

# Feasibility Evaluation of Nonconvex Systems Using Shape Reconstruction Techniques

Ipsita Banerjee and Marianthi G. Ierapetritou\*

Department of Chemical and Biochemical Engineering, Rutgers University, Piscataway, New Jersey 08854

Optimal and feasible operation of process plants demands accurate knowledge of the effect of parameter uncertainty on process design and operation. There has been considerable effort toward accurate representation of the feasible operation range, and different metrics have been proposed in the literature to quantify the operational flexibility. While these methods are largely successful in addressing convex problems, their applicability becomes restricted for general nonconvex problems. The feasibility analysis technique proposed in this paper considers the feasible region as an object and applies surface reconstruction ideas to capture and define the shape of the object. The procedure starts by first sampling the feasible region to have a representation of the feasible space and then constructing an  $\alpha$  shape with the sampled points, thus generating a polygonal representation of the feasible parameter space. Finally, any point can be checked for its feasibility by applying the point-in-polygon algorithm. This method is general and can be applied to any convex, nonconvex, or even disjoint problems without any further modifications.

## 1. Introduction

Uncertainties in chemical plants appear because of a variety of reasons. These reasons may be internal, such as fluctuations of the values of reaction constants and physical properties, or external, such as the quality and the flow rates of feed streams. The need to account for uncertainty in various stages of plant operations has been identified as one of the most important problems in chemical plant design and operations.<sup>1–3</sup>

There are two main problems associated with the consideration of uncertainty in the decision making: the quantification of feasibility and flexibility of a process design and the incorporation of the uncertainty within the decision stage. The quantification of process feasibility is most commonly addressed by utilizing the feasibility function introduced by Swaney and Grossmann that requires constraint satisfaction over a specified uncertainty space, whereas the flexibility evaluation is associated with a quantitative measure of the feasible space. Halemane and Grossmann<sup>4</sup> proposed a feasibility measure for a given design based on the worst points for feasible operation, which can be mathematically formulated as a max–min–max optimization problem:

$$\chi(d) = \max_{\theta \in T} \min_z \max_{j \in J} f_j(d, z, \theta) \quad (1)$$

where  $T$  is the feasible space of  $\theta$  described as  $T = \{\theta | \theta^L \leq \theta \leq \theta^U\}$ , where  $\theta^L$  and  $\theta^U$  are the lower and upper bounds, respectively.

The general formulation for quantifying flexibility, known as the *flexibility index problem*, can be defined as the determination of the maximum deviation,  $\Delta$ , that a given design,  $d$ , can tolerate, such that every point  $\theta$  in the uncertain parameter space ( $T(\delta)$ ) is feasible.<sup>5</sup> A well-studied choice is the hyper-rectangle representation  $\delta$ ,  $T(\delta) = \{\theta | \theta^N - \delta\Delta\theta^- \leq \theta \leq \theta^N + \delta\Delta\theta^+\}$ , where  $\Delta\theta^+$

and  $\Delta\theta^-$  are the expected deviations of the uncertain parameters in the positive and negative directions and  $\delta$  is the deviation along a specified direction. Other descriptions of  $T(\delta)$ , such as the parametric hyper-ellipsoid, have also been investigated.<sup>6</sup>

The flexibility index can be determined from the formulation proposed by Swaney and Grossmann<sup>1</sup> as

$$F = \max \delta$$

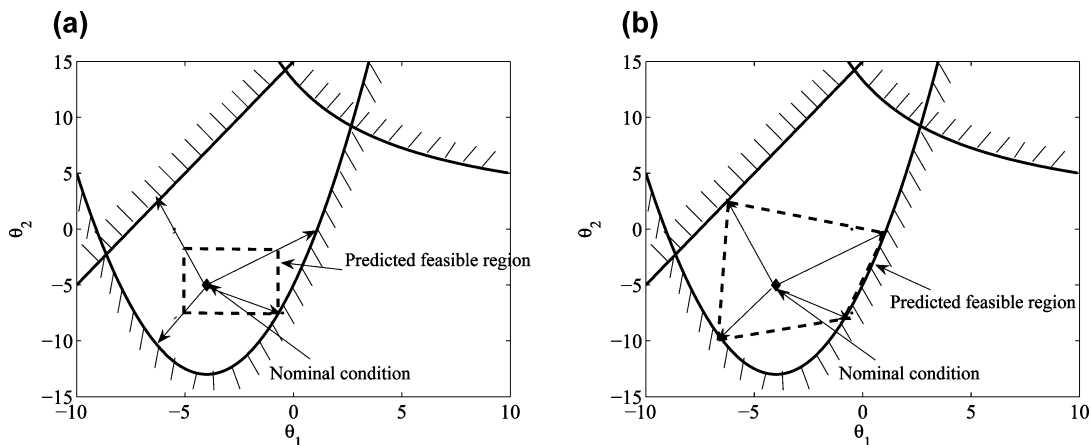
$$\text{subject to } \max_{\theta \in T(\delta)} \psi(\theta, d) \leq 0 \quad (2)$$

$$\delta \geq 0$$

One approach toward determination of the flexibility index is by vertex enumeration, in which the maximum displacement is computed along each vertex direction. This scheme is based on the assumption that the critical points,  $\theta^c$ , lie at the vertexes of  $T(\Delta^c)$ , which holds only under certain convexity conditions. Other existing approaches to quantify flexibility involve (1) deterministic measures such as the resilience index, RI, proposed by Saboo et al.<sup>7</sup> and (2) stochastic measures such as the design reliability proposed by Kubic and Stein<sup>8</sup> and the stochastic flexibility index proposed by Pistikopoulos and Mazzuchi<sup>9</sup> and Straub and Grossmann.<sup>10</sup> Recently, Ierapetritou and co-workers<sup>11</sup> introduced a new approach to quantify process feasibility based on the description of the feasible region by an approximation of the convex hull. Their approach results in an accurate representation of process feasibility. However, it also relies on the utilization of a process model and specific convexity assumptions.

This approach has been further extended in the present work to capture the detailed shape of the feasible region by utilizing the ideas of shape reconstruction used in the field of computer graphics. The main problem definition for surface reconstruction is to, given a set of range points, reconstruct a manifold that closely approximates the surface of the original model. Range data is a set of discrete points in three-

\* To whom correspondence should be addressed. Tel.: (732) 445-2971. E-mail: marianth@sol.rutgers.edu.



**Figure 1.** Feasibility analysis using (a) a flexibility index and (b) a convex hull.

dimensional space which have been sampled from the physical environment or can be obtained using laser scanners which generate data points on the surface of an object. The problem naturally arises in a variety of practical situations such as range scanning an object from multiple viewpoints, recovery of biological shapes from two-dimensional slices, interactive surface sketching, and so forth. Surface reconstruction finds extensive applications in the areas of automatic mesh generation and geometric modeling and molecular structure and protein folding analysis.

The problem of feasibility analysis is analogous to the ideas of surface reconstruction, since the main effort of feasibility analysis lies in identifying and accurately estimating the boundary of the feasible region. In the previous approaches this boundary is approximated by linear inequalities, either by incorporating a hyper-rectangle<sup>1</sup> or by describing an approximation of the convex hull<sup>11</sup> inside the feasible space. These methods can have satisfactory performance in the case of convex, connected feasible regions but will be inaccurate for the cases of nonconvex or disjoint feasible regions. On the other hand, the surface reconstruction scheme can successfully describe both nonconvex and disjoint regions by defining the bounding surface as piecewise linear functions.

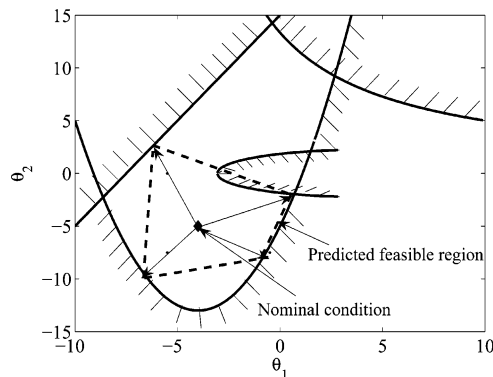
**1.1. Motivating Example.** To illustrate the shortcomings of the existing feasibility analysis approaches and the necessity for an efficient and accurate approach, a simple example has been considered in this section. The feasible region is defined by the following sets of convex and nonconvex constraints:

$$f_1 = \theta_2 - 2\theta_1 - 15 \leq 0 \quad (3)$$

$$f_2 = \frac{\theta_1^2}{2} + 4\theta_1 - 5 - \theta_2 \leq 0 \quad (4)$$

$$f_3 = \theta_2(6 + \theta_1) - 80 \leq 0 \quad (5)$$

where  $\theta_1$  and  $\theta_2$  are the uncertain parameters in the range  $-10 \leq \theta_1 \leq 10$  and  $-15 \leq \theta_2 \leq 15$ . The nominal point is  $(-4, -5)$  as illustrated in Figure 1a. To define the feasible region using the concept of flexibility index, one has to move from the nominal point toward the four vertexes until the constraint is violated. The scaled parameter deviations,  $\delta$ , obtained for the four vertexes are 0.41, 0.37, 0.25, and 0.44. The rectangle constructed by using the minimum of these values ( $\delta_{\min} = 0.25$ )



**Figure 2.** Performance of the convex hull approach in the presence of a nonconvex constraint.

defines the predicted feasible region, as shown in Figure 1a. An improvement over the construction of a rectangle, or a hyper-rectangle in higher dimensions, is to describe the feasible region by the convex hull of the evaluated boundary points. As illustrated in Figure 1b, the convex hull approximation can capture the feasible region better than the rectangle, and the performance of the convex hull approach can be improved further by including more boundary points in the construction of the hull. Another advantage of this approach is that it overcomes the dependence on the nominal point, which is a strong limitation of the flexibility index approach. If the nominal point is located close to one of the constraints or is on the constraint, then the flexibility index approach will fail, since the minimum parameter deviation ( $\delta_{\min}$ ) will be either a very small value or 0. The convex hull approach overcomes this limitation since it considers all the boundary points and not the minimum deviation from the nominal value. However, the convex hull approach is limited in its application to only convex and 1-D quasi-convex feasible regions, and its performance deteriorates in the presence of nonconvex constraints. To illustrate this, an additional nonconvex constraint is introduced in the previous example, given by

$$f_4 = 10 - \frac{(\theta_1 - 4)^2}{5} - 2\theta_2^2 \leq 0 \quad (6)$$

Figure 2 illustrates the new feasible region obtained by the introduction of this constraint. The boundary points can be evaluated as before; three of the boundary points remain the same as before, and one becomes restricted

by the new nonconvex constraint. When the convex hull is constructed with these boundary points, it is found to enclose a portion of the region rendered infeasible by the new constraint. Addition of any number of boundary points will not prevent the overprediction of the feasible region defined by the convex hull.

The presented example clearly demonstrates the need for a more efficient feasibility analysis scheme which can accurately capture the feasible region even in the presence of nonconvex constraints.

As described in the Introduction, the present work proposes a feasibility analysis scheme based on surface reconstruction ideas, in particular, the  $\alpha$ -shape methodology for surface reconstruction. The paper is organized by first giving a description of the  $\alpha$ -shape methodology for surface reconstruction. Section 3 gives a detailed description of the application of the  $\alpha$ -shape approach for the feasibility problem followed by some representative case studies in section 4.

## 2. $\alpha$ -Shape Approach

There are various approaches described in the literature for determining the shape of a pattern class from sampled points. Many of these approaches are concerned with the efficient construction of convex hulls for a set of points in the plane. Jarvis<sup>12</sup> was one of the first to consider the problem of computing the shape as a generalization of the convex hull of a planar point set. He presented several algorithms based on nearest neighbors that compute the shape of a finite point set. Akl and Toussaint<sup>13</sup> proposed a way of constructing a convex hull by identifying and ordering the extreme points of a point set, and the convex hull serves to characterize the shape of such a set. Fairfield<sup>14</sup> has put forward a notion of the shape of a planar set based on the closest point Voronoi diagram of the set. Different graph structures that serve similar purposes are the Gabriel graph,<sup>15</sup> the relative-neighborhood graph,<sup>16</sup> and their parametrized version, the  $\beta$  skeleton.<sup>17</sup>

A mathematically rigorous definition of a shape was later introduced by Edelsbrunner et al.<sup>18</sup> They proposed a natural generalization of the convex hulls which is referred to as  $\alpha$  hulls. The  $\alpha$  hull of a point set is based on the notion of generalized disks in the plane. The family of  $\alpha$  hulls includes the smallest enclosing circle, the set itself, and an essentially continuous set of enclosing regions between these two extremes.

Edelsbrunner et al.<sup>18</sup> also define a combinatorial variant of the  $\alpha$  hull called the  $\alpha$  shape of a planar set, which can be viewed as the boundary of the  $\alpha$  hull with curved edges replaced by straight edges. Conceptually,  $\alpha$  shapes are a generalization of the convex hull of a point set  $\mathcal{P}$ , with  $\alpha$  varying from 0 to  $\infty$ . The  $\alpha$  shape of  $\mathcal{P}$  is a polytope that is neither necessarily convex nor connected. For  $\alpha = \infty$ , the  $\alpha$  shape is identical to the convex hull of  $\mathcal{P}$ . However, as  $\alpha$  decreases, the  $\alpha$  shape shrinks by gradually developing cavities. When  $\alpha$  becomes small enough, the polytope disappears and reduces to the data set itself.

To provide an intuitive notion of the concept, Edelsbrunner<sup>18</sup> describes the space  $R^3$  to be filled with Styrofoam and the point set  $\mathcal{P}$  to be made up of a more solid material, such as rock. Now, if a spherical eraser with radius  $\alpha$  curves out the Styrofoam at all positions where it does not enclose any of the sprinkled rocks (the point set  $\mathcal{P}$ ), the resulting object formed will be called an  $\alpha$  hull. The surface of the object can be straightened

by substituting straight edges for the circular ones and triangles for the spherical caps. The obtained object is the  $\alpha$  shape of  $\mathcal{P}$ . It is a polytope in a fairly general sense: it can be concave and even disconnected, it can contain two-dimensional patches of triangles and one-dimensional strings of edges, and its components can be as small as single points. The parameter  $\alpha$  controls the degree of details captured by the  $\alpha$  shape.

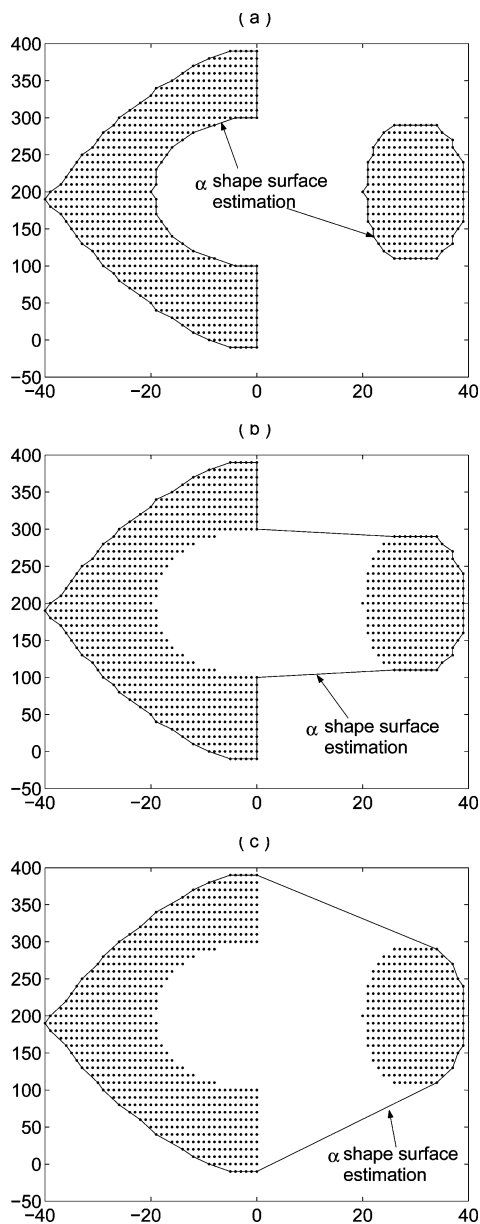
It is possible to generalize all the concepts involved in the construction of an  $\alpha$  shape (i.e.  $\alpha$  hulls,  $\alpha$  complexes, Delaunay triangulation, and Voronoi diagrams) to a finite set of points  $\mathcal{P}$  in  $R^d$ , for an arbitrary dimension  $d$ . This generalization, combined with an extension to weighted points, is developed by Edelsbrunner.<sup>19</sup> However, the implementation details of the problem become progressively more complex with increasing  $d$ , and the worst-case complexity of the problem grows exponentially.

**2.1. Selection of  $\alpha$ .** The computed  $\alpha$  shape of a given set of sample points explicitly depends on the chosen value of  $\alpha$ , which controls the level of detail of the constructed surface. Mandal et al.<sup>20</sup> present a systematic methodology for selecting the value of  $\alpha$  in  $R^2$ . They visualize the problem of obtaining the shape of  $\mathcal{P}$  as a set estimation problem where an unknown set  $\mathcal{A}$  is to be estimated on the basis of a finite number of points  $X_1, X_2, \dots, X_n \in \mathcal{A}$ . As  $n$  increases,  $\mathcal{A}(n)$  will cover many parts of  $\mathcal{A}$  and, hence, the value of  $\alpha$  for  $\mathcal{A}(n)$  should depend on the sample size ( $n$ ); thus,  $\alpha$  is a function of  $n$ . Additionally,  $\alpha$  should also be a function of the interpoint distance of the sampled  $n$  points of  $\mathcal{A}(n)$ . To account for the dependence on interpoint distance, the authors have constructed the minimum spanning tree (MST) of the sampled data points. If  $l_n$  represents the sum of edge weights of the MST, where the edge weight is taken to be the Euclidean distance between the points, then the appropriate value of  $\alpha$  for the construction of the  $\alpha$  shape is given by

$$h_n = \sqrt{\frac{l_n}{n}} \quad (7)$$

where  $n$  is the total number of sample points.

To illustrate the performance of an  $\alpha$  shape in capturing the shape of an object, a disjoint, nonconvex object is chosen, as illustrated in Figure 3. The sampled points represent a 2-D object, which is the input to the  $\alpha$ -shape construction code. The  $\alpha$  shape identifies from the input data set points which lie on the boundary of the object. These points are joined by a line to describe the surface of the object. The above figure also illustrates the dependence of the captured shape on the chosen value of  $\alpha$ . The  $\alpha$  value estimated by performing the minimum spanning tree operation is 120, at which value the  $\alpha$  shape was found to capture the nonconvex as well as the disjoint nature of the object (Figure 3a). By further increasing the value of  $\alpha$ , the performance of the  $\alpha$  shape deteriorates, as illustrated in Figure 3b, and at a very high  $\alpha$  value the  $\alpha$  shape forms a convex hull of the object (Figure 3c). Hence, the level of detail captured by the  $\alpha$  shape strongly depends on the chosen value of  $\alpha$ , and progressively decreasing the value of  $\alpha$  will capture the shape more accurately. For the case of uniform sampling, where the distance between the sampled points is uniform, the predicted shape is not very sensitive to the value of  $\alpha$  for lower  $\alpha$  values. However, extreme reduction of the value of  $\alpha$  will result



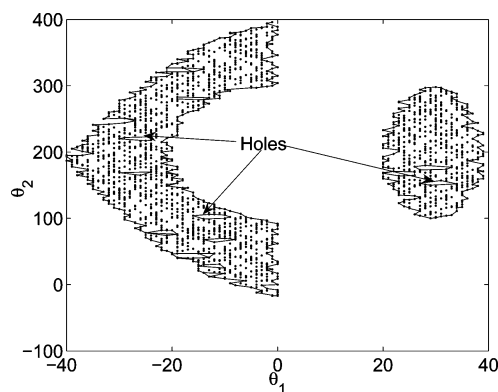
**Figure 3.** Performance of the  $\alpha$  shape for different  $\alpha$  values: (a) 120; (b) 200; (c) 100 000.

in the original data set and will fail to capture any shape.

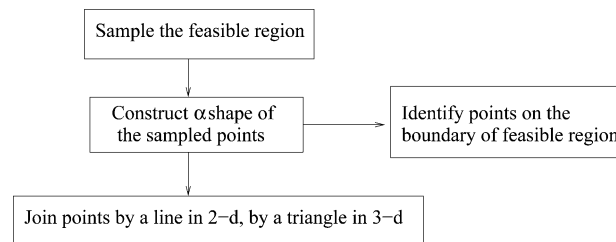
In the case of random sampling, when the interpoint distance is not constant, the predicted shape is more sensitive to the lower  $\alpha$  values. For lower values of  $\alpha$ , the predicted shape shows holes scooped out from within the samples which are not actually present but are an artifact of sampling. Such an example is illustrated in Figure 4 for the same object described by a set of random samples. The shape produced by an  $\alpha$  value of 6 has an accurate outside boundary, but holes were inserted unnecessarily inside the body. A higher value of  $\alpha = 7$ , however, eliminated the holes. For progressively lower  $\alpha$  values, the holes continue increasing; finally, for an extremely low  $\alpha$  value, the  $\alpha$  shape degenerates to the original data set without capturing any shape.

### 3. Feasibility Analysis Using $\alpha$ Shape

The overall aim of a feasibility analysis is the determination of the range of parameters over which a



**Figure 4.** Performance of the  $\alpha$  shape with random points.

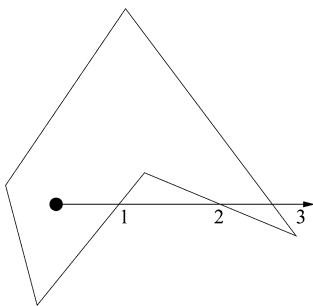


**Figure 5.** Algorithm of feasibility analysis using  $\alpha$ -shape methodology.

particular process is feasible. A formal definition of this problem is to obtain a mathematical description of the region in parameter space bounded by the process constraints. This region can be considered analogous to an object, the shape or surface of which can be estimated using the  $\alpha$ -shape technique. The input to any surface reconstruction algorithm needs to be a set of points representing the object, whose surface needs to be determined. Hence, the first step is the generation of good sample data points to represent the feasible region under consideration. The  $\alpha$  shape can then be constructed for the sampled data, using the  $\alpha$  estimate obtained from the minimum spanning tree of the data set. The  $\alpha$  shape essentially identifies points from the input data set that lie on the surface of the object. These points are then connected by a line in two dimensions and a triangle in three dimensions, giving rise to a polygon enclosing the feasible region. Figure 5 illustrates the above-mentioned steps.

Having defined the surface or shape of the feasible region, the next step involves the determination of whether a particular point belongs to the feasible region. Since the feasible region has been approximated by a polygon, a simple way to check if a point is inside the polygon is by using one of the point-in-polygon tests.<sup>21</sup> One method to determine whether a point is inside a region is the ray tracing algorithm, which states that a point is inside a polygon if, for any semi-infinite ray from this point, there is an odd number of intersections of the ray with the polygon's edges (Figure 6). Conversely, a point is outside a polygon if the ray intersects the polygon's edges an even number of times or does not intersect at all. Following this, whenever a parameter needs to be checked for feasibility in a polygon-estimated feasible region, a semi-infinite ray is drawn from the point in any direction and the number of intersections is noted, which determines whether the point is feasible or not.

**3.1. Sampling Technique.** Obtaining a good representation of the feasible region is the first and very



**Figure 6.** Point-in-polygon test: an odd number of intersections means the point is inside; an even number of intersections means the point is outside.

important step in the  $\alpha$ -shape technique of feasibility analysis. In many problems, typically, the feasible region covers only a very restricted region of the entire parameter space. Hence, sampling techniques covering the entire range of the uncertain parameters prove to be inefficient, particularly when evaluation of the process constraints is an expensive operation. Most of the common sampling techniques sample the parameter space based on the distribution of the uncertain parameters, which are considered to be uniform for the cases considered here for simplicity of presentation. Under this condition, it will lead to uniform sampling of the entire parameter space, irrespective of whether the sampled points are feasible or not. A new sampling technique is, thus, introduced here, which takes advantage of the fact that, typically, a small section of the entire parameter space is feasible. The sampling problem is formulated as an optimization problem and is solved using genetic algorithm (GA). The use of GA as a solution procedure proves to be very efficient for this problem since the search scheme has the inherent property of concentrating around regions having good solutions, which is the feasible solution for the problem addressed here.

The purpose of the sampling step is to have a good approximation of the feasible space by the minimum evaluation of the expensive feasibility function. The formulation of the sampling problem as an optimization problem is given by

$$\begin{aligned} & \max_{\theta} V_{\text{feas}} \\ & \text{subject to } f_1(\theta) \leq 0 \\ & \quad f_2(\theta) \leq 0 \\ & \quad \vdots \\ & \quad f_n(\theta) \leq 0 \end{aligned} \quad (8)$$

where  $V_{\text{feas}}$  is the volume of the feasible region, evaluated by constructing the  $\alpha$  shape using the sampled feasible points. The optimization variables are the parameter values  $\theta$ , which are sampled by GA to optimize the objective, and  $f_1$ ,  $f_2$ , and  $f_n$  are the constraints of the feasibility problem evaluated at  $\theta$ . However, in this formulation there is no optimal value of the variable  $\theta$  which will maximize the volume; we are interested in the entire sampled set of feasible  $\theta$  values, which we use to evaluate the volume by constructing an  $\alpha$  shape over the entire set of feasible  $\theta$  values. Since the objective is to maximize the volume of the feasible space, whenever a chosen value of  $\theta$  satisfies the constraint functions, the volume is evalu-

ated to update the objective function. When the value is not feasible, there is no need to reevaluate the volume since it will not change, but the fitness function is penalized by assigning it a small value. Solving this problem using GA reduces the required number of function evaluations by minimizing the unnecessary evaluation of infeasible parameter space. To solve the optimization problem by GA, the optimization variables are encoded as a string of bits, and these strings are appended together to form a chromosome. The solution procedure starts by generating a population of solutions, where each individual in the population has a particular chromosome value which can be decoded to evaluate the parameter values and the objective function, also called the fitness function. As shown by DeJong,<sup>22</sup> large population sizes exhibit slower convergence than small population sizes, but they tend to converge to a better solution. In the interest of balancing execution time and optimality, the population size is chosen to be of the order of the chromosome size, as suggested by Edwards et al.<sup>23</sup>

$$n_{\text{chromosome}} < n_{\text{population}} < 2n_{\text{chromosome}} \quad (9)$$

The populations are evolved through several generations, following rules such as reproduction, crossover, and mutation, until the objective function cannot be improved any further. The primary objective of the reproduction operator is to emphasize good solutions and eliminate bad solutions in a population. This is achieved by first identifying above-average solutions in a population, making multiple copies of the good solutions, and eliminating bad solutions from the population in order to accommodate multiple copies of the good solution. Some common methods for achieving this include *tournament selection*, *proportionate selection*, *ranking selection*, and so forth.<sup>24</sup> In all the examples presented in this work, tournament selection was chosen to be the reproduction operator. While reproduction can make more copies of the good solution and eliminate the bad solutions, it cannot create a new solution. The crossover operator is applied next to take care of that, where two strings are picked randomly from the mating pool and some portions of the strings are swapped between themselves to create two new strings. If the new strings created by a crossover are good, there will be more copies of them generated by the reproduction operator in the next mating pool. Otherwise, if the new strings are not good, they will not survive beyond the next generation, since reproduction will eliminate them. A uniform crossover between randomly selected pairs is used for this work.

The parameters chosen for GA operations largely govern the computational time as well as the quality of the solution. If the selection operator emphasizes too much on the population's best solution by assigning many copies of it, then the population loses its diversity very quickly. The crossover and mutation operators should be strong enough to retain the diversity of the solution and to create solutions fairly different from the parent solution. In the absence of this, the simulation can converge to a suboptimal solution. On the other hand, if the reproduction operator is weak and the crossover operator very strong, the GA's search procedure behaves like a random search process. The GA parameters chosen for all the examples presented here are  $p_{\text{crossover}} = 0.5$  and a small mutation probability  $p_{\text{mutation}} = 0.02$ .

The working principle of GA is that, by application of its three operators, the number of strings with similarities at certain string positions have been increased from the initial population to the new population. These similarities, referred to as *schema* in GA literature, represent a set of strings with similarity at certain string positions. Thus the schema can be thought of as representing certain regions in the search space or specific ranges of the parameter value. Hence, by having more copies of a schema with a better function value, one can reach the optimal solution without searching the entire variable space but rather by manipulating only a few instances of the search space.

This feature of GA is of particular importance for the present problem since only a limited region of the variable space is feasible, and exploring the entire variable space to have an idea of the feasible region becomes computationally expensive. As mentioned above, the working principle of GA is based on generating multiple numbers of the good solutions. Hence, evaluating the volume for each of the feasible parameter values ( $\theta$ ) will reduce the efficiency of the procedure because of the repetition of the solution. To avoid this, a memory of the sampled parameter value is maintained and updated. For every generated chromosome in the population of the GA simulation, the stored parameter values are searched to check for uniqueness of the new solution. If the new solution is unique, then the constraints are evaluated; otherwise, it is updated from the memory.

#### 4. Case Studies

The idea of using surface reconstruction for the estimation of the feasible region is illustrated by a few case studies.

**4.1. Motivation Example Revisited.** The example considered in section 1.1 is revisited here, following the proposed feasibility analysis approach. Figure 1 illustrates the actual nature of the feasible region bounded by the inequalities (eqs 3–6). To estimate this region using the technique of  $\alpha$  shape described before, first one needs to generate points representing the shape of the region by sampling the feasible space. The technique described in section 3.1 is used to sample the feasible space. Following this scheme, the optimization problem for sampling is given by

$$\begin{aligned} \max_{\theta_1, \theta_2} V_{\text{feas}} \\ \theta_2 - 2\theta_1 - 15 &\leq 0 \\ \frac{\theta_1^2}{2} + 4\theta_1 - 5 - \theta_2 &\leq 0 \\ 10 - \frac{(\theta_1 - 4)^2}{5} - \frac{\theta_2^2}{0.5} &\leq 0 \\ \theta_2(6 + \theta_1) - 80 &\leq 0 \end{aligned} \quad (10)$$

Both of the uncertain parameters are considered to vary within the range of  $-20$  to  $20$ . To solve the problem using genetic algorithm, the parameters  $\theta_1$  and  $\theta_2$  are encoded as bits, with 7 bits for each parameter, giving rise to a 14 bit chromosome. A population size of 20 is chosen for this problem following the guideline of Edwards et al.<sup>23</sup> Figure 7 illustrates the convergence of

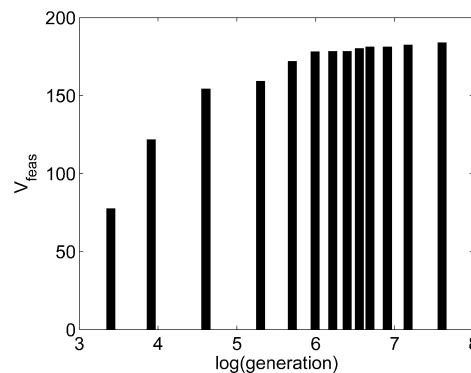


Figure 7. Variation of the optimal solution with generations.

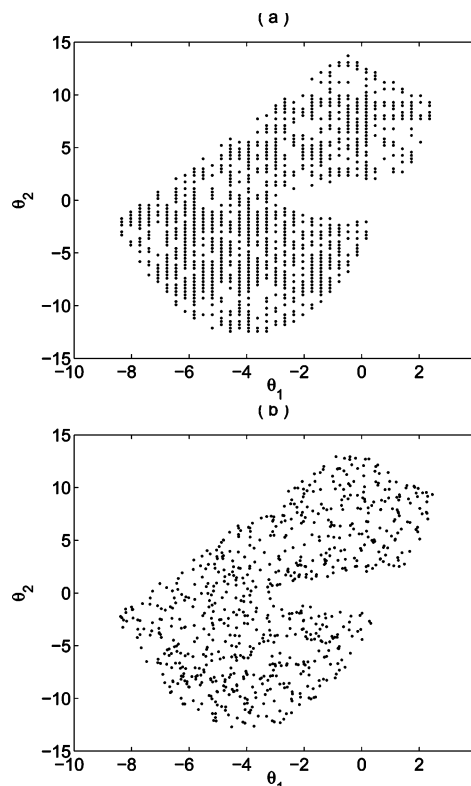
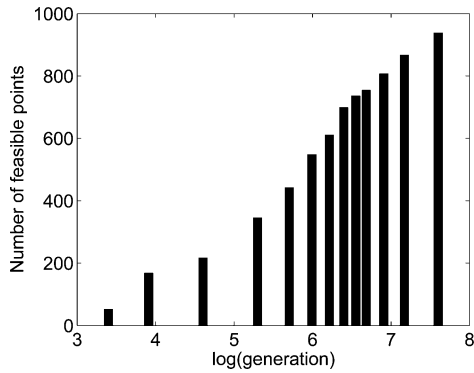
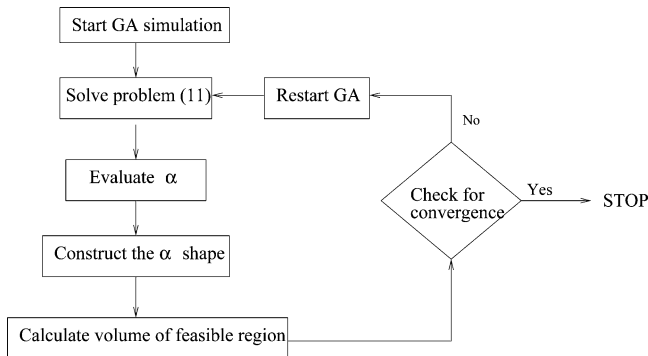


Figure 8. Sampling of the feasible space using (a) a genetic algorithm and (b) random samples.

the optimal solution with an increasing number of generations. Chromosome evolution through 1 000 generations will require a total of 20 000 function calls, which gives rise to 938 feasible points as illustrated in Figure 8. However, a large number of these function calls are for the same value of the parameters and are not unique. Hence, by keeping a memory of the parameter values and their corresponding solutions, it is possible to avoid the repeating function evaluations and, hence, increase the efficiency of the code by evaluating the constraints only for unique parameter values. By following this procedure, it was observed that out of 40 000 required function calls, only 3064 were unique. Hence, this procedure could generate 938 feasible points by 3064 function evaluations. The same problem was solved by drawing random samples in the range of  $-20$  to  $20$  for both uncertain parameters, as illustrated in Figure 8b. To generate 950 feasible points using a random search required 9830 function calls. This clearly shows the advantage of the proposed approach of sampling the feasible space.



**Figure 9.** Variation of the optimal solution with generations, using the formulation in eq 11.



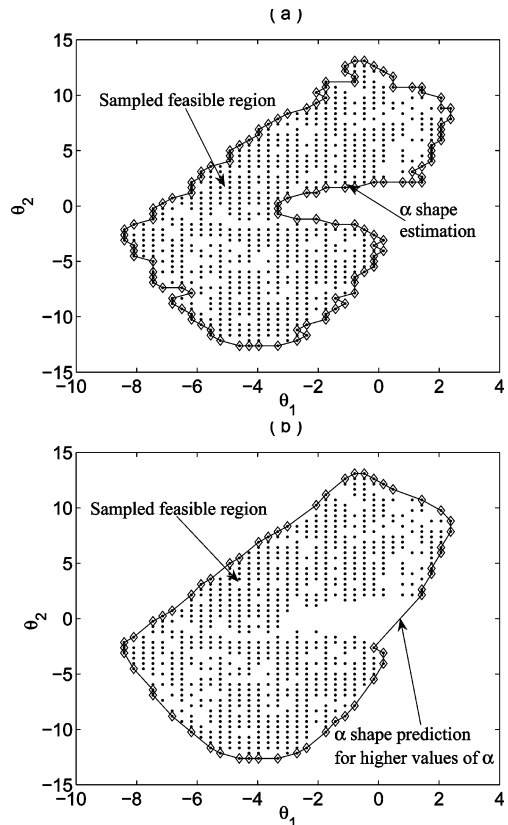
**Figure 10.** Modified algorithm for sampling of the feasible region.

The volume of the feasible region is calculated by generating an  $\alpha$  shape of the sampled points, which identifies the points forming the boundaries of the object, which are then joined by a straight line as illustrated by Figure 11. The value of  $\alpha$  plays a crucial role in determining the degree of details captured by the  $\alpha$  shape. The  $\alpha$  value determined by the procedure outlined in section 2.1 for the 938 sampled points is 25, which was found to capture the nonconvex nature of the object with adequate accuracy, as illustrated in Figure 11a. However, if the value of  $\alpha$  is increased, the  $\alpha$ -shape prediction tends to lose its accuracy, and Figure 11b illustrates the performance of the  $\alpha$  shape at a much higher value of  $\alpha = 1000$ . At this value, it failed to capture the nonconvex nature of the shape. With further increase in the value of  $\alpha$ , the  $\alpha$  shape reproduces the convex hull of the point set.

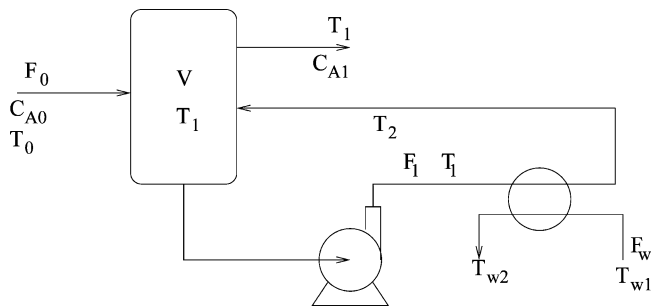
An alternative formulation for generating the sampled data set is given by

$$\begin{aligned} \max_{\theta_1, \theta_2} \quad & \sum \Theta_{\text{feas}} \\ \theta_2 - 2\theta_1 - 15 & \leq 0 \\ \frac{\theta_1^2}{2} + 4\theta_1 - 5 - \theta_2 & \leq 0 \\ 10 - \frac{(\theta_1 - 4)^2}{5} - \frac{\theta_2^2}{0.5} & \leq 0 \\ \theta_2(6 + \theta_1) - 80 & \leq 0 \end{aligned} \quad (11)$$

where  $\Theta_{\text{feas}}$  represents a feasible sample point and the



**Figure 11.** Performance of the  $\alpha$  shape in predicting the feasible space using different  $\alpha$  values: (a)  $\alpha = 25$ ; (b)  $\alpha = 1000$ .



**Figure 12.** Reactor-cooler example.

objective is to maximize the total number of sampled feasible points. This formulation is computationally less demanding since it does not require the volume evaluation of the feasible region at every step. However, it suffers from the disadvantage of a lack of a convergence criterion as illustrated in Figure 9. To overcome this problem, a hybrid of these two formulations (eqs 10 and 11), where the main algorithm is evolved according to the formulation in eq 11, is used and the volume is evaluated only at intermediate points to check for convergence of the simulation. The overall procedure is illustrated in Figure 10.

**4.2. Process Operation Example.** This example represents the flow sheet shown in Figure 12, consisting of a reactor and a heat exchanger<sup>25</sup> where a first-order exothermic reaction  $A \rightarrow B$  is taking place. The existing design has a reactor volume ( $V$ ) of 4.6 m<sup>3</sup> and a heat exchanger area ( $A$ ) of 12 m<sup>2</sup>. Two uncertain parameters are considered, the feed flow rate,  $F_0$ , and the activation energy,  $E/R$ . The mathematical model of this process is given by

$$F_0(c_{A0} - c_{A1})/c_{A0} = Vk_0 \exp(-E/RT)c_{A1}$$

$$(-\Delta H)F_0(c_{A0} - c_{A1})/c_{A0} = F_0 C_p (T_1 - T_0) + Q_{HE}$$

$$Q_{HE} = F_1 C_p (T_1 - T_2)$$

$$Q_{HE} = F_w C_{pw} (T_{w2} - T_{w1})$$

$$Q_{HE} = AU\Delta T_{ln}$$

$$\Delta T_{ln} = \frac{(T_1 - T_{w2}) - (T_2 - T_{w1})}{\ln(T_1 - T_{w2})/(T_2 - T_{w1})}$$

$$V^d \geq V$$

$$(c_{A0} - c_{A1})/c_{A0} \geq 0.9$$

$$311 \leq T_1 \leq 389$$

$$T_1 - T_2 \geq 0.0$$

$$T_{w2} - T_{w1} \geq 0.0$$

$$T_1 - T_{w2} \geq 11.1$$

$$T_2 - T_{w1} \geq 11.1$$

$$T_0 = 333 \text{ K}, \quad T_{w1} = 300 \text{ K}, \quad U = 1635 \text{ kJ}/(\text{m}^2 \text{ h K})$$

$$C_p = 167.4 \text{ kJ}/\text{kmol}, \quad c_{A0} = 32.04 \text{ kmol}/\text{m}^3, \\ -\Delta H = 23260 \text{ kJ}/\text{kmol}$$

The range of uncertain parameters  $E/R$  and  $F_0$  over which the design remains feasible is illustrated in Figure 13. The aim is to have a description of the range of the parameters  $E/R$  and  $F_0$  over which the operation remains feasible. Following the proposed approach for feasibility analysis, the feasible space is first sampled by solving the problem at different values of the parameters and a representation of the feasible region is obtained. In the next step, these sampled points are analyzed by  $\alpha$ -shape methodology to identify points lying on the boundary of the feasible region. These identified surface points are joined by straight lines to obtain a polygonal estimation of the feasible region. Figure 13 compares the actual feasible region with that of a shape estimation obtained with 400 sample points, which was found to perform with great accuracy. To understand the effect of sample density on the performance of the  $\alpha$ -shape procedure, the feasible region evaluation was performed with lower numbers of sample points, 100 and 25, as illustrated in Figure 14. The  $\alpha$ -shape prediction was found to underpredict the feasible region since the sampling density was inadequate in capturing the entire region. However, there was no overprediction of the nonconvex feasible region. Figure 14b also compares the performance of the convex hull with that of the  $\alpha$  shape, where it is observed that, even though the sampling density was very low, the  $\alpha$  shape still captured the nonconvex nature of the feasible region. The convex hull is constructed by performing line searches toward the vertexes of uncertain space to locate points on the boundary of the feasible region. The convex hull covers a larger percentage of the feasible region compared to the  $\alpha$  shape for the case of sparse sampling, but it overpredicts the feasible region over

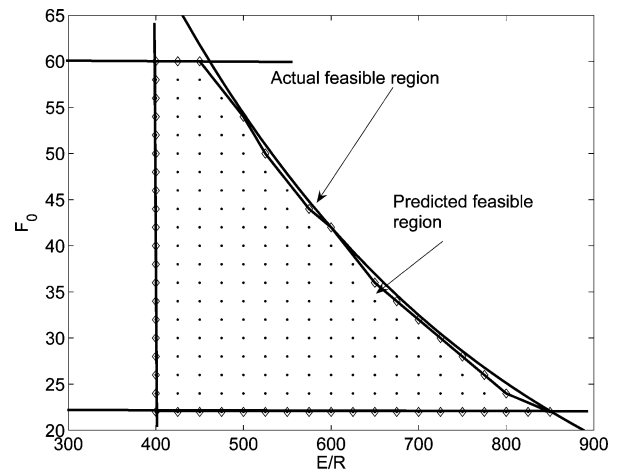


Figure 13. Performance of the  $\alpha$  shape in predicting the feasible space of the reactor-cooler example.

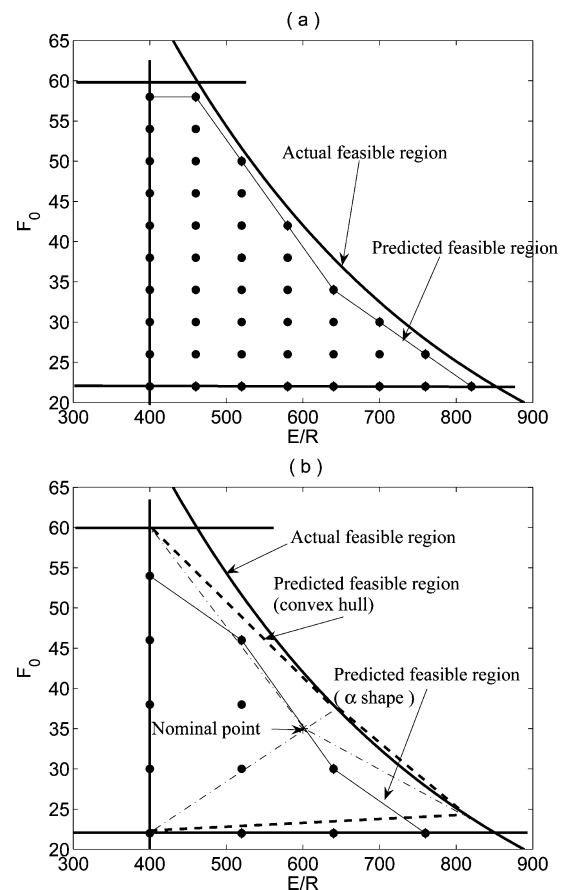
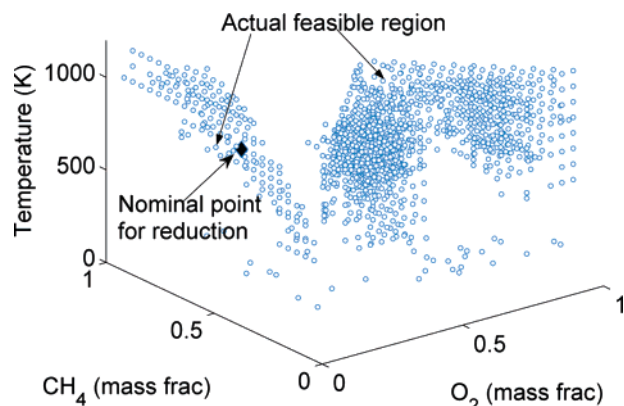


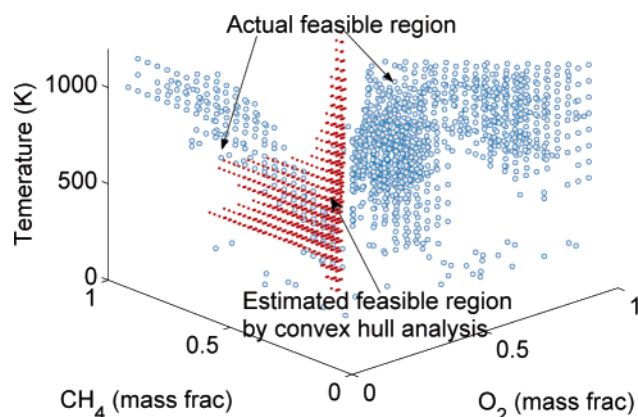
Figure 14. Effect of sampling density on the performance of the  $\alpha$ -shape prediction: (a) 100 points; (b) 25 points.

the nonconvex constraint. The performance of the  $\alpha$  shape is directly dependent on the information captured by the sampling of the feasible space. However, it is important to know that, in the absence of sufficient information, the  $\alpha$  shape will have poor prediction of the feasible space but it will not lead to erroneous results.

**4.3. Range of Validity of the Reduced Kinetic Model.** The last example comes from the area of reactive flows, in particular, combustion simulation. In the numerical simulation of combustion, it is often required to represent the detailed reaction models by simplified models. Because of this simplification, the



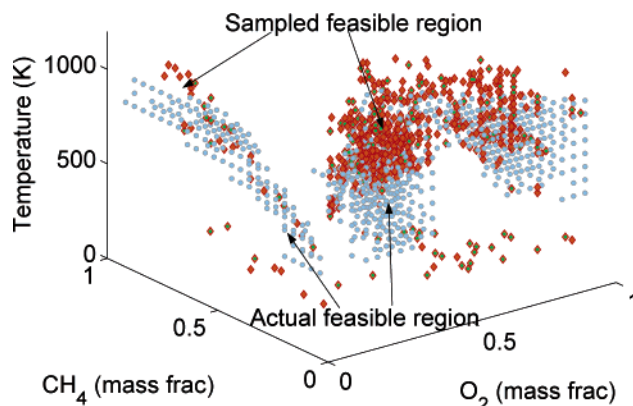
**Figure 15.** Feasible region of a reduced model of the GRI-3.0 mechanism involving 17 species and 59 reactions.



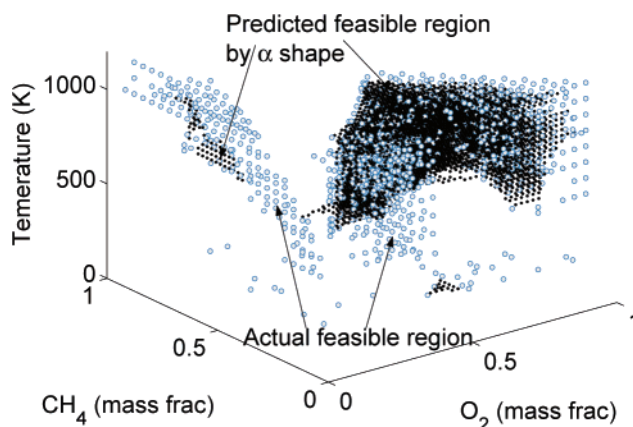
**Figure 16.** Predicted feasible region of the reduced model obtained by the convex hull approach.

reduced model loses its generality and remains valid over a limited range of conditions of species concentration and temperature. Identification of the feasible region of a reduced model and its accurate quantification is a crucial step in the reactive flow simulations.

Figure 15 represents a typical feasible region of a reduced  $\text{CH}_4$  mechanism consisting of 17 species and 59 reactions, where the detailed model consists of 53 species and 325 reactions. The nominal point of reduction for this problem is  $\text{O}_2 = 0.005$  mass fraction,  $\text{CH}_4 = 0.38$  mass fraction, and  $T = 950$  K. As illustrated in Figure 15, the feasible region can be highly nonconvex, even disjoint, for which cases the conventional techniques are not efficient. The problem is solved using the convex hull approach, where the boundary points are identified by moving in different directions from the nominal point. Figure 16 illustrates the feasible region predicted by the convex hull. The reduced mechanism is feasible in two disjoint regions, a small region at low  $\text{O}_2$  concentration and a considerable portion toward the higher  $\text{O}_2$  concentration. However, the nominal point lies in the low  $\text{O}_2$  region, hence restricting the convex hull estimate within this small portion and preventing it from providing any estimate of the major range of feasible operation. Figure 15 is generated by performing a grid search over the variable space, where at each point one needs to determine the deviation of the reduced model from the detailed mechanism due to the change in the conditions of species concentration and temperature. This evaluation is an expensive operation, and performing a grid search over the entire variable space is not feasible. The sampling technique described



**Figure 17.** Sampled feasible region of a reduced model of the GRI-3.0 mechanism.



**Figure 18.** Predicted feasible region of a reduced model.

in section 3.1 is used to have a representation of the feasible space. The three variables,  $\text{O}_2$  mass fraction,  $\text{CH}_4$  mass fraction, and temperature, are coded using 5 bits for each parameter, giving rise to a 15 bit chromosome. The GA simulation is run with a population size of 20, evolved through 250 generations, which will require 5000 function evaluations. However, due to the retained memory of the previous calls, the number of function evaluations was reduced to 1809, of which 778 points were determined to be feasible. The generation of this feasible region using GA requires approximately 25 h, whereas using random sampling to generate 778 feasible points requires 70 h.

Figure 17 illustrates the sampled feasible region as compared to the actual feasible region obtained by the grid search. An  $\alpha$  shape is then constructed with this sampled data set. In three-dimensional space, the  $\alpha$  shape identifies points on the boundary of the object, which are joined in a triangle to form the surface of the object. Having obtained the surface of the feasible region, the next step is determination of whether a point lies inside the feasible region, which can be done using the point-in-polygon test. Using this procedure, the predicted feasible region is constructed, as illustrated in Figure 18. The  $\alpha$ -shape procedure is found to capture 71% of the actual feasible space, which can still be improved by generating more sample points by allowing the GA simulation of the sampling step to evolve for more generations. This is a large improvement over the convex hull approach, which covers only 3% of the overall feasible space, mainly because of the disjoint nature of the feasible space. Moreover, as illustrated in Figure 16, the convex hull approach also overestimates

the feasible region, which is a consequence of the nonconvexity of the feasible space. Since an  $\alpha$  shape can accurately capture the nonconvexity of the shape, it results in no overprediction.

## 5. Discussion

The problem of evaluation of the feasible range of process operation is addressed in this paper using surface reconstruction ideas. The problem definition is to evaluate and quantify the uncertain parameter range over which a process retains its feasibility. In the present approach, the feasible region is viewed as an object, with process constraints defining the boundary of the object. The surface reconstruction ideas are used to define the shape of the object. The procedure starts by first sampling the feasible region to obtain a representation of the feasible space. An  $\alpha$  shape is then constructed of the sampled points, which identifies points forming the boundary of the object. These points are joined to have a polygonal representation of the feasible region. Finally, determination of whether a point is feasible or not can be done by a point-in-polygon check. Examples are presented to illustrate the performance of the proposed scheme in nonconvex and even disjoint problems.

The application of the proposed technique in higher dimensions becomes computationally challenging. One way of dealing with this issue is by reducing the dimensionality of the problem. The ideas of principal component analysis<sup>26</sup> can be utilized to map the original uncertainty space to the reduced dimensional space of important eigen directions. The  $\alpha$ -shape ideas can then be applied in the reduced space and the feasibility information mapped back to the original uncertain space. These ideas are currently being explored by the authors.

## Acknowledgment

The authors gratefully acknowledge financial support from the donors of the Petroleum Research Fund, administered by the ACS and the Office of Naval Research under Contract N00014-03-1-0207, and the National Science Foundation (CTS 0224745). The  $\alpha$ -shape code used in this paper was developed by Dr. Ken Clarkson (<http://cm.bell-labs.com/netlib/voronoi/hull.html>).

## Literature Cited

- (1) Swaney, R. E.; Grossmann, I. E. An index for operational flexibility in chemical process design. I: Formulation and theory. *AIChE J.* **1985**, *31*, 621.
- (2) Grossmann, I. E.; Sargent, R. W. H. Optimum design of chemical plants with uncertain parameters. *AIChE J.* **1978**, *24*, 1021.
- (3) Grossmann, I. E.; Halemane, K. P. A decomposition strategy for designing flexible chemical plants. *AIChE J.* **1982**, *28*, 686.

- (4) Halemane, K. P.; Grossmann, I. E. Optimal process design under uncertainty. *AIChE J.* **1983**, *29*, 425.
- (5) Biegler, L. T.; Grossmann, I. E. Retrospective on optimization. *Comput. Chem. Eng.* **2004**, *28*, 1169.
- (6) Rooney, W. R.; Biegler, L. T. Incorporating joint confidence regions into design under uncertainty. *Comput. Chem. Eng.* **1999**, *23*, 1563.
- (7) Saboo, A. K.; Morari, M.; Woodcock, D. C. Design of resilient processing plants. VIII. A resilience index for heat exchanger networks. *Chem. Eng. Sci.* **1983**, *40*, 1553.
- (8) Kubic, W. L.; Stein, F. P. A theory of design reliability using probability and fuzzy sets. *AIChE J.* **1988**, *34*, 583.
- (9) Pistikopoulos, E. N.; Mazzuchi, T. A. A novel flexibility analysis approach for processes with stochastic parameters. *Comput. Chem. Eng.* **1990**, *14*, 991.
- (10) Straub, D. A.; Grossmann, I. E. Design optimization of stochastic flexibility. *Comput. Chem. Eng.* **1993**, *17*, 339.
- (11) Goyal, V.; Ierapetritou, M. G. Determination of operability limits using simplicial approximation. *AIChE J.* **2002**, *48*, 2902.
- (12) Jarvis, R. A. Computing the shape hull of points in the plane. *Proc. IEEE Comput. Soc. Conf. Pattern Recognit. Image Processes* **1977**, 231.
- (13) Akl, S. K.; Toussaint, G. T. Efficient convex hull algorithm for pattern recognition applications. *Proc. Fourth Int. Jr. Conf. Pattern Recognit.* **1978**, 483.
- (14) Fairfield, J. Contoured shape generation forms that people see in dot patterns. *Proc. IEEE Conf. Cybernet Soc.* **1979**, 60.
- (15) Matula, D. W.; Sokal, R. R. Properties of Gabriel graphs relevant to geographic variation research and the clustering of points in the plane. *Geogr. Anal.* **1980**, *12*, 205.
- (16) Toussaint, G. T. The relative neighborhood graph of a finite planar set. *Pattern Recognit.* **1980**, *12*, 261.
- (17) Kirkpatrick, D. G.; Radke, J. D. A framework for computational morphology. In *Computational Geometry*; Toussaint, G. T., Ed.; Elsevier-North-Holland: New York, 1985; p 234.
- (18) Edelsbrunner, H.; Kirkpatrick, D. G.; Seidel, R. On the shape of a set of points in a plane. *IEEE Trans. Inf. Theory* **1983**, 551.
- (19) Edelsbrunner, H. *Weighted alpha shapes*; Tech. Rep. UIUCDCS-R-92-1760; Department of Comp. Sc., University of Illinois-Urbana-Champaign: Urbana-Champaign, IL, 1992.
- (20) Mandal, D. B.; Murthy, C. A. Selection of alpha for alpha-hull in  $\mathcal{R}^2$ . *Pattern Recognit.* **1997**, *30*, 1759.
- (21) Haines, E. Point in polygon strategies. In *Graphic Gems*; Heckbert, P., Ed.; Academic Press: New York, 1994; p 24.
- (22) DeJong, K. A. An analysis of the behavior of a class of genetic adaptive systems. Doctoral dissertation, University of Michigan, Ann Arbor, MI, 1975.
- (23) Edwards, K.; Edgar, T. F.; Manousiouthakis, V. I. Kinetic model reduction using genetic algorithm. *Comput. Chem. Eng.* **1998**, *22*, 239.
- (24) Goldberg, D. E. *Genetic algorithms in search, optimization and machine learning*; Addison-Wesley: New York, 1989.
- (25) Floudas, C. A.; Gumus, Z. H.; Ierapetritou, M. G. Global optimization in design under uncertainty: Feasibility test and flexibility index problems. *Ind. Eng. Chem. Res.* **2001**, *40*, 4267.
- (26) Vajda, S.; Valko, P.; Turanyi, T. Principal component analysis of kinetic models. *Int. J. Chem. Kinet.* **1985**, *17*, 55.

Received for review August 4, 2004

Revised manuscript received January 31, 2005

Accepted March 2, 2005

IE049294D