

# Fault-tolerant Control of a Service Robot using a LPV Robust Unknown Input Observer

Alberto San Miguel, Vicenç Puig and Guillem Alenyà<sup>1</sup>

**Abstract**—In this paper, the control problem of a service robot under exogenous forces that might be exerted on the system is addressed, considering them as faults. Proposed approach is based on using a fault estimation scheme together with a Robust Unknown-Input Observer (RUIO) that allows to estimate the fault as well as the robot state, along with a state feedback control strategy. After the fault occurrence, from the fault estimation, a feedforward control action is added to the feedback control action to compensate the fault effect. To cope with the robot non-linearity, its model is transformed into a quasi-LPV model. Both the state-feedback and RUIO are designed using an LMI-based approach considering a gain-scheduling alike scheme, where global conditions have been considered. To illustrate the proposed fault-tolerant approach, a mobile service robot TIAGo, developed by PAL robotics, is used.

**Index Terms**—Fault-tolerant control, Linear parameter varying, Unknown-input observer, Robustness, Service robot

## I. INTRODUCTION

Over the last few years service robots have been increasingly introduced in our daily lives. According to the International Federation of Robotics (IFR), since 2016 there has been a yearly increase of 15% on its sales [1]. Although service robots have been designed to perform tasks on highly dynamic and unpredictable anthropic domains, faults might appear, some of them regarding interaction, e.g. misleading interpretations of the human actions or unexpected scenarios beyond nominal operation. Also, their inherent complexity make them prone to failures at all their levels, from the low-level actuators and sensors to the high-level decision layers. All these factors can lead to a degradation on the performance of the robot or imply critical damage to it, that might even jeopardise its safety. Thus, their ability to autonomously overcome most of these situations in a safe and efficient manner must play a fundamental role in their implementation.

Diagnosis and fault-tolerant control techniques have been widely studied for many years. However, only on certain generic robotic platforms some of these approaches have been successfully applied, as e.g. for wheeled mobile robots [2], being still considered as a relative new field of study for robotic systems. Current

fault diagnosis techniques have been classified into three different categories according to their common key characteristics [3], namely Data-driven, Model-based and Knowledge-based methods. Most of the current research is focused on model-based methods, as analytical expressions and methodologies have been developed for robotics systems [4] in order to describe their behaviour from the physical point of view. Particularly, Linear Parameter Varying (LPV) modelling has been presented as an approach for dealing with non-linear behaviours presented by robotic platforms in a linear-like manner. See for e.g. [5], where the problem is stated for both additive and multiplicative faults on a simplified model, and [6], where the same method is applied for the compensation of the friction effects associated to mechanical joints. Data-driven techniques are useful when non-structured scenarios are faced, with the main drawback of requiring a high-computational on-line expense that might make their implementation unfeasible. Although some advances have been done by carrying involved training phases off-line, as in [7], they still do not successfully include all the three-stages (Detection, Isolation and Identification) of the fault diagnosis problem.

This paper is focused on tackling the problem where exogenous forces affect a robotic system, by using a model-based technique able to handle these circumstances while assuring certain performance characteristics. This forces have been denoted as faults due to its a-priori unknown magnitude, effect and eventuality, assuming that they might hinder the fulfillment of desired tasks. This problem has been motivated by the TIAGo humanoid robot by PAL Robotics<sup>1</sup>, designed as a service robot to perform in highly dynamic anthropic environments. Particularly, as a prove of concept, its 2-DOF head subsystem has been chosen, as it does present many of the non-linear behaviours associated with robotic platforms. Therefore, a Fault-Tolerant approach is presented for this problem, where from a LPV representation of the system, an optimal performance criterion is used on the design of a polytopic state-feedback strategy. Faults are estimated using a Reference Control structure, parallel to the main one, and their magnitude is directly compensated within the control scheme. One of the most remarkable points

This work is supported by the Spanish State Research Agency through the María de Maeztu Seal of Excellence to IRI MDM-2016-0656

<sup>1</sup> All the authors are from Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Llorens i Artigas, 4-6, 08028 Barcelona, Spain {asanmiguel, vpuig, galenya}@iri.upc.edu

<sup>1</sup>TIAGo by PAL Robotics: <https://tiago.pal-robotics.com/> (Accessed 22 June 2019)

has been assuming that not all the states used in the control architecture are measurable or available on-line, being necessary to obtain a estimation of state variables decoupled from the fault effects, which has been done through the Robust Unknown Input Observer (RUIO) [8].

The paper is structured as follows: In Section II, the service robot case study is presented. Section III presents the robust fault/state estimation scheme. Section IV describes the complete fault-tolerant control scheme. Section V illustrates the performance of the proposed approach using the service robot case study. Finally, Section VI draws the main conclusions and presents the future research paths.

## II. CASE OF STUDY

### A. Analytical Model

As any model-based approach, a set of analytical expressions that describe the behaviour of the system has to be determined. Considering a representation as a two-link manipulator with two rotational joints, the Newton-Euler method has been applied. In Figure 1, the head subsystem is presented along with its abstracted link scheme.

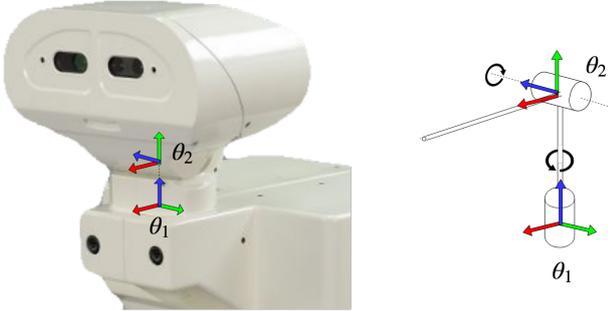


Fig. 1. Head subsystem of the TIAGo robotic platform, represented as a two-manipulator link.

Obtained expressions can be arranged into the configuration-space form, where the joint torque vector  $\tau$  is given as a function of the joint acceleration  $\ddot{\theta}$ , velocity  $\dot{\theta}$  and position  $\theta$  vectors:

$$\tau = \mathcal{M}(\theta)\ddot{\theta} + \mathcal{B}(\theta)[\dot{\theta}\dot{\theta}] + \mathcal{C}(\theta)[\dot{\theta}^2] + \mathcal{G}(\theta) \quad (1)$$

where  $\mathcal{M}(\theta) \in \mathbb{R}^{n \times n}$  describes the mass matrix of the manipulator,  $\mathcal{B}(\theta) \in \mathbb{R}^{n \times n(n-1)/2}$  the Coriolis terms,  $\mathcal{C}(\theta) \in \mathbb{R}^{n \times n}$  the centrifugal coefficients and  $\mathcal{G}(\theta) \in \mathbb{R}^n$  the gravity effects; being the number of joints  $n = 2$ .

From this formulation, the expressions for joint

accelerations and velocities can be defined as follows

$$\ddot{\theta}_1 = \frac{\mathcal{B}_{(1,1)}(\theta_2)}{\mathcal{M}_{(1,1)}(\theta_2)} \dot{\theta}_1 \dot{\theta}_2 + \frac{1}{\mathcal{M}_{(1,1)}(\theta_2)} \tau_1, \quad (2a)$$

$$\ddot{\theta}_2 = \frac{\mathcal{C}_{(2,1)}(\theta_2)}{\mathcal{M}_{(2,2)}} \dot{\theta}_1^2 + \frac{1}{\mathcal{M}_{(2,2)}} \tau_2 + \frac{\mathcal{G}_{(2,1)}(\theta_2)}{\mathcal{M}_{(2,2)}}, \quad (2b)$$

$$\dot{\theta}_1 = \frac{d}{dt} \theta_1, \quad (2c)$$

$$\dot{\theta}_2 = \frac{d}{dt} \theta_2, \quad (2d)$$

where

$$\mathcal{M}_{(1,1)}(\theta_2) = I_{zz_1} + m_2 L^2 + I_{xx_2} c_2^2 + I_{yy_2} s_2^2 - m_2 L^2 c(2\theta_2)^2,$$

$$\mathcal{M}_{(2,2)} = I_{zz_2} + m_2 L^2,$$

$$\mathcal{B}_{(1,1)}(\theta_2) = -2c_2 s_2 (I_{xx_2} - I_{yy_2}) + m_2 L^2 s(4\theta_2),$$

$$\mathcal{C}_{(2,1)}(\theta_2) = c_2 s_2 (I_{xx_2} - I_{yy_2}) - \frac{1}{2} m_2 L^2 s(4\theta_2),$$

$$\mathcal{G}_{(2,1)}(\theta_2) = -gm_2 L s(2\theta_2),$$

being  $c_i$  and  $s_i$  abbreviations for  $\cos(\theta_i)$  and  $\sin(\theta_i)$ , respectively, and  $g$  for the Earth gravity constant. Note that subindex notation  $(i, j)$  in those terms from matrices in Eq. (1) is used to denote its placement in row  $i$  and column  $j$  within its respective matrix. Regarding remaining parameters,  $I_{a_i}$  represent inertial tensor term on the  $a = x, y, z$  axis for the  $i$ -th link,  $m_i$  the mass of the  $i$ -th link and  $L$  the distance between the COG of link 2 and its joint axis, all in SI units. The value of these parameters are given by the manufacturer, gathered in Appendix A.

Further arrangements can be made on the equations to reduce the number of different terms in the expressions, and so from now on  $\mathcal{B}_{(1,1)}(\theta_2) = -2\mathcal{C}_{(2,1)}(\theta_2)$ .

### B. Quasi-LPV Model

Using expressions from Eq. (2), the following representation can be obtained into a state-space alike form,

$$\dot{x}(t) = A^c(\theta_2, \dot{\theta}_1)x(t) + B^c(\theta_2)u(t) + g_v^c(\theta_2), \quad (3)$$

considering the input torque action vector  $u = [\tau_1, \tau_2]^T$ , and the state-space vector  $x = [\dot{\theta}_1, \dot{\theta}_2, \theta_1, \theta_2]^T$ , such that  $\dot{x} = [\ddot{\theta}_1, \ddot{\theta}_2, \dot{\theta}_1, \dot{\theta}_2]^T$ . Notice that  $g_v^c(\theta_2)$  represents a variable term corresponding to the gravity effects of the mass in the second link. This term will be omitted until Section IV, where its effect within the Fault-Tolerant scheme will be discussed.

Quasi-LPV Modelling requires to define a set of variable parameters  $\Psi$  that depend on endogenous signals and linearly define the system. It should be pointed out that multiple  $\Psi$  can be defined for the same system, such that: (1) Parameter  $\Psi_i(x_1, x_2, \dots)$  has to be bounded for the (bounded) set of variables  $x_1, x_2, \dots$ , (2) controllability property is assured for the system, and (3) the minimal set that fulfill previous conditions is chosen. For the head subsystem considered in this paper, the Non-linear

Embedding technique [9] has been chosen, and regarding conditions from above,

$$\Psi_1 = \mathcal{C}_{(2,1)}(\theta_2) \dot{\theta}_1, \quad \Psi_2 = \mathcal{M}_{(1,1)}(\theta_2). \quad (4)$$

A commonly used representation in the literature for LPV systems is to confine all the possible trajectories of  $\Psi$  in a polytope  $\Theta$  defined by a set of vertex  $\omega_{n_\Psi}$  [10] :

$$\Psi(t) \in \Theta := \text{Co}\{\omega_1, \dots, \omega_{n_\Psi}\} \quad (5)$$

being  $n = 1, \dots, 2^{n_\Psi}$ , where  $n_\Psi$  is the number of parameters of  $\Psi$ . Thus, the state-space matrices (in this case, only  $A$  and  $B$ ) are confined in the polytope defined by their image at these vertex:

$$[A^c(\Psi) \ B^c(\Psi)] \in \text{Co}\left\{[A_n^c \ B_n^c] := [A^c(\omega_n) \ B^c(\omega_n)]\right\} \quad (6)$$

In this case  $n_\Psi = 2$ , so the polytopic system would have four vertices. However, as  $B^c$  only depends on  $\Psi_2$ , this matrix will only have two different representations.

Using this approach, the polytopic representation of the system at any instant  $t$  can be stated as a weighted function of their images on the vertex of the polytope as follows:

$$\dot{x}(t) = \sum_{n=1}^{2^{n_\Psi}} \mu_n(\Psi(t)) [A_n^c(\Psi) \cdot x(t) + B_n^c(\Psi_2)u(t)] \quad (7)$$

where  $\mu_n(\Psi)$  are the polytopic coefficients, such that:

$$\sum_{n=1}^{2^{n_\Psi}} \mu_n(\Psi(t)) = 1, \quad \mu_n(\Psi(t)) \geq 0 \quad (8)$$

Each  $\Psi_i \in [\underline{\Psi}_i, \overline{\Psi}_i]$ , and so the vertex will be defined for these limits. Thus,  $\mu_i(\Psi)$  can be defined :

$$\mu_n(\Psi) = \prod_{m=1}^{n_\Psi} \xi_{nm}(\eta_0^m, \eta_1^m), \quad (9)$$

where

$$\eta_0^m = \frac{\overline{\Psi}_m - \Psi_m(t)}{\overline{\Psi}_m - \underline{\Psi}_m}, \quad \eta_1^m = 1 - \eta_0^m \quad (10)$$

Recalling that  $\Psi$  is function of state variables, their limits can be straightforward defined by considering the range for the system's variables they depend on.

Henceforth, procedures and system's state-space form have been formulated in the discrete-time domain, in order to bring the implementation on the real platform together with the approach. All of them are equivalent for the continuous-time form, only with the difference that the system's matrices representation has been evaluated according to a discretization method (e.g. Zero Order Hold) for a sampling time  $T_s$ .

### III. STATE AND FAULT ESTIMATION

A robust and fault-tolerant design for a service robot has to consider the entropy and variability associated to those environments where the robot shares its workspace with humans or even has to collaborate with them, where usually there does not exist a complete information model, or its partially observable. Furthermore, own's robot internal state is sometimes unknown or inaccurate due to a lack of measurements or a low-reliability estimation of them based on measured ones.

#### A. Robust Unknown Input Observer

To tackle this challenge, an initial step has been made by means of a Robust Observer for Unknown Inputs (RUIO for short), which presents an state-observer structure that decouples the effects of possible unknown input disturbances that might affect the system.

Assuming the following LPV model, using the notation described in Section II-B for the polytopic representation:

$$x(k+1) = \sum_{n=1}^{2^{n_\Psi}} \mu_n(\Psi) [A_n x(k) + B_n u(k) + R_n f(k)] \quad (11a)$$

$$y(k) = Cx(k) + Ff(k) \quad (11b)$$

where  $f(k) \in \mathbb{R}^{n_f}$  stands for the unknown input vector, being  $n_f$  the number of unknown inputs considered. Matrices  $R_n \in \mathbb{R}^{n_x \times n_f}$  and  $F \in \mathbb{R}^{n_y \times n_f}$  represent the influence of  $f(k)$  in the system behaviour. These matrices are determined according to the desired scenario to represent, and provide the existence of a solution as described in [8].

The structure for the RUIO is:

$$z(k+1) = \sum_{n=1}^{2^{n_\Psi}} \mu_n(\Psi) [N_n z(k) + G_n u(k) + L_n y(k)] \quad (12a)$$

$$\hat{x}(k) = z(k) - Ey(k) \quad (12b)$$

where  $z(k)$  corresponds to RUIO's state vector, with the same dimensions of  $x(k)$ . Matrices  $N_n$ ,  $G_n$ ,  $L_n$  and  $E$  are the observer gains to be designed, by assuring an asymptotic stability of the dynamics of the observer. So, the estimation error converges to zero as time tends to infinite, disregarding the magnitude of the unknown inputs and disturbances. This behaviour is satisfied if there exist matrices  $X_n > 0$ ,  $S, V$  and  $W_i$  such that the following Linear Matrix Inequalities and Equalities (LMI, LME) conditions hold:

$$\begin{bmatrix} \varphi_n & -V - A_n^T(V + SC)^T - C^T W_n^T \\ (*) & X_n - V - V^T \end{bmatrix} < 0, \quad (13a)$$

$$(V + SC)R_n = W_n F, \quad (13b)$$

$$(V + SC)H_n = W_n J, \quad (13c)$$

$$S[F \ J] = 0. \quad (13d)$$

where  $\varphi_n = -X_n + (V + SC)A_n - W_n C + A_n^T(V + SC)^T - C^T W_n^T$ , and  $(*)$  denotes the transpose of the element in

the symmetric position. The solution for these matrices define RUIO's gains:

$$E = V^{-1}S, \quad (14a)$$

$$G_n = (I + V^{-1}SC)B_n, \quad (14b)$$

$$N_n = (I + V^{-1}SC)A_n - V^{-1}W_nC, \quad (14c)$$

$$L_n = V^{-1}W_n - N_nE. \quad (14d)$$

### B. Fault concept and RUIO design

In this paper, special focus has been put on those environments where both the tasks and workspace might be shared with humans. Therefore, faults have been considered as external unknown forces applied on the system as a possible consequence of a contact with a human, changing its behaviour in an unpredictable way, such that desired configurations are not reached. These forces will exert certain torques in the rotational joints of the head subsystem, i.e. the unknown input term  $f(k)$  from Eqs. (11). Thus, each  $R_n$  has to define this relation for each joint torque component  $f_i$  in the state-space form, assuming this behaviour does not depend on the point of operation within the polytope  $\Theta$ , i.e.  $R_n \equiv B_n$  for  $n = 1, \dots, 2^{n_\Psi}$ . Matrix  $F$  will be null independently on the measured state variables, as  $R_n$  matrices already describe the complete effect of the faults, and so including the same relation on the output states will be redundant.

Along with the representation of the fault scenario, RUIO matrices have to be defined in order that the existence of a solution is assured. Output matrix  $C$  has been determined such that  $y = [\hat{\theta}_2, \theta_1, \theta_2]^T$ , i.e. the fault scenario includes an incomplete information model which considers  $\hat{\theta}_1$  as not measurable, emphasizing the necessity for a decoupling observer structure. But on this consideration and remaining  $F$  null

$$R_n \triangleq [B_{n,(1,1)} \ B_{n,(2,2)} \ 0 \ 0]^T \quad (15)$$

is the closest definition to the current fault effect that fulfills the existence condition for the output model. In terms of the fault scenario it assumes that the same torque component affects both joints equally, but has been proven to assure the desired decoupling requirements for an scenario with different torque components affecting each joint, as it will be seen on simulation results presented in Section V.

### C. Fault isolation and estimation

The presented RUIO structure is able to decouple the state estimation from the effect of an unknown fault that affects the system. This behaviour is desired for autonomous robotic platforms with a partially complete or inaccurate information model, but also some insights about the fault characteristics could be useful in order to evaluate further measures. Taking as an illustrative example a robotic arm that has a direct contact with a human (i.e. and external force to the robot), information

about the fault could be crucial: in the case of an undesired contact, to avoid injuring the person; otherwise, to gather data that specifies the intentionality of the action.

Therefore, a mechanism to estimate the magnitude and isolate the fault has been implemented by means of a Reference Control structure, using the analytical expressions from Eqs. (2), without any fault (force effects) nor gravity terms being injected. The reference signals for the head subsystem will be also given to the Reference Model, and the same state-feedback control strategy will be used but the RUIO, as its states are available. Reference Control i.e. its a parallel control structure to the one for the TIAGo head on the unaffected nominal model.

The magnitude of the fault  $\hat{f}(k)$  will be obtained as the difference between the Reference Model states  $x_{ref}$  and the estimated ones from the head subsystem  $\hat{x}_{sys}$ :

$$\hat{f}(k) = O_f[x_{ref}^a(k+1) - \hat{x}_{sys}^a(k+1)], \quad (16)$$

where  $O_f$  matrix is defined so  $\hat{f}(k)$  has the components of the fault in both axis. The superindex 'a' points out that the values at  $k+1$  are obtained by means of the discrete-time analytic model of the TIAGo head subsystem, which is considered as a good enough approximation of the evolution of the states.

## IV. FAULT-TOLERANT CONTROL SCHEME

The general Fault-Tolerant Control scheme consists on a State-Feedback strategy, according to the polytopic quasi-LPV modelling of the system. Therefore, the State-Feedback gain  $K_{fb}$  has the form given by:

$$K_{fb}(\Psi(k)) = \sum_{n=1}^{2^{n_\Psi}} \mu_n(\Psi(t))K_n \quad (17)$$

The design problem for  $K_n$  has been stated as a Linear Quadratic Control (LQC) problem for the following discrete-time performance criterion:

$$J = \sum_{k=0}^{\infty} [x(k)^T Q x(k) + u(k)^T R u(k)] < \gamma, \quad (18)$$

where  $Q$  controls convergence speed of the states towards their references and  $R$  limits the required control effort, representing  $\gamma$  the upper bound to be minimised. This problem has been formulated for the LPV model so it assures global stability (according to Lyapunov's theory) and performance conditions for the system by evaluating only the LMIs on the matrix images of the vertex, as the polytope  $\Theta$  where the parameter set  $\Psi$  lies is convex.

Therefore, there exists  $K_n = W_n Y_n^{-1}$  such that the

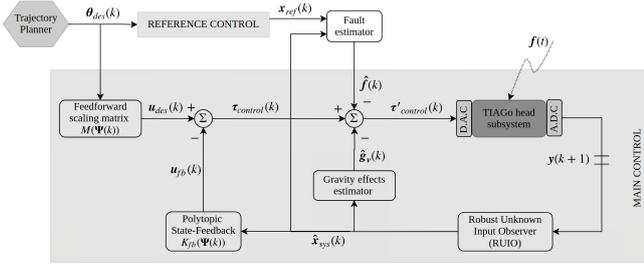


Fig. 2. Schematic representation of the complete Fault-Tolerant control approach on discrete time.

following LMIs hold:

$$\begin{bmatrix} -Y & Y A_n^T - W_n B_n^T & Y(Q^{1/2})^T & W_n^T \\ (*) & -Y & 0 & 0 \\ (*) & (*) & -I & 0 \\ (*) & (*) & (*) & -R^{-1} \end{bmatrix} < 0, \quad (19a)$$

$$\begin{bmatrix} \gamma I & I \\ I & Y \end{bmatrix} < 0. \quad (19b)$$

The complete control scheme is presented in Figure 2. At each time instant  $k$ , the State Feedback action  $u_c(k)$  will be obtained from the estimated states from the RUIO  $\hat{x}_{sys}(k)$  as  $u_c(k) = K_{fb}\hat{x}_{sys}(k)$ . Reference pose  $\theta_{des}(k)$  is given by an external trajectory planner component, and the corresponding control action  $u_{des}(k)$  is computed by means of a Feedforward Scaling matrix  $M(\Psi)$ :

$$M(\Psi(k)) = [C(I + B(\Psi(k))K_{fb} - A(\Psi(t)))^{-1}B(\Psi(k))]^{-1} \quad (20)$$

The control torque for the system's joint actuators is

$$\tau(k) = u_{des}(k) - u_{fb}(k) \quad (21)$$

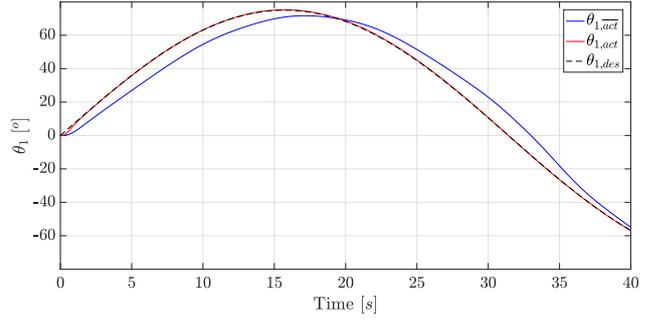
Simultaneously, the Reference Model computes the value of  $x_{ref}(k)$  using the same  $\theta_{des}$  and State-Feedback control strategy, and the estimated fault vector  $\hat{f}(k)$  is obtained. Additionally, an estimation of the gravity effects vector  $\hat{g}_v(k)$  is obtained, and both are injected using an active compensation mechanism, obtaining the compensated control torque:

$$\tau'(k) = \tau(k) - [\hat{g}_v(k) + \hat{f}(k)] \quad (22)$$

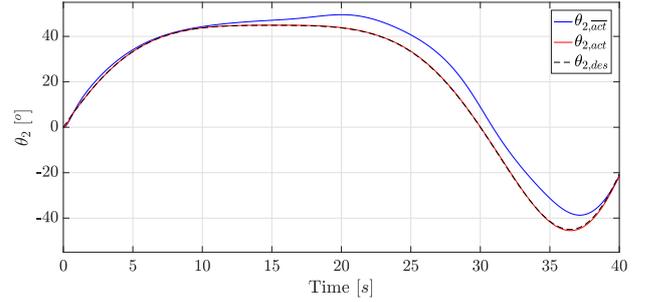
Thus, the State-Feedback plus RUIO control scheme will regard only for the part of control actions related with the nominal operation (i.e. moving to different poses), as the fault and gravity effects will be compensated.

## V. SIMULATION RESULTS

To evaluate the presented Fault-Tolerant control architecture on the head subsystem of the TIAGo Robotic Platform, a simulation has been implemented using the MATLAB programming environment. A continuous-time model has been included according to the analytic expressions (2) and integrated within a discrete-time control loop, using a discretization time  $T_s = 10[ms]$ .



(a)



(b)

Fig. 3. Joint trajectories of (a) Pan and (b) Tilt angles under the fault scenario for the presented approach with and without the active compensation and integral action.

To solve the LMI and LME based design problems of the State-Feedback gain  $K_{fb}$  and RUIO, YALMIP toolbox has been used along with SEDUMI optimisation software.

It should be pointed out that on the design conditions for the RUIO, obtained solutions through the described process did not fulfill the disturbance rejection characteristics required, due to numerical problems on computing  $V^{-1}$  on nearly singular  $V$  matrices. Therefore, the design problem has been reformulated to avoid this inversion, having a Bilinear Matrix Inequality and Equality (BMI, BME) problem, that has been solved using PENBMI solver from TOMLAB. Also, additional LMIs regarding pole placement [10] have been included to avoid couplings between RUIO and the subsystem dynamics, and matrices  $Q$  and  $R$  for the LQR State-Feedback determined for a zero overshooting and non-oscillatory response.

On early simulation results, it has been seen that the active fault mechanism compensation and State-Feedback structure did not assure zero position error on certain fault scenarios for given  $\theta_{des}$ . Therefore, an extension on the state-space model has been performed to include an integral action on angular positions. Presented polytopic approach is equivalent for this extended state-space vector, being held controllability property of the complete system.

The simulated fault scenario consists on a force exerted on the endpoint of the second link, with a time-variant magnitude  $|\mathcal{F}|$  and orientation  $\mathcal{F}^\phi$  around the vertical axis

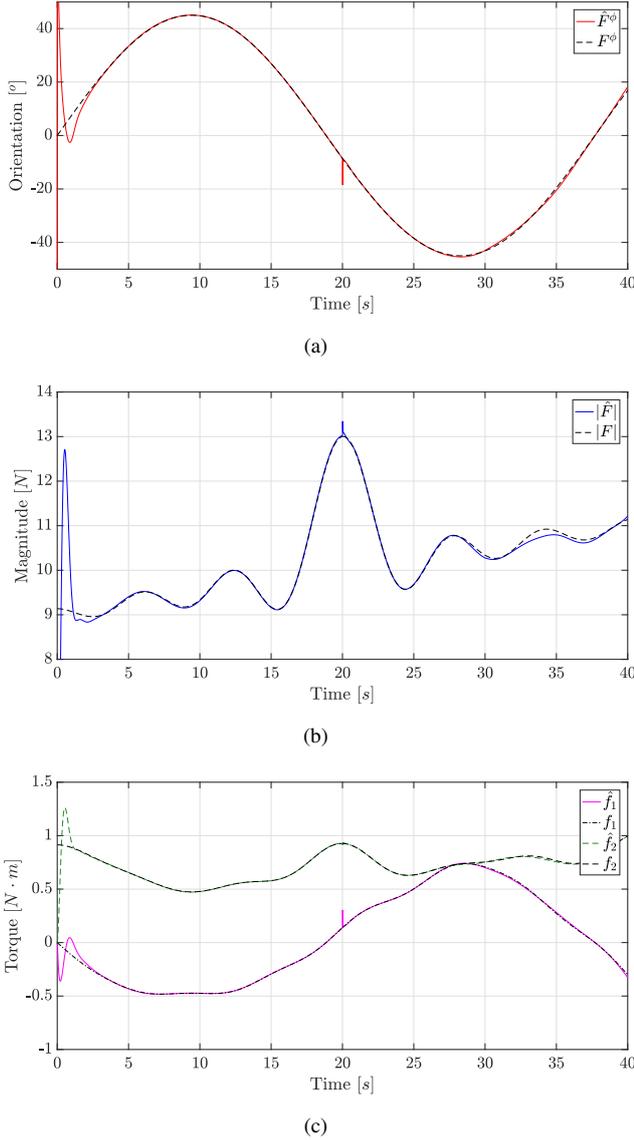


Fig. 4. External fault force and its estimated (a) Orientation, (b) Magnitude and (c) exerted torques.

of the manipulator base:

$$|\mathcal{F}|(t) = 1 + 5 \cdot 10^{-2}(t - 20) + 3 \frac{\sin(t - 20)}{(t - 20)} \quad (23a)$$

$$\mathcal{F}^\phi(t) = 45 \sin(t/6) \quad (23b)$$

The exerted torques on the joints will be also function of  $\theta_2$ , as the force will be restrained to a plane perpendicular to the ground one. Trajectories for  $\theta$  are given considering the joint limits:  $\theta_1 \in [-75, 75]^\circ$  and  $\theta_2 \in [-60, 45]^\circ$ .

Figure 3 presents the simulation results for a trajectory  $\theta_{des}$  for a control structure with the active fault compensation mechanism and integral action, denoted by  $act$  (red), and without them, denoted with  $\overline{act}$  (blue). The evolution of  $|F|$  and  $F^\phi$  and the corresponding exerted torque vector  $f(t)$  (solid line) are shown in Figure 4, along

with the estimated values (dashed lines).

On the presented simulation results, the required torques by both the state-feedback control and the active compensation mechanism are assumed to be within the physical limits of the actuator, so there does not exist any saturation phenomena.

## VI. CONCLUSIONS

This paper has addressed the problem of Fault-Tolerant control of a service robot, focusing on its 2-DOF head subsystem under a scenario where it is affected by unknown external forces. The proposed approach relies on a fault estimation scheme based on a Robust Unknown Input Observer (RUIO) that allows to estimate unmeasurable states, along with a state feedback control strategy. After the fault occurrence, a fault estimator based on a Reference Control structure actively generates a feedforward compensation action. To cope with robot non-linearity, its model is transformed into a quasi-LPV model. State-Feedback and RUIO are designed using an LMI-based approach considering a gain-scheduling scheme. Based on the simulation results, the proposed Fault-tolerant approach has been proved to successfully tackle the presented problem. As future research, the proposed scheme could be tested on the real robotic platform.

## APPENDIX A: TIAGO HEAD MODEL PARAMETERS

Parameter	Value
$I_1$	$\text{diag}(0.001192, 0.001402, 0.000889) [kg \cdot m^2]$
$I_2$	$\text{diag}(0.004620, 0.004861, 0.003132) [kg \cdot m^2]$
$m_1$	0.622 [kg]
$m_2$	0.661 [kg]
$L$	0.098 [m]

## REFERENCES

- [1] International Federation of Robotics (IFR). *Executive Summary World Robotics. 2016 Service Robot*.
- [2] Zhuo-hua D., Zi-xing C., Jin-xia Y. 'Fault diagnosis and fault tolerant control for wheeled mobile robots under unknown environments: A survey'. *Proceedings of the International Conference on Robotics and Automation (ICRA)*. IEEE; 2005.
- [3] Khalastchi E, Kalech M. 'On Fault Detection and Diagnosis in Robotic Systems'. *ACM Computing Surveys*. 2018;51(1):1-24.
- [4] Craig J.J. *Introduction to Robotics. Mechanics and control*. 3rd edn. Pearson Education International; 2005. p. 173.
- [5] Patton R. J. and Klinkhieo S. 'LPV Fault Estimation and FTC of a two-link manipulator'. Presented in American Control Conference; Baltimore, MD, USA, 2010.
- [6] Patton R. J., Lejun C. and Klinkhieo S. 'An LPV Pole-Placement Approach to Friction Compensation as an FTC Problem'. *International Journal of Appl. Math. Comput. Sci.* 2012, Vol. 22(1), pp. 149-160.
- [7] Hornung R., Urbanek H., Klodmann J., Osendorfer C., van der Smagt P. 'Model-free robot anomaly detection'. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)* IEEE; 2014; pp. 3676-3683.
- [8] Chadli M., Karimi H.R. 'Robust Observer Design for Unknown Inputs Takagi-Sugeno Models'. *IEEE Transactions on Fuzzy Systems*. 2013, vol. 21(1), pp. 158-164.
- [9] Kwiatkowski A., Boll M.T., Wener H. 'Automated Generation and Assessment of Affine LPV Models'. *Proceedings of the IEEE Conference on Decision and Control*. IEEE; 2007; 6690 - 6695.
- [10] Rotondo D., *Advances in Gain-Scheduling and Fault Tolerant Control Techniques*. Springer, 2017.