

Haptic-guided shared control for needle grasping optimization in minimally invasive robotic surgery

Mario Selvaggio, Amir M. Ghalamzan E., Rocco Moccia, Fanny Ficuciello and Bruno Siciliano

Abstract—During suturing tasks performed with minimally invasive surgical robots, configuration singularities and joint limits often force surgeons to interrupt the task and re-grasp the needle using dual-arm movements. This yields an increased operator’s cognitive load, time-to-completion and performance degradation. In this paper, we propose a haptic-guided shared control method for grasping the needle with the Patient Side Manipulator (PSM) of the da Vinci robot avoiding such issues. We suggest a cost function consisting of (i) the distance from robot joint limits and (ii) the task-oriented manipulability along the suturing trajectory. Evaluating the cost and its gradient on the needle grasping manifold allows us to obtain the optimal grasping pose for joint-limit and singularity free robot movements during suturing. We compute force cues and display them through the Master Tool Manipulator (MTM) to guide the surgeon towards the optimal grasp. As such, our system helps the operator to choose a grasping configuration that allows the robot to avoid joint limits and singularities during post-grasp suturing movements. We show the effectiveness of the proposed haptic-guided shared control method during suturing using both simulated and real experiments. The results illustrate that our approach significantly improves the performance in terms of needle re-grasping.

I. INTRODUCTION

Autonomous control methods are not yet sufficiently trusted in robotic surgery because of safety-critical and high-consequence tasks to perform. On the other hand, remote teleoperation of surgical robotic systems imposes extreme cognitive loading to the human operator, causing severe fatigue and, consequently, a progressive degeneration in performance. However, patients are highly conservative and trust more a surgeon who remotely teleoperates the robotic system within the patient’s body through a master console as in the da Vinci robot (see Fig. 1).

Haptic-guided shared control is a promising approach to reduce the human operator’s cognitive load during teleoperation. This improves performance by providing haptic cues that guide the operator in carrying out safe and effective actions. For instance, Ghalamzan *et al.* showed in [1] that haptic cues, guiding the human operator based on a predictive

Mario Selvaggio, Rocco Moccia, Fanny Ficuciello and Bruno Siciliano are with Università degli Studi di Napoli Federico II, Italy. Amir Ghalamzan is with the University of Lincoln, UK. This research was partially supported by the IEEE RAS Technical Committee on Haptics under the “Innovation in haptics” research programme, partially funded by the EC Seventh Framework Programme (FP7) within RoDyMan project 320992, partially within the Italian National Project BARTOLO (POR FESR 2014-2020). This work forms part of the UK National Centre for Nuclear Robotics (NCRN) initiative, funded by EPSRC EP/R02572X/1. It is part of EPSRC flexible partnership funding on ‘Haptic-guided shared control of mobile manipulation task’. This work is also partly funded by IUK #104587 which is part of the RASberry (Robotic and Autonomous Systems for Berry Production) programme.

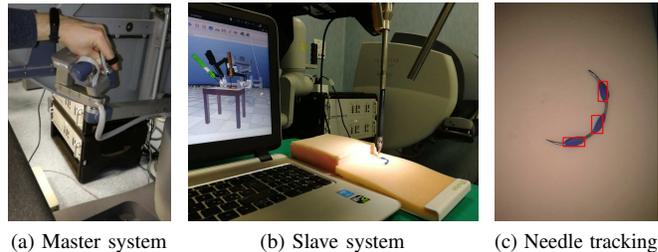


Fig. 1. Experimental setup comprising of master (1a) and slave (1b) systems of da Vinci Research Kit. An operator commands the slave tool through the master device to grasp a needle. Figure 1c shows the tracking system to retrieve the needle pose.

cost of avoiding singularities during the teleoperation task, drastically improve the manipulation performance.

In da Vinci-like surgical systems, while tremor filtering and motion scaling enhance the movements precision, the robot mechanical structure also imposes some constraints. In particular, joint limits and singularities are common issues that, for example, force the surgeon to re-grasp the needle in the middle of the suturing task. As such, surgeons waste a considerable amount of time re-configuring the robot during interventions despite the substantial training they undergo. Fontanelli *et al.* in [2] noticed that the suturing task, performed by a surgeon using a da Vinci robot, requires several hand-off movements using both arms to re-orient the needle before each stitch. This results in longer procedural time, increased fatigue, higher cognitive loading and degenerating performance.

This paper proposes a haptic-guided shared control system that assists the surgeon in avoiding such problems during suturing tasks performed with a da Vinci robot. The proposed approach ‘guides’ the operator during the reach-to-grasp phase to choose a needle grasping configuration that results in neither joint limits nor singularities in the course of suturing. The chosen grasping pose allows starting the suturing procedure immediately with no need for re-configurations during post-grasp manipulative movements, *i.e.* over the course of the suturing trajectory. While the haptic-guided shared control system informs the user about the optimal grasping configuration, it allows the operator to ultimately choose the grasp making a trade-off between the computed cost and other non-modeled objectives.

The rest of this paper is structured as follows. Section II describes the related work in robotic grasping and manipulation, needle grasping and haptic-guided shared control fields. Section III formulates the problem. Needle grasping optimization and haptic guidance are described in Section IV and V, respectively. Section VI presents simulations and

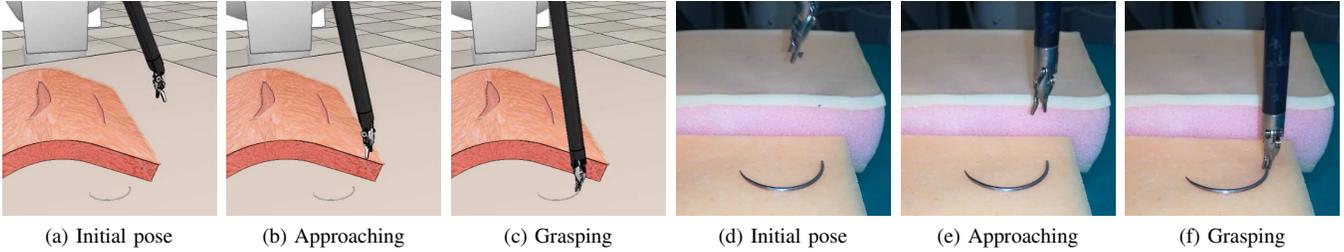


Fig. 2. Grasping a needle using the PSM in a simulated (2a, 2b and 2c) and a real (2d, 2e and 2f) environment: our proposed haptic-guided shared control system generates force cues guiding the operator to choose a grasping configuration yielding neither joint limits nor singularities during suturing movements.

experiments performed to evaluate the developed system. Section VII concludes the paper.

II. RELATED WORK

Performing a successful robotic manipulation task involves: robot approaching an object (g_0 : reach-to-grasp); making stable contacts on the object surface (g_1 : grasping); and moving the object to perform the task (g_2 : post-grasp). Figures. 2a-2b-2c and 2d-2e-2f, illustrate g_0 and g_1 phases for grasping a needle in simulated and real experiments, respectively. State-of-the-art approaches are able to efficiently computing stable grasp configurations (g_1) from a point cloud of a scene with high success rate (*e.g.* using probabilistic learning algorithms [3] or artificial neural networks [4], [5]).

In the robotic surgery domain, D’Ettorre *et al.* proposed in [6] a vision-guided method for autonomous needle grasping which allows the surgeon to immediately start suturing. Nevertheless, the authors only focused on planning the success of g_0 and g_1 . For manipulation tasks, planning g_1 and g_2 cannot be performed in isolation as a chosen grasping configuration may lead to encounter system constraints during g_2 : although the obtained needle grasping yields stable contacts, it may make it impossible for the robot arm to deliver the entire intended suturing movement.

Autonomously performing $g_0 - g_1 - g_2$ is not sufficiently trusted by conservative industries in safety-critical tasks, such as robotic surgery. However, there exist efficient approaches to autonomously perform g_2 . For instance, motion planning algorithms, such as learning from demonstration [7], generalize demonstrated motions to new knot tying examples using a da Vinci robot [8]. The generalized motions are known before $g_0 - g_1$, thus allowing the computation of limitations the robot will face during g_2 given a certain grasp. Some g_2 objectives relevant to $g_0 - g_1$ may include the object affordance [9], mass distribution [10] and collisions with the environment [11]. In contrast, a human operator is not able to foresee these issues during the future movements resulting in several cycles of (i) planning, (ii) executing part of the planned motions, (iii) facing a limitation and (iv) re-planning.

By inspecting the JIGSAW dataset [12], Fontanelli *et al.* noted that the occurrence of hand-off movements during surgical sub-tasks is significant and yields an increased procedural time [2]. They proposed to use an additional DoF in the surgical robotic tool allowing in-hand needle re-orientation. Sen *et al.* proposed in [13] a mechanical device designed to align and hold the needle in a known orientation.

The authors aim was to improve needle pose tracking capabilities of a computer vision software for automating multi-throw surgical suturing. However, these approaches require substantial modification of the robot mechanical structure that are not always practically feasible.

On the other hand, an operator can be successfully guided towards an optimal grasping configuration. To this end, haptic guidance has been used to avoid constraints during g_0 [14], [15] and g_2 [1]. These works reported significant performance improvement compared to direct teleoperation. However, they have been only used to grasp and manipulate simple objects whereas they do not account for the object shape. In this case, force cues, which continuously attract the operator’s towards optimal grasping poses, may have non-intuitive components pushing away from the object shape.

Liu *et al.* presented in [16], [17] an offline optimization-based solution to needle grasping and robotic instruments entry ports selection. An exhaustive search method finds the optimal needle grasping and entry ports poses. However, this method cannot be utilized for real time implementation of haptic-guided shared control systems due to the high computation time.

We propose a haptic-guided shared control method for suturing tasks with a da Vinci robot to assist the operator during g_0 . We use a cost function which is comprised of two terms: (i) joint limits and (ii) task-oriented velocity (TOV) manipulability¹. In contrast to previous works [1], [14], we assume the shape of the object to be grasped, *i.e.* the needle, is known. To eliminate the force cues non-intuitive components, we compute the cost and project its gradient onto the space of feasible grasping pose (*grasping manifold*). Hence, the non-intuitive components are eliminated and force cues always attract the operator’s hand towards the needle shape.

III. PROBLEM FORMULATION

Let us consider the Patient Side Manipulator (PSM) of a da Vinci surgical robot performing a suturing task (Fig. 1b). The PSM is a 6-DoFs manipulator with $\mathbf{q}_s \in \mathbb{R}^6$ being the set of its generalized coordinates (see [18] for a detailed PSM kinematic description). An operator commands the PSM through the da Vinci Master Tool Manipulator (MTM)(Fig. 1a) to grasp a needle (Fig. 1c) and performing a suturing task. Stitching trajectories can be computed based on the desired

¹In contrast, [17] used the classical manipulability measure. The superiority of TOV over classical manipulability in manipulation tasks is discussed in [1].

needle insertion and extraction points and include approaching the insertion point on a tissue, piercing, reaching the desired extraction point, piercing and pushing the needle out of the tissue [19]. We assume that the needle trajectory necessary to perform the stitching is given (Fig. 3) by either a planning algorithm [20] or learning from demonstration [21].

We denote by $\mathcal{F}_n : \{O_n; x_n, y_n, z_n\}$ a local frame attached to the needle and with $\mathcal{F}_r : \{O_r; x_r, y_r, z_r\}$ an inertial reference frame. \mathcal{F}_n can be expressed in \mathcal{F}_r through the transformation matrix² ${}^r\mathbf{T}_n \in \text{SE}(3)$

$${}^r\mathbf{T}_n(t) = \begin{bmatrix} \mathbf{R}_{3 \times 3}(t) & \mathbf{p}_{3 \times 1}(t) \\ \mathbf{0}_{1 \times 3} & \mathbf{1} \end{bmatrix}, \quad (1)$$

where, as usual, $\mathbf{p} \in \mathbb{R}^3$ and $\mathbf{R} \in \text{SO}(3)$ are the position and orientation of any point in Cartesian space. We denote by $\mathcal{F}_t : \{O_t; x_t, y_t, z_t\}$ a reference frame attached to the needle tip. The trajectory to be followed by the needle implies that \mathcal{F}_t matches a sequence of planned stitching poses (see Fig. 3)

$$\mathcal{F}_p(t) = \zeta(t) \quad t \in [0, T], \quad (2)$$

where ζ is the stitching trajectory, t denotes the time and T is the total time to complete the stitching. In other words, $\mathcal{F}_p(t)$ determines a complete object pose (position and orientation) at every time t along the trajectory ζ .

Let us denote a local frame attached to the PSM end-effector by $\mathcal{F}_e : \{O_e; x_e, y_e, z_e\}$. Let also $\mathcal{F}_g : \{O_g; x_g, y_g, z_g\}$ be a local reference frame attached to the needle shape which serves as a grasping pose candidate for the robot end-effector. Since the needle is not deformable and the PSM end-effector forms stable contacts on the needle surface, the PSM end-effector pose \mathcal{F}_g becomes equal to \mathcal{F}_e once contacts are made. As such, \mathcal{F}_g can be fully expressed at all time by a fixed transformation matrix, namely ${}^n\mathbf{T}_g$, w.r.t. needle local frame \mathcal{F}_n . The PSM end-effector trajectory during suturing can be derived given the planned needle trajectory ζ in (2) as follows

$${}^r\mathbf{T}_g(t) = \{{}^r\mathbf{T}_n(t) {}^n\mathbf{T}_g : 0 \leq t \leq T\}. \quad (3)$$

Finally, the post-grasp joint configuration trajectory corresponding to a given grasping ${}^r\mathbf{T}_e(t) = {}^r\mathbf{T}_g(t)$, can be computed as follows

$$\hat{\mathbf{q}}_g(t) = \text{IK}({}^r\mathbf{T}_g(t)) \quad (4)$$

where $\text{IK}(\cdot)$ is the PSM inverse kinematics function which computes the joint space trajectory $\hat{\mathbf{q}}_g(t)$ corresponding to the grasping frame trajectory $\mathcal{F}_g(t)$. The problem is to find ${}^n\mathbf{T}_g$ in (3) and thus $\hat{\mathbf{q}}_g(t)$ in (4) which yields no singularities or joint limits along ζ in (2).

IV. NEEDLE GRASPING

To successfully perform a suturing task without hands-off movements, we guide the human operator towards a grasping pose which yields no singularities or joint limits during the post-grasping suturing trajectory. In contrast to previous studies [1], [14], here we know the geometrical shape of the

²In general, ${}^b\mathbf{T}_a \in \text{SE}(3)$ denotes a homogeneous transformation matrix representing the pose of local frame a into local frame b .

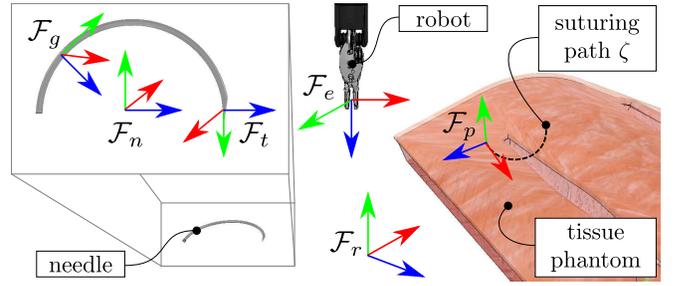


Fig. 3. A schematic surgical setting during the suturing task: \mathcal{F}_r , \mathcal{F}_e , \mathcal{F}_n and \mathcal{F}_t denote the inertial, the end-effector, the needle center and the needle tip reference frames, respectively. Furthermore, \mathcal{F}_g represents a frame attached to the needle corresponding to a generic desired grasping pose for the robot end-effector. The needle tip trajectory (dashed black curve) from the insertion point (rightmost) to the exit point (leftmost) is assumed to be given by motion planning or learning by demonstration techniques. \mathcal{F}_p denotes a generic needle-tip trajectory pose, *i.e.* position and orientation.

object to be grasped. To obtain consistent results, we should account for the needle geometry in the problem formulation and optimize in the space of feasible grasping poses. In Section IV-A, we will discuss parameterizing needle grasping poses while in Section IV-B we propose the joint limits and singularities cost function and the grasping manifold optimization method.

A. Grasp manifold parametrization

To grasp a curved needle with the PSM gripper, it is necessary to align the y_e -axis of \mathcal{F}_e (green axis in Fig. 3)³ with the needle tangent (this results in a 2-DoFs rotational constraint). In addition, grasping requires the gripper position to belong to the needle shape (additional 2-DoFs position constraint) [17]. The remaining subspace of possible grasping configurations (2-DoFs) can be conveniently parametrized using the needle curvilinear abscissa $n \in [0, n^*]$ and the angle around the needle tangent $\alpha \in [\alpha^-, \alpha^+]$ (see Fig. 4). We denote by $\mathbf{z} = [n, \alpha]^T$ the vector identifying any point in the considered subspace \mathcal{Z} denoting the needle grasping manifold (locally $\mathcal{Z} \subseteq \mathbb{R}^2$). Hence, we can explicitly express \mathcal{F}_e as function of \mathbf{z} at grasping through (3) with ${}^n\mathbf{T}_g$ being

$$\begin{aligned} {}^n\mathbf{T}_g(\mathbf{z}) &= \mathbf{T}_p(n)\mathbf{T}_r(\alpha) = \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & s(n\pi) & c(n\pi) & -rc(n\pi) \\ 0 & -c(n\pi) & s(n\pi) & -rs(n\pi) \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c(\alpha) & 0 & s(\alpha) & 0 \\ 0 & 1 & 0 & 0 \\ -s(\alpha) & 0 & c(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (5)$$

where $c(\cdot)$ and $s(\cdot)$ are shorthand notations for the $\cos(\cdot)$ and $\sin(\cdot)$ functions, r is the needle radius and n , α have been introduced above. Denoting with $\dot{\mathbf{x}}_e = [\dot{\mathbf{p}}^T, \dot{\boldsymbol{\omega}}^T]^T$ the stacked vector of the PSM linear ($\dot{\mathbf{p}} \in \mathbb{R}^3$) and angular ($\dot{\boldsymbol{\omega}} \in \mathbb{R}^3$) velocities at grasping it yields

$${}^r\dot{\mathbf{x}}_e = {}^r\bar{\mathbf{R}}_n {}^n\dot{\mathbf{x}}_g = {}^r\bar{\mathbf{R}}_n \mathbf{J}_g(\mathbf{z}) \dot{\mathbf{z}}, \quad {}^r\bar{\mathbf{R}}_n = \begin{bmatrix} {}^r\mathbf{R}_n & \mathbf{O} \\ \mathbf{O} & {}^r\mathbf{R}_n \end{bmatrix} \quad (6)$$

where ${}^r\bar{\mathbf{R}}_n \in \mathbb{R}^{6 \times 6}$ transforms the twist $\dot{\mathbf{x}}_g$ from \mathcal{F}_n to \mathcal{F}_r , $\mathbf{J}_g(\mathbf{z}) \in \mathbb{R}^{6 \times 2}$ is the *grasp Jacobian* specific to the object

³Subscript * refers to the end-effector during reach-to-grasp whereas g refers to the end-effector after making stable contacts between the end-effector and the needle.

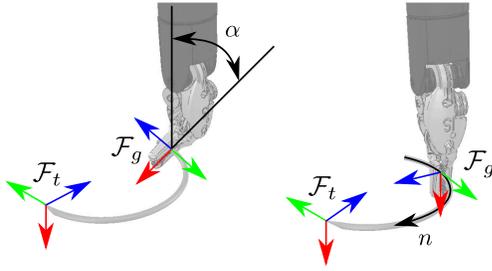


Fig. 4. Grasp parameterization: grasping configurations can be parametrized by α , representing the angle around the needle tangent, and n , representing the needle curvilinear abscissa (see also [17], [19]).

shape and the choice of grasping parameters, and mapping the grasping manifold velocities \dot{z} into ${}^n\dot{x}_g$. Combining the PSM differential forward kinematics, *i.e.* ${}^r\dot{x}_e = \mathbf{J}_s(\mathbf{q}_s)\dot{\mathbf{q}}_s$ (where \mathbf{J}_s is the conventional PSM geometric Jacobian), and (5)-(6) yields

$$\dot{\mathbf{q}}_s = \mathbf{J}_s^\dagger(\mathbf{q}_s) {}^r\bar{\mathbf{R}}_n \mathbf{J}_g(z) \dot{z}, \quad (7)$$

where \mathbf{J}_s^\dagger denotes the usual \mathbf{J}_s Moore-Penrose pseudoinverse. Equation (7) will be exploited in the following Section to optimize the needle grasping pose.

B. Optimal grasping pose selection

In this work, we are interested in avoiding joint limits and singularities during post-grasp movements, *i.e.* along the suturing trajectory. As such, we define a cost function accounting for joint limits, *i.e.*

$$h_j(\hat{\mathbf{q}}_g(t)) = \sum_{i=1}^n \frac{1}{\lambda} \frac{(\hat{\mathbf{q}}_{g,i}^+ - \hat{\mathbf{q}}_{g,i}^-)^2}{(\hat{\mathbf{q}}_{g,i}^+ - \hat{\mathbf{q}}_{g,i}(t))(\hat{\mathbf{q}}_{g,i}(t) - \hat{\mathbf{q}}_{g,i}^-)}, \quad (8)$$

and another accounting for task-oriented manipulability, *i.e.*

$$h_s(\hat{\mathbf{q}}_g(t)) = {}^r\hat{\mathbf{x}}_e^T (\mathbf{J}_s(\hat{\mathbf{q}}_g(t)) \mathbf{J}_s(\hat{\mathbf{q}}_g(t))^T)^{-1} {}^r\hat{\mathbf{x}}_e, \quad (9)$$

where $\lambda \in \mathbb{R}^+$ is a constant scalar, $\hat{\mathbf{q}}_{g,i}(t)$ is the i^{th} joint coordinate at time t , $\hat{\mathbf{q}}_{g,i}^+$ and $\hat{\mathbf{q}}_{g,i}^-$ are its corresponding upper and lower limits, respectively, and ${}^r\hat{\mathbf{x}}_e = {}^r\dot{\mathbf{x}}_e / \|{}^r\dot{\mathbf{x}}_e\|$ denotes the velocity direction along which the manipulability is measured. Thus, the total cost is (omitting t dependence)

$$h(\hat{\mathbf{q}}_g) = h_j(\hat{\mathbf{q}}_g) + h_s(\hat{\mathbf{q}}_g). \quad (10)$$

We parametrize the single stitch suturing trajectory (given in (2)) with $s \in [0, s^*]$ obtaining ${}^r\mathbf{T}_n(s)$. Substituting this in (3) and using (5) yields ${}^r\mathbf{T}_g(s, \mathbf{z}) = {}^r\mathbf{T}_n(s) {}^n\mathbf{T}_g(\mathbf{z})$. Thus, given ${}^r\mathbf{T}_g(s, \mathbf{z})$ we can compute $\hat{\mathbf{q}}_g(s, \mathbf{z})$ through (4). Hence, the cost function evaluated along the trajectory can be expressed as a function of s and \mathbf{z} as follows

$$\mathcal{H}(\mathbf{z}) = \int_0^{s^*} h(\hat{\mathbf{q}}_g(s, \mathbf{z})) ds. \quad (11)$$

Our aim is to find the vector \mathbf{z} that minimizes the cost function in (11). Mathematically, the problem writes as follows

$$\begin{aligned} & \underset{\mathbf{z}}{\text{minimize}} && \mathcal{H}(\mathbf{z}) \\ & \text{subject to} && \alpha^- \leq \alpha \leq \alpha^+, \\ & && 0 \leq n \leq n^* \end{aligned} \quad (12)$$

The problem in (12) is clearly non-convex and may have local minima that can be reached through iterative methods. For our purpose, we use the gradient descent, *i.e.* a first-order iterative optimization algorithm, to find a local minimum of the cost function in (11). At each optimization step we update \mathbf{z} as in the following succession

$$\mathbf{z}_{n+1} = \mathbf{z}_n - \gamma \nabla_{\mathbf{z}} \mathcal{H}, \quad (13)$$

where $\gamma \in \mathbb{R}^+$ represents the step increment and $\nabla_{\mathbf{z}} \mathcal{H}$ the cost function gradient with respect to \mathbf{z} . Convergence to a local minimum can be guaranteed and is not problematic for our scope since we seek only a locally optimized solution. Equation (13) requires the computation of $\nabla_{\mathbf{z}} \mathcal{H}$. Exploiting Leibniz's formulas this can be written as

$$\nabla_{\mathbf{z}} \mathcal{H} = \frac{\partial \mathcal{H}}{\partial \mathbf{z}} = \int_0^{s^*} \frac{\partial h}{\partial \mathbf{z}} ds, \quad (14)$$

and using the chain rule, we can additionally write

$$\frac{\partial h}{\partial \mathbf{z}} = \frac{\partial h}{\partial \mathbf{q}_s} \frac{\partial \mathbf{q}_s}{\partial \mathbf{z}}. \quad (15)$$

where the term $\partial h / \partial \mathbf{q}_s$ is the derivative of the cost function with respect to the generalized coordinates vector of PSM and tells in which joint space direction the cost function increases most. The term $\partial \mathbf{q}_s / \partial \mathbf{z}$ can be computed from (7) as follows

$$\frac{\partial \mathbf{q}_s}{\partial \mathbf{z}} = \mathbf{J}_s^\dagger(\mathbf{q}_s) {}^r\bar{\mathbf{R}}_n \mathbf{J}_g(\mathbf{z}), \quad (16)$$

and depends only upon the robot kinematics and the chosen grasp parametrization. Substituting in (15), it yields

$$\frac{\partial h}{\partial \mathbf{z}} = \frac{\partial h}{\partial \mathbf{q}_s} \mathbf{J}_s^\dagger(\mathbf{q}_s) {}^r\bar{\mathbf{R}}_n \mathbf{J}_g(\mathbf{z}). \quad (17)$$

The partial derivatives of (8) and (9) w.r.t. \mathbf{q}_s can be easily computed analytically (see [1], [14]) and then plugged in (17) to evaluate (14), thus finding the optimal grasping parameter vector \mathbf{z}^* according to (13). The optimal Cartesian pose for the PSM $\mathbf{x}_{g,d}$ can be easily calculated from the optimal grasping parameter vector \mathbf{z}^* given the needle kinematics (5) and its global pose (1).

V. HAPTIC GUIDANCE

As discussed above, the haptic guidance purpose is to guide the user toward the optimal Cartesian space grasping pose. Let $\mathbf{x}_{g,d} = [\mathbf{p}_{g,d}^T, \boldsymbol{\phi}_{g,d}^T]^T \in \mathbb{R}^6$ be the optimal desired pose for the PSM end-effector frame \mathcal{F}_e with $\mathbf{p}_{g,d} \in \mathbb{R}^3$ denoting the position and $\boldsymbol{\phi}_{g,d} \in \mathbb{R}^3$ any parametrization of the orientation, *e.g.* Euler angles. $\mathbf{x}_{g,d}$ can be calculated as discussed in Sect. IV-B. The corresponding desired MTM pose $\mathbf{x}_{m,d} = [\mathbf{p}_{m,d}^T, \boldsymbol{\phi}_{m,d}^T]^T$ can be calculated from $\mathbf{x}_{g,d}$ through the following master-slave transformation

$$\mathbf{x}_{m,d} = \bar{\mathbf{R}}_c \mathbf{x}_{g,d} + \mathbf{x}_c, \quad \bar{\mathbf{R}}_c = \begin{bmatrix} \mathbf{R}_c & \mathbf{O} \\ \mathbf{O} & \mathbf{R}_c \end{bmatrix} \quad (18)$$

where $\bar{\mathbf{R}}_c \in \mathbb{R}^{6 \times 6}$ is the master-slave coupling rotation matrix and $\mathbf{x}_c = [\mathbf{p}_c^T, \boldsymbol{\phi}_c^T]^T \in \mathbb{R}^6$ its offset. Given $\mathbf{x}_{m,d}$ we can display haptic cues on the MTM using impedance control. The MTM is a 7-DoF serial robot arm with $\mathbf{q}_m \in \mathbb{R}^7$ denoting

the vector of its generalized coordinates. The MTM exhibits the following joint space dynamics

$$\begin{aligned} M_m(\mathbf{q}_m)\ddot{\mathbf{q}}_m + C_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m + \\ + B_m\dot{\mathbf{q}}_m + \mathbf{g}_m(\mathbf{q}_m) = \mathbf{u}_m, \end{aligned} \quad (19)$$

where $M_m(\mathbf{q}_m) \in \mathbb{R}^{7 \times 7}$ is the symmetric and positive-definite joint space inertia matrix, $C_m(\mathbf{q}_m, \dot{\mathbf{q}}_m) \in \mathbb{R}^{7 \times 7}$ consists of the Coriolis/centrifugal terms, $B_m \in \mathbb{R}^{7 \times 7}$ accounts for the friction term, $\mathbf{g}_m \in \mathbb{R}^7$ is the gravity term and $\mathbf{u}_m \in \mathbb{R}^7$ is a vector of control torques used to display haptic guidance forces to the operator. To realize a linear and decoupled Cartesian space impedance the control, inputs are designed as follows

$$\begin{aligned} \mathbf{u}_m = M_m(\mathbf{q}_m)\mathbf{v} + C_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m + \\ + \mathbf{g}_m(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m)\mathbf{w}_h \end{aligned} \quad (20)$$

$$\begin{aligned} \mathbf{v} = \mathbf{J}_{m,A}^\dagger(\mathbf{q}_m)M_m^{-1}(M_m\ddot{\mathbf{x}}_d + \mathbf{K}_p\dot{\mathbf{x}}_m + \mathbf{K}_d\dot{\mathbf{x}}_m + \\ - M_m\dot{\mathbf{J}}_{m,A}(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m - \mathbf{w}_{h,A}), \end{aligned} \quad (21)$$

where $\tilde{\mathbf{x}}_m = \mathbf{x}_{m,d} - \mathbf{x}_m \in \mathbb{R}^6$ with $\mathbf{x}_{m,d}$ given by (18), $\mathbf{J}_m(\mathbf{q})$, $\mathbf{J}_{m,A}(\mathbf{q}) \in \mathbb{R}^{6 \times 7}$ are MTM geometric and the analytical Jacobians, respectively, $\mathbf{K}_p, \mathbf{K}_d \in \mathbb{R}^{6 \times 6}$ are diagonal and positive-definite gain matrices to be opportunely designed and $\mathbf{w}_h \in \mathbb{R}^6$ is the vector of the external forces applied by an interacting user. Notice that the term $\mathbf{w}_{h,A}$ differs from \mathbf{w}_h by a mapping depending on the adopted orientation representation. The choice in (20) and (21) allow realizing the following linear and decoupled Cartesian space impedance

$$M_m\ddot{\tilde{\mathbf{x}}}_m + \mathbf{K}_d\dot{\tilde{\mathbf{x}}}_m + \mathbf{K}_p\tilde{\mathbf{x}}_m = \mathbf{w}_h, \quad (22)$$

which represents the sought haptic-guided shared control system. More details on the use of impedance control for haptic rendering with the dVRK robot are given in [22].

VI. EXPERIMENTS AND RESULTS

A. Experimental setup

In this section, experimental results using the dVRK platform (see Fig. 1a and 1b) are shown. One PSM is commanded by one MTM through the standard dVRK teleoperation architecture. The MTM impedance control was previously developed using the robot dynamic parameters identified in [23]. External force measurement is required by the impedance control laws (20) and (21). Since forces applied to the MTM are not directly measurable, estimation is performed through the nonlinear dynamic observer developed in [24]. The application of torque inputs is possible thanks to the open-source hardware and software architecture developed in [25]. The complete list of parameters used in this work

TABLE I
PARAMETERS

Parameter	Value	Parameter	Value
M_m	diag(1.5)	K_d	diag(20)
K_p	diag(100)	$\dot{\tilde{\mathbf{x}}}_d$	$\mathbf{0}_{6 \times 1}$
$\dot{\tilde{\mathbf{x}}}_d$	$\mathbf{0}_{6 \times 1}$	α^-	$-\pi/2$
α^+	$\pi/2$	s^*	1.0
n^*	1.0		

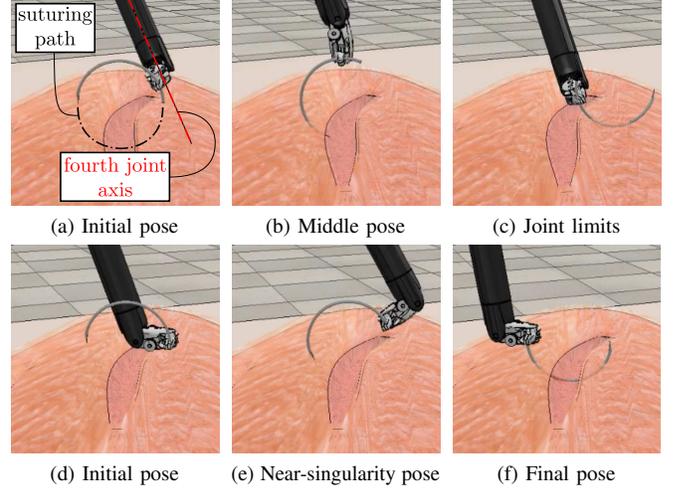


Fig. 5. The first row (5a-5b-5c) shows the effect of the joint limits occurrence during the suturing trajectory execution: the needle deviates from the desired trajectory when the fourth joint (see [18] for more information on PSM kinematics) reaches its upper limit (5c). The second row (5d-5e-5f) shows the singularity occurrence: the needle follows the path accurately but the manipulator undergoes large joint velocities (5e). Snapshot 5e corresponds to $t = 0.3$ s in Fig. 6c.

are shown in Table I, while PSM joint limits are given in Table II. The needle pose estimation follows the work in [6] (see Fig. 1c). The PSM inverse kinematics is implemented using the inverse Jacobian Closed Loop Inverse Kinematics Algorithm (CLIK) [26]. The overall pipeline, is first tested in simulation using the recently developed dVRK V-REP simulator [18]. The control loop rate is fixed at 5 ms while the lower level teleoperation loop runs at 1 ms.

B. Results

We designed three experiments to show that joint limits and singularities create serious issues. The first two experiments demonstrate the occurrence of these constraints during an autonomous suturing trajectory execution. The last experiment shows the effectiveness of our proposed haptic-guided shared control in choosing an optimal needle grasping pose (that yields neither joint limits nor singularities during the suturing task) and in guiding the operator towards it. The experiments are also shown in the accompanying video.

In experiment 1, a single stitch semi-circular trajectory (see Fig. 9) is defined between $p_i = [-0.08, -0.08, -0.106]$ and $p_e = [0.14, -0.08, -0.106]$ with center at $c = [0.03, -0.08, -0.105]$ where p_i, p_e and c are insertion point, exit point and the needle center position during stitching, respectively. All the coordinates are expressed in the world reference frame \mathcal{F}_r shown in Fig. 3. The needle is grasped at $z = [0.15, 0.0]$, see Fig. 4. This grasping configuration causes the robot to encounter joint limits during the suturing trajectory as shown in Figs. 5a-5b-5c. We apply a saturation

TABLE II
PSM JOINT LIMITS ([deg] OR [m])

Joint #	1	2	3	4	5	6
q_s^+	60	90	0.25	180	90	90
q_s^-	-60	-90	0.05	-180	-90	-90

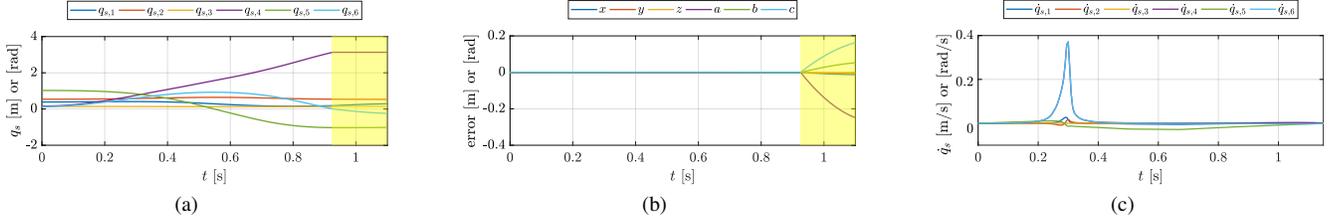


Fig. 6. *Experiment 1*: **6a** shows joint values reaching joint limits and saturated within the yellow shaded area; **6b** shows the task space tracking error along the suturing path, i.e. $p_d - p$ in Fig. 9. *Experiment 2*: **6c** shows the manipulator undergoing high joint velocities value during the suturing path. This corresponds to small TOV manipulability at that part of the path.

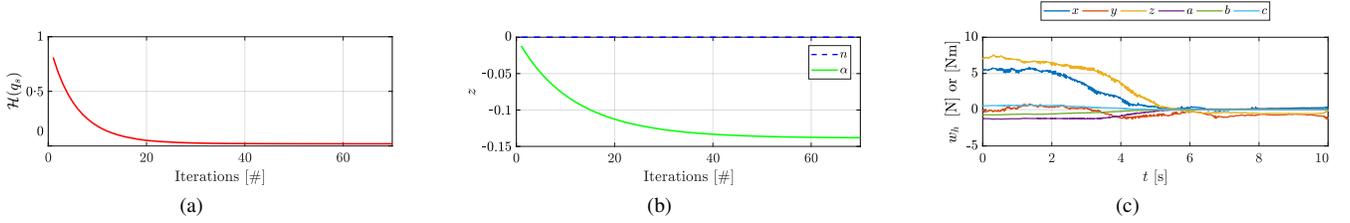


Fig. 7. *Experiment 3*: **7a** and **7b** show the evolution of the cost and parameters values during the execution of the optimization routine; **7c** shows haptic guidance force felt by the operator during the haptic-guided shared control grasping experiment. Higher forces are felt in the initial part of the experiment when the operator is far from the optimal grasping pose.

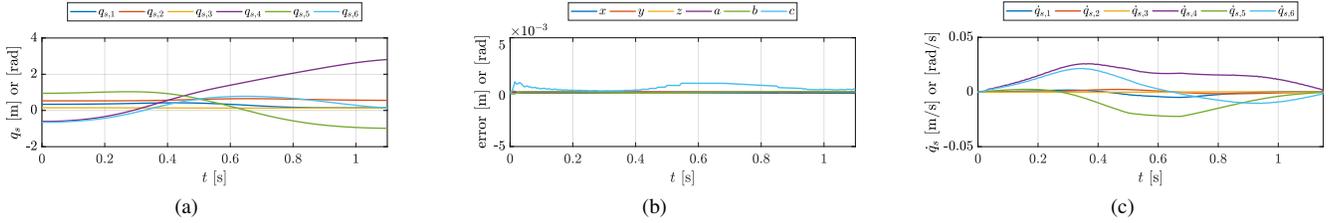


Fig. 8. *Experiment 3*: **8a** shows optimal joint values, far from reaching joint limits; **8b** shows the Cartesian space tracking error along the suturing path, i.e. $p_d - p$; **8c** shows that minimizing the cost of TOV manipulability results in reduced joint velocities (one order of magnitude smaller than those in experiment 2, shown in Fig. 6c).

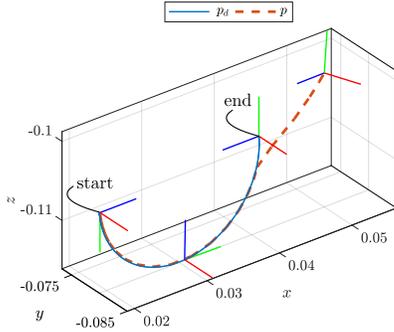


Fig. 9. Desired path p_d (defined between $p_i = [-0.08, -0.08, -0.106]$ and $p_e = [0.14, -0.08, -0.106]$ with center at $c = [0.03, -0.08, -0.105]$ where p_i , p_e and c are insertion and exit points and the position of the semi-circular path center, respectively) for the needle tip during suturing shown with blue solid line. When the needle can not follow the desired path due to joint limits, this results in actual path p shown in red dashed line.

policy in the joint values to simulate encountering joint limits. Figures 5a-5b-5c show that the needle tip diverges from the desired suturing trajectory due to the joint limit occurrence. This may be experienced by the surgeon quite often as it is not trivial for her/him to predict the joint limits and avoid them during the reach-to-grasp phase. Fig. 9 shows the desired (solid blue line) and the actual (red dashed line) trajectories. Furthermore, Fig. 6a shows the PSM joint values during the experiment. The joint limits occurrence

time slot is highlighted by the yellow shaded area. The joint exceeding its limit (in particular the fourth joint) is saturated in this region. Thus, the PSM cannot follow the desired trajectory yielding the Cartesian space error shown in Fig. 6b. In real procedures, this translates into (i) interrupting the task execution, (ii) releasing the needle and (iii) re-grasping it from a different pose. In experiment 2, we have chosen $p_i = [-0.08, -0.01, -0.106]$ and $p_e = [0.14, -0.01, -0.106]$ with center at $c = [0.03, -0.01, -0.105]$ to stress the issue caused by singularity during the suturing. Snapshots shown in Figs. 5d-5e-5f illustrates a sequence of PSM pose during this second trajectory. Although the trajectory is very similar to the one in experiment 1, the fast change of joint configuration (shown in Figs. 5d-5e-5f) occurs due to the proximity to singularity. Singularities are a common concern in manipulation that causes dangerous situations and must be avoided. Singularity occurrence translates into high joint velocities generated for small commanded Cartesian space displacements. Fig. 6c shows that joint velocities of the PSM reach a large value during the second trajectory execution.

In experiment 3, we consider again the suturing trajectory of experiment 1. The optimal grasping pose $x_{s,d}(z^* = [0, -0.138])$ is obtained through the above-explained optimization method (see Sect. IV-B) using as initial condition $z_0 = [0, 0]$. This choice is supported by real suturing

procedures observations: surgeons often grasp the needle towards its tail (to favor needle insertion), normal to the needle sagittal plane (to minimize collisions between the tool and patient’s anatomy). Fig. 7 shows the evolution of the cost function \mathcal{H} (given by (11)) and z during the optimization routine. The optimization problem can be solved relatively fast with the proposed gradient descent method. A non-optimized version of the code, over 1000 queries, takes on average $\mu_t = 1.0395$ s for each optimization step, with a variance of $\sigma_t^2 = 2.6046 \times 10^{-4}$ s.

The obtained grasping pose is used to generate force cues and inform a human operator during the reach-to-grasp phase. The operator feels the haptic cues, shown in Fig. 7c, during the real experiment shown in Fig. 2. The force cues intensity decreases by the closeness to the optimal grasping pose. Finally, Figs. 8a, 8b and 8c show the corresponding post-grasp joint positions, Cartesian space errors, and joint velocities that are obtained using the optimal grasping configuration during the suturing tasks execution. These figures demonstrate that the obtained grasping configuration allows simultaneously avoiding issues encountered in experiment 1 and 2. Hence, the proposed haptic-guided shared control method helps the operator to avoid joint limits and singularities that translate in significant Cartesian space errors and high joint velocities respectively during the suturing task execution.

VII. CONCLUSIONS

In this paper, we devised a novel haptic-guided shared control method for the da Vinci robot which assists the surgeon in grasping the needle in an optimal configuration. Our novel approach helps the surgeon grasping the needle such that the robot does not face joint limits and singularities during the post-grasping suturing movements. To achieve this, we optimize the joint limits and singularities cost by evaluating its gradient on the needle grasping manifold. This allows computing the optimal grasping pose and, sequentially, force cues (exerted via the MTM) attracting the operator’s hand towards it. Although haptic guidance forces inform the surgeon about the optimal needle grasping, the operator ultimately controls the system and decides which grasping pose to choose, thus taking into account other non-modelled objectives. We illustrate the effectiveness of our approach for optimal needle grasping using a real MTM and both simulated and real PSMs. The results show that the proposed haptic-guided shared control system significantly improves the performance during suturing in terms of distance from joint limits and singularities.

REFERENCES

- [1] A. Ghalamzan E. , F. Abi-Farraj, P. R. Giordano, and R. Stolkin, “Human-in-the-loop optimisation: Mixed initiative grasping for optimally facilitating post-grasp manipulative actions,” in *IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2017, pp. 3386–3393.
- [2] G. A. Fontanelli, M. Selvaggio, L. R. Buonocore, F. Ficuciello, L. Villani, and B. Siciliano, “A new laparoscopic tool with in-hand rolling capabilities for needle reorientation,” *IEEE Rob. Autom. Lett.*, vol. 3, no. 3, pp. 2354–2361, 2018.
- [3] M. Kopicki, R. Detry, M. Adjigble, R. Stolkin, A. Leonardis, and J. Wyatt, “One-shot learning and generation of dexterous grasps for novel objects,” *Int J Robot Res*, vol. 35, no. 8, pp. 959–976, 2016.
- [4] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt, “Grasp pose detection in point clouds,” *Int J Robot Res*, vol. 36, no. 13-14, pp. 1455–1473, 2017.
- [5] M. Gualtieri, A. ten Pas, K. Saenko, and R. Platt, “High precision grasp pose detection in dense clutter,” in *IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2016, pp. 598–605.
- [6] C. D’Ettore, G. Dwyer, X. Du, F. Chadebecq, F. Vasconcelos, E. D. Momi, and D. Stoyanov, “Automated pick-up of suturing needles for robotic surgical assistance,” in *IEEE Int. Conf. Rob. Autom.*, 2018, pp. 1370–1377.
- [7] M. Ragaglia, “Robot learning from demonstrations: Emulation learning in environments with moving obstacles,” *Rob. and Autonomous Systems*, vol. 101, pp. 45–56, 2018.
- [8] T. Osa, N. Sugita, and M. Mitsuishi, “Online trajectory planning in dynamic environments for surgical task automation,” in *Robotics: Science and Systems*, 2014, pp. 1–9.
- [9] R. Detry, J. Papon, and L. Matthies, “Task-oriented grasping with semantic and geometric scene understanding,” in *IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2017, pp. 3266–3273.
- [10] N. Mavrakis, R. Stolkin, L. Baronti, M. Kopicki, and M. Castellani, “Analysis of the inertia and dynamics of grasped objects, for choosing optimal grasps to enable torque-efficient post-grasp manipulations,” in *IEEE-RAS Int. Conf. Human. Rob.*, 2016, pp. 171–178.
- [11] T. Pardi, R. Stolkin, and A. Ghalamzan E, “Choosing grasps to enable collision-free post-grasp manipulations,” in *IEEE-RAS 18th Int. Conf. Human. Rob.*, 2018, pp. 299–305.
- [12] N. Ahmidi, L. Tao, S. Sefati, Y. Gao, C. Lea, B. Bejar, L. Zappella, S. Khudanpur, R. Vidal, and G. D. Hager, “A dataset and benchmarks for segmentation and recognition of gestures in robotic surgery,” *IEEE Trans. Biomed. Eng.*, pp. 2025–2041, 2017.
- [13] S. Sen, A. Garg, D. V. Gealy, S. McKinley, Y. Jen, and K. Goldberg, “Automating multi-throw multilateral surgical suturing with a mechanical needle guide and sequential convex optimization,” in *IEEE Int. Conf. Rob. Autom.*, 2016, pp. 4178–4185.
- [14] M. Selvaggio, F. Abi-Farraj, C. Pacchierotti, P. R. Giordano, and B. Siciliano, “Haptic-based shared-control methods for a dual-arm system,” *IEEE Rob. Autom. Lett.*, vol. 3, no. 4, pp. 4249–4256, 2018.
- [15] M. M. Marinho, B. V. Adorno, K. Harada, and M. Mitsuishi, “Active constraints using vector field inequalities for surgical robotics,” in *IEEE Int. Conf. Rob. Autom.*, 2018, pp. 5364–5371.
- [16] T. Liu and M. C. Cavusoglu, “Optimal needle grasp selection for automatic execution of suturing tasks in robotic minimally invasive surgery,” in *IEEE Int. Conf. Rob. Autom.*, 2015, pp. 2894–2900.
- [17] T. Liu and M. C. Cavusoglu, “Needle grasp and entry port selection for automatic execution of suturing tasks in robotic minimally invasive surgery,” *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 552–563, 2016.
- [18] G. A. Fontanelli, M. Selvaggio, M. Ferro, F. Ficuciello, M. Vendittelli, and B. Siciliano, “A v-rep simulator for the da vinci research kit robotic platform,” in *IEEE Int. Conf. Biomed. Rob. Biomech.*, 2018, pp. 1056–1061.
- [19] F. Nageotte, P. Zanne, C. Doignon, and M. de Mathelin, “Stitching planning in laparoscopic surgery: Towards robot-assisted suturing,” *Int J Robot Res*, vol. 28, no. 10, pp. 1303–1321, 2009.
- [20] R. C. Jackson and M. C. Cavusoglu, “Needle path planning for autonomous robotic surgical suturing,” in *IEEE Int. Conf. Rob. Autom.*, 2013, pp. 1669–1675.
- [21] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert, “Learning movement primitives,” in *Int. Symp. Rob. Res.* Springer, 2005, pp. 561–572.
- [22] M. Selvaggio, G. A. Fontanelli, F. Ficuciello, L. Villani, and B. Siciliano, “Passive virtual fixtures adaptation in minimally invasive robotic surgery,” *IEEE Rob. Autom. Lett.*, vol. 3, no. 4, pp. 3129–3136, 2018.
- [23] G. A. Fontanelli, F. Ficuciello, L. Villani, and B. Siciliano, “Modelling and identification of the da vinci research kit robotic arms,” in *IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2017, pp. 1464–1469.
- [24] A. D. Luca and R. Mattone, “Sensorless robot collision detection and hybrid force/motion control,” in *IEEE Int. Conf. Rob. Autom.*, 2005, pp. 999–1004.
- [25] P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor, and S. P. DiMaio, “An open-source research kit for the da vinci surgical system,” in *IEEE Int. Conf. Rob. Autom.*, 2014, pp. 6434–6439.
- [26] B. Siciliano, L. Sciacivco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer, 2010.