

Reducing the Effects of Inaccurate Fault Estimation in Spacecraft Stabilization

Rouzbeh Moradi¹, Alireza Alikhani¹, Mohsen Fathi Jegarkandi²

ABSTRACT: Reference Governor is an important component of Active Fault Tolerant Control. One of the main reasons for using Reference Governor is to adjust/modify the reference trajectories to maintain the stability of the post-fault system, especially when a series of actuator faults occur and the faulty system can not retain the pre-fault performance. Fault estimation error and delay are important properties of Fault Detection and Diagnosis and have destructive effects on the performance of the Active Fault Tolerant Control. It is shown that, if the fault estimation provided by the Fault Detection and Diagnosis (initial "fault estimation") is assumed to be precise (an ideal assumption), the controller may not show an acceptable performance. Then, it is shown that, if the worst "fault estimation" is considered, it will be possible to reduce the effects of fault estimation error and delay and to preserve the performance of the controller. To reduce the effects of this conservative assumption (worst "fault estimation"), a quadratic cost function is defined and optimized. One of the advantages of this method is that it gives the designer an option to select a less sophisticated Fault Detection and Diagnosis for the mission. The angular velocity stabilization of a spacecraft subjected to multiple actuator faults is considered as a case study.

KEYWORDS: Active Fault Tolerant Control, Fault estimation error and delay, Reference Governor, Angular velocity stabilization.

INTRODUCTION

Active Fault Tolerant Control (AFTC) is an important field in automatic control that has attracted a large amount of attention. The main responsibility of an AFTC is to tolerate component malfunctions while maintaining desirable performance and stability properties of the faulty system (Zhang and Jiang 2008). Latterly, a review paper published recent developments of the spacecraft AFTC system (Yin *et al.* 2016).

One of the main components of any AFTC is the Fault Detection and Diagnosis (FDD) module. There are several challenges that FDD designs have in common (Zhang and Jiang 2008). Among them, fault estimation error and delay are considered in this paper. These challenges have destructive effects on the stability and performance (Zhang and Jiang 2008).

Reference Governor (RG) is one of the components of the general AFTC structure (Zhang and Jiang 2008). The terms Command Governor (CG) and Reference Trajectory Management (RTM) have been also used in the literature. The main responsibility of RG is to adjust/modify the reference trajectories, so the post-fault model of the system remains stable, even after the occurrence of multiple actuator faults (Garone *et al.* 2016). There are several papers in the literature that have studied the effects of RG on the performance and stability of the post-fault model (Boussaid *et al.* 2010; Boussaid *et al.* 2011; Boussaid *et al.* 2014; Almeida 2011). According to these papers, RG has been able to deal with the actuator faults/failures efficiently.

To the authors' best knowledge, reducing the effects of fault estimation error and delay using the concept of RG still remains an open problem. This is the main subject that is pursued in this paper. It is shown that, as long as the estimated fault

¹Ministry of Science, Research and Technology – Aerospace Research Institute – Astronautics Department – Tehran/Tehran – Iran. ²Sharif University of Technology – Engineering College – Department of Aerospace Engineering – Tehran/Tehran – Iran.

Author for correspondence: Alireza Alikhani | Ministry of Science, Research and Technology – Aerospace Research Institute – Astronautics Department | PO box: 14665-834 – Tehran/Tehran – Iran | Email: aalikhani@ari.ac.ir

Received: Oct. 29, 2016 | **Accepted:** Mar. 25, 2017

reported by the FDD (initial “fault estimation”) is assumed to be precise (an ideal assumption), the controller may not show an acceptable performance.

However, if the maximum fault estimation error is considered (worst “fault estimation”), RG can be used to reduce the effects of FDD errors and preserve the performance of the closed-loop system. To reduce the effects of this conservative assumption (considering maximum fault estimation error), a quadratic cost function is defined and optimized.

In order to validate the results, the angular velocity stabilization of a spacecraft subjected to multiple actuator faults is considered. It is shown that, if the initial “fault estimation” (the fault estimation reported by the FDD) is considered accurate, the response will not converge to the origin. However, if RG is designed based on the worst “fault estimation”, AFTC will be able to asymptotically stabilize the faulty spacecraft in a wide range of actuator fault and despite FDD errors. This paper consists of the following sections: firstly, the modeling of the proposed RG is described. Then, the spacecraft dynamics and controller are shown. Finally, results obtained and the discussions are presented.

MODELING THE REFERENCE GOVERNOR

The structure of the considered AFTC is shown in Fig. 1. It is assumed that the FDD block provides “an estimation of” the post-fault model of the system. The RG block uses the proposed methodology to find the most suitable reference trajectories for the post-fault model, despite the presence of fault estimation error and delay. The signals ω and ω_d are the plant output (angular velocity) and the desired reference trajectory vectors, respectively.

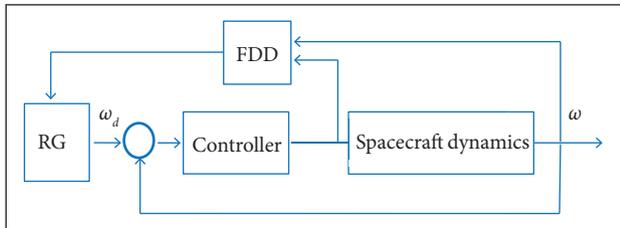


Figure 1. Structure of the AFTC.

In this paper, the mission of the controller is to make the origin an asymptotically stable equilibrium for the post-fault system, i.e. $\omega \rightarrow 0$ as $t \rightarrow t_f$ (final time).

It is assumed that the actuator fault/failure occurs at $t = t_{fault}$ and the FDD determines \hat{t}_{fault} (estimated t_{fault}) with a fault estimation delay equal to:

$$\delta t_{fault} = \hat{t}_{fault} - t_{fault} \tag{1}$$

which is a positive value, since \hat{t}_{fault} is always bigger than t_{fault} .

Fault estimation error is another property of the considered FDD block. The control inputs are bounded according to the following saturation function:

$$sat(u_i) = \begin{cases} u_i & \text{if } -u_{max} \leq u_i \leq u_{max} \\ u_{max} & \text{if } u_i \geq u_{max} \\ -u_{max} & \text{if } u_i \leq -u_{max} \end{cases} \tag{2}$$

where u_{max} is the maximum torque that can be produced by the actuators.

The reduction in the actuator region is considered as the actuator fault and is modeled according to Eq. 3 (Miksch and Gambier 2011):

$$sat_{p-f}(u_i) = \begin{cases} u_i & \text{if } -u_{max_{p-f}} \leq u_i \leq u_{max_{p-f}} \\ u_{max_{p-f}} & \text{if } u_i \geq u_{max_{p-f}} \\ -u_{max_{p-f}} & \text{if } u_i \leq -u_{max_{p-f}} \end{cases} \tag{3}$$

The subscript $p-f$ shows the post-fault condition. The relation between pre- and post-fault actuator region is given according to:

$$u_{max_{p-f}} = a u_{max} \quad a \in (0 \ 1) \tag{4}$$

where a is the actuator effectiveness coefficient (Sobhani-Tehrani and Khosravi 2009), a real value between 0 and 1; u_{max} is the pre-fault actuator region. FDD determines the estimated value of a (shown by \hat{a}). It is assumed that the FDD provides \hat{a} with an estimation error given by:

$$\delta_{a/\hat{a}} = \frac{a}{\hat{a}} \tag{5}$$

where $\delta_{a/\hat{a}}$ is a value between 0 and 1. The larger/smaller values of $\delta_{a/\hat{a}}$ show better/worse fault estimation, respectively.

According to the considered mission, the goal of RG is to determine ω_d such that the faulty model of the system remains asymptotically stable, even after the occurrence of multiple actuator faults and in the presence of fault estimation error and delay

in the FDD module. The RG flowchart is presented in Fig. 2. The consecutive steps are explained in the following paragraphs.

According to Fig. 3, $\omega_d(t_1) \dots \omega_d(t_n)$ are initialized by the solver, which is the Genetic Algorithm (GA), as will be explained in the results section.

Note 1: although the GA is used to solve the problem, other numerical solvers can be also employed. However, the main concern of this paper is to find a method to decrease the consequences of fault estimation error and delay. Therefore, any numerical solver (possibly faster than GA) that solve the problem can be considered as well.

Note 2: as will be seen in the simulation section, GA can find a solution within a reasonable time.

When these points are determined, a cubic spline is passed through them, similarly to Fig. 4. A detailed analysis about cubic

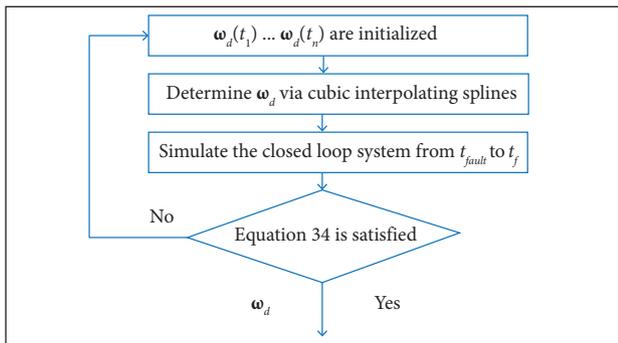


Figure 2. RG flowchart.

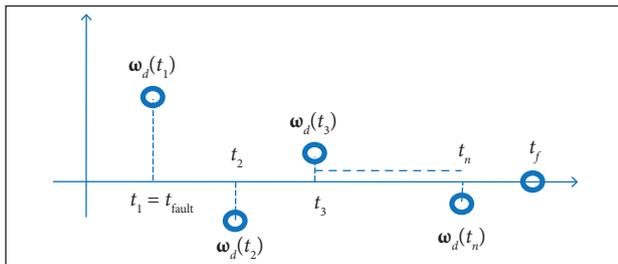


Figure 3. Initializing $\omega_d(t_1) \dots \omega_d(t_n)$.

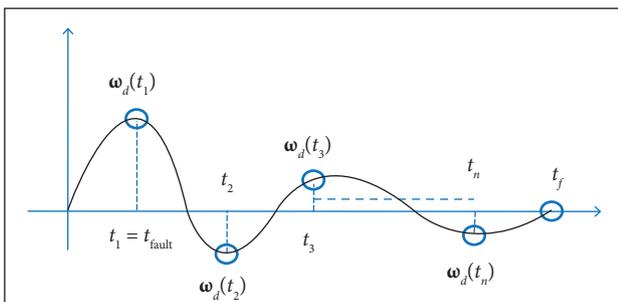


Figure 4. ω_d produced by cubic spline.

spline interpolation can be found in de Boor (1978). One of the main advantages of cubic splines is their smoothness (they are twice continuously differentiable). This will prevent the controller inputs from being discontinuous (refer to Eqs. 25 – 27).

According to the FDD information, an estimation of the post-fault model of the system is known. The faulty closed-loop system is simulated from t_{fault} to t_f . This simulation is a part of the flowchart shown in Fig. 2 and several simulations may be needed to obtain ω_d .

After simulation, the value of $\omega(t_f)$ is checked to see whether the following equality is satisfied or not:

$$\omega(t_f) = \mathbf{0} \tag{6}$$

Such a final state constraint is well-known in the literature and is introduced to ensure asymptotic stability (Fontes 2001). Since this equality will never hold numerically, Eq. 34 will be considered in simulations.

Note 3: to ensure that ω_d approaches the origin before $t = t_f$ its value is set to 0 as t passes t_s (settling time). In other words:

$$\omega_d = \mathbf{0} \forall t \geq t_s, t_s < t_f \tag{7}$$

To give the solver more flexibility, another variable (k_s) is introduced, satisfying Eq. 8:

$$t_s = k_s t_f, 0 < k_s < 1 \tag{8}$$

In addition to $\omega_d(t_1) \dots \omega_d(t_n)$, k_s is another variable that should be found by the solver.

SPACECRAFT DYNAMICS AND CONTROLLER STRUCTURE

SPACECRAFT DYNAMICS

The rigid body spacecraft rotational dynamics in the principal coordinate system is described by the following equations (Sidi 2000):

$$\dot{\omega}_1 = \alpha_1 \omega_2 \omega_3 + u'_1 \quad \alpha_1 = \left(\frac{J_2 - J_3}{J_1} \right) \tag{9}$$

$$\dot{\omega}_2 = \alpha_2 \omega_1 \omega_3 + u'_2 \quad \alpha_2 = \left(\frac{J_3 - J_1}{J_2} \right) \tag{10}$$

$$\dot{\omega}_3 = \alpha_3 \omega_1 \omega_2 + u'_3 \quad \alpha_3 = \left(\frac{J_1 - J_2}{J_3} \right) \quad (11)$$

where $\omega_1, \omega_2, \omega_3$ are the angular velocities; u'_1, u'_2, u'_3 are the normalized control inputs; J_1, J_2, J_3 are the principal moments of inertia of the rigid body. The relation between control torques and inputs are given by Eqs. 12 – 14:

$$u'_1 = u_1 / J_1 \quad (12)$$

$$u'_2 = u_2 / J_2 \quad (13)$$

$$u'_3 = u_3 / J_3 \quad (14)$$

where u_1, u_2, u_3 are the control moments acting on the spacecraft.

CONTROLLER STRUCTURE

The error signal is defined as:

$$\omega_e = \omega - \omega_d \quad (15)$$

where ω_d and ω_e are the desired and error angular velocity vectors, respectively.

Inserting the scalar form of Eq. 15 into Eqs. 9 – 11 and eliminating ω , one has:

$$\dot{\omega}_{1e} = \dot{\omega}_1 - \dot{\omega}_{1d} = \alpha_1 (\omega_{2e} + \omega_{2d})(\omega_{3e} + \omega_{3d}) + u'_1 - \dot{\omega}_{1d} = u''_1 \quad (16)$$

$$\dot{\omega}_{2e} = \dot{\omega}_2 - \dot{\omega}_{2d} = \alpha_2 (\omega_{1e} + \omega_{1d})(\omega_{3e} + \omega_{3d}) + u'_2 - \dot{\omega}_{2d} = u''_2 \quad (17)$$

$$\dot{\omega}_{3e} = \dot{\omega}_3 - \dot{\omega}_{3d} = \alpha_3 (\omega_{1e} + \omega_{1d})(\omega_{2e} + \omega_{2d}) + u'_3 - \dot{\omega}_{3d} = u''_3 \quad (18)$$

Canceling the non-linear terms using feedback linearization, the closed-loop system will change into the following simple linear time invariant form:

$$\dot{\omega}_{1e} = u''_1 \quad (19)$$

$$\dot{\omega}_{2e} = u''_2 \quad (20)$$

$$\dot{\omega}_{3e} = u''_3 \quad (21)$$

and the following form of control inputs

$$u''_1 = -k_1 \omega_{1e} \quad k_1 \in R^+ \quad (22)$$

$$u''_2 = -k_2 \omega_{2e} \quad k_2 \in R^+ \quad (23)$$

$$u''_3 = -k_3 \omega_{3e} \quad k_3 \in R^+ \quad (24)$$

will lead to the exponential stabilization of ω_e to 0; consequently, ω will converge to ω_d exponentially. The numerical values of k_1, k_2 and k_3 determine the exponential convergence rate of ω_e to 0. Therefore, larger values of k_1, k_2 and k_3 mean a faster response and vice-versa.

Considering Eqs. 16 – 18 and Eqs. 22 – 24, the following relations will be obtained:

$$u'_1 = -\alpha_1 (\omega_{2e} + \omega_{2d})(\omega_{3e} + \omega_{3d}) + \dot{\omega}_{1d} - k_1 \omega_{1e} \quad (25)$$

$$u'_2 = -\alpha_2 (\omega_{1e} + \omega_{1d})(\omega_{3e} + \omega_{3d}) + \dot{\omega}_{2d} - k_2 \omega_{2e} \quad (26)$$

$$u'_3 = -\alpha_3 (\omega_{1e} + \omega_{1d})(\omega_{2e} + \omega_{2d}) + \dot{\omega}_{3d} - k_3 \omega_{3e} \quad (27)$$

For feedback purposes, it is better to rewrite u'_1, u'_2 and u'_3 as a function of the original variables:

$$u'_1 = -\alpha_1 (\omega_2)(\omega_3) + \dot{\omega}_{1d} - k_1 (\omega_1 - \omega_{1d}) \quad (28)$$

$$u'_2 = -\alpha_2 (\omega_1)(\omega_3) + \dot{\omega}_{2d} - k_2 (\omega_2 - \omega_{2d}) \quad (29)$$

$$u'_3 = -\alpha_3 (\omega_1)(\omega_2) + \dot{\omega}_{3d} - k_3 (\omega_3 - \omega_{3d}) \quad (30)$$

According to Eqs. 28 – 30, for the control inputs to be continuous, the desired reference trajectory (ω_d) should be continuously differentiable. As stated previously, this is one of the main reasons for using cubic spline interpolation to find ω_d . These are the desired control inputs that will lead to the exponential convergence of ω to ω_d .

If $\omega_d = 0$, the equations of closed-loop system will be:

$$\dot{\omega}_1 = -k_1 \omega_1 \quad (31)$$

$$\dot{\omega}_2 = -k_2 \omega_2 \quad (32)$$

$$\dot{\omega}_3 = -k_3 \omega_3 \quad (33)$$

Clearly, as long as there is no saturation and the actuators can produce the required control inputs, will remain globally exponentially stable (GES). However, after the occurrence of severe actuator faults, GES will not be guaranteed.

RESULTS

The system/controller parameters and initial conditions are given in Table 1. The values chosen for the moments of inertia are taken from Wang *et al.* (2013), and the range of variables is presented in Table 2.

Table 1. System/controller parameters and initial conditions

Controller parameters	Initial conditions (deg/s)	Moments of inertia (kg·m ²)
$k_1 = 0.1$	$\omega_1(0) = 10$	$J_1 = 449.5$
$k_2 = 0.1$	$\omega_2(0) = -10$	$J_2 = 449.5$
$k_3 = 0.1$	$\omega_3(0) = 5$	$J_3 = 449.5$

Table 2. Range of variables.

Optimization variable	Range
ω_d	[-100 100] deg/s
k_s	[0.5 0.9]

In order to satisfy the final state constraint given by Eq. 6, the following inequality is defined:

$$\sum_{i=1}^3 \omega_i^2(t_f) \leq 0.0001, \omega: deg/s \tag{34}$$

As already mentioned, to determine ω_d , GA (Goldberg 1989) is used as the solver; $[\omega_{1d}(t_1) \dots \omega_{1d}(t_n)]$, $[\omega_{2d}(t_1) \dots \omega_{2d}(t_n)]$ and $[\omega_{3d}(t_1) \dots \omega_{3d}(t_n)]$ are initialized every 10 s ($\Delta t = 10$ s or equivalently, $n = 10$) from the beginning of the fault time (t_{fault}). Therefore, considering k_s , the total number of decision variables will be 31. The considered parameters for GA are presented in Table 3. Other GA parameters are the default values considered in MATLAB® (MathWorks® 2011).

The actuation system consists of 6 thrusters (without considering hardware redundancy), that are placed in opposite directions, and each thruster can produce maximum 50 N variable thrust. The effective moment arm of all thrusters is 1 m along the principal body axis. However, the configuration of the thrusters is such that $(T_1 - T_2)$, $(T_3 - T_4)$ and $(T_5 - T_6)$ produce net moments about the first, second and third principal axes,

Table 3. GA parameters.

Parameter	Value
Cross-over fraction	0.8
Elite count	2
Population size	$5 \times$ number of decision variables = $5 \times 31 = 155$
Initial population	$\omega_{d,initial} = 0, k_{s,initial} = 0$

respectively. The direction of the arrows shows the direction of the forces produced by the thrusters (Fig. 5). Therefore, the relation between control torques (u_1, u_2, u_3) and $T_1 - T_6$ can be obtained according to the following equations:

$$u_1^+ = T_1, u_1^- = -T_2 \tag{35}$$

$$u_2^+ = T_3, u_2^- = -T_4 \tag{36}$$

$$u_3^+ = T_5, u_3^- = -T_6 \tag{37}$$

where the superscripts + and - show the positive and negative control torques, respectively.

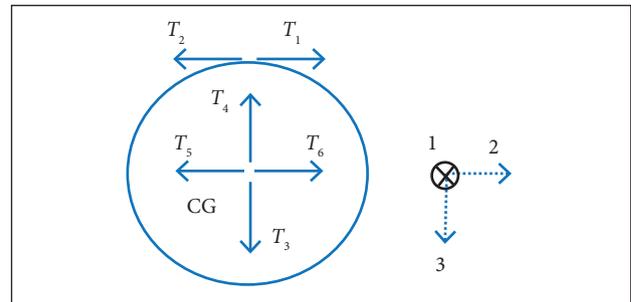


Figure 5. Thruster configuration.

Note 4: it seems that the thrusters T_3, T_4, T_5 and T_6 pass through the center of gravity. However, as indicated before, they have a moment arm of 1 m along the first body axis. Three important concepts are introduced:

- Initial “fault estimation”: the fault estimation reported by the FDD.
- Worst “fault estimation”: the biggest error of the FDD in providing the fault information. Its value is determined from the initial “fault estimation”, according to the experience or the FDD specifications.
- Real fault: the fault that happens in reality (unknown).

The fault scenario that FDD reports is:

- Initial “fault estimation”: T_5 and T_6 have lost 99% of their effectiveness ($\hat{a}_5 = \hat{a}_6 = 0.01$) and the remaining thrusters are at a good health ($\hat{a}_1 = \hat{a}_2 = \hat{a}_3 = \hat{a}_4 = 1$). The fault occurs at $\hat{t}_{fault} = 10$ s.
- Worst “fault estimation”: based on the experience or the FDD specifications; in the worst case, the following parameters are given: $\delta t_{fault} = 5$ s and $\delta_{a/\hat{a}} = 0.01$. Therefore, it can be concluded that, in the worst case, $a_5 = a_6 = 0.0001$, i.e. T_5 and T_6 can produce a maximum 0.05 N thrust and the fault occurrence time is $t_{fault} = 5$ s.

Note 5: it is assumed that the real fault is less severe than the one reported by the worst “fault estimation”. In this case, the controller will show an acceptable performance for less severe, and therefore, a wide range of faults.

Qualitatively, it is assumed that the severity of the faults satisfies the following inequalities:

$$S_{w.f.e} > S_{r.f} > S_{i.f.e} \tag{38}$$

where S is a quality that represents the severity of the fault; the subscripts $w.f.e$, $r.f$ and $i.f.e$ stand for worst “fault estimation”, real fault and initial “fault estimation”, respectively.

According to the previous discussion, the proposed method is very conservative, because it considers the worst “fault estimation”. To reduce the adverse effects of this assumption, the following quadratic cost function is introduced:

$$J = \int_{t_{fault}}^{t_{final}} \omega^T \omega dt + \int_{t_{fault}}^{t_{final}} u^T u dt \tag{39}$$

Minimizing this cost function will decrease the adverse effects of considering the worst fault estimation. The considered sample time for integration is 0.1 s. The problem consists of 2 phases: first, GA tries to satisfy the constraint given by Eq. 34. Then, the result is used as an initial solution to optimize Eq. 39. The following penalty on cost function is considered:

$$J = \begin{cases} \int_{t_{fault}}^{t_{final}} \omega^T \omega dt + \int_{t_{fault}}^{t_{final}} u^T u dt & \text{Eq. 34 is satisfied} \\ \infty & \text{Eq. 34 is not satisfied} \end{cases} \tag{40}$$

It was verified that 1,000 s elapsed time is considered as the stopping criterion for the second phase — Intel(R) Core™

2 CPU, T7200@2.00 GHz; MATLAB® (MathWorks® 2011). To observe the consequences of employing the proposed method, 2 different cases are considered and summarized in Table 4.

Table 4. Cases considered.

Case	Fault estimation
1	Considering the initial “fault estimation”
2	Considering the worst “fault estimation”

CASE 1

If the initial “fault estimation” is considered (FDD is assumed to report the precise fault information), the results shown in Figs. 6 and 7 will be obtained.

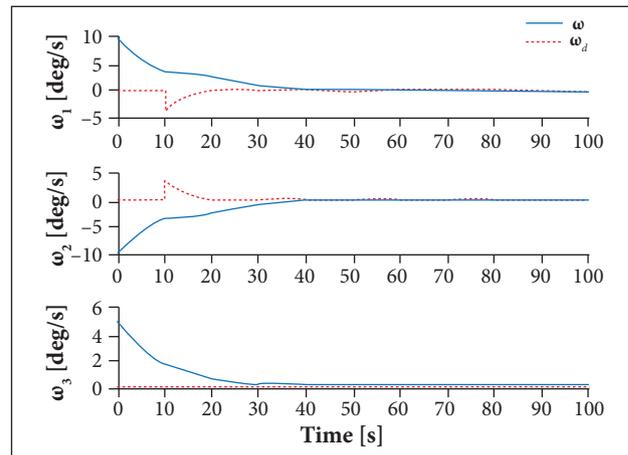


Figure 6. Angular velocities, initial “fault estimation” (case 1).

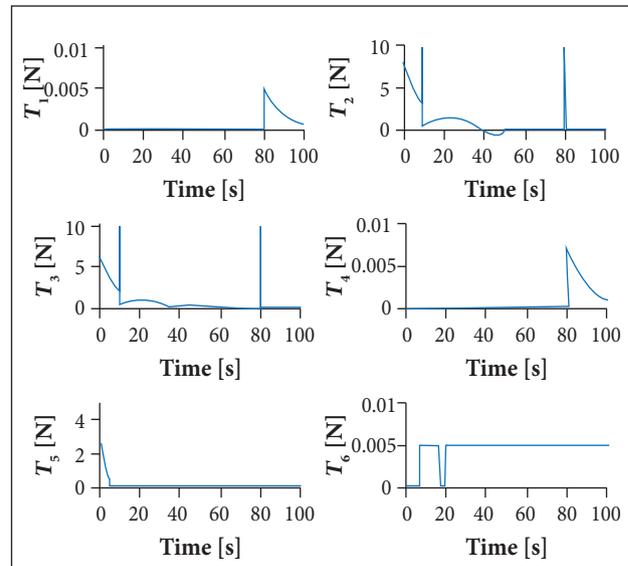


Figure 7. Control inputs, initial “fault estimation” (case 1).

Figure 6 shows that RG can not make the closed-loop system asymptotically stable, because it assumes the fault scenario reported by the FDD (initial “fault estimation”), which is precise. However, since the real fault is worse than the fault reported by the FDD (initial “fault estimation”), does not converge to the origin. This simulation shows the consequences of considering the initial “fault estimation”. The main conclusion of this simulation is: if the FDD is assumed to report the precise fault information, the response of the controller may not be acceptable.

CASE 2

The result of considering the worst “fault estimation” is illustrated in Fig. 8. The control inputs are illustrated in Fig. 9.

According to Fig. 8, RG can asymptotically stabilize the closed-loop system, when the worst “fault estimation” is considered. A comparison of Figs. 6 and 8 shows the consequences of considering the worst “fault estimation” in the RG design. Clearly, considering the initial “fault estimation” (case 1) can lead to the poor performance of the controller and even to a non-convergent response. On the other hand, if RG is designed for the worst “fault estimation” (case 2), it can cover less severe faults and stabilize the faulty system for a wide range of faults (Note 5).

Since the assumption of worst “fault estimation” is conservative, the response is optimized via minimizing the cost function (Eq. 39). The GA performance is illustrated in Fig. 10. As stated previously, the quadratic cost function has been introduced to reduce the adverse consequences of considering the worst “fault estimation” (maximum fault estimation error). According to Fig. 10, after 14 generations (1,000 s elapsed time), the cost function is reduced from 8,758 to 5,944

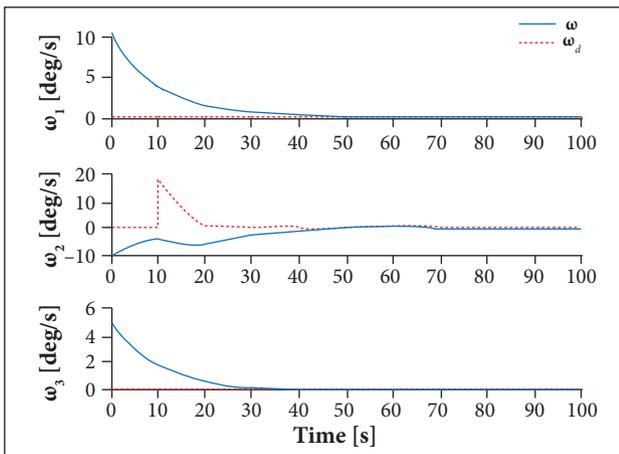


Figure 8. Angular velocities, worst “fault estimation” (case 2).

(about 32%). This reduction in the cost function decreases the adverse consequences of considering the worst fault estimation.

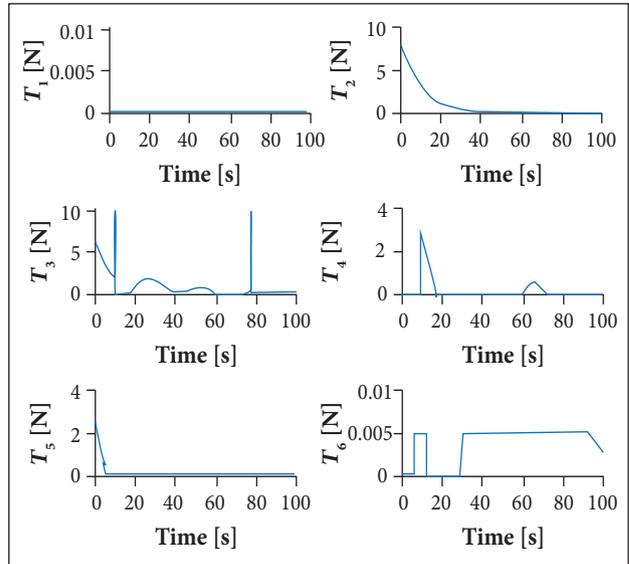


Figure 9. Control inputs, worst “fault estimation” (case 2).

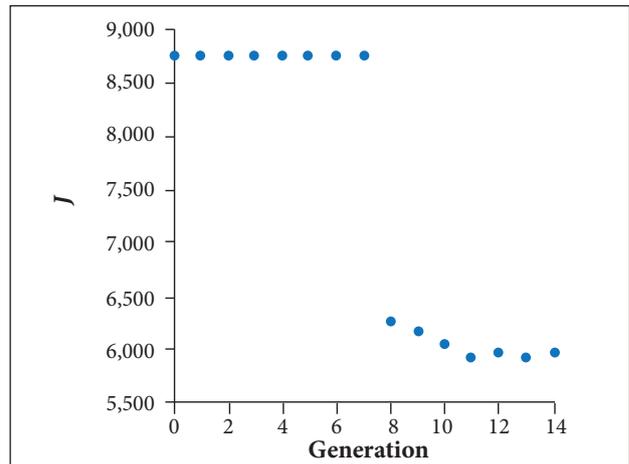


Figure 10. Cost function versus generations (1,000 s elapsed time).

DISCUSSION

Fault estimation error and delay are important characteristics of FDD schemes. RG is a method to adjust/modify the reference trajectories to handle actuator fault/failure. It was shown that, if the initial “fault estimation” was assumed to be precise (an ideal assumption), the controller might not be able to show an acceptable performance. On the other hand, if the worst “fault estimation” was considered, it

would be possible to reduce the destructive effects of fault estimation error. A quadratic cost function was defined to reduce the adverse consequences of this conservative assumption (assuming maximum fault estimation error). Therefore, a less sophisticated FDD can be used to satisfy the mission objectives.

AUTHOR'S CONTRIBUTION

Conceptualization, Moradi R; Methodology, Moradi R, Alikhani A, and Fathi Jegarkandi M; Writing – Original Draft, Moradi R and Alikhani A; Writing – Review & Editing, Moradi R, Alikhani A, and Fathi Jegarkandi M.

REFERENCES

- Almeida FA (2011) Reference management for fault-tolerant model predictive control. *J Guid Control Dynam* 34(1):44-56. doi: 10.2514/1.50938
- Boussaid B, Aubrun C, Abdelkrim MN (2010) Fault adaptation based on reference governor. *Proceedings of the Conference on Control and Fault-Tolerant Systems*; Nice, France.
- Boussaid B, Aubrun C, Abdelkrim MN (2011) Two-level active fault tolerant control approach. *Proceedings of the 8th International Multi-Conference on Systems, Signals and Devices*; Sousse, Tunisia.
- Boussaid B, Aubrun C, Jiang J, Abdelkrim MN (2014) FTC approach with actuator saturation avoidance based on reference management. *International Journal of Robust and Nonlinear Control* 24(17):2724-2740. doi: 10.1002/mc.3020
- De Boor C (1978) *A practical guide to splines*. Berlin: Springer.
- Fontes FACC (2001) A general framework to design stabilizing nonlinear model predictive controllers. *Systems and Control Letters* 42(2):127-143. doi: 10.1016/S0167-6911(00)00084-0
- Garone E, Di Cairano S, Kolmanovsky IV (2016) Reference and command governors for systems with constraints: A survey on theory and applications. *Automatica* 75:306-328. doi: 10.1016/j.automatica.2016.08.013
- Goldberg DE (1989) *Genetic algorithms in search, optimization & machine learning*. Reading: Addison-Wesley.
- MathWorks® (2011) *MATLAB® and SIMULINK®*. Natick: MathWorks®.
- Miksch T, Gambier A (2011) Fault-tolerant control by using lexicographic multi-objective optimization. *Proceedings of the 8th Asian control conference (ASCC)*; Kaohsiung, Taiwan.
- Sidi MJ (2000) *Spacecraft dynamics and control: a practical engineering approach*. Cambridge: Cambridge University Press.
- Sobhani-Tehrani E, Khosravi KH (2009) *Fault diagnosis of nonlinear systems using a hybrid approach*. *Lecture Notes in Control and Information Sciences*. Dordrecht; New York: Springer.
- Wang D, Jia Y, Jin L, Xu S (2013) Control analysis of an underactuated spacecraft under disturbance. *Acta Astronautica* 83:44-53. doi: 10.1016/j.actaastro.2012.10.029
- Yin S, Xiao B, Ding S, Zhou D (2016) A review on recent development of spacecraft attitude fault tolerant control system. *IEEE Trans Ind Electron* 63(5):3311-3320. doi: 10.1109/TIE.2016.2530789
- Zhang Y, Jiang J (2008) Bibliographical review on reconfigurable fault-tolerant control systems. *Ann Rev Contr* 32(2):229-252. doi: 10.1016/j.arcontrol.2008.03.008