2024 IEEE 63rd Conference on Decision and Control (CDC)
December 16-19, 2024. MiCo, Milan, Italy

Exponential Auto-Tuning Fault-Tolerant Control of N Degrees-of-Freedom Manipulators Subject to Torque Constraints

Mehdi Heydari Shahna[1] and Jouni Mattila[1]

Abstract— **Faulty joints in a robot manipulator adversely affect the tracking control performance and compromise the system's stability; therefore, it is necessary to design a control system capable of compensating for the effects of actuator faults to maintain control efficacy. To this end, this paper presents new amendments to the dynamical formulation of robot manipulators to account for latent actuator faults and over-generated torques mathematically. Subsequently, a novel auto-tuning subsystem-based fault-tolerant control (SBFC) mechanism is designed to force joints' states closely along desired trajectories, while tolerating actuator faults, excessive torques, and unknown modeling errors. Suboptimal SBFC gains are determined by employing the JAYA algorithm (JA), a high-performance swarm intelligence technique, standing out for its capacity to continuously approach optimal control levels without requiring meticulous tuning of algorithm-specific parameters, relying instead on its intrinsic principles. Notably, this control framework ensures uniform exponential stability (UES). The enhancement of accuracy and tracking time for reference trajectories, along with the validation of theoretical assertions, is demonstrated through simulation outcomes.**

I. Introduction

When applied to high-degree-of-freedom (DoF) manipulator systems, subsystem-based control decomposes a complex and high-order system into subsystems and can assist in the development of localized control strategies and the assessment of stability at the subsystem level [1], [2]. However, a new form of subsystem complexity is introduced based on interconnected state- and time-variant uncertainties, as well as joint faults, which are commonplace in real-world industrial settings [3]. Failures in autonomous and intelligent robotic systems can stem from various events, including internal actuator issues, power supply system failures, or wiring problems, impairing their performance, rendering them incapable of carrying out their tasks and necessitating the design of fault-tolerant control mechanisms to ensure their continued safe operation without causing harm [4]–[6]. As one potential remedy, many studies have focused on passive fault-tolerant control (PFTC) to maintain operational integrity and safety in applications that lack fault diagnosis and active intervention sections [7]. In their work [8], Van and Ge designed a passive fault-tolerant approach to mitigate the rapid effects of faults for robotic manipulators based on a robust backstepping control integrated with other methods. Likewise, in pursuit of achieving both fast response and high-precision tracking performance, Anjum and Guo in [9], proposed a PFTC system for robotic manipulators, built upon a fractional-order adaptive backstepping approach.

Furthermore, considering the limitations imposed by the magnitude of physical actuators, sensors, and interfacing devices, it becomes imperative to account for control input constraints [10]. Deviating from these constraints can result in the emergence of undesirable vibrations, degradations in system performance, and, in some cases, complete system immobilization [11]. Nohooji, as outlined in [12], enhanced the robustness of his neural adaptive proportional-integral-derivative (PID) control for manipulators by incorporating considerations of constrained behavior during system operation. Similarly, Yang et al. [13] developed an online integral reinforcement learning strategy to address the challenges of robust constrained control in nonlinear continuous-time systems.

Furthermore, to overcome a formidable challenge for subsystem-based control designers, managing the extensive array of control gains that demand meticulous tuning is imperative, as they exert a distinct impact on the system's transient and steady performance, even when deploying highly effective and top-performing control methodologies. As a promising solution to this challenge, population-based optimization algorithms have gained popularity in recent times due to their efficiency [14]. However, the improper tuning of algorithm-specific parameters can lead to increased computational effort or the attainment of suboptimal local solutions [15]. In contrast to most other optimization algorithms that necessitate the fine-tuning of algorithm-specific parameters for updating particle positions, the JAYA algorithm (JA) uniquely relies on its inherent principles to adapt and optimize a wide range of problems [16]. The JA was developed by Rao [17], with the primary objective of addressing both constrained and unconstrained optimization problems. It stems from an innovative swarm-based heuristic introduced in the work by Nanda et al. [18]. Further, in [19], Houssein and colleagues conducted an extensive review of renowned optimization algorithms. Their investigation revealed that in the task of function minimization, the JA consistently outperformed these well-established swarm-based algorithms, delivering markedly superior results in terms of both precision and convergence speed. Interestingly, in [20], Bansal and collaborators explored the capabilities of three distinct optimization algorithms for fundamental backstepping control of a single-link flexible joint manipulator system. Similar to [19], their investigative findings indicated that JA optimization consistently outperformed the

Funding for this research was provided by the Business Finland partnership project "Future All-Electric Rough Terrain Autonomous Mobile Manipulators" (Grant No. 2334/31/2022).

[1]The authors are with Faculty of Engineering and Natural Sciences, Tampere University, Finland (e-mail:{mehdi.heydarishahna, jouni.mattila}@tuni.fi).

other methods in terms of fitness value.

In consideration of the critical significance of robust control in ensuring both the safety and performance of robotic manipulators, this paper proposes a novel robust adaptive subsystem-based control to maintain the system's uniform exponential stability (UES) while tolerating various actuator faults, excessive torques, and unknown modeling errors. It not only incorporates the management of joint failures but also continuously improves the control parameters in suboptimal levels by customizing the highly promising swarm intelligence technique (JA). Therefore, the present study offers notable contributions to the field of robotics: (1) it introduces an innovative model for joint torques across different types of actuator functions: normal functioning (healthy mode), stuck failure, performance loss (encompassing incipient and abrupt faults), and saturation (excessive torque). Interestingly, this model addresses the over-generated torques even by faulty actuators; (2) it introduces a novel SBFC approach tailored for robotic manipulators with n DoF, capable of tolerating diverse actuator faults, excessive torques, and unknown modeling errors; (3) to tune the SBFC gains, a multi-population and single-phase swarm intelligence technique (JA) is amended, standing out for its capacity to continuously approaches optimal control levels without the need for meticulous tuning of algorithm-specific parameters, relying instead on its intrinsic principles; (4) and the proposed control strategy ensures the achievement of UES. The remaining sections of this paper are organized as follows: In Section II, the conventional model of an n DoF robotic manipulator dynamic is augmented with comprehensive actuator fault models and torque signal constraints. Section III outlines the step-by-step design of the SBFC strategy and presents the stability analysis. The effectiveness of the proposed strategy is thoroughly investigated in the final section.

II. Modeling the System and Defining the Problem

A. N DoF Manipulator

Considering the typical robotic manipulator dynamics, as detailed in [12], we have:

$$\boldsymbol{I}(\boldsymbol{q})\ddot{\boldsymbol{q}} = \boldsymbol{T} - \boldsymbol{C_m}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{f}(\dot{\boldsymbol{q}}) - \boldsymbol{G}(\boldsymbol{q}) - \boldsymbol{T_L}. \quad (1)$$

In the given context, $\boldsymbol{q} \in \mathbb{R}^n$ represents the generalized joint coordinate vector comprising 'n' joints. $\boldsymbol{I(q)} : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ characterizes the mass (inertia) properties, while $\boldsymbol{C_m(q,\dot{q})} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n\times n}$ accounts for the centrifugal and Coriolis forces. $\boldsymbol{G(q)} : \mathbb{R}^n \to \mathbb{R}^n$ represents the gravitational forces/torques, and $\boldsymbol{f(\dot{q})} : \mathbb{R}^n \to \mathbb{R}^n$ accounts for the resistance encountered during movement. The vector $\boldsymbol{T} = [T_1,\dots,T_n]^\top$ represents the generalized continuous torque applied at the joints, and $\boldsymbol{T_L} \in \mathbb{R}^n$ signifies unaccounted-for external disturbances that affect each joint. Notably, the inertia matrix $\boldsymbol{I}(\boldsymbol{q})$ possesses the properties of being symmetric, positive, and definite; thus, we can also say:

$$0 < I_{min}(\boldsymbol{I}(\boldsymbol{q})^{-1}) \leq \|\boldsymbol{I}(\boldsymbol{q})^{-1}\| \leq I_{max}(\boldsymbol{I}(\boldsymbol{q})^{-1}), \quad (2)$$

where $\|\cdot\|$ denotes the squared Euclidean norm, $I_{max}(.) \in \mathbb{R}^+$ and $I_{min}(.) \in \mathbb{R}^+$ represent the matrix $\boldsymbol{I}(\boldsymbol{q})^{-1}$'s maximum and minimum eigenvalues, respectively.

B. Passive Faulty Dynamic Model

Next, we integrate the fault correction functionality into the established control algorithm. To do so, we mathematically describe the actuator faults that may happen, as [6], [21]:

$$\boldsymbol{T} = \boldsymbol{T_c} + \boldsymbol{\varepsilon}(\boldsymbol{T_{sat}} - \boldsymbol{T_c}), \quad (3)$$

where $\boldsymbol{T_c} \in \mathbb{R}^n$ represents the normal command control during the system's healthy state. We use $\boldsymbol{\varepsilon} = \mathrm{diag}(\varepsilon_1,\dots,\varepsilon_n)$ and $\boldsymbol{T_{sat}} \in \mathbb{R}^n$ to characterize various types of actuator failures, with t_f signifying the period of fault occurrence. When $\varepsilon_i = 0$, the corresponding actuator is functioning normally. When $T_{sat(i)} \neq 0$ indicates a stuck failure. Meanwhile, $0 < \varepsilon_i < 1$ represents a performance loss. The behavior model of the fault, when $0 < \varepsilon_i < 1$, is extended, as follows:

$$\varepsilon_i = 1 - e^{-\gamma_i t} \quad t \in t_f, \qquad \gamma_i > 0, \quad (4)$$

where γ_i represents the rate of evolution of an undisclosed fault. A small γ_i value indicates slow fault development, termed an incipient fault. Conversely, a high γ_i value results in the time course γ_i approximating a step form, classified as an abrupt fault [22].

Remark (1): In this paper, we assume that $\varepsilon_i \neq 1$. This assumption is crucial because $\varepsilon_i = 1$ signifies an uncompensatable fault in the ith actuator, resulting in a complete loss of control access. In cases of such severe faults, control strategies become impractical, necessitating the exploration of mechanical alternatives. These alternatives are unrelated to the concept of control strategies. They may involve actions such as replacing or repairing the faulty actuator or introducing an additional actuator to compensate for the control failure, as discussed in [6].

C. Torque Signal Constraint

In addition to addressing actuator faults, our objective is to account for the torque constraints to ensure they do not exceed the specified nominal torque values. For $i = 1,\dots,n$ joints to operate in compliance with the constraints imposed on the control torque $T_i(t)$ for each joint, whether in a healthy or faulty state. This is achieved as follows:

$$S(T_i(t)) = \begin{cases} \bar{T}_i, & T(t) \geq \bar{T}_i \\ T(t) & \underline{T}_i \leq T_(t) \leq \bar{T}_i \\ \underline{T}_i & T(t) \leq \underline{T}_i \end{cases}. \quad (5)$$

In this context, $\bar{T}_i$ and $\underline{T}_i$ denote the upper and lower nominal torque bounds, respectively, of the permissible $T_i(t)$ values that can be generated. Consequently, we define:

$$\boldsymbol{S(T)} = [S_1(T_1(t)),\dots,S_n(T_n(t))]^\top. \quad (6)$$

To elaborate further, we define a constraint model as follows:

$$S_i(T_i(t)) = s_{1i}\; T_i(t) + s_{2i}, \quad (7)$$

where

$$s_{1i} = \begin{cases} \frac{1}{|T_i(t)|+1}, & T_i(t) \geq \overline{T}_i \text{ or } T_i(t) \leq \underline{T}_i \\ 1 & \underline{T}_i \leq T_i(t) \leq \bar{T}_i \end{cases} \tag{8}$$

and

$$s_{2i} = \begin{cases} \overline{T}_i - \frac{T_i(t)}{|T_i(t)|+1}, & T_i(t) \geq \overline{T}_i \\ 0 & \underline{T}_i \leq T_i(t) \leq \overline{T}_i \cdot \\ \underline{T}_i - \frac{T_i(t)}{|T_i(t)|+1} & T_i(t) \leq \underline{T}_i \end{cases} \tag{9}$$

It is evident that Eqs. (7), (8), and (9) imply Eq. (5). We have $s_{2i} \leq \max(|\underline{T}_i|+1, |\overline{T}_i|+1)$ and $s_{1i} \leq 1$. In addition, if we generally say $\boldsymbol{s}_1 = \text{diag}(s_{11}, \ldots, s_{1n})$, and $\boldsymbol{s}_2 = [s_{21}, \ldots, s_{2n}]^\top$, we can have from (3) and (6):

$$\boldsymbol{S}(\boldsymbol{T}) = \boldsymbol{s_1}\boldsymbol{T} + \boldsymbol{s_2}, \tag{10}$$

By incorporating both the faulty dynamic model defined in (3) and the torque constraints outlined in (10), we derive a novel model for joint torques that encompasses both actuator faults and torque constraints. This model is designed to prevent the generation of excessive torques by faulty actuators, as detailed below:

$$\begin{aligned} \boldsymbol{S}(\boldsymbol{T}) =& \boldsymbol{S}(\boldsymbol{T_c} + \boldsymbol{\varepsilon}(\boldsymbol{T_{sat}} - \boldsymbol{T_c})) = \boldsymbol{s_1}\boldsymbol{T_c} + \boldsymbol{s_1}\boldsymbol{\varepsilon}(\boldsymbol{T_{sat}} - \boldsymbol{T_c}) + \boldsymbol{s_2} \\ =& \boldsymbol{s}_1(\boldsymbol{I}_{n\times n} - \boldsymbol{\varepsilon})\boldsymbol{T_c} + \boldsymbol{s}_1\boldsymbol{\varepsilon}\boldsymbol{T_{sat}} + \boldsymbol{s}_2, \end{aligned} \tag{11}$$

where $\boldsymbol{I}_{n\times n} \in \mathbb{R}^{n\times n}$ represents the identity matrix. Consequently, the manipulator dynamics described in (1), incorporating the new fault model introduced in (11), can be reformulated as follows:

$$\begin{aligned} \ddot{\boldsymbol{q}} = \boldsymbol{I}^{-1}(\boldsymbol{q})[&\boldsymbol{s}_1(\boldsymbol{I}_{n\times n} - \boldsymbol{\varepsilon})\boldsymbol{T_c} + \boldsymbol{s}_2 - \boldsymbol{C_m}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{f}(\dot{\boldsymbol{q}}) - \boldsymbol{G}(\boldsymbol{q}) \\ &- \boldsymbol{T_L} + \boldsymbol{s}_1\boldsymbol{\varepsilon}\boldsymbol{T_{sat}}]\cdot \end{aligned} \tag{12}$$

For convenience, we can consider:

$$\begin{aligned} &\bar{\boldsymbol{\lambda}} = \boldsymbol{s}_1(\boldsymbol{I}_{n\times n} - \boldsymbol{\varepsilon}) = \text{diag}(\bar{\lambda}_1, \ldots, \bar{\lambda}_n), \quad 0 < \bar{\lambda}_i \leq 1 \\ &\boldsymbol{s}_{max} = \boldsymbol{s}_2 + \boldsymbol{s}_1\boldsymbol{\varepsilon}\boldsymbol{T_{sat}}\cdot \end{aligned} \tag{13}$$

where we define a positive constant $\bar{\lambda}_{min} < \inf(\bar{\lambda}_i)$ (see Remark 1). Then, the ultimate expression of the n DoF of a robotic manipulator is, as follows:

$$\ddot{\boldsymbol{q}} = \boldsymbol{I}^{-1}[\bar{\boldsymbol{\lambda}}\boldsymbol{T_c} + \boldsymbol{s}_{max} - \boldsymbol{C_m}\dot{\boldsymbol{q}} - \boldsymbol{f} - \boldsymbol{G} - \boldsymbol{T_L}]\cdot \tag{14}$$

III. Design of the SBFC Strategy

A. Adaptive SBFC Strategy

To apply the subsystem-based control methodology, the dynamics of a manipulator robot, provided in (14), can be transformed into a triangular feedback form as shown below:

$$\begin{cases} \dot{\boldsymbol{x}}_1(t) = \boldsymbol{x}_2(t) \\ \dot{\boldsymbol{x}}_2(t) = \boldsymbol{A}_1\bar{\boldsymbol{\lambda}}\boldsymbol{T_c} + \boldsymbol{g}_1(\boldsymbol{x},t) + \boldsymbol{\Delta}_1(\boldsymbol{x},t) + \boldsymbol{T}_L \end{cases}. \tag{15}$$

Let us define two state variables $\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2]^\top$, $\boldsymbol{x}_1 = \boldsymbol{q}$ as the position vector and $\boldsymbol{x}_2 = \dot{\boldsymbol{q}}$ as the velocity vector. The control torque input incorporates a non-zero coefficient $\boldsymbol{A}_1 = \boldsymbol{I}^{-1}(\boldsymbol{q})$. The term $\boldsymbol{g}_1(\boldsymbol{x},t)$ can be considered established functional elements derived from the system's model, given by $\boldsymbol{I}^{-1}(\boldsymbol{q})(-\boldsymbol{C_m}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{G}(\boldsymbol{q}))$. Meanwhile, $\boldsymbol{\Delta}_1(\boldsymbol{x},t)$ characterizes uncertain aspects arising from incomplete knowledge of system parameters or modeling inaccuracies, expressed as $\boldsymbol{I}^{-1}(\boldsymbol{q})(-\boldsymbol{f}(\dot{\boldsymbol{q}}) + \boldsymbol{s}_{max})$. In continuation of the preceding form, we can define the tracking error $\boldsymbol{e} = [\boldsymbol{e_1}, \boldsymbol{e_2}]^\top$, as follows:

$$\boldsymbol{e}_1 = \boldsymbol{x}_1 - \boldsymbol{x}_d, \quad \boldsymbol{e}_2 = \boldsymbol{x}_2 - \dot{\boldsymbol{x}}_d, \tag{16}$$

where $\boldsymbol{x}_d \in \mathbb{R}^n$ and $\dot{\boldsymbol{x}}_d \in \mathbb{R}^n$ are the position and velocity reference trajectories, and $\boldsymbol{e}_1 : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ and $\boldsymbol{e}_2 : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ are the position and velocity tracking errors, respectively. Now, we can transform the tracking system into a new form:

$$\boldsymbol{Q}_1 = \boldsymbol{e}_1, \quad \boldsymbol{Q}_2 = \boldsymbol{e}_2 - \boldsymbol{\kappa}_1\cdot \tag{17}$$

We introduce the virtual control $\boldsymbol{\kappa}_1 \in \mathbb{R}^n$, as follows:

$$\boldsymbol{\kappa}_1 = -\frac{1}{2}(\delta_1 + \zeta_1\hat{\phi}_1)\boldsymbol{Q}_1, \tag{18}$$

where δ_1 and ζ_1 are positive constants. $\hat{\phi}_1$ is an adaptive function law, which is defined, as follows:

$$\dot{\hat{\phi}}_1 = -k_1\sigma_1\hat{\phi}_1 + \frac{1}{2}\zeta_1 k_1\|\boldsymbol{Q}_1\|^2, \tag{19}$$

where k_1, ζ_1, and σ_1 are positive constants. By derivative of (17) and considering (15) and (16), we will have:

$$\begin{aligned} &\dot{\boldsymbol{Q}}_1 = \boldsymbol{Q}_2 + \boldsymbol{\kappa}_1 \\ &\dot{\boldsymbol{Q}}_2 = \boldsymbol{A}_1\bar{\boldsymbol{\lambda}}\boldsymbol{T_c} + \boldsymbol{g}_1(\boldsymbol{x},t) + \overline{\boldsymbol{\Delta}}_1(\boldsymbol{x},t) + \boldsymbol{T}_L - \ddot{\boldsymbol{x}}_d\cdot \end{aligned} \tag{20}$$

where $\ddot{\boldsymbol{x}}_d \in \mathbb{R}^n$ can be the desired acceleration of the manipulator robot. By assuming the function $\boldsymbol{\kappa}_1$ is smooth, we introduce $\overline{\boldsymbol{\Delta}}_1$, as follows:

$$\overline{\boldsymbol{\Delta}}_1 = \boldsymbol{\Delta}_1(\boldsymbol{x},t) - \frac{\partial\boldsymbol{\kappa}_1}{\partial\boldsymbol{x}_1}\frac{\mathrm{d}\boldsymbol{x}_1}{\mathrm{d}t} - \frac{\partial\boldsymbol{\kappa}_1}{\partial\hat{\phi}_1}\frac{\mathrm{d}\hat{\phi}_1}{\mathrm{d}t}\cdot \tag{21}$$

To prevent the complexity from growing unmanageable, as discussed in Wang et al. [23], we considered the time derivative of the virtual control to be an element of uncertainty in the system.

Assumption (1): If we assume the uncertainties are bounded but vary, there exists a continuously smooth and positive function $r_1 : \mathbb{R}^n \to \mathbb{R}^+$ constrained within the uncertainty bound denoted as $\bar{\boldsymbol{\Delta}}_1$. In addition, there are positive parameters Ω_1, D_{max}, Λ_1, and $\bar{g}_{max} \in \mathbb{R}^+$, which may also be unknown such that:

$$\begin{aligned} &\|\overline{\boldsymbol{\Delta}}_1\| \leq \Lambda_1 r_1, \quad \|\boldsymbol{T}_L\| \leq D_{max} \\ &\|\ddot{\boldsymbol{x}}_d\| \leq \Omega_1, \quad \|\boldsymbol{g}_1(\boldsymbol{x},t)\| \leq \bar{g}_{max}\cdot \end{aligned} \tag{22}$$

Consequently, the actual control $\boldsymbol{T_c}$ is proposed, as follows:

$$\boldsymbol{T_c} = \frac{-1}{2}(\delta_2 + \zeta_2\hat{\phi}_2){I_{min}}^{-1}\boldsymbol{Q}_2, \tag{23}$$

where δ_2 and ζ_2 are positive constants, and we know I_{min} from (2). $\hat{\phi}_2$ is an adaptive function law, which is defined, as follows:

$$\dot{\hat{\phi}}_2 = -k_2\sigma_2\hat{\phi}_2 + \frac{1}{2}\zeta_2 k_2\|\boldsymbol{Q}_2\|^2, \tag{24}$$

where k_2, ζ_2, and σ_2 are positive constants. Assume the adaptive law errors for (19) and (24) are $\tilde{\phi}_j = \hat{\phi}_j - \phi_j^*$ for $j = 1,2$, considering that there are the positive and unknown constants ϕ_1^* and $\phi_2^* \in \mathbb{R}^+$ to compensate for the adaptive estimation errors, as follows:

$$\begin{aligned} \phi_1^* =& \zeta_1^{-1} \\ \phi_2^* =& \zeta_2^{-1}[1 + 2\bar{\lambda}_{min}^{-1}(\mu_1\Lambda_1^2 + \nu_1 D_{max}^2 + \nu_2\Omega_1^2 + \nu_3\bar{g}_{max}^2)]\cdot \end{aligned} \tag{25}$$

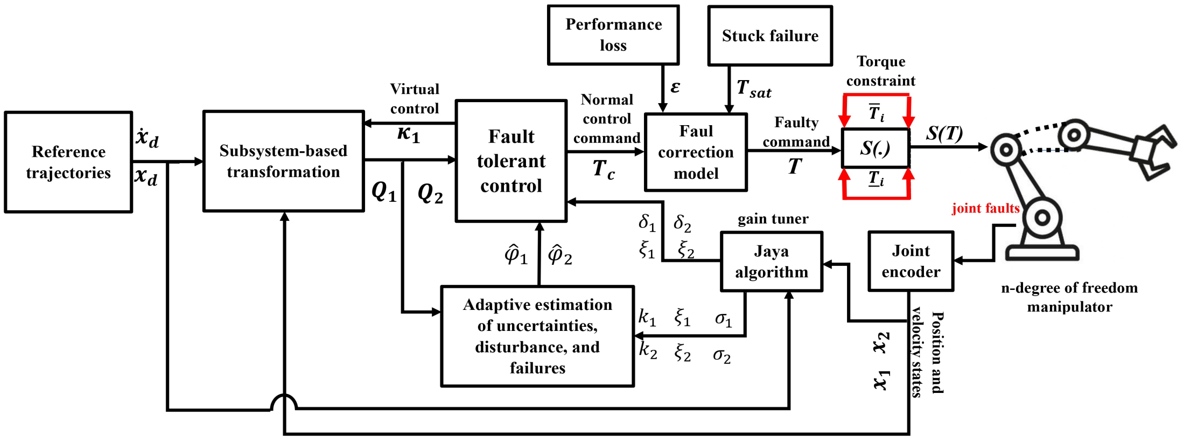

Fig. 1: The interconnection among various sections of the proposed control system.

Apart from ζ_1 and $\zeta_2 \in \mathbb{R}^+$, which are used as control design parameters, all remaining parameters in (25) are assumed positive but unknown constants. Now, we can obtain:

$$\dot{\tilde{\phi}}_j = -k_j \sigma_j \tilde{\phi}_j + \frac{1}{2}\zeta_j k_j \|\boldsymbol{Q}_j\|^2 - k_j \sigma_j \phi_j^*. \tag{26}$$

Lemma (1) [24]: According to the general solution of the given linear first-order ordinary differential equations in (19) and (24), by choosing an initial condition $\hat{\phi}_j(t_0) > 0$, given that the exponential component of $\hat{\phi}_j$ is monotonically decreasing, and considering the positivity of k_j, σ_j, and ζ_j, we assert that, for all $t \geq t_0 = 0$, it is possible to ensure $\hat{\phi}_j(t) > 0$.

Definition (1) [24], [25]: For any initial condition $\boldsymbol{x}(t_0)$, if α, β, and $\tilde{\mu} \in \mathbb{R}^+$ exist, the tracking error $\boldsymbol{e}$ between the state $\boldsymbol{x}$ and the reference states $\boldsymbol{x_r} = [\boldsymbol{x_d}, \dot{\boldsymbol{x}}_{\boldsymbol{d}}]^\top$ converges uniformly and exponentially to a defined region $g(\tau)$, such that:

$$\begin{aligned} &\|\boldsymbol{e}\| = \|\boldsymbol{x}(t) - \boldsymbol{x_r}(t)\| \leq \beta e^{-\alpha(t-t_0)} \|\boldsymbol{x}(t_0) - \boldsymbol{x_r}(t_0)\| + \tilde{\mu} \\ &g(\tau) := \{\boldsymbol{e} \mid \|\boldsymbol{e}\| \leq \tau = \tilde{\mu}\}. \end{aligned} \tag{27}$$

B. JAYA Algorithm-based Parameter Tuning

Given the eight gains in the SBFC, denoted as k_1, k_2, δ_1, δ_2, ζ_1, ζ_2, σ_1, and σ_2, it is necessary to tune each within an iterative function based on the multipopulational JA. Let us consider each gain to be associated with $c \in \mathbb{R}^+$. In this paper, the JA commences by initializing two positive collections of gains of control, known as the initial population, through a random process. For each individual within this population, the cost function is calculated, based on the standard deviation of the position and velocity tracking errors $\bar{e} = \sqrt{\|\boldsymbol{e}_1\|^2 + \|\boldsymbol{e}_2\|^2}$ representing the target objective function to be minimized. The top-performing candidate (c_{best}) is determined as the one with the most favorable value (referred to as $\bar{e}_{\text{best}}$), while the other (the poorest performer) is identified as the candidate (c_{worst}) with the least favorable value (referred to as $\bar{e}_{\text{worst}}$). Next, these values are iteratively adjusted to find the new candidate (c_{new}) in the following iterative function:

$$c_{new} = c + r_1(c_{best} - c) - r_2(c_{worst} - c), \tag{28}$$

where $c_{\text{new}} \in \mathbb{R}^+$ is the updated random c. Further, r_1 and $r_2 \in [0,1]$ are the two random numbers for each variable during the iteration, and c_{best} and c_{worst} are replaced with c_{new} if it gives a better $\bar{e}$ than $\bar{e}_{best}$ or worse $\bar{e}$ than $\bar{e}_{worst}$ values, respectively. All accepted function values at the end of the iteration are maintained, and these values become the input to the next iteration.

Remark (2): The expression $r_1(c_{best} - c)$ represents the inclination of the solution to approach the best solution, while the expression $-r_2(c_{worst} - c)$ signifies the propensity of the solution to eschew the worst solution.

Remark (3): According to the details provided in this paper, all gain parameters must be both positive and finite. To ensure adherence to this requirement, we must first choose initial populations for these parameters to be positive. Then, by following this approach and referring to (28), while bearing in mind that $c > 0$, we can suggest random c be larger than $\frac{r_2 c_{worst} - r_1 c_{best}}{1 - r_1 + r_2}$, as well. In this way, we can guarantee that all newly generated values for c_{new} will remain positive. Furthermore, by incorporating the principle constraint outlined in Eq. (5), we can impose constraints on the JA to prevent it from producing gains that exceed a predetermined threshold, as necessary.

The block diagram shown in Fig. 1 illustrates the interaction among the SBFC system sections. As depicted in the figure, the system computes variables related to subsystem-based transformation upon receiving reference trajectories. In addition, the adaptation mechanisms estimate upper bounds for disturbances, uncertainties, and actuator failures. Then, the calculated values from the subsystem-based transformation component, along with the parameters estimated through online adaptation update laws, are received by the proposed controller. Subsequently, the standard control command, denoted as $\boldsymbol{T_c}$, is generated for the novel correction fault model, which incorporates torque constraints. The input constraint section, $\boldsymbol{S(T)}$, ensures that both faulty and normal torque values do not surpass the defined constraints. Notably, the eight gains of the adaptation law and controller are automatically adjusted using the JA block in accordance with a cost function that includes the motion sensor data of the robot manipulator and the reference trajectories.

C. Stability Analysis

Theorem (1): Consider the adaptive algorithm presented in Eq. (19), and (24), the faulty dynamic model for the robotic manipulator in Eqs. (14), and the control input as given in (23). It is assumed that under these conditions, the states $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$ can attain the reference trajectories $\boldsymbol{x_d}$ and $\dot{\boldsymbol{x}}_{\boldsymbol{d}}$ through UES, as defined in Definition (3).

Proof: A Lyapunov function is suggested as follows:

$$V_1 = \frac{1}{2}\ \bar{\lambda}_{min}[\boldsymbol{Q}_1^\top \boldsymbol{Q}_1 + k_1^{-1}\tilde{\phi}_1^2]\cdot \tag{29}$$

where we define $\bar{\lambda}_{min}$ a positive constant that is less than the infimum of $\bar{\lambda}_i$ provided in (13). After differentiating V_1 and inserting (20), we obtain:

$$\dot{V}_1 = \overline{\lambda}_{min}\boldsymbol{Q}_1^\top[\boldsymbol{Q}_2 + \boldsymbol{\kappa}_1] + k_1^{-1}\overline{\lambda}_{min}\tilde{\phi}_1\dot{\hat{\phi}}_1\cdot \tag{30}$$

By using the Cauchy–Schwarz and the squared Euclidean norm concepts:

$$\dot{V}_1 \leq \frac{1}{2}\overline{\lambda}_{min}(\|\boldsymbol{Q}_1\|^2 + \|\boldsymbol{Q}_2\|^2) + \overline{\lambda}_{min}(\boldsymbol{Q}_1^\top \boldsymbol{\kappa}_1 + k_1^{-1}\tilde{\phi}_1\dot{\hat{\phi}}_1)\cdot \tag{31}$$

Then, by considering the definition of ϕ_1^* in (25), we achieve:

$$\begin{aligned}\dot{V}_1 \leq& \frac{1}{2}\overline{\lambda}_{min}\|\boldsymbol{Q}_2\|^2 + \frac{1}{2}\overline{\lambda}_{min}\zeta_1\phi_1^*\|\boldsymbol{Q}_1\|^2 + \overline{\lambda}_{min}\boldsymbol{Q}_1^\top\boldsymbol{\kappa}_1 \\ &+ k_1^{-1}\overline{\lambda}_{min}\tilde{\phi}_1\dot{\hat{\phi}}_1\cdot\end{aligned} \tag{32}$$

Now, by inserting $\dot{\hat{\phi}}_1$ and $\boldsymbol{\kappa}_1$ from (26) and (18), we obtain:

$$\begin{aligned}\dot{V}_1 \leq& \frac{1}{2}\overline{\lambda}_{min}\|\boldsymbol{Q}_2\|^2 + \frac{1}{2}\overline{\lambda}_{min}\zeta_1\phi_1^*\|\boldsymbol{Q}_1\|^2 - \frac{1}{2}\overline{\lambda}_{min}\delta_1\|\boldsymbol{Q}_1\|^2 \\ &- \frac{1}{2}\overline{\lambda}_{min}\zeta_1\hat{\phi}_1\|\boldsymbol{Q}_1\|^2 - \overline{\lambda}_{min}\sigma_1\tilde{\phi}_1^2 + \frac{1}{2}\overline{\lambda}_{min}\zeta_1\|\boldsymbol{Q}_1\|^2\tilde{\phi}_1 \\ &- \overline{\lambda}_{min}\sigma_1\phi_1^*\tilde{\phi}_1\cdot\end{aligned} \tag{33}$$

Because $\tilde{\phi}_1 = \hat{\phi}_1 - \phi_1^*$:

$$\begin{aligned}\dot{V}_1 \leq& \frac{1}{2}\overline{\lambda}_{min}\|\boldsymbol{Q}_2\|^2 - \frac{1}{2}\overline{\lambda}_{min}\delta_1\|\boldsymbol{Q}_1\|^2 - \overline{\lambda}_{min}\sigma_1\tilde{\phi}_1^2 \\ &- \overline{\lambda}_{min}\sigma_1\phi_1^*\tilde{\phi}_1\cdot\end{aligned} \tag{34}$$

After dividing $\overline{\lambda}_{min}\sigma_1\tilde{\phi}_1^2$ into $\frac{1}{2}\overline{\lambda}_{min}\sigma_1\tilde{\phi}_1^2 + \frac{1}{2}\overline{\lambda}_{min}\sigma_1\tilde{\phi}_1^2$, and considering (29), we obtain:

$$\begin{aligned}\dot{V}_1 \leq& -\Psi_1 V_1 + \frac{1}{2}\overline{\lambda}_{min}\|\boldsymbol{Q}_2\|^2 - \frac{1}{2}\overline{\lambda}_{min}\sigma_1\tilde{\phi}_1^2 \\ &- \overline{\lambda}_{min}\sigma_1\phi_1^*\tilde{\phi}_1,\end{aligned} \tag{35}$$

where $\Psi_1 = \min[\delta_1, \quad k_1\sigma_1]$. As $-\frac{1}{2}\bar{\lambda}_{min}\sigma_1\hat{\phi}_1^2 \leq 0$, we eliminate it and reach:

$$\dot{V}_1 \leq -\Psi_1 V_1 + \frac{1}{2}\bar{\lambda}_{min}\|\boldsymbol{Q}_2\|^2 + \frac{1}{2}\bar{\lambda}_{min}\sigma_1\phi_1^{*2}\cdot \tag{36}$$

Likewise, the Lyapunov function V_2 is suggested as follows:

$$V_2 = V_1 + \frac{1}{2}\ [\boldsymbol{Q}_2^\top \boldsymbol{Q}_2 + k_2^{-1}\bar{\lambda}_{min}\tilde{\phi}_2^2]\cdot \tag{37}$$

By differentiating V_2 and inserting (20), we obtain:

$$\begin{aligned}\dot{V}_2 \leq& -\Psi_1 V_1 + \frac{1}{2}\bar{\lambda}_{min}\|\boldsymbol{Q}_2\|^2 + \frac{1}{2}\bar{\lambda}_{min}\sigma_1\phi_1^{*2} + k_2^{-1}\bar{\lambda}_{min}\tilde{\phi}_2\dot{\hat{\phi}}_2 \\ &+ \boldsymbol{Q}_2^\top[\boldsymbol{A}_1\bar{\boldsymbol{\lambda}}\boldsymbol{T_c} + \boldsymbol{g}_1(\boldsymbol{x},t) + \bar{\boldsymbol{\Delta}}_1(\boldsymbol{x},t) + \boldsymbol{T}_L - \ddot{\boldsymbol{x}}_d]\cdot\end{aligned} \tag{38}$$

Then, by inserting $\boldsymbol{T_c}$ from (23):

$$\begin{aligned}\dot{V}_2 \leq& -\Psi_1 V_1 + \frac{1}{2}\bar{\lambda}_{min}\|\boldsymbol{Q}_2\|^2 + \frac{1}{2}\bar{\lambda}_{min}\sigma_1\phi_1^{*2} \\ &- \frac{1}{2}\bar{\lambda}_{min}\delta_2\|\boldsymbol{Q}_2\|^2 - \frac{1}{2}\bar{\lambda}_{min}\zeta_2\hat{\phi}_2\|\boldsymbol{Q}_2\|^2 + \boldsymbol{Q}_2^\top\boldsymbol{g}_1 \\ &+ \boldsymbol{Q}_2^\top\bar{\boldsymbol{\Delta}}_1 - \boldsymbol{Q}_2^\top\ddot{\boldsymbol{x}}_d + \boldsymbol{Q}_2^\top\boldsymbol{T}_L + k_2^{-1}\bar{\lambda}_{min}\tilde{\phi}_2\dot{\hat{\phi}}_2\cdot\end{aligned} \tag{39}$$

Now, by assuming that μ_1, ν_1, ν_2, and ν_3 are positive constants, according to Young's inequality, we can argue:

$$\begin{aligned}\boldsymbol{Q}_2^\top\bar{\boldsymbol{\Delta}}_1 \ &\leq\ \mu_1\Lambda_1^2\|\boldsymbol{Q}_2\|^2 + \frac{1}{4}\mu_1^{-1}r_1^2 \\ \boldsymbol{Q}_2^\top\boldsymbol{T}_L \ &\leq \nu_1 D_{max}^2\|\boldsymbol{Q}_2\|^2 + \frac{1}{4}\nu_1^{-1} \\ -\boldsymbol{Q}_2^\top\ddot{\boldsymbol{x}}_d \ &\leq\ \nu_2\Omega_1^2\|\boldsymbol{Q}_2\|^2 + \frac{1}{4}\nu_2^{-1} \\ \boldsymbol{Q}_2^\top\boldsymbol{g}_1(\boldsymbol{x},t) \ &\leq\ \nu_3\bar{g}_{max}^2\|\boldsymbol{Q}_2\|^2 + \frac{1}{4}\nu_3^{-1}\cdot\end{aligned} \tag{40}$$

Because we have ϕ_2^* from (25), we can obtain:

$$\begin{aligned}\dot{V}_2 \leq& -\Psi_1 V_1 + \frac{1}{2}\bar{\lambda}_{min}\sigma_1\phi_1^{*2} + \frac{1}{4}\mu_1^{-1}r_1^2 + \frac{1}{2}\overline{\lambda}_{min}\zeta_2\phi_2^*\|\boldsymbol{Q}_2\|^2 \\ &- \frac{1}{2}\bar{\lambda}_{min}\delta_2\|\boldsymbol{Q}_2\|^2 - \frac{1}{2}\bar{\lambda}_{min}\zeta_2\hat{\phi}_2\|\boldsymbol{Q}_2\|^2 + \frac{1}{4}\nu_1^{-1} + \frac{1}{4}\nu_2^{-1} \\ &+ \frac{1}{4}\nu_3^{-1} + k_2^{-1}\bar{\lambda}_{min}\tilde{\phi}_2\dot{\hat{\phi}}_2\cdot\end{aligned} \tag{41}$$

In addition, by inserting $\dot{\hat{\phi}}_2$ from (26) into

$$\begin{aligned}\dot{V}_2 \leq& -\Psi_1 V_1 + \frac{1}{2}\sum_{k=1}^{2}\bar{\lambda}_{min}\sigma_k\phi_k^{*2} + \frac{1}{4}\mu_1^{-1}r_1^2 - \frac{1}{2}\bar{\lambda}_{min}\delta_2\|\boldsymbol{Q}_2\|^2 \\ &+ \frac{1}{4}\sum_{k=1}^{3}\nu_k^{-1} - \overline{\lambda}_{min}\sigma_1\tilde{\phi}_1^2 - \overline{\lambda}_{min}\sigma_1\phi_1^*\tilde{\phi}_1\cdot\end{aligned} \tag{42}$$

Like (36), and firm (37), we can obtain:

$$\dot{V}_2 \leq -\Psi_2 V_2 + \frac{1}{4}\mu_1^{-1}r_1^2 + \frac{1}{4}\sum_{k=1}^{3}\nu_k^{-1} + \frac{1}{2}\sum_{k=1}^{2}\bar{\lambda}_{min}\sigma_k\phi_k^{*2}, \tag{43}$$

where $\Psi_2 = \min[\Psi_1, \quad \bar{\lambda}_{min}\delta_2, \quad k_2\sigma_2]$. Thus, considering $V = V_2$, we can argue:

$$V = \frac{1}{2}\bar{\lambda}_{min}\boldsymbol{Q}^\top\Upsilon\boldsymbol{Q} + \frac{1}{2}\bar{\lambda}_{min}\tilde{\boldsymbol{\phi}}^\top\boldsymbol{K}^{-1}\tilde{\boldsymbol{\phi}}, \tag{44}$$

where:

$$\begin{aligned}\boldsymbol{Q} &= \begin{bmatrix}\boldsymbol{Q}_1\\ \boldsymbol{Q}_2\end{bmatrix}, \quad \Upsilon = \begin{bmatrix}1 & 0\\ 0 & \bar{\lambda}_{\min}^{-1}\end{bmatrix}, \\ \tilde{\boldsymbol{\phi}} &= \begin{bmatrix}\tilde{\phi}_1\\ \tilde{\phi}_2\end{bmatrix}, \quad \boldsymbol{K}^{-1} = \begin{bmatrix}k_1^{-1} & 0\\ 0 & k_2^{-1}\end{bmatrix}\cdot\end{aligned} \tag{45}$$

Thus, according to (43), we obtain:

$$\dot{V} \leq -\Psi_2 V + \frac{1}{4}\mu_1^{-1}r_1^2 + \tilde{\mu}, \tag{46}$$

where $\tilde{\mu} = \frac{1}{4}\sum_{k=1}^{3}\nu_k^{-1} + \frac{1}{2}\sum_{k=1}^{2}\bar{\lambda}_{min}\sigma_k\phi_k^{*2}$. In this section, we should recall the following solution (see Lemma 1):

$$\begin{aligned}\dot{V} &= \Psi V + \mu r \\ V &= e^{\Psi t}V(0) + \int_0^t e^{\Psi(t-\tau)}\mu r(\tau)d\tau\cdot\end{aligned} \tag{47}$$

In the same way, we can solve (46), as follows:

$$\begin{aligned}V \leq& V(t_0)e^{-\{\Psi_2(t-t_0)\}} + \frac{1}{4}\mu_1^{-1}\int_{t_0}^{t}e^{\{-\Psi_2(t-T)\}}r_1^2\ \ dT \\ &+ \tilde{\mu}\int_{t_0}^{t}e^{\{-\Psi_2(t-T)\}}\ \ dT\cdot\end{aligned} \tag{48}$$

Considering (37), we know: $\frac{1}{2}\,\boldsymbol{Q}_2^{\top}\boldsymbol{Q}_2 \leq V_2 = V$. Thus, we can interpret (48) as follows:

$$\|\boldsymbol{Q}\|^2 \leq 2V(t_0)\,e^{-\{\Psi_2(t-t_0)\}} + \frac{1}{2}\mu_1^{-1}\int_{t_0}^{t} e^{\{-\Psi_2(t-T)\}} r_1^2 \; dT + 2\;\tilde{\mu}\;\Psi_2{}^{-1}. \tag{49}$$

Both μ_1, and Ψ_2 are positive constants dependent on designable control gains that are freely chosen to satisfy the following condition: $\frac{1}{2} < \mu_1\Psi_2$. To continue the stability proof, we define a continuous operator $Z(\cdot)$, as follows:

$$Z(\iota) = \frac{1}{2}\frac{1}{\mu_1(\Psi_2-\iota)} > 0, \quad \iota \in [0,\Psi_2)\cdot \tag{50}$$

It is evident that by increasing ι, $Z(\iota)$ increases, meaning that: $Z(\iota) \geq Z(0) = \frac{1}{2}\frac{1}{\mu_1\Psi_2}$. Because μ_1 is a positive constant, it becomes evident that we can find a small positive value $\bar{\iota} \in \iota$ which $\bar{\iota}\mu_1 < \frac{1}{2}$. By knowing that $\frac{1}{2} < \mu_1\Psi_2$, we can obtain: $0 < \bar{\iota} < \frac{1}{2\mu_1}$. Then, we can obtain: $0 < Z(\bar{\iota}) = \frac{1}{2}\frac{\mu_1^{-1}}{\Psi_2-\bar{\iota}} < 1$. If we say $\bar{Z} = Z(\bar{\iota})$, by multiplying $e^{\bar{\iota}(t-t_0)}$ by (49), we reach:

$$\|\boldsymbol{Q}\|^2 e^{\bar{\iota}(t-t_0)} \leq 2V(t_0)e^{-(\Psi_2-\bar{\iota})(t-t_0)} + 2\tilde{\mu}\Psi_2^{-1}e^{\bar{\iota}(t-t_0)} + \frac{1}{2}\mu_1^{-1}\int_{t_0}^{t} e^{-\Psi_2(t-T)+\bar{\iota}(t-t_0)} r_1^2 \; dT\cdot \tag{51}$$

Because $0 \leq \bar{\iota} < \Psi_2$, we can eliminate the decreasing element $e^{-(\Psi_2-\bar{\iota})(t-t_0)}$ from (51):

$$\|\boldsymbol{Q}\|^2 e^{\bar{\iota}(t-t_0)} \leq 2V(t_0) + 2\tilde{\mu}\Psi_2^{-1}e^{\bar{\iota}(t-t_0)} + \frac{1}{2}\mu_1^{-1}\int_{t_0}^{t} e^{-(\Psi_2-\bar{\iota})(t-T)} r_1^2 e^{\bar{\iota}(t-t_0)} \; dT\cdot \tag{52}$$

We represent the non-decreasing and continuous functions E_0 and E_1, as follows:

$$E_0(t) = \sup_{\omega\in[t_0,t]} \;[\|\boldsymbol{Q}(\omega)\|^2 e^{\bar{\iota}(\omega-t_0))}] \qquad E_1(t) = \sup_{\omega\in[t_0,t]} \;[(r_1^2)e^{\bar{\iota}(\omega-t_0)}]\cdot \tag{53}$$

Then, by considering Eqs. (52) and (53), we achieve:

$$\|\boldsymbol{Q}\|^2 e^{\bar{\iota}(t-t_0)} \leq 2V(t_0) + \frac{1}{2}\frac{\mu_1^{-1}}{\Psi_2-\bar{\iota}}E_1 + 2\tilde{\mu}\Psi_2^{-1}e^{\bar{\iota}(t-t_0)}. \tag{54}$$

Because E_1 is non-decreasing, the left-hand side of Eq. (54) will also not decrease. Hence, with respect to the definition of E_0 in Eq. (53), we can conclude:

$$E_0 \leq 2V(t_0) + \frac{1}{2}\frac{\mu_1^{-1}}{\Psi_2-\bar{\iota}}E_1 + 2\tilde{\mu}\Psi_2^{-1}e^{\bar{\iota}(t-t_0)}. \tag{55}$$

By defining $E = \max(E_0, E_1)$, we can obtain:

$$E_0 \leq 2V(t_0) + \bar{Z}E + 2\tilde{\mu}\Psi_2^{-1}e^{\bar{\iota}(t-t_0)}, \tag{56}$$

such that both E_0 and E are not decreasing. By choosing Ψ_2 large enough, which relies on control gains, and selecting $\bar{\iota}$ small enough, it becomes possible to ensure the existence of a sufficiently large $\Psi_2 > \iota^* > \bar{\iota}$ such that $\overset{*}{Z} = Z(\iota^*)$ and satisfy the following condition [24]:

$$\overset{*}{Z} > \bar{Z} \;\;,\;\; 0 < \overset{*}{Z} < 1 \implies \bar{Z}E \leq \overset{*}{Z}E_0\cdot \tag{57}$$

(57) is justified when we have $\iota^* \geq \frac{E_0}{E}\psi_2 - (\bar{\iota}+\psi_2)$ which is straightforward. Incorporating Eq. (57) into Eq. (56), we arrive at:

$$E_0 \leq 2V(t_0) + \overset{*}{Z}E_0(t) + 2\tilde{\mu}\Psi_2^{-1}e^{\bar{\iota}(t-t_0)}. \tag{58}$$

Afterward, we obtain:

$$E_0 \leq \frac{2V(t_0) + 2\tilde{\mu}\Psi_2^{-1}e^{\bar{\iota}(t-t_0)}}{1-\overset{*}{Z}}\cdot \tag{59}$$

Concerning the definition (53), we obtain:

$$\|\boldsymbol{Q}\|^2 \leq \frac{2V(t_0)\,e^{-\bar{\iota}(t-t_0)} + 2\tilde{\mu}\Psi_2^{-1}}{1-\overset{*}{Z}}\cdot \tag{60}$$

It is significant that:

$$\sup_{t\in[t_0,\infty]} \left(\frac{2V(t_0)e^{-\bar{\iota}(t-t_0)}}{1-\overset{*}{Z}}\right) \leq \frac{2V(t_0)}{1-\overset{*}{Z}}\cdot \tag{61}$$

Consequently, by Definition (1), it is evident from Eq. (60) that $||\boldsymbol{Q}||$ is uniformly and exponentially stable towards a specific ball $\mathscr{G}(\bar{\tau}_0)$ when employing the SBFC approach, such that:

$$\mathscr{G}(\bar{\tau}_0) := \left\{\boldsymbol{Q} \mid \|\boldsymbol{Q}\| \leq \bar{\tau}_0 = \sqrt{\frac{2\tilde{\mu}\Psi_2^{-1}}{1-\overset{*}{Z}}}\right\}\cdot \tag{62}$$

Remark (4): To the best of the authors' knowledge, there are no explicit upper or lower limits to the DoF for the SBFC-applied manipulator. Although the stability of the manipulator is ensured under the conditions defined for faulty actuators, it is important to note that an increase in the number of faulty actuators deteriorates the tracking accuracy of the manipulator's end-effector. Therefore, until the tracking accuracy of the end-effector is deemed satisfactory, there are no constraints on the DoF and the number of faulty actuators for the proposed control strategies.

IV. Numerical Validity

To evaluate the efficacy of the proposed methodology, we applied it to the 2-DoF vertical plane robot featured in the work by Humaloja et al. [26] with link lengths 1m and 0.8m. The robotic manipulator dynamic and the SBFC strategy were implemented, running at a 10-KHz frequency. The modeling of the unknown friction and external disturbance term is represented, as:

$$\boldsymbol{\Delta_1} + \boldsymbol{T_L} = \begin{bmatrix} 0.6\sin(0.8\dot{q}_1 q_2) + 3\sin(2t) \\ -1.6\sin(1.8q_2) + 1.3\sin(0.7\dot{q}_2) - 0.2q_2 \end{bmatrix}\cdot \tag{63}$$

The system's desired trajectory, based on radians, is chosen as follows:

$$\boldsymbol{x_d} = [\sin(t/4\pi) - 1, \sin(t/4\pi + \frac{\pi}{3})]^T\cdot \tag{64}$$

In this case study, we examined a fault model that occurred in both actuators, as represented in Fig. 2, where both actuators were initially in healthy and normal condition for up to 10 seconds. The effectiveness of JA and tracking control during this healthy task is illustrated in Fig. 3. The JA

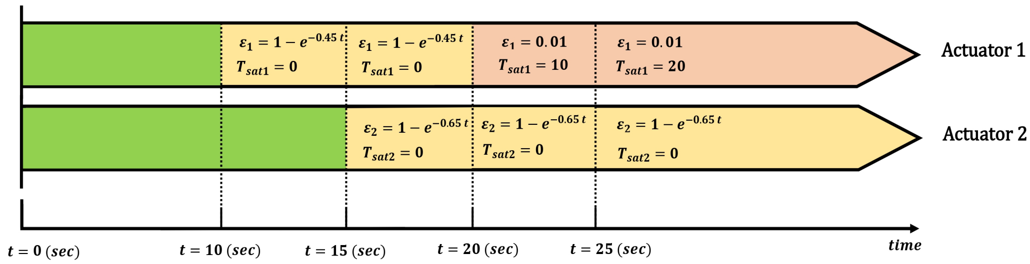

Fig. 2: Actuator fault occurrence (seconds): normal: $[0,10]$; one faulty actuator: $[10,15]$, and two faulty actuators: $[15,\infty]$.

commenced by initializing two positive collections of gains of control, through a random process. The step time for updating the candidates of the JA was set at 0.0001 seconds. This means that during each step, the control gains are updated to converge to the suboptimal values. The best SBFC gains obtained at 0.13 seconds for the mentioned manipulator and the specified task are as follows: $\delta_1 = 62$, $\delta_2 = 75$, $\zeta_1 = 0.2$, $\zeta_2 = 3.5$, $\sigma_1 = 5.6$, $\sigma_2 = 1.9$, $k_1 = 1.4$, and $k_2 = 0.96$. This depiction indicates that control parameters were suitably tuned, leading to the cost function in Fig. 3(a) reaching a minimum value at 0.25 sec. Fig. 3(b) also illustrates the potential for position tracking using SBFC, in the presence of uncertainties, in the healthy actuator mode, following parameter tuning.

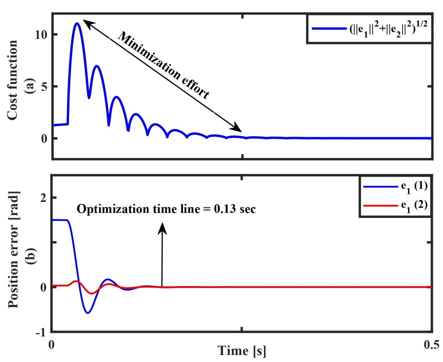

Fig. 3: Cost function (a) and tracking position error (b)

Fig. 4 illustrates the torque efforts generated by healthy and faulty joints to maintain control performance. The torque effort significantly decreased until 10 seconds before faults occurred (see Fig. 2). The first actuator experiences a variety of faults after 10 seconds, while faults in the second actuator begin at 15 seconds, although each actuator fault affects the other actuator as well. This results in the generation of more frequent torque to compensate for faults. The severe faults for the first and second actuators occur at 20 and 15 seconds, respectively. It demonstrates the effectiveness of the designed constraints, even for faulty actuators, in preventing torques larger than the defined nominal values (80Nm). Despite these challenges, control performance is successfully maintained for around 26 seconds. This is evidenced by the fact that, although the last faults persist without further change, the adaptation control efforts generated by the actuators decrease significantly, ultimately reaching the control goal. Fig. 5 illustrates the system's response to the fault model mentioned in Fig. 2 in terms of the objective function and position tracking error. This demonstrates its capability to effectively reduce tracking errors to zero, even in the presence of faults in both actuators.

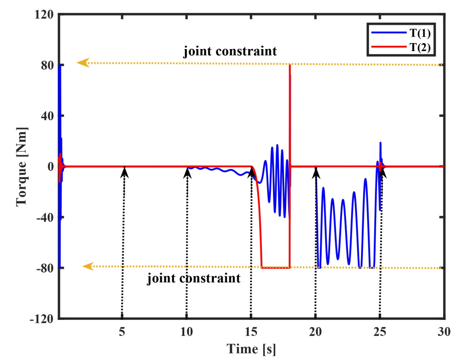

Fig. 4: Torque effort generated by healthy and faulty joints.

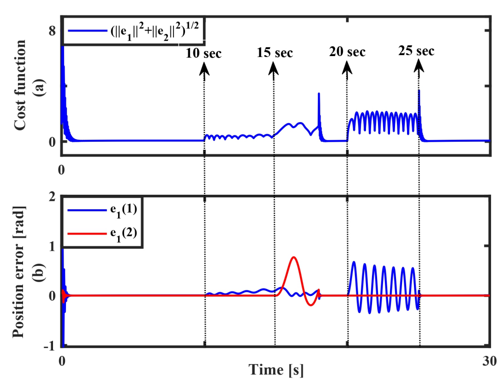

Fig. 5: Cost function (a) and tracking position error (b)

To investigate the performance of the SBFC method, we applied the same conditions to two other fault-tolerant control methods as provided in [27] and [28], which ensured asymptotic stability under identical tasks. All tasks were subject to uncertainties as defined in (63) and followed the reference trajectory (64). Additionally, we attempted to obtain suboptimal values for all three control algorithms using both Teaching Learning (TL) based optimization and

JA. Neither parameter optimization method requires specific algorithmic parameters. However, unlike JA, which consists of one phase, TL involves two phases, making JA simpler to implement. Despite SBFC demonstrating stronger stability (UES), results summarized in Table I indicate superior tracking performance when employing JA-tuning SBFC for the studied manipulator.

TABLE I: Tracking performance of the SBFC and fault-tolerant control methods provided in [27] and [28] tuned by JA and TL parameter optimization algorithms under various actuators' statuses.

Actuators' status	Tracking criteria	SBFC approach	[27] approach	[28] approach
Normal	**JA-tuning error (rad)**	**0.0009**	**0.0012**	**0.0025**
	TL-tuning error (rad)	**0.0013**	**0.0019**	**0.0045**
	JA-tuning time (sec)	**0.13**	**0.21**	**0.25**
	TL-tuning time (sec)	**0.24**	**0.22**	**0.29**
one-faulty	**JA-tuning error (rad)**	**0.0012**	**0.0031**	**0.0030**
	TL-tuning error (rad)	**0.0015**	**0.0035**	**0.0035**
	JA-tuning time (sec)	**1.1**	**2.81**	**2.2**
	TL-tuning time (sec)	**1.13**	**3.22**	**2.35**
two-faulty	**JA-tuning error (rad)**	**0.0018**	**0.0036**	**0.0038**
	TL-tuning error (rad)	**0.0035**	**0.0058**	**0.0042**
	JA-tuning time (sec)	**1.22**	**2.93**	**2.35**
	TL-tuning time (sec)	**2.33**	**4.03**	**3.45**

V. Conclusions

This study mathematically developed the conventional N-DoF robot manipulator dynamic model to address various types of actuator functions: normal operation (healthy mode), stuck failure, performance degradation (including both incipient and abrupt faults), and saturation (over-generated torque). Moreover, to ensure the joints' states track desired trajectories despite unknown modeling errors, external disturbances, and actuator faults, a novel SBFC mechanism was proposed. The parameters of this mechanism were tuned using a multi-population and single-phase swarm intelligence technique. This technique is distinguished by its ability to continuously approach optimal control levels without the need for meticulous tuning of algorithm-specific parameters, relying instead on its inherent principles. Ultimately, the use of the SBFC ensured UES for N-DoF manipulators in the presence of uncertainties and actuator faults. Looking forward, this generic control methodology opens new avenues for application across various robotic dynamics, suggesting broader implications for future research.

References

[1] W.-H. Zhu, *Virtual Decomposition Control: Toward Hyper Degrees of Freedom Robots*. STAR, vol. 60, 2010.

[2] J. Koivumäki, J.-P. Humaloja, L. Paunonen, W.-H. Zhu, and J. Mattila, "Subsystem-based control with modularity for strict-feedback form nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. 68, no. 7, pp. 4336–4343, 2023.

[3] Y.-X. Li, X. Li, and S. Tong, "Backstepping-based fuzzy adaptive stabilization of reaction-diffusion equation with state constraints," *IEEE Trans. Cybern.*, 2022.

[4] A. A. Siqueira, M. H. Terra, and M. Bergerman, *Robust Control of Robots: Fault Tolerant Approaches*. SSBM, 2011.

[5] R. Isermann, *Fault-Diagnosis Applications: Model-Based Condition Monitoring: Actuators, Drives, Machinery, Plants, Sensors, and Fault-Tolerant Systems*. SSBM, 2011.

[6] M. H. Shahna and M. Abedi, "An anti-unwinding finite time fault tolerant sliding mode control of a satellite based on accurate estimation of inertia moments," *ISA Trans.*, vol. 101, pp. 23–41, 2020.

[7] C. Ke, K.-Y. Cai, and Q. Quan, "Uniform passive fault-tolerant control of a quadcopter with one, two, or three rotor failure," *Trans. Robot.*, 2023.

[8] M. Van and S. S. Ge, "Adaptive fuzzy integral sliding-mode control for robust fault-tolerant control of robot manipulators with disturbance observer," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 5, pp. 1284–1296, 2021.

[9] Z. Anjum and Y. Guo, "Finite time fractional-order adaptive backstepping fault tolerant control of robotic manipulator," *Int. J. Control Autom. Syst.*, vol. 19, pp. 301–310, 2021.

[10] W. He, T. Meng, D. Huang, and X. Li, "Adaptive boundary iterative learning control for an euler–bernoulli beam system with input constraint," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1539–1549, 2017.

[11] H. Du and N. Zhang, "Fuzzy control for nonlinear uncertain electrohydraulic active suspensions with input constraint," *IEEE Trans. Fuzzy Syst.*, vol. 17, no. 2, pp. 343–356, 2008.

[12] H. R. Nohooji, "Constrained neural adaptive pid control for robot manipulators," *J. Franklin Inst.*, vol. 357, no. 7, pp. 3907–3923, 2020.

[13] D. Yang, T. Li, H. Zhang, and X. Xie, "Event-trigger-based robust control for nonlinear constrained-input systems using reinforcement learning method," *Neurocomputing*, vol. 340, pp. 158–170, 2019.

[14] M. H. Shahna, S. A. A. Kolagar, and J. Mattila, "Obstacle avoidance deep reinforcement learning-based trajectory planner with robust low-level control for robotic manipulators," *arXiv preprint arXiv:2402.02551*, 2024.

[15] A. R. Kashani, C. V. Camp, M. Rostamian, K. Azizi, and A. H. Gandomi, "Population-based optimization in structural engineering: a review," *Artif. Intell. Rev.*, pp. 1–108, 2022.

[16] R. V. Rao and A. Saroj, "A self-adaptive multi-population based jaya algorithm for engineering optimization," *Swarm Evol. Comput.*, vol. 37, pp. 1–26, 2017.

[17] R. Rao, "Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems," *Int. J. Ind. Eng. Comput.*, vol. 7, no. 1, pp. 19–34, 2016.

[18] J. Nanda, S. Mishra, and L. C. Saikia, "Maiden application of bacterial foraging-based optimization technique in multiarea automatic generation control," *Trans. Power Syst.*, vol. 24, no. 2, pp. 602–609, 2009.

[19] E. H. Houssein, A. G. Gad, and Y. M. Wazery, "Jaya algorithm and applications: A comprehensive review," *MOC & EE*, pp. 3–24, 2021.

[20] N. Bansal, A. Bisht, S. Paluri, V. Kumar, K. Rana, A. T. Azar, and S. Vaidyanathan, *Backstepping Control of Nonlinear Dynamical Systems*. ANDC, pp 375–406, 2011.

[21] M. H. Shahna and M. Abedi, "Design of a finite time passivity based adaptive sliding mode control implementing on a spacecraft attitude dynamic simulator," *Control Eng. Pract.*, vol. 114, p. 104866, 2021.

[22] M. Van, S. S. Ge, and H. Ren, "Finite time fault tolerant control for robot manipulators using time delay estimation and continuous nonsingular fast terminal sliding mode control," *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1681–1693, 2017.

[23] F. Wang, C. Hua, and Q. Zong, "Attitude control of reusable launch vehicle in reentry phase with input constraint via robust adaptive backstepping control," *Int. J. Adapt. Control Signal Process.*, vol. 29, no. 10, pp. 1308–1327, 2015.

[24] M. H. Shahna, M. Bahari, and J. Mattila, "Robust decomposed system control for an electro-mechanical linear actuator mechanism under input constraints," *Int. J. Robust Nonlin.*, 2024.

[25] M. Corless and G. Leitmann, "Bounded controllers for robust exponential convergence," *J. Optim. Theory Appl.*, vol. 76, no. 1, pp. 1–12, 1993.

[26] J.-P. Humaloja, J. Koivumäki, L. Paunonen, and J. Mattila, "Decentralized observer design for virtual decomposition control," *Trans. Automat. Contr.*, vol. 67, no. 5, pp. 2529–2536, 2021.

[27] M. Van, M. Mavrovouniotis, and S. S. Ge, "An adaptive backstepping nonsingular fast terminal sliding mode control for robust fault tolerant control of robot manipulators," *IEEE Trans. Syst.*, vol. 49, no. 7, pp. 1448–1458, 2018.

[28] Z. S. Awan, K. Ali, J. Iqbal, and A. Mehmood, "Adaptive backstepping based sensor and actuator fault tolerant control of a manipulator," *J. Electr. Eng. Technol.*, vol. 14, pp. 2497–2504, 2019.