Parallel Hybrid Algorithm for Process Flexibility Analysis

Jeonghwa Moon, Kedar Kulkarni, Libin Zhang, and Andreas A. Linninger*  
Laboratory for Product and Process Design, Department of Chemical and Bioengineering, University of Illinois at Chicago, Chicago, Illinois 60607

Flexibility analysis is an important task for the optimal design and synthesis of chemical processes with uncertainty. It is a challenging problem because of the discontinuity and nonconvexity of rigorous flexibility programming formulations. In this article, we propose a new parallel hybrid algorithm based on stochastic search in conjunction with a nearest constraint projection technique to numerically solve the flexibility index problem. The proposed method can be applied regardless of the convexity of the design constraints. The stochastic method robustly identifies the global solution without the need for derivative information. The new nearest constraint projection technique is used to handle the constraints of the flexibility index problem in reduced state space. In contrast to existing methods, this technique does not require the addition of artificial variables for active constraints, does not need to have access to explicit analytical forms of the problem formulation or its derivatives, and does not solve for additional artificial variables. Its implementation is well-suited for parallel computing so that computational time can be dramatically reduced. Five applications illustrate the efficacy of the proposed method.

1. Introduction

Flexibility, operability, controllability, and reliability are important design objectives for the feasible and safe operation of chemical processes.1–3 Chemical plants should perform safely within acceptable quality ranges despite the presence of operational disturbances and inherent uncertainties of the process parameters. For realistic process designs, it is important to quantify the degree of flexibility to handle uncertainty in process parameters as well as variations in operating conditions.

Rigorous flexibility analysis can be formulated by mathematical programming methods such as in the feasibility test and the flexibility index problems.4–5 The feasibility test determines whether a given design specification performs feasibly over known ranges of uncertain parameters.6 The flexibility index problem is more general in that it seeks the maximum parameter deviation from the nominal conditions that a design can tolerate until it becomes infeasible.6,7 Several numerical approaches have been developed for solving the flexibility index problem. Swaney and Grossmann6 suggested a branch-and-bound algorithm that requires the assumption of critical points lying at a vertex. Grossmann and Floudas7 developed an active constraint set method to formulate the flexibility index problem as a mixed integer nonlinear programming (MINLP) problem. With few exceptions such as the αBB deterministic global optimization by Floudas et al.8 or Lucia et al.,9 previous flexibility solution efforts appear to deploy only local optimization techniques. In contrast to other design optimization problems in which locally optimal solutions might suffice, the solution to the flexibility index problem must be global to be rigorous. Therefore, local optimization is inadequate for flexibility analysis and global solution is a must.

Another important characteristic of the flexibility index problem is its inherent discontinuity and nondifferentiability. The flexibility search space violates basic assumptions required by many deterministic local optimizations. The uncertain 2D space (θ₁ and θ₂) shown in Figure 1 illustrates this characteristic of the flexibility index. Despite the continuity and differentiability of all constraints, the feasible deviation index, δ, plotted for rays emanating from the nominal conditions with angle α displays a discontinuous and nondifferentiable functional form. This nondifferentiable δ space is problematic for gradient-based search methods.

In light of these challenges, we propose a novel hybrid algorithm for flexibility analysis. This algorithm deploys a stochastic element to search the design space globally, while using local gradient information to explore a reduced state variable space. The novel nearest constraint projection technique is used to handle design constraints. It also permits the analysis of problems in which constraint derivatives are unavailable, such as large process models of legacy code or built with commercial black-box flowsheet simulators. It might also be useful in situations in which deterministic global algorithms are unsuitable. The proposed method also lends itself readily to parallelization, so that dramatic overall computational time reductions are possible.

This article is organized as follows: Section 2 presents framework of the proposed hybrid algorithm. Section 3 introduces a novel projection technique to handle design constraints. Section 4 demonstrates the efficiency of the proposed method in realistic applications. The article then closes with conclusions and future work.

2. Methodology

2.1. Mathematical Formulation of the Flexibility Index Problem. The scalar flexibility index, \(F\), is defined as the largest normalized deviation from nominal conditions that a design can tolerate while still ensuring feasible operation. Flexible process design aims at choosing design variables and controls so that the flexibility of the design is greatest. For a design to be called flexible, \(F\) should at least be greater than unity, which means that all expected deviations can be handled without violating any design constraints. The flexibility index, \(F\), is given by Floudas et al.\(^8\)

\[
F = \min \delta \\
\text{s.t. } \quad \psi(d, \theta) = 0 \\
\quad \psi(d, \theta) = \min u \\
\text{s.t. } \quad h_i(d, z, x, \theta) = 0, \quad i \in I \\
\quad g_j(d, z, x, \theta) \leq u, \quad j \in J \tag{1}
\]
where $d$ corresponds to the vector of design variables, $x$ is the vector of state variables, $z$ is the vector of controls, and $\theta$ is the vector of uncertain parameters. The equality constraints, $h_i$, enforce mass and energy balances and equilibrium relations; the inequality constraints, $g_j$, typically represent operating and design conditions. The scalar $\delta$ measures the infinity norm of the parameter deviation in the normalized uncertain space, as defined in the equation

$$
\delta = \max_j \left( \frac{\theta_j - \theta_j^N}{\Delta \theta_j} \right) = \left| \frac{\theta_j - \theta_j^N}{\Delta \theta_j} \right|, \quad j = 1, \ldots, p
$$

where $p$ is the number of uncertain parameters and $\theta_j^N$ and $\Delta \theta_j$ refer to the nominal value and the expected deviation of the uncertain parameter, $j$. The function $\psi(d, \theta) = 0$ defines the feasible region boundary in the space of uncertain parameters, as shown in Figure 2. The flexibility index problem aims at identifying the critical point, $\theta^c$, corresponding to the worst-case scenario on the feasible region boundary closest to the nominal conditions. Geometrically, the flexibility index corresponds to the dimension of the hyperrectangle just touching the feasible region boundary at the critical point.

2.2. Hybrid Algorithm. We have chosen a hybrid method combining the robustness of stochastic global search with the superior performance of Newton-type deterministic techniques. The main idea of the hybrid algorithm is to generate samples of uncertain parameter realizations located precisely on the feasible region boundary. To accomplish this task, we use a genetic search based on the principles of natural selection and inheritance. Candidate solutions generated by the stochastic search are then projected onto the boundary of the feasible region while rigorously satisfying all state equations. This nearest constraint projection technique ensures that all candidate solutions are placed exactly on the flexibility boundary. To satisfy equality constraints, we adopt gradient-based methods such as quasi-Newton methods. Because all candidate solutions of the evolving population are members of the feasible region boundary, the sample with the minimum infinity norm is the critical point. Thus, the proposed hybrid algorithm exhibits both stochastic and deterministic features.

The information flow of the hybrid algorithm is depicted in Figure 3. In the first step, an initial random sample population is generated in the space of the uncertain parameters, $\theta$. In the next step, each crude sample is projected onto the feasible region boundary, $\psi(d, \theta) = 0$. The details of this nearest projection technique are described in section 3. This step also deploys Newton-type methods to satisfy equality constraints corresponding to the reduced state space. The fitness of each candidate solution is inversely proportional to the infinity norm $\delta$. In the natural selection step, samples with superior fitness scores are chosen to produce competitive offspring for the next generation. Candidate solutions with low fitness are unlikely to
Figure 4. Schematic depiction of crude samples and the one-dimensional projection onto the feasible boundary, ψ(d,θ) = 0. The crude samples generated by genetic inheritance do not, in general, lie on the feasibility boundary. The one-dimensional projection brings the samples to the feasibility boundary.

reproduce and tend to disappear. Offspring, θ_{child}, are computed by combining the parameter values of the selected parents, θ_1 and θ_2, according to the arithmetic crossover formula with one random variable, α, given in eq 3

θ_{child} = (1 − α)θ_1 + αθ_2, 0 ≤ α ≤ 1

Mutations occurring with given likelihood change the states of a single candidate solution according to a random event drawn from the probability density function given in eq 4

θ_{n} = θ_n + σN(0,1)

Here, σ is the standard deviation of the normal distribution N(0, 1). Mutations are needed to counter clustering of the candidate solutions in the search space and help explore the entire search space uniformly. The overall algorithm terminates when the specified generation limit is reached. The critical point is the candidate solution with the highest fitness. Because all candidate solutions lie on the feasibility boundary, the minimal deviation marks the critical point.

3. Nearest Constraint Projection Technique

Our hybrid algorithm needs to maintain an evolving population of candidate solutions located precisely at the feasibility boundary. The crude samples generated by genetic inheritance do not, in general, lie of the flexibility boundary, as schematically depicted in Figure 4. Therefore, we project the parameter values of the crude samples onto the feasibility boundary with exact methods. This correction process, known as a repair procedure, is implemented efficiently as a one-dimensional directional search. The search direction is selected to be collinear to the direction of the uncertain parameter dimension, θ^*_n, corresponding to the specific coordinate marking the infinity norm. The required parameter corrections are implemented iteratively to ensure the precise location of the feasibility boundary while simultaneously satisfying all state equations. Nevertheless, a repair procedure in a single direction is merely a one-dimensional search requiring but a small computational effort. The projection is similar to the well-known line search strategies deployed in step-size-controlled Newton methods.

Illustration of a Nonlinear Two-Dimensional Flexibility Problem. The evolution of the hybrid mechanism is illustrated graphically with the help of an example with two uncertain parameters in system 5.

\[ F = \min \delta \]
\[ \text{s.t. } g_1(\theta) \text{ or } g_2(\theta) \text{ or } g_3(\theta) = 0 \]
\[ g_1 = 30(\theta - 5)^2 - 550 + 50\theta_2 \leq 0 \]
\[ g_2 = \theta_1^2 - 500 + 50\theta_2 \leq 0 \]
\[ g_3 = 15(\theta - 3)^2 - 50\theta_2 \leq 0 \]
\[ \theta^* - \delta\Delta\theta \leq \theta \leq \theta^* + \delta\Delta\theta \]
\[ \Delta\theta_1, \Delta\theta_2 = \pm 2 \]
\[ \theta^* = (4, 4) \]

Figure 5 shows snapshots of samples belonging to successive generations leading to the global solution. As the evolution progresses, each sample is projected onto the feasibility boundary delineated by the nearest constraint. We have observed that the computational effort for bringing the crude samples to the feasibility boundary diminishes as the solutions draw closer to a critical region. After 12 generations, the candidate solutions coalesce around to the critical point (5.7429, 2.2571). This example was solved repeatedly with different nominal points in each run. Table 1 and Figure 6 summarize the results and the performance parameters. We found that the flexibility index was correctly detected in all instances within reasonable CPU times.

3.1. Nearest Constraint Projection with Controls. The flexibility of a design problem can be dramatically enhanced with adjustable controls, z. To better understand the impact of controls on the feasibility function, ψ, consider the example in eqs 6 visualized in Figure 7 with two uncertain parameters, θ_1 and θ_2, and one control, z.

\[ g_1 = (\theta_1 - z)^2 + \theta_2^2 - 1 \leq 0 \]
\[ g_2 = z - 1 \leq 0 \]
\[ g_3 = -z \leq 0 \]

For a fixed z value, the feasible region, ψ(d,θ) ≤ 0, is a circle in θ space; the feasible region changes for different realizations of the control levels z, as depicted in Figure 7a. The feasible region ψ(d,θ) ≤ 0 is the unified gray region made possible by permissible control adjustments z. Thus, the enlarged area projected onto θ space for all permissible z levels is the feasible region shown in Figure 7b. In general, an explicit expression for the feasible region boundary with control, ψ(d,θ) = 0, is not available. Hence, we augment the nearest constraint projection technique to move infeasible samples to the feasible search space with additional control adjustments. This technique will be shown to explore both the space of uncertain parameters, θ, and the space of the controls, z.

With control degrees of freedom, projection onto the feasibility boundary becomes more challenging, because control, especially with feedback, adds arbitrary nonlinearity and complexity to the design constraints. Rather than insisting on rigorous projection, we propose an opportunistic zigzag search that is easy to implement and computationally inexpensive. Because the projection onto the feasibility boundary is performed for each crude sample of the population, the zigzag control projection need not be correct in every instance. As long as one sample of the genetic search yields the flexibility expansion achievable with adaptable control, the overall algorithm will proceed to the global solution from generation to generation in a statistical sense. Hence, projection with zigzag search needs to succeed merely in a statistical sense true to the stochastic nature of the proposed methodology.

3.2. Implementation of the Nearest Constraint Projection Technique. This section introduces the detailed implementation of the nearest constraint projection technique with