ASSISTIVE HRI INTERFACE WITH PERCEPTUAL FEEDBACK CONTROL: AN APPROACH TO CUSTOMIZING ASSISTANCE BASED ON USER DEXTERITY

BY

HAN UL YOON

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2014

Urbana, Illinois

Doctoral Committee:

Professor Seth A. Hutchinson, Chair
Professor Daniel M. Liberzon
Professor Ranxiao F. Wang
Associate Professor Timothy W. Bretl

ABSTRACT

In this dissertation, we propose a novel method for customizing an assistive interface on the basis of user dexterity for human-robot interaction (HRI). Customizing the assistive HRI interface, in practice, is a challenging problem due to the variety of user's task-performance and task-performing characteristics. For this reason, we develop a method to assist a user with strategies of a high-performer who has the most similar task-performing characteristics to the user. From the experiment evaluating user's task performance, we observed that our method indeed enhanced the user's performance in terms of task-completion time and the average velocity during the task.

The backbone of our approach is based on setting virtual fixture parameters to yield the assistive control and perceptual (haptic and visual) feedbacks which are customized for a specific user. First, we model a user as a cost function using the techniques from inverse optimal control (IOC). With the underlying assumption – human users are optimizing a cost function while performing a given task – we infer the unknown parameters of the cost function from observing the user demonstration. Next, we define three features that characterize the user's task-performing characteristics as the balances of the inferred parameter vectors, and classify the user based on the closest high-performer in feature space. Finally, we set the virtual fixture parameters according to the user's task-performance, class, and features to provide the user the customized guidance with the high-performer's strategies.

We carry out human subject experiments to evaluate the user performance in the presence of various assistance modes. The results from the experiments show that the customized virtual fixturing with haptic feedback outperforms the other types of "virtual fixturing and feedback" combinations, and depicts the enhancement of both high- and low-performing users' performance as well. Hence, we conclude that our approach is an effective way to improve the user task performance.

To my parents

To my Eun-Kyeong, Sahwoo, and Jeahoo

ACKNOWLEDGMENTS

"In all thy ways acknowledge him, and he shall direct thy paths."
- Proverbs 3:6 -

I sincerely give thanks to all faculty and staff at the University of Illinois who taught and provided me the world-class materials and support. Although there are numerous people to name, I would like to acknowledge Gerald Dejong, Steve Lavalle, Harrison Kim, Sean Meyn, Mark Spong, Dusan Stipanovic, Tamer Basar, Dan Block, Jana Lenz, and Laurie Fisher.

I would never allow myself to forget my academic brothers - Steven Cloder, Sourabh Bhattacharya, James Davidson, Sal Candido, Devin Bonnie, Miles Johnson, Jaesung Moon, Pilwon Hur, and Hyoungju Park. Filled with precious moments of my life during the course of work and discussion with all of them, I enjoy reminiscing those days whenever they come up to my mind.

My committee members, Seth Hutchinson, Daniel Liberzon, Frances Wang, and Tim Bretl, were priceless mentors who supported me not only by exemplifying enthusiasm and professionalism during my coursework but also by showing their perspectives of what a researcher should be throughout the later stages of my Ph.D. study.

I acknowledge and thank my adviser, Seth Hutchinson, for being a tracer when I shoot the problems and a sage counselor in both academic matters and personal matters during days as a Ph.D. student. Especially, I truly thank him for his endless belief in and patience with me.

My family is the pillar of my life. It has always been my last shelter to remind myself of "being a son," "being a husband," and "being a father" of Yoon. By all of you, my existence could be completely redefined.

Now, I open my eyes and understand that every single footprint of mine during my life in Urbana-Champaign was envisioned by the Lord. Thanks for reshaping me as not only the Ph.D. but the Permanent heavenly Disciple.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In this dissertation, we present a method for designing an assistive human-robot interaction (HRI) interface that provides a customized level of assistance with haptic and visual feedback based on the user dexterity with a given task. Before discussing our work in theoretical detail in later chapters, we address the general motivation for working on this research and the description of the chosen approach.

1.1 Why Develop an Assistive HRI Interface that Provides the Customized Assistance Based on User Dexterity?

Assistive interfaces are currently prevalent in various applications due to their unique characteristics of constraining/regulating user-driven control input, giving more freedom to the users' control [1]. An assistive interface (together with sensory feedback devices) is known to improve user task-performance, including telerobotic tasks [2], steering tasks [3, 4], a robot-assisted manipulation [5, 6], assistive medical devices and telesurgery [7–10], and so on. In the design of assistive interface, setting the proper level of assistance or constraint is a crucial factor that affects on the user's task performance [11, 12]. While excessive assistance for a skilled user would slow down task completion, a lack of assistance for a novice user may cause a task failure.

The various applications of assistive interfaces exist for a wide range of applications, however user's dexterity is rarely considered; the user is typically assumed to be skilled and familiar to the interface. Even for skilled users, e.g., skilled drivers or pilots, their task-performance can vary due to distraction by fatigue, boredom from tedious scenic views, or possible substance abuse. The user's task-performance, on the other hand, is determined

by a strategy (so called know-how or expertise) to complete a given task. Therefore, the development in this dissertation focuses on customizing the level of assistance based on the user's task-performance, as well as on guiding a low-performing user with a high-performing user's strategy. This can be a contribution in both design frameworks and approaches in the field of assistive interface.

Although the discussion of this dissertation is focused on specific target application in which a human user teleoperates a mobile robot by a haptic input device while monitoring with a visual display, the approaches and frameworks through the dissertation can be used for customizing/adjusting/tuning parameters of generic assistive system, especially for systems that need to be customized based on user dexterity and task-performing strategy.

1.2 Challenges in Assistive HRI Interface Design and Customization

Customized assistive HRI interface design is a challenging problem, in practice, due to the following issues.

- First of all, we should determine **a way of blending a human user's intention with machine assistance** under various circumstances in which the human user and the machine need to cooperate. Especially, we aim at an assistive interface that never takes control away from the human user. In other words, system control is not switched completely to a machine, but the machine provides guidance to the human user. Therefore, control blending should adjust or modify the balance between the user's intention and the machine assistance under various circumstances.

- Second, the developed assistive interface should be able to cope with **a range of user dexterity (task performance) and task-performing characteristics**. The assistive interface should provide the right amount level of assistance, as an excessive level of assistance for a high-performing user would slow down task completion. In contrast, we can expect that a lack of constraint for a low-performing user may lead to a

task failure. Consequently, the assistive interface should provide assistance which is adequate to not only support the human user, but also refrain from deteriorating the performance based on the user dexterity and task-performing characteristics.

- Third, **the definition and the quantification of the user's task-performing characteristics** should precede the customization of the level of assistance. Specifically, the quantification allows us to define a metric for classifying the users into groups wherein their task-performing strategies are akin to each other. Furthermore, we can investigate the relationship between the group of specific characteristics and task-performance, or the task-performance in between users in the same group.

- Lastly, it is normally **difficult to define an explicit relationship between the user's task-performing characteristics and task performance**, e.g., a relationship defined by one-to-one continuous mapping. This is due to the existence of multiple task-performing strategies which might result in approximately identical task performance. Thus, we need to find a way to help the users whose task performance is identical, but whose task-performing characteristics are different.

1.3 Our Approach to Solve the Assistance Customizing Problem

To overcome the challenges addressed in Section 1.2, our approach employs a virtual fixture of which parameters and a spine are customized based on a user dexterity and task-performing characteristics, respectively. Specifically, we define the width, the minimum of active constraint, and the corresponding intensity of the virtual fixture as functions of the user dexterity. The spine of the virtual fixture is chosen based on a high-performer to whom the user's task-performing characteristics are closest.[1]

[1]The high-performer means that a person who has expertise, in both a manual control skill and strategic knowledge to complete a given task.

To achieve the parameter customization of the virtual fixture and the choice of its spine, our approach quantifies three features of the user's task-performing characteristics using techniques from inverse optimal control. This quantification allows us to associate a user to a sample point in the defined feature space, and also tells us the difference between the user's task-performing characteristics.

Our approach provides two kinds of perceptual feedback, haptic and visual feedback, which are customized as well. A user feels a guiding force when using the haptic input device, and a visual display adjusts the size of working field of view. The purpose of providing the customized perceptual feedback is to release the user from the pressure of applying the finer control, and to help him/her to be aware of the environmental condition as well as the quality of control.

1.4 Summary of Contributions

1. We develop an assistive HRI interface to provide customized assistance and perceptual feedback based on user dexterity. This development advances the state of the art in the problem of developing assistive interfaces by considering assistance customization.

2. We present an approach for modeling user task-performing characteristics using techniques from inverse optimal control. The outcome from this approach allows to represent a user as a sample point in the defined feature space.

3. We propose a method to classify a user based on a high-performer to whom the user has the most similar task-performing strategies. This classification result is utilized to customize the level of assistance that guides the user by the high-performer's strategy.

4. We experimentally evaluate the human subject's performance with our HRI interface to substantiate a set of hypotheses about the effectiveness of our approach. These evaluations and hypothesis substantiations also advance the state of the art in perspective of psycho-motor findings.

1.5 Dissertation Outline

Chapter 2 provides the background related to this research project. We present established approaches, and introduce virtual fixtures, assistive/cooperative control, inverse optimal control, and bilateral teleoperation as key concepts and utilities for our design approach.

In Chapter 3, we present our approach to designing an assistive human-robot interaction interface. We adopt the specific application in which a human user controls a mobile robot by a haptic input device while monitoring the environment with a visual display. This chapter begins by describing the developed assistive HRI interface platform in Section 3.1. Specially, we present the concepts of assistive control and perceptual feedback control in Section 3.1.2. Then, we introduce our system modeling approach from entire system model to subsystem model using a top-down approach in Section 3.2. Section 3.3 explains the role of assistive control. We also discuss a guiding force that is blended with a control input to assist a user's control, and utilized as a resource to provide haptic feedback. We discuss the stability condition for our HRI interface in Section 3.4.

Chapter 4 begins by introducing inverse optimal control (IOC) and its application in Section 4.1. We then formally describe the IOC problem statement, step-by-step derivation, and applied numerical method in Section 4.2. Section 4.3 also describes an experimental setup and procedure to obtain user-demonstrated data. These descriptions work as a prerequisite to define features for modeling user's task-performing characteristics in Section 4.4. Finally, Section 4.5 discusses a classification method that classifies the users into the classes wherein the user's task-performing characteristics are akin to each other.

Chapter 5 presents a method for customizing a virtual fixture and haptic feedback. This chapter mainly focuses on describing how we can utilize the classification result from Section 4.5 to customize the level of assistance for the users of various dexterities. We begin in Section 5.1 by introducing the meaning of soft assistance, firm assistance, and the amount of attenuation. Then, we show that the attenuation term in the original control force can be regarded as a guiding force in Section 5.2. As the magnitude and direction of the guiding force are determined by a virtual fixture setting, we discuss our approach to customizing virtual fixture parameters based on the user's

dexterity in Section 5.3.

We discuss an approach to customizing visual feedback in Chapter 6. We begin by introducing terminologies related to psychology-engineering research: steering law, the index of difficulty, and the index of performance in Section 6.1. In Section 6.2, we discuss the effect of increasing zoom level on the user's task-completion time and screen velocity in WFoV. Based on the examination and existing psycho-motor findings, our assistive HRI system also modifies visual feedback by adjusting the zoom level as defined in Section 6.3.1. Accordingly, Section 6.3.2 presents an approach to customizing visual feedback based on the user's performance.

In Chapter 7, we evaluate user task performance with the developed assistive HRI interface. We first introduce the experimental apparatus and protocol, and formulate hypotheses in Section 7.1. Then, in Section 7.2 we present the resulting evaluation data in terms of task-completion time, average velocity during the task, and enhancement in both task-completion time and average velocity according to various assistance modes. We further investigate the evaluation data to find a relationship between the assistance modes and the performance enhancement of the user with different skill levels in Section 7.3. Specifically, we test three hypothetical questions, and compare our assistance customization approach to the other types to validate the effectiveness in Section 7.3.1.

Finally, Chapter 8 concludes this dissertation by providing a discussion of the outlined work, and suggesting future directions of this research.

CHAPTER 2

BACKGROUND RELATED TO OUR DESIGN APPROACH

In this chapter, we present and discuss the background related to this research project. Organizationally, the purpose of this chapter is to present established approaches, and introduce key concepts and utilities for our design approach. Specifically, a virtual fixture, assistive/cooperative control, inverse optimal control, and bilateral teleoperation will be discussed throughout the chapter.

The first half of this chapter is related to approaches to implementing and providing assistance to a human user. In Section 2.1, we introduce a definition and historical remarks of a virtual fixture. We then present a virtual fixture and an implementation approach, and further discuss the existing applications in telemanipulation tasks. In Section 2.2, we introduce basic concepts for assistive/cooperative system and review well-known established approaches to implementing assistive control schemes by presenting existing-applications in various fields. We also discuss the effects on a user's task-performance of providing feedback as assistance in the later part of the section.

The second half of the chapter introduces tools which will be used to model a user's task-performing characteristics, and to guarantee the developed system's stability. We discuss an inverse optimal control problem and formulation with the specific example of a quadratic regulator in Section 2.3. Applied numerical methods that will be used to solve the inverse optimal control problem in this dissertation are described as well. In Section 2.4, we present various representations of bilateral teleoperation system architectures from specific to general cases. Then, we discuss the special case of bilateral teleoperation that can be regarded as a typical feedback controller with a desired reference input under specific conditions. We also introduce various examples and techniques to analyze the stability of bilateral teleoperation system.

Lastly, we conclude the chapter with further resources.

2.1 Virtual Fixture as an Approach to Implementing Assistive Control

In this section, we provide a definition and historical remarks, describe a guidance Virtual Fixture, and discuss the effect of virtual fixturing in telemanipulation tasks on a user's performance.

2.1.1 Definition and Historical Remarks

A virtual fixture (also known as an active constraint) is a software-generated user-control constraining strategy in order to improve the safety, accuracy, and speed of machine assisted manipulation tasks [5,13]. Specifically, it helps a human operator by limiting movement in restricted regions and/or influencing movement along desired paths [13]. These restrictions consequently allow the human operator to perform physical interactions with higher confidence and speed, yet retain direct control of the activity [14]. For two decades, the virtual fixture has been a prevalent strategy as a way of implementing assistive control in human-machine or human-robot collaborative tasks due to the unique characteristics of constraining/regulating user-driven control input, rendering more freedom to the user's control.

The term "virtual fixture" originally connoted mechanical characteristics that constrain tool motion, but now the term refers to the augmentation of sensory feedback from a slave to a master in a teleoperation system in order to reduce a human operator's mental workload [1,15]. After the initial development of concept and terminology, the progression of the virtual fixture was ignited from the field of robotic surgery wherein the perspectives of researchers to ensure system safety had been somewhat disparate [1,16]. Due to the requirement that the system never control surgical tools solely (i.e., never take control away from a human surgeon), the virtual fixture could be widely accepted as a suitable control scheme.

Since the concept of virtual fixture was born, various techniques and applications in many fields have been developed as follows: virtual fixtures in telemanipulation [13, 17, 18], defining flexible virtual fixtures by applying

machine learning techniques [19], virtual fixtures generated by anatomy in medical robots [20, 21], virtual fixtures for multi-robot collaborative teleoperation [22], real-time haptic assistance adaptation [11], and virtual fixtures for remotely touching and protecting a beating heart [23].

2.1.2 A Guidance Virtual Fixture and an Implementation Approach

According to constraining strategies, guidance virtual fixtures (GVF) and forbidden region virtual fixtures (FRVF) are the two main types [1]. As their name implies, GVFs serve as supportive trajectory controllers that assist a human operator in guiding a controlled object along desired paths, while FRVFs impose region constraints that help the controlled object avoid certain regions [5]. Fig. 2.1 illustrates the roles of GVFs and FRVFs respectively.[1] From now on, we use the term "virtual fixture" to refer to GVF unless its type is denoted explicitly.

Figure 2.1: (a) Guidance virtual fixture. (b) Forbidden-region virtual fixture [5].

Among the various proposed ways to implement the virtual fixture, we adopt an approach that is similar to one proposed in [14]. First, from Fig. 2.2, let $\boldsymbol{x}$ and $\mathbf{s}$ be a position vector of controlled object's $[x_1, x_2]^T$ and the closest point on a *spine* (desired path or reference trajectory) from $\boldsymbol{x}$. We define $\mathbf{e}$ to be

$$\mathbf{e}(\boldsymbol{x}(t)) = \mathbf{s} - \boldsymbol{x}(t) \tag{2.1}$$

Next, we define *preferred* direction and *non-preferred* direction. Given $\boldsymbol{x}$, we can determine the tangent vector $\mathbf{h}$ at $\mathbf{s}$. Let $\boldsymbol{f}$ be a force applied to the controlled object. Following [14], we define a vector toward the preferred

[1] For more specific illustrations of active constraint by virtual fixturing, see Fig. 6 in [1].

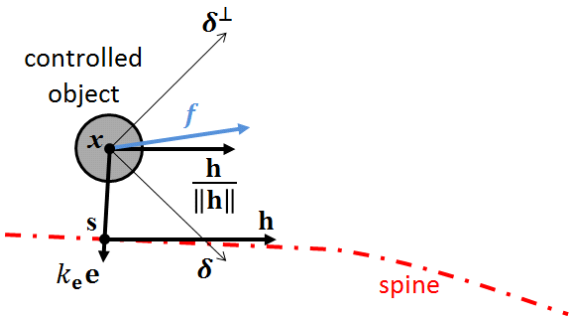


Figure 2.2: Preferred direction $\boldsymbol{\delta}$ and non-preferred direction $\boldsymbol{\delta}^{\perp}$.

direction, denoted by $\boldsymbol{\delta}$, as

$$\boldsymbol{\delta}(\boldsymbol{x}(t)) = \text{signum}\left(\boldsymbol{f}^T\mathbf{h}(\boldsymbol{x}(t))\right)\frac{\mathbf{h}(\boldsymbol{x}(t))}{\|\mathbf{h}(\boldsymbol{x}(t))\|} + k_{\mathbf{e}}\mathbf{e}(\boldsymbol{x}(t)) \tag{2.2}$$

where $k_{\mathbf{e}}$ is a positive scaling factor. Accordingly, we refer to a direction that is orthogonal to the preferred direction as a non-preferred direction, and denote a vector toward the non-preferred direction by $\boldsymbol{\delta}^{\perp}$. Fig. 2.2 shows the defined $\boldsymbol{\delta}$, and $\boldsymbol{\delta}^{\perp}$.

Now, we define *attenuating admittance* [14], denoted by $c_{\boldsymbol{\delta}^{\perp}}$, as a monotonically non-increasing function with respect to $\|\mathbf{e}\|$

$$c_{\boldsymbol{\delta}^{\perp}}(\|\mathbf{e}\|) = \begin{cases} \underline{c}_{\boldsymbol{\delta}^{\perp}}, & \text{if } \|\mathbf{e}\| > \frac{d}{2} \\ \underline{c}_{\boldsymbol{\delta}^{\perp}} + \left[\frac{d/2-\|\mathbf{e}\|}{\nu}\right]^n (1-\underline{c}_{\boldsymbol{\delta}^{\perp}}), & \text{if } \frac{d}{2}-\nu < \|\mathbf{e}\| \leq \frac{d}{2} \\ 1, & \text{if } \|\mathbf{e}\| \leq \frac{d}{2}-\nu \end{cases} \tag{2.3}$$

where $\frac{d}{2}$ is the half width of a virtual fixture, and ν determines the width of an interval (correction zone) in which $c_{\boldsymbol{\delta}^{\perp}}$ changes its value from 1 to the lower limit $\underline{c}_{\boldsymbol{\delta}^{\perp}}$. From (2.3), we know the following:

- If the controlled object position $\boldsymbol{x}(t)$ lies outside the virtual fixture, then $c_{\boldsymbol{\delta}^{\perp}}(\|\mathbf{e}\|)$ is set to $\underline{c}_{\boldsymbol{\delta}^{\perp}}$.
- If the position lies in a "correction zone", then $c_{\boldsymbol{\delta}^{\perp}}(\|\mathbf{e}\|)$ is adjusted based on the error $\|\mathbf{e}\|$.
- If the is near to the spine, then $c_{\boldsymbol{\delta}^{\perp}}(\|\mathbf{e}\|) = 1$.

The attenuating admittance $c_{\boldsymbol{\delta}^{\perp}}$ will play an important role to yield an *assisted control force* by attenuating the non-preferred directional component

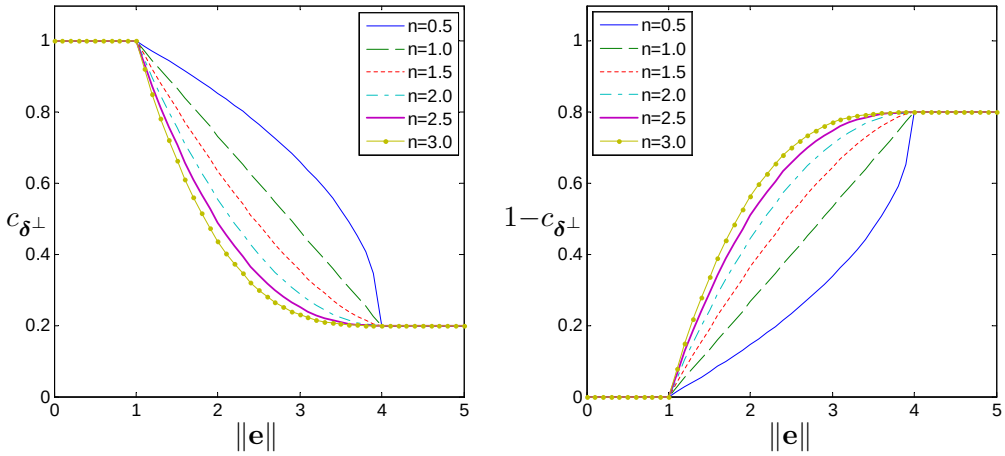


Figure 2.3: An example plot of $c_{\boldsymbol{\delta}^\perp}$(left) and $1 - c_{\boldsymbol{\delta}^\perp}$(right). We set the parameters as $\|\mathbf{e}\| : [0, 5]$, $\frac{d}{2} = 4$, $\nu = 3$, and $\underline{c_{\boldsymbol{\delta}^\perp}} = 0.2$. In both graphs, the origin $(0, 0)$ represents $\mathbf{s}$ on the spine.

in the original control force. We will discuss how $c_{\boldsymbol{\delta}^\perp}$ contributes to yielding the assisted control force in detail in Section 5.2.

Fig. 2.3 shows example plots of $c_{\boldsymbol{\delta}^\perp}$ and $1 - c_{\boldsymbol{\delta}^\perp}$. Note that it is sometimes more insightful to take into account for "the amount of attenuation", $1-c_{\boldsymbol{\delta}^\perp}$, as shown in the right side. For instance, $1 - c_{\boldsymbol{\delta}^\perp} = 0$ means that a user's control input is fully preserved without any attenuation. An example of virtual fixturing represented in terms of $1 - c_{\boldsymbol{\delta}^\perp}$ is illustrated as gradation along the spine in Fig. 2.4.

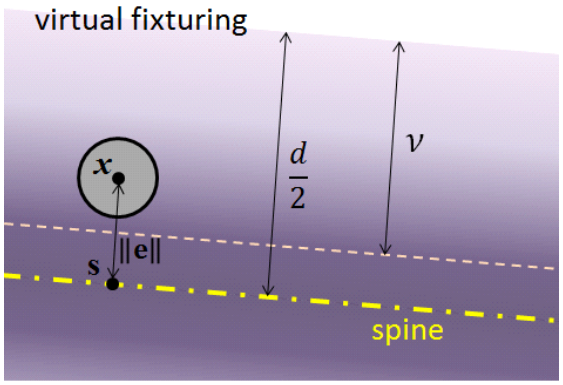


Figure 2.4: An example of virtual fixturing and parameters $\frac{d}{2}$ and ν. The amount of attenutaiton, $1 - c_{\boldsymbol{\delta}^\perp}$, is illustrated as gradation along the spine.

2.1.3 Virtual Fixturing in Telemanipulation and User's Task-Performance

As a virtual fixture is one way of implementing a human-robot collaborative system, it has been used for various application in telemanipulation. In [17], the experimental 1-DOF bilateral telemanipulation system in which master/slave consists of two haptic paddles together with a FRVF was presented. Afterward, this research led to a method of implementing a stable FRVF for bilateral telemanipulation [18], and bilateral telemanipulation scheme with a GVF [13], and robot-assisted manipulation [5].

The literature shows that the enhancement of a user's performance by employing a virtual fixture can be significant. Therefore, we describe some related work here. Related to [5, 13], the user's task-performance under steady-hand teleoperation versus steady-hand cooperative manipulation was analyzed in [6]. This research shows that cooperative manipulation outperformed the other approaches such as teleoperation and free-hand movement. In [24], it was found that a virtual fixture improves a user's task-performance both in terms of execution time and overall precision for complex tasks in virtual environments. Human performance improvement in assistive path following problems in the sense of completion time and number of collisions was presented in [25].

2.2 Assistive/Cooperative System

This section discusses the basic concept, and existing approaches to blending user's intention and machine assistance, and the effect of feedback for assistive/cooperative systems.

2.2.1 A Basic Concept

An assistive system has a controller that takes a user's control input, blends the control input with machine assistance, and finally generates the assisted control input. This blending of the user's intention and machine assistance is called by various names such as control arbitration [2], shared control [26,27], semi-autonomous control [28], cooperative manipulation [6], and guidance-

as-needed control [3]. "Shared control" normally refers to control techniques, in contrast to direct intuition from the name, that allow a machine system to take control away from a human user (which means the authority of controlling a system is switched completely to the a machine) when predetermined conditions occur [26,27,29]. "Assistive/Cooperative control" is used to mean a controller adjusts the weights of user's control and machine assisted control [2, 27]. In this dissertation, therefore, we refer to a system that blends the user's intention with machine assistance as assistive/cooperative system.

2.2.2 Existing Approaches to Blending User's Intention and Machine Assistance

In [2], an assistive telerobotic system platform was presented. A torso-type humanoid robot is teleoperated to perform a user-demonstrated action. With this platform, the humanoid robot predicts the user's intention, then blends the user's intention with its own prediction as a convex combination of the two. The level of assistance was also considered by examining the user's task-performance under providing over-assistance (providing unwanted assistance) and under-assistance (failing to provide needed assistance).

In the following approaches, the machine's prediction of either the user's intention/performance or the environmental situation was used as a semaphore for machine intervention to assist a user. The smart wheelchair system in [30] calculates the belief histogram of obstacle locations, and uses it to adjust the speed of the wheelchair. A method in [31, 32], called ability-based control, adjusts control-to-display (CD) gain based on the deviation of angles sampled during mouse movement. The CD gain is controlled as inversely proportional to angular deviation. Another approach to adjusting CD gain was proposed in [33]. The approach (called motor-model-based dynamic scaling) predicts the state of motor movement, e.g., ballistic movement, and scales CD gain dynamically. [3] also uses the deviation from the desired position and angle to adjust a guidance force applied to user and control gains.

In some medical applications, a human user's movement is physically constrained by a specific type of guidance force. An endoscope system using haptic guidance was presented in [7, 8]. Lane-keeping driver assistance systems are another example of systems constraining the user's intention [34,35].

Approaches to learning the user-demonstration for better prediction of user's intentions were proposed in [36, 37].

2.2.3 Effects on a User's Task-Performance with Providing Feedback as Assistance

According to psychological findings, it is known that providing haptic feedback typically enhances a user's task-performance [11, 15]. The results presented in [38, 39] show that a user's perceptual motor skill can be enhanced by training provided with haptic (as well as visual) feedback. In [40], the user's performance improvement by haptic feedback was discussed in a virtual car-driving task. More recently, the effects of haptic and visual aid on psychomotor task strategy have been addressed in [41], and the effect on learning spatio-temporal patterns and optimizing the human user's task performance and effort was discussed in [42] and [11], respectively.

The effect of visual feedback has been discussed in various literature: CD gains and user performance [43–45], the user's performance according to display size [46, 47], the dimension of display space, e.g., 2D or 3D display [48], and techniques for providing visual feedback in telemanipulation [49]. Generally, increasing a zoom level makes a given task easier in terms of the index of difficulty, and improves the user's performance in various tasks [6, 33].

2.3 Inverse Optimal Control to Model a User's Task-Performing Characteristics

This section presents the inverse optimal control (IOC) problem. We begin by introducing a linear quadratic regulator problem. We then discuss inverse optimal control, a specific case, and a numerical method applied to solve IOC in this dissertation.

2.3.1 An Example of Optimal Control: Linear Quadratic Regulator Problem

Let matrices R and Q be symmetric, and be nonnegative (positive semidefinite) and positive definite, respectively [50]. Suppose that a continuous time interval $[0, T]$ has been discretized into k-steps, where $k \in \{0, 1, \cdots, N-1\}$, with sampling time t_s. We define a cost function in terms of a state, $x_k \in \mathbb{R}^n$, and a control input, $u_k \in \mathbb{R}^m$, thus

$$L(x, u) = \sum_{k=0}^{N-1} \left(u_k^T R u_k + x_k^T Q x_k \right) \tag{2.4}$$

The linear quadratic regulator (LQR) problem is to choose the optimal control input, $u = (u_0, u_1, \cdots, u_{N-1})$, to minimize $L(x, u)$; namely

$$\min_u \quad L(x, u) = \sum_{k=0}^{N-1} \left(u_k^T R u_k + x_k^T Q x_k \right) \tag{2.5}$$

For a robotic system, the LQR problem is normally subject to the constraints of system equations, e.g., robot's kinematics [51, 52], which is represented by

$$\text{subject to} \quad x_{k+1} = x_k + t_s f(x_k, u_k) \tag{2.6}$$

2.3.2 Inverse Optimal Control Problem

Here, we introduce the general form of the IOC problem formulation. This generalized formulation in discrete time works as a prerequisite to discuss implementing a numerical algorithm to solve IOC problem in Section 2.3.4.

Consider the following (forward/direct) optimal control problem:

$$\begin{aligned} \min_u \quad & \sum_{k=0}^{N-1} c^T \phi(x_k, u_k) \\ \text{subject to} \quad & x_{k+1} = x_k + t_s f(x_k, u_k) \\ & x_0 = x_{\text{start}} \\ & x_N = x_{\text{goal}} \end{aligned} \tag{2.7}$$

where $x_k \in \mathbb{R}^n$ is a state, $u_k \in \mathbb{R}^m$ is a control input, $\phi : \mathbb{R}^{m+n} \to \mathbb{R}_+^\ell$

are known (pre-determined) basis functions, $f : \mathbb{R}^{m+n} \to \mathbb{R}^n$ is a kinematic equation, $c \in \mathbb{R}_+^\ell$ is a unknown parameter vector, and t_s is a sampling time.

Now, we define the inverse optimal control problem [53, 54]:

Given: data demonstrated by a user, i.e., state and input (x^*, u^*), which is assumed as locally optimal,

Infer: the user's cost function by inferring the unknown parameter vector c.

For instance, in the case of the LQR problem introduced in Section 2.3.1, the IOC amounts to inferring R and Q matrices given (x_k^*, u_k^*) for $k \in \{0, 1, \cdots, N-1\}$.

2.3.3 Specific Case IOC Problem Related to LQR with Diagonal R and Q Matrices

Suppose that we are given a specific form of (2.4)

$$\begin{aligned} L(x,u) &= \sum_{k=0}^{N-1} \left(u_k^T R u_k + x_k^T Q x_k \right) \\ &= \sum_{k=0}^{N-1} \left(c_1 u_{1,k}^2 + \cdots + c_m u_{m,k}^2 + c_{m+1} x_{1,k}^2 + \cdots + c_{m+n} x_{n,k}^2 \right) \end{aligned} \tag{2.8}$$

where

$$R = \begin{bmatrix} c_1 & 0 & \cdots & 0 \\ 0 & c_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_m \end{bmatrix} \quad \text{and} \quad Q = \begin{bmatrix} c_{m+1} & 0 & \cdots & 0 \\ 0 & c_{m+2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_{m+n} \end{bmatrix} \tag{2.9}$$

are diagonal matrices. The IOC problem related to (2.8) can be represented as a specific case of (2.7), thus

$$\begin{aligned} \min_u \quad & \sum_{k=0}^{N-1} c^T \phi(x_k, u_k) \\ \text{subject to} \quad & x_{k+1} = x_k + t_s f(x_k, u_k) \\ & x_0 = x_{\text{start}} \\ & x_N = x_{\text{goal}} \end{aligned} \tag{2.10}$$

where $\phi \in \{u_1^2, \cdots, u_m^2, x_{m+1}^2, \cdots, x_{m+n}^2\} : \mathbb{R}^{m+n} \to \mathbb{R}_+^{m+n}$ and $c \in \mathbb{R}_+^{m+n}$. The IOC problem statement is the same as in Section 2.3.2.

2.3.4 Applied Numerical Method to Solve IOC in the Dissertation

There exist well-known numerical approaches to solving the inverse optimal control problem: inverse reinforcement learning [54], IOC with linearly solvable Markov decision processes [55], continuous IOC [56], maximum margin planning [57]. In this dissertation, we apply the method developed in [58–60].

Start with recalling an equality constraint for $k \in \{0, 1, \cdots, N-1\}$

$$x_{k+1} = x_k + t_s f(x_k, u_k)$$

We define discrete time Hamiltonian H_k as [51, 61]

$$H_k(x_k, u_k, \lambda_k) = c^T \phi(x_k, u_k) + \lambda_k^T f(x_k, u_k) \tag{2.11}$$

where $\lambda_k \in \mathbb{R}^n$ is a costate vector at time step k. By applying the Pontryagin's Maximum Principle, we obtain the costate (propagation) equation [50, 62]

$$\frac{\partial H_k}{\partial x_k} = c^T \frac{\partial \phi(x_k, u_k)}{\partial x_k} + \lambda_k^T \frac{\partial f(x_k, u_k)}{\partial x_k} = -\frac{\lambda_{k+1} - \lambda_k}{t_s} \tag{2.12}$$

Rearranging (2.12) yields

$$c^T \frac{\partial \phi(x_k, u_k)}{\partial x_k}\ t_s + \lambda_{k+1}^T I + \lambda_k^T \left[-I + \frac{\partial f(x_k, u_k)}{\partial x_k}\ t_s \right] = \mathbf{0}^T \tag{2.13}$$

where $I \in \mathbb{R}^{n \times n}$ is an identity matrix. We also have

$$\frac{\partial H_k}{\partial u_k} = c^T \frac{\partial \phi(x_k, u_k)}{\partial u_k} + \lambda_k^T \frac{\partial f(x_k, u_k)}{\partial u_k} = \mathbf{0}^T \tag{2.14}$$

as a necessary condition for optimality [62, 63].

Now, we define a vector z_k as

$$z_k = \begin{bmatrix} c^T & \lambda_{k+1}^T & \lambda_k^T \end{bmatrix}^T \tag{2.15}$$

then two equations (2.13) and (2.14) can be combined and rewritten in the form, which gives us a system of equation, r_k, thus

$$\begin{aligned} r_k &= \begin{bmatrix} \left[\frac{\partial \phi(x_k,u_k)}{\partial x_k}\ t_s\right]^T & I & \left[-I + \frac{\partial f(x_k,u_k)}{\partial x_k}\ t_s\right]^T \\ \\ \left[\frac{\partial \phi(x_k,u_k)}{\partial u_k}\right]^T & O & \left[\frac{\partial f(x_k,u_k)}{\partial u_k}\right]^T \end{bmatrix} z_k \\ &= A_k z_k \end{aligned} \tag{2.16}$$

where $O \in \mathbb{R}^{m \times n}$ is a matrix of which all element are zero.

Finally, the given discrete time IOC problem becomes identical to solve the following least square problem:

$$\min_{c,\lambda} \sum_{k=0}^{N-1} \|r_k^*\|^2 \tag{2.17}$$

equivalently

$$\min_{c,\lambda} \sum_{k=0}^{N-1} \|A_k^* z_k\|^2 \tag{2.18}$$

where A_k^* represents the matrix A_k being evaluated at (x_k^*,u_k^*). We briefly present the method to solve (2.18) in Algorithm A.1.

2.4 Bilateral Teleoperation

2.4.1 A Brief Historical Review

For the past 50 years, bilateral teleoperation has brought forth results in both theory and application. In the late 1980s, a design framework and stability analysis based on the two-port network model was introduced [64, 65]. During that period, the presence of time delay had been the major impediment blocking the further movement of bilateral teleoperation because it induced instability [66]. The idea proposed in [67, 68] broke through the impediment by applying the notion of scattering operator (well-known in transmission theory) to bilateral teleoperation [66]. Since this breakthrough, bilateral teleoperation has evolved in several ways: master-slave teleoperation with various manipulator types and stability [67–70], teleoperation with time

domain passivity [71–74], supervisory control and human-machine interaction in telemanipulation [26, 75] and bilateral teleoperation of a wheeled mobile robot [76–80]. Recently, there has been research considering the influence of human operators and their perceptual and motor capabilities in order to involve the human operator as a part of teleoperation system [9, 81].

2.4.2 Representations for Bilateral Teleoperation System Architecture

For a bilateral teleoperation system architecture, there exist various models such as position exhange (position-position), position forward/force feedback (position-force), and position exchange/force feedback [17, 64, 82]. In this section, we mainly focus on discussing the position exchange model shown in Fig. 2.5, and presenting three representations: hybrid matrix, equivalent circuit, and general bilateral teleoperation system architecture. Table 2.1 depicts variables and parameters that will be used in this section.

An example system: 2-port position exchange bilateral control system

Fig. 2.5 depicts an example 2-port position exchange bilateral control system. For master/slave sides, the system equations are given by

$$F_{\rm h}(s) - F_{\rm am}(s) = V_{\rm h}(s) Z_{\rm m}(s) \tag{2.19}$$

$$F_{\rm as}(s) - F_{\rm e}(s) = V_{\rm e}(s) Z_{\rm s}(s) \tag{2.20}$$

where two interactive control forces, $F_{\rm am}$ and $F_{\rm as}$, are

$$F_{\rm am}(s) = (B_{\rm dm} + K_{\rm pm}/s)(V_{\rm h}(s) - V_{\rm e}(s)) \tag{2.21}$$

$$F_{\rm as}(s) = (B_{\rm ds} + K_{\rm ps}/s)(V_{\rm h}(s) - V_{\rm e}(s)) \tag{2.22}$$

We now discuss the three representations – hybrid matrix, equivalent circuit, general teleoperation system architecture – based on this example system.

Table 2.1: Variables and parameters for a bilateral teleoperation system representation [82].

Var & Param	Description
Z_{m}	Master impedance
Z_{s}	Slave impedance
C_{m}	Master controller
C_{s}	Slave controller
C_1	Velocity channel from master to slave
C_2	Force channel from slave to master
C_3	Force channel from master to slave
C_4	Velocity channel from slave to master
Z_{h}	Human operator impedance
Z_{e}	Environment impedance
F_{h}	Force applied by human operator to local manipulator
F_{e}	Force from environmental contact to remote telerobot
F_{am}, F_{as}	Actuator force of master/slave
V_{h}, v_{h}	Velocity of local manipulator
V_{e}, v_{e}	Velocity of remote telerobot
F_{h}^*	Human operator exogenous force input
F_{e}^*	Environment exogenous force input
M_{m}, M_{s}	Mass, master/slave
B_{m}, B_{s}	Damper, master/slave
K_{pm}, K_{ps}	Position gain or Stiffness of master/slave
B_{dm}, B_{ds}	Derivative gain or Damper of master/slave

Hybrid matrix representation

The first representation for bilateral teleoperation system is hybrid matrix[2] representation in [64]:

$$\begin{bmatrix} F_{\mathrm{h}}(s) \\ -V_{\mathrm{e}}(s) \end{bmatrix} = \begin{bmatrix} h_{11}(s) & h_{12}(s) \\ h_{21}(s) & h_{22}(s) \end{bmatrix} \begin{bmatrix} V_{\mathrm{e}}(s) \\ F_{\mathrm{h}}(s) \end{bmatrix} \tag{2.23}$$

[2]The term "hybrid matrix" comes from H-equivalent 2-port model in which input-to-output relationship is defined by h-parameters (hybrid parameters).

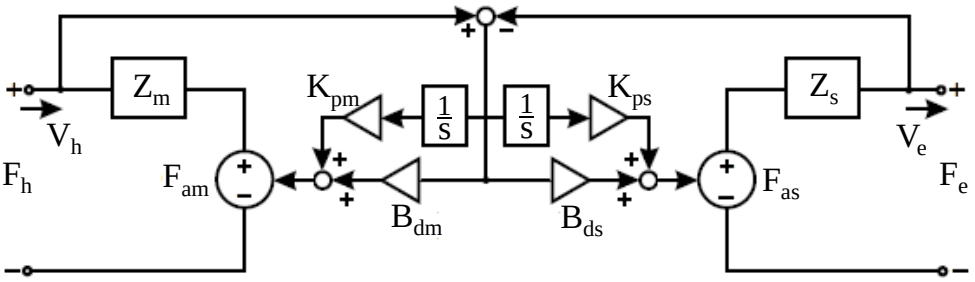


Figure 2.5: An example system for three representations.

where

$$\begin{aligned} h_{11}(s) &= \left.\frac{F_{\mathrm{h}}}{V_{\mathrm{h}}}\right|_{F_{\mathrm{e}}=0}, \quad h_{12}(s) = \left.\frac{F_{\mathrm{h}}}{F_{\mathrm{e}}}\right|_{V_h=0}, \\ h_{21}(s) &= \left.\frac{V_{\mathrm{e}}}{V_{\mathrm{h}}}\right|_{F_{\mathrm{e}}=0}, \quad \text{and} \quad h_{22}(s) = \left.\frac{V_{\mathrm{e}}}{F_{\mathrm{e}}}\right|_{V_{\mathrm{h}}=0} \end{aligned} \tag{2.24}$$

The physical interpretation of this matrix is [64, 66]

$$H = \begin{bmatrix} Z_{\mathrm{in}} & \mathrm{ReverseForceScale} \\ \mathrm{VelocityScale} & Z_{\mathrm{out}}^{-1} \end{bmatrix} \tag{2.25}$$

For an ideal teleoperation system, we want to have

$$H_{\mathrm{ideal}} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \tag{2.26}$$

The hybrid matrix representation also serves as the basis for several theoretical contributions such as the scattering operator S [66]

$$S(s) = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} (H(s) - I)(H(s) + I)^{-1} \tag{2.27}$$

which can tell us the passivity of a system by examining $\|S(jw)\|_\infty \leq 1$, where $\|\cdot\|$ represents an induced norm (operator norm) [83].

Equivalent circuit representation

A bilateral teleoperation system can also be represented as an equivalent electrical circuit. This representation allows us to analyze a mechanical system with the techniques from circuit analysis, e.g., the node-voltage method

or mesh-current method. Here we adopt an equivalent circuit representation presented in [67].

Fig. 2.6 shows the equivalent circuit representation of bilateral teleoperation system with no communication delay between master and slave. Let $x_{\rm h}, x_{\rm e}$ and $v_{\rm h}, v_{\rm e}$ be positions and velocities of master/slave manipulators. Then the system equation can obtained by applying the mesh-current method

$$F_{\rm h} - F_{\rm a} = M_{\rm m}\dot{v}_{\rm h} + B_{\rm m}v_{\rm h} = M_{\rm m}\ddot{x}_{\rm h} + B_{\rm m}\dot{x}_{\rm h} \tag{2.28}$$

$$F_{\rm a} - F_{\rm e} = M_{\rm s}\dot{v}_{\rm e} + B_{\rm s}v_{\rm e} = M_{\rm s}\ddot{x}_{\rm e} + B_{\rm s}\dot{x}_{\rm e} \tag{2.29}$$

where the interactive control force for master/slave, $F_{\rm a}$, is given by

$$\begin{aligned} F_{\rm a} &= B_{\rm d}(v_{\rm h} - v_{\rm e}) + K_{\rm p}\int (v_{\rm h} - v_{\rm e})dt \\ &= B_{\rm d}(\dot{x}_{\rm h} - \dot{x}_{\rm e}) + K_{\rm p}(x_{\rm h} - x_{\rm e}) \end{aligned} \tag{2.30}$$

From the example system shown in Fig. 2.5, we assume that there exists no communication delay between master and slave, two actuators at master/slave have the same parameters ($B_{\rm dm} = B_{\rm ds} = B_{\rm d}$ and $K_{\rm pm} = K_{\rm ps} = K_{\rm p}$), and master/slave impedance consists of mass and damper. Then, we have

$$F_{\rm am}(s) = F_{\rm as}(s) = (B_{\rm d} + K_{\rm p}/s)(V_{\rm h}(s) - V_{\rm e}(s)) = F_{\rm a}(s) \tag{2.31}$$

and

$$Z_{\rm m}(s) = M_{\rm m}s + B_{\rm m}, \text{ and } Z_{\rm s}(s) = M_{\rm s}s + B_{\rm s} \tag{2.32}$$

and the two representations in Fig. 2.5 and Fig. 2.6 become identical. Namely,

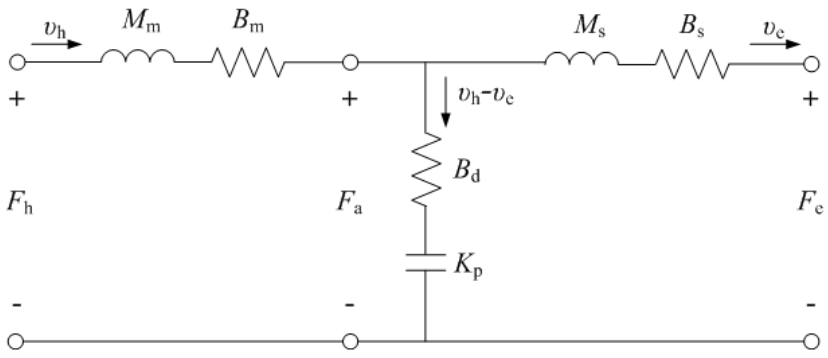


Figure 2.6: Equivalent circuit representation for the example system.

the two representations are identical under certain assumptions as presented above.

General bilateral teleoperation system representation

We have presented two representations, the hybrid matrix and the equivalent circuit representations. In fact, there exists a general representation which encompasses the former representations [82, 84].

Fig. 2.7 depicts the general bilateral teleoperation system architecture proposed in [82]. By setting the impedance and interactive control/communication block elements, $Z(s)$ and $C(s)$, we can explicitly obtain a specific type of bilateral teleoperation system. To obtain the example system of Fig. 2.5, for instance, we need to set [82]

$$\begin{aligned}
&Z_{\mathrm{m}}(s) = M_{\mathrm{m}}s + B_{\mathrm{m}}, \quad Z_{\mathrm{s}}(s) = M_{\mathrm{s}}s + B_{\mathrm{s}}, \\
&C_{\mathrm{m}}(s) = B_{\mathrm{dm}} + K_{\mathrm{pm}}/s, \quad C_{\mathrm{s}}(s) = B_{\mathrm{ds}} + K_{\mathrm{ps}}/s, \\
&C_1(s) = B_{\mathrm{ds}} + K_{\mathrm{ps}}/s, \quad C_4(s) = -(B_{\mathrm{dm}} + K_{\mathrm{pm}}/s), \\
&C_2(s) = C_3(s) = \text{ not used} \\
&Z_{\mathrm{h}}(s) = Z_{\mathrm{e}}(s) = \text{ not a function of control architecture}
\end{aligned} \tag{2.33}$$

and

$$B_{\mathrm{dm}} = B_{\mathrm{ds}} = B_{\mathrm{d}} \text{ and } K_{\mathrm{pm}} = K_{\mathrm{ps}} = K_{\mathrm{p}} \tag{2.34}$$

Then the architecture satisfying (2.33) becomes identical to the example system as well as the equivalent circuit representation.

2.4.3 Special Case of Bilateral Teleoperation System: Passivity-Based PD Controller

Under specific conditions, the bilateral teleoperation system can be interpreted as a passivity-based PD controller. This characteristic can be a useful tool when we want to design a bilateral teleoperation system of which master and slave have different kinematics, e.g., a bilateral teleoperation for a wheeled mobile robot.

Recall (2.28) and (2.29) from the equivalent circuit representation. In addition to the former assumption, no time delay, $B_{\mathrm{dm}} = B_{\mathrm{ds}} = B_{\mathrm{d}}$, and

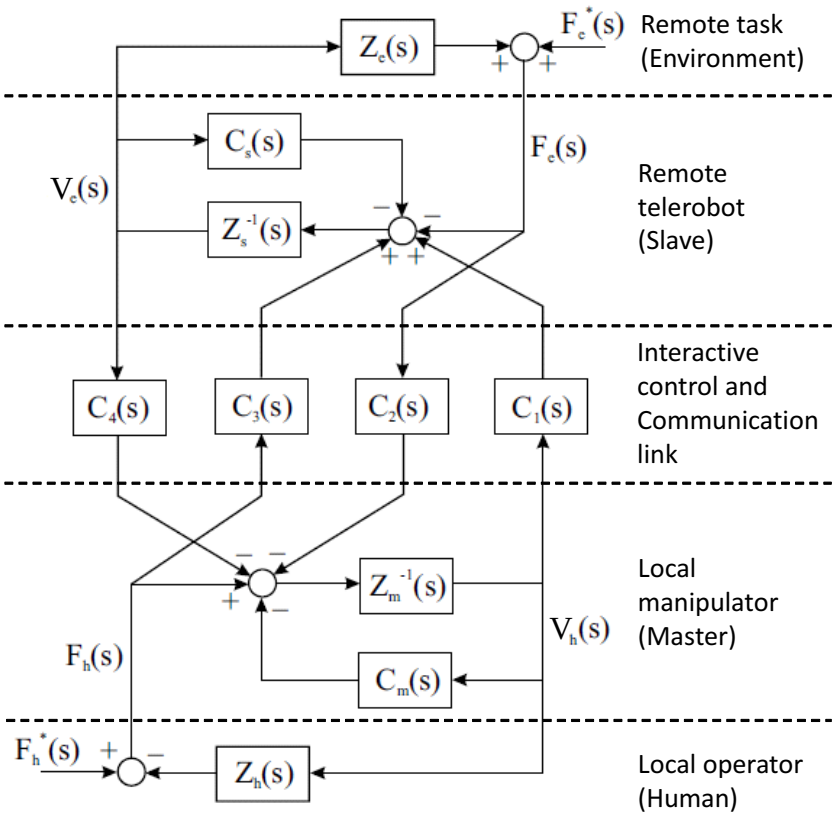


Figure 2.7: General bilateral teleoperation system architecture in [82].

$K_{\mathrm{pm}} = K_{\mathrm{ps}} = K_{\mathrm{p}}$, suppose that master and slave have the same mass and damper, $M_{\mathrm{m}} = M_{\mathrm{s}} = M$ and $B_{\mathrm{m}} = B_{\mathrm{s}} = B$. Thus, (2.28) and (2.29) become (we omit the velocity related term in below)

$$F_{\mathrm{h}} - F_{\mathrm{a}} = M\ddot{x}_{\mathrm{h}} + B\dot{x}_{\mathrm{h}} \tag{2.35}$$

$$F_{\mathrm{a}} - F_{\mathrm{e}} = M\ddot{x}_{\mathrm{e}} + B\dot{x}_{\mathrm{e}} \tag{2.36}$$

To observe the entire system behavior in a perspective of regarding the interactive control force F_{a} as a control input u to the system, we subtract (2.36) from (2.35)

$$M(\ddot{x}_{\mathrm{h}} - \ddot{x}_{\mathrm{e}}) + B(\dot{x}_{\mathrm{h}} - \dot{x}_{\mathrm{e}}) = F_{\mathrm{h}} - 2F_{\mathrm{a}} + F_{\mathrm{e}} \tag{2.37}$$

Substituting (2.30) into (2.37) yields

$$M(\ddot{x}_{\mathrm{h}} - \ddot{x}_{\mathrm{e}}) + B(\dot{x}_{\mathrm{h}} - \dot{x}_{\mathrm{e}}) = F_{\mathrm{h}} - 2B_{\mathrm{d}}(\dot{x}_{\mathrm{h}} - \dot{x}_{\mathrm{e}}) - 2K_{\mathrm{p}}(x_{\mathrm{h}} - x_{\mathrm{e}}) + F_{\mathrm{e}} \tag{2.38}$$

Now, suppose $F_{\mathrm{h}} = F_{\mathrm{e}} = 0$ which means that neither the human operator applies force to a manipulator at master nor the remote telerobot touches/contacts the environment. Let $e = x_{\mathrm{h}} - x_{\mathrm{e}}$, we have

$$\begin{aligned} M\ddot{e} + B\dot{e} &= -2B_{\mathrm{d}}\dot{e} - 2K_{\mathrm{p}}e \\ &=: u \end{aligned} \tag{2.39}$$

which takes the form of a typical system in which a mass-damper system, M and B, is controlled by passivity-based PD controller $u = -2B_{\mathrm{d}}\dot{e} - 2K_{\mathrm{p}}e$ in [85].

In sum, by imposing some specific conditions as (2.35) through (2.39), we can represent a bilateral teleoperation system as a typical passivity-based PD controlled system. The usages of the derived fact in above for our system design are as follows:

- When master/slave kinematics are different, e.g., mobile robot teleoperation with haptic joystick, we first assume virtual bilateral connection of identical mass-damper values at master/slave together with damper/spring interactive control.

- Assume that force F_{h} is applied to set desired velocity $\dot{x}_{\mathrm{h}}$. One way to define $\dot{x}_{\mathrm{h}}$ is to simply use a scalar multiple of joystick configuration. The interactive control force is defined as $F_{\mathrm{a}} = B_{\mathrm{d}}(\dot{x}_{\mathrm{h}} - \dot{x}_{\mathrm{e}}) + K_{\mathrm{p}}(x_{\mathrm{h}} - x_{\mathrm{e}})$, and no slave-environment contact $F_{\mathrm{e}} = 0$.

- For the virtual bilateral connection, we can regard master and slave as a desired velocity setter and a mass-damper system which is controlled by passivity-based PD control, respectively.

- Convert the velocity of the mass-damper system into the actual velocity of a real slave system, e.g., a mobile robot.[3]

[3] A similar approach can be found in [86].

2.5 Preview about Relationship between Background and the Subsequent Chapters

In this chapter, we discussed a virtual fixture, assistive/cooperative system, inverse optimal control, and bilateral teleoperation as the background. Relationships between the background and the subsequent chapters are as follows:

- First, a bilateral teleoperation architecture will be used to analyze stability of our developed assistive HRI system in Chapter 3. We will see that the stability of the developed system can be guaranteed by showing that it is the cascade connection of two passive systems.
- Second, we will model a user's task-performing characteristics using the techniques from inverse optimal control in Chapter 4. We will also discuss a classification method that allows us to categorize human subjects based on their task-performing characteristics.
- Third, Chapter 5 will describe our approach to employing a virtual fixture to generate assisted control and haptic feedback. We will discuss our approach to customizing haptic feedback according to the user's task-performing characteristics and performance.
- Fourth, the effect of feedback that we have discussed in assistive/cooperative systems will be considered in Chapter 6, where we present a method to customize visual feedback.

2.6 Further Resources

There has been a large volume of findings and literature related to background that we have covered through this chapter. Here we introduce selective resources that can give the readers more insight. A historical survey of bilateral teleoperation is presented in [66]. This survey paper covers not only the history of bilateral teleoperation but also frameworks, representations, approaches to stability analysis, and extended applications. For virtual fixtures, [1] introduces history, terminologies and types, and provides a wide survey of real-world virtual fixture applications. In [87], the architecture to

implement a teleoperation system according to various master/slave manipulator types is described. Lastly, [81] concerns a method to involve human operator characteristics as a part of assistive teleoperation systems.

CHAPTER 3

SYSTEM ARCHITECTURE

This chapter presents our approach to designing an assistive human-robot interaction (HRI) interface that provides haptic and visual feedback to enhance a human user's task-performance. The specific target application, in which a user controls a mobile robot by a haptic input device along a road while monitoring the robot with a visual display, will be considered throughout this chapter. We discuss system apparatus, system modeling, and stability analysis of the developed interface. Among the background material introduced in Chapter 2, the system modeling is mainly related to bilateral teleoperation. Specifically, a bilateral teleoperation architecture is used as a backbone of our design approach to generate assisted control, and to provide haptic feedback by a guiding force.

First, this chapter begins by describing the apparatus of the designed assistive HRI interface. The designed interface consists of an input device, a display device, a mobile robot, and is equipped with two controllers called an *assistive controller* and a *perceptual feedback controller*, respectively. To enhance the user's task-performance, the assistive controller yields the assisted control by blending a user's control input with the controller's intention which will be called a *guiding force*. The perceptual feedback controller provides haptic (force) feedback and visual feedback to help the user to be aware of the quality of his/her control.

Second, we introduce our system modeling method following a top-down approach. We present two representations of the entire system, feedback control system representation and bilateral teleoperation architecture representation, as an overview on our system models. Then, we discuss the subsystem models: a haptic joystick, a mobile robot, and a virtual omnidirectional mass-damper system which is employed as a surrogate system of the mobile robot to manage kinematic discrepancy to the haptic joystick. Accordingly, we explain how the velocity of the omnidirectional mass-damper

system is converted to the velocity of mobile robot.

Third, we discuss our assistive control scheme and the corresponding perceptual feedback. The user sets desired velocity by the configuration of the haptic joystick. The interactive controller generates control force so that the omnidirectional mass-damper system tracks the desired velocity. Then the assistive controller blends the generated control force with the guiding force to modify, and hopefully improve, the user's control effort. Both the control force and the guiding force are used as resources to determine the properties of haptic feedback, e.g., magnitude and direction, to the user. Visual feedback is modified based on deviation from the desired path in a given task.

Finally, as our system is designed on a basis of bilateral control architecture, we derive conditions in order to guarantee the stability of the designed system. After deriving the conditions, the stability can be guaranteed by showing the system is indeed the cascade connection of two passive systems.

3.1 Assistive HRI Interface Platform

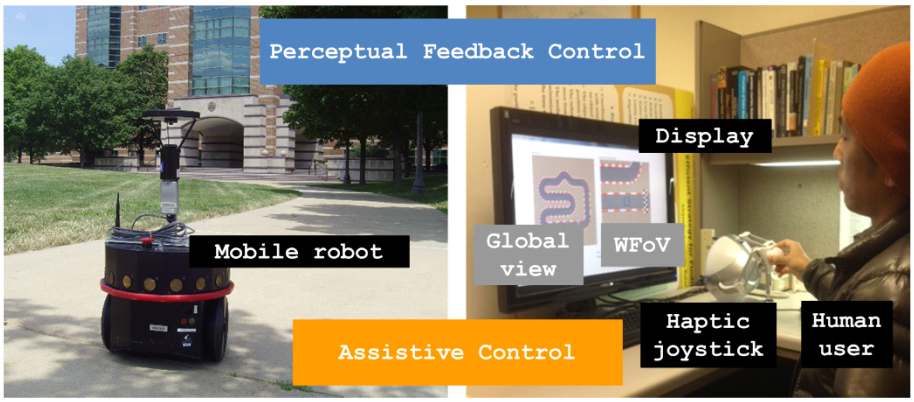
In this section, we discuss our assistive HRI interface platform illustrated in Fig. 3.1. We introduce the system apparatus, component features and specifications, and two controllers: an assistive controller and a perceptual feedback controller.

3.1.1 System Apparatus

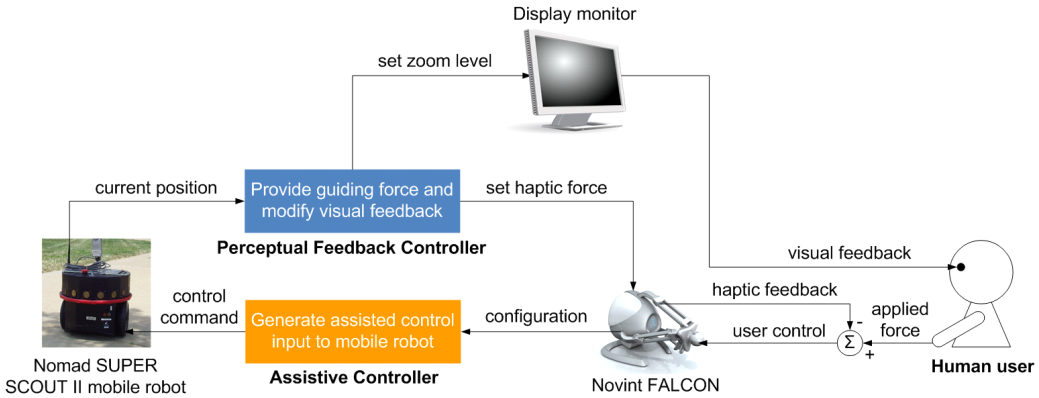
Haptic interface and visual display

We use a Novint FALCON haptic joystick as an input device. The major technical specification of Novint FALCON are: 3D touch workspace 4" $\times$ 4" $\times$ 4", force capability 2-lbs, position resolution 400-dpi, size 9" $\times$ 9" $\times$ 9", weight 6-lbs, and servo rate 1-kHz [88].[1] A typical display monitor can be used as a display device.

[1]For more details about the FALCON, see http://www.novint.com/index.php/novintxio/41.



(a) The apparatus of the assistive HRI interface



(b) Assistive control and perceptual feedback control for a human user

Figure 3.1: The assistive HRI interface platform.

Mobile robot

We adopt a Nomad Super Scout II as our controlled object (which will be simply referred to as the mobile robot). This mobile robot is driven by two differential drive wheels, with a caster at rear. Here, we introduce the brief specification as follows: height 35cm, weight 28kg, diameter 41cm, ground clearance 1.5cm, maximum speed 1m/s, maximum acceleration 2m/s^2, and battery running time 1-hour [89, 90].[2]

[2]For a manual and software, visit http://nomadic.sourceforge.net/production/scout/.

3.1.2 Assistive Control and Perceptual Feedback Control

Fig. 3.1(b) illustrates the assistive control and perceptual feedback pathways. First, by the term *assistive control* we mean blending a human user's control input with assistance of the developed HRI interface. Second, the term *perceptual feedback control* or *controlling perceptual feedback* refers to the active adjustment of two kinds of perceptual feedback – haptic feedback and visual feedback – based on control input from the human user under the particular environmental conditions. In other words, our interface adaptively provides guidance, and helps the user to be aware of the quality of his/her control.

3.1.3 Objectives of This Development

We want to develop an assistive interface with customized feedback for a specific user to enhance the user's task-performance. We have objectives for this development as follows : i) help low-performing users by incorporating strategies of high-performing users into the HRI interface, ii) allow the user to apply a control input more confidently by reducing pressure for the finer control, and iii) protect people like drivers or pilots who get distracted by fatigue, boredom, or substance abuse.

3.2 System Modeling

In this section, we describe our modeling approach from entire system model to subsystem models. For the entire system model, we present two representations, a feedback control system representation and a bilateral teleoperation architecture representation. As the former representation is a more familiar form to exemplify a control system with a state feedback, we use it to explain subsystem models and dynamics. Considering that our system is indeed a bilateral control system, i.e., providing haptic feedback to the user, we also present the latter representation, and especially refer to it during the stability discussion in Section 3.4.

We will see that the roles of both assistive control and perceptual feedback control are closely related to two blocks which are called *guiding force generator* and *zoom level adjuster*, respectively. Note that we restrict our attention

to explaining the concepts throughout the section, and will further discuss our methods to define a guiding force and a zoom level, and to customize their values based on the user's task-dexterity in Chapter 5 and in Chapter 6, respectively.

3.2.1 Entire System Model

Feedback control system representation

Fig. 3.2 shows a typical feedback control system representation of the entire system model. We now explain the assistive control pathway and the perceptual feedback control pathway.

First, the assistive control pathway starts from setting desired velocity $\dot{\boldsymbol{x}}_{\mathrm{h}}$ by a haptic joystick configuration $\boldsymbol{q}$. The error between the desired velocity and the current velocity, $\dot{\boldsymbol{x}}_{\mathrm{h}} - \dot{\boldsymbol{x}}_{\mathrm{e}}$, goes into a PD-controller, generates control force $\boldsymbol{f}$ blended with a guiding force $\boldsymbol{f}_g$, and finally $\boldsymbol{f} + \boldsymbol{f}_g$ controls an omnidirectional mass-damper system. The omnidirectional mass-damper system is employed as a surrogate system that manages kinematic discrepancy between the haptic joystick and the mobile robot. We will discuss the role of the omni-directional mass-damper system as well as velocity conversion in Section 3.2.2.

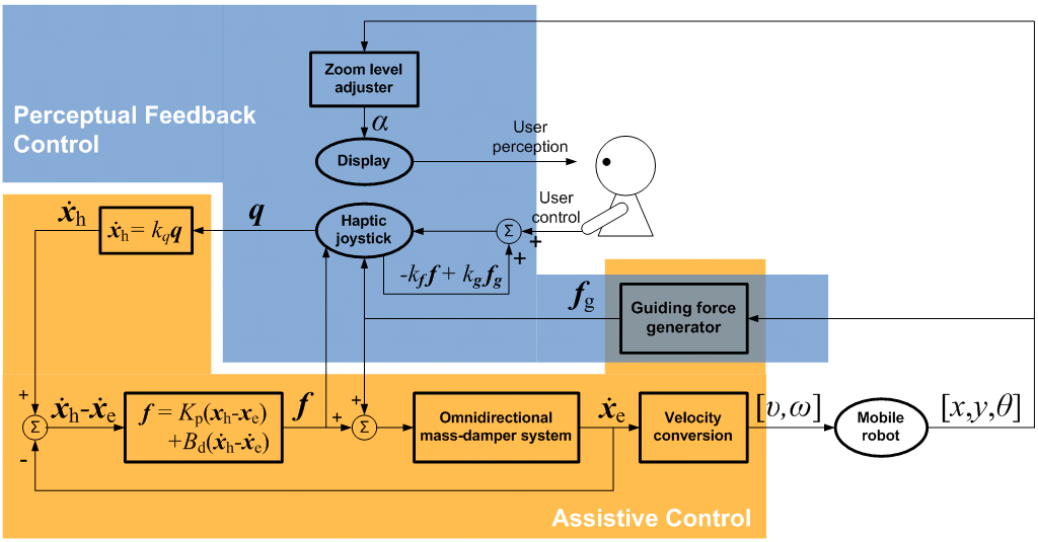


Figure 3.2: A functional block representation of the entire system.

The perceptual feedback control pathway includes two kinds of feedback: haptic feedback and visual feedback. The haptic feedback is calculated as the summation of the two scaled terms $-k_f \boldsymbol{f}$ and $k_g \boldsymbol{f}_g$. The visual feedback is an image of working field of view (WFoV) around the mobile robot. The zoom level adjuster will modify the size of the WFoV by zooming in and out by a level α.

Bilateral teleoperation architecture representation

Fig. 3.3 depicts the bilateral teleoperation architecture represention of the entire system, which is equivalent to Fig. 3.2. In Section 2.4, we discussed the special case of bilateral teleoperation: the master works as reference velocity setter, the interactive controller performs PD-control, and the slave tracks the reference velocity.

Likewise, we use the haptic joystick as the reference velocity setter by its configuration. Also, a human user receives haptic feedback $\boldsymbol{f}_{\text{haptic}}$ (same as $F_{\text{haptic}}(s)$ in s-domain). We set the impedances and the interactive control block elements as follows:

$$\begin{aligned} &Z_{\text{m}}(s) = \text{ not used}, \quad Z_{\text{s}}(s) = M_{\text{s}}s + B_{\text{s}}, \\ &C_{\text{m}}(s) = B_{\text{dm}} + K_{\text{pm}}/s, \quad C_{\text{s}}(s) = B_{\text{ds}} + K_{\text{ps}}/s, \\ &C_1(s) = B_{\text{ds}} + K_{\text{ps}}/s, \quad C_2 = 1, \quad C_4(s) = -(B_{\text{dm}} + K_{\text{pm}}/s), \\ &C_3(s) = \text{ not used} \end{aligned} \tag{3.1}$$

3.2.2 Subsystem Models

Joystick kinematics and mobile robot kinematics

First, we set joystick configuration, $(q_x, q_y) \in \mathbb{R}^2$, as follows:[3]

q_x : position on x-axis along left/right direction w.r.t. to the user

q_y : position on y-axis along up/down direction w.r.t. to the user

[3]In fact, the employed Novint FALCON haptic joystick has 3-DOF. As we deal with the specific application for mobile robot control, however, we disregard z-axis value which correspond to the depth, near or far from the user's body, of joystick configuration.

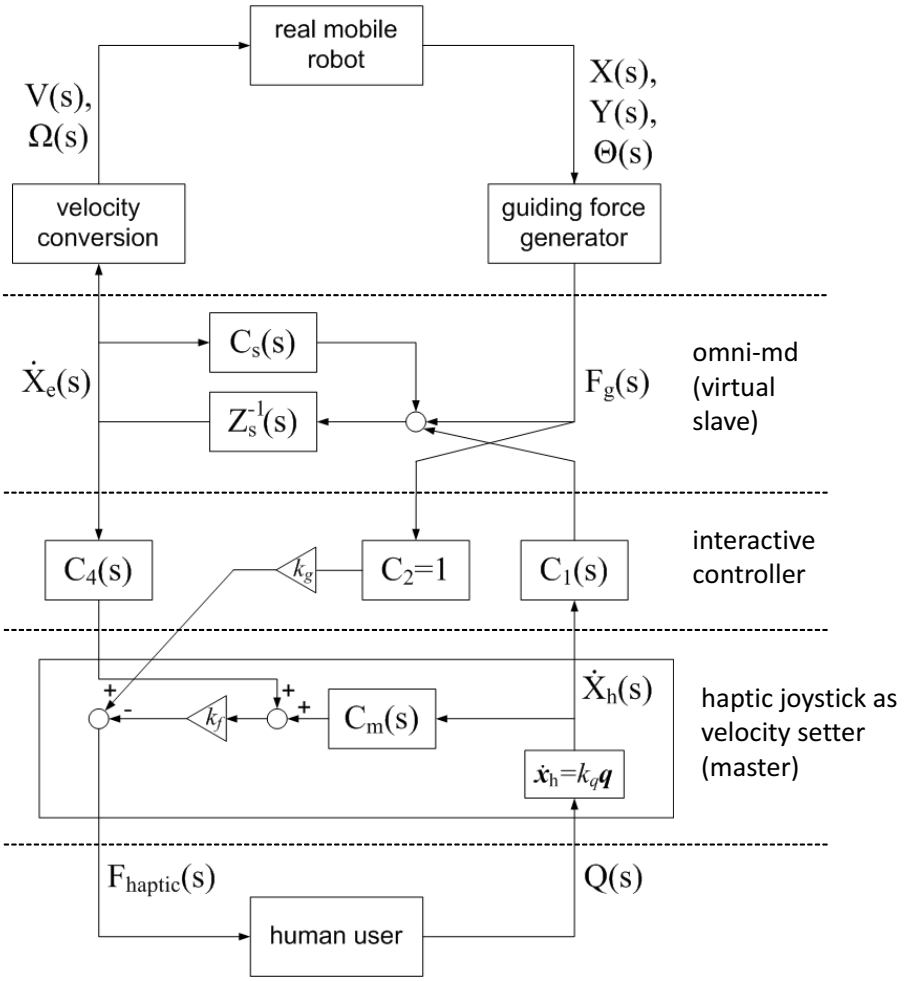


Figure 3.3: Bilateral control representation of the entire system.

Let v_x and v_y be two linear velocities that move along with a human user's hand movements toward the x-y direction, respectively. The joystick kinematics is given by

$$\begin{bmatrix} \dot{q}_x \\ \dot{q}_y \end{bmatrix} = k_v \begin{bmatrix} v_x \\ v_y \end{bmatrix} \tag{3.2}$$

where k_v is positive scalar determined by the joystick encoder specification.

Next, let x, y and θ be the position and orientation of a mobile robot, $(x, y, \theta) \in \mathbb{R}^2 \times \mathbb{S}^1$, respectively. The mobile robot is a nonholonomic differential drive robot which takes two control inputs, v(speed) and ω(steering

angle). The kinematic equation of the mobile robot is [91]

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{3.3}$$

In a typical bilateral system, the kinematics of master/slave systems are identical. From (3.2) and (3.3), however, we can see that it is not straightforward to pair up a haptic joystick and a mobile robot as master/slave due to kinematic discrepancy between them. Therefore, we introduce our method to manage the kinematic discrepancy below.

Virtual omnidirectional mass-damper system to manage kinematic discrepancy

We manage the kinematic discrepancy by inserting a virtual omnidirectional mass-damper system between the haptic joystick and the real mobile robot. The omnidirectional mass-damper system aims to substitute the real mobile robot, to work as a virtual remote telerobot, and eventually to obtain a general bilateral teleoperation architecture by allowing a human operator to control the real mobile robot via the surrogate system.[4] From now on, the omnidirectional mass-damper will be referred to as *omni-md*.

Regarding the omni-md as the surrogate system of the real mobile robot at the slave side, we use subscripts 'e' and 's' to represent the variables and parameters of the omni-md. Fig. 3.4 depicts the control and the simplified representation of the omni-md. Let $x_{\mathrm{e}}, y_{\mathrm{e}}, \dot{x}_{\mathrm{e}}, \dot{y}_{\mathrm{e}}$ be the position and velocity of the omni-md. From Fig. 3.4, we know that the omni-md takes the summation of $\boldsymbol{f} = [f_x, f_y]^T$ and $\boldsymbol{f}_{\mathrm{g}} = [f_{x,\mathrm{g}}, f_{y,\mathrm{g}}]^T$ as its control input. We call $\boldsymbol{f}$ and $\boldsymbol{f}_{\mathrm{g}}$ a *control force* and a *guiding force*, respectively. An intuitive way to describe the dynamics of omni-md system is – as a ball of M_{s} and B_{s} being either pulled or pushed by $\boldsymbol{f} + \boldsymbol{f}_{\mathrm{g}}$.

[4] A similar approach using a concept of virtual mass has been proposed in [86]. Here, we use the term "omnidirectional mass-damper" to prevent the reader's confusion with a virtual fixture, and the overuse of the word "virtual."

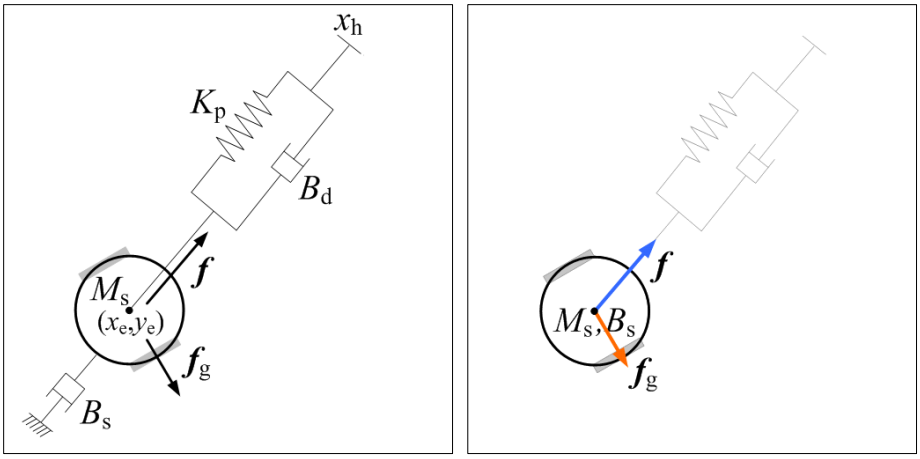


Figure 3.4: The control of omni-md by force $\boldsymbol{f} + \boldsymbol{f}_g$ (left) and the simplified representation (right).

The system equation of the omni-md is given by

$$\begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \ddot{x}_e \\ \ddot{y}_e \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{B_s}{M_s} & 0 \\ 0 & 0 & 0 & -\frac{B_s}{M_s} \end{bmatrix} \begin{bmatrix} x_e \\ y_e \\ \dot{x}_e \\ \dot{y}_e \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{M_s} & 0 \\ 0 & \frac{1}{M_s} \end{bmatrix} \begin{bmatrix} f_x + f_{x,g} \\ f_y + f_{y,g} \end{bmatrix} \tag{3.4}$$

where M_s and B_s are the mass and damping of the omni-md, respectively. The output from the omni-md block, $\dot{\boldsymbol{x}}_e$, is

$$\dot{\boldsymbol{x}}_e = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_e \\ y_e \\ \dot{x}_e \\ \dot{y}_e \end{bmatrix} \tag{3.5}$$

which are x-y direction velocities of the omni-md.

Velocity conversion: Translating omni-md's velocity into a real mobile robot's velocity

As the real mobile robot is a nonholonomic differential drive robot taking a speed v and a steering ω as its control command, we need to convert the two

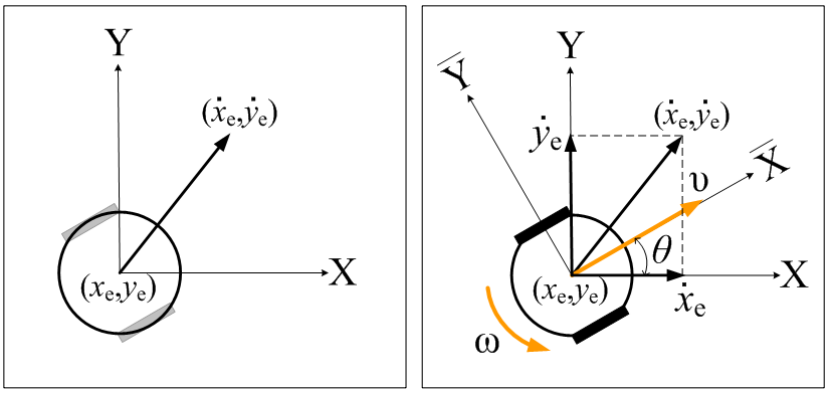


Figure 3.5: A linear velocity of the omni-md, $(\dot{x}_e, \dot{y}_e)$, resulted from $\boldsymbol{f} + \boldsymbol{f}_g$ (left), and a speed v and a steering ω control command to the mobile robot by velocity conversion (right).

linear velocities of the omni-md.

$$\begin{bmatrix} \dot{x}_e \\ \dot{y}_e \end{bmatrix} \overset{\Phi}{\longmapsto} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{3.6}$$

The velocity conversion is illustrated in Fig. 3.5. Following [86], the velocity conversion can be achieved by aligning the x-axis to the current heading direction θ of the mobile robot, then scaling the aligned velocity vectors, denoted by $\dot{\bar{x}}_e$ and $\dot{\bar{y}}_e$, of the omni-md:

$$\begin{aligned} \begin{bmatrix} v \\ \omega \end{bmatrix} &= \xi \begin{bmatrix} c_{\dot{\bar{x}}_e} & 0 \\ 0 & c_{\dot{\bar{y}}_e} \end{bmatrix} \begin{bmatrix} \dot{\bar{x}}_e \\ \dot{\bar{y}}_e \end{bmatrix} \\ &= \xi \begin{bmatrix} c_{\dot{\bar{x}}_e} & 0 \\ 0 & c_{\dot{\bar{y}}_e} \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \end{bmatrix} \end{aligned} \tag{3.7}$$

Hence, we can define Φ as

$$\Phi(a,b) := \xi \begin{bmatrix} c_{\dot{\bar{x}}_e} & 0 \\ 0 & c_{\dot{\bar{y}}_e} \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{pmatrix} a \\ b \end{pmatrix} \tag{3.8}$$

where $c_{\dot{\bar{x}}_e}$ and $c_{\dot{\bar{y}}_e}$ are the positive scaling constants that can be determined by the maximum velocity specification of real mobile robot.

In (3.8), we refer to ξ as the *control command gain*, and it is used to sustain consistent control-to-display (CD) gain with respect to varying zoom level.[5]

[5]Control-to-display (CD) gain defines a mapping between a workspace (in meters) and

We will discuss the explicit definition of ξ in Chapter 6.

3.3 Assistive Control to Support a Human User

3.3.1 Control Force to the Omni-MD by a Haptic Joystick

In a bilateral control perspective, we can regard the entire system architecture as if a haptic joystick at the master side provides the desired velocity, and that an omni-mass at the slave side asymptotically tracks the desired velocity [68]. Inspired by this fact, the control of the omni-md is achieved by the following procedure. Let $\boldsymbol{x}_{\mathrm{h}} = [x_{\mathrm{h}}, y_{\mathrm{h}}]^T$ be the desired position of the omni-mass. We define $\dot{\boldsymbol{x}}_{\mathrm{h}}$ to be

$$\dot{\boldsymbol{x}}_{\mathrm{h}} = k_q \boldsymbol{q} \tag{3.9}$$

where $\boldsymbol{q} = [q_x, q_y]^T$ is the haptic joystick configuration, and k_q is a scalar which can be determined by haptic joystick specification. We define control input force in (3.4) , $\boldsymbol{f} = [f_x, f_y]^T$, to the omni-mass as

$$\boldsymbol{f} = B_{\mathrm{d}}(\dot{\boldsymbol{x}}_{\mathrm{h}} - \dot{\boldsymbol{x}}_{\mathrm{e}}) + K_{\mathrm{p}}(\boldsymbol{x}_{\mathrm{h}} - \boldsymbol{x}_{\mathrm{e}}) \tag{3.10}$$

where B_{d} and K_{p} are position and derivative gain, respectively. We note that (3.10) is indeed identical to the interactive control force of (2.30) in Section 2.4.2.

3.3.2 Blending Control Input with Guiding Force

The assistive controller blends a control input set by a user with a guiding force. After the control blending, the *assisted control* input, $\boldsymbol{f}_{\mathrm{assisted}}$, becomes

$$\boldsymbol{f}_{\mathrm{assisted}} = \boldsymbol{f} + \boldsymbol{f}_g = \begin{bmatrix} f_x \\ f_y \end{bmatrix} + \begin{bmatrix} f_{\mathrm{x,g}} \\ f_{\mathrm{y,g}} \end{bmatrix} \tag{3.11}$$

which is the actual control input to the omni-mass in (3.4).

As we have briefly mentioned in Section 3.2.1, haptic feedback to the hu-

a screen (in pixels) [92].

man user is given by

$$\boldsymbol{f}_{\text{haptic}} = -k_f \boldsymbol{f} + k_g \boldsymbol{f}_g \tag{3.12}$$

where both $k_{\boldsymbol{f}}$ and k_g are positive scalar gain. For our specific application, we set $k_{\boldsymbol{f}}$ and k_g to fulfill

$$\boldsymbol{f}_{\text{haptic}} \simeq k_g \boldsymbol{f}_g \tag{3.13}$$

which makes the user feels the guiding force.

3.4 Stability of the Developed Assistive HRI Interface

In this section, we discuss stability of the developed assistive HRI interface. To the best of our knowledge, there exist no stability analysis techniques that concern the stability for a system which is perfectly matched to our design approach, i.e., master/slave kinematics are not identical, a guidance virtual fixture is adopted, and a guiding force is provided as haptic feedback as well as assisted control input. Therefore, we first briefly review previous approaches to show stability in similar applications. Then, we derive the conditions to guarantee the stability for our HRI assistive interface using a passivity-based approach.

3.4.1 Previous Approaches to Show the Stability in Similar Applications

In [18, 93] the stability of a telemanipulation system with a forbidden-region virtual fixture (FRVF) was presented. Stability with a guidance virtual fixture (GVF) was discussed in [13], however the definition of stability was characterized in special fashion. Furthermore, in all cases to date, the major difference from our system architecture is that master and slave devices are identical in their system architecture.

The stability of bilateral teleoperation of a wheeled mobile robot was shown in [76, 77]. In their system design, a user was allowed to apply the rotational movement of haptic manipulator along z-axis, which in turn became the steering input of the mobile robot. However, the configuration of our haptic joystick is defined by pure translations.

The closest application that we have found is mobile robot teleoperation

with varying force feedback gain in [80]. Compared to our design, [80] used potential-based function to obtain repulsive force feedback. The repulsive force only acted on the human operator, but the guiding force in our application has an effect on the human operator as well as on the control input to the controlled object. Although there exist differences in design approaches, [80] presented a stability analysis based on the Routh-Hurwitz stability criterion.

Recently, stability analysis techniques that do not depend on passivity-based analysis for bilateral control systems have been proposed in [81, 94]. However, their application design is very different from ours.

3.4.2 Stability Condition for Our HRI Interface

Regarding our system as a specific type of bilateral control system, e.g., no time delay, master as reference velocity setter, etc. We focus on deriving the conditions that must be imposed to guarantee stability instead of providing a mathematical derivation. With the derived conditions, we can guarantee stability as our HRI system is indeed the cascade connection of a haptic joystick as reference velocity setter, an interactive controller as a passivity-based PD controller, an omni-md system, and a real mobile robot.

Strictly speaking, our stability analysis approach is deriving the conditions, and heavily dependent on passivity-based analysis. However, the derived conditions can be used as a basis to implement a stable HRI system in practice.

Condition 1: A human operator is a passive system [84, 87].

In other words, the human operator can be modeled as passive system, e.g., a linear mass-damper-spring system. With this condition, we can regard the connection of a haptic device to a human user in Fig. 3.3 as a passive system (the haptic joystick has already been assumed as a stable reference velocity setter).

We have seen in Fig. 3.2 and Fig. 3.3 that the interactive controller works as a passivity-based PD-controller for the omni-md to track the reference velocity.

Condition 2: Velocity conversion is passive process, and the guiding force

is bounded. Velocity conversion has been defined in (3.7) as

$$\begin{aligned}\begin{bmatrix} v \\ \omega \end{bmatrix} &= \xi \begin{bmatrix} c_{\dot{\bar{x}}_\mathrm{e}} & 0 \\ 0 & c_{\dot{\bar{y}}_\mathrm{e}} \end{bmatrix} \begin{bmatrix} \dot{\bar{x}}_\mathrm{e} \\ \dot{\bar{y}}_\mathrm{e} \end{bmatrix} \\ &= \xi \begin{bmatrix} c_{\dot{\bar{x}}_\mathrm{e}} & 0 \\ 0 & c_{\dot{\bar{y}}_\mathrm{e}} \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \dot{x}_\mathrm{e} \\ \dot{y}_\mathrm{e} \end{bmatrix}\end{aligned}$$

We will define ξ as an inverse of a zoom level in Chapter 6, hence $0 \leq \xi < 1$. To find $c_{\dot{\bar{x}}_\mathrm{e}}$ and $c_{\dot{\bar{y}}_\mathrm{e}}$ values that make the velocity conversion process passive, we use Proposition 2 in [66] introducing the following Lyapunov function:

$$V(\dot{x}_\mathrm{m}, \dot{x}_\mathrm{s}) = \frac{1}{2} \begin{bmatrix} \dot{x}_\mathrm{m} \\ \dot{x}_\mathrm{s} \end{bmatrix} \begin{bmatrix} M_\mathrm{m} & 0 \\ 0 & M_\mathrm{s} \end{bmatrix} \begin{bmatrix} \dot{x}_\mathrm{m} \\ \dot{x}_\mathrm{s} \end{bmatrix} \tag{3.14}$$

where $\dot{x}_\mathrm{m}, M_\mathrm{m}$ and $\dot{x}_\mathrm{s}, M_\mathrm{s}$ represent the mass and the velocity of master/slave, respectively. Following [66], we use (3.14) to check the passivity of the system. By its physical interpretation, if kinetic energy loss occurs during the transfer between two ports, then we can say the transfer process is passive. Hence, we can express the passivity condition [86, 95] as

$$\frac{1}{2} M_\mathrm{s} (\dot{\bar{x}}_\mathrm{e}^2 + \dot{\bar{y}}_\mathrm{e}^2) \;\geq\; \frac{1}{2} M_\mathrm{R} v^2 + \frac{1}{2} J_\mathrm{R} \omega^2 \tag{3.15}$$

where M_R and J_R are the mobile robot's mass and the moment of inertia, respectively. From (3.7) and (3.15), we find the condition of $c_{\dot{\bar{x}}_\mathrm{e}}$ and $c_{\dot{\bar{y}}_\mathrm{e}}$ values as [86]

$$c_{\dot{\bar{x}}_\mathrm{e}} \leq \sqrt{\frac{M_\mathrm{s}}{M_\mathrm{R}}} \;\text{ and }\; c_{\dot{\bar{y}}_\mathrm{e}} \leq \sqrt{\frac{M_\mathrm{s}}{J_\mathrm{R}}} \tag{3.16}$$

The guiding force is the attenuated term from an initial control input by virtual fixturing, which will be discussed in Chapter 5; hence it is bounded.

Guranteed Stability: By condition 1 and condition 2, the assistive HRI system presented in Fig. 3.2 (and equivalently Fig. 3.3) can be regarded as the cascade connection of passive systems, hence the equilibrium $\mathbf{e} = \dot{\boldsymbol{x}}_\mathrm{h} - \dot{\boldsymbol{x}}_\mathrm{e}$ is asymptotically stable [68, 85].

3.5 Summary

In this chapter, we discussed our approach to designing an assistive HRI interface, assistive control, and perceptual feedback control. First, we described the apparatus of the designed interface platform, which consisted of a haptic joystick, visual display, mobile robot, assistive controller, and perceptual feedback controller. Second, we introduced our system modeling with a top-down approach: an entire system model, a haptic joystick, an omnidirectional mass-damper system, a real mobile robot, and velocity conversion. Specifically, the omnidirectional mass-damper system was inserted as a surrogate system of the real mobile robot to manage kinematic discrepancy. Third, we discussed an assistive control scheme and the corresponding haptic feedback. A control input generated by the user's intention was blended with assistive force to yield a guiding force, and the scaled guiding force was also transfered to the user's hand as haptic feedback. Lastly, we discussed the stability of our HRI system, and showed that our design approach satisfied the conditions to have guaranteed stability using a passivity-based analysis.

For the guiding force, haptic feedback, and visual feedback, we have not yet provided any explicit definition or mathematical expression. The definition and mathematical expression will be given for the guiding force and perceptual feedback in Chapter 5 and Chapter 6.

CHAPTER 4

MODELING USER'S TASK-PERFORMING CHARACTERISTICS

This chapter presents an approach to modeling a user's task-performing characteristics in a mobile robot steering task. Our motivation comes from the general problem of developing assistive interfaces for human-robot interaction (HRI) that guide low-performing users with a high-performing user's knowledge. In particular, the modeled task-performing characteristics can serve as design parameters that are used to customize the assistive HRI interface for the low-performing user. Our focus in this chapter is on introducing the methods to model the user's task-performing characteristics.

First, we briefly overview our approach to modeling the user's task-performing characteristics in which inverse optimal control (IOC) is used as a tool. Second, we concretely discuss a way to model a user's task-performing characteristics in a mobile robot steering task. Third, we describe the experimental setup and procedures to obtain user-demonstrated data from human subjects. Fourth, utilizing the obtained data sets, we infer the unknown parameter vector by solving an IOC problem. Then, the user's task-performing characteristics are expressed in terms of the balances of the inferred parameters, which allows us to investigate the relationship between the modeled task-performing characteristics and task-completion time. Lastly, we present a classification method which allows us to classify users relative to a high-performer whose task-performing characteristics are most similar.

4.1 Approach to Modeling User's Task-Performing Characteristics: Overview

For a decade, inverse optimal control (IOC) has been applied to a broad range of fields, e.g., learning a user's driving style [54], operating an autopilot system that mimics special maneuvers demonstrated by a human pilot [96], find-

ing an optimality principle in human walking [97,98], estimating a cost function for human arm movement [99], determining a user's driving style with continuous model [56], and predicting a probabilistic pointing target [100]. Through various applications, IOC has become a powerful tool for finding, e.g., a cost function for "flying well" or "driving hastily" [54]. Namely, when user-demonstrated data is given, IOC can be utilized in defining ambiguous characteristics, such as "driving well" or "steering well", with its capability to infer the unknown weighting vector for a set of known (pre-determined) basis functions [54, 101].

In modeling the task-performing characteristics, we use techniques from IOC, where known basis functions (expressed in terms of speed, steering, and proximities to inner/outer road boundary) are employed to design a cost function. Specifically, we model the task-performing characteristics by solving the IOC problem in order to infer how a user optimizes the balances of speed, steering, and proximities to inner/outer road boundary with observed user-demonstrations. To accomplish this, we begin by conducting experiments on human subjects to obtain the demonstrated data. The experimental setup consists of a display device and an input device, as the user (the subject) plays a video game. From the start to the end, the user is instructed to control a mobile robot under various road conditions. Afterward, we solve the IOC problem with the user-demonstrated data to infer the unknown parameter vector for each user.

We finally obtain the modeled user's task-performing characteristics in terms of the balances of the inferred parameters, which in turn designates the user's driving style. These balances of the inferred parameters will be referred to as *features*. Eventually, the features help us to investigate the relationship between the modeled task-performing characteristics and task-completion time. In general, the outcomes from this study will provide an answer to a class of questions – Are "certain type of driving styles" actually reflected on "paths" and "time" taken by users to reach a destination?

The rest of the chapter is organized as follows: In Section 4.2, we discuss how we model user's task-performing characteristics using the techniques from IOC. Section 4.3 describes the experimental setup and the task procedures of the human subject experiments that are employed to obtain user-demonstrated data. In Section 4.4, we present the modeled task-performing characteristics for all users, as well as their task-completion times. We dis-

cuss the relationship between the modeled task-performing characteristics and task-completion times in Section 4.4.2. Finally, we conclude this chapter with summary in Section 4.6.

4.2 Modeling User's Task-Performing Characteristics Using Techniques from Inverse Optimal Control

In this section, we describe how we adopt techniques from inverse optimal control (IOC) to model user's task-performing characteristics, and applied numerical method. Throughout the section, we take account of the specific application in which a user controls a mobile robot along various road shapes with a visual display.

4.2.1 IOC Formulation: Problem Statement

To model the user's task-performing characteristics in a steering task, we consider a cost function that represents how the user regulates speed, steering, and distances to road boundaries. As illustrated in Fig. 4.1, let $p_{\text{ib}} \in \gamma_{\text{ib}}$ and $p_{\text{ob}} \in \gamma_{\text{ob}}$ be the closest points from current mobile robot position $x_{1:2}$ (inspired by [102], we use MATLAB-like expression $x_{1:2} := [x_1, x_2]^T$) to inner and outer side boundaries respectively, and the robot orientation is x_3.

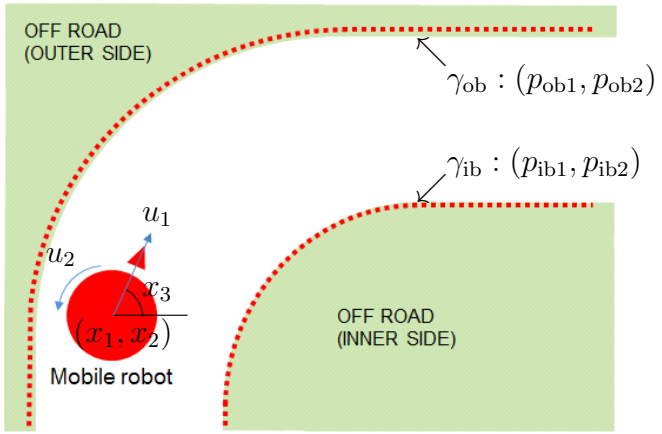


Figure 4.1: The mobile robot kinematics and the road boundaries at both sides.

Consider the following minimization problem:

$$\begin{aligned}\min_{u} \quad & \int_0^T u^T R u + (x_{1:2} - p_{\text{ib}})^T Q_{\text{ib}} (x_{1:2} - p_{\text{ib}}) \\ & \qquad + (x_{1:2} - p_{\text{ob}})^T Q_{\text{ob}} (x_{1:2} - p_{\text{ob}}) \; dt \\ \text{subject to} \quad & \dot{x}_1(t) = \cos(x_3(t)) u_1(t) \\ & \dot{x}_2(t) = \sin(x_3(t)) u_1(t) \\ & \dot{x}_3(t) = u_2(t) \\ & x(0) = x_{\text{start}} \\ & x(T) = x_{\text{goal}}\end{aligned} \tag{4.1}$$

where

$$R = \begin{bmatrix} c_v & 0 \\ 0 & c_\omega \end{bmatrix}, \quad Q_{\text{ib}} = \begin{bmatrix} c_{\text{ib}} & 0 \\ 0 & c_{\text{ib}} \end{bmatrix}, \quad \text{and} \quad Q_{\text{ob}} = \begin{bmatrix} c_{\text{ob}} & 0 \\ 0 & c_{\text{ob}} \end{bmatrix} \tag{4.2}$$

The subscripted parameters c_v, c_ω, c_{ib}, and c_{ob} represent that the parameters are related to speed, steering, proximity to the inner side, and proximity to the outer side respectively.

Now, suppose that we have user-demonstrated data, which we assume to be the locally optimal solution of (4.1) as a set of tuples (x^*, u^*). Then, the IOC problem is [53, 56, 60]:

Given: data demonstrated by a user, (x^*, u^*), which is assumed to be locally optimal solution of (4.1),

Infer: the user's cost function by inferring the unknown parameters, c_v, c_ω, c_{ib}, and c_{ob}, in matrices R, Q_{ib}, and Q_{ob} .

4.2.2 Discrete Time Formulation and Applied Numerical Method

Let t_s and k be a sampling time and corresponding time step respectively. We formulate the minimization problem of (4.1) in discrete time [28, 59]

$$\begin{aligned}
\min_{u} \Big[\sum_{k=0}^{N-1} & u_k^T R u_k + (x_{1:2,k} - p_{\text{ib},k})^T Q_{\text{ib}} (x_{1:2,k} - p_{\text{ib},k}) \\
& + (x_{1:2,k} - p_{\text{ob},k})^T Q_{\text{ob}} (x_{1:2,k} - p_{\text{ob},k}) \Big] \\
\text{subject to} \quad & \left.\begin{aligned}
x_{1,k+1} &= x_{1,k} + t_s \cos(x_{3,k}) u_{1,k} \\
x_{2,k+1} &= x_{2,k} + t_s \sin(x_{3,k}) u_{1,k} \\
x_{3,k+1} &= x_{3,k} + t_s u_{2,k}
\end{aligned}\right\} (\blacktriangle) \\
& x_0 = x_{\text{start}} \\
& x_N = x_{\text{goal}}
\end{aligned} \tag{4.3}$$

where $u = (u_0, \cdots, u_{N-1})$, R, Q_{ib}, Q_{ob}, p_{ib}, and p_{ob} are the same as before.

Now, to solve the IOC problem of (4.3), we apply the method developed in [60]. Here, we recapitulate that method, as it applied to our specific cost function. First, we rewrite the above kinematic equation ($\blacktriangle$) for clarity as

$$x_{k+1} = x_k + t_s f(x_k, u_k)$$

for $k = 0, \cdots, N-1$. A discrete time Hamiltonian H_k can be defined [51, 61]

$$H_k(x_k, u_k, \lambda_k) = L_k(x_k, u_k) + \lambda_k^T f(x_k, u_k) \tag{4.4}$$

where L_k and $\lambda_k \in \mathbb{R}^{3\times 1}$ represent a discrete Lagrangian and a costate vector at time step k, respectively. From (4.3), we have

$$\begin{aligned}
H_k = \Big[& u_k^T R u_k + (x_{1:2,k} - p_{\text{ib},k})^T Q_{\text{ib}} (x_{1:2,k} - p_{\text{ib},k}) \\
& + (x_{1:2,k} - p_{\text{ob},k})^T Q_{\text{ob}} (x_{1:2,k} - p_{\text{ob},k}) \Big] + \lambda_k^T f(x_k, u_k)
\end{aligned} \tag{4.5}$$

By the Pontryagin's Maximum principle, we have a costate equation [62, 63,

103], thus

$$\begin{aligned}\frac{\partial H_k}{\partial x_k} &= \begin{bmatrix} 2(x_{1,k} - p_{\text{ib1},k})c_{\text{ib}} + 2(x_{1,k} - p_{\text{ob1},k})c_{\text{ob}} \\ 2(x_{2,k} - p_{\text{ib2},k})c_{\text{ib}} + 2(x_{2,k} - p_{\text{ob2},k})c_{\text{ob}} \\ -\lambda_{1,k}\sin(x_{3,k})u_{1,k} + \lambda_{2,k}\cos(x_{3,k})u_{1,k} \end{bmatrix} \\ &= -\frac{\lambda_{k+1} - \lambda_k}{t_s}\end{aligned} \tag{4.6}$$

Rearranging (4.6) yields

$$\begin{bmatrix} 2t_s(x_{1,k} - p_{\text{ib1},k})c_{\text{ib}} + 2t_s(x_{1,k} - p_{\text{ob1},k})c_{\text{ob}} \\ 2t_s(x_{2,k} - p_{\text{ib2},k})c_{\text{ib}} + 2t_s(x_{2,k} - p_{\text{ob2},k})c_{\text{ob}} \\ t_s(-\lambda_{1,k}\sin(x_{3,k})u_{1,k} + \lambda_{2,k}\cos(x_{3,k})u_{1,k}) \end{bmatrix} + \begin{bmatrix} \lambda_{1,k+1} - \lambda_{1,k} \\ \lambda_{2,k+1} - \lambda_{2,k} \\ \lambda_{3,k+1} - \lambda_{3,k} \end{bmatrix} = \mathbf{0}^T \tag{4.7}$$

By a necessary condition for optimality [62, 63, 103], we have

$$\begin{aligned}\frac{\partial H_k}{\partial u_k} &= \begin{bmatrix} 2u_{1,k}c_v + (\lambda_{1,k}\cos(x_{3,k}) + \lambda_{2,k}\sin(x_{3,k}) \\ 2u_{2,k}c_\omega + \lambda_{3,k} \end{bmatrix} \\ &= \mathbf{0}^T\end{aligned} \tag{4.8}$$

Finally, if we define a vector z_k as

$$z_k = [c_v \; c_\omega \; c_{\text{ib}} \; c_{\text{ib}} \; c_{\text{ob}} \; c_{\text{ob}} \; \lambda_{1,k+1} \; \lambda_{2,k+1} \; \lambda_{3,k+1} \; \lambda_{1,k} \; \lambda_{2,k} \; \lambda_{3,k}]^T \tag{4.9}$$

then (4.7) and (4.8) can be combined and rewritten in the form of a system of equations, denoted by r_k, thus

$$\begin{aligned}r_k &= \begin{bmatrix} A_{11,k} & A_{12,k} & A_{13,k} & A_{14,k} \\ A_{21,k} & A_{22,k} & A_{23,k} & A_{24,k} \end{bmatrix} z_k \\ &= A_k z_k\end{aligned} \tag{4.10}$$

where the submatrices in the first row are

$$A_{11,k} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{4.11}$$

$$A_{12,k} = \begin{bmatrix} 2t_s(x_{1,k} - p_{\text{ib1},k}) & 0 & 0 \\ 0 & 2t_s(x_{2,k} - p_{\text{ib2},k}) & 0 \\ 2t_s(x_{1,k} - p_{\text{ob1},k}) & 0 & 0 \\ 0 & 2t_s(x_{2,k} - p_{\text{ob2},k}) & 0 \end{bmatrix}^T$$

$$A_{13,k} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A_{14,k} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ -t_s u_{1,k} \sin(x_{3,k}) & t_s u_{1,k} \cos(x_{3,k}) & -1 \end{bmatrix}$$

and in the second row

$$\begin{aligned} A_{21,k} &= \begin{bmatrix} 2u_{1,k} & 0 \\ 0 & 2u_{2,k} \end{bmatrix} \\ A_{22,k} &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}^T \\ A_{23,k} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ A_{24,k} &= \begin{bmatrix} \cos(x_{3,k}) & \sin(x_{3,k}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{4.12}$$

Hence, with the given (x^*,u^*), the problem of inferring the unknown parameters, $c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}$ (which are involved in z), becomes identical to solving the following least square problem

$$\min_{c,\lambda} \sum_{k=0}^{N-1} \|r_k^*\|^2 \;=\; \min_{c,\lambda} \sum_{k=0}^{N-1} \|A_k^* z_k\|^2 \tag{4.13}$$

where $c = [c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}]$ and A_k^* represents the matrix A_k being evaluated at (x_k^*,u_k^*).

4.3 Experiment with Human Subjects to Obtain User Demonstration

In this section, we describe the experimental setup and procedure of the human subject experiment. The objective of the experiment is to obtain the user-demonstrated data that will be utilized as (x^*,u^*) for the IOC problem.

4.3.1 Experimental Setup

A user (a human subject) was provided with an input device and a display, and instructed to complete a given task. We used a Novint FALCON haptic joystick controller as the input device. A typical 17-inch monitor was used as a display device. Fig. 4.2 shows our simulator interface provided to the user. The interface window size was 15.89cm×28.78cm (height×width) approximately in 1280×1024 display resolution, and provided a global view and a local view of the environment for the steering task. The other area of the monitor was filled with uniform gray color to prevent distractions caused by background contents.

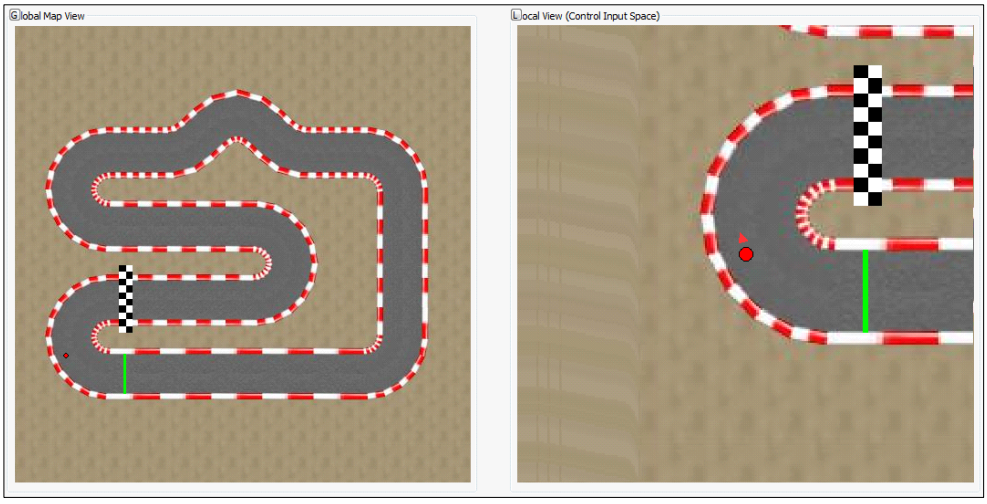


Figure 4.2: Simulator interface: a global view (left window) and a local working field of view (right window). The green (solid) line and the checkboard pattern represent the starting line and the goal, respectively. The track image can be found at http://supertuxkart.sourceforge.net.

4.3.2 Task Procedure

The experiment was divided into a practice stage and a testing stage. During the practice stage, the user was asked to familiarize him/herself with the user interface by driving a mobile robot around either in- or out-side of the road boundaries. However, we instructed the user that it would be regarded as a task failure to drive the mobile robot outside of the road boundary during the testing stage. The testing stage started when the user verbally expressed that he/she felt familiar with the apparatus and confident with the task.

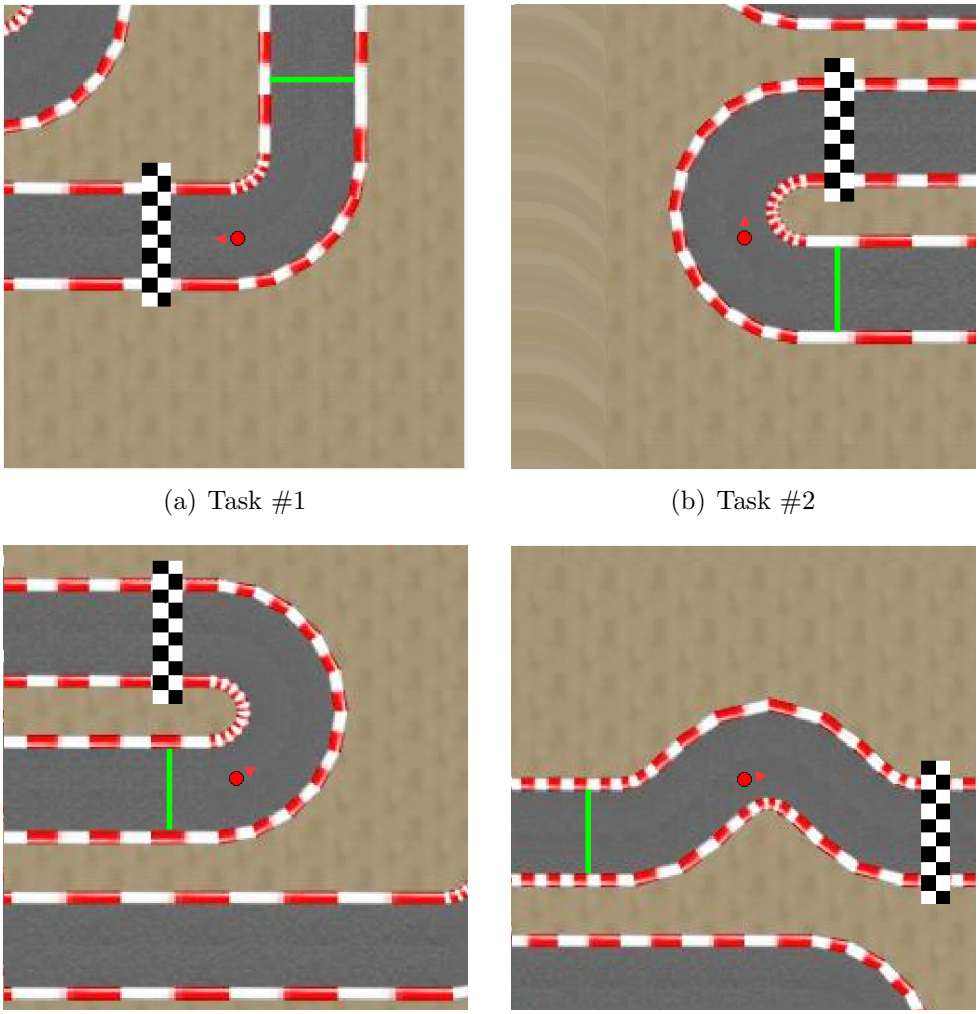
During the testing stage, the user was instructed to drive the mobile robot from the starting location to the goal. The simulator provided 4 different task-locations wherein roads had different curvatures and turning angles. Each task was performed 5 times (trials); then the user proceeded to the next task. The user data of which the task-completion time was a median among the 5 trials was regarded as (x^*, u^*) for a given task. There was a 3-second interval between every trial. While the user was performing the given task, a program read the position and the orientation of the mobile robot every 20 ms, and stored the position, orientation, two control inputs, task number, trial number, and task result (success or failure). Fig. 4.3 shows the four tasks that were given to the users to obtain the user-demonstrated trajectories.

4.4 Inferred Unknown Parameter and Defining Features

4.4.1 Inferred Unknown Parameter Vector by solving IOC

Let $D_j : \{(x_k^*, u_k^*)\}_{k=1}^{N_j}$ be a set of user-demonstrated data for task $\#j$, where N_j is task-completion time for $j = 1, \cdots, 4$ (corresponding to four different task-locations). We solve the IOC problem of multiple demonstrations by applying the method introduced in Section 4.2.2.[1]

[1] Also, see [60] for IOC with multiple user-demonstrated trajectories.



(a) Task #1 (b) Task #2

(c) Task #3 (d) Task #4

Figure 4.3: Four given tasks to obtain the sets of user-demonstrated data.

Recalling the cost function in (4.3), what we want to capture from the user-demonstrated data is how a user's control regulates speed, steering, and proximities to inner/outer boundary. Indeed, the balance of these regulations is directly related to a curvature and a route of the user-demonstrated trajectory, which in turn determines a task-completion time. Table 4.1 shows the inferred parameter vector, $[c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}]$ by solving the IOC problem for 21 user-demonstrations.

Table 4.1: Inferred parameter vector by solving IOC and the defined feature q.

USER#	$[c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}]$	q	$\sum N_j$
USER1	$[0.3646, 0.1000, 0.8486, 1.0000]$	$(3.6460, 0.8486, 0.8236)_{\text{USER1}}$	894
USER2	$[0.1000, 0.1000, 0.8873, 1.0000]$	$(1.0000, 0.8873, 0.3008)_{\text{USER2}}$	701
USER3	$[0.1000, 0.1000, 1.0939, 1.0000]$	$(1.0000, 1.0939, 0.2707)_{\text{USER3}}$	674
USER4	$[0.1000, 0.1000, 0.6005, 1.0000]$	$(1.0000, 0.6005, 0.3769)_{\text{USER4}}$	733
USER5	$[0.1000, 0.3377, 0.6744, 1.0000]$	$(0.2961, 0.6744, 0.8744)_{\text{USER5}}$	766
USER6	$[0.5274, 0.1000, 0.6934, 1.0000]$	$(5.2740, 0.6934, 1.3109)_{\text{USER6}}$	962
USER7	$[0.1000, 0.8381, 0.9002, 1.0000]$	$(0.1193, 0.9002, 1.7817)_{\text{USER7}}$	2268
USER8	$[0.1000, 0.1743, 0.6567, 1.0000]$	$(0.5737, 0.6567, 0.5069)_{\text{USER8}}$	1549
USER9	$[0.2874, 0.2935, 0.7147, 1.0000]$	$(0.9792, 0.7147, 0.9855)_{\text{USER9}}$	722
USER10	$[0.1000, 0.1047, 1.3592, 1.0000]$	$(0.9551, 1.3592, 0.2513)_{\text{USER10}}$	613
USER11	$[0.1000, 0.1000, 0.8632, 1.0000]$	$(1.0000, 0.8632, 0.3053)_{\text{USER11}}$	746
USER12	$[0.1000, 0.2355, 0.7384, 1.0000]$	$(0.4246, 0.7384, 0.6023)_{\text{USER12}}$	860
USER13	$[0.1779, 0.1874, 0.8443, 1.0000]$	$(0.9493, 0.8443, 0.5644)_{\text{USER13}}$	643
USER14	$[0.5450, 0.1000, 0.9058, 1.0000]$	$(5.4500, 0.9058, 1.1658)_{\text{USER14}}$	670
USER15	$[0.1000, 0.1000, 0.8777, 1.0000]$	$(1.0000, 0.8777, 0.3025)_{\text{USER15}}$	1167
USER16	$[0.1000, 0.1000, 0.7618, 1.0000]$	$(1.0000, 0.7618, 0.3271)_{\text{USER16}}$	833
USER17	$[0.1000, 0.2608, 0.5586, 1.0000]$	$(0.3834, 0.5586, 0.7793)_{\text{USER17}}$	707
USER18	$[0.1000, 0.1678, 0.6455, 1.0000]$	$(0.5959, 0.6455, 0.4980)_{\text{USER18}}$	767
USER19	$[0.1000, 0.6294, 0.7595, 1.0000]$	$(0.1589, 0.7595, 1.4764)_{\text{USER19}}$	1157
USER20	$[0.3428, 0.1000, 1.2853, 1.0000]$	$(3.4280, 1.2853, 0.6349)_{\text{USER20}}$	1212
USER21	$[0.1000, 0.1000, 0.4943, 1.0000]$	$(1.0000, 0.4943, 0.4275)_{\text{USER21}}$	740

$*$ c_{ob} is normalized to 1. As we solve (4.13), the IOC solver returns trivial solution $[0, 0, 0, 0]$ unless we normalize one of $[c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}]$.

4.4.2 Discussion of the Inferred Parameters and the Task-Completion Time, and Defining Three Features

We applied various techniques from regression analysis, clustering analysis, and curve fitting to find a direct relationship in the form of continuous mapping function between the inferred parameters and task-completion time in Table 4.1. It was not so straightforward, however, to find a precise mapping function between those two.

Instead, we define three values with the inferred parameters as follows:

- c_v/c_ω : ratio of linear to angular velocity
- $c_{\text{ib}}/c_{\text{ob}}$: ratio of proximity to inner/outer boundary
- $\|RQ_{\text{ib}}^{-1}\|_F + \|RQ_{\text{ob}}^{-1}\|_F$: ratio of control effort to boundary collision avoidance

In the third value, $\|\cdot\|_F$ represents the Frobenius norm, hence

$$\begin{aligned}\|RQ_{\text{ib}}^{-1}\|_F + \|RQ_{\text{ob}}^{-1}\|_F &= \sqrt{\left(\frac{c_v}{c_{\text{ib}}}\right)^2 + \left(\frac{c_\omega}{c_{\text{ib}}}\right)^2} + \sqrt{\left(\frac{c_v}{c_{\text{ob}}}\right)^2 + \left(\frac{c_\omega}{c_{\text{ob}}}\right)^2} \\ &= \left(1 + \frac{1}{c_{\text{ib}}}\right)\sqrt{{c_v}^2 + {c_\omega}^2}\end{aligned} \tag{4.14}$$

We refer to these three values as the *features* of user's *task-performing characteristics*.[2] By defining the features, a user can be associated to a sample point q in *feature space* $\mathbb{R}^3$

$$q = \left(\frac{c_v}{c_\omega}\ , \frac{c_{\text{ib}}}{c_{\text{ob}}}\ , \|RQ_{\text{ib}}^{-1}\|_F + \|RQ_{\text{ob}}^{-1}\|_F\right)_{\text{USER\#}} \tag{4.15}$$

where the subscript at the end, USER#, represents a user associated with q. The defined features are shown in Table 4.1 as well.

We investigated the distribution of samples in this feature space, and found that the samples were grouped around high-performing users, e.g., USER10 or USER13. Therefore, we will classify the users by regarding the high-performing users as centroids for a cluster wherein the samples have the

[2]In the following chapters, the terms "feature" and "task-performing characteristics" will be used interchangeably.

certain type of task-performing characteristics. We will discuss the classification method in detail in Section 4.5.

4.5 Classify the Users Based on the Closest High-Performers in Feature Space

4.5.1 A User as a Sample Point in Feature Space

Table 4.2 shows q_i for all users who were shown in Table 4.1. Here, we have sorted the users based on their task-completion times. Accordingly, the subscript i can be regarded as either the rank of user in task-completion time or the row index. By investigating Table 4.2, we see that it is possible for users to have approximately the same task-completion time even if their task-performing characteristics are located far apart in the feature space. For instance, both USER14 and USER3 are high-performing users (which will be referred to as "high-performers" shortly) who completed the task within 13.5 second approximately. However, their task-performing characteristics, q_3 and q_4, are unique without sharing any resemblance.

Recalling the main goal of our developed interface – to guide a low-performing user with a high-performer's knowledge under a newly given task – we classify the users based on the closest high-performers by regarding them as centroids in feature space. We call this classification method *Closest-High-Performer-Based (CHPB) classification*. Algorithm 4.1 describes the CHPB classification. Note that we designate USER10, USER13, USER14, and USER3 (associated with q_1, q_2, q_3, and q_4) as four high-performers, and use l_1-norm as a (distance) metric.

Fig. 4.4 shows the result of the CHPB classification. The samples in the same class are connected by a solid line. Table 4.3 depicts the assigned user class, as well as the corresponding virtual fixture parameters which will be discussed in Section 5.3.

Table 4.2: The users as sample points in feature space (sorted by task-completion time $T = \sum N_j \times \frac{1}{50}$ [sec]).

USER#	$[c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}]$	q_i	T
USER10	$[0.1000, 0.1047, 1.3592, 1.0000]$	$q_1 = (0.9551, 1.3592, 0.2513)_{\text{USER10}}$	12.26
USER13	$[0.1779, 0.1874, 0.8443, 1.0000]$	$q_2 = (0.9493, 0.8443, 0.5644)_{\text{USER13}}$	12.86
USER14	$[0.5450, 0.1000, 0.9058, 1.0000]$	$q_3 = (5.4500, 0.9058, 1.1658)_{\text{USER14}}$	13.40
USER3	$[0.1000, 0.1000, 1.0939, 1.0000]$	$q_4 = (1.0000, 1.0939, 0.2707)_{\text{USER3}}$	13.48
USER2	$[0.1000, 0.1000, 0.8873, 1.0000]$	$q_5 = (1.0000, 0.8873, 0.3008)_{\text{USER2}}$	14.02
USER17	$[0.1000, 0.2608, 0.5586, 1.0000]$	$q_6 = (0.3834, 0.5586, 0.7793)_{\text{USER17}}$	14.14
USER9	$[0.2874, 0.2935, 0.7147, 1.0000]$	$q_7 = (0.9792, 0.7147, 0.9855)_{\text{USER9}}$	14.44
USER4	$[0.1000, 0.1000, 0.6005, 1.0000]$	$q_8 = (1.0000, 0.6005, 0.3769)_{\text{USER4}}$	14.66
USER21	$[0.1000, 0.1000, 0.4943, 1.0000]$	$q_9 = (1.0000, 0.4943, 0.4275)_{\text{USER21}}$	14.80
USER11	$[0.1000, 0.1000, 0.8632, 1.0000]$	$q_{10} = (1.0000, 0.8632, 0.3053)_{\text{USER11}}$	14.92
USER5	$[0.1000, 0.3377, 0.6744, 1.0000]$	$q_{11} = (0.2961, 0.6744, 0.8744)_{\text{USER5}}$	15.32
USER18	$[0.1000, 0.1678, 0.6455, 1.0000]$	$q_{12} = (0.5959, 0.6455, 0.4980)_{\text{USER18}}$	15.34
USER16	$[0.1000, 0.1000, 0.7618, 1.0000]$	$q_{13} = (1.0000, 0.7618, 0.3271)_{\text{USER16}}$	16.66
USER12	$[0.1000, 0.2355, 0.7384, 1.0000]$	$q_{14} = (0.4246, 0.7384, 0.6023)_{\text{USER12}}$	17.20
USER1	$[0.3646, 0.1000, 0.8486, 1.0000]$	$q_{15} = (3.6460, 0.8486, 0.8236)_{\text{USER1}}$	17.88
USER6	$[0.5274, 0.1000, 0.6934, 1.0000]$	$q_{16} = (5.2740, 0.6934, 1.3109)_{\text{USER6}}$	19.24
USER19	$[0.1000, 0.6294, 0.7595, 1.0000]$	$q_{17} = (0.1589, 0.7595, 1.4764)_{\text{USER19}}$	23.14
USER15	$[0.1000, 0.1000, 0.8777, 1.0000]$	$q_{18} = (1.0000, 0.8777, 0.3025)_{\text{USER15}}$	23.34
USER20	$[0.3428, 0.1000, 1.2853, 1.0000]$	$q_{19} = (3.4280, 1.2853, 0.6349)_{\text{USER20}}$	24.24
USER8	$[0.1000, 0.1743, 0.6567, 1.0000]$	$q_{20} = (0.5737, 0.6567, 0.5069)_{\text{USER8}}$	30.98
USER7	$[0.1000, 0.8381, 0.9002, 1.0000]$	$q_{21} = (0.1193, 0.9002, 1.7817)_{\text{USER7}}$	45.36

$*$ The subscript i is the row index corresponding to user's rank in task-completion time.

Algorithm 4.1: CHPB CLASSIFICATION.

Input: $\{[c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}], \text{task-completion time}\}$ of N-users.
Output: $\{q_i,\ \text{CLASS}_i\}$ for the sorted N-users.

1. Sort N-user based on task-completion time.
2. Express a user in i^{th}-row as a sample point q_i in feature space $\mathbb{R}^3$

$$q_i = \left(c_v/c_\omega, c_{\text{ib}}/c_{\text{ob}}, \|RQ_{\text{ib}}^{-1}\|_F + \|RQ_{\text{ob}}^{-1}\|_F\right)_{\text{USER\#}}$$

3. Choose M-highest-performer for each class:$\{\ q_1, \cdots, q_M\}$.
4. Assign user's class according to q_i

for $i \leftarrow M+1$ **to** N **do**

$$\text{CLASS}_i = \arg \min_{j \in \{1,\cdots,\text{M}\}} \|q_i - q_j\|_1$$

end

Table 4.3: The assigned user's class by the CHPB classification. A USER with $*$ is the high-performer in a class.

CLASS	USER#	$\|q_i - q_*\|_1$	$\|T_i - T^*\|$
1	$*$USER10	–	–
	USER20	2.9306	11.98
2	$*$USER13	–	–
	USER17	1.0663	1.28
	USER9	0.5805	1.58
	USER4	0.4820	1.80
	USER21	0.5377	1.94
	USER5	1.1328	2.46
	USER18	0.6187	2.48
	USER16	0.3706	3.80
	USER12	0.6684	4.34
	USER19	1.7870	10.28
	USER8	0.6207	18.12
	USER7	2.1030	22.22
3	$*$USER14	–	–
	USER1	2.2039	4.48
	USER6	0.5339	5.84
4	$*$USER3	–	–
	USER2	0.2367	0.54
	USER11	0.2653	1.44
	USER15	0.2480	9.86

4.6 Summary

In this chapter, we considered the problems associated with modeling a user's task-performing characteristics in a mobile robot steering task. We performed human subject experiments, and inferred the unknown parameter vectors by solving an inverse optimal control problem. Then, we modeled the user's task-performing characteristics in terms of the balance of the inferred parameters, which were referred to as features. By designating a user as a sample in feature space, we observed that the samples were placed around that of the high-performing users. This observation allowed us to get an insight into how to classify the users of certain task-performing characteristics. Therefore, we classified the users based on the high-performers whose task-performing strategies are closest. As a result, the users were classified into the clusters wherein the user's task-performing characteristics are akin to each other.

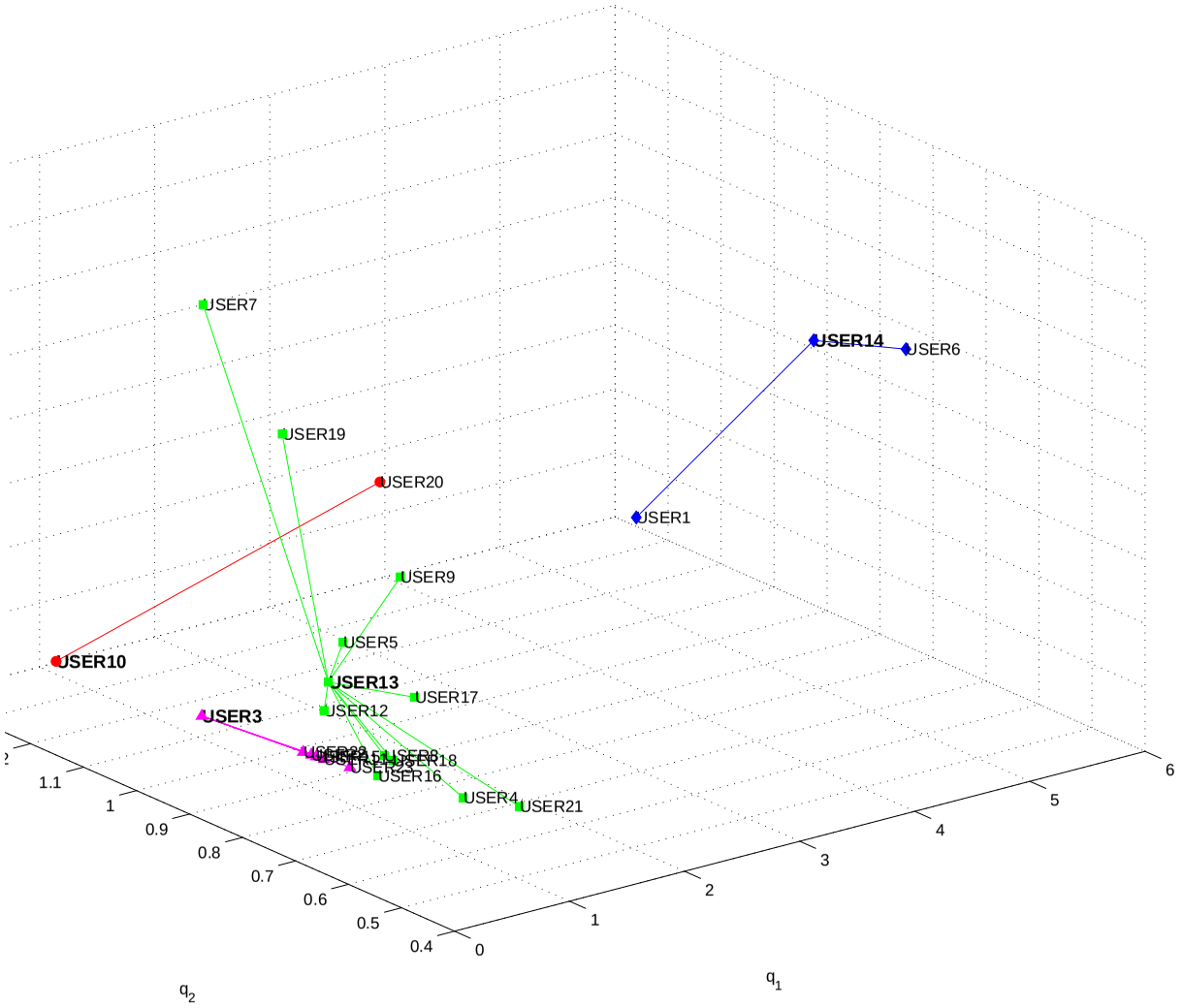


Figure 4.4: The CHPB classification result plot. The four high-performers are represented by bold face.

CHAPTER 5

CUSTOMIZING THE VIRTUAL FIXTURE AND HAPTIC FEEDBACK

In Chapter 4, we discussed our approach to modeling the user's task-performing characteristics and defined the three features representing the user's task-performing characteristics. The main purpose of this chapter is to present our approach to utilizing the defined features to set the proper level of assistance by customizing virtual fixture parameters for a user with certain characteristics. We begin by introducing the concept of the level of assistance. Second, we define assisted control and a guiding force under virtual fixturing. Third, we discuss our approach to customizing the virtual fixture, which eventually results in the customized haptic feedback. We conclude this chapter with a summary.

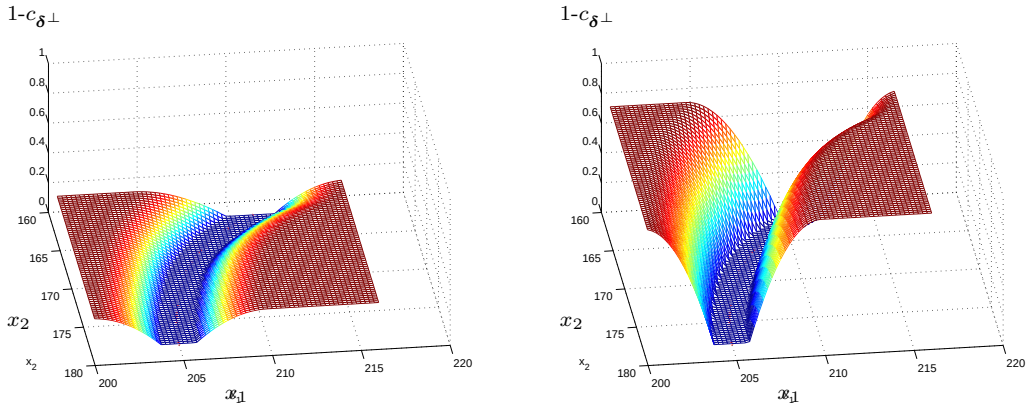
5.1 The Level of Assistance: Soft and Firm Assistance

In Chapter 2, we introduced that a virtual fixture can be defined by *attenuating admittance* [14], denoted by $c_{\boldsymbol{\delta}^\perp}$, as a monotonically non-increasing function with respect to deviation $\|\mathbf{e}\|$ from a spine

$$c_{\boldsymbol{\delta}^\perp}(\|\mathbf{e}\|) = \begin{cases} \underline{c}_{\boldsymbol{\delta}^\perp}, & \text{if } \|\mathbf{e}\| > \frac{d}{2} \\ \underline{c}_{\boldsymbol{\delta}^\perp} + \left[\frac{d/2-\|\mathbf{e}\|}{\nu}\right]^n (1-\underline{c}_{\boldsymbol{\delta}^\perp}), & \text{if } \frac{d}{2}-\nu < \|\mathbf{e}\| \leq \frac{d}{2} \\ 1, & \text{if } \|\mathbf{e}\| \leq \frac{d}{2}-\nu \end{cases}$$

where the width of virtual fixture was specified by d, correction zone by ν, and the default (the lower limit) attenuation by $\underline{c}_{\boldsymbol{\delta}^\perp}$. As the level of attenuation mainly depends on $\underline{c}_{\boldsymbol{\delta}^\perp}$, we will present our approach to customizing $\underline{c}_{\boldsymbol{\delta}^\perp}$ to assist a user based on his/her task-performing characteristics and corresponding task-performances in Section 5.3.

We refer to the two cases of $\underline{c}_{\delta^\perp}= 0.1$ and $\underline{c}_{\delta^\perp}= 0.9$ as *firm assistance* and *soft assistance* respectively [3, 4]. Accordingly, the more skilled the user, the



(a) soft assistance $\underline{c}_{\delta^\perp} = 0.9$ (b) firm assistance $\underline{c}_{\boldsymbol{\delta}^\perp} = 0.1$

Figure 5.1: Soft and firm virtual fixturings in terms of the amount of attenuation, $1 - c_{\boldsymbol{\delta}^\perp}$, with respect to $\|\mathbf{e}\|$ along the same spine.

softer assistance applied to the control input. Oppositely, the less skilled the user, the firmer assistance applied. The effect of the virtual fixture can be intuitively seen by plotting "the amount of attenuation" along the spine. Fig. 5.1 illustrates virtual fixturings for the cases of soft assistance and firm assistance in terms of the amount of attenuation, $1 - \underline{c}_{\delta^\perp}$, with respect to $\|\mathbf{e}\|$ along the same spine.

5.2 Assisted Control Force and Guiding Force under Virtual Fixturing

Let $\boldsymbol{f}$ be a control input to the controlled object as shown in Fig. 5.2. We start deriving an *assisted control force*, denoted by $\boldsymbol{f}_{\text{assisted}}$, by decomposing $\boldsymbol{f}$ into two components. First, we define a projection operator $P_{\boldsymbol{\delta}}$ that performs a vector projection $\boldsymbol{f}$ onto $\boldsymbol{\delta}$

$$P_{\boldsymbol{\delta}} = \frac{1}{\boldsymbol{\delta}^T \boldsymbol{\delta}} \boldsymbol{\delta}\boldsymbol{\delta}^T \tag{5.1}$$

Then, we can express a force input $\boldsymbol{f}$ by decomposing it into the preferred and the non-preferred direction using $P_{\boldsymbol{\delta}}$, thus

$$\begin{aligned} \boldsymbol{f} &= P_{\boldsymbol{\delta}} \boldsymbol{f} + (I - P_{\boldsymbol{\delta}}) \boldsymbol{f} \\ &= \boldsymbol{f}_{\boldsymbol{\delta}} + \boldsymbol{f}_{\boldsymbol{\delta}^\perp} \end{aligned} \tag{5.2}$$

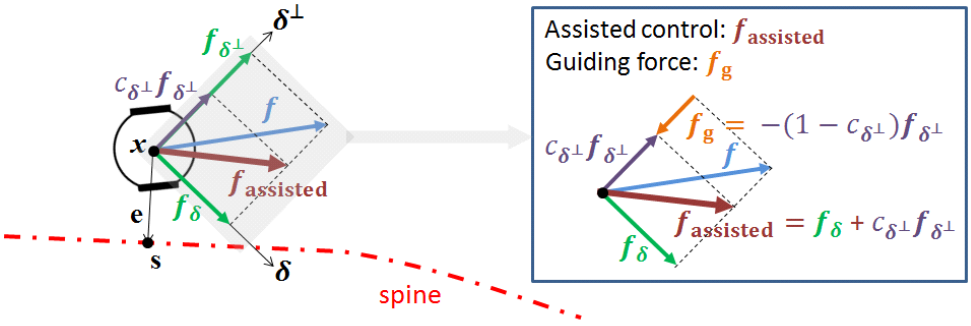


Figure 5.2: Assisted control force $\boldsymbol{f}_{\text{assisted}}$ and guiding force $\boldsymbol{f}_{\text{g}}$.

Next, the role of $c_{\boldsymbol{\delta}^{\perp}}$ is to attenuate the non-preferred direction component. Therefore, the assisted control force $\boldsymbol{f}_{\text{assisted}}$ can be expressed as

$$\boldsymbol{f}_{\text{assisted}} = \boldsymbol{f}_{\boldsymbol{\delta}} + c_{\boldsymbol{\delta}^{\perp}} \boldsymbol{f}_{\boldsymbol{\delta}^{\perp}} \tag{5.3}$$

The assisted control force (the attenuated force input) becomes an actual force input to the controlled object, e.g., an omnidirectional mass-damper system in Section 3.2.2.

This attenuation effect can be interpreted as a resulting guidance of a virtual fixture as follows. By algebraic manipulation, we can rewrite (5.3) in terms of the original term and the suppressive term

$$\begin{aligned} \boldsymbol{f}_{\text{assisted}} &= \boldsymbol{f}_{\boldsymbol{\delta}} + c_{\boldsymbol{\delta}^{\perp}} \boldsymbol{f}_{\boldsymbol{\delta}^{\perp}} \\ &= \boldsymbol{f}_{\boldsymbol{\delta}} + \boldsymbol{f}_{\boldsymbol{\delta}^{\perp}} - (1 - c_{\boldsymbol{\delta}^{\perp}}) \boldsymbol{f}_{\boldsymbol{\delta}^{\perp}} \\ &= \underbrace{\boldsymbol{f}}_{\text{original term}} + \underbrace{[-(1 - c_{\boldsymbol{\delta}^{\perp}}) \boldsymbol{f}_{\boldsymbol{\delta}^{\perp}}]}_{\text{suppressive term}} \\ &= \boldsymbol{f} + \boldsymbol{f}_{\text{g}} \end{aligned} \tag{5.4}$$

Hence, the guiding force $\boldsymbol{f}_{\text{g}}$ is equivalent to the suppressive term which works orthogonally to the preferred direction. Furthermore, the guiding force can be transferred as force feedback $\boldsymbol{f}_{\text{haptic}}$ to the user's hand, thus

$$\boldsymbol{f}_{\text{haptic}} = k_g \boldsymbol{f}_{\text{g}} \tag{5.5}$$

where k_g is a positive scaling constant. We note that the direction of $\boldsymbol{f}_{\text{g}}$ is opposite to $\boldsymbol{f}_{\boldsymbol{\delta}^{\perp}}$, and the magnitude is proportional to the amount of

attenuation $1 - c_{\boldsymbol{\delta}^\perp}$ as illustrated in Fig. 5.2.

5.3 Customizing a Virtual Fixture

For newly given tasks, we set a virtual fixture for a specific user by the following procedure. We note that the procedure uses data such as that presented in Table 4.2 and Table 4.3 of Section 4.5.

Choose a spine: γ

We generate a spine for each of the M-classes constructed by the CHPB classifier. Specifically, M-spines (denoted by $\gamma_1, \cdots, \gamma_M$) are generated by solving (forward/direct) optimal control with the inferred parameters of the M-high-performers whose feature were designated for the CHPB classification in Section 4.5. Then, we choose the spine γ_i for q_i among $\gamma_1, \cdots, \gamma_M$ according to the assigned class in Table 4.3.

Determine a width: d

The ratio of inner/outer boundary preference is a good cue to determine the width d of virtual fixture. Let $q_{2,i}$ and $q_{2,*}$ be the user's and the high-performer's second feature in the same class, respectively. To assist the user with high-performer's boundary preference, we define d as

$$d = W\Big[1 + \beta\left(q_{2,i} - q_{2,*}\right)^2\Big]^{-1} \tag{5.6}$$

where W is a road width, and β is a positive constant that the developer can choose. For instance, we set $W = 50$ (pixels) and $\beta = 1.0$ for our developed system.

Set the softness/firmness of assistance: $\underline{c}_{\boldsymbol{\delta}^\perp}$

The default attenuation, $\underline{c}_{\boldsymbol{\delta}^\perp}$, is determined by the user's task-completion time. Let c_{soft} and c_{firm} be the attenuating admittance values for "soft" and "firm" assistance, e.g., $c_{\text{soft}} = 0.9$ and $c_{\text{firm}} = 0.1$, respectively. We define $\underline{c}_{\boldsymbol{\delta}^\perp}$

Algorithm 5.1: VIRTUAL FIXTURE CUSTOMIZATION.

Input: $\{(x, u)\}$ of N-users
Output: $\{\gamma_i,\ d_i,\ \underline{c}_{\boldsymbol{\delta}^\perp,i}\}$ for the sorted N-users

1. Infer the unknown parameter vector $[c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}]$ in cost function

   $[c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}] \leftarrow \text{SOLVEINVERSEOPTIMALCONTROL}((x, u))$

2. Sort N-users based on T, and obtain q_i and CLASS_i

   $\{q_i, \text{CLASS}_i\} \leftarrow \text{HPBCLASSIFICATION}([c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}], T)$

3. Generate M-spines by high-performers' parameters

   $\{\gamma_1, \cdots, \gamma_M\} \leftarrow \text{SOLVEFOWARDOPTIMALCONTROL}([c_v, c_\omega, c_{\text{ib}}, c_{\text{ob}}]_*)$

4. Customize virtual fixture parameters for a user associated with q_i

for $i \leftarrow 1$ **to** N **do**

$\gamma_i \leftarrow \gamma_{\text{CLASS}_i}$
$d_i \leftarrow W\left[1 + \beta\left(q_{2,i} - q_{2,*}\right)^2\right]^{-1}$
$\underline{c}_{\boldsymbol{\delta}^\perp,i} \leftarrow -\frac{(c_{\text{soft}} - c_{\text{firm}})}{T_{\text{thresfirm}}}\ T_i + c_{\text{soft}}$

end

as

$$\underline{c}_{\boldsymbol{\delta}^\perp}(T_i) = -\frac{(c_{\text{soft}} - c_{\text{firm}})}{T_{\text{thresfirm}}}\ T_i\ +\ c_{\text{soft}} \tag{5.7}$$

where T_i is the user's (associated with q_i) task-completion time, and $T_{\text{thresfirm}}$ is a threshold value so that our HRI interface applies the firm assistance, $c_{\text{firm}} = 0.1$, to the user whose task-complietion time is $T_i \geq T_{\text{thresfirm}}$.

Algorithm and result table

Algorithm 5.1 describes the total procedure to customize the virtual fixture for a user. Table 5.1 shows the customized virtual fixture parameters based on the user's task-performing characteristics.

5.4 Summary

To customize a virtual fixture for a user, we processed in the former chapters and sections

- Observing the user-demonstration.

Table 5.1: The customized virtual fixture parameters the user's task-performing characteristics.

CLASS	USER#	$\lvert q_{2,i} - q_{2,*} \rvert$	$\lvert T_i - T^* \rvert$	γ_j	d	$\underline{c}_{\boldsymbol{\delta}^\perp}$
1	∗USER10	–	–	–	50.0000	0.9000
	USER20	0.0739	11.98	γ_1	43.3525	0.5450
2	∗USER13	–	–	–	50.0000	0.9000
	USER17	0.1059	1.28	γ_2	30.2480	0.8621
	USER9	0.3500	1.58	γ_2	39.1870	0.8532
	USER4	0.1296	1.80	γ_2	32.3236	0.8467
	USER21	0.2437	1.94	γ_2	27.4356	0.8425
	USER5	0.0848	2.46	γ_2	36.5373	0.8271
	USER18	0.1698	2.48	γ_2	34.7949	0.8265
	USER16	0.2857	3.80	γ_2	42.6767	0.7874
	USER12	0.0824	4.34	γ_2	40.8837	0.7714
	USER19	0.1875	10.28	γ_2	42.4900	0.5954
	USER8	0.1987	18.12	γ_2	35.4560	0.1000
	USER7	0.0560	22.22	γ_2	44.8402	0.1000
3	∗USER14	–	–	–	50.0000	0.9000
	USER1	0.2124	4.48	γ_3	44.7364	0.7673
	USER6	0.0572	5.84	γ_3	34.0157	0.7270
4	∗USER3	–	–	–	50.0000	0.9000
	USER2	0.2066	0.54	γ_4	34.3430	0.8840
	USER11	0.2161	1.44	γ_4	33.0113	0.8573
	USER15	0.2307	9.86	γ_4	33.8073	0.6079

- Inferring the unknown parameters of a cost function by solving inverse optimal control (Section 4.4.1).
- Defining the features that capture the user's task-performing characteristics (Section 4.4.2).
- Sorting the features/users based on task-completion time, and performing the closest-high-performer-based (CHPB) classification (Section 4.5).

In this chapter, we presented our approach to customizing a virtual fixture based on the defined features of the user's task-performing characteristics. We defined soft and firm assistance according to the amount of attenuation, and discussed asssited control force that results by attenuating a non-

preferred directional component from an initial control input. We also showed that the attenuating component against the initial control input can be interpreted as a suppressive term, which in turn becomes identical to the guiding force. We then described the procedure to customize the virtual fixture for a user on the basis of the class, the features, and task-completion time.

CHAPTER 6

CUSTOMIZING VISUAL FEEDBACK

In this chapter, we introduce our approach to customizing visual feedback based on user's dexterity. Our assistive HRI interface modifies the size of the working field of view (WFoV) by adjusting a zoom level to enhance a user's awareness of the quality of his/her control. First, we introduce terminology that is used in psychology-engineering-based research: steering law, the index of difficulty (ID), and the index of performance (IP). Next, we discuss the effect of increasing zoom level on the user's task-completion time and screen velocity in WFoV. We then calculate the ID for our mobile robot teleoperation task, and investigate the IP from user performance data. Finally, we describe our method to modify visual feedback, and to customize the modification for the users based on their performance.

6.1 Introduction to Steering Law

Since the early 1950s, the human factors that characterize movements and task-performance have been studied through psycho-motor behavior models [104]. Quantitative human performance models from those studies contribute not only to characterize human behavior, but also to design a human-machine interactive system [105]. For a steering task, there exists well-known models such as "Rashevsky's model" [106], "Drury's model" [107], and "Accot and Zhai's model" [92].

Fig. 6.1 shows the environment and the WFoV on a display monitor which will be considered as visual feedback. For our application, we adopt the Accot and Zhai's steering law. According to the Accot and Zhai's steering law [105], task-completion time T to successfully steer a controlled object through a path C is defined as

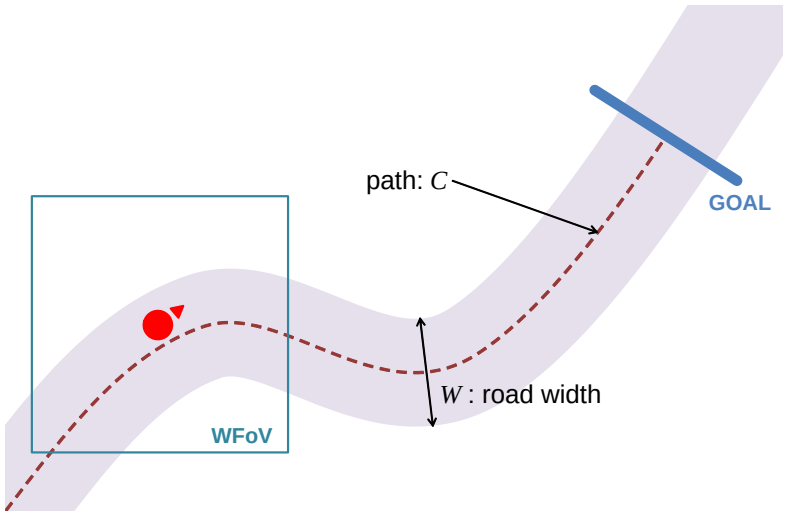


Figure 6.1: Path C and road width W to calculate ID and rel. IP using Accot and Zhai's steering law in our application.

$$T = a + b \int_C \frac{ds}{W(s)} \tag{6.1}$$

where $W(s)$ is the path width at s, and a and b are positive coefficients that depend on a user's minimum reaction time and the given task condition, respectively. In (6.1), we especially consider the integral term as the *index of difficulty (ID)* [26, 33, 104], thus

$$\mathrm{ID} = \int_C \frac{ds}{W(s)} \tag{6.2}$$

Accordingly, we can define *index of performance (IP)* [6, 92]

$$\mathrm{IP} = \frac{\mathrm{ID}}{T} \tag{6.3}$$

The IP indicates the level of user's performance for a given task. Thus, the higher IP indicates that the user is more skilled. Instead of using IP directly, we define "relative" ID as

$$\text{rel. IP} = \frac{\mathrm{IP}}{\max \mathrm{IP}} \tag{6.4}$$

which indicates a user's IP relative to the highest-performing user's IP.

Table 6.1: User's task-completion time (T), the index of performance (IP), and the relative index of performance (rel. IP). The subscripts represent a zoom level.

USER#	$T_{\alpha=1.0}$	$T_{\alpha=2.0}$	$T_{\alpha=3.0}$	$\text{IP}_{\alpha=1.0}$	$\text{IP}_{\alpha=2.0}$	$\text{IP}_{\alpha=3.0}$	rel. $\text{IP}_{\alpha=1.0}$	rel. $\text{IP}_{\alpha=2.0}$	rel $\text{IP}_{\alpha=3.0}$
USER1	17.88	26.28	36.62	0.8356	0.2842	0.1360	0.6857	0.7283	0.7640
USER2	14.02	22.80	31.52	1.0656	0.3276	0.1580	0.8745	0.8394	0.8876
USER3	13.48	19.14	27.98	1.1083	0.3903	0.1780	0.9095	1.0000	0.9999
USER4	14.66	24.08	31.84	1.0191	0.3102	0.1564	0.8363	0.7948	0.8787
USER5	15.32	23.10	30.94	0.9752	0.3234	0.1610	0.8003	0.8285	0.9043
USER6	19.24	26.34	39.96	0.7765	0.2836	0.1246	0.6372	0.7266	0.7001
USER7	42.00	33.00	43.46	0.3557	0.2264	0.1146	0.2919	0.5800	0.6438
USER8	39.86	26.54	37.34	0.3748	0.2815	0.1334	0.3076	0.7211	0.7493
USER9	14.44	21.52	32.50	1.0346	0.3471	0.1532	0.8490	0.8894	0.8608
USER10	12.26	21.46	30.62	1.2186	0.3481	0.1626	1.0000	0.8919	0.9137
USER11	14.92	24.24	35.58	1.0013	0.3082	0.1400	0.8217	0.7896	0.7863
USER12	17.20	24.96	36.92	0.8686	0.2993	0.1349	0.7128	0.7668	0.7578
USER13	12.86	23.14	31.62	1.1617	0.3228	0.1575	0.9533	0.8271	0.8848
USER14	13.40	21.76	30.14	1.1149	0.3433	0.1652	0.9149	0.8796	0.9283
USER15	23.34	30.22	38.56	0.6401	0.2472	0.1291	0.5253	0.6333	0.7256
USER16	16.66	25.38	34.34	0.8968	0.2943	0.1450	0.7359	0.7541	0.8147
USER17	14.14	22.98	33.08	1.0566	0.3251	0.1505	0.8670	0.8329	0.8458
USER18	15.34	23.16	33.08	0.9739	0.3225	0.1505	0.7992	0.8264	0.8458
USER19	23.14	24.52	35.46	0.6456	0.3046	0.1404	0.5298	0.7806	0.7890
USER20	24.24	30.60	42.32	0.6163	0.2441	0.1177	0.5058	0.6255	0.6611
USER21	14.80	20.92	29.98	1.0095	0.3571	0.1661	0.8284	0.9149	0.9332

6.2 The Effects of Zooming-in the WFoV to User's Task-Performance

6.2.1 User's Relative IP with respect to a Zoom level

If a zoom level is increased, then the road width W becomes wider in the WFoV. Suppose a total path length $C = 747$ [pixel] and a fixed road width $W = 50$ [pixel] when zoom level $\alpha = 1.0$. From (6.2), we can calculate ID for $\alpha = 1.0$, $\alpha = 2.0$, and $\alpha = 3.0$ as

$$\mathrm{ID}_{\alpha=1.0} = 14.94,\ \mathrm{ID}_{\alpha=2.0} = 7.47,\ \text{ and }\ \mathrm{ID}_{\alpha=3.0} = 4.98 \tag{6.5}$$

which shows that the ID decreases as the zoom level increases due to the wider W in the WFoV.

Now, we can calculate the relative IP for all users as depicted in Table 6.1 with T in Table 4.2. Table 6.2 shows the mean and standard deviation of relative IP of Table 6.1. From Table 6.2, we can see that the mean of relative IP increases, and the standard deviation of relative IP decreases as zoom level increases. Recall that the IP represents the user's performance for the given task; the increasing mean of relative IP represent that the average user's performance is improved. Also, the decreasing standard deviation of relative IP means that gaps among the user's performance are reduced by zooming-in the WFoV.

Table 6.2: Mean and standard deviation of relative IP in Table 6.1.

ZOOM LEVEL	$\alpha = 1.0$	$\alpha = 2.0$	$\alpha = 3.0$
rel. IP mean	0.7327	0.7919	0.8226
rel. IP std	0.1994	0.1016	0.0953

6.2.2 Improvement of Screen Velocity in WFoV

By increasing the zoom level, the mobile robot velocity will be seen α times faster in the WFoV. To sustain consistent control-to-display (CD) gain with respect to varying zoom level, we multiply control command gain ξ, which was introduced in Section 3.2.2, as a relative velocity compensating term,

thus

$$\xi = \alpha^{-1} \quad \text{and} \quad 0 < \xi \leq 1 \tag{6.6}$$

Unfortunately, multiplying ξ also limits the speed of a controlled object, and may prevent us from finding potential benefits of zooming. To check the benefit of zooming, therefore, we examine the improvement of "average screen velocity" in WFoV, which indicates that the zooming helps a human user to apply control more freely and confidently by reducing the pressure for finer control.

The average screen velocity of mobile robot, $v_{\text{s,avg}}$, can be calculated as

$$v_{\text{s,avg}} = \alpha \times v_{\text{avg}} \tag{6.7}$$

where v_{avg} represents the average velocity with respect to global coordinate.

Table 6.3: The average velocity v_{avg} and the average screen velocity $v_{\text{s, avg}}$.

USER#	v_{avg} $\alpha = 1.0$	$\alpha = 2.0$	$\alpha = 3.0$	$v_{\text{s, avg}}$ $\alpha = 1.0$	$\alpha = 2.0$	$\alpha = 3.0$
USER1	0.962	0.623	0.440	0.962	1.245	1.320
USER2	1.164	0.656	0.451	1.164	1.313	1.354
USER3	1.184	0.712	0.473	1.184	1.423	1.419
USER4	1.195	0.686	0.470	1.195	1.371	1.410
USER5	1.086	0.689	0.488	1.086	1.379	1.465
USER6	0.975	0.635	0.441	0.975	1.270	1.323
USER7	0.311	0.524	0.415	0.311	1.049	1.245
USER8	0.782	0.642	0.453	0.782	1.285	1.360
USER9	1.177	0.712	0.466	1.177	1.425	1.398
USER10	1.219	0.636	0.437	1.219	1.273	1.311
USER11	1.106	0.638	0.424	1.106	1.277	1.271
USER12	1.009	0.644	0.446	1.009	1.288	1.339
USER13	1.222	0.649	0.456	1.222	1.298	1.368
USER14	1.237	0.707	0.478	1.237	1.415	1.435
USER15	0.700	0.537	0.408	0.700	1.075	1.225
USER16	1.035	0.619	0.449	1.035	1.237	1.347
USER17	1.282	0.723	0.497	1.282	1.445	1.490
USER18	1.108	0.683	0.469	1.108	1.366	1.407
USER19	0.758	0.665	0.460	0.758	1.329	1.379
USER20	0.687	0.525	0.384	0.687	1.050	1.151
USER21	1.285	0.719	0.482	1.285	1.437	1.446

Table 6.3 shows the average velocity v_{avg} and the average screen velocity $v_{\text{s, avg}}$. Now, we calculate the improvement of $v_{\text{s, avg}}$ according to zoom levels α as

$$\begin{aligned} &\text{improvement of } v_{\text{s, avg}} \text{ by } \alpha \\ &= \frac{(v_{\text{s, avg}} \text{ at } \alpha) - (v_{\text{s, avg}} \text{ at } \alpha = 1.0)}{(v_{\text{s, avg}} \text{ at } \alpha = 1.0)} \times 100 \ [\%] \end{aligned} \tag{6.8}$$

Table 6.4 presents the improvement of $v_{\text{s, avg}}$ by α. From Table 6.4, we can see the positive effect of zooming to $v_{\text{s, avg}}$. Therefore, we design our assistive HRI system so as to modify visual feedback by adjusting a zoom level based on the quality of the user's control under specific environmental conditions.

Table 6.4: The improvement of the average screen velocity $v_{\text{s, avg}}$ by α in terms of percent [%].

USER#	$\alpha = 1.0$	$\alpha = 2.0$	$\alpha = 3.0$
USER1	0	29.48	37.27
USER2	0	12.74	16.30
USER3	0	20.16	19.85
USER4	0	14.74	18.00
USER5	0	26.99	34.94
USER6	0	30.31	35.72
USER7	0	237.09	300.24
USER8	0	64.40	74.07
USER9	0	21.05	18.74
USER10	0	4.41	7.53
USER11	0	15.49	14.95
USER12	0	27.68	32.80
USER13	0	6.24	11.99
USER14	0	14.40	16.02
USER15	0	53.61	75.12
USER16	0	19.51	30.15
USER17	0	12.74	16.26
USER18	0	23.20	26.90
USER19	0	75.43	81.98
USER20	0	52.85	67.47
USER21	0	11.85	12.51

6.3 Visual Feedback Modification and Customization

6.3.1 Modifying WFoV by Zoom Level Adjustment

Psycho-motor findings introduce the idea that changing the level of zooming affects the user's task-performance in multi-scale navigation [46, 47], surgical robot interface [44], and pointing tasks [45]. Likewise, a perceptual feedback controller for our assistive HRI interface also modifies visual feedback by adjusting a zoom level. The purpose of this modification is to provide assistance by zooming the WFoV based on the high-performing user's routing strategy by constraining the WFoV, and helping the human user to have an obvious awareness of the quality of his/her control.[1]

We define a zoom level, denoted by α, similarly to how attenuating impedance was introduced in Section 5.1, thus

$$\alpha(\|\mathbf{e}\|) = \begin{cases} \bar{\alpha}, & \text{if } \|\mathbf{e}\| > \frac{d}{2} \\ \bar{\alpha} + \left[\frac{d/2 - \|\mathbf{e}\|}{\nu}\right](1-\bar{\alpha}), & \text{if } \frac{d}{2} - \nu < \|\mathbf{e}\| \leq \frac{d}{2} \\ 1, & \text{if } \|\mathbf{e}\| \leq \frac{d}{2} - \nu \end{cases} \tag{6.9}$$

where $\bar{\alpha} > 1$ is the maximum zoom level. From (6.9), we see that if $||\mathbf{e}||$ is larger than $\frac{d}{2} - \nu$, then the zoom level is increased linearly as the mobile robot further deviates from the spine set by using the result from the CHPB classifier. According to the human operator model in Section 3.2.1, consequently, the increased level of zooming will make the user apply a larger corrective control action in the preferred direction.

6.3.2 Visual Feedback Customization

From (6.9), we know that $\bar{\alpha}$ determines the range of the function $\alpha(\|\mathbf{e}\|)$, i.e., $(1, \bar{\alpha}]$, for the deviation within a correction zone, thus $\|\mathbf{e}\| : \left(\frac{d}{2} - \nu, \frac{d}{2}\right]$. Suppose that $\frac{d}{2}$ and ν are fixed values pre-determined for a road width W. Let T_i and T_{thresmax} be the USERi's task-completion time and a threshold value to set the zoom level as the maximum zoom level, $\alpha_{\max}$, if $T_i \geq T_{\text{thresmax}}$, respectively. Now, the following definition of $\bar{\alpha}$ allows us to provide the

[1] In fact, this is one of our hypothetical questions that will be validated in Chapter 7

customized visual feedback

$$\bar{\alpha}(T_i) = -\frac{(\alpha_{\max} - 1.0)}{T_{\text{thresmax}}} \; T_i \; + \; 1.0 \tag{6.10}$$

For our HRI system, we set $\alpha_{\max}$ to be 2.0.

6.4 Summary

In this chapter, we discussed our approach to customizing visual feedback based on user's task performance. We first defined the index of difficulty and the index of performance for a mobile robot teleoperation task. Then, we investigated the effect of zooming-in the working field of view on the user's task-completion time and the screen velocity in WFoV. We observed that the average screen velocity in WFoV was improved as zoom level increased, and the standard deviation of the users' performance decreases as well; these observations provided the cues for improving the low-performing user's performance by providing the customized visual feedback. Lastly, we presented a method to adjust the zoom level to modify visual feedback, and to customize the modification for a specific user. In the next chapter, the results of user-performance evaluation under the various types of assistance will be discussed and analyzed.

CHAPTER 7

EVALUATING USER'S TASK PERFORMANCE

This chapter presents the results of evaluating user performance with the developed assistive HRI interface. In Chapters 5 and 6, we described a method for customizing haptic and visual feedback based on a user's task-performing characteristics and performance, respectively. For evaluation, 10 assistance modes are applied to the user. With the results, we validate our assistance customization approach by comparing the user performance to the other cases. We begin by introducing the experiment design: experimental apparatus and protocol, assistance modes, and hypothetical questions. Then, we present user task-completion time and average velocity under various assistance modes. Finally, we investigate the enhancement of user performance, and answer to our hypothetical questions based on the evaluation results.

7.1 Experiment Design for Evaluating User's Task-Performance

7.1.1 Experimental Apparatus and Protocol

Subjects were instructed to sit comfortably on an office chair with wheels in front of a computer desk, and were provided with an input device and a display monitor. A Novint FALCON haptic joytick was used as the input device, and we used a typical 17-inch monitor as the display. Fig. 7.1 shows the apparatus of the experimental setup.

A brief introduction about the experiment procedure was given to all subjects as well as short demonstration. Then the subjects were given time to practice, familiarizing themselves with the interface by driving a mobile robot along a road. We limited the practice time to 5-min to prevent performance enhancement by learning effects.

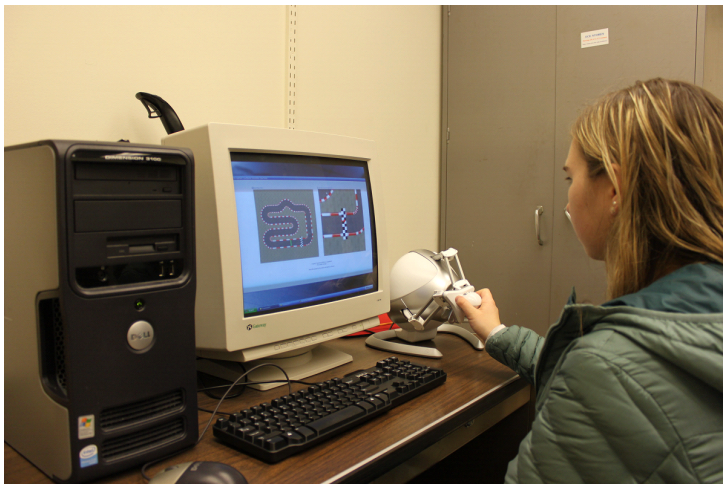


Figure 7.1: The apparatus of the experimental setup. The subject is performing a given task (the photo is used by courtesy of the participant).

After the practice stage, the subjects started the first stage (testing stage). During the first stage, our simulator provided 4 different task locations wherein roads had various curvatures but with the fixed width. For each task, the subjects performed 5 trials. Among the 5 trials, we picked the median data in terms of task-completion time, and followed the procedure of solving IOC, classifying the subjects, and customizing the virtual fixture parameters.

Finally, the subjects were instructed to complete an entire lap along a given race track during the second stage (performance evaluation stage). When the mobile robot collided with the road boundaries, the task began from the center of the road at the point of failure. The maximum number of collisions was limited to ten. There were 10 tasks, according to various assistance modes, from no-assistance to haptic+visual feedback with potential function-based guidance.

7.1.2 Acronyms for Representing Assistance Modes

Throughout this chapter, we use acronyms to represent either sole or combined assistance modes as follows:

- V : visual feedback
- H : haptic feedback

- D : use default parameters for a virtual fixture: $d = 50.0$ and $\underline{c}_{\boldsymbol{\delta}^\perp} = 0.2$ were used as the default parameters.
- C : set customized parameters for a virtual fixture
- S : spine generated from the highest-performer's cost function.
- P : define attractive/repulsive potential field function with respect to road center/boundaries.
- R : road center was used to define attractive potential function.

We use XX/YY format in which the first two capital letters represent feedback types and the next ones describe virtual fixture settings. For instance, VH/CS means "visual + haptic feedback" and a virtual fixture with "customized parameters and spine."

7.1.3 Hypothetical Questions

The three hypothetical questions we want to know are as follows:

- **Question 1:** (Overall effects) Is performance different for different feedback types and different active constraint settings?

With question 1, we first want to verify the main effect and the interaction effect of assistance mode, respectively.

- **Question 2:** (Individual effects) What will be the results of examining each individual assistance mode's effect on performance (either positive or negative)?

As our assistive HRI interface provides assistive control and a guiding force for the user's control, we want to know the answer to question 2 to investigate the enhancement of the subject performance.

- **Question 3:** (The advantage of haptic assistance customization) Does H/CS outperform the other assistance modes?

In addition to question 2, the above query verifies the advantage of our approach that provides the customized assistance based on the user dexterity.

We note that "task-completion time" and "average velocity" will be used as performance criteria.

7.2 Result Data

Table 7.1 and Table 7.2 show subject task-completion time and average velocity, respectively. We also present the subject performance enhancement in Table 7.3 and Table 7.4. For task-completion time T, we calculate the "positive" enhancement in percent as

$$\begin{aligned} &\text{Enhancement in task-completion time } T \text{ [\%]} \\ &\quad = \frac{T \text{ with No Assist} - T \text{ with Asssitance type}}{T \text{ with No assist}} \times 100 \end{aligned} \tag{7.1}$$

which means "decrement" of task-completion time as shown in Table 7.3. For average velocity, the enhancement is defined by

$$\begin{aligned} &\text{Enhancement in average velocity } v_{\text{avg}} \text{ [\%]} \\ &\quad = \frac{v_{\text{avg}} \text{ with Asssitance type} - v_{\text{avg}} \text{ with No assist}}{v_{\text{avg}} \text{ with No Assist}} \times 100 \end{aligned} \tag{7.2}$$

which represents "increment" of average velocity as depicted in Table 7.4. By glancing at the result data, it can be seen from the results that the assistance mode H/CS shows the best performance enhancement in both T and v_{avg}. We will further analyze the result data applying statistical methods in Section 7.3.

We note that Table 7.1 through Table 7.4 are presented as bar plots for each user in Appendix B.

7.3 Checking Hypothetical Question and Data Analysis

In this section, we answer our hypothetical questions that were framed in Section 7.1.3 and further examine the subject data.

7.3.1 Answers to the Three Hypothetical Questions

To examine the three hypothetical questions, we use the enhancement data of the task-completion time and the average velocity presented in Table 7.3

Table 7.1: Task-completion time in seconds with respect to assistance modes.

USER#	No Assist	V/DS	H/DS	VH/DS	V/CS	H/CS	VH/CS	V/PR	H/PR	VH/PR
USER1	27.34	30.74	24.86	31.06	31.84	25.56	32.80	27.50	31.60	30.46
USER2	22.06	24.90	20.72	23.92	24.92	18.68	25.62	27.94	23.10	24.08
USER3	32.48	31.98	34.04	34.16	33.46	23.28	31.00	29.14	28.46	30.94
USER4	21.14	25.72	21.32	24.28	23.54	21.98	25.08	25.16	21.40	23.76
USER5	35.26	37.50	33.34	34.78	30.82	26.76	35.58	36.50	29.38	30.70
USER6	27.22	31.16	27.20	30.46	32.94	24.16	32.36	29.50	26.14	28.18
USER7	38.44	40.60	38.52	33.68	40.16	37.36	36.06	40.06	34.48	34.66
USER8	34.12	33.82	24.08	30.26	35.76	25.50	34.22	27.08	23.46	33.26
USER9	73.04	59.96	46.90	56.78	43.74	31.20	49.12	52.90	34.52	36.08
USER10	33.12	35.02	37.34	40.10	33.18	29.44	31.38	29.50	27.80	31.98
USER11	22.02	27.80	23.40	28.50	29.80	23.06	27.56	28.68	22.58	28.30
USER12	24.14	38.86	26.16	30.82	32.30	25.42	28.96	30.44	24.68	31.32
USER13	26.04	29.12	27.68	28.90	29.58	23.94	28.82	24.40	21.88	25.46
USER14	33.02	30.82	28.76	27.88	35.24	24.30	34.24	26.76	26.46	30.74
USER15	32.98	36.60	29.68	36.48	35.54	26.84	33.42	33.90	26.74	31.22
USER16	23.84	25.60	25.10	28.16	28.40	23.02	30.24	28.38	25.98	29.64
USER17	24.30	32.34	31.74	30.42	31.08	25.48	30.54	26.60	32.60	26.48
USER18	24.26	29.98	27.30	28.00	29.66	24.24	29.84	25.98	28.44	31.04
USER19	28.94	31.06	24.30	28.66	30.80	22.70	30.40	26.96	24.04	30.56
USER20	38.04	35.42	29.60	28.2	30.50	24.7	30.10	26.72	23.76	25.88
USER21	23.48	30.6	26.78	30.42	34.26	23.58	29.94	28.78	24.70	31.36
USER22	30.32	33.04	29.62	33.86	34.62	26.84	32.00	26.9	27.94	27.08
USER23	30.78	31.22	28.68	32.16	31.08	24.46	32.96	31.64	30.88	34.76
MEAN	30.71	33.21	29.01	31.82	32.31	25.33	31.84	30.06	27.00	29.91
STD	10.60	7.12	6.00	6.57	4.33	3.63	4.69	6.16	3.86	3.32
MEDIAN	28.94	31.22	27.68	30.42	31.84	24.46	31.00	28.38	26.46	30.70

Table 7.2: Average speed in pixel/second with respect to assistance modes.

USER#	No Assist	V/DS	H/DS	VH/DS	V/CS	H/CS	VH/CS	V/PR	H/PR	VH/PR
USER1	1.131	0.943	1.177	0.952	0.947	1.159	0.898	1.099	0.996	1.041
USER2	1.448	1.162	1.487	1.248	1.236	1.525	1.142	1.131	1.425	1.254
USER3	0.928	0.963	0.882	0.898	0.911	1.283	0.981	1.076	1.105	0.994
USER4	1.384	1.163	1.444	1.190	1.203	1.417	1.199	1.243	1.447	1.283
USER5	0.897	0.817	0.951	0.888	0.992	1.163	0.964	0.955	1.073	1.032
USER6	1.135	0.964	1.204	1.020	0.908	1.296	0.934	1.022	1.237	1.106
USER7	0.906	0.822	0.954	0.951	0.837	0.995	0.884	0.840	0.959	0.942
USER8	1.015	0.910	1.323	1.015	0.876	1.242	0.917	1.069	1.355	0.928
USER9	0.483	0.555	0.680	0.563	0.772	1.024	0.653	0.684	0.932	0.874
USER10	0.896	0.865	0.802	0.755	0.904	1.030	0.947	1.004	1.131	0.957
USER11	1.327	1.073	1.340	1.086	1.023	1.311	1.066	1.083	1.370	1.123
USER12	1.308	0.960	1.190	1.011	0.974	1.243	1.071	1.078	1.311	0.998
USER13	1.139	1.014	1.098	1.027	1.013	1.245	1.044	1.233	1.345	1.161
USER14	0.908	0.961	1.049	1.070	0.849	1.228	0.858	1.136	1.179	0.992
USER15	1.038	0.864	1.060	0.853	0.882	1.203	0.933	0.964	1.208	1.027
USER16	1.276	1.145	1.227	1.064	1.036	1.339	0.992	1.111	1.226	1.058
USER17	1.267	0.990	1.111	1.041	0.999	1.229	1.018	1.155	1.010	1.177
USER18	1.225	0.974	1.103	1.038	1.022	1.239	0.979	1.151	1.073	0.990
USER19	1.122	0.999	1.285	1.056	1.028	1.369	1.022	1.169	1.313	1.027
USER20	0.872	0.837	1.015	1.071	0.983	1.264	0.975	1.144	1.291	1.170
USER21	1.309	1.030	1.229	1.065	0.890	1.347	1.037	1.103	1.282	0.997
USER22	0.975	0.888	1.009	0.873	0.855	1.099	0.905	1.106	1.100	1.107
USER23	1.052	0.975	1.087	0.998	0.986	1.273	0.956	1.063	1.020	0.926
MEAN	1.089	0.951	1.118	0.988	0.962	1.240	0.973	1.070	1.191	1.051
STD	0.220	0.132	0.195	0.142	0.109	0.125	0.107	0.123	0.154	0.107
MEDIAN	1.122	0.963	1.103	1.020	0.974	1.243	0.975	1.099	1.208	1.027

Table 7.3: Enhancement in task-completion time in terms of percent: the ratio of an assistance mode to no-assistance.

USER#	No Assist	V/DS	H/DS	VH/DS	V/CS	H/CS	VH/CS	V/PR	H/PR	VH/PR
USER1	-	-12.4	9.1	-13.6	-16.5	6.5	-20.0	-0.6	-15.6	-11.4
USER2	-	-12.9	6.1	-8.4	-13.0	15.3	-16.1	-26.7	-4.7	-9.2
USER3	-	1.5	-4.8	-5.2	-3.0	28.3	4.6	10.3	12.4	4.7
USER4	-	-21.7	-0.9	-14.9	-11.4	-4.0	-18.6	-19.0	-1.2	-12.4
USER5	-	-6.4	5.4	1.4	12.6	24.1	-0.9	-3.5	16.7	12.9
USER6	-	-14.5	0.1	-11.9	-21.0	11.2	-18.9	-8.4	4.0	-3.5
USER7	-	-5.6	-0.2	12.4	-4.5	2.8	6.2	-4.2	10.3	9.8
USER8	-	0.9	29.4	11.3	-4.8	25.3	-0.3	20.6	31.2	2.5
USER9	-	17.9	35.8	22.3	40.1	57.3	32.7	27.6	52.7	50.6
USER10	-	-5.7	-12.7	-21.1	-0.2	11.1	5.3	10.9	16.1	3.4
USER11	-	-26.2	-6.3	-29.4	-35.3	-4.7	-25.2	-30.2	-2.5	-28.5
USER12	-	-61.0	-8.4	-27.7	-33.8	-5.3	-20.0	-26.1	-2.2	-29.7
USER13	-	-11.8	-6.3	-11.0	-13.6	8.1	-10.7	6.3	16.0	2.2
USER14	-	6.7	12.9	15.6	-6.7	26.4	-3.7	19.0	19.9	6.9
USER15	-	-11.0	10.0	-10.6	-7.8	18.6	-1.3	-2.8	18.9	5.3
USER16	-	-7.4	-5.3	-18.1	-19.1	3.4	-26.8	-19.0	-9.0	-24.3
USER17	-	-33.1	-30.6	-25.2	-27.9	-4.9	-25.7	-9.5	-34.2	-9.0
USER18	-	-23.6	-12.5	-15.4	-22.3	0.1	-23.0	-7.1	-17.2	-27.9
USER19	-	-7.3	16.0	1.0	-6.4	21.6	-5.0	6.8	16.9	-5.6
USER20	-	6.9	22.2	25.9	19.8	35.1	20.9	29.8	37.5	32.0
USER21	-	-30.3	-14.1	-29.6	-45.9	-0.4	-27.5	-22.6	-5.2	-33.6
USER22	-	-9.0	2.3	-11.7	-14.2	11.5	-5.5	11.3	7.8	10.7
USER23	-	-1.4	6.8	-4.5	-1.0	20.5	-7.1	-2.8	-0.3	-12.9
MEAN	-	-11.6	2.4	-7.3	-10.2	13.4	-8.1	-1.7	7.3	-2.9
STD	-	16.3	14.9	16.0	18.4	15.3	15.6	17.3	19.0	19.9
MEDIAN	-	-9.0	0.1	-11.0	-11.4	11.2	-7.1	-2.8	7.8	-3.5

Table 7.4: Enhancement in average speed in terms of percent: the ratio of an assistance mode to no-assistance.

USER#	No Assist	V/DS	H/DS	VH/DS	V/CS	H/CS	VH/CS	V/PR	H/PR	VH/PR
USER1	-	-16.6	4.0	-15.8	-16.2	2.5	-20.6	-2.8	-11.9	-8.0
USER2	-	-19.7	2.7	-13.8	-14.6	5.3	-21.1	-21.9	-1.6	-13.4
USER3	-	3.8	-4.9	-3.2	-1.8	38.3	5.8	15.9	19.1	7.1
USER4	-	-16.0	4.4	-14.0	-13.1	2.4	-13.4	-10.2	4.6	-7.3
USER5	-	-8.9	6.1	-1.0	10.6	29.7	7.5	6.5	19.7	15.2
USER6	-	-15.1	6.1	-10.2	-20.1	14.2	-17.7	-10.0	9.0	-2.6
USER7	-	-9.3	5.3	5.0	-7.6	9.9	-2.4	-7.3	5.9	4.0
USER8	-	-10.4	30.3	-0.1	-13.8	22.3	-9.7	5.2	33.4	-8.6
USER9	-	14.8	40.7	16.4	59.7	111.8	35.0	41.5	92.9	80.8
USER10	-	-3.4	-10.5	-15.8	0.9	15.0	5.8	12.1	26.2	6.8
USER11	-	-19.2	1.0	-18.1	-22.9	-1.2	-19.6	-18.4	3.2	-15.4
USER12	-	-26.6	-9.0	-22.7	-25.5	-4.9	-18.1	-17.6	0.3	-23.7
USER13	-	-10.9	-3.6	-9.8	-11.1	9.4	-8.3	8.3	18.1	2.0
USER14	-	5.8	15.4	17.8	-6.6	35.2	-5.5	25.0	29.7	9.2
USER15	-	-16.8	2.2	-17.9	-15.0	15.9	-10.1	-7.1	16.4	-1.0
USER16	-	-10.3	-3.8	-16.6	-18.8	4.9	-22.2	-13.0	-4.0	-17.1
USER17	-	-21.8	-12.3	-17.8	-21.1	-3.0	-19.6	-8.8	-20.2	-7.1
USER18	-	-20.5	-10.0	-15.2	-16.5	1.1	-20.1	-6.0	-12.4	-19.2
USER19	-	-10.9	14.6	-5.9	-8.3	22.0	-8.8	4.2	17.1	-8.5
USER20	-	-4.0	16.4	22.8	12.8	45.0	11.8	31.3	48.1	34.3
USER21	-	-21.3	-6.1	-18.7	-32.0	2.9	-20.8	-15.7	-2.0	-23.9
USER22	-	-8.9	3.6	-10.4	-12.3	12.8	-7.2	13.5	12.8	13.6
USER23	-	-7.3	3.3	-5.2	-6.3	21.0	-9.2	1.0	-3.1	-12.0
MEAN	-	-11.0	4.2	-7.4	-8.7	17.9	-8.2	1.1	13.1	0.2
STD	-	9.8	12.8	12.6	18.3	24.6	13.9	16.5	23.7	22.3
MEDIAN	-	-10.9	3.3	-10.4	-13.1	12.8	-9.7	-2.8	9.0	-7.1

and Table 7.4, respectively. We present the results in APA format introduced in [108].

Answer to Question 1

The first hypothetical question that we examined was:

- **Question 1:** (Overall effects) Is performance different for different feedback types and different active constraint settings?

To answer the question 1, we applied rANOVA (repeated measures analysis of variance). We set feedback types (V/H/VH) and active constraint types (DS/CS /PR) as two repeated-measure factors.

First, we ran the test with the enhancement of T data in Table 7.3. The rANOVA on task-completion time yielded a main effect for the feedback types, $F(2, 44) = 62.959, p < 0.01$, indicating that the mean change score was significant. The main effect of the active constraint types was significant, $F(2, 44) = 8.741, p < 0.01$. Also, the interaction effect was significant, $F(4, 88) = 8.951, p < 0.01$.

Next, we tested the enhancement of v_{avg} data in Table 7.4. The rANOVA on average velocity showed a significant main effect for the feedback types, $F(2, 44) = 87.369, p < 0.01$. The main effect of the active constraint types was significant, $F(2, 44) = 10.915$, $p < 0.01$. Lastly, the interaction effect was also significant, $F(4, 44) = 10.558$, $p < 0.01$.

Consequently, we answered to the question 1 by showing the main effects and the interaction effects from rANOVA results.

Answer to Question 2

Second, we investigated the following hypothetical question:

- **Question 2:** (Individual effects) What will be the results of examining each individual assistance mode's effect on performance (either positive or negative)?

We performed one-sample t-test for each assistance mode as presented in Table 7.5. First, the results showed that H/CS (significant for both T and v_{avg}) and H/PR (significant for v_{avg} and marginally significant for T) enhanced

Table 7.5: The effect of overall assistance mode for the enhancement in T and v_{avg}.

T or v_{avg}	assist mode	MEAN(STD)	t-score	p-value	enhancement
T	V/DS	-11.626(16.292)	t(22)=-3.423	0.002	-
	H/DS	2.353(14.910)	t(22)=0.757	0.457	+
	VH/DS	-7.324(16.014)	t(22)=-2.193	0.039	-
	V/CS	-10.250(18.437)	t(22)=-2.666	0.014	-
	H/CS	13.389(15.335)	t(22)=4.187	0.001	+
	VH/CS	-8.119(15.632)	t(22)=-2.491	0.021	-
	V/PR	-1.735(17.316)	t(22)=-0.481	0.636	-
	H/PR	7.316(19.012)	t(22)=1.845	0.078	+
	VH/PR	-2.908(19.937)	t(22)=-0.700	0.491	-
v_{avg}	V/DS	-11.023(9.784)	t(22)=-5.403	0.001	-
	H/DS	4.161(12.790)	t(22)=1.560	0.133	+
	VH/DS	-7.396(12.578)	t(22)=-2.820	0.010	-
	V/CS	-8.679(18.258)	t(22)=-2.280	0.033	-
	H/CS	17.931(24.561)	t(22)=3.501	0.002	+
	VH/CS	-8.206(13.901)	t(22)=-2.831	0.010	-
	V/PR	1.122(16.529)	t(22)=0.325	0.748	+
	H/PR	13.097(23.716)	t(22)=2.648	0.015	+
	VH/PR	0.228(22.349)	t(22)=0.049	0.961	+

user performance. Second, V/DS, VH/DS, V/CS, and VH/CS significantly impaired user performance (both in T and v_{avg}). Lastly, there was that H/DS, V/PR, and VH/PR had a significant effect on the subject performance.

The result of the second hypothetical question examination is in accordance with the properties of V (visual feedback), DS (virtual fixture set by default parameters), and PR (potential function-based method). We know that V not only reduces a user's mental pressure to apply delicate control, but also slows down the speed of a controlled object (a mobile robot); hence the effect on user performance is determined by trade-off between those two. To further investigate this, we performed paired t-test for the average screen velocity $v_{\text{s, avg}}$ of various zoom levels α, denoted by $v_{\text{s, avg}}^{\alpha=\text{level}}$, presented in Table 6.3. First, the paired t-test for $v_{\text{s, avg}}^{\alpha=2.0}$ (mean=1.298) and $v_{\text{s, avg}}^{\alpha=1.0}$ (mean=1.023) yielded $t(20) = 7.642$, $p < 0.01$, indicating that increasing zoom level from $\alpha = 1.0$ to $\alpha = 2.0$ improves the screen velocity significantly. The paired t-test for $v_{\text{s, avg}}^{\alpha=3.0}$ (mean=1.355) and $v_{\text{s, avg}}^{\alpha=2.0}$ (mean=1.289) resulted in $t(20) = 5.080$, $p < 0.01$, which also indicated the screen velocity improvement.

Answer to Question 3

The third hypothetical question was

- **Question 3:** (The advantage of haptic assistance customization) Does H/CS outperform the other assistance modes?

To examine the question 3, we performed paired t-tests for H/CS and H/PR.[1] First, the paired t-test for H/CS (mean=13.39, std=15.34) and H/PR (mean= 7.32, std=19.01) on task-completion time yielded $t(22) = 2.744$, $p < 0.05$, indicating that H/CS significantly outperforms H/PR. We observed the same result on the average velocity, $t(22) = 2.322$, $p < 0.05$ for H/CS(mean=17.93, std=24.56) and H/PR(mean=13.10, std=23.72); hence, H/CS outperforms the other assistance modes. Finally, we found the answer to the hypothetical question 3.

7.3.2 Further Observations

We observed some interesting phenomena in subject data. We first sorted the users in ascending order according to their task-completion time under "No Assist" mode. Then, we investigate the amount of positive enhancement under various assistance modes for all users. Table 7.6 and Table 7.7 present positive enhancement in T and v_{avg} under the assistance modes for all users. In both tables, it can be seen that the more assistive modes become positively effective as we look up from top to bottom. However, we note that no further generalization can be made from these observations.

7.4 Summary

In this chapter, we showed the results and data analysis for user performance evaluation with our assistive HRI interface and assistance customization method. First, we introduced the experimental apparatus and protocol as well as three hypothetical questions. We then presented task-completion time and average velocity, and the enhancement in task-completion time and

[1] From the answer to the question 2, we had found that the best two assistance modes, H/CS and H/PR, brought force (marginally) significant enhancement.

Table 7.6: Positive enhancement in task-completion time. The users are sorted in ascending order according to T under No Assist mode.

USER#	No Assist	V/DS	H/DS	VH/DS	V/CS	H/CS	VH/CS	V/PR	H/PR	VH/PR
USER4	-	-21.7	-0.9	-14.9	-11.4	-4.0	-18.6	-19.0	-1.2	-12.4
USER11	-	-26.2	-6.3	-29.4	-35.3	-4.7	-25.2	-30.2	-2.5	-28.5
USER2	-	-12.9	6.1	-8.4	-13.0	15.3	-16.1	-26.7	-4.7	-9.2
USER21	-	-30.3	-14.1	-29.6	-45.9	-0.4	-27.5	-22.6	-5.2	-33.6
USER16	-	-7.4	-5.3	-18.1	-19.1	3.4	-26.8	-19.0	-9.0	-24.3
USER12	-	-61.0	-8.4	-27.7	-33.8	-5.3	-20.0	-26.1	-2.2	-29.7
USER18	-	-23.6	-12.5	-15.4	-22.3	0.1	-23.0	-7.1	-17.2	-27.9
USER17	-	-33.1	-30.6	-25.2	-27.9	-4.9	-25.7	-9.5	-34.2	-9.0
USER13	-	-11.8	-6.3	-11.0	-13.6	8.1	-10.7	6.3	16.0	2.2
USER6	-	-14.5	0.1	-11.9	-21.0	11.2	-18.9	-8.4	4.0	-3.5
USER1	-	-12.4	9.1	-13.6	-16.5	6.5	-20.0	-0.6	-15.6	-11.4
USER19	-	-7.3	16.0	1.0	-6.4	21.6	-5.0	6.8	16.9	-5.6
USER22	-	-9.0	2.3	-11.7	-14.2	11.5	-5.5	11.3	7.8	10.7
USER23	-	-1.4	6.8	-4.5	-1.0	20.5	-7.1	-2.8	-0.3	-12.9
USER3	-	1.5	-4.8	-5.2	-3.0	28.3	4.6	10.3	12.4	4.7
USER15	-	-11.0	10.0	-10.6	-7.8	18.6	-1.3	-2.8	18.9	5.3
USER14	-	6.7	12.9	15.6	-6.7	26.4	-3.7	19.0	19.9	6.9
USER10	-	-5.7	-12.7	-21.1	-0.2	11.1	5.3	10.9	16.1	3.4
USER8	-	0.9	29.4	11.3	-4.8	25.3	-0.3	20.6	31.2	2.5
USER5	-	-6.4	5.4	1.4	12.6	24.1	-0.9	-3.5	16.7	12.9
USER20	-	6.9	22.2	25.9	19.8	35.1	20.9	29.8	37.5	32.0
USER7	-	-5.6	-0.2	12.4	-4.5	2.8	6.2	-4.2	10.3	9.8
USER9	-	17.9	35.8	22.3	40.1	57.3	32.7	27.6	52.7	50.6

Table 7.7: Positive enhancement in average velocity. The users are sorted in ascending order according to T under No Assist mode.

USER#	No Assist	V/DS	H/DS	VH/DS	V/CS	H/CS	VH/CS	V/PR	H/PR	VH/PR
USER4	-	-16.0	4.4	-14.0	-13.1	2.4	-13.4	-10.2	4.6	-7.3
USER11	-	-19.2	1.0	-18.1	-22.9	-1.2	-19.6	-18.4	3.2	-15.4
USER2	-	-19.7	2.7	-13.8	-14.6	5.3	-21.1	-21.9	-1.6	-13.4
USER21	-	-21.3	-6.1	-18.7	-32.0	2.9	-20.8	-15.7	-2.0	-23.9
USER16	-	-10.3	-3.8	-16.6	-18.8	4.9	-22.2	-13.0	-4.0	-17.1
USER12	-	-26.6	-9.0	-22.7	-25.5	-4.9	-18.1	-17.6	0.3	-23.7
USER18	-	-20.5	-10.0	-15.2	-16.5	1.1	-20.1	-6.0	-12.4	-19.2
USER17	-	-21.8	-12.3	-17.8	-21.1	-3.0	-19.6	-8.8	-20.2	-7.1
USER13	-	-10.9	-3.6	-9.8	-11.1	9.4	-8.3	8.3	18.1	2.0
USER6	-	-15.1	6.1	-10.2	-20.1	14.2	-17.7	-10.0	9.0	-2.6
USER1	-	-16.6	4.0	-15.8	-16.2	2.5	-20.6	-2.8	-11.9	-8.0
USER19	-	-10.9	14.6	-5.9	-8.3	22.0	-8.8	4.2	17.1	-8.5
USER22	-	-8.9	3.6	-10.4	-12.3	12.8	-7.2	13.5	12.8	13.6
USER23	-	-7.3	3.3	-5.2	-6.3	21.0	-9.2	1.0	-3.1	-12.0
USER3	-	3.8	-4.9	-3.2	-1.8	38.3	5.8	15.9	19.1	7.1
USER15	-	-16.8	2.2	-17.9	-15.0	15.9	-10.1	-7.1	16.4	-1.0
USER14	-	5.8	15.4	17.8	-6.6	35.2	-5.5	25.0	29.7	9.2
USER10	-	-3.4	-10.5	-15.8	0.9	15.0	5.8	12.1	26.2	6.8
USER8	-	-10.4	30.3	-0.1	-13.8	22.3	-9.7	5.2	33.4	-8.6
USER5	-	-8.9	6.1	-1.0	10.6	29.7	7.5	6.5	19.7	15.2
USER20	-	-4.0	16.4	22.8	12.8	45.0	11.8	31.3	48.1	34.3
USER7	-	-9.3	5.3	5.0	-7.6	9.9	-2.4	-7.3	5.9	4.0
USER9	-	14.8	40.7	16.4	59.7	111.8	35.0	41.5	92.9	80.8

in average velocity for all subjects. Finally, we tested the three hypothetical questions by rANOVA, one-sample t-test, and paired t-test, respectively. With the answer to the hypothetical questions, we especially validated the effectiveness of our assistance customization method by showing that the resulting performance of assisting the user with haptic feedback and the customized virtual fixture, H/CS, surpassed the other assistance modes.

CHAPTER 8

CONCLUSION

This dissertation is about developing an assistive human-robot interaction (HRI) interface that provides the customized assistance based on the user's dexterity. The backbone of our approach is to customize virtual fixture parameters so that they adjust the level of assistance according to the user's task-performing characteristics and task-performance. Our approach also considers providing the customized two types of perceptual feedback, haptic and visual feedback, as guidance to help low-performing users by incorporating strategies of high-performing users into the developed HRI interface.

Our approach utilizes inverse optimal control to model a user's task-perfor ming characteristics. The underlying assumption for this approach is that the human user is optimizing a cost function during a given task. After inferring the cost function by solving inverse optimal control with the user-demonstrated data, we defined three features to capture the user's task-performing characteristics. We used the high-performing user's features as centroids for our classification method which allows us to classify users relative to a high-performing user whose task-performing characteristics are most similar. The defined features also serve as reasonable cues for customizing virtual fixture parameters, which, in turn, determine the amount of guiding force and zoom level adjustment.

We evaluated subject performance with our developed HRI interface. Various assistance modes were applied to the subjects. By examining the three hypothetical questions, we answered the overall effects of assistance modes, individual effect of each assistance mode, and the advantage of haptic assistance customization. Especially, we validated the effectiveness of our assistance customization approach by comparing two assistance modes, H/CS and H/PR, which showed the better enhancement of user task-performance against the other assistance modes. We concluded that the enhancement under the assistance mode of "haptic feedback+customized virtual fixture"

outperformed the other assistance modes.

8.1 Future Work

As this research is the first milestone to apply our assistance customization for a specific target application of mobile robot teleoperation, we have chosen the three intuitive features highly related to car-driving strategies. Also, by limiting the number of features to three, we could get an insight for our classification method by visualizing each user's task-performing characteristic in the feature space. Further research should explore ways of both finding other relative features and determining principle components when the feature space dimension becomes high.

In human subject experiments for modeling user's task-performing characteristics and evaluating system, our subject pool size was around 20 subjects. We should increase the size of the subject pool and obtain more various users' task-performing characteristics. Definitely, we can expect that data analysis and hypothesis tests with the larger subject pool will yield a more robust relationship between assistance modes and user performance.

The larger subject pool can also contribute to finding the distribution, e.g., Gaussian distribution, of specific task-performing characteristics. This allows us to define general (or most typical) user strategies for a specific given task. Another expected advantage is that we may find a relationship between task-performing characteristics and task-performance as a continuous (mapping) function. The function can be utilized as a powerful tool to customize the level of assistance.

Lastly, this approach should be applied and tested to more complex real world applications. For instance, in practice a robot usually sends the human operator a 3D image instead of a 2D bird view image. Also, most robots are equipped with multiple types of sensors to obtain more robust information by sensor-fusioning. Therefore, the future implementation should concern the perspectives from another basis function for IOC, e.g., depth of the image or 360 degree range information, to new features related to the user's task-completing strategies.

APPENDIX A

APPLIED ALGORITHM TO SOLVE INVERSE OPTIMAL CONTROL

Algorithm A.1: Method to Solve (2.18)

Input: (x^*, u^*)
Output: c

$c, \lambda_0 \leftarrow$ RandomVectors
$N \leftarrow$ Length(x^*, u^*)
$oldsum \leftarrow$ SomeLargeNumber

while do
  $temp \leftarrow 0$
  for $k \leftarrow 0$ **to** $N-1$ **do**
    $A_k^* \leftarrow A\big|_{(x_k^*, u_k^*)}$ (2.16)
    $\lambda_{k+1} \leftarrow$ CostateEvaluation$(c, \lambda_k, x_k^*, u_k^*)$ (2.13)
    $z_k \leftarrow \begin{bmatrix} c^T & \lambda_{k+1}^T & \lambda_k^T \end{bmatrix}^T$ (2.15)
    $temp \leftarrow temp + A_k^* z_k$
  end
  $sum \leftarrow temp$
  $termcond \leftarrow$ Tolerance$(sum, oldsum)$
  if $termcond < \epsilon$ **then**
    return c
  else
    $c, \lambda_0 \leftarrow$ FindMinimizer$(sum, oldsum, c, \lambda_0)$
  end
  $oldsum \leftarrow sum$
end

APPENDIX B

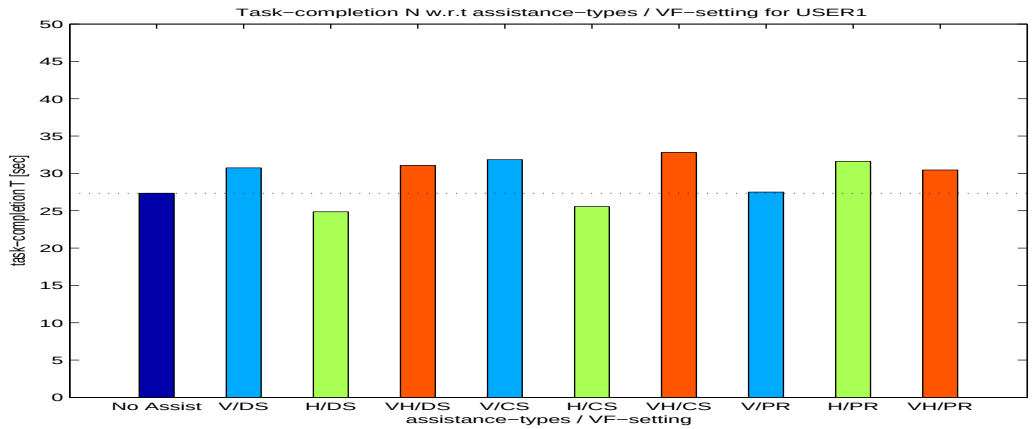
RESULT PLOTS

Fig. B.1 shows the subjects performing given task during the evaluation stage introduced in Section 7.1. In this appendix, we present Table 7.1 through Table 7.4 as bar plots. As more than 40-figures are presented, we specify the figure data as follows:

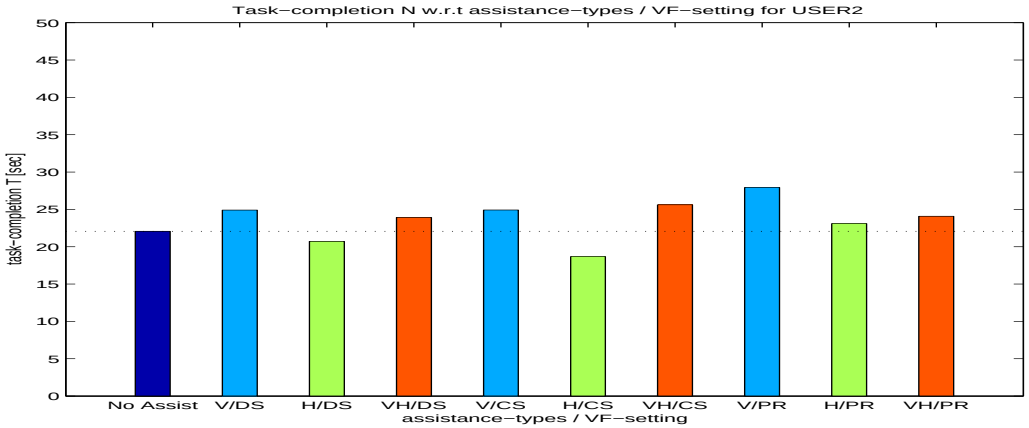
- Task-completion time with respect to assistance types (Fig. B.2)
- Enhancement in task-completion time T in terms of percent with respect to assistance types (Fig. B.3)
- Average velocity with respect to assistance types (Fig. B.4)
- Enhancement in average velocity v_{avg} in terms of percent with respect to assistance types (Fig. B.5)

We note that USER4* and USER9‡ were the highest-performer and the lowest-performer respectively.

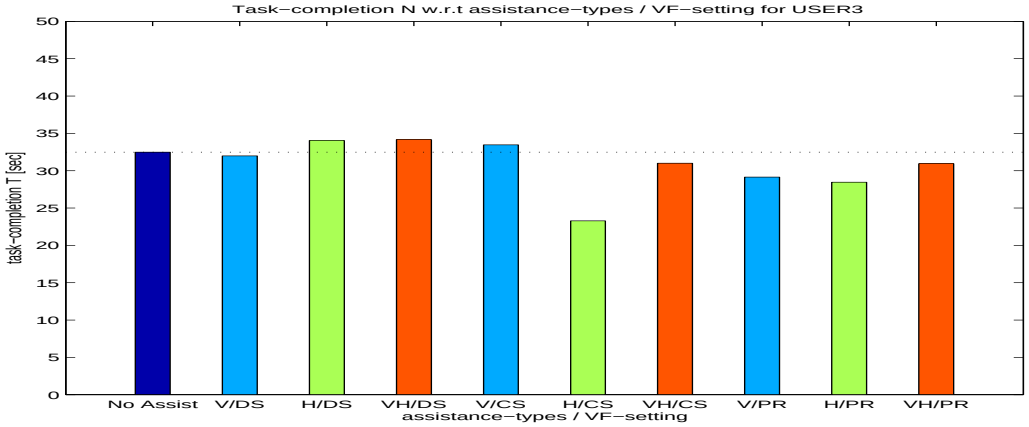
Figure B.1: The subjects are performing given task during the evaluation (the photos are used by courtesy of the participant with agreement).



(a) USER1

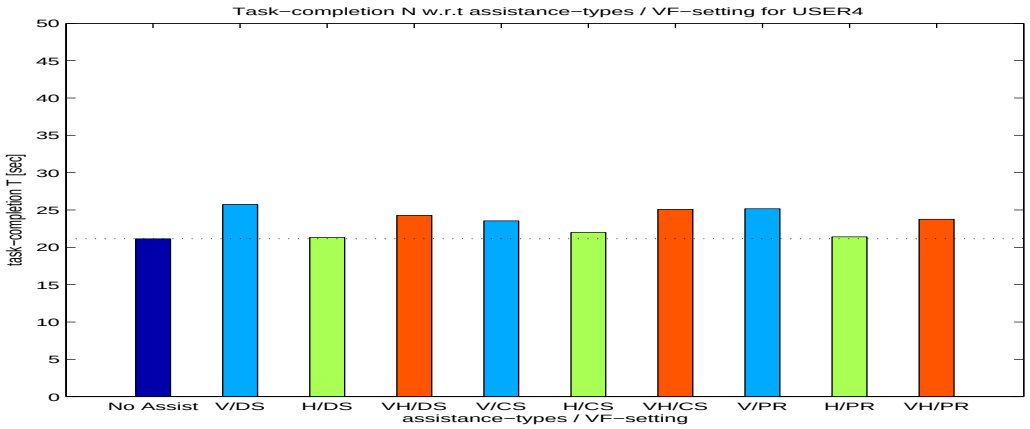


(b) USER2

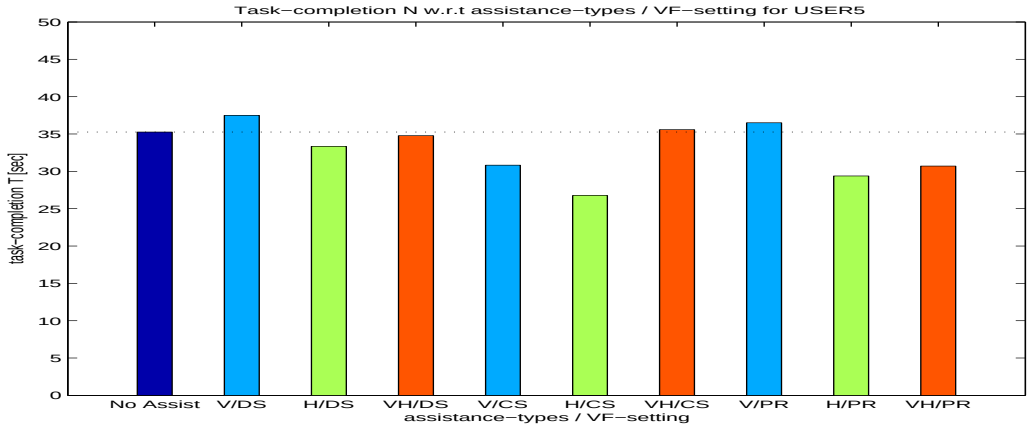


(c) USER3

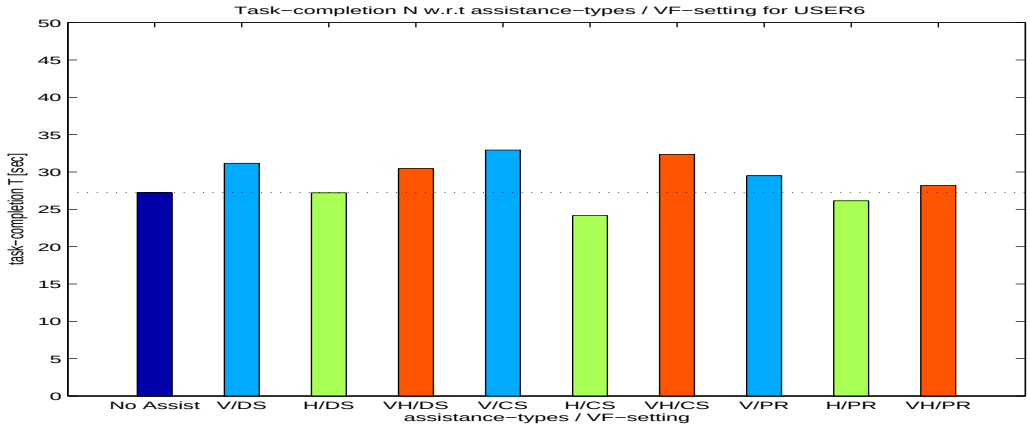
Figure B.2: Task-completion time in [second] with respect to feedback types for USER1 through USER23.



(d) USER4*

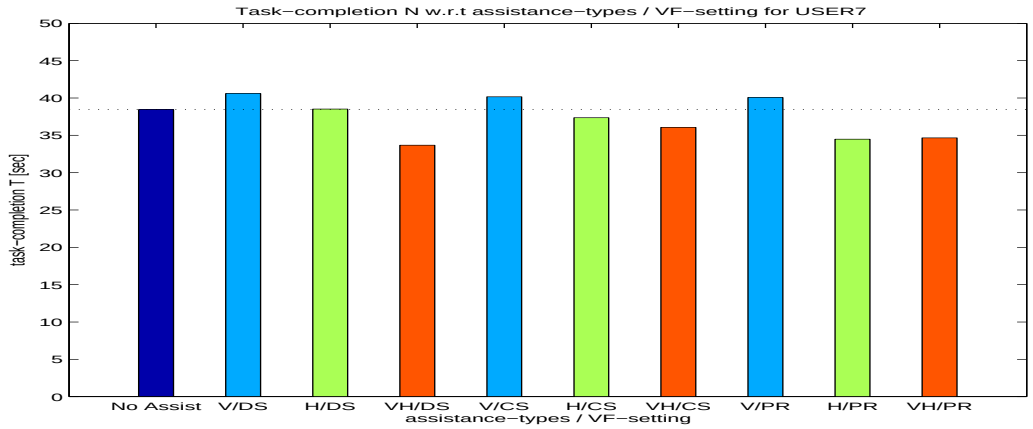


(e) USER5

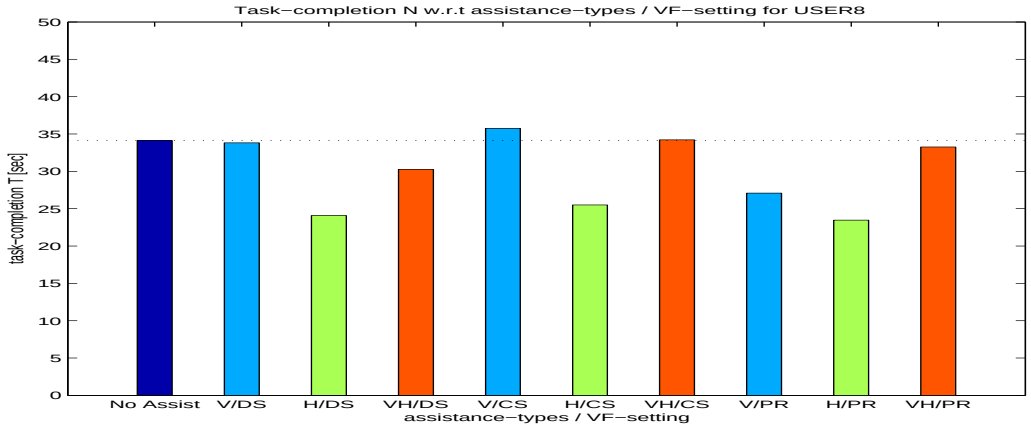


(f) USER6

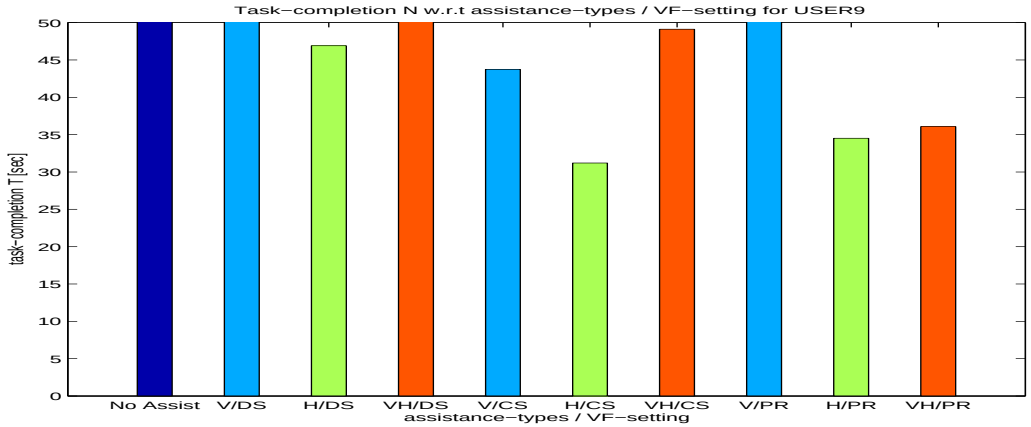
Figure B.2: Continued.



(g) USER7

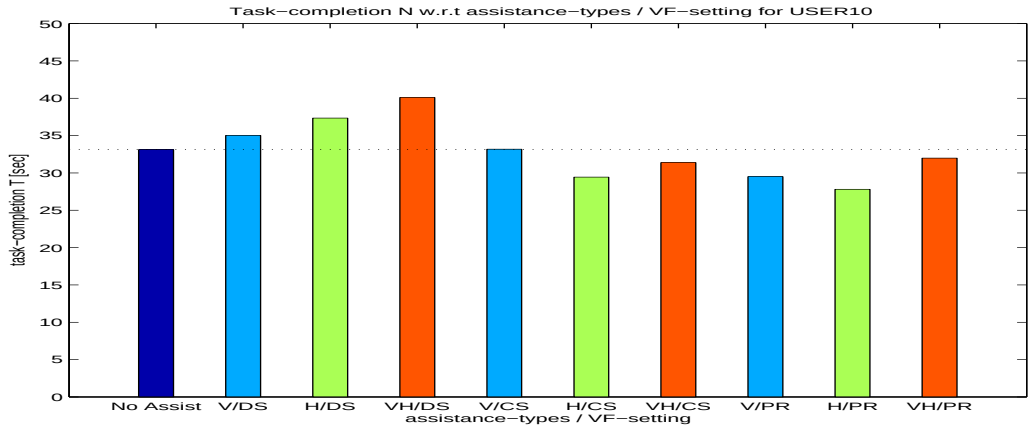


(h) USER8

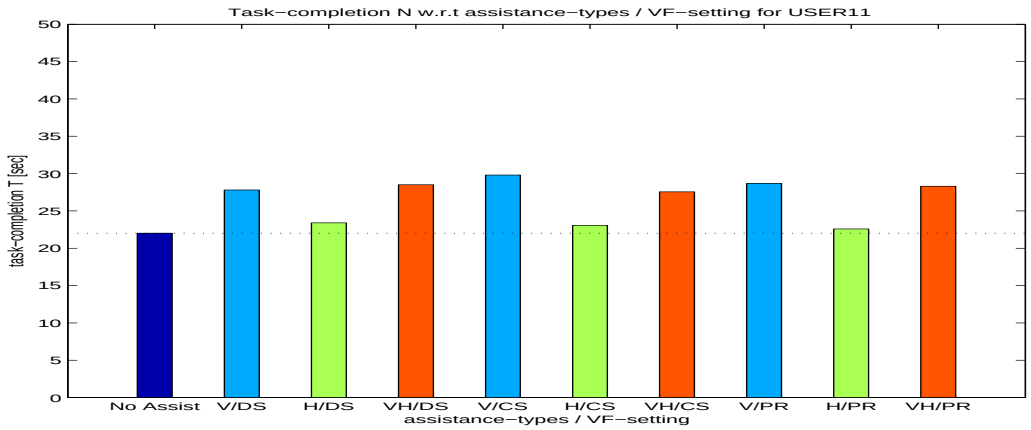


(i) USER9‡

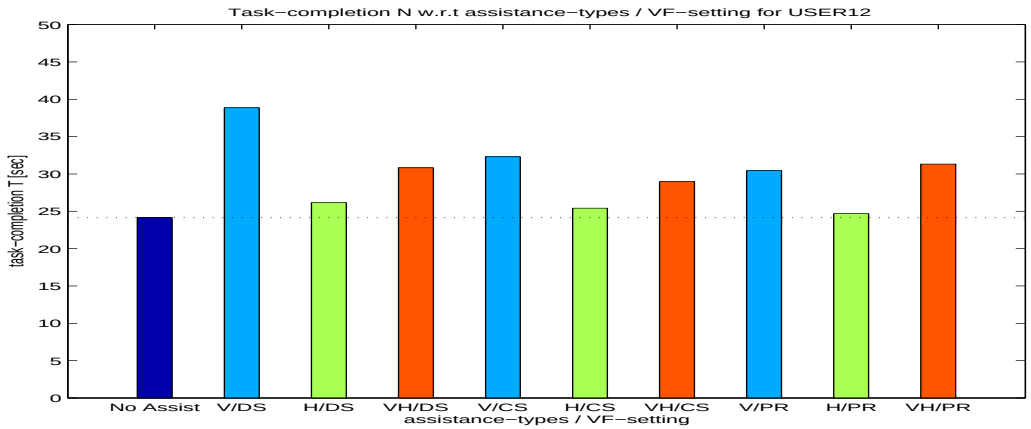
Figure B.2: Continued.



(j) USER10

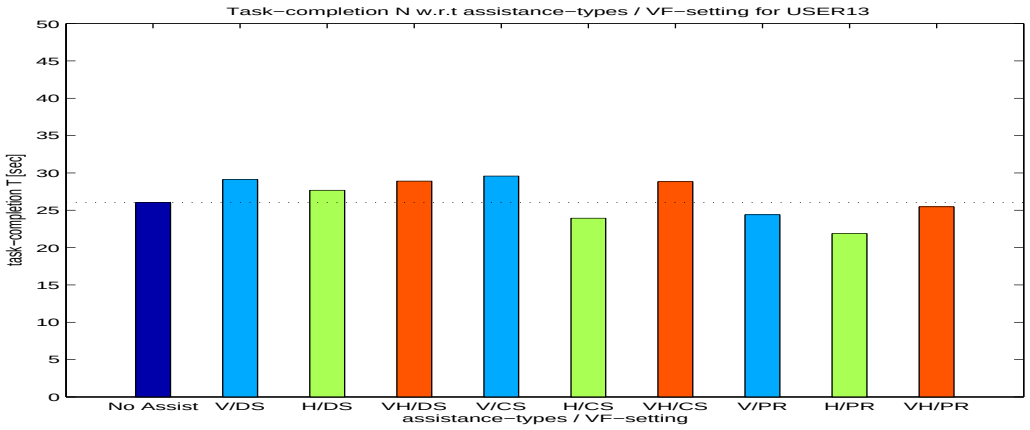


(k) USER11

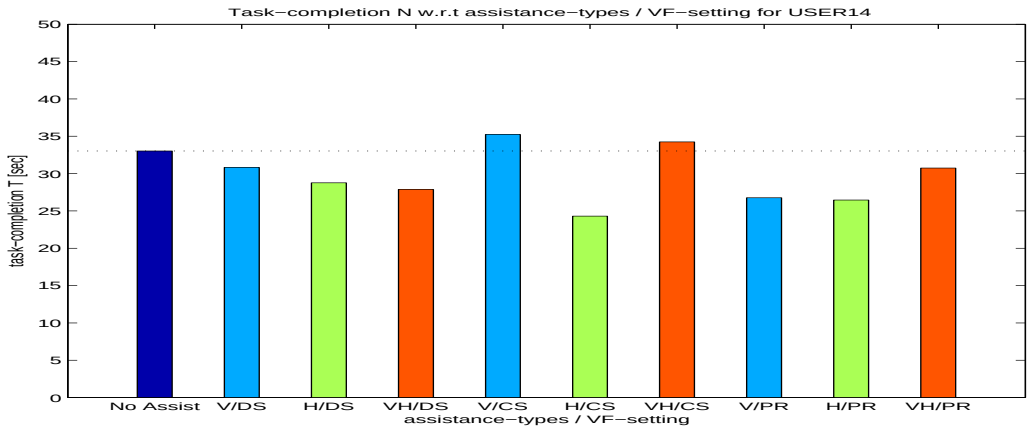


(l) USER12

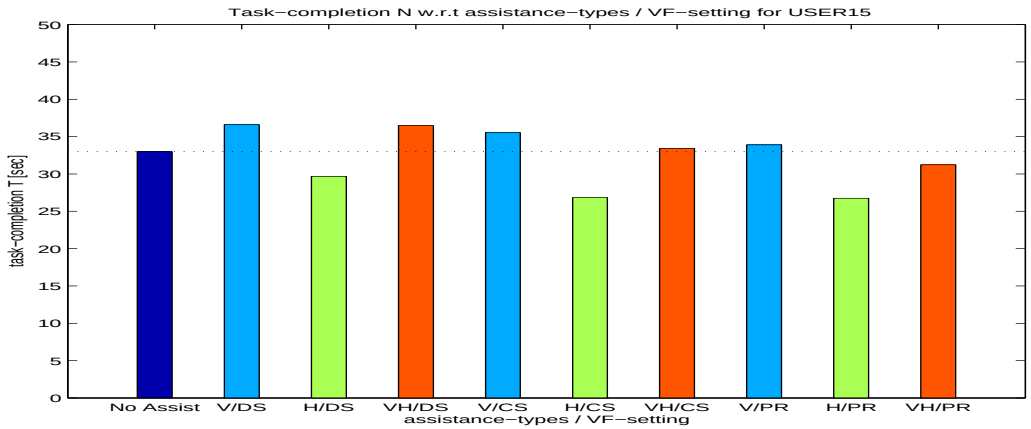
Figure B.2: Continued.



(m) USER13

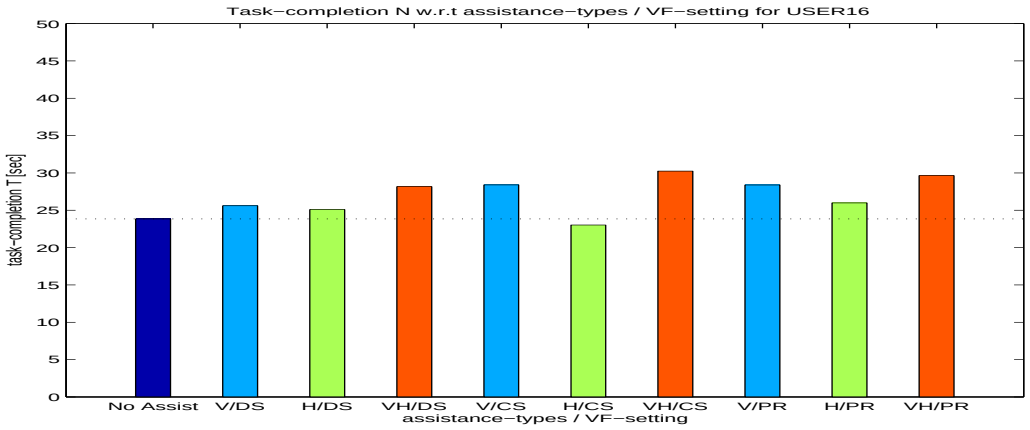


(n) USER14

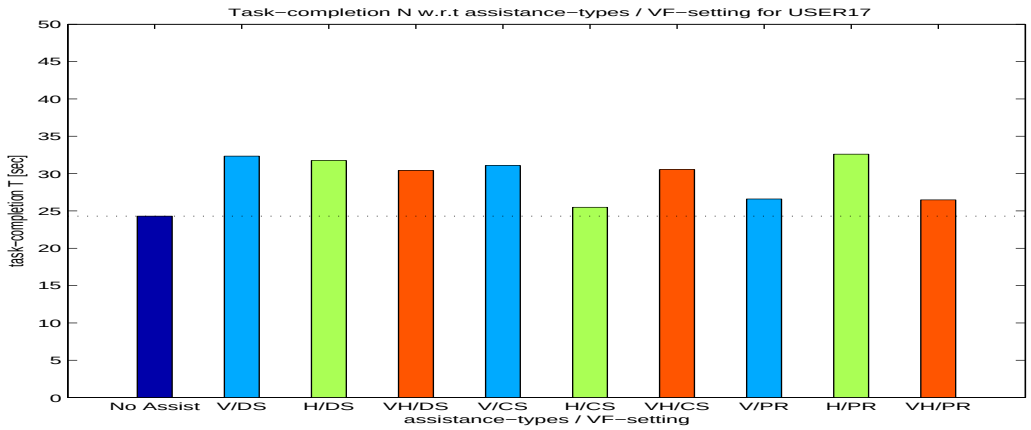


(o) USER15

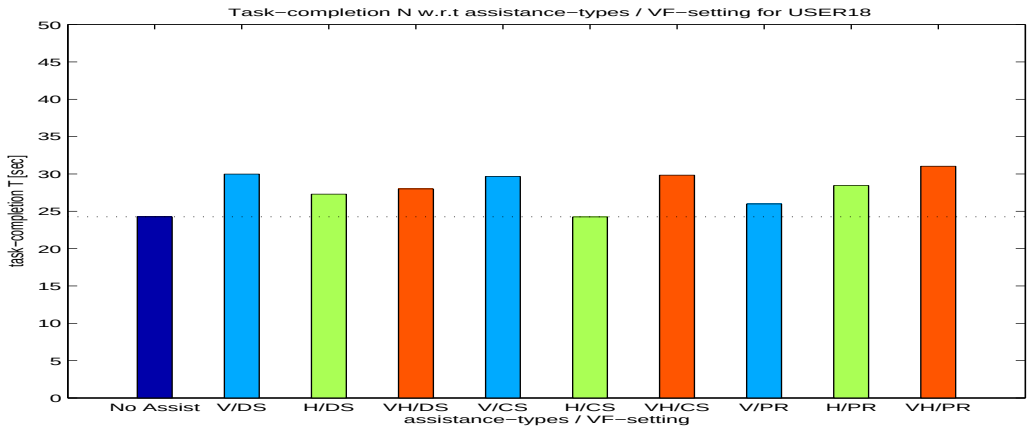
Figure B.2: Continued.



(p) USER16

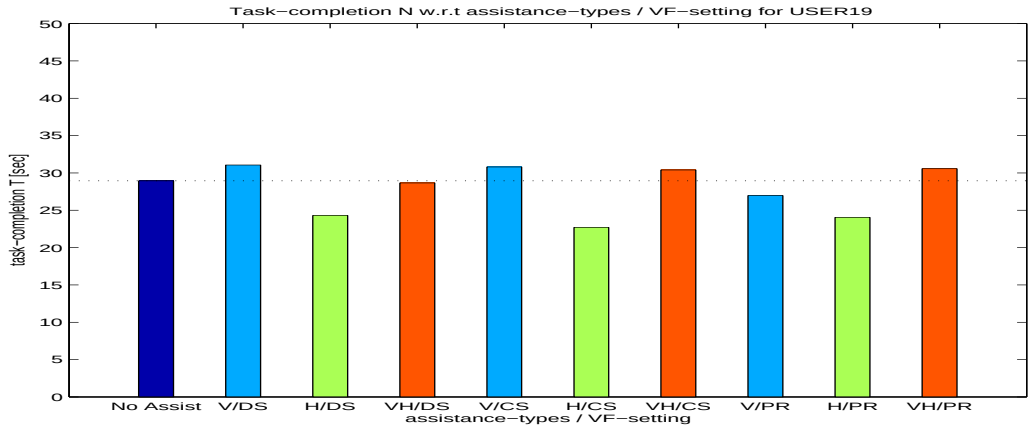


(q) USER17

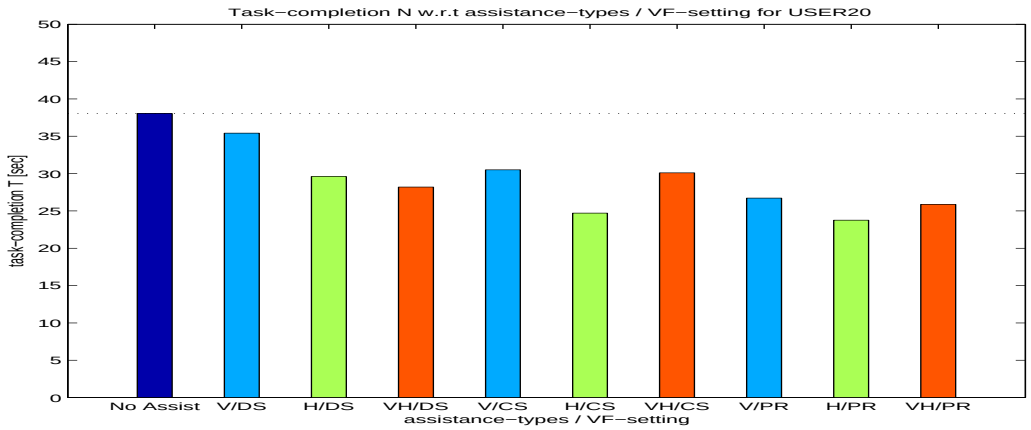


(r) USER18

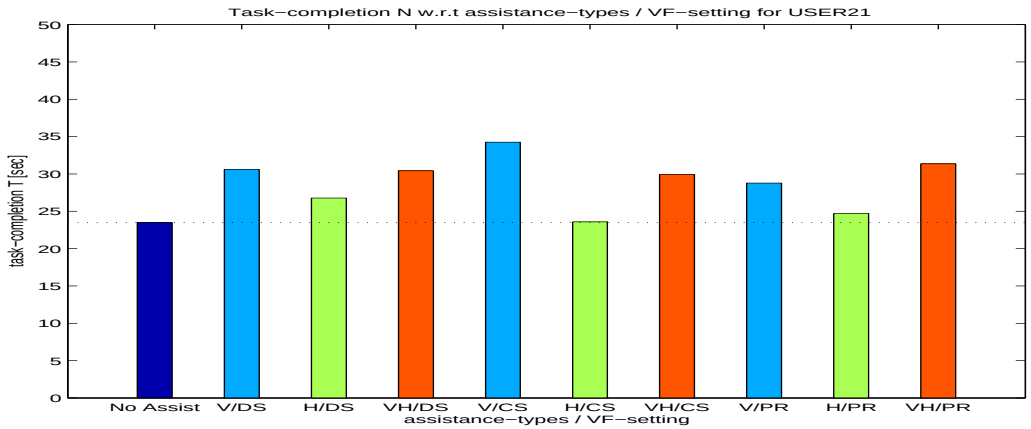
Figure B.2: Continued.



(s) USER19

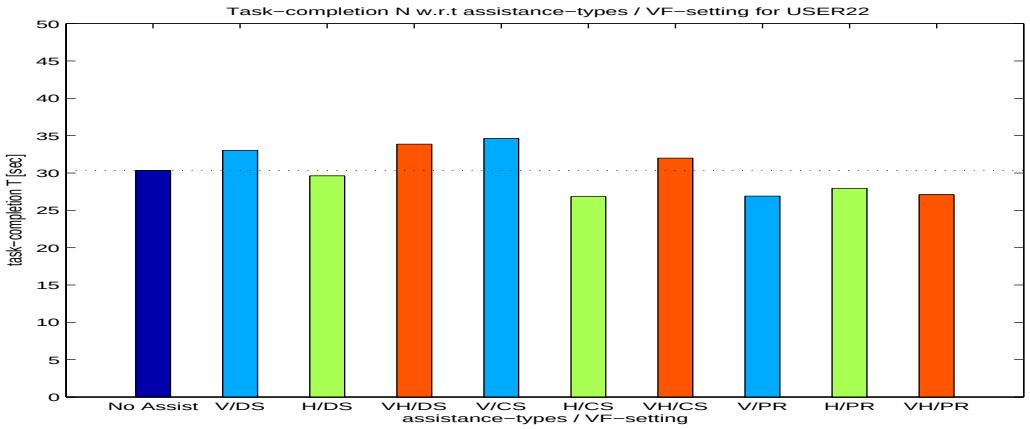


(t) USER20

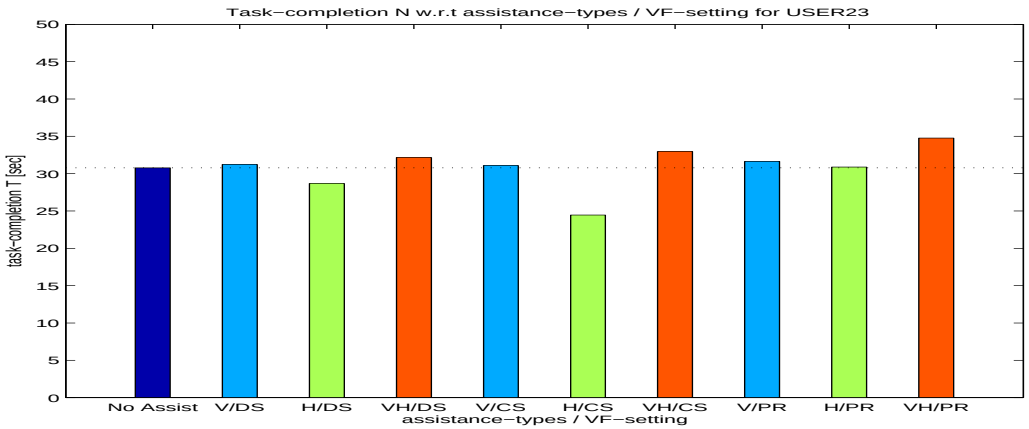


(u) USER21

Figure B.2: Continued.

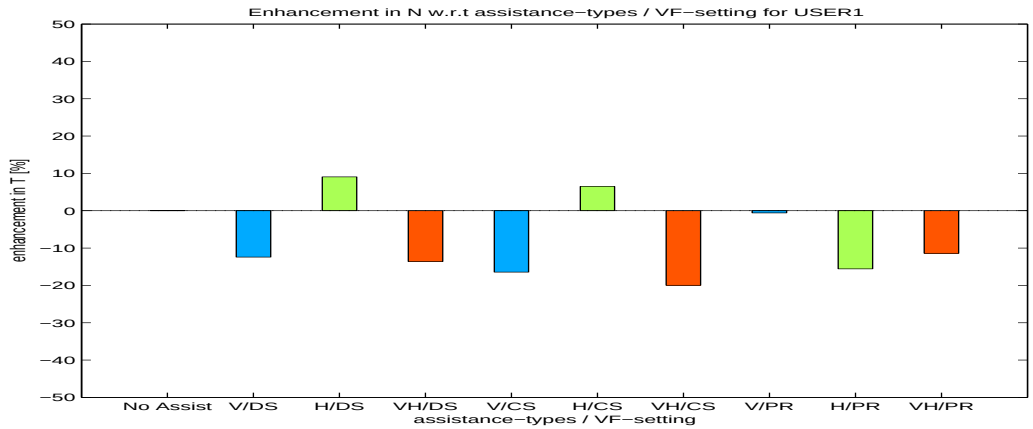


(v) USER22

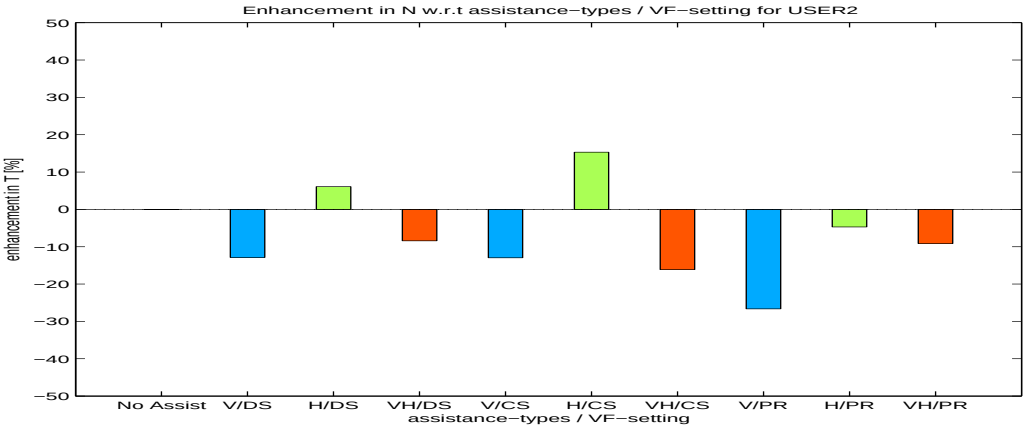


(w) USER23

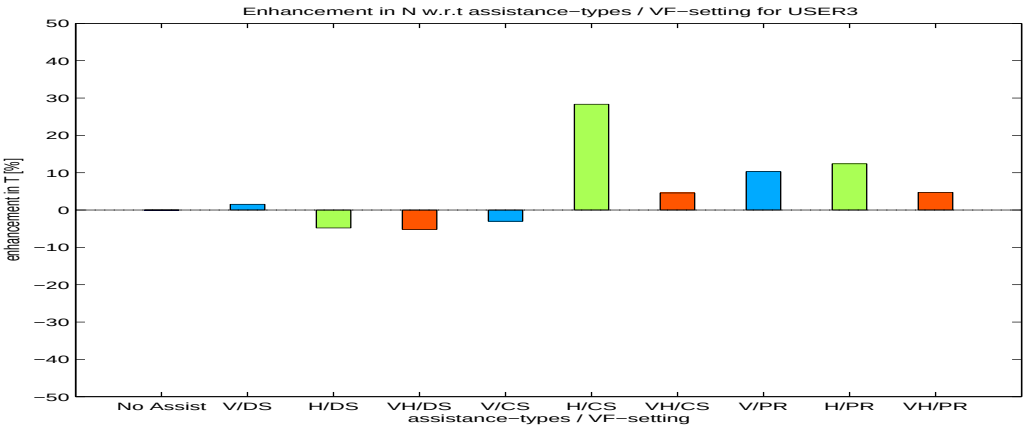
Figure B.2: Continued.



(a) USER1

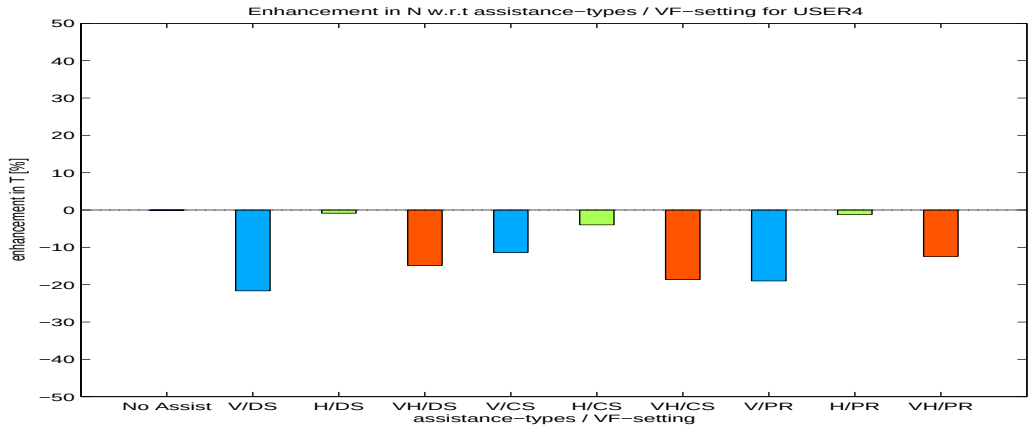


(b) USER2

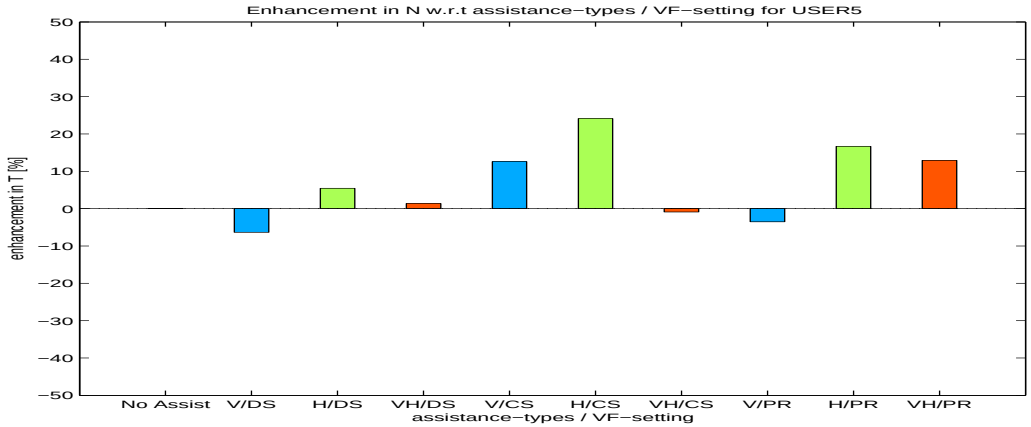


(c) USER3

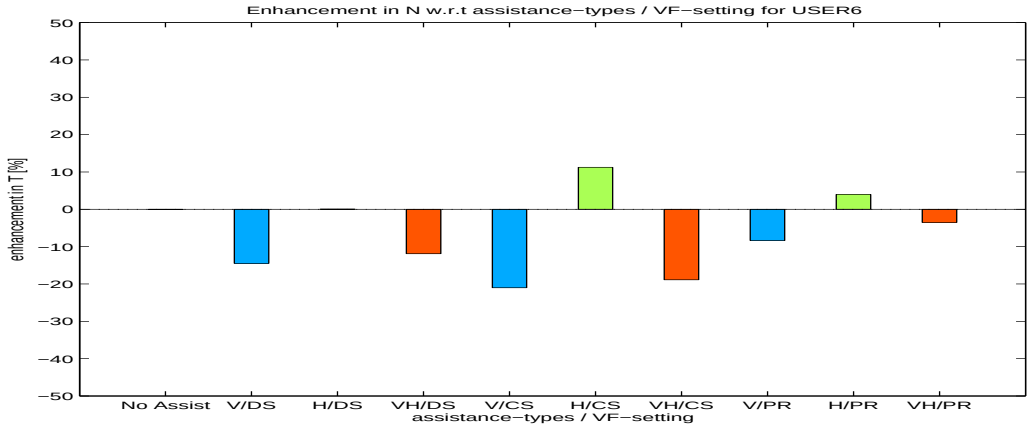
Figure B.3: Performance enhancement in task-completion time as percent with respect to feedback types for USER1 through USER23.



(d) USER4*

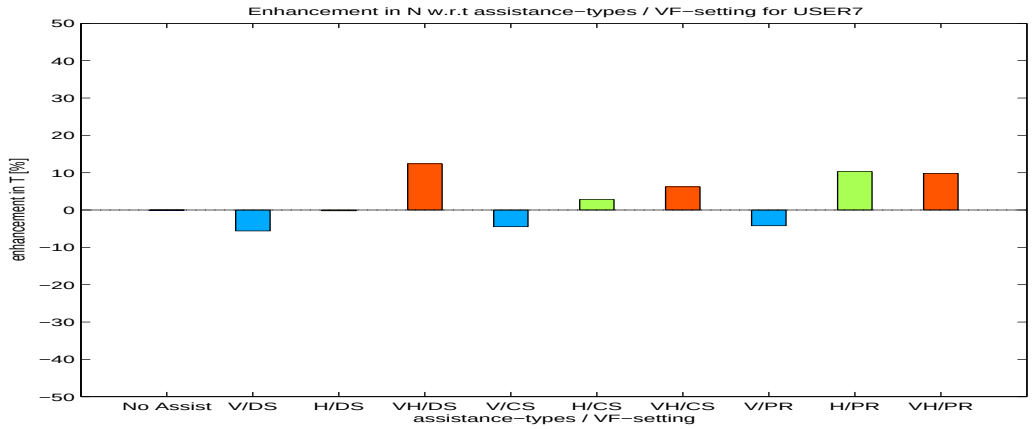


(e) USER5

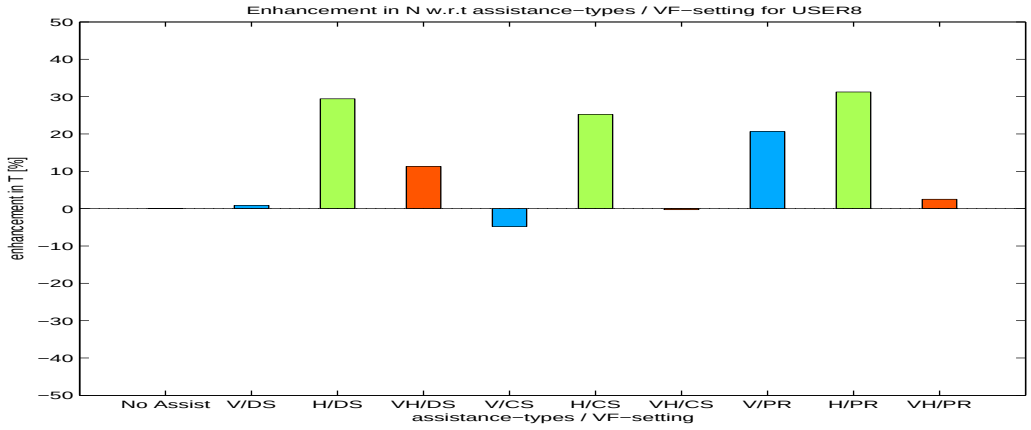


(f) USER6

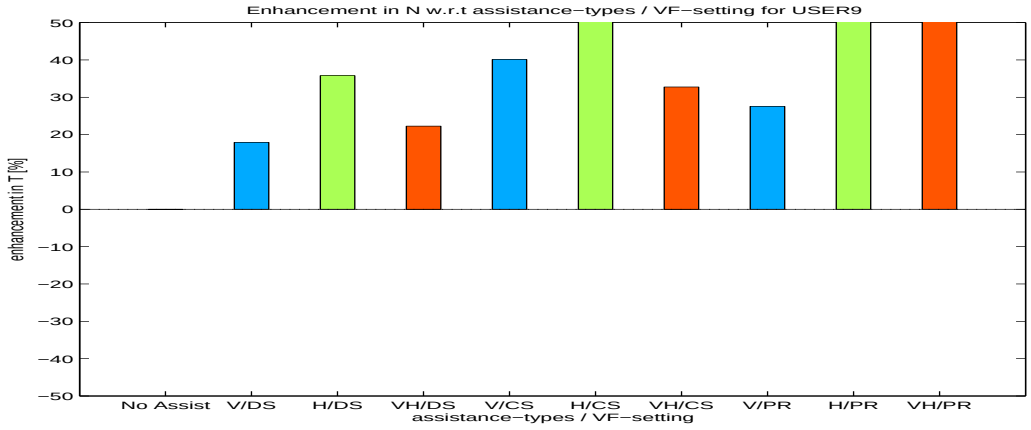
Figure B.3: Continued.



(g) USER7

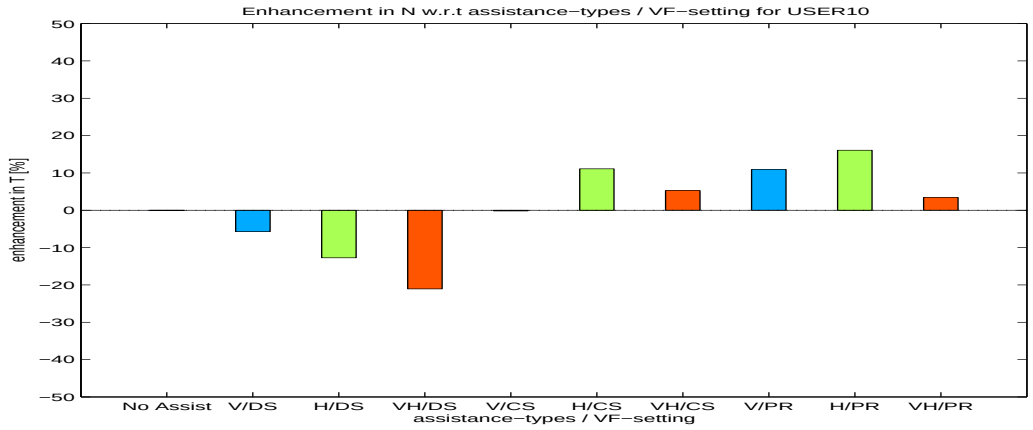


(h) USER8

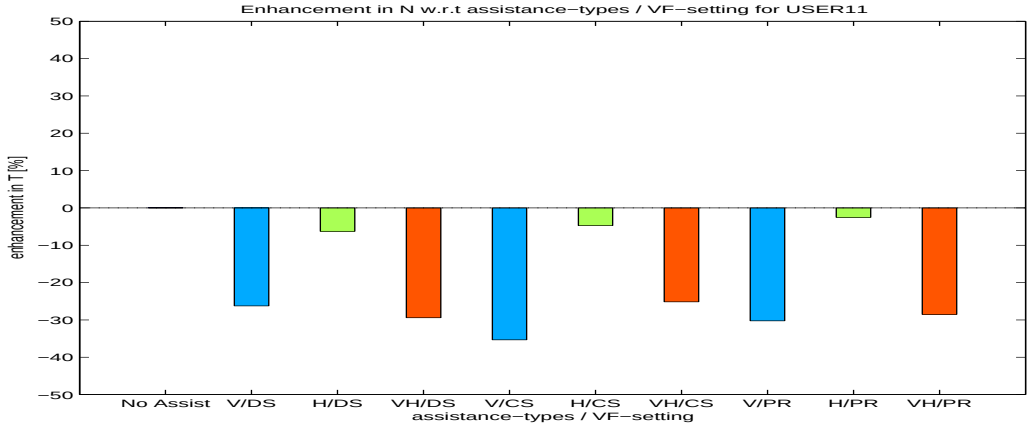


(i) USER9‡

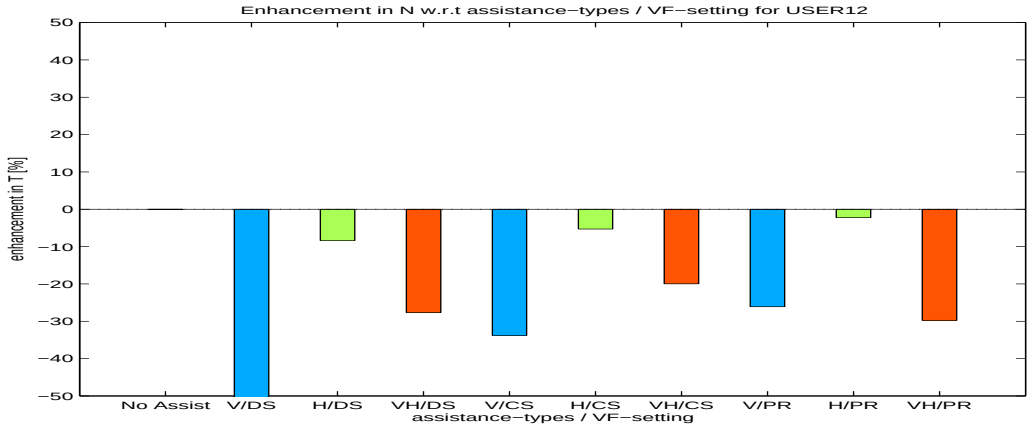
Figure B.3: Continued.



(j) USER10

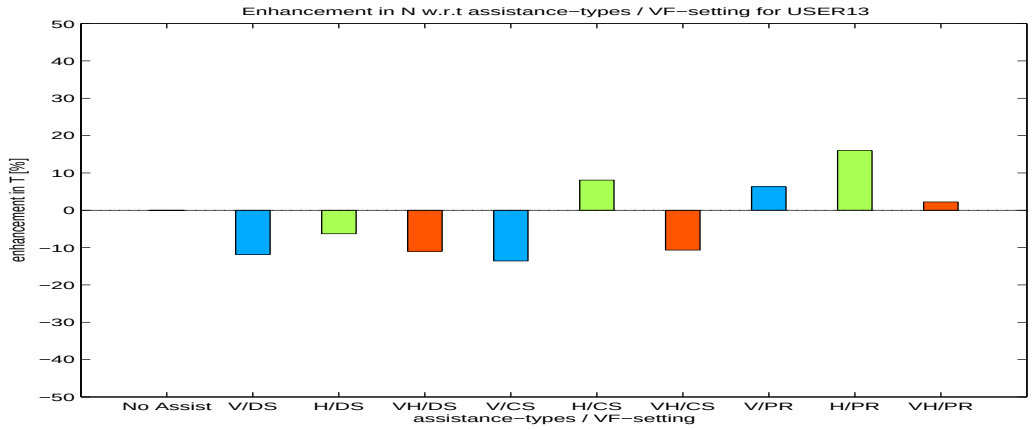


(k) USER11

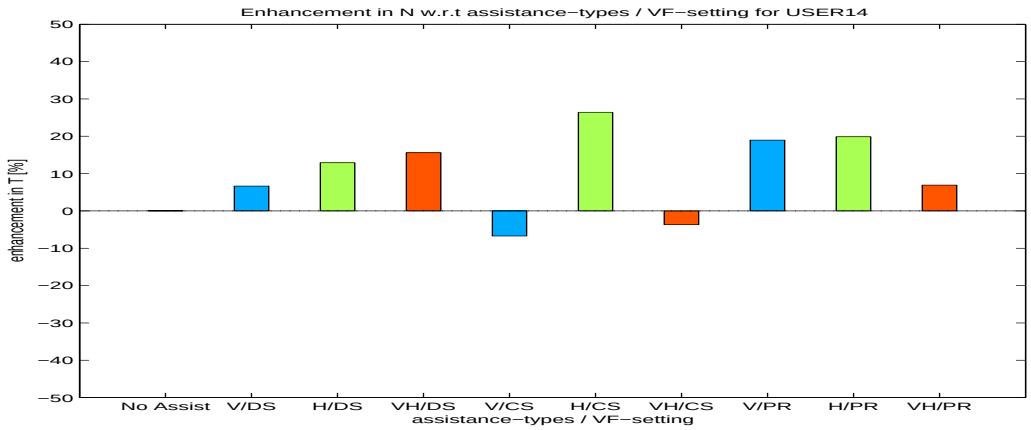


(l) USER12

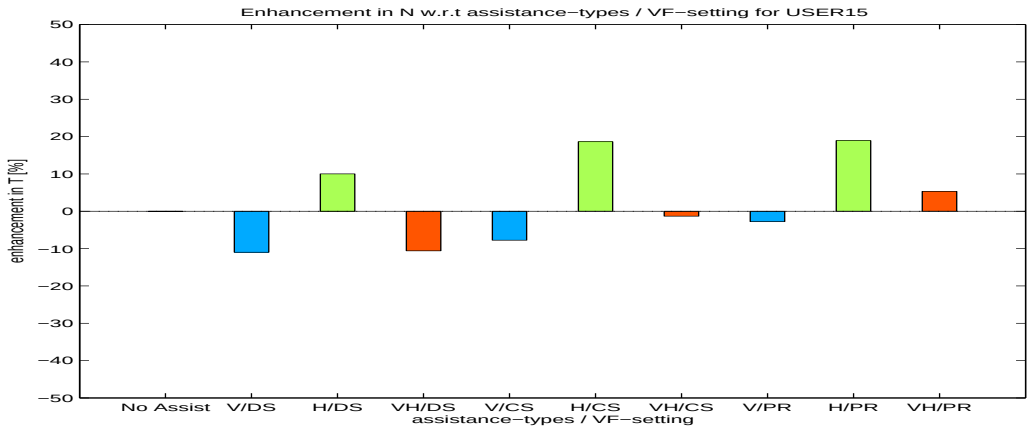
Figure B.3: Continued.



(m) USER13

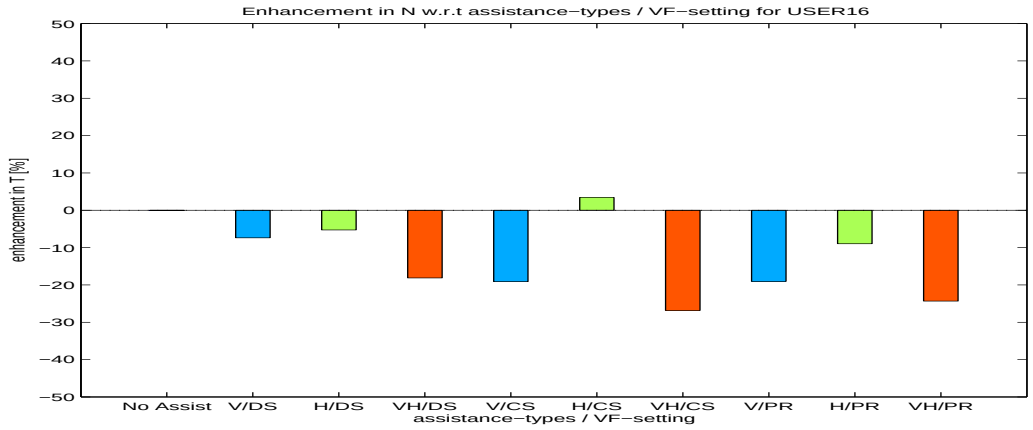


(n) USER14

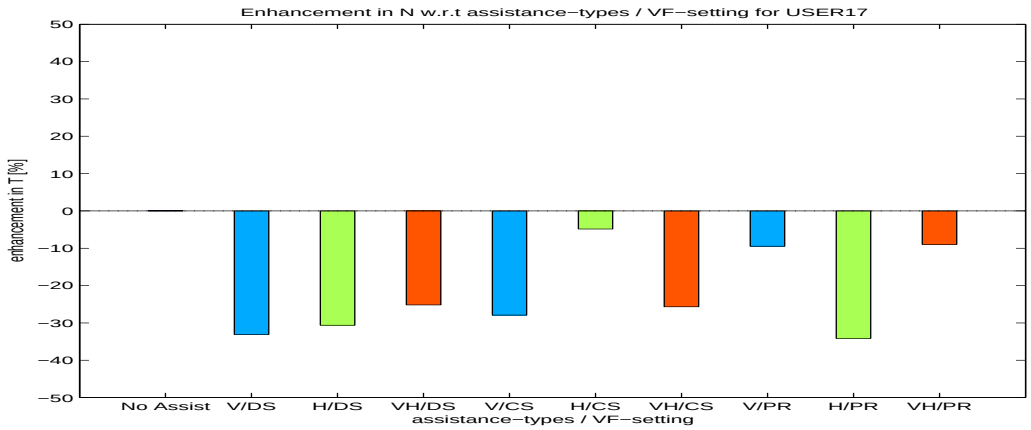


(o) USER15

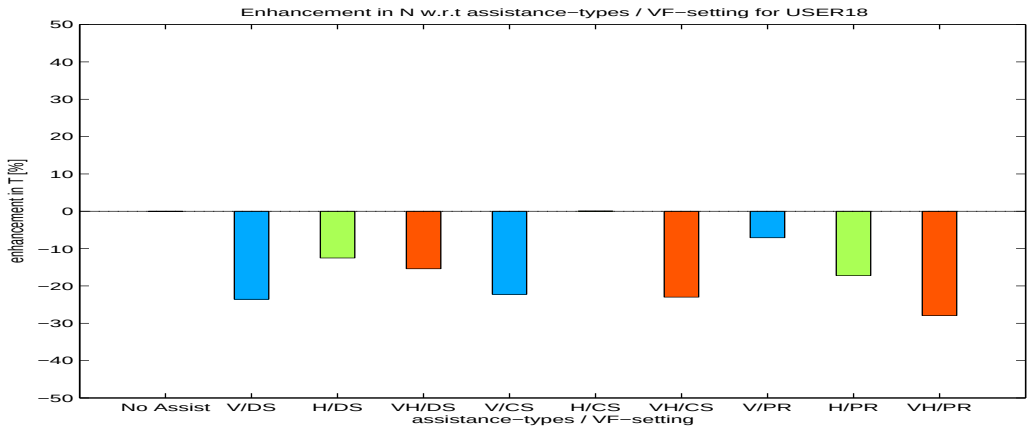
Figure B.3: Continued.



(p) USER16

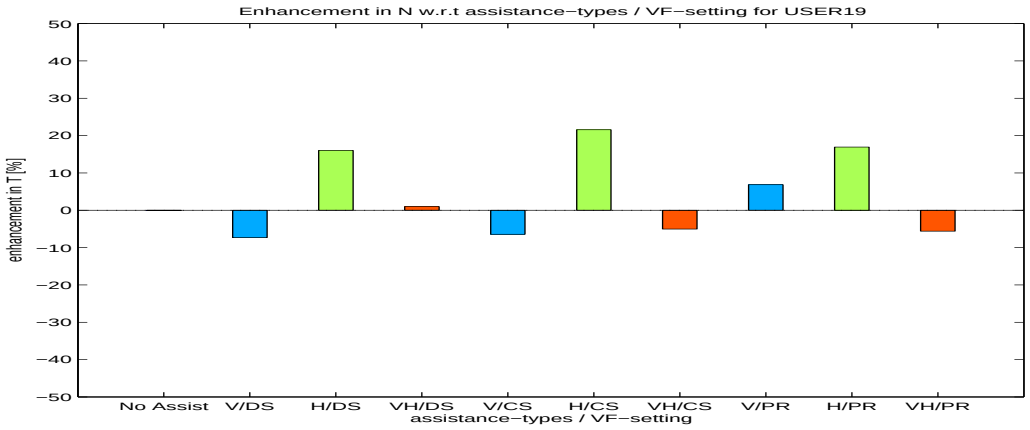


(q) USER17

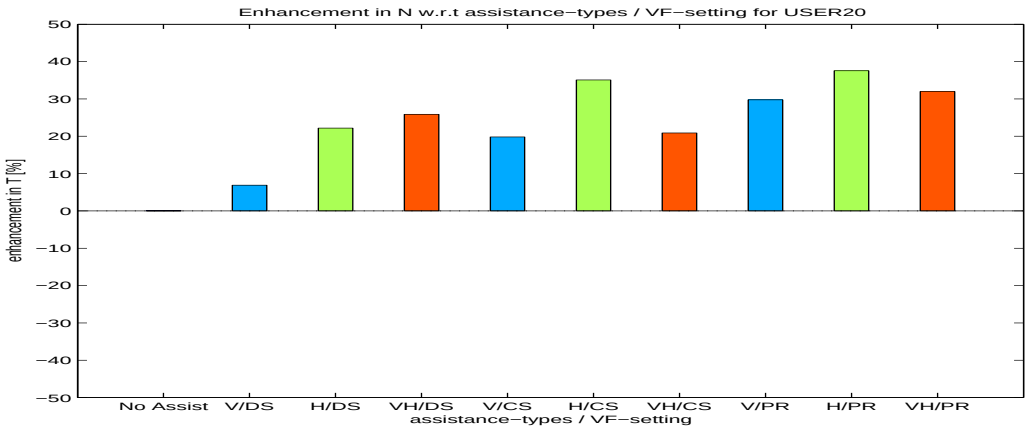


(r) USER18

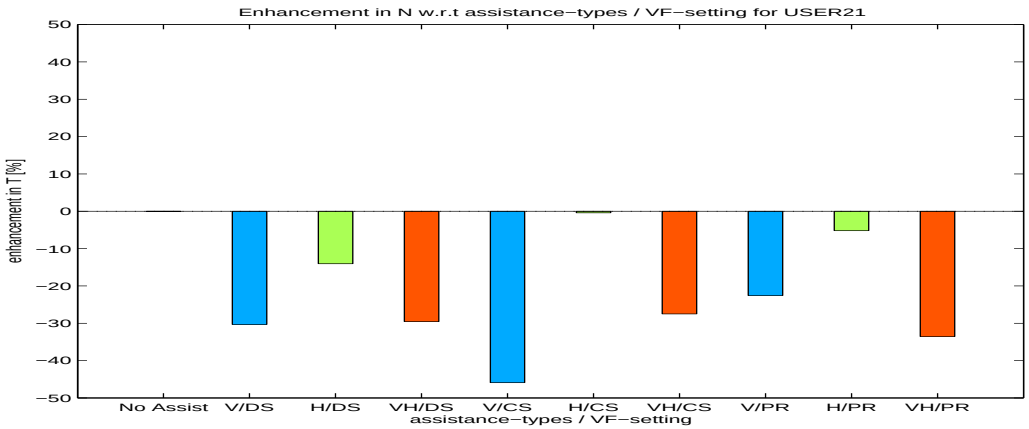
Figure B.3: Continued.



(s) USER19

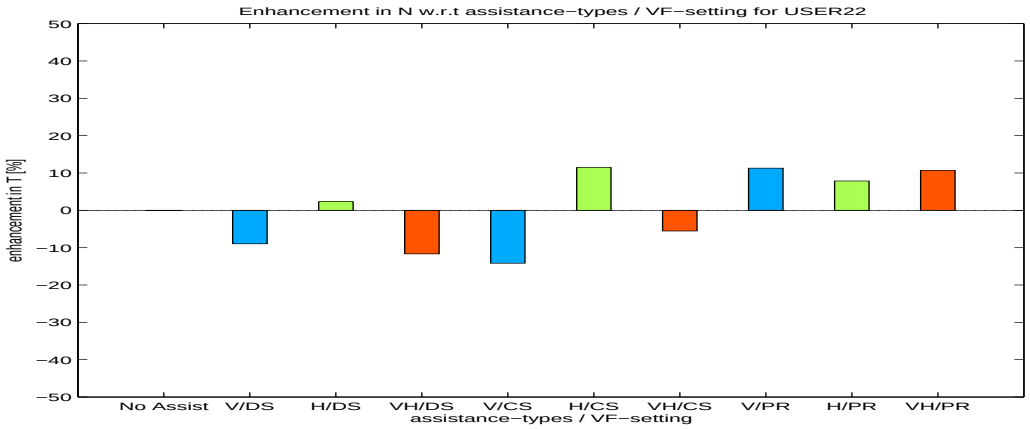


(t) USER20

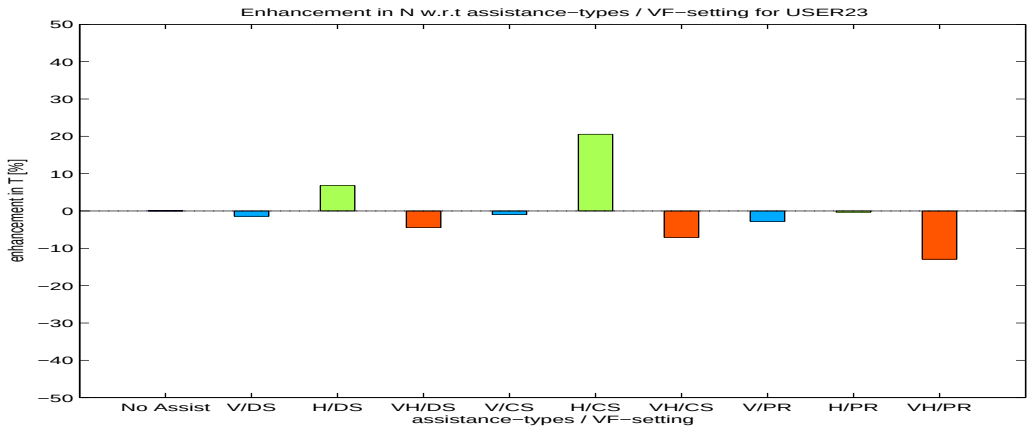


(u) USER21

Figure B.3: Continued.

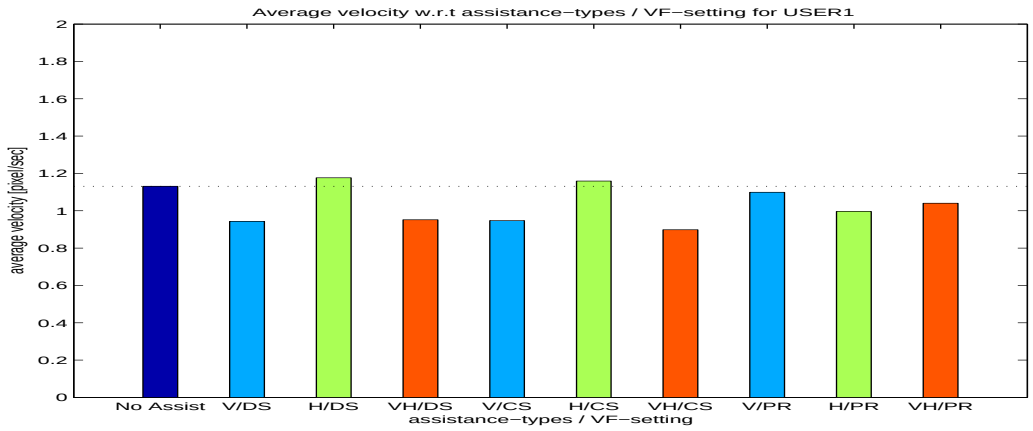


(v) USER22

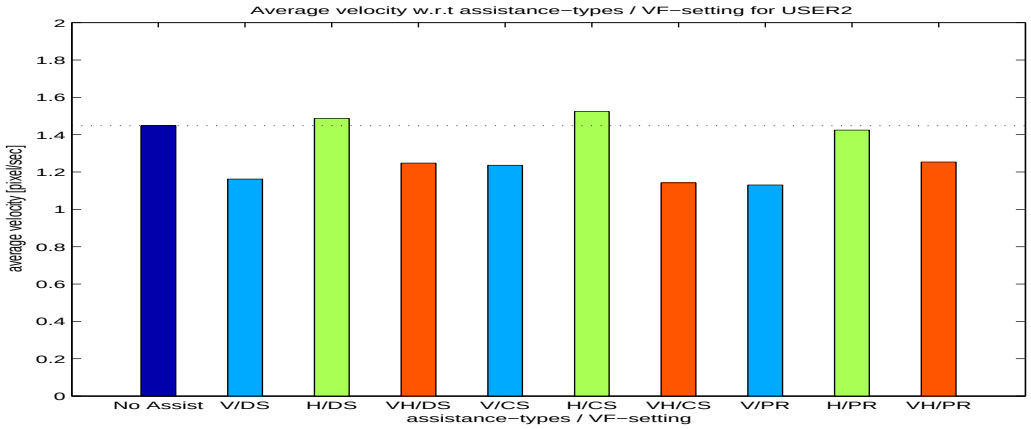


(w) USER23

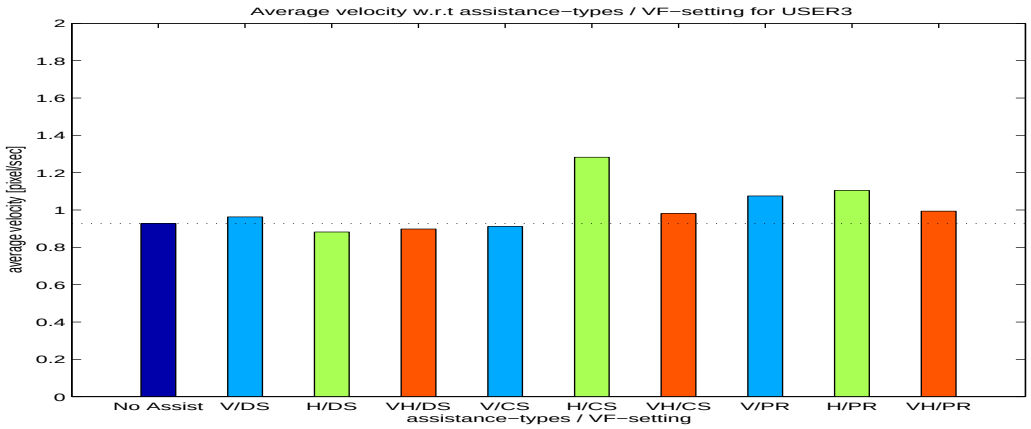
Figure B.3: Continued.



(a) USER1

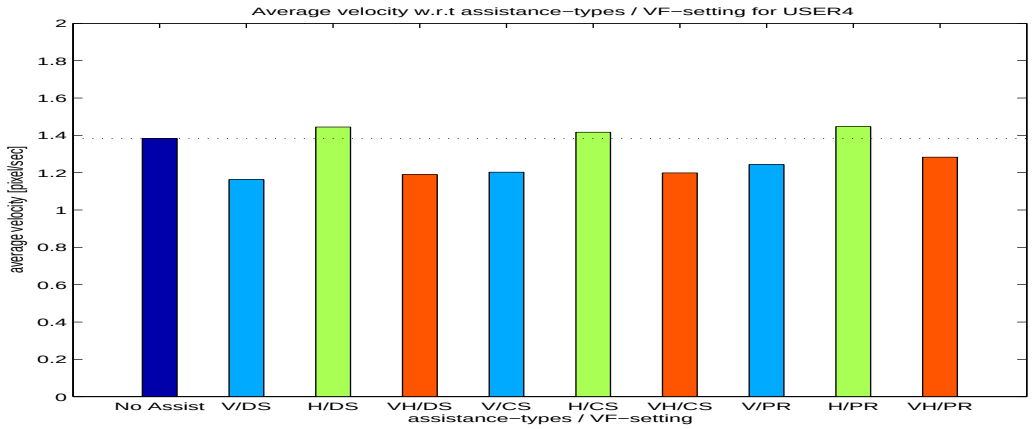


(b) USER2

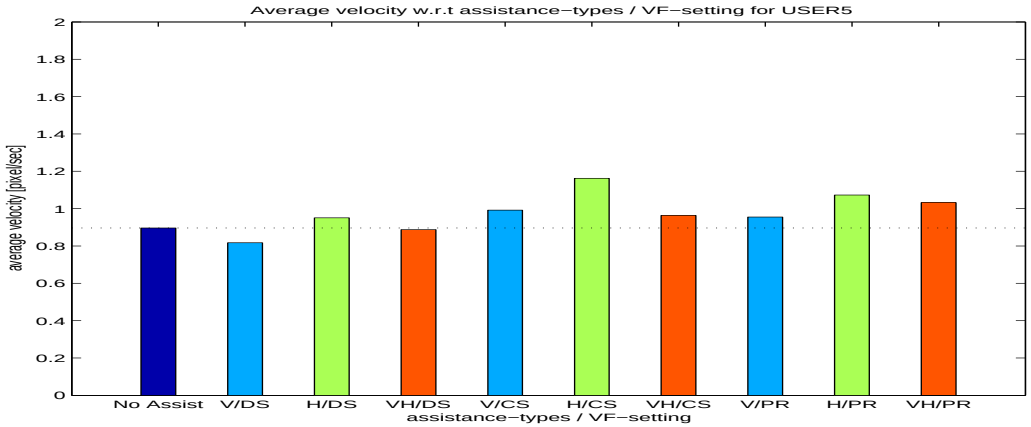


(c) USER3

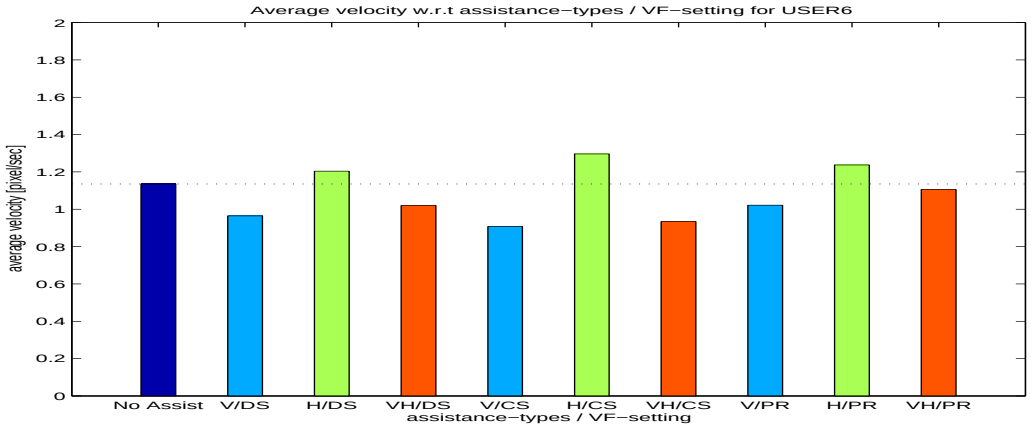
Figure B.4: Average velocity in [pixel/second] with respect to feedback types for USER1 through USER23.



(d) USER4*

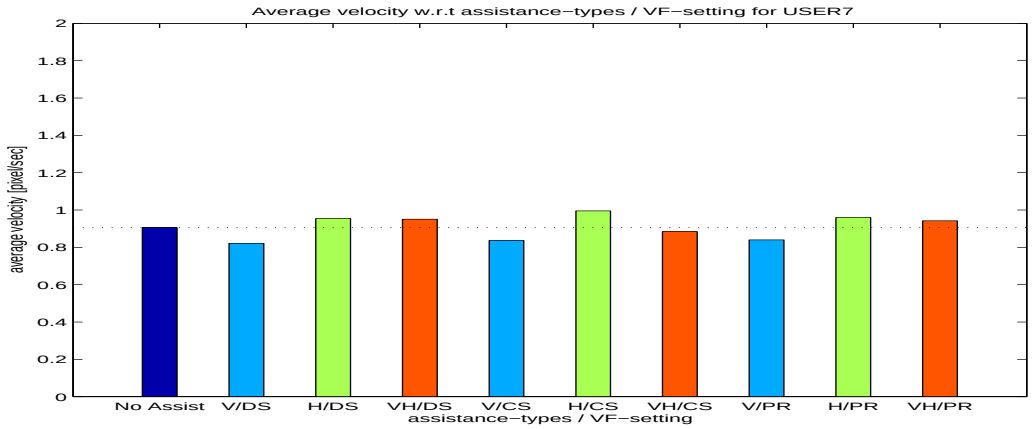


(e) USER5

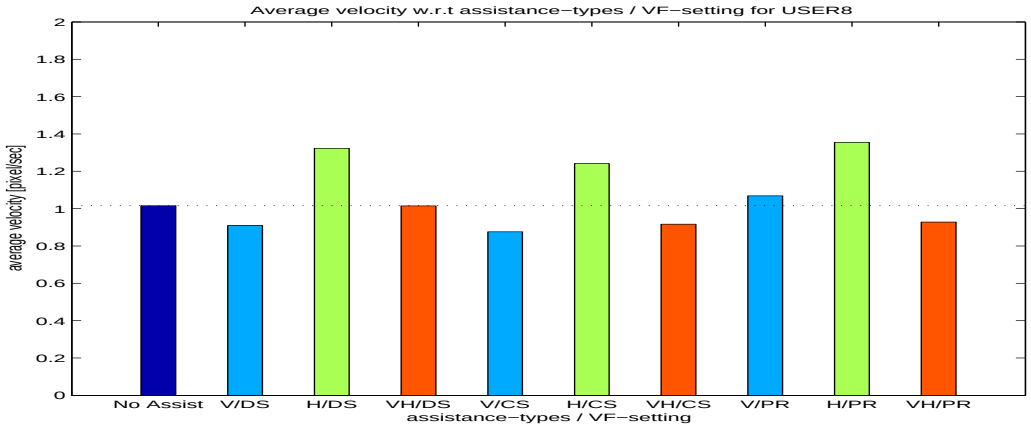


(f) USER6

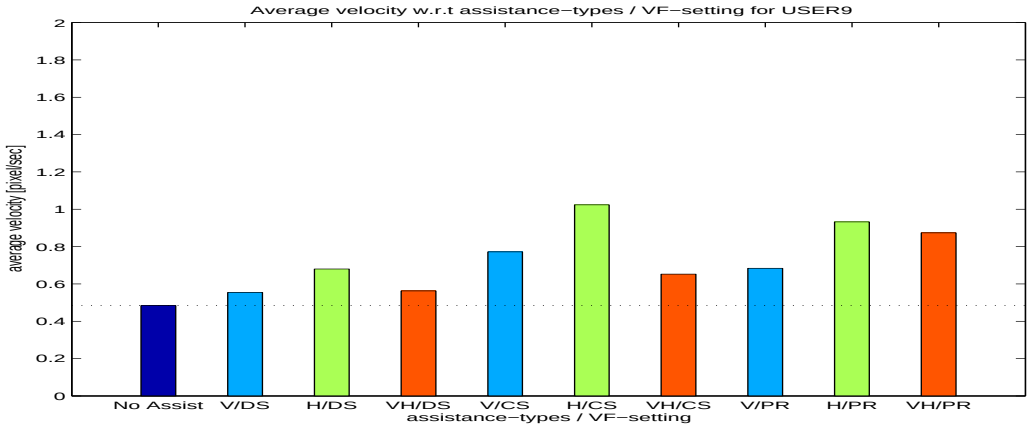
Figure B.4: Continued.



(g) USER7

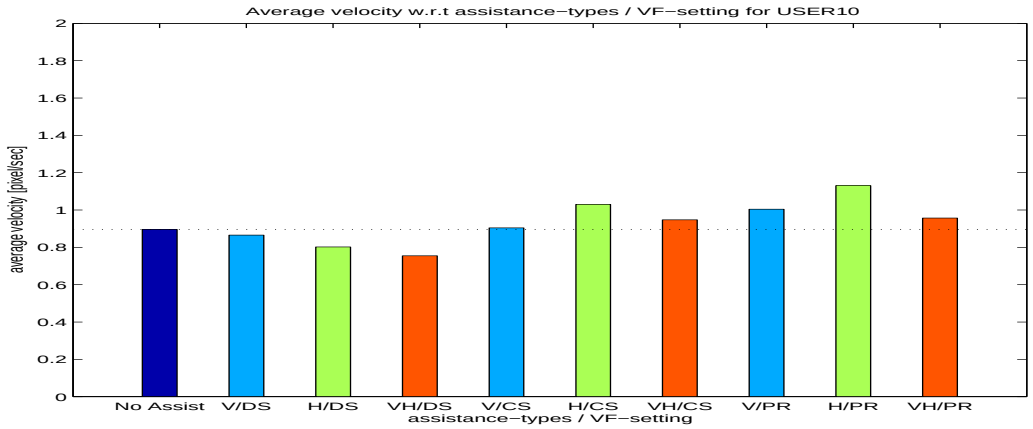


(h) USER8

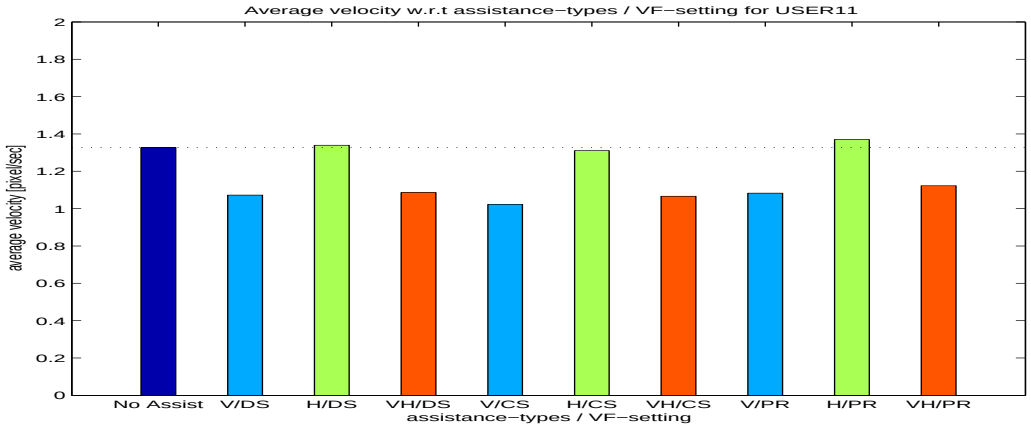


(i) USER9‡

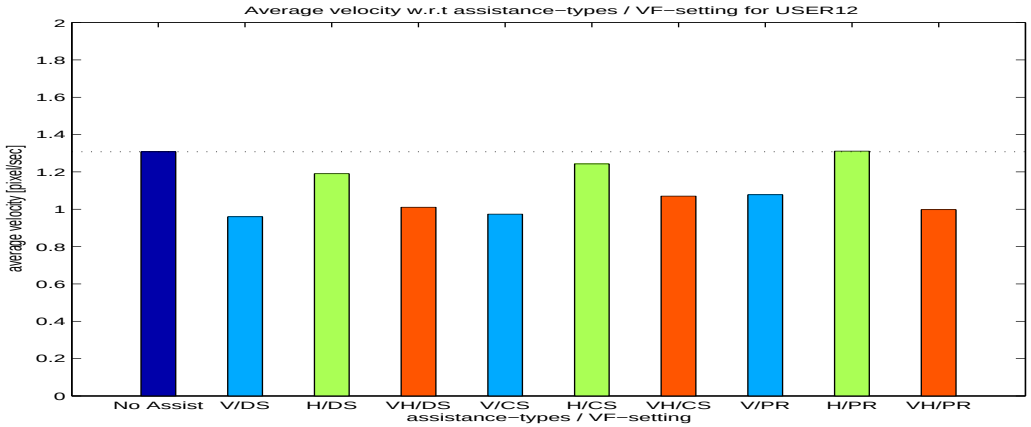
Figure B.4: Continued.



(j) USER10

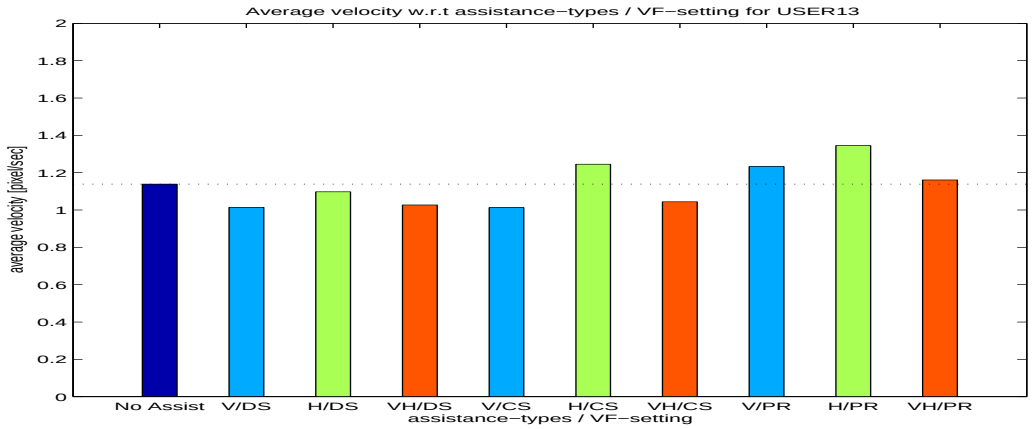


(k) USER11

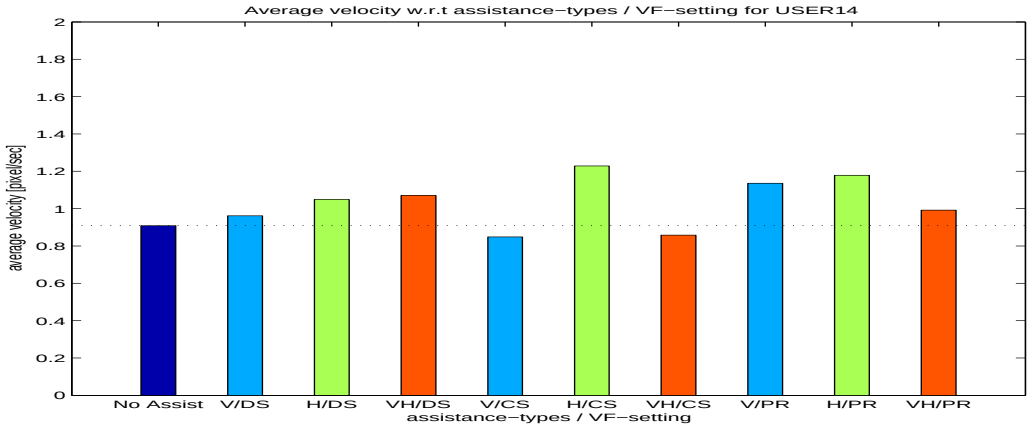


(l) USER12

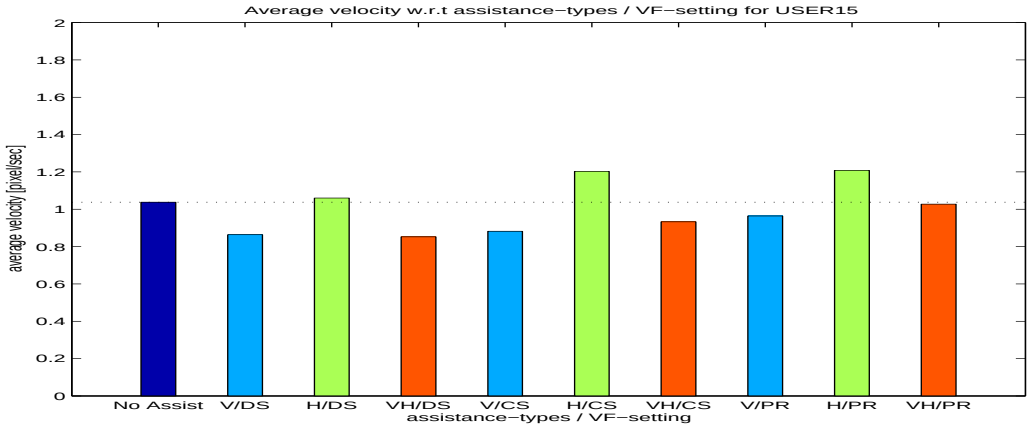
Figure B.4: Continued.



(m) USER13

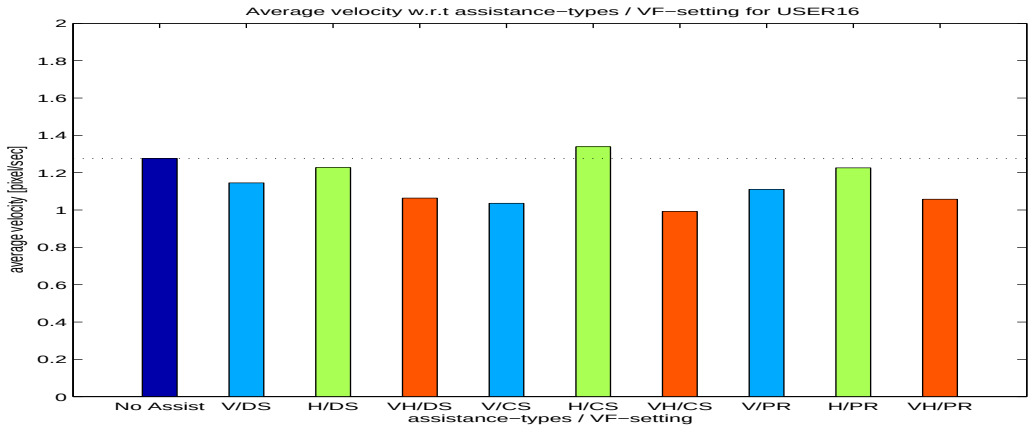


(n) USER14

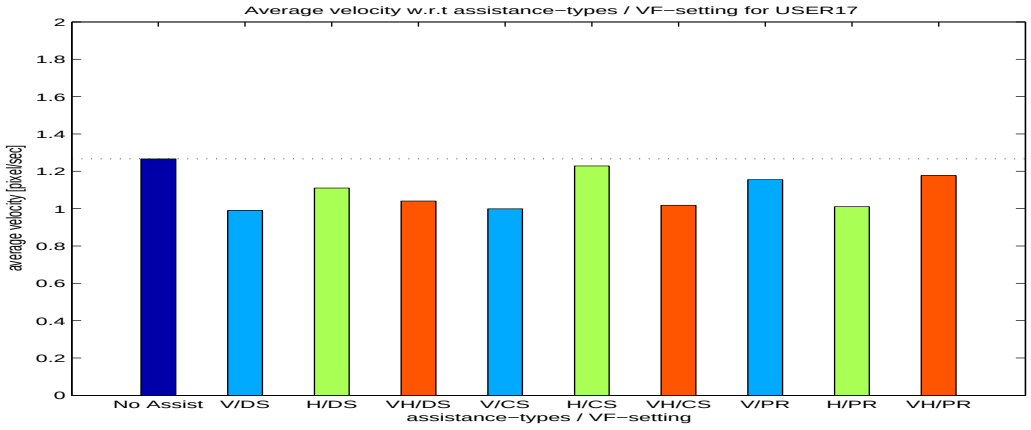


(o) USER15

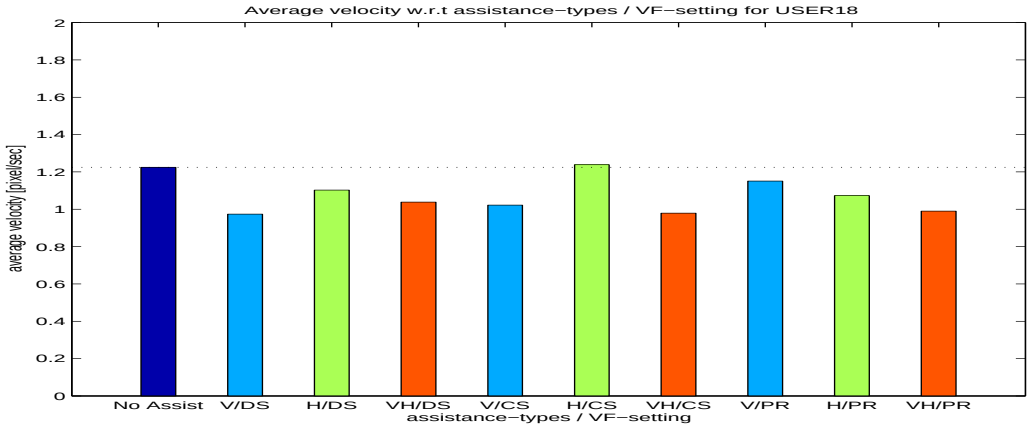
Figure B.4: Continued.



(p) USER16

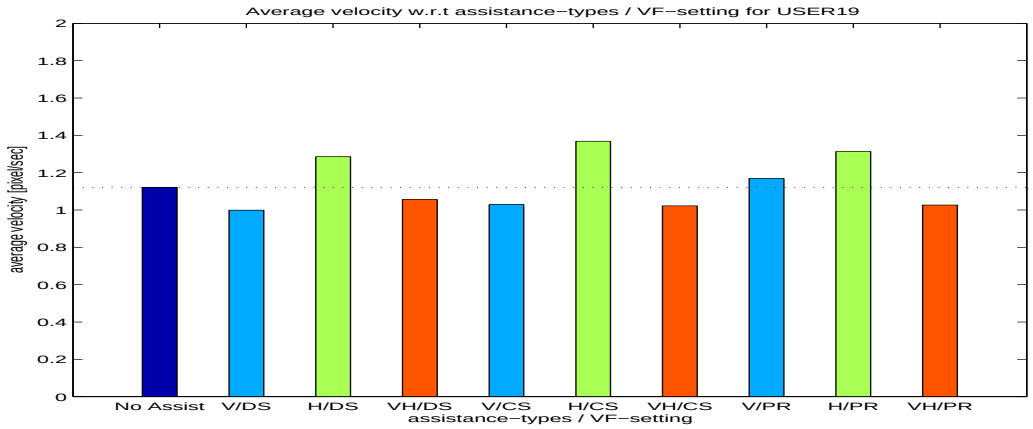


(q) USER17

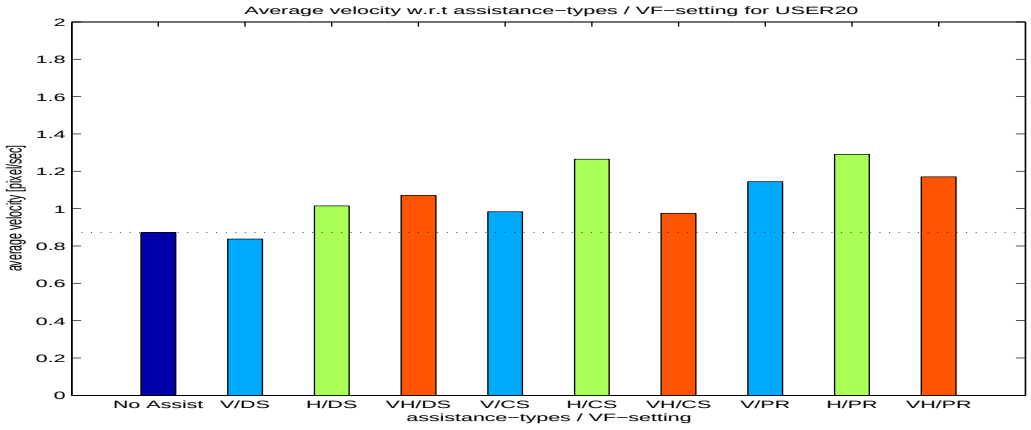


(r) USER18

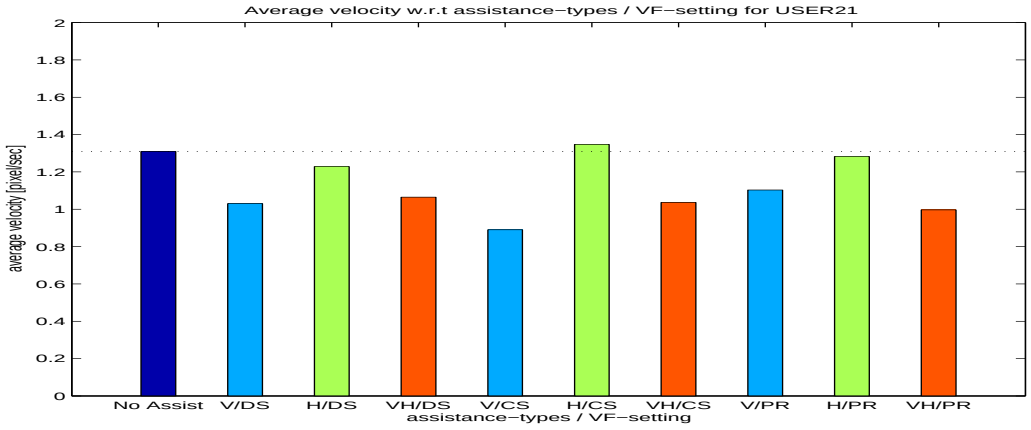
Figure B.4: Continued.



(s) USER19

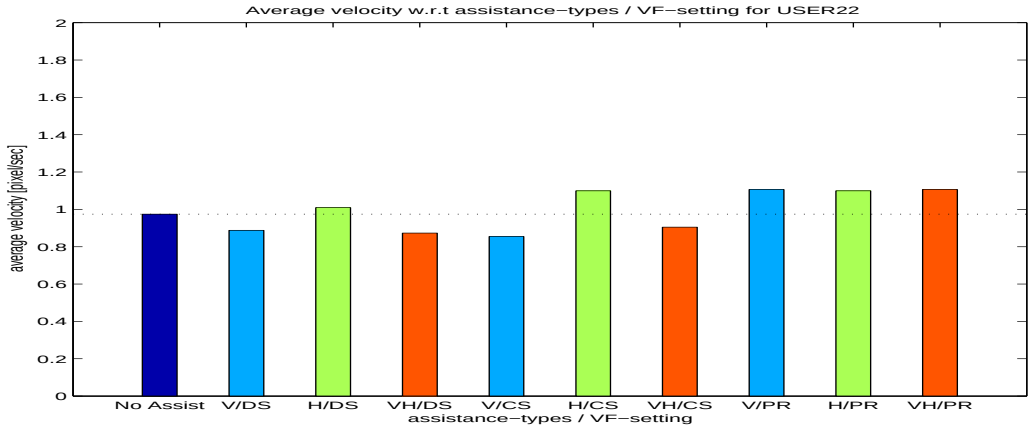


(t) USER20

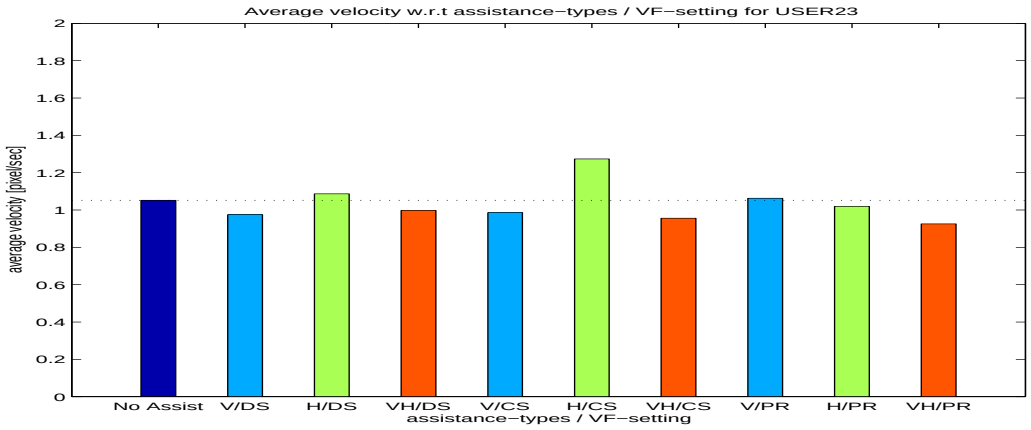


(u) USER21

Figure B.4: Continued.

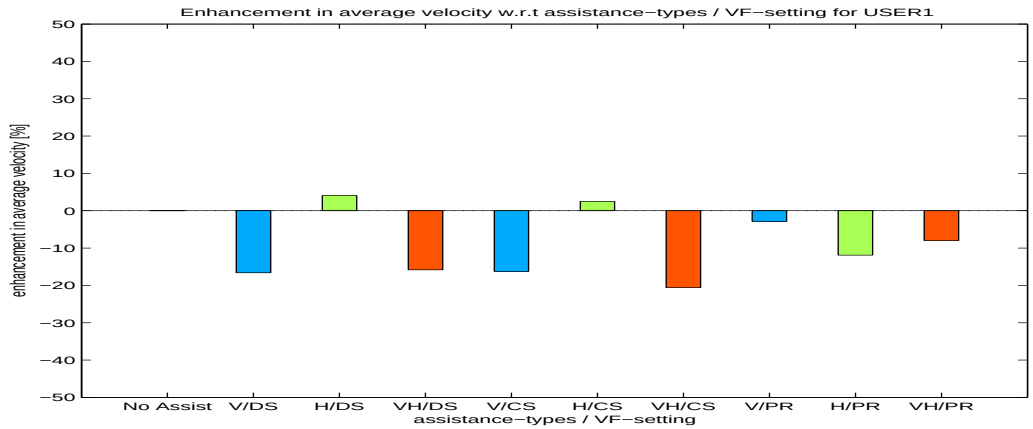


(v) USER22

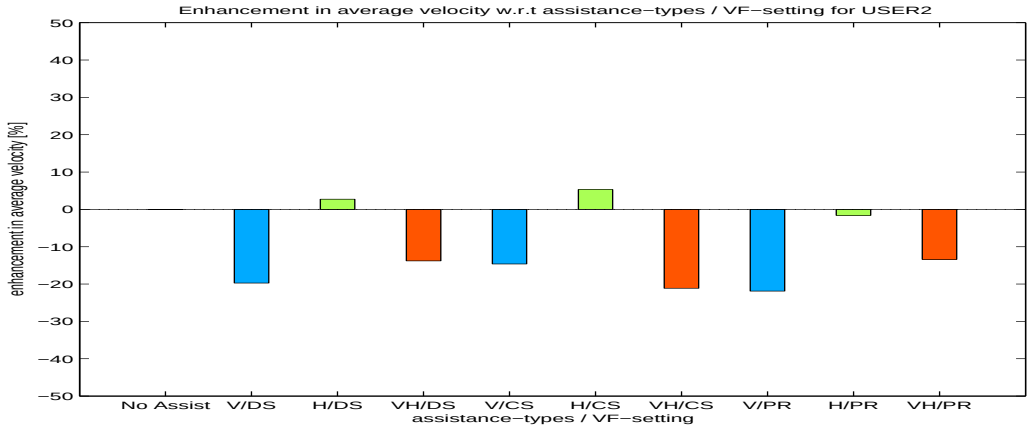


(w) USER23

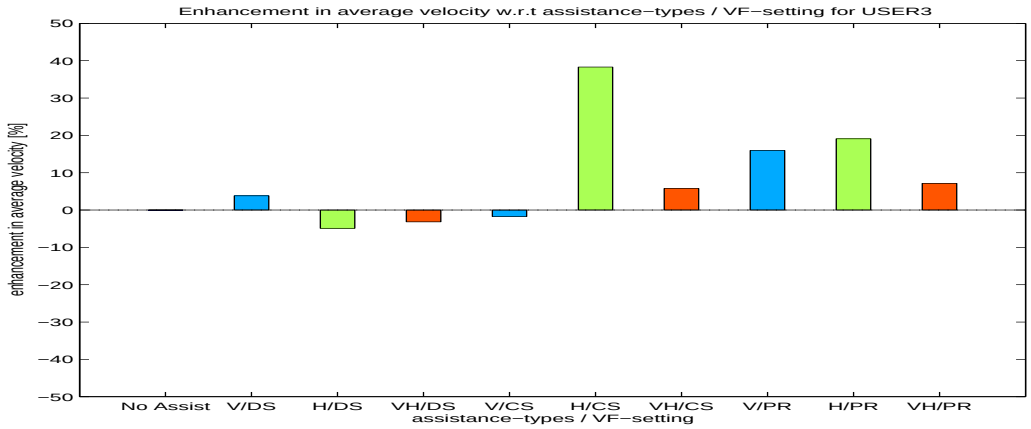
Figure B.4: Continued.



(a) USER1

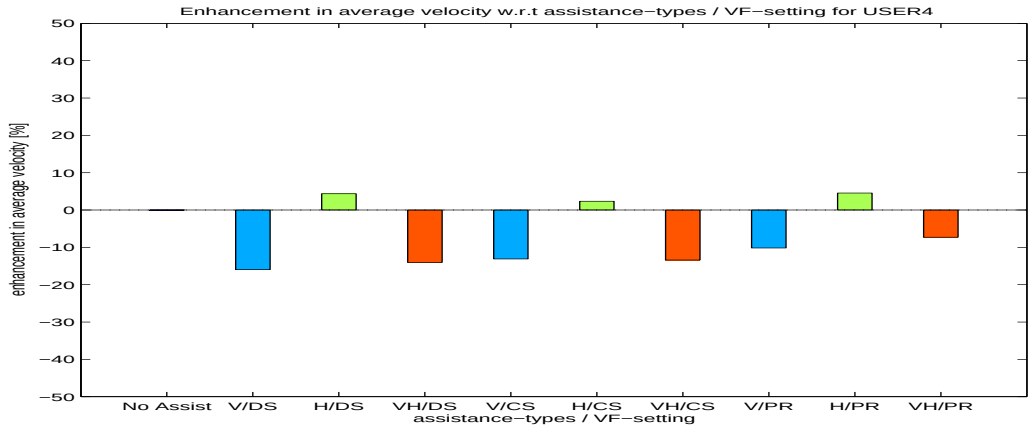


(b) USER2

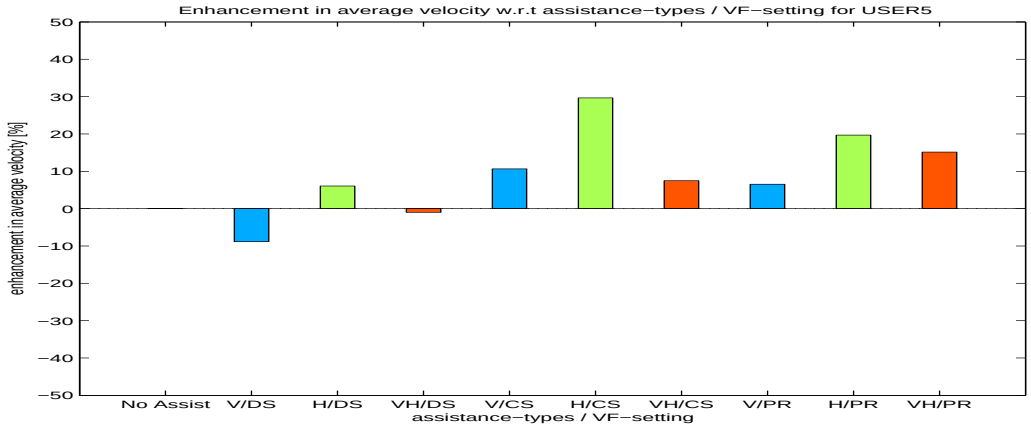


(c) USER3

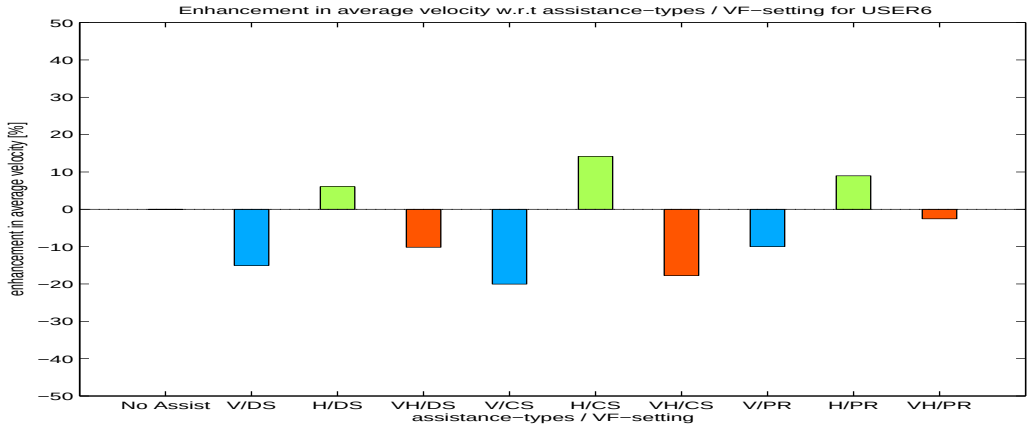
Figure B.5: Performance enhancement in average velocity as percent with respect to feedback types for USER1 through USER23.



(d) USER4*

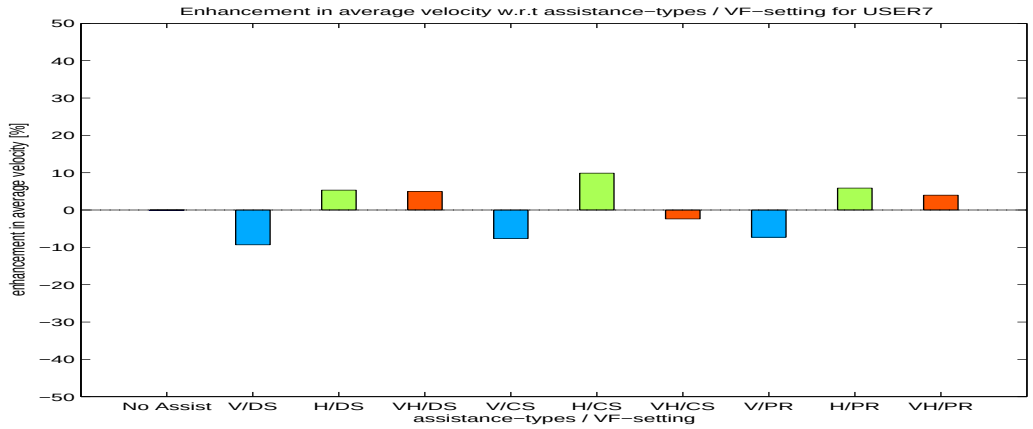


(e) USER5

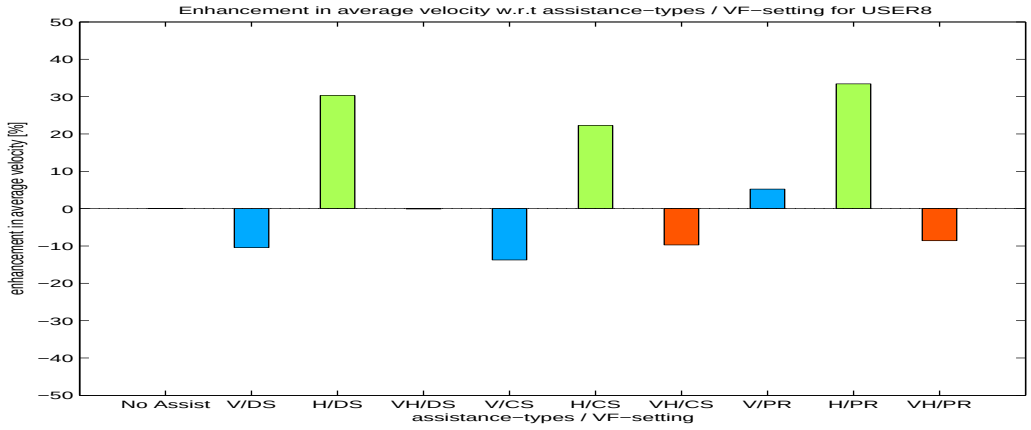


(f) USER6

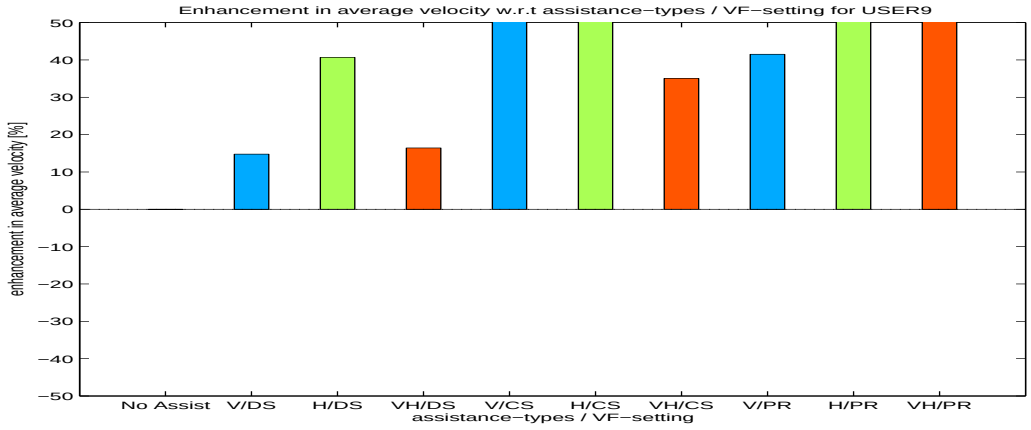
Figure B.5: Continued.



(g) USER7

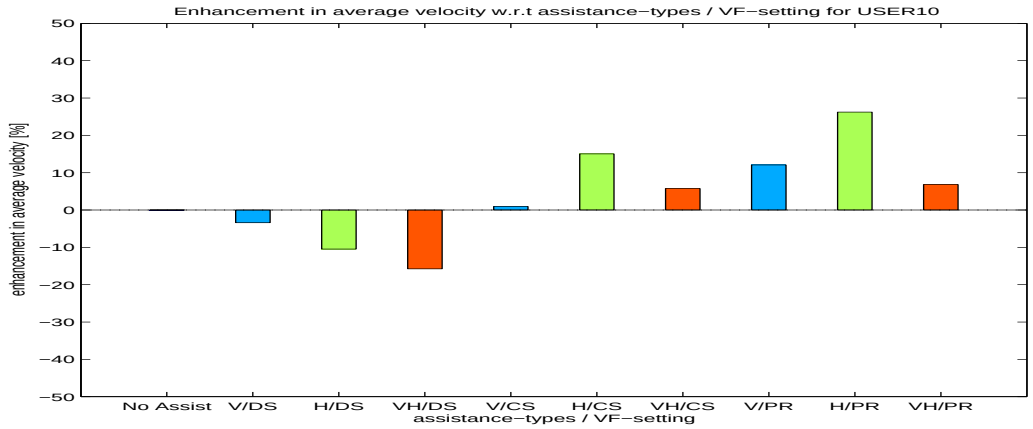


(h) USER8

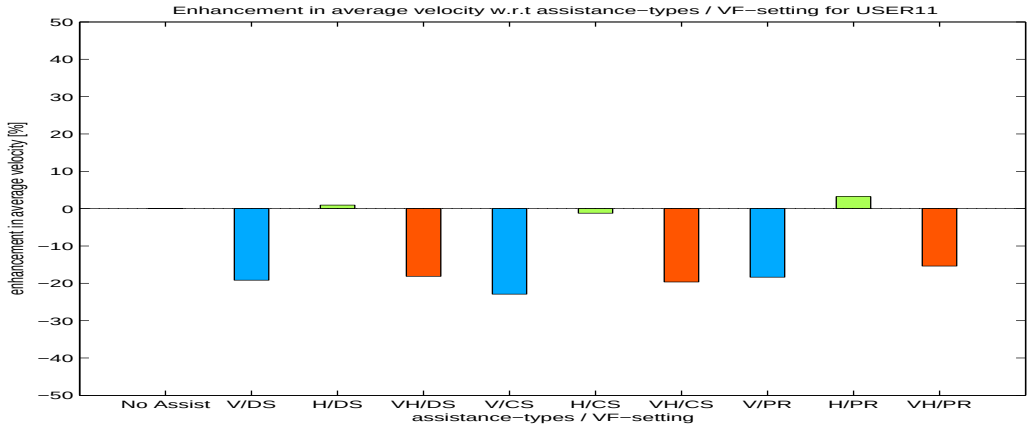


(i) USER9‡

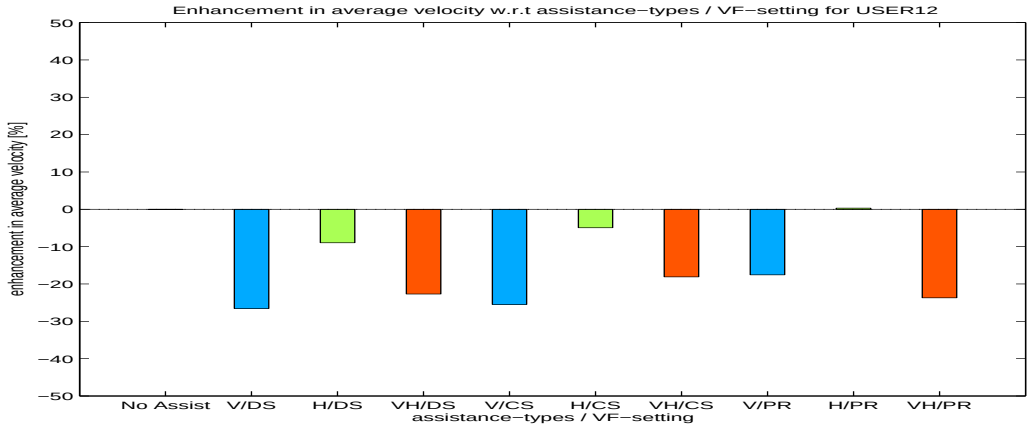
Figure B.5: Continued.



(j) USER10

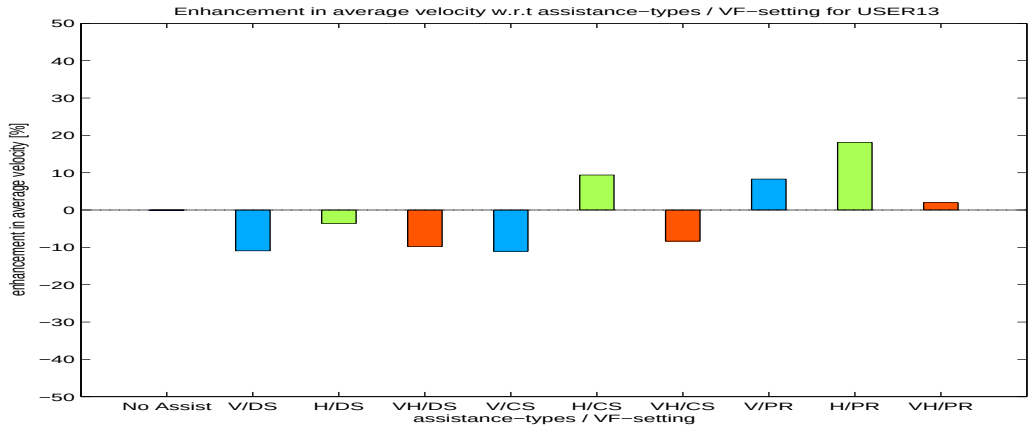


(k) USER11

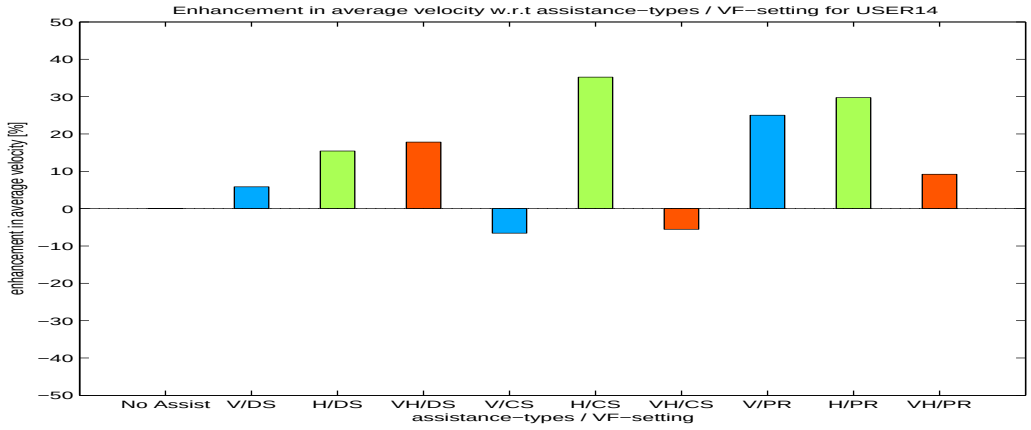


(l) USER12

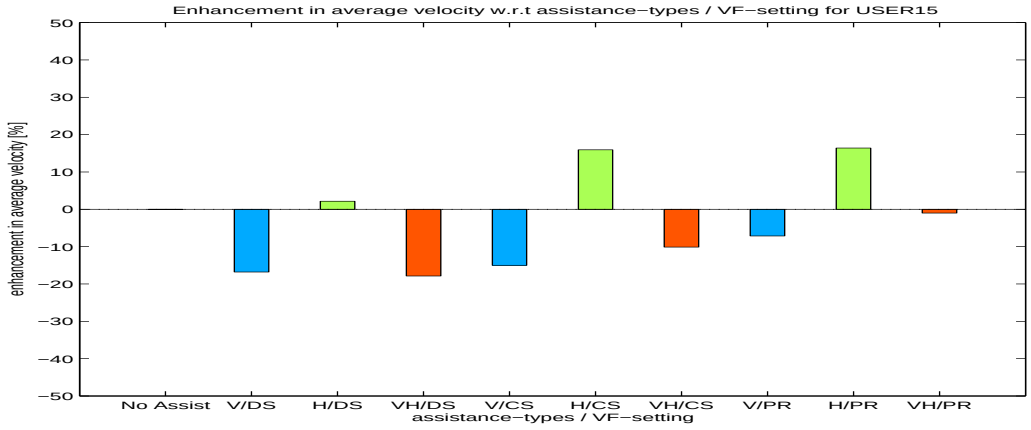
Figure B.5: Continued.



(m) USER13

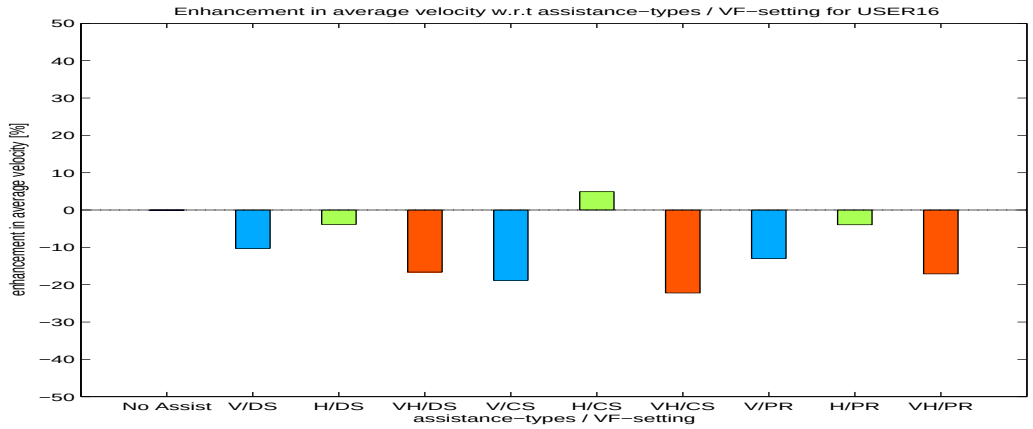


(n) USER14

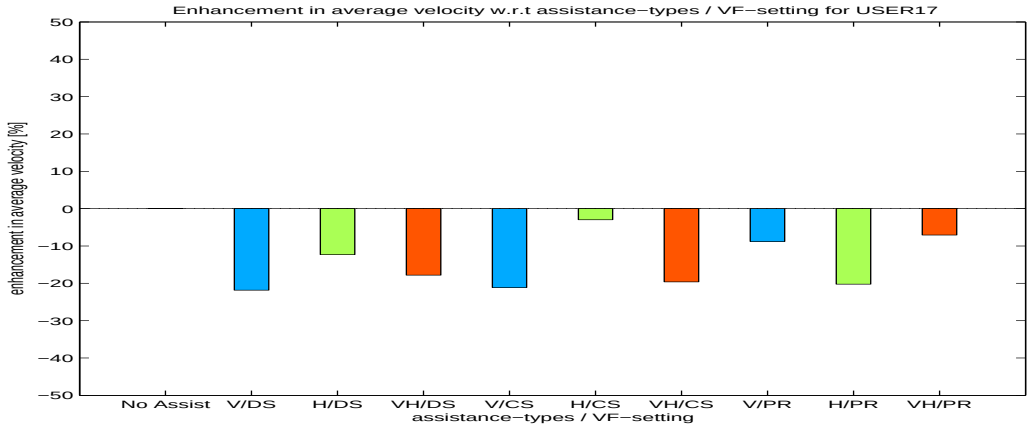


(o) USER15

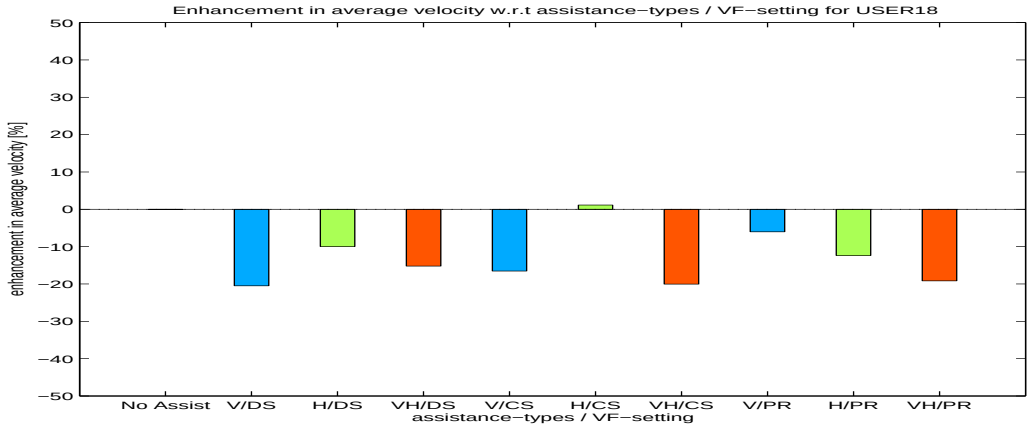
Figure B.5: Continued.



(p) USER16

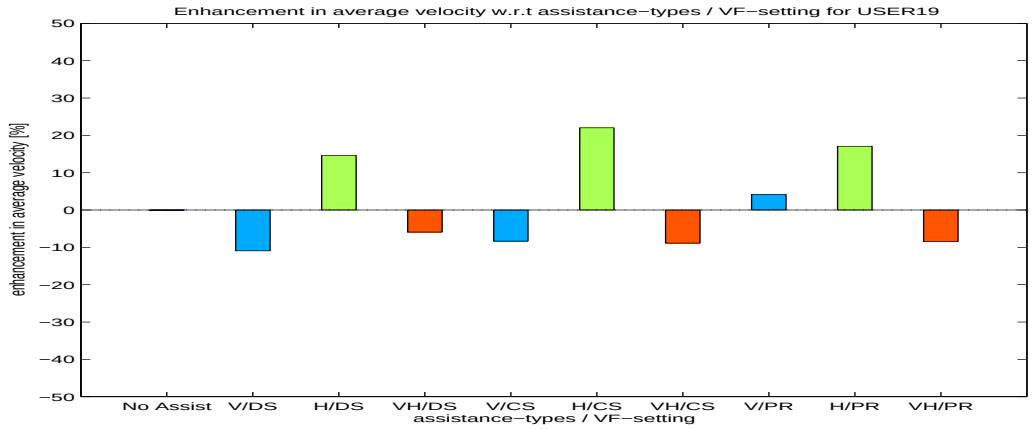


(q) USER17

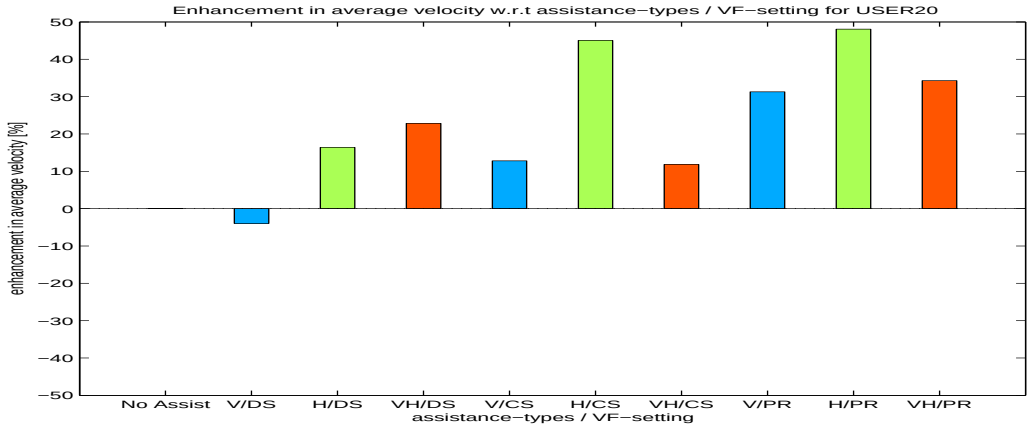


(r) USER18

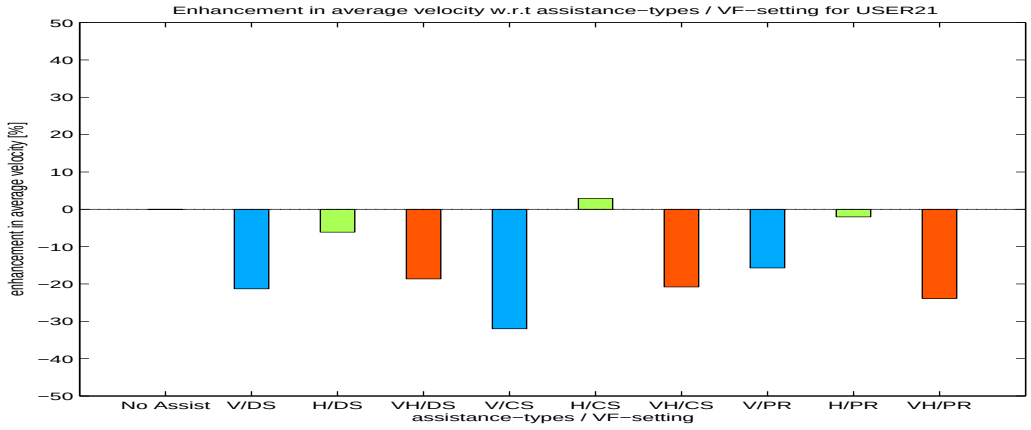
Figure B.5: Continued.



(s) USER19

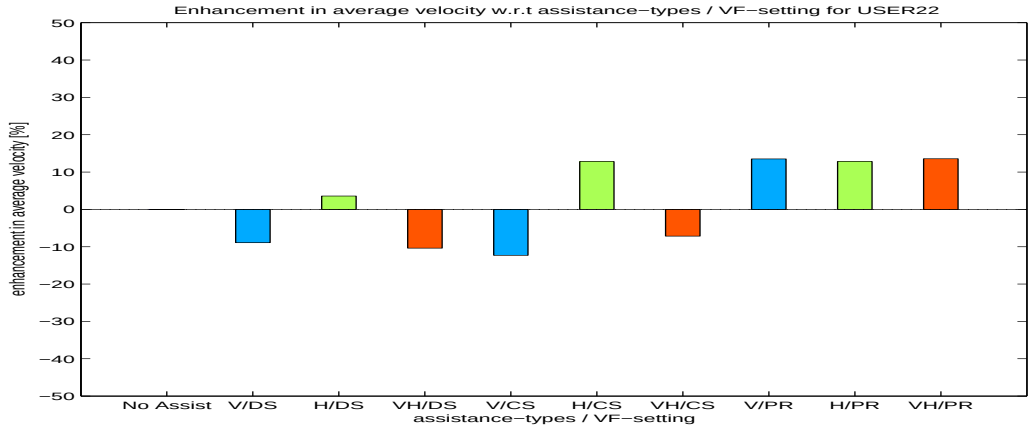


(t) USER20

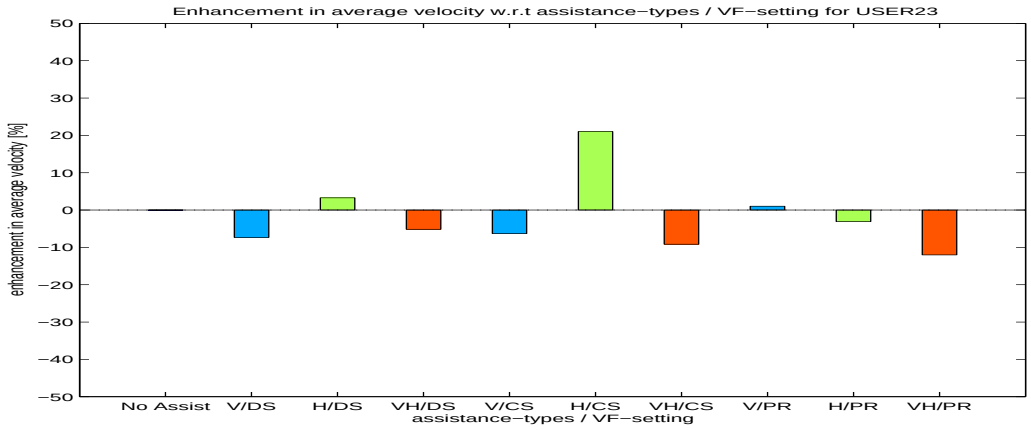


(u) USER21

Figure B.5: Continued.



(v) USER22



(w) USER23

Figure B.5: Continued.

REFERENCES

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

[2] A. Dragan and S. Srinivasa, "Formalizing assistive teleoperation," *Robotics: Science and Systems*, 2012.

[3] L. M. Crespo and D. J. Reinkensmeyer, "Haptic guidance can enhance motor learning of a steering task," *Journal of Motor Behavior*, vol. 40, no. 6, pp. 545–557, 2008.

[4] L. Marchal-Crespo, S. McHughen, S. C. Cramer, and D. J. Reinkensmeyer, "The effect of haptic guidance, aging, and initial skill level on motor learning of a steering task," *Experimental Brain Research*, vol. 201, no. 2, pp. 209–220, 2010.

[5] J. J. Abbott, P. Marayong, and A. M. Okamura, "Haptic virtual fixtures for robot-assisted manipulation," in *Robotics Research*. Springer, 2007, pp. 49–64.

[6] I. Emeagwali, P. Marayong, J. J. Abbott, and A. M. Okamura, "Performance analysis of steady-hand teleoperation versus cooperative manipulation," in *Proceedings of 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2004, pp. 316–322.

[7] R. Reilink, S. Stramigioli, and S. Misra, "Image-based flexible endoscope steering," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 2339–2344.

[8] R. Reilink, S. Stramigioli, A. M. Kappers, and S. Misra, "Evaluation of flexible endoscope steering using haptic guidance," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 7, no. 2, pp. 178–186, 2011.

[9] I. Nisky, A. Pressman, C. M. Pugh, F. A. Mussa-Ivaldi, and A. Karniel, "Perception and action in simulated telesurgery," in *Haptics: Generating and Perceiving Tangible Sensations*. Springer, 2010, pp. 213–218.

[10] N. Padoy and G. Hager, "Human-machine collaborative surgery using learned models," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2011, pp. 5285–5292.

[11] C. Passenberg, R. Groten, A. Peer, and M. Buss, "Towards real-time haptic assistance adaptation optimizing task performance and human effort," in *World Haptics Conference (WHC), 2011 IEEE*, June 2011, pp. 155–160.

[12] H. Boessenkool, D. Abbink, C. Heemskerk, F. van der Helm, and J. Wildenbeest, "A task-specific analysis of the benefit of haptic shared control during tele-manipulation," *IEEE Transactions on Haptics*, vol. 6, no. 1, 2013.

[13] J. J. Abbott and A. M. Okamura, "Pseudo-admittance bilateral telemanipulation with guidance virtual fixtures," *The International Journal of Robotics Research*, vol. 26, no. 8, pp. 865–884, 2007.

[14] A. Bettini, P. Marayong, S. Lang, A. M. Okamura, and G. D. Hager, "Vision-assisted control for manipulation using virtual fixtures," *IEEE Transactions on Robotics*, vol. 20, no. 6, pp. 953–966, 2004.

[15] L. B. Rosenberg, "The use of virtual fixtures as perceptual overlays to enhance operator performance in remote environments," USAF Armstrong Laboratory, Tech. Rep., AL/CF-TR-1994-0089, 1992.

[16] B. L. Davies, "19 a discussion of safety issues for medical robots," *Computer-Integrated Surgery: Technology and Clinical Applications*, p. 287, 1996.

[17] J. J. Abbott and A. M. Okamura, "Virtual fixture architectures for telemanipulation," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2. IEEE, 2003, pp. 2798–2805.

[18] J. J. Abbott and A. M. Okamura, "Stable forbidden-region virtual fixtures for bilateral telemanipulation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 128, no. 1, pp. 53–64, 2006.

[19] D. Aarno, S. Ekvall, and D. Kragic, "Adaptive virtual fixtures for machine-assisted teleoperation tasks," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2005, pp. 1139–1144.

[20] M. Li and R. H. Taylor, "Spatial motion constraints in medical robot using virtual fixtures generated by anatomy," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, 2004, pp. 1270–1275.

[21] M. Li, A. Kapoor, and R. H. Taylor, "A constrained optimization approach to virtual fixtures," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 1408–1413.

[22] T. Xia, A. Kapoor, P. Kazanzides, and R. Taylor, "A constrained optimization approach to virtual fixtures for multi-robot collaborative teleoperation," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 639–644.

[23] F. Ryden and H. J. Chizeck, "Forbidden-region virtual fixtures from streaming point clouds: Remotely touching and protecting a beating heart," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 3308–3313.

[24] A. B. Kuang, S. Payandeh, B. Zheng, F. Henigman, and C. L. MacKenzie, "Assembling virtual fixtures for guidance in training environments," in *Proceedings of 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2004, pp. 367–374.

[25] F. Wang and S. Payandeh, "A study of hybrid virtual fixtures in assistive path following problems," in *Proceedings of IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL)*, 2012, pp. 1–2.

[26] T. Sheridan, *Telerobotics, Automation, and Human Supervisory Control*, 2nd ed. Cambridge, MA: MIT Press, 1992.

[27] T. Carlson and Y. Demiris, "Collaborative control in human wheelchair interaction reduces the need for dexterity in precise manoeuvres," in *Proceedings of Workshop at ACM/IEEE HRI, Robotic Helpers: User Interaction, Interfaces and Companions in Assistive and Therapy Robotics*, 2008, pp. 59–66.

[28] C. Burns, J. Zearing, R. F. Wang, and D. Stipanović, "Autonomous and semiautonomous control simulator," in *Proceedings of the 2010 AAAI Spring Symposium on Embedded Reasoning*, 2010.

[29] D. A. Abbink, M. Mulder, and E. R. Boer, "Haptic shared control: smoothly shifting control authority?" *Cognition, Technology & Work*, vol. 14, no. 1, pp. 19–28, 2012.

[30] R. Simpson, D. Poirot, and F. Baxter, "The hephaestus smart wheelchair system," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 10, no. 2, pp. 118–122, 2002.

[31] J. O. Wobbrock, S. K. Kane, K. Z. Gajos, S. Harada, and J. Froehlich, "Ability-based design: Concept, principles and examples," *ACM Transactions on Accessible Computing (TACCESS)*, vol. 3, no. 3, p. 9, 2011.

[32] J. O. Wobbrock, J. Fogarty, S.-Y. S. Liu, S. Kimuro, and S. Harada, "The angle mouse: target-agnostic dynamic gain adjustment based on angular deviation," in *Proceedings of the 27th International Conference on Human Factors in Computing Systems*, 2009, pp. 1401–1410.

[33] L.-M. Muñoz, A. Casals, M. Frigola, and J. Amat, "Motor-model-based dynamic scaling in human-computer interfaces," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 2, pp. 435–447, april 2011.

[34] E. J. Rossetter, J. P. Switkes, and J. C. Gerdes, "A gentle nudge towards safety: Experimental validation of the potential field driver assistance system," in *American Control Conference, 2003. Proceedings of the 2003*, vol. 5, 2003, pp. 3744–3749.

[35] E. J. Rossetter, "A potential field framework for active vehicle lanekeeping assistance," Ph.D. dissertation, Stanford University, 2003.

[36] C. E. Reiley, E. Plaku, and G. D. Hager, "Motion generation of robotic surgical tasks: Learning from expert demonstrations," in *Proceedings of IEEE International Conference on Engineering in Medicine and Biology Society (EMBC)*, 2010, pp. 967–970.

[37] N. Padoy and G. D. Hager, "Human-machine collaborative surgery using learned models," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2011, pp. 5285–5292.

[38] D. Feygin, M. Keehner, and R. Tendick, "Haptic guidance: Experimental evaluation of a haptic training method for a perceptual motor skill," in *Proceedings of IEEE 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2002, pp. 40–47.

[39] J. Lee and S. Choi, "Effects of haptic guidance and disturbance on motor learning: Potential advantage of haptic disturbance," in *Proceedings of IEEE Haptics Symposium*, 2010, pp. 335–342.

[40] B. A. Forsyth and K. E. MacLean, "Predictive haptic guidance: Intelligent user assistance for the control of dynamic tasks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 1, pp. 103–113, 2006.

[41] M. Clamann and D. B. Kaber, "The effects of haptic and visual aiding on psychomotor task strategy development during virtual reality-based training," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 56, no. 1. SAGE Publications, 2012, pp. 2570–2574.

[42] J. Lüttgen and H. Heuer, "The influence of haptic guidance on the production of spatio-temporal patterns," *Human movement science*, vol. 31, no. 3, pp. 519–528, 2012.

[43] R. Blanch, Y. Guiard, and M. Beaudouin-Lafon, "Semantic pointing: improving target acquisition with control-display ratio adaptation," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, ser. CHI '04, 2004.

[44] R. Ellis, A. Cao, A. Pandya, A. Composto, M. Chacko, M. Klein, and G. Auner, "Optimizing the surgeon-robot interface: the effect of control-display gain and zoom level on movement time," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 48, no. 15, 2004, pp. 1713–1717.

[45] G. Casiez, D. Vogel, R. Balakrishnan, and A. Cockburn, "The impact of control-display gain on user performance in pointing tasks," *Human–Computer Interaction*, vol. 23, no. 3, pp. 215–250, 2008.

[46] M. D. Plumlee and C. Ware, "Zooming versus multiple window interfaces: Cognitive costs of visual comparisons," *ACM Trans. Comput.-Hum. Interact.*, vol. 13, June 2006.

[47] Y. Guiard, M. Beaudouin-Lafon, J. Bastin, D. Pasveer, and S. Zhai, "View size and pointing difficulty in multi-scale navigation," in *Proceedings of the working conference on Advanced visual interfaces*, 2004, pp. 117–124.

[48] I. Haskell and C. Wickens, "Two-and three-dimensional displays for aviation: A theoretical and empirical comparison," *The International Journal of Aviation Psychology*, vol. 3, no. 2, pp. 87–109, 1993.

[49] P. Galambos, A. Roka, G. Soros, and P. Korondi, "Visual feedback techniques for telemanipulation and system status sensualization," in *Proceedings of IEEE 8th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, Jan. 2010, pp. 145–151.

[50] B. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*. Prentice-Hall, Inc., 1990.

[51] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *Circuits and Systems Magazine, IEEE*, vol. 9, no. 3, pp. 32–50, 2009.

[52] M. W. Klein, *Mathematical Methods for Economics*. Addison-Wesley Massachusetts, 1998.

[53] R. E. Kalman, "When is a linear control system optimal?" *Journal of Basic Engineering*, vol. 86, p. 51, 1964.

[54] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the 21st International Conference on Machine Learning*. ACM Press, 2004.

[55] K. Dvijotham and E. Todorov, "Inverse optimal control with linearly-solvable mdps," in *Proceedings of 27th International Conference on Machine Learning*, 2010, pp. 335–342.

[56] S. Levine and V. Koltun, "Continuous inverse optimal control with locally optimal examples," in *ICML '12: Proceedings of the 29th International Conference on Machine Learning*, 2012.

[57] N. D. Ratliff, J. A. Bagnell, and M. A. Zinkevich, "Maximum margin planning," in *Proceedings of the 23rd International Conference on Machine Learning*, 2006, pp. 729–736.

[58] A. Keshavarz, Y. Wang, and S. Boyd, "Imputing a convex objective function," in *Proceedings of IEEE International Symposium on Intelligent Control (ISIC)*, 2011.

[59] A.-S. Puydupin-Jamin, M. Johnson, and T. Bretl, "A convex approach to inverse optimal control and its application to modeling human locomotion," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 531–536.

[60] M. Johnson, "Inverse optimal control for deterministic nonlinear systems," Ph.D. dissertation, University of Illinois at Urbana-Champaign, 2013.

[61] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. I, 3rd Ed.* Belmont, MA: Athena Scientific, 2005.

[62] D. Liberzon, *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton University Press, 2012.

[63] H. J. Sussmann and J. C. Willems, "300 years of optimal control: from the brachystochrone to the maximum principle," *Control Systems, IEEE*, vol. 17, no. 3, pp. 32–44, 1997.

[64] B. Hannaford, "A design framework for teleoperators with kinesthetic feedback," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 4, pp. 426–434, 1989.

[65] B. Hannaford, "Stability and performance tradeoffs in bilateral telemanipulation," in *Proceedings of IEEE International Conference on Robotics and Automation*, 1989.

[66] P. F. Hokayem and M. W. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, no. 12, pp. 2035 – 2057, 2006.

[67] R. J. Anderson and M. W. Spong, "Bilateral control of teleoperators with time delay," *IEEE Transactions on Automatic Control*, vol. 34, no. 5, pp. 494–501, 1989.

[68] R. J. Anderson and M. W. Spong, "Asymptotic stability for force reflecting teleoperators with time delay," *International Journal of Robotics Research*, vol. 11, no. 2, pp. 135–149, 1992.

[69] N. Chopra, M. W. Spong, S. Hirche, and M. Buss, "Bilateral teleoperation over the internet: The time varying delay problem," in *Proceedings of the American Control Conference*, vol. 1, 2003, pp. 155–160.

[70] P. Berestesky, N. Chopra, and M. W. Spong, "Discrete time passivity in bilateral teleoperation over the internet," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2004.

[71] J.-H. Ryu, B. Hannaford, C. Preusche, and G. Hirzinger, "Time domain passivity control with reference energy behavior," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, Oct. 2003, pp. 2932–2937.

[72] C. Preusche, G. Hirzinger, J.-H. Ryu, and B. Hannaford, "Time domain passivity control for 6 degrees of freedom haptic displays," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, oct. 2003, pp. 2944–2949.

[73] J.-H. Ryu, D.-S. Kwon, and B. Hannaford, "Stable teleoperation with time-domain passivity control," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 365–373, apr. 2004.

[74] J.-H. Ryu, D.-S. Kwon, and B. Hannaford, "Stability guaranteed control: time domain passivity approach," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 6, pp. 860–868, nov. 2004.

[75] T. B. Sheridan, "Space teleoperation through time delay: review and prognosis," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 5, pp. 592–606, 1993.

[76] D. Lee, O. Martinez-Palafox, and M. W. Spong, "Bilateral teleoperation of a wheeled mobile robot over delayed communication network," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2006.

[77] O. Martinez-Palafox, D. Lee, M. W. Spong, I. Lopez, and C. T. Abdallah, "Bilateral teleoperation of mobile robot over delayed communication network: Implementation," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, 2006, pp. 4193–4198.

[78] O. Martinez-Palafox and M. W. Spong, "Bilateral teleoperation of a formation of nonholonomic mobile robots under constant time delay," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, 2009, pp. 2821–2826.

[79] I. Farkhatdinov, J.-H. Ryu, and J. An, "A preliminary experimental study on haptic teleoperation of mobile robot with variable force feedback gain," in *Proceedings of IEEE Haptics Symposium*, 2010, pp. 251–256.

[80] I. Farkhatdinov and J.-H. Ryu, "Stability analysis of mobile robot teleoperation with variable force feedback gain," in *Proceedings of the 2010 International Conference on Haptics: Generating and Perceiving Tangible Sensations, Part I*, 2010, pp. 177–182.

[81] I. Nisky, F. A. Mussa-Ivaldi, and A. Karniel, "Analytical study of perceptual and motor transparency in bilateral teleoperation," *IEEE Transactions on Human-Machine Systems*, vol. 43, no. 6, pp. 570–582, nov 2013.

[82] D. A. Lawrence, "Stability and transparency in bilateral teleoperation," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 5, pp. 624–637, 1993.

[83] D. G. Luenberger, *Optimization by Vector Space Methods*, 1st ed. New York, NY, USA: John Wiley & Sons, Inc., 1997.

[84] J. J. Abbott and A. M. Okamura, "Generating a state-space formulation of a class of telemanipulator," Haptic Exploration Lab Technical Report, pp. 03–1, 2003.

[85] H. K. Khalil, *Nonlinear Systems*, 3rd ed. New Jersey: Prentice Hall, 2002.

[86] N. Diolaiti and C. Melchiorri, "Haptic tele-operation of a mobile robot," in *Proceedings of the 7th IFAC Symposium of Robot Control*, 2003, pp. 2798–2805.

[87] K. Hashtrudi-Zaad and S. E. Salcudean, "Analysis of control architectures for teleoperation systems with impedance/admittance master and slave manipulators," *The International Journal of Robotics Research*, vol. 20, no. 6, pp. 419–445, 2001.

[88] S. Martin and N. Hillier, "Characterisation of the novint falcon haptic device for application as a robot manipulator," in *Australasian Conference on Robotics and Automation (ACRA)*. Citeseer, 2009, pp. 291–292.

[89] P. P. Jonker, J. Caarls, W. J. Bokhove, W. Altewischer, and I. T. Young, "Robosoccer: autonomous robots in a complex environment," in *Second International Conference on Image and Graphics*. International Society for Optics and Photonics, 2002, pp. 47–54.

[90] A. Chopra, M. Obsniuk, and M. R. Jenkin, "The nomad 200 and the nomad superscout: Reverse engineered and resurrected," in *Proceedings of IEEE The 3rd Canadian Conference on Computer and Robot Vision*. IEEE, 2006, pp. 55–55.

[91] S. Mastellone, D. Stipanovic, and M. Spong, "Remote formation control and collision avoidance for multi-agent nonholonomic systems," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2007, pp. 1062–1067.

[92] J. Accot and S. Zhai, "Scale effects in steering law tasks," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2001, pp. 1–8.

[93] J. J. Abbott and A. M. Okamura, "Analysis of virtual fixture contact stability for telemanipulation," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3. IEEE, 2003, pp. 2699–2706.

[94] E. Malakhovski and L. Mirkin, "On stability of second-order quasi-polynomials with a single delay," *Automatica*, vol. 42, no. 6, pp. 1041–1047, 2006.

[95] A. J. Van Der Schaft, *L2-Gain and Passivity Techniques in Nonlinear Control*. London: Springer, 2000.

[96] A. Coates, P. Abbeel, and A. Y. Ng, "Learning for control from multiple demonstrations," in *Proceedings of the 25th International Conference on Machine Learning*, 2008, pp. 144–151.

[97] G. Arechavaleta, J.-P. Laumond, H. Hicheur, and A. Berthoz, "An optimality principle governing human walking," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 5–14, Feb. 2008.

[98] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotion-an inverse optimal control approach," *Autonomous Robots*, vol. 28, no. 3, pp. 369–383, 2010.

[99] W. Li, E. Todorov, and D. Liu, "Inverse optimality design for biological movement systems," in *Proceedings of World Congress*, vol. 18, no. 1, 2011, pp. 9662–9667.

[100] B. Ziebart, A. Dey, and J. A. Bagnell, "Probabilistic pointing target prediction via inverse optimal control," in *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, 2012, pp. 1–10.

[101] S. Levine, Z. Popovic, and V. Koltun, "Feature construction for inverse reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 23, 2010.

[102] P. Abbeel and A. Y. Ng, "Exploration and apprenticeship learning in reinforcement learning," in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 1–8.

[103] I. M. Gelfand and S. V. Fomin, *Calculus of variations*. Dover publications, 2000.

[104] P. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement," *Journal of Experimental Psychology*, vol. 47, no. 6, pp. 381–391, 1954.

[105] S. Zhai, J. Accot, and R. Woltjer, "Human action laws in electronic virtual worlds: an empirical study of path steering performance in VR," *Presence: Teleoperators and Virtual Environments*, vol. 13, no. 2, pp. 113–127, 2004.

[106] N. Rashevsky, "Mathematical biophysics of automobile driving," *The Bulletin of Mathematical Biophysics*, vol. 21, no. 4, pp. 375–385, 1959.

[107] C. Drury, "Movements with lateral constraint," *Ergonomics*, vol. 14, no. 2, pp. 293–305, 1971.

[108] University of Washington, "Reporting results of common statistical tests in APA format." [Online]. Available: http://www.psych.uw.edu/writingcenter/writingguides/pdf/stats.pdf