Robot-Assisted Drilling on Curved Surfaces with Haptic Guidance under Adaptive Admittance Control

by

Alireza Madani

A Dissertation Submitted to the

Graduate School of Sciences and Engineering

in Partial Fulfillment of the Requirements for

the Degree of

Master of Science

in

Mechanical Engineering

KOÇ ÜNİVERSİTESİ

August 2022

Robot-Assisted Drilling on Curved Surfaces with Haptic Guidance under Adaptive Admittance Control

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Alireza Madani

and have found that it is complete and satisfactory in all respects,

and that any and all revisions required by the final

examining committee have been made.

Committee Members:

Prof. Dr. Cagatay Basdogan (Advisor)

Assist. Prof. Yusuf Aydin (Co-Advisor)

Prof. Dr. Volkan Patoglu

Assist. Prof. Ozgur Salih Oguz

Date: ______________________________

To all the people who have helped and understood me through the tough chapters of my life; not hampered my path, and gave me the chance to grow forward.

To my mother... I owe you a lot.

ABSTRACT

Robot-Assisted Drilling on Curved Surfaces with Haptic Guidance under Adaptive Admittance Control

Alireza Madani

Master of Science in Mechanical Engineering

August 2022

Drilling a hole on a curved surface with a desired angle is prone to failure when done manually, due to the difficulties in drill alignment and also inherent instabilities of the task, potentially causing injury and fatigue to the workers. On the other hand, it can be impractical to fully automate such a task in real manufacturing environments because the parts arriving at an assembly line can have various complex shapes where drill point locations are not easily accessible making automated path planning difficult. In this work, an adaptive admittance controller with 6 degrees of freedom is developed and deployed on a KUKA LBR iiwa 7 cobot such that the operator is able to manipulate a drill mounted on the robot with one hand comfortably and open holes on a curved surface with haptic guidance of the cobot and visual guidance provided through an AR interface. Real-time adaptation of the admittance damping provides more transparency when driving the robot in free space while ensuring stability during drilling. After the user brings the drill sufficiently close to the drill target and roughly aligns to the desired drilling angle, the haptic guidance module fine tunes the alignment first and then constraints the user movement to the drilling axis only, after which the operator simply pushes the drill into the workpiece with minimal effort. Two sets of experiments were conducted to investigate the potential benefits of the haptic guidance module quantitatively (Experiment **I**) and also the practical value of the proposed pHRI system for real manufacturing settings based on the subjective opinion of the participants (Experiment **II**). The results of Experiment **I** conducted with 3 naïve participants, show that haptic guidance improves task completion time by 26% while decreasing human effort by 16% and muscle activation levels by 27% compared to no haptic guidance condition. The results of Experiment **II**, conducted with 3 experienced industrial workers, show that the proposed system is perceived to be easy to use, safe, and helpful in carrying out the drilling task.

ÖZETÇE

Uyarlamalı Admitans Kontrolcü ve Haptik Yönlendirme ile

Kavisli Yüzeylerde Robot Destekli Delme

Alireza Madani

Makine Mühendisliği, Yüksek Lisans

Ağustos 2022

Kavisli bir yüzey üzerine belirli bir açıyla manuel olarak delik açma işlemi, matkap hizalanmasında karşılaşılacak olası problemler ve görevin doğasından kaynaklanan kararsızlık sebebiyle oldukça zordur. Aynı zamanda bu tür bir manuel görev, kullanıcının yaralanmasına veya çabuk yorulmasına sebep olabilir. Ayrıca, gerçek bir üretim ortamında bu görevin tamamen otomatik olarak tasarlanması da mümkün olmayabilir. Çünkü, üretim hattına gelen parçalar farklı kompleks şekillere sahip olabilir ve dolayısıyla seçili noktaları otomatik olarak delmek için yörünge planlaması yapmak kolay olmayabilir. Bu çalışmada, 6 serbestlik derecesine sahip olan uyarlamalı bir admitans kontrolcüsü tasarlanmıştır. Ucuna matkap monte edilmiş KUKA LBR iiwa 7 işbirlikçi robot, tasarlanan kontrolcü aracılığıyla operatöre konforlu bir şekilde ve tek elle kavisli bir yüzey üzerinde delikler açabilmesini sağlamaktadır. Operatöre delik açma sırasında, robot tarafından dokunsal (haptik) yönlendirme ve bir artırılmış gerçeklik arayüzü vasıtasıyla görsel yönlendirme sağlanmaktadır. Admitans kontrolcüsünün sönümleme parametresinin gerçek zamanlı adaptasyonu, robotun çalışma uzayındaki çevre ile etkileşim olmaksızın gerçekleşen hareketi sırasında şeffaflık (robotun insana olan direncinin düşük olması) sağlarken delme işlemi sırasında kararlılık sağlamaktadır. Operatör, matkabı delme noktasına yeteri kadar yaklaştırdıktan ve istenilen delme açısına kabaca hizaladıktan sonra ilk olarak haptik yönlendirme modülü, hizalamayı hassas şekilde ayarlamakta ve operatörün hareketi sadece delme ekseni boyunca olacak şekilde kısıtlamaktadır. Ardından operatör, minimal bir eforla matkabı delinecek parçaya doğru iterek delme işlemini tamamlamaktadır. Haptik yönlendirme modülünün potansiyel yararlarını nicel olarak araştırmak için bir deney (Deney 1) ve önerilen fiziksel insan-robot etkileşim sisteminin gerçek bir üretim ortamındaki pratik değerini katılımcıların kişisel görüşüne dayanarak araştıran başka bir deney olmak üzere (Deney 2) iki farklı deney tasarlanmıştır. Bu alanda tecrübesi olmayan 3 katılımcı ile yapılan birinci deneyin sonucunda (Deney 1), haptik yönlendirme modülünün görev tamamlama süresini %26 oranında iyileştirdiği, insanın uyguladığı eforu

%16 ve kas aktivasyon seviyelerini %27 oranında azalttığı bulunmuştur. 3 deneyimli işçi ile yapılan ikinci deney (Deney 2), delme işlemi sırasında önerilen sistemin kullanımının kolay, güvenli ve yararlı olduğunu göstermiştir.

AKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, **Prof. Dr. Cagatay Basdogan** for his guidance, savvy and understanding throughout my MSc study. This is definitely a lifetime honour of my career that I got the chance to conduct multidisciplinary research under the supervision of such an experienced, renowned and outstanding scientist in our field. Without his support, this thesis would not have been possible. One of the most important things that I got the chance to learn during the last 2 years from him was how to break a research problem to smaller pieces and meticulously trying to address them "step-by-step". He has proved that he always wants the best for me. Putting the academics aside, he has always been supporting me and mentoring me to make the best decisions in each stage during the pas two years.

I would like to warmly thank **Assist. Prof. Yusuf Aydin** for never hesitating to share his invaluable amount of knowledge and hands-on experience with me. Unquestionably, he is one of the smartest, most patient, and knowledgeable scientist I know around me in this domain. I want to kindly thank him for always showing me the right path to follow and being supportive regardless of them date and time. I will always be proud of having Dr. Aydin as one of my older friends.

I would like to kindly thank **Prof. Dr. Volkan Patoglu** and **Assist. Prof. Ozgur S. Oguz** who are both outstanding scientist in the scope of robotics and intelligent systems, both nationally and internationally, for accepting to join the jury committee of my thesis defence.

Our team in Robotics and Mechatronics Laboratory of Koc University, would like to thank the project's industrial partner company, As-Metal Inc. and its CEO, **Dr. Serkan Necipoglu**, for providing the curved workpieces used in both experiments and the experienced industrial workers participated in the second experiment.

Last but not least, I would like to warmly thank my dear friends and labmates, **Pouya P. Niaz** and **Berk Guler**. At the time of starting my MSc degree they both had significant coding skills. Now that I am able to take care of all the coding technicalities on my own, I am realizing that without their help and support this would not be possible. I have learned a lot from both of them.

Hereby, I acknowledge that The Scientific and Technological Research Council of Turkey (TUBITAK) supported this study under contract EEEAG-117E645. I would like to also acknowledge the research fellowship generously provided by KUIS Research Center during my MSc studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

pHRI	Physical Human-Robot Interaction
AR	Augmented Reality
DoF	Degree of Freedom
Cobot	Collaborative Robot
s	Laplace variable
$G(s)$	The transfer function of the robot and its internal controller, whose output is the actual velocity of the robot's end effector
$\mathbf{v}$	General velocity vector of robot's end-effector, consisting of 3 translational and 3 rotational components
$Z_h(s)$	Mechanical impedance of human
$Z_{env}(s)$	Mechanical impedance of environment
$\mathbf{F}_{\mathbf{env}}$	Wrench vector exerted by environment, consisting of 3 force and 3 torque components
$\mathbf{F}_{\mathbf{h}}$	Wrench vector exerted by human, consisting of 3 force and 3 torque components
$\mathbf{F}_{\mathbf{int}}$	Interaction wrench vector which is the resultant of human and environmental wrenches, consisting of 3 force and 3 torque components
Y(s)	Admittance controller transfer function
$\mathbf{v}_{\mathbf{ref}}$	General reference velocity vector of end-effector, consisting of 3 translational and 3 rotational components
$\mathbf{v}_{\mathbf{des}}$	Human operator's general desired velocity vector of end-effector, consisting of 3 translational and 3 rotational components

6D	6 Dimensional
v_x	X-component of a translational velocity vector
v_y	Y-component of a translational velocity vector
v_z	Z-component of a translational velocity vector
ω_x	X-component of an angular velocity vector.
ω_y	Y-component of an angular velocity vector.
ω_z	Z-component of an angular velocity vector.
$V_{ref}(s)$	A component of reference velocity of robot's end-effector in Laplace domain.
$F_{int}(s)$	A component of interaction force in Laplace domain.
m	Admittance Mass Parameter
b	Admittance Damping Parameter
{LFF}	Local Force Sensor Cartesian Coordinate Frame
$F_{x,LFF}$	X-component of force vector measured and expressed in {LFF}
$F_{y,LFF}$	Y-component of force vector measured and expressed in {LFF}
$F_{z,LFF}$	Z-component of force vector measured and expressed in {LFF}
$V_{x,LFF}$	X-component of reference computed velocity vector of {LFF} expressed in {LFF}
$V_{y,LFF}$	Y-component of reference computed velocity vector of {LFF} expressed in {LFF}
$V_{z,LFF}$	Z-component of reference computed velocity vector of {LFF} expressed in {LFF}
{KBF}	KUKA Robot's Base Cartesian Coordinate Frame, i.e., Denavit-Hartenberg convention 0^{th} coordinate frame

D-H	Denavit-Hartenberg
${}^{0}_{LFF}T$	The homogenous transformation matrix of {LFF} with respect to {KBF}
${}^{7}_{LFF}T$	The homogenous transformation matrix of {LFF} with respect to D-H 7th coordinate frame
${}^{0}_{7}T$	The homogenous transformation matrix of D-H 7th coordinate frame with respect to {KBF}
$\boldsymbol{P}_{des,LFF}$	Desired position vector of {LFF} that we want to go to with respect to and expressed in KBF
$\boldsymbol{P}_{current,LFF}$	Current position vector of {LFF} with respect to and expressed in KBF
$\boldsymbol{V}$	Computed velocity vector of LFF by admittance controller expressed in KBF
$M_{x,LFF}$	X-component of moment vector measured and expressed in {LFF}
$M_{y,LFF}$	Y-component of moment vector measured and expressed in {LFF}
$M_{z,LFF}$	Z-component of moment vector measured and expressed in {LFF}
$\omega_{x,LFF}$	Desired rate of 1st Euler angle of {LFF} with respect to current orientation of {LFF}
$\omega_{y,LFF}$	Desired rate of 2nd Euler angle of {LFF} with respect to current orientation of {LFF}
$\omega_{z,LFF}$	Desired rate of 3rd Euler angle of {LFF} with respect to current orientation of {LFF}
$\boldsymbol{\omega}$	Rate vector of Euler angles of {LFF} with respect to current orientation of {LFF}

$\boldsymbol{\Delta\theta_{LFF}}$	Desired angular displacement vector in terms of Euler angles of {LFF} with respect to current orientation of {LFF}
$^{Current}_{Des}R^{LFF}$	Desired orientation of {LFF} with respect to current orientation of {LFF} expressed as 3 by 3 rotation matrix
$_{LFF}^{0}T_{des}$	Homogenous transformation matrix which indicates the desired position and orientation of {LFF} with respect to {KBF}.
$\overrightarrow{AB}$	Vector defined by probed points A and B on curved surface.
$\overrightarrow{AC}$	Vector defined by probed points A and C on curved surface.
$\vec{N}$	Local unnormalized approximate surface normal vector derived by cross product of vectors $\overrightarrow{AB}$ and $\overrightarrow{AC}$
$\vec{n}$	Local normalized approximate surface normal vector derived by cross product of vectors $\overrightarrow{AB}$ and $\overrightarrow{AC}$
$\overrightarrow{aux}$	Auxiliary Vector
$\overrightarrow{comp}$	Completer Vector
{LNF}	Local Surface Normal Cartesian Coordinate Frame
φ	Polar Angle
θ	Azimuth Angle
{T}	Tool Cartesian Coordinate Frame
{A}	Arbitrary Cartesian Coordinate Frame A
{B}	Arbitrary Cartesian Coordinate Frame B
$^{B}_{A}R$	Rotational matrix which expresses orientation of {A} with respect to {B}
$\boldsymbol{q}_i$	The current 7×1 join position vector that we the robot has and it available for us
$\boldsymbol{q}_f$	The computed joint configuration of robot for the final aligned pose of drill bit

$\hat{r}$, $\hat{\theta}$, $\hat{\varphi}$	The conventional unit vectors of polar 3D coordinate system
${}^{polar}\hat{r}$,	Polar coordinate system unit vector $\hat{r}$ expressed in the polar reference.
${}^{polar}\hat{\theta}$	Polar coordinate system unit vector $\hat{\theta}$ expressed in the polar reference.
${}^{polar}\hat{\varphi}$	Polar coordinate system unit vector $\hat{\varphi}$ expressed in the polar reference.
{HLWF}	HoloLens World Frame
{HLLF}	HoloLens Local Frame
{EEF}	Robot's End Effector Frame
{TF}	AprilTag Frame
ε_{φ}	Error in polar angle adjustment
ε_{θ}	Error in azimuth angle adjustment
sEMG	Surface Electromyography
t_{tot}	Task completion time
s_{lin}^{avg}	Average linear speed
s_{ang}^{avg}	Average angular speed
V_{avg}	Average Flexor Carpi muscle activation
E_F	Human effort caused by forces
E_{τ}	Human effort caused by torques
E_{tot}	Total human effort
$\varepsilon_{\varphi}^{avg}$	Average alignment error for polar angle
$\varepsilon_{\theta}^{avg}$	Average alignment error for azimuth angle
ANOVA	Analysis of Variance
VR	Virtual Reality

Chapter 1:
INTRODUCTION

In the near future, humans and robots are expected to perform collaborative tasks involving physical interaction in various different environments such as homes, hospitals, and factories.

To this end, impedance and admittance controllers are often used for regulating the physical interaction between the human and the robot (pHRI). In admittance control, the input is the interaction force between human, robot and the environment, and the output is the reference velocity which the robot is to follow [1], [2].

One of the application areas where pHRI can be used for improving task efficiency is the manufacturing industry. Robots have been used in industrial settings at an increasing rate since 1980s as a result of the improvements in their capabilities and ease of use [3], [4]. Many large-scale manufacturing tasks have therefore been successfully automated thanks to these advances in robotics. In the meantime, several small-batch tasks requiring higher level decision-making and supervision are still performed by humans. In recent years though, pHRI has enabled humans to collaborate with robots, resulting in such tasks being done more efficiently [5], [6].

In fact, the number of pHRI studies that focus on improving task performance, practicality, completion time and ergonomics in manufacturing industries has increased dramatically during the last few years [7], [8]. For instance, Cherubini et al. [9] used pHRI for mitigating muscular pain in an automotive assembly line. Zhang et al. [10] proposed a framework for task planning in pHRI settings for minimizing fatigue. Lamon et al. [11] introduced a visuo-haptic guidance interface for mobile manipulators to understand instructions better in manufacturing environments. Guerin et al. [12] proposed a pHRI framework with different levels of autonomy for collaborative tasks in manufacturing settings. Later on, they used a similar framework to make pHRI more convenient in tasks such as bending metallic pieces and spot welding, based on predefined sets of robot capabilities [13]. Ochoa et al. [14] proposed an impedance control scheme that captured human skills in a glass mold micro drilling task, then programmed a robot to automate the drilling. Kena et al. [15] also proposed an impedance control scheme for pHRI-enabled curve tracing techniques in edge chamfering and polishing. Perez-Ubeda et al. [16] used the milling of soft materials as a practical example for determining the capabilities and limitations of collaborative robots in machining applications.

Despite many studies on automated robotic drilling in the literature [17], [18], the number of studies on robotic-assisted drilling with physical human guidance is limited. However, fully automated drilling does not always obviate the demands of numerous realistic scenarios in manufacturing settings such as opening a hole on a curved piece in which the robot does not have any prior knowledge of what may be located beneath the drilling surface with which collision should be avoided. In another scenario, we have the human trying to keep a piece of another assembly part on the workpiece and drill both of them together. Under all such circumstances, the human operator will not be able to work alongside a pre-programmed robot. In the meantime, manual drilling, requiring the stabilization of the workpiece and drill bit with fixtures, is time consuming yet not precise enough. Accounting for the abovementioned rationales, fusing human intelligence and supervision with robot's robustness and precision is the key to our concerns. Aydin et al. [19] worked on a collaborative drilling task by identifying the suitable

Figure 1, Our pHRI system designed for drilling on a curved surface consists of a cobot, a drill attached to it, two force sensors, and a HoloLens AR goggle.

control parameters of an admittance controller and adapting them for a safe and stable operation. Sirintuna et al. [20] utilized a fractional-order admittance control scheme to better balance the trade-off between transparency (minimal resistance to the human) during free motion and stability during drilling with a cobot. Both of the drilling studies mentioned above only considered drilling on a flat surface using a single degree of freedom admittance controller.

In this study, an industry-standard pHRI system was developed for performing small-batch manufacturing tasks with haptic guidance of cobot, visual feedback provided through an augmented reality (AR) interface, and an interaction controller for admittance adaptation. Drilling holes on a curved surface with a desired angle was selected as a case study to demonstrate the potential use of the proposed pHRI system in manufacturing settings (Figure 1). It is difficult to open a hole on a curved surface at a custom angle using a manual drill. On the other hand, even with full knowledge of drill point location and drilling angle, workpieces with different curvature and material properties may arrive at the station in an assembly line, some with protrusions and cavities, some with sensitive or loose pieces attached to them, making full automation unfeasible. Hence, we proposed to use a cobot for haptic guidance, weight compensation, and precise alignment, and augmented reality (AR) interface for visual guidance, and an adaptive controller for admittance adaptation to assist the human operator in opening holes on a curved surface. We therefore aim to fuse the intellectual capacity of the human (where, at what angle, and how deep the whole should be drilled, as well as how the robot should be guided to that location) with geometric precision and mechanical robustness of the robot to execute the task. In our drilling scenario, the user brings the drill attached to the robot sufficiently close to the target point on the curved surface utilizing an augmented reality (AR) interface, an then the robot aligns the drill bit with the selected drilling angle using the so-called haptic guidance module, allowing the human to drill the selected hole with exact desired angle by simply pushing the drill forwards to the target point.

In this work, two sets of human experiments were conducted aiming to evaluate the effectiveness of haptic guidance in task performance, and to assess the user's willingness to accept the technology, as well as feasibility of the proposed system. The first experiment was conducted with three naïve participants who had no prior experience with our system. Their drilling task performance under haptic guidance condition was compared with that of no haptic guidance condition using some quantitative metrics including accuracy in drilling orientation, task completion time, and effort made by operator. In the second experiment, the focus was on the subjective experience of the user about the proposed system under haptic guidance condition. This experiment was performed with three industrial workers who had extensive experience in drilling in assembly lines. Following the experiments, the participants were asked to fill out a questionnaire regarding their subjective opinions about the system and its potential use in real manufacturing environments.

In this study, an adaptive 6-DoF admittance controller is used for drilling a curved surface. The adaptation mechanism enables a good balance between transparency (minimal resistance to the human) and stability (see [20], [21], [22]). It keeps the pHRI system transparent as long as the robot is driven freely in space. Once it is sufficiently close to the target point (i.e., within a predefined spherical radius), the robot becomes stiffer to prevent sudden collision with the curved workpiece and help the human with the rough alignment of the drill bit. For the final drilling phase, after the robot aligns the drill bit precisely and confines the human movement to the desired drilling axis, the robot becomes even stiffer, to maximize stability and safety during the drilling to open a hole.

The remainder of this thesis is organized as follows:

Chapter 2, Approach, presents our approach including the hardware components, the control architecture, and the details of our drilling application. Chapter 3, Experiments, presents both of our 2 sets of experiments. In our first experiment we compare the task performance of 3 naïve participants when they perform the drilling task with and without haptic guidance. In our second experiment we examine the feasibility of deploying the proposed system in real manufacturing settings based on the subjective opinion of 3 well-experienced industrial workers. Finally, in Chapter 4, Discussion and Conclusion, we discuss the experiment results and present our final remarks.

Chapter 2:

APPROACH

We propose a solution for the task of drilling holes on curved surfaces with potentially unknown geometries, at locations marked by the user and for user-defined drilling angles. To this end, we developed a pHRI system whose components and their inner workings are explained below in detail.

2.1 *Hardware*

The major hardware components of our system are (1) a cobot: a 7R, KUKA LBR iiwa7 R800 robot, (2) a powered drill: includes a DC motor operable between 0 and 48 Volts, and drill bit, (3) two ATI Mini45 force/torque sensors: one of them is attached between the drill and the robot's end-effector measuring interaction forces applied to it, while the other is attached between the drill and the handle which is held by the operator, measuring solely the human force, and (4) an AR goggle: a Microsoft HoloLens augmented reality interface. (Figure 2)

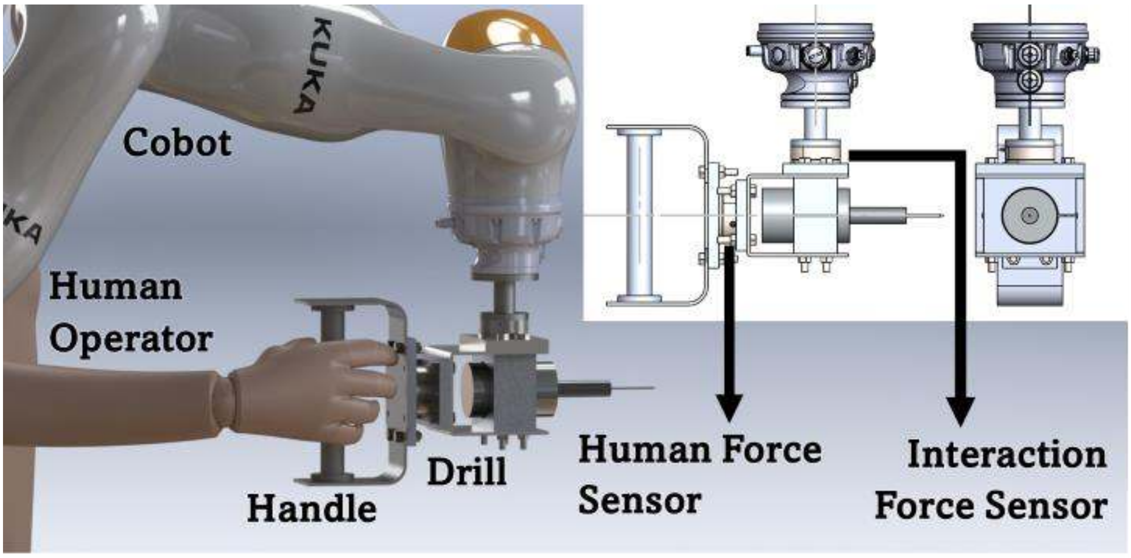


Figure 2, A closer shot of the hardware components used in our study.

2.2 *Control Loop*

The closed-loop control system used in this study is shown in Figure 3. Here, G(s) is the transfer function of the robot and its internal controller, whose output is the actual velocity of the robot's end effector **v**. The mechanical impedance of human and environment are shown as $Z_h(s)$ and $Z_{env}(s)$ respectively. Resultant of the environmental force $\mathbf{F_{env}}$ and the force applied by human $\mathbf{F_h}$ is the interaction force $\mathbf{F_{int}}$, which is sent to the admittance controller Y(s), whose output is the reference end-effector velocity $\mathbf{v_{ref}}$. In Figure 3, $\mathbf{v_{des}}$ is the desired velocity of the human operator. When there is no haptic guidance, the operator has to manually align the drill with the desired drill orientation. Therefore, switches S1 and L1 are closed. However, when haptic guidance is activated, switch S1 and L1 are closed only when driving the robot in free space. When the robot reaches the 5-cm vicinity of the drill target, it locks for the automatic alignment, at which point switch S1 is opened. During a 4 second interval in which the robot aligns itself with the desired drilling orientation, the admittance control is not active. After alignment, switches S1 and L2 are closed, admittance control is reactivated, and the robot's motion is confined to the drilling axis, so the operator can push the drill forwards to the drill target to open a hole. In Figure 3, all variables are 6D vectors, including three translation terms and three rotational terms. For example, the velocity vectors in the figure are expressed as $\mathbf{v} = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^T$ where v and ω stand for linear and angular velocities, respectively. The same applies to wrench vectors in the figure, which include 3 components of force and 3 components of torque with respect to three orthogonal axes.

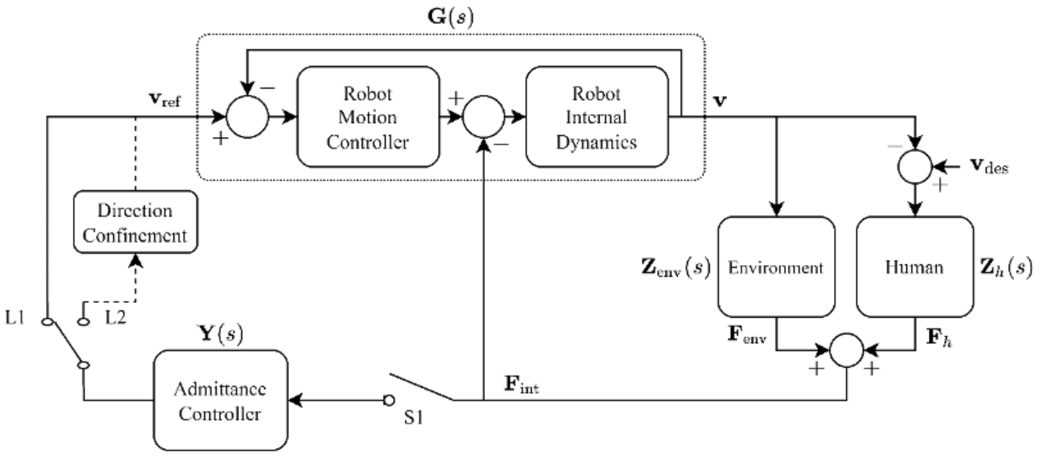


Figure 3, Closed-loop control system used in our pHRI study.

2.3 *Admittance Controller*

So as to prevent the reader from getting lost in the implementation details, this section is divided into two parts, namely, "general picture" and "detailed elaboration". In the first part, a general picture of the 6-DoF decoupled admittance controller is alluded. So, one can read this first section without losing the main concept. In the second part, an extensively detailed picture of the working and implementation principles of a 6-DoF admittance controller for our pHRI setup is depicted.

General Picture: We utilize 6 decoupled admittance controllers, one for each degree of freedom, to regulate the translational and rotational interactions between the robot and the human. For each translational degree of freedom, the admittance controller can be expressed in the Laplace domain as below:

$$Y(s) = \frac{V_{ref}(s)}{F_{int}(s)} = \frac{1}{ms + b} \tag{2.1}$$

where $V_{ref}(s)$ is the reference velocity in the Laplace domain that is given by the controller, and will be followed by the robot, $F_{int}(s)$ is the input interaction force in the Laplace domain, m and b are our desired mass and damping parameters, respectively, that we want to impose on the pHRI system so it behaves like a mass connected to a damper all the time during our interactions with the robot. And s is the Laplace variable. A similar relationship can be written between torque and reference angular velocity for each rotational degree of freedom as well.

Detailed Elaboration: The approach that we have used for driving a robotic manipulator through interaction forces in 6-DoF general motion could be the first of its kind. In Figure 3 and all other control loops in our recent pHRI papers there is a block called admittance controller. So far ([19], [20], [21]) since our pHRI application was 1 dimensional drilling on a flat workpiece placed perpendicular to the drill bit the admittance controller used was not taking into account the rotation of end effector with respect to the robot's base frame. But in this general pHRI study where we are dealing with a general motion of a tool connected to the robot, i.e., a drill in 3D space which involves 6-DoF motion of a rigid body in space, developing a controller which can take care of all 6 degrees of motion and runs in real-time without reducing our pHRI, hardware-software closed loop frequency is a challenging task. In this work, we present a decoupled approach which solves the 6 differential equations mentioned in (2.1) for all 6 DoF simultaneously in real-time and gives joint-position commands to the KUKA cobot accordingly.

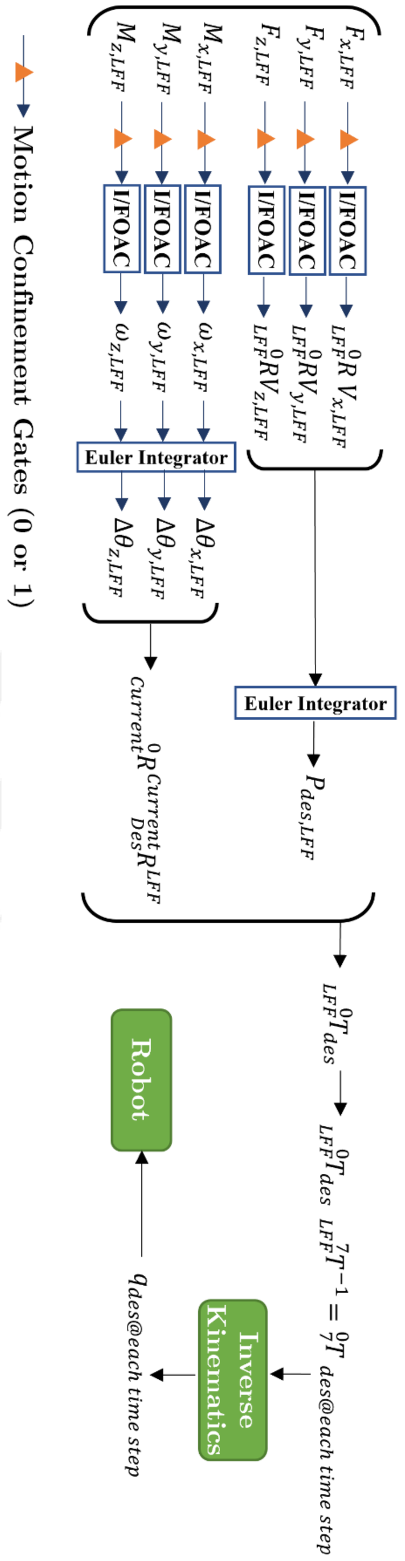


Figure 4, Internal computation architecture of the proposed 6 DoF admittance controller.

As you can see in Figure 4, we follow a multi-stage approach which involves solving the admittance control differential equations, forward kinematics, and inverse kinematics in real-time. To explain the execution flow of the controller, we divide the computations into 2 branches. We will discuss execution flows for translational and rotational components separately up to some level and them we fuse the results to run the robot in 6D.

Computations Due to Translation: We measure the forces $F_{x,LFF}$,$F_{y,LFF}$, and $F_{z,LFF}$ with respect to and expressed in Local Force Sensor Coordinate Frame ({LFF}). Then for each of the three translational components we solve the admittance equation (2.1) which will give us the desired velocity vectors ($V_{x,LFF}$,$V_{y,LFF}$, and $V_{z,LFF}$) with respect to and expressed in the {LFF}. In parallel to these computations, we compute the ${}_{LFF}^{0}T$ which is the homogenous transformation matrix between {LFF} and KUKA Base Frame ({KBF}). Note that computing ${}_{LFF}^{0}T$ is straight forward since due to the known geometry of our drill machine assembly we know ${}_{LFF}^{7}T$ and ${}_{7}^{0}T$ is computed using forward kinematics function[1]. We use the rotation matrix portion of ${}_{LFF}^{0}T$ to express the computed desired velocities mentioned above in the {KBF}. Now we integrate the velocity through time for only one time-step and add it to our current position vector of {LFF} with respect to and expressed in {KBF} which will give us the next desired position of {LFF} with respect to and expressed in {KBF}.

$$\boldsymbol{P}_{des,LFF} = \boldsymbol{P}_{current,LFF} + \boldsymbol{V}\,dt \tag{2.2}$$

Computations Due to Rotation: The measured torques around each axis of the force sensor ($M_{x,LFF}$,$M_{y,LFF}$, and $M_{z,LFF}$) are also fed to the admittance controller which has our desired rotational mass (second moment of inertia) and

[1] The number 7 here corresponds to 7th coordinate system installed on the 7th joint of the robot using the Denavit-Hartenberg convention.

damping characteristics. We will have $\omega_{x,LFF}$, $\omega_{y,LFF}$, and $\omega_{z,LFF}$ out of the admittance controller which are the desired Euler angle rates of {LFF} with respect to current orientation of {LFF}. Using the same integration method as in (2.2) we can compute the desired angular displacement in terms of Euler angles of {LFF} with respect to current orientation of {LFF}.

$$\Delta\boldsymbol{\theta}_{LFF} = \boldsymbol{\omega}_{LFF}\, dt \tag{2.3}$$

Using a conventional function which converts the Euler angles to rotation matrix we compute the ${}^{Current}_{Des}R^{LFF}$ which indicates the desired orientation of {LFF} with respect to current orientation of {LFF} as a 3 by 3 rotation matrix. We here also use the rotation matrix portion of ${}^{0}_{LFF}T$ to express this desired orientation of {LFF} in {KBF} frame.

Merging the Translational and Rotational Computations: Using the results of the last two sections, we can shape the ${}^{0}_{LFF}T_{des}$ homogenous transformation matrix which indicates the desired position and orientation of {LFF} with respect to {KBF}.

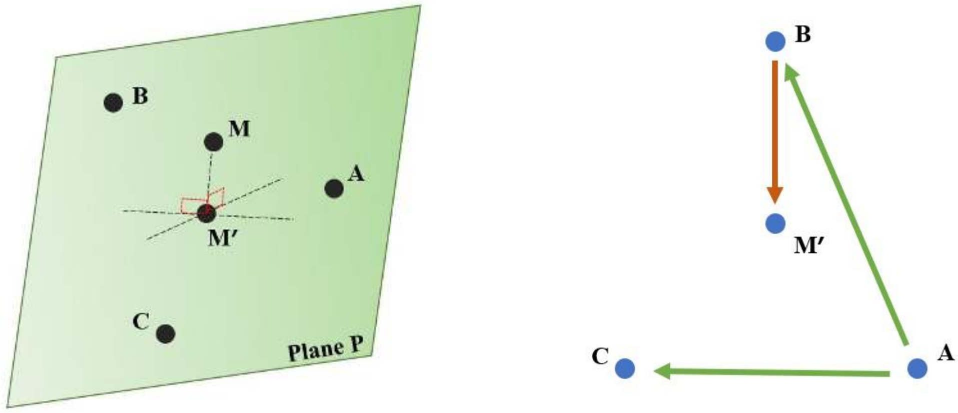
$${}^{0}_{LFF}T_{des}\ {}^{7}_{LFF}T^{-1} = {}^{0}_{7}T_{\ des@each\ time\ step} \tag{2.4}$$

Using (2.4) we can compute the desired position and orientation of 7th coordinate system of our robot in real-time which could be fed to the inverse kinematics function to generate the next joint positions of the robot ($q_{des@each\ time\ step}$).

2.4 *Extracting Surface Information*

The scenario explored in this study includes a curved workpiece with an arbitrary shape arriving at the station, after which the operator marks a point on it for drilling and selects an appropriate drilling angle (not necessarily perpendicular). In order to open a hole at the marked point (drill target) with a desired drilling axis, the location of the drill target must be known with respect to the base frame of the robot({KBF}). For this purpose, a 3D surface mesh model of the workpiece was obtained by a depth camera first and then registered with the real surface using a virtual tag attached to it. Alternatively, the drill target and three points around it with specific configuration which will be explained in detail can be sampled(probed) with the tip of the drill bit manually to register the target location. Though, time consuming and not as accurate as optical scanning techniques, this approach was tested by us, eliminates the need for a complete 3D surface mesh model of the curved surface.

As mentioned above, we have two different approaches for registering the position of drilling target point and computing the local surface normal of the curved surface with respect to the robot base coordinate system, namely, depth sensor-based approach and probing-based approach. The depth-sensor based approach is straight forward since the software of the depth-camera system will give us the local surface normal vectors for each point of the mesh. But the probing-based approach in which we take advantage of our robot's forward kinematics precision (using the robot as a Coordinate Measuring Machine (CMM)) to extract the surface topology features locally at the drilling site, has lots of technicalities that we will discuss in this subsection. It is noteworthy that regardless of the method that we use to extract the local surface normal at the drilling target, the self-alignment software module should be given a homogenous transformation matrix which expresses desired position and orientation (polar and azimuth angles) of the robot with respect to KBF at the position of parking before motion confinement and consecutively 1D drilling. Computing this homogenous transformation matrix is of critical importance in our work since its principles will be used to monitor the drilling error in real-time during our Experiment I.



(a) (b)

Figure 5, (a) configuration of the 3 points (A, B, and C) that we probe with the tip of the drilling bit around the main drilling point(M). M' is the perpendicular projection of point M on plane P which contains points A, B, and C. (b) group of vectors we use to define the local coordinate system on the drilling point. This coordinate system will be registered in the robot coordinate frame {KBF} and used to locate the tool coordinate system with respect to the drilling point.

To register the drilling target and the local surface normal in {KBF} we will take the following steps (see Figure 5):

- We know that a plane can be passed through 3 distinct points in 3D cartesian space. Now if we locate these three points (A, B, and C) in a specific configuration and close enough to each other around the desired drilling target point (M) which we mark on the curved surface as soon as it arrives in the assembly line, we can define a local coordinate system with the combination of these points.
- Since points A, B, and C lay on the same plane (P) we can define two vectors $\overrightarrow{AB}$ and $\overrightarrow{AC}$ and be sure that the cross product of these two vectors is also perpendicular to the plane in which these three points are located. See Figure 5. We call this vector $\vec{N}$.

$$\vec{N} = \overrightarrow{AB} \times \overrightarrow{AC} \tag{2.5}$$

which we normalize it with its length as follows:

$$\vec{n} = \frac{\vec{N}}{\|\vec{N}\|} \tag{2.6}$$

- Now we project the drilling target point M on the plane P, and we call it the point M'.
- We define the vector $\overrightarrow{BM'}$ as an auxiliary vector calling its normalized version $\overrightarrow{aux}$.

$$\overrightarrow{aux} = \frac{\overrightarrow{BM'}}{\|\overrightarrow{BM'}\|} \tag{2.7}$$

- Up to now we have 2 vectors, $\vec{n}$ and $\overrightarrow{aux}$ perpendicular to each other. If we define another vector which can complete this set with the right hand-rule with which we can define a coordinate system which from now on we will call it **L**ocal Surface **N**ormal Cartesian Coordinate **F**rame ({LNF}). Hence, we derive this completer vector named $\overrightarrow{COMP}$ by cross product of $\vec{n}$ and $\overrightarrow{aux}$. By normalizing this vector, we have the coordinate system {LNF} ready to use in the next stages.

$$\overrightarrow{COMP} = \overrightarrow{aux} \times \vec{n}$$
$$\overrightarrow{comp} = \frac{\overrightarrow{COMP}}{\|\overrightarrow{COMP}\|} \tag{2.8}$$

2.5 *Haptic Guidance*

Contemplating the method mentioned in the previous section, plane P in Figure 5, could be considered as a good approximation of tangent plane at the drilling point. Using this tangent plane at the drill target and all the vectors that we defined using the adjacent points A, B, and C around the drilling point M we

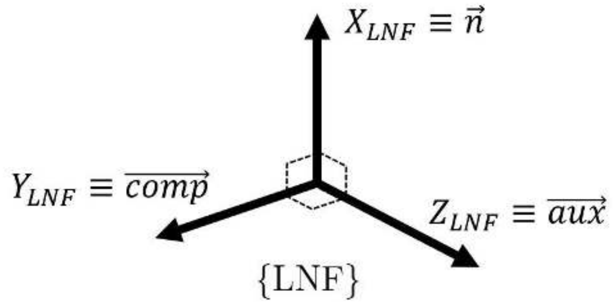


Figure 6, Cartesian coordinate frame {LNF} defined at the drilling target point. Note that the drilling target point M is located in the origin of this coordinate system.

defined the coordinate system {LNF} which has its X, Y, and Z axes aligned with vectors $\vec{n}$, $\overrightarrow{comp}$, and $\overrightarrow{aux}$, respectively (Figure 6). After defining {LNF} with its origin at drilling point M any arbitrary drilling axis can be defined with respect to it. As shown in Figure 7, a vector for drilling axis can be expressed by a pair of polar (φ) and azimuth (θ) angles.

Practically, when the operator selects the desired angle and brings the drill sufficiently close to the drill target and roughly aligns the drill bit using the visual feedback provided by the AR interface, the robot can align the drill bit precisely and quickly with the selected angle with an linear distance from the drilling point. After this alignment, the movement of the drill bit (and hence the user movement) is restricted to the desired drilling vector (axis) only. The user can then simply push the drill bit forward to open a hole on the curved surface with a safe manner. In terms of implementation there is a considerable amount of mathematical and analytic geometry involved in order to realize this self-alignment and motion confinement workflow which we will illustrate in this section.

As mentioned above, we can always define a drilling vector with a specific polar and azimuth angle pair with respect to the {LNF}. But this is not enough to do the trajectory planning and make the robot align the drill bit from the position we release it nearby the drilling point to the desired drilling angle at a desired distance away from the target ready to push.

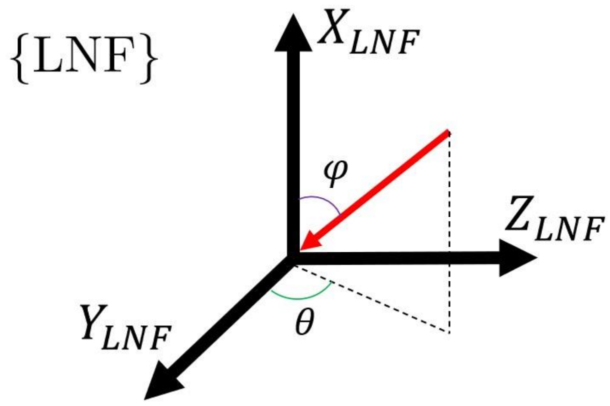


Figure 7, Defining the drilling vector (in red) using polar (φ) and azimuth (θ) angles in the {LNF} coordinate frame.

To plan the trajectory between these two poses (i.e., where we release the drill nearby the drilling point and where the drill bit is perfectly aligned with the desired drilling angle), we define a new coordinate system located at the tip of the drill bit calling it **T**ool Cartesian Coordinate Frame ({T}). {T}'s origin is located at the tip of drilling bit and its orientation is always the same as 7th Denavit-Hartenberg (D-H) coordinate frame for KUKA cobot's kinematic chain. Since the drill structure is rigid itself and is rigidly screwed to the flange of the robot, the respective position and orientation of 7th D-H system and {T} are always the same. We find this transformation matrix using KUKA's tool calibration methods available in its native operating system and we denote it as ${}^{7}_{Tool}T$. On the other hand, from analytic geometry we know that *if we have each axis of a cartesian coordinate system {A} as unit vectors which are all expressed in cartesian coordinate frame {B} and we concatenate them horizontally we will have the rotation matrix* ${}^{B}_{A}R$.

$$ {}^{B}_{A}R = \left[{}^{B}\hat{X}_{A_{3\times1}} \quad {}^{B}\hat{Y}_{A_{3\times1}} \quad {}^{B}\hat{Z}_{A_{3\times1}} \right] \tag{2.9} $$

Hence, if we concatenate each vector that shapes the {LNF} frame then we will have the rotation matrix from Local Surface Normal Coordinate Frame with respect to the KUKA Base Coordinate Frame, ${}^{KBF}_{LNF}R$. Since we have our drilling target point M probed with the robot, we have the position of drilling point M (origin of {LNF}) with respect to {KBF}. The homogenous transformation matrix will eventually be formed as

$$ {}^{KBF}_{LNF}T = \begin{bmatrix} {}^{KBF}_{LNF}R_{3\times3} & {}^{KBF}_{LNF}P_{3\times1} \\ 0_{1\times3} & 1 \end{bmatrix} \quad (2.10) $$

which we will use soon. Now that we have the ${}^{7}_{Tool}T$, if we compute the desired position and orientation of {T} coordinate frame with respect to {LNF} i.e., ${}^{LNF}_{Tool}T$ for a desired polar angle (φ), azimuth angle (θ), and distance from the drilling point (d), since we have already computed ${}^{KBF}_{LNF}T$ in (2.10) we can compute ${}^{KBF}_{Tool}T$ as below.

$$ {}^{KBF}_{Tool}T = {}^{KBF}_{LNF}T \; {}^{LNF}_{Tool}T \quad (2.11) $$

So as to make the robot move smoothly from the pose that it is parked nearby the drilling target and the pose that it is aligned with the angle of attack for drilling, we need two 7×1 sets of joint positions ($\boldsymbol{q}_i$ and $\boldsymbol{q}_f$) between which we can define a cubic spline in time domain. q_i is the current join position that the robot has and it available for us. $\boldsymbol{q}_f$ is the computed joint configuration of robot for the final aligned pose of drill bit. To compute $\boldsymbol{q}_f$ we use the inverse kinematic functions of our cobot. *The inverse kinematic should be given a desired pose of our cobot (${}^{KBF}_{7}T_{des}$) to give us the desired joint angles (q_{des})*. Hence, still by having ${}^{KBF}_{Tool}T$ we are not able to plan the trajectory between the two aforementioned poses. To do so we take advantage of the invertibility feature of homogenous transformation matrices to find ${}^{KBF}_{7}T_{des}$ as follows.

$$ {}^{KBF}_{7}T_{des} = {}^{KBF}_{Tool}T_{des} \; {}^{7}_{Tool}T^{-1} \quad (2.12) $$

By feeding the matrix ${}^{KBF}_{7}T_{des}$ to inverse kinematics we can find the desired join position array of our cobot in the aligned pose ($q_f = q_{des}$).

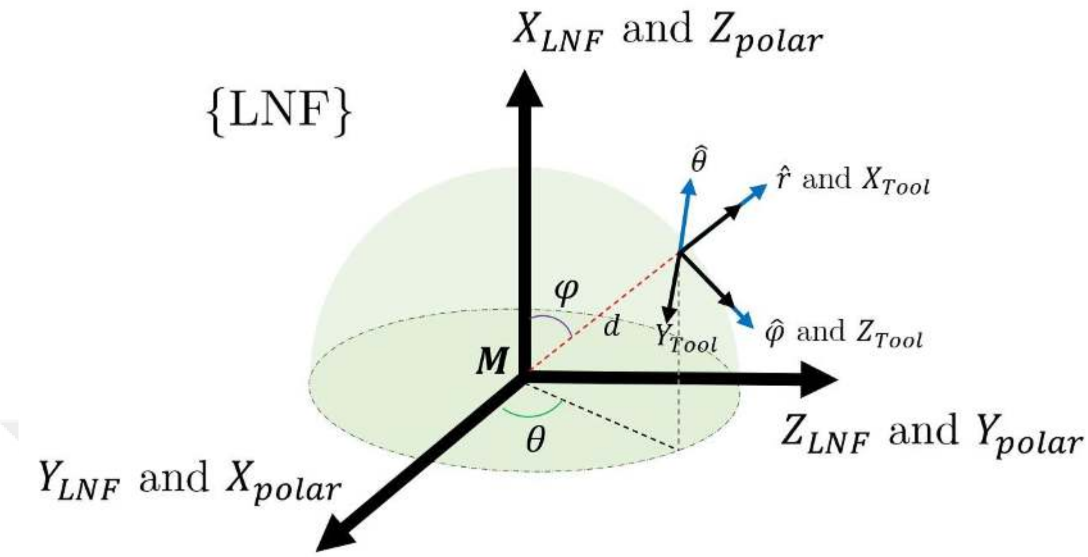


Figure 8, Breaking down all the coordinate systems that we use to define ${}^{LNF}_{Tool}T$. Drilling target point M. Green hemisphere has a radius of d. Our goal is to locate the tool coordinate frame {T} at the desired point on the hemisphere. $\hat{r}$, $\hat{\theta}$, and $\hat{\varphi}$ are the conventional unit vectors of polar 3D coordinate system.

So far, we took ${}^{LNF}_{Tool}T$ as granted by knowing our desired polar angle (φ), azimuth angle (θ) and distance (d) to the drilling point. Now, it is time to delve into the computations of this section. If we consider a virtual sphere having it's center on drilling target point M with radius $\boldsymbol{d}$ then we can locate the tool coordinate system {T} on the surface of this sphere having it's X axis pointing outward the sphere, Y, and Z axes would also be defined as illustrated in Figure 8. To compute ${}^{LNF}_{Tool}T$, we need rotation matrix of {T} with respect to {LNF}, ${}^{LNF}_{Tool}R$, and position vector of origin of {T} expressed in {LNF}, ${}^{LNF}_{Tool}P$. Computing the position vector having polar and azimuth angle as well as the distance d is straight forward.

$$ {}^{LNF}_{Tool}\boldsymbol{P} = [d\,cos(\varphi) \quad d\,cos(\theta)\sin(\varphi) \quad d\,sin(\theta)\sin(\varphi)]^T \tag{2.13} $$

For computing the rotation matrix, we take advantage of the 3D polar coordinate system's unit vectors ($\hat{r}$, $\hat{\theta}$, *and* $\hat{\varphi}$) definitions in cartesian space. From analytic geometry we know the expression of each of these three unit vectors in the $X_{polar} - Y_{polar} - Z_{polar}$ coordinate system as follows.

$$^{polar}\hat{r} = [cos(\theta)\sin(\varphi) \quad sin(\theta)\sin(\varphi) \quad cos(\varphi)]^T \tag{2.14}$$

$$^{polar}\hat{\theta} = [-\sin(\theta) \quad \cos(\varphi) \quad 0]^T \tag{2.15}$$

$$^{polar}\hat{\varphi} = [\cos(\theta)\cos(\varphi) \quad \sin(\theta)\cos(\varphi) \quad -\sin(\theta)]^T \tag{2.16}$$

Considering Figure 8, we can notice the following relation between the polar coordinate system unit vectors and the {T} unit vectors.

$$^{polar}X_{Tool} = {}^{polar}\hat{r} \tag{2.17}$$

$$^{polar}Y_{Tool} = -{}^{polar}\hat{\theta} \tag{2.18}$$

$$^{polar}Z_{Tool} = {}^{polar}\hat{\varphi} \tag{2.19}$$

We should not forget that our goal is to find $^{LNF}_{Tool}R$. Since we have both the polar unit vectors ($^{polar}\hat{r}$, $^{polar}\hat{\theta}$, and $^{polar}\hat{\varphi}$) and our desired tool coordinate system {T} unit vectors ($^{polar}X_{Tool}$, $^{polar}Y_{Tool}$, and $^{polar}Z_{Tool}$) expressed in the $X_{polar} - Y_{polar} - Z_{polar}$ coordinate system shown in Figure 8 we should express them in {LNF} to be able to use them in the later steps of our calculations.

$$X_{LNF} = Z_{polar} \tag{2.20}$$

$$Y_{LNF} = X_{polar} \tag{2.21}$$

$$Z_{LNF} = Y_{polar} \tag{2.22}$$

Considering (2.14) to (2.20) we can define each of the axes of {T} coordinate system located with particular polar and azimuth angles with respect to {LNF}.

$$^{LNF}X_{Tool} = [\cos(\varphi) \quad \cos(\theta)\sin(\varphi) \quad \sin(\theta)\sin(\varphi)]^T \tag{2.23}$$

$$^{LNF}Y_{Tool} = [0 \quad \sin(\theta) \quad -\cos(\theta)]^T \tag{2.24}$$

$$^{LNF}Z_{Tool} = [-\sin(\varphi) \quad \cos(\theta)\cos(\varphi) \quad \cos(\varphi)\sin(\theta)]^T \tag{2.25}$$

According to the principle mentioned in equation (2.9) we can generate ${}^{LNF}_{Tool}R$ as follows.

$$ {}^{LNF}_{Tool}R = \left[{}^{LNF}X_{Tool} \quad {}^{LNF}Y_{Tool} \quad {}^{LNF}Z_{Tool} \right] \tag{2.26} $$

Eventually, having ${}^{LNF}_{Tool}R$ and ${}^{LNF}_{Tool}P$ we can derive ${}^{LNF}_{Tool}T$.

$$ {}^{LNF}_{Tool}T = \begin{bmatrix} {}^{LNF}_{Tool}R_{3\times3} & {}^{LNF}_{Tool}P_{3\times1} \\ 0_{1\times3} & 1 \end{bmatrix} \tag{2.27} $$

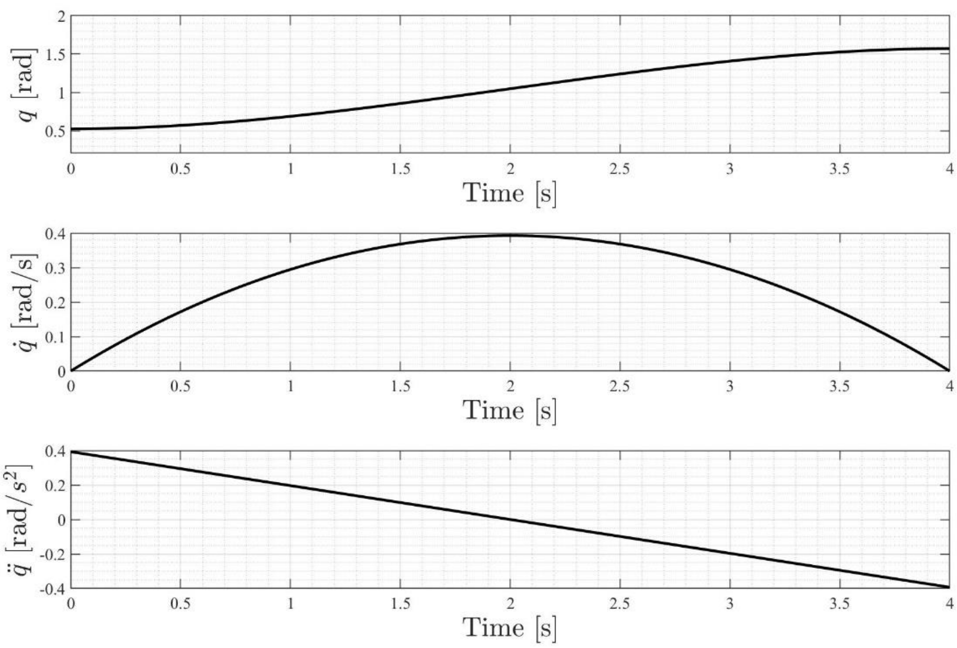


Figure 9, Sample of a cubic spline in time-domain that we use to design the trajectory between start and stop positions of auto alignment phase.

This means that for a desired polar angle, azimuth angle and distance from the drilling point M, we can compute the desired ${}^{KBF}_{Tool}T$ in (2.11) and feed its corresponding ${}^{KBF}_{7}T$ to the inverse kinematics of our cobot and get the desired joint angle configuration. As mentioned earlier by having $q_f = q_{des}$ and $q_i = q_{current}$ we can design a cubic spline in time domain with the general form as below.

$$ \boldsymbol{q}(t)_{7\times1} = \boldsymbol{a_0}_{7\times1} + \boldsymbol{a_1}_{7\times1}t + \boldsymbol{a_2}_{7\times1}t^2 + \boldsymbol{a_3}_{7\times1}t^3 \tag{2.28} $$

By deriving any of the $\boldsymbol{a}_{i_{7\times1}}$where $i \in \{1,2,3\}$ using the total time interval, position, and zero velocity initial and final conditions; we will have a desired trajectory for each of joint motors to follow. This cubic time-domain spline guarantees a smooth trajectory between the place we release the drill and the pose that the drill is aligned by the robot perfectly with the desired angles of attack and distance. As an example, you can see the trajectory of a joint in Figure 9. Just the same as joint trajectory depicted in this figure, all joints follow a smooth trajectory which starts from zero velocity and ends at zero velocity as well.

After the drill is parked at the desired pose with respect to point M, only the 1st motion confinement control gate (associated with X direction of force sensor) shown in Figure 4 is activated and the rest are off. This will allow the admittance controller to only execute based on the force readings in X direction of {LFF} which is aligned with the $\hat{X}_{Tool}$ vector of {T} frame confining the operator's motion solely in the prescribed drilling direction.

2.6 *Admittance Adaptation*

In order to cope with the varying requirements of the tasks in its different phases, adaptive admittance control is utilized in this study. As long as the drill is far from the drill target, low admittance damping is used to keep the system transparent to human input, so the robot can be driven faster in free space. To prevent accidental contacts with the workpiece and help the user roughly align the drill bit in a more controlled manner, medium admittance damping was used when drill bit is sufficiently close (10 cm) to the target. Once the drill bit is at the 5-cm vicinity of the target, the robot takes over the control and aligns the drilling axis precisely. After this point, as mentioned in the previous subsection, the user is constrained to move along the drilling axis only under high admittance damping for maximum stability during drilling to open a hole.

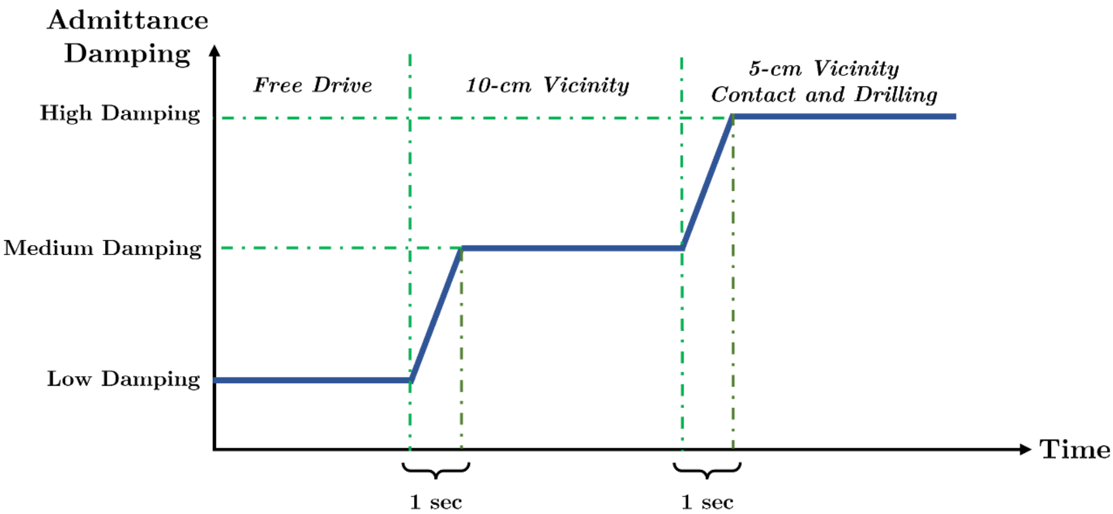


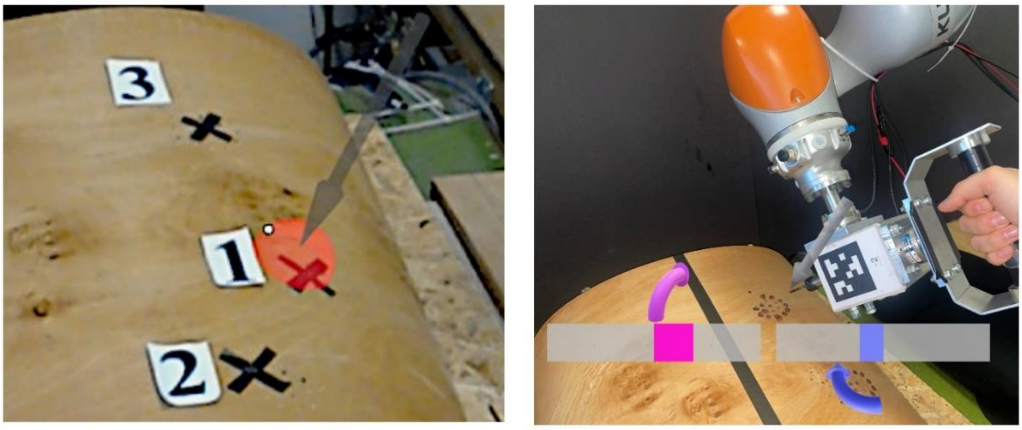
Figure 10, The time-domain profile of adaptive damping. Please note that the values are only for visual representation.

It is noteworthy that transition of damping values follows a ramp profile with a length of 1 second (Figure 10). Using the analysis conducted in our earlier work [23] we observe that this will not lead to instability of the pHRI system. The values used for admittance damping in this study are tabulated in Table 1.

Table 1, Control parameters used in the 6D drilling experiments.

Task Phase	**Degrees of Freedom**	**Translational Mass [Kg]**	**Translational Damping [Ns/m]**	**Rotational Mass [kgm²]**	**Rotational Damping [Nms]**
Free Motion	6	50	100	10	5
Close to Target	6	50	600	10	20
Drilling	1	50	1000	-	-

2.7 Augmented Reality Interface



(a) (b)

Figure 11, The visual feedback displayed to the user by the augmented reality (AR) interface under (a) haptic guidance and (b) no haptic guidance. In (b) the pink(blue) vector is guiding the user to the direction to which he has to tilt the drill to reduce the error bar on the left(right) for polar(azimuth) angle. In both pictures the whiteish-gray straight arrow shows the angle desired angle of attack on top of the drilling spot of interest. Note that the small offset that you can see in (a) between the main drilling target and the location of red sphere is due to the fact that the picture is a screen shot taken by HoloLens' on-board single lens camera. In fact, what the human operators see with two eyes (stereo vision effect) the arrow and hemisphere are precisely on top of the drilling point.

The AR interface used in this study assist the operator during the drilling task by superimposing visual images on top of real ones (Figure 11). In a manufacturing environment involving a scenario such as ours, it can be difficult for the operator to remember the steps of the whole task and the sequence of drilling targets. In our implementation, the text and images displayed trough the AR interface guide the user throughout the task. Furthermore, it can be hard to visualize how one should roughly align the drill bit in 3D space before the final and precise alignment made by the robot. A white colored and semi-transparent arrow with its tip pointing to the drill target is displayed through the AR interface to help the user with the rough alignment and reduce the task execution time. Furthermore, the AR interface highlights the current drill target by displaying a red colored hemisphere and its location (Figure 11: a). This hemisphere also informs the operator about how close the drill tip is to the target before the robot takes over the control for finer alignment of the drilling axis.

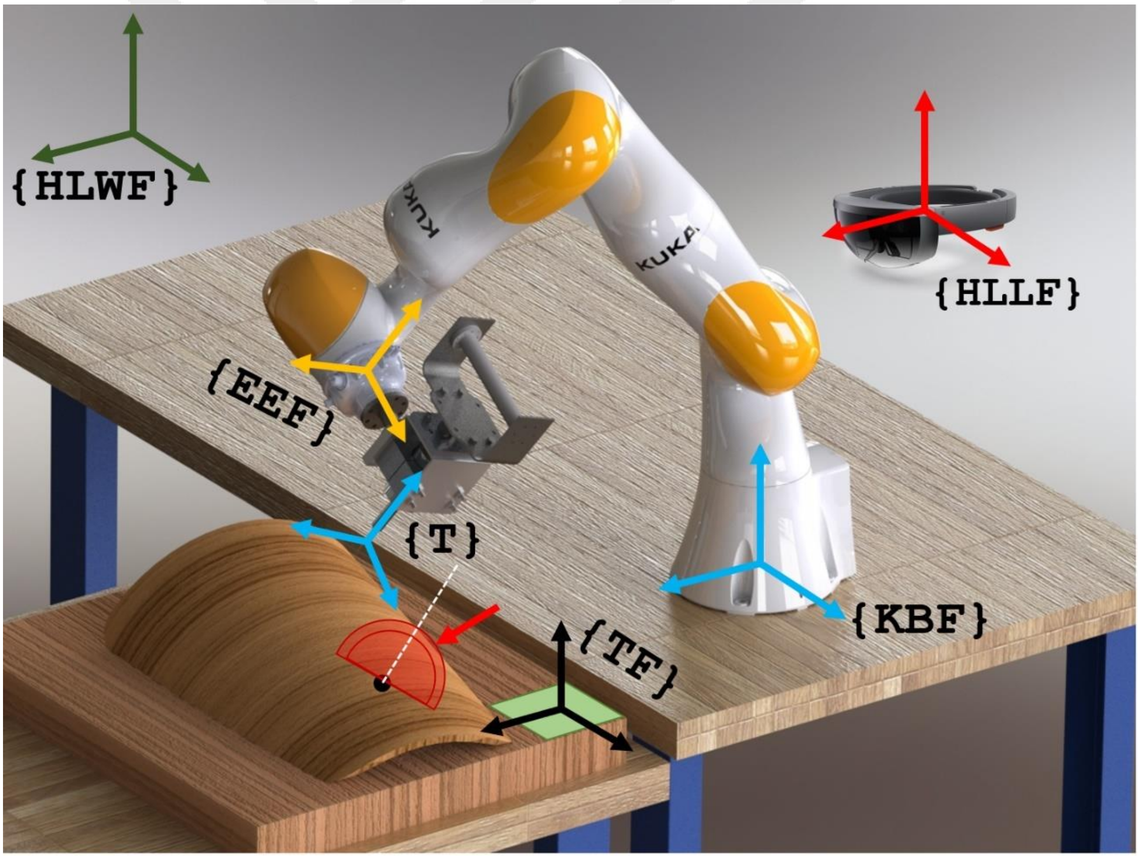


Figure 12, The coordinate systems that we have used in the design of our HoloLens application. {HLWF}: HoloLens World Frame, {HLLF}: HoloLens Local Frame, {EEF}: Robot's End Effector Frame, {T}: Tool Frame, {KBF}: KUKA Base Frame, {TF}: AprilTag Frame.

What mentioned above depicts the general functionality of augmented reality interface that we utilized in this work. In this part of this subsection, we will shed light into the inner workings of this component of our application. The first thing to get familiarized with is the spatial mapping and visualization module of Microsoft HoloLens. At the startup time, HoloLens' operating system using its two short-throw and long-throw depth sensing sensors as well as its 3 cameras defines a static cartesian coordinate frame named HoloLens World Frame ({HLWF}) and anchors it somewhere in the room randomly (Figure 12). So basically, what HoloLens does is instantiating virtual objects with a prescribed position and orientation with respect to {HLWF}. To show these virtual objects to the user, HoloLens uses its on-board gyroscopes, accelerometers, depth and visual sensors data to compensate for the user's movements i.e., estimate the position and orientation of HoloLens Local Frame ({HLLF}). Taking into account these translation and orientation compensations HoloLens updates the scene shown its transparent displays (visors). On the other hand, as mentioned in subsection 2.4, with whichever method we use to register the drilling point in the robot base coordinate system {KBF} we will have the position and orientation of the drilling vector, i.e., desired pose of tool frame {T} with respect to this {KBF} frame. The HoloLens operating system needs the homogenous transformation matrix of a coordinate frame attached to any virtual object with respect to {HLWF} to be able to show it on its visor. Since the position and orientation of our desired tool frame {T} (corresponding to our desired drilling angle) is expressed in the {KBF} we use a visual marker named AprilTag [24] to define the position and orientation of {T} with respect to {HLWF}. First, we probe the 4 corners of the AprilTag and using positions of these 4 coordinates we define a coordinate frame attached to the visual tag named Tag Frame ({TF}). Hence

${}^{KBF}_{TF}T$ is computable. Since we have already derived ${}^{KBF}_{Tool}T_{des}$ based on our desired polar and azimuth drilling angles for each of the drilling points on the workpiece, we are able to find ${}^{TF}_{Tool}T_{des}$ which is what we need to represent any of the white colored arrows as well as the red hemisphere described in the beginning of this subsection. As it will be meticulously explained in the next chapter, in order to showcase the merit of presence of self-alignment module of our system, we have designed experiments in which in one scenario the operators are asked to manually align the drill by moving the drill in 6D with the desired angles of attack shown to them through the AR interface. In these experiments in order to assist the operator with alignment to the desired drilling angle in each of the 3 drilling points on the workpiece, we provided them visual feedback of their error in terms of polar (ε_{φ}) and azimuth (ε_{θ}) angles using error bars as well as some instructive arrows which tell them how to move the drill to reduce the error(Figure 11: b).

$$\varepsilon_{\varphi} = \varphi - \varphi_{des} \tag{2.29}$$

$$\varepsilon_{\theta} = \theta - \theta_{des} \tag{2.30}$$

The C++ software that we have developed which runs the robot, has a **live polar and azimuth angle monitoring module**. Basically, these series of functions available in the code computes the polar and azimuth angles of the drilling vector depicted in Figure 7 in real-time while the operator is moving the drill and tries to adjust the angle of attack manually with the desired drilling vector that he/she is seeing through the AR goggle. We use TCP connection protocol to establish a real-time dataflow between the KUKA computer and HoloLens' onboard computer to transfer the error values for both polar and azimuth drilling angles. The error data is interpreted in the application that we have developed for HoloLens using Unity Software and UWP[2] to show the

[2] Universal Windows Platform

graphics that you see in Figure 11: b. Inner working of the **live polar and azimuth angle monitoring module** is described in great detail with all the associated computations and equations in the appendix of this thesis.

Chapter 3:
EXPERIMENTS

3.1 Experiment I

In this experiment, the goal is to investigate the potential benefits of haptic guidance in task performance by comparing it with no guidance condition (i.e., manually aligning the drill bit with the desired drilling angle). Three naïve subjects with an average age of 29.7 ± 4.6 years participated in this experiment. To monitor the muscle activation levels of participants during the task execution, surface electromyography (sEMG) sensors were attached to their Flexor Capri muscle. The procedures for sensor placement and data acquisition are identical to those presented in [25].

3.1.1 Experimental Conditions

There were two experimental conditions in this experiment: 1) haptic guidance by the robot and limited visual guidance by the AR interface and 2) no haptic guidance by the robot but more visual guidance by the AR interface with respect to the first condition. Under the second condition, the participant had to manipulate the drill manually to bring it to the drill target and align it with the desired drilling angle. Because there was no automated alignment, the system was not restricted to 1D motion in the drilling phase, unlike the haptic guidance condition (condition 1). Under this condition, again for giving haptic feedback to the operator and easing the subtle orientation adjustments, as soon as the drill bit gets closer than 10 cm to the drilling point, the damping increases to its high value (b = 1000 Ns/m) with a 1 second ramp profile. This ensures safe and stable contact interaction with our workpiece as well. Also, there was no red hemisphere displayed on top of the drill target in condition 2 since there participant needed to visually see the exact target location without any visual obstacles on the way. The white transparent arrow remained still active under this condition, pointing to the drill target, and showing the axis with which, the participants should align the robot as precisely as they can. Furthermore, to compensate for the lack of haptic guidance in condition 2, the orientation errors with respect to the desired azimuth (θ) and polar (φ) angles were displayed as horizontal bars, which were updated in real-time as the user manipulated the drill bit and rotate it in space. Finally, two virtual arrows (one for the azimuth angle and one for polar angle) showed which way the drill needs to be rotated to get closer to the desired orientation. The values for the parameters used in this experiment are tabulated in Table 2.

Table 2, Parameters used in experiment 1.

Item	Value	Unit
Control adaptation distance from drill target	10	cm
Locking distance from the drill target (radius of red hemisphere)	5	cm
Distance from target after alignment	5	cm
Autopilot alignment time	4	sec
First drill target polar angle $\boldsymbol{\varphi_1}$	5	deg
First drill target azimuth angle $\boldsymbol{\theta_1}$	0	deg
Second drill target polar angle $\boldsymbol{\varphi_2}$	30	deg
Third drill target polar angle $\boldsymbol{\varphi_3}$	45	deg
Second and third drill target azimuth angles $\boldsymbol{\theta_2}$ and $\boldsymbol{\theta_3}$	10	deg

3.1.2 Protocol

We first present the protocol followed by the participants under the haptic guidance condition below (see Figure 13).[3]

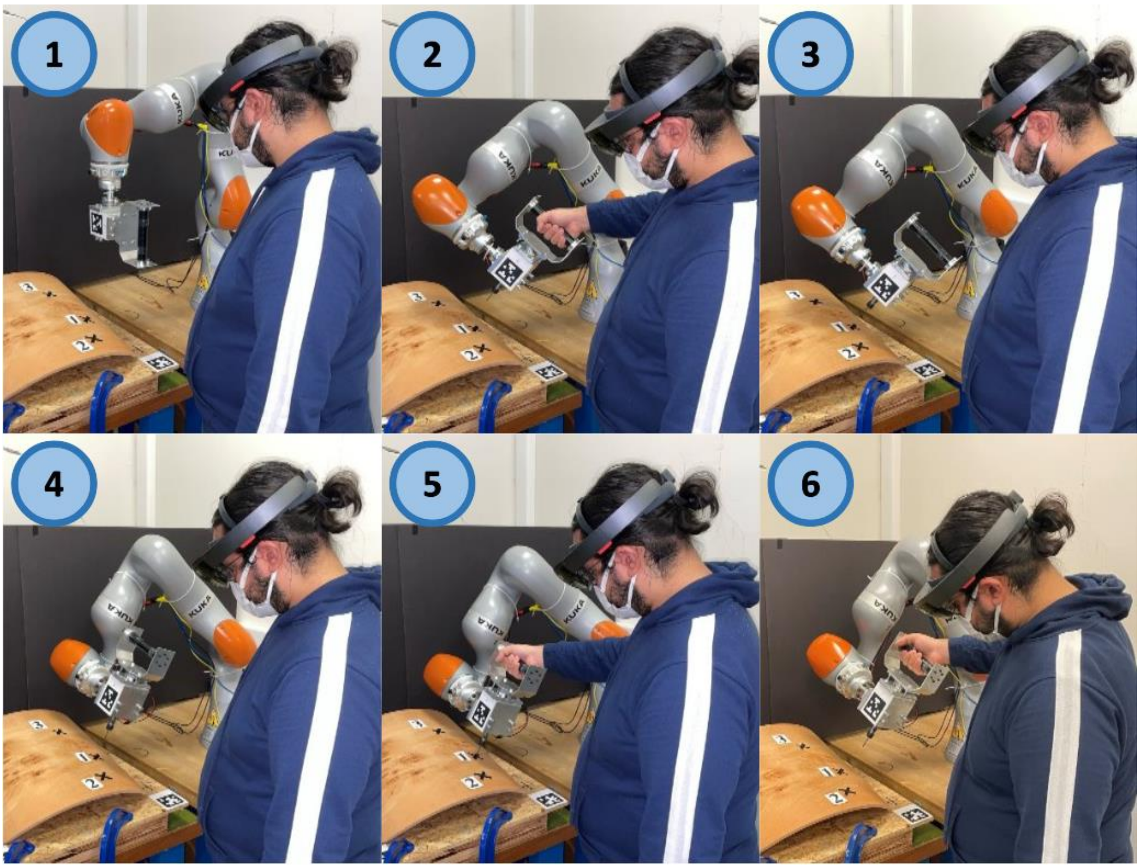


Figure 13, Steps followed by the participants in Experiment 1 under haptic guidance condition.

1) The participant approaches the robot, and stands on a marked space on the ground, waiting for the system to initialize (Figure 13: 1).
2) After being told to so do so, the participant grabs the handle of the robot, and moves the drill bit close to a red transparent hemisphere shown in the AR google (Figure 11: a), while roughly aligning it with a white

[3] A demonstrative video of the proposed system can be found at https://www.yusuf-aydin.com/iros-2022-robot-assisted-drilling/

transparent arrow (i.e. desired drilling axis) displayed through the AR interface (Figure 13: 2).

3) When the drill tip touches the red virtual hemisphere(r = 5 cm), the robot takes over the control, while the message "Please do not touch the robot" is displayed to the participant through the visor of the AR interface. The participant stands clear of the robot (Figure 13: 3).
4) The robot aligns the drill bit with the drilling axis automatically in 4 seconds (Figure 13: 4).
5) When the alignment is over and the robot is ready to be guided towards the target for drilling, the user is restricted to 1D forwards/backwards movements along the drilling axis. At this point, the message "Continue..." is displayed to the participant through the AR goggle.
6) The participant gently grabs the handle again and simply pushes the drill into the workpiece to open a hole. The haptic guidance provided by the robot keeps the movement on the drilling axis. After drilling the hole, the participant retracts the drill out of the workpiece by moving back along the same axis again. (Figure 13: 5-6).

After the first point is drilled, the user is ready to go to the next drilling point. The 6-DoF admittance controller is reactivated, the hemisphere and arrow are displayed at the next target location, and the same steps are repeated for the new drill target.

Under condition 2 (no haptic guidance), the participants were asked to align the drill manually as accurately as possible using the white transparent arrow pointing to the drill target only. Moreover, the red hemisphere was not displayed to the participants through the AR interface at the drill target to prevent the obstruction of their view. Furthermore, to assist the user during alignment, angular deviations from the desired drilling angles were displayed through the AR interface in the form of real-time error bars as well as guiding arrows which help them correct those deviations (Figure 11: b). These features (horizonal bars and guiding arrows) were not displayed to the participants under haptic guidance condition (Figure 11: a).

Before the actual experiments, participants were given two training trials, executed with and without haptic guidance, to familiarize themselves with the setup and the experimental procedures, without opening a hole on the workpiece. Then, each participant performed two trials of each condition, amounting to 4 trials in total; two trials with haptic guidance and two without haptic guidance.

3.1.3 Results

The quantitative metrics used for comparing the experimental conditions (with and without haptic guidance) are the following:

- Task completion time, t_{tot} [sec]
- Average linear speed, $s_{lin}^{avg} = \frac{1}{t_{tot}} \int_0^{t_{tot}} \|\mathbf{v}_{tool}\| dt$ [m/s]
- Average angular speed, $s_{ang}^{avg} = \frac{1}{t_{tot}} \int_0^{t_{tot}} \|\boldsymbol{\omega}_{tool}\| dt$ [rad/s]
- Average Flexor Carpi muscle activation, $V_{avg} = \frac{1}{t_{tot}} \int_0^{t_{tot}} v(t)\, dt$ [Volts]
- Human effort caused by forces, E_F [J]
- Human effort caused by torques, E_τ [J]
- Total human effort, E_{tot} [J] $= E_F + E_\tau$
- Average alignment error, $\varepsilon_\varphi^{avg}$ for polar, and ε_θ^{avg} for azimuth angle [deg]

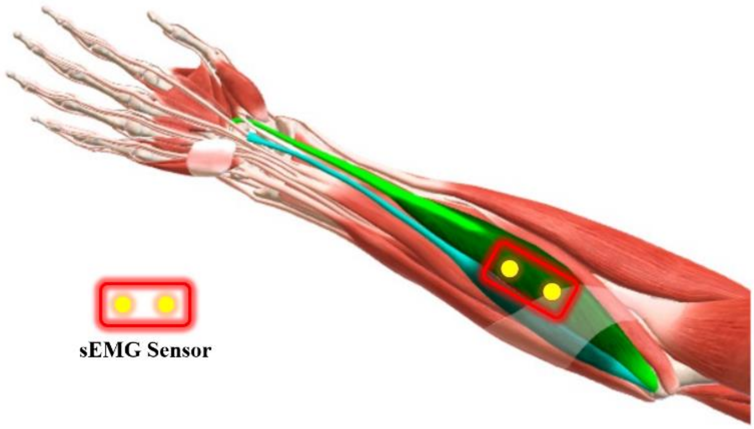


Figure 14, We attached the sEMG sensor on the Flexor Carpi (colored green) muscle of right hand. All our participants were right-handed. When the sensor is connected to the region shown in the picture, it reads the activations mainly generated from the Flexor Carpi muscle. Since the Palmaris Longus muscle (blue colored) is located underneath the Flexor Carpi and generally behaves synchronously with this muscle the signal that we read from this sensor is clear, distinctive and promising for showing the degree to which the human is getting involved neuromotorically with the task.

Task completion time t_{tot} refers to the entire time span of a drilling session where the participant has to use the robot and the AR interface to drill 3 holes on the curved surface. Since it takes 4 seconds per hole (Table 2) for the robot to align the drill to the desired orientation under the haptic guidance condition, a total of 12 seconds was spent by the robot for the alignment process. Linear (angular) speed s_{lin}^{avg} (s_{ang}^{avg}) is simply the mean magnitude of the linear (angular) velocity vector $\mathbf{v}_{tool}$ ($\boldsymbol{\omega}_{tool}$) of drill bit throughout one full session (for the haptic guidance mode we exclude the 12 seconds from the timespan). Average alignment errors $\varepsilon_{\varphi}^{avg}$ and $\varepsilon_{\theta}^{avg}$ basically the average of absolute value of errors for each of these angles, are calculated at the time of drilling, just before the drilling starts for every one of the three drilling target points. This metric is obviously close to zero when the haptic guidance is enabled since the robot automatically brings the drill to the desired orientation. To calculate the total human effort, E_{tot}, first, each component of the wrench vector, $\mathbf{F}_h = [F_h^x, F_h^y, F_h^z, \tau_h^x, \tau_h^y, \tau_h^z]$, is multiplied with the corresponding component of velocity vector, $\mathbf{v} = [v_x, v_y, v_z, \omega_x, \omega_x, \omega_x]$, and sum of the absolute value of aforementioned element-wise products are computed which gives us the instantaneous human mechanical power. Ultimately, this instantaneous power is integrated over time as follows.

$$E_{tot} = \int_{t_{begin}}^{t_{end}} \sum_{i=1}^{6} \left| F_h^i v^i \right| dt \quad (3.1)$$

Average muscle activation V_{avg} is simply the mean of activation levels of the Flexor Carpi muscle as measured and postprocessed ([25]) by the sEMG sensors, throughout each trial (Figure 14). It is noteworthy that this muscle is mainly contracted when generally we are trying to do fine motor actions with our hands. In our task, fine tuning the orientation of the drill is a task which demands precise forces to be exerted from our hand to the drill handle. Hence, as we try to be more cautious with magnitude and direction of the wrench, we are exerting to the robot we contract the muscles in this region more.

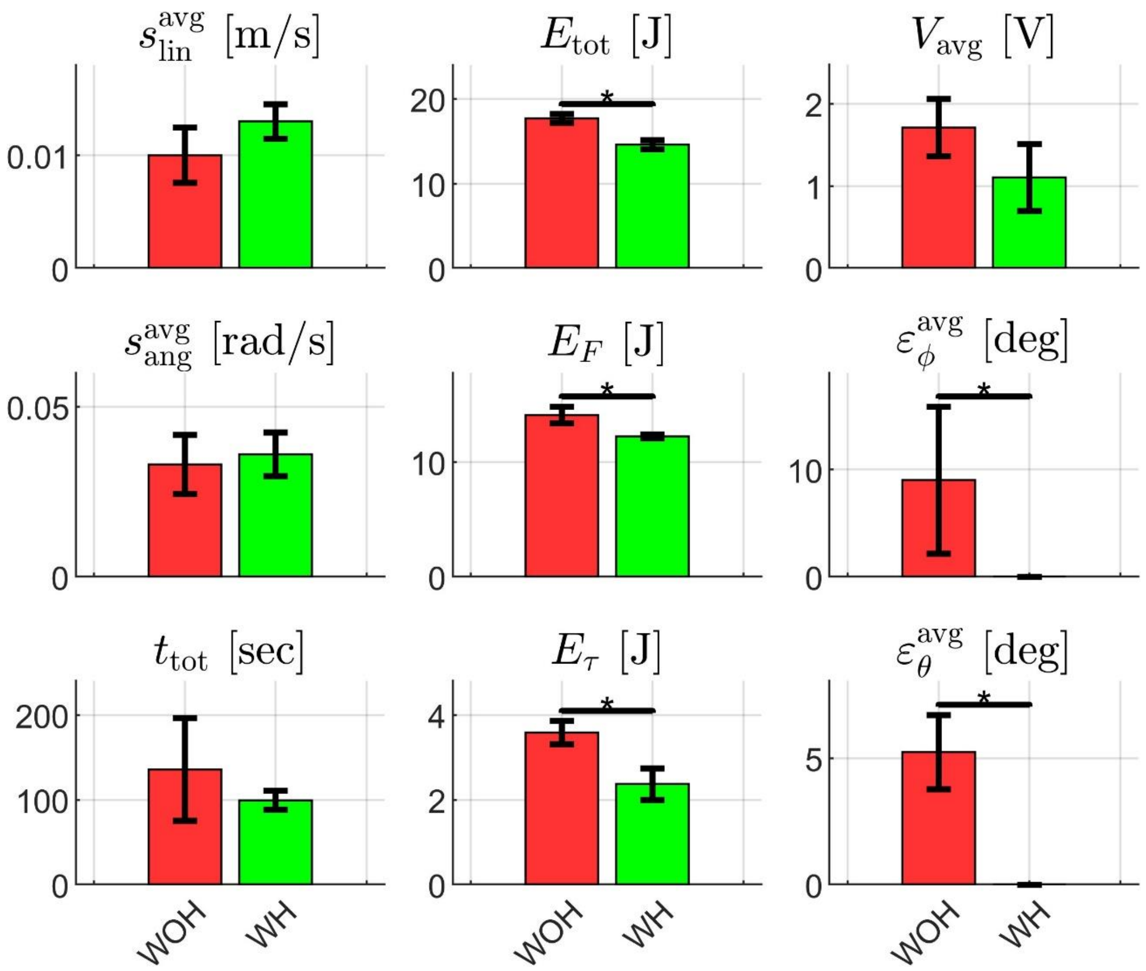


Figure 15, Performance metrics for Experiment 1, with and without haptic guidance (labeled as "WH" and "WOH" respectively). Vertical bars show population means, and vertical lines show 95% confidence intervals. Horizontal lines with asterisks on top show statistically significant pairwise comparisons with p = 0.05 as the threshold.

Figure 15 shows the average task performances of the participants under both experimental conditions based on the aforementioned metrics. To test the null hypothesis, one-way ANOVA (Analysis of Variance) with repeated measures was performed by considering the experimental conditions (with/without haptic guidance) as the main factor. According to the ANOVA results, the effect of the experimental conditions was significant on human effort (the middle column in Figure 15) and average alignment error ($\varepsilon_{\varphi}^{avg}$ and $\varepsilon_{\theta}^{avg}$). Specifically, both linear and angular components of the human effort were significantly lower under haptic guidance. The alignment errors were between 3° and 16° without haptic guidance.

According to Figure 15, average speeds, s_{lin}^{avg} and s_{ang}^{avg}, are generally higher under guidance (though the difference between the conditions is not statistically significant), leading to a lower task completion time t_{tot}. The same applies to average muscle activation V_{avg}, whose population means is lower under haptic guidance, but the difference between the conditions is not statistically significant. A large amount of variance can be observed between in task completion time t_{tot} in Figure 15 when not using haptic guidance since the whole alignment is done manually by the participant, and the time it takes to do the alignment can vary considerably between the trials of the same participant and also between the participants.

3.2 Experiment II

In this experiment, the objective was to examine the acceptability of the proposed pHRI system and the drilling scenario, if deployed in real manufacturing settings and used by real industrial workers. For this reason, three industrial workers with extensive experience in small batch manufacturing tasks including drilling were invited to our lab for experimentation. They were asked to open the wholes on curved surface under haptic guidance condition only. The target locations and drilling angles were the same as the one used in Experiment 1 (Table 2). Following the experiment, the participants were asked to fill out a detailed questionnaire designed to acquire their subjective opinions about the developed system and its applicability in real manufacturing settings.

3.2.1 Protocol

The protocol followed in this experiment is identical to the one mentioned in section 3.1 for the haptic guidance condition. After the training, every participant performed one trial (3 holes) of the experiment under haptic guidance only.

3.2.2 Questionnaire

Methods suggested by [26] and [27] were utilized for designing a subjective questionnaire[4] that utilizes a 7-point Likert scale for collecting the personal opinions of the participants about the developed pHRI-based drilling system. A series of statements were put forth in the questionnaire, with which the participants can specify to what extent they agree. The 7-point Likert scale varied from a scale of 1 (strongly disagree) to 7 (strongly agree). The statements were arranged in a fixed but shuffled order, and asked twice with different wordings, meaning every statement also as a paraphrased opposite pair in the questionnaire.

In the first part of the questionnaire, demographic data were collected from the participants about their gender, age, and years of work experience in the manufacturing industry. This section also included a question asking whether the participant believes drilling holes on a curved surface using a manual drill is difficult, and if so, why.

In the second section of questionnaire, first, four general questions were asked about how much prior experience the participant had with mechanical tools (such as screw drivers, chain saw, etc.), robots, and AR/VR interfaces in general. Then 11 questions about our pHRI system and 3 questions about our AR interface were asked in particular, along with their opposite pairs. Therefore, this section of

[4] Full text of the questionnaire can be found at https://www.yusuf-aydin.com/iros-2022-robot-assisted-drilling/#questionnaire

questionnaire included a total of 2 × (11 + 3) + 4 = 32 questions. The questions in this section were related to the following topics:

- Personal prior experience with: non-motorized devices (hammer, screw driver, etc.), motorized devices (drill, chain saw, polisher, etc.), robotic arms, and augmented/virtual reality interfaces
- Using the robot in the experiments: Overall task success, ease of use, being in control, task performance, trust, feeling natural, enjoyment, safety, fear, exhaustion, willingness to use again
- Using the AR interface in the experiments: Ease of use, enjoyment, and utility (helpfulness).

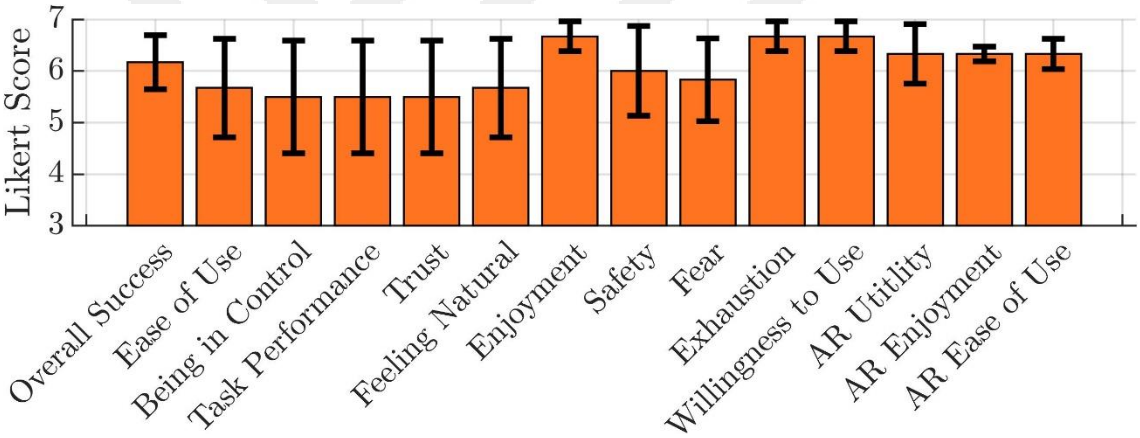


Figure 16, Subjective questionnaire results in Experiment 2. Bars indicate means and vertical lines indicate standard deviations.

3.2.3 Results

1) *Personal Background:* The average age of the workers was 31.67 ± 8.62 years. They had 15.67 ± 9.02 years of experience in the manufacturing industry. None of the participants stated that they encountered any problem during the experiments. 2 out of the 3 participants states that drilling holes on curved surfaces with custom drilling angles is hard because it is difficult to manually align the drill to a defined orientation

by hand and keep the angle of attack the same throughout the drilling. All 3 participants stated that they had extensive experience with non-motorized mechanical devices (Strongly Agree; 7.0/7.0 scores were given). The average Likert score for experience with robotic arms was 6.7/7.0, indicating that the participants had worked with robotic arms before. The mean score for prior experience with AR interfaces was 3.7/7.0. The participants later indicated that they became familiar with AR/VR interfaces as they have attended technical fairs and exhibits.

2) *Subjective Questionnaire:* Figure 16 shows that the mean response of the 3 participants to the questions. Despite low scores occasionally, all questions were scored higher than 5.0/7.0 on average by the participants, suggesting that they generally had a positive opinion about our pHRI system. According to the results shown in Figure 16, the participants on average found the robot easy to use, safe, convenient, and natural. They were not afraid of using the robot and were willing to use it again. The questions about the AR interface received scores on average higher than 6.0/7.0, suggesting that the participants found the AR goggle to be user friendly and helpful in carrying out the task.

Chapter 4:

DISCUSSION AND CONCLUSION

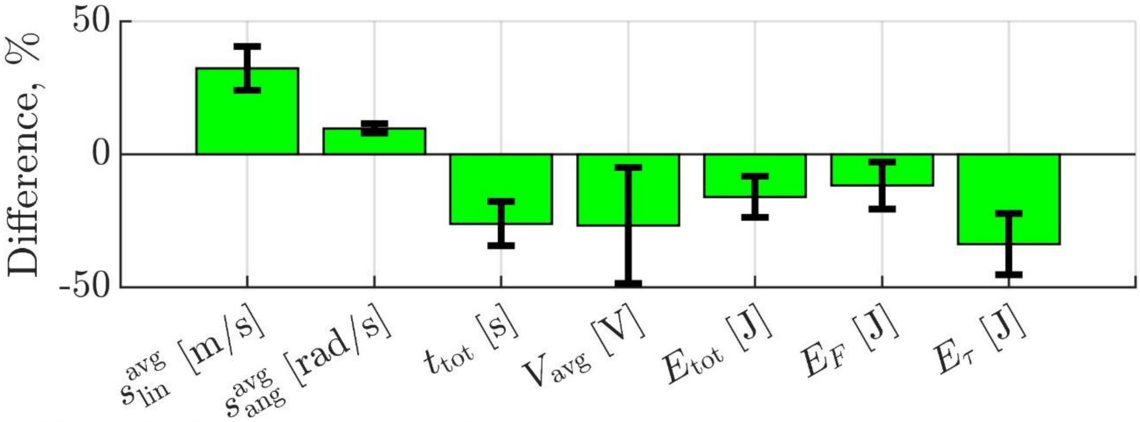


Figure 17, Percent relative difference in the performance metrics of the participants under experimental conditions of with and without haptic guidance in Experiment 1. The negative (positive) values in the plot indicate decrease (increase) in the metrics of haptic guidance condition with respect to the no haptic guidance condition. Vertical bars show average differences among the three subjects, and vertical lines indicate their standard deviations.

In this study, a pHRI system was developed to perform small-batch manufacturing tasks efficiently using haptic guidance, an AR interface, and an adaptive admittance controller. As a case study, collaborative drilling of holes on a curved surface was selected. The proposed system provides haptic guidance to the user by constraining the user's movement to the desired drilling axis and visual guidance through the AR interface for the steps of the drilling task and rough alignment of the drill bit to the desired angle. Furthermore, the damping parameter of the admittance controller that regulates the interaction between the robot and the human was adapted based on the instantaneous position of drill tip with respect to the location of drill target. During the free motion, in which the user brings the drill to the 5-cm vicinity of target point, low admittance damping was utilized so that the robot showed minimal resistance to the human intended motion. On the other hand, high admittance damping was used during the drilling phase for stable and safe operation.

The benefits of haptic guidance were investigated in Experiment 1 by comparing the task performance of the participants with and without haptic guidance, quantitatively. In order to compensate for the lack of haptic guidance under no haptic guidance condition, more visual guidance was provided to the participants through the AR interface. Since the participants manually adjusted the drill bit by themselves under no haptic guidance condition, the angular deviations of drill bit from the desired drill orientation were computed in real-time and displayed through the AR interface and guiding arrows were used to help the participant make corrections for those deviations.

The results of Experiment 1 showed that the task performance under haptic guidance was superior to that of the no haptic guidance condition. Figure 17 presents the relative differences in percentage between the performance metrics of the two conditions. According to this figure, the haptic guidance condition led to 26% shorter task completion time and 16% less human effort. Furthermore, 27% less muscle activation was observed in the participants under the haptic guidance condition, suggesting that the haptic guidance could potentially improve the ergonomics in pHRI tasks by decreasing the chances of muscular fatigue in the users.

The focus of Experiment 2 was to test the potential use of the proposed system in real manufacturing environments based on the subjective opinions of individual workers. For this reason, Experiment 2 was conducted with 3 industrial workers who had extensive experience in performing small bath tasks such as drilling. The results of Experiment 2 showed that the workers found the system easy to use and helpful in carrying out the drilling task on curved surfaces. Also, the results showed that they enjoyed the system, felt safe and successful, and were willing to use it again.

For future work, a depth camera can be attached to the robot for extracting surface information locally only rather than for the whole surface, as was done in our study. Moreover, the workpiece could be tracked in real-time, thereby eliminating the need to rigidly fix it in place. Consequently, a significant amount of time can be saved when working on a part with unknown geometry held in human hand.

Appendix: Angle Monitoring Module

In subsection 2.7, it was mentioned that we show the live errors of polar and azimuth angle, i.e., ε_{φ} and ε_{θ} through the AR interface (Figure 11: b) as error bars that the user tries to minimize their length by moving the drill and approach to the desired angle of attack. Computing these two angles in real time on the robot's computer side has some mathematical work which might be fruitful for an astute reader. In this appendix we will delve into details of the software module that takes care of computing these angles in real-time.

Let's start by remembering that in each execution of our control loop (Figure 3) in 500 Hz, after we exert the interaction wrench vector $\mathbf{F}_{int} = [F_{int}^{x}, F_{int}^{y}, F_{int}^{z}, \tau_{int}^{x}, \tau_{int}^{y}, \tau_{int}^{z}]$ to the drill and solve for our admittance control equation we will eventually have

$$ {}^{KBF}_{Tool}T = \begin{bmatrix} {}^{KBF}_{Tool}R_{3\times3} & {}^{KBF}_{Tool}P_{3\times1} \\ 0_{1\times3} & 1 \end{bmatrix} \tag{A.1} $$

which is used for computing ${}^{KBF}_{7}T$ by implementing the calculations presented in subsection 2.2. The computed ${}^{KBF}_{7}T$ in each time step, is being fed to inverse kinematic functions to generate the desired join configuration $q_{7\times1_{des@each\,time\,step}}$. This 7×1 vector is basically what we give the robot's internal controller to go to. In a nutshell, in each time step of our control loop we have ${}^{KBF}_{Tool}T$ and using this homogenous transformation matrix, and the known desired polar and azimuth angles for the drilling target of our interest we should compute the errors ε_{φ} and ε_{θ}. According to the principle mentioned in equation (2.9), we can expand the way we write ${}^{KBF}_{Tool}R_{3\times3}$ as follows.

$$ {}^{KBF}_{Tool}R = \begin{bmatrix} {}^{KBF}X_{Tool_{3\times1}} & {}^{KBF}Y_{Tool_{3\times1}} & {}^{KBF}Z_{Tool_{3\times1}} \end{bmatrix} \tag{A.2} $$

Each column of this 3×3 matrix is the components of the axis unit vectors of {T} coordinate frame expressed in {KBF}. So, let's unravel these vectors as follows.

$$ {}^{KBF}_{Tool}R = \begin{bmatrix} {}^{KBF}X_{Tool_x} & {}^{KBF}Y_{Tool_x} & {}^{KBF}Z_{Tool_x} \\ {}^{KBF}X_{Tool_y} & {}^{KBF}Y_{Tool_y} & {}^{KBF}Z_{Tool_y} \\ {}^{KBF}X_{Tool_z} & {}^{KBF}Y_{Tool_z} & {}^{KBF}Z_{Tool_z} \end{bmatrix} \quad (A.3) $$

The vectors, ${}^{KBF}X_{Tool}$, ${}^{KBF}Y_{Tool}$, and ${}^{KBF}Z_{Tool}$ are expressed in {KBF} frame but the polar (φ) and azimuth (θ) angles we are looking for are measured with respect to {LNF}. To get these two values into the game we will re-write the vectors shaping each column of ${}^{KBF}_{Tool}R$ as follows.

$$ {}^{KBF}_{Tool}R = \begin{bmatrix} {}^{KBF}_{LNF}R\,{}^{LNF}X_{Tool} & {}^{KBF}_{LNF}R\,{}^{LNF}Y_{Tool} & {}^{KBF}_{LNF}R\,{}^{LNF}Z_{Tool} \end{bmatrix} \quad (A.3) $$

We have in mind that from the solver of our controller and the forward kinematics of our robot in each time step we have matrix ${}^{KBF}_{Tool}R$ granted.

$$ {}^{KBF}_{Tool}R = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (A.4) $$

Also, from equation (2.10) we have ${}^{KBF}_{LNF}R$ known for a designated drilling point of interest.

$$ {}^{KBF}_{LNF}R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (A.5) $$

Taking into account equations (A.5), (2.23), (2.24), and (2.25) we will have the matrix equation set which could be solved for the current φ and θ as an output in each timestep of the pHRI system control loop.

$$
{}_{LNF}^{KBF}R = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}_{3\times3}
= [[r_{11}\cos(\varphi) + r_{12}\cos(\theta)\sin(\varphi) + r_{13}\sin(\theta)\sin(\varphi), r_{21}\cos(\varphi) + r_{22}\cos(\theta)\sin(\varphi) + r_{23}\sin(\theta)\sin(\varphi), r_{31}\cos(\varphi) + r_{32}\cos(\theta)\sin(\varphi) + r_{33}\sin(\theta)\sin(\varphi)]^{T}, [r_{12}\sin(\theta) - r_{13}\cos(\theta), r_{22}\sin(\theta) - r_{23}\cos(\theta), r_{32}\sin(\theta) - r_{33}\cos(\theta)]^{T}, [-r_{11}\sin(\varphi) + r_{12}\cos(\theta)\cos(\varphi) + r_{13}\sin(\theta)\cos(\varphi), -r_{21}\sin(\varphi) + r_{22}\cos(\theta)\cos(\varphi) + r_{23}\sin(\theta)\cos(\varphi), -r_{31}\sin(\varphi) + r_{32}\cos(\theta)\cos(\varphi) + r_{33}\sin(\theta)\cos(\varphi)]^{T}]_{3\times3} \quad \text{(A.6)}
$$

In our experiments where we extracted the local surface normal vectors (i.e., $\hat{X}_{LNF}$, $\hat{Y}_{LNF}$, and $\hat{Z}_{LNF}$ – see Figure 6) from the point cloud that we generated using the Microsoft Kinect RGBD camera, while we were defining the {LNF} we selected the $\hat{Z}_{LNF}$ wisely to be able to solve equation (A.6) more conveniently. Let's represent $\hat{X}_{LNF}$ as $\vec{n}$.

$$
\vec{n} = [n_1 \quad n_2 \quad n_3] \quad \text{(A.7)}
$$

Now we can define the $\hat{Z}_{LNF}$ wisely enough that it happens to be perpendicular to $\vec{n}$ and have a zero X component as follows.

$$
\hat{Z}_{LNF} = [0 \quad 1 \quad -\frac{n_2}{n_3}] \quad \text{(A.8)}
$$

The last axis $\hat{Y}_{LNF}$ could be easily defined by the cross product if two others complying with the write and rule. Now again considering the principle mentioned in equation (2.9), together with the equation (A.5) we can conclude that component r_{13} of ${}_{LNF}^{KBF}R$ is identical to zero ($r_{13} = 0$). This smart decision will reduce the complexity of equation (A.6) and let us solve for polar and azimuth angles more conveniently. By equating the second column of both sided of equation (A.6) and considering $r_{13} = 0$, one can write the following equations.

$$
r_{12}\sin(\theta) = a_{12} \quad \text{(A.9)}
$$

$$
r_{22}\sin(\theta) - r_{23}\cos(\theta) = a_{22} \quad \text{(A.10)}
$$

$$r_{32}\sin(\theta) - r_{33}\cos(\theta) = a_{32} \tag{A.11}$$

From (A.9) one can easily compute the value of $\sin(\theta)$ as $\sin(\theta) = a_{12}/r_{12}$, having the value of $\sin(\theta)$, considering (A.10), $\cos(\theta)$ could be computed as $\cos(\theta) = (a_{22} - r_{22}a_{12}/r_{12})/-r_{23}$. Now that we have computed both $\sin(\theta)$ and $\cos(\theta)$ the exact value of θ in that time step could be calculated as follows:

$$\theta = atan2\left(\frac{a_{12}}{r_{12}}, -\frac{a_{22} - \frac{r_{22}}{a_{12}}}{r_{23}}\right) \tag{A.12}$$

For computing the polar angle, we should consider the third columns of matrices io equation (A.6) and equate them from both sides. We only consider the first two rows.

$$-r_{11}\sin(\varphi) + r_{12}\cos(\theta)\cos(\varphi) + r_{13}\sin(\theta)\cos(\varphi) = a_{13} \tag{A.13}$$

$$-r_{21}\sin(\varphi) + r_{22}\cos(\theta)\cos(\varphi) + r_{23}\sin(\theta)\cos(\varphi) = a_{22} \tag{A.14}$$

Considering that we have already computed the values of $\sin(\theta)$ and $\cos(\theta)$ as well as a few mathematical manipulations we can derive the value of polar angle as follows.

$$\sin(\varphi) = -\frac{a_{23} - \frac{(r_{22}\cos(\theta) + r_{23}\sin(\theta))a_{13}}{r_{12}\cos(\theta)}}{r_{21} - \frac{\left(\frac{(r_{22}\cos(\theta) + r_{23}\sin(\theta))a_{13}}{r_{12}\cos(\theta)}\right)r_{11}}{r_{12}\cos(\theta)}} \tag{A.15}$$

$$\varphi = atan2\left(\sin(\varphi), \frac{a_{13} + r_{11}\sin(\varphi)}{r_{12}\cos(\theta)}\right) \tag{A.16}$$

Hence, using equations (A.12) and (A.16) we can compute the live polar and azimuth angles as the user is moving the robot and tries to adjust itself with the desired angles of attack.

BIBLIOGRAPHY

[1] V. Duchaine and C. M. Gosselin, "General model of human-robot cooperation using a novel velocity based variable impedance control," in *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, World Haptics 2007*, 2007.

[2] V. Duchaine and C. Gosselin, "Safe, stable and intuitive control for physical human-robot interaction," in *2009 IEEE International Conference on Robotics and Automation*, 2009.

[3] C. Heyer, "Human-robot interaction and future industrial robotics applications," in *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010*, 2010.

[4] M. Ben-Ari and F. Mondada, Elements of robotics (robots and their applications), 2017, p. 1–308.

[5] A. Ajoudani, A. Zanchettin, S. Ivaldi, A. Albu-schäffer and K. Kosuge, "Progress and prospects of the human-robot collaboration," *Autonomous Robots,* vol. 42, p. 957–975, 2018.

[6] M. Selvaggio, M. Cognetti, S. Nikolaidis, S. Ivaldi and B. Siciliano, "Autonomy in physical human-robot interaction: A brief survey," *IEEE Robotics and Automation Letters,* vol. 6, p. 7989–7996, October 2021.

[7] E. Matheson, R. Minto, E. G. G. Zampieri, M. Faccio and G. Rosati, "Human–Robot Collaboration in Manufacturing Applications: A Review," *Robotics,* vol. 8, 2019.

[8] Y. Hu, M. Benallegue, G. Venture and E. Yoshida, "Interact With Me: An Exploratory Study on Interaction Factors for Active Physical Human-Robot Interaction," *IEEE Robotics and Automation Letters,* vol. 5, pp. 6764-6771, 2020.

[9] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier and P. Fraisse, "Collaborative manufacturing with physical human-robot interaction," *Robotics and Computer-Integrated Manufacturing,* vol. 40, p. 1–13, 2016.

[10] M. Zhang, C. Li, Y. Shang and Z. Liu, "Cycle Time and Human Fatigue Minimization for Human-Robot Collaborative Assembly Cell," *IEEE Robotics and Automation Letters,* pp. 1-1, 2022.

[11] E. Lamon, F. Fusaro, P. Balatti, W. Kim and A. Ajoudani, "A Visuo-Haptic Guidance Interface for Mobile Collaborative Robotic Assistant (MOCA)," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[12] K. R. Guerin, S. D. Riedel, J. Bohren and G. D. Hager, "Adjutant: A framework for flexible human-machine collaborative systems," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014.

[13] K. R. Guerin, C. Lea, C. Paxton and G. D. Hager, "A framework for end-user instruction of a robot assistant for manufacturing," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[14] H. Ochoa and R. Cortesão, "Impedance control architecture for robotic-assisted micro-drilling tasks," *Journal of Manufacturing Processes,* vol. 67, pp. 356-363, 2021.

[15] S. Kana, S. Lakshminarayanan, D. M. Mohan and D. Campolo, "Impedance controlled human–robot collaborative tooling for edge chamfering and polishing applications," *Robotics and Computer-Integrated Manufacturing,* vol. 72, p. 102199, 2021.

[16] R. Perez-Ubeda, S. C. Gutierrez, R. Zotovic and J. Lluch-Cerezo, "Study of the application of a collaborative robot for machining tasks," *Procedia Manufacturing,* vol. 41, pp. 867-874, 2019.

[17] A. Frommknecht, J. Kuehnle, I. Effenberger and S. Pidan, "Multi-sensor measurement system for robotic drilling," *Robotics and Computer-Integrated Manufacturing,* vol. 47, p. 4–10, October 2017.

[18] Y. Bu, W. Liao, W. Tian, J. Zhang and L. Zhang, "Stiffness analysis and optimization in robotic drilling application," *Precision Engineering,* vol. 49, p. 388–400, July 2017.

[19] Y. Aydin, D. Sirintuna and C. Basdogan, "Towards collaborative drilling with a cobot using admittance controller:," *Transactions of the Institute of Measurement and Control,* vol. 43, p. 1760–1773, July 2020.

[20] D. Sirintuna, Y. Aydin, O. Caldiran, O. Tokatli, V. Patoglu and C. Basdogan, "A variable-fractional order admittance controller for pHRI," in *IEEE International Conference on Robotics and Automation*, 2020.

[21] Y. Aydin, O. Tokatli, V. Patoglu and C. Basdogan, "Stable physical human-robot interaction using fractional order admittance control," *IEEE Transactions on Haptics,* vol. 11, p. 464–475, 2018.

[22] Y. Aydin, O. Tokatli, V. Patoglu and C. Basdogan, "A computational multicriteria optimization approach to controller design for physical human-robot interaction," *IEEE Transactions on Robotics,* vol. 36, p. 1791–1804, December 2020.

[23] B. Guler, P. P. Niaz, A. Madani, Y. Aydin and C. Basdogan, "An adaptive admittance controller for collaborative drilling with a robot based on subtask classification via deep learning," *Mechatronics,* vol. 86, p. 102851, 2022.

[24] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *IEEE Intrernational Conference on Robotics and Automation*, 2011.

[25] D. Sirintuna, I. Ozdamar, Y. Aydin and C. Basdogan, "Detecting human motion intention during pHRI using artificial neural networks trained by EMG signals," in *29th IEEE International Conference on Robot and Human Interactive Communication, RO-MAN 2020*, 2020.

[26] C. Basdogan, C. H. Ho, M. A. Srinivasan and M. Slater, "An Experimental Study on the Role of Touch in Shared Virtual Environments," *ACM Transactions on Computer-Human Interaction,* vol. 7, p. 443–460, December 2000.

[27] A. Kucukyilmaz, T. M. Sezgin and C. Basdogan, "Intention recognition for dynamic role exchange in haptic collaboration," *IEEE Transactions on Haptics,* vol. 6, p. 58–68, 2013.