

Process Systems Engineering, 3. Mathematical Programming (Optimization)

LUKE E. K. ACHENIE, Virginia Polytechnic Institute and State University, Department of Chemical Engineering, Blacksburg, Virginia, United States 24061

ANGELO LUCIA, University of Rhode Island, Department of Chemical Engineering, Kingston, Rhode Island, United States 02811

URMILA DIWEKAR, Vishwamitra Research Institute, Center for Uncertain Systems: Tools for Optimization and Management, Clarendon Hills, Illinois, United States 60514

GUENNADI OSTROVSKY, Karpov Institute of Physical Chemistry, Moscow, Russia

1.	Introduction	1	4.	Optimization under Uncertainty in Process Systems Engineering	13
2.	General Optimization Concepts	2	4.1.	Methods and Algorithms	15
3.	Global Terrain Method	8	4.2.	Uncertainty Analysis and Sampling	16
3.1.	Simple Example of the Global Terrain Method	10	4.3.	Optimization Algorithms	18
3.2.	General Optimization Problems	12	4.4.	Applications	19
3.3.	Advanced Techniques	13	4.5.	Future Directions	23
3.4.	Applications and Extensions of the Terrain Method	13			

1. Introduction

According to Wikipedia, optimization is defined as: "In mathematics and computer science, optimization, or mathematical programming, refers to choosing the best element from some set of available alternatives. In the simplest case, this means solving problems in which one seeks to minimize or maximize a real function by systematically choosing the values of real or integer variables from within an allowed set . . ." Optimization has well over 10 major subfields, each of which has several researchers/adherents. For this article, the above definition which is quite good and pertains to several disciplines including process systems will be adopted. In process systems a systematic and thorough study of a chemical process to understand the relationship among the possibly many process variables is done. Although some of the relationships are observable, for most processes the understanding of the relationships is rather limited. Various mathematical formulae (commonly referred to as process models) have been developed to mimic the observed

relationships. With these process models, one is equipped to devise ways to systematically improve the process using process optimization concepts. This is done by appropriately setting values of a subset of the process variables at levels that would make another subset of process variables attain their desired values. This chapter discusses process systems optimization.

A process systems optimization model [1–3] involves the minimization or maximization of a function while keeping constraints within acceptable limits. A fairly general optimization model is given as

$$\begin{aligned} \min_{x,u} \xi(z, y, u) \\ \varphi_i(z, y, u) = 0 \quad i = 1, \dots, m \\ \psi_j(z, y, u) \leq 0 \quad j = 1, \dots, p \\ z \in \mathcal{R}^{n_z} y \in \{-n_1, \dots, n_2\}^{n_y} \quad u \in \mathcal{R}^{n_u} \end{aligned} \quad (1)$$

Here, z and y are vectors of continuous/discrete valued process variables, respectively. In several process systems models, each

member of y tends to take the value of 0 or 1 ($y \in \{0, 1\}^{m_2}$) indicating the absence or presence of a process unit. The vector of decision variables are given by u . Examples of process variables are $z = [\text{concentration}, \text{temperature}]^T$ and $u = [\text{feedconcentration}, \text{reactorsize}]^T$. It is straightforward to convert a maximization problem into a minimization problem (and vice-versa). Specifically $\min_w \xi(w)$ is equivalent to $\max_w (-\xi(w))$. If minimization is considered, there is no loss of generality.

The function $\xi(z, y, u)$ is more commonly referred to as the performance objective (e.g., profit); it is a measure of how good the design is. Sometimes there are several competing performance objectives as follows: $\{\xi_1, \dots, \xi_p\}$; in such a case a new objective function is formulated from the available set, for example, by using a weighted average of the set. A more general combination of the individual objectives leads to a convolution operator [4]. The set $\{\varphi_i(z, y, u)\}$ includes material and energy balances while the set $\{\psi_j(z, y, u)\}$ may represent process constraints such as safety, environmental limitations, and process specifications. Sometimes the model is more conveniently expressed as

$$\begin{aligned} \min_x \xi(x) \\ \varphi_i(x) = 0 \quad i = 1, \dots, m \\ \psi_j(x) \leq 0 \quad j = 1, \dots, p \\ x^L \leq x \leq x^U \end{aligned} \quad (2)$$

where $x = [z, y, u]$ and the superscripts $[L, U]$ denote lower and upper bounds which are also called box-constraints.

The strategy for the solution of the optimization model may either result in a locally optimal solution or a globally optimal solution.

2. General Optimization Concepts

To be a viable optimization problem at the very least one degree of freedom is required, in other words $n_z + n_y > m + p$ (Eqs. 1, 2) and [5] for an in depth exposition of conditions under which solutions exist. For now it is assumed the optimization model has a solution.

There are many process optimization formulations that are multimodal, i.e., lead to multiple local optima. Sometimes one is interested in finding the best optima and global optimization algorithms have to be employed [6, 7]. The methods that are discussed in this section are local optimization methods. A crude but practical way to find or get close to global optima (using local optimization algorithms) is to run the algorithm for multiple initial starting points and keeping the best solution. A good collection of mostly free optimization software can be found at \rightarrow Mathematics in Chemical Engineering, Chap. 10 and the following web sites:

Local optimization: http://www.mat.univie.ac.at/~neum/glopt/software_1.html [8]

Global optimization: http://www.mat.univie.ac.at/~neum/glopt/software_g.html [9]

Unconstrained Optimization. For unconstrained optimization the equality and inequality constraint sets $\{\varphi_i\}, \{\psi_j\}$ and the box-constraints are all absent. For simplicity all variables are assumed to be continuous (i.e., the discrete variable y is absent). The explicit consideration of discrete variables leads to combinatorial optimization [10] and the unconstrained optimization problem becomes

$$\min_x \xi(x) \quad (3)$$

Several solution approaches have been developed [11]. The necessary conditions for optimality is obtained by setting the derivative equal to zero as follows

$$\nabla \xi(x) = f(x) = 0 \quad f \in \mathbb{R}^n \quad (4)$$

Gradient based iterative methods for solving Equation (4) determine a new improved point as

$$x_{j+1} = x_j + \alpha_j s_j = x_j + p_j \quad (5)$$

where x_j is the current point and α_j determines the step length along the search direction s_j , given as

$$s_j = -G_j g_j$$

$$g_j = \nabla f(x_j) \quad (6)$$

Here, g_j is the gradient of $f(x)$ at x_j . The choice of the matrix G_j depends on the solution strategy. Common to most solution approaches for unconstrained minimization is the following generic algorithm 1:

- Step 1: Set $j = 0$. Give an initial guess x_0 and G_0
- Step 2: If $j > 0$ determine G_j
- Step 3: Find the search direction $s_j = -G_j g_j$ $g_j = \nabla f(x_j)$
- Step 4: Perform a line search along s_j ; that is, determine α_j such that $f(x_j + \alpha_j s_j) < f(x_j)$
- Step 5: STOP with the solution if $\|\nabla f\| \leq \epsilon$
- Step 6: Set $j = j+1$. Go to Step 2

Line search methods are described in detail in [12]. If $G_j = I$ (the identity matrix) then algorithm 1 is the *steepest descent method*. If $G_j = (H(x_j))^{-1}$ (where $H(x)$ is the Hessian of $f(x)$), then the algorithm is the *Newton method*.

The steepest descent method only requires the gradient of $f(x)$ and has a linear rate of convergence close to the solution. On the other hand the Newton method has quadratic convergence rate close to the solution and is therefore faster close to the solution of Equation (3). However, it requires second derivatives (Hessian) of $f(x)$. To avoid the computational burden of calculating the Hessian, quasi-Newton methods were developed, in which some approximation of the Hessian (or inverse Hessian) is employed. Quasi-Newton methods have super-linear convergence rate close to the solution and require only first derivative information. Finally, steepest descent has *global convergence* (for convex, differentiable functions) in the sense that the method converges from any starting point as long as first derivatives can be calculated with enough accuracy. In contrast the Newton method does not have global convergence for a number of reasons; one of these is the possibility of $H(x)$ becoming singular.

The main distinguishing feature of any of the above methods is the technique for forming $G(x)$.

Newton Method. Consider a set of equations described by the vector quantity $h(x) = [h_1(x), \dots, h_m(x)]^T = 0$, where the expression from Equation (4) is a special case of this. For $h(x) = 0$, the Jacobian is defined as

$$J = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m}{\partial x_1} & \dots & \frac{\partial h_m}{\partial x_n} \end{bmatrix} \quad (7)$$

Subsequently the update steps in the Newton method can be written as

$$x_{j+1} = x_j + \alpha_j s_j \quad 0 \leq \alpha_j \leq 1$$

$$J_j s_j = -h_j \quad h_j \equiv h(x_j) \quad (8)$$

In order to speed up the method in the early stages of iteration and at the same time improve convergence of the method close to the solution, a good rule of thumb is to initially pick the *step length* α_j to be 1 and systematically reduce its value. The step length is based on

$$\|h_{j+1}\| < \|h_j\|$$

$$\|h_j\| \equiv \sqrt{\sum_{i=1}^n h_{ij}^2} \quad (9)$$

where h_{ij} is the i -th component of h_j . In a vicinity of the solution the algorithm has quadratic convergence; in other words if x^* is the solution of Equation (4), then

$$\|x_{i+1} - x\| \leq c \|x_i - x\|^2.$$

The Newton method has the following properties:

- The method requires a good initial approximation x_0 . If the latter is bad the method often converges slowly or does not converge at all
- It is necessary to solve at each iteration the set of linear equations in Equation (8) for the search direction, thus exacting a significant computational overhead
- There is a need to calculate the Jacobian at each iteration

Step 3 leads to significant computational overhead at each iteration because quite often, the difference approximation is the only viable way to obtain estimates of the derivatives in the Jacobian matrix; thus $(n+1)$ evaluations of $h(x)$ are needed; on the other hand quasi-Newton methods have less computational burden at each iteration since they do not require the explicit evaluation of the Jacobian.

Quasi-Newton Method. In the quasi-Newton method x_{j+1} is assumed to be known and Equation (7) is used to determine J_j . In Table 1 differences in the Newton and quasi-Newton methods are shown.

Table 1. Comparison of Newton and quasi-Newton methods

	x_{j+1}	J_j	Goal
Newton	unknown	known	determine x_{j+1}
Quasi-Newton	known	unknown	determine J_j

The update steps in the quasi-Newton method can be written as

$$s_j = -G_j h_j \quad (10)$$

The updated G_j can be computed using the Broyden–Fletcher–Goldfarb–Shanno (BFGS) [13] formula

$$G_{j+1} = G_j + \frac{y_j y_j^T}{s_j^T y_j} - \frac{G_j s_j s_j^T G_j}{s_j^T G_j s_j} \quad (11)$$

Constrained Optimization. Methods for solving Equation (2) are considered. Let D be the feasible region of the problem, thus $D = \{x : \phi(x) = 0; \psi(x) \leq 0\}$. The Lagrange function is defined as

$$L = f(x) - \sum_{i=1}^m \lambda_i \phi_i + \sum_{j=1}^p \mu_j \psi_j \quad (12)$$

The Kuhn–Tucker necessary conditions for the minimum of L [12, 13] are given as follows

$$\frac{\partial L}{\partial x_i} = 0 \quad (13)$$

$$\mu_j \geq 0 \quad j = 1, \dots, p \quad (14)$$

$$\mu_j \psi_j = 0 \quad j = 1, \dots, p \quad (15)$$

$$\phi_i(x) = 0 \quad i = 1, \dots, m \quad (16)$$

$$\psi_j(x) \leq 0 \quad j = 1, \dots, p \quad (17)$$

Let $\{x, f^*\}$ (where $f = f(x^*)$) be the solution of Equation (2). The equality in Equation (15) implies that

- If $\mu_j > 0$ then $\psi_j(x) = 0$
- If $\psi_j(x) < 0$ then $\mu_j = 0$.

Methods for solving the nonlinear program in Equation (2) are listed below:

1. Successive solution of unconstrained optimization methods (penalty methods [14], Langrange multipliers methods [15], and augmented Lagrangian methods [16])
2. Successive solution of quadratic programming subproblems; this method is more commonly referred to as sequential quadratic programming (SQP) [1, 17] and \rightarrow Mathematics in Chemical Engineering, Section 10.2
3. Successive solution of nonlinear optimization problems with linear constraints [18]

Langrange Multiplier Method. For considering the Langrange multiplier method [15] the *duality function* is introduced

$$h(\lambda, \mu) = \min_x L(x, \lambda, \mu) \quad (18)$$

where $h(\lambda, \mu)$ gives a lower bound of the optimal value of the objective function in Equation (2). Therefore

$$h(\lambda, \mu) \leq f(x) \quad (19)$$

The following relation also holds

$$h(\lambda, \mu) \leq L(x, \lambda, \mu) \quad \forall x \in X, \quad \forall \lambda, \quad \forall \mu > 0 \quad (20)$$

hence, in particular

$$h(\lambda, \mu) \leq L(x, \lambda, \mu) \quad (21)$$

Under some convexity conditions the optimal objective function value from Equation (2) satisfies

$$f = \max_{\lambda, \mu \geq 0} h(\lambda, \mu) = \max_{\lambda, \mu \geq 0} \min_x L(x, \lambda, \mu) \quad (22)$$

Thus the constrained optimization problem is reduced to a two-level optimization procedure. At the first level the unconstrained optimization problem in Equation (18) is solved and at the second level a problem with simple bound constraints is solved as follows

$$\begin{aligned} \max_{\lambda, \mu} h(\lambda, \mu) \\ \mu \geq 0 \end{aligned} \quad (23)$$

If Equation (18) has a single solution then the derivatives of $h(\lambda, \mu)$ are calculated as

$$\frac{\partial h}{\partial \lambda_i} = \frac{\partial L}{\partial \lambda_i} = -\phi_i(x)$$

$$\frac{\partial h}{\partial \mu_j} = \frac{\partial L}{\partial \mu_j} = \psi_j(x, \lambda, \mu) \quad (24)$$

Penalty Method. The outer penalty method [14] may be used for solving the problems stated in Equation (2); the following auxiliary function results

$$F(x, \xi_i, \gamma_j) = f + \sum_{i=1}^m \xi_i \phi_i^2 + \sum_{j=1}^p \gamma_j Q(\psi_j) \quad (25)$$

where $\xi_i \gg 1$, $\gamma_j \gg 1$ and $Q(\psi_j)$ is given by

$$Q(\psi_j) = \begin{cases} 0, & \text{if } \psi_j \leq 0 \\ \psi_j^2 & \text{if } \psi_j > 0. \end{cases}$$

Looking for the minimum of $F(x, \xi_i, \gamma_j)$ with respect to x one can show that under certain conditions the solution of the unconstrained problem

$$\min_x F(x, \xi_i, \gamma_j) \quad (26)$$

will tend to the solution of Equation (2) as $\xi_i \rightarrow \infty$ and $\gamma_j \rightarrow \infty$ [14]. However, for large ξ_i and γ_j the function $F(x, \xi_i, \gamma_j)$ is ill conditioned [14]. To partially avoid this problem, minimization of $F(x, \xi_i, \gamma_j)$ for relatively small values of the penalty coefficients $\{\xi_i, \gamma_j\}$ is carried out. Subsequently, the penalty coefficients are systematically increased and minimization of $F(x, \xi_i, \gamma_j)$ is repeated. Also, similar to of the Lagrange multiplier method, the procedure for solving Equation (2) is two-level. Specifically, minimization of F with respect to x corresponds to the first (lower) level and the increment in the coefficients $\{\xi_i, \gamma_j\}$ corresponds to the second (upper) level.

The Modified Lagrange Function Method.

For simplicity the case when inequality constraints are absent is considered and the modified Lagrange function is defined as

$$\Phi(x, \lambda, \xi) = L + \xi \sum_{i=1}^m \varphi_i^2 \quad (27)$$

where L is the original Lagrange function

$$L = f + \sum_{i=1}^m \lambda_i \varphi_i$$

Recalling that $\{x, f^*\}$ is the minimizer of Equation (2) let λ^* be the corresponding vector of Lagrange multipliers. At x^* the necessary condition for optimality of Equation (2) is given by

$$\nabla L \equiv \nabla f + \sum_{i=1}^m \lambda_i \nabla \varphi_i = 0 \quad (28)$$

Note the following

$$\nabla \Phi = \nabla f + \sum_{j=1}^m (\lambda_j + 2\xi \phi_j) \nabla \phi_j \quad (29)$$

Since at x^* constraints of Equation (5) are met then according to Equation (28)

$$\nabla \Phi = \nabla L = 0 \quad (30)$$

One can show that for a sufficiently large and finite ξ , the minimum of Φ will correspond to the minimum of Equation (2) [16]. In this case, again, the constrained optimization problem in Equation (2) is reduced to a two-level procedure in which a sequence of unconstrained optimization problems is solved. Here, to the lower level minimization of Φ corresponds

$$\min_x \Phi(x, \lambda^{(i)}, \xi) \quad (31)$$

where $\lambda^{(i)}$ is the i -th approximation of the Lagrange multipliers and x^{*i} is the solution of the problem. On the second level we must change λ in order to obtain the solution of Equation (2) which is done as follows:

First minimization of Φ for fixed λ is performed. At the minimum point the following condition is met

$$\nabla \Phi = 0 \quad (32)$$

Since at the solution of Equation (2), Equation (28) must be satisfied, then the following approximation for the Lagrange multipliers can be used:

$$\begin{aligned} \lambda_j^{(i+1)} &= \lambda_j^{(i)} + 2\xi \varphi_j^* \\ \varphi_j^* &= \varphi_j(x^{*i}). \end{aligned} \quad (33)$$

The modified Lagrange function method does not have the drawbacks of either the penalty method or the method of Lagrange multipliers. In contrast to the method of Lagrange multipliers it does not require convexity of Φ in the vicinity of the solution of Equation (2). Moreover, in contrast to the penalty method there is no need to systematically increase the penalty coefficients towards infinity. In other words, in general Φ is not ill conditioned.

Sequential Quadratic Programming.

Some text books also refer to sequential quadratic programming (SQP) as *successive quadratic programming* [1]. Recalling that the

Lagrange function for Equation (2) is

$$L = f(x) - \lambda^T \phi + \mu^T \psi \quad (34)$$

the following definitions are needed

$$\chi_i(x, \lambda, \mu) = \begin{cases} \frac{\partial L}{\partial x_i}, & i = 1, \dots, n \\ \varphi_i(x), & i = (n+1), \dots, (n+m) \end{cases}$$

The vector function $\chi = (\chi_1, \chi_2 \dots \chi_{n+m})$ can be written as

$$\chi(x, \lambda, \mu) = \begin{cases} \nabla f - J_1^T \lambda + J_2^T \mu \\ \phi(x) \end{cases} \quad (35)$$

where J_1 and J_2 are Jacobian matrices

$$J_1 = \frac{\partial(\varphi_1, \dots, \varphi_m)}{\partial(x_1, \dots, x_n)} = \begin{pmatrix} \frac{\partial \varphi_1}{\partial x_1}, & \dots, & \frac{\partial \varphi_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial \varphi_m}{\partial x_1}, & \dots, & \frac{\partial \varphi_m}{\partial x_n} \end{pmatrix} \quad (36)$$

Example:

for $m = 2$ and $n = 3$,

$$J_1 = \frac{\partial(\varphi_1, \varphi_2)}{\partial(x_1, x_2, x_3)} = \begin{pmatrix} \frac{\partial \varphi_1}{\partial x_1} & \frac{\partial \varphi_1}{\partial x_2} & \frac{\partial \varphi_1}{\partial x_3} \\ \frac{\partial \varphi_2}{\partial x_1} & \frac{\partial \varphi_2}{\partial x_2} & \frac{\partial \varphi_2}{\partial x_3} \end{pmatrix}$$

$$J_2 = \frac{\partial(\psi_1, \dots, \psi_p)}{\partial(x_1, \dots, x_n)} = \begin{pmatrix} \frac{\partial \psi_1}{\partial x_1}, & \dots, & \frac{\partial \psi_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial \psi_p}{\partial x_1}, & \dots, & \frac{\partial \psi_p}{\partial x_n} \end{pmatrix} \quad (37)$$

Using the Karush–Kuhn–Tucker (KKT) necessary optimization conditions [19] and → Mathematics in Chemical Engineering, Section 10.2 one obtains

$$\chi(x, \lambda, \mu) = 0$$

$$\text{diag}(\psi)\mu = 0$$

$$\psi(x) \leq 0$$

$$\mu \geq 0 \quad (38)$$

Here, $\mu = (\mu_1, \dots, \mu_p)^T$, $\text{diag}(\psi)$ is a diagonal matrix with elements $\{\psi_1, \dots, \psi_p\}$. Thus for solving the problem in Equation (2), it is necessary to find the solution of a system of $(n + m + p)$ nonlinear equations and $2p$ inequalities with respect to $(n + m + p)$ variables $\{x, \lambda, \mu\}$.

If Equation (2) does not contain inequalities then the KKT conditions are reduced to a system of nonlinear equations with $(n + m)$ variables $\{x, \lambda\}$ which can be solved using, for example, the Newton method. In general, the presence of the inequalities complicates the problem.

Now an extension of Newton's method for solving a set of nonlinear equations is considered. Let $\{x^*, \lambda^*, \mu^*\}$ be the solution of Equation (38) and $\{\bar{x}, \bar{\lambda}, \bar{\mu}\}$ be values at some iteration. Furthermore, the point $\{\bar{x}, \bar{\lambda}, \bar{\mu}\}$ is sufficiently close to $\{x^*, \lambda^*, \mu^*\}$ such that for small values $\{\Delta x, \Delta \lambda, \Delta \mu\}$

$$x^* = \bar{x} + \Delta x; \lambda^* = \bar{\lambda} + \Delta \lambda; \mu^* = \bar{\mu} + \Delta \mu \quad (39)$$

Let $g = (\chi_1, \dots, \chi_{n+m}, \psi_1, \dots, \psi_p)$ and $z = (x_1, \dots, x_n, \lambda_1, \dots, \lambda_m, \mu_1, \dots, \mu_p)$ then the Jacobian matrix is of the form

$$J = \frac{\partial g}{\partial z} = \frac{\partial(\chi_1, \dots, \chi_{n+m}, \psi_1, \dots, \psi_p)}{\partial(x_1, \dots, x_n, \lambda_1, \dots, \lambda_m, \mu_1, \dots, \mu_p)} \quad (40)$$

In the matrix J , the first $n + m$ rows correspond to the functions $\chi_1, \dots, \chi_{n+m}$ and the last rows with indices $(n + m + 1, \dots, n + m + p)$ correspond to the functions ψ_1, \dots, ψ_p . The first n columns correspond to x , the following m columns correspond to λ , and the last p columns correspond to μ . The matrix J can be represented in the following block form

$$J = \begin{pmatrix} \frac{\partial(\chi_1, \dots, \chi_n)}{\partial(x_1, \dots, x_n)} & \frac{\partial(\chi_1, \dots, \chi_n)}{\partial(\lambda_1, \dots, \lambda_m)} & \frac{\partial(\chi_1, \dots, \chi_n)}{\partial(\mu_1, \dots, \mu_p)} \\ \frac{\partial(\chi_{n+1}, \dots, \chi_{n+m})}{\partial(x_1, \dots, x_n)} & \frac{\partial(\chi_{n+1}, \dots, \chi_{n+m})}{\partial(\lambda_1, \dots, \lambda_m)} & \frac{\partial(\chi_{n+1}, \dots, \chi_{n+m})}{\partial(\mu_1, \dots, \mu_p)} \\ \frac{\partial(\psi_1, \dots, \psi_p)}{\partial(x_1, \dots, x_n)} & \frac{\partial(\psi_1, \dots, \psi_p)}{\partial(\lambda_1, \dots, \lambda_m)} & \frac{\partial(\psi_1, \dots, \psi_p)}{\partial(\mu_1, \dots, \mu_p)} \end{pmatrix} \quad (41)$$

Using the fact that ϕ and ψ do not depend on $\{\lambda, \mu\}$, J takes the form

$$J = \begin{pmatrix} G & -J_1^T & +J_2^T \\ J_1 & 0 & 0 \\ J_2 & 0 & 0 \end{pmatrix} \quad (42)$$

where G is the Hessian (with respect to x) of L ($G = \nabla^2 L$). Now substitute $\{x^*, \lambda^*, \mu^*\}$ from Equation (39) into the system Equation (38) and carrying out a second-order Taylor series expansion of the left-hand side of the system

Equation (38), this system is reduced to

$$J \begin{pmatrix} \Delta x \\ \Delta \lambda \\ \Delta \mu \end{pmatrix} = \begin{pmatrix} -\nabla L(\bar{x}, \bar{\lambda}, \bar{\mu}) \\ -\phi(\bar{x}) \\ \leq -\psi(\bar{x}) \end{pmatrix} \quad (43)$$

and

$$\text{diag}(\psi + \delta\psi)(\mu + \Delta\mu) = 0$$

$$\bar{\mu} + \delta\mu \geq 0 \quad (44)$$

where $\delta\psi = J_2\Delta\mu$.

In order to determine $\{\Delta x, \Delta\lambda, \Delta\mu\}$ it is necessary to solve the above system of linear inequalities. One approach to solving the problem is to consider the following quadratic programming problem:

$$\begin{aligned} \min_y & \left(g^T y + \frac{1}{2} y^T G y \right) \\ J_1 y + \bar{\phi} &= 0 \\ J_2 y + \bar{\psi} &\leq 0 \end{aligned} \quad (45)$$

where $\bar{\phi} = \phi(\bar{x})$; $\bar{\psi} = \psi(\bar{x})$; $g = \nabla f$. The Lagrange function for this problem is

$$\widehat{L} = g^T y + \frac{1}{2} y^T G y - \nu^T (J_1 y + \bar{\phi}) + \theta^T (J_2 y + \bar{\psi})$$

where ν and θ are Lagrange multipliers corresponding to the equality and inequality constraints. The relevant KKT conditions are

$$\begin{aligned} \nabla_y \widehat{L} &= g + G y - J_1^T \nu + J_2^T \theta = 0 \\ J_1 y + \bar{\phi} &= 0 \\ J_2 y + \bar{\psi} &\leq 0 \\ \text{diag}(J_2 y + \bar{\psi}) \theta &= 0 \\ \theta &\geq 0 \end{aligned} \quad (46)$$

Substituting the quantities $y = \Delta x$, $\nu = \bar{\lambda} + \Delta\lambda$, $\theta = \bar{\mu} + \Delta\mu$ into Equation (46), one obtains Equation (43). Thus $\{\Delta x, \Delta\lambda, \Delta\mu\}$ can be obtained by solving the quadratic programming (QP) problem to obtain a new point in the space of x and so on. It should be noted that such a procedure could diverge just like the Newton method. Therefore, a modification similar to the

modified Newton method can be used. Specifically, starting from $\{\bar{x}, \bar{\lambda}, \bar{\mu}\}$ one can do a *line search* in the space of the variables $\{x, \lambda, \mu\}$ along the direction $p = \{\Delta x, \Delta\lambda, \Delta\mu\}$. There are other line search strategies, e.g., based on a minimization of the norm of $L(x, \lambda, \mu)$. The Armijo line search has been used extensively [1] with great success.

In the following an SQP algorithm for Equation (4) is outlined:

- Step 0: Initialize z , iteration counter $i \leftarrow 0$, and other initializations
- Step 1: At the i -th iteration form the Lagrange function $L^{(i)} = f(x) - (\lambda^{i-1})^T \phi(x) + (\mu^{i-1})^T \psi(x)$ where λ^{i-1}, μ^{i-1} are Lagrange multipliers
- Step 2: Find the Jacobian for ϕ and ψ , the gradient ∇f and the Hessian $\nabla^2 L^{(i)}$
- Step 3: Solve the QP in Equation (45) to yield a new search direction p^i
- Step 4: Do a line search along p^i and find next iteration point $z^i = z^{i-1} + \alpha^i p^i$ ($z^i \equiv \{x^i, \lambda^i, \mu^i\}$), where α^i is the search length
- Step 5: Check the termination criterion. If the Karush–Kuhn–Tucker error is less than the desired tolerance ϵ , STOP with the optimal solution
- Step 6: Set $i \leftarrow i+1$ and go to Step 1

The drawbacks of this variant of the SQP algorithm consist in:

1. At each search point the Hessian ($G \equiv \nabla^2 L$) needs to be evaluated
2. $\nabla^2 L$ may not be positive semi-definite, which means the search direction may not be a descent direction (i.e. will not lead to the minimum of Equation (4))

To overcome the first drawback one can employ a quasi-Newton approximation of the Hessian, given by the BFGS relations

$$B_i = B_{i-1} + \frac{y_i y_i^T}{y_i^T s_i} - \frac{B_{i-1} s_i s_i^T B_{i-1}}{s_i^T B_{i-1} s_i} \quad (47)$$

where i is an iteration counter, B_i is an approximation of $\nabla^2 L$. That is employing $G \equiv B_i$ instead of $G \equiv \nabla^2 L$. Here $y_i = (\nabla L_i - \nabla L_{i-1})$, $s_i = (x_i - x_{i-1})$, and B_i is positive definite if

B_{i-1} is positive definite [17] leading to

$$y_i^T s_i = s_i^T y_i > 0 \quad (48)$$

Consider, for example, the quadratic function $y = z^T A z + b^T z + c$. In this case $y_i = A z$ and $s_i^T y_i = s_i^T A s_i$. If the quadratic function is positive definite then $A > 0$ and Equation (26) holds for all search points. Since $\nabla^2 L$ can be indefinite, one cannot ensure satisfaction of Equation (48) at all search points. Therefore, instead of the transformation Equation (47), one can employ the following transformation [17]

$$B_i = B_{i-1} + \frac{w_i w_i^T}{w_i^T s_i} - \frac{B_{i-1} s_i s_i^T B_{i-1}}{s_i^T B_{i-1} s_i} \quad (49)$$

where

$$w = \theta y_i + (1 - \theta) B_{i-1} s_i$$

$$\theta = \begin{cases} 1 & \text{if } s_i^T y_i \geq 0.2 s_i^T B_{i-1} s_i \\ 0.8 \frac{s_i^T B_{i-1} s_i}{s_i^T B_{i-1} s_i - s_i^T y_i} & \text{if } s_i^T y_i < 0.2 s_i^T B_{i-1} s_i \end{cases} \quad (50)$$

It should be noted that B_i obtained from Equation (49) will be positive definite if B_{i-1} is positive definite.

3. Global Terrain Method

The global terrain method [20, 21] is a deterministic method of global optimization. The basic idea behind the terrain method is to intelligently explore the terrain of an objective function by following valleys. The terrain method does this by moving up and down the objective function surface using well known numerical methods for each of its basic tasks (i.e., moving downhill, moving uphill, accelerating convergence to singular points) and by using eigenvalue and eigenvector information to decide what to do next. The novel aspects of the global terrain method include (1) the use of well established building blocks (numerical methods such as Newton's method, a trust region strategy, an SQP method, etc.) assembled in a unique manner, and (2) a novel SQP corrector formulation and algorithm for returning iterates to valleys.

The terrain method assumes that the objective function, ξ , is a twice continuously differentiable of unknown variables $z \in R^n$ with a gradient given by $g = g(z)$ and a Hessian matrix or matrix of second partial derivatives denoted by $H = H(z)$. It is also assumed that the feasible

region is defined by upper and lower bounds on variables of the form

$$z_i^L \leq z_i \leq z_i^U \quad i = 1, \dots, n \quad (51)$$

where z_i^L and z_i^U are the lower and upper bounds on the i^{th} variable, respectively. Other constraints (e.g., linear mass balance constraints, etc.) are handled using projection or elimination.

It is easiest to view the terrain method by considering the nonlinear least squares problem. Let $\xi = \xi(z)$ and $J = J(z)$ denote the Jacobian matrix of the function ξ ; thus the nonlinear least squares objective function is $\xi^T \xi$. The appropriate global optimization problem then becomes that of finding all stationary and singular points of

$$\xi^T \xi \quad \text{subject to} \quad z_i^L \leq z_i \leq z_i^U \quad i = 1, \dots, n \quad (52)$$

The corresponding gradient of $\xi^T \xi$ is

$$g = g(z) = J^T \xi \quad (53)$$

The corresponding Hessian matrix, $H(z)$, of ξ is given by

$$H = H(z) = J^T J + \sum \xi_i \nabla^2 \xi_i \quad (54)$$

where $\nabla^2 \xi_i$ are the element Hessian matrices of the component functions, ξ_i .

The global terrain method is comprised of five basic steps:

1. Downhill movement
2. Uphill movement
3. Second-order acceleration to singular points
4. Eigenvalue–eigenvector decomposition
5. Termination criterion

Downhill Movement. Downhill movement in the terrain method is accomplished using a trust region strategy [22] and given by

$$\Delta = -\beta \Delta_N + (\beta - 1)g \quad (55)$$

where $\Delta_N = J^{-1} \xi$ is the Newton step and where $\beta \in [0, 1]$ is determined by the following simple rules. If $\|\Delta_N\| \leq R$, then $\beta = 1$, where R is the trust region radius. If $\|\Delta_N\| > R$ and $\|g\| \geq R$, then $\beta = 0$. Otherwise, β is the unique value on $[0, 1]$ that satisfies $\|\Delta\| = R$. The new iterate is accepted if it reduces $\|\xi\|$. Otherwise, the new iterate is rejected, the trust region radius is reduced and the calculations are repeated until a reduction in $\|g\|$ occurs. Furthermore, if

during downhill movement, $\|g\|/\|\Delta_N\| \leq s$, then quadratic acceleration is used. (See Eq. 59 below).

Downhill movement is terminated when either $\|\xi\| \leq \epsilon$ or $\|g\| \leq \epsilon$ where ϵ is a convergence tolerance and can result in convergence to a minimum, saddle point, or singular point of $\xi^T \xi$.

Uphill Movement. Uphill movement consists of predictor–corrector steps. Uphill movement uses Newton predictor steps which can drift from the valley because of approximation. To correct for this drift, intermittent SQP corrector steps are used to force iterates back to the valley.

Predictor Steps: Uphill predictor steps are Newton steps, Δz_N , given by

$$\Delta z_N = \alpha_p J^{-1} \xi \quad (56)$$

where α_p is a step size such that $0 < \alpha_p \leq 1$, which can either remain fixed or be adjusted using available information. Uphill Newton steps tend to follow valleys reasonably well but do drift some. Therefore, corrector steps are used intermittently to return iterates to the current valley and are invoked when the angle test

$$\theta = 57.295 \arccos [(\Delta_N^T v) / (\|\Delta_N\| \|v\|)] \geq \Theta \quad (57)$$

is satisfied, where v is the current estimate of the eigenvector associated with the smallest positive eigenvalue of H and Θ is 5 degrees.

Corrector Steps: Corrector steps are computed by solving the following nonlinear programming problem

$$g^T g \quad \text{subject to} \quad \xi^T \xi = L \quad (58)$$

where L denotes the current value of the objective function contour (or level set).

Second-Order Acceleration. Acceleration is needed to converge quickly to singular points and for this second-order Newton steps, given by

$$\Delta z_{N2} = -H^{-1} g \quad (59)$$

are used.

Eigenvalue–Eigenvector Decomposition. After a stationary point of the objective function

is found, the terrain method uses an eigen-decomposition of the Hessian matrix of the objective function to characterize the stationary point as a minimum, saddle point, or singular point and to decide what to do next. If H has all nonnegative eigenvalues, then the current stationary point is a minimum and uphill movement is indicated. Uphill movement from a local minimum or global minimum of the least-squares function always takes place along a direction of smallest positive curvature. If, on the other hand, H has one or more negative eigenvalues, then the current stationary point is a saddle point and downhill exploration of the objective function landscape is indicated next. Downhill movement from a saddle point to either a singular point of lower norm or a solution always takes place along an eigen-direction, v , of negative curvature.

However, it is important to understand that only a few eigenvalues and eigenvectors are needed. For these calculations we use the inverse power method together with incomplete factorization to compute a few eigenvalues and eigenvectors. Also we never actually form matrix products like $H^T H$ to avoid rounding errors.

Termination. Termination is based on the number of times the terrain method strikes the boundary of the feasible region. Collisions with a boundary of the feasible region are used to signal an end to the usefulness of exploration in a particular eigen-direction. LUCIA and YANG [20, 21] base their termination criterion on something they call *limited connectedness*. This means that stationary points are really only connected to neighboring stationary points along specific eigen-directions. It is assumed that the number of important connections between stationary points along valleys and ridges is limited to four or less and is related to dominant geometric distortions (i.e., +/- the smallest positive eigen-direction and +/- the most negative eigen-direction) caused by the strongest ‘attractions’ between neighboring stationary points. This makes it possible to dynamically catalogue connections in a set, C , and to conclude that all of the important connections between stationary points have been explored when C is empty. Thus termination occurs when C is the empty set.

3.1. Simple Example of the Global Terrain Method

A two-dimensional example of a nonlinear continuous stirred tank reactor (CSTR) taken from [23] is used to illustrate the basic features of the global terrain method. Figure 1 shows the contours of the least squares objective function and the stationary points and their connectedness for this example. There is a low lying valley that runs diagonally across the bottom of the figure.

Problem Statement. The equations that describe the behavior of the CSTR are the mass-balance equation

$$\xi_1(x, T) = 120x - 75k(1-x) = 0 \quad (60)$$

and the energy-balance equation

$$\xi_2(x, T) = -x(873-T) + 11(T-300) = 0 \quad (61)$$

where x denotes the conversion, T is the reactor temperature in Kelvin and k , the reaction rate constant, which is given by $k = 0.12 \exp [12581(T - 298)/(298T)]$. The bounds on conversion and temperature for this example are

$$0 \leq x \leq 1 \quad \text{and} \quad 300 \leq T \leq 400 \quad (62)$$

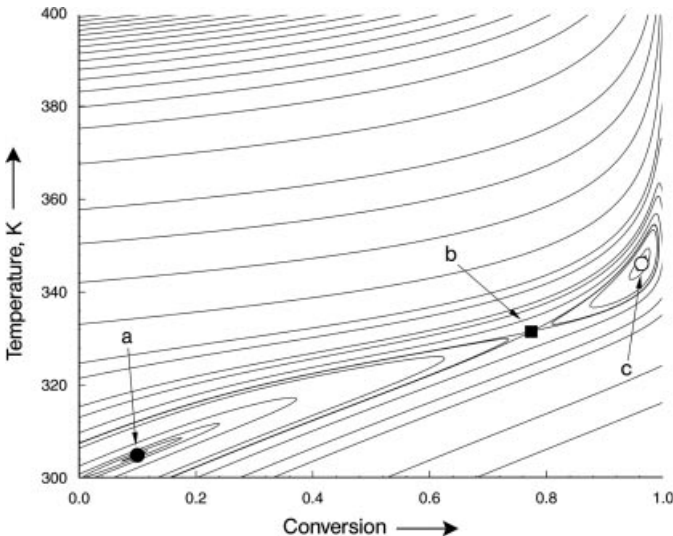


Figure 1. Connectedness of stationary points for a nonlinear CSTR
a) Local minimum; b) Saddle point; c) Solution

Problem and Solver Attributes. All first and second partial derivatives are calculated analytically. The equations are solved to an accuracy of $\xi^T \xi \leq 10^{-16}$. The maximum number of downhill steps for each downhill subproblem is set to 50. The initial trust region radius is $\Delta = 10$ and downhill Newton steps are adjusted automatically using the rule

$$\|\Delta z_N\| < 0.1\Delta, \quad \text{then} \quad \alpha_d = \min(1, 2\alpha_d) \quad (63)$$

where α_d is the downhill Newton step size. Otherwise, α_d remains the same. The step size for uphill Newton predictor steps is fixed at $\alpha_p = 0.05$ and the maximum number of uphill Newton predictor steps is 10. After 10 predictor steps, corrector steps are used to force iterates back to the valley if the angle test indicates drift. The maximum number of corrector steps per corrector subproblem is 50. Acceleration occurs if either the ratio of the Newton step to the gradient step is less than 10^{-6} or the Newton step reverses itself during uphill movement. The maximum number of acceleration steps is set to 50.

Problem Solution. Sufficient numerical details for all subproblems are given so that readers interested in implementing the global terrain method can validate their implementation of individual components (downhill trust region,

second-order acceleration, uphill predictor–corrector steps, eigenvalue–eigenvector computations, etc.) of the terrain method. The starting value for the unknown variables is $x = 0.2$ and $T = 328\text{ K}$.

Downhill Steps (First Set). The initial set of downhill steps for this example is shown in Table 2.

All steps in Table 2 are Newton steps and note that the value of $\xi^T \xi$ is monotonically decreasing.

Acceleration occurs after 20 downhill steps because the ratio of the Newton step to the gradient step becomes small. Table 3 shows the second-order Newton acceleration steps converge to a local minimum of $\xi^T \xi$. The local minimum is $z^* = (x, T) = (0.094595, 304.677\text{ K})$ and indicated by the fact that $g^T g \leq 10^{-8}$. Note $\xi^T \xi$ and thus ξ is not zero! However, $g^T g$ is approximately zero and this implies that $g = J^T \xi = 0$. Since $\xi \neq 0$, this means J^T and therefore J is singular.

The eigenvalues and eigenvectors of the Hessian matrix, H , at this local minimum are all positive and shown in Table 4.

Uphill Steps (First Set). Uphill step are initiated in the eigen-direction

Table 2. First set of downhill steps for CSTR example

Iteration	x	$T, \text{ K}$	$\xi^T \xi$
0	0.2	328.0	1.40874×10^5
1	0.214034	327.795	1.27280×10^5
2	0.239982	327.370	1.03586×10^5
3	0.283752	326.468	6.77424×10^4
4	0.341531	324.443	2.73817×10^4
5	0.350459	322.000	1.12087×10^4
6	0.331081	319.336	4.62678×10^3
7	0.296777	316.622	1.92990×10^3
8	0.254796	313.927	8.19848×10^2
9	0.207496	311.195	3.61077×10^2
10	0.151637	308.161	1.73688×10^2
11	0.063339	303.541	1.21009×10^2
12	0.063339	303.541	1.21009×10^2
13	0.068223	303.791	1.19814×10^2
14	0.073105	304.041	1.18948×10^2
15	0.077984	304.290	1.18417×10^2
16	0.079702	304.079	1.18313×10^2
17	0.080100	304.399	1.18295×10^2
18	0.080182	304.403	1.18292×10^2
19	0.080198	304.404	1.18291×10^2
20	0.080201	304.404	1.18291×10^2

Table 3. Second-order acceleration to local minimum of CSTR example

Iteration	x	$T, \text{ K}$	$\xi^T \xi$
0	0.080201	304.404	1.18291×10^2
1	0.081526	304.426	1.12931×10^2
2	0.083945	304.466	1.04701×10^2
3	0.087935	304.537	9.53293×10^1
4	0.093037	304.638	9.02564×10^1
5	0.094578	304.676	9.00427×10^1
6	0.094595	304.677	9.00427×10^1

Table 4. Eigenvalues and eigenvectors of H at local minimum

Eigenvalues	Eigenvectors
2.10713	(0.019607, 0.999808)
3.43487×10^5	(0.999808, -0.019607)

$\Delta z = (-0.019607, -0.999808)$ and strike the boundary of the feasible region at $z = (0.038036, 300\text{ K})$ in 6 iterations. Uphill steps from the initial stationary point are subsequently initiated in the eigen-direction $v = (-0.019607, -0.999808)$ and take 30 uphill predictors step followed by 9 corrector iterations to get back to the valley. This is then followed 5 uphill steps where Newton predictor steps go through a maximum in $\xi^T \xi$ and trigger acceleration again, this time to a saddle point.

The acceleration steps to the saddle point are shown in Table 5. Note that norm reduction cannot be imposed on acceleration steps. The saddle point is $z^* = (0.774548, 331.507\text{ K})$. The eigenvalues and eigenvectors of H at the saddle point are given in Table 6.

The negative eigenvalue of H indicates that the stationary point is a saddle point and that the

Table 5. Second-order acceleration to saddle point for CSTR example

Iteration	x	$T, \text{ K}$	$\xi^T \xi$
0	0.755706	331.709	5.71861×10^3
1	0.806247	334.907	9.89472×10^3
2	0.804901	334.682	9.55795×10^3
3	0.802153	334.253	9.02562×10^3
4	0.796517	333.474	8.38298×10^3
5	0.785435	332.254	7.99690×10^3
6	0.777929	331.711	7.98447×10^3
7	0.774669	331.512	7.98500×10^3
8	0.774548	331.507	7.98501×10^3
9	0.774548	331.507	7.98501×10^3

Table 6. Eigenvalues and eigenvectors of H at saddle point

Eigenvalues	Eigenvectors
-92.7333	(0.026170, 0.999658)
8.7427×10^5	(0.999658, -0.026170)

next move should be downhill from the saddle. Also because the location of the first stationary point is known, it is straightforward to determine that the correct movement downhill is in a direction away from the local minimum.

Downhill Steps (Second Set). Next the global terrain method initiates downhill movement in the direction $v = (0.026170, 0.999658)$ scaled by $1/\lambda$. Table 7 shows the steps of the second downhill subproblem.

All steps in Table 7 are Newton steps. Also note that $\xi^T \xi$ decreases monotonically and the asymptotic rate of convergence is quadratic to the global minimum

Table 7. Second set of downhill steps for CSTR example

Iteration	x	T, K	$\xi^T \xi$
0	0.785408	331.922	7.96902×10^3
1	0.842049	334.605	7.13659×10^3
2	0.866098	336.005	6.28044×10^3
3	0.880198	336.934	5.60146×10^3
4	0.890527	337.676	5.01492×10^3
5	0.898670	338.305	4.49846×10^3
6	0.905347	338.853	4.03999×10^3
7	0.910963	339.341	$3.63120E \times 10^3$
8	0.915770	339.781	3.26572×10^3
9	0.919942	340.180	2.93837×10^3
10	0.923599	340.546	2.64479×10^3
11	0.926833	340.882	2.38125×10^3
12	0.929713	341.193	2.14450×10^3
13	0.932293	341.481	1.93168×10^3
14	0.934615	341.749	1.74030×10^3
15	0.938816	342.248	1.40007×10^3
16	0.942257	342.682	1.12769×10^3
17	0.945116	343.062	9.09132×10^2
18	0.947518	343.396	7.33456×10^2
19	0.949555	343.691	5.92067×10^2
20	0.951295	343.952	4.78159×10^2
21	0.954287	344.416	3.02492×10^2
22	0.956483	344.778	1.91955×10^2
23	0.958128	345.063	1.22071×10^2
24	0.950378	345.288	7.77474×10^1
25	0.961299	345.644	2.74419×10^1
26	0.962368	345.854	9.77472×10^0
27	0.963596	346.103	3.66418×10^{-1}
28	0.963869	346.164	1.10322×10^{-5}
29	0.963868	346.164	4.01405×10^{-16}
30	0.963868	346.164	1.23355×10^{-25}

Table 8. Eigenvalues and eigenvectors of H at the global minimum

Eigenvalues	Eigenvectors
98.2769	(0.0041241, 0.999991)
1.13079×10^7	(0.999991, -0.0041241)

$z^* = (0.963868, 346.164 K)$. The eigenvalues and eigenvectors of H at the global minimum are shown in Table 8.

The eigenvalues in Table 8 show that the last entry in Table 7 is a global minimum of $\xi^T \xi$ and that the next exploration should be uphill exploration of the least squares objective function surface.

Uphill Steps (Second Set). Uphill movement from the global minimum is initiated in the eigen-direction $v = (0.0041241, 0.999991)$. The global terrain uses an additional 291 function and gradient evaluations, taking 10 uphill predictor steps and then correcting along the valley before it strikes the boundary at $z = (0.999752, 400.0 K)$ and terminates because it has encountered the boundary twice.

Table 9 gives a summary of the computational work needed by the global terrain method on the example problem measured in terms of iterations and function and gradient evaluations for the downhill, uphill (predictor and corrector steps), acceleration, and eigenvalue-eigenvector computations. Also shown is the number of function and gradient evaluations need to go from a stationary point to a boundary of the feasible region.

The time required for all of the computations shown in Table 9, including input and output, is 0.16 s on a 3.6 GHz Dell High Precision Workstation using the Lahey-Fijitsu LF95 compiler.

3.2. General Optimization Problems

Up to this point the discussion has focused on nonlinear least squares problems. To use the global terrain method for any general objective function, ξ , it is only necessary to recognize that the equations to be solved become

$$g(x) = 0 \tag{64}$$

and therefore ξ should be replaced with g , J of ξ with H of ξ , and $\nabla^2 \xi_i$ with $\nabla^2 G_i$ in the

Table 9. Summary of computational work for global terrain method on example problem

Stationary point	Value x, T, K	Iterations (function and gradient evaluations)				
		downhill	predictor	corrector	acceleration	eigens
Local minimum	(0.094595, 304.677)	20 (20)	n.a.	n.a.	6 (6)	8 (8)
Saddle	(0.774548, 331.507)	n.a.	35 (35)	9 (36)	9 (9)	8 (8)
Global minimum	(0.963868, 346.164)	30 (30)	n.a.	n.a.	n.a.	8 (8)
Boundary		6 (6)	164	21 (119)	n.a.	8 (8)

n.a. = not applicable.

right-hand sides of all equations presented [24]. Finite difference derivatives can be used in place of analytical derivatives and often lead to successful convergence.

3.3. Advanced Techniques

Advanced techniques such as those that handle integral path bifurcations and nondifferentiability of the objective function can be found in [24, 25]. Integral path bifurcations can be tangent or pitchfork bifurcations and techniques for dealing with this behavior monitor and exploit Gauss curvature. Nondifferentiability of the objective function can occur in process engineering problems in which model switching can occur (e.g., in phase stability computations where different phases are modeled using different phase models).

3.4. Applications and Extensions of the Terrain Method

There are many process engineering examples described in [24, 25] which include finding

1. The glass temperature of a polymer mixture
2. Equilibrium structures to nanostructured materials
3. Roots to the SAFT equation
4. All azeotropes of a binary mixture
5. Equilibrium phase compositions and pressure for a retrograde flash
6. The temperature and conversion for nonadiabatic CSTRs with isola
7. All solutions to heterogeneous distillation problems
8. Phase split and phase stability calculations nonideal mixtures
9. Molecular conformation of small molecules

More recently LUCIA et al. [26] have extended the global terrain method by combining it with logarithmic barrier functions to solve heat and mass transfer in a catalyst pellet modeled using collocation over finite elements.

4. Optimization under Uncertainty in Process Systems Engineering

The deterministic optimization literature classifies problems as linear programming (LP), nonlinear programming (NLP), integer programming (IP), mixed integer LP (MILP), and mixed integer NLP (MINLP), depending on the decision variables, objectives, and constraints → Mathematics in Chemical Engineering, Section 10.1. However, the future cannot be perfectly forecast but instead should be considered random or uncertain. Optimization under uncertainty refers to this branch of optimization where there are uncertainties involved in the data or the model, and is popularly known as *stochastic programming* or *stochastic optimization problems*. In this terminology, stochastic refers to the randomness, and programming refers to the mathematical programming techniques such as LP, NLP, IP, MILP, and MINLP. In discrete (IP, MILP, MINLP) optimization, there are probabilistic techniques like *simulated annealing* and *genetic algorithms*; these techniques are sometimes referred to as the *stochastic optimization techniques* because of the probabilistic nature of the method. In general, however, stochastic programming and stochastic optimization involve optimal decision making under uncertainties. Parametric programming which is a method based on sensitivity analysis also involves consideration of uncertainties and provides how optimal decisions and designs vary with uncertainties.

Table 10. Uncertainties in demand

j	Demand, d_j	Probability, P_j
1	50	5/7
2	100	1/7
3	140	1/7

The need for including uncertainty in complex decision models arose early in the history of mathematical programming. The first model forms, involving action followed by observation and reaction (or recourse), appeared in the 1950s [27, 28]. The commonly used example of a recourse problem is the news vendor problem described below [29]. The news vendor or news boy problem has a rich history that has been traced back to the economist EDGEWORTH [30], who applied a variance to a bank cash-flow, problem. However, it was not until the 1950s that this problem, like many other OR/MS models seeded by the war effort, became a topic of serious and extensive study by academicians [31].

Example: The simplest form of a stochastic program may be the *news vendor* (also known as the newsboy) problem. In the news vendor problem, the vendor must determine how many papers (x) to buy now at the cost of c cents for a demand which is uncertain. The selling price is sp cents per paper. For a specific problem, whose weekly demand is shown below, the cost of each paper is $c = 20$ cents and the selling price is $sp = 25$ cents. Suppose we need to solve the problem assuming the news vendor knows the demand uncertainties (Table 10) but does not know the demand curve for the coming week (Table 11) a priori. Assume no salvage value $s = 0$, so that any papers bought in excess of demand are simply discarded with no return.

Table 11. Weekly demand

Day	Demand
Monday	50
Tuesday	50
Wednesday	50
Thursday	50
Friday	50
Saturday	100
Sunday	140

Solution: In this problem, the question is how many papers the vendor must buy (x) to maximize the profit. Let r be the effective sales and w be the excess which is going to be thrown away. This problem falls under the category of stochastic programming with recourse where there is action (x), followed by observation (profit), and reaction (or recourse) (r and w). The first idea to solve this problem is to find the average demand and the optimal supply x corresponding to this demand. Since the average demand from the Table 10 is 70 papers, $x = 70$ should be the solution. However, with this solution where supply is 70 papers per day, the news vendor will be making a loss of 50 cents per week. This probably is not the optimal solution. Can we do better? For that one needs to propagate the uncertainty in the demand to see the effect of uncertainty on the objective function and then find the optimum value of x . The above information can be transformed for daily profit as follows.

$$\begin{aligned} \text{Profit} &= -cx + \frac{5}{7}sp d_1 + \frac{1}{7}s_p x + \frac{1}{7}sp x = -cx + \frac{5}{7}sp d_1 + \frac{2}{7}sp x \\ &\text{if } d_1 \leq x \leq d_2 \end{aligned} \quad (65)$$

or

$$\text{Profit} = -cx + \frac{5}{7}sp d_1 + \frac{1}{7}sp d_2 + \frac{1}{7}sp x \quad \text{if } d_2 \leq x \leq d_3 \quad (66)$$

Notice that the problem represents two equations for the objective function (Eqs. 65 and 66); this results in an objective function that is a discontinuous function and is therefore no longer a LP. The optimal solution to this problem is $x = d_1 = 50$ providing the news vendor with an optimum profit of 1750 cents per week.

The difference between taking the average value of the uncertain variable as the solution as compared to using stochastic analysis (propagating the uncertainties through the model and finding the effect on the objective function as above) is defined as the *value of stochastic solution, VSS*. If we take the average value of the demand, i.e., $x = 70$ as the solution, we obtain a loss of 50 cents per week, therefore, the value of stochastic solution, VSS, is $1750 - (-50) = 1800$ cents per week.

Now consider the case, where the vendor knows the exact demand (Table 11) a priori. This is the perfect information problem where

Table 12. Supply and profit

Supply	Profit
50	250
50	250
50	250
50	250
50	250
100	500
140	700

the solution x_i for each day i has to be found. The problem could be described in terms of x_i .

$$\underset{x_i}{\text{Maximize}} \text{ profit} = -cx_i + \text{sales}(r, w, d_i)$$

subject to

$$\text{sales}(r, w, d_i) = sp r_i + sw_i$$

$$r_i = \min(x_i, d_i) = \begin{cases} x_i, & \text{if } x_i \leq d_i \\ d_i, & \text{if } x_i > d_i \end{cases}$$

$$w_i = \max(x_i - d_i, 0) = \begin{cases} 0, & \text{if } x_i \leq d_i \\ x_i - d_i, & \text{if } x_i > d_i \end{cases} \quad (67)$$

Here each problem (for each i) need to be solved separately, leading to the following decisions shown in Table 12.

One can see that the difference between the two values, (1) when the news vendor has the perfect information (2450 cents per week) and (2) when he does not have the perfect information (1750 cents per week) but can represent it using probabilistic functions, is the expected value of perfect information, EVPI. The latter is 700 cents per week for this problem.

Maximizing expected profit given that there are uncertainties associated with demand and supply is a classic problem in any manufacturing industry including the chemical industry [29]. However, decision making in chemical manufacturing is not restricted to just supply and demand cost minimization problems but involves models that are linear, nonlinear, and mixed integer in nature.

4.1. Methods and Algorithms

The literature on optimization under uncertainties very often divides the problems into

categories such as “wait and see”, “here and now”, and “chance constrained optimization” [32, 33]. In “wait and see” one waits until an observation is made on the random elements, and then solves the (deterministic problem). This is the “wait and see” problem of MADANSKY [34], originally called “stochastic programming” [35], is not in a sense one of decision analysis. In decision making, the decisions have to be made “here and now” about the activity levels. The “here and now” problem involves optimization over some probabilistic measure — usually the expected value. By this definition, chance constrained optimization problems can be included in this particular category of optimization under uncertainty. Chance constrained optimization involves constraints which are not expected to be always satisfied; only in a proportion of cases, or “with given probabilities”. “Parametric programming” is based on sensitivity analysis of optimal solution and is similar to the “wait and see” problems. These various categories require different methods for obtaining their solutions. It should be noted that many problems have both here and now, and wait and see problems embedded in them.

Here and Now Problems. The “here and now” problems require that the objective function and constraints be expressed in terms of some probabilistic representation (e.g., expected value, variance, fractiles, or most likely values). For example, in chance constrained programming, the objective function is expressed in terms of expected value, while the constraints are expressed in terms of fractiles (probability of constraint violation), and in Taguchi’s off-line quality control method [36, 37], the objective is to minimize variance. These problems can be classified as “here and now” problems.

Is it possible to propagate the uncertainty using moments of input uncertainties (such as mean and variance) thereby obtaining a deterministic representation of the problem? This is the basis of the chance constrained programming method, developed very early in the history of optimization under uncertainty, principally by CHARNES and COOPER [38]. In the chance constrained programming (CCP) method, some of the constraints likely need not hold as was assumed in earlier problems. CCPs can be

represented as follows:

$$\text{Optimize } \xi(x, u) = P_1(j(x, u)) = E(z(x, u)) \quad (68)$$

$$P(g(x) \leq u) \leq a \quad (69)$$

In the above formulation, Equation (69) is the chance constraint. In chance constraint formulation, this constraint (or constraints) is (are) converted into a deterministic equivalent under the assumption that the distribution of the uncertain variables, u , is a stable distribution [39]. Normal, Cauchy, Uniform, and Chi-square are all stable distributions that allow the conversion of probabilistic constraints into deterministic ones. The deterministic constraints are in terms of moments of the uncertain variable u (input uncertainties).

MARANAS et al. [40, 41] have used chance constraint programming to solve (1) a chemical synthesis problem of polymer design and (2) a metabolic pathways synthesis problem in biochemical reaction engineering.

Wait and See Problems. In contrast to “here and now” problems, which yield optimal solutions that achieve a given level of confidence, “wait and see” problems involve a category of formulations that shows the effect of uncertainty on optimum design. A “wait and see” problem involves deterministic optimal decisions at each scenario or random sample, equivalent to solving several deterministic optimization problems. Parametric programming is one way to solve the “wait and see” problems [42, 43].

The difference between the “here and now” and “wait and see” solutions is the EVPI. The concept of EVPI was first developed in the context of decision analysis and can be found in classical references [44]. In the above problem, there was action (x), followed by observation (profit), and reaction (or recourse) (r and w). Recourse problems with multiple stages (similar to the multiperiod problems in chemical engineering) involve decisions that are taken before the uncertainty realization (here and now) and recourse actions which can be taken when information is disclosed (wait and see). These problems can be solved using decomposition methods.

As can be seen from the above description, both “here and now” and “wait and see” problems require the representation of uncertainties

in the probabilistic space and then the propagation of these uncertainties through the model to obtain the probabilistic representation of the output.

4.2. Uncertainty Analysis and Sampling

The probabilistic or stochastic modeling [45] iterative procedure involves:

1. Uncertainty quantification which involves specifying the uncertainty in key input parameters in terms of probability distributions
2. Sampling the distribution of the specified parameter in an iterative fashion
3. Propagating the effects of uncertainties through the model and applying statistical techniques to analyze the results

Uncertainty Characterization and Quantification. In general, uncertainties can be characterized and quantified in terms of probabilistic distributions. The type of distribution chosen for an uncertain variable reflects the amount of information that is available. For example, the uniform and log-uniform distributions represent an equal likelihood of a value lying anywhere within a specified range, on either a linear or logarithmic scale, respectively. Further, a normal (Gaussian) distribution reflects a symmetric but varying probability of a parameter value being above or below the mean value. In contrast, lognormal and some triangular distributions are skewed such that there is a higher probability of values lying on one side of the median than the other. A beta distribution provides a wide range of shapes and is a very flexible means of representing variability over a fixed range. Modified forms of these distributions, uniform* and log-uniform*, allow several intervals of the range to be distinguished. Finally, in some special cases, user-specified distributions can be used to represent any arbitrary characterization of uncertainty, including chance distribution (i.e., fixed probabilities of discrete values).

It is easier to assume the upper and lower bound of uncertain variables and, hence, uniform distribution is the first step towards uncertainty quantification. Most of the papers in

chemical engineering use this simplistic approach and use upper and lower bounds of uncertain variables. Few studies [46–48] identified most likely values and use triangular distributions. Extensive data which are obtained from DECHEMA and represent realistic quantification of uncertainties related to UNIFAC parameters are used by GANI and DIWEKAR [49, 50].

Interval methods are also proposed [52–54] for handling uncertainties where there is no information about uncertainties. For interval methods, interval mathematics is used to propagate the uncertainty through the model. For this purpose, the use of probability boxes (P-box methods) instead of just intervals is proposed [54]. However, these methods are not yet applicable to large scale or black box models.

Sampling Techniques. One of the most widely used techniques for sampling from a probability distribution is the *Monte Carlo sampling technique*, which is based on a pseudo-random generator used to approximate a uniform distribution (i.e., having equal probability in the range from 0 to 1). The specific values for each input variable are selected by inverse transformation over the cumulative probability distribution. Crude Monte Carlo methods can result in large error bounds (confidence intervals) and variance. Variance reduction techniques are statistical procedures designed to reduce the variance in the Monte Carlo estimates [55]. Importance sampling, latin hypercube sampling (LHS, [56, 57]), descriptive sampling [58], and Hammersley sequence sampling (HSS) [59–61] are examples of variance reduction technique.

Importance Sampling. In importance Monte Carlo sampling, the goal is to replace a sample using the distribution of u with one that uses an alternative distribution that places more weight in the areas of importance. Obviously such a distribution function is problem dependent and is difficult to find. One of the examples of importance sampling in chemical engineering is the Metropolis criterion used in molecular simulations [62].

The following two sampling methods provide a generalized approach to improve the computational efficiency of sampling.

Latin Hypercube Sampling. The main advantage of Monte Carlo methods lies in the fact that the results from any Monte Carlo simulation can be treated using classical statistical methods; thus results can be presented in the form of histograms, and methods of statistical estimation and inference are applicable. Nevertheless, in most applications, the actual relationship between successive points in a sample has no physical significance; hence, the randomness/independence for approximating a uniform distribution is not critical [63]. Moreover, the error of approximating a distribution by a finite sample depends on the equidistribution properties of the sample used for $U(0,1)$ rather than its randomness. Once it is apparent that the uniformity properties are central to the design of sampling techniques, constrained or stratified sampling techniques become appealing [64].

LHS [56, 57, 65] is one form of stratified sampling that can yield more precise estimates of the distribution function. In LHS, the range of each uncertain parameter X_i is subdivided into nonoverlapping intervals of equal probability. One value from each interval is selected at random with respect to the probability distribution in the interval. The n values thus obtained for X_1 are paired in a random manner (i.e., equally likely combinations) with n values of X_2 . These values are then combined with values of X_3 to form n -triplets, and so on, until n k -tuplets are formed. In median LHS (MLHS) this value is chosen as the mid-point of the interval. MLHS is similar to the descriptive sampling described by [58]. In chemical engineering LHS is used in early stochastic modeling and optimization frameworks [45, 47, 66].

The main drawback of this stratification scheme is that, it is uniform in one dimension and does not provide uniformity properties in k -dimensions. Sampling based on quadrature, cubature techniques [67], or collocation techniques [68] face similar drawback. These sampling techniques perform better for lower dimensional uncertainties. Therefore, many of these sampling techniques use correlations to transform the integral into one or two dimensions. However, this transformation is possible only for limited distribution functions when the uncertain variables are tightly correlated. For highly correlated samples similar to what has been observed in thermodynamic phase

equilibria, a sampling technique based on confidence region estimates [69] can be used.

Hammersley Sequence Sampling. HSS, based on Hammersley points, was developed by [59, 60]; the technique uses an optimal design scheme for placing the n points on a k -dimensional hypercube. This scheme ensures that the sample set is more representative of the population, showing uniformity properties in multidimensions, unlike Monte Carlo, LHS, and MLHS techniques. It is clearly observed that HSS shows greater uniformity than other stratified techniques such as LHS, which are uniform only along a single dimension and do not guarantee a homogeneous distribution of points over the multivariate probability space. Similar behavior is observed for correlated samples also. The implementation of correlation structures is based on the use of rank correlations [57, 65]. Some of the new variants of HSS include the HSS2 and latin hypercube Hammersley sampling (LHHS) [61, 70, 71].

4.3. Optimization Algorithms

Numerical optimization techniques constitute a fundamental part of theoretical and practical science and engineering as can be seen from the number of papers published each year in literature. Although in real world systems uncertainties cannot be ignored, the literature on optimization under uncertainty methods and/or applications is sparse as compared to any other optimization area. This can be attributed to the fact that in optimization under uncertainty area, one has to pay attention to uncertainty analysis as well as optimization.

Decomposition Algorithms. Even a linear problem in optimization under uncertainty (e.g., the news boy problem) results in nonlinearities (discontinuous function) due to the probabilistic functional. A decomposition method is a common approach to solve such problems. In decomposition approach, the problem is decomposed into a master problem and a subproblem. The master problem is a simplified (usually) linear approximation of the complete problem and this approximation is derived from a number of subproblems. First proposed for mixed

integer problems, Bender's decomposition [72] forms the basis for decomposition algorithms proposed for deterministic mixed integer programming and stochastic linear as well as mixed integer programming problems [73, 74]. The two main algorithms commonly used in the stochastic programming literature for stochastic linear (and mixed integer) programming with fixed recourse are the L-shaped [75–77] and the stochastic decomposition methods [78, 79]. The L-shaped method is used when the uncertainties are described by discrete distribution. On the other hand, the stochastic decomposition method uses sampling when random variables are represented by continuous distribution functions. Although chemical engineering applications involve a large number of stochastic linear programming (and stochastic MILP) problems, researchers in chemical engineering have not exploited these generalized algorithms. On the other hand, they have used specific structure of a particular problem (e.g., flexibility) to derive different decomposition schemes and/or bounds [80–83]. The domain (flexibility) specific nature of the problem restricts the applicability of these methods.

Stochastic Nonlinear Programming. Since engineering models are usually nonlinear, and uncertainties are not restricted to uniform or normal distributions, the algorithms and approaches presented above have restrictions that limit their application potential to large-scale problems in the various domains of chemical engineering. In order to develop a more general approach the better optimization of nonlinear uncertain systems (BONUS) algorithm was proposed [84, 85], which uses sampling and replaces the inner model evaluation loop with a reweighting scheme. Deterministic NLP problems use quasi-Newton's methods and require calculation of derivatives with respect to each decision variable, at each optimization iteration. These methods are widely used and the codes are made robust over the years. BONUS extends these algorithms to stochastic NLP problems. BONUS provides significant reductions in model iterations.

Discrete Optimization under Uncertainty. Many chemical engineering applications involve discrete decision variables (mixed integer

problems). Although decomposition methods have been used to solve these problems, the applicability of these algorithms to solve problems of real world scale is very limited in [86]. This is mainly due to the fact that most of these methods rely heavily on convexity conditions and simplified approximations of the probabilistic functions containing uncertainties. Variants of probabilistic methods like the stochastic annealing algorithm [87] show great promise in this regard. In the stochastic annealing algorithm, the optimizer not only obtains the decision variables but also the number of samples required for the stochastic model. Furthermore, it provides the trade-off between accuracy and efficiency by selecting an increased number of samples as one approaches the optimum. Stochastic annealing and its recent variants have been used to solve real world problems such as:

- Nuclear waste-management problem [88]
- Computer-aided molecular design, and polymer design under uncertainty [89]
- Greener solvent selection [50]
- Methylene chloride process synthesis [90]

New variants of genetic algorithms based on the theory used in stochastic annealing, were developed and applied to solvent selection and recycling problems [91–93].

Parametric programming is based on the sensitivity analysis theory, distinguishing from the latter in the targets. Sensitivity analysis provides solutions in the neighborhood of the nominal value of the varying parameters, whereas parametric programming provides a complete map of the optimal solution in the space of the varying parameters. Theory and algorithms for the solution of a wide range of parametric programming problems have been reported in the literature [43].

4.4. Applications

Problems in chemical engineering are linked to the life cycle of the process that extends from raw material selection (chemical synthesis), process development (process synthesis and design) through the planning and management

of the production process. Process design starts with chemical synthesis where the chemical pathway from reactants to the product is defined at the laboratory scale. Process synthesis translates the chemical synthesis to a chemical process. It involves decisions about process unit operations and connections.

In general, process design activities start at this level and process simulation is the last step in computer-aided process design which predicts the behavior of the process if it was constructed. Incorporation of pollution prevention, operability, and other concepts into design and development at the initial stages lead to processes that are less cost intensive, thereby reducing the technical and economic risks. Therefore, process synthesis remains an important step in analyzing and designing environmentally benign processes. Furthermore, researchers have realized the importance of including the chemical synthesis in the process design, and methods for solvent selection, molecular design are gaining importance [50]. Most of the current systems analysis approaches focus on the simulation step. This is because several simulation programs (like the ASPEN simulator [94]) and models are available to describe these basic phenomena. However, this is the last step in decision making, as it predicts the behavior of the plant (or strategy) if it was constructed (or implemented). As the envelope extends to include process synthesis, chemical synthesis, or management and planning considerations, the ability to include uncertainties become important. Unlike the traditional process design where engineers are looking for low cost options, objectives of optimization in greener and robust design is not restricted to maximizing profit but includes several other criteria like reliability, flexibility, operability, controllability, environmental and ecological impacts, safety, and quality need to be considered at the different stages of analysis.

Stochastic Objectives and Constraints The formulation of objective function is one of the crucial steps in the application of optimization to a practical problem [95].

The earliest work related optimization under uncertainty in chemical engineering used the “here and now” formulation where expected value of cost is used as the objective function for

a given level of risk [96–98]. Soon researchers realized the importance of including operability considerations like feasibility and flexibility [99–102] in the optimal design problem. Flexibility is concerned with the problem of ensuring feasible steady-state operation over a variety of operating conditions. On the other hand, reliability is concerned with the probability of normal operation given that failures can occur, while safety is concerned with hazards that are consequence of failure. The other aspects of operability are controllability which deals with the concept of quality and stability of the dynamic response of the system.

Earlier approaches to obtaining flexible design dealt with finding overdesign factors [98]. By introducing the concept of *flexibility index*, a well-structured optimization problem that could be easily converted to a mini–max deterministic problem, was provided [102]. Flexibility index is defined as a scalar matrix whose value for any fixed design characterizes is equal to the size of the region of feasible operation in the space of uncertain parameters. The basic assumption behind the definition of flexibility index is that the uncertainty parameters vary within upper and lower bounds with equal probability (equivalent to uniform distribution). These parameters are assumed to vary independently. Geometrically, this approach corresponds to inscribing within the feasible region a hyper-rectangle which is centered at the nominal point. The size of the feasible region is then characterized by the lengths of the sides of the rectangle, which in turn define the lower and upper bounds of the parameter. This hyper-rectangle then defines the actual ranges for each uncertain parameter over which the feasible operation can be guaranteed. The maximum hyper-rectangle that can expand around the nominal parameter point touches the boundary at a vertex of the hyper-rectangle. This concept was extended further to stochastic flexibility expressing it in terms of probability of feasible operation. The design for “optimal flexibility” falls under the category of “here and now” problem. On the other hand, the flexibility index or stochastic flexibility with multiperiod optimization problems involve “wait and see” decisions and solution of a deterministic optimization problem for each scenario so that one gets a probabilistic representation of optimal solutions.

Many researchers use flexibility synonymous to uncertainty analysis. For example, solution techniques for optimization under uncertainty was categorized into three categories by [42]:

1. Multiperiod
2. Stochastic programming
3. Parametric programming [103]

It should be noted that these categories are related to the optimal flexibility and flexibility index problems and cannot be generalized for all optimization under uncertainty problems in engineering or operations research. Further, the flexibility is one of the concepts in this field and is still an abstract measure [104]. The problem of flexibility is a well structured problem amenable to solutions using deterministic approaches with decomposition [80–82] or using new approximation to derive the probabilistic functions describing the effect of uncertainties [105]. Further, the concept of flexibility is also extended to include reliability and controllability in a similar fashion, demanding similar approaches. However, controllability also deals with the concept of quality control.

The aim of a control system is to keep the process output specifications on target, despite changes in the process input. In such an approach, the control engineer is often presented with difficult control problems that may require extensive and expensive modifications to both process and control system hardware to obtain satisfactory performance of the control system. Furthermore, the effectiveness of the control system is highly dependent upon the nominal values of the operating variables and the mechanical design which are set by the designer of the processing unit. Parameter design methodology is an off-line quality control method, popularized by TAGUCHI [36], for designing products and processes that are robust to uncontrollable variation at the design stage. In parameter design the objective is to find settings of the product or process design parameters which minimize an average quadratic loss function defined as the average standard deviation of the response from a target value. The rapid growth of interest in the Taguchi approach over the last few years led to a great expansion in the number of published case studies relating to

different areas of industrial activities [106]. Although the popularity of the Taguchi approach seems to be pervasive in all engineering branches; application of this procedure to chemical industries was not widely reported until 1990. BOUDRIGA [107] presented one of the first systematic studies of using different statistical approaches to the problem of off-line quality control for chemical processes. DIWEKAR and RUBIN [37] posed this problem as a stochastic optimization problem where the objective is to minimize output variance from the specified nominal value. This approach is the basis of robust design. A similar approach (objective function) was used for dynamic models [108]. However, this approach is derived from the financial literature using the mean-variance optimization models for portfolio optimization [109].

In view of growing environmental concerns, there is a critical need for designing large chemical processes with environmental considerations. Earlier design under uncertainty studies [46, 47] focused on design for the environment, included environmental considerations as probabilistic constraints in terms of risk of exceeding the specified emission limits. These papers considered the end-of-pipe treatments like new environmental control technology designs. Nowadays, industries are practicing the art of pollution prevention, which involves fundamental changes in the processes to minimize the formation of pollutants, as opposed to pollution control, involving end-of-pipe treatment of process emissions. This also means instead of including environmental considerations as constraints, they should be included as objectives like minimum environmental impacts. Current methodologies, such as the generalized waste reduction algorithm (WAR), provide a first step towards evaluating impacts such as ozone depletion, global warming and acid rain potentials [110, 111]. However, environmental impacts must also be weighted and balanced against other concerns, such as their cost and long-term sustainability. These multiple, often conflicting, goals pose a challenging and complex optimization problem, requiring multi-objective optimization under uncertainty [90, 93, 94, 110, 112, 113]. Environmental considerations are also translated into specific objectives for problems like expected value of

solvent selectivity in the solvent selection problem [49, 96], and minimizing expected value of “glass” formed in vitrification of nuclear waste problems [88].

Chemical Synthesis (→ Process Systems Engineering 4. Process and Product Synthesis, Design, Analysis, Chap. 6) Chemical synthesis involves search for molecules possessing desired physical, chemical, biological, and health properties for easy manufacture of products. Group contribution methods rely on experimental data and theoretic formulations to assign numerical values to chemical groups, which form the basis building blocks for computer-aided chemical synthesis. By combining these building blocks, it is possible to determine a wide range of characteristics for any given chemical. The reverse approach, computer-aided molecular design (CAMD) → Molecular Modeling uses group contribution techniques to determine physical characteristics by generating test molecules using primary building blocks. CAMD uses a set of groups as a starting point. These groups are uniquely designed to generate all possible molecules by exploring all combinations. The properties of each group and/or the interaction parameters between groups can be theoretically calculated, experimentally obtained, or statistically regressed. There are three main CAMD approaches: generation-and-test, mathematical optimization, and combinatorial optimization approaches. All methodologies for CAMD are exposed to uncertainties that arise from experimental errors, imperfect theories or models and their parameters, improper knowledge, or ignorance of systems. In addition, available group parameters may not be present, and current group contribution models (GCM) cannot estimate all necessary properties.

A technique for designing polymers under uncertainty was developed by [40]. This approach uses MINLP formulations, where the chance constraints represent the probability of meeting the target values. Following traditional chance constrained programming techniques, a deterministic equivalent is obtained. Two formulations are presented: (1) stochastic property matching identifies molecules that meet all characteristics with a given probability a , (2) stochastic property optimization determines the molecules that have a maximum value for a

given property with probability a while all other probabilities are met with given probability. A generalized framework to solve the same polymer design problem based on HSS sampling and stochastic annealing was presented [89]. The framework allowed them to study:

- Impact of uncertainties
- Effect of various probability distributions including stable and nonstable distributions (as CCP does not allow nonstable distributions), mixed distributions, and
- Different forms of objective functions

The framework presented a set of solutions to choose from instead of a single optimal solution, providing flexibility to the designer.

SINHA et al. [115] proposed that locally optimal solutions to the traditional CAMD problem are potential sources of uncertainties. To bypass this, a global optimization algorithm is presented. The CAMD is modeled as a MILP, and a case study is presented. Solvent selection using a CAMD approach that selects environmentally benign molecules through group combination tools have been analyzed by [49, 50, 91]. This is the first study where the uncertainties in group contribution methods are systematically characterized using available literature and experimental data. New variants of stochastic annealing are developed and used to find environmentally benign solvents. They also coupled process synthesis along with chemical synthesis to obtain environmentally friendly and cost effective solutions [92, 116].

***Process Synthesis and Design* ((→ *Process Systems Engineering 4. Process and Product Synthesis, Design, Analysis, Chap. 2*))**

Process synthesis translates the chemical synthesis to a chemical process. It involves decisions about process unit operations and connections. In general, process design activities start at this level and process simulation, which is shown to be the last step in computer-aided process design, predicts the behavior of the process if it was constructed.

One of the main goals in synthesis problems is to establish methodologies for selecting optimal flowsheet configurations. Approaches to process synthesis problems essentially fall under the following areas:

1. Thermodynamic approach
2. Evolutionary methods
3. Hierarchical approach, based on intuition and engineering judgment
4. Optimization approach based on mathematical programming techniques

The optimization approach to process synthesis involves (a) formulation of a conceptual flowsheet incorporating all the alternative process configurations (superstructure) and (b) identification of an optimal design configuration based on optimal structural topology and the optimal parameter level settings for a system to meet specified performance and cost objectives. Once the superstructure is known, combinatorial optimization methods like MINLP algorithms can be used to solve the synthesis problem. The first step in the solution of the process synthesis problem is to develop the superstructure containing all alternative designs to be considered for the optimal solution. The design of new processes is, however, complicated by the fact that technical and economic uncertainties arise, which lead to uncertainties in the prediction of plant performance and overall plant economics. An example where such technical and economic uncertainties occur and are not treated or characterized rigorously, is in the design of integrated environmental control processes for advanced power systems [45]. Since the conceptual design of any chemical process involves the identification of possible flowsheet configurations, design methods must also address the issues of process synthesis under uncertainty, as it has important implications on process viability, and other quality measures such as controllability, safety, and environmental compliance.

The literature in the area of process synthesis and process design under uncertainty have been concentrated on two focused application areas:

1. Pollution prevention by design
2. Designing for flexibility

Pollution Prevention by Design. The earlier papers in design under uncertainty with pollution prevention focus dealt with integrated environmental control systems for coal-based power systems. The work continued and extended to address synthesis problems in this

area [45–47]. Nuclear waste management posed a very hard synthesis problem [117]. This is a large-scale, real world problem related to the environmental restoration of Hanford site. Conversion of high-level radioactive waste into glass is crucial to the disposal of toxic waste dumps generated over 40–50 years at the Hanford nuclear waste site. The procedure essentially consists of mixing the sources of wastes into blends, to which appropriate glass formers (frit) is added to make glass. The objective is to maximize the amount of wastes per glass log, by keeping the amount of frit added to a minimum. Processibility and durability conditions require that certain restrictions on crystallinity, solubility and glass properties are met. Increasing the number of wastes, increases the combinatorial size of the problem. The combinatorial, non-convex nature of the problem was hard to solve even for the deterministic optimization methods. Uncertainties associated with the tank contents and models caused further problems and demanded new algorithms [88]. The new stochastic annealing algorithm provided optimal and robust solution to this problem in the face of uncertainties with reasonable computational time. A multiobjective extension of this problem to include policy aspect was possible due to these new algorithms. This new algorithm was also used for methylene chloride process synthesis [90].

Designing for Flexibility (→ Process Systems Engineering 4. Process and Product Synthesis, Design, Analysis, Chap. 3; → Process Systems Engineering, 5. Process Dynamics, Control, Monitoring, and Identification). Process flexibility is an area that received significant attention, as it ensures that processes are operational and safe when exposed to variations in operating conditions. The studies include distillation network design, heat exchanger network synthesis [82], reactor network synthesis [105], and batch processing plant design and operation for waste treatment [118].

Management, Scheduling, and Planning (→ Process Systems Engineering 8. Plant Operation, Integration, Planning, Scheduling and Supply Chain) Most problems in management, scheduling, and planning include

combinatorics (discrete choices and decisions) and uncertainties [104, 119]. These problems belong to batch processing due to time dependent nature of these chemical processes. Batch processing is generally used in high-value added, low-volume speciality chemicals and pharmaceuticals. Uncertainties abound in batch plant operation. Although batch processing is faced with all kinds of uncertainties, most of the literature in this area deals with demand uncertainties.

In general the scheduling problem is to determine a time-based assignment of tasks to equipment so that no process constraints are violated. The problems in the domain of process scheduling and planning can be conveniently described by resource–task–equipment network. The process design and retrofit problems add longer time horizons to the scheduling problems and include decisions regarding additions of equipments. Supply chain management problems extend the scheduling problem in the spatial dimension and consider the coordinated management of multiple facilities and the shipment of materials through an associated transportation network. The product and research pipeline management problem has much overlap both with supply chain management and process scheduling problems. These problems are closely related to pharmaceutical industries where new drugs and products are invented regularly. Obviously these problems have great deal of uncertainty. Research management in general is also related to prioritization and reducing uncertainties. The chemical engineering literature is concentrated in problems in batch scheduling and planning including design and retrofit problems [104, 119]. Supply chain management problems are rare [120], and research management problems have only recently being studied [48, 122–124].

4.5. Future Directions

Research in all aspects of optimization will continue at a fast pace. However, the solution strategies for optimization models with smooth functions, continuous derivatives and continuous-valued optimization variables appears to be maturing especially with regard to locally optimal solutions. Evolutionary methods (e.g.,

genetic algorithms, simulated annealing, particle swarm optimization, etc.) for global optimization have made a lot of inroads into engineering applications for two main reasons: (a) they are conceptually easy to understand and easy to program into a computer working code, (b) they are capable of handling a mixture of continuous-valued and discrete-valued optimization variables, and (c) they are capable of handling nonsmooth functions, for example, those that appear in material, energy and momentum balances. However, these evolutionary methods have the drawback of not being able to prove that they have reached a globally optimal solution and that their theoretical foundation is not as well-developed as deterministic methods. The challenge is for researchers in deterministic optimization to reach out to users of evolutionary methods, for example, researchers in the biological and life-sciences areas.

Research into optimization under uncertainty will continue to grow at a rapid pace. For most problems in life sciences, physical sciences, and engineering, the presence of uncertainty is the *overwhelming norm* and researchers can no longer pretend uncertainty does not matter or is insignificant. Multiobjective optimization in which several conflicting criteria need to be optimized is another growth area. In the current age when we have to build sustainable systems that have to maximize profit, minimize resource use, do minimal harm to the environment, make policy makers happy and get the politics to work out right, multiobjective optimization will have to be the *modus operandi*.

Interval and P-box methods will also be focus of future articles. In this keyword, we did not treat time-dependent uncertainties which have been recently studied for batch process design [125–128] and sustainability [129] “see also the special issues on “energy and sustainability” (Computers and Chemical Engineering, 35(8), 2011); “energy systems engineering” (Computers and Chemical Engineering, 35(9), 2011)”. These areas present new trends in optimization under uncertainty.

References

- L.T. Biegler, I.E. Grossmann, A.W. Westerberg: *Systematic Methods of Chemical Process Design*, Prentice Hall, Englewood Cliffs, New York 1997.
- J. Douglas: “*Conceptual Design of Chemical Processes*”, McGraw-Hill, New York 1998.
- W.D. Seider, J.D. Seader, D.R. Lewin: *Process Design Principles*, 2nd ed., J. Wiley & Sons, New York 2004.
- G. Ostrovsky, L.E.K. Achenie, I. Datskov, Y. Volin “An Approach to Multicriteria Optimization under Uncertainty”, *Chem. Eng. Sci.* **61** (2006) 2379–2393.
- P.E. Gill, W. Murray, H.W. Wright: *Practical Optimization*, Academic Press, New York 1992.
- C.A. Floudas: “*Deterministic Global Optimization: Theory, Methods and Applications*”, Kluwer Academic Publishers, Norwell, MA 1999.
- E. Hendrix, G.-B. Tóth: “*Introduction to Nonlinear and Global Optimization*”, Springer Verlag, Berlin 2010.
- Local optimization software: http://www.mat.univie.ac.at/~neum/glopt/software_l.html (accessed 14 August 2011).
- Global optimization software: http://www.mat.univie.ac.at/~neum/glopt/software_g.html (accessed 14 August 2011).
- G.L. Nemhauser, L.A. Wolsey: “*Integer and Combinatorial Optimization*”, Wiley-Interscience, New York 1999.
- J.E. Dennis, R. Schnabel: *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia 1996.
- R. Fletcher: *Practical Methods of Optimization*, 2nd ed., J. Wiley & Sons, New York 2001.
- J. Nocedal, S.J. Wright: *Numerical Optimization*, 2nd ed., Springer Verlag, Berlin 2006.
- A.V. Fiacco, G.P. McCormick: *Nonlinear Programming: Sequential Unconstrained Minimization Technique*, Society for Industrial and Applied Mathematics, Philadelphia 1990.
- L.S. Lasdon: *Optimization Theory for Large Systems*, General Publishing Comp., Toronto, Canada 2002.
- D.A. Pierre, R.J. Lowe: “*Mathematical Programming via Augmented Lagrangians: An Introduction with Computer Programs*”, Addison-Wesley Pub. Co., London 1975.
- M.J.D. Powell: “A Fast Algorithm for Nonlinearly Constrained Optimization Calculations”, in G.A. Watson (ed.): *Numerical Analysis, Lecture Notes in Mathematics 630*, Springer Verlag, Berlin 1977.
- B.A. Murtagh, M.A. Saunders: “A projected Lagrange algorithm and its implementation for sparse nonlinear constraints”, *Mathematical Programming Study* **16** (1982) 84–117.
- T.F. Edgar, D.M. Himmelblau: “*Optimization of Chemical Process*”, 2nd ed., McGraw-Hill, New York 2001.
- A. Lucia, F. Yang: “Global Terrain Methods”, *Comp. Chem. Eng.* **26** (2002) 529.
- A. Lucia, F. Yang: “Multivariable Terrain Methods”, *AIChEJ.* **49** (2003) 2553.
- M.J.D. Powell: “A Hybrid Method for Nonlinear Equations” in P. Rabinowitz (ed.): *Numerical Methods for Nonlinear Algebraic Equations*. Gordon and Breach, New York 1970.
- M. Shacham: “Comparing Software for the Solution of Systems of Nonlinear Algebraic Equations Arising in Chemical Engineering”, *Comp. Chem. Eng.* **9** (1985) 103.
- A. Lucia, P.A. DiMaggio, P. Depa: “A Geometric Terrain Methodology for Global Optimization”, *J. Global Optim.* **29** (2004) 297.
- A. Lucia, F. Yang: “Solving Distillation Problems by Terrain Methods”, *Comp. Chem. Eng.* **28** (2004) 2541.
- A. Lucia et al.: “A Barrier-Terrain Methodology for Global Optimization”, *Ind. Eng. Chem. Res.* **47** (2008) 2666.

- 27 E.M.L. Beale: "On minimizing a convex function subject to linear inequalities", *J. Royal Statistical Society* **17B** (1955) 173.
- 28 G.B. Dantzig: "Linear programming under uncertainty", *Management Science* **1** (1955) 197.
- 29 U.M. Diwekar: "Introduction to Applied Optimization", Kluwer Academic Publ., Boston 2002.
- 30 E. Edgeworth: "The mathematical theory of banking", *J. Royal Statistical Society* **51** (1888) 113.
- 31 N.C. Petruzzi, M.M. Dada: "Pricing and the newsvendor problem: a review with extensions", *Operations Research* **47** (1999) no. 2, 183.
- 32 S. Vajda: *Probabilistic programming*, Academic Press, New York 1972.
- 33 G.L. Nemhauser, A.H.G. RonnooyKan, M.J. Todd: *Optimization: Handbooks in operations research and management science*, vol. 1. North Holland Publ., Amsterdam 1989.
- 34 A. Madansky: "Inequalities for stochastic linear programming problems", *Man. Sci.* **6** (1960), 197.
- 35 G. Tintner: "Stochastic linear programming with applications to agricultural economics", *Proc. 2nd Symp. Lin. Progr.*, Washington 1955, p. 197.
- 36 G. Taguchi: "Introduction to Quality Engineering", Asian Productivity Center, Tokyo, Japan 1986.
- 37 U.M. Diwekar, E.S. Rubin: "Parameter design method using Stochastic Optimization with ASPEN", *Ind. Eng. Chem. Res.* **33** (1994) 292.
- 38 A. Charnes, W.W. Cooper: "Chance-constrained programming", *Management Science* **5** (1959) 73.
- 39 E. Luckacs: *Characteristic Functions*, Charles Griffin, London 1960.
- 40 C.D. Maranas: "Optimal molecular design under property prediction uncertainty", *AIChE J.* **43** (1997) no. 5.
- 41 S.B. Petkov, C.D. Maranas: "Quantitative assessment of uncertainty in the optimization of metabolic pathways", *Biotechnol. Bioeng.* **56** (1997) no. 2, 145.
- 42 T. Hene, S.V. Dua, E.N. Pistikopoulos: "A hybrid parametric stochastic programming approach for mixed-integer nonlinear problems under uncertainty", *Ind. Eng. Chem. Res.* **41** (2002) 67.
- 43 E.N. Pistikopoulos, M.C. Georgiadis, V. Dua: *Multi-parametric programming: theory, algorithms, and applications*, Wiley-VCH Verlag, Weinheim 2007.
- 44 H. Raiffa, R. Schlaifer: *Applied Statistical Decision Theory*, Harvard University, Boston 1961.
- 45 U.M. Diwekar, E.S. Rubin: "Stochastic modeling of chemical Processes", *Comp. Chem. Eng.* **15** (1991) 105.
- 46 U.M. Diwekar, E.S. Rubin, H.C. Frey: "Optimal design of advanced power systems under uncertainty", *Energy Conversion and Management Journal* **38** (1997) 1725.
- 47 U.M. Diwekar: "A process analysis approach to pollution prevention", *AIChE Symposium Series on Pollution Prevention through Process and Product Modifications* **90** (1995) 168.
- 48 D. Subramanian, J.F. Pekny, G.V. Reklaitis: "A simulation optimization framework for research and development pipeline management", *AIChE J.* **47** (2001) no. 10, 2226.
- 49 M. Hostrup, P.M. Harper, R. Gani: "Design of environmentally benign processes: integration of solvent design and separation process synthesis," *Comp. Chem. Eng.* **23** (1999) 1395.
- 50 K. Kim, U.M. Diwekar: "Efficient combinatorial optimization under uncertainty-2: Application to stochastic solvent selection", *Ind. Eng. Chem. Res.* **41** (2002) 1285.
- 51 K. Kim, U.M. Diwekar, K.G. Joback: "Greener Solvent Selection under Uncertainty", *ACS Symposium Series No. 819 Clean Solvents: Alternative Media for Chemical Reactions and Processing*, Washington, USA, 2001.
- 52 M.A. Stadtherr: "Interval Analysis: Application to Chemical Engineering Design Problems", in C.A. Floudas, P.M. Pardalos (eds.): *Encyclopedia of Optimization*, Kluwer Academic Publ., 2001.
- 53 C.R. Gwaltney et al.: "Interval Methods for Nonlinear Equation Solving Applications" in W. Pedrycz, A. Skowron, V. Kreinovich (eds.): *Handbook of Granular Computing*, J. Wiley & Sons, New York 2008.
- 54 J.A. Enszer et al.: "Probability bounds analysis for nonlinear dynamic process models", *Aiche J.* **57** (2011) 404-422.
- 55 B.A.P. James: "Variance reduction techniques", *J. Operations Research Society* **36** (1985) no. 6, 525.
- 56 M.D. McKay, R.J. Beckman, W.J. Conover: "A comparison of three methods of selecting values of input variables in the analysis of output from a computer code", *Technometrics* **21** (1979) no. 2, 239.
- 57 R.L. Iman, W.J. Conover: "Small sample sensitivity analysis techniques for computer models, with an application to risk assessment", *Communications in Statistics* **A17** (1982) 1749.
- 58 E. Saliby: "Descriptive sampling: A better approach to Monte Carlo simulations", *J. Operations Research Society* **41** (1990) no. 12, 1133.
- 59 J.R. Kalganam, U.M. Diwekar: "An efficient sampling technique for off-line quality control", *Technometrics* **39** (1997) no. 3, 308.
- 60 U.M. Diwekar, J.R. Kalaganam: "An efficient sampling technique for optimization under uncertainty", *AIChE J.* **43** (1997) 440.
- 61 U. Diwekar, S. Ulas: "Sampling Techniques", *Kirk-Othmer Encyclopedia of Chemical Technology*, vol. 26, J. Wiley & Sons, New York 2007, p. 998.
- 62 D. Frenkel, B. Smit: "Understanding Molecular Simulations", Academic Press, New York 1996.
- 63 D.E. Knuth: "The Art of Computer Programming, Volume 1: Fundamental Algorithms", Addison-Wesley Publ., Reading, MA 1973.
- 64 G. Morgan, M. Henrion: *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*, Cambridge University Press, Cambridge 1990.
- 65 R.L. Iman, J.C. Helton: "An investigation of uncertainty and sensitivity analysis techniques for computer models", *Risk Analysis* **8** (1988) no. 1, 71.
- 66 T.-M. Tong, M.E. Reed, W.B. Whiting: "The effect of uncertainties in thermodynamic data and model parameters on calculated process performance", *Ind. Eng. Chem. Res.* **32** (1993) 1367.
- 67 F.P. Bernardo, E.N. Pistikopoulos, P.M. Saraiva: "Quality costs and robustness criteria in chemical process design optimization", *Comp. Chem. Eng.* **25** (2001) 27.
- 68 M. Wendt, G. Wozny: "Nonlinear Chance-Constrained Process Optimization under Uncertainty", *Ind. Eng. Chem. Res.* **41** (2002) 3621.
- 69 V. Vasquez, W. Whiting: "Uncertainty and sensitivity analysis of thermodynamics models using equal probability sampling (EPS)", *Comp. Chem. Eng.* **23** (2000) 1825.
- 70 R. Wang, U. Diwekar, C. Gregoire-Padro: "Latin Hypercube Hammersley Sampling for risk and uncertainty analysis", *Environm. Progr.* **23** (2004) 141.
- 71 R. Wang, U. Diwekar: "A novel approach for off-line quality control", *AIChE Annual Meeting*, Los Angeles, CA, 2000.

- 72 J.F. Bender: "Partitioning procedures for solving mixed-variables programming problems", *Numer. Math.* **4** (1962) 238.
- 73 G.B. Dantzig, G. Infanger: "Large scale stochastic linear programs—Importance sampling and bender decomposition" in Brezinski, U. Kulisch (eds.): *Computational and Applied Mathematics*, North-Holland Publ., Amsterdam 1991, p. 111.
- 74 C. Bouza: "Stochastic programming: the state of the art", *Revista Investigacion Operacional* **14** (1993) no. 2.
- 75 R. VanSlyke, R.J.B. Wets: "L-shaped linear programs with application to optimal control and Stochastic Programming", *SIAM Journal on Applied Mathematics* **17** (1969) 638.
- 76 J.R. Birge, F. Louveaux: "A multicut algorithm for two-stage stochastic linear programming", *European Journal of Operations Research* **34** (1988) 384.
- 77 J.R. Birge, F. Louveaux: *Introduction to stochastic programming*, Springer Verlag, Berlin 1997.
- 78 J.L. Hige, S. Sen: "Stochastic decomposition: an algorithm for two-stage linear programs with recourse", *Mathematics of Operations Research* **16** (1991) no. 3, 650–669.
- 79 J.L. Hige, S. Sen: "Stochastic decomposition", Kluwer Academic Publ., Boston, MA 1996.
- 80 M.G. Ierapetritou, E.N. Pistikopoulos: "Novel optimization approach of stochastic planning models", *Ind. Eng. Chem. Res.* **33** (1994) 1930.
- 81 S. Ahmed, N.V. Sahinidis, E.N. Pistikopoulos: "An improved decomposition algorithm for optimization under uncertainty", *Comp. Chem. Eng.* **23** (2000) 1589.
- 82 C.A. Floudas, Z.H. Guemues, M.G. Ierapetritou: "Global optimization in design under uncertainty: feasibility test and flexibility index problems", *Ind. Eng. Chem. Res.* **40** (2001) 4267.
- 83 G. Ostrovsky, Yu.M. Volin: "On new problems in the theory of flexibility and optimization of chemical process under uncertainty", *Theor. Found. Chem. Eng.* **33** (1999) no. 5, 524.
- 84 K. Sahin, U. Diwekar: "Better optimization of nonlinear uncertain systems (BONUS): A new algorithm for stochastic programming using reweighting through kernel density estimation", *Annals of OR* **132** (2004) 47.
- 85 K. Sahin, U. Diwekar: "Stochastic nonlinear optimization -The BONUS algorithm", *SIAM Newsletter* **13** (2002) 1.
- 86 J.R. Birge: "Stochastic Programming computation and applications", *INFORMS journal on computing* **9** (1997) no. 2, 111.
- 87 L.A. Painton, U.M. Diwekar: "Stochastic annealing under uncertainty", *European Journal of Operations Research* **83** (1995), 489.
- 88 P. Chaudhuri, U. Diwekar: "Synthesis approach to optimal waste blend under uncertainty", *AIChE J.* **45** (1999) 1671.
- 89 M. Tayal, U. Diwekar: "Novel sampling approach to optimal molecular design under uncertainty: A polymer design case study", *AIChE J.* **47** (2001) no. 3, 609.
- 90 M.M. Dantus, K.A. High: "Evaluation of waste minimization alternatives under uncertainty: a multiobjective optimization approach", *Comp. Chem. Eng.* **23** (1999) 1493.
- 91 W. Xu, U. Diwekar: "Improved genetic algorithms for deterministic optimization and optimization under uncertainty-Part I: Algorithms development", *Ind. Eng. Chem. Res.* **44** (2005) 7132.
- 92 W. Xu, U. Diwekar: "Improved genetic algorithms for deterministic optimization and optimization under uncertainty-Part II: Solvent selection under uncertainty", *Ind. Eng. Chem. Res.* **44** (2005) 7138.
- 93 W. Xu, U. Diwekar: "Multi-objective integrated Solvent selection and solvent recycling under uncertainty using new genetic algorithms", *International Journal of Environment and Pollution* **29** (2006) 70.
- 94 Massachusetts Institute of Technology (1982), *ASPEN User's Manual*, vol. 1, Reports DOE/MC/16481-1203.
- 95 T.F. Edgar, D.M. Himmelblau, L.S. Lasdon: *Optimization of chemical processes*, 2nd ed., McGraw-Hill, New York 2001.
- 96 J.R. Kittrel, C.C. Watson: "Don't overdesign equipment", *Chem. Eng. Prog.* **63** (1966) 79.
- 97 C.Y. Wen, T.M. Chang: "Optimal design of systems involving parameter uncertainty", *Ind. Eng. Chem. Proc. Des. Dev.* **7** (1973) 49.
- 98 N. Watanbe, Y. Nishimura, M. Matsubara: "Optimal design of chemical processes involving parameter uncertainty", *Chem. Eng. Sci.* **28** (1973) 905.
- 99 R.K. Malik, R.R. Hughes: "Optimal design of flexible chemical processes", *Comp. Chem. Eng.* **7** (1983) 423.
- 100 J. Weisman, A.G. Holzman: "Optimal process system design under conditions of risk", *Ind. Eng. Chem. Proc. Des. Dev.* **11** (1972) 386.
- 101 N. Nishida, A. Ichikawa, E. Tazaki: "Synthesis of optimal process systems with uncertainty", *Ind. Eng. Chem. Res.* **13** (1974) 209.
- 102 R.S. Swaney: "Analysis of Operational Flexibility in Chemical Process Design", Ph.D. Thesis, Carnegie Mellon University, Pittsburgh 1983.
- 103 V. Bansal, J.D. Perkins, E.N. Pistikopoulos: "Flexibility analysis and design using a parametric programming framework", *AIChE J.* **48** (2002) 2851.
- 104 J.F. Pekny: "Algorithm architecture to support large-scale process systems engineering applications involving combinatorics, uncertainty, and risk management", *Comp. Chem. Eng.* **26** (2002) 239.
- 105 C.G. Raspanti, J.A. Bandoni, L.T. Biegler: "New strategies for flexibility analysis and design under uncertainty", *Comp. Chem. Eng.* **24** (2000) 2193.
- 106 A. Bendell, J. Disney, W.A. Pridmore: *Taguchi methods: applications in world industry*, IFS Publication, Springer Verlag, Berlin 1989.
- 107 S. Boudriga: "Evaluation of parameter design methodologies for the design of chemical processing units", Master's thesis, University of Ottawa, Ottawa 1990.
- 108 J. Dearlington et al.: "An algorithm for constrained nonlinear optimization under uncertainty", *Automatica* **35** (1999) 217.
- 109 H. Markowitz: *Portfolio Selection: Efficient Diversification of Investments*, J. Wiley & Sons, New York 1959.
- 110 Y. Fu et al.: "Process design for environment: A multi-objective framework under uncertainty", *Clean Products and Processes* **2** (2000) 92.
- 111 H. Cabezas, J.C. Bare, S.K. Mallick: "Pollution prevention with chemical process simulators: the generalized waste reduction (WAR) algorithm", *Comp. Chem. Eng.* **22** suppl. (1997) 305.
- 112 A. Goyal, U. Diwekar: "Multi-objective process synthesis for designing cost effective environment friendly processes", *Proceedings of 6-th World Congress of Chemical Engineering*, Sydney, Australia, 2001.
- 113 K. Subramanian, U. Diwekar: "Optimizing model complexity with application to fuel cell based power systems", *Chem. Eng. Process.* **46** (2007) 1116.
- 114 G. Ostrovsky et al.: "An approach to multicriteria optimization under uncertainty", *Chemical Engineering Science* **61** (2006), 2379.

- 115 M. Sinha, L.E. Achenie, G.M. Ostrovsky: "Environmentally benign solvent design by global optimization", *Comp. Chem. Eng.* **23** (1999) 1381.
- 116 K. Kim, U.M. Diwekar: "Integrated solvent selection and recycling for continuous processes", *Ind. Eng. Chem. Res.* **41** (2002) 4479.
- 117 V. Narayan, U.M. Diwekar, M. Hoza: "Synthesizing optimal waste blends", *Ind. Eng. Chem. Res.* **35** (1996) 3519.
- 118 A. Linninger A. Chakraborty: "Pharmaceutical waste management under uncertainty", *Comp. Chem. Eng.* **25** (2001) 675.
- 119 Z. Li, M.G. Ierapetritou: "Process scheduling under uncertainty: Review and challenges", *Comp. Chem. Eng.* **32** (2008) 715.
- 120 A. Gupta, C.D. Maranas, C.M. McDonald: "A two-stage modeling and solution framework for multisite midterm planning under demand uncertainty", *Comp. Chem. Eng.* **39** (2000) 3799–3813.
- 121 D.A. Straub, I.E. Grossmann: "Integrated stochastic metric of flexibility for systems with discrete state and continuous parameter uncertainties", *Comp. Chem. Eng.* **14** (1990) , 967.
- 122 T. Johnson, U. Diwekar: "The value of design research: stochastic optimization as a policy tool", *Foundations of Computer-Aided Design, AIChE Symposium Series* **96** (2000) 454.
- 123 T.L. Johnson, U.M. Diwekar: "Hanford waste blending and the value of research: stochastic optimization as a policy tool", *Journal of Multi-criteria Decision Analysis* **10** (2001) 87.
- 124 F. You, J.M. Wassick, I.E. Grossmann: "Risk management for a global supply chain planning under uncertainty: models and algorithms", *AIChE J.* **55** (2009) 931.
- 125 V. Rico-Ramirez, B. Morel, U. Diwekar: "Real option theory from finance to batch distillation", *Comp. Chem. Eng.* **27** (2003) 1867.
- 126 S. Ulas, U. Diwekar: "Thermodynamic uncertainties in batch processing and optimal control", *Comp. Chem. Eng.* **28** (2004) 2245.
- 127 S. Ulas, U. Diwekar: "Integrating product and process design with optimal control: A case study of solvent recycling", *Chem. Eng. Sci.* **61** (2006) 2001.
- 128 S. Ulas, U. Diwekar, M. Stadtherr: "Uncertainties in parameter estimation and optimal control in batch distillation", *Comp. Chem. Eng.* **29** (2005) 1805.
- 129 U. Diwekar, Y. Shastri: "Green process design, green energy, and sustainability: A systems analysis perspective", *Comp. Chem. Eng.* **34** (2010) 1348.