A Unified Framework for the Teleoperation of Surgical Robots in Constrained Workspaces

Murilo M. Marinho, Bruno V. Adorno, Kanako Harada, Kyoichi Deie, Anton Deguet, Peter Kazanzides, Russell H. Taylor, and Mamoru Mitsuishi

***Abstract*— In adult laparoscopy, robot-aided surgery is a reality in thousands of operating rooms worldwide, owing to the increased dexterity provided by the robotic tools. Many robots and robot control techniques have been developed to aid in more challenging scenarios, such as pediatric surgery and microsurgery. However, the prevalence of case-specific solutions, particularly those focused on non-redundant robots, reduces the reproducibility of the initial results in more challenging scenarios. In this paper, we propose a general framework for the control of surgical robotics in constrained workspaces under teleoperation, regardless of the robot geometry. Our technique is divided into a slave-side constrained optimization algorithm, which provides virtual fixtures, and with Cartesian impedance on the master side to provide force feedback. Experiments with two robotic systems, one redundant and one non-redundant, show that smooth teleoperation can be achieved in adult laparoscopy and infant surgery.**

I. Introduction

The da Vinci Surgical System (Intuitive Surgical, Inc., Sunnyvale, CA) has received considerable attention in the context of minimally invasive surgery, which involves procedures performed through small incisions. The robot is teleoperated: the surgeon generates motion commands on the *master side,* using a *master* interface; then, the commands are translated into motion by the *slave* robot, which interacts with the patient on the *slave side*.

The success of the da Vinci in adult laparoscopy has led to attempts to use it in surgical scenarios with workspaces more constrained than those in the initial target applications, such as infant surgery [1] and paranasal sinuses and skull base surgery [2]. However, these attempts have had limited success owing to the to large diameter and length of the da Vinci's tools and its large operating-room footprint. The fixed remote center-of-motion (RCM) is also a limitation. Alternative designs try to compensate for some of those drawbacks in adult laparoscopy [3], [4].

Other robotic systems have been developed to provide assistance in areas in which the da Vinci is hindered by its design. For instance, robots have been developed for procedures in restricted workspaces such as brain microsurgery [5], eye surgery [6], endonasal surgery [7], and pediatric surgery [8]. These robotic systems have several designs, such as serial linkage, as in the da Vinci system and others [5], [6], parallel linkage [8], and flexible tubes [7], [9]. There are also many control methodologies for the autonomous generation of constrained motion using *active constraints/virtual fixtures* [10]–[17].

An in-depth survey on active constraints is presented by Bowyer *et al*. [18], who show that most of the research in the field of virtual fixtures for teleoperated robots has focused on impedance control on the master side, along with techniques such as *proxy and linkage simulation* and *reference virtual fixtures*. Impedance control on the master side has been successful in pose[1] control of non-redundant robotic systems, such as the da Vinci, because generating virtual fixtures on the master means the slave can be effectively kept away from undesired interactions with the patient's anatomy [19].

However, such techniques, if applied only on the master side, are not suitable when the slave robot is redundant because, even if the master's and the slave's end-effector poses are the same (with respect to their own reference frames), the slave robot may have infinite configurations in joint-space [20]. Consequently, some slave robot's links can have harmful interactions with the patient, despite any feedback on the master.

As an alternative to master-side techniques, slave-side techniques have also been proposed, some of which use conventional control algorithms based on the Jacobian pseudoinverse and nullspace projection [20] to generate an RCM [13]–[15] or even more complex constrained workspaces [9]. Nevertheless, these standard techniques struggle to deal with *hard* constraints[2] that are important in the medical field, such as joint and actuation limits.

Considering hard limits, *constrained optimization* [10]–[12], [16], [17] is a more suitable approach to designing motion control laws on the slave side, because it naturally

This work was funded in part by the ImPACT Program of the Council for Science, Technology and Innovation (Cabinet Office, Government of Japan), in part by NSF grant 1637789, and in part by Johns Hopkins internal funds.

Murilo M. Marinho, Kanako Harada, and Mamoru Mitsuishi are with the Department of Mechanical Engineering, the University of Tokyo, Tokyo, Japan. Emails:{murilo, kanako, mamoru}@nml.t.u-tokyo.ac.jp.

Murilo M. Marinho's Ph.D. was supported by the Japanese Ministry of Education, Culture, Sports, Science, and Technology (MEXT). His stay in Johns Hopkins University was supported by the Graduate Program for Mechanical Systems Innovation (GMSI), from the University of Tokyo

Bruno V. Adorno is with the Federal University of Minas Gerais, Brazil. Email: adorno@ufmg.br. He has been supported by the Brazilian agencies CAPES, CNPq (grants 424011/2016-6 and 303901/2018-7), FAPEMIG, and by the INCT (National Institute of Science and Technology) under the CNPq grant 465755/2014-3.

Kyoichi Deie is with the Department of Pediatric Surgery, The University of Tokyo Hospital, Tokyo, Japan.

Anton Deguet, Peter Kazanzides, and Russell H. Taylor are with the Department of Computer Science, Johns Hopkins University, Baltimore, MD 21218 USA Email:{anton.deguet,pkaz,rht}@jhu.edu.

[1] *Pose* stands for combined position and orientation.

[2] Hard constraints cannot be violated [18], in contrast with *soft* constraints [11], in which small violations are allowed for short periods of time.

considers both inequality and equality constraints, while taking into account all of the system's DOF.

II. Related works

Initial approaches to constrained joint optimization in the generation of virtual fixtures [10]–[12] have been successful in providing constrained motion in complex scenarios, but have had issues such as being "computationally demanding and inconsistent for some constraint and cost functions" [18].

The computational demand resulting from the use of quadratic positional constraints and the difficulty of balancing virtual fixture and teleoperation terms in the objective function is reported by Kapoor *et al*. [11]. A follow-up work by Li *et al*. [12] has been shown to be computationally more efficient, as long as there is a single tool moving in the workspace , which is not the case in most surgical scenarios. Kwok *et al.* have proposed *ad hoc* techniques for snake robots [21]. Lastly, several validation studies have focused on a single robotic system in laparoscopic scenarios [10]–[12] and in sinus surgery [12], or on two robotic systems that follow a predefined trajectory in the contexts of deep neurosurgery [16] and transnasal surgery [17].

A general framework for constrained motion control that does not depend on specific robot designs can have several advantages. First, once constraints are defined to achieve a desired behavior (e.g., avoiding joint limits, preventing self-collisions and collisions with the workspace), those same constraints can be applied to other type of robots to achieve similar behavior. Second, the theoretical properties of the motion controller (e.g., time response, closed-loop stability, computational complexity) depend mostly on the framework, and a particular robotic platform has little or no influence on the closed-loop behavior. Third, researchers can focus on defining *new* relevant constraints for a particular robot design using a coherent theoretical framework, instead of resorting to *ad hoc* techniques. Thus, constrained optimization allows for the the most generalizable solution, once the aforementioned issues are solved.

A. Statement of contributions

In this paper, we propose a novel unified framework for robot control under teleoperation, which is presented in Section IV. First, we tackle the issue of teleoperation in constrained optimization approaches by proposing a teleoperation-oriented objective function, without adding to it any virtual fixture terms, which facilitates parameter tuning. Second, we combine the proposed objective function with the vector field inequality method (VFI) [16], [17] to provide dynamic active constraints. Third, we add *Cartesian impedance* to our framework, effectively solving the lack of haptic feedback of our earlier proposals [16], [17].

These three contributions allow us to perform teleoperation in complex scenarios, regardless of the to robot geometry. The generality of the proposed unified framework is tested in two bi-manual experiments using different robotic systems, as shown in Section V.

III. Mathematical Background

The proposed unified framework for surgical robot teleoperation uses quadratic programming for closed-loop inverse kinematics. To generate dynamic virtual fixtures, geometrical primitives are modeled using dual quaternion algebra, and linear constraints are added to the quadratic program using the VFI method. The basics of quadratic programming for closed-loop inverse kinematics, and the vector field inequalities method are briefly explained in this section.

A. Centralized quadratic programming for differential inverse kinematics of multiple robots

Differential kinematics is the relation between task-space velocities and joint-space velocities, in the general form $\dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{q}}$, in which $\boldsymbol{q} \triangleq \boldsymbol{q}(t) \in \mathbb{R}^n$ is the vector of manipulator joints' configurations, $\boldsymbol{x} \triangleq \boldsymbol{x}(\boldsymbol{q}) \in \mathbb{R}^m$ is the vector of m task-space variables, and $\boldsymbol{J} \triangleq \boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{m\times n}$ is a Jacobian matrix. The Jacobians relating the robot's joint velocities to its end-effector's unit dual quaternion pose ($\boldsymbol{J}_{\underline{\boldsymbol{x}}}$), rotation ($\boldsymbol{J}_r$), and translation ($\boldsymbol{J}_t$) can be found using dual quaternion algebra [22].

Suppose that p robots should reach their own independent task-space targets $\boldsymbol{x}_{i,d}$ ($\dot{\boldsymbol{x}}_{i,d} = \boldsymbol{0}$, $\forall i, t$), for $i = 1, \ldots, p$. Let each robot R_i have n_i joints, joint velocity vector $\dot{\boldsymbol{q}}_i$, task Jacobian $\boldsymbol{J}_i$, and task error $\tilde{\boldsymbol{x}}_i = \boldsymbol{x}_i - \boldsymbol{x}_{i,d}$. A suitable kinematic control law (assuming velocity inputs—i.e, $\boldsymbol{u} \triangleq \dot{\boldsymbol{q}}$) with *linear* constraints is given by

$$\begin{aligned} \boldsymbol{u} = \underset{\dot{\boldsymbol{q}}}{\arg\min}\ & \left\|\boldsymbol{J}\dot{\boldsymbol{q}} + \eta\tilde{\boldsymbol{x}}\right\|_2^2 + \lambda\left\|\dot{\boldsymbol{q}}\right\|_2^2 \\ \text{subject to } & \boldsymbol{W}\dot{\boldsymbol{q}} \preceq \boldsymbol{w}, \end{aligned} \tag{1}$$

where

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{J}_1 & \cdots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & \boldsymbol{J}_p \end{bmatrix}, \qquad \boldsymbol{q} = \begin{bmatrix} \boldsymbol{q}_1 \\ \vdots \\ \boldsymbol{q}_p \end{bmatrix}, \qquad \tilde{\boldsymbol{x}} = \begin{bmatrix} \tilde{\boldsymbol{x}}_1 \\ \vdots \\ \tilde{\boldsymbol{x}}_p \end{bmatrix},$$

$\boldsymbol{W} \triangleq \boldsymbol{W}(\boldsymbol{g}) \in \mathbb{R}^{r\times\sum n_i}$, $\boldsymbol{w} \triangleq \boldsymbol{w}(\boldsymbol{g}) \in \mathbb{R}^r$, $\eta \in (0,\infty)$ is a proportional gain, and $\boldsymbol{0}$ is a matrix of zeros with appropriate dimensions. The damping factor $\lambda \in [0,\infty)$ provides robustness to singularities [23].

B. Vector field inequalities in the generation of dynamic virtual fixtures

The VFI method for dynamic elements [17] first requires a function $d \triangleq d(\boldsymbol{q},t) \in \mathbb{R}$ that encodes the (signed) distance between two geometric primitives. Second, it requires a distance Jacobian and a residual relating the time derivative of the distance function and the joints' velocities in the general form

$$\dot{d} = \underbrace{\frac{\partial\left(d(\boldsymbol{q},t)\right)}{\partial\boldsymbol{q}}}_{\boldsymbol{J}_d}\dot{\boldsymbol{q}} + \zeta(t), \tag{2}$$

where the residual $\zeta(t) = \dot{d} - \boldsymbol{J}_d\dot{\boldsymbol{q}}$ contains the distance dynamics unrelated to the joints' velocities. The required distance function, distance Jacobians, and residuals for all relevant primitives used in this paper are shown in [17]. Lastly, the VFI method requires the definition of a safe distance $d_{\text{safe}} \triangleq d_{\text{safe}}(t) \in [0, \infty)$ and a distance error $\tilde{d} \triangleq \tilde{d}(\boldsymbol{q}, t) = d - d_{\text{safe}}$ to generate restricted zones or $\tilde{d} \triangleq d_{\text{safe}} - d$ to generate safe zones.

With these definitions, and given $\eta_d \in [0, \infty)$, the signed distance dynamics for each pair of primitives is constrained by

$$\dot{\tilde{d}} \geq -\eta_d \tilde{d}. \tag{3}$$

Constraint 3 assigns to each primitive a velocity constraint that actively filters the robot motion in the direction of the restricted zone boundary so that the primitives do not collide. At most, each primitive will converge to the boundary, and velocities tangential to the boundary itself are unaffected.

To use VFIs to generate restricted zones, we use the constraint

$$-\boldsymbol{J}_d \dot{\boldsymbol{q}} \leq \eta_d \tilde{d} + \zeta_{\text{safe}}(t), \tag{4}$$

for $\zeta_{\text{safe}}(t) \triangleq \zeta(t) - \dot{d}_{\text{safe}}$. Finally, safe zones are generated by using the constraint

$$\boldsymbol{J}_d \dot{\boldsymbol{q}} \leq \eta_d \tilde{d} - \zeta_{\text{safe}}(t). \tag{5}$$

IV. Proposed unified framework

The proposed framework is divided into two parts, with a constrained optimization algorithm that runs on the slave side and a Cartesian impedance feedback that runs on the master side. Both are explained in this section.

The technique proposed in this paper can be used to control any robotic system, as long as the forward kinematics model and Jacobian are available. Therefore, this includes serial-link, parallel-link, and even flexible robots [9].

A. Slave side: Constrained optimization

Existing approaches to constrained optimization have terms in the objective function for both trajectory tracking and virtual fixture generation, which is a major source of parameter tuning difficulties [11] and inconsistencies in constraints and cost functions [18]. To prevent issues related to having these mixed terms, the proposed technique includes only those terms related to trajectory tracking in the objective function.

In the proposed framework, translation and rotation are represented by quaternions. The quaternion set is $\mathbb{H} \triangleq \left\{h_1 + \hat{\imath}h_2 + \hat{\jmath}h_3 + \hat{k}h_4 : h_1, h_2, h_3, h_4 \in \mathbb{R}\right\}$, in which $\hat{\imath}^2 = \hat{\jmath}^2 = \hat{k}^2 = \hat{\imath}\hat{\jmath}\hat{k} = -1$. The conjugate of a quaternion $\boldsymbol{h} = h_1 + \hat{\imath}h_2 + \hat{\jmath}h_3 + \hat{k}h_4$ is given by $\boldsymbol{h}^* = h_1 - \left(\hat{\imath}h_2 + \hat{\jmath}h_3 + \hat{k}h_4\right)$ and $\operatorname{vec}_4 \boldsymbol{h} \triangleq \begin{bmatrix} h_1 & h_2 & h_3 & h_4 \end{bmatrix}^T$. Analogously, given a pure quaternion $\boldsymbol{t} = \hat{\imath}x + \hat{\jmath}y + \hat{k}z$, we define $\operatorname{vec}_3 \boldsymbol{t} \triangleq \begin{bmatrix} x & y & z \end{bmatrix}^T$.

Without loss of generality, suppose two identical slave robots are controlled through teleoperation, each by an independent master interface that generates a desired pose signal $\underline{\boldsymbol{x}}_{i,d}$. In this paper, we propose the following constrained optimization problem

$$\begin{aligned} \min_{\dot{\boldsymbol{q}}} \; & \beta \mathscr{F}_1 + (1 - \beta) \mathscr{F}_2 \\ \text{subject to } & \boldsymbol{W}\dot{\boldsymbol{q}} \preceq \boldsymbol{w}, \end{aligned} \tag{6}$$

where

$$\mathscr{F}_i \triangleq \alpha f_{t,i} + (1 - \alpha) f_{r,i} + f_{\Lambda,i},$$

in which $f_{t,i} \triangleq \left\|\boldsymbol{J}_{i,\boldsymbol{t}}\dot{\boldsymbol{q}}_i + \eta \operatorname{vec}_3 \tilde{\boldsymbol{t}}_i\right\|_2^2$, $f_{r,i} \triangleq \left\|\boldsymbol{J}_{i,\boldsymbol{r}}\dot{\boldsymbol{q}}_i + \eta \operatorname{vec}_4 \tilde{\boldsymbol{r}}_i\right\|_2^2$, and $f_{\Lambda,i} \triangleq \left\|\boldsymbol{\Lambda}\dot{\boldsymbol{q}}_i\right\|_2^2$ are the unweighted cost functions related to the end-effector translation, end-effector rotation, and joint velocities of the i-th robot, respectively; furthermore, each i-th robot has a vector of joint velocities $\dot{\boldsymbol{q}}_i$, a translation Jacobian (obtained using vec_3 instead of vec_4 as in [16], [22]) $\boldsymbol{J}_{i,\boldsymbol{t}}$, a translation error $\tilde{\boldsymbol{t}}_i \triangleq \boldsymbol{t}_i - \boldsymbol{t}_{i,d}$, a rotation Jacobian $\boldsymbol{J}_{i,\boldsymbol{r}}$, and a switching rotational error

$$\tilde{\boldsymbol{r}}_i \triangleq \begin{cases} (\boldsymbol{r}_i)^* \boldsymbol{r}_{i,d} - 1 & \text{if } \left\|\boldsymbol{r}_i^* \boldsymbol{r}_{i,d} - 1\right\|_2 < \left\|\boldsymbol{r}_i^* \boldsymbol{r}_{i,d} + 1\right\|_2 \\ (\boldsymbol{r}_i)^* \boldsymbol{r}_{i,d} + 1 & \text{otherwise,} \end{cases}$$

based on the dual quaternion invariant error [24], where $\boldsymbol{r}_{i,d}$ and $\boldsymbol{r}_i$ are the desired and current end-effector orientations, respectively. In addition, $\dot{\boldsymbol{q}} = \begin{bmatrix} \dot{\boldsymbol{q}}_1^T & \dot{\boldsymbol{q}}_2^T \end{bmatrix}^T$ and $\boldsymbol{\Lambda} \in \mathbb{R}^{n \times n}$ is a positive definite damping matrix, usually diagonal. Lastly, $\alpha, \beta \in [0, 1]$ are weights used to define the priorities between robots and between the translation and the rotation.

We use the linear constraints $\boldsymbol{W}\dot{\boldsymbol{q}} \preceq \boldsymbol{w}$ to avoid joints limits [23] and to generate active constraints using the VFIs [17]. Each parameter is explained in more detail in the following subsections.

1) The translation and rotation weight, α: The weight $\alpha \in [0, 1]$ is used to balance translational and rotational gains. In our application, the translation error is usually on a millimeter scale or lower. Therefore, the rotation error may overtake the translation error, depending on the units used to represent distance. Adding the weight α allows us to intuitively set that balance without other modifications to the optimization problem.

2) The robot prioritization weight, β: The weight $\beta \in [0, 1]$ is used to set a *soft* priority between robotic systems. To understand this parameter, first note that if Problem 6 has a solution, the objective function will be optimized, *given that the linear constraints are satisfied*. This means that the linear constraints prevent any collisions, even if this causes the trajectory tracking error of a particular robot to increase. In such cases, the parameter β can be used to weight the priority between the two robots. If $\beta > 0.5$, then minimizing the trajectory tracking error for robot 1 is favored over robot 2, effectively prioritizing robot 1. The reverse is true for $\beta < 0.5$. No explicit priority is given if $\beta = 0.5$.

3) The joint weight matrix, $\boldsymbol{\Lambda}$: Whenever the robot is redundant and has a heterogeneous structure, for instance a robotic manipulator with n_R DOF attached to a customized forceps with n_F DOF, the damping matrix $\boldsymbol{\Lambda}$ can be written in the form

$$\boldsymbol{\Lambda} \triangleq \begin{bmatrix} \boldsymbol{\Lambda}_R & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Lambda}_F \end{bmatrix},$$

in which $\mathbf{\Lambda}_R \in \mathbb{R}^{n_R \times n_R}$ and $\mathbf{\Lambda}_F \in \mathbb{R}^{n_F \times n_F}$ are matrices used to increase the relative weights of joints we wish to have move less than others. For instance, given $\mathbf{\Lambda}_R \triangleq \lambda_R \boldsymbol{I}_{n_R}$ and $\mathbf{\Lambda}_F \triangleq \lambda_F \boldsymbol{I}_{n_F}$, with $\lambda_R, \lambda_F \in (0, \infty)$, we can favor forceps motion over manipulator motion by setting $\lambda_R > \lambda_F$.

4) The switching unit quaternion controller: Because the group of unit quaternions double covers $\mathrm{SO}(3)$, both $\boldsymbol{r} \in \mathbb{S}^3$ and $-\boldsymbol{r}$ represent the same orientation, which causes the unwinding problem [25]. In practice, this problem results in undesired motions whenever a continuous control law is employed. In order to see that, suppose that the orientation error is given only by $(\boldsymbol{r}_i)^* \boldsymbol{r}_{i,d} - 1$; if $\boldsymbol{r}_{i,d} = \boldsymbol{r}_i$, then the orientation error is equal to 0. However, if $\boldsymbol{r}_{i,d} = -\boldsymbol{r}_i$, then the orientation error is equal to -2, although the current orientation is already the desired one. In that case, the robot moves unnecessarily until again reaching the new equilibrium point. A way to circumvent this problem is to use discontinuous or hybrid control laws [25], which in our case is done by switching the error. This way, if $(\boldsymbol{r}_i)^* \boldsymbol{r}_{i,d}$ is closer to 1, the error is given by $(\boldsymbol{r}_i)^* \boldsymbol{r}_{i,d} - 1$; conversely, if $(\boldsymbol{r}_i)^* \boldsymbol{r}_{i,d}$ is closer to -1, the error is given by $(\boldsymbol{r}_i)^* \boldsymbol{r}_{i,d} + 1$.

B. Master side: Cartesian impedance

In order to provide haptic feedback based on virtual fixtures, we add a Cartesian force feedback on the master side that is proportional to the current error on the slave side, in the form

$$\mathbf{\Gamma}_{i,\text{master}} \triangleq -\eta_f \tilde{\boldsymbol{t}}_i^{\text{master}} - \eta_V \dot{\boldsymbol{t}}_{i,\text{master}}, \qquad (7)$$

for each master–slave pair, where $\mathbf{\Gamma}_{i,\text{master}}$ is the reflected force on the master side, $\eta_f, \eta_V \in (0, \infty)$ are, respectively, stiffness and viscosity parameters, $\tilde{\boldsymbol{t}}_i^{\text{master}}$ is the translation error of the slave, but seen from the point of view of the master, and $\dot{\boldsymbol{t}}_{i,\text{master}}$ is the linear velocity of that master interface. This proportional force feedback with viscosity allows the operator to "feel" any task-space directions in which the robot has difficulty moving.

V. Experiments[3]

In order to evaluate the technique proposed in this paper, we first present experiments to evaluate the effects of β and the dynamic active constraints using the da Vinci Research Kit (dVRK) [26], which is a research-friendly robotic system comprising the same master and slave robotic systems of the da Vinci Surgical System. Second, we present a peg transfer experiment to evaluate the proposed framework in complex tasks. For this second experiment, a seven-DOF robot was operated by a medical doctor.

The software implementation was the same for both systems, namely Ubuntu 16.04 x64 running ROS Kinetic Kame.[4] Robot kinematics was implemented using the DQ Robotics[5] library, and constrained convex optimization was implemented using IBM ILOG CPLEX Optimization Studio[6] with Concert Technology.

[3]See accompanying video.
[4]http://wiki.ros.org/kinetic/Installation/Ubuntu
[5]http://dqrobotics.sourceforge.net
[6]https://www.ibm.com/bs-en/marketplace/ibm-ilog-cplex

A. dVRK experiments

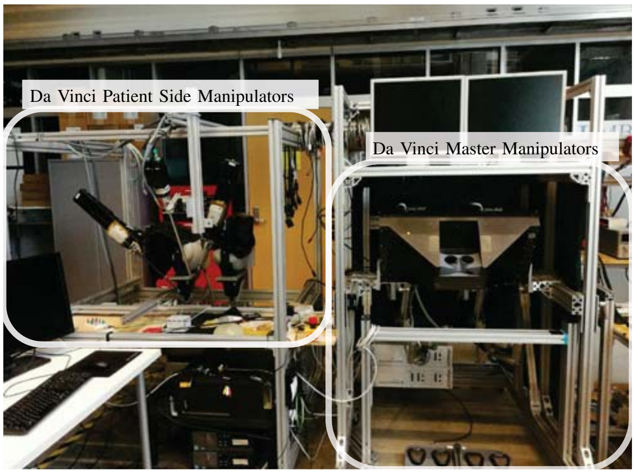

Fig. 1. The dVRK experimental setup. Two slave arms were commanded through two master arms.

The first set of experiments used the experimental setup shown in Fig. 1 and was devised to evaluate the effects of a change in the prioritization weight β, while dynamic active constraints to prevent collisions between shafts were enabled. Three types of constraints were added: a shaft-to-shaft distance constraint, to prevent collisions between tool shafts; a plane-to-point constraint, to prevent collisions between the right tool and the peg transfer board; and a joint limit constraint. All were implemented using VFIs [17].

The experiment involved manipulating a triangle on a peg transfer board, which is the same peg transfer board used in the Fundamentals of Laparoscopic Surgery (FLS) curriculum.[7] For repeatability, before the task began, the right tool was positioned on a central peg and the triangle was placed on the bottom-right peg closest to the right tool. Only the left tool was allowed to move. The right tool was commanded to stay in a constant pose throughout the procedure.

The user had to pick and place the triangle in a clockwise motion, which required the triangle to be transferred between five pegs. Reaching the four initial pegs should not induce any collisions between tools and were useful to show whether the prioritization was cumbersome outside of collision situations. The last target peg was the same as the first to close the peg transfer circle. Reaching the last peg required the left tool to push on the right tool's shaft. The behavior of the system was evaluated under three different levels of prioritization, as shown in Table I. The other parameters are shown in Table II.

1) Results and discussion: Snapshots of the peg transfer task using the dVRK are shown in Fig. 2, for each of the three experimental cases. Complete footage of each experiment is shown in the accompanying video. Trajectory and force data for all cases are shown in Fig. 3.

[7]http://www.flsprogram.org

TABLE I

THE PRIORITIES USED IN THE dVRK TELEOPERATION EXPERIMENTS.

	Same priority	Left tool higher priority	Left tool lower priority
β	0.5	0.99	0.01

Suppose the tracking error of the tools should be 10 mm in order to prevent a shaft–shaft collision. $\beta = 0.5$ means that, in order to prevent a collision, both arms' trajectory tracking errors are increased by the same amount, therefore, 5 mm each. $\beta = 0.99$ means that the left tool has a tracking error of 0.1 mm and the right tool an error of 9.9 mm. The reverse holds for $\beta = 0.01$.

TABLE II

CONTROL PARAMETERS OF PROBLEM 6 USED IN EACH EXPERIMENT.

	α	β	η	η_d	η_f	Λ_R	Λ_F	η_V	MS
dVRK	0.99	(Tab. I)	1	1	350	0.01	0.01	10	1/2
Infant	0.99	0.5*	80	1	100	0.01	0.0	10	1/3

*In the case of the infant experiment, there was no active constraint relating both robots; therefore, we set $\beta = 0.5$.
η, η_d: proportional gain of the kinematic controller and the VFI, respectively.
α: translation error to orientation error weight (Section IV-A.1).
β: robot prioritization weight (Section IV-A.2).
Λ_R, Λ_F: Robot and forceps joint gains, respectively (Section IV-A.3).
η_F, η_V: Cartesian impedance proportional and viscosity gains, respectively (Section IV-B).
MS: Motion scaling. A motion scaling of X means that a relative translation of the master was multiplied by X before being sent to the slave.

In the first case ($\beta = 0.5$), the left tool could reach all pegs, as required by the task. The right tool autonomously evaded the left tool whenever the left tool was commanded to a region that would cause a collision. Although the positioning of the triangle on the last peg was possible, it required considerable force from the operator to push the right tool, which peaked at about 10N.

In the second case ($\beta = 0.99$), the left tool could reach all pegs, as required by the task. The force feedback on the left tool was weak and barely distinguishable from the viscosity-induced feedback; therefore, the left tool could even place the triangle on the peg over which the right tool was initially located.

Finally, in the last case ($\beta = 0.01$), the left tool was not able to reach all pegs in the prescribed order. The user could feel a strong force feedback whenever forcing the left tool against the right tool. Even with considerable force from the operator, 20N, the right tool did not move away.

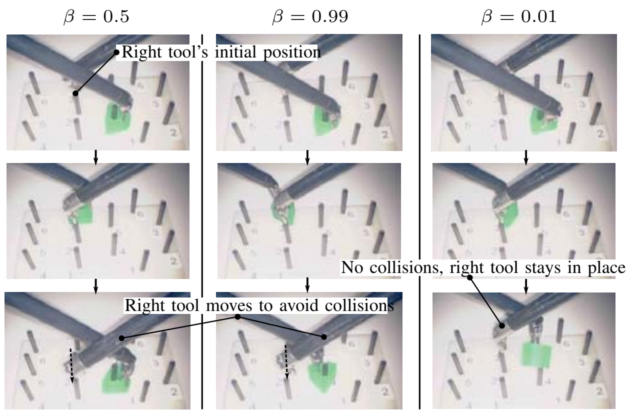

Fig. 2. Snapshots of the dVRK experiments used to evaluate the influence of parameter β. The first column corresponds to $\beta = 0.5$ (same priority), the second corresponds to $\beta = 0.99$ (left tool with higher priority), and the third column corresponds to $\beta = 0.01$ (left tool with lower priority).

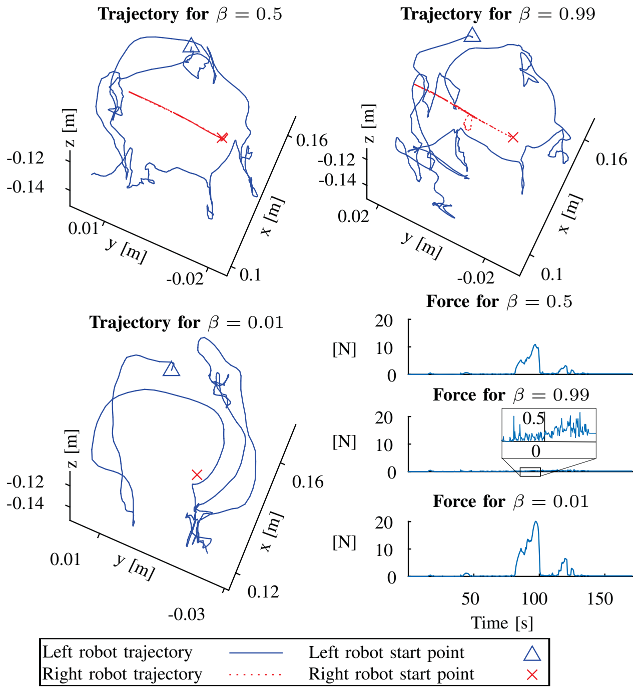

Fig. 3. Force feedback, based on virtual fixtures, for the controlled tool and trajectories of both tools in all three trials. The trajectories of the left and right tools are shown in *blue* and *dotted red*, respectively.

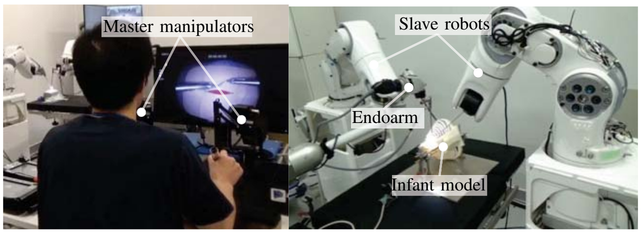

Fig. 4. The master–slave configuration used in the peg transfer infant experiments. The medical doctor commanded two haptic interfaces and the robot automatically generated constraints to avoid dangerous collisions with the infant model.

These results show that the parameter β can be used to prioritize tools in an intuitive manner. How to effectively use this in a surgical task is left to future work.

B. Infant peg transfer experiments

In this task, our target was to determine whether a medical doctor could perform a difficult task under teleoperation in a constrained workspace. Therefore, an expert in manual laparoscopic pediatric surgery was invited to participate in this preliminary experiment.

The constraints in infant surgery are considerably more complex than those in adult laparoscopy, and the da Vinci was shown to be inadequate for this type of surgery [1]. In

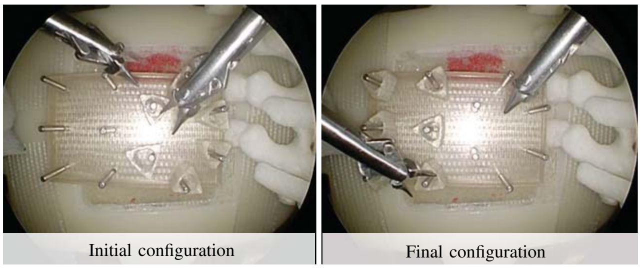

Fig. 5. Snapshots of the initial and final states of the peg transfer experiment with the medical doctor.

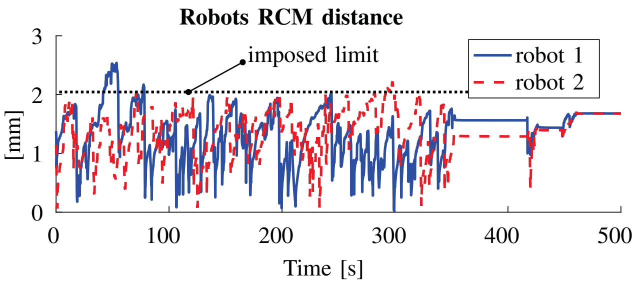

Fig. 6. Distance of each robot's tool shaft to the center of their respective entry-sphere.

this context, we employed a surgical system that is being developed in parallel to this work.

Three types of constraints are required in infant surgery. First, medical doctors use the compliance of the infant's skin to increase the reachable workspace. This compliance can be considered in our framework by generating an entry-sphere (shaft-to-point distance with safe distance larger than zero), rather than using an entry point. Second, the tool might move outside of the camera's field-of-view owing to the small size of the workspace. Even though this situation is common in manual surgery, because medical doctors rely on their spatial perception of their bodies to locate tools, such out-of-bounds motion is highly undesirable in robot-aided surgery owing to safety concerns. In this context, a safety cuboid constraint was added for each individual robotic arm. Lastly, joint limits were also considered.

As in the FLS curriculum, the medical doctor was asked to transfer the triangles from one side of the peg transfer board to the other.

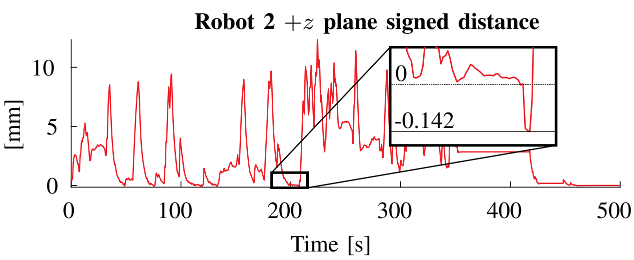

Fig. 7. Distance of the second robot's tool tip to the upper wall of the workspace cuboid, in which the constraint was slightly violated. No effective violation was measured in the other walls.

1) Results and discussion: The medical doctor participated in three trials, one of which is illustrated in Fig. 5. With very little experience using the proposed system, the medical doctor was able to perform a full peg transfer experiment in about 7 min. Overall, the medical doctor gave a high evaluation of the robotic system usability.

Qualitatively, after inspecting a video recording of the robot motion during the peg transfer experiment, it was visible that the entry-sphere constraint was properly maintained. There were no rib dislocations and no model motion, which happened when using the da Vinci [1].

Quantitatively, the tool shaft distance to the entry-sphere center is shown in Fig. 6, as measured from the robot's encoders. The maximum distances between each robot shaft and the center of its entry-sphere were 2.54 mm and 2.41 mm, respectively. This means there was a maximum constraint violation of 0.5 mm. Understanding the source of this constraint violation is a topic of ongoing research. The culprit is thought to be the discrete time implementation of Problem 6.

Another important set of constraints was the planar constraints making up the cuboid workspace. Among the 12 plane constraints, the maximum constraint violation corresponded to the plane that impeded the right robot's tool tip from being retracted from the model, as shown in Fig. 7. The magnitude of the violation was 0.142 mm, which is of a similar magnitude to the constraint violation of the entry-sphere. Other planes showed negligible constraint violations of under 0.1 mm. Because the right robot tool tip was kept at the border of that plane during most of the experiment, this indicates why a higher violation of that plane was observed.

These results show that a complex task, with several active constraints, can be performed smoothly under teleoperation by a medical doctor using the proposed framework. How well the framework can operate in still more complex scenarios, including flexible tools, is a topic of ongoing research.

VI. Conclusions and Future Work

In this paper, a novel unified framework for robot control under teleoperation was proposed. The method can be used to provide smooth teleoperation, regardless of the robot geometry and under workspace constraints. On the slave side, a constrained optimization algorithm provides virtual fixtures for collision avoidance and the avoidance of joint limits. On the master side, a Cartesian impedance algorithm allows the user to "feel" directions in which the robot has difficulty moving. The proposed framework is evaluated in two scenarios, with different robot geometries. First, we demonstrate a shaft–shaft collision avoidance with tool prioritization under teleoperation using the dVRK. Second, we show a peg transfer experiment performed by a medical doctor using a redundant robot system in an infant surgery scenario.

In future works, we plan to test the performance of the framework in the teleoperation of flexible robots.

REFERENCES

[1] S. Takazawa, T. Ishimaru, K. Harada, K. Deie, A. Hinoki, H. Uchida, N. Sugita, M. Mitsuishi, T. Iwanaka, and J. Fujishiro, "Evaluation of surgical devices using an artificial pediatric thoracic model: A comparison between robot-assisted thoracoscopic suturing versus conventional video-assisted thoracoscopic suturing," *Journal of Laparoendoscopic & Advanced Surgical Techniques*, vol. 28, pp. 622–627, May 2018.

[2] J. S. Schneider, J. Burgner, R. J. Webster, and P. T. Russell, "Robotic surgery for the sinuses and skull base," *Current Opinion in Otolaryngology & Head and Neck Surgery*, vol. 21, pp. 11–16, Feb. 2013.

[3] M. J. Lum, D. C. Friedman, G. Sankaranarayanan, H. King, K. Fodero, R. Leuschke, B. Hannaford, J. Rosen, and M. N. Sinanan, "The Raven: Design and validation of a telesurgery system," *The International Journal of Robotics Research*, vol. 28, no. 9, pp. 1183–1197, 2009.

[4] V. Larocca, F. Marino, A. De Filippis, S. Gidaro, and A. Lococo, "A new operative telesurgical system: Telelap Alf-X experimental study on animal model," *Journal of Advanced Biotechnology and Bioengineering*, vol. 2, no. 1, pp. 12–15, 2014.

[5] M. Mitsuishi, A. Morita, N. Sugita, S. Sora, R. Mochizuki, K. Tanimoto, Y. M. Baek, H. Takahashi, and K. Harada, "Master–slave robotic platform and its feasibility study for micro-neurosurgery," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 9, no. 2, pp. 180–189, 2013.

[6] R. Taylor, P. Jensen, L. Whitcomb, A. Barnes, R. Kumar, D. Stoianovici, P. Gupta, Z. Wang, E. Dejuan, and L. Kavoussi, "A steady-hand robotic system for microsurgical augmentation," *The International Journal of Robotics Research*, vol. 18, no. 12, pp. 1201–1210, 1999.

[7] J. Burgner, D. C. Rucker, H. B. Gilbert, P. J. Swaney, P. T. Russell, K. D. Weaver, and R. J. Webster, "A telerobotic system for transnasal surgery," *IEEE/ASME Transactions on Mechatronics*, vol. 19, pp. 996–1006, June 2014.

[8] T. Looi, B. Yeung, M. Umasthan, and J. Drake, "Kidsarm - an image-guided pediatric anastomosis robot," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Nov. 2013.

[9] K. Leibrandt, P. Wisanuvej, G. Gras, J. Shang, C. A. Seneci, P. Giataganas, V. Vitiello, A. Darzi, and G. Yang, "Effective manipulation in confined spaces of highly articulated robotic instruments for single access surgery," *IEEE Robotics and Automation Letters*, vol. 2, pp. 1704–1711, July 2017.

[10] J. Funda, R. H. Taylor, B. Eldridge, S. Gomory, and K. G. Gruben, "Constrained cartesian motion control for teleoperated surgical robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 3, pp. 453–465, 1996.

[11] A. Kapoor, M. Li, and R. H. Taylor, "Constrained control for surgical assistant robots," in *Robotics and Automation (ICRA), 2006 IEEE International Conference on*, pp. 231–236, 2006.

[12] M. Li, M. Ishii, and R. H. Taylor, "Spatial motion constraints using virtual fixtures generated by anatomy," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 4–19, 2007.

[13] 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*, pp. 5807–5812, IEEE, 2013.

[14] C. D. Pham, F. Coutinho, A. C. Leite, F. Lizarralde, P. J. From, and R. Johansson, "Analysis of a moving remote center of motion for robotics-assisted minimally invasive surgery," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 1440–1446, IEEE, 2015.

[15] B. Dahroug, B. Tamadazte, and N. Andreff, "Visual servoing controller for time-invariant 3d path following with remote centre of motion constraint," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3612–3618, May 2017.

[16] M. M. Marinho, B. V. Adorno, K. Harada, and M. Mitsuishi, "Active constraints using vector field inequalities for surgical robots," in *Robotics and Automation (ICRA), 2018 IEEE International Conference on*, pp. 5364–5371, IEEE, May 2018.

[17] M. M. Marinho, B. V. Adorno, K. Harada, and M. Mitsuishi, "Dynamic active constraints for surgical robots using vector field inequalities," *preprint available at https://arxiv.org/abs/1804.11270*.

[18] S. A. Bowyer, B. L. Davies, and F. R. y Baena, "Active constraints/virtual fixtures: A survey," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 138–157, 2014.

[19] M.-A. Vitrani, C. Poquet, and G. Morel, "Applying virtual fixtures to the distal end of a minimally invasive surgery instrument," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 114–123, 2017.

[20] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Advanced Textbooks in Control and Signal Processing, London: Springer-Verlag London, 2009.

[21] K.-W. Kwok, K. H. Tsoi, V. Vitiello, J. Clark, G. C. Chow, W. Luk, and G.-Z. Yang, "Dimensionality reduction in controlling articulated snake robot for endoscopy under dynamic active constraints," *IEEE Transactions on Robotics*, vol. 29, no. 1, pp. 15–31, 2013.

[22] B. V. Adorno, *Two-arm Manipulation: From Manipulators to Enhanced Human-Robot Collaboration [Contribution à la manipulation à deux bras : des manipulateurs à la collaboration homme-robot]*. PhD Dissertation, Université Montpellier 2, 2011.

[23] F.-T. Cheng, T.-H. Chen, and Y.-Y. Sun, "Resolving manipulator redundancy under inequality constraints," *IEEE Transactions on Robotics and Automation*, vol. 10, pp. 65–71, Feb. 1994.

[24] L. Figueredo, B. Adorno, J. Ishihara, and G. Borges, "Robust kinematic control of manipulator robots using dual quaternion representation," in *2013 IEEE International Conference on Robotics and Automation*, pp. 1949–1955, IEEE, May 2013.

[25] H. T. Kussaba, L. F. Figueredo, J. Y. Ishihara, and B. V. Adorno, "Hybrid kinematic control for rigid body pose stabilization using dual quaternions," *Journal of the Franklin Institute*, vol. 354, pp. 2769–2787, May 2017.

[26] 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 Intl. Conf. on Robotics and Auto. (ICRA)*, (Hong Kong, China), pp. 6434–6439, 2014.