Article

Not peer-reviewed version

Progressive Optimal Fault-Tolerant Control Combining Active and Passive Control Manners

Dan Du , Zetao Li * , B. Dahhou

Posted Date: 15 March 2024

doi: 10.20944/preprints202403.0920.v1

Keywords: passive fault-tolerant control; active fault-tolerant control; progressive optimal fault-tolerant control; guaranteed robust cost control

Article

Progressive Optimal Fault-Tolerant Control Combining Active and Passive Control Manners

Dan Du [1], Zetao Li [1,2,*] and B. Dahhou [3]

1 Electrical Engineering College, Guizhou University, Guiyang, 550025 China; imdudan@163.com
2 Key Laboratory of "Internet+" Collaborative Manufacture in Guizhou Provence, Guizhou University, Guiyang, 550025 China; gzgylzt@163.com
3 University de Toulouse, UPS, LAAS, F-31400 Toulouse, France; boutaib.dahhou@laas.fr
* Correspondence: gzgylzt@163.com

Abstract: This study develops a progressive optimal fault-tolerant control method based on insufficient fault information. By combining passive and active fault-tolerant control manners during the process of fault diagnosis, insufficient fault information is fully used, and optimal fault-tolerant control effect is achieved. In addition, the fault-tolerant control method based on guaranteed robust cost control is introduced. The proposed progressive optimal fault-tolerant control method considers two aspects. First, as the amount of fault information continually increases, the performance index of the progressive optimal fault-tolerant controller improves. Second, at each moment, based on the corresponding insufficient fault information and prior knowledge, optimal fault-tolerant control is achieved according to current fault information. The process of progressive optimal fault-tolerant control converges to active fault-tolerant control when the fault is completely identified and the optimal fault-tolerant controller is no longer reconfigured until no more useful fault information can be provided. Furthermore, a progressive optimal fault-tolerant control algorithm based on the grid segmentation in the parameter uncertainty domain and the selection of different auxiliary center points is introduced. The simulation results verify the feasibility of the proposed algorithm and the validity of the proposed theory.

Keywords: passive fault-tolerant control; active fault-tolerant control; progressive optimal fault-tolerant control; guaranteed robust cost control

1. Introduction

In industrial systems, fault diagnosis (FD) and fault-tolerant control (FTC) are closely related. In recent years, there have been many studies on fault diagnosis and fault-tolerant control in various fields, including aerospace [1–3], power systems [4–6], high-speed rail [7–11], and satellites [12,13]. Based on previous research, fault diagnosis will inevitably develop in a faster and more accurate direction in the future. Obtaining fast and accurate fault information has been an essential requirement of active fault-tolerant control (AFTC) to ensure safe system operation; otherwise, the AFTC performance cannot be guaranteed, and even its implementation cannot be guaranteed. As well known, passive fault-tolerant control (PFTC) has been relatively conservative. However, unlike the AFTC, the PFTC does not require real-time fault information, which represents its inherent advantage. Traditional fault diagnosis has predominantly been mostly based on mathematical models [14–16]; however, model-based methods for fault diagnosis require an accurate system model, and the majority of these methods can only provide only two types of results (i.e., fault undiagnosed or fault diagnosed). Unfortunately, such approaches do not provide continuous fault information which is insufficient but crucial for effective fault-tolerant control. In recent years, there have been many studies on non-model-based methods (e.g., data-driven methods for fault diagnosis), especially artificial intelligence-based algorithms [17–19]. Fault diagnosis by artificial intelligence-based algorithms, such as neural networks, have a common problem that the behavior inside the "black box" is difficult to determine [20]. Thus, both model-based and non-model-based fault diagnosis methods have their own advantages and disadvantages. However, how to develop a more accurate and faster fault diagnosis method has still been a challenge.

When fault information cannot be rapidly and accurately obtained, a fault-tolerant control strategy which performs the PFTC before the fault is fully identified and then switch to the AFTC after the fault has been completely diagnosed seems to be a good choice. This idea has been proposed in [21]. However, its main disadvantage is the failure to utilize incomplete fault information, which is still valuable. Mechanically combining two fault-tolerant control methods is not ideal due to their respective shortcomings [22–24]. Therefore, it would be a perfect solution if the advantages of AFTC and PFTC can be combined under the condition of insufficient fault information. Existing research has indicated that the PFTC and AFTC are typically applied independently, with fewer studies exploring the combination of these two methods. Although the hybrid fault-tolerant control [25] combines the AFTC and the PFTC to a certain extent, these two control types are used separately depending on whether a fault has been fully diagnosed. In [26], a fault-tolerant control system (FTCS) design based on imprecise fault identification and robust reconfigurable control is proposed. This method reduces the time delay between the onset of a fault and the controller reconfiguration so that the system stability after the fault occurrence can be recovered rapidly. However, this method mainly emphasizes system stability rather than performance optimization. Moreover, the control object of this method does not have generality. In [27], actuator faults were considered as additive faults, and a combined passive-active FTC method based on reliable control is proposed, achieving a balance between performance and complexity. However, the predefined control laws were obtained offline rather than online by designing bottom-up extensible controllers with a minimal acceptable configuration, and the nominal performance remained at a sub-optimal level after a fault.

Although the above-mentioned methods combining the PFTC and AFTC methods are most mechanical, they have been valuable, but insufficient fault information has not been fully used. Moreover, the concept of progressive performance optimization has not been reflected, according to which, as fault information increases, the fault-tolerant control effect improves. In view of that, this study examines how to use insufficient fault information fully and combine the PFTC and AFTC manners efficiently to achieve optimal performance. This idea has been partly introduced in [28].

The comparison of the existing fault diagnosis methods has indicated that the parameter interval algorithm shows superiority over the other algorithms [29]. The parameter interval algorithm can continuously obtain increasingly accurate fault information during the fault diagnosis process. The smaller the parameter interval, the more accurate the fault identification. From the moment of fault occurrence to the moment of complete fault identification, the obtained fault information can be fully used to reconfigure the controller to ensure an optimal fault-tolerant control performance.

The pursuit of achieving a robust and optimal control effect while addressing the limitations of practical methods and ensuring system stability, even at the expense of some system performance, has long been a research focus [30–33]. Notably, Xue's study [34] on robust and optimal control, which considers both robustness and the control system's effectiveness, holds significant reference value. Since a system fault can be viewed as system uncertainty, Xue's research result on robust and optimal control can be applied to the field of fault-tolerant control for faulty systems.

The process of a progressive optimal fault-tolerant control method combines the PFTC and AFTC manners, as introduced in this article. When a fault occurs, the maximum uncertainty domain could be determined based on the prior knowledge. Moreover, the more fault information is obtained, the smaller the uncertainty domain of the faulty system. The progressive optimal fault-tolerant control method based on robust guaranteed cost fault-tolerant control, has been used to reconfigure the controller and ensure the optimal fault-tolerant control effect with the improving fault information. When a fault is completely identified the process of progressive optimal fault-tolerant control converges to active fault-tolerant control and the optimal fault-tolerant controller is no longer reconfigured until no more useful fault information can be provided. The essence of the progressive optimal fault-tolerant control method lies in combining active and passive fault-tolerant control manners by using the continuously improving fault information.

The rest of this article is structured as follows. In Section 2, the necessary preliminaries and the problem formulation of progressive optimal fault-tolerant control are provided. Section 3 explores a progressive optimal fault-tolerant control method in a linear uncertain system. A case study is presented in Section 4. Finally, Section 5 concludes this article.

2. Preliminaries and Problem Formulation

A system fault can be considered as a deviation of the system parameters [35]. Therefore, a faulty system can be modeled as an uncertain dynamic system with parameter uncertainty. The area where an actual value point of the system parameter vector might exist is called the uncertainty domain.

An uncertain dynamic system is defined by (1), and its state feedback is given by (2).

$$\begin{cases} \dot{x} = f(x,\theta,u) \\ y = Cx \\ \theta = \theta_0 + \Delta\theta \end{cases} \tag{1}$$

$$u = -Kx \tag{2}$$

In (1) and (2), $x \in R^n$ represents the system's state parameter, $u \in R^p$ represents the input, and $y \in R$ represents the output, respectively; $f(\bullet)$ is a non-linear function of x and u parameterized by a vector θ ; C represents the output matrix with a proper dimension; $\Delta\theta$ denotes the uncertainty of the parameter vector related to the uncertainty domain Ω, i.e., $\theta_0 + \Delta\theta \in \Omega$. In this study, it is assumed that the uncertainty domain Ω surrounds the nominal value θ_0 of the system parameter vector θ, and $K \in R^q$ is the controller parameter vector.

The selection of the controller parameter vector K is called controller configuration. It is assumed that this selection is related to the object function (3).

$$J = F(x,u) \tag{3}$$

Furthermore, assume that $p_1(x,u), p_2(x,u), \ldots p_r(x,u) \ldots p_g(x,u)$ are g parameters of a closed-loop system with certain constraint conditions. These g parameters can take eigenvalues of the closed-loop system (1) or other values, depending on the application context. The constraint condition of the controller parameter vector K can be defined by (4).

$$p_r(x,u) \in \Lambda_r, \forall r = 1,2,\ldots,v, \forall K \in \Psi(\Omega), \forall \theta \in \Omega \tag{4}$$

The constraint condition (4) implies a set of crucial indexes that should be satisfied and represents the basic constraint condition of the controller parameter selection. In (4), Λ_r represents a certain domain in a complex plane. For instance, if p_r is an eigenvalue, then Λ_r can be a left-half s plane. The closed-loop system (1) is considered to have good stability if condition (4) is satisfied.

Definition 1: All values of controller vector K under constraint condition (4) form a feasible domain $\Psi(\Omega)$ corresponding to an uncertainty domain Ω [28].

Then, the objective of controller configuration is that the closed-loop system (1) satisfies the following condition:

$$\min J = \min F(x,u) \tag{5}$$

where $\min J$ can be analytic or non-analytic expression; for instance, in the general case, it can be a minimizing operation of a quadratic function of system state variables. Alternatively, it could be described non-analytically, such as "the controller is the simplest to obtain".

For the state feedback controller (2), (6) is selected as one of the constraint conditions.

$$J \le J^* \tag{6}$$

In (6), J^* is a positive number and denotes the upper bound of the performance index J .

For the closed-loop system (1) and performance index (3), all controller parameter values K corresponding to the uncertainty domain Ω that satisfy condition (6) form as feasible domain $\Psi(\Omega)$. Therefore, the progressive optimal fault-tolerant control is discussed in the feasible domain $\Psi(\Omega)$ corresponding to the uncertainty domain Ω.

Each time with the narrowing of the uncertainty domain Ω of a fault, i.e., the fault information becomes increasingly sufficient, intuitively, there exists the following relation: $\Omega \supset \Omega_1 \supset \Omega_2 \supset ... \supset \Omega_i \supset ... \supset \Omega_j$, where Ω_i denotes the uncertainty domain at the ith moment. And the jth moment is after the ith moment, if $j > i$. This indicates that the uncertainty domain Ω of a fault and its narrowed sub-domains exhibit the nested property.

With the continuous increase and improvement of fault information, the uncertainty domain of the fault shrinks.

To illustrate the progressive optimal fault-tolerant control method, we first introduce Lemma 1.

Lemma 1: For an uncertain dynamic system (1), the smaller the range of uncertainty domain Ω of a fault, the lower the upper bound of the performance index.

Proof: Consider an arbitrary sub-domain Ω_i of an uncertainty domain Ω of a fault. For uncertain system (1), suppose that the upper bounds $J^*(\Omega)$ and $J^*(\Omega_i)$ of performance index corresponding to Ω and Ω_i, respectively, satisfy the following condition:

$$J^*(\Omega_i) > J^*(\Omega) \tag{7}$$

As long as the actual system parameter value satisfies $\theta \in \Omega$, it holds that $J \le J^*(\Omega)$. Under the condition of $\Omega_i \subset \Omega$, the actual system parameter value θ locates in Ω_i, but it also locates in Ω simultaneously due to the nested property. Therefore, the performance index J corresponding to Ω_i satisfies the condition of $J \le J^*(\Omega)$ according to (6). Thus, the upper bound of the performance index $J^*(\Omega_i)$ corresponding to Ω_i satisfies the condition of $J^*(\Omega_i) \le J^*(\Omega)$, and (7) is not true.

Hence the proof.

Based on Lemma 1, $J^*(\Omega_i) \le J^*(\Omega)$ is valid, and in accordance with the nested property, when the range of the uncertainty domain is narrowing (i.e., $\Omega_j \subset \Omega_i$), then it holds that

$$J^*(\Omega_j) \le J^*(\Omega_i),\ j \le i \tag{8}$$

It should be noted that $J^*(\Omega_j)=J^*(\Omega_i)$ indicates that regardless of the sub-domain where an actual system parameter value can be located, the upper bound of the performance index will not change. That further means the fault has been identified or the fault diagnosis procedure cannot provide more useful fault information.

Definition 2: With each narrowing of the uncertainty domain $\Omega_i \subset \Omega_j \subset \Omega$ of a fault depending on the progressively sufficient fault information, the controller with the minimum upper bound of the performance index can be defined as follows:

$$u = K_i x \tag{9}$$

$$K_i = \arg\min(J^*(\Omega_i)), \forall \theta \in \Omega_i \tag{10}$$

$$\min(J^*(\Omega_i)) \le \min(J^*(\Omega_j)), \Omega \supset \Omega_1 \supset ...\Omega_j \supset ... \supset \Omega_i, \forall j < i, \forall j = 1,2,3... \tag{11}$$

Controller K_i that satisfies (10) and (11) corresponding to uncertainty sub-domain Ω_i of a fault represents a progressive optimal fault-tolerant controller, and the whole control process is progressive optimal fault-tolerant control.

Theorem 1. When dynamic system (1) satisfies the following three conditions in a different and continuously narrowing uncertainty domain Ω_i of a fault,

1) $J(\Omega_i) \le J^*(\Omega_i), \forall K_i \in \Psi(\Omega_i), \Omega \supset \Omega_1 \supset \Omega_2 \supset ... \supset \Omega_j \supset ... \supset \Omega_i, \forall j < i, \forall j = 1,2,3...$;
2) $K_i = \arg\min(J^*(\Omega_i))$;
3) $\min(J^*(\Omega_i)) \le \min(J^*(\Omega_j))$;

then system (1) is a progressive optimal fault-tolerant control system, where $\Psi(\Omega_i)$ is the feasible domain formed by controller parameter vectors K_i that satisfy constraint condition 1) for the uncertainty domain Ω_i.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 March 2024 doi:10.20944/preprints202403.0920.v1

Proof of Theorem 1. According to Definition 2, with the narrowing of uncertainty sub-domain Ω_i of a fault, a progressive optimal fault-tolerant controller K_i is currently optimal with $\min(J^*(\Omega_i))$.

When the uncertainty sub-domain Ω_i of a fault decreases with the gradually improving fault information, in accordance with Lemma 1 and the nested property of the uncertainty domain, the upper bound of the performance index decreases, i.e.,

$$J^*(\Omega_i) \le J^*(\Omega) \tag{12}$$

Then, it holds that

$$\min(J^*(\Omega)) \ge \min(J^*(\Omega_j)) \ge \min(J^*(\Omega_i)), \Omega \supset \Omega_1 \supset \Omega_2 \supset ... \supset \Omega_j \supset ... \supset \Omega_i, \forall j = 1,2,3... \tag{13}$$

Hence the proof.

From (13), it's obvious that the narrower the uncertainty domain of a fault, the better the control effect achieved during the process of progressive optimal fault-tolerant control. In the current uncertainty domain, a fault-tolerant controller is optimal with a minimum upper bound $\min(J^*(\Omega_i))$ of the performance index. The progressive optimal fault-tolerant control is performed until the fault is fully identified or the diagnosis process cannot provide more useful fault information.

3. Progressive Optimal Fault-Tolerant Control in A Linear Uncertain System

Consider a linear system defined as follows:

$$\dot{x} = Ax + Bu, y = Cx \tag{14}$$

Assume that there is a parameter fault in a linear uncertain system (14), which can be expressed by

$$\dot{x} = (A+\Delta A)x + (B+\Delta B)u, y = Cx \tag{15}$$

where A and B represent the nominal system matrix and control matrix, respectively, and $A \in R^{n\times n}, B \in R^{n\times m}$; C is the output matrix. The possible deviation domains of the faulty parameters are considered to be uncertainty domains; ΔA and ΔB denote the parameter uncertainties caused by a fault of the controlled object and actuator, respectively, and these two types of fault are reflected in changes in the matrices A and B. ΔA and ΔB denote uncertain real-value matrices with appropriate dimension. According to [34], it can be written that

$$\Delta A = M_1 F_1(t) E_1 \tag{16}$$

$$\Delta B = M_2 F_2(t) E_2 \tag{17}$$

$$\|\Delta A\| \le \alpha, \|\Delta B\| \le \beta \tag{18}$$

where $M_1, M_2 \in R^{n\times r}, E_1 \in R^{q\times n}, E_2 \in R^{q\times m}$, and they are all rational real matrices; α and β are known scalars that means ΔA and ΔB are norm bounded; $F_1(t), F_2(t) \in R^{r\times q}$ are uncertainty function matrices that represent the time degeneration of a parameter fault.

Assume that matrices $F_1(t), F_2(t)$ belong to a set Θ as defined below [34]:

$$\Theta = \left\{ F_z(t) \middle| F_z^T(t) F_z(t) \le I, \forall z = 1,2, \forall t \right\} \tag{19}$$

Consider a progressive optimal fault-tolerant control method for a linear uncertain system, as discussed below. With the constraint condition of guaranteed robust cost control, the progressive optimal fault-tolerant control methods is achieved by searching for a feasible domain on the uncertainty domain of the fault.

3.1. Progressive Optimal Fault-Tolerant Control from the Perspective of Guaranteed Robust Cost Control

According to Theorem 8.3.2 in [34], which defines that for system (15) and performance index (20), the sufficient and necessary condition for a linear state feedback controller (21) to make a closed-loop system (15) guaranteed robust cost is that there exists a symmetric matrix X, matrix Y, and a suitable constant $\varepsilon > 0$ that make the linear matrix inequality (22) hold.

$$J = E\left\{ \int_0^\infty (x^T Q x + u^T R u) dt \right\} \tag{20}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 March 2024 doi:10.20944/preprints202403.0920.v1

$$u = -Kx = -YX^{-1}x \tag{21}$$

$$\begin{bmatrix} \psi & D & XE_1^T - Y^T E_2^T & X & Y^T \\ * & -\varepsilon^{-1} I & 0 & 0 & 0 \\ * & * & -\varepsilon I & 0 & 0 \\ * & * & * & -Q^{-1} & 0 \\ * & * & * & * & -R^{-1} \end{bmatrix} < 0 \tag{22}$$

where:

$X = H^{-1}$;

$Y = KH^{-1}$;

$\psi = A_0 X + X A_0^T - B_0 Y - Y^T B_0^T$;

I is a unit matrix;

$*$ is a transpose matrix with the corresponding term.

Furthermore, the corresponding upper bound of performance index (20) is defined by

$$J \le J^* = \mathrm{tr}(H) \tag{23}$$

From above, there is an implicit precondition that the uncertainty domain surrounds the normal system parameter value, that is, the nominal parameter value is used to design a guaranteed robust cost controller, as shown in Figure 1.

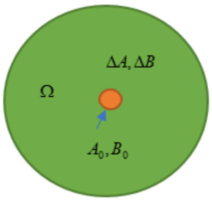

Figure 1. The uncertainty domain diagram.

As more fault information becomes available, the uncertainty domain of a fault where the value point of the system parameter vector can be located will become narrower. According to Theorem 1 of progressive optimal fault-tolerant control, the minimum upper bound of performance index (23) continuously decreases until a fault is fully identified or no more fault information can be provided. Furthermore, due to the currently insufficient fault information, the fault-tolerant controller should be currently optimal.

Obviously, after a fault occurs, a system must deviate from the nominal state and if the nominal parameter value is used to design a progressive optimal fault-tolerant controller, the fault-tolerant control can be conservative or even invalid. Therefore, in this study, domain segmentation is performed for the uncertainty domain of a fault to obtain the auxiliary center point to design a progressive optimal fault-tolerant controller.

At each time, uncertainty domain Ω_i of a fault can be determined according to the current and insufficient fault information. Then, domain segmentation is performed on uncertainty domain Ω_i of the fault. Each sub-domain $\Omega_{iw}, w = 1,2,3,\ldots s, i = 1,2,3\ldots$ of uncertainty domain Ω_i satisfies the condition of $\Omega_i = \Omega_{i1} \cup \Omega_{i2} \cup \ldots \cup \Omega_{iw} \cup \ldots \cup \Omega_{is}, w = 1,2,3\ldots s, i = 1,2,3\ldots$, where s represents the number of sub-domains. Then, the center point for each sub-domain Ω_{iw} is selected as an auxiliary center point.

For the auxiliary center points, the farthest distance from each auxiliary center point to the boundary of the current uncertainty domain of a fault is used as the maximum uncertainty magnitude of that point. At each segmentation part for Ω_i, there is a different uncertainty domain $\Upsilon_{iw}, w = 1,2,3\ldots s, i = 1,2,3\ldots$ for each auxiliary center point for each sub-domain Ω_{iw}. Meanwhile, s is also the number of auxiliary center points. For instance, for a rectangle uncertainty domain Ω_1 in

Figure 2, the grid segmentation is performed on the uncertainty domain, dividing it into four uncertainty sub-domains $\Omega_{1w}, w=1,2,3,4$. Next, the center point $T_w, w=1,2,3,4$ is selected for each uncertainty sub-domain $\Omega_{1w}, w=1,2,3,4$ as an auxiliary center point. Then, the length $l_w, w=4,3,2,1$ from the auxiliary center point $T_w, w=1,2,3,4$ to the farthest boundary point $O_w, w=4,3,2,1$ in the whole rectangle uncertainty domain is denoted as the maximum amplitude of uncertainty domain $\Upsilon_{1w}, w=1,2,3,4$. Furthermore, controller (21) is designed with the corresponding auxiliary center for each uncertainty domain $\Upsilon_{1w}, w=1,2,3,4$. Finally, the controllers for all uncertainty domains $\Upsilon_{1w}, w=1,2,3,4$ form the feasible domain $\Psi(\Omega_1)$.

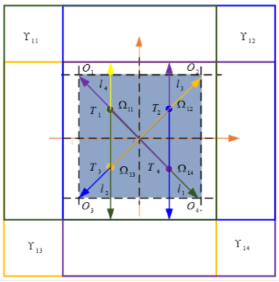

Figure 2. The diagram of the uncertainty domain determination.

Theorem 2. For the guaranteed robust cost controller vectors K designed for each uncertainty domain $\Upsilon_{iw}, w=1,2,3...s, i=1,2,3...$, controller with the minimum upper bound of performance index represents a progressive optimal fault-tolerant controller.

$$K = \arg\min(H) = \arg\min(\mathrm{tr}(H)) = YX^{-1} \tag{24}$$

Proof of Theorem 2. It is obvious that controller K in (21) designed for each uncertainty domain $\Upsilon_{iw}, w=1,2,3...s, i=1,2,3...$ is also feasible for $\Omega_{iw}, w=1,2,3...s, i=1,2,3...$ due to $\Omega_i = \Upsilon_{i1} \cap \Upsilon_{i2} \cap ... \cap \Upsilon_{iw} \cap ... \cap \Upsilon_{is}, w=1,2,3,...s, i=1,2,3...$. Namely, controllers K in (21) corresponding to the auxiliary center points of $\Upsilon_{iw}, w=1,2,3...s, i=1,2,3...$ form a feasible domain $\Psi(\Omega_i)$ on Ω_i. Thus, controller (24) with the minimum upper bound of performance index (23) in the feasible domain $\Psi(\Omega_i)$ denotes the current progressive optimal fault-tolerant controller.

Hence the proof.

As the uncertainty domain of a fault decreases with the progressively sufficient fault information, the aim is to find a progressive optimal fault-tolerant controller corresponding to (24) in the feasible domain $\Psi(\Omega_i)$ to achieve progressive optimal fault-tolerant control. The progressive optimal fault-tolerant control process based on a guaranteed robust cost control considers both stability and performance simultaneously.

3.2. Progressive Optimal Fault-Tolerant Algorithm

From above, it is necessary to segment the uncertainty domain of a fault and set the auxiliary center point to design a progressive optimal fault-tolerant control algorithm according to the aforementioned control method. Furthermore, to determine the center point for each segmented domain as an auxiliary center point more easily, grid segmentation is selected as a division method for the uncertainty domains of a fault. The number of grids to be divided is determined according to the specific uncertainty domain. Then, the center points of each grid are regarded as auxiliary center points to design a controller. For the rectangle uncertainty domain Ω of a fault, as shown in Figure 3, the uncertainty domain Ω is divided into four grids, denoted by $\Omega_1, \Omega_2, \Omega_3$ and Ω_4. The grid center points $D_1 - D_4$ of each grid are used as auxiliary center points.

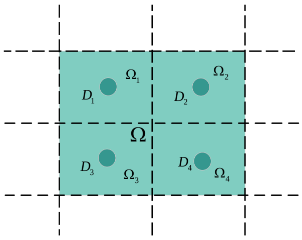

Figure 3. An example of grid segmentation.

The pseudo-code of the progressive optimal fault-tolerant control algorithm is presented below.

Progressive optimal fault-tolerant control algorithm from the perspective of guaranteed robust cost control

---

Input: A, B

for i from the first time to the nth time with an increment of 1

for w form 1 to s with an increment of 1

Assure uncertainty domain Ω_i of a fault and perform the grid segmentation on Ω_i , dividing it into s uncertainty sub-domains denoted by $\Omega_{iw}, i = 1, 2, 3, \ldots, n, w = 1, 2, 3, \ldots, s, i = 1, 2, 3\ldots$. Use the grid center points of the grids as auxiliary center points and the farthest distance from the center points of each grid to the boundary of Ω_i is used as a maximum uncertainty magnitude of that point. Determine ΔA_{iw} and ΔB_{iw} of each uncertainty domain $\Upsilon_{iw}, w = 1, 2, 3\ldots s, i = 1, 2, 3\ldots$ and realize the singular value decomposition of $\Delta A_{iw} = M_{iw} F_{iw}(t) E_{iw}$, $\Delta B_{iw} = M_{iw} F_{iw}(t) E_{iw}$;

Solve linear matrix inequality (22) for each grid.

All guaranteed robust cost controllers (21) for each grid form a feasible domain $\Psi(\Omega_i)$ on Ω_i. Find the controller K satisfying (24) in $\Psi(\Omega_i)$ as the current progressive optimal fault-tolerant controller.

end for

if no more useful fault information is provided

end for

end if

return controller K

end for

---

4. Simulations

In the simulation part, progressive optimal fault-tolerant control of a DC motor with the state space model is considered [36],

$$\begin{aligned} \dot{x} &= (A + \Delta A)x + Bu \\ y &= Cx \end{aligned} \tag{25}$$

with $x = [i_a \quad \omega]^T$ and ΔA being the uncertainty caused by the parameter fault.

$$A = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_v}{L_a} \\ \frac{K_m}{J_i} & -\frac{G}{J_i} \end{bmatrix}, B = \begin{bmatrix} \frac{1}{L_a} \\ 0 \end{bmatrix},$$

$$D = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \Delta A = \begin{bmatrix} 0 & -\frac{\sigma_v}{L_a} \\ \frac{\sigma_m}{J_i} & 0 \end{bmatrix} \tag{26}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 March 2024 doi:10.20944/preprints202403.0920.v1

$$u = Kx = [k_1, k_2]x \tag{27}$$

where i_a, ω, and v_a denote the armature current, angular velocity, and armature voltage, respectively. R_a is the armature resistance and L_a is the inductance of the DC motor. K_v and K_m are the voltage and motor constants, which are supposed to have parameter variations of $|\sigma_{kv}| \le 2$ and $|\sigma_{km}| \le 2$ due to the fault, respectively. J_i is the moment of inertia, and G is the friction coefficient. $K \in R^{1\times 2}$ is the controller parameter vector.

The purpose of the simulation is to regulate the output error of y, which represented the error of armature current and angular velocity, to be near zero under a fault. The normal parameter values used in the simulations are presented in Table 1.

Table 1. Parameter values [37].

Parameter	Value
R_a	1.2 Ω
L_a	0.05 mH
K_v	0.6
K_m	0.6
J_i	0.1352
G	0.3

The faulty system (28) is considered.

$$A_f = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_v - 0.4}{L_a} \\ \frac{K_m + 0.6}{J_i} & -\frac{G}{J_i} \end{bmatrix}, B = \begin{bmatrix} \frac{1}{L_a} \\ 0 \end{bmatrix}, D = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{28}$$

Then, Algorithm is performed below:

Case 1: After a parametric fault occurred, the maximum uncertainty domain could be assured according to prior knowledge and $\Delta A_1 = \begin{bmatrix} 0 & -\frac{|2|}{L_a} \\ \frac{|2|}{J_i} & 0 \end{bmatrix}$. $\sigma_{kv} = 0, \sigma_{kv} = 2, \sigma_{kv} = -2, \sigma_{km} = 0, \sigma_{km} = 2, \sigma_{km} = -2$ are used to perform grid segmentation on the uncertainty domain, as shown in Figure 4 The center points of the grids, $H_1 - H_4$, are used as auxiliary center points. The optimization performance index $\min(J^*)$ corresponding to the feasible domain and the progressive optimal fault-tolerant controller are shown in Table 2.

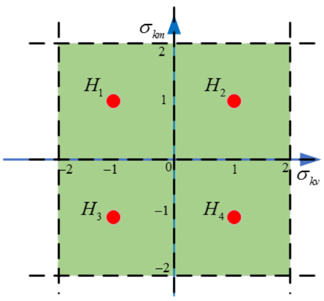

Figure 4. The grid segmentation in Case 1.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 March 2024 doi:10.20944/preprints202403.0920.v1

Table 2. The progressive optimal fault-tolerant controller parameters in case 1.

Auxiliary center point	σ_{kv} **,** σ_{km}	$\Delta A_{1r}\,(r=1,2,3,4)$	$K_n, n=1,2,3,4$	J_1^*	$\min(J_1^*)$	**Progressive optimal controller**
(1,1)	$-3\le\sigma_{kv}\le 1$ $-3\le\sigma_{km}\le 1$	$\begin{bmatrix} 0 & -\frac{-3}{L_a} \\ \frac{-3}{J_i} & 0 \end{bmatrix}$	$K_1=$ [-0.0912, 0.0020]	15.2337	15.2337	$K_1=$ [-0.0912, 0.0020]
(-1,1)	$-1\le\sigma_{kv}\le 3$ $-3\le\sigma_{km}\le 1$	$\begin{bmatrix} 0 & -\frac{3}{L_a} \\ \frac{-3}{J_i} & 0 \end{bmatrix}$	$K_2=$ [0.2630, 0.5141]	36.4675		
(-1,-1)	$-1\le\sigma_{kv}\le 3$ $-1\le\sigma_{km}\le 3$	$\begin{bmatrix} 0 & -\frac{3}{L_a} \\ \frac{3}{J_i} & 0 \end{bmatrix}$	$K_3=$ [0.0084, -0.0503]	32.9878		
(1,-1)	$-3\le\sigma_{kv}\le 1$ $-1\le\sigma_{km}\le 3$	$\begin{bmatrix} 0 & -\frac{-3}{L_a} \\ \frac{3}{J_i} & 0 \end{bmatrix}$	$K_4=$ [0.2203,-1.7682]	34.5717		

The fault-tolerant control result of Case 1 obtained using fault-tolerant controllers K_1, K_2, K_3, and K_4 is shown in Figure 5. Based on Table 2, the progressive optimal fault-tolerant controller is K_1, with the minimum upper bound of optimization performance index of $\min(J_1^*)$=15.2337 comparing to K_2 with $\min(J_1^*)$=36.4675, K_3 with $\min(J_1^*)$=32.9878 and K_4 with $\min(J_1^*)$=34.5717. As shown in Figure 5, controller K_1 performs better with a smaller overshot and better comprehensive performance, than controllers K_2, K_3, and K_4.

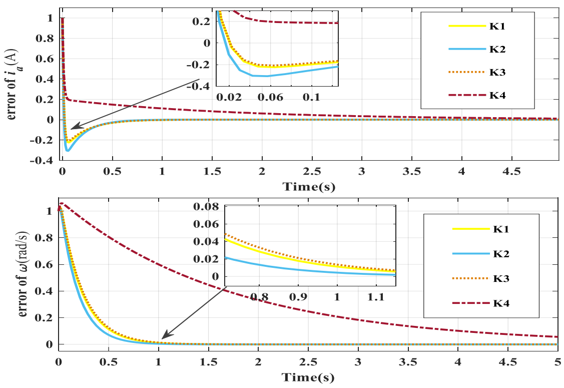

Figure 5. The control result of Case 1.

Case 2: In this case, it is assumed that the uncertainty domain narrowed with the increase of the fault information amount and $\Delta A_2=\begin{bmatrix} 0 & -\frac{|1|}{L_a} \\ \frac{|1|}{J_i} & 0 \end{bmatrix}$. Furthermore, $\sigma_{kv}=0, \sigma_{kv}=1, \sigma_{kv}=-1, \sigma_{km}=0, \sigma_{km}=1, \sigma_{km}=-1$ are used to perform grid segmentation on the uncertainty domain, as shown in Figure 6. The center points of the grids, D_1-D_4, are used as auxiliary center points. The optimization performance index $\min(J^*)$ corresponding to the feasible domain and the progressive optimal fault-tolerant controller are shown in Table 3.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 March 2024 doi:10.20944/preprints202403.0920.v1

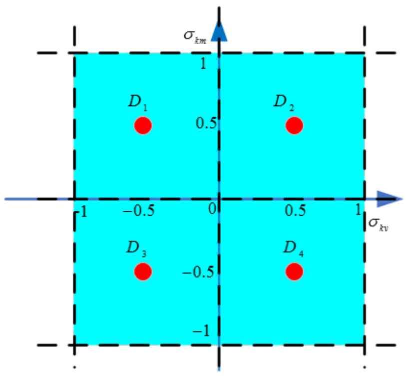

Figure 6. The grid segmentation in Case 2.

Table 3. The progressive optimal fault-tolerant controller parameters in case 2.

Auxiliary center point	σ_{kv}, σ_{km}	$\Delta A_{2r}(r=1,2,3,4)$	$K_n, n=5,6,7,8$	J_2^*	$\min(J_2^*)$	**Progressive optimal controller**
(0.5,0.5)	$-1.5 \le \sigma_{kv} \le 0.5$ $-1.5 \le \sigma_{km} \le 0.5$	$\begin{bmatrix} 0 & -\frac{-1.5}{L_a} \\ \frac{-1.5}{J_i} & 0 \end{bmatrix}$	$K_5=$ [-0.2389, -0.1059]	9.7222	9.7222	$K_5=$ [-0.2389, -0.1059]
(-0.5,0.5)	$-0.5 \le \sigma_{kv} \le 1.5$ $-1.5 \le \sigma_{km} \le 0.5$	$\begin{bmatrix} 0 & -\frac{1.5}{L_a} \\ \frac{-1.5}{J_i} & 0 \end{bmatrix}$	$K_6=$ [0.1685, 0.1665]	18.8913		
(-0.5,-0.5)	$-0.5 \le \sigma_{kv} \le 1.5$ $-0.5 \le \sigma_{km} \le 1.5$	$\begin{bmatrix} 0 & -\frac{1.5}{L_a} \\ \frac{1.5}{J_i} & 0 \end{bmatrix}$	$K_7=$ [0.0547, 0.0400]	22.6618		
(0.5,-0.5)	$-1.5 \le \sigma_{kv} \le 0.5$ $-0.5 \le \sigma_{km} \le 1.5$	$\begin{bmatrix} 0 & -\frac{-1.5}{L_a} \\ \frac{1.5}{J_i} & 0 \end{bmatrix}$	$K_8=$ [0.0187, 0.0543]	44.2438		

The fault-tolerant control result of Case 2 obtained using fault-tolerant controllers K_5, K_6, K_7, and K_8 is shown in Figure 7. Based on Table 3, the progressive optimal fault-tolerant controller is K_5, with the minimum upper bound of optimization performance index of $\min(J_2^*)$=9.7222 comparing to K_6 with $\min(J_2^*)$=18.8913, K_7 with $\min(J_2^*)$=22.6618 and K_8 with $\min(J_2^*)$=44.2438. As shown in Figure 7, controller K_1 performs better with a smaller overshot and better comprehensive performance, than controllers K_6, K_7, and K_8.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 March 2024 doi:10.20944/preprints202403.0920.v1

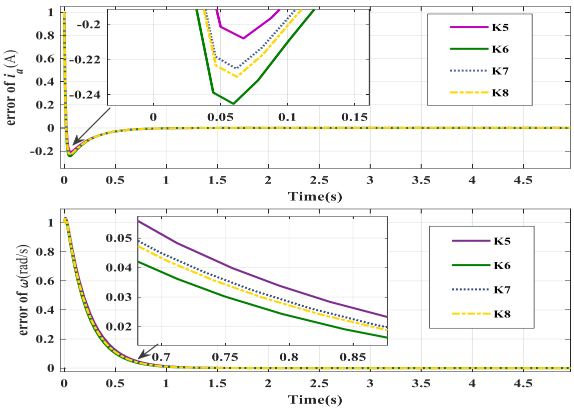

Figure 7. The control result of Case 2.

Case 3: In this case, it is assumed that the uncertainty domain decreases with the increase in the fault information amount, and a fault parameter has been identified, that is, $\sigma_{3v}=0$. Then, it is obtained that $\Delta A_3=\begin{bmatrix} 0 & 0 \\ \frac{|0.8|}{J_i} & 0 \end{bmatrix}$. Furthermore, $\sigma_{kv}=0.4, \sigma_{km}=0, \sigma_{km}=0.8, \sigma_{km}=-0.8$ are used to perform grid segmentation on the uncertainty domain, as shown in Figure 8. The center points of the grids, G_1 and G_2, are used as auxiliary center points. The optimization performance index $\min(J^*)$ corresponding to the feasible domain and the progressive optimal fault-tolerant controller are shown in Table 4.

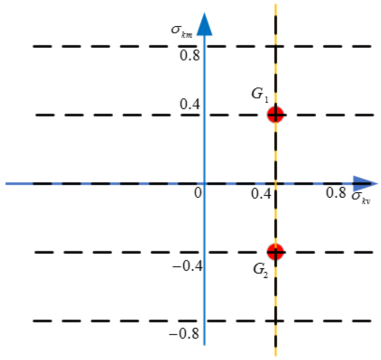

Figure 8. The grid segmentation in Case 3.

Table 4. The progressive optimal fault-tolerant controller parameters in case 3.

Auxiliary center point	σ_{kv}, σ_{km}	$\Delta A_{3r}\,(r=1,2)$	$K_n, n=9,10$	J_3^*	$\min(J_3^*)$	**Progressively optimal controller**
(0.4,0.4)	$\sigma_{kv}=0$ $-1.2 \le \sigma_{km} \le 0.4$	$\begin{bmatrix} 0 & 0 \\ \frac{-1.2}{J_i} & 0 \end{bmatrix}$	K_9 = [-0.4347,-0.2762]	6.8313	6.8313	K_9 = [-0.4347,-0.2762]
(0.4,-0.4)	$\sigma_{kv}=0$ $-0.4 \le \sigma_{km} \le 1.2$	$\begin{bmatrix} 0 & 0 \\ \frac{1.2}{J_i} & 0 \end{bmatrix}$	K_{10} = [-0.2968, -0.2409]	10.2335		

The fault-tolerant control result of Case 3 obtained using fault-tolerant controllers K_9 and K_{10} is shown in Figure 9. Based on Table 4, the progressive optimal fault-tolerant controller is K_9, with the minimum upper bound of the performance index of $\min(J_3^*)=6.8313$ compare to K_{10} with

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 March 2024 doi:10.20944/preprints202403.0920.v1

$\min(J_3^*)=10.2335$. As shown in Figure 9, controller K_9 performs better than controller K_{10}, having smaller overshot and better comprehensive performance.

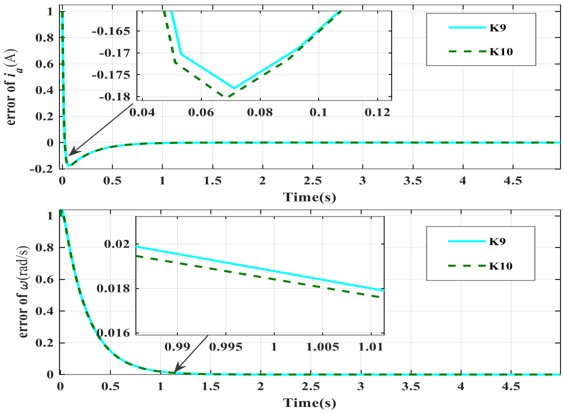

Figure 9. The control result of Case 3.

According to the data presents in Tables 4-6, the minimum upper bound $\min(J^*)$ of the performance index decreases as the uncertainty domain becomes narrower, it is $\min(J_1^*)=15.2337$ for $\Delta A_1=\begin{bmatrix} 0 & -\frac{|2|}{L_a} \\ \frac{|2|}{J_i} & 0 \end{bmatrix}$, $\min(J_1^*)=9.7222$ for $\Delta A_2=\begin{bmatrix} 0 & -\frac{|1|}{L_a} \\ \frac{|1|}{J_i} & 0 \end{bmatrix}$, and $\min(J_1^*)=6.83134$ for $\Delta A_3=\begin{bmatrix} 0 & 0 \\ \frac{|0.8|}{J_i} & 0 \end{bmatrix}$, which met the theory of progressive optimal fault-tolerant control. After each grid segmentation of the uncertainty domain, the progressive optimal fault-tolerant controller could be obtained. In addition, the corresponding progressive optimal fault-tolerant controller is currently optimal before the uncertainty domain stops narrowing, that is, before the useful fault information amount stops increasing or the fault is identified. The above simulation results also verify the feasibility of the algorithm.

5. Conclusion

This paper studies the progressive optimal fault-tolerant control method combining the AFTC and PFTC manners by fully using insufficient fault information. In this study, a system fault is considered as system uncertainty. The progressive optimal fault-tolerant control method based on guaranteed robust cost control is proposed. The proposed method addresses two aspects. First, as the domain of parameter uncertainty of a fault becomes narrower, the fault-tolerant effect improves. Second, at each time, based on the uncertainty domain of the corresponding fault information, an currently optimal fault-tolerant controller is determined. In the process of progressive optimal fault-tolerant control, the optimal fault-tolerant controller is no longer reconfigured until no more useful fault information can be provided. Finally, the process of progressive optimal fault-tolerant control converges to active fault-tolerant control while the fault is completely identified. A progressive optimal fault-tolerant control algorithm is introduced, and it's based on grid segmentation of the uncertainty domains of a fault and the selection of auxiliary center points. The proposed method is validated by the theoretical analysis and simulation. The proposed method has potential application value in practical control systems. In future work, attention will be focused on exploring progressive optimal fault-tolerant control with weaker conservatism and lower computational complexity.

Author Contributions: Conceptualization, Z.L. and B.D.; methodology, Z.L. and D.D.; software, D.D.; validation, D.D.; formal analysis, Z.L. and D.D.; investigation, D.D.; resources, Z.L.; data curation, D.D.; writing—original draft preparation, D.D.; writing—review and editing, Z.L. and D.D.; visualization, D.D.; supervision, Z.L.; project administration, Z.L.; funding acquisition, Z.L. All authors have read and agreed to the published version of the manuscript.

 doi:10.20944/preprints202403.0920.v1

Funding: This research was funded by National Natural Science Foundation of China, grant number 61963009.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Xiao, B.; Karimi, H.R.; Yu, X.; Gao, Q. IEEE access special section: Recent advances in fault diagnosis and fault-tolerant control of aerospace engineering systems. *IEEE Access* 2020, 8, 61157-61160.
2. Fekih, A. Fault diagnosis and fault tolerant control design for aerospace systems: A bibliographical review. In Proceedings of the American Contr. Conf., Portland, OR, USA, June 4-6, 2014.
3. Castaldi, P.; Mimmo, N.; Simani, S. Fault diagnosis and fault tolerant control strategies for aerospace systems. In Proceedings of the 3rd Conf Control Fault-Tolerant Syst., Barcelona, Spain, September 7-9, 2016.
4. Liu, H.; Loh, P.C.; Blaabjerg, F. Review of fault diagnosis and fault-tolerant control for modular multilevel converter of HVDC. In Proceedings of the 39th Annual Conf IEEE Ind. Elec. Society, Vienna, Austria, November 10-13, 2013.
5. Pazouki, E.; Sozer, Y.; De Abreu-Garcia, J.A. Fault diagnosis and fault-tolerant control operation of non-isolated DC–DC converters. *IEEE Trans Ind Appl.* 2017, 54, 310-320.
6. Khan, S.S.; Wen, H. A comprehensive review of fault diagnosis and tolerant control in DC-DC converters for DC microgrids. *IEEE Access* 2021, 9, 80100-80127.
7. Mao, Z.; Yan, X.G.; Jiang, B.; Chen, M. Adaptive fault-tolerant sliding-mode control for high-speed trains with actuator faults and uncertainties. *IEEE Trans. Intell. Transp. Syst.* 2019, 21, 2449-2460.
8. Zhai, M.; Long, Z.; Li, X. Fault-tolerant control of magnetic levitation system based on state observer in high speed maglev train. *IEEE Access* 2019, 7, 31624-31633.
9. Dong, H.; Lin, X.; Gao, S.; Cai, B. Ning B. Neural networks-based sliding mode fault-tolerant control for high-speed trains with bounded parameters and actuator faults. *IEEE Trans Veh Technol.* 2019, 69, 1353-1362.
10. Liu, S.; Jiang, B.; Mao, Z.; Ding, S.X. Adaptive backstepping based fault-tolerant control for high-speed trains with actuator faults. *Int J Contr Autom Syst.* 2019, 17, 1408-1420.
11. Yao, X.; Li, S.; Li, X. Composite adaptive anti-disturbance fault tolerant control of high-speed trains with multiple disturbances. *IEEE Trans Intell Transp Syst.* 2022, 23, 21799-21809.
12. Zhang, Z.; Ye, D.; Xiao, B.; Sun, Z. Third-order sliding mode fault-tolerant control for satellites based on iterative learning observer. *Asian J Contr.* 2019, 21, 43-51.
13. Liu, M.; Zhang, A.; Xiao, B. Velocity-Free State Feedback Fault-Tolerant Control for Satellite with Actuator and Sensor Faults. *Symmetry* 2022, 14, 157.
14. Patton, R.J. Robust model-based fault diagnosis: the state of the art. *Proc IFAC*. 1994, 27, 1-24.
15. Leonhardt, S.; Ayoubi, M. Methods of fault diagnosis. *Contr Engineering Practice.*1997, 5, 683-692.
16. Simani, S.; Fantuzzi, C.; Patton, R.J. *Model-Based Fault Diagnosis Techniques*.; Springer: Britain, 2003; PP. 19-60.
17. Zhang, T.; Chen, J.; Li, F.; Zhang, K.; Lv, H. et al. Intelligent fault diagnosis of machines with small & imbalanced data: A state-of-the-art review and possible extensions. *ISA Trans.* 2022, 119, 152-171.
18. Zhou, S.; Wang, K.; Shan, J.; Bao, D.; Hou, Z. et al. Data-Driven Multi-Type and Multi-Level Fault Diagnosis of Proton Exchange Membrane Fuel Cell Systems Using Artificial Intelligence Algorithms. *SAE Technical Paper*. 2022, 1, 693.
19. Chang, Y.; Chen, Q.; Chen, J.; He, S.; Li, F. et al. Intelligent fault diagnosis scheme via multi-module supervised-learning network with essential features capture-regulation strategy. *ISA Trans.* 2022, 129, 459-475.
20. Patton, R.J.; Chen, J. Benkhedda H. A study on neuro-fuzzy systems for fault diagnosis. *Int J Syst Sci.* 2000, 31, 1441-1448.
21. Jiang, J.; Yu, X. Fault-tolerant control systems: A comparative study between active and passive approaches. *Annu Rev Control.* 2012, 36, 60-72.
22. Abbaspour, A.; Mokhtari, S.; Sargolzaei, A.; Yen, K.K. A survey on active fault-tolerant control systems. *Electronics* 2020, 9, 1513.
23. Bavili, R.E.; Mohammadzadeh, A.; Tavoosi, J.; Mobayen, S.; Assawinchaichote, W. et al. A new active fault tolerant control system: Predictive online fault estimation. *IEEE Access* 2021, 9, 118461-118471.
24. Zhou, H.; Ye, H.; Wu, M. *Fault Detection and Fault-Tolerant Control Based on Sliding Mode Theory*; National Defence Industry Press: Beijing, 2014; PP. 2-25.
25. Yu, X.; Jiang, J. Hybrid fault-tolerant flight control system design against partial actuator failures. *IEEE Trans Contr Syst Technol.* 2011, 20, 871-886.
26. Jiang, J.; Zhao, Q. Fault tolerant control system synthesis using imprecise fault identification and reconfigurable control. In Proceedings of the IEEE Int. Symposium Intell. Control (ISIC) held jointly with IEEE Int. Symposium Comput. Intell. Robotics Automation (CIRA), Gaithersburg, MD, USA, September 14-17, 1998.

27. Tu, Y.; Wang, D.; Ding, S.X.; Fu, F.; Li, W. A Reconfiguration-Based Fault-Tolerant Control Method for Nonlinear Uncertain Systems. *IEEE Trans Autom Contr.* 2021, 61, 6060-6067.
28. Li, Z.; Dahhou, B. Fault-tolerant control Based on Insufficient Fault Information. In Proceedings of the 7th National Technical Process Fault Diagnosis Safety Academic Conf., 2011.
29. Li, Z.; Dahhou, B. An observers based fault isolation approach for nonlinear dynamic systems. In Proceedings of the 2nd Int Symposium Comm, control signal process, 2006.
30. Park, Y. Robust and optimal attitude control of spacecraft with disturbances. *Int J Syst Sci.* 2015, 46, 1222-1233.
31. Pan, H.; Xin, M. Nonlinear robust and optimal control of robot manipulators. *Nonlinear Dynamics* 2014, 76, 237-254.
32. Lao, Y.; Scruggs, J.T. Robust control of wave energy converters using unstructured uncertainty. In Proceedings of the American Control Conf(ACC), Denver, The United States, July 1-3,2020.
33. Cao, Z.; Xiao, Q.; Huang, R.; Zhou, M. Robust neuro-optimal control of underactuated snake robots with experience replay. *IEEE Trans Neural Netw Learn Syst.* 2017, 29, 208-217.
34. Xue, A. *Robust And Optimal Control Theory and Application*; Science press: Beijing, 2008; PP. 15-130.
35. Gao, Z.; Cecati, C.; Ding, S.X. A survey of fault diagnosis and fault-tolerant techniques-Part I: Fault diagnosis with model-based and signal-based approaches. *IEEE Trans Ind Electron.* 2015, 62, 3757–3767.
36. Shen, T. H_∞ *Control Theory and Applications*; Tsinghua University Press: Beijing, 1996; PP. 2-142.
37. Zhang, M.; Shi, D. Algebraic Riccati inequality and multi-objective optimization. *Inf Contr.* 2000, 29, 65-69.
38. Lan, J.; Patton, R.J. A new strategy for integration of fault estimation within fault-tolerant control. *Automatica* 2016, 69, 48-59.
39. Belanger, P.R. *Control Engineering: A Modern Approach*; Oxford University Press: Oxford,1995;PP. 12-200 .