Master's thesis

Improving the Quality of Haptic Feedback Yields Only Marginal Improvements in Teleoperated Task Performance

J.G.W. Wildenbeest

Delft, 24th of September 2010

Student No.: 1232630
Report No.: 1185

BioMechanical Engineering
Department of Mechanical Engineering
Delft University of Technology

Improving the Quality of Haptic Feedback Yields Only Marginal Improvements in Teleoperated Task Performance

Jeroen Gerrit Willem Wildenbeest

Committee:

Prof. Dr. F.C.T. van der Helm	*BioMechanical Engineering, Department of Mechanical Engineering*
Dr. Ir. D.A. Abbink	*BioMechanical Engineering, Department of Mechanical Engineering*
Dr. Ir. C.J.M. Heemskerk	*Heemskerk Innovative Technology B.V., The Netherlands*
Dr. E. Garcia Canseco	*Eindhoven University of Technology, Control Systems Technology, Department of Mechanical Engineering*

Coaches:

Dr. Ir. D.A. Abbink
Dr. Ir. C.J.M. Heemskerk

Preface

Throughout my Master's study I have been compelled by the endless possibilities of robotics on one hand and the stunning mechanics and versatility of the human being on the other hand. Teleoperations combine these two aspects. They join human and machine strengths, leaving the potential to the imaginary. Yet, this relatively young and multidisciplinary field yields many challenges, focused around the understanding of human, machine and, moreover, their synergy.

This Master's thesis aims to answer the question: *"To what extend does the quality of haptic feedback contribute to human-in-the-loop task performance, using a telemanipulator as a tool?"*

The focal point of this Master's thesis is a human factors experiment, which is extensively described by a paper. While the appendices provide a comprehensive overview of the accomplishments during this thesis, the objective of the appendices is merely to provide background information and moreover, to allow possible future students or researchers to get insight in the challenges as encountered in this study.

Appendix A describes the efforts with respect to the experimental setup, whereas Appendix B is devoted to a human identification experiment. The ambition to simulate a human operator performing a task in a remote environment is described in Appendix C. Several elements of the human factors experiment are discussed in more detail in Appendix D. Moreover, Appendix D provides a comprehensive overview of the results.

A USB-disk, containing all raw measurement data, all code, all pictures, all literature, etc., that might be of interest has been submitted to the Biomechanical Engineering depository, all of which available on request.

Improving the Quality of Haptic Feedback Yields Only Marginal Improvements in Teleoperated Task Performance

J.G.W. Wildenbeest

Improving the Quality of Haptic Feedback Yields Only Marginal Improvements in Teleoperated Task Performance

Jeroen G.W. Wildenbeest[1,2], David A. Abbink[1], Cock J.M. Heemskerk[2] and Frans C.T. van der Helm[1]

Abstract—In teleoperations, haptic feedback allows the human operator to touch the remote environment. Yet it is only partially understood to what extent the quality of haptic feedback contributes to human-in-the-loop task performance. This paper presents a human factors experiment in which telemanipulated task performance is assessed for three stages of haptic feedback: low-frequency haptic feedback, combined low- and high-frequency haptic feedback and a complete spectrum of haptic feedback in a manual equivalent of the task. Four generalized fundamental tasks have been defined, namely i) Free Air Movement Tasks, ii) Contact Transition Task, iii) Constrained Position Tasks and iv) Force Tasks, all of which identified in a single bolt-and-spanner task. The results show that the overall task performance is predominantly improved by providing low-frequency haptic feedback, especially in Constrained Position and Force Tasks. However, by further improving the quality of the haptic feedback, the task performance is only marginally increased.

Index Terms—Teleoperations, Task Performance, Haptics, Force feedback, Multimodal Feedback

1 Introduction

Telemanipulators allow humans to complete tasks in remote environments, by virtually relocating the human into the remote environment; *telepresence* as Sheridan [40] called it. A broad variety of teleoperated tasks exist, which all impose different requirements on the telemanipulator and its human operator. While haptic feedback is generally considered to improve task performance for most tasks, it is not very well understood how important the quality of the haptic feedback is.

A telemanipulator consists out of a pair of robotic manipulators and a controller. Movements executed by the *human operator* on the *master* device are translated to the *slave* device, which interacts with the *remote environment*. Master and slave are connected through a *controller*. The connected system of human operator, telemanipulator and environment is referred to as the Connected Telemanipulator System (CTS) [8]. This is illustrated in Fig. 1.

In order to understand the significance of haptic feedback, it is important to understand the three main elements of the Connected Telemanipulator System: the *telemanipulator*, the *human operator* and the *environment*.

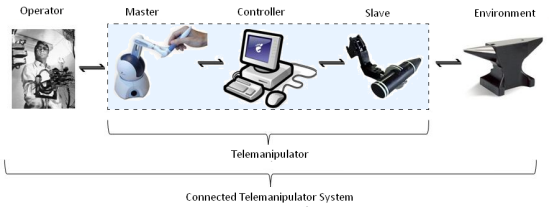


Figure 1: Elements of the Connected Telemanipulator System, adapted from Christiansson [8]

Telemanipulators allow humans to make use of their unique skills in remote environments (e.g. their judgement and insight), by inherently providing a bilateral information flow. Forces and movements are transfered from the human operator through the telemanipulator to the environment and vice versa. It has been shown that providing force feedback from the environment to the human operator yields a reduction in task completion time [13] [17] [32], energy consumption [17], errors indices [13] [17], the magnitudes of the contact forces [13] and the user's cognitive workload [43].

To control the telemanipulator, a diversity of architectures exists, using position or force channels for information transfer in different kinds of combinations (e.g. [3] [8] [11] [14] [21] [31]). Models like the widely used two-port network analogy [15], allow

[1] *BioMechanical Engineering, Department of Mechanical Engineering, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands*
[2] *Heemskerk Innovative Technology, Jonckerweg 20, 2201 DZ Noordwijk, The Netherlands*

for accurate evaluation of device performance and stability [11] [16] and for unambiguously comparison of different control architectures [3]. Whereas it is generally suggested that closed-loop control by means of force and position (4-Channel Control) yields superior performance [3] [8] [31], less sophisticated control architectures like the position-error controller (PERR-Control) are preferred in conservative fields where system availability is a key factor [39].

Commonly, telemanipulators are optimized towards a virtual rod with infinite small mass and infinite large stiffness (optimize towards transparency [14]). But this is not necessarily the right approach to obtain Connected Telemanipulator System task performance. For example, Christiansson showed that an inferior controller in terms of device performance (e.g. reflected mass, damping, transparency), can yield an increase in terms of task performance (e.g. time-to-complete, contact forces) compared to the classic 4-Channel controller, due to the inherent slack properties of the slave [7]. This illustrates that improving the quality of the tool, and thus increasing device performance, does not necessarily improve task performance. Task performance is primarily dependent on how effectively the *human* can control the system. However, for the human to control the system, adequate haptic feedback needs to be provided. But when is sufficient haptic feedback provided?

Given the feedback cues available, the *human operator* creates a somatosensory representation of the situation. This representation is often referred to as the body image or body scheme [6]. Ideally, tools (e.g. a teleoperator) are completely included into the body scheme, such that the tools become 'transparent'. In haptic teleoperation, this body scheme is build up out of the two feedback sources available; visual and haptic feedback. It is suggested that this combination of feedback is most effective out of all combinations of the human's three main modalities (auditory, haptics, visual) [43].

While human operators are biased to attend vision, haptic feedback driven manipulation is faster than the visual equivalent [27]. For telemanipulators, the task determines the contributions of the feedback channels: if the task is familiar or if the task does not require accuracy (e.g. free air movements) vision will be dominating. If the situation is unknown, accuracy is required or if visual feedback is decreased, the human operator will increasingly rely on haptic information. Klatzy strengthens this, by stating that explorative tasks are generally performed rapid and accurate when the bias to attend vision is removed [26]. For which teleoperated tasks does the human operator bias the visual system and for which tasks does the bias shift towards the the haptic channel, and moreover, (how) does the bias affect task performance?

Based on the physiological properties, Daniel and McAree argue that there is a natural partition of the haptic modality, namely a highly energetic part below roughly 20 Hz and an information conveying part above roughly 30 Hz [11]. Proprioceptors (interoceptive haptic receptors), like Golgi-tendon organs and Muscle spindles, foresee the information for the *low-frequency* haptic channel. Mechanoreceptors (exteroceptive touch receptors) are the dominant factors in the *high-frequency* haptic channel. Situated in the muscles, proprioception receptors are the dominating sensors in coordinating movements. Mechanoreceptors, in contrary, contribute to the awareness of the environment. Their reflexive properties and high sensitivity make them susceptible for low-amplitude high-frequency signals (vibrations) [41].

Derived from what the human is physically able to perceive, Brooks states that a presented force signal is required to have a minimal bandwidth of 20-30 Hz for "meaningful perception", while "near optimal performance" is achieved for a 0-400 Hz bandwidth [5]. Moreover, Brooks itemizes the bandwidth of environmental signals, by estimated slow skin motion at $\sim$0 Hz, compressive stresses at $\sim$10 Hz, controlled slip at $\sim$30 Hz and vibrations up to $\sim$400 Hz). But how to these bandwidths relate to teleoperated task performance? At what bandwidth does task performance saturate when performing a teleoperated task?

Telemanipulators are typically used in hostile *environments*. In, for example, nuclear [33], space [4] and (deep-)sea applications, the human is preferably on a remote location with respect to the manipulated objects. Telemanipulators used in (minimal invasive) surgery have the ability to improve the precision and dexterity with which surgeons can position surgical instruments [12], for example by filtering hand tremor [24]. Some task will involve accurate positioning of components (e.g. assembly), other tasks will require a force to be applied along a complex trajectory (e.g. applying a torque with an open-ended spanner).

These two tasks are typical examples of the distinct parameter sets humans apply when execution different tasks. When accurately positioning, a human will act very stiff, such that overshoot of their own movements and external perturbations is avoided. When applying a torque, the human will roughly apply a force in the (radial) movement direction, while he will act compliant in the axial direction and the direction perpendicular to the movement plane. Hence, in force tasks, like the one in this example, humans act compliant [1]. The bolt-and-spanner task could not be executed if the human would act stiff. This is just one example of the dynamic task behaviour of humans. Situations of humans quickly adapting their proprioceptive feedback gain to changing environmental conditions have also been reported [23].

It is important to understand that for every different task, humans will tune their visual and haptic feedback differently. Hence, to properly assess task performance a fundamental task span needs to be defined, which describes the wide variety of tasks. Aliaga acknowledged this problem and proposed a set of four device parameters that are representable for the task span: free motion impedance, position tracking in free movement, force tracking in hard contact tasks and maximum transmittable impedance [3]. As this is a theoretical approach, human-in-the-loop performance is not an integral part of the analysis. Nonetheless, the set can be easily extrapolated to create a set of fundamental tasks, oriented to define task performance instead of device performance. Whether fusion remote handling, minimal invasive surgery or (deep-)sea or space operations are concerned, it is proposed that many tasks can be separated in the following four fundamental tasks:

1) *Free Air Movement Tasks (FAMT),* are the the tasks for which the external forces on the slave device are zero, with or without tool.
2) *Contact Transition Tasks (CTT),* are the transitional stages in between unconstrained movement tasks and environmental interaction. Adequately perceiving contact establishment is critical to verifying spatial position and orientation of objects.
3) *Constrained Position Tasks (CPT),* are tasks in which one or more degrees of freedom are constrained. This can either appear by mechanical constraints (e.g. coaxial sliding of pipes) or a constrained distance to an object (e.g. welding).
4) *Force Tasks (FT),* are tasks in which forces (or torques) in or or more directions are controlled. A lot of material processing operations are force tasks, like polishing a pipe with a cloth or sanding a surface. Self-evident examples from surgical applications are cutting and sewing in human tissue.

Previous work on the assessment of task performance with multi-modal feedback showed that for a peg-in-hole task, low-frequency haptic feedback reduces impact forces, but does not influence environmental contact forces or task completion time [10]. High-frequency haptic feedback on the other hand does not improve task performance, but in combination with low-frequency feedback, it reduces mental workload and increases subjective task performance [10]. For material identification tasks, it has been shown that high-frequency information improves performance [29] [30]. However, these results are only applicable for specific fundamental tasks.

The main objective of this research is to understand how quality of haptic feedback influences human-in-the-loop task performance. Subjects are presented with four haptic feedback conditions that address the complete haptic feedback spectrum, namely no haptic feedback (Numb), low-frequency haptic feedback (LF), combined low- and high-frequency haptic feedback (LFHF) and a manually controlled task (HandsOn). Four generalized fundamental tasks have been defined and identified in a single bolt-and-spanner task. The main question of this research is: To what extend does the quality of haptic feedback contribute to human-in-the-loop task performance, using a telemanipulator as a tool?

Governed by Brook's findings, it is expected that Free Air Movement Tasks will not benefit from any form of haptic feedback. Contact Transition Tasks however, yield compressive stresses and high-frequent vibrations on impact and therefore both low-frequency and high-frequency haptic feedback are expected to increase task performance. For Constrained Position Tasks and Force Tasks, slow skin motions, compressive stresses and controlled slip will occur; it is expected that providing low-frequency feedback improves task performance. The hypotheses have been summarized in Table 1. Notice that the HandsOn task has been defined as the baseline scenario and therefore '0' stands for baseline task performance.

Table 1: Hypothesized task performance effect of haptic feedback quality per fundamental task

Task	Numb	LF	LFHF	HandsOn
Free Air Movement	0	0	0	0
Contact Transition	- -	-	0	0
Constrained Position	- -	-	-	0
Force	- -	-	-	0

2 Methods

2.1 Subjects

Ten subjects with an age range of 23 to 28 years participated in a human factors experiment. Three of the subjects did have previous experience with telemanipulators, while the other seven had no experience at all. The subjects were given a short introduction to the system and were, after some practice, asked to complete a telemanipulated bolt-and-spanner task. Each combination of conditions is repeated eight times. All subjects gave their informed consent.

2.2 Experimental Setup

The experiments were performed on a 3-DOF planar teleoperation system. The system consists of a double-rhomb force-redundant parallel master (see Fig. 2) and a serial slave device (see Fig. 3). The telemanipulator has an positional accuracy of 0.03 mm and runs on a Mathworks xPC Target real time operating system at 1 kHz. The minimum time delay between master and slave is estimated at 1.5 ms (1 ms measurement interval and 0.5 ms due to the zero-order hold of the analog output [8]). The design of this device is discussed in detail in [9].

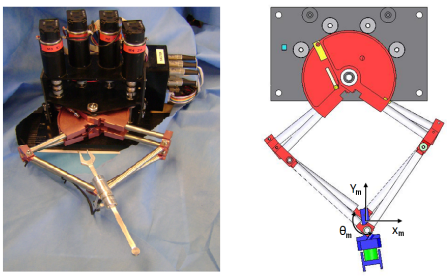


Figure 2: The master device

On both master and slave a spanner interface is constructed. The master includes a force sensor mount to measure the operator's input forces. The six degrees-of-freedom force sensor is of the type ATI Nano 17-SI 12. The slave has been stiffened to fit a generic (2-channel PD) PERR-controller and is instrumented with an accelerometer to measure the high-frequency contact forces. A more detailed description of the system can be found in Appendix A.

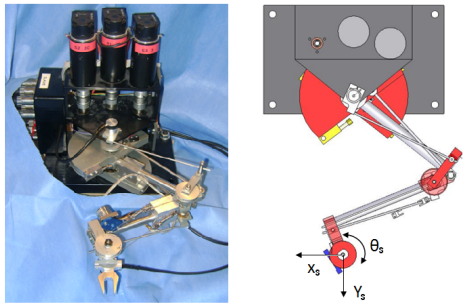


Figure 3: The slave device

The environment the telemanipulator is interacting with, consists out of a M6 bolt constructed on a aluminum plate. The implementation of the environment on the slave side is shown in Fig. 4. The moment on the bolt is created artificially with a spring, such that the tightening torque is constant. The tightening torques to overcome the static and dynamic friction are measured for twenty instances and are respectively estimated at 35.7 (2.0) Nmm and 31.6 (6.1) Nmm.

Using the HapticAnalysis package [19] the device performance and stability is evaluated. The HapticAnalysis package is based on the two-port network modeling framework [15]. Table 1 shows an overview of common analytical performance parameters, calculated for the spanners point of rotation.

In order to assess closed-loop stability, the impedances of the environment and the human operator were estimated. The environment is defined as a constant stiffness of $Ze = 1e10N/m$. The human arm impedance (Zh) for the specific bolt-and-spanner posture is estimated in the frequency domain using a closed-loop frequency domain identification procedure [23]. The impedance is measured for a Position Task (PT) and a Relax Task (RT), while the parameters for a Force Task (FT) are estimated based on previous experience. Assuming a zero reflexive gain for Relax Tasks, the human operator is able to decrease its stiffness for Force Tasks (k_{FT}) by tuning its reflexive feedback gains with the same magnitude,

Table 2: Numerical Performance Metrics

Analytic Performance Metric	Value
m-to-s velocity tracking	2.5
s-to-m velocity tracking	2.4
s-to-m force tracking	2.1
m-to-s force tracking	2.4
m-to-s velocity bandwidth	42.5
s-to-m velocity bandwidth	Inf
s-to-m force bandwidth	42.5
m-to-s force bandwidth	Inf
scaling product	3.4
free air m	0.14
free air b	11
free air k	0.027
stiff contact m	0.23
stiff contact b	4.5
stiff contact k	400
transparency	68
Zwidth	31

but opposite to the stiffness increase for Position Tasks [1] [34]: $k_{FT} \approx k_{RT} - (k_{PT} - k_{RT})$. The mass and damping of the force task are approximately equal to the mass and damping of the relax task, as the operator does not change posture and thus passive properties do not change. The results for the estimated mass (m), damping (b) and stiffness (k) are shown in Table 3.

Table 3: Operator Impedance Parameters

Property	PT_{mean} (std)	RT_{mean} (std)	$FT_{estimate}$
m [kg]	2.5 (0.1)	2.6 (0.4)	2.6
b [Ns/m]	38 (1)	28 (3)	28
k [N/m]	538 (130)	377 (45)	216

According to the Passivity [18] and Absolute Stability [22] criteria, the controller is stable. Furthermore, the controller suffices the Closed Loop Stability [16] criteria for all environmental and operator impedances as described in Table 3.

2.3 Task Description

The subjects are asked to complete a modified bolt and spanner task, starting at point A and moving to subsequently B and C (see Fig. 4). A, B and C are respectively located at (x y θ) = (0 0 0), (-0.06m 0.08m 0°) and (0.06m 0.08m -60°). From C, the subjects have the objective to slide the spanner over the bolt, which is located at D = (x,y) = (0m 0.12m). Finally a 80 degree stroke has to be made. Reference points and angles are all visually indicated.

This bolt-and-spanner task contains the four fundamental subtask. *Unconstrained Movement* is performed in moving from A to B to C (see Fig. 5). *Contact Transition* is performed in moving from C to the bolt and is considered completed when the contact evolves in a sliding motion. The sliding motion over the bolt is considered a *Constrained Position Task*, while applying a torque to turn the bolt is a *Force Task*. Fig. 4 shows the implementation on the experimental setup.

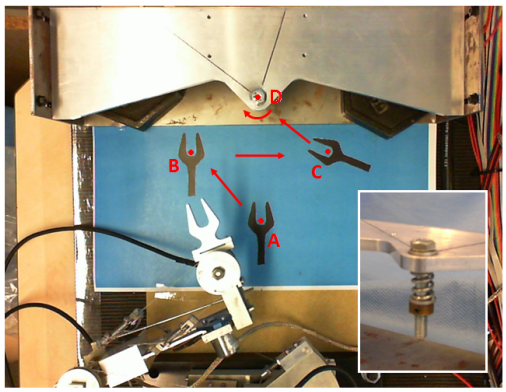


Figure 4: Implementation of the bolt-and-spanner task, slave side (scale approx. 1:10)

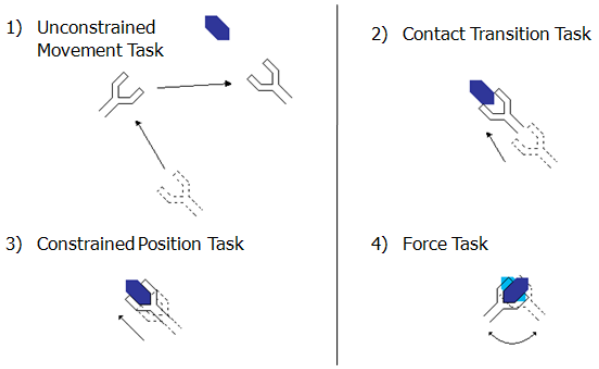


Figure 5: The four fundamental subtasks as identified in a bolt-and-spanner task

2.4 Experiment Design

The following experimental conditions are applied:

- Haptic Feedback
- Visual Feedback
- Task Instruction

2.4.1 Haptic Feedback

The subjects were presented with four levels of haptic feedback:

1) *Numb*, no force feedback is provided at al. Effectively this is achieved by setting the PERR slave-to-master PD-gains to zero.

2) *LF*, low-frequency haptic feedback. A classic PERR-controller provides the LF haptic feedback. From Table 1 is obtained that the s-to-m force bandwidth is equal to 42.5 rad/s. Hence, the bandwidth of the force signal presented to the human operator is equal to about 7 Hz.
3) *LFHF*, besides low-frequency haptic feedback a high-frequency signal is presented to the human operator. For this purpose an accelerometer is mounted on the slave, which measures the contact accelerations from the environment. The signal is filtered with a 4th order Butterworth high-pass filter with a cut-off frequency of 30 Hz. If the accelerometer is excited above the tuned threshold of 0.015 m/s^2, a signal is generated on the master side. This signal contains a 50 Hz and 100 Hz sinusoid, displayed for the duration of one 50 Hz period (20ms), and is superimposed on the PERR-signal. The amplitude of the signal is proportional to the impact acceleration. The gain on the high-frequency signal is hand-tuned to provide an intuitive feeling.
4) *HandsOn*, the manual equivalent of the bolt-and-spanner task, with a force feedback bandwidth in the order of thousands of Hz. The environment is relocated to the master side and the task is executed under equal conditions. This allows for reference comparison of the task performance in teleoperated conditions.

2.4.2 *Visual Feedback*

While the visual feedback channel is already heavily stressed, suboptimal camera views are often encountered in teleoperations (e.g. the manipulated object is in the line of sight of the contact plane). Therefore two opposites of visual feedback are tested. Either a perpendicular view of the scene is provided *(V)*, or no visual feedback is provided at all such that the subject has to perform the task blind *(B)*. The perpendicular view is displayed on a computer screen next to the setup using a Microsoft Lifecam Cinema webcam which is position above the experimental setup. Fig. 4 shows the view the subject is provided with.

The subjects are not asked to perform the task blind in the Numb condition. Furthermore, if the task is to be completed blind, the Free Air Movement Task is not performed and the subject will start its trial at point C (see Fig. 4).

2.4.3 *Task Instruction*

When performing a task, subjects have a natural preference for a certain control strategy. Often, these strategies are trade-offs between force magnitude, positional accuracy and/or time. Hence, the subjects are given two opposing task instructions:

1) *Accurate (A)*, to evoke a tasks execution optimized towards low forces and positional accuracy, or
2) *Fast (F)*, to evoke a task execution optimized towards time

2.5 Data Acquisition & Task Performance Metrics

The position encoder signals of the master and slave device are recorded with am accuracy of 0.03 mm. On the master side a force sensor is placed, which measures the contact force between the human operator and master device in three degrees-of-freedom. On the slave side, the accelerometer signal is recorded and an estimate of the slave forces is made based on the series-elastic principle [38]. All data is recorded at 1 kHz. Fig. 6 shows a typical example of a subject completing the bolt-and-spanner-task. The Free Air Movement Task is represented by the blue colored trajectory, the Contact Transition Task is colored red, while the Constrained Position Task and Force Task are respectively colored cyan and green.

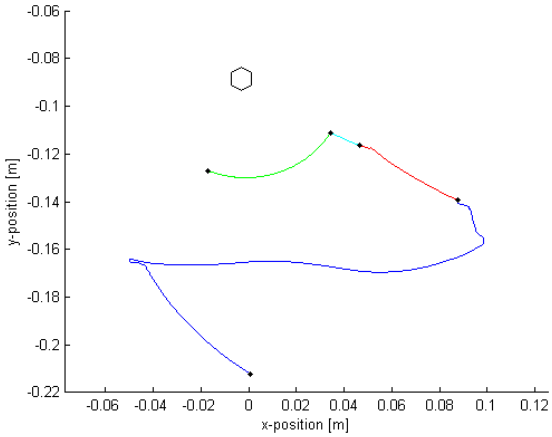


Figure 6: Trajectory plot of the manipulator endpoint position, showing the four fundamental subtasks

Based on the recorded data, a set of task performance metrics has been evaluated. The set can be

separated in a group that describes task performance in terms of environmental parameters:

t_{ttc} Time-to-complete, the time it requires for the subject to complete the task.

x_{npd} Nominal Path Deviation, the deviation from the nominal path.

$F_{c,a}$ Average Contact Forces, the estimated average interaction force with the environment.

$F_{c,m}$ Maximum Contact Force, the estimated maximum interaction force with the environment.

And a group describing task performance in terms of control effort:

$F_{i,a}$ Average Input Force, average interaction force between human operator and master.

$F_{i,m}$ Maximum Input Force, maximum interaction force between human operator and master.

W_{ph} Physical Workload, measured as the integral of the absolute input force over time.

n_{rr} Reversal Rate, the amount of steering corrections by the human operator. The reversal rate can be seen as a measure for the mental effort of the operator to control the device.

Furthermore, for each haptic feedback condition the following metrics are tracked:

- The self-reported workload is captured using the NASA Task Load Index (NASA-TLX) on a scale from 0 to 100 [20]. A lower score represents a lower mental workload.
- The subjects are asked to grade their own performance on a 20 point scale from 1 to 10, whereas 1 represents failure and 10 represents perfect.
- The failure rate is tracked. A trial is considered failed if the subject unintentionally does not complete (a part of) the predescribed task. Commonly occurring failures are situations where the subject is incapable of sliding the spanner over the bolt, the subject does not perceive that the spanner is not completely slid over the bolt while turning or when the spanner slips of the bolt at the end of the force task. The failure rate is expressed as a percentage.

2.6 Data Analysis

The comparison of experimental conditions is made on basis of populations, assuming a normal distribution. As there is a large variation between subjects, a multi-way Anova is considered most suitable for the statistical analysis. A p-level equal or below 0.05 is considered significant ($\alpha = 0.05$). In the Anova tables as presented in the results, '•••', '••' and '•' respectively denote the significance of $p \leq 0.001$, $p \leq 0.01$ and $p \leq 0.05$, while '-' denotes no significance.

The analysis is executed in stages: a first statistical analysis determines if the quality of the haptic information displayed influences performance. In a second stage the exact effects of the different haptic feedback levels are evaluated.

3 Results

Fig. 7 shows a bar chart of the entire time-to-complete for the bolt-and-spanner task, separated for the different subtasks. The data is compared in a three-way Anova, with factors representing haptic feedback, task instruction and subject variation. For all three factors a difference is found ($p < 0.001$). This means that the time-to-complete is influenced because of the large variation among subjects, the task instruction and the type of haptic feedback provided.

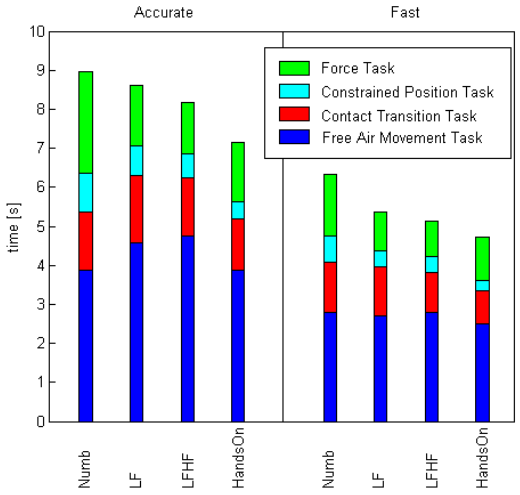


Figure 7: Time-To-Complete for the entire bolt-and-spanner task, separated for the four fundamental subtasks

With respect to haptic feedback, it shows that the HandsOn condition yields the shortest time-to-complete, whereas no difference with the LFHF condition can be found ($p = 0.0571$). Compared to LFHF, LF does not further decrease performance ($p = 0.36$). However, compared to HandsOn, the

LF condition shows an increased time-to-complete of approximately 19% ($p = 0.0237$). Providing haptic feedback decreases the time-to-complete with 10% to 30% for respectively the LF and HandsOn conditions ($p < 0.01$) if compared to the Numb condition.

The task performance metrics as defined in section 2.5 are separately presented for each of the fundamental tasks in the upcoming paragraphs, followed by the subjective measures and failure rate.

3.1 Free Air Movement Task

For the Free Air Movement Task, the experimental results are compared in a three-way Anova. The main factor results are shown in Table 4. The factors represent haptic feedback (HapticF), task instruction (TaskInst) and subject variation (SubjectV). Notice that as no contact takes place, the environmental contact forces ($F_{c,a}$ and $F_{c,m}$) are not evaluated.

Table 4: Anova results for the Free Air Movement Task. '• • •', '••' and '•' respectively denote the significance of $p \leq 0.001$, $p \leq 0.01$ and $p \leq 0.05$, while '-' denotes no significance.

	HapticF	TaskInst	SubjectV
t_{ttc}	-	• • •	••
x_{npd}	-	• • •	• • •
$F_{i,a}$	•	-	• • •
$F_{i,m}$	• • •	-	• • •
W_{ph}	••	• • •	• • •
n_{rr}	-	• • •	•

The third column, representing the subject variation, shows that the experiment is subjected to a large inter-variation. The second column shows that task instruction influences time-to-complete, nominal-path-deviation, operator workload and reversal rate, but does not influence the average and maximum input forces of the operator. In the first column it is shown that task performance in terms of average input forces ($F_{i,a}$), maximum input force ($F_{i,m}$) and physical workload (W_{ph}) is influenced by the type of haptic information displayed, while time-to-complete (t_{ttc}), nominal-path-deviation (x_{npd}) and reversal rate (n_{rr}) are not influenced.

Fig. 8 shows box plots of the time-to-complete for the Free Air Movement Task. The fast (F) tasks are clearly executed faster than the accurate (A) tasks, while no differences are observed between a complete spectrum of haptic feedback and completely none.

As illustrated in Fig. 9, the operator provides higher input forces when PERR-control is enabled

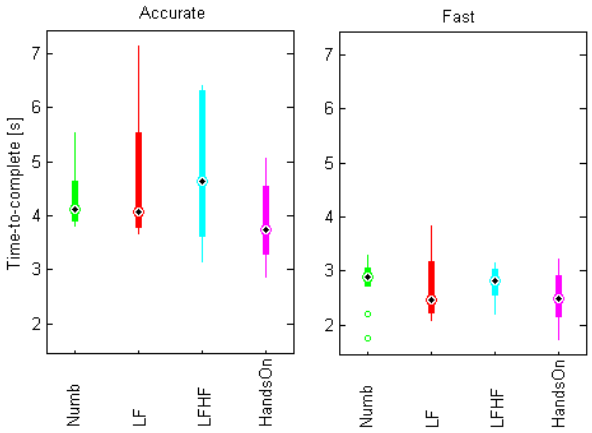


Figure 8: Free Air Movement Task - Time-To-Complete. There is no difference between the force feedback conditions.

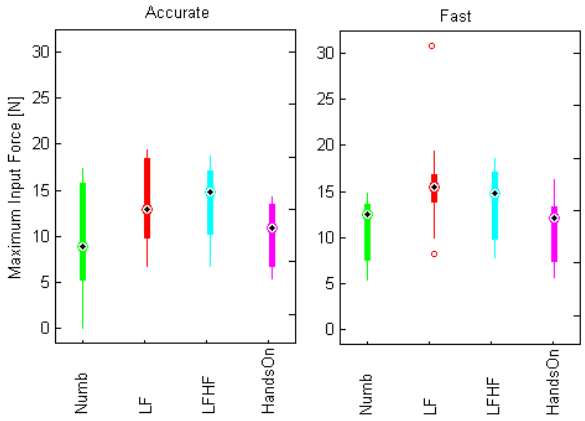


Figure 9: Free Air Movement Task - Maximum Operator Input Force. LF feedback yields a higher operator input force

(LF and LFHF). The same phenomenon is observed for the operators physical workload.

3.2 Contact Transition Task

The Contact Transition Task is performed with one addition condition, namely the visual feedback condition. Hence the experimental results for this subtasks are compared in a four-way Anova with factors representing haptic feedback (HapticF), visual feedback (VisualF) and task instruction (TaskInst) and subject variation (SubjectV). The results are shown in Table 5.

A first observation is that the subject inter-variance is large. Secondly, whether subjects are

Table 5: Anova results for the Contact Transition Task

	HapticF	VisualF	TaskInst	SubjectV
t_{ttc}	●	-	●	●●●
x_{npd}	●●	-	-	●●
$F_{c,a}$	●●●	-	-	●●
$F_{c,m}$	●●●	-	-	-
$F_{i,a}$	-	-	-	●●●
$F_{i,m}$	●●●	-	-	●●●
W_{ph}	-	●	-	●
n_{rr}	-	●	-	●●●

asked to complete the task accurately or fast, only the time-to-complete is affected measurably. Regarding visual or blind task performance it shows that workload, either measured physically or as the reversal rate, is reduced when providing visual feedback. Finally, the task performance depends on haptic feedback for time-to-complete, nominal path deviation, average and maximum contact force, maximum input force, while physical workload and reversal rate are not influenced.

Fig.10 shows that the time-to-complete is higher for blind (B) performed tasks compared to task with visual feedback (V), but only when the task is performed accurately.

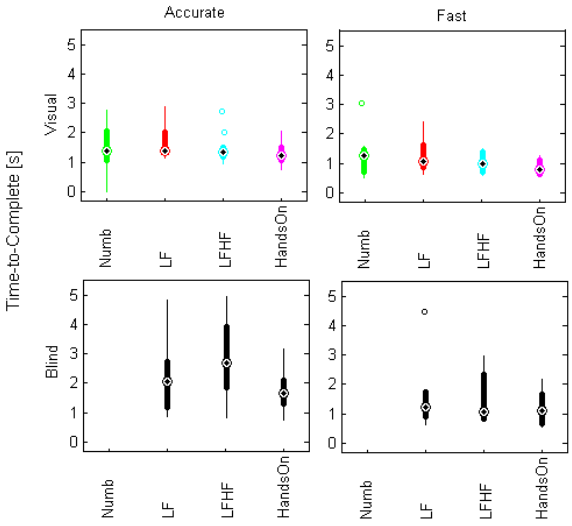


Figure 10: Contact Transition Task - Time-To-Complete. The time-to-complete is higher for blind performed tasks compared to task with visual feedback, but only when the task is performed accurately.

The difference in environmental contact forces and maximum input forces is caused by the direct contact of the HandsOn task with the environment, compared to the indirect contact of the teleoperated conditions ($p < 0.001$). Between the Numb, LF and LFHF conditions, no measurable difference is found.

3.3 Constrained Position Task

The results of the Constrained Position Task are compared using the same method as the Contact Transition Task, namely by using a four-way Anova. The results are shown in Table 6. As the path is predescribed the nominal-path-deviation is not evaluated.

Table 6: Anova results for the Constrained Position Task

	HapticF	VisualF	TaskInst	SubjectV
t_{ttc}	●●●	●●●	●	●●●
$F_{c,a}$	●●●	-	-	-
$F_{c,m}$	●●●	-	-	●
$F_{i,a}$	●●	-	-	●●●
$F_{i,m}$	●●●	-	-	●●●
W_{ph}	●●	●●●	-	●●
n_{rr}	●●●	●●●	●	●●●

From Table 6 it is obtained that task instruction influences time-to-complete and reversal rate. Providing visual feedback yields a reduction in time-to-complete, physical workload and reversal rate . The type of haptic information displayed influence the performance for all performance metrics.

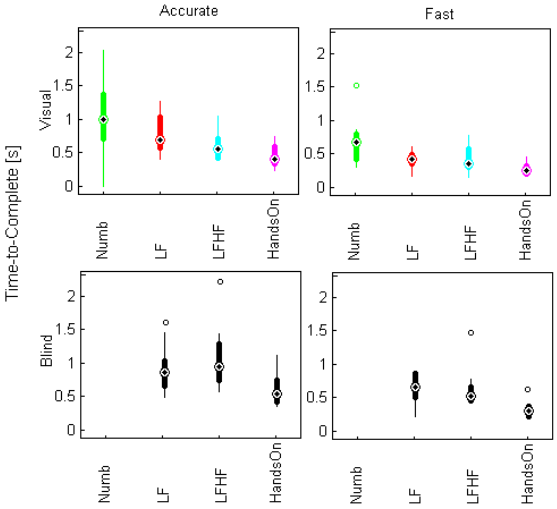


Figure 11: Constrained Position Task - Time-to-complete. Numb performance is inferior, LF and LFHF show no measurable difference, while HandsOn yields superior performance

As was expected, the task performance for the Numb condition is inferior with respect to the force

feedback conditions. LF and LFHF show no measurable difference, while HandsOn yields superior performance. This order of rank is illustrated by Fig. 11, which shows the time-to-complete. It is seen that LF feedback reduces time-to-complete compared to Numb ($p = 0.015$), while LFHF feedback does not yield an improvement compared to LF ($p = 0.915$). HandsOn show further reduction of time-to-complete ($p < 0.001$). The same classification of haptic feedback holds for maximum contact force, input forces, physical workload and reversal rate.

For the average contact force however no differences between Numb and LF can be found ($p = 0.389$), while LFHF does show an improvement in performance ($p = 0.018$) but only for the visual (V) condition. This is shown in in Fig. 12.

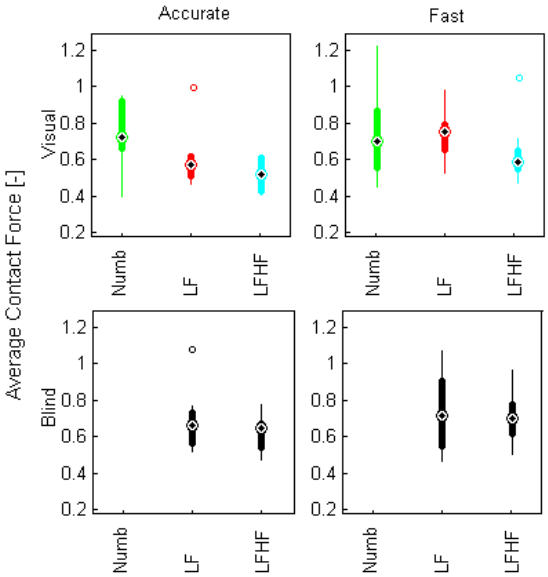


Figure 12: Constrained Position Task - Average Contact Force. LFHF shows a reduced environmental contact force, but only for the visual condition

3.4 Force Task

The four-way Anova results are shown in Table 7. The nominal-path-deviation is not evaluated. It is shown that task instruction and the visual feedback condition influence time-to-complete and reversal rate for the Force Task ($p < 0.001$). Haptic feedback influences all task performance metrics ($p \leq 0.043$), except the average operator input force.

The dominant pattern is that the Numb condition performs inferior to LF and LFHF, while no

Table 7: Anova results for the Force Task

	HapticF	VisualF	TaskInst	SubjectV
t_{ttc}	●●●	●●●	●●●	●
$F_{c,a}$	●●●	-	-	●
$F_{c,m}$	●●●	-	-	●●
$F_{i,a}$	-	-	-	●●●
$F_{i,m}$	●●●	-	-	●●●
W_{ph}	●	-	-	●●
n_{rr}	●●●	●●●	●●●	-

significant difference can be found between the latter two. This can be seen for the time-to-complete ($p < 0.001$), average contact force ($p < 0.001$), maximum contact force ($p < 0.001$), physical workload ($p < 0.001$) and reversal rate ($p = 0.001$). Fig. 13 illustrates this; the reversal rate is higher for the Numb condition compared to the other conditions.

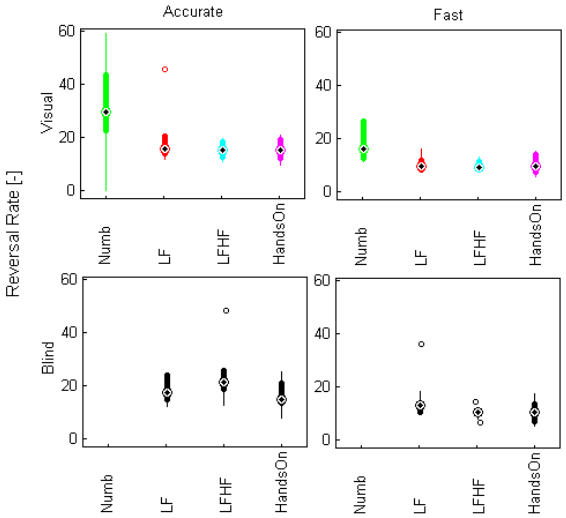


Figure 13: Force Task - Reversal Rate. The Numb condition shows inferior performance compared to the other conditions.

The operator input forces differ from this pattern. In the average operator input forces no differences show at all. For the maximum input force a difference is found between LF and LFHF ($p = 0.023$). This is shown in Fig. 14

3.5 Subjective Measures & Failure Rate

The TLX-scores for each of the haptic feedback conditions are compared. On average the mental workload of the Numb, LF, LFHF and HandsOn condition is respectively rated at 61, 51, 50 and 38. Seven out of ten subjects rated their mental workload equally or lower when more haptic feedback was

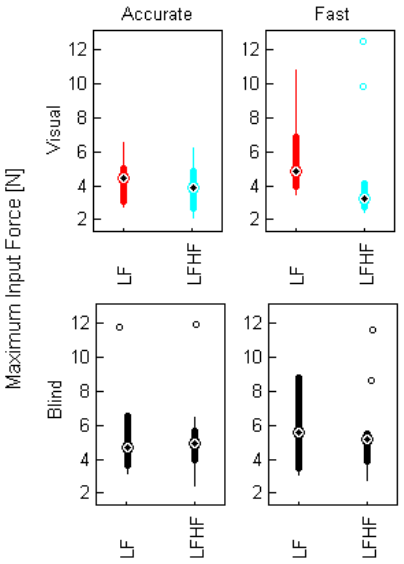


Figure 14: Force Task - Maximum Input Force. LFHF reduces the operator's maximum input force compared to LF.

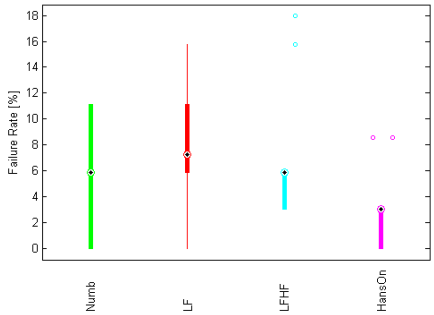


Figure 15: The Failure Rate per Haptic Feedback Condition. LFHF shows an increased failure rate compared to HandsOn. LF decreases the performance even more, while the failure rates for the Numb and LF conditions are approximately equal.

available. One subject found the HandsOn task more mentally demanding than LF teleoperated task. For two subjects, the LFHF condition was more mentally demanding than the LF condition. One of these subjects consciously noticed the high-frequency pulse and explicitly mentioned that he did not like the LFHF, as it felt unnatural.

The mean of the self-reported performance for the haptic feedback conditions was 5.6 for Numb, 6.8 for LF, 7.1 for LFHF and 7.9 for hands on. Eight out of ten subjects rate their performance higher when more haptic feedback is available. The two participants that reported a higher mental workload for LFHF compared to LF, also rated their performance lower.

The mean failure rates for the visual feedback conditions are 5.1%, 7.8%, 7.2% and 3.2% for respectively Numb, LF, LFHF and HandsOn. Fig. 15 shows a box plot of the failure rates. Compared to manual performance (HandsOn), LFHF increases the failure rate. Performing the task with only low-frequency feedback (LF) decreases the performance even more, while the failure rates for the Numb and LF conditions are approximately equal.

4 Discussion

The results show a classification of haptic feedback quality, illustrated by Fig. 7. The more haptic feedback is provided, the more the task performance is increased. However, the many significant differences found in the broad spectrum of haptic feedback that has been analyzed, only show a small magnitude of change. The largest improvement is a 54% reduction in time-to-complete for HandsOn compared to Numb. Even for the Force Task, which describes a complex path, the tasks could be performed fairly well for all conditions. Apparently, the quality of haptic feedback does not substantially affect the bolt-and-spanner task performance, not even if the task is performed manually and a complete spectrum of haptic feedback is provided. This is contrary to the expectations.

For the *Free Air Movement Task (FAMT)*, physical workload, time-to-complete and reversal rate are arguably the most important metrics. For the latter two, no difference was found when performed under different haptic feedback conditions. The physical workload is higher for the LF and LFHF conditions compared to the Numb and HandsOn condition. This can be explained by the fact that the PERR-controller, responsible for the low-frequency force feedback, partially reflects the slave's mass, effectively causing the operator to move more mass. This argues that a controller without haptic feedback is the most beneficial controller for situations where the operator has to perform workspace conveying free air movements, for example when the operator moves from his work piece to a tool tray. However, switching to Numb control can potentially create controller instabilities, as the total inertia decreases

with the decreased mass. It would be interesting to explore the dimensions of a controller without haptic feedback for tasks dedicated to free air movements, in combination with a FAMT specific mental workload assessment, taking the advantages and disadvantages of such a controller into account.

The combination of the *Contact Transition Task (CTT)* and the *Constrained Position Task (CPT)* shows many similarities with common peg-in-hole tasks; the operator is to align the relatively long spanner properly, whereupon it is slid over the bolt. Peg-in-hole tasks are surprisingly difficult to perform using robotic manipulators, partly due to physical limitations of the mechanical compliance of the robotic device, and partly due to the lack of a mating strategy [37]. For human-in-the-loop telemanipulation, the successfulness of a mating strategy is largely determined by the feedback quality, while the mechanical compliance is a property of the slave device. The construction of the manipulator used in this research, however, inherently has a high passive compliance ($\sim$93 N/rad for the slave's rotational degree of freedom). As a result, the slave design seems to facilitate peg-in-hole tasks [10] to an extent where a mating strategy, and thus haptic feedback quality, is redundant. On the one hand this shows that compliance allows for effective execution of peg-in-hole tasks, but on the other hand assessment of the effect of force feedback cues is hindered. This could be the reason why the results for the CTT and CPT somewhat contradict with previous findings. It has been shown that providing low-frequency haptic feedback improves the performance of peg-in-hole tasks [17] [10], while this study shows that CPTs benefit from low-frequency haptic feedback, but CTTs do not. Furthermore, it was expected that providing high-frequency feedback would improve the performance of the CTT, but this has not been observed.

Hence, due to the slave's compliance, it is still unclear whether low- and/or high-frequency haptic feedback affects the performance of the *Contact Transition Task (CTT)*. It would be interesting to study whether low- or high-frequency haptic feedback does influence CTT performance for a more complex task with frequent contact (e.g. multi-dof assembly), possibly in combination with a more sophisticated method of high-frequency feedback generation (e.g. by Kuchenbecker's High Frequency-Accelerations Matching [29]).

Providing low-frequency haptic feedback yields an improvement in six out of seven performance metrics for the *Constrained Position Task (CPT)*. However, there is one irregularity; the average environmental contact forces are equal for Numb and LF, while LFHF does show an improvement. This suggests high- frequency haptic feedback reduces the average contact force with the environment, but not low-frequency feedback. A possible explanation can be found in the definition of the CPT, which ends when the subjects starts rotating. When the spanner is completely slid over the bolt, an undesirable high-frequency pulse is generated within the CPT. This is strictly not part of the CPT.

As expected, the results show that the *Force Task (FT)* benefits from low-frequency haptic feedback. The lower operator input forces of LFHF compared to LF could be an artifact of the previous task, the CPT task. As stated, just before the FT starts an undesired HF pulse is generated. The reduction of input forces after such a high-frequency impulse, suggests that the operator acts more cautiously and reduce his input. Notice that this phenomenon is particularly observed when the operator moves fast and thus and larger pulse is excited. It could be that for more complex tasks, the addition of high-frequency feedback shows an improvement in other performance metrics (e.g. environmental contact forces or workload).

The importance of mental workload should not be underestimated. It has been shown that optimization of mental workload can reduce human error, improve system safety, increase productivity and increase operator satisfaction [44]. Hence, mental workload directly influences the operator's ability to perform tasks [43]. For teleoperations, it has been shown that the combination of low- and high-frequency feedback reduces mental workload and subjective task performance [10]. This effect is not observed in this study; the reduction in mental workload by providing LFHF feedback is marginal (a 2% increase) just as the subjective performance is only slightly increased (a 4% increase). An explaination might be that HF feedback consumes only a very small part of the task in the time domain. However, the majority of the failures (>80%) occurs during the execution of the CTT under the blind condition. This means that if the failure rates are observed, the majority of the differences between LF and LFHF occur exactly during this subtask. Hence, if decreasing mental workload reduces the failure rate [44], than

this reduction is attributable to the high-frequency part of the feedback.

Low-frequency haptic feedback improves performance for CPTs and FTs, but not for FAMTs and CTTs. High-frequency haptic feedback reduces the mental workload for CTTs, but does not improve any objective performance metric. Further improvement of the haptic feedback quality, by providing LFHF or HandsOn haptic feedback, does not substantially improve task performance. O'Malley and Goldfarb state, that while the lower boundary of Connected Telemanipulator System performance is limited by the tool (telemanipulator) capabilities, the upper boundary is limited by the human operator himself [35]. In other words, there are certain minimum requirements to the content of the information projected at the master side, but it is redundant to provide more information [8]. Apparently, providing low-frequency haptic feedback suffices the requirements to optimize the performance of the Connected Telemanipulator System. Hence, low-frequency haptic feedback out of all haptic feedback, is predominantly responsible for improvements in task performance for the experimental conditions studied.

Humans, in general, are biased to use their visual channel for task control [27]. The same holds for the subjects that participated in this experiment; they struggle to verify the correctness of their movements, shown the increase in time-to-complete (for CPT, FT), physical workload (for CTT, CPT) and reversal rate (for CTT, CPT, FT) in the absence of visual feedback. Their movements show characteristics of highly explorative tasks, with indecisiveness and unstructured motion control. Based on the task performance metrics, subjects are quite right to bias visual feedback; providing visual feedback most often yields a performance improvement for the CTT, CPT and FT. However, while the visual cue dominates, tasks are not necessarily performed better under visual feedback conditions. Operators are perfectly capable of performing certain tasks blind. If subjects are forced to perform the CTT under tight time constraints, the time-to-complete is equal for blind performed tasks compared to tasks with visual feedback, while the contact forces with the environment remain unchanged. This could indicate that, specifically for the fast executed CTT, the haptic channel dominates independent of the visual feedback conditions.

Although the bolt-and-spanner task is in principle a three degree of freedom (3 DoF) task, containing two translations and one rotation, performing a similar task in a 6 DoF environment would be considerably harder. Firstly, the quality of the camera views is often much better in a 3 DoF environment. A 3 DoF environment can be optimally visualized using only one perpendicular camera view. In a 6 DoF environment however, one would need at least three perpendicular camera views for optimal verification of the three translations and rotations. This is most often practically not realizable. Secondly, it is fundamentally much more difficult to track all 6 DoF simultaneously via the visual channel, than it is via the haptic channel. This is due to the fact that the human operator can focus on only one camera view, and thus only 3 DoF, at the same time. Hence, for the visual channel, going from 3 DoF to 6 DoF yields complications in (depth) perception and introduces virtual translations between the camera views and the physical situation; implications that do not occur when making the same transition for the haptic cue. Thus, the haptic channel is potentially much more promising to improve the performance of 6 DoF manipulation tasks, than it is for 3 DoF tasks.

Another important acknowledgement is that the repeating nature of experiments in general, and this bolt-and-spanner task specifically, could lead to a situation where subjects are able to include the telemanipulator dynamics and task instructions into their body scheme, to an extent where the focus shifts from feedback-based task control to feedforward based control. Although partially deliberately evoked through the blind condition, feedforward control has been observed in all subtasks. Even the Force Task, which describes a complex path, could be performed fairly well for all conditions, including blind control. Apparently, the subjects were well aware of the telemanipulator dynamics and could therefore include the telemanipulator dynamics into their body scheme fairly well for this particular task and setting. While learning effects should be avoided, one should be aware of the fact that feedforward control through extensive practice can hinder the analysis of small, isolated, segments of a task.

The realized low-frequency and high-frequency force feedback bandwidths are subjected to discussion. The (electro-)mechanical properties limit the bandwidth of the low-frequency information transfered from slave-to-master to approximately 7 Hz, while the high-frequency information is modeled as

an impulse containing a 50 Hz and a 100 Hz sinusoid. This leaves a gap in the force feedback bandwidth. For example, compressive stresses (∼10 Hz) and controlled slip (∼30 Hz) [5] cannot be sensed and displayed with the current setup. A 4-channel controller, with force sensors located on both master and slave side, could theoretically increase the bandwidth to infinity [8]. In practice this bandwidth will be limited by the (electro-)mechanical properties of the slave. Still, this type of control could multiply the currently realized slave-to-master force bandwidth. Moreover, the slave-to-master force bandwidth could be adjusted steplessly, using a filter. With such a setup, it would be interesting to study at what bandwidth Connected Telemanipulator System task performance saturates.

The fundamental tasks as defined in this study are mainly relevant for tasks in which mostly hard-hard contact occurs (e.g. maintenance in power plants or space). For telemanipulation tasks like those in the medical field, where a lot of soft tissue interaction is occurs, the implementation of the fundamental tasks and the high-frequency feedback require customization.

Manual control preserves the human operator's skills, attention, and its abilities to resolve unexpected situations [36]. Furthermore, manual control keeps the human operator responsible for all actions. Presently, society simply does not accept robotic devices to take over delicate manipulation tasks from the human like those encountered in surgery or nuclear maintenance. Exactly for those reasons, telemanipulation will find its purpose for many more decades, despite all the rapid developments in automation in general, and artificial intelligence in particular.

Possibly the most promising strategy to improve telemanipulated task performance, is to support the operator by providing additional, virtual feedback forces. These feedback forces translate environmental variables into haptic feedback signals [2]. An example that stands out in literature is shared control, first described by Steele and Gillespie [42]. Shared control offers the ability to highlight preferred control strategies to the operator in order to enhance performance or safety in dynamic work-environments [2]. Hence, shared control directly supports the operator in his task, by looking at the task requirements. As a result, it improves performance of the system as a whole. Improving teleoperator performance however, only improves a single element in the Connected Telemanipulator System, but only marginally improves the overall system performance. Only by combining state-of-the-art developments in control techniques and thorough understanding of human motion control with sophisticated support strategies like shared control, the task performance of the Connected Telemanipulation System can be optimized.

5 Conclusions

The influence of haptic feedback quality on task performance is assessed for a bolt-and-spanner task, containing four generalized fundamental tasks, namely i) Free Air Movement Tasks, ii) Contact Transition Task, iii) Constrained Position Tasks and iv) Force Tasks.

- Improving the quality of haptic feedback yields only a marginal improvement in task performance, even if a complete spectrum of haptic feedback is provided.
- Task performance is predominantly improved by providing low-frequency haptic feedback; low-frequency haptic feedback improves the performance of Constrained Position as well as Force Tasks.
    - Constrained Movement Tasks and Force Tasks show a reduction in time-to-complete, average environmental contact force, physical workload and reversal rate. Furthermore, the operator input force is reduced for Constrained Movement Tasks and the maximum contact force is reduced for Force Tasks.
- Free Air Movement Tasks are best performed without any form of haptic feedback
- The combination of low- and high-frequency feedback reduces the mental workload for Contact Transition Tasks
- Providing visual feedback generally yields a performance increase. However, operators are perfectly capable of performing Contact Transition Tasks without visual feedback.

References

[1] D.A. Abbink, *"Task Instruction: the largest Influence on Human Operator Motion Control Dynamics,"* Second Joint EuroHaptics Conf. and Symp. on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07), 2007

[2] D.A. Abbink and M. Mulder, *"Exploring the Dimensions of Haptic Feedback Support in Manual Control,"* Journal of Computing and Information Science in Engineering, Vol. 9, Issue 1, March 2009

[3] Aliaga et al., *"Experimental Quantitative Comparison of Different Control Architectures for Master–Slave Teleoperation,"* IEEE Trans. on Control System Technology, No. 1, January 2004

[4] R.O. Ambrose et al., *"ROBONAUT: NASA's Space Humanoid,"* IEEE Intelligent Systems Journal, Vol. 15, No. 4, pp. 57-63, 2000

[5] T.L. Brooks, *"Telerobotic response requirements,"* Proc. the IEEE International Conference on Systems, Man, and Cybernetics, p. 113-120, 1990

[6] L. Cardinali et al., *"Tool-use induces morphological updating of the body schema,"* Current Biology, Vol. 19, Issue 12, pp. 478-479, June 2009

[7] G.A.V. Christiansson et al., *"The Low-Stiffness Teleoperator Slave - a Trade-off between Stability and Performance,"* Int. Journal of Robotics Research, Vol. 26, No. 3, pp. 287-299, 2007

[8] G.A.V. Christiansson, *"Hard Master Soft Slave Haptic Teleoperation,"* PhD Dissertation, Delft University of Technology, The Netherlands, 2007

[9] G.A.V. Christiansson, *"A novel 3-DOF Planar Haptic Teleoperation System,"* Proc. of IEEE WorldHaptics Conference, pp. 361-366, Tsukuba, Japan, 2007

[10] G.A.V. Christiansson, *"An Experimental Study of Haptic Feedback in a Teleoperated Assembly Task,"* Journal of Computing and Information Science in Engineering, Vol. 8, Issue 4, December 2008

[11] R.W. Daniel and P.R. McAree, *"Fundamental limits of performance for force reflecting teleoperation,"* The Int. Journal of Robotics Research, Vol. 17, No. 8, p. 811-830, 1998

[12] H. Das et al., *"A telerobot to extend the skills of microsurgeons,"* Technical Report from Jet Propulsion Lab., 1998

[13] J.V. Draper et al., *"Effects of force reflection on servomanipulator task performance,"* Proc. of the International Topical Meeting on Remote Systems and Robots in Hostile Environments, Pasco, WA, pp. 654-660, 1986

[14] J. Dudragne et al., *"A generalized bilateral control applied to masterslave manipulators,"* Proc. of the 20th ISIR, p. 435-442, 1989

[15] B. Hannaford, *"A Design Framework for Teleoperators with Kinesthetic Feedback,"* IEEE Trans. on Robotics and Automation, Vol. 5, No. 4, Aug 1989

[16] B. Hannaford, *"Stability and performance tradeoffs in bilateral teleoperation,"* IEEE Intern. Conf. on Robotics and Automation, 1989

[17] B. Hannaford et al., *"Performance evaluation of a six-axis generalized force-reflecting teleoperator,"* IEEE Trans. on Systems, Man and Cybernetics, Arizona, Vol. 21, No. 3, 1991

[18] B. Hannaford et al., *"Testing time domain passivity control of haptic enabled systems,"* Report, 2002

[19] *http://www.hapticanalysis.org,* a set of toolboxes for GNU/Maxima and Mathworks Matlab to analyze models of haptic teleoperators by G.A.V. Christiansson

[20] S.G. Hart and L.E. Staveland, *"Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research,"* Human Mental Workload (P. A. Hancock and N. Meshkati (Eds.)), 1988

[21] K. Hashtrudi-Zaad et al., *"On the use of local force feedback for transparent teleoperation,"* Proc. of the IEEE Int. Conf. on Robotics and Automation, Vol. 3, May 1999

[22] S.S. Haykin, *"Active Network Theory,"* Addison-Wesley, 1970

[23] F.C.T. van der Helm et al., *"Identification of intrinsic and reflexive components of human arm dynamics during postural control,"* Journal of Neuroscience Methods, 2002

[24] I.W. Hunter et al., *"A teleoperated Microsurgical Robot and Associated Virtual Environment for Eye Surgery,"* Presence, Vol. 2, No. 4, p. 265-280, 1993

[25] P. Jenmalm and R.S. Johansson, *"Visual and Somatosensory Information about Object Shape Control Manipulative Fingertip Forces,"* The Jounal of Neuroscience, Vol. 17, No. 11, 1997

[26] R.L. Klatzky, et al., *"There's more to touch than meets the eye: The salience of object attributes for haptics with and without vision,"* Journal of Experimental Psychology. General, 116, p. 356–369, 1987

[27] R.M. Klein, *"Attention and visual dominance: A chronometric analysis,"* Journal of Experimental Psychology, Human Perception Performance 3, p. 365–378, 1977

[28] Mathias Kölsch et al., *"The Postural Comfort Zone for Reaching Gestures,"* Technical Report, University of California, 2003

[29] K.J. Kuchenbecker, *"Characterizing and controlling the high-frequency dynamics of haptic interfaces,"* PhD Dissertation, Stanford University, The Netherlands, 2006

[30] R.H. LaMotte, *"Tactual discrimination of softness,"* Journal of Neurophysiology, 2000

[31] D.A. Lawrence, *"Stability and Transparency in Bilateral Teleoperation,"* Proc. of the 31st Conf. on Position and Control, 1993

[32] M. Massimo and T. Sheridan, *"Variable force and visual feedback effects on teleoperator man-machine performance,"* Proc. of NASA Conf. on Space Robotics, Publication No. JPL-87-7, Vol. 1, pp. 89-98, 1989

[33] S. Mills et al., *"Remote Handling of JET In-Torus Components - A Practical Experience,"* Proc. of the Symp. on Fusion Engineering, 2000

[34] W. Mugge et al., *"A rigorous model of reflex function indicates that position and force feedback are flexibly tuned to position and force tasks,"* Experimental Brain Research, 2010

[35] M.K. O'Malley and M. Goldfarb, *"The effect of virtual surface stiffness on the haptic perception of detail,"* IEEE/ASME Trans. on Mechatronics, Vol. 9, Issue 2, pp. 448-454, 2004

[36] M.K. O'Malley, A. Gupta, M. Gen, and Y. Li, *"Shared Control in Haptic Systems for Performance Enhancement and Training,"* ASME Journal of Dynamic Systems Measurement and Control, Vol. 128, pp. 75–85, 2006

[37] G.C. Pettinaro, *"Behaviour-based peg-in-hole,"* Robotica, Vol. 17, Issue 2, pp. 189-201, 1999

[38] G.A. Pratt and M.M. Williamson, *"Series elastic actuators,"* IEEE Int. Conf. on IntelligentRobots and Systems, Vol. 1, pp. 399-406, 1995

[39] A.C. Rolfe et al., *"A perspective on fusion relevant remote handling techniques,"* Fusion Engineering and Design, Vol. 82, Issues 15-24, pp. 1917-1923, 2007

[40] T.B. Sheridan, *"Telerobotics,"* Automatica, Vol. 25, No. 4, 1989

[41] K.B. Shimoga, *"Finger force and touch feedback issues in dexterous telemanipulation,"* Proc. of the Conf. on Intelligent Robotic Systems for Space Exploration, 1992

[42] M. Steele and R.B. Gillespie, *"Shared Control Between Human and Machine: Using a Haptic Steering Wheel to Aid in Land Vehicle Guidance,"* 45th Annual Meeting of the Human Factors and Ergonomics Society, pp. 1671–1675, 2001

[43] H.S. Vitense, *"Multimodal feedback: an assessment of performance and mental workload,"* Ergonomics, Volume 46, Issue 1-3, pages 68 - 87, 2003

[44] B. Xie et al., *"Prediction of mental workload in single and multiple tasks environments,"* Int. Journal of Conginitive Ergonomics, 4, pp. 213-242, 2000

Appendices belonging to the Master's thesis:

Improving the Quality of Haptic Feedback Yields Only Marginal Improvements in Teleoperated Task Performance

J.G.W. Wildenbeest

Contents

Appendix A: Experimental Setup 5

A1 Hardware Design 8

A1.1 Hardware Design - Master 9

A1.2 Hardware Design - Slave 18

A1.3 Hardware Design - Environment 20

A2 Kinematics 24

A2.1 Master Kinematics & Jacobian 24

A2.2 Slave Kinematics & Jacobian 37

A3 Controllers 40

A3.1 Low-Frequency Force Feedback 42

A3.2 Improved Low-Frequency Force Feedback (Series-Elastic) 47

A3.3 High-Frequency Force Feedback 54

A3.4 Controller Realization 57

A4 Performance Characteristics 59

A5 A short how to for the Munin 66

Appendix B: Human Identification Experiment 69

Appendix C: CTS Simulations 79

C1 Task Model 80

C2 Human Operator Model 80

C2.1 Human Cortex Model 80

C2.2 Human Neuromuscular Model 87

C3 Contact Dynamics Model 88

C4 Telemanipulator Model 91

C5 Environmental Model 92

C6 Preliminary Results & Future Work 92

Appendix D: Human Factors Experiment 97

D1 Task Completion Criteria 97

D2 Description Performance Metrics 101

D3 Data Management 104

D4 Results 105

References 135

Appendix A: Experimental Setup

The experimental setup is based on the 3-DOF telemanipulation system, referred to as '*Munin*'. The design of manipulator is described in Christiansson, 2007b. The Munin consists of a stiff double-rhomb force-redundant parallel master (see Fig. A1) and a serial slave device (see Fig. A2).

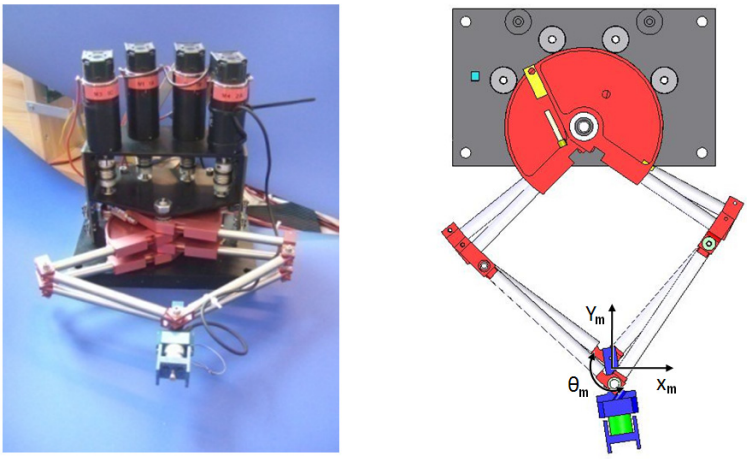


Figure A1: The Munin master by Christiansson

The Munin was designed to evaluate the hard master - soft slave haptic telemanipulation concept, devised by Dr. Göran Christiansson (Christiansson, 2007a). Hard-soft haptic telemanipulation aims to improve task performance by adapting telemanipulator design to the inherent properties of both the human operator, which controls the master, and the environment in which the task is executed.

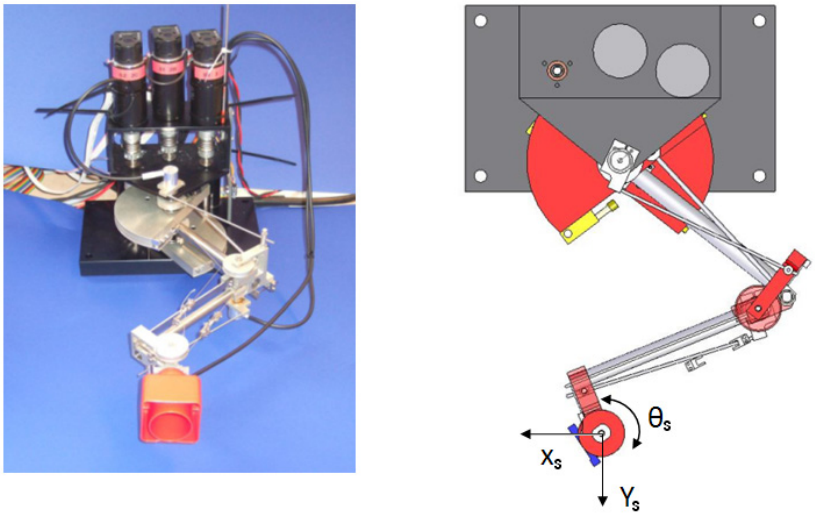


Figure A2: The Munin slave by Christiansson

Hard-soft telemanipulation reflects on the human's inherently asymmetric information flows, having a outgoing information stream (manipulation) with a frequency content up to < 10 Hz, while the

incoming information stream (sensing) can have a content up to several hundreds of Hz (Brooks, 1990; Christiansson, 2007a; Shimoga, 1992). This is illustrated in Fig. A3. In hard-soft telemanipulation, the master is constructed as a high-bandwidth, stiff device, such that high stiffnesses and haptic information with a high frequency content can be displayed. The slave on the other hand, is a low-bandwidth, intrinsically compliant device which improves telemanipulator stability. Hence, hard-soft telemanipulation tries to solve one of the fundamental problems of telemanipulation; the trade-off between stability, complexity and performance (Hannaford, 1989b). However, the inherent compliance of the slave device in hard-soft telemanipulation adds an additional dimension to the complexity of the control, but more important, it adds a design concept, from which the implications on the operator's control behaviour are not fully understood. Hence, if one wants to assess properties not specifically related to hard-soft telemanipulation, it is required to generalize the manipulator. This means the control algorithm has to change to generic control algorithms like position-error control, position-force control or 4-channel control (Aliaga et al., 2004), and an optimization toward 'transparency' has to be made; an optimization of the telemanipulator towards a virtual rod with infinite small mass, and infinite large stiffness (Drudgane et al., 1989).

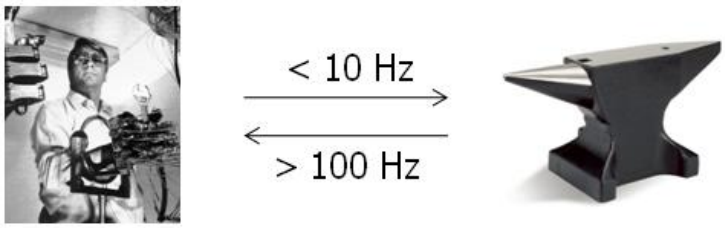


Figure A3: The Humans Asymmetric Bilateral Information Flow

The Munin was realized in the year 2007. The device was first used in an experiment addressing operator cues in a telemanipulated Lego assembly task (Christiansson, 2008). For the past three years, the device has been used for demonstration purposes and some minor projects. For example, the control algorithm has been refurbished and a construction has been designed to increase the slave stiffness to allow for more generic master-slave research.

To fit the specific requirements of the experiments described in this thesis, the Munin has undergone significant hard- and software (control) changes. The main changes are listed below. Notice that changes to the hardware are denoted as '▪…', while software changes are denoted as '• …'.

To increase telemanipulator performance:

- Recalculated master kinematics to increase performance in terms of power and accuracy (forward and inverse)
- Recalculated slave kinematics (forward and inverse)
- Redone the series-elastic slave force estimator, based on the renewed slave kinematics (control was successfully tested with control based on joint and motor encoders independently)

- A common D-action has been introduced next to the P-action of the control architecture, which is PERR-control. This allows for more smooth control/
- Torque limits implied by the software have been raised as the original limits resulted in awkward behavior.

Specifically for the slave's end-effector rotational (Θ_s):

- The stiffness of the slave's end-effector rotation (Θ_s) has been increased, by exchanging springs (from 1.22 N/mm to 6.67 N/mm)
- The Power Supply Unit (PSU) for the slave's end-effector rotation (Θ_s) has been replaced (the original PSU was the bottleneck to improve power output)
- The Quansar Q8 Analog Output Board output range has been changed for the Maxon LSC linear amplifier from [-5,5] to [-10,10].

To allow for execution of a bolt-and spanner task:

- A bolt and spanner interface has been designed for both master and slave, including a mount for the force sensor on the master side
- Changed Point Of Rotation (POR) to create a kinematical identical situation for master and slave for the spanner hardware

To allow for high-frequency force feedback:

- An accelerometer has been mounted on the master side
- The high-frequency force feedback information have been adjusted

These aspects will be described in the upcoming paragraphs.

A1 Hardware Design

In order to perform the task as assessed in this study, the bolt-and-spanner task, specific hardware to fit the Munin has been designed. This hardware consists out of three subdesigns: a spanner construction on the master side, a spanner head on the slave side and an environment (see Fig. A4). The CAD models and drawings are created using SolidWorks 2009 and can be found in the <* \CAD> folder. All, except the base of the environment (steel), have been manufactured from aluminum.

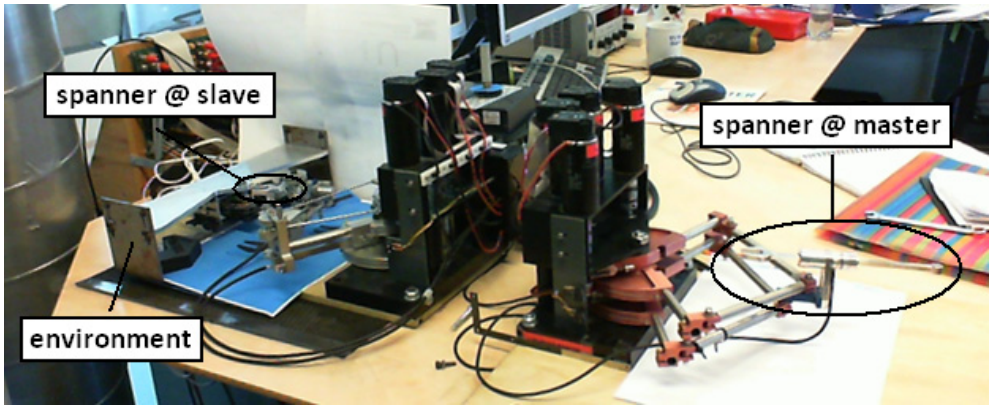


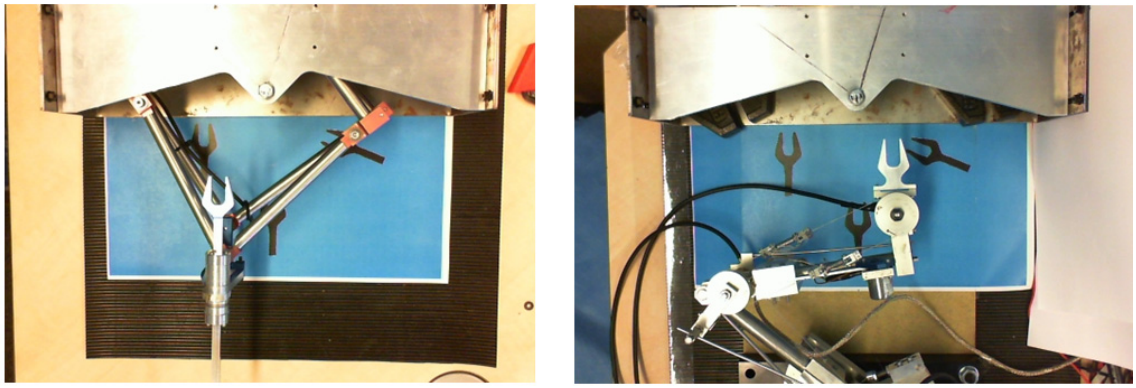
Figure A4: Hardware subdesigns

Requirements master & slave:

- Kinematic identical spanners for master and slave
- Bolt format: M6-compatible (spanner no. 10)
- Compatible with current master/slave interfaces
- Force sensor mount on master side
- Excessive long spanner heads to exaggerate the spanner-over-bolt sliding motion
- 'real' spanner look-a-likes (e.g. the bulb on the end of the grip which serves as reference)
- As light and stiff as reasonably possible

Requirements environment:

- Adjustable tightening torque of M6 bolt
- Adjustable height, to allow placement on both master and slave sides (see Fig. A5)



a) b)

Figure A5: Implementation of environment on a) master side, and b) slave side

A1.1 Hardware Design - Master

Besides a spanner and a grip, the hardware design for the master includes an interface to elevate the working plane above the parallelogram mechanisms. This allows to execute the bolt-and-spanner task on the master side. Furthermore, the design facilitates a mounting interface for the ATI Nano 17-SI 12 force sensor. Notice that, while it was foreseen to put the force sensor on the environmental side of the spanner, in the actual experiments the force sensor has been constructed on the human gripper side (to measure the human operator's input). To maintain a kinematical identical situation of master and slave, a 'dummy' cylinder replaced the force sensor. To ease manufacturing, the spanner has been subdivided in six parts. See Fig. A6 for a CAD impression of the master. Fig. A7 shows the realization.

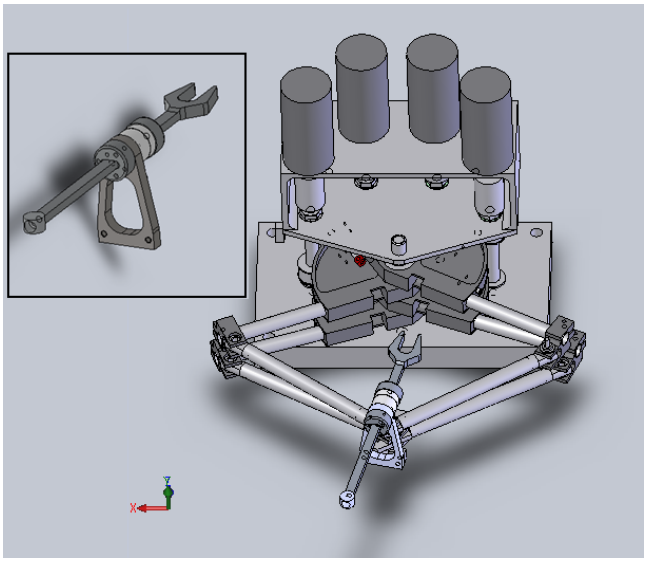


Figure A6: The master - CAD

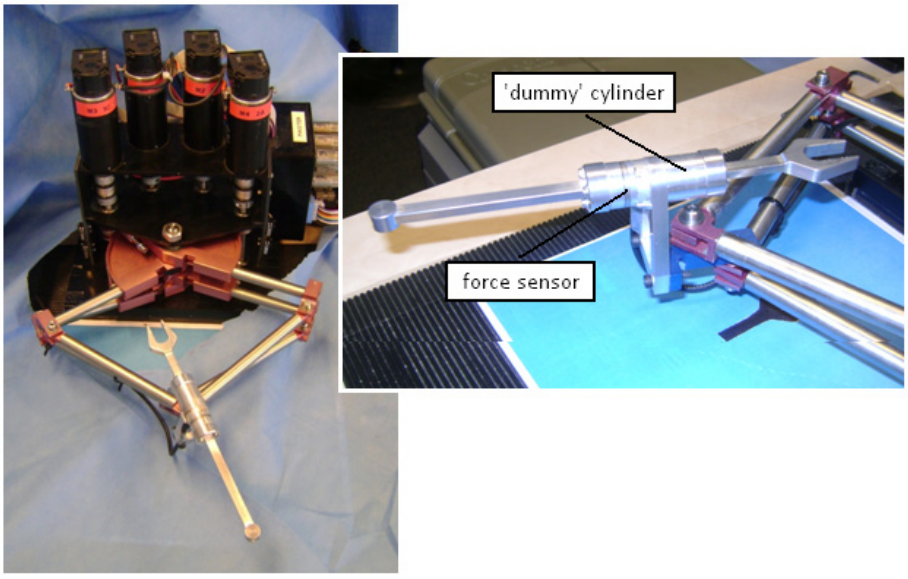
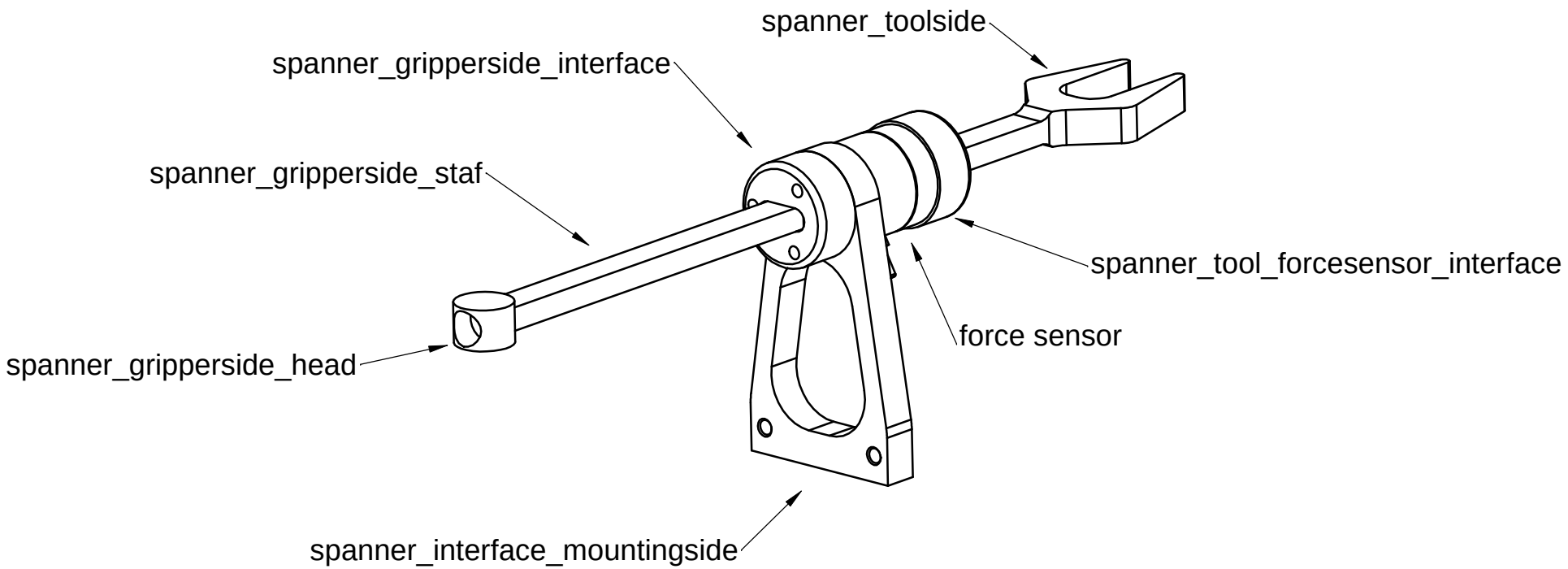
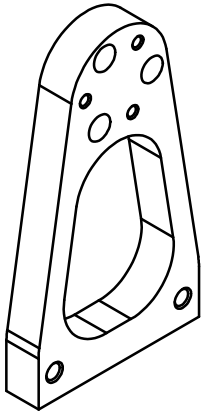


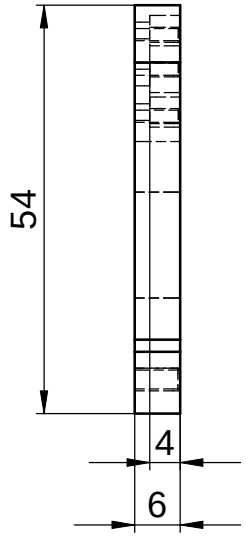
Figure A7: The master - realization



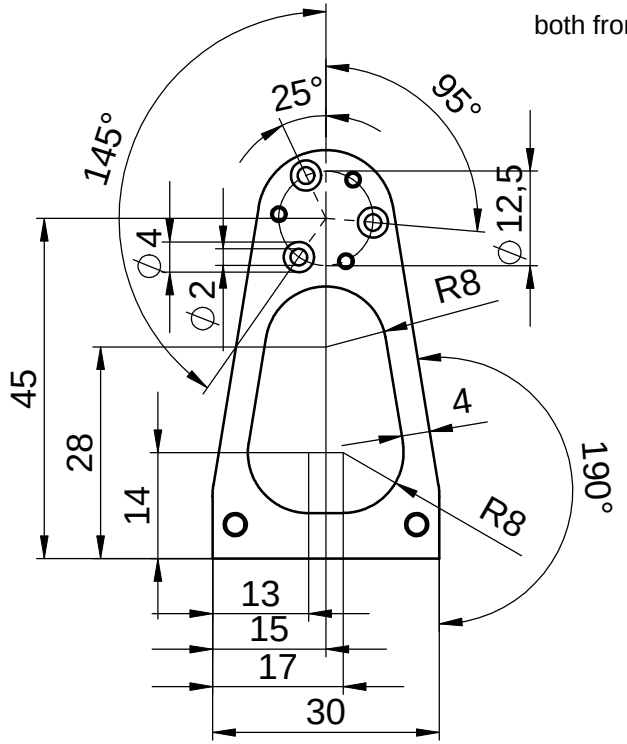
TU Delft	benaming spanner_master_side		datum 21-1-2010	maateenheid <<maateenheid>>
		schaal 1:2	groep <<groep>>	formaat A4 / tekeningnummer <<tekeningnummer>>
		getekend J. Wildenbeest		

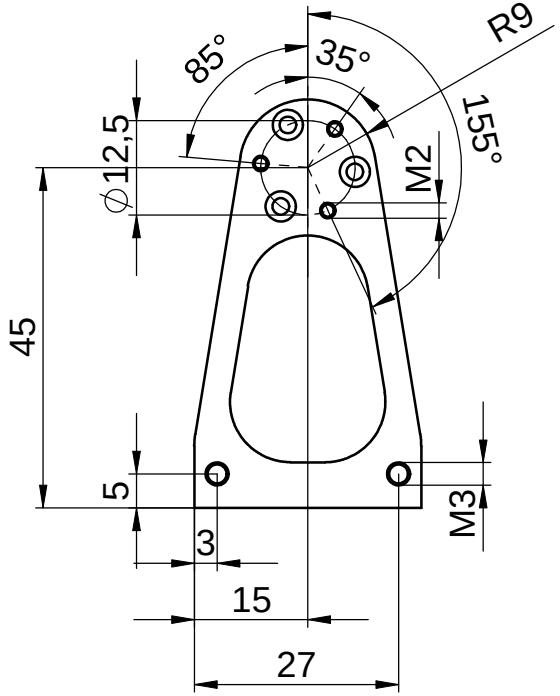
SolidWorks Student License
Academic Use Only





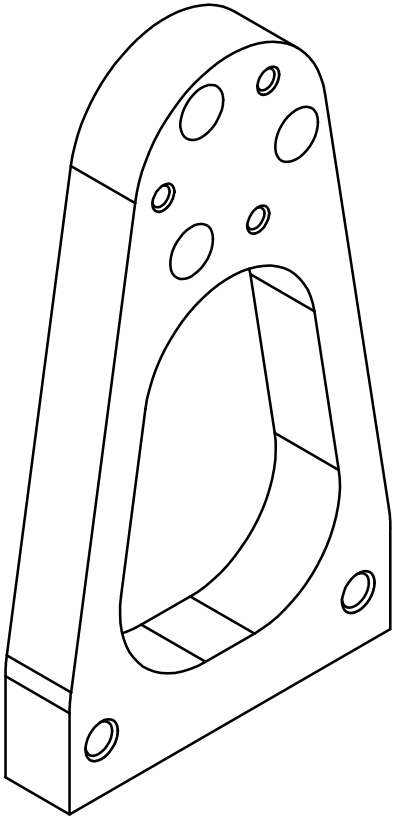
both frontviews!

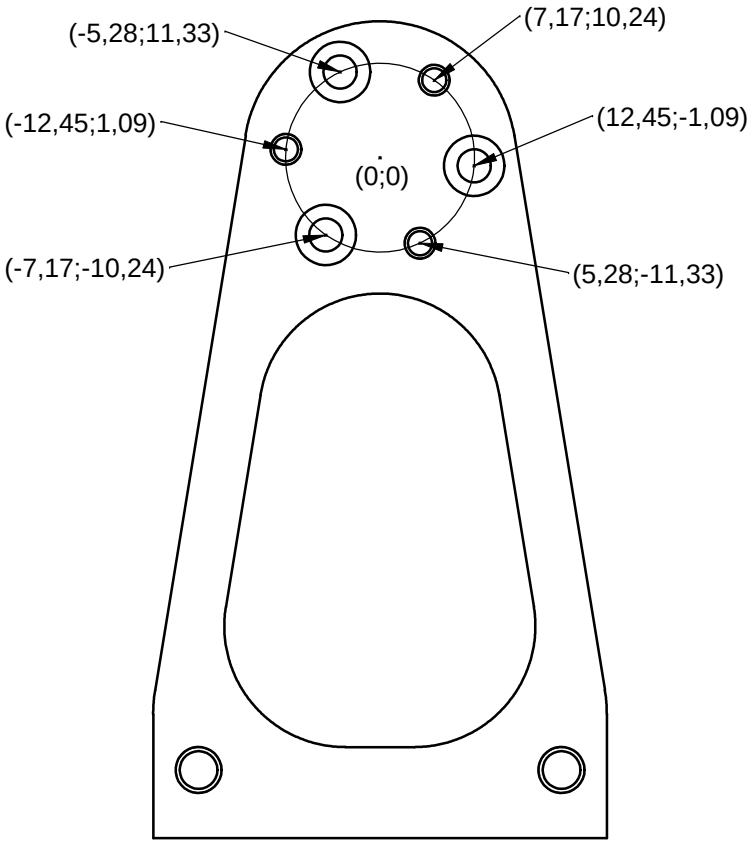




TU Delft	benaming: spanner_interface_mountingside		datum: 21-1-2010	maateenheid: mm
		schaal: 1:1	groep: <<groep>>	formaat: A4
	material: Aluminium	getekend: J. Wildenbeest		tekeningnummer: <<tekeningnummer>>

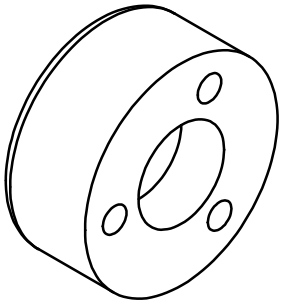
SolidWorks Student License
Academic Use Only

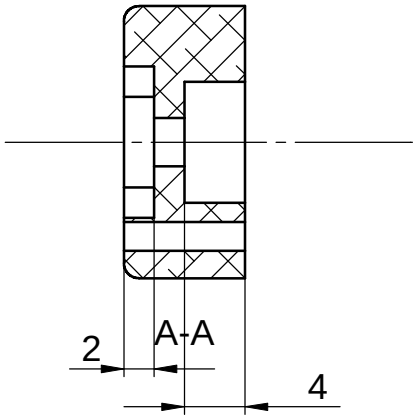


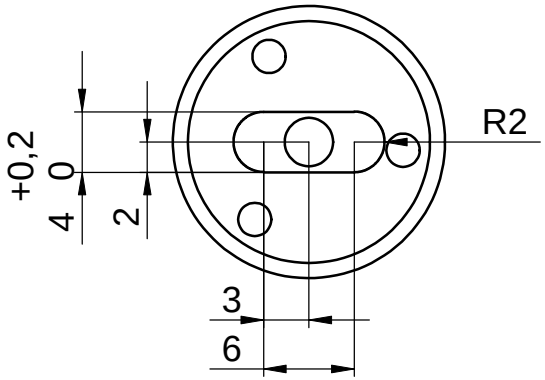


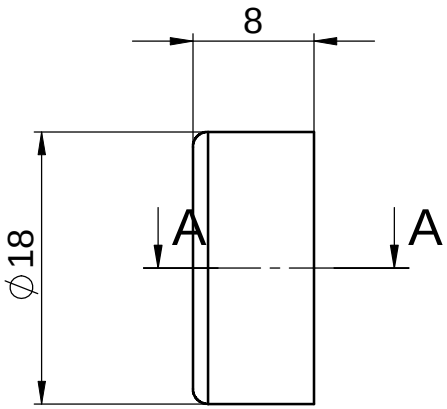
TU Delft	benaming spanner_interface_mountingside_cartesian	datum 21-1-2010	maateenheid <<maateenheid>>	
		schaal 1:1	groep <<groep>>	formaat A4	tekeningnummer <<tekeningnummer>>
		getekend <<naam>>		

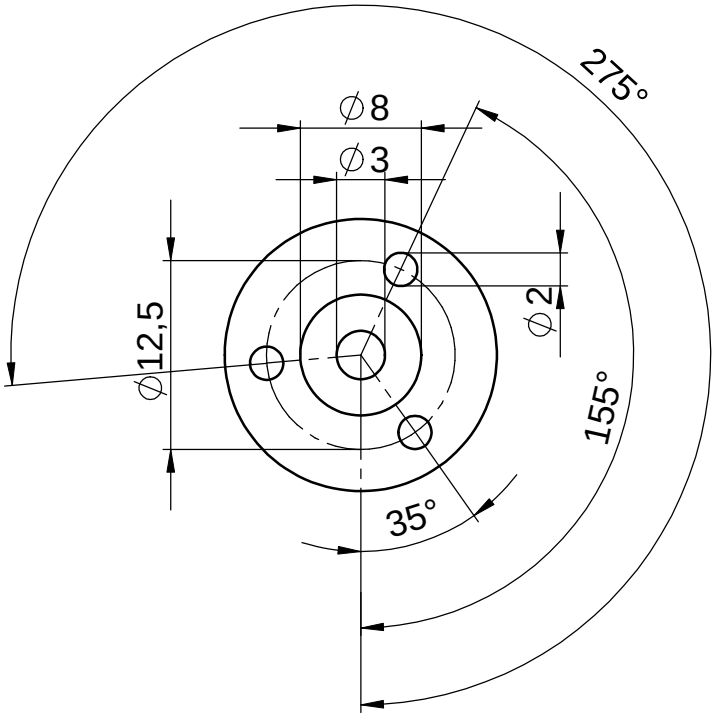
SolidWorks Student License
Academic Use Only





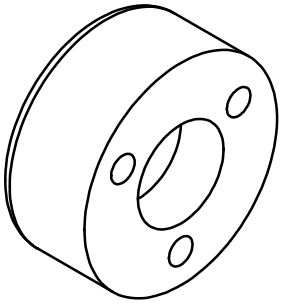


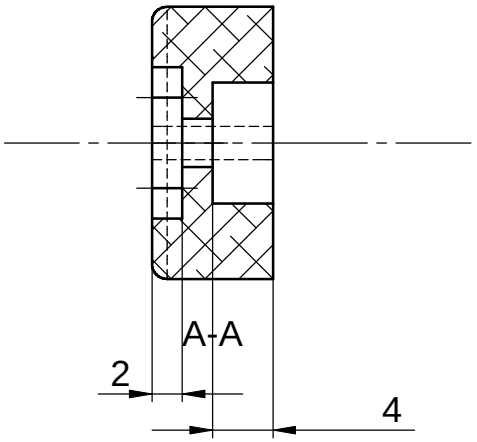


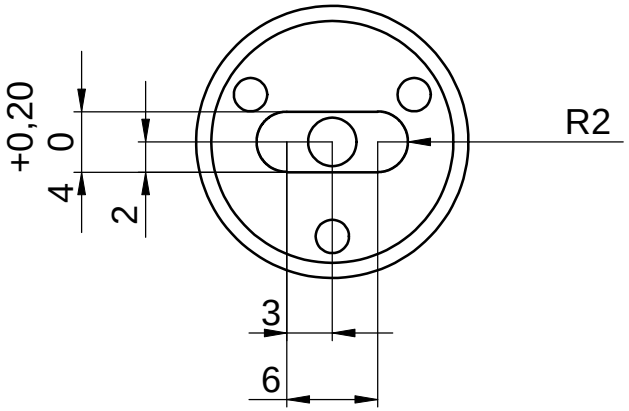


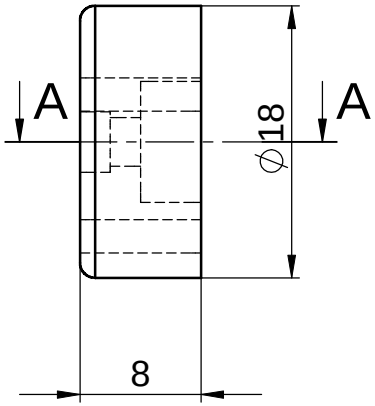
TU Delft	benaming: spanner_gripperside_interface		datum: 21-1-2010	maateenheid: mm
	ial: Aluminium	schaal: 2:1	groep: <<groep>>	formaat: A4
		getekend: J. Wildenbeest		tekeningnummer: <<tekeningnummer>>

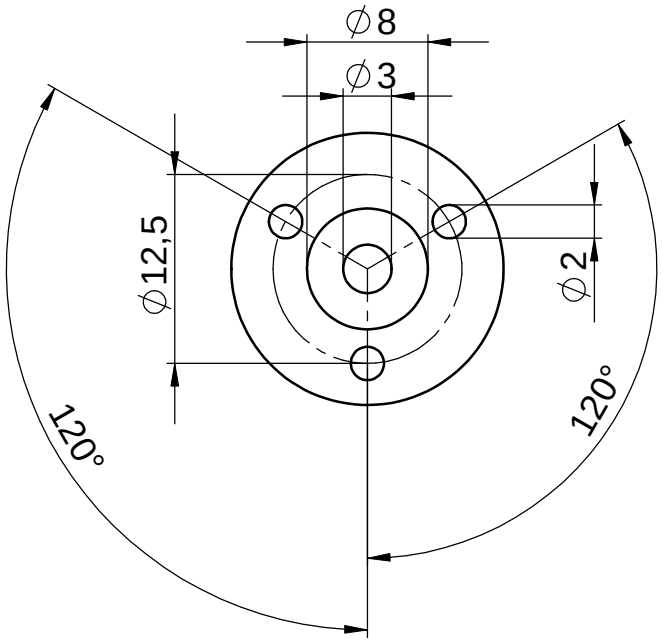
SolidWorks Student License
Academic Use Only





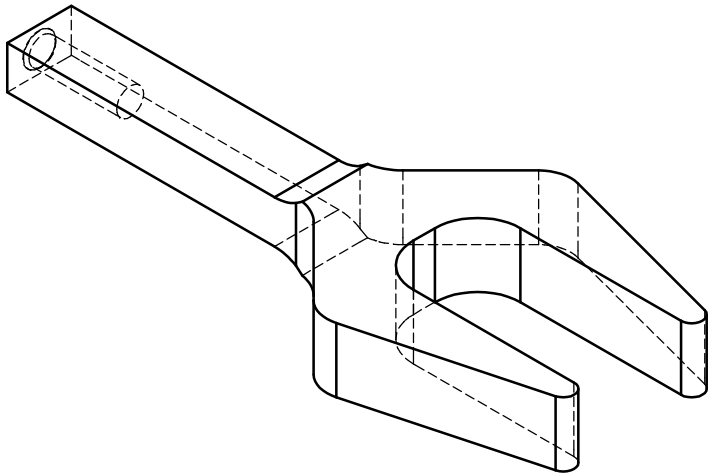


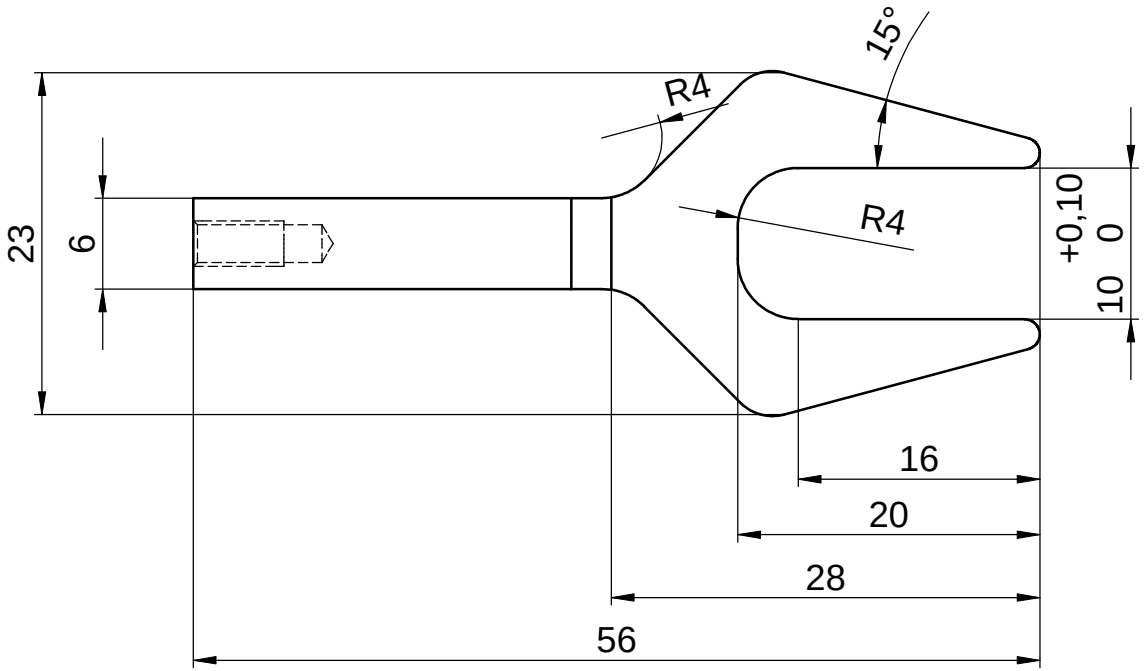


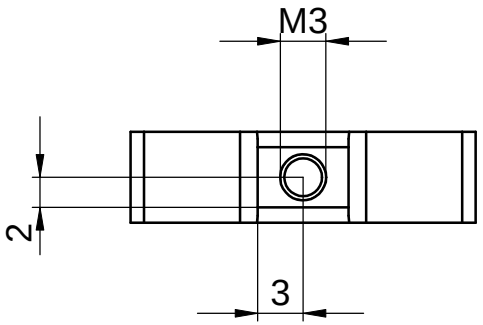


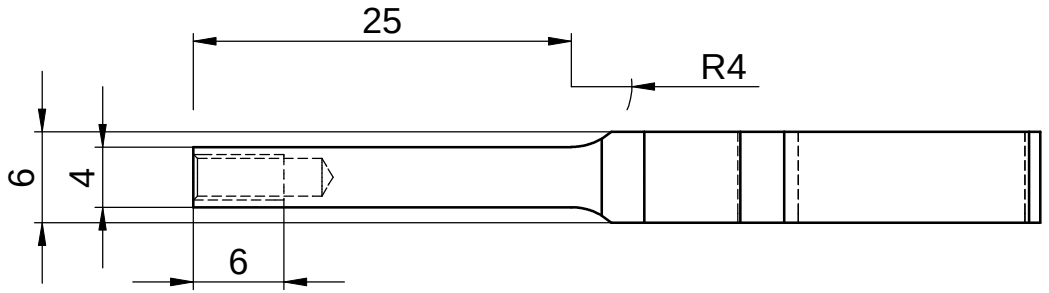
TU Delft	benaming: spanner_tool_forcesensor_interface		datum: 21-1-2010	maateenheid: mm
		schaal: 2:1	groep: <<groep>>	formaat: A4 / tekeningnummer: <<tekeningnummer>>
	al: Aluminium	getekend: J. Wildenbeest		

SolidWorks Student License
Academic Use Only



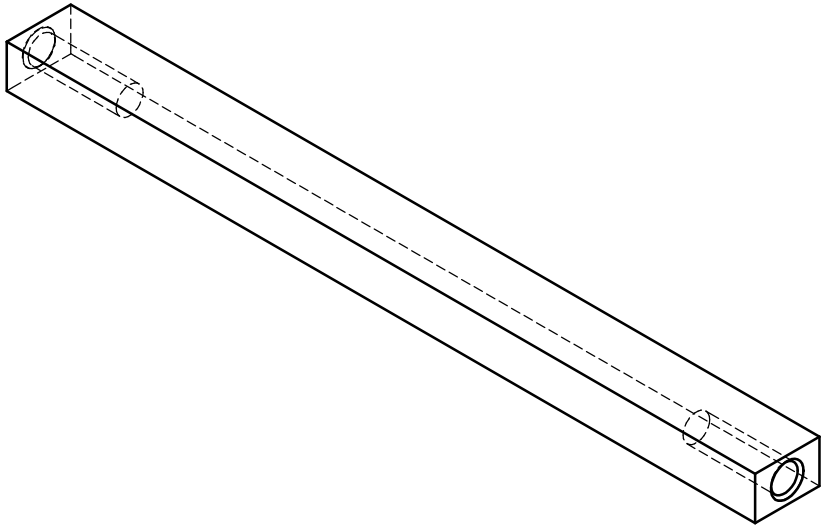


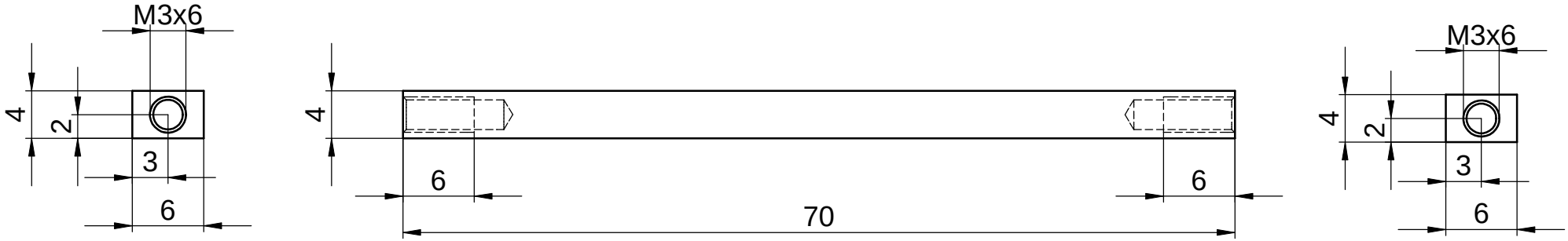




TU Delft	benaming spanner_toolside		datum 21-1-2010	maateenheid mm
		schaal 2:1	groep <<groep>>	formaat A4
	al: Aluminium	getekend J. Wildenbeest		tekeningnummer <<tekeningnummer>>

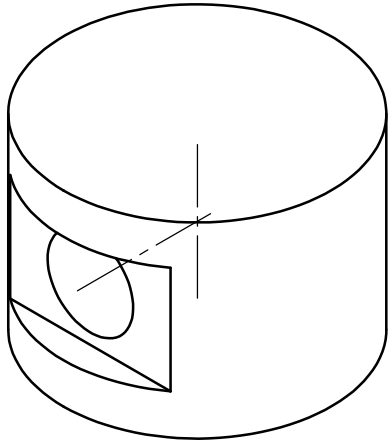
SolidWorks Student License
Academic Use Only

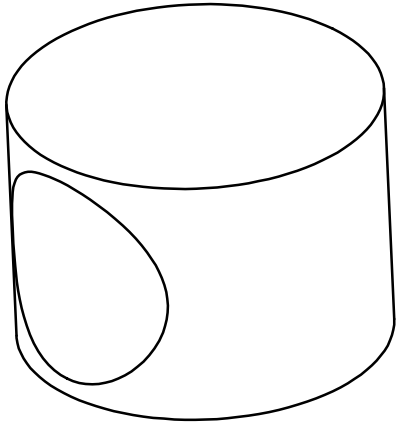


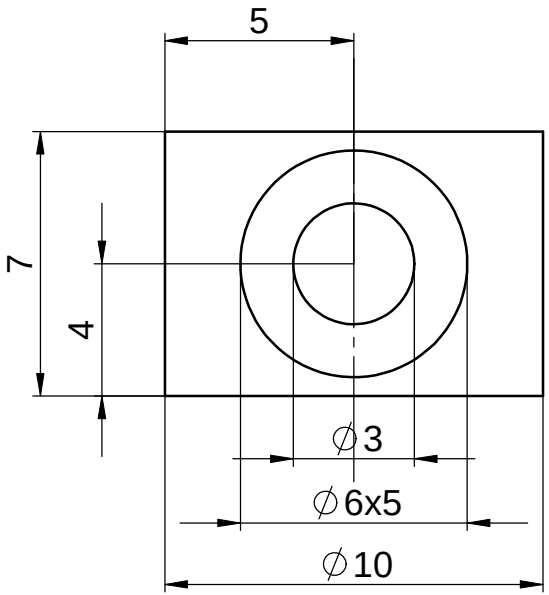


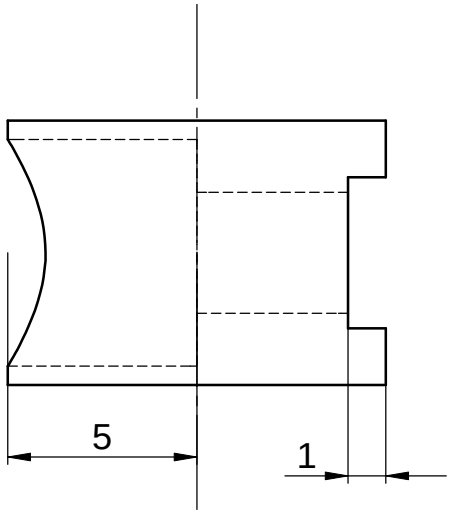
TU Delft	benaming spanner_gripperside_staf		datum 21-1-2010	maateenheid mm
		schaal 2:1	groep <<groep>>	formaat A4 / tekeningnummer <<tekeningnummer>>
	al: Aluminium	getekend J. Wildenbeest		

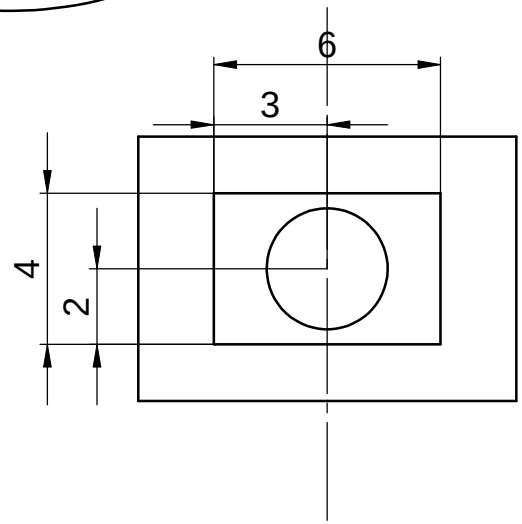
SolidWorks Student License
Academic Use Only











TU Delft	benaming spanner_gripperside_head		datum 21-1-2010	maateenheid mm
		schaal 5:1	groep <<groep>>	formaat A4 / tekeningnummer <<tekeningnummer>>
	al: Aluminium	getekend J. Wildenbeest		

SolidWorks Student License
Academic Use Only

A1.2 Hardware Design - Slave

For the slave, only one single part had to be designed. This part is colored blue in Fig. A8, which shows a CAD impression of the slave. Fig. A9 shows the realization.

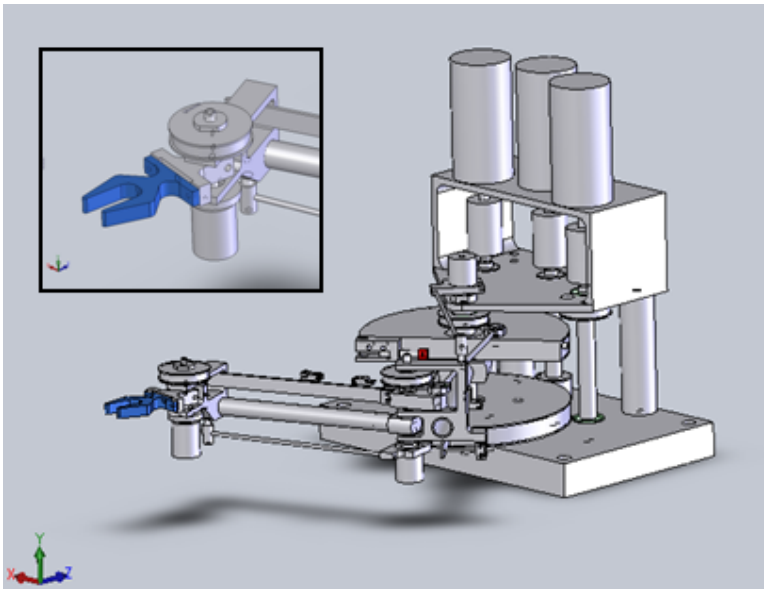


Figure A8: The slave – CAD

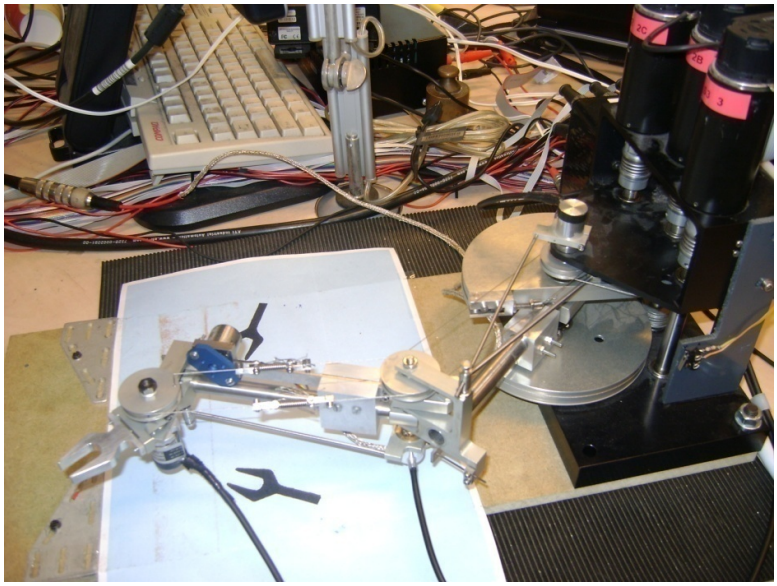
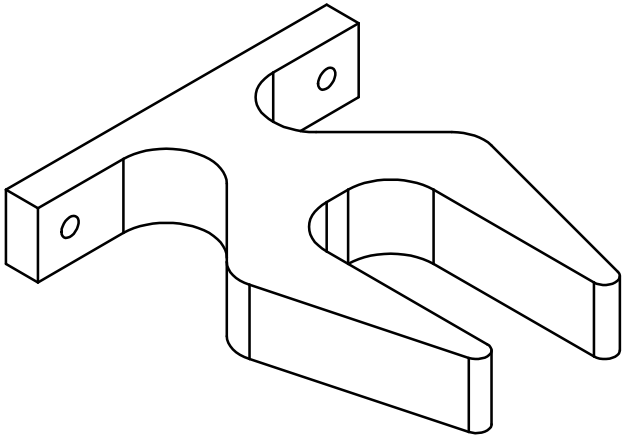
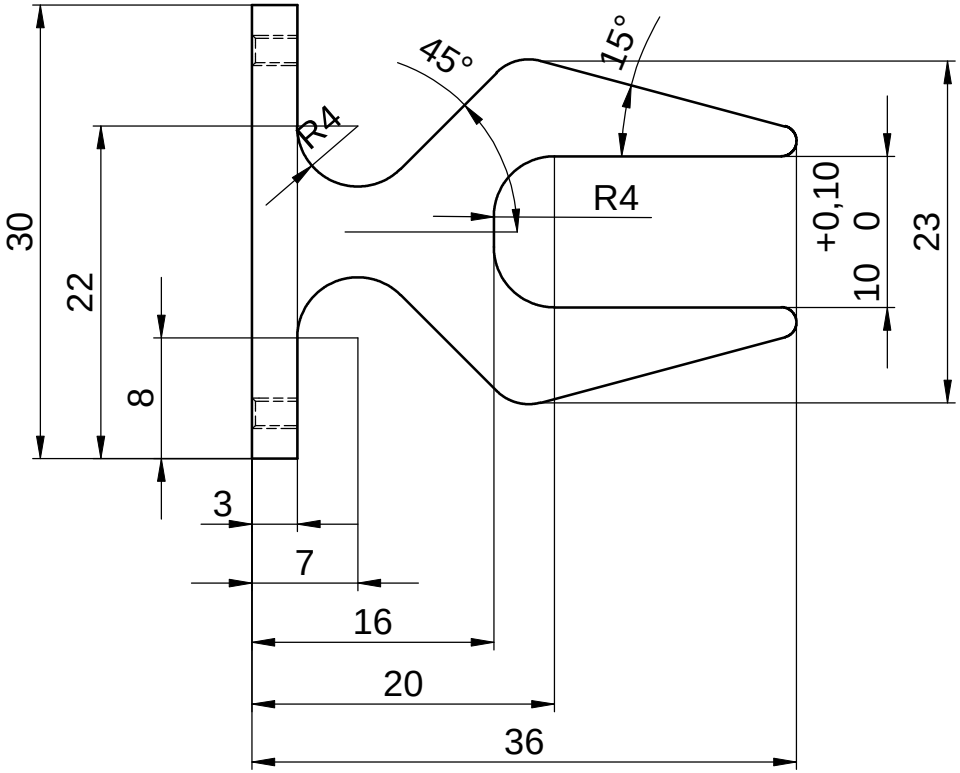
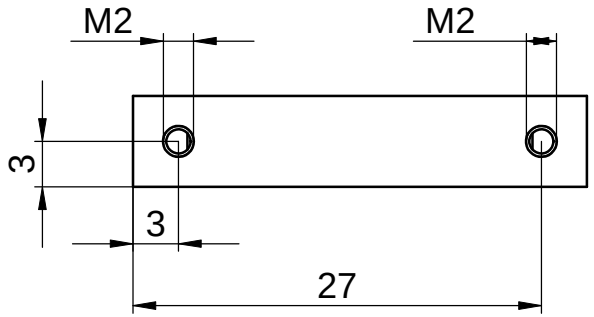
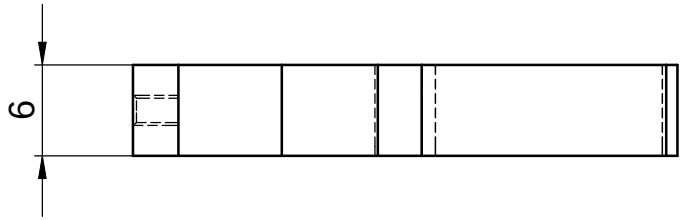


Figure A9: The slave - realization









TU Delft	benaming spanner_slave		datum 21-1-2010	maateenheid mm
		schaal 2:1	groep <<groep>>	formaat A4
	al: Aluminium	getekend <J. Wildenbeest		tekeningnummer <<tekeningnummer>>

SolidWorks Student License
Academic Use Only

A1.3 Hardware Design - Environment

The environment the telemanipulator is interacting with, consists out of a M6 bolt constructed on a aluminum plate. The moment on the bolt is created artificially with a spring, such that the tightening torque is constant. Sliders in the stainless steel base allow to adjust the height of the aluminum plate. See Fig. A10 for a CAD impression of the environment. Fig. A11 shows the realization.
The static friction is measured by slowly raising the tightening torque, starting from 0 Nmm. The tightening torque is measured using the ATI 6-dof force sensor, at the instant the bolt starts rotating. The bolt's dynamic friction is determined by rotating the bolt at a constant speed, while the tightening torques are measured. Both the static and dynamic friction are measured for twenty instances and are respectively estimated at mean (std) = 35.7 (2.0) Nmm and 31.6 (6.1) Nmm. This is typically a torque which cannot be delivered (continuously) bare-handed and therefore tools are used. For the Matlab code see the folder <*\Experiments\PlanarSetup\BoltFriction>.

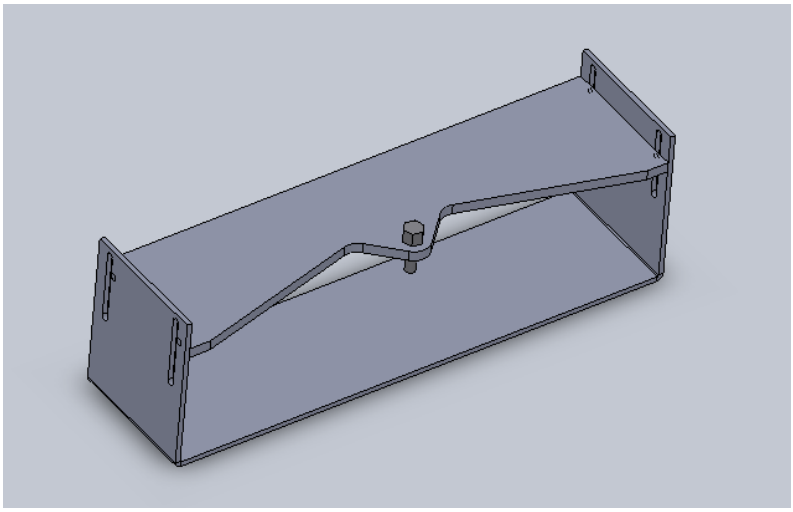


Figure A10: The environment - CAD

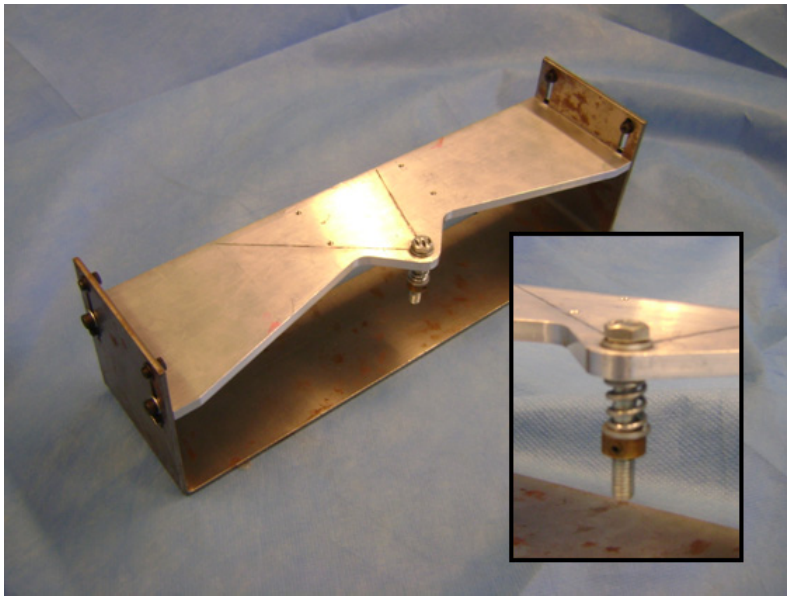
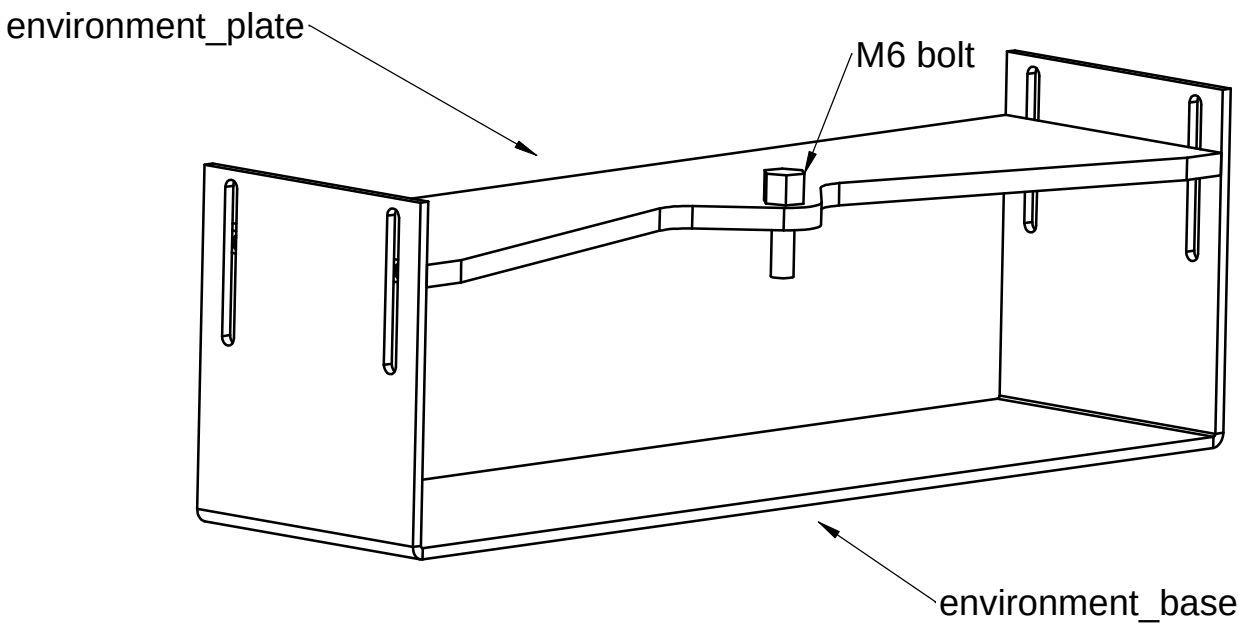
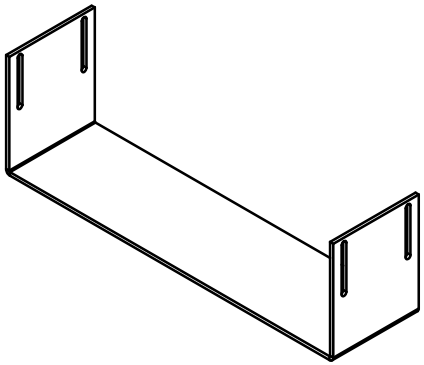


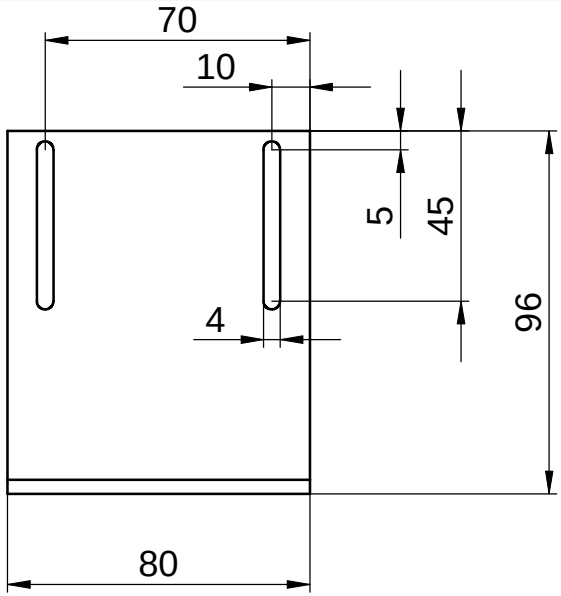
Figure A11: The environment - realization

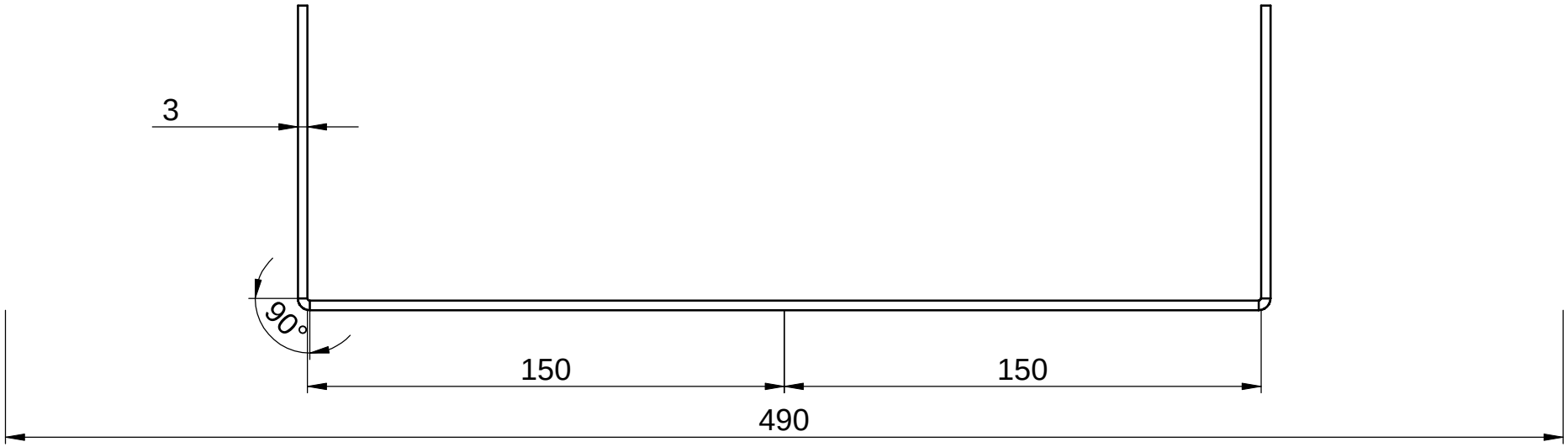


TU Delft	benaming: environment		datum: 21-1-2010	maateenheid: <<maateenheid>>
		schaal: 1:2	groep: <<groep>>	formaat: A4
	ial: Aluminium	getekend: J. Wildenbeest		tekeningnummer: <<tekeningnummer>>

SolidWorks Student License
Academic Use Only

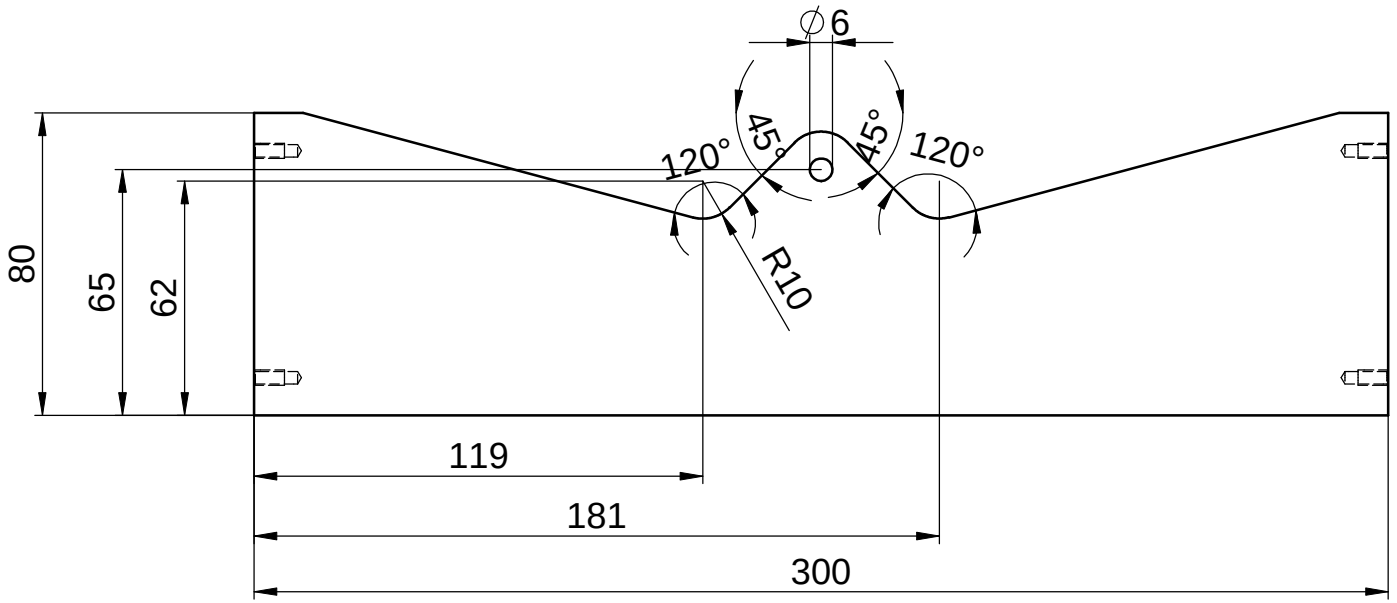


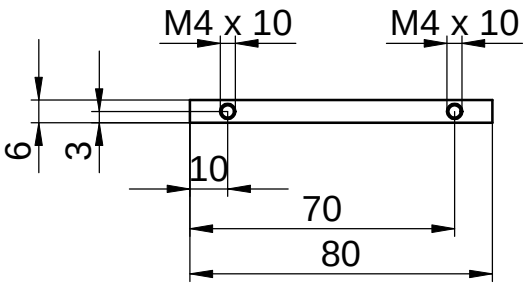


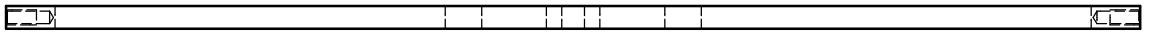


TU Delft	benaming environment_base		datum 21-1-2010	maateenheid mm
		schaal 1:5	groep <<groep>>	formaat A4
	material: sheet metal	getekend J. Wildenbeest		tekeningnummer <<tekeningnummer>>

SolidWorks Student License
Academic Use Only







TU Delft	benaming: environment_plate		datum: 21-1-2010	maateenheid: <<maateenheid>>
		schaal: 1:5	groep: <<groep>>	formaat: A4
	al: Aluminium	getekend: <<naam>>		tekeningnummer: <<tekeningnummer>>

SolidWorks Student License
Academic Use Only

A2 Kinematics

A2.1 Master Kinematics & Jacobian

Acknowledgement: The master kinematics and Jacobian have been derived in cooperation with Henri Boessenkool.

Munin's master is based on a double-rhomb parallel architecture. Two parallelograms can be distinguished, an upper and a lower parallelogram (see Fig. A12), each driven by two actuators.

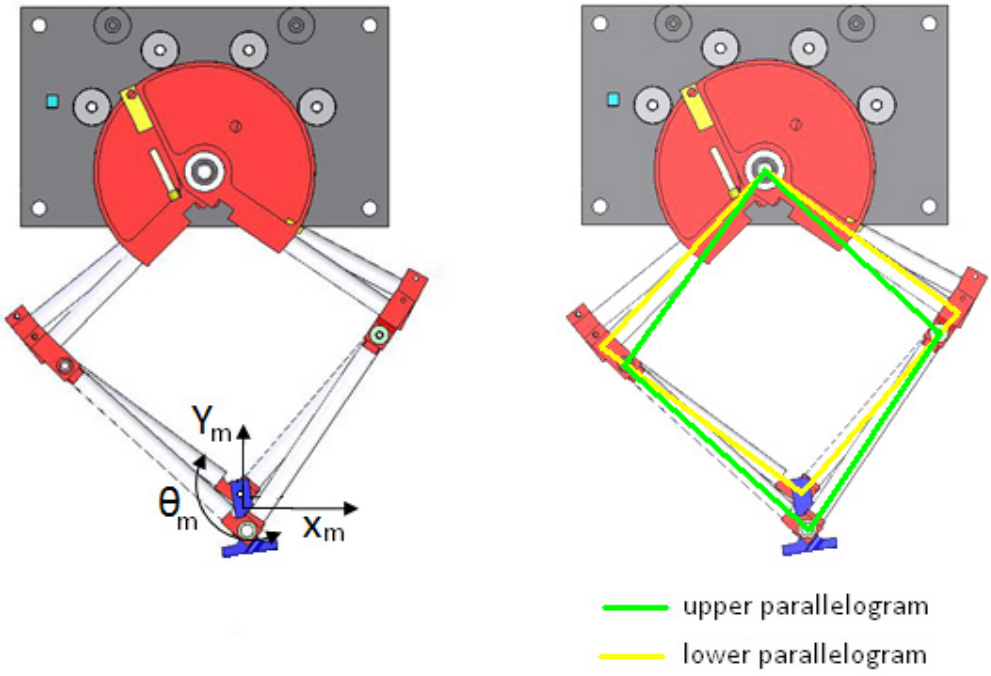


Figure A12: The master architecture

Fig. A13 shows a kinematic drawing of a single parallelogram structure. The angles γ_1 and γ_2 correspond to the motor angles, respectively q_1 and q_2 for the lower parallelogram and q_3 and q_4 for the upper parallelogram. Point E defines the endpoint position of the parallelogram.

In the original control model, the link lengths are assumed equally ($L = L_1 = L_2 = L_3 = L_4$), which severely simplifies the equations. However, strictly speaking, the structure is not a parallelogram as the opposite sides are not equal in length. The result of this simplification of the system is a nonlinear error of the endpoint position of max. 9 mm in y-direction and max. 10 degrees in Θ-direction (due to symmetry, the error in x-direction averages out).

In previous research, this inaccuracy of the master was not a big issue. After all, the data analysis is performed on the slave side, while the operator does not notice the error. However, for tasks in which the master position is used in the analysis (HandsOn task) or more sophisticated control

algorithms (e.g. shared control, preliminary MSc thesis H. Boessenkool, 2010), these errors are not acceptable, not for data analysis, not for control stability. Hence, the master kinematics are recalculated for the actual link lengths. Table 1 lists the actual link lengths for the lower and upper parallelogram.

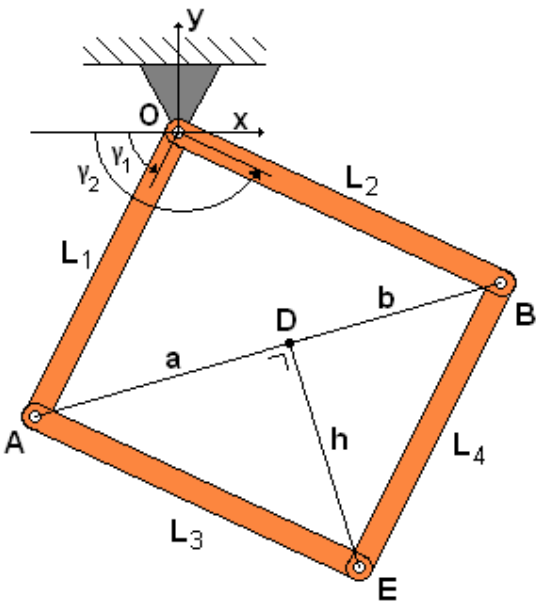


Figure A13: Master's Forward Kinematics

Table 1: Master Link Lenghts

Link No.	Lower Parallelogram [mm]	Upper Parallelogram [mm]
L1	130.0	130.0
L2	130.0	130.0
L3	134.0	129.5
L4	129.5	134.0

Forward Kinematics

The Munin master is equipped with rotary encoder on the motor axis. The forward kinematics describe the Cartesian endpoint position X_m as a function of the motor angles (q_m).

$$X_m = \begin{bmatrix} x_m \\ y_m \\ \theta_m \end{bmatrix}, \text{ and } q_m = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}$$

Notice that for analysis of the lower parallelogram, angles γ_1 and γ_2 correspond to motors angles q_1 and q_2 , while for analysis of the upper parallelogram, angles γ_1 and γ_2 correspond to motors angles q_3 and q_4.

In the original control model (where L = L_1 = L_2 = L_3 = L_4) the endpoint position E is given by the simple equation:

$$E = \begin{bmatrix} E_x \\ E_y \end{bmatrix} = \begin{bmatrix} L\cos\gamma_1 + L\cos\gamma_2 \\ L\sin\gamma_1 + L\sin\gamma_2 \end{bmatrix} \quad (1)$$

However, if the kinematics are calculated using the actual link lengths, endpoint E can be calculated as follows:

From Fig. A13 can be obtained that:

$$A = \begin{bmatrix} A_x \\ A_y \end{bmatrix} = \begin{bmatrix} -L_1\cos\gamma_1 \\ -L_1\sin\gamma_1 \end{bmatrix} \quad (2)$$

and

$$B = \begin{bmatrix} B_x \\ B_y \end{bmatrix} = \begin{bmatrix} -L_2\cos\gamma_2 \\ -L_2\sin\gamma_2 \end{bmatrix} \quad (3)$$

Endpoint E can n be determined using the rules of circle intersection (see Fig. A14).

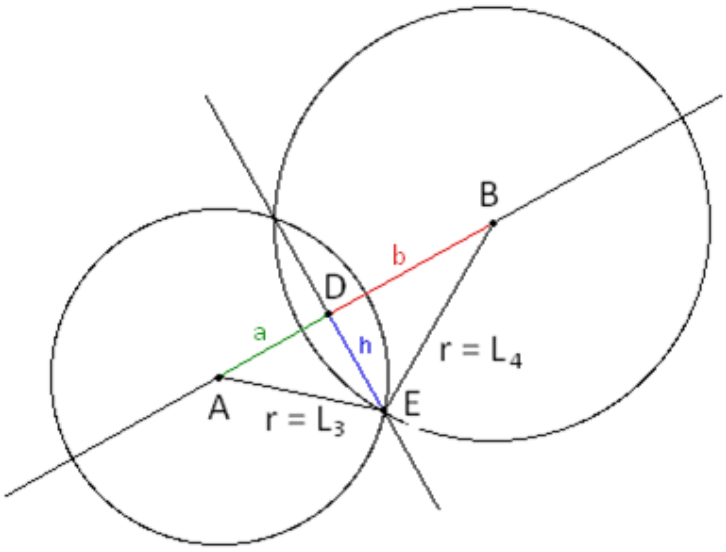


Figure A14: Intersection of circles

Distance d between points A and B can written:

$$d = a + b = \sqrt{(B_x - A_x)^2 + (B_y - A_y)^2} \tag{4}$$

Considering triangles ADE and BDE, $L_3^2 = a^2 + h^2$ and $L_4^2 = b^2 + h^2$. As $d = a + b$

$$a = \frac{L_3^2 - L_4^2 + d^2}{2d} \tag{5}$$

$$h = L_3^2 - a^2 \tag{6}$$

Now D can be determined:

$$D = \begin{bmatrix} D_x \\ D_y \end{bmatrix} = \begin{bmatrix} A_x + (B_x - A_x) \cdot (a/d) \\ A_y + (B_y - A_y) \cdot (a/d) \end{bmatrix} \tag{7}$$

Then endpoint position E is given by:

$$E = \begin{bmatrix} E_x \\ E_y \end{bmatrix} = \begin{bmatrix} D_x + (B_x - A_x) \cdot (h/d) \\ D_y + (B_y - A_y) \cdot (h/d) \end{bmatrix} \tag{8}$$

The Cartesian master endpoint position (X_m) is described as a function of the upper and lower parallelogram endpoint positions (respectively E_{upper} and E_{lower}):

$$X_m = \begin{bmatrix} x_m \\ y_m \\ \theta_m \end{bmatrix} = \begin{bmatrix} (E_{upper,x} + E_{lower,x})/2 \\ (E_{upper,x} + E_{lower,x})/2 \\ \arctan\left((E_{lower,y} + E_{upper,y})/(E_{lower,x} + E_{upper,x})\right) \end{bmatrix} \tag{9}$$

Finally by substituting the equation for endpoint position E (8) for both the upper and lower parallelogram into (9), the Cartesian endpoint position X_m is be obtained.

The master's Cartesian velocities ($\dot{X}_m$) and the motor torques (τ_m) can be calculated by deriving the Jacobian ($J_{\partial X_m / \partial q_m}$) for the following equations:

$\dot{X}_m = J_{\partial X_m / \partial q_m} \dot{q}_m$, whereas $\dot{q}_m$ are the measured motor velocities (10)

and

$\tau_m = J^T_{\partial X_m / \partial q_m} F_m$, whereas F_m are the control forces in Cartesian space . (11)

However, deriving the Jacobian ($J_{\partial X_m / \partial q_m}$) from the forward kinematics is not valid, as $q_1 ... q_4$ are not independent. In the original control model, this problem was solved by assuming an equal distribution of endpoint velocities and forces (respectively $\dot{X}_m$ and F_m) along the parallelogram endpoints E_{upper} and E_{lower}. Effectively, this separates the problem in two systems of independent equations and solves for a number of equations, such that the set of equations becomes solvable. Nonetheless, this approach yields suboptimal performance of the master device, as de distribution of motor torques is not very well controlled. For example one motor could be working at maximum performance, while the other motors have zero output.

Obviously, this approach is only limiting if the device is working close to its maximum performance. Fig. A15 shows a plot of the motor torques during execution of the bolt and spanner task. It can be seen that the motor frequently operate at their maximum output torque output (0.05 Nm).

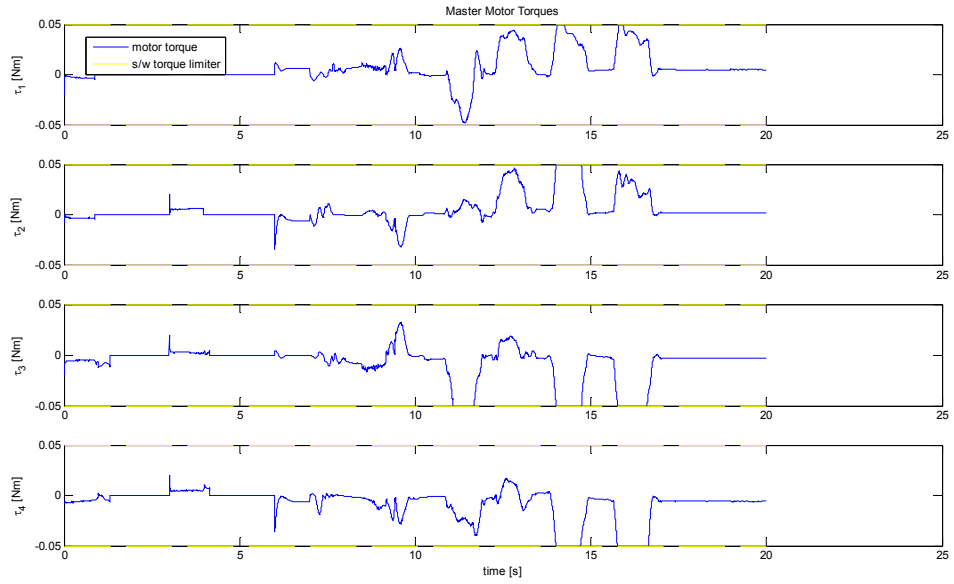


Figure A59: Limited motor torques

The impact of this problem can be best illustrated with an example. The motors need to generate a Cartesian endpoint force F_m at position $X_m = \begin{bmatrix} .05 & .15 & .5\pi \end{bmatrix}$:

$$F_m = \begin{bmatrix} F_{m,x} \\ F_{m,y} \\ F_{m,\theta} \end{bmatrix} = \begin{bmatrix} 1 \text{ [N]} \\ 1 \text{ [N]} \\ -0.001 \text{ [Nm]} \end{bmatrix}$$

Using $\tau_m = J^T F_m$ it is obtained that $\tau_m = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} = \begin{bmatrix} 0.0200\ [\text{Nm}] \\ 0.0820\ [\text{Nm}] \\ 0.0150\ [\text{Nm}] \\ 0.0819\ [\text{Nm}] \end{bmatrix}$

As the motor torques are limited to 0.05 Nm, the realized motor torques are:

$\hat{\tau}_m = \begin{bmatrix} 0.0200\ [\text{Nm}] \\ 0.0500\ [\text{Nm}] \\ 0.0150\ [\text{Nm}] \\ 0.0500\ [\text{Nm}] \end{bmatrix}$, which result in a Cartesian endpoint force $\hat{F}_m = \begin{bmatrix} 0.0040\ [\text{N}] \\ 0.0048\ [\text{N}] \\ 0.0152\ [\text{Nm}] \end{bmatrix}$ (!).

Thus, not only the magnitude of the forces is heavily affected by the limiter; it is even possible to create forces in the opposing directions! When operating, this is observed as very awkward behavior, sometimes described as 'a negative stiffness'.

A 'quick fix' for the performance problem is to offset the slave's Point of Rotation (PoR), such that rotations at the slave side are partially fed back as translations on the master side. The Cartesian endpoint (X_s) is then defined at the slave's tool interface (see the green dot in Fig. A16) and not at its 3rd joint. This quick fix is an efficient solution for tasks with minimal rotations and rotational arms (e.g. the experiments conducted in Christiansson, 2008). However, for tasks like the bolt-and-spanner tasks, this offset is quickly noticed by the operator and experienced as 'weird' and 'hard to control'.

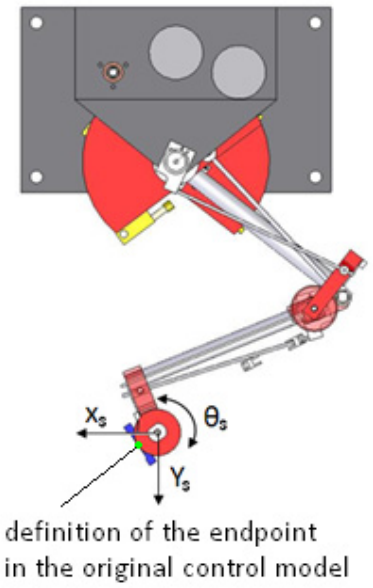


Figure A16: The slave's point of rotation offset

Instead, three other measures have been taken to prevent saturation of the master's motor torques:

- The master Jacobian is recalculated based on the principle of the Moore-Penrose pseudoinverse. The Moore-Penrose theorem offers a least square solution for the force redundancy problem (Penrose, 1955). Hence a more optimal distribution of the motor torques is attained. The Moore-Penrose pseudoinverse is calculated from the inverse kinematics.
- The saturation value implied on the motor torques (τ_m) by the software has been increase from 0.05 Nm to 0.09 Nm. The nominal current (max. continuous current) of the Maxon RE35 90W motors is 2.5 A. As the torque constant of the motor is 0.039 Nm/A, a maximum torque of 0.09 Nm corresponds to approximately 2.3 A.
- The endpoint forces in Cartesian space F_m are limited to |10| [N] for the translations ($F_{m,x}$ and $F_{m,y}$), and |0.125| [Nm] for the rotation ($F_{m,\theta}$).

For the specific task performed in this research, the bolt-and-spanner task, these three measures solves the saturation problem of the motors at the master side.

Notice that due to the kinematical design (small actuator arm), generating endpoint torques ($F_{m,\theta}$) is very problematic in general and saturates the motor torques rapidly. This is quite inconvenient when performing bolt-and-spanner-like tasks.
Also notice that the maximum torque is very dependent of the master's position in its workspace. The master is much better capable of producing endpoint torques close the base (small y) than it is at the border of its workspace (large y).

Inverse Kinematics

The inverse kinematics describe the motor angles (q_m) as a function of the Cartesian endpoint position X_m . Although there is a redundant number of actuators (four) to drive the endpoint degrees of freedom (three), kinematically there is always only one solution possible. However, the Munin's master is force redundant; there is an infinite number of motor torques (τ_m) that lead an Cartesian endpoint force (F_m). As stated, the Moore-Penrose theorem offers a least square solution for the force redundancy problem.

If the distance between the endpoints of the lower and upper parallelogram is described by L_{crank} , than:

$$E_{lower} = \begin{bmatrix} E_{lower,x} \\ E_{lower,y} \end{bmatrix} = \begin{bmatrix} x_m + .5L_{crank}\cos(\theta_m) \\ y_m + .5L_{crank}\sin(\theta_m) \end{bmatrix} \quad (12)$$

And

$$E_{upper} = \begin{bmatrix} E_{upper,x} \\ E_{upper,y} \end{bmatrix} = \begin{bmatrix} x_m - .5L_{crank}\cos(\theta_m) \\ y_m - .5L_{crank}\sin(\theta_m) \end{bmatrix} \quad (13)$$

Fig. A17 shows the definitions of points, angles and lengths.

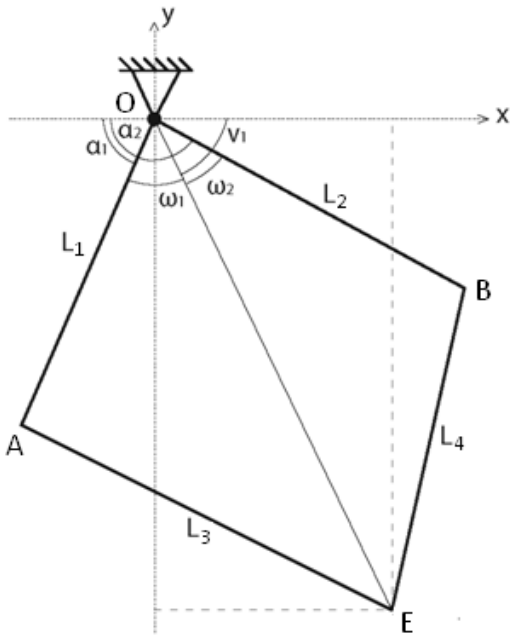


Figure A17: Master's Inverse Kinematics

Let the bar lengths be described by:

$$L_a = 0.1300 \text{ m}$$
$$L_b = 0.1295 \text{ m}$$
$$L_c = 0.1340 \text{ m}$$

For the lower parallelogram structure it can be deducted that:

$$E_{lower} = \sqrt{E_{lower,x}^2 + E_{lower,y}^2}$$
$$\nu_{lower} = \arccos(E_{lower,x} / E_{lower})$$
$$\omega_{lower,1} = \arccos\left(\frac{L_c^2 - L_a^2 - E_{lower}{}^2}{-2L_a E_{lower}}\right)$$
$$\omega_{lower,2} = \arccos\left(\frac{L_b^2 - L_a^2 - E_{lower}{}^2}{-2L_a E_{lower}}\right)$$

And thus:

$$q_1 = \pi - \nu_{lower} - \omega_{lower,1}$$
$$q_2 = \pi - \nu_{lower} + \omega_{lower,2} \quad (14)$$

The upper parallelogram is analyzed with slightly different bar lengths:

$$E_{upper} = \sqrt{E_{upper,x}^2 + E_{upper,y}^2}$$
$$\nu_{upper} = \arccos(E_{upper,x} / E_{upper})$$
$$\omega_{upper,1} = \arccos\left(\frac{L_b^2 - L_a^2 - E_{upper}{}^2}{-2L_a E_{upper}}\right)$$
$$\omega_{upper,2} = \arccos\left(\frac{L_c^2 - L_a^2 - E_{upper}{}^2}{-2L_a E_{upper}}\right)$$

And thus:

$$q_3 = \pi - \nu_{upper} - \omega_{upper,1}$$
$$q_4 = \pi - \nu_{upper} + \omega_{upper,2} \quad (15)$$

Equations (14) and (15) together describe all four angles of the master's motors:

$$q_m = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} \pi - \nu_{lower} - \omega_{lower,1} \\ \pi - \nu_{lower} + \omega_{lower,2} \\ \pi - \nu_{upper} - \omega_{upper,1} \\ \pi - \nu_{upper} + \omega_{upper,2} \end{bmatrix} \quad (16)$$

The Jacobian ($J_{\partial q / \partial X, m}$), can be obtained, using equations (9) and (16):

$$J_{\partial q_m / \partial X_m} = \begin{bmatrix} \frac{\partial q_1}{\partial x_m} & \frac{\partial q_1}{\partial y_m} & \frac{\partial q_1}{\partial \theta_m} \\ \frac{\partial q_2}{\partial x_m} & \frac{\partial q_2}{\partial y_m} & \frac{\partial q_2}{\partial \theta_m} \\ \frac{\partial q_3}{\partial x_m} & \frac{\partial q_3}{\partial y_m} & \frac{\partial q_3}{\partial \theta_m} \\ \frac{\partial q_4}{\partial x_m} & \frac{\partial q_4}{\partial y_m} & \frac{\partial q_4}{\partial \theta_m} \end{bmatrix} \quad (17)$$

The master's Cartesian velocities ($\dot{X}_m$) and the motor torques (τ_m) can be calculated from the Jacobian ($J_{\partial q_m / \partial X_m}$), using the following equations:

$\dot{X}_m = J^{-1}_{\partial q_m / \partial X_m} \dot{q}_m$, whereas $\dot{q}_m$ are the motor velocities (18)

and

$\tau_m = J^{-T}_{\partial q_m / \partial X_m} F_m$, whereas F_m are the control forces in Cartesian space (19)

The Jacobian ($J_{\partial q_m / \partial X_m}$, equation (17)) can be inverted using the Moore-Penrose pseudoinverse (Penrose, 1955). First a Singular Value Decomposition is made of $J_{\partial q_m / \partial X_m}$:

$$J_{\partial q_m / \partial X_m} = U \Sigma V^T$$

Then the psuedoinverse is calculated using:

$$J^{-1}_{\partial q_m / \partial X_m} = V \Sigma^{-1} U^T \quad (20)$$

By substituting equation (20) into equation (18) or (19), respectively the master's Cartesian endpoint velocities and motor torques can be determined.

Notice that $J^{-1}_{\partial q_m / \partial X_m}$ is not a 'real' inverse and is therefore usually denoted as $J^{+}_{\partial q_m / \partial X_m}$ in literature. In Matlab, the pseudoinverse is easily calculated using the function 'pinv()'.

The Matlab code (see *\Experiments\PlanarSetup\Kinematics\Master\MasterKinematics.m)

```
%% Jeroen Wildenbeest & Henri Boessenkool
%  May 27th 2010
%
%  Master Kinematics
%  In the original controller the master kinematics are approximated by
%  assuming a parallelogram structure. As the bar lengths of the master are
%  not identical, and thus the parallelogram approach introduces an error
%  fatal to the LF control, the master kinematics are now accurately
%  calculated.

clear all
close all
clc

% syms L1_l L2_l L3_l L4_l % parallelogram bar lengths
% syms L1_u L2_u L3_u L4_u % parallellogram bar lengths
syms q1 q2 q3 q4 % motor angles

L1_l = .13;
L2_l = .13;
```

```
L3_l = .134;
L4_l = .1295;
L1_u = .13;
L2_u = .13;
L3_u = .1295;
L4_u = .134;

%% Kinematics of the lower parallelogram structure

x_A_l = -L1_l*cos(q1);
y_A_l = -L1_l*sin(q1);

x_B_l = -L2_l*cos(q2);
y_B_l = -L2_l*sin(q2);


d_l = sqrt((x_B_l - x_A_l)^2 + (y_B_l - y_A_l)^2);

a_l = (L3_l^2 - L4_l^2 + d_l^2)/(2*d_l);

h_l = sqrt(L3_l^2 - a_l^2);


x_D_l = x_A_l + (x_B_l - x_A_l)*(a_l/d_l);
y_D_l = y_A_l + (y_B_l - y_A_l)*(a_l/d_l);

x_E_l = x_D_l + (y_B_l - y_A_l)*(h_l/d_l);
y_E_l = y_D_l - (x_B_l - x_A_l)*(h_l/d_l);

% x_E_l = simple(x_E_l);
% y_E_l = simple(y_E_l);


%% Kinematics of the upper parallelogram structure

x_A_u = -L1_u*cos(q3);
y_A_u = -L1_u*sin(q3);

x_B_u = -L2_u*cos(q4);
y_B_u = -L2_u*sin(q4);


d_u = sqrt((x_B_u - x_A_u)^2 + (y_B_u - y_A_u)^2);

a_u = (L3_u^2 - L4_u^2 + d_u^2)/(2*d_u);

h_u = sqrt(L3_u^2 - a_u^2);


x_D_u = x_A_u + (x_B_u - x_A_u)*(a_u/d_u);
y_D_u = y_A_u + (y_B_u - y_A_u)*(a_u/d_u);

x_E_u = x_D_u + (y_B_u - y_A_u)*(h_u/d_u);
y_E_u = y_D_u - (x_B_u - x_A_u)*(h_u/d_u);

% x_E_u = simple(x_E_u);
% y_E_u = simple(y_E_u);
```

```
%% Master Endpoint Kinematics

C = [x_E_l ; y_E_l ; x_E_u ; y_E_u];

x_m = (x_E_l + x_E_u)/2;
y_m = (y_E_l + y_E_u)/2;
% theta_m = atan2((y_E_l-y_E_u),(x_E_l-x_E_u)); % Matlab does not accept
this symbolically
s = sqrt((x_E_u - x_E_l)^2 + (y_E_u - y_E_l)^2);
theta_m = (acos((-x_E_u + x_E_l)/s));
X_m = [x_m ; y_m ; theta_m];


%% Master Jacobian (not valid, q's are not independent)

% q = [q1 q2 q3 q4];
% J = jacobian(X_m,q);
% J = simple(J)


%% Inverse Kinematics

L1 = .13;         %[cm] length base arms
L2 = .1295;         %[cm] length arm 2
L3 = .134;         %[cm] length arm 3

Lcrank = .02;

x1 = x + .5*Lcrank*cos(theta);
y1 = y + .5*Lcrank*sin(theta);
x2 = x - .5*Lcrank*cos(theta);
y2 = y - .5*Lcrank*sin(theta);

% Lower parallellogram
OA1 = sqrt(x1^2 + y1^2);
v1 = acos(x1/OA1);
w1 = acos((L3^2 - L1^2 - OA1^2) / (-2*L1*OA1));
w2 = acos((L2^2 - L1^2 - OA1^2) / (-2*L1*OA1));

q1 = pi - v1 - w1;
q2 = pi - v1 + w2;

% Upper parallellogram
OA2 = sqrt(x2^2 + y2^2);
v2 = acos(x2/OA2);
w3 = acos((L2^2 - L1^2 - OA2^2) / (-2*L1*OA2));
w4 = acos((L3^2 - L1^2 - OA2^2) / (-2*L1*OA2));

q3 = pi - v2 - w3;
q4 = pi - v2 + w4;


% Motor angles
q = [q1 q2 q3 q4]


% Master Jacobian
```

```
J_dq_dX = jacobian(q,X_m)
J_dq_dX_inv = pinv(J_dq_dX)
J_dq_dX_inv_T = J_dq_dX_inv '
```

A2.2 Slave Kinematics & Jacobian

Fig. A18 shows an illustration of the Munin's slave. Each of the angles (α_1, α_2, α_3) is driven by an individual actuator. Notice that the slave contains two sets of three encoders to measure the angles. One encoder set is located on each of motor axis (Θ_m), and another set is located on the joint axis (Θ_j). As the transmission from motor to joint is not entirely stiff, approx. 1-10 N/rad, the difference between the encoder angles can be used as an estimate for the contact force with the environment. This form of force estimation is based on the principle of series-elastic actuation (Pratt et al., 1995).

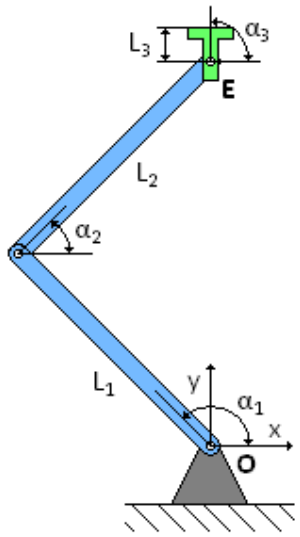


Figure A18: Slave Kinematics

From Fig. A18 can be deducted that:

$$E = \begin{bmatrix} E_x \\ E_y \\ E_\theta \end{bmatrix} = X_s = \begin{bmatrix} x_s \\ y_s \\ \theta_s \end{bmatrix} = \begin{bmatrix} L_1 \cos\alpha_1 + L_2 \cos\alpha_2 \\ L_1 \sin\alpha_1 + L_2 \sin\alpha_2 \\ \alpha_3 \end{bmatrix} \quad (21)$$

Where $L_1 \ldots L_3$ are given by:

$$L_1 = 0.134 \text{ m}$$
$$L_2 = 0.130 \text{ m}$$
$$L_3 = 0.150 \text{ m}$$

Notice that in the Matlab Simulink control model the vector containing the slave angles is defined as: $\theta_s = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 & \theta_5 & \theta_6 \end{bmatrix}^T$, whereas $\theta_1 \ldots \theta_3$ correspond to the motor angles (Θ_m), while $\theta_4 \ldots \theta_6$ correspond to the joint angles(Θ_j). By default, the slave is controlled using the joint angles, hence:

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \theta_j = \begin{bmatrix} \theta_4 \\ \theta_5 \\ \theta_6 \end{bmatrix} \tag{22}$$

Substituting equation (22) into (21) and taking the Jacobian $J_{\partial X_s/\partial \theta_j}$ yields

$$J_{\partial X_s/\partial \theta_j} = \begin{bmatrix} -L_1 \sin\alpha_1 & -L_2 \sin\alpha_2 & 0 \\ L_1 \cos\alpha_1 & L_2 \cos\alpha_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{23}$$

Using (23), the slave's Cartesian velocities ($\dot{X}_s$) and the motor torques (τ_s) can be calculated:

$\dot{X}_s = J_{\partial X_s/\partial \theta_j} \dot{q}_s$, whereas $\dot{q}_s$ are the (measured) motor velocities

and

$\tau_s = J^T_{\partial X_s/\partial \theta_j} F_s$, whereas F_s are the slave's control forces in Cartesian space (endpoint forces).

Finally, the Cartesian contact forces with the environment $F_{c,slave}$ can be estimated on basis of joint forces, using a joint positional difference and a joint stiffness.

$F_{c,slave} = J^{-T}_{\partial X_s/\partial \theta_j} \cdot \left(\left(\theta_j - \theta_m \right) \cdot k_j \right)$, whereas k_j is a 3x1 vector containing the joint stiffness's.

The Matlab code (see *\Experiments\PlanarSetup\Kinematics\Slave\SlaveKinematics.m)

```
%% Jeroen Wildenbeest
%  May 20th 2010
%
%  Slave Kinematics
%  Calculation of the slave's Cartesian endpoint, using the encoders
mounted on the motors (theta_m) or the encoders on the joints (theta_j).
both  encoder angles

clear all
close all
clc

% syms LOne LTwo LThree
syms theta_j_s1 theta_j_s2 theta_j_s3
syms theta_m_s1 theta_m_s2 theta_m_s3

LOne = .134;
LTwo = .13;
LThree = .15;

theta_j = [theta_j_s1 theta_j_s2 theta_j_s3]';
```

```
theta_m = [theta_m_s1 theta_m_s2 theta_m_s3]';

Xslave =    [LOne*cos(theta_j_s1)+LTwo*cos(theta_j_s2) ;
            LOne*sin(theta_j_s1)+LTwo*sin(theta_j_s2) ;
            theta_j_s3];

Js =        [-LOne*sin(theta_j_s1), -LTwo*sin(theta_j_s2), 0 ;
            LOne*cos(theta_j_s1), LTwo*cos(theta_j_s2), 0 ;
            0, 0, 1];

Js_T = Js';

LF_gain = 1;

dF_1 = inv(Js_T)*((theta_j-theta_m)*LF_gain);

dF_2 = inv(Js_T)*(theta_j*LF_gain) - inv(Js_T)*(theta_m*LF_gain);

test = dF_1 - dF_2

test_rewritten = simple(test);
```

A3 Controllers

Numerous kinds of control methods for telemanipulators exist. For the purpose of this research, to better understand what the influence of haptic feedback quality on human-in-the-loop task performance is, a suitable control architecture had to be selected.

The requirements of the controller are derived from human physiological properties, Daniel and McAree argue that there is a natural partition of the human's haptic modality, namely a highly energetic part below roughly 20 Hz and an information conveying part of roughly 30 Hz to hundreds of Hz for hard-hard contact (Daniel and McAree, 1998). Proprioceptors (interoceptive haptic receptors), like Golgi-tendon organs and Muscle spindles, foresee the information for the low-frequency (LF) channel. Mechanoreceptors (exteroceptive touch receptors) are the dominant factors in the high-frequency (HF) channel.

A suitable control architecture for this study provides both the low-frequency haptic channel as well as the high-frequency haptic channel with information.

A classification of control methods can be made according to the kind of flow variables (also called power variables) the control algorithms are communicating. There are two types of flow variables, position (or equivalent velocity) and force, which can both be communicated from master to slave and vice versa. The most commonly used control architectures are PERR-control, position-force control and 4-channel control.

PERR-controllers, also called position-error or position-position controllers, use the position power variables to control the error between master and slave. A position servo on the slave side and a position servo on the master side communicate the shared reference position, while a controller (usually a PD-controller) minimizes the error. Force feedback is acquired as a result of a positional error between master and slave when the slave interacts with the environment. Hence, the maximum transmittable impedance (the stiffness displayed by the master) is limited by the controller. Moreover, due to the absence of any 'real' force sensors, the bandwidth of force feedback provided to the human operator is relatively low compared to other control architectures, and the inertia of the slave robot is partially reflected to the human operator.

Position-force controllers communicate position from master to slave and force from slave to master. Where position sensors at the master side measure the position commands given by the human operator, force sensors at the slave side measure reaction forces at the environment. This is equivalent to the classic way of controlling industrial robots, impedance control. The main concern of this control architecture is stability; stability is only guaranteed if the proportional force gain is less than roughly the quotient of the masses of the master and the slave (Daniel and McAree, 1998).

Finally, Lawrence created a control architecture which uses all power variables in its control loop; position and force is communicated from master to slave and vice versa. Hence, 4-channel control (Lawrence, 1993) requires position and force sensors on both sides of the telemanipulator. 4-Channel control was improved by introducing local force feedback loops (Hashtrudi-Zaad et al., 1999). Using this architecture, master and slave inertias can be filtered away, while the force and position bandwidths are theoretically infinite. An interesting modification of the 4-channel control architecture is Christiansson' hard master - soft slave telemanipulation concept (Christiansson, 2007a). Christiansson reduced the impedance of the slave device by introducing a spring, practically decoupling the slave into two interconnected elements. It has been shown that certain applications benefit from this increased compliance (Christiansson, 2006), while the system stability is also increased. However, to control the slave's additional system element, an additional control channel is added; the base velocity of the slave. Hence, one could call this control architecture a '5-Channel' architecture.

It is clear that PERR and position-force controllers suffer from the deficiency that not all information is measured and used for control. This greatly restricts the search space for optimal controllers (Christiansson, 2007a). 4-Channel Control is regarded to yield superior performance

(Christiansson, 2007a; Aliaga, 2004; Lawrence, 1993). However, as position-force control and 4/5-Channel controllers require force sensors at the slave side to function, PERR-control is often favoured in environments where system robustness is key (e.g. space and nuclear applications). If one decides to place delicate electronics (force sensor) on the slave side, it is logical to decide in favour of the 4-channel controller instead of the force-position controller. The trade-off between a PERR-controller and a 4-channel control is a trade-off between performance, costs and system reliability.

For this research the most basic controller was chosen, the PERR-controller, for two main reasons:

- Implementation of 4-channel control is restricted by practical issues. 4-Channel control requires two force sensors, one on the master side and one on the slave side. However, the only available force sensor is reserved to measure the human's input on the master side. As a fundamental part of this study, we want to understand how the human deals with different qualities of force feedback. Notice that Christiansson's implementation of 4-channel (or 5-channel) control was realized by placing the force sensor on the master side, while a force estimate of the slave side was made based on the series-elastic principle. As series-elastic force sensing requires some kind of stiffness, which is by definition low bandwidth, series-elastic force sensing is inherently not suitable to measure the high-frequency force information.
- This research is governed by the remote handling work in the context of ITER that is being performed at FOM Rijnhuizen. Their remote handling philosophy is to a large extend based on the experiences at ITER's predecessor JET, in Oxford UK. At JET, full remote handling of the plant is being performed for many successful years, in which complicated tasks (e.g. welding, cutting) have been performed using a PERR-controlled manipulator (the Mascot). For ITER, would it be interesting to enhance this PERR-controlled manipulator by, for example, shifting to different control strategy?

While the PERR-controller itself suffices the low-frequency force feedback requirement of this research, high-frequency information is not displayed by a PERR-controller. The high-frequency information is measured by an accelerometer placed on the slave side.

Paragraph A3.1 discusses the realization of the low-frequency feedback, based on PERR-controller. In paragraph A3.2 it is investigated whether it is possible to improve the low-frequency feedback, by adding feedback from the series-elastic force measurement. The high-frequency force feedback information is discussed in paragraph A3.3. Finally paragraph A3.4 shows an overview of the realizations of the force feedback conditions.

A3.1 Low-Frequency Force Feedback

The low-frequency force feedback is created using a PERR-controller. A position servo on the slave side and a position servo on the master side communicate the shared reference position, while a controller minimizes the error between the flow variables, the master and slave position, using manually tuned PD-controllers. Fig. A19 shows a control scheme for a PERR-controller (Aliaga et al., 2004) for a single degree of freedom. The PD-controllers are denoted as *C* in Fig A19. Notice that the P- and D-gains, respectively K_p and K_d, are not necessarily equal for master and slave. F_{human} and F_{env} denote the external contact forces on the manipulator of respectively the human operator with the master device, and the environment with the slave device. From the control scheme, it is clearly seen that not only the master and slave impedances (Z_{master} and Z_{slave}) influence the manipulator positions, but the impedance of both the environment and then human (Z_{env} and Z_{human}) are in the loop as well.

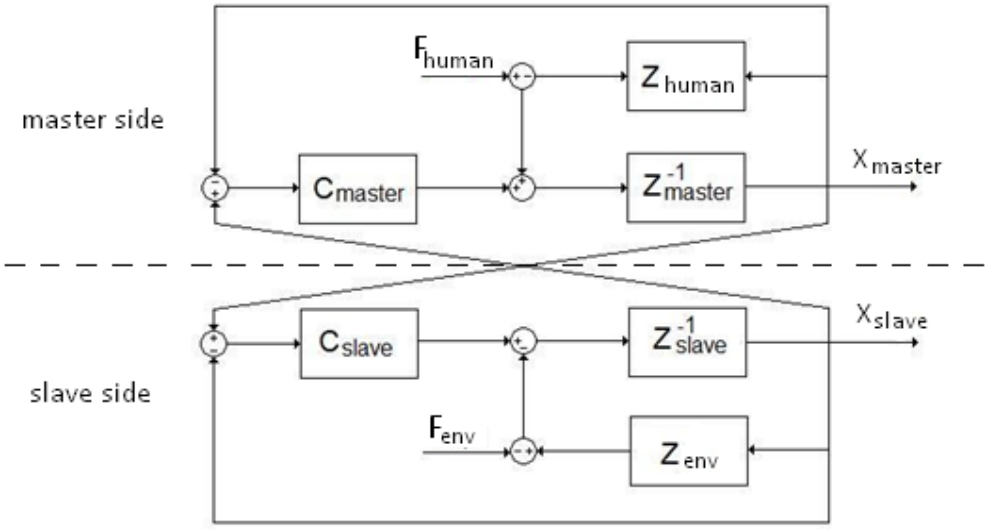


Figure A19: PERR-control scheme, adapted from Aliaga et al., 2004

A detailed analysis of the PERR-controller is described in Appendix A4. However, a first intuitive performance measure of the low-frequency feedback is the performance expressed in terms of stiffness. What is the reflected stiffness at master side, when the slave is in contact with a rigid wall? This can be analytically calculated (see Appendix A4), but this can also be easily measured. While the slave is fixated, the master is slowly moved in a certain direction. The forces are measured using a force sensor, while the manipulator itself tracks the position.

Fig. A20-22 respectively show plots of the master's position vs force for the x-direction, the y-direction and the Θ-direction. The solid line represents the measured (force) signal. The dashed line represents a first-order fit of the measured signal, created using a least square error optimization procedure. The results are shown in Table 2. Notice that the variation in y-direction is relatively high. This is due to the fact that a movement exclusively in y-direction is hard to create.

Table 2: PERR-controller stiffness

Degree Of Freedom	Stiffness Mean (Standard Deviation)
x	449 (7.5) N/m
Y	380 (64) N/m

Θ 261 (13) Nmm/rad

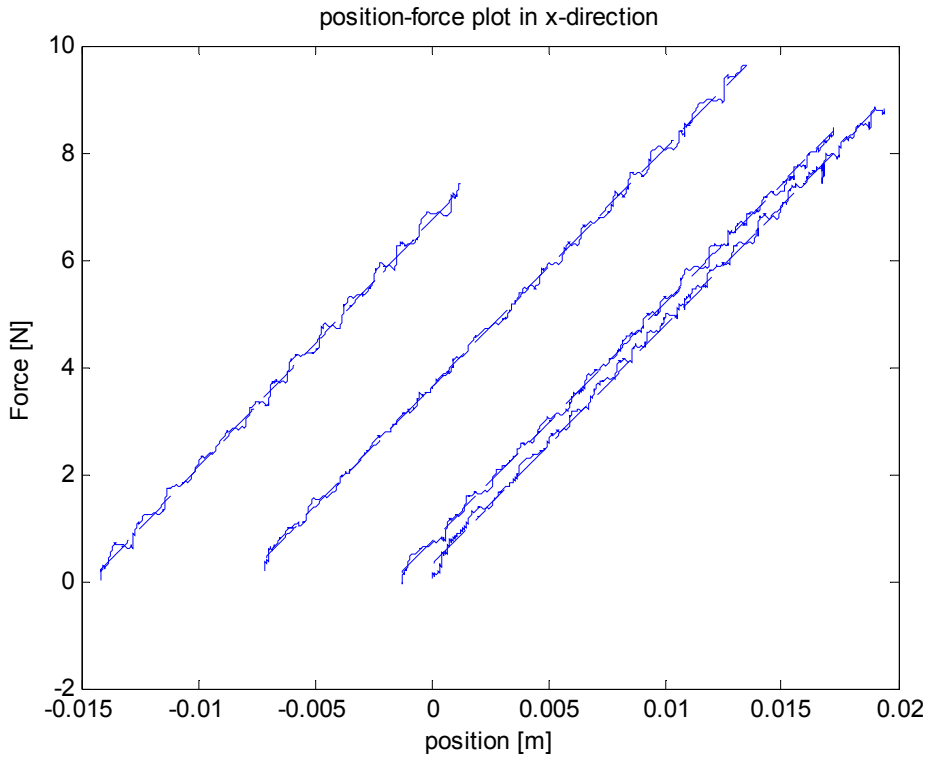


Figure A20: PERR-controller stiffness in x-direction

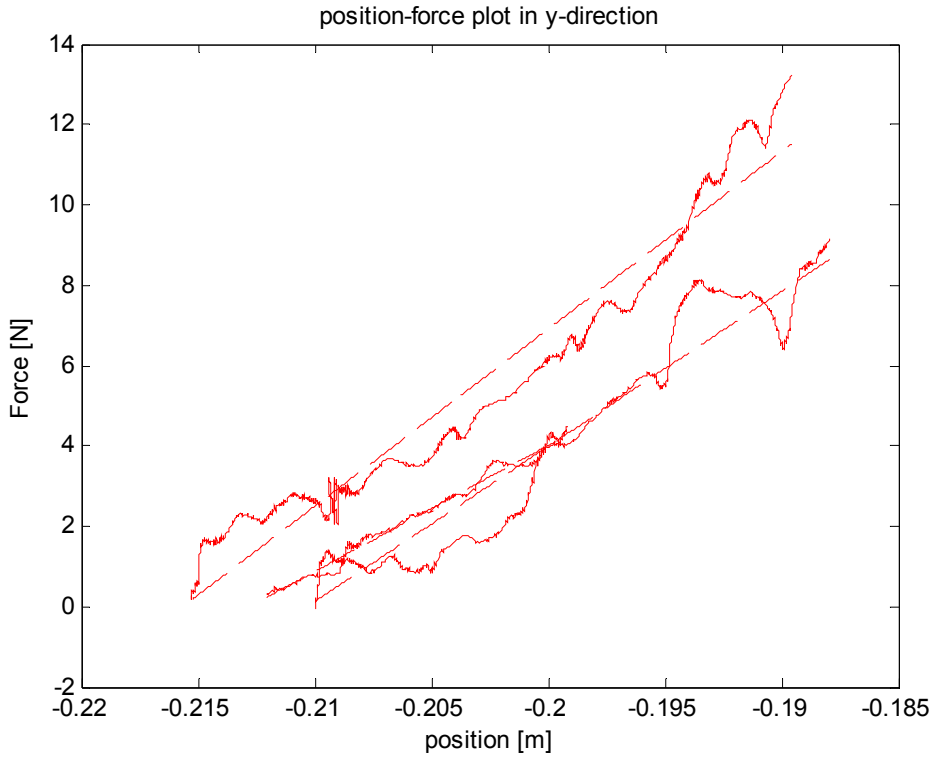


Figure A21: PERR-controller stiffness in y-direction

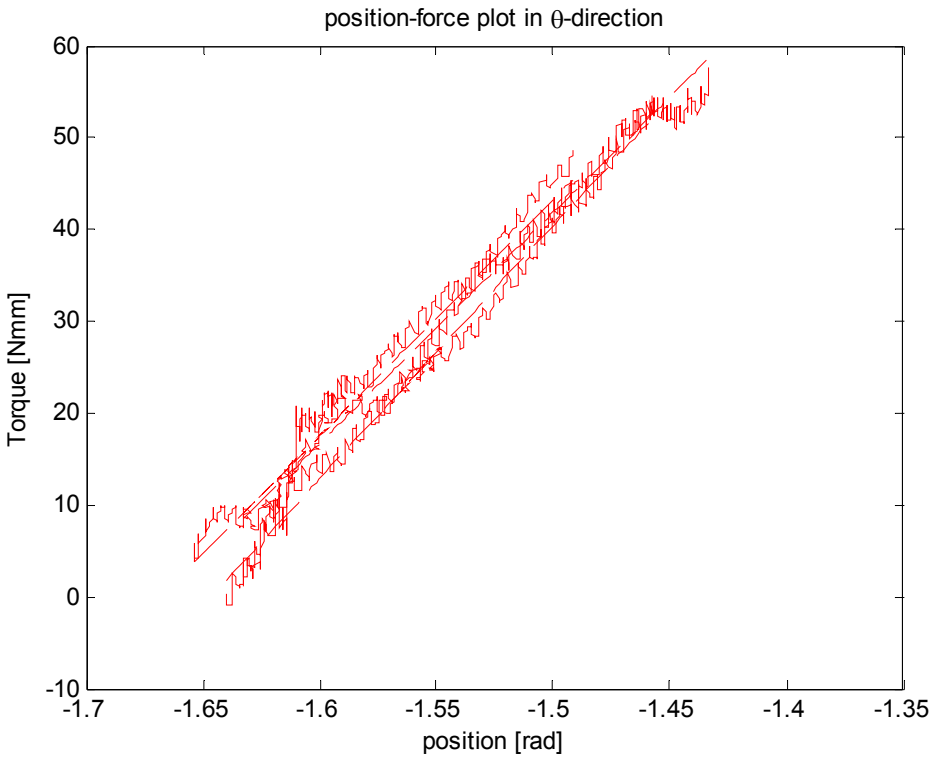


Figure A22: PERR-controller stiffness in Θ-direction

The Matlab code (see <*\Experiments\PlanarSetup\LF_feedback\AnalyzeStiffness.m>.

```
% Script to analyze the stiffness of the PERR-controller and the
% PERR-controller enhanced with low frequency force feedback
%
% Author: Jeroen Wildenbeest
% Created: March 17, 2010

clear all
close all
clc

DoF = 'theta'; % degree of freedom to analyze, defines data folder
trials_to_analyze = [2 3 4 6 7 8]; % vector containing the trials to
analyze
amount_of_trials = length(trials_to_analyze);

r = 45; % [mm] radius from force sensor mounting plate to master's point of
rotation

if DoF == 'x'
    i = 1;
elseif DoF == 'y'
    i = 2;
else
    i = 3;
end
```

```
for TrialNr = 1:amount_of_trials

    close all
    clear L time F_sensor X_m n RawData

    eval(['load ' DoF '\' num2str(trials_to_analyze(TrialNr))])

    fs = 1000;                      % Sampling frequency
    h = 1/fs;                       % Sample time
    [L, dummy] = size(RawData);     % L = length data vector

    time = 0:h:(L-1)/fs;            % Time vector

    if DoF == 'x'
        F_sensor = -RawData(:,i);
        X_m = RawData(:,3+i);
    elseif DoF == 'y'
        F_sensor = -RawData(:,i);
        X_m = RawData(:,3+i);
    else
        F_sensor = RawData(:,i) + r.*-RawData(:,1) ;
        X_m = -RawData(:,3+i);
    end

%     figure(22)
%     plot(time,RawData(:,i),'r'), hold on
%     plot(time,r.*-RawData(:,1),'g')
%     plot(time,F_sensor,'b')
%     title('Master torque vs time')
%     xlabel('time [s]')
%     ylabel('Torque [Nmm]')
%     legend('T_{\theta}','F_{x}*r','T_{\theta} + F_{x}*r')

    figure(21)
    linehandle = line(time,F_sensor); hold on
    title('Master torque vs time')
    xlabel('time [s]')
    ylabel('Torque [Nmm]')

    for n = 1:2
        [dummy dummy index(n,TrialNr)] = ginputdata(linehandle,F_sensor);
    end

    eval(['x_' num2str(TrialNr) ' =
X_m(index(1,TrialNr):index(2,TrialNr));']) % puts the data of a single
trial into one column
    eval(['F_' num2str(TrialNr) ' =
F_sensor(index(1,TrialNr):index(2,TrialNr));']);

end

close all

%% 1st order data fit

% set options
options = optimset('lsqnonlin');
```

```
options =
optimset(options,'display','final','diffminchange',0.001,'MaxIter',10000,'M
axFunEvals',10000);

% fit: y = a*x + b
%       a    b
lb = [100 -100]; % lower bound parameter vector
ub = [2000 3000]; % upper bound parameter vector
x0 = [200   40]; % initial guess parameter vector

% normaliseren van de parameters
norm_vector = x0;
lb_norm = lb./norm_vector;
ub_norm = ub./norm_vector;
x0_norm = x0./norm_vector;

for TrialNr = 1:amount_of_trials
    eval(['fit(:,TrialNr) = lsqnonlin(@error_fun, x0_norm, lb_norm,
ub_norm, options, norm_vector, x_' num2str(TrialNr) ', F_' num2str(TrialNr)
');'])
end

fit(1,:) = fit(1,:)*norm_vector(1);
fit(2,:) = fit(2,:)*norm_vector(2);


%% Plot

cc = {'b','b','b','r','r','r','r'};
cc2 = {'b--','b--','b--','r--','r--','r--','r--'};

for TrialNr = 1:amount_of_trials

    eval(['F_fit = fit(1,TrialNr)*x_' num2str(TrialNr) '+
fit(2,TrialNr);']);

    figure(22)
    eval(['plot(x_' num2str(TrialNr) ', F_' num2str(TrialNr) ',
cc{TrialNr})']), hold on
    eval(['plot(x_' num2str(TrialNr) ', F_fit, cc2{TrialNr})']), hold on
    % axis([.11 .15 0 5])
    title('position-force plot in x-direction')
    xlabel('position [rad]')
    ylabel('Torque [Nmm]')
    %legend('PERR-control trial 1','PERR-control trial 2','PERR-control &
LF feedback trial 1','PERR-control & LF feedback trial 2')

    clear F_fit
end

%% Calculate mean & standard deviation

fit
PERR_stiffness = mean(fit(1,1:amount_of_trials/2))
std_PERR_stiffness = std(fit(1,1:amount_of_trials/2))

LF_stiffness = mean(fit(1,amount_of_trials/2+1:amount_of_trials))
std_LF_stiffness = std(fit(1,amount_of_trials/2+1:amount_of_trials))
```

A3.2 Improved Low-Frequency Force Feedback (Series-Elastic)

In general, telemanipulators are optimized to what is known as 'transparency'; an optimization of the telemanipulator towards a virtual rod with infinite small mass, and infinite large stiffness (Drudgane et al., 1989). However, with respect to this optimization many trade-offs have to be made. For example, a higher stiffness often means a higher mass (e.g. see industrial robots). As a result, telemanipulators are relatively compliant compared to their industrial counterparts. The telemanipulator used at JET for example, yields a displacement of 10 mm at a load of 12 kg (Galbiati et al., 1991); an unacceptable inaccuracy for most industrial robots.

However, this inherent compliancy could actually be useful: Would it be possible to exploit the compliancy to yield a force estimate based on the series-elastic principle (Pratt and Williamson, 1995)? This force can then be used to enhance the master's force feedback, in principle adding a third channel (a force channel), but without the need for any delicate electronics like force sensors. The only additional hardware required is an additional set of encoders on the slave side. For the Munin, two encoder sets have already been implemented; one set on the motor axis, one set on the joint axis.

For the hardware changes as described in Appendix A1, can the force estimate be used for control to increasing the fidelity of the low-frequency forces? What is the increase in stiffness at the master sided compared to the common PERR-controller? But moreover, would this improved low-frequency force feedback based on series-elastic force sensing, result in a task performance increase?

The series-elastic force estimate times a gain is added to the existing PERR-controller master force (see Fig. A23). The gain is hand-tuned to guarantee system stability. The potential improvement of the controller by adding series-elastic based force feedback have been analyzed separately for the three main degrees of freedom (x, y and Θ). For the Matlab code see Appendix A3.1.

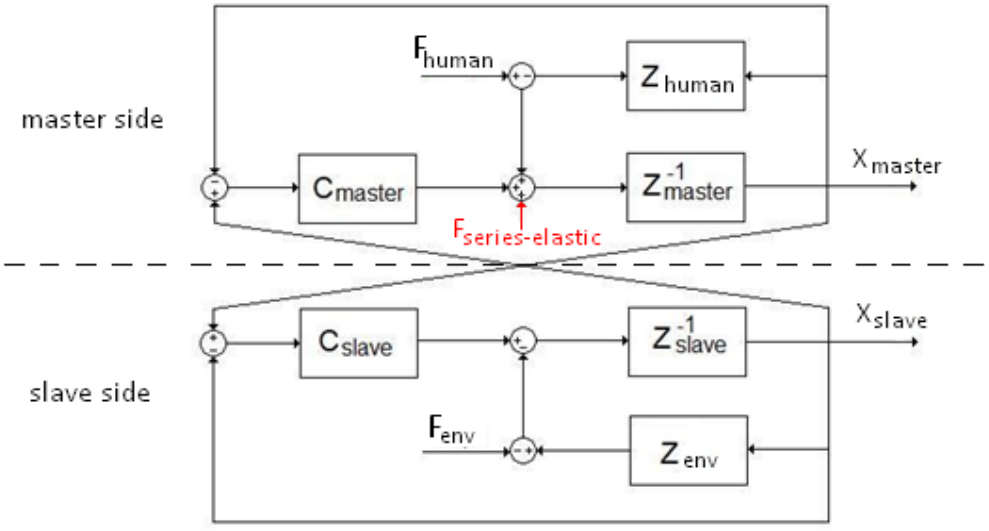


Figure A23: Series-Elastic Force Estimate Implemented in a PERR-control scheme

x-direction

Fig. A23 shows a position force-plot in the x-direction. The blue lines represent measurements performed under PERR-control, while the red lines represent the stiffness for PERR-control with additional series-elastic force (PERR-SE-control). The stiffness has been determined using a first-order least square error optimization procedure. Notice that the PERR-controller is different from the controller as used in the actual experiment (described in Appendix A3.1).

The PERR-stiffness is equal to mean (std) = 269.5 (7.1) N/m over four measurements. The PERR-SE-stiffness is equal to mean (std) = 378.0 (20) N/m over four measurements. Fig. A24 shows the forces as calculated by the PERR-controller and the SE-controller, as well as their sum. It can be seen that the SE-controller increases the reflected force to the master side for the x-direction. The dashed yellow line shows the output limits as implied by the software.

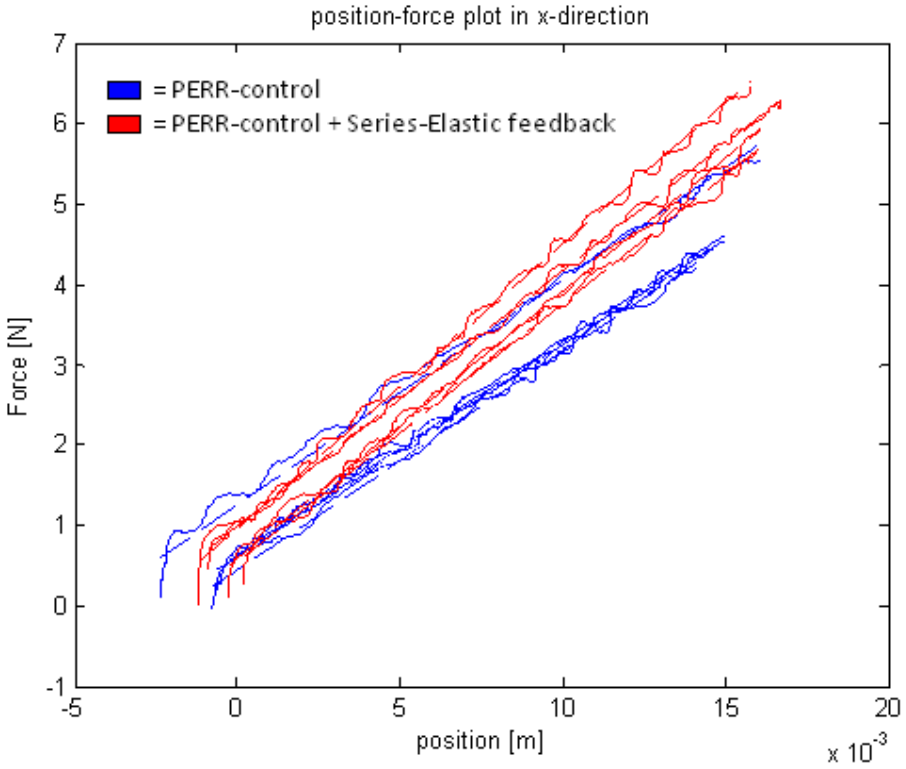


Figure A23: PERR- and PERR-SE-stiffness in the x-direction

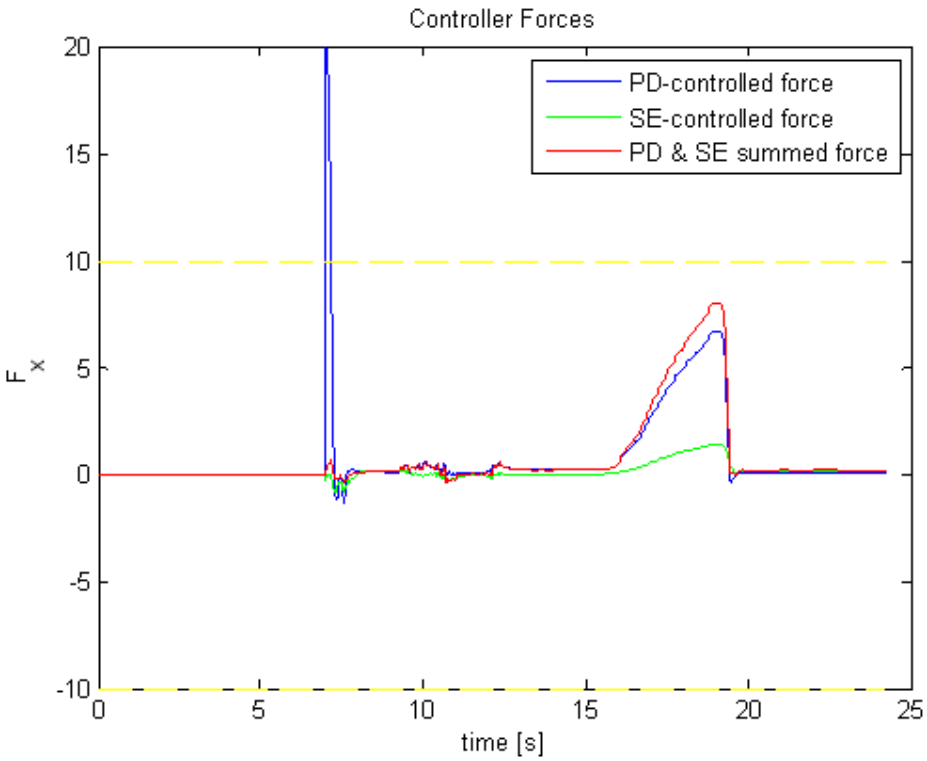


Figure A24: PERR- and SE- control forces in x-direction

y-direction

Fig. A25 shows a position force-plot in the y-direction. The PERR-stiffness is equal to mean (std) = 230.2 (3.1) N/m over four measurements, while the PERR-SE-stiffness is equal to mean (std) = 250.4 (4.9) N/m. Fig. A26 shows the forces as calculated by the PERR-controller and the SE-controller, as well as their sum. It can be seen that the SE-controller increases the reflected force to the master side for the y-direction.

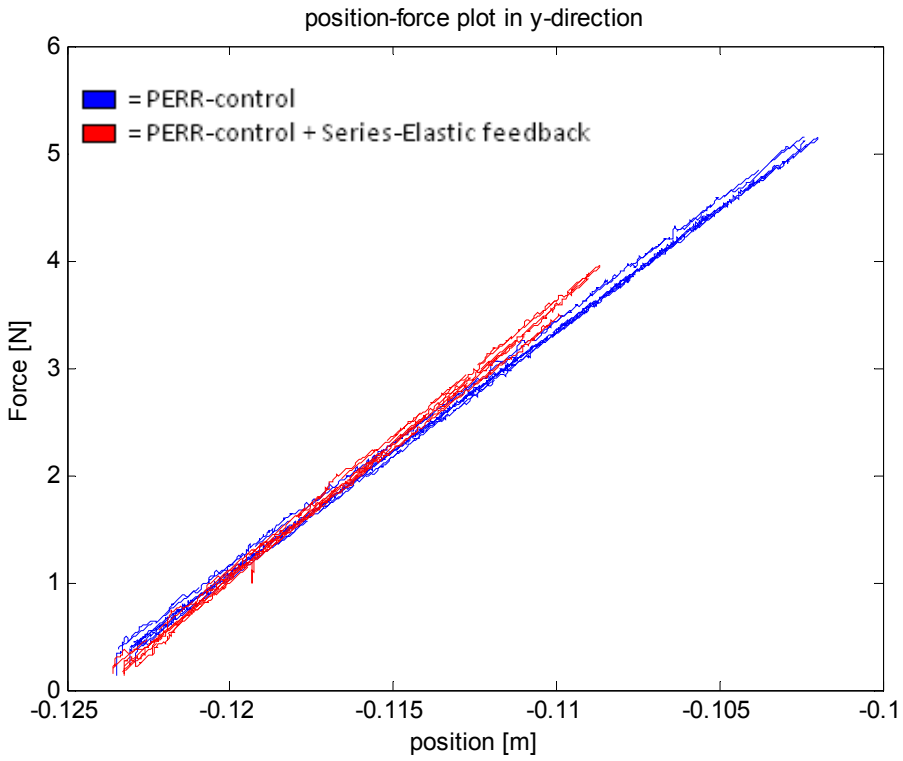


Figure A25: PERR- and PERR-SE-stiffness in the y-direction

theta-direction

Fig. A27 shows a position force-plot in the Θ-direction. The PERR-stiffness is equal to mean (std) = 386.1 (57.7) N/m over four measurements, while the PERR-SE-stiffness is equal to mean (std) = 1086 (140) N/m. Fig. A28 shows the forces as calculated by the PERR-controller and the SE-controller, as well as their sum. It can be clearly seen that the SE-controller increases the reflected force to the master side for the Θ-direction.

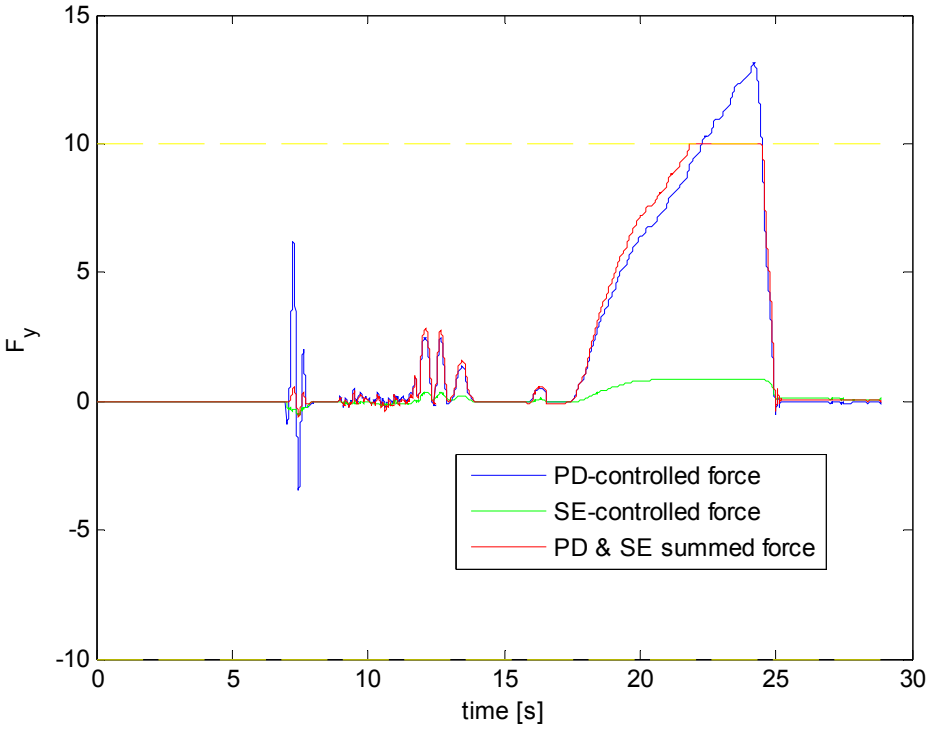


Figure A26: PERR- and SE- control forces in y-direction

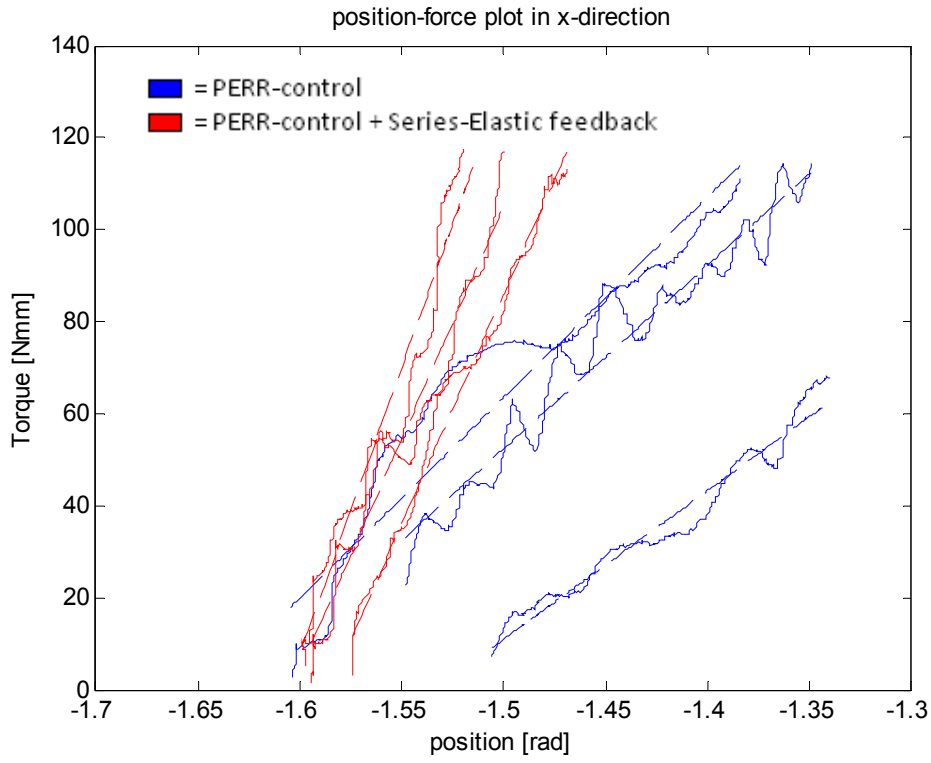


Figure A27: PERR- and PERR-SE-stiffness in the Θ-direction

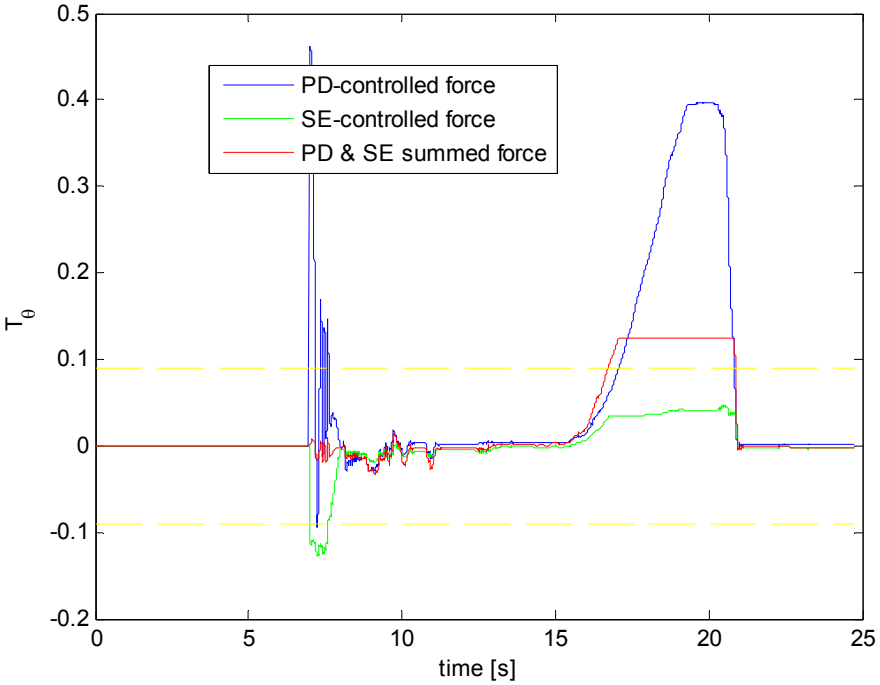


Figure A28: PERR- and SE- control forces in Θ-direction

Conclusion

It has been shown that the series-elastic force signal can be used to increase the stiffness of the manipulator. In the Θ-direction a significant increase of the master stiffness is realized (approx. 180%). For the x- and y-direction however, the increase in stiffness for the PERR-SE controller compared to the PERR-controller is far less (respectively approx. 40% and 7%). It is often stated that humans not able to distinguish differences below 10% of the current stimulus level (*the Weber fraction*, Gaydos, 1958). Therefore, is not expected that such a small increase will result in significant performance increase and hence the series-elastic based force feedback is not further investigated.

A reason for the unsuccessful implementation of the series-elastic force sensing can be found by looking at the slave encoder signals at joint level. It seems that the order of the control signal is in the same order as the error due to hysteresis (possibly caused by friction). Fig. A29 and A30 show plots of the error ($d\Theta = \Theta_{joint} - \Theta_{motor}$) of respectively the first and second joint of the slave device (α_1 and α_2, Fig. A18). Using the master, the slave device is moved +x direction, in -x direction, +y direction, in -y direction while being released meanwhile, and finally in +x, –x, +y and –y with a fixated slave in that exact order.

The first four movements cannot be distinguished very well. The control signal shows four horizontal plateaus; four from the actual movements, four created by an error (possible friction) when the master is released. This means the error is approx. 0.02-0.03 rad, while the control signal is in the same order. As a result, increasing the series-elastic feedback gains, will not only increase the force feedback, but will also increase the error. Hence the feedback signal, does not necessarily converge, which may lead to instabilities.

Also notice that the control signal is only one order higher than the resolution of the motor encoders. The joint encoders (ScanCon 2MC-H5000) have a 20000 ticks per revolution, which equals 2*pi/20000 = 3.1e-4 = 0.000314 rad/tick. The motor encoders (Agilent HEDL-5500) have a resolution of 5000 ticks per revolution, times the capstan gain(5.89) yields approx. 30000 ticks per revolution on joint level, which equals 2*pi/30000 = 2.1e-4 rad/tick.

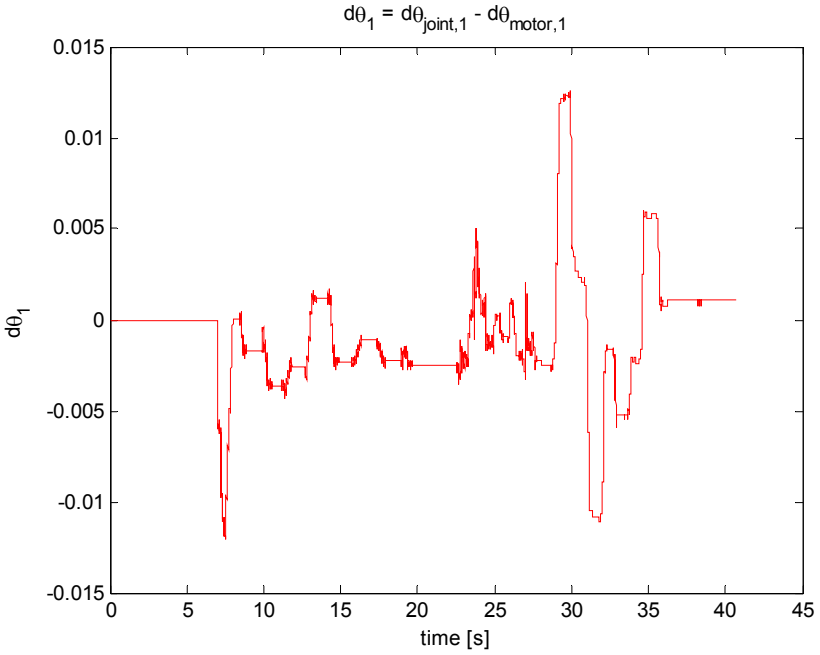


Figure A29: Error between joint and motor encoder for the first slave joint

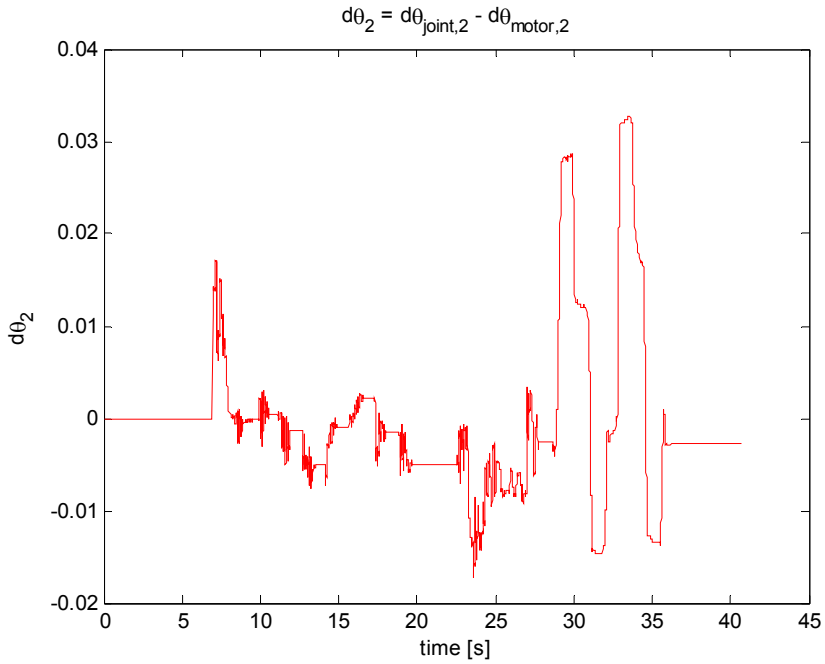


Figure A30: Error between joint and motor encoder for the second slave joint

A3.3 High-Frequency Force Feedback

High-frequency tool-mediated vibrations (>30 Hz) provide rich information about an object's properties, including geometry, texture, and material composition (Kuchenbecker, 2006). But does providing high-frequency information yield a performance increase?

The high-frequency signal are displayed to the human operator using an accelerometer (see Fig. A31). This accelerometer is mounted on the slave and measures the contact accelerations from the environment. The signal is filtered with a 4th order Butterworth high-pass filter with a cut-off frequency of 30 Hz. If the accelerometer is excited above the tuned threshold of 0.015 m/s^2, a signal is generated on the master side. The gain on the high-frequency signal is hand-tuned to provide an intuitive feeling.

This signal contains a 50 Hz and 100 Hz sinusoid; frequencies typically occurring upon hard-hard contact. The signal is displayed for the duration of one 50 Hz period (20ms), and is superimposed on the PERR-signal. The impulse can only occur once every 50 ms. Fig. A31 shows the Matlab Simulink implementation of the high-frequency signal generator. Fig. A33 shows a plots of the modeled signal. The amplitude of the signal is proportional to the impact acceleration. This is illustrated by Fig. A34, which shows the realization of the modeled signal in Mastlab Simulink for two different amplitudes. Finally, Fig. A35 shows the measured output at the tip of the master device.

Daniel and McAree argue that the natural partition of the haptic modality, contains a highly energetic part below roughly 20 Hz and an information conveying part above roughly 30 Hz (Daniel and McAree,). Fig. A36 illustrates this: the average force is 0.3 N over 13 secs. The high frequency signal has a amplitude of 0.05-0.6 N, for only 20 ms. This means that the energy content of the high frequency signal is by definition low energetic.

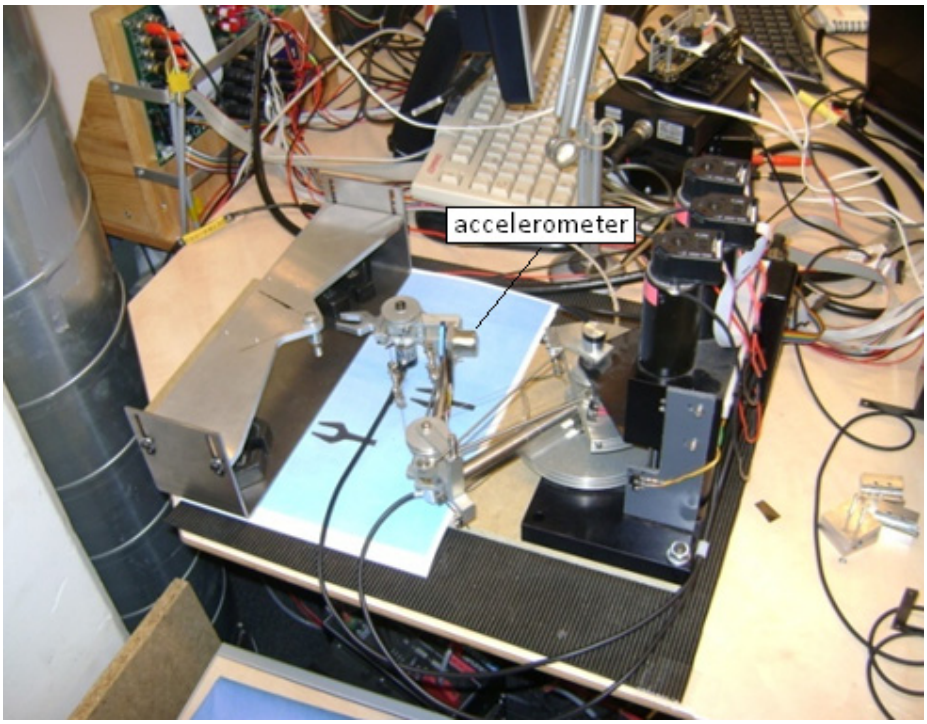


Figure A31: The accelerometer as mounted on the slave device

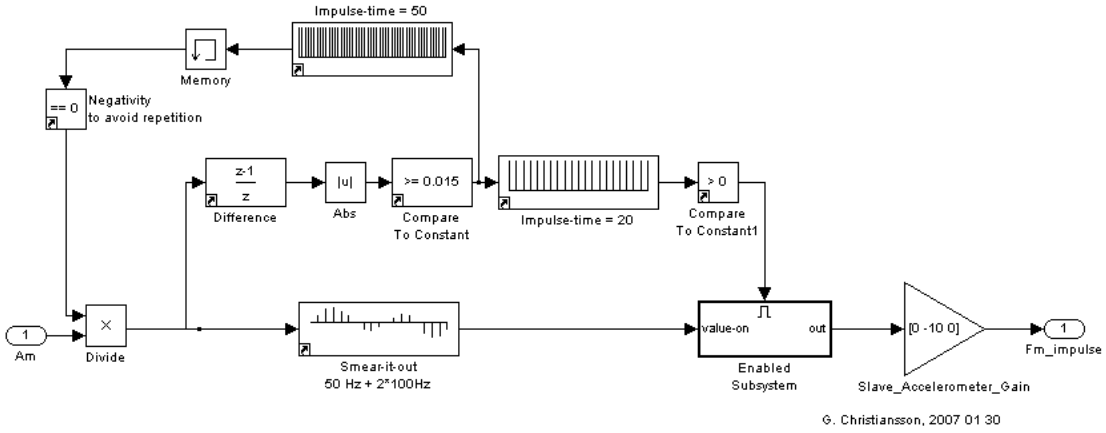


Figure A34: Matlab Simulink model of the high-frequency signal generator, adapted from Christiansson, 2007a

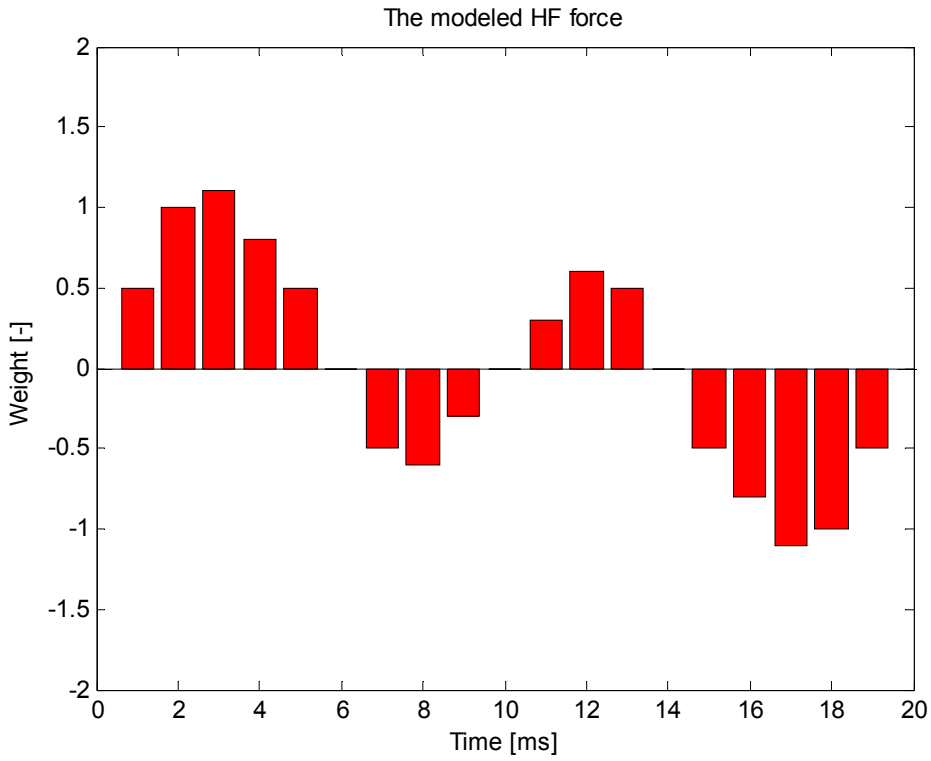


Figure A33: The modeled high-frequency signal, containing a 50 Hz and a 100 Hz sinusoid

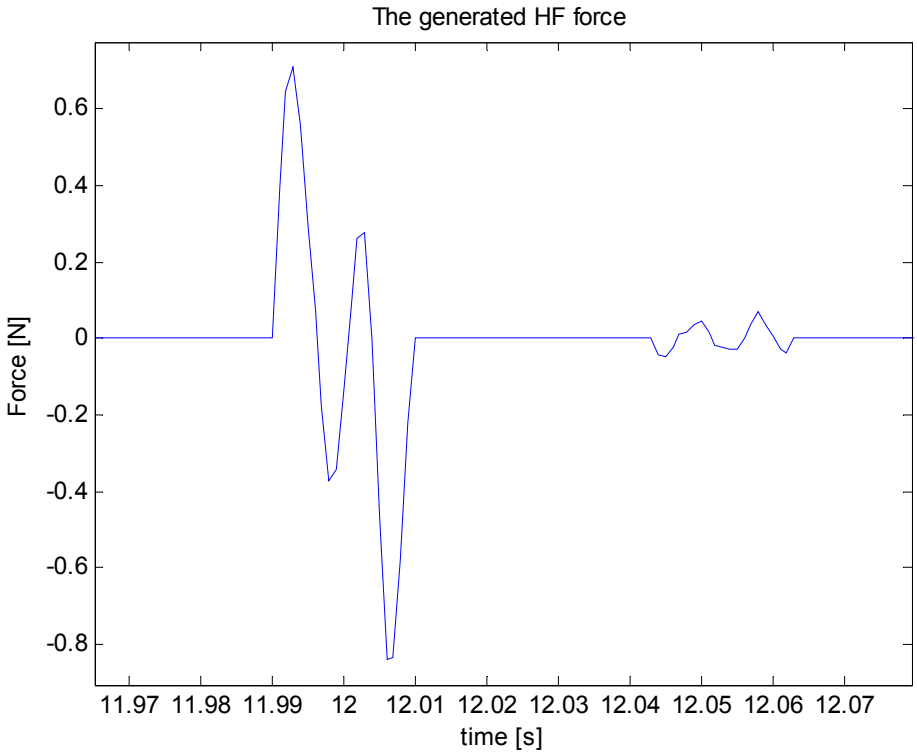


Figure A34: Two realizations of the high-frequency signal, with a large and a small amplitude

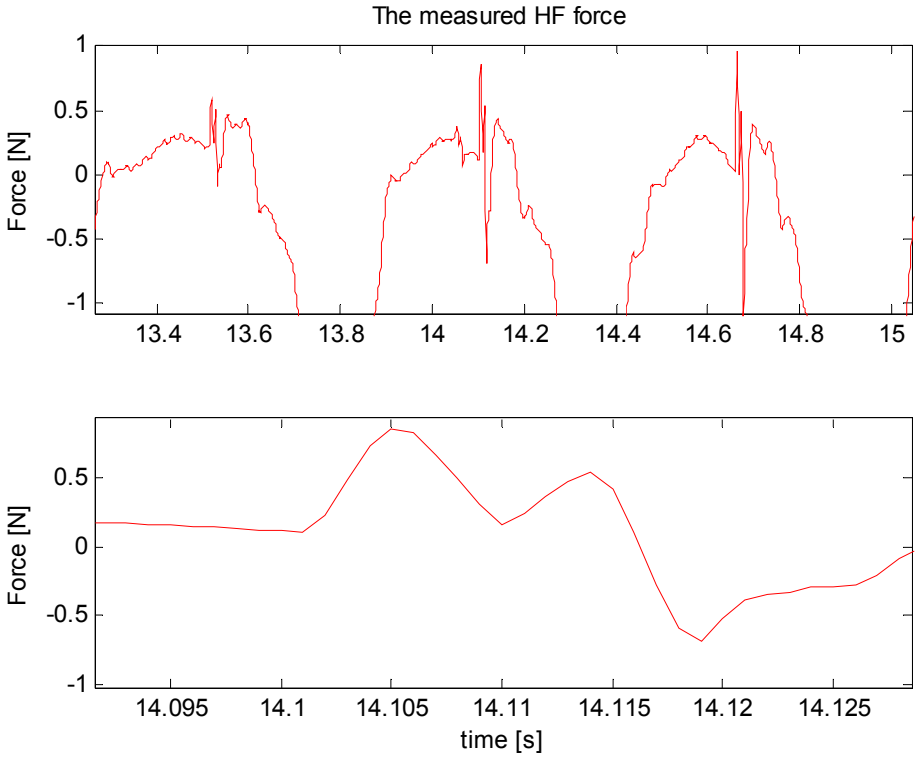


Figure A35: The measured high-frequency signal at the tip of the master device

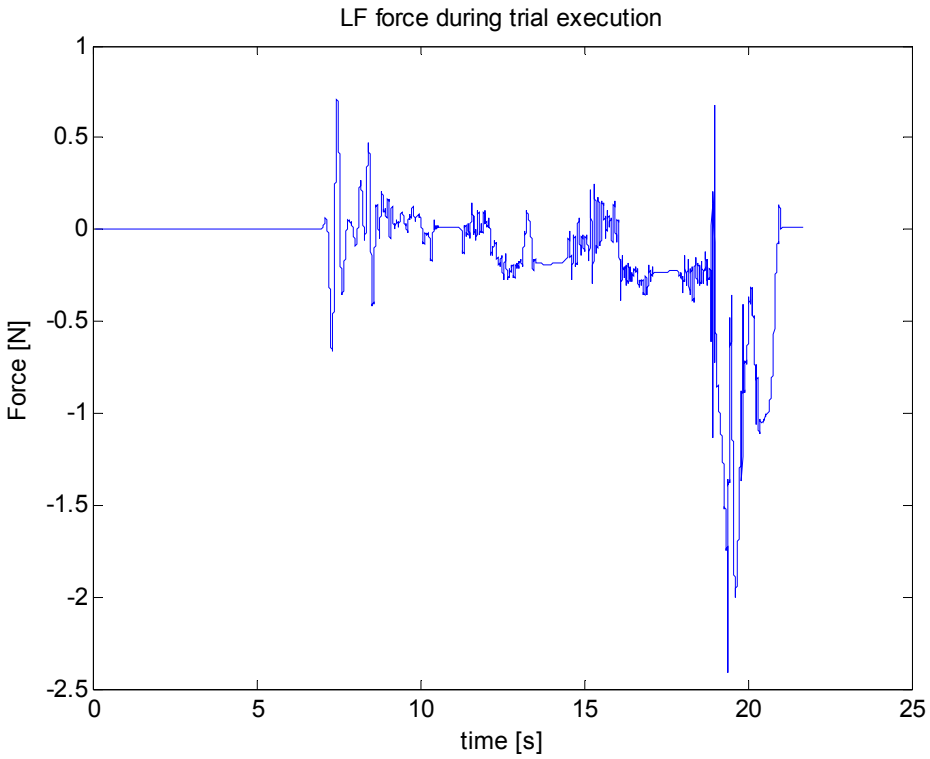


Figure A36: A typical force profile during execution of the bolt-and-spanner task (subject 9, C3, trial 19)

A3.4 Controller Realization

In the human factors experiment four different haptic feedback conditions have been defined, namely Numb, LF, LFHF and HandsOn. The conditions are realized by applying a different control algorithm. Fig. A37 shows the controller as realized in Matlab Simulink.

The inputs of the controller are colored yellow, the positions and velocities of both the master and the slave and the accelerometer signal. The outputs, the master and slave forces, are colored red. The P- and D- gains are colored blue. The high-frequency part of the feedback is modeled in the green block.

The Numb-controller is realized by setting the slave-to-master gains to zero, practically creating a one way servo from master-to-slave. The affected gains are shown in the orange block. The LF-controller is the straightforward PERR-controller, while for the LFHF-controller the high-frequency part is enabled (the green block). For the HandsOn controller all control gains are set to zero (the blue colored blocks); the Munin is used as a data logger.

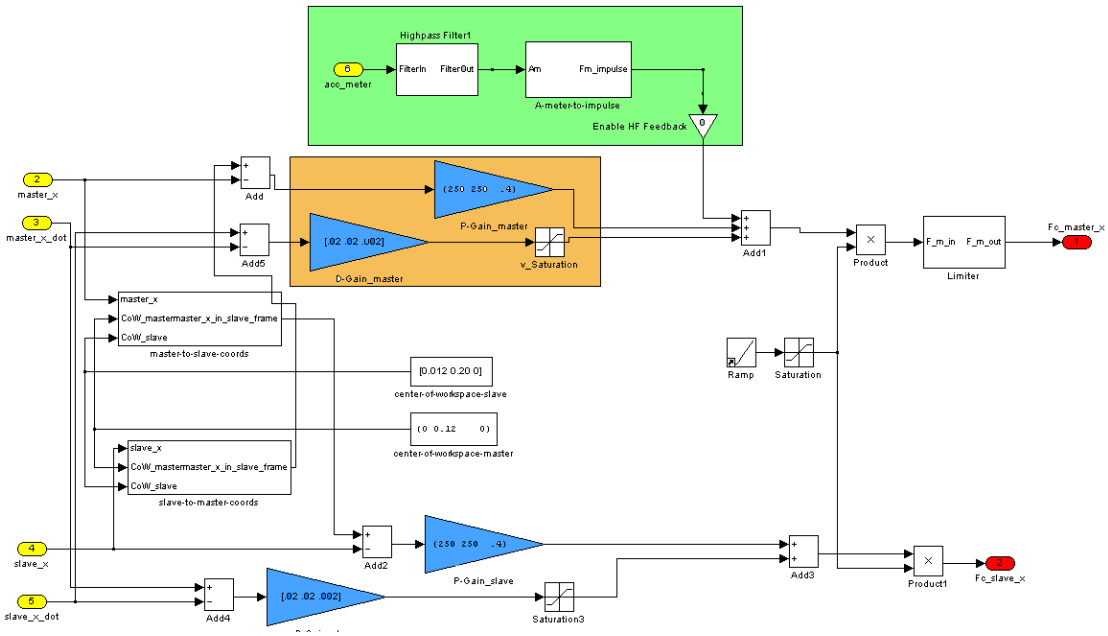


Figure A37: The controller as implemented in Matlab Simulink

A4 Performance Characteristics

A common method to analyze the performance of a telemanipulator is the two-port network analogy. This framework is derived from a common architecture to analyze electric circuits with bidirectional energy flow, known as the two-port network, four-terminal network or quadripole. The system itself is modeled as a black box, while the inputs and outputs of the system, the so-called power variables, are analyzed. One side of the system is defined by current I_1 and voltage V_1 the other side is defined by I_2 and V_2. The terminals have to suffice a requirement called the port condition: $I_{in} = I_{out}$.

If this approach is projected onto teleoperators, the energy flow can be represented by forces and velocities (or integrated velocity, position), whereas:

- current $I \sim$ velocity $\dot{x}$
- voltage $V \sim$ force F
- and the port condition becomes : $\dot{x}_{in} = \dot{x}_{out}$

Telemanipulator Model

Fig. A38 shows a schematic representation of the two-port network method. The dynamic relation of the power variables $\dot{x}$ and F is defined as impedance *Z*: $Z = F / \dot{x}$. The master device, the slave device, the human operator and the environment can all be described as an impedance. As the theory expresses its variables in the frequency domain, the Laplace definition of impedance is given by the transfer function $Z(s) = F(s) / \dot{x}(s)$. Impedance is often parameterized by the components of a (linear time-invariant!) mass-spring-damper system, thus $Z(s) = F(s) / \dot{x}(s) = ms + b + k / s$.

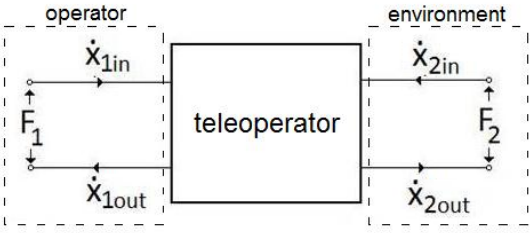


Figure A38: The two-port network analogy

To describe the relation of master, slave and environment, often the hybrid notation is used. The hybrid notation is described by the following equation:

$$\begin{bmatrix} F_{master} \\ \dot{x}_{slave} \end{bmatrix} = \underbrace{\begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}}_{H} \begin{bmatrix} F_{slave} \\ \dot{x}_{master} \end{bmatrix} \tag{24}$$

Whereas the H-matrix for a PERR-controller is defined as:

$$H = \begin{bmatrix} \dfrac{(sZ_{master} + K)Z_{slave} + KZ_{master}}{sZ_{slave} + K} & \dfrac{K}{sZ_{slave} + K} \\ -\dfrac{K}{sZ_{slave} + K} & \dfrac{K}{sZ_{slave} + K} \end{bmatrix} \tag{25}$$

Fig. A39 shows a schematic representation of the hybrid notation as implemented in the two-port framework. The impedances of the master and the slave are incorporated in the transfer functions of the H-matrix (h_{ij}), as can be seen in equation (25). $F_{ext,master}$ are active forces applied by the human operator, while $F_{ext,slave}$ includes external (active) forces from the environment (e.g. friction).

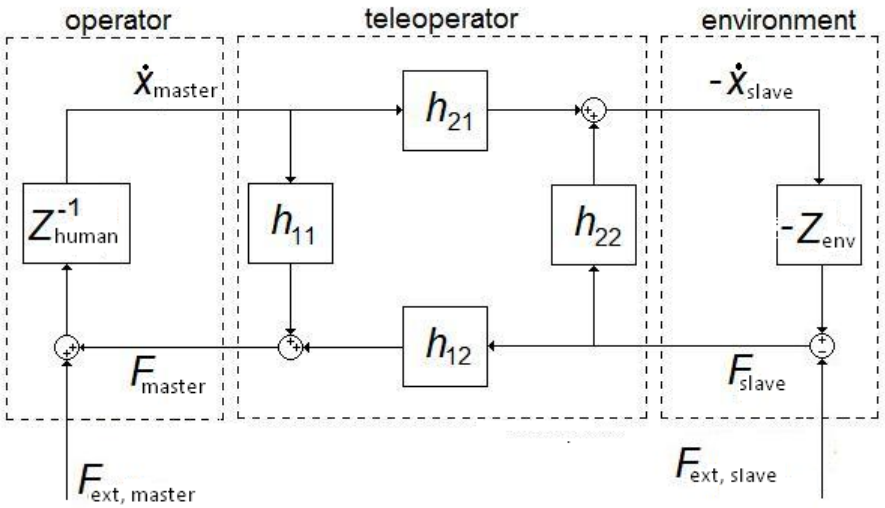


Figure A39: H-matrix representation of a telemanipulator, adapted from Christiansson, 2007a

The control gain (K) consists out of a proportional gain (K_P) and a derivative gain (K_D). Hence,

$$K = K_P + K_D s \tag{26}$$

While K_P and K_D are equal for master and slave, both gains are degree of freedom dependant. The gains have been hand tuned to guarantee system stability.

$$K_P = \begin{bmatrix} K_{P,x} & K_{P,y} & K_{P,\theta} \end{bmatrix} = \begin{bmatrix} 400 & 400 & 0.4 \end{bmatrix} \tag{27}$$

$$K_D = \begin{bmatrix} K_{D,x} & K_{D,y} & K_{D,\theta} \end{bmatrix} = \begin{bmatrix} 0.02 & 0.02 & 0.002 \end{bmatrix} \tag{28}$$

The master and slave are both modeled as a mass-damper system. For the system as described in Christiansson, 2007b, the master and slave's inertia and damping have been determined using a parametric model fit. The inertia and damping estimates of this study are based on those estimates. Notice that the intention of this performance analysis is to obtain an indication of the telemanipulator performance expressed in analytic metrics, not to obtain accurate data of the telemanipulator performance. For the latter, the setup should be identified using a procedure as described in Christiansson, 2004.

The master's mass has been slightly changed; the gripping interface has increased in length (more rotational inertia), but is also weighs less (90g to 61g). Hence, the mass in x- and y-direction is estimated at 230 g, while the inertia in Θ-direction direction is assumed not to change unchanged, namely 0.025 gm^2. The master's damping (4.5 Ns/m for x- and y-direction, 0.02 Ns/m for Θ-direction) is also assumed not to change.
For the slave device, it is assumed that the inertia and damping have not changed compared to the original system. Hence, the slave's inertia is estimated at 280 g in the radial direction and 0.025 gm^2 in the Θ-direction. The slave's damping is estimated at 6 Ns/m for x- and y-direction, and 0.04 Ns/m for the Θ-direction.

Notice that the inertia and damping for both master and slave are subjected to a large variation over the workspace.

Summarized the master is modeled as:

$Z_{m,xy} = m_{m,xy}s + b_{m,xy} = 0.23s + 4.5$, for the x- and y-direction (29)

$Z_{m,\theta} = m_{m,\theta}s + b_{m,\theta} = 0.025 \cdot 10^{-3} s + 0.02$, for the Θ-direction (30)

For the slave, the following equations apply:

$Z_{s,xy} = m_{s,xy}s + b_{s,xy} = 0.28s + 6$, for the x- and y-direction (31)

$Z_{s,\theta} = m_{s,\theta}s + b_{s,\theta} = 0.025 \cdot 10^{-3} s + 0.04$, for the Θ-direction (32)

Human Operator Model

The impedance model of the human operator specifically for this bolt-and-spanner task has been parameterized using a closed-loop frequency domain identification procedure (van der Helm et al., 2002). A more detailed description of the experiment can be found in Appendix B. $Z_{human} = m_{human}s + b_{human} + k_{human} / s$, whereas the values for m, b and k are shown in Table 3.

Table 3: Human Operator Impedance Properties

Property	PT_{mean} (std)	RT_{mean} (std)	$FT_{estimate}$
m [kg]	2.5 (0.1)	2.6 (0.4)	2.6
b [Ns/m]	38 (1)	28 (3)	28
k [N/m]	538 (130)	377 (45)	216

Environment model

The environment is defined as a constant stiffness. For a steel environment with E = 200 GPa, the stiffness can be estimated in the order of 1e10 N/m. Hence the environmental impedance is:
Z_{env} = 1e10 N/m.

Performance Analysis

Using this definition of the telemanipulator, the Munin's performance as a electro-mechanical device can be described by a variety of metrics. Christiansson's open source HapticAnalysis package provides a set of toolboxes (see http://www.hapticanalysis.org) which include the most commonly used metrics to assess the device performance and stability. The device performance and stability metrics as presented in this report are based on the metrics as defined in the HapticAnalysis

package. For a detailed description of the metrics see Christianson, 2007a. Table 4 shows an overview of the performance metrics independently calculated for x-, y- and Θ-directions. Notice that as the impedances for x- and y-direction are assumed equal, the performance is assumed equal.

Table 4: Model-Analytic Performance Metrics

Performance Metric	***x- and y-direction***	***Θ-direction***
m-to-s velocity tracking	2.4714	1.9999
s-to-m velocity tracking	2.3889	0.9140
s-to-m force tracking	2.0707	0.9525
m-to-s force tracking	2.3889	0.9140
m-to-s velocity bandwidth	42.4567	14.1246
s-to-m velocity bandwidth	Inf	Inf
s-to-m force bandwidth	42.4567	14.1246
m-to-s force bandwidth	Inf	Inf
scaling product	3.3557	0.9999
free air m	0.1355	-0.0001
free air b	11.2962	0.0469
free air k	-0.0269	0.0003
stiff contact m	0.2300	0.0000
stiff contact b	4.5200	0.0220
stiff contact k	399.7956	0.4006
transparency	67.5770	87.0882
Zwidth	30.5110	9.1954

It can be seen that the rotational performance (Θ) is inferior to the translational performance (x and y). This is inherent to the design of this device; it is not very well suited for rotations (Christiansson, 2007b). However, the point of rotation from the task's point of view, is not the Cartesian endpoint of the manipulator, but the point of rotation of the spanner (see also Fig. D1). Due to the kinematic relation that exist between spanner point of rotation and the manipulator Cartesian endpoint (see Appendix D1), a rotation around the spanner's point of rotation causes the manipulator to translate. For that reason, it is assumed that the performance metrics for the manipulator translation are also applicable around the spanners point of rotation.

It should be acknowledged that these kind of model-analytic approaches only partially include the two utmost elements of a telemanipulator as a functional system; the human operator and the environment. More specific, these approaches do not explicitly account for the task, which the operator is trying to complete in the (remote) environment. This means that device performance metrics only have a marginal, difficult to concretize relation with the actual performance of the task.

Stability Analysis

The HapticAnalysis package defines three metrics to determine the system's stability, namely Closed Loop Stability, Absolute Stability and Passivity. Multiple stability theorems are used, as each of these theorems has different advantages and disadvantages (Christiansson, 2007a).

A system suffices the closed loop stability requirements if the roots of the characteristic polynomial are negative or have negative real parts. Closed loop stability is assessed using the Nyquist and Root Locus methods. While closed loop stability criteria is rather strict, the Root Locus method allows to analyse only one variable at a time, while quantification of the phase and gain margins is only limitedly possible when the Nyquist criterion to telemanipulators. The latter is

caused by the fact that the telemanipulator open loop transfer function is not presented in the form of GK, whereas G is the system and K is the controller.

Absolute or stability means that the system is stable for all possible passive operators and environments. However, the stability conditions can be conservative and only one quantitative parameter is available.

The passivity theorem states that if each of the system components loses energy over time, the components together will also lose energy, and therefore be stable. While the graphical presentation of passivity is very informative, the stability conditions are most conservative of the described metrics.

Running the HapticAnalysis package yields that the telemanipulator suffices the criteria of all three stability metrics, and hence is stable. Notice that the stability analysis has been performed with the objective to verify the telemanipulator, rather than to use the analysis to optimize its performance.

Another interesting item to look at is the force feedback bandwidth to the human operator when the human is manually performing a task, using the Munin as a data logger (the HandsOn task).

Contact forces from the environment are transmitted through the spanner. The spanner is modeled as a impedance, having only a mass and a stiffness. The damping is assumed zero. Thus:

$$Z_{spanner} = m_{spanner} s + k_{spanner} \tag{33}$$

The spanner weighs 61 grams. The stiffness is based on the spanner's E-modulus and geometry. For an ideal spanner:

$$\hat{k}_{spanner} = \frac{A_{spanner} E_{spanner}}{L_{spanner}} \approx 5e7 \text{ [N/m]} \tag{34}$$

Now the spanner is not a rigid object; the spanner parts are interconnected bolts interfaces, which reduce the stiffness. Assuming the bolts reduce the stiffness with one order of magnitude:

$$k_{spanner} \approx 5e6 \text{ [N/m]} \tag{35}$$

Hence, for these parameters the force bandwidth is 2.2e3 Hz. Because the amount of uncertainties in this calculation it is said that the force feedback bandwidth for the HandsOn task is in the order of 1e3 Hz, which is beyond the bandwidth of the humans sensoric system (Brooks, 1990).

The Matlab Code (*\HapticAnalysis\matlab\RunMeToAnalyseMunin.m)

```
% Script to analyze the performance and stability of the
% PERR-controller
%
% Author: Jeroen Wildenbeest
% Created: June 5th, 2010
% Adapted from Christiansson's HapticAnalysis kit

clear all
```

```
close all
clc

% H-matrix converted from Maxima to Matlab
% (from HmatrixPERR_Kp_ZmZs.mc)
% Symbolical definition of the H-matrix
% Converted Thu Dec 15 14:54:00 2005
s = tf('s');


% x- and y-directions
% Mm = 0.23; % x-y
% bm = 4.5;
% Ms = 0.28; % x-y
% bs = 6;
% Zm = Mm*s + bm;
% Zs = Ms*s + bs;
% Kp = 400+.02*s;

% theta-directions
Mm = .000025;
bm = .02;
Zm = Mm*s + bm;

Ms = .00002;
bs = .04;
Zs = Ms*s + bs;

Kp = .4+0.002*s;


% see Christiansson 2004, "Measuring asymmetric haptic teleoperation device
% properties"


K = [0 -Kp/s 0 Kp/s ; % uitgaande van een velocity als power variable een
positite signaal geeft Kp ipv Kp/s
         0 Kp/s 0 -Kp/s];


%% H-matrix for PERR-controller:

% h11 =  ((s*Zm+Kp)*Zs+Kp*Zm)/(s*Zs+Kp);
% h12 = Kp/(s*Zs+Kp) ;
% h21 =          -Kp/(s*Zs+Kp);
% h22 = s/(s*Zs+Kp)   ;
% % Combine these transfer functions into a matrix:
% H = [h11,h12;h21,h22];


%% H-matrix Universal

h11 = ((Zm-K(1,2))*Zs - K(2,4)*Zm + K(1,2)*K(2,4) - K(1,4)*K(2,2)) /
((K(1,1)+1)*Zs + (-K(1,1)-1)*K(2,4) + K(1,4)*K(2,1));
h12 = - (K(1,3)*Zs - K(1,3)*K(2,4) + K(1,4)*K(2,3) - K(1,4)) /
((K(1,1)+1)*Zs + (-K(1,1)-1)*K(2,4) + K(1,4)*K(2,1));
h21 = - (K(2,1)*Zm + (K(1,1)+1)*K(2,2) - K(1,2)*K(2,1)) / ((K(1,1)+1)*Zs +
(-K(1,1)-1)*K(2,4) + K(1,4)*K(2,1));
```

```
h22 = - ((K(1,1)+1)*K(2,3) - K(1,3)*K(2,1) - K(1,1) - 1) / ((K(1,1)+1)*Zs +
(-K(1,1)-1)*K(2,4) + K(1,4)*K(2,1));

H = [h11 h12 ;
     h21 h22];

%% Human operator & environment properties
Mh_pos = 2.5;
Bh_pos = 38;
Kh_pos = 538;
Mh_force = 2.59;
Bh_force = 27.6;
Kh_force = 216;

Zh_pos = Mh_pos*s + Bh_pos + Kh_pos/s;
Zh_force = Mh_force*s + Bh_force + Kh_force/s;

% k_env
E_env = 200e9;
A_env = .01*.05;
L_env = .01;
k_env = (A_env*E_env)/L_env; %[N/m]
%(http://en.wikipedia.org/wiki/Young's_modulus)

% environmental properties
Ze = k_env/s;
%Ze = 100000/s; % 100 N/mm

%%
P = Haptic_performance(H)'
Spos = Haptic_stability(H,Zh_pos,Ze)
Sforce = Haptic_stability(H,Zh_force,Ze)

H12 = H(1,2);
SM_force_bandwidth = bandwidth( H12 + 1)

clf; hold on;
bode( H12+1 );

%% Bandwidth of the HandsOn task
% Script to determine the bandwidth of the HandsOn Task
 E_spanner = 200e9; % E-modulus steel
A_spanner = .006*.004; % sectional surface van de spanner
L_spanner = .1; % length of the spanner
k_spanner = (A_spanner*E_spanner)/L_spanner; %[N/m]
%(http://en.wikipedia.org/wiki/Young's_modulus)

m_spanner = 0.06;
k_spanner = k_spanner/10; % it is assumed that due to the bolts, the
stiffness of the spanner drops 1 order of magnitude.
b_spanner = 0;

Hspanner = 1/(m_spanner*s^2 + b_spanner*s + k_spanner);

bandwidth_handson = bandwidth(Hspanner);
bandwidth_handson/(2*pi);

%bode(Hspanner)
```

A5 A short how to for the Munin

The setup that drives the Munin constists out of two PCs. The first PC, the 'target PC', runs on a Mathworks xPC target real-time operating system and contains the interface cards with the telemanipulator hardware. The operating system is installed on the floppy drive. The second PC, the 'host PC', is a Windows XP based PC on which the models to control the manipulator are build and from which the work pc is operated.

To start, turn on both PCs. The target PC will show the xPC screen and will wait for further orders from the host PC.

1) On the host PC double-click on the control model you want to load. The controller used in this study is named 'Controller_Jeroen.mdl', found in the host PC directory C:\Documents and Settings\haptics\Desktop\Jeroen\Experiments\Controller.

2) Build the model and download to the Target PC by selecting Tools -> Real-time Workshop -> Build Model. (or press Ctrl+B). Notice that the model requires resources from the default directory (D:\haptics\planarsetup\realtime_software\basiscontroller\), and thus this directory needs to be set as the current directory.

3) Typing <+tg> in the Matlab command window will start the model. The model will finish running when its simulation time expires or when you type <-tg>.

4) Alternatively, the target PC can be controlled using the xPC Explorer on the host PC. Type <xpcexplr> in the Matlab command window to start the xPC Explorer. In the xPC Explorer, right click on the icon for the target PC (TargetPC1) displayed in the hierarchy and select 'Connect to Target'. Pressing the play icon will now start the model, pressing the stop icon will stop the model. The xPC Explorer can also be used to change parameters of the control model. For example, the controller gains can be tuned, well setting 'Enable HF Feedback' to zero disables the generation of high-frequency force feedback.

5) Finally data can be saved by executing 'save_OutputLog.m'.

Appendix B: Human Identification Experiment

As part of this thesis, an experiment has been performed to identify the (neuromuscular) properties of the human operator. The results of this human operator identification experiment have been used to analyze the performance and stability of the telemanipulator (see Appendix A4), and in the model to analyze the task performance (see Appendix C).

Parametric Model

For a first analysis of the Connected Telemanipulator System, the human operator is modeled as a linear time-invariant mass-spring-damper model, having only intrinsic properties (mass, damping and stiffness). Hence, for small displacements the human can be described by:

$$Z_{human} = \frac{F}{x} = m_{human}s^2 + b_{human}s + k_{human} \quad .$$

As admittance is the inverse of impedance: $H_{human} = \frac{1}{Z_{human}} = \frac{x}{F} = \frac{1}{m_{human}s^2 + b_{human}s + k_{human}}$

As the identification has been performed for each of the degrees of freedom separately:

$$M_{human} = \begin{bmatrix} m_{human,x} & m_{human,y} & m_{human,\theta} \end{bmatrix}$$
$$B_{human} = \begin{bmatrix} b_{human,x} & b_{human,y} & b_{human,\theta} \end{bmatrix}$$
$$K_{human} = \begin{bmatrix} k_{human,x} & k_{human,y} & k_{human,\theta} \end{bmatrix}$$

Whereas the degrees of freedom (x, y and Θ) have been defined according to Fig. B1:

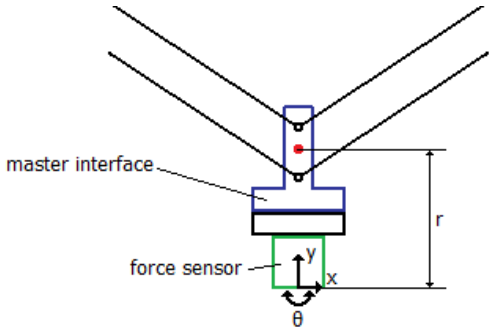


Figure B1: The three degrees of freedom of the identification procedure

Method

The identification was performed using continuous random force inputs and system identification in the frequency domain (van der Helm et al., 2002). The apparatus used was a linear hydraulic actuator. The identification procedure has been repeated three times for each of the different degrees of freedom.

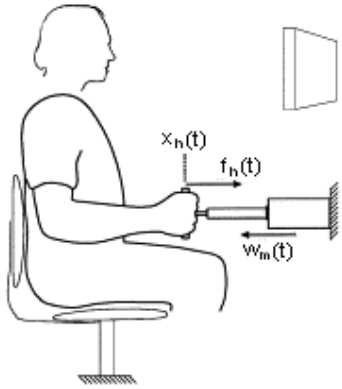


Figure B2: Experimental Setup, adapted from van der Helm, 2002

Fig. B2 shows the experimental setup: the subject is seated in front of the hydraulic manipulator, with a 90 degree angle between upper and lower arm. At the end of the piston a spanner is attached. The subject interacts with the manipulator via the spanner, gripping the spanner as shown in Fig. B3. A force perturbation, $w_m(t)$, is displayed to the subject. The resultant hand force, $f_h(t)$, applied to the manipulator is recorded by a force transducer mounted directly behind the handle, as well as the handle position, $x_h(t)$.

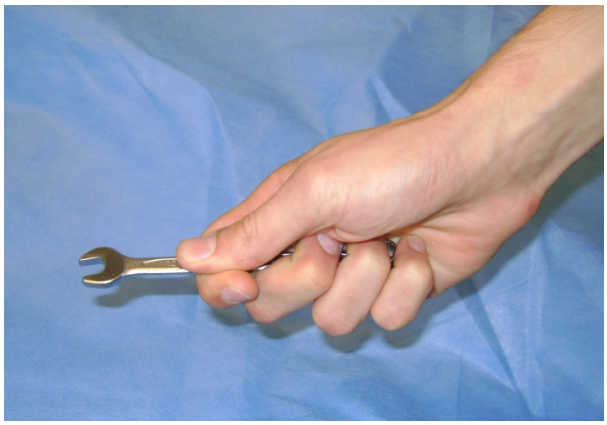


Figure B3: The grip with which subjects are asked to hold the spanner

Fig. B4 shows the human operator closed-loop block scheme of posture control. The transfer function of the human operator H_{human} can be obtained using the spectral densities:

$$H_{human} = \frac{S_{wx}}{S_{wF}} = \frac{conj(W) \cdot X}{conj(W) \cdot F}$$

Here, F, X and W are the Forward Fourier Transforms (FFT) of respectively the contact force, the position and the force perturbation.

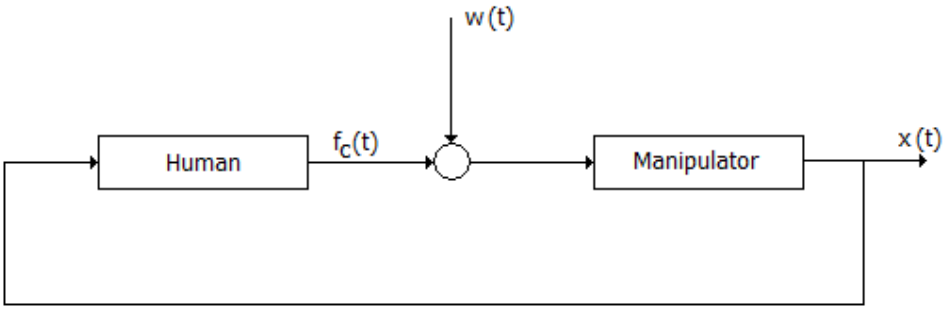


Figure B4: Closed-loop block scheme of posture control, adapted from van der Helm, 2002

The force perturbation (w) consists out of band-limited white noise, with a cut-off frequency of approximately 11 Hz (see Fig. B5). The force perturbation is sampled at 2.5 kHz for 30 seconds, resulting in a signal of 75000 samples. To avoid transient effects, the first three and last seconds are removed from all obtained signals.

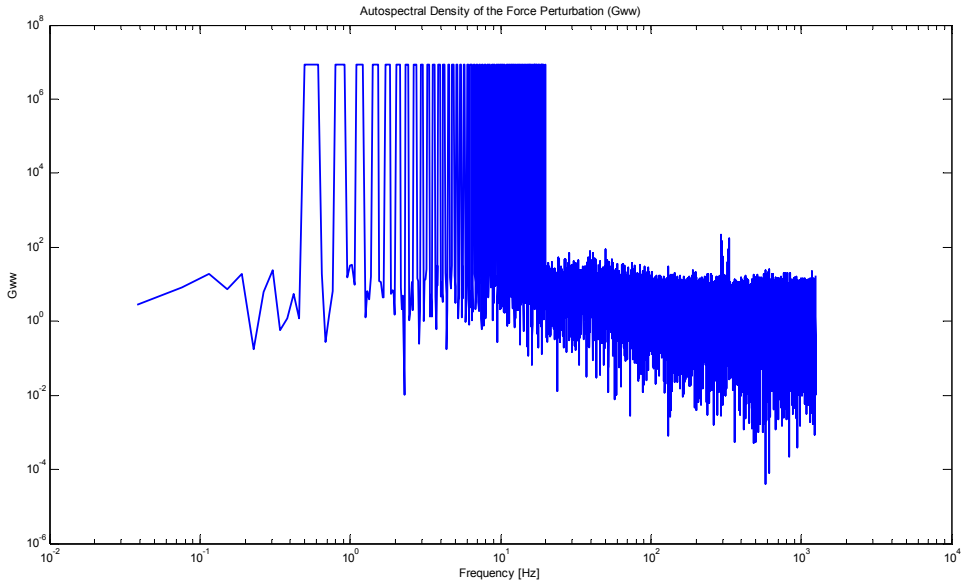


Figure B5: The force perturbation G_{ww} [N^2/Hz]

In order to analyze the complete spectrum of impedances of the human's arm when performing a bolt-and-spanner tasks, subjects are asked to complete either a position task ('minimize displacements') or a relax task ('do not react on the perturbations'). No force tasks were executed; the power spectrum of the force perturbation is not suited to perform force tasks. With the perturbations containing power up to approx. 11 Hz, the human operator has to control a significant high-frequent part. However, the human operator cannot conscious control perturbations with these frequencies, as its manipulation capabilities are limited to approx 6-9 Hz. Hence, the operator will act stiff to minimize the motion, practically performing a position task, and therefore he will not fully benefit from his reflexes, making a force task redundant.

The human operator's stiffness for force tasks is based on previous research. Assuming a zero reflexive gain for relax tasks, the human operator is able to decrease its stiffness for force tasks) by tuning its reflexive feedback gains with the same magnitude, but opposite to the stiffness increase for position tasks (Abbink & Mugge 2009, Abbink 2007): $k_{force} \approx k_{relax} - (k_{position} - k_{relax})$. The

mass and damping of the force task are equal to the mass and damping of the relax task (it is assumed the operator's grip is equal). Thus:

$k_{force} \approx k_{relax} - (k_{position} - k_{relax})$
$b_{force} \approx b_{relax}$
$m_{force} \approx m_{relax}$

Notice that if it would be required to measure the parameters for force tasks, new disturbance forces need to be generated. This disturbance forces should have a cut-off frequency in the human's control domain, as defined in reduced power identification methods (e.g. Mugge et al, 2007).

The analysis is performed for two subjects, performing two to four repetitions:

Degree of Freedom	***Task Instruction***	***Amount of repetitions Subject 1***	***Amount of repetitions Subject 2***
X	position task	4	4
	relax task	4	4
y	position task	4	2
	relax task	4	2
Θ	position task	4	2
	relax task	3	3

Parameter Estimation

The parameters of human operator model have been estimated using a least-square error optimization procedure. A least square error criterion function is used:

$$J(f) = \sum_{f} e(f)^2$$

Whereas, the error function is defined as:

$$e(f) = \sqrt{1/f} \cdot \gamma(f) \cdot \ln \frac{|H_{\text{est}}|}{|H_{\text{adm}}|}$$

Here, H_{est} is the spectral estimator and H_{human} is the transfer function for the admittance as defined. The weights in the error function, $\gamma(f)$ and $\sqrt{1/f}$, respectively put more emphasis on the frequencies with a high coherence and compensates for the lack of data in the low frequencies.

To quantify the goodness of the model fits, the Variance Accounted For (VAF) is calculated.

$$VAF = 1 - \frac{\sum_{i=1}^{n} |x_h(t_i) - \hat{x}_h(t_i)|^2}{\sum_{i=1}^{n} |x_h(t_i)|^2},$$

where $\hat{x}_h(t_i)$ is the forward simulated handle position, i indexes the time vector.

Results

During the identification process a problem occurred. As Fig. B6 shows, the magnitude plot does not end with a -2 slope, while the phase does not go to -180, as a pure mass would do. Instead, the magnitude goes to a zero slope and the phase goes up again (to -90 for damping, to 0 for stiffness). This suggest the presence of viscous damping and/or stiffness in the system; the presence of contact dynamics. Probably, this behavior is generated because of the adhesive tape that was used to connect the spanner to the piston. Hence for the remainder of the identification procedure, the measured data will only be fit up to 5 Hz; the domain where the contact dynamics play a minor role.

Fig. B7 shows a parameter fit of the human operator. Although the fit seems to be acceptable, the Variance Accounted For is low. For this typical example, the VAF is 0.45. This phenomenon can be explained by the fact that the VAF is largely determined by the low frequency content of the signal. As Fig. B8 shows, the the fit is suboptimal for lower frequencies, while it is acceptable for high frequencies.

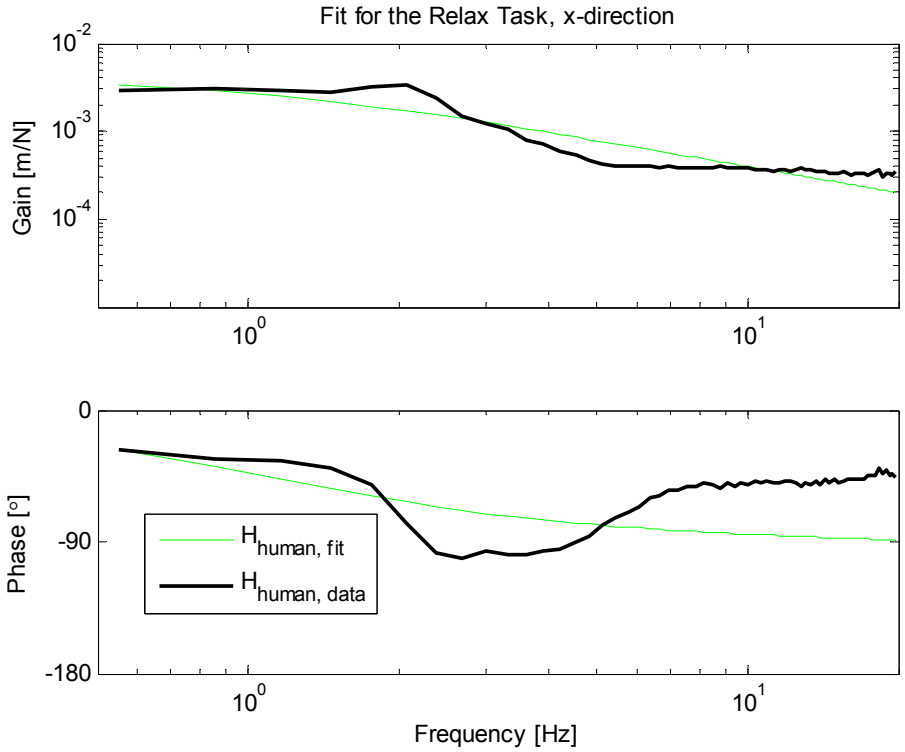


Figure B6: Bode plot of the measured data and a parametric fit; the measured data cannot be fit with a 2nd order model

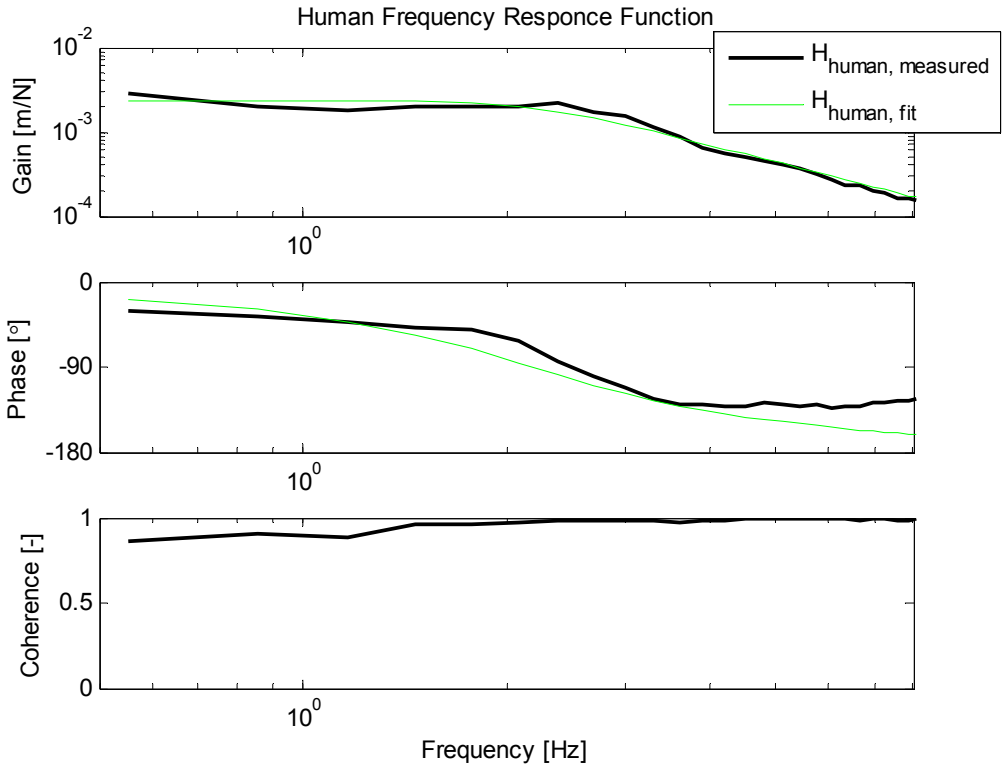


Figure B7: Frequency-domain parameter fit of the human operator, showing the measured signal (black) and the fit (green)

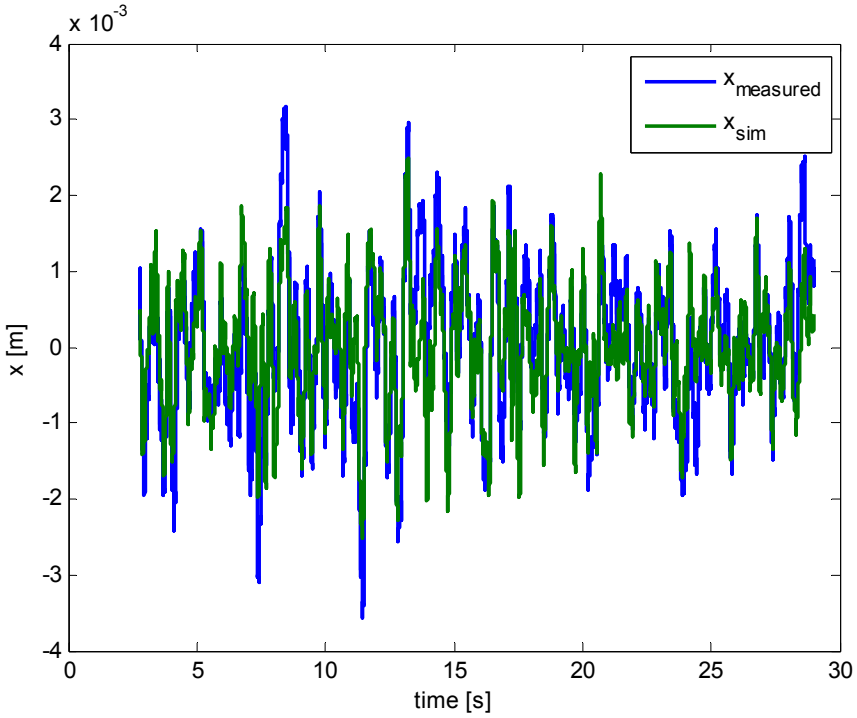


Figure B8: Time-domain parameter fit of the human operator, showing the measured signal (blue) and the fit (green)

Finally, Fig. B9 shows the parameter fit of the manipulator (H_{virt}). Although the manipulator is parameterized as a mass-spring-damper model, the stiffness and the damping are close to zero; the manipulator can be approximated as pure mass, just as expected.

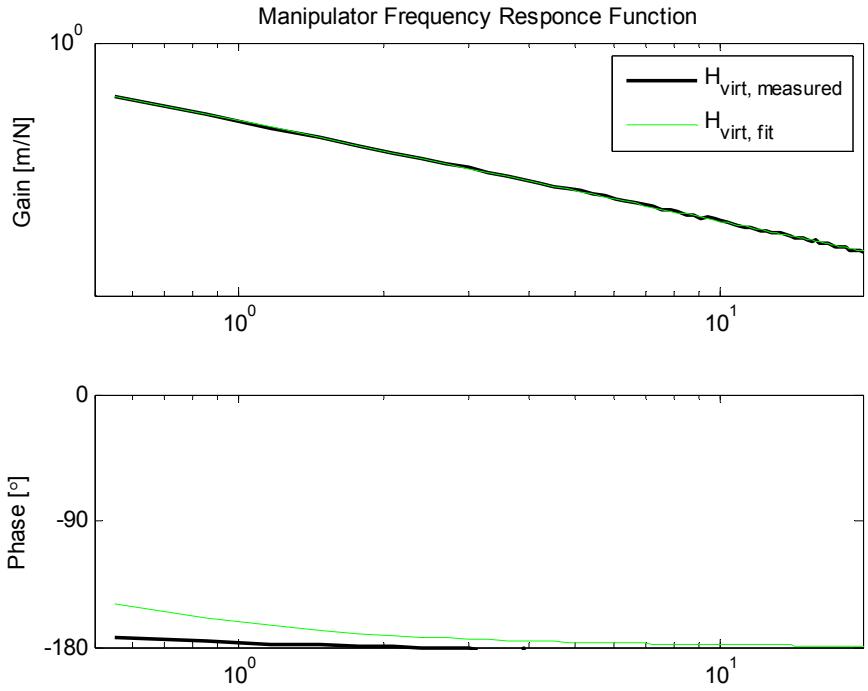


Figure B9: Frequency-domain fit of the manipulator

Tables B1 to B3 respectively show the parameter estimates of the x-direction, the y-direction and the Θ-direction. Notice that the results are subjected to a small intra variance, but a relatively large inter variance.

Table B1: Human operator parameters for the x-direction

	x-direction								
	Position Task			Relax Task			Force task		
	M (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]
Subject 1	2.38 (0.09)	39.0 (4.3)	446 (46)	2.33 (0.16)	29.4 (1.1)	345 (39)	-		
Subject 2	2.61 (0.16)	37.5 (1.2)	630 (73)	2.85 (0.15)	25.7 (0.8)	409 (16)	-		
Average	2.50 (0.16)	38.3 (1.06)	538 (130)	2.59 (0.37)	27.6 (2.6)	377 (45)	2.59	27.6	216

Table B2: Human operator parameters for the y-direction

	y-direction								
	Position Task			Relax Task			Force task		
	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]

Subject 1	0.59 (0.03)	7.5 (0.6)	120 (10)	0.68 (0.08)	5.3 (0.3)	81 (15)	-		
Subject 2	0.39 (0.03)	6.6 (2.2)	160 (21)	0.55 (0.10)	3.6 (2.1)	70 (15)	-		
Average	0.49 (0.1)	7.1 (0.63)	140 (28)	0.62 (0.09)	4.5 (1.2)	76 (7.8)	0.62	4.5	12

Table B3: Human operator parameters for the Θ-direction

	θ-direction								
	Position Task			Relax Task			Force task		
	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]
Subject 1	0.23 (0.07)	2.8 (0.5)	49 (14)	0.09 (0.02)	3.2 (0.8)	22 (7)	-		
Subject 2	0.23 (0.01)	4.1 (1.5)	97 (6)	0.12 (0.09)	1.3 (1.0)	32 (6)	-		
Average	0.23 (0)	3.5 (0.9)	73 (34)	0.11 (0.02)	2.3 (1.3)	27 (7)	0.11	2.3	0

For the Matlab Code see *\Experiments\Identification_Human_Operator\IdentificationAlgorithm.m. The file contains comments on the steps of the algorithm as well as a comments on the data structure and trials.

Appendix C: CTS Simulations

In this Appendix, the efforts to simulate a human operator performing a task in the remote environment will be described. While the objective of the simulations was mainly to gain insight in the system characteristics and problems, hopefully these models can be used to assess the task performance of the telemanipulator in future research.

A telemanipulator consists out of a pair of robotic manipulators and a controller. Movements executed by the *human operator* on the *master* device are translated to the *slave* device, which interacts with the remote *environment*. Master and slave are connected through a *controller*. The connected system of human operator, telemanipulator and environment is referred to as the Connected Telemanipulator System (Christiansson, 2007a). This is illustrated in Fig. C1. This Appendix will present models for each of elements of the Connected Teleoperation System.

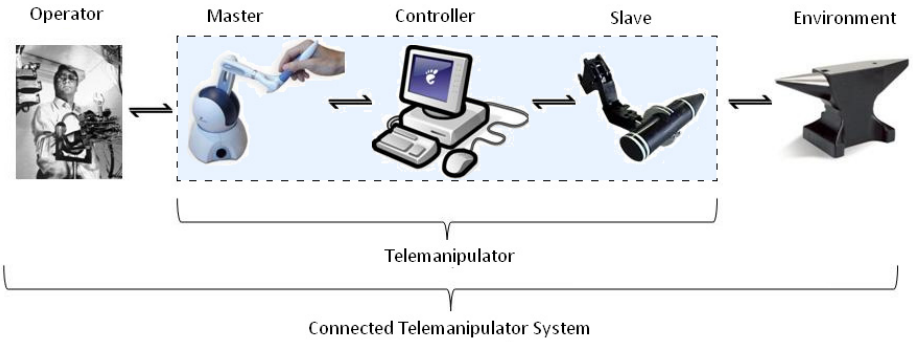


Figure C1: Elements of the Connected Telemanipulator System, adapted from Christiansson 2007a

Fig. C2 shows the top-level of the Connected Telemanipulator Model. The task (colored purple) defines the path the human operator is asked to follow. The human operator has been separated in two: i) a model of the human cortex (colored yellow), where information is received, process and interpreted, and ii) a model of the human's neuromuscular system (colored orange) which translates the outputs from the cortex into physical movements. The contact dynamics (colored green) describe the physical relations ship between the human operator and the telemanipulator. The telemanipulator model (colored blue) calculates the master and slave positions from the force inputs. Finally, the environment (colored grey) describes the physical properties of the environment which the telemanipulator is interacting with.

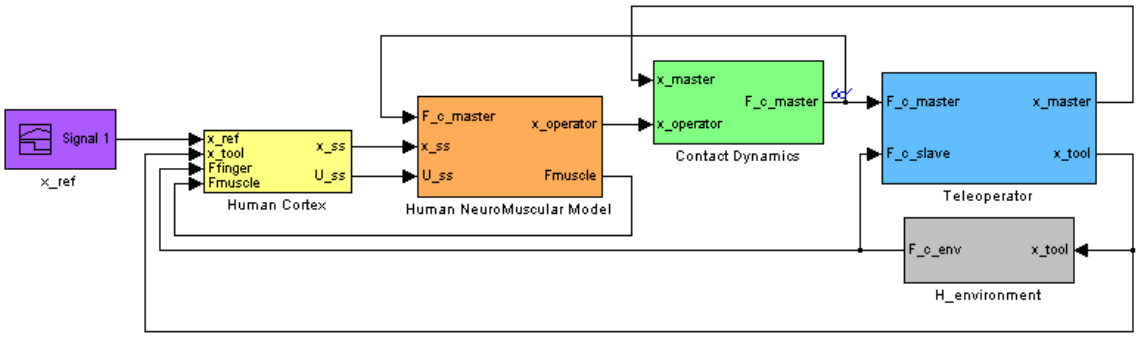


Figure C2: The top-level of the Connected Telemanipulator Model as implemented in Matlab Simulink

C1 Task Model

The task is modeled as an one degree of freedom reference position the human operator is asked to follow. Hence the trajectory inherently contains information about the task that has to be performed. Fig. C3 shows an example of an input. Notice that the human operator is not able to realize this exact trajectory, due to muscle activation dynamics and limb inertia.

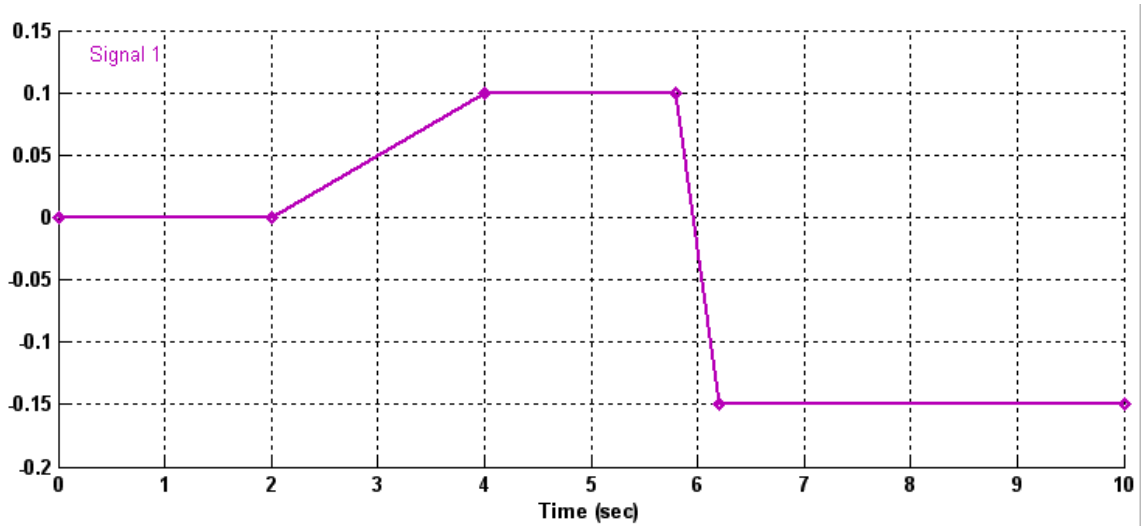


Figure C3: A step input of the CTS model

C2 Human Operator Model

The human operator has been separated in two: i) a model of the human cortex, where information is received, process and interpreted, and ii) a model of the human's neuromuscular system which translates the inputs from the cortex into physical movements.

C2.1 Human Cortex Model

Fig. C4 shows the model of the human cortex. The model contains four inputs. The first input is the reference signal (x_ref); the signal that defines the task. The second input is the position of the tool, in this case the manipulator. The tool position serves as an input for the visual cortex subsystem; the system that visually interprets the tool position and if necessary, adjusts the output. The visual signal is one of three feedback signals from the environment to the human operator. The other two feedback signals are given by the low-frequency feedback force (F_{force}), obtained from the neuromuscular model, and the high-frequency feedback force (F_{finger}), obtained from the environment. Notice that the high-frequency force feedback is created using an accelerometer, which detects contact transitions. Hence, using this high-frequency feedback it is, for example, possible to perceive contact while the limitations of the visual channel (e.g. latency, resolution) would not allow to perceive this contact.

The visual cortex is modeled as a 2nd order Butherworth filter with a cut-off frequency of 0.5 Hz (see Fig. C5) and a time delay of 100 ms(McRuer, 1967). Fully functioning models for the low- and high-feedback have not been implement.

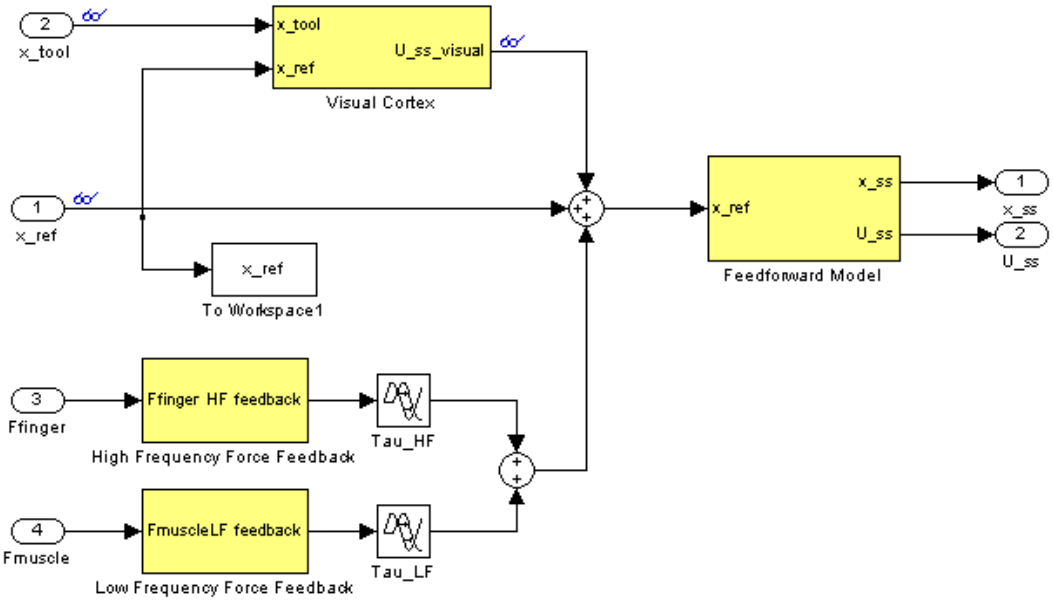


Figure C4: The model of the human cortex as implemented in Matlab Simulink

Among many, Wolpert at al. state that "…the central nervous system internally simulates the dynamic behavior of the motor system in planning, control, and learning…" (Wolpert et al., 1995). In order to perform goal-directed movements, the human operator incorporates an internal model in his control loop to estimate his control inputs. This internal modeled is modeled in the block labeled 'Feedforward Model'.

The Feedforward Model, on one side, calculates the trajectory that can physically be realized by the human operator (x_ss, see Fig. C5). The intermediate factor are the muscle activation dynamics, modeled as a second order Butterworth-filter with a cut-off frequency (activation constant) of 2 Hz. Thus $x_{ss} = (1/H_{act})x_{ref}$. This 'internal' trajectory is used to separate the conscious movements from unexpected perturbations; the human operator's damping and stiffness should only works against unexpected perturbations.

On the other side, the Feedforward Model estimates the supraspinal input of the muscles (U_ss, see Fig. C5); the input from the nervous system that is required to attain the 'internal' trajectory. U_ss is calculated from the inverse of the system. However, the telemanipulator is a 4th order system, while humans are not able to control 3rd order or higher systems (McRuer, 1967). It is assumed that the internal model of the telemanipulator is a 2nd order approximation of the actually telemanipulator, parameterized by a mass, a damping and a stiffness. An estimate of the internal model can be created by time-domain fitting a 2nd order model on the 4th order telemanipulation model.

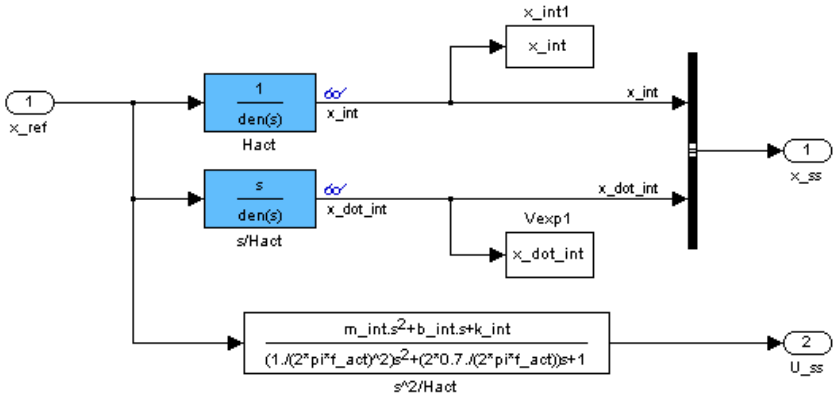


Figure C5: The Feedforward Model as implemented in Matlab Simulink

Hence the internal model is given by:

$$Z_{\text{int}} = m_{\text{int}}s^2 + b_{\text{int}}s + k_{\text{int}} \tag{1}$$

If the human operator is moving his own hand:

$$\begin{aligned} m_{\text{int}} &= m_{human} \\ b_{\text{int}} &= b_{human} \\ k_{\text{int}} &= k_{human} \end{aligned} \tag{2}$$

If the human operator is manipulating a tool:

$$\begin{aligned} m_{\text{int}} &= m_{human} + m_{tool} \\ b_{\text{int}} &= b_{human}\left(+ b_{tool}\right) \\ k_{\text{int}} &= k_{human}\left(+ k_{tool}\right) \end{aligned} \tag{3}$$

Now if the human operator is moving a telemanipulator, the reflected telemanipulator properties are added:

$$\begin{aligned} m_{\text{int}} &= m_{human} + m_{refl} \\ b_{\text{int}} &= b_{human}\left(+ b_{refl}\right) \\ k_{\text{int}} &= k_{human}\left(+ k_{refl}\right) \end{aligned} \tag{4}$$

There are two ways to determine the reflected parameter properties: I) a m-b-k estimation of the analytical transfer function and II) a m-b-k fit on the model-output.

Method I)

The reflected telemanipulator properties (m_{refl}, b_{refl} and k_{reflf}) can be deducted from the telemanipulator's transmitted impedance (Christiansson, 2007a):

$$Z_{\text{trans}} = \left.\frac{F_h}{X_h}\right|_{F_{e^*}=0} = \frac{h_{11} + (h_{11}h_{22} - h_{12}h_{21})Z_e}{1 + h_{22}Z_e} \tag{5}$$

For free air movement, the environmental impedance (Z_e) is zero, which yields:

$$Z_{\text{trans, free air}} = h_{11} = \frac{(Z_m - K_{12})Z_s - K_{24}Z_m + K_{12}K_{24} - K_{14}K_{22}}{(K_{11}+1)Z_s + (-K_{11}-1)K_{24} + K_{14}K_{21}} \tag{6}$$

Whereas the control gains are defined by:

$$K = \begin{bmatrix} 0 & -K/s & 0 & K/s \\ 0 & K/s & 0 & -K/s \end{bmatrix} \tag{7}$$

And (also see Appendix A4):

$$\begin{aligned} Z_{master} &= m_{master}s^2 + b_{master}s \\ Z_{slave} &= m_{slave}s^2 + b_{slave}s \end{aligned} \tag{8}$$

Substituting the values for K, Z_{master} and Z_{slave} as derived in Appendix A4 in equations (7) and (8) yields:

$$h_{11} = \frac{(Z_m - K_{12})Z_s - K_{24}Z_m + K_{12}K_{24} - K_{14}K_{22}}{(K_{11}+1)Z_s + (-K_{11}-1)K_{24} + K_{14}K_{21}} = \frac{0.12s^4 + 5.5s^3 + 190s^2 + 3000s}{0.3s^2 + 10s + 200} \tag{9}$$

Finally, the analyitical transfer function as described by (9) can be approximated by a 2nd order m-b-k-model as described by (1):

$$h_{11} = \frac{0.12s^4 + 5.5s^3 + 190s^2 + 3000s}{0.3s^2 + 10s + 200} \approx m_{refl}s^2 + b_{refl}s + k_{refl} \tag{10}$$

Method II)

A disturbance signal is presented to the telemanipulator model of Fig. C2, whereas a m-b-k fit is made on the output of the model (x_model). The simulated disturbance (Fdist) is a white Gausian noise signal of 10 seconds sampled at 1 kHz.

Results

The results are shown in Fig. C6. The blue graph represents the model output, while the green and red graph respectively represent the transfer function approximation (Method I)) and the model-output fit (Method II)). The fit of the later method seems more accurately. This is expressed by the Variance Accounted For (VAF) which yields 0.9937 for the model output fit.

A least-square error parameter fit on the model output returns:

$$m_{refl} = 0.3257 \text{ kg}$$
$$b_{refl} = 13.1863 \text{ Ns/m}$$
$$k_{refl} = 0.0154 \text{ N/m}$$

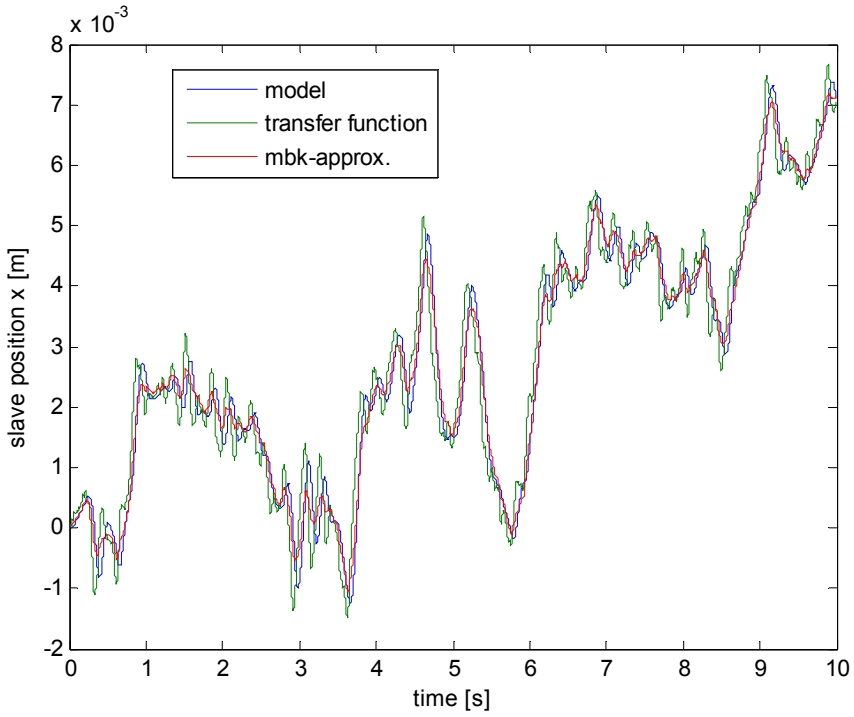


Figure C6: Approximation of the telemanipulator using a 2nd order model

The Matlab Code for the Visual Cortex Model (see *\Simulations\Models\B_visual\B_visual.m)

```
% This Butterworth filter is based on McRuers crossover model:
% D. T. McRuer and H. R. Jex, A Review of Quasi-Linear
% Pilot Models, IEEE Trans. Human Factors in Eletronics,
% HFE-8, 3, pp. 231-208, 1967.

f_c = 0.5; % Hz
WMIN = 1e-2;
WMAX = 1e2;
```

```
B_visual = tf(1,[1./(2*pi*w_c)^2 1.4142/(2*pi*w_c) 1]);

figure(1)
bode(B_visual,'r', B_visual2,'b',{WMIN,WMAX})
```

The Matlab Code for the reflected impedance approximation and fit (see *\Simulations\Models\Reflected Impedance\reflected_impedance.m)

```
clear all
close all
clc

%% Definitions & Constants

s = tf('s');

Kp = 200; % proportional gain
m_master = 0.4; % kg*m^2
b_master = 5; % Ns/m
m_slave = 0.3; % kg*m^2
b_slave = 10; % Ns/m

% uitgaande van een velocity signaal. een positite signaal geeft Mn*s^2 +
bm*s
Zm = m_master*s + b_master;
Zs = m_slave*s + b_slave;


%% K-matrix
% uitgaande van een velocity signaal. een positie signaal geeft Kp ipv Kp/s
K = [0 -Kp/s 0 Kp/s
     0 Kp/s 0 -Kp/s];


%% H-matrix

% h11 = ((Zm-K(1,2))*Zs - K(2,4)*Zm + K(1,2)*K(2,4) - K(1,4)*K(2,2)) /
((K(1,1)+1)*Zs + (-K(1,1)-1)*K(2,4) + K(1,4)*K(2,1));
% h12 = - (K(1,3)*Zs - K(1,3)*K(2,4) + K(1,4)*K(2,3) - K(1,4)) /
((K(1,1)+1)*Zs + (-K(1,1)-1)*K(2,4) + K(1,4)*K(2,1));
% h21 = - (K(2,1)*Zm + (K(1,1)+1)*K(2,2) - K(1,2)*K(2,1)) / ((K(1,1)+1)*Zs
+ (-K(1,1)-1)*K(2,4) + K(1,4)*K(2,1));
% h22 = - ((K(1,1)+1)*K(2,3) - K(1,3)*K(2,1) - K(1,1) - 1) / ((K(1,1)+1)*Zs
+ (-K(1,1)-1)*K(2,4) + K(1,4)*K(2,1));
%
% H = [h11 h12
%      h21 h22];
%
% figure(1)
% bode(h11)
% [PCONST, SCONST] = Haptic_Constants;
%
% [m,b,k] = Haptic_impedance_fit(h11, PCONST)
%

%% H-matrix II
```

```
h11_alt = ((s*Zm + Kp)*Zs + Kp*Zm) / (s*Zs + Kp); % p34 proefschrift Goran
%
% figure(2)
% bode(h11_alt)
%
% [PCONST, SCONST] = Haptic_Constants;
% [m_alt,b_alt,k_alt] = Haptic_impedance_fit(h11_alt, PCONST)

f_act = 2;
Hact = tf(1,[(1./(2*pi*f_act)^2) (2*0.7./(2*pi*f_act)) 1]);

H_int = Hact*h11_alt;

%% Teleoperator Properties

m_master = 0.4; % kg*m^2
b_master = 5; % Ns/m

m_slave = 0.3; % kg*m^2
b_slave = 10; % Ns/m

P_gain_master = 200;
P_gain_slave = 200;
D_gain_master = 0;
D_gain_slave = 0;


%% Simulations Parameters

Fdist = wgn(10000,1,2);
time = 0:.001:9.999;

m_2O_fit = 1;
b_2O_fit = 1;
k_2O_fit = 1;

a0 = 1;
a1 = 1;
a2 = 1;
a3 = 1;

[dummy, dummy, x_slave] = sim('teleoperator',time);
x_slave_model = x_slave(:,1);

%% mbk-parameter estimation

%       M   B   K
lb = [.01   0   0]; % lower bound parameter vector
ub = [4.0 100   1]; % upper bound parameter vector
x0 = [1.0  50  .5]; % initial guess parameter vector

% normaliseren van de parameters
norm_vector = ub;
lb_norm = lb./norm_vector;
ub_norm = ub./norm_vector;
x0_norm = x0./norm_vector;

% set options
```

```
options = optimset('lsqnonlin');
options =
optimset(options,'display','final','diffminchange',0.001,'MaxIter',10000,'M
axFunEvals',10000);
lsqn_out = lsqnonlin(@error_fun_tele, x0_norm, lb_norm, ub_norm, options,
time, Fdist, x_slave_model, norm_vector);

% m_2O_fit = .23
% b_2O_fit = 15
% k_2O_fit = 0

m_2O_fit = lsqn_out(1)*norm_vector(1)
b_2O_fit = lsqn_out(2)*norm_vector(2)
k_2O_fit = lsqn_out(3)*norm_vector(3)

[dummy, dummy, x_slave] = sim('teleoperator',time);

x_slave_model = x_slave(:,1);
x_slave_tf = x_slave(:,2);
x_slave_mbk = x_slave(:,3);

VAF = 1-sum(abs(x_slave_model-x_slave_mbk).^2)/sum(abs(x_slave_model).^2)

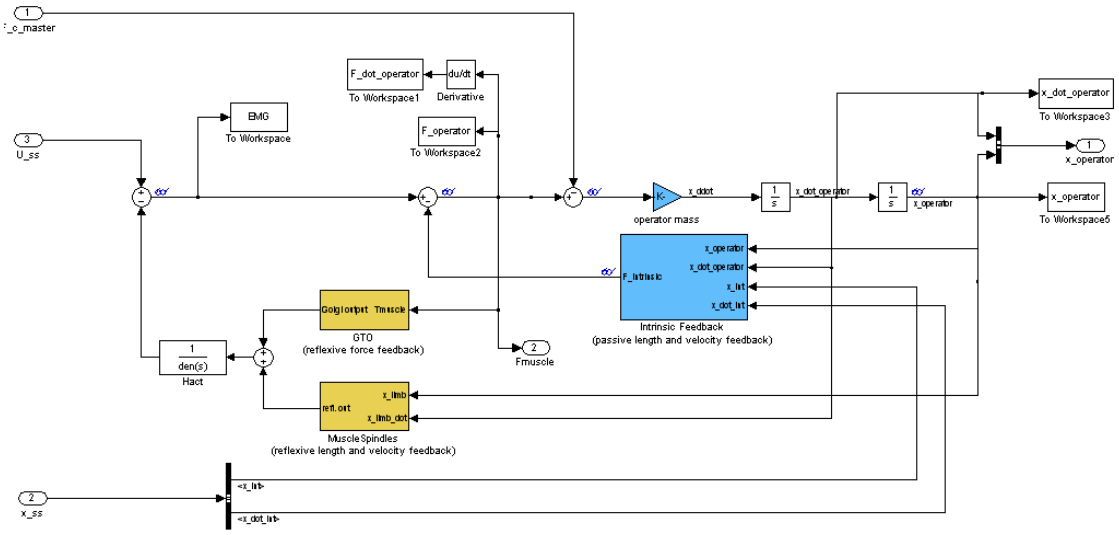

%% Plot

figure(3)
plot(time, [x_slave_model x_slave_tf x_slave_mbk])
ylabel('slave position x [m]')
xlabel('time [s]')
legend('model','transfer function','mbk-approx.')
```

C2.2 Human Neuromuscular Model

The human neuromuscular model has been based on the model as described in de Vlugt et al., 2006. The model contains the human intrinsic properties (mass, stiffness and damping) as well as the reflexive properties (muscle spindle and GTO feetback). This is shown in Fig. C7. Initially, the reflexive gains have been set to zero.

However, this model is only applicable as a static model, as it does not allow for movements (e.g. the stiffness is defined around one reference point). For movements however, the reference position and velocity have to be made dynamically; the position and velocity should be updated in time. To solve this issue, the current operator position (x_operator) is compared to the expected operator position as determined by the internal model (x_int) . The human operator's damping and stiffness will therefore apply on the error between the expected and actual position and velocity. This separates conscious movements from unexpected perturbations, such that the stiffness and damping only works against unexpected perturbations



Figuur C7: The human neuromuscular model implemented in Matlab Simulink, adapted from de Vlugt et al., 2007

C3 Contact Dynamics Model

While the contact dynamics do have a real physical meaning, they describe the dynamics (e.g. the skin) between the human operator and the telemanipulator, the most important reason to implement the contact dynamics block is due to calculation technical reasons; it is impossible to calculate the telemanipulator positions from the output of the human operator model, the human operator position (differentiating is impossible). Therefore, contact dynamics are introduced. Fig. C8 illustrates the connection between human operator, contact dynamics and master device. It can be seen that the difference between the operator and master position and velocity times respectively a stiffness or a damping yield as contact force.

Thus:

$$F_{contact} = \left(x_{operator} - x_{master}\right)k_{contact} + \left(\dot{x}_{operator} - \dot{x}_{master}\right)b_{contact} \tag{11}$$

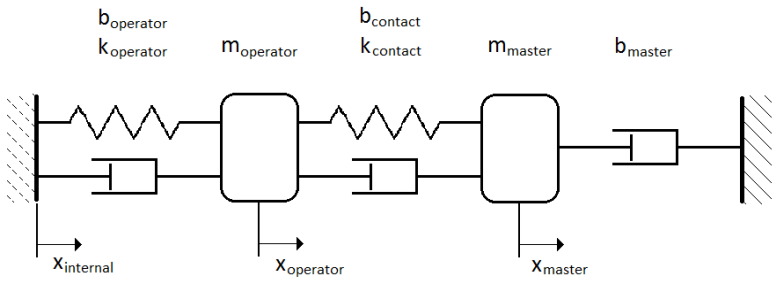


Figure C8: Connection of human operator, contact dynamics and master device

Notice that if the contact would be rigid, the influence of the contact dynamics on the simulation is neglectable. However, infinite stiffness and damping cannot be modeled; they are limited by simulation time and stability. Instead, the contact stiffness and damping are chosen several orders higher than the human operator properties. Fig. C9 to C11 show that increasing the contact stiffness by one order, decreases the error caused by the infinite stiffness assumption by one order. For $k_{contact}$ = 100.000 N/m and $b_{contact}$ = 1000 Ns/m, the simulation time is acceptable.

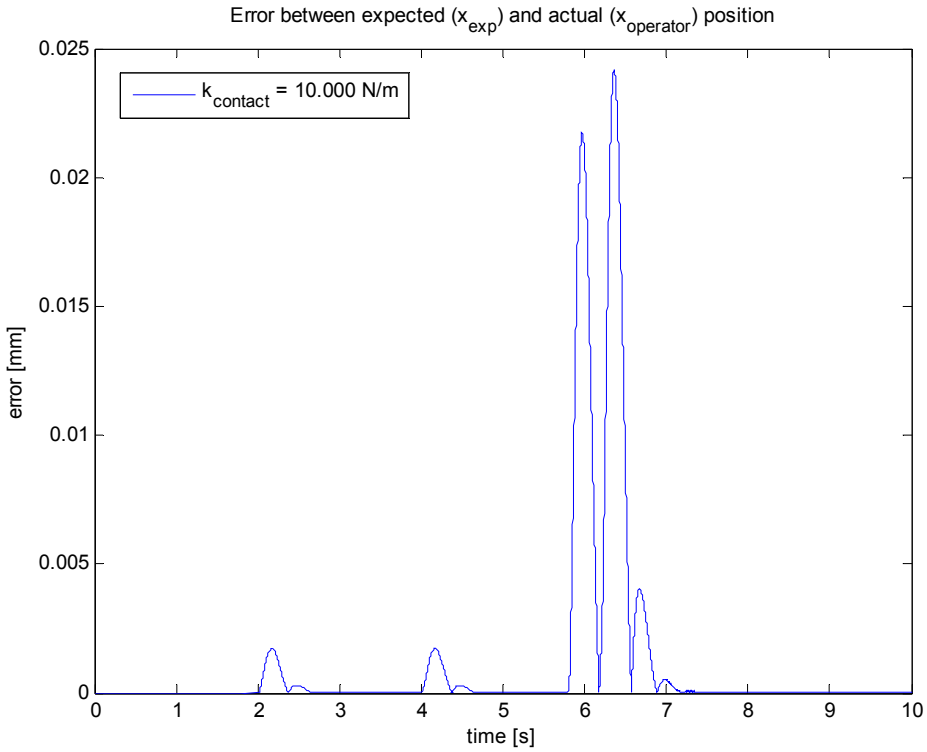


Figure C9: The error between the realized and expected positions for k = 10.000 N/m.

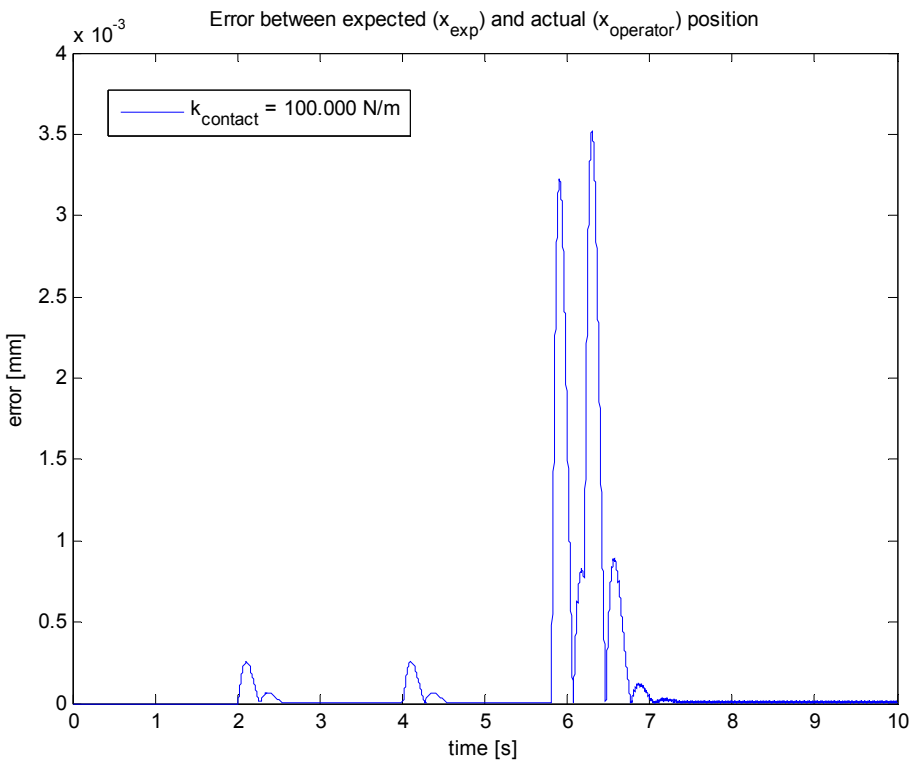


Figure C10: The error between the realized and expected positions for k = 100.000 N/m.

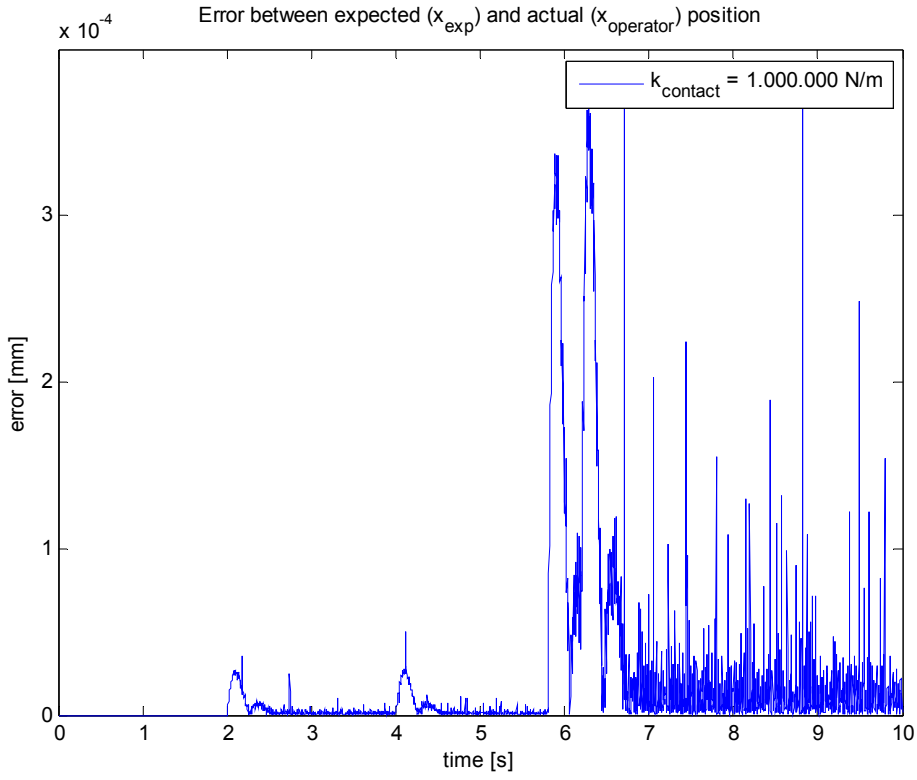


Figure C11: The error between the realized and expected positions for k = 1.000.000 N/m.

C4 Telemanipulator Model

The telemanipulator has been according to the two-port network analogy as described in Appendix A4. The implementation of this model in Matlab Simulink is shown in Fig. C9. The parameters of the telemanipulator model (K_P, K_D, m_{master}, b_{slave}, m_{slave}, b_{master}) can be obtained from Appendix A4.

Notice that the telemanipulator model also allows to simulate the HandsOn condition of the experiment; the slave is decoupled from the simulation, and the master position is seen as the system's output, instead of the slave position.

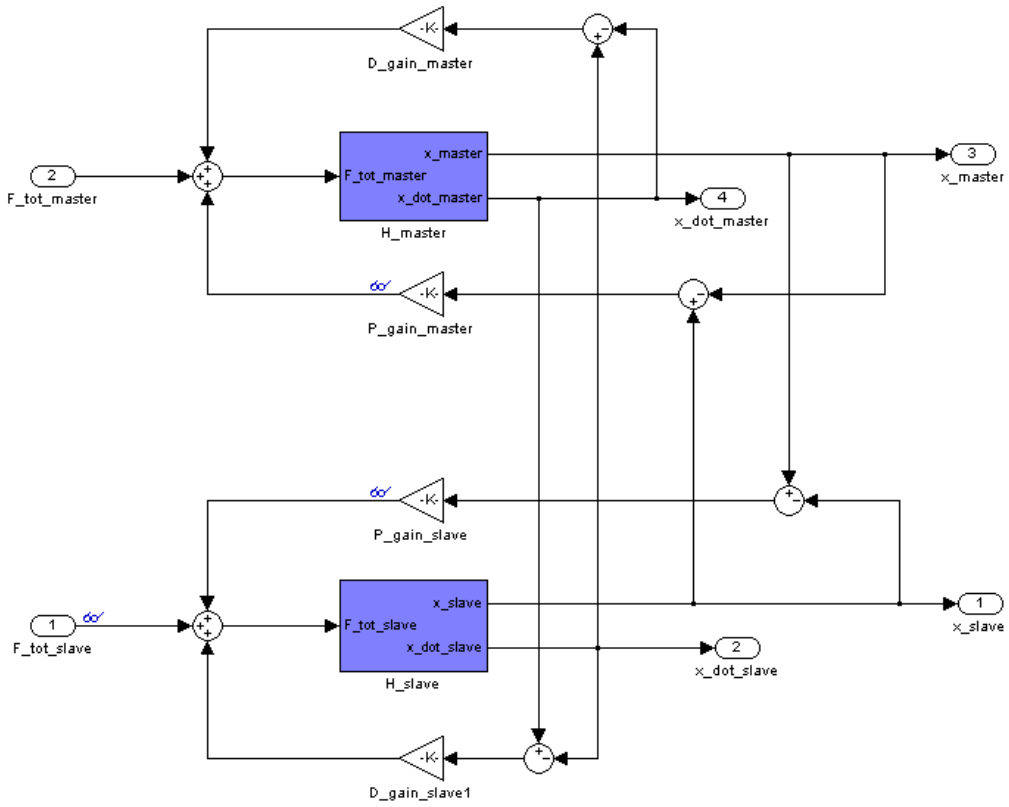


Fig. C9: Implementation of the telemanipulator model in Matlab Simulink

The transfer function of master is shown in Fig. C10. The same model is used for the slave, but now with different values for the inertia and damping parameters.

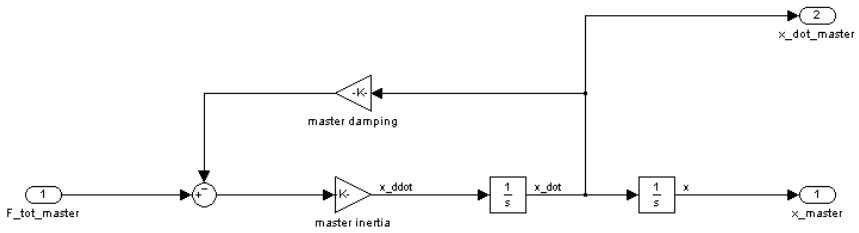


Figure C10: Implementation of the master transfer function in Matlab Simulink

C5 Environmental Model

The environment is modeled an impedance having stiffness and damping properties:

$$Z_{env} = b_{env}s + k_{env} \tag{11}$$

The environment is included in the simulation using an IF-condition. If the manipulator endpoint positions crosses a certain threshold, the environment as a subsystem is enabled and the contact forces as a result of the contact between manipulator and environment are calculated. This is illustrated in Fig. C11

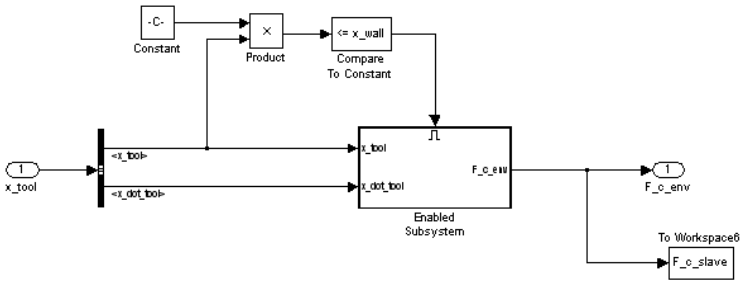


Figure C11: Implementation of the environment in Matlab Simulink

C6 Preliminary Results & Future Work

Fig. C12 shows the results of a 10 second simulation of the trajectory as described in Appendix C1. The reference position is described by the solid black line. The humans internal model, the cyan colored solid line, shows the translation of the reference trajectory as calculated by the internal model. It can be seen that the internal model accounts for the fact that transitions cannot occur instantaneously. Furthermore, the dashed blue line and the dashed red line respectively present the positions of the operator and the slave.

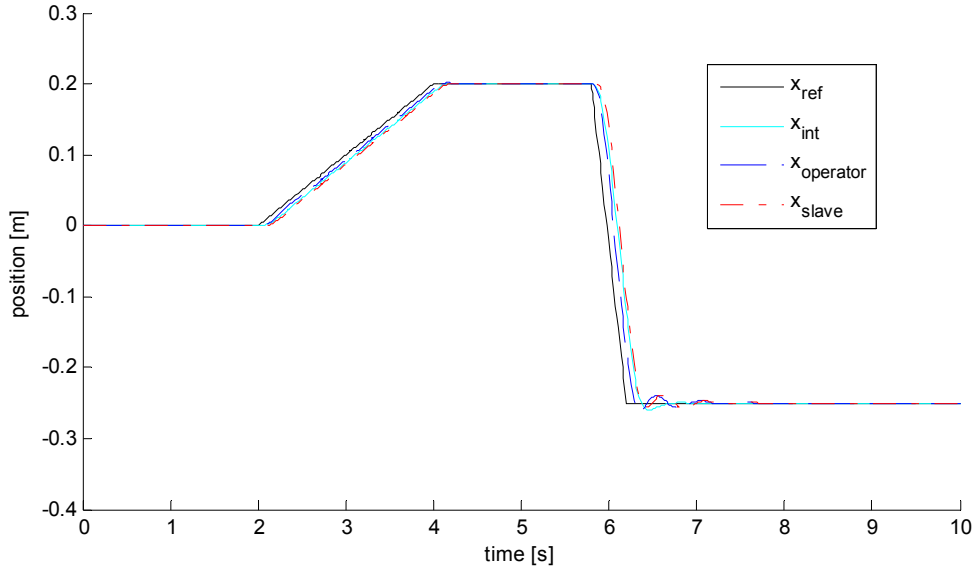


Figure C12: CTS Simulation results

The errors between expected position and realized position is introduced because:

1) The internal model is only an approximation of the actual telemanipulator model. It approximates the telemanipulator as a 2nd order model.
2) The internal model does not account for the contact dynamics; it assumes the contact between human operator and master is rigid and thus the stiffness and damping are infinite.

However, this error is acceptable. From Fig. C10 is obtained that the error for a movement of 0.2 m is in the order of 1e-5 m. Taking in account that the human's position sensing accuracy is approximately 1e-3 m (Tan et al., 1994), than it is acceptable that the error is two orders of magnitude smaller. Moreover, as the error is reduced by one order if the contact stiffness is increased by one order, the contact dynamics dominate the magnitude of the error, and not the 2nd order telemanipulator approximation.

To summarize, it has been shown that the different subsystems as well as the complete model function. The model in its current state can be used to study the influence (expressed using positions, forces, etc.) of different properties of the human operator, the environment and the telemanipulator. Task performance metrics have also been implemented. In order to analyze the tasks as described in the paper, the following issues have to be addressed:

- Refine the human cortex model
  - Implement a model for the low frequent force feedback (e.g. a 4-channel channel yields better force feedback than a PERR-controller)
  - Implement a model for the high frequent force feedback
  - Improve the visual feedback model
- Refine the task instruction (e.g. by defining the fundamental tasks as described in the paper)
- Implement a model for task instruction ('perform the task fast' or 'perform the task accurately')
- Expand the model to multiple degrees of freedom

The four fundamental tasks as defined in this research, the Free Air Movement Task, the Contact Transition Task, the Constrained Position Task and the Force Task can be described using the following models for human operator and environment:

Free Air Movement Task
- Operator control model: position control model (all 3 dof)
- Environment model: $H_{env}=0$ for x,y,Θ

Contact transition (1 dof)
- Operator control model: from 3 dof position control to 2 dof position, 1 dof force control
- Environment model: $H_{env}=0$ for x,y,Θ to $H_{x,env}= 1/(bs + k)$

Constrained position task
- Operator control model: 2 dof force control, 1 dof position (the movement direction)
- Environment model: $H_{y,\Theta,env} =1/(bs + k)$ and $H_{x,env}= 0$, $F_{resultant} = N*\mu$

Force movement
- Operator control model: 3 dof force control
- Environment model: $H_{y,x,env} =1/(bs + k)$ and $H_{\Theta,env}= 0$, $M_{resultant} = N*\mu$

It is assumed that in a compliant environment, the human operator will try to manipulate using position control. In stiff environments, the operator will apply force control. The properties of the human operator for the position task and force task are given in Table C1 (also see Appendix B).

Table C1: Human operator properties for the position task and the force task

	Position Task			Force task		
	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]	m (σ) [kg]	b (σ) [Ns/m]	k (σ) [N/m]
x	2.50	38.3	538	2.59	27.6	216
y	0.49	7.1	140	0.62	4.5	12
θ	0.23	3.5	73	0.11	2.3	0

The simulation of the connected teleoperator system can be run by executing “loadparameters.m” (see *\Simulations\Models\loadparameters.m). This file contains the parameter definitions and initializes and runs the connected teleoperator model. It has been extensively commented to ensure readability.

Appendix D: Human Factors Experiment

In this Appendix several elements of the human factors experiment as described in the paper will be addressed in more detail. Appendix D1 describes how the different subtasks have been separated, while the definition of the performance metrics is described in Appendix D2. In Appendix D3 the structure of the measurement data and the data analysis is explained. Finally, Appendix D4 provides a comprehensive overview of this study's results.

D1 Task Completion Criteria

Subjects are asked to complete a bolt-and-spanner task, which contains the four fundamental tasks as defined in this research. Starting at point A, subjects subsequently move to B and C (see Fig. D1). A, B and C are respectively located at (x y Θ) = (0 0 0), (-0.06m 0.08m 0°) and (0.06m 0.08m -60°). From C, the subjects have the objective to slide the spanner over the bolt, which is located at D = (x,y) = (0m 0.12m). Finally a 80 degree stroke has to be made.

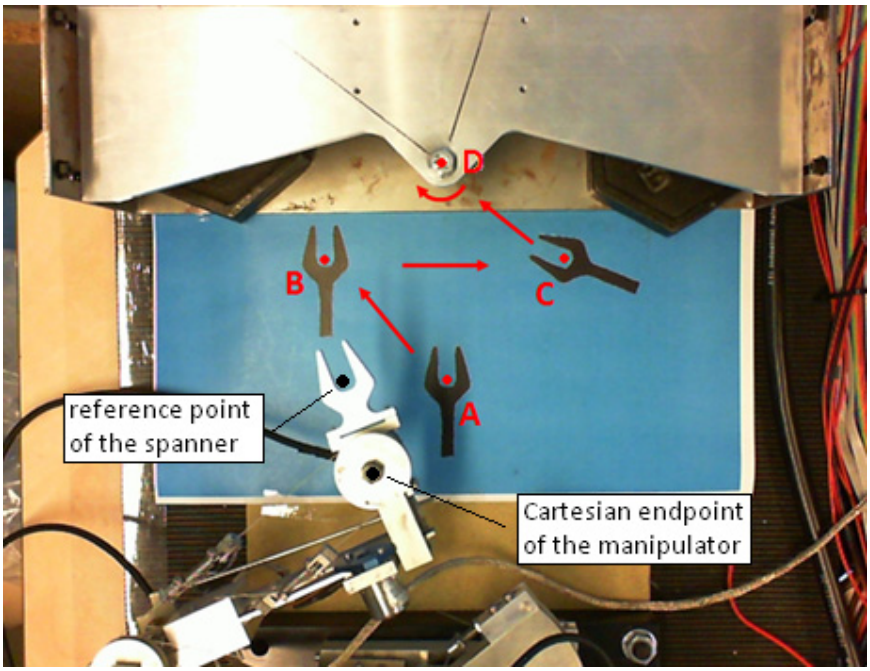


Figure D1: Implementation of the bolt-and-spanner task on the slave side

Fig. D2 shows a typical example of a subject completing the bolt-and-spanner-task. The Free Air Movement Task is represented by the blue colored trajectory, the Contact Transition Task is colored red, while the Constrained Position Task and Force Task are respectively colored cyan and green.

Notice that this trajectory plot, shows the Cartesian endpoint position of the manipulator, rather than the reference point of the spanner. This is illustrated in Fig. C1. It is irrelevant which of the two points is chosen, as the spanner reference point can be easily calculated from the Cartesian endpoint position.

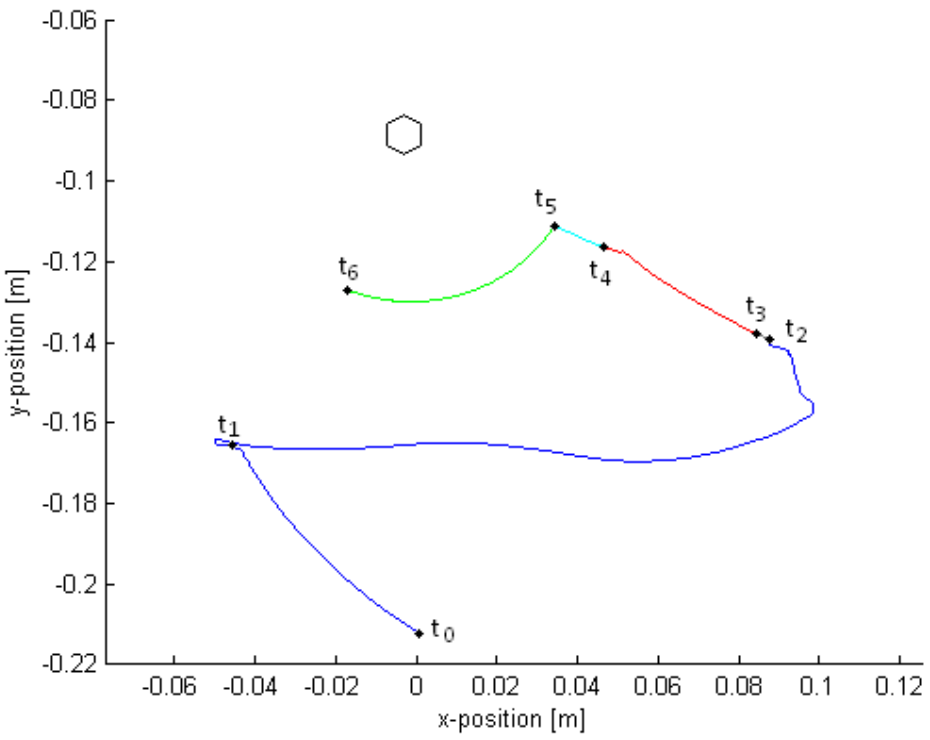


Figure D2: Manipulator endpoint position X, showing the four different subtasks

Let X_s be the Cartesian endpoint position of the slave:,

$$X_s = \begin{bmatrix} x_s \\ y_s \\ \theta_s \end{bmatrix}$$

than the spanner reference point (X_{ref}) is calculated by:

$$X_{ref} = \begin{bmatrix} x_{ref} \\ y_{ref} \\ \theta_{ref} \end{bmatrix} = \begin{bmatrix} x_{ref} + r\cos\theta_s \\ y_{ref} + r\sin\theta_s \\ \theta_{ref} \end{bmatrix}$$

Whereas r is the distance between X_{ref} and X_s (r = 0.04 m). For tasks performed on the master side (HandsOn tasks) a similar equation holds.

The traversed trajectory is cut into six segments, defined by seven time indices, namely t_0 to $t_{6=}$ (see Fig. D2). The first two segments, t_0 (the time index where the subjects starts moving) to t_2, define the Free Air Movement Task. As the Free Air Movement Task is not performed blind, subjects are asked to hold their movement for a fraction of a second after completing this subtask. The index where subjects stop moving is defined as t_2, while at t_3 the subjects start moving again (in practice, points t_3

and t_4 often describe identical positions). Hence t_3 defines the start of the Contact Transition Task. The Contact Transition Task is ended at time index t_4. This is the point where subjects have made contact, have oriented the spanner and commence for a sliding motion. This sliding motion is the Constrained Position Task, starting at t_4 and ending at t_5. For t_5 holds, that the subjects have aware of the fact that they have completed the Constrained Position Task, and hence commence the rotational movement; the Force Task. The force tasks ends at t_6, which is the point where subjects stop their movements.

Let c_1 to c_{21} be positional constraints, than the indices have to suffice the following conditions:

t_0

$$c_1 < x(t_0) < c_2$$
$$c_3 < y(t_0) < c_4$$
$$\sqrt{\dot{x}(t_0)^2 + \dot{y}(t_0)^2} = 0,$$
$$\sqrt{\dot{x}(t_0 : t_0 + 100)^2 + \dot{y}(t_0 : t_0 + 100)^2}$$
$$\dot{x}(t_0 + 1) > 0$$
$$\dot{y}(t_0 + 1) < 0$$

t_1

$$c_5 < x(t_1) < c_6$$
$$c_7 < y(t_1) < c_8$$
$$\sqrt{\dot{x}(t_1)^2 + \dot{y}(t_1)^2} = 0$$

t_2

$$c_9 < x(t_2) < c_{10}$$
$$c_{11} < y(t_2) < c_{12}$$
$$\sqrt{\dot{x}(t_2)^2 + \dot{y}(t_2)^2} = 0$$

t_3

$$c_{13} < x(t_3) < c_{14}$$
$$c_{15} < y(t_3) < c_{16}$$
$$\sqrt{\dot{x}(t_3)^2 + \dot{y}(t_3)^2} = 0$$
$$\sqrt{\dot{x}(t_3 + 1 : t_3 + 100)^2 + \dot{y}(t_3 + 1 : t_3 + 100)^2} \neq 0$$

t_4

$$c_{17} < \sqrt{(x(t_4) - x_{bolt})^2 + (y(t_4) - y_{bolt})^2} < c_{18}$$

t_5

$$c_{19} < \sqrt{(x(t_5) - x_{bolt})^2 + (y(t_5) - y_{bolt})^2} < c_{20}$$
$$\sqrt{\dot{x}(t_5)^2 + \dot{y}(t_5)^2} = 0$$
$$\sqrt{\dot{x}(t_5 + 1 : t_5 + 100)^2 + \dot{y}(t_5 + 1 : t_5 + 100)^2} > 0$$
$$\theta(t_5) > 130^\circ$$

t_6

$$c_{21} < \sqrt{(x(t_6) - x_{bolt})^2 + (y(t_6) - y_{bolt})^2} < c_{22}$$
$$\sqrt{\dot{x}(t_6)^2 + \dot{y}(t_6)^2} = 0$$
$$\sqrt{\dot{x}(t_6 : t_6 + 100)^2 + \dot{y}(t_6 : t_6 + 100)^2} = 0$$
$$\theta < 80^\circ$$

These criteria are rather strict. For example, it often occurs that the point that defines t_1 is being passed while the squared velocity is not zero ($\sqrt{\dot{x}(t_1)^2 + \dot{y}(t_1)^2} \neq 0$). In such cases the index is searched which has the lowest squared velocity, while still sufficing the other criteria. If then still no index can be found, the index is picked manually.

Notice that there are two distinct parameter sets for the task completion criteria: one for telemanipulated tasks (tasks performed on the slave side) and one for manual performed tasks (tasks performed on the master side).

The Matlab code to separate the trajectory in subtasks can be found in *\Master Thesis\ Experiments\PlanarSetup\HumanFactorsExperiment\ProcessData.m, and contains detailed comments on how things have performed. Unfortunately, the code is too long to paste into this appendix (~ 800 lines).

D2 Description Performance Metrics

A broad variety of metrics exists to quantify the performance of the Connected Telemanipulator System. For many teleoperated tasks, task performance metrics like time-to-complete and contact forces are the most relevant metrics. However, as we also want to understand how the human operator deals with different feedback cues, a wide set of task performance metrics is analyzed as shown in Table D2. The '✓' indicates this subtask for which the metric has been analyzed (not every metric is applicable to every subtask). While metrics like time-to-complete and contact forces define the slave side, metrics like input forces and reversal rate illustrate the behavior at the master side.

Table D2: Overview of the analyzed task performance metrics

No.	Task Performance Metric	Free Air Movement Task	Contact Transition Task	Constrained Position Task	Force Task
TP1*	Time-to-Complete [s]	✓	✓	✓	✓
TP2	Avg. Velocity [m/s]	✓	✓	✓	✓
TP3*	Nominal Path Deviation [m^2]	✓	✓		
TP4*	Avg. Contact Force [N]			✓	✓
TP5*	Max. Contact Force [N or m/s^2]		✓	✓	✓
TP6	Max. Contact Impulse [m/s]		✓	✓	✓
TP7*	Avg. Input Force [N]	✓	✓	✓	✓
TP8*	Max. Input Force [N]	✓	✓	✓	✓
TP9*	Physical Workload [Ns]	✓	✓	✓	✓
TP10*	Reversal Rate, x-direction [-]	✓	✓	✓	✓
TP11*	Reversal Rate, y-direction [-]	✓	✓	✓	✓
TP12*	Reversal Rate, Θ-direction [-]	✓	✓	✓	✓
TP13	Angular Offset [deg]				✓
TP14	Avg. F on Contact 100ms [N]		✓		
TP15	Input Force Drop 100ms [N]		✓		
TP16	Avg. F on Contact 300ms [N]		✓		
TP17	Input Force Drop 300ms [N]		✓		

Metrics addressed with a '*' are the metrics which have been presented in the paper. This are also the only metrics on which a statistical analysis has been performed. These metrics are:

t_{ttc} Time-to-complete, the time it requires for the subject to complete the task.

x_{npd} Nominal Path Deviation, the deviation from the nominal path in.

$F_{c,avg}$ Average Contact Forces, the average interaction force of the manipulator with the environment.

$F_{c,max}$ Maximum Contact Force, the maximum interaction force of the manipulator with the environment.

$F_{i,avg}$ Average Input Force, average interaction force between human operator and master.

$F_{i,max}$ Maximum Input Force, maximum interaction force between human operator and master.

W_{phys} Physical Workload, measured as the integral of the absolute input force over time.

n_{rr} Reversal Rate, the amount of steering corrections by the human operator. The reversal rate can be seen as a measure for the mental effort.

t_{ttc} - Time-to-complete (TP1)

The time to complete of the subtasks is simple calculated by subtracting the corresponding time indices. For example, for the Contact Transition the time in seconds is given by $(t_4-t_3)/1000$ (the signal is sampled at 1000 Hz).

x_{npd} - Nominal Path Deviation (TP3)

The nominal path deviation is measured as the surface area (in m^2) that exists due to the differences of the nominal path and the realized path. The nominal path is defined as the shortest route between two points. Hence:

$$x_{npd} = \int (x - x_{nom}) dx$$

F_c - Contact Forces (TP7 & TP8)

For the master-slave configuration, the average contact forces are based on the (dimension less) series-elastic force estimate. The maximum contact forces are based on the maximum output of the accelerometer (m/s^2). For manually performed tasks, both the average and maximum forces are based on the force sensor signal (expressed in N).

F_i - Input Forces (TP4 & TP5)

The human operators average and maximum forces are based on the force sensor signal (expressed in N).

W_{phys} - Physical Workload (TP9)

The physical workload is expressed as the integral as the force sensor signal over time (Ns). Hence $W_{phys} = \int F dt$ The common definition of workload, $W = \int F ds$, is not practical as this definition means that the human operator would not perform any work when it is pushing a rigid wall.

n_{rr} - Reversal Rate (TP10, TP11 & TP12)

The reversal rate of the manipulator is defined as the amount of times the human operator's input force changes direction. The reversal rate is determined separately for x-, y- and Θ-directions, while the signals have first been filtered using a 10 Hz low-pass filter (4^{rd} order Butterworth). Notice that in the paper the reversal rate has been described with only one metric. This metric is the maximum of the significance-values returned by the Anava test, thus describing the degree of freedom in which the differences are the 'least significant':

$$n_{rr} = \max(p_x, p_y, p_\theta)$$

Average velocity (TP2) has shown to be a redundant measure; the distance is approximately equal for each trial and thus, this metrics is in principle not different from time-to-complete (TP1). The maximum contact impulse (TP6) has been defined as a result of the effort to define a generic metric for forces applicable to all conditions (the slave does not contain a force sensor, the master does not contain a accelerometer). The angular offset (TP13) is a measure for the steady state angular deviation from the final angle (20°). However, with so many metrics already being analyzed this metric, which is only applicable for one subtask (the force task), is put aside. Finally, as many subjects indicated they did benefit from the high-frequency feedback, TP14-TP17 have been defined to analyze the forces after contact in more detail. Regretfully, the definition of the task completion criteria does not allow for proper analysis of these metrics; the meaning of TP14-TP17 is unclear as the exact moment where the high-frequency pulse is generated is not very well defined.

The Matlab code to evaluate the task performance metrics for each subtasks can be found in *\Master Thesis\ Experiments\PlanarSetup\HumanFactorsExperiment\AnalyzeData.m, and contains detailed comments on how things are performed. The task performance numbers as defined in Table D2 correspond to the numbers used in the code.

D3 Data Management

34 Output signals are logged for 112 trials per subjects, yielding 2 GB of raw data for 10 subject. Each trial is separated in 4 subtasks, while each subtask is analyzed for 17 performance metrics are analyzed. As a result 2300 lines of code are required to analyze the data. Fig. D3 shows the flowchart of the data analysis algorithms.

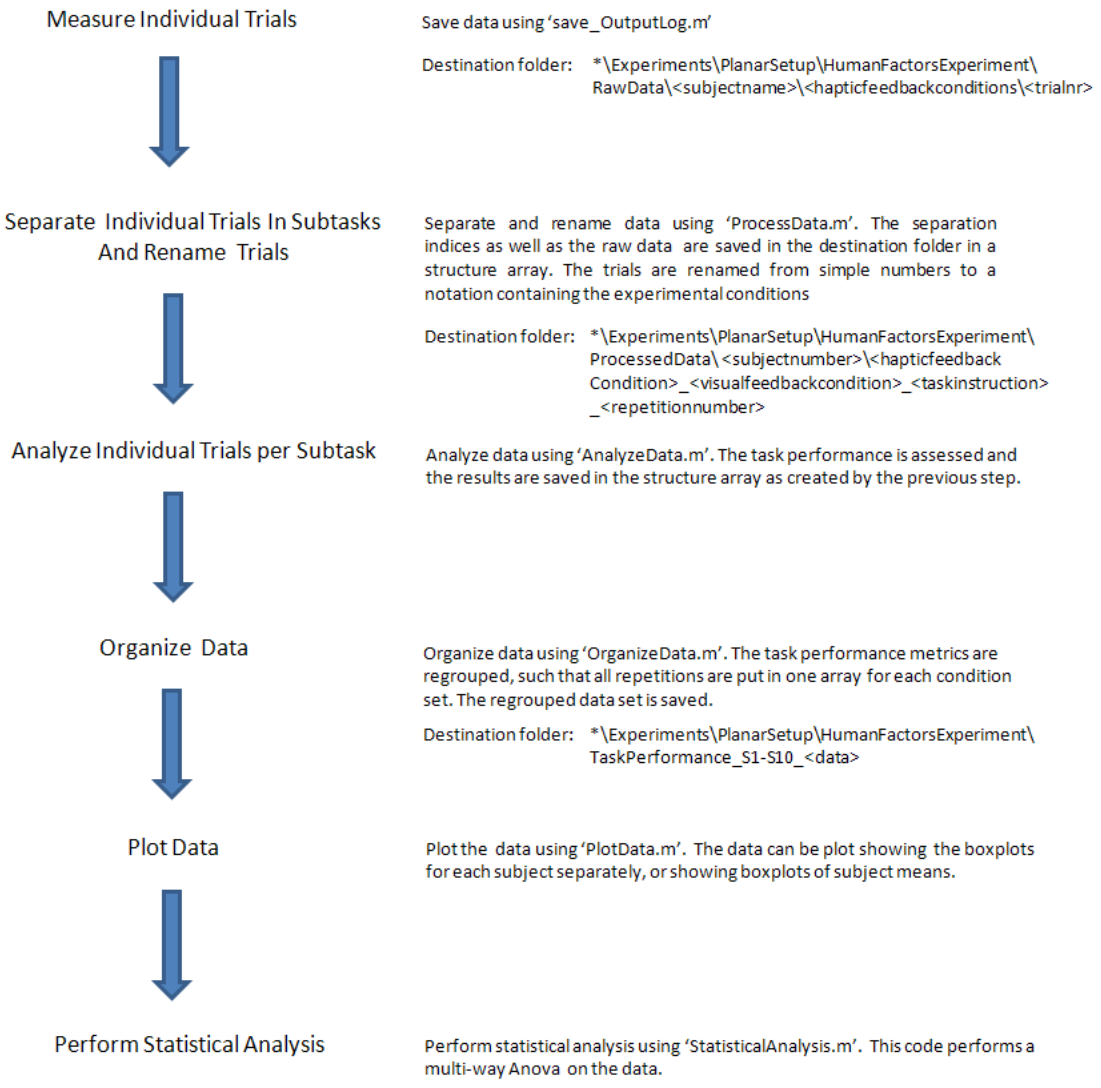


Figure D3: The flowchart of the data analysis algorithms

D4 Results

This appendix will present the time-to-complete, the NASA TLX Workload assessment, the subjective task performance and the failure rate for the entire bolt-and-spanner task. The results for the subtasks will be presented for according to Table D2. Notice that the statistical analysis of the data has only been performed on the metrics as defined in Appendix D2.

The measurement protocols, containing handwritten comments, and the questionnaires have only been added to the version of the appendices as handed over to the BioMechanical Engineering depository. The raw measurement data and Matlab code can be found on a USB-disk, that has also been submitted to the depository.

Abbreviations:

FAMT	Free Air Movement Task
CTT	Contact Transition Task
CPT	Constrained Position Task
FT	Force Task
Numb	The 'Numb' feedback condition; no haptic feedback is presented
LF	Low-Frequency Haptic Feedback
LFHF	Low-Frequency Haptic Feedback & High-Frequency Haptic Feedback
HandsOn	A maximal spectrum of haptic feedback; manual execution of the task.
V	Visual feedback
B	Blind, no visual feedback
A	Accurate, the task is performed accurately
F	Fast, the task is performed fast
M1	The Time-to-Complete
M2	Nominal Path Deviation
M3	Average Contact Forces
M4	Maximum Contact Forces
M5	Average Input Forces
M6	Maximum Input Forces
M7	Physical Workload
M8	Reversal Rate

Fig. D4 shows a bar chart of the entire time-to-complete for the bolt-and-spanner task, separated for the different subtasks. Fig. D5 shows the NASA TLX-results, while Fig. D6 shows the self-reported performance. Every subject is represented by a certain color and character.

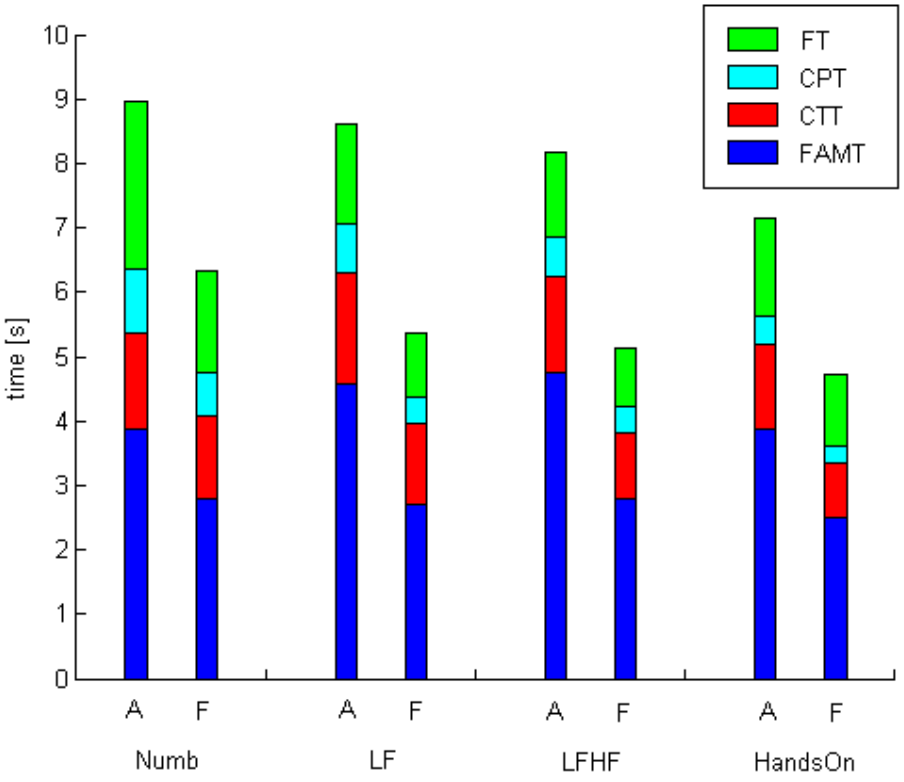


Figure D4: Bolt-and-Spanner Task, Time-to-Complete

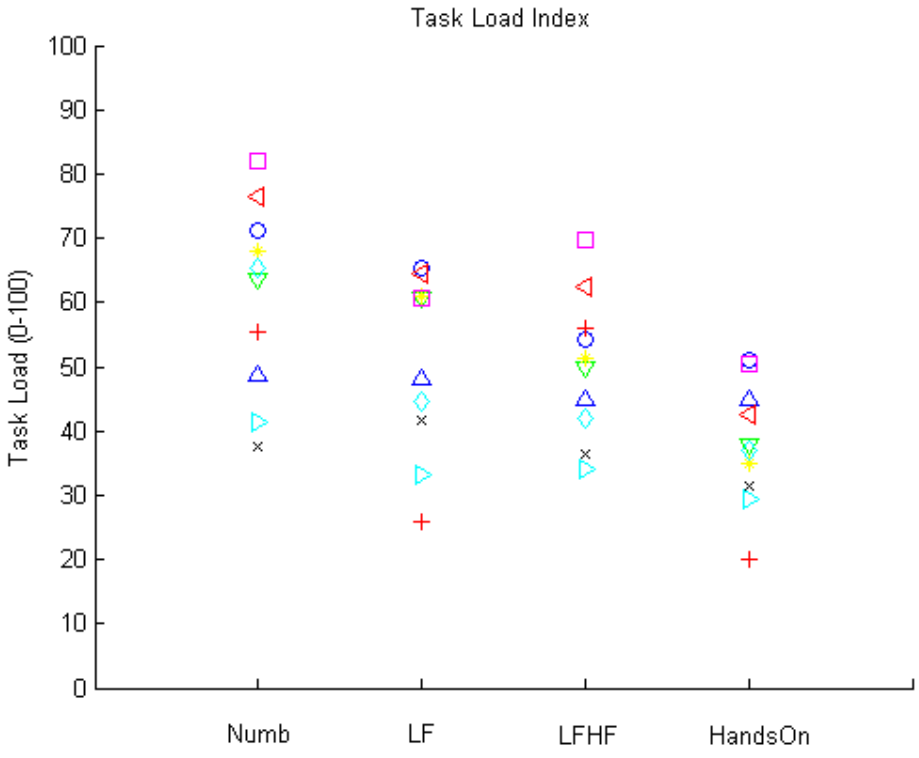


Figure D5: The NASA TLX-results (lower is better)

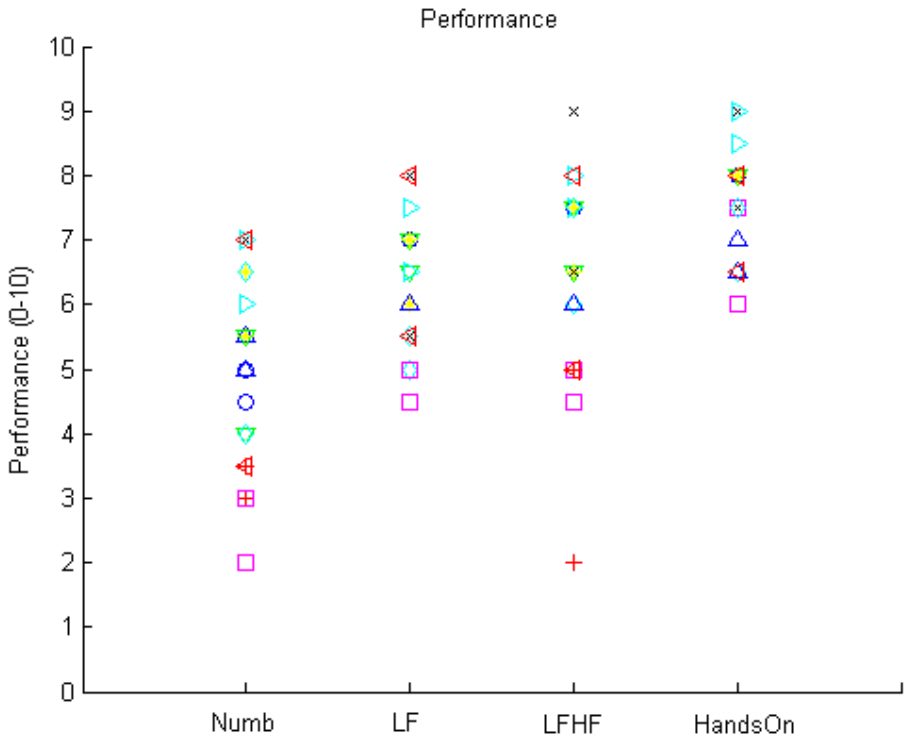


Figure D5: The self-reported performance (higher is better)

Free Air Movement Task

Plots of TP1 to TP12 for the Free Air Movement Task are shown in Fig. D6 to Fig. D14. The experimental results are compared in a three-way Anova, but only for the main task performance metrics (see Appendix D2). The main factor results are shown in Table D3. The factors F1, F2 and F3 respectively represent haptic feedback, task instruction and subject variation. Notice that the statistical analysis of the data has only been performed on the metrics as defined in Appendix D2.

Table D3: Anova results for the FAMT

	F1	F2	F3
M1	$p = 0.090$	$p < 0.001$	$p = 0.0093$
M2	$p = 0.086$	$p < 0.001$	$p < 0.001$
M5	$p = 0.014$	$p = 0.130$	$p < 0.001$
M6	$p < 0.001$	$p = 0.131$	$p < 0.001$
M7	$p = 0.005$	$p < 0.001$	$p < 0.001$
M8	$p = 0.497$	$p < 0.001$	$p = 0.010$

The third column, representing the subject variation, shows that the experiment is subjected to a large inter-variation. The second column shows that task instruction influences time-to-complete, nominal-path-deviation, operator workload and reversal rate, but does not influence the average and maximum input forces of the operator. In the first column it is shown that task performance in terms of average input forces (M5), maximum input force (M6) and physical workload (M7) is influenced by the type of haptic information displayed, while time-to-complete (M1), nominal-path-deviation (M2) and reversal rate (M8) are not influenced.

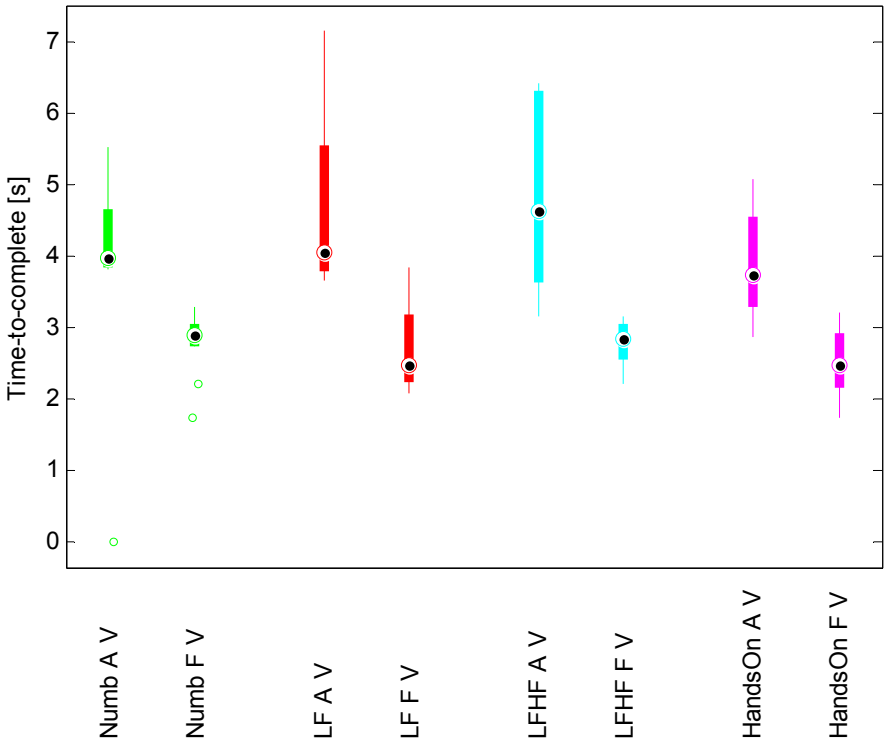


Figure D6: FAMT – Time-to-Complete (TP1)

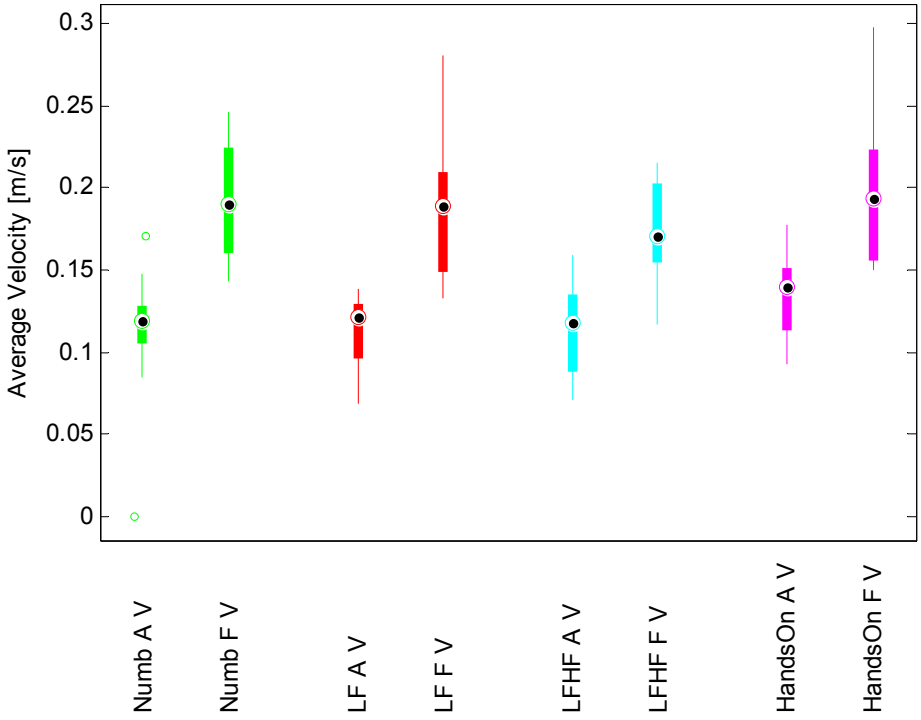


Figure D7: FAMT – Average Velocity (TP2)

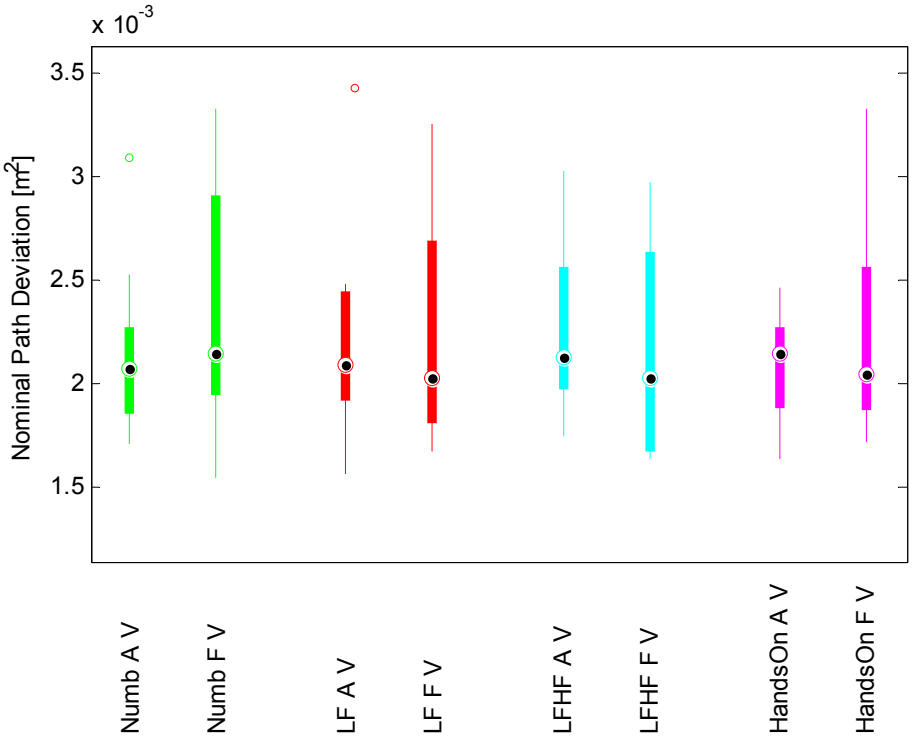


Figure D8: FAMT – Nominal Path Deviation (TP3)

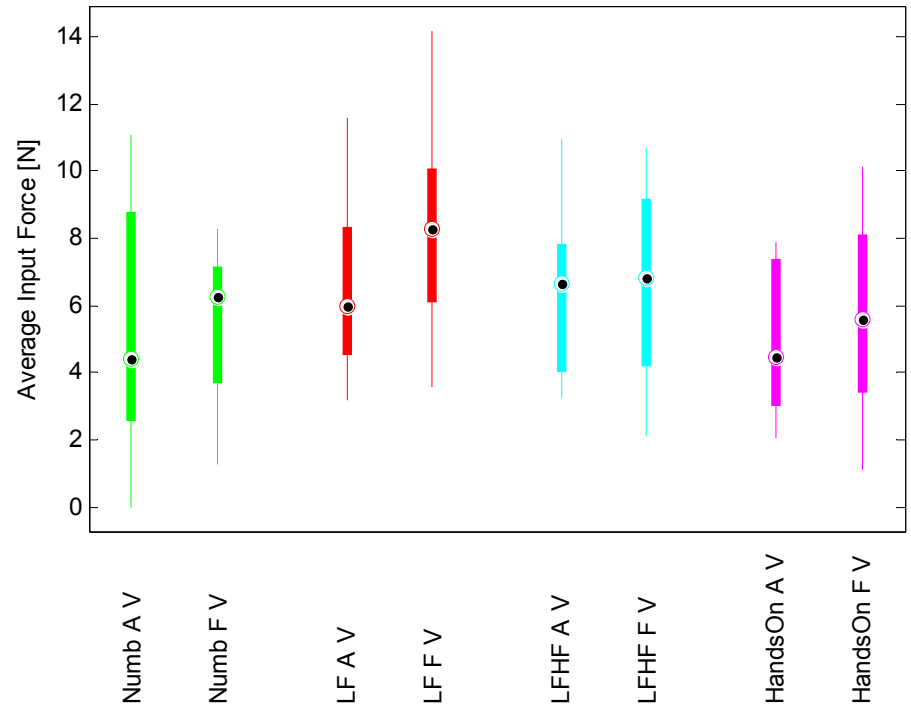


Figure D9: FAMT – Average Input Force (TP7)

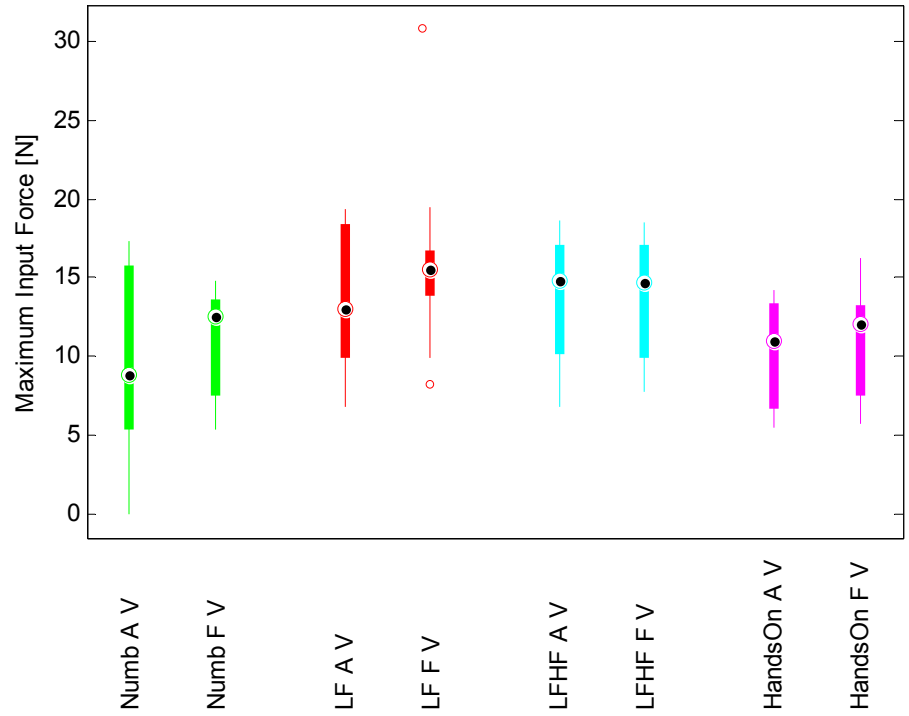


Figure D10: FAMT – Maximum Input Force (TP8)

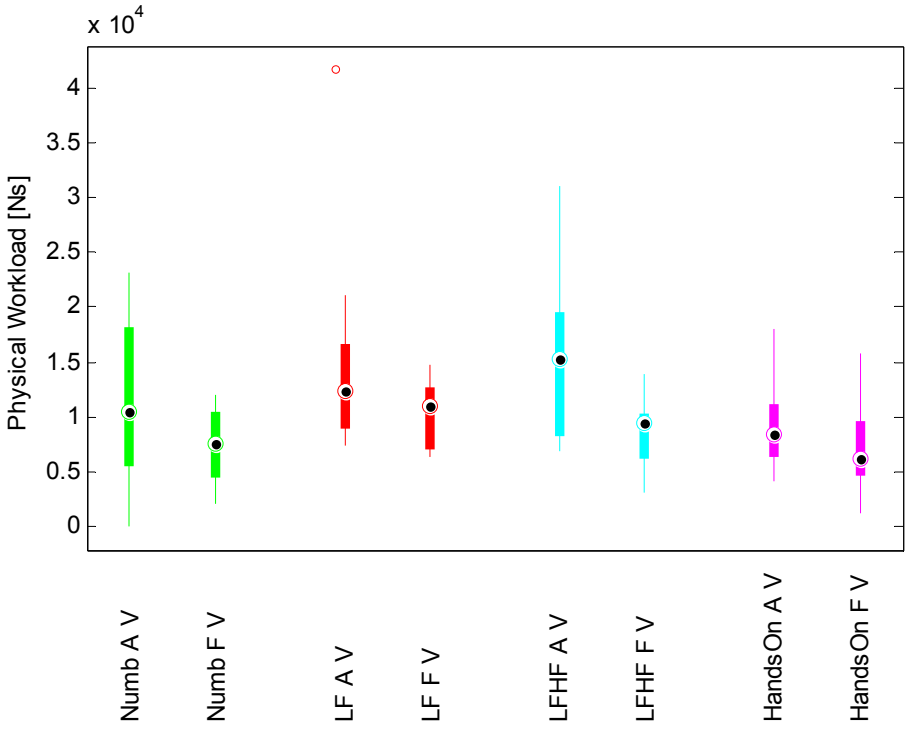


Figure D11: FAMT – Physical Workload (TP9)

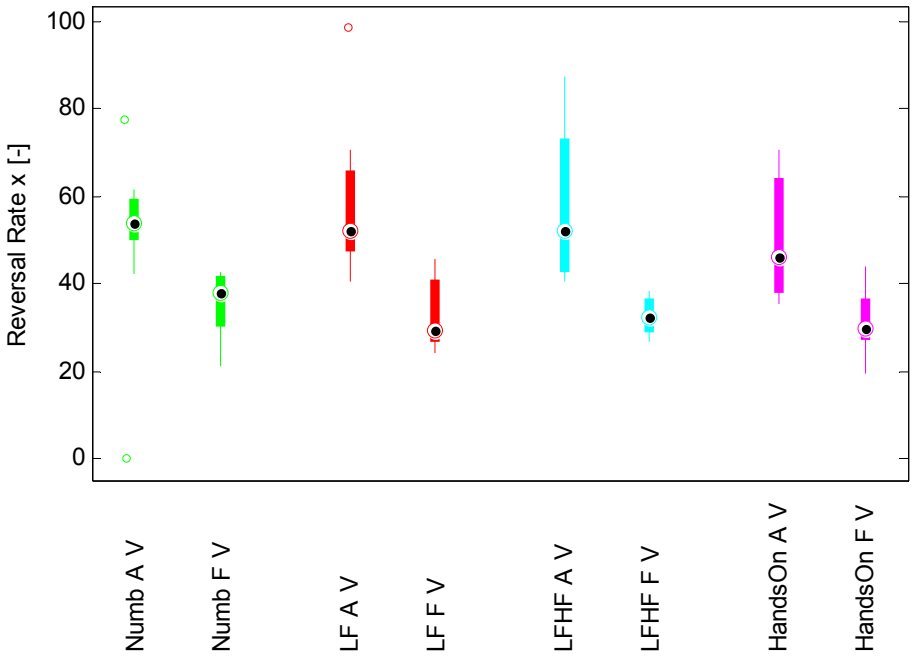


Figure D12: FAMT – Reversal Rate x-direction (TP10)

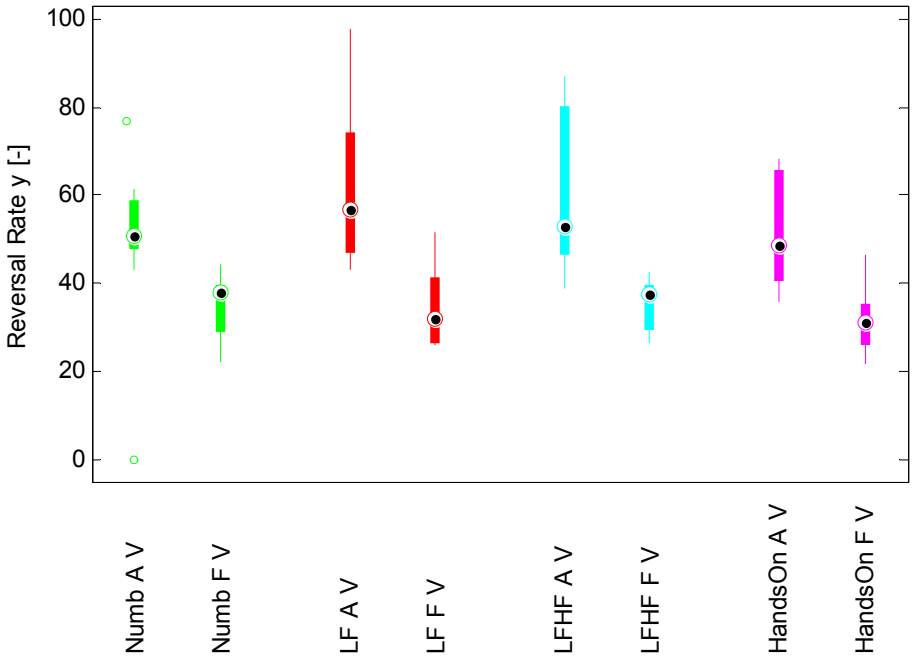


Figure D13: FAMT – MRR in the y-direction (TP11)

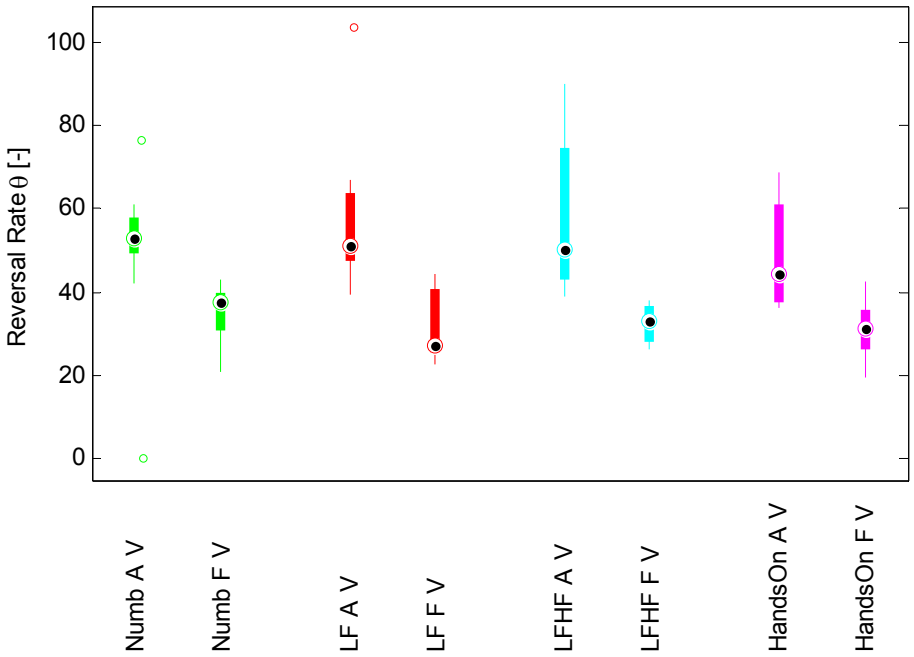


Figure D14: FAMT – MRR in the Θ-direction (TP12)

Contact Transition Task

Plots of TP1 to TP17 for the Free Air Movement Task are shown in Fig. D15 to Fig. D28. Compared to the FAMT, the CTT is performed with one addition condition, namely the visual feedback condition. Hence the experimental results for this subtasks are compared in a four-way Anova with factors F1, F2, F3 and F4 respectively representing haptic feedback and task instruction and subject variation. The results are shown in Table D4.

Table D4: Anova results for the CTT

	F1	F2	F3	F4
M1	$p = 0.034$	$p = 0.591$	$p = 0.037$	$p < 0.001$
M2	$p = 0.002$	$p = 0.500$	$p = 0.199$	$p = 0.004$
M3	$p < 0.001$	$p = 0.348$	$p = 0.912$	$p = 0.009$
M4	$p < 0.001$	$p = 0.312$	$p = 0.981$	$p = 0.149$
M5	$p = 0.121$	$p = 0.577$	$p = 0.946$	$p < 0.001$
M6	$p < 0.001$	$p = 0.260$	$p = 0.797$	$p < 0.001$
M7	$p = 0.173$	$p = 0.044$	$p = 0.580$	$p = 0.015$
M8	$p = 0.063$	$p = 0.048$	$p = 0.621$	$p < 0.001$

A first observation is that the subject intervariance is large (F4). Secondly, whether subjects are asked to complete the task accurately or fast (F3), only the time-to-complete is affected measurably. Regarding visual or blind task performance (F2) it shows that workload, either measured physically (M7) or as the reversal rate (M8), is reduced 8 when providing visual feedback. Finally, the task performance depends on haptic feedback for time-to- complete (M1), nominal path deviation (M2), contact forces (M3 & M4), maximum input force (M6), while physical workload (M7) and reversal rate are not influenced (M8).

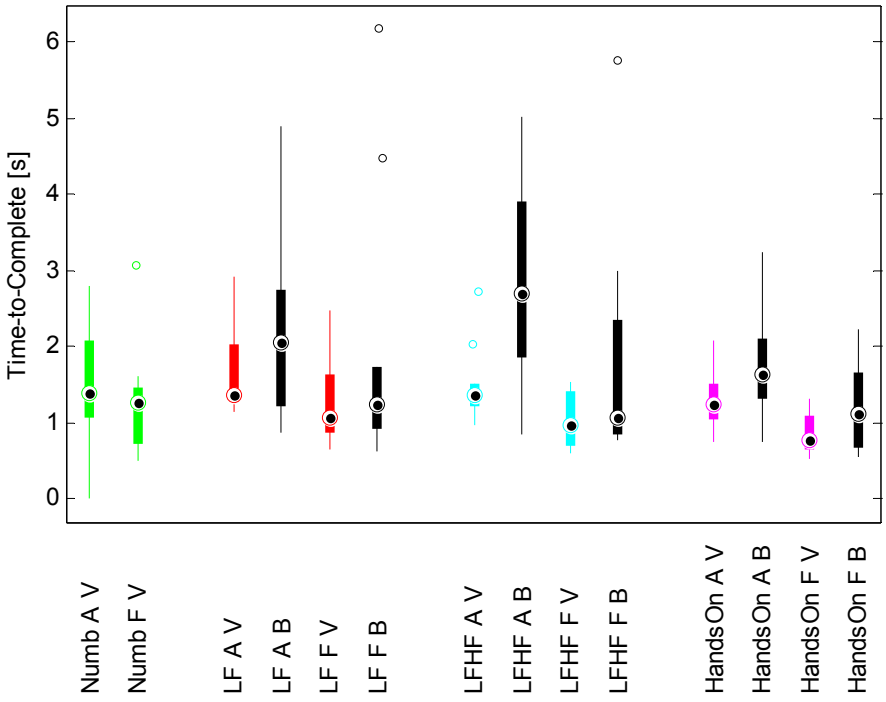


Figure D15: CTT – Time-to-Complete (TP1)

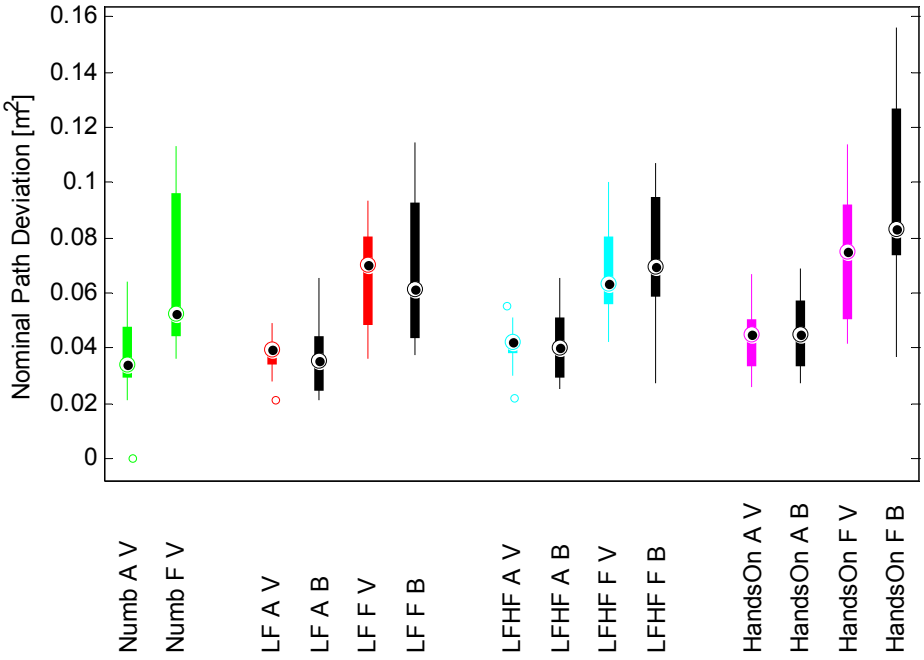


Figure D16: CTT – Nominal Path Deviation (TP2)

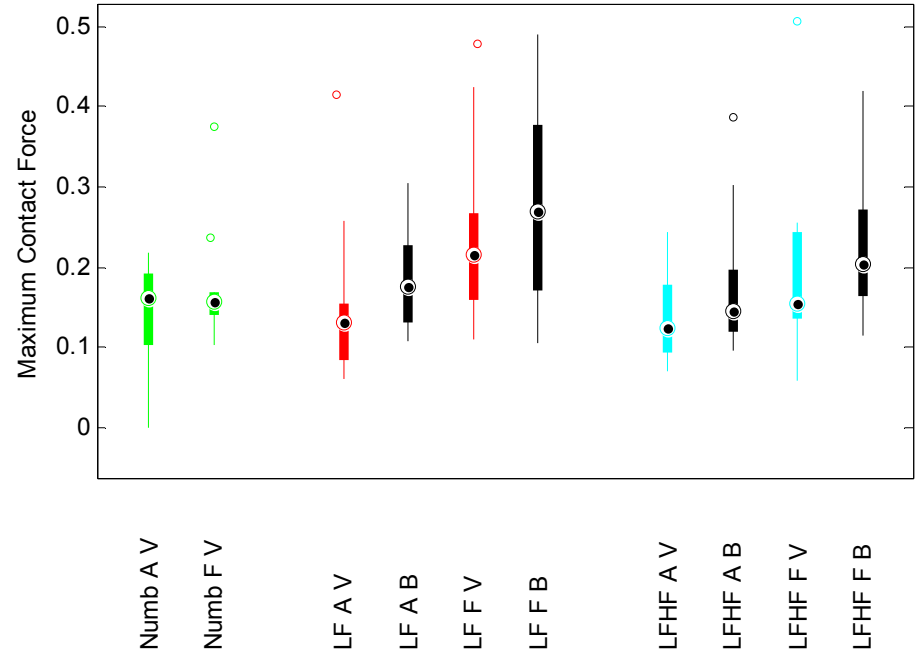


Figure D17: CTT – Maximum Contact Force (TP5)

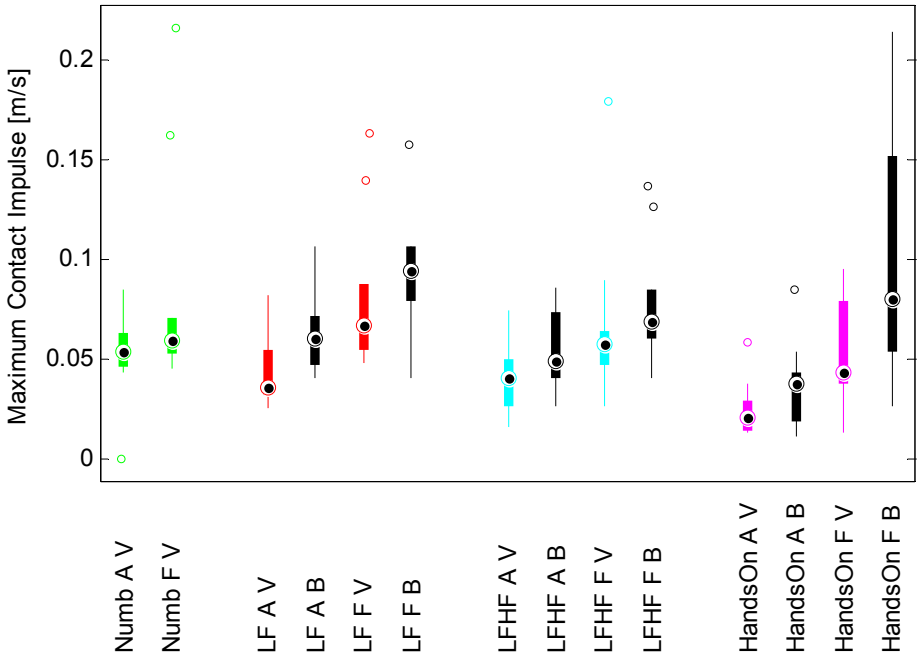


Figure D18: CTT – Maximum Contact Impulse (TP6)

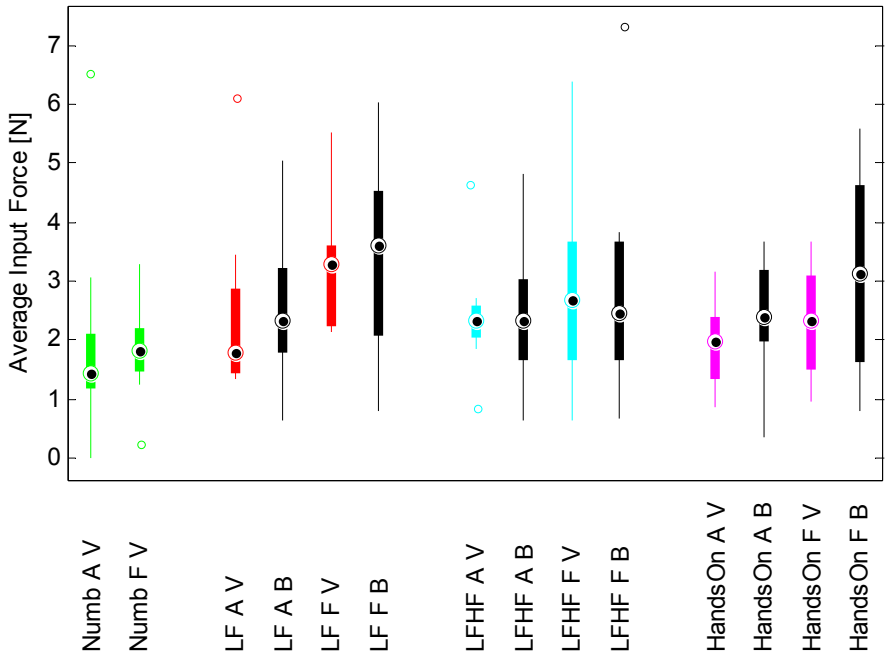


Figure D19: CTT – Average Input Force (TP7)

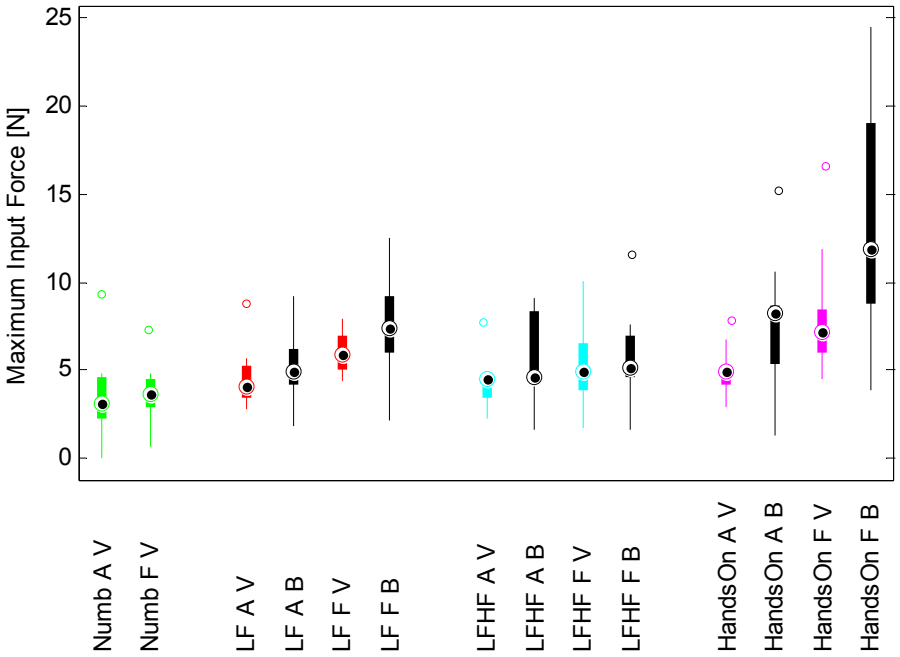


Figure D20: CTT – Maximum Input Force (TP8)

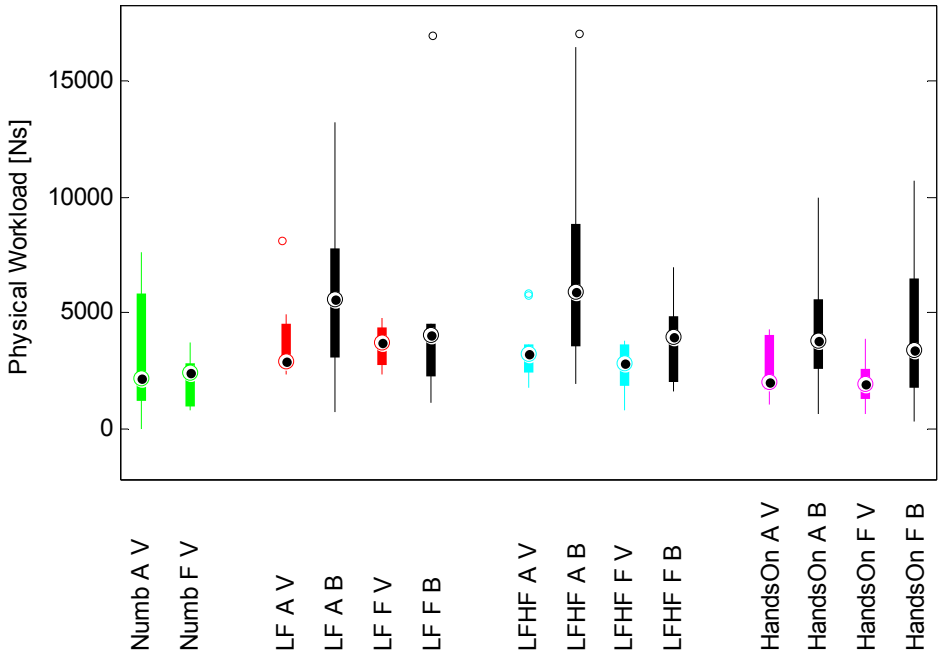


Figure D21: CTT - Physical Workload (TP9)

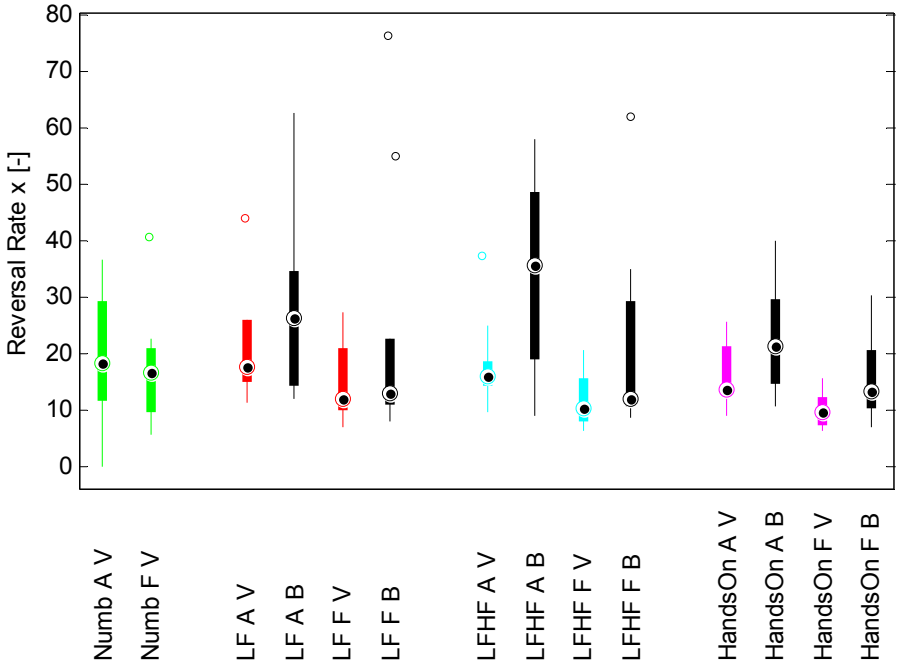


Figure D22: CTT – Reversal Rate in x-direction (TP10)

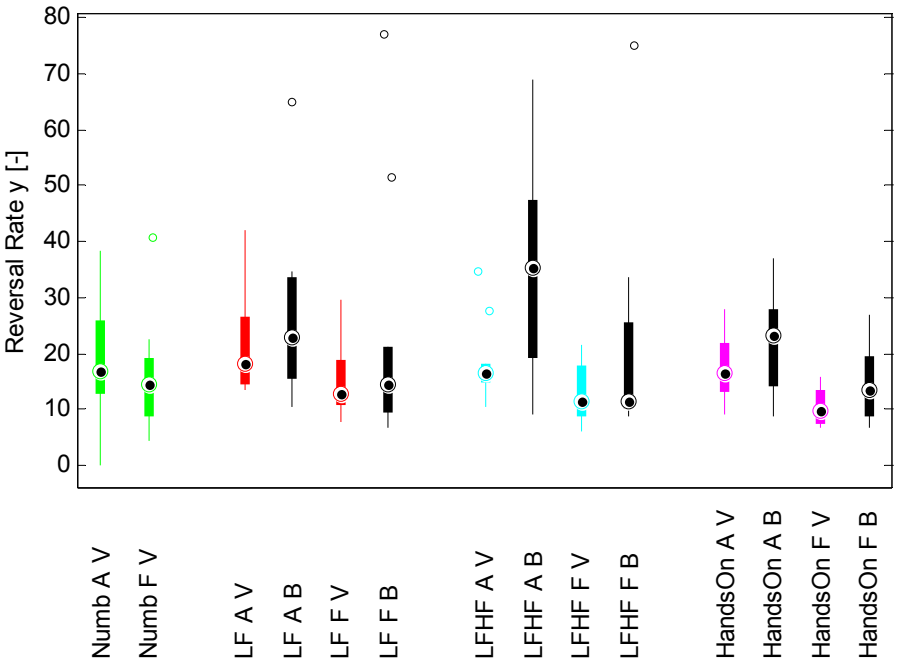


Figure D23: CTT – Reversal Rate in y-direction (TP11)

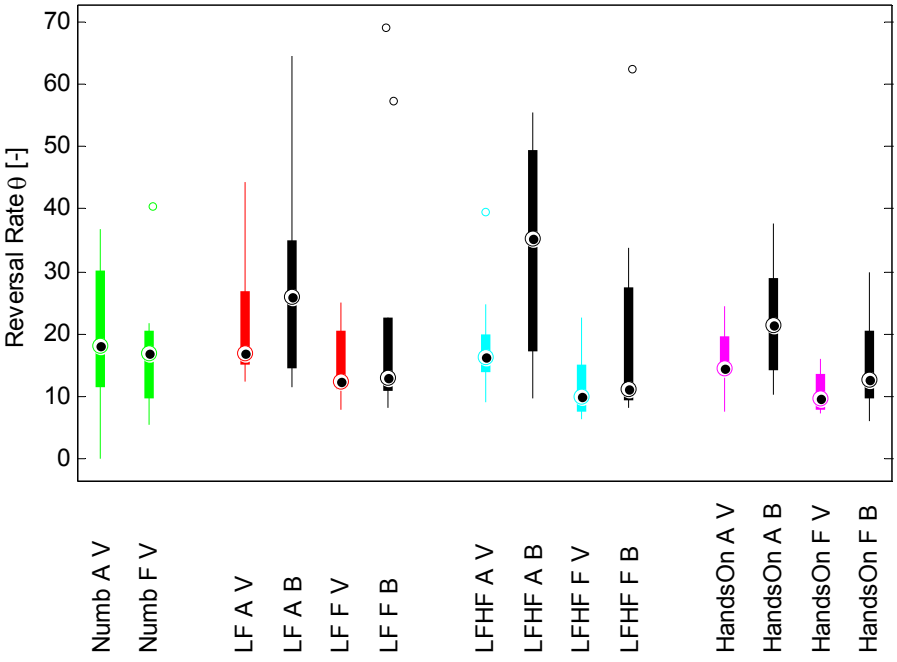


Figure D24: CTT – Reversal Rate in Θ-direction (TP12)

The measured human operator input force in the direction of contact shows a rapid decrease after attaining contact. This suggests that providing HF reduces the human operators input force shortly after contact, but only when tasks are performed fast ($p < 0.001$). In order to find out if this force drop is caused by a reflex or a conscious decision, this phenomenon is analyzed in more detail for the LF and LFHF conditions. A maximum absolute difference in operator input force is defined, a force 'drop', as well as an average input force. The signals are analyzed for intervals of 100ms and 300ms.

For the average forces no significant differences were found. As Fig. 26 and Fig. 28 illustrate, a force drop is found for both intervals. If the drop would have been found only in the 100ms interval, it would have meant that the force drop is a result of a reflex. With this results, the force drop can be either a conscious decision by the human operator or a reflex. However as stated before , the definition of the task completion criteria does not allow for proper analysis of these metrics; the meaning of TP14-TP17 is unclear as the exact moment where the high-frequency pulse is generated is not very well defined.

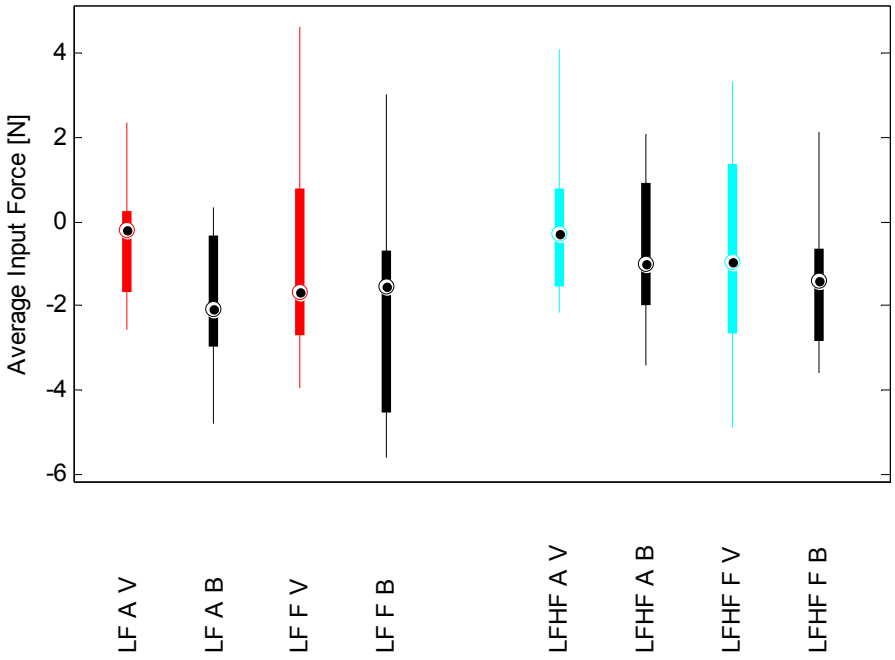


Figure D25: CTT – Average Input Force 100ms after Contact (TP14)

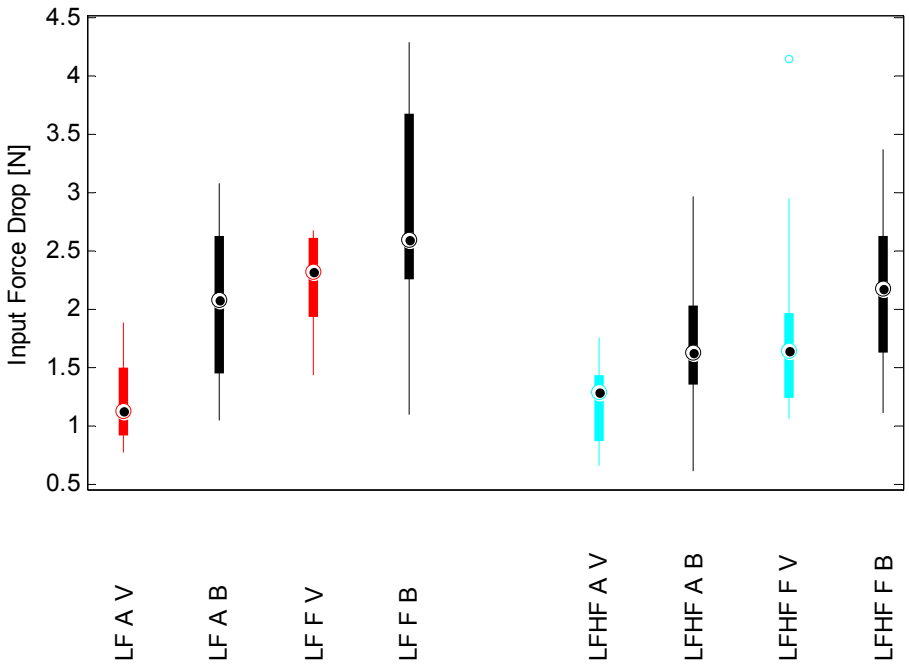


Figure D26: CTT –Input Force Drop 100ms after Contact (TP15)

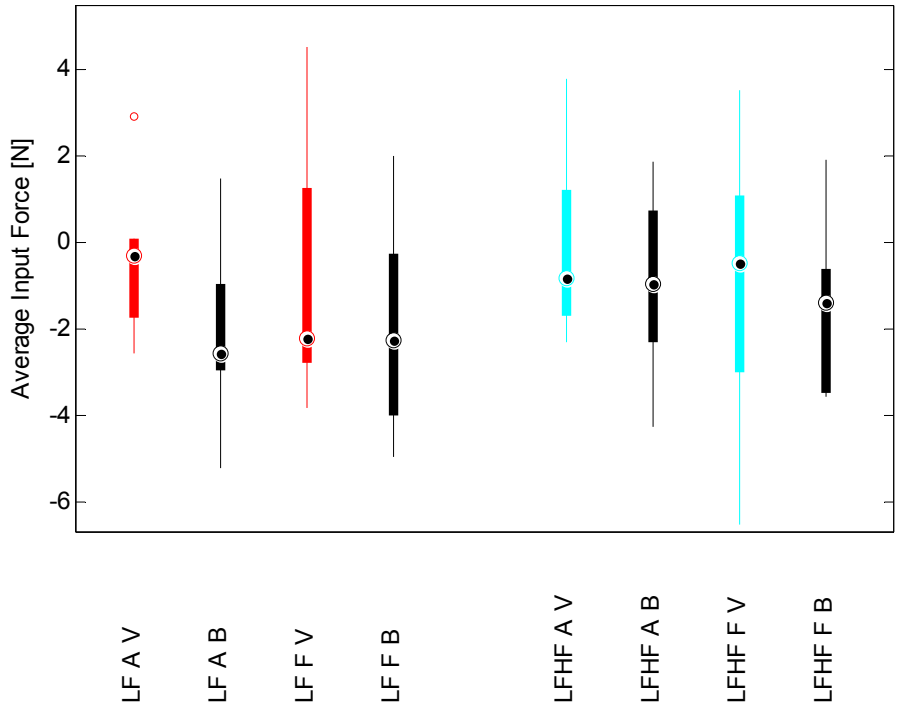


Figure D27: CTT – Average Input Force 300ms after Contact (TP16)

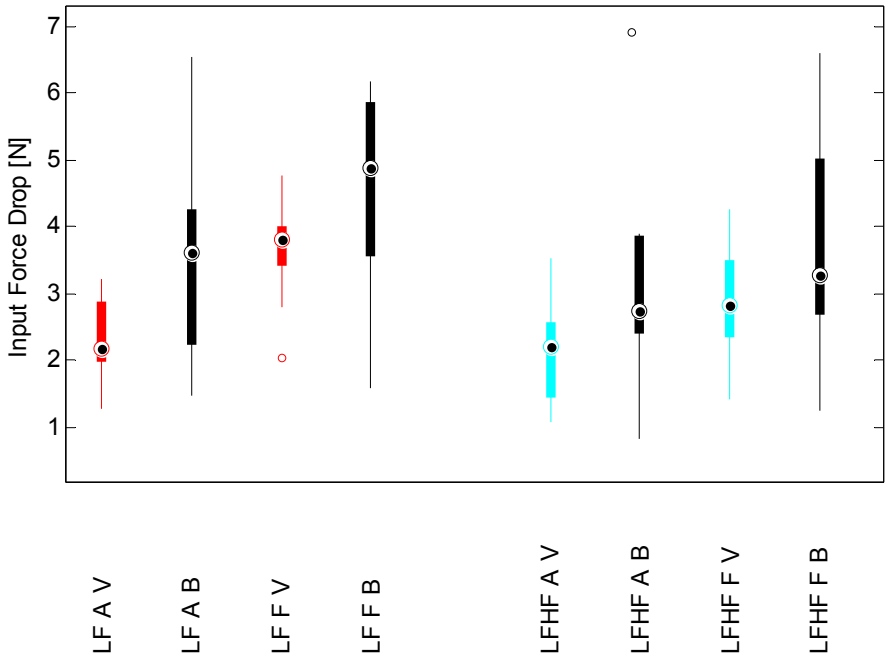


Figure D28: CTT – Input Force Drop 300ms after Contact (TP17)

Constrained Position Task

Plots of TP1 to TP12 for the Constrained Position Task are shown in Fig. D29 to Fig. D39. The factors for the statistical analysis (multi-way Anova) are the same factors as used for the CTT Task, namely F1, F2, F3, F3 and F4. The results are shown in Table D5.

Table D5: The Anova results of the Constrained Position Task

	F1	F2	F3	F4
M1	$p < 0.001$	$p < 0.001$	$p = 0.026$	$p < 0.001$
M3	$p < 0.001$	$p = 0.312$	$p = 0.860$	$p = 0.077$
M4	$p < 0.001$	$p = 0.577$	$p = 0.866$	$p = 0.032$
M5	$p = 0.003$	$p = 0.110$	$p = 0.854$	$p < 0.001$
M6	$p < 0.001$	$p = 0.345$	$p = 0.804$	$p < 0.001$
M7	$p = 0.003$	$p < 0.001$	$p = 0.118$	$p = 0.005$
M8	$p < 0.001$	$p < 0.001$	$p = 0.022$	$p < 0.001$

From Table D5 it is obtained that task instruction (F3) influences time-to-complete (M1) and reversal rate (M8). Providing visual feedback yields a reduction in time-to-complete (M1) and physical workload (M7) and reversal rate (M8). The type of haptic information displayed influence the performance for all defined performance metrics.

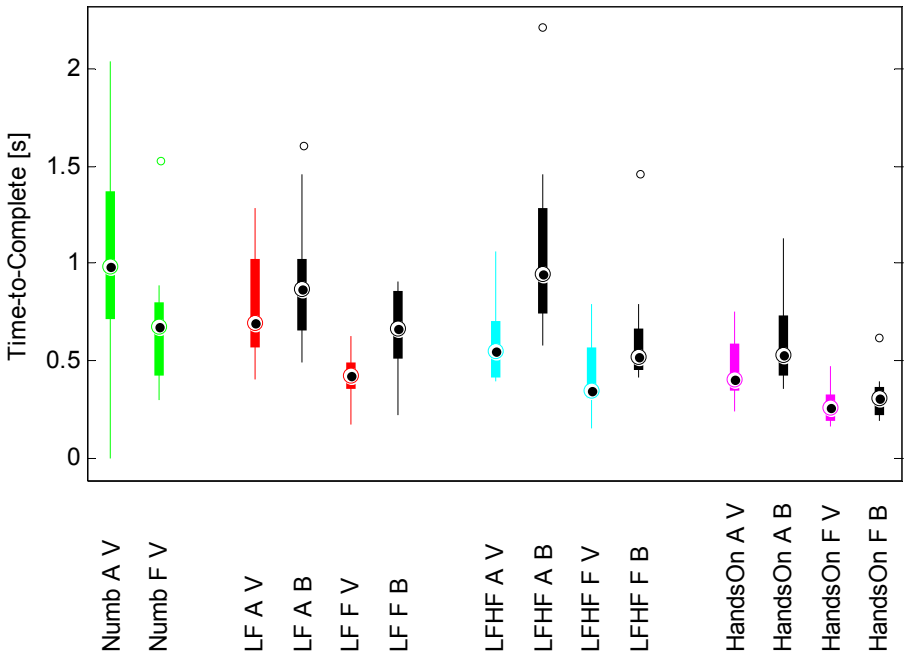


Figure D29: CPT – Time-to-Complete (TP1)

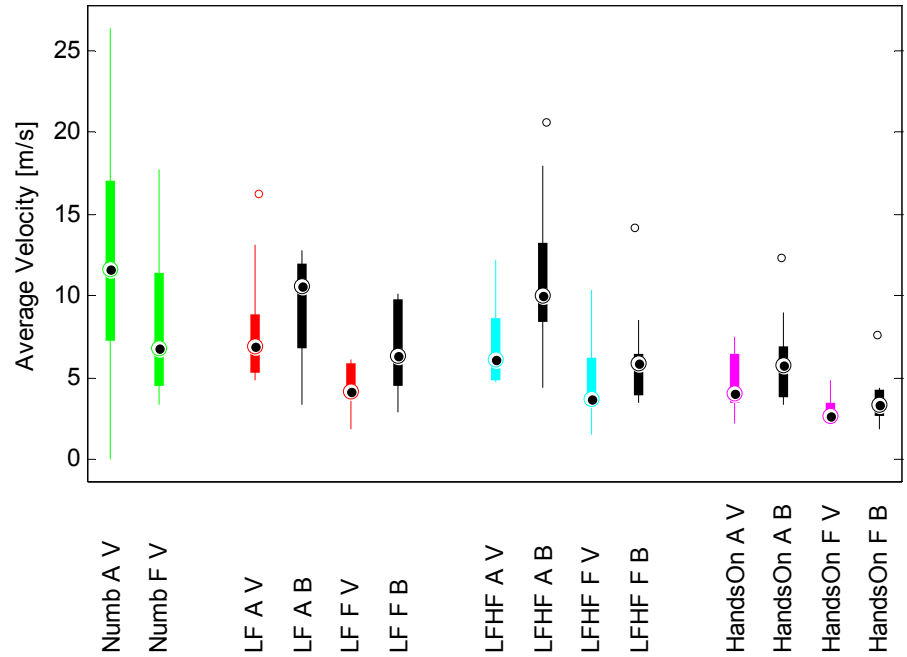


Figure D30: CPT – Average Velocity (TP2)

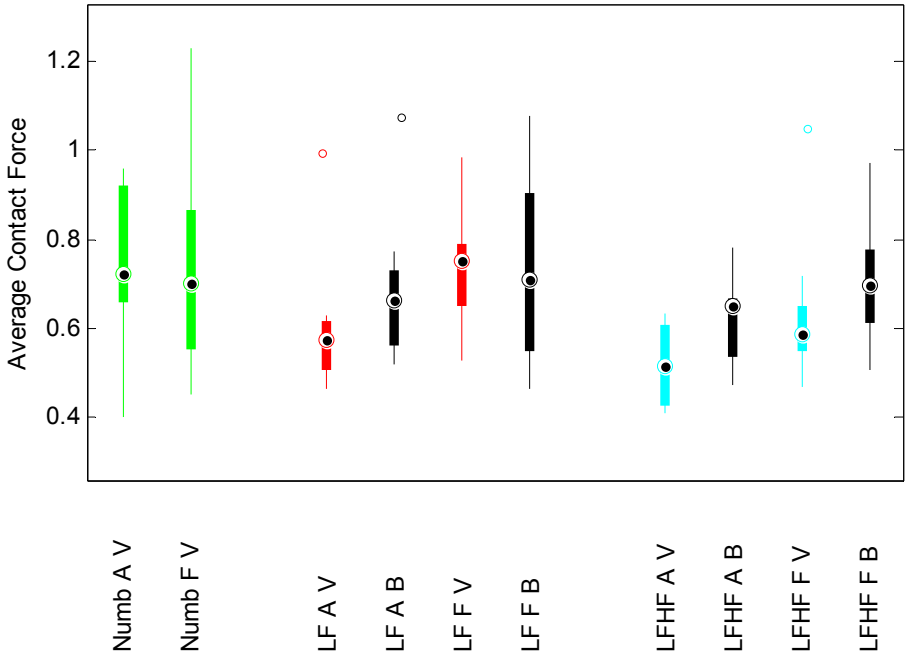


Figure D31: CPT – Average Contact Force (TP4)

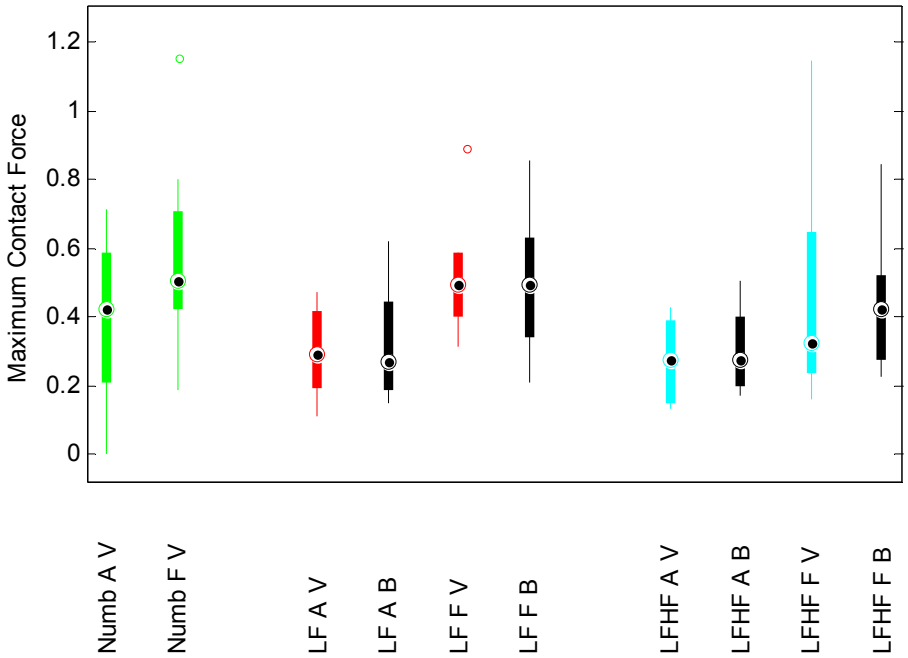


Figure D32: CPT – Maximum Contact Force (TP5)

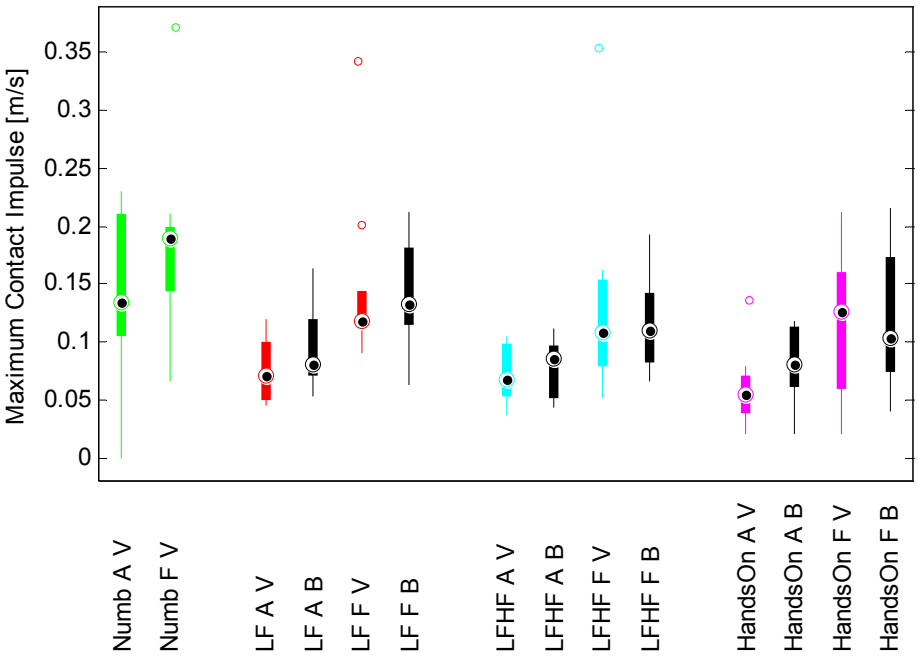


Figure D33: CPT – Maximum Contact Impulse (TP6)

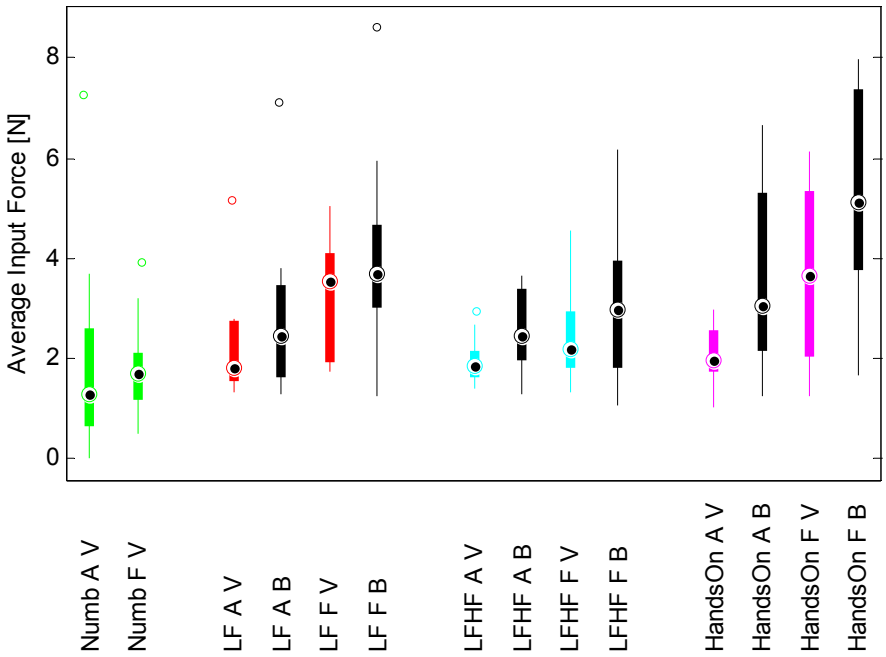


Figure D34: CPT – Average Input Force (TP7)

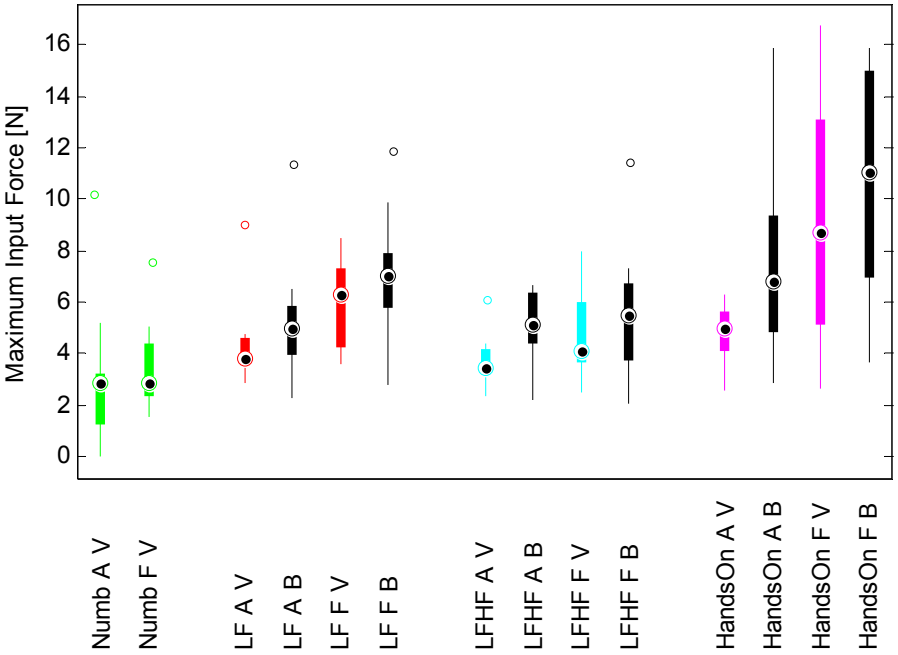


Figure D35: CPT – Maximum Input Force (TP8)

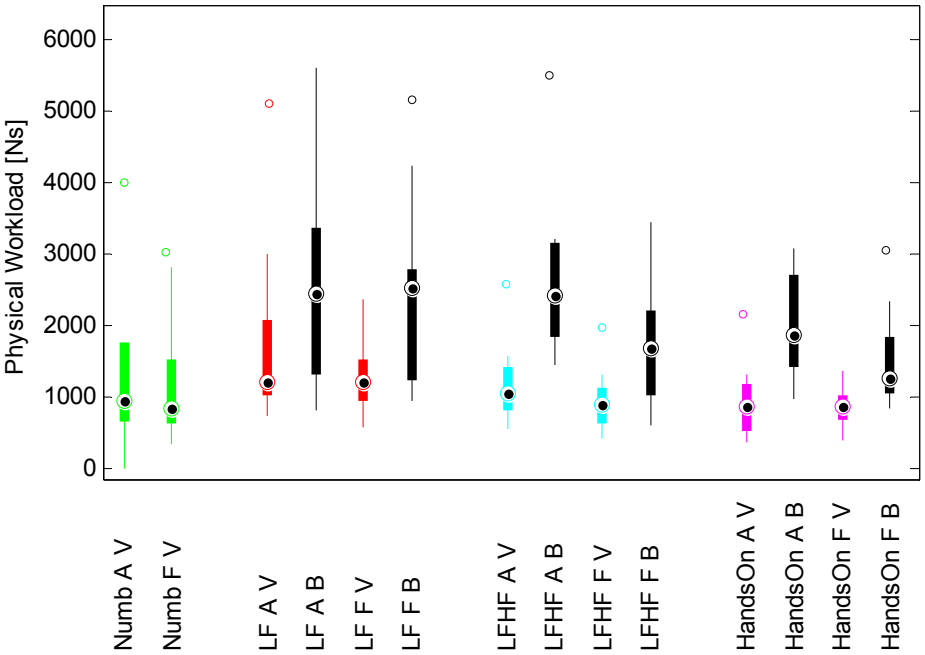


Figure D36: CPT – Physical Workload (TP9)

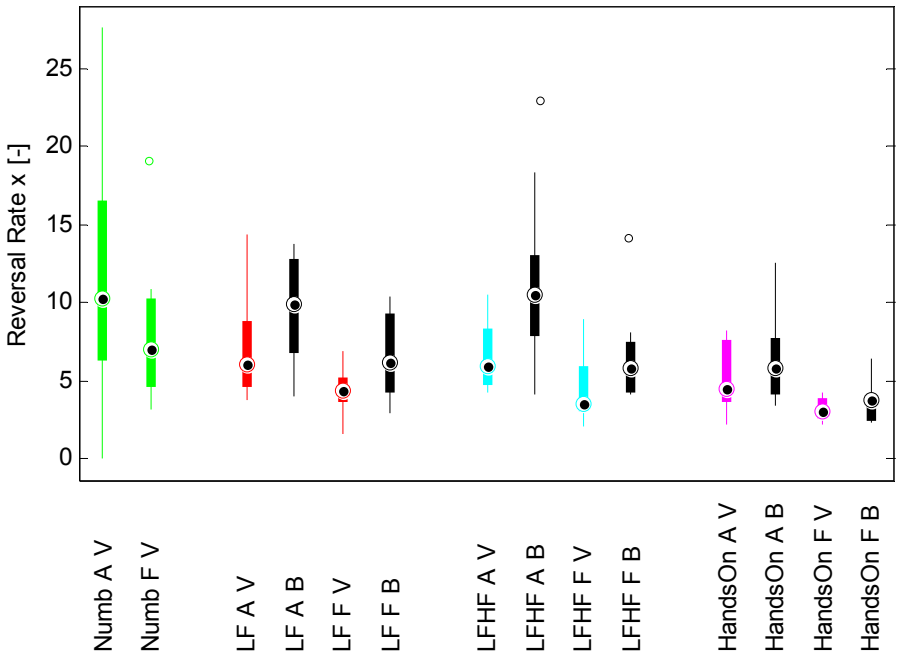


Figure D37: CPT – Reversal Rate in x-direction (TP10)

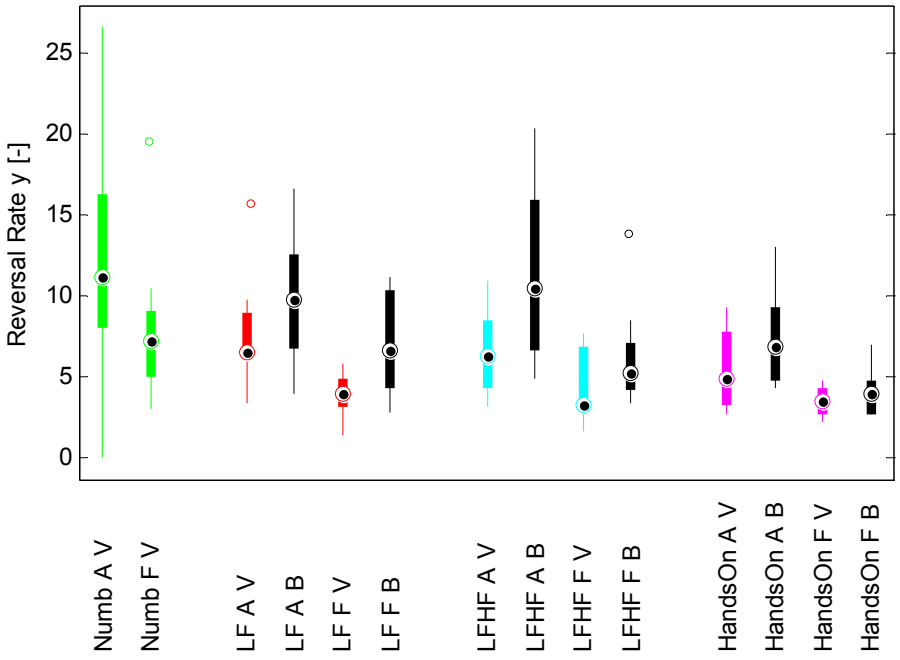


Figure D38: CPT – Reversal Rate in y-direction (TP11)

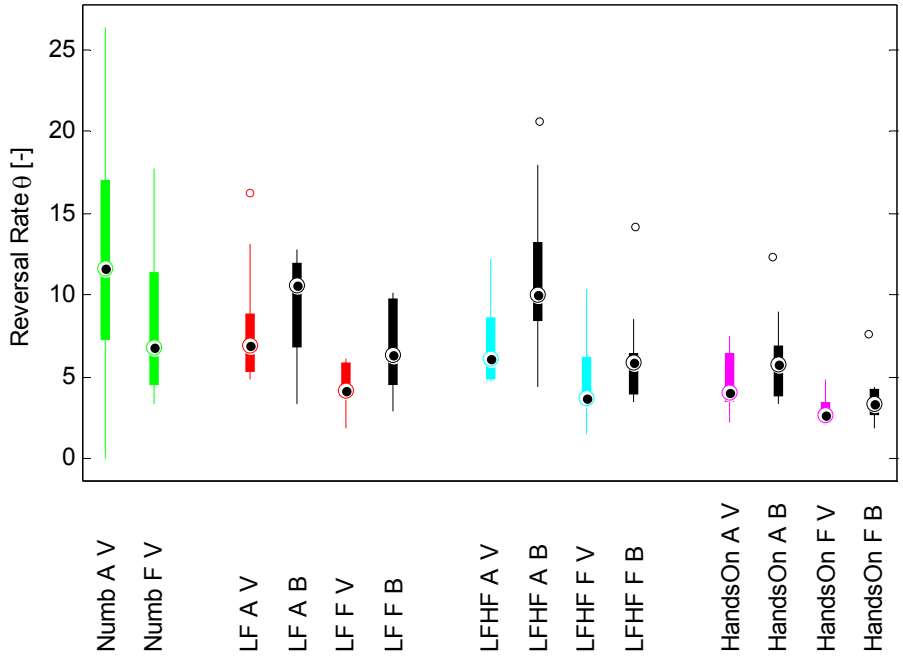


Figure D39: CPT – Reversal Rate in Θ-direction (TP11)

Force Task

Plots of TP1 to TP13 for the Force are shown in Fig. D40 to Fig. D51. The factors for the statistical analysis are the same factors as used for the CTT and CPT task, namely F1, F2, F3, F3 and F4. The results are shown in Table D6.

Table D6: The Anova results of the Force Task

	F1	F2	F3	F4
M1	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p = 0.049$
M3	$p < 0.001$	$p = 0.608$	$p = 0.817$	$p = 0.010$
M4	$p < 0.001$	$p = 0.533$	$p = 0.890$	$p = 0.003$
M5	$p = 0.076$	$p = 0.541$	$p = 0.750$	$p < 0.001$
M6	$p < 0.001$	$p = 0.535$	$p = 0.642$	$p < 0.001$
M7	$p = 0.043$	$p = 0.053$	$p = 0.375$	$p = 0.002$
M8	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p = 0.083$

From Table D5 it i It is shown that task instruction (F3) and the visual feedback condition influence time-to-complete (M1) and reversal rate (M8) for the FT ($p < 0.001$). Haptic feedback influences all the task performance metrics ($p \leq 0.043$), except the average operator input force.

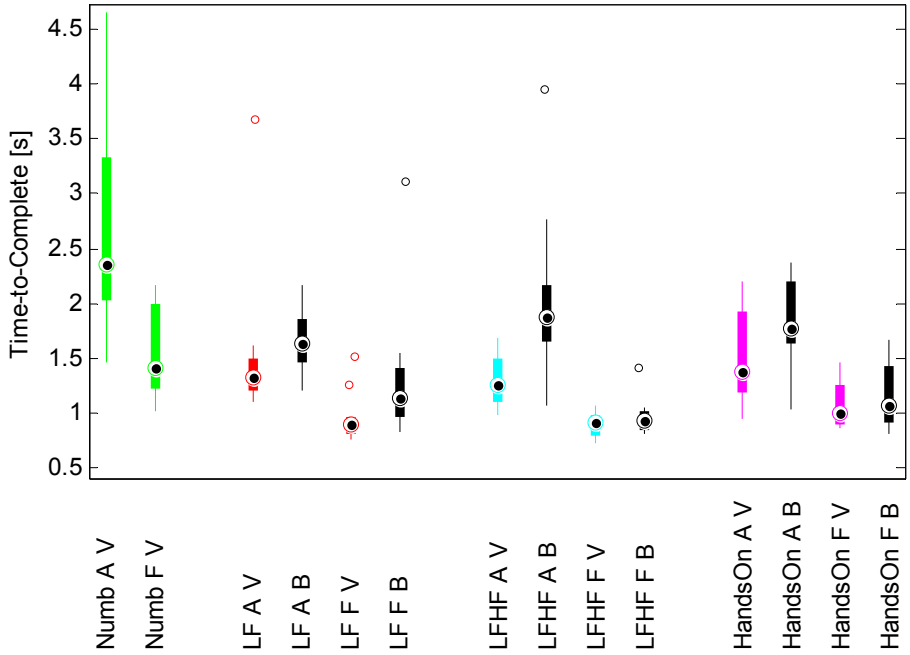


Figure D40: FT – Time-to-Complete (TP1)

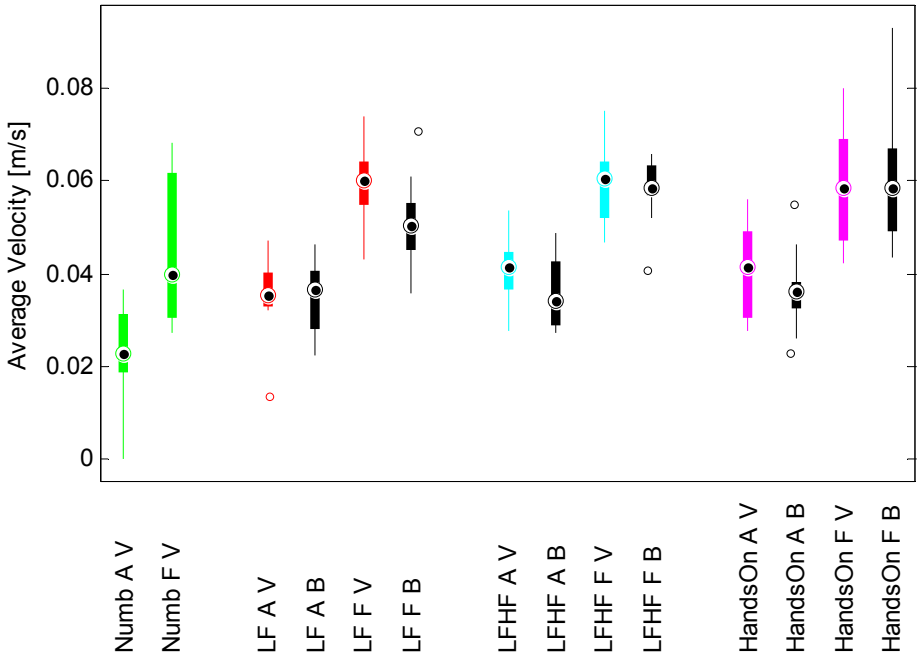


Figure D41: FT – Average Velocity (TP2)

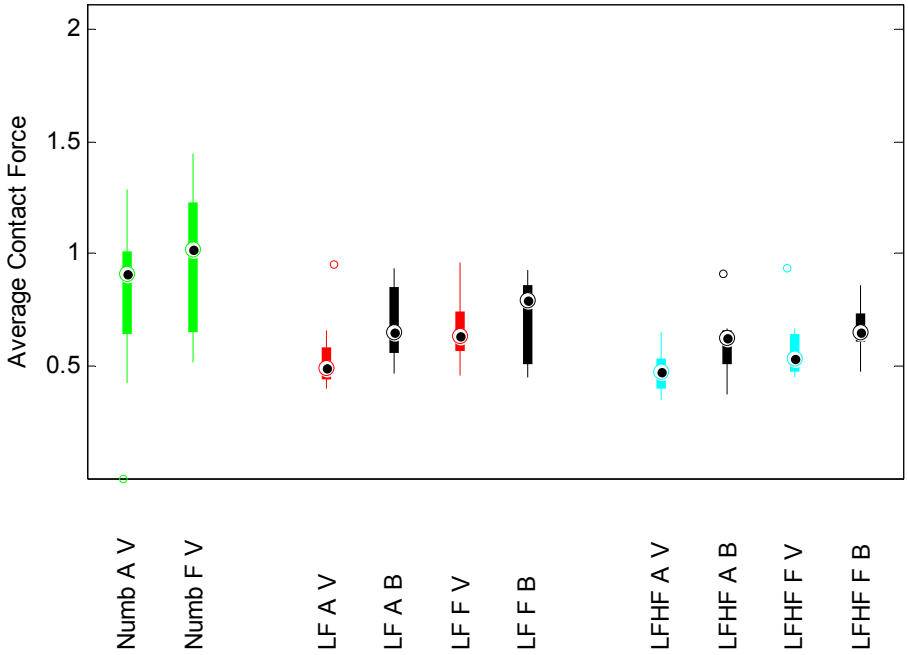


Figure D42: FT – Average Contact Force (TP4)

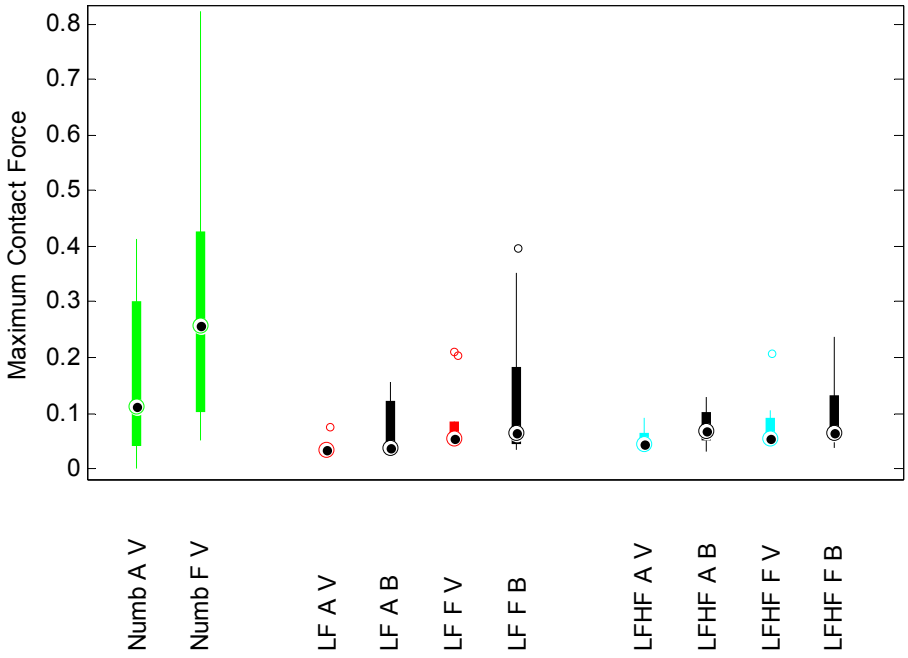


Figure D43: FT – Maximum Contact Force (TP5)

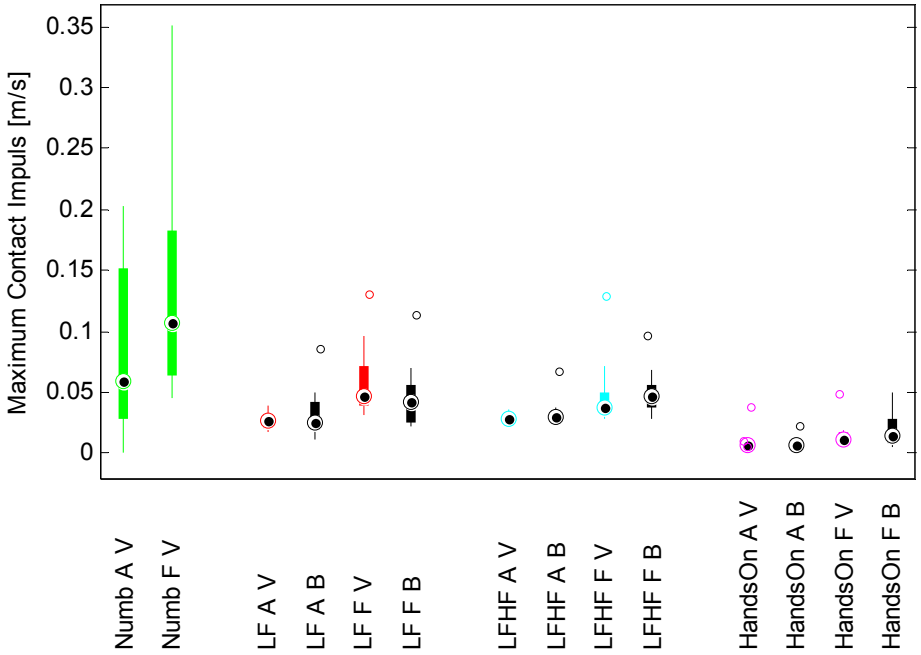


Figure D44: FT – Maximum Contact Impulse (TP6)

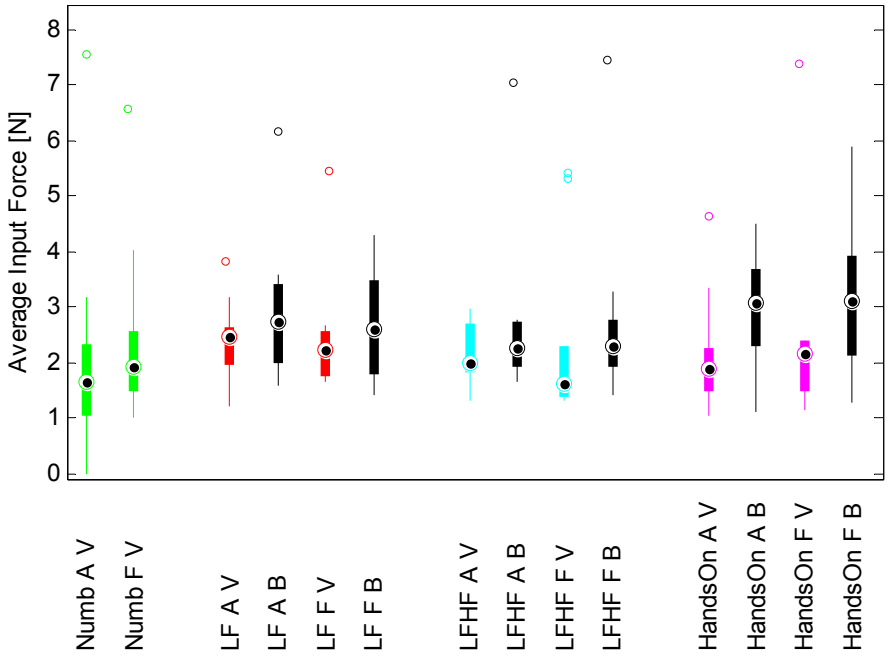


Figure D45: FT – Average Input Force (TP7)

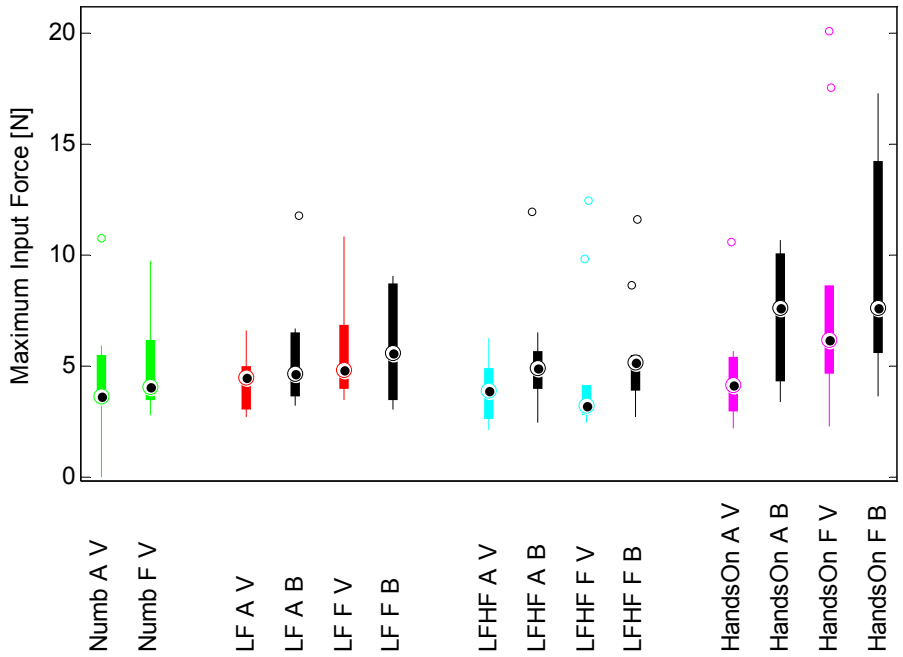


Figure D46: FT – Maximum Input Force (TP8)

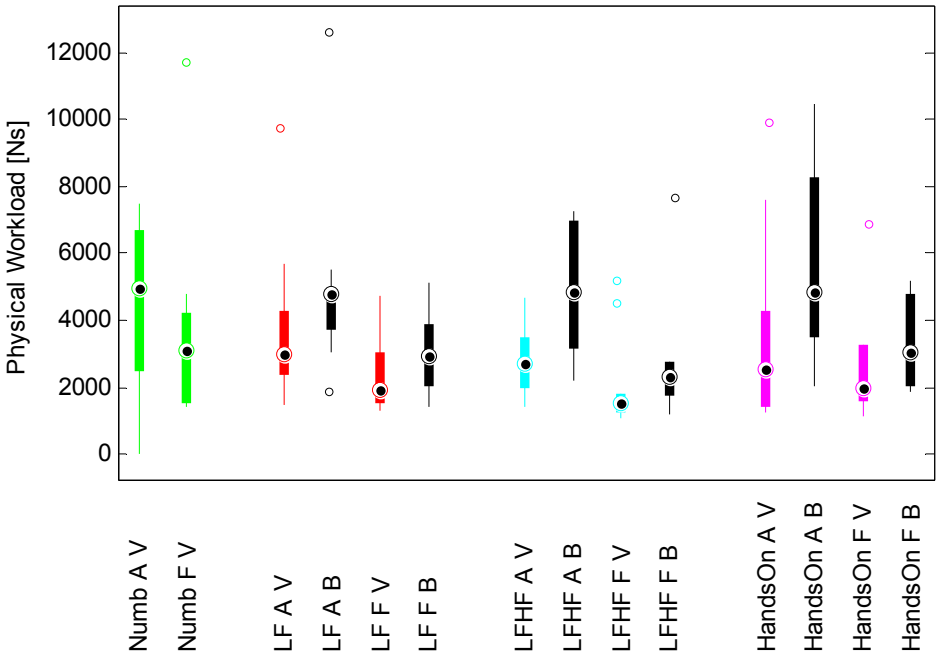


Figure D47: FT – Physical Workload (TP9)

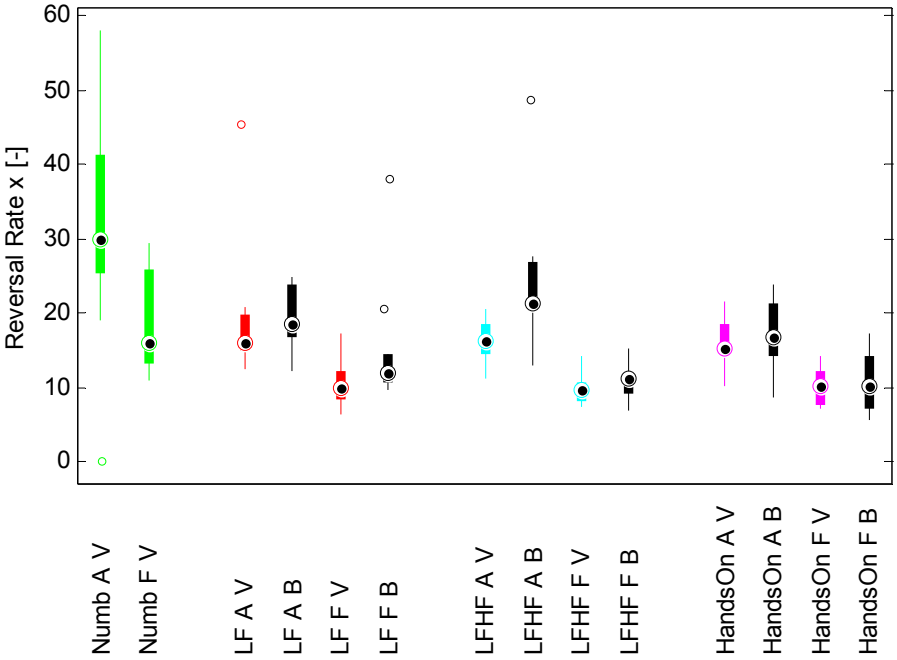


Figure D48: FT – Reversal Rate in x-direction (TP10)

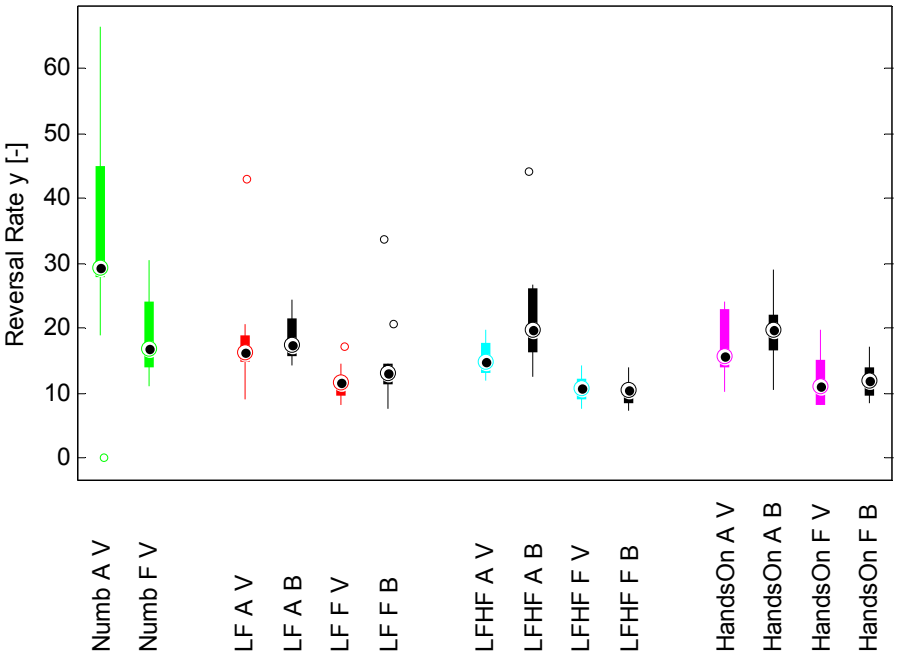


Figure D49: FT – Reversal Rate in y-direction (TP11)

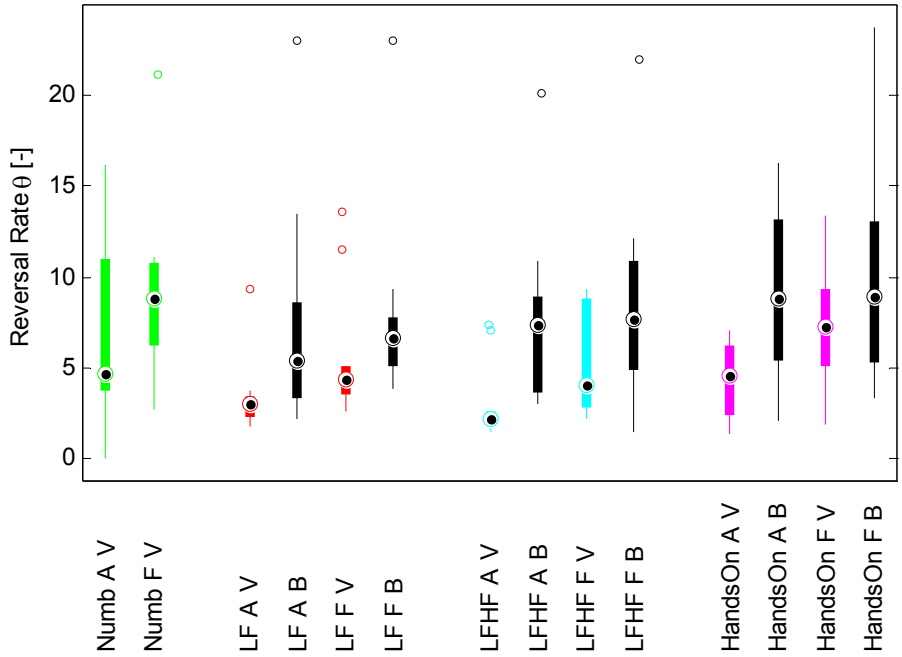


Figure D50: FT – Reversal Rate in Θ-direction (TP12)

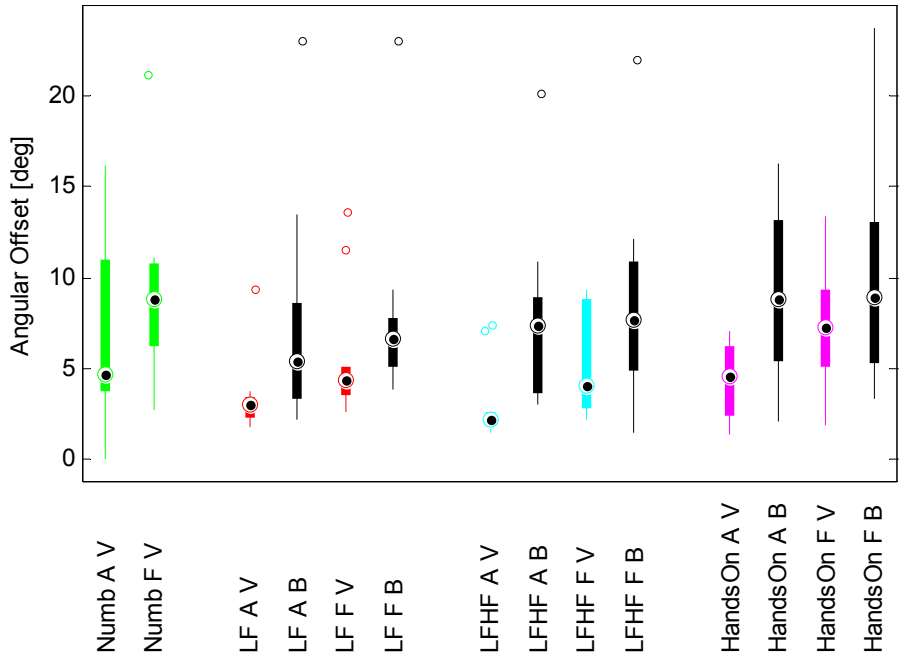


Figure D51: FT – Angular Offset (TP13)

References

Aliaga et al., *"Experimental Quantitative Comparison of Different Control Architectures for Master–Slave Teleoperation,"* IEEE Transactions on Control System Technology, No. 1, January 2004

T.L. Brooks, *"Telerobotic response requirements,"* Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, p. 113-120, 1990

G.A.V. Christiansson, *"Measuring asymmetric haptic teleoperation device properties,"* Procedures of the International Conference on Systems, Man and Cybernetics," The Hague, Netherlands, pp 2454–2458, Oct 2004

G.A.V. Christiansson, Matthias Mulder, and Frans C. T. van der Hel, *"Slave device stiffness and teleoperator stability,"* EuroHaptics Conference, Paris, France, 2006a

G.A.V. Christiansson, *"Hard Master Soft Slave Haptic Teleoperation*," PhD Dissertation, Delft University of Technology, The Netherlands, 2007a

G.A.V. Christiansson, *"A novel 3-DOF Planar Haptic Teleoperation System,"* Proceedings of IEEE WorldHaptics Conference, pp. 361-366, Tsukuba, Japan, 2007b

G.A.V. Christiansson, *"An Experimental Study of Haptic Feedback in a Teleoperated Assembly Task,"* Journal of Computing and Information Science in Engineering, Vol. 8, Issue 4, December 2008

R.W. Daniel and P.R. McAree, *"Fundamental limits of performance for force reflecting teleoperation,"* The International Journal of Robotics Research, Vol. 17, No. 8, pp. 811-830, 1998

Galbiati et al., *"Control and Operational Aspects of the Mascot 4 Force Feedback Manipulator of JET,"* Proceedings of the 14th IEEE/NPSS Symposium on Fusion Engineering, 1991

H.F. Gaydos, *"Sensitivity in the judgement of size by finger span. American Journal of Psychology,"* 71: 557–562, 1958.

B. Hannaford, *"A Design Framework for Teleoperators with Kinesthetic Feedback,"* IEEE Transactions on Robotics and Automation, Vol. 5, No. 4, Aug 1989a

B. Hannaford, *"Stability and performance tradeoffs in bilateral teleoperation,"* IEEE International Conference on Robotics and Automation, 1989b

F.C.T. van der Helm, A.C. Schouten, E. de Vlugt and G.G. Brouwn, *"Identification of intrinsic and reflexive components of human arm dynamics during postural control,"* Journal of Neuroscience Methods, Vol. 119, 2002

D. T. McRuer and H. R. Jex, *"A Review of Quasi-Linear Pilot Models, IEEE Trans. Human Factors in Eletronics,"* Vol. 3, pp. 231-208, 1967.

W. Mugge, D.A. Abbink and F.C.T. van der Helm, *"Reduced Power Method: how to evoke low-bandwidth behaviour while estimating full-bandwidth dynamics,"* IEEE 10th International Conference on Rehabilitation Robotics, pp. 575-581, 2007

R. Penrose, *"A generalized inverse for matrices,"* Proceedings of the Cambridge Philosophical Society , pp.406–413, 1955

G.A. Pratt and M.M. Williamson, *"Series elastic actuators,"* IEEE International Conference on IntelligentRobots and Systems, Vol. 1, pp. 399-406, 1995.

H.Z. Tan et al., "*Human factors for the design of force reflecting interfaces*," Dynamic Systems and Control, Vol. 55, 1994

E. de Vlugt, A.C. Schouten and F.C.T. van der Helm, *"Quantification of intrinsic and reflexive properties during multi-joint arm posture,"* Journal of Neuroscience Methods 155, pp. 328–349, 2006

D.M. Wolpert, Z. Ghahramani and M.I. Jordan, *"An internal model for sensorimotor integration,"* *S*cience, *Vol. 269. no. 5232, pp. 1880 – 1882, September 1995*

K.B. Shimoga, "*Finger force and touch feedback issues in dexterous telemanipulation,"* Proceedings of the Conference on Intelligent Robotic Systems for Space Exploration, 1992

G. Westling and R.S. Johansson, *"Responses in glabrous skin mechanoreceptors during precision grip in humans,"* Experimental Brain Research, 66(1):128–140, March 1987.