NATIONAL TECHNICAL UNIVERSITY OF ATHENS

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

SCHOOL OF MECHANICAL ENGINEERING

INTERDISCIPLINARY POSTGRADUATE PROGRAMME
“Translational Engineering in Health and Medicine”

Implementation of Trajectory-Based Attractive Virtual Fixtures for Surgical Knot Tying on a Simulated Surgical Robot

Postgraduate Diploma Thesis

Postgraduate student

Kanellos E. Theodoros Fanourios

Supervisor: **Tzafestas Costas**
Professor ECE NTUA

Athens, October 2025

NATIONAL TECHNICAL UNIVERSITY OF ATHENS

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

SCHOOL OF MECHANICAL ENGINEERING

INTERDISCIPLINARY POSTGRADUATE PROGRAMME

“Translational Engineering in Health and Medicine”

Implementation of Trajectory-Based Attractive Virtual Fixtures for Surgical Knot Tying on a Simulated Surgical Robot

Postgraduate Diploma Thesis

Postgraduate student

Kanellos E. Theodoros Fanourios

Supervisor: **Tzafestas Costas**
Professor ECE NTUA

The postgraduate diploma thesis has been approved by the examination committee on *29 October 2025*

1st member	*2nd member*	*3rd member*
Tzafestas Costas, Prof. *NTUA / School of ECE*	*Maragos Petros, Em. Prof.* *NTUA / School of ECE*	*Papadopoulos Evangelos, Em. Prof.* *NTUA / School of Mechanical Engineering*

Athens, October 2025

Kanellos E. Theodoros Fanourios

Graduate of the Interdisciplinary Postgraduate Programme,

“Translational Engineering in Health and Medicine”, Master of Science,

School of Electrical and Computer Engineering, National Technical University of Athens

The ideas and conclusions presented in this paper are the author’s and do not necessarily reflect the official views of the National Technical University of Athens.

Abstract

Robot-Assisted Surgery (RAS) constitutes the cornerstone of minimally invasive procedures, enhancing precision and dexterity. The present thesis focuses on the implementation of a ROS-Based attractive Virtual Fixtures (VFs) in a simulated version of the Da Vinci Research Kit (dVRK). For this implementation, V-REP (now named CoppeliaSim) has been used to support surgical knot tying tasks. The main objective is to develop a force-based guidance system that enhances instrument stability by dynamically applying virtual attractive forces to predefined surgical trajectories.

Our approach integrates surgeme recognition, to identify the closest predefined surgical paths and guide dVRK's robotic manipulators within optimal trajectories. Patient Side Manipulators (PSMs) are teleoperated using computer mouse and keyboard, allowing users to control robotic instruments. The integration of attractive forces results in dynamic deviation corrections when exceeding a proximity threshold. Furthermore, pulling the PSMs back into the intended knot_tying path while maintaining surgical precision. This method aims to assist operators in overcoming challenges related to hand-eye coordination, ensuring smoother execution of surgical tasks.

Our system calculates the closest surgeme, from the JIGSAWS dataset to the last tool position if it deviates over a predefined threshold, to guide dVRK's PSMs back on trajectory. PSMs are teleoperated using computer mouse and keyboard. The integration of attractive virtual forces results in corrective adjustments when deviation exceeds a proximity threshold, pulling the PSMs back toward the knot-tying trajectory. This thesis is an introduction to VFs applied on a simulated surgical robot and tries to contribute to the field of haptic assistance in robotic surgery.

Keywords: Robotic surgery, haptics, attractive virtual fixtures, active constraints, surgemes, dVRK, CoppeliaSim

Table of contents

Table of Figures

Chapter 1: Robot-Assisted Surgery (RAS)

The role of robots has expanded beyond task automation and autonomous planning to encompass increasingly collaborative roles in completing tasks alongside human operators and experts. We can see a futuristic visualization of a robot assisted procedure in Figure 1. This paradigm shift is particularly evident in the domain of medical robots; wherein, unlike typical industrial robots designed for manufacturing automation, the exclusion of human subjects is often not, if ever, an option. The human-centered nature of medical robot applications requires different sets of design specifications and safety considerations. Although control and planning techniques for human–robot collaboration is being developed by an increasing number of researchers, the interaction between humans, robots, and the environment and its feedback remains a challenge. In recent years, there has been a large amount of research providing visual feedback for human–robot collaboration. However, providing visual feedback alone will have a significant impact on tasks that require a "hands-on" approach, such as in surgery, where the "feel in the hand" of the surgeon is very important to the success or failure of the surgery. Therefore, it is necessary to provide force feedback along with visual feedback [1]. Robotic assistance allows surgeons to perform dexterous and tremor-free procedures, but robotic aid is still under-represented in procedures with constrained workspaces, such as deep brain neurosurgery and endonasal surgery. In these procedures, surgeons have restricted vision to areas near the surgical tooltips, which increases the risk of unexpected collisions between the shafts of the instruments and their surroundings [2].

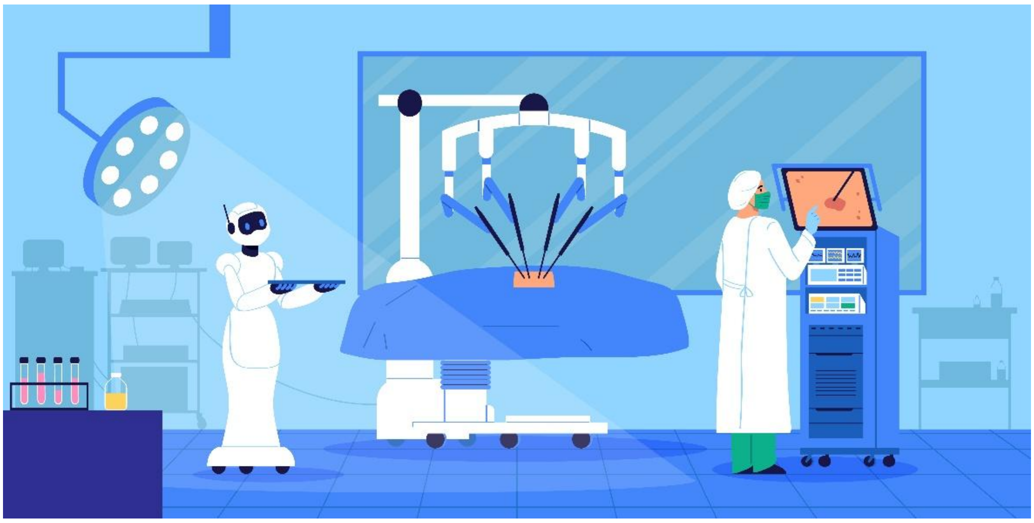


Figure 1: Envision of a Robotic OR

Today, many robots and robot enhancements are being researched and developed. Authors of [3] at Eberhard Karls University's section for minimally invasive surgery have developed a master-slave manipulator system that they call ARTEMIS. This system consists of 2 robotic arms that are controlled by a surgeon at a control console. Authors of [4] at the MiTech laboratory of Scuola Superiore Sant'Anna in Italy have developed a prototype miniature robotic system for computer-enhanced colonoscopy. This system provides the same functions as conventional colonoscopy systems, but it does so with an inchworm-like locomotion using vacuum suction. By allowing the endoscopist to teleoperate or directly supervise this endoscope and with the functional integration of endoscopic tools, they believe this system is not only feasible but may expand the applications of endoluminal diagnosis and surgery. Several other laboratories are designing and developing systems and models in minimally invasive surgery and combining visual servoing with haptic feedback for robot-assisted surgery [5]

Robotic systems such as Prodoc, ROBODOC, the AESOP system (Computer Motion Inc., Santa Barbara, CA), a voice-activated robotic endoscope, the comprehensive master-slave surgical robotic systems, Da Vinci (Intuitive Surgical Inc., Mountain View, CA) and Zeus (Computer Motion Inc., Santa Barbara, CA) have been commercially developed and approved by the FDA for general surgical use [5] .

The da Vinci and Zeus systems are similar in their capabilities but different in their approaches to robotic surgery. Both systems are comprehensive master-slave surgical robots with multiple arms operated remotely from a console with video assisted visualization and computer enhancement. In the da Vinci system, which evolved from the telepresence machines developed for NASA and the US Army, there are essentially 3 components: a vision cart that holds a dual light source and dual 3-chip cameras, a master console where the operating surgeon sits, and a moveable cart, where 2 instrument arms and the camera arm are mounted. The camera arm contains dual cameras and the image generated is 3-dimensional. The master console consists of an image processing computer that generates a true 3-dimensional image with depth of field; the view port where the surgeon views the image; foot pedals to control electrocautery, camera focus, instrument/camera arm clutches, and master control grips that drive the servant robotic arms at the patient's side. The instruments are cable driven and provide 7 degrees of freedom. This system displays its 3-dimensional image above the hands of the surgeon so that it gives the surgeon the illusion that the tips of the instruments are an extension of the control grips, thus giving the impression of being at the surgical site [5] .

1.2 Surgical Robotics Classification

Surgical robotic systems are often classified according to their level of autonomy into two categories. The first group, autonomous systems, execute specific tasks automatically with little intervention of the practitioner. Non-autonomous systems, on the other hand, aim to reproduce the surgeon's motion in either a master/slave teleoperated configuration or a hands-on configuration. Except for a few examples such as the Robodoc (Curexo Technology Corporation, CA, USA) [6] in total knee arthroplasty (TKA) or the Cyber-Knife system (Accuray, CA, USA) [7] in radiosurgery, most surgical robots belong to the second category. This is mainly due to substantial technical complications involved in executing automatic actions in medical interventions with the demanded reliability due to the criticality of the field. The current stage of autonomous execution of surgical subtasks is encouraging, but far from suitable for clinical implementation and replacing clinicians. Robots of non-autonomous type are not designed to replace the surgeon. Rather, they intend to augment capabilities of the medical staff and enable operations that would not otherwise be physically possible [8].

1.3 Advantages and Limitations of Current RAS systems

RAS has matured from a novel technological interest to a sophisticated clinical tool in many surgical disciplines. Its widespread adoption has been driven by tangible benefits in surgical performance, patient recovery, and ergonomics for the surgeon. However, despite all its state-of-the-art advancements, such systems carry inherent limitations, both technologically and practically, that need further research.

1.3.1 Advantages

- Enhance dexterity in several ways. Instruments with increased degrees of freedom greatly enhance the surgeon's ability to manipulate instruments and thus the tissues.
- Surgeons' tremors can be compensated for the end-effector motion through appropriate hardware and software filters. In addition, these systems can scale movements so that large movements of the control grips can be transformed into micromotions inside the patient [5]
- Restoration of proper hand-eye coordination and an ergonomic position. These robotic systems eliminate the fulcrum effect, making instrument manipulation more intuitive. With the surgeon sitting at a remote, ergonomically designed workstation, current systems also eliminate the need to twist and turn in awkward positions to move the instruments and visualize the monitor [5]
- Smaller incisions, important reduction of blood loss, postoperative pain, and hospital stay

1.3.2 Limitations

- Lack of tactile and force sensation, which can result in excessive tissue force application
- High Cost, their cost is nearly prohibitive
- Robotic surgery is a relatively new technology, and its uses and efficacy have not yet been well established. Mostly studies of feasibility have been conducted, and almost no long-term follow-up studies have been performed [5]

Chapter 2: Haptics in RAS

2.1 Introduction

Haptic technology has shown the potential to restore tangibility to human–computer interaction. Despite substantial technological challenges, the spread of haptically active devices in numerous fields appears to be inevitable. In addition to its mainstream applications, haptic feedback has been employed to transform several specialized tasks by providing an auxiliary sensory channel in addition to the often overly burdened visual and auditory channels. The research efforts on specialized tasks have so far been dominated by surgical robotics and simulation [8].

Vision and haptic are the principal sensory inputs employed by humans in object manipulation tasks. While visual display technology has reached an advanced stage, haptic feedback has remained rather underutilized because of practical challenges such as control loop stability. The lack of haptic feedback in a teleoperated system forces the surgeon to depend merely on visual cues, such as the deformation of tissue under load, to estimate the forces [8]. The likely outcome of misreading these cues is torn tissue or broken suture [9].

The objective in providing force information to the operator is often to enhance the telemanipulation experience so that the operator feels as if he/she were present at the remote site. This is commonly referred to as transparency of haptic interfaces [10]. However, haptic technology aims to go even further than providing a realistic illusion of telepresence, by providing the possibility of interacting with operative information via virtually generated force constraints [11]. An active constraint (also known as VF) [12] is generated by software and attempts to encourage a user's movement along desired paths or prevent he/she from moving into forbidden regions. These methods allow one to exploit the precision of robotic systems while keeping the operator in charge [8].

Haptic technology has had yet another impact on medical community, which is that of computer simulations for surgical training systems [13], [14].

2.2 Fundamentals of Haptics

The word haptics is derived from the Greek word “haptikos,” which refers to “a sense of touch,” and the Oxford English dictionary defines haptics as: “Relating to the sense of touch, in particular relating to the perception and manipulation of objects using the senses of touch and proprioception”. “Haptics” is a broad construct that encompasses a number of different sensory inputs arising from receptors embedded within the body. These receptors can provide information about the state of the body known as kinesthetic sensation (i.e., the angles formed by the various joints within the arm), and the physical characteristics of objects within the world, known as tactile sensations. In contrast to vision, haptic information regarding an object’s physical properties requires the body to come into contact with the object. Mechanical interactions between an object and the body provide a rich source of information that is not readily available from vision, and the usefulness of this information to humans is a matter of common observation [15]

The scientific examination of haptics started in the 1800s and this type of investigation can be attributed directly to two of the founding fathers of experimental psychology, Ernst Weber and his student Fechner [16]. Since the time of Weber, psychophysics has highlighted the remarkable performance of human tactile sensation; the temporal resolution of touch is approximately 5 ms and the spatial resolution at the fingertip as low as 0.5 mm [17]. This spatial and temporal resolution of touch provides humans with rich data about the physical characteristics of objects contacted by the hand. Haptic information comes in part through mechanoreceptors (nerve endings which react to a mechanical stimulus), and a variety of different mechanoreceptor types are found across the body. These receptors provide specialized somatosensory information input (information regarding haptics) to the central nervous system (CNS). The mechanoreceptors work through changes in the physical properties of their plasma membranes. The rate of adaptation and threshold of activation varies according to the mechanoreceptor type. Human skin can be considered to contain four types of mechanoreceptors that are specialized to provide tactile input to the CNS (Meissner's corpuscles, Pacinian corpuscles, Merkel's disks, and Ruffini's corpuscles). These high-sensitivity/low-threshold receptors can be found on glabrous (hairless) skin and the hand. Fingertips have an intense distribution of low-threshold receptors and are thus one of the areas of the human body most sensitive to external contact with objects [18], [19]. We have already drawn attention to the fact that the term "haptics" is also used to refer to the sensory system that provides the body with information about the relative position of body parts. This class of information relates to the sense of position and movement of the limbs in space and is often described as "kinesthesia" [20]. The source of this information arises from mechanoreceptors embedded in muscles, tendons, and joints. These receptors allow humans to sense the angular position of a joint and have a fundamental role in the control of human movement [15].

Today, the study of haptics transcends philosophy and psychology, bringing together a diverse range of disciplines, from neurophysiology through to computer science. Haptics has become a "broad church," covering the study of human and/or computer interactions with the environment and most often related to the manipulation of objects through touch and force feedback. A large degree of research activity in haptics has focused on practical solutions, from supporting activities of daily living through to the incorporation of haptics in surgical technologies. The optimization of robotic surgery will require further practical consideration of how information can be best extracted and used to support the high levels of accuracy and precision required when physically interacting with tissue in surgery [15].

2.3 Haptic Modalities in Surgical Robotics

Minimally Invasive Surgery (MIS), also known as laparoscopic or keyhole surgery, is an operation of performing surgery through small incisions with specially designed slender instruments and endoscopic cameras. Compared to open surgery, MIS aims to minimize invasiveness, which results in distinct advantages such as reduction of tissue trauma, intraoperative blood loss, postoperative infection risk, patient pain, aesthetic aspects, and recovery time. Nevertheless, there are some drawbacks, namely counterintuitive motion of the instrument (that pivots around the incision point), deteriorated vision, and depriving the surgeon of direct haptic sensation. In some cases, such as colorectal surgery, MIS has been associated with an even higher intraoperative complication rate than open surgery. These drawbacks motivated the introduction of RMIS with promising advantages such as helping the surgeon regain dexterity, optimal hand–eye alignment, motion scaling, and tremor filtering. While there is no consensus on benefits of RMIS for all surgical procedures, many studies report positive results [8].

The majority of studies in the literature report that the absence of haptic feedback in MIS can be a cause of error and, therefore, a possible safety concern. In study [21], a modified observation-based human reliability analysis approach, was adopted to categorize errors encountered during the practice of minimally invasive cholecystectomy. Most of the errors recorded were errors associated with the motor control of instruments where an inappropriate level of force was applied by the surgeon because of the diminished haptic feedback. It must be noted that the role of haptic feedback in RMIS can be even more significant if the visual feedback is deteriorated during the RMIS operation, for example, when the camera's view is clouded by fluids or by the smoke generated from the electrosurgical hook operations [8].

Chapter 3: Active Constraints in Robotic Systems

3.1 Active Constraints

During robot manipulation of hands-on systems intraoperatively, the robotic platform can provide significant levels of support to the operator. One of the ways of assisting a surgeon is the concept of "active constraints." This term can be used to describe control algorithms that provide some degree of assistance to the human operator during robotic manipulation. Specifically, the process of establishing and applying an active constraint comprises two steps, namely the constraint definition and the constraint enforcement. Active constraint definition is the process of generating a geometry with a fully defined boundary in the workspace of a robot. Once the constraint geometry has successfully been established in the robot workspace, it is enforced upon the robot. This is achieved by identifying the configuration of the robot end-effector with respect to the constraint boundary and subsequently influencing the motion of the human operator [12], or generating other robot responses, such as speed reduction or even deactivation [22].

Active constraint enforcement can usually be implemented through haptic feedback. In such cases, a physical reaction from the robot is communicated to the operator. Haptic feedback may be categorized into two groups.

- The first haptic feedback category is force feedback, which involves the application of a force from the robotic platform to the operator [22].
- The second category is tactile feedback, most notably implemented in the form of vibrations produced by the robot and transmitted to the operator's hand [22].

Of the two categories, force feedback has received significantly more attention [23]. Some developments can also be noted in visual feedback techniques, where the operator is able to better visualize the position of the robot with respect to either the patient tissue or the predefined boundary through external monitors and augmented reality systems. External monitors may allow for the visualization of important parameters, such as the force that a haptic system would theoretically apply to the operator or other visual clues [24]. Similarly, augmented reality solutions allow for the superimposition of virtual objects on the operated patient [25], thus allowing for more detailed visualization of the operated anatomy [22].

Substantial research has been undertaken on the topic of active constraints, and a general summary is available [12]. Examining the process of active constraint definition, it is generally assumed that the constraint geometry has been defined *a priori*, or can be defined using some generated point-cloud [26], [27]. Another frequent implementation involves the generation and combination of primitive shapes (e.g. spheres, cylinders etc.). These primitive shapes can then be combined to generate a more complex active constraint geometry [28]. It should be noted, however, that when exploring the concept of active constraint definition, generating the constraint geometry alone is not sufficient. The generated active constraint boundary needs to be “anchored” to the operated patient tissue. In doing so, any registered robotic system becomes capable to localise the generated geometry [22].

Having understood the process of deploying a collaborative surgical robot in the operating room and registering it to the patient, as well as the process of imposing an active constraint, it is also important to understand the accuracy achieved by such systems. These values can be used as a reference point from any novel robotic platforms. In the sector of orthopedics, collaborative, hands-on surgical robotic platforms deployed for cutting, as opposed to robots that support jig placement, can be arranged into two categories, depending on the type of support the robot provides to the operator who is manipulating the robot using their hands. These categories are haptic robots, and boundary control robots. The function of haptic feedback has been outlined in this section. Boundary control robots are similar to haptic feedback systems in the sense that an active constraint boundary across the patient geometry is established. However, instead of the robot providing haptic feedback to the operator upon interaction with the boundary, the robot instead partially or completely interrupts the function of the surgical tool used for the operation upon interaction with the boundary [22].

3.2 Dynamic Active Constraints

Dynamic active constraints are those where the constraint geometry moves continuously, because of changes in the physical environment or task being undertaken [12].

An abstract investigation of dynamic active constraints was carried out using simple proximity-based constraints by authors of [29]. They constructed an experiment where a linear actuator was used to move a soft-tissue phantom in 1 DOF, while a human user attempted to depress the phantom by a fixed amount using a teleoperated robot. By constructing a regional dynamic active constraint, which followed the tissue phantom at the required tissue depth, the user was assisted while the tissue was moved periodically and randomly. Authors of [29] used two methods for computing the necessary position of the dynamic constraint; one based on the current tissue position, and one based on its predicted position. They found that the two methods were comparable, and both were significant improvements on unconstrained teleoperation and static constraints [12].

Much of the remaining dynamic active constraint research has focused on applications for surgery on a beating heart. Authors of [30], considered the heart's left ventricle and generated multidimensional dynamic active constraints based on a simple proximity function. They constrained the end effector of a simulated surgical tool to follow a “dynamic guidance curve” between the inner walls of a beating heart which were generated in real time. They used a haptic master device to render a constraint force to the user. The results of experiments with this system show that off-path error was reduced compared with cases with only visual guidance and cases without any guidance [12].

Potential fields were used by authors in [31] to constrain a simulated surgical tool to the heart's external surface. In their method, a series of preoperative images are taken of the heart throughout its pulsatile cycle. Each 3-D time slice is segmented to extract the geometry of the surface of the heart, and image dilation is used to create an explicit distance map from the heart's surface. The distance map is then used to train a B-spline-based artificial neural network, as described previously, allowing for highly efficient real-time evaluation. Intraoperatively, the time slices are synchronized with the patient using electrocardiography and the controller uses the neural network to efficiently establish the distance of the CTG from the closest point on the heart's surface. Values for the potential are computed from the proximity values using either a generalized sigmoid function (for regional constraints) or a Gaussian function (for guidance constraints), allowing the constraint forces to be computed in the standard way [12].

In [32], the properties of the haptic interaction between a user and a dynamic active constraint were considered. As the boundary of a dynamic active constraint can move, it is possible for it to move past a static tool so that it switches from unconstrained to constrained regions. When this happens, conventional constraint enforcement methods can lead to undesirable active tool motion. In this research, a dynamic constraint based on a friction model is proposed, which remains passive and avoids autonomous motion [12].

Chapter 4: Haptic Assistance using VFs

4.1 Introduction

VFs are task-dependent computer-generated constraints that limit the robot's movement into restricted regions and/or influence its movement along desired paths. The goal of the VF algorithm is to generate a sequence of incremental motion commands for the robot such that certain task-specific constraints are satisfied. To keep the system intuitive, the incremental motion should be proportional to the user input. For a surgical assistant robot, it is very important to be able to place absolute bounds on the spatial motion of the different parts of the instrument [33].

Often, surgical robots are designed to be kinematically redundant for providing dexterous assistance. At the same time, certain tasks such as passing a tool through a cavity place have certain requirements and constraints on the robot motion and restrict dexterity. Indeed, some special purpose designs for minimally invasive surgery, such as the IBM LARS and the JHU Steady Hand robot provide such constraints through mechanism design. Other robots such as the Intuitive daVinci and Endorobotics combine a kinematically constrained remote center of motion (RCM) mechanism with a kinematically redundant wrist. Thus, it is important for the robot control algorithm to be able to accommodate unique, special purpose mechanical designs (such as kinematically redundant or deficient mechanisms). In this section we present an overview of constrained optimization approach, followed by certain approximations that allow us to execute the algorithm in real-time [33].

VFs show great promise for tasks that require better-than human levels of accuracy and precision, but also require the intelligence provided by a human directly in the control loop. Human-machine manipulation systems make up for many of the shortcomings of autonomous robots (e.g., limitations in artificial intelligence, sensor-data interpretation, and environment modeling), but the performance of such systems is still fundamentally constrained by human capabilities. VFs, on the other hand, provide an excellent balance between autonomy and direct human control. VFs can act as safety constraints by keeping the manipulator from entering potentially dangerous regions of the workspace, or as macros that assist a human user in carrying out a structured task. Applications for VFs include robot-assisted surgery, difficult assembly tasks, and inspection and manipulation tasks in dangerous environments [34].

VFs can be applied to two types of human-machine robotic manipulation systems: cooperative manipulators and telemanipulators. In cooperative manipulation, humans use a robotic device to directly manipulate the environment. In telemanipulation, a human operator manipulates a master robotic device, and a remote slave robot manipulates an environment while following the commands of the master [34].

4.2 Attractive/Repulsive VFs

The way in which active constraints act is considered in the literature as either "attractive" (encourages motion toward the constraint) or "repulsive" (encourages motion away from the constraint). The difference between these two types of constraint is best exemplified by considering a unilateral (one-sided), regional active constraint. If an attractive constraint was used, then the user's motion would only be modulated once the tool had already passed into the restricted region, whereas for a repulsive constraint the modulation would come into effect before the tool reached the constraint. As with regional/guidance constraints, the difference between attractive and repulsive constraints is largely superficial rather than inherent in control. This similarity is seen when considering an attractive constraint that encourages the user toward a linear pathway. This can, in fact, act identically to a large repulsive cylindrical constraint, which has been centered on the pathway [12].

Chapter 5: Implementation and Experiments

5.1 Introduction

In the previous chapters, we established the theoretical background of RAS, haptic feedback, and VFs. In this chapter we pass from theory to practice by documenting the system development and experimental verification that constitutes the core contribution of this thesis. The primary objective is to create and evaluate a real-time active constraint framework that employs visible surgical trajectories to improve task accuracy and safety during robotic suturing.

The implementation was done using the simulated da Vinci Research Kit (dVRK) platform, on the CoppeliaSim simulator through a modular Python–ROS (Robot Operating System) framework. Dvrk's patient-side manipulators (PSMs) were kinematically modeled, using forward and inverse kinematic formulas to be controlled using mouse and keyboard. The use of CoppeliaSim allowed rapid prototyping and visualization of dual arm knot-tying task, while the ROS middleware facilitated real-time communication between simulation, trajectory generation, and constraint enforcement nodes.

One of the key features of this study is the integration of surgemes, trajectories from the JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS) dataset [35]. JIGSAWS provides labeled motion primitives for canonical tasks such as suturing, knot tying, and needle passing recorded from the da Vinci system. By integrating surgemes into the simulation, reference trajectories were defined to serve as virtual guides.

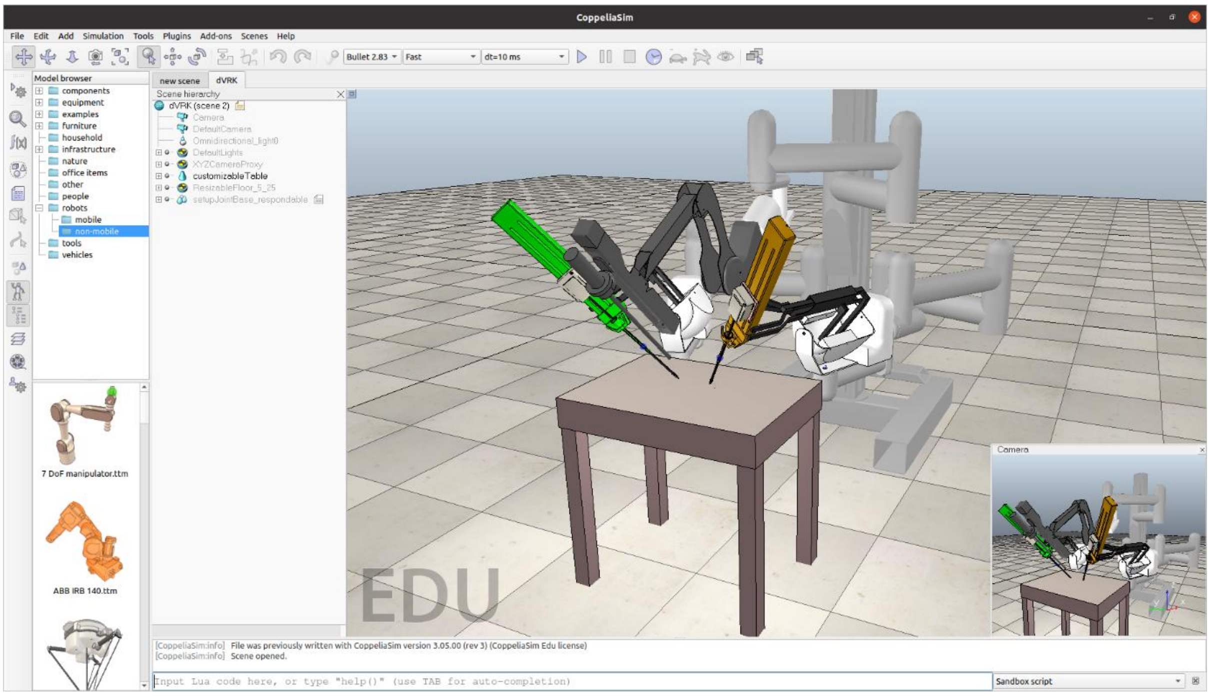
The fundamental control notion is the activation of an active constraint force whenever the tool tip varies beyond a preset threshold from the reference surgeme path. When deviation exceeds threshold, a corrective force is computed based on attractive principles, driving the tool tip back toward the correct point of the trajectory. The VF therefore functions as a dynamic safety boundary, unnoticed if the instrument is close to the correct path, but corrective if deviation can cause task failure.

5.2 System Architecture

5.2.1 Simulation Process

CoppeliaSim was selected as the simulation environment due to its high flexibility in robotics prototyping, built-in physics engine, and direct ROS integration using dynamic topics. Two PSMs (PSM1, PSM2) were employed. The simulator continuously streams joint states and tool-tip joint poses over ROS topics, providing real-time feedback.

CoppeliaSim's dVRK simulation as seen in Fig. 2, was used as a point of reference to ensure compatibility with existing interfaces. ROS topics were organized using publishers to control desired joint states and poses, and subscribers to receive feedback from simulated PSMs.



```
[CoppeliaSim:info]  File was previously written with CoppeliaSim version 3.05.00 (rev 3) (CoppeliaSim Edu license)
[CoppeliaSim:info]  Scene opened.
```

Figure 2: CoppeliaSim Environment with dVRK scene implemented

5.2.2 ROS Node Architecture

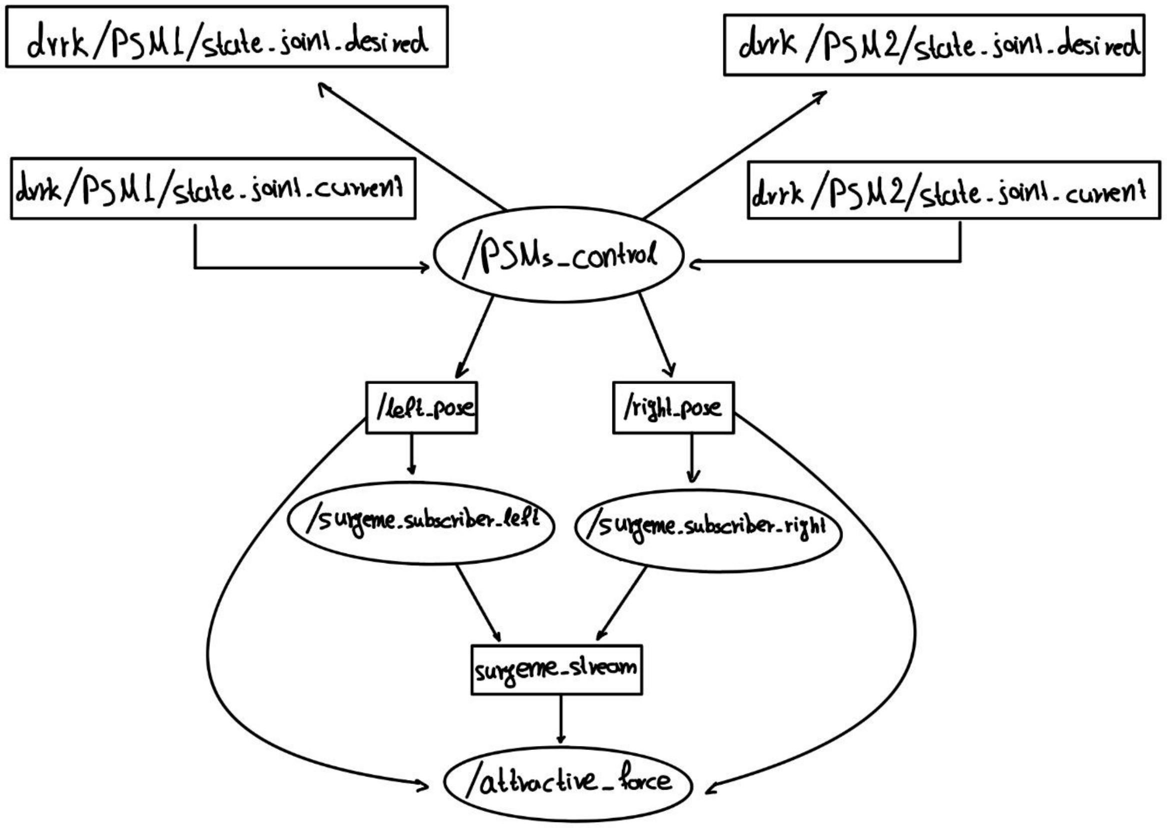

dvrk/PSM1/state.joint.desired
dvrk/PSM2/state.joint.desired
dvrk/PSM1/state.joint.current
dvrk/PSM2/state.joint.current
/PSMs_control
/left_pose
/right_pose
/surgeme.subscriber_left
/surgeme.subscriber_right
surgeme_stream
/attractive_force

Figure 3: ROS node architecture

The architecture is structured into three independent nodes as shown in Fig.3:

- **PSMs Control Node**:
  - Kinematic teleoperation using mouse and keyboard.
  - Contains Subscribers to topics */dvrk/PSM1/state_joint_current* and */dvrk/PSM2/state_joint_current*, to get joint values (θ_1, θ_2, d_3) of each PSM and Publishers to topics */dvrk/PSM1/state_joint_desired* and */dvrk/PSM1/state_joint_desired* for moving each PSM's tool tip to the desire pose.
  - Computes forward and inverse kinematics of both PSM arms. Transforms joint states into Cartesian poses for Cartesian control using mouse and keyboard and vice versa to publish to dVRK topics. Note that dVRK's topics listens and publishes only in joint states.
  - Visualizes using Rviz the PSMs
- **Surgeme Subscriber Nodes (left and right)**:
  - Loads surgeme trajectories from the JIGSAWS database.
  - Create a dictionary to save 10 surgemes based on the txt file.
  - Preprocesses them (resample, scale) and publishes knot tying trajectories using ROS topics.
  - Communicate with PSMs control node and dynamically calculate Euclidean distance between PSMs tool tip position and every point of each surgeme to find the closest (active) surgeme and activate also for visualization puproses the next one.
  - Visualize using Rviz, active and upcoming surgemes, and current point position of each tool on trajectories.
- **Attractive Force Node**:
  - Communicates with PSMs control node to track PSMs position
  - Computes the deviation using Euclidean distance based on the idea of last valid position. More specifically the node, computes distance between left or right tool tip position which is the current pose of each PSM and the last surgeme point from which tool deviated over the accepted threshold.
  - Creates a correction force vector and updates tool new position.
  - Logs the new pose

5.3 PSM arm kinematics

Each PSM as shown in Fig. 4 is a 7-DoF actuated arm, which moves a surgical instrument about a Remote Center of Motion (RCM), i.e., a fixed fulcrum point that is invariant with respect to the configuration of the PSM joints. The first 6 DoFs correspond to Revolute (R) or Prismatic (P) joints, combined in a RRPRRR sequence. The last DoF corresponds to the opening and closing motion of the gripper, with respect to the base frame BP: $\{O_{bp}; x_{bp}, y_{bp}, z_{bp}\}$ [36]. In this experiment, we use only the first three RRP joints received from the dVRK ROS topics which are θ_1, θ_2 and d_3 as shown in Fig. 5.

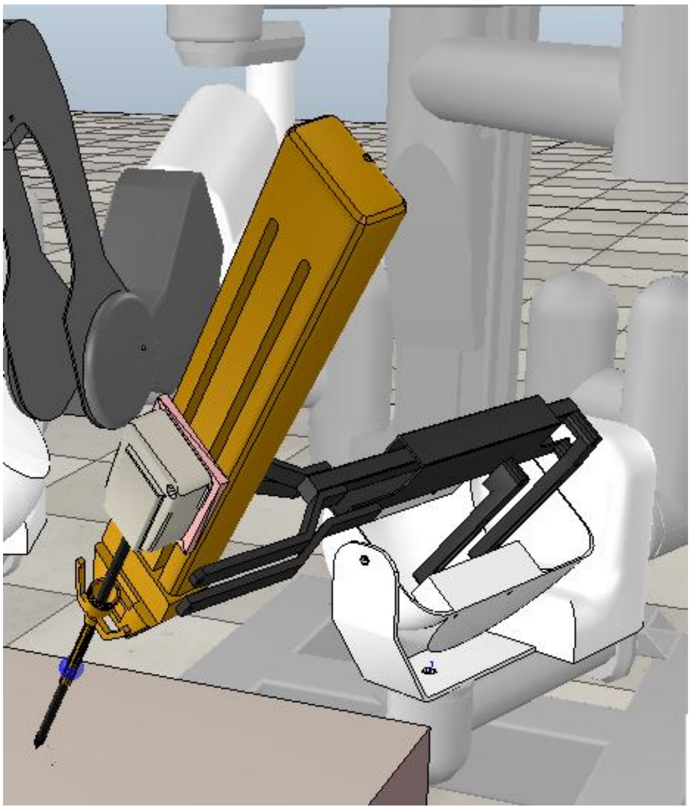


Figure 4: Left PSM arm

5.3.1 Forward Kinematics

Forward kinematics maps a set of joint values q= [θ_1, θ_2, d_3] as shown in Fig.5, given from dVRK topic to a Cartesian position x = [x, y, z] in order to use Cartesian control via mouse and keyboard. In the simplified PSM model, θ_1 and θ_2 represents Revolute (R), and d_3 Prismatic (P) joints. The tool-tip position is obtained as:

$$r = d_3 * cos(\theta_2) \qquad (1)$$

$$x = r * cos(\theta_1) \qquad (2)$$

$$y = r * sin(\theta_2) \qquad (3)$$

$$z = d_3 * sin(\theta_2) \qquad (4)$$

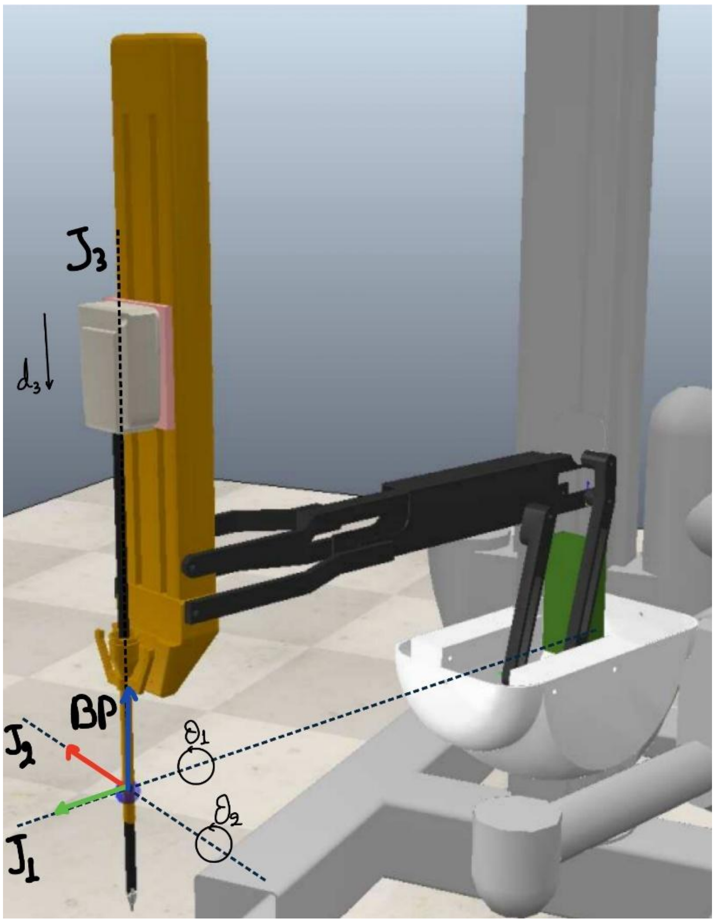


Figure 5: Left PSM kinematics

Orientation was approximated using elementary rotation matrices:

$$R = R_x(\theta_2) * R_y(\theta_1) \quad (5)$$

from which Euler angles and quaternions were extracted. Although this does not reproduce the full dVRK wrist kinematics, it provides sufficient orientation data to visualize tool motion.

5.3.2 Inverse Kinematics (IK)

IK computes the desired joint position based on Cartesian control to publish them to Dvrk topic:

$$r = \sqrt{x^2 + y^2} \quad (6)$$

$$\theta_1 = atan2(y,x) \quad (7)$$

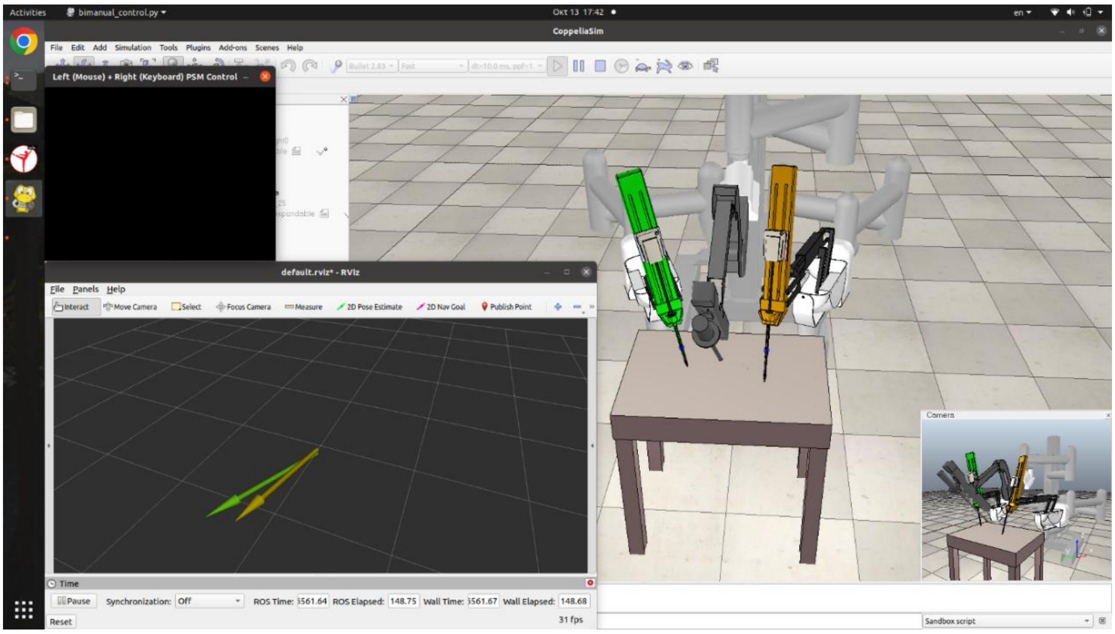
$$\theta_2 = atan2(z,r) \quad (8)$$

$$d_3 = \sqrt{x^2 + y^2 + z^2} \quad (9)$$

5.4 Mouse–Keyboard Teleoperation

Without the physical da Vinci console or haptic devices, a useful teleoperation interface was created implemented by Pygame to enable simultaneous control of both PSMs using common peripherals. The left patient-side manipulator (PSM1) is controlled with the mouse and the right manipulator (PSM2) with the keyboard as shown in Fig.6. This division of input channels enabled each hand to control a tool.

- Left PSM (mouse control):
  - Movement on the x and y plane was controlled by dragging the mouse with the left button down.
  - Scroll wheel events caused increase or decrease in the displacement along the z-axis, mimicking insertion and retraction movements.
- Right PSM (keyboard control):
  - Q/E controlled motion along the x-axis,
  - A/D controlled motion along the y-axis
  - W/S keys produced positive and negative displacement along the z-axis.



Synchronization: Off ROS Time: 5561.64 ROS Elapsed: 148.75 Wall Time: 5561.67 Wall Elapsed: 148.68

Figure 6: Left and Right PSMs control via Pygame (up left) and visualization using Rviz (bottom left)

5.6 JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS)

5.6.1 Language of Surgery

The authors in [35] considered that surgical motion is analogous to human language because it is a composition of elementary activities that are sequentially performed with certain constraints. Consequently, surgical motion can be modeled using techniques that have successfully been applied for analyzing human language and speech. They defined the Language of Surgery as a systematic description of surgical activities or proceedings in terms of constituents and rules of composition.

The overall goals for the project was to establish archives of surgical motion datasets, procedures and protocols to curate and securely store and share the datasets, develop and evaluate models to analyze surgical motion data, develop applications that use our models for teaching and assessing skillful motion to surgical trainees, and conduct research towards human machine collaboration in surgery. Surgical activity data may be considered to encompass several types of variables related to human activity during surgery such as surgical tool motion (kinematics), video, log of events happening within and beyond the surgical field, and other variables such as surgeon's posture, speech, or manual annotations.

5.4.2 Surgemes

A key feature of the JIGSAWS dataset is the manually annotated ground-truth for atomic surgical activity segments called "gestures" or "surgemes". A surgical gesture is defined as an atomic unit of intentional surgical activity resulting in a perceivable and meaningful outcome. The authors in [35] specified a common vocabulary comprised of 15 elements as shown in Table 1, to describe gestures for all three tasks in the dataset through consultation with an experienced cardiac surgeon with an established robotic surgical practice.

Gesture index	Gesture Description
G1	Reaching for needle with right hand
G2	Positioning needle
G3	Pushing needle through tissue
G4	Transferring needle from left to right
G5	Moving to center with needle in grip
G6	Pulling suture with left hand
G7	Pulling suture with right hand
G8	Orienting needle
G9	Using right hand to help tighten suture
G10	Loosening more suture
G11	Dropping suture at end and moving to end points
G12	Reaching for needle with left hand
G13	Making C loop around right hand
G14	Reaching for suture with right hand
G15	Pulling suture with both hands.

Table 1: *Gestures Vocabulary*

5.6.3 Surgeme Application on this Thesis

In this thesis, the JIGSAWS dataset is used specifically for the recovery of Cartesian tool-tip surgemes for the left and right PSMs. Surgemes, are integrated into the simulation platform. The knot tying task of JIGSAWS has been chosen because it is surgeme-annotated and is not as complex as suturing task. The initial .txt files have 76 kinematics columns per time step.

- For the right PSM: columns 59–61 extracted, which contain the Cartesian end-effector coordinates of the da Vinci's right tool.
- For the left PSM: columns 38–40 extracted, which contain the da Vinci's left tool coordinates.

The code loads the entire demonstration into memory just once at node initialization time and slices it into ten surgemes based on the index ranges set in the dataset. This gives a library of Cartesian reference trajectories for both arms.

5.6.4 Real-Time Surgeme Tracking Nodes

We employed two ROS nodes: *surgeme_subscriber_right* and *surgeme_subscriber_left*. Their functionality is:

- Input: Subscribe to live master Cartesian pose of each PSM (*/right_pose* topic or */left_pose* topic).
- Computation: Compute Euclidean distance from the tool's current position to all points of all surgemes of the given arm. Select the point and surgeme with minimum distance. If the distance is less than the threshold, make that surgeme "active" for that arm.
- Output:
  - o Publishes the selected point as a PoseStamped (ROS message which is a representation of pose in free space, composed of position and orientation, with reference coordinate frame and timestamp) on */surgeme_stream* topic.
  - o For visualization in Rviz, publishes a MarkerArray (ROS message which represents an array of markers which could be simple shapes or complex data) on */active_surgeme_marker* with the current active surgeme (green), the upcoming surgeme (grey) and the closest point so far (blue sphere) as shown in Fig.7, Fig.8 and Fig.9.

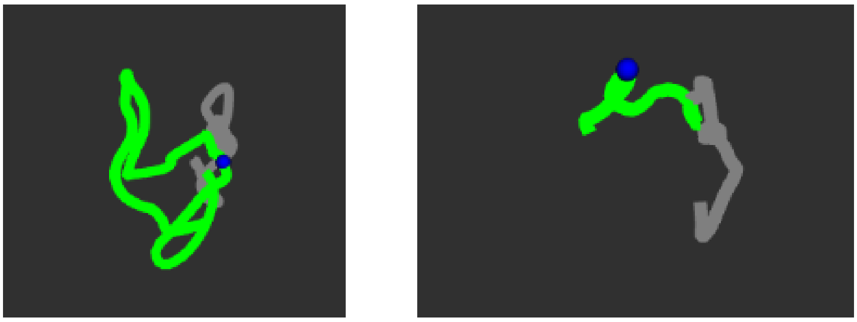
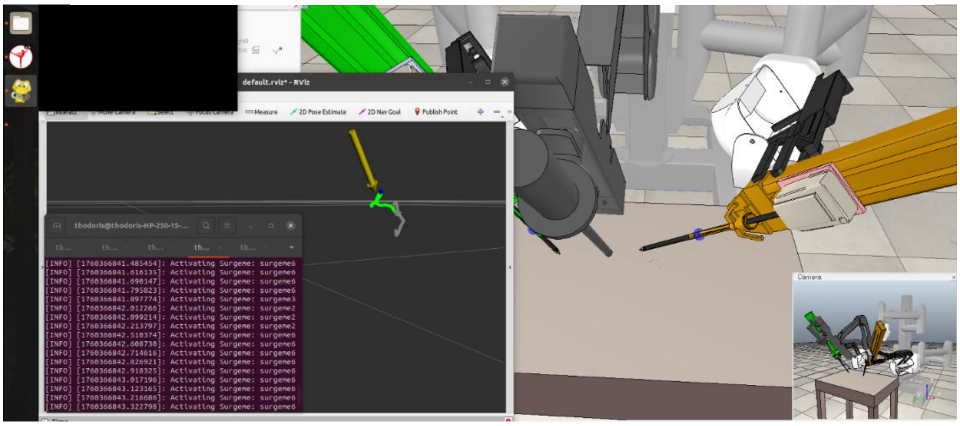


Figure 7: Active Surgemes (green), fourthcoming inactive Surgemes (grey) and tool tracker (blue point)



[redacted]

```
[INFO] [1760366841.485454]: Activating Surgeme: surgeme6
[INFO] [1760366841.616135]: Activating Surgeme: surgeme6
[INFO] [1760366841.698147]: Activating Surgeme: surgeme6
[INFO] [1760366841.795823]: Activating Surgeme: surgeme6
[INFO] [1760366841.897774]: Activating Surgeme: surgeme3
[INFO] [1760366842.012266]: Activating Surgeme: surgeme2
[INFO] [1760366842.099214]: Activating Surgeme: surgeme2
[INFO] [1760366842.213797]: Activating Surgeme: surgeme2
[INFO] [1760366842.518374]: Activating Surgeme: surgeme6
[INFO] [1760366842.608738]: Activating Surgeme: surgeme6
[INFO] [1760366842.714616]: Activating Surgeme: surgeme6
[INFO] [1760366842.826921]: Activating Surgeme: surgeme6
[INFO] [1760366842.918325]: Activating Surgeme: surgeme6
[INFO] [1760366843.017196]: Activating Surgeme: surgeme6
[INFO] [1760366843.123165]: Activating Surgeme: surgeme6
[INFO] [1760366843.216686]: Activating Surgeme: surgeme6
[INFO] [1760366843.322798]: Activating Surgeme: surgeme6
```

Figure 8: Closest Surgemes visualization and Console logs for dynamic recognition of Active Surgemes

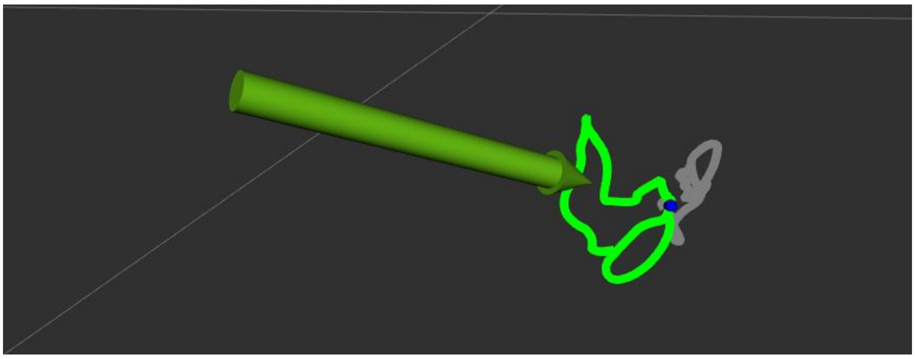


Figure 9: Closer look on Active and Forthcoming Surgeme and tool tracker

5.7 Force Feedback

The necessity of such node creation stems from the vulnerability of geometrically surgical guidance. A surgeon's instrument may be pushed away from the optimal route through reasons that range from hand tremor to visual distraction. Without corrective feedback, these deviations can lead to errors or inefficiencies. Conversely, an overstrictly overriding surgeon motion can compromise usability and trust.

5.7.1 Attractive Forces Node

- **Subscriptions**
  - ***/surgeme_stream*** topic (PoseStamped): current reference point on the active surgeme. On every callback, the node saves this as *last_valid_position*. This is the target the force will attract the tool to.
  - ***/left_pose*** and ***/right_pose*** (PoseStamped): the actual measured tool pose. On each callback, the node computes deviation from *last_valid_position*.
- **Publications**
  - ***/computed_force*** topic (Vector3: ROS message represents a vector in free space): the Cartesian correction force vector.
  - ***/visualized_force_marker*** topic (MarkerArray): for visualization of the attractive force as shown in Fig.10.

5.7.2 Attractive Forces Logic

In the present section we can see the formulas used to calculate attractive forces and their visualization using Rviz. In Table 2, we can see the mapping of mathematical symbols used for the calculations with their definitions.

Symbol	Meaning
$P_{current}$	Current position of left or right tooltip in x,y,z axis
P_{target}	Target position on active surgeme before the tool deviates
δ	Deviation between tooltip and last valid position on active surgeme

$\delta_{threshold}$	Tool proximity threshold
k	Force scaling
F	Total force

***Table 2:** Table of notations*

Two nodes for left and right PSMs, implemented to calculate attractive forces as follows:

$$\delta = \| P_{current} - P_{target} \| \qquad (10)$$

- If $\delta \leq \delta_{threshold}$, publish no force (transparent region). This lets the operator move freely when already on-track.
- Otherwise compute force vector based on the Hooke’s Law formula:

$$F = k * (P_{target} - P_{current}) \qquad (11)$$

- Publish Vector3 (F_x, F_y, F_z) on */computed_force* topic.

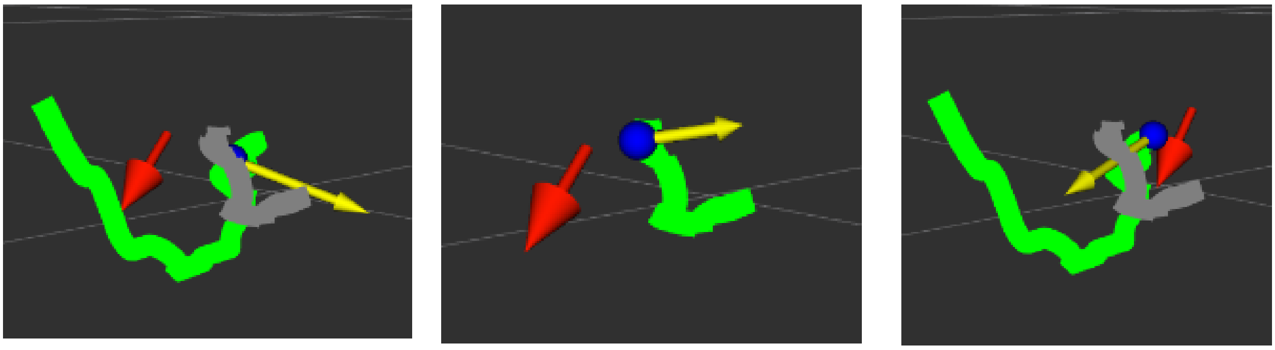


Figure 10: Attractive VFs (yellow arrow) visualization when deviation between active surgeme (green) and tool tip position (red arrow) overcomes proximity threshold

Chapter 6: Summary and Limitations

6.1 Summary of Contributions

The general aim of this thesis is to investigate feasibility of implementing a trajectory-based VFs in a simulated surgical robot. The thesis combines understanding of haptic feedback, active constraints and attractive VFs applied in a simulation environment. The study focuses on the role of VFs as a guidance tool, assisting accuracy and dexterity in surgical motions. In practical terms, the thesis contributes to the modular software framework for implementing and testing VFs in simulation. This thesis' key points are:

- Kinematic control using ROS nodes to enable teleoperation for the two dVRK PSMs.
- Surgeme integration from the JIGSAWS dataset and dynamic activation of closest surgemes, in every tool move.
- Application of attractive VFs, activated through force calculations and utilized to correct tool deviations more than a specified threshold.
- Visualization of PSMs, active and upcoming surgemes and attractive VFs using Rviz.

6.2 Limitations

Several limitations must be acknowledged. First, the kinematics model we use is limited to only three degrees of freedom. We can only teleoperate the PSM but can't control the gripper in order to provide wrist articulation. This results in reduction of accuracy while trying to track the trajectory with full dexterity.

Second, keyboard and mouse control help us control the PSMs but can't replicate the actual surgical robotic system's controllers for effective validation. This leads us to reduction of control, force perception and actual haptic feedback, while we have only visualization of forces.

Finally, CoppeliaSim environment, although it is highly flexible, it is a simulation environment, which can't give us the full prospect of using and controlling an actual robotic system or feel the actual tissue interaction. In general simulation environments are trying to replicate physics but are limited as an algorithmic approach.

Chapter 7: Bibliography

[1] Y. Shi *et al.*, “Dynamic Virtual Fixture Generation Based on Intra-Operative 3D Image Feedback in Robot-Assisted Minimally Invasive Thoracic Surgery,” *Sensors*, vol. 24, no. 2, p. 492, Jan. 2024, doi: 10.3390/s24020492.

[2] M. M. Marinho, B. V. Adorno, K. Harada, and M. Mitsuishi, “Dynamic Active Constraints for Surgical Robots Using Vector-Field Inequalities,” *IEEE Transactions on Robotics*, vol. 35, no. 5, pp. 1166–1185, Oct. 2019, doi: 10.1109/TRO.2019.2920078.

[3] M. O. Schurr, G. Buess, B. Neisius, and U. Voges, “Robotics and telemanipulation technologies for endoscopic surgery,” *Surg Endosc*, vol. 14, no. 4, pp. 375–381, Apr. 2000, doi: 10.1007/s004640020067.

[4] P. Dario, M. C. Carrozza, and A. Pietrabissa, “Development andin vitro testing of a miniature robotic system for computer-assisted colonoscopy,” *Computer Aided Surgery*, vol. 4, no. 1, pp. 1–14, 1999, doi: 10.1002/(SICI)1097-0150(1999)4:1<1::AID-IGS1>3.0.CO;2-J.

[5] A. R. Lanfranco, A. E. Castellanos, J. P. Desai, and W. C. Meyers, “Robotic Surgery,” *Ann Surg*, vol. 239, no. 1, pp. 14–21, Jan. 2004, doi: 10.1097/01.sla.0000103020.19595.7d.

[6] R. H. Taylor *et al.*, “An image-directed robotic system for precise orthopaedic surgery,” *IEEE Transactions on Robotics and Automation*, vol. 10, no. 3, pp. 261–275, Jun. 1994, doi: 10.1109/70.294202.

[7] J. R. Adler Jr., S. D. Chang, M. J. Murphy, J. Doty, P. Geis, and S. L. Hancock, “The Cyberknife: A Frameless Robotic System for Radiosurgery,” *Stereotact Funct Neurosurg*, vol. 69, no. 1–4, pp. 124–128, 1997, doi: 10.1159/000099863.

[8] N. Enayati, E. De Momi, and G. Ferrigno, “Haptics in Robot-Assisted Surgery: Challenges and Benefits,” *IEEE Rev Biomed Eng*, vol. 9, pp. 49–65, 2016, doi: 10.1109/RBME.2016.2538080.

[9] A. M. Okamura, “Methods for haptic feedback in teleoperated robot-assisted surgery,” *Industrial Robot: An International Journal*, vol. 31, no. 6, pp. 499–508, Dec. 2004, doi: 10.1108/01439910410566362.

[10] P. S. Green, J. W. Hill, J. F. Jensen, and A. Shah, “Telepresence surgery,” *IEEE Engineering in Medicine and Biology Magazine*, vol. 14, no. 3, pp. 324–329, 1995, doi: 10.1109/51.391769.

[11] L. B. Rosenberg, “Virtual fixtures: Perceptual tools for telerobotic manipulation,” in *Proceedings of IEEE Virtual Reality Annual International Symposium*, IEEE, 1993, pp. 76–82. doi: 10.1109/VRAIS.1993.380795.

[12] S. A. Bowyer, B. L. Davies, and F. Rodriguez y Baena, “Active Constraints/Virtual Fixtures: A Survey,” *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 138–157, Feb. 2014, doi: 10.1109/TRO.2013.2283410.

[13] A. Liu, F. Tendick, K. Cleary, and C. Kaufmann, “A Survey of Surgical Simulation: Applications, Technology, and Education,” *Presence: Teleoperators and Virtual Environments*, vol. 12, no. 6, pp. 599–614, Dec. 2003, doi: 10.1162/105474603322955905.

[14] A. G. Gallagher *et al.*, “Virtual Reality Simulation for the Operating Room,” *Ann Surg*, vol. 241, no. 2, pp. 364–372, Feb. 2005, doi: 10.1097/01.sla.0000151982.85062.80.

[15] P. Culmer, A. Alazmani, F. Mushtaq, W. Cross, and D. Jayne, “Haptics in Surgical Robots,” in *Handbook of Robotic and Image-Guided Surgery*, Elsevier, 2020, pp. 239–263. doi: 10.1016/B978-0-12-814245-5.00015-3.

[16] G. T. Fechner, “Elements of psychophysics, 1860.,” in *Readings in the history of psychology.*, East Norwalk: Appleton-Century-Crofts, 1948, pp. 206–213. doi: 10.1037/11304-026.

[17] M. A. Heller and E. Gentaz, *Psychology of Touch and Blindness*. Psychology Press, 2013. doi: 10.4324/9781315887555.

[18] A. B. Vallbo and R. S. Johansson, “Properties of cutaneous mechanoreceptors in the human hand related to touch sensation,” *Hum Neurobiol*, vol. 3, no. 1, pp. 3 – 14, 1984, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-0021154799&partnerID=40&md5=39f2de9e5b453691f8b39b327986a1db

[19] R. S. Johansson and J. R. Flanagan, “Coding and use of tactile signals from the fingertips in object manipulation tasks,” *Nat Rev Neurosci*, vol. 10, no. 5, pp. 345–359, May 2009, doi: 10.1038/nrn2621.

[20] U. Proske and S. C. Gandevia, “The kinaesthetic senses,” *J Physiol*, vol. 587, no. 17, pp. 4139–4146, Sep. 2009, doi: 10.1113/jphysiol.2009.175372.

[21] P. Joice, G. B. Hanna, and A. Cuschieri, “Errors enacted during endoscopic surgery—a human reliability analysis,” *Appl Ergon*, vol. 29, no. 6, pp. 409–414, Dec. 1998, doi: 10.1016/S0003-6870(98)00016-7.

[22] S. Souipas, A. Nguyen, S. G. Laws, B. L. Davies, and F. Rodriguez y Baena, “Real-time active constraint generation and enforcement for surgical tools using 3D detection and localisation network,” *Front Robot AI*, vol. 11, Mar. 2024, doi: 10.3389/frobt.2024.1365632.

[23] A. M. Okamura, “Haptic feedback in robot-assisted minimally invasive surgery,” *Curr Opin Urol*, vol. 19, no. 1, pp. 102–107, Jan. 2009, doi: 10.1097/MOU.0b013e32831a478c.

[24] M. E. Hagen, J. J. Meehan, I. Inan, and P. Morel, “Visual clues act as a substitute for haptic feedback in robotic surgery,” *Surg Endosc*, vol. 22, no. 6, pp. 1505–1508, Jun. 2008, doi: 10.1007/s00464-007-9683-0.

[25] J. Seetohul, M. Shafiee, and K. Sirlantzis, “Augmented Reality (AR) for Surgical Robotic and Autonomous Systems: State of the Art, Challenges, and Solutions,” *Sensors*, vol. 23, no. 13, p. 6202, Jul. 2023, doi: 10.3390/s23136202.

[26] T. Kastritsi, D. Papageorgiou, I. Sarantopoulos, S. Stavridis, Z. Doulgeri, and G. A. Rovithakis, “Guaranteed Active Constraints Enforcement on Point Cloud-approximated Regions for Surgical Applications,” in *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, May 2019, pp. 8346–8352. doi: 10.1109/ICRA.2019.8793953.

[27] A. Sharp and M. Pryor, “Virtual Fixture Generation for Task Planning With Complex Geometries,” *J Comput Inf Sci Eng*, vol. 21, no. 6, Dec. 2021, doi: 10.1115/1.4049993.

[28] A. Bettini, P. Marayong, S. Lang, A. M. Okamura, and G. D. Hager, “Vision-Assisted Control for Manipulation Using Virtual Fixtures,” *IEEE Transactions on Robotics*, vol. 20, no. 6, pp. 953–966, Dec. 2004, doi: 10.1109/TRO.2004.829483.

[29] T. L. Gibo, L. N. Verner, D. D. Yuh, and A. M. Okamura, “Design considerations and human-machine performance of moving virtual fixtures,” in *2009 IEEE International Conference on Robotics and Automation*, IEEE, May 2009, pp. 671–676. doi: 10.1109/ROBOT.2009.5152648.

[30] N. V. Navkar, Z. Deng, D. J. Shah, K. E. Bekris, and N. V. Tsekos, “Visual and force-feedback guidance for robot-assisted interventions in the beating heart with real-time MRI,” in *2012 IEEE International Conference on Robotics and Automation*, IEEE, May 2012, pp. 689–694. doi: 10.1109/ICRA.2012.6224582.

[31] Jing Ren, R. V. Patel, K. A. McIsaac, G. Guiraudon, and T. M. Peters, “Dynamic 3-D Virtual Fixtures for Minimally Invasive Beating Heart Procedures,” *IEEE Trans Med Imaging*, vol. 27, no. 8, pp. 1061–1070, Aug. 2008, doi: 10.1109/TMI.2008.917246.

[32] S. A. Bowyer and F. Rodriguez y Baena, “Dynamic frictional constraints for robot assisted surgery,” in *2013 World Haptics Conference (WHC)*, IEEE, Apr. 2013, pp. 319–324. doi: 10.1109/WHC.2013.6548428.

[33] M. Li, A. Kapoor, and R. H. Taylor, “Telerobotic Control by Virtual Fixtures for Surgical Applications,” in *Advances in Telerobotics*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 381–401. doi: 10.1007/978-3-540-71364-7_23.

[34] J. J. Abbott, P. Marayong, and A. M. Okamura, “Haptic Virtual Fixtures for Robot-Assisted Manipulation,” in *Robotics Research*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 49–64. doi: 10.1007/978-3-540-48113-3_5.

[35] Y. Gao *et al.*, “JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS) : A Surgical Activity Dataset for Human Motion Modeling,” 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:16185857

[36] G. A. Fontanelli, M. Selvaggio, M. Ferro, F. Ficuciello, M. Vendittelli, and B. Siciliano, “A V-REP Simulator for the da Vinci Research Kit Robotic Platform,” in *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, IEEE, Aug. 2018, pp. 1056–1061. doi: 10.1109/BIOROB.2018.8487187.