Different-Level Simultaneous Minimization Scheme for Fault Tolerance of Redundant Manipulator Aided with Discrete-Time Recurrent Neural Network

Long Jin[1], Bolin Liao[2]*, Mei Liu[1], Lin Xiao[2], Dongsheng Guo[3] and Xiaogang Yan[4]

[1] School of Information Science and Engineering, Lanzhou University, Lanzhou, China, [2] College of Information Science and Engineering, Jishou University, Jishou, China, [3] School of Information Science and Engineering, Huaqiao University, Xiamen, China, [4] Department of Computer Science, University of Otago, Dunedin, New Zealand

By incorporating the physical constraints in joint space, a different-level simultaneous minimization scheme, which takes both the robot kinematics and robot dynamics into account, is presented and investigated for fault-tolerant motion planning of redundant manipulator in this paper. The scheme is reformulated as a quadratic program (QP) with equality and bound constraints, which is then solved by a discrete-time recurrent neural network. Simulative verifications based on a six-link planar redundant robot manipulator substantiate the efficacy and accuracy of the presented acceleration fault-tolerant scheme, the resultant QP and the corresponding discrete-time recurrent neural network.

Keywords: redundant manipulator, different level, fault tolerance, physical constraint, discrete-time recurrent neural network

1. INTRODUCTION

In recent decades, robotics has drawn more and more attention in scientific areas and engineering applications. Many researches have been focused on this topic, and various kinds of robots have thus been developed and investigated (Roberts and Maciejewski, 1996; Sun et al., 2009; Zhang and Zhang, 2012, 2013; Li et al., 2014; Jin and Zhang, 2015; Jin et al., 2016; Jin and Li, 2017; Jin et al., 2017a,b; Zhang et al., 2017a,b). As a typical kind of robot, redundant manipulators have played a more and more important role in numerous fields of engineering applications (Roberts and Maciejewski, 1996; Zhang and Zhang, 2012, 2013; Jin and Zhang, 2015; Liao and Liu, 2015; Jin et al., 2016, 2017a,b; Jin and Li, 2017; Zhang et al., 2017a,b). For redundant manipulators, they can accomplish subtasks easily and dexterously and optimization of various performance criteria, since they possess more degrees of freedom (DOF) than the minimum amount required to accomplish a given end-effector primary task. Recent progresses have shown advantages of using adaptive neural networks for the control of nonlinear systems (Tang et al., 2014, 2017). For example, an adaptive control scheme was provided in Tang et al. (2014) for robot manipulator systems with unknown functions and dead-zone input by using adaptive neural networks, of which the parameters of the dead zone are assumed to be unknown but bounded. One of the fundamental issues in operating such a redundant manipulator is the inverse-kinematics problem (or termed, redundancy-resolution problem). Specifically, the joint trajectories need to be generated online

based on the provided desired Cartesian trajectories of the end-effector. That is, if the manipulator control scheme is fault tolerant, the end-effector can fulfill the required task even if its joint fails. As one essential and challenging issue in inverse kinematics, it is important to tolerate joint failure online during the task execution. For example, in the remote applications such as space or sea exploration, repairing broken actuators is costly or even impossible and the probability of their failure is increased due to the hostile operating environment. To say the least, it may induce the damage to people and/or manipulator if a manipulator without a fault-tolerant scheme being equipped suddenly encounters a joint-lock situation during the execution. Thus, the fault-tolerant ability is an important or even indispensable design criterion for robotic systems.

More recently, the fault tolerance has attracted significant interest in the societies of academia and industry because of the increased demands in practical applications for safety and productivity as well as operating efficiency. Numerous researches on the topic of fault tolerance have thus been presented and investigated (Roberts and Maciejewski, 1996; Izumikawa et al., 2005; Siqueira and Terra, 2009; Li and Zhang, 2012). Authors in Siqueira and Terra (2009) described the fault occurrence for a three-link manipulator based on a Markovian jump model, which took into account all possible fault sequences in a three-link manipulator robot, and defined guidelines to control an n-link manipulator robot with several faults. Izumikawa et al. (2005) designed a controller of a fault-tolerant system with a signal observer for a strain gauge sensor fault. By switching from the controller for normal mode to the controller for unnormal mode, the stability and the control performance of such a system were maintained. Generally speaking, the existing methods for fault tolerance can be categorized as the passive type and the active type (Zhang and Jiang, 2008). The former one fixes and designs the corresponding controllers to be robust against the presumed faults, which needs neither fault detection nor controller reconfiguration, with limited capabilities for fault tolerance. The latter one reacts to the failures of the system actively by reconfiguring control actions to maintain the stability and acceptable performance of the entire system (Bustan et al., 2014).

A fault-tolerant scheme was presented by adding a matrix-vector form equality constraint for the faulty joint, which took the limits of joint angle and joint velocity into consideration (Li and Zhang, 2012). However, this fault-tolerant scheme can not handle the joint-acceleration limits. More seriously, this scheme may introduce the undesirable discontinuity phenomenon in the velocity solution because it was investigated at the joint-velocity level. Thus, it is worthy to investigate the scheme for fault-tolerance of redundant robot manipulators at the joint-acceleration level, which can effectively remedy the discontinuity phenomenon at the joint-velocity level and incorporates robot dynamic. However, up to now there is almost no study on the fault tolerance of redundant robot manipulators on the combination of different level. In other words, the study on the fault tolerance of redundant robot manipulators at the different level is still rare despite its appealingness.

In this paper, we make progress along this direction by presenting a different-level simultaneous minimization scheme, which takes both the robot kinematics and robot dynamics into account. The scheme is then reformulated as a quadratic program (QP) subject to equality and bound constraints. In order to solve such a QP problem, a discrete-time recurrent neural network is developed and applied to online solution of the QP problem. Simulative results based on a six-link planar robot manipulator further illustrate the efficacy and superiority of the proposed fault-tolerant scheme for fault tolerance of redundant robot manipulators.

2. PRELIMINARIES AND SCHEME FORMULATION

To lay a basis for further discussion, the relationship between the end-effector velocity $\dot{r} \in R^m$ and the joint velocity $\dot{q} \in R^n$ for redundant robot manipulators can be given hereinafter directly (Zhang and Zhang, 2012)

$$J(q)\dot{q} = \dot{r}, \qquad (1)$$

where $J(q) \in R^{m \times n}$ is the Jacobian matrix of the end-effector with q being the joint-angle vector. By differentiating Equation (1) with respect to time t, the relationship between the end-effector acceleration $\ddot{r}$ and the joint acceleration $\ddot{q}$ is obtained as follows (Zhang and Zhang, 2012):

$$J(q)\ddot{q} = \ddot{r}_b = \ddot{r} - \dot{J}(q)\dot{q}, \qquad (2)$$

where $\dot{J}(q)$ is the time derivative of $J(q)$. Note that, since the manipulator system is redundant (i.e., $m < n$), Equations (1) and (2) are under-determined, and generally admit an infinite number of feasible solutions in terms of inverse kinematics. This implies the ability to accommodate more functional constraints such as fault tolerance. For example, once a joint is stuck, other joints can take over its workload and move the end-effector to its goal via a fault-tolerant scheme. It is worth mentioning here that the fault-tolerant scheme investigated in Li and Zhang (2012) was based on Equation (1) (i.e., at the joint-velocity level), while the fault-tolerant scheme presented in the ensuing subsections is based on Equation (2) (i.e., at the joint-acceleration level).

For the online solution of Equation (2), the following QP-oriented formulation can be used (Zhang and Zhang, 2012):

$$\text{minimize} \quad \ddot{q}^T \Lambda \ddot{q}/2 + c^T \ddot{q}, \qquad (3)$$

$$\text{subject to} \quad J(q)\ddot{q} = \ddot{r}_b, \qquad (4)$$

where coefficients $\Lambda \in R^{n \times n}$ and $c \in R^n$ are defined accordingly for a specific redundancy-resolution scheme. In addition, superscript T denotes the transpose of a matrix or vector.

However, without appropriate remedied measures, when a manipulator suffers a joint failure, its performance would be significantly affected, and even worse, the manipulator fails to complete the prescribed path task. In safety-critical systems, the consequences of a minor fault in a system component can

be catastrophic. Therefore, the demands on reliability, safety and fault tolerance are generally high. It is necessary to take fault tolerance into consideration in the above inverse-kinematic scheme (i.e., Equations 3 and 4) in order to improve the reliability and availability while tracking a desirable path. Inspired by Roberts and Maciejewski (1996), at a time instant, if there are n_f joints being locked (e.g., the ith, $\cdots$, and jth joints with $i, j \in [1, 2, \cdots, n]$ and $i \neq j$), then we directly remove the corresponding joint-acceleration variables in the scheme. For example, if a failed joint i is locked, the corresponding relationship between the end-effector acceleration and the joint acceleration is obtained as

$$[^iJ(q)][^i\ddot{q}] = \ddot{r}_b = \ddot{r} - [^i\dot{J}(q)][^i\dot{q}], \qquad (5)$$

where $^iJ(q) = [j_1, \cdots, j_{i-1}, j_{i+1}, \cdots, j_n]$ and $^i\dot{J}(q) = [\dot{j}_1, \cdots, \dot{j}_{i-1}, \dot{j}_{i+1}, \cdots, \dot{j}_n]$ with j_h and $\dot{j}_h$ denoting the hth column of $J(q)$ and $\dot{J}(q)$, respectively. In addition, $^i\dot{q} = [\dot{q}_1, \cdots, \dot{q}_{i-1}, \dot{q}_{i+1}, \cdots, \dot{q}_n]$ and $^i\ddot{q} = [\ddot{q}_1, \cdots, \ddot{q}_{i-1}, \ddot{q}_{i+1}, \cdots, \ddot{q}_n]$. The reduced Equation (5) then determines the kinematic properties of the degraded system.

By incorporating the joint physical constraints as well as the robot dynamic presented in Appendix (Supplementary Material), the different-level simultaneous minimization scheme for fault tolerance of robot manipulators is written as

$$\text{minimize} \quad \alpha(\ddot{\theta}^T W \ddot{\theta}/2 + p^T \ddot{\theta}) + \beta \tau^T \tau/2, \qquad (6)$$
$$\text{subject to} \quad J(\theta)\ddot{\theta} = \ddot{r}_a, \qquad (7)$$
$$\tau = H\ddot{\theta} + c_\tau(\dot{\theta}, \theta) + g_\tau(\theta), \qquad (8)$$
$$\theta^- \leqslant \theta \leqslant \theta^+, \qquad (9)$$
$$\dot{\theta}^- \leqslant \dot{\theta} \leqslant \dot{\theta}^+, \qquad (10)$$
$$\ddot{\theta}^- \leqslant \ddot{\theta} \leqslant \ddot{\theta}^+, \qquad (11)$$
$$\tau^- \leqslant \tau \leqslant \tau^+, \qquad (12)$$

where $\alpha \in (0, 1)$ and $\beta \in (0, 1)$ are the weighting factors with $\alpha + \beta = 1$ numerically; $\theta, \dot{\theta}, \ddot{\theta}$ and τ denote the n_r dimensional joint-angle, joint-velocity, joint-acceleration and joint-torque vectors, respectively; $W = I \in R^{n_r \times n_r}$, and $J(\theta) \in R^{m \times n_r}$, $p \in R^{n_r}$; $b = \ddot{r}_a + K_v(\dot{r}_d - J(\theta)\dot{\theta}) + K_p(r_d - f(\theta)) \in R^m$; $\ddot{r}_a = \ddot{r} - \dot{J}(\theta)\dot{\theta}$ with $\dot{J}(\theta) \in R^{m \times n_r}$ and $n_r = n - n_f$ [n_f denotes the number of faulty joint(s)]. In addition, H denotes the $n_r \times n_r$ dimensional inertia matrix; c_τ denotes the n_r dimensional Coriolis/centrifugal force vector and g_τ denotes the n_r dimensional gravitational force vector. Besides, $\tau^\pm = \pm 140 \times 1_\nu$ N·m. For simplicity and for example, α is set to be 0.995 (i.e., $\beta = 0.005$) in the ensuing simulations.

Remark 1: Fault detection is a fundamental, specialized and relatively independent part for fault tolerance, for which many methods have been proposed. These methods can be classified into two categories: model-based methods and data-based methods (Yüksel and Sezgin, 2010). Model-based methods obtain the deviations signals between the model and the real system named as residuals to detect faults. Data-based methods are based on only processing input and output signals of the system to detect faults. In this paper, for focusing on the superior performance of the fault-tolerant scheme in faulty situation, it

can be simply assumed that the fault detection/diagnosis system can always detect and diagnose an unexpected joint fault and immediately give the feedback to the fault-tolerant system. Once the fault-tolerant system receives such a feedback, it activates the reconfiguration mechanism and removes the corresponding joint-acceleration variables in the scheme.

Remark 2: Note that the model disturbance and computational round-off errors always exist in practical application. In order to improve the accuracy of the scheme, the feedback control needs to be incorporated into the forward kinematics equation. One effective approach is to add feedbacks of Cartesian position and velocity error, i.e., instead of using $J(\theta)\ddot{\theta} = \ddot{r}_a$, Equation (7) can be replaced with

$$J(\theta)\ddot{\theta} = \ddot{r}_a + K_v(\dot{r}_d - J(\theta)\dot{\theta}) + K_p(r_d - f(\theta)),$$

where K_p and K_v are positive-definite symmetric $m \times m$ gain matrices for position-error and velocity-error feedbacks, respectively. In the ensuing simulations and experiments, the diagonal elements of the gain matrices K_p and K_v are set as 10 and the off-diagonal elements are set as 0 for simplicity and clarity.

With the aid of conversion techniques given in Cheng et al. (1994), Cheng et al. (1995), and Park et al. (1998), the new bound constraint can thus be written as $\eta^- \leqslant \ddot{\theta} \leqslant \eta^+$, where the ith elements of η^- and η^+ are defined respectively as

$$\eta_i^- = \max\{\gamma_p(\theta_i^- + \vartheta_i - \theta_i), \gamma_v(\dot{\theta}_i^- - \dot{\theta}_i), \ddot{\theta}_i^-\},$$
$$\eta_i^+ = \min\{\gamma_p(\theta_i^+ - \vartheta_i - \theta_i), \gamma_v(\dot{\theta}_i^+ - \dot{\theta}_i), \ddot{\theta}_i^+\},$$

where $\vartheta > 0$ is a small constant vector used to scale the safety region. Besides, coefficients $\gamma_p > 0$ and $\gamma_v > 0$ determine the deceleration magnitude.

Based on the above analysis, the proposed scheme for physically-constrained redundant robot manipulators can be reformulated into the following standard QP in terms of $\ddot{\theta}$:

$$\text{minimize} \quad \alpha(\ddot{\theta}^T W \ddot{\theta}/2 + p^T \ddot{\theta}) + \beta \tau^T \tau/2 \qquad (13)$$
$$\text{subject to} \quad A\ddot{\theta} = b, \qquad (14)$$
$$\eta^- \leqslant \ddot{\theta} \leqslant \eta^+, \qquad (15)$$

where $W = I \in R^{n_r \times n_r}$, $A = J(\theta) \in R^{m \times n_r}$, $b = \ddot{r}_a + K_v(\dot{r}_d - J(\theta)\dot{\theta}) + K_p(r_d - f(\theta)) \in R^m$, $\tau = H\ddot{\theta} + c_\tau(\dot{\theta}, \theta) + g_\tau(\theta)$, and $p = (\mu + \nu)\dot{\theta} + \mu\nu(\theta - \theta(0)) \in R^{n_r}$ with $\mu > 0$ and $\nu > 0$. In addition, $\ddot{\theta}$ and $\eta^\pm$ are defined the same as before.

Neural networks have been recognized as a powerful tool for real-time processing and successfully applied widely in various control systems (Wang et al., 2015, 2016). In particular, we use the following discrete-time recurrent neural network for solving online the QP problem (Xiao and Zhang, 2013; Zhang and Zhang, 2013).

$$u^{k+1} = u^k - \frac{\|e(u^k)\|_2^2}{\|(M^T + I)e(u^k)\|_2^2}(M^T + I)e(u^k), \qquad (16)$$

where $\| \cdot \|_2$ denotes the two-norm of a matrix; the decision variable vector u and its upper and lower bounds $u^\pm \in R^N$ (with

$N = n_r + m$) are defined respectively as

$$u = \begin{bmatrix} \ddot{\theta} \\ y \end{bmatrix}, \quad u^+ = \begin{bmatrix} \eta^+ \\ \varpi 1_v \end{bmatrix}, \quad u^- = \begin{bmatrix} \eta^- \\ -\varpi 1_v \end{bmatrix},$$

with $y \in R^m$ being the auxiliary decision vector (i.e., dual decision vector) defined corresponding to equality constraint (Equation 14), $1_v = [1, \cdots, 1]^T$ denoting an appropriately dimensioned vector composed of ones, constant $\varpi = 10^{10} \in R$ being defined to replace $+\infty$ numerically, and the augmented matrix M and vector z being defined respectively as

$$M = \begin{bmatrix} \alpha W + \beta H & -A^T \\ A & 0 \end{bmatrix} \in R^{N \times N}, \quad z = \begin{bmatrix} p + c_\tau \\ -b \end{bmatrix} \in R^N.$$

Besides, $P_\Omega(\cdot) : R^N \to \Omega$ is a piecewise-linear projection operator defined from space R^N onto set Ω, and the ith element of $P_\Omega(u)$ is defined as

$$[P_\Omega(u)]_i = \begin{cases} u_i^-, & \text{if } u_i \leqslant u_i^-, \\ u_i, & \text{if } u_i^- < u_i < u_i^+, \quad \forall i \in \{1, 2, \cdots, N\}. \\ u_i^+, & \text{if } u_i \geqslant u_i^+, \end{cases}$$
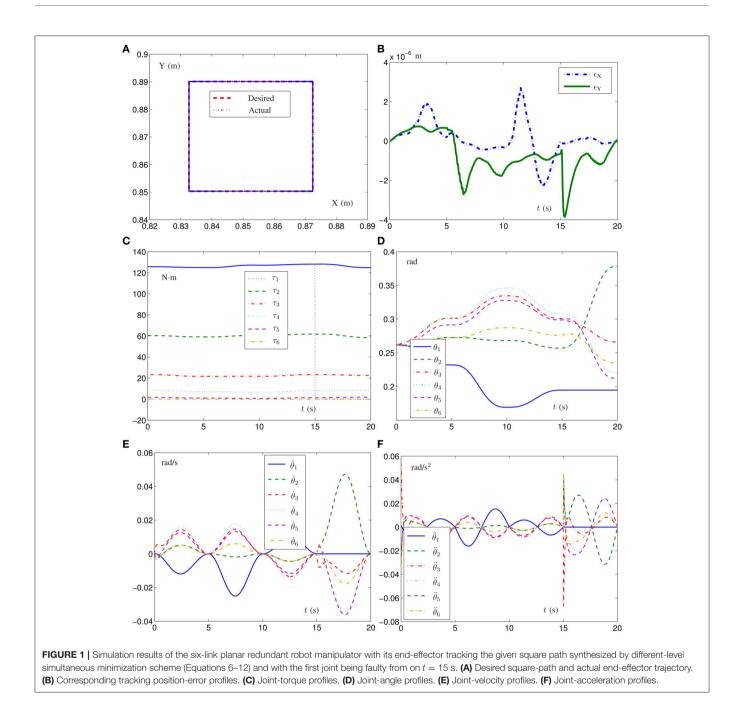
3. SIMULATIVE RESULTS

In this paper, a six-link planar manipulator with motor-driven push-rod type joints is presented as a simulative platform to illustrate the effectiveness of the scheme. The hardware system of the six-link planar manipulator, which has six joints, is presented in Zhang and Zhang (2013). The physical parameters of the six-link planar manipulator are shown in **Table 1**, of which θ_i^+ and θ_i^- denote respectively the upper and lower limits of the ith joint-angle vector and l_i denotes the length of the ith link, $i = 1, 2, \cdots, 6$. Besides, in the simulations and the ensuing experiment, the final error tolerance of $\|e(u^k)\|$ is set as 10^{-5} for the discrete-time QP solver Equation (16) with the sampling gap being 0.01 s. The end-effector of the six-link planar redundant robot manipulator is expected to track a square-path with side-length being 0.039 m. In addition, the duration of the path-tracking task is 20 s, $\vartheta = 0.1$ rad, joint-velocity limits $\dot{\theta}^\pm = \pm 1.5 \times 1_v$ rad/s, joint-acceleration limits $\ddot{\theta}^\pm = \pm 2 \times 1_v$ rad/s² and $\mu = \nu = 4$.

For comparison and for illustrating the efficacy of the different-level simultaneous minimization scheme (Equations 6–12) in the faulty situation, the simulation results

TABLE 1 | Physical limits of the six-link robot manipulator.

i	θ_i^- (rad)	θ_i^+ (rad)	l_i (m)	a_i (m)	b_i (m)
1	−1.536	1.431	0.301	–	–
2	0.052	0.816	0.290	0.250	0.080
3	0.035	0.621	0.230	0.250	0.080
4	0.052	0.599	0.225	0.190	0.080
5	0.035	0.599	0.214	0.185	0.080
6	0.009	0.445	0.103	0.174	0.080

synthesized by scheme (Equations 6–12) with the first joint being faulty from on $t = 15$ s are shown in **Figure 1**. As observed from **Figure 1A**, the end-effector motion trajectory is close enough to the desired square path (i.e., with the robot dynamics taken into account, the tracking task is also completed), which substantiates the effectiveness of the different-level simultaneous minimization scheme (Equations 6–12) in the faulty situation. In addition, the tracking position error with the maximal position error being less than 4×10^{-6} m shown in **Figure 1B** further shows the efficacy and accuracy of such a different-level simultaneous minimization scheme. Besides, in **Figure 1C**, for the first joint torque (i.e., τ_1 denoted by the blue lines), the solid lines and dashed lines respectively denote the joint-torque profiles corresponding to the no-fault situation and fault-tolerant situation with the first joint being faulty from on $t = 15$ s. As observed from **Figure 1C**, after $t = 15$ s, the value of the first joint torque becomes zero. With the first joint being faulty from on $t = 15$ s, the values of the joint velocities and joint accelerations are zero, which implies the efficacy of the different-level simultaneous minimization scheme (Equations 6–12) in the faulty situation. In summary, the above simulation results substantiate the efficacy and accuracy of the the different-level simultaneous minimization scheme (Equations 6–12), which takes both the robot kinematics and robot dynamics into account.

Remark 3: Note that, for a fault-tolerant task, it can be classified into the following two cases. (i) The equality constraint is always satisfied; e.g., once some joints are simultaneously faulty, the remainder joints can take over the workload and move the end-effector to its goal via a fault-tolerant scheme. (ii) The equality constraint can not be always satisfied; e.g., with some joints being faulty, the equality constraint does not hold at some time instants, and thus the path-tracking task can not be fulfilled in this situation. Theoretically speaking, the equality constraint should be satisfied all the time. However, strictly speaking, the equality constraint can not be satisfied exactly even for the first case. Specifically, the tracking position-error profiles synthesized by the different-level simultaneous minimization scheme (Equations 6–12) are numerically near zero but nonzero (i.e., 10^{-6}). That is because the simulation and computation are performed on a finite-arithmetic finite-memory digital computer. Then, the tracking position error may be inevitable between the desired path and actual trajectory, which is used to be the input of the feedback to track the task in a more accurate manner. To show clearly the second case (i.e., the equality constraint is not always satisfied), the corresponding motion trajectories and tracking errors with the first five joints being locked from on $t = 15$ s are visualized in **Figure 2**. Specifically, as seen from **Figure 2**, with the first five joints being faulty from on $t = 15$ s, the values of the corresponding joint velocities and joint accelerations are zero and the corresponding joint angles remain the same as $\theta(t = 15)$. To distinguish those two types of position error (i.e., the usual computational error, and the failure error due to the lack of feasible solution) as well as to keep the robotic system more reliable, a criterion can be added to the scheme. For example, the emergency brake of the system can be activated for the position error larger than 0.01 m with an increasing trend. As shown in **Figure 2D**, with the red dotted line denoting the
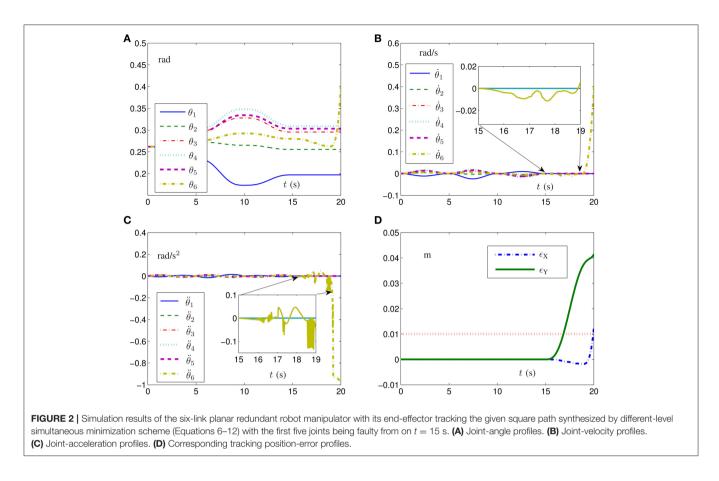
FIGURE 1 | Simulation results of the six-link planar redundant robot manipulator with its end-effector tracking the given square path synthesized by different-level simultaneous minimization scheme (Equations 6–12) and with the first joint being faulty from on $t = 15$ s. **(A)** Desired square-path and actual end-effector trajectory. **(B)** Corresponding tracking position-error profiles. **(C)** Joint-torque profiles. **(D)** Joint-angle profiles. **(E)** Joint-velocity profiles. **(F)** Joint-acceleration profiles.

criterion, the robotic system can be stopped at time instant $t \approx 16$ s to prevent the potential damage(s) to nearby people and/or robot arms.

It is worth pointing out that, although the investigations are based on the joint-lock situation, the efficacy of the proposed different-level simultaneous minimization scheme (Equations 6–12) as well as the resultant QP is not limited to the joint-lock situation. The joint-lock situation is just a representative of lots of joint faulty situations, such as the failure of one joint actuator. In addition, being the representative, the joint-lock situation extensively exists in types of joints (e.g., rotational joints and translational joints). For example,

the joints may be locked with a greater probability when the robot works with the sludge. Thus, the proposed different-level simultaneous minimization scheme (Equations 6–12) is generally applicable, and the feasibility and efficacy of such a proposed scheme are not limited by the specific robot structure and failure mode.

4. CONCLUSIONS

In this paper, by incorporating the physical constraints in joint space, a different-level simultaneous minimization scheme, which takes both the robot kinematics and robot dynamics

FIGURE 2 | Simulation results of the six-link planar redundant robot manipulator with its end-effector tracking the given square path synthesized by different-level simultaneous minimization scheme (Equations 6–12) with the first five joints being faulty from on $t = 15$ s. **(A)** Joint-angle profiles. **(B)** Joint-velocity profiles. **(C)** Joint-acceleration profiles. **(D)** Corresponding tracking position-error profiles.

into account, has been presented and investigated for fault-tolerant motion planning of redundant manipulator in this paper. Then, the scheme has been reformulated as a quadratic program (QP) with equality and bound constraints, which has been solved by a discrete-time recurrent neural network. Simulative verifications based on a six-link planar redundant robot manipulator have substantiated the efficacy and accuracy of the presented acceleration fault-tolerant scheme, the resultant QP and the corresponding discrete-time recurrent neural network.

AUTHOR CONTRIBUTIONS

LJ and BL presented the scheme. BL and ML designed experiments. LX and DG carried out experiments. XY analyzed experimental results. LJ and BL wrote the manuscript.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: http://journal.frontiersin.org/article/10.3389/fnbot. 2017.00050/full#supplementary-material

REFERENCES

Bustan, D., Sani, H., and Pariz, N. (2014). Adaptive fault-tolerant spacecraft attitude control design with transient response control. *IEEE/ASME Trans. Mechatr.* 19, 1404–1411. doi: 10.1109/TMECH.2013.2288314

Cheng, F. T., Chen, T. H., and Sun, Y. Y. (1994). Resolving manipulator redundancy under inequality constraints. *IEEE Trans. Robot. Autom.* 10, 65–71. doi: 10.1109/70.285587

Cheng, F. T., Sheu, R. J., and Chen, T. H. (1995). The improved compact QP method for resolving manipulator redundancy. *IEEE Trans. Syst. Man Cybern.* 25, 1521–1530. doi: 10.1109/21.467718

Craig, J. J. (2008). *Introduction to Robotics: Mechanics and Control.* New Jersey, NJ: Pearson Education.

Izumikawa, Y., Yubai, K., and Hirai, J. (2005). Fault-tolerant control system of flexible arm for sensor fault by using reaction force observer. *IEEE/ASME Trans. Mechatr.* 10, 391–396. doi: 10.1109/TMECH.2005.852442

Jin, L., and Li, S. (2017). Distributed task allocation of multiple robots: a control perspective. *IEEE Trans. Syst. Man Cybern. Syst.* doi: 10.1109/TSMC.2016.2627579

Jin, L., Li, S., Hung, L., and Luo, X. (2017a). Manipulability optimization of redundant manipulators using dynamic neural networks. *IEEE Trans. Ind. Electron.* 64, 4710–4720. doi: 10.1109/TIE.2017.2674624

Jin, L., Li, S., Xiao, L., Lu, R., and Liao, B. (2017b). Cooperative motion generation in a distributed network of redundant robot manipulators with noises. *IEEE Trans. Syst. Man Cybern. Syst*. doi: 10.1109/TSMC.2017.2693400

Jin, L., and Zhang, Y. (2015). G2-type SRMPC scheme for synchronous manipulation of two redundant robot arms. *IEEE Trans. Cybern.* 45, 153–164. doi: 10.1109/TCYB.2014.2321390

Jin, L., Zhang, Y., Li, S., and Zhang, Y. (2016). Modified ZNN for time-varying quadratic programming with inherent tolerance to noises and its application to kinematic redundancy resolution of robot manipulators. *IEEE Trans. Ind. Electron.* 63, 6978–6988. doi: 10.1109/TIE.2016.2590379

Li, K., and Zhang, Y. (2012). Fault-tolerant motion planning and control of redundant manipulator. *Control Eng. Pract.* 20, 282–292. doi: 10.1016/j.conengprac.2011.11.004

Li, S., Kong, R., and Guo, Y. (2014). Cooperative distributed source seeking by multiple robots: algorithms and experiments. *IEEE/ASME Trans. Mechatr.* 19, 1810–1820. doi: 10.1109/TMECH.2013.2295036

Liao, B., and Liu, W. (2015). Pseudoinverse-type bi-criteria minimization scheme for redundancy resolution of robot manipulators. *Robotica* 33, 2100–2113. doi: 10.1017/S0263574714001349

Park, K. C., Chang, P. H., and Kim, S. H. (1998). The enhanced compact QP method for redundant manipulators using practical inequality constraints. *Proc. IEEE Int. Conf. Robot. Autom.* 1998, 107–114.

Roberts, R., and Maciejewski, A. (1996). A local measure of fault tolerance for kinematically redundant manipulators. *IEEE Trans. Robot. Automat.* 12, 543–552. doi: 10.1109/70.508437

Saha, S. K. (2008). *Introduction to Robotics.* Noida: Tata McGraw-Hill Education.

Siqueira, A., and Terra, M. H. (2009). A fault-tolerant manipulator robot based on H_2, H_∞, and mixed H_2/H_∞ Markovian controls. *IEEE/ASME Trans. Mechatr.* 14, 257–263. doi: 10.1109/TMECH.2008.2009442

Sun, L., Meng, M., Li, S., Liang, H., and Mei, T. (2009). A novel CPG with proprioception and its application on the locomotion control of quadruped robot. *Int. J. Inf. Acquisition* 6, 33–46. doi: 10.1142/S0219878909001771

Tang, H., Liu, P., and Liu, S. (2017). Adaptive neural synchronization control for bilateral teleoperation systems with time delay and backlash-like hysteresis. *IEEE Trans. Cybern.* 1–9. doi: 10.1109/TCYB.2016.2644656

Tang, L., Liu, Y., and Tong, S. (2014). Adaptive neural control using reinforcement learning for a class of robot manipulator. *Neural Comput. Appl.* 25, 135–141. doi: 10.1007/s00521-013-1455-2

Wang, H., Liu, K., Liu, X., Chen B., and Lin, C. (2015). Neural-based adaptive output-feedback control for a class of nonstrict-feedback stochastic nonlinear systems. *IEEE Trans. Cybern.* 45, 1977–1987. doi: 10.1109/TCYB.2014.2363073

Wang, H., Liu, X., and Liu, K. (2016). Robust adaptive neural tracking control for a class of stochastic nonlinear interconnected systems. *IEEE Trans. Neural Netw. Learning Syst.* 27, 510–523. doi: 10.1109/TNNLS.2015.2412035

Xiao, L., and Zhang, Y. (2013). Acceleration-level repetitive motion planning and its experimental verification on a six-link planar robot manipulator. *IEEE Trans. Control Syst. Technol.* 21, 906–914. doi: 10.1109/TCST.2012.2190142

Yüksel, T., and Sezgin, A. (2010). Two fault detection and isolation schemes for robot manipulators using soft computing techniques. *Appl. Soft Comput.* 10, 125–134. doi: 10.1016/j.asoc.2009.06.011

Zhang, Y., and Jiang, J. (2008). Bibliographical review on reconfigurable fault-tolerant control systems. *Annu. Rev. Control* 32, 229–252. doi: 10.1016/j.arcontrol.2008.03.008

Zhang, Z., and Zhang, Y. (2012). Acceleration-level cyclic-motion generation of constrained redundant robots tracking different paths. *IEEE Trans. Syst. Man Cybern. B Cybern.* 42, 1257–1269. doi: 10.1109/TSMCB.2012.2189003

Zhang, Z., and Zhang, Y. (2013). Variable joint-velocity limits of redundant robot manipulators handled by quadratic programming. *IEEE/ASME Trans. Mechatr.* 18, 674–686. doi: 10.1109/TMECH.2011.2181977

Zhang, Y., and Zhang, Z. (2013). *Repetitive Motion Planning and Control of Redundant Robot Manipulators.* New York, NY: Springer-Verlag. doi: 10.1007/978-3-642-37518-7

Zhang, Z., Lin, Y., Li, S., Li, Y., and Luo, Y. (2017b). Tri-criteria optimization-coordination-motion of dual redundant robot manipulators for complex path planning. *IEEE Trans. Control Syst. Technol.* 27, 178–1186. doi: 10.1109/TCST.2017.2709276

Zhang, Z., Zheng, L., Yu, J., Li, Y., and Yu, Z. (2017a). Three recurrent neural networks and three numerical methods for solving repetitive motion planning scheme of redundant robot manipulators. *IEEE/ASME Trans. Mechatr.* 22, 1423–1434. doi: 10.1109/TMECH.2017.2683561