15. Plot of the objective function given in Problem (20).

[Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

Table 6. Optimization Results Obtained for Problem (23) Under the Condition that $\sigma = 0$, Based on Application of the KC-RSM Algorithm

No. of Iterations		No. of Function Calls			Optima Found			CPU Time (s)		
KC	R	KC	R	Total = KC + $\sum(R)$	x_1	x_2	F	KC	R	Total
4	3	57	26	101	10.1178	8.3787	0.74221	1.5313	0.2188	1.75
N/A	2	0	18		13.2025	3.1756	1.23644	N/A		

Table 7. Optimization Results Obtained for Problem (23) Under the Condition that $\sigma = 0$, Based on Application of the KNC-RSM Algorithm

No. of Iterations		No. of Function Calls			No. of Optima Found (N_{Opt})	No. of Cases in which only N_{Opt} Optima Are Found/Total No. of Cases	CPU Times (s)		
KNC	R	KNC	R	Total = KNC + $\sum(R)$			KNC	R	Total
5	3	42	21	63	1 (global only)	47/100	1.899	0.168	2.067
6	7	44	57	101	2 (global + local)	53/100	1.921	0.495	2.416

whose corresponding objective value is 1.0424, and (2) a local optimum at $[C_A{}^0, C_C{}^0] = [13.2025, 3.1756]$, whose corresponding objective value is 1.2364. The optimization results obtained are presented in Tables 10 and 11. The KC-RSM algorithm is terminated after two iterations, or after the global model has been updated only once. Because of the smaller feasible region size, early termination is applied to avoid obtaining sampling information at smaller Delaunay triangle centroids that might fail to provide any new function behavior not already captured by the previous sampling set.

Both optima are found using the KC-RSM method in approximately the same number of average function calls required for just the global optimum to be identified by the KNC-RSM method. The discovery of both optima using the KNC-RSM method requires a 27% increase in the number of function calls relative to the 76 required by the KC-RSM method. Furthermore, both optima are identified with only a 64% success rate using the KNC-RSM algorithm.

Diffusion and Reaction in a Catalyst Pellet

For this case study, taken from Lucia and Gattupalli,[1] the kriging-RSM algorithms are applied as equation-solving methods. The goal of this study is to identify the best modeled reaction rate of a chemical species undergoing simultaneous diffusion-reaction through a spherical catalyst pellet as shown in Figure 17. The physical behavior of this coupled heat-mass transfer problem is described by a PDE having specified boundary conditions. Although NLP solvers may fail to yield closed-form solutions of this differential equation system, they may alternatively succeed at identifying the solutions to a discretized complement of the original problem. The physical system models and discretization methods described below are the same as given by Lucia and Gattupalli.[1] The KC-RSM and KNC-RSM algorithms are applied as equation solvers once the optimization problem has been reformulated in terms of a minimum set of nonlinear equations.

The dimensionless nonisothermal species and heat transport model is given by Eq. 25.

$$\frac{d^2y}{dx^2} + \frac{2}{x}\frac{dy}{dx} - \phi^2 y \exp\left[\frac{\gamma\beta(1-y)}{1+\beta(1-y)}\right] = 0 \qquad (25)$$

The boundary conditions specify the following behavior: (1) at the pellet surface, the dimensionless species concentration

Table 8. Optimization Results Obtained for Problem (23) Under the Condition that $\sigma = 0.011$, Based on Application of the KC-RSM Algorithm

No. of Iterations		No. of Function Calls			No. of Optima Found (N_{Opt})	No. of Cases in which only N_{Opt} Optima Are Found/Total No. of Cases	CPU Time (s)		
KC	R	KC	R	Total = KC + $\sum(R)$			KC	R	Total
4	7	57	59	116	1 (local only)	17/100	1.783	0.563	2.347
4	9	57	75	132	1 (global only)	25/100	1.901	0.858	2.758
4	10	57	79	136	2 (global + local)	39/100	1.869	0.765	2.634

Table 9. Optimization Results Obtained for Problem (23) Under the Condition that $\sigma = 0.011$, Based on the Application of the KNC-RSM Algorithm

No. of Iterations		No. of Function Calls			No. of Optima Found (N_{Opt})	No. of Cases in which only N_{Opt} Optima Are Found/Total No. of Cases	CPU Time (s)		
KNC	R	KNC	R	Total = KNC + $\sum(R)$			KNC	R	Total
6	9	48	65	113	1 (local only)	9/100	2.450	0.721	3.170
6	5	45	39	84	1 (global only)	48/100	2.169	0.394	2.563
5	11	41	78	119	2 (global + local)	22/100	1.874	0.818	2.691

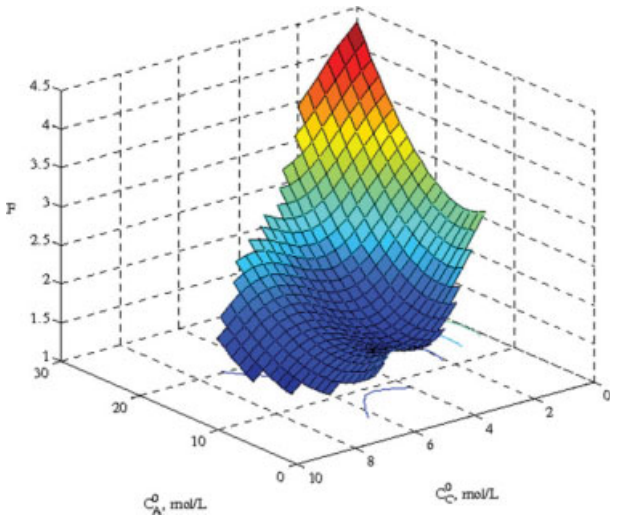

Figure 16. Plot of the objective function given in Problem (24).

[Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

is unity as shown in Eq. 26, and (2) at the pellet center, the species concentration change is zero, as given in Eq. 27. The reaction rate η is given by Eq. 28.

$$y\Big|_{x=1} = 1 \tag{26}$$

$$\frac{dy}{dx}\Big|_{x=0} = 0 \tag{27}$$

$$\eta = \frac{3}{\phi^2}\frac{dy}{dx}\Big|_{x=1} \tag{28}$$

The complete optimization problem is formulated in problem (29):

$$\begin{aligned} \min \quad & F \\ \text{s.t.} \quad & F = \frac{d^2y}{dx^2} + \frac{2}{x}\frac{dy}{dx} - \phi^2 y \exp\left[\frac{\gamma\beta(1-y)}{1+\beta(1-y)}\right] \\ & y\Big|_{x=1} = 1 \\ & \frac{dy}{dx}\Big|_{x=0} = 0 \\ & 0 \le y \le 1 \\ & [\beta, \phi, \gamma] = [0.6, 0.2, 30] \end{aligned} \tag{29}$$

To identify all possible reaction rates η, orthogonal collocation over finite elements (OCFE) is used to discretize the problem given by Eqs. 25–28. From the discretized equations, variable elimination can reduce the problem to a set of nonlinear equations given in terms of dimensionless concentrations $y(x_i)$ for i equispaced intervals between [0, 1]. Once a vector $y(x_i)$ satisfying the nonlinear equations has been obtained, the appropriate elements of this vector are substituted into the OCFE derivative expression at $x = 1$ to generate the value for η. This process is repeated for all solution vectors y until all values of η have been determined.

The dimensionless radius x, which varies between zero at the pellet center and unity at the surface, is subdivided into N_E equidistant elements, each one containing N_N nodes. For each element, a unique approximation is generated for the dimensionless species concentration y as a function of radial distance x from the pellet center. C^0 continuity is enforced at adjacent finite element junctions whereby the piecewise approximations corresponding to two adjacent finite elements must agree at the common node location x. In addition, C^1 continuity is similarly enforced by requiring that the piecewise approximation gradients also agree. The orthogonal collocation equations are given as follows:

$$y^{[i]} = \sum_{j=1}^{N_N} L_{(i-1)(N_E-1)+j} y_{(i-1)(N_E-1)+j} \quad i = 1 \ldots N_E \tag{30}$$

$$L_{(i-1)(N_E-1)+j} = \prod_{\substack{\{j,k\}=N_N(i-1)+1 \\ j \ne k}}^{N_{Nl}} \frac{\chi - \chi_k}{\chi_j - \chi_k} \qquad i = 1 \ldots N_E \tag{31}$$

$$\begin{aligned} &\frac{d^2y[l]}{dx^2} + \frac{2}{x}\frac{dy[l]}{dx} \\ &\quad - \phi^2 y^{[i]} \exp\left(\frac{\gamma\beta\left(1-y^{[i]}\right)}{1+\beta\left(1-y^{[i]}\right)}\right)\Bigg|_{x=(i-1)/N_E+j/[(N_N-1)N_E]} = 0 \\ &\qquad i = 1 \ldots N_\varepsilon - 1,\ j = 1 \ldots N_N - 2 \end{aligned} \tag{32}$$

$$y^{[N_E]}\Big|_{x=1} = 1 \tag{33}$$

$$\frac{dy^{[1]}}{dx}\Big|_{x=0} = 0 \tag{34}$$

$$y^{[i]}\big|_{x=i/N_E} = y^{[i+1]}\Big|_{x=i/N_E} \qquad i = 1 \ldots N_E - 1 \tag{35}$$

$$\frac{dy^{[i]}}{dx}\Big|_{x=i/N_E} = \frac{dy^{[i+1]}}{dx}\Big|_{x=i/N_E} \qquad i = 1 \ldots N_E - 1 \tag{36}$$

Equation 30 provides the piecewise approximation for y over a given element i, defined as $y[i]$ and valid over the

Table 10. Optimization Results Obtained for Problem (24) Based on Application of the KC-RSM Algorithm

No. of Iterations		No. of Function Calls			Optima Found			CPU Time (s)		
KC	R	KC	R	Total = KC + $\sum(R)$	$C_A{}^0$	$C_c{}^0$	F	KC	R	Total
2	3	11	24	76	12.7946	7.1133	1.0424	0.7500	0.3438	1.0938
N/A	3	0	38		13.203	3.1944	1.2366	N/A		

Table 11. Optimization Results Obtained for Problem (24) Based on Application of the KNC-RSM Algorithm

No. of Iterations		No. of Function Calls			No. of Optima Found (N_{Opt})	No. of Cases in which N_{Opt} Optima Are Found/Total No. of Cases	CPU Time (s)		
KNC	R	KNC	R	Total = KNC + $\sum(R)$			KNC	R	Total
4	8	31	46	77	1 (global only)	36/100	1.210	0.369	1.579
4	11	32	65	97	2 (global + local)	64/100	1.148	0.533	1.681

interval $[(i-1)/N_E, i/N_E]$, $i = 1 \ldots N_E$. For each approximation, $y^{[i]}$ is expressed as the sum of N_N nodal approximations $y_{(i-1)}(NE-1)_{+j}$, each weighted by a corresponding Lagrange polynomial $L_{(i-1)(NE-1)+j}$, for $i = 1 \ldots N_E$ and $j = 1 \ldots N_N$, as given in Eq. 31. Equation 32 is the diffusion-reaction transport equation that must be satisfied at all interior nodes for each one of the i finite element approximations $y[i]$. Equations 33 and 34 correspond to the boundary conditions specified by (26) and (27). Equations 35 and 36 are the C^0 and C^1 continuity equations which state that the ith piecewise approximation $y^{[i]}$ and its gradient $dy^{[i]}/dx$ must match the corresponding $i^{th} + 1$ approximations $y^{[i+1]}$ and $dy^{[i+1]}/dx$ at common interior node locations x, respectively. In the following subsections, two discretization cases are considered: (1) two elements and three nodes, and (2) five elements and three nodes.

Discretization based on two elements and three nodes

For a discretization containing two elements and three nodes, $N_E = 2$ and $N_N = 3$. Sample values for β, γ, and ϕ are arbitrarily given values of 0.6, 30, and 0.2, respectively. Six equations are obtained as the two-element, three-node OCFE approximation, given by Eqs. S-23–S-28 in the Supporting Information. After variable elimination, the problem can be reduced to the least-squares NLP given by problem (37):

$$\begin{aligned} \min\ & F = \sum_{n=1}^{2} z_{1,n}^2 \\ \text{s.t. } & z_{1,1} = -32y_2 + 32y_3 - \phi^2 y_2 \exp\left(\frac{\gamma\beta(1-y_2)}{1+\beta(1-y_2)}\right) \\ & z_{1,2} = 10.\overline{6}y_3 - 32y_5 + 21.\overline{3} - \phi^2 y_5 \exp\left(\frac{\gamma\beta(1-y_5)}{1+\beta(1-y_5)}\right) \\ & 0 \le [y_2, y_5] \le 1 \\ & [\beta, \phi, \gamma] = [0.6, 0.2, 30] \end{aligned} \tag{37}$$

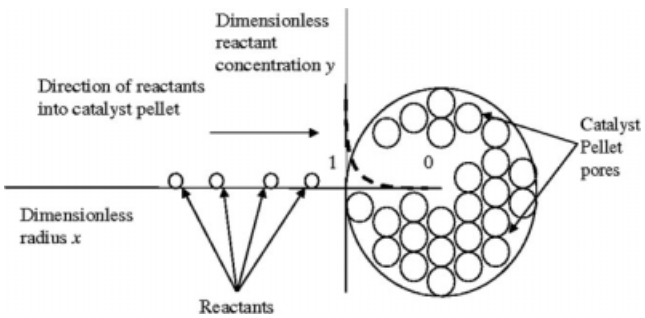

Figure 17. Dimensionless concentration profile of reactants undergoing simultaneous diffusion and reaction as they move through the pores of a spherical catalyst pellet.

The variables y_2 and y_5 represent the dimensionless concentrations at corresponding locations $x_2 = 0.25$ and $x_5 = 0.75$, and the range of possible values is between [0, 1]. A solution to this problem is one in which the sum of the squared constraint violations $z_{1,1}$ and $z_{1,2}$, uniquely specified for a vector $\{y_2, y_5\}$ $[0, 1]^2$, falls below a given tolerance. A plot of the objective function given in terms of y_2 and y_5 is shown in Figure 18. This problem's solution consists of two minima located at $[y_2, y_5] = [0.66594, 0.92606]$ and $[0.99313, 0.99685]$. For both minima, the objective value is below a tolerance of 1e-7. The vector [0.66594, 0.92606] is a global minimum with respect to the optimal reaction rate since the species concentration profile y is lower relative to the one given by $[y_2, y_5] = [0.99313, 0.99685]$.

CONOPT fails to identify the two minima unless the initial bounds for y_2 and y_5 are set away from [0, 0]. As a feasible path solver, CONOPT requires accurate derivative information from the model equations to identify an optimum. By default, CONOPT's algorithm first initializes all independent variables at their lower bounds. If this vector is infeasible, or the initial derivative information is inaccurate, the search directions found may not match the path to an optimum, and premature termination can occur.

Because y_2 and y_5 can have values between [0, 1], the initial values for y_2 and y_5 are therefore set at zero. In addition to $[y_2, y_5] = [0, 0]$ being an infeasible point, the derivatives

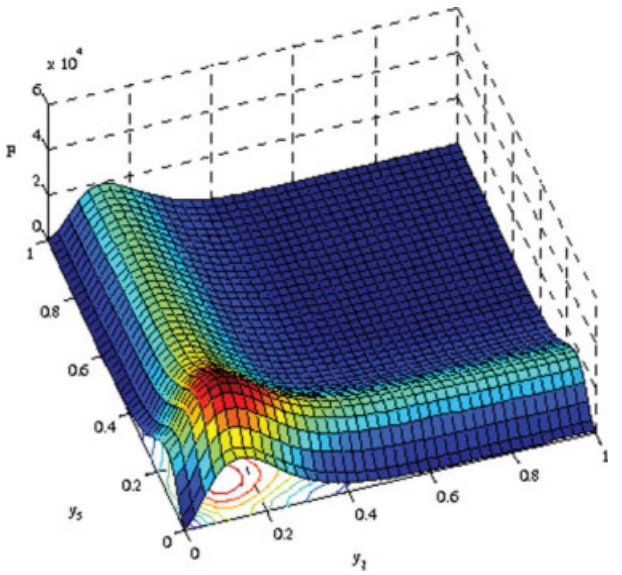

Figure 18. Plot of the dimensionless concentration y for a 2-element, 3-node OCFE approximation of the spherical catalyst pellet model given by Eqs. 26–28.

[Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

Table 12. Optimization Results Obtained for Problem (37) Based on Application of the KC-RSM Algorithm

No. of Iterations		No. of Function Calls			Optima Found			CPU Time		
KC	R	KC	R	Total = KC + $\sum(R)$	x_1	x_2	F	KC	R	Total
4	5	57	34	237	0.99310	0.99684	5.65e-7	3.7188	1.9063	5.625
N/A	10	0	72		0.66600	0.92606	1.91e-6	N/A		

containing y_2 and y_5 terms will also be zero. Since these terms appear as multilinear terms in the objective due to the squared $z_{1,1}$ and $z_{1,2}$ equations, the objective's derivative will be zero. The algorithm will then treat the objective as independent of y_2 and y_5.

In contrast, when the lower bounds of y_2 and y_5 are set at [0.1, 0.1], the solution [0.99313, 0.99685] is found. The ridge system existing along the $y_2 = 0.2$ and $y_5 = 0.2$ axes isolates the two optima. When the lower bounds for [y_2, y_5] are set at [0.1, 0.1], the algorithm converges to the [0.99313, 0.99685] optimum, even though the Euclidean distance between [0.1, 0.1] and the other solution, [0.66594, 0.92606], is shorter, because [0.1, 0.1] resides on the same side of the ridge as the [0.99313, 0.99685] solution. Conversely, when the lower bounds of [y_2, y_5] are instead given as any other point on the other side of the ridge system, such as [0.4, 0.4], the [0.66594, 0.92606] solution is found instead.

The kriging-RSM method is therefore applied as an alternative technique to the terrain method for identifying both optima. Tables 12 and 13 present the results obtained when the KC-RSM and KNC-RSM algorithms are used to solve problem (37).

The number of sampling experiments required to obtain both the local and global optimum using the KC-RSM algorithm is 237, 80% of the number required by KNC-RSM. Furthermore, using the KNC-RSM method, both optima are identified at a 69% success rate. The discovery of at least the local optimum [0.99313, 0.99685] is identified in 90% of the cases and is easy to find since it is located close to a feasible point vertex at [1, 1]. This sampling point is included in the nominal sampling set for the centroid-based method, and is a likely sampling point during the early stages of the KNC-RSM method since feasible vectors having high model uncertainty are often identified at points along the feasible region boundaries.

Discretization based on five elements and three nodes

The optimization problem corresponding to problem (29), for a discretization consisting of five elements and three nodes by the OCFE method, is given as follows. The complete set of OCFE equations is given in the Supporting Information.

$$\begin{aligned}
&\min F = \sum_{n=1}^{5} z_{1,n}^2 \\
&\text{s.t. } z_{1,1} = -200y_2 + 200y_3 - \phi^2 y_2 \exp\left(\frac{\gamma\beta(1-y_2)}{1+\beta(1-y_2)}\right) \\
&\quad z_{1,2} = 66.\overline{6}y_4 - 200y_5 + 133.\overline{3}y_6 - \phi^2 y_5 \exp\left(\frac{\gamma\beta(1-y_5)}{1+\beta(1-y_5)}\right) \\
&\quad z_{1,3} = 80y_7 - 200y_8 + 120y_9 - \phi^2 y_8 \exp\left(\frac{\gamma\beta(1-y_8)}{1+\beta(1-y_8)}\right) \\
&\quad z_{1,4} = 85.\overline{714285}y_{10} - 200y_{11} + 114.\overline{285714}y_{12} \\
&\qquad - \phi^2 y_{11} \exp\left(\frac{\gamma\beta(1-y_{11})}{1+\beta(1-y_{11})}\right) \\
&\quad z_{1,5} = 88.\overline{8}y_{13} - 200y_{14} + 111.\overline{1} \\
&\qquad - \phi^2 y_{14} \exp\left(\frac{\gamma\beta(1-y_{14})}{1+\beta(1-y_{14})}\right) \\
&\quad 0 \le [y_2, y_5, y_8, y_{11}, y_{14}] \le 1 \\
&[\beta, \phi, \gamma] = [0.6, 0.2, 30]
\end{aligned} \tag{38}$$

In Tables 14–16, results are presented based on the application of the KC-RSM and KNC-RSM algorithms to solve problem (38). Table 15 contains the optimal vectors.

Table 13. Optimization Results Obtained for Problem (37) Based on Application of the KNC-RSM Algorithm

No. of Iterations		No. of Function Calls					CPU Time (s)		
KNC	R	KNC	R	Total = KNC + $\sum(R)$	No. of Optima Found (N_{Opt})	No. of Cases in which Only N_{Opt} Optima Are Found/Total No. of Cases	KNC	R	Total
9	25	73	143	215	1 (local only)	21/100	8.461	1.653	10.114
7	25	56	155	212	1 (global only)	8/100	6.676	1.318	7.994
10	33	80	216	296	2 (global + local)	69/100	9.583	2.179	11.762

Table 14. Optimization Results Obtained for Problem (38) Based on Application of the KC-RSM Algorithm

No. of Iterations		No. of Function Calls			CPU Time (s)		
KC	R	KC	R	Total = KC + $\sum(R)$	KC	R	Total
4	5	280	876	9,686	52.3	36.5	88.8
N/A	36	0	8,530		N/A		

Table 15. Optima Vectors Obtained for Problem (38) Based on Application of the KC-RSM Algorithm

Optima Found					
y_2	y_5	y_8	y_{11}	y_{14}	F
0.99199	0.99263	0.99394	0.99594	0.99853	8.80e-5
0.45717	0.76123	0.89781	0.95473	0.98784	1.81e-7

Table 16. Optimization Results Obtained for Problem (38) Based on Application of the KNC-RSM Algorithm

No. of Iterations		No. of Function Calls			No. of Optima Found (N_{Opt})	No. of Cases in which only N_{Opt} Optima Are Found/Total No. of Cases	CPU Time (s)		
KNC	R	KNC	R	Total = KNC + $\sum(R)$			KNC	R	Total
6	12	52	1,918	1,970	1 (local only)	14/100	11.84	2.72	14.57
7	55	53	12,588	12,641	1 (global only)	35/100	12.41	88	100
11	67	96	15,029	15,124	2 (global + local)	40/100	22.13	144	166

The optima vectors differ from those found using a two-element, three-node OCFE discretization because the equations for y, dy/dx, and d^2y/dx^2 are different. The number of sampling experiments required to find both optima using the KC-RSM method is 9,686, 64% of the 15,124 required by the KNC-RSM algorithm. Furthermore, both optima are identified at a 40% success rate. The discovery of just the global optimum is identified in 75% of the cases. The computational time required for global modeling using the KC-RSM algorithm is now higher than that required for the KNC-RSM method because the computational time required for Delaunay triangulation increases rapidly when the number of simplex points, and, to a lesser extent, the number of sampling points used in the triangulation, both increase. The solution of this problem was also attempted using OCFE approximations for a five-element, four-node problem, whose reduced problem dimensionality was ten. However, the computational time required for performing the third-iteration Delaunay triangulation became prohibitive, as a new sampling set comprised of 11-point Delaunay triangle centroids failed to be obtained after 52,380 CPU s. It is emphasized that since the limiting factor in the application of the KC-RSM algorithm to a 10D problem is qHull's computational complexity, one extension of this work is the application of a Delaunay triangulation method using an alternative convex hull algorithm having lower computational complexity. The KC-RSM algorithm could then be applied as an equation-solver for problems of higher dimension. Furthermore, a comparison of the results obtained for the 10-D problem using KC-RSM could then be evaluated.

Conclusions and Future Work

The novel contribution of this work has been the application of a centroid-based sampling algorithm for iterative global modeling using kriging. The new sampling algorithm is motivated by the opportunity to reduce modeling sampling expense, which has practical implications in terms of the resource costs associated with performing field experiments or conducting computationally expensive simulations. The new sampling technique is applied within a kriging-RSM algorithm to obtain all solutions of two global optimization test problems and two case studies. The promise of the centroid-based sampling algorithm is measured in terms of the sampling expense required to identify all optima for each problem, and is compared against the corresponding sampling expense associated with the application of a sampling method using (1) random sampling for initial modeling, and (2) sampling at locations of minimum prediction, maximum uncertainty, and where there is high model change between the current and previous model. The application of the new sampling technique results in all optima being found at lower or equivalent sampling expense compared to the sampling algorithm relying on randomized/heuristic sampling, when the problem dimensionality is low. When the problem dimensionality is greater than five, the computational limitations associated with generating a Delaunay triangulation result in a prohibitive increase in the global modeling time. As a result, the centroid-based sampling algorithm has currently been applied for problems containing only a low number of variables. Each kriging model is refined using additional sampling information, and once the global model is accurate, the best solutions serve as starting iterates for further optimization using RSM. The additional costs resulting from global modeling are offset by convergence to improved local solutions. Future work is targeted at: (1) implementing a more efficient Delaunay triangulation algorithm so that the new sampling technique can be applied to problems of arbitrary dimension, and (2) applying the centroid-based sampling technique to problems in which integer variables are present both inside and outside the black-box models.

Acknowledgments

The authors gratefully acknowledge the financial support from the NSF CTS 0625515 grant and also the USEPA-funded Environmental Bioinformatics and Computational Toxicology Center under the GAD R 832721-010 grant.

Literature Cited

1. Lucia A, Gattupalli RR. A barrier-terrain methodology for global optimization. *Ind Eng Chem Res*. 2008;47:2666–2680.
2. Davis E, Ierapetritou M. A Kriging-based approach to MINLP containing black-box models and noise. *Ind Eng Chem Res*. 2008;47:6101–6125.
3. Davis E, Ierapetritou M. A kriging based method for the solution of mixed-integer nonlinear programs containing black-box functions. *J Global Optim*. 2009;43:191–205.

4. Davis E, Ierapetritou M. A kriging method for the solution of nonlinear programs with black-box functions. *AIChE J*. 2007;53:2001–2012.
5. Floudas CA, Akrotirianakis IG, Caratzoulas S, Meyer CA, Kallrath J. Global optimization in the 21st century: advances and challenges. *Comp Chem Eng*. 2005;29:1185–1202.
6. Meyer C, Floudas C, Neumaier A. Global optimization with nonfactorable constraints. *Ind Eng Chem Res*. 2002;41:6413–6424.
7. Barton RR, Ivey JS. Nelder-Mead simplex modifications for simulation optimization. *Manage Sci*. 1996;42:954–973.
8. Jones D, Perttunen C, Stuckman B. Lipschitzian optimization without the Lipschitz constant. *J Opt Theory Appl*. 1993;79:157–181.
9. Huyer W, Neumaier A. Global optimization by multilevel coordinate search. *J Global Optim*. 1999;14:331–355.
10. Storn R, Price K. Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. *J Global Optim*. 1997;11:341–359.
11. Gilmore P, Kelley CT. An implicit filtering algorithm for optimization of functions with many local minima. *SIAM J Optim*. 1995;5:269–285.
12. Choi TD, Kelley CT. Superlinear convergence and implicit filtering. *SIAM J Optim*. 2000;10:1149–1162.
13. Brekelmans R, Driessen L, Hamers H, Hertog D. Gradient estimation schemes for noisy functions. *J Opt Theory Appl*. 2005;126:529–551.
14. Myers R, Montgomery D. *Response Surface Methodology*. New York: Wiley, 2002.
15. Gutmann H-M. A radial basis function method for global optimization. *J Global Optim*. 2001;19:201–227.
16. Jones D. A taxonomy of global optimization methods based on response surfaces. *J Global Optim*. 2001;21:345–383.
17. Jones D, Schonlau M, Welch W. Efficient global optimization of expensive black-box functions. *J Global Optim*. 1998;13:455–492.
18. Regis R, Shoemaker C. Improved strategies for radial basis function methods for global optimization. *J Global Optim*. 2007;37:113–135.
19. Goovaaerts P. *Geostatistics for Natural Resources Evolution*. New York: Oxford University Press, 1997.
20. Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. *Statist Sci*. 1989;4:409–423.
21. Romero V, Burkardt J, Gunzburger M, Petersen J. Comparison of pure and "Latinized" centroidal Voronoi tessellation against various other statistical sampling methods. *Rel Eng Sys Saf*. 2006;91:1266–1280.
22. Giunta A, Wojtkiewicz S, Eldred M. Overview of modern design of experiments methods for computational simulations. *AIAA*. 2003;0649:1–17.
23. Simpson T, Lin D, Chen W. Sampling strategies for computer experiments: design and analysis. *Int J Rel Appl*. 2001;2:209–240.
24. Barber C, Dobkin D, Huhdanpa H. The quickhull algorithm for convex hulls. *ACM Trans Math Soft*. 1996;22:469–483.
25. Box G, Hunter S, Hunter WG. *Statistics for Experimenters: Design, Innovation, and Discovery*, 2nd ed. New York: Wiley-Interscience, 2005.
26. Bindal A, Ierapetritou MG, Balakrishnan S, Armaou A, Makeev AG, Kevrekidis IG. Equation-free, coarse-grained computational optimization using timesteppers. *Chem Eng Sci*. 2006;61:779–793.
27. Gillespie DT. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J Comput Phys*. 1976;22:403–434.

Manuscript received Aug. 15, 2008, revision received Jan. 11, 2009, and final revision received Feb. 15, 2009.