Distributed Fault Diagnosis using Minimal Structurally Over-Determined Sets: Application to a Water Distribution Network

Vikas Gupta and Vicenç Puig

Abstract- In this paper, a distributed fault diagnosis algorithm for large scale systems using agents has been proposed. In the off-line phase, this fault diagnosis approach starts from obtaining the minimal structurally over-determined (MSO) sets using the system model and the set of available sensors. These MSO sets are converted into a graph. This graph is further divided into various subgraphs using a partition algorithm. Each subgraph corresponds to a subsystem. From various subgraphs, different local fault signature matrices for various subsystems are obtained. Finally, in the on-line phase, using various local fault signature matrices, a set of diagnoser agents are created that allow the global diagnosis in a large scale system. The entire proposed distributed fault diagnosis approach is divided into five different blocks. In order to illustrate the application of the proposed approach, a case study based on the Barcelona drinking water network (DWN) is used.

I. INTRODUCTION

Distributed fault diagnosis is becoming more and more common in industries, to diagnose faults in any large scale system. There are a lot of disadvantages using centralized fault diagnosis in large-scale systems, since in a centralized implementation all the information has to be collected in one location which is generally not possible or very difficult. Moreover, a centralized system needs a high performance centralized unit which generally in most cases is not available. Due to these difficulties in recent years distributed fault diagnosis techniques have been investigated [10]. In distributed fault diagnosis [1] [2], the global diagnoses for the complete system can be computed from the results in all agents and local diagnose is computed from the results of one agent. In distributed fault diagnosis [3] [8], a global coordination process is not necessary and each subsystem depends on a local diagnoser for local diagnosis tasks and communicating with the remaining local diagnosers until a global diagnosis is produced.

In this paper, a distributed fault diagnosis algorithm for large scale systems using agents has been proposed. In the off-line phase, this fault diagnosis approach starts from obtaining the minimal structurally over-determined (MSO) sets using the system model and the set of available sensors. These MSO sets are converted into a graph. This graph is further divided into various subgraphs using a partition algorithm. Each subgraph corresponds to a subsystem. From various subgraphs, different local fault signature matrices for various subsystems are obtained. Finally, in the on-line phase, using various local fault signature matrices, a set of diagnoser agents are created that allow the global diagnosis in a large scale system. For each subsystem, a local fault signature matrix is then obtained as next step. Using each local fault signature matrix, a local diagnoser is implemented by means of an *agent*. This agent is responsible for local diagnoses in a subsystem and communicating with other agents of others subsystems to perform global diagnosis. In order to demonstrate the applicability of the proposed approach, a case study based on the Barcelona drinking water network (DWN) is used.

The remaining part of the paper is structured as follows: Section II presents the problem description. Section III illustrates the proposed approach. Section IV describes the implementation of the solution by using the Barcelona DWN case study. Finally, in Section V conclusions are presented.

II. OVERVIEW OF THE PROPOSED APPROACH

The goal of proposed approach is to obtain a set of local diagnosers that are able to perform distributed diagnosis by means of an agent based implementation. This approach has an off-line phase that starts by obtaining a set of minimal structurally over-determined sets (MSOs) that can be obtained from a system structural matrix *M (z, x)* [15]. *M* consists of set of constraints (equations) *z* and variables *x,* some of them known and other unknown. A MSO can be defined as a part of the over-constrained part of a system structure (represented in graph from) which if one constraint is removed it will make the subsystem to become just-constrained. The procedure to find MSO sets is by examining the set *M* of constraints of a proper structurally over-constrained structure graph. Then, *MSO* sets are converted into vertex and edge graph, any variable present in rows of *MSO* sets make that particular row, a vertex of the graph and all the variables present at same location of two rows is connected edge between the vertexes. From this vertex and edge graph, a set of subgraphs are generated by using a partition

Vikas Gupta and Vicenç Puig are with Advanced Control Systems (SAC) research group at Universitat Politècnica de Catalunya (UPC), Barcelona, Spain (e-mail: vikas.gupta@upc.edu).

algorithm. The first step to implement the partition algorithm is to find the strongly connected vertices. A strongly connected vertex is the one which has maximum number of edges. This vertex will be the basis for forming the first subsystem being its core. Second subsystem is formed by second strongly connected vertex and so on. The important condition is that no two subgraphs can have same vertex but same edge can be shared. Together all the subgraphs must contain all the vertices of a system, that is, no vertex must be left. Every vertex must be part of any one subsystem and the subsystem should be least connected. After this, the fault signature matrix associated to the subset of MSOs generated for each subsystem is obtained. Every subsystem has one local fault signature which contains unshared and shared variables and also all the subsystems have one common global fault signature matrix which contains shared variables among various subsystems. A fault signature matrix is created by converting all elements of each subsystem matrix MSO_i into 0 and 1. 0 is maintained as zero while all nonzero elements are converted into 1. After creating local fault signature matrix for each subsystem, from each local fault signature matrix a local diagnoser is created and implemented by means of an *agent* named $A_1, A_2, A_3, \ldots, A_n$. The agent in each local fault signature is responsible for local diagnoses in a subsystem and communicating with other agents of others subsystems, allows producing global diagnoses.

III. Description of the Proposed Approach

The proposed approach presented in the previous section can be decomposed in five blocks. Block diagram of proposed methodology is shown in Figure 1.

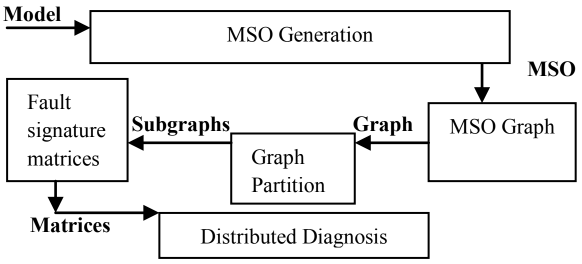

Fig 1. Various blocks of proposed approach

Block 1: Formation of MSO sets
Input to the Block: The system structural matrix (*M*)
Output of the Block: MSO sets

Let $z= \{z_1, z_2, \ldots z_m\}$ be the set of the constraints which represent the system model and let $x=\{x_1, x_2, \ldots, x_n\}$ be the set of the variables which contains three subsets: let $k \cup y \cup u$ be the set of variables:u is the subset of input variables, y is the subset of the output variables and k is the subset of the unknown (non-measured) variables. The structure of the model is described by the binary relation:

$$M: z \times x \rightarrow \{0, 1\}$$

where: $(z_i, x_j) \rightarrow M(z_i, x_j)=1$ if z_i applies to x_j and $M(z_i, x_j)=0,$ otherwise.

Block 1 obtains the set of MSOs from the system structural matrix $M\ (z, x)$ previously converted to a graph representation. To generate MSO sets, all subsets $M_{\mathrm{MSO}} \subseteq S^+$ of an over-constrained structure graph are calculated which have exactly one constraint more than the just constrained subsystem. The over-constrained part of the structural graph of the system is obtained using the DM (Dulmage Mendelsohn) decomposition [10].To determine all MSO sets in a structure graph the following algorithm is used.

Algorithm for Block 1:

```
$M_{\mathrm{MSO}}$ :=findMSO($M$)
If $\rho(M) = 1$ then
    $M_{\mathrm{MSO}}$: = $\{M\}$;
else
    $M_{\mathrm{MSO}}$: =$\emptyset$;
    for any $ci \in M$
        $M'$:= $(M \setminus \{ci\})^+$;
        $M_{\mathrm{MSO}}$=$M_{\mathrm{MSO}} \cup$ find MSO($M'$);
    end
end if
return $M_{\mathrm{MSO}}$
end
```

Block 2: MSO graph formation:
Input to the Block: Set of MSOs
Output of the Block: MSOGraph

Block 2 obtains the MSO graph from the set of MSOs which is obtained in Block 1. If any variable present in rows of MSO sets makes that particular row a vertex of the graph and all the variable present at same location of two rows is connected edge between the vertexes. The MSO set is the input to Block 2. From MSO set, the vertex and edges of graph are obtained and finally graph $G\ (V, E)$, where V denotes the set of vertices and E is the set of edges, is created which is the output of Block 2. The graph $G\ (V, E)$ can be represented in form of incidence matrix denoted as I_M, which is defined such that

$$I_{Mij}=\begin{cases}-1 \quad if \quad the \quad edge \quad e_j \quad leaves \quad vertex \quad v_i \\ 1 \quad if \quad the \quad edge \quad e_j \quad enters \quad vertex \quad v_i \\ \qquad\qquad 0 \quad otherwise\end{cases}$$

This matrix has dimensions $\phi\times\eta_e$, where ϕ corresponds with the total number of vertices and η_e denotes the total number of edges.

Block 3: Partition of MSO graph

Input to the Block: MSO graph

Output of the Block: Partitioned MSO graph

The maximum weight ω is found for the vertex $v_j \in V$, for j $\in \{1, 2 \ldots \phi\}$. The maximum weight ω for each vertex is equal to number of edges each vertex has. The heaviest vertex is the vertex which has maximum number of edges, the heaviest vertex forms the first subgraph and the centre of the first subgraph G_1 is defined. Those vertices which are connected to this heaviest vertex are included in G_1 or in the first subgraph. The set of non-selected [11] vertices is defined as $V_r = \{v_j \in V{:}v_j \notin V_1\}$.The above procedure is repeated for all vertices $v_j \in V_r$ ($j = \{1, 2, \ldots, \#Vr\}$) until V_r is empty. The subgraph of higher connectivity is highlighted by the above method. The subgraphs which have only one vertex are merged to the closest subgraph and thus a set of subgraphs G_i (V_i, E_i), for i=1, 2, . . ., k, is obtained.

Algorithm for Block 3:

```
$I_M$← System topology
% Start up
$G(V, E)$← $I_M$
    for $j$ = 1 to $\phi$ do
        Compute $\omega_j$
    end for
% Partitioning
$V_r$← $V_i$= 1
repeat
Find $v \in V_r$ with maximum $\omega$
$V_i$← $v$ and all its neighbor vertices
$V_r$=$V - \{ \bigcup_{h=1}^{i} V_h \}$
$i = i + 1$
until $V_r$=Ø
```

Block 4: Local Fault Signature Matrices Formation

Input to the Block: Partitioned MSO graph

Output of the Block: A set of fault signatures matrices, one for each subgraph

MSO sets obtained in Block 1 are constraints that only involve known parameters θ and measured [7] variables (y, u). The sets of MSO are represented as

$$R = \{r_i \,|\, r_i = \Psi_i(y_k, u_k, \theta_k), i = 1, \ldots, n_\tau\}$$

ψ_i is the mathematical expression for MSO sets and n_r is the MSO sets number obtained. Fault diagnosis is done by identifying the set of consistent MSO sets

$$R_0 = \{r_i \,|\, r_i = \Psi_i(y_k, u_k, \theta_k), = 0, i = 1, \ldots, n_\tau\}$$

and inconsistent MSO sets

$$R_1 = \{r_i \,|\, r_i = \Psi_i(y_k, u_k, \theta_k), \neq 0, i = 1, \ldots, n_\tau\}$$

when some inconsistency at time instant k is detected, the process of fault isolation starts by obtaining the observed fault signature, where each single fault signal indicator is defined as follows:

$$\varphi_i(k) = \begin{cases} 0 \quad if \quad r_i(k) \in R_0, \\ 1 \quad if \quad r_i(k) \in R_{1.} \end{cases}$$

Fault isolation is the binary relation between the considered fault hypothesis set $\{f_1(k), f_2(k) \ldots\ldots f_{nf}(k)\}$ and the fault signal indicators $\varphi_i(k)$, stored in the *Fault Signature Matrix F*. The fault hypothesis f_j is expected to affect the residual r_i when F_{ij}, is equal to 1 and in such case the related fault signal $\varphi_i(k)$ is equal to 1. This means that this fault is affecting the monitored system, otherwise, the element $F_{ij}$is zero-valued. A column of this matrix is known as a *theoretical fault signature*. The fault isolation starts by finding a match between the observed fault signatures with some of theoretical fault signatures.

Block 5: Distributed Fault Diagnosis

Input to the Block: Various fault signature matrices of original and observed subsystems each represented with an agent.

Output of the Block: Diagnosed fault in the subsystems

After creating local fault signature matrix for each subsystem, each local fault signature matrix is represented by help of an *agent*. The agent in each local fault signature is responsible for local diagnoses in a subsystem and communicating with other agents of others subsystems to diagnose a fault in a given subsystem. How agents will communicate with each other is shown below in various steps. Suppose there are n subsystems, each represented by an agent named $A_1, A_2, A_{3,} \ldots, A_n$. Suppose MSO_x and a shared variable y of 1st subsystem is faulty and it is shared with subsystem 2nd. The subsystem 1st and 2nd are also having shared variables v, w apart from variable y. Then, the distributed fault diagnosis procedure works as follows:

1: Agent A_1 checks all its *MSO* sets for fault, to detect whether fault is occurred in unshared or shared variables. If the fault has occurred in unshared variables then no need for broadcast. But, if the fault has occurred in a shared variable, then A_1 has to communicate with other agents.

2: Agent A_1 broadcast to agents A_2, A_3,..., A_n that a fault has occurred in MSO_x

3: The broadcast message received by A_2. A_2 checks the residual values of its all *MSO* and find that *MSO* is faulty.

4: A_2 broadcast message to A_1 that a fault has occurred in MSO_z and some of the variables y, v, w can be faulty.

5: The broadcast message received by A_3. A_3 checks the residual values of its all *MSO* and find that none of its MSO_x is faulty.

6: A_3 broadcast message to A_1 that all its *MSO* are fine and no fault has occurred.

Up to n till agent A_1 do not communicate with all other agents of various subsystems the distributed diagnosis process has not ended. The process how the agent of each subsystem communicates with each other to diagnose and isolate a fault or faults in a system is described below

Actually, the distributed diagnosis process consists of three separate algorithms. All the three algorithms are described below.

Algorithm 1: Transmission of messages by various agents

Input: Fault diagnosis matrix for each subsystem represented by an agent.

Output: Transmission message sent by each agent

```
1: TR =Ø
2:int data[$n_r$]
3: for each MSO $Ms_k$
4:if residual value $\neq 0$
5:     for each shared variable $Va_m$
6:     data [$n_r$] = $Va_m$
7:      $n_{r2}$= $n_{r1}$+1;
8:   end for
9:  end if
10: end for
11: TR=Sending shared faulty candidates::: data[$n_r$]
```

Explanation for Algorithm 1

1: For each MSO (Ms_k) sets local diagnoses is done to detect whether any of its MSO is faulty or not by checking the residual value of each MSO. If residual value of a particular MSO is zero than the MSO is not faulty but if the MSO value is not zero, it indicates that the particular MSO is faulty or has fault or faults.

2: After finding out which MSO is affected by the fault, each variable within that particular faulty MSO is checked to find out which variables in given MSO is or are faulty.

3: The faulty variables of a faulty MSO is transmitted in form of message (TR) containing variable number of faulty shared variable (Va_m) or variables as data (data[n_r]) by an agent to rest of the agents.

Algorithm 2: Reception of messages by various agents

Input: Transmission message by each agent.

Output: Received message by each agent

```
1: SM=Ø
2: int RX[$m_p$]
3: for each $A_j$ except $A_i$
4:   for each received data [$n_r$]
5:      ifresidual value $\neq 0$
6:         if$B_{1m}$== data [$n_r$]
7:            RX[$m_{p1}$]== $B_{1m}$
8:            $m_{p2}$= $m_{p1}$ +1;
9:         else RX[$m_{p1}$]== 00
10:           $m_{p2}$= $m_{p1}$ +1;
11:        end if
12:     end if
13:   end for
14:  end for
15: SM=receiving shared faulty      candidates::RX[$m_p$]
```

Explanation for Algorithm 2

1: All agents A_j except the agent A_i, who had sent the message, receive the data (data [n_r]) in form of faulty shared variable or variables number of a particular faulty MSO.

2: After receiving the data each MSO perform local diagnoses to check which MSO is faulty and whether the faulty shared variables received in transmitted message belongs to any faulty variable (B_{1m}) of a non-consistent MSO.

3: If the faulty shared variables received in transmitted message belongs to any non-consistent MSO, same variable number or numbers (RX [m_p]) are received by the sending agent and if the faulty shared variables received in transmitted message do not belongs to any non-consistent MSO, 00 is received by the sending agent.

Algorithm 3: Computation of global diagnosis from local diagnosis by each agent
Input: Inconsistent residuals by each agent.
Output: Global diagnosis

```
1: for each Agent $A_i$ do
          local diagnosis
2: for each Agent $A_i$ do
3:   compute TR
4:   broadcast TR on the network
5: end for
6: for each Agent $A_i$ do
7:    compute SM
8: end for
9: end for
```

Explanation for Algorithm 3
1: For each agent (A_i) local diagnoses aims to detect whether any of its MSO is faulty or not and the faulty shared variable or variables of that particular MSO is broadcasted in form of data message (TR) by the agent in which fault occurs to all other agents. Step 1 of Algorithm 3 will compute entire Algorithm 1.
2: Each agent will send message (SM) to the agent from which they receive the message. Step 2 of Algorithm 3 will compute entire Algorithm 2.
*3:*After receiving messages from all agents, a particular agent will detect and isolate a fault.

IV. Implementation of the Solution

The proposed algorithm is illustrated using Barcelona water network shown in Figure 2 with its various subsystems.

***MSO sets* generation (Block 1):**
The proposed fault diagnosis algorithm starts from discrete-time space state model

$$x(k+1) = A\,x(k) + B_u\,u(k) + B_p d(k)$$

$$y(k) = C\,x(k)$$

where $A \in R^{nxn}$, $B_u \in R^{nxm}$, $C \in R^{rxn}$ are the state space matrices and $B_p \in R^{nxp}$ is the disturbance matrix, $x \in R^n$ is the state vector corresponding to the volume of deposits, $u \in R^m$ is the vector of input variables, $d \in R^p$ corresponds the vector of disturbances (in this case are the water demands) and $y \in R^r$ is the vector of outputs.
This model is the starting point for obtaining the structural matrices applying the algorithmic implementation of Block 1.

As result of Block 1, the MSO sets are obtained (the procedure of obtaining MSO sets described in detail in Section II).

Formation of MSO Graph (Block 2):

Block 2 allows obtaining the MSO graph presented in Figure 3.

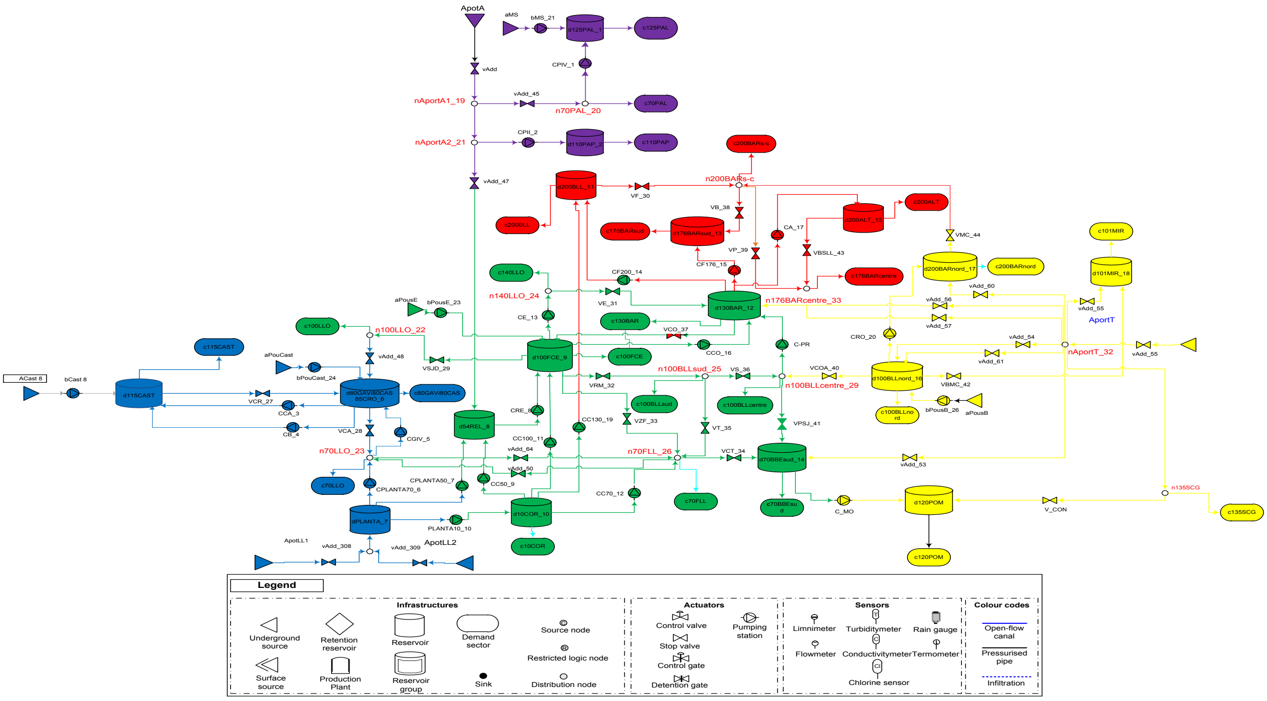

Figure 2. Barcelona water network

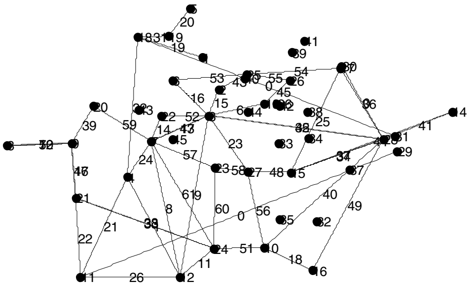

Figure 3. MSO graph

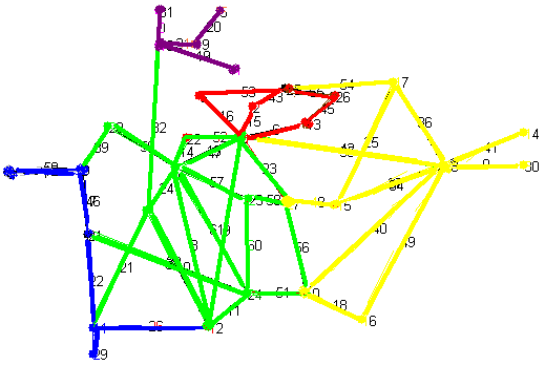

Figure 4.Subgraphs of Barcelona water network

MSO graph partition (Block 3):

Using partition algorithm defined in Block 3, the system or is partitioned into five subgraphs or subsystems shown in (Figure 4). These five subsystems are also shown in Figure 2 of Barcelona water network and their details are shown in Table I. Figure 5 shows various shared variables of Barcelona water network.

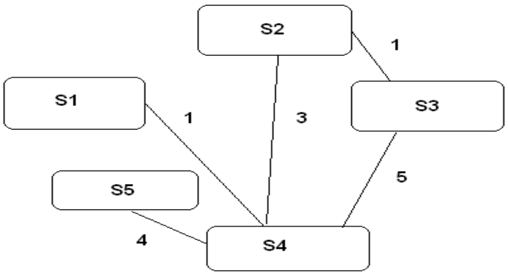

Figure 5. Shared variables of Barcelona water network

Number	Color	# ARRs	# Shared variables
1	purple	4	1
2	red	5	4
3	yellow	7	6
4	green7	1	3
5	blue	5	4

Table I: Barcelona DWN subsystems and number of both MSO and shared elements

Fault Signature Matrices formation (Block 4):
Local fault signature matrices for various subsystems of Barcelona water network shown in Figure 6.1 to 6.5. The rows and columns of various local fault signature matrices consist of MSO sets and unshared as well as shared variables.

Distributed Fault Diagnosis (Block 5):
A fault scenario is considered involving three subsystems (3, 4, 5) that are monitored with three agents A_3, A_4, A_5. Suppose in the subsystem 3, MSO 10 and variable 51 is faulty or violated and similarly in subsystem 4, MSO 24 and variable 51 is faulty and in subsystem 5, no MSO is faulty. So, the 3 agents should communicate with each other to diagnose and isolate the fault using concept of distributed fault diagnosis (local diagnosis with minimum global diagnosis) following traditional FDI approach. This entire process is explained below in point wise manner. Everything happening together and parallel using the FSM tables presented in Figure 6.1 to 6.5:

1. Agent A_3 sends a communication message containing shared variable number 51 as data to agent A_4 and A_5. As soon as agent A_4 and A_5 receives the data 51 from agent A_3, they start local diagnosis by checking all there MSO sets whether any MSO sets has been violated or not. Similarly at the same time A_4 will send a message containing shared variable numbers 33, 38 and 51 to A_3 and A_5

2. In reply of message from A_3, A_4 and A_5 will first perform local diagnoses after that A_5 will send message containing data 00 to A_3, indicating that there is no error or fault. A_4 will send 51 to A_3, indicating that there is a fault in 51. At the same time in reply of the messages from A_4, A_5 will send a message containing data 00 00 00 indication no fault while A_3 will send a message to A_4, containing 00, 00 and 51 indicating a fault in 51.

Step 3. After each agent receives the information, on the basis of this information the agents will detect and isolate the variable in which the fault occurred.

IV. Conclusions

In this paper, a distributed fault diagnosis based on agents has been proposed. First, from a system structural model, MSO sets are obtained. From MSO sets, a vertex and edge graph is generated. This vertex and edge graph is subdivided into various subgraphs through a graph partition algorithm. Each subgraph corresponds to a subsystem. For each subgraph, a local fault signature matrix is generated which contains both local and shared variables of subgraph. Each subsystem and associated sets of MSO is then represented by an agent and the agent in each local fault signature is responsible for local diagnoses in a subsystem and communicating with other agents of others subsystems to diagnose a fault in a given subsystem.
Finally, the proposed approach has been satisfactorily illustrated in the Barcelona water network. As future work the proposed approach will be implemented using a multi-agent system platform.

Acknowledgments

This work has been partially funded by the Spanish Government (MINECO) through the project CICYT ECOCIS (ref. DPI2013-48243-C2-1-R), by MINECO and FEDER through the project CICYT HARCRICS (ref. DPI2014-58104-R).

References

[1] F. Boem, R.M.G. Ferrari, T. Parisini, and M. M. Polycarpou. Distributed fault diagnosis for continuous time nonlinear systems: The input-output case. Annual Reviews in Control, 37(1):163 – 169, 2013.

[2] J. Biteus, E. Frisk, and M. Nyberg. Distributed diagnosis using a condensed representation of diagnoses with application to an automotive vehicle. IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, 41(6):1262–1267, November 2011.

[3] I. Roychoudhury, G. Biswas, and X. Koutsoukos. Designing distributed diagnosers for complex continuous systems. IEEE Transactions on Automation Science and Engineering, 6(2):277–290, April 2009.

[4] C. Ocampo-Martinez, V. Puig, G. Cembrano, R. Creus, and M. Minoves. Improving water management efficiency by using optimization-based control strategies: the Barcelona case study. Water Science & Technology: Water supply, 9(5):565–575, 2009.

[5] C. Ocampo-Martinez, V. Puig, G. Cembrano, and J. Quevedo. Application of predictive control strategies to the management of complex networks in the urban water cycle [applications of control]. IEEE Control Systems Magazine, 33(1):15–41, 2013.

[6] J. Biteus, EC. Ocampo-Martinez, S. Bovo, and V. Puig. Partitioning approach oriented to the decentralised predictive control of large-scale systems. Journal of Process Control, 21(5):775–786, 2011.

[7] S. Tornil-Sin, C. Ocampo-Martinez, V. Puig, and T. Escobet. Robust fault diagnosis of nonlinear systems using interval constraint satisfaction and analytical redundancy relations. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 44(1):18–29, Jan 2014.

[8] Riccardo Ferrari, Distributed Fault Detection and Isolation of Large-scale Nonlinear Systems: an Adaptive Approximation Approach, pages 1- 133, 2007.

[9] Ana Gabriela Guzman. Techniques and Tools for Analysis Systems Large Scale Application to Water Transport Network Barcelona, pages 1-93, 2009.

[10] Mogens Blanke Michel Kinnaert, Jan Lunze, Marcel Staroswiecki. Diagnosis and Fault-Tolerant Control. Springer. 3rd Edition, 2016.

	19	20	31	32
1	1	0	0	0
5	0	1	0	0
18	0	0	1	1
19	0	0	1	0

Figure 6.1 Local fault signature matrix of subsystem 1 diagnosed by agent A_1

	6	15	16	43	45	53	54	55
2	0	1	0	1	0	0	0	0
6	0	0	1	0	0	1	0	0
13	1	0	0	0	1	0	0	0
25	0	0	0	0	0	1	1	1
26	0	0	0	0	1	0	0	1

Figure 6.2 Local fault signature matrix of subsystem 2 diagnosed by agent A_2

	18	23	25	34	35	36	37	40	41	42	44	48	49	51	54	56	58
10	1	0	0	0	0	0	0	1	0	0	0	0	0	**1**	0	**1**	0
14	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0
15	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0	0
16	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
17	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0
27	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
28	0	0	0	1	1	1	1	1	1	1	0	0	1	0	0	0	0

Figure 6.3 Local fault signature matrix of subsystem 3 diagnosed by agent A_3

	6	8	9	10	11	13	14	15	16	23	24	26	32	33	35	38	42	47	51	52	57	58	59	60	61
3	1	0	1	0	0	1	0	1	1	1	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
7	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	1	0	1
12	0	1	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0
24	0	0	0	0	1	0	0	0	0	0	0	0	1	**1**	0	**1**	0	0	**1**	0	0	0	0	1	1

Figure 6.4 Local fault signature matrix of subsystem 4 diagnosed by agent A_4

	7	12	17	21	22	26	33	38	39	46	50	59
8	1	1	0	0	0	0	0	0	0	0	1	0
9	1	1	1	0	0	0	0	0	1	1	1	0
11	0	0	0	1	1	1	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	1	0	0	1
21	0	0	1	0	1	0	**1**	**1**	0	1	0	0

Figure 6.5 Local fault signature matrix of subsystem 5 diagnosed by agent A_5