On the Solution of the Instrumentation Selection Problem

M. Carnero,[†] J. L. Hernández,[†] M. Sánchez,*[,‡] and A. Bandoni[‡]

Departamento Ciencias Básicas, Facultad de Ingeniería, Universidad Nacional de Río Cuarto, Campus Universitario, (5800) Río Cuarto, Argentina, and Planta Piloto de Ingeniería Química, UNS-CONICET, Camino La Carrindanga km 7, (8000) Bahía Blanca, Argentina

In this work, a procedure for solving the optimal design and upgrade of linear sensor networks, subject to quality constraints on a set of key variable estimates, is presented. The strategy aims to select the optimal set of flowmeters without imposing restrictions on the mathematical nature of the objective function and constraints. An evolutionary technique based on genetic algorithms (GAs) is proposed that combines the benefits of using structured populations in the form of neighborhoods and a local search strategy. Both procedures take advantage of existing process knowledge. Application examples are provided for the instrumentation design of a steam metering network of a methanol plant, which show that the algorithm has a good balance among its exploration and exploitation capabilities.

1. Introduction

Basic and high-level plant activities, such as monitoring, regulatory and supervisory control, real-time optimization, planning and scheduling, etc., provide valuable results only if a reliable and complete knowledge of current plant state is available. The quality and availability of variable estimates strongly depend on the structure of instruments installed in the process and the software tools applied to enhance its precision.

The design and upgrade of sensor structures consists of selecting the type, number, accuracy, failure rate, and location of new sensors that provide the quantity and quality of information required from the process. Depending on the number and location of the selected measurements, different types of instrument arrangements are defined. A minimum-number sensor network contains the smallest number of instruments that allows the estimation of all unmeasured variables. This quantity is known in advance to problem resolution, as it is the system degree of freedom. If more sensors are used than the minimum required to fulfill the aforementioned condition, a so-called redundant sensor network is obtained. Frequently, it is necessary to satisfy quality or availability constraints on only a subset of key measured or unmeasured variable estimates. In this case, a general sensor network is designed without advance knowledge of the cardinality of the optimal sensor set. As only a subset of variables are of real practical interest, the optimal selection of measurements for general sensor networks is a powerful tool in both the design and upgrade stages of large-scale plants.

Different criteria are addressed in the optimal selection of sensor structures: instrumentation cost, global error of variable estimates, system reliability, variable availability, economic value of an instrumentation project, etc. In any case, a discrete optimization problem, frequently subject to constraints, arises. Both deterministic and stochastic approaches are applied to solve this problem.

For the design of minimum-number sensor networks, the concepts of cost-edged graphs and minimum spanning trees of these graphs were used by Madron[1] to obtain minimum-cost or maximum-overall-precision linear sensor configurations. Then, Ali and Narasimhan[2] introduced the concept of variable reliability and proposed a procedure based on graph theory to maximize the least variable reliability among all variables. Later, this approach was extended to tackle the design of minimum-cost and minimum-overall-error-estimation linear sensor structures.[3] For minimum-number bilinear systems, Ali and Narasimhan[4] used graph theory to solve a max−min problem using variable reliability as the objective function, and Héraud and Mazzour[5] proposed an equation-oriented approach for the design of minimum-cost instrument arrangements. Regarding stochastic approaches, Sen et al.[6] presented a graph-based genetic algorithm (GA) that can be applied to any objective function, and two evolutionary strategies for multiobjective design appeared[7,8] recently.

The design of redundant sensor structures was first addressed by Kretsovalis and Mah.[9] They used a combinatorial search to incorporate measurements into an observable system. This method tries to make a weighted average between the cost of the measurements and the precision of the estimates. Later, Ali and Narasimhan[10] developed an algorithm for the design of linear redundant sensor networks, using a given number of sensors that maximizes the reliability of variables. Furthermore, the evolutionary technique by Viswanath and Narasimhan[7] can also be applied to design this kind of sensor structure.

For the design of general networks, Madron and Veverka[11] proposed that the variables of linear systems be categorized as required and nonrequired. Unmeasured variables are later ordered from "measured with difficulty" to "easily measured". The cost and a measure of the overall accuracy of the system are used as objective functions. Meyer et al.[12] presented a branch-and-bound algorithm to select the measurements that lead to the total or partial observability of the process while minimizing the instrumentation cost. Luong et al.[13] developed a strategy based on the analysis of the process graph that provides solutions that feature the minimal observability of variables required for control

* To whom correspondence should be addressed. Tel.: +54 91 86 1700. Fax: +54 91 86 16 00. E-mail: msanchez@plapiqui.edu.ar.

[†] Universidad Nacional de Río Cuarto.

[‡] UNS-CONICET.

10.1021/ie049856t CCC: $30.25
Published on Web 12/23/2004

and a high degree of redundancy for a particular set of variables. They used reliability as means of screening alternatives with equal costs. Later, Bagajewicz[14] introduced binary variables to indicate the presence or absence of a sensor and formulated an MINLP problem to obtain cost-optimal network structures for linear systems subject to constraints on the precision and robustness of variable estimates. The algorithm used here is an implicit tree-type enumeration with a branching stopping criteria. Chmielewski et al.[15] reformulated the problem by replacing unmeasured variables by measurements obtained using dummy sensors of large variance and no cost and solved a convex MILP formulation using a branch-and-bound algorithm. An alternative MILP approach was addressed by Bagajewicz and Cabrera[16] that was also applied to multiobjective instrumentation design.[17] The two aforementioned MILP strategies are useful for small- to medium-sized problems and required all constraints to be explicit in terms of the binary variables.[18] Regarding stochastic algorithms, Chao-An et al.[19] presented the design of maximum-availability general sensor networks subject to cost and precision constraints, but they solved the problem for a small-sized network using the classic GA. A comprehensive discussion of solution procedures for the design and upgrade of instrumentation can be found in Bagajewicz's book[20] and his review works.[21−23]

In this paper, a new evolutionary strategy for the design and upgrade of general linear sensor networks is presented. It was developed to solve any single-objective optimization problem subject to constraints, without restrictions on the mathematical nature of either the objective function or constraints. It provides a good solution for problems of larger scale than those solved up to the present time in the open literature. The developed procedure is based on the use of GAs and concepts from linear algebra.

This work is organized as follows: In section 2, the general sensor design and upgrade problem is briefly introduced. The algorithm is described in section 3, and results are presented for an industrial steam metering network in section 4. Conclusions and future research topics are addressed in section 5.

2. Problem Formulation

Let us assume the operation of a process under steady-state conditions can be represented by the following set of linear equations

$$\mathbf{Dz} = \mathbf{A}_1\mathbf{x} + \mathbf{A}_2\mathbf{u} = \mathbf{0} \tag{1}$$

where $\mathbf{D}$ is the incidence matrix of dimensions $m \times n$; $\mathbf{z}$ is the n-dimensional vector of flow rates; $\mathbf{x}$ and $\mathbf{u}$ represent the vectors of measured and unmeasured variables, respectively; and $\mathbf{A}_1$ and $\mathbf{A}_2$ are submatrices of $\mathbf{D}$ of compatible dimensions. The problem of optimal selection of instruments during plant design or upgrade consists of determining the optimal partition of vector $\mathbf{z}$ into vectors $\mathbf{x}$ and $\mathbf{u}$, and it is formulated in general form as follows

$$\begin{aligned} &\min f(\mathbf{q}) \\ \text{s.t.} \quad & g_j(\mathbf{q}) \le g_j^*(\mathbf{q}) \qquad \forall\, j \in S_J \\ & q_i = 1 \qquad \forall\, i \in I_0 \\ & \mathbf{q} \in \{0, 1\}^{n-|I_0|} \end{aligned} \tag{2}$$

where $\mathbf{q}$ is an n-dimensional vector of binary variables such that $q_i = 1$, if variable i is measured and $q_i = 0$ otherwise, $f(\mathbf{q})$ represents a one-dimensional objective function, and $g_j(\mathbf{q})$ indicates the constraint imposed on the quality of the jth key variable estimate. Furthermore, S_J is the set of key process variables, and I_0 is the initial set of instruments that is empty at the network design stage. For the sake of simplicity, it is assumed that only one constraint exists for each key variable.

Different performance criteria for the sensor structure, $f(\mathbf{q})$, arise depending on the specific application. Frequently, the life-cycle instrumentation cost for the design or upgrade project[24] leads the selection; nevertheless, reliability measures are sometimes preferred for safety reasons. A wide variety of objective functions have been used: instrumentation cost, global error of variable estimates, system reliability, variable reliability and availability, economic value of an instrumentation project, etc.

Regarding the set of constraints, $\mathbf{g}(\mathbf{q})$, engineers not only require that the values of key variables be known for economic, safety, or environmental reasons, but also impose conditions on their estimate precision, reliability, or availability.

For large-scale processes, the dimensions of the search space for problem 2 increase significantly; consequently, the design turns out to be a huge combinatorial optimization problem that might have many local optima. In these cases, it is valuable that the solution procedure provide at least a good solution, if not the global optimum, and that it also can be run in parallel computers to reduce execution time. Evolutionary strategies based on GAs fulfill these features; thus, they are considered as an attractive approach to solve this complex problem.

3. Novel Evolutionary Strategy for Instrumentation Design and Upgrade

Evolutionary algorithms (EAs) are probabilistic search algorithms that maintain a population of tentative solutions of an optimization problem. These are manipulated competitively by applying variation operators to find a satisfactory, if not globally optimum, solution. Among the different types of EAs, GAs remain the most recognized form. The standard GA applies stochastic operators (selection, crossover, and mutation) to an initially random population to compute new solutions. The new population is evaluated, and the cycle is repeated until some stopping criteria are reached.[25]

Traditionally, the population structure was panmictic, meaning that the whole population is considered as a single pool of individuals, each one of which can potentially mate with any other and can even leave the pool and be replaced by another one. In contrast, a structured population arises if some partition of the single pool is undertaken in the form of islands or neighborhoods. The structured-population model enhances the sampling of the search space and improves the numerical and run-time behavior of the basic strategy. Alba and Tomassini[26] provide a good review of EAs with distributed population in the context of parallel computing.

The incorporation of problem-specific knowledge into the algorithm has proven to be a powerful strategy to increase its convergence. To include this knowledge, ad hoc initialization techniques, chromosome representa-

tions, decoding approaches, and heuristics for genetic operators are applied. Furthermore, hybrid methods result as a combination of metaheuristic techniques with local search procedures.[27]

Taking into account the aforementioned strategies to improve algorithm performance, a procedure is developed that has the following distinctive features: (i) feasible initial population with respect to the estimability of the required variables; (ii) selection based on ranking, followed by matching of the best individuals in the neighborhood to form the mating pool; (iii) incorporation of the new individuals into their corresponding neighborhood if they are better than their parents; and (iv) local optimization of the best current solutions.

In the rest of this section, a detailed description of the proposed solution procedure is provided.

3.1. Initial Population. Although the random generation of the initial population is usually the first step in a standard GA, it is well-known that the incorporation of problem knowledge enhances the performance of EAs. Therefore, an ad-hoc procedure is developed to generate an initial population composed of feasible individuals regarding the estimability condition of key process variables. A variable is estimable if it is measured or unmeasured but observable. Consequently, each member of the initial population represents a set of sensors that allows the estimation of all required variables using measurements and mass balance calculations.

Initially, an individual of the population is represented by a matrix $\mathbf{T}$ of dimensions $r_v \times (m + 1)$, where r_v is the number of required variables and m represents the number of rows of the incidence matrix $\mathbf{D}$. The ith row of $\mathbf{T}$ is associated with a required variable and contains (a) one way of calculating it, expressed in terms of the ordinal position of the equation or the combination of equations from the set $\mathbf{Dz} = \mathbf{0}$ used to estimate it (if the number of equations in the combination is lower than m, then zeros are added to complete the first m columns) and (b) a discrete random variable, called the measurement index, MI, whose value is 1 if the required variable is a nonredundant measurement, -1 if the required variable is a redundant measurement, or 0 if the required variable is an unmeasured variable. A chromosome is represented in Figure 1.

A special convention has been adopted to write the rows of $\mathbf{D}$ involved in the calculation of a required variable. For each row of $\mathbf{T}$, the first place is occupied by the ordinal position of a row of $\mathbf{D}$ with nonzero coefficients for this variable. The other places are filled with the ordinal positions of equations with zero coefficients, and the remaining positions with zeros.

Example 1. To illustrate chromosome representation, let us assume that two variables z_2 and z_4 are required for the process represented by the set of linear equations $\mathbf{Dz} = \mathbf{0}$, where $\mathbf{D}$ is the following incidence matrix

$$\mathbf{D} = \begin{bmatrix} 1 & -1 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & -1 & 0 \end{bmatrix}$$

Rows 1 and 2 of $\mathbf{D}$ contain nonzero coefficients for variable z_2. Thus, this variable can be calculated in four ways: (a) using eq 1, (b) using eq 2, (c) using a combination of eqs 1 and 3, and (d) using a combination of eq 2 and 3. These equations are explicitly written as follows

$$z_1 - z_2 - z_4 = 0$$
$$z_2 - z_3 + z_6 = 0$$
$$z_4 - z_5 = 0$$

T_{11}	T_{12}	……	T_{1m}	MI_1
T_{21}	T_{22}	……	T_{2m}	MI_2
T_{rv1}	T_{rv2}	……	T_{rvm}	MI_{rv}

Figure 1. Chromosome representation.

Now suppose that the chromosome represented by Figure 2 is generated. This means that the required variable z_2 is calculated using the linear combination of eqs 1 and 3, the required variable z_4 is calculated using the linear combination of eqs 1 and 2, and z_2 and z_4 are considered a redundant measurement and an unmeasured variable, respectively. ■

In contrast to previous chromosome representations,[28] the discrete random variable MI has an outcome equal to -1 that allows the introduction of redundant measured variables in the initial population. Furthermore, higher probabilities for MI $= -1$ are considered to increase redundancy and, consequently, to improve process knowledge, for example, by increasing precision estimates and variable availability.

The maximum population size is the total number of combinations among the $2^{m-1} \times r_v$ subsets of equations that can be used to calculate the required variables and the three outcomes of variable MI. A fraction of the maximum population size is used to solve the problem. Its value is adjusted for each particular problem.

To form a member of the initial population, $\mathbf{T}$ $[r_v \times (m + 1)]$, the following procedure is applied for each required variable p ($p = 1, ..., r_v$) associated with a row of $\mathbf{T}$:

(a) Define $I = \{1, ..., m\}$.
(b) Define O as the set of the elements of I corresponding to rows of $\mathbf{D}$ with nonzero entries for variable p. At most, two elements of I satisfy this condition.
(c) Select the first bit of the chromosome row (T_{p1}). If O has two elements, a random selection of equal probability between them is performed, and a parameter a is set equal to 2. Otherwise, the only element of O constitutes the first bit, and $a = 1$.
(d) Set $l = 1$.
(e) Define $I' = I - O$ as the set of equations with zero entries for variable p (the cardinality of I' is represented by $|I'|$)
(f) Initialize $T_{pw} = 0$ ($w = 2, ..., m$).
(g) Generate T_{pw} ($w = 2, ..., m$).
  (i) Randomly generate an integer $b \in \{0, 1, ..., |I'|\}$.
  (ii) Obtain a random subset C of b elements that belong to I'.
  (iii) For each element $c_i \in C$, if at least one nonzero coefficient of equation c_i is opposite in sign with respect to the coefficient of a variable contained in the current combination (algebraic sum of

1	3	0	-1
1	2	0	0

Figure 2. Chromosome representation for example 1.

equations T_{pw} with $w = 1, ..., l$), then set $l = l + 1$ and $T_{pl} = c_i$, and update the current combination.

(h) Select MI randomly. Different probabilities are assigned to the outcomes of MI. A greater probability for MI = −1 increases sensor network redundancy.

The procedure described previously provides a random generation of a member of the initial population, **T**, that satisfies the estimability condition of all required variables. For each key variable p associated with a row of **T**, the equations from the set $\mathbf{Dz} = \mathbf{0}$ used to calculate it are selected sequentially. The element T_{p1} represents the ordinal position of a row of **D** with nonzero coefficient for this variable, and it is selected randomly between the two possible alternatives (steps a−c). The ordinal positions of equations with zero coefficients for p are included in set I' (step e), and after the elements T_{pw} ($w = 2, ..., m$) are set to zero (step f), a random integer b is generated. If $b \geq 0$, one combination of equations represented by their ordinal positions in **D** is randomly selected from the set (I'/b). Each equation of this combination is sequentially analyzed to determine whether it has at least one nonzero coefficient that is opposite in sign with respect to the equations just involved in the calculation of variable p. If this is the case, the equation is incorporated into the set of equations used to calculate the required variable. In this way, an equation is included in the set if it eliminates at least one existing variable from the calculus (step g).

The described procedure represents an individual as a matrix **T**. This representation is now transformed into a vector **q** of binary variables as follows:

(a) Initialize vector **q** and the memory vector **mem** as null vectors. Both vectors are of dimension n, with **q** having components equal to 1 for measured variables. The second vector has components equal to 1 for measured or observable unmeasured variables, that is, for estimable variables

(b) For each variable p ($p = 1, ..., r_v$) that belongs to the key variable vector **req**, do the following:

(i) If MI(p) = 1, the variable is a nonredundant measurement, and the element of the **q** vector that corresponds to **req**(p) is equal to 1, that is, **q**(**req**(p)) = 1; also **mem**(**req**(p)) = 1.

(ii) If MI(p) = 0, the variable is unmeasured, so it will be calculated using the linear combination proposed in the chromosome. The variables that participate in this combination, except p, should be measured and are included in **q**. Also, the new measured variables and the observable unmeasured variable, p, are incorporated into **mem**.

(iii) If MI(p) = −1, the variable is a redundant measurement. Consequently, it is measured and also can be calculated using the linear combination given in the chromosome. The above procedures are applied simultaneously.

(iv) If vector **mem** contains 1's in the positions corresponding to all required variables, stop.

If condition iv is satisfied, key variables can be estimated with the set of sensors represented by nonzero elements of vector **q**. A feasible individual regarding the estimability condition of key variables is obtained.

3.2. Fitness Calculation. In this work, the following evaluation function or fitness,[29] F, is used to measure the quality of a given vector **q** as a solution of the optimization problem defined by eq 2

$$F = \begin{cases} f(\mathbf{q}) & \text{if } \mathbf{q} \text{ is feasible} \\ f_{\max}[1 + Q(\mathbf{q})] & \text{if } \mathbf{q} \text{ is infeasible} \end{cases} \tag{3}$$

where $f(\mathbf{q})$ stands for the objective function of optimization problem 2, $f_{\max}$ represents an upper bound of $f(\mathbf{q})$ for feasible individuals, and $Q(\mathbf{q})$ takes into account constraint violations as follows

$$Q(\mathbf{q}) = \frac{1}{R}\sum_{r=1}^{R}\frac{g_r(\mathbf{q}) - g_r^*}{g_r(\mathbf{q})} \tag{4}$$

and R represents the number of unsatisfied constraints.

The fitness value of any infeasible solution is calculated as the sum of constraint violations plus the highest objective function value of all feasible solutions. Consequently, any infeasible solution has a fitness value worse than that of any of the feasible ones. In this constraint-handling technique, two solutions are compared on the basis of their objective function values if they are feasible or their constraint violations if they are infeasible.

3.3. Selection. The population is separated into neighborhoods in cellular EAs. Each individual has its own mating pool defined by its neighbors and also belongs to many pools. Structures with overlapping neighborhoods provide a smooth diffusion of good solutions across one- or two-dimensional grids.

In this work, a local selection strategy is applied that allows the interchange of genetic material among individuals corresponding to a linear neighborhood. The neighborhood of the kth population member is constituted by its $k - \nu$ predecessors and $k + \nu$ successors, where ν is the size of the neighborhood.

Before individuals are selected for reproduction, linear ranking is applied to produce a selection pressure that is more independent of the actual fitness values. In ranking selection methods, individuals are sorted in descending order of their fitness, and their selection probability is evaluated in terms of their ranks without considering their original fitness. For linear ranking, the selection probability of an individual ranked in position k is proportional to its rank (a rank of 1 indicates the best individual; a rank equal to the population size, N, represents the worst) and is calculated as follows

$$p_s(k) = \frac{1}{N}\left[\eta_{\max} - 2(\eta_{\max} - 1)\frac{k-1}{N-1}\right] \tag{5}$$

where $\eta_{\max}$ is the expected number of offspring of the best individual and it has been demonstrated that $1 \leq \eta_{\max} \leq 2$. This parameter controls the selection pressure.[25]

Then, the set of fathers is obtained by universal stochastic sampling. As all individuals have the same probability of being selected, fathers are uniformly distributed throughout the population. The best individuals in the linear neighborhood of each father constitute the set of mothers. Let us consider a small example to illustrate the local selection procedure.

Table 1. Population Data

individual	chromosome	fitness	p_s
1	11111110	11500	0.0909
2	10000101	5000	0.1182
3	11010110	7500	0.1073
4	10100101	7500	0.1073
5	01010111	8000	0.1018
6	11010110	7500	0.1073
7	01011110	8000	0.1018
8	11100100	7500	0.1073
9	11010010	18000	0.0636
10	00011110	6500	0.1073
11	01010110	6500	0.1073

Table 2. Neighborhood of Each Individual in the Population

predecessors	individual	successors
9, 10, 11	1	2, 3, 4
11, 1, 2	3	4, 5, 6
2, 3, 4	5	6, 7, 8
4, 5, 6	7	8, 9, 10
7, 8, 9	10	11, 1, 2

Table 3. Mating Pool

father	mother
1	2
3	2
5	2
7	10
10	2

Example 2. Suppose a population consists of 11 individuals with the chromosome representations, fitness values, and selection probabilities after the application of linear ranking with a selection pressure η_{max}=1.3 presented in Table 1. The set of fathers is $F_a = \{1, 3, 5, 7, 10\}$. For a neighborhood size of $\nu = 3$, the neighbors of each father are shown in Table 2. For the first father, the best neighbor is individual number 2, which is selected as the corresponding mother. The mating pool for this example is presented in Table 3. ■

3.4. Crossover and Mutation Operators. Each couple in the mating pool interchanges genetic material by applying a uniform crossover strategy. For each bit of the first offspring, the parent who supplies the value in that position is selected (with a certain probability). The second offspring receives the bit from the other parent. Regarding the mutation operator, the classical technique is used.

3.5. Offspring Location. Let us assume that the individual $\mathbf{i}_i$ has been selected as the father and its best neighbor, $\mathbf{i}_j$, is the corresponding mother. After the crossover and mutation operations have been applied, the offspring $\mathbf{i}_i'$ and $\mathbf{i}_j'$ are created. These offspring are incorporated into the population if the following conditions are independently satisfied, where $F(\mathbf{i}_i)$ represents the fitness of the ith individual $\mathbf{i}_i$: (a) If $F(\mathbf{i}_i')$ is better than $F(\mathbf{i}_i)$, $\mathbf{i}_i$ is replaced by $\mathbf{i}_i'$. (b) If $F(\mathbf{i}_j')$ is better than $F(\mathbf{i}_j)$, $\mathbf{i}_j$ is replaced by $\mathbf{i}_j'$.

The advantages of this offspring-location technique are that (a) new individuals are introduced into their corresponding neighborhoods and (b) elitism is incorporated into the algorithm, allowing good solutions to remain in the population after successive generations.

3.6. Local Search. If problem specific knowledge is available, this can be used to develop a local search technique that works in combination with the GA. The hybrid procedure allows both the global search feature of EAs and the convergence rate of the local search to be exploited.[27] Different implementation forms of hybrid strategies exist; for example, a percentage of the last-generation population (usually 5−10%) undergoes local optimization, or a portion of the best individuals of the current population go through local search and then are relocated in the population to continue the EA procedure. In this work, the latter alternative is used to implement the local search.

The developed procedure is applied on 5% of the best-adapted individuals, which are feasible solutions regarding the estimability condition of key variables. The search space around an individual is constituted by the set of solutions located at a Hamming distance of ≤2.[30] The technique performs a local search to find a solution of better fitness taking into account two alternatives: (a) interchange of a measured variable by an unmeasured one (Hamming distance = 2) or (b) elimination of a measured variable without replacement (Hamming distance = 1). The implemented local search used results provided by the variable classification procedure based on the Q−R orthogonal decomposition of matrix $\mathbf{A}_2$.[31] Following this approach, the portion $\mathbf{u}_r$ of the unmeasured variable vector $\mathbf{u}$ can be formulated as follows

$$\mathbf{u}_r = -\mathbf{R}_1^{-1}\mathbf{Q}_1^{T}\mathbf{A}_1\mathbf{x} - \mathbf{R}_1^{-1}\mathbf{R}_2\mathbf{u}_{n-r} = \mathbf{H}\mathbf{x} - \mathbf{R}_{IU}\mathbf{u}_{n-r} \tag{6}$$

where $\mathbf{R}_1$, $\mathbf{R}_2$, and $\mathbf{Q}_1$ come from the Q−R decomposition of matrix $\mathbf{A}_2$, which has a rank equal to r. Furthermore, a variable in $\mathbf{u}_r$ is observable if the corresponding row of matrix $\mathbf{R}_{IU}$ is a zero vector and a variable in $\mathbf{u}_{n-r}$ is unobservable.

If $\mathbf{q}_0$ is a solution selected to undergo a local search, the instrumentation set that represents this solution guarantees that key variables are estimable (measured or unmeasured but observable). The inspection of matrices $\mathbf{H}$ and $\mathbf{R}_{IU}$ allows the identification of all of the couples (unmeasured variable/measurement) belonging to $\mathbf{q}_0$ that can be interchanged satisfying the estimability condition of key variables. A pair of variables (u_i, x_j) can be interchanged if the coefficient $H_{ij} \neq 0$ and the ith row of $\mathbf{R}_{IU}$ is the zero vector. Regarding the set of solutions located at a Hamming distance = 1, a nonrequired measured variable j can be eliminated if $H_{ij} = 0$, $\forall\ i$ associated with the calculation of an unmeasured key variable.

Example 3. To illustrate this procedure, let us consider the following matrix $\mathbf{D}$

$$\mathbf{D} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \end{bmatrix}$$

and vectors $\mathbf{req}$ = [1 3 6] and $\mathbf{q}_0$ = [1 1 0 0 0 0 0 1]. Variable z_1 is measured and consequently estimable, and variables z_3 and z_6 are unmeasured but observable because they can be estimated using the following expression

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} - \begin{bmatrix} 0 \\ -1 \\ 0 \\ 1 \end{bmatrix} [u_5]$$

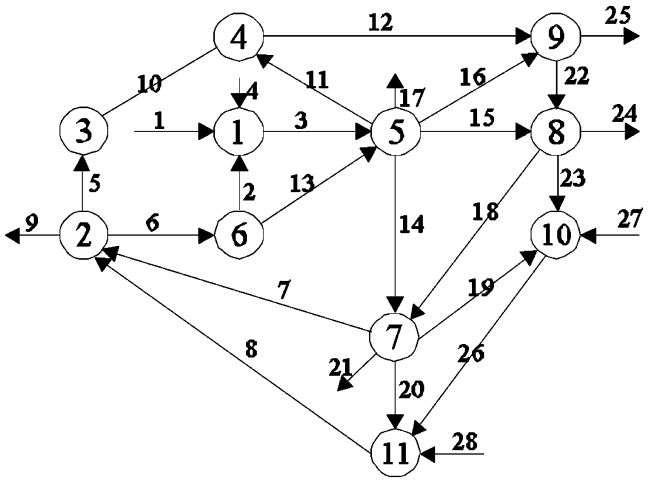

Figure 3. Steam metering network.

Table 4. Data for Steam Metering Network

variable	z	c	σ_s
1	0.86	3.7	0.0215
2	1	4.5	0.025
3	111.82	132.2	2.8
4	109.95	129.2	2.749
5	53.27	65.3	1.332
6	112.27	132.4	2.807
7	2.32	5.0	0.058
8	164.05	193.9	4.101
9	0.86	2.06	0.0215
10	52.41	62.8	1.31
11	14.86	20.2	0.3715
12	67.27	80.0	1.682
13	111.27	130.4	2.782
14	91.86	109.8	2.296
15	60.	71.6	1.5
16	23.64	29.7	0.591
17	32.73	39.5	0.8182
18	16.23	20.4	0.4057
19	7.95	11.1	0.1987
20	10.5	13.6	0.2625
21	87.27	102.9	2.182
22	5.45	8.1	0.1362
23	2.59	6.3	0.0648
24	46.64	55.5	1.166
25	85.45	101.0	2.136
26	81.32	93.7	2.033
27	70.77	84.7	1.769
28	72.23	85.4	1.806

where variables u_1-u_5 and x_1-x_3 correspond to flows z_3, z_4, z_6, z_5, z_7, z_1, z_2, and z_8 respectively. The couples of feasible interchange among unmeasured/measured variables are the following: (x_1, u_3), (x_2, u_1), (x_2, u_3), and (x_3, u_3). These couples preserve the estimability condition of key variables. For the select key variables, it is not feasible to eliminate measured variables, so no solution for a Hamming distance of 1 is possible. Considering a new vector of key variables **req** = [1 3], then variable x_3 can be eliminated from the set of measurements.

4. Application Examples

The procedure described above is applied to the instrumentation design of a steam metering network of a methanol production plant[32] represented in Figure 3. The process consists of 11 units interconnected by 28 streams. It is assumed that there is no restriction for the location of sensors on any stream, so the search space is made up of 2^{28} solutions.

Precision requirements are imposed on a subset of key variables; the remaining ones need to satisfy only the estimability condition, so they should be measured or observable unmeasured variables. The acquisition cost of the sensor network, CT, is selected as the objective function of the combinatorial optimization problem and is calculated as follows

$$\mathrm{CT} = \sum_{i=1}^{n} c_i q_i \tag{7}$$

where c_i is the cost of the flowmeter used to measure the ith streamflow. Furthermore, the features of the available flowmeters that can be located on the process are included in Table 4, where z is the true flow rate; c is the cost; and σ_s represents the measurement standard deviation, which is assumed as 2.5% of the true flow rate.

The results of four case studies are presented in Table 5. The following information is provided for each case: key variables; standard deviation bounds of reconciled variable estimates for a subset of key variables, along with the standard deviation of these estimates; the set of sensors; and the objective function value for the best solution. The parameters used for the EA runs are in accordance with the ranges recommended in the literature, and they are included in Table 6. If they are changed within their recommended ranges, the algorithm maintains its efficiency.

Regarding case 1, Figure 4 shows the best solutions for 100 runs with 100 generations each. The best solution (CT) is obtained in 67% of the runs. The average and dispersion of the objective function value are \$555.7468 and \$49.17, respectively, which indicates that good solutions are obtained in the execution of the

Table 5. Application Results of the Algorithm

case	key flow rates	σ^*	σ	measurements	CT
1	1, 2, 6	$\sigma_2^* = 0.025$ $\sigma_6^* = 1.7851$	$\sigma_2 = 0.025$ $\sigma_6 = 1.6552$	1, 2, 6, 7, 9, 10, 13, 20, 26, 28	533.56
2	2, 10, 28	$\sigma_{10}^* = 1.0482$ $\sigma_{28}^* = 1.4446$	$\sigma_{10} = 0.8861$ $\sigma_{28} = 1.435$	1, 2, 5−10, 13, 19, 20, 23, 26−28	894.86
3	4, 8, 17, 21, 23, 25	$\sigma_4^* = 2.1990$ $\sigma_8^* = 3.281$ $\sigma_{21}^* = 1.754$ $\sigma_{25}^* = 1.7090$	$\sigma_4 = 2.1857$ $\sigma_8 = 2.5644$ $\sigma_{21} = 1.5018$ $\sigma_{25} = 1.4864$	1, 4, 6, 7, , 9, 10, 11, 14, 16−24	752.26
4	4, 5, 7, 8, 12, 16, 18, 20, 27, 28	$\sigma_4^* = 2.1990$ $\sigma_5^* = 1.0654$ $\sigma_8^* = 3.2810$ $\sigma_{12}^* = 1.3454$ $\sigma_{27}^* = 1.4154$ $\sigma_{28}^* = 1.4446$	$\sigma_4 = 2.0368$ $\sigma_5 = 0.8878$ $\sigma_8 = 1.4967$ $\sigma_{12} = 0.9588$ $\sigma_{27} = 1.2002$ $\sigma_{28} = 1.4437$	1, 2, 4, 5−7, 9−11, 13, 15−24, 26−28	1178.06

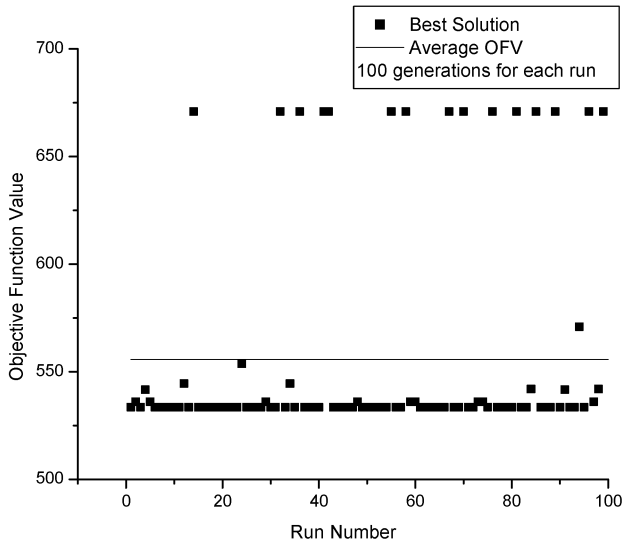

Figure 4. Objective function value for case 1.

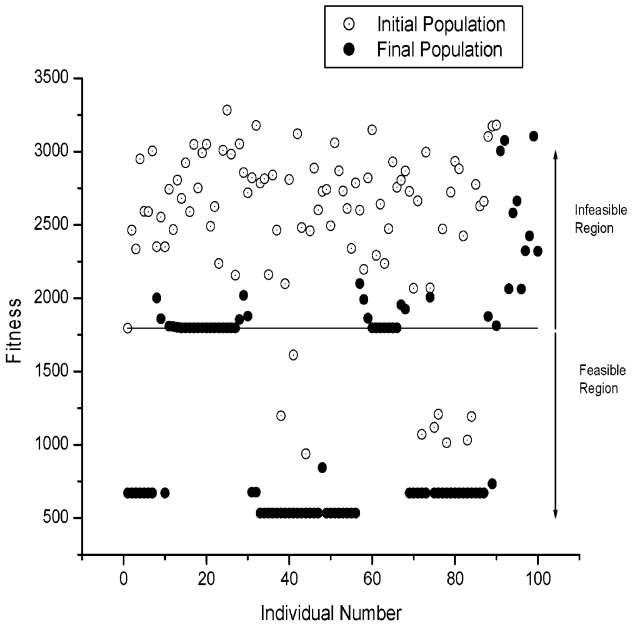

Figure 5. Population evolution for case 1.

Table 6. Parameters of the Evolutionary Algorithm

parameter	value
population size	100
probability MI = −1	0.4
probability MI = 1	0.3
selection method	Ranking
selection pressure	1.3
neighborhood size	7
mutation probability	0.025
crossover probability	0.7

remaining 33 runs. The initial and final populations after 100 generations for one run are represented in Figure 5. Even though the initial population is made up of a great number of infeasible individuals with respect to precision constraints, the algorithm evolves toward a feasible population composed by optimal-fitness individuals. Furthermore, it can be seen that individuals are agglomerated into suboptimal solutions, as evidence of the use of structured populations.

To compare the different behavior of the EA due to the application of a structured population, case 1 is run

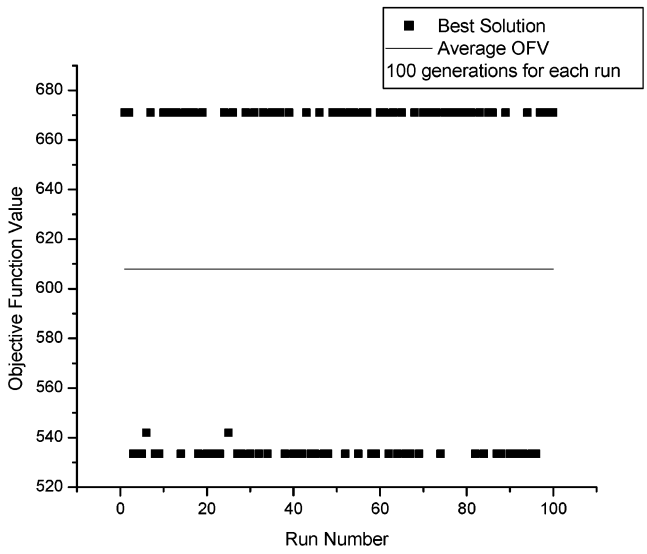

Figure 6. Objective function value for case 1 using a panmitic population.

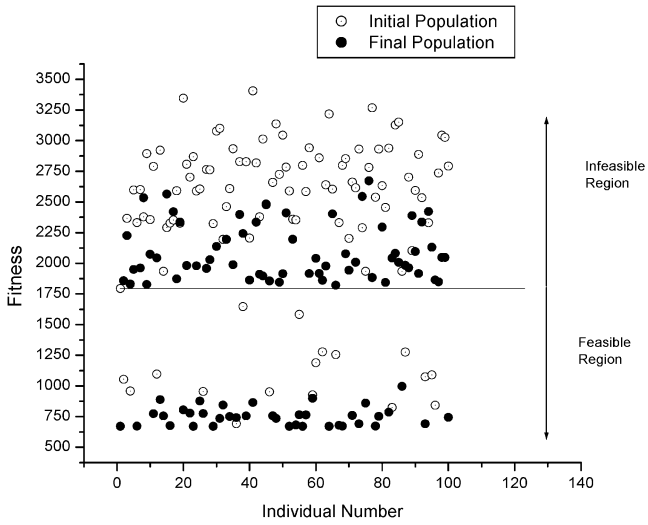

Figure 7. Population evolution for case 1 using a panmitic population.

with the same parameters but using a panmictic population. The best solution, CT = $533.56, is obtained in 44% of the runs, and CT = $671 in 54%. The average and dispersion of the objective function value are $607.95 and $68.67, respectively. In this case, the performance of the algorithm is poor, and premature convergence problems are detected, as is shown in Figures 6 and 7.

The set of key variables for case 2 consists of the flow rates of distant streams. One hundred runs are performed with 200 generations each. Good repeatability of the best solution is observed in Figure 8. The solution distribution has a mean value of $943.158 and a standard deviation of $89.71. The best solution, CT = $894.86, is obtained in 56 runs. Figure 9 shows that, from an infeasible population with respect to precision, the algorithm evolves to regions of good fitness.

In case 3, a better ratio of feasible to infeasible individuals exists in the initial population, which enhances convergence to the best solution (CT = $752.26). This solution is obtained in 84% of runs; the average objective function value is $752.66, and the standard deviation of the solution distribution is very low, $ 0.9211. Figures 10 and 11 represent the algorithm

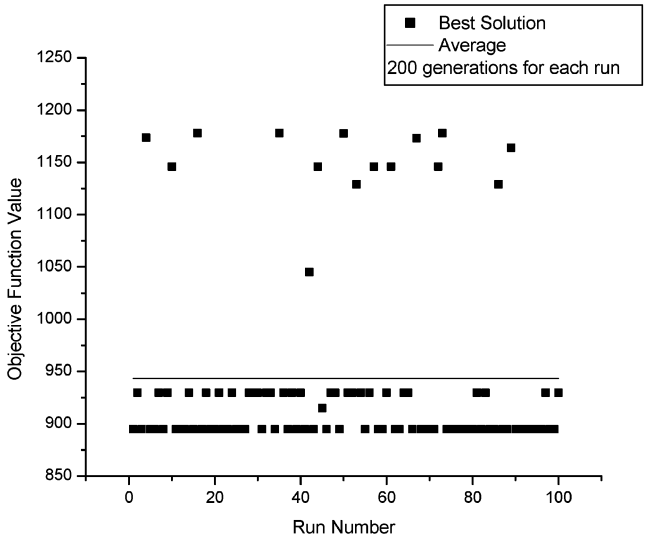

Figure 8. Objective function value for case 2.

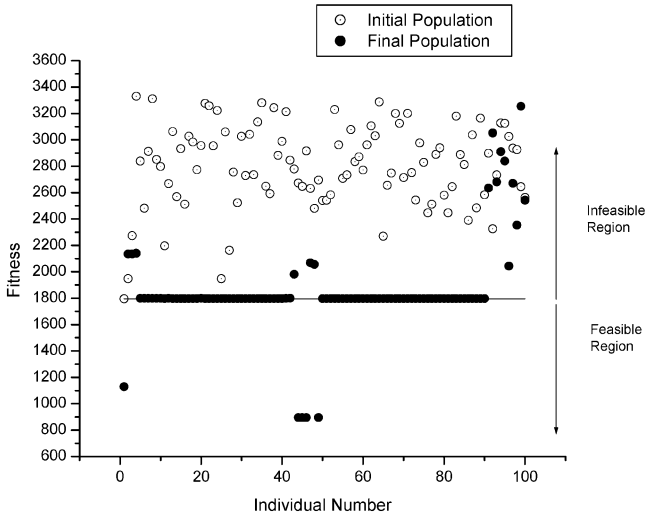

Figure 9. Population evolution for case 2.

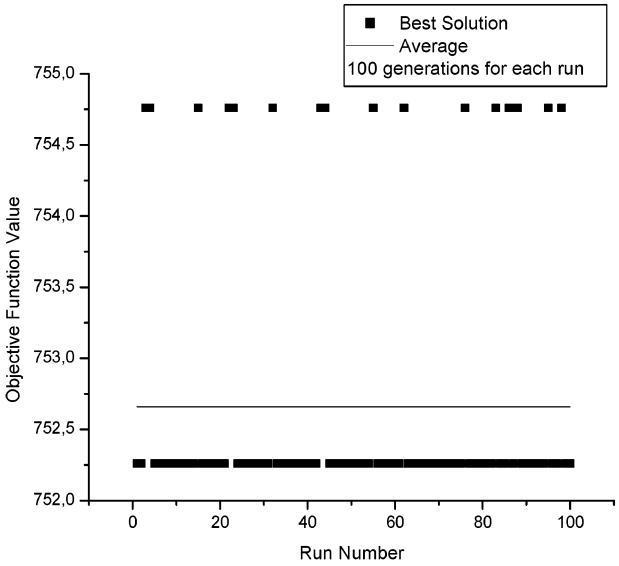

Figure 10. Objective function value for case 3.

behavior for this case study. Results for case 4 (Figures 12 and 13) show similar behavior of the technique for a great number of key flow rates. The best solution is obtained for all the runs.

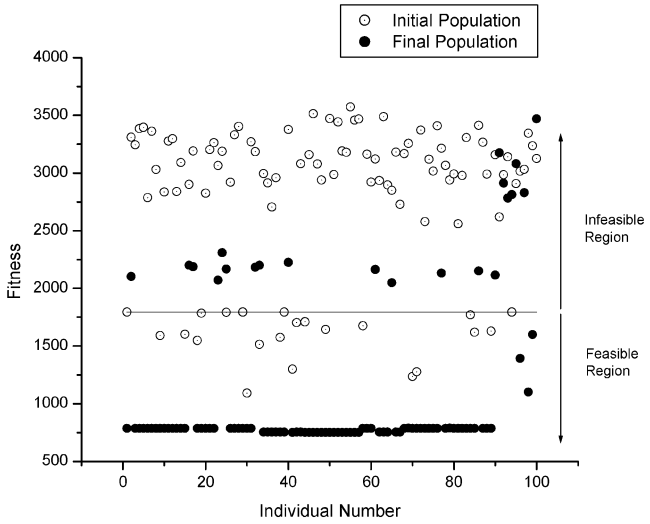

Figure 11. Population evolution for case 3.

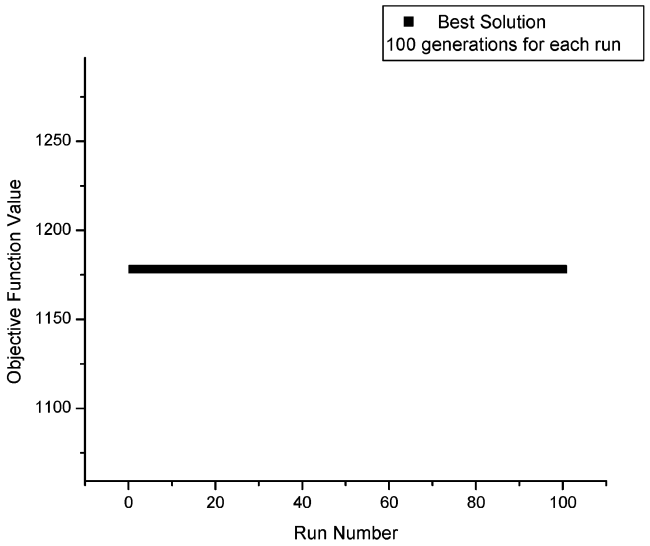

Figure 12. Objective function value for case 4.

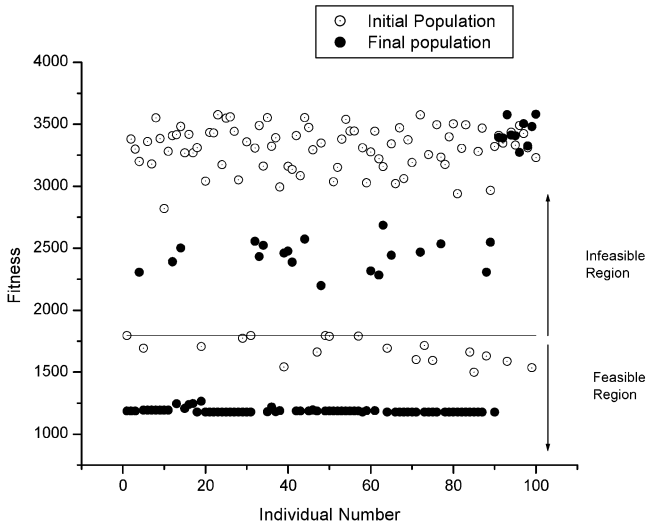

Figure 13. Population evolution for case 4.

The following conclusions arise from the analysis of results:

(a) The proposed EA evolves toward the best solution, even though it starts from an initial population constituted by a great number of infeasible individuals, in a

Table 7. Sensitivity Analysis of Measurements' Standard Deviation on the Design for Case 3

flow rate	σ_s	σ_s^n	measurements	CT
4	2.749	$1.1\sigma_s$	1−3, 7, 9−11, 13−17, 19−24	808.96
		$1.2\sigma_s$	1−3, 7, 9−11, 13−17, 19−24	808.96
6	2.807	$1.1\sigma_s$	1, 2, 4, 7, 9−11, 13, 14, 16−24	754.76
		$1.2\sigma_s$	1, 2, 4, 7, 9−11, 13, 14, 16−24	754.76
10	1.31	$1.1\sigma_s$	1, 4, 6, 7, 9−11, 14, 16−24	752.26
		$1.2\sigma_s$	1, 4, 5, 6, 7, 9, 11, 14, 16−24	754.76
		$1.3\sigma_s$	1, 4, 5, 6, 7, 9, 11, 14, 16−24	754.76
14	2.296	$1.1\sigma_s$	1, 4, 6, 7, 9−11, 14, 16−24	752.26
		$1.2\sigma_s$	1, 4, 6, 7, 9−11, 14, 16−24	752.26
		$1.3\sigma_s$	1, 4, 6, 7, 9−11, 14−17, 19−24	803.46
		$1.4\sigma_s$	1, 4, 6, 7, 9−11, 14−24	823.86
		$1.6\sigma_s$	1, 2, 4, 7, 9−11, 13−24	826.36
		$2.0\sigma_s$	1, 2, 3, 4, 7, 9−11, 13−24	958.56
21	2.182	$1.1\sigma_s$	1, 4, 6, 7, 9−11, 14, 16−24	752.26
		$1.2\sigma_s$	1, 2, 4, 7, 9−11, 13, 14, 16−24	754.76
		$1.3\sigma_s$	1, 4, 6, 7, 9−11, 14−24	823.26
		$1.4\sigma_s$	1, 4, 6, 7, 9−11, 14−24	823.26
		$1.5\sigma_s$	4, 7, 9−11, 14, 16−24, 26−28	879.96

reasonable time (approximately 140 s using a 1 GHz Pentium III PC).

(b) The ratio feasible/infeasible individuals has an influence on the number of generations needed to obtain the best solution.

(c) A great percentage of repeatability of the best solution is achieved.

(d) The neighborhood selection model provides a good balance between the exploration of the search space and the exploitation of good solutions that outperforms the panmictic model, which is entrapped in local suboptima.

Furthermore, a sensitivity analysis of standard deviations of the measurements was performed. For each flowmeter, the standard deviation, σ_s, was replaced by a new value, σ_s^n, and case 3 was solved again using the proposed algorithm. For some sensors, the solution of the optimization problem changed as a consequence of the modification of the standard deviations. These sensors are the flowmeters available to measure the flows of streams 4, 6, 10, 14, and 21. Table 7 presents the results of the new runs that have a different solution from the original case 3.

The optimum sensor network design under study is constrained by the precision of key variable estimates. It is well-known that, to satisfy the required precision after data reconciliation procedures are applied, the number of sensors in the network and the cost increase with increasing standard deviation of the sensors. Consequently, those sensors of higher standard deviation whose measurements participate in the alternative ways of calculating a key flow rate have more influence on the design.

5. Conclusions

In this paper, a hybrid solution technique is presented for the optimal design or upgrade of linear sensor networks, subject to quality constraints on key variable estimates. No restrictions are imposed on the mathematical nature of the objective function or constraints.

The proposed EA is robust and efficient. It converges to the best solution with high repeatability, and a great part of the final population is feasible and has evolved toward the best solution. This indicates that the neighborhood selection model of the GA enhanced by local search succeeds in providing a good balance between the algorithm's exploration and exploitation capabilities.

Nomenclature

$\mathbf{A}_1$ = submatrix of $\mathbf{D}$ associated with measured variables
$\mathbf{A}_2$ = submatrix of $\mathbf{D}$ associated with unmeasured variables
b = random integer
c = sensor cost
CT = total instrumentation cost
$\mathbf{D}$ = incidence matrix
f = objective function
F = fitness
g = constraint
I_0 = initial instrumentation set
m = number of units
MI = measurement index
n = number of streams
N = population size
$\mathbf{q}$ = binary vector
Q = constraint violation function
r_v = number of required variables
r = rank of $\mathbf{A}_2$
R = number of unsatisfied constraints
S_J = set of key process variables
$\mathbf{T}$ = chromosome representation to generate the initial population
$\mathbf{u}$ = vector of unmeasured flow rates
$\mathbf{x}$ = vector of measured flow rates
$\mathbf{z}$ = vector of flow rates

Greek Letters

η_{max} = expected number of offspring of the best individual
σ_s = sensor standard deviation
σ = standard deviation of variable estimates
ν = size neighborhood

Symbols

$|\cdot|$ = set cardinality

Literature Cited

(1) Madron F. *Process Plant Performance. Measurement and Data Processing for Optimization and Retrofits*: Ellis Horwood Ltd.: Chichester, U.K., 1992.

(2) Ali, Y.; Narasimhan, S. Sensor Network Design for Maximizing Reliability of Linear Processes. *AIChE J.* **1993**, *39*, 820−828.

(3) Bansal, P.; Ali, Y.; Narasimhan S. Sensor Network Design for Linear Processes. Presented at the IFAC Workshop on Integration of Process Design and Control, Baltimore, MD , Jun 27−28, 1994.

(4) Ali, Y.; Narasimhan, S. Sensor Network Design for Maximizing Reliability of Bilinear Processes. *AIChE J.* **1996**, *42*, 2563−2575.

(5) Héraud, N.; Mazzour, E. Sensor Network Design for Process Described by Linear−Bilinear Equations. *Proc. Inst. Mech. Eng.* **1999**, 213 (Part I), 321−329.

(6) Sen, S.; Narasimhan, S.; Deb, K. Sensor network design of linear processes using genetics algorithms. *Comput. Chem. Eng*. **1998**, *22*, 385−390.

(7) Viswanath, A.; Narasimhan, S. Multi-objective Sensor Network Design Using Genetic Algorithms. In *4th IFAC Workshop on On-Line Fault Detection and Supervision in the Chemical Process Industries*; Elsevier Science: New York, 2001; p 265.

(8) Carnero, M.; Hernández, J.; Sánchez, M.; Bandoni, A. An Evolutionary Approach for the Design of Nonredundant Sensor Networks. *Ind. Eng. Chem. Res.* **2001**, *40*, 5578−5584.

(9) Kretsovalis, A.; Mah, R. Effect of Redundancy on the Estimation Accuracy in Process Data Reconciliation. *Chem. Eng. Sci.* **1987**, *42*, 2115−2121.

(10) Ali, Y.; Narasimhan, S. Redundant Sensor Network Design for Linear Processes. *AIChE J.* **1995**, *41*, 2237−2249.

(11) Madron, F.; Veverka, V. Optimal Selection of Measuring Points in Complex Plants by Linear Models. *AIChE J.* **1992**, *38*, 227−236.

(12) Meyer M.; Le Lann, J.; Koehret, B.; Enjalbert, M. Optimal Selection of Sensor Location on a Complex Plant Using a Graph Oriented Approach. *Comput. Chem. Eng*. **1994**, *18* (Suppl), S535−S540.

(13) Luong, M.; Maquin, D.; Huynh, C.; Ragot, Observability, Redundancy, Reliability and Integrated Design of Measurement Systems. Presented at the 2nd IFAC Symposium on Intelligent Components and Instrument Control Applications, Budapest, Hungary, Jun 8−10, 1994.

(14) Bagajewicz, M. Design and Retrofit of Sensor Networks in Process Plants. *AIChE J.* **1997**, *43*, 2300.

(15) Chmielewski, D.; Palmer, T. E.; Manousiouthakis, V. Cost Optimal Retrofit of Sensor Networks with Loss Estimation Accuracy. Presented at the AIChE Annual Meeting, Dallas, TX, Oct 31−Nov 5, 1999.

(16) Bagajewicz, M.; Cabrera, E. A New MILP Formulation for Instrumentation Network Design and Upgrade. In *4th IFAC Workshop on On-Line Fault Detection and Supervision in the Chemical Process Industries*; Elsevier Science: New York, 2001; pp 303−308.

(17) Bagajewicz, M.; Cabrera E. Pareto Optimal Solutions Visualization Techniques for Multiobjective Design and Upgrade of Instrumentation Networks. *Ind. Eng. Chem. Res.* **2003**, *42*, 5195−5203.

(18) Bagajewicz, M.; Cabrera, E. New MILP Formulation for Instrumentation Network Design and Upgrade. *AIChE J.* **2002**, *48*, 2271−2282.

(19) Chao-An, L.; Chuei-Tin, K.; Chin-Leng; Chen-Liang, C. Optimal Sensor Placement and Maintenance for Mass-Flow Networks. *Ind. Eng. Chem. Res.* **2003**, *42*, 4366.

(20) Bagajewicz, M. *Design and Upgrade of Process Plant Instrumentation*; Technomic Publishing Company: Lancaster, PA, 2000.

(21) Bagajewicz, M. Review of Recent Results in Instrumentation Design and Upgrade for Process Plants. In *IFAC Workshop on On-Line Fault Detection and Supervision in the Chemical Process Industries*; Elsevier Science: New York, 2001; pp 259−264.

(22) Bagajewicz, M. A Review of Techniques for Instrumentation Design and Upgrade in Process Plants. *Can. J. Chem. Eng*. **2002**, *80*, 3−16.

(23) Bagajewicz, M. Data Reconciliation and Instrumentation Upgrade. Overview and Challenges. In *Proceedings of Foundations of Computer-Aided Process Operations (FOCAPO 2003): A View to the Future Integration of R&D, Manufacturing and the Global Supply Chain*; Grossmann, I., McDonald, C., Eds.; CACHE Corporation: Austin, TX, 2003; pp 103−116.

(24) Sánchez, M.; Bagajewicz, M. On the Impact of Corrective Maintenance in the Design of Sensor Network. *Ind. Eng. Chem. Res.* **2000**, *39*, 977.

(25) Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*; Springer-Verlag: New York, 1996.

(26) Alba, E.; Tomassini, M. Parallelism and Evolutionary Algorithms. *IEEE Trans. Evol. Comput*. **2002**, *6*, 443−462.

(27) Goldberg, D. *Genetic Algorithms in Search, Optimization, and Machine Learning*; Addison-Wesley: New York, 1989.

(28) Carnero M.; Hernández, J.; Sánchez, M.; Bandoni, A. A Strategy for Sensor Network Design Based on an Evolutionary Approach. Presented at the AIChE Annual Meeting, Dallas, TX, Oct 31−Nov 5, 1999.

(29) Deb, K. An Efficient Handling Method for Genetic Algorithms. *Comput. Methods Appl. Mech. Eng.* **1998**, *186*, 311.

(30) Arts, E.; Lenstra, J. *Local Search in Combinatorial Optimization*; John Wiley & Sons: New York, 1997.

(31) Romagnoli, J.; Sánchez, M. *Data Processing and Reconciliation for Chemical Process Operations*; Academic Press: San Diego, 1999.

(32) Serth, R.; Heenan, W. Gross Error Detection and Data Reconciliation in Steam-Metering Systems. *AIChE J.* **1986**, *32*, 733−741.

Received for review February 20, 2004
Revised manuscript received October 11, 2004
Accepted November 1, 2004

IE049856T