A Generalized Index for Fault-Tolerant Control in Operational Space Under Free-Swinging Actuation Failure

Lee, Yisoo; Tsagarakis, Nikos; Ott, Christian; Lee, Jinoh

A Generalized Index for Fault-tolerant Control in Operational Space under Free-swinging Actuation Failure

Yisoo Lee[1], Nikos Tsagarakis[2], Christian Ott[3], and Jinoh Lee[3,†]

***Abstract*—Actuation failure and fault-tolerant control under the actuation failure scenario have drawn more attention in accordance with the recent increasing demand for reliable robot control applications such for long-term and remote operation. The emergence of control torque loss, i.e., the free-swinging failure, is particularly challenging when the robot performs dynamic operational space tasks due to complexities stemming from redundancies in the kinematic structure as well as a dynamical disturbance in the under-actuated multi-body system. To reinforce robustness and accuracy of task-space control under the failure condition, this paper proposes a performance index, named generalized failure-susceptibility (GFS), which is formulated to render thorough dynamic and kinematic effects caused by the un-actuated joints. The GFS index is then exploited with the hierarchical task controller, where self-motion is controlled to minimize the index in real-time. Several experiments with a seven-degrees-of-freedom torque-controlled robot verify that the proposed control strategy with the GFS index effectively improves fault tolerance against anticipating actuation failure.**

***Index Terms*—Failure detection and recovery, redundant Robots, underactuated robots.**

I. Introduction

INCREASING demand for modern robots to perform a task in hazardous, complex, and remote environments escalates the potential risk of hardware failure and urges robustness of the robot operating in abnormal situations with actuator, system, and sensor failures [1]. In robotic systems, the failure even in a single actuator or sensor can bring out critical damage to the entire system since its components are highly coupled in both hardware and software perspectives. Thus, fault-tolerant control methods against the failure problem are requested for reliable and safe robot control [2].

Manuscript received: September 4, 2021; Revised December 4, 2021; Accepted January 1, 2022. This paper was recommended for publication by Associate Editor Paolo di Lillo and Editor Clement Gosselin upon evaluation of the Associate Editor and Reviewers' comments. This work was partly supported by the Korea Institute of Science and Technology Institutional Programs under grant number 2E31593, and by ITECH R&D programs of MOTIE/KEIT under project Numbers 20014398 and 20014485. (†*Corresponding author: Jinoh Lee.*)

[1]Y. Lee is with Center for Intelligent and Interactive Robotics, Korea Institute of Science and Technology (KIST), 02792 Seoul, South Korea yisoo.lee@kist.re.kr

[2]N. Tsagarakis is with the Humanoids and Human Centered Mechatronics, Istituto Italiano di Tecnologia, 16163 Genoa, Italy nikos.tsagarakis@iit.it

[3]C. Ott and J. Lee are with the institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany {christian.ott, jinoh.lee}@dlr.de

Digital Object Identifier 10.1109/LRA.2022.3140425

The fault-tolerant control has been particularly interested in an actuation failure case, which is considered twofold: free-swinging joint and locked joint modes. The former is caused by faults of hardware or software in the robotic system inducing the loss of control torque [3]. Whereas, the latter is often resulted by hardware faults in the actuator (e.g., mechanical jamming in the gear train), or transformed from the former by physically locking the joint position using brake mechanisms [4], [5], because the free-swinging failure is treated as a bigger threat in safety [6].

To cope with the free-swinging failure, accordingly, considerable studies have been carried out in the 1990s. However, their solution and practical applications are limited due to the complexity of the problem regarding the dynamic motion of under-actuated systems. Whereas, unlike conventional industrial-robot-like designs dominated over the past three decades, a recent advance and trend in robotics demonstrates robot systems requesting better safety and versatility under free-swinging failure mode such as legged robots with light and powerful direct (or quasi-direct) drives without the brake, and torque-controlled manipulators oriented for human-robot interaction/collaboration tasks. Accordingly, this paper revisits the free-swinging failure problem and aims at enhancing fault-tolerant control capability to accomplish given tasks.

The main challenge for controlling the free-swinging failure joints comes from the highly nonlinear behavior of the multi-body system while the controller is forced to sustain an under-actuation problem. Furthermore, when the actuator failure happens, the robot controller is perturbed by significant dynamic effects such as uncompensated gravity and centrifugal torques at the faulted joints and, in turn, the dynamic effect creates disturbances against the targeted task execution. This complication of the coupled system [7], [8] instigates issues when controlling the end-effector in the Cartesian coordinate under the actuator failure as discussed in early studies [9]–[12]. There have been several approaches developed for a planar robot [13], [14] and a space manipulator [15], where the free-swinging failure problem is simplified or bypassed owing to their gravity-free design. In [16], a fault-tolerant index is proposed for parallel robots to avoid critical damage from free-swinging joints, and authors in [17] propose a fault control strategy for locomotion of a quadruped robot deploying the remaining non-faulted legs. However, these methods are not applicable to general robots such as articulated manipulators.

An interesting and effective approach to mitigate the aforementioned issue is the performance index based fault-tolerant

control which is to minimize expected failure effects and to prevent the robot from uncontrollable or other risky configurations such as singularity, self-collision, and joint limit violation; and kinematic redundancy is deployed to effectively create self-motion in the null-space of the main task. However, most of the failure indices have typically addressed the locked joint problem for workspace [18], singular value of the Jacobian matrix [19], condition number [20], and self-motion manifolds [21]. On the other hand, literature on the free-swinging failure index has been rarely found: as a pioneering work, the authors in [3] propose the failure-susceptibility (FS) which can measure the free-swinging effect caused by gravity force acting on the failure joints; and of more concern for the task-level performance, the Euclidean-space effect measure is developed in [22] based on the FS which can describe object position errors caused by free-swinging joints. However, since those approaches disregard other dynamic influences except gravity force, their applicability is limited to quasi-static motion control.

In this paper, main contributions are as follows: (i) we propose a performance measure of the fault-tolerance, named *generalized failure-susceptibility* (GFS) which can render not only the gravitational effect but also the complete dynamic effects stemmed from actuation-failed joints, acting as disturbance forces when the robot performs any operational-space tasks. It is based on the operational space formulation (OSF) methodology [23]–[25]; more specifically, the GFS is formulated by exploiting the result of the authors' previous study [25], where the operational space (OS) dynamics is mathematically analysed with consideration of the un-actuated joint and analytic dynamic equations are derived regarding the nonlinear dynamic effect in the OS caused by actuation failure; (ii) we then propose the online fault-tolerant control leveraging the optimal OS task performance, which minimizes the GFS via posture control in the null-space of the main task by exploiting the hierarchical OSF control framework; and (iii) finally, the effectiveness of the proposed GFS and the control approach are verified through real experiments with the torque-controlled 7 degrees-of-freedom (DoFs) robot arm under scenarios of the joint actuation failure. This paper also offers comparative analyses of both theoretic formulation and experiments to clarify differences from the previous approach and to reveal its limitations which have been often underestimated.

II. Preliminaries

A. Robot Dynamics

The rigid body dynamics equation of the robot with k joints is expressed in the joint space as follows:

$$\mathbf{A}\ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g} = \mathbf{\Gamma}, \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{k\times k}$ denotes the inertia matrix; $\mathbf{q} \in \mathbb{R}^k$ the joint angle vector; $\mathbf{b} \in \mathbb{R}^k$ the Coriolis/centrifugal force vector; $\mathbf{g} \in \mathbb{R}^k$ the gravity force vector; and $\mathbf{\Gamma} \in \mathbb{R}^k$ the joint torque vector.

The OSF describing the motion space $\mathbf{x}\in\mathbb{R}^n$ when $n\leq k$ is derived with the Jacobian matrix $\mathbf{J}\in\mathbb{R}^{n\times k}$ as follows [23]:

$$\mathbf{\Lambda}\ddot{\mathbf{x}} + \boldsymbol{\mu} + \mathbf{p} = \mathbf{F}, \tag{2}$$

$$\text{where } \begin{cases} \mathbf{\Lambda} = (\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T)^{-1}, \\ \boldsymbol{\mu} = \bar{\mathbf{J}}^T\mathbf{b} - \mathbf{\Lambda}\dot{\mathbf{J}}\dot{\mathbf{q}},\ \mathbf{p} = \bar{\mathbf{J}}^T\mathbf{g},\ \bar{\mathbf{J}}^T = \mathbf{\Lambda}\mathbf{J}\mathbf{A}^{-1}, \end{cases} \tag{3}$$

and $\mathbf{F} \in \mathbb{R}^n$ is the force vector in OS; $\mathbf{\Lambda} \in \mathbb{R}^{n\times n}$ the inertia matrix in OS; $\boldsymbol{\mu} \in \mathbb{R}^n$ the Coriolis/centrifugal force vector in OS; $\mathbf{p} \in \mathbb{R}^n$ the gravity force vector in OS; and $\bar{\mathbf{J}}^T$ the dynamically consistent inverse of $\mathbf{J}^T$.

B. The OSF for Under-actuated Robots

The robot undergoing free-swinging actuation failure can be regarded as the under-actuated system. For the under-actuated robot such with u un-actuated joints whose actuation torques are zero, the actuation torque vector $\mathbf{\Gamma}_a \in \mathbb{R}^{k-u}$ can be described as $\mathbf{\Gamma}_a = \mathbf{S}\mathbf{\Gamma}$ where $\mathbf{S} \in \mathbb{R}^{(k-u)\times k}$ is the selection matrix. The inverse relationship from the torque to the operational space force can then be expressed as follows:

$$\mathbf{F} = \bar{\mathbf{J}}^T\mathbf{\Gamma} = \bar{\mathbf{J}}^T\mathbf{S}^T\mathbf{\Gamma}_a. \tag{4}$$

From the above (4), one can obtain the relationship between $\mathbf{\Gamma}_a$ and $\mathbf{F}$ as follows [24]:

$$\mathbf{\Gamma}_a = \overline{\bar{\mathbf{J}}^T\mathbf{S}^T}\mathbf{F}, \tag{5}$$

where $\overline{\bar{\mathbf{J}}^T\mathbf{S}^T}$ is the dynamically consistent inverse of $\bar{\mathbf{J}}^T\mathbf{S}^T$. For the kinematically redundant robot with un-actuated joints, i.e., $n<(k-u)$, the dynamically consistent inverse is obtained by the weighted pseudo inverse as

$$\overline{\bar{\mathbf{J}}^T\mathbf{S}^T} = \mathbf{W}^{-1}(\bar{\mathbf{J}}^T\mathbf{S}^T)^T\{\bar{\mathbf{J}}^T\mathbf{S}^T\mathbf{W}^{-1}(\bar{\mathbf{J}}^T\mathbf{S}^T)^T\}^{-1}, \tag{6}$$

where the weighting matrix $\mathbf{W} = \mathbf{S}\mathbf{A}^{-1}\mathbf{S}^T \in \mathbb{R}^{(k-u)\times(k-u)}$ is full-rank and invertible.

Desired force $\mathbf{F}_d\in\mathbb{R}^n$ to generate a given reference acceleration vector in the OS $\ddot{\mathbf{x}}^*\in\mathbb{R}^n$ is obtained from (2) as

$$\mathbf{F}_d = \mathbf{\Lambda}\ddot{\mathbf{x}}^* + \boldsymbol{\mu} + \mathbf{p}. \tag{7}$$

By substituting (7) into (5), the control torque of the under-actuated robot $\mathbf{\Gamma}_a$ is expressed as follows:

$$\mathbf{\Gamma}_a = \overline{\bar{\mathbf{J}}^T\mathbf{S}^T}(\mathbf{\Lambda}\ddot{\mathbf{x}}^* + \mathbf{p} + \boldsymbol{\mu}). \tag{8}$$

If the robot has kinematic redundancy, the lower priority task can be controlled without affecting high priority task with the following hierarchical control formulation [24]:

$$\mathbf{\Gamma}_a = \overline{\bar{\mathbf{J}}^T\mathbf{S}^T}(\mathbf{\Lambda}\ddot{\mathbf{x}}^* + \boldsymbol{\mu} + \mathbf{p}) + \mathbf{N}_a^T\mathbf{\Gamma}_{a,0}, \tag{9}$$

where $\mathbf{N}_a^T \in \mathbb{R}^{(k-u)\times(k-u)}$ is the null-space projection matrix defined as $\mathbf{N}_a^T = \mathbf{I} - \overline{\bar{\mathbf{J}}^T\mathbf{S}^T}(\bar{\mathbf{J}}^T\mathbf{S}^T)$, while $\mathbf{\Gamma}_{a,0} \in \mathbb{R}^{(k-u)}$ is an arbitrary torque vector for the low priority task. When the gravity and Coriolis/centrifugal forces are compensated in the joint space, the actuation torque vector can be expressed as

$$\mathbf{\Gamma}_a = \overline{\bar{\mathbf{J}}^T\mathbf{S}^T}\mathbf{\Lambda}\ddot{\mathbf{x}}^* + \mathbf{N}_a^T\mathbf{\Gamma}_{a,0} + \bar{\mathbf{S}}^T(\mathbf{b} + \mathbf{g}), \tag{10}$$

where $\bar{\mathbf{S}}^T = (\mathbf{S}\mathbf{A}^{-1}\mathbf{S}^T)^{-1}\mathbf{S}\mathbf{A}^{-1}$ is the dynamically consistent inverse of $\mathbf{S}^T$. Finally, the desired control torque $\mathbf{\Gamma}$ for

the under-actuated robot can be determined by substituting (10) into $\mathbf{\Gamma}_a = \mathbf{S}\mathbf{\Gamma}$ as follows:

$$\mathbf{\Gamma} = \mathbf{S}^T\mathbf{\Gamma}_a = \mathbf{S}^T\{\overline{\bar{\mathbf{J}}^T\mathbf{S}^T}\mathbf{\Lambda}\ddot{\mathbf{x}}^* + \mathbf{N}_a^T\mathbf{\Gamma}_{a,0} + \bar{\mathbf{S}}^T(\mathbf{b}+\mathbf{g})\}. \tag{11}$$

III. Generalized Failure-Susceptibility and Fault-tolerant Control

A. Problem Definition

In this study, we deploy the OSF approach (11) to control the end-effector motion of general articulated robots under actuation failure scenarios, where feasible control torques can be found by employing the zero torque constraint for the faulted joints. However, the challenge is to guarantee the task performance given in OS, since the conventional OSF (11) itself cannot compensate the dynamic perturbation stemming from gravity, Coriolis/centrifugal force, and other internal forces acting on the failure joint [25].

The main objectives are thus (i) to formulate a performance index fully describing those dynamic effects on the given task, and (ii) to develop a task-space controller assuring the main task performance under the actuation failure. The controller deploys optimization of the index to effectively produce self-motion harmonizing the kinematic redundancy of the system and remaining actuation capability after the failure. In particular, it is of necessity that control designers can select the joint of interest in the performance index to protect vulnerable joints even prior to the failure (e.g., joints under high torque and speed request or severe operating condition). For further reliability and safety for the task execution, considering the fact that the completion of the assigned task is often physically limited by the robot configuration and degrees-of-redundancy, as well as under-actuation condition, the proposed index is designed to bestow precedence for certain components within the main task (e.g., a particular direction of motion).

B. Proposition of Generalized Failure-Susceptibility (GFS)

In the authors' previous study [25], the dynamic effect from the un-actuated joint to the OS task is analysed, which is derived by the robot dynamics (1) and the control torque equation (11) as the following symbolic form:

$$\ddot{\mathbf{x}}_f(\mathbf{S}) = \ddot{\mathbf{x}}_{f1} + \ddot{\mathbf{x}}_{f2} \tag{12}$$
$$\begin{cases} \ddot{\mathbf{x}}_{f1} = \mathbf{J}(\mathbf{I}-\mathbf{S}^T\mathbf{S})\bar{\mathbf{J}}(\ddot{\mathbf{x}}^* - \dot{\mathbf{J}}\dot{\mathbf{q}}), \\ \ddot{\mathbf{x}}_{f2} = -\mathbf{J}\mathbf{A}^{-1}(\mathbf{I}-\mathbf{S}^T\bar{\mathbf{S}}^T)(\mathbf{b}+\mathbf{g}), \end{cases}$$

where the former term $\ddot{\mathbf{x}}_{f1}$ denotes the OS acceleration component which cannot be realized due to the un-actuated joints when applying the reference command $\mathbf{x}^*$, while the latter term $\ddot{\mathbf{x}}_{f2}$ is the OS acceleration caused by gravity and Coriolis/centrifugal forces at the un-actuated joints which cannot be physically created (detailed derivation of (12) can be found in Appendix A.)

Preparing the joint actuation failure, the proposed GFS index $\mathcal{F}$ is designed with the corresponding $\ddot{\mathbf{x}}_f(\breve{\mathbf{S}})$ as follows:

$$\mathcal{F} \triangleq \ddot{\mathbf{x}}_f^T(\breve{\mathbf{S}})\mathbf{W}_{\mathcal{F}}\ddot{\mathbf{x}}_f(\breve{\mathbf{S}}), \tag{13}$$

where $\mathbf{W}_{\mathcal{F}} \in \mathbb{R}^{n\times n}$ is a positive semi-definite matrix for weighting, which is set for the OS acceleration components of the assigned task, and $\breve{\mathbf{S}} \in \mathbb{R}^{(k-\alpha)\times k}$ is a selection matrix of fault anticipation for α joints whose elements are chosen with 1 or 0 by the control designer.

It is noted that $\breve{\mathbf{S}}$ can be different from $\mathbf{S}$, since $\mathbf{S}$ is strictly set by the actual actuation state. For example, consider the case that one concerns the failure of the first and the last joints of a 7-DoFs robot ($\alpha = 2$) yet the first joint actuation is actually failed ($u = 1$). The GFS, $\mathcal{F}$, is then calculated with $\breve{\mathbf{S}} \in \mathbb{R}^{(7-2)\times 7}$ chosen as

$$\breve{\mathbf{S}} = \begin{bmatrix}\mathbf{0}^{5\times 1} & \mathbf{I}^{5\times 5} & \mathbf{0}^{5\times 1}\end{bmatrix},$$

while the control torque is calculated by (11) with the following $\mathbf{S} \in \mathbb{R}^{(7-1)\times 7}$ representing the actual joint torque failure:

$$\mathbf{S} = \begin{bmatrix}\mathbf{0}^{6\times 1} & \mathbf{I}^{6\times 6}\end{bmatrix}.$$

The weighting matrix $\mathbf{W}_{\mathcal{F}}$ allows to design which control element in operational space is more important. For example, in a 3 dimensional motion space $\mathbf{x} = [x\ y\ z]^T$, if the GFS has to express the failure of a task related to the z-axis as the most important, $\mathbf{W}_{\mathcal{F}}$ can be set as

$$\mathbf{W}_{\mathcal{F}} = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 100\end{bmatrix}.$$

When $\mathbf{W}_{\mathcal{F}}$ is minimized through fault-tolerant control (introduced in the next subsection), z-axis motion can be less affected by failure joints than other components. If $\mathbf{W}_{\mathcal{F}} = \mathbf{I}$ is set to treat all task components equally, all motion components will be affected by the nature of robot dynamics.

In addition, it is worth comparing the proposed GFS with the existing FS introduced in [3]— a torque-based fault-tolerant measure, given as follows:

$$f_g \triangleq \mathbf{g}^T\mathbf{W}_f\mathbf{g}, \tag{14}$$

where $\mathbf{W}_f \in \mathbb{R}^{k\times k}$ is a positive semi-definite weighting matrix which is generally set as a diagonal matrix with positive entries. As seen in the above equation, the FS represents the weighted magnitude of the gravity torque acting on the robot affected by the failed joints. For example, by setting diagonal entries of $\mathbf{W}_f$ which correspond to failure expected joints as 1 and the other diagonal entries as zero, the FS can indicate how large the gravitational force can cause the free-swinging failure effect.

The FS index is easy to implement and intuitive to use. It allows to set the weighting directly on the joints associated with failure, and has a low computational cost since the calculation of the index is based on a simple joint space formulation. However, the FS is feasible only when the robot is in a static state, i.e., $\dot{\mathbf{q}} = \ddot{\mathbf{q}} = \mathbf{0}$, since it only considers the gravity force $\mathbf{g}$. Moreover, unlike the proposed GFS provides a straightforward relation to OS task performance, the FS concerns joint-space dynamics. This notion is also found in the difference between the weighting matrix of $\mathbf{W}_{\mathcal{F}}$ and $\mathbf{W}_f$, where the former (proposed GFS) is designed in the OS while the latter (FS) in the joint space.

Interestingly, the FS (14) can also be associated with the OS task by setting a particular $\mathbf{W}_f$. From (12), one can take out the gravity force solely affecting the OS acceleration as $\mathbf{g}_f = -\mathbf{J}\mathbf{A}^{-1}(\mathbf{I} - \mathbf{S}^T\bar{\mathbf{S}}^T)\mathbf{g}$. Then, the FS concerning the OS task can be computed as $f_g = \mathbf{g}_f^T\mathbf{W}_f\mathbf{g}_f$, which yields the following relation to $\mathbf{W}_\mathcal{F}$ selected for the OS task:

$$\mathbf{W}_f = \{\mathbf{J}\mathbf{A}^{-1}(\mathbf{I} - \mathbf{S}^T\bar{\mathbf{S}}^T)\}^T\mathbf{W}_\mathcal{F}\{\mathbf{J}\mathbf{A}^{-1}(\mathbf{I} - \mathbf{S}^T\bar{\mathbf{S}}^T)\}. \tag{15}$$

Nevertheless, the FS is still confined to the gravity effect while the proposed GFS includes the dynamic effects at the OS caused by the joint velocity term ($\dot{\mathbf{J}}\dot{\mathbf{q}}$ and $\mathbf{b}$) and the reference acceleration term ($\ddot{\mathbf{x}}^*$). From this fact, it is noted that the proposed GFS can be treated as a generalized index not limited to the static situation.

C. Fault-tolerant Control

In this section, we propose an online fault-tolerant control strategy exploiting the GFS index, $\mathcal{F}$, which can be minimized with the self-motion control approach utilizing the kinematic redundancy of the robot while controlling the OS task as a high priority task. The OS control performance under the actuator failure can be improved since the minimization of the GFS reduces the negative impact on the OS.

Prioritized tasks can be controlled by the conventional hierarchical OSF to manage redundant robots before and after actuator failure as follows:

- *Before* free-swinging failure [23]:

$$\boldsymbol{\Gamma} = \mathbf{J}^T\boldsymbol{\Lambda}\ddot{\mathbf{x}}^* + \mathbf{N}^T\boldsymbol{\Gamma}_\mathcal{F} + \mathbf{b} + \mathbf{g}, \tag{16}$$

- *After* free-swinging failure [24]:

$$\boldsymbol{\Gamma} = \mathbf{S}^T\{\overline{\bar{\mathbf{J}}^T\mathbf{S}^T}\boldsymbol{\Lambda}\ddot{\mathbf{x}}^* + \mathbf{N}_a^T\mathbf{S}\boldsymbol{\Gamma}_\mathcal{F} + \bar{\mathbf{S}}^T(\mathbf{b} + \mathbf{g})\}, \tag{17}$$

where $\mathbf{N}^T \in \mathbb{R}^{k\times k}$ is the null-space projection matrix for the fully-actuated robot which is calculated as $\mathbf{I} - \mathbf{J}^T\bar{\mathbf{J}}^T$, and $\boldsymbol{\Gamma}_\mathcal{F} \in \mathbb{R}^k$ is the self-motion control torque. In (17), to consider under-actuation, the selection matrix is multiplied by $\boldsymbol{\Gamma}_\mathcal{F}$ to obtain the low priority task torque $\boldsymbol{\Gamma}_{a,0}$, i.e., $\boldsymbol{\Gamma}_{a,0} = \mathbf{S}\boldsymbol{\Gamma}_\mathcal{F}$.

Here, $\boldsymbol{\Gamma}_\mathcal{F}$ is determined by the gradient optimization approach [26]. By taking the gradient descent of $\mathcal{F}$ describing the posture behavior, the desired torque for each joint to produce optimized posture is then obtained as follows:

$$\boldsymbol{\Gamma}_\mathcal{F} = -k_\mathcal{F}\nabla\mathcal{F}, \quad \text{or} \quad \tau_j = -k_\mathcal{F}\frac{\partial\mathcal{F}}{\partial q_j}, \tag{18}$$

where τ_j is the j-th joint element of the torque $\boldsymbol{\Gamma}_\mathcal{F}$, $k_\mathcal{F} > 0$ denotes the step-size gain, and q_j is the joint angle of j-th joint. Note that the gradient, $\partial\mathcal{F}/\partial q_j$, is numerically computed in this paper. By incorporating the gradient optimization into the OSF-based hierarchical controller, $\mathcal{F}$ can be locally minimized during the main task execution such as end-effector motion control.

It is worth noting that the GFS may not completely converge to zero with the null space torque (18) which is mainly influenced by the following three reasons: first, since it is the torque command creating an acceleration, not velocity, the control action cannot immediately move the joints towards the desired direction. More importantly, when dynamic perturbation torque from un-actuated joints is greater than that of the torque (18), $\mathcal{F}$ can be increased. Thus, if the actuation failure occurs when the GFS is large, the performance of the proposed method is physically restricted; second, $\boldsymbol{\Gamma}_\mathcal{F}$ may not generate proper motion in a certain posture since the desired joint torque is projected to the null space of the task; and last, the gradient optimization approach is a method to find a local minimum solution. Nevertheless, in practice, $\mathcal{F}$ can be effectively minimized and we demonstrate several experimental results in the following section.

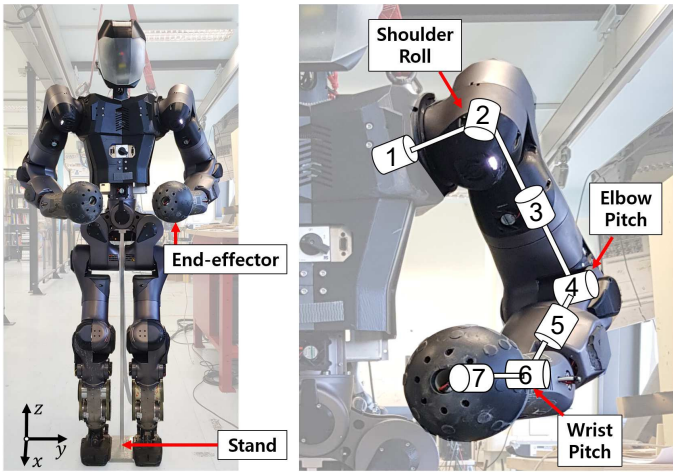


Fig. 1. The humanoid robot COMAN+ (left) and the picture of the schematic kinematic structure of the 7 DoFs left arm (right). The numbers at joints indicate the order of the joint. The direction of the global coordinate is depicted with arrows in the left bottom of the picture.

IV. Experimental Verification

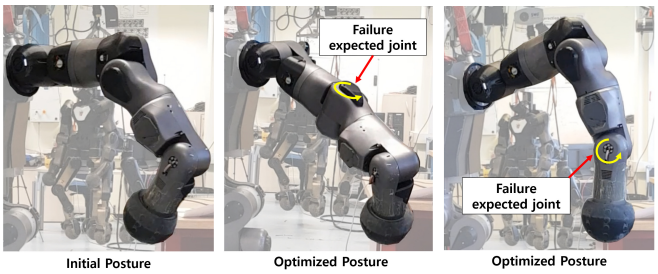
To verify the effectiveness of the proposed GFS and the fault-tolerant control approach, actual robot experiments with a robotic arm are conducted (Please also refer to the supplementary video.)

A. Settings

The left arm of the human-sized humanoid robot COMAN+ [27] shown in Fig. 1 is used for the experiments. The robot arm has 7 DoFs with torque-controlled actuators and a controller is implemented on a Linux-Xenomai based hard realtime operating system, called XBotCore [28], running with the sampling frequency of 0.5 kHz. During experiments, to attenuate any impacts from the other bodies of COMAN+ to the left arm, the pelvis as the base frame of the robot is firmly fixed on a stand, while the joints except those of the left arm are position-controlled with high gains to keep the initial posture.

The robot is controlled with the hierarchical control structure (16) and (17) as proposed. The 3 DoFs Cartesian-space position of the end-effector, i.e., the center of the ball-shaped hand shown in Fig. 1, is controlled as the highest priority task. To track the desired OS position $\mathbf{x}_d$, and velocity $\dot{\mathbf{x}}_d$, reference acceleration $\ddot{\mathbf{x}}^*$ for the end-effector control is calculated as

$$\ddot{\mathbf{x}}^* = k_p(\mathbf{x}_d - \mathbf{x}) + k_v(\dot{\mathbf{x}}_d - \dot{\mathbf{x}}) + \ddot{\mathbf{x}}_d, \tag{19}$$



(a) (b) (c)

Fig. 2. Results of the GFS optimization: (a) the initial posture of the robot arm; and the results of the optimized posture when the actuation failure is expected at (b) the elbow-pitch joint (q_4), and (c) the wrist-pitch joint (q_6).

TABLE I
OPTIMIZED GFS ($\mathcal{F}$) AND RESULTING CONTROL TORQUE (τ)

failure-anticipated joint	initial posture		**optimized posture**	
	$\mathcal{F}$	τ	$\mathcal{F}$	τ
elbow pitch (q_4)	30.1	4.98 Nm	**0.11**	0.61 Nm
wrist pitch (q_6)	46.5	-1.91 Nm	**0.20**	-0.23 Nm

where $k_p > 0$ and $k_d > 0$ are control gains. The torque for fault-tolerant control, $\mathbf{\Gamma}_{\mathcal{F}} \in \mathbb{R}^7$, is generated to minimize the proposed GFS with (18) or the existing measure FS with the gradient descent of (14) as a comparison. The OSF-based control equation ensures the null-space projection of the torque will not have any dynamic effects on the end-effector task with higher priority. The matrix $\mathbf{W}_{\mathcal{F}}$ is set as $\mathbf{I}$ to measure the fault effect on all the OS task components with equal weight.

B. Verification of the GFS index

To examine the efficacy of the proposed GFS, two example experiments are conducted in this subsection. In both examples, the robot is fully actuated with the suggested hierarchical control equation (16) andthe given OS task is to regulate the initial XYZ position of the end-effector. Then, we select 'failure anticipated' joints and investigate results when the GFS $\mathcal{F}$ is minimized.

The first example is for the case when the actuator failure is anticipated at the elbow-pitch joint (q_4), while the second example is for the one at the wrist-pitch joint (q_6). For each example, corresponding selection matrices for the GFS are composed as follows, respectively:

$$\breve{\mathbf{S}} = \begin{bmatrix} \mathbf{I}^{3\times3} & \mathbf{0}^{3\times1} & \mathbf{0}^{3\times3} \\ \mathbf{0}^{3\times3} & \mathbf{0}^{3\times1} & \mathbf{I}^{3\times3} \end{bmatrix}, \breve{\mathbf{S}} = \begin{bmatrix} \mathbf{I}^{5\times5} & \mathbf{0}^{5\times1} & \mathbf{0}^{5\times1} \\ \mathbf{0}^{1\times5} & 0 & 1 \end{bmatrix}.$$

Anticipating failure at joints, the robot optimally reconfigures the arm posture by self-motion minimizing the GFS measure. The results are shown in Fig. 2 and Table I. For both examples, the desired torque of the failure expected joint become approximately zero according to the posture optimization which reduces $\mathcal{F}$ to near zero. The desired torque and $\mathcal{F}$ are not perfectly converged to zero, as observed in Table I, which is mainly because the joint friction torques are not considered in the dynamics model used for experiments.

As the result of the GFS optimization, the robot posture converges to that with zero force acting at the failure anticipated joint. For example, one can intuitively notice from Fig. 2c that the center of mass of the forearm links after the failure expected joint is along with the direction of the gravity so that the potential disturbance force is minimized. Hence, free-swinging motion can be prevented even if the actuation failure occurs in the optimized posture since the failure expected joint requests small desired torque to complete the OS task.

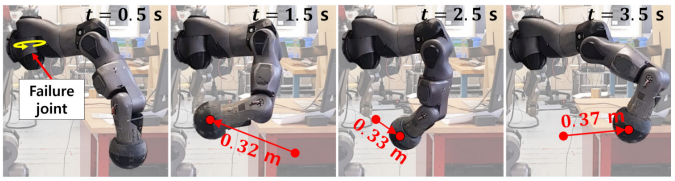


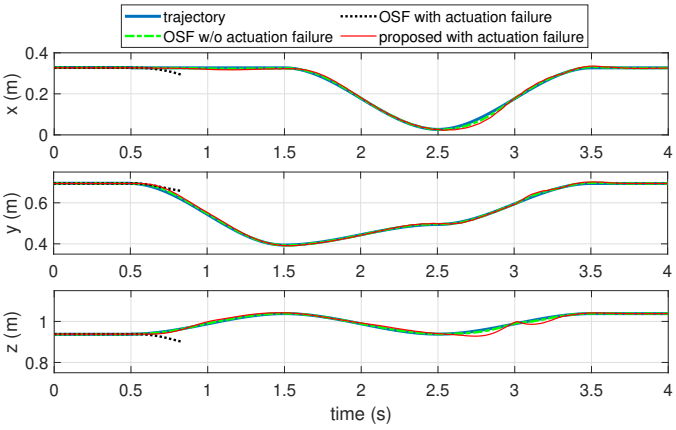
Fig. 3. Snapshots of the fault-tolerant control experiment. The actuation failure at the shoulder-roll joint (q_2) occurs at t=0.5s, while the hand is commanded to consecutively move 0.32m, 0.33m, and 0.37m for 3 seconds.

C. Fault-tolerant Control under Shoulder Actuation Failure

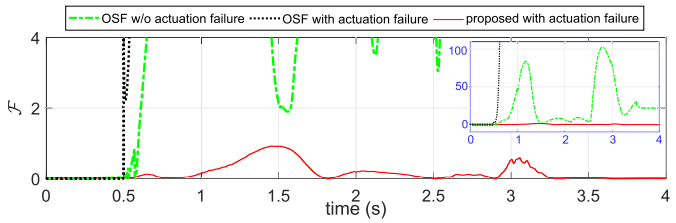
This subsection further investigates the fault-tolerant control performance via the proposed GFS by artificially emulating a sudden actuation failure and recovery situation. To mimic the free-swinging failure, the torque-controlled actuator at the corresponding joint is commanded by a zero reference torque. After recovery from the failure condition, the torque command computed from the task-space controller is normally sent to the actuator. This actually occurs in torque-controlled robots when a joint actuator fails to receive the torque command for a certain time period due to sporadic errors such as communication fault (software) or unstable cable connection (hardware); the actuator should then create zero torque for safety.

A fast and dynamic OS motion task is assigned at the end-effector as shown in Fig. 3: initially, the robot is controlled by (16) via the GFS optimization without actuation failures; and at the beginning of the motion trajectory starts (t=0.5s), the shoulder-roll joint (q_2) fails to generate the torque yet the manipulator continues to be controlled for tracking the OS task by (17). For three seconds (t=0.5-3.5s), the end-effector is directed to move the total of 1.02 m distance with high speed while changing the direction twice at t=1.5s and t=2.5s. From t=0.5-1.5 s, the end-effector position trajectory of the x-axis is maintained at the initial configuration, while that of y- and z-axes moves by -0.3 m and 0.1 m, respectively; from t=1.5-2.5 s, that of x-, y-, and z-axes moves by -0.3 m, 0.1 m, and -0.1 m, respectively; and from t=2.5-3.5 s, that of x-, y-, and z-axes moves by 0.3 m, 0.2 m, and 0.1 m, respectively. Note after t=3.5s, the shoulder roll joint is recovered from the failure and the controller is switched back to the full-actuation control (16).

Fig. 4a presents the trajectory of the OS task and the position tracking results. When the shoulder roll joint loses the torque, the end-effector successfully tracks the given OS task trajectory with the proposed fault-tolerant control approach which finds the optimal posture minimizing the GFS online. Throughout the operation, the GFS $\mathcal{F}$ is kept small enough as



(a)



(b)

Fig. 4. Plots of control results: (a) XYZ-position trajectory and response of the end-effector task, and (b) the measured response of the GFS, where the red line denotes the GFS value with the proposed method, while the green dashed line is the one with the OSF without actuation failure, black dotted line is the one with the OSF with actuation failure. The subplot in a blue box shows the same result with larger limit of y-axis.

seen in Fig. 4b, and accordingly free-swinging motion at the faulted joint is effectively suppressed.

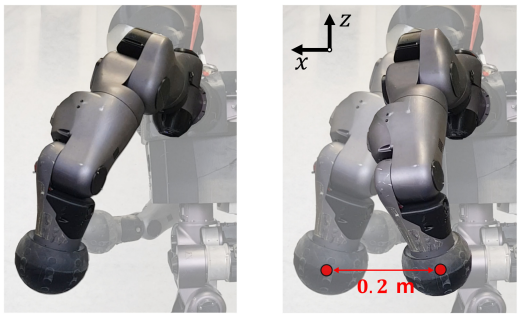
As a baseline, the GFS $\mathcal{F}$ is measured while performing the same task without optimizing $\mathcal{F}$. The task is executed by the conventional OSF when all the joints are operating normally and when actuation fails. As seen in Fig. 4b, the OSF fails to track a given trajectory shortly after the joint actuation fails due to the free-swinging motion of the faulted joints. The experiment is promptly halted at t=0.84 s to prevent a risky situation. Interestingly, one can observe that $\mathcal{F}$ suddenly increases after the actuator failure occurs although the initial $\mathcal{F}$ value is close to zero. This implies that fault tolerance can be implementable when $\mathcal{F}$ is continuously controlled to be minimized. Unless there is actuation failure, the trajectory tracking is well-achieved by the OSF, yet the GFS $\mathcal{F}$ value increases up to 102.8 (25.0 on average), which is a consistently high number. Thus, one can predict that the task execution can be failed without the proposed fault-tolerant control strategy.

V. Comparative Experiments

This section demonstrates the effectiveness of the proposed GFS index for fault-tolerant control with three comparative experiments under the actuation failure scenario. (Please also refer to the supplementary video.)

A. Setting for Comparisons

In the experiments, the robot starts to be controlled from the initial posture shown in Fig. 5a. For the first 3 seconds, the



(a) initial pose (b) motion tracking

Fig. 5. Snapshots for comparative experiments (left side view): (a) initial posture, and (b) generated motion during the trajectory tracking control. During the task execution, the shoulder roll joint is suddenly failed after 6 seconds yet recovered after 20 seconds.

end-effector is controlled to regulate its initial position while optimizing its posture to minimize $\mathcal{F}$ for the actuator failure at the shoulder roll joint (q_2) with (16), and the end-effector position is then controlled to repeatedly travel 20 cm in every 4 seconds along the x-axis with the sinusoidal trajectory shown in Figs. 5b and 6a. The actuation failure is artificially implemented as the shoulder roll actuator suddenly loses the torque at $t = 6$ s yet is recovered after $t = 20$ s.

The proposed method is compared with two other methods: the FS index by minimizing (14) and the null-space joint damping method. The null-space joint damping method is one of the most popular methods for stabilizing the self-motion of a redundant robot. It creates braking torques to decrease joint velocities with $\boldsymbol{\Gamma}_{a,0} = -k_{d,j}\mathbf{S}\dot{\mathbf{q}}$, where $k_{d,j} > 0$ is the damping gain. For the FS, the elements of $\mathbf{W}_f$ for the failure joint (q_2) is set to 1 as follows:

$$\mathbf{W}_f = \begin{bmatrix} 0 & 0 & \mathbf{0}^{1\times5} \\ 0 & 1 & \mathbf{0}^{1\times5} \\ \mathbf{0}^{5\times1} & \mathbf{0}^{5\times1} & \mathbf{0}^{5\times5} \end{bmatrix},$$

while the selection matrix for the GFS is set as

$$\breve{\mathbf{S}} = \begin{bmatrix} 1 & 0 & \mathbf{0}^{1\times5} \\ \mathbf{0}^{5\times1} & \mathbf{0}^{5\times1} & \mathbf{I}^{5\times5} \end{bmatrix},$$

and the task weighting is set as $\mathbf{W}_{\mathcal{F}}$=$\mathbf{I}$ for a fair comparison.

B. Results

As seen in the control results in Fig. 6, only the GFS-based method is able to complete the given trajectory tracking task after the failure. When the FS is minimized, the end-effector motion is well-controlled until t=10 s even after the actuation failure occurs at t=6 s. However, the failure joint starts to move from $t \approx 10$ s, i.e., free-swinging motion emerges, and the control error rapidly increases as the velocity of the joint increases as shown in Fig. 6b. For the case of the null-space joint damping method, although the operating joints are controlled to create braking torques in the null-space, it cannot restrain the free-swinging motion immediately after the actuator failure happens. Note when excessive free-swinging motion at the failure joint is observed, the robot system is promptly halted to prevent any risky situations such as self-collision.

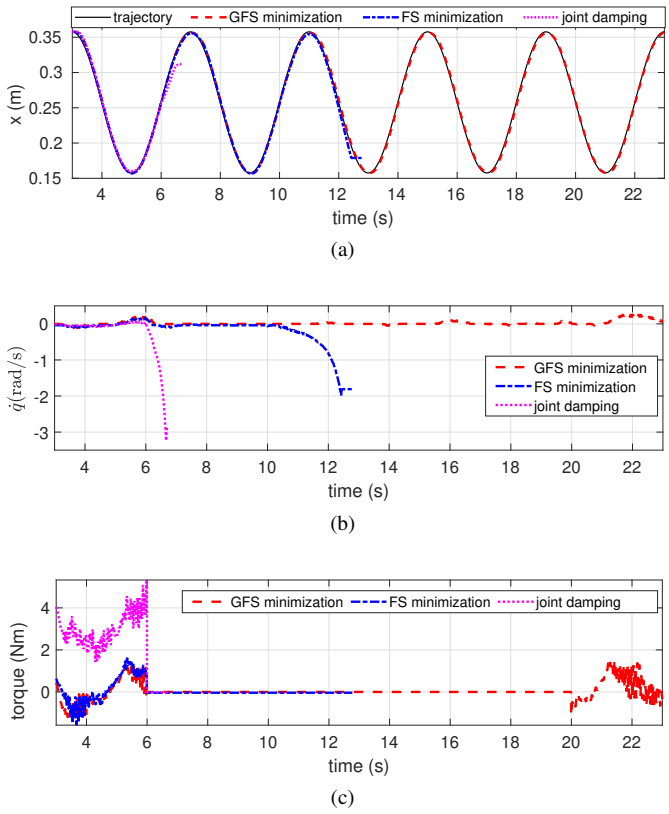


Fig. 6. Control results for comparative experiments: (a) trajectory and position of the end-effector, (b) joint velocity of the shoulder roll joint, and (c) reference joint torque of the shoulder roll joint.

The reference torques for the shoulder roll joint are shown in Fig. 6c. When the GFS or FS is minimized, the reference torque for the joint becomes close to zero unlike the case that the posture is not optimized. Thus, even if the actuation torque suddenly becomes zero due to the failure, i.e., loses the torque, the end-effector can be well-controlled without suffering from the impact of the actuator failure. During the actuation failure, t=6-20 s, since the under-actuation considered control torque (17) is applied, the reference torque is zero in all three experiments.

VI. Conclusion

This study presents a fault-tolerant control method along with the newly proposed performance index, the generalized failure-susceptibility (GFS), to reinforce task space performance under the free-swinging actuation failure. Since the GFS not only reflects the gravity torque but also expresses the full dynamic effects from the un-controllable joints, it can be exploited to assure better fault tolerance even with an agile motion task which increases potential risk when the failure occurs. The effectiveness of the proposed method is verified on the kinematically redundant and torque-controlled robot manipulator. Especially, the experimental results demonstrate that the use of the GFS is advantageous to successfully accommodate dynamic motion tasks, while compared conventional methods are significantly disturbed by free-swinging motion easily induced even with small acceleration of the task under the actuation failure.

One may notice that the proposed method encompasses interesting questions such as "how to detect the joint failure?", "how to estimate which joints are likely to fail?", or "what if actuation is partially degraded?" Since these are mainly associated with an issue of selecting $\breve{\mathbf{S}}$, our future work will focus on diagnosis and estimation methods to identify appropriate $\breve{\mathbf{S}}$. In addition, performance limitations originating from local optima of the gradient projection method is to be further investigated and anticipated to be overcome by employing global optimization method and advanced optimal control approach such as a model predictive control.

Appendix A

This section presents the derivation of (12) (please also refer to more details in [25].) The closed-loop response of the robot joints with the control torque (11) can be calculated by substituting $\boldsymbol{\Gamma}$ (11) into the robot dynamics (1) as follows:

$$\mathbf{A}\ddot{\mathbf{q}} = \mathbf{S}^T\overline{\bar{\mathbf{J}}^T\mathbf{S}^T}\boldsymbol{\Lambda}\ddot{\mathbf{x}}^* + \mathbf{S}^T\mathbf{N}_a^T\boldsymbol{\Gamma}_{a,0} - (\mathbf{I}-\mathbf{P}_a)(\mathbf{b}+\mathbf{g}), \tag{20}$$

where $\mathbf{P}_a = \mathbf{S}^T\bar{\mathbf{S}}^T$ is the projection matrix.

To explicitly find out the terms affecting the OS, the first two terms of the right-hand side of (20) is re-written as follows: the first term, $\mathbf{S}^T\overline{\bar{\mathbf{J}}^T\mathbf{S}^T}\boldsymbol{\Lambda}\ddot{\mathbf{x}}^*$, can be expressed by substituting (3) and $\ddot{\mathbf{x}}^* = \mathbf{J}\ddot{\mathbf{q}}^* + \dot{\mathbf{J}}\dot{\mathbf{q}}$ as

$$\begin{aligned}\mathbf{S}^T\overline{\bar{\mathbf{J}}^T\mathbf{S}^T}\boldsymbol{\Lambda}\ddot{\mathbf{x}}^* =& \mathbf{S}^T\overline{\bar{\mathbf{J}}^T\mathbf{S}^T}\bar{\mathbf{J}}^T\mathbf{A}\bar{\mathbf{J}}(\mathbf{J}\ddot{\mathbf{q}}^* + \dot{\mathbf{J}}\dot{\mathbf{q}})\\ =& \overline{\bar{\mathbf{J}}^T}\bar{\mathbf{J}}^T\mathbf{A}(\bar{\mathbf{J}}\mathbf{J}\ddot{\mathbf{q}}^* + \bar{\mathbf{J}}\dot{\mathbf{J}}\dot{\mathbf{q}})\\ =& \mathbf{P}_o\mathbf{A}(\mathbf{P}_t\ddot{\mathbf{q}}^* + \bar{\mathbf{J}}\dot{\mathbf{J}}\dot{\mathbf{q}}),\end{aligned} \tag{21}$$

where $\mathbf{P}_o = \overline{\bar{\mathbf{J}}^T}\bar{\mathbf{J}}^T$ and $\mathbf{P}_t = \bar{\mathbf{J}}\mathbf{J}$ are the projection matrices, and $\overline{\bar{\mathbf{J}}^T}$ is the dynamically consistent inverse of $\bar{\mathbf{J}}^T$; and the second term, $\mathbf{S}^T\mathbf{N}_a^T\boldsymbol{\Gamma}_{a,0}$, can be re-expressed as

$$\begin{aligned}\mathbf{S}^T\mathbf{N}_a^T\boldsymbol{\Gamma}_{a,0} =& \{\mathbf{S}^T - \mathbf{S}^T(\overline{\bar{\mathbf{J}}^T\mathbf{S}^T})\bar{\mathbf{J}}^T\mathbf{S}^T\}\boldsymbol{\Gamma}_{a,0}\\ =& (\mathbf{S}^T - \mathbf{P}_o\mathbf{S}^T)\boldsymbol{\Gamma}_{a,0}\\ =& (\mathbf{I}-\mathbf{P}_o)\mathbf{S}^T\boldsymbol{\Gamma}_{a,0}.\end{aligned} \tag{22}$$

Multiplying $\bar{\mathbf{J}}^T$ to (20) and substituting (21) and (22) into (20) yield the following OS response equation:

$$\begin{aligned}\boldsymbol{\Lambda}\ddot{\mathbf{x}} =& \bar{\mathbf{J}}^T\mathbf{P}_o\mathbf{A}(\mathbf{P}_t\ddot{\mathbf{q}}^* + \bar{\mathbf{J}}\dot{\mathbf{J}}\dot{\mathbf{q}}) - \bar{\mathbf{J}}^T(\mathbf{I}-\mathbf{P}_a)(\mathbf{b}+\mathbf{g})\\ =& \bar{\mathbf{J}}^T\mathbf{A}\bar{\mathbf{J}}\mathbf{J}\ddot{\mathbf{q}}^* + \bar{\mathbf{J}}^T\mathbf{A}\bar{\mathbf{J}}\dot{\mathbf{J}}\dot{\mathbf{q}} - \bar{\mathbf{J}}^T(\mathbf{I}-\mathbf{P}_a)(\mathbf{b}+\mathbf{g})\\ =& \boldsymbol{\Lambda}\{\mathbf{J}\ddot{\mathbf{q}}^* + \dot{\mathbf{J}}\dot{\mathbf{q}} - \mathbf{J}\mathbf{A}^{-1}(\mathbf{I}-\mathbf{P}_a)(\mathbf{b}+\mathbf{g})\}.\end{aligned} \tag{23}$$

Note that the lower priority task term multiplied by (22), i.e., $\bar{\mathbf{J}}^T(\mathbf{I}-\mathbf{P}_o)\mathbf{S}^T\boldsymbol{\Gamma}_{a,0}$, is nullified by $\bar{\mathbf{J}}^T(\mathbf{I}-\mathbf{P}_o) = \mathbf{0}$.

To extract the terms caused by the un-actuated joints, we can simply decompose the joint acceleration reference $\ddot{\mathbf{q}}^*$ into two vectors as

$$\ddot{\mathbf{q}}^* = \underbrace{\mathbf{S}^T\mathbf{S}\ddot{\mathbf{q}}^*}_{=\ddot{\mathbf{q}}_a^*} + \underbrace{(\mathbf{I}-\mathbf{S}^T\mathbf{S})\ddot{\mathbf{q}}^*}_{=\ddot{\mathbf{q}}_u^*}, \tag{24}$$

where $\ddot{\mathbf{q}}_a^*$ denotes the reference acceleration vector for the actuated joints and $\ddot{\mathbf{q}}_u^*$ denotes that of un-actuated joints.

Then, the expected OS acceleration response (23) can be re-described as follows:

$$\ddot{\mathbf{x}} = (\mathbf{J}\ddot{\mathbf{q}}_a^* + \underbrace{\mathbf{J}\ddot{\mathbf{q}}_u^*}_{\ddot{\mathbf{x}}_{f1}} + \dot{\mathbf{J}}\dot{\mathbf{q}}) \underbrace{-\mathbf{J}\mathbf{A}^{-1}(\mathbf{I}-\mathbf{P}_a)(\mathbf{b}+\mathbf{g})}_{\ddot{\mathbf{x}}_{f2}}. \quad (25)$$

Here, $\ddot{\mathbf{x}}_{f1}$ and $\ddot{\mathbf{x}}_{f2}$ present the OS acceleration terms caused by un-actuated joints and is further expressed as the form shown in (12):

$$\begin{cases} \ddot{\mathbf{x}}_{f1} = \mathbf{J}(\mathbf{I}-\mathbf{S}^T\mathbf{S})\ddot{\mathbf{q}}^* = \mathbf{J}(\mathbf{I}-\mathbf{S}^T\mathbf{S})\bar{\mathbf{J}}(\ddot{\mathbf{x}}^* - \dot{\mathbf{J}}\dot{\mathbf{q}}), \\ \ddot{\mathbf{x}}_{f2} = -\mathbf{J}\mathbf{A}^{-1}(\mathbf{I}-\mathbf{S}^T\bar{\mathbf{S}}^T)(\mathbf{b}+\mathbf{g}). \end{cases}$$

REFERENCES

[1] M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki, and J. Schröder, *Diagnosis and fault-tolerant control*. Springer, 2006, vol. 2.

[2] J. Guiochet, M. Machin, and H. Waeselynck, "Safety-critical advanced robots: A survey," *Robot. and Auton. Syst.*, vol. 94, pp. 43–52, 2017.

[3] J. D. English and A. A. Maciejewski, "Fault tolerance for kinematically redundant manipulators: Anticipating free-swinging joint failures," *IEEE Trans. Robot. Autom.*, vol. 14, no. 4, pp. 566–575, 1998.

[4] M. Plooij, G. Mathijssen, P. Cherelle, D. Lefeber, and B. Vanderborght, "Lock your robot: A review of locking devices in robotics," *IEEE Robot. and Automat. Mag.*, vol. 22, no. 1, pp. 106–117, 2015.

[5] B. Xie and A. A. Maciejewski, "Maximizing the probability of task completion for redundant robots experiencing locked joint failures," *IEEE Trans. Robot.*, 2021, (*in press*).

[6] H. Chang, P. Huang, M. Wang, and Z. Lu, "Locked-joint failure identification for free-floating space robots," in *2014 IEEE Int. Conf. Inf. Autom. (ICIA)*. IEEE, 2014, pp. 170–175.

[7] A. Jain and G. Rodriguez, "An analysis of the kinematics and dynamics of underactuated manipulators," *IEEE Trans. Robot. Autom.*, vol. 9, no. 4, pp. 411–422, 1993.

[8] M. Bergerman, C. Lee, and Y. Xu, "A dynamic coupling index for underactuated manipulators," *J. Robot. Syst.*, vol. 12, no. 10, pp. 693–707, 1995.

[9] A. De Luca, S. Iannitti, R. Mattone, and G. Oriolo, "Control problems in underactuated manipulators," in *Proc. 2001 IEEE/ASME Int. Conf. Adv. Intell. Mechatronics. (AIM)*, vol. 2. IEEE, 2001, pp. 855–861.

[10] G. Oriolo and Y. Nakamura, "Control of mechanical systems with second-order nonholonomic constraints: Underactuated manipulators," in *Proc. 1991 IEEE Conf. Decis., Control (CDC)*. IEEE, 1991, pp. 2398–2403.

[11] R. M. Bianchini and G. Stefani, "Controllability along a trajectory: A variational approach," *SIAM J. Control Optim.*, vol. 31, no. 4, pp. 900–927, 1993.

[12] M. W. Spong, "Underactuated mechanical systems," in *Control problems in robotics and automation*. Springer, 1998, pp. 135–150.

[19] K. N. Groom, A. A. Maciejewski, and V. Balakrishnan, "Real-time failure-tolerant control of kinematically redundant manipulators," *IEEE Trans. Robot. Autom.*, vol. 15, no. 6, pp. 1109–1115, 1999.

[13] Y. Wang, X. Lai, L. Chen, H. Ding, and M. Wu, "A quick control strategy based on hybrid intelligent optimization algorithm for planar n-link underactuated manipulators," *Information Sciences*, vol. 420, pp. 148–158, 2017.

[14] A. Freddi, S. Longhi, A. Monteriù, D. Ortenzi, and D. P. Pagnotta, "Fault tolerant control scheme for robotic manipulators affected by torque faults," *IFAC-PapersOnLine*, vol. 51, no. 24, pp. 886–893, 2018.

[15] J. Qingxuan, Y. Bonan, C. Gang, and F. Yingzhuo, "Adaptive fuzzy terminal sliding mode control for the free-floating space manipulator with free-swinging joint failure," *Chinese J. Aeronautics*, vol. 34, no. 9, pp. 178–198, 2021.

[16] M. Isaksson, K. Marlow, A. Maciejewski, and A. Eriksson, "Novel fault-tolerance indices for redundantly actuated parallel robots," *J. Mechanical Design*, vol. 139, no. 4, p. 042301, 2017.

[17] M. M. Gor, P. M. Pathak, A. K. Samantaray, J. M. Yang, and S. Kwak, "Fault-tolerant control of a compliant legged quadruped robot for free swinging failure," *Proc.Institution Mech. Engineers, Part I: J. Syst. Control Eng.*, vol. 232, no. 2, pp. 161–177, 2018.

[18] C. L. Lewis and A. A. Maciejewski, "Fault tolerant operation of kinematically redundant manipulators for locked joint failures," *IEEE Trans. Robot. Autom.*, vol. 13, no. 4, pp. 622–629, 1997.

[20] H. Abdi and S. Nahavandi, "Minimum reconfiguration for fault tolerant manipulators," in *Proc. 2010 ASME Int. Design Eng. Tech. Conf., Computers, Inf. Eng. Conf. (IDETC-CIE)*. ASME Digital Collection, 2010, pp. 1345–1350.

[21] A. A. Almarkhi and A. A. Maciejewski, "Maximizing the size of self-motion manifolds to improve robot fault tolerance," *IEEE Robot. Automat. Lett.*, vol. 4, no. 3, pp. 2653–2660, 2019.

[22] J. D. English and A. A. Maciejewski, "Measuring and reducing the euclidean-space effects of robotic joint failures," *IEEE Robot. and Automat. Mag.*, vol. 16, no. 1, pp. 20–28, 2000.

[23] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE J. Robot. Autom.*, vol. 3, no. 1, pp. 43–53, 1987.

[24] J. Park, "The relationship between controlled joint torque and end-effector force in underactuated robotic systems," *Robotica*, vol. 29, no. 4, pp. 581–584, 2011.

[25] Y. Lee, N. Tsagarakis, and J. Lee, "Study on operational space control of a redundant robot with un-actuated joints: experiments under actuation failure scenarios," *Nonlinear Dynamics*, vol. 105, pp. 331–344, 2021.

[26] A. Liegeois *et al.*, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *IEEE Trans. Syst., Man, Cybern. Cybern.*, vol. 7, no. 12, pp. 868–871, 1977.

[27] Y. Lee, S. Kim, J. Park, N. Tsagarakis, and J. Lee, "A whole-body control framework based on the operational space formulation under inequality constraints via task-oriented optimization," *IEEE Access*, vol. 9, pp. 39 813–39 826, 2021.

[28] L. Muratore, A. Laurenzi, E. M. Hoffman, and N. Tsagarakis, "The XBot real-time software framework for robotics: From the developer to the user perspective," *IEEE Robot. & Automat. Mag.*, vol. 27, no. 3, pp. 133–143, 2020.