Master's thesis

The Effect of Online Adaptation on Conflicts in Haptic Shared Control for Free-Air Teleoperation Tasks

A.W. de Jonge

Delft, 28th of August 2012

Student No.: 1263757
Report No.: 1245

Department of BioMechanical Engineering
Faculty 3mE
Delft University of Technology

The Effect of Online Adaptation on Conflicts
in Haptic Shared Control for Free-Air Teleoperation Tasks

Arnold W. de Jonge

Committee:

Prof. dr. F.C.T van der Helm	Department of BioMechanical Engineering, 3mE
Dr. ir. D.A. Abbink	Department of BioMechanical Engineering, 3mE
Dr. ir. E.R. Boer	Entropy Control

Coaches:
Dr. ir. D. A. Abbink
ir. J.G.W. Wildenbeest
ir. H. Boessenkool

Preface

The combination of strengths of man and machine has always been of great interest for me. The idea of shared control is one of the best examples of this combination; the human versatile intelligent and judgmental abilities and the power of machines complement each other. Ideally a supporting system provides support matching the desired way of execution tasks of the human. The goal of this study is to find optimal support that adapts to the human operator to reduce conflicts between the human and support system.

The focus of my Master's thesis is presented in a research paper, which describes the human factors experiment. The appendices provide background on the study and allow all who is interested to gain more insight in challenges and results of this study.

In Appendix 1, a model for task performance conflicts is presented. Appendix 2 contains information on the experimental setup. In Appendix 3 and 4 the results of initial pilot studies are presented. Appendix 5 addresses the design and implementations of the propose adaptation. Appendix 6 presents some details on the human factors experiment. Appendix 7 presents an extensive overview of the results of the human factors experiment.

A USB-stick containing all raw measurement data, software, information, literature etc, has been submitted to the BioMechancial Engineering Department repository and is available on request.

Finally, I would like to thank all who contributed to this study, first of all my coaches David, Henri and Jeroen for their efforts in reviewing and the lively in-depth discussions with them. I would also like to thank all other members of the Delft Haptics Lab for their contributions.

The Effect of Online Adaptation on Conflicts in Haptic Shared Control for Free-Air Teleoperation Tasks

Arnold W. de Jonge[1], David A. Abbink[1], Frans C.T. van der Helm[1]
Jeroen G.W. Wildenbeest[1], Henri Boessenkool[1]

***Abstract*—Previous research on haptic shared control for telemanipulation has shown that artificial guiding forces can improve task performance and reduce control effort. However, guided trajectories are chosen beforehand, and conflicts between the human operator and support system are often reported in literature; momentary increases of force, reduced comfort and sometimes even deteriorated performance. Conflict solutions have been proposed for unexpected avoidance of obstacle on the supported trajectory, but solving conflicts due to trajectory negotiation mismatches have not been proposed. One way of minimizing these conflicts is to base support on individual preferred trajectories. Another way proposed in literature is by online adapting the supporting trajectory. The goal of this study is to provide evidence for the hypotheses that 1) an individualized supported trajectory yields less conflicts between operator and guidance than common general trajectory, and that 2) online adaptation of the supported trajectory will also yield reduced conflicts, regardless of the initially chosen support trajectory (individualized or general).**

In a human factors experiment, subjects (n=12) conducted a repetitive two degrees of freedom task in which they were provided with four different types of support. Both the recorded individual trajectories of operators and the common used centerline of the environment trajectory were provided with and without adaptation. The results show no effect of adaptation nor the support path on performance. Adaptation reduces support forces for both types of initial support paths, while the type of the path does not influence the support forces. The control activity of the non-adaptive support on the centerline of environment is higher than the adaptive support on the centerline of environment and the adaptive and non-adaptive support based on the recorded individual trajectory.

Those results provide evidence that recorded individual trajectory support reduces control effort for non-adaptive support, albeit only in control activity. The results furthermore provide evidence that adaptation of both types of initial support path reduce support forces and control effort.

In conclusion, online adaptation of haptic shared control reduces trajectory negotiation conflicts and the associated increased forces. It adapts to subject-specific preferences in trajectory, regardless of initially chosen supported trajectories.

***Index Terms*—Haptic Shared Control, Adaptation, Learning, Virtual Fixtures, Task Performance, Control Effort**

I. Introduction

Haptic shared control

The idea of haptic shared control is that, a human operator and an intelligent autonomous system share control of a system, by exerting forces on the control interface. This approach allows a balance between manual control which is prone to human errors and supervised automation that yield traditional human factor issues such as skill degradation, inattention, decreased responsibility, unawareness and misuse [1], [2]. The support system continuously provides supporting forces, allowing the operator to feel the support actions and the operator can always overrule the system. It has been shown that haptic shared control can improve task performance while reducing operator workload in many application, such as vehicle steering [3], [4], surgery [5], [6] and telemanipulation [7], [8]. Although in general haptic guidance is beneficial, studies also report that the human operator and support system can have conflicts on task execution and negotiation, which deteriorates task performance [3], [9].

Conflict resolving haptic shared control in literature

In literature there are generally two types of solutions proposed to reduces and solve conflicts. The first is to alter the level of support when conflicts arise. The second type provides different support in parallel and switching between them.

The conflict resolving by alteration of support level as proposed by Li [10] is based on the intensity and direction of the control force that toggled support on or off. This methods improved performance of a tracking task with obstacles on the support path compared to obstacle avoidance with fixed level support and no support. Marayong et al. [9] explored the effects of different levels of support in normal conditions and with obstacle avoidance. They suggest an optimal balance between a high level of support with good performance in normal conditions and a lower support level that allows good performance with unsupported obstacle avoidance. It is suggested that this balance is operator-specific and this value should therefore be set to individual preferences.

Passenberg [11] extended the previous work and proposed a solution for real-time adaptation of the level of support. Instead of choosing an optimal balance, the level of support is real-time adapted to the level of conflict, which is measured with interaction forces on the control interface.

In the field of automotive haptic support system, Tsoi [12] has proposed a lane-keeping support system, that is able to detect lane changes. The system evaluates system states (time to lane crossing [13]) and successfully deduces the human intention to change lanes and support those actions. Previous

[1] Department of BioMechanical Engineering, Faculty 3me, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands

work in automotive have shown successful lane support that deduces lane change intention by switching of turn indicators [14] and driver torque [15], without supporting the lane change.

Summing up the conflict resolving in literature, there are two general situations for which solutions have been presented. On one hand the systems that decrease their support when an unforeseen situation occurs. On the other hand, support systems that allow switching between various types of support. However, no research has been found that addresses solving of mismatches between the operator preferred trajectory and the supported trajectory as found in [12], [16], which can lead to counteracting support.

Adaptation for trajectory negotiation conflicts

Conventionally shared control is implemented as support on the centerline of the road [12], [16] or centerline of the environment [11]. For straight sections this support is likely to be similar to the human preferred task execution. However the human preferences in cutting corners do not match in the Centerline of Environmental support (CoE-support). Recording the Individual Manual Control Strategy (IMCS) of the human operator [17] and providing this as support might give a better form of support as it is closer to human preferences. Fig. 2 shows an exemplary CoE-support path and the recorded IMCS-support path. It is expected that it decreases conflicts, since the interaction forces are lower at the corners. It is expected that performance will remain similar, as it has been shown that the level of support does not have major influence on performance [11].

However by providing support, the task performance is increased, allowing faster task execution and higher velocities, which affect the preferred human trajectory [18], [19]. To provide support on the preferred trajectory it is proposed that the system adapts the support path online. This adaptation has to be gradual and slow to avoid adaptation to natural variation in human trajectory negotiation, which means that this kind of adaptation is suited to repetitive tasks only. High interaction forces can be contributed to conflicts [9], [11], [20] and will serve as the input for adaptation. It is expected that this decreases conflicts even more, since the support path is closer to the human preferred trajectory. It is expected that the support paths will adapt to the human preferred trajectory, regardless of the initial path.

The two main research questions of this study are:

- Can conflicts between the human operator and the support system be reduces by providing individualized support?
- Can adaptation of support reduce conflicts between the human operator and the support system?

The goal of this research is to find whether adaptive haptic shared control for repetitive tasks is beneficial, especially compared to providing support based on manual control trajectories. This is done with a human factors experiment in which a restricted free-space two degrees of freedom movement repetitive task is performed. Both the effects of the type of provided support (CoE and IMCS-support) and adaptation are studied. It is hypothesized that:

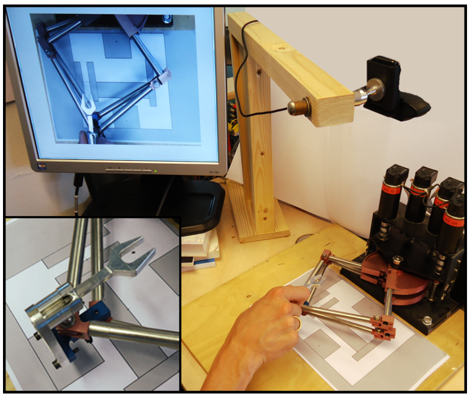

Fig. 1. Experimental apparatus on which the task is completed. The subjects hold the control interface (depicted in inset) and must remain in the white area as shown on the monitor.

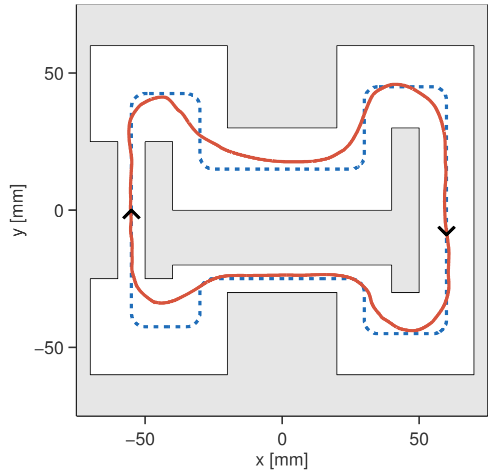

Fig. 2. Task environment for repetitive task in clockwise direction. Blue dotted line represents the centerline of environment support path. The red solid line represents the manual control trajectory of subject 6.

Hyp. A1 Individual manual control strategy (IMCS) support path yield less conflicts between the operator and support system compared to conventional Centerline of Environment (CoE) support.

Hyp. A2 Online adaptation of the support paths will also reduce conflicts, regardless of the initial provided support path.

II. Methods

Subjects

Twelve subjects performed the human factors experiment. All subjects are or have been students at the TU Delft, all right-handed and the mean age is 26.3 (±2.2) years. The subjects participated voluntarily and did not receive financial compensation for their participation.

Experimental setup

The apparatus used for the experiment consist of the parallel force-redundant (four actuators for three degrees of freedom) master device of the Munin-telemanipulator [21] as depicted in figure 1. The master device is controlled with MathWorks®

xPC Target™ real-time operating system running at 1 kHz. The master device is equipped with position sensing using motor encoders. The position accuracy is 0.03 mm, the maximal force in 18 N in y-direction and 4.5 N in x-direction due to the non-isotropic workspace [21]. The stiffness of the device is 1.6 N/mm in x and y-direction. The orientation of the device is fixed with a 0.25 Nm/rad stiffness controller. The mass of the master device is approximately 130 g [22].

A web-camera with a resolution of 640x480 provided visual feedback to the operator on a 17 inch monitor with the top view of the environment.

Task description

The subjects were instructed to move trough the environment as depicted in Fig. 2 in clockwise direction in 20 consecutive repetitions. They were given no explicit instruction on speed or accuracy, but the subjects are given the penalty of restarting the trial when making more than two errors. This resembles typical telemanipulation tasks in which speed and accuracy are less relevant, but damage in the remote environment must be avoided.

The subjects are instructed as if they have to move a nuclear rod through a nuclear power plant. The rod is represented by a single black dot as a top view as shown in Fig. 1. Hitting the wall more than two times will break the imaginary nuclear rod and the trial has to be restarted. A small vibration on the device will inform the subjects when they hit the walls (sinusoid force with amplitude of 0.4 N and frequency of 32 Hz).

The environment (Fig. 2) is designed for a repetitive task, since the adaptation of the support is based on repetitive adaptation. The environment is designed with sharp corner-edges and wider and smaller sections. In initial experiment on the same setup with different types of corners it appeared that sharp corner edges and alterations between narrow and wide areas yielded different trajectories for operators with similar performance.

Haptic shared control design

The haptic shared control supporting system consists of passive path attractive support; the system will pull the master orthogonal to the support path, without forcing any motion in line with the support path. The guiding force is not based on the position prediction, which have shown to be beneficial over instantaneous support in a 1DoF steering task [23]. The predicted future position is estimated as the estimated position in 0.1 s with given the current velocity vector, adapted from [24] and [16]. The guiding force is based on the error between the look ahead-position and the closest point on the path p_{path} multiplied by a shared control stiffness gain.

$$\hat{e}_{lookahead} = d(x_{current} + 0.1 * \dot{x}, p_{path}) \quad (1)$$

$$F_{sc} = k_{sc} * \hat{e}_{lookahead} \quad (2)$$

The stiffness gain k_{sc} is set to 100 N/m based on previous experiments with shared control on this particular telemanipulator [24], resulting in maximal support forces between 0.5 N and 1.6 N for small respectively wide areas.

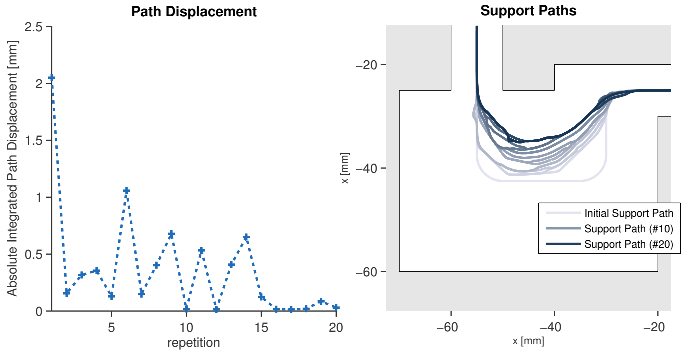

Fig. 3. The adaptation of a subsection of the support path in a training session of subject 1. The right picture shows the support paths, the initial provided shared control path is the centerline of environment path. The left picture shows the distance of displacement of the support paths over the repetitions.

Adaptive shared control design and implementation

Interaction forces on the control interface are a common measure for conflicts [9]–[11]. The support force is used as an approximation of the human interaction force, since there is no force sensor on the control interface. Adaptation consists of updating the individual points of the path. Every point on the path is updated only when the error distance $\hat{e}_{lookahead}$ between the look-ahead position x_{la} and that particular point p overcomes a certain threshold e_{th}. The updated path point p_n is calculated with an recursive exponential moving average filter on the previous path position p_{n-1}, with learning factor γ:

$$p_n = \gamma x_{la} + (1-\gamma)p_{n-1} \quad \forall\, d(x_{la}, p_{n-1}) \geq e_{th} \quad (3)$$

The weight w_n of every n^{th}-to-last input on the average is:

$$w_n = (1-\gamma)\gamma^n \quad (4)$$

The number of repetitions was set to 20, to avoid fatigue of subjects. Since the last 5 repetitions are used for measurements, the support system has to be adapted before the 15^{th} repetition. The effect of the initial path is marginalized before the 15^{th} by setting the learning factor γ to 0.2 and the contribution of the initial path is below 1% after 14 repetitions.

Adaptation is calculated real-time, but the updated path is provided after execution of each trial to avoid an unreliable system that might feel as dodging or drifting.

To gain basic understanding in the effects of providing shared control on the trajectories initial experiments have been performed. Two trained operators (of which one participated in this experiment) performed a repetitive task with shared control. The standard deviation of the performed trajectory is 1.25 mm, meaning that 95% of the repetitions were within ±2.45 mm of the average. Therefore 2.5 mm is set as threshold for adaptation e_{th}.

The adaptation is limited so that it cannot adapt the path closer than 2.5 mm to the wall. This basic form of intelligence avoids supporting the operator in performing dangerous trajectories.

In Fig. 3 a typical example of adaptation of the support path is shown. The figure shows the adaptation process in the bottom-left corner, when the subject is provided with the CoE

TABLE I
EXPERIMENTAL CONDITIONS

Adaptive (F1)	Path shape (F2)	
	Centerline of Environment	Individual Manual Control Strategy
Non-Adaptive	I (NA - CoE)	II (NA - IMCS)
Adaptive	III (A - CoE)	IV (A - IMCS)

support path. The right picture shows the actual support paths and the left picture shows the amount of adaptation. In this example the adaptation after repetition 15 is very limited.

Experimental Conditions

The two controlled experimental factors were adaptation (adapting/ not-adapting) (F1) and the support path that is provided (F2) and therefore four conditions were tested as shown in TABLE I.

First the subject was given a written instruction on the how the task should be performed. After that the subject was trained in performing the task without shared control in three trials, each consisting of ten consecutive task repetitions. The subject was allowed to make more than two errors in each repetitions and was given feedback on the number of errors.

After the training the subjects were instructed to perform a validation trial, in which they had to show they are capable of performing the task (make no more then two errors while performing 20 repetitions). The average performed trajectory in this manual control validation trial was recorded without their knowledge and used as their individualized manual control strategy path. Task repetitions with an error were excluded for the support path. They had to repeat the validation until they succeeded. After this, subjects were presented with the subjective measurement questionnaires.

The actual experiment consisted of 16 trials, four trials per conditions, 20 repetitions per trial. Only the second, third and fourth trial of each condition were used for analysis, the first one allows training. After each condition the subjective measurement questionnaires were presented to evaluate the condition. Previously filled in forms were also handed out, allowing subjects to compare the conditions.

Measured variables and metrics

Data is logged at 1 kHz. Performance is measured with the following metrics:

- **Task Completion Time [s]** Time required for the subject to perform one sequential round.
- **Number of Errors**: The number of times the wall has been hit during one repetitions (20 rounds).

Conflicts and control effort are measured with the following metrics:

- **Shared control force [N]** The average force applied by the support system.
- **Steering Corrections [deg/s]** The integrated angular corrections over the entire trajectory, as a measure of the control activity, measured with the absolute change of direction of motion on every point on the trajectory, filtered with a 2^{nd}-order ButterWorth low-pass filter at 10 Hz.

The metrics (except the number of errors) are measured over the five last repetitions of each of the three trials per condition and a 100 Hz low-pass filter is applied.

Furthermore, subjective measurements were recorded for each of the four conditions:

- **NASA-TLX** NASA-Task Load Index for perceived workload [25]. Rating six different sub-scales, gives a subjective workload on a scale from 0 to 100.
- **Questionnaire** Subjective rating to gain understanding in how useful and appreciated the four different conditions are perceived. Subjects were asked to answer questions. The first question is 'How difficult was the task for you to perform?', with rating from 1 ('very easy') to 9 ('very hard'). Another question asked was 'do you like the kind of support you were provided with?', rating 1 ('not at all') to 9 ('definitely'). The third question was 'do you have the feeling the system is helping or counteracting you', rating 1 ('counteracting') to 5 ('neither') to 9 ('helping').

TABLE II
ANOVA RESULTS FOR THE FIVE METRICS . [1]

Metric	Factor: Adaptive (F1)	Path (F2)	Subject (F3)
Task Completion Time	-	-	•
Number of Errors	-	-	•
Shared Control Force	•••	-	-
Steering Corrections	•••	•	-
Subjective Workload	-	-	•

[1] •••,••, • respectively denote significance values $p \leq 0.001$,$p \leq 0.01$,≤ 0.05.

Data analysis

The effects of the two experimental factors are compared with an three-way ANOVA, accounting for (interaction of) the two experiment factors: adaptation (F1), support path (F2) and inter-subject variation factor (F3). p-values equal or below α=0.05 are considered to be significant.

Next to the experimental factors effects, the four individual conditions are compared with a Tukey's least significant difference post-hoc test if one of the two factors is significant. The critical values are not adjusted because of the low number of comparisons. Questionnaire results are also presented.

III. RESULTS

The experimental results are compared using a three way ANOVA, with three factors: adaptation (F1), support path (F2) and inter-subject variation (F3) as presented in TABLE II.

The first two rows of the table show that the task performance (task completion time and number of errors) is not affected by adaptation nor the type of support path and is only subject-dependent. Both the shared control force and steering corrections are affected by adaptation, and steering corrections is also affected by the type of support path. Finally,

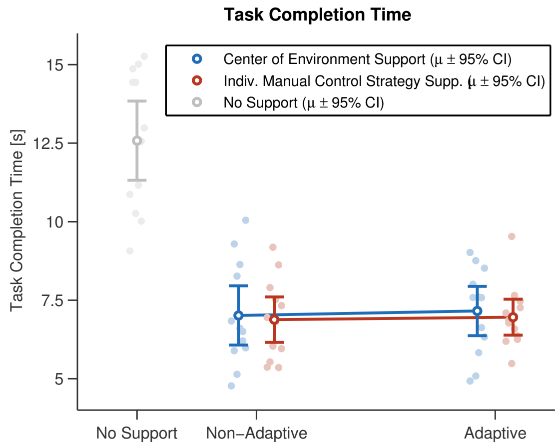

Fig. 4. Time to complete, mean (μ) and 95% confidence interval for 12 subjects, dots represent individual data points per subject. The task completion time is not affected by the adaptation nor the type of support. The task completion time of the no-support is shown for reference.

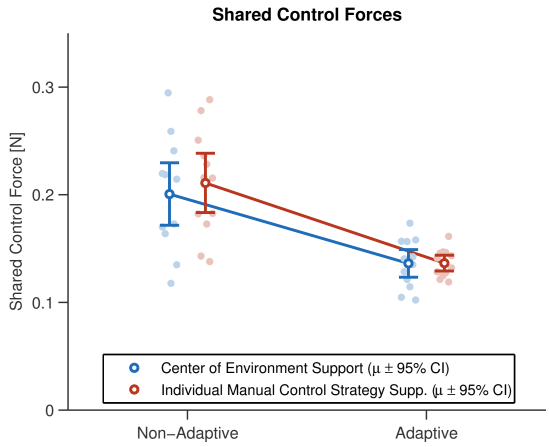

Fig. 5. The Shared Control Forces of the Support System, mean (μ) and 95% confidence interval, dots represent individual data points of each subject. The Shared Control Forces is affected by the adaptation and the shared control forces for the adaptive conditions are significantly lower.

the subjective workload is only subject-dependent and is not affected by the adaption nor the support path.

The metrics are described in detail in the following sections, followed by the performed trajectories and support paths of a typical subject.

Effects of adaptation and path on task performance

Fig. 4 shows the task completion time of the four conditions and the manual task control completion time for reference. The completion time is similar for all conditions and only a difference can be seen between the support and manual control times. It is not affected by adaptation (p=0.668, F=0.194) or support path (p=0.583, F=0.320) and is only subject dependent (p=0.025, F=4.287).

The average number of errors were 0.33 (σ=0.45) for cond. I, 0.28 (σ=0.42) for cond. II, 0.31 (σ=0.46) for cond. III and 0.25 (σ=0.32) for cond. IV. The number of errors is not affected by the factor adaptation (p=0.658, F=0.208) and the factor support path (p=0.594, F=0.301) and are only subject dependent (p=0.018, F=3.533).

The results show no significant differences in task performance, measured in both task completion time and the number of errors. The completion times with support appear to be lower than without support.

Effect of adaptation and path on shared control forces

The shared control forces are shown in Fig. 5. There is a clear difference between the adaptive and non-adaptive conditions. The shared control forces are affected by the factor adaptation ($p<0.001$, F=55.846) and not affected by the support path (p=0.669, F=0.194) nor is it subject-dependent (p=0.464, F=1.047).

Post-hoc analysis show that adapting CoE-support shared control force (μ=0.136, σ=0.023) is 32% lower than that of the non-adapting CoE-support (μ=0.201, σ=0.051) and 35% lower than the non-adapting IMCS-support (μ=0.211, σ=0.049). The adapting IMCS-support (μ=0.137, σ=0.013) is 32% lower than the non-adapting CoE-support and 35% lower than the non-adapting IMCS-support.

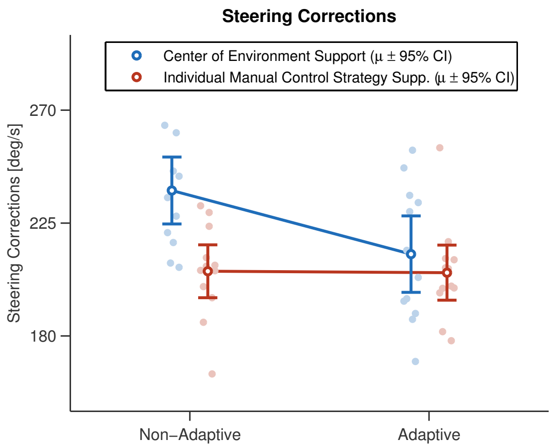

Fig. 6. Steering corrections, mean(μ) and 95% confidence interval, dots represent individual data points per subject. The steering corrections are affected by the adaptation and the path and the non-adaptive CoE-support steering corrections is significantly higher than the steering corrections in the other conditions.

Effects of adaptation and path on steering corrections

Fig. 6 shows steering corrections for the four conditions. The steering corrections are effect by the adaptation affects(p=0.008, F=10.401) and the support path (p=0.012, F=9.162).

Post hoc analysis showed that the steering corrections of the non-adapting CoE-support is higher than the other conditions. Compared to the steering corrections for non-adapting CoE-support (μ=238, σ=24), the non-adapting IMCS-support (μ=205, σ=19) is 14% lower, the adapting CoE-support (μ=213, σ=27) is 11% lower and the adapting IMCS-support (μ=205, σ=19) is 14% lower.

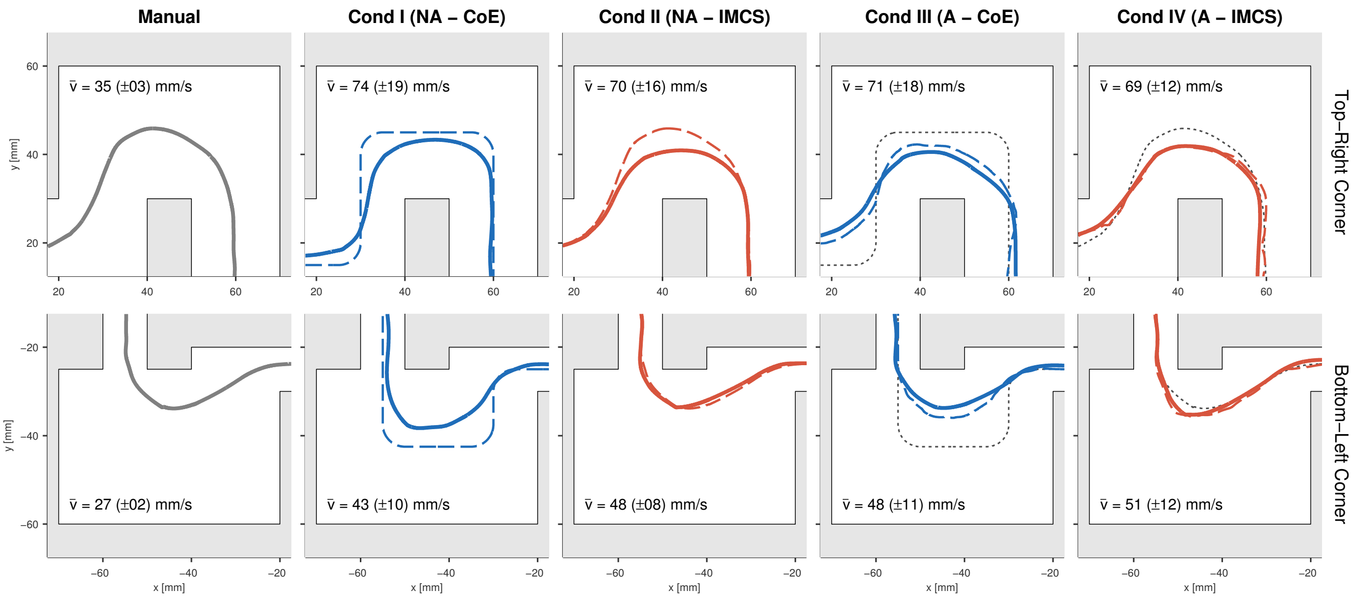

Fig. 7. The trajectories and support paths of subject 1 for each condition, displayed as the average of last five repetitions of second trail. Both the top-right corner (upper row) and bottom-left corner (lower row) are shown. The left most picture shows the average manual control trajectory, that is provided as support path in Cond. II and serves as initial support path for Cond. IV. The solid colored line represents the performed trajectory, the colored dashed line the adapted support path and for Cond. III and Cond. IV the grey dotted line the initial support path. The average velocities in the corners is also shown.

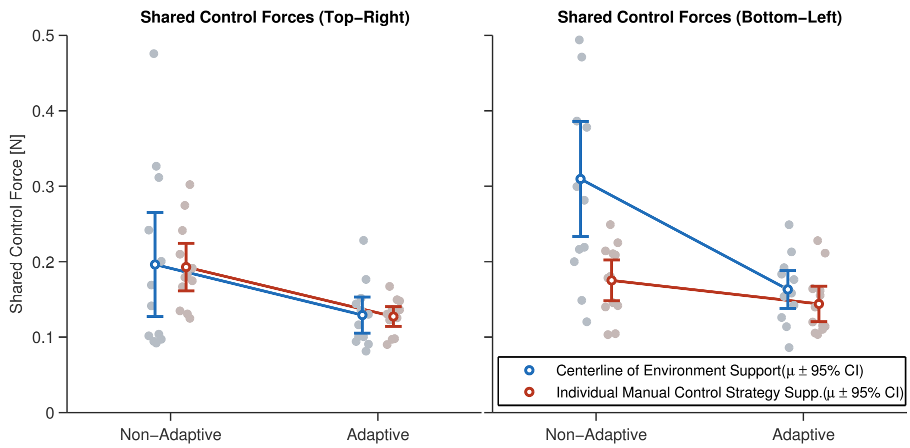

Fig. 8. The Shared Control Forces of the Support System, mean (μ) and the 95% confidence intervals, dots represent individual data points. The non-adaptive support shows higher forces in the fast (top-right) corner in the left picture. The right picture shows higher forces for the non-adaptive CoE-support compared to the other conditions.

Effects of adaption and guidance path on trajectories

In Fig. 7 the performed trajectory and the support path of the typical subject (1) is shown in the top-right corner with high velocities and the bottom-left corner with low velocities. The left-most picture shows the trajectories performed without support, which are presented as support paths in condition II and IV. The picture of condition I shows the difference between the support path (dotted line) and the actual performed trajectory.

The implementation of the support system is such that the difference between the actual position and the support path is multiplied with a stiffness, the difference between those is therefore directly related to the support system force. Distance between the support path and the trajectory is for the non-adaptive conditions higher than for the adaptive conditions in the fast corner (top-right) corner and therefore the shared control forces are also higher as shown in Fig. 8.

In the slow bottom-left corner (Fig. 7) the difference between the support path and the performed trajectory seems larger in the non-adaptive CoE-support, which is reflected in the shared control forces in that corner (Fig. 8).

Effects of adaptation and guidance path on subjective workload and other subjective measures

The subjective workload measured with the NASA-TLX was not affected by adaptation (p=0.186, F=1.993) nor the support path (p=0.551, F=8.589). However the subjective workload is affected by the subject (p=0.045, F=8.589).

The average reported difficulty was 3.6 (1.4) and there is no significant difference between any of the conditions. The non-adaptive conditions were rated more counter-acting than the adaptive conditions (F=14.004, p=0.003). 11 out of 12 subjects rated the likability of the NA IMCS-support higher than NA CoE-support, 1 subject lower. 7 subjects rated condition A IMCS-support higher than A CoE-support, 4 lower.

IV. Discussion

The results did not show that task performance benefits from real-time adaptation nor providing Individual Manual Controls Strategy (IMCS) support compared to Centerline of Environment (CoE) support. The support force are decreased due to adaptation and for non-adaptive support the IMCS-support showed less steering corrections. Furthermore the NASA-TLX showed no difference in subjective workload for any condition.

Individualized manual control strategy-support

The performed trajectory and support path in the fast (top-right) corner (Fig. 7) and the shared control forces in the fast corner (Fig. 8) show no differences for both non-adaptive CoE and IMCS-support conditions. This effect on shared control forces is seen in the entire task (Fig. 5), which suggests that providing support based on IMCS is not an improvement over the more commonly used CoE-support.

However the slow (bottom-left) corner shows different results. The difference between the support path and the trajectories appears to be much larger for non-adaptive CoE then for non-adaptive IMCS support. It seems as if the non-adaptive IMCS shared control support forces are lower than the non-adaptive CoE. The IMCS appears to be appropriate support for this low-velocity corner.

The difference in velocity may be one of the sources in differences between the slow and fast corner. Since the environment in the bottom-left section is more constraining than the top-right section, the velocities are lower due to the two small straight sections. Providing support seems to yield less increase in velocity in the bottom-left corner. Since the trajectories are strongly correlated to the velocity [19], [26], the trajectories of the bottom-left section are not likely to be strongly affected by the support, while in the top-right section the absolute velocity increases and therefore also the preferred trajectory.

The results show that the steering corrections are lower with IMCS for non-adaptive support. The IMCS seems to provide sufficient improvements to reduce steering corrections.

Adaptive support

The effect of adaptation on the support paths are shown in Fig. 7. The trajectories and the support paths appear to be close to each other for both conditions, which is reflected in the lower shared control forces as seen in Fig. 8. Most notably the two types of initial support path both seems to have little influence on the final trajectory, this implies a form of adaptation which is robust to sub-optimal initial support paths.

The adaptation design has showed to be able to reduce the shared control forces and the gradual adaptation was appropriate for this particular task.

Adaptation and individualized manual control support

The adaptation has shown to reduce the shared control forces which are related to the conflicts [9], [11] and the IMCS-support has shown to decrease steering corrections. The effects of adaptation on shared control forces are clearly present. Considering these shared control forces, adaptation seems a more promising solution to decrease conflicts.

However the proposed adaptation requires a repetitive task and requires time and effort to allow adaptation. For non-repetitive tasks the IMCS-support can reduce the steering corrections and results suggest that it might decrease shared control forces, and therefore conflicts, for low velocities.

Performance

No differences in performance were found in this experiment. The instruction on task conduction was relatively free, the subjects could determine at which velocity the task was executed and how much errors they were willing to make (maximal two). Some subjects did not make any errors during the entire experiment others accepted one or two. This explains the subject-dependency of the performance. Research has suggested that the performance was not influenced by the level of support [11], which agrees with these finding.

Effect of providing shared control

For the entire task the task completion time is around 40% lower with support than without support, which correspond to previous findings in literature [3], [8]. The average velocity of the entire task is therefore also higher, which can explain why the effects as seen in the top-right corner are dominant.

Observations of individual subjects

Out of the twelve subjects, only one liked the CoE-support more than the IMCS-support of the non-adaptive support. This subject performed the experiment with the adaptive and non-adaptive CoE-support as the first two conditions. The subject appeared to have adapted to this kind of support, since both the adapted support paths are shaped similar to the CoE-support path.

Subjective measurements

Subjects did not report differences in subjective workload between the four conditions, while the shared control forces were lower with adaptation. The subjective workload was measured at the end of each condition, effectively measuring all task repetitions of that condition. Unadapted task repetitions might have contributed to the workload rating of subjects. However the questionnaire, effectively measuring over the same as the NASA-TLX, showed some significant difference. The adaptive support was perceived as less counter-acting and more helping. However the effect of on the total perceived workload appeared to be marginal.

Limitations and Recommendations

The experiment is performed in a simplified telemanipulation environment and the coupling of master and slave device will greatly affect the task. Coupled system dynamics will be introduced and it can be expected that this will influence adaptation, the shared control forces will be higher and it is more difficult to discriminate the intentional forces and preferred trajectory. This also advocates using more sophisticated forms of adaptation that account for mechanical system dynamics.

Furthermore, the results from this two degrees of freedom task 2D task cannot easily be generalized to more complex 3D six degrees of freedom tasks in teleoperation. The task was regarded medium easy (subjects rated 3.6 on scale 1-9), but six degrees of freedom tasks are more difficult, mainly due to the limitation of visual feedback in 3D. Visual depth perception for especially 2D displays is limited. Unless operators are extensively trained in performing tasks without shared control support it is likely that they will rely on the information through the haptic channel and are more reluctant to adapt to the provided shared control support. For more complex tasks, the presented adaptation might only be beneficial for experienced and extensively trained operators.

The focus of this experiment was towards small conflicts in task conduction with gradual adaptation over repetitions. The proposed adaptation has shown to be beneficial, but only for those small conflicts in repetitive tasks. The proposed adaptation is not intended as alternative solution for adaptive support systems for unexpected obstacle avoidance situations [11]. Moreover it is believed that the two types of adaptation can coexist in one support system, incorporating task conduction adaptation and unexpected avoidance situation adaptation, responding to different conflict cues such as different force levels or other observed signals.

V. Conclusion

The effect of trajectory negotiation conflict solving shared control support on task performance, shared control forces and control effort is assessed in twofold. First, the effects of providing individualized manual control strategy (IMCS) support compared to the more common used centerline of the environment (CoE) support. Secondly, the effects of adaptation of the shared control support path was studied for both IMCS and CoE support. The former was evaluated by providing support based on recorded individual task trajectories. The second was evaluated by adaptation of the support path based on the level of shared control forces.

The human factors experiment consists four condition of non-adaptive and adaptive IMCS-support and non-adaptive and adaptive CoE-support. Comparing the results of those conditions, the following conclusions can be drawn:

- Non-adaptive IMCS support reduces steering corrections as compared co non-adaptive CoE-support.
- Adaptation of both IMCS and CoE-support reduces shared control forces
- Control effort of adaptive IMCS and CoE-support is similar
- No difference in performance between the forms of haptic shared control were found

It can be concluded that adaptation based on support force levels seems a promising solution for trajectory negotiation conflicts between the human operator and the support system, since shared control forces are reduced.

References

[1] L. Bainbridge, “Ironies of automation,” *Automatica*, vol. 19, no. 6, pp. 775–779, Nov. 1983.

[2] R. Parasuraman and V. Riley, “Humans and Automation: Use, Misuse, Disuse, Abuse,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 39, no. 2, pp. 230–253, Jun. 1997.

[3] P. G. Griffiths and R. B. Gillespie, “Sharing control between humans and automation using haptic interface: primary and secondary task performance benefits.” *Human factors*, vol. 47, no. 3, pp. 574–90, Jan. 2005.

[4] D. Abbink and M. Mulder, “Exploring the dimensions of haptic feedback support in manual control,” *Journal of Computing and Information Science In Engineering*, vol. 9, no. 1, p. 011006, 2009.

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

[6] D. Kragic, “Human-Machine Collaborative Systems for Microsurgical Applications,” *The International Journal of Robotics Research*, vol. 24, no. 9, pp. 731–741, Sep. 2005.

[7] D. Aarno, S. Ekvall, and D. Kragic, “Adaptive virtual fixtures for machine-assisted teleoperation tasks,” in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, no. April. IEEE, 2005, pp. 1139–1144.

[8] H. Boessenkool, D. A. Abbink, C. J. M. Heemskerk, and F. C. T. Van der Helm, “Haptic Shared Control Improves Tele-Operated Task Performance towards Performance in Direct Control,” in *World Haptics Conference*, 2011, pp. 433–438.

[9] P. Marayong and A. M. Okamura, “Speed-Accuracy Characteristics of Human-Machine Cooperative Manipulation Using Virtual Fixtures With Variable Admittance,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 46, no. 3, pp. 518–532, 2004.

[10] M. Li and a.M. Okamura, “Recognition of operator motions for real-time assistance using virtual fixtures,” in *11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2003. HAPTICS 2003. Proceedings.* IEEE Comput. Soc, 2003, pp. 125–131.

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

[12] K. K. Tsoi, M. Mulder, and D. Abbink, “Balancing Safety and Support : Changing Lanes with a Haptic Lane-keeping Support System,” in *Proceedings of the IEEE SMC Conference 2010, Istanbul*, 2010.

[13] N. M. Enache, Y. Sebsadji, S. Mammar, B. Lusetti, and S. Glaser, “Driver’s influence on the performance of an integrated lane departure avoidance and lane keeping assistance system,” in *2009 IEEE International Conference on Control Applications*. IEEE, Jul. 2009, pp. 119–124.

[14] J.-F. Liu, J.-H. Wu, and Y.-F. Su, “Development of an Interactive Lane Keeping Control System for Vehicle,” in *2007 IEEE Vehicle Power and Propulsion Conference*. IEEE, Sep. 2007, pp. 702–706.

[15] V. Cerone, M. Milanese, and D. Regruto, “Combined Automatic Lane-Keeping and Driver’s Steering Through a 2-DOF Control Strategy,” *IEEE Transactions on Control Systems Technology*, vol. 17, no. 1, pp. 135–142, Jan. 2009.

[16] M. Mulder, D. A. Abbink, and E. R. Boer, “The effect of haptic guidance on curve negotiation behavior of young, experienced drivers,” in *2008 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, Oct. 2008, pp. 804–809.

[17] S. Ekvall, D. Aarno, and D. Kragic, “Online task recognition and real-time adaptive assistance for computer-aided machine control,” *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 1029–1033, Oct. 2006.

[18] S. Vieilledent, Y. Kerlirzin, S. Dalbera, and A. Berthoz, "Relationship between velocity and curvature of a human locomotor trajectory," *Neuroscience Letters*, vol. 305, no. 1, pp. 65–69, Jun. 2001.

[19] P. Viviani, "Minimum-jerk, two-thirds power law, and isochrony: converging approaches to movement planning." *Journal of Experimental Psychology: Human . . .* , vol. 21, no. 1, pp. 32–53, 1995.

[20] N. Stefanov, A. Peer, and M. Buss, "Role determination in human-human interaction," in *World Haptics 2009 - Third Joint EuroHaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*. Ieee, 2009, pp. 51–56.

[21] G. Christiansson, "Hard master, soft slave haptic teleoperation," Ph.D. dissertation, 2007.

[22] J. G. W. Wildenbeest, "Thesis - Improving the Quality of Haptic Feedback Yields Only Marginal Improvements in Teleoperated Task Performance," 2010.

[23] B. A. C. Forsyth and K. E. MacLean, "Predictive haptic guidance: intelligent user assistance for the control of dynamic tasks." *IEEE transactions on visualization and computer graphics*, vol. 12, no. 1, pp. 103–13, Jan. 2006.

[24] H. Boessenkool, D. A. Abbink, C. J. M. Heemskerk, and F. C. T. Van der Helm, "Haptic shared control improves tele-operated task performance towards performance in direct control," *IEEE Transactions on Haptics*, pp. 1–12.

[25] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research," *Human mental workload*, vol. 1, no. 11, pp. 139–183, 1988.

[26] P. N. Sabes and M. I. Jordan, "Obstacle avoidance and a perturbation sensitivity model for motor planning." *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 17, no. 18, pp. 7119–28, Sep. 1997.

Appendices belonging to the Master's thesis:

The Effect of Online Adaptation on Conflict in Haptic Shared Control for Free-Air Teleoperation Tasks

A.W. de Jonge

Contents

Appendix 1: Task performance conflict model

In this chapter an effort is made to describe types of conflicts that occur between the human operator and the shared control support system. A model is therefore presented that describes conflicts on three levels. The model is explained with an exemplary task and the implications and limitations of the model are presented.

Providing shared control can improve task performance (Boessenkool, 2010) and decrease workload, but in case of conflict performance can deteriorate (Griffits, 2005; Marayong; 2004). Conflicts may arise from different sources and in literature the focus is primarily on conflict solving for unforeseen obstacles on the trajectory (Marayong, 2004; Passenberg, 2011). This study however focuses on conflicts on trajectory conduction and negotiation, which are different conflicts.

Now the question is whether there are more types of conflicts and what the sources are for those conflicts. In this section a model is proposed that can help to identify and classify task performance conflicts. First the model is briefly explained with a telemanipulator task and after that the implication and limitation of the model are presented.

Human operator-support system interaction model

The model as shown in Fig. 1-1 is inspired by the widely accepted task performance model of Rasmussen (Rasmussen, 1983) and the neuromuscular model of shared control interaction (Abbink, 2010). Rasmussen describes task performance on knowledge, rule and skill level for human operators. The presented new model applies this distinction between levels of performance also within the shared control support system. The skill-level task performance is extended with the neuromuscular interaction model.

The human and shared control agent (support system) interact on skill-level by exchanging forces on the control interface (passive physical interface-block; *green*). The control interface characteristics determine the actual control input to the system. Both the human operator and the support system get feedback from the environment to correct the error between the desired system states and the perceived system states. The skill-level controller block translates the state error to the desired input. The desired input is translated into a force (depending on the type of control interface) and the error between the desired input and the actual input state can be suppressed by neuromuscular contraction ($H_{nms\ adaptive}$).

The rule-level performance generates those desired system states. Situations in the environment are recognized and associated with stored rules on how to perform the task. These will affect the desired system state, but can also influence the skill-level task performance. The human can adapt the arm stiffness (A-block at $H_{nms\ adaptive}$) or the feed forward gains used to control the system (F-block at H_{ppi}), which can be implemented in the shared control support system as well.

The knowledge performance for humans determines the planning of the task and handles unforeseen situations, in which possible problems must be identified and decisions about the task plan must be made. Currently implementation of shared control is such that predefined assumptions determine the planning of the task, but in case of unforeseen events the shared control does not know how to handle those situations.

In the following section the conflicts that arise from the differences in the task performance levels are presented.

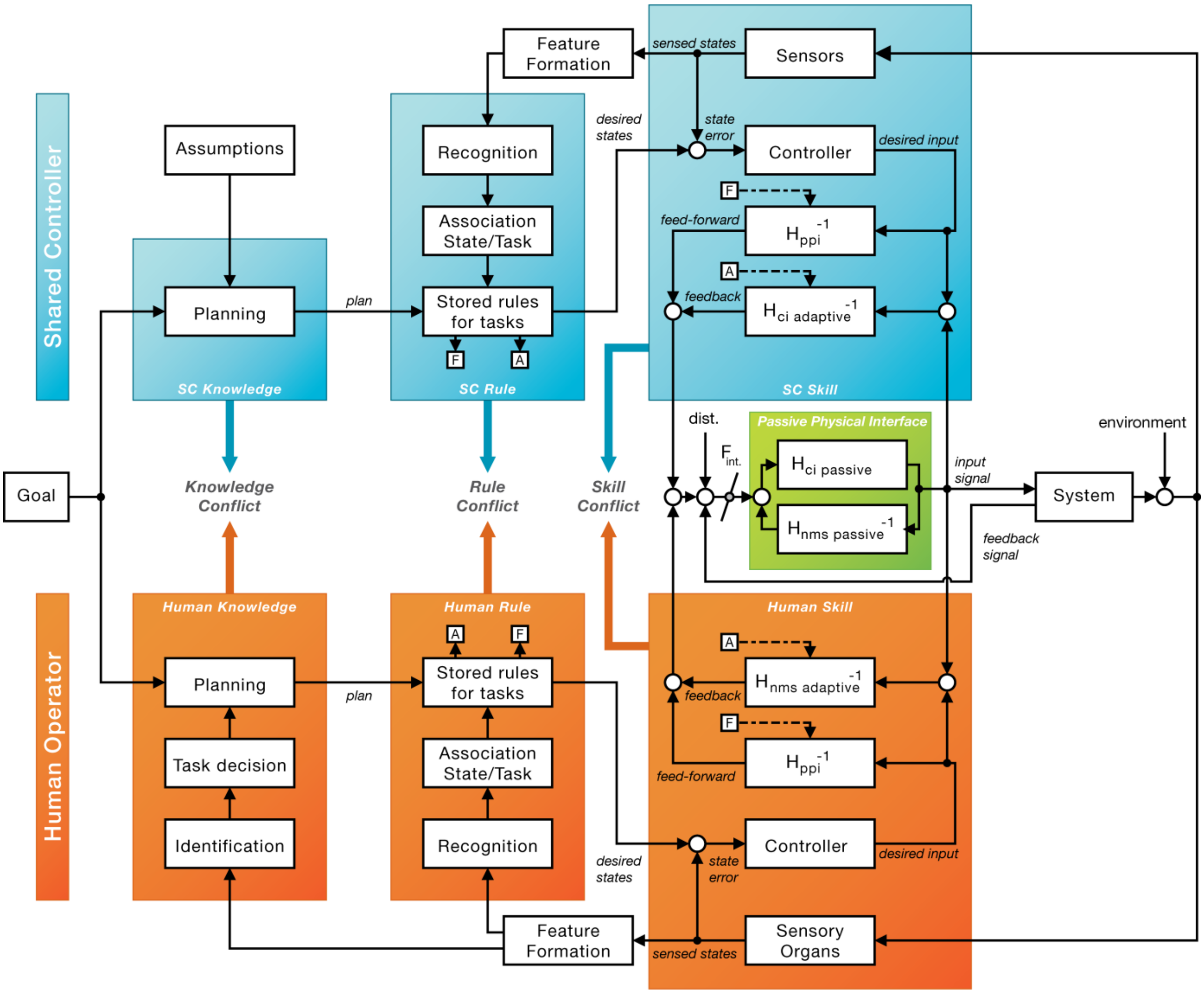

Fig. 1-1: Schematic symmetric simplified model of a human operator interacting with a shared control agent to classify conflicts. The model is based on task performance (Rasmussen, 1983) and neuromuscular models (Abbink, 2010). The performance and conflicts of the attentive skilled operator (red) and of the shared control agent (blue) are presented on three levels: knowledge, rule and skill. On skill-level the operator and agent interact with the passive physical interface (green).

Conflicts

The model is explained in detail with the aid of an example teleoperation task. The goal of this task is to unscrew and remove a bolt using a teleoperation system. Each of the three levels of performance and interaction for the human operator (human) and shared control agent (SC) is presented to find the sources of conflicts.

Knowledge-based performance and conflicts

The operator creates a planning based on his knowledge and might decide to alter this planning while performing the task, based on identified unforeseen events. *(Human Knowledge-block)* The agent has a fixed planning based on assumptions made before performing the task and cannot handle unforeseen and unfamiliar situations. *(SC Knowledge-block)* This is the source for knowledge-based conflicts, the operator might be aware of events that the agent cannot perceive.

Example of knowledge-based conflicts

The spanner breaks while unscrewing the bolt. The operator can improvise and decide to fix the spanner or use another tool. Since the assumption was made that the spanner remains intact, the agent is not aware of this rare event. Only in the best scenario, it perceives that the operator is well aware of this event and will not counteract the operator. In some situations is it also possible that the operator incorrectly identifies an unforeseen situation and decides to change the planning, while the agent is correct.

Rule performance and conflicts

The plan of performing the task consists of several known and familiar subtasks and their associated `standard solution'. The operator recognizes a typical situation and associates it with standard rules to obtain desired system states. *(Human rule-block)* The same applies to the intelligent agent that is able to recognize and handle `standard' situations. *(SC rule-block)* Conflicts on rule level can be caused by three sources: 1) Situations are recognized inappropriately or not at all. 2) Situations are associated differently. 3) The rules of the operator and agent differ.

Example of rule conflicts

An obstacle is blocking the access to the bolt. First of all, a conflict occurs when the blocking obstacle is not properly perceived and recognized by operator or agent. Secondly, association of solutions (rules) to this particular situation can differ. Passing the obstacle on the left or right are viable solutions and the choice for one of those can be arbitrary. While this might lead to a conflict, it is possible that neither of the solutions is a better solution. Thirdly, while the operator and agent agree on passing the obstacle on the left, they might differ in how large the distance to the obstacle should be.

Skill performance and conflicts

On the skill level the operator and agent perform similarly. *(Human and SC skill-blocks)* The state error, the differences between the actual and desired system states, serves as input for the controller. The controller has an internal model of the system and uses this to create the desired system input. Actual input forces to achieve this input are generated based on a feed-forward model of the physical interface. Feedback forces are generated from the co-contraction around the input signal by adapting the neuromuscular properties of the operator or by altering the control interface properties by the agent. Conflicts on skill level can exist on four levels: 1) The operator and agent perceive the states of the system and environment differently. 2) The operator and agent have a different controller with an internal model of the system, yielding different desired inputs. 3) The conversion from desired input signal to feed-forward forces is different, yielding conflict forces. 4) The feedback forces (by co-contraction or additional agent interface stiffness) differ.

Example of skill conflicts

The spanner has to grip the bolt. First of all, due to lacking depth perception of the operator or invalid sensor measurements of the agent the sensed position of the spanner might conflict. Secondly, the controller with the internal model of the teleoperation spanner system of the operator and agent are likely to be different, and therefore the exact input the operator wants to generate is likely to be different from the agent. Thirdly, even when the desired input is exactly similar, converting this to input forces can be different. Fourthly, the levels on co-contraction or additional control interface stiffness can be different and result in conflicting levels of co-contraction.

Implications and limitation of the proposed model

The presented model demonstrates methods to differentiate between conflicts. Nevertheless, the model needs to be more refined, since some possible conflicts are not included. This model assumes that the levels of task performance for both the human operator and support system are equal and conflict only arises on the same level. However, as stated by Rasmussen, the distinction between the levels of task performance cannot be strictly defined. Therefore task performance on the human knowledge-level might actually correspond to the rule-level performance of the shared controller. When conflicts arise, these conflicts cannot be attributed to either knowledge-level conflict or rule-level conflicts. It is suggested that this model is extended and refined to match the application conflicts.

Appendix 2: Experimental setup

The experimental setup used for the experiment as shown in Fig. 2-1 consists of a telemanipulator master-device and visual feedback system. In the following sections the master-device, the connected control computer and the visual feedback system are presented. This chapter is concluded with a small guide on how to operate the experimental setup.

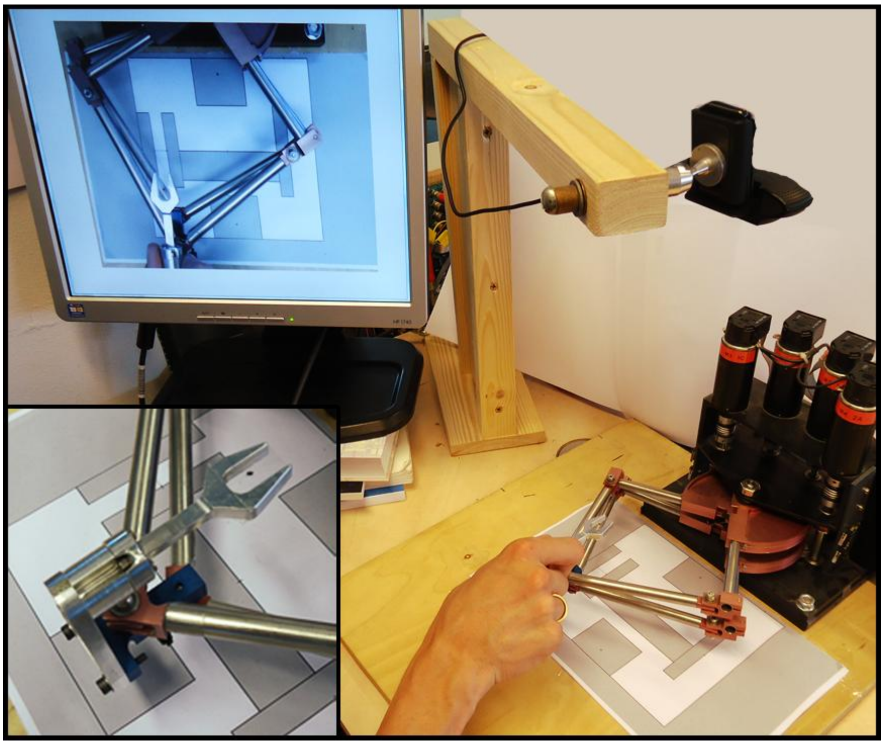

Fig. 2-1: Experimental setup on which the experiments have been performed. The subjects hold the master-device to perform the task and the visual feedback is provided on the computer screen. The inset shows the details of the master-device.

Telemanipulator

The experimental hardware setup used as control interface for the experiments is the Munin telemanipulator developed by Christiansson (Christiansson, 2007). It is a 3DoF planar telemanipulator consisting of a force redundant master device (Fig. 2-2) and a serial slave device (Fig. 2-3).

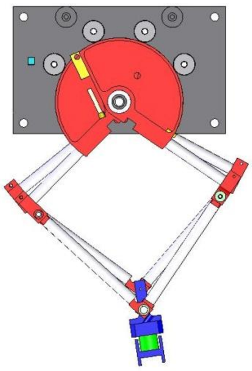
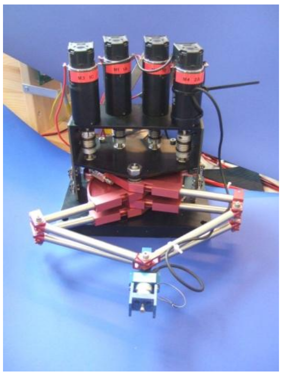

Fig. 2-2: Schematic & photo of Munin master device (from: Christiansson, 2007)

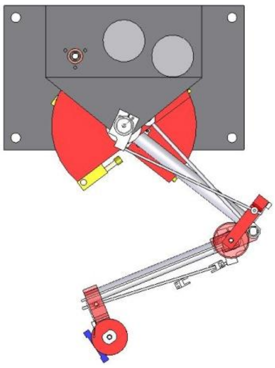
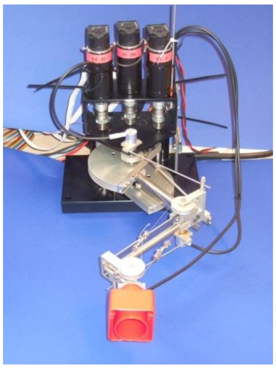

Fig. 2-3: Schematic & photo of Munin slave device (from: Christiansson, 2007)

The Munin-setup has been used to show the validness of the concept of hard master, soft-slave with an experiment on telemanipulation LEGO assembly tasks. (Christiansson, 2008) The Munin has been consequently used for research on the effects of quality of haptic feedback (Wildenbeest, 2010) and the effects of haptic shared control on task performance (Boessenkool, 2011). The setup has been altered to fit those experiments, mainly by improvement of the controller, increase of slave stiffness, adding a force sensor on the master device and redesign of interface to fit the evaluated bolt-and-spanner manipulation task.

For the experiment only the master-device has been used. The properties of the master devices for free-air movement are presented in Table 2-1 (Christiansson, 2007; Wildenbeest 2010).

Table 2-1: Master-device properties for free-air movement

Property	Value
Mass	0.136 [kg]
Stiffness	-0.027 [N/m]
Damping	11.3 [Ns/m]
Inertia	0.025 [gm^2]
Rotational Stiffness	0.003 [Nm/rad]
Rotational Damping	0.02 [Ns/m]

Telemanipulator and target computer modifications

The setup has been updated, the control computer is replaced and the telemanipulator is adjusted to be suitable for the experiments.

Slave device

Although the slave device was not used in the final experiment, the amplifier of the endpoint rotation motor has been replaced to avoid clipping of maximal motor currents.

Master device

The spanner interface as designed by Wildenbeest (Wildenbeest, 2010) has been altered for this experiment. The part of the spanner that is held with the palm of the hand was removed for the experiment (weight is approximately 10 g). The operator must hold the interface with two fingers only to make the task a two degrees of freedom-position task. The rotation of the spanner device is aligned with a stiffness controller avoiding that the operators perform a rotation task as well. Unfortunately a force sensor at the master device was not available.

Target Computer

The old target computer was limited in the logging capabilities and connection reliability was shortcoming. Therefore it is replaced with a new one. The xPC target system required changing some default values for better performance. The benchmark results of the Matlab xPC Target benchmark tool (`xpcbench('this')`) are shown in Fig. 2-4. Important configuration properties are presented in Table 2-2. Most remarkably the *Intel Xeon E3-1245 (3.30Ghz, 8MB, QC)* is a newer and faster CPU than any other in the list, but cannot outperform the two fastest.

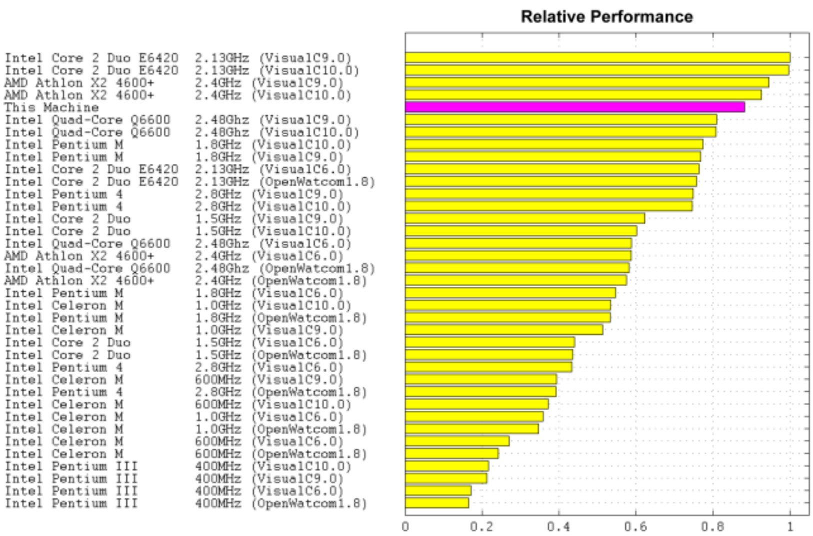

Fig. 2-4: Target Computer Benchmark results

Table 2-2: XPC Configuration parameters

Property	Value
MaxModelSize	16MB
MulticoreSupport	Off
NonPentiumSupport	Off
TargetRamSizeMB	2048

Control model

The Simulink model for the target control software was adopted from Wildenbeest (Wildenbeest, 2010) and Boessenkool (Boessenkool, 2011). The entire model was rebuilt to optimize performance. The adaptation of the shared control will be demanding and requires a relatively large percentage of the cycle time. Every subsystem is rebuilt and some subsystems are replaced by functions, primarily in kinematic calculations.

The model is rebuilt with the intention to be able to use the slave device as well. The slave device is still present in the system, only the PD position-position controller is disabled. It happened that in some cases that one of the amplifiers failed during initializing (not during the experiment). Amplifier A controls 3 out of 4 motors of the master device, amplifier B controls 1 master and 2 slave motors and amplifier C controls 1 slave motor. When amplifier B fails, the master device still works but with a limited non-homogeneous power distribution over the workspace. By enabling the slave device, failure of amplifier B is always detected and that situation is avoided.

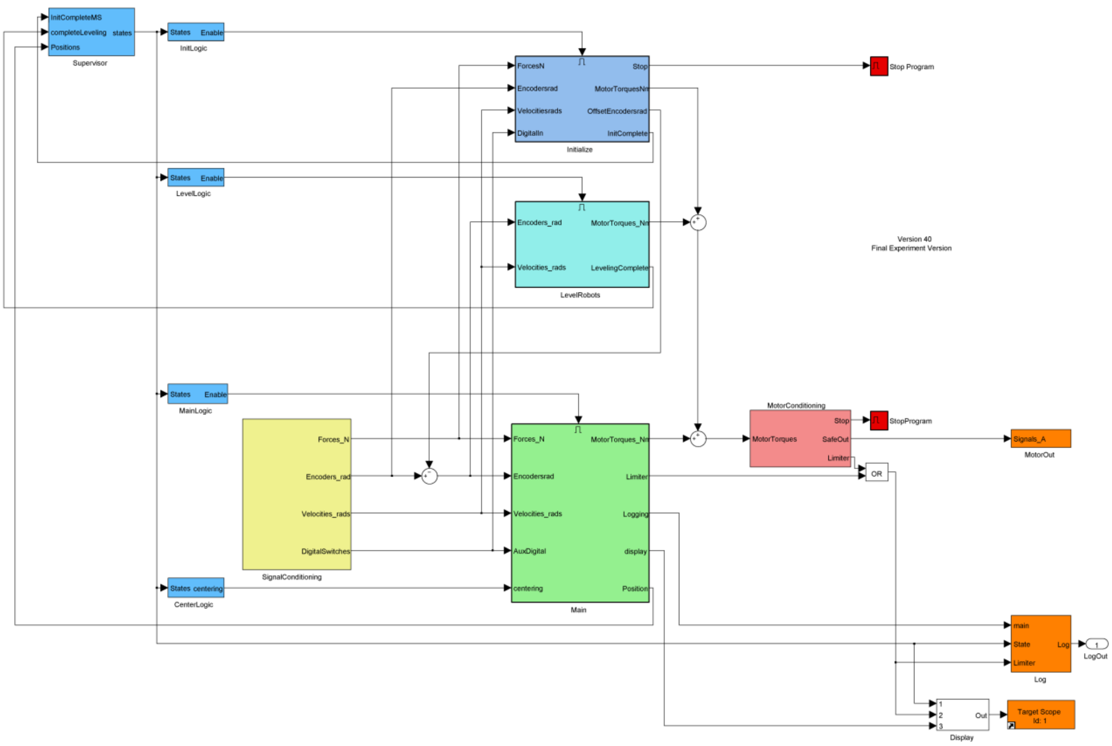

Fig. 2-5: xPC Target Simulink Model – Top level system

Target command and control software

MatWorks xPC Target is used for the control software of the Munin-setup. Matlab Simulink models are compiled and uploaded on the target computer. The graphical interface of xPC Target (Target Explorer) is unfortunately not available for Windows 64-bit computers (The previous host computer was outdated and is replaced.) Therefore a new dedicated graphical interface has been developed (Fig. 2-6: Target Control GUI). The interface provides control of the target computer and provides functionality for data logging and parameter tuning. A generic version for the interface is available for any other xPC Target application and can be easily extended (Fig. 2-7: Target Control GUI (simplified)). Full source code is available.

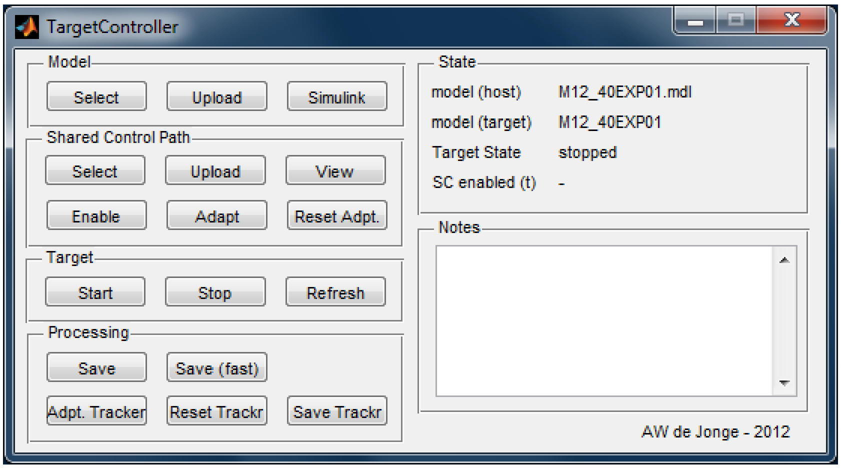

Fig. 2-6: Target Control GUI

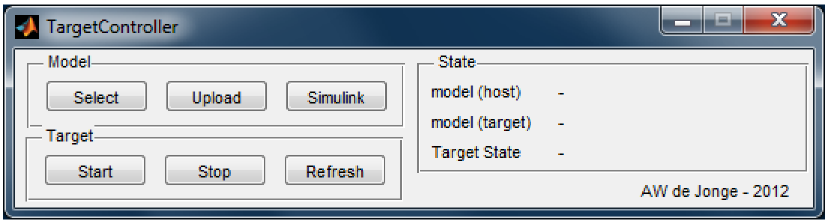

Fig. 2-7: Target Control GUI (simplified)

Visual feedback system

In the experiment the subjects is provided with visual feedback. Since the task environment is an underlay beneath the master-device the camera has to be position right above it. The master endpoint position is in the plane 9 cm above the base, the camera is placed 48 cm above the base and therefore the environment on the base is enlarged with $\frac{48}{48-9} * 100\% = 123\%$.

The camera system is a MATLAB video-script capturing data from the Logitec HD Pro C920 USB-webcam. Fig. 2-8 shows the video as displayed on a 17 inch screen at resolution of 800x600 pixels and video resolution of 640x480, the rest of the screen is gray. The frame rate is set to 20 Hz. In the bottom left corner occasionally minor image distortions were present, but that should not hinder task execution.

Creating a visual overlay might also be a proper way to provide an environment, however it appeared that the MATLAB video-capturing is heavily demanding computer resources and the MATLAB environment was not suitable for video overlay. Writing or acquiring dedicated software can allow visual overlays, if required.

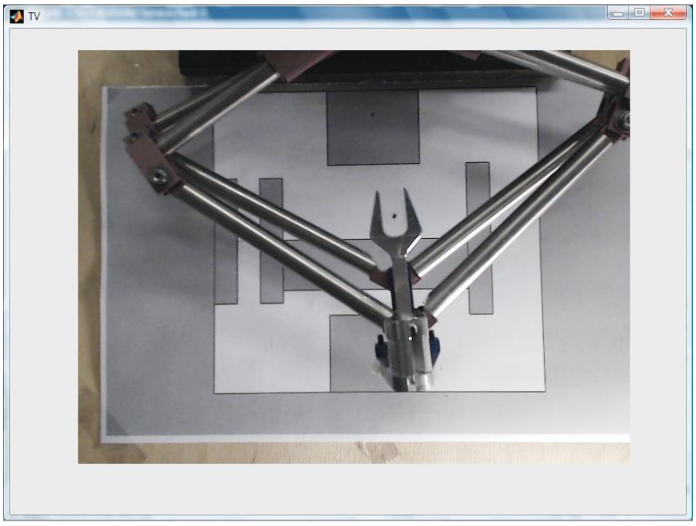

Fig. 2-8: Screenshot of visual feedback as given to the operator

Quick start on using the Munin-telemanipulator

The Munin telemanipulator consists of a master and slave device, powered with three amplifiers and connected with interface boards with the control computer (target computer). The host computer communicates with the target computer and allows control of the target computer through the MathWorks MATLAB interface in a Microsoft Windows 7 (64-bit) environment. The target computer runs the xPC Target operation system which is loaded from a USB-stick.

Take the following steps to use the Munin-telemanipulator:

1. **Initialize Host**
    a. Open MATLAB on host computer
    b. Run D:\haptics\Demo\startup.m
    c. Answer Yes to 'Open Controller?

2. **Prepare target computer**
    a. Power on target computer. *Make sure the amplifiers are disabled.*
    b. On Host: Press button Model->Select in TargetController GUI, then select 'MuninDemo.mdl'
    c. On Host: Press button Model->Upload. The model will be built and uploaded to the target computer. Wait until the uploading is completed. The target is now ready to use.

3. **Using Munin telemanipulator**
    a. Power on the target amplifiers, *always be careful!*
    b. Press Target->Start to start target. The target now will start initializing and leveling. *In case of failure, immediately switch of the target amplifiers.*
    c. Press Target->Stop to stop the machine. It is possible to save the data with Processing->Save

4. **Using adaptive shared control** *(All commands can be given while the target is running)*
    a. Enable with Shared Control Path->Enable
    b. Adapt with Shared Control Path->Adapt
    c. Reset adaptation with Shared Control Path->Reset Adpt.
    d. To follow adaptaion real-time on host computer Processing->Adpt. Tracker

Appendix 3: Pilot Experiment A – Manual control performance and behavior pilot-study

A preliminary study has been performed to gain more insight before developing an adaptive system and validating its usefulness. The idea is to find whether inter-subject variance is higher than intra-subject variance. High inter-subject variance is an indicating that it is useful to adapt to specific teleoperator, rather than adapting to a general population or trying to design an optimal shared control that is independent of operators. High inter-subject variability shows that trajectories are consistently performed different by various human operators.

This experiment does not have any form of shared control, to allow studying natural operator behavior. The set-up is very similar to the final experiment setup, except for the simplified task environment. Based on findings in this experiment the final experiment environment will be designed.

Usefulness of the metrics is also considered. The number of errors and time to complete will be used as performance measures. In automotive studies (Enache, 2009) time to line crossing is used as an indication of driver risk and might even function as a performance metric. High time to line crossing essentially means more safety and in teleoperation high safety levels can be seen as higher performance. For this application the time to line crossing is converted to time to contact as there are not lines, but hard environments. The question is whether this variable is useful for this kind of application. To gain more insight the distance to contact is calculated in the direction of movement as a velocity independent measure.

First a short rational is given, followed by the experimental methods. Then the results with performed trajectory and metrics are presented, which discussed after that.

Rational

Goal:	Validate the usefulness of adaptive shared control that adapts to the specific operator for the given workspace, environment and task. Statistically significant results are not required; the experiment is performed to gain insight.
Reasoning:	Operator specific adaptive shared control is beneficial when operators perform tasks consistently and consistently different from other operators. Therefore the inter-subject (operator) variance of the executed path must be lower than the intra-subject variance.
Approach:	Human factors experiment in which a small number of subjects that perform a free-air movement task in a constrained environment. The effects of shared control are not evaluated.

Methods

Subjects: Six subjects will participate in the experiment. All subjects must be right-handed and have no or limited experience with this particular hardware setup and task.

Apparatus: The set-up that will be used is the 3DoF planar Munin telemanipulator. For this experiment the slave device is decoupled and only the master device is active. The real-time OS runs at 1 kHz. Visual feedback of a camera positioned direct above the workspace is presented on a computer-screen.

Task description:

The subjects are asked to be seated in front of the master device and hold the master device with their thumb and index finger while their hand rests on the spanner device. A free-air movement task must be performed in the depicted environment as shown in Fig. 4-2. The operators are only allowed in the white area. A small vibration is presented inside the forbidden area to provide feedback.

The environment consist of 4 different corners with varying corner radii and widths to study the effect of criticality (corner radius) and freedom of strategy (widths) and see how operators respond to those situations.

The subjects are instructed to move in clockwise direction and do this fast without making any errors (going outside the boundaries).

Experiment design:

All subjects are tested on the same condition. The experiment consists of:

Training. The subjects are asked to perform the task as a training session. They repeat this task 15 times.

Experiment. The subjects are asked to perform the task, but for the real experiment. They have to perform 10 sessions of five tasks in a row. Between each session is a small break of one minute.

Metrics: Data is logged at 1 kHz. The main metrics are:

1. The average time to contact for each session. (The time it takes to hit the boundaries while maintaining the measured velocity at each measured point.)
2. Time to complete. The average time it takes to complete each task for each session.
3. Error rate. The number of faulty task executions for each session.

Results

For every subject all the trajectories for the repetitions are presented by a dotted line in Fig. 3-1. The average trajectories of all subjects are presented in Fig. 3-2.

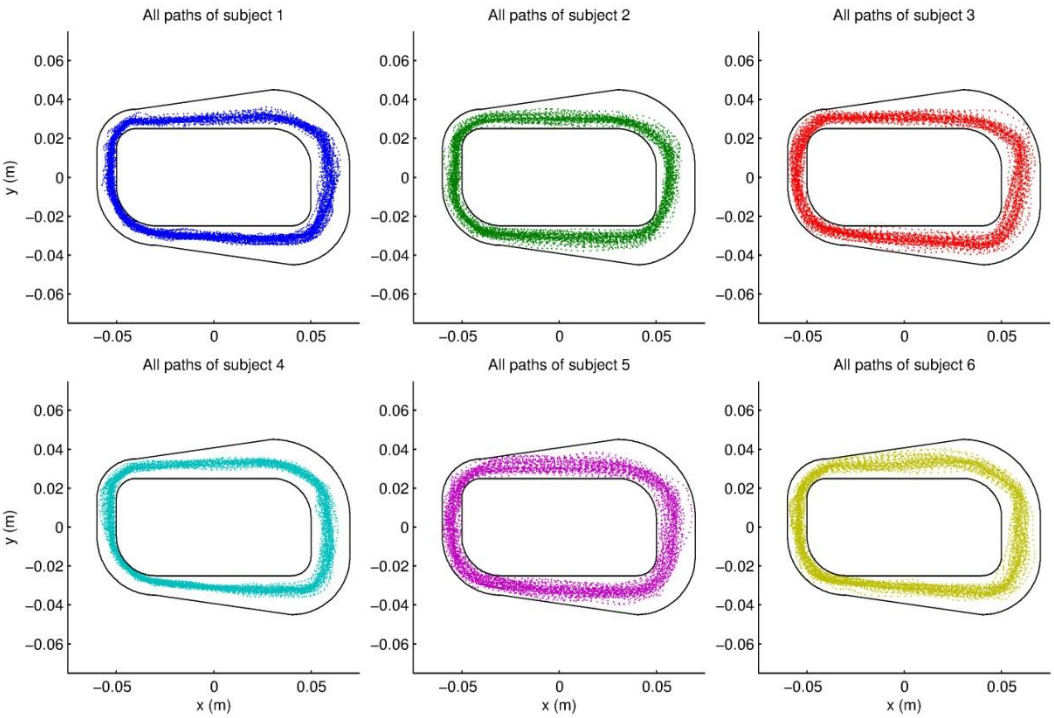

Fig. 3-1: All performed trajectories per subject

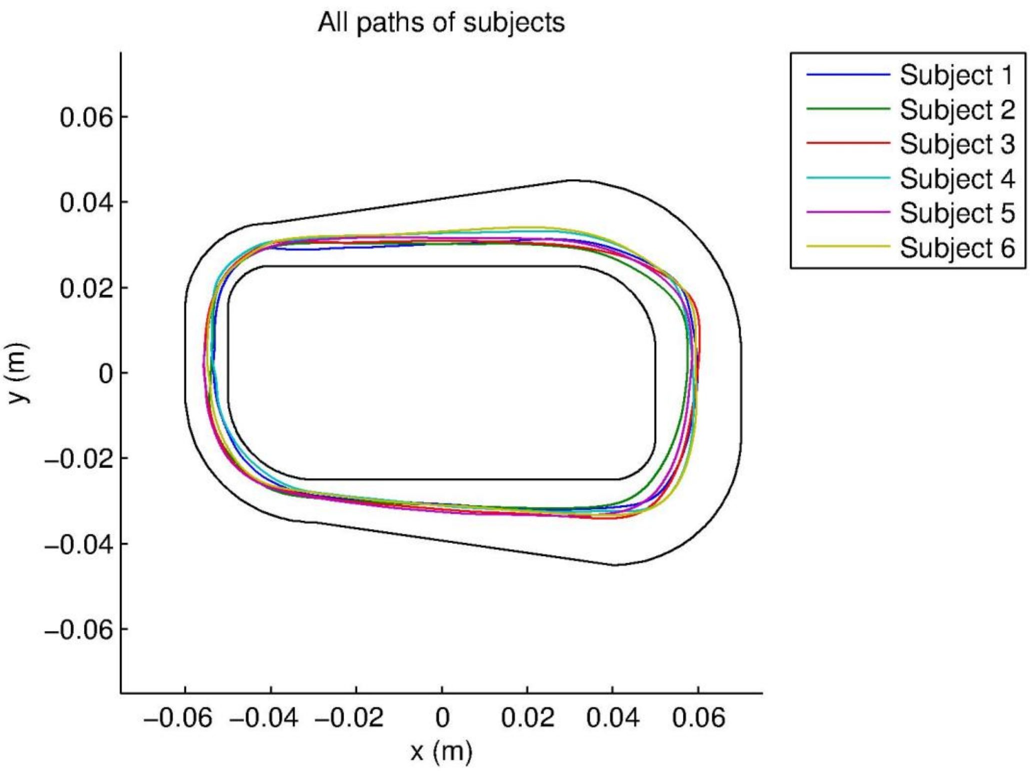

Fig. 3-2: Average trajectories of every subject

The performance metrics defines as completion time and numbers of errors are presented in Fig. 3-4 and Fig. 3-3.

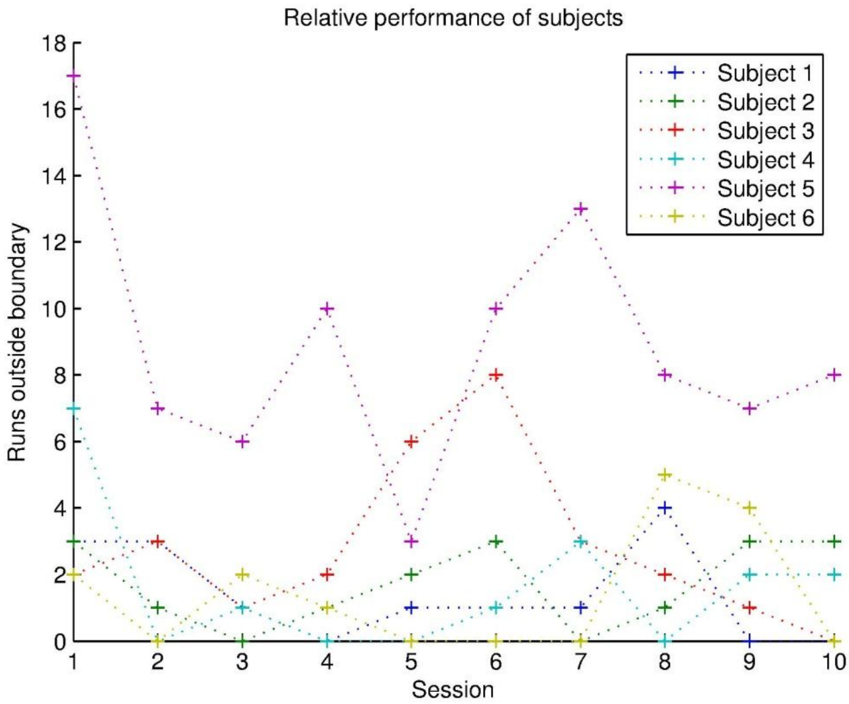

Fig. 3-3: Number of errors per five proper repetitions

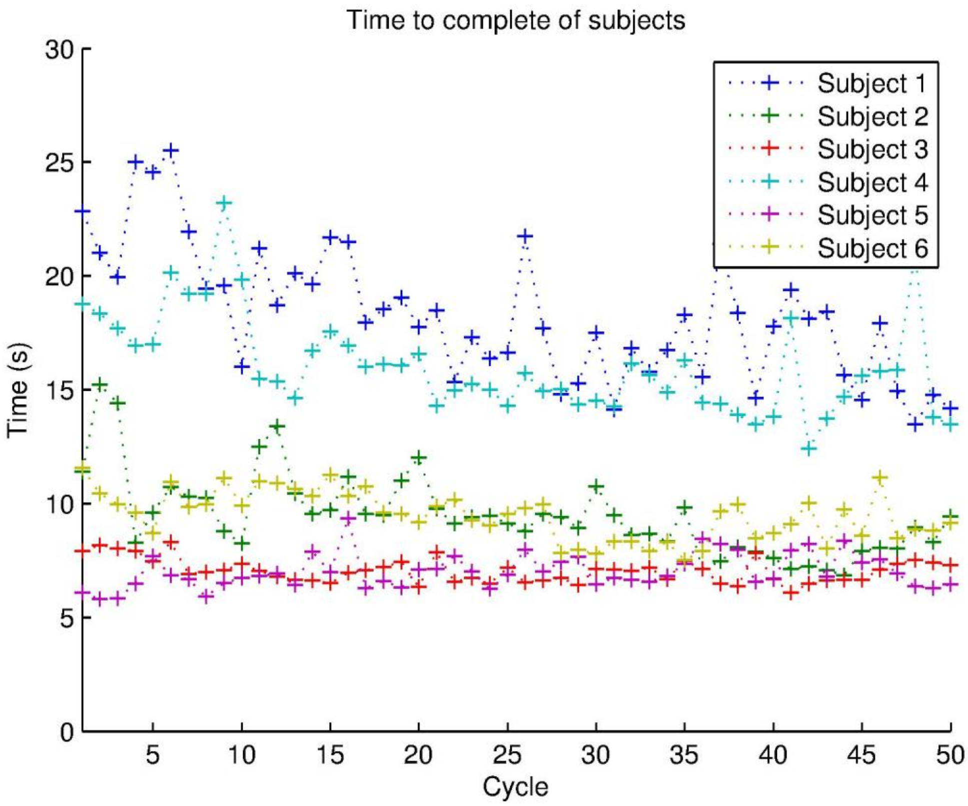

Fig. 3-4: Time to complete over the progress of the experiment

The time to contact and distance to contact are presented in the following figures:

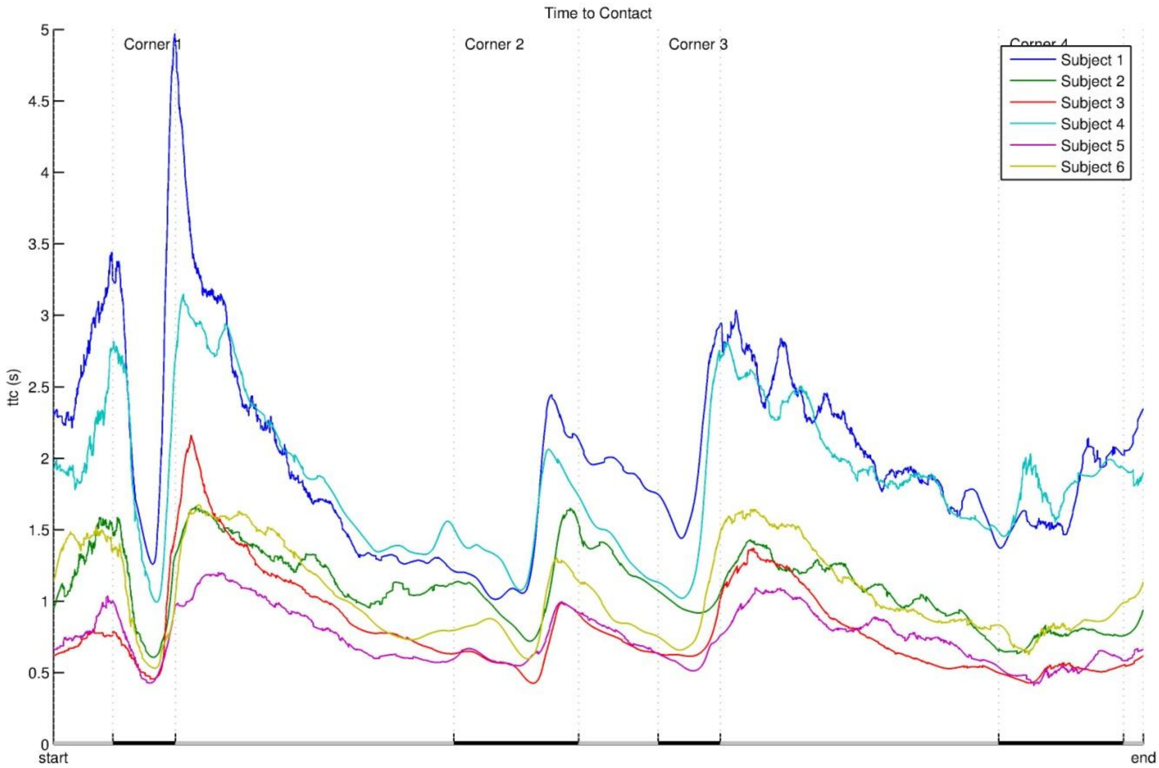

Fig. 3-5: Average time to contact for every subject, shown over the entire performed trajectory

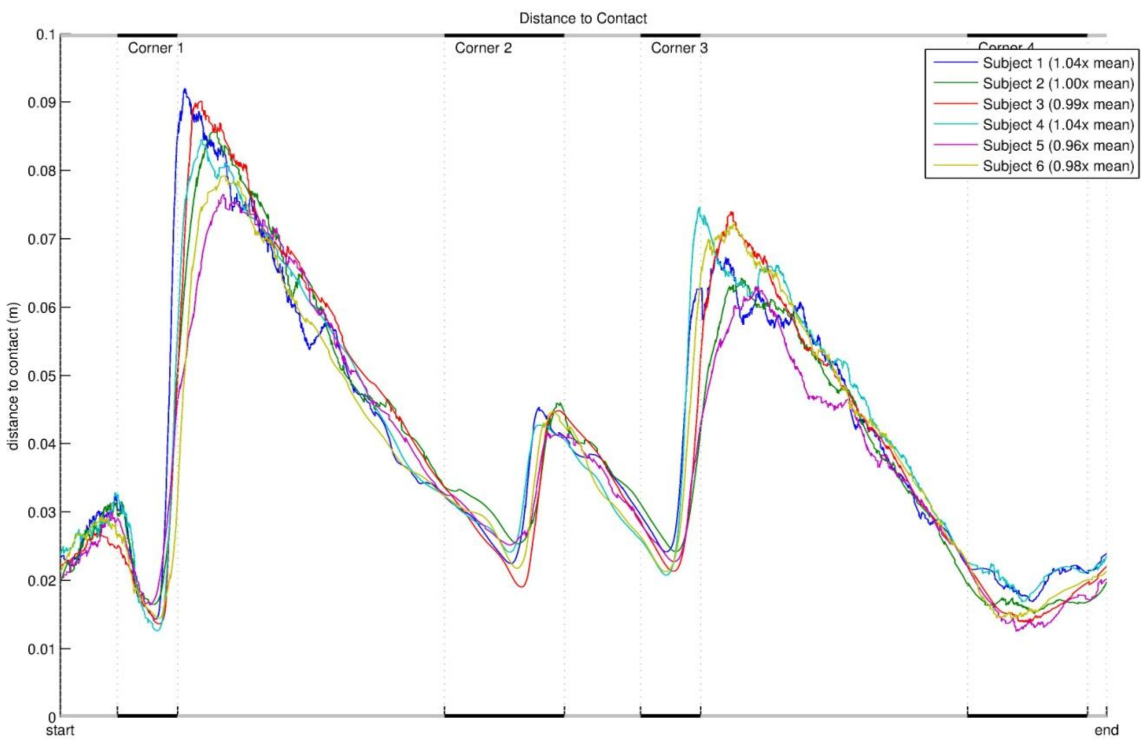

Fig. 3-6: Average distance to contact in the direct of motion for every subject, shown over the performed trajectory

Discussion

The time to complete is different between human operators. Subjects 1 and 4 show learning effects in time to complete, the other subjects seem to achieve similar performance in time to complete. It seems that over repetitions time to complete converges to the same value for every subject, although this might require a lot more repetitions.

The number of errors of every subject depends on the subject. For instance, subject 3 requires approximately two erroneous repetitions for every good repetition, while others make approximately one error per five repetitions. Intra-subject variations are clearly present.

From the two trajectory plots it appears that each the subject performs trajectories different, where the difference between subjects appears to be fairly consistent and larger than the variation in subjects. In the top left the average trajectories of every subject seems to be rather similar. The bottom left shows that subjects 1 and 4 seem to perform different trajectories, while the others seem to perform similar trajectories. Subjects 1 and 4 are also the slowest subject, which can be an explanation for the difference.

The bottom right corner shows various trajectories. While subjects 2 and 6 perform similar in terms of number of errors and time to complete, their trajectories are the two extremes of the followed trajectories. Apparently there is not optimal path for certain performance, at least for this particular corner. This corner yields different strategies on task execution.

The time to contact plot shows similar shapes at different levels for each subject. It seems as if the time to contact is only dependant on the velocity, analyzing the distance to contact there appears to be hardly any difference in the shapes. Especially in the corner there seems to be hardly any difference and it seems difficult to extract certain trends from the data. The sensitivity of this metric might be less suitable for this type of task both distance to contact and time to contact are in this form appear to be not very valuable metrics.

Conclusion

- Task instruction on performance needs to be limited. Interpretation of instructions ‘perform as good as possible’ allows different interpretation. Limitation on the number of errors is highly recommended.
- Sharp corners in wide areas allows freedom in trajectory strategy and are therefore recommended in an environment in which adaptation of shared control is evaluated
- The time to contact and distance to contact appear to have limited power in this task and environment and have limited sensitivity to describe operator behavior and performance.

Appendix 4: Pilot Experiment B – Performance and behavior pilot study with haptic shared control

Using the results from the previous experiment this experiment will be conducted to gain basic understanding of the effects of applying shared control. Especially the variance while using shared control is of interest. The distribution of the trajectories might show appropriate levels for adaptation thresholds, since adaptation requires some threshold to avoid adapting to natural variation in trajectory performance. The experiment will be performed on a two subjects from the first subject population.

Rational

Goal: Evaluate the change of behavior when operators are provided with shared control, continuing with the results from experiment A. Statistically significant results are not required; the experiment is performed to gain insight.

Reasoning: Inter subject variability appeared to be rather low, concluding from experiment A. However subtle differences were present. It is still a question what the effect will be of adding shared control with those subtle differences. Will human operator adapt to the shared control in this case? Or will there be are reason to make the support adaptive.

Approach: Human factors experiment in which a small number of subjects that perform a free-air movement task in a constrained environment. Subjects of the previous experiment are asked to participate in this research as well. (Evaluate the effects on velocity, position, time to contact, time to complete)

Methods

Subjects: Two subjects from the previous experiment will participate in the experiment.

Apparatus: The set-up that will be used is the 3DoF planar Munin telemanipulator. For this experiment the slave device is decoupled and only the master device is active. The real-time OS runs at 1 kHz. Visual feedback of a camera positioned direct above the workspace is presented on a computer-screen.

Task description: The subjects are asked to be seated in front of the master device and hold the master device with their thumb and index finger while their hand rests on the spanner device. A free-air movement task must be performed in the depicted environment. The operators are only allowed in the white area. A small vibration is presented inside the forbidden area to provide feedback.

The subjects are instructed to move in clockwise direction and do this fast without making any errors (going outside the boundaries).

Experiment design: All subjects are tested on the same condition. The experiment consists of:

1. Training. The subjects are asked to perform the task as a training session. They repeat this task 20 times.

2. Experiment. The subjects are asked to perform the task, but for the real experiment. They have to perform 10 sessions of five tasks in a row. Between each session is a small break of one minute.

Metrics: Data is logged at 1 kHz. The five main metrics are:

1. Path error. The integrated path error with respect to the average path of each session.

2. The average time to contact for each session. (The time it takes to hit the boundaries while maintaining the measured velocity at each measured point.)

4. Time to complete. The average time it takes to complete each task for each session.

5. Error rate. The number of faulty task executions for each session.

Results

The results of the experiment are shown in the following figures. Fig. 4-1 show the performed trajectories of the operators. The average trajectories are shown in Fig. 4-2. The number of errors is shown in Fig. 4-3 and the time to complete in Fig. 4-4. The measured standard deviations of subject 1 and 2 are resp. 1.0 mm and 1.5 mm and the average standard deviation is 1.2 mm.

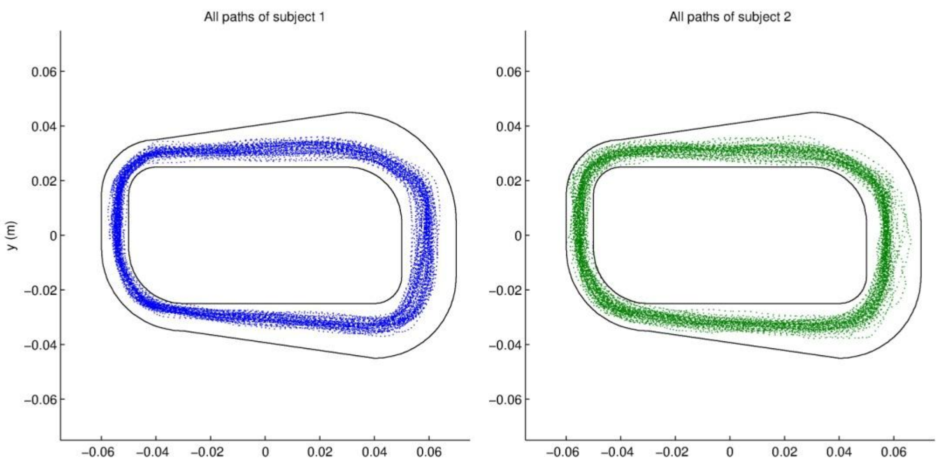

Fig. 4-1: Performed trajectories of the subjects

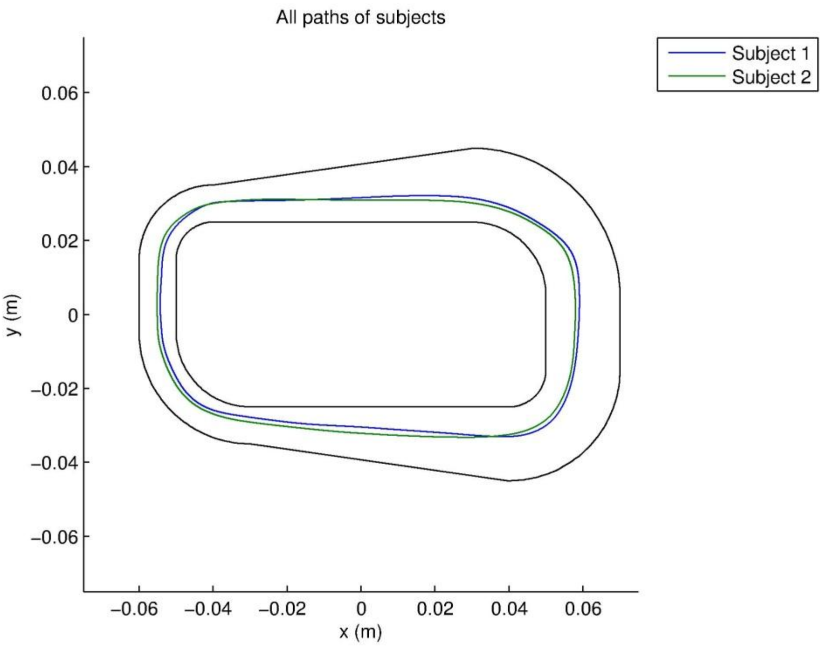

Fig. 4-2: Average trajectories of the subjects

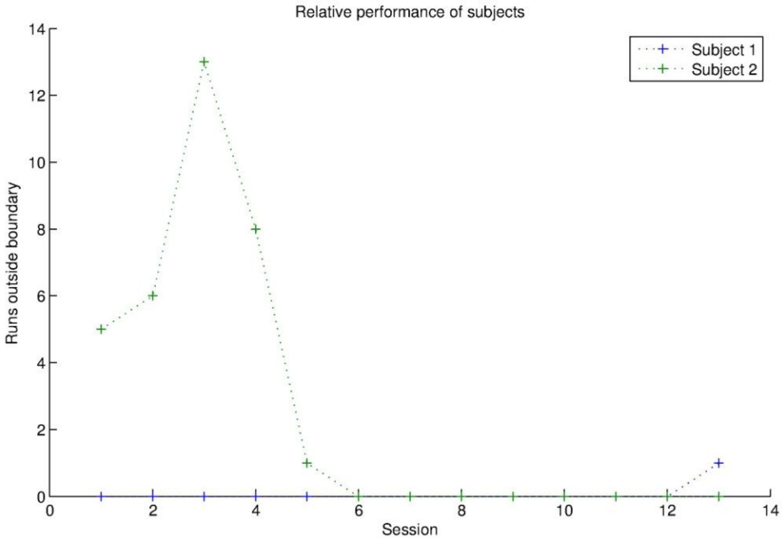

Fig. 4-3: Number of error per session

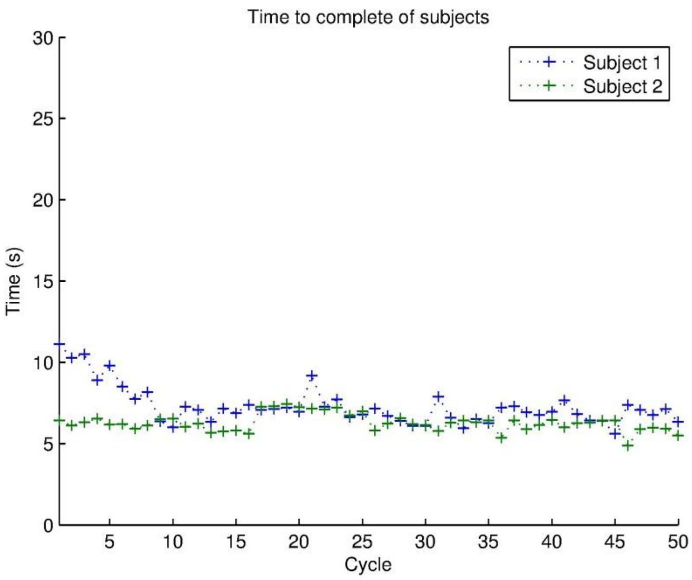

Fig. 4-4: Time to complete over the trials

Discussion & Conclusion

The average standard deviation of the trajectory is 1.2 mm. 95% of the trajectories are within ±2.5 mm of the average. Therefore 2.5 mm seems as a reasonable threshold for adaptation.

Although the number of subjects is very limited it appears that providing support does not extensively reduce the difference between performed trajectories.

- 2.5 mm seems an acceptable level of threshold for adaptation

Appendix 5: Adaptive haptic shared control design

The design of the adaptive haptic shared control support system is discussed in this chapter. The non-adaptive support system is discussed, followed by the adaptation method. The chapter is concluded with the implementation as used in the experiment.

Design of haptic shared control

In the work by Passenberg (Passenberg, 2010) three types of support systems for teleoperation have discriminated. The environment-, operator-, and task specific support system have their typical benefits. Task specific support systems substantially improve performance when the task and environment are known, with typical applications such as virtual fixtures and path support. Path support has shown to be at least as effective as the constraining virtual fixtures (Griffits, 2005) and will be used in this experiment.

In the experiment two types of support will be provided, the Centerline of Environment (CoE) support and the Individualized Manual Control Strategy (IMCS) support. The CoE-support is based on the exact middle between every walls of every section. At the intersection of the orthogonal lines a corner with a radius of 5 mm is applied to connect the centerlines. The CoE-support path is shown in Fig. 5-1a.

The other support path is the IMCS-path, which is recorded in a manual control condition without support for every individual subject. A typical path is shown in Fig. 5-1 as recorded from subject 1 of the experiment.

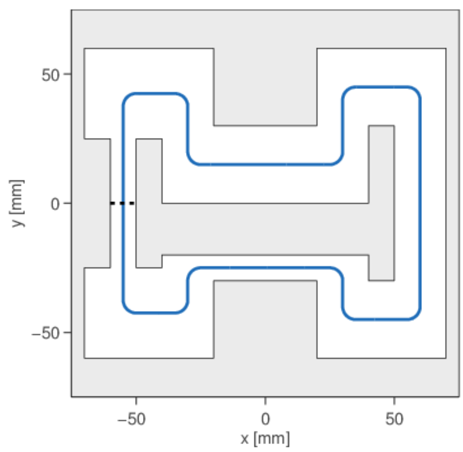

Fig. 5-1a: Centerline of Environment support path

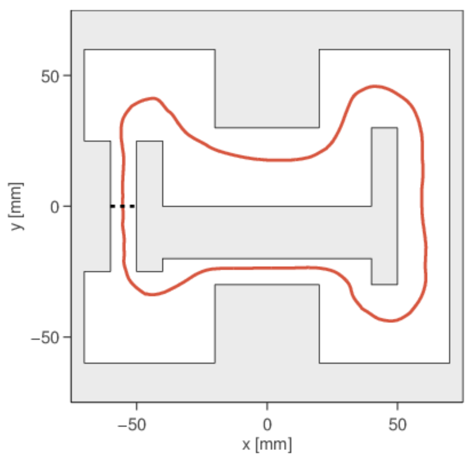

Fig. 5-1b: Individualize Manual Control Strategy support path

Design of adaptation

Measuring interaction forces is a common approach to detect conflicts between the human operator and support system (Marayong, 2004; Passenberg, 2011). Therefore the interaction forces are used as input for adaptation. Unfortunately the experimental setup does not have a force sensor that can measure the forces applied by the human operator. Nevertheless the forces applied by the support system are known (the error between the actual position and support path) and used as an approximation of the human operator forces.

The human operator has natural variance in task performance. The initial experiments showed that the standard deviation on the paths position is 1.2 mm measuring over the repetitions, corresponding with a 95% confidence interval of ±2.5 mm. This is therefore also the minimal level of adaptation, since it is unwanted to adapt to natural variations in task performance.

The adaption system has also some basic intelligence and will not adapt to paths that allow performing of dangerous trajectories. Therefore the path will not be adapted to trajectories closer than 2.5 mm to the wall.

The adaptation is based on a recursive exponential moving average filter. Essentially each repetition the path will be updated to the 0.8 time the current path position and 0.2 the actual position, if the adaptation threshold is overcome. The initial path will contribute only around 50% after 3 repetitions and after 14 repetitions the contribution of the initial path is less than 1%.

Implementation of adaptive haptic shared control

The adaptive shared control support is implemented as a subsystem in the Simulink Munin-controller model (files are available from the repository). The system is implemented in two-fold, one system for haptic shared control and one for adaptation of the shared control path. It is designed to adapt real-time and allow real-time changes commanded from the host computer. The system allows uploading of different paths real-time, enabling/ disabling adaptation and support, reset of adaption and changing of parameters (look-ahead time, stiffness etc)

Shared Control Path

The shared control path is a vector of 10.000 equally spaced x and y-positions. The maximal wall distance path has a length of 0.47 m, which means that every point is spaced approximately 0.05 mm, which is sufficiently close to the 0.03 mm position accuracy of the apparatus. Next to that completion time for this trajectory was in general somewhere between 5 and 20 seconds, meaning 5.000 to 20.000 control cycles which is in the same order as the resolution of the path.

Haptic Shared Control Support

The haptic shared control support system implementation is inspired by the support implementation by Boessenkool (Boessenkool, 2011). The support consists of an embedded m-function with inputs *look ahead position*, *shared control path*, *stiffness*, *enable* and the output *Shared Control Force*.

```
function [Force, index] = ApplyPath(x, path, stiffness, enable, previous_index)
% Haptic Shared Control Assistant
% Generates Shared Control Forces based on the position input and path
% input
% June 2012 - AW De Jonge
```

The current path-index is also available for the logging, adaptation and iteration counter. Every control cycle the minimal distance between the look-ahead position and the closest path position is calculated. Of the path only the 1000 points around the previous path point (-500 to +500) are calculated, wrapped around the total length of the path. The interval is chosen to avoid jumping to the other half of the path, while in the corner smooth cutting of paths is allowed.

```
    % interval settings
    steps = 500;
    index_length = 10000;

    % allowed path indices
    path_indices = zeros(1,2*steps+1);

    % create allowed path index vector and prevent overflows
    for k = 1:2*steps+1
        path_indices(k) = k - steps + previous_index;
        if path_indices(k) < 1
            path_indices(k) = path_indices(k) + index_length;
        elseif path_indices(k) > index_length
            path_indices(k) = path_indices(k) - index_length;
        end;
    end;

    % find minimum
    dist_fun = sqrt((x(1) - path(path_indices,1)).^2 + (x(2) -
path(path_indices,2)).^2);
    [~, index_local] = min(dist_fun);

    % retrieve path index (already wrapped around)
    index = path_indices(index_local);
```

The distance between the position of the closest path-index point and look-ahead position are multiplied with the shared control stiffness and applied as the shared control force.

```
    if enable
        Force = [0;0;0];
        % find carthesian forces = k * distances
        Force(1) = stiffness * (path(index,1) - x(1));
        Force(2) = stiffness * (path(index,2) - x(2));
        return;
    else
        Force = [0;0;0];
        return;
    end;
```

Adaptive Haptic Shared Control

The adaption system consist of a embedded m-function with the inputs *look ahead position, initial shared control path, path-index and reset and enable* and outputs the adapted *shared control path*. First it is checked whether the position is within the safety boundaries of the environment, if so no adaptation is done.

```
function path_out  = adaptPath(index, x, enable, path_init, innerBound, outerBound,
reset)
    % Update shared control path to current position

    ...

    % define environment boxes
    xb = [0;    0;    0;    0;    -4.5; 4.5;  -6.5] / 100;
    yb = [0;    4.5;  -4.5; -1;   0;    0;    0   ] / 100;
    w  = [14;   4;    4;    8;    1;    1;    1   ] / 100;
    h  = [12;   3;    3;    2;    5;    6;    5   ] / 100;

    % define safety margin
    w = w + 0.0025 * 2;
```

```
    h = h + 0.0025 * 2;

    % define environment sizes
    left   = xb - w/2;
    right  = xb + w/2;
    bottom = yb - h/2;
    top    = yb + h/2;

    % check whether position is inside box 1 (total environment) and not
    % inside other boxes (obstacles)
    mustbeinside = [1;0;0;0;0;0;0];
    OK = true;
    for k=1:length(mustbeinside)
        if left(k)   <= x(1)  && x(1) <= right(k)  && ...
           bottom(k) <= x(2)  && x(2) <= top(k)
            if mustbeinside(k) == 0
                OK = false;
                break;
            end;
        elseif mustbeinside(k) == 1
            OK = false;
            break;
        end;
    end;
```

After that it is check if the current index is higher than the previous one, to prevent re-updating of the path. The path may only be updated once every repetition.

```
    % adapt only if is allowed
    if OK == true

        % calculate step in index
        di = index - index_local;

        % if step is negative, skip
        if di <= 0
            return;
        end;
```

The minimal distance between the path and the actual position is checked and if so the path-position is updated contributing 20% of the current position.

```
        % current path position
        xp = path_local(index,:);

        % minimal error
        if sqrt((xp(1) - x(1))^2 + (xp(2)-x(2))^2) > 0.0025
            % new x position of path
            xn = p * [x(1) x(2)] + (1-p) * xp;
            path_local(index,:) = xn;
```

If required linear interpolation is performed is path points have been skipped (due to cutting of corners).

```
% linear interpolation if required
            if di > 1
                % step value
                if index_local < 1
                    dx = (xn - path_local(index_local+10000,:)) ./ di;
                else
                    dx = (xn - path_local(index_local,:)) ./ di;
                end;
                for k=0:di-1
                    ii = index-k;
```

```
                if ii < 1
                    path_local(ii+10000,:) = xn - k*dx;
                else
                    path_local(ii,:) = xn - k*dx;
                end;
            end;
        end; % end interpolation
```

As a final step the entire path is smoothened by a moving average on the 50 steps before the current index. The moving average is applied over -15 and + 15 path indices.

```
% apply moving average filter
    if index - index_local < 2500 % ensure not initializing
        for k=index_local - 50:index - 50
            new_val = zeros(1,2);
            for m=-15:15
                if k+m < 1
                    new_val(1,:) = new_val(1,:) + path_local(k+m+10000,:);
                elseif k+m > 10000
                    new_val(1,:) = new_val(1,:) + path_local(k+m-10000,:);
                else
                    new_val(1,:) = new_val(1,:) + path_local(k+m,:);
                end;
            end;
            if k < 1
                path_local(k+10000,:) = new_val(1,:) ./ 31;
            else
                path_local(k,:) = new_val(1,:) ./ 31;
            end;
        end;
    end;
```

To prevent direct updating of the path the entire path is buffered and delayed by a separate function. This function updates the first half of the path at 75% path progress and the second half at 25% path progress.

```
function path_out  = DelayBuffer(index, path_new, reset)
    persistent path_buffer;

    if isempty(path_buffer)
        path_buffer = path_new;
    end;

    if reset
        path_buffer = path_new;
    elseif index > 7400 && index < 7600
        path_buffer(1:5000,:) = path_new(1:5000,:);
    elseif index > 2400 && index < 2600
        path_buffer(5001:10000,:) = path_new(5001:10000,:);
    end;
    path_out = path_buffer;
```

Appendix 6: Adaptive haptic shared control experiment

This appendix describes some details of the experiment. First of all the task instruction of the subjects is presented. Then the condition experiments are given, followed by the metrics used for analysis including the two subjective questionnaires.

Task instruction as presented to the subjects

All subjects were given this instruction verbally in English. All non-native English speakers reported to have no problem understanding the instructions.

You will perform a task in the presented environment. The computer screen will provide you with visual feedback to perform the task. You must place your thumb and index finger and/or middle finger of your right hand on the master device to control it. You have to move through the presented environment from the start line to the start line in clockwise direction.

Imagine that this environment a nuclear power plant and you have to move a radio-active bar from the start point to the end point. The dot you see on the device is the top view of this fragile bar. The gray areas are the walls of the nuclear power plant and you are not allowed to hit the wall with the nuclear bar. If you hit the wall more than two times, the bar will break and the power plant has to be stopped.

For this experiment you have to continuously move the bar a twenty times from start to endpoint in clockwise direction without hitting the wall. If you hit the wall, you will feel indicating vibration. If you hit the wall more than two times during the experiment, we have to stop the session and restart. The experiment starts with training session to familiarize with the device and the task. After that we will do the actual experiment itself. The experiment will take approximately one hour in total.

Now we will start with three training session, in which the (imaginary) radio-active bar is replaced with a rubber one. So you are allowed to hit the wall, but keep in mind what task you are training for. The training session is concluded with a validation session in which you have to perform the task properly; you have to move twenty times through the environment without hitting the wall more than twice. If you do fail, you have to perform the validation session again until you succeed.

While you are performing the task, you do not have to count the number of trials, I (the experiment conductor) will count the trials so that you can focus on your task. If you have any questions, feel free to ask. Are you ready to start?

Task environment

The task environment in which the operators have to perform the task is based on the initial experiment. It appeared that sharp corners and alternation wide and narrow areas yielded different strategies for operators. The task environment is shown in Fig. 6-1, the dashed line is the starting point for the task and subjects have to perform the task in clockwise direction. The wide corners allow much freedom in the corners. The left and bottom straight sections are narrow enforcing lower velocities, while higher velocities can be achieve in the wider top and right straight parts.

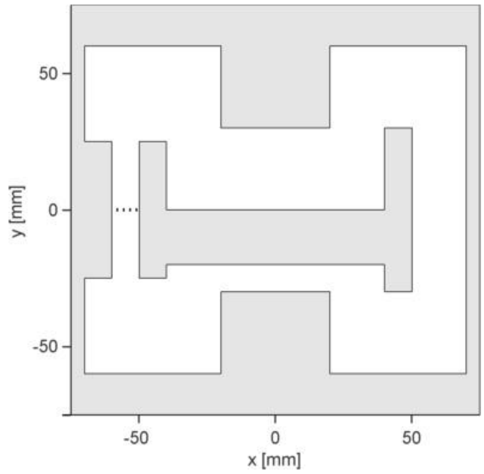

Fig. 6-1: Task Environment

Randomized Experiment Conditions

The subjects were presented with randomized experiment conditions as listed in Table 6-1. The conditions are ordered such that every condition order is balanced. Optimal balance however can only be achieved by 24 subjects, which was for this experiment unpractical.

Table 6-1: Experimental Conditions per Subject

Subject	Condition 1	Condition 2	Condition 3	Condition 4
1	NA CoE	NA IMCS	A CoE	A IMCS
2	A IMCS	A CoE	NA IMCS	NA CoE
3	NA IMCS	A IMCS	NA CoE	A CoE
4	A CoE	NA CoE	A IMCS	NA IMCS
5	NA CoE	A CoE	NA IMCS	A IMCS
6	A IMCS	NA IMCS	A CoE	NA CoE
7	A CoE	A IMCS	NA CoE	NA IMCS
8	NA IMCS	NA CoE	A IMCS	A CoE
9	NA CoE	A IMCS	A CoE	NA IMCS
10	NA IMCS	A CoE	A IMCS	NA CoE
11	A IMCS	NA IMCS	NA CoE	A CoE
12	A CoE	NA CoE	NA IMCS	NA IMCS

Metrics

The results of the experiment are analyzed with the following metrics:

- Task performance metrics
    - Task Completion Time
    - Number of Errors
- Control Effort metrics
    - Shared Control Force
    - Steering Corrections
- Subjective Metrics
    - NASA-TLX
    - Questionnaire

The metrics are described in detail in the following sections.

To compare different trajectories, every position is normalized and mapped to a path through the environment, such that every instance in time is normalized to the progress in the environment with an index ranging from 1 to 10000. The metrics are low-pass filtered at 100 Hz.

The time to contact and distance to contact are not studied. As concluded in the previous experiments, the sensitivity and usefulness of these metrics is limited. After rudimentary analysis the usefulness of these metrics for this experiment could not be shown.

Performance: Task completion time

The task completion time is recorded by measuring the time between consecutive crossings of the x-axis on the left side. The time is measured in 1 ms resolution.

Performance: Number of errors

When the operator hits the wall a trigger in the control program will register this event. Retriggering can occur after 500 ms, to allow the operator to continue with the trial.

Control Effort: Shared Control Force

The shared control force as applied by the support system is recorded in x and y-direction, and the length of this vector (magnitude) is used for the analysis.

Control Effort: Steering Corrections

The steering corrections are an indication of the amount of steering performed by the operator. The absolute value of the angle of direction of motion between individual points of the trajectory is summed, multiplied with 1000 this gives the total amount of steering correction in one second, measured in degrees.

Subjective Metrics: NASA-TLX

The NASA-TLX (Hart, 1988) is a common measurement for the subjective workload of operators and has been used in many human factors experiments. Subjects can indicate their perceived workload in six scales and that combined and weighted score can be used as a measure of workload. The assessment is performed in two parts. First the operators are presented with a computer questionnaire in which they have to indicate their individual contributions of the six scales to workload (available from: http://humansystems.arc.nasa.gov/groups/TLX/). After the four repetitions per condition the subjects were presented with a paper version of the TLX along with the rating scale definitions as shown in the following section.

Subjective Metrics: Questionnaire

To gain more understanding in how subjects perceive the different types of support a questionnaire was presented as shown on the next page. Subjective evaluation is split up into six questions. The purpose of the first question (about difficulty) was to comprehend the experience task difficulty, since the task difficulty will be likely to affect the level of usage of the support system. When the task is more difficult, it is expected that the human operator will rely more on the support system and might have less reason to fight with the system.

The second question about the level of training is used to validate the level of training from the subject perspective. The third question (is the system helping of counteracting) is about the perceived support type and whether it does conflict with the desired subject trajectory. The fourth question is strongly related (matches your preferences), however this exclude performance from the rating.

The fifth question (do you like the kind of support) is a direction question on how the subjects perceived the supporting system, directly implying they were provided with support. The sixth question (does the system make the task easier for you), is somewhat similar to the third question, but it is possible that the system counteracts the subjects preferred task execution, but makes the task easier.

The questions are strongly related and the differences between the questions are not strongly articulated. However subtle differences in the questions can yield more articulated differences between the conditions and therefore these six questions are asked.

Questionnaire

Gender: M / F Age:

1) How difficult was the task for you to perform?

Very easy								Very hard
1	2	3	4	5	6	7	8	9

2) Do you think you were trained enough to perform the task?

Not at all								Definitely
1	2	3	4	5	6	7	8	9

3) Do you have the feeling that the system is helping or counteracting you?

Counteracting				Neither				Helping
1	2	3	4	5	6	7	8	9

4) Do you feel the system matches your preferences?

Not at all								Definitely
1	2	3	4	5	6	7	8	9

5) Did you like the support you were provided?

Not at all								Definitely
1	2	3	4	5	6	7	8	9

6) Do you feel that the system made the task easier for you?

Not at all								Definitely
1	2	3	4	5	6	7	8	9

7) Do you have any form of limitation *(physical, mental, etc.)* or advantage *(experience with similar systems, etc.)* that has or might have influenced your performance?

8) Do you have other comments you would like to make concerning this experiment?

NASA-TLX Score Assessment

Figure 8.6

NASA Task Load Index

Hart and Staveland's NASA Task Load Index (TLX) method assesses work load on five 7-point scales. Increments of high, medium and low estimates for each point result in 21 gradations on the scales.

Name	Task	Date

Mental Demand — How mentally demanding was the task?

Very Low — Very High

Physical Demand — How physically demanding was the task?

Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?

Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low — Very High

Source: Hart, 1988

NASA-TLX Rating Scale Definitions

RATING SCALE DEFINITIONS

Title	Endpoints	Descriptions
MENTAL DEMAND	*Low/High*	How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?
PHYSICAL DEMAND	*Low/High*	How much physical activity was required (e.g., pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?
TEMPORAL DEMAND	*Low/High*	How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?
EFFORT	*Low/High*	How hard did you have to work (mentally and physically) to accomplish your level of performance?
PERFORMANCE	*Good/Poor*	How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?
FRUSTRATION LEVEL	*Low/High*	How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Adopted from: Hart, 1988.

Data management

The experiment generated a vast amount of data. 4.2 GB of data was recorded for the twelve subjects and their 370 to 400 trials (including training and validation). The raw data is available from the repository as well as the processing script. Processing, calculating data to retrieve metrics can take up to 4 hours. Therefore processing, calculating and metric calculations are performed in intermediate steps to have fast access to metrics.

Folder structure

The software and experiment data are organized such that working on multiple machines is not a problem. The general structure of the software divides results from this experiment from the actual application. The application folder can be used for any other experiment, while the experiment folder contains all experiment specific data. The structure is as follows, starting from the root folder:

```
/application                Control application of Munin Target Computer
      /config               Configuration files for xPC Target
      /lib                  Simulink Control Model dependencies
      /models               Simulink Control Models for xPC Target
      /paths                Shared Control Paths and Virtual Environments for Model
      /run                  Build folder of Compiler
      /tools                Host computer tools and scripts
            /experiment     Scripts required for this experiment during experiment
            /scripts        Host xPC Target Scripts
            /target         Target Controller GUI and Scripts
            /visualizer     Visualize Tool for Replay of performed trajectory
/experiment                 Data and script for analysis for this experiment
      /data                 Experiment data
            /local          Generated intermediate data
            /raw            Raw logged experiment data
            /subjectinfo    Contains links between subjects and raw data files
            /TLX            Retrieved TLX results
      /output               Output (figures etc.)
      /scripts              Post-processing scripts
/startup.m                  Initializing script, sets path dependencies
```

Data Processing

The processing of was performed with the following steps

Online measurements	Raw data retrieved from the xPC target PC is stored as a timestamped file in `experiment/data/raw/` Raw data from the Adapt Tracker (script on Host PC tracking the adaptation of shared control path) is also store in `experiment/data/raw/` For every subject a list of files is maintained (`experiment/data/subjectinfo/`).
Preprocessing	Only relevant signals are extracted and store in `/experiment/data/local/`
Structuring	The individual rounds of each trial is extracted and position, velocity and force data is normalized to a tracking path to allow comparison between cycles and trials. These structures are saved in `experiment/data/local` The adapt tracker data is also restructured, matched to the xPC target structure (requires matching of timing) and saved.
Retrieval of Metrics	The metrics are retrieved from the structured data and stored in `experiment/data/local`

Plotting	Many visualization of the data
Statistics	Of the relevant metrics the ANOVA, Multi-comparison test are performed and corresponding figures are generated.

The function `experiment/data/scripts/PrepareAll.m` will invoke all processing scripts for preprocessing, structuring and retrieval of metrics. Visualization of many metrics can be done with `/application/tools/experiment/visualize/Visualize_All.m`. Statistics and clean plots can be created for the entire experiment or sections of the environment with `experiment/data/scripts/{PerSection/}GenerateOutput.m`

Appendix 7: Experimental results

This section shows some more details of the results of the experiment described in the previous appendix chapter. The first section describes the results of the metrics, followed by the results per subject and the adaptation progress.

Metrics

The mean values and confidence intervals of the five experimental metrics are shown in Fig. 7-1, with as addition the velocity metric that is not used in the final analysis, but is shown to compare with performance effects. The ANOVA-results are shown in Table 7-1 (p-values) and Table 7-2 (F-values).

Table 7-1: ANOVA Results: p-values of interaction model

Metric	Adaptive	Path	Subject	Adaptive*Path	Adaptive*Subject	Path*Subject
Completion Time	*0.668*	*0.583*	**0.025**	*0.889*	*0.354*	*0.188*
Errors	*0.658*	*0.594*	**0.018**	*1.000*	*0.180*	**0.007**
Shared Control Forces	*0.000*	*0.669*	*0.464*	*0.467*	*0.141*	**0.030**
Steering Corrections	**0.008**	**0.012**	*0.208*	**0.042**	*0.826*	*0.264*
NASA-TLX	*0.186*	*0.551*	**0.045**	*0.320*	*0.561*	*0.502*

Table 7-2: ANOVA results: F-values of interaction model

Metric	Adaptive	Path	Subject	Adaptive*Path	Adaptive*Subject	Path*Subject
Completion Time	*0.194*	*0.320*	**4.287**	*0.021*	*1.261*	*1.734*
Errors	*0.208*	*0.301*	**3.533**	*0.000*	*1.767*	**4.867**
Shared Control Forces	*55.846*	*0.194*	*1.047*	*0.567*	*1.954*	**3.302**
Steering Corrections	**10.401**	**9.162**	*2.655*	**5.323**	*0.558*	*1.477*
NASA-TLX	*1.993*	*0.378*	**8.589**	*1.086*	*0.910*	*0.997*

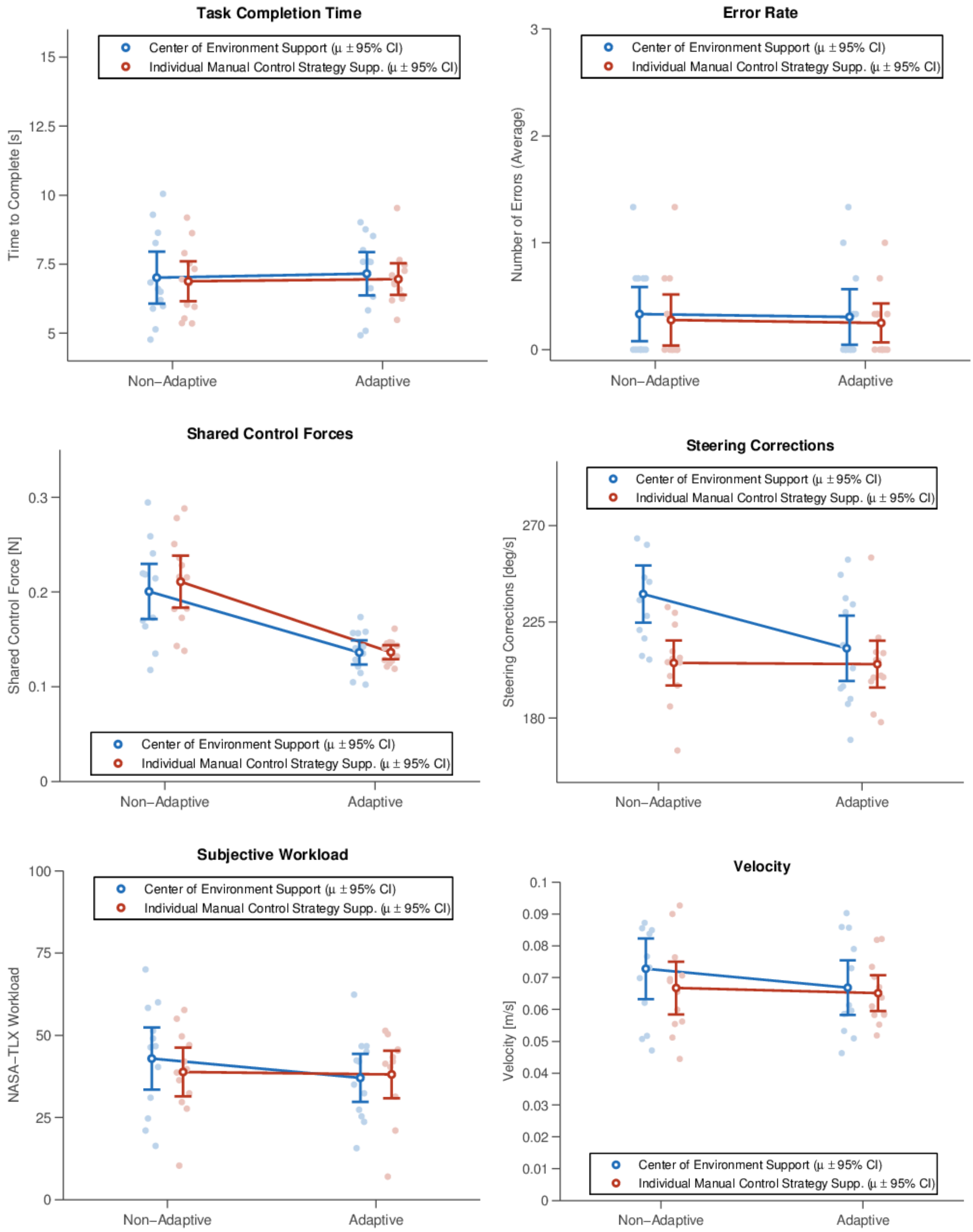

Fig. 7-1: Experiment metrics and velocity metric

Subjective Metrics

The subject responses to the questionnaire are shown in Fig. 7-2. The results of the three-way ANOVA on these 6 metrics are shown in Table 7-3. The ANOVA shows that the adaptation had an effect on the perceived matching of preference and on whether the subjects liked the system. The provided path has an effect on the perceived level of helping, the matching of preferences and the likability of the support. The level of training is depending on the subject and the perceived difficulty is effected by the combination of support path and subjects.

The subjective measurements suggest that both the adaptation and the supported path seem to have a position effect on the subjective responses.

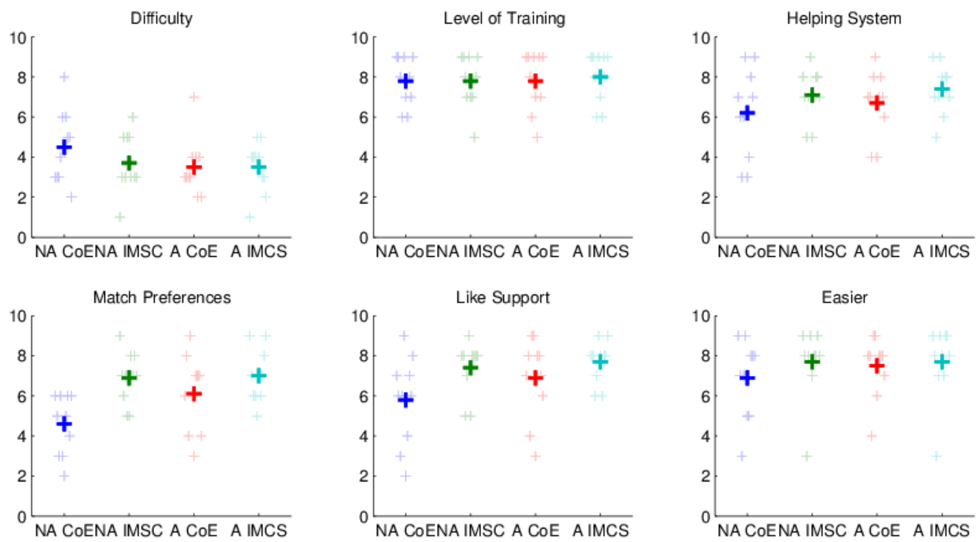

Fig. 7-2: Questionnaire Results

Table 7-3: ANOVA Questionnaire Results: p-values of main effects and interaction effects

Metric	Adaptive	Path	Subject	Adaptive*Path	Adaptive*Subject	Path*Subject
Difficulty	*0.119*	*0.461*	*0.068*	*0.206*	*0.236*	**0.037**
Training	*1.000*	*0.296*	**0.029**	*0.658*	*0.146*	*0.054*
Helping	*0.267*	**0.027**	*0.060*	*0.889*	*0.692*	*0.419*
Preference	**0.003**	**0.008**	*0.309*	*0.277*	*0.990*	*0.284*
Like Support	**0.023**	**0.023**	*0.169*	*0.509*	*0.947*	*0.353*
Easier	*0.815*	**0.016**	*0.408*	*0.834*	*0.642*	*0.890*

Results per subject

The following section addresses the individual results for the metrics and the performed trajectories.

Performance metrics per subject

In the following sections the metrics of individual subjects are shown. The ANOVA showed no effect of the two experimental factors (adaptation and support path) in the performance metrics completion time and number of errors. The subject-factor in the ANOVA showed to affect the metrics, therefore individual results are shown. The completion times are shown per subject in Fig. 7-3 per condition. The conditions are randomized and possible learning effects are shown in Fig. 7-4. While learning effects appear to be present for some subjects, the effect is not so dominant yielding invalid analysis.

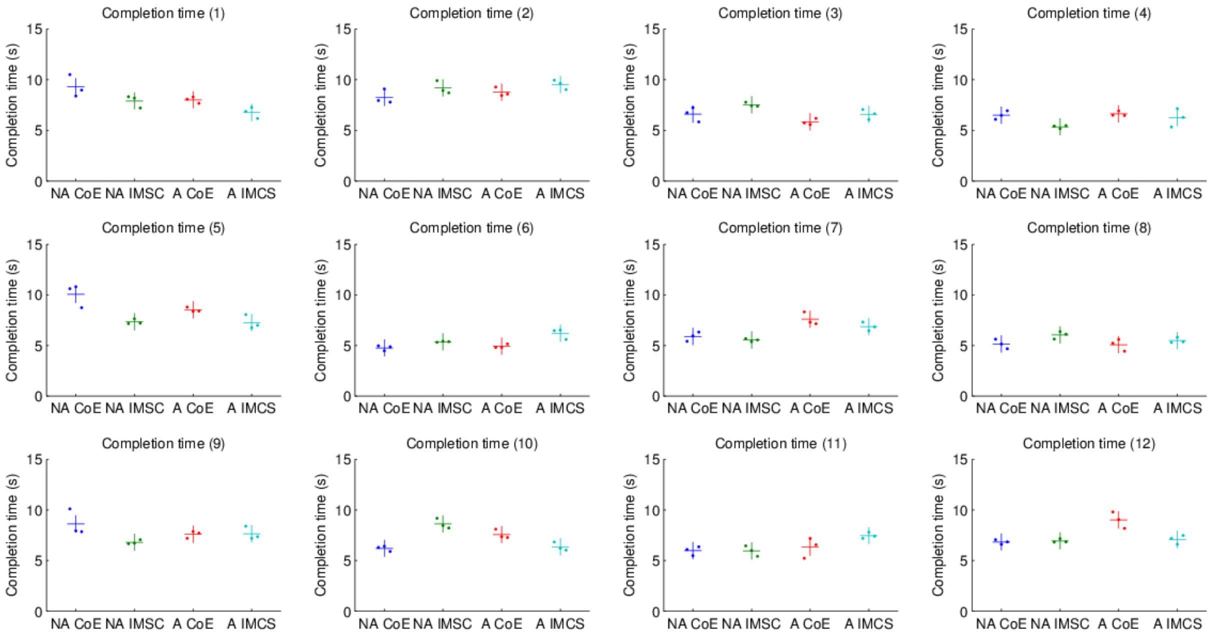

Fig. 7-3: Task Completion Time per Subject per Condition. The cross represent the average value for each condition. The dots represent the values for each repetition of condition (averaged over last rounds of each repetition).

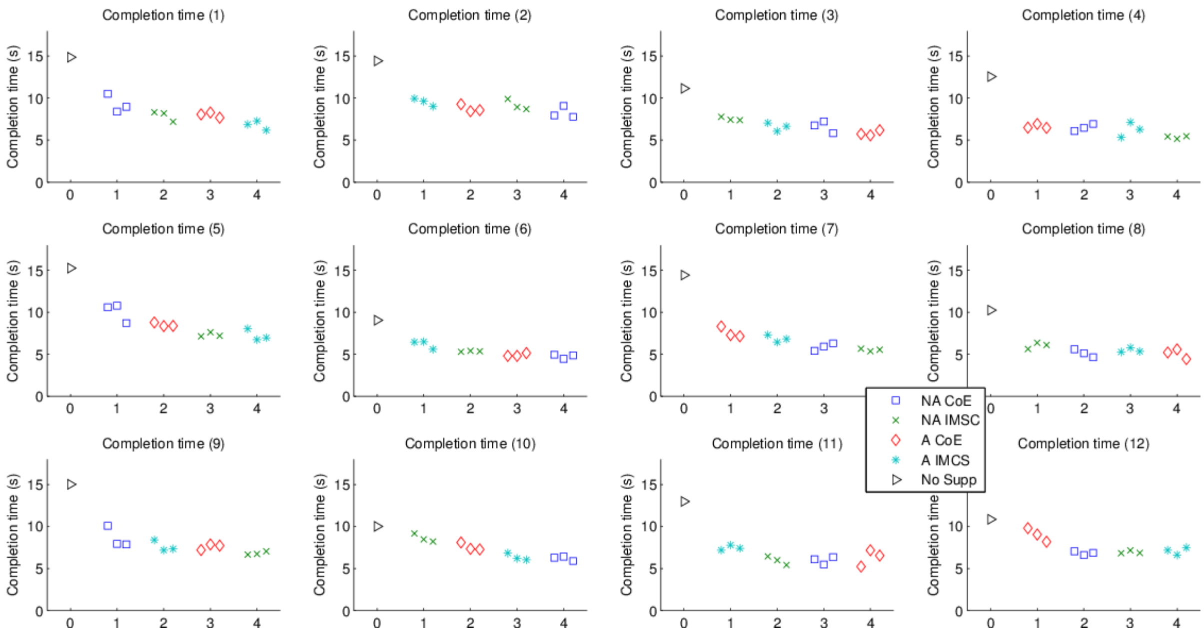

Fig. 7-4: Task Completion Time per subject per condition. The conditions are shown in order of execution. Per condition the three repetitions are shown in order of execution.

The number of errors is affected by the subject factor of the ANOVA. Fig. 7-5 shows the number of errors per individual subject.

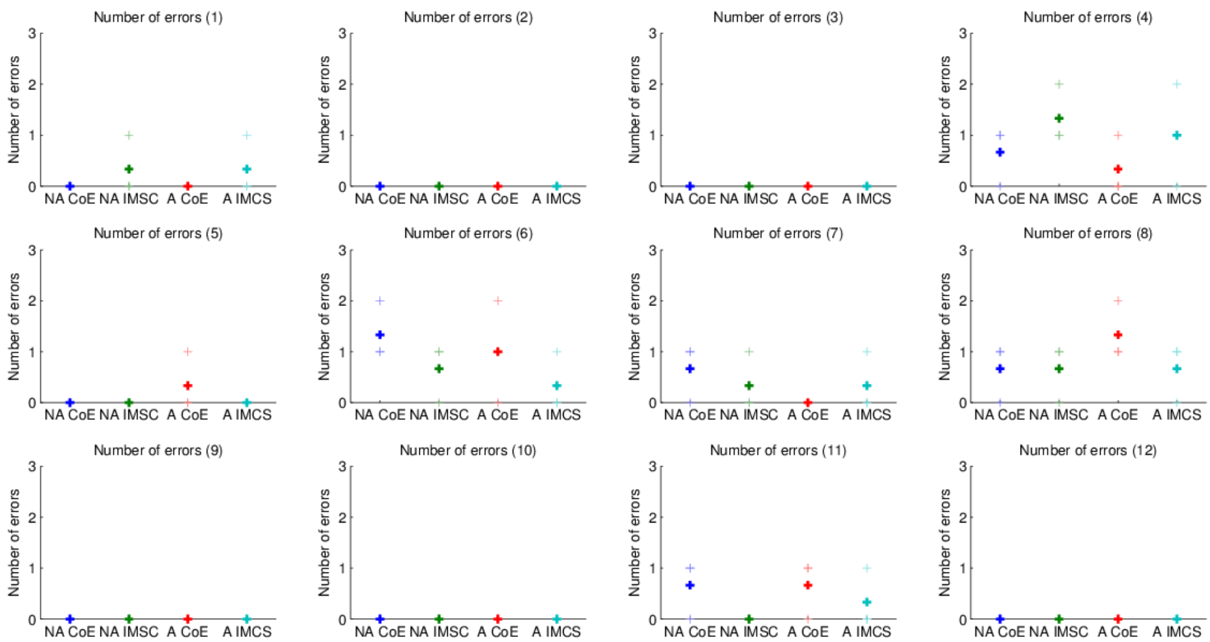

Fig. 7-5: The number of errors per condition as shown per subject.

Control effort metrics per subject

The following figures show the shared control forces averaged per subject. In Fig. 7-7 the shared control forces of every subject are shown in order of performance to evaluate the effect of the order. Fig. 7-8 shows the steering corrections for every individual, Fig. 7-9 shows this in the order of performance. No direct relation between the order of performance and the control efforts can be deduced from the pictures.

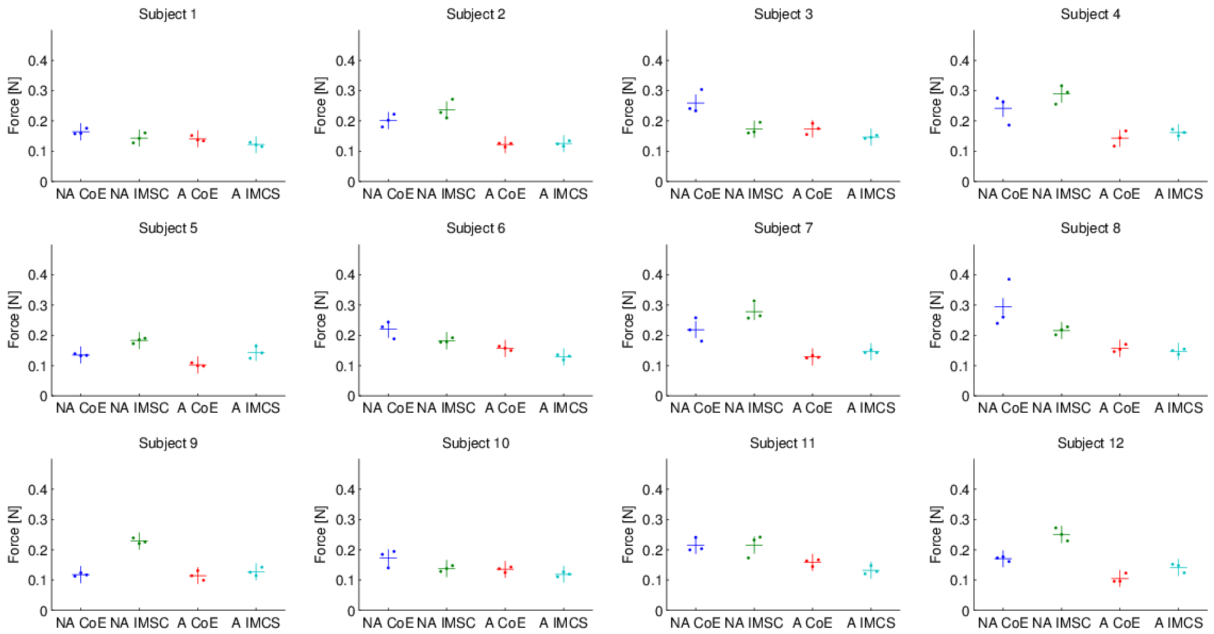

Fig. 7-6: Average Shared Control Force per Subject

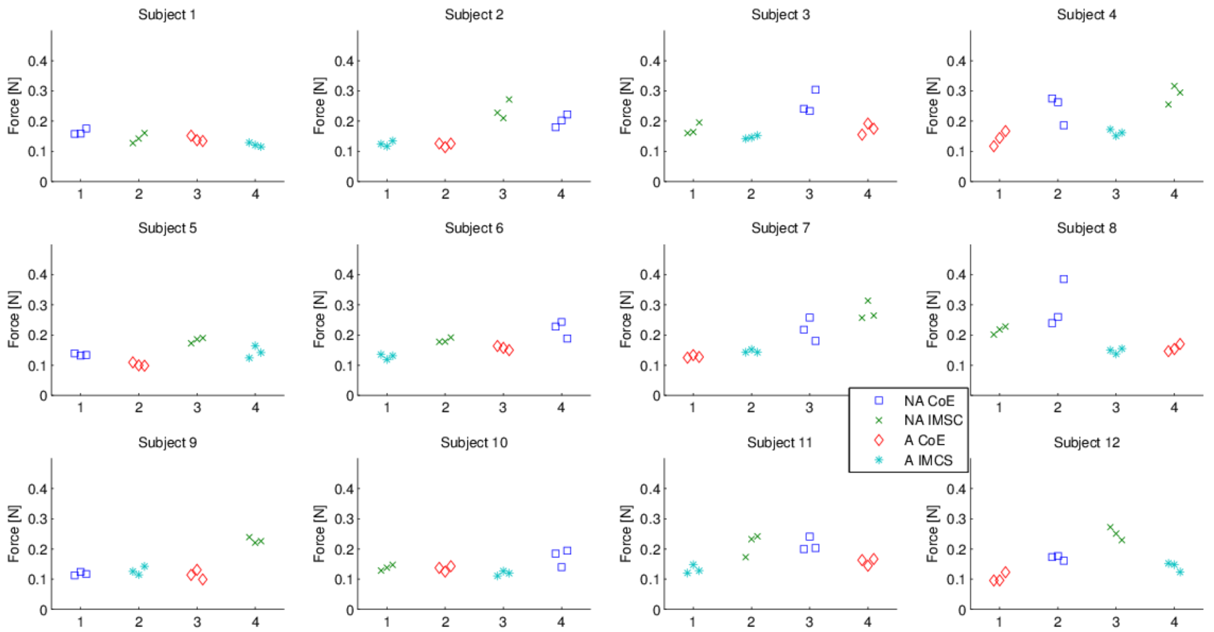

Fig. 7-7: Average Shared Control Force per Subject, in order of performance

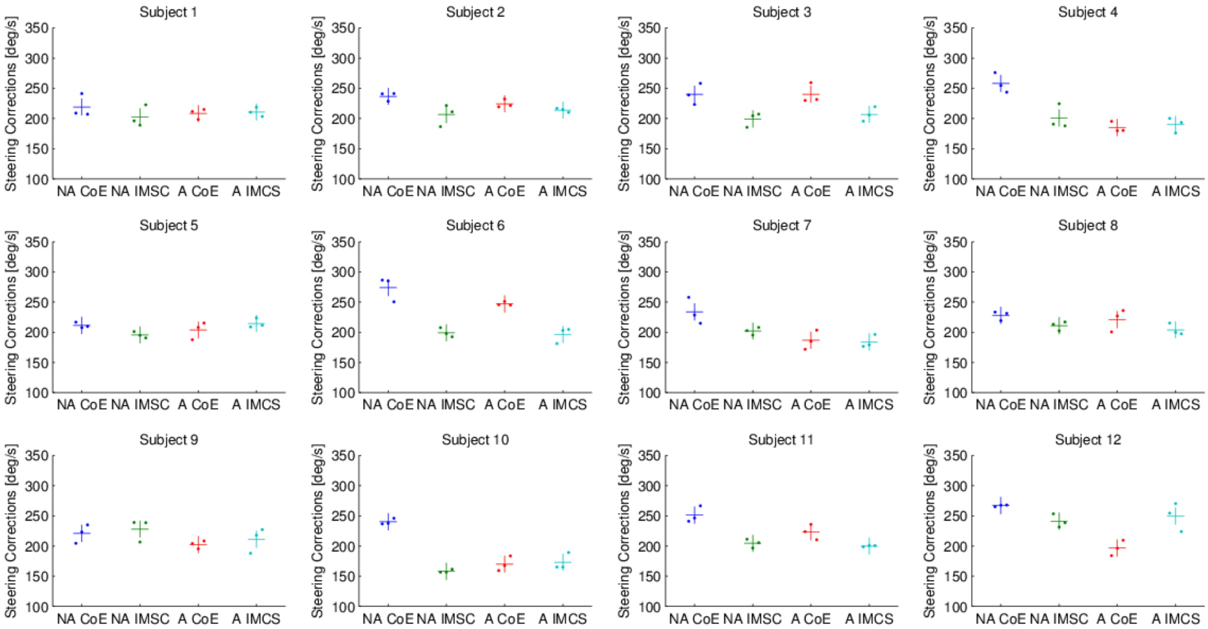

Fig. 7-8: Steering Corrections per Subject for every condition

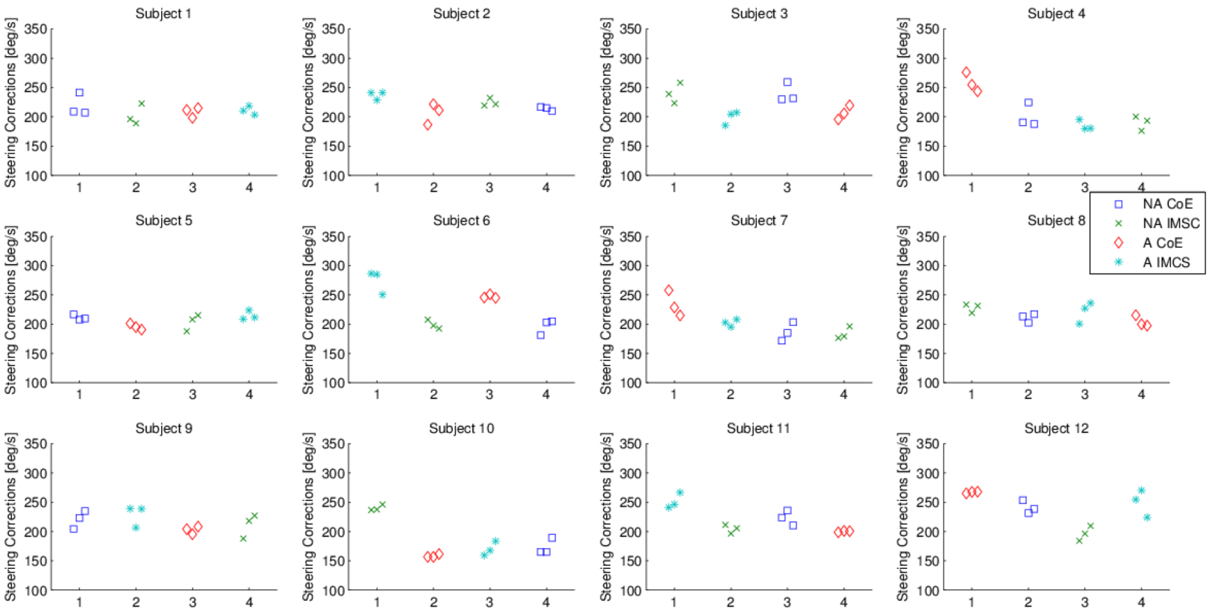

Fig. 7-9: Steering Corrections per Subject, for every condition in order of performance

Performed trajectories and support path

In this section, the performed trajectories and support paths for all conditions are presented of a selected set of subjects. The individual trajectories and support paths per sections are shown in more detail. Most notably there appears to be a substantial difference between the trajectories of individual operators.

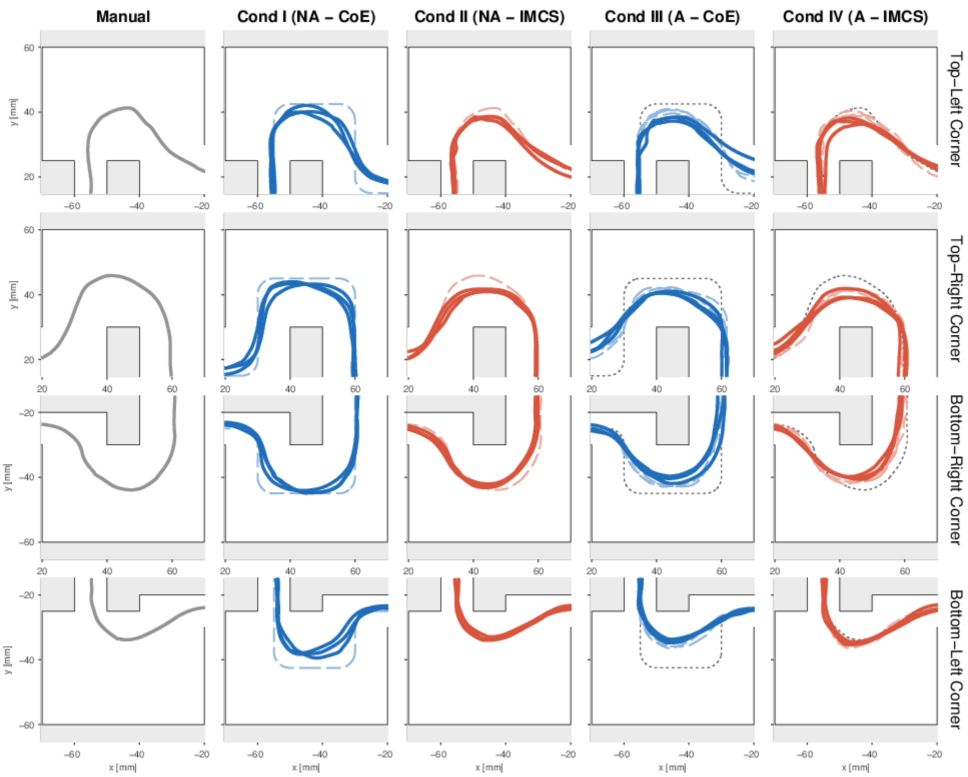

Fig. 7-10: The performed trajectories and support paths of subject 1. The three performed trajectories and the support paths are presented for every corner and every condition.

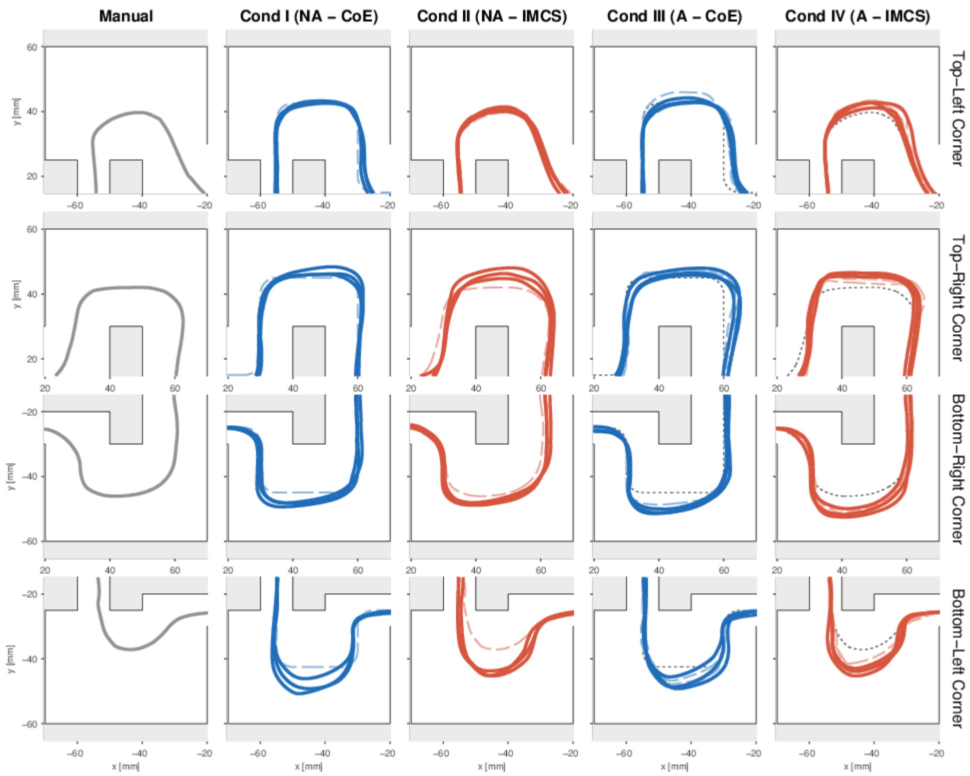

Fig. 7-11: The performed trajectories and support paths of subject 2. The three performed trajectories and the support paths are presented for every corner and every condition.

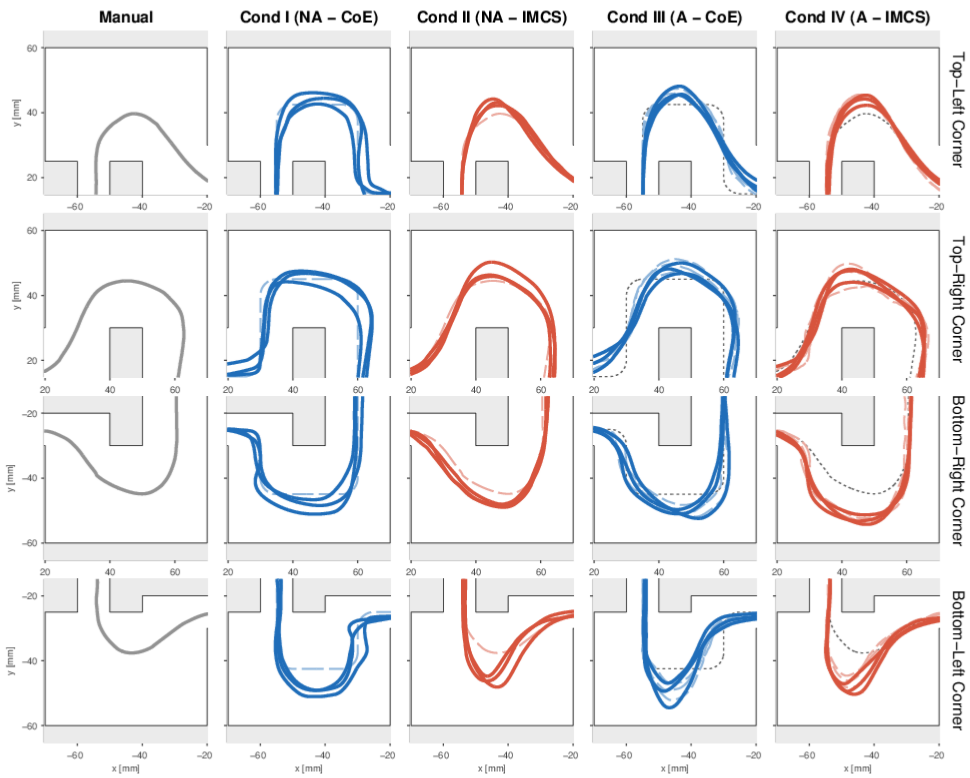

Fig. 7-12: The performed trajectories and support paths of subject 4. The three performed trajectories and the support paths are presented for every corner and every condition

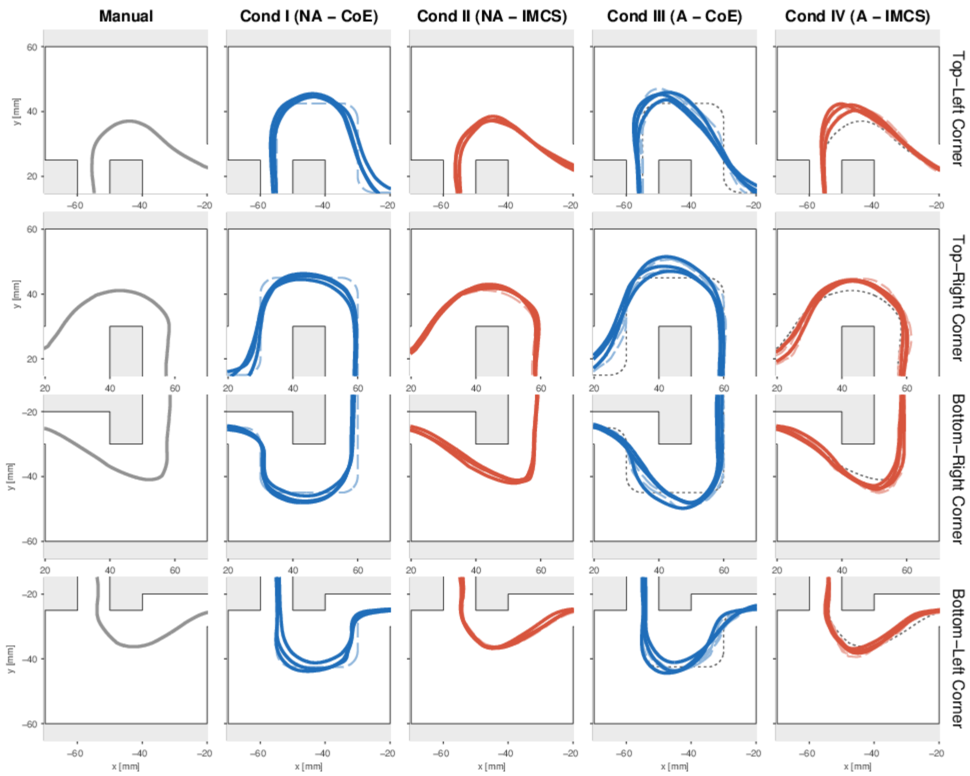

Fig. 7-13: The performed trajectories and support paths of subject 7. The three performed trajectories and the support paths are presented for every corner and every condition

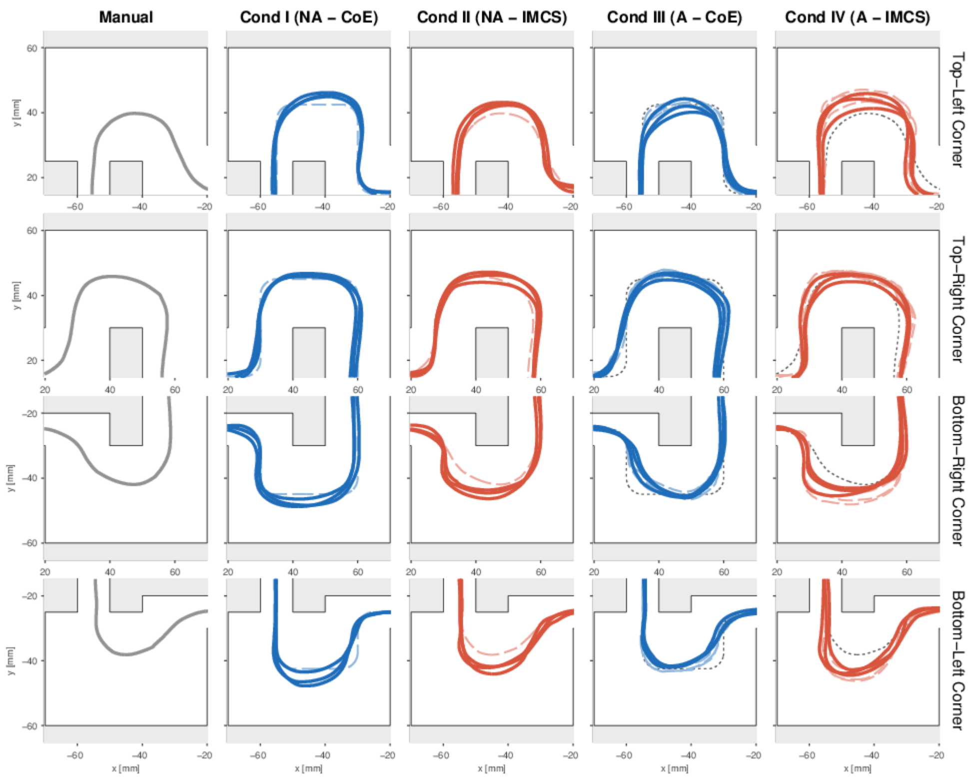

Fig. 7-14: The performed trajectories and support paths of subject 12. The three performed trajectories and the support paths are presented for every corner and every condition

Adaptation of support path

The average shared control path of every subject is presented in Fig. 7-15, to give some insight in individual adaptation. It shows that for some subjects the most subject the support paths are similar, but that some subject are more adapting to the system than vice versa. Most notable subject 12 seems to prefer a support path that is more like the centerline to environment than its own individual trajectory.

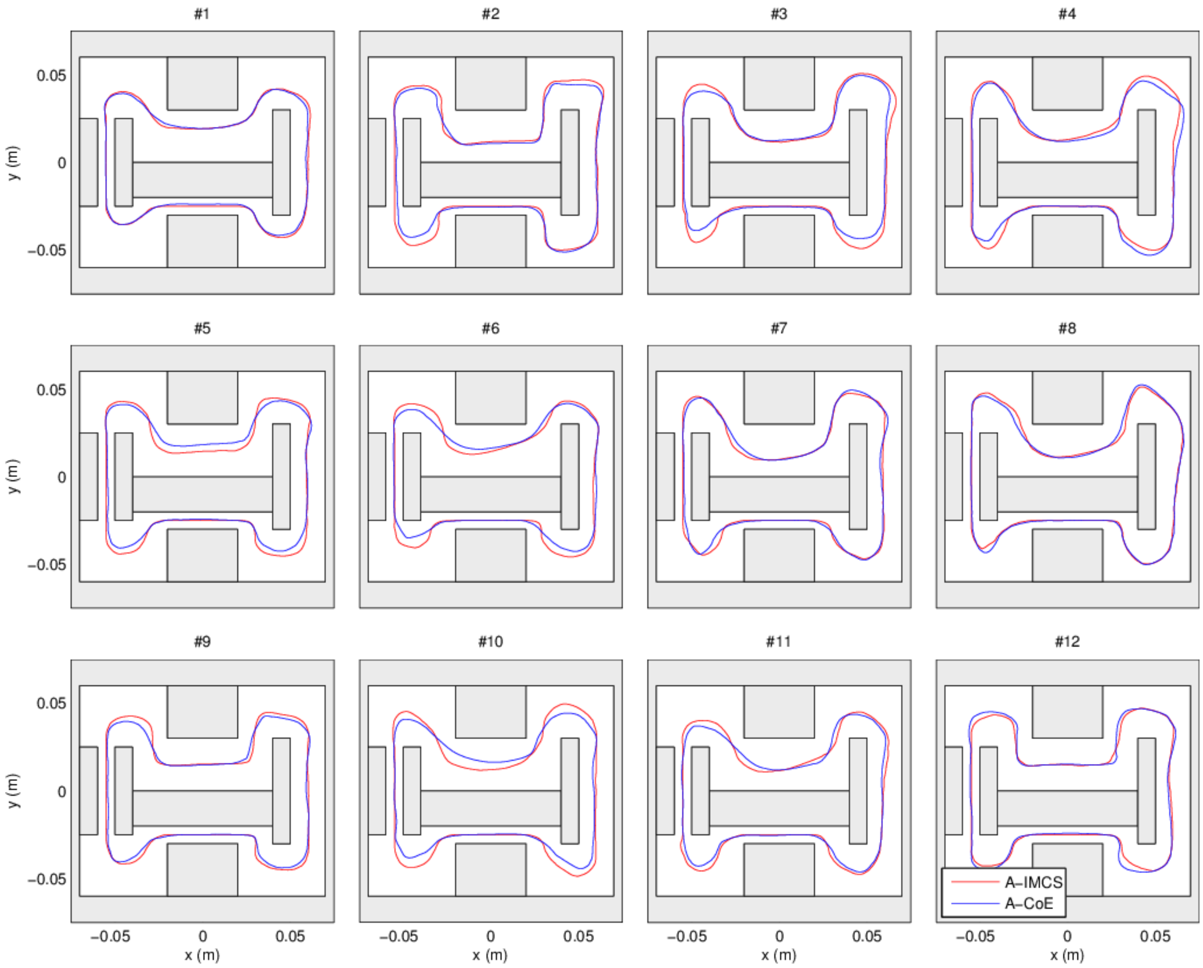

Fig. 7-15: Average Shared Control Path per Subject

It is hypothesized that after 15 rounds the support path has sufficiently adapted and that it remains in steady state support path and will not drift. Fig. 7-16 shows the path adaptation at every round. The difference between the support paths at every round should be zero ideally between round 15 and 20 indicating no more adaptation is present. The average value is around zero for the last 5 trials. Nevertheless, this value is averaged over the entire path and mirrored deviations cannot be seen.

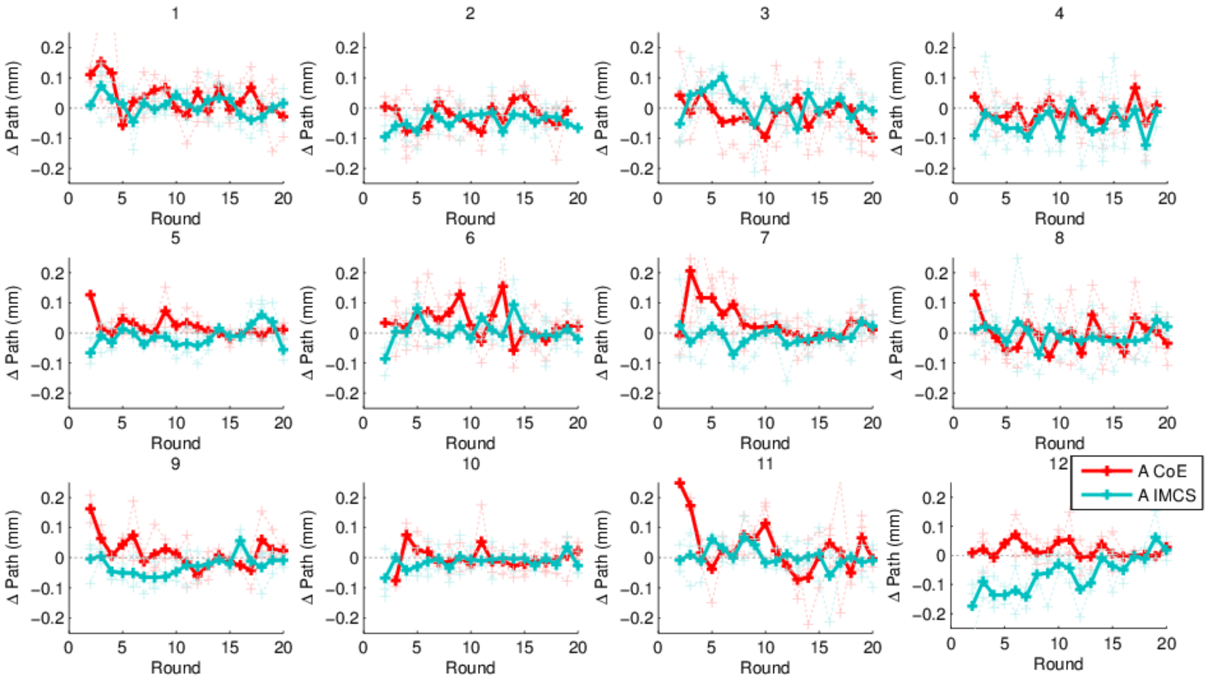

Fig. 7-16: Difference in path and path of previous round, integrated over the entire path and normalized to path length. Positive values indicate that the path is growing in outwards direction. The average value for the two adaptation conditions are shown, the path adaptation for every condition repetition are shown with high opacity.

In the following figures the differences between paths are shown for every segment of the environment. The segments are shown in their location in the environment (segment 1: left-straight, segment 2: top-left-corner, etc.). The differences in path are shown for a selected set of subjects in Fig. 7-17 up to Fig. 7-20. Although the ideal of no adaptation is not achieve in the last five runs, the adaptation in the last five runs appears to be minimal in most situation.

This particular implementation of adaptation seems to work acceptable for the overall task, but also when inspecting individual sections.

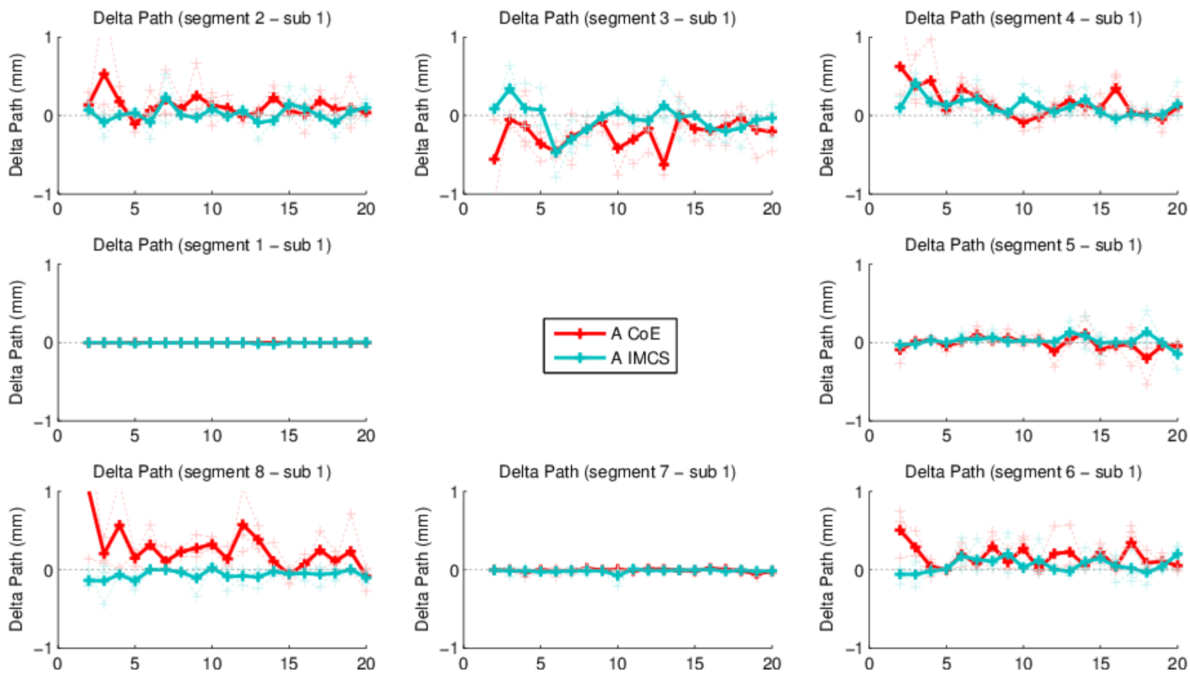

Fig. 7-17: Level of adaptation of support paths for subject 1, shown for the eight segments.

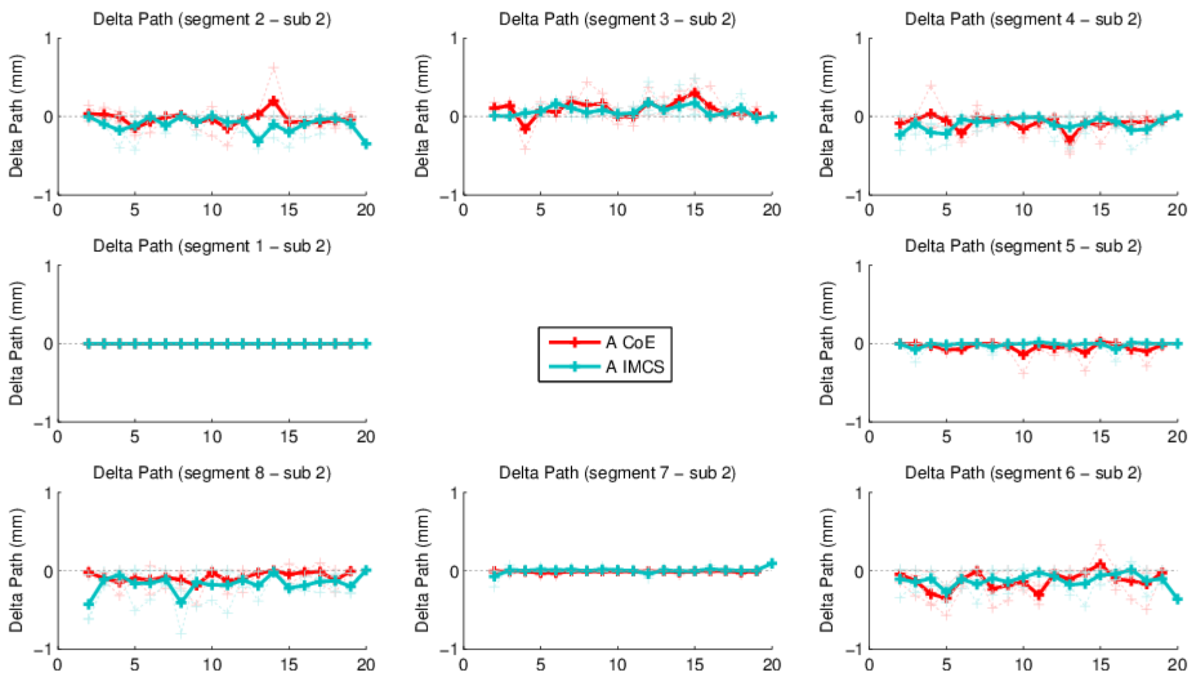

Fig. 7-18: Level of adaptation of support paths for subject 2, shown for the eight segments.

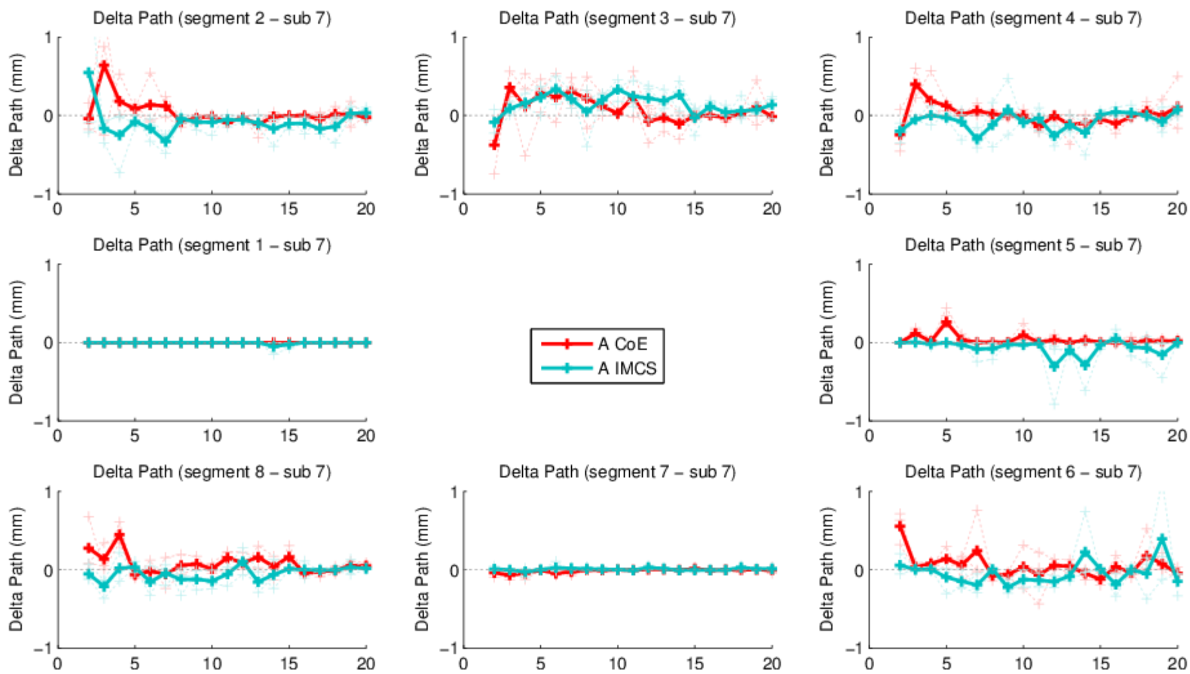

Fig. 7-19: Level of adaptation of support paths for subject 7, shown for the eight segments.

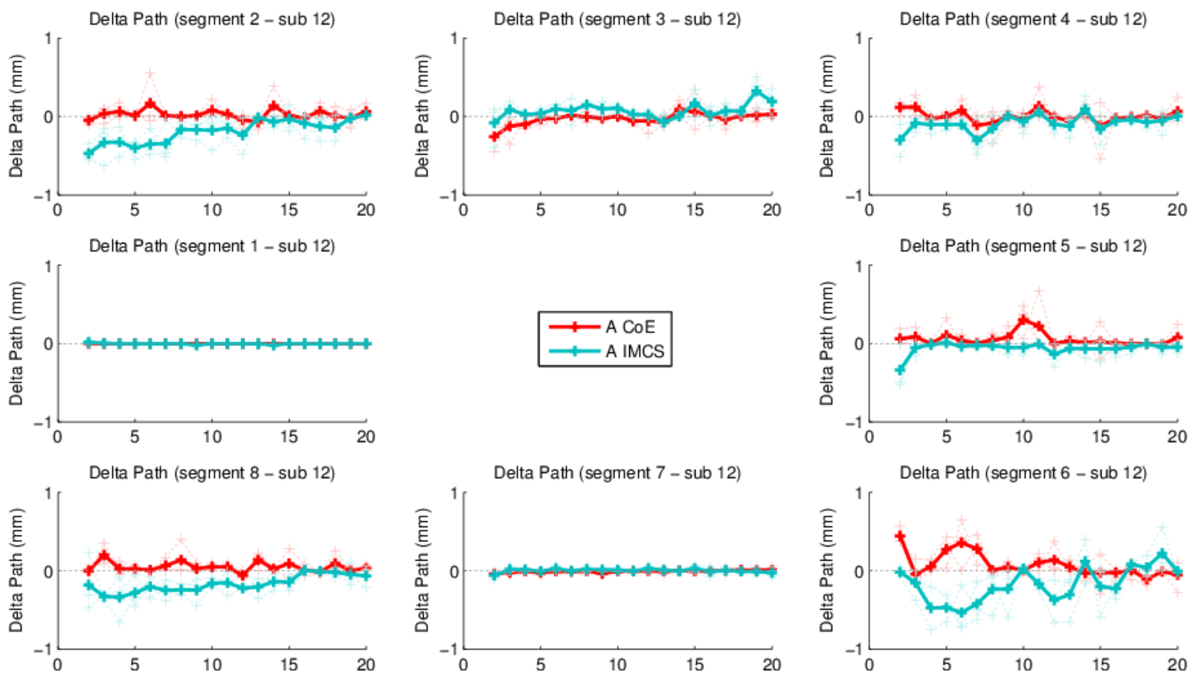

Fig. 7-20: Level of adaptation of support paths for subject 12, shown for the twelve segments.

Results per corner

The task environment was divided into eight separate sections, four corners and for straight sections. The straight sections are defined as the sections between the two constraining walls. Between the left and right part of the entrance and the left and right part of the exit the straight parts are defined. The remaining sections are defined as corners.

The time in section, the velocity, the shared control force and steering corrections are presented in the following figures. Comparing the four conditions the shared control force shows the overall effects (higher forces for non-adaptive conditions), although the bottom-left (slow) corner shows a distinction between the non-adaptive IMCS support and the CoE-support. The effects in steer corrections, measured over the entire task are also shown in the corner steer corrections. The average velocities show that the environment design yielded different velocities.

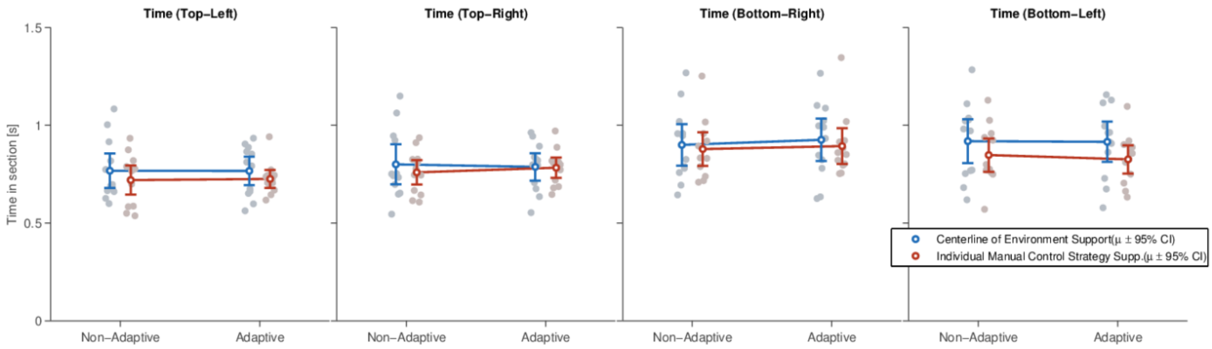

Fig. 7-21: Time in corner

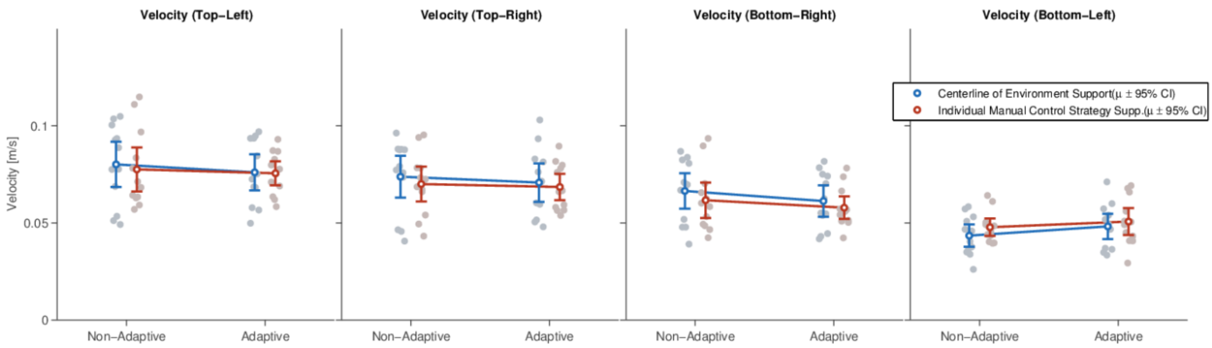

Fig. 7-22: Average velocity in corner

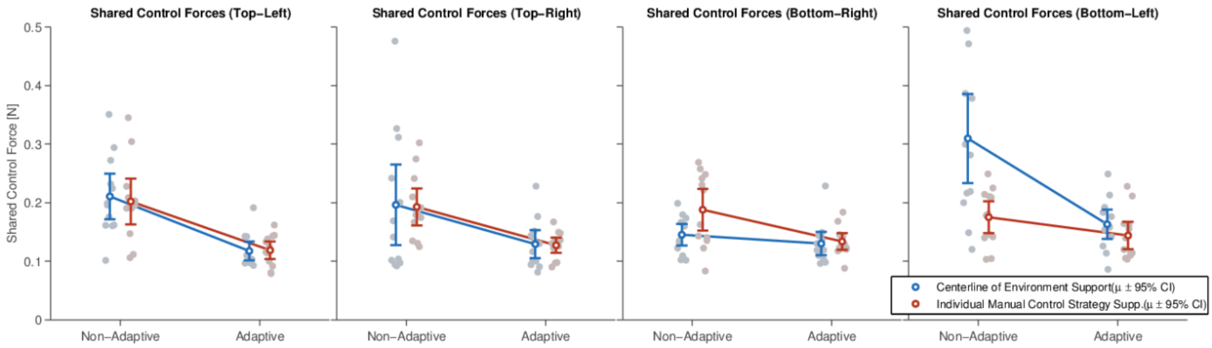

Fig. 7-23: Shared Control Force for individual corner

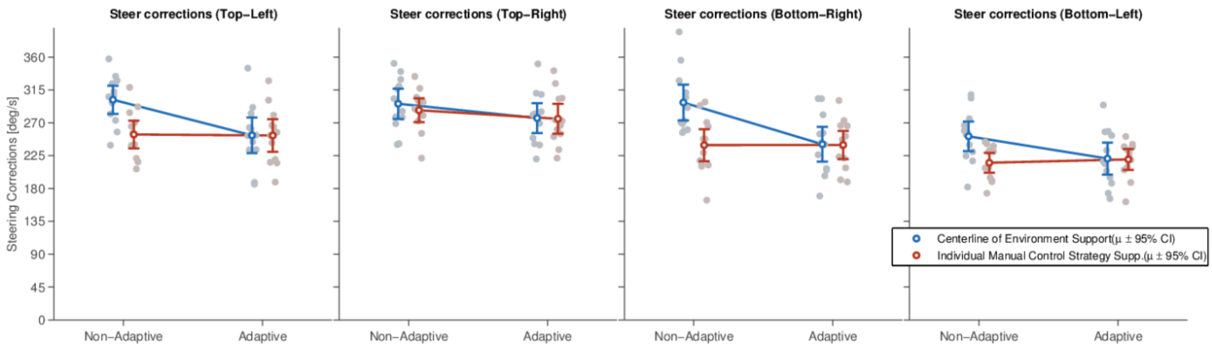

Fig. 7-24: Steering correction for each corner

Results per straight section

The time in each section, the velocity, the shared control force and the steering corrections are presented for each straight section of the environment. For individual straight sections the effects of adaptation and the support paths appears limited. Most notably in the right section of the environment the steering rate appears to be much lower than in the other conditions. In this section the operator has to pull the device towards him, which can be done apparently with less steering corrections. The effects on steering corrections as shown in the entire task performance are not strongly presented in the individual straight sections. Considering shared control forces, the major findings (decrease of shared control forces with adaption) appear to be present, although not very strong.

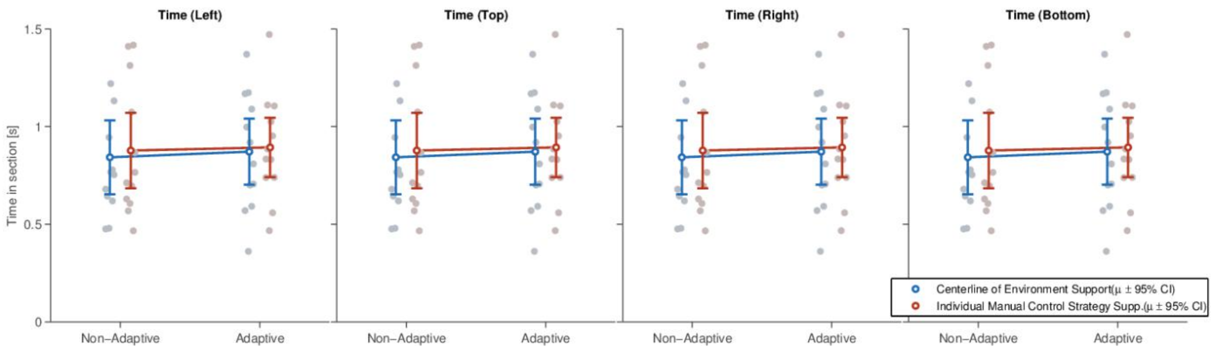

Fig. 7-25: Time in corner section

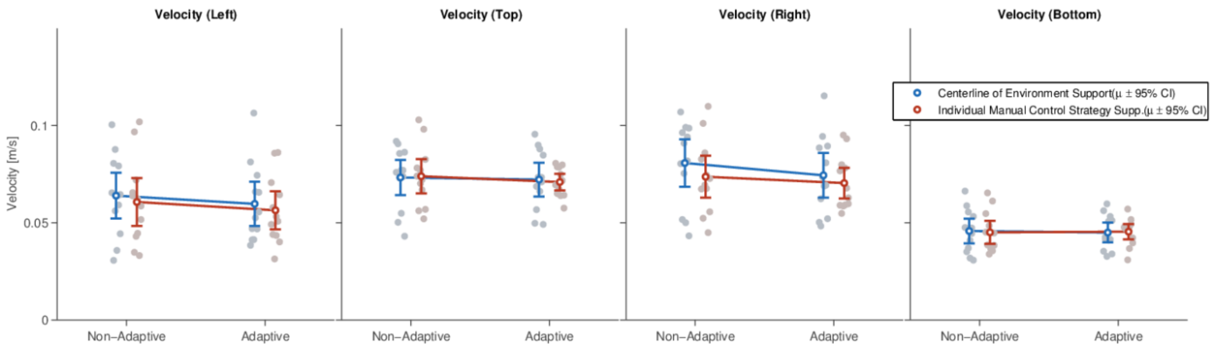

Fig. 7-26: Average Velocity in section

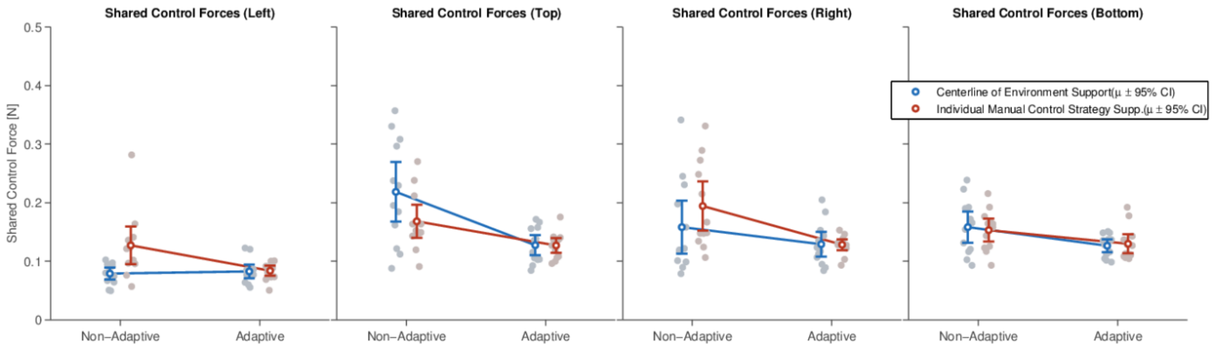

Fig. 7-27: Average Shared Control Force in section

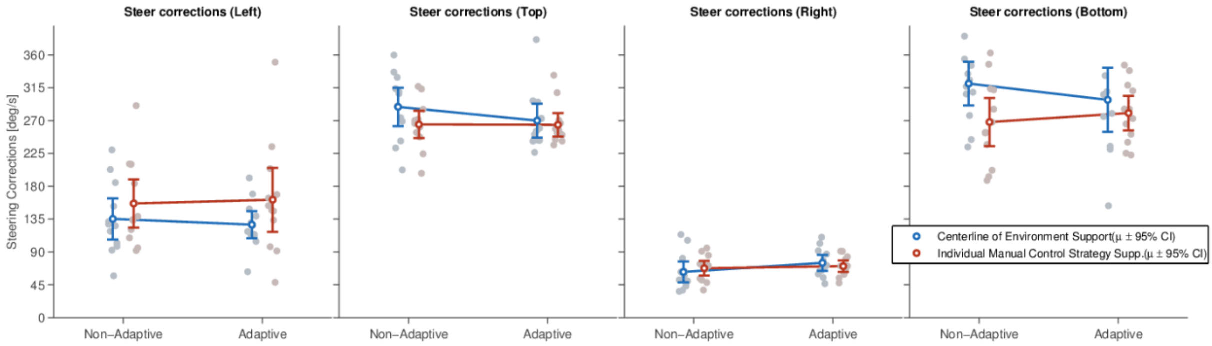

Fig. 7-28: Average Steer activity in corners

Appendix 8: Literature

Abbink, 2010	D. Abbink, M. Mulder, Neuromuscular Analysis as a Guideline in designing Shared Control, Advances in Haptics, chpt. 7, p. 499-516, 2010
Boessenkool, 2011	H. Boessenkool, "Thesis – Haptic shared control improves teleoperated task performance towards performance in direct control", 2011
Christiansson, 2007	G. Christiansson, "Hard master, soft slave haptic teleoperation", Ph. D. dissertation, 2007
Enache, 2009	N. M. Enache, Y. Sebsadji, S. Mammar, B. Lusetti, and S. Glaser, "Driver's influence on the performance of an integrated lane departure avoidance and lane keeping assistance system," in 2009 IEEE International Conference on Control Applications. IEEE, Jul. 2009, pp. 119–124
Griffiths, 2005	P. Griffiths, R. Gillespie, "Sharing control between humans and automation using haptic interface: primary and secondary task performance benefits.", Human Factors, 47-3, p.574-590, 2004
Hart, 1988	S. G. Hart and L. E. Staveland, "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research," Human mental workload, vol. 1, no. 11, pp. 139–183, 1988
Marayong, 2004	P. Marayong, A. Okamura, "Speed-Accuracy Characteristics of Human-Machine Cooperative Manipulation Using Virtual Fixtures With Variable Admittance", Human Factors: The Journal of the Human Factors and Ergonomics Society, 46-3, p. 518-532, 2004
Passenberg, 2010	Passenberg, C., Peer, A. & Buss, M. A survey of environment-, operator-, and task-adapted controllers for teleoperation systems. Mechatronics 20, 787–801 (2010)
Passenberg, 2011	C. Passenberg et al, "Towards real-time haptic assistance adaptation optimizing task performance and human effort", World Haptics Conference 2011, p 155-160, 2011
Rasmussen, 1983	J. Rasmussen, "Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models.", IEEE Transactions on Systems, Man, and Cybernetics, SMC-13-3, p 257-26, 1983
Wildenbeest, 2010	J.G.W. Wildenbeest, "Thesis – Improving the Quality of Haptic Feedback Yields Only Marginal Improvements in Teleoperation Task Performance", 2010