NORTHWESTERN UNIVERSITY

Shared Control for Robot-Aided Task Training and Robotic Assistance

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

MASTER OF SCIENCE

Field of Mechanical Engineering

By

Aleksandra Kalinowska

EVANSTON, ILLINOIS

December 2018

ABSTRACT

Shared Control for Robot-Aided Task Training and Robotic Assistance

Aleksandra Kalinowska

Robots have the potential to assist with both learning and executing difficult tasks, particularly in populations with motor deficit. However, adequate shared control schemes are needed to effectively allocate control between the human and the user to either facilitate training or provide assistance without unnecessarily limiting the user's intended actions. Here, we propose a shared control paradigm based on filtering that can be used for both training and assistance in human-robot interfaces with known tasks. We use a novel criterion for evaluating user input—the mode insertion gradient (MIG), commonly used in hybrid control—that allows us to instantaneously assess the impact of user actions on the evolution of a dynamic system over a time window into the future. As a result, the filter is permissive to many chosen strategies and minimally interfering—qualities desired when evaluating human actions.

We explore a version of the shared control paradigm for assistance. We show in simulation that it is able to assist in task completion and enforce safety even for unskilled

users simulated with Gaussian noise input. At the same time we show that, when possible, the interface remains transparent to users, particulary skilled users simulated using near-optimal model predictive controllers. Finally, given a stable controller for providing assistance, we show that this filter-based shared control scheme can to first order guarantee system safety with respect to a pre-defined objective. Future work will involve implementing the shared control paradigm for real-time assistance on hardware devices, such as a lower-limb exoskeleton.

In parallel, we explore the shared control paradigm for task training. Through a human subject study, we show significant improvement in task learning using the proposed scheme. For the task of inverting a simulated cart pendulum, users continually improve when training with this filter-based feedback, while controls plateau shortly into the training and overall reach a lower skill level. Interestingly, we observe the shared control paradigm to exhibit multiple features that have, in previous studies, been shown to be conducive to learning. For one, it only rejects user actions, allowing failure at the task and avoiding user passivity. Secondly, it improves performance when actively engaged. And thirdly, it is sensitive to user skill level and real-time performance, behaving similarily to traditional assist-as-needed paradigms. Further experiments will involve studies of impaired individuals in clinically relevant tasks as well as direct comparisons with other assist-as-needed shared control schemes.

Overall there is significant interest in enabling difficult tasks or making them safer through online robotic assistance as well as in facilitating task training through forceful robotic interaction—the proposed filter-based shared control paradigm with MIG as an evaluation criterion can facilitate these human-robot collaborations.

Acknowledgements

acknowledgements here

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

Shared control algorithms have been developed for robotic assistance and robot-supported training in applications ranging from assisted vehicle navigation [**2**] and surgical robotics [**32, 39**] to brain-computer interface manipulation [**31**] and exoskeleton-assisted gait [**22, 46**]. The aims and safety requirements of these systems vary greatly, but one challenge is often the same—how do we most effectively allocate control between the user and the machine?

User trust in the robot is a critical factor in the overall performance of the joint human-machine system [**18**]. Trust, in this context, mainly depends on robot performance and its attributes, such as dependability, predictability, and level of automation [**21**]. Thus, to encourage user trust, shared control algorithms should avoid robot behavior that is difficult for the human to understand [**23**], unpredictable, or unnecessary. In task-based assistance, avoiding such behavior can be challenging, because there are often many ways of accomplishing a task and the individual is likely to take an approach that is different from the controllers calculated strategy. However, the issue can be addressed by developing shared control paradigms that instantenously adjust to operator performance and real-time adapt to varying user strategies, such as in [**48**] and in the work proposed here.

Another factor to consider is user preference. In [**38**], machine learning techniques were used to model user preferences for autonomous systems; in other approaches, users were

able to manually choose their level of assistance [**20**]. In general, studies show that users of assistive devices prefer to maintain as much control authority as possible [**5, 51, 27**]. Overconstraining user inputs may prevent them from utilizing valid control strategies and hence leads to frustration and device abandonment [**6**]. That said, users may be willing to accept some loss of control authority, but only if the improvements in performance are significant [**51, 27**]. Therefore, devices are most likely to be used if they make tasks significantly easier without limiting users' actions [**6**].

Finally, in robotic training, providing too much assistance or overconstraining users undermines the therapeutic impact of the device. Engaging the user has been shown to be critical to robotic training [**29**]. Likewise, allowing mistakes or errors and overall task failure has been shown to be critical to learning [**43**]. In some cases, successful interfaces have adopted strategies of amplifying error [**14, 34**]. On the flip side, interfaces that provide too much support oftentimes lead to user passivity, resulting in low skill retention and little transferable skills [**40, 49**].

Many shared control schemes adapt their level of support online [**15, 37, 50, 13**] using an algorithm or schedule that prescribes changes based on some notion of the user's need for assistance. These levels of support can be modulated based on performance measures such as error [**16, 29, 36, 35**], movement speed [**24**], and task adeptness [**26**], or based on the user's strength and fitness level [**30, 22**] or current cognitive engagement in the task [**8**]. At other times, the level of support can be manually adjusted by a physical therapist or the users themselves [**20**]. These adaptive shared control schemes are often referred to as assist-as-needed and are characterized by dynamically updating the relative contributions of the robot and human.

Here, we present a filter-based assist-as-needed shared control paradigm. While other assist-as-needed schemes adjust the level of provided support from trial-to-trial, we implement a filter that acts based on real-time assessment of user intent. It requires no predefined trajectory, runs on an indefinite time horizon, and automatically adapts to operator skill. *A key contribution of this work is the criterion used for evaluating and selecting admissible user input.* As our criterion, we use the Mode Insertion Gradient (MIG)—a concept from hybrid control that allows us to assess how users' inputs affect the human-machine system over a time window into the future.

The result is a mechanical filter that remains transparent when users are skilled and high performing, allowing inputs that are safe and/or do not hinder achieving a task goal, but interferes when users are underperforming, disallowing inputs that are unsafe or incorrect with respect to the task objective. It is worth noting that the filter requires user input and does not drive the system when the user is not actively engaged in the task, encouraging user engagement. It also does not require pre-defining a desired trajectory, allowing subjects to explore various strategies for completing a task.

In chapters to follow, we present experimental results on the performance and training effects of this real-time assist-as-needed shared control scheme. We evaluate the MIG-based filter in two modes: assistance and training. In the assistance mode, we replace "incorrect" user inputs with a controller-calculated action to impose safety and/or task completion. In the training mode, we reject "incorrect" user actions to allow failure and facilitate learning. Experimental and theoretical analyses of the shared control scheme in both modes are presented in Chapters 3 and 4, respectively. We demonstrate the assist-as-needed feature of the filter by evaluating correlations between initial user skill level

and current user performance on the one side and controller engagement on the other. We conclude our work and suggest future steps with a discussion in Chapter 5. A large part of the results was published in June 2018 in the Proceedings of Robotics: Science and Systems [**25**].

CHAPTER 2

Background and Methods

2.1. Filter-Based Shared Control

2.1.1. Maxwell Demon's Algorithm

The Maxwell's Demon Algorithm (MDA) is a combined filter-controller computational unit that operates similar to the philosophical Maxwell's Demon from thermodynamics in that it accepts selected actions that meet a pre-defined criterion and disallows others. Specifically, it allows control actions that drive a system towards a pre-specified desired objective and disallows actions that drive the system away from that objective. The algorithm was originally proposed in [**44**] by Tzorakoleftherakis and Murphey, where the authors demonstrate that even noisy inputs, generated using i.e. a Gaussian distribution, can be a rich source of control authority if filtered in a meaningful manner.

The algorithm was further developed by Fitzsimons et al. [**17**] as a shared control paradigm for joint human-robot systems. In that work, authors show that MDA is able to improve task performance for technology-assisted tasks by simply rejecting "wrong" user inputs (rather than actively assisting in task completion). In the study, 9 subjects attempt the task of inverting a simulated cart-pendulum with MDA-based feedback. They achieve better performance at the task when controlling the system with either an MDA-based software filter or MDA-based haptic feedback, providing initial results that MDA could be applied to design useful human-machine interfaces.

In this work, we build on the idea of using MDA for shared control. We discriminate and evaluate two application scenarios and hence define two respective MDA modes: training and assistance. In training mode, if user actions are rejected, no control is applied to the system; in assistance mode, if user inputs are rejected, they are replaced with a controller-calculated action—this principle is illustrated in Fig. 2.1 on the example of a hand pushing a mass.

In training mode, MDA improves task performance but allows for failure at the task without a rich enough input, making it well suited for robot-aided task training, where active engagement and failure are conducive to learning. Assistance mode, on the other hand, shows promise for being useful in allocating control of assistive devices, where safety and/or task success are of utmost importance. As mentioned above, we evaluate MDA in both modes—in training mode in Chapter 4, and in assistance mode in Chapter 3.

2.1.2. Mode Insertion Gradient (MIG) as a Criterion

In shared control applications, a key part of the Maxwell's Demon Algorithm is the criterion used for evaluating user actions. In both [**44**] and [**17**], MDA was implemented with a criterion based on a direct comparison of user actions to the actions calculated by an optimal controller. In the studies presented here, we instead use the Mode Insertion Gradient (MIG) as an evaluation criterion of user inputs, which allows us to give users more flexibility in the way they approach a task.

Usually, the MIG $\frac{dJ}{d\lambda}$ is used, in mode scheduling problems, to determine the optimal time τ to insert control modes from a predetermined set [**11, 47, 19, 4, 7**]. Here we use

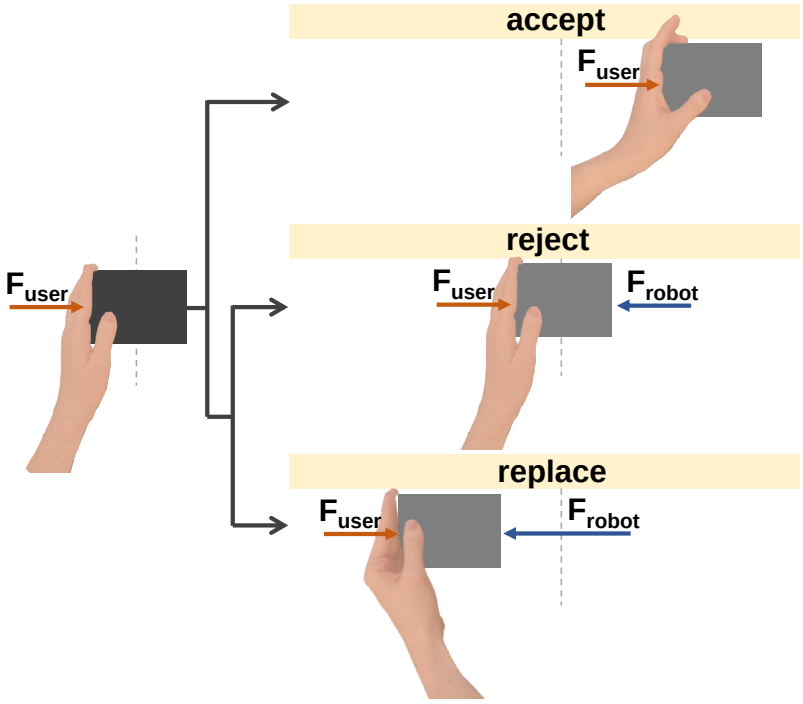

Figure 2.1. Filter-based robotic responses on the example of a hand pushing a mass. The robot filters user input by physically accepting, rejecting, or replacing it. When a user action is accepted, the robot admits the force. When a user action is not accepted, the robot either rejects it by applying an equal and opposite force or replaces it by applying a force, such that the net effect on the system is equal to the controller-calculated input.

the mode insertion gradient,

$$\frac{dJ}{d\lambda}(\tau) = \rho(\tau)^T \left[f(x(\tau), u_2(\tau)) - f(x(\tau), u_1(\tau)) \right], \tag{2.1}$$

as a measure of the sensitivity of the cost to a change from the nominal control, u_1, to a particular user input, u_2. In (2.1), state x is calculated using nominal control, u_1, and ρ is the adjoint variable calculated from the nominal trajectory $x(t)$,

$$\dot{\rho} = -\nabla l_1(x) - D_x f(x, u_1)^T \rho,$$

where $l_1(x,t)$ is the incremental cost and $\rho(t_f) = \nabla m(x(t_f))$. Moreover, in the work presented here, we define the nominal control, u_1, to be equivalent to a null action ($u_1(t) = 0$), and we define u_2 with the piece-wise function below,

$$u_2(t) = \begin{cases} u_{user} & t \leq t_s \\ u_1 & t_s \leq t \leq T \end{cases}$$

where t_s is the sampling time, T is the time window over which we are evaluating system behavior, and u_{user} is a user input recorded at current time t. It is worth noting that, in future work, u_2 could instead be defined by a combination of user input at current time t and actions from an optimal controller over time T into the future. This would add further flexibility to the criterion and give the user more control authority over the joint system, because any user action that could be corrected for by a future optimal action or sequence of optimal actions without destabilizing the system during the time window T would be admitted.

When using MIG as an evaluation criterion, we calculate the integral of the mode insertion gradient over a time window T into the future

$$\int_{t_{now}}^{t_{now}+T} \frac{dJ}{d\lambda}(t)\delta t, \tag{2.2}$$

to evaluate the impact of user control u_2 on the system over time T. We show that, when negative, the integral indicates that u_2 is a descent direction over the entire time horizon, and thus can serve as the basis for evaluating the impact of a current user action on the evolution of a dynamic system over that time window into the future. Moreover, stability can be inferred if (2.2) satisfies a contractive constraint [**10, 45**].

Algorithm 1 A filter with MIG criterion.

Set sampling time t_s and time horizon T. Set mode m to either training (T) or assistance (A). Define objective function for filter and controller.

while $t_0 \leq t_f$ **do**
  Infer user control vector u_{user} from sensor data
  Simulate $x(t)$ and $\rho(t)$ in $[t_{now}, t_{now} + T]$ assuming

$$u = \begin{cases} u_{user} & t \leq t_s \\ 0 & t_s \leq t \leq T \end{cases}$$

  Compute $\int \frac{dJ}{d\lambda}$
  if $\int \frac{dJ}{d\lambda} \leq 0$ **then**
    $u_{now} = u_{user}$
  else
    if $m = T$ **then**
      Assign controller value $u_{now} = 0$
    else if $m = A$ **then**
      Calculate optimal control $u_{controller}$*
      Assign controller value $u_{now} = u_{controller}$
    end if
  end if
  Apply u_{now} for $t \in [t_{now}, t_{now} + t_s]$
end while

*Note that the filter can be used with any model predictive controller (MPC) that can complete the task. Here a controller similar to the conjugate gradient descent method[**28**] was used.

In our experimental study, we utilize the MIG criterion in a filter-based shared control scheme. For an outline of the approach, refer to Algorithm 1.

2.2. Simulated Users

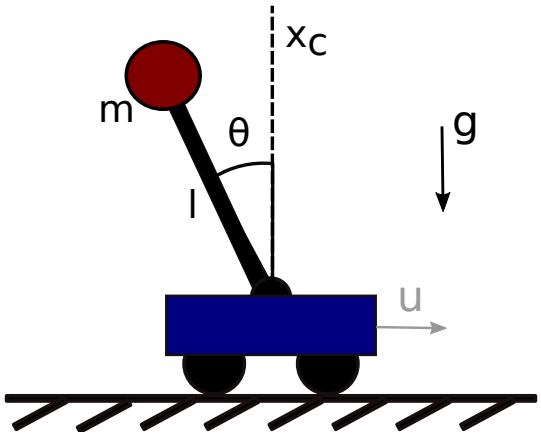

Figure 2.2. The cart-pendulum system with state vector $x = [\theta, \dot{\theta}, x_c, \dot{x_c}]$ and horizontal acceleration of the cart as control input.

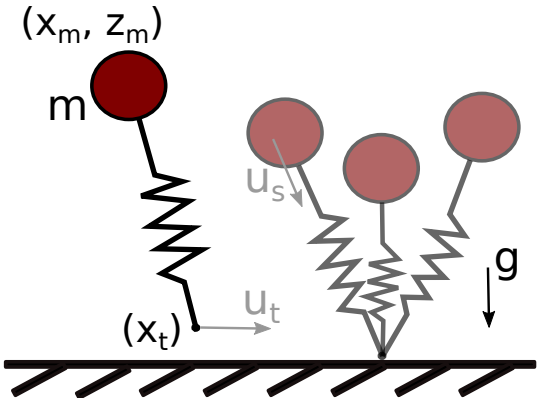

Figure 2.3. The SLIP model with state vector $x = [x_m, \dot{x}_m, z_m, \dot{z}_m, x_t]$ and control vector $u = [u_s, u_t]$, where u_s is the leg thrust applied during stance and u_t is the toe velocity control applied during flight.

To test the proposed algorithm in assistance mode, we run experiments with simulated users. We use Gaussian noise as well as a Model Predictive Controller (MPC) with differing objective functions to simulate users of varying skill levels. Specifically, an objective function defined as

$$J = \frac{1}{2} \int_{t_o}^{t_f} \|x(t) - x_d(t)\|_Q^2 + \|u(t)\|_R^2 \delta t, \tag{2.3}$$

with $Q \geq 0$ and $R \geq 0$ being metrics on state error and control effort and $x_d(t)$ being the desired trajectory is used in the simulated experiments. The systems and specific objective functions are described in the sections below.

2.2.1. Cart-Pendulum Task

A cart-pendulum system as illustrated in Fig. 2.2 is one of the systems used. It has a state vector $x = [\theta, \dot{\theta}, x_c, \dot{x_c}]$ and a horizontal acceleration of the cart u as control input. The task associated with this system is to invert the pendulum to its unstable equilibrium, where $\theta = 0$ and $\dot{\theta} = 0$.

TABLE I: Cart Pendulum Simulation Setup.

	OBJECTIVE
simulated skilled user	$x_d(t) = [0, 0, 0, 0]$ $Q = [200, 0, 50, 0]$ $R = [10]$
MIG filter	$x_d(t) = [0, 0, 0, 0]$ $Q = [200, 0, 0, 0]$ $R = [0.2]$

To create the simulated skilled users, we utilize an MPC with objectives representing successful inversion strategies. An example objective includes inverting the cart-pendulum while minimizing energy and staying close to the origin—the exact function parameters are given in Table 1. To approximate an unskilled user, we generate noise input, using a Gaussian distribution within the saturation limits. While we made the above choices, there are other reasonable alternatives for representing simulated users.

In either case, we then filter user actions using a MIG-based algorithm with a high-level objective function also listed in Table 1. For the MIG filter, we choose to place a high weight only on the angle θ. The goal is to emphasize the task of inversion while minimally limiting users' varying approach strategies.

2.2.2. SLIP Hopper

Another system used is the spring-loaded inverted pendulum. The SLIP is a hybrid, low-dimensional system that has been shown to be a reliable approximation of human running [**41**] and is therefore used to model running dynamics in robotic locomotion [**3**]. Here, a 2D SLIP model (Fig. 2.3) is tested with a state vector described by $x = [x_m, \dot{x}_m, z_m, \dot{z}_m, x_t]$, where x_m and z_m are the coordinates of the mass, and x_t is the coordinate of the toe, and a control vector described by $u = [u_s, u_t]$, where u_s is the leg thrust applied during stance and u_t is the toe velocity control applied during flight. Hybrid dynamics of the form

$$f_{stance} = \begin{pmatrix} \dot{x}_m \\ \frac{(k(l_0 - l_s) + u_s)(x_m - x_t)}{m l_s} \\ \dot{z}_m \\ \frac{(k(l_0 - l_s) + u_s) z_m}{m l_s} - g \\ 0 \end{pmatrix}$$

and $f_{flight} = (\dot{x}_m, 0, \dot{z}_m, -g, \dot{x}_m + u_t)$ are used. Parameters k, l_0, and m describe the SLIP model spring constant, resting spring length, and mass, respectively. All parameters were given a value of 1 in our simulations.

To determine switches between stance and flight modes, a guard equation $\phi(x)$ is employed

$$\phi_{stance \to flight}(x) = \phi_{flight \to stance}(x) = x_m - \frac{l_0}{l_s} z_m$$

with l_s being the leg length during stance

$$l_s = \sqrt{(x_m - x_t)^2 + z_m^2}.$$

In the experiments, we again use input from simulated users of different skill level, which we generate using MPC with objective functions outlined in Table 2. We approximate an unskilled user using Gaussian noise; a low-skill user using MPC with a height objective lower than the spring length, which causes the SLIP to fall if simulated without assistance; and a skilled user using MPC with a feasible objective such that the controller can achieve forward motion without assistance.

TABLE II: SLIP Simulation Setup.

	OBJECTIVE
low-skill user	$x_d(t) = [0, 0.7, 0.7, 0, 0]$ $Q = [0, 150, 100, 0, 0]$ $R = [0.1, 0.1]$
skilled user	$x_d(t) = [0, v(t), 1.7, 0, 0]$ $Q = [0, 150, 100, 0, 0]$ $R = [0.1, 0.1]$ $v(t) = \begin{cases} 0.2m/s & 0s < t \leq 8s \\ 0.2 + 0.05(t-8)m/s & 8s < t \leq 16s \\ 0.6m/s & 16s < t \leq 24s \end{cases}$
MIG filter & controller	$x_d(t) = [0, 0, 1.4, 0, 0]$ $Q = [0, 0, 5, 0, 0]$ $R = [0.1, 0.1]$

Here, the MIG-based filter uses a high-level objective function defining the height of the center of mass z_m, also listed in Table 2. The goal is to emphasize the safe upright

orientation of the SLIP hopper while minimally limiting its movements and velocity in the x-direction.

2.3. Human Subject Experiments

In addition to testing simulated users, we conducted a human subject study, where we implemented and tested the proposed shared control paradigm in the form of a mechanical filter. Subjects used an upper limb robotic platform as an interface to control a simulated cart-pendulum system. During experimental trials, users were instructed to invert the pendulum to its unstable equilibrium and keep it there for as long as possible. User input was inferred from a force sensor at the robot's end-effector and was continually evaluated at 100Hz. During trials when the filter was engaged, user actions were either accepted or rejected based on the criterion described in Section 2.1.2.

2.3.1. Experimental Platform

All human subject data was collected using the robotic platform shown in Fig. 2.4. The device is a powerful haptic admittance-controlled robot that can be used to render virtual objects, forces, or perturbations in three degrees of freedom. It is similar to the robotic platform used in [**12**] and [**42**] to provide a means to modulate limb weight support during reaching and to quantify upper limb motor impairments in stroke survivors.

During the experiment, each subject was seated in a Biodex chair with their arm secured in a forearm-wrist-hand orthosis. The orthosis could rotate passively, and the device could move its end-effector within a workspace defined both by its design limits and limits set by the investigators. At the point where the orthosis was mounted, a

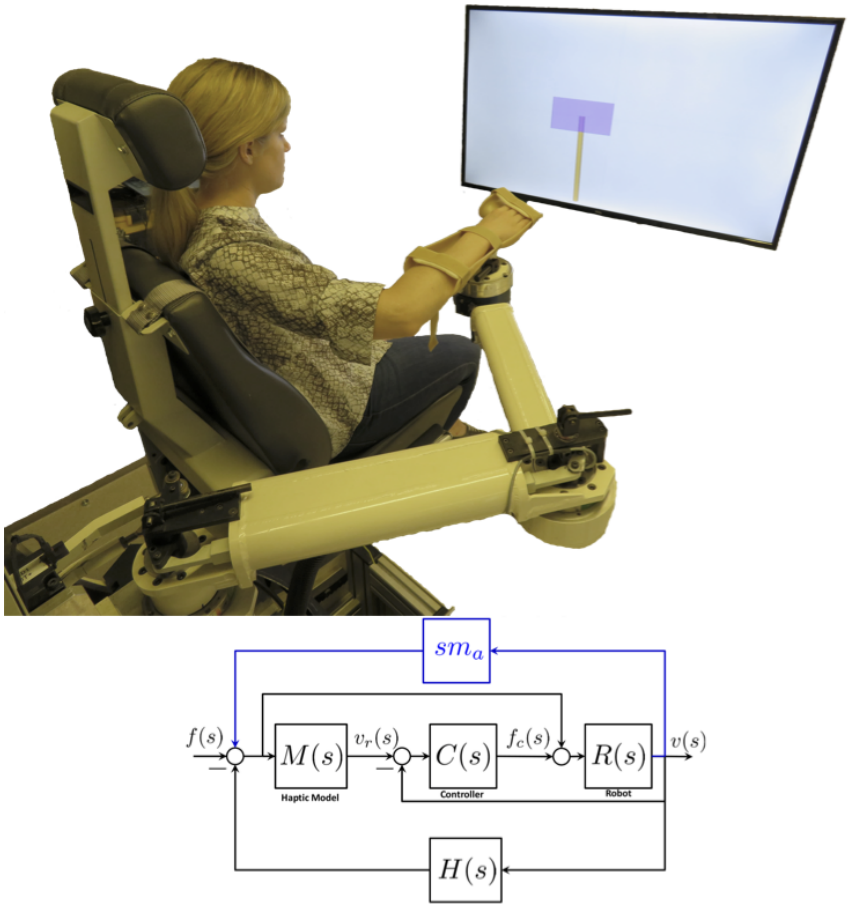

Figure 2.4. (top) Upper limb robotic platform used during experiments. (bottom) The platform provides haptic feedback to simulate a specified inertial model via an admittance control scheme. A voluntary force $f(s)$ is measured by a force-torque sensor at the end-effector and passed through a model $M(s)$ that determines velocity $v_r(s)$ at which the robot should move. The reference velocity is tracked by the low level velocity controls, $C(s)$, of each motor drive. In addition to a force input, the user delivers involuntary impedance forces due to movement, given by dynamics $H(s)$. Acceleration information is fed back as a pseudo-force sm_a for extra inertia reduction of the system.

force-torque sensor measured subject input, which was then fed back to the admittance controller. In our experiments, the device was set up to physically support the upper limb of the participant in the z-direction while allowing them to move freely on the x-y plane.

During testing, a display provided real-time visual state feedback to the user about the cart-pendulum system they were attempting to invert. High stiffness virtual springs in the haptic model were used to restrict user motion to a horizontal plane corresponding to the path of the cart in the virtual display. When user inputs were accepted, the control scheme behaved as described in Fig. 2.4 and the end-effector motion changed according to the applied force. When user inputs were rejected, the measured user input $f(s)$ was ignored in the control scheme, such that the robot continued to move under its predefined dynamics as if no force had been applied by the user.

2.3.2. Experimental Protocol

Twenty-eight subjects (9 males and 19 females) consented to participate in this study.[1] All subjects completed three sets of thirty 30-second trials with short breaks between sets. Each trial consisted of the subject attempting to invert a simulated cart-pendulum system, using cart acceleration as input. At the beginning of each session, the system and task was demonstrated to the subject using a video of a sample task completion. Subjects were instructed to attempt to swing up the simulated pendulum to the upward unstable equilibrium and balance it there for as long as possible. Subjects were instructed to continue to try to do this until the 30 seconds were over even if they succeeded at balancing near the equilibrium at some point throughout the trial.

Upon enrollment, subjects were randomly placed into either a control ($n = 10$) or training group ($n = 18$). During the second set, feedback in the form of a filter was engaged for the training group, while the control group completed each of the three sets

[1]This study protocol was approved by the Institutional Review Board and all participants signed an informed consent form.

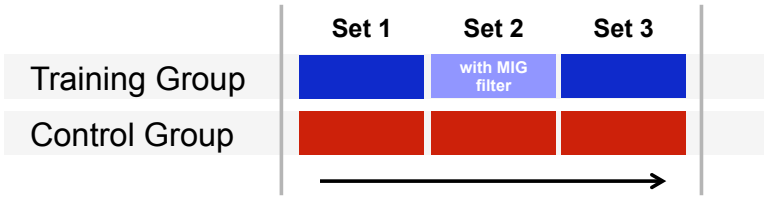

Figure 2.5. Experimental Protocol. Upon enrollment, subjects were randomly placed into either a control ($n = 10$) or training group ($n = 18$). Each session consisted of three sets of thirty trials (each block above represents one such set). In set 1, both groups received no feedback, in set 2 the control group trained without feedback while the training group trained with feedback in the form of a mechanical filter, and in set 3 both groups attempted the task again with no feedback.

without any feedback. Again, each user did three sets of thirty trials: set 1 (both groups: no feedback), set 2 (control: no feedback, training: feedback in the form of a mechanical filter), set 3 (both groups: no feedback). The experimental protocol is illustrated in Fig. 2.5.

2.3.3. Performance Measures

Several measures were calculated to quantify user performance in individual trials. Specifically, time to success, balance time, and error were calculated for all trials and subsequently each trial was classified as successful or unsuccessful.

A trial was considered successful when a subject reached an angle of ± 0.4 rad and angular velocity of ± 0.75 rad/s for at least 2 seconds. This success definition was used to determine the success rate and time to success of the users in each set. In addition, if a subject was successful, the total time spent at an angle of ± 0.4 rad and angular velocity of ± 0.75 rad/s was recorded as the balance time. Note that when users were successful

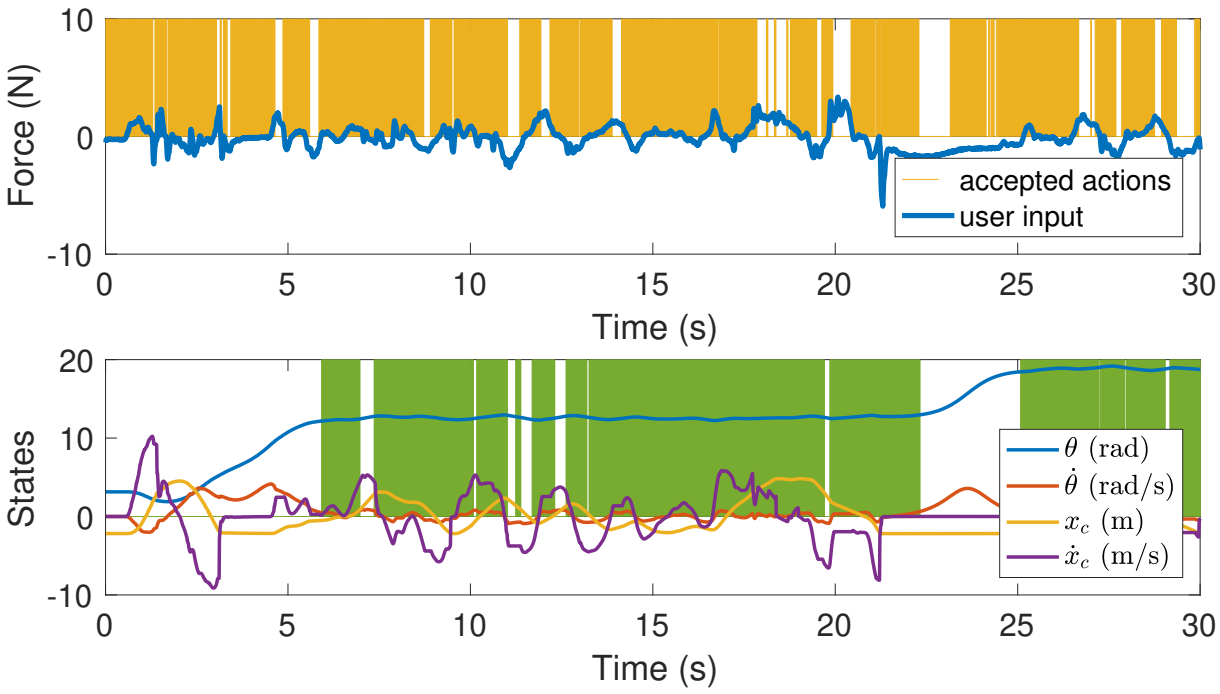

Figure 2.6. Example trial data from study. (top) User force input with an indication of allowed actions in yellow. (bottom) System evolution with green highlighting of the time during which success was recorded. Note: Angle wrapping was not used on θ in the plot above, but it was used in the calculation of all performance measures.

multiple times in the same trial, time spent in the balance region was cumulative. Lastly, an RMS error of each trajectory generated by the users was calculated with respect to the desired position in an inverted unstable equilibrium (zero-vector of the states). RMS error was normalized by the RMS error of a constant trajectory at the stable equilibrium, equivalent to the error of the user not moving from the initial conditions.

A percent of rejected actions (PRA) was also recorded. PRA measured the fraction of user inputs that were rejected up to the time of a successful inversion, where we define an action to be a non-zero user input.

Data from an example trial is visible in Fig. 2.6. In this case the trial was successful, with time to success = 8.3s, balance time = 19.7s, and RMS error = 0.57. The PRA was 13%.

CHAPTER 3

Robotic Assistance with Minimal Interference

When assisting users in physical tasks, there are two highly desired features of a shared control paradigm. For one, we may want to insist on task success, user safety, or both even for unskilled users. Secondly, it should remain as transparent as possible to users at times when they do not require assistance at the task. In the simulation results presented below, we show that the proposed filter-based shared control paradigm in assistance mode possesses both features—it can successfully assist in task completion and/or keep a simulated user safe, while remaining sensitive to the user's skill level and engaging minimally.

3.1. Performance-Oriented Assistance

Experiments were completed using simulated users attempting the task of cart-pendulum inversion, as described in Section 2.2 of Chapter 2.

A series of 100 Monte Carlo simulations were run for unskilled users (as modeled using Gaussian noise) attempting the task of cart pendulum inversion. System behavior, simulated user input, and controller intervention during an example trial are visible in Fig. 3.1. Results of the 100 trials with noise input demonstrate a 100% success rate and are also shown in the figure.

What is more, we are interested in how controller intervention changes according to the skill level of a user. We note close to 0% intervention for a simulated skilled user and

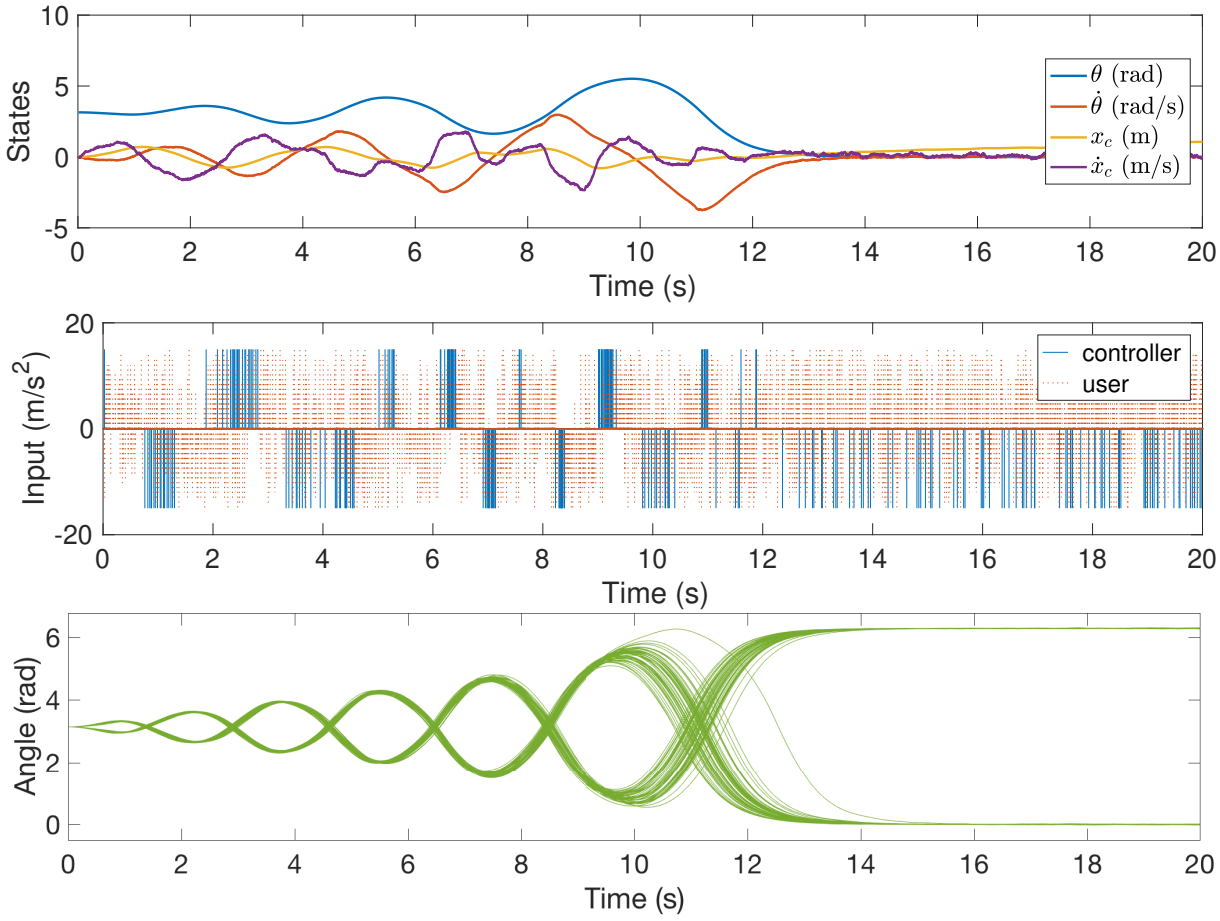

Figure 3.1. For the cart pendulum inversion task, noise input with a MIG-based filter in assistance mode is able to invert the pendulum in 100 out of 100 of the simulation ran. (top, middle) An example trial with the system evolution and filtered input are shown. (bottom) Convergence results from all 100 trials.

$\sim$ 50% intervention for noise input, which makes sense in this one-dimension-controlled task of inverting a pendulum.

Note that for simulated users the relationship between skill and controller intervention is explicit (0% intervention for an always successful user and $\sim$ 50% for noise input). With human subjects, such a relationship is more difficult to assess, because we can only

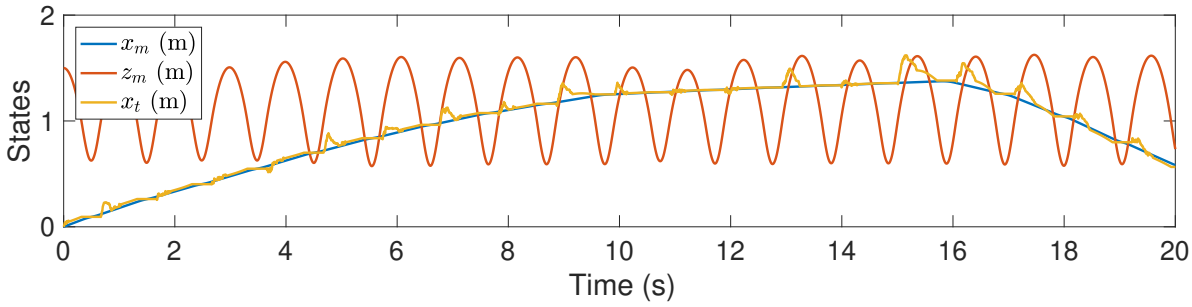

Figure 3.2. We simulate a SLIP model using Gaussian noise as user input and the MIG-filter in assistance mode for support. Note that the filter allows the foot to make random movements and the SLIP to change direction, while keeping the center of mass oscillating around a safe constant height. The controller overrides the user's input for $\sim 70\%$ of the simulation time.

approximate users' skill level and cannot account for momentary mistakes. However, we perform an analysis on data collected in a human subject experiment described in Chapter 4. We identify a weak but analogous relationship between subjects' estimated skill level and controller engagement when using the MIG-based filter in training mode.

3.2. Safety-Oriented Assistance

Next, we analyze the performance of MIG-based assistance on a spring-loaded inverted pendulum (SLIP) model. For our experiments, we use simulated users modeled using MPCs with differing objective functions as described in Section 2.2 of Chapter 2.

We show that with the MIG filter in assistance mode the SLIP can be kept upright even when input is provided by Gaussian noise or a low-skill user. From Fig. 3.2 we see that for noise input the filter allows the foot to make random movements and the SLIP to change direction, while keeping the center of mass oscillating around a safe constant height.

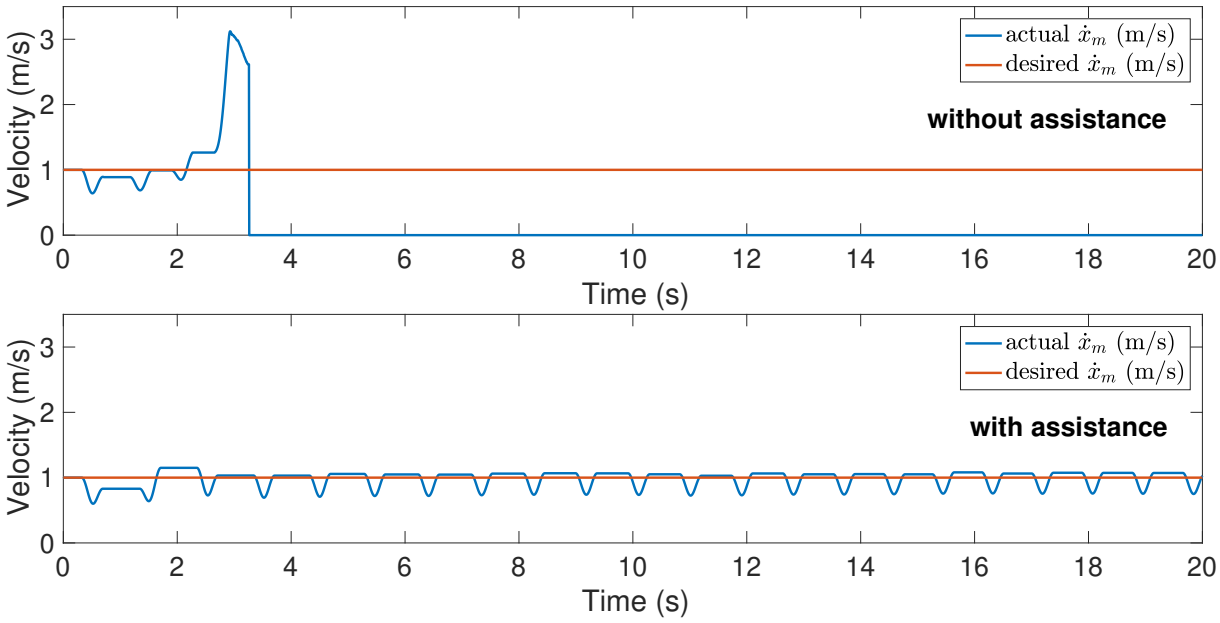

Figure 3.3. (top) We simulate a low-skill user that attempts to move forward with no assistance. The SLIP falls after $\sim 5.5s$. (bottom) We use the same user simulation but now the controller helps the user keep balance without restricting its forward motion. With under 40% controller intervention, the SLIP establishes a cyclic gait and maintains an average speed of 0.98 m/s (close to the user's desired 1 m/s).

For a low-skill user, the assistance prevents the SLIP from falling, while allowing it to maintain its desired forward velocity, as visible in Fig. 3.3. Finally, when provided with input from a skilled user, the filter allows the user to dictate its desired forward velocity and interferes only minimally with its desired motion, as visible in Fig. 3.4.

What is worth noting is that the controller overrides user input for $\sim$ 70% of the time for noise input, for under 40% of the time for a low-skill user, and for $\sim$ 20% of the time for a skilled user. Again, we see a relationship between the skill of the user and controller engagement. The more skilled a user is at the task, the less the controller interferes, while even for noise input the controller is able to keep the hopper upright and safe.

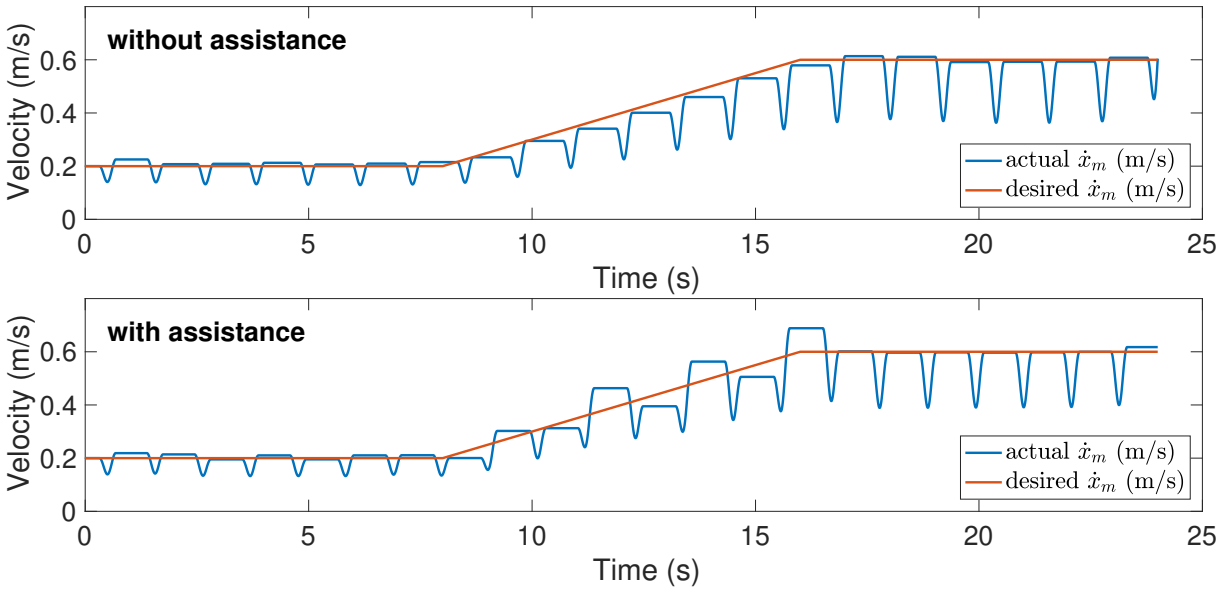

Figure 3.4. (top) We simulate a capable user that attempts to move forward with varying velocity. (bottom) We simulate the same user with added assistance. Note that the assistance does not impede the user's forward motion, even though the controller has no *a priori* knowledge of the user's desired velocity. The controller intervenes $\sim$ 20% of the time.

Based on these results, the MIG criterion shows promise to be used in assistance-oriented applications, such as lower-limb exoskeletons [**20**]. In walking assistance, we want to at all cost prevent users from falling, while at the same time giving them freedom to follow their natural gait pattern, walk at a desired pace, and change speeds or stop when convenient. Such an implementation of the shared control paradigm will be evaluated in future work.

CHAPTER 4

Training Through Forceful Interaction

Whereas during assistance, we prevent users from failing at a task, during training we may want to allow failure for improved learning. In these situations, we can use the proposed filter without actively providing assistance. Instead of using a calculated controller input as the alternative to user input, we can simply reject "wrong" user actions and wait for the user to figure out the "correct" next action.

In the following chapter, we demonstrate results of a human subject study, where we compare learning of the cart-pendulum inversion task between a group training with feedback in the from of the filter-based shared control and a group training for an equivalent duration of time without feedback. Details of how the experiment was set up can be found in Section 2.3 of Chapter 2.

4.1. Training Effect

The main result we were looking for in the study was whether providing filter-based feedback through forceful interaction would improve learning. And indeed, we observed a significant training effect from training with filter-based feedback compared to training with no feedback for an equivalent duration of time, as visible in Fig. 4.1. *Post hoc* analysis confirmed that both groups started the experiment at comparable skill levels and although training time was the main factor impacting improvement in performance

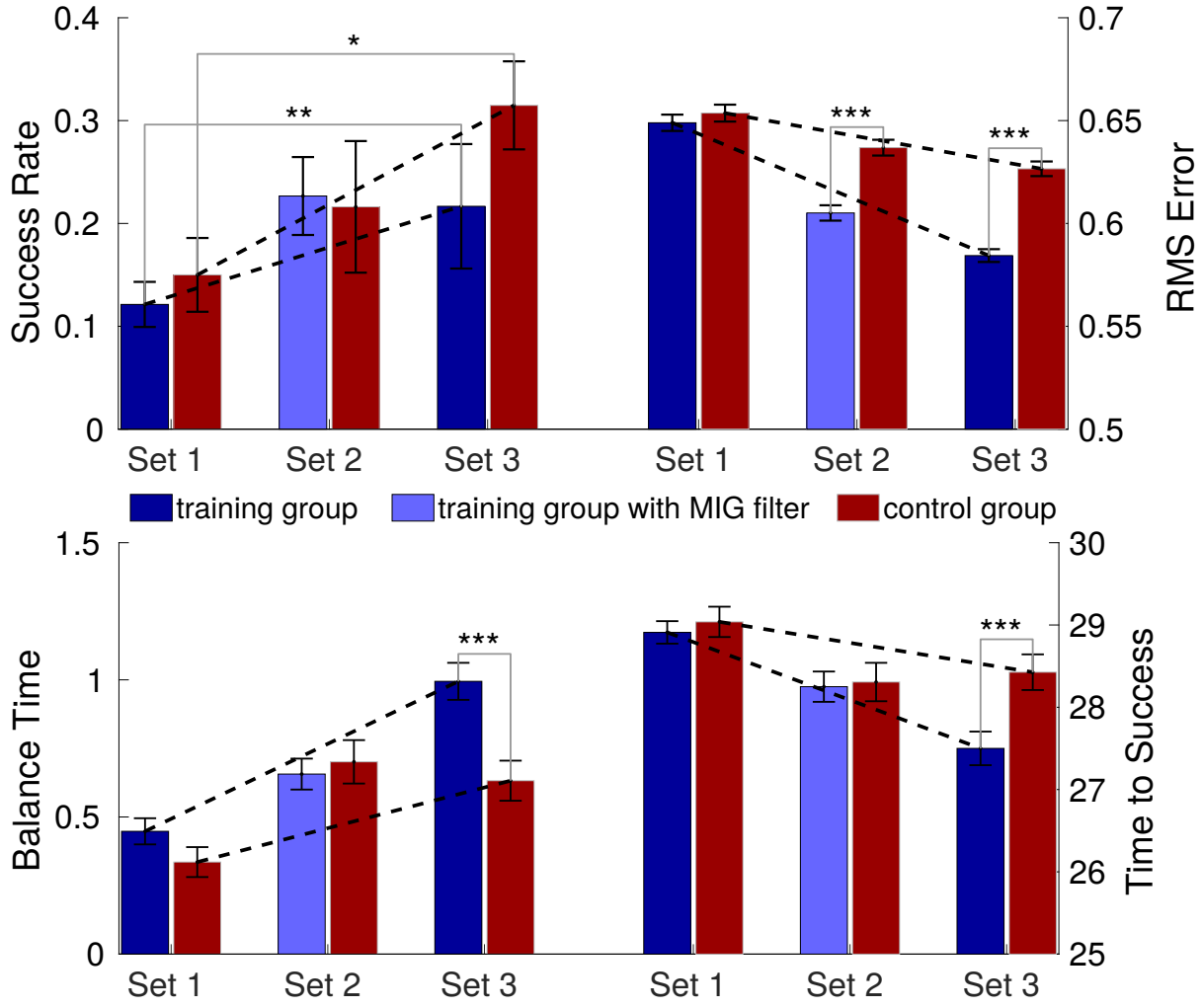

Figure 4.1. There was a training effect from training with the filter-based feedback compared to training with no feedback. The RMS error, balance time, and time to success of the training group in the final set was significantly better than that of the control group. It is also worth noting that as expected, pairwise comparisons of the two groups in set 1 show that there was not a significant difference in their baseline performance measurements. Moreover, set was consistently the most significant factor in performance improvements from set 1 to set 3, showing that both groups (training and control) improved significantly over time. Finally note that error bars indicate standard error and significance is indicated by $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

(both groups improved over time), the training group improved significantly more than the control. Detailed statistical analysis is described below.

A two-factor repeated measures ANOVA was used to assess the effects of the group (between-subjects) and set (within-subjects) on all performance measures listed in Section 2.3.3 of Chapter 2. The training group and control group were evaluated based on set 1 and set 3 only. Set 2 was left out of the ANOVA, so that effects of the assistance itself would not be measured in the analysis. Additionaly, pairwise comparisons were made between set 1 for both groups, between set 3 for both groups, and between set 1 and set 3 within each group using a paired 2 sample t-test.

Firstly, we confirmed that there was on average no difference in skill at the onset of the experiment. Pairwise comparisons within each of the four measures (success rate, RMS error, balance time, and time to success) showed that in set 1 there was not a significant difference between the training group and control group. For instance, the main effect of group in set 1 on the RMS error was not significant ($F(1, 26) = 1.615, p = 0.215$)—there was not a significant difference between the training group ($mean = 0.612, SD = 0.088$) and the control group ($mean = 0.639, SD = 0.067$) in set 1. The same was true for the three other metrics. According to 2-sample t-tests, the differences between the two groups were also not significant across all 4 metrics ($p > 0.01$).

Secondly, we found that time had the most significant impact on learning. The factorial ANOVA revealed that for success rate the main effect of set yielded an F ratio of $F(1, 50) = 7.555,\ p = 0.008$, meaning that users were more successful in set 3 ($mean = 0.280, SD = 0.223$) than in set 1 ($mean = 0.140,\ SD = 0.100$) regardless of the type of practice in set 2. Similarily, the RMS error showed that the main effect of set yielded an

F ratio of $F(1, 26) = 41.551, p < 0.001$, indicating a significant difference between set 1 ($mean = 0.651, SD = 0.085$) and set 3 ($mean = 0.599, SD = 0.072$). The main effect for set on the balance time yielded an F ratio of $F(1, 26) = 15.328,\ p < 0.001$, indicating a significant difference between the balance time in set 1 ($mean = 0.408, SD = 1.053$) and set 3 ($mean = 0.866, SD = 1.476$). The main effect of set on time to success was also significant ($F(1, 26) = 18.992,\ p < 0.001$), with set 3 ($mean = 27.830, SD = 4.433$) outperforming set 1 ($mean = 28.955, SD = 3.175$). Set had a significant effect on increases across all metrics, indicating that participants were continuously improving with time regardless of the feedback that was provided.

Thirdly, we observed that there was more improvement among members of the training group than control. When training group and control group were compared in set 3 using 2-sample t-tests, the training group performed better. The set 3 balance time of the control group ($mean = 0.632, SD = 1.261$) was significantly less ($t(728) = 3.643, p < 0.001$) than the balance time of the training group ($mean = 0.994, SD = 1.568$). The time to success was also significantly better ($t(738) = 3.110, p = 0.002$) in set 3 of the training group ($mean = 27.500, SD = 4.744$) compared to set 3 of the control group ($mean = 28.43, SD = 3.74$). The same was true for the RMS error ($t(699) = -8.919, p < 0.001$)—the RMS error of the training group in set 3 ($mean = 0.58, SD = 0.003$) was lower than the RMS error of the control ($mean = 0.63, SD = 0.004$). The RMS error, balance time, and time to success of the training group in the final set was significantly better than that of the control, while the difference for success rate was not statistically significant.

Moreover, the RMS error showed that there was a significant interaction effect between group and set ($F(1,26) = 5.099, p = 0.0326$). This is indicated in Fig. 4.1 by the two groups having similar means in set 1 but significantly different means in set 3, implying that the training had a greater impact on set 3 performance than the unassisted practice of the control group. For the other measures, the interaction effects were statistically insignificant—there was no significant interaction effect of group and set on balance time ($F(1,26) = 1.048,\ p = 0.315$), time to success ($F(1,26) = 1.512,\ p = 0.22983$) or success rate ($F(1,50) = 0.111,\ p = 0.740$). There was also no signifcant effect of group on any of the metrics: success rate ($F(1,50) = 0.981,\ p = 0.327$), balance time ($F(1,26) = 1.562,\ p = 0.223$), time to success ($F(1,26) = 1.114,\ p = 0.301$), or RMS error ($F(1,26) = 1.824,\ p = 0.189$). In summary, although there was not a significant effect of group on any of the metrics, the RMS error showed that there was a significant interaction effect between group and set.

Finally, in order to assess how subject performance evolved over time, the baseline and post-training sets were analyzed using subsets containing five individual trials, rather than the sets of thirty trials used for the analysis above. As such, there were 6 subsets in each set as shown in Fig. 4.2. We can see that the training and control groups diverage in performance during the subsets of set 2. Particularly with respect to the RMS error, the control group plateaus in improvement while the training group keeps improving. These results support the hypothesis that subjects experience accelerated learning while receiving feedback through filter-based forceful interaction.

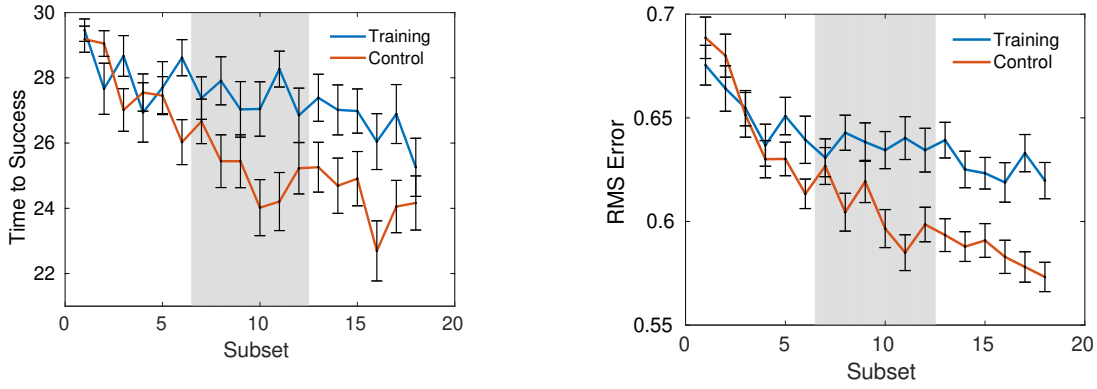

Figure 4.2. Grey band covers subsets included in set 2 (the set, where the training group received feedback). Note that training and control groups showed similar improvement during set 1, while the training group showed faster improvement during set 2. Particularly with respect to the RMS error, the control group's improvement slows down drastically during set 2, while the training group keeps improving.

4.2. Real-Time Performance Improvement

Like detailed in the section above, the two experimental groups performed similarly in their baseline trials. However, in set 2, the group using the filter outperformed the control group in terms of time to success, RMS error, and balance time.

The experimental group ($mean = 25.168, SD = 7.686$) had a lower time to success ($t(798.03) = -4.999, p = 7.067 \times 10^{-7}$) than the control group ($mean = 27.418, SD = 5.266$). The RMS error of the experimental group ($mean = 0.605, SD = 0.087$) was also significantly lower ($t(753.59) = -5.925, p = 4.738 \times 10^{-9}$) than that of the control group ($mean = 0.636, SD = 0.066$). Finally, the balance time of the experimental group ($mean = 2.358, SD = 3.469$) was higher ($t(691.51) = 7.28, p = 9.1 \times 10^{-13}$) than that of the control group ($mean = 1.026, SD = 1.418$), showing that the filter was able to effectively assist subjects in the task and improve all task-specific metrics. There was no

significant difference between the success rate of the control group and the experimental group during set 2 ($p = 0.335$), which makes sense because success rate can only be calculated per set rather than per trial and hence many less samples are available. Comparisons between the control and experimental groups are shown in Fig. 4.1. Overall, these results demonstrate that the MIG filter meets the commonly reported requirement of shared control schemes in that it assists subjects with the task while in use.

4.3. Features of Assist-As-Needed

Like described in the Introduction in Chapter 1, studies emphasize the need for training interfaces to be assist-as-needed. This real-time adjustment to user performance is critical due to two factors:

(1) users start at different skill levels requiring different levels of feedback and assistance

(2) users exhibit varying performance levels over time because of either overall improvement at the task or temporary distractions and fatigue again requiring adequately adjusting levels of shared control.

By remaining sensitive to user skill and current performance, the interface avoids overreliance on the assistance and encourages learning. Uniquely in comparsion to most available solutions, it adjusts its engagement on an instantenous basis without the need to specify or approximate additional parameters, such as machine trust in the user. In the sections below, we report experimental results that demononstrate correlations between controller engagement and both initial skill level and within-trial performance.

4.3.1. Correlation with Initial Skill Level

From the experimental results, a relationship was observed between participant skill level—estimated based on performance in unassisted trials—and the frequency of controller intervention in the training set. In this case, we calculate the success rate of the 30 trials from set 1 to approximate user skill level. We then use percent of rejected actions (PRA) values from individual trials in set 2 from the same users to identify the correlation—a scatter plot with the results is visible in Fig. 4.3. A Pearson product-moment correlation coefficient was computed and a low negative correlation ($r = -0.14$, confidence interval $(-0.22) - (-0.06)$, $p = 0.001$) was identified between overall success rate in set 1 and PRA in individual trials of set 2 for the training group ($n = 18$). Similar but weaker correlations were identified between controller intervention and other task-specific metrics, such as balance time ($r = -0.09$, confidence interval $(-0.17) - (-0.007)$, $p = 0.03$) and time to success ($r = 0.11$, confidence interval $0.086 - 0.25$, $p = 0.01$).

Since subjects showed significant improvement during set 1 while getting used to the task and testing platform, we ran the same statistics using only the last 10 trials of set 1 to estimate participant skill level. Again, a Pearson product-moment correlation coefficient was computed and a low negative correlation ($r = -0.23$, confidence interval $(-0.31) - (-0.14)$, $p = 1.0 \times 10^{-7}$) was identified between overall success rate in the last 10 trails of set 1 and PRA in individual trials of set 2. Similar correlations were identified between controller intervention and other task-specific metrics, such as balance time ($r = -0.14$, confidence interval $(-0.23) - (-0.06)$, $p = 0.0007$) and time to success ($r = 0.24$, confidence interval $0.16 - 0.32$, $p = 1.2 \times 10^{-8}$).

Measure	Test Sign	**r**	p
Success Rate	$-$	$\mathbf{-0.23}$	1.0×10^{-7}
Balance Time	$-$	$\mathbf{-0.14}$	0.0007
Time to Success	$+$	$\mathbf{+0.24}$	1.2×10^{-8}

Table 4.1. Correlation results between initial skill level and controller engagement. Pearson's correlation tests were performed by applying a linear model to the participant skill level, as estimated from performance measures in the last 10 trials of set 1, and the frequency of controller intervention in the training set—the PRA in set 2. The expected sign of the correlation coefficient (r) for a shared control scheme that is sensitive to the performance of the user is indicated in the column on the left. There was a weak but significant correlation between the current performance of the user and PRA for success rate, balance time, and time to success. For RMS error, results were statistically insignificant ($p = 0.86$).

Overall, for an experimental group of 18 participants, we obtained low but significant correlations [**9**] between independently measured performance metrics and rejection rate (PRA) in assisted trials, suggesting a relationship between the users' skill level and the MIG filter's rate of intervention. Because the correlations are weak, additional subjects and analysis of other tasks are needed before the skill sensitivity is conclusive. However, our initial findings suggest that a MIG-based filter is a skill-sensitive paradigm that can be used for shared control. MIG-based feedback resembles assist-as-needed shared control schemes, which have been in previous studies shown to be advantageous for learning.

4.3.2. Correlation with User Performance

Since the filter adjusts on an instaneous basis, we also looked at how it would adapt to a user's overall performance during engagement. We found a weak correlation ($r = 0.17$) between the success rate and percent of rejected actions, which demonstrate that the

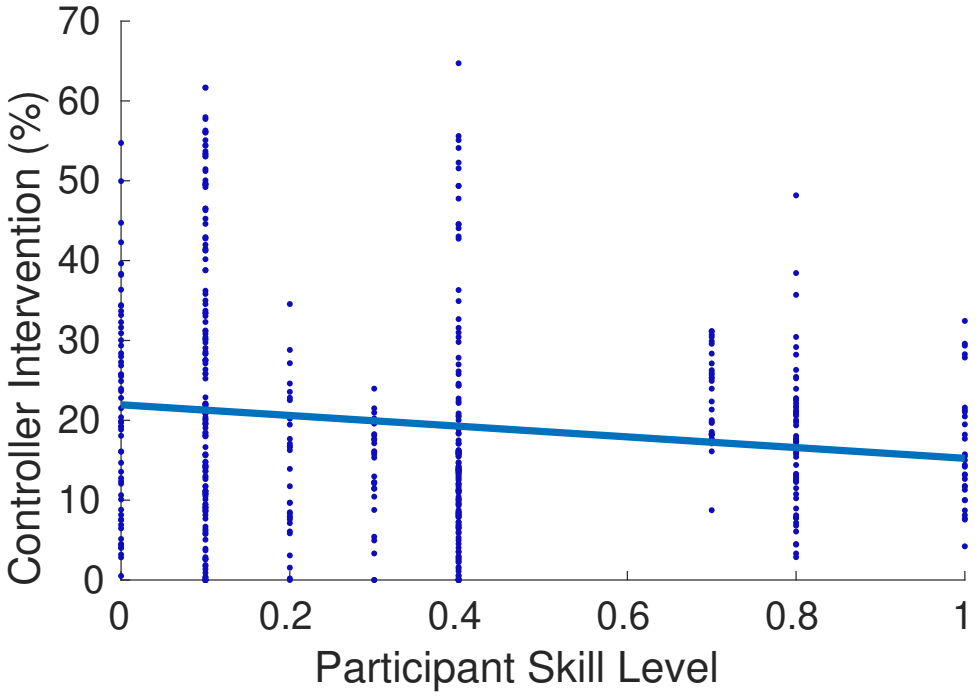

Figure 4.3. A correlation coefficient of -0.23 is observed between the success rate of the users in set 1 with no assistance and the rejection rate of the users' inputs in set 2 with assistance, suggesting a correlation between the users' adeptness at the task and the controller's intervention rate during assistance.

controller engages less often when the user is performing better at the task. Correlation coefficients for other metrics were statistically insignificant, so more testing would have to be conducted to further evaluate the filter.

Overall though, these results suggest that the proposed filter exhibits features of assist-as-needed paradigms both in terms of adaptation to initial user skill level and adaptation to real-time performance.

CHAPTER 5

Conclusions and Future Work

A variety of shared control paradigms have been implemented to provide feedback and assistance to users in settings where the task is known *a priori*. Although users might prefer to maintain control and user engagement is necessary for learning, many applications require a certain level of control authority to be allocated to the machine in order to guarantee safety, improve successful task completion, and/or accelerate learning. As such, most interfaces employ support strategies that in various ways restrict or adjust users' actions in order to enable the subject and the device to move in a safe and constructive manner.

In this work, we present and evaluate an assessment criterion for user input that can be used in these shared control paradigms. We carry out experiments by using the MIG as an evaluation criterion in a filter-based shared control scheme, similar to [**1, 17**], where user actions deemed by the filter as incorrect are either blocked or hindered by the hardware interface. With only current state information, our proposed filter can both reject unhelpful inputs and remain transparent to operators with significant skill avoiding unnecessary interference improving safety or accelerating training.

An important feature of the presented interface is its adaptability in real-time. It requires no predefined trajectory, runs on an indefinite time horizon, and automatically adapts to operator skill. It can, like existing adaptive methods, enhance performance while avoiding some of the common long-term pitfalls of "static" automation such as

over-reliance, skill degradation, and reduced situation awareness [**33**]. For complex tasks that require human operators, such as operating a crane or flying an aircraft, these features can alleviate or minimize the need for training with virtual simulators by ensuring safety of the physical system in action. For therapeutic applications, a MIG MDA interface may accelerate recovery or provide assistance by preventing patient slacking, relieving frustration, and utilizing intentioned but noisy signals (e.g., tremor and spasticity) of patients with sensorimotor disorders. Particularly during support in dynamic tasks, such as walking with an exoskeleton, the algorithm can help provide meaningful assistance and ensure safety of the operator-machine system without limiting the user's freedom to maneouver while coupled with the device.

In future work, the proposed evaluation criterion and filter-based shared control scheme should be explored further. Experimental and theoretical work with MDA in assistance mode should be conducted to establish whether guarantees can be made about the safety of the human-machine system. Further experiments should be carried out to make direct comparisons of MIG MDA to other assist-as-needed controllers. Additionally, we would like to explore other objective functions, such as ergodic objectives or barrier functions, instead of currently used quadratic cost on state to test whether we can make the interface even less restrictive to the users' chosen strategies. At the same time, subjects should be tested in higher dimensional tasks to see if all features of the paradigm hold true even when task complexity increases. And lastly, studies of impaired subjects training in clinically relevant tasks should be performed.

References

[1] David A Abbink, Mark Mulder, and Erwin R Boer. Haptic shared control: smoothly shifting control authority? *Cognition, Technology & Work*, 14(1):19–28, 2012.

[2] Javier Alonso-Mora, Pascal Gohl, Scott Watson, Roland Siegwart, and Paul Beardsley. Shared control of autonomous vehicles based on velocity space optimization. In *IEEE International Conf. on Robotics and Automation*, pages 1639–1645, 2014.

[3] Richard Altendorfer, Uluc Saranli, Haldun Komsuoglu, Daniel Koditschek, H Benjamin Brown, Martin Buehler, Ned Moore, Dave McMordie, and Robert Full. Evidence for spring loaded inverted pendulum running in a hexapod robot. In *Experimental Robotics VII*, pages 291–302. 2001.

[4] Alexander R Ansari and Todd D Murphey. Sequential action control: closed-form optimal control for nonlinear and nonsmooth systems. *IEEE Transactions on Robotics*, 32(5):1196–1214, 2016.

[5] Brenna D Argall. Turning assistive machines into assistive robots. In *Quantum Sensing and Nanophotonic Devices XII*, volume 9370, 2015.

[6] Elaine A Biddiss and Tom T Chau. Upper limb prosthesis use and abandonment: a survey of the last 25 years. *Prosthetics and Orthotics International*, 31(3):236–257, 2007.

[7] Timothy M Caldwell and Todd D Murphey. Projection-based optimal mode scheduling. *Nonlinear Analysis: Hybrid Systems*, pages 59–83.

[8] Tom Carlson and Yiannis Demiris. Increasing robotic wheelchair safety with collaborative control: Evidence from secondary task experiments. In *IEEE International Conf. on Robotics and Automation*, pages 5582–5587, 2010.

[9] Jacob Cohen. A power primer. *Psychological bulletin*, 112(1):155, 1992.

[10] Simone Loureiro de Oliveira Kothare and Manfred Morari. Contractive model predictive control for constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 45(6):1053–1071, 2000.

[11] Magnus Egerstedt, Yorai Wardi, and Henrik Axelsson. Transition-time optimization for switched-mode dynamical systems. *IEEE Transactions on Automatic Control*, 51(1):110–115, 2006.

[12] Michael D Ellis, Yiyun Lan, Jun Yao, and Julius PA Dewald. Robotic quantification of upper extremity loss of independent joint control or flexion synergy in individuals with hemiparetic stroke: a review of paradigms addressing the effects of shoulder abduction loading. *Journal of NeuroEngineering and Rehabilitation*, 13(1):95, 2016.

[13] Michael D. Ellis, Theresa Sukal-Moulton, and Julius P. A. Dewald. Progressive shoulder abduction loading is a crucial element of arm rehabilitation in chronic stroke. *Neurorehabilitation and Neural Repair*, 23(8):862–869, 2009.

[14] Jeremy L Emken, Raul Benitez, Athanasios Sideris, James E Bobrow, and David J Reinkensmeyer. Motor adaptation as a greedy optimization of error and effort. *Journal of neurophysiology*, 97(6):3997–4006, 2007.

[15] Jeremy L Emken, Susan J Harkema, Janell A Beres-Jones, Christie K Ferreira, and David J Reinkensmeyer. Feasibility of manual teach-and-replay and continuous impedance shaping for robotic locomotor training following spinal cord injury. *IEEE Transactions on Biomedical Engineering*, 55(1):322–334, 2008.

[16] Moria E Fisher, Felix C Huang, Zachary A Wright, and James L Patton. Distributions in the error space: Goal-directed movements described in time and state-space representations. In *IEEE International Conf. on Engineering in Medicine and Biology*, pages 6953–6956, 2014.

[17] Kathleen Fitzsimons, Emmanouil Tzorakoleftherakis, and Todd D Murphey. Optimal human-in-the-loop interfaces based on Maxwell's Demon. In *American Control Conference*, pages 4397–4402, 2016.

[18] Amos Freedy, Ewart DeVisser, Gershon Weltman, and Nicole Coeyman. Measurement of trust in human-robot collaboration. In *International Symposium on Collaborative Technologies and Systems*, pages 106–114, 2007.

[19] Humberto Gonzalez, Ram Vasudevan, Maryam Kamgarpour, Shankar S. Sastry, Ruzena Bajcsy, and Claire Tomlin. A numerical method for the optimal control of switched systems. In *IEEE Conf. on Decision and Control*, pages 7519–7526.

[20] Peter Gwynne. Technology: mobility machines. *Nature*, 503(7475):S16–S17, 2013.

[21] Peter A Hancock, Deborah R Billings, Kristin E Schaefer, Jessie YC Chen, Ewart J De Visser, and Raja Parasuraman. A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors*, 53(5):517–527, 2011.

[22] Walid Hassani, Samer Mohammed, and Yacine Amirat. Real-time EMG driven lower limb actuated orthosis for assistance as needed movement strategy. In *Proceedings of Robotics: Science and Systems*, 2013.

[23] Sandy H. Huang, David Held, Pieter Abbeel, and Anca D. Dragan. Enabling robots to communicate their objectives. In *Proceedings of Robotics: Science and Systems*, 2017.

[24] LE Kahn, WZ Rymer, and DJ Reinkensmeyer. Adaptive assistance for guided force training in chronic stroke. In *IEEE International Conf. on Engineering in Medicine and Biology*, pages 2722–2725, 2004.

[25] Aleksandra Kalinowska, Kathleen Fitzsimons, Julius Dewald, and Todd D Murphey. Online user assessment for minimal intervention during task-based robotic assistance. 2018.

[26] Hermano Igo Krebs, Jerome Joseph Palazzolo, Laura Dipietro, Mark Ferraro, Jennifer Krol, Keren Rannekleiv, Bruce T Volpe, and Neville Hogan. Rehabilitation robotics: Performance-based progressive robot-assisted therapy. *Autonomous Robots*, 15(1):7–20, 2003.

[27] Axel Lankenau and Thomas Rofer. A versatile and safe mobility assistant. *IEEE Robotics & Automation Magazine*, 8(1):29–37, 2001.

[28] L. Lasdon, S. Mitter, and A. Waren. The conjugate gradient method for optimal control problems. *IEEE Transactions on Automatic Control*, 12(2):132–138, 1967.

[29] Laura Marchal-Crespo and David J Reinkensmeyer. Review of control strategies for robotic movement training after neurologic injury. *Journal of Neuroengineering and Rehabilitation*, 6(1):20–35, 2009.

[30] Andreas Mayr, Markus Kofler, Ellen Quirbach, Heinz Matzak, Katrin Fröhlich, and Leopold Saltuari. Prospective, blinded, randomized crossover study of gait rehabilitation in stroke patients using the lokomat gait orthosis. *Neurorehabilitation and Neural Repair*, 21(4):307–314, 2007.

[31] Katharina Muelling, Arun Venkatraman, Jean-Sebastien Valois, John E Downey, Jeffrey Weiss, Shervin Javdani, Martial Hebert, Andrew B Schwartz, Jennifer L Collinger, and J Andrew Bagnell. Autonomy infused teleoperation with application to brain computer interface controlled manipulation. *Autonomous Robots*, pages 1–22, 2017.

[32] Allison M Okamura. Methods for haptic feedback in teleoperated robot-assisted surgery. *Industrial Robot: An International Journal*, 31(6):499–508, 2004.

[33] Raja Parasuraman, Michael Barnes, Keryl Cosenzo, and Sandeep Mulgund. Adaptive automation for human-robot teaming in future command and control systems. *The International C2 Journal*, 1(2):43–68, 2007.

[34] James L Patton, Mary Ellen Stoykov, Mark Kovic, and Ferdinando A Mussa-Ivaldi. Evaluation of robotic training forces that either enhance or reduce error in chronic hemiparetic stroke survivors. *Experimental brain research*, 168(3):368–383, 2006.

[35] James L Patton, Mary Ellen Stoykov, Mark Kovic, and Ferdinando A Mussa-Ivaldi. Evaluation of robotic training forces that either enhance or reduce error in chronic hemiparetic stroke survivors. *Experimental Brain Research*, 168(3):368–383, 2006.

[36] David J Reinkensmeyer and Volker Dietz. Introduction: Rational for machine use. In *Neurorehabilitation Technology*, pages xvii–xxii. 2016.

[37] Robert Riener, Lars Lunenburger, Saso Jezernik, Martin Anderschitz, Gery Colombo, and Volker Dietz. Patient-cooperative strategies for robot-aided treadmill training: first experimental results. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 13(3):380–394, 2005.

[38] Dorsa Sadigh, Anca D Dragan, Shankar Sastry, and Sanjit A Seshia. Active preference-based learning of reward functions. In *Proceedings of Robotics: Science and Systems*, 2017.

[39] Siddharth Sanan, Stephen Tully, Andrea Bajo, Nabil Simaan, and Howie Choset. Simultaneous compliance and registration estimation for robotic surgery. In *Proceedings of Robotics: Science and Systems*, 2014.

[40] Richard A Schmidt and Robert A Bjork. New conceptualizations of practice: Common principles in three paradigms suggest new concepts for training. *Psychological Science*, 3(4):207–218, 1992.

[41] Manoj Srinivasan and Andy Ruina. Computer optimization of a minimal biped model discovers walking and running. *Nature*, 439(7072):72, 2006.

[42] Arno HA Stienen, Jacob G McPherson, Alfred C Schouten, and Jules PA Dewald. The ACT-4D: a novel rehabilitation robot for the quantification of upper limb motor impairments following brain injury. In *IEEE International Conf. on Rehabilitation Robotics*, pages 1–6, 2011.

[43] Kurt A Thoroughman and Reza Shadmehr. Learning of action through adaptive combination of motor primitives. *Nature*, 407(6805):742, 2000.

[44] Emmanouil Tzorakoleftherakis and Todd D Murphey. Controllers as filters: Noise-driven swing-up control based on Maxwell's Demon. In *IEEE Conf. on Decision and Control*, pages 4368–4374, 2015.

[45] Emmanouil Tzorakoleftherakis and Todd D Murphey. Iterative sequential action control for stable, model-based control of nonlinear systems. *IEEE Transactions on Automatic Control*, 2018.

[46] Letian Wang, Edwin HF van Asseldonk, and Herman van der Kooij. Model predictive control-based gait pattern generation for wearable exoskeletons. In *IEEE International Conf. on Rehabilitation Robotics*, pages 1–6, 2011.

[47] Yorai Wardi and Magnus Egerstedt. Algorithm for optimal mode scheduling in switched systems. In *American Control Conference*, pages 4546–4551, 2012.

[48] Ronald Wilcox, Stefanos Nikolaidis, and Julie Shah. Optimization of temporal dynamics for adaptive human-robot interaction in assembly manufacturing. In *Proceedings of Robotics: Science and Systems*, 2012.

[49] Carolee J Winstein, Patricia S Pohl, and Rebecca Lewthwaite. Effects of physical guidance and knowledge of results on motor learning: support for the guidance hypothesis. *Research Quarterly for Exercise and Sport*, 65(4):316–323, 1994.

[50] Eric T Wolbrecht, Vicky Chan, David J Reinkensmeyer, and James E Bobrow. Optimizing compliant, model-based robotic assistance to promote neurorehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(3):286–297, 2008.

[51] Erkang You and Kris Hauser. Assisted teleoperation strategies for aggressively controlling a robot arm with 2D input. In *Proceedings of Robotics: Science and Systems*, 2012.