Improved Optimization Methods for the Multiobjective Design of Bioprocesses

ARTICLE *in* INDUSTRIAL & ENGINEERING CHEMISTRY RESEARCH · SEPTEMBER 2006

Impact Factor: 2.59 · DOI: 10.1021/ie0605433

4 AUTHORS:

José-Oscar Hernández Sendín
Spanish National Research Council
6 PUBLICATIONS **94** CITATIONS
SEE PROFILE

Irene Otero-Muras
Spanish National Research Council
26 PUBLICATIONS **121** CITATIONS
SEE PROFILE

Antonio A. Alonso
École Polytechnique Fédérale de Lausanne
156 PUBLICATIONS **1,988** CITATIONS
SEE PROFILE

Julio R. Banga
Spanish National Research Council
224 PUBLICATIONS **4,102** CITATIONS
SEE PROFILE

Improved Optimization Methods for the Multiobjective Design of Bioprocesses

José-Oscar H. Sendín,[†] Irene Otero-Muras,[‡] Antonio A. Alonso,[§] and Julio R. Banga*

Process Engineering Group (IIM-CSIC), Spanish Council for Scientific Research, C/Eduardo Cabello 6, 36208 Vigo, Spain

In this work, we consider the multiobjective design of bioprocesses, i.e., those from the biotechnological, pharmaceutical, and food industries. We present and compare extensions for three solution strategies for multiobjective optimization of this class of problems, which can be a very challenging task due to the frequent nonconvex nature of the associated nonlinear programming problems (NLPs), a consequence of the highly nonlinear character of most bioprocess models. Details of these new implementations are provided, especially focusing on several modifications of the original methods which we have developed in order to improve their efficiency and robustness. In addition, bifurcation analysis will be used to provide the decision maker with additional stability criteria to further select a suitable compromise solution among the set of optimal design alternatives. As a case study, we consider the optimal design of a fermentation process with respect to two economic criteria.

Introduction

Traditionally, the design of continuous processes operated at steady state has been carried out by considering the optimization of a single performance index subject to nonlinear constraints. Typically, the objective function is some economic measure, such as the profit or the capital investment. However, for most real world applications, the existence of multiple criteria, often conflicting, which must be optimized simultaneously (i.e., multiobjective optimization), is quite common. Although this is a more realistic and desirable approach, in general there is no unique solution but a set of optimal tradeoffs among the different criteria, i.e., the set of *Pareto-optimal* points. In the case of bioprocesses, computing this set of optimal solutions can be a very challenging task due to the frequent nonconvex nature of the associated nonlinear programming problems (NLPs), which is in turn a consequence of the highly nonlinear character of most bioprocessing models.[1]

Considering only economic issues in the design stage may lead to economically optimal steady states which are, however, either unstable or lying in critical regions where a loss of stability can occur by slight perturbations. Bifurcation analysis via continuation methods is an established tool to characterize the qualitative nonlinear behavior of chemical and biochemical systems with respect to the model parameters. Numerical bifurcation tools have been traditionally used for analyzing the behavior of the system in the vicinity of the optimum[2] and, more recently, have been applied in the synthesis problem to a priori establish the stability boundary region used in the optimization step.[3]

The purpose of this work is threefold. In the first place, the performance of several single objective optimization methods, both of local and global nature, is evaluated within the framework of multiobjective optimization (MO). Second, we present extensions for three well-known solution strategies for MO problems. Details of the implementations are provided, especially focusing on several modifications of the original methods which have been developed in order to improve their efficiency and robustness. Finally, the use of bifurcation analysis as a tool to provide additional decision criteria based on the stability of the designs is illustrated. These new methods are compared by solving a challenging case study concerning the optimal design of a fermentation process.[2]

Problem Statement and Basic Concepts

Assuming a biochemical system operating at steady state, the multiobjective optimization design problem (MOP) can be mathematically stated, without loss of generality, as

$$\min_{\mathbf{x}} \mathbf{J}(\mathbf{x}) = \begin{pmatrix} J_1(\mathbf{x}) \\ J_2(\mathbf{x}) \\ \vdots \\ J_m(\mathbf{x}) \end{pmatrix}$$

$$\text{subject to}$$

$$\mathbf{h}(\mathbf{x}) = 0$$

$$\mathbf{g}(\mathbf{x}) \leq 0$$

$$\mathbf{x}^{\mathrm{L}} \leq \mathbf{x} \leq \mathbf{x}^{\mathrm{U}} \qquad (1)$$

where $\mathbf{x} \in \mathscr{R}^n$ is a vector of n decision variables; $\mathbf{J} \in \mathscr{R}^m$ is a vector of m objective functions or criteria $J_i(\mathbf{x})$: $\mathscr{R}^n \rightarrow \mathscr{R}$; $\mathbf{h}$ and $\mathbf{g}$ are possible sets of equality and/or inequality constraints, respectively, representing the process model and additional requirements for the system; and $\mathbf{x}^{\mathrm{L}}$ and $\mathbf{x}^{\mathrm{U}}$ are the lower and upper bounds for the decision variables. This set of constraints defines the *feasible design space* $\mathbf{S} \in \mathscr{R}^n$, while the *feasible objective space* $\mathbf{Z}$ is the set $\{\mathbf{J}(\mathbf{x})|\mathbf{x} \in \mathbf{S}\}$.

Very often the objective functions are in conflict with each other. In this case, it will not be possible to find a unique solution which is simultaneously optimal for all the objectives. It is clear that if there exists such a point, it will provide a solution to the MOP. Unlike single objective optimization, in a MOP, there will be in general multiple points which are optimal in the sense that an improvement in one objective can only be achieved with a worsening of one or more of the others. These optimal solutions, known as Pareto-optimal, are formally defined as follows:

* To whom correspondence should be addressed. E-mail: julio@iim.csic.es. Phone: +34 986 214 473. Fax: +34 986 292 762.
[†] E-mail: osendin@iim.csic.es.
[‡] E-mail: ireneotero@iim.csic.es.
[§] E-mail: antonio@iim.csic.es.

10.1021/ie0605433 CCC: $33.50
Published on Web 09/02/2006

A solution $\mathbf{x}^* \in \mathbf{S}$ is said to be a *Pareto-optimal* (or *efficient*) solution if and only if there is no $\mathbf{x}$ such that $J_i(\mathbf{x}) \leq J_i(\mathbf{x}^*)$, for all $i = 1, ..., m$, with at least one strict inequality. The vector $\mathbf{J}(\mathbf{x}^*)$ is said to be *nondominated*.

A related concept is that of a *weakly* Pareto-optimal point. A solution $\mathbf{x}^* \in \mathbf{S}$ is *weakly Pareto-optimal* if there does not exist another solution $\mathbf{x}$ such that $J_i(\mathbf{x}) < J_i(\mathbf{x}^*)$, for all $i = 1, ..., m$.

The set of all Pareto-optimal solutions is usually referred as the *Pareto front*. In the absence of any further information about the problem, no solution can be said to be better than another and, ideally, the entire Pareto-optimal set should be found.

When solving a MOP, other points of interest are the *utopia* vector and the *nadir* vector:

- The utopia vector $\mathbf{J}^*$ is the vector of objective functions containing the individual global minima of the objectives, i.e., for each $i = 1, ..., m$, $J_i^* = \min_{\mathbf{x}} \{J_i(\mathbf{x}) | \mathbf{x} \in \mathbf{S}\}$.
- The nadir vector $\mathbf{J}^{\text{Nadir}}$ is the vector of objective functions containing the upper bounds of the objectives in the Pareto-optimal set. The ith component of the nadir vector can be estimated as $J_i^{\text{Nadir}} = \max \{J_i(\mathbf{x}_j^*)\}$, where $\mathbf{x}_j^*$, $j = 1, ..., m$, is the global minimizer (or *anchor* point) of the jth objective.

Multiobjective Optimization Methods

During the past decades, many methods have been proposed to deal with multiobjective optimization problems. Full reviews can be found in the books by Miettinen[4] and Deb,[5] and the references therein. Traditionally, MOPs are solved by scalarization,[4] e.g., by means of a weighted sum of the objectives. Using some characteristic parameters, the original MOP is transformed into a single objective optimization problem whose solution is expected to be Pareto-optimal. The parameters can either represent the relative importance of the objectives or be a mere mathematical device which is varied systematically to obtain different solutions.

It is important to note that, in a mathematical sense, all Pareto-optimal solutions (potentially an infinite number) are equivalent and it cannot be said that one solution is better than another. From a practical point of view, however, the user is only interested in one final solution. An additional element in comparison to the single objective optimization is that of the decision maker (DM), who is responsible for selecting such a solution. In general, the DM will assign preferences to the objectives using some additional information which sometimes is subjective and/or difficult to express in mathematical terms.

In this work, we focus on methods that are able to produce a set of Pareto-optimal solutions which can be readily used by the DM to choose a suitable compromise. In theory, this final solution will represent more accurately the preferences of the DM but it is crucial to generate a good distribution of points capturing the complete tradeoff among the objectives.

Scalarization techniques require solving repeatedly a set of single nonlinear programming (NLP) problems, but very often, it is not obvious how to change the parameters of the method in order to obtain a satisfactory solution or a good distribution of points. In contrast, other methods such as normal boundary intersection[6] and normalized normal constraint[7] are able to produce an even spread of points on the Pareto front with a systematic change in their parameters. In the field of bioprocess engineering, computing the Pareto-optimal set can be a very challenging task due to the highly constrained and nonlinear nature of most biochemical systems. In this regard, it is important to keep in mind that the majority of the existing implementations ultimately rely on local, gradient-based, optimization routines (e.g., sequential quadratic programming (SQP)) for solving the NLPs, so they can fail if the MOP is nonconvex; the solution can only be guaranteed to be *local* Pareto-optimal. This drawback can be addressed by using suitable global optimization (GO) methods.

A different approach to solve MOPs is based on the use of evolutionary algorithms, in which there has been an increasing interest in the past decade.[8] These methods mimic the mechanisms of natural selection and genetics by using a *population* of possible solutions. Thus, they can find simultaneously multiple nondominated points in one single optimization run. This fact, together with the ability of these multiobjective evolutionary algorithms (MOEAs) to deal with problems involving nonconvex Pareto fronts, makes them attractive to solve highly nonlinear MOPs. It should be noted that single objective evolutionary algorithms can also be used in combination with scalarization techniques.

As a drawback, MOEAs usually require a large population size which is translated into a large number of nondominated points. In addition to the associated increase in computational effort, such a large set can be very difficult to handle, especially as the number of objectives increases.

In the following sections, we present a description of the techniques applied in this work (two scalarization techniques and a well-known genetic algorithm) and the single objective optimization methods used to solve the associated NLPs, along with the details of the implementation and the suggested modifications.

Normal Boundary Intersection (NBI). The NBI[6] method essentially works by solving a set of NLPs of the following form:

$$\begin{aligned} &\max_{\mathbf{x},\lambda} \ \lambda \\ &\text{subject to} \\ &\mathbf{x} \in \mathbf{S} \\ &\mathbf{\Phi w} + \lambda \mathbf{n} = \mathbf{J}(\mathbf{x}) - \mathbf{J}^* \end{aligned} \tag{2}$$

$\mathbf{\Phi}$ is an $m \times m$ *payoff matrix* in which the ith column is $\mathbf{J}(\mathbf{x}_i^*) - \mathbf{J}^*$ (the objective functions are shifted to the origin), $\mathbf{x}_i^*$ being the respective global minimizer of the ith objective; $\mathbf{w}$ is a vector of weights such that $\sum_{i=1}^{m} w_i = 1$ and $w_i \geq 0$; $\mathbf{\Phi w}$ defines a point in the so-called *convex hull of individual minima* (CHIM), i.e., the set of points that are convex combinations of the individual minima; $\mathbf{n}$ is the unit normal to the CHIM (in practice, $\mathbf{n}$ is the *quasi-normal* vector defined as $-\mathbf{\Phi e}$, where $\mathbf{e}$ is a column vector of ones); λ is the distance along the normal to the CHIM pointing toward the origin. The philosophy of NBI is that the intersection between the normal from any point on the CHIM and the boundary of the objective space closest to the origin is expected to be Pareto-optimal.

The above NLP (called the NBI subproblem) is solved for various values of $\mathbf{w}$, in such a way that an equally distributed set of weights produces an even spread of points on the CHIM, which in turn yields an even spread of solutions in the boundary of the objective space. The method requires a user-supplied parameter, *spac*, whose inverse is the uniform spacing between two consecutive weights. It should be noted that the equality constraints introduced by NBI ensure that the solution is actually on the normal to the CHIM. As a drawback, this method can also yield non-Pareto-optimal points.

Normalized Normal Constraint (NNC). The main idea of NNC[7] is similar to that of NBI, but it differs in that NNC uses inequality constraints for reducing the feasible objective space. The utopia vector and the nadir vector are used to normalize

the objectives, and an equally distributed set of points is generated in the so-called *utopia hyperplane* (i.e., the CHIM in the NBI method). For each one of these points, the following NLP is solved:

$$\min_{\mathbf{x}} \bar{J}_k(\mathbf{x}) \quad k \in \{1, ..., m\}$$

$$\text{subject to}$$

$$\mathbf{x} \in \mathbf{S}$$

$$\bar{\mathbf{N}}_i(\bar{\mathbf{J}}(\mathbf{x}) - \mathbf{X}_P) \leq 0 \quad i = 1, ..., m;\ i \neq k$$

$$\bar{\mathbf{N}}_i = \bar{\mathbf{J}}(\mathbf{x}_k^*) - \bar{\mathbf{J}}(\mathbf{x}_i^*) \quad i = 1, ..., m;\ i \neq k \tag{3}$$

where $\mathbf{X}_P$ is a point in the utopia hyperplane.

Nondominated Sorting Genetic Algorithm (NSGA-II). NSGA-II[9] (or its precursor NSGA[10]) is one of the most used algorithms in chemical engineering.[11] It exploits a ranking selection method based on nondomination levels to emphasize the good solutions, and a crowded comparison operator is used to maintain diversity in the population. For details, see ref 9.

Single Objective Optimization Solvers

Both the NBI and NNC strategies mentioned above ultimately require the solution of a set of single objective nonlinear programming problems (NLPs), so the use of robust and efficient NLP solvers becomes a critical issue. Further, in the particular case of highly nonlinear constraints, as it is usually the situation in bioprocess engineering, one should be particularly worried about the likely nonconvexity of these NLPs. Here, we consider a standard single objective local optimization solver and two global methods which have been found to be efficient and robust even for large nonlinear problems:[1,12]

- FMINCON: This code is part of the MATLAB optimization toolbox (The Mathworks, Inc, MA) and uses sequential quadratic programming (SQP). Thus, only local minima can be obtained.
- GLOBAL:[13] This is a hybrid GO method consisting of a random search (sampling step) followed by a local search routine. Initially, it carries out a clustering phase by means of the single linkage method. Next, a robust random walk direct search method, UNIRANDI,[14] is applied using the best points from the sampling step which have not been clustered.
- SRES:[15] This is a (μ,λ)-evolution strategy with the ability to solve global optimization problems and a rather good handling of arbitrary constraints. A population of size λ is sorted according to the so-called *stochastic ranking* scheme, in which there exists a balance between preserving feasible individuals and rejecting infeasible ones. The best μ solutions are then selected, and the individuals for the next generation are created through mutation. One of the biggest advantages of evolution strategies, like SRES, is that they have been designed with self-tuning properties in mind; thus, they are quite independent from the required values of the tuning parameters.

Implementation Details

Multiobjective Optimization Methods. The three multiobjective optimization methods (NBI, NNC, and NSGA-II) considered in this work were implemented in MATLAB (version 6.5). The main reason for this is that it provides a convenient environment to postprocess and visualize the results with little programming effort.

In the case of NBI, an old version of this technique is available on the Internet (www.caam.rice.edu/~indra/NBIhomepage.html). The code has been rewritten in order to use the above single objective optimization solvers. The original implementation solves the NLPs very efficiently using, as an initial guess, the solution of a "nearby" subproblem.

An important issue in NBI is how to handle the equality constraints introduced by the method. FMINCON can handle them directly, but this is not the case for the other solvers. In this case, the equality constraints are added to the objective function as a quadratic penalty function. Thus, the NBI subproblem takes the following form:

$$\max_{\mathbf{x},\lambda} \lambda - R_P \sum (\bar{\mathbf{h}}_{NBI})^2 \tag{4}$$

where R_P is a penalty coefficient and $\bar{\mathbf{h}}_{NBI}$ are the following normalized NBI-equality constraints:

$$\bar{\mathbf{h}}_{NBI} = \lambda - \frac{(\mathbf{J}(\mathbf{x}) - \mathbf{J}^* - \mathbf{\Phi w})}{\mathbf{n}} \tag{5}$$

It should be noted that the division in eq 5 is performed element-by-element. Since the objective functions may have different scales, writing the NBI constraints like eq 5 means that violations for all the equalities take more or less the same order of magnitude, so that only one single penalty factor is needed.

NNC has been implemented following a similar strategy to that of NBI for solving the associated NLPs (i.e., taking as an initial guess the solution of the previous subproblem).

Finally, NSGA-II has been combined with a *k*-means clustering algorithm[16] to reduce the nondominated set. As mentioned before, the outcome of a MOEA can be a very large set of nondominated points. Selecting a final solution from such a set can be a very intimidating task and, in general, the DM will be interested in a "small" number of optimal alternatives capturing the entire Pareto front.

Single Objective Optimization Solvers. In the case of SRES, we have relied on an existing Matlab implementation (which is available at http://cerium.raunvis.hi.is/~tpr/software/sres/index.html) to create a customized version with the following modifications.

- User-supplied points in the initial population: Within the NBI and NNC frameworks, these points are the solutions of the subproblems that have already been solved (alternatively, the rest of the individuals in the initial population can be created by using a normal distribution around the solution of a nearby subproblem).
- Alternative termination criterion: Originally, SRES terminates the optimization when the maximum number of generations defined by the user is reached. In our implementation, the optimization will be also terminated when the difference between the mean of the best objective values of the last 15 generations and the best value found so far is below a certain tolerance.[17] In this way, we do not have to invest a significant effort in the adjustment of the tuning parameters for each one of the NLPs.
- Out of bounds saturation: During the search, if any variable is out of bounds, it is set to the value of the corresponding bound.

Regarding GLOBAL, we have made a Matlab translation from the original FORTRAN code.[13] The solutions of the subproblems that have been already solved are also included in the initial random search step. In addition, handling of inequality constraints has been added by means of a quadratic penalty function or by rejecting infeasible points (i.e., dead-penalty).

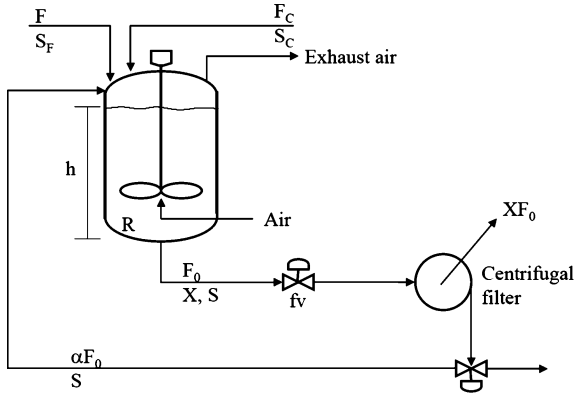

Figure 1. Flowsheet of the fermentation process.

Bifurcation Analysis. We have made use of the package CL_MATCONT,[18] which implements a continuation method based on the More−Penrose matrix pseudo-inverse. Starting from a stationary solution of the model under study, a sequence of points is generated that corresponds to the equilibria of the ordinary differential equations system representing the model. CL_MATCONT enables us both to detect singularities and bifurcation points in the solution path and to calculate new branches of the solutions starting from the bifurcation points. This package is compatible with the standard Matlab ode representation of differential equations.

Case Study: Multiobjective Optimal Design of a Biochemical Reactor

This case study is the fermentation process design problem considered by Brengel and Seider[2] and Banga and Seider,[19] among others. The process consists of a well-stirred, aerobic fermentor in which *Saccharomyces cerevisiae* grows in a medium of sugar cane molasses, and a centrifuge is used for the recovery of cell mass (Figure 1). The fermentor outlet stream contains biomass and substrate in concentrations X and S, respectively. After centrifugation, a fraction α of the centrifuge effluent is recycled to the fermentor. A diluted feed stream (with a substrate concentration $S_F = 0.3$ kmol·m^{-3}) and a concentrated one (with $S_C = 1.0$ kmol·m^{-3}) are available.

Assuming steady-state operation, the above-mentioned authors solved a single objective optimization problem in which the *venture profit* is maximized. This cost function is composed of two terms: the product profit P for the biomass and the fixed capital investment (FCI). To illustrate the performance of the methods described above, in this work, we have first considered a MOP which is derived from the original cost function problem. Thus, the MOP is formulated to maximize the product profit P and minimize simultaneously the fixed capital investment FCI by adjusting seven design variables:

$$\min_{h,X,S,\alpha,F_C,F,f_v} \times \begin{pmatrix} J_1 = -P(\$10^7/\text{yr}) = -504 \times 10^{-4}(c_1XF_0 - c_2F - c_3F_C - c_4XF_0/3600) \\ J_2 = \text{FCI}(\$10^7) = 1.18 \times 10^7(F_{bm,v}B_vV^{0.724} + F_{bm,c}B_C(XF_0)^{0.444}) \end{pmatrix} \quad (6)$$

subject to

$$\frac{F + F_C - (1-\alpha)f_v h^{0.5}}{\pi R^2} = 0 \quad (7)$$

$$\frac{-X(F + F_C + \alpha f_v h^{0.5})}{\pi R^2 h} + \mu(S)X = 0 \quad (8)$$

$$\frac{F(S_F - S) + F_C(S_C - S)}{\pi R^2 h} - \frac{\mu(S)X}{Y(S)} = 0 \quad (9)$$

$$f_v h^{0.5} X \geq Q_{min} \quad (10)$$

$$1.5R \leq h \leq 4.5R \quad (11)$$

$$\alpha \geq 0 \quad (12)$$

where F and S_F are the flow rate and substrate concentration of the feed (F_C and S_C of the concentrated feed); h, R, and V are the height of the liquid holdup, radius, and volume of the fermentor; f_v is the valve constant; c_1, ..., c_4 are cost coefficients; B_V, B_C, $F_{bm,V}$, and $F_{bm,C}$ are base cost and bare module factors for the fermentor and the centrifuge, respectively; and $\mu(S)$ and $Y(S)$ are the specific growth rate and the biomass yield coefficient, respectively, which are given by[20]

$$\mu(S) = kSe^{-S/K} \quad (13)$$

$$Y(S) = a + bS \quad (14)$$

Values for the parameters of the model can be found in Brengel & Seider.[2] The equality constraints 7−9 are the mass balances at steady state, and the inequality constraints consist of several design specifications, with Q_{min} being the minimum capacity of production for the system (fixed arbitrarily at 1.0 kg·min^{-1}). The occurrence matrix of the equality constraints shows that they can be solved sequentially taking h, F_C, F, and S as decision variables. Their respective lower and upper bounds are presented in Table 1.

Table 1. Lower and Upper Bounds on the Decision Variables

design variable	lower bound	upper bound
h	3.3750	10.125
F_C	0.0000	10.000
F	0.0000	10.000
S	0.0000	1.0000

It has been shown that complex operation regimes with periodic behavior or multiple steady states can appear in this process.[2] We will study the dynamic behavior of each one of the design alternatives provided by the multiobjective approach. Considering the reactor alone, and denoting by F_e and S_e the effective inlet flowrate and the effective inlet substrate concentration, respectively, the mass balances can be expressed in terms of the dimensionless concentrations of cell mass, $C_1 = X/(S_eY\{S_e\})$, and substrate conversion, $C_2 = (S_e - S)/S_e$:

$$\frac{dC_1}{d\tau} = -C_1 + Da(1 - C_2)e^{C_2/\gamma}$$

$$\frac{dC_2}{d\tau} = -C_2 + Da\left(\frac{\beta + 1}{\beta + 1 - C_2}\right)(1 - C_2)C_1e^{C_2/\gamma} \quad (15)$$

where $Da = \mu\{S_e\}(F_e/V)$ is the Damkholer number (i.e., the ratio of the specific growth rate in the bioreactor to the dilution rate); $\beta = a/(bS_e)$ and $\gamma = K/S_e$ are dimensionless parameters related to the kinetics and the initial substrate concentration.

Table 2. Tuning Parameters Utilized in the Algorithms

FMINCON	SRES	GLOBAL	NSGA-II
tolerances = 10^{-6}	population size = 100	N_{sampl} = 500	population size = 100
max iterations = 400	generations (max.) = 200	N_{sel} = 5; N_{sig} = 6	generations = 1000
max. function evaluations = 150 × (no. of variables)	termination criterion = 10^{-4}	max. function evaluation = 10000	crossover prob. = 0.9
	parent number = 15		mutation prob. = 0.25
	P_f = 0.45		

Table 3. Individual Optima of the Objectives for the Biobjective Optimization Problem

	min FCI (weakly Pareto)	min FCI	max P (local)	max P (global)
h (m)	3.37500	3.37500	10.1250	10.1250
F_C ($m^3 \cdot min^{-1}$)	0.167815	0.00281245	0.00000	3.75705
F ($m^3 \cdot min^{-1}$)	0.00000	0.191130	7.08505	0.00000
S ($kmol \cdot m^{-3}$)	0.00321137	0.156817	0.13009	0.31067
X ($kg \cdot m^{-3}$)	5.95859	0.438892	4.89616	42.2707
f_v ($m^{2.5} \cdot min^{-1}$)	0.0913523	1.24024	2.22662	1.18073
α	0.00000	0.91488	0.00000	0.00000
P ($\$10^7 \cdot y^{-1}$)	−1.01668	0.12670	4.88521	11.6173
FCI ($\$10^7$)	0.0392314	0.0392314	0.14387	0.24543

Results and Discussion

The multiobjective optimization process is composed of three steps:

(1) Search for the global individual minima of the objectives (required by the NBI and NNC methods). These points will provide a first insight into the tradeoff involved among the criteria.

(2) Generate a set of Pareto-optimal solutions.

(3) Analyze the solutions and select one.

The tuning parameters of the optimization solvers are listed in Table 2. In the majority of the cases, we have used the recommended values, together with some tuning guided by the feedback obtained after a few preliminary runs.

Utopia Vector. First of all, each of the objectives is minimized individually subject to the constraints. For the maximization of the product profit P (i.e., the minimization of $-P$), a multistart SQP was used in the first place with a set of 50 randomly generated initial points. Here, 80% of the runs converged to the global maximum (P = 11.6173 $\$10^7 \cdot y^{-1}$). A local solution was also found, with P = 4.8852 $\$10^7 \cdot y^{-1}$. These results, which highlight the nonconvexity of the problem, were confirmed with SRES and GLOBAL. An interesting feature of GLOBAL is that the output of this solver contains all the local optima found during the search. In this case, GLOBAL located both maxima for the product profit in the majority of runs.

The minimization of FCI results in a weakly Pareto-optimal solution, i.e., there exist multiple designs with the same value of FCI (3.9231 $\$10^5$) and values of the product profit between −1.0167 and 0.1267 $\$10^7 \cdot y^{-1}$. Here, it is worth mentioning that a lexicographic approach could have been used to find the Pareto-optimal solution corresponding to the minimum of the FCI. The extreme solutions of the Pareto-optimal set are presented in Table 3.

Results with NBI. NBI was run with all the single objective solvers using an evenly spread set of weights in which the uniform spacing between two consecutive w_i is 1/16 (resulting in 15 NLPs). Using FMINCON, results were different depending on the order in which these subproblems were solved. Starting at $\mathbf{x}_2^*$ (i.e., the minimizer of the FCI), the local solver did not converge to a solution in four NBI subproblems. On the other hand, if the method was started at $\mathbf{x}_1^*$, several solutions were found to be at a local Pareto front. In principle, it can be expected that refining the discretization of the Pareto front (that is, decreasing the uniform spacing between weights) improves the performance of the local solver (the initial guess would be closer to the solution). We have run NBI with a uniform spacing of 1/100, but the same difficulties were observed. However, the combination of both sets of solutions provides a true representation of the Pareto-optimal set, as confirmed later by the global solvers. Points obtained are plotted in the objective space in Figure 2, where the solid line represents the global Pareto front, whereas the dotted one is the local Pareto-optimal set. From the inspection of these results, the existence of a discontinuity located at the local maximum for the profit is clear.

It should be noted that, although global Pareto-optimal solutions could be obtained with FMINCON for this case following the above procedure, more complex systems may cause difficulties to local gradient-based methods, as shown in a previous work;[21] thus, global single optimization methods will be needed.

Twenty independent runs of NBI−SRES and NBI−GLOBAL were carried out with a penalty factor of 1.0 for the NBI-equality constraints. The experimental results for each weight vector are presented in Table 4, which shows the statistics for the 20 runs (best objective value found, mean, and standard deviation), together with the total computational effort required by the method (in terms of the CPU time and the number of function evaluations). Both solvers yielded global Pareto-optimal solutions, which are similar to those shown in Figure 2, but in general, GLOBAL produced better results than SRES. The main difficulties were found in the NLPs whose solutions are located close to the discontinuity, especially for the weight vectors $\mathbf{w}$ = [0.4375, 0.5625] and $\mathbf{w}$ = [0.3750, 0.6250]. For these NLPs, SRES converged to local solutions in 50% and 20% of the runs. We carried out another set of experiments in which the initial population in SRES was generated using a normal distribution around the previous solution, but no significant improvement was observed. In terms of the number of function evaluations, the performance of both solvers was similar.

There are slight differences between the solutions found with FMINCON and those obtained with the global solvers. The latter depended on the value of the penalty factor used in the handling of the NBI-equality constraints. However, these equalities can

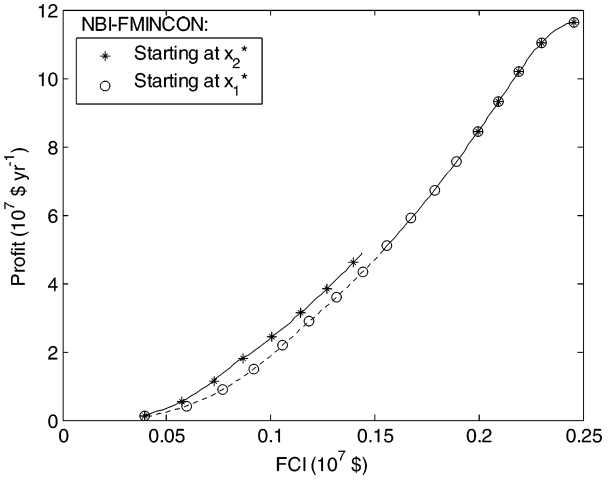

Figure 2. Points in the objective space obtained using NBI−FMINCON.

Table 4. Results with NBI Using SRES and GLOBAL

		SRES			GLOBAL		
w_1	w_2	best value	mean	std. dev.	best value	mean	std. dev.
0.9375	0.0625	−0.258942	−0.258746	0.000245	−0.258811	−0.256774	0.001662
0.8750	0.1250	−0.146169	−0.140898	0.009287	−0.146056	−0.143566	0.003734
0.8125	0.1875	0.023302	0.028454	0.009799	0.023377	0.024481	0.001516
0.7500	0.2500	0.185208	0.191806	0.012704	0.185318	0.189327	0.005003
0.6875	0.3125	0.329831	0.332372	0.006465	0.329887	0.336878	0.010656
0.6250	0.3750	0.455289	0.457484	0.004188	0.455383	0.466642	0.021910
0.5625	0.4375	0.559389	0.561403	0.004018	0.559693	0.576966	0.016647
0.5000	0.5000	0.639491	0.640730	0.002019	0.639676	0.653122	0.016311
0.4375	0.5625	0.430211	0.574945	0.134280	0.430206	0.445566	0.058962
0.3750	0.6250	0.464144	0.514413	0.103004	0.464146	0.478909	0.063205
0.3125	0.6875	0.466208	0.466271	0.000059	0.466265	0.466653	0.000277
0.2500	0.7500	0.430316	0.430361	0.000038	0.430309	0.430650	0.000300
0.1875	0.8125	0.348021	0.348054	0.000023	0.348029	0.348203	0.000154
0.1250	0.8750	0.296532	0.296545	0.000011	0.296530	0.296532	0.000003
0.0625	0.9375	0.157377	0.157395	0.000016	0.157376	0.157382	0.000009
CPU time (s)		43	47		73	91	
function evaluations		122500	131810		111333	134467	

Table 5. Results with NNC Using SRES and GLOBAL

		SRES			GLOBAL		
w_1	w_2	best value	mean	std. dev.	best value	mean	std. dev.
0.9375	0.0625	0.926071	0.927008	0.001069	0.925398	0.926122	0.001509
0.8750	0.1250	0.873796	0.877095	0.011744	0.873283	0.873376	0.000098
0.8125	0.1875	0.826037	0.828759	0.010610	0.825642	0.825724	0.000117
0.7500	0.2500	0.777521	0.778313	0.000666	0.777097	0.777532	0.000662
0.6875	0.3125	0.727484	0.732881	0.014894	0.727047	0.727474	0.000579
0.6250	0.3750	0.675818	0.676355	0.000562	0.675328	0.675668	0.000635
0.5625	0.4375	0.622314	0.624414	0.008170	0.621767	0.622950	0.001590
0.5000	0.5000	0.566644	0.569651	0.010232	0.566106	0.567657	0.001904
0.4375	0.5625	0.485894	0.496281	0.011205	0.485313	0.491853	0.011312
0.3750	0.6250	0.426284	0.429889	0.008695	0.425678	0.428939	0.010053
0.3125	0.6875	0.363959	0.366128	0.006512	0.363295	0.363571	0.001149
0.2500	0.7500	0.298398	0.299603	0.005267	0.297660	0.297666	0.000006
0.1875	0.8125	0.228918	0.228958	0.000030	0.228079	0.228114	0.000113
0.1250	0.8750	0.161774	0.161790	0.000015	0.161027	0.161028	0.000001
0.0625	0.9375	0.088027	0.088042	0.000010	0.086992	0.086996	0.000007
CPU time (s)		114	147		79	100	
function evaluations		188900	205085		85898	105593	

be seen as "soft" constraints in the sense that only a solution sufficiently "close" to the normal direction is needed. Figure 3 shows the set of points obtained using NBI−GLOBAL and the normal directions to the CHIM corresponding to the considered set of weights. Although there are points which are not feasible with respect to the NBI-equality constraints, it can be observed that the global solutions of the penalized NBI subproblems lie on the global Pareto-optimal front and maintain a good distribution. Our experiments indicate that, due to the normalization of the NBI-equality constraints (eq 5), similar distributions are obtained with different values of the penalty coefficient.

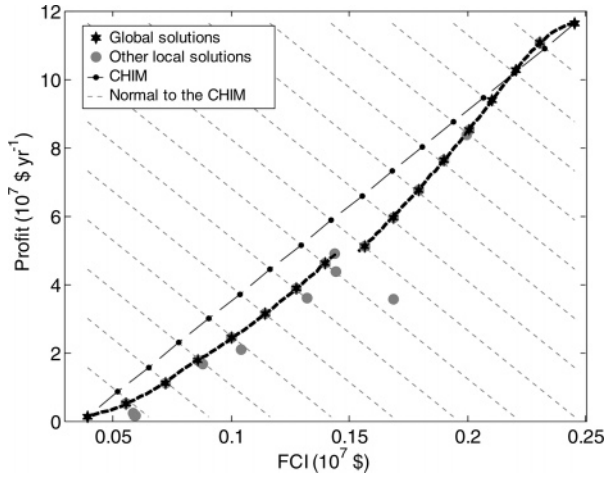

Figure 3. Points in the objective space obtained using NBI−GLOBAL (the dashed lines represent the discontinuous global Pareto optimal front).

Figure 3 shows all the local minima of the penalized NBI subproblems located by GLOBAL. This interesting feature can be useful to provide a better representation of the boundary of the objective space. In this regard, the local maximum of the profit ($P = 4.8852$ \$$10^7$·y^{-1}) was obtained in the majority of runs as part of the global Pareto-optimal set. This point, which defines the discontinuity in the Pareto front, may be difficult to detect using NBI or NNC.

Results with NNC. The NNC method was run with the same set of weights used in NBI. The objective to minimize is the FCI, subject to the additional constraints defined by eq 3. The search is started at $\mathbf{x}_1^*$, so that the initial points used in each subproblem are always feasible. Results with FMINCON were similar to those obtained with NBI−FMINCON when the search was started at $\mathbf{x}_1^*$, i.e., part of the solution set is located at the local Pareto front. SRES and GLOBAL performed quite well (Table 5), resulting in similar solutions as those obtained using NBI. As in the NBI approach, some difficulties were also found in the vicinity of the discontinuity. In terms of computational effort, NNC−GLOBAL is clearly superior to NNC−SRES. In about 30% of the runs, SRES converged to local Pareto-optimal solutions for the NLPs corresponding to the weight vectors $\mathbf{w} = [0.4375, 0.5625]$ and $\mathbf{w} = [0.3750, 0.6250]$. GLOBAL also reported many false local minima which are located along the normal directions to the utopia hyperplane.

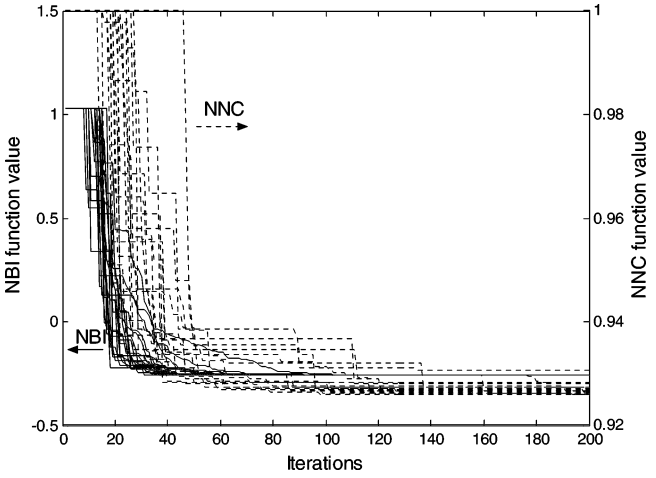

Figure 4. SRES convergence curves for the NLP corresponding to $\mathbf{w} = [0.9375, 0.0625]$ in NBI (solid lines) and NNC (dotted lines).

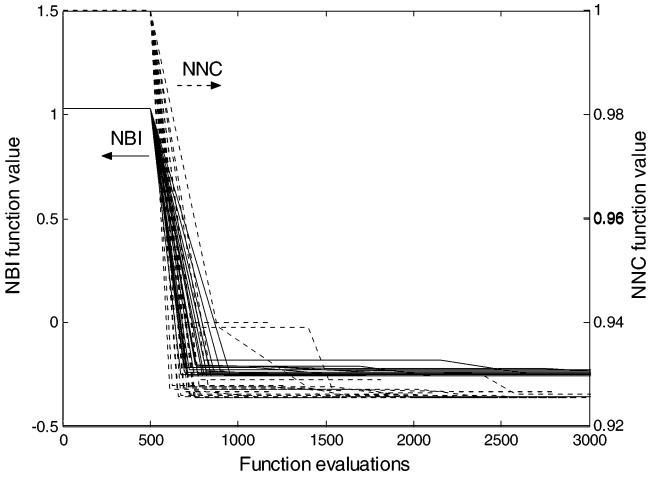

Figure 5. GLOBAL convergence curves for the NLP corresponding to $\mathbf{w} = [0.9375, 0.0625]$ in NBI (solid lines) and NNC (dotted lines).

It is worth comparing the performance of NBI and NNC, since they are based on similar ideas. In terms of quality of solutions, both methods are similar and produce an even spread of points in the Pareto front. In principle, the NNC approach should be computationally more efficient since it introduces inequality constraints instead of equalities. However, results with the SRES solver indicate the opposite. The convergence curves with SRES for the first NLP (corresponding to the weight vector $\mathbf{w} = [0.9375, 0.0625]$) in both NBI and NNC are depicted in Figure 4. It should be noted that 1 iteration (or generation) of the SRES solver corresponds to 100 evaluations of the objective function.

It can be seen that SRES presents a faster convergence in the NBI subproblems and it is able to arrive to values relatively close to the best solution in about 20−40 iterations. On the other side, the reduction of the objective space in the NNC formulation causes difficulties in creating feasible solutions and slows down the convergence of the population to the vicinity of the global minimum. This can be due to the fact that the constraints introduced by these methods are managed differently, but in the case of GLOBAL, both of them are included in the objective function by means of a penalty term; thus, the computational effort required by the NNC subproblems is inferior. The convergence curves of GLOBAL are depicted in Figure 5. Since a "nearby" point is always used in the initial sampling, the best solution is usually found in the first call to the local solver.

The efficiency of GLOBAL is related to that of UNIRANDI.

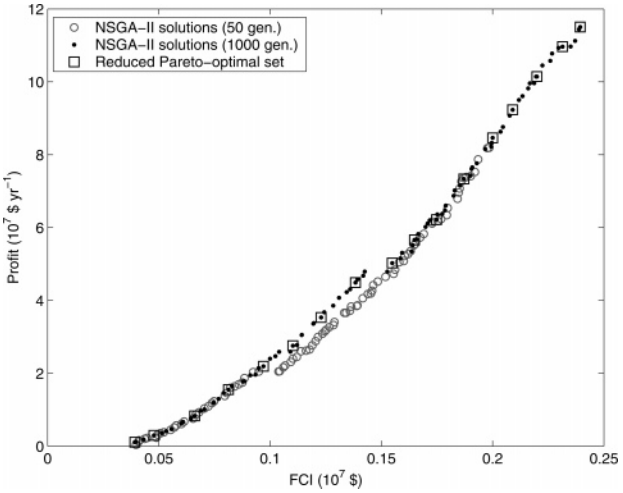

Figure 6. Points in the objective space obtained using NSGA-II.

GLOBAL will terminate the search if during one iteration (i.e., sampling, clustering, and local search) no new local minima are found. In this regard, the accuracy of methods such as UNIRANDI is rather low and it has been observed that this solver reports many false minima which are similar but are not close enough to be clustered around existing points. This prevents GLOBAL from stopping and increases the total number of function evaluations.

Results with NSGA-II. The NSGA-II algorithm was run with a population size of 100 during 1000 generations in order to ensure convergence and a good spread of points. In general, the algorithm is able to produce solutions covering the entire Pareto front, but in 10% of the runs, only solutions in the lower part (i.e., with values of the profit below the local maximum) were obtained. After the clustering phase, the distribution of points is not as good as that obtained with NBI and NNC (Figure 6).

Regarding computational effort, NSGA-II performed worse than the other methods, with a mean CPU time of 1300 s, but it is worth mentioning that after 50 generations all the individuals in the population are nondominated. It can be seen in Figure 6 that some of them have already converged to the global front but a good distribution of points has not been achieved yet.

Analysis of Solutions

Characteristics of the Optimal Solutions. The maximization of the venture profit (defined as $\phi = P - r \times \mathrm{FCI}$, with r being the investment rate of return) results in the same maxima as those found for the product profit P, indicating that P is the objective with the highest contribution to the cost function. Furthermore, the Pareto front for this problem is nonconvex. That means that points in the nonconvex parts cannot be obtained using a weighted sum of the objectives. Thus, this approach would ignore other possible designs which could be more appropriate, e.g., in terms of stability.

On the other side, the multiobjective optimization carried out here provides a set of optimal alternatives with respect to both economic criteria. This set of optimal solutions is characterized by a discontinuity located at a point which is a local maximum for the product profit P. The two parts of the Pareto front differ mainly in the values of the feed flow rates, which are plotted in Figure 7.

Designs with a product profit inferior to the local maximum are characterized by $F > 0$ and $F_C = 0$. As the FCI increases, this flow rate, together with the liquid holdup, increase up to

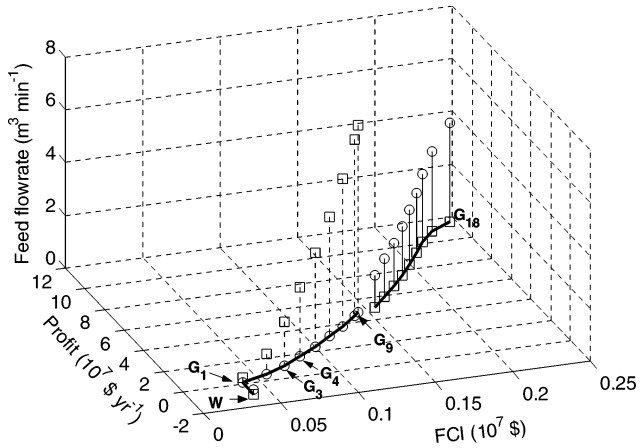

Figure 7. Values of the feed flow rates in the Pareto front: (□) diluted feed stream, F; (○) concentrated feed stream, F_C; (W) weakly Pareto optimal point; (G_X) global Pareto optimal points.

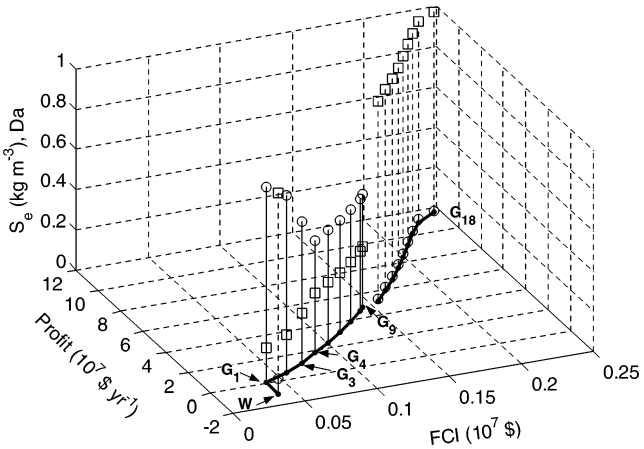

Figure 8. Values of the effective inlet substrate concentration, S_e (□) and *Da* (○) in the Pareto front: (W) weakly Pareto optimal point; (G_X) global Pareto optimal points.

the point G_9, i.e., the local maximum for P. For designs close to the minimum FCI (points G_1-G_3), the fraction α of the outlet stream recycled to the fermentor is also positive. This variable decreases progressively as FCI increases and eventually becomes zero at point G_4. On the other hand, points in the Pareto front with $P > 4.8852 \times \$10^7$ have $F = 0$ and $F_C > 0$. However, for these designs, the elevated concentrations can be out of the range of validity of the expression for the yield coefficient.[2]

Bifurcations Analysis. The Pareto-optimal points are evaluated hereafter in order to characterize each solution in terms of stability, providing an additional criterion for selecting a final solution among the optimal ones. A *bifurcation* diagram of the reactor dynamic model (eq 15) will provide an overall picture of the attractors of the system as a function of different parameters.

In the first instance, the bifurcation analysis on the reactor alone was carried out by the numerical continuation of different fixed points of the Pareto front with respect to the Damkholer number, thus illustrating the behavior of the system in the vicinity of each one of the solutions. If the kinetic parameters of the model remain unchanged, the dimensionless parameters β and γ depend exclusively on the effective inlet substrate concentration (S_e), which must be thus considered as the second significant parameter for a complete bifurcation analysis. The values of S_e and *Da* along the Pareto front (including the weakly Pareto optimal point found during the minimization of FCI) are plotted in Figure 8. Regarding these values, three regions can be identified in the Pareto front:

- Region I (points G_4-G_9) corresponds to designs with $Da = 0.5597$, $S_e = 0.3$ kmol·m^{-3}, $\gamma = 0.4$, and $\beta = 0.1$. Only the diluted feed stream is used, differing in the flow rate and the reactor volume. By performing a numerical continuation of equilibrium, we have obtained the curve of solutions for these points (Figure 9). A Hopf bifurcation point (H) was detected that divides the curve in a stable lower branch and an unstable upper branch of saddle points. As it can be observed, there is another steady state at a lower value of dimensionless biomass with the same *Da*, but it lies as well in the unstable branch above the Hopf bifurcation. This fact reaffirms the results of the simulations that reveal the *cell washout* of the reactor and can be overcome, for example, by acting over the kinetic factors. Starting a new continuation in the Hopf bifurcation point by varying simultaneously the parameters *Da* and γ, a new branch of periodic solutions emerges where stable configurations are found. In Figure 10, the attractors of the system are depicted with respect to *Da* and the parameter γ concerning kinetics.

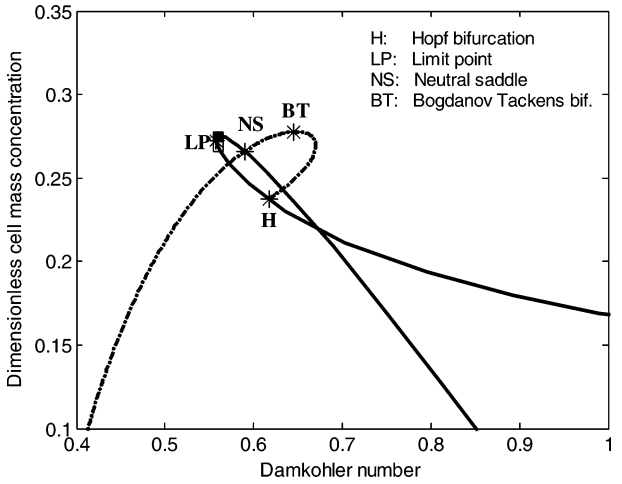

Figure 9. Steady-state attractors as a function of *Da* for $S_e = 0.3$ kmol·m^{-3}: (□) points G_4-G_9. Another solution (■) at the same *Da* is depicted lying also in the unstable branch.

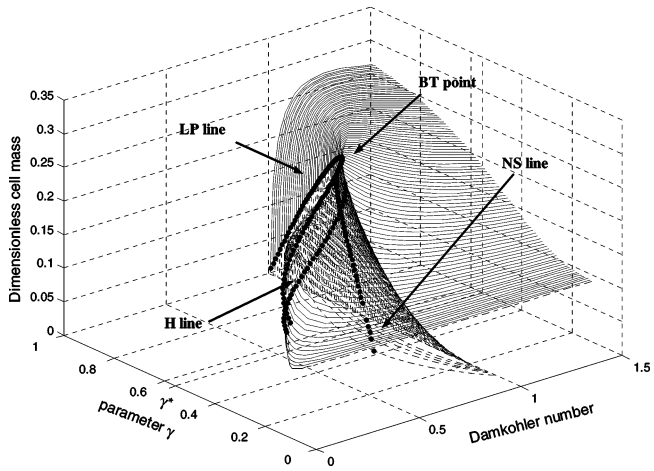

Figure 10. Bifurcation surface with respect to the parameters γ and *Da* with $\beta = 0.1$. The Hopf bifurcation curve coalesces with the neutral saddle curve in a Bogdanov−Takens bifurcation point at a critical γ^*. Black dashed lines represent unstable attractors; gray lines stand for stable attractors.

Under a critical value of γ (as that of designs G_4-G_9), the upper branches above the Hopf line are unstable, but once the system undergoes the Bogdanov−Takens bifurcation (BT), the upper stable branches over the limit point (LP) line turn to be stable, that is, stable attractors appear with high values of the biomass concentration.

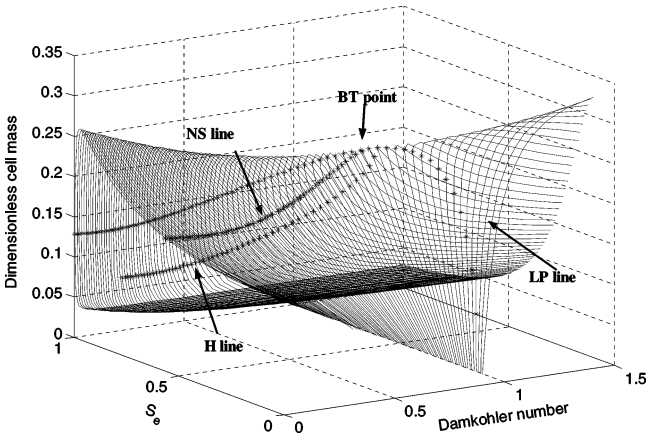

Figure 11. Steady-state attractors as a function of the Damkholer number and the effective inlet substrate concentration S_e. Both curves of Hopf and neutral saddle points coalesce in a Bogdanov−Takens bifurcation point at a critical S_e^*.

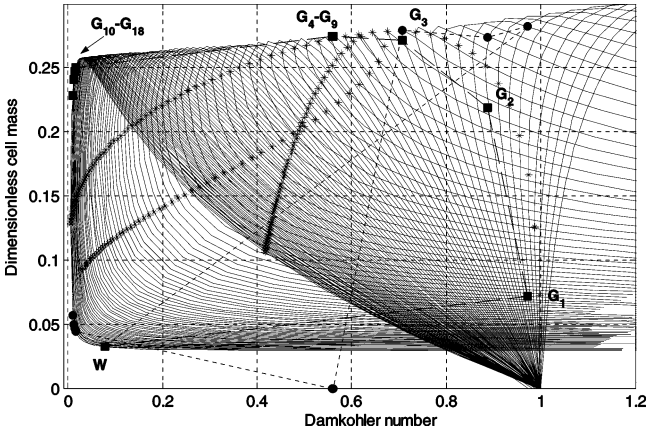

Figure 12. Pareto-optimal solutions (■) and stable steady states (●) plotted over the corresponding curves of attractors as a function of the Damkohler number for a range of S_e values between 0 and 1.

• Region II consists of points G_1-G_3. In these designs, the recirculation of the residual substrate stream affects the values of both S_e and *Da*. A unique singularity was found, but all of them lie in unstable branches under the limit point (LP). In this case, however, the system evolves until it reaches a stable steady state at its corresponding Damkholer.

The bifurcation analysis is performed by varying both parameters *Da* and S_e in order to cover all the designs not only in this region but also in the entire Pareto front. Results are shown in Figure 11. The equilibrium curve for each design as a function of *Da* can be seen by tracing a section at the corresponding value of S_e. As an example, the transversal section at $S_e = 0.3$ corresponds to the equilibrium curve depicted in Figure 9. At a critical value of S_e, where the Hopf bifurcation (H) and neutral saddle (NS) coalesce in a Bogdanov−Takens bifurcation point (BT), a new behavior arises corresponding to the equilibrium curves for points G_1-G_3.

• Region III consists of points $G_{10}-G_{18}$. For these points, the value of S_e is that of the concentrated feed stream and they also lie in unstable branches. The simulation of these designs indicates that they evolve to stable steady states which are relatively far from the optimal solutions but close to the weakly Pareto-optimal point, which is the *only* stable solution found in the optimization process.

Figure 12 shows the position of the Pareto-optimal points in the curves of solutions and the stable attractors (with the same Damkholer number) to which they evolve are indicated. It can be observed that design G_3 is the Pareto-optimal solution which is closest to the stable region.

Figure 12 summarizes all the relevant information provided by the bifurcation analysis:

(1) The stability or unstability of all the designs in the Pareto front.

(2) The relative "distance" from the Pareto-optimal solutions to possible bifurcation points.

(3) The relative "distance" from the unstable Pareto-optimal points to stable realizable designs.

Although the final decision may ultimately depend on other case-specific considerations, the decision maker can use this information to gain further insight into the process.

Conclusions

In this study, we have evaluated the performance of several single objective optimization methods within the framework of two multicriteria optimization schemes (NBI and NNC) and compared the results with those from a third multiobjective optimization solver, NSGA-II. As a case study, we have considered the optimal design of a fermentation process with respect to two economic criteria.

NBI and NNC are based on similar ideas and they are able to provide a good representation of the Pareto front since they produce an even spread of Pareto-optimal points. However, the associated NLPs can be very challenging to solve. In principle, the discretization of the Pareto front carried out by these methods provides good initial guesses to be used with local solvers. However, it should be noted that although the case study is a simple problem, it could not be solved satisfactorily with standard gradient-based optimization solvers and it caused difficulties even to a robust global optimization method like SRES. On the basis of the presented results, clustering algorithms such as GLOBAL are a good alternative to solve the set of NLPs resulting from these methods. GLOBAL is ultimately based on the application of a local solver, and its efficiency depends to a great extent on that of the local strategy.

Evolutionary algorithms such as NSGA-II are also good choices when the number of objective functions of the multicriteria optimization problem is low. They are able to generate the complete Pareto-optimal set in one single optimization run. It is important to note that the considered case study can be solved by NSGA-II with a relatively low number of function evaluations, but the actual computation time was surprisingly larger than that spent by the scalarizing techniques. Implementation differences apart, this is probably due to the computational effort of the nondominated sorting. Clustering techniques can be applied to reduce the size of the set of optimal solutions, but the success in providing a good representation of the Pareto front depends on the ability of the evolutionary algorithm to generate a Pareto front in which no region is over- or underrepresented.

Finally, bifurcation analysis via continuation methods has been applied to the system of ordinary differential equations that governs the dynamics of the fermentor, where the information of each specific design is contained in a critical parameter S_e. The manifolds of bifurcations which divide the space of solutions in different stability regions have been detected, observing that the points of the Pareto front lie in unstable regions. The information obtained via bifurcation analysis provides an additional decision criterion to modify the designs in a constructive manner in order to get a new stable and near-optimal solution design.

Acknowledgment

This work was supported by the Spanish Government (MEC Grant AGL2004-05206-C02-01/ALI) and Xunta de Galicia (PGIDIT05PXIC40201PN). Author J.-O.H.S. also acknowledges the CSIC for an I3P fellowship.

Literature Cited

(1) Banga, J. R.; Moles, C. G.; Alonso, A. A. Global Optimization of Bioprocesses using Stochastic and Hybrid Methods. In *Frontiers in Global Optimization. Nonconvex Optimization and Its Applications*; Floudas, C. A., Pardalos, P. M., Eds.; Kluwer Academics Publishers: Dordrecht, 2003; vol 74, pp 45−70.

(2) Brengel, D. D.; Seider, W. D. Coordinated Design and Control Optimization of Nonlinear Processes. *Comput. Chem. Eng.* **1992**, *16*, 861.

(3) Mönnigmann, M.; Marquardt, W. Steady-State Process Optimization with Guaranteed Robust Stability and Feasibility. *AIChE J.* **2003**, *49*, 3110.

(4) Miettinen, K. *Nonlinear Multiobjective Optimization*; Kluwer Academic Publishers: Dordrecht, 1999.

(5) Deb, K. *Multi-Objective Optimization using Evolutionary Algorithms*; Wiley: Chichester, 2001.

(6) Das, I.; Dennis, J. E. Normal Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems. *SIAM J. Optim.* **1998**, *8*, 631.

(7) Messac, A.; Mattson, C. A. Normal Constraint Method with Guarantee of Even Representation of Complete Pareto Frontier. *AIAA J.* **2004**, *42*, 2101.

(8) Coello, C. A. C. A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques. *Knowledge Inf. Syst. Int. J.* **1999**, *1*, 269.

(9) Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182.

(10) Srinivas, N.; Deb, K. Multiobjective Optimization using Nondominated Sorting in Genetic Algorithms. *Evol. Comput.* **1995**, *2*, 221.

(11) Bhaskar, V.; Gupta, S. K.; Ray, A. K. Applications of Multiobjective Optimization in Chemical Engineering. *Rev. Chem. Eng.* **2000**, *16*, 1.

(12) Moles, C. G.; Gutierrez, G.; Alonso, A. A.; Banga, J. R. Integrated Process Design and Control Via Global Optimization: A Wastewater Treatment Plant Case Study. *Chem. Eng. Res. Des.* **2003**, *81*, 507.

(13) Csendes, T. Nonlinear Parameter Estimation by Global Optimization: Efficiency and Reliability. *Acta Cybernetica* **1988**, *8* (4), 361.

(14) Järvi, T. A Random Search Optimizer with an Application to a Max-Min Problem. *Publications of the Institute for Applied Mathematics*; University of Turku: Finland, 1973; No. 3.

(15) Runarsson, T. P.; Yao, X. Stochastic Ranking for Constrained Evolutionary Optimization. *IEEE Trans. Evol. Comput.* **2000**, *4*, 284.

(16) MacQueen, J. B. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, (Berkeley, CA, June 21−July 18, 1965 and December 27, 1965−January 7, 1966), University of California Press: Berkeley, 1967; Vol. 1, pp 281−297.

(17) Jain, B. J.; Pohlheim, H.; Wegener, J. On Termination Criteria of Evolutionary Algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-2001*, (San Francisco, CA, July 7−11, 2001). Spector, E., Goodman, E., Wu, A., Langdon, W. B., Voigt, H. M., Gen, M., Sen, S., Dorigo, M., Pezesh, S., Garzon, M., Burke, E., Eds.; Morgan Kaufmann Publishers: San Francisco, CA, 2001.

(18) Dhooge, A.; Govaerts, W.; Kuznetsov, A.; Mestrom, W.; Riet, A. M. CL_MATCONT: A Continuation Toolbox in Matlab. *Proceedings of the 2003 ACM Symposium on Applied Computing*, Melbourne, FL, March 2003; pp 161−166 (http://www.matcont.ugent.be).

(19) Banga, J. R.; Seider, W. D. Global Optimization of Chemical Processes using Stochastic Algorithms. In *State of the Art in Global Optimization: Computation Methods and Applications*; Floudas, C. A., Pardalos, P. M., Eds.; Kluwer Academic Publisher: Dordrecht, 1996; pp 563−583.

(20) Agrawal, P.; Lee, C.; Lim, H. C.; Ramkrishna, D. Theoretical investigations of dynamic behaviour of isothermal continuous stirred tank biological reactors. *Chem. Eng. Sci.* **1982**, *37*, 453.

(21) Sendin, J. O. H.; Moles, C. G.; Alonso, A. A.; Banga, J. R. Multiobjective Integrated Design and Control using Stochastic Global Optimization Methods. In *CACE Book on Integration of Design and Control*; Georgiados, M., Seferlis, P., Eds.; Elsevier Science: Amsterdam, 2004; pp 555−581.

Received for review April 28, 2006
Revised manuscript received July 14, 2006
Accepted July 17, 2006

IE0605433