I&EC research
Industrial & Engineering Chemistry Research

Article

Cite This: Ind. Eng. Chem. Res. 2018, 57, 12149−12164

pubs.acs.org/IECR

Optimal Sensor Location in Chemical Plants Using the Estimation of Distribution Algorithms

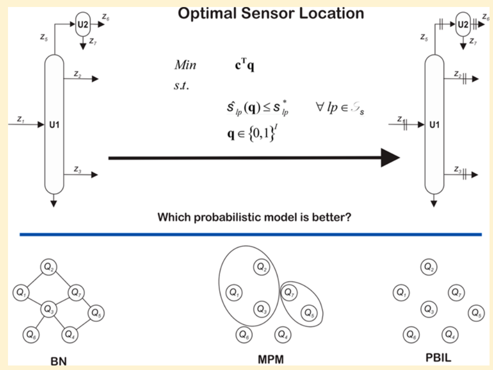
Mercedes Carnero,[†] José L. Hernández,[†] and Mabel Sánchez*[,‡]

[†]Departamento Ciencias Básicas, Facultad de Ingeniería, Universidad Nacional de Río Cuarto, Campus Universitario, 5800 Río Cuarto, Argentina

[‡]Departamento de Ingeniería Química, Universidad Nacional del Sur (UNS) and Planta Piloto de Ingeniería Química - PLAPIQUI (UNS-CONICET), 8000 Bahía Blanca, Argentina

Supporting Information

ABSTRACT: The optimal selection of sensor structures improves the knowledge of the current plant state, which is a central issue for the decision-making process. Instrumentation design is a challenging optimization problem that involves a huge amount of binary variables that represent the possible sensor locations. In this work, the limitations of the current design strategies are discussed, and they support the application of evolutionary solution methods. Among them, the estimation of distribution algorithms (EDAs) arises as a convenient alternative to solving the problem. These are stochastic optimization strategies devised to capture complex interactions among problem variables by learning the probabilistic model of candidate solutions and its sampling to generate the next population. From the broad spectrum of EDAs that use multivariate models, two representative procedures are selected that significantly differ in the methods used for learning and sampling those models. Furthermore, a comparative performance study is conducted to evaluate the benefits of increasing the complexity of the distribution model with respect to a memetic procedure based on univariate models.



1. INTRODUCTION

The design of the instrumentation system of a chemical plant is a complex multilevel task that is composed of the definition of the global objectives, the selection of the measured variables, and the specification of the implementation details, such as measurement intervals, sample procedures, interfaces, maintenance activities, etc.

During the online operation, the quality and availability of process knowledge essentially depends on the instrumentation selection performed on the second design stage. The structure of a sensor network (SN) is defined by the type, amount, precision, reliability, and location of its instruments.

To determine the optimal SN that minimizes the total instrumentation cost and simultaneously satisfies the quantity and quality of the required information, a combinatorial optimization problem is formulated that involves binary variables.[1] Because the dimension of the search space increases significantly for large-scale processes, a challenging optimization problem arises that may have many local optima. Its solution has been tackle using exact and stochastic algorithms. Comprehensive reviews of these methods can be found elsewere.[2−5]

Diverse exact procedures have been proposed to solve the sensor network design problem (SNDP) but they do not guarantee that the solution can be attained in polynomial time for any instance of the problem. The tree search and mathematical programming techniques are representative of the exact solution procedures. The last ones are appropriate if the constraints of the design problem can be explicitly defined in terms of the binary variables, and the mathematical problem can be associated with a certain formulation, e.g., mixed integer linear or nonlinear programming problems arise.

Stochastic solution procedures provide a good balance between the quality of the solution and the computational resources for any instance of the problem even though the global optima may not be attained. Techniques based on genetic algorithms (GAs) were proposed initially. In this sense, the GA toolbox of the MATLAB program[6] and parallel versions of the classic GA[7] were used. An algorithm that combined a structured population in the form of neighborhoods and a local optimizer of the best current solutions was also presented.[4] Furthermore, a design strategy within the framework of Tabu search (TS) that used a strategic oscillation (SO) technique around the feasibility boundary, called SOTS, was developed.[8] It significantly reduced the number of required calls to the evaluation function in comparison with previous methodologies.

Recently a metaheuristic approach based on the estimation of distribution algorithms (EDAs) and SOTS was presented.[5]

Received: April 17, 2018
Revised: August 16, 2018
Accepted: August 18, 2018
Published: August 20, 2018

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

The solution scheme made use of the population-based incremental learning algorithm (PBIL) developed by Baluja,[9] which assumed independent relationships among variables. Application results of that procedure, called PBIL-SOTS, demonstrated the synergistic effect of the combination of EDAs and SOTS advantages on the solution of the SNDP.

According to the model complexity, EDAs can be broadly divided into univariate, bivariate, or multivariate approaches.[10] To handle variable interdependencies, the second and third classes of EDAs require complex learning algorithms and significant additional computational resources. The question of deciding on the type of model to be used for a given problem is not solved.

In contrast to any previous work, the applicability of different approaches for solving a standard formulation of the SNDP is analyzed using case studies of different size and complexity. At first, the behavior of exact procedures is discussed based on the results obtained using a tree-search algorithm. The scalability analysis of those procedures allows the understanding of the necessity of applying evolutionary optimization approaches for solving large-scale design problems. The EDAs have demonstrated a rewarding performance for this purpose, but the benefits and drawbacks of using complex probabilistic models have not been addressed before. From the broad spectrum of multivariate EDAs, two representative procedures are selected that strongly differ in the methods used for learning and sampling the probabilistic model. One of them is the affinity EDA, AffEDA,[11] whose models are constructed using the mutual information between pairs of variables and an affinity propagation algorithm. The other one is the estimation of bayesian network algorithm, EBNA,[12] which determines the Bayesian network (BN) structure that optimizes the Bayesian Network Criterion score from the database containing the selected individuals at each generation. Furthermore, a comparative performance study is conducted to evaluate the benefits of increasing the complexity of the distribution model with respect to the memetic procedure PBIL-SOTS.

This work is structured as follows. In section 2, a standard SNDP is formulated. Then, in section 3 an exact procedure is applied for solving problems of incremental size. Complex EDAs are introduced in section 4. In section 5, results are provided and discussed for various case studies presented in the literature. Section 6, the conclusion section, ends the article.

2. SENSOR-NETWORK DESIGN PROBLEM

The minimum-cost SNDP that satisfies precision and estimability constraints for a set of key variable estimates is stated by eq 1, where $\mathbf{q}$ is an I-dimensional vector of binary variables such that $q_i = 1$ if variable i is measured, and $q_i = 0$ otherwise, $\mathbf{c}^{\mathrm{T}}$ is the cost vector; $\hat{\sigma}_k$ is the estimate standard deviation of the k-th variable contained in S_σ after a data reconciliation procedure is applied,[13] and E_l stands for the degree of estimability of the l-th variable included in S_{E}.[13] Furthermore, S_σ and S_{E} are the set of key process variables with requirements in precision and ability to be estimated, respectively:

$$\begin{aligned} \min \quad & \mathbf{c}^{\mathrm{T}}\mathbf{q} \\ \text{s.t.} \quad & \\ & \hat{\sigma}_k(\mathbf{q}) \le \sigma_k^*(\mathbf{q}) \quad \forall\, k \in S_\sigma \\ & E_l(\mathbf{q}) \ge 1 \qquad\quad \forall\, l \in S_{\mathrm{E}} \\ & \mathbf{q} \in \{0, 1\}^I \end{aligned} \tag{1}$$

In this formulation, it is assumed that a linearized algebraic model represents plant operation, measurements are subject to noncorrelated random errors, there is only one potential measuring device for each variable, and there are no restrictions for the localization of instruments. Regarding the definition of degree of estimability, let us first denote $\mathbf{A}(p)$ as the set of all possible combinations of p measurements, and call $A_j(p)$, the j-th element (combination) of this set. The l-th variable (measured or not) has a degree of ability for estimation E_l, if it remains estimable after the elimination of any combination $A_j(E_l - 1) \in \mathbf{A}(E_l - 1)$ and it becomes unobservable when at least one set, $A_j(E_l) \in \mathbf{A}(E_l)$, is eliminated.[14] If E_l is set equal to one, the feasibility of the constraint can be checked by executing a variable classification procedure, which can be accomplished by matrix projection, QR decomposition, or matrix co-optation.[15,16]

3. EXACT SOLUTION PROCEDURES

The exact solution procedures provide the global optimal solution of the SNDP. Up to the present time, diverse algorithms have been presented that belong to that cathegory, but any of them guarantees attaining the global optimum for any instance size in polynomial time. In general, two types of exact solution procedures can be distinguished. The first one considers the unmeasured variables as infinite-variance measurements to explicitly calculate certain network performance measures in terms of $\mathbf{q}$. Then available commercial software packages are applied to solve the resulting SNDPs. In general, mixed-integer nonlinear programming, MINLP, problems arise.[17−19]

The tree search is the second approach used to obtain the global optimum of the SNDP. That versatile method does not use any transformation of the problem into well-established optimization formulations. It only requires that the solution set can be divided into mutually exclusive subsets and that a lower bound of the objective-function can be estimated for any solution of a given subset. If these conditions are satisfied, the tree search allows the calculation of the optimal solution of optimization problems that involve objective functions and constraints of any type. The tree search is a branch and bound methodology without node relaxation that performs a smart exploration of the tree using appropriate bound definitions and stopping criteria. Next, an exact solution procedure, called level traversal tree search, LTTS,[2] is briefly reviewed. It efficiently combines depth and wide tree searches to obtain minimum cost-sensor structures.

At first, the LTTS sorts the sensors in ascending order of cost, i.e., $c\ (i) < c(i + 1)$ $(i = 1 \cdots I - 1)$ and generates a tree whose base unit may be a single measurement. Starting with a root node with no variables being measured, at each level one extra element of $\mathbf{q}$ is made active. The tree structure is generated considering that the descendents of any node are sorted in ascending order of cost from the left to the right. This allows the disregarding of the set of solutions and family of

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

nodes when the wide search is performed. If $u_p = \mathbf{c}^T\mathbf{q}^*$ is the upper bound, where $\mathbf{q}^*$ is the better current solution, then the branching is stopped if (a) the cost of the current node is greater than u_p, i.e., $c_c^n > u_p$; (b) the current node is feasible. In this case, if $c_c^n < u_p$; the value u_p is updated as follows: $u_p = c_c^n$.

The algorithm LTTS proceeds as follows:

(1) A standard depth-first tree search is performed until a pre-specified number of nodes is inspected. That number depends on the size problem. The best current solution ($\mathbf{q}^*$), the level where it was found (i), and the number of measured variables in $\mathbf{q}^*$ are saved. Up to this point, all the nodes located to the left of $\mathbf{q}^*$ and their sons are inspected.

(2) If $\mathbf{q}^*$ is the cheapest node visited of the i-th level, then the parent of $\mathbf{q}^*$, which belongs to the level $i - 1$, is identified and the nodes located to the right of the $\mathbf{q}^*$ parent are inspected. In contrast, the examination of the nodes located to the right of $\mathbf{q}^*$ follows.

(3) For each family, the node that satisfies the stopping criteria is identified. If it is the cheapest node of the family, all of the nodes located to the right of it that share the same parent are disregarded. If it is not the case, the next families should be examined.

(4) If all of the nodes of the current level are examined or disregarded, then the upper level is inspected.

(5) If all of the nodes of a given level are infeasible, then the procedure stops.

The methodology previosuly described is applied to locate the flowmeters of a process that involves 11 units and 28 streams (case study 1). It is assumed that variables are related only by mass balance equations. The standard deviation of flowmeters is 2.5% of the true variable values. Table 1 shows the complexity of the set of constraints imposed for case studies 1.a and 1.b. More information regarding process variables is required for the second design; therefore, the number of constraints of the optimization problem increases. Table 2 reports the global optimum solutions and the computational time required to solve both cases studies when the procedure was executed using a Processor Intel Core i7 CPU 920 @ 3.40 GHz, 8GB RAM, using MatLab. Results show that the computational load is high even though the instance size is relatively small.

Table 1. Set of Constraints

case study	constraints
1.a	$E_{le} \geq 1$ for streams 17 and 23
	$\sigma_4^* = 2.199$, $\sigma_8^* = 3.281$, $\sigma_{21}^* = 1.754$, and $\sigma_{25}^* = 1.709$
1.b	$E_{le} \geq 1$ for streams 7, 16, 18, and 20
	$\sigma_4^* = 2.199$ $\sigma_5^* = 1.065$ $\sigma_8^* = 3.281\sigma_{12}^* = 1.345$ $\sigma_{27}^* = 1.415\sigma_{28}^* = 1.445$

Table 2. Global Optimum Solutions

case	optimal solution	cost	computational time (h)
1.a	1, 4, 6, 7, 9–11, 14, 16–24	752.26	85.6
1.b	1, 2, 4–11, 13, 16–24, 26–28	1106.50	106.5

The computation time of the described methodology depends on where the global optimum is located in the tree search. If the SNDP involves I binary variables, the complete tree search is composed of I levels, and the number of nodes of the i-th level is equal to $\binom{I}{i}$. This value increases exponentially with I, and its maximum is approximately obtained for $I/2$. If the number of measured variables of the optimal solution is around $I/2$, then it may be intractable to obtain it in sensible times due to the huge amount of nodes that should be visited. Recall that the procedure ends when all of the nodes of the level where the solution is and those belonging to the previous level have been explored. For example, the optimal solution of Case 1.b corresponds to the 22th level. The number of nodes of the 21th level is of the order of 10^6, while the whole search space is 2^{28}, i.e., of the order of 10^8. In this case, the computational time required to get the optimal solution was 106.5 h (4 days and 10.45 h).

Previous results show that optimal solutions that are composed of approximately $I/2$ measured variables are a bottleneck for the methodology. In consequence, the guarantee of optimality that techniques based on tree search ensure cannot be attained in practice when I increases, and, depending on the optimization problem complexity, the computational time constitutes a limiting factor to select the most appropriate solution strategy. This problem is shared by many combinatorial optimization problems, and this motivated that metaheuristics have been developed. They propose different mechanisms to provide good solutions in limited time.

Genetic algorithms have been used to solve the SNDP. In general the performance of the standard GA is poor, this is why many techniques presented to tackle that problem include some type of enrichment strategy that helps them to find better solutions. The main inconvenient of GAs is that they loose genotypic material in the evolution procedure. For the SNDP, that means GAs loose measured variables associations that are repeated in good quality solutions. They may be formed at the earlier stages of the algorithm but they disappear latter due to the cross-over operator. This disadvantage has motivated the development of other evolutionary algorithms (EA). In this sense the EDAs have significantly improved the performance of the classic GA.[10,20] In the next section, the basic theoretical concepts related to EDAs are revisited.

4. ESTIMATION OF DISTRIBUTION ALGORITHMS

Evolutionary algorithms based on probabilistic models are distinguished as a new computing paradigm in evolutionary computation. The EDAs adopt an evolutionary mode for searching the best solutions that consists olf building a probabilistic model about the distribution of good individuals in the search space and then sampling that model to build the next population. In this fashion, EDAs succeed to attain the required knowledge for approaching the global optimum in the search space step by step.

The relationships among variables are explicitly considered in EDAs by means of the probabilistic model associated with the individuals of each generation. The estimation of that model constitutes one of the main issues of EDAs. According to the model complexity, EDAs can be broadly divided into three categories: univariate, bivariate, and multivariate approaches.[10]

Regarding discrete EDAs, the univariate techniques assume that all variables are independent. Thus, the joint probability of a candidate solution is decomposed into the product of the marginal probabilities of individual variables. Other EDAs are capable of capturing some pair-wise interactions between variables using tree-based models, in which the conditional

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149–12164

probability of a variable may only depend on, at most, one other variable, its parent in a tree structure. Univariate and bivariate EDAs can be efficiently applied to separable problems or to problems with low degrees of dependency among the variables. However, those approaches might rapidly lose their efficiencies when variable interactions increase. Thus, multivariate EDAs are devised to capture a broad variety of possible relations among variables, but their modeling flexibility comes at the expense of an extra computational work.

Among multivariate EDAs, there exist procedures based on marginal product models (MPM). In this sense, the methodology of the extended compact genetic algorithm, EcGA,[21] factorizes the joint probability distribution into a number of marginal distributions defined over non-overlapping clusters of variables. At each generation, the model building starts assuming that all the variables are independent. Then the two clusters whose merging most improves the quality of the model, measured in terms of the minimum description length, are jointed. This procedure is repeated until no link of two clusters improves that metric. Next, a probability table is computed for each cluster using a set of selected solutions, and the new population is generated by sampling each group. The AffEDA[11] also uses MPM. Given a set of points and a measure of similarity, that procedure forms groups of similar points without fixing in advance the number of clusters. After this stage of model building, it is sampled in the same way as EcGA does.

Many problems contain highly overlapping subproblems that cannot be accurately modeled by dividing the problem into an independent set of variables. In these cases, Bayesian networks, BNs, can be used. They allow us to encode probability distributions through a structure, which expresses explicit independence relations among variables and a set of parameters. In contrast to MPM, BNs are capable of capturing more complex problem decompositions in which subproblems interact. The EBNA uses the Bayesian information criterion to evaluate network structures in the greedy network construction algorithm. That metric contains a strong implicit bias toward simple models; therefore, it does not require a limit on the number of allowable parents or any prior bias toward simpler models.[22,23]

In addition, Markov networks can encode multivariate interactions among variables. The structure of these networks is similar to the Bayesian ones except that the connections between variables are undirected. A Markov network that ensures convergence to the global optimum may sometimes be considerably less complex than an adequate BN, at least with respect to the number of edges, but its sampling is more difficult.[24]

Regarding BNs, the theoretical and practical issues of their application for solving benchmarks of combinatorial optimization problems have been extensively analyzed in the literature. Because there exist a remarkable effort to develop EDAs based on BNs, EBNA was selected to solve the SNDP for the first time even though the structural learning of the probabilistic model by BNs is a non-deterministic polynomial-time hard (NP-hard) problem, and its applicability may be restricted to medium-size problems. Because this work also aims at studying multivariate EDAs that strongly differ in the methods used for learning and sampling the probabilistic model, a representative technique of MPM, the methodology AffEDA, is selected for comparative purposes. It shares one common feature with all MPM; that is, the joint probability distribution is decomposed into marginal distributions. Next brief descriptions of the solution schemes devised to solve Problem 1 using AffEDA and EBNA are presented, and their pseudocodes are included in the Supporting Information.

4.1. Affinity Estimation of Distribution Algorithm.

When an EDA is used to solve Problem 1, a solution vector $\mathbf{q} = \{q_1, q_2, \cdots, q_I\}$ is considered a realization of an I dimensional random vector $\mathbf{Q} = \{Q_1, Q_2, \cdots, Q_I\}$, where Q_i is a binary variable. A graphical probabilistic model of $\mathbf{Q}$ provides a factorization of the joint probability distribution of $\mathbf{Q}$, denoted as $p_{\mathbf{Q}}$ that represents $\Pr(\mathbf{Q} = \mathbf{q}) = p_{\mathbf{Q}}(\mathbf{q})$. Furthermore, if the marginal distribution function of each unidimensional random variable Q_i is p_i, then $\Pr(Q_i = q_i) = p_i(q_i)$.

The AffEDA[11] generates a random population of M individuals, which are evaluated using a fitness function. At each generation a tournament selection is applied to choice a t percent of the individuals of the current population, which are used for the structural learning.

The AffP proposed by Frey and Dueck[25] is applied to group the variables into nonoverlapping sets of strong interacting variables; each set constitutes a factor of the probability factorization.The technique considers the mutual information matrix, **MI**, between the variables as the similarity measure. Given two random variables Q_x and Q_y, the mutual information between them, $IM\ (Q_x, Q_y)$, is defined as follows:

$$IM(Q_x, Q_y) = \sum_{q_x} \sum_{q_y} p(q_x, q_y) \log_2\left(\frac{p(q_x, q_y)}{p(q_x)p(q_y)}\right) \quad (2)$$

where $p(q_x)$ and $p(q_y)$ are the values of the marginal probability distributions of variables Q_x and Q_y, respectively, and $p(q_x, q_y)$ represents the value of the joint probability distribution of the random vector $[Q_x\ Q_y]$; that is, $p(q_x, q_y) = \text{probability}(Q_x = q_x, Q_y = q_y)$. If Q_x and Q_y are independent, $IM\ (Q_x, Q_y) = 0$ because the joint probability distribution of these variables is equal to the product of their marginal probability distributions.

The clustering procedure works by exchanging messages between points until a stop condition is fulfilled. A responsibility message is sent from a point to a candidate exemplar and indicates the accumulated evidence of the suitability of the message receiver to become the exemplar of the data point, taking into account the other potential exemplars of that point. An availability message is sent from a candidate exemplar to a data point and measures how appropriate would be for the point to choose the message sender as its exemplar, given the support provided by other points that the sender should be an exemplar. Data points receive availability messages from different candidate exemplars, while these collect responsibility messages from diverse data points.

After the structural model is learned, the frequency tables are calculated and the model building stage finishes. Next, the population of the next generation, which contains $M - k$ individuals chosen by sampling the probabilistic model and the best k individuals of the current population, is formed. The stages of selection, probabilistic model learning, and sampling are repeated until the maximum number of generations is reached.

For illustrative purposes, let us consider a process constituted by eight streams and five units. In this case, the dimension of the random vector $\mathbf{Q}$ is eight. For a selected

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

population of 11 individuals, Figure 1 shows the **MI** and the clusters obtained from the application of the AffP. The

(a)

# ind	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	Q_8
1	0	1	1	0	0	0	0	1
2	0	1	1	0	1	1	0	0
3	1	0	1	0	0	1	0	0
4	0	1	0	0	1	1	1	0
5	0	0	1	1	1	0	0	0
6	1	0	0	1	0	0	0	1
7	0	1	1	1	0	0	1	1
8	0	0	1	1	0	1	0	0
9	1	0	0	0	0	1	1	1
10	1	0	1	1	1	1	0	1
11	1	1	0	1	0	0	1	0

(b)

$$\mathbf{MI} = \begin{bmatrix} 0.0 & 28.6 & 26.30 & 1.3 & 12.5 & 1.3 & 0.6 & 9.0 \\ 28.6 & 0.0 & 0.6 & 9.0 & 0.6 & 9.0 & 26.0 & 1.3 \\ 26.0 & 0.6 & 0.0 & 0.6 & 4.1 & 0.6 & 47.5 & 0.6 \\ 1.3 & 9.0 & 0.6 & 0.0 & 0.6 & 28.6 & 0.6 & 1.3 \\ 12.5 & 0.6 & 4.1 & 0.6 & 0.0 & 12.5 & 4.1 & 12.5 \\ 1.3 & 9.0 & 0.6 & 28.6 & 12.5 & 0.0 & 0.6 & 9.0 \\ 0.6 & 26.0 & 47.5 & 0.6 & 4.1 & 0.6 & 0.0 & 0.6 \\ 9.0 & 1.3 & 0.6 & 1.3 & 12.5 & 9.0 & 0.6 & 0.0 \end{bmatrix}$$

Figure 1. (a) Selected population. (b) **MI**. 10^3 for AffEDA.

elements of the diagonal of **MI** are zero because only the degree of dependence between two different variables is considered to form the clusters. The eight variables are divided into three groups: $\{Q_2, Q_3, Q_7\}$, $\{Q_1, Q_5, Q_8\}$, and $\{Q_6, Q_4\}$.

In this example the structural model learning provides $p_{\mathbf{Q}}(\mathbf{q})$, which is formulated as follows:

$$p(\mathbf{q}) = p(q_2, q_3, q_7)p(q_1, q_5, q_8)p(q_6, q_4) \tag{3}$$

The model parameters learning is based on the frequency tables presented in Figure 2. For example, the combination of values $Q_2 = 1$, $Q_3 = 1$ and $Q_7 = 0$ for the first cluster is found twice (individuals 1 and 2). Therefore, the occurrence probability of that combination is 2/11 = 0.18. For each isolated group, an inspection of the values of its variables in the population is performed to estimate its accumulated probability vector, from which the new population is sampled.

4.2. Estimation of Bayesian Network Algorithm. The initial population is generated assuming variables independence and uniform marginal distributions. At each generation, a set of individuals selected by tournament is used to learn the probabilistic model structure.

In this work, algorithm B[26] is implemented for building the model. The procedure starts with a network with no edges. A greedy algorithm is then used to add edges to the network. The edge that gives the most improvement according to the Bayesian information criterion (BIC), proposed by Schwarz,[27] is added at each iteration. The algorithm finishes when an arc addition no provides a metric increment. This model represents the conditional dependencies among the variables, and its parameters are estimated using the maximum likelihood principle.

New candidate solutions are generated by sampling the probability distribution encoded by the built network using probabilistic logic sampling.[28] Elitism is applied as in the case of AffEDA.

Figure 3 shows the model structure learnt by EBNA for the selected population depicted in Figure 1. Consequently, $p_{\mathbf{Q}}(\mathbf{q})$ is defined as follows:

$$p_{\mathbf{Q}}(\mathbf{q}) = p(q_1)p(q_2/q_1)p(q_3/q_1)p(q_4/q_2, q_3)p(q_5/q_3) \\ p(q_6/q_5)p(q_7/q_6)p(q_8/q_6) \tag{4}$$

It should be noted that the significant increment in the number of factors of $p_{\mathbf{Q}}(\mathbf{q})$ with respect to MPM models originates an important increase in the computational time required to estimate model parameters, as is shown in Figure 4 for the $p(q_2/q_1)$factor.

The parent set of the *i*-th node is formed by the variables from which an arrow comes out to the node. Those sets are denoted as $\boldsymbol{Pa}_i$. It can be seen from Figure 3 that parent sets are variable Q_1, the set composed of Q_2 and Q_3, variable Q_5, and variable Q_6.

Group 1

# indiv	Q_2	Q_3	Q_7
1	1	1	0
2	1	1	0
3	0	1	0
4	1	0	1
5	0	1	0
6	0	0	0
7	1	1	1
8	0	1	0
9	0	0	1
10	0	1	0
11	1	0	1

values			count	Prob
0	0	0	1	0,091
0	0	1	1	0,091
0	1	0	4	0,364
0	1	1	0	0,000
1	0	0	0	0,000
1	0	1	2	0,182
1	1	0	2	0,182
1	1	1	1	0,091

Group 2

# indiv	Q_1	Q_5	Q_8
1	0	0	1
2	0	1	0
3	1	0	0
4	0	1	0
5	0	1	0
6	1	0	1
7	0	0	1
8	0	0	0
9	1	0	1
10	1	1	1
11	1	0	0

values			count	Prob
0	0	0	1	0,091
0	0	1	2	0,182
0	1	0	3	0,273
0	1	1	0	0,000
1	0	0	2	0,182
1	0	1	2	0,182
1	1	0	0	0,000
1	1	1	1	0,091

Group 3

# indiv	Q_4	Q_6
1	0	0
2	0	1
3	1	0
4	0	1
5	0	1
6	1	0
7	0	0
8	0	0
9	1	0
10	1	1
11	1	0

values		count	Prob
0	0	3	0,273
0	1	3	0,273
1	0	4	0,364
1	1	1	0,091

Figure 2. Example of parameter estimation for AffEDA.

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

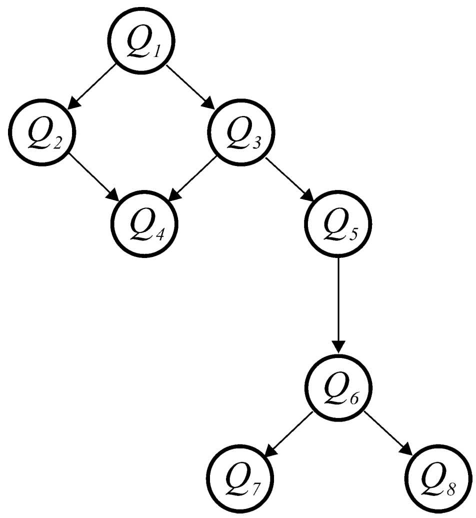


Figure 3. Bayesian network for the example population.

# indiv	Q_1	Q_2	Conditional probability
1	0	1	
2	0	1	$p(Q_2=0/Q_1=0)=\frac{p(Q_2=0,Q_1=0)}{p(Q_1=0)}=\frac{2/11}{6/11}=0.33$
3	1	0	
4	0	1	$p(Q_2=0/Q_1=1)=\frac{p(Q_2=0,Q_1=1)}{p(Q_1=1)}=\frac{4/11}{5/11}=0.80$
5	0	0	
6	1	0	
7	0	1	$p(Q_2=1/Q_1=0)=\frac{p(Q_2=1,Q_1=0)}{p(Q_1=0)}=\frac{4/11}{6/11}=0.67$
8	0	0	
9	1	0	
10	1	0	$p(Q_2=1/Q_1=1)=\frac{p(Q_2=1,Q_1=1)}{p(Q_1=1)}=\frac{1/11}{5/11}=0.20$
11	1	1	

Figure 4. Parameter estimation and conditional probabilities evaluation.

The factorization of $p_{\mathbf{Q}}(\mathbf{q})$ can be expressed, in a general form, as follows:

$$p(\mathbf{q}) = \prod_{i=1}^{n} p(q_i/\boldsymbol{pa}_i) \tag{5}$$

where $\boldsymbol{p}_{ai}$ represents an assumed value for the variable $\boldsymbol{Pa}_i$ taken from its set of possible instances.

The estimation of the conditional probabilities, i.e., the parameters, consists in calculating a finite set of parameters $\boldsymbol{\theta}_i$ (i = 1,⋯, n) whose cardinality increases with the number of parents associated with the i-th node. As is shown in Figure 4, the set $\boldsymbol{\theta}_2$ associated with node Q_2 has four elements because Q_2 only depends on the two possible values of its parent node. However, for variable Q_4, whose parent set contains two variables, $\boldsymbol{\theta}_4$ is composed of 8 parameters.

The estimation of the probabilistic graphic model for BNs consists in defining the pair (G, ($\boldsymbol{\theta}_1$,⋯,$\boldsymbol{\theta}_n$)), where G is an acyclic directed graph that represents the structural component of the model. The estimation of the model structure and parameters consume huge computational resources when the number of dependencies of each node increases.

5. RESULTS AND ANALYSIS

In this section, the application results of AffEDa and EBNA for solving diverse SNDPs are presented and compared with those provided by the strategy pPBIL-SOTS.[5] An outline of this memetic solution procedure and its pseudocode are provided in the Supporting Information. The application examples comprise one small-scale, two middle-scale, and two large-scale processes.[3] Regarding the small and middle-scale designs, reference values of the global optima are at hand that allows a more-precise evaluation of the metaheuristics performance. These values are the global optima obtained for case studies 1.a and 1.b (see section 3) and the ones reported in the literature for case studies 2 and 3, which were computed using other methodology.[3]

5.1. Experimental Setup. A total of five process flowsheets (case studies 1−5) are considered. The first one corresponds to the example used in section 3, for which the global optima of two designs of different complexity are available. Case study 2 is a continuous stirred tank reactor (CSTR)[29] whose operation is represented by a nonlinear model that is composed of 13 variables (total flow rates, compositions, and temperatures) and 5 mass and energy balances. The mineral flotation problem,[30] MFP, is selected as case study 3. Its model is bilinear and composed of 24 variables (8 flow rates and 16 compositions) related by total and component mass balances. For case studies 2 and 3, the model is linearized around the nominal operation point of the process. Finally, case studies 4 and 5 correspond to large-scale process flowsheets, and variables are related by total mass balances. (case study 4, 19 units and 52 streams; case study 5, 47 units and 82 streams). Interested readers can gain access to the files containing information about the case studies from https://www.ing.unrc.edu.ar/archivos/sndp_cases.doc.

It is assumed that there are no restrictions for the location of sensors to measure any variable; consequently, the search spaces for case studies 1−5 are made up of 2^{28}, 2^{13}, 2^{24}, 2^{52}, and 2^{82} solutions, respectively. The standard deviation of flowmeters is 2.5%, 1%, 2%, 2%, and 2% of the corresponding true flow rates for case studies 1−5, respectively. Data about the precision of temperature and composition observations is extracted from the literature.[3] Information about sensor costs is not provided for the sake of space.

Table 3 shows the complexity of the set of constraints imposed on case studies 2−5, and the parameters of the solution procedures are presented in Tables 4 and 5. In general the selected values are the commonly used in the literature for the metaheuristics involved in this study. In particular, the population size has been analyzed in detail because it originates

Table 3. Set of Constraints

case study	constraints
2	$E_{le} \geq 1$ for streams 3, 4, and 10
	$\sigma_3^* = 2.227 \times 10^{-3}$. $\sigma_4^* = 5.700$, $\sigma_{10}^* = 3.800 \times 10^{-1}$
3a	$E_{le} \geq 1$ for streams 1, 7, 9, and 22
	$\sigma_1^* = 1.500$, $\sigma_7^* = 0.142$, $\sigma_9^* = 2.850 \times 10^{-4}$ $\sigma_{22}^* = 1.045 \times 10^{-2}$
3b	$E_{le} \geq 1$ for streams 1, 4, 6, 7, 9, 10, 15, 16, 19, 20, 21, and 22
	$\sigma_1^* = 1.500$, $\sigma_4^* = 1.267$, $\sigma_6^* = 0.126$, $\sigma_7^* = 0.142$, $\sigma_9^* = 2.850 \times 10^{-4}$, $\sigma_{10}^* = 6.840 \times 10^{-3}$, $\sigma_{15}^* = 2.000 \times 10^{-5}$, $\sigma_{16}^* = 8.200 \times 10^{-5}$, $\sigma_{19}^* = 4.232 \times 10^{-3}$, $\sigma_{20}^* = 9.900 \times 10^{-4}$, $\sigma_{21}^* = 1.020 \times 10^{-4}$. $\sigma_{22}^* = 1.045 \times 10^{-2}$
4	$E_{le} \geq 1$ for streams 2, 5, 15, 29, 31, 32, 38, 39, 40, 44, 45, 46, 47, 48, 49, 50, 51, and 52
	$\sigma_{15}^* = 12\,410$, $\sigma_{31}^* = 1370$, $\sigma_{32}^* = 130$, $\sigma_{40}^* = 1380$, $\sigma_{44}^* = 570$, $\sigma_{45}^* = 590$, $\sigma_{46}^* = 720$, $\sigma_{47}^* = 550$. $\sigma_{49}^* = 1440$
5	$E_{le} \geq 1$ for streams 5, 10, 12, 14, 17, 35, 37, 39, 44, 56, 62, 69, 70, and 77
	$\sigma_{10}^* = 1584$, $\sigma_{17}^* = 1359$, $\sigma_{35}^* = 200$, $\sigma_{39}^* = 1580$, $\sigma_{56}^* = 123$, $\sigma_{69}^* = 1284$

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

Table 4. Parameter Settings for EDAs

parameter	pPBIL-SOTS	AffEDA	EBNA
NPBIL	4	–	–
M	12	50–100–300	50–100–300
#MaxGeneration	200	200	200
LR	0.1	–	–
PMUTA	0.02	–	–
MS	0.05	–	–
$P_{\text{interaction}}$	0.25	–	–
t	–	50%	50%
k	1	10	10

Table 5. Parameter Settings for SOTS

parameter	value
P_{so}	0.05
N_{so}	25
N_c	2
#maxiter	120
P_t	$1.5\sqrt{I}$
P_h	$I/2$

differences in the performance for AffEDA and EBNA, as can be seen below. Regarding the pPBIL-SOTS parameters, the population size and the frequency of calls to the local search are determined empirically. It was verified that the solutions and the execution times have negligible variations when parameters values are around the select ones.

All procedures are executed using a Processor Intel Core i7 CPU 920 at 2.67 GHz, 6GB RAM using MatLab Release 14. The parallel implementation of PBIL-SOTS is simulated by sequentially running NPBIL instances of that procedure and updating their probability vectors using uniform cross-over. Therefore, computation times are only reported for illustrative and comparative purposes.

In this work, three main metaheuristics based on EDAs are considered. For the SNDP defined by eq 1, the fitness function, F, used by the stochastic methodologies to evaluate a solutions $\mathbf{q}$, is formulated as follows:

$$F = \begin{cases} \sum_{i=1}^{n} c_i q_i & \text{if } \mathbf{q} \text{ is feasible} \\ \sum_{i=1}^{n} c_i(1 + S(\mathbf{q})) & \text{if } \mathbf{q} \text{ is infeasible} \end{cases} \tag{6}$$

where $\sum_{i=1}^{n} c_i$is the cost of the SN when all variables are measured (upper bound of the SNDP objective function), and $S(\mathbf{q})$ takes into account constraint violations as follows:

$$S(\mathbf{q}) = \frac{\text{rno}}{\#S_E} + \frac{1}{\text{rnp}} \sum_{i}^{\text{rnp}} \frac{\hat{\sigma}_i - \sigma_i^*}{\hat{\sigma}_i} \tag{7}$$

where rno and rnp are the number of unsatisfied observability and precision constraints, respectively, for a solution vector $\mathbf{q}$.

The stochastic methodologies perform the same number of calls (evaluation) to fitness or evaluation function if the algorithms are executed using the same values for the population size and the number of generations. However, it is interesting to compare their performances when different tools are added to the basic algorithms. Regarding pPBIL, a local search procedure is subordinated to the main metaheuristics. This necessarily increases the number of calls to the fitness function. With respect to multivariate EDAs, the additional tool is the capability to estimate the structure of the underlying probabilistic model. This task requires solving a clustering optimization problem for the AffEDA. Regarding EBNA, it is necessary to optimally estimate the BN that best represents the samples (that is, a NP complete problem). Therefore, multivariate algorithms require solving additional optimization problems, which also consume resources. Instead of calls to the fitness function, they perform calls to the objective function of the aforementioned optimization subproblems.

Even though the number of evaluations of the fitness function is a commonly used stopping criterion, in this work, the number of generations is used with that purpose, taking into account the previous discussion. In this way, algorithms evolve using all of the resources they need, which are of different nature, until the stopping criterion is achieved. After a certain number of trials, solutions are evaluated taking into account industrial criteria: solution quality (instrumentation cost) and computational time.

For problems with unknown global optima, the minimum objective function values achieved by an stochastic algorithm, which correspond to the best solutions it can attain, are important performance measures, and they are analyzed in detail in this work.

Regarding the computational load, in general the computational time used to solve an instance of the SNDP can be divided into four parts:

(1) the total time consumed in evaluating the objective function, which depends on the time required to

Table 6. Best Solutions

case	best solution	standard deviation of the estimates	cost
1.a	1, 4, 6, 7, 9, 10, 11, 14, and 16–24	$\hat{\sigma}_4 = 2.186$, $\hat{\sigma}_8 = 2.564$, $\hat{\sigma}_{17} = 0.804 =$, $\hat{\sigma}_{21} = 1.502$, $\hat{\sigma}_{23} = 0.065$, $\hat{\sigma}_{25} = 1.486$	752.26
1.b	1–15, 18–20, 22, 24, 25, and 27	$\hat{\sigma}_4 = 2.061$, $\hat{\sigma}_5 = 0.888$, $\hat{\sigma}_7 = 0.058$, $\hat{\sigma}_8 = 1.499$, $\hat{\sigma}_{12} = 0.959$, $\hat{\sigma}_{16} = 0.585$, $\hat{\sigma}_{18} = 0.405$, $\hat{\sigma}_{20} = 0.261$, $\hat{\sigma}_{27} = 1.200$, $\hat{\sigma}_{28} = 1.444$	1106.50
2	2, 3, 9, and 12	$\hat{\sigma}_3 = 2.193 \times 10^{-3}$, $\hat{\sigma}_4 = 0.339$, $\hat{\sigma}_{10} = 0.369$	735
3a	1, 3, 5–9, 11, 17, and 22	$\hat{\sigma}_1 = 1.272$, $\hat{\sigma}_7 = 0.142$, $\hat{\sigma}_9 = 2.833 \times 10^{-4}$ $\hat{\sigma}_{22} = 1.045 \times 10^{-2}$	1448
3b	1, 3, 5–11, 15–17, and 20–22	$\hat{\sigma}_1 = 1.269$, $\hat{\sigma}_4 = 1.261$, $\hat{\sigma}_6 = 0.103$, $\hat{\sigma}_7 = 0.141$, $\hat{\sigma}_9 = 2.832 \times 10^{-4}$, $\hat{\sigma}_{10} = 1.188 \times 10^{-3}\hat{\sigma}_{15} = 2.000 \times 10^{-5}$, $\hat{\sigma}_{16} = 8.199 \times 10^{-5}$, $\hat{\sigma}_{19} = 3.347 \times 10^{-3}$, $\hat{\sigma}_{20} = 9.899 \times 10^{-4}\hat{\sigma}_{21} = 1.020 \times 10^{-4}\hat{\sigma}_{22} = 1.042 \times 10^{-2}$	2928
4	10, 16, 31–33, 35, 37, 39–41, 43–48, and 50–52	$\hat{\sigma}_2 = 2549$, $\hat{\sigma}_5 = 2549$, $\hat{\sigma}_{15} = 2495$, $\hat{\sigma}_{29} = 2455$, $\hat{\sigma}_{31} = 916$, $\hat{\sigma}_{32} = 84$ $\hat{\sigma}_{38} = 2058$, $\hat{\sigma}_{39} = 92$, $\hat{\sigma}_{40} = 919$ $\hat{\sigma}_{44} = 379$, $\hat{\sigma}_{45} = 40$, $\hat{\sigma}_{46} = 478$, $\hat{\sigma}_{47} = 364$, $\hat{\sigma}_{48} = 36$, $\hat{\sigma}_{49} = 566$, $\hat{\sigma}_{50} = 30$ $\hat{\sigma}_{51} = 40$, $\hat{\sigma}_{52} = 25$	1154.34
5	1, 2, 5, 10, 12, 15, 21, 30, 33–37, 44, 50, 55, 56, 60, 62, 63, 66–68, 74–78, and 82	$\hat{\sigma}_5 = 386$, $\hat{\sigma}_{10} = 1531$, $\hat{\sigma}_{12} = 1531$, $\hat{\sigma}_{14} = 985$ $\hat{\sigma}_{17} = 985$ $\hat{\sigma}_{35} = 198$, $\hat{\sigma}_{37} = 210$, $\hat{\sigma}_{39} = 1166$ $\hat{\sigma}_{44} = 8$,$\hat{\sigma}_{56} = 108$,$\hat{\sigma}_{62} = 24$, $\hat{\sigma}_{69} = 988\hat{\sigma}_{70} = 988$, $\hat{\sigma}_{10} = 18$	50845.16

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149–12164

Table 7. Statistics of the Best Solutions (Case Study 1.a)

strategy	mean	SD	CV	min	median	max	P99
AffEDA M = 50	763.75	19.13	0.03	752.26	754.36	864.06	850.76
EBNA, M = 50	1511.30	343.25	0.23	754.76	1801.60	1835.87	1832.50
pPBIL-SOTS,	752.26	0.00	0.00	752.26	752.26	752.26	752.26

perform one evaluation and the number of calls to the fitness function (this quantity is related to the population size and the number of solutions explored and evaluated by the local search if the algorithm performs this step);

(2) the time required for building the probabilistic model, which involves the calculation of the marginal probabilities and the structural learning;

(3) the time used to generate the neighboring structure of the local search; and

(4) other times of lower incidence.

For pPBIL-SOTS, the first and third times are dominant, and the second one is negligible. In contrast, this is relevant for AffEDA and EBNA, and the time they use for the local search is zero.

5.2. Analysis of Results. At first 100 runs of AffEDA and EBNA are performed for $M = 50$ individuals. To compare results using similar population sizes, pPBIL-SOTS is run setting $M = 12$ and NPBIL = 4; therefore, 48 individuals are considered at each generation of that algorithm. Table 6 reports the best attained solutions. These are expressed in terms of the set of measured variables, the standard deviations of the estimates calculated by the data reconciliation procedure for all the variables contained in S_E and S_σ, and the total instrumentation cost. That table shows that all of the constraints on the ability to be estimated are satisfied because estimates can be calculated for all of the variables included in S_E; that is, they are measured or unmeasured but observable. From Table 6, it can also be observed that the standard deviations of the estimates for the variables included in S_σ are equal or lower than their upper bounds.

For cases 1.a and 1.b, the global optima were calculated using an exact procedure (see section 3). This allows us to perform an accurate analysis regarding the quality of the solutions obtained running the proposed heuristics. Regarding case study 1.a, Table 7 shows the statistics [mean, standard deviation (SD), coefficient of variation (CV), minimum, median, maximum, and 99% quartile (P99)] for the best solutions obtained for 100 runs of each algorithm. The evaluation function values of the best solutions are sorted in ascending order. Furthermore, a new integer variable called the index is defined in the range of 0 to 100, such that index = 1 for the lowest value of the evaluation function and ondex = 100 for the highest one. That is, the index represents the position of objective function value obtained for each run in the sorted vector. To show the variability of the best solutions, the sorted values of F are displayed in Figure 5. For case study 1.b, the same information is provided in Table 8 and Figure 6. Regarding the quality of the solution, it can be seen that pPBIL-SOTS out-performs the other algorithms, given that it obtains the global minimum for all the runs. In contrast, EBNA shows the poorest behavior. It can not find the reference minimum for case study 1.b, and all statistics are greater than those provided by pPBIL-SOTS and AffEDA. With respect to AffEDA, it efficiently solves case study 1.a, but its performance decreases for case study 1.b, which is a more-challenging

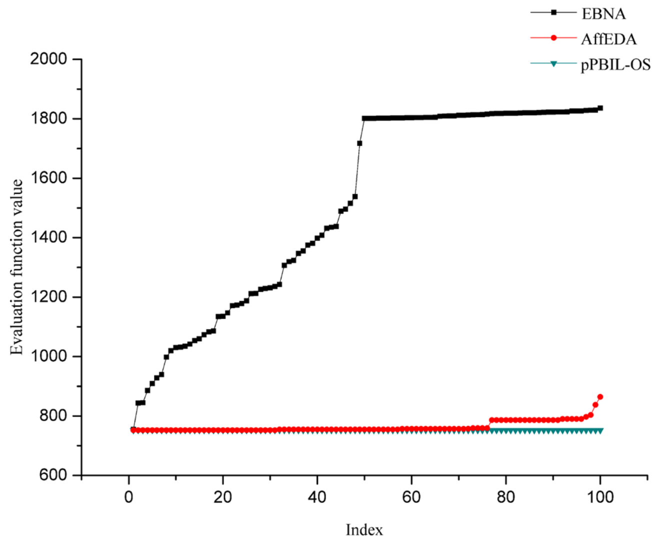


Figure 5. Evaluation-function values of the best solutions (case study 1.a).

optimization problem. It achieves the best solution of this problem only for 10% of the runs.

The statistics of the execution times for case studies 1.a and 1.b are included in Tables 9 and 10, respectively. It can be seen that the computation requirements for pPBIL-SOTS are the highest, and AffEDA out-performs EBNA regarding that performance measure. Runs execution times of AffEDA are similar; and the same behavior is observed for EBNA. In contrast, pPBIL-SOTS shows variability in the computation time required by different runs. The fact that pPBIL-SOTS randomly performs additional calls to the fitness function explains that variability.

For case study 2 (CSTR), the key variables are total flow rates, compositions, and temperatures. The statistics of the best solutions and the execution times are reported in Tables 11 and 12, respectively. Furthermore, the evaluation function values are displayed in Figure 7. It can be seen that both AffEDA and pPBIL-SOTS attain the same value ($735.00) for each run. This optimal value has been previously obtained using different exact procedures.[3] In contrast, the behavior of EBNA is poor. The mean and the maximum of the evaluation function values for 100 runs are 746.47 and 817, respectively. Furthermore, execution times show the behavior observed for case studies 1.a and 1.b. The computational time of AffEDA is the lowest and pPBIL-SOTS requires the highest computational resources.

For the MFP, two sets of design constraints are analyzed that involve an increasing number of key variables (flow rates and compositions). Tables 13 and 14 reports the statistics of the evaluation function values for case studies 3a and 3b, respectively. Furthermore, Figures 8 and 9 display these values for each run. Regarding case study 3a, both pPBIL-SOTS and AffEDA attain the minimum cost ($1448.00) for the 100% of the executions. This value is the same reported in the literature

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

Table 8. Statistics of the Best Solutions (Case Study 1.b)

strategy	mean	SD	CV	min	median	max	P99
AffEDA M = 50	1262.81	148.83	0.12	1106.50	1224.11	1798.95	1798.85
EBNA, M = 50	1722.37	157.70	0.09	1288.86	1805.88	1824.96	1821.19
pPBIL-SOTS,	1106.50	0.00	0.00	1106.50	1106.50	1106.50	1106.50

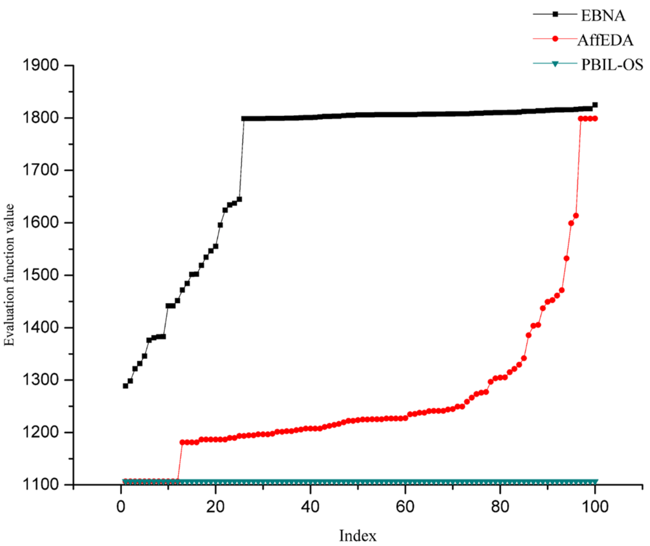


Figure 6. Evaluation-function values of the best solutions (case study 1.b).

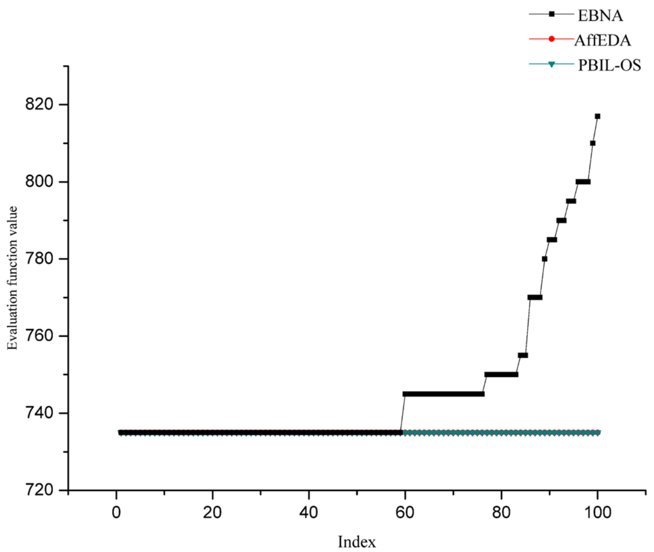


Figure 7. Evaluation-function values of the best solutions (case study 2).

Table 9. Statistics of the Execution Times (s) (Case Study 1.a)

strategy	mean	SD	CV	min	median	max
AffEDA	35.02	1.29	0.04	32.74	36.67	39.66
EBNA	113.72	2.67	0.02	107.36	113.67	119.87
pPBIL-SOTS,	185.38	17.53	0.09	142.23	181.95	220.10

Table 10. Statistics of the Execution Times (s) (Case Study 1.b)

strategy	mean	SD	CV	min	median	max
AffEDA M = 50	34.54	0.98	0.03	32.34	34.42	37.54
EBNA M = 50	113.15	2.50	0.02	108.50	112.67	119.93
pPBIL-SOTS,	197.00	14.70	0.10	15.73	198.70	223.30

Table 11. Statistics of the Best Solutions (Case Study 2)

strategy	mean	SD	CV	min	median	max	P99
AffEDA M = 50	735.00	0.00	0.00	735	735	735	735
EBNA, M = 50	746.47	20.07	0.03	735	735	817	813.5
pPBIL-SOTS,	735	0.00	0.00	735	735	735	735

Table 12. Statistics of the Execution Times (s) (Case Study 2)

strategy	mean	SD	CV	min	median	max
AffEDA M = 50	17.09	17.09	0.38	16.42	17.07	18.35
EBNA M = 50	36.63	0.71	0.02	35.75	36.43	38.71
pPBIL-SOTS	117.97	46.99	0.39	23.4	106.2	286.2

for this SNDP.[3] The procedure EBNA only achieves the minimum cost for some runs, and 50% of the evaluation function values for the best solutions are greater than \$2113.00.

Remarkable differences in the performance of the methodologies are observed for case study 3b, which involves a complex set of design constraints. From Table 14, it can be observed that the behavior of EBNA is rather poor. It attains an evaluation function value equal to \$2978 for the best solution, while the minimum value achieved by pPBIL-SOTS is \$2928. The reproducibility of EBNA is low; in fact, the median of the objective function values for 100 runs is equal to the cost of installing sensors to measure all the variables. In contrast, AffEDA behaves better than EBNA. It can be seen that the aforementioned median lowers to \$3300.50, the procedure attains the reference solution (\$2928) but the CV (0.22) increases with respect to those achieved for the previous cases. Regarding the procedure pPBIL-SOTS, it obtains the best solution for almost all the runs, the CV is equal to 0.002, and moreover, it achieves a reference minimum that is lower than the value reported in the literature for this SNDP.[3]

Tables 15 and 16 report the statistics of the execution time for case studies 3a and 3b, respectively. The computational time of AffEDA is the lowest, and EBNA shows the opposite behavior.

With respect to case study 4, Tables 17 and 18 show the statistics of the best solutions and the execution times for 100 runs of each algorithm. In Figure10, the evaluation function values of the best solutions are displayed in ascending order. Figure 10 clearly indicates that when the size of the optimization problem increases the quality of the solutions obtained using pPBIL-SOTS is the best. Even though AffEDA attains the reference minimum cost for case studies 1, 2, and 3a, it fails to obtain that value for case studies 3b and 4. The

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

Table 13. Statistics of the Best Solutions (Case Study 3a)

strategy	mjean	SD	CV	min	median	max	P99
AffEDA M = 50	1448.00	0.00	0.00	1448.00	1448.00	1448.00	1448.00
EBNA, M = 50	2265.12	745.77	0.32	1448.00	2113.00	5213.00	5213.00
pPBIL-SOTS,	1448.00	0.00	0.00	1448.00	1448.00	1448.00	1448.00

Table 14. Statistics of the Best Solutions (Case Study 3b)

strategy	mean	SD	CV	min	median	max	P99
AffEDA M = 50	3608.77	801.54	0.22	2928.00	3300.50	5213.00	5213.00
EBNA, M = 50	5017.21	570.27	0.11	2978.00	5213.00	5213.00	5213.00
pPBIL-SOTS,	2929.20	6.85	0.002	2928.00	2928.00	2968.00	2968.00

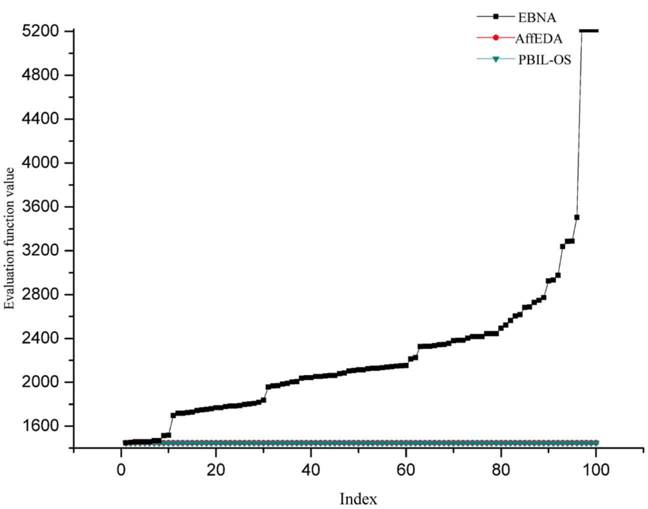


Figure 8. Evaluation-function values of the best solutions (case study 3a).

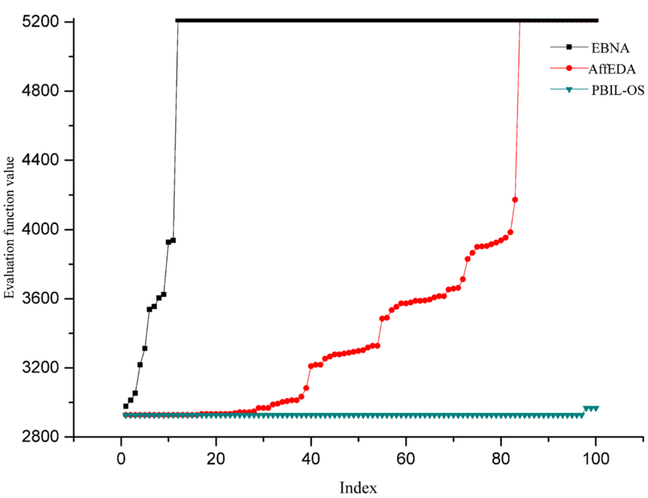


Figure 9. Evaluation-function values of the best solutions (case study 3b).

Table 15. Statistics of the Execution Times (s) (Case Study 3a)

strategy	mean	SD	CV	min	median	max
AffEDA	17.09	0.38	0.02	16.42	17.08	18.35
EBNA	109.02	2.40	0.02	105.54	108.48	119.31
pPBIL-SOTS	85.86	5.85	0.07	72.23	85.11	104.06

Table 16. Statistics of the Execution Times (s) (Case Study 3b)

strategy	mean	SD	CV	min	median	max
AffEDA M = 50	32.38	2.77	0.08	28.1	32.31	39.54
EBNA M = 50	102.98	4.05	0.04	95.31	102.50	118.81
pPBIL-SOTS,	87.88	5.73	0.06	72.74	87.57	102.01

results of EBNA are the poorest. Regarding the computational time, pPBIL-SOTS presents the highest time requirements, but it should be noticed that the reported elapsed times correspond to a simulated parallel execution, in which four PBIL-SOTS procedures are run sequentially.

Regarding case study 5, Tables 19 and 20 include the statistics of the best solutions and the execution times for 100 runs of each algorithm. Furthermore, Figure 11 shows the evaluation function values of the best solutions in ascending order.

In relation to the quality of the solutions, the minimum SN cost obtained using EBNA is almost twice greater than the reference minimum, and the CV (36.6%) is 1 and 2 orders of magnitude greater than those provided by AffEDA and pPBIL-SOTS, respectively. In comparison to EBNA, the repeatability of the solution of AffEDA is better (CV = 6.8%), but the minimum SN cost obtained using AffEDA exceeds in 70% the one achieved using pPBIL-SOTS. The best solution is only attained by the methodology pPBIL-SOTS. Consequently, its statistics are also the best ones. The remarkable capability of that procedure to replicate good solutions is revealed by its very low CV (0.8%). In Figure 11, the gap between the reference minimum and the best solutions obtained using AffEDA and pPBIL-SOTS should be noted. With respect to the computation time, Table 20 indicates that EBNA is the most-time-consuming algorithm for this case study, even though the number of individuals of the population is relatively small.

From Tables 9, 10, 12, 14, 16, 18, and 20, it can be observed that all of the algorithms require more computational time for case studies of incremental size. However, the influence of the size instance is notorious for EBNA. The time used by EBNA to model the interrelations among all the variables increases at the highest rate in comparison with the requirements of the other procedures.

The compared algorithms have structural differences. The pPBIL-SOTS procedure manages a small population, but the use of local search allows the exploration of a neighborhood of the solution constituted by similar genotypes, which may significantly differ in the evaluation function value. Because the local search provides an additional exploratory capability to pPBIL-SOTS, the population size is not a key parameter

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

Table 17. Statistics of the Best Solutions (Case Study 4)

strategy	mean	SD	CV	min	median	max	P99
AffEDA	1455.73	153.98	0.11	1197.38	1429.93	1974.36	1936.62
EBNA	3907.56	1336.34	0.34	2355.33	3777.65	12669.31	9736.83
pPBIL-SOTS,	1154.34	0	0.00	1154.34	1154.34	1154.34	1154.34

Table 18. Statistics of the Execution Times (min) (Case Study 4)

strategy	mean	SD	CV	min	median	max
AffEDA	1.42	0.02	0.01	1.38	1.42	1.50
EBNA	8.02	0.16	0.02	7.68	8.01	8.41
pPBIL-SOTS,	10.18	0.96	0.09	7.89	10.16	12.83

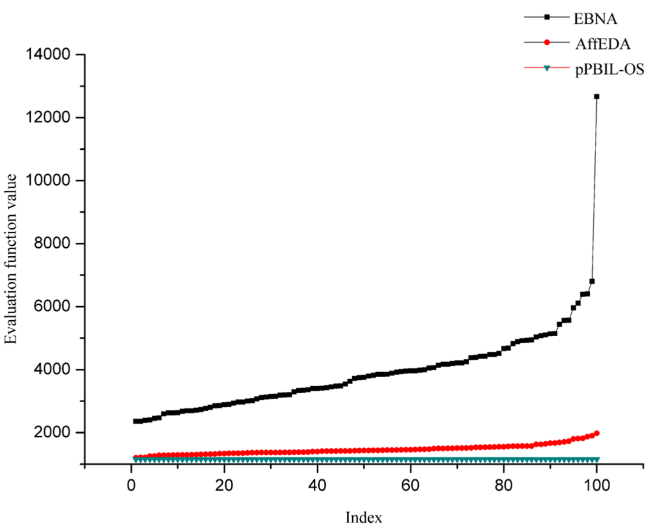


Figure 10. Evaluation-function values of the best solutions (case study 4).

regarding the quality of the solution. In contrast, AffEDA and EBNA do not use a local search procedure, and perform a complex structural learning to estimate the probabilistic model. It is expected that the increment of the population size enhances the quality of their solutions given that it allows the exploration of different regions of the search space.

Next the effect of increasing the population size on the performances of AffEDA and EBNA are analyzed for 100 runs of cases 1.a, 1.b, and 4. Tables 21 and 22 report the statistics of the best solutions and execution times for the aforementioned strategies and also include the same measures for pPBIL-SOTS. Regarding the most complex design, only the performance of AffEDA can be analyzed because EBNA's computational load is extremely huge.

For case study 1.a, Table 21 and Figure 12 show the statistics of the best solutions for 100 runs of AffEDA and EBNA with different values of M (M = 50, 100, and 300), and the results previously obtained using PBIL-SOTS (M = 12, NPBIL = 4). Furthermore, the elapsed time statistics are displayed in Table 22. Considering the solution quality, a

Table 20. Elapsed Time Statistics (min) (Case Study 5)

strategy	mean	SD	CV	min	median	max
AffEDA	5.90	0.04	0,01	5.82	5.90	6.07
EBNA	31.59	1.47	0.05	27.86	31.54	35.61
pPBIL-SOTS,	25.12	2.65	0,11	17.62	24.90	30.99

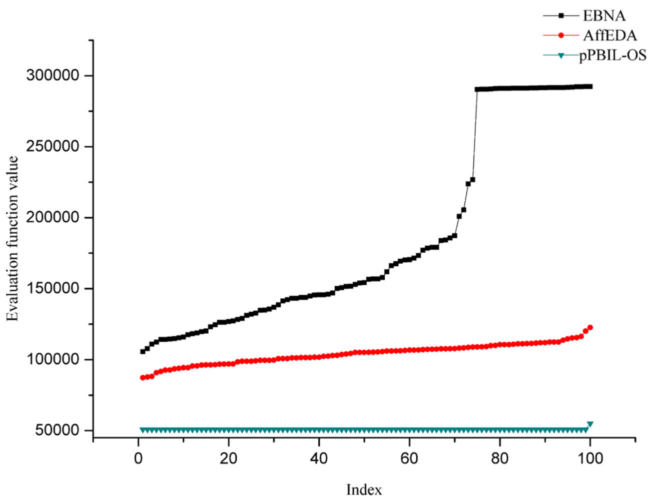


Figure 11. Evaluation-function values of the best solutions (case study 5).

significant increment of the population size enhances AffEDA performance more than EBNA behavior.

For case study 1.b, Tables 23 and 24 and Figure 13 show the results of experiments analogous to those performed for the previous case. It can be noticed that an increment of the number of individuals in the population from 50 to 300 allows us to reduce the difference between the solution mean and its lower bound from 1.53% to 0.33% for AffEDA and from more than 100% to 11% for EBNA.

Regarding case study 4, results are presented in Tables 25 and 26 and Figure 14.

Based on previous results, it is sensible to make comparisons only between AffEDA with increasing population sizes and pPBIL-SOTS. Thus, the performances of both algorithms are analyzed for the large-size example (that is, case study 5). Table 27 shows the statistics of the best solutions for 100 runs of AffEDA with different values of M (M = 50, 100, and 300) and the results previously obtained using pPBIL-SOTS (M = 12, NPBIL = 4). It can be seen that pPBIL-SOTS outperforms AffEDA even though its population size has been increased to 300 individuals. The minimum value of the best solutions achieved using pPBIL-SOTS is 50 845.16, and the median and

Table 19. Statistics of the Best Solutions (Case Study 5)

strategy	mean	SD	CV	min	median	max	P99
AffEDA	104 055.68	7142.19	0.07	87 282.49	105 128.89	122 632.53	121 365.27
EBNA	184 736.07	67 670.99	0.37	105 643.06	105 128.89	292 409.45	292 404.24
pPBIL-SOTS	50 886.63	412.88	0.01	50 845.16	50 845.37	54 974.18	52 909.78

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

Table 21. Best-Solution Statistics for Different Population Sizes (Case Study 1.a)

	mean	SD	CV	min	median	max	P99
AffEDA M = 50	763.75	19.13	0.03	752.26	754.36	864.06	850.76
AffEDA M = 100	758.32	12.35	0.02	752.26	752.26	803.46	786.26
AffEDA M = 300	754.74	829.00	0.01	752.26	752.26	786.26	786.26
EBNA, M = 50	1511.30	343.26	0.23	754.76	1801.60	1835.87	1832.50
EBNA, M = 100	1207.56	346.33	0.29	757.26	1080.11	1813.00	1812.02
EBNA, M = 300	836.39	58.11	0.07	752.26	840	995.36	982.36
pPBIL-SOTS,	752.26	0.00	0.00	752.26	752.26	752.26	752.26

Table 22. Elapsed Time Statistics (Case Study 1.a)

	mean	SD	CV	min	median	max
AffEDA M = 50	35.02	1.29	0.04	32.74	36.67	39.66
AffEDA M = 100	55.95	0.88	0.02	54.25	55.80	59.93
AffEDA M = 300	147.58	0.56	0.01	146.19	147.58	149.50
EBNA M = 50	113.72	2.67	0.02	107.36	113.67	119.87
EBNA, M = 100	143.07	4.65	0.03	131.19	143.19	154.27
EBNA M = 300	250.64	4.70	0.02	237.55	250.68	262.25
pPBIL-SOTS,	185.38	17.53	0.09	142.23	181.95	220.10

Table 24. Elapsed Time Statistics (s) (Case Study 1.b)

	mean	SD	CV	min	median	max
AffEDA M = 50	34.54	0.98	0.03	32.34	34.42	37.54
AffEDA M = 100	56.13	1.10	0.02	53.78	56.93	60.07
AffEDA M = 300	149.73	1.24	0.01	145.81	149.70	156.00
EBNA M = 50	113.15	2.50	0.02	108.50	112.67	119.93
EBNA, M = 100	140.93	5.19	0.04	132.88	139.80	155.09
EBNA M = 300	253.74	9.43	0.04	224.90	256.00	268.65
pPBIL-SOTS,	197.00	14.70	0.10	15.73	198.70	223.30

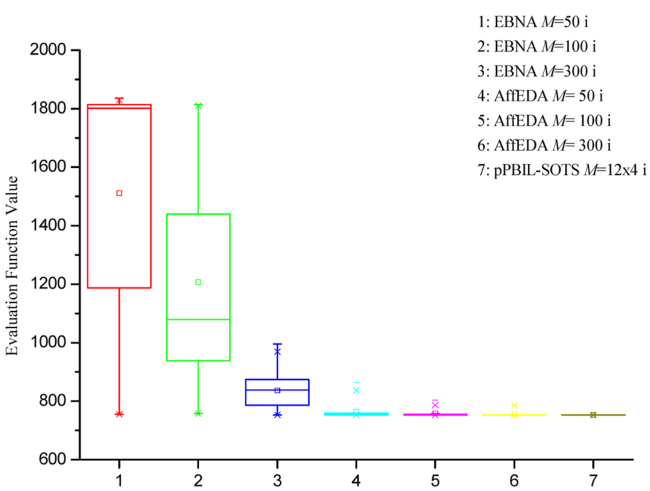


Figure 12. Comparison of evaluation-function values (case study 1.a).

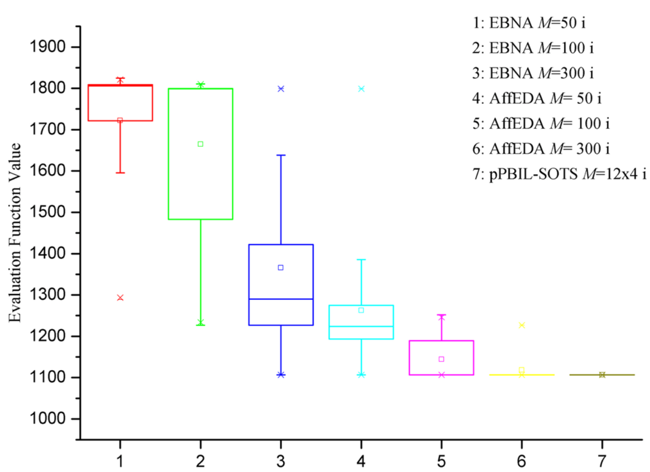


Figure 13. Comparison of evaluation-function values (case study 1.b).

the CV values are 50 845.37 and 0.8%, respectively. The 99% of the best solutions are lower than 52 909.78 for pPBIL-SOTS, but the same measure for AffEDA when M = 300 is 77 391.18. The evaluation function values of the best solutions are represented in Figure 15 in ascending order. Furthermore, the execution times are shown in Figure 16, and Table 28 provides a statistical analysis of these elapsed times. When the population size increases a proportional increment of the execution times is observed for AffEDA. Furthermore, the elapsed times of pPBIL-SOTS and AffEDA with M = 300 are similar, which is a sensible result when taking into account that the performance of AffEDA for that population size approximates the behavior of pPBIL-SOTS.

Table 23. Best-Solution Statistics for Different Population Sizes (Case Study 1.b)

	mean	SD	CV	min	median	max	P99
AffEDA M = 50	1262.81	148.83	0.11	1106.50	1224.11	1798.95	1798.85
AffEDA M = 100	1144.36	48.98	0.04	1106.50	1106.50	1251.86	1246.26
AffEDA M = 300	1118.66	33.75	0.03	1106.50	1106.50	1226.56	1226.56
EBNA, M = 50	1722.37	157.70	0.09	1288.86	1805.88	1824.96	1821.19
EBNA, M = 100	1664.97	195.68	0.11	1226.56	1798.74	1810.42	1809.95
EBNA, M = 300	1365.98	215.69	0.16	1106.50	1224.11	1798.95	1798.85
pPBIL-SOTS,	1106.50	0.00	0.00	1106.50	1106.50	1106.50	1106.50

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

Table 25. Best-Solution Statistics for Different Population Sizes (Case Study 4)

	mean	SD	CV	min	median	max	P99
AffEDA M = 50	1455.73	153.98	0.11	1197.38	1429.93	1974.36	1936.62
AffEDA M = 100	1215.35	63.31	0.05	1154.34	1207.66	1605.96	1522.52
AffEDA M = 300	1258.75	38.13	0.03	1154.34	1154.34	1532.20	1371.27
EBNA, M = 50	3907.56	1336.34	0.34	2355.33	3777.65	12 669.31	9736.83
EBNA, M = 100	2435.97	572.45	0.24	1361.81	2295.48	4400.73	4284.00
EBNA, M = 300	1484.34	267.80	0.18	1154.83	1367.19	2406.18	2279.04
pPBIL-SOTS,	1154.34	0.00	0.00	1154.34	1154.34	1154.34	1154.34

Table 26. Elapsed Time Statistics (min) (Case Study 4)

	mean	SD	CV	min	median	max
AffEDA M = 50	1.42	0.02	0.01	1.38	1.42	1.50
AffEDA M = 100	2.33	0.02	0.01	2.28	2.33	2.39
AffEDA M = 300	6.20	0.02	0.01	6.16	6.20	6.25
EBNA M = 50	8.02	0.16	0.02	7.67	8.01	8.41
EBNA, M = 100	9.55	0.20	0.02	9.07	9.55	10.13
EBNA M = 300	14.26	0.38	0.03	13.44	14.22	15.09
pPBIL-SOTS,	10.18	0.96	0.09	7.89	10.17	12.83

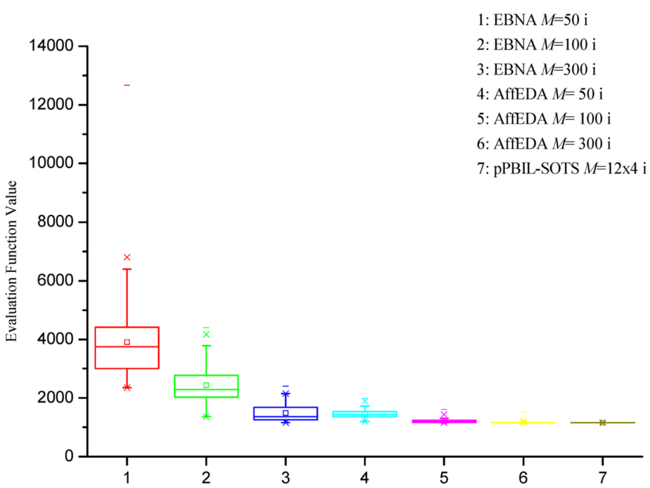


Figure 14. Comparison of evaluation function values (case study 4).

Table 27. Best-Solution Statistics for Different Population Sizes (Case Study 5)

	AffEDA M = 50	AffEDA M = 100	AffEDA M = 300	pPBIL-SOTS
mean	104 055.68	65 670.80	56 176.08	50 886.63
standard deviation	7142.19	7880.64	7981.59	412.88
coefficient of variation	0.069	0.12	0.142	0.008
minimum	87 282.49	53 061.31	50 846.18	50 845.16
median	105 128.89	64 656.74	51 657.44	50 845.37
maximum	122 632.53	81 189.39	80 547.8	54 974.18
P99	121 365.27	84 641.94	77 391.18	52 909.78

The best SNs attained using pPBIL-SOTS (Table 6) are also compared with the solutions of the minimum number sensor network design problem (MNSNDP). It determines the set of variables that should be measured with the objective of calculating all the unmeasured variables, as a function of the observations and the process model, at minimum cost.[16] For the MNSNDP, the percentage of unmeasured variables that are unobservable is 0%.

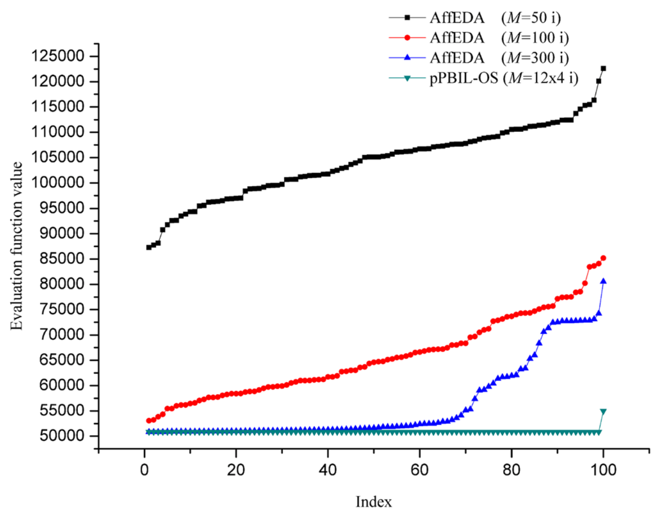


Figure 15. Evaluation-function values of the best solutions for AffEDA and pPBIL-SOTS.

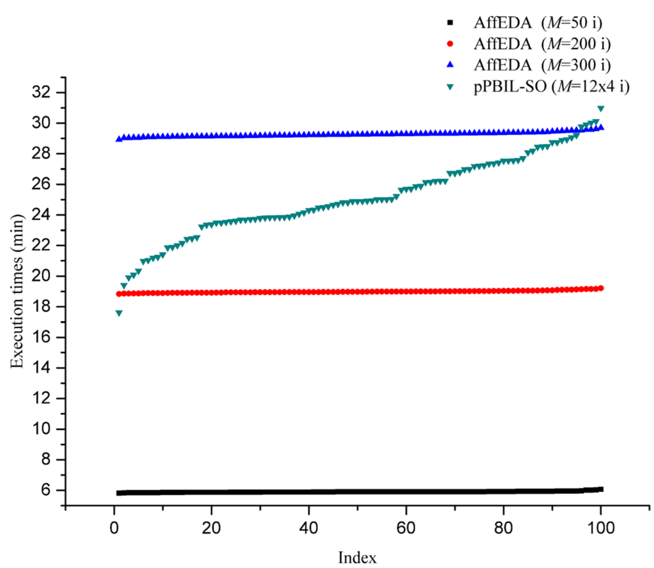


Figure 16. Execution times for AffEDA and pPBIL-SOTS.

Table 28. Elapsed Time Statistics (min) (Case Study 5)

	mean	SD	CV	min	median	max
AffEDA M = 50	5.90	0.04	0.007	5.82	5.90	6.07
AffEDA M = 100	10.42	0.03	0.003	10.35	10.41	10.53
AffEDA M = 300	29.26	0.14	0.005	28.92	29.26	29.69
pPBIL-SOTS,	25.13	2.66	0.106	17.62	24.89	30.99

Regarding case study 1, the solution of the MNSNDP is [1, 2, 7, 9−11, 16−24, 27, 28], and the cost of this SN is $555.46.

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

For case study 1.a, both the solution of the SNDP that satisfies precision and estimability constraints (eq 1) and the corresponding one to the MNSNDP are composed of the same number of sensors (17) but the optimal sets of instruments are different. Precision restrictions are imposed on the 14.3% of the variables. Even though the cost of the SN that fulfills constraints on a set of key variables increases ($752.26) in relation to the one obtained for the MNSNDP, some unmeasured variables are not estimable. The set of observable unmeasured variables is [5 8 12 15 25]; therefore, the 54% of the unmeasured variables remain unobservable. Regarding case study 1.b, the complexity of the constraints increases with respect to case study 1.a. Precision restrictions involve the 21.4% of the variables. The solution is composed of more sensors (23) than the one obtained for case study 1.a and its cost increases to $1106.50. For case study 1.b, all of the unmeasured variables are observable; that is, the percentage of unobservable variables is 0%.

With respect to case study 2, the solution of the MNSNDP is [4−6 8 12] (5 sensors), and the cost of this SN is $280.00. The solution of the SNDP defined by eq 1 is [2 3 9 12]. Even though it is composed of 4 sensors, its cost is higher ($735.00) than the corresponding one to the MNSNDP ($ 280.00). Precision constraints are related to the 23% of the process variables. As in case studies 1.a and 1.b, it is observed that the incorporation of precision constraints on key variables increases the cost more than the reduction achieved by decreasing the number of required estimable variables. The set of observable unmeasured variables is [1 4 10 11], and the 55.5% of the unmeasured variables are unobservable.

Regarding case study 3, the solution of the MNSNDP is [3 5−7 9 10 13 15 17 20 22 24] (12 sensors), and the cost of this SN is $2365.00. For case study 3a, the solution of the SNDP defined by eq 1 involves only 10 sensors [1 3 5−9 11 17 22], and its cost is $1448.00. Precision constraints are imposed on the 16.6% of the variables. The set of unmeasured observable variables is [2 4], and the 85.7% of the unmeasured variables remain unobservable. For this case study, the reduction in the number of required estimable variables lowers the cost more than the incorporation of precision restrictions. The opposite happens for case study 3b. Precision constraints are related to the 50% of the variables. The optimal SN that satisfies constraints on a set of key variables [1 3 5−11 15−17 20 21 22] (15 sensors), its cost is $2928.00, the set of unmeasured observable variables is [2 4 13 14 19 23], and only the 30% of the unmeasured variables are unobservable.

Case study 4 is similar to case study 3a. The solution of the MNSNDP for case study 4 is [3 7 9 10 14 17−27 31−33 35 37−39 41 43−52] (34 sensors), and the cost of this SN is $3478.38. Precision constraints are considered for the 17.3% of the variables. The solution of case study 4 involves only 19 sensors [10, 16, 31−33, 35, 37, 39−41, 43−48, 50−52], its cost is $1154.34, the set of unmeasured observable variables is [1 2 5 11 12 15 16 29 30 34 36 38 42 49], and 57.6% of the unmeasured variables remain unobservable.

With respect to case study 5, the solution of the MNSNDP is [1 2 5 13 15 21 25 31 35 37 44- 46 49 51 54−57 61 62 64 65 67 68 71 74−82] (35 sensors). The cost of this SN is $42 287.56. Precision constraints are considered for 7.3% of the variables. The solution of case study 5 contains 29 sensors, and its cost is $50 845.16. Only the unmeasured variables [52 58 69 70] are observable; therefore, the percentage of the unobservable ones is 92.4%.

Previous results are obtained for different process flowsheets. Therefore, results depend both on the process configuration and the percentage of variables for which restrictions are considered. In general, it can be concluded that the incorporation of precision constraints originates an increment of the cost greater than the reduction achieved by decreasing the number of required estimable variables. Only case studies 3a and 4 do not follow this tendency. Furthermore, the formulation of the SNDP defined by eq 1 favors an increment in the number of unobservable variables with respect to the obtained one solving the MNSNDP. Results show this fact except for case study 1.b. In this example, the solution is an expensive SN that satisfies the precision constraints and allows us to observe all of the unmeasured variables.

6. CONCLUSIONS

In this work, the solution of the SNDP for chemical plants is addressed using exact and stochastic algorithms. This challenging optimization problem involves a huge amount of binary variables, which represent the possible location of instruments.

First, an exact solution procedure, the LTTS, is implemented and used to obtain the global optima of middle scale SNDPs at the expense of consuming high amounts of computational resources. This shows that the guarantee of optimality that techniques based on tree search ensure may not be attained in practice when the number of the binary variable increases. Depending on the optimization problem complexity, the computational time may constitute a limiting factor to select the most appropriate solution strategy.

Besides the solution of the SNDP is tackled using multivariate EDAs. Among the broad spectrum of EDAs that use multivariate models, two representative procedures are selected that differ in the methods used for learning and sampling those models. In this sense, AffEDA uses MPM models and EBNA manages BNs. Furthermore, a comparative performance study is conducted to evaluate the benefits of increasing the complexity of the distribution model with respect to a memetic procedure based on univariate models, PBIL-SOTS.

Diverse SNDPs are formulated and solved using the procedures AffEDA, EBNA, and PBIL-SOTS. Problems differ in the type of model used to represent process operation (linear−nonlinear), restriction complexity, and the number of binary variables. In this sense, 3 flowsheets of incremental size (28, 52, and 82 streams) are analyzed for linear models, and two nonlinear examples extracted from the literature (CSTR and MFP) are used. For three case studies, reference values of the global optima are available that allow us to validate the best solutions attained using metaheuristics.

For the analyzed case studies, EBNA provides low-quality solutions and consumes significant computation resources when the problem size increases. The increment of the number of individual of its population is also analyzed for middle-scale and large-scale instances, but it does not enhance its performance. This study can not be carried out for the largest-size problem because the computation requirements are huge.

The performance of AffEDA is better than the corresponding to EBNA. For medium-size instances, AffEDA attains the reference minimum, but the convergence to that solution is not achieved for all the runs. For large problems, AffEDA is not able to find high-quality solutions, and the search is entrapped

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

in suboptimal regions. The increment of the population size enhances the behavior of the algorithm.

The procedure for pPBIL-SOTS has shown itself to be a rewarding performance for the solving of the formulated SNDPs. Good quality and reproducibility of solutions is achieved for problems of incremental size. The null or extremely low dispersion of the set of possible solutions indicates an appropriate behavior of the procedure for solving the combinatorial problem.

A direct relationship obviously exists between the computation time and the problem size. As mentioned before, the computation times are only reported for comparative purposes because the procedures are executed by an interpreter program such as Matlab, and the parallelization capability of PBIL-SOTS has not been properly exploited because PBIL instances are run sequentially.

Application results indicate that the best performances are achieved using pPBIL-SOTS and AffEDA for large population sizes. In contrast to AffEDA, pPBIL-SOTS uses a smaller number of individuals in the population. This reduces its exploration capability of the search space, but the incorporation of the local search mechanisms significantly enhances the exploitation capability of good regions.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.iecr.8b01680.

An outline of the pPBIL-SOTS procedure and the pseudocodes of pPBIL-SOTS, AffEDA, and EBNA (PDF)

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: msanchez@plapiqui.edu.ar. Phone: +54-291-486-1700.

ORCID

Mabel Sánchez: 0000-0002-0647-2717

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors are thankful for the financial support of Consejo Nacional de Investigaciones Científicas y Técnicas, Universidad Nacional de Rio Cuarto, and Universidad Nacional del Sur, Bahía Blanca, Argentina.

■ NOMENCLATURE

B = best individual of the generation
C = acquisition cost vector
E_{le} = degree of estimability of variable le
F = evaluation function
k = number of elite solutions
I = number of process variables
LR = learning rate
M = number of individuals of the population/subpopulation
MI = mutual information matrix
MS = mutation amount
nc = number of clusters
N = neighborhood of possible solutions
NPBIL = number of instances of PBIL executed in parallel
#MaxGeneration = maximum number of iterations for pPBIL
#maxiter = maximum number of iterations for SOTS
nso = number of iterations between allowable calls to SOTS
P = probability vector
ph = number of iterations before **h** is reset
$P_{interaction}$ = cross-over probability
P_{MUTA} = mutation probability
p_Q = joint probability distribution
$p(q_i/q_j)$ = conditional probability
Pr = probability
pso = application probability of SOTS
Pt = Tabu tenure period
Q = solution vector
Q = Rrandom I-dimensional vector
S_σ = set of key variables subject to precision constraints
S_E = set of key variables subject to estimability constraints
t = truncation selection parameter
$\hat{\sigma}_{lj}$ = standard deviation of the lp-th variable estimate

■ REFERENCES

(1) Bagajewicz, M. Design and retrofit of sensor networks in process plants. *AIChE J.* **1997**, *43*, 2300.
(2) Nguyen, D.; Bagajewicz, M. New efficient breadth_first/level traversal tree search method for the design and upgrade of sensor networks. *AIChE J.* **2011**, *57*, 1302.
(3) Nguyen, D. Q.; Bagajewicz, M. Design of Nonlinear Sensor Networks for Process Plants. *Ind. Eng. Chem. Res.* **2008**, *47*, 5529.
(4) Carnero, M.; Hernández, J.; Sánchez, M.; Bandoni, A. On the solution of the instrumentation selection problem. *Ind. Eng. Chem. Res.* **2005**, *44*, 358.
(5) Carnero, M.; Hernández, J.; Sánchez, M. A new metaheuristic based approach for the design of sensor networks. *Comput. Chem. Eng.* **2013**, *55*, 83.
(6) Benqlilou, C.; Graells, M.; Musulin, E.; Puigjaner, L. Design and retrofit of reliable sensor networks. *Ind. Eng. Chem. Res.* **2004**, *43*, 8026.
(7) Gerkens, C.; Heyen, G. Use of parallel computers in rational design of redundant sensor networks. *Comput. Chem. Eng.* **2005**, *29*, 1379.
(8) Carnero, M.; Hernández, J.; Sánchez, M. Optimal sensor network design and upgrade using tabu search. *Comput.-Aided Chem. Eng.* **2005**, *20*, 1447.
(9) Baluja, S. *Population-based incremental learning: a method for integrating genetic search based function optimization and competitive learning*. Technical Report CMU-CS-94−163; Carnegie Mellon University: Pittsburgh, PA, 1994.
(10) Hauschild, M.; Pelikan, M. An introduction and survey of estimation of distribution algorithms. *Swarm Evol. Comput.* **2011**, *1*, 111.
(11) Santana, R.; Larrañaga, P.; Lozano, J. Learning Factorizations in Estimation of Distribution Algorithms Using Affinity Propagation. *Evol. Comput.* **2010**, *18*, 515.
(12) Larrañaga, P.; Karshenas, H.; Bielza, C.; Santana, R. A review on probabilistic graphical models in evolutionary computation. *J. Heuristics* **2012**, *18*, 795.
(13) Bagajewicz, M.; Sánchez, M. Reallocation and Upgrade of Instrumentation in Process Plants. *Comput. Chem. Eng.* **2000**, *24*, 1945.
(14) Bagajewicz, M.; Sánchez, M. Design and upgrade of non-redundant and redundant linear sensor networks. *AIChE J.* **1999**, *45*, 1927.
(15) Romagnoli, J.; Sánchez, M. *Data Processing and Reconciliation for Chemical Process Operations*; Academic Press: San Diego, CA, 2000.
(16) Narasimhan, S.; Jordache, C. *Data Reconciliation and Gross Error Detection*; Gulf Publishing Company: Houston, TX, 2000.

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164

(17) Bagajewicz, M.; Cabrera, E. A New MILP Formulation for Instrumentation Network Design and Upgrade. *AIChE J.* **2002**, *48* (10), 2271.
(18) Chmielewski, D.; Palmer, T.; Manousiouthakis, V. On the Theory of Optimal Sensor Placement. *AIChE J.* **2002**, *48*, 1001.
(19) Kelly, J.; Zyngier, D. A new and improved MILP formulation to optimize observability, redundancy and precision for sensor network problems. *AIChE J.* **2008**, *54*, 1282.
(20) Larrañaga, P., Lozano, J.. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*. Kluwer; Boston, MA, 2002.
(21) Harik, G.; Lobo, F.; Sastry, K. Linkage Learning via Probabilistic Modeling in the Extended Compact Genetic Algorithm (ECGA). *Stud. Comput. Intell.* **2006**, *33*, 39.
(22) Echegoyen, C.; Santana, R.; Mendiburu, A.; Lozano, J. Comprehensive characterization of the behaviors of estimation of distribution algorithms. *Theor. Comput. Sci.* **2015**, *598*, 64.
(23) Martins, M. S. R.; Delgado, M. R. B. S.; Lüders, R.; Santana, R.; Goncalves, R. A.; de Almeida, C. P. Hybrid multi-objective Bayesian estimation of distribution algorithm: a comparative analysis for the multi-objective knapsack problem. *J. Heuristics* **2018**, *24* (1), 25.
(24) Alden, M.; Miikkulainen, R. MARLEDA: Effective distribution estimation through Markov random fields. *Theoretical Computer Science* **2016**, *633*, 4.
(25) Frey, B.; Dueck, D. Clustering by passing messages between data points. *Science* **2007**, *315*, 972.
(26) Buntine, W.Theory refinement on Bayesian networks; *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*; Morgan Kaufmann Publishers: San Francisco, CA, 1991, 52.
(27) Schwarz, G. Estimating the dimension of a model. *Ann. Statist* **1978**, *6*, 461.
(28) Henrion, M. Propagating uncertainty in Bayesian networks by probabilistic logic sampling; *Uncertainty in Artificial Intelligence*; Lemmer, J. F.; Kanal, L. N.; Eds.; Morgan Kaufmann Publishers: North Holland, Amsterdam, 1998; p 149.
(29) Bhushan, M.; Rengaswamy, R. Design of Sensor Network Based on the Signed Directed Graph of the Process for Efficient Fault Diagnosis. *Ind. Eng. Chem. Res.* **2000**, *39*, 999.
(30) Smith, H. W.; Ichiyen, N. Computer Adjustment of Metallurgical Balances. *Can. Znst. Mining Metall. Bull.* **1973**, *66*, 97.

DOI: 10.1021/acs.iecr.8b01680
Ind. Eng. Chem. Res. 2018, 57, 12149−12164