2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-7281-6212-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/IROS45743.2020.9341389

Hierarchical optimization Control of Redundant Manipulator for Robot-assisted Minimally Invasive Surgery

Yingbai Hu[1], Hang Su[2], Guang Chen[3*], Giancarlo Ferrigno[2], Elena De Momi[2], and Alois Knoll[1]

***Abstract*— For the time varying optimization problem, the tracking error cannot converge to zero at the finite time because of the optimal solution changing over time. This paper proposes a novel varying parameter recurrent neural network (VPRNN) based hierarchical optimization of a 7-DoF surgical manipulator for Robot-Assisted Minimally Invasive Surgery (RAMIS), which guarantees task tracking, Remote Center of Motion (RCM) and manipulability index optimization. A theoretically grounded hierarchical optimization framework based is introduced to control multiple tasks based on their priority. Finally, the effectiveness of the proposed control strategy is demonstrated with both simulation and experimental results. The results show that the proposed VPRNN-based method can optimal three tasks at the same time and have better performance than previous work.**

I. INTRODUCTION

Robotic technology is increasingly implemented to assist surgeon [1]. Robot-assisted surgery has several advantages such as better surgical accuracy, increased workspace, enhanced dexterity, and improved vision for surgeons [2]. Multiple tasks need to be considered during the surgical operation [3], such as the control of the surgical tip and the manipulability of the surgical manipulator, which is vital in Robot-Assisted Minimally Invasive Surgery (RAMIS). The accurate tracking control of the surgical tip is of vital importance for surgical operations [4]. Manipulability index [5], [6] of a surgical tool tip determines the maximum distance from singularities, flexible motion, and more extensive operational space of the robot manipulators. Usually, for surgical robot operation, multiple tasks are characterized by different priority levels. The multiple general tasks are listed as follows (T1–T3):

T1: The tracking control of the surgical tip must be accurate, which guarantees the success rate of surgery using the robot manipulator [7].

T2: In order to ensure the safety and rationality of the operation, the surgical tip should pass through a small incision in the abdominal wall of the patient. Kinematic constraints produced by each small incision should be respected, generally identified as the Remote Center of Motion (RCM) constraint [8].

T3: The manipulability of the robot manipulator should be enough to perform the surgical operation [9].

To effectively evaluate the multiple operational tasks on the robot manipulator, a lot of research activity has been attracted and performed in this area. Various approaches have been implemented to track the desired position and guarantee the RCM constraint at the same time using kinematic solutions [8]. Similarly, Su *et al.* [10] introduced an adaptive decoupling impedance controller to exploit the redundancy of the manipulator to maintain the RCM constraint, as well. Except for fulfilling the RCM constraint, Jin *et al.* [11] proposed to adopt neural networks to optimize the manipulability index during the tracking without influence its accuracy. Nevertheless, few works are proposed to handle all the operational tasks at the same time.

In our previous work [12], [13], the hierarchical operational space formulation is utilized to unite the three parts: the main surgical tracking task based on the Cartesian compliance control, the RCM constraint in its null space of the end-effector, and a manipulability optimization control in the null space of the robot wrist using a constrained quadratic programming (QP). Although it achieved better performance in terms of accuracy and manipulability index, the convergence rate is slow, and the tracking error is larger.

In this paper, varying parameters recurrent neural network (VPRNN) based hierarchical control of a 7-DoF robot manipulator for robot-assisted minimally invasive surgery has been proposed, where it integrates multiple tasks based on their priority and guarantees task tracking [14], RCM, and manipulability optimization at the same time. The gradient descent neural network-based traditional methods cannot guarantee that the convergence of the error to 0 at the finite time because of the optimal solution changing over time. The proposed VPRNN framework can solve the time-vary QP problem with fast convergence rate performance, which is promising for online solution of the time-varying optimization problem [15]. The proposed methodology represents an advance concerning the manipulability optimization solution proposed in [12]. It utilizes a VPRNN based hierarchical control to combine the multiple tasks into a single controller [16]. Furthermore, the overall convergence rate has also been improved with respect to our previous works [11],

*This work was supported in part by the European Commission Horizon 2020 research and innovation program, under the project SMARTsurg, grant agreement No. 732515 and in part by the Human Brain Project SGA2, under the Specific Grant Agreement No. 785907. Guang Chen is the corresponding author.

[1]Yingbai Hu and Alois Knoll are with the Department of Informatics, Technical University of Munich, Munich, 85748, Germany. yingbai.hu@tum.de; knoll@in.tum.de

[2]Hang Su, Giancarlo Ferrigno and Elena De Momi are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133, Milano, Italy. {hang.su, giancarlo.ferrigno, elena.demomi}@polimi.it

[3]Guang Chen is with College of Automotive Engineering, Tongji University, Shanghai, 201804, China and with the Department of Informatics, Technical University of Munich, Munich, 85748, Germany. guangchen@tongji.edu.cn

Authorized licensed use limited to: Technische Universitaet Muenchen. Downloaded on May 11,2022 at 13:13:15 UTC from IEEE Xplore. Restrictions apply.

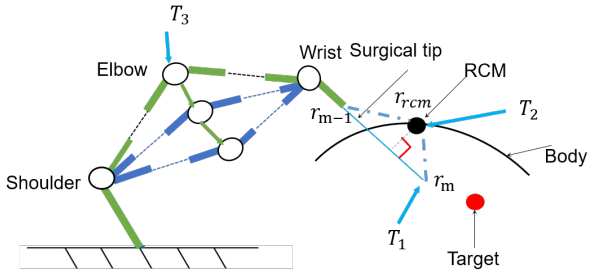

Fig. 1. Multiple operational tasks in the Robot-assisted Minimally Invasive Surgery

[12]. Finally, simulations and experiments applying a 7-DoF serial robot KUKA LWR4+ are performed to demonstrate the effectiveness of the proposed method.

II. Related Works

Generally, there are active and passive RCM constraints. Moreover, it is prevalent in non-clinical research since its low cost and flexible task space. As it is shown in Fig. 1, the multiple operational tasks (T1–T3) should be considered at the same time.

Sandoval *et al.* [16] proposed to exploit the redundancy to combine the T1 and the T2 at the same time. In fact, an improved dynamic control method is proposed to apply the task redundancy for the RCM constraint (T2), without the influence of the surgical operation (T1). Jin *et al.* [11] proposed to use neural networks to combine the T1 and the T3. In our previous works [12], [17], we utilized the hierarchical operational space formulation [18] to combine the three tasks T1, T2, and T3.

III. Control Methodology

A. Remote Center of Motion Constraint

For the surgical tracking tasks, the robot's tooltip should respect the RCM constraint, which is shown in Fig. 1. Obviously, $\boldsymbol{r}_{rcm}$ should be held in a line of $\boldsymbol{r}_m$ and $\boldsymbol{r}_{m-1}$, where $\boldsymbol{r}_m$ is the end of tooltip and $\boldsymbol{r}_{m-1}$ is the Cartesian position of joint $m-1$. Thus, the geometric relationship of RCM error model is defined as,

$$\overrightarrow{\boldsymbol{r}_{m-1}\boldsymbol{r}_{rcm}} \times \overrightarrow{\boldsymbol{r}_{m-1}\boldsymbol{r}_m} = \boldsymbol{0} \tag{1}$$

where $\overrightarrow{\boldsymbol{r}_{m-1}\boldsymbol{r}_{rcm}}$ represents the vector of line $\boldsymbol{r}_{m-1}\boldsymbol{r}_{rcm}$, and $\overrightarrow{\boldsymbol{r}_{m-1}\boldsymbol{r}_m}$ represent the vector of line $\boldsymbol{r}_{m-1}\boldsymbol{r}_m$, respectively. It can be seen that line $\boldsymbol{r}_{m-1}\boldsymbol{r}_{rcm}$ and line $\boldsymbol{r}_{m-1}\boldsymbol{r}_m$ should be parallel. Then, the RCM error constraint can be expressed as,

$$\boldsymbol{E}_{rcm}(\boldsymbol{q}) = \frac{\overrightarrow{\boldsymbol{r}_{m-1}\boldsymbol{r}_{rcm}} \times \overrightarrow{\boldsymbol{r}_{m-1}\boldsymbol{r}_m}}{d} \tag{2}$$

B. Problem formulation

The kinematic formulation of multi-DOFs redundant manipulator is,

$$\begin{aligned} \boldsymbol{X}_d &= f_1(\boldsymbol{q}) \\ \boldsymbol{E}_{rcm} &= f_2(\boldsymbol{q}) \end{aligned} \tag{3}$$

where $\boldsymbol{X}_d \in \mathbb{R}^n$ and $\boldsymbol{E}_{rcm} \in \mathbb{R}^n$ $(n=3)$ are end-effector and RCM Cartesian coordinate, respectively. The relationship between the end-effector velocity $\dot{\boldsymbol{X}}$ and the joint velocity $\dot{\boldsymbol{q}} \in \mathbb{R}^m (m=7)$ is expressed as,

$$\begin{aligned} \dot{\boldsymbol{X}}_d &= \boldsymbol{J}_T \dot{\boldsymbol{q}} \\ \dot{\boldsymbol{E}}_{rcm} &= \boldsymbol{J}_{rcm} \dot{\boldsymbol{q}} \end{aligned} \tag{4}$$

where $\boldsymbol{J}_T \in \mathbb{R}^{n\times m}$ and $\boldsymbol{J}_{rcm} \in \mathbb{R}^{n\times m}$ denote the end-effector and RCM Jacobian matrix, respectively. As it is well-known the manipulability measure gives a scalar representation of the gain between joint velocities $\dot{\boldsymbol{q}}$ and task velocities $\dot{\boldsymbol{X}}$, and, consequently, measures the ability of the robot to move its end-effector, i.e., a configuration $\boldsymbol{q}$ for which the Jacobian is rank deficient. The manipulability measure depends on the $\boldsymbol{J}_T$ and is given by:

$$\boldsymbol{\mu} = \sqrt{det\left(\boldsymbol{J}_T(\boldsymbol{q})\,\boldsymbol{J}_T^{\mathrm{T}}(\boldsymbol{q})\right)} = \sqrt{\mu_1\mu_2\ldots\mu_m} \tag{5}$$

where $\mu_i (i=1,2\ldots,m)$ denotes the i-th largest eigenvalue of $\boldsymbol{J}_T\boldsymbol{J}_T^T$

For a redundant manipulator with the desired workspace task $\boldsymbol{X}_d$ and a RCM constraint $\boldsymbol{E}_{rcm}$, the manipulability optimization problem can be formulated as [11]:

$$\begin{aligned} \min \quad & -\boldsymbol{\mu} && (6)\\ \text{s.t.} \quad & \boldsymbol{X}_d = f_1(q) && (7)\\ & \boldsymbol{E}_{rcm}(\boldsymbol{q}) = f_2(q) && (8)\\ & \boldsymbol{q}, \dot{\boldsymbol{q}} \in \mathbb{R}^m \end{aligned}$$

where $f_1(\boldsymbol{q})$ and $f_2(\boldsymbol{q})$ are the forward kinematic functions of the end-effector and of the "wrist", respectively. The two equality constraints has been introduced to guarantee that the optimization of the manipulability index does not affect tracking of the desired end-effector trajectory (first equality constraint) and fulfilment of RCM constraint (second equality constraint).

The previous optimization problem is characterized by a cost function that is usually non-convex, and by nonlinear equality constraints, and it thus represents a challenging problem. In order to address the non-convexity of the cost function, we can reformulate the optimization problem as follows:

$$\begin{aligned} \min \quad & -\dot{\boldsymbol{\mu}} && (9)\\ \text{s.t.} \quad & \boldsymbol{J}\dot{q} = \boldsymbol{v}_d && (10)\\ & \boldsymbol{J}_{rcm}(\boldsymbol{q}) = \boldsymbol{v}_{rcm} && (11)\\ & \boldsymbol{q}, \dot{\boldsymbol{q}} \in \mathbb{R}^m \end{aligned}$$

where $v_d = \dot{X}_d$, and $v_{rcm} = \dot{E}_{rcm}$

On the other side, a way to handle the two nonlinear equality constraints is introduced in Section III-C.

Considering the safety in surgical task [19], [20], the joint velocities and joint angles cannot exceed kinematic limitations. The convex sets of admissible joint positions and velocities can be introduced as follows:

$$\begin{aligned} \Omega_{\boldsymbol{q}} &= \{q_i \in \mathbb{R} \mid \underline{q}_i \le q_i \le \bar{q}_i,\ i=1,2,\ldots,m\} \\ \Omega_{\dot{\boldsymbol{q}}} &= \{\dot{q}_i \in \mathbb{R} \mid \underline{\dot{q}}_i \le \dot{q}_i \le \bar{\dot{q}}_i,\ i=1,2,\ldots,m\} \end{aligned}$$

Authorized licensed use limited to: Technische Universitaet Muenchen. Downloaded on May 11,2022 at 13:13:15 UTC from IEEE Xplore. Restrictions apply.

where $\underline{\dot{q}}_i$ and $\bar{\dot{q}}_i$ are lower and upper bounds on joint velocities, respectively, and $\underline{q}_i$ and $\bar{q}_i$ are lower and upper bounds on joint angles, respectively.

In order to express these two constraints as a single constraint on joint positions and velocities, the constraint on joint positions can be reformulated [11] as follows:

$$\Omega_{\boldsymbol{q}} = \{\dot{q}_i \in \mathbb{R} \mid -\alpha\left(q_i - \underline{q}_i\right) \leq \dot{q}_i \leq -\alpha\left(q_i - \bar{q}_i\right),\ i = 1, 2, \ldots, m\}$$

where α is a positive constant, and the two constraints can be rewritten as:

$$\Omega_{\dot{\boldsymbol{q}}} = \{\dot{q}_i \in \mathbb{R} \mid \max(\underline{\dot{q}}_i, -\alpha(q_i - \underline{q}_i)) \leq \dot{q}_i \leq \min(\bar{\dot{q}}_i, -\alpha(q_i - \bar{q}_i)),\ i = 1, 2, \ldots, m\}$$

The derivative level manipulability can be obtained as,

$$\begin{cases} \frac{d(\boldsymbol{\mu}^2/2)}{dt} = \det\left(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right) tr\left(\dot{\boldsymbol{J}}_T\boldsymbol{J}_T^{\mathrm{T}}(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right) \\ \frac{d(\boldsymbol{\mu}^2/2)}{dt} = \boldsymbol{\mu}\dot{\boldsymbol{\mu}} = \sqrt{\det\left(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)}\dot{\boldsymbol{\mu}} \end{cases} \tag{12}$$

From (12), the $\dot{\boldsymbol{\mu}}$ can be obtained,

$$\dot{\boldsymbol{\mu}} = \sqrt{\det\left(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)} tr\left(\dot{\boldsymbol{J}}_T\boldsymbol{J}_T^{\mathrm{T}}(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right) \tag{13}$$

We define the $\dot{\boldsymbol{J}}_T$ as,

$$\dot{\boldsymbol{J}}_T = \sum_{i=1}^{m} \frac{\partial \boldsymbol{J}_T}{\partial q_i}\dot{q}_i = \sum_{i=1}^{m} h_i\dot{q}_i \tag{14}$$

Therefore,

$$tr\left(\dot{\boldsymbol{J}}_T\boldsymbol{J}_T^{\mathrm{T}}(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right) = \sum_{i=1}^{m}\dot{q}_i \cdot tr\left(h_i\boldsymbol{J}_T^{\mathrm{T}}(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right) \tag{15}$$

Considering the convenience of computing of $(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}$, we transform the variable with vectorization as,

$$\begin{aligned} &tr\left(h_i\boldsymbol{J}_T^{\mathrm{T}}(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right) = tr\left(\left(\boldsymbol{J}_T h_i^{\mathrm{T}}\right)^{\mathrm{T}}\left((\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right)\right) \\ &= \mathrm{vec}^{\mathrm{T}}\left(\boldsymbol{J}_T h_i^{\mathrm{T}}\right)\mathrm{vec}\left(\left((\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right)\right) \end{aligned} \tag{16}$$

Therefore the $\dot{\boldsymbol{\mu}}$ can be further expressed as,

$$\begin{aligned} \dot{\boldsymbol{\mu}} &= \boldsymbol{\mu}\sum_{i=1}^{m}\dot{q}_i\mathrm{vec}^{\mathrm{T}}\left(\boldsymbol{J}_T h_i^{\mathrm{T}}\right)\mathrm{vec}\left((\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right) \\ &= \boldsymbol{\mu}\left[\dot{q}_i, \dot{q}_2 \ldots, \dot{q}_m\right]\left[d_1, d_2 \ldots, d_m\right]^{\mathrm{T}}\mathrm{vec}\left((\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right) \end{aligned} \tag{17}$$

where $d_i = \mathrm{vec}^{\mathrm{T}}\left(\boldsymbol{J}_T h_i^{\mathrm{T}}\right)$. Moreover, the new symbol '$\diamond$' is defined to simplify the expression in (17),

$$\begin{aligned} &\boldsymbol{J}_T\diamondsuit h = \left[d_1, d_2 \ldots, d_m\right]^{\mathrm{T}} = \\ &\left[\mathrm{vec}^{\mathrm{T}}\left(h_1^{\mathrm{T}}\right), \mathrm{vec}^{\mathrm{T}}\left(h_2^{\mathrm{T}}\right) \ldots, \mathrm{vec}^{\mathrm{T}}\left(h_m^{\mathrm{T}}\right)\right]^{\mathrm{T}}\left(I_n \otimes \boldsymbol{J}_T^{\mathrm{T}}\right) \end{aligned} \tag{18}$$

where the rules of Kronecker product '$\otimes$' satisfy: $\mathrm{vec}\,(abc) = \left(b^{\mathrm{T}} \otimes \mathrm{a}\right)\mathrm{vec}\,(c)$.

We define the notation $\boldsymbol{\psi} = \mathrm{vec}\left((\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}})^{-1}\right)$ ($\boldsymbol{\psi} \in \mathbb{R}^{n^2}$). Therefore, $\dot{\boldsymbol{\mu}} = \boldsymbol{\mu}\dot{\boldsymbol{q}}^{\mathrm{T}}(\boldsymbol{J}_T\diamondsuit h)\boldsymbol{\psi}$, and $I_n = \boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\left(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)^{-1} = \boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\boldsymbol{\psi}$, $\mathrm{vec}\,(I_n) = \mathrm{vec}\left(\boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\boldsymbol{\psi}\right) = \left(I_n \otimes \boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)\boldsymbol{\psi}$.

It should be noted that $\boldsymbol{\mu}$ is nonnegative and independent of vector $\dot{\boldsymbol{q}}$ and $\boldsymbol{\psi}$, so the optimization function is equivalent to $\dot{\boldsymbol{q}}^{\mathrm{T}}(\boldsymbol{J}_T\diamondsuit h)\boldsymbol{\psi}$. Therefore, the manipulability optimization problem, including all the aforementioned constraints, can be formulated as:

$$\begin{aligned} \min_{\boldsymbol{q},\dot{\boldsymbol{q}}\in\mathbb{R}^m} \quad & -\dot{\boldsymbol{q}}^{\mathrm{T}}(\boldsymbol{J}_T\diamondsuit h)\boldsymbol{\psi} \\ \text{s.t.} \quad & \left(I_n \otimes \boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)\boldsymbol{\psi} = \mathrm{vec}\,(I_n) \\ & \boldsymbol{J}_T\dot{\boldsymbol{q}} = \boldsymbol{v}_d \\ & \boldsymbol{J}_{rcm}\dot{\boldsymbol{q}} = \boldsymbol{v}_{rcm} \\ & \dot{\boldsymbol{q}} \in \Omega_{\boldsymbol{q},\dot{\boldsymbol{q}}} \end{aligned} \tag{19}$$

C. Reformulation as a constrained quadratic program

There exist the joint angle drift because of the loss of explicit information on $\boldsymbol{X}_d$ and $\boldsymbol{E}_{rcm}$. Therefore, we design the feedback controller to restrict the movement of robot for end-effector and RCM velocity constraint in (4),

$$\begin{aligned} \boldsymbol{v}_d &= -k_1\left(f_1(\boldsymbol{q}) - \boldsymbol{X}_d\right) + \dot{\boldsymbol{X}}_d \\ \boldsymbol{v}_{rcm} &= -k_2\left(f_2(\boldsymbol{q}) - \boldsymbol{E}_{rcm}\right) + \dot{\boldsymbol{E}}_{rcm} \end{aligned} \tag{20}$$

where k_1, k_2 are positive feedback gain.

Furthermore, in order to guarantee that the problem is convex and compliant with the formulation proposed in [11], the objective function including three tasks is defined as,

$$\begin{aligned} &f(\dot{\boldsymbol{q}}, \boldsymbol{\psi}) = -h_0\dot{\boldsymbol{q}}^{\mathrm{T}}(\boldsymbol{J}_T\diamondsuit h)\boldsymbol{\psi} + \tfrac{h_1}{2}\left\|\dot{\boldsymbol{q}}\right\|^2 + \tfrac{h_2}{2}\left\|\boldsymbol{J}_T\dot{\boldsymbol{q}} - \boldsymbol{v}_d\right\|^2 \\ &+\tfrac{h_3}{2}\left\|\boldsymbol{J}_{rcm}\dot{\boldsymbol{q}} - \boldsymbol{v}_{rcm}\right\|^2 + \tfrac{h_4}{2}\left\|\left(I_n \otimes \boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)\boldsymbol{\psi} - \mathrm{vec}\,(I_n)\right\|^2 \end{aligned} \tag{21}$$

where h_0, h_1, h_2, h_3, and h_4 are positive constants.

The optimization problem in (19) can be reformulated as:

$$\begin{aligned} \min_{\boldsymbol{q},\dot{\boldsymbol{q}}\in\mathbb{R}^m} \quad & f(\dot{\boldsymbol{q}}, \boldsymbol{\psi}) \\ \text{s.t.} \quad & \left(I_n \otimes \boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)\boldsymbol{\psi} = \mathrm{vec}\,(I_n) \\ & \boldsymbol{J}_T\dot{\boldsymbol{q}} = \boldsymbol{v}_d \\ & \boldsymbol{J}_{rcm}\dot{\boldsymbol{q}} = \boldsymbol{v}_{rcm} \\ & \dot{\boldsymbol{q}} \in \Omega_{\boldsymbol{q},\dot{\boldsymbol{q}}} \end{aligned} \tag{22}$$

D. Varying-Parameter Hierarchical Optimization Scheme

To obtain the solution of QP problem in (22), the Lagrange function is defined as,

$$\begin{aligned} &L(\dot{\boldsymbol{q}}, \boldsymbol{\psi}, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \boldsymbol{\lambda}_3) = -h_0\dot{\boldsymbol{q}}^{\mathrm{T}}(\boldsymbol{J}_T\diamondsuit h)\boldsymbol{\psi} + \tfrac{h_1}{2}\left\|\dot{\boldsymbol{q}}\right\|^2 + \\ &\tfrac{h_2}{2}\left\|\boldsymbol{J}_T\dot{\boldsymbol{q}} - \boldsymbol{v}_d\right\|^2 + \tfrac{h_3}{2}\left\|\boldsymbol{J}_{rcm}\dot{\boldsymbol{q}} - \boldsymbol{v}_{rcm}\right\|^2 + \\ &\tfrac{h_4}{2}\left\|\left(I_n \otimes \boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)\boldsymbol{\psi} - \mathrm{vec}\,(I_n)\right\|^2 + \boldsymbol{\lambda}_1^{\mathrm{T}}\left(\boldsymbol{v}_d - \boldsymbol{J}_T\dot{\boldsymbol{q}}\right) + \\ &\boldsymbol{\lambda}_2^{\mathrm{T}}\left(\boldsymbol{v}_{rcm} - \boldsymbol{J}_{rcm}\dot{\boldsymbol{q}}\right) + \boldsymbol{\lambda}_3^{\mathrm{T}}\left(\mathrm{vec}\,(I_n) - \left(I_n \otimes \boldsymbol{J}_T\boldsymbol{J}_T^{\mathrm{T}}\right)\boldsymbol{\psi}\right) \end{aligned} \tag{23}$$

where $\boldsymbol{\lambda}_1 \in \mathbb{R}^n$, $\boldsymbol{\lambda}_2 \in \mathbb{R}^n$, $\boldsymbol{\lambda}_3 \in \mathbb{R}^{n^2}$

Authorized licensed use limited to: Technische Universitaet Muenchen. Downloaded on May 11,2022 at 13:13:15 UTC from IEEE Xplore. Restrictions apply.

We define $\partial L = [\partial L/\partial \dot{\boldsymbol{q}}, \partial L/\partial \boldsymbol{\psi}, \partial L/\partial \boldsymbol{\lambda}_1, \partial L/\partial \boldsymbol{\lambda}_2, \partial L/\partial \boldsymbol{\lambda}_3]$.

$$
\partial L = \begin{cases} \partial L/\partial \dot{\boldsymbol{q}} = -h_0 \left(\boldsymbol{J}_T \diamondsuit h\right) \boldsymbol{\psi} + h_1 \dot{\boldsymbol{q}} + h_2 \boldsymbol{J}_T^{\mathrm{T}} \left(\boldsymbol{J}_T \dot{\boldsymbol{q}} - \boldsymbol{v}_d\right) \\ \quad + h_3 \boldsymbol{J}_{rcm}^{\mathrm{T}} \left(\boldsymbol{J}_{rcm} \dot{\boldsymbol{q}} - \boldsymbol{v}_{rcm}\right) - \boldsymbol{J}_T^{\mathrm{T}} \boldsymbol{\lambda}_1 - \boldsymbol{J}_{rcm}^{\mathrm{T}} \boldsymbol{\lambda}_2 \\ \partial L/\partial \boldsymbol{\psi} = -h_0 (\boldsymbol{J}_T \diamondsuit h)^{\mathrm{T}} \dot{\boldsymbol{q}} - \left(I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}\right) \boldsymbol{\lambda}_3 \\ \quad + h_4 \left(\left(I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}\right) \left(\left(I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}\right) \boldsymbol{\psi} - \mathrm{vec}\left(I_n\right)\right)\right) \\ \partial L/\partial \boldsymbol{\lambda}_1 = -\left(\boldsymbol{J}_T \dot{\boldsymbol{q}} - \boldsymbol{v}_d\right) \\ \partial L/\partial \boldsymbol{\lambda}_2 = -\left(\boldsymbol{J}_{rcm} \dot{\boldsymbol{q}} - \boldsymbol{v}_{rcm}\right) \\ \partial L/\partial \boldsymbol{\lambda}_3 = -\left(\left(I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}\right) \boldsymbol{\psi} - \mathrm{vec}\left(I_n\right)\right) \end{cases} \tag{24}
$$

If ∂L are continuous, the optimal solution satisfy $\partial L = 0$.

To solve the problem in (22), the VPRNN is proposed. Firstly, the decision vector $\boldsymbol{z}$ is defined as: $\boldsymbol{z} = [\dot{\boldsymbol{q}}, \boldsymbol{\psi}, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \boldsymbol{\lambda}_3]^{\mathrm{T}}$ ($\boldsymbol{z} \in \mathbb{R}^{m+2n^2+2n}$). Then, the equation in (24) can be rewritten as form of a matrix,

$$
\boldsymbol{W} \boldsymbol{z} = \boldsymbol{l}, \quad \boldsymbol{z} \in \Omega \tag{25}
$$

where $\boldsymbol{W} \in \mathbb{R}^{(m+2n^2+2n)\times(m+2n^2+2n)}$, $\boldsymbol{l} \in \mathbb{R}^{m+2n^2+2n}$,

$$
\boldsymbol{W} = \begin{bmatrix} \boldsymbol{w}_{11} & \boldsymbol{w}_{12} & -\boldsymbol{J}_T^{\mathrm{T}} & -\boldsymbol{J}_{rcm}^{\mathrm{T}} & \boldsymbol{0} \\ \boldsymbol{w}_{21} & \boldsymbol{w}_{22} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{w}_{25} \\ \boldsymbol{J}_T & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{J}_{rcm} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{w}_{52} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}
$$

$$
\boldsymbol{l} = [\boldsymbol{l}_{11}, \boldsymbol{l}_{12}, \boldsymbol{v}_d, \boldsymbol{v}_{rcm}, \mathrm{vec}\left(I_n\right)]^{\mathrm{T}}
$$

where

$$
\begin{aligned} \boldsymbol{w}_{11} &= h_1 + h_2 \boldsymbol{J}_T^{\mathrm{T}} \boldsymbol{J}_T + h_3 \boldsymbol{J}_{rcm}^{\mathrm{T}} \boldsymbol{J}_{rcm}; \\ \boldsymbol{w}_{12} &= -h_0 \left(\boldsymbol{J}_T \diamondsuit h\right); \boldsymbol{w}_{21} = -h_0 (\boldsymbol{J}_T \diamondsuit h)^{\mathrm{T}}; \\ \boldsymbol{w}_{22} &= h_4 \left(I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}\right) \left(I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}\right); \\ \boldsymbol{w}_{25} &= -\left(I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}\right); \boldsymbol{w}_{52} = I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}; \\ \boldsymbol{l}_{11} &= h_2 \boldsymbol{J}_T^{\mathrm{T}} \boldsymbol{v}_d + h_3 \boldsymbol{J}_{rcm}^{\mathrm{T}} \boldsymbol{v}_{rcm}; \\ \boldsymbol{l}_{12} &= h_4 \left(I_n \otimes \boldsymbol{J}_T \boldsymbol{J}_T^{\mathrm{T}}\right) \mathrm{vec}\left(I_n\right). \end{aligned}
$$

To obtain the optimization solution of (25), the error model of neuro-dynamics optimization is defined as,

$$
\boldsymbol{\delta}(t) = \boldsymbol{W}(t) \boldsymbol{z}(t) - \boldsymbol{l}(t) \tag{26}
$$

where $\boldsymbol{\delta}(t) \in \mathbb{R}^{m+2n^2+2n}$. To make the error model in (26) converge to 0, the varying parameter neuro-dynamics optimization scheme is designed as,

$$
\frac{d\boldsymbol{\delta}(t)}{dt} = -\beta \exp(t) \boldsymbol{\phi}(\boldsymbol{\delta}(t)) \tag{27}
$$

where $\beta > 0$ is the constant which can scale the convergence rate. The activation function of (27) is defined as,

$$
\boldsymbol{\phi}(\boldsymbol{\delta}(t)) = \begin{cases} \delta_i^-, & \text{if } \delta_i(t) < \delta_i^- \\ \delta_i, & \text{if } \delta_i^- < \delta_i(t) < \delta_i^+ \\ \delta_i^+, & \text{if } \delta_i(t) > \delta_i^+ \end{cases} \tag{28}
$$

where $\boldsymbol{\delta}_i^-$ and δ_i^+ are lower bound and upper bound of i-the element. Then, substituting (26) into (27), the extend expression of (27) can be rewritten as,

$$
\begin{aligned} \boldsymbol{W}(t) \dot{z}(t) = &-\dot{\boldsymbol{W}}(t) \boldsymbol{z}(t) + \dot{\boldsymbol{l}}(t) \\ &- \beta \exp(t) \phi(\boldsymbol{W}(t) \boldsymbol{z}(t) - \boldsymbol{l}(t)) \end{aligned} \tag{29}
$$

TABLE I

EXPERIMENTAL CONTROLLER PARAMETERS

Controller	Controller parameters
RNN	$\boldsymbol{K}_X = diag[3000, 3000, 3000]$ $\boldsymbol{D}_X = diag[30, 30, 30]$ $\boldsymbol{K}_N = diag[800, 800, 800]$ $\boldsymbol{D}_N = diag[10, 10, 10]$, $\lambda = 0.5$ $\boldsymbol{K}_{N2} = [80, 120, 60, 30, 20, 10, 10]$ $\boldsymbol{K}_{D2} = [6, 10, 5, 3, 2, 1, 1]$ $h_0 = 0.01, h_1 = 0.01, h_2 = 1, h_4 = 1$ $k_1 = 5, \beta = 10^3$
VPRNN	$\boldsymbol{K}_X = diag[3000, 3000, 3000]$ $\boldsymbol{D}_X = diag[30, 30, 30]$ $\boldsymbol{K}_N = diag[800, 800, 800]$ $\boldsymbol{D}_N = diag[10, 10, 10]$, $\boldsymbol{K}_{N2} = [80, 120, 60, 30, 20, 10, 10]$ $\boldsymbol{K}_{D2} = [6, 10, 5, 3, 2, 1, 1]$ $h_0 = 0.1, h_1 = 10, h_2 = 50, h_3 = 30$ $h_4 = 30, k_1 = 5, k_2 = 5, \beta = 10^3$

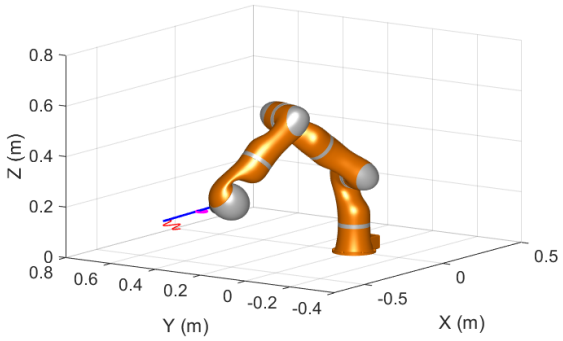

Fig. 2. Demonstration with Kuka Simulator.

IV. EXPERIMENTAL COMPARISON

To evaluate the proposed control scheme, Simulation and experiments are carried out.The magnitude of the Cartesian position error $\boldsymbol{E}_{end}$, the RCM constraint error $\|\boldsymbol{E}_{rcm}\|$ and the manipulability index μ, defined in [7][21], are recorded for analysis. The detailed parameters of the controller are shown in Table I. For DCAC and MOC, the parameters can be found in our previous works [12].

Firstly, as it is shown in Fig. 2, a simulation demonstration using the KUKA simulator is performed to check the feasibility of the proposed methods. Here, the sinusoid task is designed for testing.

To compare the performance of the proposed method with related works, and experimental comparisons, including RNN, Nullspace methods are performed n the same trajectory for comparison. The detailed configuration and development of the system can be found in our previous works [21].

The operative procedure is organized as follows:

1) User 1 uses hands-on control to move the robot and pass through the RCM constraint;
2) Then, the robot autonomously tracks the set trajectory $\boldsymbol{X}_d$ to perform the surgical task, and User 2 is in charge of urgency issues in front of the visual interface.

Authorized licensed use limited to: Technische Universitaet Muenchen. Downloaded on May 11,2022 at 13:13:15 UTC from IEEE Xplore. Restrictions apply.

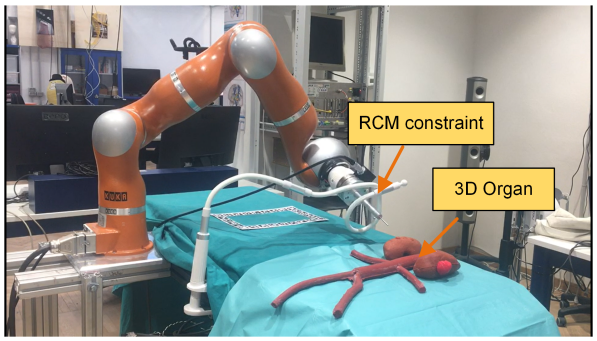

Fig. 3. Experimental setup scene: 1) hands-on control to move the robot manipulator to pass through the RCM constraint (small incision on the patient's body); 2) autonomous tracking is activated to run the application.

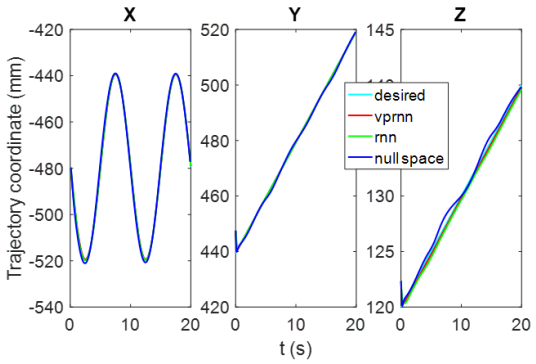

Fig. 4. Comparison results: Trajectory tracking.

Fig. 4 shows the comparison of online performance during the tracking task. Fig. 5 and Fig. 6 show the comparison results of tracking error and RCM error, respectively. Fig. 7 show the comparison results of the manipulability index. Fig. 8 show the joint angles solution with VPRNN methods. From Fig. 5, it can be seen that all the errors of the end-effector are constrained in an acceptable error range within 5mm. From Fig. 6, it can be seen that the RCM error is also in an acceptable error range within 15mm.

Obviously, the proposed VPRNN-based method has an overall promising performance, including end-effector error, RCM error, manipulability index. In addition, the RNN method in [11] shows a better performance than the null space of end-effector error, RCM error, manipulability index. Therefore, we can conclude that VPRNN has the best ability to guarantee task tracking, RCM constraint, manipulability, and has the fastest convergence rate.

V. Discussion and Conclusion

This paper addresses varying parameters recurrent neural network (VPRNN) based on hierarchical control of a 7-DoF robot manipulator for Robot-Assisted Minimally Invasive Surgery to achieve task tracking, Remote Center of Motion (RCM) and manipulability optimization at the same time. In order to efficiently accomplish the Cartesian compliance control RCM constraint, surgical task, and manipulability optimization, a hierarchical operational space formulation is

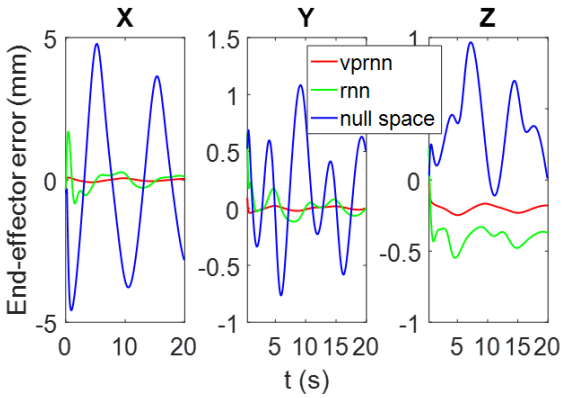

Fig. 5. Comparison results: End-effector error $\boldsymbol{E}_{end}$.

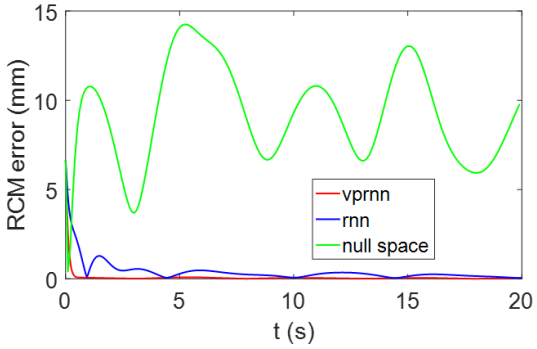

Fig. 6. Comparison results: RCM error $\|\boldsymbol{E}_{rcm}\|$.

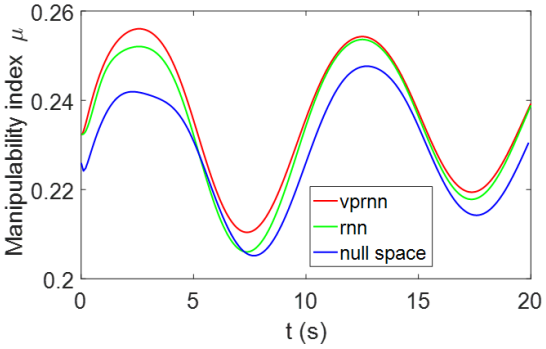

Fig. 7. Comparison results: Manipulability index μ.

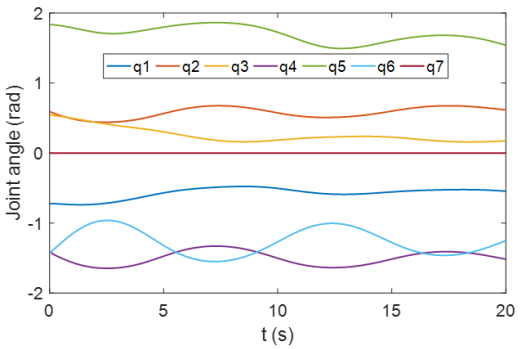

Fig. 8. Joint angles solution using VPRNN.

Authorized licensed use limited to: Technische Universitaet Muenchen. Downloaded on May 11,2022 at 13:13:15 UTC from IEEE Xplore. Restrictions apply.

investigated. The new optimization problem is the real-time resolution for given tasks and has an excellent convergence performance even in the random initial position. Finally, in order to evaluate the accuracy of the proposed scheme, experimental evaluation has been discussed on virtual surgical tasks. Several remarks connected to [12] here detailed explaining that the recommended control scheme not only ensures the RCM constraint facing the auto-tracking phase but also develops the robot manipulability.

VI. Appendix

A. Convergence Analysis

Theorem 1: Considering the optimization problem in (22), if there exist the optimal solution $\boldsymbol{z}^*$ when the activation function in 28) is mapped to error model of varying parameter neural network, the decision variable $\boldsymbol{z}$ is defined as: $\boldsymbol{z} = [\dot{\boldsymbol{q}}, \boldsymbol{\psi}, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \boldsymbol{\lambda}_3]^{\mathrm{T}}$ ($\boldsymbol{z} \in \mathbb{R}^{m+2n^2+2n}$) globally converges to the optimal solution $\boldsymbol{z}^* = [\dot{\boldsymbol{q}}^*, \boldsymbol{\psi}^*, \boldsymbol{\lambda}_1^*, \boldsymbol{\lambda}_2^*, \boldsymbol{\lambda}_3^*]^{\mathrm{T}}$ ($\boldsymbol{z} \in \mathbb{R}^{m+2n^2+2n}$) from any initial point.

Proof: The candidate Lyapunov function is defined as,

$$\boldsymbol{V}(t) = \frac{1}{2}\boldsymbol{\delta}^T\boldsymbol{\delta} \tag{30}$$

The time derivative of $\boldsymbol{V}(t)$ is expressed as,

$$\dot{V}(t) = \frac{d\boldsymbol{V}(t)}{dt} = \boldsymbol{\delta}^T(t)\,\dot{\boldsymbol{\delta}}(t) \tag{31}$$

Substituting (28) into (31),

$$\begin{aligned}\dot{\boldsymbol{V}}(t) &= -\beta\exp(t)\,\boldsymbol{\delta}^T(t)\,\boldsymbol{\phi}(\boldsymbol{\delta}(t)) \\ &= -\beta\exp(t)\sum_{i=1}^{N}\boldsymbol{\delta}_i(t)\,\boldsymbol{\phi}(\boldsymbol{\delta}_i(t))\end{aligned} \tag{32}$$

As mentioned in (28), the function $\boldsymbol{\phi}(\boldsymbol{\delta}(t))$ is the monotone nondecreasing, thus we have,

$$\delta_i(t)\,\boldsymbol{\phi}(\delta_i(t)) = \begin{cases} > 0, & \text{if } \delta_i(t) \neq 0 \\ = 0, & \text{if } \delta_i(t) = 0 \end{cases} \tag{33}$$

Finally, the time derivative $V(t)$ is obtained as,

$$\dot{\boldsymbol{V}}(t) = \begin{cases} < 0, & \text{if } \delta_i(t) \neq 0 \\ = 0, & \text{if } \delta_i(t) = 0 \end{cases} \tag{34}$$

From the (34), it can conclude that if and only if $\delta_i(t) = 0$, $\dot{\boldsymbol{V}} = 0$; otherwise $\dot{\boldsymbol{V}} < 0$. The proof is finished. ■

References

[1] A. M. Okamura, "Haptic feedback in robot-assisted minimally invasive surgery," *Current opinion in urology*, vol. 19, no. 1, p. 102, 2009.

[2] A. R. Lanfranco, A. E. Castellanos, J. P. Desai, and W. C. Meyers, "Robotic surgery: a current perspective," *Annals of surgery*, vol. 239, no. 1, p. 14, 2004.

[3] K. A. Nichols and A. M. Okamura, "A framework for multilateral manipulation in surgical tasks," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 68–77, 2015.

[4] N. Enayati, A. M. Okamura, A. Mariani, E. Pellegrini, M. M. Coad, G. Ferrigno, and E. De Momi, "Robotic assistance-as-needed for enhanced visuomotor learning in surgical robotics training: An experimental study," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6631–6636.

[5] N. Vahrenkamp, T. Asfour, G. Metta, G. Sandini, and R. Dillmann, "Manipulability analysis." in *Humanoids*, 2012, pp. 568–573.

[6] B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," *Journal of intelligent and robotic systems*, vol. 3, no. 3, pp. 201–212, 1990.

[7] H. Su, J. Sandoval, M. Makhdoomi, G. Ferrigno, and E. De Momi, "Safety-enhanced human-robot interaction control of redundant robot for teleoperated minimally invasive surgery," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6611–6616.

[8] N. Aghakhani, M. Geravand, N. Shahriari, M. Vendittelli, and G. Oriolo, "Task control with remote center of motion constraint for minimally invasive robotic surgery," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 5807–5812.

[9] T. M. Sobh and D. Y. Toundykov, "Optimizing the tasks at hand [robotic manipulators]," *IEEE robotics & automation magazine*, vol. 11, no. 2, pp. 78–85, 2004.

[10] H. Su, G. Ferrigno, and E. De Momi, "Adaptive decoupling control of a serial redundant robot for teleoperated minimally invasive surgery," in *IEEE ICRA Workshop on Supervised Autonomy in Surgical Robotics*, 2018.

[11] L. Jin, S. Li, H. M. La, and X. Luo, "Manipulability optimization of redundant manipulators using dynamic neural networks," *IEEE Transactions on Industrial Electronics*, 2017.

[12] H. Su, S. Li, J. Manivannan, L. Bascetta, G. Ferrigno, and E. D. Momi, "Manipulability optimization control of a serial redundant robot for robot-assisted minimally invasive surgery," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 1323–1328.

[13] Z. Li, T. Zhao, F. Chen, Y. Hu, C.-Y. Su, and T. Fukuda, "Reinforcement learning of manipulation and grasping using dynamical movement primitives for a humanoidlike mobile manipulator," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 1, pp. 121–131, 2017.

[14] H. Su, J. Sandoval, P. Vieyres, G. Poisson, G. Ferrigno, and E. De Momi, "Safety-enhanced collaborative framework for teleoperated minimally invasive surgery using a 7-dof torque-controlled robot," *International Journal of Control, Automation and Systems*, vol. 16, no. 6, pp. 2915–2923, 2018.

[15] Y. Hu, H. Su, L. Zhang, S. Miao, G. Chen, and A. Knoll, "Nonlinear model predictive control for mobile robot using varying-parameter convergent differential neural network," *Robotics*, vol. 8, no. 3, p. 64, 2019.

[16] J. Sandoval, G. Poisson, and P. Vieyres, "Improved dynamic formulation for decoupled cartesian admittance control and rcm constraint," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1124–1129.

[17] Y. Hu, G. Chen, X. Ning, J. Dong, S. Liu, and A. Knoll, "Mobile robot learning from human demonstrations with nonlinear model predictive control," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5057–5062.

[18] A. Dietrich, C. Ott, and J. Park, "The hierarchical operational space formulation: Stability analysis for the regulation case," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1120–1127, 2018.

[19] B. Lacevic, P. Rocco, and A. M. Zanchettin, "Safety assessment and control of robotic manipulators using danger field," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1257–1270, 2013.

[20] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 882–893, 2015.

[21] H. Su, C. Yang, G. Ferrigno, and E. De Momi, "Improved human-robot collaborative control of redundant robot for teleoperated minimally invasive surgery," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1447–1453, 2019.

Authorized licensed use limited to: Technische Universitaet Muenchen. Downloaded on May 11,2022 at 13:13:15 UTC from IEEE Xplore. Restrictions apply.