Article

Network Distance-Based Simulated Annealing and Fuzzy Clustering for Sensor Placement Ensuring Observability and Minimal Relative Degree

Daniel Leitold [1,2] , **Agnes Vathy-Fogarassy** [1,2] **and Janos Abonyi** [2,*]

[1] Department of Computer Science and Systems Technology, University of Pannonia, Egyetem u. 10, H-8200 Veszprém, Hungary; leitold@dcs.uni-pannon.hu (D.L.); vathy@dcs.uni-pannon.hu (A.V.-F.)

[2] MTA-PE Lendület Complex Systems Monitoring Research Group, University of Pannonia, Egyetem u. 10., POB. 158, H-8200 Veszprém, Hungary

[*] Correspondence: janos@abonyilab.com; Tel.: +36-88-624-000 (ext. 6078)

check for
updates

Abstract: Network science-based analysis of the observability of dynamical systems has been a focus of attention over the past five years. The maximum matching-based approach provides a simple tool to determine the minimum number of sensors and their positions. However, the resulting proportion of sensors is particularly small when compared to the size of the system, and, although structural observability is ensured, the system demands additional sensors to provide the small relative order needed for fast and robust process monitoring and control. In this paper, two clustering and simulated annealing-based methodologies are proposed to assign additional sensors to the dynamical systems. The proposed methodologies simplify the observation of the system and decrease its relative order. The usefulness of the proposed method is justified in a sensor-placement problem of a heat exchanger network. The results show that the relative order of the observability is decreased significantly by an increase in the number of additional sensors.

1. Introduction

The placement of sensors significantly affects the performance of identification, state estimation as well as fault detection and isolation (FDI) algorithms. The goal-oriented placement of sensors for dynamical systems is a challenging task [1]. Parameter estimation-oriented information entropy-based optimal sensor placement was investigated in [2]. Based on robust information entropy, a Bayesian sequential sensor placement algorithm for multi-type of sensor is also proposed [3]. Computational Fluid Dynamics (CFD) models were generated to predict wind-flow that used the data of a sensor placement utilised with prediction-value joint entropy [4]. For fault detection and isolation, an incremental analytical redundancy relation (ARR)-based algorithm was introduced in [5]. Structural analysis-based sensor placement strategies place additional sensors to isolate otherwise undisguisable faults [6]. In the initial sensor-constrained problems, the number of sensors is given and the detection time has to be minimized [7], while, as far as time-constrained problems are concerned, the aim is to minimize the number of required sensors that ensure a predefined detection time. In the

case of dynamical systems, the previously mentioned detection time is related to the relative degree of the system [8]. Besides good FDI performance, the optimal placement of the sensors should also ensure the observability of the systems [9].

Network representation-based structural controllability and observability analysis of dynamical systems is a hot topic in literature [10–16] since the introduction of this new approach [17]. The methodology utilizes the maximum matching algorithm to determine the minimum set of actuators and sensors required to grant structural controllability and observability for an arbitrary dynamical system. Although the results are very promising, the methodology can overestimate the number of necessary actuators [18] when the details of the edge dynamic are neglected [14].

When the observability analysis of the system is performed on a properly defined network, the problem concerning the observability analysis is reduced to a reachability problem [19]. In some cascaded systems, this highlights that the number of necessary sensors is significantly smaller than the size of the state variables, which makes the observation vulnerable and the relative degree, i.e., the minimum number of derivatives that is necessary to observe at least one signal of inputs, is very high. To deal with the vulnerability, Liu et al. proposed a methodology to grant robustness in the undirected representation of the system [20].

There is an almost infinite number of objectives that can be used to define a sensor placement problem [21]. In most of the cases, the optimization problem is formulated as that the instrumentation cost should be minimized and the precision of reconciled values should be ensured in a reliable, resilient and robust manner [22]. As the problem is NP-hard (non-deterministic polynomial-time hard), the mixed-integer programming-based approach is suitable to small-to-medium sized networks [23]. To handle the complexity of the problem, a heuristic approach is mostly followed. For example, to reconstruct the internal states of biochemical reaction systems with the minimum number of sensors, a graphical method has been developed [24]. Among population-based optimization algorithms, genetic algorithms (GAs) are the most frequently applied, e.g., GA was utilized to find a sensor configuration that minimizes cost and maximizes the reliability and observability of the system [25]. The main benefit of these gradient-free heuristic optimization methods is that they can be utilized with a wide range of models [26]. Simulated annealing (SA) is also widely employed optimization algorithm in sensor placement [27]. SA lends itself to be combined with other search algorithms, e.g., with local search heuristics [28] or other SA inner loop, which results in dual representation simulated annealing (DRSA) [29]. These techniques are often combined also in sensor placement, e.g., GA and SA were successfully integrated [30]. Recently, Particle Swarm Optimization (PSO) was applied in an optimal sensor location problem (OSLP) [31] and in the design of an ultrasonic Structural Health Monitoring (SHM) system [32]. According to these publications, heuristic optimization techniques are efficient in finding a small number of sensors with less time-consuming computations [33].

In this paper, to ensure observability and minimize the relative order of dynamical systems, two methodologies for sensor placement are introduced. The presented methods combine the metaheuristic SA optimization method with clustering. SA is chosen as it exhibits excellent levels of performance with regard to graph-represented combinatorial optimization (CO). The use of simulated annealing is strengthened by the fact that the relative degree of the system can be interpreted as a path in a network, and simulated annealing models a random walk on the search graph. Based on this concept, SA was applied to place micro-hydropowers in a water supply network [34] and network alignment in biological systems [35].

As sensors practically group the observed state variables, sensor placement can be considered as a special k-medoid clustering problem. This concept has been utilized in Ref. [36], where fuzzy c-means clustering algorithm was utilized to characterize the spatial distribution of the sensor placement problem.

The Clustering Large Applications (CLASA) algorithm [37] developed as a fast and robust solution to the well-known k-medoid clustering problem. The fundamental idea of the paper is that instead of considering the distance between the cluster centers and the clustered objects, a goal-oriented

sensor-placement algorithm can be derived by the introduction of a problem-relevant objective function into the scheme of the CLASA algorithm. The second novelty of the paper is that the random search of the resulted algorithm is fine-tuned by calculating the selection probabilities based on network distance-related fuzzy membership functions.

The structure of the paper is the following. In Section 2.1, the theoretical foundations and nomenclature of the relative degree of multiple-input and multiple-output (MIMO) dynamical systems are introduced. Section 2.2 presents the maximum matching-based sensor placement algorithm that ensures structural observability. In Section 2.3, the proposed optimization algorithms are introduced. Section 3 provides examples of application. Finally, Section 4 concludes the paper.

2. Sensor Placement to Ensure Observability and Minimal Relative Order

In Section 2.1, the relative degree of MIMO systems is defined and the sensor placement task is considered to be an optimization problem. Section 2.2 presents the first step of our methodology, which places the sensors as the structural observability of the system is guaranteed. Section 2.3 presents the proposed simulated annealing-based clustering algorithms that optimally place the additional sensors to minimize the relative degree of the system.

2.1. Problem Formulation

A nonlinear MIMO system can be represented by state-space model:

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t), \tag{1}$$

$$y(t) = h(x(t)), \tag{2}$$

where the vectors x, u and y stand for the state variables, inputs and outputs, respectively.

The model can be linearized and the resultant linear approximation represented by a linear state-space model [38]:

$$\dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}u(t), \tag{3}$$

$$y(t) = \mathbf{C}x(t). \tag{4}$$

When the number of state variables is denoted by N, the number of actuators by M and the number of sensors by K, then the matrices $\mathbf{A} \in \Re^{N \times N}$, $\mathbf{B} \in \Re^{N \times M}$ and $\mathbf{C} \in \Re^{K \times N}$ define how state variables influence each other, how the actuators influence the state variables, and how the sensors record the state variables, respectively.

For each pair of output y_i and input u_j ($i = 1, \ldots, K$ and $j = 1, \ldots, M$), the relative degree of the system $r_{i,j}$ can be defined as the minimum number of derivatives of $y_i(t)$ that is directly influenced by the change in $u_j(t)$. The relative degree for an arbitrary output i is defined as $r_i = \min_{j=1}^{M} r_{i,j}$, which requires the observer to observe the effect of at least one of the inputs.

When the aim of the observer is to estimate the effect of the $d(t) \in \Re^N$ disturbance vector that independently influences all state variables,

$$\dot{x}(t) = \mathbf{A}x(t) + \mathbf{E}d(t), \tag{5}$$

$$y_i(t) = \mathbf{C}x(t), \tag{6}$$

then $\mathbf{E}$ is an identity matrix $\mathbf{I}_N$ and the relative order of the system is $r = max_{i=1}^{K} r_i$.

As Figure 1 illustrates the relative degree of an x_j state variable can be interpreted as the $dist(i, j)$ length of the path between the j-th state variable and the nearest sensor, $C(i)$, selected as $\arg\min_j dist(i, j)$, while the relative order of the whole system is defined by the maximum of these minimal distances.

As the generation of a balanced placement of sensors is of interest, the set of sensor nodes S is determined by minimizing the following cost function:

$$\min_{S} cost(G, S, \beta) = \beta \max_{i=1}^{K} r_i + (1 - \beta) \frac{\sum_{i=1}^{K} r_i}{N}, \tag{7}$$

where parameter $\beta = [0, 1]$ weights between the maximum and the balance-related average of the relative order of the system.
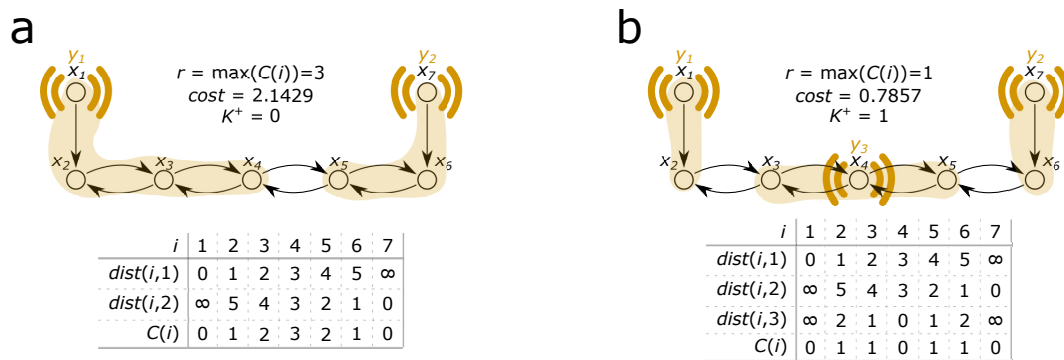

a

i	1	2	3	4	5	6	7
$dist(i,1)$	0	1	2	3	4	5	∞
$dist(i,2)$	∞	5	4	3	2	1	0
$C(i)$	0	1	2	3	2	1	0

b

i	1	2	3	4	5	6	7
$dist(i,1)$	0	1	2	3	4	5	∞
$dist(i,2)$	∞	5	4	3	2	1	0
$dist(i,3)$	∞	2	1	0	1	2	∞
$C(i)$	0	1	1	0	1	1	0

Figure 1. Illustration of the concept of relative degrees of the state variables and how one additional sensor decreases the relative degree of the system. (**a**) relative degrees in the case of two sensors that are necessary to grant observability; (**b**) relative degrees in the case of one additional sensor.

2.2. Placing Sensors to Ensure Observability

Besides the minimisation of the relative order, the observability of the system should also be ensured. Based on the Kalman rank criterion, a linear dynamical system is said to be structurally observable, if and only if the observability matrix $\mathcal{O} = \left[[\mathbf{C}]^T, [\mathbf{CA}]^T, [\mathbf{CA}^2]^T, \dots, [\mathbf{CA}^{N-1}]^T \right]^T$ is of full rank ($rank(\mathcal{O}) = N$) [39].

Based on the state-transition matrix $\mathbf{A}$, a graph $G(V, E)$ can be constructed where the set of vertices V represent the state variables and the edges $E = V \times V$ are determined by the nonzero elements of $\mathbf{A}$ (see Figure 2). In the graph representation, the relative degree $r_{i,j}$ can be defined as the distance between the input/disturbance d_j and the output y_i, which is the distance between the j-th and i-th nodes in the directed network $G(V, E)$, so the the optimization problem can be considered as a special graph partitioning problem. This graph-based representation is beneficial as the maximum matching of $MM \subseteq E$ can be used to determine the minimal number and places of the sensor nodes that are required to ensure the observability of the system.

As directed graphs can be represented as bipartite graphs, the maximum matching of the related bipartite graph was studied [14,17]. The endpoints of the directed edges in MM are the matched nodes ($V(MM)$), while others are the unmatched nodes ($\overline{V(MM)}$), thus $V(MM) \cup \overline{V(MM)} = V$. Based on [17], the unmatched nodes $\overline{V(MM)}$ of $G(V, E)$, where E is determined by $\mathbf{A}^T$, define the driver nodes, i.e., where the actuators should be placed. Furthermore, unmatched nodes of $G(V, E)$, where E is defined by $\mathbf{A}$, should be the sensor nodes. In other words, the nodes where sensors should be placed. If the determined driver nodes or sensor nodes are assigned to the system, then it becomes a structurally controllable or structurally observable system [14]. As the aim of this paper is to deal with system observability, $G(V, E)$ was used with the adjacency matrix based on $\mathbf{A}$. To ensure the structural observability of the system, the path-finding method was used to define the minimum number of mandatory sensors and their location [14].

In order to generate the output matrix $\mathbf{C}$ such that the relative order is minimal and the system observable, first the unmatched set of nodes S_f is generated, followed by the set of candidate sensor nodes S_c. The cardinality of the candidate sensor nodes is K^+ ($K^+ = |S_c|$). The resulting $\mathbf{C}$ is designed

based on the set of sensor nodes $S = S_f \cup S_c$ such that for each $x_i \in S$, c_j is a one-hot row vector whose i-th element is non-zero, so $\mathbf{C} = [c_1^T, \ldots, c_K^T]^T$.
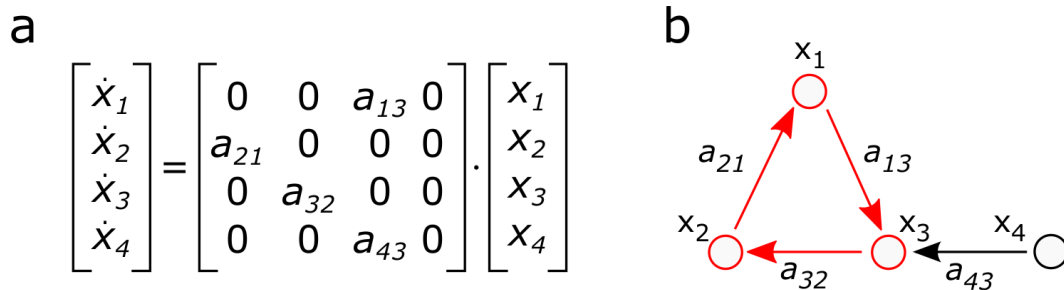
a

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & a_{13} & 0 \\ a_{21} & 0 & 0 & 0 \\ 0 & a_{32} & 0 & 0 \\ 0 & 0 & a_{43} & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

b

Figure 2. (**a**) example system $x(t) = \mathbf{A}x(t)$; (**b**) graph representation of (**a**); state variables $\{x_1, x_2, x_3\}$ are matched by $MM = \{a_{13}, a_{21}, a_{32}\} = \{(x_1, x_3), (x_2, x_1), (x_3, x_2)\}$ and the system is observable, since a sensor on x_4 can observe all state variables. If x_4 is removed from the system, then $\overline{V(MM)}$ is empty, which would mean that no sensor node could be identified as belonging to the system. This lack of the maximum matching-based approach can be eliminated by the path-finding method [14].

As the number of sensors which can be allocated without any restrictions is K^+, $\frac{K^+!}{|V \setminus S_f|!(|V \setminus S_f| - K^+!)}$ different combinations of possible placements of sensors exists. To solve this NP-hard problem, simulated annealing-based heuristic optimization algorithms are proposed in the following section.

2.3. Simulated Annealing and Fuzzy Clustering-based Output Configuration Design

To optimize the relative order and smoothness of these orders, the objective function (Equation (7)), which can be considered as a linear combination of the maximum and average distance of the sensors from the observed state variables was defined. Based on this interpretation, the minimisation problem can also be seen as a k-medoid clustering problem, where the centroid elements are the sensor nodes and the members of the clusters are the observed state variables.

To determine the locations of the additional sensors two approaches are proposed inspired by Clustering Large Applications based on Simulated Annealing algorithm (CLASA) [37] and the Geodesic Distance-based Fuzzy c-Medoid Clustering method (GDFCM) [40]. In the first algorithm, the CLASA algorithm is modified as follows. The original objective function of CLASA that calculates the distances between the objects and the medoids (in our case, the distances between the state variables and the sensors) is replaced by what was proposed in Equation (7). With the new objective function, not just the minimum relative order can be granted, but different sensor configurations with the same relative order can be distinguished to balance the load of the sensors. The search mechanism should be fine-tuned as the medoids of the fixed sensors (S_f) have fixed positions to ensure the observability of the system. The second algorithm enhances the random search of the resulted mCLASA algorithm by the introduction of distance-dependent selection probability calculated based on the Geodesic Distance-based Fuzzy c-Medoid Clustering method (GDFCM) [40].

In both algorithms, firstly the S_f fixed sensor nodes are determined to grant the observability property based on the unmatched set of nodes. Then, the candidate sensor nodes S_c are generated randomly, and the cost is calculated. Following this, in each iteration, one of the candidate sensor nodes (s^-) is randomly changed, and the new cost is calculated. From this, the difference between the new cost and the previously applied cost is calculated (Δ). The new placement of the sensors is accepted if $\Delta < 0$.

If $\Delta \geq 0$, the worsened placement of the sensor is accepted only with the probability defined as $\exp(\frac{-\Delta}{T^i})$, where T^i denotes the temperature of the cooling process of the SA in iteration i, whose dynamics can be seen in Equation (8). This randomization of the search allows for the algorithm to explore the search space and converge with the increase of the number of iterations:

$$T^{i+1} = \alpha T^i, \qquad \alpha = \left(\frac{T_{min}}{T_{max}}\right)^{1/maxiter}. \tag{8}$$

The pseudo code of the proposed modified Clustering Large Applications based on Simulated Annealing algorithm (mCLASA) can be seen in Algorithm 1. The algorithm possesses the following inputs and parameters: G denotes the network representation of the system from Equation (5), $\beta \in (0,1)$ stands for the parameter of the cost function from Equation (7), K^+ represents the number of additional sensors, *maxiter* is the number of iterations of the simulated annealing, T_{max} denotes the maximum, i.e., the initial temperature, and T_{min} stands for the minimum temperature. The following values were used for the parameters: $\beta = 0.5$, $maxiter = 500$, $T_{max} = 10$ and $T_{min} = 10^{-2}$.

Algorithm 1 Pseudo code of the modified CLASA algorithm (mCLASA).

1: **procedure** MCLASA($G, \beta, K^+, maxiter, T_{max}, T_{min}$)
2:     $\alpha = \left(\frac{T_{min}}{T_{max}}\right)^{1/maxiter}, T^1 = T_{max}$
3:     $S_f = getSensors(G)$ ▷ using the path finding method: [14]
4:     Let S_c be the set of the K^+ randomly chosen elements from $V \setminus S_f$
5:     $cost = cost(G, S_f \cup S_c, \beta)$ ▷ Equation (7)
6:     **for** $i = 1$ to *maxiter* **do**
7:         Let s^- be a randomly selected element from S_c
8:         $S_c' = S_c \setminus \{s^-\}$
9:         Let s^+ be a randomly chosen node from $V \setminus (S_f \cup S_c')$
10:        $S_c' = S_c' \cup \{s^+\}$
11:        $newcost = cost(G, S_f \cup S_c', \beta)$
12:        $\Delta = newcost - cost$
13:        **if** $\Delta < 0$ **then**
14:            $cost = newcost$
15:            $S_c = S_c'$
16:        **else**
17:            **if** $random() < \exp(\frac{-\Delta}{T^i})$ **then**
18:                $cost = newcost$
19:                $S_c = S_c'$
20:            **end if**
21:        **end if**
22:        $T^{i+1} = \alpha T^i$
23:     **end for**
24:     **return** *cost*
25: **end procedure**

The second method utilizes the GDFCM method [40] to extend the previously introduced mCLASA method. This extension increases the probability that in each iteration potentially better neighboring sensors are selected to replace the sensors placed in the previous iteration steps s^-. The algorithm considers the sensors as cluster centers (medoids), determines the cluster assignments of the state variables, and swaps the medoid (the sensor) with a randomly selected state variable. The random selection utilizes distance-based fuzzy membership values. The cluster memberships $\mu_{i,j}$ of the state variables $R(s_j)$ are determined by the fuzzy membership function (see Equation (9) in the description of the algorithm), where the distances are calculated in the undirected version of the graph (G^u). A roulette wheel selection [41] selects the new medoid according to the membership values. The search space of the algorithm is also controlled by the fuzzy exponent m which is decreased in each iteration by α_m, in the same manner as α in Equation (8).

　　　　The pseudo code of the suggested Geodesic Distance-based Fuzzy *c*-medoid Clustering with Simulated Annealing algorithm (GDFCMSA) can be seen in Algorithm 2. The algorithm possesses the following inputs and parameters: G denotes the network representation of the system, $\beta \in (0, 1)$ stands for the balance parameter of the cost function, K^+ represents the number of additional sensors, *maxiter* is the number of iterations of the simulated annealing, m_{max} denotes the maximum fuzzy component value and m_{min} stands for the minimum fuzzy component value. The following values were used for the parameters: $\beta = 0.5$, *maxiter* $= 500$, $T_{max} = 10$, $T_{min} = 10^{-2}$, $m_{max} = 10$ and $m_{min} = 2$.

Algorithm 2 Pseudo code of the Geodesic Distance-based Fuzzy *c*-Medoid Clustering with Simulated Annealing algorithm (GDFCMSA).

1: **procedure** GDFCMSA($G, \beta, K^+, maxiter, T_{max}, T_{min}, m_{max}, m_{min}$)

2:　　　$\alpha = \left(\frac{T_{min}}{T_{max}}\right)^{1/maxiter}, T^1 = T_{max}, \alpha_m = \left(\frac{m_{min}}{m_{max}}\right)^{1/maxiter}, m^1 = m_{max}$

3:　　　$S_f = getSensors(G)$　　　　　　　　　　　　　　　　　▷ using the path finding method: [14]

4:　　　Let S_c be the set of the K^+ randomly chosen state elements from $V \backslash S_f$

5:　　　$cost = cost(G, S_f \cup S_c, \beta)$　　　　　　　　　　　　　　　　　　　　　▷ Equation (7)

6:　　　**for** $i = 1$ to *maxiter* **do**

7:　　　　　Calculate the fuzzy membership values for G^u, $1 \leq i \leq K$ and $1 \leq j \leq N$:

$$\mu_{i,j} = \frac{1}{\sum_{k=1}^{K} \left(\frac{r_{i,j}}{r_{k,j}}\right)^{\frac{2}{m^i - 1}}} \tag{9}$$

8:　　　　　Let s^- be a randomly selected sensor node from S_c

9:　　　　　$S_c' = S_c \backslash \{s^-\}$

10:　　　　$R(s^-) = \{x_j | x_j \in V, s^- = \arg \max_{k=1}^{K} \mu_{k,j}\}$

11:　　　　**for all** $s_j \in R(s^-)$ **do**

12:　　　　　　$P(s_j) = \mu_{s^-, s_j}$

13:　　　　**end for**

14:　　　　$s^+ = \{x_j | x_j = roulette\_wheel\_selection(P)\}$

15:　　　　$S_c' = S_c' \cup \{s^+\}$

16:　　　　$newcost = cost(G, S_f \cup S_c', \beta)$

17:　　　　$\Delta = newcost - cost$

18:　　　　**if** $\Delta < 0$ **then**

19:　　　　　　$cost = newcost$

20:　　　　　　$S_c = S_c'$

21:　　　　**else**

22:　　　　　　**if** $random() < \exp(\frac{-\Delta}{T^i})$ **then**

23:　　　　　　　$cost = newcost$

24:　　　　　　　$S_c = S_c'$

25:　　　　　　**end if**

26:　　　　**end if**

27:　　　　$T^{i+1} = \alpha T^i, m^{i+1} = \alpha_m m^i$

28:　　　**end for**

29:　　　**return** *cost*

30: **end procedure**

---

3. Results

In this section, four case studies will be introduced to demonstrate the applicability of the previously presented methods. Firstly, the details of the case studies will be given. We evaluate the proposed methods from two aspects. Firstly, the effect of the additional sensor nodes on the cost function and to the relative order of the system is studied; secondly, the speed of converge is analyzed.

3.1. Description of the Case Studies

The first case study is related the control of Heat Exchanger Networks (HENs). HENs are widely studied dynamical systems because the complexity of interlinked heat exchangers requires advanced process monitoring and control algorithms. The network topology of the studied HEN is shown in Figure 3. The network consists of six hot streams, two cold streams, ten heat exchanger cells and two utility coolers [42]. The state-transition matrix **A** of the problem can be easily determined based on the structure of the process [19], which results in results in 22 state variables.

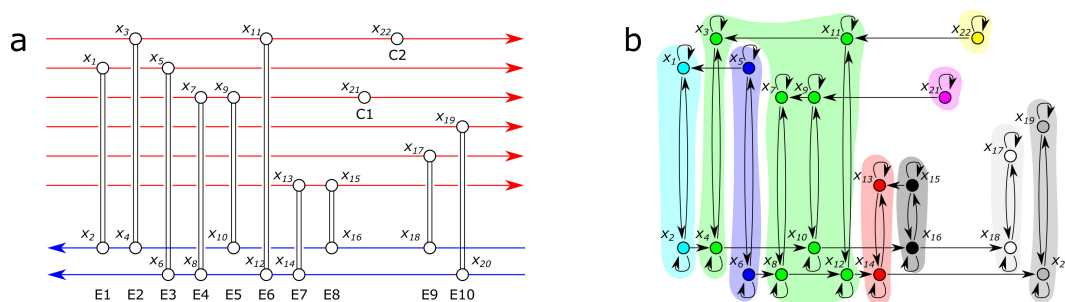**Figure 3.** (**a**) the structure of the studied heat exchanger network represents how cold and hot streams influence each other; (**b**) the network representation of the state variables are related to the temperatures of the streams depicted in (**a**). Colors indicate strongly connected components.

Following the detailed analysis of the problem with regard to the placement of sensors in the HEN, three other dynamical systems will be analyzed to illustrate the applicability of the methods on larger examples. These benchmark examples are not typical control-relevant problems as they are used to study model-reduction algorithms [43], but these problems illustrate how the algorithms are able to detect structurally efficient placements. The first two additional examples are based on Modified Nodal Analysis (MNA) [44] and have 578 and 980 state variables, respectively. The last case study is based on the state space model of a partial element equivalent circuit (PEEC) of a patch antenna structure with 172 inductances, 6990 mutual inductances, and 2100 capacitances that define 480 state variables [45]. Further information about the case studies can be found in the Supplementary Material of the paper.

The parameters of the resulted case studies are summarized in Table 1. The network representations of the case studies and the resulted sensor placements are presented in the Supplementary Material of the paper.

Table 1. The main parameters of the networks studied in the examples. $|V|$ denotes the number of state variables, $|E|$ stands for the number of the connections between the state variables, $|S_f|$ represents the number of fixed sensor nodes, and K^+ denotes the number of additional nodes to be analyzed.

Network	Description	$|V|$	$|E|$	$|S_f|$	K^+
HEN	A simple heat exchanger network.	22	56	3	$0, \ldots, 15$
MNA_1	Modified Nodal Analysis of a multiport, voltage sources.	578	1694	3	$0, \ldots, 35$
MNA_4	Modified Nodal Analysis of a multiport, voltage sources.	980	2872	2	$0, \ldots, 35$
peec	Partial element equivalent circuit (PEEC) model.	480	1346	1	$0, \ldots, 35$

As random search-based techniques were evaluated, each result was evaluated based on 100 independent runs of the algorithms. The parameters of the algorithms were the same in each scenario: $\beta = 0.5$, $T_{max} = 10$, $T_{min} = 10^{-2}$, $m_{max} = 10$, $m_{min} = 2$. The reduction rates α and α_m were determined by Equation (8), as can be seen in Algorithms 1 and 2.

3.2. Effect of the Additional Nodes and the Robustness of the Optimization Algorithms

In all the examples, firstly the S_f set of fixed sensors that are required to ensure the observability of the system is determined. The number of sensors added to improve the dynamical properties of the system is represented by the cardinality of $|K^+|$.

To evaluate the robustness of the random search-based algorithms, the results of 100 independent runs were visualized in boxplots to show the distribution of the optimized cost functions. As can be seen in Figures 4 and 5, the algorithms are consistent in most of the cases; however, the presence of outliers show that the algorithms can fail in local optimums.
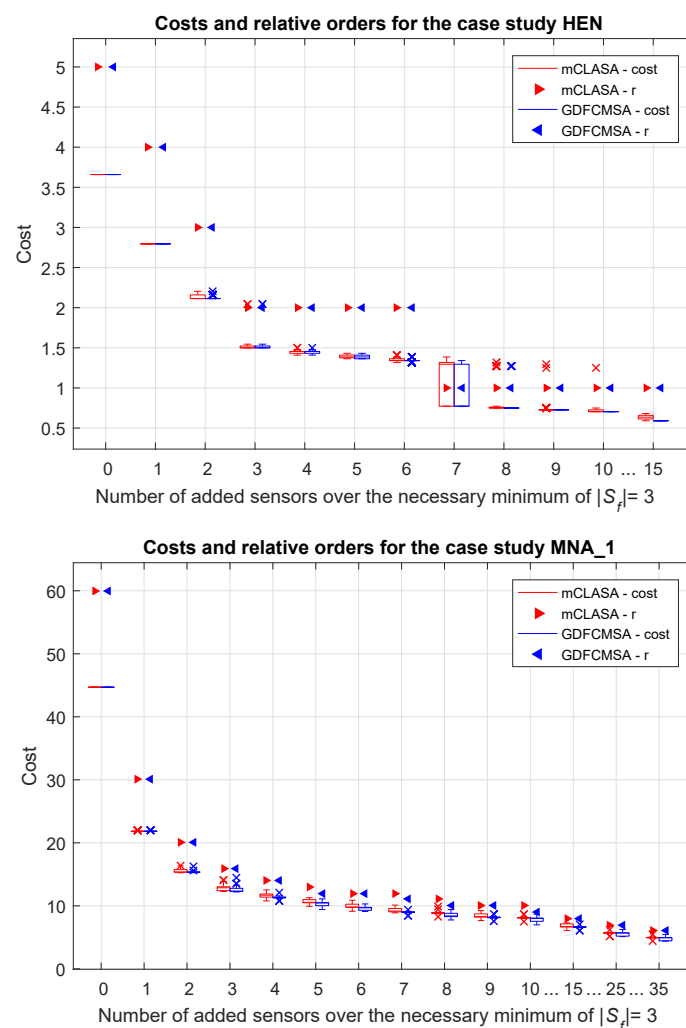

Figure 4. The values of the cost functions and relative degrees with regard to the number of added sensors in the case studies HEN and MNA_1. The box plots show the interquartile ranges and the medians of the cost functions, while the scatter plots visualize the minimum value of the relative order for 100 independent runs. For each value of the *x*-axis, the results provided by the mCLASA are shown on the left-hand side, and the results provided by the GDFCMSA algorithm are shown on the right-hand side. The triangles represent the relative orders, while the plus signs denote the outliers (if they exist) of the corresponding boxes.
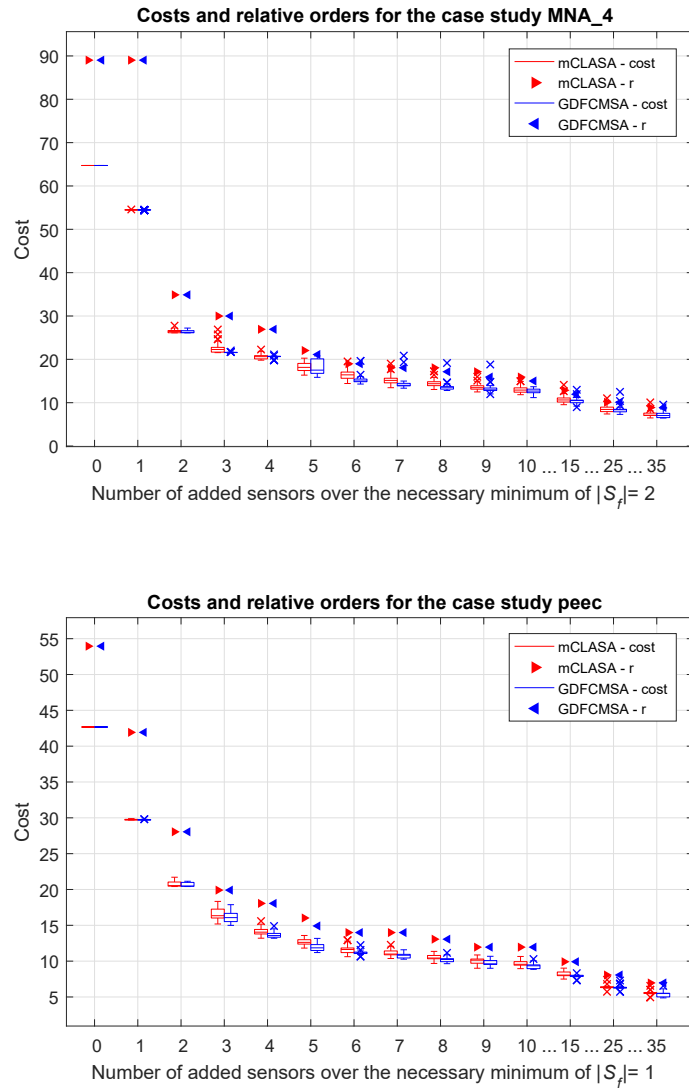
Figure 5. The values of the cost functions and relative degrees with regard to the number of added sensors in the case studies MNA_4 and peec. The box plots show the interquartile ranges and the medians of the cost functions, while the scatter plots visualize the minimum value of the relative order for 100 independent runs. For each value of the *x*-axis, the results provided by the mCLASA are presented on the left-hand side, while the results provided by the GDFCMSA algorithm are shown on the right-hand side. The triangles denote the relative orders, while the plus signs represent the outliers of the corresponding boxes.

It is visible that in all cases both the cost functions and the relative orders of the observed state variables decrease significantly as the number of sensors increases, so a relatively small quantity of additional sensors can cause a noticeable improvement in terms of the relative order compared to the observer designed only to ensure observability. It can also be pointed out that additional sensors ($K^+ = 25$ or 35) do not significantly reduce the order of the system.

The heights of the box plots show the interquartile ranges of the optimized cost functions. Higher box plots reflect higher uncertainty in the solutions which could indicate the increased difficulty of the sensor placement problem. For example, in Figure 4, for $K^+ = 7$, the boxplot shows a much wider distribution than for other K^+ that can be explained there is a transient in the structure of the solutions.

3.3. Convergence Analysis

In the second phase of the analysis, the convergence of the proposed methods was tested. In Figure 6, the convergences of the methods can be seen for the HEN case study with the parameter setting $K^+ = 3$. The results and the network representation of the other case studies are presented in the Supplementary Material. In the figure, it can be seen that how fast the methods converge in terms of the number of iterations. With the addition of three sensors, the cost function was reduced by more than 50% over the 500 iterations, which illustrate well the benefit of the proposed methods.
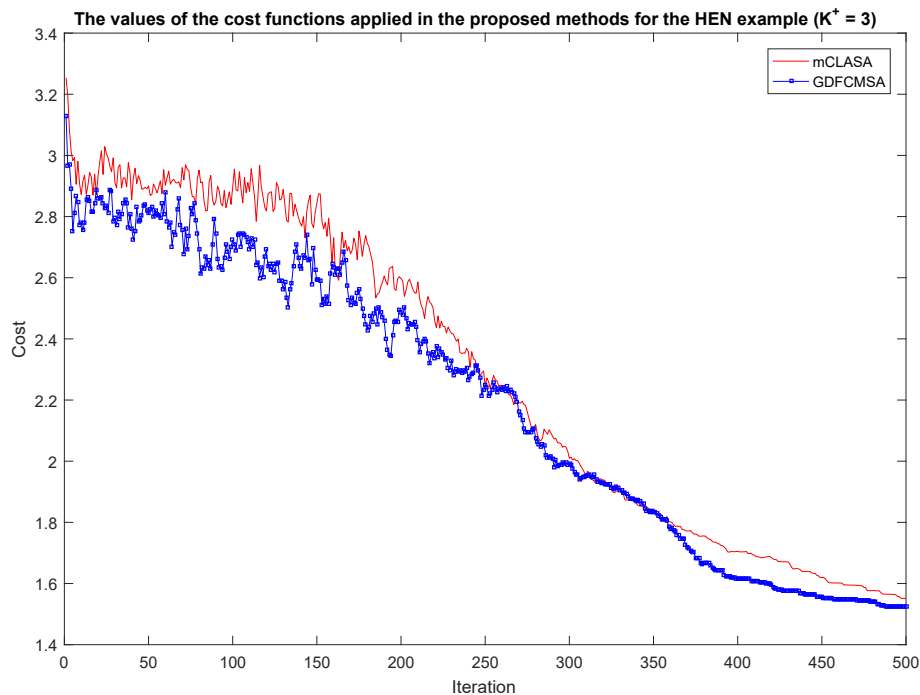

Figure 6. Analysis of the convergence with regard to the proposed methods in the case of the HEN. The colour red denotes the method mCLASA while the colour blue denotes the method GDFCMSA. It is clearly visible that the value of *maxiter* should be high because the methods also converge after 500 iterations. GDFCMSA provides a slightly better solution.

In this figure, it can also be seen that it is worth selecting a high value of the parameter *maxiter* because, even during the 500th iteration, the cost functions still show a small decrease.

From this figure, it is evident that the fuzzy extension of the algorithm improves the speed of convergence which is also reflected from the slightly better performances shown in Figures 4 and 5.

The durations of the calculations were also measured with *maxiter* = 500 to provide an illustrative comparison between the algorithm and the complexity of the problem. The results are presented in Table 2. The durations of calculations are valid for our MATLAB implementation (R2016a, The MathWorks, Natick, MA, USA) when run on a notebook using Windows 10 (Microsoft, Redmond, WA, USA), an Intel Core i7-6600U processor (Santa Clara, CA, USA) and 16 GB of RAM.

The short running times shown in Table 2 raise the question of whether the exhaustive enumeration of the solutions could handle the problem and if the heuristic search is necessary. The problem is combinatorially complex, for each K^+ $\binom{N-|S_f|}{K^+}$ possible solutions should be examined. In the HEN case study for $K^+ = 9$, the SA examines only 500 solutions instead of the 92,378 possible configurations. For larger problems, the difference is more significant, e.g., the example of MNA_4 with $K^+ = 35$ would require $\binom{978}{35} \approx 2.4 \times 10^{64}$ function evaluations, although this problem cannot be considered as a huge system.

Table 2. Durations of calculations in seconds in terms of the optimal placement of K^+ additional sensors. The presented results are the means of 100 independent runs.

K^+	HEN		MNA_1		MNA_4		peec	
	mCLASA	GDFCMSA	mCLASA	GDFCMSA	mCLASA	GDFCMSA	mCLASA	GDFCMSA
1	0.0158 s	0.0164 s	0.1311 s	0.1396 s	0.1795 s	0.1892 s	0.0704 s	0.0715 s
2	0.0173 s	0.0183 s	0.1752 s	0.1689 s	0.2240 s	0.2306 s	0.0945 s	0.0955 s
3	0.0205 s	0.0227 s	0.2230 s	0.2120 s	0.2676 s	0.2922 s	0.1176 s	0.1179 s
5	0.0233 s	0.0280 s	0.2638 s	0.2999 s	0.3724 s	0.3749 s	0.1602 s	0.1614 s
10	0.0310 s	0.0315 s	0.3965 s	0.4106 s	0.6353 s	0.7106 s	0.2829 s	0.2850 s
15	0.0427 s	0.0428 s	0.5984 s	0.6721 s	0.9049 s	0.9176 s	0.4631 s	0.4237 s
25	n.a.	n.a.	0.8052 s	0.8170 s	1.5417 s	1.4754 s	0.6385 s	0.6862 s
35	n.a.	n.a.	1.3258 s	1.3404 s	2.1008 s	2.0255 s	0.9229 s	1.0077 s

4. Conclusions

To ensure the structural observability in addition to fast and robust observer response, two clustering- and simulated annealing-based methods were proposed.

Additional sensors are placed into the system based on the CLASA algorithm by the mCLASA algorithm. The placements of additional sensors is further improved by the Geodesic Distance-based Fuzzy *c*-Medoid Clustering with Simulated Annealing algorithm (GDFCMSA) as their positioning is based on a geodesic distance-based fuzzy membership functions. Simulated annealing is applied by both algorithms to minimize the cost function, which in turn minimizes the maximum and average of the relative orders of the system to generate a balanced placement of sensors.

A slightly better solution is provided by the GDFCMSA algorithm than the mCLASA method at the expense of some additional computational resources for the evaluation of the membership degrees. Solutions to massive problems are generated by both methods over a short period of time, which illustrates the applicability of these methods in an industrial setting.

Although this paper deals with the placement of sensors, the proposed methodology can be applied to control configuration design as well, since this is the dual problem of the design of the output configuration.

Abbreviations

Abbreviations used in this article:

CLASA: Clustering Large Applications based on Simulated Annealing
GDFCM: Geodesic Distance-based Fuzzy c-Medoid Clustering method
GDFCMSA: Geodesic Distance-based Fuzzy c-Medoid Clustering with Simulated Annealing method
HEN: Heat Exchanger Network
mCLASA: modified Clustering Large Applications based on Simulated Annealing
MIMO: Multiple-input and multiple-output
SA: Simulated Annealing

Nomenclature

$\boldsymbol{x}$	vector of state variables
$\boldsymbol{u}$	vector of inputs
$\boldsymbol{d}$	vector of disturbances
$\boldsymbol{y}$	vector of outputs
$\mathbf{A}$	state-transition matrix
$\mathbf{B}$	input matrix
$\mathbf{E}$	disturbance matrix
$\mathbf{C}$	output matrix
$\mathcal{O}$	observability matrix
N	number of state variables
M	number of inputs
K	number of outputs
K^+	number of additional sensor nodes
$\boldsymbol{b}^j$	jth column of matrix $\mathbf{B}$
$\boldsymbol{b}_j$	jth row of matrix $\mathbf{B}$
$r_{i,j}$	relative degree of y_i and u_j
r_i	relative degree of y_i
r	relative order of the system
$G(V, E)$	network representation of the system
V	set of vertices of the network representation of the system
E	connections between the state variables
G^u	undirected representation of G
S	set of sensor nodes
S_f	set of fixed sensor nodes
S_c	set of candidate sensor nodes
β	weighting parameter of cost function
MM	edges of a maximum matching
$V(MM)$	matched nodes
$\overline{V(MM)}$	unmatched nodes
Δ	difference between cost functions in SA
T_{max}, T_{min}	maximum and minimum temperature in SA
m_{max}, m_{min}	maximum and minimum fuzzy exponent in SA
T^i	temperature in iteration i
m^i	fuzzy exponent in iteration i
α	reduction rate of temperature
α_m	reduction rate of the fuzzy exponent
$maxiter$	number of iterations of SA
$R(s_j)$	set of the state variables of the cluster of sensor node s_j
$\mu_{i,j}$	fuzzy membership function

References

1. Chmielewski, D.J.; Palmer, T.; Manousiouthakis, V. On the theory of optimal sensor placement. *AIChE J.* **2002**, *48*, 1001–1012. [CrossRef]
2. Papadimitriou, C.; Lombaert, G. The effect of prediction error correlation on optimal sensor placement in structural dynamics. *Mech. Syst. Signal Process.* **2012**, *28*, 105–127. [CrossRef]
3. Yuen, K.V.; Kuok, S.C. Efficient Bayesian sensor placement algorithm for structural identification: A general approach for multi-type sensory systems. *Earthq. Eng. Struct. Dyn.* **2015**, *44*, 757–774. [CrossRef]
4. Papadopoulou, M.; Raphael, B.; Smith, I.F.; Sekhar, C. Hierarchical sensor placement using joint entropy and the effect of modeling error. *Entropy* **2014**, *16*, 5078–5101. [CrossRef]
5. Rosich, A.; Sarrate, R.; Puig, V.; Escobet, T. Efficient optimal sensor placement for model-based FDI using an incremental algorithm. In Proceedings of the 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, 12–14 December 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 2590–2595.
6. Düştegör, D.; Frisk, E.; Cocquempot, V.; Krysander, M.; Staroswiecki, M. Structural analysis of fault isolability in the DAMADICS benchmark. *Control Eng. Pract.* **2006**, *14*, 597–608. [CrossRef]
7. Berger-Wolf, T.Y.; Hart, W.E.; Saia, J. Discrete sensor placement problems in distribution networks. *Math. Comput. Model.* **2005**, *42*, 1385–1396. [CrossRef]
8. Isidori, A. *Nonlinear Control Systems*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
9. Boukhobza, T.; Hamelin, F. State and input observability recovering by additional sensor implementation: A graph-theoretic approach. *Automatica* **2009**, *45*, 1737–1742. [CrossRef]
10. Yan, G.; Tsekenis, G.; Barzel, B.; Slotine, J.J.; Liu, Y.Y.; Barabási, A.L. Spectrum of controlling and observing complex networks. *Nat. Phys.* **2015**, *11*, 779–786. [CrossRef]
11. Ruths, J.; Ruths, D. Control profiles of complex networks. *Science* **2014**, *343*, 1373–1376. [CrossRef] [PubMed]
12. Pósfai, M.; Liu, Y.Y.; Slotine, J.J.; Barabási, A.L. Effect of correlations on network controllability. *Sci. Rep.* **2013**, *3*, 1067. [CrossRef] [PubMed]
13. Leitold, D.; Vathy-Fogarassy, Á.; Süle, Z.; Manchin, R.; Abonyi, J. Measuring sustainability—Network science based analysis of water resources models. In Proceedings of the 7th EDSI Conference, Helsinki, Finland, 24–27 May 2016; p. 189.
14. Leitold, D.; Vathy-Fogarassy, Á.; Abonyi, J. Controllability and observability in complex networks—The effect of connection types. *Sci. Rep.* **2017**, *7*, 151. [CrossRef] [PubMed]
15. Zhang, X.; Lv, T.; Yang, X.; Zhang, B. Structural controllability of complex networks based on preferential matching. *PLoS ONE* **2014**, *9*, e112039. [CrossRef] [PubMed]
16. Kang, L.; Liu, Y. Design of Control Structure for Integrated Process Networks Based on Graph-Theoretic Analysis. *Chem. Eng. Trans.* **2017**, *61*, 1711–1716.
17. Liu, Y.Y.; Slotine, J.J.; Barabási, A.L. Controllability of complex networks. *Nature* **2011**, *473*, 167–173. [CrossRef] [PubMed]
18. Müller, F.J.; Schuppert, A. Few inputs can reprogram biological networks. *Nature* **2011**, *478*, E4–E5. [CrossRef] [PubMed]
19. Varga, E.; Hangos, K.; Szigeti, F. Controllability and observability of heat exchanger networks in the time-varying parameter case. *Control Eng. Pract.* **1995**, *3*, 1409–1419. [CrossRef]
20. Liu, X.; Mo, Y.; Pequito, S.; Sinopoli, B.; Kar, S.; Aguiar, A.P. Minimum robust sensor placement for large scale linear time-invariant systems: A structured systems approach. *IFAC Proc. Vol.* **2013**, *46*, 417–424. [CrossRef]
21. Sharon, L.P.; Rex, K.K. *Optimization Strategies for Sensor and Actuator Placement*; Technical Report; NASA Langley Research Center: Hampton, VA, USA, 1999.
22. Bagajewicz, M.J. Design and retrofit of sensor networks in process plants. *AIChE J.* **1997**, *43*, 2300–2306. [CrossRef]
23. Watson, J.P.; Hart, W.E.; Berry, J.W. Scalable high-performance heuristics for sensor placement in water distribution networks. In *Impacts of Global Climate Change*; American Society of Civil Engineers: Reston, VA, USA, 2005; pp. 1–12.
24. Liu, Y.Y.; Slotine, J.J.; Barabási, A.L. Observability of complex systems. *Proc. Natl. Acad. Sci. USA* **2013**, *110*, 2460–2465. [CrossRef] [PubMed]

25. Carballido, J.A.; Ponzoni, I.; Brignole, N.B. CGD-GA: A graph-based genetic algorithm for sensor network design. *Inf. Sci.* **2007**, *177*, 5091–5102. [CrossRef]

26. Aguirre-Salas, L.; Begovich, O.; Ramirez-trevino, A. Sensor assignment for observability in interpreted petri nets. *IFAC Proc. Vol.* **2004**, *37*, 441–446. [CrossRef]

27. Chiu, P.; Lin, F.Y. A simulated annealing algorithm to support the sensor placement for target location. In Proceedings of the 2004 Canadian Conference on Electrical and Computer Engineering, Niagara Falls, ON, USA, 2–5 May 2004; IEEE: Piscataway, NJ, USA, 2004; Volume 2, pp. 867–870.

28. Martin, O.C.; Otto, S.W. Combining simulated annealing with local search heuristics. *Ann. Oper. Res.* **1993**, *63*, 57–75. [CrossRef]

29. Torres-Jimenez, J.; Izquierdo-Marquez, I.; Garcia-Robledo, A.; Gonzalez-Gomez, A.; Bernal, J.; Kacker, R.N. A dual representation simulated annealing algorithm for the bandwidth minimization problem on graphs. *Inf. Sci.* **2015**, *303*, 33–49. [CrossRef]

30. Worden, K.; Burrows, A. Optimal sensor placement for fault detection. *Eng. Struct.* **2001**, *23*, 885–901. [CrossRef]

31. Qin, B.Y.; Lin, X.K. Optimal sensor placement based on particle swarm optimization. *Adv. Mater. Res. Trans. Technol. Publ.* **2011**, *271*, 1108–1113. [CrossRef]

32. Blanloeuil, P.; Nurhazli, N.A.; Veidt, M. Particle swarm optimization for optimal sensor placement in ultrasonic SHM systems. In *Nondestructive Characterization and Monitoring of Advanced Materials, Aerospace, and Civil Infrastructure 2016*; International Society for Optics and Photonics: Bellingham, WA, USA, 2016; Volume 9804, p. 98040E.

33. Zhang, X.; Li, J.; Xing, J.; Wang, P.; Yang, Q.; Wang, R.; He, C. Optimal sensor placement for latticed shell structure based on an improved particle swarm optimization algorithm. *Math. Probl. Eng.* **2014**, *2014.* [CrossRef]

34. Samora, I.; Franca, M.J.; Schleiss, A.J.; Ramos, H.M. Simulated annealing in optimization of energy production in a water supply network. *Water Res. Manag.* **2016**, *30*, 1533–1547. [CrossRef]

35. Mamano, N.; Hayes, W. SANA: Simulated Annealing Network Alignment Applied to Biological Networks. *arXiv* **2016**, arXiv:1607.02642.

36. Zhang, X.X.; Li, H.X.; Qi, C.K. Spatially constrained fuzzy-clustering-based sensor placement for spatiotemporal fuzzy-control system. *IEEE Trans. Fuzzy Syst.* **2010**, *18*, 946–957. [CrossRef]

37. Chu, S.C.; Roddick, J.F.; Pan, J.S. *A Comparative Study and Extension to K-Medoids Algorithms*; Contemporary Development Company: Hongkong, China, 2001; pp. 1–10.

38. Sastry, S.S.; Isidori, A. Adaptive control of linearizable systems. *IEEE Trans. Autom. Control* **1989**, *34*, 1123–1131. [CrossRef]

39. Kalman, R.E. Mathematical description of linear dynamical systems. *J. Soc. Ind. Appl. Math. Ser. A Control* **1963**, *1*, 152–192. [CrossRef]

40. Király, A.; Vathy-Fogarassy, Á.; Abonyi, J. Geodesic distance based fuzzy c-medoid clustering–searching for central points in graphs and high dimensional data. *Fuzzy Sets Syst.* **2016**, *286*, 157–172. [CrossRef]

41. Goldberg, D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*; Addison-Wesley: Boston, MA, USA, 1989.

42. Westphalen, D.L.; Young, B.R.; Svrcek, W.Y. A controllability index for heat exchanger networks. *Ind. Eng. Chem. Res.* **2003**, *42*, 4659–4667. [CrossRef]

43. Chahlaoui, Y.; Van Dooren, P. *A Collection of Benchmark Examples for Model Reduction of Linear Time Invariant Dynamical Systems*; The University of Manchester: Manchester, UK, 2002; pp. 1–26.

44. Odabasioglu, A.; Celik, M.; Pileggi, L.T. PRIMA: Passive reduced-order interconnect macromodeling algorithm. In Proceedings of the 1997 IEEE/ACM International Conference on Computer-Aided Design, Santa Clara, CA, USA, 9–13 November 1997; pp. 58–65.

45. Heeb, H.; Ruehli, A.E.; Bracken, J.E.; Rohrer, R.A. Three dimensional circuit oriented electromagnetic modeling for VLSI interconnects. In Proceedings of the IEEE 1992 International Conference on Computer Design: VLSI in Computers and Processors, ICCD'92, Cambridge, MA, USA, 11–14 October 1992; pp. 218–221.