Olfa Hrizi, Boumedyen Boussaid, Ahmed Zouinkhi, Med Naceur Abdelkrim, Christophe Aubrun

Robust Adaptive Observer based on Multi wheeled Mobile Robot Cooperation Algorithm

Wheeled mobile robots (WMRs) are of great importance. Therefore, it is necessary to make sure that they are not defected. But, in case of failures, the diagnosis task is very important to predict then solve the problem. The most useful techniques in diagnosis are observers which are based on the observability of the monitored system that is not usually ensured by WMR. Thus, to overcome this drawback, an intelligent cooperative diagnosis algorithm is proposed and tested for a group of mobile robots. The diagnosis algorithm is based on robust adaptive unknown input observer applied on unobservable robot. The local non-observability of each robot is solved by cooperative communication. The idea consists on considering all WMRs as a Large Scale System (LSS) even these robots may have not common task. Then, the LSS is decomposed into subsystems that everyone refers to each robot communicating with its neighbors. Next, a design of cooperative interconnected systems is studied to reassure the new condition of observability. Besides, Fast Adaptive Fault Estimation (FAFE) algorithm is proposed to improve the performances of the fault estimation. Finally, to illustrate the efficiency of the proposed algorithm, a model of three-wheel omnidirectional mobile robot is presented.

Key words: Large Scale System (LSS), Cooperative diagnosis, local and global observability, Robust Unknown Input Observer (UIO), multi-robot system, Wheeled Mobile Robot (WMR), Fast Adaptive Fault Estimation (FAFE)

Robusni adaptivni observer temeljen na algoritmu za kooperaciju mobilnih robota s više kotača. Mobilni roboti na kotačima od velike su važnosti. Stoga, nužno je osigurati da ne odlutaju. U slučaju kvara važna je dijagnoza kako bi se predvidio i onda riješio problem. Najkorisnije dijagnostičke tehnike su observeri koji se zasnivaju na osmotrivosti nadgledanih sustava koja kod mobilnih robota na kotačima najčešće nije osigurana. Stoga, kako bi se nadišao ovaj problem, koristi se inteligentan algoritam za kooperativnu dijagnozu i testira se na grupi mobilnih robota. Dijagnostički algoritam zasniva se na robusnom adaptivnom observeru s nepoznatim ulazom koji je primijenjen na neosmotrivom robotu. Lokalna neosmotrivost svakog robota riješena je koopreativnom komunikacijom. Ideja je da se svi mobilni roboti promatraju kao sustav velikih razmjera iako roboti nemaju isti zadatak. Sustav velikih razmjera se tada rastavlja na podsutave tako da se svaki odnosi na jednog robota koji komunicira sa svojim susjedima. Zatim se proučava dizajn kooperativnih povezanih sustava kako bi se osigurali uvjeti za osmotrivost. Dodatno, predlaže se korištenje brze adaptivne estimacije pogreške kako bi se poboljšala estimacije pogreške. Konačno, prikazan je model višesmjernog mobilnog robota na tri kotača kako bi se ilustrirala učinkovitost predloženog algoritma.

Ključne riječi: sustavi velikih razmjera, kooperativna dijagnoza, lokalna i globalna osmotrivost, robusni observer s nepoznatim ulazom, višerobotski sustav, mobilni robot na kotačima, brza adaptivna estimacija pogreške

1 INTRODUCTION

Generally, large scale systems are naturally divided into many smaller interacting subsystems. A class of them is usually controlled by a distributed or decentralized control framework. The interaction between each subsystem and some neighboring subsystems is getting by their states and inputs. The technical goal is to achieve a specific global performance of the entire system. When the number of input and output variables for a class of large scale system is up to a hundred (or thousand), since centralized control is forbidden for the less flexibility and the large cost of computation, the distributed framework is usually adopted regardless of losing global performance into interconnected subsystems [1, 2]. Each subsystem is controlled

by a subsystem-based controller and these controllers are interconnected via network. Without loss of generality, there are many tools of large scale system decomposition. The choice of the manner used depends to domain of work and the goal to obtain such as mobile robotics. At this level, it must be noted that in the last several years, considerable attention has been focused on the emerging field of robotics. Therefore, many successful robotic manipulator designs have been introduced thanks to their good terrain adaptability and high mobility. Many researchers have proposed different ways of control robot design, classified types and ensure performance of robots [3, 4]. A large effort has been devoted by the scientific community especially to the field of mobile robot systems [5]. In particular, wheeled robots will be expected to provide many convenient and user friendly transport solutions for both people and objects [6]. The importance of the wheeled mobile robots has long been recognized by the robotic research community. In addition to that, coordination in robotic networks becomes an important and promising research area [7, 8]. There are many coordination tasks of multiple mobile robots including formation, coverage, and target aggregation, but the basic idea for the collaborate control design is to make the relative position and orientation of the robots in a desired formation (may be within a given target set) or all the robots move effectively as a whole. However, as more detail is included, the dimensionality of such simulations may increase to unmanageable levels of storage and computational requirements. One approach to overcoming this is through model reduction. The goal is to produce a low dimensional system that has the same response characteristics as the original system with far less storage requirements and much lower evaluation time. The resulting reduced model might be used to replace the original system as a component in a larger simulation or it might be used to develop a low dimensional controller suitable for real time applications. On the other hand, there are diverse objectives to decompose the multi-agent system. In this work, the main contribution is to diagnose a system of multi-robots based on fast adaptive observer. In other words, thanks to robot cooperation, the condition of system observability can be more overcame to ensure the diagnosis system. This is the principle of cooperative diagnosis. The last theme "diagnosis" is considered the focus of several researches [9]. This procedure, Fault Detection and Isolation (FDI), consists in detecting and isolating faults in a physical system by monitoring its inputs and outputs [10]. A typical system for fault detection and isolation is made of three parts: fault detection indicates that there is a mistake in the operating system, i.e., the occurrence of a fault and the time of the fault occurrence; secondly, fault isolation determines the location and the type of the fault and finally, fault identification determines the size of the fault. Diverse

FDI methods have been reported in the literature such as: generating redundancy in the case of physical redundancy between sensors or parity space formulation. After FDI block [11], Fault tolerant control systems are needed in order to maintain the performance objectives [12], or if that turns out to be impossible, to assign achievable objectives so as to avoid catastrophic failures [13]. Since the process used in this work is from the family of mobile robot, it is better to refer to some researches that are interested by both diagnosis term [14, 15] and robotic theme [16]. The rest of this paper is organized as follow: firstly, a decomposition of large scale system is presented. A robot neighborhood is then discussed to ensure the cooperation between robots, its neighbors and some beacons. This is followed by a summary of observability design and robust estimation algorithm. After that, simulations results are given to illustrate the effectiveness of the proposed contribution using the model of three-wheel omnidirectional mobile robot. Finally, this manuscript is ended by concluding remarks and a discussion of future work.

2 LARGE SCALE SYSTEM

2.1 Presentation

In the study of LSS, the development of decomposition techniques is considered one of the major concerns. Decomposition of a LSS into interconnection of lower-dimensional subsystems is instrumental to the analysis, estimation and control of LSS [17]. While various decomposition methods have been developed in the past research, the computational aspects of the associated numerical algorithms are not yet fully explored. It is not necessarily desirable to deal with each component of a system in a uniformed way, since there exist a limitation in computational amount and available information about the systems. Hence, if the system of our interest is highly large-scale and complex, we should create an innovation which reduces the amount within the range which does not lose accuracy. One of the ways to solve difficulties is to divide a large-scale system into some hierarchical layers by corresponding to its physical scale, and to give a comprehensive notion of a dynamical system modeling.

2.2 Decomposition

The main goal of LSS decomposition in this context is to create an interface of systems with reduced size communicating with each other. At this step, we introduce preliminaries for the model used. Consider a global LSS given as continuous-time linear dynamical model described by the state-space equation:

$$(S) \begin{cases} \dot{X} = AX + BU \\ Y = CX \end{cases} \tag{1}$$

where $X \in R^{n_x}$ is the state of system (S), $U \in R^{n_u}$ is the input of system and $Y \in R^{n_y}$ is the output vector.

Suppose that (S) consists of k subsystems $S_i, i = 1...k$ where $n_x = \sum_{i=1}^{k} n_i$, $n_u = \sum_{i=1}^{k} m_i$, $n_y = \sum_{i=1}^{k} p_i$ and n_i, m_i, p_i are the dimensions of the state, input and output vectors of the subsystem. This leads to the next scripture of system.

$$(S) \begin{cases} \dot{X} = \begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_i \\ \vdots \\ \dot{x}_k \end{bmatrix} = A \begin{bmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_k \end{bmatrix} + B \begin{bmatrix} u_1 \\ \vdots \\ u_i \\ \vdots \\ u_k \end{bmatrix} \\ Y = \begin{bmatrix} y_1 \\ \vdots \\ y_i \\ \vdots \\ y_k \end{bmatrix} = C \begin{bmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_k \end{bmatrix} \end{cases} \quad (2)$$

where $A = \begin{bmatrix} A_{11} & \cdots & A_{1i} & \cdots & A_{1k} \\ \vdots & \ddots & & & \vdots \\ A_{i1} & \cdots & A_{ii} & \cdots & A_{ik} \\ \vdots & & & \ddots & \vdots \\ A_{k1} & \cdots & A_{ki} & \cdots & A_{kk} \end{bmatrix}$,

$B = \begin{bmatrix} B_{11} & \cdots & B_{1i} & \cdots & B_{1k} \\ \vdots & \ddots & & & \vdots \\ B_{i1} & \cdots & B_{ii} & \cdots & B_{ik} \\ \vdots & & & \ddots & \vdots \\ B_{k1} & \cdots & B_{ki} & \cdots & B_{kk} \end{bmatrix}$

and $C = \begin{bmatrix} C_{11} & \cdots & C_{1i} & \cdots & C_{1k} \\ \vdots & \ddots & & & \vdots \\ C_{i1} & \cdots & C_{ii} & \cdots & C_{ik} \\ \vdots & & & \ddots & \vdots \\ C_{k1} & \cdots & C_{ki} & \cdots & C_{kk} \end{bmatrix}$

Therefore, from (2) the global system can be decomposed into subsystems described by the following state-space (S_i):

$$\begin{cases} \dot{x}_i(t) = A_{ii}x_i(t) + B_{ii}u_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^{k} A_{ij}x_j + \sum_{\substack{j=1 \\ j \neq i}}^{k} B_{ij}u_j \\ y_i(t) = C_{ii}x_i(t) \end{cases}$$
$$(3)$$

where $x_i(t) \in \Re^{n_i}$ is the state vector of (S_i), $u_i(t) \in \Re^{m_i}$ is the input vector and $y_i(t) \in \Re^{p_i}$ is the output vector.

A_i, B_i are variable parameter matrices of dimensions $(n_i \times n_i)$, $(n_i \times m_i)$ and C_i is known matrix of dimension $(p_i \times n_i)$. A_{ij} and B_{ij} are the interaction matrices between two neighbor subsystems (S_i) and (S_j).

3 PROBLEM FORMULATION

As it is known from the literature that there is some problem to ensure the observability of LSS. So, the main purpose of this work is to expand the linear system's observability condition thanks to the new algorithm of cooperation between mobile robots.

That's why, in the following, we will propose an hierarchical procedure with diverse steps to solve at the end the problem in this kind of system (LSS).

The context for this strategy is introduced by briefly discussing the observability of WMR and then considering the new decomposition of LSS that might happen to ensure the observability theory and therefore the observer existence. Note that in the following A_{ii}, B_{ii} and C_{ii} will be noted as A_i, B_i and C_i respectively.

3.1 Observability Analysis

Considering that there is neither mechanical connection between robots nor common task between neighbor subsystems. So, the matrices A_{ij} and B_{ij} will be neglected.

Therefore, the subsystem model given in (3) will be written as follow:

$$(S_i) \begin{cases} \dot{x}_i(t) = A_i x_i(t) + B_i u_i(t) + K_i w_i(t) \\ y_i(t) = C_i x_i(t) \end{cases} \quad (4)$$

where $w_i(t) \in \Re^{l_i}$ is the unknown input vector which contains faults vector and/or disturbances vector. K_i is a known input matrix of dimension $(n_i \times l_i)$.

Note that in practice the WMR used is unobservable. So, $rank(O_i) = r_i \prec n_i$, where

$$O_i = \begin{pmatrix} C_i & C_i A_i & C_i A_i^2 & \dots & C_i A_i^{n_i - 1} \end{pmatrix}^T$$

is the observability matrix. Thus, the design of Unknown Input Observer (UIO) is not possible. The next section will propose a solution for this problem.

3.2 Subsystem Decomposition

If we suppose that $rank(O_i) = r_i < n_i$, the WMR number i is not observable. But the main goal is to synthesize an adaptive observer for this system. That's why; it shall rewrite the model given (4) into a hierarchical form, by dissociating the observable part and unobservable one. Otherwise, to ensure observability of (S_i), it must have $rank(A_i) = r_i$. Hence, this system will be decomposed

into an observable part with dimension r_i to extract the measurable states and a unobservable one, it will be written in its decentralized form (S_i) as follow:

$$
\begin{cases}
\dot{x}_i(t) = A_i \begin{bmatrix} x_{i1}(t) \\ x_{i2}(t) \end{bmatrix} + \begin{bmatrix} B_{i1} \\ B_{i2} \end{bmatrix} u_i(t) + \begin{bmatrix} K_{i1} \\ K_{i2} \end{bmatrix} w_i(t) \\
y_i(t) = \begin{bmatrix} C_1 & 0 \end{bmatrix} \begin{bmatrix} x_{i1}(t) \\ x_{i2}(t) \end{bmatrix}
\end{cases}
\tag{5}
$$

where $A_i = \begin{pmatrix} A_{i11} & A_{i12} \\ A_{i21} & A_{i22} \end{pmatrix}$

Observable part

The expression of observable part of (S_i) is written as follow (S_{i1}):

$$
\begin{cases}
\dot{x}_{i1}(t) = \begin{pmatrix} A_{i11} & A_{i12} \end{pmatrix} \begin{bmatrix} x_{i1}(t) \\ x_{i2}(t) \end{bmatrix} + B_{i1} u_i(t) + K_{i1} w_i(t) \\
y_i(t) = C_1 x_{i1}(t)
\end{cases}
\tag{6}
$$

where $x_{i2}(t)$, $u_i(t)$ and $w_i(t)$ are considered unknown inputs.

Hence, the new expression (S_{i1}) is given by (7):

$$
\begin{cases}
\dot{x}_{i1}(t) = A_{i11}(x) x_{i1}(t) + B_{c1} \begin{bmatrix} x_{i2}(t) \\ u_i(t) \\ w_i(t) \end{bmatrix} \\
y_i(t) = C_{i1} x_{i1}(t)
\end{cases}
\tag{7}
$$

where $x_{i1}(t) \in \Re^{r_i}$ and $B_{c1} = \begin{bmatrix} A_{i12}(x) & B_{i1}(x) & K_{i1} \end{bmatrix}$.

Suppose that O_{i1} is the observability matrix of (S_{i1}):

$O_{i1} = \begin{pmatrix} C_1 & C_1 A_{i11} & C_1 A_{i11}^2 & \dots & C_1 A_{i11}^{n_i-1} \end{pmatrix}^T$

Now, it's clear that $rank(O_{i1}) = r_i$.

Thus, (S_{i1}) is observable and the Observer design is possible. As result, a robust Unknown input observer analysis which is developed in my previous work [18] can be applicable.

Remark 1 *It must permute the rows of A_i, B_i, C_i to have the first vector as observable state if they are not in order.*

Unobservable part

The unobservable part (S_{i2}) can be described by (8)

$$
\begin{cases}
\dot{x}_{i2}(t) = A_{i22}(x) x_{i2}(t) + B_{c2} \begin{bmatrix} x_{i1}(t) \\ u_i(t) \\ w_i(t) \end{bmatrix} \\
y_i(t) = C_2 x_{i2}(t)
\end{cases}
\tag{8}
$$

where $B_{c2} = \begin{bmatrix} A_{i21}(x) & B_{i2}(x) & K_{i2} \end{bmatrix}$, $x_{i2}(t) \in \Re^{n_i-r_i}$ Given that (S_{i2}) is not observable, the observer design can be realizable if all states of (S_{i2}) are measurable.

So, to solve this problem, we try to estimate x_{i2} from cooperation with other neighbor subsystems. This is the subject of Fig. 1.
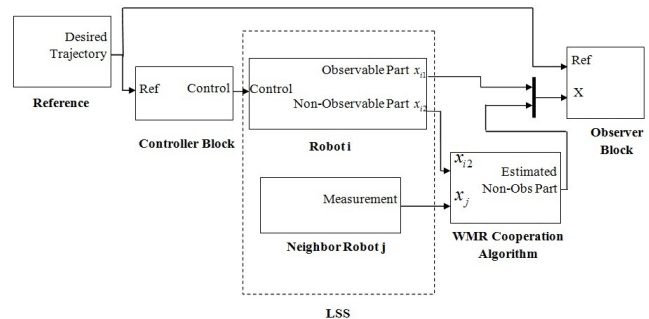

Fig. 1. Observability Synthesis based on Cooperation Algorithm.

4 MULTI WMR COOPERATION

From this section, we will propose a solution of non-observability problem based on system's cooperation. Multi-Robot Systems (MRSs) are employed for diverse reasons; however, one of the main motivations is that multi-robot systems can be used to increase the system effectiveness [19]. To ensure the MRSs cooperation, it must validate the robot neighborhood test developed in the following section.

4.1 Robot Neighborhood Test

If there is no obstacle, two robots i and j, equipped on the same communication range $range_c$, are considered neighbors means could communicate reciprocally if and only if $d_{ij} \leqslant range\_c$, where d_{ij} is the Euclidian distance between two robots.

Otherwise, in presence of obstacles, this test is insufficient. It must send signals between all robots to check if there are answers. Hence, the neighborhood test is verified.

4.2 Cooperation principle

In this work, the MRSs cooperation is treated for the diagnosis systems. The word cooperating underlines the interaction or the integration among multiple robots, that means, the robots have to communicate, exchange information or interact in some way to achieve an overall mission as illustrated in Fig.2.

The abbreviations MR, NR and B mean respectively Master Robot, Neighbor Robot and Beacon. N_i and N_j are the Neighborhood of Robot i and j, respectively.

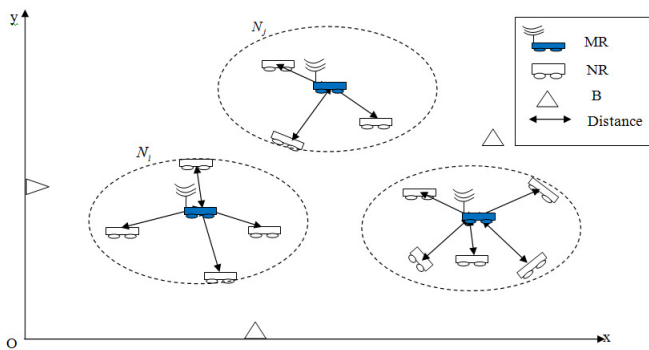To implement the WMR cooperation algorithm with successful result, it must follow the next theorem.

Fig. 2. Illustration of cooperation between team of robots.

Hypothesis 1 : *Suppose a LSS (S) is globally observable.*
$S = S_i \cup S_j$.
There exists a subsystem (S_i) given as previous model described by (4) such as: (S_i) is locally unobservable.

Hypothesis 2 : *Suppose that*
$S_i = S_{i1} \cup S_{i2}$ and $x_i(t) = \begin{bmatrix} x_{i1}(t) \\ x_{i2}(t) \end{bmatrix}$
where $x_{i1}(t) \in \mathbb{R}^{r_i}$, $x_{i2}(t) \in \mathbb{R}^{n_i - r_i}$.
Note that (S_{i1}) is the observable part of (S_i).

Theorem 1 *Let Hypothesis 1 and 2 be satisfied, if there exist a function $f_i(x_{i2}, x_j)$ that described interconnection parameter of neighboring systems such as $x_j \in S_j$ with S_i, S_j are neighbors, x_{i2} becomes measurable.*
Then, (S_i) turns into an observable system locally.

In the following, with more details the proof of the last theorem is given.
Proof: :

Then, a detailed algorithm is proposed to explain the theory of robot cooperation. It must firstly satisfy the following remarks.

Remark 2 : *Suppose that the unknown variables in state vector are robot position (x_i, y_i). Therefore, the number of unknown variables for each subsystem i (robot i) is two.*

Remark 3 : *In this work, interconnection parameters can be summarized in the calculation of distance between robots. While this computation is written based on square of position robot, we have therefore more than one solution. This means that each equation admits two solutions. To reduce the system to unique solution, it must consider 2 more constraints for each one subsystem.*

The problem is formulated by these statistics:
Set of n Master Robots, p Beacons and q Neighbor Robots. Firstly, referring to remark 2 and 3, there are $(4 \times n)$ unknown variables according to number of MR. In the neighborhood N_i, distances between the relative master robot MR_i and p beacons provide p equations. Then, there are $(n-1)$ equations derived from distances between MR_i and MR_j, where MR_j is the master robot of other neighborhood N_j. As conclusion, to solve the problem, it must respect the following condition because the number of equations must be equal or more than the number of unknown variables:

$$p + n - 1 \geqslant 4n$$

meaning,
$$p \geqslant 3n + 1 \qquad (9)$$

That is to say for one MR $(n = 1)$, it must provide at least 4 beacons. To affirm this condition, if there are p beacons which are linked by other Master Robot MR_j in addition to MR_i, the number of equations increases to $(p + (n-1)p)$.
If we give also the relations derived from distances between MR_i and each MR_j, there are $((n-1) \times n)$ relations. So to conclude, by testing the number of unknown variables against the equations that can be deduced, it must respect the following inequality:

$$p + (n-1)p + (n-1)n \geqslant 4n$$

which implies
$$n \geqslant \frac{1}{p - 3} \qquad (10)$$

Hence, $(p > 3)$. So, it's clear that for one MR, we need at least 4 beacons. Therefore, generally speaking, for n MR, there are:
$C_n^2 = \frac{n!}{2!(n-2)!}$ equations explained by distances between MR_i and MR_j. Furthermore, following the two previous remarks, we have $(4n)$ unknown variables.
However, this is cannot solve the problem because

$$\frac{n!}{2!\,(n-2)!} = \frac{n(n-1)}{2} \prec 4n$$

That's why; a number of beacons is discussed to satisfy the suitable condition. Consider the distances between p beacons and all MR. So, the number of equations increases to have:
$\frac{n(n-1)}{2} + np \geqslant 4n$ means $n \geqslant 9 - 2p$
which is affirmed by the initial proposition means for 4 beacons, we have 1 MR.
Otherwise, in each neighborhood, there are more connections, if we consider firstly the distances between MR and their neighbors (q equations), then secondly, the distances

between q neighbor robots each other. So, the number of equations increases to: $q + C_q^2$.

It should be noted also that there are $(q+1)$ unknown variables: the MR and q NR. Therefore, it's necessary to respect the following condition:

$q + \frac{q!}{2!(q-2)!} \geqslant 4(q+1)$, this implies that $q \geqslant 8$ ■

Remark 4 *: In practice, in the case of several neighborhoods, the number of relations between master robot and its neighbors increases. Therefore, the system of equations to be solved will be larger. Hence, the delay resulting from multi-robots cooperation algorithm is more significant. To avoid this problem, the number of neighbor robots should be reduced to q_{min}, where q_{min} satisfies the relation which leads to resolve the equation's system for n and p fixed.*

To recap, we will propose a detailed algorithm to explain the theory of robot cooperation.

Note that n, p and q are respectively the number of MR, B and NR. u means the number of unknown variables and e is the number of equations derived from distances between MR_i, MR_j and B.

After satisfying the mobile robots cooperation algorithm, the observability of the overall system is guaranteed. Then, it is possible now to synthesize an adaptive observer to ensure the detection of fault occurred.

5 COOPERATIVE DIAGNOSIS: APPLICATION ON THREE-WHEEL OMNI-DIRECTIONAL MOBILE ROBOT: THEORETICAL RESULTS

5.1 Omni-directional Robot Model

The current section describes the model of three-wheel Omni-directional robot: Consider the WMR in the pose shown in Fig.3.
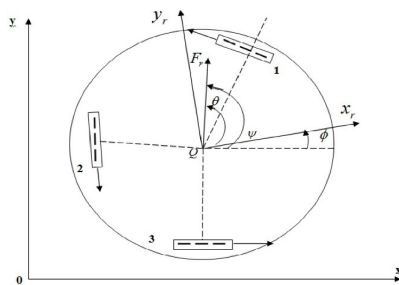

Fig. 3. Geometry of the three-wheel omnidirectional WMR.

By following the calculation given in [20], we obtain the state space dynamic model of the WMR's motion:

$$(S) \begin{cases} \dot{x} = A(x)x + B(x)u \\ y = Cx \end{cases} \qquad (11)$$

Algorithm 1 Multi WMR Cooperation Algorithm

Input: n, q, p
Output: $u, e = f(x, y) = d(MR, NR, B)$
1. **if** (there are n MR) **then**
2. $(u = 4 \times n)$ */test loop
3. **end if**
   (Goal: Seek to find the enough number of equations for this problem)

Case $(n = 1)$ */ In the same Neighborhood N_i

---

4. **if** $(p \succeq 4)$ **then**
5. $(u = 4)$;
6. $e = d(MR, B)$ */fix the number of equations and unknown variables from n and p
7. **end if**
8. **if** $(u \preceq e)$ **then**
9.   **return** ('Problem Resolved')
10. **else**
11.   (Fix q NR)
12.   **if** $(q \succeq 8)$ **then**
13. $(u = q+1)$;
14. $e = d(MR, NR) = (q + C_q^2)$
15.   **end if**
16. **end if**
17. **if** $u \preceq e$ **then**
18.   **return** ('Problem Resolved')

Case $(n \succ 1)$ */ There is cooperation with another neighborhood N_j

---

19. $e = d(MR_i, MR_j) + d(MR_{1..n}, B_{1..p}) = C_n^2 + n \times p$;
20.   **if** $(n \succeq 9 - 2 \times p)$ **then**
21.   **return** ('Problem Resolved')
22.   **end if**
23. **end if**

---

where, $x = \begin{bmatrix} x_Q & y_Q & \phi & \dot{x}_Q & \dot{y}_Q & \dot{\phi} \end{bmatrix}^T$, $y = \begin{bmatrix} \dot{x}_Q & \dot{y}_Q & \phi \end{bmatrix}^T$ and $u = \begin{bmatrix} \tau_1 & \tau_2 & \tau_3 \end{bmatrix}^T$

are respectively the state vector, the output vector and the input one.

Then, the variable parameter matrices are given by A and B. C is the constant known matrix:

$$A(x) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & s_1 & -s_2\dot{\phi} & 0 \\ 0 & 0 & 0 & s_2\dot{\phi} & s_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & s_3 \end{pmatrix},$$

$$B(x) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ v_1\beta_1 & v_1\beta_2 & 2v_1\cos\phi \\ v_1\beta_3 & v_1\beta_4 & 2v_1\sin\phi \\ v_2 & v_2 & v_2 \end{pmatrix},$$

$$C = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}.$$

where the constant parameters of matrices are: $s_1 = -3\beta(3I_0 + 2mr^2)$, $s_2 = 3I_0/(3I_0 + 2mr^2)$, $s_3 = -3\beta D^2/(3I_0D^2 + I_Qr^2)$, $v_1 = Kr/(3I_0 + 2mr^2)$, $v_2 = Kr/(3I_0D^2 + I_Qr^2)$
and the variable elements are: $\beta_1 = -\sqrt{3}\sin\phi - \cos\phi$, $\beta_2 = \sqrt{3}\sin\phi - \cos\phi$
$\beta_3 = \sqrt{3}\cos\phi - \sin\phi$, $\beta_4 = -\sqrt{3}\cos\phi - \sin\phi$

Let define following notations relating to robot parameters: m is the robot's mass, r is the common radius of the wheels, D is the distance of the wheels from the rotation point Q, β is the linear friction coefficient, K is the driving torque gain, I_0 is the common moment of inertia of the wheels and I_Q is the driving input torque of wheel Q.

The observability matrix of this system is given as follow: $O = \begin{pmatrix} C & CA & CA^2 & CA^3 & CA^4 & CA^5 \end{pmatrix}^T$, $rank(O) = 4 \prec n_x = 6$

Therefore, the system is unobservable. At this level, we need to decompose it into two parts: a subsystem observable locally and another unobservable locally also.

5.2 Decomposition of Omni-directional WMR

The system is divided into two parts as explained in the theory described above. Firstly, suppose that the state vector is: $x_1 = \begin{bmatrix} \phi & \dot{x}_Q & \dot{y}_Q & \dot{\phi} \end{bmatrix}^T$
So, referring to remark1 given previously , the first subsystem (S_1) is written as follow:

$$\begin{cases} \dot{x}_1(t) = A_{11}(x)x_1(t) + \begin{bmatrix} A_{12}(x) & B_1(x) \end{bmatrix} \begin{bmatrix} x_2(t) \\ u(t) \end{bmatrix} \\ y(t) = C_1x_1(t) \end{cases}$$

(12)

where the system's matrices are:
$$A_{11}(x) = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & s_1 & -s_2\dot{\phi} & 0 \\ 0 & s_2\dot{\phi} & s_1 & 0 \\ 0 & 0 & 0 & s_3 \end{pmatrix},$$

$$B_1(x) = \begin{pmatrix} 0 & 0 & 0 \\ v_1\beta_1 & v_1\beta_2 & 2v_1\cos\phi \\ v_1\beta_3 & v_1\beta_4 & 2v_1\sin\phi \\ v_2 & v_2 & v_2 \end{pmatrix}$$

Knowing that the second state vector is $x_2 = \begin{bmatrix} x_Q & y_Q \end{bmatrix}^T$, and the matrix A_{12} given as a null matrix. Verifying now the local observability of this subsystem: $O_1 = \begin{pmatrix} C_1 & C_1A_{11} & C_1A_{11}^2 & C_1A_{11}^3 \end{pmatrix}^T$, $rank(O_1) = 4 = n_{x1}$.

Thus (S_1) is observable. Automatically, the synthesis of observer is possible now.

However, in the second part, the state vector x_2 is unmeasurable and (S_2) is locally unobservable subsystem. At this level, we must estimate the robot position (x_Q, y_Q) using the interconnection links with other neighbor robots. Hence, the principle of cooperative diagnosis is clear.

5.3 Cooperation between WMR

So, after reassuring the neighborhood of robots, the objective of this work is to present in a cohesive way the principle of cooperation between mobile robots and its importance to ensure the system's observability. Consider the unobservable part; we try to make its state vector measurable. The problem can be reformulated to solve a system of equations with a number of unknown variables. In other words, it is intended to localize the cooperated mobile robot for attaching the robot parameters means the system's states. Referring to remark 2 and 3 in the theorem 1, it's clear that the number of unknown variables is (4×1). Therefore, it must provide at least 4 equations to solve the problem.

Proposition 1 *For one Master Robot, we can ensure 4 beacons. But, this solution is undesirable because it requires a large number of beacons that is not always available. Also, it must be noted, in other configurations, that the large number of beacons used can increase the delay resulted by cooperation algorithm computational complexity. So, this can destabilize the system. In this case, it should fix a q_{min} neighbor robots. That's why, we will move on to a second proposal.*

Proposition 2 *If we consider 2 other robots means 2 neighborhoods: $n = 3$. Referring to the organizational structure of cooperation algorithm, it remains to discuss the number of beacons used according to the following condition: $n \geqslant 9 - 2p$. Therefore, it must fix $p = 3$.*

The last proposition is resumed by Fig.4.
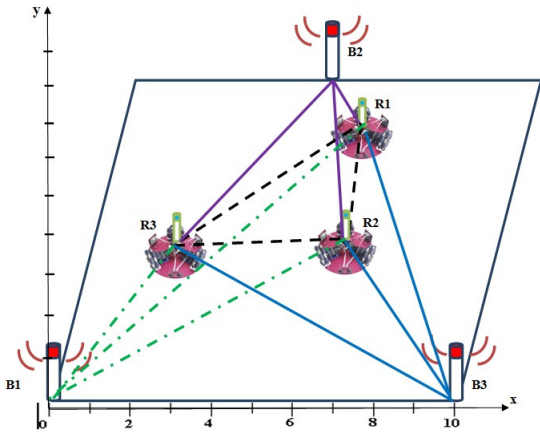
Consider now B1, B2 and B3 three beacons whose

Fig. 4. Diverse possible connections between WMR.

known positions are respectively (x_{b1}, y_{b1}), (x_{b2}, y_{b2}) and (x_{b3}, y_{b3}). Then, there are two robots R2 and R3 which are neighbors to R1. In this proposition, by applying the previous theory, there are $(np + C_n^2)$ equations with $(4n)$ unknown variables. On the other hand, to determine the number of equations which is derived from distances between robots and beacons, it must apply the formula given in the previous algorithm: $3 \times 3 + C_3^2 = 9 + \frac{3!}{1! \times 2!} = 12$ equations.

$$
\begin{cases}
d_{r1r2}^2 = (x_2 - x_Q)^2 + (y_2 - y_Q)^2 \\
d_{r1r3}^2 = (x_3 - x_Q)^2 + (y_3 - y_Q)^2 \\
d_{r2r3}^2 = (x_3 - x_2)^2 + (y_3 - y_2)^2 \\
d_{b1r1}^2 = (x_Q - x_{b1})^2 + (y_Q - y_{b1})^2 \\
d_{b1r2}^2 = (x_2 - x_{b1})^2 + (y_2 - y_{b1})^2 \\
d_{b1r3}^2 = (x_3 - x_{b1})^2 + (y_3 - y_{b1})^2 \\
d_{b2r1}^2 = (x_Q - x_{b2})^2 + (y_Q - y_{b2})^2 \\
d_{b2r2}^2 = (x_2 - x_{b2})^2 + (y_2 - y_{b2})^2 \\
d_{b2r3}^2 = (x_3 - x_{b2})^2 + (y_3 - y_{b2})^2 \\
d_{b3r1}^2 = (x_Q - x_{b3})^2 + (y_Q - y_{b3})^2 \\
d_{b3r2}^2 = (x_2 - x_{b3})^2 + (y_2 - y_{b3})^2 \\
d_{b3r3}^2 = (x_3 - x_{b3})^2 + (y_3 - y_{b3})^2
\end{cases}
\tag{13}
$$

The unknowns' vector is $\begin{bmatrix} x_Q & y_Q & x_2 & y_2 & x_3 & y_3 \end{bmatrix}^T$ which will be ob-

tained by resolving this system. We try to find the suitable solution (x_Q, y_Q) verifying the distances given.

d_{ij} is supposed as an Euclidian distance between two systems i and j. To guarantee the robustness of system resolution, note that it will be considered for two equations derived from d_{ij} and d_{ji} the average value $\overline{d_{ij}} = \frac{d_{ij} + d_{ji}}{2}$. First, we write the 12 functions to resolve as this form:

$$f_{s=1:12}(x_i, x_j, y_i, y_j) = d_{ij}^2 - (x_j - x_i)^2 - (y_j - y_i)^2$$

Then, computing the desired solution is based on Newton Raphson Algorithm (see Appendix)
On this level, the observability condition becomes ensured. Therefore, the observer synthesis is available now to estimate the state vector.

5.4 UIO Design

To develop our work, a fault will occur in the input channel and an unknown disturbance will be taken. Let define the global model system given as follow:

$$
\begin{cases}
\dot{x}(t) = A(x)x(t) + B(x)u_f(t) + D_x d(t) \\
y(t) = Cx(t)
\end{cases}
\tag{14}
$$

Assuming that the fault already occurred may be modeled by an additive term in (14), if we suppose:

$$u_f(t) = u(t) + u_0(t), u_0(t) = f(t)$$

$F_x = B(x)$ where F_x is a variable matrix with dimension $(n \times r)$ and $f(t) \in \Re^r$ is considered as an actuator fault. Therefore, the previous model can be written as follow

$$
\begin{cases}
\dot{x}(t) = A(x)x(t) + B(x)u(t) + F_x(x)f(t) + D_x d(t) \\
y(t) = Cx(t)
\end{cases}
\tag{15}
$$

where D_x is the known disturbance matrix with dimension $n \times q$. Therefore, the Unknown input observer's expression adopted in this work is given with a linear transformation as follow:

$$
\begin{cases}
\dot{z}(t) = Mz(t) + Nu(t) + P_y y(t) + EF_x \hat{f}(t) \\
\hat{x}(t) = z(t) - L_y y(t)
\end{cases}
\tag{16}
$$

Where M, N, P_y, L_y are matrices that will be designed such that the unknown input will be decoupled from other inputs. $z \in \Re^{n \times 1}$ is the state of UIO, obtained by the linear transformation $z = Ex$ and $\hat{x}$ is the estimated state vector. The state estimation error is defined by:

$$e_x(t) = \hat{x}(t) - x(t) = z(t) - L_y y(t) - x(t)$$
$$e_x(t) = z(t) - (I + L_y C)x(t) = z(t) - Ex(t)$$

Let's calculate the UIO matrices: If M is Hurwitz matrix and the following relationships are true:

$$ME + P_yC = EA$$
$$N = EB \qquad (17)$$
$$ED_x = 0$$

Then, it's clear that we can obtain the UIO parameters, and thereafter, the state error. The next stage is the fault estimation part.

5.5 Fast Adaptive Fault Estimation Algorithm

To implement this algorithm with successful result, it must follow some assumption and lemma which is given also to verify the linear matrix inequality LMI.

Assumption 1 $rank(CD_x) = q$

Lemma 1 *[21] Given a scalar $\mu \succ 0$ and a symmetric positive definite matrix P which justify the following inequality:*

$$2x^Ty \leqslant \frac{1}{\mu}x^TPx + \mu y^TP^{-1}y \qquad (18)$$

The FAFE algorithm is proposed to ameliorate performances of time varying actuator fault estimation: rapidity, stability and accuracy.

Theorem 2 *Under Assumption and conditions (17) verified, given scalars $\sigma, \mu \succ 0$, if there exist symmetric positive definite matrices $P \in \Re^{n\times n}, G \in \Re^{r\times r}$ and matrices $M \in \Re^{n\times n}, F \in \Re^{r\times p}$ such that*

$$F_x^TP = FC \qquad (19)$$

and the following condition hold,

$$\begin{pmatrix} M^TP + PM & 2PEF_x \\ -2\frac{1}{\sigma}F_x^TPM - 2F_x^TP & -2\frac{1}{\sigma}F_x^TPEF_x + \frac{1}{\sigma\mu}G \end{pmatrix} \prec 0 \qquad (20)$$

then the FAFE algorithm

$$\dot{\hat{f}}(t) = -\Psi F(\dot{e}_y(t) + \sigma e_y(t)) \qquad (21)$$

can realize $e_x(t)$ and $e_f(t)$ uniformly ultimately bounded. Note $\psi \in \Re^{r\times r}$ is the learning rate matrix. So, as a conclusion, the fault estimates expression is:

$$\hat{f}(t) = -\psi F(e_y(t) + \sigma \int_{t_f}^{t} e_y(\tau)d\tau) \qquad (22)$$

Remark 5 *: Solving conditions in Theorem 2 needs the LMI toolbox in MATLAB. So, it is easy to solve (20). But, there are some difficulties in solving (19) and (20) simultaneously to extract P, F and G. So, it must transform (19) into the following optimization problem [22]: Minimize η subject to (20) and*

$$\begin{pmatrix} \eta I & F_x^TP - FC \\ (F_x^TP - FC)^T & \eta I \end{pmatrix} \succ 0 \qquad (23)$$

The proof and more explanation of this theorem are given in my previous work [18].

Remark 6 *: As the model of robot used is of variable settings for each sampling period, it makes the LMI resolution slowly. That's why, to accelerate the simulation and restrict the margin solution, it's better to propose constraints relaxation depending on inequality. As the parameters system contain the non linear element $\sin(\phi)$ and $\cos(\phi)$, it can be limited by a minimum and maximum region.*

$$-1 \leqslant \sin(\phi) \leqslant 1, -1 \leqslant \cos(\phi) \leqslant 1$$

To solve (20) and (23), it should consider this constraint:

$$B_{\min} \leqslant F_x = B(x) \leqslant B_{\max} \qquad (24)$$

where

$$B_{\min} = \begin{bmatrix} 0 & 0 & 0 \\ -b_1(\sqrt{3}+1) & -b_1(\sqrt{3}+1) & -2b_1 \\ -b_1(\sqrt{3}+1) & -b_1(\sqrt{3}+1) & -2b_1 \\ b_2 & b_2 & b_2 \end{bmatrix}$$

$$B_{\max} = \begin{bmatrix} 0 & 0 & 0 \\ b_1(\sqrt{3}+1) & b_1(\sqrt{3}+1) & 2b_1 \\ b_1(\sqrt{3}+1) & b_1(\sqrt{3}+1) & 2b_1 \\ b_2 & b_2 & b_2 \end{bmatrix}$$

In (20), it should use $B_{\max}$ to ensure the limit part of solution. In other hand, in (23) it should use $B_{\min}$ to respect constraints.

6 SIMULATION RESULT

If we consider the model given in (14), numerically, fixed the following values of system's parameters which are listed in Table 1.

So, we obtain the array's elements (given for A and B)

$$s_1 = -0.0034, s_2 = 0.4286, s_3 = -1.8750$$

$$v_1 = 18.5, v_2 = 809.375$$

Table 1. Wheeled mobile robot's parameters.

Symbol	Designation	Value
m	Robot's mass (Kg)	10
r	Common radius of the wheels (m)	0.04
D	Distance of the wheels from rotation point Q (m)	0.2
K	Driving torque gain	25.9
β	Linear friction coefficient	0.02
I_0	Common moment of inertia of the wheels (Kgm^2)	0.008
I_Q	Moment of inertia of the robot about Q (Kgm^2)	0.2

$\beta_{1,2,3,4}$ are variable elements because of dependence on state vector.

In this work, we will introduce an additive actuator fault $u_0 = \begin{bmatrix} 5 & 0 & 0 \end{bmatrix}^T$. Then, assuming d is the unknown input (or disturbance) vector where is the known disturbance matrix $D_x = \begin{bmatrix} 0.05 & 0.2 & 2 & 0 & 0.6 & 0 \\ 0 & 0.3 & 0 & 1 & 0 & 0.02 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$.

6.1 Cooperation Task

From a database that contains at each sampling time 12 distances, we can define the parameter of cooperation used in this work. Figure 5 shows the evolution of distances at each iteration since only beacons are fixed and each robot from the three robots used follows its paths. Suppose as given in Fig.4, there are three beacons:
B1(x_{b1},y_{b1}) = (0,0),   B2(x_{b2},y_{b2}) = (5,8),
B3(x_{b3},y_{b3}) = (10,0)
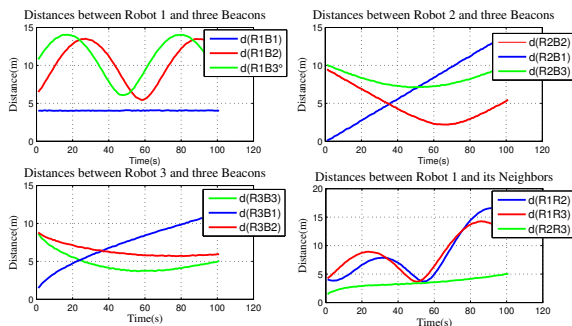
Then, by application to Newton Raphson algorithm

Fig. 5. Distances between MR, Beacons and NR.

we can extract the position of MR and the two NR at each

sampling time. Their trajectories are given in Fig.6. This figure concerns only the three robots and its trajectories. As for beacons, their positions are already fixed from the beginning. It should be noted that the unit used for all positions of robots in following figures is the meter (m). To ensure the effectiveness of cooperation algorithm, it
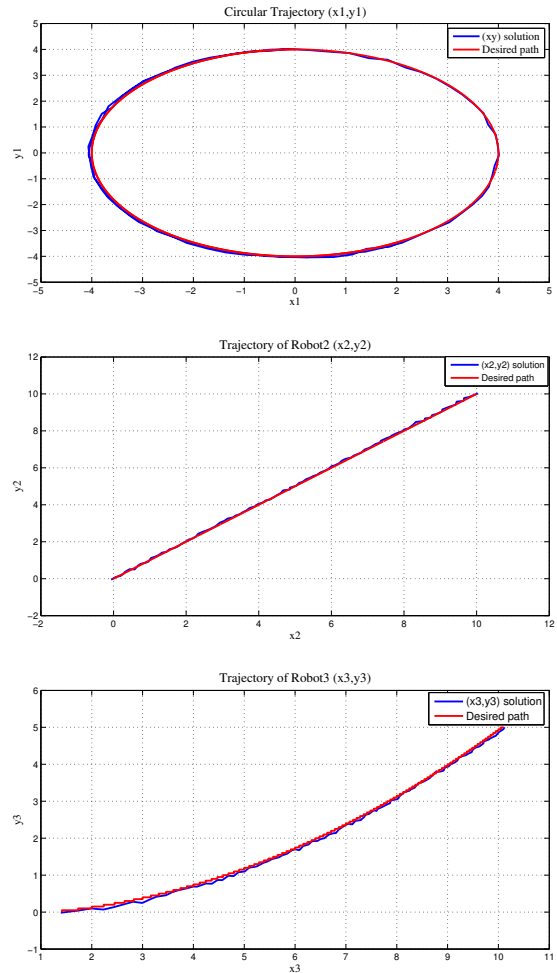

Fig. 6. Comparison between desired path and the trajectory obtained by cooperation algorithm.

shall identify the position robot error. It's clear from Fig.7 that this error is almost neglected. It does not exceed $0.1m$. Oscillations alternating between $\pm 0.1m$ are due to disturbance and time calculation that takes the cooperation algorithm. As conclusion, at this level, the position of the three-WMR is well-known. So, it's possible to estimate the state vector by an observer block.

6.2 Nominal case

It was assumed that the total experimental time is $100s$ with a sampling time $T_s = 1s$. It can be noted clearly
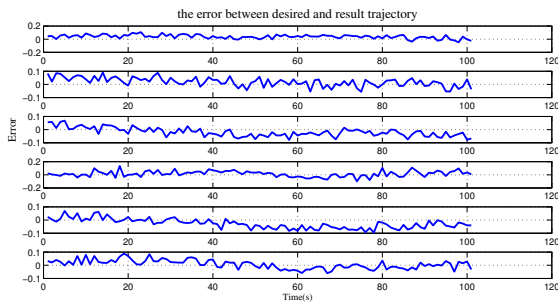
Fig. 7. difference between desired path and the trajectory obtained by cooperation algorithm

from Fig.8 that the systems response converges to the desired path in nominal conditions in absence of fault. It was assumed that a circular trajectory has a radius of curvature $4m$. There are some oscillations in the trajectory followed because of input disturbance. These oscillations can damage a mobile robot because of their electronic composition which can be affected by the vibration throughout the trajectory. That's why, disturbances must be decoupled.
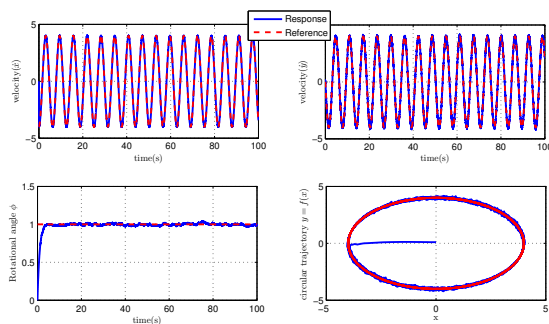
Fig. 8. Trajectory tracking of three-wheel omnidirectional MR.

6.3 Faulty case

Considering a partial actuator failure occurring at time $t_f = 40s$, the faulty-model response is shown in Fig .9. In the faulty case detailed in this figure, the deviation takes a few instants. So, the systems behavior and the controls evolution were changed at instant of fault included. Therefore, fault and disturbance vector can harm the systems responses even in the presence of a Lunberger observer. Therefore, this work exceeds this problem by the robust unknown input observer which not only estimates the state vector despite of disturbances but also estimates the fault included with better features. This may engender nice performances when correcting system. Thus, the advantage of the fault diagnosis method given is that it provides a good

estimate of faults, not only for the estimation step but also for the reconfiguration of control law. To detect the actuator fault, the observer block must be included. Hence, the residue is illustrated in Fig.10.
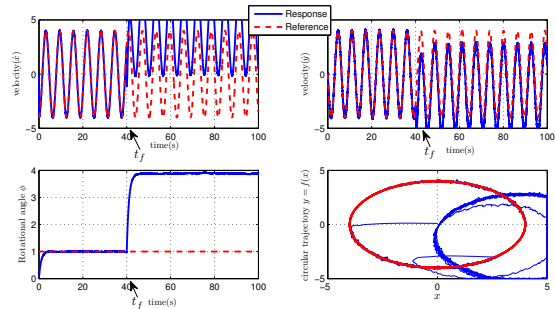

Fig. 9. Trajectory tracking of three-wheel omnidirectional MR in the faulty case.
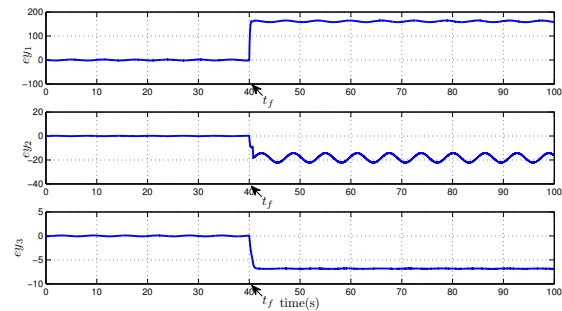

Fig. 10. The error output.

6.4 Fault estimation

Hereafter, we will highlight the fault estimation obtained thanks to a robust unknown input observer. Firstly, if there is no fault occurred and under the influence of disturbance, the fault estimation is shown in Fig.11. So, it's the order of 10^{-3} near to zero. Secondly, when introducing a fault vector $f(t) = \begin{bmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \end{bmatrix}$ where

$$f_1(t) = \begin{cases} 5 & if \quad t \geqslant t_f \\ 0 & if \quad t \prec t_f \end{cases} , \quad f_2(t) = 0 \quad and \quad f_3(t) = 0$$

It is easy from Fig.12 to select the peak achieved by an additive actuator fault thanks to the robust estimator used despite of the disturbance vector. Therefore, this estimation is characterized by the robustness of fault estimation
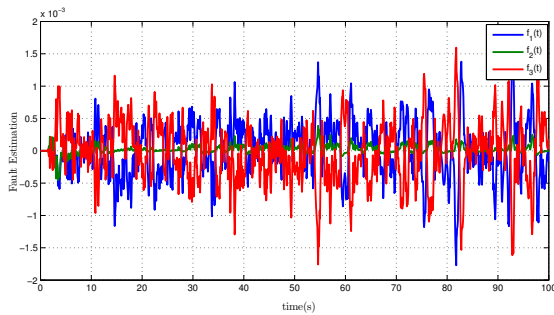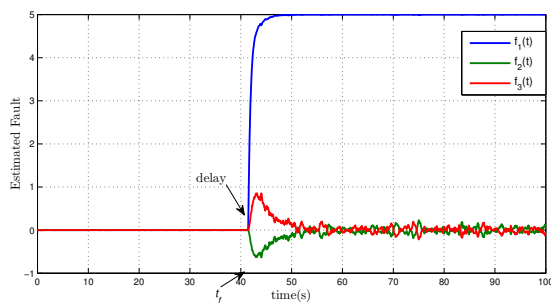
Fig. 11. Null Fault Estimation.

Fig. 12. Robust Fast Adaptive Fault Estimation.

response and the clear evolution of fault included. The delay to detect the fault is due to time calculation that takes the cooperation algorithm.

If we compare the results of fault estimation obtained in my previous work [23] with those mentioned in Fig.12, the difference is then clear in term of robustness.

Once the fault estimation phase is determined, the correction step is possible now. In the case of WMR, diagnosis module must be provided by the technique based on a fast and robust adaptive observer to drive the robot to desired path despite of the inclusion of an actuator fault.

7   CONCLUSION

In this brief, decomposition for a class of large-scale system was discussed, in which the whole system is naturally divided into observable subsystem and a unobservable one to study the global and local observability. These subsystems interacts each other by their neighbouring systems. The cooperation between mobile robots was developed to ensure the robust observer design and then guarantee the fast adaptive fault estimator. Further investigation might focus on the principle of robot network to benefit from several tasks in real time.

8   APPENDIX

Newton Raphson Algorithm
Goal: find the solution vector that fits with functions sys-

tem.

$$i = 0$$
$$x_0 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^T$$
$$while \left| \dot{f}(x_i) \right| \succ tolerance$$
$$do \begin{cases} i = i + 1 \\ x_i = x_{i-1} - \dfrac{1}{\ddot{f}(x_{i-1})} \dot{f}(x_{i-1}) \end{cases}$$
$$x = x_i$$
$$return(x)$$

The desired solution is $(x, y) = (x(1), x(4))$.

REFERENCES

[1] J. B. Rawlings and B. T. Stewart, "Coordinating multiple optimization-based controllers: New opportunities and challenges," *Journal of process control*, vol. 18, no. 9, pp. 839–845, 2008.

[2] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *Industrial Informatics, IEEE Transactions on*, vol. 9, no. 1, pp. 427–438, 2013.

[3] M. Egerstedt, "Behavior based robotics using hybrid automata," in *Hybrid Systems: computation and control*, pp. 103–116, Springer, 2000.

[4] T. Lozano-Perez, I. J. Cox, and G. T. Wilfong, *Autonomous robot vehicles*. Springer Science & Business Media, 2012.

[5] D. Smith, "Mobile robotic assistant for multipurpose applications," Mar. 10 2015. US Patent 8,977,396.

[6] G. Indiveri, "Swedish wheeled omnidirectional mobile robots: kinematics analysis and control," *IEEE transactions on robotics*, vol. 25, no. 1, pp. 164–171, 2009.

[7] L. E. Parker, "Multiple mobile robot systems," in *Springer Handbook of Robotics*, pp. 921–941, Springer, 2008.

[8] M. M. Zavlanos, A. Ribeiro, and G. J. Pappas, "Network integrity in mobile robotic networks," *Automatic Control, IEEE Transactions on*, vol. 58, no. 1, pp. 3–18, 2013.

[9] S. Ding, *Model-based fault diagnosis techniques: design schemes, algorithms, and tools*. Springer Science & Business Media, 2008.

[10] R. Isermann, *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer Science & Business Media, 2006.

[11] X. Wei, L. Liu, and L. Jia, "Fault diagnosis for high order systems based on model decomposition," *International Journal of Control, Automation and Systems*, vol. 11, no. 1, pp. 75–83, 2013.

[12] M. Sami and R. J. Patton, "Active fault tolerant control for nonlinear systems with simultaneous actuator and sensor faults," *International Journal of Control, Automation and Systems*, vol. 11, no. 6, pp. 1149–1161, 2013.

[13] P. Weber, B. Boussaid, A. Khelassi, D. Theilliol, and C. Aubrun, "Reconfigurable control design with integration of a reference governor and reliability indicators," *International Journal of Applied Mathematics and Computer Science*, vol. 22, no. 1, pp. 139–148, 2012.

[14] N.-B. Hoang and H.-J. Kang, "A model-based fault diagnosis scheme for wheeled mobile robots," *International Journal of Control, Automation and Systems*, vol. 12, no. 3, pp. 637–651, 2014.

[15] R. Challoo and S. Dubey, "Simultaneous state and actuator fault estimation with fuzzy descriptor pmid and pd observers for satellite control systems," *International Journal of Robotics and Automation (IJRA)*, vol. 2, no. 5, p. 344, 2011.

[16] S. I. Roumeliotis, G. S. Sukhatme, and G. A. Bekey, "Fault detection and identification in a mobile robot using multiple-model estimation," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 3, pp. 2223–2228, IEEE, 1998.

[17] P. Benner, V. Mehrmann, and D. C. Sorensen, *Dimension reduction of large-scale systems*, vol. 45. Springer, 2005.

[18] O. Hrizi, B. Boussaid, A. Zouinkhi, and M. N. Abdelkrim, "Robust unknown input observer-based fast adaptive fault estimation: Application to mobile robot," in *Handbook of Research on Advanced Intelligent Control Engineering and Automation*, pp. 443–473, IGI GLobal, 2015.

[19] J. Liu and J. Wu, *Multiagent Robotic Systems*. CRC press, 2001.

[20] S. G. Tzafestas, *Introduction to mobile robot control*. Elsevier, 2013.

[21] B. Jiang, J. L. Wang, and Y. C. Soh, "An adaptive technique for robust diagnosis of faults with independent effects on system outputs," *International Journal of Control*, vol. 75, no. 11, pp. 792–802, 2002.

[22] M. Corless and J. Tu, "State and input estimation for a class of uncertain systems," *Automatica*, vol. 34, no. 6, pp. 757–764, 1998.

[23] O. Hrizi, B. Boussaid, M. Abdelkrim, and C. Aubrun, "Fast adaptive fault estimation algorithm: Application to unicycle robot," in *Control and Fault-Tolerant Systems (SysTol), 2013 Conference on*, pp. 714–719, IEEE, 2013.

B. Boussaid born in 1972, is associate professor at National School of Engineers of Gabes, Tunisia, in Electrical Engineering and Automatic Control Department. He received his PhD degree in 2011 in Automatic Control and Computer Sciences from both University of Nancy and University of Gabes. He received his Engineering Degree in 1997 in Electrical engineering from National School of Engineers of Tunis. He is currently a member of the Research Centre in Automation of Nancy (CRAN) and the Research Laboratory on Modeling, Analysis and Control of Systems (MACS). His research interests include fault tolerant control and fault detection in wireless sensor networks with application to wind firms and robots networks.

A. Zouinkhi is Associate Professor at the National Engineering School of Gabes (Tunisia) and a member of Modeling, Analysis and Control Systems laboratory. He received the Notional engineering Degree from the Notional Engineering School of Monastir (ENIM), Tunisia in 1997 in industrial computing. He received his PhD degree in 2011 in Automatic Control from the National Engineering School of Gabes (Tunisia) and a PhD degree in Computer Engineering from the Nancy University (France). His research activities are focused on Distributed Systems, Smart Objects theory and applications, Ambient Intelligence systems and architectures, RFID and Wireless Sensors Network Concepts and Applications in manufacturing and supply chain.

M. N. Abdelkrim obtained a Diploma in Technical Sciences in 1980, his Master Degree in Control in 1981 from the ENSET school of Tunis (Tunisia), and his PhD in Control in 1985 and the Doctorate in Sciences Degree (Electrical engineering) in 2003 from the ENIT School of Tunis. Since 2003 he is Professor at the Electrical Engineering Department (Control) of the National Engineering School of Gabes (Tunisia) and he is manager of the Modeling, Analysis and Control Systems (MACS) laboratory.

C. Aubrun received a Ph.D. in control engineering from the University of Nancy, France, in 1992. He is currently a member of the Research Centre in Automation of Nancy (CRAN). Since 2005 he has been a professor in the Department of Electrical Engineering, Institute of Technology, University of Nancy. He has been involved in many projects with industry as well as European projects. His research interests lie in complex systems diagnosis and fault tolerant control areas with particular applications to water treatment processes and networked control systems.

O. Hrizi is Ph.D. Student at National School of Engineers of Gabes in Electrical Engineering department. She received the Engineer diploma in 2011, and the Master Degree in 2012 from the university of Gabes. Her research interests include fault detection and fault tolerant control and its application to multi-robot sytems.

AUTHORS' ADDRESSES

O. Hrizi, Ph.D.
Asst. Prof. B. Boussaid.
Asst. Prof. A. Zouinkhi.
Prof. M. N. Abdelkrim.
University of Gabes, MACS Research Laboratory
National School of Engineers of Gabes
Omar Ibn Khattab street, Zrig, 6029, Gabes, Tunisia
email: hrizi.olfa@gmail.com,
boumedyen.boussaid@gmail.com

Prof. C. Aubrun.
Centre de Recherche en Automatique de Nancy (CRAN)
University of Lorraine, BP 70239, 54506 Vandoeuvre Cedex,
France
email: christophe.aubrun@univ-lorraine.fr