



OPEN ACCESS

EDITED BY

Long Jin,
Lanzhou University, China

REVIEWED BY

Boyu Ma,
Harbin Institute of Technology, China
Haoen Huang,
Huazhong University of Science and
Technology, China

*CORRESPONDENCE

Haitao Fu
✉ fht@jlau.edu.cn

RECEIVED 20 June 2024

ACCEPTED 22 July 2024

PUBLISHED 19 August 2024

CITATION

Li J, Su J, Yu W, Mao X, Liu Z and Fu H (2024)
Recurrent neural network for trajectory
tracking control of manipulator with unknown
mass matrix. *Front. Neurobot.* 18:1451924.
doi: 10.3389/fnbot.2024.1451924

COPYRIGHT

© 2024 Li, Su, Yu, Mao, Liu and Fu. This is an
open-access article distributed under the
terms of the [Creative Commons Attribution
License \(CC BY\)](#). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that the
original publication in this journal is cited, in
accordance with accepted academic practice.
No use, distribution or reproduction is
permitted which does not comply with these
terms.

Recurrent neural network for trajectory tracking control of manipulator with unknown mass matrix

Jian Li, Junming Su, Weilin Yu, Xuping Mao, Zipeng Liu and
Haitao Fu*

College of Information Technology, Jilin Agricultural University, Changchun, China

Real-world robotic operations often face uncertainties that can impede accurate control of manipulators. This study proposes a recurrent neural network (RNN) combining kinematic and dynamic models to address this issue. Assuming an unknown mass matrix, the proposed method enables effective trajectory tracking for manipulators. In detail, a kinematic controller is designed to determine the desired joint acceleration for a given task with error feedback. Subsequently, integrated with the kinematics controller, the RNN is proposed to combine the robot's dynamic model and a mass matrix estimator. This integration allows the manipulator system to handle uncertainties and synchronously achieve trajectory tracking effectively. Theoretical analysis demonstrates the learning and control capabilities of the RNN. Simulative experiments conducted on a Franka Emika Panda manipulator, and comparisons validate the effectiveness and superiority of the proposed method.

KEYWORDS

recurrent neural network (RNN), trajectory tracking, manipulator control, dynamic model, unknown mass matrix

1 Introduction

With the rapid development of modern robot research and development technology, manipulators have permeated various aspects of human life, such as space explorations (Ma et al., 2023a) and smart factories (Abate et al., 2022). Its fundamental functionality lies in trajectory tracking, where specific tasks are accomplished by executing predefined end-effector trajectories (Jin et al., 2024b). This involves the control of robot kinematics and dynamics (Liao et al., 2022; Lian et al., 2024; Sun et al., 2024). To exert control over the robot, desired joint attributes should be obtained according to the task trajectory and converted into the corresponding joint torques (Müller et al., 2023). Numerous algorithms, such as pseudoinverse methods (Guo et al., 2018; Sun et al., 2023a) and model predictive control method (Jin et al., 2023), have been designed to achieve precise control of the manipulator. However, these algorithms rely on accurate robot models and struggle to control the robot effectively when its parameters change. In practical applications, it is common for robot model parameters to vary, especially when robots are modified to perform different tasks in diverse application scenarios (Xiao et al., 2022; Xie and Jin, 2024). Reliable model-free control methods need to be designed to enable effective control of robots after the parameter changes.

In recent decades, there are many emerging algorithms to address the control issues of manipulators (Liao et al., 2024; Yan et al., 2024), which are considered from the velocity level (Zhang et al., 2019; Sun et al., 2022), acceleration level (Wen and Xie, 2024), or torque level (Hua et al., 2023). For instance, to eliminate the joint-angle drift and prevent excessive joint velocity, a velocity-level bi-criteria optimization scheme is provided for coordinated path tracking of manipulators, focusing on the velocity aspect (Xiao et al., 2017). Additionally, a data-driven acceleration-level scheme is introduced to address control continuity and stability issues for manipulator (Wen and Xie, 2024). However, most of these studies focus solely on kinematics, neglecting dynamic factors (Tang and Zhang, 2022). Robot kinematics and dynamics are two fundamental domains within the field of robotics. Robot kinematics focuses on the study of the motion capabilities of robots in space, encompassing aspects such as joint angles, positions, velocities, and accelerations, without considering the effects of forces (Xie et al., 2023). In addition, robot dynamics is concerned with the impact of forces and torques on the motion of the manipulator, including the interactions between the robot and its environment. In the control of robotic manipulators, considering dynamic factors can help precisely predict the actual motion trajectory of the manipulator under various load and motion conditions, thereby improving the overall motion accuracy (Sun et al., 2023b; Xiao et al., 2023). It can also compensate for the oscillation and coupling effects in joint motion, making the movement of the manipulator smooth and stable. Furthermore, the dynamics-based studies aid in selecting the optimal drive scheme, reducing energy consumption, and enhancing energy utilization efficiency. However, manipulators frequently encounter issues with dynamic uncertainties due to the diversity of robotic grippers and uncertainties in load (Bruder et al., 2021; Liu et al., 2024b). Specifically, surgical manipulators may be equipped with different end-effectors to meet various task requirements, implying changes in dynamic parameters (Liu et al., 2024b). Moreover, in robotic-grasping tasks, unknown loads also lead to variations in robot dynamic parameters (Bruder et al., 2021). Dynamic uncertainties significantly impede the accurate control of manipulators, highlighting its research significance.

Recurrent neural networks (RNNs) have emerged as effective robot control algorithms in recent years (Liao et al., 2023; Ma et al., 2023b; Jin et al., 2024a). RNN is utilized to establish a scheme for addressing the coordination problem for multirobot systems (Cao et al., 2023; Liu et al., 2024a). In addition, RNN can mitigate uncertainties in the robot systems by enabling online learning of robot parameters (Xie et al., 2022). However, further research is needed to explore the integration of synchronous dynamic parameter learning with the kinematic model to achieve accurate trajectory tracking (Tang et al., 2024). To this aim, this study assumes the presence of deviations in the robot dynamic model and proposes an RNN for the model-free control of manipulators. Specifically, relevant control algorithms are designed at both the kinematic and dynamic levels, and an estimator of the mass matrix is proposed to compensate for the uncertainty of the dynamic model. Further verifications are carried out on a Franka Emika Panda manipulator to perform a trajectory-tracking task, taking into account dynamic uncertainties. In addition, compared with the

existing methods, the superiorities of the proposed RNN lie in the following two aspects:

- Compared with kinematics-based methods (Guo et al., 2018; Jin et al., 2023, 2024b), the proposed RNN bridges the robot kinematics and robot dynamics models through joint acceleration signals, considering the motion feature and the dynamic behavior of manipulators.
- Compared with dynamics-based methods (Shojaei et al., 2021; Zong and Emami, 2021), the proposed RNN addresses the dynamic uncertainty problem by estimating the mass matrix online and realizes synchronous trajectory tracking.

Through the introduction of the above basic content, the specific research of this study is organized as follows. Section 2 explains the kinematic relationship between the joint angle of the manipulator and the end-effector. In Section 3, a corresponding RNN is designed. Subsequently, the learning and control ability of the proposed RNN are analyzed theoretically in Section 4. Finally, simulations and comparisons are carried out in Section 5.

2 Kinematic controller

The forward kinematics of the manipulator describes the mapping relationship between the joint angle and the end-effector position, described as $f(\mathbf{q}) = \mathbf{r}$, where $\mathbf{q} \in \mathbb{R}^a$ is the joint angle, $\mathbf{r} \in \mathbb{R}^b$ denotes the position of the end-effector, and $f(\cdot)$ stands for the non-linear mapping. Furthermore, the time derivative of the forward kinematics is derived as

$$\mathbf{L}\dot{\mathbf{q}} = \dot{\mathbf{r}}, \quad (1)$$

where $\mathbf{L} = \partial f(\mathbf{q})/\partial \mathbf{q} \in \mathbb{R}^{b \times a}$ is the Jacobian matrix, $\dot{\mathbf{q}}$ denotes the joint velocity, and $\dot{\mathbf{r}}$ is the velocity of the end-effector. Concerning the joint acceleration level, taking the time derivative of Equation 1 leads to

$$\dot{\mathbf{L}}\dot{\mathbf{q}} + \mathbf{L}\ddot{\mathbf{q}} = \ddot{\mathbf{r}}, \quad (2)$$

where $\dot{\mathbf{L}}$ is the time derivative of \mathbf{L} , $\ddot{\mathbf{q}}$ represents the joint acceleration, and $\ddot{\mathbf{r}}$ denotes the acceleration of the end-effector. Building upon Equation 2, the desired joint acceleration can be obtained by the following kinematic controller:

$$\ddot{\mathbf{q}} = \mathbf{L}^\dagger(\ddot{\mathbf{r}}_d - \dot{\mathbf{L}}\dot{\mathbf{q}} - \mathbf{v}), \quad (3)$$

where $\mathbf{v} = \beta(\dot{\mathbf{r}} - \dot{\mathbf{r}}_d) + \zeta(\mathbf{r} - \mathbf{r}_d)$ denotes the error feedback term; \mathbf{r}_d , $\dot{\mathbf{r}}_d$, and $\ddot{\mathbf{r}}_d$ are the desired position, velocity, and acceleration of the trajectory tracking task; superscript \dagger denotes the pseudoinverse operation of a matrix with $\mathbf{L}^\dagger = \mathbf{L}^T(\mathbf{L}\mathbf{L}^T)^{-1}$; and $\beta > 0$ and $\alpha > 0$ are convergence coefficients. On the one hand, kinematic controller (Equation 3) utilizes the minimization function of the pseudoinverse operation to obtain the desired joint acceleration (Wen and Xie, 2024). On the other hand, it takes the desired trajectory tracking task as input and incorporates feedback of the tracking error, leading to improved trajectory tracking performance. In addition, kinematic controller

(Equation 3) can apply traditional methods to avoid the singularity issues, such as the damped least squares method (Xie et al., 2024). Specifically, a damped term can be added in the computation of the pseudoinverse. The specific calculation formula is $L^T(LL^T + \mu I)^{-1}$, where $\mu > 0$ denotes a tiny parameter and I denotes the identity matrix. By doing so, the infinite values caused by zero eigenvalues in the pseudoinverse operation can be avoided.

3 Recurrent neural network design

Robot dynamics refers to the mathematical description of the relationship between joint torques, dynamic parameters, and joint motions in a robotic system. Specifically, the dynamic model of a manipulator can be written as

$$\tau = M(q)\ddot{q} + c(q, \dot{q}) + g(q), \quad (4)$$

where $\tau \in \mathbb{R}^a$ represents the joint torque, $M(q) \in \mathbb{R}^{a \times a}$ is the mass matrix, $c(q, \dot{q}) \in \mathbb{R}^a$ is the Coriolis and centrifugal vector, and $g(q) \in \mathbb{R}^a$ denotes the gravity vector. Generally, traditional methods, such as Guo et al. (2018), are capable of performing accurate dynamic control by relying on precise dynamic models (Equation 4). However, in real-world applications, it is common for manipulators to undergo modifications to perform various tasks, resulting in changes in their dynamic parameters. Given the assumption that the change occurs in the inertia matrix, we design an estimated inertia matrix $\bar{M} \in \mathbb{R}^{a \times a}$ to effectively mitigate dynamic uncertainties. As a result, the following state equation is established:

$$\bar{\tau} = \bar{M}(q)\ddot{q} + c(q, \dot{q}) + g(q), \quad (5)$$

where $\bar{\tau} \in \mathbb{R}^a$ is the corresponding joint torque. When the estimated inertia matrix converges to the actual one, it indicates that the dynamic uncertainty issue is solved. To this aim, an estimation equation is presented as follows:

$$\dot{\bar{M}} = \alpha(\tau - \bar{\tau})\bar{q}^\dagger, \quad (6)$$

where $\dot{\bar{M}}$ determines the evolution direction of \bar{M} , and $\alpha > 0$ stands for the convergence coefficient. In Equation 6, τ is measured in real time. Combining Equations 3, 5, 6, an RNN is designed as follows:

$$\dot{\bar{M}} = \alpha(\tau - \bar{M}(q)\ddot{q} + h(q, \dot{q}))\bar{q}^\dagger, \quad (7a)$$

$$\tau_{\text{out}} = \bar{M}(q)(L^\dagger(\ddot{r}_d - \dot{L}\dot{q}) - v) + h(q, \dot{q}), \quad (7b)$$

where τ_{out} is the output signal and $h(q, \dot{q}) = c(q, \dot{q}) + g(q)$. In addition, a control flow chart of RNN (Equation 7) is shown in Figure 1. Notably, the joint acceleration generated by Equation 3 in a kinematic manner serves as the input for Equation 7b. Furthermore, Equation 7a utilizes measurement data τ to estimate the mass matrix, which, in turn, facilitates the precise control of Equation 7b. In this context, RNN (Equation 7) demonstrates its capability to learn the mass matrix and achieve synchronous trajectory tracking via the joint torque. The parameters in RNN (Equation 7) include α , β , and ζ , which are determined through trial and error methods.

In RNN (Equation 7), we first apply kinematic controller (Equation 3) to output the joint acceleration corresponding to the

trajectory task. This process belongs to the inverse kinematics solution. Subsequently, we further obtain the joint torque by calculating the obtained joint acceleration. This process belongs to the inverse dynamics of solution. Finally, the output joint torque can directly control the manipulator to perform the given task. In this control mode, robot kinematics and dynamics are combined together with joint acceleration to form a bridge.

4 Theoretical analysis

The following theorem provides a verification of the learning and control capabilities of the proposed RNN (Equation 7).

Theorems: Assuming a sufficiently large value of α , the estimated error $M - \bar{M}$ generated by Equation 7a is global convergent to a zero matrix. Based on the estimated mass matrix, Equation 7b enables accurate trajectory-tracking control of the manipulator with an unknown mass matrix.

Proof: By incorporating Equations 2, 3, we can rewrite Equation 7b as $\dot{\bar{M}} = \alpha(M - \bar{M})\ddot{q}\bar{q}^\dagger$. Multiplying both sides of the equation by \bar{q} yields $\dot{\bar{M}}\bar{q} = \alpha(M - \bar{M})\bar{q}$. Then, it follows that $\dot{\bar{M}} = \alpha(M - \bar{M})$. Define an estimated error as $e = (M_i - \bar{M}_i)$ with M_i and \bar{M}_i being the i -th column of M and \bar{M} ($i = 1, \dots, a$), respectively. Set a Lyapunov function as $V = (M_i - \bar{M}_i)^T(M_i - \bar{M}_i)$, and then, its time derivative is calculated as follows:

$$\begin{aligned} \dot{V} &= (M_i - \bar{M}_i)^T \dot{M}_i - \alpha(M_i - \bar{M}_i)^T(M_i - \bar{M}_i) \\ &= e^T \dot{M}_i - \alpha e^T e \\ &\leq \|e\|_2 \|\dot{M}_i\|_2 - \alpha \|e\|_2^2 \\ &= \|e\|_2 (\|\dot{M}_i\|_2 - \alpha \|e\|_2), \end{aligned} \quad (8)$$

with $\|\cdot\|_2$ being the Euclidean norm of a vector. The above equation leads to three different situations as follows:

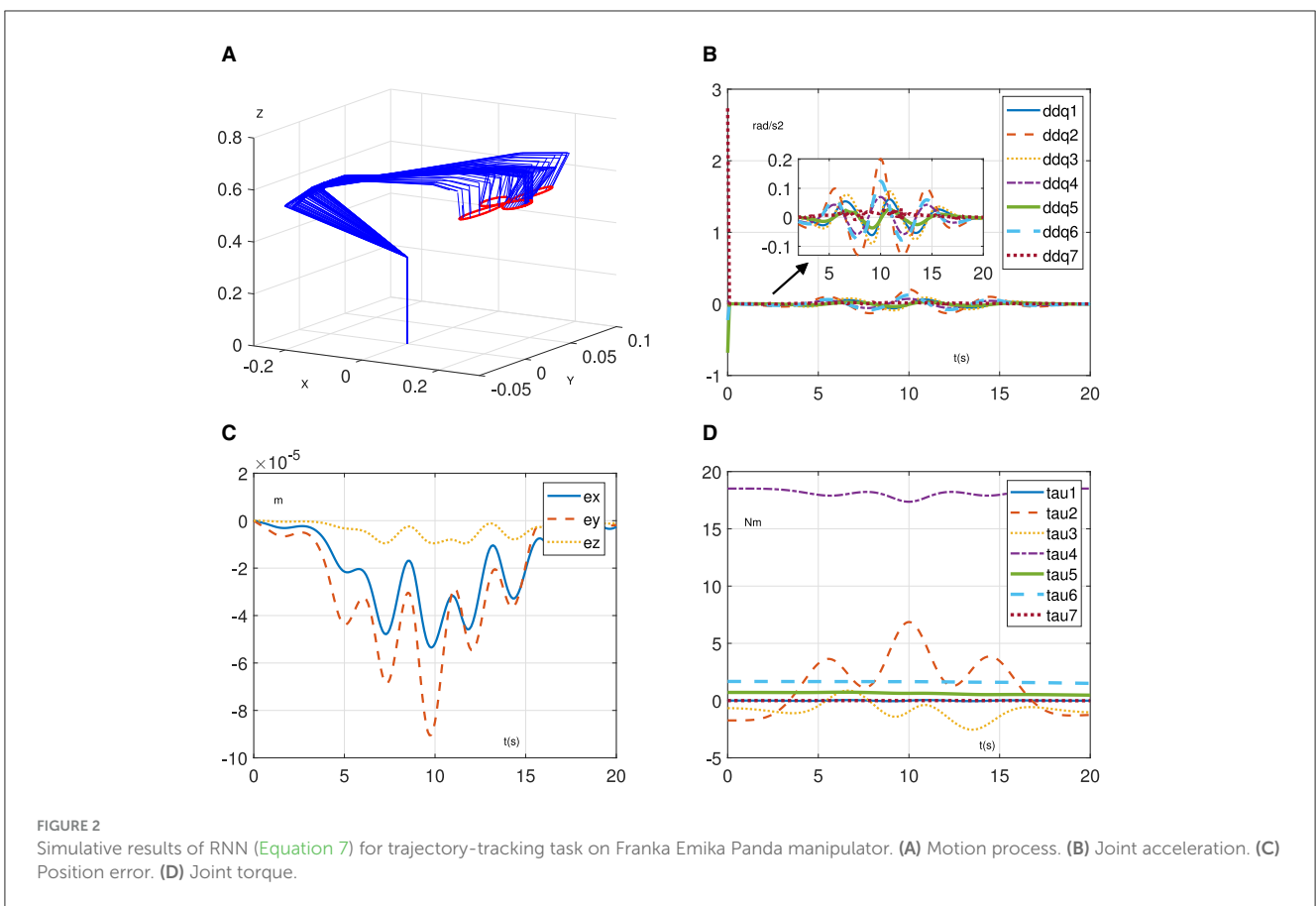
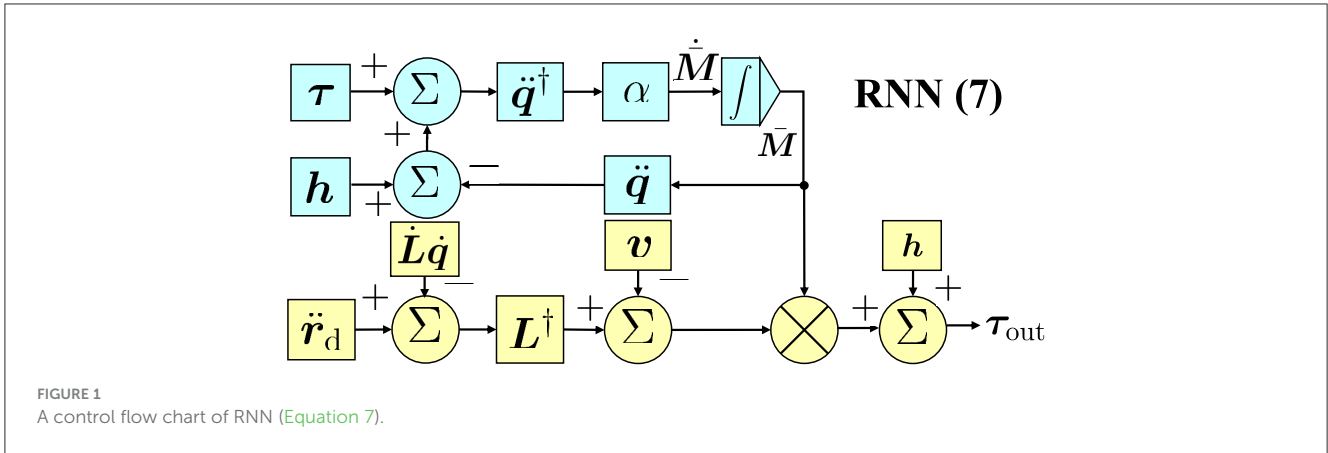
- Situation i: $\|e\|_2 > \|\dot{M}_i\|_2/\alpha$. This situation contributes to $\dot{V} < 0$ and $V > 0$, which implies that $\|e\|_2$ is convergent until $\|e\|_2 = \|\dot{M}_i\|_2$.
- Situation ii: $\|e\|_2 = \|\dot{M}_i\|_2/\alpha$. It leads to $\dot{V} \leq 0$. This suggests that $\|e\|_2$ converges to zero or maintains at $\|e\|_2 = \|\dot{M}_i\|_2$.
- Situation iii: $\|e\|_2 < \|\dot{M}_i\|_2/\alpha$. We deduce that $\dot{V} > 0$ or $\dot{V} \leq 0$. Subsequently, it can be inferred that $\|e\|_2$ continues to increase until it reaches Situation ii or it remains unchanged or convergent.

Considering the above three situations, it can be obtained that $\|e\|_2 \leq \|\dot{M}_i\|_2/\alpha$ when $t \rightarrow \infty$. Provided a sufficiently large value of α , we have that $\|e\|_2$ reaches zero when $t \rightarrow \infty$. In conclusion, the estimated error $M - \bar{M}$ generated by Equation 7b globally converges to a zero matrix. Hence, applying LaSalle's invariance principle (K.Khalil, 2001), we replace \bar{M} with M in Equation 7b and deduce

$$\tau_{\text{out}} = M(q)(L^\dagger(\ddot{r}_d - \dot{L}\dot{q}) - v) + h(q, \dot{q}). \quad (9)$$

Therefore, Equation 7b enables dynamic control of the manipulator depending on the desired joint acceleration in kinematic controller (Equation 3).

The desired joint acceleration allows the manipulator to precisely follow a given trajectory, which is proven through



the following proof. Primarily, (Equation 3) can be equivalently converted into

$$L\ddot{q} + \dot{L}\dot{q} - \ddot{r}_d = \ddot{r} - \ddot{r}_d = -\beta(\dot{r} - \dot{r}_d) - \zeta(r - r_d). \quad (10)$$

Assuming the position error as $u = r - r_d$, the above equation is reorganized as $\ddot{u} + \beta\dot{u} + \zeta u = 0$, which belongs to a second-order differential equation system. The roots of the corresponding characteristic equation are $s_1 = (-\beta + \sqrt{\beta^2 - 4\zeta})/2$ and $s_2 = (-\beta - \sqrt{\beta^2 - 4\zeta})/2$. Furthermore, According to Equations 8–10, we can analyze that the convergence of this system can be categorized into the three cases (Jin et al., 2017).

- Case i: When $\beta^2 - 4\zeta > 0$, we obtain that $s_1 < 0$ and $s_2 < 0$ are real numbers with $s_1 \neq s_2$. Then, the solution satisfies $u(t) = c_1 \exp(s_1 t) + c_2 \exp(s_2 t)$ with $c_1 \in \mathbb{R}^b$ and $c_2 \in \mathbb{R}^b$ being coefficient vectors determined by the initial state of the system.
- Case ii: When $\beta^2 - 4\zeta = 0$, the system has two equivalent characteristic roots with $s_1 = s_2 < 0$. Therefore, the solution can be deduced as $u(t) = (c_1 + c_2 t) \exp(s_1 t)$.
- Case iii: When $\beta^2 - 4\zeta < 0$, $s_1 = z + iy$ and $s_2 = z - iy$ are conjugate complex numbers with $z < 0$. As a result, the solution can be deduced as $u(t) = \exp(zt)(c_1 \cos yt + c_2 \sin yt)$.

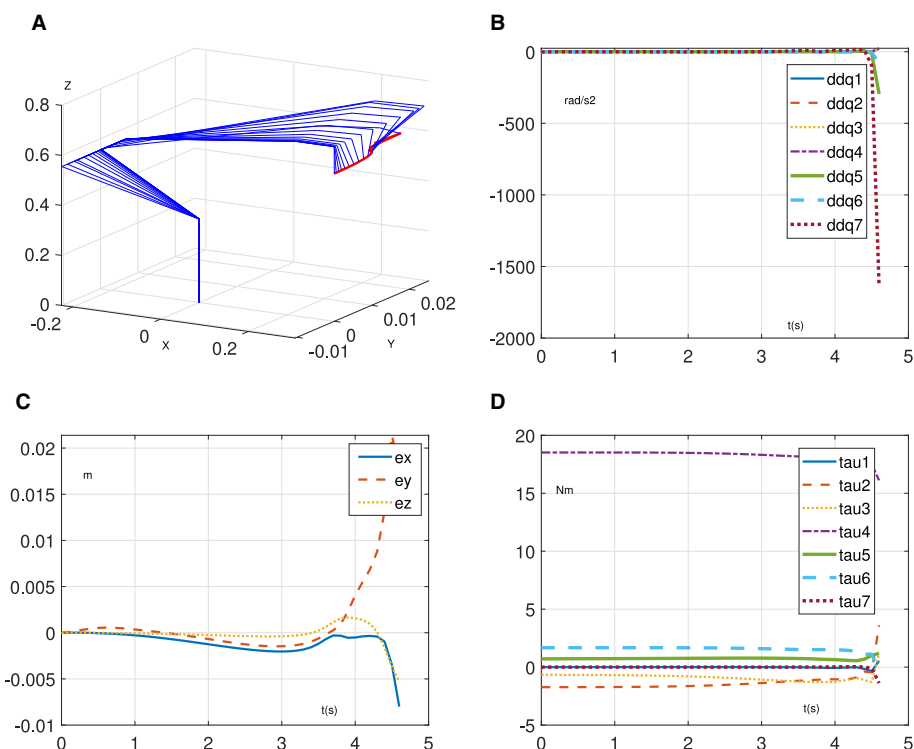


FIGURE 3

Simulative results of Equation 7b without Equation 7a for trajectory-tracking task on Franka Emika Panda manipulator. (A) Motion process. (B) Joint acceleration. (C) Position error. (D) Joint torque.

These cases demonstrate that the position error \mathbf{u} generated by Equation 3 exponentially converges to a zero vector from any initial states. In other words, it is concluded that Equation 7b enables trajectory tracking control of the manipulator with the unknown mass matrix. The proof is thus completed.

5 Simulative results and comparisons

This section provides simulation experiments to demonstrate the learning and control performance of RNN (Equation 7). Specifically, we test it on a 7-degree-of-freedom manipulator called Franka Emika Panda (Liu and Shang, 2024) to task a four-leaf clover path with $\alpha = 10^4$, $\beta = 1$, and $\zeta = 5$. In addition, we assume that the mass matrix is unknown and design a random noise matrix with elements <0.5 to represent its uncertainties. The related results are shown in Figures 2, 3. Figure 2A demonstrates the effectiveness of the proposed method in enabling the manipulator to accomplish trajectory-tracking tasks, even in the presence of an unknown mass matrix. In addition, the initial joint acceleration in Figure 2B is relatively large due to the initial mass matrix error and becomes smooth and normal. Furthermore, the position error keeps the order of 10^{-5} m in Figure 2C. Similarly, in Figure 2D, it can be observed that the joint torque exhibits reasonable variations. However, when the estimation Equation 7b is not considered, achieving the trajectory tracking task based on Equation 7b becomes challenging due to the presence of the unknown mass matrix. As shown in Figure 3A, the

manipulator driven by Equation 7a cannot complete the tracking task. Evidently, the joint acceleration became uncontrollable at ~ 4.5 s, as shown in Figure 3B, and the manipulator system is no longer operational. Furthermore, the position error exhibits divergence in Figure 3C. Similarly, the joint torque in Figure 3D is out of control at 4.5 s. Through the above results, the learning and control ability of the proposed method are verified.

To further demonstrate the feasibility of the proposed method, we additionally apply the proposed RNN (Equation 7) to control the Franka Emika Panda manipulator performing a Lissajous trajectory-tracking task. It is noteworthy that the parameters involved are identical to the previous simulation, except for the trajectory-tracking task. The specific results are presented in Figure 4. Specifically, Figures 4A, B demonstrate that the manipulator successfully executes the given trajectory tracking task, taking into account dynamics uncertainties. Furthermore, the positional error, as shown in Figure 4C, is maintained at the order of 10^{-5} m. Additionally, the joint acceleration exhibits normal variations, as shown in Figure 4D. In addition, Figures 4E, F illustrate that the proposed method is capable of compensating for the dynamics uncertainties, with tiny estimated errors of the joint torque. The aforementioned results indicate the effectiveness of the proposed RNN (Equation 7).

In addition, the advantages of the proposed RNN are shown in Table 1, compared with the existing methods. One notable advantage of the proposed RNN (Equation 7) is its simultaneous consideration of both the kinematic and dynamic models. This approach enables the realization

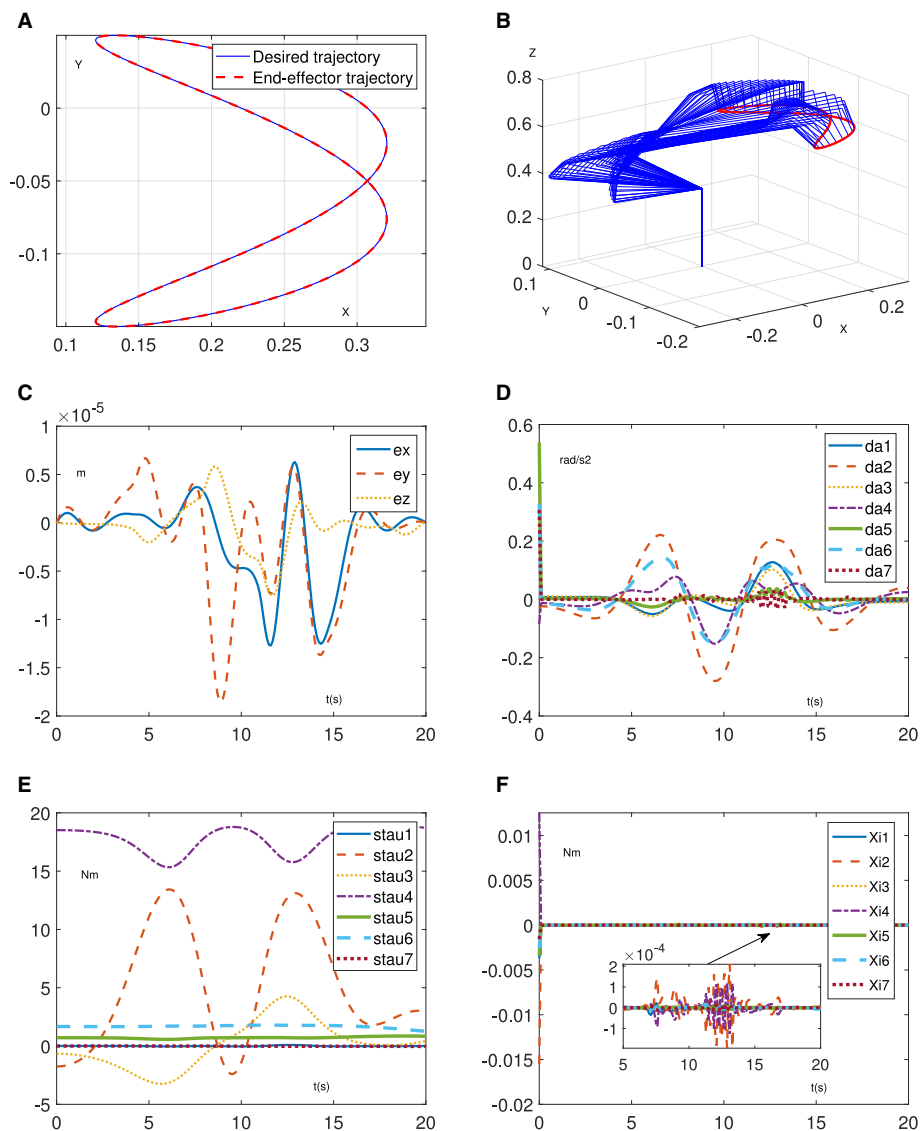


FIGURE 4 Simulative results of RNN (Equation 7) for trajectory-tracking task on Franka Emika Panda manipulator. (A) Desired trajectory and end-effector trajectory. (B) Motion process of manipulator. (C) Position error. (D) Joint acceleration. (E) Estimated joint torque. (F) Joint torque error with $\xi = \tau - \bar{\tau}$.

TABLE 1 Comparisons among different methods for controlling manipulator.

Different methods	Dynamic control	Unknown issue	Trajectory tracking	Mass matrix Online estimation
RNN (Equation 7)	Yes	Yes	Yes	Yes
Guo et al. (2018)	No	No	Yes	No
Jin et al. (2023)	No	No	Yes	No
Ma et al. (2023b)	No	No	Yes	No
Jin et al. (2017)	No	No	Yes	No
Liu and Shang (2024)	No	No	Yes	No
Shojaei et al. (2021)	Yes	Yes	No	No
Zong and Emami (2021)	Yes	No	No	No

of online estimation of the mass matrix and synchronous trajectory tracking.

6 Conclusion

In this study, we have proposed a recurrent neural network (RNN) to address the challenges of trajectory tracking in manipulator systems with unknown mass matrices. The key idea of our proposed RNN is to establish a connection between the kinematics and dynamics models using joint acceleration signals, considering the motion characteristics and dynamic behavior of manipulators. Primarily, it has incorporated a kinematic controller to generate the desired joint acceleration based on the given task. On this basis, the robot dynamics model and a mass matrix estimator have been designed and integrated into the RNN to enable trajectory tracking in the presence of an unknown mass matrix. Subsequently, theoretical analysis has demonstrated the learning and control capabilities of the RNN. Through simulation experiments and comparisons, we have validated the effectiveness and superiority of the proposed RNN for trajectory tracking control of the manipulator with unknown mass matrix.

In addition to the robot's mass matrix, other dynamic parameters of the manipulator, such as the gravity vector, may also change. In addition, joint constraints help to improve the safety of robot operation. Therefore, future research will focus on estimating multiple dynamic parameters and considering multiple levels of joint constraints.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

References

- Abate, A. F., Cimmino, L., Cuomo, I., Nardo, M. D., and Murino, T. (2022). On the impact of multimodal and multisensor biometrics in smart factories. *IEEE Trans. Industr. Informat.* 18, 9092–9100. doi: 10.1109/TII.2022.3178376
- Bruder, D., Fu, X., Gillespie, R. B., Remy, C. D., and Vasudevan, R. (2021). Koopman-based control of a soft continuum manipulator under variable loading conditions. *IEEE Robot. Automat. Lett.* 6, 6852–6859. doi: 10.1109/LRA.2021.3095268
- Cao, X., Peng, C., Zheng, Y., Li, S., Ha, T. T., Shutyaev, V., et al. (2023). Neural networks for portfolio analysis in high-frequency trading. *IEEE Trans. Neural Netw. Learn. Syst.* 2023:3311169. doi: 10.1109/TNNLS.2023.3311169
- Guo, D., Xu, F., and Yan, L. (2018). New pseudoinverse-based path-planning scheme with PID characteristic for redundant robot manipulators in the presence of noise. *IEEE Trans. Contr. Syst. Technol.* 26, 2008–2019. doi: 10.1109/TCST.2017.2756029
- Hua, C., Cao, X., Liao, B., and Li, S. (2023). Advances on intelligent algorithms for scientific computing: an overview. *Front. Neurobot.* 17, 1–21. doi: 10.3389/fnbot.2023.1190977
- Jin, L., Liu, L., Wang, X., Shang, M., and Wang, F. Y. (2024a). Physical-informed neural network for MPC-based trajectory tracking of vehicles with noise considered. *IEEE Trans. Intell. Vehicl.* 9, 4493–4503. doi: 10.1109/TIV.2024.3358229
- Jin, L., Zhang, F., Liu, M., and Xu, S. S. D. (2023). Finite-time model predictive tracking control of position and orientation for redundant manipulators. *IEEE Trans. Industr. Electr.* 70, 6017–6026. doi: 10.1109/TIE.2022.3196372
- Jin, L., Zhang, Y., Li, S., and Zhang, Y. (2017). Noise-tolerant ZNN models for solving time-varying zero-finding problems: a control-theoretic approach. *IEEE Trans. Automat. Contr.* 62, 992–997. doi: 10.1109/TAC.2016.2566880
- Jin, L., Zhao, J., Chen, L., and Li, S. (2024b). Collective neural dynamics for sparse motion planning of redundant manipulators without hessian matrix inversion. *IEEE Trans. Neural Netw. Learn. Syst.* 2024:3363241. doi: 10.1109/TNNLS.2024.3363241
- Khalil, H. (2001). *Nonlinear Systems, 3rd Edn.* Englewood Cliffs, NJ: Prentice Hall.
- Lian, Y., Xiao, X., Zhang, J., Jin, L., Yu, J., and Sun, Z. (2024). Neural dynamics for cooperative motion control of omnidirectional mobile manipulators in the presence of noises: a distributed approach. *IEEE/CAA J. Automat. Sin.* 11, 1605–1620. doi: 10.1109/JAS.2024.124425
- Liao, B., Hua, C., Xu, Q., Cao, X., and Li, S. (2023). A predefined-time and anti-noise varying-parameter ZNN model for solving time-varying complex stein equations. *Neurocomputing* 526, 158–168. doi: 10.1016/j.neucom.2023.01.008
- Liao, B., Hua, C., Xu, Q., Cao, X., and Li, S. (2024). Inter-robot management via neighboring robot sensing and measurement using a zeroing neural dynamics approach. *Expert Syst. Appl.* 244:122938. doi: 10.1016/j.eswa.2023.122938
- Liao, B., Wang, Y., Li, J., Guo, D., and He, Y. (2022). Harmonic noise-tolerant ZNN for dynamic matrix pseudoinversion and its application to robot manipulator. *Front. Neurobot.* 16, 1–13. doi: 10.3389/fnbot.2022.928636
- Liu, M., Li, Y., Chen, Y., Qi, Y., and Jin, L. (2024a). A distributed competitive and collaborative coordination for multirobot systems. *IEEE Trans. Mob. Comput.* 2024:3397242. doi: 10.1109/TMC.2024.3397242

Author contributions

JL: Funding acquisition, Investigation, Methodology, Project administration, Writing – review & editing, Supervision. JS: Data curation, Writing – original draft. HF: Methodology, Supervision, Writing – review & editing. WY: Methodology, Writing – review & editing. XM: Methodology, Writing – review & editing. ZL: Data curation, Methodology, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This research was funded by the Changchun Science and Technology Development Program, grant number: 21ZGN26 and by the Jilin Province Science and Technology Development Program, grant number: 20230508026RC.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- Liu, M., Liu, K., Zhu, P., Zhang, G., Ma, X., and Shang, M. (2024b). Data-driven remote center of cyclic motion (RC²M) control for redundant robots with rod-shaped end-effector. *IEEE Trans. Industr. Informat.* 20, 6772–6780. doi: 10.1109/TII.2024.3353930
- Liu, M., and Shang, M. (2024). Orientation tracking incorporated multicriteria control for redundant manipulators with dynamic neural network. *IEEE Trans. Industr. Electr.* 71, 3801–3810. doi: 10.1109/TIE.2023.3273253
- Ma, B., Jiang, Z., Liu, Y., and Xie, Z. (2023a). Advances in space robots for on-orbit servicing: a comprehensive review. *Adv. Intell. Syst.* 5, 1–21. doi: 10.1002/aisy.202200397
- Ma, B., Xie, Z., Zhan, B., Jiang, Z., Liu, Y., and Liu, H. (2023b). Actual shape-based obstacle avoidance synthesized by velocity–acceleration minimization for redundant manipulators: an optimization perspective. *IEEE Trans. Syst. Man Cybernet.* 53, 6460–6474. doi: 10.1109/TSMC.2023.3283266
- Müller, A., Kumar, S., and Kordik, T. (2023). A recursive lie-group formulation for the second-order time derivatives of the inverse dynamics of parallel kinematic manipulators. *IEEE Robot. Automat. Lett.* 8, 3804–3811. doi: 10.1109/LRA.2023.3267005
- Shojaei, K., Kazemy, A., and Chatraei, A. (2021). An observer-based neural adaptive PID² controller for robot manipulators including motor dynamics with a prescribed performance. *IEEE/ASME Trans. Mechatr.* 26, 1689–1699. doi: 10.1109/TMECH.2020.3028968
- Sun, Z., Tang, S., Jin, L., Zhang, J., and Yu, J. (2023a). Nonconvex activation noise-suppressing neural network for time-varying quadratic programming: application to omnidirectional mobile manipulator. *IEEE Trans. Industr. Informat.* 19, 10786–10798. doi: 10.1109/TII.2023.3241683
- Sun, Z., Tang, S., Zhang, J., and Yu, J. (2023b). Nonconvex noise-tolerant neural model for repetitive motion of omnidirectional mobile manipulators. *IEEE/CAA J. Automat. Sin.* 10, 1766–1768. doi: 10.1109/JAS.2023.123273
- Sun, Z., Wang, G., Jin, L., Cheng, C., Zhang, B., and Yu, J. (2022). Noise-suppressing zeroing neural network for online solving time-varying matrix square roots problems: a control-theoretic approach. *Expert Syst. Appl.* 192:116272. doi: 10.1016/j.eswa.2021.116272
- Sun, Z., Xu, C., Gu, J., Zhao, L., and Hu, Y. (2024). Design, modeling and optimal control of a novel compliant actuator. *Contr. Eng. Pract.* 148:105967. doi: 10.1016/j.conengprac.2024.105967
- Tang, Z., and Zhang, Y. (2022). Refined self-motion scheme with zero initial velocities and time-varying physical limits via zhang neurodynamics equivalency. *Front. Neurobot.* 16, 1–15. doi: 10.3389/fnbot.2022.945346
- Tang, Z., Zhang, Y., and Ming, L. (2024). Novel snap-layer MMPC scheme via neural dynamics equivalency and solver for redundant robot arms with five-layer physical limits. *IEEE Trans. Neural Netw. Learn. Syst.* 2024:3351674. doi: 10.1109/TNNLS.2024.3351674
- Wen, L., and Xie, Z. (2024). A data-driven acceleration-level scheme for image-based visual servoing of manipulators with unknown structure. *Front. Neurobot.* 18, 1–14. doi: 10.3389/fnbot.2024.1380430
- Xiao, L., He, Y., Dai, J., Liu, X., Liao, B., and Tan, H. (2022). A variable-parameter noise-tolerant zeroing neural network for time-variant matrix inversion with guaranteed robustness. *IEEE Trans. Neural Netw. Learn. Syst.* 33, 1535–1545. doi: 10.1109/TNNLS.2020.3042761
- Xiao, L., He, Y., Wang, Y., Dai, J., Wang, R., and Tang, W. (2023). A segmented variable-parameter ZNN for dynamic quadratic minimization with improved convergence and robustness. *IEEE Trans. Neural Netw. Learn. Syst.* 34, 2413–2424. doi: 10.1109/TNNLS.2021.3106640
- Xiao, L., Zhang, Y., Liao, B., Zhang, Z., and Jin, L. (2017). A velocity-level Bi-criteria optimization scheme for coordinated path tracking of dual robot manipulators using recurrent neural network. *Front. Neurobot.* 11, 1–7. doi: 10.3389/fnbot.2017.00047
- Xie, Z., and Jin, L. (2024). A fuzzy neural controller for model-free control of redundant manipulators with unknown kinematic parameters. *IEEE Trans. Fuzzy Syst.* 32, 1589–1601. doi: 10.1109/TFUZZ.2023.3328545
- Xie, Z., Jin, L., Luo, X., Hu, B., and Li, S. (2022). An acceleration-level data-driven repetitive motion planning scheme for kinematic control of robots with unknown structure. *IEEE Trans. Syst. Man Cybernet.* 52, 5679–5691. doi: 10.1109/TSMC.2021.3129794
- Xie, Z., Li, S., and Jin, L. (2023). A Bi-criteria kinematic strategy for motion/force control of robotic manipulator. *IEEE Trans. Automat. Sci. Eng.* 2023:3313564. doi: 10.1109/TASE.2023.3313564
- Xie, Z., Zheng, Y., and Jin, L. (2024). A data-driven image-based visual servoing scheme for redundant manipulators with unknown structure and singularity solution. *IEEE Trans. Syst. Man Cybernet.* 2024:3420882. doi: 10.1109/TSMC.2024.3420882
- Yan, J., Liu, M., and Jin, L. (2024). Cerebellum-inspired model predictive control for redundant manipulators with unknown structure information. *IEEE Trans. Cogn. Dev. Syst.* 16, 1198–1210. doi: 10.1109/TCDS.2023.3340179
- Zhang, Y., Li, S., Kadry, S., and Liao, B. (2019). Recurrent neural network for kinematic control of redundant manipulators with periodic input disturbance and physical constraints. *IEEE Trans. Cybernet.* 49, 4194–4205. doi: 10.1109/TCYB.2018.2859751
- Zong, L., and Emami, M. R. (2021). Control verifications of space manipulators using ground platforms. *IEEE Trans. Aerospace Electr. Syst.* 57, 341–354. doi: 10.1109/TAES.2020.3016877