This paper appears in Elsevier Mechatronics
https://www.sciencedirect.com/science/article/pii/S0957415821000301/

Constrained Haptic-Guided Shared Control for Collaborative Human-Robot Percutaneous Nephrolithotomy Training

Olivia Wilz[a], Ben Sainsbury[b], and Carlos Rossa[a]

[a] *Faculty of Engineering and Applied Science, Ontario Tech University, Oshawa, ON, Canada.*
[b] *Marion Surgical, Toronto, ON, Canada.*

Abstract

Percutaneous nephrolithotomy is a procedure used to treat patients with large or irregularly shaped kidney stones. Surgical instruments are inserted through a small incision to access the kidney and remove the calculi. Surgeons who have less experience with the procedure manifest significantly higher rates of complications due to the procedure's steep learning curve. This issue is further exacerbated by a lack of training opportunities in clinical settings. This paper introduces a teleoperative framework that can provide training to surgeons as well as assistance during procedures, based on two main components. Firstly, a type of constrained inverse kinematics that decouples the tooltip position from its orientation using a remote centre of motion, and incorporates the joint limits analytically. This reduces the workload of the procedure by having the surgeon control only the tooltip position rather than the position and the orientation while preventing the inverse kinematics from returning joint angels outside of the robot's abilities. This kinematic framework also allows a three-degrees-of-freedom haptic device to control a six-degrees-of-freedom manipulator. Secondly, haptic feedback is provided to help guide and teach the surgeon during the procedure. Haptic feedback allows the surgeon to remain in full control during the procedure while still receiving haptic cues and assistance.

Experimental results indicate that the haptic cues improved user's accuracy, and they had shorter and smoother paths. This leads to a shorter procedure time overall. The results also indicate that the haptic assistance helped teach users the ideal trajectory of the procedure and that users who were taught with haptic feedback performed better than those who never experienced any haptic feedback.

Keywords: Percutaneous nephrolithotomy, medical robotics, haptic assistance, constrained inverse kinematics, remote centre of motion, human-robot interface

1. Introduction

Percutaneous nephrolithotomy (PCNL) has rapidly been established as the main management of urinary calculi for patients with large or irregularly shaped kidney stones. The procedure involves accessing the kidney via a small incision in the patient's back, through which thin instruments are threaded. A biplanar fluoroscope captures real-time moving images of the kidney allowing the surgeon to see its internal structure, the stones, and the surgical tools. Once the stones are localized they are either removed through a tube or broken up with high-intensity ultrasound waves before being removed [1].

Despite over 40 years of continuous development and widespread clinical acceptance, PCNL remains an arduous procedure to learn. Particularly in less experienced hands, the procedure can lead to higher complication rates compared to other endoscopic interventions [1, 2]. Developing adequate expertise to achieve surgical competence in PCNL is not trivial, in addition to gaining access to the kidney, the procedure also requires proficiency with stone fragmentation, tool navigation, and stone scavenging. Despite the documented safety and efficacy of percutaneous renal access, as few as 11% of urologists successfully gain access to the kidney stones themselves [3]. This low success rate is generally attributed to a lack of skills and limited training opportunities. Due to the procedure's steep learning curve and inherent complexity, PCNL residents will inevitably perform a number of inadequate interventions during their training process [2, 4].

Several complications related to PCNL are due to mistakes in tool steering while gaining kidney access; this includes puncturing the pleural, which can lead to pneumothorax, hydrothorax, or hemothorax; and puncturing nearby organs such as the colon; it is possible to suffer a vascular injury from the puncture of the renal pelvis or a calyceal infundibulum [5–8]. The tool must be inserted through a calyx in the kidney, missing the calyx leads to additional bleeding or complications like infec-

[1] Email: olivia.wilz@ontariotechu.ca; ben@marionsurgical.com; carlos.rossa@ontariotechu.ca.

[2] We acknowledge the support of Marion Surgical and the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference number 2018-06074].

[3] Cette recherche a été menée en collaboration avec Marion Surgical et a été financée par le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG), [numéro de référence 2018-06074].

tions [5, 6]. Another possible cause of excessive bleeding is the sharp angulation of the nephroscope or sheath during PCNL [5, 6], therefore, careful control of the orientation is vital.

A critical question in PCNL is how to define and teach an optimal tool path that is necessary to gain proper access to the kidney stones. The tool path obtained from experienced surgeons can serve as a standard measure of performance for novices and also help them develop the required surgical skills. The question then arises: if what one defines as the optimal tool path is known from experience, how can this information complement or compensate for a resident's lack of skill during surgery? The answer may be sought in two distinct yet complementary paradigms: surgery simulators for training and robotic assistance.

In the first paradigm, virtual and augmented reality are used as a platform for skill development. These simulators are becoming an elemental building block in medical training for PCNL [9]. One example of this is the PCNL simulator from Marion Surgical [10] with haptic feedback capabilities. Another example is [11], where augmented reality was incorporated with robotic assistance to perform PCNL. Haptic feedback in both virtual and teleoperative training environments is known to provide reduced learning times, improve task performance, quality, dexterity, and retention rates [12–14] and create better acceptance of simulator training by professionals [15]. A common method of generating constraints for haptic feedback are virtual fixtures; their application is explored in [16–18]. An alternative, is the use of sensors to provide haptic cues based on tissue characteristics. In [19], haptic feedback is based on tissue impedance. A force sensor was used in [20] to measure the axial force on the needle and apply the same force through haptic feedback. Chowriappa *et al.* propose providing haptic assistance to novices based on expert demonstrations for trocar insertion using Gaussian mixture models and Gaussian mixture regression to generate a predictive model [21]. Alternatively, in [22], haptic feedback is created based on obstacles in the workspace and an expert demonstration given in a simulation.

The second paradigm to improve PCNL performance proposes incorporation of some level of robotic assistance. For example, one can automate a surgical sub-task such as tool orientation, stone fragmentation, stone removal, etc., while the surgeon learns to perform a complementary task. Sub-task automation allows the workload to be shared between the operator. In addition, data acquired from expert surgeons through a simulator can define an optimal tool path, and subsequently a robotic agent can assist the surgeon in following that path during surgery. In this context, less experienced surgeons would benefit from the expertise of more experienced surgeons.

Kidney access and percutaneous tool manoeuvring are prime examples of where robotic assistance can be beneficial. Establishing access to the kidney is the first and the most crucial step in PCNL [23]. Typically done under two-dimensional image guidance, it becomes challenging to visualize and mentally recreate the three-dimensional anatomy of the kidney and the relative location of stones and tools. To simplify the workload, the entry point into the kidney should be constrained to minimize damage to the surrounding tissue. With the assistance of a robotic manipulator, a remote centre of motion (RCM) can be used to constrain the entry point in this fashion.

RCMs are used in a variety of other applications. The most common use of an RCM is through a physical mechanism that restricts the available motion such as those used in [24–27]. Less common is the inclusion of a remote centre of motion to the inverse kinematics directly. Some examples of this do exist, such as [28], where the incision point during teleoperation was constrained by considering it a linear joint in the kinematic framework. Similarly, [29] applied the concept of an RCM, like those used in teleoperative surgery, to a mobile robot with a robotic manipulator. Applying an RCM in the inverse kinematics ensures that the tool always passes through the original incision point in the patient. In addition to the RCM, it is important to consider the joint limits of the robotic manipulator to prevent entering error states, colliding with itself, and posing an additional risk to the patient. Together, the RCM and the joint limits play a crucial role to reduce tissue damage and the risk of complications.

With a teleportation scheme, the robotic agent helps the operator follow a predetermined kidney access path while maintaining the RCM. Further, the incorporation of haptic feedback allows the surgeon to compensate for trajectory tracking errors, while maintaining full control of the procedure. To address these issues, this paper proposes merging the two paradigms described earlier to implement a novel cyber-physical simulator with robotic assistance and haptic feedback for PCNL training.

The cyber-physical PCNL simulator proposed in this paper has three complementary contributions. First, a teleoperation framework is used where the position and orientation of the robot are decoupled and the operator only controls the Cartesian position of the tool tip while the robot ensures that the position of the entry point in the tissue remains unchanged. This is the idea behind an RCM. A key difference here compared to previous works is that the RCM is incorporated into the inverse kinematics by considering the entry point as the tip of a tool with variable length, while the joint limits are included analytically in the formulation using a saturation function for the joint speeds. This is the focus of section 2.

The second contribution lies in the implementation of the haptic assistance. The simulator bases the potential field on an expert demonstration provided in the physical slave environment, where the demonstrator controls the robot through the haptic device, and the stiffness of the potential energy field is dependent on the proximity to the phantom tissue. This potential field is then used to determine the magnitude and direction haptic feedback force to apply. This is discussed in section 3. Force fields are defined based on demonstrations acquired in the phys-

ical environment directly. This way, uncertainties and unmodelled dynamics of the environment are accounted for and the haptic feedback reflects the same conditions faced by the experts demonstrating the procedure.

The final contribution proposes a method of evaluating the effectiveness of the framework proposed in this paper. Current literature primarily evaluates performance in virtual environments, while the experiments performed for this paper consider a cyber-physical environment. Sixteen participants took part in a total of five trials each. The participants were each placed in one of three groups; the first group received no haptic assistance, the second group received haptic assistance for all trials, and the third group is trained using haptic assistance for the first three trials and completed the last two evaluation trials without any haptic assistance. To the best of the author's knowledge, this is the first implementation of a cyber-physical simulator with haptic and robotic assistance for PCNL.

The remainder of the paper is organized as follows. Section 4 focuses on user trials along with the validation of the analytical joint limitations. It was found that the analytical joint limits functioned as anticipated; as a joint approaches a limit it begins to saturate and the inverse kinematics returns an alternative joint configuration to reach the desired point. The user trials demonstrated that those with haptic assistance had shorter path lengths overall as well as a shorter procedure time; they also demonstrated smoother and more efficient trajectories when compared to the groups without haptic assistance. The third group - trained with haptic feedback - demonstrated an overall better performance than those without, however, they did not perform as well as the participants who received haptic assistance during all trials. These results are discussed in more detail in section 5. The overall performance of the contributions is discussed in section 6, along with potential future improvements.

2. Constrained Robot Kinematics for Kidney Access in PCNL

During PCNL a long thin tool is inserted into a small incision into the patient's back to gain access to the kidney; this requires the control of the position of the tooltip as well as the orientation of the tool to ensure that it continues to pass through the entry point during insertion.

In the context of robot-assisted PCNL, these two points are sufficient to solve for the inverse kinematics of the robotic manipulator using an RCM. Since the robot will control the tool to ensure that it passes through the two points in space, the inverse kinematics need to be developed in terms of six Cartesian coordinates (two 3D points) rather than three Cartesian coordinates defining the position and three angles defining the orientation. The inverse kinematics should take into account the joint limits of the manipulator so as not to exceed them during operation.

To configure the inverse kinematics as described above, the forward kinematics must be defined to get the RCM and the tooltip position based on the joint angles of the manipulator.

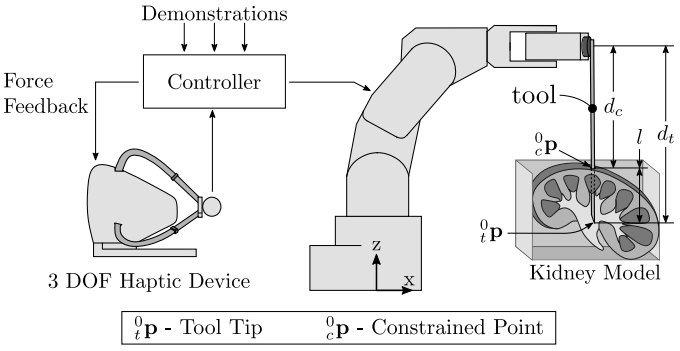


Figure 1: The general setup of cyber-physical robot-aided PCNL training framework. The 3 DOF haptic device provides position information to the controller while receiving force feedback information. The controller sends joint angles to the 6 DOF robotic arm, allowing the 3 DOF haptic device to control the 6 DOF robotic arm.

2.1. Forward Kinematics

The forward kinematics of the robot arm is derived using the modified Denavit-Hartenberg convention which specifies a set of homogeneous transformations ${}^{i-1}_{i}\mathbf{T}$ that expresses the position and orientation of the kinematic chain's i^{th} joint with respect to joint $i-1$ as:

$$ {}^{i-1}_{i}\mathbf{T} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -d_i s\alpha_{i-1} \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & d_i c\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) $$

where $c(\cdot)$ stands for $\cos(\cdot)$ and $s(\cdot)$ for $\sin(\cdot)$, θ_i is the angle of the i^{th} joint about its rotational axis, α_{i-1} is the angle from the previous $(i-1)$ rotational axis to the current (i^{th}) rotational axis sometimes called *link twist*, a_{i-1} is the distance from the previous joint axis to the current joint axis (generally the previous link length) and d_i is the displacement of the current link along its axis of rotation also known as the *link offset*.

The transformation matrices are multiplied, as shown in (2) and (3) where j is the total number of degrees-of-freedom (DOF). The tooltip ${}^{j}_{t}\mathbf{p} \in \mathbb{R}^{4\times1}$ and constrained point ${}^{j}_{c}\mathbf{p} \in \mathbb{R}^{4\times1}$ can be described relative to the j^{th} reference frame by using only the fourth column of (1), since their orientations are not necessary. The tooltip position ${}^{0}_{t}\mathbf{p} \in \mathbb{R}^{4\times1}$ is

$$ {}^{0}_{t}\mathbf{p} = \prod_{i=1}^{j} ({}^{i-1}_{i}\mathbf{T})\ {}^{j}_{t}\mathbf{p}. \quad (2) $$

The constrained point position ${}^{0}_{c}\mathbf{p} \in \mathbb{R}^{4\times1}$ is

$$ {}^{0}_{c}\mathbf{p} = \prod_{i=1}^{j} ({}^{i-1}_{i}\mathbf{T})\ {}^{j}_{c}\mathbf{p}. \quad (3) $$

Since the constrained point is along the tool its location can be defined in the same way as the tooltip with the only

notable difference being the tool length d_t and d_c as shown in Fig. 1. The tool itself will have a constant length, while the distance to the constrained point is variable depending on the insertion depth l of the tool inside the kidney such that $d_c = d_t - l$. This means that ${}^j_t\mathbf{p}$ can be defined as:

$$ {}^j_t\mathbf{p} = \begin{bmatrix} a_j & -d_t s\alpha_j & d_t c\alpha_j & 1 \end{bmatrix}^T, \tag{4} $$

where the transpose is denoted by $(\cdot)^T$, and ${}^j_c\mathbf{p}$ can be defined relative to ${}^j_t\mathbf{p}$ as:

$$ {}^j_c\mathbf{p} = \begin{bmatrix} a_{i-1} & -d_c s\alpha_j & d_c c\alpha_j & 1 \end{bmatrix}^T = {}^j_t\,\mathbf{p} - \begin{bmatrix} 0 \\ -ls\alpha_j \\ lc\alpha_j \\ 0 \end{bmatrix}, \tag{5} $$

The vectors created in (2) and (3) together contain six equations for describing the tool's tip and constrained point positions relative to the base. Altogether they make up the forward kinematic solution. These equations are used to find the inverse kinematic formulas; however, the results need to be bounded by the joint limits of the manipulator. To take the joint limits into account during the inverse kinematic model, a nonlinear saturation function is implemented that limits the speed of a given joint when it approaches its limit. This is implemented as a nonlinear gain in the Newton Raphson method. Which has the added benefit of providing a unique and feasible solution rather than multiple solutions where a particular configuration needs to be selected.

2.2. Constrained Inverse Kinematics

The tooltip and constrained point positions and velocities can both be found using the current joint angles and angular velocities. To find the constrained inverse kinematics the opposite must be done, such that the joint angles can be found based on the tooltip and constrained point positions.

Let p_i be the i^{th} entry in vector $\mathbf{p}$ defined in (2) and (3). One can define the task space vector $\mathbf{x} \in \mathbb{R}^{6\times 1}$ as

$$ \mathbf{x} = [{}^0_t p_1 \;\; {}^0_t p_2 \;\; {}^0_t p_3 \;\; {}^0_c p_1 \;\; {}^0_c p_2 \;\; {}^0_c p_3]^T. \tag{6} $$

Normally, the Jacobian is constructed using three equations describing the position and three describing the orientation of the manipulator. In (6), however, the Jacobian is constructed from the six position equations, three for the tooltip position and three for the RCM. These equations define the position of the tool; it is important to note that while this allows the 6 DOF manipulator to be controlled with a 3 DOF device, only 5 DOF are being accounted for since the tool can be rotated about its longitudinal axis which does not affect its Cartesian position. Let:

$$ \mathbf{\Theta} = [\theta_1 \;\; \theta_2 \;\; \ldots \;\; \theta_j]^T. \tag{7} $$

be the joint space vector of the manipulator ($\in \mathbb{R}^{j\times 1}$). The Jacobian describes the relationship between $\dot{\mathbf{x}}$ (the vector of the Cartesian velocities) and $\dot{\mathbf{\Theta}}$ (the vector of the angular velocities) as

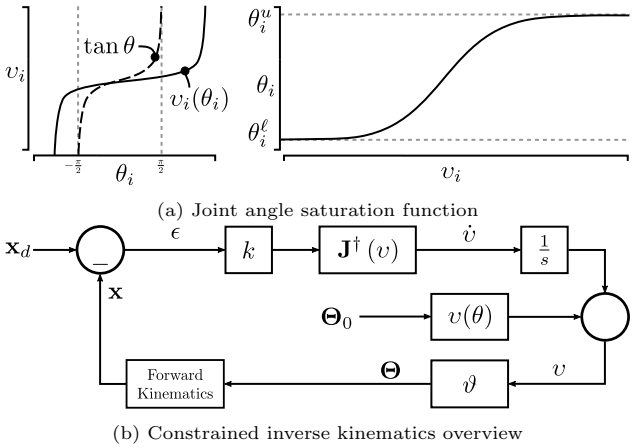
$$ \dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{\Theta}}, \tag{8} $$

here the time derivative is denoted by the $(\dot{\cdot})$ operator. In (8), $\mathbf{J} \in \mathbb{R}^{6\times j}$ is defined as:

$$ \mathbf{J} = \frac{\partial \mathbf{x}}{\partial \mathbf{\Theta}} = \begin{bmatrix} \frac{\partial {}^0_t p_1}{\partial \theta_1} & \frac{\partial {}^0_t p_1}{\partial \theta_2} & \cdots & \frac{\partial {}^0_t p_1}{\partial \theta_j} \\ \frac{\partial {}^0_t p_2}{\partial \theta_1} & \frac{\partial {}^0_t p_2}{\partial \theta_2} & \cdots & \frac{\partial {}^0_t p_2}{\partial \theta_j} \\ \frac{\partial {}^0_t p_3}{\partial \theta_1} & \frac{\partial {}^0_t p_3}{\partial \theta_2} & \cdots & \frac{\partial {}^0_t p_3}{\partial \theta_j} \\ \frac{\partial {}^0_c p_1}{\partial \theta_1} & \frac{\partial {}^0_c p_1}{\partial \theta_2} & \cdots & \frac{\partial {}^0_c p_1}{\partial \theta_j} \\ \frac{\partial {}^0_c p_2}{\partial \theta_1} & \frac{\partial {}^0_c p_2}{\partial \theta_2} & \cdots & \frac{\partial {}^0_c p_2}{\partial \theta_j} \\ \frac{\partial {}^0_c p_3}{\partial \theta_1} & \frac{\partial {}^0_c p_3}{\partial \theta_2} & \cdots & \frac{\partial {}^0_c p_3}{\partial \theta_j} \end{bmatrix}. \tag{9} $$

Joint angles must be found for the manipulator that satisfy the desired tooltip and constrained point positions. These joint angles must be achievable for the physical manipulator. In this formulation, the joint limits will be considered during the inverse kinematics to find a feasible solution. This is done by converting the joint angles to a different variable that saturates as it approaches a joint limit. The variable is then converted back to the original joint space where it is now bounded by the limits.

First, let θ_i^u and θ_i^ℓ be the upper and lower limits of joint i respectively. The desired transformation function used to convert the joint angles to a new space must be continuously increasing within the open interval $\left(\theta_i^\ell \;\; \theta_i^u\right)$. The arctangent function is one that meets these criteria. It is used by linearly mapping θ_i from the joint limits $\left(\theta_i^\ell \;\; \theta_i^u\right)$



(a) Joint angle saturation function

(b) Constrained inverse kinematics overview

Figure 2: The inverse kinematics of the system, left (a): the arctangent function as given in (10), where υ_i is the remapped joint angle; and right (a): converting the joint angles back to the joint space (11) where they are restricted by the joint limits. (b) shows the control loop of the inverse kinematics, it takes the desired tooltip position $\mathbf{x}_d$ and initial joint angles $\mathbf{\Theta}_0$ as inputs and outputs the corresponding joint angles $\mathbf{\Theta}$ by minimizing error ϵ, k is a constant gain and the pseudo-inverse, constrained Jacobian is $\mathbf{J}^\dagger(\upsilon)$.

to the open interval $\left(-\frac{\pi}{2}\ \frac{\pi}{2}\right)$ by

$$\upsilon_i(\theta_i) = \tan\left(\frac{\pi(2\theta_i - \theta_i^u - \theta_i^\ell)}{2(\theta_i^u - \theta_i^\ell)}\right) \tag{10}$$

which is shown in Fig. 2(a) on the left-hand side. The inverse of (10) converts the transformed joint angle back into physical joint angles θ_i as follows

$$\vartheta_i(\upsilon_i) = \theta_i(\upsilon_i) = \frac{\theta_i^u - \theta_i^\ell}{\pi}\tan^{-1}(\upsilon_i) + \frac{\theta_i^u + \theta_i^\ell}{2}. \tag{11}$$

Note that θ_i is now bounded between the joint limits θ_i^ℓ and θ_i^u as shown in Fig. 2(a) on the right hand side and is now denoted as ϑ_i.

To compute the inverse kinematics, a new Jacobian has to be constructed based on the modified joint space. The constrained Jacobian $\mathbf{J_c}$ is now computed as the partial derivative of the task space for the transformed joint space as

$$\mathbf{J}_c(\upsilon) = \frac{\partial \mathbf{x}}{\partial \upsilon} = \mathbf{J}(\mathbf{\Theta})d\vartheta, \tag{12}$$

where $\mathbf{J}(\mathbf{\Theta})$ is defined in (9) and $d\zeta$ is a $j \times j$ diagonal matrix given by

$$d\vartheta = \begin{bmatrix} \frac{\partial \vartheta_1}{\partial \upsilon_1} & 0 & \cdots & 0 \\ 0 & \frac{\partial \vartheta_2}{\partial \upsilon_2} & \cdots & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \frac{\partial \vartheta_j}{\partial \upsilon_j} \end{bmatrix}, \tag{13}$$

and

$$\frac{\partial \vartheta_i}{\partial \upsilon_i} = \frac{\theta_i^u - \theta_i^\ell}{\pi(1+\upsilon_i^2)}. \tag{14}$$

Now the inverse solution of (8) can be found considering the change of variable and the joint limits as:

$$\dot{\upsilon} = \mathbf{J}^\dagger(\upsilon)\dot{\mathbf{x}} \tag{15}$$

where $\mathbf{J}^\dagger(\upsilon)$ is the pseudo-inverse, constrained, Jacobian matrix defined as

$$\mathbf{J}^\dagger(\upsilon) = \mathbf{J}_c(\upsilon)^T\left[\mathbf{J}_c(\upsilon)\mathbf{J}_c(\upsilon)^T + \mu\mathbf{I}\right]^{-1} \tag{16}$$

where $\mathbf{I} \in \mathbb{R}^{6\times 6}$ is an identity matrix and $\mu \in \mathbb{R}^+ << 1$ is a damping constant scalar used to avoid possible discontinuity of the pseudo-inverse at a singular configuration of the manipulator.

Let υ_d be a solution to a desired Cartesian position of the tooltip and constrained point $\mathbf{x}_d \in \mathbb{R}^{6\times 1}$, and ϵ define the error between the desired and actual Cartesian position as

$$\epsilon = \mathbf{x}_d - \mathbf{x}. \tag{17}$$

A proportional control law in the form of

$$\dot{\upsilon} = k\mathbf{J}^\dagger(\upsilon)\epsilon \tag{18}$$

guarantees that $\epsilon = \mathbf{x}_d - \mathbf{x} \to 0$ (that is $\mathbf{x}_d = \mathbf{x}$), and thus υ approaches υ_d provided that the constant k is positive. This gives a least squared solution to (15) when the Jacobian is full rank. The solution ensures that the $\dot{\upsilon}$ is minimized given that (13) increases monotonically in the open interval $(\theta_i^\ell\ \theta_i^u)$. The inverse kinematics process is shown in Fig. 2(b), where the desired position and constrained point $\mathbf{x}_d$ is given as an input along with the initial joint angles $\mathbf{\Theta}_0$. The error ϵ is calculated as the difference between the desired position $\mathbf{x}_d$ and the current position $\mathbf{x}$. The error is multiplied by the constant gain and the pseudo-inverse Jacobian $\mathbf{J}^\dagger$ which gives the change in angle $\dot{\upsilon}$. Integrating the change in angle and adding the initial joint values provides the new joint values which just need to be converted to the joint space through (11). This loop is repeated until $\epsilon \cong 0$ at which point the joint angles are sent to the manipulator.

This completes the inverse kinematics section. The inverse kinematics have been defined based on the forward kinematics while applying the joint limits. The forward kinematics were defined based on joint angles and the physical structure of the manipulator. A 6 DOF (or greater) robotic arm can now be controlled using a 3 DOF haptic device through the inverse kinematics. The next step is to generate haptic feedback based on the tooltip position and expert demonstrations. The haptic feedback can then be used to guide the user through the procedure.

3. Haptic Assistance from Expert Demonstrations

To guide the current operator with the help of expert demonstrators, first the expert demonstrations have to be collected. These demonstrations are the recorded tooltip positions during the procedure. These points provide information about the path the demonstrator used to accomplish the specified task.

To provide assistance in the preferred directions of motion, we propose using the well-known concept of potential fields introduced in [30]. The use of potential fields allows the haptic feedback provided to be time-invariant; this is very important because it keeps the surgeon in full control of the operation, allowing them to make active adjustments during the procedure such as: correct for possible changes in the environment, deal with issues that did not appear in preoperative imaging, or a change in the position of the kidney or kidney stones, *etc.* The term interaction impedance defines the ratio of applied force to the magnitude of deviation from the reference trajectory. It allows adapting the haptic feedback properties by making it more or less compliant in certain regions, while cooperatively accomplishing a task such as following a given path. We propose to calculate the assistive forces based on a non-parametric potential field function [30]. The forces applied by the haptic device and its spatial impedance are captured by a potential function gradient and curvature.

3.1. Potential Field Functions from trajectory demonstrations

Potential fields have been used to guide robotic manipulators through actions given by human demonstrations. These potential fields can also be applied as haptic feedback to guide the human user through an action. This would allow the user to make their own path and have an active role in the procedure while still receiving information from experts. This time-invariant controller with bounded force range guarantees that the robot is stable when in contact with passive environments.

The tooltip position is used as an index to get the haptic feedback that needs to be applied to the user based on the location of the tooltip in 3D space.

All of the re-sampled demonstration data points can now be concatenated in a single vector $\xi_d \in \mathbb{R}^{3\times N}$, where N is the total number of demonstration data points from all demonstrations. The n^{th} column in ξ_d is given by

$$\xi_d^n = \begin{bmatrix} {}^0_t p_1 & {}^0_t p_2 & {}^0_t p_3 \end{bmatrix}^T. \tag{19}$$

The potential field has to be created for a specified workspace whose points in 3D space are denoted $\xi \in \mathbb{R}^{3\times k}$ in which ξ represents the 3D Cartesian coordinate of a point within the defined workspace boundary and k is the total number of points used to represent the workspace. The higher the k value is, the higher the resolution and smoothness of the potential field, at the cost of computational time and memory usage.

A potential energy field ϕ is generated for demonstration point n as follows:

$$\phi_i^n(\xi) = \phi_0^n + \frac{1}{2} S^n \left(\xi_i - \xi_d^n\right)^T \left(\xi_i - \xi_d^n\right), \quad \forall n \in 1 \dots N, \forall i \in 1 \dots k \tag{20}$$

where the stiffness and the initial potential energy for the n^{th} data point, are defined as S^n and ϕ_0^n respectively and ξ_i is the i^{th} entry in ξ.

A Gaussian Kernal is used to create a weighting element for each demonstration point as

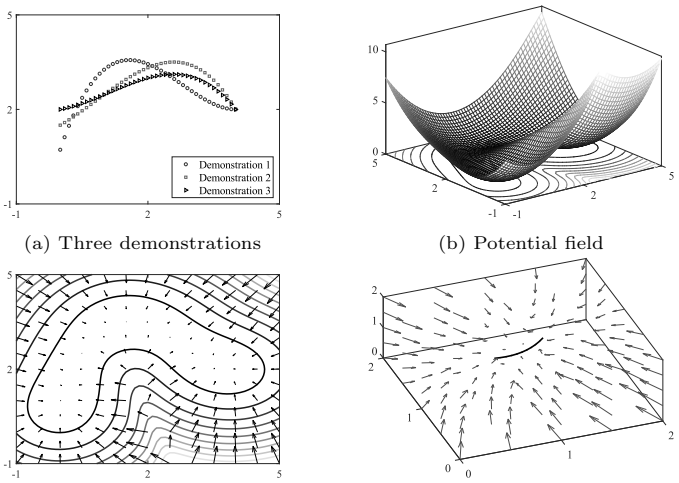
$$\omega_i^n(\xi) = e^{-\frac{1}{2(\sigma^n)^2}(\xi_i - \xi_d^n)^T(\xi_i - \xi_d^n)} \quad \forall n \in 1 \dots N, \forall i \in 1 \dots k, \tag{21}$$

in which, $\sigma^n \in \mathbb{R}^+$ is a smoothing parameter that controls the region of influence for the n^{th} data point. Using weighted sums, the potential field $\mathbf{\Phi} \in \mathbb{R}^{1\times k}$ at ξ can be computed using element-wise multiplication and division as:

$$\mathbf{\Phi}(\xi) = \frac{\sum_{n=1}^{N} \omega^n(\xi)\, \phi^n(\xi)}{\sum_{n=1}^{N} \omega^n(\xi)}. \tag{22}$$

One can also implement a dissipative field in the form of a controllable damper parameter to take into account the robot's speed in the haptic forces. The damping element ψ^n is created for each demonstration point n as follows:

$$\psi^n\left(\dot{\xi}\right) = \mathbf{D}^n \dot{\xi} \tag{23}$$



(a) Three demonstrations (b) Potential field

(c) Gradient of Potential Field (d) 3D example

Figure 3: Potential field construction starting with 2D reference trajectories (a) shows three sample demonstrations, in (b) the potential field for the three demonstrations is shown along with the potential field projected onto the bottom plane. Plot (c) represents the gradient of the potential field. The length of the arrow is determined by the magnitude at the gradient of that point. Plot (d) shows an example of a three dimensional demonstration (the black line in the centre), where the arrows again represent the gradient that is taken of the potential field.

where $\mathbf{D}^n$ is a dissipative gain and $\dot{\xi}$ is the velocity at ξ. These damping elements are then combined similarly to the way that the potential fields are combined to generate a dissipative field $\mathbf{\Psi} \in \mathbb{R}^{1\times k}$ by computing the weighted sums for element ξ using element-wise multiplication and division as:

$$\mathbf{\Psi}\left(\xi, \dot{\xi}\right) = \frac{\sum_{n=1}^{N} \omega^n(\xi)\, \psi^n\left(\dot{\xi}\right)}{\sum_{n=1}^{N} \omega^n(\xi)}. \tag{24}$$

Finally, the dissipative field and the potential field are combined to create a force field that will be used to apply a force that guides the user along a desired trajectory $\xi_{\mathbf{d}}$. The Force field is generated by subtracting the dissipative field from the negative gradient of the potential field as follows:

$$\mathbf{F} = -\nabla \mathbf{\Phi}(\xi) - \mathbf{\Psi}\left(\xi, \dot{\xi}\right) \tag{25}$$

where ∇ is the gradient. In Fig. 3(b) an example of a potential field $\mathbf{\Psi}$ is shown, while in Fig. 3(c)&(d) the arrows are the force $\mathbf{F}$. The force field will now be used as a lookup table when providing haptic assistance.

The reference trajectories are obtained from demonstrations given by experts. These demonstration points, however, are not evenly distributed in 3D space. This means that the sample points may be clustered together in one area while being spread apart widely in another. This unevenness can lead to some areas being too heavily weighted. Because of this, the demonstrations should be re-sampled so that all of the points are evenly spread apart. Alternatively, an optimization method can be used to assign the

Table 1: Modified Denavit-Hartenberg Parameters for Meca500

i	1	2	3	4	5	6	t	c	**unit**
α_{i-1}	0	-90	0	-90	90	-90	-90	-90	*deg*
a_{i-1}	0	0	135	38	0	0	0	0	*mm*
d_i	135	0	0	120	0	70	$-L$	$-\ell$	*mm*
θ_i	θ_1	θ_2-90	θ_3	θ_4	θ_5	θ_6	0	0	*deg*

ideal value of stiffness $\mathbf{S}$ and smoothing parameter σ to each demonstration to generate a smooth potential field such as in [30].

Since each demonstration point is composed of three Cartesian coordinates, the potential field will exist in the fourth dimension. This increase in dimensionality is shown in Fig. 3(a)&(b), part (a) shows several two-dimensional demonstration points while part (b) shows the potential field that was built for these sample demonstrations; note that the potential field is three dimensional.

After the gradient is taken of the potential field the dimensionality returns to that of the demonstrations, as shown in 3c&d the arrows representing the gradient of the potential field, are all two-dimensional for the 2D demonstrations and three-dimensional for the 3D demonstrations.

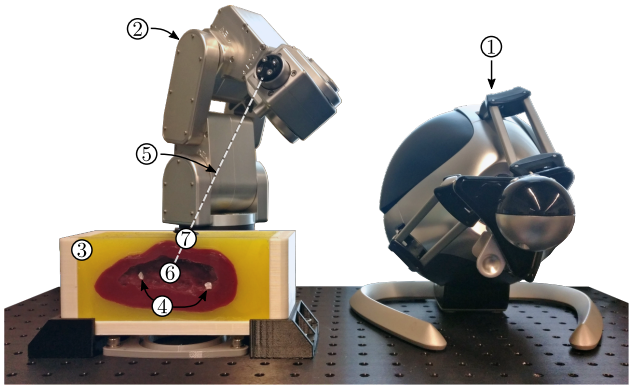
3.2. Using potential fields for haptic assistance in PCNL

Since it is a 3 DOF haptic device it outputs three Cartesian coordinates to the inverse kinematics. Given a set of reference trajectories, the potential field forces can now be used to provide haptic assistance based on the tooltip location. As the user controls the robot's tooltip using a haptic device they receive feedback based on its location to help direct them towards the optimal path. The force field does not control the robot directly and a human is kept in full control of the actual robot position. Therefore, the haptic device receives two inputs, one from the user and the other from the force field; this is shown in Fig. 4(b).

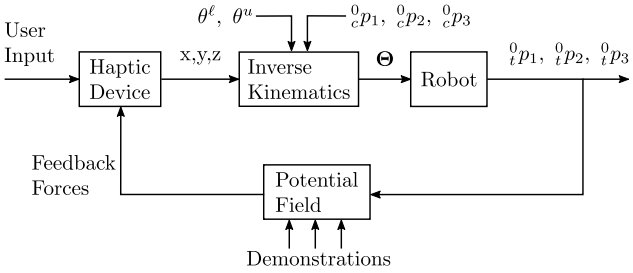
4. Experimental Validation

The experimental setup is shown in Fig. 4(a). The setup consists of a 3-DOF Novint Falcon, a 6-DOF robotic manipulator, the Meca500R from Mecademic (Montreal, CA), and a phantom kidney kindly provided by Marion Surgical. This setting does not reflect a surgical setting as the scope of this paper focuses on training and skills development in a cyber-physical environment. The modified Denavit-Hartenberg parameters used in the inverse kinematics can be found in Table 1 and the joint limits are summarized in Table 2.

The control loop used during experimental trials with haptic feedback can be seen in Fig. 4(b); for those trials without haptic feedback there is simply no feedback loop. During the user trials, the haptic device sends its 3D position to the computer which then processes it and uses it as



(a) Experimental set-up used to perform the user trials with the robotic arm, the haptic device and the phantom kidney



(b) Control loop used during experiments

Figure 4: Experimental setup (a), does not depict a clinical setting, rather represents the equipment and relative position used during the user trials including ① the haptic device, ② the robot, ③ the kidney model, ④ phantom kidney stones in the kidney model, ⑤ the tool used during experiments, ⑥ the tooltip, ⑦ the constrained point along the tool; (b) shows a general loop of how the system operates and provides haptic feedback to the user based on the robot's position.

the desired tooltip position in the robot's inverse kinematics. A demonstration of the procedure is done where the tooltip information is collected. This is done by collecting the joint angles of the manipulator during the procedure. The tooltip information is determined by applying the forward kinematics to the collected joint angles; this assumes that the tool is perfectly rigid. The data from the demonstration is used to generate the potential field that will be used to provide the haptic feedback. A user very familiar with the existing system and the ideal trajectory provides the demonstration. Once the potential field has been constructed the robot tooltip position is the input into the potential field function to get the required haptic assistance force to be applied to the haptic device. The computer uses an *Intel i7* processor and *GeForce RTX 2080* GPU. Communication between the computer, the haptic device, and the robot is handled by the Robot Operating System (ROS).

4.1. Joint Limit Experimental Validation

An experiment is performed to validate the effectiveness of the proposed constrained inverse kinematics. Two trials need to be conducted to observe the effects of the joint limits. The first only uses the default joint limits of the manipulator (see Table 2, *Default* rows), these limits are determined by the manufacturer based on avoiding

Table 2: Robot joint limits given in radians for the joint limit validation experiments

joint i	1	2	3	4	5	6
Default θ_i^ℓ	-3.05	-1.22	-2.36	-2.97	-2.01	$-\pi$
Default θ_i^u	3.05	1.57	1.22	2.97	2.01	π
Modified θ_i^ℓ	-1.00	-1.22	-2.36	-2.97	-2.01	$-\pi$
Modified θ_i^u	0.4	1.57	1.22	2.97	2.01	π

collisions or the rotational limit of the motors used when constructing the manipulator. During the second experiment, joint 1 is further limited (see Table 2, *Modified* rows) from -3.05 to 3.05 rad to -1 to 0.4 rad to demonstrate the response of the inverse kinematics as a particular joint approaches its limit.

The tooltip reference trajectory during these experiments is a path defined such that under the *Default* joint constraints the joints do not reach their limits. The trajectory can be found in the bottom right of Fig. 5.

Fig. 5 shows that as a joint approaches its limit, it begins to saturate. To achieve the desired position, the inverse kinematics finds an alternative joint configuration to compensate for the limited joint. It can be noted that the constrained and unconstrained angles are initially the same, and as time goes on the angles of each joint begin to change relative to the original; thus, compensating for the limitation imposed on joint 1.

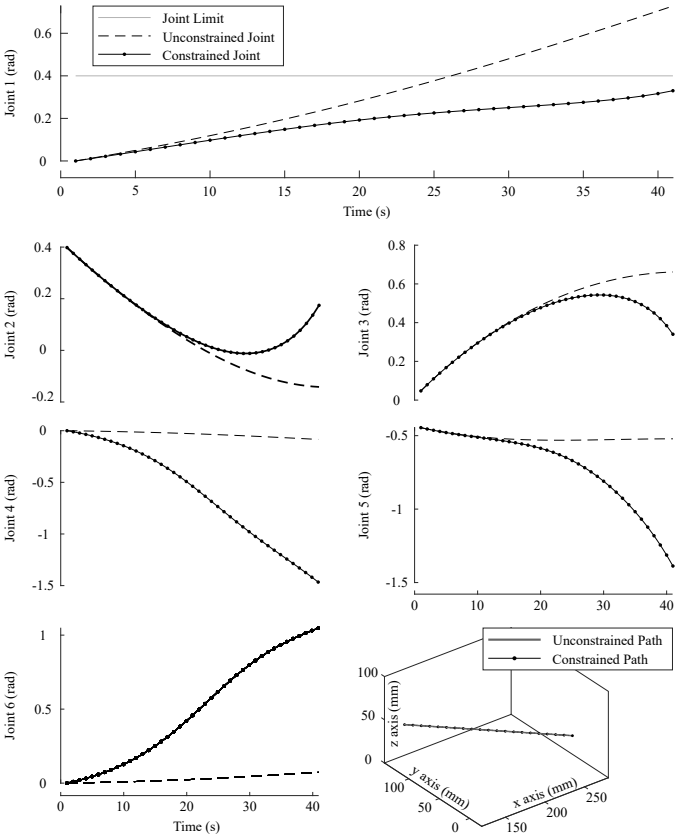


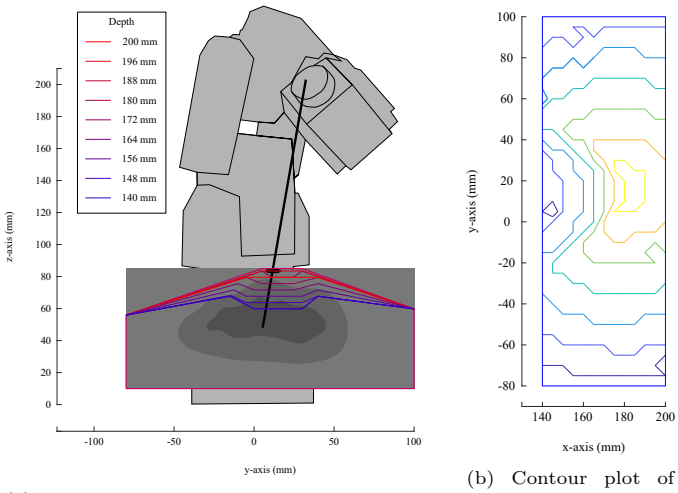
Figure 5: The results from the constrained inverse kinematics. In the 2D graphs, a dashed line depicts the joint angle when no additional joint constraints were added, a dotted line depicts the joint angles after an additional joint limit was applied to joint 1, and a solid grey line shows the added joint constraint. In the 3D plot the trajectory of the experiment is shown.

4.2. Constrained Workspace Evaluation

The second set of preliminary experiments evaluated the reachable workspace inside the kidney model. The constrained tooltip workspace was determined by considering the phantom kidney volume and determining which points within it are achievable when the RCM is set at the entry point. The previously validated inverse kinematics were used to attempt to reach points within the phantom kidney volume. This data was used to create the tooltip workspace shown in Fig. 6.

The workspace shown considers a tool-tip position in the robot's workspace when the angle between the tool and the surface of the kidney is no less than 20∘. All of the axis and coordinates are given relative the the robot's reference frame which considers the origin to be at the center of the robot's base.

As can be seen, most of the kidney volume can be accessed through the same entry point. The peak of the workspace (yellow region of the contour plot) is the location of the entry point. There is an additional concavity in the workspace behind the entry point; this is due to the kinematic constraints of the robot.



(a) tooltip constrained workspace, considering the phantom-kidney volume.

(b) Contour plot of the top surface of the workspace.

Figure 6: The above plots show the workspace of the constrained tool within the phantom kidney. On the left, plot (a), is a plot of the outline of the workspace (in red), with contours (ranging from red to blue) included for the top surface as it changes depending on depth. While plot (b) depicts the contour of the workspace surface in the x-y plane.

4.3. Experimental Scenarios for User Trials

Three different experiments were performed to validate the functionality of the system. They were done both

with and without haptic feedback. There were sixteen participants, 12 males, 4 females ranging in age from 18 to 31 with the average age being 24, are separated into 3 groups. The participants in each group were exclusive to that group and only provided data for their assigned group. Each participant completed 5 trials in total, the first 3 trials were training while the last 2 trials were used to evaluate the performance of each group. Every user completed the same scenario in each trial; the goal in each trial is to reach a particular kidney stone. The user who provided the demonstration data for the potential field was not a participant in the actual tests; they only provided the demonstration data.

Group 1 (6 participants): The first group performed all 5 trials without any haptic assistance. The data collected from the last two of these trials is the control data to compare the other groups against.

Group 2 (5 participants): The second group had haptic feedback during all 5 trials. This provided information on the effectiveness of haptic assistance when compared to those without it.

Group 3 (5 participants): Finally, the third group completed their 3 training rounds with the haptic assistance and then two evaluation trials without assistance. This was done to evaluate how well participants learned from the haptic assistance and to analyze if there were any lasting improvements in skill compared to those without haptic feedback.

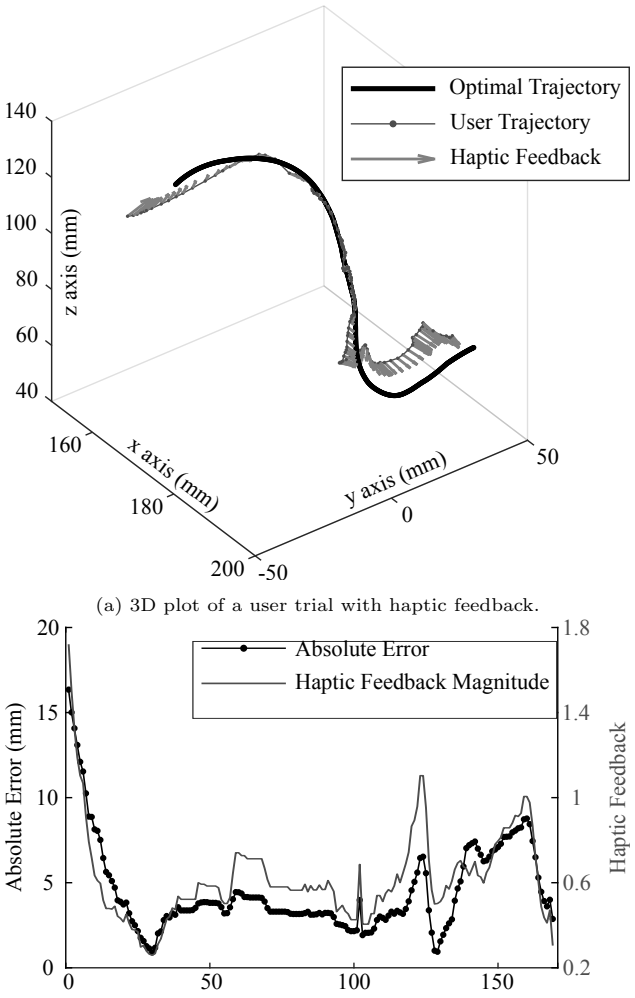
The stiffness of the potential field used to derive the haptic feedback was kept constant during all trials.

4.4. Experimental Procedure for User Trials

Users are expected to use the haptic device to control the six DOF robotic manipulator. A sheath located on the top surface of the kidney model is the entry point into the tissue ⑦ in Fig. 4(a). Users were requested to manipulate the tooltip through the sheath towards the phantom kidney stones ④ in Fig. 4(a). The procedure had 3 phases:

1. The tool was initially located outside of the kidney model. During this phase, the constrained point was just above the tooltip and was moved along with it such that the tool remains vertical. The 3D position of the haptic device controlled the position of the tooltip;
2. Once the user had positioned the tooltip inside the sheath, the user pressed a button and the tooltip location was recorded as the entry point in the tissue. The constrained point was then fixed and assigned the recorded entry point position.
3. The user further manoeuvred the tooltip towards the kidney stones. The 3D position of the haptic device controlled the 3D position of the tooltip while the constrained inverse kinematics ensured that the tool shaft passed through the constrained point.

An example of the haptic feedback provided to a user is shown in Fig. 7. In Fig. 7(a), the haptic feedback (grey



(a) 3D plot of a user trial with haptic feedback.

(b) Comparison between user's error from the ideal trajectory and the haptic feedback provided.

Figure 7: Plot (a) provides an example of haptic feedback provided to the user during one of their user trials; the optimal trajectory is shown as the solid black line, the user's trajectory is the dark grey dotted line, and the haptic feedback are light grey arrows. The bottom plot (b) exhibits the absolute error between the user's trajectory and the optimal path as the dotted black line (its y-axis is on the left), while the magnitude of the haptic feedback is given as a grey line (its y-axis is on the right).

arrows) is always directed towards the optimal path (the black solid line). Plot (b) shows the absolute error between the user's trajectory and the optimal path.

The magnitude of the applied haptic feedback was roughly proportional to the error. Due to how the potential field was constructed, with a slight pull along the optimal trajectory and varying spatial stiffness, the haptic feedback was similar to but not perfectly proportional to the user's error. When the user is in phase 1, moving towards the entry point, the haptic feedback provided should feel like a funnel due to the stiffness of the potential field being kept low and linearly increased up to the entry point. As the stiffness increases, larger force feedback is felt when the user diverges from the ideal path. At the entry point and through the flesh of the kidney model, the stiffness was kept constant; this is the highest stiffness used when generating the force feedback. While inside the kidney, phase 3, the feedback resembles a tube-like structure with relatively firm walls. The stiffness is linearly decreased somewhat once the tooltip has passed through the sheath, then the stiffness is constant inside the kidney.

4.5. Assessment Criteria for User Trials

Six different metrics are used to compare the experimental results between groups of participants. These metrics are largely based on those described in [31]. To begin, let the collected data points of the tooltip be $\xi_t \in \mathbb{R}^{3\times M}$ where M is the total number of collected data points for one trial, now the m^{th} column can be represented by:

$$\xi_t^m = \begin{bmatrix} {}^0_t p_1 & {}^0_t p_2 & {}^0_t p_3 \end{bmatrix}^T. \tag{26}$$

The 5 assessment criteria can now be defined as:

Criterion 1 is the time to complete task, t_{tot}, *i.e.*, the total time needed to reach the goal;

Criterion 2 is the distance the tooltip travels, *i.e.*, the path length ξ_L, given by:

$$\xi_L = \sum_{m=2}^{M} |\xi_t^m - \xi_t^{m-1}|, \tag{27}$$

where the magnitude of a vector $\left(\xi_t^m - \xi_t^{m-1}\right)$ is given as:

$$|\xi_t^m - \xi_t^{m-1}| = \sqrt{\left(\xi_t^m - \xi_t^{m-1}\right)^T \left(\xi_t^m - \xi_t^{m-1}\right)}. \tag{28}$$

The magnitude of the tooltip's velocity $\nu \in \mathbb{R}^{1\times M}$ at every point can be computed as the first time derivative of ξ_t by:

$$\nu_m = \left| \frac{\xi_t^{m+1} - \xi_t^{m-1}}{2\delta t} \right| \quad \forall m \in 1 \dots M, \tag{29}$$

where δt is the time step from one data-point to the next, assuming the time step is constant for all data-points. **Criterion 3** can now be defined as the average velocity ν_μ, i.e.:

$$\nu_\mu = \overline{\nu} = \frac{1}{M} \sum_{m=1}^{M} \nu_m. \tag{30}$$

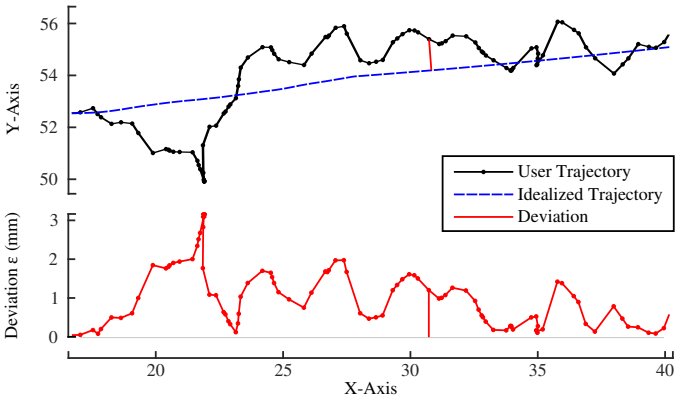


Figure 8: An example of a user's actual trajectory compared with their idealized trajectory. The dashed blue line is the idealized trajectory while the dotted black line is a user's actual trajectory during one trial.

The magnitude of motion acceleration $a \in \mathbb{R}^{1\times M}$ can be defined similar to velocity with the main difference being that acceleration requires the second time derivative of ξ_t as opposed to the first time derivative. Therefore, the vector of motion acceleration magnitudes can be computed by:

$$a_m = \left| \frac{\nu_{m+1} - \nu_{m-1}}{2\delta t} \right| \quad \forall m \in 1 \dots M, \tag{31}$$

Now, **Criterion 4** is defined as the mean acceleration, that is:

$$a_\mu = \overline{a} = \frac{1}{M} \sum_{m=1}^{M} a_m \tag{32}$$

Criterion 5 is related to the user's deviation from an idealized version of their trajectory. The idealized version ξ_I removes small redundancies and tremors leaving a close approximation to what the user intended to do. It is created by applying the Savitzky-Golay smoothing algorithm that fits a polynomial curve to sets of data points. An example of an idealized trajectory can be seen in Fig. 8, where the smooth blue line is the idealized trajectory based on the actual trajectory (the dotted black line).

Using the idealized version of the path the magnitude of the user's deviation $\varepsilon \in \mathbb{R}^{1\times M}$ for every point is defined as:

$$\varepsilon_m = \left| \xi_t^m - \xi_I^i \right| \quad \forall m \in 1 \dots M, \tag{33}$$

where ξ_I^i is the closest point along ξ_I. The average deviation is given by:

$$\varepsilon_\mu = \overline{\varepsilon} = \frac{1}{M} \sum_{m=1}^{M} \varepsilon_m. \tag{34}$$

Finally, **criterion 6** is the targeting accuracy for each user. This considers the approximate location of the phantom calculi ε_c and finds the closest data point from the user trial. As such the targeting accuracy TA can be defined as:

$$TA = \left| \varepsilon_c - \varepsilon_t^k \right|, \tag{35}$$

Table 3: Average and standard deviation for each assessment criterion and targeting accuracy (denoted TA)

	unit	Group 1	Group 2	Group 3
t_{tot}	sec	52.07 ± 28.09	32.98 ± 10.89	44.49 ± 16.83
ξ_L	mm	257.65 ± 64.92	178.32 ± 29.81	232.97 ± 58.84
ν_μ	mm/s	5.28 ± 1.91	5.32 ± 1.49	5.31 ± 2.01
a_μ	mm/s^2	22.58 ± 8.65	21.66 ± 5.12	24.53 ± 11.46
ε_μ	mm	1.63 ± 0.32	1.28 ± 0.33	1.48 ± 0.23
TA	mm	1.13 ± 3.03	2.22 ± 1.26	0.72 ± 0.94

where $\exists k \in M \ni \left|\varepsilon_c - \varepsilon_t^k\right| \leq |\varepsilon_c - \varepsilon_t^m| \, \forall m \in M$.

These criteria can now be used to evaluate the performance of the three different experimental groups to determine the effectiveness of the haptic feedback on performance, skill acquisition, and path consistency between experimental trials.

5. Results and Discussion

5.1. Experimental Results

The results from the last two trials of every user can be found in Fig. 9 and Fig. 10, as box-plots. At the top of Fig. 9 is the path length as determined by (27). Second from the top in the figure, is the total time to complete each trial. In the middle of the figure is (34) which is the average deviation per trial between the user's actual motion and the idealized version of their trajectory. Average velocity is the plot second from the bottom in Fig. 9, which was calculated in (30). At the bottom of the figure is the average acceleration (32) per trial. Lastly the targeting accuracy is shown in Fig. 10; the results are similar between each of the three groups. However, more consistent targeting was achieved by group 3, this suggests that the training with haptics leads to a much better understanding of the target position and the particular motion required to reach it. It should be noted that while the tool was considered to be rigid; it did bend when approaching the phantom calculi, which can lead to less accurate results when comparing the tooltip location to a static location in space. The average and standard deviation for each assessment criterion per group are summarized in Table 3. For each evaluation criteria the table contains the mean for the respective group along with its standard deviation.

5.2. Discussion

The experimental results obtained from a total of 32 trials and 3 independent groups reveal that, overall, users who receive haptic assistance perform better on average according to the assessment criteria specified earlier. The results further suggest that haptics is an effective way of developing surgical skills in the context of teleoperated PCNL.

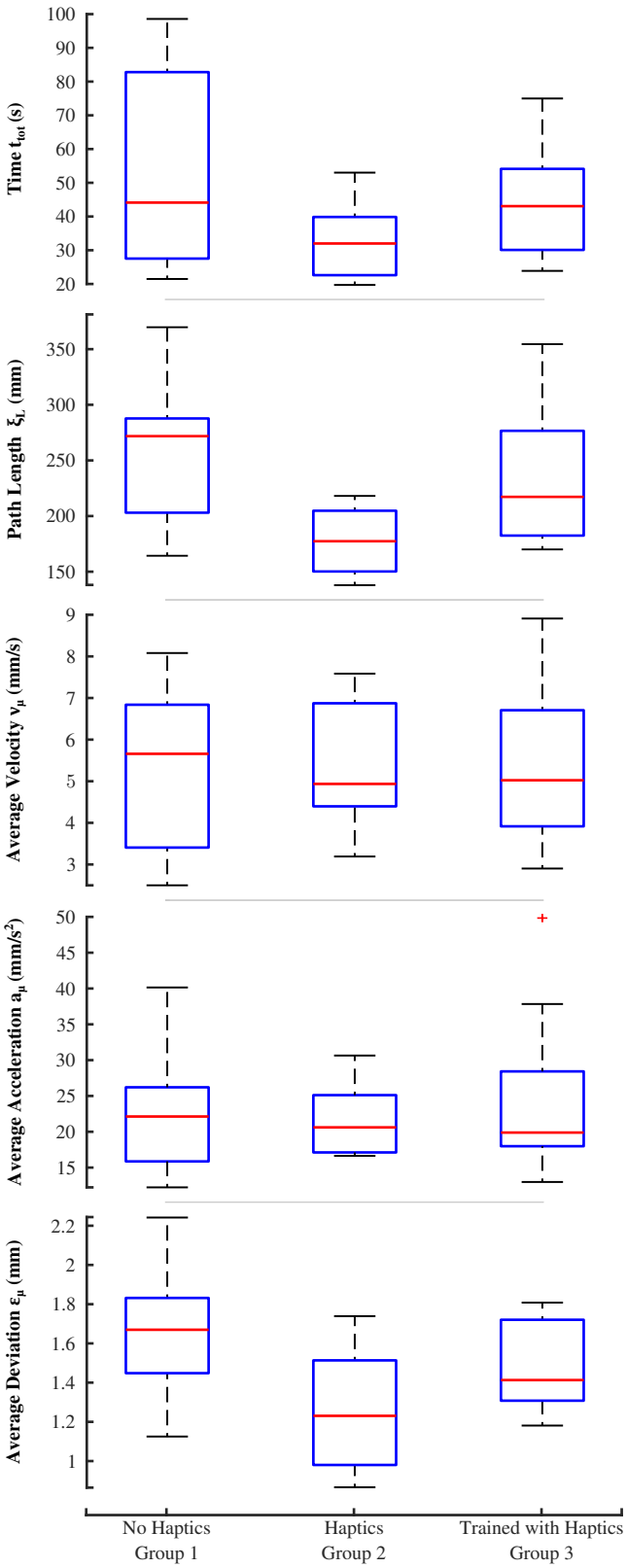


Figure 9: Box-plots of the experimental results for the last two trials of every user. The data is separated into the three groups of users, those who received no haptic assistance ('No Haptics'), those who completed it with haptic assistance ('Haptics') and those who completed their training rounds with haptic assistance but the evaluation rounds without haptic assistance ('Trained with Haptics').

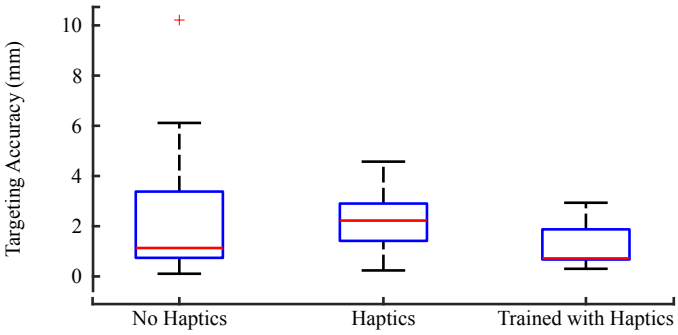


Figure 10: Box plots of the targeting accuracy for each group. Experimental data was separated into the 3 groups as defined in previous section. This plot indicates a similar targeting accuracy between each group.

The most evident benefit of haptic assistance can be seen in the time taken to complete the procedure as shown in Fig. 9. The addition of haptic feedback (group 2) made all trials more consistent when compared to those without haptic feedback (group 1). This claim is further supported by the fact that group 3 outperformed group 1. The results indicate that the subjects from group 3, who are trained with haptic feedback and are evaluated without it, did better than those who never experienced the haptic assistance. This suggests that group 3 demonstrates the degree to which users can learn from the haptic feedback.

Path length is another indicator of the effectiveness of the haptic assistance on user performance. Both the mean and the variability of the path length decrease with haptic assistance. Additionally, the lower median and variability of group 3 compared to group 1 is recurrent evidence of learning from the haptic assistance.

The mean deviation from the idealized path for each user trial follows the same trends as described for time and path length. A lower average deviation and range suggests that fewer redundant motions occurred during the procedure. It further implies that the path taken is smoother compared to those without haptic assistance or training (such as group 1). Finally, the average acceleration and velocity decrease with haptic assistance and training, which is another indication of smoother trajectories and motion during the procedure.

The targeting accuracy criterion indicates the effectiveness of learning from the haptic feedback. Once the haptic feedback was removed users were able to reach the phantom kidney stone with very consistent results. Group 2, who had haptics during every trial, did manage to target the kidney stone with slightly higher accuracy than the participants in group 1. Although participants in group 1 have a lower median targeting accuracy, they have a standard deviation larger than double the standard deviation of group 2, as can be seen in table 3.

6. Conclusion

Although PCNL is the leading procedure used for the management of urinary calculi for large and irregularly shaped urinary stones, it remains a challenging procedure to learn and perform efficiently. A fully automated system would pose an increased risk to patients due to unmodelled aspects of the surgical environment. In addition, they require accurate alignment of the optimal path to patient anatomy, [11] considers this issue and found they had a 15.8 mm position error and 4.12° orientation error. This paper introduces a collaborative human-robot teleoperative training framework to assist the surgeon and teach surgical skills. It has two integrant parts: The constrained inverse kinematics that decouples tooltip orientation and position using a remote centre of motion, and haptic assistance from past demonstration(s) based on data collected from the slave. This was experimentally validated with 16 participants and 80 trials.

Constraining the entry point in the kidney effectively allowed users to use a 3-DOF device for a procedure that required 6-DOF. This allows the workload to be shared between the surgeon and the robot while preserving the surgeon's control over the tool-path. This reduces the cognitive load on the surgeon during the procedure which makes it possible for them to be more receptive towards haptic feedback [32]. Experimental results obtained with haptic feedback consistently have shorter path lengths, shorter time for the procedure, a lower average deviation, and more consistent velocity and acceleration during the procedure. The group trained with haptic assistance performed better than those without, however, not as well as those who had haptic feedback for all trials. By most evaluation metrics they had a worse median value and larger variability; these problems were even more exaggerated in the group who never experienced any haptic feedback, thereby suggesting that haptic feedback reduces redundant motions during a procedure and can help teach inexperienced users.

Future work will explore the use of teleoperated PCNL under 2-dimensional ultrasound image guidance and semi-autonomous visual servoing for tool tracking. Further, this framework will be integrated with Marion Surgical's existing PCNL surgical simulation [10]. Simulation collected data from experienced surgeons will serve as the optimal tooltip path in the potential field functions for different simulated conditions to guide a teleoperated robotic arm and make it perform the surgery either autonomously or under teleoperated human supervision.

References

[1] J. de la Rosette, M. Laguna, J. Rassweiler, P. Conort, Training in percutaneous nephrolithotomy—a critical review, European urology 54 (5) (2008) 994–1003 (2008).

[2] C.-F. Ng, Training in percutaneous nephrolithotomy: The learning curve and options, Arab journal of urology 12 (1) (2014) 54–57 (2014).

[3] V. G. Bird, B. Fallon, H. N. Winfield, Practice patterns in the treatment of large renal stones, Journal of endourology 17 (6) (2003) 355–363 (2003).

[4] A. R. El-Nahas, A. A. Shokeir, A. M. El-Assmy, T. Mohsen, A. M. Shoma, I. Eraky, M. R. El-Kenawy, H. A. El-Kappany, Post-percutaneous nephrolithotomy extensive hemorrhage: A study of risk factors, The Journal of Urology 177 (2) (2007) 576–579 (2007).

[5] I. Kyriazis, V. Panagopoulos, P. Kallidonis, M. Ozsoy, M. Vasilas, E. Liatsikos, Complications in percutaneous nephrolithotomy.(report), World Journal of Urology 33 (8) (2015-08-01) 1069–1077 (2015-08-01).

[6] M. S. Michel, L. Trojan, J. J. Rassweiler, A. Breda, Complications in percutaneous nephrolithotomy, European Urology 51 (4) (2007-04) 899,906 (2007-04).

[7] S. Rais-Bahrami, J. I. Friedlander, B. D. Duty, Z. Okeke, A. D. Smith, Difficulties with access in percutaneous renal surgery, Therapeutic Advances in Urology 3 (2) (2011) 59–68 (2011). doi:10.1177/1756287211400661.

[8] C. Chu, S. Masic, M. Usawachintachit, W. Hu, W. Yang, M. Stoller, J. Li, T. Chi, Ultrasound-guided renal access for percutaneous nephrolithotomy: A description of three novel ultrasound-guided needle techniques, Journal of Endourology 30 (09 2015). doi:10.1089/end.2015.0185.

[9] A. Lelevé, T. McDaniel, C. Rossa, Haptic training simulation, Frontiers in Virtual Reality 1 (2020).

[10] B. Sainsbury, M. Lacki, M. Shahait, M. Goldenberg, A. Baghdadi, L. Cavuoto, J. Ren, M. Green, J. Lee, T. D'Averch, , C. Rossa, Evaluation of a virtual reality percutaneous nephrolithotomy (pcnl) surgical simulator, Frontiers in Robotics and AI 6:145 (2020). doi:10.3389/frobt.2019.00145.

[11] F. Ferraguti, M. Minelli, S. Farsoni, S. Bazzani, M. Bonfe, A. Vandanjon, S. Puliatti, G. Bianchi, C. Secchi, Augmented reality and robotic-assistance for percutaneous nephrolithotomy, IEEE robotics and automation letters 5 (3) (2020) 4556–4563 (2020).

[12] M. O'Malley, A. Gupta, M. Gen, Y. Li, Shared control in haptic systems for performance enhancement and training, Journal of Dynamic Systems, Measurement, and Control 128 (1) (2006) 75–85 (2006).

[13] C. Wagner, R. Howe, Mechanisms of performance enhancement with force feedback, in: First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics Conference, IEEE, 2005, pp. 21–29 (2005).

[14] L. Panait, E. Akkary, R. L. Bell, K. E. Roberts, S. J. Dudrick, A. J. Duffy, The role of haptic feedback in laparoscopic simulation training, Journal of Surgical Research 156 (2) (2009) 312–316 (2009).

[15] S. Botden, F. Torab, S. Buzink, J. Jakimowicz, The importance of haptic feedback in laparoscopic suturing training and the additive value of virtual reality simulation, Surgical endoscopy 22 (5) (2008) 1214–1222 (2008).

[16] J. J. Abbott, P. Marayong, A. M. Okamura, Haptic virtual fixtures for robot-assisted manipulation, in: S. Thrun, R. Brooks, H. Durrant-Whyte (Eds.), Robotics Research, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 49–64 (2007).

[17] A. Okamura, Methods for haptic feedback in teleoperated robot-assisted surgery, Industrial Robot: An International Journal 31 (6) (2004-12-01) 499,508 (2004-12-01).

[18] S. A. H. Perez, K. Harada, M. Mitsuishi, Haptic virtual fixtures to assist endonasal micro robotic surgery through virtual reality simulation, in: 2018 International Symposium on Micro-NanoMechatronics and Human Science (MHS), IEEE, 2018-12, pp. 1,3 (2018-12).

[19] B. Kent, A. Cusipag, C. Rossa, Tissue discrimination through force-feedback from impedance spectroscopy in robot-assisted surgery (Dec 2019).

[20] H. Su, W. Shang, G. Li, N. Patel, G. S. Fischer, An mri-guided telesurgery system using a fabry-perot interferometry force sensor and a pneumatic haptic device, Annals of Biomedical Engineering 45 (8) (2017) 1917–1928 (Aug 2017). doi:10.1007/s10439-017-1839-z.

[21] A. Chowriappa, R. Wirz, Y. W. Seo, A. Reddy, T. Kesavadas, P. Scott, K. Guru, A predictive model for haptic assistance in robot assisted trocar insertion, in: 2013 World Haptics Conference (WHC), IEEE, 2013-04, pp. 121,126 (2013-04).

[22] L. Xiong, C. B. Chng, C. K. Chui, P. Yu, Y. Li, Shared control of a medical robot with haptic guidance, International journal for computer assisted radiology and surgery 12 (1) (2016) 137–147 (2016).

[23] R. Munver, F. C. Delvecchio, G. E. Newman, G. M. Preminger, Critical analysis of supracostal access for percutaneous renal surgery, The Journal of urology 166 (4) (2001) 1242–1246 (2001).

[24] Y. Yang, S. Jiang, Z. Yang, W. Yuan, H. Dou, W. Wang, D. Zhang, Y. Bian, Design and analysis of a tendon-based computed tomography–compatible robot with remote center of motion for lung biopsy, Proceedings of the Institution of Mechanical Engineers. Part H, Journal of engineering in medicine 231 (4) (2017) 286–298 (2017).

[25] Y. He, P. Zhang, H. Jin, Y. Hu, J. Zhang, Type synthesis for remote center of motion mechanisms based on coupled motion of two degrees-of-freedom, Journal of mechanical design (1990) 138 (12) (2016).

[26] M. Hadavand, A. Mirbagheri, S. Behzadipour, F. Farahmand, A novel remote center of motion mechanism for the force-reflective master robot of haptic tele-surgery systems, The international journal of medical robotics + computer assisted surgery 10 (2) (2014) 129–139 (2014).

[27] K.-Y. Kim, H.-S. Song, J.-W. Suh, J.-J. Lee, A novel surgical manipulator with workspace-conversion ability for telesurgery, IEEE/ASME Transactions on Mechatronics 18 (1) (2013) 200–211 (2013).

[28] H. Sadeghian, F. Zokaei, S. Hadian Jazi, Constrained kinematic control in minimally invasive robotic surgery subject to remote center of motion constraint, Journal of intelligent & robotic systems 95 (3-4) (2018) 901–913 (2018).

[29] M. A. R. Garcia, R. A. Rojas, F. Pirri, Object-centered teleoperation of mobile manipulators with remote center of motion constraint, IEEE robotics and automation letters 4 (2) (2019) 1745–1752 (2019).

[30] S. Khansari-Zadeh, O. Khatib, Learning potential functions from human demonstrations with encapsulated dynamic and compliant behaviors, Autonomous Robots 41 (1) (2017-01) 45,69 (2017-01).

[31] K. Liang, Y. Xing, J. Li, S. Wang, A. Li, J. Li, Motion control skill assessment based on kinematic analysis of robotic end-effector movements, International Journal of Medical Robotics and Computer Assisted Surgery 14 (1) (2018-02) n/a,n/a (2018-02).

[32] M. Zhou, D. Jones, S. Schwaitzberg, C. Cao, Role of haptic feedback and cognitive load in surgical skill acquisition, Proceedings of the Human Factors and Ergonomics Society Annual Meeting 51 (11) (2007) 631–635 (2007). doi:10.1177/154193120705101106.