

Monitoring of robot path tracking: Reconfiguration strategy design and experimental validation

S. Benmoussa, R. Loureiro, Y. Touati, and R. Merzouki

Abstract—This paper presents experimental validation in real time of fault detection and isolation and fault tolerant control algorithms for healthy monitoring of an Omni-directional platform, called Robotino@. The latter is composed of three actuated subsystems. The purpose of using fault diagnosis algorithms is to supervise the safe operating of the system, and to study the system reconfigurability strategies in order to ensure that the system remains able to follow a desired trajectory. For such purpose, the fault detectability and isolability is based on analytical redundancy relations. The latter are constraint relations expressing the nominal system behavior and they are written in terms of the measured system variables. Once a fault is detected and the faulty actuated subsystem is determined, the system reconfigurability algorithm analyses the redundancy presented on the former and an appropriate control strategy is applied.

I. INTRODUCTION

Several techniques have been developed to track a desired trajectory by an autonomous robots. Among them ones which are based on the dynamic and the kinematic models of the system [1], [2]. The latter algorithm has been experimented on real vehicles and it has shown its efficiency. However, the occurrence of a fault on the system can cause the loss of the path tracking. For that propose, a selection of the tracking controllers should take into account the healthy state of the system. Many advantages can be drawn out of the health monitoring of path tracking of a mobile robot such as: increase of safety, accomplish the required mission even in a degraded mode ...etc. For those reasons, ensuring that a mobile robot continues following a desired path even in the presence of fault is essential.

In fact, the ability to keep a process system running even when it is subject to faults is directly dependent on the implemented fault tolerant control (FTC) strategy. The latter is qualified by a passive strategy if the used control techniques is considered robust in the sense that the closed loop system is insensitive to a sub-set of possible faults (usually only one fault is considered), [3]. An active FTC one makes use of the fault information provided by fault detection and isolation (FDI) algorithms to reconfigure the system or accommodate the fault, [4]. Among FDI algorithm, one can found those based on the historical data of the process, which are referred to Data-based methods [5], [6], and those based on the knowledge of the system model or structure, which are referred to as Model-based methods [7], [8], [9]. FDI and FTC strategies applied on mobile

robot were the subject of a survey [10]. In [11], a FTC strategy for a mobile robot in combination with a model-based FDI algorithm was considered. The latter is a hybrid automaton model which represents the robot's components state. [12] developed a fault diagnosis and a FTC procedure applied to an over-actuated vehicle. Almost the presented FTC strategies are based on the inverse kinematics model.

The present paper considers model-based fault diagnosis based on Bond Graph (BG) approach to supervise the robot dynamic. In fact, BG tool is devoted to represent graphically the structure as well the dynamic of the system, it is the interface between the physical system and its mathematical model. More details about BG can be found in [13]. BG model-based FDI is based on analytical redundancy relations (ARRs) [9]. They are generated by exploiting the causal and structural properties of the BG approach. The structure of ARRs forms a fault signature matrix, from where fault detectability and isolability is studied. BG was used also for the design of FTC strategies in the presence of a fault [14].

The main interest of the proposed paper relies on a real-time implementation of the control and fault diagnosis algorithms that enable to keep a mobile robot operating even under faulty conditions. In fact, based on the fault diagnosis information, whether the robot is in normal operation or in degraded mode, the reconfiguration strategy considers the path tracking controller. In the degraded mode, the path tracking controller is based on the inverse kinematic model of the robot.

The other sections of the paper are organized as follows: in section two, the robot description is developed. The path tracking monitoring procedure is presented in the third section. In the fourth section, the obtained experimental results in real-time are given. The paper ends with general synthesis and remarks.

II. SYSTEM DESCRIPTION

Robotino@ (Fig.1) is a didactic robot built by Festo company [15]. It is equipped with three omnidirectional wheels, placed at 120° from one to another on a circular frame. This allows it to move along three axes (Longitudinal, lateral, yaw). The traction torque, computed by controllers is furnished to each wheel through a DC-motor. Moreover, the instrumentation architecture of Robotino@ is composed of gyroscope measuring the yaw velocity, and by two other sensors in each electromechanical subsystem providing the current and motor position measurements. By using the latter information and the gyroscope, Robotino@ position (X, Y) in the plane is estimated by using the odometry technique.

This work was not supported by any organization
The authors are with LAGIS UMR CNRS 8219,
Université Lille Nord du France, 59000 Lille Cedex.
samir.benmoussa@polytech-lille.fr



Fig. 1. Image of Robotino@ and its actuation subsystem.

The technical specifications of Robotino@ are:

- Maximum velocity : 10 km/h,
- Dimensions :
 - Diameter: 0.370 m
 - Height incl. housing: 0.210 m
- Weight : 11 Kg.

A. Bond Graph Model of Robotino@

In this section, we present the BG model of Robotino@. The former is given by Fig. 2 and it is composed of the following dynamics:

- 1) Electromechanical systems dynamics in interaction with the ground,
- 2) Longitudinal dynamics,
- 3) Lateral dynamics,
- 4) Yaw dynamics.

Each j^{th} electromechanical part, with $j \in [1, 2, 3]$ is divided in three parts : a DC motor part, a gear part and a wheel part. The DC motor part is a combination of an electrical part and a mechanical dynamics. The electrical part is composed of an electrical resistance R_j and inductance L_j , a voltage source U_j and electromotive force feedback EMF (with a constant k_{ej}), which is linear to the angular velocity of the rotor. The mechanical part is characterized by the rotor inertia J_{ej} , and a viscous friction parameter f_{ej} . The gear part is the link between the DC motor part and the wheel part. It is modeled on BG by a TF element with a reduction constant N_j . The wheel part in interaction with the ground represents the load of the electromechanical system, characterized by its inertia J_{sj} , and a viscous friction parameter f_{sj} . The radius r_j is considered constant. In addition, in this work the longitudinal contact efforts F_{xj} are considered constant representing a punctual coulomb effort equal to $2N$. It has been identified experimentally by identifying the minimum input voltage used to start the wheel motion on a homogeneous surface. This is done under the assumption that the road is identical and the robot moves at low speeds.

The measurement scheme of each electromechanical part is represented by "Df" elements representing the current sensor ($Df : i_j$) and the rotor velocity measurement ($Df : \dot{\theta}_{ej}$). We remark that the real system is equipped with a position sensor. However, since the latter information is

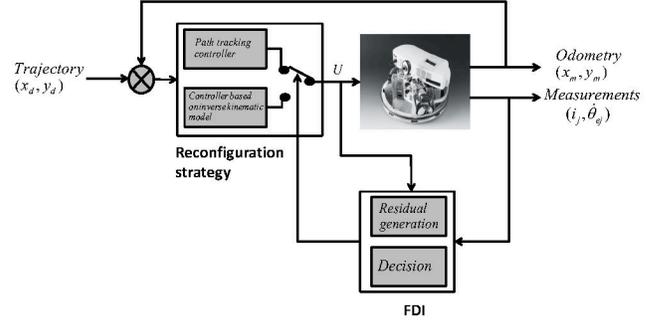


Fig. 3. General scheme of the reconfiguration strategy applied to Robotino@

neither effort nor flux variable, the position sensor is replaced by a velocity one on BG model. The latter measurement is obtained by differentiating the position variable. The quality of the position measurement is clean, and acquired with an accurate encoder of 500 points/rotation, where the velocity estimation through differentiation is allowed.

This BG model considers also the longitudinal, the lateral and the yaw dynamics. Where N and Z are respectively the distances from the Robotino@ CoG to the front and the rear axles. L is the distance between the 1st and the 3rd electromechanical sub-systems. The Robotino@ inertia is represented by an $I : m$ -element. The numerical parameters of the mobile robot are identified experimentally and given in Table. I.

TABLE I
SPECIFICATION CHARACTERISTICS OF ROBOTINO@

R_j	3.5	(Ω)	J_{sj}	0.0063	($Kg.m^2$)
L_j	0.0089	(H)	f_{sj}	0.0004	($N.m.s.rad^{-1}$)
k_{ej}	0.055	($N.m.A^{-1}$)	N_j	16	-
J_{ej}	0.0000795	($Kg.m^2$)	r	0.04	(m)
f_{ej}	0.0004	($N.m.s.rad^{-1}$)	N	0.11	(m)
Z	0.14	(m)	L	0.21	(m)

III. PATH TRACKING MONITORING

The idea of this work is to use the robot to follow a desired path in both healthy and faulty conditions. Therefore, in this paper we have developed an FDI algorithm and combine it with a reconfiguration strategy. The aim of the fault diagnosis step is to verify the healthy behavior of electromechanical systems. Once a fault is detected in an electromechanical subsystem, this information is used by the FTC step. The latter consists on re-generating the path tracking controller. In this section we start by introducing the fault diagnosis step. The general scheme of the FTC strategy adopted in this work is given by Fig. 3.

A. Fault Detection and Isolation

The considered FDI algorithm in the presented paper is a model-based one. It consists on comparing the behavior of the physical system obtained from the measurement data and

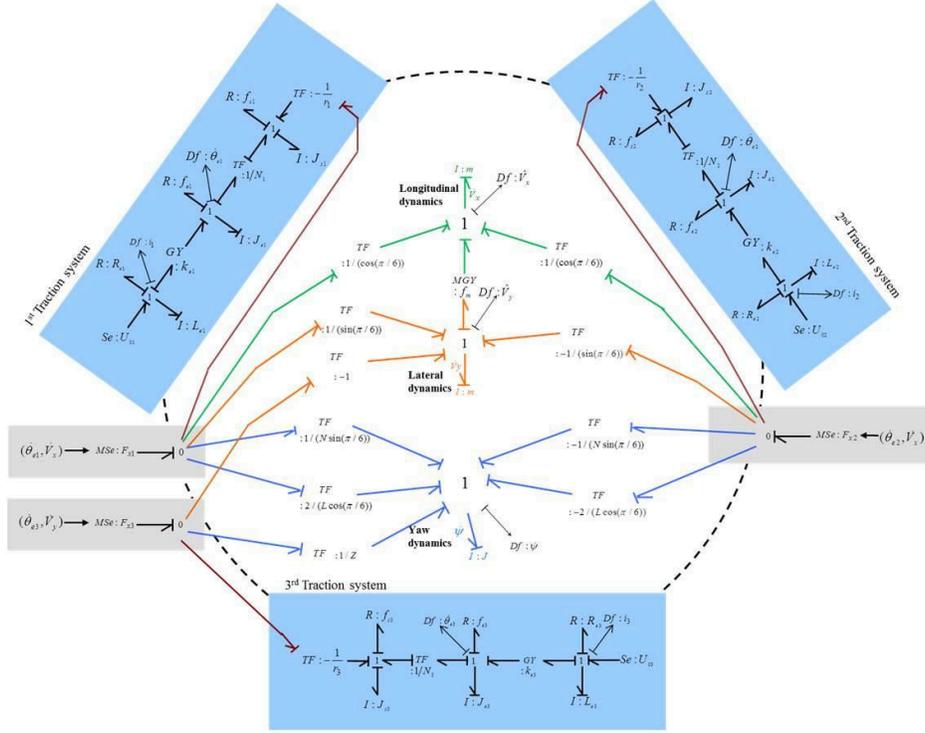


Fig. 2. BG model of Robotino@

the information issued from the mathematical model of the system. The difference between this information is a residual. In fault-free case, the residuals should approach zero or near to it due to measurement, parameters, and model uncertainties. Or in fault-case, the residuals are triggered. The latter information is used for fault detection and isolation. The residuals are the evaluation of analytical redundancy relations (ARR). They are obtained from the BG model of the system by using the methodology based on causality inversion, introduced in [9]. It consists on dualisation of the system measurements to sources of information, and assigns a preferred derivative causality to the BG model of the system. Then, the ARRs can automatically be computed by following the causal path to eliminate the unknown variables [16].

For example, ARR_1 associated with the 3rd electromechanical subsystem of the BG model of the Fig. 2, is obtained as following:

From the junction 1_① of the 3rd electromechanical subsystem, the next constitutive equation is obtained:

$$\begin{aligned} \sum e_i &= 0, i = 1..4 \\ f_1 &= f_2 = f_3 = f_4 = i_3 \end{aligned} \quad (1)$$

where :

$$\begin{cases} e_1 = U_3 \\ e_2 = -Re_3 \cdot i_3 \\ e_3 = -L_{e3} \frac{d}{dt} i_3 \\ e_4 = -k_{e3} \theta_{e3} \end{cases} \quad (2)$$

Thus, ARR_{13} is given by :

$$ARR_{13} = U_3 - L_{e3} \frac{di_3}{dt} - Re_3 i_3 - k_{e3} \theta_{e3} \quad (3)$$

In similar way, the ARR_{23} is obtained, and it is given by

$$ARR_{23} = k_{e3} i_3 - (f_{e3} + \frac{f_{s3}}{N_3^2}) \theta_{e3} - (J_{e3} + \frac{J_{s3}}{N_3^2}) \frac{d\theta_{e3}}{dt} - \frac{F_{x3} r}{N_3} \quad (4)$$

These ARRs can be obtained automatically by using a dedicated software [17].

The structure of ARRs of the Eq.3 and Eq.4 forms a fault signature matrix (FSM) where each entry of this matrix (s_{ij}) holds Boolean values, and the signature vector of each component fault (E_j) is given by the row vector $V_{E_j} = [s_{1j} s_{2j} \dots s_{mj}]$ where the values s_{ij} are affected as follows:

$$s_{ij} = \begin{cases} 1, & \text{if the component } E_j \text{ influences } r_j \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

A fault on a component is detectable (D_b) if at least one residual is affected by it. It is isolable (I_d) if its associated fault signature vector V_{E_j} is unique, i.e., no other fault has the same signature. Thus, the FSM deduced from the ARRs of Eq.3 is given by the Table.II.

B. Path tracking controller

Given a planar trajectory coordinates (x_d, y_d) , the controller aims to track the considered trajectory with accuracy. The latter is ensured by tracking the lateral motion of

TABLE II
FAULT SIGNATURE MATRIX (FSM)

Part	Comp.	Residuals struc.		D_b	I_b
-	-	r_{1j}	r_{2j}	-	-
Electrical	$Se : U_j$	1	0	1	0
	L_j	1	0	1	0
	R_{ej}	1	0	1	0
	$Df : i_j$	1	1	1	0
	k_{ej}	1	1	1	0
Mech.	$Df : \theta_{ej}$	1	1	1	0
	f_{ej}	0	1	1	0
	J_{ej}	0	1	1	0
	N_j	0	1	1	0
Wheel	f_{sj}	0	1	1	0
	J_{sj}	0	1	1	0
	r	0	1	1	0
	F_{xj}	0	1	1	0

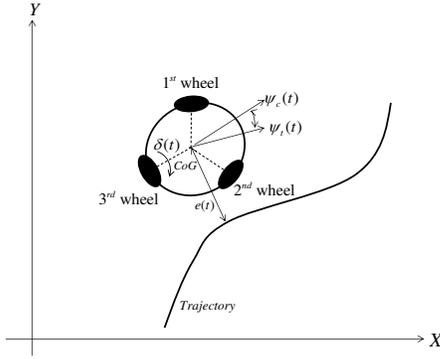


Fig. 4. Kinematic motion scheme of Robotino@

Robotino@. In the presented paper, the control strategy is similar the one described in [1] and used in [18]. It is given in Eq. (6) and its output is the desired steering angle $\delta(t)$:

$$\delta(t) = \psi_e(t) + \text{atan}\left(\frac{k_1 e(t)}{v(t) + k_2}\right) \quad (6)$$

Where, $\psi_e(t)$ is the difference between the desired path angle and the actual Robotino@ angle:

$$\psi_e(t) = \psi_c(t) - \psi_l(t). \quad (7)$$

The error $e(t)$ is the transversal distance between the Robotino's CoG to the desired trajectory. $v(t)$ is Robotino@ velocity. The desired steering angle $\delta(t)$ is applied to 2nd electromechanical sub-system through a PI controller.

The longitudinal motion of Robotino@ is ensured by a PI controller implemented on the 1st and 3rd electromechanical sub-systems. The desired velocity is computed based on the predictions of the desired angle of Robotino@ over a finite window, and on its variations.

C. Fault tolerant control

FTC strategies can be either system reconfiguration or fault accommodation depending on the fault information. In this work, the adopted FTC strategy is a reconfiguration

strategy. This choice is made when the fault cannot be estimated and its effect can be removed from the system. In our system, the nominal controller used for path tracking exploits the three free-fault wheels. The longitudinal motion of Robotino@ is ensured by two traction wheels (1st and 3rd wheels) and the lateral motion is tracked by using the steering wheel (2nd wheel). When a fault occurs on one of the three wheels' elements, the fault diagnosis step detects and isolates the faulty component. If the fault is not isolable, the faulty subsystem is identified.

Based on this information, the FTC step aims to reconsider the path tracking controller strategy. It is performed by transcending the faulty wheel and by adopting a second control law. Based on the use of the inverse kinematic model of Robotino@ by considering only the traction wheels, the desired angular velocity of each wheel is obtained. This allows to Robotino@ continues to follow the desired path even in degraded mode (i.e, presence of fault). The representation of the inverse kinematic model is given as follows:

$$\begin{bmatrix} V_x \\ V_y \\ \dot{\psi} \end{bmatrix} = \quad (8)$$

$$\begin{bmatrix} \frac{R}{3} \cos(\pi/6) & \frac{R}{3} \cos(\pi/6) \\ -\frac{R}{3} \sin(\pi/6) & -\frac{R}{3} \sin(\pi/6) \\ \frac{R}{L} \cos(\pi/6) - \frac{R}{L} \sin(\pi/6) & -\frac{R}{L} \cos(\pi/6) + \frac{R}{L} \sin(\pi/6) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_3 \end{bmatrix}$$

Thus, the considered fault scenario is the loss of one wheel. In other word, if two wheels are being faulty, the FTC strategy cannot be applied.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The aim of our experiments is to track a desired path depicted in Fig. 5-(solid line) under either normal or faulty situations.

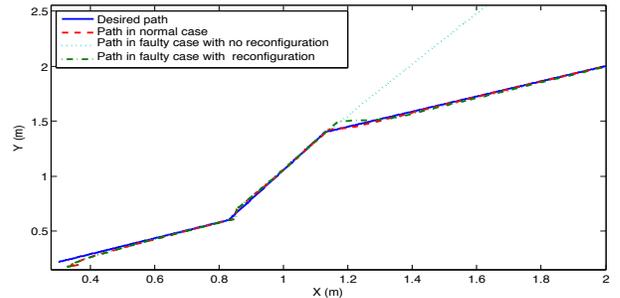


Fig. 5. Desired and tracked paths in the different scenarios.

The tracked path in the free-fault case is showed in Fig. 5-(dashed line), and the residuals of the steering sub-system are presented in Fig. 6. As defined, the Robotino@ is able to track with accuracy the desired path and the residuals remain bounded between their robust thresholds.

The measured signals used to compute the residuals are depicted in Fig. 7.

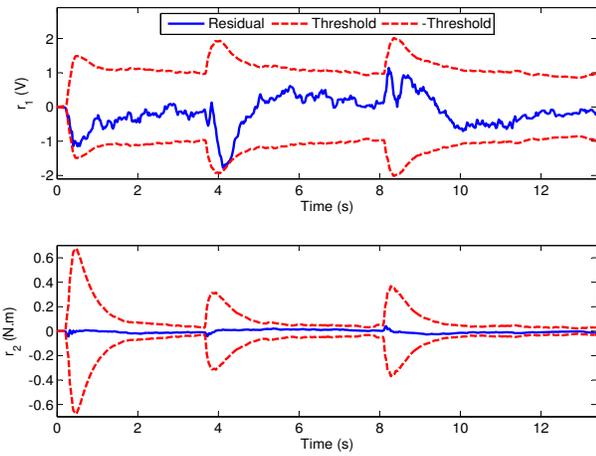


Fig. 6. Residuals evolution (r_1 and r_2) of the 3rd electromechanical traction wheel under nominal conditions.

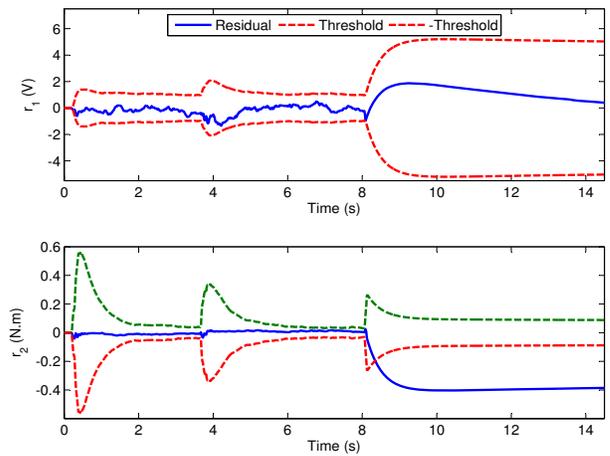


Fig. 8. Residuals evolution (r_1 and r_2) under faulty conditions with no reconfiguration.

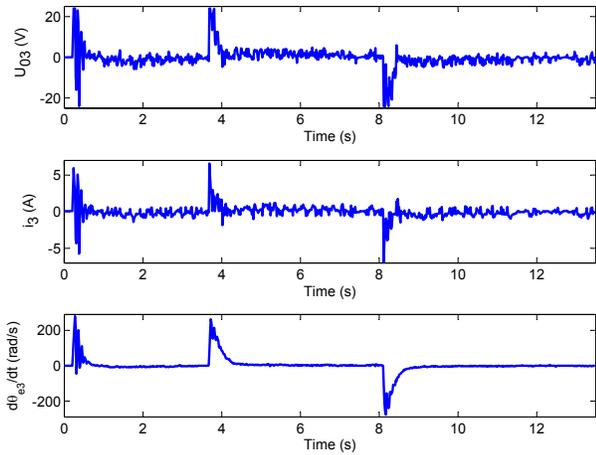


Fig. 7. Measured input and output signals of the 3rd electromechanical sub-system under healthy conditions.

In the second scenario, the 3rd wheel is stopped on traction at around 8 seconds. Two experiments were conducted with this scenario. The first experiment do not provides any reconfiguration measure, while the second one exploits the actuation redundancy presented in Robotino@ to keep tracking the desired trajectory with reduced performance.

The residual evolution during operation of the experiment when the wheel is stopped on traction and no reconfiguration actions are performed is given in Fig. 8. As expected, only the second residual (r_2) crosses the thresholds. Nevertheless, even if r_1 does not cross the thresholds, there is a variation of its value. This can be justified by some modeling errors.

The tracked paths of both experiments are given in Fig. 5. The path when no reconfiguration is performed is depicted in dotted lines, while the tracked path with the reconfiguration procedure is illustrated dash-dotted lines. It is clear from Fig. 5 that without reconfiguration actions, Robotino@ deviates from its desired position. This occurs because Robotino@

with the current control law is not able to turn after the fault occurrence. On the other hand, if the reconfiguration strategy is performed, Robotino@ is able to return to its desired trajectory after a slight deviation. This deviation happens due to the fault detection delay caused by the time that r_2 takes to cross the robust thresholds. This delay is about 0.5 second and it is depicted in Fig. 9 and 10.

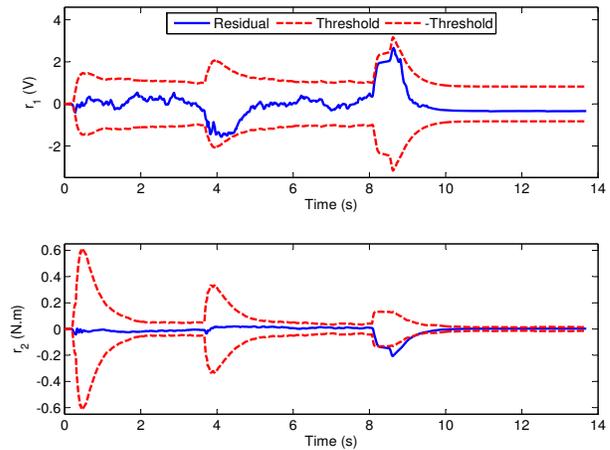


Fig. 9. Residuals evolution (r_1 and r_2) under faulty conditions with reconfiguration.

The measured signals used to compute the residuals are depicted in Fig. 10.

It can be seen in Fig. 10, that after the reconfiguration strategy is activated, the input voltage (U_{03}) and the measured current (i_3) become zero.

Moreover, once the reconfiguration strategy selects the new control law, Robotino@ is then controlled in term of steering with differentiation of wheel velocities. The desired wheel velocities of both traction sub-systems are depicted in Fig. 11.

The tracking error of all experiments is given in Fig.

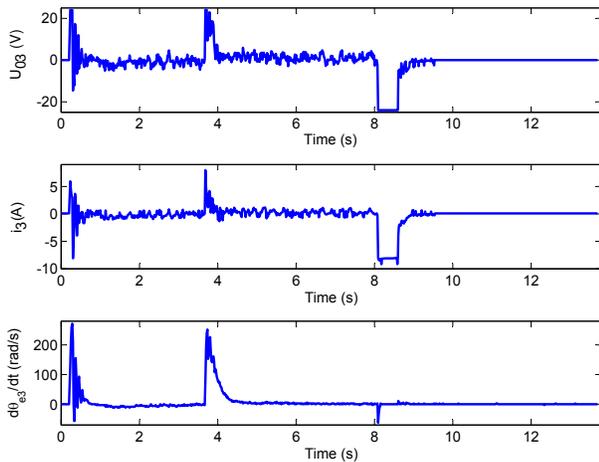


Fig. 10. Measured input and output signals of the 3rd electromechanical traction system under faulty conditions.

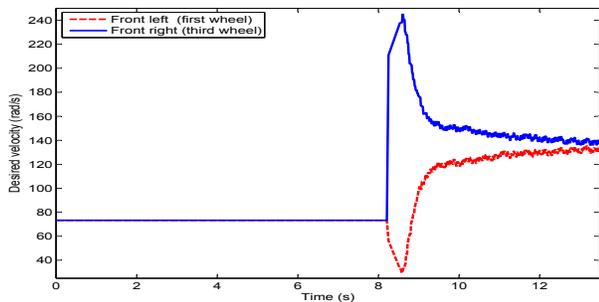


Fig. 11. Desired wheel velocity of both traction subsystems.

12. The error is the smallest when operating under nominal conditions. If the fault is not compensated, the error increases over time (dotted line). When the reconfiguration is performed the tracking error increases until the reconfiguration strategy is activated and then the error decreases again.

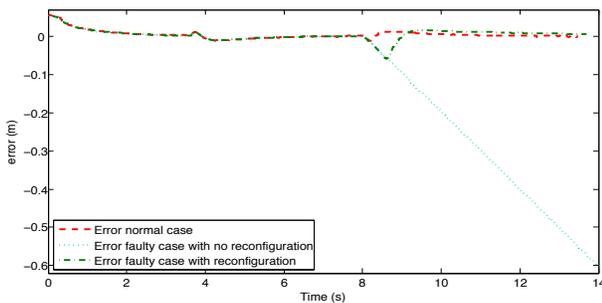


Fig. 12. Tracking error in the different scenarios.

V. CONCLUSIONS

In this work, we presented a fault tolerant control strategy for healthy monitoring of robot path tracking. The experimental results have demonstrated that when the robot is not equipped with such procedure, the occurrence of a fault can lead to a deviation of the robot desired path. To overcome this

problem, we integrated a fault diagnosis algorithm based on analytical redundancy relations and a fault tolerant control strategy based on inverse kinematic model of the robot. The experimental results done in real-time have shown the efficiency of the proposed strategy and the need of such algorithms for a healthy path tracking.

REFERENCES

- [1] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo, and S. Thrun, "Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing," in *proceedings of the 26th american control conference*, 2007.
- [2] J. Wang, J. Steiber, and B. Surampudi, "Autonomous ground vehicle control system for high-speed and safe operation," in *American Control Conference, 2008*, June 2008, pp. 218–223.
- [3] R. J. Patton, "Fault-tolerant control systems. the 1997 situation," in *Proceedings of the 3rd IFAC symposium on fault detection, supervision and safety for technical processes*, August 1997, pp. 1033–1055.
- [4] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual Reviews in Control*, vol. 32, pp. 229–252, 2008.
- [5] Z. Yongli, H. Limin, and L. Jinling, "Bayesian networks-based approach for power systems fault diagnosis," *IEEE Transactions on Power Delivery*, vol. 21, no. 2, pp. 634–639, 2006.
- [6] B. Li, M. Y. Chow, Y. Tipsuwan, and J. C. Hung, "Neural-network-based motor rolling bearing fault diagnosis," *Industrial Electronics, IEEE Transactions on*, vol. 47, no. 5, pp. 1060–1069, Oct 2000.
- [7] R. Patton and J. Chen, "observer-based fault detection and isolation : robustness and applications," *Control Engineering Practice*, vol. 5, pp. 671–682, 1997.
- [8] D. Henry, A. Zolghadri, F. Castang, and M. Monson, "A new multi-objective filter design for guaranteed robust fdi performance," in *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, vol. 1, 2001, pp. 173–178 vol.1.
- [9] B. Ould-Bouamama, A. K. Samantaray, M. Staroswiecki, and G. Dauphin-Tanguy, "Derivation of constraint relations from bond graph models for fault detection and isolation," in *International Conference on Bond Graph Modeling and Simulation*, 2003, pp. 104–109.
- [10] Z. Duan, Z. Cai, and Y. Jin-xia, "Fault diagnosis and fault tolerant control for wheeled mobile robots under unknown environments: A survey," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, April, pp. 3428–3433.
- [11] M. Brandstotter, M. Hofbauer, G. Steinbauer, and F. Wotawa, "Model-based fault diagnosis and reconfiguration of robot drives," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSSJ International Conference on*, 29 Oct–Nov. 2, pp. 1203–1209.
- [12] M. Djeziri, R. Merzouki, B. Bouamama, and M. Ouladsine, "Fault diagnosis and fault-tolerant control of an electric vehicle overactuated," *Vehicular Technology, IEEE Transactions on*, vol. 62, no. 3, pp. 986–994, 2013.
- [13] D. Karnopp and R. Rosenberg, *System Dynamics : A unified approach*, J. W. S. Inc, Ed. John Wiley & Sons, 1975.
- [14] R. Loureiro, R. Merzouki, and B. Bouamama, "Bond graph model based on structural diagnosability and recoverability analysis: Application to intelligent autonomous vehicles," *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 3, pp. 986–997, March 2012.
- [15] "Festo." [Online]. Available: <http://www.festo.com/net/startpage/>
- [16] C. B. Low, D. Wang, S. Arogeti, and J. B. Zhang, "Monitoring ability analysis and qualitative fault diagnosis using hybrid bond graph," in *Proceedings of the 17th World Congress The International Federation of Automatic Control Seoul, Korea, July 6-11, 2008*, pp. 10516–10521.
- [17] B. Ould-Bouamama, M. Staroswiecki, and A. K. Samantaray, "Software for supervision system design in process engineering," in *IFAC World Congress*, 2006, pp. 691–695.
- [18] V. B. Sabbagh, E. J. R. Freitas, G. M. M. Castro, M. M. Santos, M. F. Baleeiro, T. M. da Silva, P. Iscold, L. A. B. Torres, and G. A. S. Pereira, "Desenvolvimento de um sistema de controle para um carro de passeio autônomo," in *Congresso Brasileiro de Automática*, 2010.