RICE UNIVERSITY

Implementation and Analysis of Shared-Control Guidance Paradigms for Improved Robot-Mediated Training

by

Dane Powell

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

Master of Science

APPROVED, THESIS COMMITTEE:

Marcia K. O'Malley, Chair
Associate Professor of Mechanical
Engineering and Materials Science

Michael Byrne
Associate Professor of Psychology

Andrew Meade
Professor of Mechanical Engineering and
Materials Science

HOUSTON, TEXAS

NOVEMBER, 2010

Abstract

Implementation and Analysis of Shared-Control Guidance Paradigms for Improved Robot-Mediated Training

by

Dane Powell

Many dynamic tasks have a clearly defined optimal trajectory or strategy for completion. Human operators may discover this strategy naturally through practice, but actively teaching it to them can increase their rate of performance improvement. Haptic devices, which provide force feedback to an operator, can physically guide participants through the optimal completion of a task, but this alone does not ensure that they will learn the optimal control strategy. In fact, participants may become dependent on this guidance to complete the task. This research focuses on developing and testing ways in which guidance can be modulated such that it conveys the proper task completion strategy without physically dominating the operator and thus encouraging dependency. These guidance schemes may also be applied to the real-time execution of tasks in order to convey computer-generated task completion strategies to a user without allowing the computer to physically dominate control of the task.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The purpose of this work is to implement and experimentally analyze a number of shared-control haptic guidance paradigms intended to improve robot-mediated training for dynamic tasks.

There are many examples of dynamic tasks in our everyday lives. Shooting a basketball, driving a car, or simply taking a sip of water are all characteristically dynamic tasks that require sensory feedback (especially haptic feedback), on-line movement planning, and adaptation to changing task conditions. Most importantly, these are all tasks that have one or more optimal solutions that either maximize a "positive" metric, such as likelihood of making a basket, or minimize a "negative" metric, such as the amount of effort required. These optimal solutions are learned through a combination of practice and training, either by direct intervention from a coach or through focused observation of other people performing the task. Similarly, there are many less common but more consequential dynamic tasks requiring extensive training, such as performing a laparoscopic surgery, flying an airplane, or teleoperating a remotely-operated vehicle.

Training for these tasks can be either human-mediated (Figure 1.1) or robot-mediated (Figure 1.2). An expert surgeon gripping a novice's hand in order to physically help that novice complete a surgery would be an example of human-mediated training. Conversely, a novice training to complete the surgery in a virtual environment with the assistance of either a live or virtual expert surgeon would be an example of robot-mediated training. Robot-mediated training has many potential advantages over human-mediated training, as will be discussed in Section 3.3. Moreover, if the expert and novice share control over the virtual scalpel according to some algorithm (perhaps based on the amount of control allotted to the novice by the expert), this is an example of a shared-control system. Shared-control systems offer potential advantages over other types of robot-mediated training methodologies because they allow the inclusion of either a real or virtual expert in the training program, as will be discussed in Section 3.4.

Figure 1.1: Gillespie et al. [1]'s Virtual Teacher paradigms. From left to right: indirect-contact, double-contact, and single-contact paradigms.

While the question of how to apportion control of the system between expert and novice has been studied to some extent in the literature, the question of how to provide feedback, especially to the novice, has been studied comparatively little. Of particular interest and challenge is the question of how to provide haptic feedback to the novice. While haptic feedback can greatly enhance a novice's sense of presence and cooperation [3, 4] and potentially enhance training if used properly, it can also be distracting and introduce problems of its own. Such feedback, if coming directly from an expert, is generally referred to as haptic "guidance," as it is generally used to guide a novice through the successful completion of a task and thus enhance training

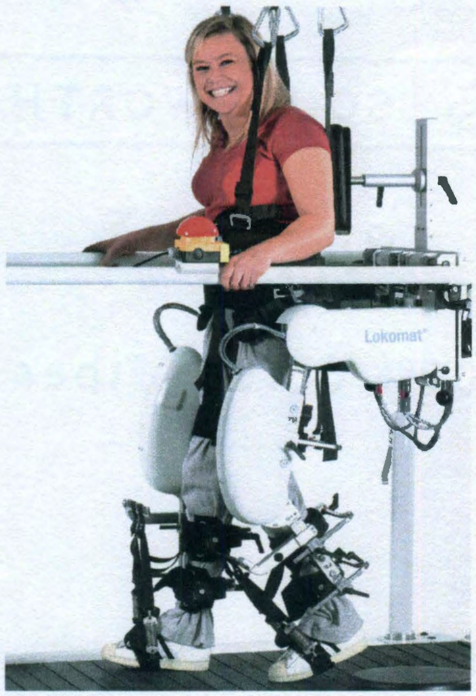

Figure 1.2: Patient undergoing robot-mediated rehabilitation using the Lokomat system (Jezernik et al. [2])

outcomes.

Chapter 2

Motivation and Novel Contributions

2.1 Guidance Paradigm Taxonomy

Most guidance schemes used for robot-mediated training have been developed in an ad-hoc fashion, that is, to work with a specific device or fill a specific need. This makes it difficult to compare the multitude of guidance schemes present in the literature because the type of guidance provided is confounded with the hardware that it is implemented on. For example, Gillespie et al. [1] developed the haptic device shown in Figure 2.1 specifically to test the "double-contact virtual teacher" guidance paradigm illustrated in Figure 1.1. This device is highly task-specific, requiring the novice to grip the handle (analogous to the racket) and providing guidance via the band/cable coupling (analogous to the coach). It would be difficult or impossible to test that paradigm on any commercially available haptic device because of the specific way in which it was described and the custom hardware on which it was implemented. Thus, it would be equally difficult to compare the effectiveness of that paradigm to other paradigms in the literature, especially if they also share a dependence on specific or proprietary hardware.

I propose that the "double-contact virtual teacher" paradigm, along with others like it in the literature, can be distilled into a set of essential and representative characteristics, and that these characteristics can be used to develop a taxonomy for classifying guidance paradigms. By abstracting the principles of existing guidance

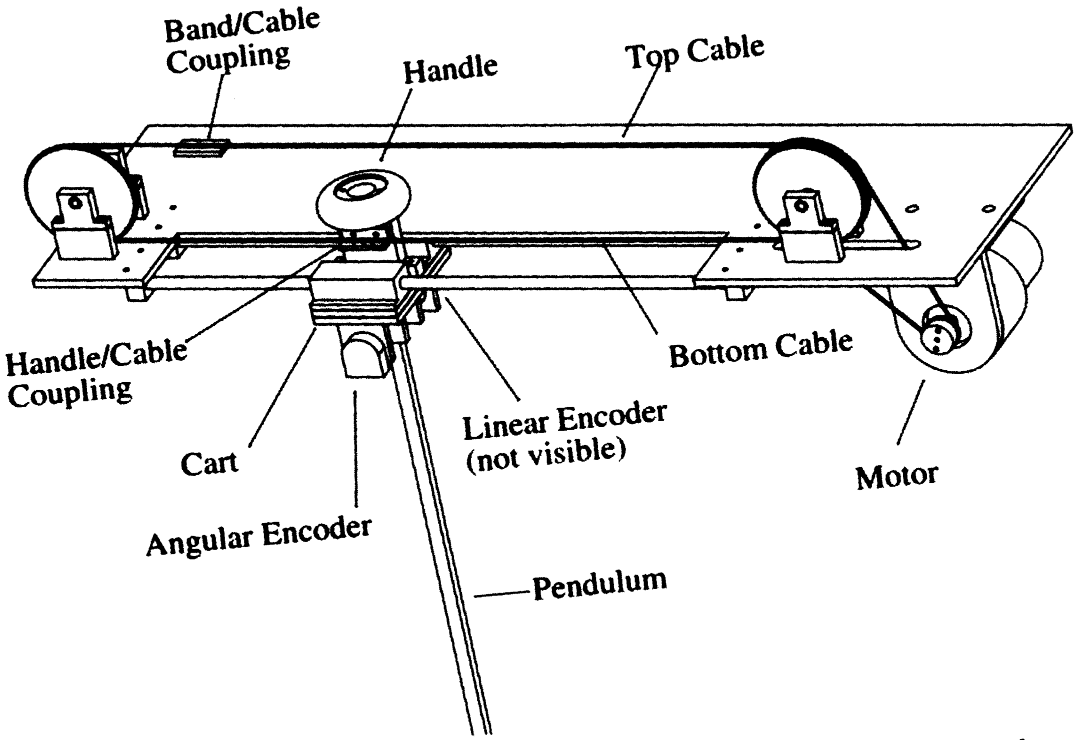

Figure 2.1: Haptic device developed by Gillespie et al. [1] to test double-contact paradigm. This device simulates an expert teaching a novice to perform a one-dimensional dynamic task. The novice grips the handle and uses it to control the task, and is attached to the top cable via a special glove. This top cable simulates the expert and provides guidance.

paradigms from their specific implementations, we can develop a set of representative paradigms from the taxonomy and then compare the effectiveness of each of those paradigms while holding constant the specifics of the implementation (such as the choice of haptic device and dynamic task).

This taxonomy is discussed in detail in Chapter 4.

2.2 Dynamic Task Platform

While there are a number of haptic simulation development toolkits available today (such as CHAI and H3D), none are geared specifically to the task of robustly collecting large amounts of data from human subject testing. Virtual environments (similar to the guidance paradigms they are intended to test) have generally been developed in an ad-hoc fashion and tailored for a specific implementation (i.e. a certain combination of hardware, guidance, etc...). This leads to an unnecessary repetition of labor and a loss of potential accumulated experience from one experimenter to the next. If researchers had an existing tried-and-tested development platform on which to base new experimenters, this could improve their efficiency and quality of results at the same time.

To that end, I have developed the Dynamic Task Platform, an object-oriented, modular, and extensible experimental platform writen in C++ with the goal of facilitating the further study of human performance in dynamic tasks. This platform is discussed in detail in Chapter 5.

2.3 Shared-Control Proxy Model

In many virtual environments, haptic feedback is rendered using a simple proxy model, where a massless proxy in the virtual environment is connected to the representation of the haptic device (user) by a spring and damper, as shown in Figure 2.2. The proxy must obey all of the physical constraints of the virtual environment (i.e. walls, friction, etc...), while the user is bound by no constraint other than the virtual spring and damper link to the proxy. Thus, the forces on the user calculated by that spring and damper link can simply be amplified to generate a force output for the haptic

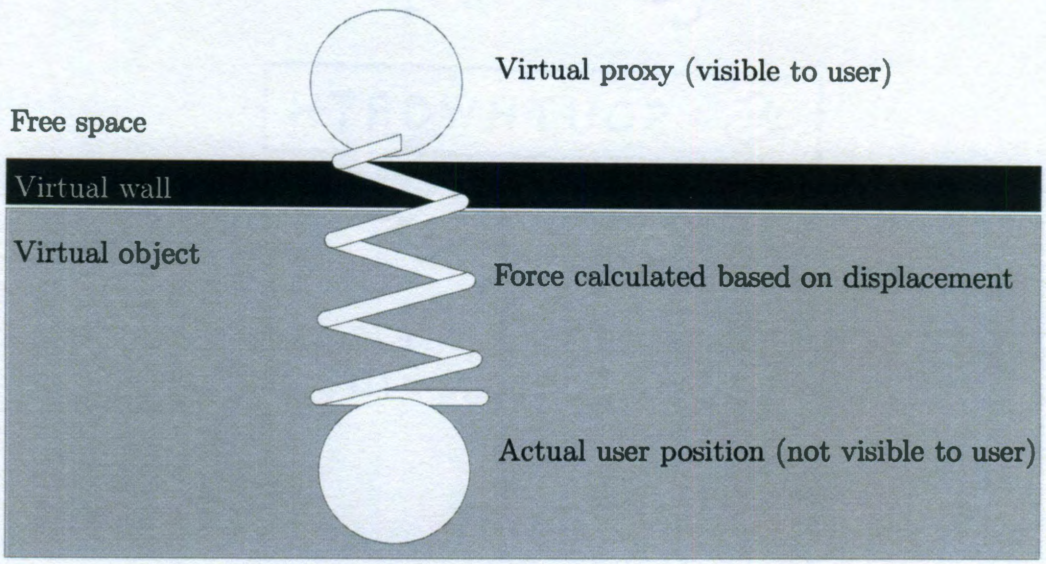

Figure 2.2: Proxy colliding with a virtual wall. This illustrates how forces are calculated when a user controlling a haptic device collides with a wall in a virtual environment. In free space, the user and proxy are coincident. When the user collides with the virtual wall, the user penetrates it while the proxy remains outside. A force is applied to the user based on the displacement between the proxy and the user.

device.

If a perceptual overlay or virtual expert is added to the environment, one can imagine that there are two qualitatively different types of forces in the system: "guidance" forces, which arise from interactions with the perceptual overlay or virtual expert, and "task" forces, which arise from interactions with the virtual environment. A distinction should be made between these types of feedback because they should contribute to a user's learning in fundamentally different ways: "guidance" forces should be used to shape the user's actions, whereas "task" forces should be incorporated into the user's internal model of the environment. The problem with the traditional proxy model is that it cannot discriminate between guidance and task forces in shared-control systems, and thus the forces are confounded when displayed to the user. This could lead to impaired training and understanding by the user.

In Chapter 6, I propose a shared-control proxy model to allow for the discrimination of task and guidance forces.

2.4 Evaluation of Four Guidance Paradigms

Four prototypical guidance paradigms (one of each major type from the proposed taxonomy) are developed and implemented on the Dynamic Task Platform using commercially-available hardware. One of these paradigms is more "traditional" in the sense that it accounts for most of the guidance currently provided in robot-mediated training, while three of the paradigms are relatively novel or at least "non-traditional." These paradigms are used to train subjects to perform a number of dynamic tasks in a controlled experiment, and the effectiveness of each of these paradigms is evaluated.

Demonstrating that the "traditional" paradigm is generally superior would be important as it would reinforce the construct validity of its plethora of implementations in the literature. Conversely, demonstrating that the "non-traditional" paradigms are superior would be a boon for robot-mediated training, as it would stand to improve training outcomes throughout the field. The cumulative effect of even a modest increase in effectiveness could be significant, because of the broad applicibility of the paradigms (as discussed in Section 2.1). Finally, these experiments will help to characterize other aspects of the paradigms (such as workload imposed on users), and thus their suitability for use in different types of environments.

These experiments and their results are presented in Chapter 7.

Chapter 3

Background

3.1 Haptic Interfaces

> **Hap•tic** *adj.* relating to or based on the sense of touch. [Greek *haptesthai*, to touch.]

A haptic interface is a special type of human-machine interface that allows the machine to provide controlled feedback to the human via his or her sense of touch. While "haptic interfaces" may not yet be part of the common parlance, they are increasingly common parts of our everyday lives. Mobile phones that vibrate in response to touch, "Rumble Paks" and other video game controllers that vibrate in response to cues in the game's virtual environment, and force feedback joysticks are all examples of haptic interfaces in consumer electronics that have been available for a decade or more. Industrial and commercial examples include the "stick-shaker" mechanism used to alert pilots to a stall condition on most modern aircraft and laparoscopic surgical simulators that provide a surgeon with realistic force-feedback from a virtual surgical environment. Haptic interfaces can also be used for research in order to discover how humans interact with each other [5] and learn new motor skills via the sense of touch. Finally, haptic interfaces can be used to enhance the quality of training and rehabilitation, as will be discussed in Section 3.3.

What we commonly refer to as our sense of touch actually consists of at least two distinct senses: tactile perception (the sense of vibration, temperature, and texture) and kinesthetic or proprioceptive perception (the sense of force and position). Haptic devices are usually designed to provide either primarily tactile feedback (such as the vibration feature of a mobile phone) or primarily kinesthetic feedback (such as the force-feedback capabilities of a joystick in a flight simulator). Because our kinesthetic and proprioceptive senses are most important in learning to perform dynamic tasks and construct internal models of dynamic systems, the remainder of this work will focus primarily on haptic interfaces that provide force feedback.

3.2 Haptic Rendering

Haptic rendering refers to the process of calculating stable and realistic force feedback based on a user's interactions with a virtual environment in order to either increase the user's sense of presence or provide supplementary feedback such as guidance cues via what are known as "perceptual overlays" [6]. This process is made decidedly non-trivial by a number of technological limitations and consequences of natural laws. The details of these difficulties are not relevant to this work, and the reader is referred to the works of Adams and Hannaford [7] and Gillespie and Cutkosky [8] for a more thorough discussion of the challenges associated with haptic rendering. Suffice it to say that rendering a perfectly stiff virtual wall (the "gold standard" for a haptic device) is simply not possible with most available haptic devices. Because of these limitations, direct calculation of forces based on the user's position (the "penalty-based" approach) can lead to often explosive instability and rendering artifacts, such as "clipping" through very thin objects. Thus, a more general way of rendering interaction forces is required.

A commonly used rendering method developed by Ruspini and Khatib [9] allows a user to interact with a virtual environment by means of a "virtual proxy", as shown in Figure 2.2. This massless proxy visually represents the user in the virtual environment and is bound by all of the physical constraints of that environment, but does not necessarily represent the user's actual position. Instead, a virtual spring and damper system connect the virtual proxy to the user, and the forces generated by this system are amplified and displayed (haptically) to the user. The example of a user colliding with a wall in a virtual environment is illustrated in Figure 2.2. When the user is moving through free space, the proxy and user share essentially the same position. As the user collides with and begins to penetrate the wall, the proxy remains outside the wall, causing a displacement in the virtual spring and damper coupling. The force generated by this displacement is then output to the user via the haptic device, completing the rendering process.

This traditional "proxy model" is advantageous for several reasons. First, this model can be used to render a virtual environment full of arbitrary objects without special attention needing to be paid to how each individual object will be rendered. For instance, if a user penetrates a thin object far enough to pass through it completely, the proxy will obey the physical constraints of the environment and not penetrate the object, while the resistive force provided to the user will continue to grow or saturate. This is certainly more realistic than a penalty-based approach, which will allow the user to clip through the object completely if enough force is applied. Secondly, this model can effectively render higher virtual stiffnesses due to the way in which the central nervous system (CNS) integrates multi-modal sensory information. Visual feedback will tend to "dominate" proprioceptive feedback when the CNS updates its internal model [10], and thus users are more likely to rate a surface as being stiffer if they do not see themselves (their proxy) penetrate the surface. Similarly, auditory and tactile cues can be used to enhance the apparent stiffness of a surface without increasing the force-feedback gains. For instance, Kuchenbecker et al. [11] found that simply displaying high-frequency transient vibratory cues in conjunction with collisions with virtual surfaces greatly enhanced the perceived realism of the collisions.

The reader is referred to the work of Salisbury et al. [12] for a more detailed overview of haptic rendering.

3.3 Robot-Mediated Training

The sense of touch is an integral part of the learning process for visuo-motor dynamic tasks (tasks requiring "hand-eye coordination"), and thus it makes sense that haptic interfaces are increasingly being used for training and rehabilitation, as mentioned in Section 3.1. The defining characteristic of robot-mediated training is that guidance is administered physically to a patient or novice via a haptic interface. Thus, a therapist or coach might still retain high-level control over the course of training or even participate teleoperatively, but all physical interactions with the novice are mediated by the haptic interface and related control systems (the "robot").

Robot-mediated training offers many potential advantages over traditional "human-mediated" training. If enough autonomy can be given to the robot or a "virtual expert", one human expert could potentially train a large number of novices simultaneously, increasing the reach of training. This is an example of shared-control guidance, and will be discussed in detail in Section 3.4. If the haptic interface is linked to a virtual environment, this offers the advantages of being able to quickly change or reset the training environment in order to facilitate training and help keep the novice's attention through long training sessions. More importantly, a robot can also offer objective measures of performance much more frequently than a human expert [13]. Winstein [14] and others have shown that providing accurate and timely feedback to a novice can directly improve training outcomes, and such measures can be used by a real or virtual expert to tune or adapt other aspects of the training as the novice improves over time. For instance, Li et al. [15] and Huegel and O'Malley [16] used these measures to progressively decrease the amount of guidance provided to novices as their performances improved.

Guidance during robot-mediated training is usually provided via simple perceptual overlays such as virtual fixtures. Virtual fixtures, as proposed by Rosenberg [17], are simply perceptual overlays that passively prevent participants from entering forbidden regions of a work environment, and are most often used to constrain a novice's motions to an optimal trajectory. Guidance might also take a more active form, such as the "record-and-replay" strategy used by Gillespie et al. [1] to train novices to balance a inverted pendulum.

Such "assistive" methods are based on a number of intuitions about how people learn to perform visuo-motor tasks. Unfortunately, there is little evidence to back up some of these intuitions or to suggest how they can best be applied to enhance the efficacy of assistive strategies. A common assumption is that physically guiding a novice through the successful completion of a task will help the novice to somehow internalize and encode that pattern, and thus help the novice to repeat the pattern on his or her own in the future. While sounding plausible, this assumption is only weakly supported by the literature in the context of rehabilitation [18, 19], and has been refuted in many cases in the context of training healthy individuals [20–22]. Schmidt and Bjork [23] showed that guidance in many sorts of training (not just in visuo-motor tasks) can actually impair learning and retention, and proposed the "guidance hypothesis" to account for this discrepancy between the expected and actual results of guidance-based training.

The probable flaw in the assumption that assistive guidance improves training is that while the proprioceptive sensory pathways are active in the presence of guidance, the motor pathways are comparatively less active. Israel et al. [24] showed that when physically guided through a task, novices tend to become "passive participants" and exert less energy (reflecting less motor pathway activity) than when they perform the task on their own. Shadmehr and Mussa-Ivaldi [25] showed that the CNS relies on encoding and storing control loops between proprioceptive input and motor output in order to perform dynamic tasks, and thus if this control loop is weak or absent in the presence of guidance, the CNS will not be able to encode and retain it as it would during practice.

Another problem with assistive guidance is that because novices are passive and constrained to an optimal trajectory, they are going to make fewer errors than they would during practice. Error has been shown to drive learning of dynamic tasks and building of internal models, and thus assistive guidance is likely to impair learning by preventing the commission of error.

Finally, a significant problem with assistive guidance is that it corrupts the inherent dynamics of a task as perceived by the novice. Most guidance methods are impedance-based, meaning that they apply a force in order to control the novice's position. Thus, a movement made during practice will result in force-feedback based on the inherent task dynamics, while an identical movement during training will result in force-feedback based on some combination of the task dynamics and guidance forces. If novices spend a bulk of their time in training, then in effect they will be learning the wrong task! Crespo and Reinkensmeyer [26] showed that "subjects who trained with guidance reacted as if the assistance provided on assisted trials was a perturbation rather than following its example," lending credence to this hypothesis.

This brings to light a problem with the traditional proxy model, as mentioned in Section 2.3: it cannot discriminate between guidance and task forces, and thus the two types of forces are confounded when displayed to the user. This could lead to impaired training and understanding by the user. For instance, imagine that a novice is being trained to grasp a virtual egg with an appropriate level of force to lift it without crushing it. The task forces in this environment are calculated based on the pressure applied to the egg, which is a function of the displacement of the fingers. A potential training scheme would then be to calculate the error between the actual displacement of the fingers and the optimal displacement, and display a force to the novice in order to correct the error. However, using the traditional proxy model, this guidance force will be unavoidably confounded with the task force and result in confusing feedback and suboptimal training. If the guidance force gain is low, the novice will grasp the egg too tightly or loosely and will essentially be practicing. If the guidance force gain is high, the novice will seek to minimize the apparent force (to "give in" to the guidance) and will grasp the egg with a nearly-but-not-quite-optimal level of "task" force, but will not actually feel that force being rendered (since the rendered force is a combination of the task and guidance forces). Thus, they will not be able to build an internal model of the task in the same way that they could in practice, and their learning of the task could actually be hindered.

Part of the guidance hypothesis is that challenge is integral to the learning process, and a number of "resistive" methods have been developed based on this principle. These methods will be discussed in detail in Section 4.5.

3.4 Shared-Control Guidance

In general, the problem with robot-mediated training is that it has been unable to replicate many of the human-human training and cooperation paradigms that novices are accustomed to. In fact, the type of guidance provided in robot-mediated training is relatively limited and primitive compared to the rich and varied interactions that occur between human experts and novices. Thus, there have been some pushes to more closely emulate human-human training and cooperation paradigms in human-robot or human-robot-human environments.

Traditionally, such as in fly-by-wire aircraft control systems, conflicting control inputs by multiple users are reconciled by simply averaging the inputs. However, Summers et al. [27] showed that this is not necessarily the best cooperation paradigm. For instance, Reed and Peshkin [5] make the following excellent point:

> Averaging the input command is a simple strategy but not necessarily the best combination since each individual's motion will be diluted. Imagine the effect if one pilot attempts to avoid an obstacle by turning to the left while the other to the right: the average effect is straight into the obstacle.

This logic also applies to the traditional guidance schemes described in previous sections.

Nudehi et al. [28] proposed a shared-control scheme for telesurgical training that essentially calculated a control output based on the weighted average of the control inputs of two operators. By adjusting this weight or control authority "α", control could be shifted between the novice or expert surgeon.

Gillespie et al. [1] proposed the use of a virtual teacher, a more active form of guidance than virtual fixtures that instructs novices to perform dynamic tasks by giving them shared control of a task with a virtual expert. O'Malley et al. [29] showed that such shared-control systems were as effective as virtual fixtures at facilitating skill transfer. The model of a virtual teacher proposed by Gillespie et al. replicates real-world teaching methods in order to facilitate skill transfer and reconcile the problem of guidance force corrupting task dynamics. He presents the example of a tennis expert teaching a novice how to swing a racket using hands-on demonstration. There are three ways that this demonstration could occur. In an "indirect contact" paradigm, the expert and the novice grasp the racket in different locations and perform the swing together. In a "double contact" paradigm, the novice grasps the racket while the expert grasps the novice's hand and guides the novice through the swing. In a "single contact" paradigm, the expert grasps the racket and the novice grasps the expert's hand. In the indirect and single contact paradigms, the task forces (those generated by the dynamics of the tennis racket) are simply summed with the guidance forces (those generated by the expert exerting control over the racket). In the double contact paradigm, the forces are separated spatially, with task forces being applied to the novice's palm and guidance forces to the back of his or her hand. Gillespie et al. [1] hypothesized that this double contact paradigm would be the most effective at eliciting skill transfer, because it passes the greatest amount of haptic information to the novice and allows the novice to easily discriminate between guidance and task forces. However, they were not able to conclusively determine whether this paradigm was indeed better than the others.

Chapter 4

Guidance Paradigm Taxonomy

I propose that all guidance paradigms currently implemented in the literature in human-human, human-robot, and human-robot-human training architectures can be classified as one of the five types in this chapter based on three characterizing factors.

The most apparent and important factor that differentiates guidance paradigms is whether they assist or resist the novice in completing the task. Guidance schemes will thus be classified as either "assistive" or "resistive."

The second major distinction that can be made is based on how paradigms reconcile the co-presentation of task and guidance forces. As mentioned in Section 2.3, task and guidance forces should be interpreted by the novice in fundamentally different ways. If the novice cannot clearly distinguish between the two, the guidance forces will alter the perceived dynamics of the task and potentially impair training. Most existing guidance schemes confound task and guidance forces in just such a way by combining them using a simple weighted average function so that both forces can be displayed simultaneously via a single haptic device. I will refer to this traditional method of reconciling task and guidance forces as "gross" guidance.

Finally, many guidance schemes will adjust the relative weights (gains) of these forces over time in response to a novice's performance improvement. Such schemes will be referred to as "progressive."

4.1 Gross Assistance

Classic virtual fixtures are the archetypal example of gross assistance (GA). By their nature, virtual fixtures have to be relatively stiff in order to keep novices from entering forbidden regions of the workspace, and thus guidance forces generated by collisions with virtual fixtures will dominate any extant task forces. Simple spring-damper couplings or attractor potential models used to "pull" novices towards a target are also typically implemented as GA, and can interfere with the perceived dynamics of tasks in a more subtle way than virtual fixtures. Shared-control guidance schemes also sometimes fall into this category, including the indirect-contact and single-contact virtual teacher paradigms proposed by Gillespie et al. [1].

Gross assistance has been shown to be generally ineffective at improving training outcomes compared to practice without guidance. Reinkensmeyer [18] showed in simulation that "continual guidance" (GA) is "never beneficial compared to no assistance". Crespo and Reinkensmeyer [30] showed that "fixed guidance" (GA) produced only "slightly better immediate retention than did training without guidance," but did not show that this improvement was statistically significant. "Triggered" assistance is a type of GA that requires the novice to exert a certain amount of control effort before assistance is provided. There is little evidence in the literature to suggest that this variation is superior to standard GA. O'Malley et al. [31] implemented a force-based triggered mode on the MIME/RiceWrist exoskeleton, while Kahn et al. [32] implemented a displacement-based triggered mode on the ARM Guide, but neither showed any significant improvement over practice for the rehabilitation of stroke patients.

Generally speaking, most of the assistive paradigms discussed in Section 3.3 that can be classified as GA were shown to be ineffective compared to practice without guidance. This negative outcome might have been predicted and explained in part by the "guidance hypothesis" proposed by Salmoni et al. [33], which states that subjects will tend to become reliant on guidance when it is present in order to improve performance instead of relying on "other cues in the task that are important for motor learning". In this case, the "other cues" might be the task dynamics, which are being dominated by guidance forces.

One possible exception to the generally negative efficacy of GA is for tasks that are extraordinarily difficult and for novices in the very early stages of training for a new task. Crespo and Reinkensmeyer [30] showed that there was a significant improvement of the GA groups over the practice groups in the very first stages of training, but that this improvement quickly diminished and became insignificant as training continued.

4.2 Progressive Gross Assistance

Some researchers have attempted to capitalize on this early-stage benefit of GA by decreasing the guidance gains over time as a novice's performance improves. This "progressive gross assistance" (PGA) theoretically allows the novice to make more errors in later stages of training and further refine his or her motor control. Indeed, many of the same studies in Section 3.3 showing that GA was ineffectual also showed that PGA was superior to both GA and practice conditions.

For instance, Reinkensmeyer [18] showed in simulation that "guidance as-needed" (PGA) was superior to both GA and practice without guidance. Crespo and Reinkensmeyer [30] validated these findings using healthy subjects and a steering task. Li et al. [15] showed that "progressive guidance" (PGA) was superior to GA at training subjects to perform a dynamic target-hitting task.

However, PGA has some potential downfalls. First, PGA requires complex gain-reduction algorithms that depend on accurate and objective performance metrics. Choosing the correct algorithm and performance metrics is highly task-dependent and potentially difficult. Additionally, PGA suffers from the same pitfall of traditional GA in that it confounds guidance and task forces during the majority of training, and it may in fact exacerbate this impairment by subtly changing guidance gains (and thus the task dynamics as well) over time.

4.3 Temporally Separated Assistance

The characterizing factor of temporally separated assistance (TSA) is that it separates guidance and task forces temporally, displaying each type alternately in quick succession via a single haptic device. Novices primarily experience unadulterated task forces, but training is frequently (on the order of 1 Hz) punctuated by brief periods of pure guidance, intended to "cue" novices as to the appropriate movements to make. In this way, the expert exerts "cognitive dominance" over the novice, while allowing the novice to retain "physical dominance"- in other words, allowing a novice to commit errors and actively generate movement plans in order to better learn the task dynamics. With this advantage, I hypothesize that TSA can achieve the same level of performance as PGA without being subject to the complexities of adaptive algorithms. Additionally, compared to progressive paradigms that provide all of the guidance during training "up front," TSA provides guidance consistently and predictably throughout training, hopefully improving training outcomes.

Endo et al. [34] are the only group known to have proposed and tested a TSA paradigm. In a pilot study, they showed that TSA was effective at training participants to grip a virtual object using proper grasping forces and fingertip placements. However, they did not study its effectiveness at training for dynamic tasks, and I could find no other implementations of TSA in the literature.

4.4 Spatially Separated Assistance

Whereas TSA separates the presentation of task and guidance forces temporally in order to present them via a single haptic channel, spatially separated assistance (SSA) makes use of two haptic channels in order to present task and guidance forces simultaneously via the separate channels. The first and perhaps best example of SSA is the double-contact paradigm proposed by Gillespie et al. [1]. As described in Section 3.4, this paradigm makes use of a specialized haptic device in order to present guidance from a virtual expert via one haptic channel (through the back of a novice's hand) and forces arising from the task dynamics via a second channel (through the novice's palm). The results of this study were inconclusive as to whether SSA was superior to practice conditions.

Similarly, Wulf et al. [35] showed that a weak form of SSA was superior to practice without physical guidance at training novice's to perform a simulated skiing task. This might be considered a "weak" form of SSA because haptic feedback was provided via actual mechanical fixtures rather than electromechanical systems and a virtual expert. However, this guidance paradigm still qualifies as SSA because guidance was provided via a spatially distinct channel (i.e. the poles) from the primary interface with the simultator (i.e. the simulated skis).

There are no other known implementations of SSA in the literature, likely due to the relative complexity and propriety of the haptic devices necessary to implement e.g.

the double-contact paradigm. While replicating a real-world teacher in this way is an elegant and intuitive approach to implementing SSA, the utility of the double-contact paradigm is limited to cases where the physical constraints of the task being taught allow for this specific type of spatial separation of forces. Presenting forces in this manner effectively requires haptic devices with up to twice as many degrees of actuation and significantly higher complexity. In some cases, presenting forces in this manner may simply not be possible given the physical constraints of the task. Providing guidance and task feedback via separate but identical haptic devices might be a more feasible solution. This method of spatial separation is tested in this study.

4.5 Gross Resistance

Gross resistance (GR) can take a number of different forms, but is generally characterized by increasing the difficulty of a task or resisting a novice's optimal completion of a task in some way. The theory behind GR is simply based on over-training: after training extensively in the presence of artificial resistance, novices will find it relatively easy to execute the same task in the absence of the resistance. There are three common implementations of GR: as a constant force-field or viscous force opposing movement, as a force that augments errors, or as forces producing random disturbances.

Constant (Coulomb) or velocity-dependent (viscous) forces opposing the direction of movement have been shown to improve training outcomes particularly in the field of rehabilitation. For instance, Lambercy et al. [36] designed a haptic knob offering varying levels of resistive force in order to help stroke patients regain grasp strength and coordination. A meta-review by Morris et al. [37] showed that resistance training (though not necessarily robot-mediated) can help reduce musculoskeletal impairment in stroke patients.

Error augmentation has also been shown to improve training by taking advantage of the CNS' error-driven learning process. Emken and Reinkensmeyer [38] showed that amplifying the task dynamics and in turn producing larger movement errors improved the adaptation of healthy novices to a viscous force-field. In rehabilitation, Patton et al. [39] showed that force-fields that amplified the movement errors made by stroke patients in a reaching task improved training outcomes compared to practice.

Finally, Lee and Choi [40] showed that training in the presence of random noise-based disturbance was superior to PGA and practice at training healthy novice's to perform a path-following task. Such noise-based GR has not been discussed elsewhere in the literature and is a prime candidate for further evaluation.

Chapter 5

Dynamic Task Platform

The Dynamic Task Platform (DTP) is an object-oriented, modular, and extensible software package written in C++ that allows for the rapid development of human studies involving dynamic tasks. This platform is unique among similar rapid-prototyping haptic packages such as CHAI3D in that it is designed from the ground up to handle the specific needs of human studies, such as high-frequency data collection and robust haptic rendering even on a non-realtime operating system. It easily accomodates various experimental designs, automatically tracks subjects' performance and handles their progression over the course of several sessions, accommodates any number of experimental groups and conditions, and supports multiple user input and output methods (such as mice, multiple simultaneous haptic devices, and multiple displays).

5.1 Robust Haptic Rendering in a Non-Realtime Environment

Ensuring robust and high-fidelity rendering in haptic-enabled environments is particularly challenging due to the relatively large disparity between the computational complexity and necessary loop rates of the different feedback mechanisms that must be present. For instance, in a typical haptic simulation, advancing the physical simulation and computing haptic feedback might be computationally simple, but must occur at extremely high loop rates (1000 Hz) in order to ensure a stable and high-fidelity simulation. Conversely, rendering the visual output is very computationally intensive even for simple environments but need only occur at a frequency of 30 Hz to 60 Hz. Various other components of the simulation (such as data logging) might need to run at intermediate frequencies.

The DTP maintains high simulation loop rates (1000 Hz) by rendering visual output in a separate thread from the simulation. The "haptic" thread is thus responsible for advancing the physical simulation, performing collision detection, logging data, rendering haptic feedback, and controlling program flow at a fixed loop rate, while the "display" thread renders the corresponding visual output on a "best-effort" basis (nominally 60Hz). The display thread carefully culls information from the physical simulation in a non-blocking manner while using semaphore locks to ensure thread-safety.

5.2 Ensuring Experimental Integrity

The DTP has a number of features and specific design considerations intended to ensure overall experimental integrity by guaranteeing stable loop rates (to the greatest extent possibility on a non-realtime operating system), maintaining a consistent testing environment between sessions, reducing the chance of human error, and continuously monitoring all of these safeguards so that anomalies can be brought to the attention of the experimenter.

As mentioned, high loop rates are ensured by running time-critical tasks in separate threads (and on separate processors, if available). Special attention is also given to high-frequency data logging, since disk input and output is a traditional bottleneck in high-performance computing and cannot be threaded in the same manner. Access times for typical modern hard disk drives average around 10 ms, but can grow suddenly and unpredictably if another process is writing to disk. This severely limits the rate at which data can be reliably logged without impacting the overall performance of the program. The DTP overcomes this limitation by buffering data in system memory and only accessing the hard disk during periods of inactivity, such as the brief period between trials. A separate watchdog process monitors the haptic, display, and data logging loop rates and immediately pauses the experiment if a significant drop in loop rate is detected.

The most common source of error during an experiment is simply human error, and thus the DTP is designed to avert some of the most common sources of human error. For instance, the entire experimental progression, from trial to trial and session to session, is computer controlled. When starting a session, subjects only need to enter a single piece of data (a user ID), which is then validated and used to configure the session based on parameters provided by the experimenter. Additionally, metadata on the progress and outcomes of each session is stored separately from the bulk low-level data, allowing for cross-checking and validation. Calibration of the haptic devices is performed automatically and without any human intervention, while special algorithms simultaneously check for adequate power, loose capstans, and worn out bearings. Finally, the color scheme is carefully chosen to take into account the most common types of color-blindness, and colors are used with consistent meanings throughout the tasks in order to make task objectives as clear and intuitive as possible.

5.3 Future-Proofing and Accessibility

The DTP relies on a set of standard design methodologies and open source tools that in turn ensure a high degree of portability, accessibility, and sustainability by future researchers.

Instead of being written directly in an integrated development environment (IDE) such as Microsoft Visual Studio (MSVS), DTP is packaged as a set of C++ source files and related resources. Accompanying these source files are a set of "make" files that can be used by a number of open-source programs to generate project files compatible with nearly any build environment. For instance, CMake can be used to generate MSVS solutions for MSVS 6, MSVS 2005, MSVS 2010, and presumably any future version of MSVS as well. Thus, multiple researchers collaborating on the same task are no longer required to have the same IDE or even the same operating system! The open-source program Doxygen can also be used to generate comprehensive documentation (including automatically generated class diagrams, inheritance graphs, etc...) in formats such as PDF and HTML.

Finally, besides the drivers and libraries necessary to interface with individual haptic devices, the only third-party code that DTP relies on is *freeglut*, an open-source and cross-platform implementation of the OpenGL Utility Toolkit (GLUT) with wider platform support and active development.

5.4 Simple Construction of Tasks and Experiments

Because of its object-oriented design, DTP is easy to customize to suit a wide variety of tasks and experiments.

Each experimental session is broken down into a number of phases and trials. Trials are typically 20-30 seconds in length, with a brief pause between each. A series of similar trials is considered a "phase" of the experiment; between phases there is a brief pause and chime that alerts subjects to a change in experimental conditions. Creating the desired experimental structure is accomplished by simply deriving from base phase and trial classes, and then adding phases to a session using simple statements or logical conditions (for instance, based on a user's experimental group). Data from each trial is logged in a tabular format, and any class in the program can at any point add data to the log by simply specifying a column name.

Each task environment is built from a combination of three simple "primitives": nodes, linkages, and constraints. Nodes are any object that can be interacted with by the user, including masses, targets, and haptic devices. Nodes are coupled by linkages, which are spring-damper couplings that can dynamically modulate force output based on experimental conditions. Finally, constraints bind nodes to move in proscribed manners, for instance reducing a two degree-of-freedom task to one degree-of-freedom by constraining the novice to move along a straight line. Of course, any of these objects' base class can be derived in order to support new haptic devices and build new tasks.

5.5 Rapid Real-Time Physics Simulation

The DTP physics engine ensures high-fidelity real-time simulation by using a simple Newtonian particle-based physics model. "Real-time" in this context indicates that the simulation is advanced at a one-to-one ratio with reality with each iteration. Each node's state is described by a position vector x, velocity vector v, and body force vector F. The body force F is a sum of the forces due to gravity, friction, linkages, and constraints(F_g, F_f, F_l, F_c, respectively). With each timestep (1 ms), the state of each node is updated according to the following equations:

$$\begin{aligned} dt &= t_1 - t_0 \\ F(t_0) &= F_g + F_f + F_l + F_c \\ a(t_0) &= \frac{F(t_0)}{m} \\ x(t_1) &= x(t_0) + v(t_0)dt + \frac{1}{2}a(t_0)dt^2 \\ v(t_1) &= a(t_0)dt \end{aligned}$$

The force exerted on each node by a linkage is calculated according to the linkage stiffness k, damping b, and scaling factors R_a and R_b. In a real-world context, the forces exerted by each end of the linkage must be equal and opposite, but in the virtual environment the forces can be scaled using these scaling factors. For instance, if the expert is "node a" and the shared proxy (S.P.) is "node b" (as shown in Figure 6.1), then $R_a = 0$ and $R_b = 1$, because the expert-S.P. linkage only transmits force information unilaterally (from the expert to the S.P.). The forces exerted on nodes connected by a linkage (F_a and F_b) are calculated according to:

$$\begin{aligned} F_l &= (x_b - x_a)k + (v_b - v_a)b \\ F_a &= (-1)F_l R_a \\ F_b &= F_l R_b \end{aligned}$$

Finally, because most haptic devices provide only a position and not a velocity reading with each iteration, velocity must be calculated by differentiating the position signal. This can be problematic at high sampling frequencies (such as 1000 Hz), leading to noisy and highly non-continuous velocity readings. Thus, a low-pass filter with a cutoff frequency of 16 Hz is applied to produce continuous and steady velocity

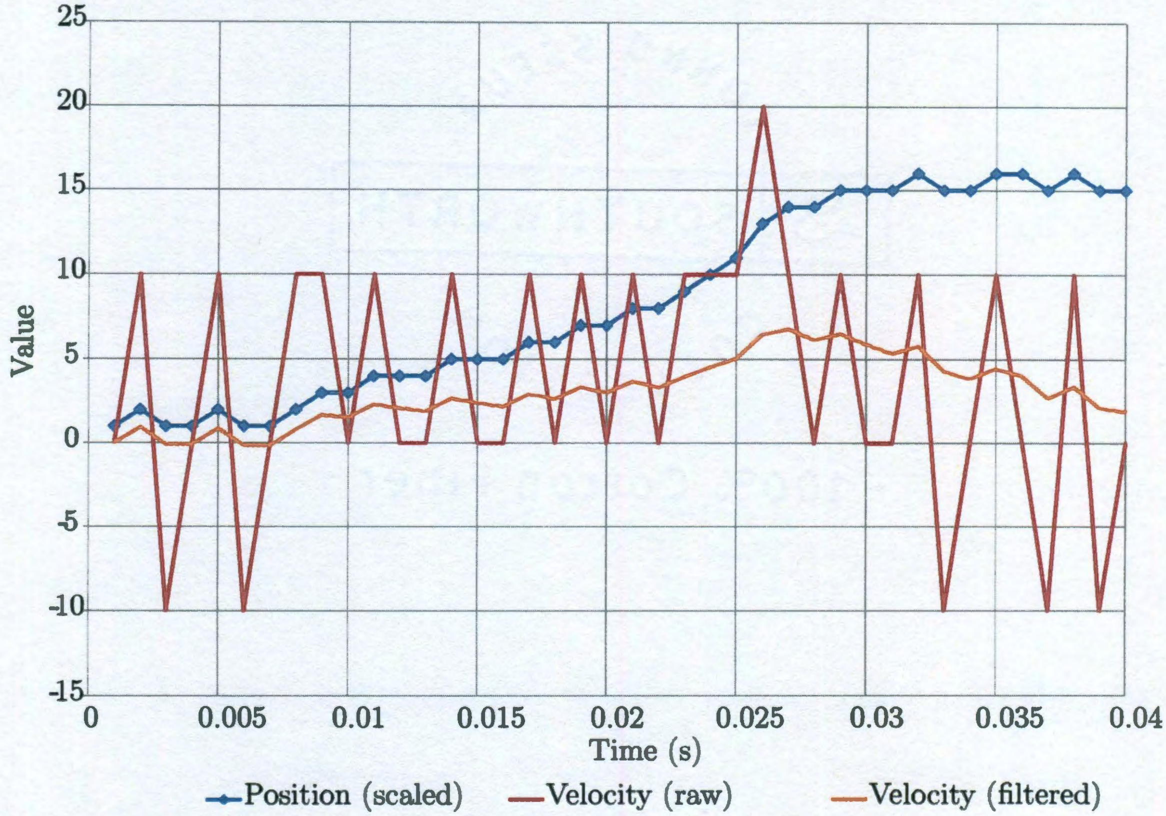

Figure 5.1: Effect of a low-pass filter on velocity readings at high sampling rates. Note how the raw velocity (calculated using the forward-difference method) is noisy and very discretely valued, while the filtered velocity is a nearly continuous function. Note that this accurately represents a low-pass filter with a 16 Hz cutoff frequency, but is not based on actual position data.

values, as shown in Figure 5.1. Kuchenbecker et al. [41] showed that encoder noise tends to dominate inputs from the human hand at frequencies above 30 Hz, and for this combination of haptic device and task it was expected that noise would dominate above roughly 16 Hz. The discretized time-domain form of this filter is $v_1 = 0.09516(x_1 - x_0) + 0.9048v_0$, where v_0, v_1, x_0, and x_1 are the old and new velocities and positions at each timestep.

Chapter 6

Shared-Control Proxy Model

A number of factors make stable rendering of the interaction between a haptic interface and virtual environment a non-trivial task. Foremost among these are the physical limitations of even the most modern haptic devices, which tend to be relatively compliant compared to the virtual objects that they interact with. In order to maintain a one-to-one relationship between the position of the haptic device in real space and in the virtual environment, the device would have to penetrate unrealistically far into the virtual object. Thus, direct calculation of interaction forces based on a physics model is generally not possible, as the forces would tend to saturate quickly enough to lead to explosive instability, and some other general haptic rendering algorithm is required.

Zilles and Salisbury [42] proposed a "constraint-based god-object" rendering algorithm (commonly referred to as a "proxy model") for calculating and displaying interactions between a haptic interface and a virtual environment. In this model, a massless "god-object", "avatar", or "proxy" represents the user in the virtual environment, and must obey all of the physical constraints of the virtual environment (i.e. walls, friction, etc...). This proxy is then connected to the haptic device by a virtual spring and damper coupling, and the force output to the device is simply calculated based on this coupling (usually amplified by a constant gain). This allows the haptic device to penetrate virtual surfaces without necessarily leading to instability or requiring a specialized physical model.

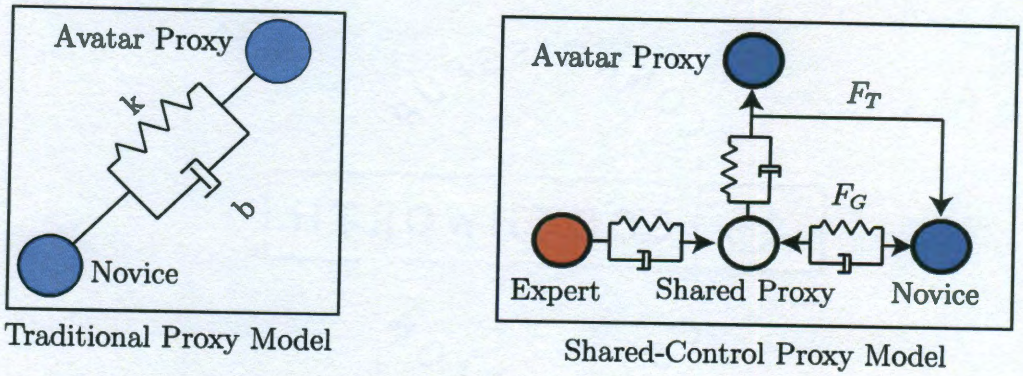

Figure 6.1: Traditional and Shared-Control Proxy Models.

As mentioned in Section 2.3, one potential problem with the traditional proxy model is that it cannot discriminate between guidance and task forces in shared-control systems, and thus the forces are confounded when displayed to the user. This could lead to impaired training and understanding by the user.

The proposed shared-control proxy model overcomes this deficiency by adding a second proxy and replacing the traditional spring-damper couplings with a series of "biased" spring and damper couplings. Whereas traditional couplings can only exert equal and opposite forces on attached nodes, biased couplings can exert opposite but arbitrarily scaled forces on each node and are only realizable in a virtual environment, as they essentially break Newton's Third Law. These couplings link the novice, expert, "shared proxy", and "avatar proxy" as illustrated in Figure 6.1, where arrows indicate the general directions of force transfer (in other words, the end of the coupling with a higher force gain).

The massless shared proxy's position is influenced equally by the expert and the novice, but is not influenced at all by the position of the avatar proxy, nor does it interact with the virtual environment. Because of this, the shared proxy remains exactly between the novice and expert at all times, representing the averaged input of the novice and expert. Note that this average could be weighted in order to adjust the control authority α (as proposed by Nudehi et al. [28]) by simply changing the relative stiffnesses of the expert and novice couplings. The force generated by the coupling between the novice and shared proxy represents a pure guidance force F_G, since it is proportional to the deviation from the expert and unaffected by the virtual environment. Note that in this case, the expert will not receive any force feedback and thus will not be affected by the novice, which is the logical setup for tasks with a virtual expert. However, with a human expert present, force-feedback could be provided in a way similar to how the novice receives force feedback.

The avatar proxy must obey all of the constraints of the virtual environment and is coupled to the shared proxy, so that in free space both proxies ideally share the same position. However, when the user comes into contact with a virtual surface, the invisible shared proxy will penetrate the surface to the same extent as the haptic device, while the avatar proxy will remain outside the surface. The force generated by the coupling between the two proxies then represents a pure task force F_T, since it is proportional to the deviation between the commanded position of the shared proxy and actual position of the avatar proxy.

The presentation of guidance and task forces to the novice can now be modulated in any of the manners discussed in Chapter 4.

Chapter 7

Evaluation of Four Guidance Paradigms

7.1 Methods

7.1.1 Experimental Design

Four prototypical guidance schemes chosen from the guidance taxonomy proposed in Chapter 4 were implemented on the Dynamic Task Platform, and their effectiveness at training subjects to perform dynamic tasks was evaluated in a 16-subject pilot study and 50-subject primary study. Subjects trained with the assistance of a virtual expert using the shared-control proxy model described in Figure 6.1, which allowed for the discrimination of task and guidance forces.

Evaluation and Training Trials Subjects performed the tasks over a number of trials. Each trial was 20 seconds long and generally categorized as either an "evaluation" trial or a "training" trial. In evaluation trials, participants had sole control over the system via a single joystick and were instructed to perform the task to the best of their ability. During training trials, a virtual expert was also present in the system. This expert followed a predefined optimal trajectory for each task, and shared control of the system with each participant under one of the experimental conditions. Participants were instructed to track the expert as closely as possible during training, and by exactly matching the expert they could achieve the best score possible in each task.

Structure Participants performed a target-hitting task and path-following task on two consecutive days, with a single session and type of task per day. The order of task presentation was balanced between groups. Each session consisted of 106 trials grouped into a number of different blocks or "phases." Evaluation phases consisted of three evaluation trials, training phases consisted of 12 training trials, and a final generalization phase consisted of 12 generalization trials. Generalization trials were similar to evaluation trials but with slightly modified task parameters, and were intended to test the robustness of acquired motor skills to changing task dynamics. Participants were also allowed a one-minute familiarization trial before attempting each task for the first time. During this trial, subjects completed a representative but substantially different and easier version of the task, which allowed them to become familiar with the task without developing any significant task-specific skills. The structure of each session is illustrated in Table 7.1.

Workloads At the end of each session, participants also reported their perceived workload during the task by completing the NASA TLX questionnaire developed by Hart [43]. This questionnaire allows participants to rate their perceived workload on six different sub-scales: mental demand, physical demand, temporal demand, performance, effort, and frustration. It then lets them weight the contributions of each type of workload to the overall workload, and uses this information to compute a weighted average of the overall workload.

Pilot Study The purpose of the pilot study was to gather preliminary information about how to best organize the experiment and implement the guidance paradigms.

Trial type	F	E	T	E	T	E	T	E	T	E	T	E	T	E	G
Number of trials	1	3	12	3	12	3	12	3	12	3	12	3	12	3	12

Table 7.1: Order of trials in each session: Familiarization (F), Evaluation (E), Training (T), and Generalization (G) trials. Note that there was a mandatory 5-minute break midway through each session.

The size of the study was too small to draw significant conclusions about the paradigms, and the design changed too much between the pilot and primary studies for the data to be pooled; thus, the results of the pilot study are not included in this work.

A total of 16 participants enrolled in the pilot study. Participants performed a target-hitting task over the course of 10 sessions on consecutive days, with each session consisting of 30 trials and thus taking 10-15 minutes. This configuration of sessions was determined to elicit the fastest training in a pre-pilot study as compared to fewer sessions of greater length or more sessions of shorter length. Each session consisted of five evaluation trials ("pre-evaluation" trials), then twenty training trials, and finally five more evaluation trials ("post-evaluation" trials). After analyzing the results of the pilot study, it was determined that this configuration overcomplicated the data collection and analysis procedures, and thus the primary study was organized into only two sessions (one for each task).

7.1.2 Subjects

A total of 50 participants enrolled in the primary study, and were divided evenly between 5 experimental groups: no guidance, GA, TSA, SSA, and SR. Five participants were left-handed, 45 right-handed, 33 male, and 17 female. All participants controlled the task with their dominant or preferred hand. All participants provided their in-

Guidance	Force (Joystick 1)		Force (Joystick 2)
Control	$F_T(t)$		0
GA	$F_T(t) + F_G(t)$		0
TSA	$F_T(t) + sin(\frac{t*\pi}{t_0})F_G(t)$	if $t \bmod t_1 \leq t_0$;	0
	$F_T(t)$	if $t \bmod t_1 > t_0$.	
SSA	$F_T(t)$		$F_G(t)$
GR	$F_T(t) + F_{PN}(t)$		0

Table 7.2: Force outputs for guidance conditions. The task force F_T is composed of forces inherent to the task environment, such as from the swinging mass in the target-hitting task. The guidance force F_G is a perceptual overlay intended to guide the novice towards the expert's position. Both F_T and F_G are calculated as shown in Figure 6.1 using the rendering algorithms described in Section 5.5. The resistive force F_{PN} is calculated according to the Perlin noise function shown in Figure 7.1. For the purposes of these experiments, $t_0 = 100\,\text{ms}$ and $t_1 = 500\,\text{ms}$.

formed consent as approved by the Rice University Institutional Review Board, had no significant visual or motor impairments and no or little prior experience with virtual dynamic target-hitting tasks. In order to encourage subjects to perform to the best of their ability and follow the given instructions, gift cards were awarded to each subject that scored highest in evaluation trials and followed the expert the closest in training trials.

7.1.3 Guidance Conditions

The mathematical representations of the guidance paradigms used during training trials are given in Table 7.2.

No Guidance (Control) Only task forces were displayed. Thus, participants could track the expert visually on-screen but received no haptic indication of his position. This served as a control condition.

Gross Assistance (GA) Task forces and guidance forces were combined by simple summation and presented via a single joystick. The two types of forces were scaled so as to each have a peak magnitude of about half of the maximum force output level of the joystick.

Temporally Separated Assistance (TSA) Task forces were displayed at all times, and guidance forces were overlaid in 100 ms sinusoidal pulses at a frequency of 2 Hz (the optimal frequency and ratio as experimentally derived by Endo et al. [34]). Participants described these guidance forces as "pulsating" and interpreted them as nudges or resistance that indicated the direction that they should be moving. The pulses were not frequent enough or large enough in magnitude to exert significant control over the task; thus, this mode prevented participants from becoming reliant on guidance forces, a problem described by Li et al. [44].

Spatially Separated Assistance (SSA) Participants in this group used two joysticks during the experiment. Participants controlled the system using the primary joystick, onto which only task forces were displayed. Guidance forces were displayed on the secondary joystick so that its trajectory matched that of the expert's, also visible on-screen. Participants were instructed to lightly grasp this joystick with their non-dominant hand and to replicate the movements displayed there on the primary joystick. This allowed participants to intuitively mimic the expert's trajectory while still experiencing undistorted task dynamics. This paradigm also shares with temporal separation the advantage of forcing the participant to take control of the task and not rely on the expert to do any "heavy lifting".

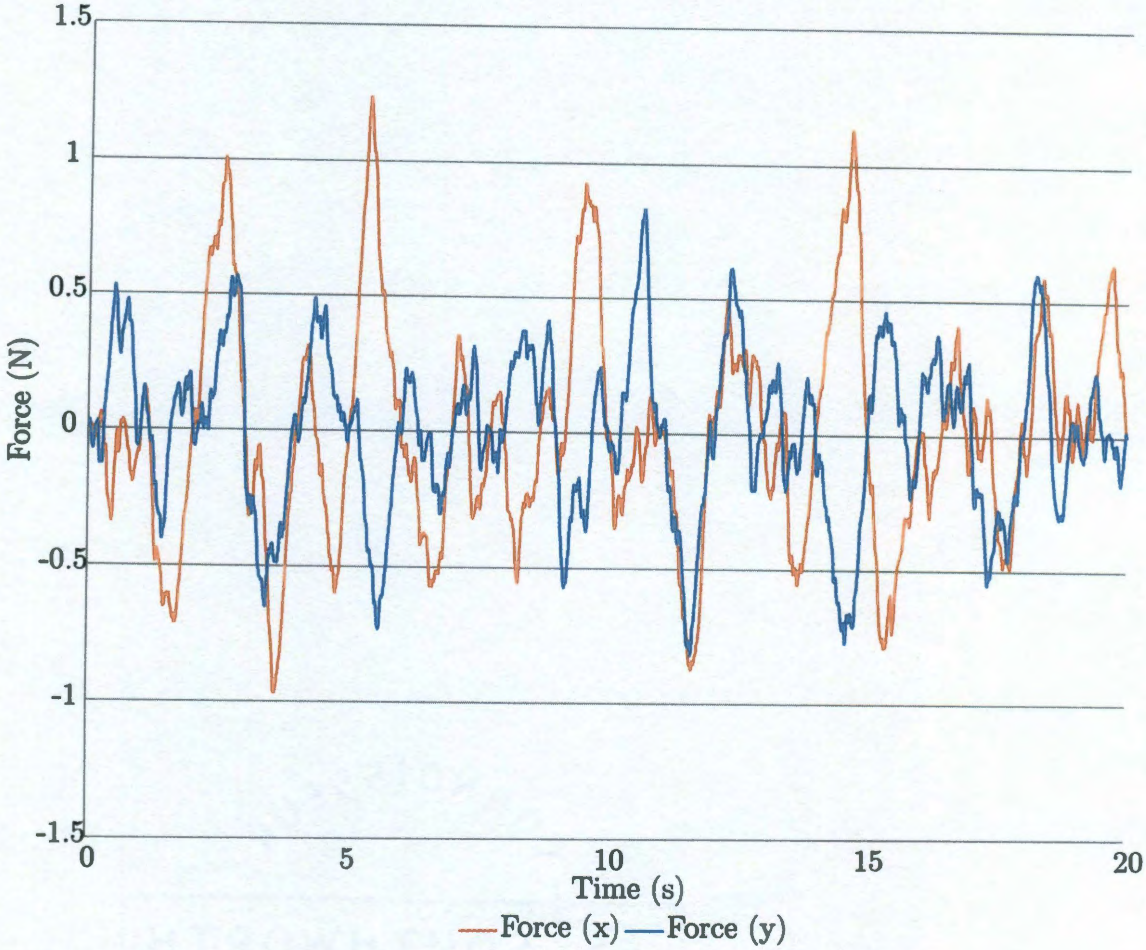

Figure 7.1: Joystick force output from Perlin noise function under GR condition.

Gross Resistance (GR) Task forces were combined with a randomly-generated disturbance force in the manner described by Lee and Choi [40]. A Perlin noise function with a nominal range of $-1.2\,\mathrm{N}$ to $1.2\,\mathrm{N}$ was randomly generated for each joystick axis as shown in Figure 7.1 using the open-source *libnoise* library. At each timestep, the guidance force was generated from the values of these functions and summed with the task force to produce the net force displayed to the joystick.

7.1.4 Tasks

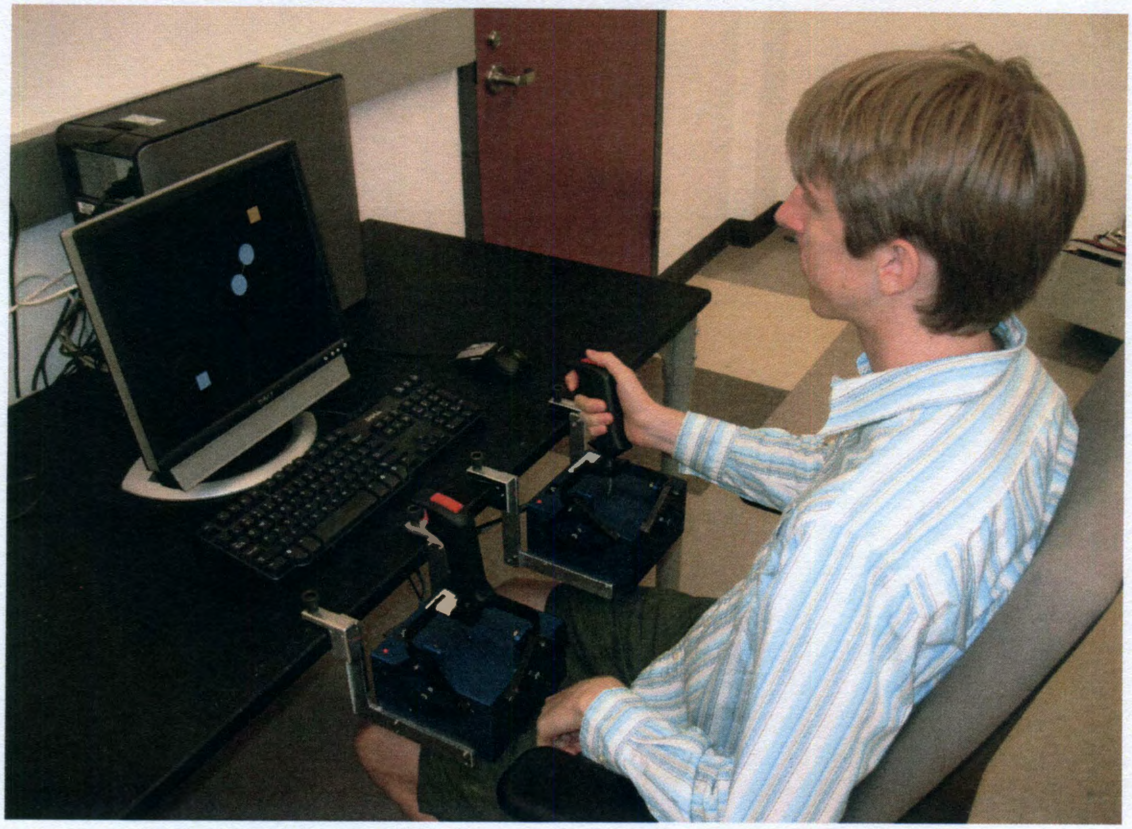

Figure 7.2: A participant performing a target-hitting evaluation trial in a pilot study. Note that screen objects have been enlarged 10x for illustrative purposes.

Target-Hitting Task The target-hitting task used in these experiments was based largely on a task originally used by O'Malley et al. [29], O'Malley and Gupta [45]. Participants controlled the position of an on-screen pointer using a 2-DOF haptic joystick (Immersion, Inc.'s IE2000), as shown in Figure 7.2. This was connected to a 5 kg mass by a spring with stiffness $k = 100\,\mathrm{N/m}$ and damping $b = 3\,\mathrm{Ns/m}$, as shown in Figure 7.3. Details about how this spring-mass system was rendered can be found in Section 5.5. Thus, participants could control the position of the mass only indirectly. Two targets were positioned equidistant from the center of the screen and at a 45° angle to the horizontal. At any given time, one target was inactive (blue) and the other active (orange). The active target could only be "hit" by the swinging

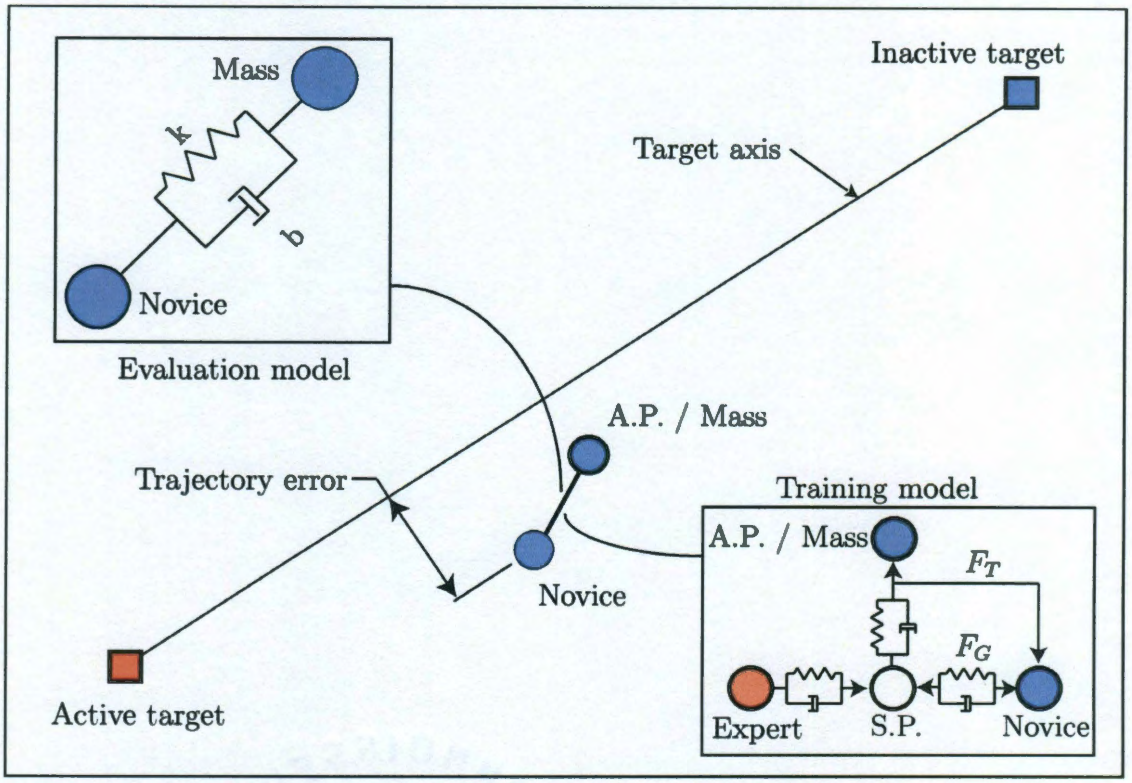

Figure 7.3: Task layout and dynamic models for evaluation and training trials. Participants control a pointer that is linked to a mass-spring-damper system and try to guide the mass to alternating targets. In evaluation trials, a simple spring and damper link the mass and the novice. In training trials, a series of "directional" spring and damper systems link the novice, expert, proxy, and mass, where arrows indicate the directions of force transfer.

mass, at which point the opposite target would become active. Each task trial was 20 seconds long, and the goal during evaluation trials was to hit as many targets as possible in this time frame. Thus, by moving the pointer at the resonant frequency of the system (0.71 Hz) along a straight line connecting the targets participants could achieve the highest hit-count possible (approximately 28 hits).

During training, participants shared control of this system with a virtual expert, represented on-screen by an orange pointer that tracked the optimal trajectory (a

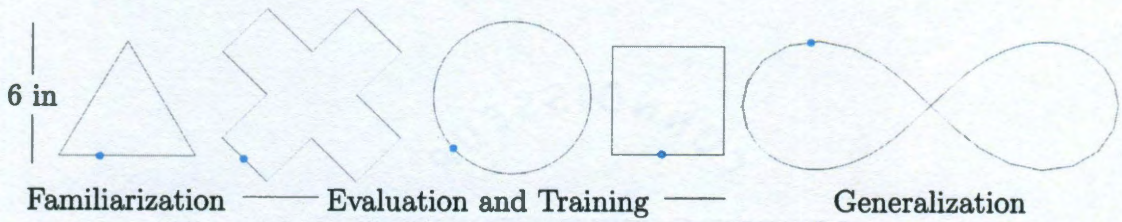

Figure 7.4: Shapes used in the path-following task.

straight-line path between the targets at a frequency of 0.71 Hz). Shared control was implemented via the shared-control proxy model described in Figure 6.1.

During evaluation trials, participants were instructed to hit as many targets as possible, while in training trials they were instructed to follow the expert as closely as possible. Gift cards were awarded to the subjects that best achieved each of these goals.

In the primary study, the target sizes were doubled (as compared to the pilot study) in order to facilitate greater performance improvement over a smaller number of trials.

Path-Following Task The path-following task was similar to a traditional pursuit-rotor task in that it required novices to track a virtual expert as closely as possible as the expert traced the outline of a simple shape at a constant speed. The task is based loosely on that proposed by Lee and Choi [40]. In evaluation and training trials, subjects traced the outline of one of the three shapes shown in Figure 7.4 (a circle, square, or X). In any given phase, the shapes would be presented in equal number but a random order. In the familiarization and generalization phases, subjects were shown triangles and lemniscates (respectively). In all cases, the goal was to trace the expert as closely as possible, and thus performance was defined as cumulative deviation from the expert's position (in centimeters) over the course of each trial. Gift cards were awarded to the subjects that achieved the lowest deviation.

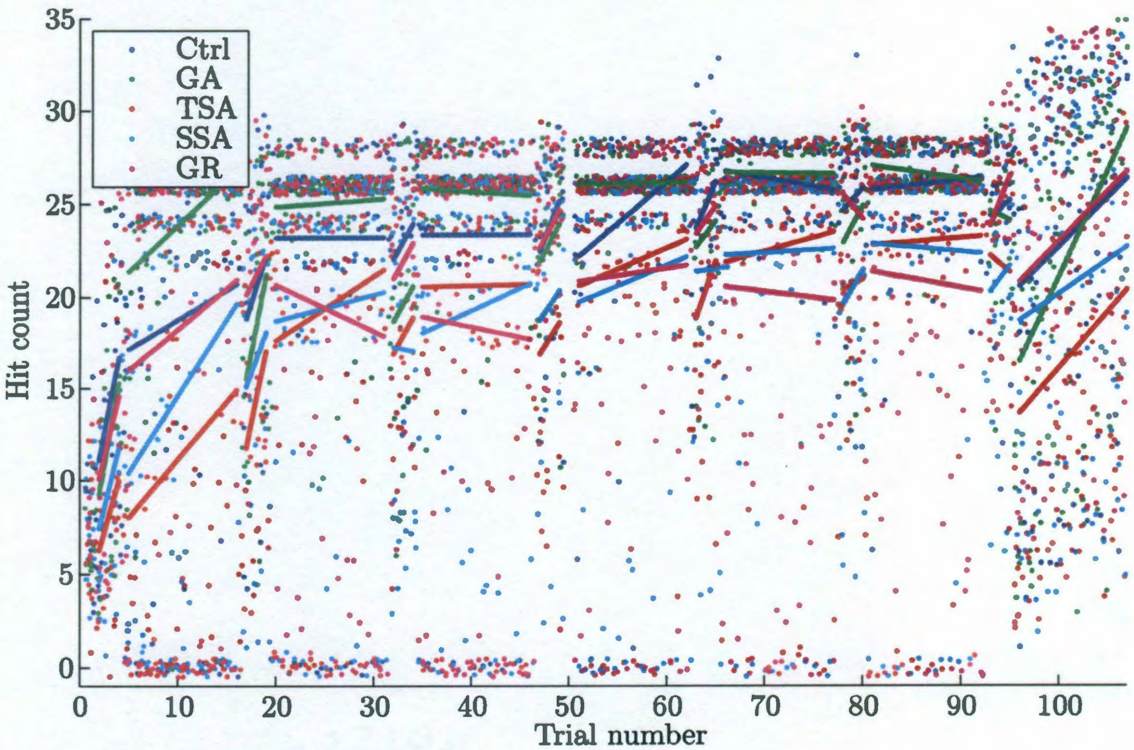

Figure 7.5: Target-hitting task: Hit counts achieved by subjects versus trial number. Each point represents the number of hits achieved by a subject in a given trial. Subjects are color-coded by group. Lines of best fit are shown for each combination of phase and group. The shorter phases are evaluation phases (consisting of three evaluation trials), the longer phases are training phases (consisting of twelve training trials), and the final phase is the generalization phase. A small amount of "noise" has been introduced in the plotting process to make the data appear more continuous and prevent multiple data points from directly overlapping one another.

7.2 Results

The raw data collected in tabular flat files during experimental sessions was processed using Matlab to produce a set of data coded at the level of individual trials. This data includes such information as the trial number, trial type, number of hits achieved during the trial ("hit count") for target-hitting trials, and cumulative deviation ("deviation") in centimeters for path-following trials. Outliers were defined for each cell

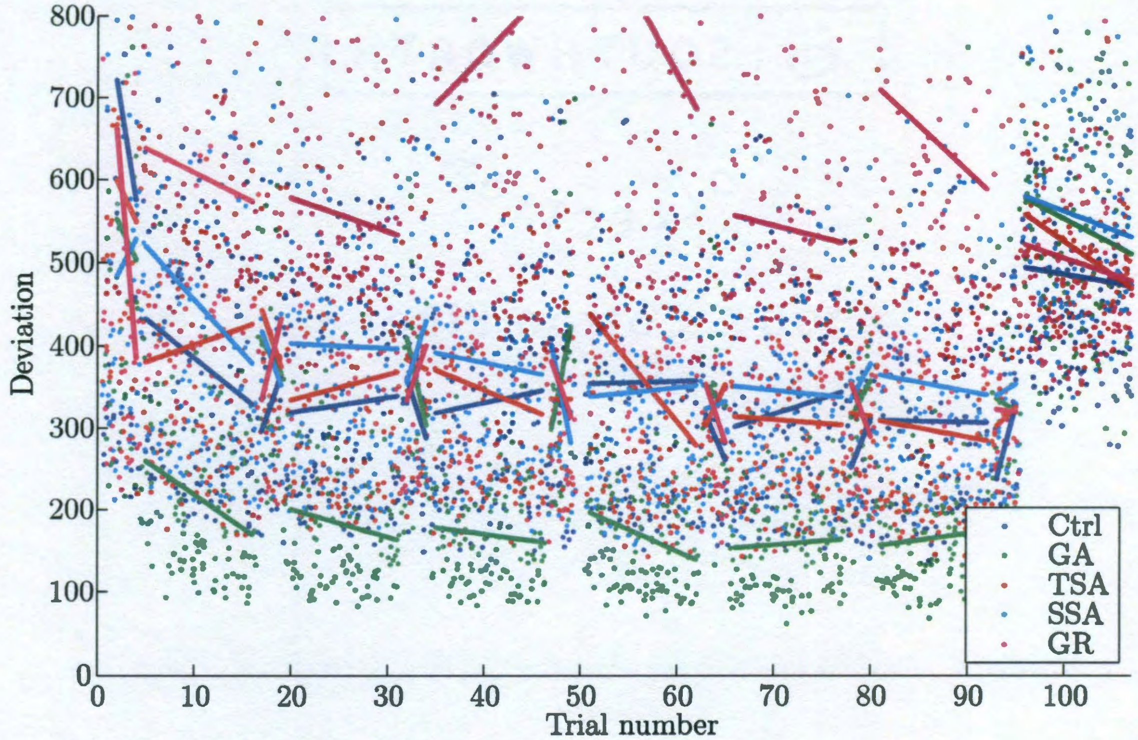

Figure 7.6: Path-following task: Cumulative deviation (cm) of subjects versus trial number. Each point represents the cumulative deviation (in centimeters) accrued by a subject in a given trial. Subjects are color-coded by group. Lines of best fit are shown for each combination of phase and group. The shorter phases are evaluation phases (consisting of three evaluation trials), the longer phases are training phases (consisting of twelve training trials), and the final phase is the generalization phase. A small amount of "noise" has been introduced in the plotting process to make the data appear more continuous and prevent multiple data points from directly overlapping one another.

(each unique combination of group and trial) as points further than 1.5 interquartile ranges (IQR) from the cell mean. In repeated-measures data analysis, outliers are often replaced with the outlying subject's mean or with the grand mean. However, both of these means tended to be even more extreme than the outliers themselves, so outliers were instead replaced with their respective cell mean.

This trial-level data is shown for target-hitting trials in Figure 7.5 and for path-following trials in Figure 7.6. Plotting the data initially at this low level allows for many valuable observations and insights:

- Horizontal striations in target-hitting training data can be noted for subjects achieving between 18 and 30 hits, and indicate that subjects nearly always achieved an even number of hits during training trials. While the exact process leading to these striations is difficult to explain, it's generally suspected to be due to the oscillatory nature of the task and the consistent initial state of the expert between trials. It is not suspected to be due to any error in the experiment or data collection process. However, its existence is potentially detrimental to the analysis of training trial data, as it effectively reduces the number of possible hit count outcomes for most subjects in the later stages of training from 6-8 to 3-4.

- Subjects' performances tend to increase dramatically over the course of individual evaluation phases, while increasing comparatively little during training phases. Note that this applies to the control group as well as other groups. This would seem to indicate that subjects are recovering from some detrimental impact of the transition between evaluation and training trials, rather than actually improving their performance in a global sense during evaluation trials. Due to this observation, many of the data analysis procedures described below were also attempted with the first evaluation trial in each phase removed, in order to obviate this interference effect, but this did not show any improvement in the consistency of the data or significance of the results.

- The GA group's training performance improves and plateaus quickly, which qualitatively matches the behavior of the GA group observed in prior studies.

- Performance during training follows a bimodal distribution, with many subjects scoring zero hits even during the final training trials. It is helpful and germane to remember that subjects were instructed to focus only on following the expert during training, and not necessarily to maximize hit counts. Thus, this might indicate that not all subjects were following the instructions, or that there are actually multiple distinct populations of subjects being tested based on natural aptitude. Prior studies and pilot studies have shown that some participants can be classified as "natural experts" based on high entrance scores, while others are "poor learners" and remain at a low level of performance throughout training. Regardless of the explanation, these results make certain kinds of analysis difficult or impossible because each type of subject might show little or no performance improvement over the course of training, and additionally the variance of the cumulatively sampled population is very high. Due to this observation, many of the data analysis procedures described below were also attempted on smaller subsets extracted from the main data set based on the initial performance of subjects at entry into the study. However, this did not improve the significance of the results.

Two general types of statistical analysis were employed for evaluating subjects' performance: mixed ANOVAs on trial data and one-way (between-subjects) ANOVAs on curve-fit parameters, both followed by family-wise error-corrected multiple com-

Group	Group	Diff.	p (raw)	p (adj)	Sig.
Ctrl	GA	2.04	.081	.402	
Ctrl	TSA	5.99	$< .001$	$< .001$	*
Ctrl	SSA	4.89	$< .001$	.001	*
Ctrl	GR	1.21	.300	.836	
GA	TSA	3.95	.001	.008	*
GA	SSA	2.84	.016	.110	
GA	GR	−0.83	.473	.952	
TSA	SSA	−1.11	.343	.875	
TSA	GR	−4.78	$< .001$	.001	*
SSA	GR	−3.68	.002	.017	*

Table 7.3: Target-hitting task: Multiple pairwise comparisons for evaluation trials. Adjusted p-values use the Tukey-Kramer adjustment.

parisons.

7.2.1 Mixed ANOVA on Evaluation Trials

Mixed ANOVAs were performed on evaluation trial outcomes with either hit counts (for target-hitting) or deviation (for path-following) as the dependent variable, guidance condition ("group") as a between-subjects factor, and trial number ("trial") as the within-subjects factor (repeated measure). Plots of the data for evaluation trials alone are shown at subject-level in Figure 7.7 and Figure 7.8 and at group-level in Figure 7.9 and Figure 7.10. Outcomes for the omnibus ANOVA and for pairwise multiple comparisons, corrected using a Tukey-Kramer adjustment, are shown for target-hitting in Figure 7.11 and Table 7.3 and for path-following in Figure 7.12. A Ryan-Einot-Gabriel-Welch adjustment would preserve more power and be preferred, but was unavailable in the statistics software procedure used for analysis.

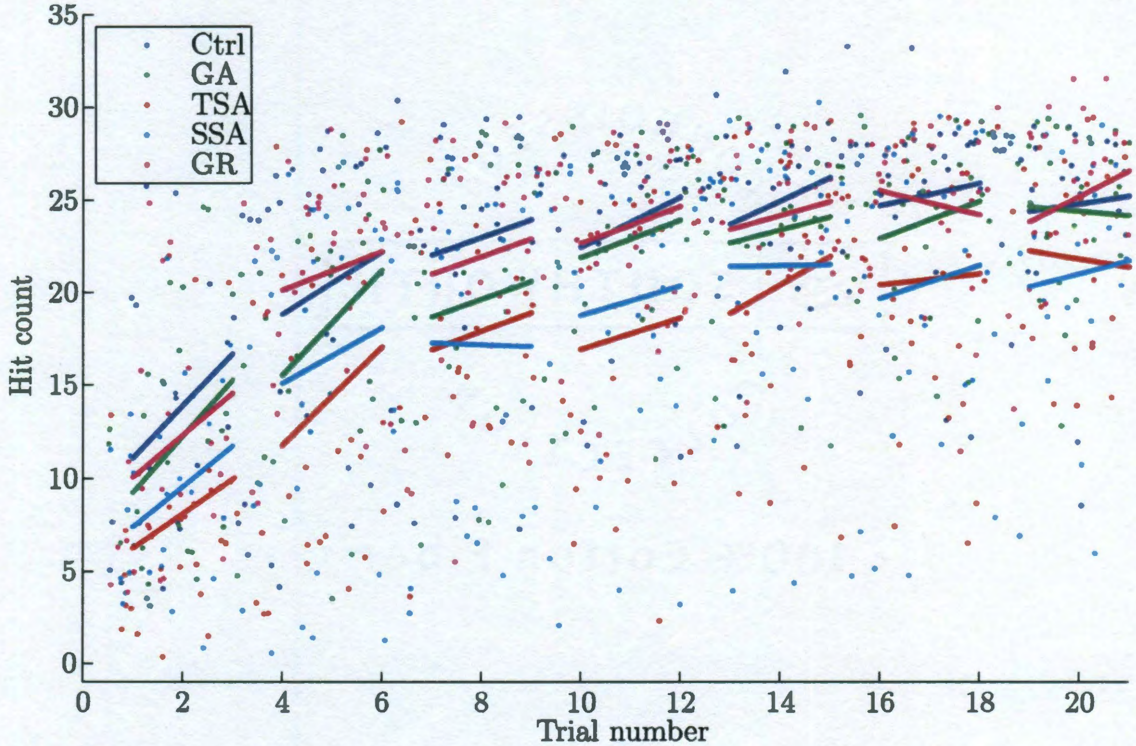

Figure 7.7: Target-hitting evaluation trials: Hit counts achieved by subjects versus trial number.

This procedure had the benefits of requiring minimal conditioning of the data before performing the statistical tests, being well-established, and having a higher statistical power. However, it also suffered from a few limitations. First, it only produces useful information on mean performance and does not give specific information on the rate or amount of performance improvement. Second, traditional ANOVA procedures assume normality, but unfortunately the data sets for both target-hitting and path-following were skewed due to ceiling and floor effects, respectively. Third, the statistical procedure and process of removing outliers is made complicated by the inclusion of a repeated-measures factor. Visual inspection indicates that the data is probably heteroscedastic and non-spherical (the variances are significantly different between groups and tend to diminish over time). Logarithmic and power (including

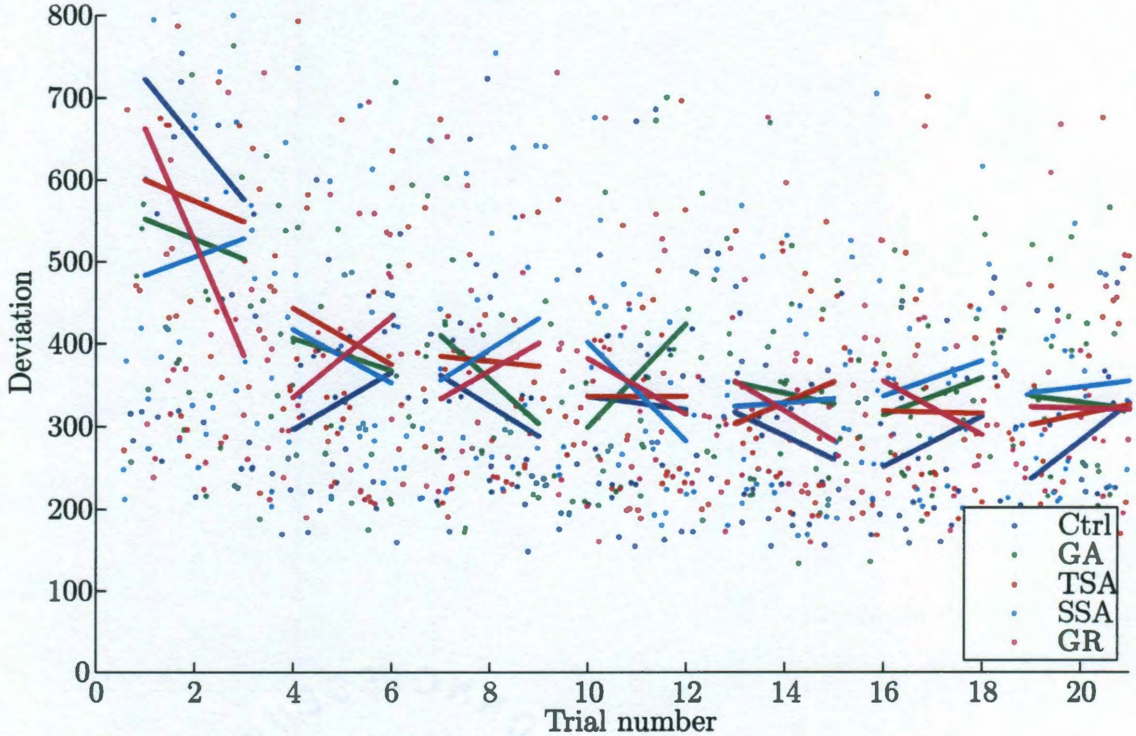

Figure 7.8: Path-following evaluation trials: Cumulative deviation (cm) of subjects versus trial number.

square root) transformations do not help to normalize the data or improve homogeneity of variance, and non-parametric analysis is not an option for mixed-measures designs. However, ANOVA is robust in the presence of distribution assumption violations when sample sizes are equal, meaning that the results will probably be too conservative (sacrificing power) rather than too liberal (producing Type I errors) [46].

7.2.2 Mixed ANOVA on Training Trials

Although the primary goal of this research is to improve robot-mediated training methodologies, the guidance conditions being tested could also be used for online correction of user inputs in the midst of task execution, such as when the autopilot

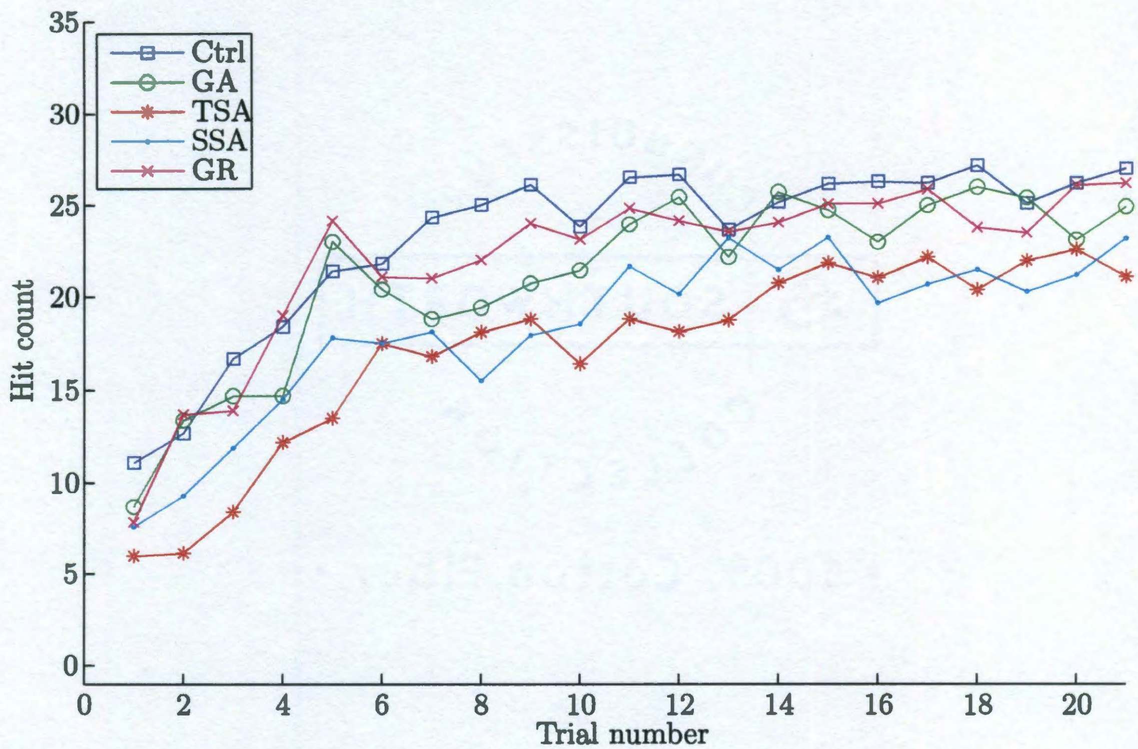

Figure 7.9: Target-hitting evaluation trials: Mean group hit counts versus trial number, outliers replaced.

or stick shaker mechanism in an aircraft shares control with the human pilot. Thus, it's useful to know how each guidance condition affects task performance while the guidance is actually active (during training), as shown in Figure 7.13 and Table 7.4.

7.2.3 Mixed ANOVA on Generalization Trials

Also of interest is how motor learning effects transfer to similar tasks with slightly modified task dynamics (generalization trials). Unfortunately, the omnibus ANOVA was not statistically significant. Visually inspecting Figure 7.5, it appears that performance may have varied drastically between the start and end of the generalization phase, so the same analysis was attempted on each half of the generalization phase

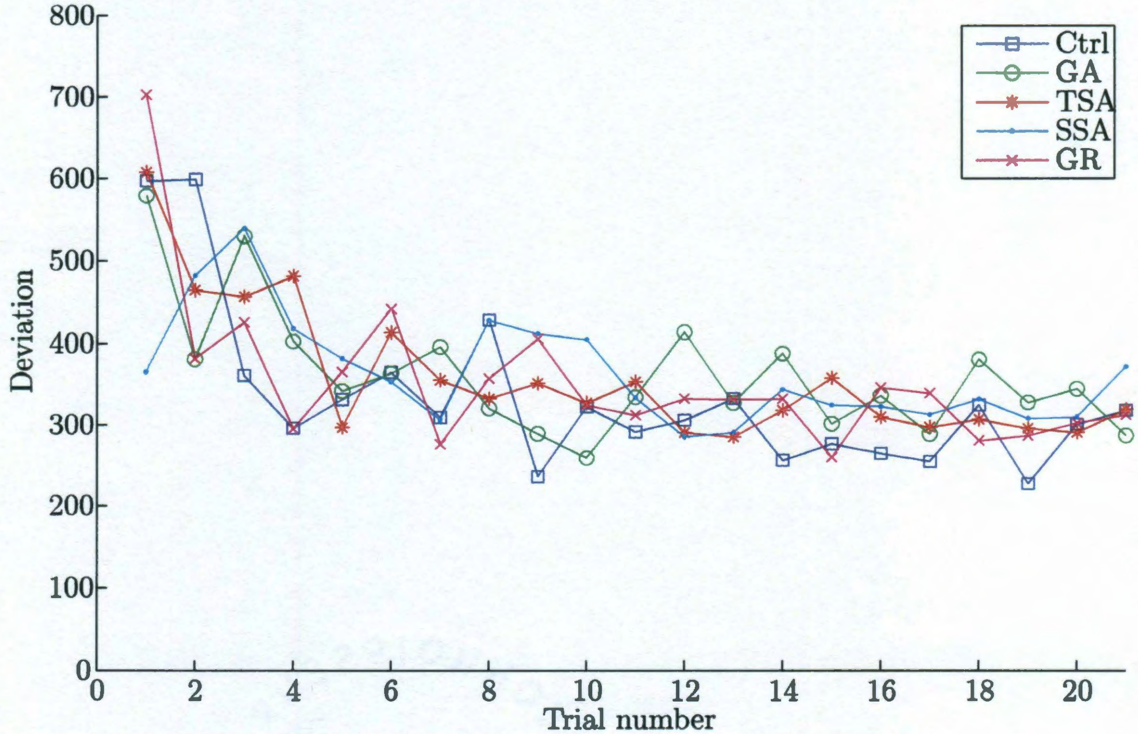

Figure 7.10: Path-following evaluation trials: Mean group deviation (cm) versus trial number, outliers replaced.

Group	Group	Diff.	p (adj)	Sig.
Ctrl	GA	−1.95	.282	
Ctrl	TSA	3.86	.035	*
Ctrl	SSA	3.76	.040	*
Ctrl	GR	3.99	.029	*
GA	TSA	5.81	.001	*
GA	SSA	5.71	.002	*
GA	GR	5.95	.001	*
TSA	SSA	−0.10	.956	
TSA	GR	0.13	.940	
SSA	GR	0.23	.897	

Table 7.4: Target-hitting task: Multiple pairwise comparisons for training trials. Adjusted p-values use the Tukey-Kramer adjustment.

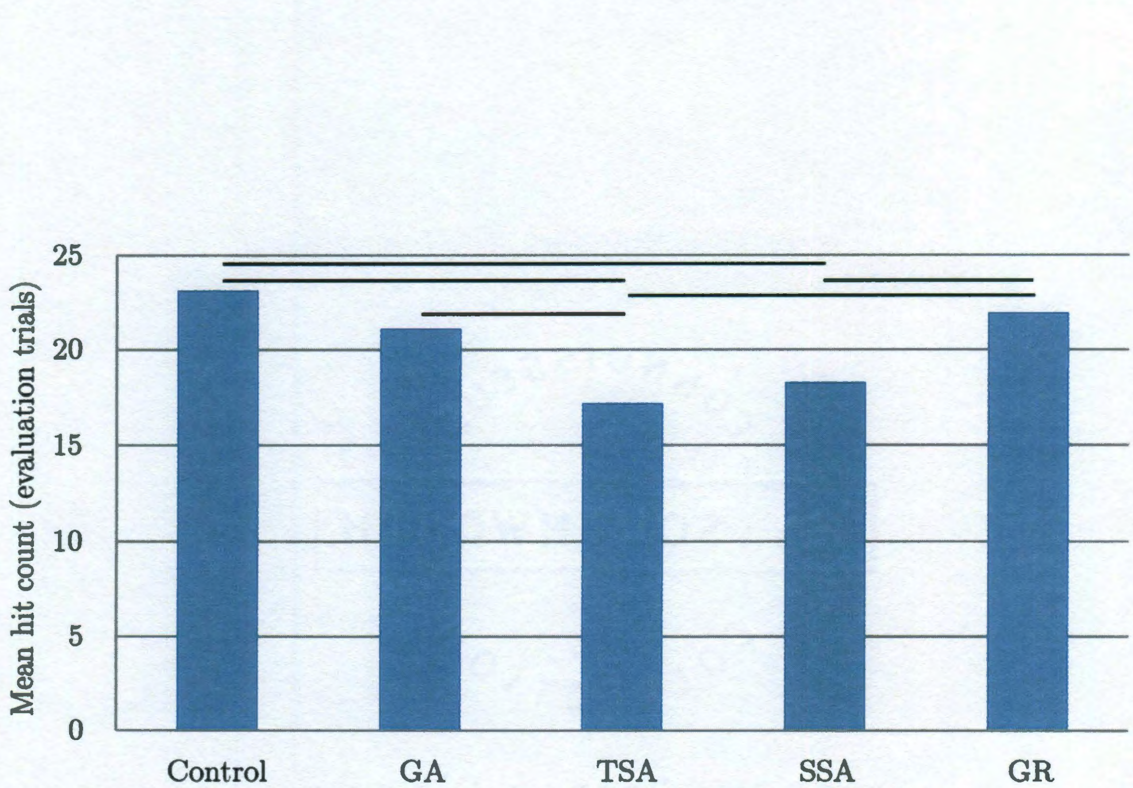

Figure 7.11: Target-hitting task: Mixed ANOVA results for evaluation trials. Fixed effect of group: $F(4, 113) = 9.46$, $p < .001$. The interaction effect of group and trial was not significant. Lines indicate pairwise significance at $\alpha = .05$, family-wise error-corrected using a Tukey-Kramer adjustment.

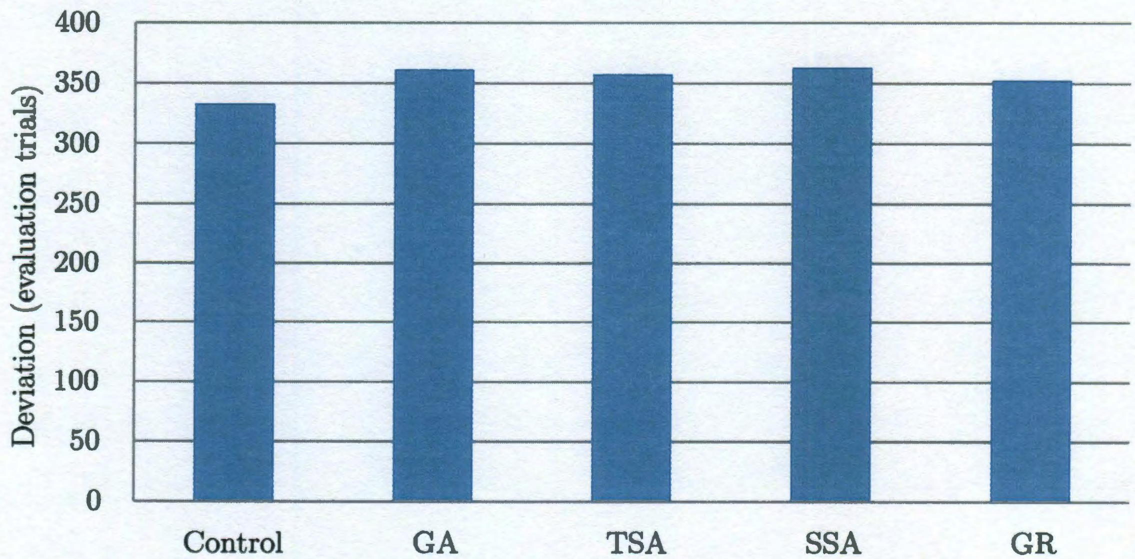

Figure 7.12: Path-following task: Mixed ANOVA results for evaluation trials. Fixed effect of group: $F(4, 258) = 1.45$, $p = .219$. Interaction effect of group and trial: $F(80, 869) = 1.68$, $p < .001$. No pairwise significance was found at $\alpha = .05$, family-wise error-corrected using a Tukey-Kramer adjustment.

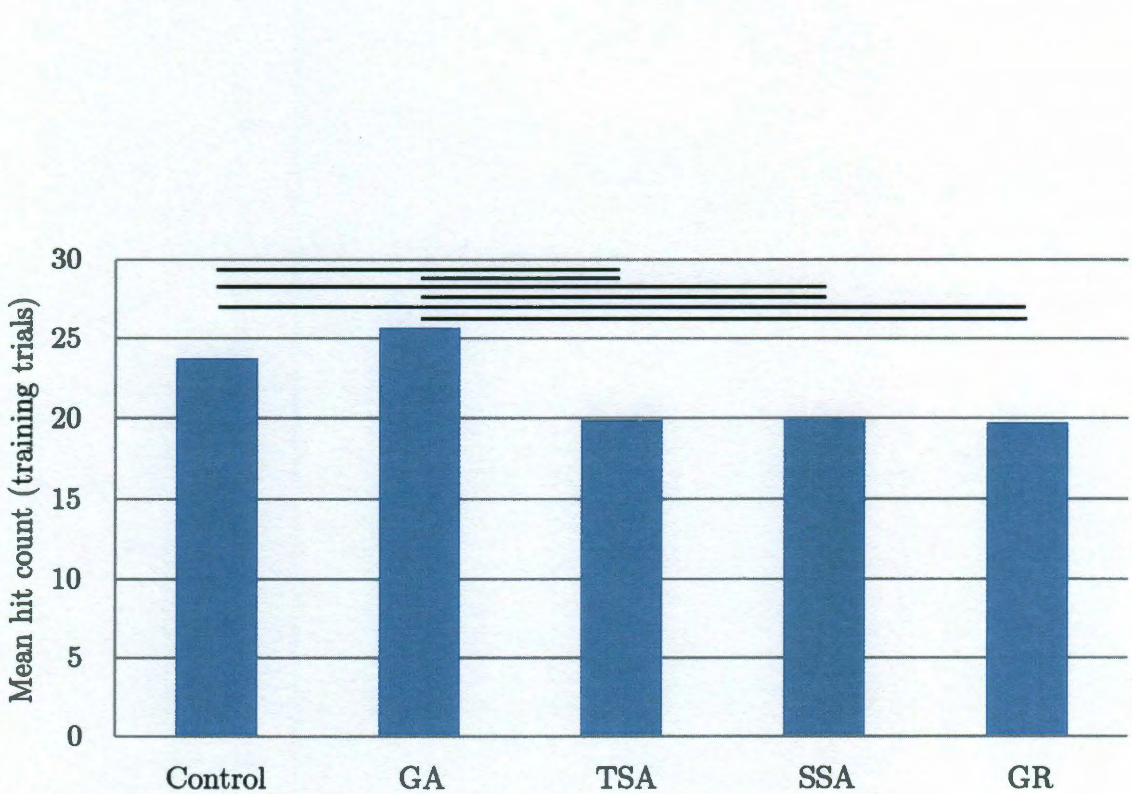

Figure 7.13: Target-hitting task: Mixed ANOVA results for training trials. Fixed effect of group: $F(4, 104) = 4.60$, $p = .002$. Interaction effect of group and trial: $F(284, 3061) = 2.31$, $p < .001$. Lines indicate pairwise significance at $\alpha = .05$, family-wise error-corrected using a Tukey-Kramer adjustment.

individually, but this too did not produce statistically significant results.

7.2.4 Curve-Fitting on Evaluation, Training, and Generalization Trials

An alternative means of analysis that has been successful in prior studies (especially the pilot study) is to fit an exponential curve to each subject's trial performance outcomes, as described by Heathcote et al. [47]. This produces three meaningful and independent coefficients for each subject that can be used as dependent variables in a one-way ANOVA between guidance groups. The main advantages of curve-fitting over mixed ANOVA analysis are that curve-fitting provides specific information on the rate and amount of training and allows for the use of simpler and more robust statistical tests. Unfortunately, it appears that in the primary study curve-fitting analysis had lower statistical power than mixed ANOVA analysis.

An exponential curve of the form $y = \pm ae^{-\frac{x}{b}} + c$ was fit to the trial outcomes for each subject, as shown in Figure 7.14, where the parameters have the following meanings:

a **Amount of learning** Difference between the final and initial performance levels, where larger values indicate more learning

b **Rate of learning** A time constant, where smaller values indicate faster learning

c **Final performance level** Larger values indicate the subject achieved a higher performance level at the end of the study

x **Trial number**

y **Performance measure** Either hit counts (for target-hitting) or deviation (for path-following)

Each parameter (a, b, and c) was considered as a dependent variable in a one-way ANOVA between groups of subjects. Unfortunately, a Kolmogorov-Smirnov test for normality rejected the null hypothesis that the parameters were drawn from a normal distribution for almost every combination of parameter and group. This non-normality is obvious from visual inspection, as in Figure 7.15. Additionally, the data is clearly heteroscedastic (the variances are heterogeneous between groups), as shown in Figure 7.16. This precludes the use of a traditional ANOVA- instead, the Kruskal-Wallis non-parametric ANOVA was used to test curve-fit parameters. However, the omnibus ANOVA was not significant for any curve-fit parameter ($F(4, 45) < 1$).

Permutation testing, another form of non-parametric testing, was also used to compare the curve-fit parameters. The only significant pairwise comparisons were found

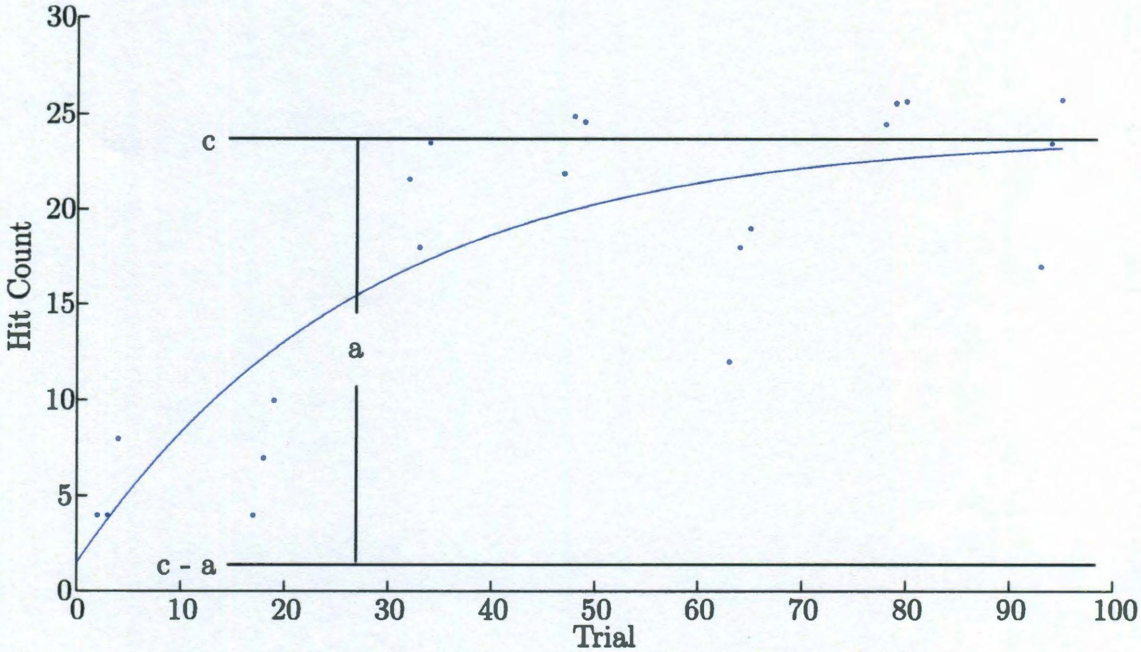

Figure 7.14: Curve-fitting example for a subject's target-hitting session. The solid line is the best fit of the hit counts (y) from each trial (x) to the equation $y = -ae^{-\frac{x}{b}} + c$. Each coefficient ("parameter") in the fit equation has a physical meaning: a is the amount of learning, b is the speed of learning, and c is the final performance value.

Group 1	Group 2	Group 1 Mean	Group 2 Mean	p
TSA	SSA	31.7	16.4	.049
TSA	GR	31.7	9.7	.008

Table 7.5: Target-hitting task: Permutation testing results.

for the b (speed of learning) parameter in the target-hitting task, as shown in Table 7.5. These results indicate that the SSA and GR groups each improved significantly faster than the TSA group.

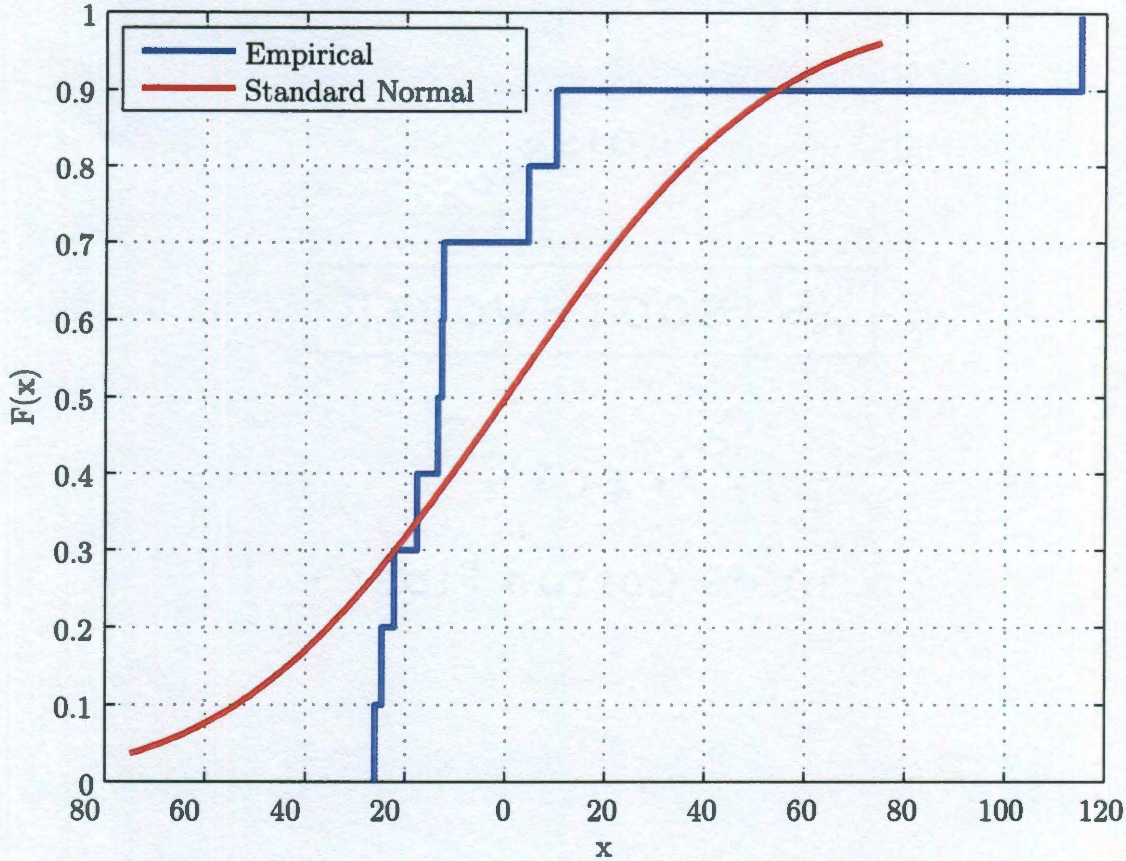

Figure 7.15: Cumulative distribution plot of actual vs ideal (normally distributed) data for a representative curve-fit parameter.

7.2.5 Workloads

As mentioned in subsection 7.1.1, 7 measures of workload were recorded for each participant's session. These ratings tended to have homogeneous variances but non-normal distributions, as shown in Figure 7.17. Minor violations of the normality assumptions are sometimes acceptable if the data is skewed consistently in one direction (as is the case with hit counts or deviation), but these are rather severe violations, and the skew is often in opposite directions from one group to another. Thus, the Kruskal-Wallis non-parametric ANOVA is warranted. The only statistically significant omnibus test was for Mental Demand ($F(4, 41) = 10.38$, $p = .035$), but no

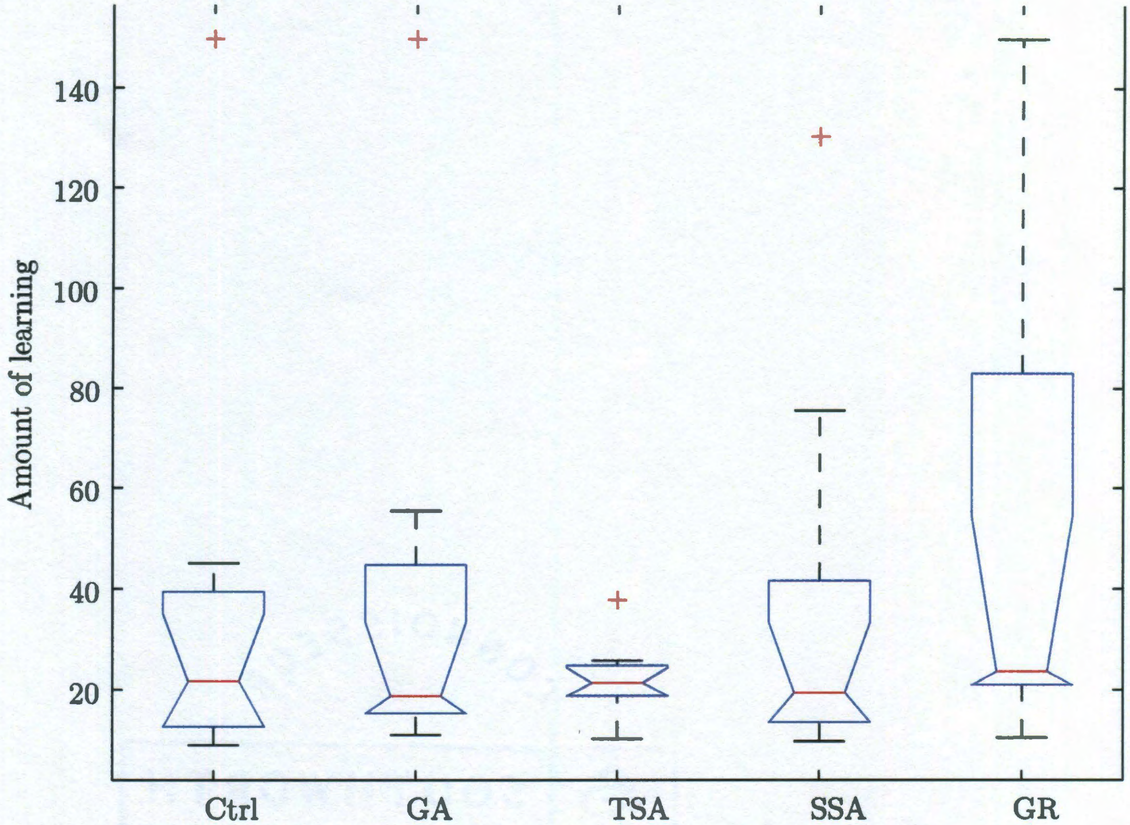

Figure 7.16: Box plot of the curve-fit parameter a for each group. Note the non-normal distribution of values of a for each group and the largely unequal variances between groups, as well as the relatively similar means between groups.

pairwise comparisons survived a Tukey-Kramer adjustment.

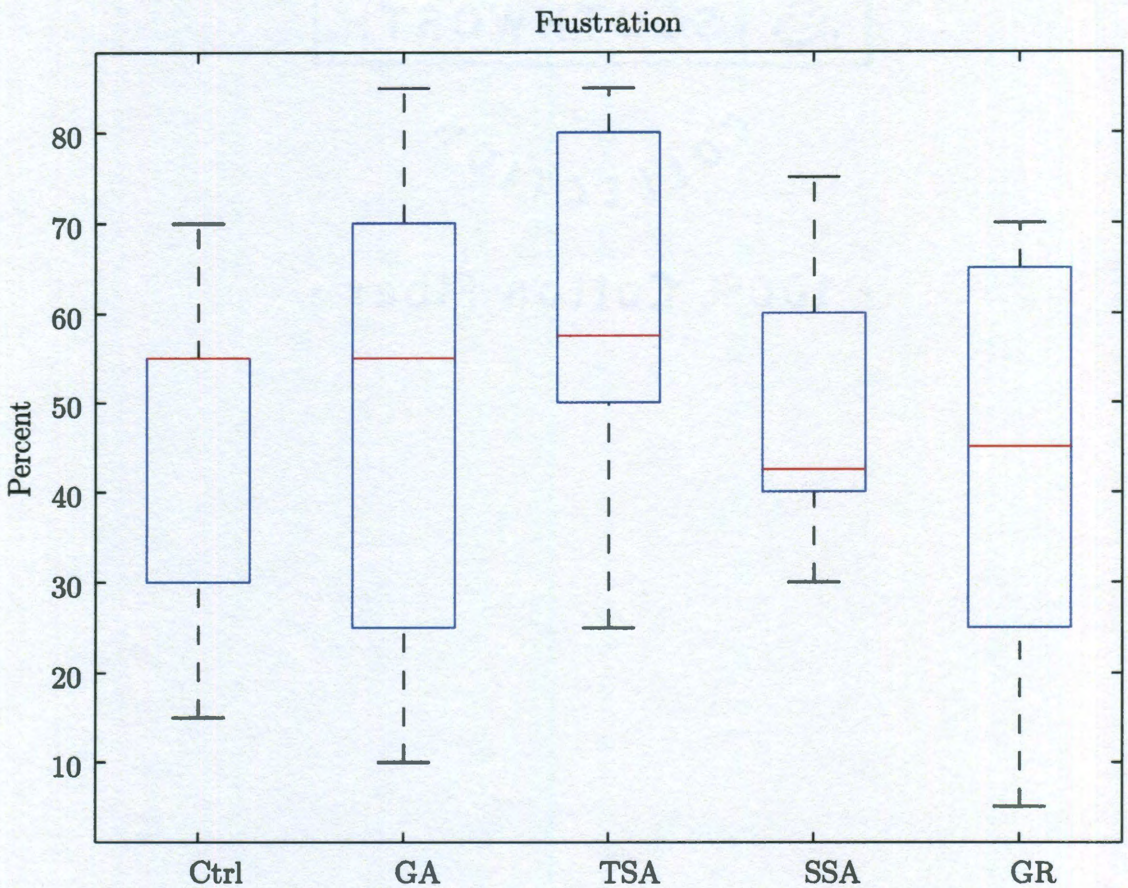

Figure 7.17: Box plot of frustration self-ratings. While the variances are fairly homogeneous, the ratings are clearly not normally distributed.

Chapter 8

Discussion

The results in Section 7.2 corroborate a strong form of the guidance hypothesis: namely, that any attempts at guidance (even resistive forms) can impair training as compared to practice. Previous studies have shown that subjects can become dependent on assistive guidance, and the guidance hypothesis theorizes that challenge is necessary to the learning process. The results of this study support that notion, but also suggest that even additional challenge can impair learning, and indeed that any interference with the task (through attempts at guidance) will impair training compared to straight practice. This is supported by the fact that in the target-hitting task, both of the novel guidance separation paradigms (TSA, SSA), which were designed specifically to discourage dependency, led to significantly worse performance than the control group. Additionally, the results suggest that the gross guidance paradigms (GA, GR) also led to worse performance than the control group. In the path-following task, while none of the results were statistically significant, the control group still appeared to perform better than the guidance groups.

It is especially interesting that the separation paradigms (TSA, SSA) actually led to worse performance than either of the gross guidance paradigms (GA, GR). There are several factors that might explain this. Generally speaking, it is possible that subjects were simply unfamiliar with these novel forms of guidance, and did not fully understand how they were supposed to use the guidance. Anecdotally, subjects had a relatively easy time understanding the operating principles of GA and GR, while they had a comparatively harder time understanding how to use TSA and SSA. Thus, it is possible that more thorough training for how to use these somewhat complex paradigms would lead to improved results. Additionally, subjects were not informed of the theory behind the separation paradigms in order to obviate any placebo effects resulting from the power of suggestion or experimenter bias. However, it is possible that if participants had understood why such complex guidance methodologies were being used, then they would be less frustrated with the guidance and better understand how to fully utilize it.

Specific to TSA, many subjects reported that they found the constant "nudging" to be frustrating, although higher frustration in the TSA group was not reflected in the workload results. Additionally, it is possible that the poor performance of TSA in the target-hitting task was due to the rhythmic nature of the task. While there is an optimal excitation frequency and a clearly defined optimal path that minimizes trajectory error, the initial conditions of the task will produce optimal trajectories that are out of phase with each other in time. In other words, while it is true that following the expert precisely would elicit the highest hit count in the task, following the expert is not a necessary condition for achieving the highest hit count. It is entirely possible to follow the expert at a phase lag and still achieve the maximum hit count- in fact, in the GA condition, guidance forces and task forces are actually equal and opposite when the novice is out of phase with the expert by a certain amount. By contrast, the TSA group reported that they found the assistance from the virtual expert to be pervasive and annoying even during later sessions, confirming that participants were likely performing the task out of phase with the expert. The fact the the TSA group did not perform significantly differently from other groups in the path-following task corroborates the potentially detrimental effects of rhythmic tasks on the effectiveness of TSA.

These results suggest that perhaps the best way to enhance training is to increase the difficulty of a task without altering the inherent task dynamics or interfering with task execution through explicit guidance. For instance, decreasing the target size in the target-hitting task or augmenting the perceived error in the path-following task might both be effective ways of enhancing training.

Considering just the effect of the guidance conditions on performance during training, it is worthwhile to note that the control and GA groups outperformed most other groups, and results suggest that the GA group outperformed the control group. This suggests that if guidance is being used to assist an operator in the real-time execution of tasks, for instance to prevent the operator from entering dangerous or forbidden regions of the workspace, then GA is the guidance method of choice. Further study is warranted on how to effectively guide users through the real-time completion of a task with minimal interference.

It is unfortunate and unexpected that the curve-fitting analysis did not produce significant results, as it offers the most detailed information on the effects of the guidance paradigms and was significant in a pilot study with fewer subjects. There are two factors that likely contributed to the low power. First, the groups were not well balanced in terms of subject aptitude and initial perfomance level. Some subjects seemed to be "natural experts", performing well in the initial evaluation and improving little over the course of the study, while others performed very poorly at entrance. Additionally, some subjects exhibited a consistent learning trend with decreasing variance in performance over time, while others had wildly varying performance over the course of the study, with only a weak upward trend (as demonstrated by the number of subjects achieving zero hits even in the final stages of target-hitting training in Figure 7.5). The distribution of these subjects across groups was not even, and this limited the ability to account for these factors during data analysis. Secondly, the tasks may not have been challenging enough to elicit significant changes in performance over time for all subjects and guidance conditions. Especially for the path-following task, subjects tended to quickly reach the performance ceiling (saturation level of performance). Because the effective training period was so short, there was relatively little data to draw conclusions from, and the effects of training may have been overshadowed by the effects of subjects reaching the performance ceiling.

Generally speaking, it is important to keep in mind there was quite a large variability in performance between subjects, and guidance may not affect all subjects equally. For instance, it is possible that complete novices who find a task extraordinarily difficult will still benefit from certain types of guidance, as mentioned in Section 4.1. Further study on different subject populations is warranted.

Finally, it is also unfortunate and unexpected that the workload analysis did not produce significant results, as it did in pilot study. Anecdotally, many subjects reported that the NASA TLX workload scale was simply not intuitive or easy to use. Specifically, they did not understand the pairwise comparisons between subscales used to produce an overall weighted workload score, and did not have a reference on which to base their reported workload on each subscale. From a statistical point of view, this meant that the between-subject variance for each subscale was quite high. In the future, better training on how to use the TLX scale or the addition of qualitative reference values (i.e., "could perform the task in my sleep" through "task is physically impossible") might improve power. Other workload scales should also be considered, such as the Workload Profile scale developed by Tsang and Velazquez [48]. Rubio et al. [49] showed that the Workload Profile scale has a higher sensitivity and diagnostic power than NASA TLX.

Chapter 9

Conclusions

The results of these two studies involving 66 subjects and two disparate dynamic tasks have shown that many of our instincts about haptic guidance are wrong: conventional approaches to haptic robot-mediated training are not significantly better than practice, and more complex guidance paradigms can in fact be detrimental to the learning process. Further study is warranted to determine the generalizability of these results to training for other types of tasks, to robot-mediated rehabilitation, and to real-time mediation of task execution.

To facilitate continued research, this work has made a number of additional contributions. A guidance paradigm taxonomy has been proposed that will allow for easier discussion, classification, and comparison of haptic guidance paradigms. A new software platform for studying training of dynamic tasks has been developed with a specific focus on portability and experimental integrity. Finally, the traditional shared-control proxy model has been improved in order to accommodate a number of more complex guidance paradigms. Although these paradigms were not shown to be superior in the context of training, it is hypothesized that they would be very advantageous in the context of robot-mediated real-time task execution, and could improve shared-control human-machine interfaces ranging from autopilots to robotic surgical systems.

Bibliography

[1] R. B Gillespie, M. O'Modhrain, P. Tang, D. Zaretzky, and C. Pham. The virtual teacher. In Proceedings of the ASME Dynamic Systems and Control Division, volume 64, pages 171—178, 1998.

[2] Saso Jezernik, Gery Colombo, Thierry Keller, Hansruedi Frueh, and Manfred Morari. Robotic orthosis lokomat: A rehabilitation and research tool. Neuromodulation, 6(2):108–115, 2003. ISSN 1094-7159. doi: 10.1046/j.1525-1403.2003.03017.x. URL `http://onlinelibrary.wiley.com/doi/10.1046/j.1525-1403.2003.03017.x/abstract`.

[3] C. Ho, C. Basdogan, M. Slater, N. Durlach, and M. A. Srinivasan. An experiment on the influence of haptic communication on the sense of being together. In BT Presence Workshop, 1998.

[4] I. Oakley, S. Brewster, and P. Gray. Can you feel the force? an investigation of haptic collaboration in shared editors. In proceedings of EuroHaptics, page 5459, 2001.

[5] K.B. Reed and M.A. Peshkin. Physical collaboration of Human-Human and Human-Robot teams. IEEE Transactions on Haptics, 1(2):108–120, 2008. ISSN 1939-1412. doi: 10.1109/TOH.2008.13. URL `10.1109/TOH.2008.13`.

[6] L. B Rosenberg. The use of virtual fixtures as perceptual overlays to enhance operator performance in remote environments. Technical report, Stanford University Center for Design Research, 1992.

[7] R.J. Adams and B. Hannaford. Stable haptic interaction with virtual environ-

ments. IEEE Transactions on Robotics and Automation, 15(3):465–474, 1999. ISSN 1042296X. doi: 10.1109/70.768179.

[8] R. B Gillespie and M. R Cutkosky. Stable user-specific haptic rendering of the virtual wall. In Proceedings of the ASME International Mechanical Engineering Congress and Exhibition, volume 58, page 397406, 1996.

[9] D. Ruspini and O. Khatib. Dynamic models for haptic rendering systems. Advances in Robot Kinematics: ARK98, page 523532, 1998.

[10] I Rock and J Victor. Vision and touch: An experimentally created conflict between the two senses. Science (New York, N.Y.), 143:594–596, February 1964. ISSN 0036-8075. URL `http://www.ncbi.nlm.nih.gov/pubmed/14080333`. PMID: 14080333.

[11] K. J Kuchenbecker, J. Fiene, et al. Improving contact realism through event-based haptic feedback. IEEE Transactions on Visualization and Computer Graphics, page 219230, 2006.

[12] K. Salisbury, F. Conti, and F. Barbagli. Haptic rendering: Introductory concepts. IEEE Computer Graphics and Applications, 24(2):2432, 2004.

[13] Ozkan Celik. Comparison of robotic and clinical motor function improvement measures for sub-acute stroke patients, 2008.

[14] C. J Winstein. Knowledge of results and motor learningimplications for physical therapy. Physical Therapy, 71(2):140, 1991.

[15] Yanfang Li, Joel Huegel, Volkan Patoglu, and Marcia O'Malley. Progressive shared control for training in virtual environments, 2009.

[16] Joel Huegel and Marcia O'Malley. Visual versus haptic progressive guidance for training in a virtual environment, 2009.

[17] L.B. Rosenberg. Virtual fixtures: Perceptual tools for telerobotic manipulation. In Virtual Reality Annual International Symposium, 1993., 1993 IEEE, pages 76–82, 1993. doi: 10.1109/VRAIS.1993.380795.

[18] D. J. Reinkensmeyer. How to retrain movement after neurologic injury: a computational rationale for incorporating robot (or therapist) assistance. In Proceedings of the 2003 IEEE Engineering in Medicine and Biology Society Meeting, volume 2, page 14791482, 2003.

[19] S. J Harkema. Neural plasticity after human spinal cord injury: application of locomotor training to the rehabilitation of walking. The Neuroscientist, 7(5): 455, 2001.

[20] C. J. Winstein, P. S. Pohl, and R. Lewthwaite. Effects of physical guidance and knowledge of results on motor learning: support for the guidance hypothesis. Res Q Exerc Sport, 65(4):316323, 1994.

[21] J. D Hagman. Presentation-and test-trial effects on acquisition and retention of distance and location. Journal of experimental psychology: Learning, memory, and cognition, 9(2):334345, 1983.

[22] T. R. Armstrong. Training for the production of memorized movement patterns. In Proceedings of the 6th Annual Conference on Manual Control. Air Force Flight Dynamics Laboratory, WPAFB, Ohio, 1970.

[23] R. A Schmidt and R. A Bjork. New conceptualizations of practice: Common principles in three paradigms suggest new concepts for training. Psychological Science, 3(4):207, 1992.

[24] J. F Israel, D. D Campbell, J. H Kahn, and T. G Hornby. Metabolic costs and muscle activity patterns during robotic-and therapist-assisted treadmill walking in individuals with incomplete spinal cord injury. Physical Therapy, 86(11):1466, 2006.

[25] R. Shadmehr and F.A. Mussa-Ivaldi. Adaptive representation of dynamics during learning of a motor task. Journal of Neuroscience, 14(5 II):3208–3224, 1994. URL `http://www.scopus.com/inward/record.url?eid=2-s2.0-0028270380&partnerID=40&md5=1ecc0e303428bfd02c2e54a8e52752f7`.

[26] L. M Crespo and D. J Reinkensmeyer. Effect of robotic guidance on motor learning of a timing task. In 2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics, 2008. BioRob 2008, page 199204, 2008.

[27] L. G. Summers, J. H. Shannon, T. R. White, and R. J. Shiner. Fly-by-wire sidestick controller evaluation. SAE Technical Paper Series, 1987.

[28] S.S. Nudehi, R. Mukherjee, and M. Ghodoussi. A shared-control approach to haptic interface design for minimally invasive telesurgical training. IEEE Transactions on Control Systems Technology, 13(4):588–592, 2005. ISSN 1063-6536. doi: 10.1109/TCST.2004.843131.

[29] Marcia K. O'Malley, Abhishek Gupta, Matthew Gen, and Yanfang Li. Shared control in haptic systems for performance enhancement and training. Journal of Dynamic Systems, Measurement, and Control, 128(1):75–85, March 2006. doi: 10.1115/1.2168160. URL `http://link.aip.org/link/?JDS/128/75/1`.

[30] L. M Crespo and D. J Reinkensmeyer. Haptic guidance can enhance motor learning of a steering task. Journal of Motor Behavior, 40(6):545557, 2008.

[31] Marcia O'Malley, Alan Sledd, Abhishek Gupta, Volkan Patoglu, Joel Huegel, and Charles Burgar. The RiceWrist: a distal upper extremity rehabilitation robot for stroke therapy, 2006.

[32] Leonard E Kahn, Michele L Zygman, W Zev Rymer, and David J Reinkensmeyer. Robot-assisted reaching exercise promotes arm movement recovery in chronic hemiparetic stroke: a randomized controlled pilot study. Journal of NeuroEngineering and Rehabilitation, 3:12–12, 2006. doi: 10.1186/1743-0003-3-12. PMID: 16790067 PMCID: 1550245.

[33] Alan W. Salmoni, Richard A. Schmidt, and Charles B. Walter. Knowledge of results and motor learning: A review and critical reappraisal. Psychological Bulletin, 95(3):355–386, 1984. ISSN 0033-2909. doi: 10.1037/0033-2909.95.3.355. URL `http://psycnet.apa.org.ezproxy.rice.edu/journals/bul/95/3/355/`.

[34] Takahiro Endo, Haruhisa Kawasaki, Kazushige Kigaku, and Tetsuya Mouri. Transfer method of force information using Five-Fingered haptic interface robot. In Proceedings of the Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, pages 599–600. IEEE Computer Society, 2007. ISBN 0-7695-2738-8. URL `http://portal.acm.org/citation.cfm?id=1264885`.

[35] G Wulf, C H Shea, and C A Whitacre. Physical-guidance benefits in learning a complex motor skill. Journal of Motor Behavior, 30(4):367–380, December 1998. ISSN 0022-2895. doi: 10.1080/00222899809601351. URL `http://www.ncbi.nlm.nih.gov/pubmed/20037040`. PMID: 20037040.

[36] O. Lambercy, L. Dovat, R. Gassert, E. Burdet, C. L Teo, and T. Milner. A

haptic knob for rehabilitation of hand function. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 15(3):356366, 2007.

[37] Susan L Morris, Karen J Dodd, and Meg E Morris. Outcomes of progressive resistance strength training following stroke: a systematic review. Clinical Rehabilitation, 18(1):27 –39, January 2004. doi: 10.1191/0269215504cr699oa. URL `http://cre.sagepub.com/content/18/1/27.abstract`.

[38] Jeremy L Emken and David J Reinkensmeyer. Robot-enhanced motor learning: accelerating internal model formation during locomotion by transient dynamic amplification. IEEE Transactions on Neural Systems and Rehabilitation Engineering: A Publication of the IEEE Engineering in Medicine and Biology Society, 13(1):33–39, March 2005. ISSN 1534-4320. doi: 10.1109/TNSRE.2004.843173. URL `http://www.ncbi.nlm.nih.gov/pubmed/15813404`. PMID: 15813404.

[39] J. L. Patton, M. Kovic, and F. A. Mussa-Ivaldi. Custom-designed haptic training for restoring reaching ability to individuals with stroke. Journal of Rehabilitation Research and Development, 43:643656, 2006.

[40] J. Lee and S. Choi. Effects of haptic guidance and disturbance on motor learning: Potential advantage of haptic disturbance. 2010 IEEE Haptics Symposium, 2010.

[41] K. J Kuchenbecker, J. G Park, and G. Niemeyer. Characterizing the human wrist for improved haptic interaction. In Proc. ASME Int. Mechanical Engineering Congress and Exposition, volume 2, page 42017, 2003.

[42] C. B. Zilles and J. K. Salisbury. A constraint-based god-object method for haptic display. In 1995 IEEE/RSJ International Conference on Intelligent Robots and

Systems 95.'Human Robot Interaction and Cooperative Robots', Proceedings, page 146151, 1995.

[43] S. G Hart. Development of NASA-TLX (Task load index): Results of empirical and theoretical research. Technical report, NASA Ames Research Center, 1988.

[44] Yanfang Li, Volkan Patoglu, and Marcia K. O'Malley. Negative efficacy of fixed gain error reducing shared control for training in virtual environments. ACM Trans. Appl. Percept., 6(1):1–21, 2009. doi: 10.1145/1462055.1462058. URL `http://portal.acm.org/citation.cfm?id=1462058`.

[45] M.K. O'Malley and A. Gupta. Skill transfer in a simulated underactuated dynamic task. In Proceedings of the 2003 IEEE International Workshop on Robot and Human Interactive Communication, pages 315–320, 2003.

[46] Brad Thiessen. ANOVA assumptions, 2007. URL `http://homepage.mac.com/bradthiessen/BAT/M301.html`.

[47] A. Heathcote, S. Brown, and D. J. K. Mewhort. The power law repealed: The case for an exponential law of practice. Psychonomic Bulletin and Review, 7(2): 185207, 2000.

[48] P. S. Tsang and V. L. Velazquez. Diagnosticity and multidimensional subjective workload ratings. Ergonomics, 39(3):358, 1996.

[49] Susana Rubio, Eva Diaz, Jesus Martin, and Jose M. Puente. Evaluation of subjective mental workload: A comparison of SWAT, NASA-TLX, and workload profile methods. Applied Psychology, 53(1):61–86, 2004. ISSN 0269-994X. doi: 10.1111/j.1464-0597.2004.00161.x. URL `http://onlinelibrary.wiley.com/doi/10.1111/j.1464-0597.2004.00161.x/full`.