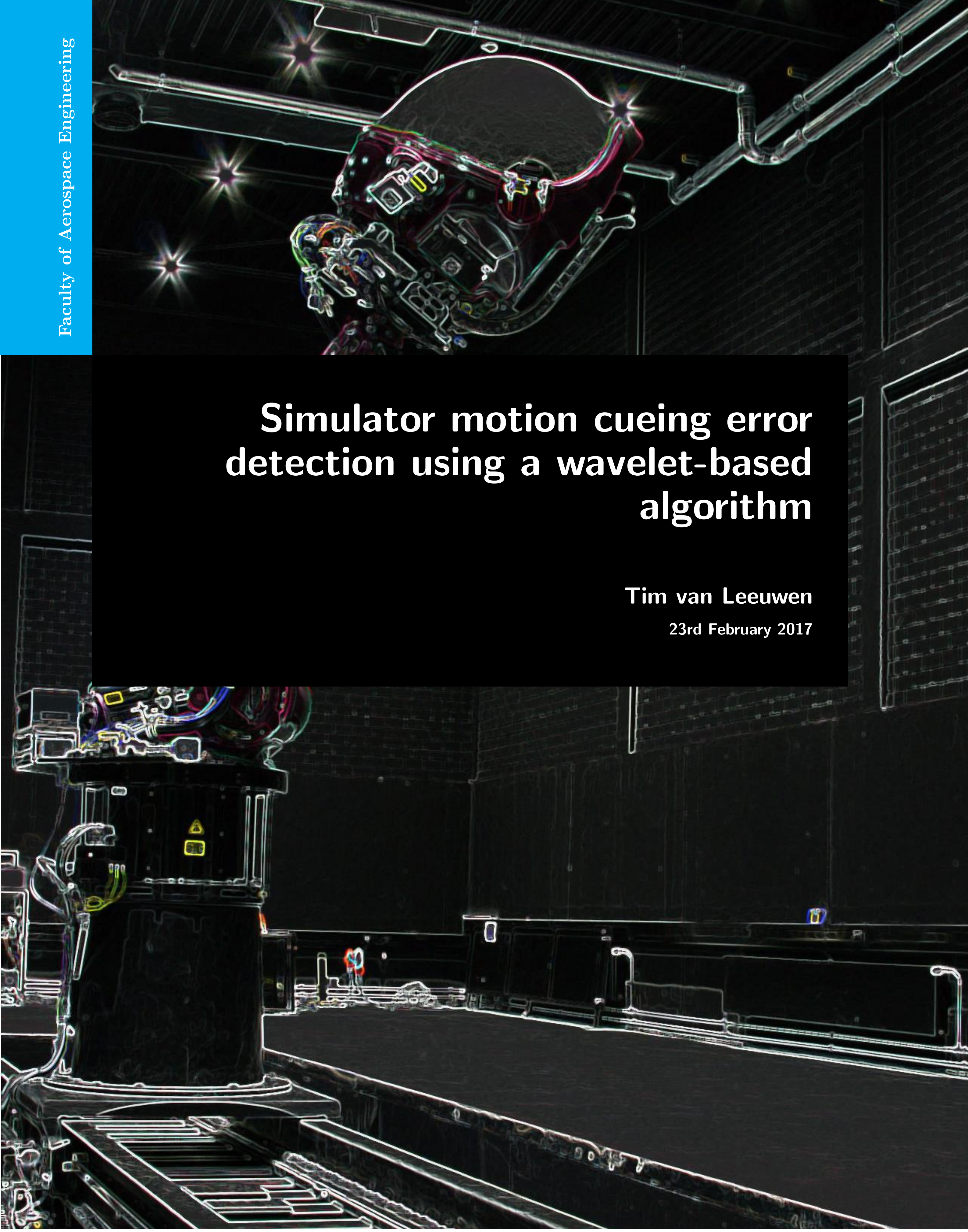
Faculty of Aerospace Engineering

Simulator motion cueing error detection using a wavelet-based algorithm

Tim van Leeuwen

23rd February 2017



Delft University of Technology

Challenge the future

Simulator motion cueing error detection using a wavelet-based algorithm

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

Tim van Leeuwen

23rd February 2017

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

Delft University Of Technology
Department Of
Control and Simulation

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **"Simulator motion cueing error detection using a wavelet-based algorithm"** by **Tim van Leeuwen** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 23rd February 2017

Readers:

prof.dr.ir. M. Mulder

ir. D. Cleij

dr.ir. D. M. Pool

dr. M. Mitici

Acknowledgements

For completion of this master's thesis research project I owe a great deal of thanks to many people. First and foremost I'd like to thank both my daily supervisors, Diane and Daan, for their ever-present willingness to help and their valuable feedback on whatever I'd thought of every time I spoke them. You've managed to balance reassuring comments with critical ones, and as such have definitely raised the bar for this research project. I'd like to thank Max too for his continuous interest in my (and not only mine, but all) master student research project(s). Your comments at our very first meeting (*"It's OK if nothing successful comes out of this project, then at least you've shown us where not to look"*) have reassured me that all actions taken in my project would be useful in some way or another, even if they did not show promising results.

I am also very grateful to the Max Planck Institute for Biological Cybernetics for hosting me during my research project, so I'd like to thank Mr. Bülthoff, and Joost and Paolo, for giving me a warm welcome when I first arrived in Germany. I should say thank you to all members of the Motion Perception & Simulation group for creating a very nice work environment, and special thanks go to Maria, Michael and Harald for learning me how to operate the CyberMotion Simulator and helping me out with all troubles I encountered there.

On a less work-related note, I must say *danke schön* to my German housemates Lily, Timo, Luca, Eva, Paul, Julius, Laura and Emilia for contributing massively to my wonderful time in Tübingen, *es war super*!

My final thanks go to the people closest to me. I could thank my parents for many things, but them giving me the freedom to choose the study and career path I desire(d) is something I'm particularly grateful for. And finally, thank you Lotte, for supporting this aerospace engineer as he spread his wings and flew out to countries far away, but also for keeping him with his feet on the ground.

Tim
10 February 2017

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Motion simulators are used frequently for both vehicle operator training and research into human perception (Reid & Nahon, 1986). These devices aim to reproduce the sensation of motion a vehicle operator would experience in a real vehicle. This replication is done through a combination of visual, vestibular, somatosensory and auditory cues (Fischer, 2007). In general, however, the exact replication of a vehicle's physical motion is not possible, since motion simulators have limited motion envelopes (Reid & Nahon, 1986).

To ensure a motion simulator remains within its motion envelope, a Motion Cueing Algorithm (MCA) is used to map vehicle inertial motion to simulator inertial motion (Stahl, Abdulsamad, Leimbach & Vershinin, 2014). One of the first (and, to this day, most commonly implemented) MCAs is the classical washout filter (CWF), as introduced by Schmidt & Conrad (Schmidt & Conrad, 1970) and explained by Reid & Nahon (Reid & Nahon, 1986). Since its introduction various improved versions of this algorithm, using adaptive and optimal control techniques, have been introduced (Salisbury & Limebeer, 2016). Also, novel motion cueing techniques based on human perception models have been proposed (Venrooij et al., 2015). All these techniques feature parameters which can be tuned to improve an MCA's performance for a given motion simulator and simulation scenario.

Several frameworks have been developed to tune MCA performance. Schroeder (Schroeder, 1999) developed an objective method for tuning MCA parameters based on the motion fidelity criterion established by Sinacori (Sinacori, 1977), which specifies acceptable gains and phase delays for 1 rad/s signals. A similar criterion was proposed by Advani & Hosman (Advani & Hosman, 2006), based on vehicle natural frequencies.

Research by Valente Pais et al. (Valente Pais, van Paassen & Mulder, 2009), however, showed that human perception of motion cueing errors changes with the frequency of the motion cueing signals. This suggests that basing an MCA tuning criterion on a single frequency is not correct. Additionally, a recent review by Nash, Cole & Bigler (Nash, Cole & Bigler, 2016) shows human perception thresholds to vary with signal frequency as well. A subjective tuning method for classical washout filters based on pilot evaluations was proposed by Grant (Grant, 1995). This paradigm, however, is only valid for a single type of MCA. Also, subjective tuning of an MCA will likely lead to different outcomes in case the tuning process is repeated with different evaluation pilots (Hosman & Advani, 2016).

Research by Cleij et al. (Cleij et al., n.d.) showed that subjects' perception of the realism of a motion simulation depends both on the magnitude and type of motion cueing errors (mismatches between the presented visual motion and experienced simulator motion). Motion cueing algorithms could thus be improved by focusing on removing the worst types of errors from a simulation.

To do so, it should be investigated what errors are the most important in determining a human's perception of a motion simulation. Additionally, it should be possible to objectively identify both the time of occurrence and type of motion cueing error throughout a simulation. To make this approach suitable for all MCAs, no knowledge of the applied MCA should be used in this identification process.

As of now, however, there is no way of objectively identifying the presence of various motion cueing error types in time histories of a simulation, other than by (subjective) visual inspection. With the aim of improving the performance of motion simulators in the future, it is worth investigating whether a breakdown in motion cueing error types can be made, and whether these can be objectively identified in motion simulation traces.

1-1 Research goal

Motion cueing algorithms could be improved by focusing tuning efforts on the worst parts of a simulation. To do so, it should be possible to (objectively) identify different types of motion cueing error in a motion simulation. Additionally, it should be known which of these error types is perceived to be worst by humans.

This leads to three goals for this research:

1) To define a breakdown of motion cueing error types.
2) To develop a methodology which identifies the error types defined in 1) in a motion simulation input and output trace.
3) To investigate which error types defined in 1) are perceived to be worst by humans during a motion simulation.

1-2 Methodology

Research goal 1) was fulfilled through a literature study, the results of which were presented in a Preliminary Thesis. In this Preliminary Thesis, research goal 2) was also addressed for idealised motion cueing errors. Using the results of this initial research, the methodology developed in the Preliminary Thesis was extended to real simulator traces. The final algorithm developed as a result of this process was explained in an IEEE paper. The experiment carried out to answer research goal 3) is also discussed in this paper.

The motion cueing error detection algorithm described in both the Preliminary Thesis and IEEE paper was developed using Matlab. For implementation of wavelet analysis, Matlab's wavelet toolbox was used. Several modifications to this toolbox have been made, these changes are described in both the Preliminary Thesis and IEEE paper. The experiment carried out to validate this algorithm was run in the CyberMotion Simulator (CMS) of the Max Planck Institute for Biological Cybernetics. The analysis of data from this experiment was done using Matlab, with statistical tests being performed in SPSS.

For the Preliminary Thesis, a literature survey was carried out. This literature survey focused on three main topics: Motion cueing error types, human perception and wavelet analysis. Literature was collected from various online databases, including Scopus, Google Scholar, AIAA, IEEE Xplore and Wiley online library. Search terms included (combinations of) the following keywords:
Motion simulation, motion cueing algorithm, motion cueing (errors), perception, thresholds, coherence zones, error detection, algorithm, wavelet analysis, coherence

1-3 Report structure

This report is divided in two parts. In part I, the IEEE paper summarising the work performed in this research project is presented. This paper is followed by a set of appendices, which provide extra data supporting the research presented in the IEEE paper. In part II of this report the Preliminary Thesis, written prior to executing the experiment described in the IEEE paper, is presented. In this Preliminary Thesis relevant literature for the research is discussed, and a first version of the motion cueing error detection algorithm is presented.

Part I

IEEE Paper: Simulator motion cueing error detection using a wavelet-based algorithm

Simulator Motion Cueing Error Detection Using a Wavelet-based Algorithm

Tim D. van Leeuwen, Diane Cleij, Daan M. Pool, *Member, IEEE,* Heinrich H. Bülthoff, Max Mulder

***Abstract*—Understanding the process of human motion perception is essential for optimizing the motion cueing algorithms (MCAs) used by motion simulators. Previous research has shown that different motion cueing error types can be distinguished, at least some of which are rated differently by humans for equal error magnitude. Optimizing a motion simulation for both error magnitude and error type requires a method to objectively identify the presence of different motion cueing error types from simulation input and output traces. This paper presents a wavelet-based motion cueing error detection algorithm that exploits the time and frequency characteristics of these error types. A simulator experiment is presented through which parameters of the algorithm are determined. It is shown that motion cueing errors can be detected by the algorithm without prior knowledge of the MCA used. The need for the algorithm is demonstrated by showing that motion cueing error types can be divided in clusters of severity. Additionally, it is shown that in curve driving, given a reference of unity gain, the lateral specific force can be scaled down to 70% and up to at least 130% before subjects indicate motion to be too weak or too strong. A required input and output coherence of 0.7 is found for simulator input and output to be considered coherent. The motion cueing error detection algorithm can be used to improve motion cueing algorithms in the future.**

***Index Terms*—Motion simulation, motion cueing, motion perception, error detection, algorithm, wavelets**

I. INTRODUCTION

MOTION simulators are used extensively for vehicle operator training and research into human perception [1]. By combining visual, vestibular, auditory and somatosensory cues these devices try to replicate the sensation of motion an operator would have in an actual vehicle [2]. The exact replication of a vehicle's physical motion is generally not possible, however, as motion simulators have limited motion envelopes [1].

Mapping vehicle motion to the simulator motion space is done using a motion cueing algorithm (MCA) [3]. Various types exist, amongst others the classical washout filter as introduced by Schmidt & Conrad [4] and described by Reid & Nahon [1], improved versions of this algorithm using adaptive and optimal control techniques [5], and novel techniques based on human perception models [6]. All these techniques feature parameters that can be tuned to optimize an MCA's performance for a given simulator and simulation scenario.

Several frameworks have been developed previously for MCA tuning. Schroeder [7] developed an objective method of tuning MCA parameters based on the motion fidelity criterion established by Sinacori [8], which specifies acceptable gains and phase delays for 1 rad/s signals. A similar criterion was proposed by Advani & Hosman [9], based on vehicle natural frequencies. Research by Valente Pais *et al.* [10], however, showed that human perception of motion cueing errors changes with the frequency of the motion cueing signals. This suggests that basing an MCA tuning criterion on a single frequency is not correct. Additionally, a recent review by Nash, Cole & Bigler [11] shows human perception thresholds to vary with signal frequency as well. A subjective tuning method for classical washout filters based on pilot evaluations was proposed by Grant [12]. This paradigm, however, is only valid for a single type of MCA. Also, subjective tuning of an MCA will likely lead to different outcomes in case the tuning process is repeated with different evaluation pilots [13].

Research by Cleij *et al.* [14] showed that subjects' perception of the realism of a motion simulation depends both on the magnitude and type of motion cueing errors (mismatches between the presented visual motion and experienced simulator motion). Motion cueing algorithms could thus be improved by focusing on removing the worst types of errors from a simulation. To do so, it should be investigated what errors are the most important in determining a human's perception of a motion simulation. Additionally, it should be possible to objectively identify both the *time of occurrence* and *type of motion cueing error* throughout a simulation. To make this approach suitable for all MCAs, no knowledge of the applied MCA should be used in this identification process.

As of now, however, there is no way of objectively identifying the presence of various motion cueing error types in time histories of a simulation, other than by (subjective) visual inspection. The goal of this research therefore is to first define categories of motion cueing errors. Next, an algorithm is designed which is capable of detecting these types of errors in input/output time traces of a simulation. This is done using five characteristics: error frequency (shown to influence error severity by Valente Pais *et al.* [10]), error duration, the output over input signal ratio, input and output coherence and phase difference (all shown to influence error severity by Casas *et al.* [15]). Since both time and frequency characteristics of the signal are considered, the spatio-temporal analysis method of wavelet analysis is used. All characteristics are determined without knowledge of the MCA used. Various parameters (e.g., the minimum and maximum allowable scaling factors for motion, and human perception threshold levels) are included

T.D. van Leeuwen, D.M. Pool and M. Mulder are with the section Control and Simulation, Faculty of Aerospace Engineering, Delft University of Technology, 2629 HS Delft, The Netherlands. Corresponding author: t.d.vanleeuwen@student.tudelft.nl.

D. Cleij and H.H. Bülthoff are with the Max Planck Institute for Biological Cybernetics, 72076 Tübingen, Germany.

Manuscript received Februari 23, 2017; revised ????, 2017.

in the algorithm. These can be varied to tune the sensitivity of the algorithm.

To validate the motion cueing error detection algorithm, an experiment was performed in the CyberMotion Simulator of the Max Planck Institute for Biological Cybernetics. In this experiment subjects passively experienced a car drive through identical curves, with all curves featuring a modified lateral specific force. Subjects were asked to continuously rate their perceived visual-motion mismatch using the continuous rating method explained by Cleij *et al.* [16]. These ratings allow for commenting on what types of motion cueing errors are the most detrimental to the realism of a simulation, and allow for finding suitable parameter values for the error detection algorithm.

This paper is structured as follows: first the categorization of motion cueing errors is discussed in Section II. Since both time- and frequency-domain information of these errors is taken into account, the spatio-temporal analysis method of wavelets is investigated in Section III. The design of the motion cueing error detection algorithm is then discussed in Section IV. Following this, the set-up of the simulator experiment to tune and validate the algorithm is discussed in Section V, with the results of this experiment being presented in Section VI. A discussion of these results is presented in Section VII, with conclusions being drawn from this paper in Section VIII.

II. MOTION CUEING ERROR CLASSIFICATION

Research by Cleij *et al.* [14] has shown that not all motion cueing errors of equal magnitude are rated as being equally detrimental to motion simulator fidelity. This observation was already made by Grant *et al.* [17] and Fischer [18], who proposed a distinction between scale and shape errors after observing that downscaling a simulator input signal has a different effect on the perceived motion fidelity than distorting this motion but keeping a unity gain.

This finding was confirmed by Casas *et al.* [15] in a study into a more elaborate breakdown of motion cueing errors. They concluded that the coherence between a simulator's input and output signal (i.e., how similar the input and output signal are in their shape/appearance) mainly determines a subject's perception of the simulation (a correlation value of 0.4-0.5 was found). In addition, [15] shows the absolute average scale and estimated delay indicators to correlate (correlation values of 0.2-0.3) with humans' perception of a simulation.

All indicators used in these previous studies, however, were computed using the *entire* simulator input and output signal. This means all time resolution is lost when using these indicators, as they produce just one value for the entire simulation. This property is undesirable given that the motion cueing error detection algorithm should indicate errors over time, such that tuning activities can be focused on the worst sections in a simulation. Such a tuning approach is expected to show improvements over the current approaches suggested by Schroeder [7] and Grant [12], which tune MCAs based on an algorithm's performance over a whole simulation. The error breakdown method proposed by Grant *et al.* [17] and Fischer [18] implicitly assumes constant input scaling to be applied, which limits its applicability to motion cueing algorithms for which this scaling is constant and known.

It is for these reasons that a new method of analysis, which requires no a priori knowledge, will be presented in this paper. The method is based on a motion cueing error type breakdown derived from the comprehensive overview of Grant & Reid [19]. Grant & Reid distinguish three types of motion cueing error: *false cues*, *scaling errors* and *phase errors*. This distinction will lead to three error categories in this paper.

Objective detection of the different motion cueing error types will be done based on five mathematical characteristics of these errors. As shown by Valente Pais *et al.* [10], error frequency influences the perceived severity of a motion cueing error. Hence, this frequency is investigated. Since errors should be localized in time, error duration is also discussed. As an extension to the results of Casas *et al.* discussed previously [15], the time-varying input and output signal coherence (a measure of signal similarity, referred to as the I/O coherence) will be investigated. Additionally, a time-varying version of the absolute average scale and estimated delay indicators of Casas *et al.* is also investigated. The phase difference between the output and input signal is measured in degrees. The time-varying output over input ratio at time t, referred to as the Power Ratio or PR from now on, is defined as:

$$\mathrm{PR}(t) = \frac{\mathrm{Output}(t)}{\mathrm{Input}(t)} \qquad (1)$$

In the following, error types and their properties are described. Examples of error causes are given for classical washout filters, since these are most common [5]. However, note that most error causes (e.g., washout, tilt-coordination and motion scaling) also occur in non filter-based algorithms [20].

False cues - Grant & Reid [19] note that false cues are the most detrimental to motion simulator fidelity. Three different types can be distinguished:

- *Type I false cue (F1)* - F1 errors are typified by the simulator output being of opposite sign to the simulator input. Typical causes for such an error are simulator washout and braking near the end of the motion envelope. Additionally, these errors are caused by high-pass filters at the end of a sustained cue. Since the input and output are of opposite sign the Power Ratio is negative and I/O coherence is low. F1 errors typically last in the order of seconds and are expected at mid-to-low frequencies since they originate from simulator washout, which operates at these frequencies.
- *Type II false cue (F2)* - F2 errors are described by Grant & Reid as *"a motion cue in the simulator when none was expected"* [19]. These errors are mostly caused through the tilt coordination process. This process introduces false rotational rates into a simulation. To limit the effect of these false rotational rates, the tilt rate is often limited [21]. However, due to this limited tilt rate, it takes time to return a tilted simulator back to its neutral position. This prolonged tilt introduces false specific forces. The latter

cause is referred to as the tilt coordination remnant [19]. Since the output is (much) larger than the input, the PR will be larger than 1 and I/O coherence will be low. Again these errors are expected at mid-to-low frequencies since they are often the result of a motion cueing algorithm's low-pass filters.

Since motion simulators have a limited motion envelope, deliberate exaggeration of motion is not expected. For this reason, exaggerated motion is identified as an F2 error too. This motion could occur at any frequency and for long durations. Should this happen, then the I/O coherence remains high, the PR is larger than 1 and there is no phase difference between input and output.

- *Type III false cue (F3)* - Unexpected cues of high frequency are designated Type III false cues. A distinction is made in this paper between short-duration F3a errors (e.g., caused by a simulator hitting a displacement limit) and long-duration F3b errors (e.g., caused by simulator noise). For both error types the I/O coherence is low and the PR can take on any value, as these errors can occur in combination with any other error.

Scaling errors - The second category of motion cueing errors is scaling errors. Research has shown that these errors are not detrimental per se [22], [23], meaning scaling does not have to be one to one. Even severe scaling errors usually lead to lower perceived mismatches than false cues [14]. Two types of scaling error can be distinguished, with the following mathematical properties:

- *Type I scaling error (S1)* - S1 errors are pure scale errors as defined by Grant *et al.* [17] and Fischer [18]. They originate from a constant gain for all frequencies being applied to the simulator input. Input scaling is usually applied throughout a simulation, hence these errors are expected for long durations at all frequencies. As indicated by Casas *et al.* [15] the I/O coherence is high for these errors. Since downscaling is applied, the Power Ratio will be between 0 and 1 by definition. S1 errors finally are characterized by the fact that no phase shift is applied between the simulation's input and output.
- *Type II scaling error (S2)* - In case not all frequencies are attenuated equally (as is the case for an S1 error), the error is designated an S2 error. These errors are typically caused by non-complementary high-pass and low-pass filters in MCAs [19]. In case such filters are used, both high-frequency and low-frequency motion are simulated with a (near) unity gain, but mid-frequency motion is attenuated. Similar to the previous, S2 errors have a Power Ratio between 0 and 1 by definition. Unlike for S1 errors, however, the signal coherence will drop in case of an S2 error since not all frequencies are attenuated equally. Like F1 and F2 errors, these errors mostly originate from filter effects, hence they are expected at mid-to-low frequencies. The error duration is expected to be in the order of seconds.

Phase errors (P) - The final category of motion cueing errors are phase errors, which are caused by delays between the simulator input and output signal. These delays can be caused by delays in sending commands to a simulator's motion system, a simulator's filter settings, its rate- and acceleration-limits or its inertia [19]. In case of a delay between simulator input and output, the I/O coherence will remain high and (on average) the Power Ratio will remain close to 1 (though the delay may cause it to take on any instantaneous value). P errors are expected to be noticeable at mid-to-low frequencies and are expected to last in the order of seconds.

For reference an overview of all distinguished motion cueing error types and their mathematical characteristics is given in Table I. From this table it can be concluded that F1, F2, S1 and S2 errors are mutually exclusive. F3 errors can occur in combination with all other error types, P errors could occur in combination with an S1 error.

III. WAVELET ANALYSIS

In the previous section both time- and frequency-based characteristics were identified as motion cueing error properties to be used for error identification, meaning that a spatio-temporal analysis method is required. Several such analysis methods exist, such as the Short-Term Fourier Transform (STFT) and the Wavelet Transform (WT) [24]. It has been shown that the WT is more efficient in making a time-frequency decomposition [25], for this reason the WT is selected as the method of analysis in this paper.

Wavelet analysis has already been applied in a variety of research fields, including geophysics [26]–[28], biology [29], [30] and economics [31]. Most studies use wavelet analysis to analyze the behavior of a single variable, but this paper uses wavelet analysis to compare two signals (simulator input and output). Also wavelet analysis is applied in this paper to signals that do not necessarily display periodic behavior (as opposed to sea temperatures [26], daily migrations [29] and exchange rates [31]).

1) Mathematical development: The explanation of wavelet analysis in this section is based on the work of Torrence & Compo [26]. Like the Fourier transform (FT), the wavelet transform (WT) is based on computing the inner product of a time function $f(t)$ with an analyzing function $\psi(t)$. For the FT this analyzing function $\psi(t)$ is a complex exponential function of frequency ω [32]:

$$\begin{aligned} F(\omega) = \mathcal{F}\{f(t)\} &= \int_{-\infty}^{\infty} f(t)\psi(t)\mathrm{d}t \\ &= \int_{-\infty}^{\infty} f(t)e^{-i\omega t}\mathrm{d}t \end{aligned} \tag{2}$$

In (2) $\mathcal{F}\{\cdot\}$ denotes the Fourier Transform operator and $F(\omega)$ denotes the Fourier Transform of $f(t)$. In case of the WT the analyzing function $\psi(t)$ is a stretched and translated version of a mother wavelet $\psi_M(t)$, the requisites for which are:

$$\int_{-\infty}^{\infty} \psi_M(t)\ \mathrm{d}t = 0 \tag{3}$$

$$\int_{-\infty}^{\infty} |\psi_M(t)|^2\ \mathrm{d}t = 1 \tag{4}$$

Unlike the analyzing function of the FT, wavelet functions are bounded in both time and frequency. As such, the mother wavelet can not only be stretched by a scale s (which changes

TABLE I: Descriptions and mathematical properties of error types. Properties denoted with * are used by the algorithm.

Category & Type		**Abbr.**	**Description**	**Frequency** ω [rad/s]	**Coherence** ρ^2 [-]	**Power Ratio** PR [-]	**Duration** Δt [s]	**Phase difference** $\Delta\phi$ [deg]
False	Type I	F1	Perceived simulator motion in the opposite direction to the direction expected based on visual cues.	Mid-Low *	Low *	$PR < 0$ *	Any	Yes
	Type II	F2	Perceived simulator motion when no/less vehicle motion was expected based on visual cues.	Mid-Low * All *	Low * High *	$PR > 1$ * $PR > 1$ *	Any Any	Yes No *
	Type IIIa	F3a	A short high-frequency disturbance of simulator motion which was not expected based on visual cues.	High *	Low *	Any	$<$ 0.5 s *	Yes
	Type IIIb	F3b	Continuous high-frequency disturbance of simulator motion which was not expected based on visual cues.	High *	Low *	Any	$\geq$ 0.5 s *	Yes
Scaling	Type I	S1	Perceived simulator motion is scaled down (equally over all frequencies) compared to visual cues.	All *	High *	$0 < PR < 1$ *	Any	No *
	Type II	S2	Perceived simulator motion is scaled down (unequally over all frequencies) compared to visual cues.	Mid-Low *	Low *	$0 < PR < 1$ *	Any	Yes
Phase		P	Perceived simulator motion appears out of phase with the visual cues.	Mid-Low *	High *	$\bar{PR} \approx 1$	Any	Yes *

the wavelet's frequency), but also be translated by τ seconds (to change the point in time at which the time function $f(t)$ is inspected). In case a wavelet is stretched horizontally by a scale s, a vertical scaling factor $\frac{1}{\sqrt{s}}$ is applied to meet requirement (4). The inner product for a given scale s over all points in time is then computed as:

$$W(s,t) = \int_{-\infty}^{\infty} f(\tau) \cdot \frac{1}{\sqrt{s}} \psi_M^* \left(\frac{t-\tau}{s} \right) \mathrm{d}\tau \tag{5}$$

$$= \mathcal{F}^{-1} \left[F(\omega) \cdot \frac{1}{\sqrt{s}} \cdot |s| \Psi_M(s\omega) \right] \tag{6}$$

In (5) * denotes the complex conjugate. $W(s,t)$ represents the wavelet transform of $f(t)$ at time t for a wavelet scale s. τ denotes the integration variable used in the convolution in (5). The convolution operation in (5) can also be evaluated in the frequency domain, leading to (6) (in which $\mathcal{F}^{-1}$ is the inverse Fourier Transform and $|s|\Psi_M(s\omega)$ is the Fourier Transform of $\psi_M(s,t)$. The end result of a repeated evaluation of (5) and (6) for a set of scales s is a two-dimensional spectrum, showing which frequency content is present at which points in time.

2) Choice of wavelet: A wide variety of analyzing functions satisfy conditions (3) and (4). As a result, one has to choose both a wavelet family and parameter set to use based on the time resolution (a wavelet's capability to distinguish signal features in time) and frequency resolution (a wavelet's capability to distinguish signal features in the frequency domain) offered. A complete overview of wavelet families and their parameters can be found in Appendix A. For this paper the frequency-based complex Morlet wavelet with ω_0 (the parameter which defines the wavelet's frequency at scale $s = 1$) set to 4 rad/s was chosen, for its balance between time and frequency resolution [33].

3) Post-processing: Wavelet analysis provides a limited time and frequency resolution, as a result signal power tends to spread out over time and frequency. This effect can partially be mitigated using two corrections: 1) determining the region of influence of edge effects and 2) significance testing.

The region of influence is discussed by Torrence & Compo [26]. However, their proposed methodology is not suitable for all wavelets discussed in Appendix A. For this reason a modified methodology was used, which determines the effective width of a wavelet $\psi(s,t)$ by investigating between which points in time 95% of a wavelet's energy is contained. This is done by twice solving (7), once for E = 0.025 and once for E = 0.975. This method is further explained in Appendix B.

$$\int_{-\infty}^{t_{\text{lim}}} |\psi_M(t)|^2 \, \mathrm{d}t = E \tag{7}$$

In a second post-processing step the power of the time-frequency based wavelet spectrum $|W(s,t)|^2$ is compared to a pre-determined background spectrum. This background spectrum is the power spectrum of red noise with the same variance as the analyzed time function $f(t)$ [26]. $W(s,t)$ is only considered the result of a feature in $f(t)$ in case $|W(s,t)|^2$ is significantly larger than this red noise background spectrum. In this way, the effects of power spread are filtered out. This method is further explained in Appendix B.

4) Wavelet coherence: As explained by Casas *et al.* [15] input/output coherence is an important factor in determining a subject's reaction to motion simulation. Wavelet analysis allows for determining this coherence per frequency over time, this process was explained by Torrence & Webster [34]. As a first step, the cross-wavelet spectrum of two signals is computed:

$$W_{X,Y}(s,t) = W_X(s,t) \cdot W_Y^*(s,t) \tag{8}$$

In (8) * again denotes the complex conjugate. The cross-wavelet spectrum is similar to the cross-spectral density $S_{X,Y}(\omega)$ of two signals. $S_{X,Y}(\omega)$ is the Fourier transform of two signals' cross-correlation and is defined by [35]:

$$S_{X,Y}(\omega) = \mathbf{E}\left[F_X^*(\omega) \cdot F_Y(\omega) \right] \tag{9}$$

In (9) $\mathbf{E}[\cdot]$ denotes the expected value. Note that (9) is similar to (8), except that the complex conjugate of $F_X(\omega)$ is taken, rather than that of $F_Y(\omega)$.

Using the wavelet cross-spectrum, the wavelet coherence $\rho^2(s,t)$ can be computed using [34]:

$$\rho^2(s,t) = \frac{\left| S\left(s^{-1} \cdot W_{X,Y}(s,t) \right) \right|^2}{S\left(s^{-1} \cdot |W_X(s,t)|^2 \right) \cdot S\left(s^{-1} \cdot |W_Y(s,t)|^2 \right)} \tag{10}$$

Additionally the phase difference $\phi(s,t)$ between two signals can be computed using:

$$\phi(s,t) = \tan^{-1}\left(\frac{\Im\left[S(s^{-1}\cdot W_{X,Y}(s,t))\right]}{\Re\left[S(s^{-1}\cdot W_{X,Y}(s,t))\right]}\right) \tag{11}$$

In (10) and (11) $S(\cdot)$ is a smoothing function in both time and frequency, as discussed by Grinsted, Moore and Jevrejeva [33]. $\Re[\cdot]$ and $\Im[\cdot]$ denote the real and imaginary part of a number, respectively. In [33] smoothing in time is only implemented for the frequency-based, complex-valued Morlet wavelet. An extension to this method is made in Appendix C, in which modulating functions (which are used as a filter in time) are discussed for all wavelets listed in Appendix A.

Again, (10) and (11) are similar to the expressions for determining the coherence and phase difference between two signals using a Fourier analysis, which can be done using [35]:

$$\rho^2{}_{X,Y}(\omega) = \frac{|S_{X,Y}(\omega)|^2}{S_{X,X}(\omega)\cdot S_{Y,Y}(\omega)} \tag{12}$$

$$\phi_{X,Y}(\omega) = \tan^{-1}\left(\frac{-\Im\left[S_{X,Y}(\omega)\right]}{\Re\left[S_{X,Y}(\omega)\right]}\right) \tag{13}$$

The difference between (11) and (13) is the omission of the smoothing function $S(\cdot)$ in (13) and the minus operator in the numerator of (13). The latter stems from the fact that the complex conjugate of $F_X(\omega)$ was taken, rather than that of $F_Y(\omega)$, in (9).

IV. MOTION CUEING ERROR DETECTION ALGORITHM

The motion cueing error detection algorithm (to be referred to from now on as "the algorithm") described in this section uses wavelet analysis to detect the motion cueing error characteristics listed in Table I. The algorithm's structure is based on Grant's motion cueing algorithm tuning paradigm [12], however, our algorithm does not combine the detection or errors and tuning of the MCA. Instead, by only detecting motion cueing errors without using knowledge of the MCA used, our algorithm can be used for any MCA. In the specific case of a classical washout filter, the algorithm presented in this paper could replace the subjective evaluation pilot used to provide input to the algorithm in [12].

A. Algorithm description

The algorithm's process, which is repeated for all six motion channels, consists of five steps and is illustrated in Fig. 1.

Step 0 *Threshold evaluation* – Given a simulator input and output signal, the algorithm first checks whether *all* input and output is below threshold level Φ. If so, then a subject is assumed not to have experienced any motion nor to have expected any based on the visual information. Thresholds have been used from research by Nesti *et al.* [21] (for rotational rates) and Heerspink *et al.* [36] (for linear accelerations), see Table II.

TABLE II: Motion perception thresholds [21], [36]–[38].

Motion channel	**Perceptual Theshold**	PR_{min}	PR_{max}
a_x	0.04 m/s^2	0.5	1.3
a_y	0.04 m/s^2	0.5	1.3
a_z	0.08 m/s^2	0.5	1.3
ω_x	3 deg/s	0.6	1.1
ω_y	3 deg/s	0.6	1.1
ω_z	3 deg/s	0.6	1.1

Step 1 *Error analysis* – The error signal is first computed from the simulator input and output. This error signal is wavelet transformed using a frequency-based, complex-valued Morlet wavelet with ω_0 = 4 rad/s, to obtain a time-frequency decomposition of the error signal. Using the significance test discussed in Section III, regions of significant error (at a 95% significance level, with a red-noise autocorrelation coefficient of 0.9) are identified in this time-frequency space. Steps 2-4 are then repeated for each significant region.

Step 2A *Frequency analysis* – From Table I it can be seen that the frequency content of an error is important in determining its type, hence an error is first designated high or mid-low frequency. This is done by comparing the minimal error frequency found in Step 1 to the reference value of 0.5 Hz proposed by Grant [12]. Errors containing solely power at frequencies higher than 0.5 Hz are designated high-frequency.

Step 2B *High-frequency analysis* – In case an error is high-frequency, the algorithm continues with Step 2B to determine whether the error is an F3 or S1/F2 error, the distinction between which is the I/O coherence. To determine this quantity, a coherence spectrum $\rho^2(s,t)$ is first computed using (10). Next, a high-frequency coherence value $\rho^2_{\text{HF}}(t)$ is found for all points in time by averaging the wavelet coherence $\rho^2(s,t)$ over all frequencies between 2 Hz and 10 Hz (frequencies between 0.5 Hz and 2 Hz are left out as they tend to contain artifacts of signal spread at lower frequencies).

In case the coherence level $\rho^2_{\text{HF}}(t)$ remains above a set threshold $\rho^2_{\text{HF,min}}$, it is concluded that either an S1 error or an F2 error occurred, or nothing is wrong. The latter is concluded in case the Power Ratio computed in (1) falls within the coherence zones described by Correia Grácio *et al.* [37] and Valente Pais *et al.* [38]. Threshold values for this scaling are given in Table II. In case the Power Ratio in (1) is lower than PR_{min} an error is designated an S1 error, in case the PR exceeds PR_{max} an F2 error is detected.

In case the high-frequency coherence level $\rho^2_{\text{HF}}(t)$ drops below a certain threshold $\rho^2_{\text{HF,min}}$, an F3 error is found. To distinguish between F3a and F3b errors, the error duration is determined from the region of significant error of Step 1. In case this duration is below a certain Δt_{min} (set to 0.5 s, as errors resulting from hitting a simulator displacement limit are expected to last shorter than that) the error is designated an F3a error, otherwise an F3b error is found.

Step 3 *Threshold analysis* – In order to prevent false positives or false negatives in case of subthreshold input or output (e.g., labelling subthreshold simulator washout a false cue), it is first checked whether the input and output at time t are above threshold level Φ, see Table II. In case both are below threshold level Φ no error is called, in case only the input is above threshold level Φ then an S2 error is assigned. Finally, if there is noticeable output when no input is expected an F2 error is found.

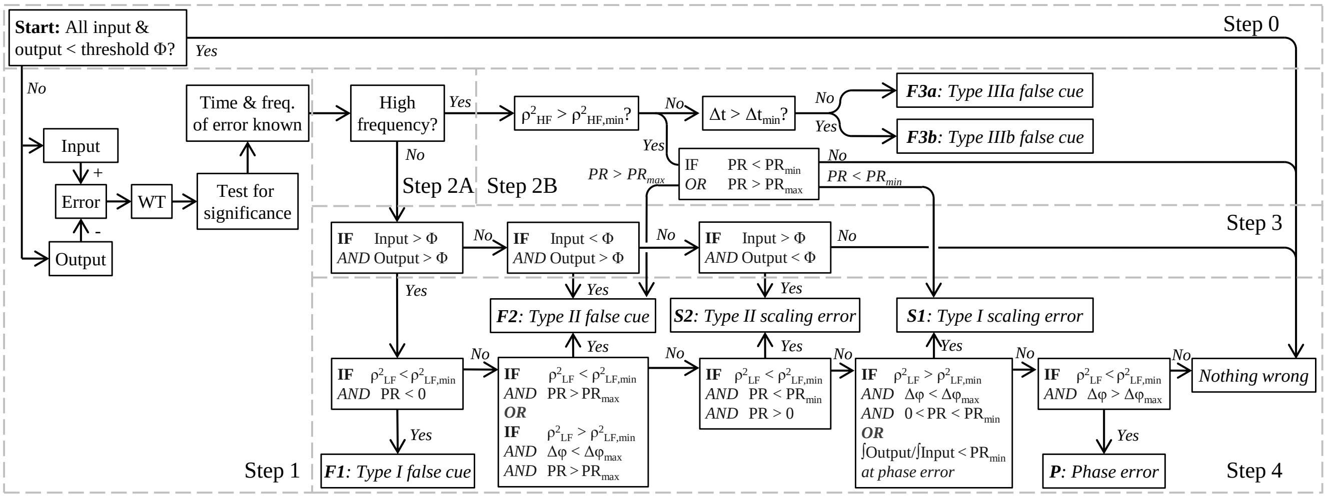


Fig. 1: Flowchart of the motion cueing error detection algorithm discussed in this paper, with its four steps highlighted. Φ denotes a perceptual threshold.

Step 4 *Power Ratio, Coherence & Phase analysis* – Should the error be of mid-to-low frequency and both the input and output be above threshold level, then three properties are determined prior to assessing of what type(s) the error is.

Like in the high-frequency case, a low-frequency coherence value $\rho^2_{\text{LF}}(t)$ is computed by averaging the wavelet coherence spectrum $\rho^2(s,t)$ over all frequencies between 0.05 and 0.5 Hz (since effects with a period longer than 20 s are not expected to be dominant). The Power Ratio is computed from (1). As a third indicator the phase difference $\Delta\phi(t)$ between the input and output is computed by averaging the phase spectrum $\phi(s,t)$ obtained in (11) over all frequencies between 0.05 and 10 Hz. Using PR(t), $\rho^2_{\text{LF}}(t)$ and $\Delta\phi(t)$ as indicators, the remaining error types can be identified.

Phase errors stand out due to their high I/O coherence and large phase shifts between input and output. Beckers *et al.* [39] concluded that phase shifts larger than $\Delta\phi_{\text{max}}$ = 16.4 deg would lead to humans perceiving a phase error. In case a phase error is detected, the integrals of the (absolute) input and output during the phase error are compared to assess whether an S1 error was present simultaneously. The Power Ratio is not suitable for use here, since the delay between input and output could cause the PR to take on any value.

In case the low-frequency coherence $\rho^2_{\text{LF}}(t)$ is higher than $\rho^2_{\text{LF,min}}$ and $\Delta\phi_{\text{max}}$ is not exceeded, an S1 error is detected in case the PR is positive and smaller than PR_{min}. Since deliberate exaggeration of motion is not expected in motion simulation (given simulators' limited motion envelopes), any error for which the PR is larger than PR_{max} whilst having high coherence and no phase error will be designated an F2 error.

This means two causes of F2 errors have already been covered, the final cause is identified in case the I/O coherence drops below $\rho^2_{\text{LF,min}}$ and the PR is larger than PR_{max}. F1 errors are characterized by their negative PR, hence in case this negative PR is accompanied by a low I/O coherence this error can be singled out.

As a final error type, S2 errors are called out in case the low-frequency I/O coherence drops below $\rho^2_{\text{LF,min}}$ and the PR is positive, but smaller than PR_{min}.

Step 5 *Post-processing* – Two steps of post-processing are performed on the output of Steps 1-4. First it is checked that gaps between equal error types are at least 0.2 s long. If not, then errors are made continuous. Should an F3a error longer than 0.5 seconds be found (resulting from multiple F3a errors being lumped together in the first step of post-processing) then this error is post hoc redesignated as F3b.

B. Algorithm demonstration

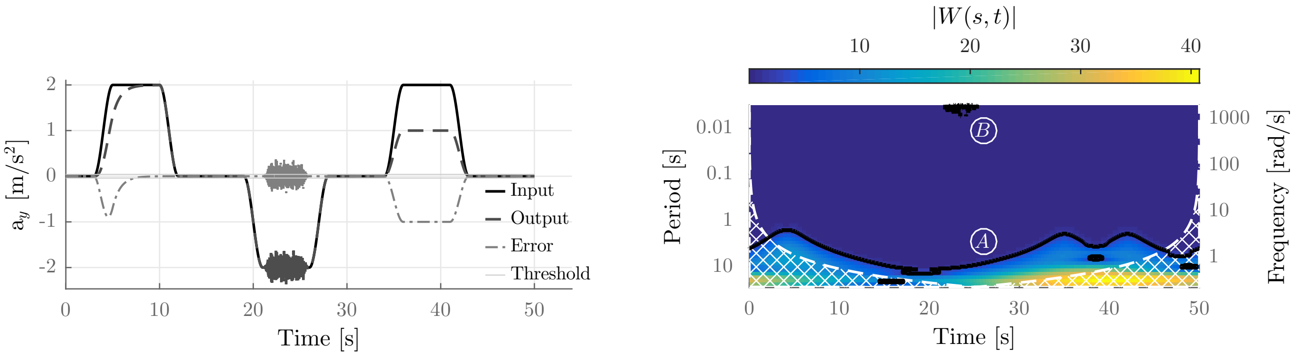
The detection algorithm will be demonstrated in this section, using the input and output signals shown in Fig. 2a. The input signal (sampled at 1000Hz) is given by:

$$\begin{aligned} I(t) &= 2\sin^2\left(0.25\pi(t-3)\right) && \text{for } 3 < t \leq 5 \\ I(t) &= 2 && \text{for } 5 < t \leq 10 \\ I(t) &= 2\cos^2\left(0.25\pi(t-10)\right) && \text{for } 10 < t \leq 12 \end{aligned}$$

This shape is repeated (negatively) for t = 19-28 s and for t = 34-43 s, with the signal being 0 outside these intervals. The output was created by filtering the input using a first-order high-pass filter (ω_n = 1.2 rad/s) between t = 2-10 s, adding white noise (σ = 0.1) for t = 21-26 s (building up and dying out in the first and last second) and scaling the input down 50% for t = 34-43 s.

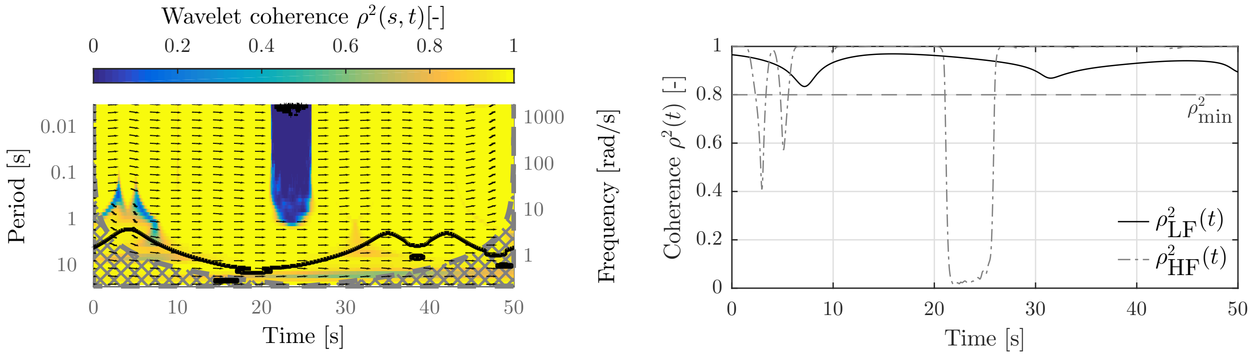
Step 1 *Error analysis* – A wavelet transform of the error signal shown in Fig. 2a results in the spectrum shown in Fig. 2b. Two areas of significant error are found: The entire low-frequency area of the plot (A) and a high-frequency region around t = 22 s (B). The latter consists of many small regions, due to the very limited width of a wavelet at these small scales s (which correspond to high frequencies). This shows wavelet analysis is not able to continuously identify high-frequency noise.

Step 2A *Frequency analysis* – From Fig. 2b it is easily seen that the bottom region stretches to frequencies below 0.5



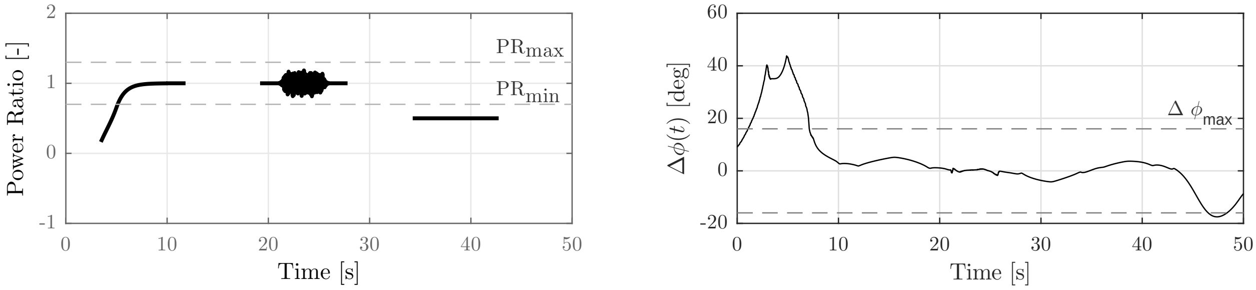
(a) Simulator input, output and motion cueing error trace. Perception thresholds for lateral acceleration are also shown.

(b) Step 1: Wavelet transform of the error signal shown in Fig. 2a. Significant regions are enclosed by the black contour.



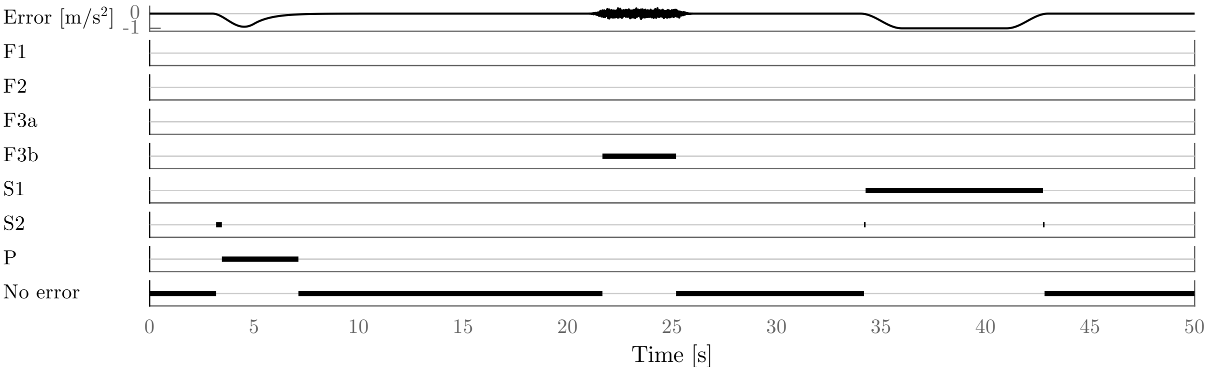
(c) Step 2B & 4: Coherence spectrum $\rho^2(s,t)$ and phase difference $\phi(s,t)$ (indicated by arrows) for the input and output signal shown in Fig. 2a. Arrows pointing right indicate no phase difference, arrows pointing down indicate a 90° output lag, etc. The significance contour of Fig. 2b has been copied.

(d) Step 2B & 4: Low-frequency coherence $\rho^2_{\text{LF}}(t)$ and high-frequency coherence $\rho^2_{\text{HF}}(t)$ for the input and output signals shown in Fig. 2a.



(e) Step 4: Power Ratio PR for the input and (filtered) output signal, computed only for points were both the input and output are above threshold level Φ.

(f) Step 4: Phase difference $\Delta\phi(t)$ between the input and output over time. Positive values indicate output lag.



(g) Algorithm output for the example shown.

Fig. 2: Visual algorithm output for the demonstration discussed.

Hz and that the top region does not. The latter is therefore labelled high-frequency and treated further in Step 2B.

Step 2B *High-frequency analysis* – The high-frequency coherence $\rho^2_{\mathrm{HF}}(t)$ is obtained by averaging the wavelet coherence spectrum $\rho^2(s,t)$ shown in Fig. 2c over all frequencies between 2 Hz and 10 Hz, resulting in the line shown in Fig. 2d. Since the high-frequency coherence $\rho^2_{\mathrm{HF}}(t)$ drops below ρ^2_{min} (= $\rho^2_{\mathrm{HF,min}}$ = $\rho^2_{\mathrm{LF,min}}$ in this demonstration) between t = 22 s and t = 26 s the high-frequency error identified in Step 1 is an F3 error. The significant region shown in the top of Fig. 2b consists of many small regions, each lasting less than 0.5 s, meaning a succession of F3a errors is (incorrectly) identified here.

Step 3 *Threshold analysis* – In Fig. 2a the input exceeds the perceptual threshold without the output exceeding it on three occasions: at t = 3.2 s, t = 34.2 s and t = 42.6 s. All are marked as S2 errors.

Step 4 *Power Ratio, Coherence & Phase analysis* – For completing Step 4 the low-frequency coherence $\rho^2_{\mathrm{LF}}(t)$, Power Ratio PR(t) and phase difference $\Delta\phi(t)$ of the input and output are required. The first is shown in Fig. 2d, having averaged the coherence $\rho^2(s,t)$ of Fig. 2c over all frequencies between 0.05 Hz and 0.5 Hz. The Power Ratio is shown in Fig. 2e, $\Delta\phi(t)$ (obtained by averaging $\phi(s,t)$ over all frequencies between 0.05 and 10 Hz) is shown in Fig. 2f.

From Fig. 2d it can be seen that the coherence $\rho^2_{\mathrm{LF}}(t)$ never drops below the threshold value of 0.8. Since Fig. 2f shows a large phase difference between t = 4 s and t = 7 s, this error part is labelled a P error. Between t = 34 s and t = 43 s one can see the coherence $\rho^2_{\mathrm{LF}}(t)$ remaining high but the PR dropping below PR$_{\mathrm{min}}$, so this is an S1 error. Between t = 19 s and t = 28 s the coherence $\rho^2_{\mathrm{LF}}(t)$ is high, the PR is within bounds and no lead or lag is identified, so no error is called here in addition to the F3a errors found in Step 2B.

Step 5 *Post-processing* – In Step 2B a succession of F3a errors was identified. Since the gap between these is under 0.2 s these are connected in Step 5, meaning the total length exceeds 0.5 s. As a result, now an F3b error is identified here. This means the output of the algorithm is as shown in Fig. 2g, with three brief S2 errors and a P, F3b and S1 error being identified.

C. Algorithm limitations

In this section the algorithm for detecting motion cueing errors has been explained and demonstrated. Based on findings of this demonstration a number of improvements and topics of further research are still foreseen.

- The output of the algorithm could have been different in case other parameter values were chosen. For example, increasing $\rho^2_{\mathrm{LF,min}}$ would result in an S2 error being detected rather than a P error between t = 4 s and t = 7 s. Also, lowering PR$_{\mathrm{min}}$ would result in no S1 error being found between t = 34 s and t = 43 s. Experiments are thus needed to determine suitable parameter values.
- The error between t = 4 s and t = 7 s is identified as a phase error, however one could also argue that this is an S2 error. Phase errors are essentially combinations of S2, F1 and F2 errors, hence they could also be identified as such. This ambiguity could be resolved by requiring the input signal to be periodic for a phase error to be possible. This change would require an extra check on periodicity to be implemented in the algorithm.
- Currently, the algorithm treats error recognition by humans as a binary process with fixed thresholds. It is, however, well established that human perception is a probabilistic process best described using psychometric functions [11]. Should such psychometric functions be included in the algorithm, then the algorithm could indicate a probabilistic chance of noticing a certain error type, rather than producing a binary output. To do so, research on psychometric functions for thresholds and coherence zones for all six motion channels would have to be combined.
- The coherence $\rho^2(t)$ and phase difference $\Delta\phi(t)$ are currently determined by averaging the coherence spectrum $\rho^2(s,t)$ and phase spectrum $\phi(s,t)$ over a fixed frequency band. The frequency bands used in this paper were shown to work well for the demonstration given, however they might be less suited for other analyses.
  The determination of coherence $\rho^2(t)$ and phase difference $\Delta\phi(t)$ might be improved by averaging $\rho^2(s,t)$ and $\phi(s,t)$ over frequencies at which the input and output signal actually contain power, instead of using fixed frequency bands. This change would prevent the drops in high-frequency coherence ρ^2_{HF} seen in Fig. 2d at t = 3 s and t = 6 s. These drops are caused by spread of lower-frequency effects (see Fig. 2c). From Fig. 2b, however, one can observe the error not to contain power at these frequencies.
- In Fig. 2g three brief S2 errors are displayed. All are caused by the input signal exceeding threshold level Φ slightly earlier than the output signal, or the output signal decreasing to below the threshold level Φ earlier than the input. These brief errors are not expected to be noticed by humans. An experiment assessing what the minimum duration of an S2 or F2 error should be for it to be noticeable would allow for removing identified errors which are too short.
- In further experiments with the algorithm it was found that the algorithm has most difficulty with detecting high-frequency errors. Decreasing the amplitude of the noise added to the output signal with 20% (to σ = 0.08) for example results in most of region B of Fig. 2a no longer being marked as significant. Lowering the sampling rate had the same effect on the algorithm's output.
  This finding is explained by the wavelet normalization used by Torrence & Compo [26], who multiply all wavelet values with $\sqrt{\frac{2\pi s}{\Delta t}}$ to ensure unit energy. This normalization causes (5) to produce low values for $W(s,t)$ for small s (high frequencies) and lower sampling frequencies. This wavelet normalization has been disputed [40], however the significance test of [26] used in this paper is based on normalized wavelets. Further research into significance testing for non-normalized

wavelets might thus improve the performance of the algorithm for high-frequency signals and for signals sampled at a lower rate.

- In this paper, the algorithm was used in an offline fashion, i.e., the entire time history of the input and output are known. For driving experiments, however, online MCAs are required and the future driver input is unknown. An online implementation of the algorithm would pose two challenges: the available computation time and the lack of future input and output values.
  Currently, the wavelet transform and coherence computations performed for each motion channel require large computation times. To reduce computation times the signals could be downsampled and/or less scales could be used (in case the previous point is addressed). Both options, however, would reduce the time/frequency resolution of the analysis. Further reductions in computation times could be achieved by replacing the wavelet transform with a less computationally expensive analysis method, e.g., the Windowed Fourier Transform using a short window. This would, however, limit the lowest frequency that can be analyzed.
  Additionally, (5) and (6) are non-causal, as they make use of values of $f(t)$ for $t > t_1$ when computing $W(t_1, s)$. In case of online simulation, these values are unknown. Predictive filters would thus be required to estimate future input and output values.

V. METHODS

In order to tune the parameters $\mathrm{PR}_{\mathrm{min}}$, $\mathrm{PR}_{\mathrm{max}}$ and $\rho^2_{\mathrm{LF,min}}$, and determine what motion cueing error types should be distinguished by the algorithm, a between-subjects simulator experiment was performed. During this experiment subjects passively experienced a car drive, meaning they did not have to operate the throttle or steering wheel. As such, they could fully concentrate on their task, which was to continuously rate their perceived visual-motion mismatch (PVMM) using a rating method described by Cleij *et al.* [16].

A. Participants

In total 16 participants (11 male, 5 female) aged between 23 and 40 years (μ = 28.4 yrs, σ = 6.1 yrs) took part in the experiment, all naive to the goal of the experiment. All participants were, or had been, in possession of a driver's license for at least 3 years (μ = 9.3 yrs, σ = 4.6 yrs) and drove a car 3.8 hours per week on average (σ = 4.4 hr). Simulator experience ranged from novice (first time in a simulator, 6 subjects) to intermediate (have been in a simulator before, 6 subjects) and expert (researcher on motion cueing, 5 subjects).

B. Apparatus

The experiment was performed in the CyberMotion Simulator (CMS) of the Max Planck Institute for Biological Cybernetics [41], [42], see Fig. 3. This eight degree-of-freedom simulator is based on an industrial robot (Kuka GmbH, Germany). Car inertial motion and visual images were recorded using CarSim (Mechanical Simulation Corporation), a vehicle simulation program. An impression of the visual images shown is shown in Fig. 4. Subjects indicated their rating by turning a rotary knob (SensoDrive GmbH, Germany), see Fig. 5. The current rating of a subject was shown on the vehicle dashboard, see Fig. 4.

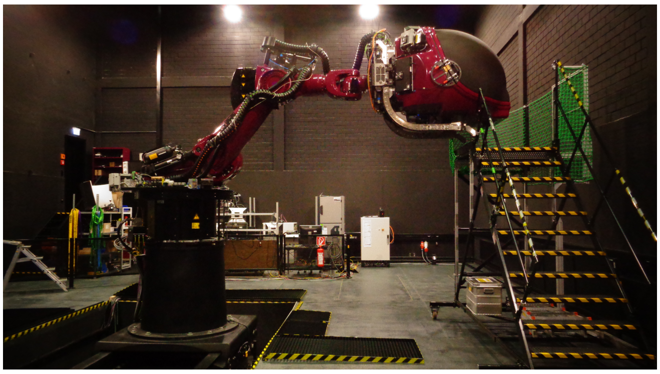


Fig. 3: The CyberMotion Simulator (CMS) at the Max Planck Institute for Biological Cybernetics.

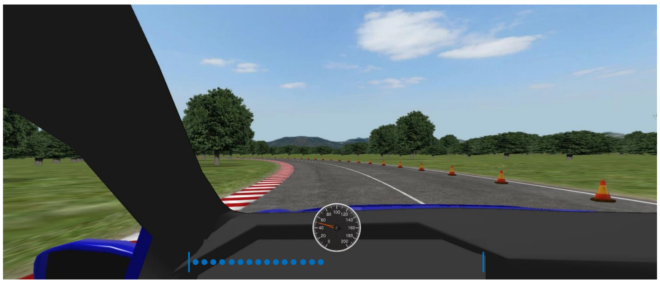


Fig. 4: Outside visuals shown during the experiment, showing the reference line drawn on the road and the displayed rating bar at the bottom in blue.

C. Scenario and conditions

For the experiment a car drive featuring a straight line acceleration from standstill to 50 km/h, followed by 22 90-degree turns with a 120 m radius (separated by 150 m straights) and a deceleration back to standstill was recorded using CarSim. Each complete drive took 9 minutes in total. The vehicle specific forces and rotational rates for a curve to the left are shown in Fig. 6.

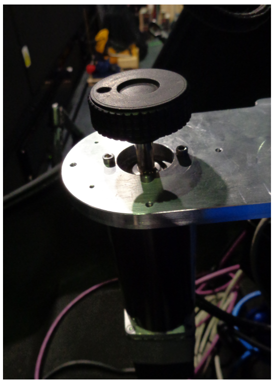


Fig. 5: Rotary knob used by subjects to indicate their PVVM.

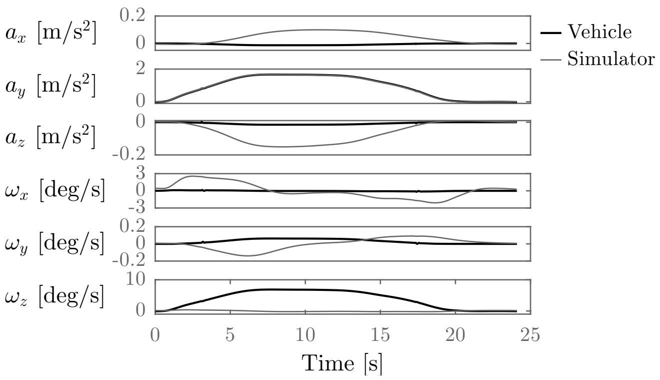


Fig. 6: Vehicle (black) and simulator (gray) specific forces and rotational rates during a curve to the left, as recorded in CarSim.

The automatic CarSim driver steered in and out of curves relatively early and slowly. This steering behavior reduced the lateral jerk, ensuring that the tilt-coordination roll rate for lateral specific force simulation remained below 3 deg/s throughout the experiment [21]. To assist subjects in judging when the automatic driver steered, a line was drawn on the center of the road, see Fig. 4. On straight road sections this line aligned with the head of the subjects, in the curves the automatic driver would deviate from this line.

For simulating this motion in the CMS a washout filter was used. This MCA first multiplies all input by a fixed gain k for each motion channel. Next, the high-frequency motion (which is simulated by directly translating or rotating the simulator cabin) is obtained by filtering the scaled input with the following high-pass filter $H_{\text{HP}}(s)$:

$$H_{\text{HP}}(s) = \frac{s^2}{s^2 + 2\zeta_{\text{HP}}\omega_{n_{\text{HP}}}s + \omega_{n_{\text{HP}}}^2} \cdot \frac{s}{s + \omega_{n,2_{\text{HP}}}} \quad (14)$$

In this equation s is the Laplace variable and $\omega_{n_{\text{HP}}}$ is the high-pass break frequency in rad/s, with ζ_{HP} the damping ratio. $\omega_{n,2_{\text{HP}}}$ ensures additional input attenuation at lower frequencies, and ensures the simulator returns to its neutral position for constant specific force input. Sustained motion (which was simulated using tilt-coordination) was filtered using the following low-pass filter $H_{\text{LP}}(s)$:

$$H_{\text{LP}}(s) = \frac{\omega_{n_{\text{LP}}}^2}{s^2 + 2\zeta_{\text{LP}}\omega_{n_{\text{LP}}}s + \omega_{n_{\text{LP}}}^2} \quad (15)$$

Again, $\omega_{n_{\text{LP}}}$ is the low-pass break frequency in rad/s, with ζ_{LP} the damping ratio. In the baseline condition the lateral specific force was simulated one to one through a high-pass filter, a low-pass filter and a complementary band-pass filter which provided commands to the simulator's linear rail. Parameter settings for all motion channels are summarized in Table III. The resulting motion of the MCA is compared to the vehicle motion in Fig. 6.

Two curves featured a one-to-one simulation of the lateral specific force (the baseline condition), 20 curves featured a modified lateral specific force. For most conditions, this modified lateral specific force was obtained through modifications to the washout filter. For seven conditions (C14, C15, C17 and C19-C22), the lateral specific input force was obtained by modifying the MCA input. An overview of these conditions is given in Table IV. All 22 curves were spread random over each of the 6 runs used in the experiment.

Fig. 7: Rating bar displayed to subjects.

D. Independent variables & Dependent measures

The specific lateral force error conditions specified in Table IV are the independent variable of the experiment. Conditions C1 and C2 serve as the baseline of the experiment and allow for checking the consistency in subjects' ratings. Conditions C3-C8 were chosen to investigate the effect of lateral specific force scaling on a simulator's fidelity. The maximum scaling factor was chosen such that the maximum tilt-coordination roll rate was 3 deg/s.

Conditions C4, C9, C14, C16, C18, C19 and C22 were included to study the effect of different motion cueing error types on subjects' perception of a simulation. Conditions C10-C13 each feature a different minimum value of $\rho^2_{\text{LF}}(t)$ and thus serve to determine a suitable value for $\rho^2_{\text{LF,min}}$. Condition C15 was included to study the effect of the magnitude of an F1 error, condition C17 was included to investigate whether an F2 error with a more distorted shape increased subjects' ratings. Conditions C20-C22 finally were included to study the effect of delays between input and output.

As a dependent measure the method of Continuous Rating as introduced by Cleij *et al.* was used [16]. Subjects provided a continuous indication of their PVMM through turning a rotary knob with a range of -60 to 60 degrees (see Fig. 5). Feedback on their current rating was displayed on their screen using a bar as shown in Figs. 4 and 7. This indicated PVMM was converted to a score between 0 and 100 for later analysis using:

$$\text{Score} = \frac{\text{Rating knob angle} + 60}{120} \cdot 100 \quad (16)$$

E. Experiment procedure

Prior to the experiment all subjects were briefed on the experiment through an instruction form and additional verbal explanation. Subjects were explained that the motion they saw would not always correspond to the motion they felt during this experiment. They were instructed to continuously rate (by turning the rotary knob) the discrepancy between

TABLE III: Motion washout filter parameters.

Motion channel	k [-]	$\omega_{n_{\text{HP}}}$ [rad/s]	ζ_{HP} [-]	$\omega_{n,2_{\text{HP}}}$ [rad/s]	$\omega_{n_{\text{LP}}}$ [rad/s]	ζ_{LP} [-]
a_x	0.3	1.5	1	0.8	1.4	1
a_y	1	-	-	-	-	-
a_z	1	0.5	1	0.3	0	-
ω_x	0.5	0	-	0.5	0	-
ω_y	0.5	0.5	1	-	0	-
ω_z	0.3	0	-	0.5	0	-

TABLE IV: Specific lateral force conditions. Conditions marked with * were obtained through input modification rather than washout setting modifications.

	Error type	Washout filter modifications k_y, ζ *in [-],* ω_n *in [rad/s]*
C1	-	None
C2	-	None
C3	S1	Scaled with k_y = 0.4
C4	S1	Scaled with k_y = 0.5
C5	S1	Scaled with k_y = 0.6
C6	S1	Scaled with k_y = 0.8
C7	F2	Scaled with k_y = 1.2
C8	F2	Scaled with k_y = 1.275
C9	S2	LP-filtered using ω_n = 0.6, ζ = 1 for t<10 s
C10	S2	Filtered using $\omega_{n_{\text{HP}}}$ = 0.5, ζ_{HP} = 0.8, $\omega_{n_{\text{LP}}}$ = 2.4, ζ_{LP} = 1.2
C11	S2	Filtered using $\omega_{n_{\text{HP}}}$ = 0.5, ζ_{HP} = 0.6, $\omega_{n_{\text{LP}}}$ = 2.0, ζ_{LP} = 1.2
C12	S2	Filtered using $\omega_{n_{\text{HP}}}$ = 0.55, ζ_{HP} = 0.35, $\omega_{n_{\text{LP}}}$ = 2.0, ζ_{LP} = 1.2
C13	S2	Filtered using $\omega_{n_{\text{HP}}}$ = 0.5, ζ_{HP} = 0.125, $\omega_{n_{\text{LP}}}$ = 1.2, ζ_{LP} = 1
C14*	F1	$a_y(t) = \pm 0.6 \left(e^{\left(\frac{t-2.5}{1.2}\right)^2} - e^{\left(\frac{t-17}{1.6}\right)^2} \right)$
C15*	F1	$a_y(t) = \pm 0.85 \left(e^{\left(\frac{t-4}{2}\right)^2} - e^{\left(\frac{t-15.6}{2}\right)^2} \right)$
C16	F2	LP-filtered using ω_n = 0.6, ζ = 1 for t>12 s
C17*	F2	Error of C9 added to input after t>10 s
C18	F3a	Scaled with k_y = 0.5 for Δt = 0.05 s
C19*	F3b	-24dBW white noise added to input
C20*	P	Input delayed by 0.6s
C21*	P	Input delayed by 1.5s
C22*	P	Input delayed by 2.5s

the visual motion they saw and the inertial motion they felt on a scale ranging from zero (for perfect correspondence between visual and inertial motion) to a maximum rating (for the biggest mismatch subjects encountered during a run), see Fig. 7. Additionally examples of visual-inertial mismatches were explained to subjects.

Following these instructions, subjects signed an informed consent form and filled in a simulator sickness questionnaire [43]. The latter form would be repeated after the experiment to verify whether simulator sickness could have influenced the results of this study. Two subjects reported slight sickness effects, however, it is not believed these effects influenced the results of the experiment so all data (n = 16) was used.

After the experiment preparations, subjects started with a familiarization run consisting of 10 curves of condition C1. This run served to make subjects familiar with the driving style of the automatic CarSim driver, which steered in and out of curves relatively slowly in order to keep the simulator tilt coordination roll rate below 3 deg/s in all conditions [21]. Following this familiarization run, subjects were instructed to take this run as a reference for coherent motion.

Next, subjects experienced two training runs, which featured the full drive including the lateral specific force conditions thought to cause the smallest and largest PVMM. During the first training run, subjects were asked to focus on what motion corresponded most and least to the motion they saw, allowing them to determine the end points of their rating scale (see Fig. 7). Following this training run, subjects were specifically asked to describe the motion that felt worst. During the second training run, subjects were asked to continuously provide ratings of their PVMM on the scale developed during the first training run. Subjects were instructed they could change their rating scale during this second training run (should they encounter motion that was even worse than previously experienced), but were instructed to keep a constant rating scale from the training onwards. After going through the training phase of the experiment, the four experimental runs were presented in random order.

To prevent fatigue from influencing the experiment's results a break outside the simulator was introduced after the first block of four runs (consisting of the familiarization run, both training runs and the first experimental run). Following the break the familiarization run was repeated, followed by the final three experimental runs. Between runs within a block, subjects had the possibility to take a break in the stationary simulator. In total, the experiment lasted just under two hours.

F. Hypotheses

The goal of this experiment was to tune parameters of the motion cueing error detection algorithm and to investigate what motion cueing errors should be distinguished. Prior to the experiment the following hypotheses were set up:

- **H1A** *The minimum power ratio* PR_{min} *is 0.5.*
  Based on research by Correia Grácio *et al.* [37] it is hypothesized that subjects will indicate motion as being too small for scaling factors lower than 0.5. To investigate this hypothesis, conditions C3-C6 will be compared to C1 during the steady-state part of the curve.
- **H1B** *The maximum power ratio* PR_{max} *is 1.3.*
  Based on research by Correia Grácio *et al.* [37] it is hypothesized that subjects will indicate motion as being too strong for scaling factors higher than 1.3. To investigate this hypothesis, conditions C7-C8 will be compared to C1 during the steady-state part of the curve.
- **H1C** *The minimum coherence value* $\rho^2_{LF,min}$ *is 0.8.*
  Based on algorithm demonstration in Section IV a coherence threshold of 0.8 is hypothesized. To investigate this hypothesis, conditions C10-C13 will be compared to C1 during the steady-state part of the curve. These conditions have a minimal coherence value $\rho^2_{\text{LF,min}}$ of 0.85, 0.75, 0.65 and 0.45, respectively.
- **H2** *All error types are rated differently.*
  The goal of this research was to define a breakdown of motion cueing errors and to investigate which of these should be distinguished in analyzing motion simulator performance. The effect of all lateral specific force cueing errors was assessed by analyzing the average maximum rating given to each condition. To allow for fair comparison between all seven motion cueing error types, the maximum lateral specific force error of conditions C4, C9, C16, C18 and C22 was kept equal. Additionally conditions C14 and C19 were analyzed, for these conditions it was not possible to set the maximum error equal to that of the other five conditions.

VI. RESULTS

For all subjects, the continuous ratings were averaged over their four repeated runs (one subject indicated to have misunderstood instructions after the first run, for this subject the final three runs were used), yielding 16 average ratings per condition. For all analyses a significance level of α = 0.05 was used.

A. Threshold determination

Algorithm thresholds (PR_{min}, PR_{max} and $\rho^2_{\text{LF,min}}$) are determined based on conditions C1, C3-C8 and C10-C13. For these conditions the lateral specific force and the average rating of all participants are shown in Fig. 8, together with an indication of the spread in ratings.

From Fig. 8 one can see subjects rated the straight sections of the trajectory as (almost) perfect. Since ratings went back to zero between curves, it can be concluded that the ratings for one curve do not directly influence the ratings of the next. Fig. 8 shows downscaled conditions to be rated worse than conditions for which k_y >1. Also, ratings for C1-C6 increase with decreasing k_y from k_y = 1 onwards. This finding matches the observations made by Jamson [44] and Berger *et al.* [45], who in a similar experiment (in terms of motion profile and intensity) showed that subjects preferred one-to-one inertial and visual motion.

Ratings for C10-C13 only differ from that of C1 at times when an error is present, with subjects showing consistent ratings up until t = 8 s. Interesting to note is that condition C4 is rated far worse than condition C13, even though the maximum error in lateral specific force is equal. Based on the division between shape and scale errors proposed by Grant *et al.* [17] and Fischer [18], one would have expected the shape error of C13 to be rated worse than the scale error of C4.

To comment on suitable algorithm threshold values the (average) ratings given by participants were averaged between $t = 9$ s and $t = 15$ s, which corresponds to the part of the curve featuring a constant lateral specific force (taking into account the observed 1.5 s delay in subject ratings, which matches the delay found by Cleij *et al.* [14]). The distribution of these values is shown in Fig. 9 for conditions C1-C8 and C10-C13.

A first observation from Fig. 9 is that both repetitions of the unmodified lateral specific force condition (C1 and C2, see Table IV) are not rated as perfect. This indicates that subjects used real car driving as their reference. As such, subjects will have rated the filtered yaw rate (see Fig. 6) or the false roll rates (see Fig. 10). Based on comments made by subjects it is most likely that subjects rated false roll angles, with the roll angles of around 10° exceeding the somatosensory threshold of 9° found by Bringoux *et al.* [46].

Fig. 9 also shows that the ratings for C1 and C2 are very similar. For this reason, only C1 is used for comparison. Non-parametric tests are run since not all data are normally distributed and there is no homogeneity of variances. A Friedman test performed on this data showed a significant effect (χ^2(10) = 73, p <0.001) over the eleven conditions.

Ten Sidak-corrected post-hoc Wilcoxon signed-rank tests were performed to compare conditions C3-C13 to C1, at a corrected significance level of p <0.0051. At this significance level conditions C3-C5 (Z <-3.464, p <0.001), C12 (Z = -3.103, p = 0.002) and C13 (Z = -3.258, p = 0.001) were found to differ significantly from condition C1.

Based on these results hypothesis H1A (PR_{min} = 0.6) is rejected, instead it is concluded that a value of 0.7 is more suitable. Hypothesis H1B (PR_{max} = 1.3) is rejected nor confirmed, as none of the two gains larger than unity were rated as too strong.

Based on the comparison of conditions C10-C13 with C1, hypothesis H1C ($\rho^2_{\text{LF,min}}$ = 0.8) is also rejected, instead a threshold value of 0.7 was found. It should be kept in mind that this value is only applicable in case the I/O coherence is determined by averaging signal coherence $\rho^2(s,t)$ over the 0.05-0.5 Hz frequency range.

B. Error discrimination

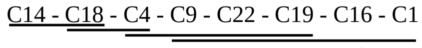
In order to determine which error types are best discriminated by the subjects, conditions C1, C4, C9, C14, C16, C18, C19 and C22 were compared. All but conditions C9 and C19 had equal maximum lateral specific force errors. Lateral specific force traces and average ratings for these conditions are shown in Fig. 11.

To compare these conditions, the maximum average rating (over four runs) per subject was determined, yielding 16 datapoints per condition. The distribution of these values is shown in Fig. 12.

Fig. 12 shows large variations in maximum ratings, especially for conditions C4 (an S1 error), C9 (an F1 error) and C22 (a P error). This finding matches an observation made during the training phase, during which subjects all indicated paying attention to different signal characteristics (scaling, timing, false roll rates, etc.). This different focus could explain some subjects rating particular errors and others not, explaining the increased spread in ratings. Conditions C1 and C2 are again rated consistently, indicating that the training phase provided subjects with a well-defined baseline.

Since the distribution for conditions C1 and C2 is again similar, only condition C1 is compared to other conditions. The requirements for parametric testing are not met for this data, hence again non-parametric tests are used. A Friedman test of the data showed a significant effect (χ^2(7) = 66, p <0.001) over the eight conditions.

To determine which error types were distinguished, a factorial post hoc Wilcoxon signed-rank test was performed on these eight conditions, yielding a Sidak-corrected significance level p <0.0018. At this significance level, all conditions not placed together on a line in the figure below are significantly different:



These results lead to the conclusion that F1, F3a and S1 errors should be identified separately at the very least, with other errors not differing significantly from condition C1 in terms of maximum rating. Due to the number of comparisons

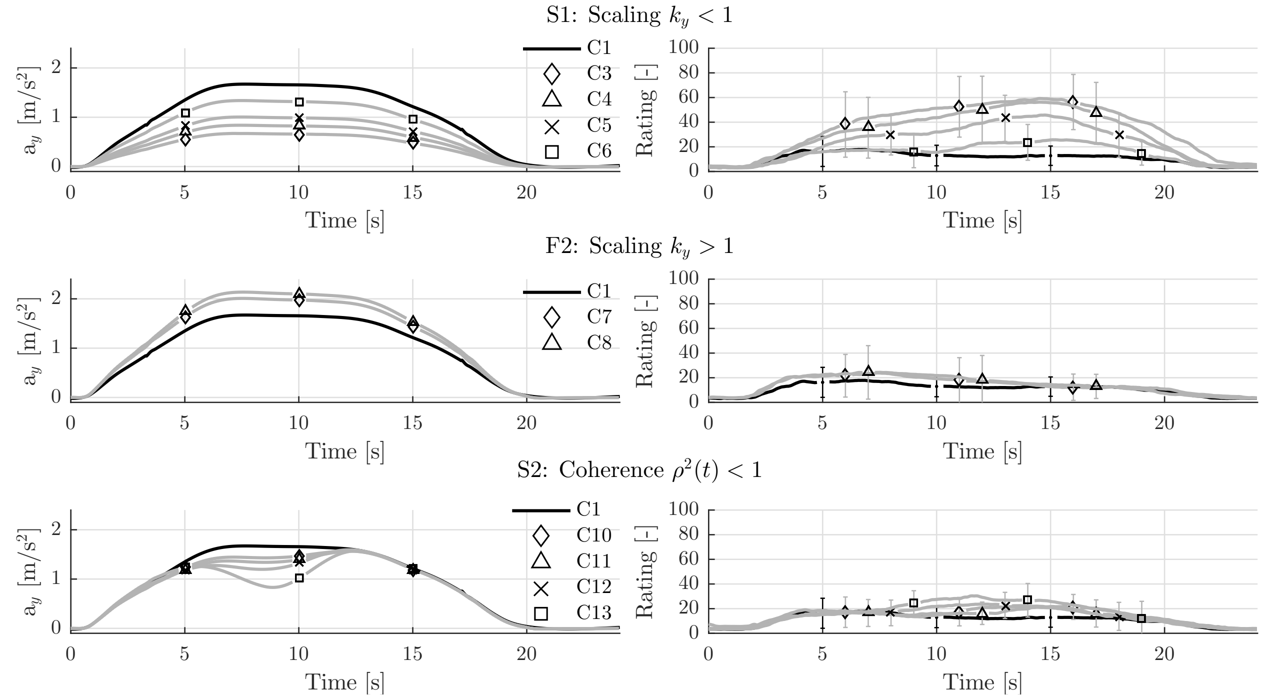


Fig. 8: Lateral specific force (left) and average ratings (right) for conditions C1, C3-C6, C7-C8 and C10-C13. Error bars indicate the confidence interval $\mu \pm 1\sigma$.

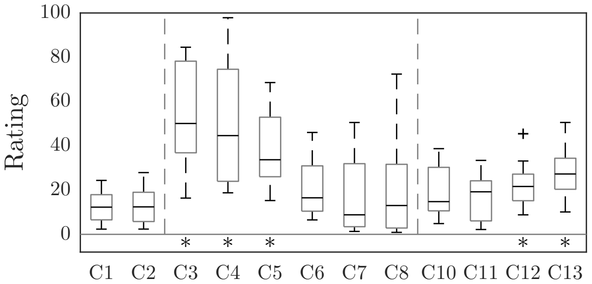


Fig. 9: Distribution of average ratings during the sustained part of the curve ($t = 9 - 15$ s). * indicates a significant difference (p <0.0051) compared to condition C1.

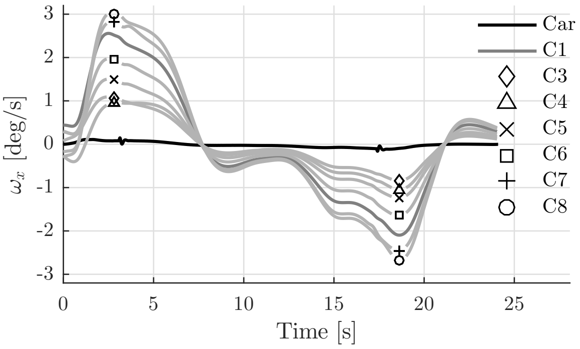


Fig. 10: Roll rate ω_x as experienced in the actual vehicle, compared to conditions C1 and C3-C8. Initial and final mismatches are caused by simulator washout of the previous condition.

(28), however, this conclusion is very conservative. In case no correction factor is used (p <0.05) the results are as follows:

C14 - C18 - C4 - C9 - C22 - C19 - C16 - C1

This comparison shows only S2 errors not to be noticed. Based on this comparison four clusters of errors should be identified: 1) F1 and F3a errors, 2) S1 errors, 3) S2, F3b and P errors and 4) F2 errors.

This finding might allow for some simplification of the algorithm, however, no cluster features error types differing on only one aspect of Table I. A different structure of the algorithm might better exploit this finding, however.

VII. DISCUSSION

In this paper, an algorithm capable of detecting various types of motion cueing errors has been developed with a view of improving motion cueing algorithms in future. The algorithm uses wavelet analysis to detect motion cueing error characteristics in both the time and frequency domain. A simulator experiment was performed to determine algorithm parameters and assess what error types are distinguished by humans. In the experiment, subjects were asked to rate the incongruence between visual and inertial motion during a curve-driving maneuver. In the following, the findings from these ratings will be discussed.

A. Threshold determination

Based on the ratings shown in Fig. 8 three parameter values for the algorithm were determined: PR_{min} = 0.7, PR_{max} >1.3 and $\rho^2_{\text{LF,min}}$ = 0.7.

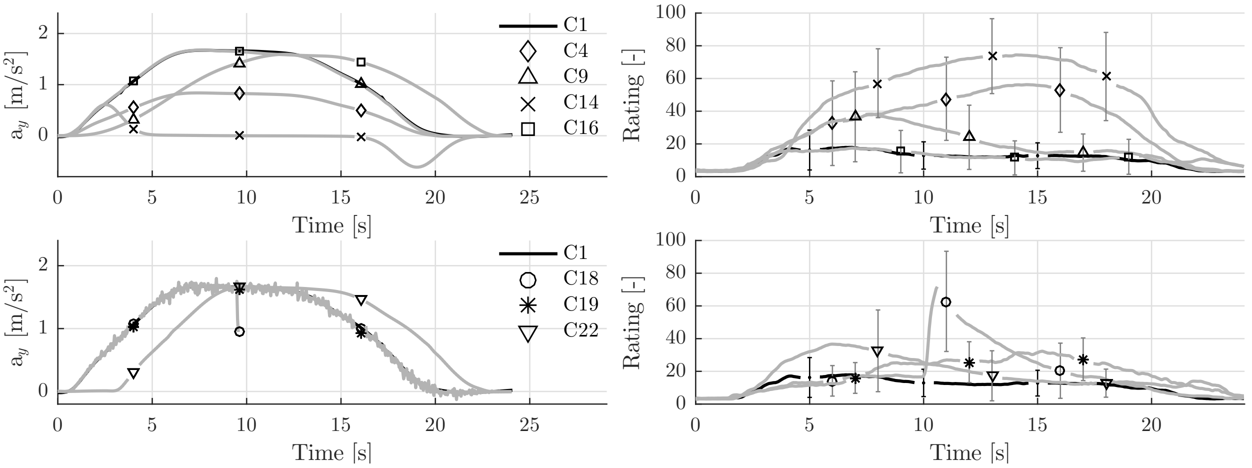


Fig. 11: Lateral specific force (left) and average ratings (right) for conditions C1, C4, C9, C14, C16, C18, C19 and C22. Error bars indicate the confidence interval $\mu \pm 1\sigma$.

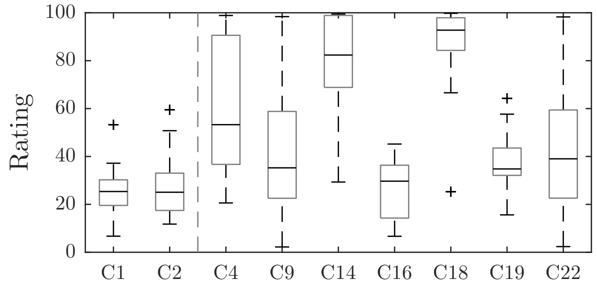


Fig. 12: Maximum average ratings for all motion cueing error types.

The values for both PR_{min} and PR_{max} are slightly larger than the values found by Correia Grácio *et al.* [37]. As such, an upward shift of the coherence zone was found compared to [37]. This finding is attributed to the fact that, unlike in [37], subjects had been given a reference condition in this experiment with a unity lateral specific force gain. The influence of this reference could also explain the fact that no significant increase in ratings is observed for the conditions with a lateral specific force gain larger than 1.

The values found for PR_{min} and PR_{max} in this experiment are valid for the lateral specific force in low-frequency curve driving maneuvers. Separate research is needed to investigate whether these parameters are still valid for higher-frequency cornering maneuvers (e.g., driving a slalom). Also, further experiments are needed to comment on these parameters for longitudinal and vertical motion, as well as rotational rates. For flight simulation, research into scaling of longitudinal motion and rotational rates is especially important, since aircraft usually turn in a coordinated fashion, featuring no (or very little) lateral specific sideforce.

Since longitudinal and lateral motion are usually simulated using the same MCA structure [1], the value found for $\rho^2_{\text{LF,min}}$ might also be valid for longitudinal motion. Further research is required to confirm this. Since vertical motion and rotational rates are usually high-pass filtered (either directly using filters or indirectly using a cost function), different values for $\rho^2_{\text{LF,min}}$ are expected for these motion channels. Further research is required to comment on $\rho^2_{\text{LF,min}}$ for these motion channels.

A final observation made from Fig. 8 is that conditions C3-C6 are only rated worse than condition C1 after t = 4.5 s. This finding may be explained by subjects' delay in noticing the scaled motion, as well as the reduced false roll rates for C3-C6 during the curve onset, as shown in Fig. 10. For condition C6 it could thus be that the benefits of a reduced false roll rate during curve onset offsets the effect of the scaled-down motion, explaining why subjects only increase their rating from t = 10 s onwards.

B. Error discrimination

Through a comparison of different motion cueing error types with a similar maximum lateral specific force error, four clusters of error types were identified: 1) F1 and F3a errors, 2) S1 errors, 3) S2, F3b and P errors and 4) F2 errors.

The worst ratings for F1 and F3a errors match the observation made by Grant & Reid [19], who note that false cues are the most detrimental to simulator fidelity. Fig. 11 shows the maximum rating of the F3a error (C18) occurs shortly after the disturbance. The maximum rating of the F1 error (C14), however, occurs when no motion is present, rather than when the motion presented is of opposite sign. This either suggests that not perceiving any motion is even worse than perceiving false motion (a statement contradicting the observation made by Valente Pais *et al.* [38]) or shows that subjects did not notice the motion to be of opposite sign.

S1 errors (C4), for which the coherence $\rho^2(t)$ remains high, were rated worse than other conditions for which the coherence dropped. This contradicts the finding by Casas *et al.* [15], who concluded that input/output coherence plays a large role in determining a subject's perception of a simulation, and both Grant *et al.* [17] and Fischer [18], who suggest it is better to scale down motion and not distort it than to not scale down motion but distort it.

From Fig. 11 one can conclude that S2 errors (C9) are rated worse than F2 errors (C16), this was also shown by the post

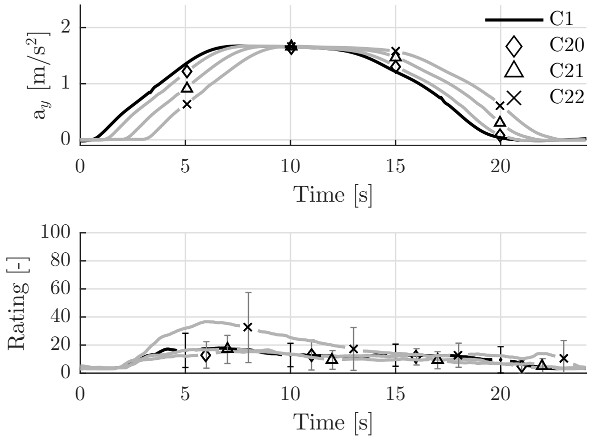


Fig. 13: Ratings (bottom) for the conditions featuring a delayed lateral specific sideforce (C20-C22). Error bars indicate the confidence interval $\mu \pm 1\sigma$ of ratings.

hoc analysis of the maximum ratings. The low ratings of the F2 error are surprising, since Valente Pais *et al.* [38] found these F2 errors at the end of curves to cause a classical washout filter to be rejected in favour of no motion. Cleij *et al.* [14] (who label S2 and F2 errors missing and false cues) also found F2 errors to be worse than S2 errors. In [38], however, 90° curves of smaller radius were driven in a shorter time. In [14], the curve onset and exit were also faster than in this study. Both [14] and [38] therefore investigated higher-frequency motion, which might explain the differences between the studies.

In the experiment carried out it was shown that a P error is rated equally bad as an S2 or F3b error, however, some caution should be exercised here. From Fig. 12 one can see that condition C22 is almost a combination of the error parts of C9 (an S2 error) and C16 (an F2 error), as such it is not surprising that the rating of C22 is equal to that of C9 initially and C16 near the end. For proper assessment of the ratings of a phase error, a slalom (featuring a delayed sinusoidal lateral specific force signal) should be simulated.

The ratings for all delayed conditions (C20-C22) are shown in Fig. 13. From this figure it can be concluded that subjects did not notice delays of at least up to 1.5 seconds, which is significantly higher than the delay of 175 ms (of visuals w.r.t. motion) reported by Beckers *et al.* [39] or the results in the order of 100 ms reported by Chung & Schroeder [47] (both for visuals leading the motion and vice versa). This finding may be explained by the fact that the lateral specific force profile used in this experiment was of very low frequency (its effective period is 40 s, so a delay of 1.5 s represents a 13.5° phase delay, which is below the value of $\Delta\phi_{\text{max}}$=16.4° reported by Beckers *et al.* [39]). Also, subjects reported having difficulty with determining exactly when the autopilot steered into the curve, meaning small phase differences might have been attributed to subjects' uncertainty in the timing of the curve onset.

C. Future applications

The motion cueing error detection algorithm developed in this paper can be used to improve motion cueing algorithms in the future. Rather than evaluating a simulation as a whole, the algorithm allows for focusing MCA tuning efforts on those errors most detrimental to simulator fidelity. In the paper, these were shown to be F1, F3a and S1 errors in case of curve driving. To comment on other car maneuvers (e.g., longitudinal motion) and flight simulation, further experiments would have to be performed to investigate the severity of error types in these situations.

In the specific case of a classical washout filter, this algorithm can be combined with the tuning paradigm of [12] to form a two-step tuning process, in which errors are first detected and then resolved. Should tuning paradigms be available for other MCA types, then the algorithm can be combined with these too. In this way, a step can be taken to optimizing MCAs, taking into account both motion cueing error magnitude error type.

VIII. CONCLUSION

In this paper an algorithm capable of detecting seven different motion cueing error types from a motion simulation's input and output signal was developed. This algorithm satisfies the need for objective motion cueing error detection (without assuming knowledge of the motion cueing algorithm used) and can be used to gain further insight into what motion characteristics are most important for simulator motion perception. Using wavelet analysis, five error characteristics (frequency, duration, scaling, coherence and phase difference) are analyzed in order to successfully distinguish between error types. The algorithm was tuned through a simulator experiment, showing that the lateral specific force during curve driving can be scaled down to 70% or scaled up to at least 130% without subjects reporting this. A threshold of 0.7 was also determined for input/output signal coherence. Finally, the need for error type recognition has been demonstrated by showing that different motion cueing error types of equal maximum error magnitude are rated differently. The algorithm developed in this paper allows MCA tuning efforts to be focused on parts of a simulation that are the most detrimental to simulator fidelity, allowing for motion cueing improvements in the future.

REFERENCES

[1] L. Reid and M. Nahon, "Flight simulation motion-base drive algorithms: Part 1 - Developing and testing the equations," University of Toronto Institute for Aerospace Studies, UTIAS Report, no. 296, 1986.

[2] M. Fischer, "A survey of state-of-the-art motion platform technology and motion cueing algorithms," in *2nd Motion Simulator Conference, Braunschweig, Germany, August, 2007*, 2007.

[3] K. Stahl, G. Abdulsamad, K. D. Leimbach, and Y. A. Vershinin, "State of the art and simulation of motion cueing algorithms for a six degree of freedom driving simulator," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Oct 2014, pp. 537–541.

[4] S. Schmidt and B. Conrad, "Motion drive signals for piloted flight simulators," National Aeronautics and Space Administration, Ames Research Center, Technical report contract NASA CR-1601, 1970.

[5] I. Salisbury and D. Limebeer, "Optimal motion cueing for racing cars," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 1, pp. 200–215, 2016.

[6] J. Venrooij, P. Pretto, M. Katliar, S. Nooij, A. Nesti, M. Lächele, K. de Winkel, D. Cleij, and H. Bülthoff, "Perception-based motion cueing: Validation in driving simulation," in *DSC 2015 Europe: Driving Simulation Conference & Exhibition, Tübingen, Germany, Sep. 16-18, 2015*. Max Planck Institute for Biological Cybernetics, 2015.

[7] J. Schroeder, "Helicopter flight simulation motion platform requirements," Stanford University. Dept. of Aeronautics and Astronautics, Tech. Rep. NASA/TP-1999-208766, July 1999.

[8] J. Sinacori, "The determination of some requirements for a helicopter flight simulation facility," NASA, Tech. Rep. CR-152066, 1977.

[9] S. Advani and R. Hosman, "Towards standardising high-fidelity cost-effective motion cueing in flight simulation," in *Royal Aeronautical Society Conference on: Cutting Costs in Flight Simulation. Balancing Quality and Capability. London, Nov. 7-8, 2006*, 2006.

[10] A. Valente Pais, M. van Paassen, and M. Mulder, "Perception coherence zones in flight simulation," in *Proceedings of the AIAA Modeling and Simulation Technologies Conference, Chicago, Illinois, Aug. 10-13, 2009*. American Institute of Aeronautics and Astronautics, 2009.

[11] C. Nash, D. Cole, and R. Bigler, "A review of human sensory dynamics for application to models of driver steering and speed control," *Biological Cybernetics*, vol. 110, no. 2, pp. 91–116, 2016.

[12] P. Grant, "The development of a tuning paradigm for flight simulator motion drive algorithms," PhD Thesis, University of Toronto - Department of Aerospace Science and Engineering, Canada, 1995.

[13] R. Hosman and S. Advani, "Design and evaluation of the objective motion cueing test and criterion," *The Aeronautical Journal*, vol. 120, no. 1227, p. 873891, 2016.

[14] D. Cleij, J. Venrooij, P. Pretto, D. Pool, M. Mulder, and H. Bülthoff, "Continuous subjective rating of perceived motion incoherence during driving simulation," *IEEE Transactions on Human-Machine Systems*, vol. Submitted for publication.

[15] S. Casas, C. Inmaculada, J. Riera, and M. Fernández, "Motion-cuing algorithms: Characterization of users perception," *Human Factors*, vol. 57, no. 1, pp. 144–162, 2015.

[16] D. Cleij, J. Venrooij, P. Pretto, D. Pool, M. Mulder, and H. Bülthoff, "Continuous rating of perceived visual-inertial motion incoherence during driving simulation," in *Driving simulation conference & exhibition, Tübingen, Germany, Sep. 16-18, 2015*. Max Planck Institute, 2015.

[17] P. Grant, M. Blommer, B. Artz, and J. Greenberg, "Analysing classes of motion drive algorithms based on paired comparison techniques," in *Proceedings of the Driving Simulation Conference (DSC) North America, Dearborn, USA, Oct. 8-10, 2003*. University of Toronto & Ford Motor Company, 2003.

[18] M. Fischer, "Motion-cueing-algorithmen für eine realitätsnahe bewegungssimulation," PhD Thesis, Technische Universität Braunschweig - Fakultät für Maschinenbau, Germany, 2009.

[19] P. Grant and L. Reid, "Motion washout filter tuning: Rules and requirements," *Journal of Aircraft*, vol. 34, no. 2, pp. 145–151, 1997.

[20] J. Venrooij, D. Cleij, M. Katliar, P. Pretto, H. Bülthoff, D. Steffen, F. Hoffmeyer, and H.-P. Schöner, "Comparison between filter- and optimization-based motion cueing in the daimler driving simulator," in *DSC 2016 Europe: Driving Simulation Conference & Exhibition, Paris, France, Sep. 1-8, 2016*, 2016.

[21] A. Nesti, C. Masone, M. Barnett-Cowan, R. Robuffo Giordano, H. Bülthoff, and P. Pretto, "Roll rate thresholds and perceived realism in driving simulation," in *DSC 2012 Europe: Driving Simulation Conference & Exhibition, Paris, France, Sep. 6-7, 2012*. Max Planck Institute for Biological Cybernetics, 2012.

[22] A. Berthoz, W. Bles, H. Bülthoff, B. Correia Grácio, P. Feenstra, N. Filliard, R. Hühne, A. Kemeny, M. Mayrhofer, M. Mulder, H. Nusseck, P. Pretto, G. Reymond, R. Schlüsselberger, J. Schwandtner, H. Teufel, B. Vailleau, M. van Paassen, M. Vidal, and M. Wentink, "Motion Scaling for High-Performance Driving Simulators," *IEEE Transactions on Human-Machine Systems*, vol. 43, no. 3, pp. 265–276, 2013.

[23] E. Groen, M. Valenti Clari, and R. Hosman, "Evaluation of perceived motion during a simulated takeoff run," *Journal of Aircraft*, vol. 38, no. 4, pp. 600 – 606, 2001.

[24] E. Sejdić, I. Djurović, and J. Jiang, "Timefrequency feature representation using energy concentration: An overview of recent advances," *Digital Signal Processing*, vol. 19, no. 1, pp. 153 – 183, 2009.

[25] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Transactions on Information Theory*, vol. 36, no. 5, pp. 961–1005, Sep 1990.

[26] C. Torrence and G. Compo, "A practical guide to wavelet analysis," *Bulletin of the American Meteorological Society*, vol. 79, no. 1, pp. 61–78, 1998.

[27] G. Cooper and D. Cowan, "Comparing time series using wavelet-based semblance analysis," *Computers & Geosciences*, vol. 34, no. 1, pp. 95–102, 2008.

[28] O. Jonah, E. de Paula, M. Muella, S. Dutra, E. Kherani, P. Negreti, and Y. Otsuka, "TEC variation during high and low solar activities over South American sector," *Jounal of Atmospheric and Solar-Terrestrial Physics*, vol. 135, no. 1, pp. 22–35, 2015.

[29] A. Valle-Levinson, L. Castro, M. Cáceres, and O. Pizarro, "Twilight vertical migrations of zooplankton in a Chilean fjord," *Progress in Oceanography*, vol. 129, pp. 114–124, 2014.

[30] M. Yochum, C. Renaud, and S. Jacquir, "Automatic detection of P, QRS and T patterns in 12 leads ECG signal based on CWT," *Biomedical Signal Processing and Control*, vol. 25, pp. 46–52, 2016.

[31] G. Papaioannou, C. Dikaiakos, G. Evangelidis, P. Papaioannou, and D. Georgiadis, "Co-movement analysis of Italian and Greek electricity market wholesale prices by using a wavelet approach," *Energies*, vol. 8, no. 10, pp. 11 770–11 799, 2015.

[32] G. Bergland, "A guided tour of the fast fourier transform," *IEEE Spectrum*, vol. 6, no. 7, pp. 41–52, July 1969.

[33] A. Grinsted, J. Moore, and S. Jevrejeva, "Application of the cross wavelet transform and wavelet coherence to geophysical time series," *Nonlinear processes in geophysics*, vol. 11, pp. 561–566, 2004.

[34] C. Torrence and P. Webster, "Interdecadal changes in the ensomonsoon system," *Journal of Climate*, vol. 12, no. 8, pp. 2679–2690, 1999.

[35] J. Bendat and A. Piersol, *Random Data: Analysis and Measurement Procedures*, ser. Wiley Series in Probability and Statistics. Wiley, 2011.

[36] H. Heerspink, W. Berkouwer, O. Stroosma, M. van Paassen, M. Mulder, and J. Mulder, "Evaluation of vestibular thresholds for motion detection in the Simona Research Simulator," in *AIAA Modeling and Simulation Technologies Conference and Exhibit, San Francisco, California, Aug. 15-18, 2005*. American Institute of Aeronautics and Astronautics, 2005.

[37] B. Correia Grácio, A. Valente Pais, M. van Paassen, M. Mulder, L. Kelly, and J. Houck, "Optimal and coherence zone comparison within and between flight simulators," *Journal of Aircraft*, vol. 50, no. 2, pp. 493–507, 2013.

[38] A. Valente Pais, M. Wentink, M. van Paassen, and M. Mulder, "Comparison of three motion cueing algorithms for curve driving in an urban environment," *Presence*, vol. 18, no. 3, pp. 200–221, 2009.

[39] N. Beckers, D. Pool, A. Valente Pais, M. van Paassen, and M. Mulder, "Perception and behavioral phase coherence zones in passive and active control tasks in yaw," in *Modeling and simulation technologies conference, Minneapolis, Minnesota, Aug. 13-16, 2012*. American Institute of Aeronautics and Astronautics, 2012.

[40] T. Verspecht, "Identification of Time Variant Neuromuscular Admittance," Masters Thesis, Delft University of Technology - Faculty of Aerospace Engineering, Kluyverweg 1, Delft, The Netherlands, 2011.

[41] H. Teufel, H.-G. Nusseck, K. Beykirch, J. Butler, M. Berger, and H. H. Bülthoff, "MPI Motion Simulator: Development and Analysis of a Novel Motion Simulator," in *Proceedings of the AIAA Modeling and Simulation Technologies Conference and Exhibit, Hilton Head, South Carolina, Aug. 20-23, 2007*. American Institute of Aeronautics and Astronautics, 2007.

[42] C. Masone, P. Robuffo Giordano, and H. Bülthoff, "Mechanical design and control of the new 7-dof cybermotion simulator," in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 4935–4942.

[43] R. Kennedy, N. Lane, K. Berbaum, and M. Lilienthal, "Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness," *The International Journal of Aviation Psychology*, vol. 3, no. 3, pp. 203–220, 1993.

[44] A. Jamson, "Motion cueing in driving simulators for research applications," PhD Thesis, The University of Leeds - Institute for Transport Studies, United Kingdom, 2010.

[45] D. Berger, J. Schulte-Pelkum, and H. Bülthoff, "Simulating believable forward accelerations on a Stewart motion platform," *ACM Transactions on Applied Perception*, vol. 7, no. 1, 2010.

[46] L. Bringoux, V. Nougier, P.-A. Barraud, L. Marin, and C. Raphel, "Contribution of somesthetic information to the perception of body orientation in the pitch dimension." *The Quarterly journal of experimental psychology. A, Human experimental psychology*, vol. 56, no. 5, pp. 909–23, 2003.

[47] W. Chung and J. Schroeder, "Visual and roll-lateral motion cueing synchronization requirements for motion-based flight simulations," vol. 2, 1997, pp. 994–1006.

[48] A. Teolis, *Computational Signal Processing with Wavelets*, ser. Applied and Numerical Harmonic Analysis. Cambridge, MA 02139, USA: Birkhäuser Boston, 1999.

[49] D. Lee and A. Yamamoto, "Wavelet analysis: theory and applications," *Hewlett-Packard Journal*, vol. 45, no. 6, pp. 44–54, December 1994.

[50] M. Misiti, Y. Misiti, G. Oppenheim, and J. Poggi, *Wavelets and their Applications*, ser. ISTE. Wiley, 2013.

[51] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1992.

[52] D. Gilman, F. Fuglister, and J. Mitchell, "On the power spectrum of red noise," *Journal of the Atmospheric Sciences*, vol. 20, pp. 182–184, 1962.

APPENDIX A
WAVELET FAMILIES & PARAMETERS

Wavelet analysis can be carried out using various wavelet families, all satisfying the requirements of (3) and (4). Several overviews of these families already exist, see for example [48]–[50]. These overviews, however, do not comment on the effects of changes to wavelet parameters. An overview including these influences is discussed here.

Wavelet families can be broken down along two axes: they are either time- or frequency-based and are either real or complex. Frequency-based wavelets offer the advantage of faster computation through (6), however, there is less variation between different wavelet families than there is between time-based wavelet families. For the first reason, only frequency-based wavelets are discussed by Torrence & Compo [26].

In contrast to real wavelets, the inner product of complex wavelets with a harmonic function (as computed in (5)) is never zero. This means that complex wavelets produce continuous non-zero output over time for such a function, whereas a convolution with a real wavelet oscillates, producing an inner product of zero in the process. Real-valued wavelets therefore can not indicate signal power continuously.

An overview of common wavelets is given in Table V. Time descriptions of the mother wavelet (not including the scale factor s) are given, these are also shown in Fig. 14. Frequency descriptions (if known) are given too, again for an unscaled mother wavelet.

For wavelet families for which this was possible, a modulating (or bounding) function in time has been given. This function will be used for smoothing in time when determining wavelet coherence, as explained in Appendix C. Finally, Table V also indicates the effects of parameter changes to a wavelet's frequency resolution (FR, a wavelet's ability to distinguish signal properties in the frequency domain) and time resolution (TR, a wavelet's ability to distinguish signal properties in the time domain). Note that these changes are mostly modest compared to switching wavelet family. Of the wavelets listed, the Haar and Daubechies wavelets provide good TR but poor FR (due to their limited width in time and non-smooth behavior). The Shannon and Frequency B-Spline wavelet offer the best FR of the wavelets listed here, due to their large width in the time domain and compact support in the frequency domain. All other wavelet families offer a trade-off between TR and FR.

APPENDIX B
WAVELET POST-PROCESSING

Due to wavelets' limited time and frequency resolution, wavelet spectra are typically characterized by power spread.

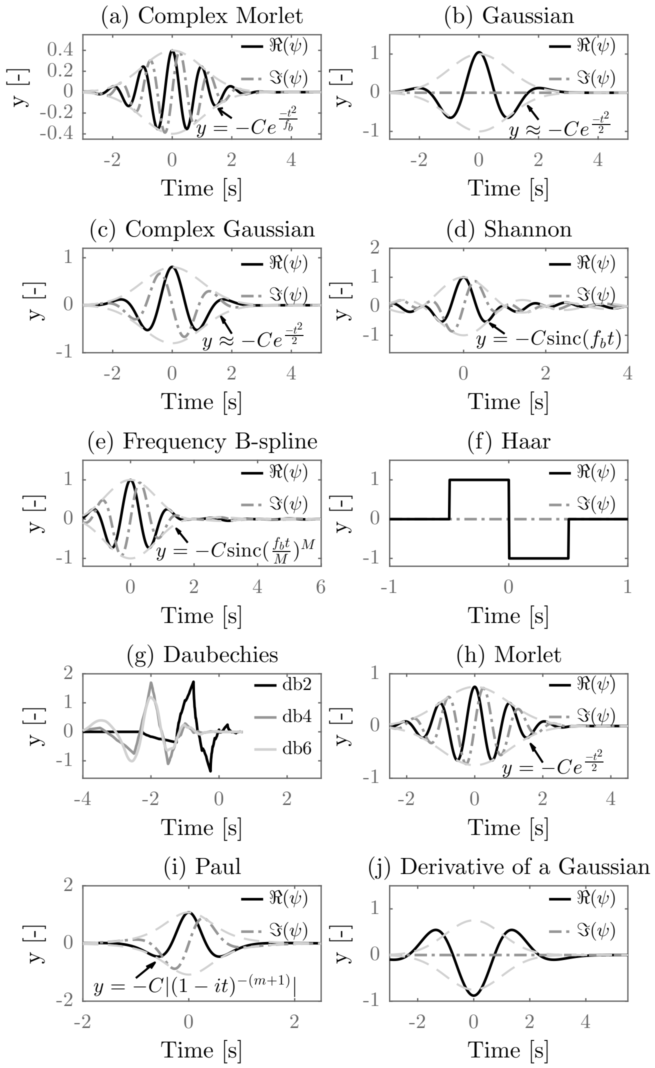


Fig. 14: Real (solid) and imaginary (dashed) parts of the mother wavelets of ten wavelet families. From left to right, top to bottom: The time-based complex Morlet wavelet (f_b = 2, f_c = 1 Hz), Gaussian wavelet (n = 4), complex Gaussian wavelet (n = 4), Shannon wavelet (f_b = 1 Hz, f_c = 0.8 Hz), Frequency B-Spline wavelet (f_b = 1 Hz, f_c = 0.8 Hz, M = 2), Haar wavelet and Daubechies wavelets, followed by the frequency-based Morlet wavelet (ω_0 = 6 rad/s), Paul wavelet (m = 4) and Derivative of a Gaussian wavelet (n = 4). Modulating functions are shown if available.

Two ways of dealing with this spread are discussed in this section, that is, dealing with edge effects and significance testing.

A. Edge effects

The inner product of a wavelet centered at time t_1 and a function $f(t)$ is based on values of $f(t)$ on the interval $[t_1 - t_{\text{inf}}, t_1 + t_{\text{inf}}]$, where t_{inf} denotes the time it takes the wavelet to decay to 0. If t_1 is less than t_{inf} away from the edge of the domain of $f(t)$, then the inner product is computed based either on values outside of the domain (assumed zero),

TABLE V: Time description, frequency description, modulating function and parameter effects for time- and frequency-based mother wavelets. Constants C ensure unit energy. FR denotes Frequency Resolution, TR denotes Time Resolution, DOG denotes Derivative of a Gaussian. $H(\omega)$ indicates the Heaviside step function. Details taken from [26], [48], [51].

Wavelet	**Function in time**	**Function in frequency**	**Modulating wave**	**Parameter effects**
Time-based wavelets				
Complex Morlet	$\psi_M(t) = \frac{1}{\sqrt{(\pi f_b)}} \cdot e^{2\pi i f_c t} \cdot e^{-\frac{t^2}{f_b}}$	$\Psi_M(\omega) = e^{-\pi^2 f_b(\omega - f_c)^2}$	$\pm C \cdot e^{-t^2/f_b}$	$f_b^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $f_b^{\uparrow} \rightarrow$ TR$^{\downarrow}$ $f_c^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $f_c^{\uparrow} \rightarrow$ TR$^{\downarrow}$
Gaussian	$\psi_M(t) = \frac{\mathrm{d}^n}{\mathrm{d}t^n}\left(\mathrm{C}e^{-t^2}\right)$	–	$\approx \pm C e^{-t^2/2}$	$n^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $n^{\uparrow} \rightarrow$ TR$^{\downarrow}$
Complex Gaussian	$\psi_M(t) = \frac{\mathrm{d}^n}{\mathrm{d}t^n}\left(\mathrm{C}e^{-it} \cdot e^{-t^2}\right)$	–	$\approx \pm C \cdot e^{-t^2/2}$	$n^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $n^{\uparrow} \rightarrow$ TR$^{\downarrow}$
Shannon	$\psi_M(t) = \sqrt{f_b} e^{2\pi i f_c t} \mathrm{sinc}(f_b t)$	–	$\pm C \cdot \mathrm{sinc}(f_b t)$	$f_b^{\uparrow} \rightarrow$ FR$^{\downarrow}$ & $f_b^{\uparrow} \rightarrow$ TR$^{\uparrow}$ $f_c^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $f_c^{\uparrow} \rightarrow$ TR$^{\downarrow}$
Frequency B-spline	$\psi_M(t) = \sqrt{f_b} e^{2\pi i f_c t} \left[\mathrm{sinc}(\frac{f_b t}{M})\right]^M$	–	$\pm C \left[\mathrm{sinc}(\frac{f_b t}{M})\right]^M$	$f_b^{\uparrow} \rightarrow$ FR$^{\downarrow}$ & $f_b^{\uparrow} \rightarrow$ TR$^{\uparrow}$ $f_c^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $f_c^{\uparrow} \rightarrow$ TR$^{\downarrow}$ $M^{\uparrow} \rightarrow$ FR$^{\downarrow}$ & $M^{\uparrow} \rightarrow$ TR$^{\uparrow}$
Haar	$\psi_M(t) = 1_{(-\frac{1}{2}, 0]} - 1_{(0, \frac{1}{2}]}$	$\Psi_M(\omega) = 2i \frac{\sin^2(\frac{\pi\omega}{2})}{\pi\omega}$	–	–
Daubechies	See [51]	–	–	$n^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $n^{\uparrow} \rightarrow$ TR$^{\downarrow}$
Frequency-based wavelets				
Morlet	$\psi_M(t) = \pi^{1/4} e^{i\omega_0 t} e^{-t^2/2}$	$\Psi_M(\omega) = \pi^{1/4} \cdot \ldots$ $H(\omega) e^{-(\omega - \omega_0)^2/2}$	$\pm C e^{-t^2/2}$	$\omega_0^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $\omega_0^{\uparrow} \rightarrow$ TR$^{\downarrow}$
Paul wavelet	$\psi_M(t) = \frac{2^m (i)^m (m)!}{\sqrt{\pi(2m)!}} (1 - it)^{-(m+1)}$	$\Psi_M(\omega) = \frac{2^m}{\sqrt{m(2m-1)!}} \cdot \ldots$ $H(\omega)(\omega)^m e^{-\omega}$	$\pm C \left\lvert (1 - it)^{-(m+1)} \right\rvert$	$m^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $m^{\uparrow} \rightarrow$ TR$^{\downarrow}$
DOG wavelet	$\psi_M(t) = \frac{(-1)^{n+1}}{\sqrt{\Gamma(n + \frac{1}{2})}} \frac{\mathrm{d}^n}{\mathrm{d}t^n}\left(e^{-t^2/2}\right)$	$\Psi_M(\omega) = \frac{(-i)^n}{\sqrt{\Gamma(n + \frac{1}{2})}} \cdot \ldots$ $(\omega)^n e^{-(\omega)^2/2}$	–	$n^{\uparrow} \rightarrow$ FR$^{\uparrow}$ & $n^{\uparrow} \rightarrow$ TR$^{\downarrow}$

or based on a repeated version of $f(t)$ outside the domain. Both methods introduce errors in the WT and, as such, WT values near the edge of the domain of $f(t)$ should be treated with care.

Torrence & Compo name this region in which care should be taken the Cone of Influence (COI), defining it as *"the e-folding time for the autocorrelation of wavelet power ($|W(s,t)|^2$) at each scale"* [26], which is equal to *"the area in which the wavelet power caused by a discontinuity at the edge has dropped to e^{-2} of the value at the edge"* [33].

Two problems exist with this definition. Firstly, it does not hold for real symmetric wavelets, such as the derivative of a Gaussian (DOG) wavelets with even order n, since the inner product of a step function at t = 0 s and such a wavelet centered at t = 0 s is zero. Secondly, it can be shown that varying m for a Paul wavelet changes the COI, the expression in [26], however, is independent of m.

Torrence & Compo [26] provide expressions for the COI width as a function of scale only for frequency-based wavelets. In this paper an extended method suitable for all wavelet families is presented, based on a wavelet's distribution of energy. Since (4) holds for all wavelets, wavelet energy allows for a natural way of determining on what interval a wavelet contains most of its energy. By checking whether the function $f(t)$ is defined on this interval, a Cone of Influence can be determined.

First, the two time points between which 95% of wavelet energy is contained (recall that total wavelet energy is 1 as per (3)) are determined and are designated $t_{0.025}$ and $t_{0.975}$. Next the influence time t_{inf} of the wavelet is determined using:

$$t_{\text{inf}} = \frac{t_{0.975} - t_{0.025}}{2} \tag{17}$$

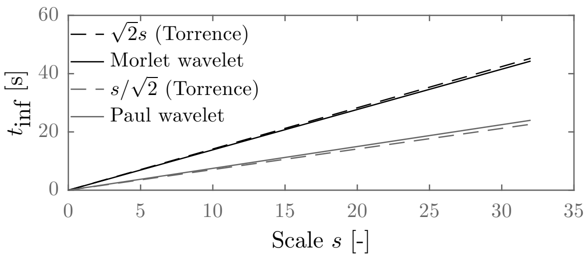


Fig. 15: Comparison of the influence time t_{inf} found by Torrence & Compo (dotted lines) and the methodology of this paper (solid lines), for the frequency-based complex Morlet wavelet with ω_0 = 4 rad/s (black) and Paul wavelet with m = 4 (gray).

The inner product domain then becomes $[t_1 - t_{\text{inf}}, t_1 + t_{\text{inf}}]$. If the function $f(t)$ is defined on this entire interval the wavelet transform $W(s, t_1)$ is valid, if not then $W(s, t_1)$ is designated part of the COI. Using this approach produces results very similar to the $\sqrt{2}s$ and $s/\sqrt{2}$ values for t_{inf} found by Torrence & Compo [26] for the frequency-based complex Morlet and Paul wavelets respectively (see Fig. 15). The approach presented here can be extended to any wavelet.

B. Significance testing

In order to discriminate between parts of the WT containing genuine power and parts of the WT influenced by signal spread, Torrence & Compo [26] propose a significance check on the wavelet spectrum. The null hypothesis of this check is that the analyzed signal originates from a white or red noise process, with a data point x_n in this noise given by:

$$x_n = \alpha x_{n-1} + z_n \tag{18}$$

In (18), α is the lag-1 autocorrelation coefficient, which for simulator signals is near 1 since the next data point is the sum of the previous and some small change. For this reason α is set to 0.9. The noise input z_n is white noise. For a red noise signal the normalized Fourier power spectrum is given by [52]:

$$P_k = \frac{\sigma^2(1-\alpha^2)}{1+\alpha^2-2\alpha\cos(2\pi k/N)} \tag{19}$$

In this equation, σ^2 is the variance of the signal $f(t)$ which was wavelet transformed, and k is the frequency index used in the Fast Fourier Transform of the input signal, with N the total number of data points. A piece of the WT spectrum is significant in case the WT power ($|W(s,t)|^2$) in a time-slice of this WT is significantly larger than this background spectrum. The reader is referred to Torrence & Compo [26] for a discussion showing that this confidence value can be obtained from a χ^2-distribution with either two Degrees of Freedom (DoF) (for complex wavelets) or one DoF (for real wavelets). As a result, WT-spectra regions are significant in case (n_{df} denotes the number of DoFs):

$$|W(s,t)|^2 \geq \frac{\chi^2_{n_{df}}(0.95)}{n_{df}} \cdot P \tag{20}$$

Appendix C
Wavelet coherence extension

Determination of the coherence between two signals based on wavelet analysis is done using (8), (10) and (11). Grinsted, Moore and Jevrejeva describe $S(\cdot)$ in (10) and (11) as a smoothing function in both time and frequency [33]. In [33] an expression for $S(\cdot)$ is only available for the frequency-based Morlet wavelet. In this appendix, an extension is made to all wavelet families.

According to [34], the smoothing in time for a (frequency-based) Morlet wavelet is done by applying a filter $e^{\frac{-t^2}{2s^2}}$ to the wavelet spectrum $W(s,t)$. The latter expression is the modulating function $m(t/s)$ of the Morlet wavelet for scale s = 1, see Table V. This means that smoothing in time is done by convoluting $W(s,t)$ with the (absolute) modulating function of the wavelet used to determine $W(s,t)$. To extend the method of [33] to all wavelets, further modulating functions $m(t/s)$ of wavelet families have been given in Table V. It should be noted that the constants C, which ensure unit wavelet energy, drop out of (10) and (11) and can thus be omitted.

Alternatively, one can multiply $W(s,t)$ with the Fourier Transform of the (absolute) modulating function (if available) for faster computation. For the time-based complex Morlet wavelet this modulating wave $M(s\omega)$ is $\mathrm{C}e^{-f_b(s\omega)^2/4}$, for the Gaussian, complex Gaussian and frequency-based Morlet wavelet this is $\mathrm{C}e^{-(s\omega)^2/2}$.

For wavelet families for which only a time-description of $|m(t/s)|$ is known (or for which no modulating function is available) the wavelet spectrum $W(s,t)$ is convoluted with $|m(t/s)|$, if available, or otherwise $|\psi(t/s)|$, with $\psi(t/s)$ denoting the wavelet time description including wavelet scaling.

According to [34], smoothing in frequency is done by using a moving average filter over all frequencies. For this paper this approach was copied. A moving average filter measuring four scales (and hence frequencies) wide was used.

Appendix D
List of Symbols

Greek Symbols

α	Significance level [-]
α	Lag-1 red noise autocorrelation coefficient [-]
Δ	Difference operator
ζ	Damping ratio [-]
μ	Average
ρ^2	(Wavelet) Coherence [-]
σ	Signal standard deviation
τ	Time shift of a wavelet [s]
Φ	Perceptual threshold for acceleration or rotation
ϕ	Signal phase (difference) [deg]
Ψ	Fourier transform of wavelet function
ψ	Analyzing function
ψ	(Daughter) Wavelet function
ω	Rotational rate [deg/s]
ω	Frequency [rad/s]
ω_0	Complex Morlet wavelet frequency at a scale of 1 [rad/s]

Roman Symbols

a	Specific force [m/s^2]
C	Constant
E	Wavelet energy
$\mathcal{F}$	Fourier transform operator
F	Fourier transform of time signal f
f	Time signal
f_b	Complex morlet, Shannon and Frequency B-spline modulating wave decay parameter [-]
f_c	Complex morlet, Shannon and Frequency B-spline wavelet carrier wave frequency parameter [Hz]
H	Heaviside step function
H	Transfer function
i	Unit imaginary number
k	Simulator input scaling factor [-]
k	Frequency index
M	Modulating function Fourier transform
M	Sinc-function power for the Frequency B-spline wavelet
m	Modulating function
m	Paul wavelet parameter
N	Total number of data points [-]
n	Daubechies wavelet number
n	Derivative order for a (complex) Gaussian wavelet
n	Number of
P	Fourier power spectrum
PR	Power Ratio
p	p-value of statistical test
S	Scaling function in time and frequency
S	Power spectral density
s	Scale factor of a daughter wavelet [-]
s	Laplace variable
t	Time [s]

W	Wavelet transform
x	Time signal
Z	Standard score of statistical test
z	White noise signal

Subscripts

df	Degree(s) of freedom
HP	High-pass
inf	Influence
lim	Limit
LP	Low-pass
max	Maximum value
min	Minimum value
M	Mother
n	Natural
n	Index number
X	Signal X
x	Longitudinal direction
Y	Signal Y
y	Lateral direction
z	Vertical direction

Appendices

Appendix A

Experiment Instructions

Prior to the simulator experiment, subjects were instructed both through an instruction form and through verbal explanation. The instruction form can be found on the next three pages. Subjects first read through this instruction form and were given additional verbal explanation afterwards. During the verbal instruction, the following points were emphasised:

- Subjects were told that during the experiment, the motion they felt would not always correspond to the motion they saw. Subjects were reassured that, should these mismatches cause severe sickness symptoms, subjects were free to end the experiment whenever they wished to.
- Subjects were explained that they would experience a passive car drive, meaning an automatic driver would steer the vehicle. Subjects were told that the driving style of this automatic driver might differ from their own, and that for this reason a familiarisation run (without rating) would be run first.
- Subjects were instructed that this familiarisation run would serve as a reference, and that all changes in motion felt thereafter could be attributed to the experimenter (and not to changes in road conditions, weather conditions, etc.).
- Subjects were explained the rating scale during the experiment, and were explained the purpose of the two training runs. These were:
  - In the first run, subjects were instructed to focus on what parts of the drive felt best (this would define their minimum rating) and what part felt worst (this would define their maximum rating). Subjects were told that after this run, they would be asked to describe the worst motion they experienced.
  - In the second run, subjects would practise assigning ratings to their perceived visual-motion mismatch on the scale developed in the first run.
  - Subjects were told they could change their rating scale during the second training run, should they encounter motion that felt worse than any motion of training

run 1. This, however, was not to be expected, since both runs featured identical motion. Subjects were also instructed not to change their rating scale once the training phase had ended.

Subjects were asked to briefly comment on each condition during the training runs, such that the experimenter could assess whether they were paying attention to the correct features (i.e., to check whether they rated the motion, rather than for example the quality of the visual images or the driving style of the driver). This request was dropped prior to the first experiment run.

- Subjects were instructed to *always* provide a rating for a feature they felt, even if realising something was wrong had taken multiple seconds (i.e., a delayed rating was preferred over no rating).

Experiment instructions

Aim of the experiment

The goal of this experiment is to investigate the performance of different settings of the CyberMotion Simulator during curve driving. We will do so by asking you how well the inertial motion presented by the CyberMotion Simulator feels compared to what you are seeing.

Your task in this session

You are sitting in a virtual car and will experience four ten-minute driving scenarios, in which your car first accelerates from 0 to 50 kilometres an hour, then drives through 22 90^0 curves (either left-hand or right-hand) and comes to a standstill afterwards. Each drive will take around ten minutes.

In order to ensure you can fully concentrate on the inertial motion you feel and the visual motion you see you do not have to drive or steer the car yourself, instead the car is driven automatically. The autopilot of the vehicle has been set such that the distance travelled is minimised, as a result the car drives along the inner curve in each corner. In order to do so the autopilot 'steers into' the corner earlier than you might do when driving yourself. In the absence of a steering wheel (which would indicate the autopilot starting a turn), a line has been drawn on the road to use as a lateral reference.

An illustration of your vehicle and the typical view in a curve are shown below.

Figure 1 *Your car and the surrounding landscape*

Figure 2 *View of the track (car interior not shown)*

As a result of the different experiment conditions the motion you feel might not always fully correspond to what you see. You will be asked to rate this incoherence between the simulator motion and the visual images (taking the latter as reference) during each run. You can indicate the incoherence between motion and visuals *at that moment* by turning a knob placed in the cabin with your right hand. The current rating value of this knob is indicated on your screen through a blue rating bar. An illustration of this rating bar is shown below.

Figure 3 - *Illustration of your rating scale during the experiment*

There are various reasons which might lead you to rate motion as being incoherent, reasons include (but are not limited to, anything which doesn't feel 'right' for you should be rated):

- ✓ The motion presented feels too weak compared to the visuals, or feels too strong
- ✓ The direction of the motion you feel is wrong compared to the visuals
- ✓ The motion is presented too late compared to the visuals, or too early
- ✓ You feel motion, but you weren't expecting any based on the visuals
- ✓ You don't feel motion when you were expecting motion based on the visuals

Please indicate motion which feels coherent (i.e. as you would expect) by turning the knob to the left/CCW (so all blue dots in the previous figure disappear), and indicate motion which feels highly incoherent with what you are seeing by turning the knob to the right/CW (fully filling the bar with blue dots). A rating of zero (maximally CCW) indicates perfect coherence, a maximum rating indicates the worst motion you have encountered throughout the trial. Before starting a trial, turn the knob fully CCW (to zero).

Training / calibration

Since the driving behaviour of the autopilot might be different than your driving behaviour in a car, you are first presented a short drive (ten corners) where the simulator moves exactly with the simulated car. During this trial you are asked to fully focus on what the motion feels like to become familiar with the way the autopilot performs the drive.

In order to calibrate your rating scale and get used to providing continuous ratings, you will then go through two training trials. Each training trial features (amongst others) the (presumed) best and worst setting/condition tested in the experiment, allowing you to 'calibrate' your internal rating scale. That means that after the training phase you should be able to distinguish your perceived 'best' motion and rate it with a zero score, and perceive your 'worst' motion and rate it with a maximum score.

Each trial will be initiated by the experiment operator after you've indicated that you are ready for the next trial. Before each trial you will be repositioned to the starting point for the next trial. This process may take several seconds.

Experiment

During the experiment you are asked to rate the perceived incoherence between inertial and visual motion through four car drives. Rate this incoherence using the scale you developed during the training phase. Hence continuously ask yourself 'does this motion feel right' and provide a rating to the degree to which it feels incoherent, on the scale illustrated below. Rate *everything* you feel and *always* provide a rating for something you felt, even if your rating might be a little late (a late rating is better than no rating).

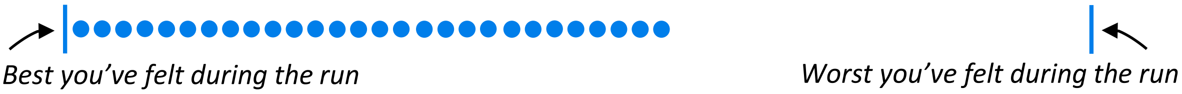


Figure 4 - *Illustration of your rating scale during the experiment (repeated)*

During the experiment some conditions are repeated multiple times, and the differences between conditions may be small. Therefore, you should not necessarily expect to perceive a difference in incoherence between each turn, and just rate the incoherence as you perceive it.

Break & Timing

Between each car manoeuvre you can take a small break in the simulator if desired, after three out of six trials (including training) a break outside the simulator can be taken. If you need more breaks just let the experimenter know. Following a break outside the simulator the short run of ten autopilot corners is played again to remind you of the driving behaviour of this autopilot.

The experiment lasts just over two hours in total.

Important

When giving your response:

- Always use the **motion you saw (the visual scene) as a reference**
- Don't think too much about it, **respond quickly**. We are only interested in your impression
- Rate all incoherent features, even if your rating is a little late
- Try to use **consistent ratings** after the training phase, for the duration of the experiment

For your comfort and safety

If at any time during the study you start feeling bad and/or tired and/or wish to have a break, please communicate to the researchers so that the experiment can be paused or stopped. Note that stopping is possible at all times, your participation in this experiment is completely voluntary.

Appendix B

Participants & Informed consent

Prior to entering the CyberMotion Simulator and starting the experiment, subjects were asked to sign an informed consent form. In this form, subjects indicated to:

1. Be aware of their voluntary participation in the experiment.
2. Have been explained the risks involved in using the CyberMotion Simulator.
3. Have been explained the safety features and emergency procedures of the CyberMotion Simulator.
4. Be aware that their data is stored and that their anonymity is protected.

Additionally, subjects indicated on the form their:

1. Age
2. Number of years they'd owned a car driving license for
3. Average number of car driving hours per week

This produced the data summarised in table B-1.

Table B-1: Self-reported data of the participants who took part in the experiment described in the IEEE paper.

Participant	Gender	Age [Years]	Driver's license [Years]	Car driving [Hours/week]
1	M	28	9	6
2	M	23	5	12
3	M	24	6	15
4	F	26	7	2
5	M	31	10	0
6	F	21	5	0
7	M	37	11	2
8	F	40	15	0
9	M	27	8	1
10	M	21	3	5
11	F	25	8	4
12	F	26	9	1.5
13	M	28	9	0
14	M	25	7	6
15	M	40	22	4
16	M	33	14	2

Appendix C

Experimental order

In the experiment carried out, subjects rated the perceived visual-motion mismatch during four experimental runs. Prior to the experiment, two training runs were simulated to enable subjects to get familiar with the experienced conditions. In this appendix, the experimental order is elaborated on. Details of all runs used in the experiment are given in section C-1, the experiment matrix used in this experiment is given in section C-2.

C-1 Order of conditions

In this experiment six runs were used (two training runs, four experiment runs) in addition to a shorter familiarisation run. Each training/experiment run consisted of 22 90-degree corners with a radius of 120m, with 150m straight sections in between each corner. Each corner featured a different lateral specific force condition, for completeness all conditions are repeated in table C-1:

An overview of the distribution of conditions over the runs is shown in table C-2. Further explanation is given below:

Familiarisation - The familiarisation run consisted of ten curves, each with condition C1.

Training - For the training runs, a selection was made of motion conditions thought to result in the smallest perceived visual-motion mismatches (C1 (2x), C2 (2x), C6, C10 and C20). Also, conditions thought to result in the biggest perceived-visual-motion mismatches (C9, C14, C15, C16, C18 (2x), C19 and C22 (2x)) were selected. Finally, some medium-severity errors were included (C3(2x), C8 (2x), C13 (2x)). These three groups of severity would allow a subject to determine his/her minimum and maximum perceived visual-motion mismatch. Also, it allowed for practising giving a rating on this scale. It was ensured all types of motion cueing error were present during training.

Experiment - During the four experiment runs, each condition featured twice in a left-handed turn and twice in a right-handed turn. Also, the conditions were spread out over all four runs as much as possible.

Table C-1: Specific lateral force conditions used in the experiment.

Condition	**Error type**	**Washout filter modifications** k_y, ζ *in [-],* ω_n *in [rad/s]*
C1	-	None
C2	-	None
C3	S1	Scaled with $k_\mathrm{y} = 0.4$
C4	S1	Scaled with $k_\mathrm{y} = 0.5$
C5	S1	Scaled with $k_\mathrm{y} = 0.6$
C6	S1	Scaled with $k_\mathrm{y} = 0.8$
C7	F2	Scaled with $k_\mathrm{y} = 1.2$
C8	F2	Scaled with $k_\mathrm{y} = 1.275$
C9	S2	LP-filtered using $\omega_n = 0.6$, $\zeta = 1$ for t <10 s
C10	S2	Filtered using $\omega_{n_{HP}} = 0.5$, $\zeta_{HP} = 0.8$, $\omega_{n_{LP}} = 2.4$, $\zeta_{LP} = 1.2$
C11	S2	Filtered using $\omega_{n_{HP}} = 0.5$, $\zeta_{HP} = 0.6$, $\omega_{n_{LP}} = 2.0$, $\zeta_{LP} = 1.2$
C12	S2	Filtered using $\omega_{n_{HP}} = 0.55$, $\zeta_{HP} = 0.35$, $\omega_{n_{LP}} = 2.0$, $\zeta_{LP} = 1.2$
C13	S2	Filtered using $\omega_{n_{HP}} = 0.5$, $\zeta_{HP} = 0.125$, $\omega_{n_{LP}} = 1.2$, $\zeta_{LP} = 1$
C14	F1	$a_y(t) = \pm 0.6\left(e^{\left(\frac{t-2.5}{1.2}\right)^2} - e^{\left(\frac{t-17}{1.6}\right)^2}\right)$
C15	F1	$a_y(t) = \pm 0.85\left(e^{\left(\frac{t-4}{2}\right)^2} - e^{\left(\frac{t-15.6}{2}\right)^2}\right)$
C16	F2	LP-filtered using $\omega_n = 0.6$, $\zeta = 1$ for t >12 s
C17	F2	Error of C9 added to input after t >10 s
C18	F3a	Scaled with $k_\mathrm{y} = 0.5$ for $\Delta t = 0.05$ s
C19	F3b	-24dBW white noise added to input
C20	P	Input delayed by 0.6s
C21	P	Input delayed by 1.5s
C22	P	Input delayed by 2.5s

Table C-2: Distribution of left- (L) and right-handed (R) corners and experimental conditions over the runs used in the experiment.

Corner	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Run																						
F: Familiarisation	L	R	R	R	L	L	R	L	L	R												
	C1	C1	C1	C1	C1	C1	C1	C1	C1	C1												
T1: Training 1	L	R	R	L	R	R	L	L	L	R	L	R	R	L	L	R	R	L	R	R	L	L
	C1	C13	C3	C20	C8	C6	C19	C14	C22	C2	C13	C18	C10	C3	C9	C1	C15	C8	C16	C22	C2	C18
T2: Training 2	R	L	R	L	L	L	R	R	L	R	R	L	R	R	L	L	R	L	L	R	L	R
	C20	C22	C2	C18	C8	C13	C1	C14	C3	C8	C18	C6	C22	C19	C2	C16	C9	C15	C1	C13	C10	C3
E1: Experiment 1	L	R	L	R	R	L	L	R	L	L	R	R	L	R	L	R	R	R	L	L	R	L
	C14	C3	C18	C20	C12	C7	C13	C15	C19	C6	C17	C10	C2	C9	C22	C4	C21	C1	C11	C5	C8	C16
E2: Experiment 2	R	R	L	L	R	L	L	R	L	R	L	R	L	L	R	R	L	R	R	L	L	R
	C6	C2	C17	C9	C22	C4	C11	C16	C21	C5	C20	C8	C1	C15	C13	C18	C7	C19	C10	C3	C14	C12
E3: Experiment 3	L	L	R	R	L	R	R	L	R	L	R	L	L	R	R	L	R	L	R	L	L	R
	C8	C12	C16	C17	C3	C13	C14	C10	C5	C19	C4	C18	C9	C1	C11	C20	C7	C21	C15	C22	C6	C2
E4: Experiment 4	R	L	L	R	L	R	L	R	R	L	R	L	L	R	L	R	R	L	L	R	R	L
	C19	C4	C15	C22	C2	C18	C5	C21	C14	C1	C11	C8	C17	C3	C12	C6	C20	C16	C13	C7	C9	C10

C-2 Experiment matrix

The two training runs and four experimental runs presented in the previous section were presented to the subjects in random order. During the experiment, the experiment matrix shown in table C-3 was used.

Between the two blocks of the experiment (which each lasted around 35 minutes) a break outside the simulator was taken. Subjects were instructed to let the experimenter know when they felt comfortable to continue again, this usually was after around 10-15 minutes. In between runs within a block, subjects were given the opportunity to take a short break in the (stationary) simulator.

Table C-3: Experiment matrix used during the experiment.

Participant	Block 1					Block 2			
	Run 1	Run 2	Run 3	Run 4	Break	Run 1	Run 2	Run 3	Run 4
1	F	T2	T1	E4		F	E1	E2	E3
2	F	T1	T2	E1		F	E2	E4	E3
3	F	T2	T1	E3		F	E4	E2	E1
4	F	T2	T1	E3		F	E2	E4	E1
5	F	T2	T1	E2		F	E3	E1	E4
6	F	T2	T1	E3		F	E4	E1	E2
7	F	T1	T2	E2		F	E1	E3	E4
8	F	T2	T1	E1		F	E2	E3	E4
9	F	T1	T2	E1		F	E3	E4	E2
10	F	T2	T1	E4		F	E2	E1	E3
11	F	T1	T2	E4		F	E1	E3	E2
12	F	T1	T2	E4		F	E2	E3	E1
13	F	T1	T2	E2		F	E1	E4	E3
14	F	T1	T2	E2		F	E4	E3	E1
15	F	T2	T1	E2		F	E3	E1	E4
16	F	T2	T1	E2		F	E3	E4	E1

Appendix D

Sickness Scores

Both prior to and after the experiment, subjects were asked to fill in a simulator sickness scoring form introduced by Kennedy *et al.* (Kennedy, Lane, Berbaum & Lilienthal, 1993). The total simulator sickness scores before and after the experiment are shown in table D-1.

From table D-1 one can observe participants 8, 9, 10 and 15 to have experienced an increase larger than 50 points. These subjects all indicated to have provided ratings continuously, despite their simulator sickness effects. Inspecting the individual ratings of these four subjects showed no ratings to seem abnormal, for this reason data from these subjects was retained.

Table D-1: Total simulator sickness scores, before and after the experiment.

Participant	**Total score** *Pre-experiment*	**Total score** *Post-experiment*
1	7.5	18.7
2	15.0	11.2
3	11.2	33.7
4	0.0	11.2
5	3.7	7.5
6	18.7	48.6
7	22.4	44.9
8	0.0	93.5
9	0.0	71.1
10	3.7	56.1
11	33.4	44.9
12	37.4	48.6
13	7.5	37.4
14	29.9	37.4
15	22.4	97.2
16	7.5	15.0

Appendix E

IMU Measurements

To validate the motion presented by the Cybermotion Simulator (CMS), IMU measurements were taken of the inertial motion of the CMS. A description of the apparatus used is given in section E-1, measurement results are shown in section E-2. The IMU measurements are discussed in section E-3.

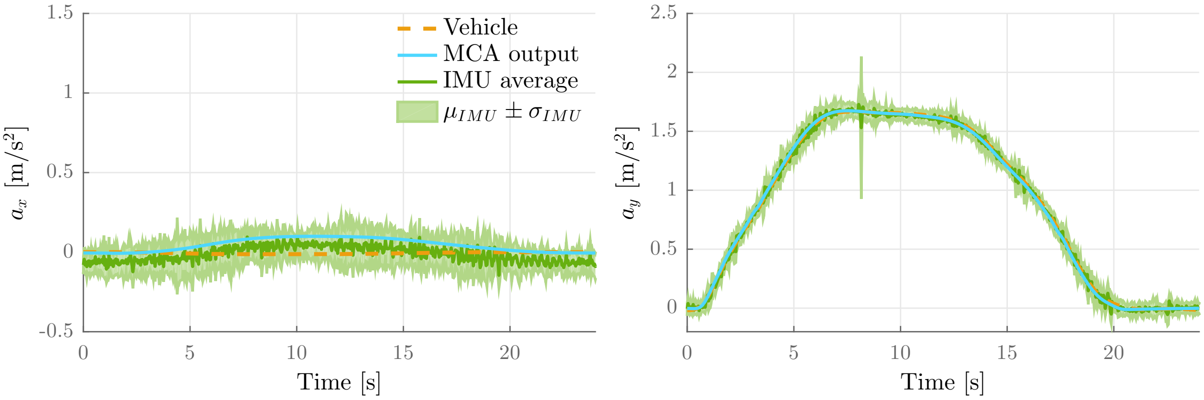
E-1 Apparatus

An inertial measurement unit (Stim300, Sensonor AS, Norway) was placed in the cabin of the CMS. This IMU was positioned 20 centimetres below the eye height of a subject, some 20 centimetres behind the head of a subject and 30 centimetres to the right of a subject's head. Due to the low tilt rates presented in this simulation (<3 deg/s) the IMU should present a reasonably accurate estimation of the inertial motion experienced by subjects, even though it was mounted some distance away from a subject's head.

The IMU has a sampling rate of 250 Hz. To synchronise the IMU clock with the CMS clock, the IMU clock time was sent to a CMS log file at the start of each run.

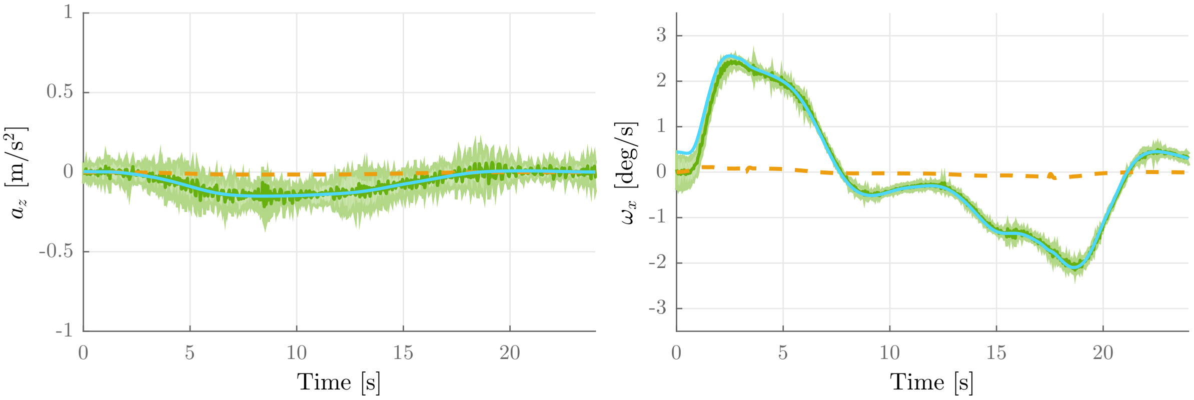
E-2 Measurement results

The IMU measurements for all 22 conditions are shown in figures E-1 through E-22. Data collected for left- and right-handed curves has been merged, by changing the sign of the lateral specific force a_y, roll rate ω_x and yaw rate ω_z for all right-handed curves. The confidence interval shown is based on between 23 and 35 samples, since not all IMU measurements were successful.



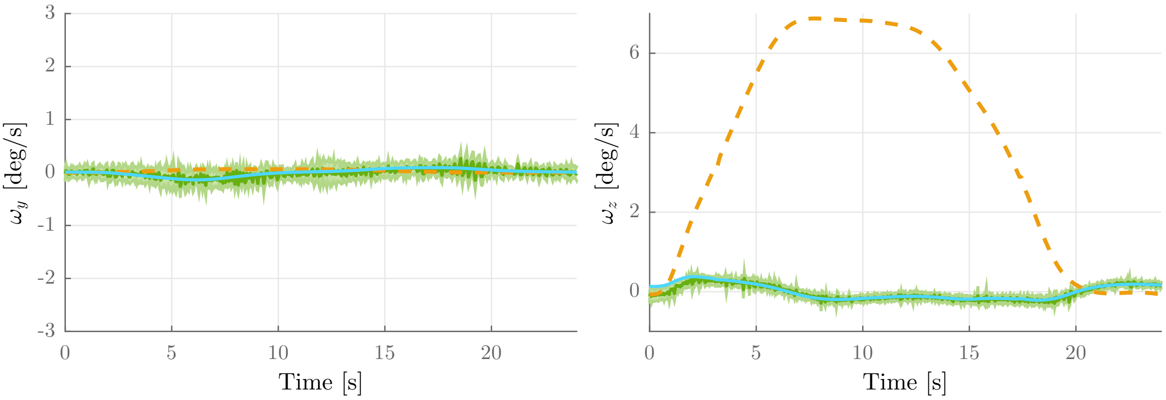
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

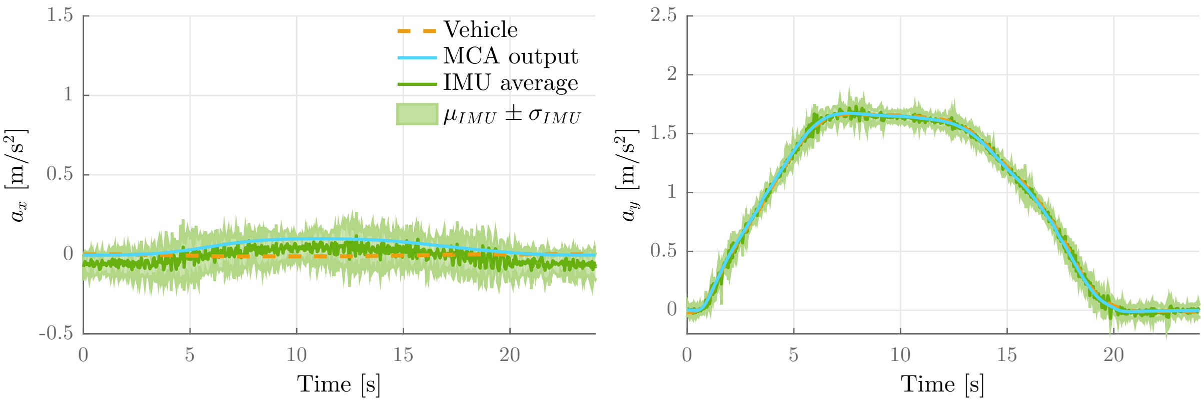
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

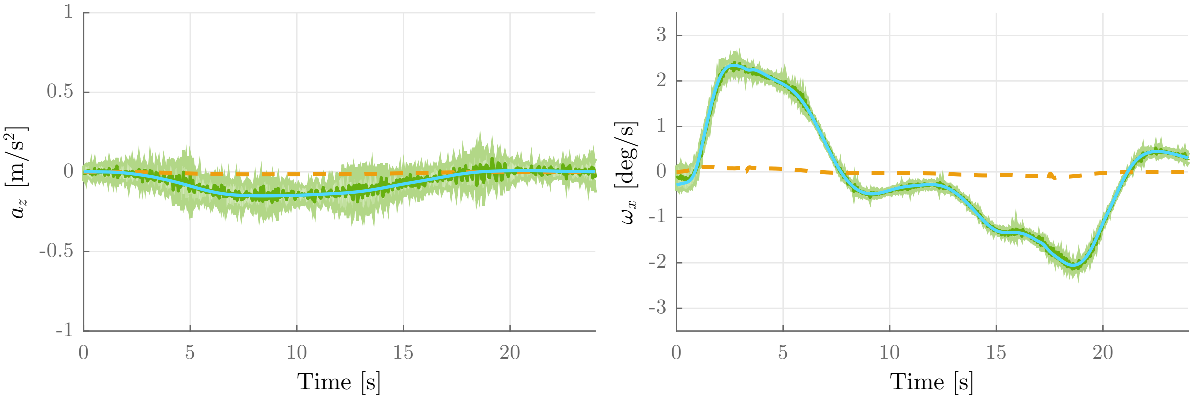
(f) Yaw rate ω_z.

Figure E-1: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C1. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



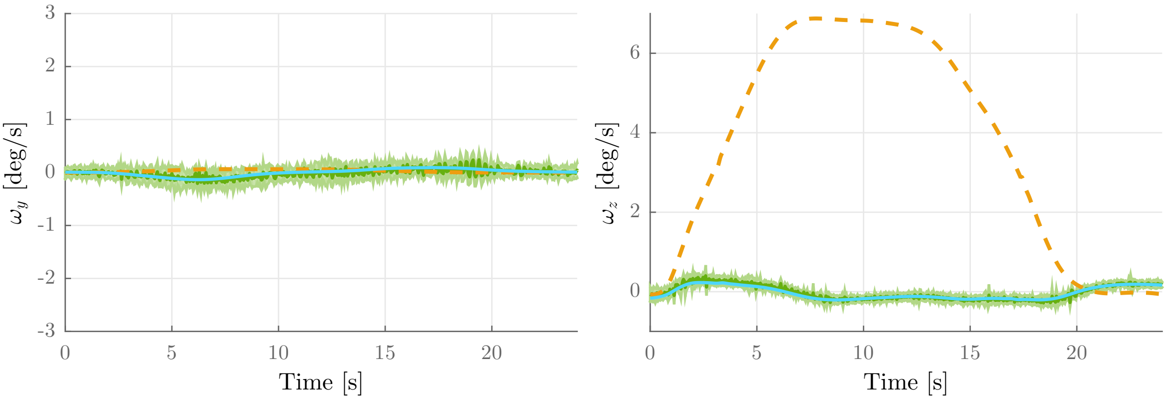
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

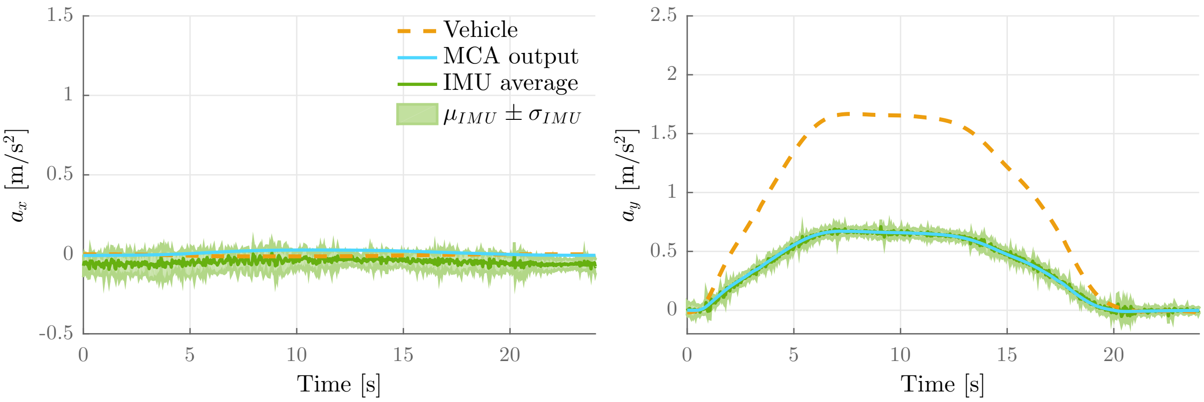
(d) Roll rate ω_x.



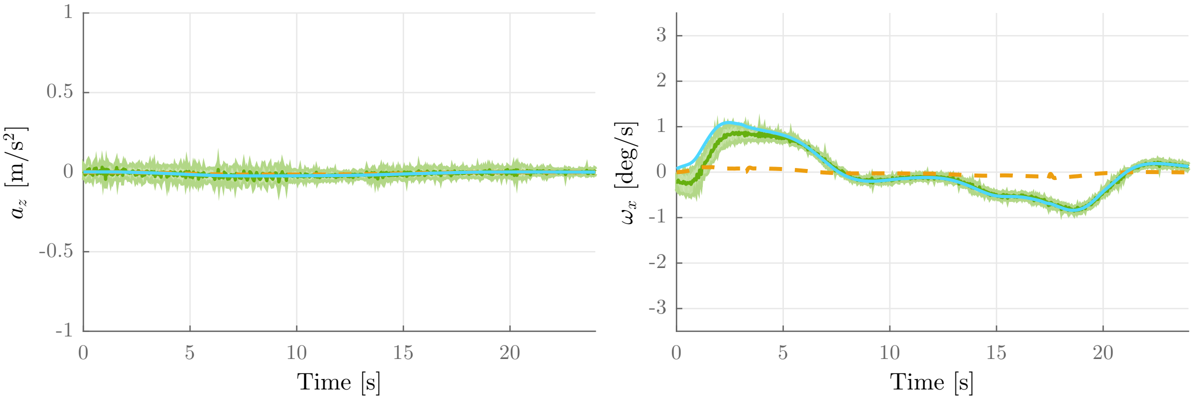
(e) Pitch rate ω_y.

(f) Yaw rate ω_z.

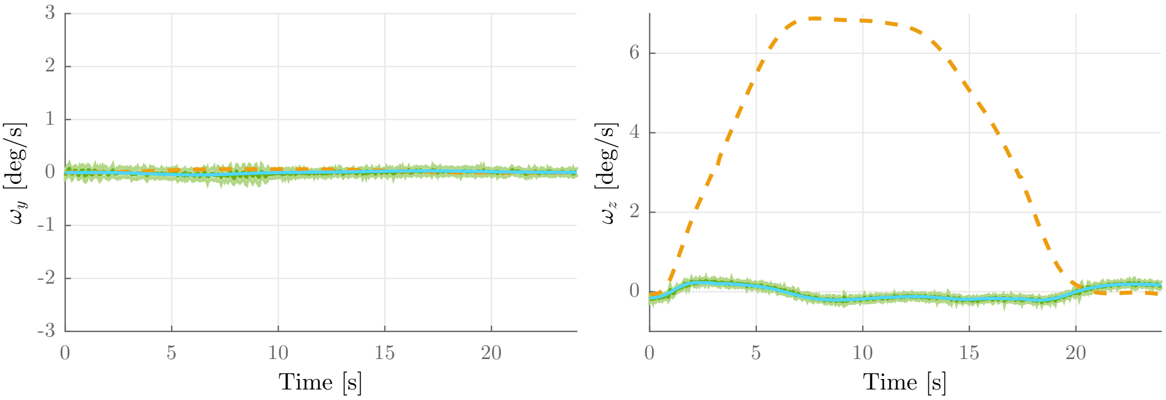
Figure E-2: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C2. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



(a) Longitudinal specific force a_x. **(b)** Lateral specific force a_y.

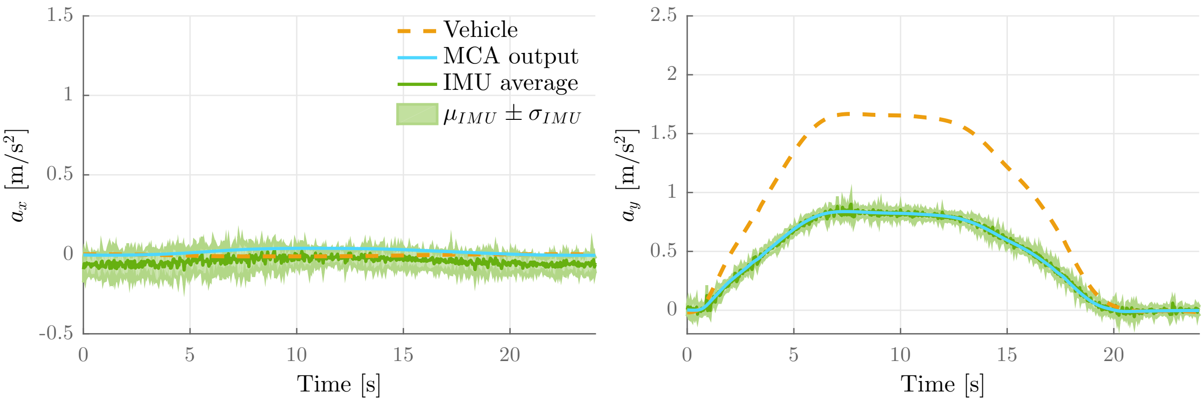


(c) Vertical specific force a_z. **(d)** Roll rate ω_x.



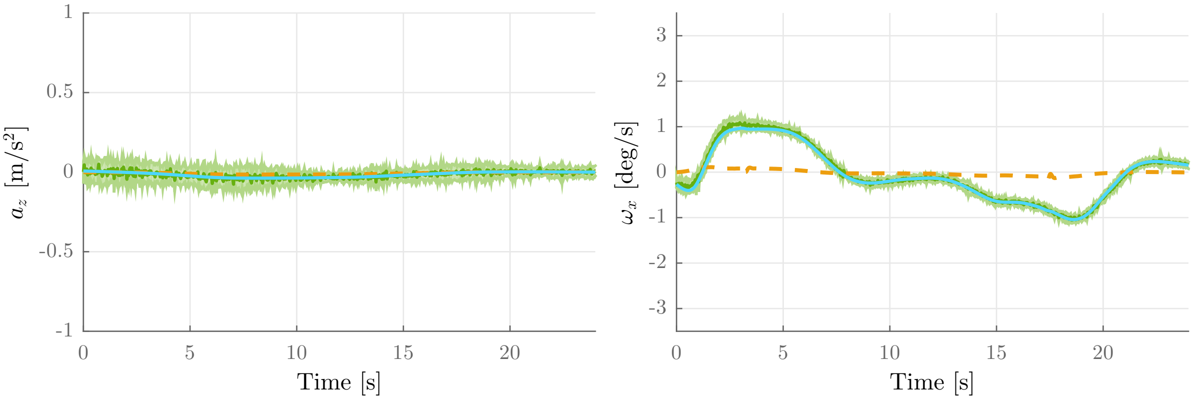
(e) Pitch rate ω_y. **(f)** Yaw rate ω_z.

Figure E-3: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C3. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



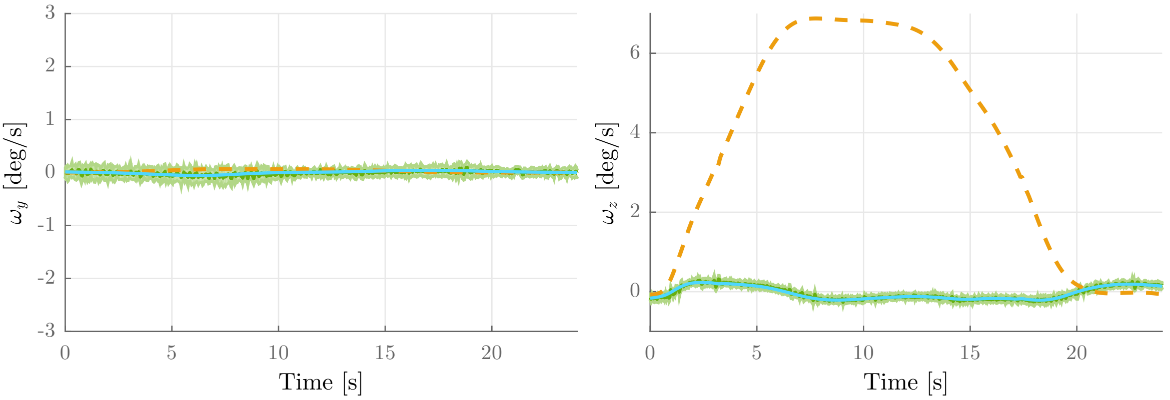
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

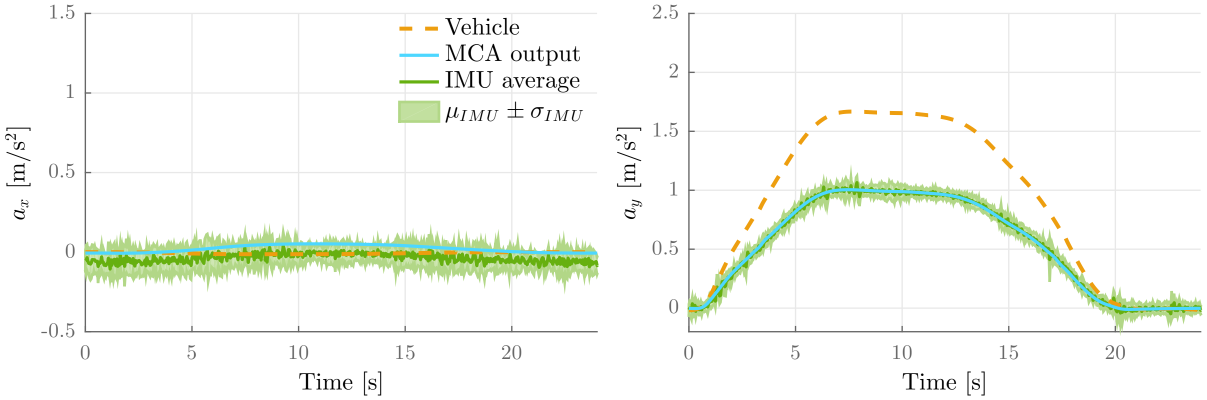
(d) Roll rate ω_x.



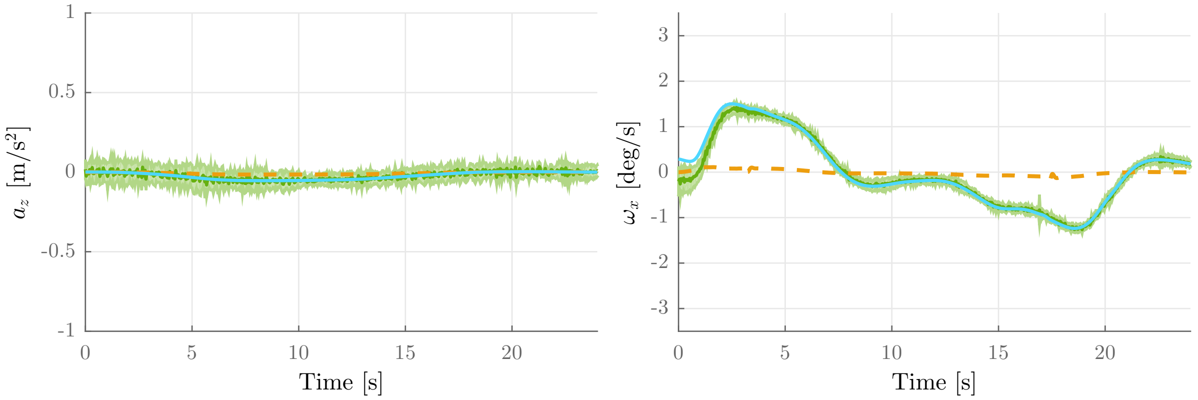
(e) Pitch rate ω_y.

(f) Yaw rate ω_z.

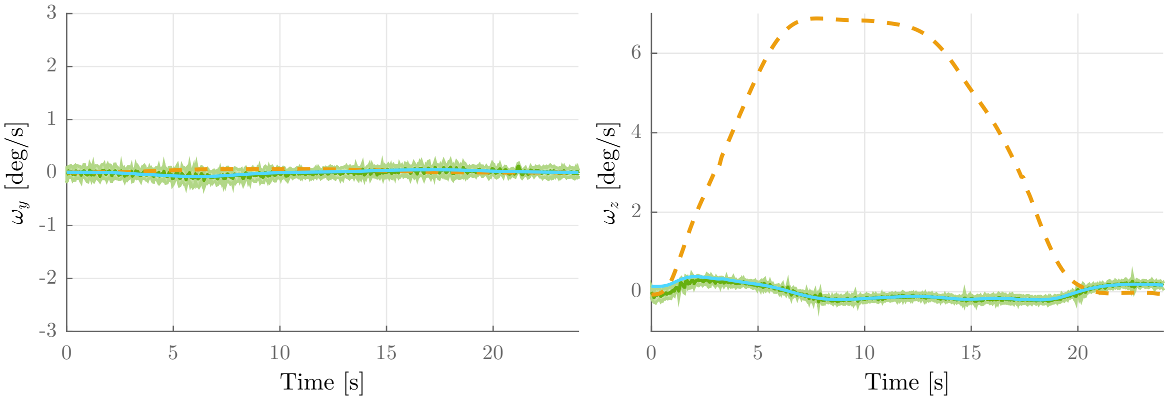
Figure E-4: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C4. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



(a) Longitudinal specific force a_x. **(b)** Lateral specific force a_y.

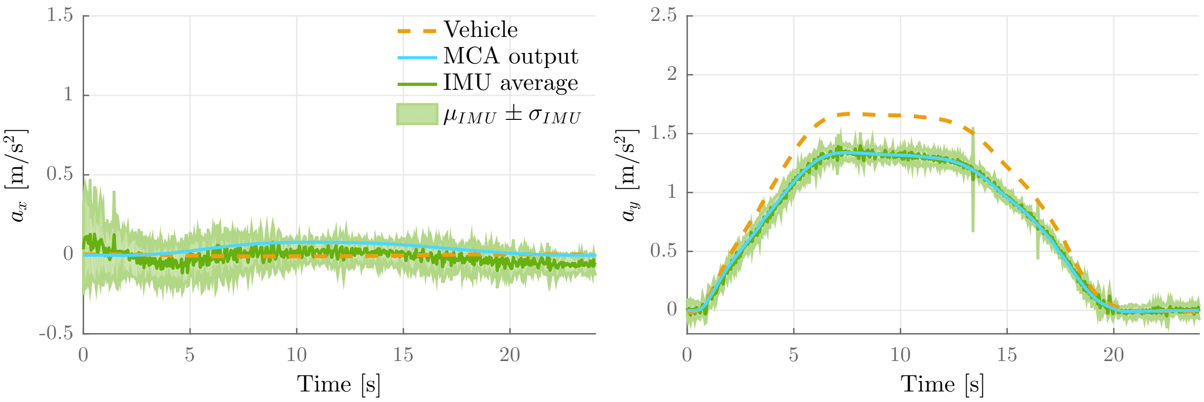


(c) Vertical specific force a_z. **(d)** Roll rate ω_x.



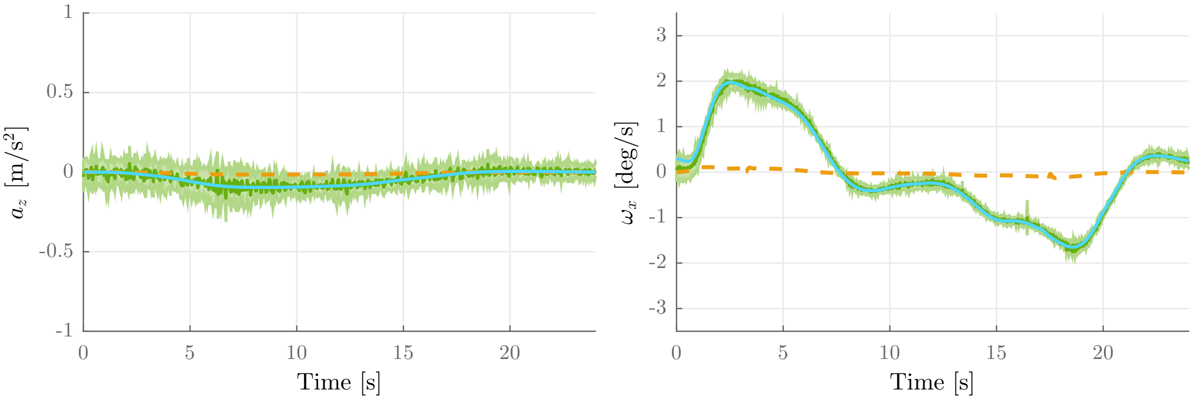
(e) Pitch rate ω_y. **(f)** Yaw rate ω_z.

Figure E-5: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C5. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



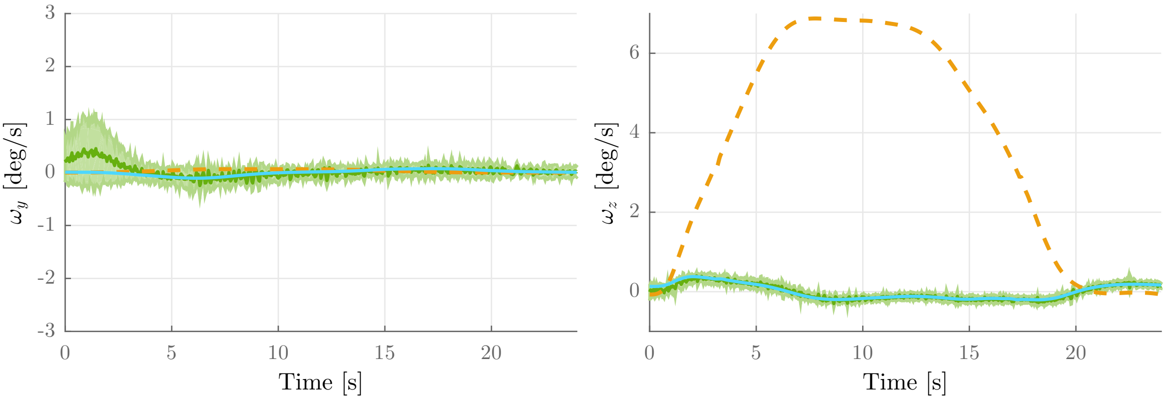
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

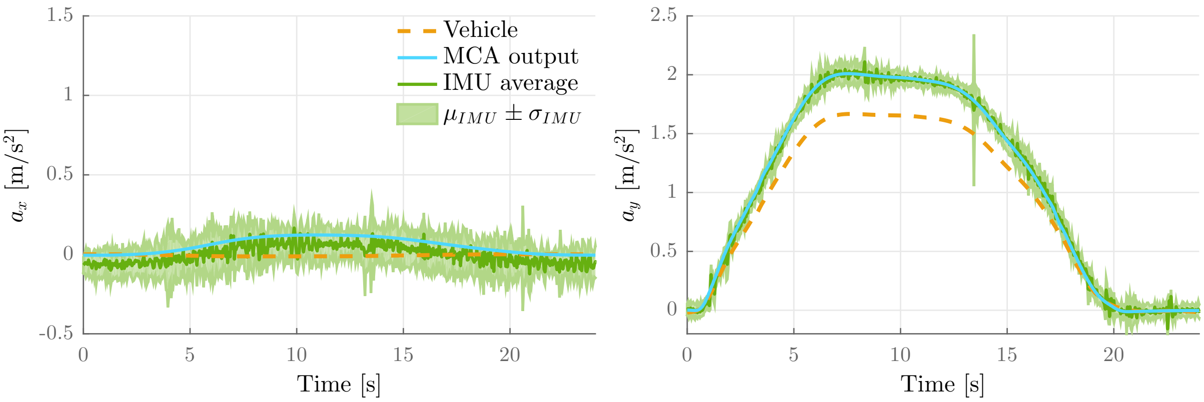
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

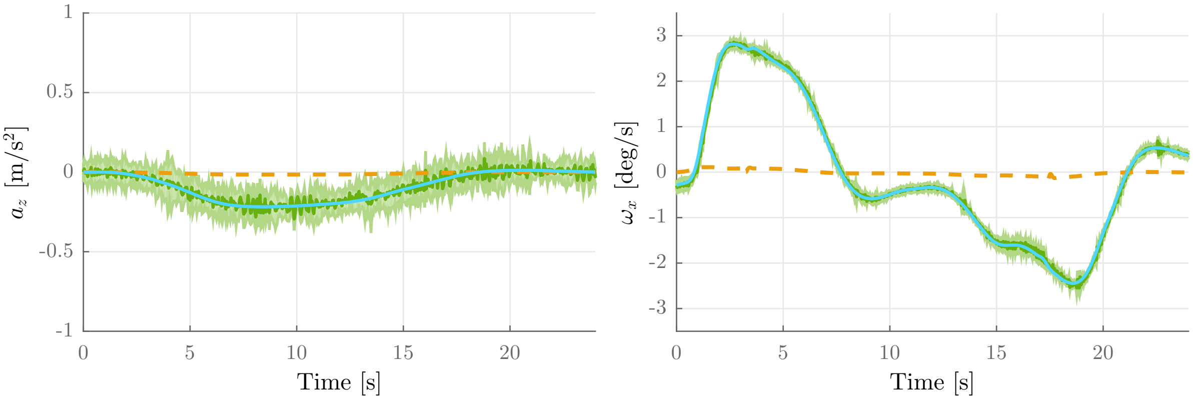
(f) Yaw rate ω_z.

Figure E-6: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C6. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



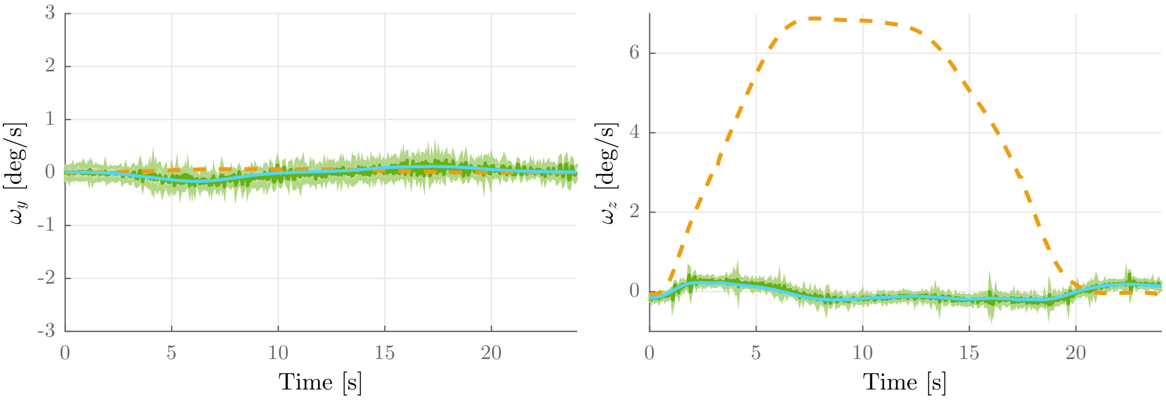
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

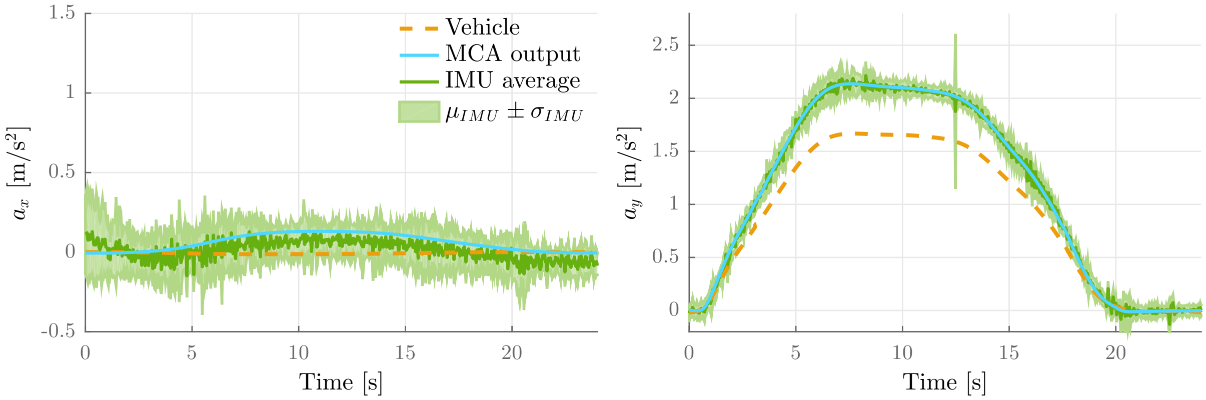
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

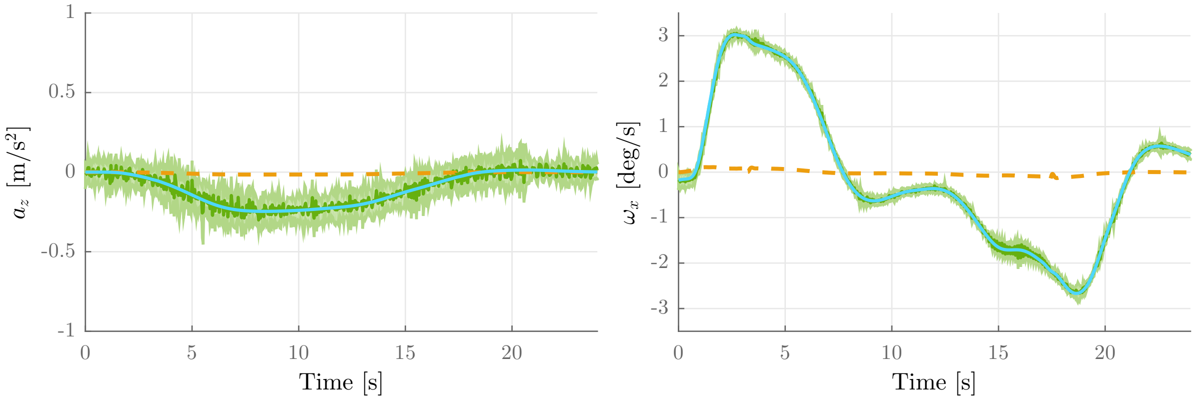
(f) Yaw rate ω_z.

Figure E-7: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C7. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



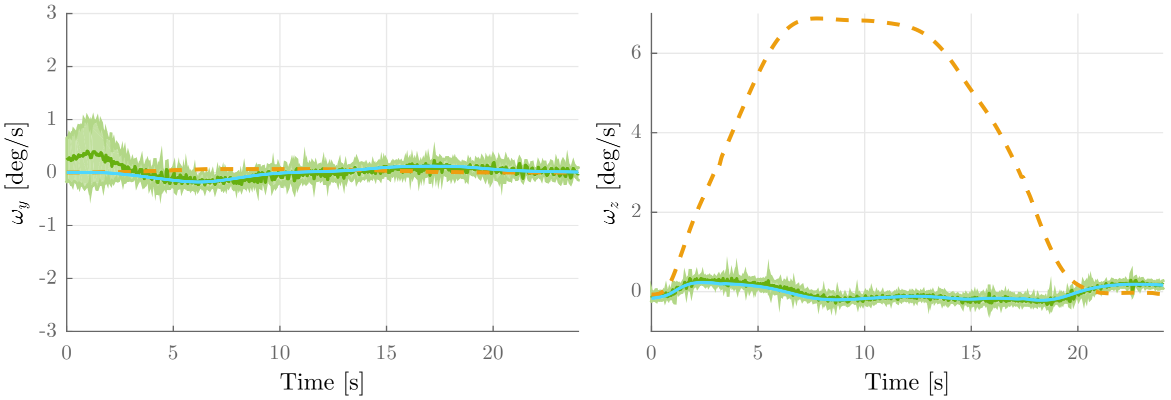
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

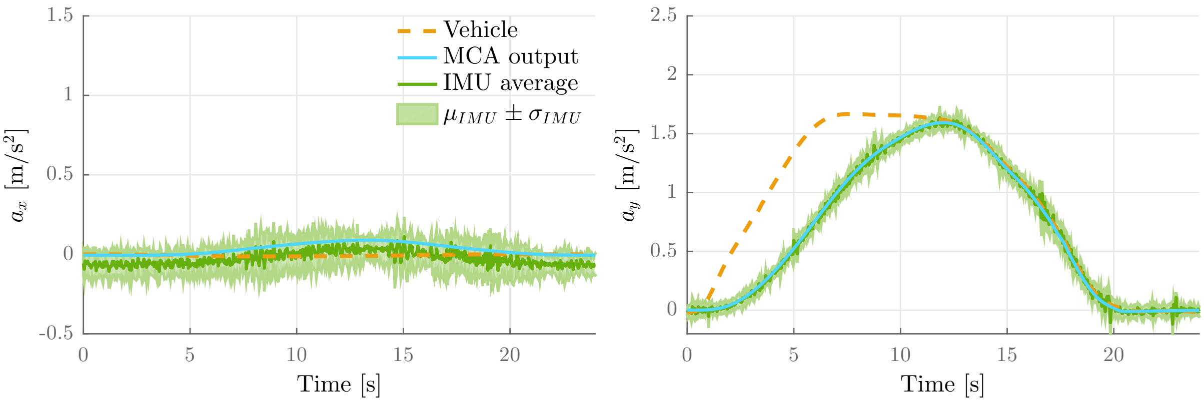
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

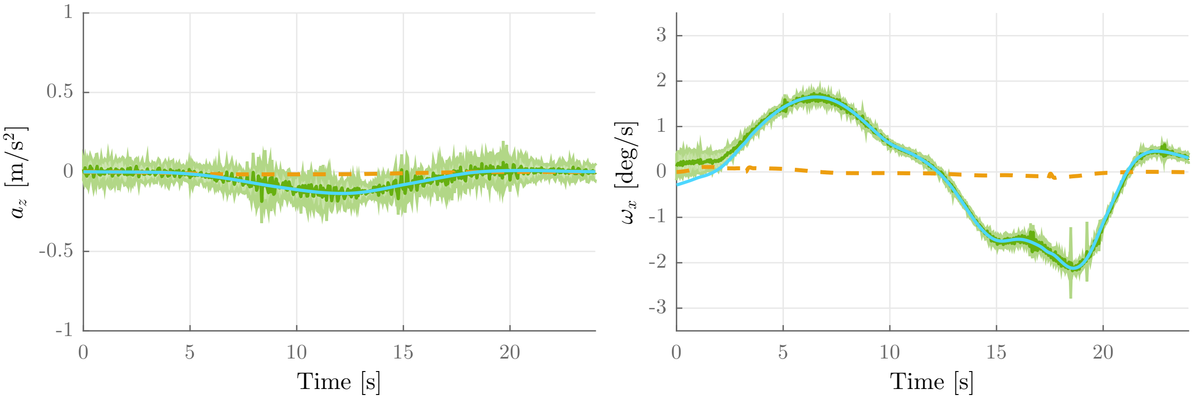
(f) Yaw rate ω_z.

Figure E-8: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C8. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



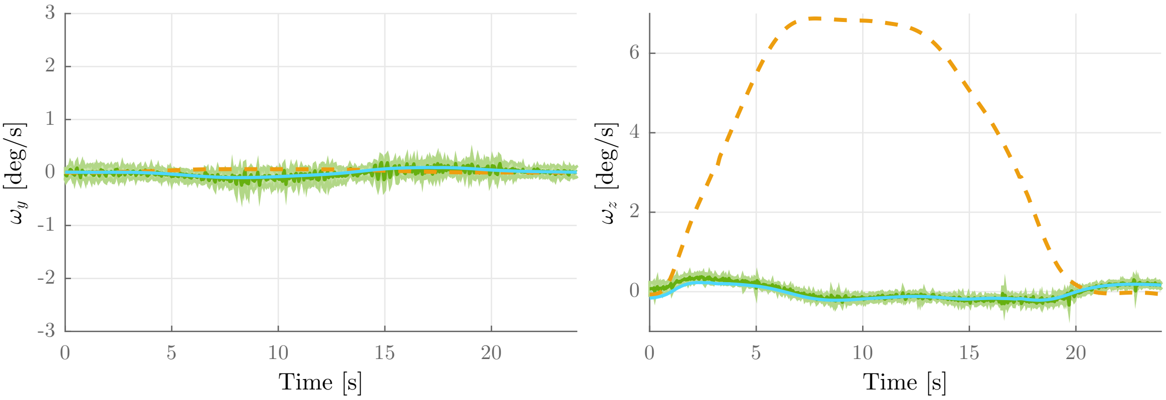
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

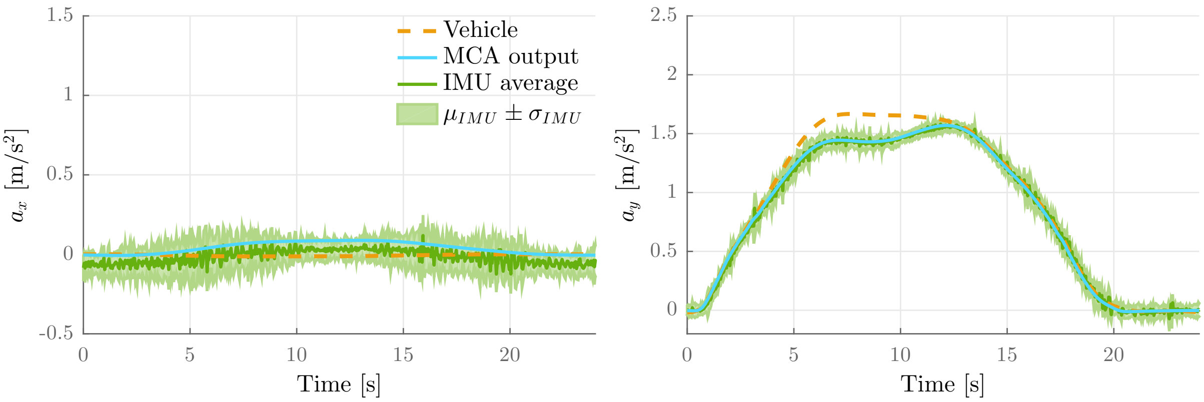
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

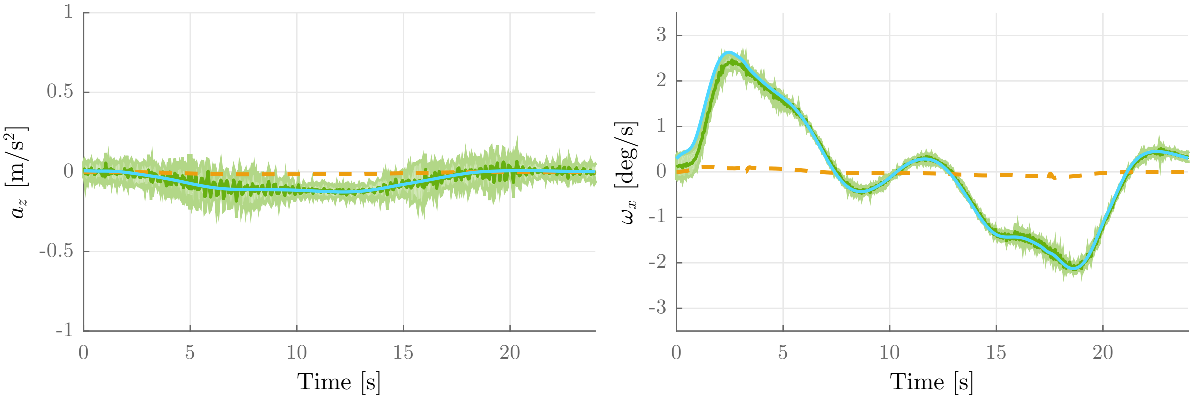
(f) Yaw rate ω_z.

Figure E-9: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C9. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



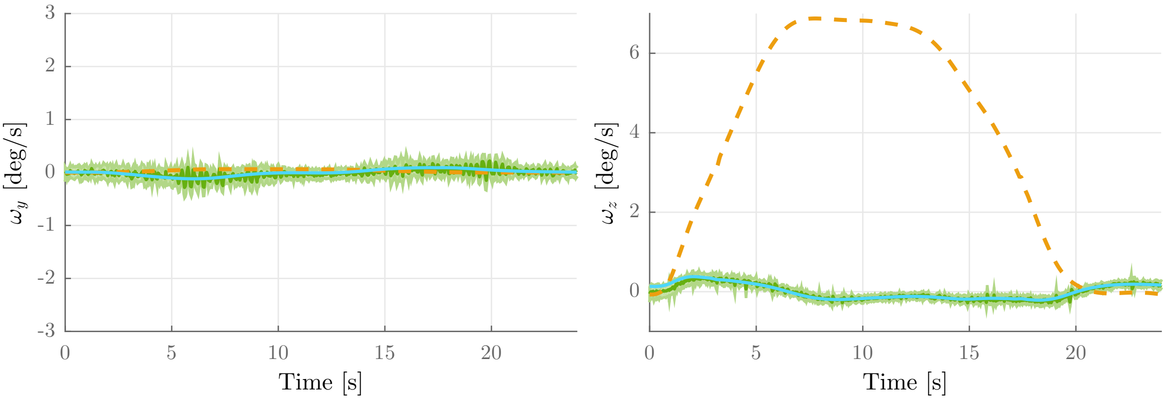
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

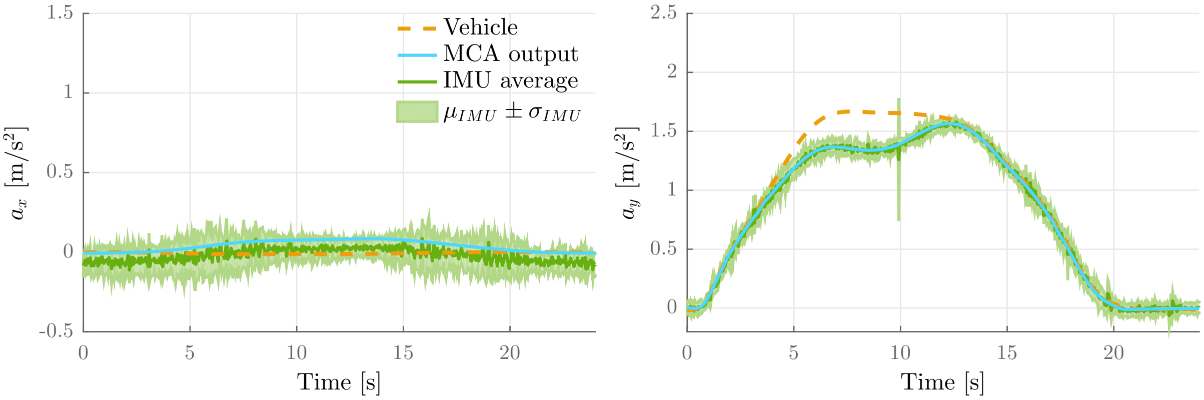
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

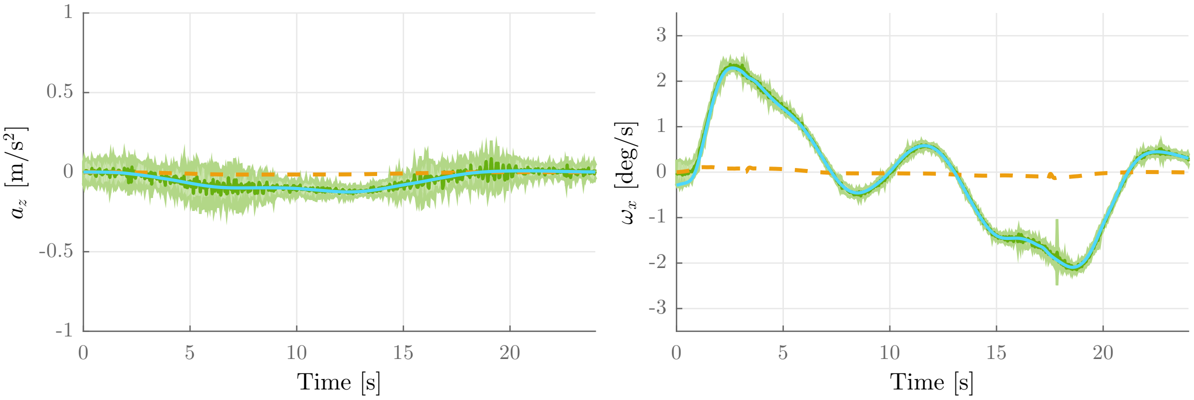
(f) Yaw rate ω_z.

Figure E-10: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C10. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



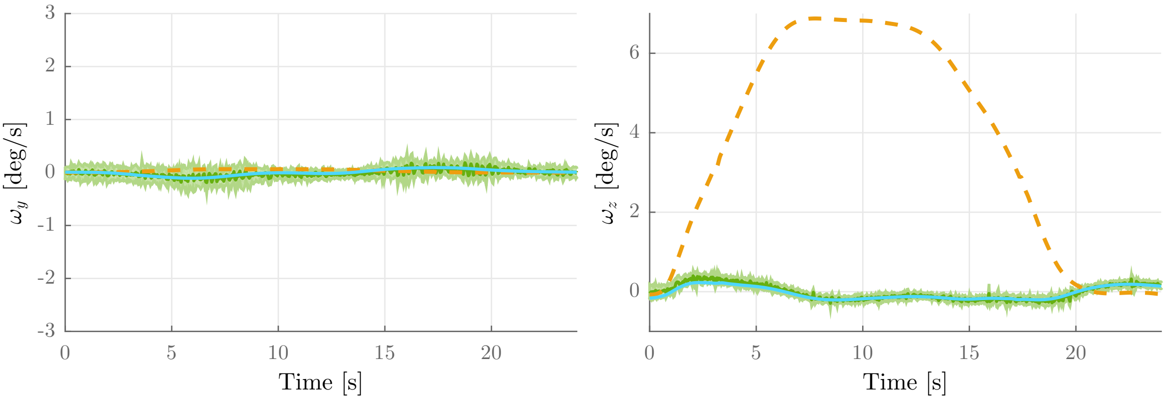
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

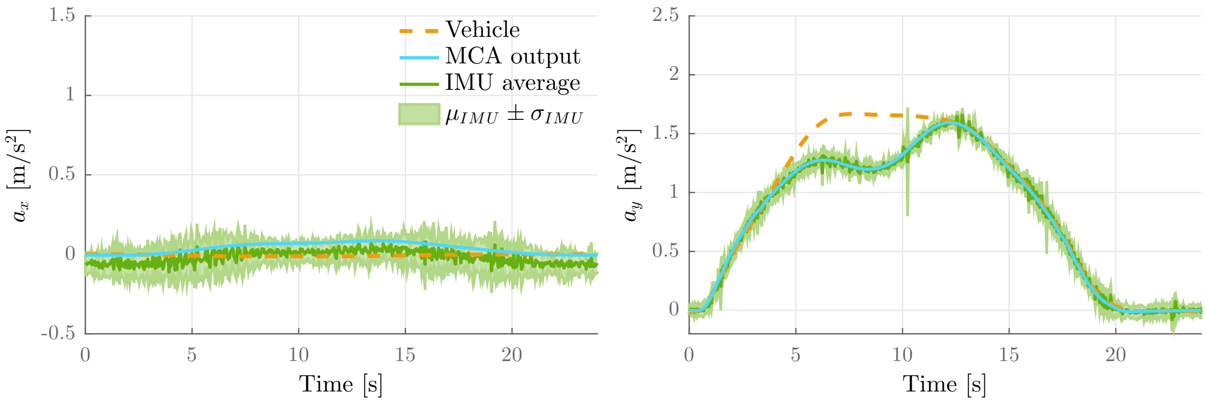
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

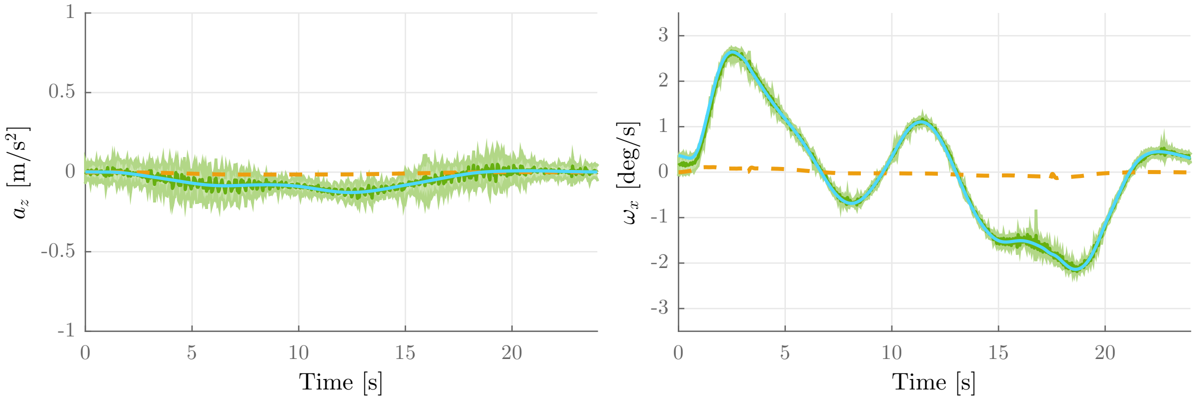
(f) Yaw rate ω_z.

Figure E-11: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C11. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



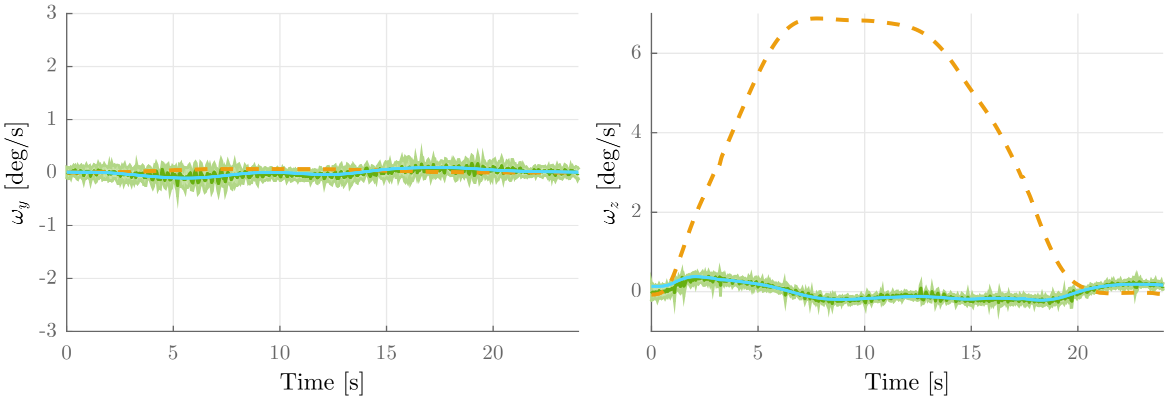
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

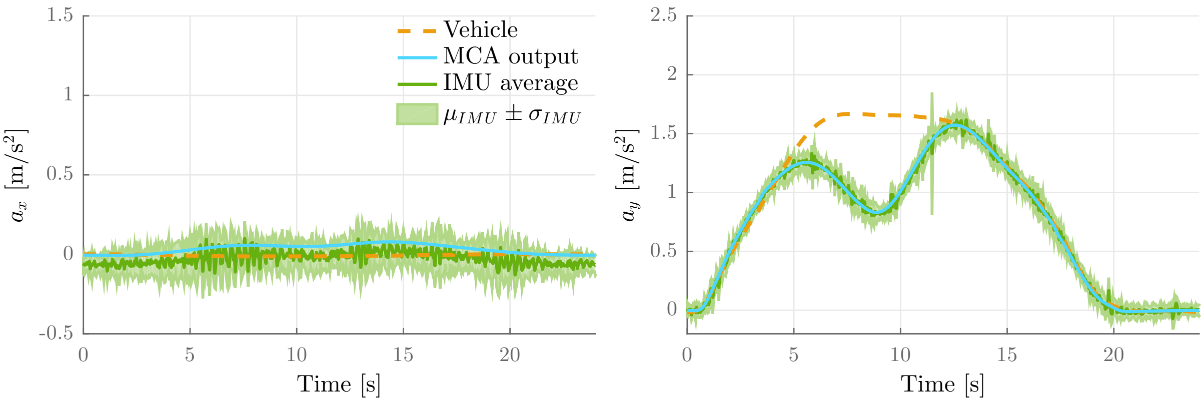
(d) Roll rate ω_x.



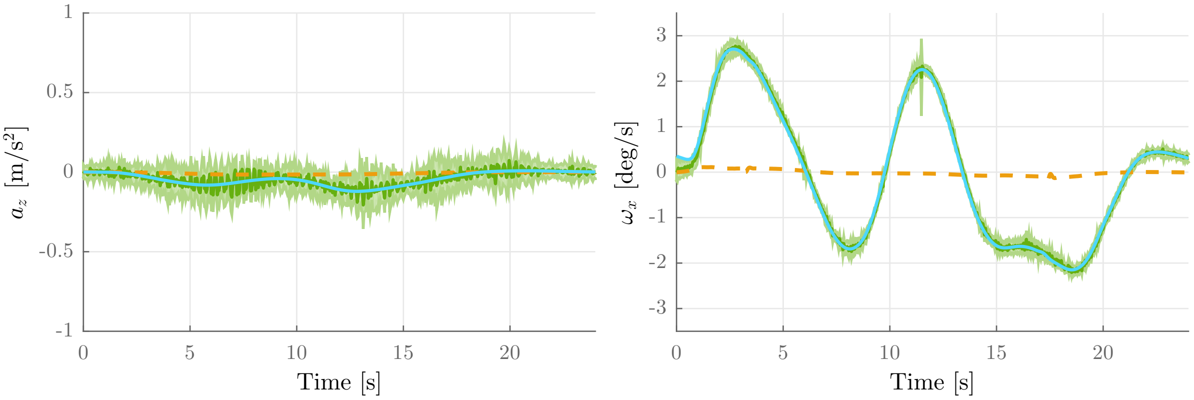
(e) Pitch rate ω_y.

(f) Yaw rate ω_z.

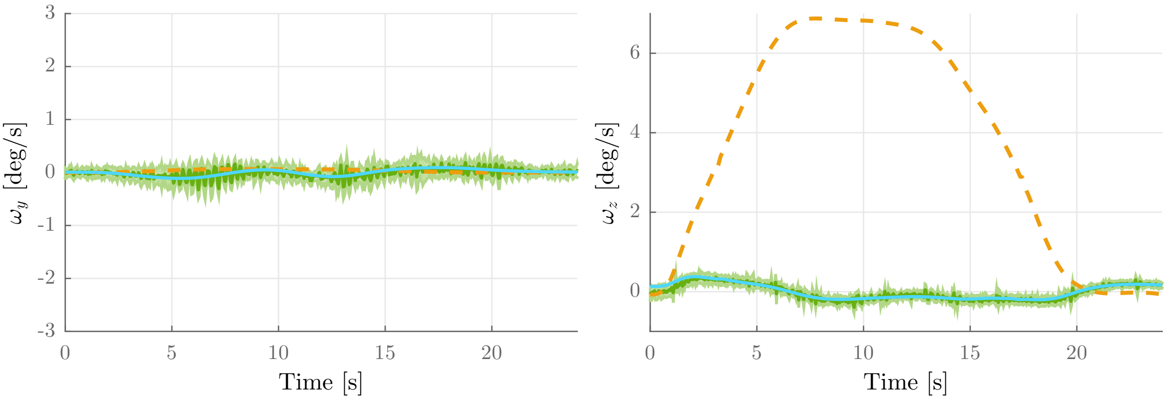
Figure E-12: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C12. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



(a) Longitudinal specific force a_x. **(b)** Lateral specific force a_y.

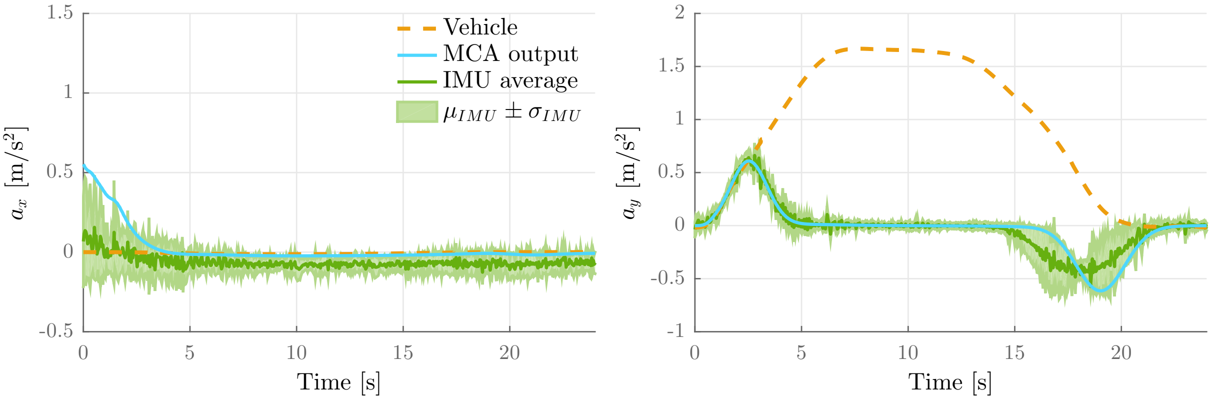


(c) Vertical specific force a_z. **(d)** Roll rate ω_x.

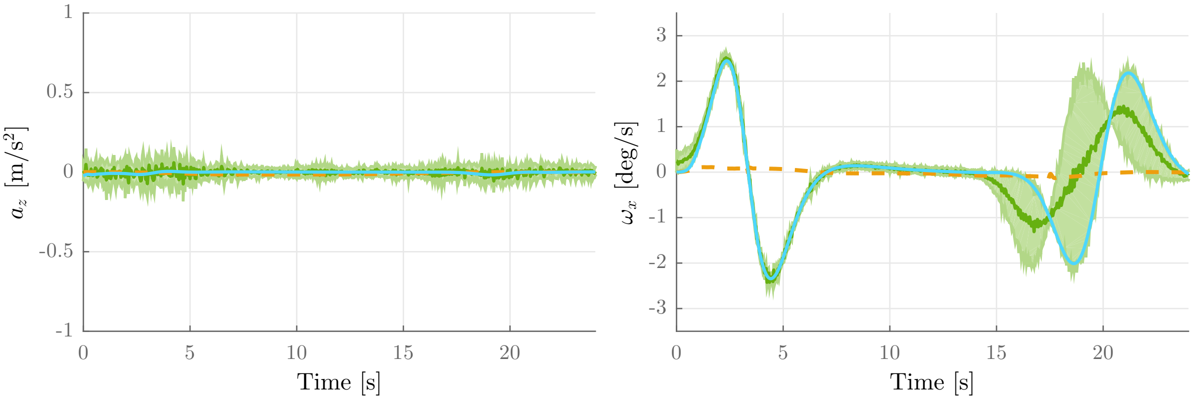


(e) Pitch rate ω_y. **(f)** Yaw rate ω_z.

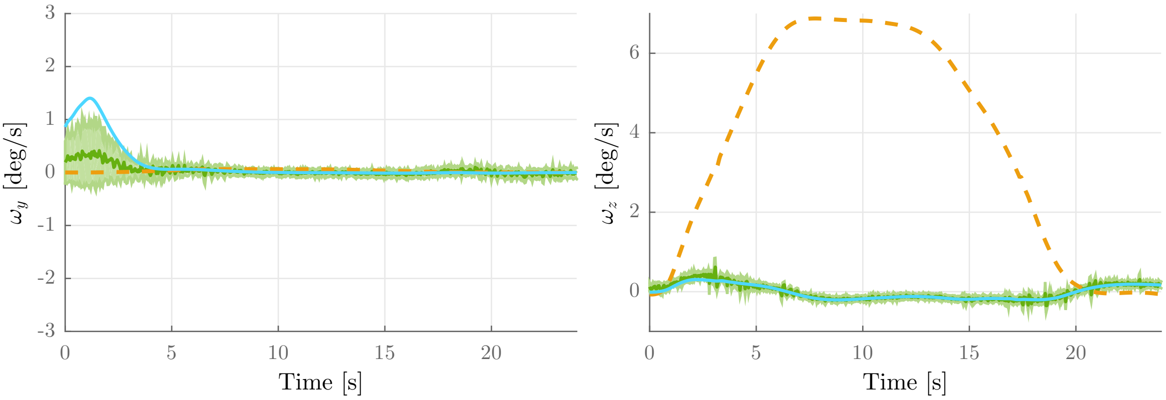
Figure E-13: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C13. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



(a) Longitudinal specific force a_x. **(b)** Lateral specific force a_y.

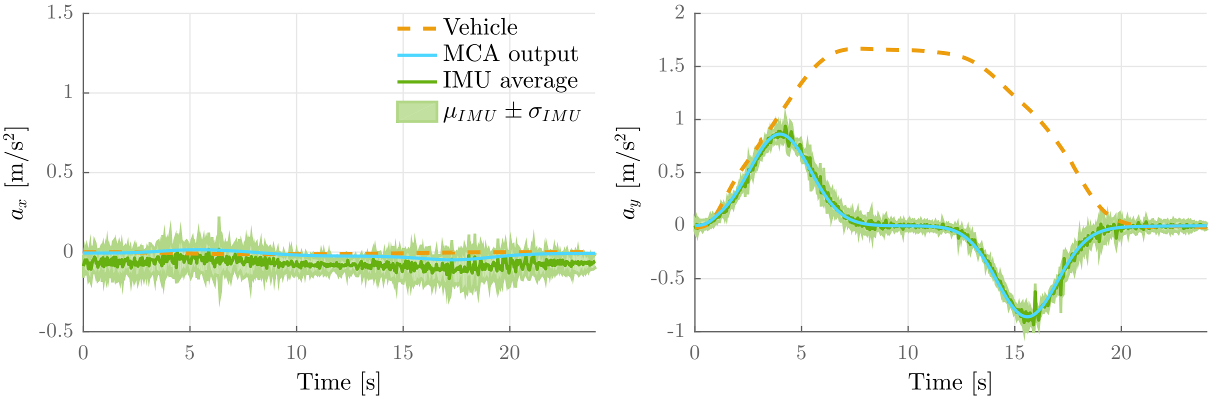


(c) Vertical specific force a_z. **(d)** Roll rate ω_x.



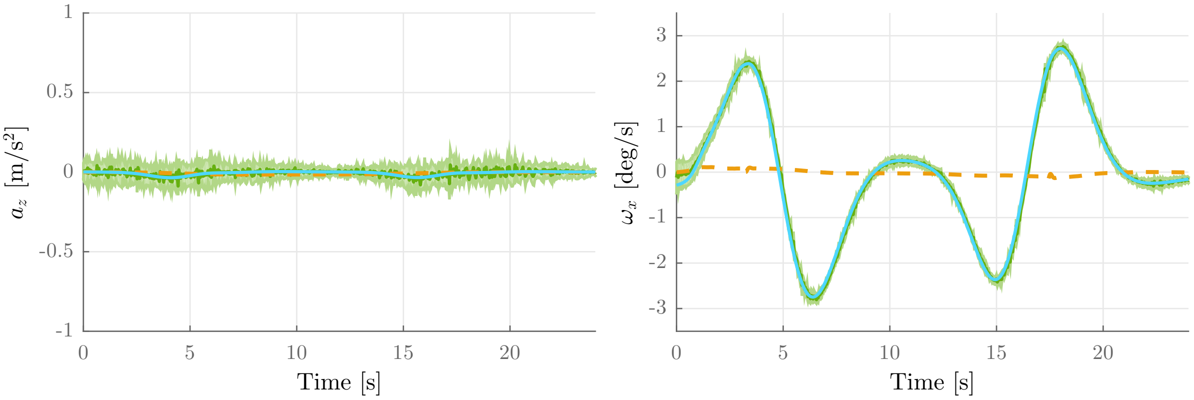
(e) Pitch rate ω_y. **(f)** Yaw rate ω_z.

Figure E-14: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C14. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



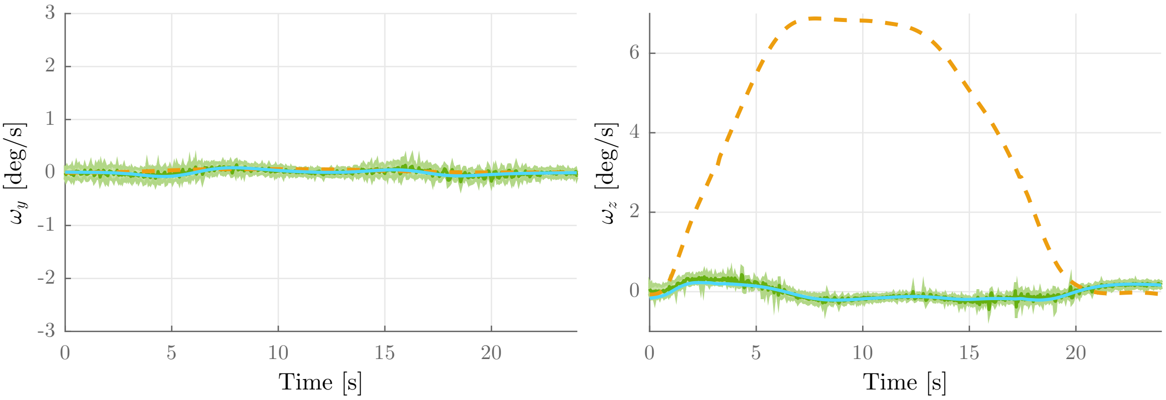
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

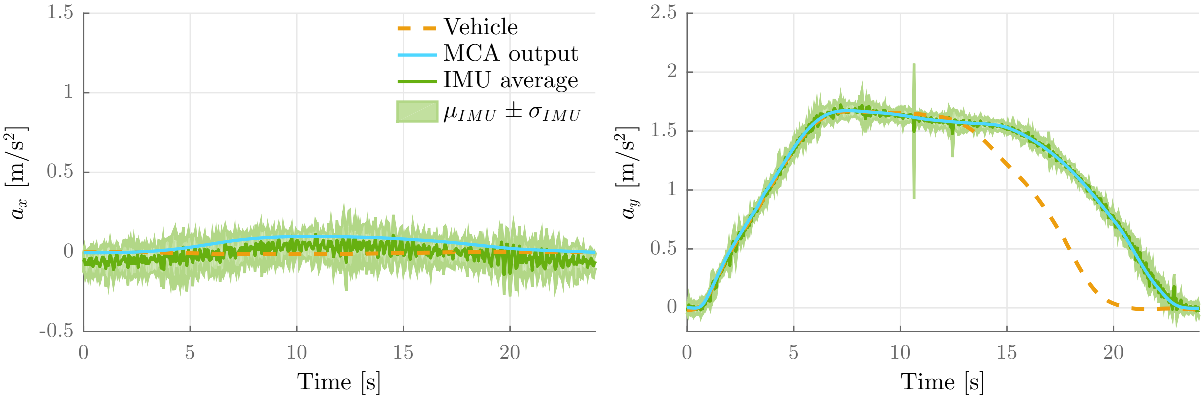
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

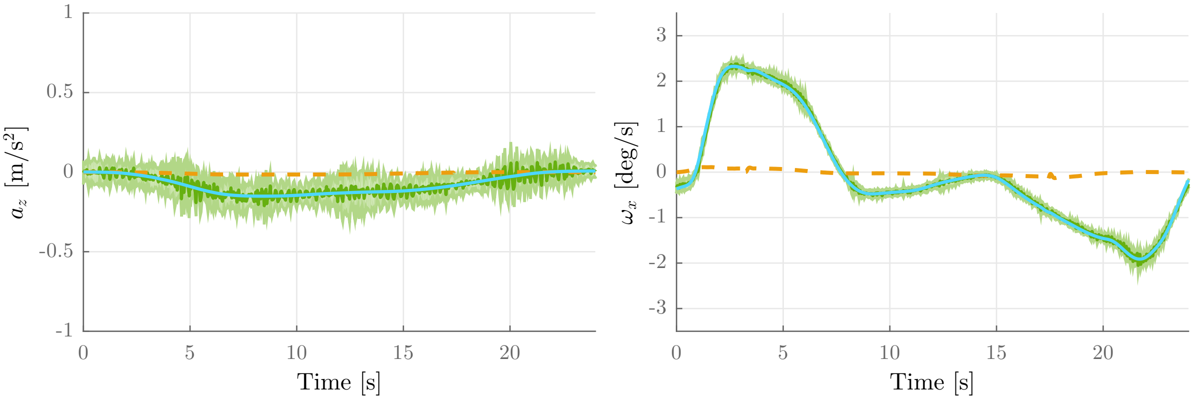
(f) Yaw rate ω_z.

Figure E-15: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C15. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



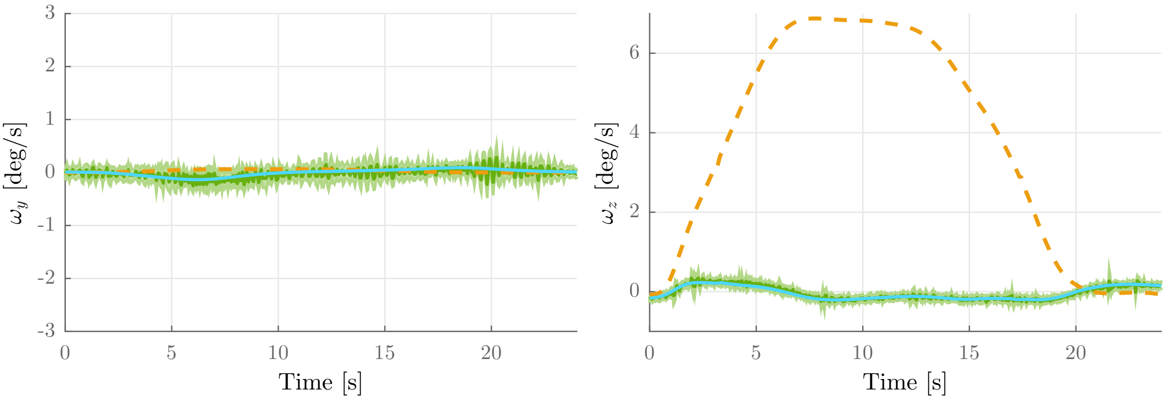
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

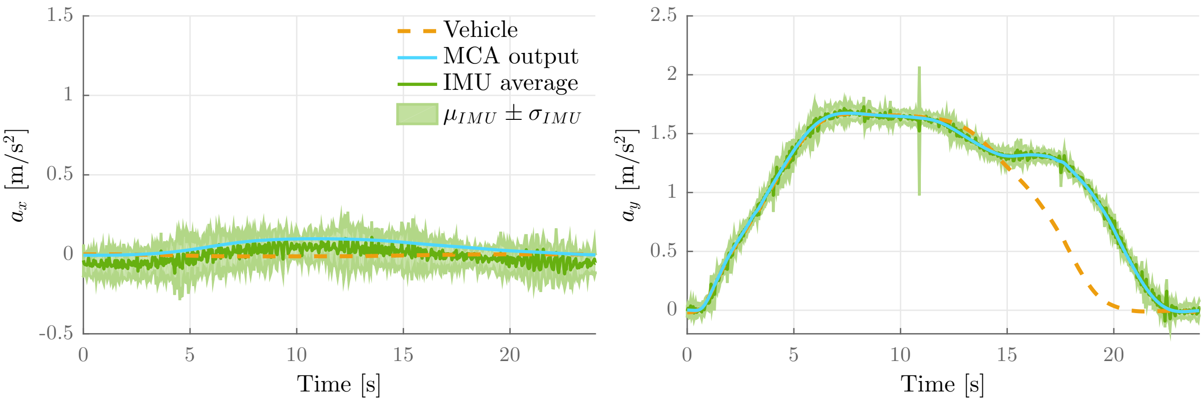
(d) Roll rate ω_x.



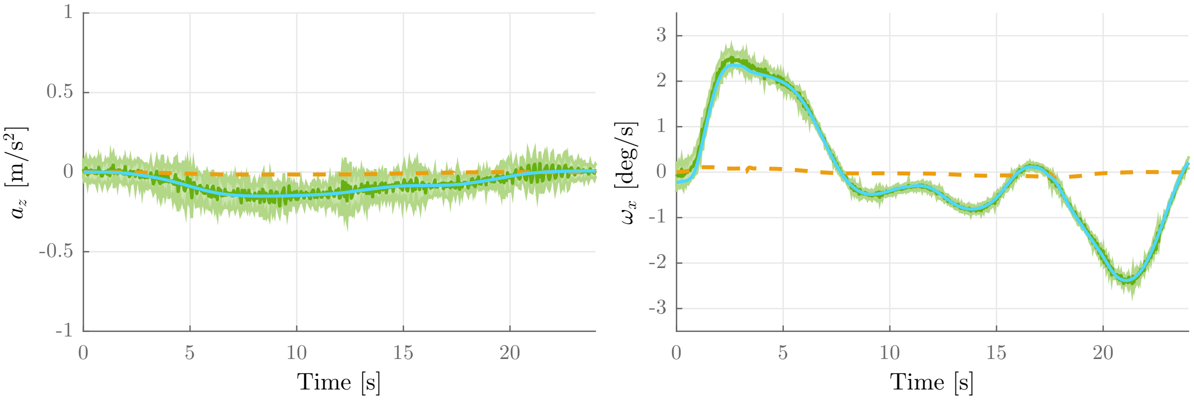
(e) Pitch rate ω_y.

(f) Yaw rate ω_z.

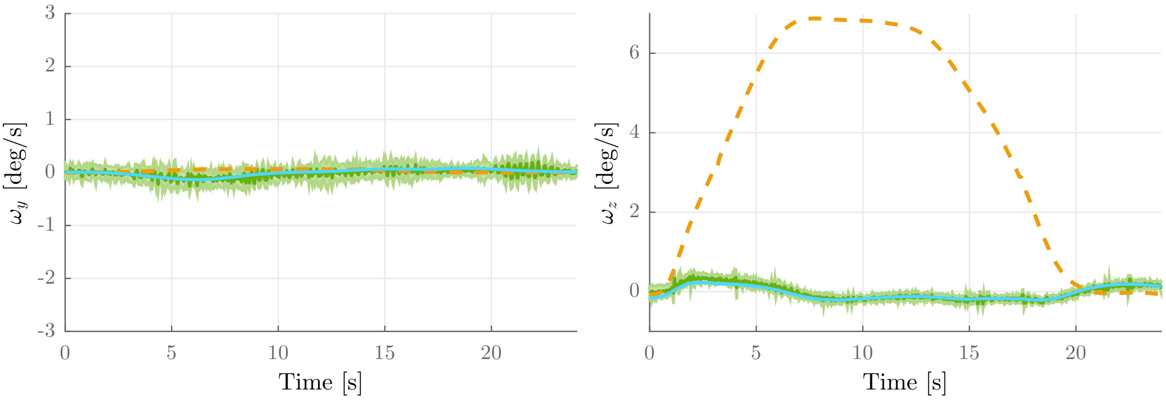
Figure E-16: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C16. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



(a) Longitudinal specific force a_x. **(b)** Lateral specific force a_y.

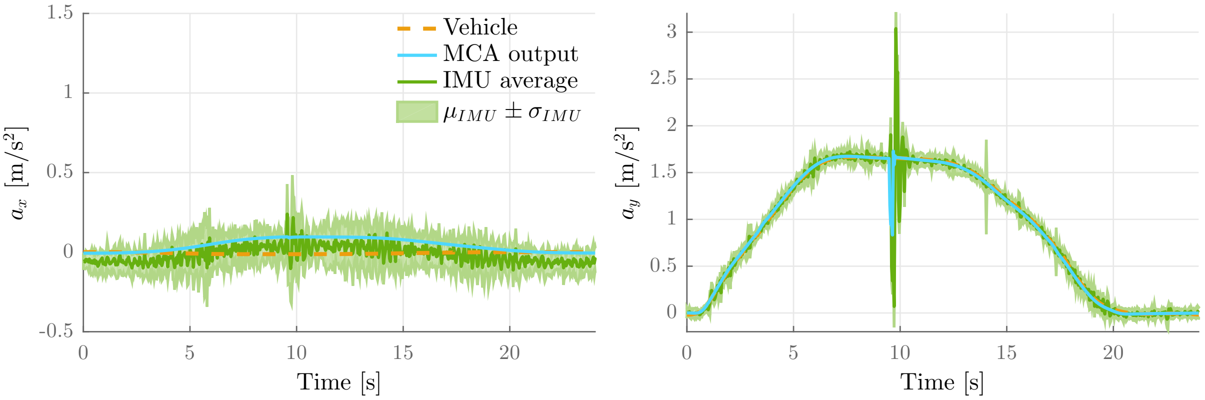


(c) Vertical specific force a_z. **(d)** Roll rate ω_x.



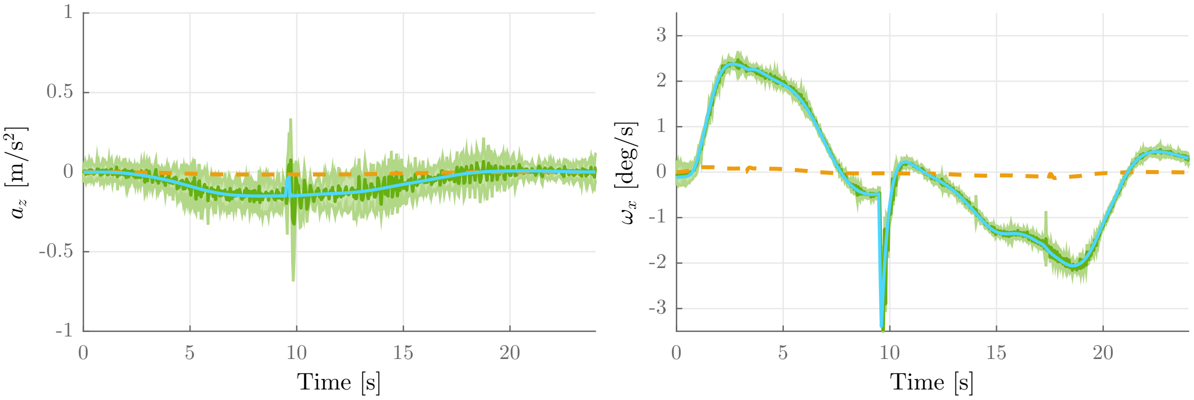
(e) Pitch rate ω_y. **(f)** Yaw rate ω_z.

Figure E-17: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C17. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



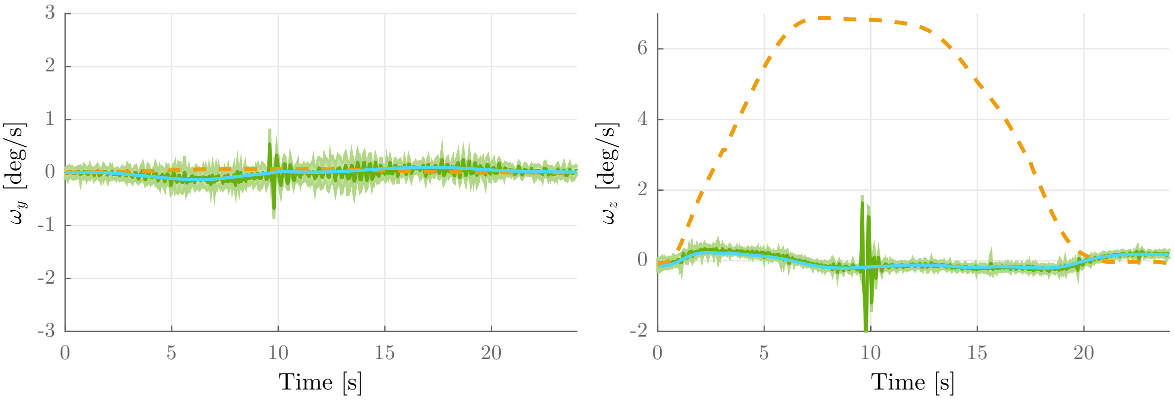
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

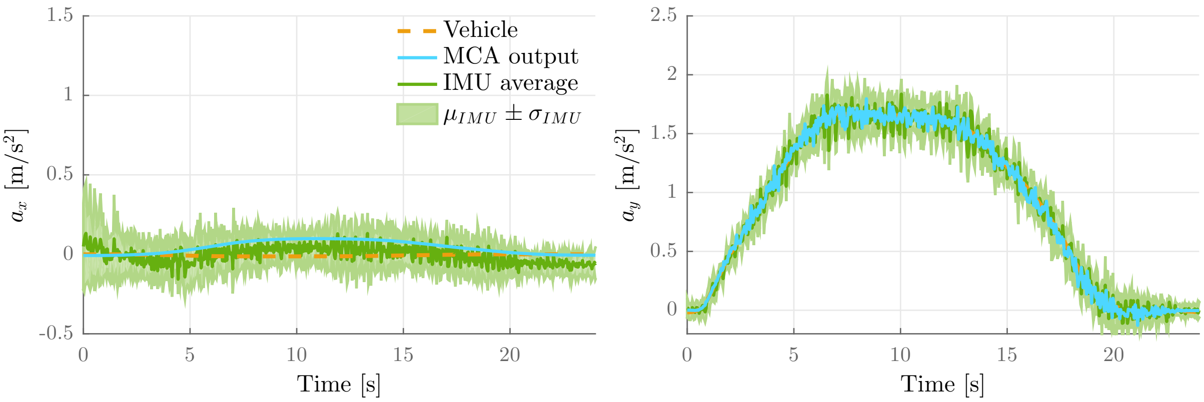
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

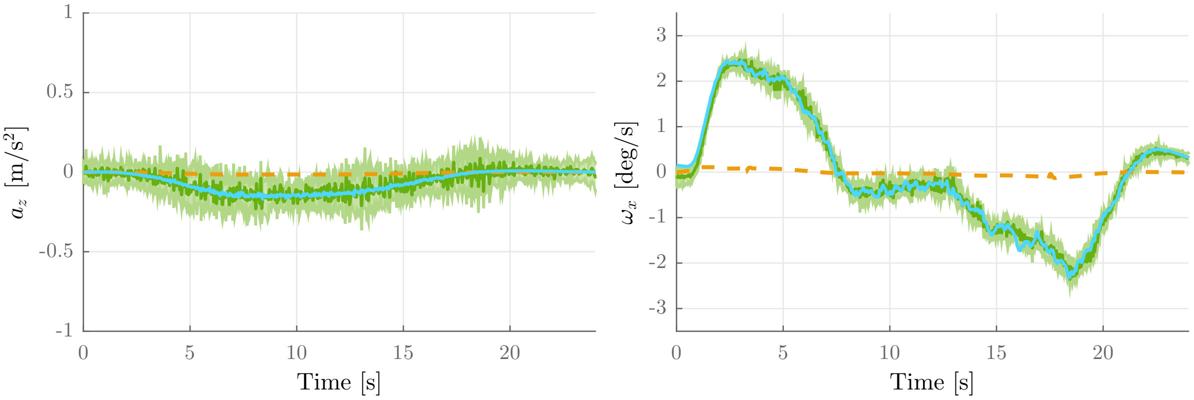
(f) Yaw rate ω_z.

Figure E-18: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C18. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



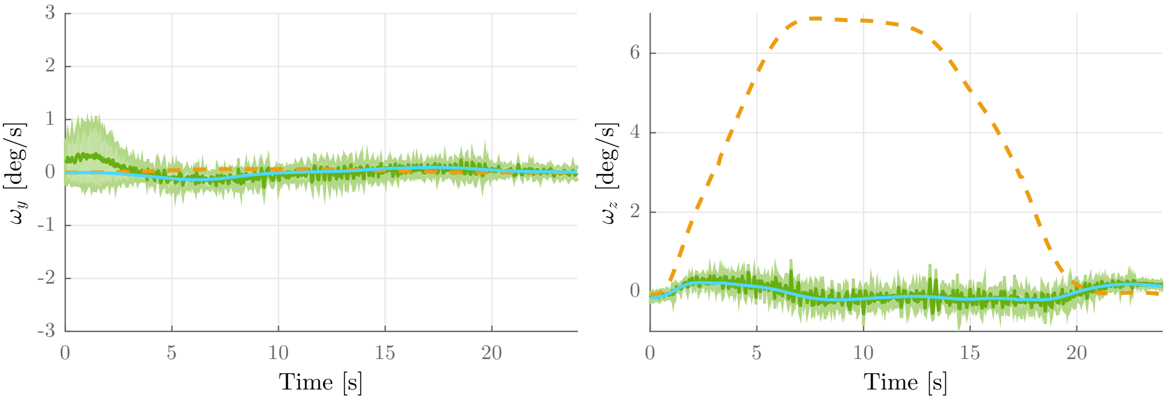
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

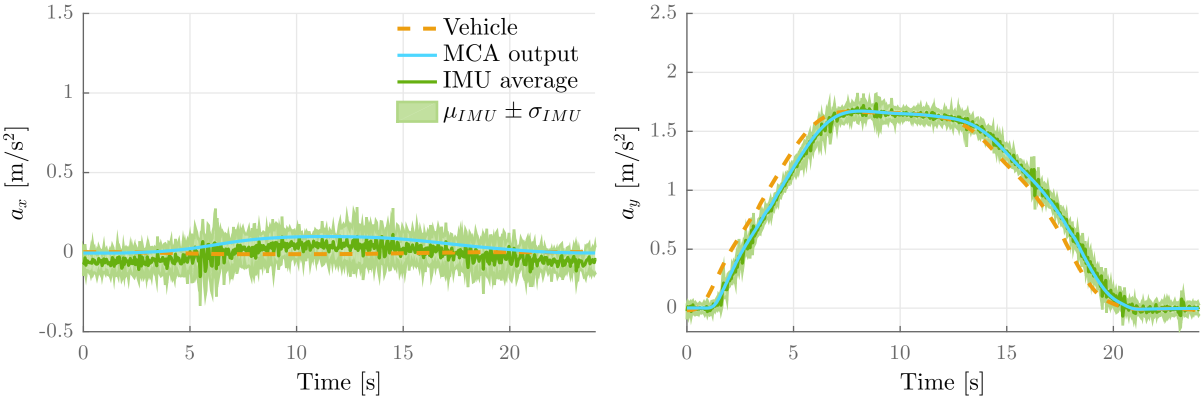
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

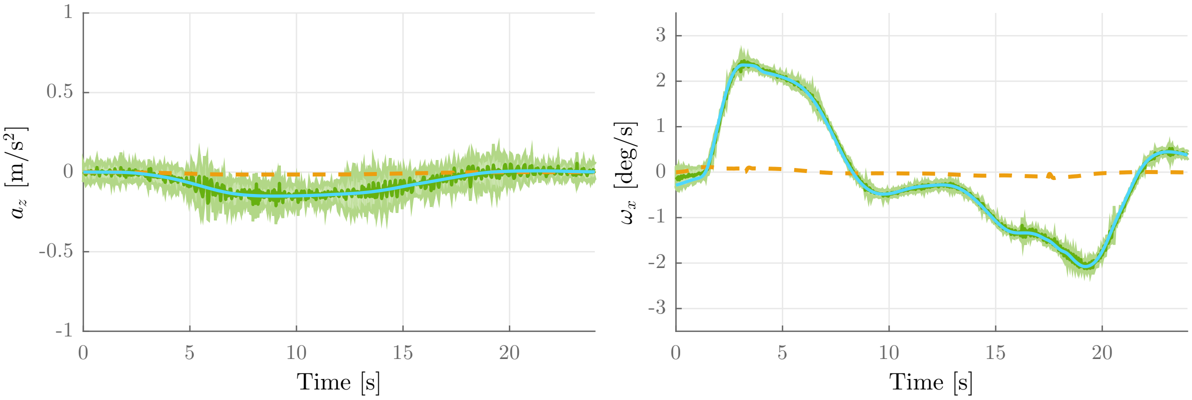
(f) Yaw rate ω_z.

Figure E-19: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C19. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



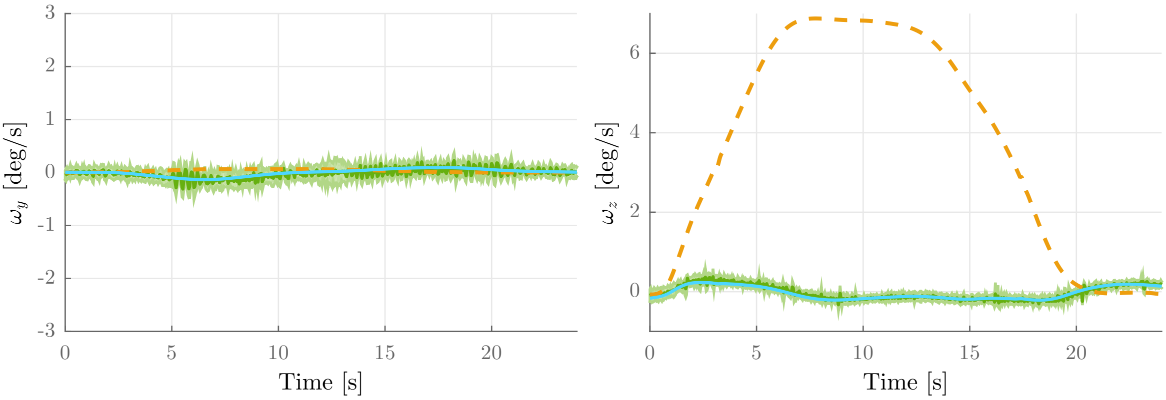
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

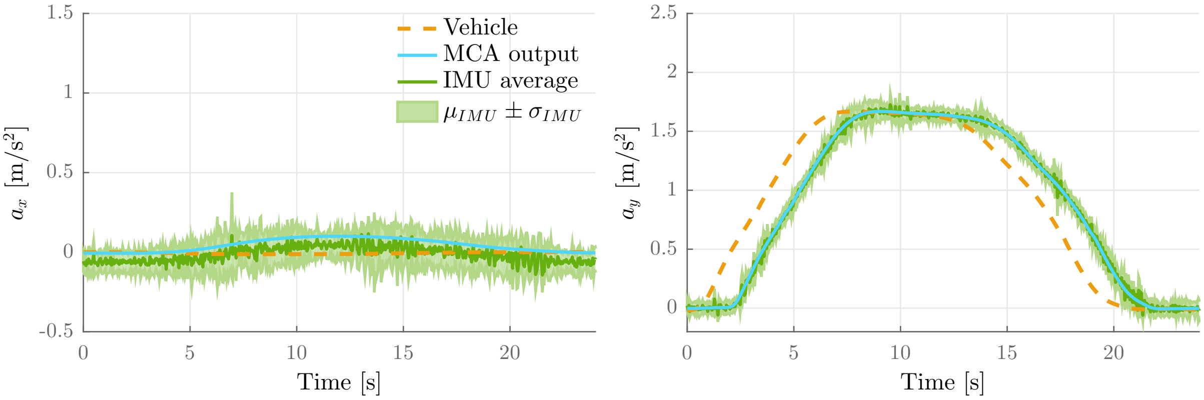
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

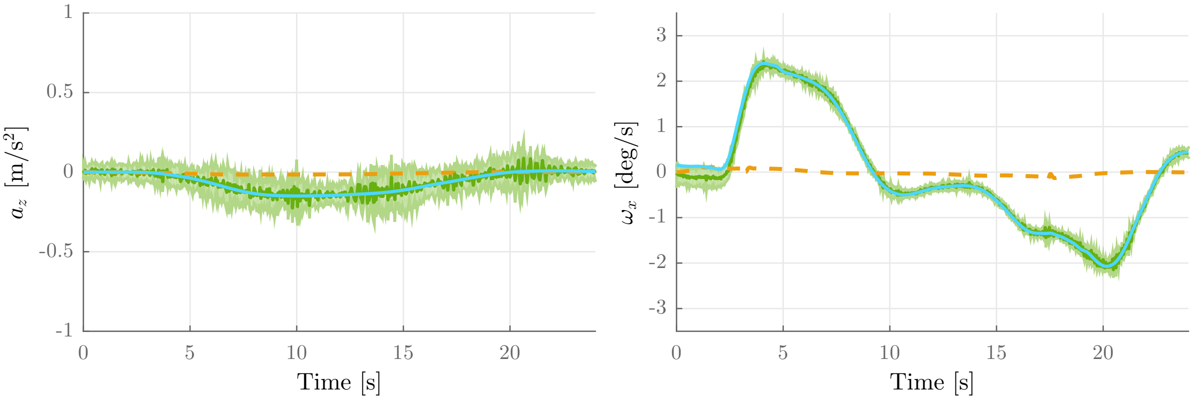
(f) Yaw rate ω_z.

Figure E-20: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C20. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



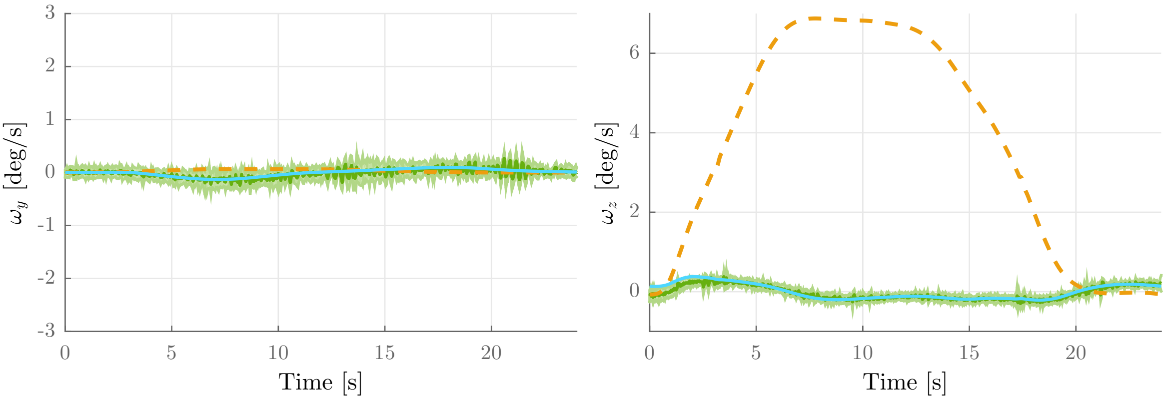
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

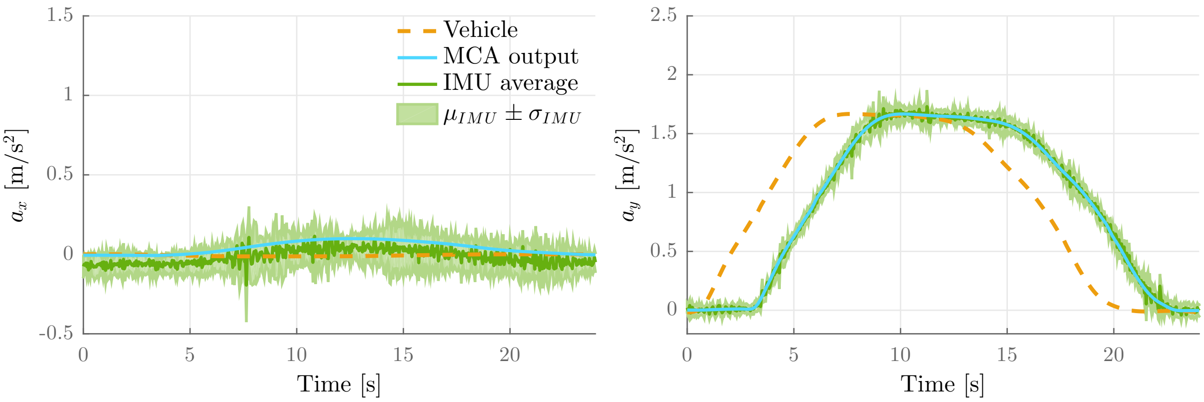
(d) Roll rate ω_x.



(e) Pitch rate ω_y.

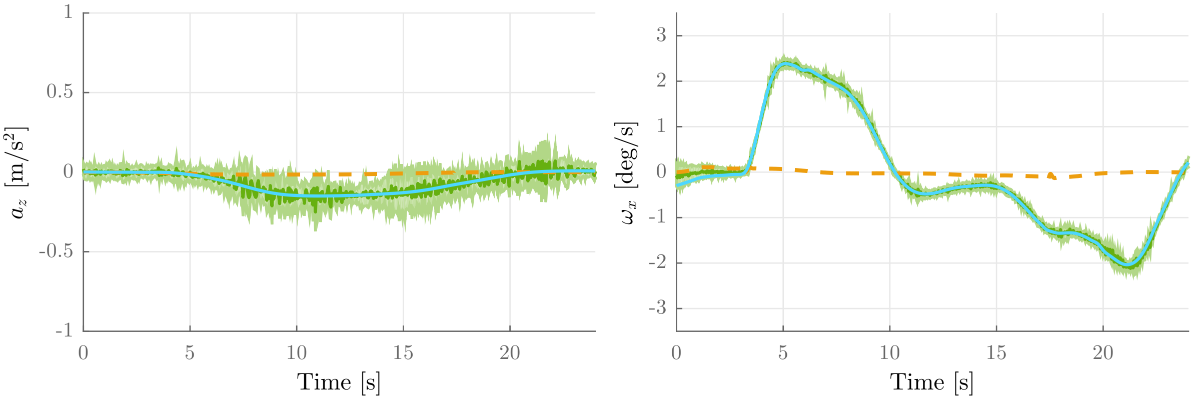
(f) Yaw rate ω_z.

Figure E-21: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C21. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.



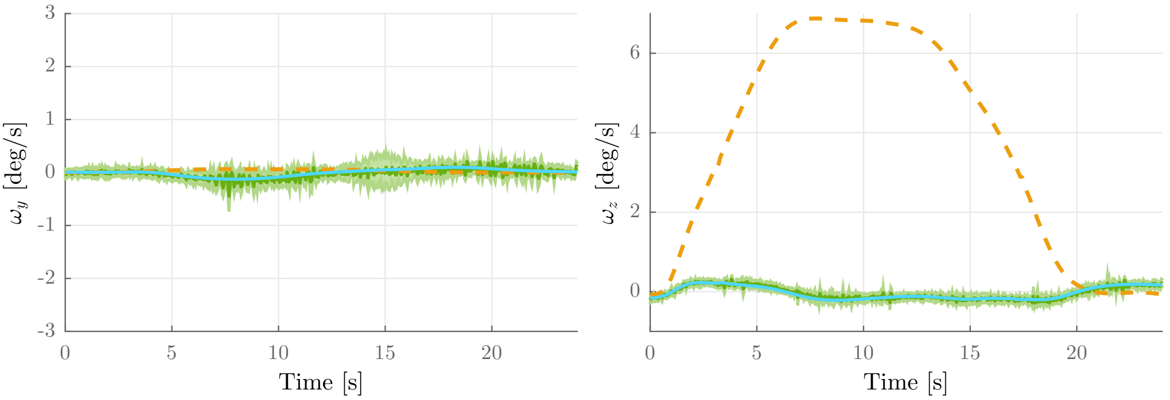
(a) Longitudinal specific force a_x.

(b) Lateral specific force a_y.



(c) Vertical specific force a_z.

(d) Roll rate ω_x.



(e) Pitch rate ω_y.

(f) Yaw rate ω_z.

Figure E-22: Comparison between the inertial motion experienced in the real vehicle during a left-hand turn (shown in yellow), the inertial motion simulated by the MCA (blue) and the inertial motion experienced by subjects in the CyberMotion Simulator (green) in condition C22. Data of the CMS inertial motion was obtained using an IMU. IMU-data for right-hand curves has been switched sign for a_y, ω_x and ω_z.

E-3 Discussion

From figures E-1 through E-22 it can be seen that the CMS was capable of accurately reproducing the inertial motion as desired by the MCA. Two conditions stand out in terms of the match between the MCA output and the CMS inertial motion:

Condition C14 - From figure E-14 one can observe the IMU measurements to show strange behaviour from $t = 15$ s onwards. Further research into the motion for this condition (which represents an F1 error) revealed that, unlike for all other conditions, the motion in a left- and right-handed curve was not mirrored for this condition, see figure E-23. This was attributed to a mistyping in setting up the conditions, the intention was to make all curves like the right-handed one (with the second peak at $t = 17$ s).

Condition C14 was used to compare different motion cueing error types. As such, only the maximum average rating of all subjects was used as a metric. To investigate the influence of this mistake, the maximum average ratings over the entire run were compared to the maximum average ratings before $t = 15$ s (the time at which the motion of opposite sign starts), see table E-1.

Only four (out of sixteen) subjects gave their maximum rating after $t = 15$ s. This is a surprising result in itself, as the maximum ratings were expected during the part of the curve featuring a lateral specific force in the wrong direction. Of these four, only one lowers the maximum rating by more than five points (out of a hundred). A t-test between these two distributions showed no significant effect at an $\alpha = 0.05$ significance level ($p = 0.22$). It is therefore concluded that this mistake had little effect on the results of this study.

Condition C18 - Condition C18 (see figure E-18) features an F3a error at $t = 9.5$ s. This desired sudden motion however is presented too strong by the CMS. Also, vibrations are introduced in the CMS, which take around 0.6 seconds to die out. This is close to (but slightly higher than) the $\Delta t = 0.5$ s threshold used in the IEEE paper for distinguishing F3a from F3b errors.

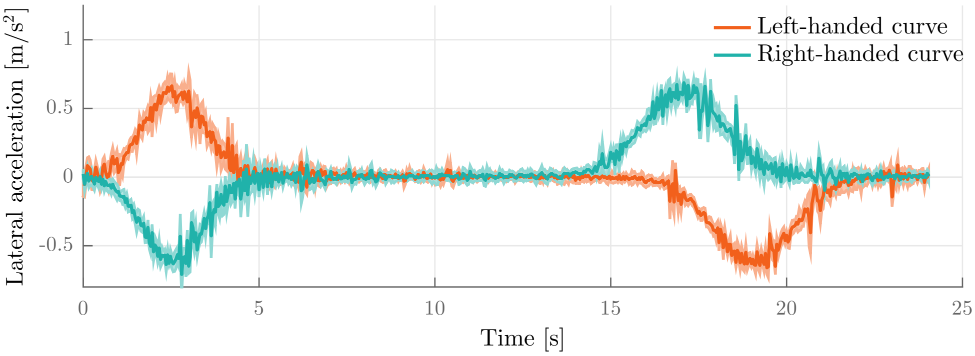


Figure E-23: Lateral specific force a_y distribution ($\mu \pm 1\sigma$) for left-handed and right-handed curves featuring condition C14.

Table E-1: Maximum average ratings $\bar{R}_{\max}$ for condition C14 (which features an F1 error), excluding and including the curve exit.

Participant	$\bar{R}_{\max}$ **before** $t = 15$s	$\bar{R}_{\max}$ **entire curve**
1	71.3	72.4
2	99.4	99.4
3	94.5	94.5
4	94.0	94.0
5	98.9	98.9
6	77.4	77.4
7	98.8	98.8
8	49.1	49.1
9	73.5	78.1
10	65.2	65.2
11	29.3	29.3
12	86.6	86.6
13	99.3	99.3
14	54.9	54.9
15	98.8	99.5
16	52.3	77.1

Appendix F

Experiment Ratings

During the experiment described in the IEEE paper, all 16 participants experienced four runs during the experiment phase. This leads to a total of 64 ratings provided for each motion condition. In this appendix, the ratings presented in the IEEE paper are broken down further. In section F-1 the average ratings (over four runs) of individual subjects are shown for all 22 conditions of the experiment. Single-run ratings for all subjects are shown in section F-2.

F-1 Subject average ratings

Figures 8, 11 and 13 of the IEEE paper showed the average ratings over all subjects for various conditions, and gave an indication of the spread of the ratings. In this section, the average ratings for individual subjects are shown for all conditions in figures F-1 through F-22.

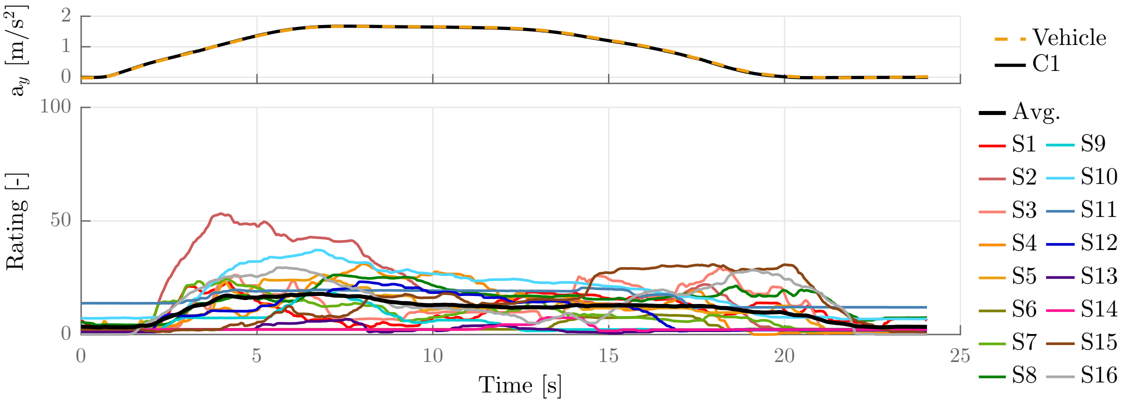


Figure F-1: Average rating for individual subjects during a curve of Condition C1.

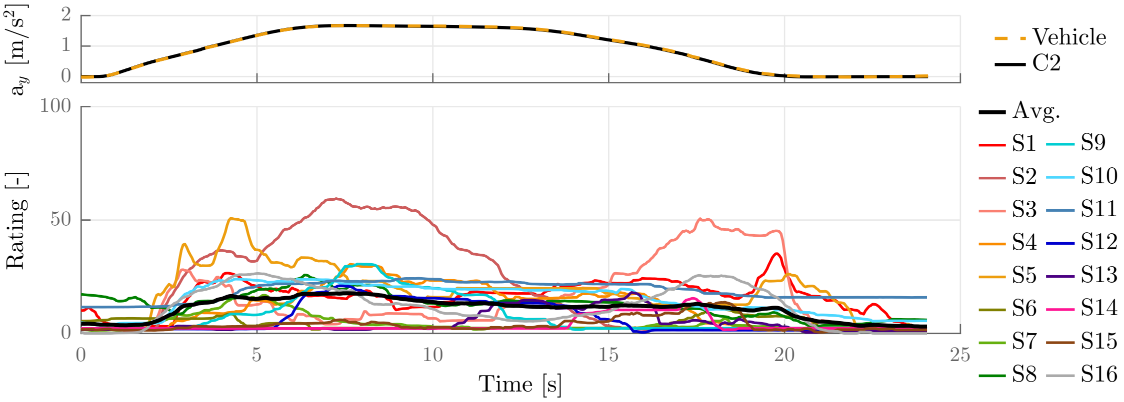


Figure F-2: Average rating for individual subjects during a curve of Condition C2.

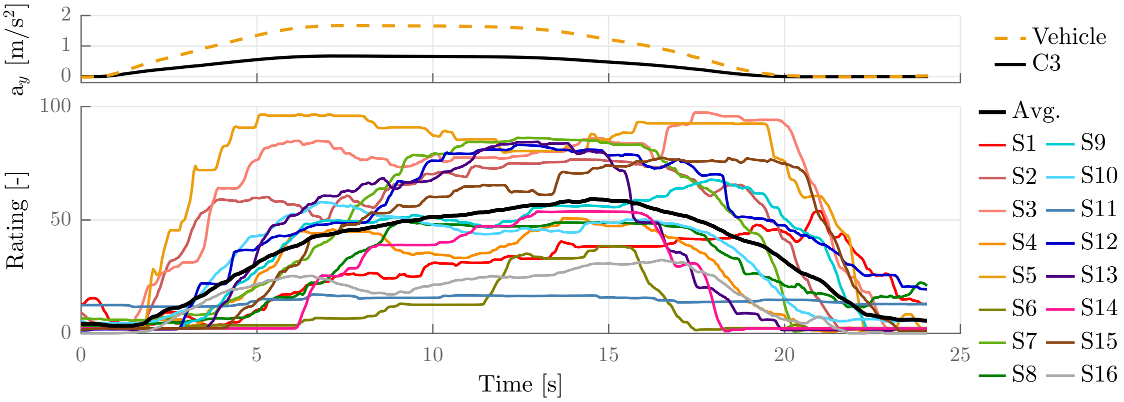


Figure F-3: Average rating for individual subjects during a curve of Condition C3.

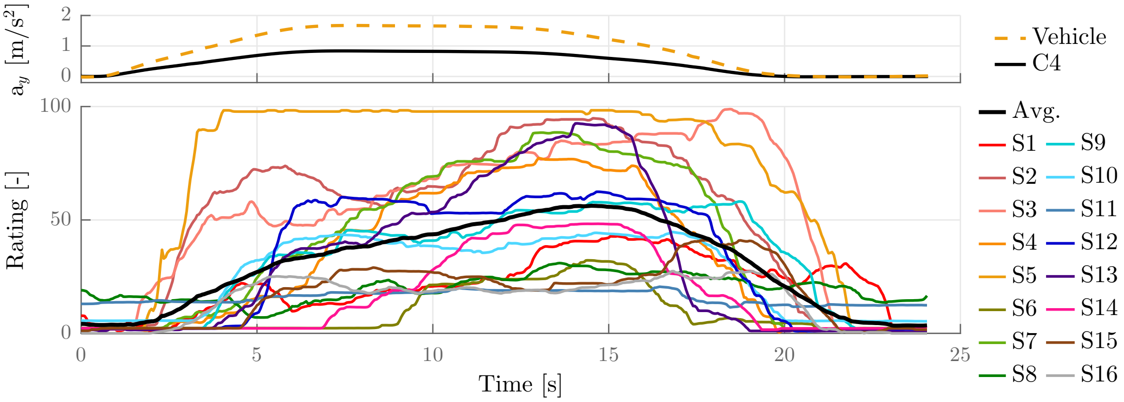


Figure F-4: Average rating for individual subjects during a curve of Condition C4.

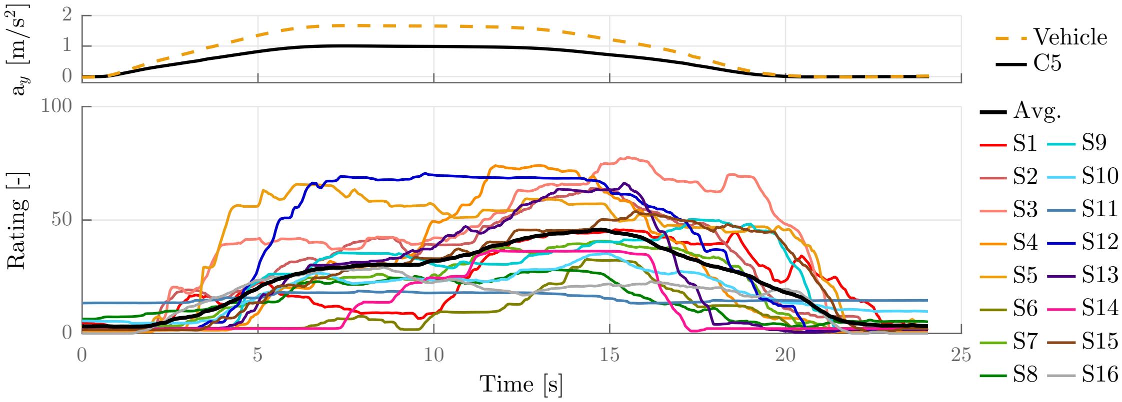


Figure F-5: Average rating for individual subjects during a curve of Condition C5.

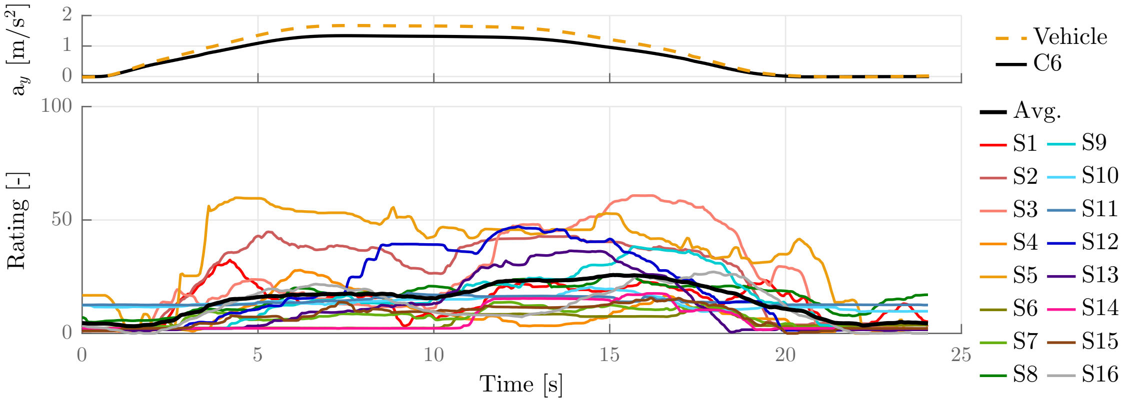


Figure F-6: Average rating for individual subjects during a curve of Condition C6.

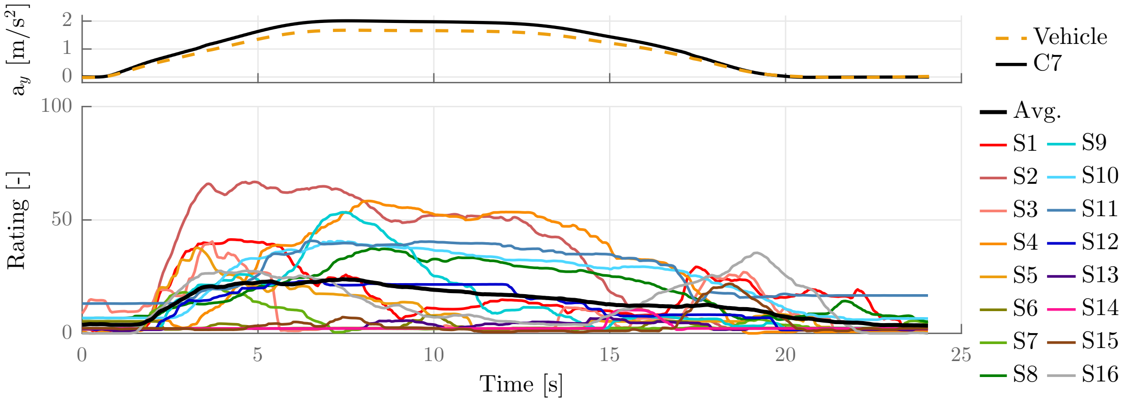


Figure F-7: Average rating for individual subjects during a curve of Condition C7.

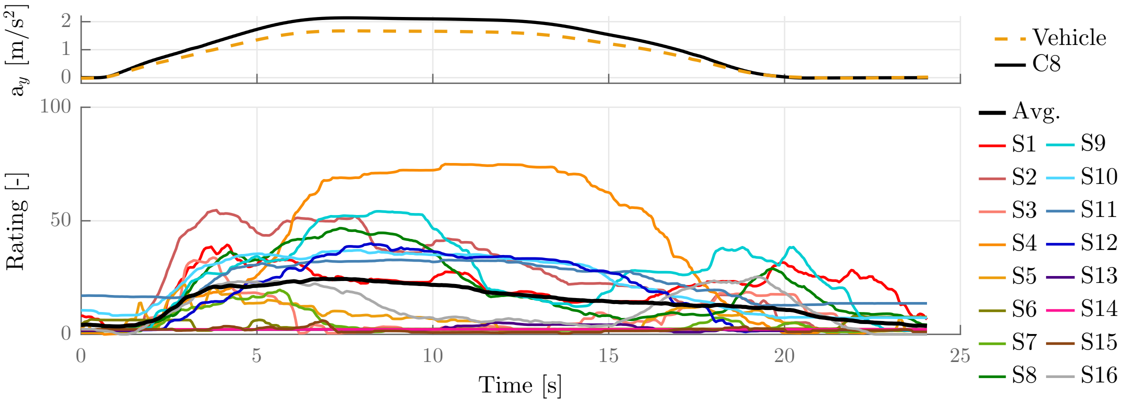


Figure F-8: Average rating for individual subjects during a curve of Condition C8.

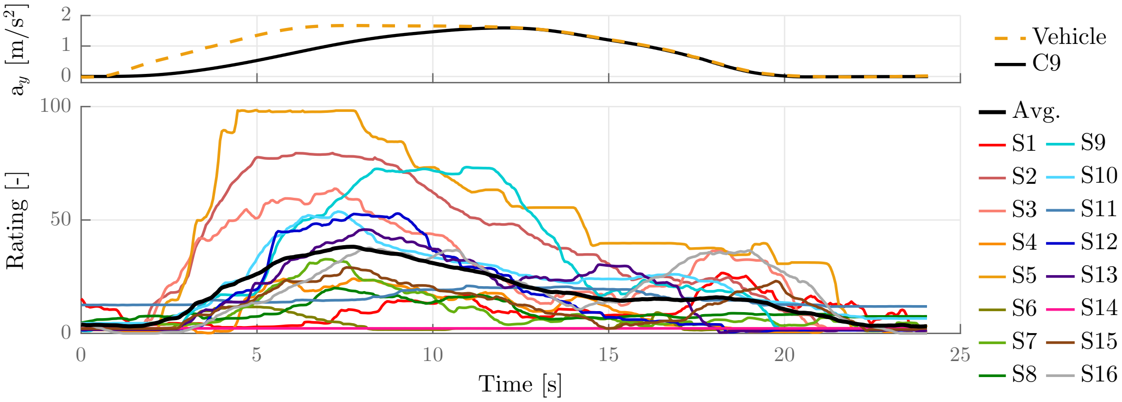


Figure F-9: Average rating for individual subjects during a curve of Condition C9.

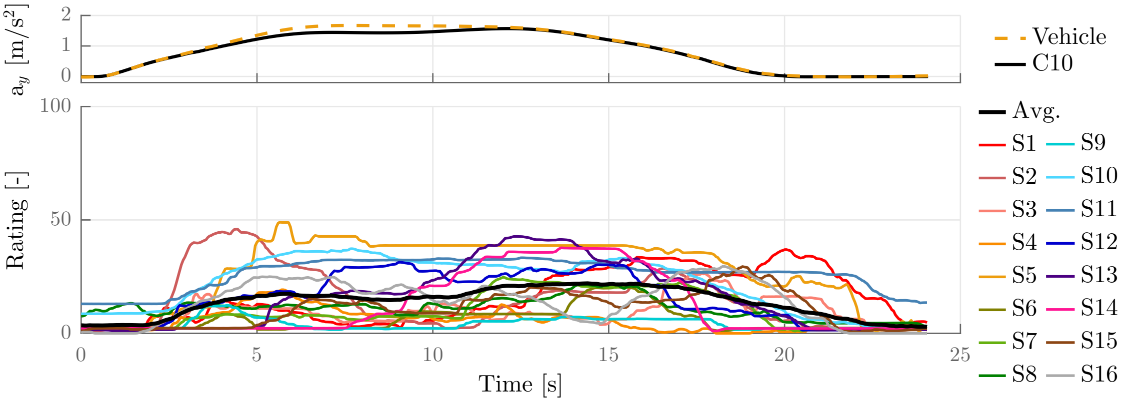


Figure F-10: Average rating for individual subjects during a curve of Condition C10.

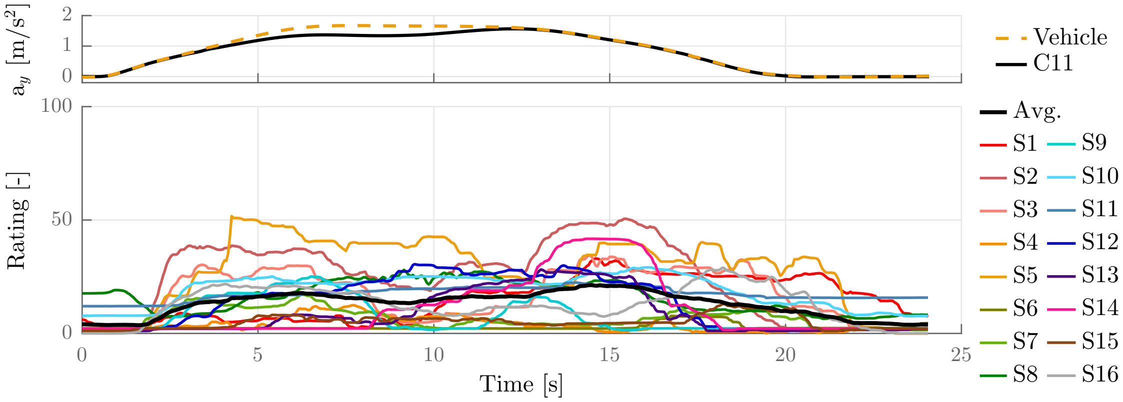


Figure F-11: Average rating for individual subjects during a curve of Condition C11.

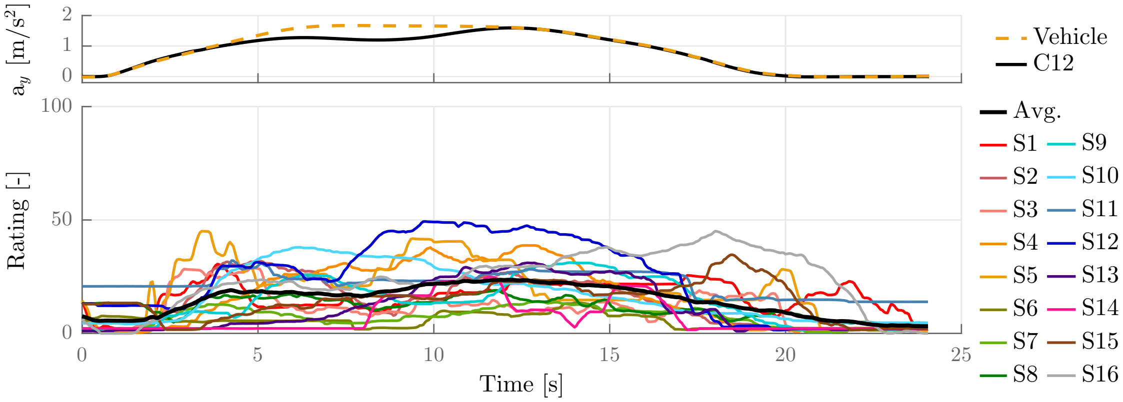


Figure F-12: Average rating for individual subjects during a curve of Condition C12.

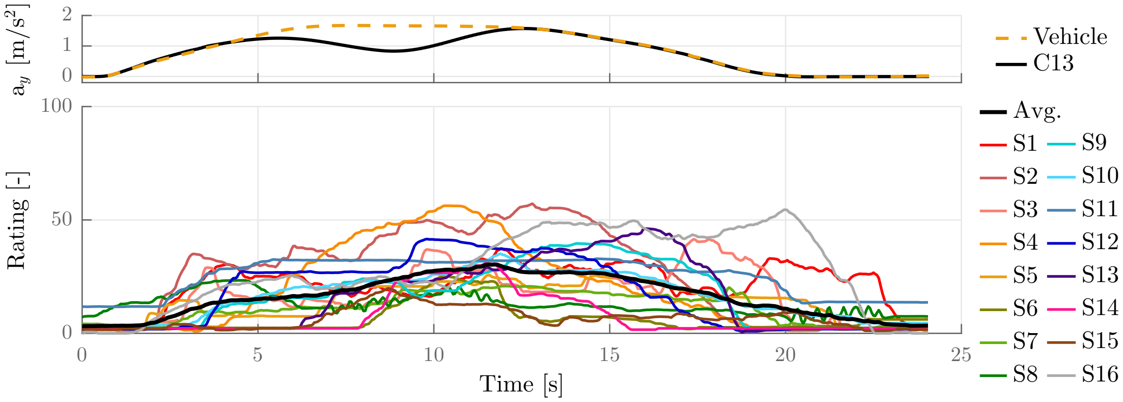


Figure F-13: Average rating for individual subjects during a curve of Condition C13.

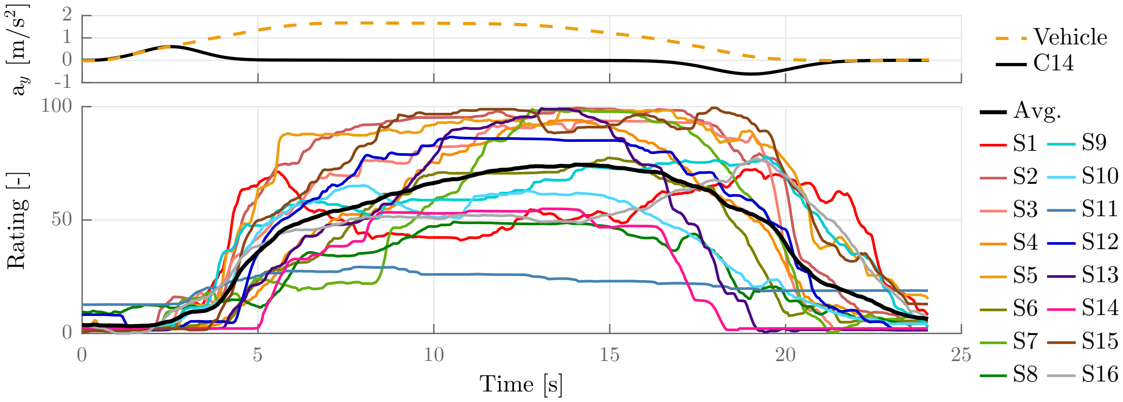


Figure F-14: Average rating for individual subjects during a curve of Condition C14.

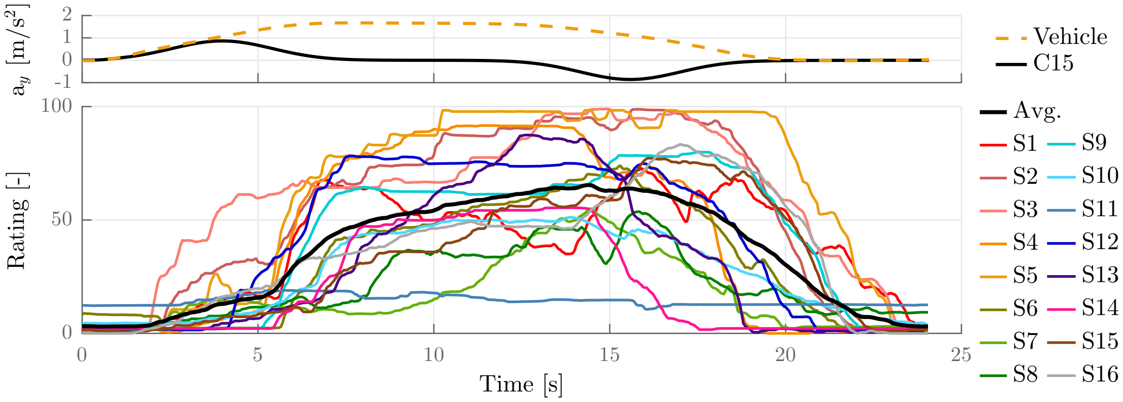


Figure F-15: Average rating for individual subjects during a curve of Condition C15.

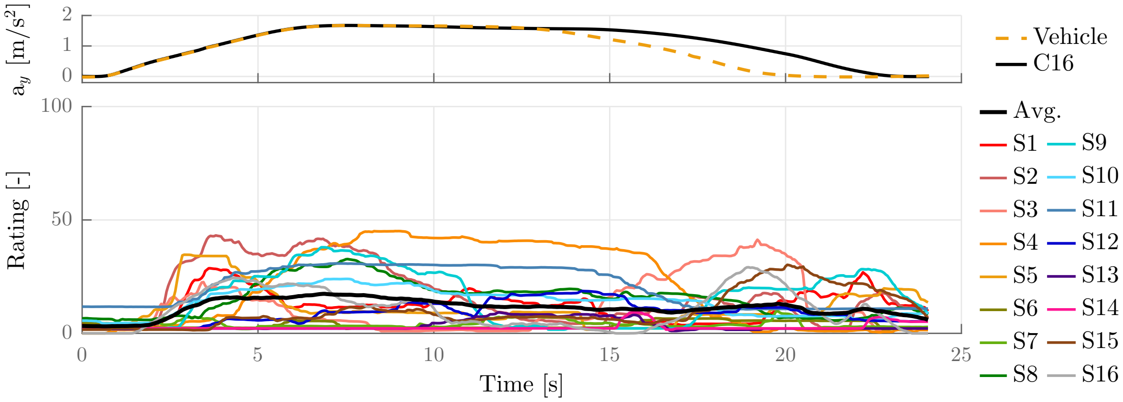


Figure F-16: Average rating for individual subjects during a curve of Condition C16.

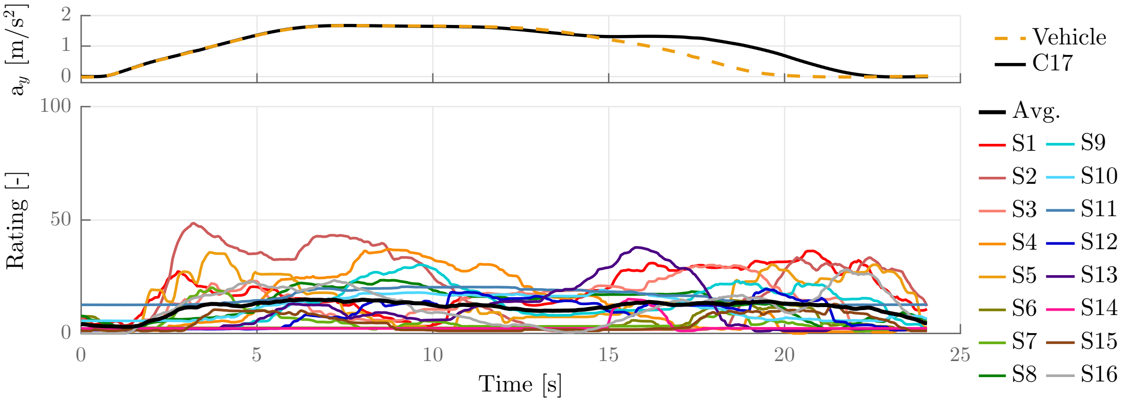


Figure F-17: Average rating for individual subjects during a curve of Condition C17.

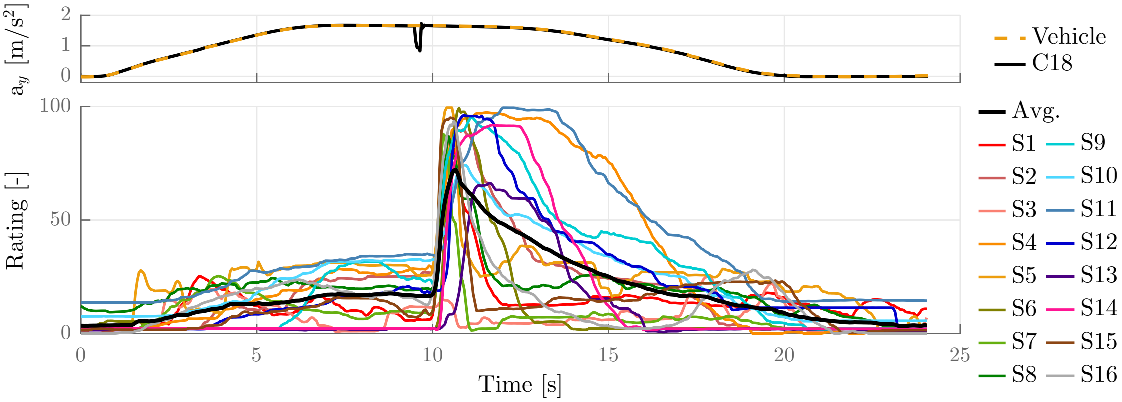


Figure F-18: Average rating for individual subjects during a curve of Condition C18.

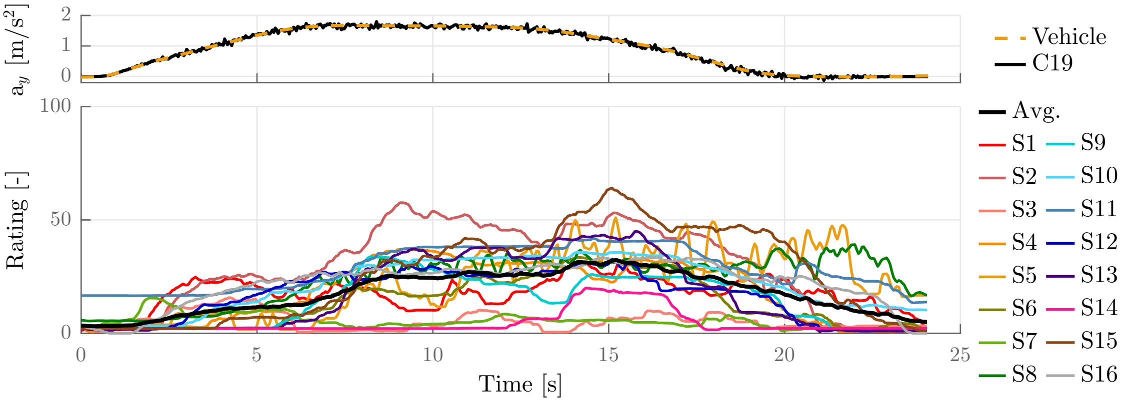


Figure F-19: Average rating for individual subjects during a curve of Condition C19.

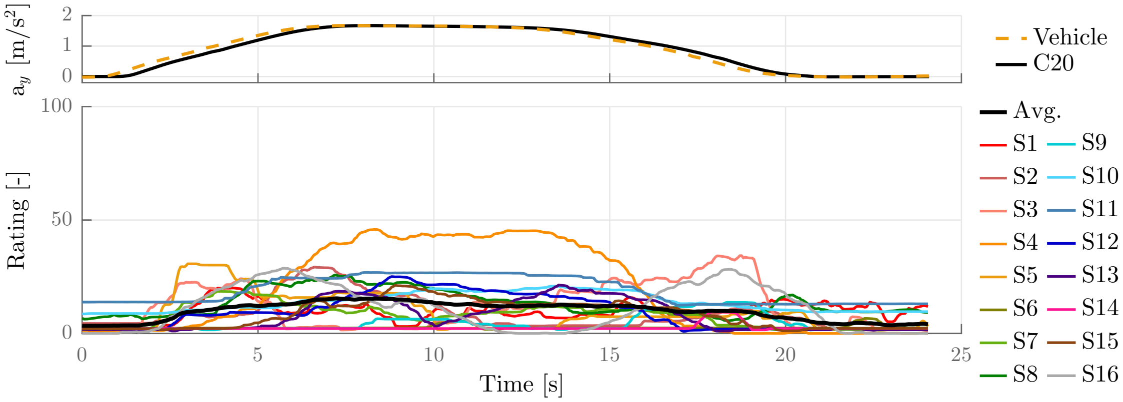


Figure F-20: Average rating for individual subjects during a curve of Condition C20.

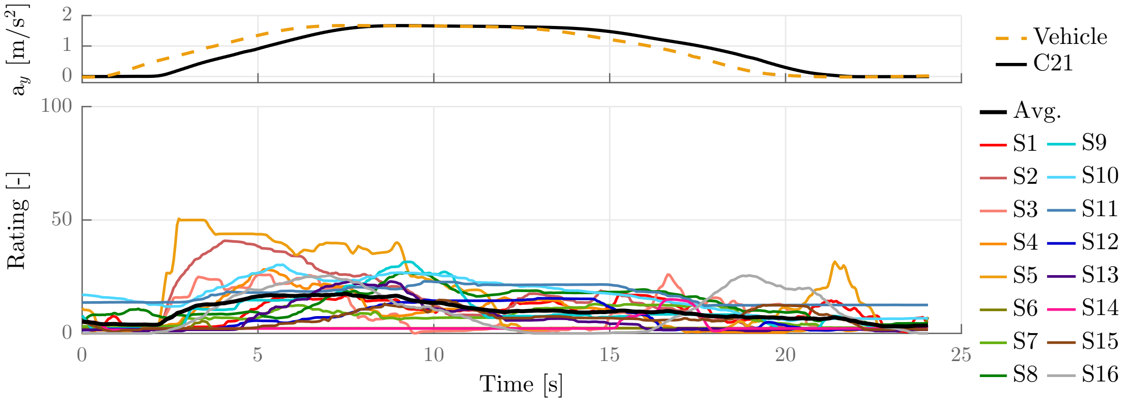


Figure F-21: Average rating for individual subjects during a curve of Condition C21.

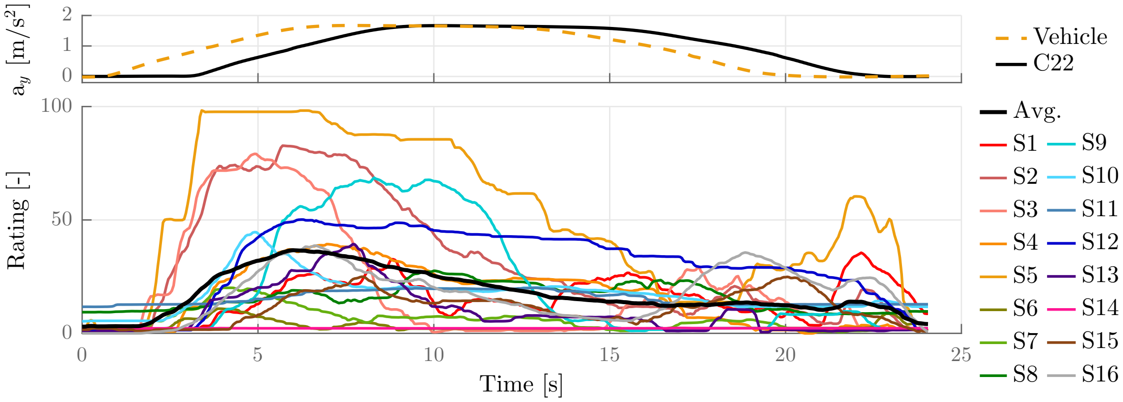


Figure F-22: Average rating for individual subjects during a curve of Condition C22.

F-2 Single-run ratings

In the figures shown in the previous section, all ratings for individual subjects were the average of four repetitions of all conditions. For subject 4 only the final three repetitions were used, as the subject indicated to have misunderstood the instructions after the first experiment run. The ratings for each repetition, for all subjects and all conditions are shown in figures F-23 through F-374. For subject 4, only the three final ratings are shown.

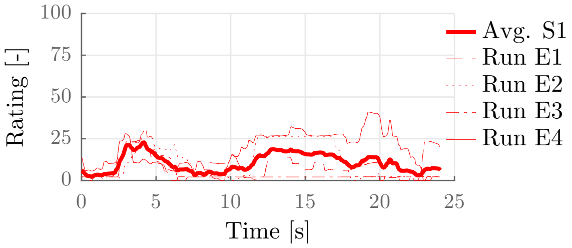


Figure F-23: Ratings by subject S1 for Condition C1. The mean rating is repeated in figure F-1.

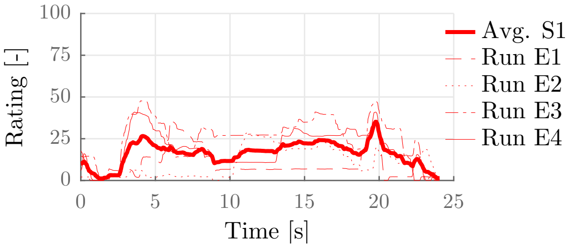


Figure F-24: Ratings by subject S1 for Condition C2. The mean rating is repeated in figure F-2.

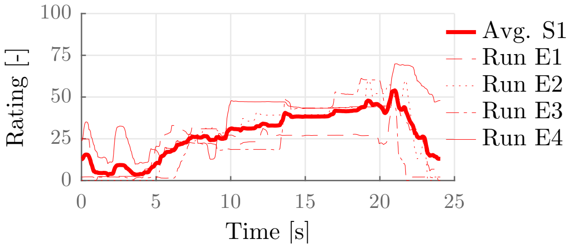


Figure F-25: Ratings by subject S1 for Condition C3. The mean rating is repeated in figure F-3.

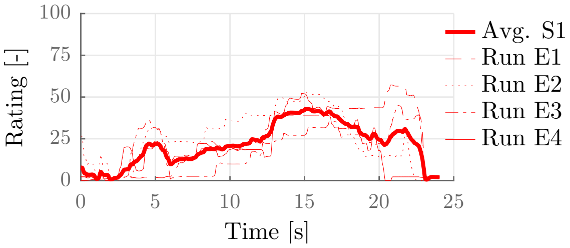


Figure F-26: Ratings by subject S1 for Condition C4. The mean rating is repeated in figure F-4.

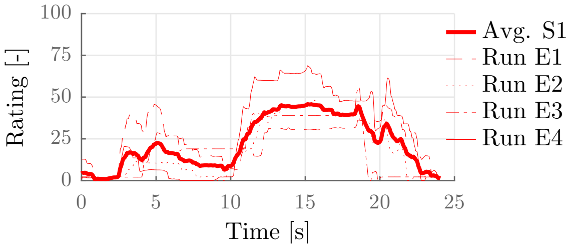


Figure F-27: Ratings by subject S1 for Condition C5. The mean rating is repeated in figure F-5.

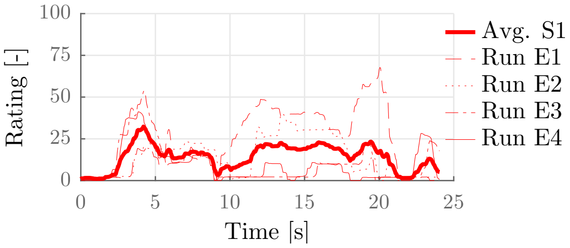


Figure F-28: Ratings by subject S1 for Condition C6. The mean rating is repeated in figure F-6.

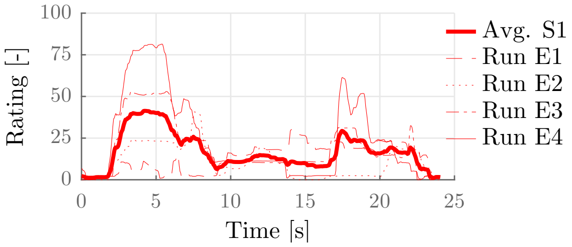


Figure F-29: Ratings by subject S1 for Condition C7. The mean rating is repeated in figure F-7.

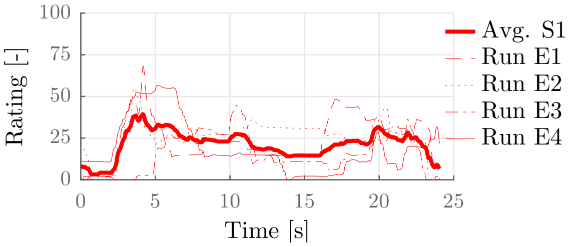


Figure F-30: Ratings by subject S1 for Condition C8. The mean rating is repeated in figure F-8.

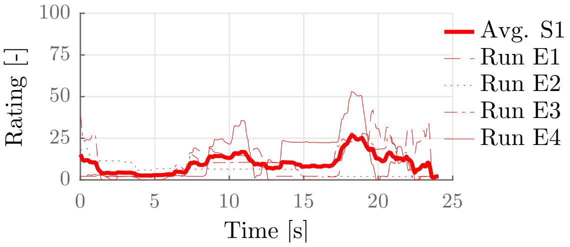


Figure F-31: Ratings by subject S1 for Condition C9. The mean rating is repeated in figure F-9.

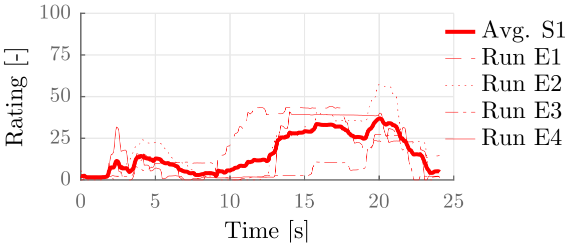


Figure F-32: Ratings by subject S1 for Condition C10. The mean rating is repeated in figure F-10.

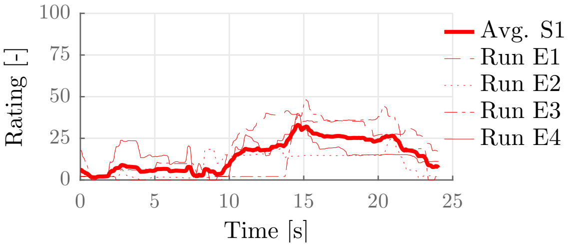


Figure F-33: Ratings by subject S1 for Condition C11. The mean rating is repeated in figure F-11.

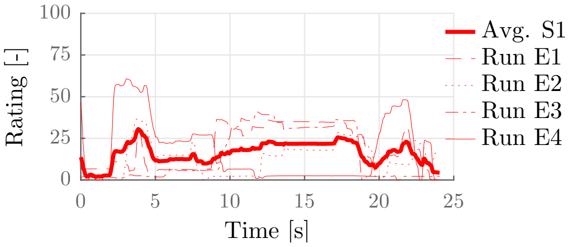


Figure F-34: Ratings by subject S1 for Condition C12. The mean rating is repeated in figure F-12.

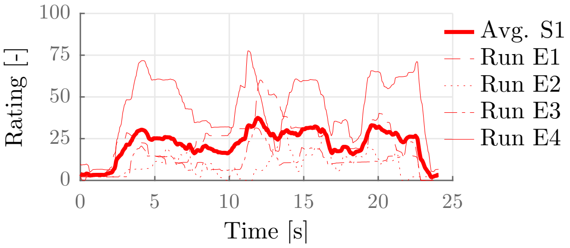


Figure F-35: Ratings by subject S1 for Condition C13. The mean rating is repeated in figure F-13.

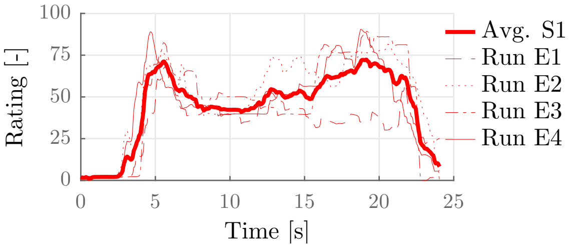


Figure F-36: Ratings by subject S1 for Condition C14. The mean rating is repeated in figure F-14.

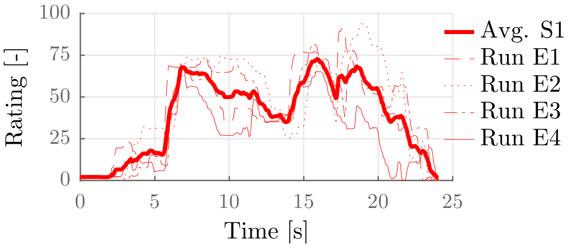


Figure F-37: Ratings by subject S1 for Condition C15. The mean rating is repeated in figure F-15.

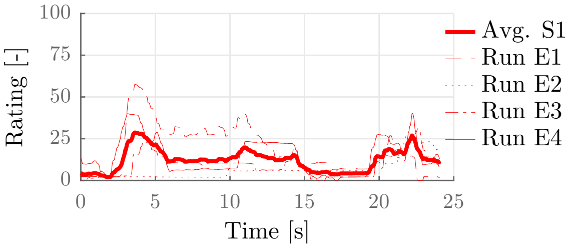


Figure F-38: Ratings by subject S1 for Condition C16. The mean rating is repeated in figure F-16.

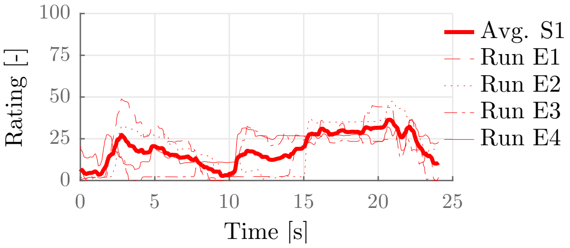


Figure F-39: Ratings by subject S1 for Condition C17. The mean rating is repeated in figure F-17.

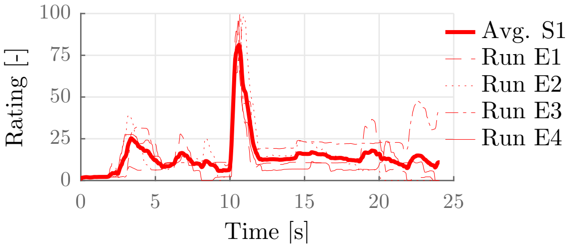
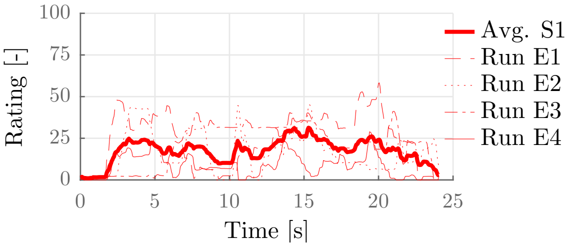


Figure F-40: Ratings by subject S1 for Condition C18. The mean rating is repeated in figure F-18.



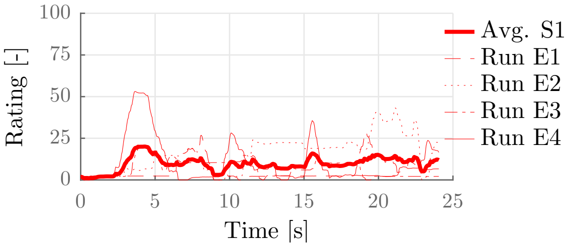
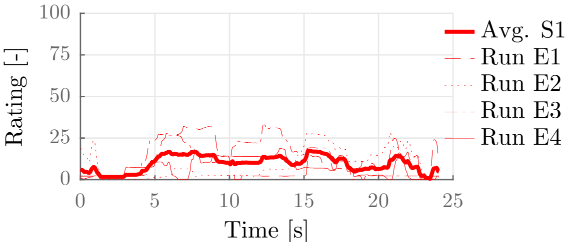


Figure F-41: Ratings by subject S1 for Condition C19. The mean rating is repeated in figure F-19.

Figure F-42: Ratings by subject S1 for Condition C20. The mean rating is repeated in figure F-20.



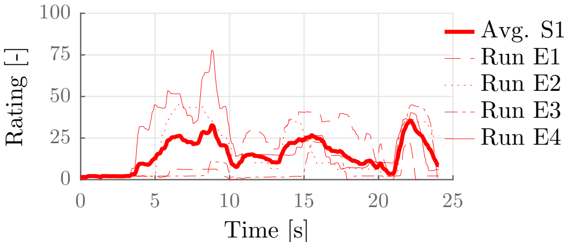
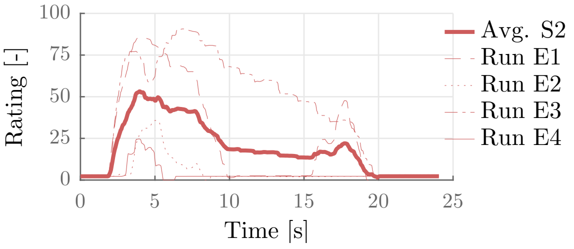


Figure F-43: Ratings by subject S1 for Condition C21. The mean rating is repeated in figure F-21.

Figure F-44: Ratings by subject S1 for Condition C22. The mean rating is repeated in figure F-22.



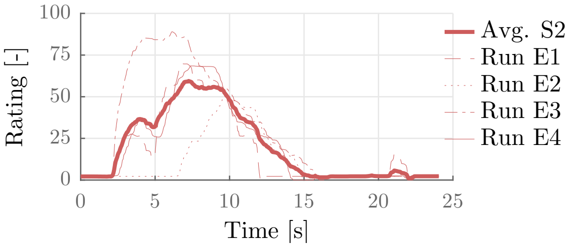
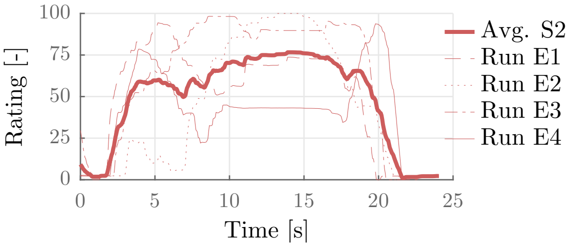


Figure F-45: Ratings by subject S2 for Condition C1. The mean rating is repeated in figure F-1.

Figure F-46: Ratings by subject S2 for Condition C2. The mean rating is repeated in figure F-2.



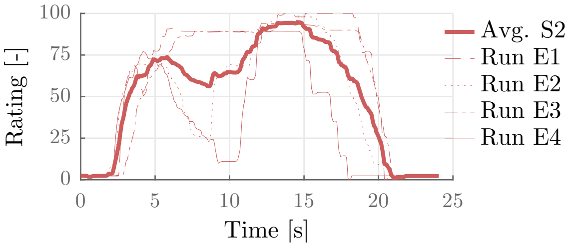
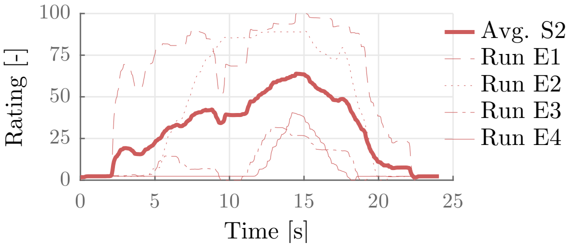


Figure F-47: Ratings by subject S2 for Condition C3. The mean rating is repeated in figure F-3.

Figure F-48: Ratings by subject S2 for Condition C4. The mean rating is repeated in figure F-4.



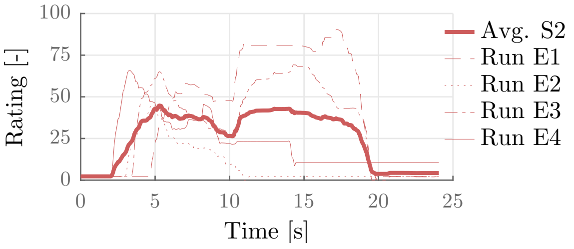


Figure F-49: Ratings by subject S2 for Condition C5. The mean rating is repeated in figure F-5.

Figure F-50: Ratings by subject S2 for Condition C6. The mean rating is repeated in figure F-6.

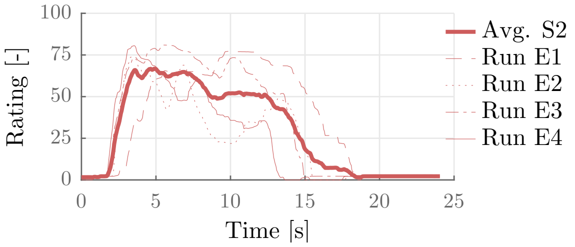


Figure F-51: Ratings by subject S2 for Condition C7. The mean rating is repeated in figure F-7.

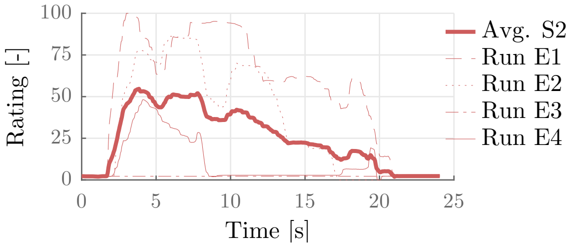


Figure F-52: Ratings by subject S2 for Condition C8. The mean rating is repeated in figure F-8.

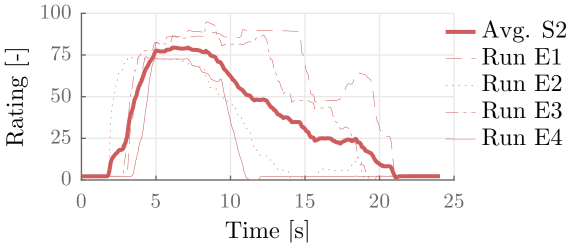


Figure F-53: Ratings by subject S2 for Condition C9. The mean rating is repeated in figure F-9.

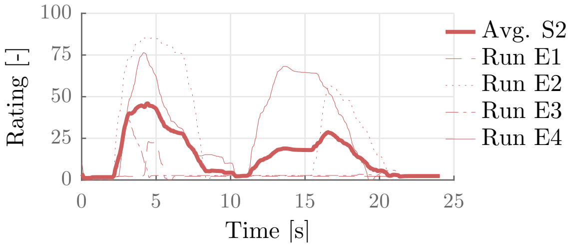


Figure F-54: Ratings by subject S2 for Condition C10. The mean rating is repeated in figure F-10.

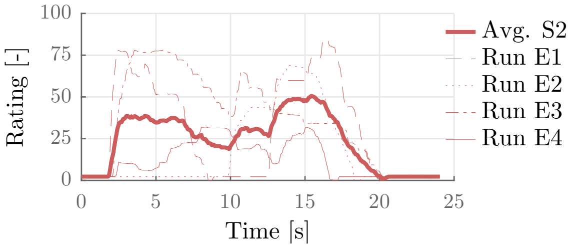


Figure F-55: Ratings by subject S2 for Condition C11. The mean rating is repeated in figure F-11.

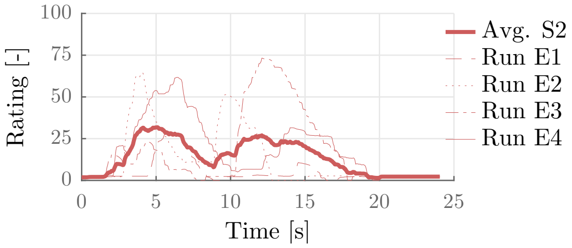


Figure F-56: Ratings by subject S2 for Condition C12. The mean rating is repeated in figure F-12.

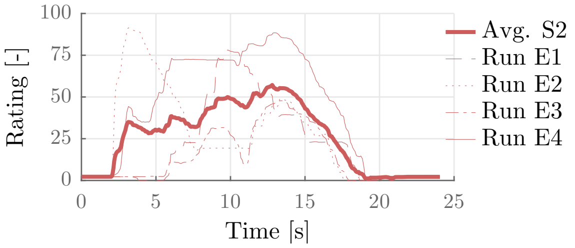


Figure F-57: Ratings by subject S2 for Condition C13. The mean rating is repeated in figure F-13.

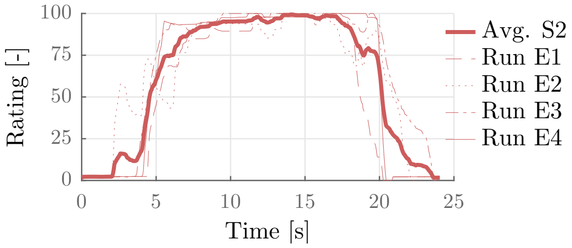


Figure F-58: Ratings by subject S2 for Condition C14. The mean rating is repeated in figure F-14.

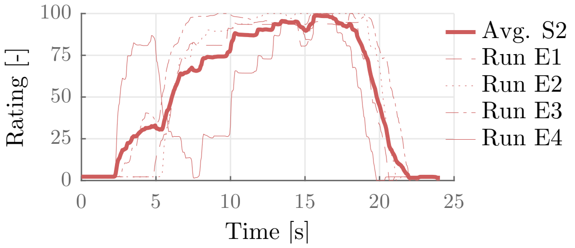


Figure F-59: Ratings by subject S2 for Condition C15. The mean rating is repeated in figure F-15.

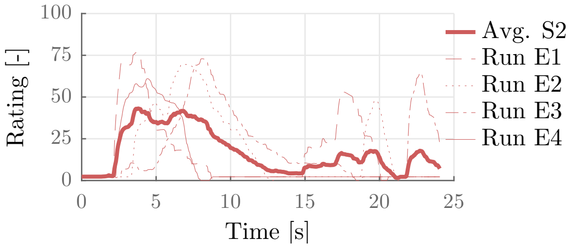
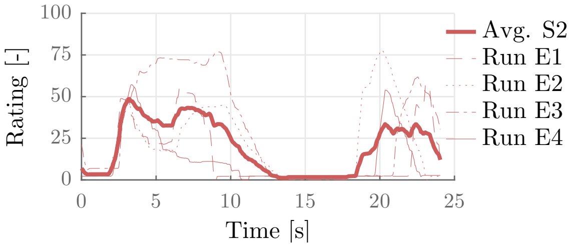


Figure F-60: Ratings by subject S2 for Condition C16. The mean rating is repeated in figure F-16.



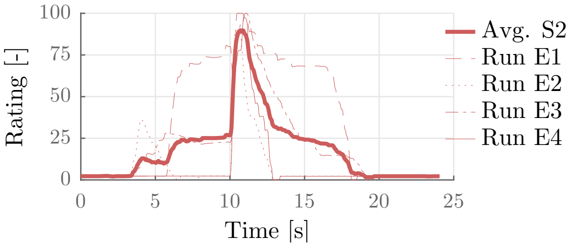
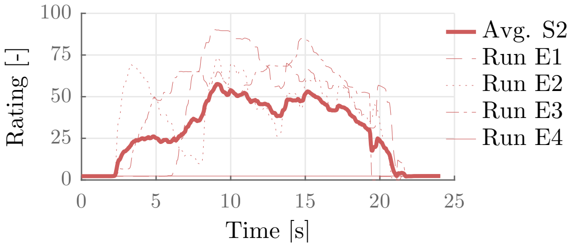


Figure F-61: Ratings by subject S2 for Condition C17. The mean rating is repeated in figure F-17.

Figure F-62: Ratings by subject S2 for Condition C18. The mean rating is repeated in figure F-18.



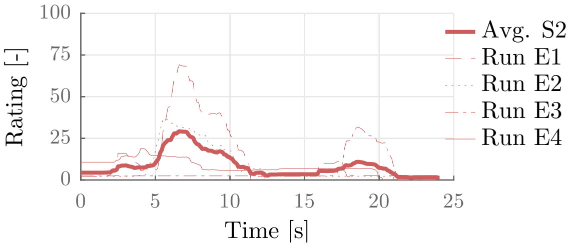
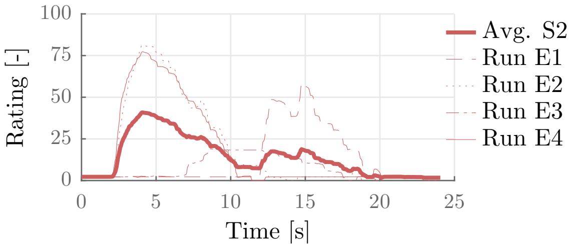


Figure F-63: Ratings by subject S2 for Condition C19. The mean rating is repeated in figure F-19.

Figure F-64: Ratings by subject S2 for Condition C20. The mean rating is repeated in figure F-20.



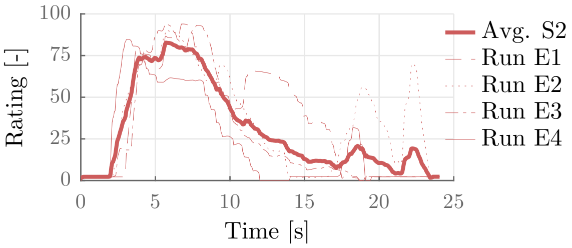
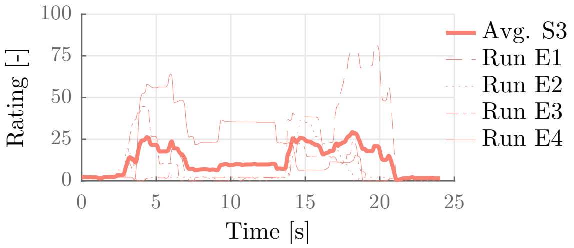


Figure F-65: Ratings by subject S2 for Condition C21. The mean rating is repeated in figure F-21.

Figure F-66: Ratings by subject S2 for Condition C22. The mean rating is repeated in figure F-22.



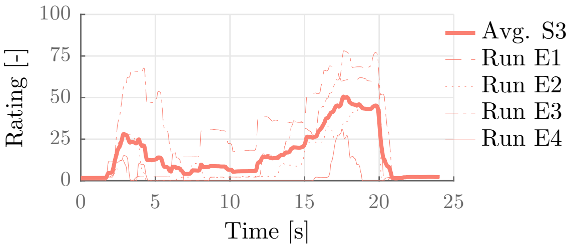
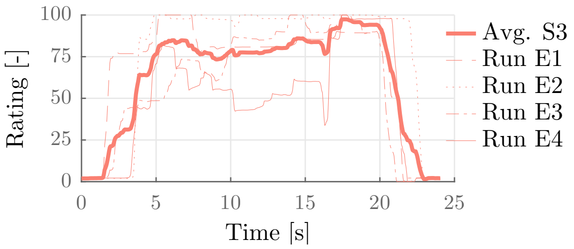


Figure F-67: Ratings by subject S3 for Condition C1. The mean rating is repeated in figure F-1.

Figure F-68: Ratings by subject S3 for Condition C2. The mean rating is repeated in figure F-2.



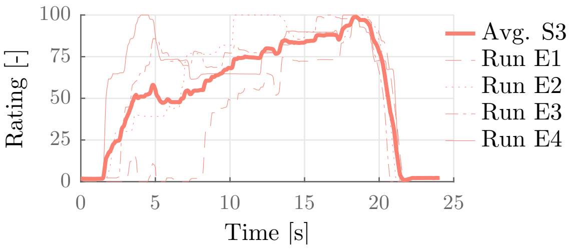


Figure F-69: Ratings by subject S3 for Condition C3. The mean rating is repeated in figure F-3.

Figure F-70: Ratings by subject S3 for Condition C4. The mean rating is repeated in figure F-4.

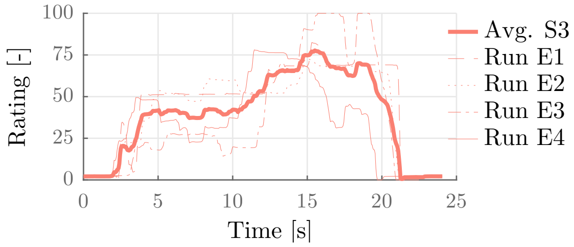


Figure F-71: Ratings by subject S3 for Condition C5. The mean rating is repeated in figure F-5.

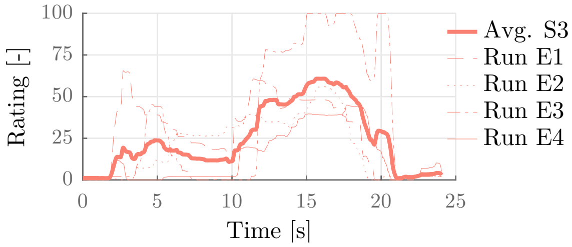


Figure F-72: Ratings by subject S3 for Condition C6. The mean rating is repeated in figure F-6.

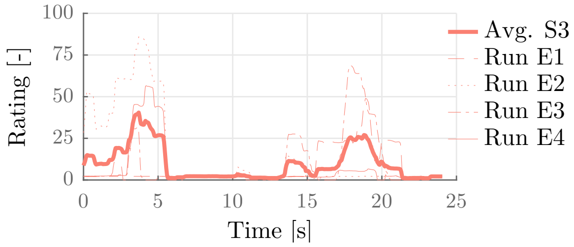


Figure F-73: Ratings by subject S3 for Condition C7. The mean rating is repeated in figure F-7.

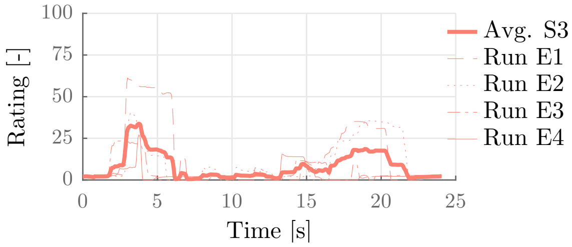


Figure F-74: Ratings by subject S3 for Condition C8. The mean rating is repeated in figure F-8.

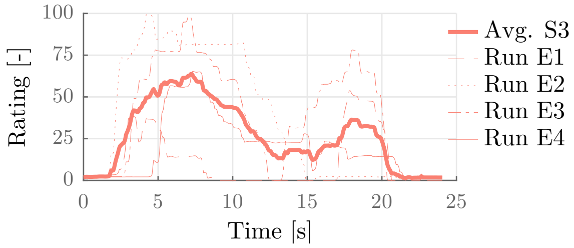


Figure F-75: Ratings by subject S3 for Condition C9. The mean rating is repeated in figure F-9.

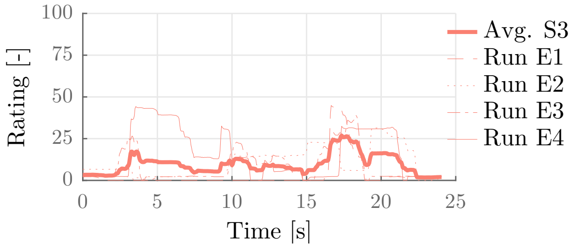


Figure F-76: Ratings by subject S3 for Condition C10. The mean rating is repeated in figure F-10.

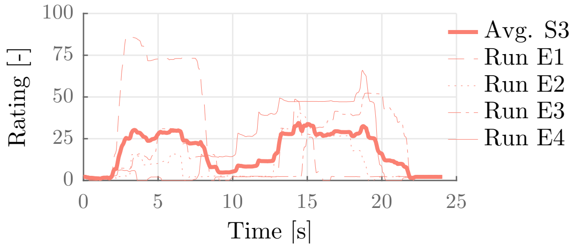


Figure F-77: Ratings by subject S3 for Condition C11. The mean rating is repeated in figure F-11.

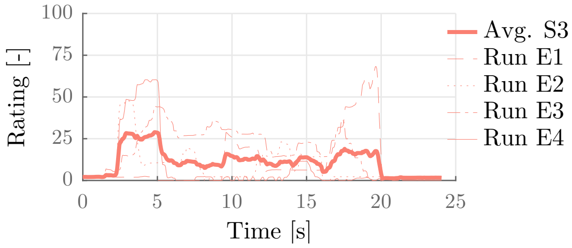


Figure F-78: Ratings by subject S3 for Condition C12. The mean rating is repeated in figure F-12.

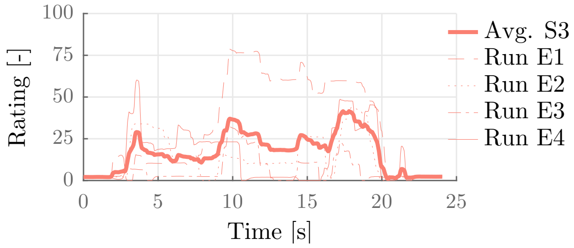


Figure F-79: Ratings by subject S3 for Condition C13. The mean rating is repeated in figure F-13.

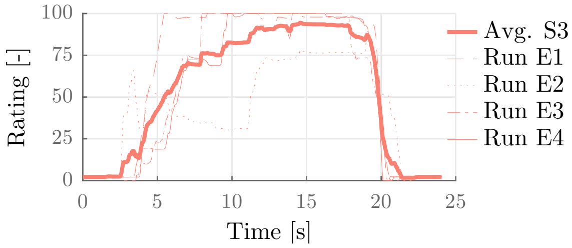


Figure F-80: Ratings by subject S3 for Condition C14. The mean rating is repeated in figure F-14.

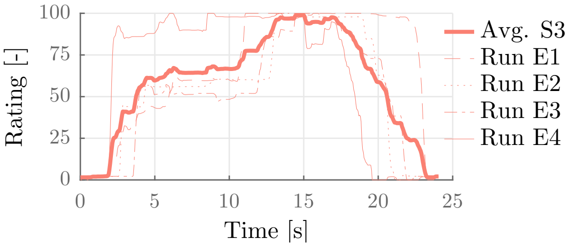


Figure F-81: Ratings by subject S3 for Condition C15. The mean rating is repeated in figure F-15.

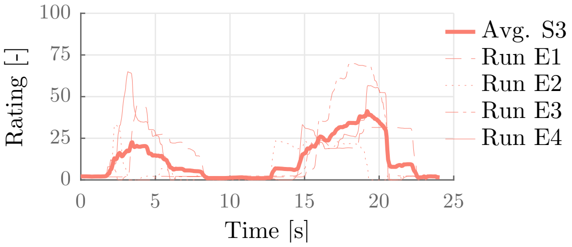


Figure F-82: Ratings by subject S3 for Condition C16. The mean rating is repeated in figure F-16.

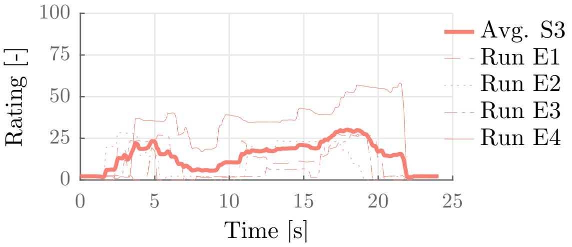


Figure F-83: Ratings by subject S3 for Condition C17. The mean rating is repeated in figure F-17.

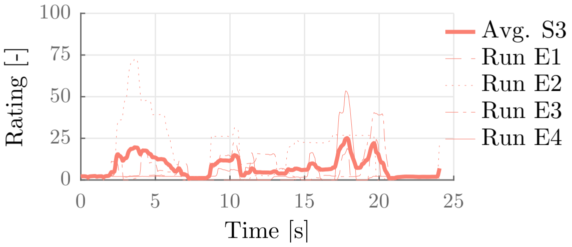


Figure F-84: Ratings by subject S3 for Condition C18. The mean rating is repeated in figure F-18.

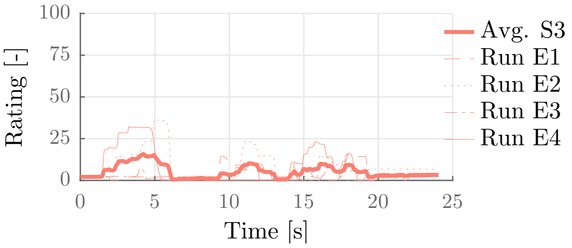


Figure F-85: Ratings by subject S3 for Condition C19. The mean rating is repeated in figure F-19.

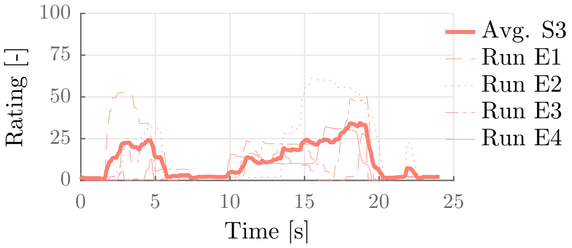


Figure F-86: Ratings by subject S3 for Condition C20. The mean rating is repeated in figure F-20.

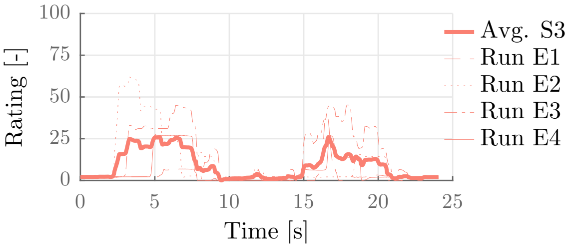


Figure F-87: Ratings by subject S3 for Condition C21. The mean rating is repeated in figure F-21.

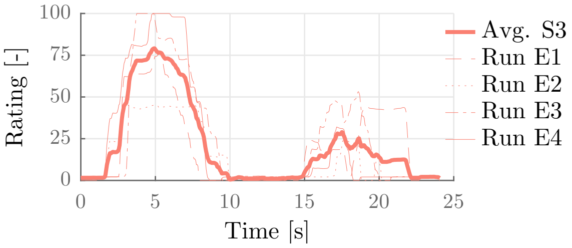


Figure F-88: Ratings by subject S3 for Condition C22. The mean rating is repeated in figure F-22.

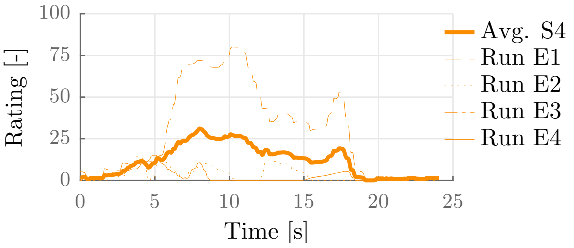


Figure F-89: Ratings by subject S4 for Condition C1. The mean rating is repeated in figure F-1.

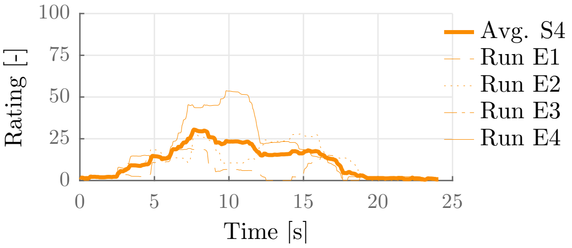


Figure F-90: Ratings by subject S4 for Condition C2. The mean rating is repeated in figure F-2.

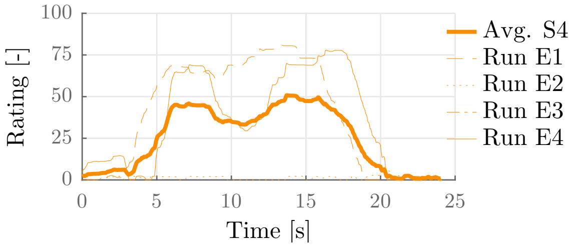


Figure F-91: Ratings by subject S4 for Condition C3. The mean rating is repeated in figure F-3.

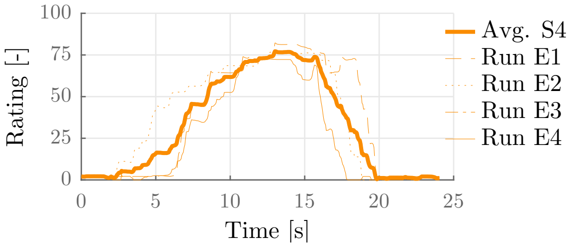


Figure F-92: Ratings by subject S4 for Condition C4. The mean rating is repeated in figure F-4.

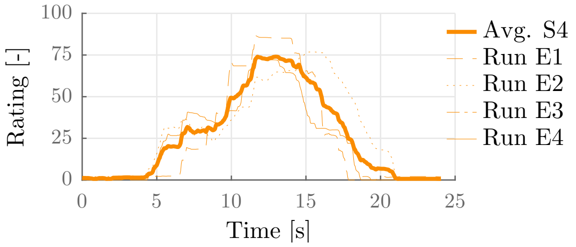


Figure F-93: Ratings by subject S4 for Condition C5. The mean rating is repeated in figure F-5.

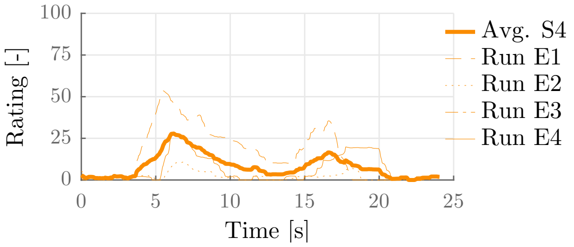


Figure F-94: Ratings by subject S4 for Condition C6. The mean rating is repeated in figure F-6.

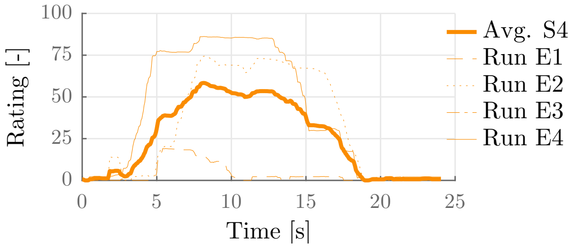


Figure F-95: Ratings by subject S4 for Condition C7. The mean rating is repeated in figure F-7.

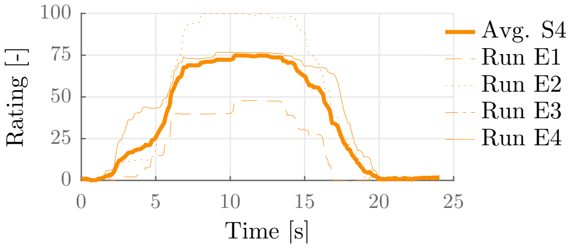


Figure F-96: Ratings by subject S4 for Condition C8. The mean rating is repeated in figure F-8.

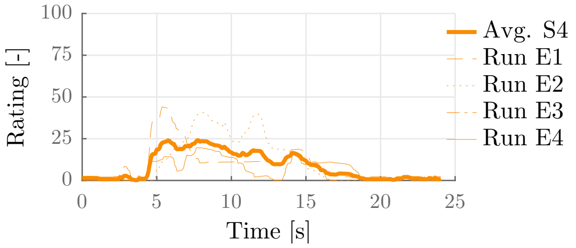


Figure F-97: Ratings by subject S4 for Condition C9. The mean rating is repeated in figure F-9.

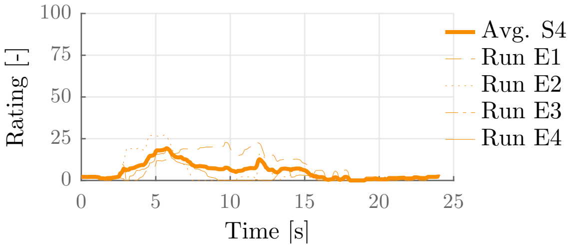


Figure F-98: Ratings by subject S4 for Condition C10. The mean rating is repeated in figure F-10.

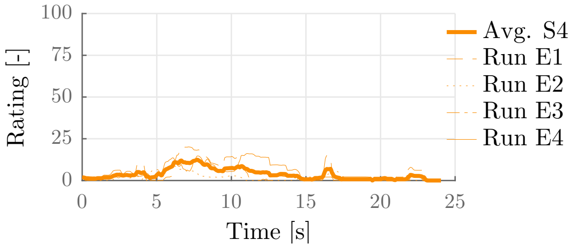


Figure F-99: Ratings by subject S4 for Condition C11. The mean rating is repeated in figure F-11.

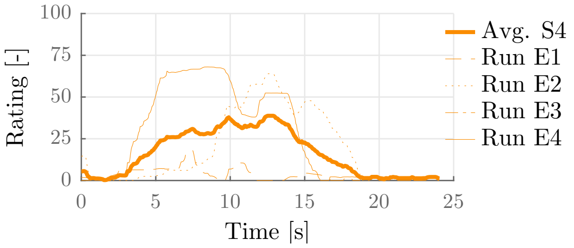
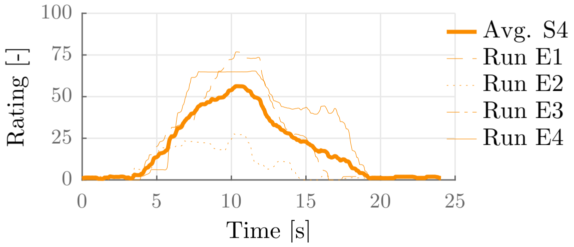


Figure F-100: Ratings by subject S4 for Condition C12. The mean rating is repeated in figure F-12.



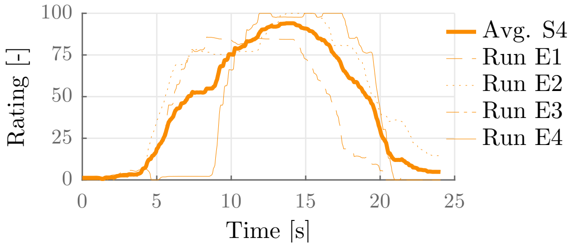
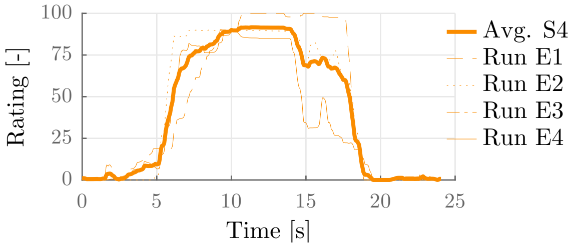


Figure F-101: Ratings by subject S4 for Condition C13. The mean rating is repeated in figure F-13.

Figure F-102: Ratings by subject S4 for Condition C14. The mean rating is repeated in figure F-14.



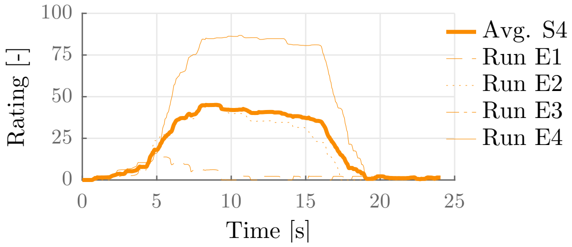
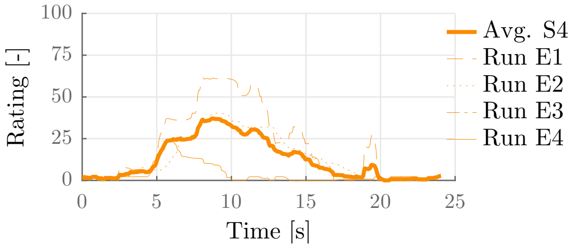


Figure F-103: Ratings by subject S4 for Condition C15. The mean rating is repeated in figure F-15.

Figure F-104: Ratings by subject S4 for Condition C16. The mean rating is repeated in figure F-16.



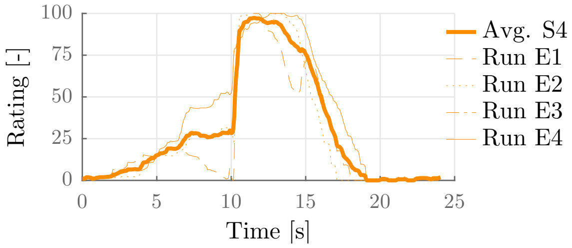
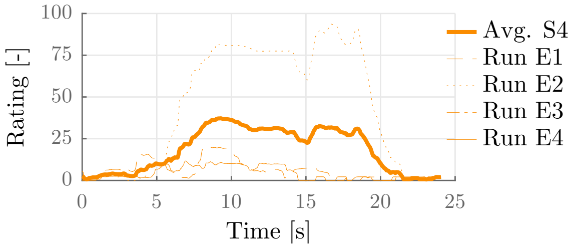


Figure F-105: Ratings by subject S4 for Condition C17. The mean rating is repeated in figure F-17.

Figure F-106: Ratings by subject S4 for Condition C18. The mean rating is repeated in figure F-18.



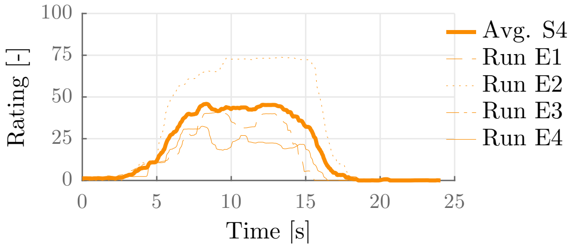
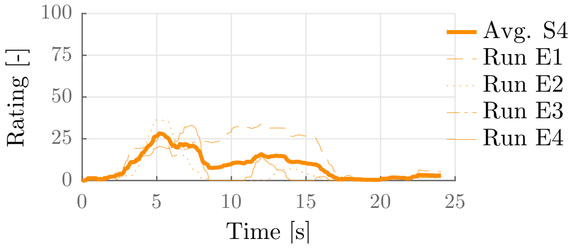


Figure F-107: Ratings by subject S4 for Condition C19. The mean rating is repeated in figure F-19.

Figure F-108: Ratings by subject S4 for Condition C20. The mean rating is repeated in figure F-20.



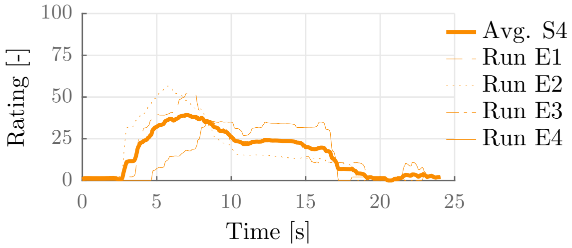


Figure F-109: Ratings by subject S4 for Condition C21. The mean rating is repeated in figure F-21.

Figure F-110: Ratings by subject S4 for Condition C22. The mean rating is repeated in figure F-22.

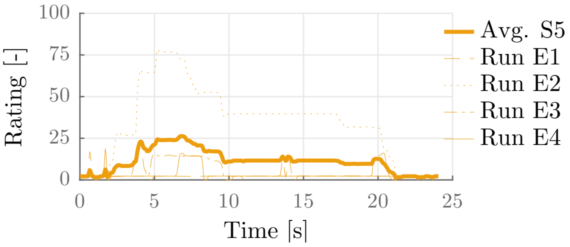


Figure F-111: Ratings by subject S5 for Condition C1. The mean rating is repeated in figure F-1.

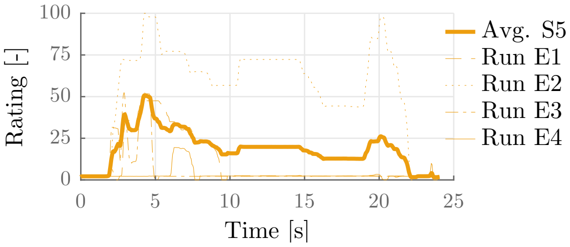


Figure F-112: Ratings by subject S5 for Condition C2. The mean rating is repeated in figure F-2.

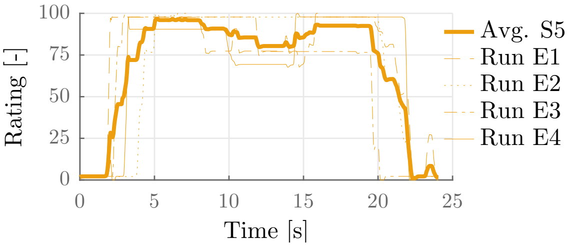


Figure F-113: Ratings by subject S5 for Condition C3. The mean rating is repeated in figure F-3.

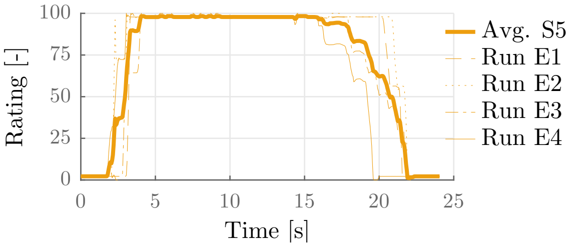


Figure F-114: Ratings by subject S5 for Condition C4. The mean rating is repeated in figure F-4.

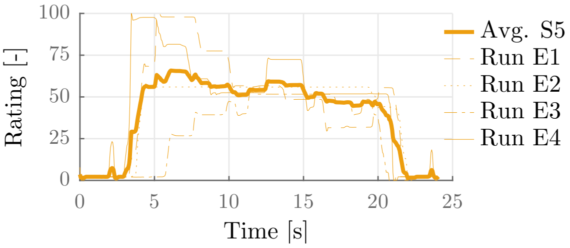


Figure F-115: Ratings by subject S5 for Condition C5. The mean rating is repeated in figure F-5.

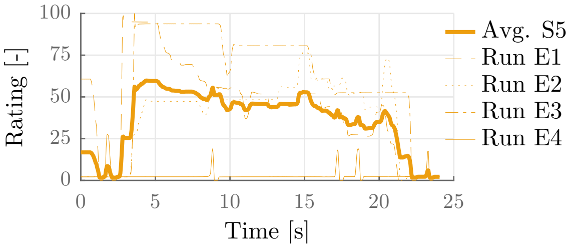


Figure F-116: Ratings by subject S5 for Condition C6. The mean rating is repeated in figure F-6.

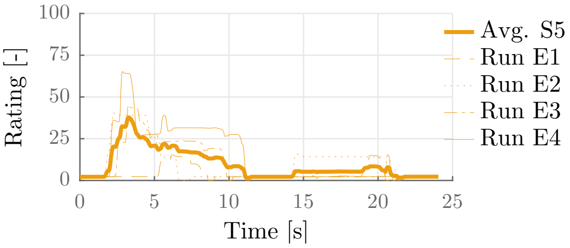


Figure F-117: Ratings by subject S5 for Condition C7. The mean rating is repeated in figure F-7.

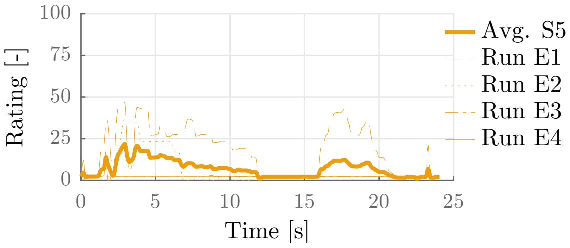


Figure F-118: Ratings by subject S5 for Condition C8. The mean rating is repeated in figure F-8.

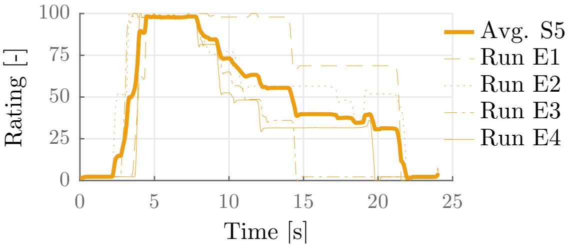


Figure F-119: Ratings by subject S5 for Condition C9. The mean rating is repeated in figure F-9.

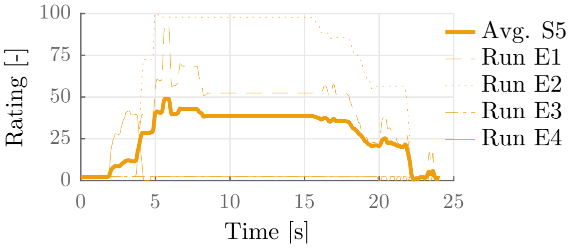
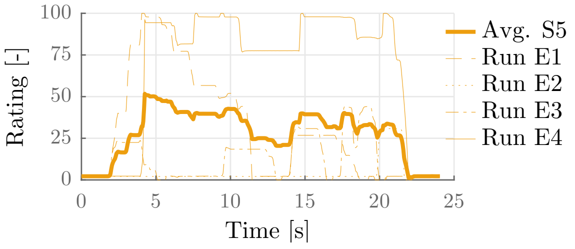


Figure F-120: Ratings by subject S5 for Condition C10. The mean rating is repeated in figure F-10.



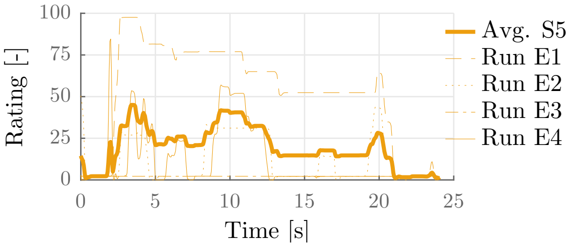
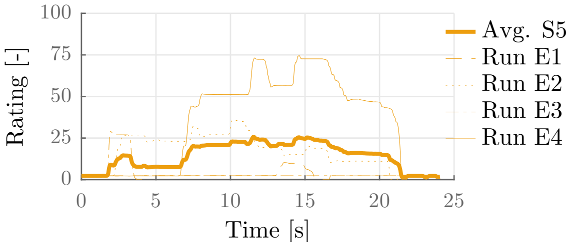


Figure F-121: Ratings by subject S5 for Condition C11. The mean rating is repeated in figure F-11.

Figure F-122: Ratings by subject S5 for Condition C12. The mean rating is repeated in figure F-12.



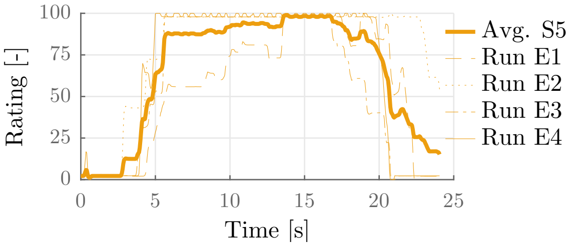
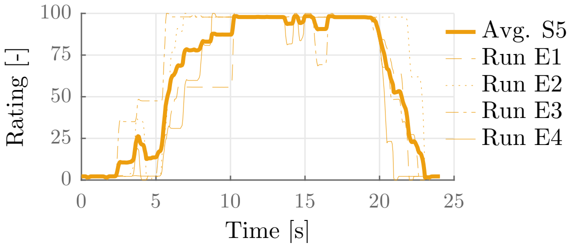


Figure F-123: Ratings by subject S5 for Condition C13. The mean rating is repeated in figure F-13.

Figure F-124: Ratings by subject S5 for Condition C14. The mean rating is repeated in figure F-14.



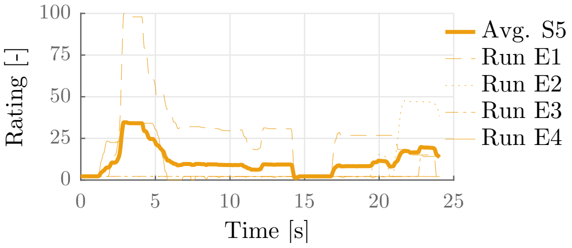
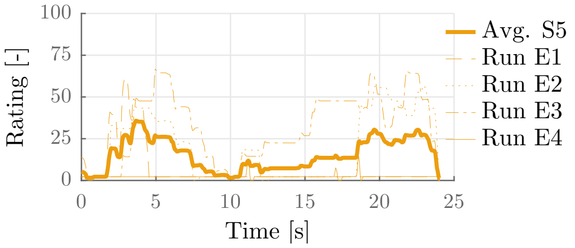


Figure F-125: Ratings by subject S5 for Condition C15. The mean rating is repeated in figure F-15.

Figure F-126: Ratings by subject S5 for Condition C16. The mean rating is repeated in figure F-16.



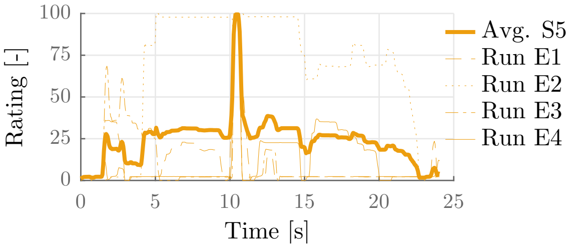
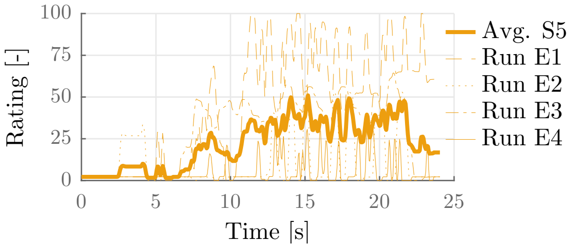


Figure F-127: Ratings by subject S5 for Condition C17. The mean rating is repeated in figure F-17.

Figure F-128: Ratings by subject S5 for Condition C18. The mean rating is repeated in figure F-18.



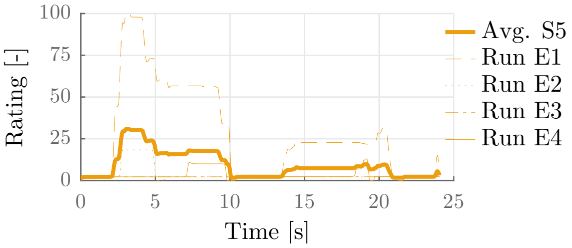


Figure F-129: Ratings by subject S5 for Condition C19. The mean rating is repeated in figure F-19.

Figure F-130: Ratings by subject S5 for Condition C20. The mean rating is repeated in figure F-20.

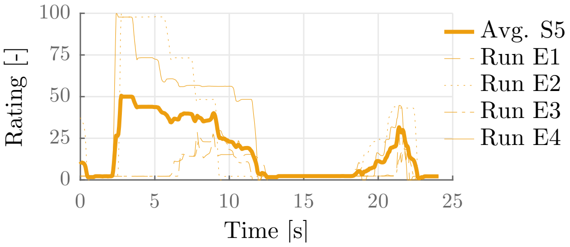


Figure F-131: Ratings by subject S5 for Condition C21. The mean rating is repeated in figure F-21.

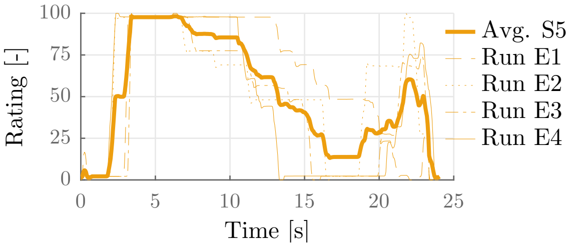


Figure F-132: Ratings by subject S5 for Condition C22. The mean rating is repeated in figure F-22.

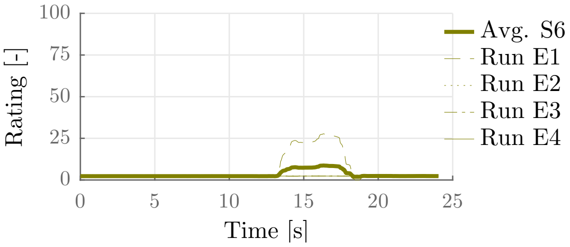


Figure F-133: Ratings by subject S6 for Condition C1. The mean rating is repeated in figure F-1.

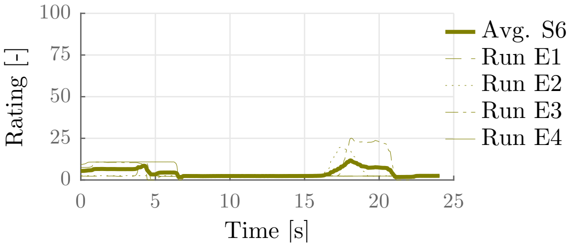


Figure F-134: Ratings by subject S6 for Condition C2. The mean rating is repeated in figure F-2.

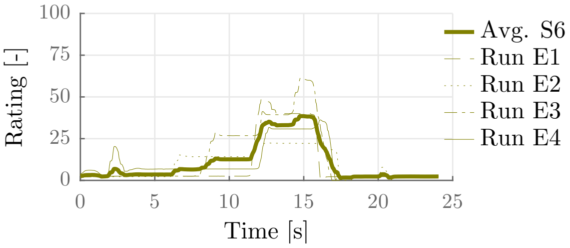


Figure F-135: Ratings by subject S6 for Condition C3. The mean rating is repeated in figure F-3.

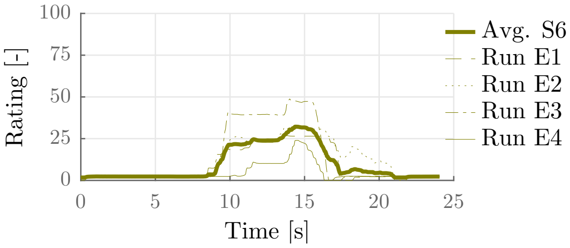


Figure F-136: Ratings by subject S6 for Condition C4. The mean rating is repeated in figure F-4.

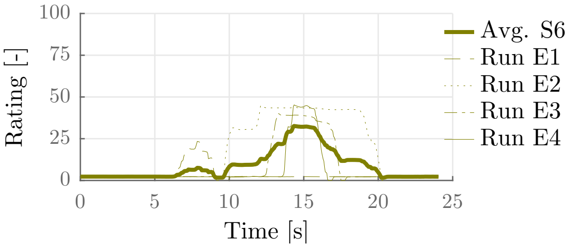


Figure F-137: Ratings by subject S6 for Condition C5. The mean rating is repeated in figure F-5.

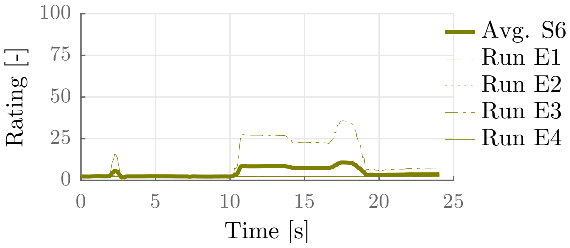


Figure F-138: Ratings by subject S6 for Condition C6. The mean rating is repeated in figure F-6.

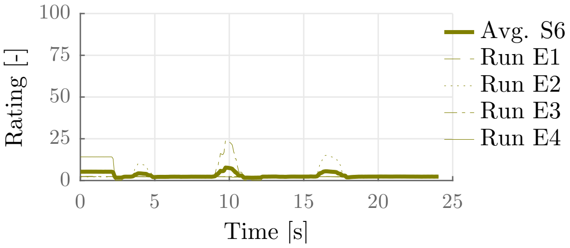


Figure F-139: Ratings by subject S6 for Condition C7. The mean rating is repeated in figure F-7.

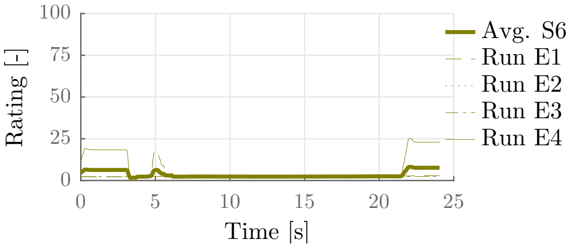


Figure F-140: Ratings by subject S6 for Condition C8. The mean rating is repeated in figure F-8.

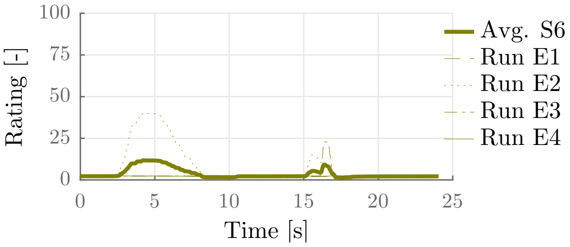


Figure F-141: Ratings by subject S6 for Condition C9. The mean rating is repeated in figure F-9.

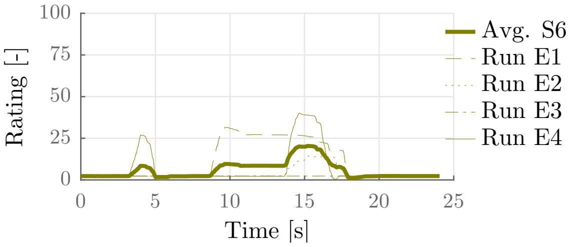


Figure F-142: Ratings by subject S6 for Condition C10. The mean rating is repeated in figure F-10.

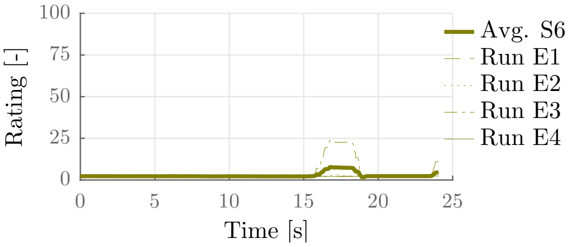


Figure F-143: Ratings by subject S6 for Condition C11. The mean rating is repeated in figure F-11.

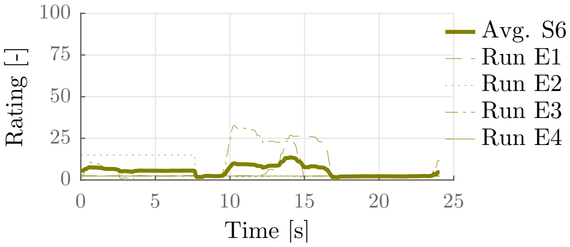


Figure F-144: Ratings by subject S6 for Condition C12. The mean rating is repeated in figure F-12.

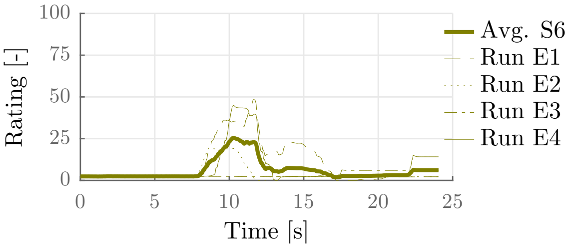


Figure F-145: Ratings by subject S6 for Condition C13. The mean rating is repeated in figure F-13.

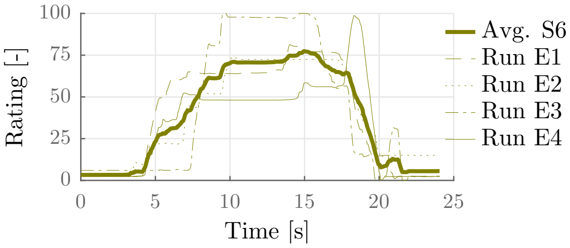


Figure F-146: Ratings by subject S6 for Condition C14. The mean rating is repeated in figure F-14.

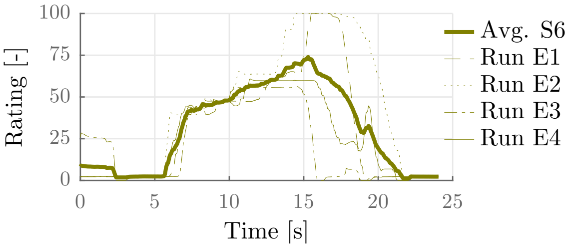


Figure F-147: Ratings by subject S6 for Condition C15. The mean rating is repeated in figure F-15.

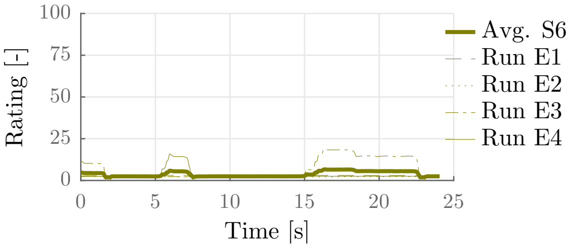


Figure F-148: Ratings by subject S6 for Condition C16. The mean rating is repeated in figure F-16.

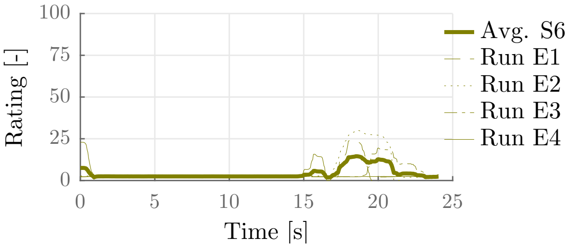


Figure F-149: Ratings by subject S6 for Condition C17. The mean rating is repeated in figure F-17.

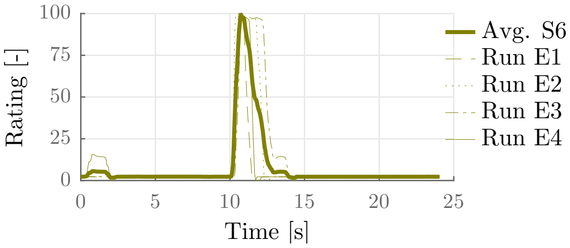


Figure F-150: Ratings by subject S6 for Condition C18. The mean rating is repeated in figure F-18.

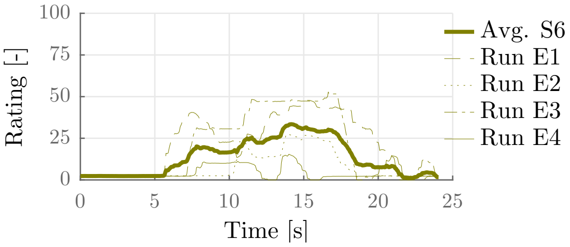


Figure F-151: Ratings by subject S6 for Condition C19. The mean rating is repeated in figure F-19.

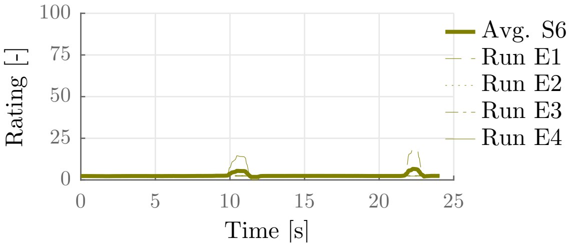


Figure F-152: Ratings by subject S6 for Condition C20. The mean rating is repeated in figure F-20.

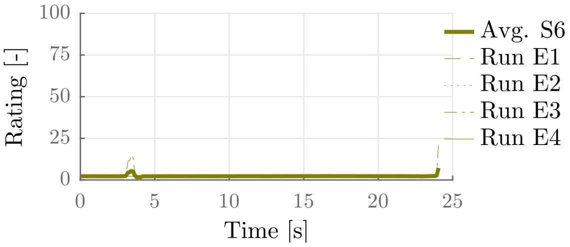


Figure F-153: Ratings by subject S6 for Condition C21. The mean rating is repeated in figure F-21.

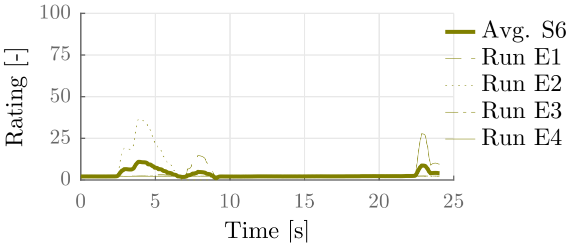


Figure F-154: Ratings by subject S6 for Condition C22. The mean rating is repeated in figure F-22.

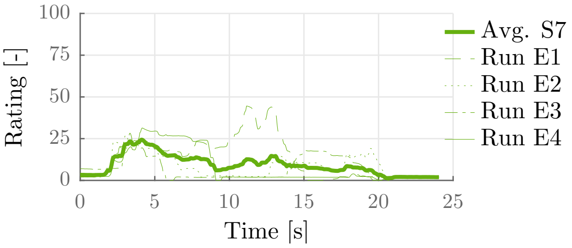


Figure F-155: Ratings by subject S7 for Condition C1. The mean rating is repeated in figure F-1.

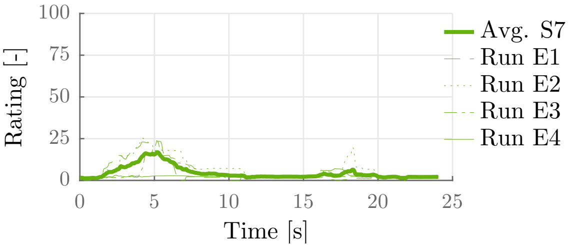


Figure F-156: Ratings by subject S7 for Condition C2. The mean rating is repeated in figure F-2.

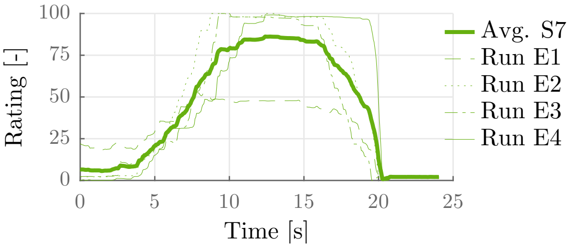


Figure F-157: Ratings by subject S7 for Condition C3. The mean rating is repeated in figure F-3.

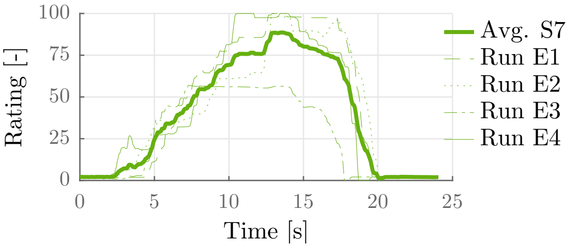


Figure F-158: Ratings by subject S7 for Condition C4. The mean rating is repeated in figure F-4.

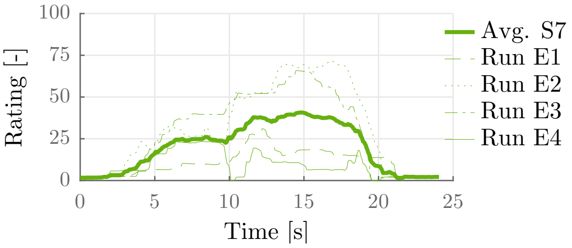


Figure F-159: Ratings by subject S7 for Condition C5. The mean rating is repeated in figure F-5.

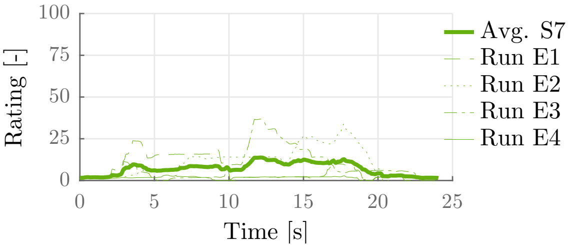


Figure F-160: Ratings by subject S7 for Condition C6. The mean rating is repeated in figure F-6.

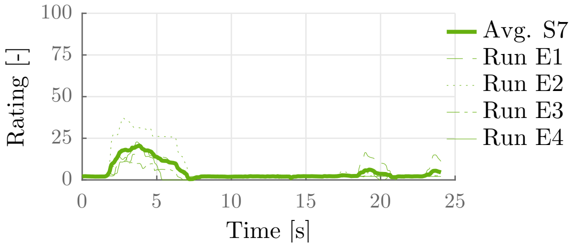


Figure F-161: Ratings by subject S7 for Condition C7. The mean rating is repeated in figure F-7.

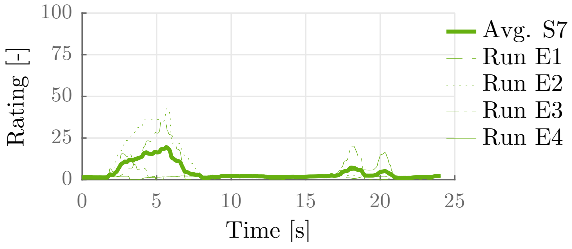


Figure F-162: Ratings by subject S7 for Condition C8. The mean rating is repeated in figure F-8.

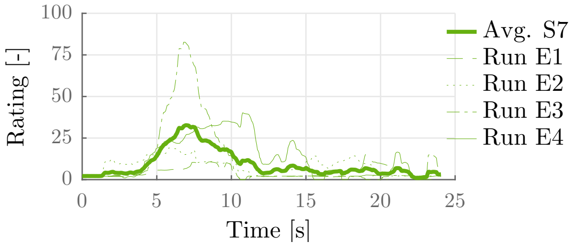


Figure F-163: Ratings by subject S7 for Condition C9. The mean rating is repeated in figure F-9.

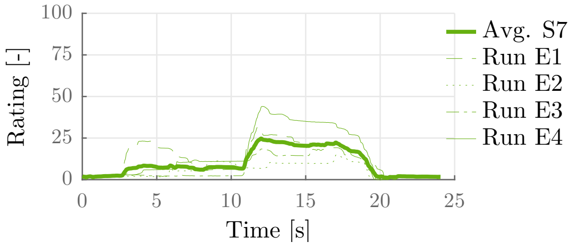


Figure F-164: Ratings by subject S7 for Condition C10. The mean rating is repeated in figure F-10.

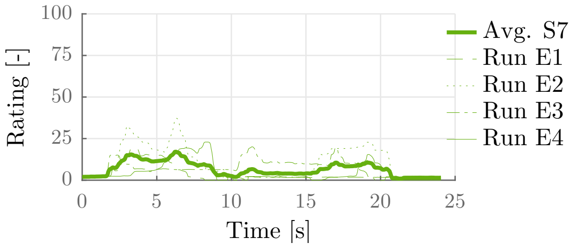


Figure F-165: Ratings by subject S7 for Condition C11. The mean rating is repeated in figure F-11.

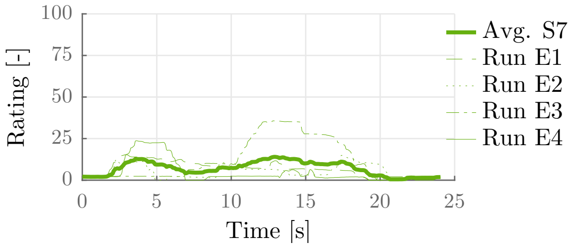


Figure F-166: Ratings by subject S7 for Condition C12. The mean rating is repeated in figure F-12.

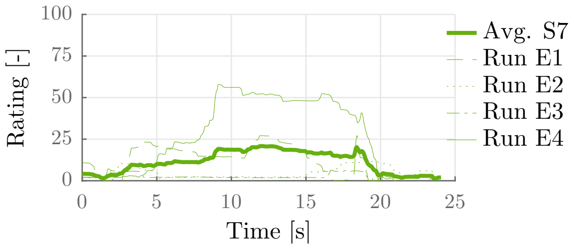


Figure F-167: Ratings by subject S7 for Condition C13. The mean rating is repeated in figure F-13.

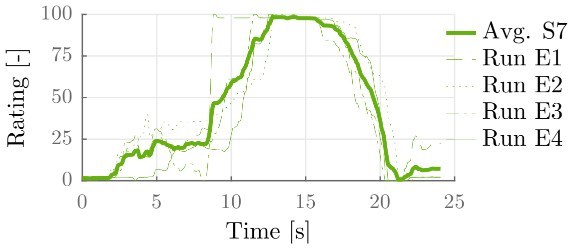


Figure F-168: Ratings by subject S7 for Condition C14. The mean rating is repeated in figure F-14.

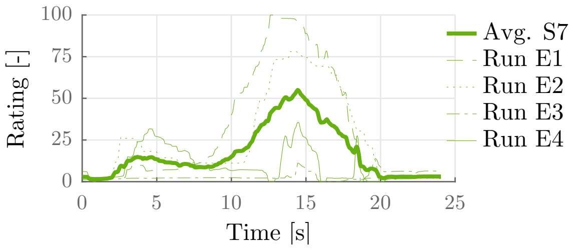


Figure F-169: Ratings by subject S7 for Condition C15. The mean rating is repeated in figure F-15.

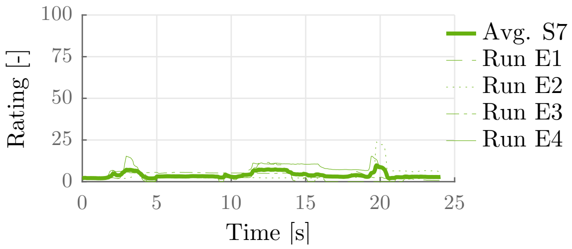


Figure F-170: Ratings by subject S7 for Condition C16. The mean rating is repeated in figure F-16.

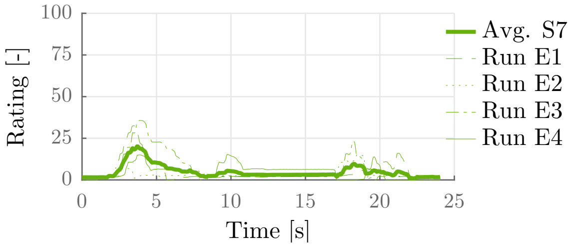


Figure F-171: Ratings by subject S7 for Condition C17. The mean rating is repeated in figure F-17.

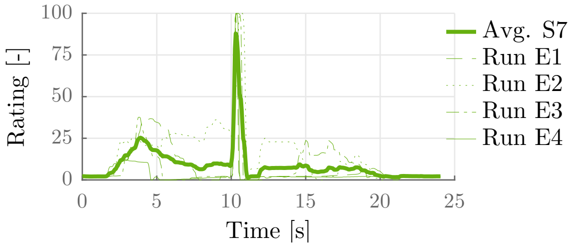


Figure F-172: Ratings by subject S7 for Condition C18. The mean rating is repeated in figure F-18.

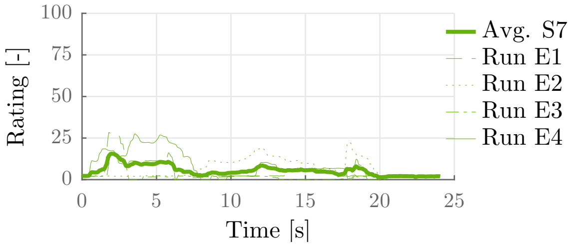


Figure F-173: Ratings by subject S7 for Condition C19. The mean rating is repeated in figure F-19.

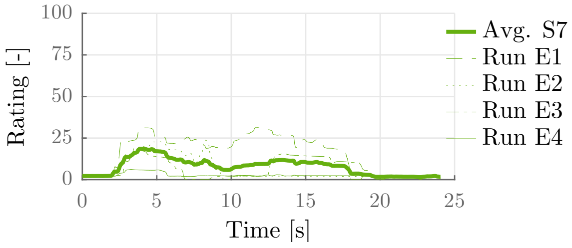


Figure F-174: Ratings by subject S7 for Condition C20. The mean rating is repeated in figure F-20.

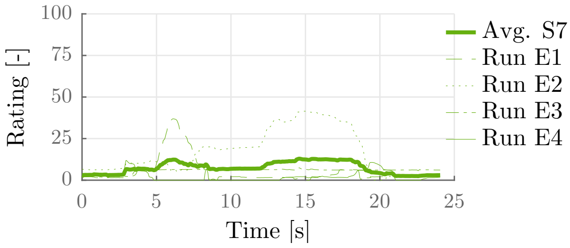


Figure F-175: Ratings by subject S7 for Condition C21. The mean rating is repeated in figure F-21.

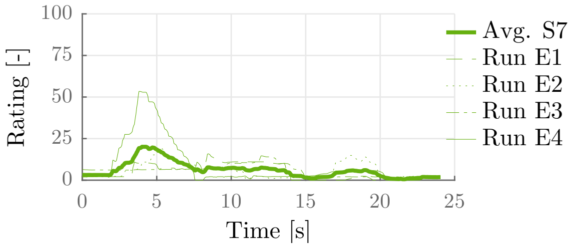


Figure F-176: Ratings by subject S7 for Condition C22. The mean rating is repeated in figure F-22.

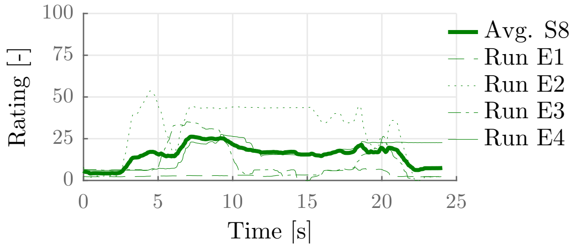


Figure F-177: Ratings by subject S8 for Condition C1. The mean rating is repeated in figure F-1.

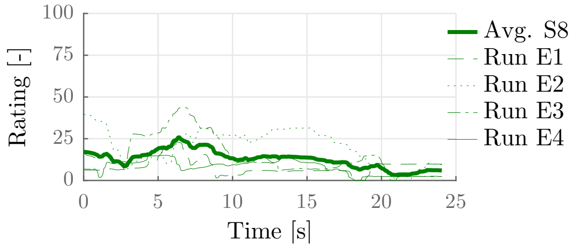


Figure F-178: Ratings by subject S8 for Condition C2. The mean rating is repeated in figure F-2.

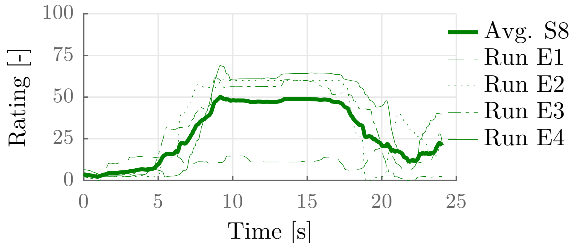


Figure F-179: Ratings by subject S8 for Condition C3. The mean rating is repeated in figure F-3.

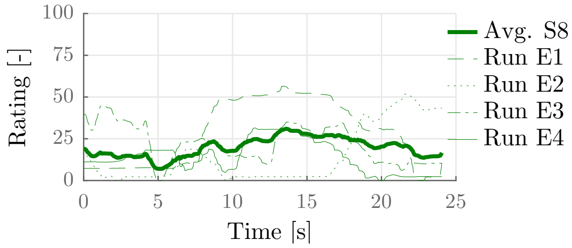


Figure F-180: Ratings by subject S8 for Condition C4. The mean rating is repeated in figure F-4.

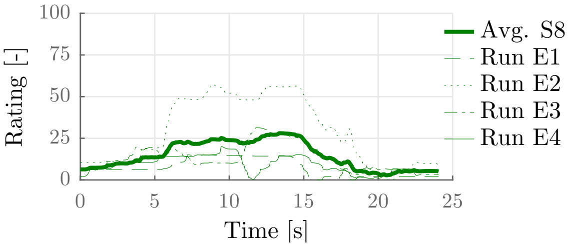


Figure F-181: Ratings by subject S8 for Condition C5. The mean rating is repeated in figure F-5.

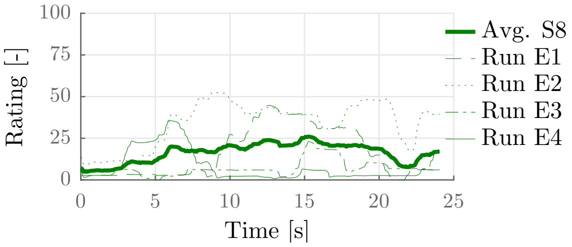


Figure F-182: Ratings by subject S8 for Condition C6. The mean rating is repeated in figure F-6.

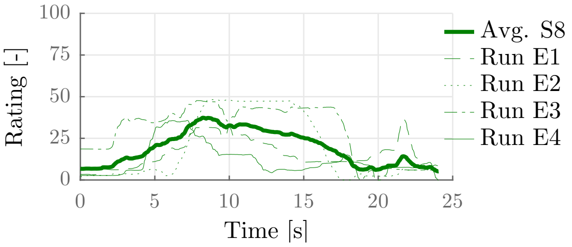


Figure F-183: Ratings by subject S8 for Condition C7. The mean rating is repeated in figure F-7.

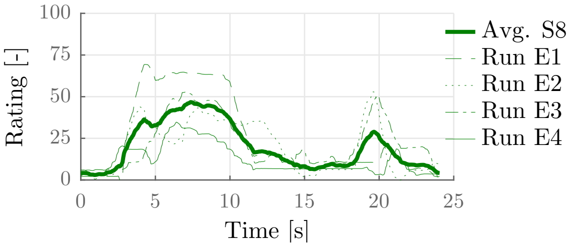


Figure F-184: Ratings by subject S8 for Condition C8. The mean rating is repeated in figure F-8.

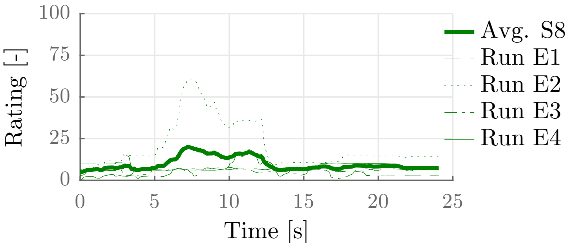


Figure F-185: Ratings by subject S8 for Condition C9. The mean rating is repeated in figure F-9.

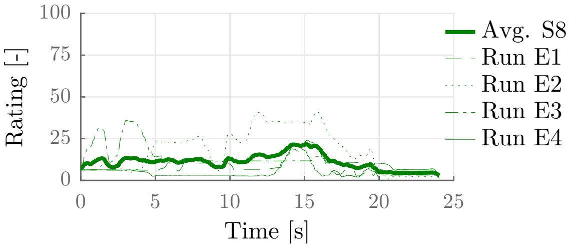


Figure F-186: Ratings by subject S8 for Condition C10. The mean rating is repeated in figure F-10.

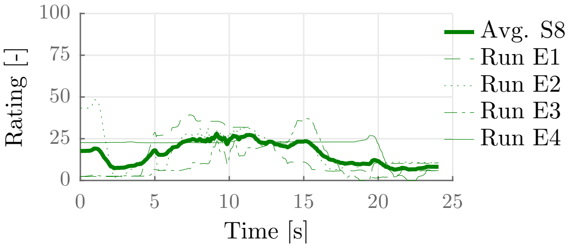


Figure F-187: Ratings by subject S8 for Condition C11. The mean rating is repeated in figure F-11.

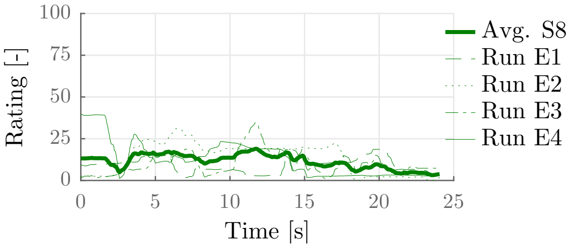


Figure F-188: Ratings by subject S8 for Condition C12. The mean rating is repeated in figure F-12.

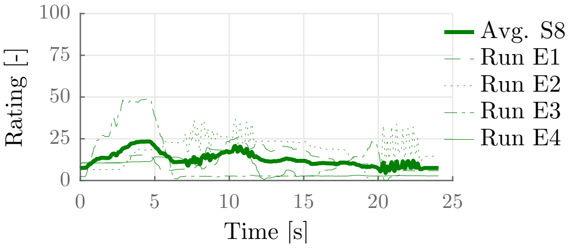


Figure F-189: Ratings by subject S8 for Condition C13. The mean rating is repeated in figure F-13.

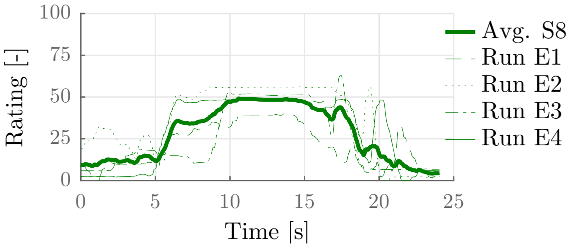


Figure F-190: Ratings by subject S8 for Condition C14. The mean rating is repeated in figure F-14.

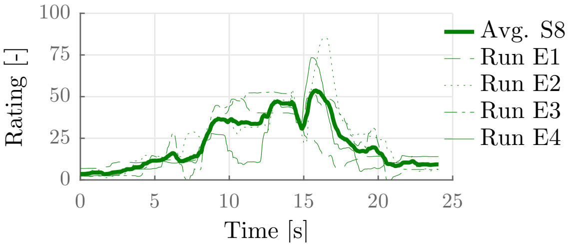


Figure F-191: Ratings by subject S8 for Condition C15. The mean rating is repeated in figure F-15.

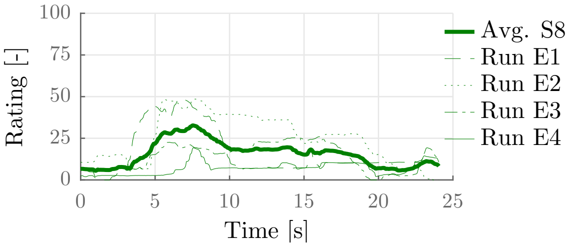


Figure F-192: Ratings by subject S8 for Condition C16. The mean rating is repeated in figure F-16.

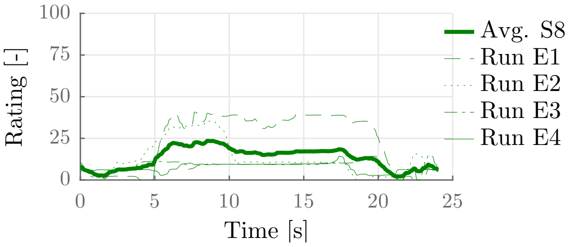


Figure F-193: Ratings by subject S8 for Condition C17. The mean rating is repeated in figure F-17.

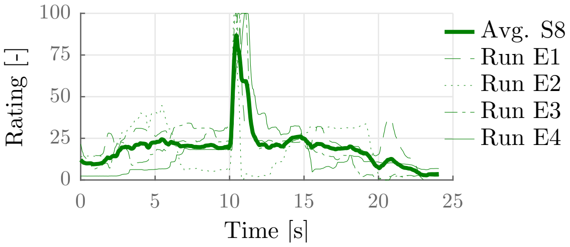


Figure F-194: Ratings by subject S8 for Condition C18. The mean rating is repeated in figure F-18.

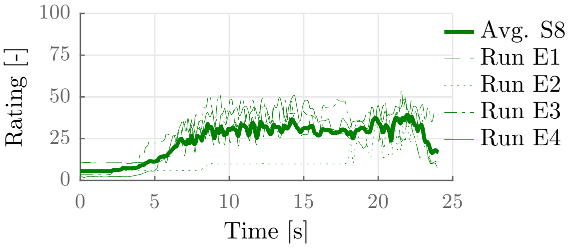


Figure F-195: Ratings by subject S8 for Condition C19. The mean rating is repeated in figure F-19.

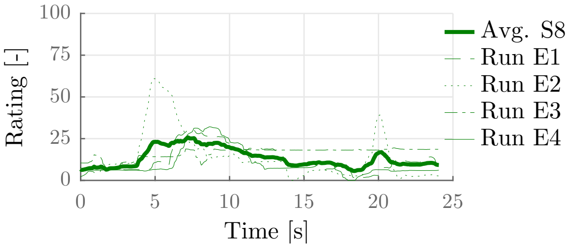


Figure F-196: Ratings by subject S8 for Condition C20. The mean rating is repeated in figure F-20.

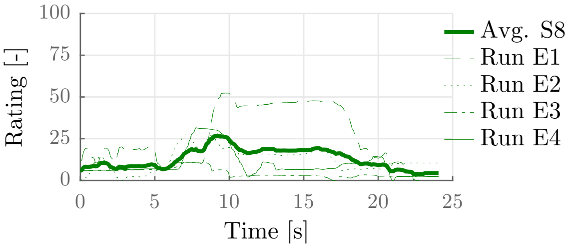


Figure F-197: Ratings by subject S8 for Condition C21. The mean rating is repeated in figure F-21.

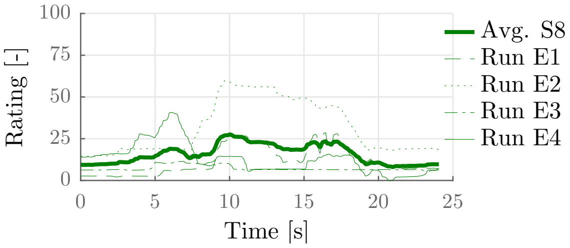


Figure F-198: Ratings by subject S8 for Condition C22. The mean rating is repeated in figure F-22.

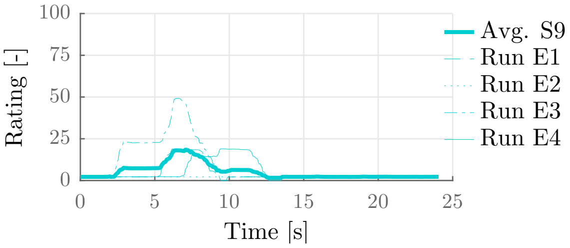


Figure F-199: Ratings by subject S9 for Condition C1. The mean rating is repeated in figure F-1.

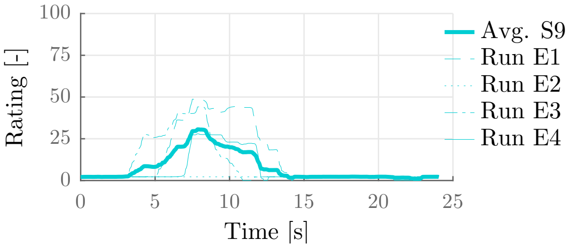


Figure F-200: Ratings by subject S9 for Condition C2. The mean rating is repeated in figure F-2.

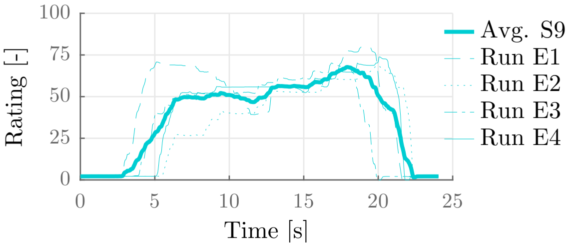


Figure F-201: Ratings by subject S9 for Condition C3. The mean rating is repeated in figure F-3.

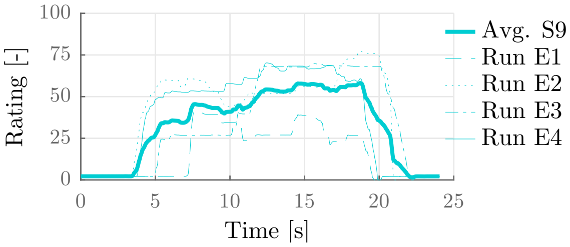


Figure F-202: Ratings by subject S9 for Condition C4. The mean rating is repeated in figure F-4.

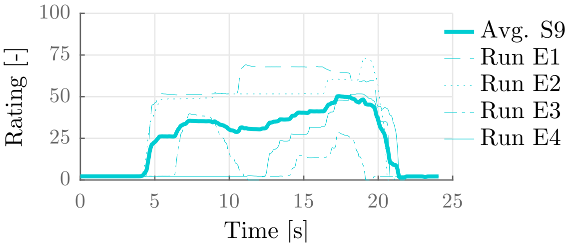


Figure F-203: Ratings by subject S9 for Condition C5. The mean rating is repeated in figure F-5.

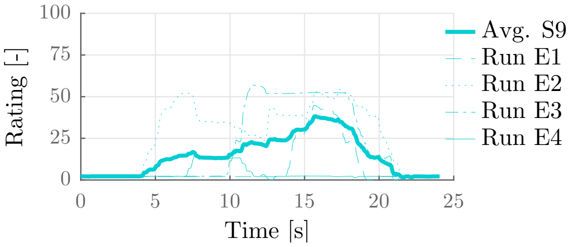


Figure F-204: Ratings by subject S9 for Condition C6. The mean rating is repeated in figure F-6.

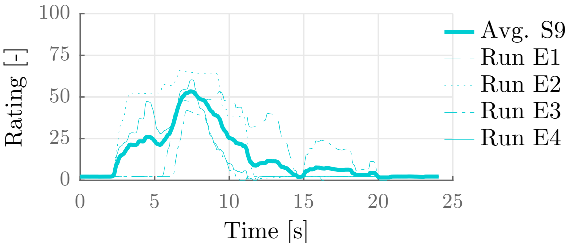


Figure F-205: Ratings by subject S9 for Condition C7. The mean rating is repeated in figure F-7.

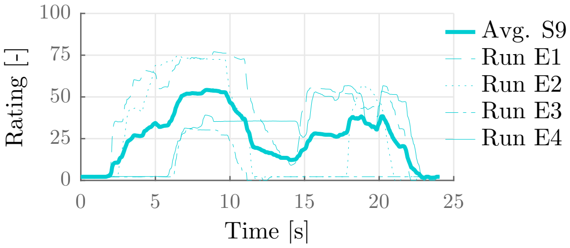


Figure F-206: Ratings by subject S9 for Condition C8. The mean rating is repeated in figure F-8.

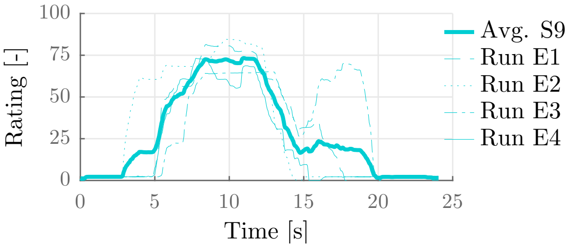


Figure F-207: Ratings by subject S9 for Condition C9. The mean rating is repeated in figure F-9.

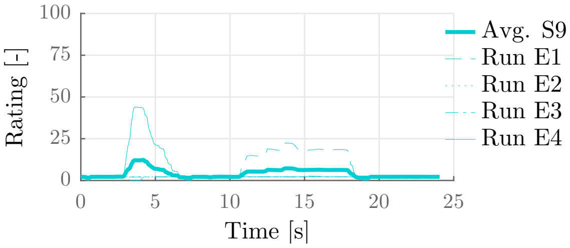


Figure F-208: Ratings by subject S9 for Condition C10. The mean rating is repeated in figure F-10.

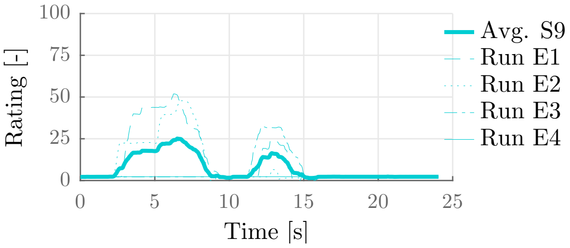


Figure F-209: Ratings by subject S9 for Condition C11. The mean rating is repeated in figure F-11.

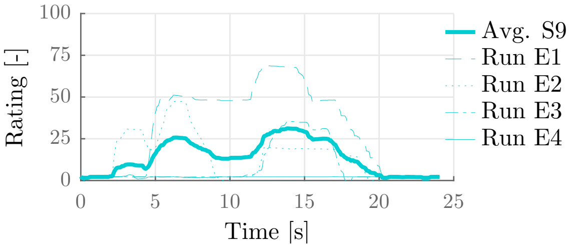


Figure F-210: Ratings by subject S9 for Condition C12. The mean rating is repeated in figure F-12.

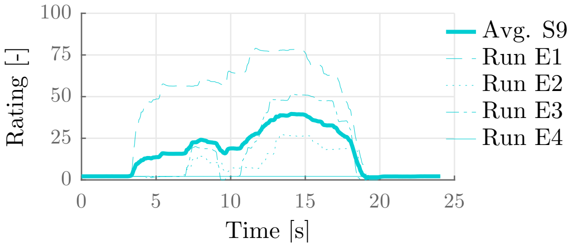


Figure F-211: Ratings by subject S9 for Condition C13. The mean rating is repeated in figure F-13.

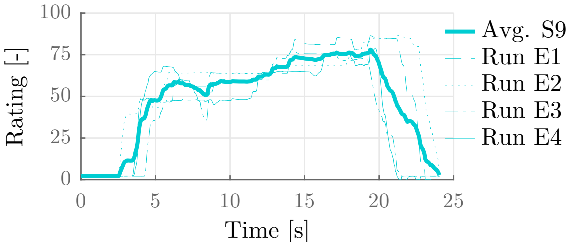


Figure F-212: Ratings by subject S9 for Condition C14. The mean rating is repeated in figure F-14.

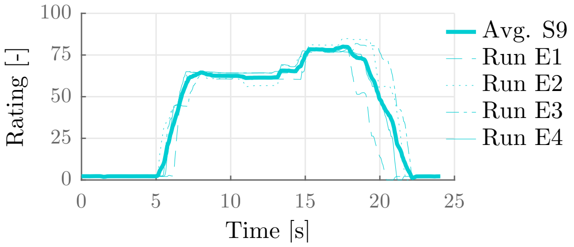


Figure F-213: Ratings by subject S9 for Condition C15. The mean rating is repeated in figure F-15.

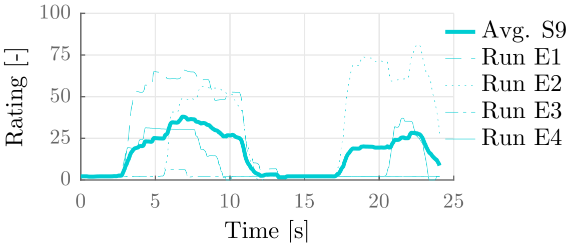


Figure F-214: Ratings by subject S9 for Condition C16. The mean rating is repeated in figure F-16.

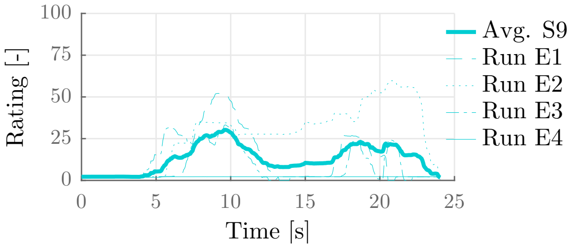


Figure F-215: Ratings by subject S9 for Condition C17. The mean rating is repeated in figure F-17.

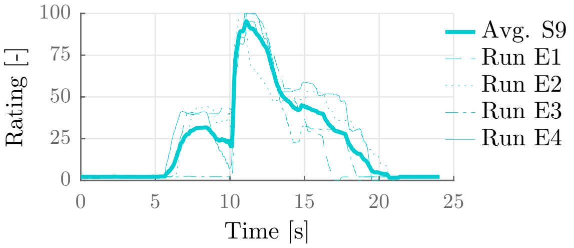


Figure F-216: Ratings by subject S9 for Condition C18. The mean rating is repeated in figure F-18.

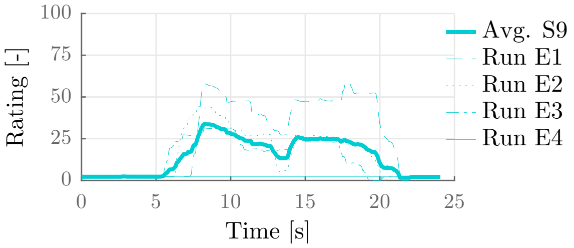


Figure F-217: Ratings by subject S9 for Condition C19. The mean rating is repeated in figure F-19.

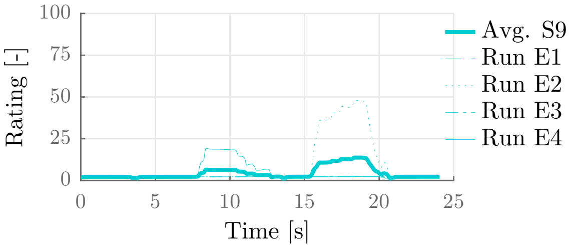


Figure F-218: Ratings by subject S9 for Condition C20. The mean rating is repeated in figure F-20.

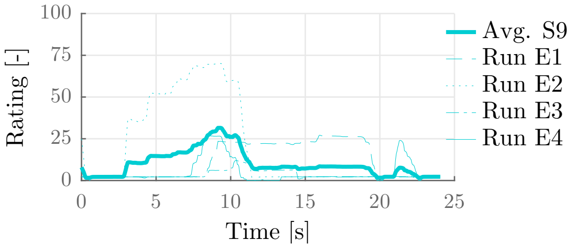


Figure F-219: Ratings by subject S9 for Condition C21. The mean rating is repeated in figure F-21.

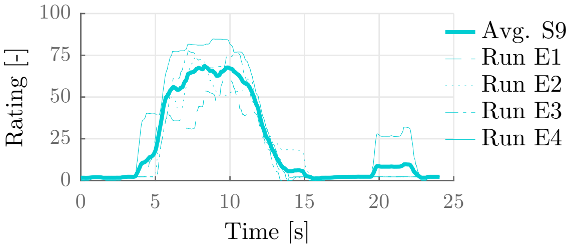


Figure F-220: Ratings by subject S9 for Condition C22. The mean rating is repeated in figure F-22.

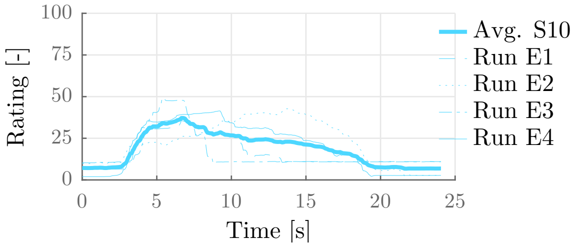


Figure F-221: Ratings by subject S10 for Condition C1. The mean rating is repeated in figure F-1.

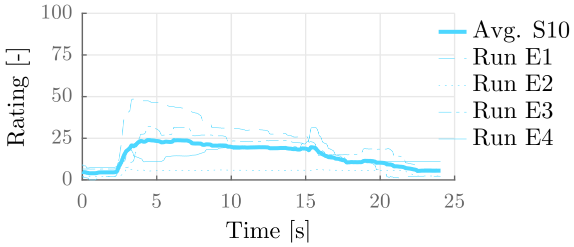


Figure F-222: Ratings by subject S10 for Condition C2. The mean rating is repeated in figure F-2.

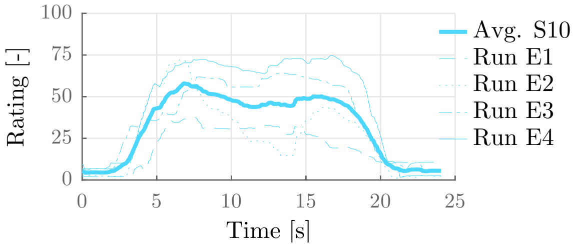


Figure F-223: Ratings by subject S10 for Condition C3. The mean rating is repeated in figure F-3.

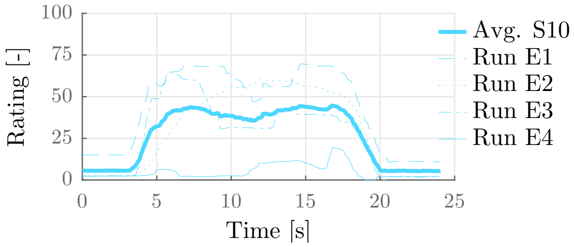


Figure F-224: Ratings by subject S10 for Condition C4. The mean rating is repeated in figure F-4.

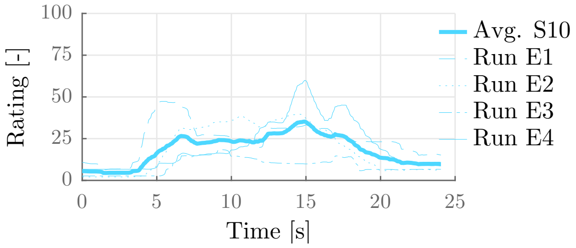


Figure F-225: Ratings by subject S10 for Condition C5. The mean rating is repeated in figure F-5.

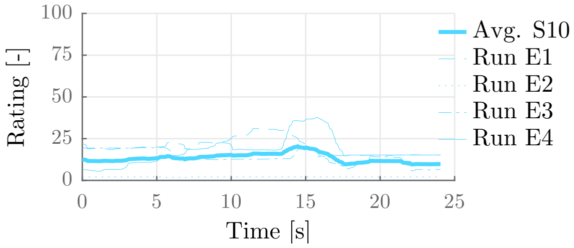


Figure F-226: Ratings by subject S10 for Condition C6. The mean rating is repeated in figure F-6.

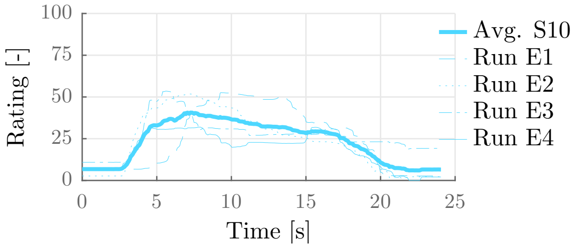


Figure F-227: Ratings by subject S10 for Condition C7. The mean rating is repeated in figure F-7.

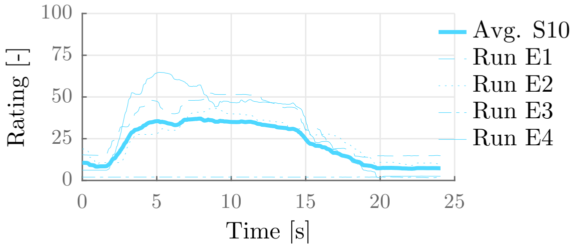


Figure F-228: Ratings by subject S10 for Condition C8. The mean rating is repeated in figure F-8.

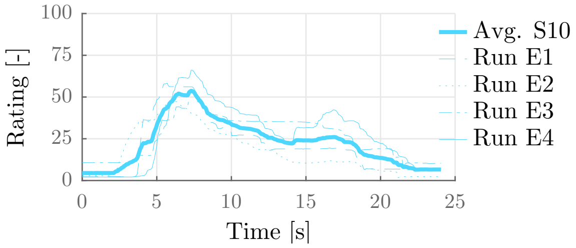


Figure F-229: Ratings by subject S10 for Condition C9. The mean rating is repeated in figure F-9.

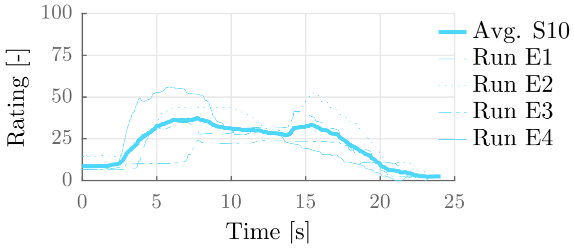


Figure F-230: Ratings by subject S10 for Condition C10. The mean rating is repeated in figure F-10.

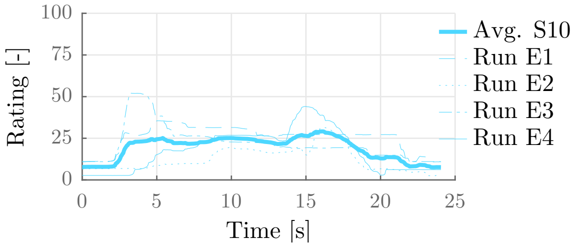


Figure F-231: Ratings by subject S10 for Condition C11. The mean rating is repeated in figure F-11.

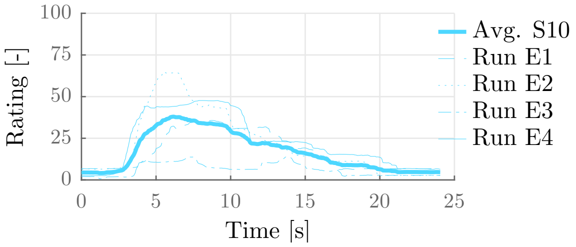


Figure F-232: Ratings by subject S10 for Condition C12. The mean rating is repeated in figure F-12.

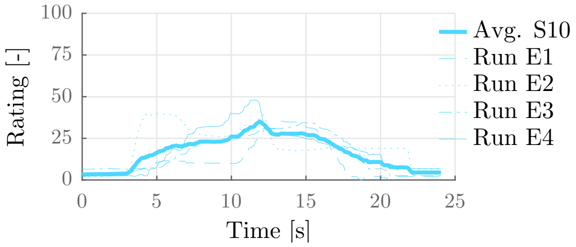


Figure F-233: Ratings by subject S10 for Condition C13. The mean rating is repeated in figure F-13.

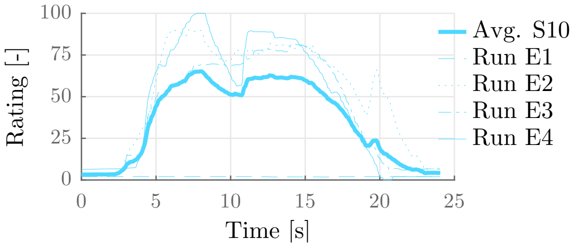


Figure F-234: Ratings by subject S10 for Condition C14. The mean rating is repeated in figure F-14.

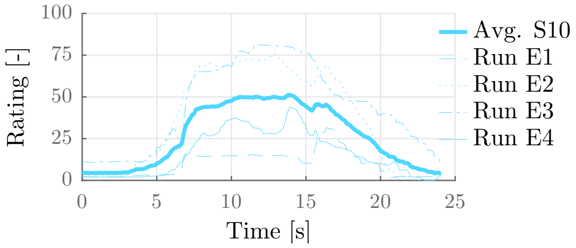


Figure F-235: Ratings by subject S10 for Condition C15. The mean rating is repeated in figure F-15.

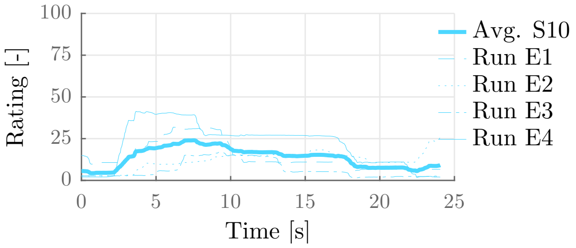


Figure F-236: Ratings by subject S10 for Condition C16. The mean rating is repeated in figure F-16.

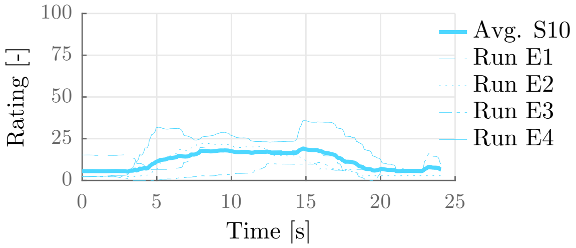


Figure F-237: Ratings by subject S10 for Condition C17. The mean rating is repeated in figure F-17.

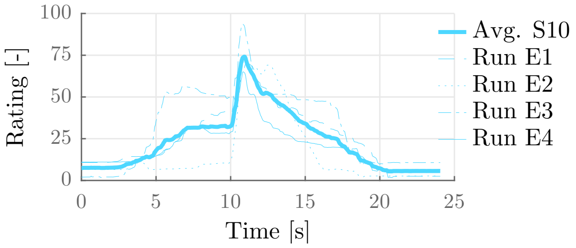


Figure F-238: Ratings by subject S10 for Condition C18. The mean rating is repeated in figure F-18.

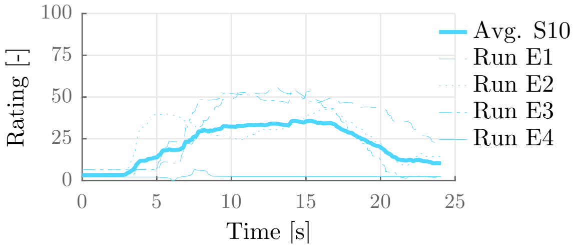


Figure F-239: Ratings by subject S10 for Condition C19. The mean rating is repeated in figure F-19.

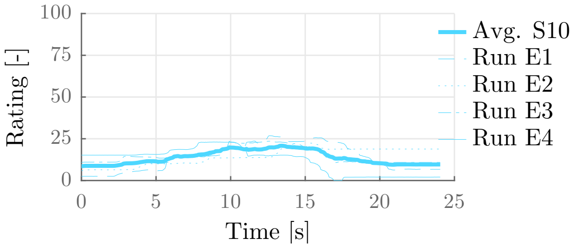


Figure F-240: Ratings by subject S10 for Condition C20. The mean rating is repeated in figure F-20.

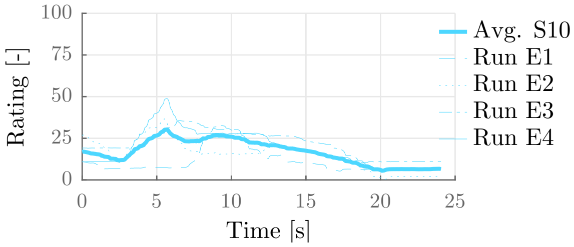


Figure F-241: Ratings by subject S10 for Condition C21. The mean rating is repeated in figure F-21.

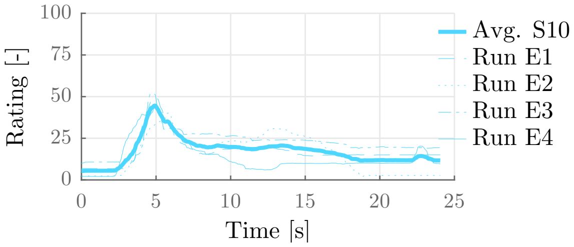


Figure F-242: Ratings by subject S10 for Condition C22. The mean rating is repeated in figure F-22.

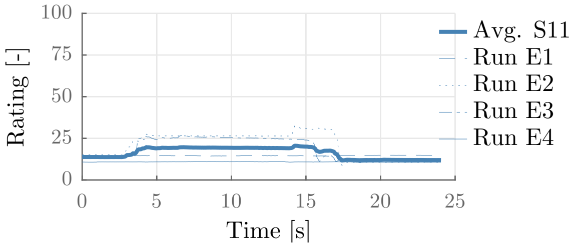


Figure F-243: Ratings by subject S11 for Condition C1. The mean rating is repeated in figure F-1.

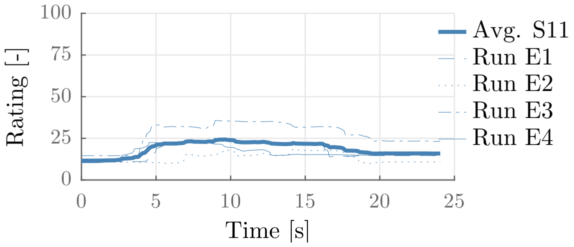


Figure F-244: Ratings by subject S11 for Condition C2. The mean rating is repeated in figure F-2.

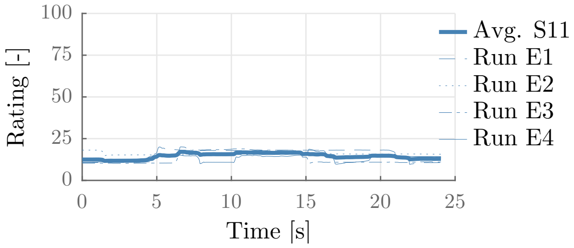


Figure F-245: Ratings by subject S11 for Condition C3. The mean rating is repeated in figure F-3.

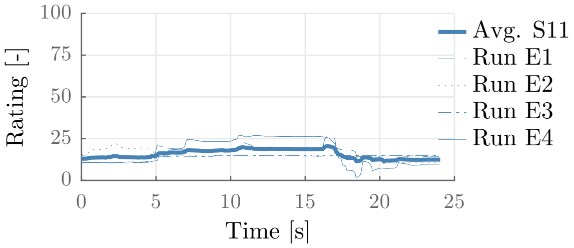


Figure F-246: Ratings by subject S11 for Condition C4. The mean rating is repeated in figure F-4.

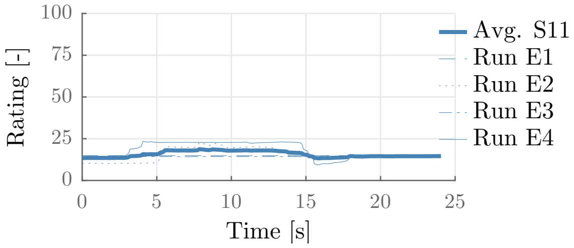


Figure F-247: Ratings by subject S11 for Condition C5. The mean rating is repeated in figure F-5.

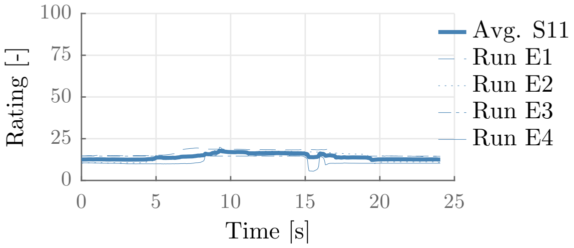


Figure F-248: Ratings by subject S11 for Condition C6. The mean rating is repeated in figure F-6.

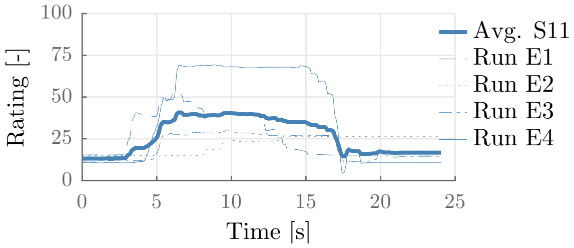


Figure F-249: Ratings by subject S11 for Condition C7. The mean rating is repeated in figure F-7.

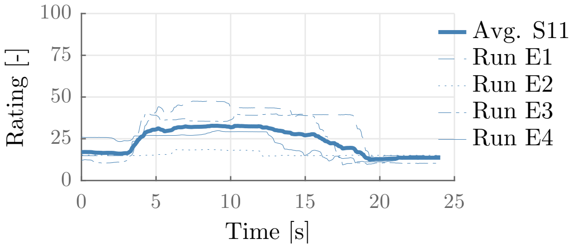


Figure F-250: Ratings by subject S11 for Condition C8. The mean rating is repeated in figure F-8.

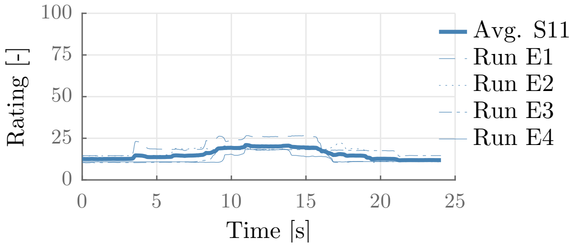


Figure F-251: Ratings by subject S11 for Condition C9. The mean rating is repeated in figure F-9.

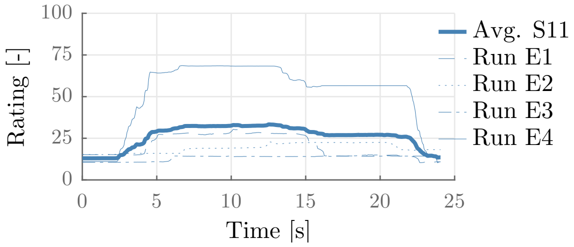


Figure F-252: Ratings by subject S11 for Condition C10. The mean rating is repeated in figure F-10.

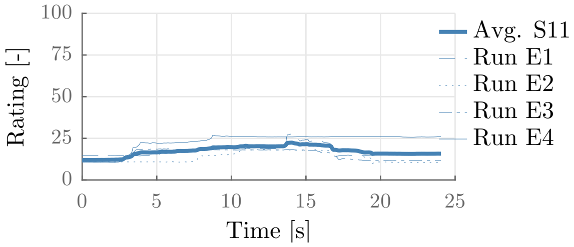


Figure F-253: Ratings by subject S11 for Condition C11. The mean rating is repeated in figure F-11.

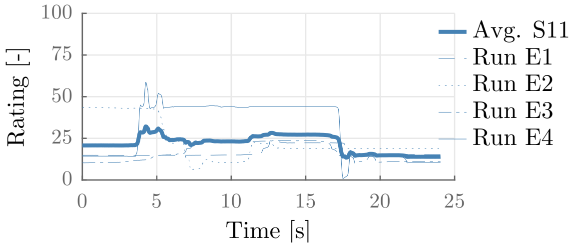


Figure F-254: Ratings by subject S11 for Condition C12. The mean rating is repeated in figure F-12.

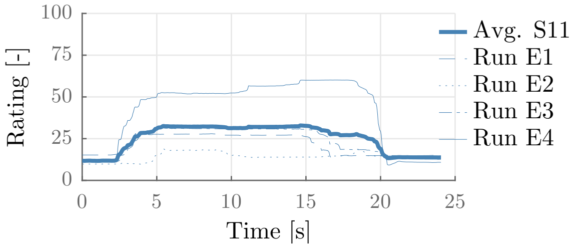


Figure F-255: Ratings by subject S11 for Condition C13. The mean rating is repeated in figure F-13.

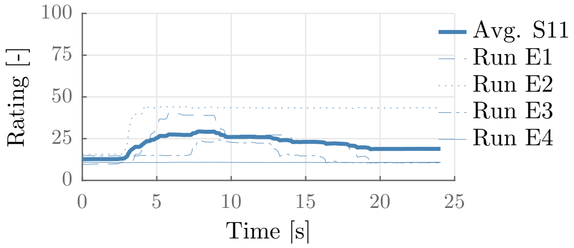


Figure F-256: Ratings by subject S11 for Condition C14. The mean rating is repeated in figure F-14.

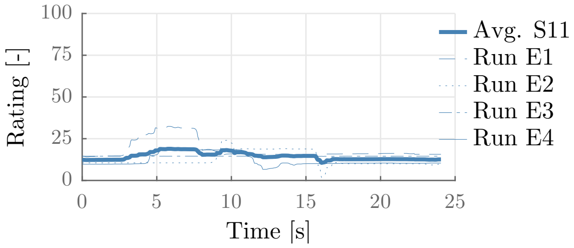


Figure F-257: Ratings by subject S11 for Condition C15. The mean rating is repeated in figure F-15.

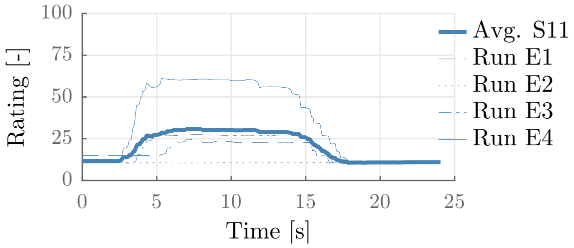


Figure F-258: Ratings by subject S11 for Condition C16. The mean rating is repeated in figure F-16.

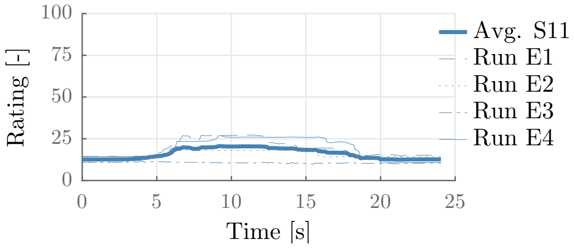


Figure F-259: Ratings by subject S11 for Condition C17. The mean rating is repeated in figure F-17.

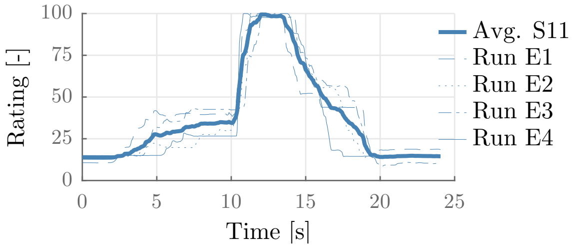
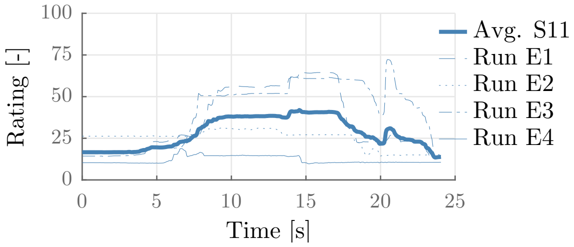


Figure F-260: Ratings by subject S11 for Condition C18. The mean rating is repeated in figure F-18.



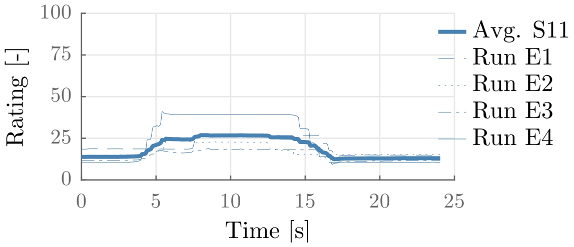
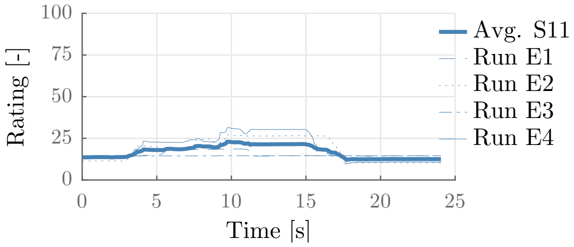


Figure F-261: Ratings by subject S11 for Condition C19. The mean rating is repeated in figure F-19.

Figure F-262: Ratings by subject S11 for Condition C20. The mean rating is repeated in figure F-20.



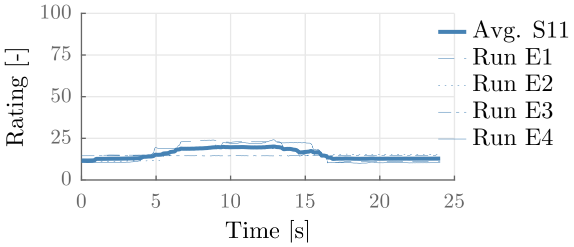
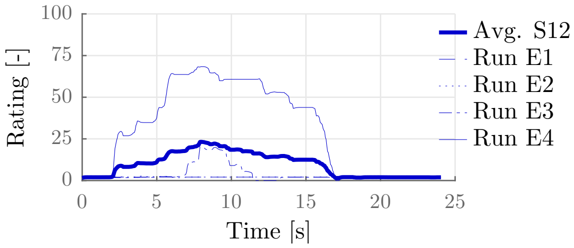


Figure F-263: Ratings by subject S11 for Condition C21. The mean rating is repeated in figure F-21.

Figure F-264: Ratings by subject S11 for Condition C22. The mean rating is repeated in figure F-22.



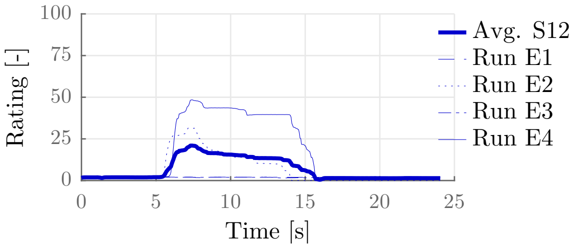
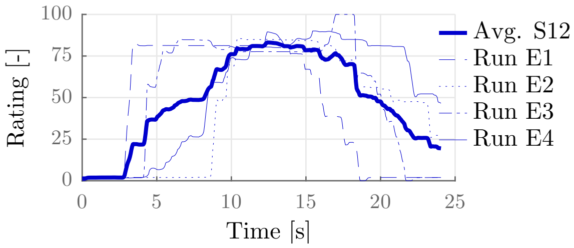


Figure F-265: Ratings by subject S12 for Condition C1. The mean rating is repeated in figure F-1.

Figure F-266: Ratings by subject S12 for Condition C2. The mean rating is repeated in figure F-2.



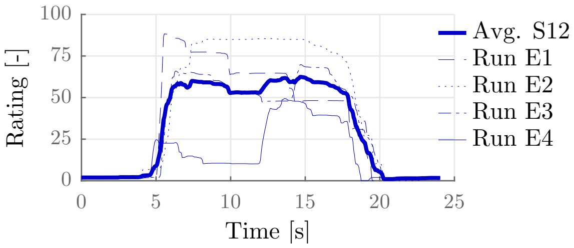
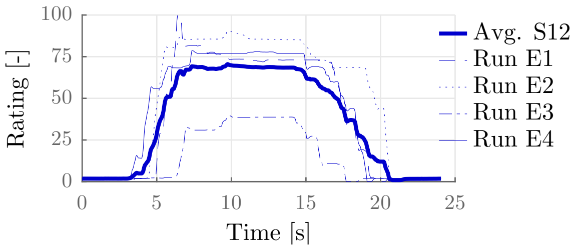


Figure F-267: Ratings by subject S12 for Condition C3. The mean rating is repeated in figure F-3.

Figure F-268: Ratings by subject S12 for Condition C4. The mean rating is repeated in figure F-4.



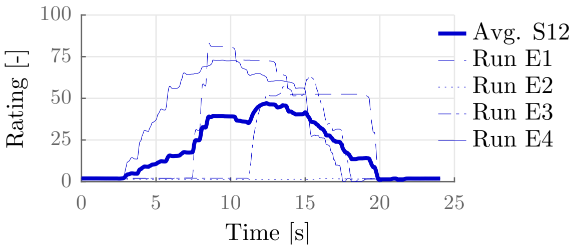


Figure F-269: Ratings by subject S12 for Condition C5. The mean rating is repeated in figure F-5.

Figure F-270: Ratings by subject S12 for Condition C6. The mean rating is repeated in figure F-6.

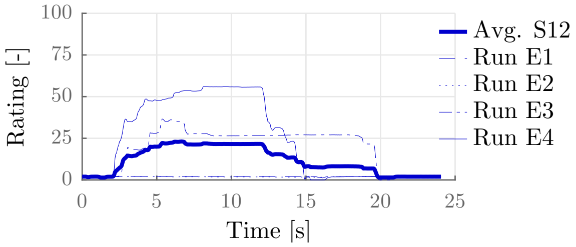


Figure F-271: Ratings by subject S12 for Condition C7. The mean rating is repeated in figure F-7.

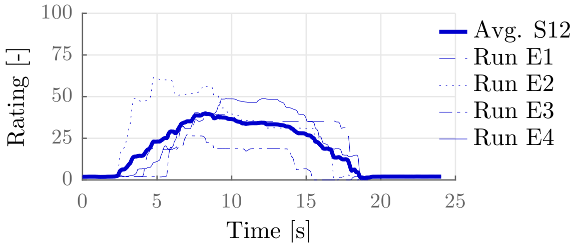


Figure F-272: Ratings by subject S12 for Condition C8. The mean rating is repeated in figure F-8.

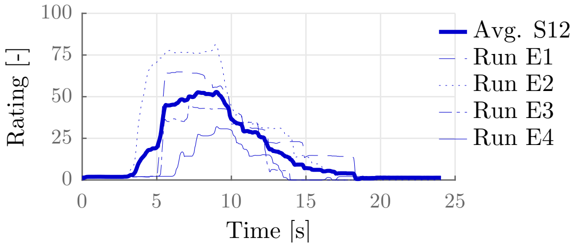


Figure F-273: Ratings by subject S12 for Condition C9. The mean rating is repeated in figure F-9.

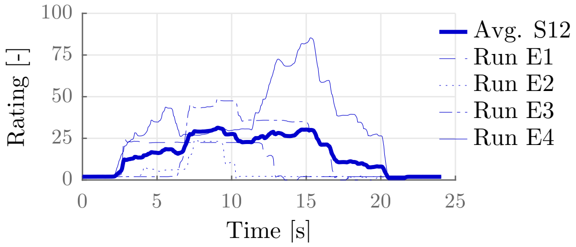


Figure F-274: Ratings by subject S12 for Condition C10. The mean rating is repeated in figure F-10.

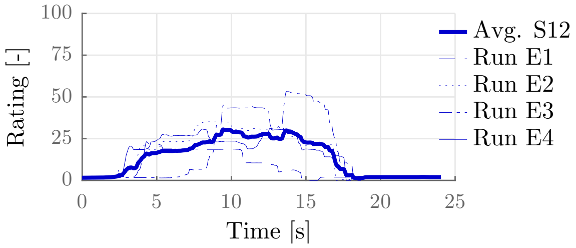


Figure F-275: Ratings by subject S12 for Condition C11. The mean rating is repeated in figure F-11.

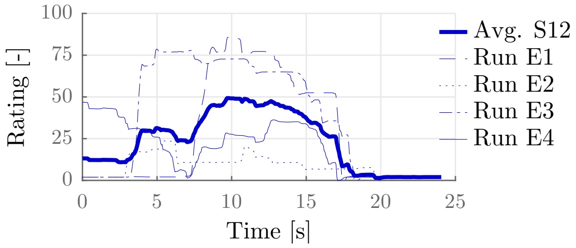


Figure F-276: Ratings by subject S12 for Condition C12. The mean rating is repeated in figure F-12.

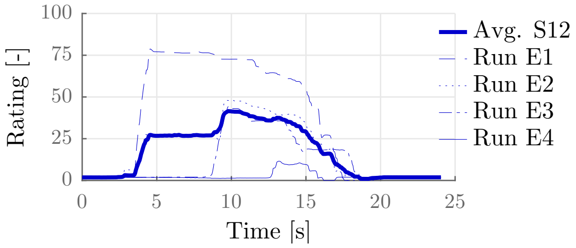


Figure F-277: Ratings by subject S12 for Condition C13. The mean rating is repeated in figure F-13.

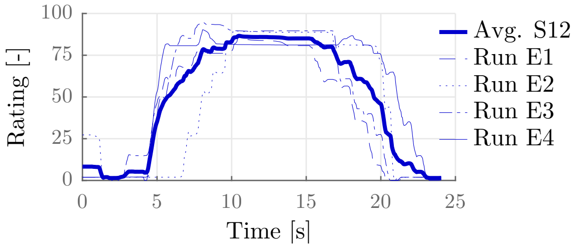


Figure F-278: Ratings by subject S12 for Condition C14. The mean rating is repeated in figure F-14.

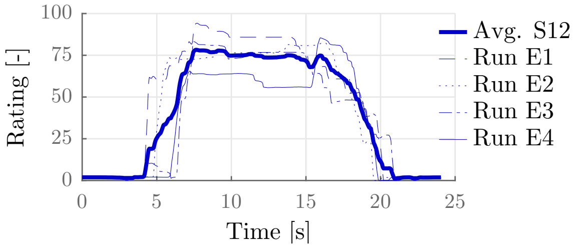


Figure F-279: Ratings by subject S12 for Condition C15. The mean rating is repeated in figure F-15.

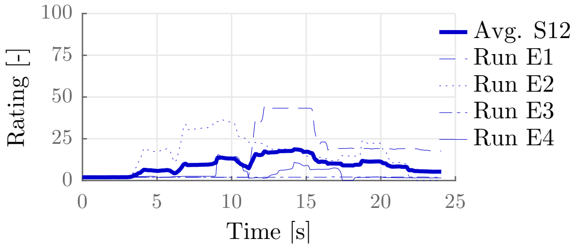


Figure F-280: Ratings by subject S12 for Condition C16. The mean rating is repeated in figure F-16.

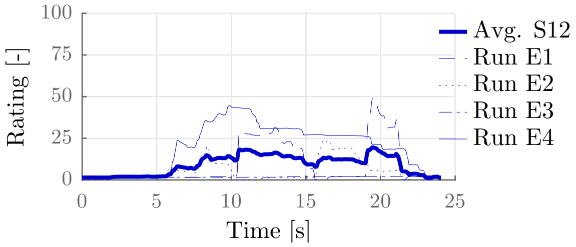


Figure F-281: Ratings by subject S12 for Condition C17. The mean rating is repeated in figure F-17.

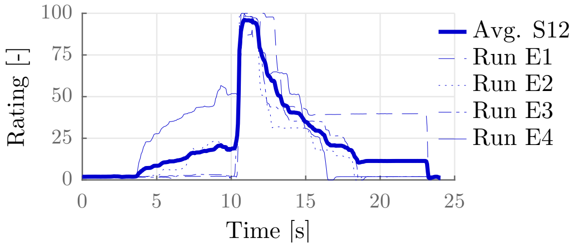


Figure F-282: Ratings by subject S12 for Condition C18. The mean rating is repeated in figure F-18.

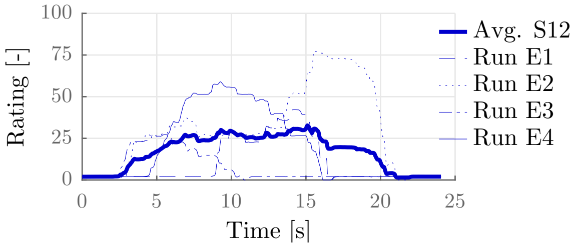


Figure F-283: Ratings by subject S12 for Condition C19. The mean rating is repeated in figure F-19.

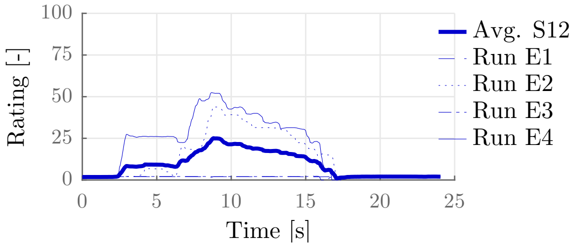


Figure F-284: Ratings by subject S12 for Condition C20. The mean rating is repeated in figure F-20.

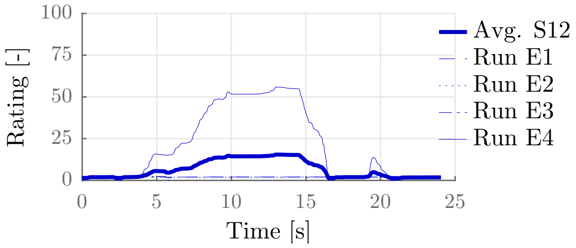


Figure F-285: Ratings by subject S12 for Condition C21. The mean rating is repeated in figure F-21.

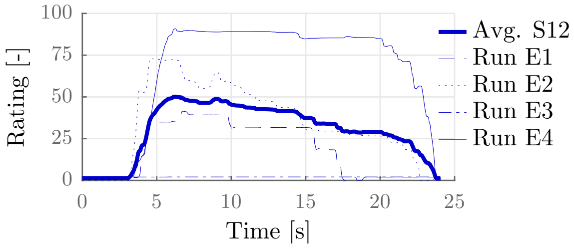


Figure F-286: Ratings by subject S12 for Condition C22. The mean rating is repeated in figure F-22.

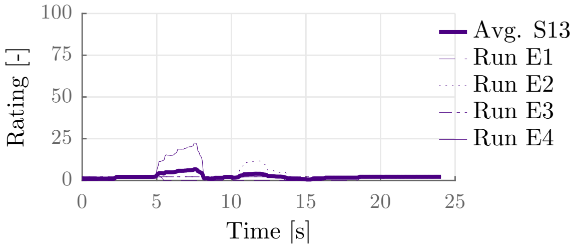


Figure F-287: Ratings by subject S13 for Condition C1. The mean rating is repeated in figure F-1.

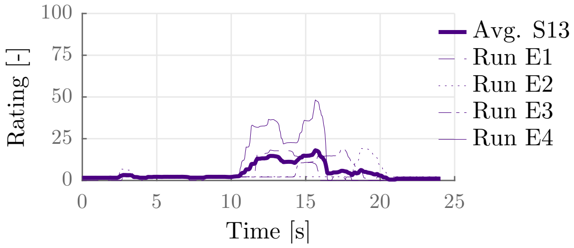


Figure F-288: Ratings by subject S13 for Condition C2. The mean rating is repeated in figure F-2.

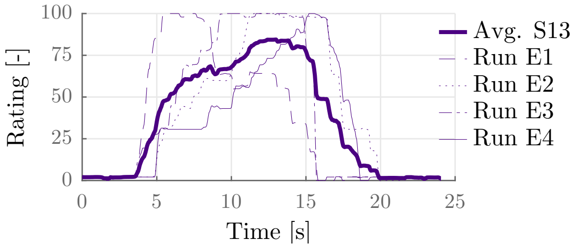


Figure F-289: Ratings by subject S13 for Condition C3. The mean rating is repeated in figure F-3.

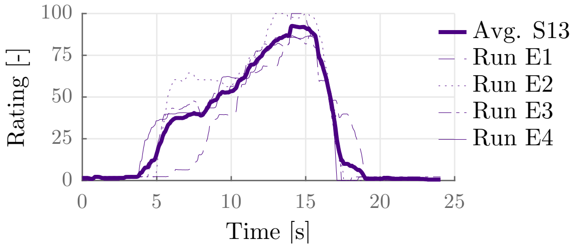


Figure F-290: Ratings by subject S13 for Condition C4. The mean rating is repeated in figure F-4.

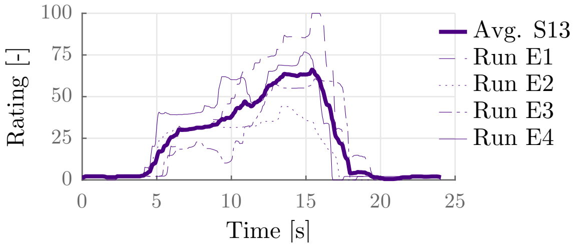


Figure F-291: Ratings by subject S13 for Condition C5. The mean rating is repeated in figure F-5.

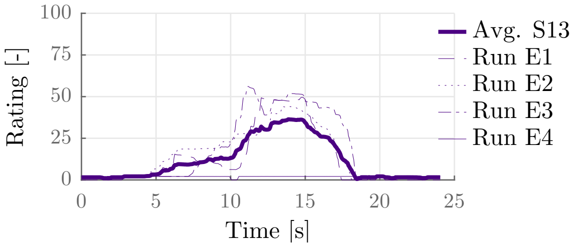


Figure F-292: Ratings by subject S13 for Condition C6. The mean rating is repeated in figure F-6.

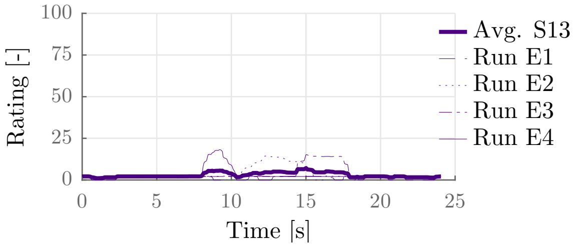


Figure F-293: Ratings by subject S13 for Condition C7. The mean rating is repeated in figure F-7.

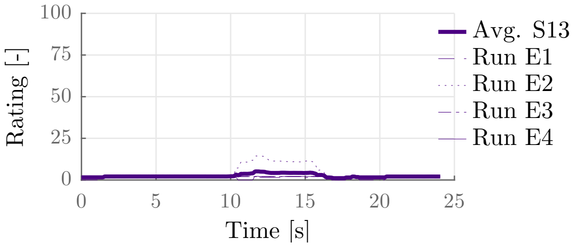


Figure F-294: Ratings by subject S13 for Condition C8. The mean rating is repeated in figure F-8.

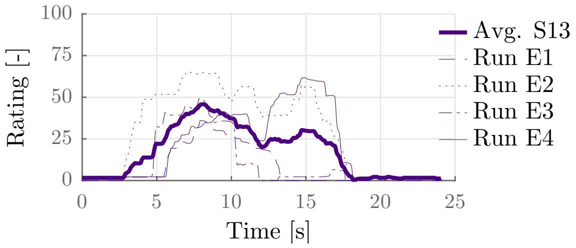


Figure F-295: Ratings by subject S13 for Condition C9. The mean rating is repeated in figure F-9.

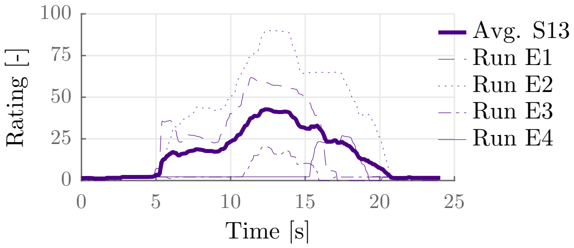


Figure F-296: Ratings by subject S13 for Condition C10. The mean rating is repeated in figure F-10.

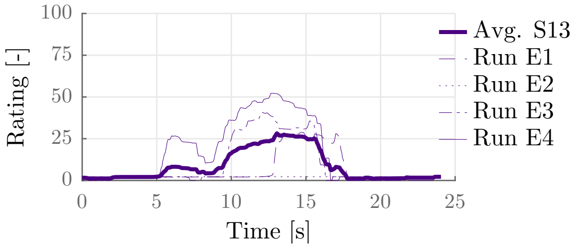


Figure F-297: Ratings by subject S13 for Condition C11. The mean rating is repeated in figure F-11.

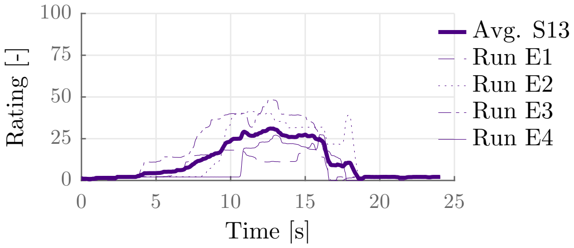


Figure F-298: Ratings by subject S13 for Condition C12. The mean rating is repeated in figure F-12.

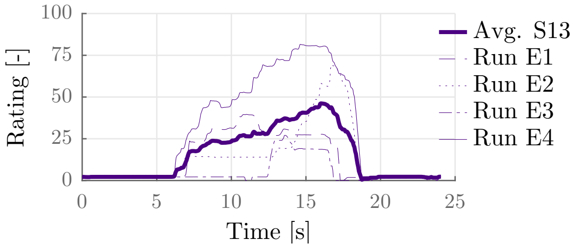


Figure F-299: Ratings by subject S13 for Condition C13. The mean rating is repeated in figure F-13.

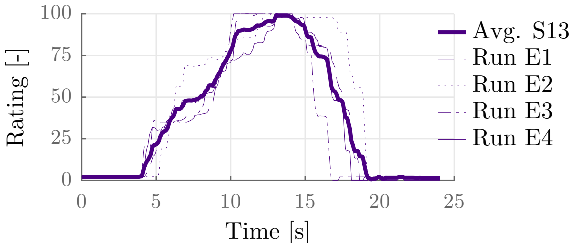


Figure F-300: Ratings by subject S13 for Condition C14. The mean rating is repeated in figure F-14.

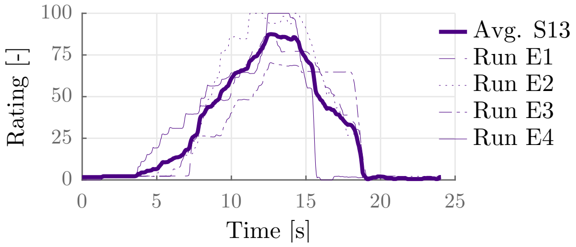


Figure F-301: Ratings by subject S13 for Condition C15. The mean rating is repeated in figure F-15.

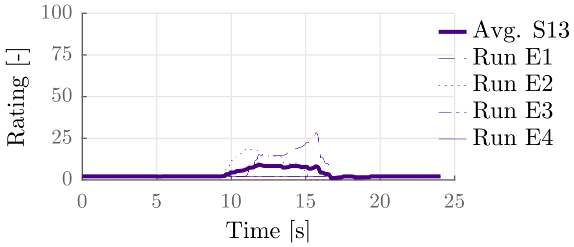


Figure F-302: Ratings by subject S13 for Condition C16. The mean rating is repeated in figure F-16.

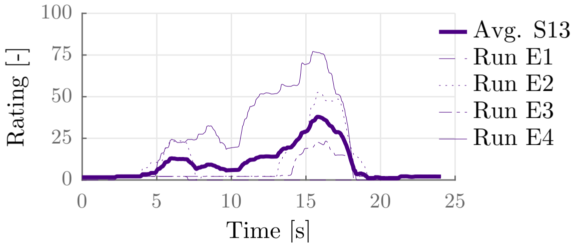


Figure F-303: Ratings by subject S13 for Condition C17. The mean rating is repeated in figure F-17.

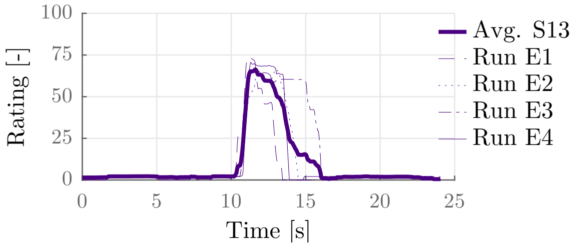


Figure F-304: Ratings by subject S13 for Condition C18. The mean rating is repeated in figure F-18.

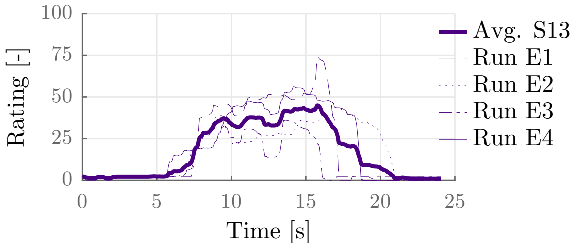


Figure F-305: Ratings by subject S13 for Condition C19. The mean rating is repeated in figure F-19.

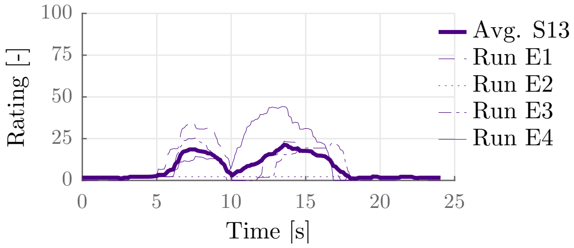


Figure F-306: Ratings by subject S13 for Condition C20. The mean rating is repeated in figure F-20.

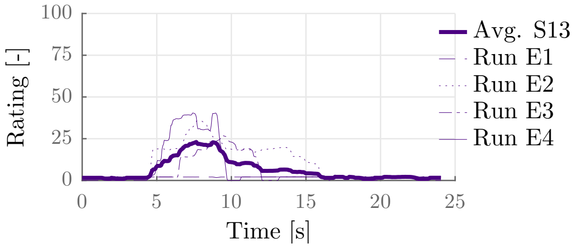


Figure F-307: Ratings by subject S13 for Condition C21. The mean rating is repeated in figure F-21.

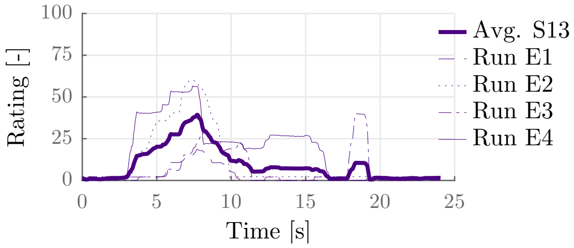


Figure F-308: Ratings by subject S13 for Condition C22. The mean rating is repeated in figure F-22.

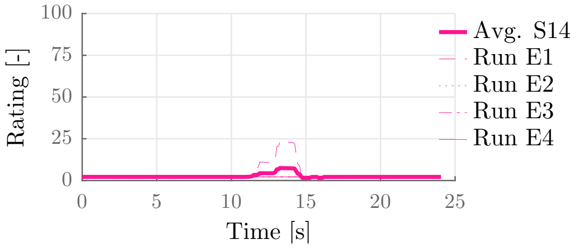


Figure F-309: Ratings by subject S14 for Condition C1. The mean rating is repeated in figure F-1.

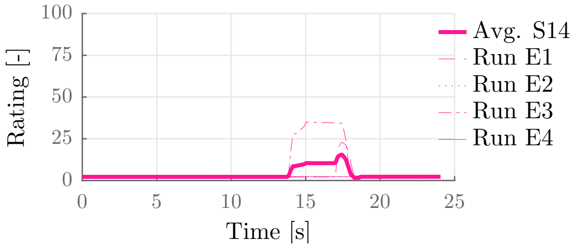


Figure F-310: Ratings by subject S14 for Condition C2. The mean rating is repeated in figure F-2.

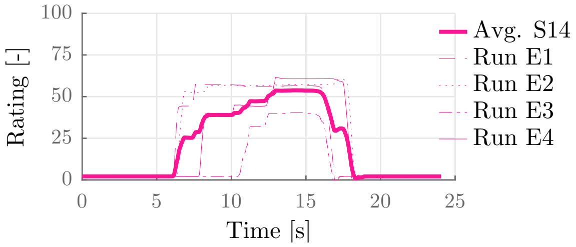


Figure F-311: Ratings by subject S14 for Condition C3. The mean rating is repeated in figure F-3.

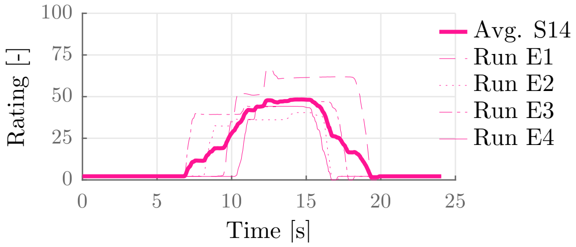


Figure F-312: Ratings by subject S14 for Condition C4. The mean rating is repeated in figure F-4.

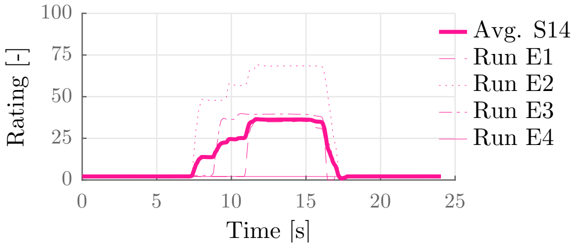


Figure F-313: Ratings by subject S14 for Condition C5. The mean rating is repeated in figure F-5.

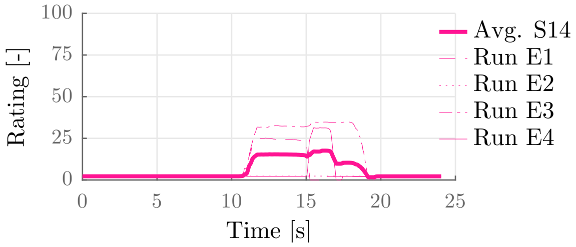


Figure F-314: Ratings by subject S14 for Condition C6. The mean rating is repeated in figure F-6.

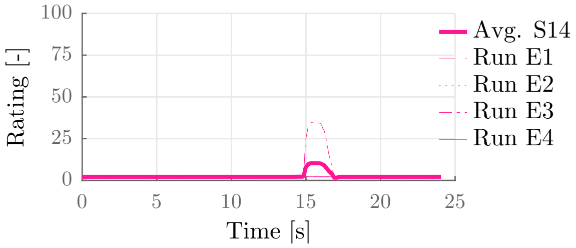


Figure F-315: Ratings by subject S14 for Condition C7. The mean rating is repeated in figure F-7.

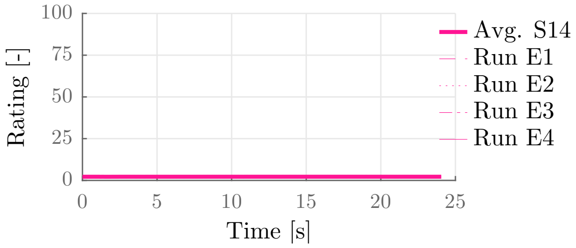


Figure F-316: Ratings by subject S14 for Condition C8. The mean rating is repeated in figure F-8.

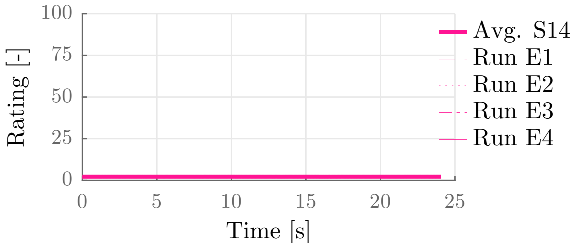


Figure F-317: Ratings by subject S14 for Condition C9. The mean rating is repeated in figure F-9.

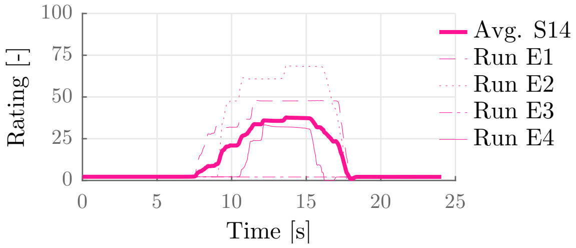


Figure F-318: Ratings by subject S14 for Condition C10. The mean rating is repeated in figure F-10.

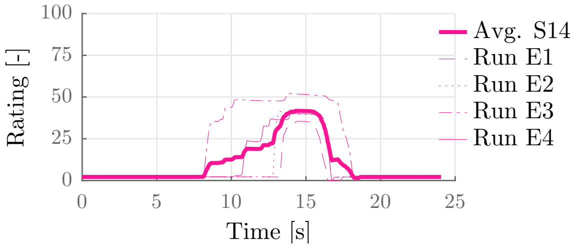


Figure F-319: Ratings by subject S14 for Condition C11. The mean rating is repeated in figure F-11.

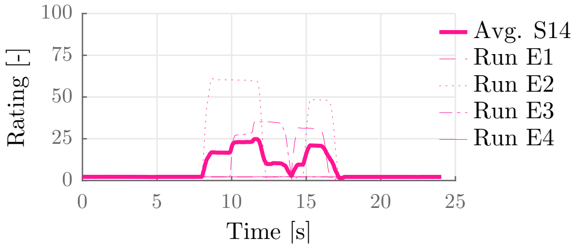


Figure F-320: Ratings by subject S14 for Condition C12. The mean rating is repeated in figure F-12.

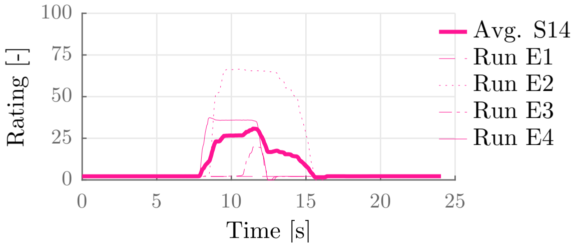


Figure F-321: Ratings by subject S14 for Condition C13. The mean rating is repeated in figure F-13.

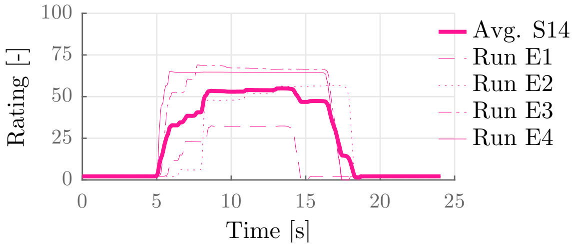


Figure F-322: Ratings by subject S14 for Condition C14. The mean rating is repeated in figure F-14.

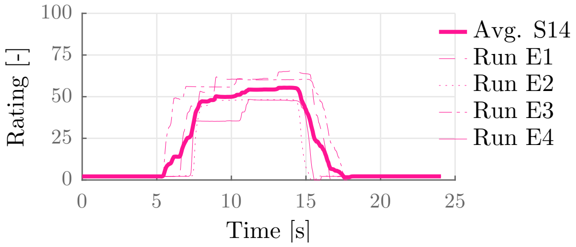


Figure F-323: Ratings by subject S14 for Condition C15. The mean rating is repeated in figure F-15.

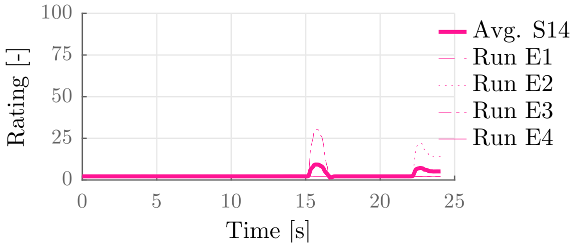


Figure F-324: Ratings by subject S14 for Condition C16. The mean rating is repeated in figure F-16.

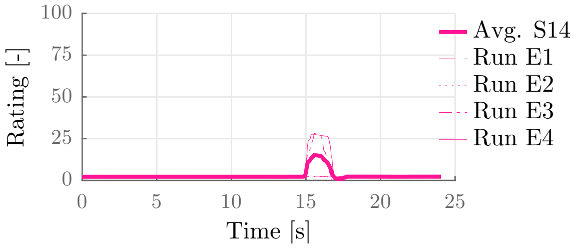


Figure F-325: Ratings by subject S14 for Condition C17. The mean rating is repeated in figure F-17.

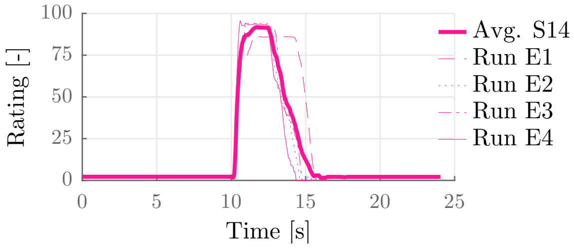


Figure F-326: Ratings by subject S14 for Condition C18. The mean rating is repeated in figure F-18.

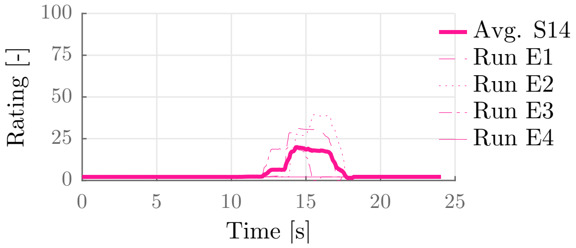


Figure F-327: Ratings by subject S14 for Condition C19. The mean rating is repeated in figure F-19.

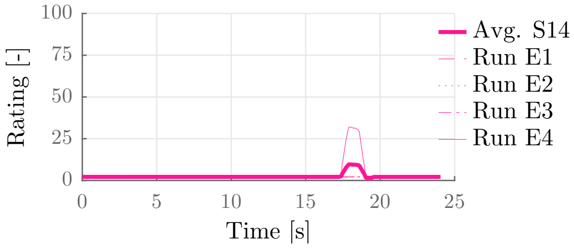


Figure F-328: Ratings by subject S14 for Condition C20. The mean rating is repeated in figure F-20.

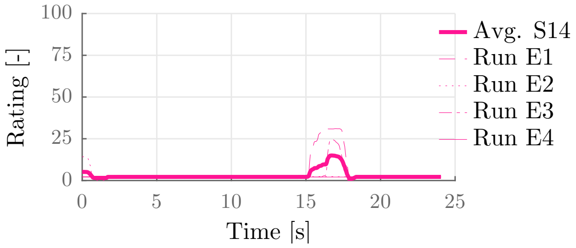


Figure F-329: Ratings by subject S14 for Condition C21. The mean rating is repeated in figure F-21.

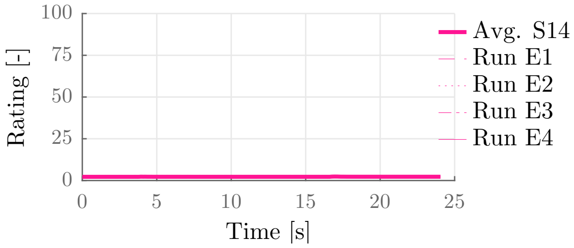


Figure F-330: Ratings by subject S14 for Condition C22. The mean rating is repeated in figure F-22.

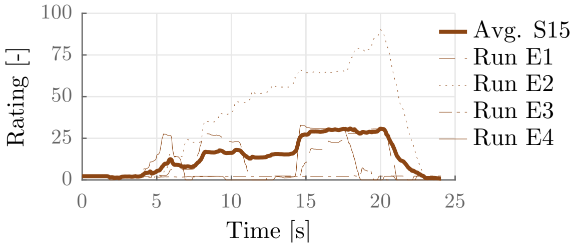


Figure F-331: Ratings by subject S15 for Condition C1. The mean rating is repeated in figure F-1.

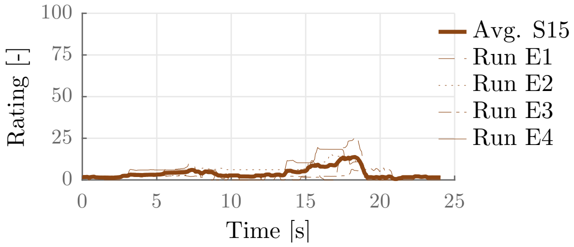


Figure F-332: Ratings by subject S15 for Condition C2. The mean rating is repeated in figure F-2.

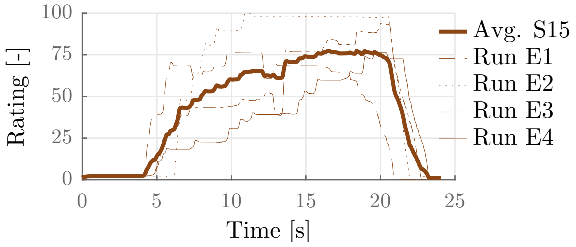


Figure F-333: Ratings by subject S15 for Condition C3. The mean rating is repeated in figure F-3.

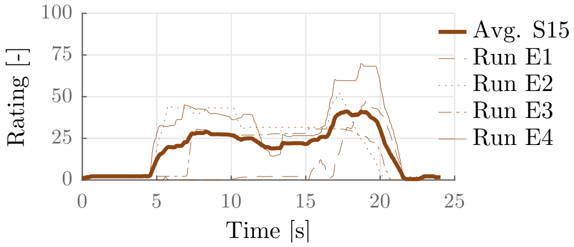


Figure F-334: Ratings by subject S15 for Condition C4. The mean rating is repeated in figure F-4.

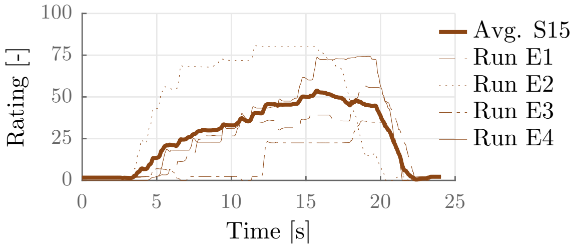


Figure F-335: Ratings by subject S15 for Condition C5. The mean rating is repeated in figure F-5.

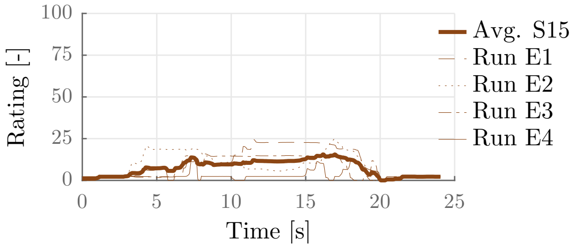


Figure F-336: Ratings by subject S15 for Condition C6. The mean rating is repeated in figure F-6.

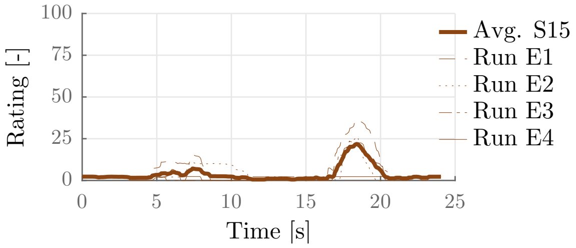


Figure F-337: Ratings by subject S15 for Condition C7. The mean rating is repeated in figure F-7.

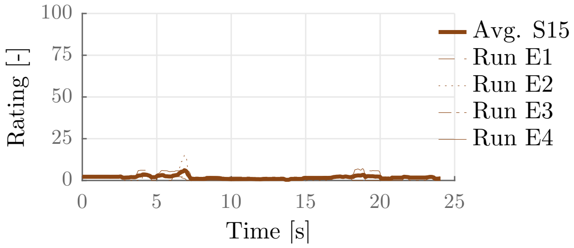


Figure F-338: Ratings by subject S15 for Condition C8. The mean rating is repeated in figure F-8.

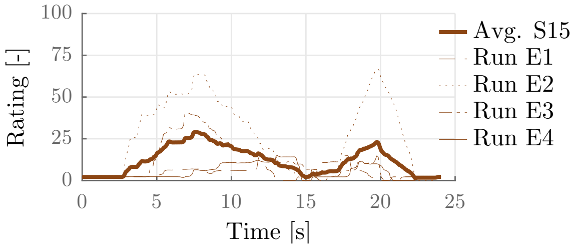


Figure F-339: Ratings by subject S15 for Condition C9. The mean rating is repeated in figure F-9.

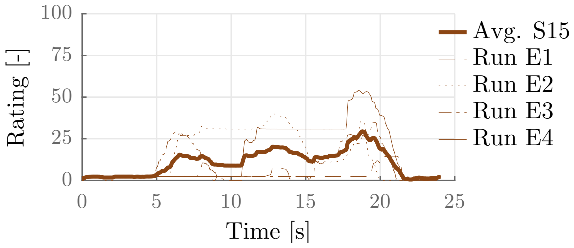


Figure F-340: Ratings by subject S15 for Condition C10. The mean rating is repeated in figure F-10.

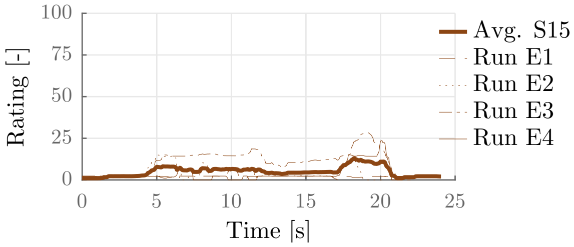


Figure F-341: Ratings by subject S15 for Condition C11. The mean rating is repeated in figure F-11.

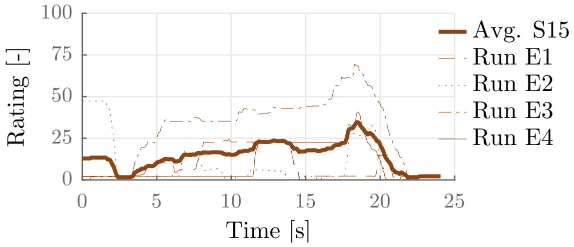


Figure F-342: Ratings by subject S15 for Condition C12. The mean rating is repeated in figure F-12.

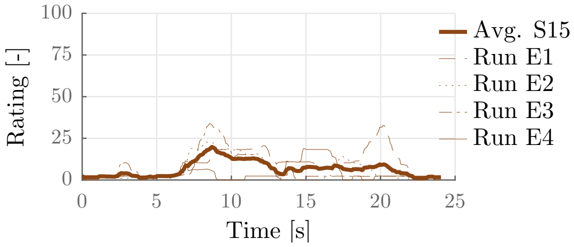


Figure F-343: Ratings by subject S15 for Condition C13. The mean rating is repeated in figure F-13.

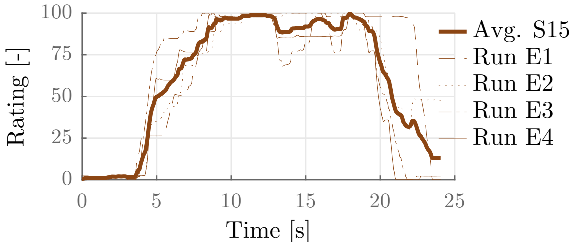


Figure F-344: Ratings by subject S15 for Condition C14. The mean rating is repeated in figure F-14.

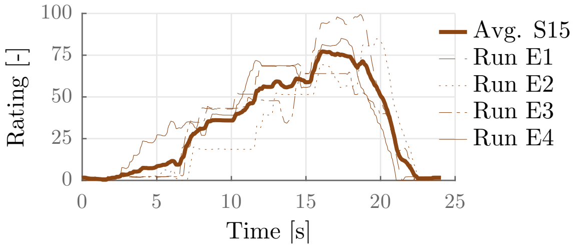


Figure F-345: Ratings by subject S15 for Condition C15. The mean rating is repeated in figure F-15.

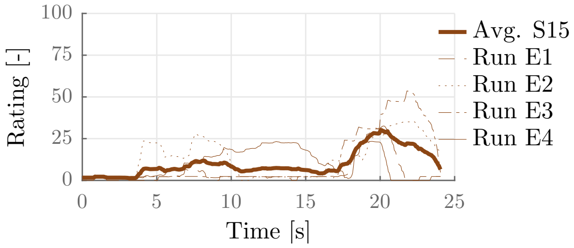


Figure F-346: Ratings by subject S15 for Condition C16. The mean rating is repeated in figure F-16.

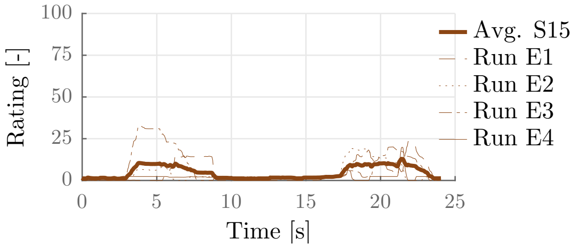


Figure F-347: Ratings by subject S15 for Condition C17. The mean rating is repeated in figure F-17.

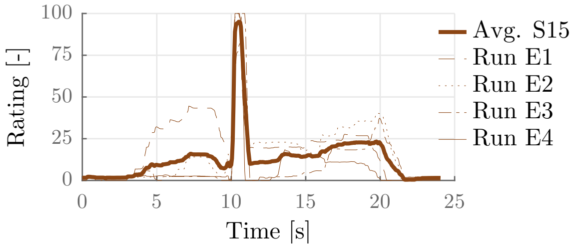


Figure F-348: Ratings by subject S15 for Condition C18. The mean rating is repeated in figure F-18.

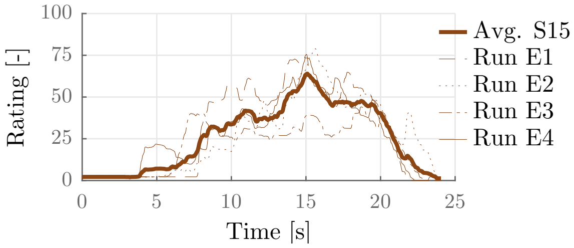


Figure F-349: Ratings by subject S15 for Condition C19. The mean rating is repeated in figure F-19.

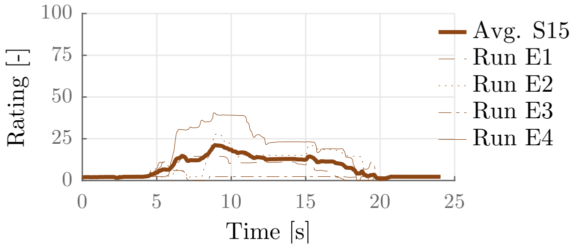


Figure F-350: Ratings by subject S15 for Condition C20. The mean rating is repeated in figure F-20.

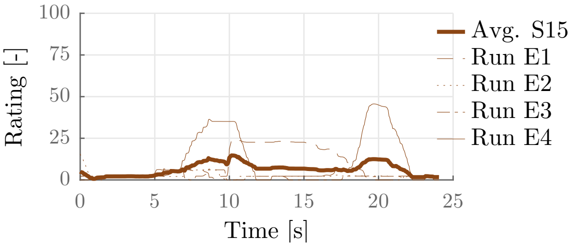


Figure F-351: Ratings by subject S15 for Condition C21. The mean rating is repeated in figure F-21.

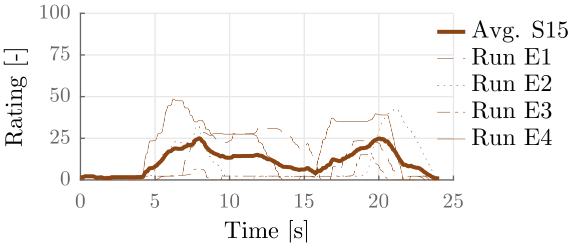


Figure F-352: Ratings by subject S15 for Condition C22. The mean rating is repeated in figure F-22.

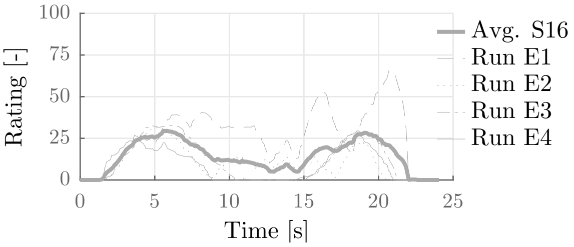


Figure F-353: Ratings by subject S16 for Condition C1. The mean rating is repeated in figure F-1.

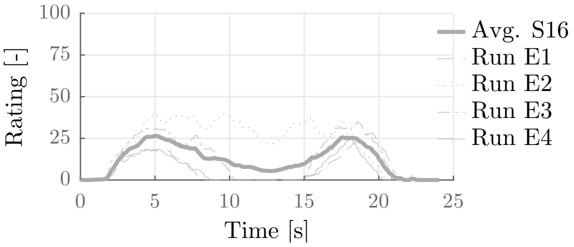


Figure F-354: Ratings by subject S16 for Condition C2. The mean rating is repeated in figure F-2.

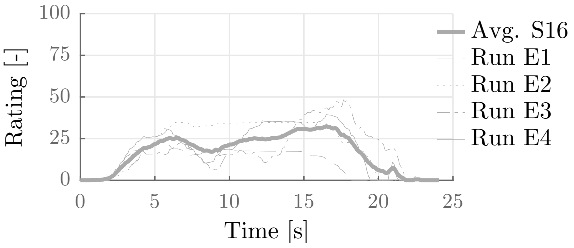


Figure F-355: Ratings by subject S16 for Condition C3. The mean rating is repeated in figure F-3.

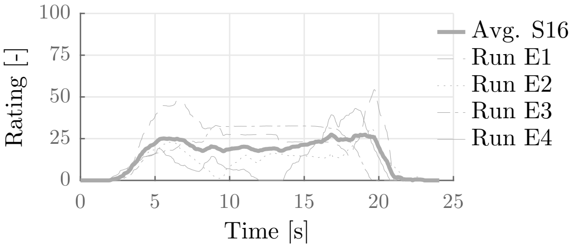


Figure F-356: Ratings by subject S16 for Condition C4. The mean rating is repeated in figure F-4.

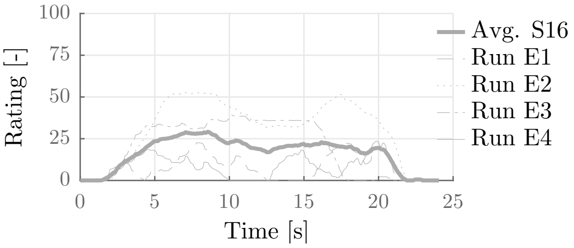


Figure F-357: Ratings by subject S16 for Condition C5. The mean rating is repeated in figure F-5.

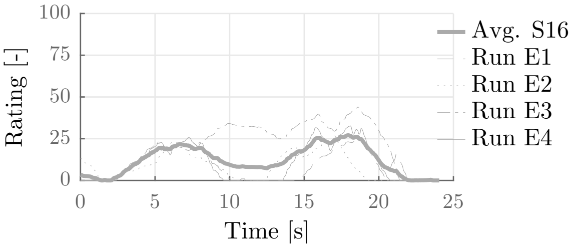


Figure F-358: Ratings by subject S16 for Condition C6. The mean rating is repeated in figure F-6.

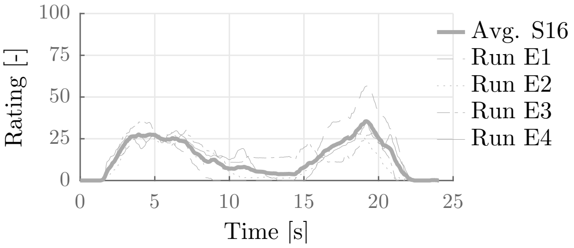


Figure F-359: Ratings by subject S16 for Condition C7. The mean rating is repeated in figure F-7.

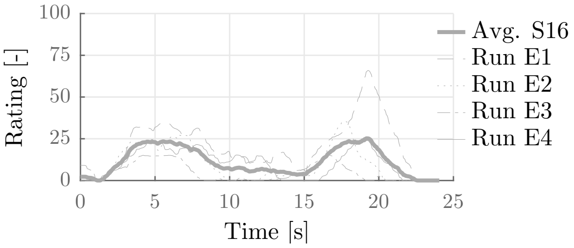


Figure F-360: Ratings by subject S16 for Condition C8. The mean rating is repeated in figure F-8.

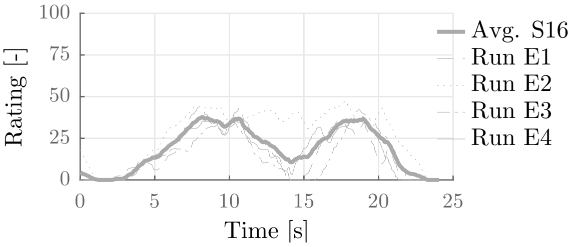


Figure F-361: Ratings by subject S16 for Condition C9. The mean rating is repeated in figure F-9.

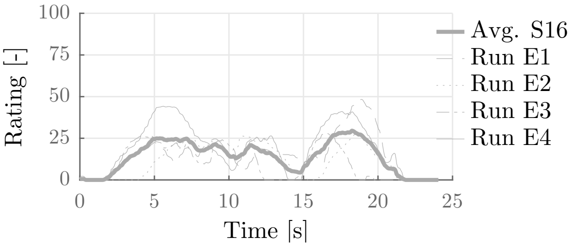


Figure F-362: Ratings by subject S16 for Condition C10. The mean rating is repeated in figure F-10.

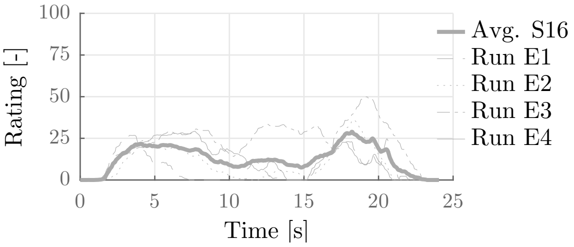


Figure F-363: Ratings by subject S16 for Condition C11. The mean rating is repeated in figure F-11.

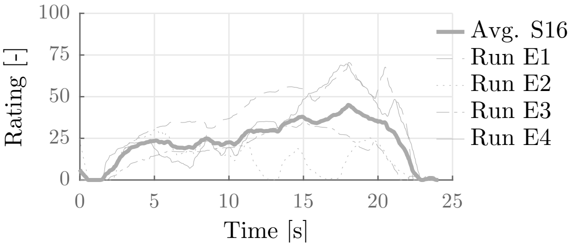


Figure F-364: Ratings by subject S16 for Condition C12. The mean rating is repeated in figure F-12.

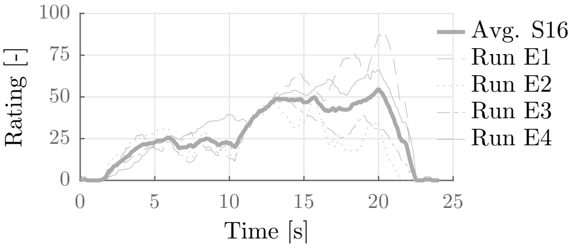


Figure F-365: Ratings by subject S16 for Condition C13. The mean rating is repeated in figure F-13.

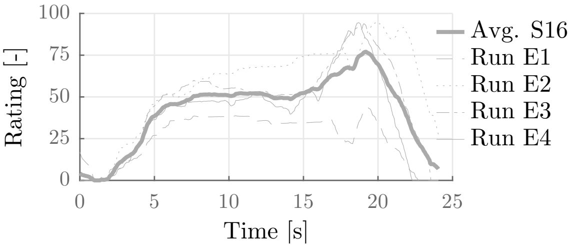


Figure F-366: Ratings by subject S16 for Condition C14. The mean rating is repeated in figure F-14.

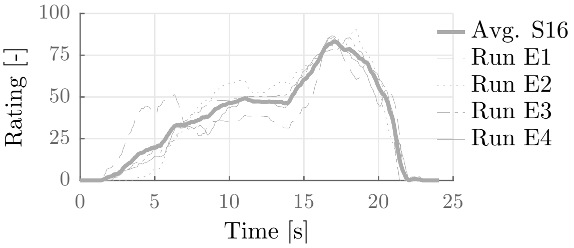


Figure F-367: Ratings by subject S16 for Condition C15. The mean rating is repeated in figure F-15.

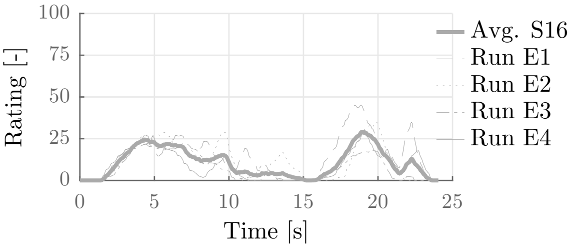


Figure F-368: Ratings by subject S16 for Condition C16. The mean rating is repeated in figure F-16.

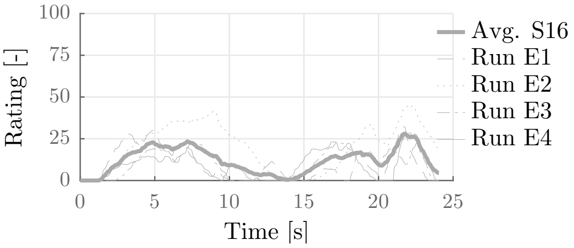


Figure F-369: Ratings by subject S16 for Condition C17. The mean rating is repeated in figure F-17.

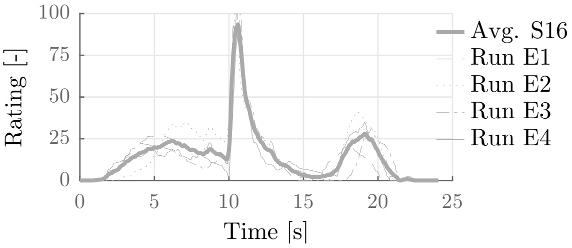


Figure F-370: Ratings by subject S16 for Condition C18. The mean rating is repeated in figure F-18.

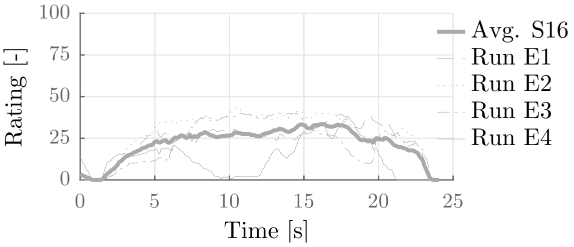


Figure F-371: Ratings by subject S16 for Condition C19. The mean rating is repeated in figure F-19.

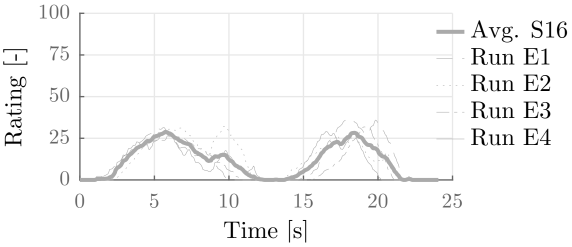


Figure F-372: Ratings by subject S16 for Condition C20. The mean rating is repeated in figure F-20.

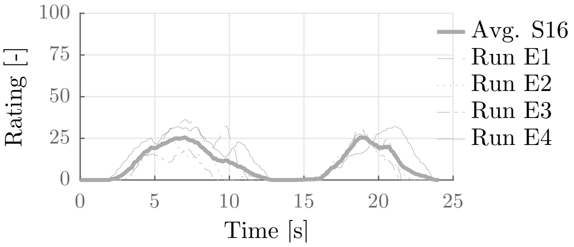


Figure F-373: Ratings by subject S16 for Condition C21. The mean rating is repeated in figure F-21.

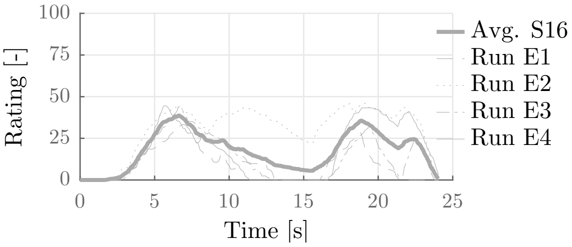


Figure F-374: Ratings by subject S16 for Condition C22. The mean rating is repeated in figure F-22.

Appendix G

Algorithm experiment output

To demonstrate the workings (and shortcomings) of the algorithm, the vehicle inertial motion and MCA output used during the experiment described in the IEEE paper has been analysed using the motion cueing error detection algorithm. The vehicle inertial motion for the drive recorded for experimental run E1 is shown in figure G-1 for all six motion channels. MCA output for this motion condition is also shown in this figure. As shown in Appendix E, this MCA output was reproduced accurately by the CMS.

The vehicle inertial motion and MCA output were given to the motion cueing error detection as an input. The parameters of this algorithm were set to those found in the experiment. For completeness, an overview of these parameters is given in table G-1.

The output of the algorithm is shown in figure G-2. The following comments can be made about this output:

a_x **-** For the longitudinal specific force a_x scaling errors were detected at the begin and end of the run. Given that a scaling factor of 0.3 was applied for longitudinal motion, this finding was expected. The algorithm also detects F2 errors in nearly every corner. This finding is attributed to the combination of the applied tilt coordination and the slight yaw of the cabin, which introduces a small false longitudinal specific force slightly above threshold level Φ.

a_y **-** The lateral specific force signal features the most motion cueing errors, of varying type. Most errors are identified correctly, however, a number of comments should be made.

1. The F3a error near $t = 75$ s is detected as both an F3a and F3b error. The same holds for the accompanying error in ω_x. Increasing the maximum allowable gap between equal errors would ensure these errors are detected as one single F3b error, increasing the minimum duration of an F3b error (Δt_{min}) could ensure this error is labelled as an F3a error only instead.
2. The two smallest phase errors (with a delay of 0.6 s and 1.5 s) are detected, the largest phase error (with a delay of 2.5 s, at $t =$ 350-375 s), however, is not labelled

as a phase error. Instead, an S2 and F2 are identified in succession. This deficiency highlights the problems encountered with phase errors, and suggests these errors might have to be redefined in terms of properties.

3. The F3b error between t = 200-225 s is not detected by the algorithm. This is most probably caused by the high-frequency disturbance not containing enough power.
4. Experiment run E1 features condition C12 between t = 100-125 s. The coherence level $\rho^2_{\mathrm{LF}}(t)$ of condition C12 drops to 0.65, which is below the threshold level $\rho^2_{\mathrm{LF,min}} = 0.7$. This error is not detected, however, since the Power Ratio does not drop below 0.7.

a_z **-** For the vertical force a false cue is detected in nearly every corner. This finding is the result of the applied tilt coordination angle, which decreases the gravity component along a subject's z-axis.

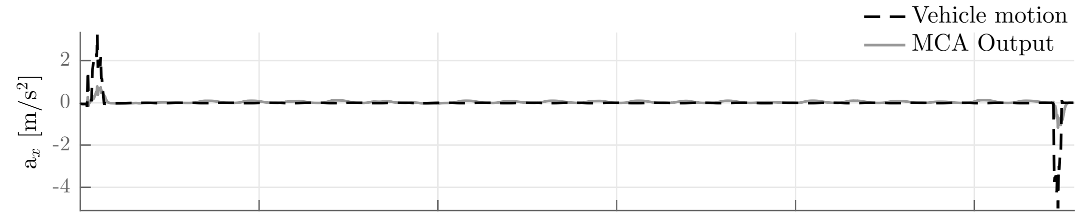
ω_x **-** The roll rate ω_x of the vehicle is always below threshold level, the MCA output only exceeds the threshold level Φ around $t = 500$ s. This error is detected by the algorithm as an F2 error.

ω_y **-** The pitch rate ω_y only exceeds the threshold level Φ on one occasion, around $t = 545$ s. This error is detected by the algorithm as an F2 error.

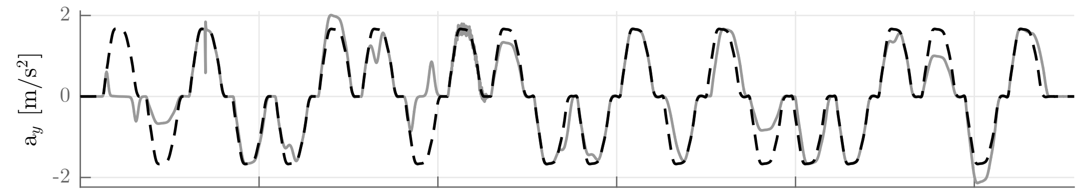
ω_z **-** The yaw rate ω_z of the vehicle is not simulated by the MCA, as a result a succession of S2 errors is detected in each corner.

Table G-1: Parameter settings and values used during the analysis of the inertial motion presented in the experiment described in the IEEE paper.

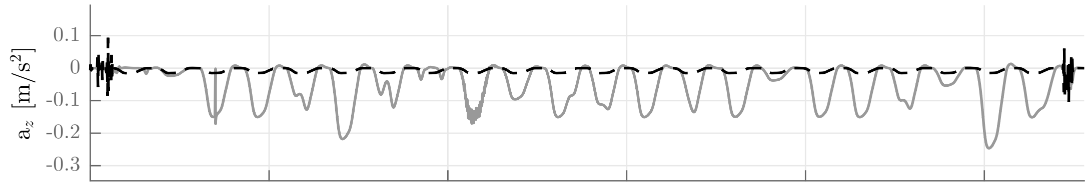
Algorithm step	**Parameter**	**Value**
Input	Δt (Input)	0.001 s
	Δt (Output)	0.001 s
Step 0&3	$\Phi(a_x)$	0.04 m/s^2
	$\Phi(a_y)$	0.04 m/s^2
	$\Phi(a_z)$	0.08 m/s^2
	$\Phi(\omega_x)$	3 deg/s
	$\Phi(\omega_y)$	3 deg/s
	$\Phi(\omega_z)$	3 deg/s
Step 1	Wavelet function	Frequency-based
	Wavelet type	Complex
	Wavelet name	Morlet
	ω_0	4 rad/s
	Lag-1 autocorrelation α	0.9
	Sig. confidence level α	0.95
Step 2A	f_{min} of HF error	0.5 Hz
Step 2B	Δt_{min}	0.5 s
	PR_{min}	0.7
	PR_{max}	1.3
	f_{min} for $\rho^2_{\text{HF}}(t)$	2 Hz
	f_{max} for $\rho^2_{\text{HF}}(t)$	10 Hz
	$\rho^2_{\text{HF,min}}$	0.7
Step 4	f_{min} for $\rho^2_{\text{LF}}(t)$	0.05 Hz
	f_{max} for $\rho^2_{\text{LF}}(t)$	0.5 Hz
	$\rho^2_{\text{LF,min}}$	0.7
	f_{min} for $\Delta\phi(t)$	0.05 Hz
	f_{max} for $\Delta\phi(t)$	0.5 Hz
	$\Delta\phi_{\text{max}}$	16.4 deg
Post-process	$\Delta t_{\text{gap,max}}$	0.2 s



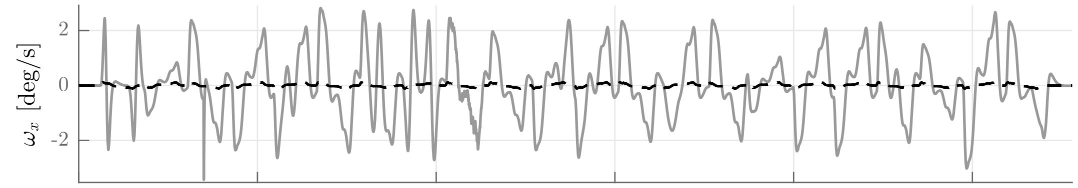
(a) Longitudinal specific force a_x (forward positive).



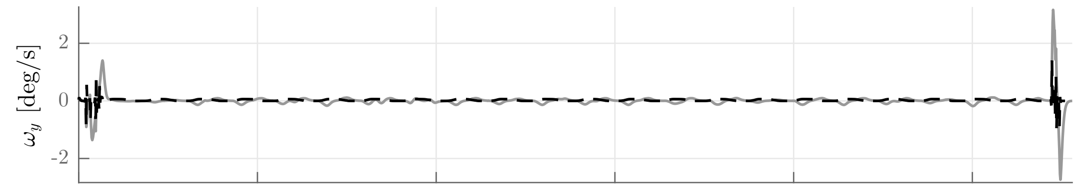
(b) Lateral specific force a_y (left positive).



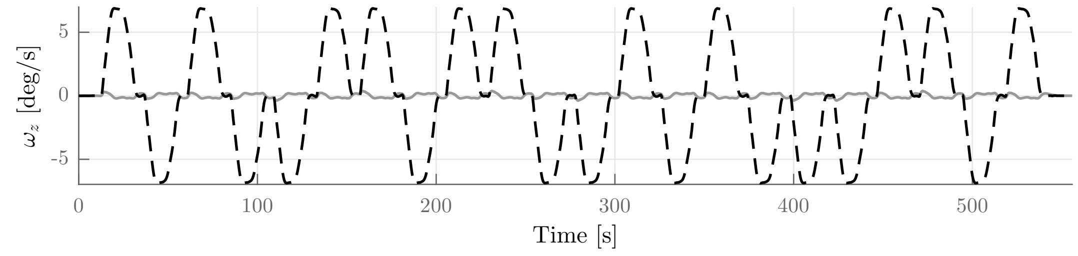
(c) Vertical specific force a_z (upwards positive).



(d) Roll rate ω_x.



(e) Pitch rate ω_y.



(f) Yaw rate ω_z.

Figure G-1: Inertial vehicle motion and MCA output for experiment run E1. Positive specific force directions have been given, rotational rates follow the right-hand rule.

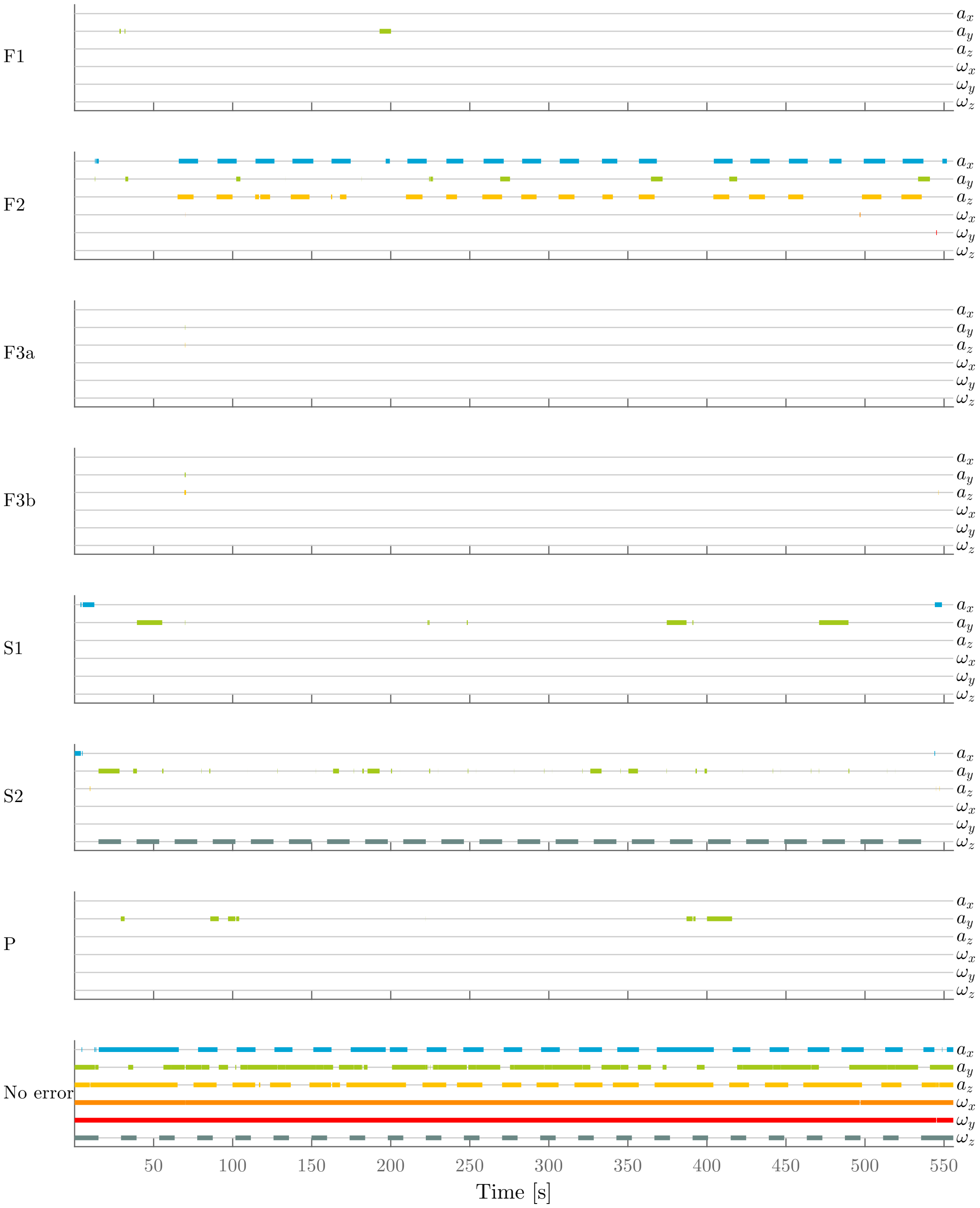


Figure G-2: Motion cueing error detection algorithm output for the vehicle inertial motion and algorithm output of experiment run E1.

Part II

Preliminary Thesis: Objective identification of motion cueing errors in motion simulation using wavelet analysis

Acronyms

CMS	CyberMotion Simulator
COI	Cone of Influence
CWT	Continuous Wavelet Transform
DOF	Degrees of Freedom
DOG	Derivative of a Gaussian
FR	Frequency Resolution
MCA	Motion Cueing Algorithm
MPI	Max Planck Institute for Biological Cybernetics
PR	Power Ratio
ROI	Region of Influence
SRS	Simona Research Simulator
TR	Time Resolution
TU Delft	Delft University of Technology

List of Symbols

Greek Symbols

α	Lag-1 red noise autocorrelation coefficient [-].
Δ	Difference operator.
ρ	Wavelet coherence [-].
σ	Signal standard deviation.
τ	Time shift of a wavelet [s].
Φ	Perceptual threshold for acceleration or rotation.
ϕ	Signal phase (difference) [°].
Ψ	Fourier transform of wavelet function.
ψ_M	Mother wavelet function.
ψ	Daughter wavelet function, a scaled and translated version of ψ_M.
ω	Rotational rate [°/s]
ω	Frequency [rad/s]
ω_0	Complex Morlet wavelet frequency at a scale of 1 [rad/s].

Roman Symbols

a	Acceleration [m/s^2].
$\mathcal{F}$	Fourier transform operator.
F	Fourier transform of time signal.
f	Time signal.
f_b	Complex morlet, Shannon and Frequency B-spline modulating wave decay parameter [-].

f_c	Complex morlet, Shannon and Frequency B-spline wavelet carrier wave frequency parameter [Hz].
$H(\omega)$	Heaviside step function.
i	Unit imaginary number.
k	Frequency index.
M	Sinc-function power for the Frequency B-spline wavelet.
m	Paul wavelet parameter.
N	Total number of data points [-].
n	Daubechies wavelet number.
n	Derivative order for a (complex) Gaussian wavelet.
n_{df}	Number of degrees of freedom.
P	Fourier power spectrum.
P	Period [s].
S	Scaling function in time and frequency.
s	Scale factor of a daughter wavelet [-].
t	Time [s].
$t_{0.025}$	Point in time before which 2.5% of wavelet energy is contained.
$t_{0.975}$	Point in time after which 2.5% of wavelet energy is contained.
W	Wavelet transform.

Chapter 1

Introduction

Ever since humans mastered the art of flying the importance of pilot training has been recognised, especially with the first aircraft being unstable in flight. Since in-flight training was dangerous in the early days of flight methods were quickly developed to train pilots on the ground, leading to the invention of the first flight simulators. Flight simulators have evolved together with aircraft ever since, with modern flight simulators being able to provide pilots with visual, vestibular, auditory and proprioceptive cues in a realistic environment.

Two examples of modern day high-end flight simulators are shown in Figure 1-1, the Simona Research Simulator (SRS) of the Delft University of Technology (TU Delft) and the CyberMotion Simulator (CMS) of the Max Planck Institute for Biological Cybernetics (MPI). The SRS is a typical example of a hexapod motion simulator, in which a cabin is placed on six hydraulic or electric actuators. Such hexapod simulators offer the possibility to provide high accelerations for big payloads and are inherently safe because the actuators function in a closed chain (Teufel et al., 2007). The CMS is an example of a serial robot simulator, in which the actuators are placed in series rather than parallel. This positioning offers the advantage of a bigger dexterity and a larger motion envelope (Robuffo Giordano et al., 2010).

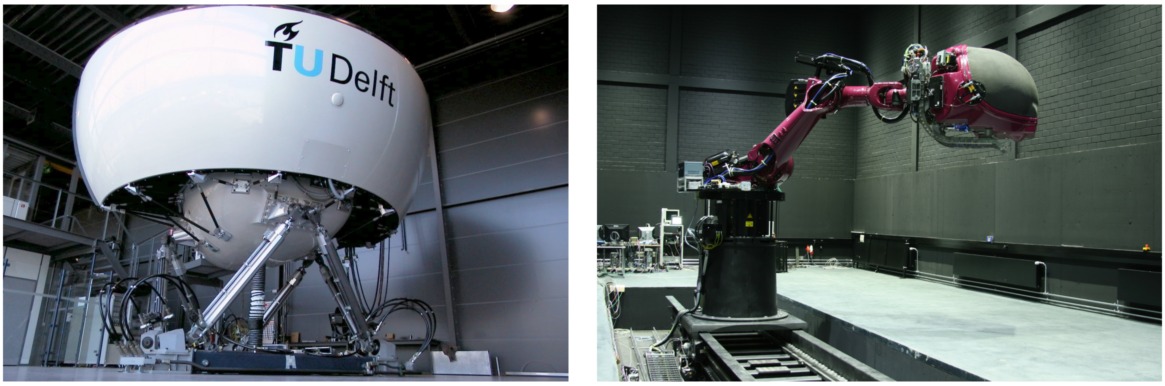


Figure 1-1: Two modern simulators, the Simona Simulator of the TU Delft (left) and the MPIs CyberMotion Simulator (right).

Using these simulators for training and research instead of real aircraft offers advantages through reduction of cost, pollution, noise and air traffic density (Rolfe & Caro, 1982; Baarspul, 1990, p18). Additionally simulators allow for (repeated) exact replication of certain situations and execution of only part of a task (Sanders, 1991, p1016), a feature which is beneficial for both training and research. To validate the use of simulators for these purposes various research has been done into the effectiveness of motion simulators, leading to the conclusion that motion simulators indeed lead to effective knowledge transfer in training (de Winter, Dodou & Mulder, 2012; Pool, Harder & van Paassen, 2016).

1-1 Problem statement

Because of their limited motion envelopes motion simulators are unable to exactly replicate large input vehicle motion. As a result discrepancies occur between the desired motion and the motion presented by the simulator (leading to a decrease in objective simulator fidelity) and the desired motion and the motion experienced by a simulator occupant (decreasing a simulator's perceptual fidelity) (Advisory Group for Aerospace Research & Development, 1980, p1). Understanding how such motion cueing errors influence perceptual (or perceived) simulator fidelity is key to improving motion simulators in the future.

To investigate this issue research is ongoing at the MPI. With simulator fidelity mostly defined as an overall concept, the time-varying nature of simulator fidelity has been investigated through the development of a Continuous Rating tool (Cleij et al., 2015). In continuation of Cleij's project attempts are now being made to predict Continuous Rating traces for a given simulation by analysing the magnitude of simulation errors being made. Preliminary results of this research look promising, however they appear to show that humans rate errors not solely on the magnitude of that error but also on the type of error. As such, predicting the perceived fidelity of a simulation can be improved by taking into account the effect of different types of motion simulation errors. Currently no algorithm exists however to objectively identify different types of motion cueing errors.

1-2 Project goal

The goal of this master thesis project is to design, test and evaluate an algorithm capable of detecting various types of motion cueing errors in a motion simulator input/output trace. In short the algorithm should make the conversion shown in Figure 1-2 (p127).

In this figure three different motion cueing errors (discrepancies between simulator input and output) are shown, all with equal magnitude (as indicated by the shaded areas). It can be argued however that all three errors belong to a different type. This distinction should be detected by the algorithm, showing what errors are present at what points in time. Such a detection would then allow for assigning different weights to these errors in the Continuous Rating reconstruction, leading to better prediction of human rating behaviour.

Since both the time of occurrence and frequency content of a motion cueing error are of interest (the latter allowing to comment on the type of error) the method of wavelet analysis is chosen as as method of analysis in this project.

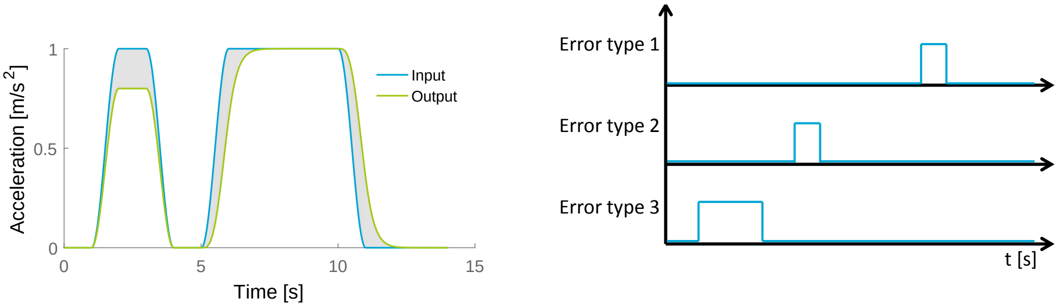


Figure 1-2: Visual representation of the goal of this master's thesis project. Different types of errors in motion simulator input/output signals should be detected by the algorithm.

1-3 Research questions

The project goal described in the previous section has been summarised in the following research question:

How to design an objective identification algorithm for both time- and frequency-localisation of different types of motion cueing errors to improve the knowledge on these errors in motion simulation?

This main research question has been divided into several subquestions, which can be answered one by one in answering the research question as a whole.

1. Can we come up with a set of mutually exclusive categories of motion cueing errors?
   - **1a.** What are possible causes of these categories of errors?
   - **1b.** What properties distinguish these error types?
   - **1c.** What mathematical technique(s) allow(s) for identifying these properties and differences?
2. How does the wavelet transform of a single signal work?
   - **2a.** Are there any prerequisites on the signal?
   - **2b.** What computations are necessary to perform a wavelet transform?
   - **2c.** What kind of output does one obtain following a wavelet transform?
   - **2d.** Does the wavelet transform have any limitations?
3. Which wavelet types are best suited for motion cueing error detection?
   - **3a.** What different types of wavelets are commonly used?
   - **3b.** What are the advantages of these types of wavelets?
4. How do you compare two wavelet-transformed signals?
   - **4a.** What options for comparison are available?
   - **4b.** Is there a specific type of data these methods are most suited to?
   - **4c.** What method is best suited for the comparison of motion simulator input and output?
5. How can we verify and validate that the algorithm functions correctly?

1-4 Report structure

The structure of this preliminary thesis is as follows. A general introduction into motion simulation is given in chapter 2, allowing for explanation of the various errors made (and their origins) in such a simulation. In this chapter a categorisation of motion cueing errors is then proposed, as such answering subquestion 1.

Chapter 3 then discusses the mathematical technique of wavelet analysis, the spatio-temporal analysis method which is to be used in this research project. Both the general idea of wavelet analysis and the options available within this analysis are discussed, answering subquestions 2 and 3. Chapter 4 then continues on this explanation to investigate how two time signals can be compared using wavelet analysis (providing answers to subquestion 4) and how the error properties discovered in chapter 2 can be identified from such an analysis.

Having then categorised motion cueing errors and their properties on the one hand and evaluated the possibilities wavelet analysis offers on the other hand, chapter 5 then brings these two fields of knowledge together by proposing an algorithm for motion cueing error detection which uses the possibilities of wavelet analysis to identify mathemathical error characteristics from simulator input and output signals. The algorithm is then tested by running simulations on idealised motion cueing errors.

This preliminary thesis report is then concluded in chapter 6, in which the results of this preliminary thesis are presented and discussed, an outlook to future work is given and a proposal for a simulator experiment is treated.

Chapter 2

Motion Cueing Errors

In this chapter a description and categorisation of various types of motion cueing errors will be given. To do so, the process of motion simulation will first be described in section 2-1. Although this process is strictly beyond the scope of this project (the algorithm to be developed simply takes an input and output signal without bothering how they came to be), the description in this section will make it easier to understand what sources of motion cueing errors can be distinguished.

Following this explanation of motion cueing a distinction between various error types will be proposed in section 2-2, with section 2-3 discussing the (mathematical) characteristics of these errors. These characteristics should then allow for error identification later on. The conclusions of this chapter with respect to the research questions introduced in chapter 1 are then summarised in section 2-4.

2-1 Motion cueing

As explained previously all motion simulators are limited by the amount of motion space available to them. In order to simulate vehicle motion a mapping is therefore needed from vehicle motion to simulator motion, this mapping is taken care of by a so-called Motion Cueing Algorithm (MCA). Various types of MCA exist, all of which exploit flaws in the human sensory system. One of the first MCA techniques is (classical) washout filtering, first introduced by Schmidt and Conrad and later well explained by Reid and Nahon (Schmidt & Conrad, 1970; Reid & Nahon, 1986). This technique will be used to illustrate the principles of motion cueing and to highlight sources of errors in motion simulators.

In washout filtering motion is split into two cues: a (high-frequency) onset cue and a (low-frequency) sustained cue. This split is illustrated in Figure 2-1, in which a step input in acceleration (in longitudinal direction in case of the example) is split up in its high- and low-frequency components. In order to keep the simulator within limits only the onset cues are presented to an occupant directly by translating the simulator, sustained accelerations are simulated by tilting the entire simulator in a process called tilt coordination. By tuning the

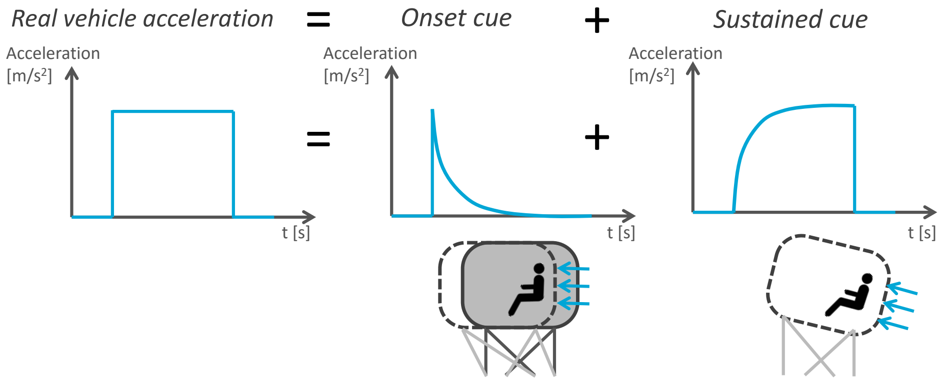


Figure 2-1: Illustration of the washout filtering process, in which motion is split into a high-frequency onset part and a low-frequency sustained part.

classical washout filter's parameters specific motion frequencies can be designated to either onset cueing or to tilt coordination, in this way one can assure a simulator will always remain within its operational limits.

Both washout (the slow return of a simulator cabin to its neutral position following an onset cue) and tilt coordination make use of the imperfections of the human sensory system. Washout exploits human perceptual thresholds by moving the simulator back to its neutral position using accelerations not perceivable to humans, as such ensuring the simulator retains its capacity to move in all directions. Tilt coordination on the other hand lets humans perceive an acceleration by rotating them at a sub-threshold angular velocity, leading them to mistakenly perceive a component of gravity as a longitudinal or lateral acceleration (Salisbury & Limebeer, 2016; Baarspul, 1990, p108). Using these two techniques enables vehicle motion to be replicated without exceeding a simulator's motion envelope, if the classical washout filter is well-tuned.

2-2 Error types

Irrespective of the tuning of a simulator the process of classical washout filtering (or any other MCA) is never able to exactly reproduce motion profiles from a certain motion intensity onwards (depending on the simulator motion envelope and the tilt coordination rate limiting used). As a result the motion perception induced by simulator motion will not fully correspond to the visual motion presented to a subject (which is a one-to-one copy of the vehicle motion), this mismatch will from now on be defined as the Motion Cueing Error.

Currently the quality of a simulation is often assessed by analysing the (squared) sum of these errors made in a simulation, applying a weighting factor for each motion channel (Naseri & Grant, 2005; Ko, 2012, p20). Such an approach assumes all motion cueing errors to be equally bad, as mentioned in the introduction however research suggests this is not the case. It is helpful therefore to define different types of motion cueing errors.

One such distinction was proposed by Fischer (Fischer, 2009, p52), who distinguished between scale errors (the difference between a simulator input and the scaled version of that input used by the MCA of that simulator) and shape errors (the difference between the scaled input and the simulator output). The term shape error had already been introduced by Grant, Blommer, Artz and Greenberg (Grant et al., 2003), who argued that *"the error due to scaling is to be considered separately from the shape error because the reduction in fidelity may be of a different order"*.

A limitation of this breakdown of error types is that the use of an MCA with constant scaling by a known scaling factor (such as a Classical Washout Filter) is assumed, allowing for separating scale errors from shape errors. This assumption of a constant scaling factor brings a limitation: novel motion cueing techniques such as the perception-based motion cueing (PBMC) algorithm developed at the Max Planck Institute for Biological Cybernetics make use of scaling factors varying over time (Venrooij et al., 2015). It is for this reason that the aforementioned division between shape and scale errors is not suitable for this project, since that would require prior knowledge of the MCA used and would exclude algorithms applying varying input scaling.

Another research project investigating the relation between error signal characteristics and perceived simulator fidelity was performed by Casas, Inmaculada, Riera and Fernández (Casas et al., 2015). Besides investigating the effect of error magnitude through the (Normalised) Average Absolute Difference and the effect of input scaling by determining the Average Absolute Scale, Casas et al. also investigated the influence of the correlation between simulator input and output. All these quantities however were computed over two input and output signals as a whole, providing no basis of identifying motion cueing error types in time. The mathematical characteristics identified in this research will be helpful for this project however.

A comprehensive overview showing a distinction between and definition of motion cueing error types was given by Grant and Reid (Grant & Reid, 1997), who distinguished three different error types: false cues, scaled/missing cues and phase errors. This categorisation of errors allows for identifying errors produced by any simulator and MCA without requiring prior knowledge, hence this error type distinction will serve as a basis for the error type distinction used in this report.

Grant and Reid propose a threefold definition of false cues, in this thesis we will let this lead to three types of false cues:

Type I false cue - Grant and Reid refer to the first type of false cue as *"a motion cue in the simulator that is in the opposite direction to that in the aircraft"* (naturally this can be extended to any vehicle). A typical cause for such an error is simulator washout (a simulator's return to its neutral position which requires an acceleration opposite to the acceleration being simulated) or a simulator braking near the end of its displacement envelope.

Type I false cues also arise at the (smooth) ending of a sustained rotational rate. Since rotational rates are often only high-passed a simulator's output near the end of a sustained rotational input will be close to zero, and then increase as the input decreases back to zero. An illustration of this effect for an acceleration trace is given in the top-left of Figure 5-5 (page 171).

Another cause of such an error might be the tilt-coordination process. When simulating a sustained forward acceleration a subject will be tilted backwards. In case the subject's head is centred above the centre of rotation of the simulator cabin this tilting will result in a backward head acceleration, leading to a Type I false cue.

Type II false cue - Grant and Reid describe a Type II false cue as *"a motion cue in the simulator when none was expected in the aircraft"*. Two typical examples of this stem from the tilt-coordination process applied by all motion cueing algorithms. One example shows in Figure 1-2 around $t = 10$s, where the relatively slow tilt coordination response causes the subject to perceive an acceleration even when the vehicle acceleration has dropped already. Grant and Reid refer to this cause as the tilt-coordination remnant.

The tilt-coordination process also introduces false tilt rates to simulate sustained accelerations (when simulating an aircraft take-off the subject will pitch backwards during the acceleration, a real aircraft doesn't pitch at the moment yet). Usually these tilt rates are kept at sub-threshold levels (such that a subject won't notice them), this is not always the case however. A final cause of a Type II false cue is referred to by Grant and Reid as G-tilt. In case an aircraft makes a banked (coordinated turn) the simulator cabin will be tilted to simulate the banking, this however introduces a sideways acceleration due to gravity which is not present in an aircraft performing a coordinated turn.

Type III false cue - This type of false cue is referred to by Grant and Reid as *"a relatively high-frequency distortion of a sustained cue in the simulator for an expected sustained cue in an aircraft"*. A typical cause for this error is a badly-tuned simulator hitting its safety/displacement limit, causing an abrupt stop of the simulator.

Another cause for Type III false cues are high-frequency simulator vibrations/noise as a result of the motion presented by a simulator. This especially is a problem for serial simulators such as the MPIs CMS, in which vibrations build up through all serial links. Hexapod simulators (which are actuated in parallel) suffer less from this phenomenon.

Scaling errors (in which the simulator presents a scaled-down version of the input motion at certain frequencies) are present in virtually all motion simulations, as in general all motion is scaled down in a simulator to limit simulator displacement. This scaling has been shown not to be detrimental for perceived simulator fidelity per se, possibly even improving perceived fidelity as humans tend to rate simulator motion as being 'too strong' in case one-to-one visual and motion cues are presented (Berthoz et al., 2013). Again two types of scaling errors are distinguished.

Type I scaling error - The first type of scaling error could be described as an ideal scaling error, in which the scaling constant applied to the simulator input signal is constant for all frequencies. An example of such an error is shown between $t = 2$s and $t = 4$s in Figure 1-2, showing a scaling factor of about 0.8.

Type II scaling error - Type II scaling errors are caused by signal attenuation which is not constant over all frequencies. A typical cause of such errors are the motion filters applied in Motion Cueing Algorithms: often a high-pass filter is used to simulate onset cues and a low-pass filter is used to regulate the tilt-coordination process. These filters are often not complementary however, meaning that mid-frequency signal content is often attenuated. Referring to Figure 1-2 such an error can be seen at $t = 6$s: even though both the actual vehicle and the simulator present motion the simulator response misses power in certain middle-frequency bands.

The final error type identified by Grant and Reid are phase errors, which we define as "simulator cues presented with a delay with respect to that same cue in the real vehicle". Possibly causes of such a delay are rate- or acceleration-limits of a simulator and simulator inertia (although the latter has less influence for properly-actuated simulators).

As a wrap-up all types of motion cueing errors are summarised in Table 2-1.

Error category	Error type	Description
False cue	Type I	Perceived simulator motion at a certain frequency opposite to the vehicle motion being simulated.
	Type II	Perceived simulator motion at a certain frequency when there is no (or far less) vehicle motion in that channel.
	Type III	A noticeable high-frequency disturbance in a sustained simulator cue during a sustained vehicle manoeuvre.
Scale error	Type I	Perceived motion in a sensory channel scaled at all frequencies with respect to the input signal, leading to similar signal behaviour.
	Type II	Perceived motion on a sensory channel scaled at a subset of frequencies with respect to the input signal, altering the signal behaviour.
Phase error		A simulator cue is presented with a perceivable delay with respect to that same cue in a real vehicle.

Table 2-1: Proposed definitions of categories and types of motion cueing errors, based on Grant and Reid (Grant & Reid, 1997).

Of these three error types false cues have proven to be most detrimental to perceived simulator fidelity (Grant & Reid, 1997; Reid & Nahon, 1987), partially explaining why most research focusses on the aforementioned shape errors and explaining why scaling errors and phase errors are discussed little in literature. Scaling errors occur most frequently as any simulator makes use of motion scaling. As a result these two error types are most important to distinguish in motion simulator input/output signals, hence distinguishing these five types of errors is the main goal of this thesis.

2-3 Error characteristics

In the previous section a distinction has been made between the various types of motion cueing errors. This section will highlight (mathematical) characteristics of these errors, which will help recognising these errors from given simulator input and output signals. Emphasis will be put on the duration, frequency and power ratio (defined as ratio of the output signal value and the input signal value over time) of the errors, as these characteristics will allow for discriminating the errors. Also the coherence between the input and output signal will be discussed. The first two error characteristics allow for time and frequency localisation of the errors, the latter two error/signal characteristics follow the work of Casas et al. (Casas et al., 2015).

Type I false cues - As mentioned previously a Type I false cue typically originates from simulator washout, the tilt coordination process or simulator braking. All of these processes take place at middle to low frequencies, as a result Type I false cues are expected at all but the very high frequencies. In case a Type I false cue is caused by the high-pass rotational filters then the error will also contain high-frequency power, with additional mid-frequency power caused by the damping out of the error.

False cues tend to last in the order of 1 second, depending on the tilt rate being applied in the tilt coordination process and the filter settings of an MCA. In case of a Type I false cue the input and output signals won't match too well (being of opposite sign), leading to poor coherence. The power ratio at the specific frequency of the Type I false cue will be negative by definition.

Type II false cues - As mentioned previously Type II false cues are mainly caused by tilt-coordination artefacts. The tilt-coordination process takes place at middle and low frequencies, Type II errors therefore are expected at these frequencies and last in the order of 1 second. Since the simulator presents motion which is not or hardly present in the input signal the coherence between the two signals is low, and the power ratio at the error frequency is (much) larger than 1 (since the output signal contains power whereas the input signal did not/contained little).

Type III false cues - Type III false cues stand out due to them only containing high-frequency content (much higher than common simulated motion). In case this high-frequency content is caused by the simulator hitting a displacement limit the signal power is usually very high, in case a Type III false cue is caused by simulator vibrations power will be relatively low. Displacement limit errors are also expected to last very short, whereas simulator vibrations are ever-present. For this reason a distinction will be made between these two origins of Type III false cues, to be referred to as Type III-1 and Type III-2 false cues from now on.

Since these high-frequency vibrations are not present in the input signal the correlation between simulator input and output will be low. For Type III-1 false cues the power ratio can take on any value, for Type III-2 errors the power ratio is expected to be close to 1 in case the simulator input is non-zero (for simulator input close to zero, the noisy output can cause the power ratio to take on any value).

Type I scaling errors - A Type I scaling error is characterised by the fact that the shape of the simulator output matches the input's shape but is scaled down. As a result the coherence between the two signals remains high and the power ratio at all frequencies is smaller than 1. Since Type I scaling errors usually arise from the scaling of the whole input signal they occur at all frequencies and last long.

Type II scaling errors - As illustrated by Figure 1-2 a typical Type II scaling error (or a missing cue) is the opposite of a Type II false cue. As a result the expected frequency at which the error occurs and duration for which it lasts are similar: all but the very high frequencies are expected (since Type II scaling errors also mostly originate from the tilt coordination process) at a relatively long duration. Again the coherence at the frequency of the error is low, since the input signal contains power the output signal does not. As a result the power ratio will be smaller than 1.

Phase errors - Finally phase errors are characterised by their phase difference, meaning that the simulator output signal lags behind the input signal at specific (or all) frequencies. Phase errors can occur at any frequency, though will be most noticeable and occur at similar durations as the errors mentioned previously (apart from the Type III false cues). In case of a pure phase error the coherence between two signals will remain high (as their frequency content is equal but slightly time-shifted), setting this error apart from most other errors. Like for a Type III-2 false cue the power ratio can take on any value, but will in general be close to 1.

To summarise, an overview of all mathematical properties of the different error types is given in Table 2-2 below. These properties will be exploited in chapter 5 in order to identify different motion cueing error types in a simulator input/output signal.

Error category	Type	Cause	Duration	Frequency	Coherence	Power ratio
False cue	Type I	All	Long	Mid-Low	Low	$P < 0$
	Type II	All	Long	Mid-Low	Low	$P > 1$
	Type III	Limit	Short	High	Low	
		Noise	Long	High	Low	$P \approx 1$
Scaling error	Type I	All	Long	All	High	$0 < P < 1$
	Type II	All	Long	Mid-Low	Low	$0 < P < 1$
Phase error		All	Long	All	High	$P \approx 1$

Table 2-2: Mathematical characteristics of the motion cueing error types defined in Table 2-1.

2-4 Chapter conclusions

In this chapter most of research question 1 (introduced in section 1-3 and discussing the categorisation of motion cueing errors) has been treated. The following answers have been provided:

1. *Can we come up with a set of mutually exclusive categories of motion cueing errors?*
   Three categories of motion cueing errors are distinguished: false cues, scaling errors and phase errors. Three further types of false cues are distinguished, along with two types of scaling errors.
   - **1a.** *What are possible causes of these categories of errors?*
     Most motion cueing errors are caused by overall input scaling applied by most MCAs, as well as the tilt-coordination process in motion simulation (which is the main solution motion cueing algorithms provide for simulating sustained accelerations). Errors are primarily caused by the limited tilt rate allowable for this solution (having to remain below perceptual threshold levels), leading to lag in the onset of cues (missing cues) and lag in returning to zero tilt (false cues).
   - **1b.** *What properties distinguish these error types?*
     The three categories of motion cueing errors (and their type subdivision) can be distinguished by their frequency content, duration, signal coherence and power ratio.

Chapter 3

Wavelet Analysis

As mentioned in chapter 1 the algorithm to be designed will make use of a spatio-temporal mathematical analysis technique. Multiple such techniques exist (such as the Short Time Fourier Transform and Windowed Fourier Transform), for this project however wavelet analysis is chosen as it is more efficient in making a time-frequency decomposition (Daubechies, 1992, p9). This chapter will detail on this technique as well as the options it provides.

First the reason for choosing wavelet analysis will be explained in section 3-1, with section 3-2 then explaining how a wavelet analysis is carried out. Available options are then explored in section 3-3 by providing an overview of the different wavelet types available as well as the effects of their parameters. Finally artefacts in wavelet analyses are discussed in section 3-4, with ways of dealing with these effects being proposed. Again chapter conclusions are summarised in section 3-5.

3-1 Choosing wavelet analysis

Signal analysis can be performed in both the time- and frequency domain. Both methods of analysis have their benefits and disadvantages, with time-domain plots allowing for precise localisation of signal features in time but offering little information on signal content. Frequency analysis methods (such as Fourier transforms) on the other hand are able to give insight into signal content but lose all time resolution. As a result, neither pure time- nor frequency based methods are able to comment on both a signal's frequency content (needed to comment on error type) and its instance of occurrence.

Spatio-temporal analysis methods combine the best of both worlds by splitting a signal out in both the time and frequency domain. Daubechies compares such a representation to music notation, in which a musician is told to play what tone (frequency) when (in time) (Daubechies, 1992, p1). Wavelet analysis allows for analysis of the frequency of a signal in a more extended way than for example the Fourier transform does. An illustration of this difference is given in Figure 3-1.

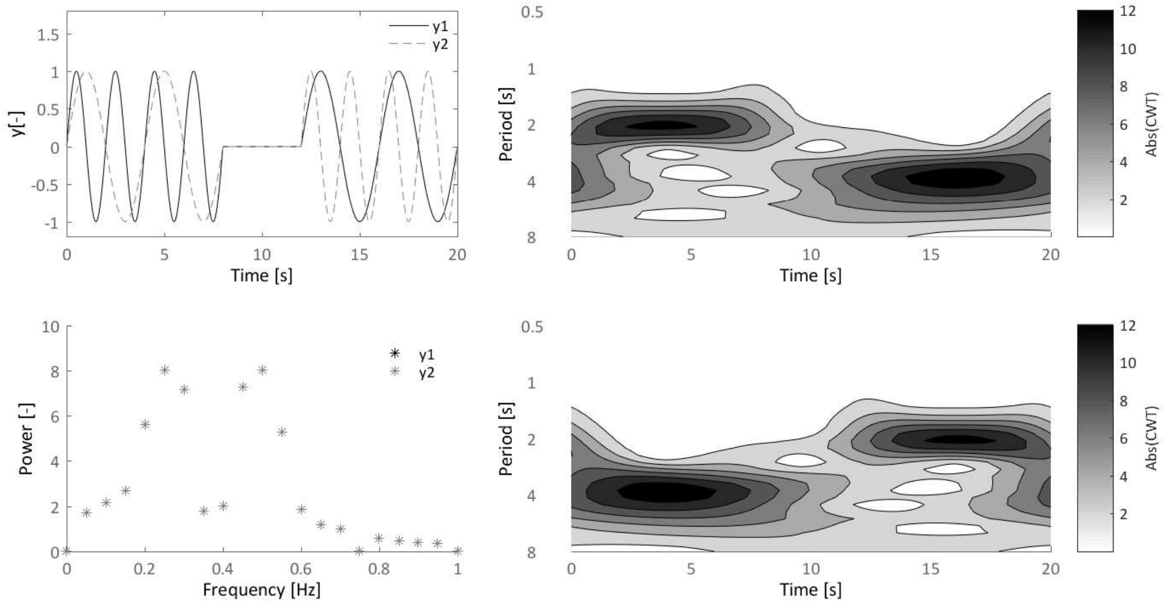


Figure 3-1: Illustration of the advantage of wavelet analysis over pure time/frequency domain analysis. The Fourier transforms of signals y1 and y2 are equal, but their wavelet transforms (right) show frequency power evolving over time.

Figure 3-1 shows two signals which are each other's mirror image in time. Two different periods can be recognised (2s and 4s), corresponding to frequencies of 0.5Hz and 0.25Hz respectively. The two signals' Fourier transforms (shown at the bottom left) are actually identical, as both signals contain equal power (albeit at different points in time). Since the Fourier transform loses all time resolution it is unable to tell when each frequency occurs.

This is where the advantage of the wavelet transform shows, as from the wavelet transform plots on the right we can clearly distinguish signal 1 (top right) to contain power at a period of 2 seconds first and then 'drop' to a period of 4 seconds. For signal 2 this is exactly the other way around, consistent with the visual observation from the top left image.

It is this ability to provide both time- and frequency-information of a signal that makes wavelet analysis an interesting technique for many signal analysis applications. As a result wavelet analysis has already been used in a host of research fields, including geophysics (Torrence & Compo, 1998; Cooper & Cowan, 2008; Jonah et al., 2015), biology (Valle-Levinson, Castro, Cáceres & Pizarro, 2014; Yochum, Renaud & Jacquir, 2016; Chang & Glover, 2010) and economics (Jammazi & Aloui, 2015; Kristoufek, 2015; Papaioannou, Dikaiakos, Evangelidis, Papaioannou & Georgiadis, 2015; Pei, Zhang, Li & Lee, 2015).

Most of these examples use wavelet analysis for investigating the evolution of a single signal's frequency content over time, such as the variation in sea temperature, heart rate or exchange rate. The current project differs from these studies in two ways: firstly the current study does not aim to investigate the content of a (single) signal but aims to comment on the content of the difference between two signals, and secondly the motion cueing signals in this study are not expected to be of periodic/oscillating nature (like sea temperatures, which vary per season, and exchange rates are). As a result the current project represents a new application of wavelet analysis in a new research domain.

3-2 Wavelet theory

Having established the benefits of the Continuous Wavelet Transform (CWT) in the previous section, the current section will explain the computations performed during a CWT. This explanation will be done in a graphical way in subsection 3-2-1, followed by a mathematical explanation in subsection 3-2-2.

3-2-1 Graphical explanation

The key idea behind wavelet analysis is that the similarity between an input signal and a wavelet with a specific frequency is determined at various points in time. By stretching or compressing the wavelet (and thus altering its characteristic frequency) different frequencies can be analysed, different points in time are evaluated by translating this wavelet.

A graphical explanation of this process is shown in Figure 3-2, in which the CWT of signal y1 in Figure 3-1 has been repeated. Highlighted in the top-left of this figure are three time-frequency points, each corresponding to one of the other three plots.

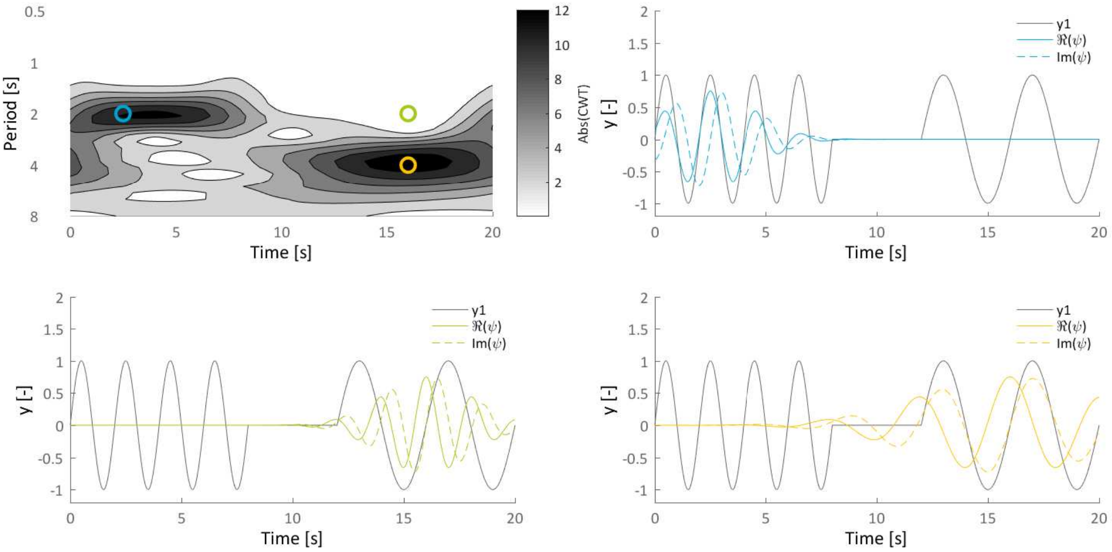


Figure 3-2: Illustration of the CWT, in which a signal is compared to a stretched and translated version of a wavelet at various locations in the time-frequency domain.

Starting with the blue dot, the top-right figure shows signal y1 together with a complex Morlet wavelet ($\omega_0 = 6$) translated by $\tau = 2.5$s and stretched by a factor $s = 2$. For this particular wavelet this scale corresponds to a period of roughly 2 seconds too. As is easily observed the period of signal y1 and the wavelet are nearly equal, meaning the inner product of the signal and the (real part of the) wavelet will result in a high value. As such the CWT-value at $t = 2.5$s and $P = 2$s becomes high.

Time-shifting the wavelet to $\tau = 16$s (as illustrated by the green dot and bottom left figure) results in two signals that don't align, therefore the inner product (with both the real and complex part of the wavelet) will now be low. For this reason the top left plot indicates little power at a period of 2s at $t = 16$s.

This changes when we also stretch the wavelet to a scale of 4 (bottom right figure), now the wavelet (with $P = 4$s) matches the period of the signal after $t = 12$s. As a result, the inner product of signal y1 with the complex part of the wavelet now becomes large, resulting in a high value when we plot the absolute value of the CWT (as we do in Figure 3-2).

This process of translating and scaling a wavelet and then computing its inner product with the signal is repeated for all scales of interest and all time points available, resulting in the construction of a CWT contour plot as shown in the top left of Figure 3-2.

This graphical explanation reveals a requirement placed on CWT input signal: Since most signals used for wavelet analysis are discrete (stemming from measurements for example) the inner product described will be executed in a discretised fashion as well. To allow for shifting of the wavelet this means that the signal evaluated has to be sampled with a constant time interval Δt. For simulator signals this should not be a problem, with most motion simulators sampling at a constant sample rate (and IMUs for example sampling at a constant rate too).

3-2-2 Mathematical explanation

In this subsection the mathematical details of the process explained graphically in the previous subsection will be gone through, based on the explanation given by Torrence & Compo (Torrence & Compo, 1998). As explained previously the wavelet transform of a signal is determined by computing its inner product with a wavelet at various scales at various points in time. Two conditions need to be satisfied for a signal to be suitable as a wavelet:

$$\int_{-\infty}^{\infty} \psi_M(t)\,\mathrm{d}t = 0 \tag{3-1}$$

$$\int_{-\infty}^{\infty} |\psi_M(t)|^2\,\mathrm{d}t = 1 \tag{3-2}$$

The first requirement states that a wavelet should have zero mean, the second requires a wavelet to oscillate around zero and go to zero for large values of $|t|$. This requirement to oscillate guarantees that all wavelets have a characteristic period, which allows for retrieving frequency information. From the mother wavelet a daughter wavelet is then formed by translating by τ and stretching by a scale factor s:

$$\psi(t,\tau,s) = \frac{1}{\sqrt{s}} \cdot \psi_M\left(\frac{t-\tau}{s}\right) \tag{3-3}$$

It can easily be verified that a daughter wavelet satisfies conditions (3-1) and (3-2) in case the mother wavelet ψ_M does so too. Using Equation (3-3) allows for computing the inner product of the analysed signal $f(t)$ and the daughter wavelet at any time shift τ and scale s:

$$W(\tau,s) = f(t) \cdot \psi^*(t,\tau,s) \tag{3-4}$$

In this equation $\psi^*(t, \tau, s)$ represents the complex conjugate of the wavelet function. Repeating this process for all τ the CWT of signal $f(t)$ for a given scale can be computed by convoluting $f(t)$ with the daughter wavelet:

$$W(s,t) = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} f(\tau) \cdot \psi_M{}^*(\frac{t-\tau}{s}) \mathrm{d}\tau \tag{3-5}$$

This convolution operation is computationally expensive and can be performed easily in the frequency domain in case a Fourier Transform of the wavelet function is available ($\mathcal{F}$ denotes the Fourier transform):

$$W(s,t) = \mathcal{F}^{-1}\left[\mathcal{F}(f(t)) \cdot \mathcal{F}(\psi^*(t,s)\right] = \mathcal{F}^{-1}\left[F(\omega) \cdot \Psi(s\omega)\right] \tag{3-6}$$

Evaluating Equation (3-6) at multiple scales allows for constructing a time-frequency wavelet power decomposition such as the one shown in Figure 3-1. Since the CWT is often complex (as multiple wavelets are complex) there are various ways of plotting the transform, in this thesis the absolute value of the CWT will be plotted unless stated otherwise.

From this mathematical description a number of limitations of the continuous wavelet transform become apparent. These are summed up below:

1. In case $f(t)$ is a constant, combining Equation (3-1) with Equation (3-4) leads to the conclusion that the CWT of a constant equals zero. This means that zero-frequency signal content can't be detected by the CWT, in fact when computing the CWT in the frequency domain a signal's average is first removed.
2. The use of the Fourier transform means that the maximum frequency to be analysed is determined by the Nyquist frequency, equal to half the signal sampling rate.
3. The minimum observable frequency (or maximum period) is determined by total observation time of a signal.
4. In case Equation (3-5) is used to determine the CWT the signal is padded with zeros in case the wavelet extends to outside the signal observation period. The same problem is encountered in Equation (3-6), in which the Discrete Fourier Transform assumes the analysed signal to be periodic. As a result results near the edges of the CWT can not be trusted, this artefact is further discussed in section 3-4.

3-3 Wavelet options

Within the mathematical process described in the previous section a wide variety of wavelets can be used, with both the wavelet type and wavelet parameters being variable. In current literature no complete overview of these wavelet types and parameters has been found, therefore an attempt is made to create such an overview in subsection 3-3-2. Before such an overview is given the topic of wavelet resolution has to be discussed, this is done in subsection 3-3-1.

3-3-1 Wavelet resolution

Just like discrete time signals have a finite time resolution and a signal's Fourier Transform only provides a certain frequency resolution (depending on the signal's sampling time), the CWT has both a time and frequency resolution. Both these properties are described below:

Time Resolution (TR) - A wavelet's time resolution indicates the ability of that wavelet to pinpoint signal features in time. Since the inner product of Equation (3-4) is influenced by all time points where a wavelet has power, a wavelet which is narrower in time will be less influenced by signal features 'far away'. As a result wavelets which are narrow in time will have better time resolution.

Frequency Resolution (FR) - A wavelet's frequency resolution indicates its ability to pinpoint the frequency content of a signal and discriminate between slightly higher/lower frequencies. The ability to identify a particular frequency in a signal increases when more information is taken into account in Equation (3-4), hence wavelets which are wide in time (and thus narrow in frequency) lead to better frequency resolution.

From these descriptions it is obvious that a trade-off between time and frequency resolution should always be made, the outcome of which depends on the aim of the analysis.

3-3-2 Wavelet overview

In this section an overview will be given of commonly used wavelets which are all available in Matlab's toolboxes. A distinction has to be made between wavelets which are implemented in the frequency domain using Matlab's `cwtft` command (the (complex) Morlet, Paul and DOG wavelets) and those implemented in the time domain using Matlab's `cwt` command (the (complex) Morlet, Gaussian, complex Gaussian, Shannon, Frequency B-spline, Haar and Daubechies wavelets). Wavelets implemented in the time domain are discussed first, followed by wavelets implemented in the frequency domain.

For all wavelet families discussed Continuous Wavelet Transforms have been performed on a test signal. This test signal features multiple characteristics such as signal peaks, multiple constant-frequency sines, signal discontinuities and a frequency sweep (or chirp) signal. By using various parameter values the effect of these parameters can be demonstrated. All figures can be found in Appendix A, pages 186-190. It should be noted that these plots are shown with an inversed y-axis, as will be standard from now on. This way of plotting ensures that high-frequency peaks in the CWT plot will point directly to events in time in the plots above. The white shaded areas indicated the Regions of Influence, which are discussed in subsection 3-4-1.

Complex Morlet - The Morlet wavelet is one of the most commonly used wavelet types. The Morlet wavelet is a wave modulated by a Gaussian (Torrence & Compo, 1998), as shown in Figure 3-3. In the time domain two parameters tune this wavelet, the modulating Gaussian decay parameter $f_b > 0$ and the carrier wave frequency $f_c > 0$. Increasing f_b leads to a slower decay of the Gaussian, broadening the wavelet in time and hence increasing frequency resolution and decreasing time resolution. Increasing f_c increases the carrier wave frequency (and hence lowers the characteristic period), meaning more time information is now available for analysing this frequency. As a result the frequency resolution increases and time resolution decreases. These two findings can be verified in Figure A-2, in which the spread in power over frequencies decreases as f_b and f_c are increased. Table 3-1 summarises these findings.

Complex Morlet wavelet	
Function in time	$\psi_M(t) = (f_b \cdot \pi)^{-1/2} \cdot e^{2\pi \cdot i \cdot f_c \cdot t} \cdot e^{-t^2/f_b}$
Function in frequency	$\psi_M(s\omega) = e^{-\pi^2 f_b (s\omega - f_c)^2}$
Modulating wave	$\pm$ C$\cdot e^{-t^2/f_b}$
Parameter effects	$f_b{}^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $f_b{}^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$
	$f_c{}^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $f_c{}^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$

Table 3-1: Details of the time-based complex Morlet wavelet (Teolis, 1999, p66).

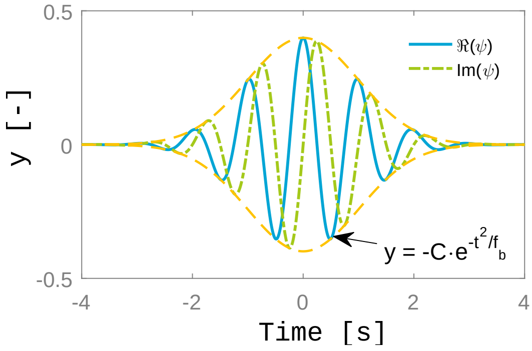


Figure 3-3: The time-based complex Morlet wavelet (f_b = 2, f_c = 1 Hz) with its modulating wave highlighted.

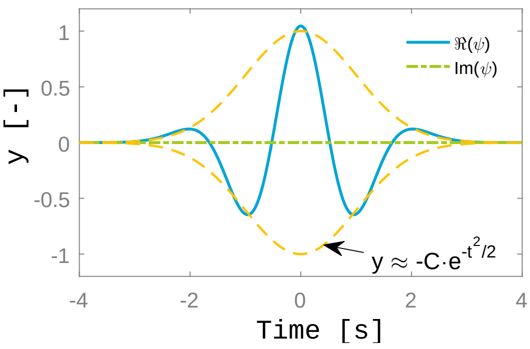


Figure 3-4: The time-based DOG wavelet (n = 4) with its modulating wave highlighted.

Gaussian - Gaussian (or DOG) wavelets such as the one shown in Figure 3-4 are based on derivatives of the Gaussian function (any derivative between first and eighth order is accepted in Matlab). As opposed to the complex Morlet wavelet this wavelet is real, meaning that it is unable to continuously detect signal power (instead the inner product will oscillate between positive and negative values as the DOG traces a sine). This effect can also be seen in Figure A-2, in which both the sine wave and frequency sweep show up as intermittent peaks. Increasing the derivative order n (an integer) increases the number of sign changes and hence the wavelet frequency. Again this leads to more time information used to evaluate the inner product at higher frequencies, leading to better frequency resolution and decreasing time resolution.

Gaussian wavelet	
Function in time	$\psi_M(t) = \frac{\mathrm{d}^n}{\mathrm{d}t^n}\left(\mathrm{C} \cdot e^{-t^2}\right)$
Function in frequency	–
Modulating wave	$\approx \pm\ \mathrm{C}{\cdot}e^{-t^2/2}$
Parameter effects	$n^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $n^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$

Table 3-2: Details of the time-based Gaussian wavelet (constant C ensures unit energy).

Complex Gaussian - The complex Gaussian wavelet is similar to the DOG wavelet, however for this wavelet the DOG wavelet is used as a modulating function of a complex exponential (see Figure 3-5). The order parameter has a similar effect to that of the Gaussian wavelet: increasing the integer n increases the frequency resolution and decreases the time resolution. Since this wavelet is complex it is able to continuously indicate power and indicate the phase of a signal.

Complex Gaussian wavelet	
Function in time	$\psi_M(t) = \frac{\mathrm{d}^n}{\mathrm{d}t^n}\left(\mathrm{C} \cdot e^{-it} \cdot e^{-t^2}\right)$
Function in frequency	–
Modulating wave	$\approx \pm\ \mathrm{C}{\cdot}e^{-t^2/2}$
Parameter effects	$n^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}\ \&\ n^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$

Table 3-3: Details of the time-based complex Gaussian wavelet (constant C ensures unit energy).

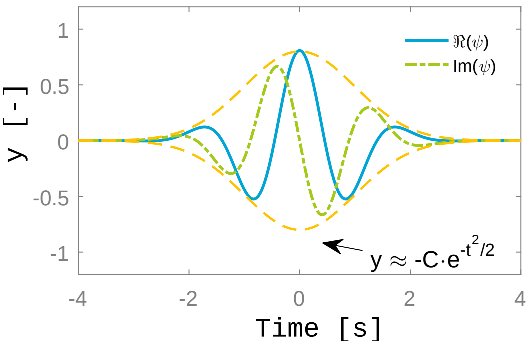


Figure 3-5: The time-based complex Gaussian wavelet ($n = 4$) with its modulating wave highlighted.

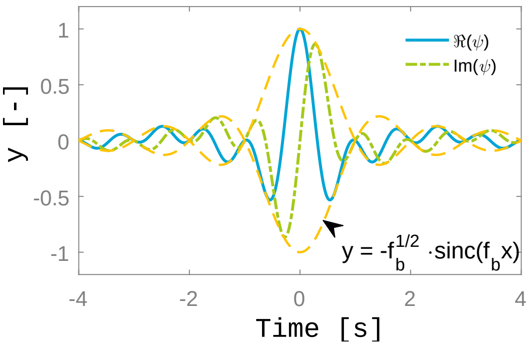


Figure 3-6: The time-based Shannon wavelet ($f_b = 1$ Hz, $f_c = 0.8$ Hz) with its modulating wave highlighted.

Shannon wavelet - A fourth class of time-based wavelets is the Shannon wavelet, which consists of a plane wave modulated by a sinc-function, as shown in Figure 3-6. Two parameters can be used to tune this wavelet: $f_b > 0$ determines the frequency of the sinc-function and $f_c > 0$ determines the carrier wave frequency and hence the period of the wavelet.

Similar to the complex morlet wavelet, increasing f_c increases the wavelet frequency (and hence decreases the characteristic period) whilst keeping the wavelet equally broad in time. As a result the frequency resolution increases whilst the wavelet's time resolution decreases. Increasing f_b narrows the modulating sinc function in time, as such increasing time resolution and decreasing frequency resolution. This means that increasing both f_b and f_c by an equal factor leaves the time and frequency resolution of the Shannon wavelet unchanged, however it does influence the signal frequency range which can be analysed.

Shannon wavelet	
Function in time	$\psi_M(t) = \sqrt{f_b} \cdot e^{2\pi \cdot i \cdot f_c \cdot t} \cdot sinc(f_b t)$
Function in frequency	–
Modulating wave	$\pm 1/\sqrt{f_b} \cdot sinc(f_b t)$
Parameter effects	$f_b{}^{\uparrow} \rightarrow \mathrm{FR}^{\downarrow}$ & $f_b{}^{\uparrow} \rightarrow \mathrm{TR}^{\uparrow}$ $f_c{}^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $f_c{}^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$

Table 3-4: Details of the time-based complex Shannon wavelet (Teolis, 1999, p64).

Frequency B-spline wavelet - The frequency B-spline wavelet in Figure 3-7 is similar to the Shannon wavelet, however in this wavelet the sinc-function is raised to the power M (an integer ≥ 1). As a result, for $M = 1$ the Shannon and Frequency B-spline wavelet are equal and the effect of parameters f_b and f_c is similar to that described for the Shannon wavelet parameters. From $M = 2$ onwards increasing M broadens the modulating sinc-function in time, hence increasing the frequency resolution and decreasing time wavelet's time resolution.

Frequency B-spline wavelet	
Function in time	$\psi_M(t) = \sqrt{f_b} \cdot e^{2\pi \cdot i \cdot f_c \cdot t} \cdot \left[sinc(\frac{f_b t}{M}) \right]^M$
Function in frequency	–
Modulating wave	$\pm 1/\sqrt{f_b} \cdot \left[sinc(\frac{f_b t}{M}) \right]^M$
Parameter effects	$f_b{}^{\uparrow} \rightarrow \mathrm{FR}^{\downarrow}$ & $f_b{}^{\uparrow} \rightarrow \mathrm{TR}^{\uparrow}$ $f_c{}^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $f_c{}^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$ $M^{\uparrow} \rightarrow \mathrm{FR}^{\downarrow}$ & $M^{\uparrow} \rightarrow \mathrm{TR}^{\uparrow}$

Table 3-5: Details of the time-based complex Frequency B-spline wavelet (Teolis, 1999, p64).

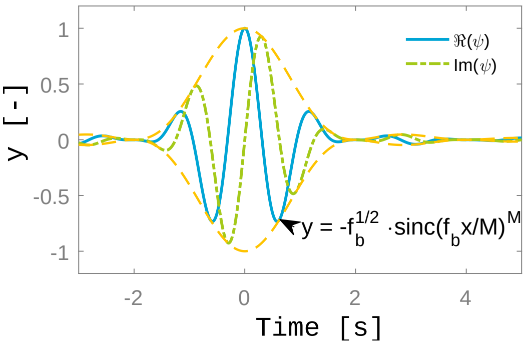


Figure 3-7: The Frequency B-spline wavelet ($M = 2$, $f_b = 1$ Hz, $f_c = 0.8$ Hz) with its modulating wave highlighted.

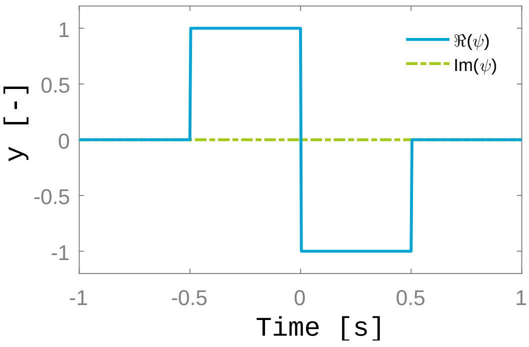


Figure 3-8: The time-based Haar wavelet.

Haar wavelet - The real-valued Haar wavelet, shown in Figure 3-8, is one of the simplest wavelets and one of the first to be developed. Taking on a value of -1 before its centre in time and 1 after, the Haar wavelet essentially monitors signal changes. Its quick changes in time mean that the Haar wavelet is able to provide excellent time resolution, with its frequency resolution being poor as the Haar wavelet is broad in frequency (Teolis, 1999).

Haar wavelet	
Function in time	$\psi_M(t) = 1_{(-\frac{1}{2},0]} - 1_{(0,\frac{1}{2}]}$
Function in frequency	$\Psi(\omega) = 2i \cdot \frac{\sin^2(\frac{\pi\omega}{2})}{\pi\omega}$
Modulating wave	–
Parameter effects	–

Table 3-6: Details of the time-based Haar wavelet (Teolis, 1999, p62).

Daubechies wavelet - The final time-based family of wavelets to be discussed is the Daubechies family, shown in Figure 3-9. This wavelet family was developed and described by Daubechies in her "Ten lectures on Wavelets", in which mathematical details are given as well (Daubechies, 1992, p195-199). A special member of this family is the 'db1' wavelet, which is actually just the Haar wavelet discussed previously. From Figure 3-9 one can see that increasing the Daubechies number n stretches the wavelet in time whilst keeping the characteristic frequency almost constant. As a result this increases the frequency resolution and decreases the time resolution.

Daubechies wavelet	
Function in time	–
Function in frequency	–
Modulating wave	–
Parameter effects	$n^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $n^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$

Table 3-7: Details of the time-based Daubechies wavelet.

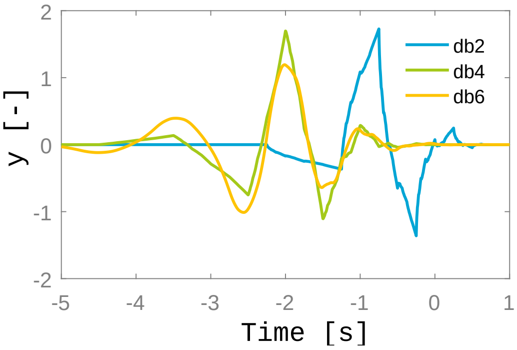


Figure 3-9: Three Daubechies wavelets (db2, db4 and db6).

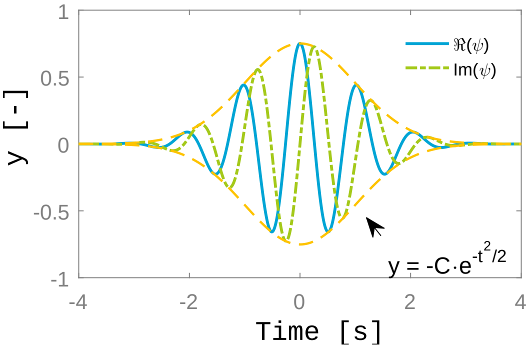


Figure 3-10: The frequency-based Morlet wavelet (ω_0 = 6 rad/s), with its modulating wave highlighted.

The Daubechies family is the final time-based wavelet family to be discussed. Next to the time-based implementation of the CWT one can also make use of the frequency-implementation of the CWT, which requires a Fourier transform of the wavelet to be available. Three frequency-based wavelets are discussed below:

Complex Morlet wavelet - The complex Morlet wavelet can also be implemented in the frequency domain in Matlab's `cwtft` function. This frequency-implemented complex Morlet wavelet is most commonly used in wavelet analysis so far, as it provides a good balance between time and frequency resolution (Teolis, 1999). As opposed to its time-based counterpart the frequency-based Morlet wavelet has only one parameter, $\omega_0 > 0$, which alters the carrier wave frequency in Figure 3-10. Like parameter f_c in the time-domain increasing ω_0 leads to better frequecy resolution and poorer time resolution.

Frequency-based Morlet wavelet	
Function in time	$\psi_M(t) = \pi^{1/4} \cdot e^{i \cdot \omega_0 \cdot t} \cdot e^{-t^2/2}$
Function in frequency	$\Psi(s\omega) = \pi^{1/4} \cdot H(\omega) \cdot e^{-(s\omega - \omega_0)^2/2}$
Modulating wave	$\pm \ \mathrm{C} \cdot e^{-t^2/2}$
Parameter effects	$\omega_0{}^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $\omega_0{}^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$

Table 3-8: Details of the frequency-based Morlet wavelet (Torrence & Compo, 1998, p65).

Paul wavelet - A second complex frequency-implemented wavelet is the Paul wavelet, illustrated in Figure 3-11. Unlike previous complex wavelets the Paul wavelet does not consist of a separate modulating and carrier wave, instead combining the two in one function. The Paul wavelet parameter m (an integer between 2 and 7) determines the number of zero crossings of the Paul wavelet, hence increasing m increases the wavelet characteristic frequency and thus the frequency resolution. Increasing parameter m also slightly narrows the modulating wave in time (decreasing frequency resolution), however this effect is smaller than the effect mentioned previously.

Frequency-based Paul wavelet	
Function in time	$\psi_M(t) = \frac{2^m \cdot (1i)^m \cdot (m)!}{\sqrt{\pi \cdot (2m)!}} \cdot (1 - it)^{-(m+1)}$
Function in frequency	$\Psi(s\omega) = \frac{2^m}{\sqrt{m \cdot (2m-1)!}} H(\omega)(s\omega)^m e^{-sm}$
Modulating wave	$\pm \ \mathrm{C} \cdot \left\lvert (1 - it)^{-(m+1)} \right\rvert$
Parameter effects	$m^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $m^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$

Table 3-9: Details of the frequency-based Paul wavelet (Torrence & Compo, 1998, p65).

Derivative of a Gaussian (DOG) wavelet - The final frequency-based wavelet is the real-valued DOG wavelet shown in Figure 3-12. This wavelet is almost equal to the time-domain based version discussed earlier, the only difference being that the exponential in the time description is now divided by a factor 2. As a result this wavelet is slightly wider in time. Its properties are summed up in Table 3-10.

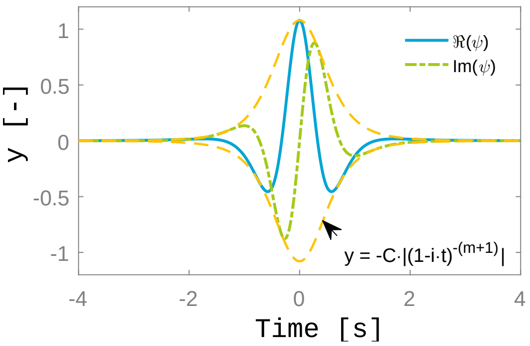


Figure 3-11: The frequency-based Paul wavelet ($m = 4$), with its modulating wave highlighted.

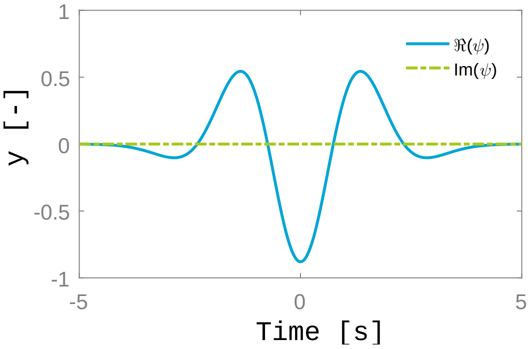


Figure 3-12: The frequency-based DOG wavelet ($n = 4$).

Frequency-based DOG wavelet	
Function in time	$\psi_M(t) = \frac{(-1)^{n+1}}{\sqrt{\Gamma(n+\frac{1}{2})}} \frac{\mathrm{d}^n}{\mathrm{d}t^n} \left(e^{-t^2/2}\right)$
Function in frequency	$\Psi(s\omega) = \frac{(-i)^n}{\sqrt{\Gamma(n+\frac{1}{2})}} \cdot (s\omega)^n \cdot e^{-(s\omega)^2/2}$
Modulating wave	–
Parameter effects	$n^{\uparrow} \rightarrow \mathrm{FR}^{\uparrow}$ & $n^{\uparrow} \rightarrow \mathrm{TR}^{\downarrow}$

Table 3-10: Details of the frequency-based DOG wavelet (Torrence & Compo, 1998, p. 65).

3-4 Wavelet artefacts

In the previous sections the process of wavelet analysis has been discussed, all commonly used wavelet types have been introduced and the effect of their parameters on the time and frequency resolution of an analysis has been highlighted. During wavelet analysis two artefacts are introduced in the CWT which both have to do with the broad time basis of wavelets. These two artefacts are discussed in subsection 3-4-1 and subsection 3-4-2, with solutions to deal with these artefacts being proposed.

3-4-1 Edge effects

As can be seen in Figure 3-2 the determination of the inner product of Equation (3-4) at a specific time t_1 takes into account signal values around t_1 as well in order to detect frequency characteristics. The size of this time interval increases with the characteristic period of the wavelet used.

This effect is illustrated in Figure 3-13, in which the time-period domain for a hypothetical signal with a length of 10 seconds is shown. The blue cone (to be referred to as a Cone of Influence (COI) from now on) illustrates the time interval taken into account to determine any CWT value at $t = 3$s (the left grey line) using a Shannon wavelet with $f_b = 0.5$ and $f_c = 1$. Note that this definition of a COI is different from the definition used by Torrence & Compo (Torrence & Compo, 1998), this difference will be addressed later.

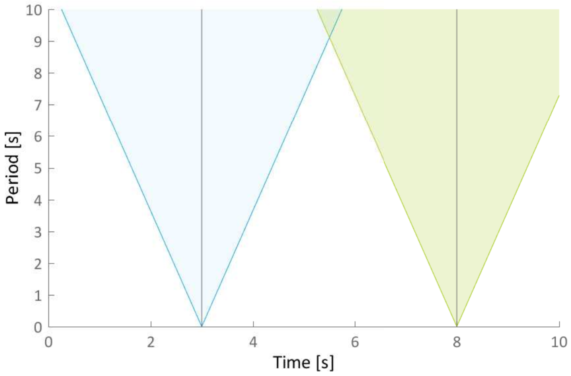


Figure 3-13: Illustration of the Cone of Influence at two different points in time. At $t = 8$s low-frequency data can't be trusted anymore.

As can be seen from the blue COI the CWT values at $t = 3$s are all determined based on points in time at which the hypothetical signal is defined ($0\text{s} \leq t \leq 10\text{s}$). As a result, all CWT values at $t = 3$s can be considered correct.

As noted in subsection 3-2-2 problems occur near the edges of the time-frequency domain, as illustrated by the green COI. As is easily seen the CWT values at a period of more than 7 seconds are partially based on time values outside the signal domain, which consist of zeros (in case of time-domain analysis) or a repeat of the signal from $t = 0$s onwards (in case of frequency-domain analysis, which assumes periodicity). Near the edges of the CWT plot therefore a region is found where the CWT values should be treated with caution.

Torrence & Compo name this region the Cone of Influence, in this thesis this region will be referred to as the Region of Influence (ROI). Torrence & Compo define this region as the *"e-folding time for the autocorrelation of wavelet power* ($|\text{CWT}|^2$) *at each scale"* (Torrence & Compo, 1998), which is equal to *"the area in which the wavelet power caused by a discontinuity at the edge has dropped to* e^{-2} *of the value at the edge"* (Grinsted, Moore & Jevrejeva, 2004). Torrence & Compo offer expressions for the width of this ROI as a function of scale for all frequency-based wavelets but not for the time-based wavelet families. Besides this incompleteness there are two further problems with the definition given by Torrence & Compo:

1. By definition the inner product of any symmetric real wavelet (such as the Gaussian wavelet for even n) centred at a discontinuity is zero (see Equation (3-1)), meaning that the width of the ROI can't be determined for such wavelets.
2. Torrence & Compo give a constant ROI width for the complex Paul wavelet, indepent of its parameter m. Subsection 3-3-2 however showed that the width in time of the Paul wavelet changes with m, meaning that the ROI width should vary with m.

In order to define a ROI for each type of wavelet a different approach is suggested in this thesis. For each wavelet the two points in time between which 95% of wavelet energy is contained (recall that the total wavelet energy is 1 by definition) are determined and are designated $t_{0.025}$ and $t_{0.975}$. The boundaries of the cones in Figure 3-13 are then $t + t_{0.025}$ (assuming $t_{0.025}$ is negative) and $t + t_{0.975}$. Now we define the ROI to be limited by those (time,Period)-points where the lower/upper bound of this interval matches the lower/upper bound of the signal interval. Shading the ROI when plotting the results of a CWT allows for neglecting areas in which edge effects may have influenced results.

Using this approach produces results very similar to the $\sqrt{2}s$ and $s/\sqrt{2}$ values found by Torrence for the frequency-based complex Morlet and Paul wavelets respectively and enables the approach to be extended to any wavelet. A ROI is shown with all CWT-plots shown in Figures A-1 through A-5 in Appendix A, allowing for investigating the effect of different wavelets and parameters on the size of the ROI. It can be seen for example that the Shannon and Frequency B-spline wavelets have a very small ROI, with the Morlet wavelet having a relatively large ROI. In general any parameter change that results in a higher frequency resolution (which essentially means it makes the signal broader in time) will also increase the size of the ROI.

3-4-2 Power spread

The limited time and frequency resolution of a wavelet introduce a second artefact in the results of a wavelet transform: spread in wavelet power over both time and frequency. This effect shows in Figure 3-1: although both signal parts contain only one frequency (0.5Hz and 0.25Hz respectively) and start and end abruptly the wavelet power spreads over multiple frequencies and over time.

In order to discriminate between parts of the CWT containing genuine power and parts of the CWT influenced by this signal spread Torrence & Compo propose a significance check on the wavelet spectrum (Torrence & Compo, 1998). The null hypothesis of this check is that the signal analysed originates from a white or red noise process, with a data point in this noise being given by:

$$x_n = \alpha \cdot x_{n-1} + z_n \tag{3-7}$$

In Equation (3-7) α is the red noise lag-1 autocorrelation coefficient, which indicates to what extent the previous datapoint (x_{n-1}) is copied to the next (x_n). Added to this is a sample from Gaussian white noise (z_n), meaning that if α is set to zero Equation (3-7) describes a white noise process. For such a white or red noise process, the normalised Fourier power spectrum is given by (Gilman, Fuglister & Mitchell, 1962):

$$P_k = \frac{\sigma^2(1-\alpha^2)}{1+\alpha^2-2\alpha\cos(2\pi k/N)} \tag{3-8}$$

In this equation σ^2 is the time signal variance and k is the frequency index used in the Fast Fourier Transform of the input signal, with N the total number of data points. Using this spectrum as a background allows for marking pieces of the CWT as significant in case CWT power (or $|\text{CWT}|^2$) in a time-slice of this CWT is sufficiently above this background spectrum, with the "sufficient" factor being determined by the confidence value chosen (in this report a 95% confidence value will be used). The reader is referred to Torrence & Compo for a discussion showing that this confidence value can be obtained from a χ^2 distribution with either two Degrees of Freedom (DOF) (for complex wavelets) or one DOF (for real wavelets). As a result, CWT-spectra regions are significant in case (n_{df} denotes the number of degrees of freedom):

$$|CWT(f(t))|^2 \geq \frac{1}{n_{\text{df}}} P \cdot \chi^2_{n_{\text{df}}}(0.95) \tag{3-9}$$

To show the effect of this significance check, the signal used in Figure 3-1 was copied and analysed using a lag-1 autocorrelation coefficient of 0.95 and two degrees of freedom (since the complex Morlet ($\omega_0 = 6$) was used to perform the wavelet analysis). Statistically significant regions are indicated using the yellow contour in Figure 3-14 (note that the discontinuous contours are caused by the discrete frequency scales).

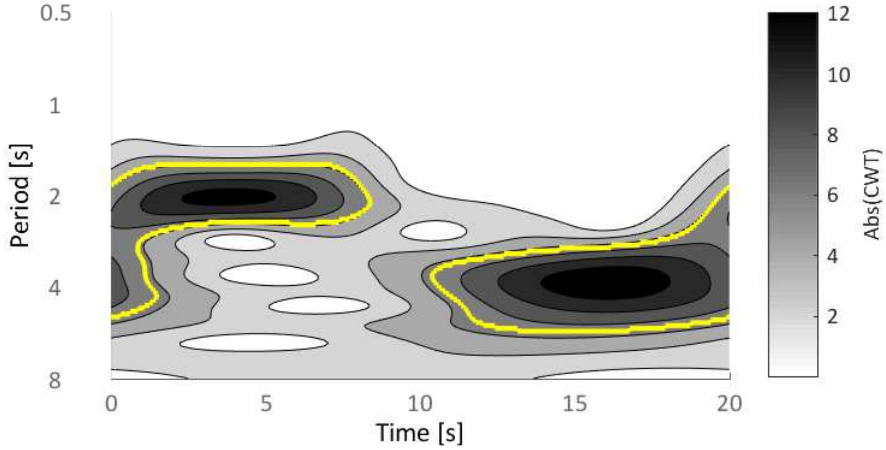


Figure 3-14: Illustration of the wavelet transform of signal y1 in Figure 3-1, with significant regions indicated.

As can be seen from Figure 3-14 the regions of significance enclose the true significance regions (at $P = 2$ between t=0s and $t = 8$s, and at $P = 4$ between t=12s and $t = 20$s) and indicate other regions where power was detected as not significant. Also note that the false low-frequency content identified at $t < 2$s and the false high-frequency content identified for $t > 19$s would be excluded by the Region of Influence.

For signals consisting of only periodic segments (such as the example signals in Figure 3-1) detecting significant power is relatively easy, more difficult are cases in which the signal does not show periodic behaviour (as explained earlier, most simulator input/output and error signals do not show this behaviour). An example of a typical motion simulator error signal (which is discussed later) is shown in the top left of Figure 3-15. This error has been scaled to ensure the variance of this signal is 1, simplifying Equation (3-8).

As can be seen from Figure 3-15 the error only contains power between $t = 4$s and $t = 7$s, the CWT spectrum below however shows the power being spread over the entire time

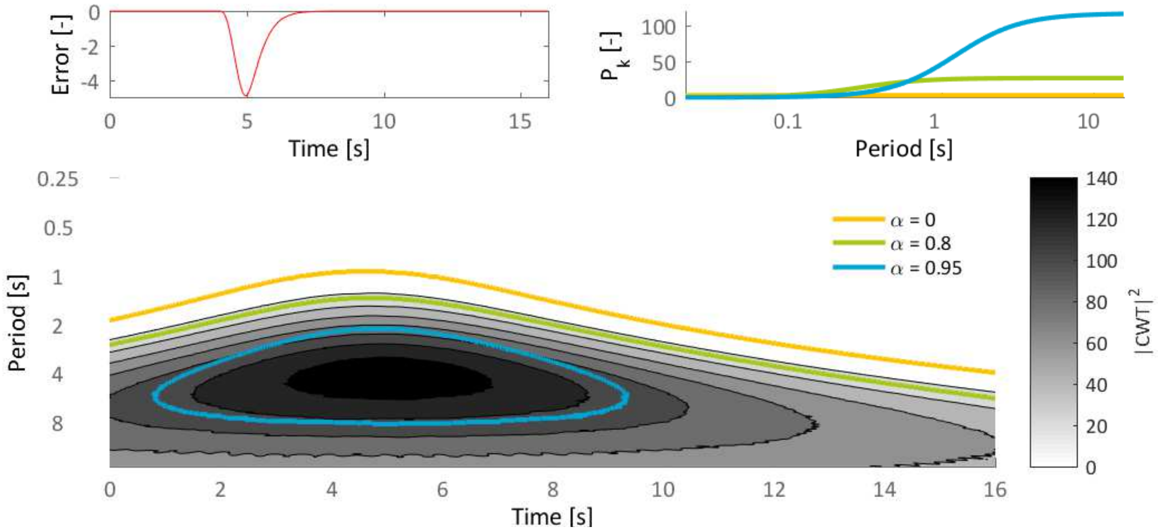


Figure 3-15: Illustration of the effect of the lag-1 autocorrelation coefficient on the CWT of a typical motion cueing error signal.

window. Correctly identifying the genuine region containing power in this case requires the right selection of the lag-1 autocorrelation coefficient α.

The effect of choosing three different lag-1 autocorrelation coefficients ($\alpha = 0$ (effectively assuming a white noise background), $\alpha = 0.8$ and $\alpha = 0.95$) is also shown in Figure 3-15. Shown in the top right are the three theoretical Fourier power spectra for these three lag-1 coefficients (including the $\chi^2(0.95)$ multiplication factor, which is around 3). As seen from this figure increasing the lag-1 autocorrelation coefficient increases the significance threshold for lower frequencies and slightly decreases this threshold for higher frequencies. This results in the fact that the area deemed to be significant (which is the area in which a vertical slice through this CWT spectrum is higher than the background spectrum) decreases for a higher α. Note that an even higher α-value could have been chosen to decrease the significant area even further, however that would have made the lines in the top right plot indistinguishable.

For the simulator signals considered in this project such a high lag-1 autocorrelation coefficient is a logical choice, since in a simulator the next datapoint is the sum of the previous one plus a small addition. For this reason α-values between 0.9 and 1 will be used in this project.

A problem of this method that should be noted concerns the wavelet normalisation to unit energy introduced by Torrence & Compo (Torrence & Compo, 1998). To make different wavelets at different scales comparable Torrence & Compo normalise all wavelets by a factor $\sqrt{\frac{2\pi s}{\Delta t}}$, meaning that in case the signal is sampled four times as fast the outcome of the CWT becomes twice as large. The background spectrum in Equation (3-8) does not change with Δt however, meaning that the size of significant regions changes with varying sample rates.

The effects of this normalisation have been discussed previously (Verspecht, 2011), showing that the normalisation can better be omitted for identification of neuromuscular admittance. Since the significance check proposed by Torrence & Compo only holds for normalised wavelets however this normalisation will remain to be used in this project, with another solution having been implemented instead.

Since all significance checks in this report having been performed for $\Delta t = 0.01$s and results were found to be satisfactory, wavelet transforms performed will from now on be normalised to a Δt of 0.01s by multiplying the CWT outcome by $\sqrt{\frac{\Delta t}{0.01}}$. In this way all wavelet transforms and significance checks will be directly comparable.

3-5 Chapter conclusions

In this chapter the following answers to research questions 2 and 3 posed in section 1-3 have been given:

2. *How does the wavelet transform of a single signal work?*
The Continuous Wavelet Transform (CWT) of a single signal determines the inner product of that signal with various stretched versions of a wavelet (all having a characteristic frequency) at all signal points in time. By doing so the frequency content of a signal as a function of time is determined.

- **2a.** *Are there any prerequisites on the signal?*
  The CWT requires a signal to have been sampled at a constant rate. This should not be a problem for simulator signals (as simulators usually function at a constant sampling rate).
- **2b.** *What computations are necessary to perform a wavelet transform?*
  To construct a CWT the inner product between a signal and a wavelet is computed at all points in time for all scales. This computation can be done both in the time domain (as a convolution) and the frequency domain (as a multiplication). Different wavelets are available in both domains.
- **2c.** *What kind of output does one obtain following a wavelet transform?*
  Since the wavelet transform is often complex, this transform can be presented on a (time,frequency)-grid (with each frequency corresponding to a certain scale) by plotting either the real part, imaginary part or absolute value of the CWT in a contour plot.
- **2d.** *Does the wavelet transform have any limitations?*
  The wavelet transform has a number of limitations:
  1. The maximum analysable frequency of the signal is limited by the Nyquist frequency, the minimum analysable frequency is limited by the signal length.
  2. Wavelet analysis is not able to determine the value of a constant signal / doesn't take the mean of a signal into account.
  3. Wavelet analysis has a limited time and frequency resolution, meaning that concentrated signal power content (in either time or frequency) spreads out over the (time-frequency grid).
  4. Near the edges of a signal a wavelet stretches beyond the interval of the signal, making use of false data. A Region of Influence is drawn to indicate caution should be taken with interpreting this data.

3. *Which wavelet types are best suited for motion cueing error detection?*
In previous work using wavelet transformations little overview has been given of all available wavelets and their (dis)advantages. Such an overview has been created in this chapter, allowing for choosing a suitable wavelet given the goal of a particular wavelet analysis. In case of motion cueing error detection it turns out that complex-valued wavelets with a good time resolution (such as the time-based complex Gaussian with n = 2 or the complex Morlet wavelet with $\omega_0 = 4$) are best suited, for reasons discussed in chapter 5.

3a. *What different types of wavelets are commonly used?*
Broadly speaking two types of wavelets are often used: time-based and frequency-based wavelets. An overview has been provided of ten different wavelets (seven time-based, three frequency-based) and their parameters. The frequency-based complex Morlet wavelet is used most often as it was introduced first by Torrence & Compo and in general offers a good trade-off between time resolution and frequency resolution (Torrence & Compo, 1998). A second distinction between wavelets can be made on whether they are complex or real.

3b. *What are the advantages of these types of wavelets?*
All wavelets have a different time and frequency resolution, with the time-based Haar wavelet offering the best time resolution (but worst frequency resolution) and the time-based Shannon/Frequency B-spline wavelets offering the best frequency resolution (but worst time resolution). All other wavelets are somewhere in between, with their parameters offering the possibility to improve either time or frequency resolution.

Complex-valued wavelets (such as the complex Morlet wavelets) are able to continuously indicate power at a certain frequency over time, whereas real-valued wavelets will show oscillating power behaviour. An additional advantage of complex-valued wavelets is their ability to indicate the phase of a signal's frequency content, a feature real-valued wavelets do not offer. Finally frequency-based wavelets have an advantage over time-based wavelets in terms of computational effort required, as the convolution operation in the time domain is computationally expensive.

Chapter 4

Signal Comparison

In the previous chapter the details of the Continuous Wavelet Transform have been discussed, showing the result of the wavelet analysis on a single signal. In this chapter the use of the CWT to compare two different signals will be investigated. The possibilities of the analysis of a simulator error signal will be discussed in section 4-1, the possibilities of the wavelet coherence method will be discussed in section 4-2. Again the answers this chapter provides with respect to the research questions formulated in chapter 1 are summarised in section 4-3.

4-1 Error signal analysis

The easiest way of comparing two time signals using wavelet analysis would be to compute the CWT of both signals and compare these two spectra. Should one choose this approach, then the linearity of the wavelet transform (which shows in Equation (3-4) in subsection 3-2-2) makes this approach simpler, since $\text{CWT}(f_2(t) - f_1(t)) = \text{CWT}(f_2(t)) - \text{CWT}(f_1(t))$. Hence one can simply compute the error signal between a motion simulator's input and output and analyse this signal.

Analysis of this error signal allows for identifying at what points in time and at what frequencies errors occur between the simulator input and output signal. Such an analysis does not give a complete picture however, since only *relative* differences are determined but the reference input value is lost. As a result the power ratio discussed in section 2-3 can't be determined. This reference value could be determined from the (CWT of the) original input and output signal of course.

4-2 Wavelet coherence

Wavelet analysis also allows for determining the coherence between two signals at specific points in time and at specific frequencies, this concept will from now on be referred to as wavelet coherence. The determination of wavelet coherence was introduced by Torrence &

Webster (Torrence & Webster, 1999), who indicate that the first step in determining wavelet coherence is the computation of the cross-wavelet spectrum. This cross-wavelet spectrum is defined as the product of the CWTs of two signals (for one of which the complex conjugate is taken):

$$W_{X,Y}(s,t) = W_X(s,t)W_Y^*(s,t) \tag{4-1}$$

The cross-wavelet spectrum shows the points in the (scale,time)-domain where the two signals share power content. The cross-wavelet spectrum of a sine (with constant frequency) and a chirp signal (with an initial frequency lower than the sine frequency but increasing to above the sine frequency) will show high power at those points in time where the frequencies of the two signals are (nearly) equal.

After computing this cross-wavelet spectrum the wavelet squared coherence can be computed through Equation (4-2):

$$\rho^2(s,t) = \frac{\left|S\left(s^{-1} \cdot W_{X,Y}(s,t)\right)\right|^2}{S\left(s^{-1} \cdot |W_X(s,t)|\right)^2 \cdot S\left(s^{-1} \cdot |W_Y(s,t)|\right)^2} \tag{4-2}$$

In this equation all wavelet spectra are multiplied by the scale inverse to convert back to an energy density and $S(\cdot)$ denotes a smoothing operator in both time and frequency which is further explained in (Grinsted et al., 2004). Both in these articles and in Matlab this wavelet coherence is only implemented for the frequency-based complex Morlet wavelet discussed in subsection 3-3-2, in this thesis an extension will be made to all other wavelets.

Grinsted, Moore and Jevrejeva indicate that the smoothing operator applied in time is given a similar shape as the wavelet and Torrence & Webster say *"The time smoothing uses a filter given by the absolute value of the wavelet function at each scale, normalized to have a total weight of unity"*, which for the frequency-based complex Morlet wavelet is $C \cdot e^{-t^2/2s^2}$ (see Table 3-8). To extend this operation to other wavelet types the shape of other wavelets has been given in subsection 3-3-2, for completeness these shapes are summarised below:

Wavelet type	**Shape in time**	**Shape in frequency**
Time-based complex Morlet	$\mathrm{C}\cdot e^{-t^2/f_b s^2}$	$\mathrm{C}\cdot e^{-(s\omega)^2 \cdot f_b/4}$
Time-based Gaussian	$\mathrm{C}\cdot e^{-t^2/2s^2}$	$\mathrm{C}\cdot e^{-(s\omega)^2/2}$
Time-based complex Gaussian	$\mathrm{C}\cdot e^{-t^2/2s^2}$	$\mathrm{C}\cdot e^{-(s\omega)^2/2}$
Time-based Shannon	$\left\lvert \mathrm{C} \cdot sinc(\frac{f_b t}{s}) \right\rvert$	–
Time-based Frequency B-spline	$\left\lvert \mathrm{C} \cdot \left[sinc(\frac{f_b t}{Ms})\right]^M \right\rvert$	–
Haar wavelet	$\mathrm{C}_{(-\frac{s}{2},0)} - \mathrm{C}_{(0,\frac{s}{2})}$	–
Daubechies wavelet	–	–
Frequency-based Morlet	$\mathrm{C}\cdot e^{-t^2/2s^2}$	$\mathrm{C}\cdot e^{-(s\omega)^2/2}$
Frequency-based Paul	$\mathrm{C}\cdot \left\lvert (1 - it/s)^{-(m+1)} \right\rvert$	–
Frequency-based Gaussian	–	–

Table 4-1: Wavelet shape used for smoothing in time in the wavelet coherence.

The constant C present in these smoothing functions scales the time function to ensure unit energy but does not alter the shape, and drops out in case Equation (4-2) is applied (since the constant is squared in the numerator and features twice in the denominator). It is for this reason that no further investigation into the value of this constant is done.

In Matlab this smoothing in time is usually applied in the frequency domain, as this allows for a regular multiplication rather than a convolution between the CWT spectrum and the smoothing function. Hence computational effort can be saved in case a Fourier transform of the smoothing functions are known, these are given (as far as possible) in Table 4-1 too. Again scaling factors in the Fourier transform have been combined into one constant C (which is not necessarily equal to the constant in the time domain functions). Note that although frequency descriptions were given for more wavelet modulating functions in subsection 3-3-2, the fact that the absolute value is used for the smoothing function means these frequency descriptions are no longer valid.

The smoothing in frequency is done in Matlab using a moving average filter, the width of which can be varied. A narrower moving average provides higher resolution in the coherence plots.

The wavelet coherence method also allows for determination of the phase difference between two signals, defined by Torrence & Webster as:

$$\phi(s,t) = \tan^{-1}\left(\frac{\mathfrak{Im}\left[S(s^{-1}\cdot W_{X,Y}(s,t))\right]}{\mathfrak{Re}\left[S(s^{-1}\cdot W_{X,Y}(s,t))\right]}\right) \tag{4-3}$$

It is common practise to plot the wavelet coherence and phase together in one contour plot, using arrows to illustrate the phase difference between the two signals at different points in time and frequency. Right-pointing arrows in that case indicate no phase difference, left-pointing arrows indicate antiphase, downward pointing arrows indicate signal X leading signal Y by 90° and arrows pointing upwards indicate signal X trailing signal Y by 90°.

An example of the result of a coherence analysis is shown in Figure 4-1, in which two multisine signals are compared (using a frequency-based complex Morlet wavelet with $\omega_0 = 6$). Signal X is a multisine featuring a component with a period of 4 seconds and a component with a period of 8 seconds. Additionally an 8Hz component ($P = 0.125$s) is added between $t = 5$s and $t = 10$s. Signal Y features the same components as signal X (though scaled differently), with the 8s period component being anti-phase, the 4s component lagging by 0.5 seconds (so 45°) and the 8Hz component being in perfect phase. Added to these components is a low-amplitude random Gaussian noise.

These phase differences are also shown by the arrows in Figure 4-1, which have only been plotted in regions where $\rho^2 > 0.8$. The right-pointing arrows between $t = 5$s and $t = 10$s at $P = 0.125$s indicate the signal components are in phase, the left-pointing arrows at $P = 8$s indicate anti-phase and the arrows pointing downwards to the right at $P = 4$s indicate signal X leading signal Y.

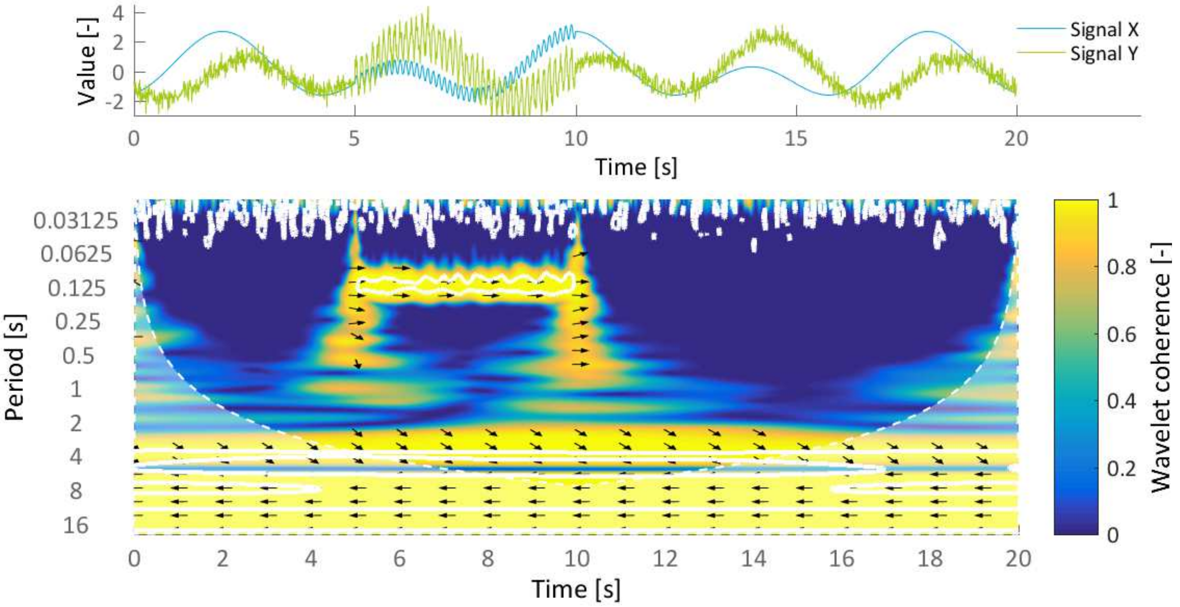


Figure 4-1: Illustration of the coherence analysis between two multisine signals X and Y. Both share (temporary) content at a period of 0.125s, 4s and 8s (as shown by high coherence values), with arrows indicating relative phase.

Besides high coherence at the expected frequencies and points in time the wavelet coherence algorithm also detects signal coherence at other instances, most notably at $t = 5$s and $t = 10$s (with the width of this column increasing for larger periods). This high coherence is attributed to the sudden onset and disappearance of the 8Hz signal, which causes high-frequency power at these instances in the Continuous Wavelet Transform spectra of the individual signals. This high-frequency power then spreads out to lower frequencies.

To distinguish between parts of Figure 4-1 showing actual signal coherence and parts influenced by CWT noise, the significance contour of the error between the signals in the top of Figure 4-1 has been added to the bottom figure. As is easily seen this significance contour excludes the areas of high coherence around $t = 5$s and $t = 10$s while indicating parts of the plot at $P = 0.125$s, $P = 4$s and $P = 8$s as being significant.

4-3 Chapter conclusions

In this chapter the following research questions of this project have been answered:

4. *How do you compare two wavelet-transformed signals?*

- **4a.** *What options for comparison are available?*
  Generally speaking two ways of analysis are possible for two wavelet-transformed signals: one can subtract two Continuous Wavelet Transforms to obtain the CWT of the error signal and analyse that CWT (using significance testing described in subsection 3-4-2) and one can determine the wavelet coherence using the CWTs of two signals. The latter method also allows for commenting on the relative phase difference between the two signals.
- **4b.** *Is there a specific type of data these methods are most suited to?*
  Analysing the error signal between an input and output signal is possible for any type of input. The determination of wavelet coherence will only hold value in case signals are expected to be at least partly coherent (or coherent at a particular frequency), as in case signals are not coherent at all the procedure will only return noise.
- **4c.** *What method is best suited for the comparison of simulator input and output?*
  Both methods described hold value for the analysis and identification of motion cueing errors, with error signal analysis allowing for the identification of errors, and wavelet coherence assisting in the determination of the motion cueing error type.

Chapter 5

Error Recognition

In this thesis different types of motion cueing errors have been discussed in chapter 2, with mathematical properties of these errors having been described in section 2-3. The possibilities of wavelet analysis to detect signal characteristics and errors have then been described in chapters 3 and 4. In this chapter the contents of these chapters will be combined by assessing how wavelet analysis can be used to detect these different motion cueing errors, this process is described in section 5-1. The workings of this process are then demonstrated by applying the error detection scheme to idealised motion cueing errors in section 5-2, with chapter conclusions being given in section 5-3.

5-1 Error detection

Based on the error properties summarised in Table 2-2 and the wavelet analysis possibilities given in chapters 3 and 4 an error detection scheme as depicted in Figure 5-1 is proposed. The scheme has a decision-making structure similar to Grant's MCA tuning paradigm (Grant, 1995, p105-p137), contrary to that paradigm however the current algorithm detects Motion Cueing errors rather than tuning an MCA to mitigate them (with error detection in Grant's algorithm being done through subjective pilot evaluation). In the following the reasoning behind this scheme will be explained.

Step 1: Error Analysis - The starting point for motion cueing error detection will be a simulator's input and output signal. By subtracting these two the simulator error signal is obtained. As discussed in section 4-1 performing a Continuous Wavelet Transform of this error signal allows for determination of the time of occurrence and frequency content of motion cueing errors. This is done by applying the significance test discussed in subsection 3-4-2.

In case no significant region is found one can conclude that no errors were present in the simulation, in case any error is present the result of the significance test is a collection of significant regions in the (time,frequency)-domain.

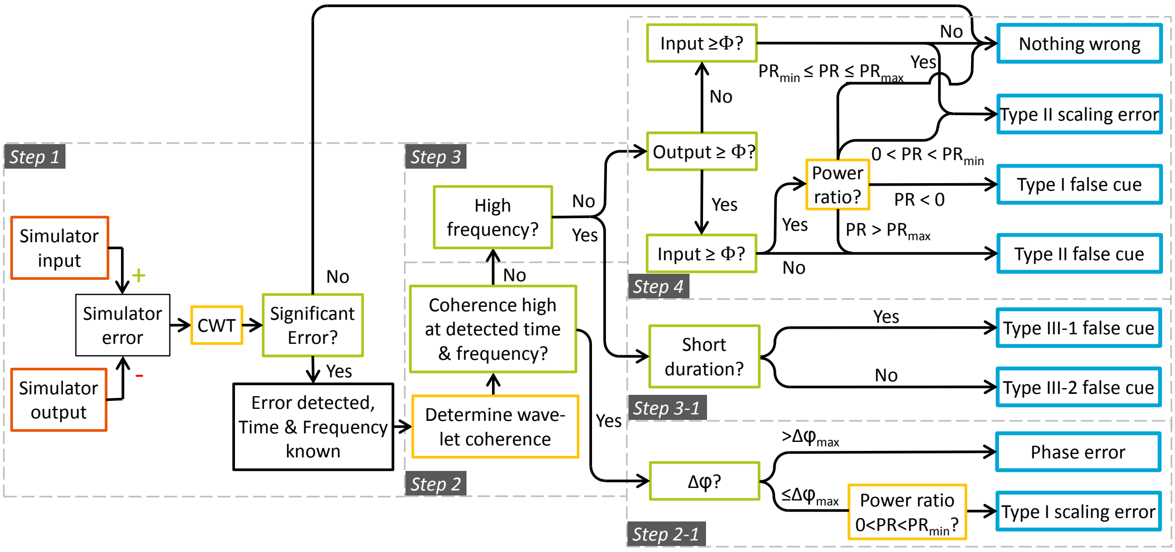


Figure 5-1: Proposed detection scheme for motion cueing errors, with orange blocks indicating inputs, yellow blocks indicating computational operations, green blocks indicating choice blocks and blue blocks representing conclusions.

For the purpose of identifying motion cueing errors it is best if a statistically significant area shows up as one whole, rather than multiple smaller regions. For this reason real wavelets (which produce CWT spectra oscillating between positive and negative in case continuous power is detected, leaving statistically insignificant regions in between peaks) are unsuited for this detection. It is for this reason that only complex wavelets will be considered for this step. As will further be explained later wavelets with a good time resolution are preferred for this detection (with the frequency resolution less relevant), meaning the parameters discussed in subsection 3-3-2 are set to provide decent time resolution.

Having established a collection of statistically significant regions at this stage, the rest of the motion cueing error detection scheme will be gone through for each significant region separately.

Step 2: Signal coherence analysis - From Table 2-2 one can observe that Type I scaling errors and phase errors stand out due to their high coherence between simulator input and output. For this reason the signals' coherence is determined using the methods described in chapter 4. Should all coherence values within the detected significant region be above 0.9 then an error is designated either a Type I scaling error or a phase error.

Step 2-1: Lag analysis - In order to discriminate between Type I scaling errors and phase errors the phase lag (denoted by $\Delta\phi$ in Figure 5-1) between signals X and Y will be determined from the coherence plot determined in step 2. In case a significant lag is present an error will be designated a phase error, otherwise a Type I scaling error is called in case the Power Ratio (discussed in Step 4) is below threshold level. From previous research it is known that a phase lag exceeding a threshold value of $\Delta\phi_{\text{max}} = 16.4°$ leads humans to perceiving a phase error

(Beckers, Pool, Valente Pais, van Paassen & Mulder, 2012; Jonik, Valente Pais, van Paassen & Mulder, 2011).

Step 3: Frequency analysis - Should the coherence between simulator input and output drop then from Table 2-2 one can observe that (having already dealt with Type I scaling errors) Type III false cues stand out due to their high-frequency content, with all other errors expected at lower frequencies. This characteristic is exploited to identify Type III false cues: in case a statistically significant region lies fully below the threshold level of $P = 2$s an error is attributed a Type III false cue (Grant, 1995, p96), if not signal coherence is investigated.

Step 3-1: Duration analysis - Having determined that an error is a Type III false cue, one would still like to discriminate between errors caused by a simulator hitting a displacement limit or errors caused by simulator vibrations or IMU measurement noise. From Table 2-2 the difference between these two causes is found to be the duration of the error. High-frequency errors lasting under 0.5 seconds will be attributed to simulator displacements, errors lasting longer will be attributed to simulator vibrations.

Step 4: Power Ratio determination - In case signal coherence drops below 0.9 in the identified region of significant error three options remain: the error is either a Type I false cue, a Type II false cue or a Type II scaling error. From Table 2-2 it is observed that the difference between the remaining errors shows through the Power Ratio (PR) of the simulator input and simulator output signal.

Ideally one would compute this power ratio from the CWTs of both signals (as that allows for taking into account only those frequencies at which an error is present), this however proved not to be possible due to the variability of and artefacts in the Continuous Wavelet Transform. Therefore the PR will be determined from the time signals, using:

$$\mathrm{PR}[n] = \frac{\mathrm{Output}[n]}{\mathrm{Input}[n]} \tag{5-1}$$

In this equation n denotes the time index of the signal. From the significance test in step 1 the start time and end time of the particular error are known, meaning that the indices n to be used in Equation (5-1) are known.

This power ratio does not yet tell the full story however, since it only comments on the relative difference between output and input. This problem is illustrated in Figure 5-2.

If one would go by the power ratio only then one would designate simulator washout (a simulator's return to neutral during a sustained cue, a process shown in section II of the left figure in Figure 5-2) as a Type I false cue, since the simulator accelerates in the wrong direction. Section I would be labelled a Type II scaling error (since the power ratio is smaller than 1), with section III being labelled a Type II false cue (since there is output in the absence of input).

In a well-designed simulator however the subject won't notice the washout because the washout is done at sub-threshold levels. This effect is illustrated on the right-hand side of Figure 5-2. Again section I would be identified as a Type II scaling error (however only once both the input and output surpass threshold levels), but since the washout

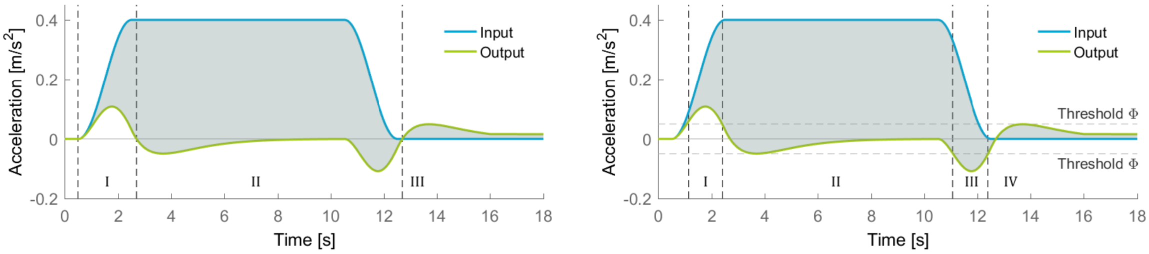


Figure 5-2: Effect of thresholds being taken into account during motion cueing error identification.

takes place at sub-threshold levels the subject won't notice the erroneous output. It is for this reason that Section II should be identified as a Type II scaling error (absent output in the presence of input), with only section III being identified as a Type I false cue. Since the output in section IV is below the perceptual threshold level, no error should be reported here.

The use of perceptual thresholds (denoted Φ) therefore allows for error determination in case either the simulator input or output signal is below the perceptual threshold level, using the scheme shown in Table 5-1.

	Input $< \Phi$	**Input** $> \Phi$
Output $< \Phi$	No error	Type II scaling error
Output $> \Phi$	Type II false cue	Use Power Ratio

Table 5-1: Decision scheme in case either the simulator input or simulator output signal is below the perceptual threshold level Φ.

For this reason a Type I error is only called in case the power ratio is negative and both the simulator input and output signal are above the human perceptual threshold level. These reference values differ for each motion axis, in this preliminary study the values given in Table 5-2 will be used (which assumes these thresholds to be frequency-independent).

Motion	**Perceptual threshold**
a_x	0.04 m/s^2 (Zaichik et al., 1999; Heerspink et al., 2005)
a_y	0.04 m/s^2 (Zaichik et al., 1999; Heerspink et al., 2005)
a_z	0.08 m/s^2 (Zaichik et al., 1999; Heerspink et al., 2005)
ω_x	3 °/s (Groen & Bles, 2004)
ω_y	3 °/s (Groen & Bles, 2004)
ω_z	3 °/s (Groen & Bles, 2004)

Table 5-2: Perceptual thresholds used by the motion cueing error detection algorithm.

The concept of perceptual thresholds allows for the determination of the scaling threshold below which subjects perceive motion to be too weak (hence implying a Type II scaling error) and the scaling threshold above which subjects perceive motion as being too strong (introducing a Type II false cue). Again assuming these thresholds to be frequency-independent (and constant for different motion amplitudes), the following critical power ratio values have been determined:

Motion		Lower gain	Upper gain
Acceleration	(Correia Grácio et al., 2013)	0.5	1.3
Rotation	(Valente Pais, van Paassen & Mulder, 2009)	0.6	1.1

Table 5-3: Minimum and maximum allowable scaling factors for simulator accelerations and rotations.

In case both the simulator input and output are above the perceptual threshold level (see Table 5-1) and a PR lower than the minimum allowable scaling factor is determined a Type II scaling factor is called at the time point under consideration, in case a PR higher than the maximum allowable scaling factor is detected the point in time is assigned a Type II false cue. Should the Power Ratio be within the bounds of the coherence zone, no error is present.

5-2 Proof of concept

In this section the error detection scheme introduced in the previous section will be applied to idealised versions of the motion cueing error types discussed in section 2-2. These idealised motion cueing errors are shown in the top left of Figures 5-3 through 5-9. Notice that only the Type I false cue isn't a 'perfect' motion cueing error, with its output having been created by high-pass filtering the input. As a result one could partly typify this error as a Type II scaling error too (in regions where the accelerations are not of opposite sign, but rather any output is missing). In this section it is assumed these signals represent longitudinal accelerations, meaning the perceptual threshold for the input and output signal is set to 0.04 m/s^2, the minimum allowable power ratio is 0.5 and the maximum allowable power ratio is 1.3.

For all input and output pairs the error signal has been shown on the top-right of Figures 5-3 through 5-9. For all these error types the error detection scheme proposed in the previous section is gone through.

Step 1: Error Analysis - As stated in section 5-1 the Continuous Wavelet Transformation of the motion cueing error signal should be done using a complex wavelet with a relatively high time resolution (since the start and end time of error regions will be used to pinpoint errors in time). In order to be able to compare spectra to one another the time-based complex Gaussian wavelet was chosen for all error types, and to improve time resolution the order was set to $n = 2$.

As explained in subsection 3-4-2 the choice of lag-1 autocorrelation coefficient α has a major influence on the regions being rated as significant. For all but the Type III false cues it was found that high α-values were required to raise the threshold level for low-frequency signal content (which characterises these errors), hence analysis of these errors was done using $\alpha = 0.98$. For Type III errors it was found that lowering the autocorrelation value to 0.9 resulted in better time and frequency resolution of the error detection (owing to high-frequency thresholds being raised when lowering the α-value, see Figure 3-15). Both values correspond to the α-interval proposed in subsection 3-4-2.

The Continuous Wavelet Transform spectrum of all errors is shown in the middle left plots of Figures 5-3 through 5-9. As can be seen most errors are characterised by a single significant region, the exception being the phase error.

Step 2: Signal Coherence analysis - From the middle right plots in Figures 5-3 through 5-9 it is easily observed that only the Type I scaling error shows consistently high coherence, separating it from the rest. Notice that besides the period being shown in the left axis, the frequency (in radians per second) corresponding to these periods is shown as well to allow for relating these plots to the dynamics of the human senses.

Step 2-1: Lag analysis - From the middle right plot in Figure 5-3 it is easily seen the arrows indicate no phase lag, meaning this error is likely a Type I scaling error. Taking the minimum and maximum time value from the identified region of significant error, the Power Ratio (see the bottom left of Figure 5-3) is then computed in case the output acceleration is above the threshold level. Based on this Power Ratio a Type I scaling error is called.

It should be noted that, as opposed to what was expected, signal coherence does not remain high for the perfect phase error in Figure 5-9. As a result, the algorithm will not identify this error as a perfect phase error, as shown in the bottom right of Figure 5-9. This finding is attributed to the fact that signal coherence will only be high for a phase error in case multiple periodic cycles are gone through at a certain frequency, something that will seldomly be the case for motion cueing signals. As a result it is expected that many phase errors in motion simulation will be identified as a combination of other errors, as is the case in Figure 5-9.

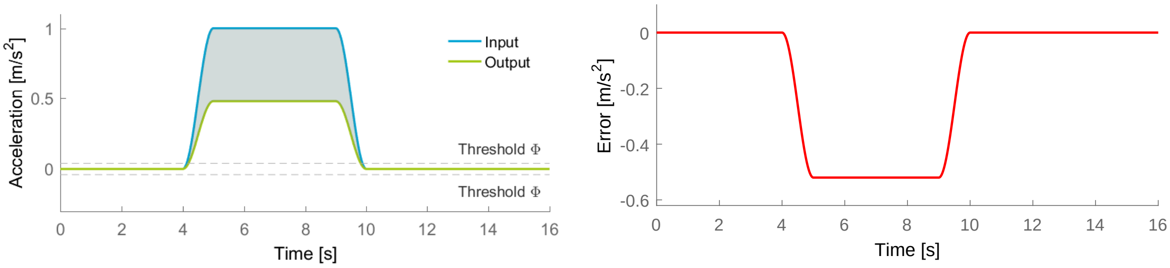
Step 3: Frequency analysis - From the middle left motion cueing error CWT plots (which again show both period and frequency) the frequency content of the error can be investigated. As expected, the two false cues (Figures 5-7 and 5-8) only show high-frequency content, all other errors possess power at periods of over 2 seconds.

Step 3-1: Duration analysis - From Figures 5-7 and 5-8 it is easily noted that the high-frequency power lasts for a short period of time in Figure 5-7, whereas power lasts much longer in Figure 5-8. As a result, the algorithm considers the error in Figure 5-7 a Type III-1 false cue due to a simulator hitting a displacement, and considers the other a Type III-2 false cue due to simulator vibrations (as can be seen from the bottom left figures).

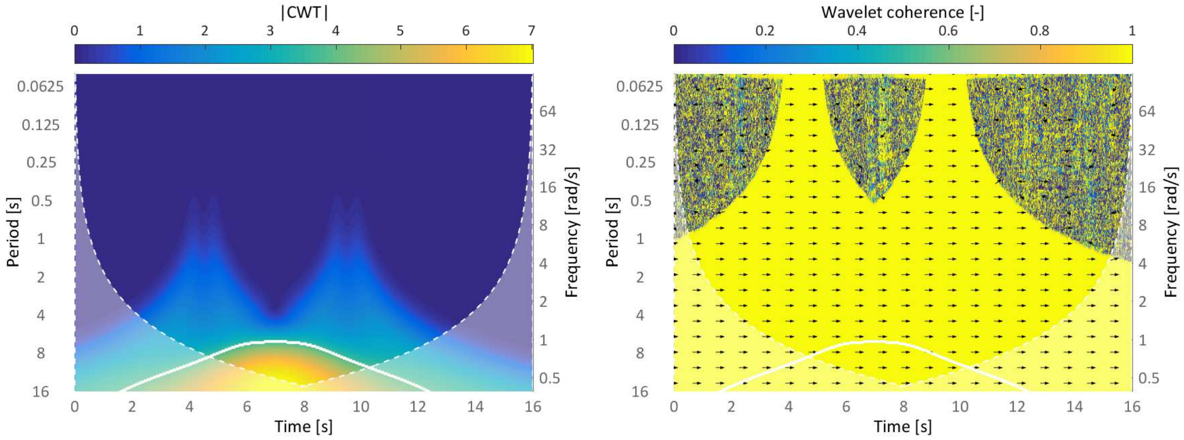
Having detected a high-frequency error in the region of significant error, the other algorithm steps are skipped (explaining why no Power Ratio is determined).

Step 4: Power Ratio analysis - For the four remaining errors Table 5-1 is gone through to determine what error is present. In case either the input or output is below threshold levels either no error, a Type II scaling error or a Type II false cue is called, otherwise the Power Ratio is determined as shown in the bottom left of Figures 5-3 through 5-9. Based on this power ratio a Type I false cue, Type II false cue or Type II scaling error is called. In case the power ratio is within the perception coherence zone discussed in section 5-1 no error is called.

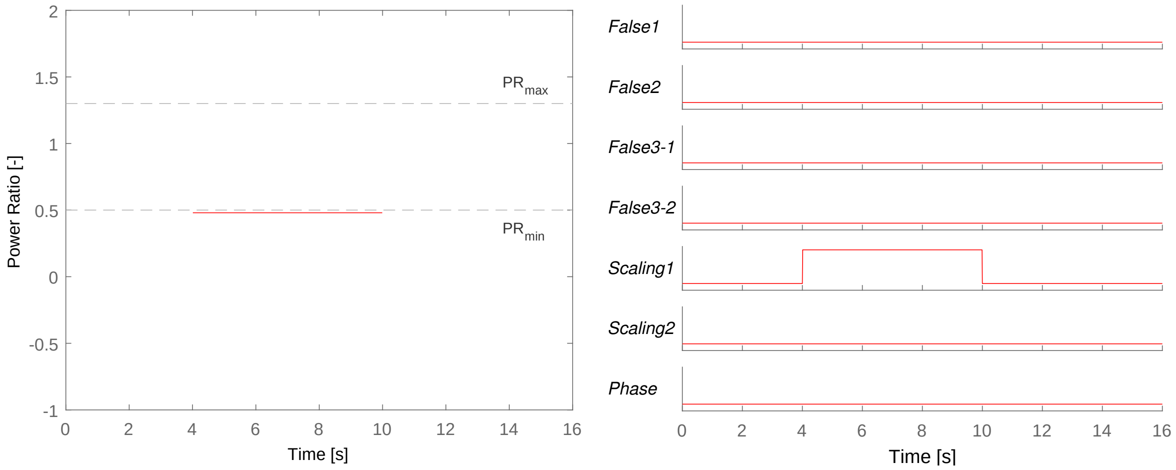
From Figures 5-4, 5-5, 5-6 and 5-9 one can see that all three type of errors are identified correctly. Noticeable is the fact that an intended phase error is detected as a combination of a Type II scaling error and Type II false cue (for reasons discussed previously).



(a) Input & Output signals of a Type I scaling error. **(b)** Error signal of a typical Type I scaling error.

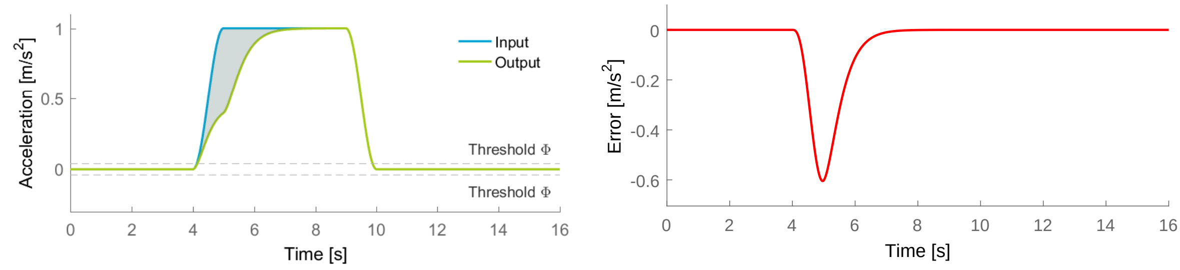


(c) CWT (using the time-based complex Gaussian ($n = 2$) of a Type I scaling error. Significant regions are enclosed in white contours. **(d)** Wavelet coherence of the input and output signal of a Type I scaling error. Significant regions are enclosed in white contours.



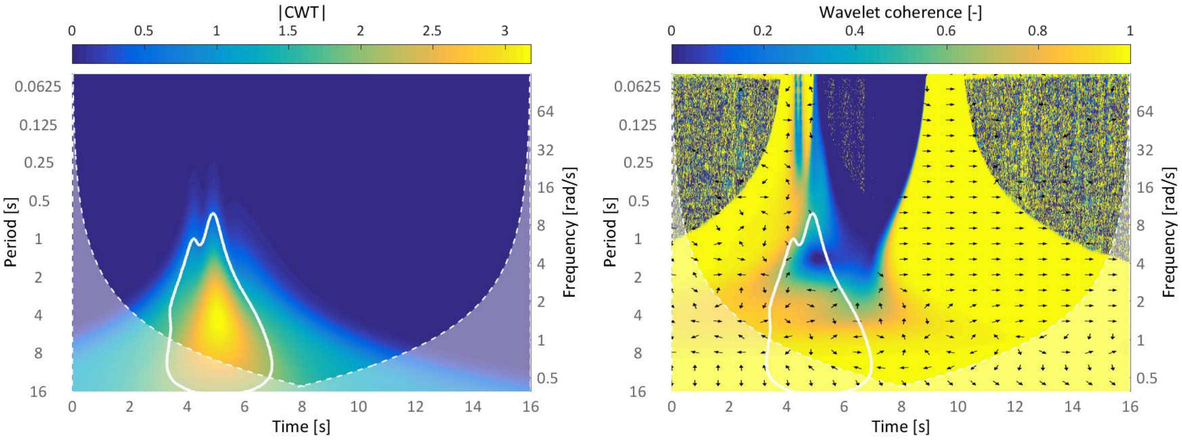
(e) Power ratio (at time points required) for the Type I scaling error, with the coherence zone indicated. **(f)** Algorithm outcome, highlighting the presence of a Type I scaling error.

Figure 5-3: Graphical output of the motion cueing error type detection algorithm for an idealised Type I scaling error.



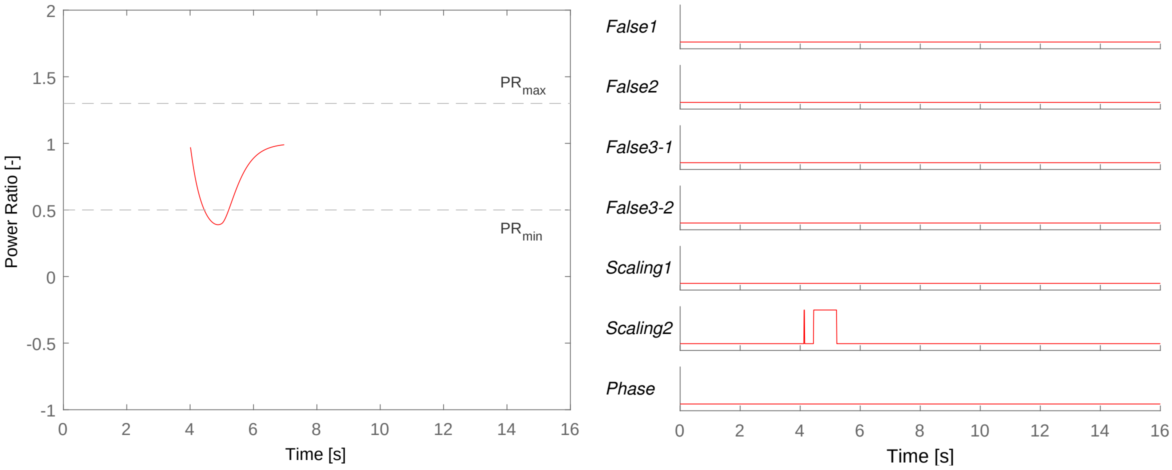
(a) Input & Output signals of a Type II scaling error.

(b) Error signal of a typical Type II scaling error.



(c) CWT (using the time-based complex Gaussian $(n = 2)$ of a Type II scaling error. Significant regions are enclosed in white contours.

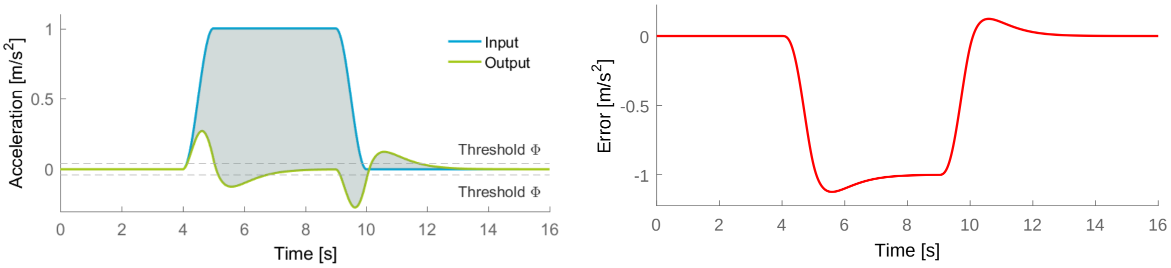
(d) Wavelet coherence of the input and output signal of a Type II scaling error. Significant regions are enclosed in white contours.



(e) Power ratio (at time points required) for the Type II scaling error, with the coherence zone indicated.

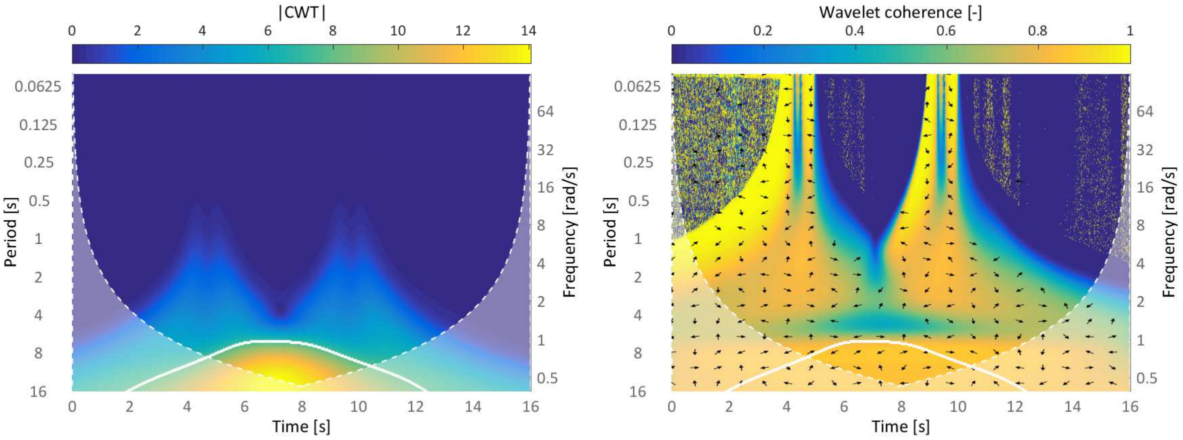
(f) Algorithm outcome, highlighting the presence of a Type II scaling error.

Figure 5-4: Graphical output of the motion cueing error type detection algorithm for an idealised Type II scaling error.



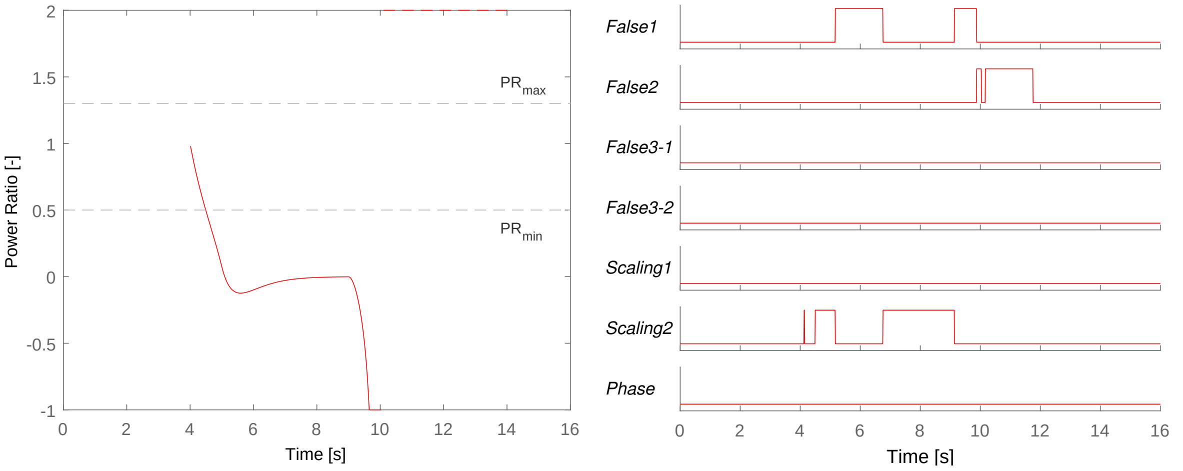
(a) Input & Output signals of a Type I false cue.

(b) Error signal of a typical Type I false cue.



(c) CWT (using the time-based complex Gaussian ($n = 2$) of a Type I false cue. Significant regions are enclosed in white contours.

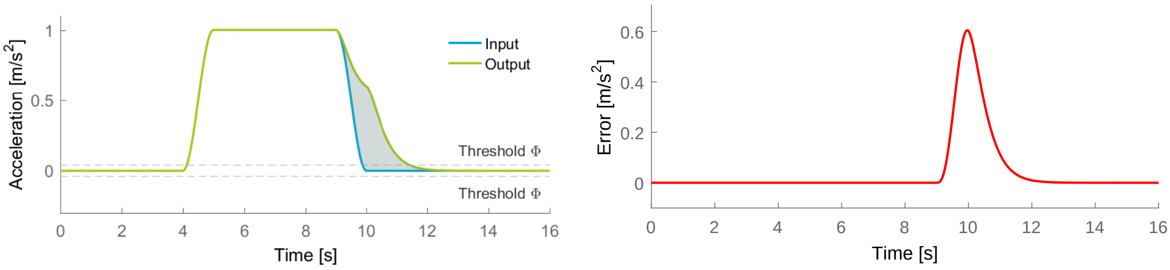
(d) Wavelet coherence of the input and output signal of a Type I false cue. Significant regions are enclosed in white contours.



(e) Power ratio (at times required) for the Type I false cue, with the coherence zone indicated. Dashed lines at PR = 2 and PR = -1 indicate PR-values outside the figure boundaries.

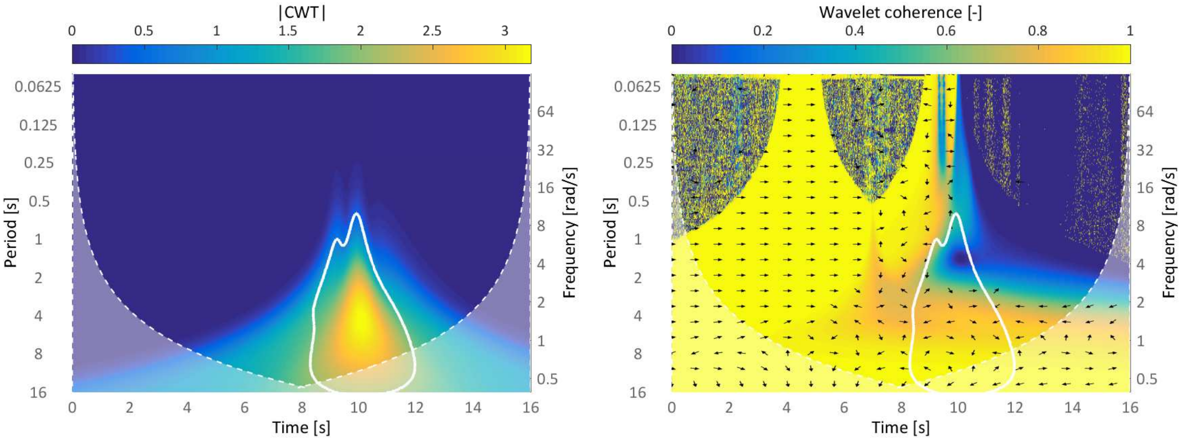
(f) Outcome of the algorithm, highlighting the presence of a Type I false cue at two instances in time, as well as Type II false cues and Type II scaling errors.

Figure 5-5: Graphical output of the motion cueing error type detection algorithm for an idealised Type I false cue.



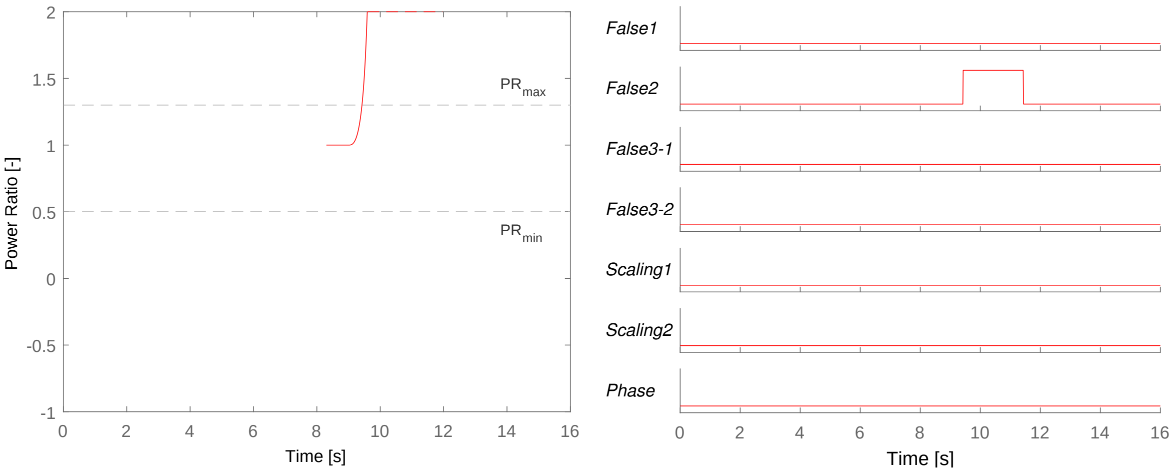
(a) Input & Output signals of a Type II false cue.

(b) Error signal of a typical Type II false cue.



(c) CWT (using the time-based complex Gaussian ($n = 2$) of a Type II false cue. Significant regions are enclosed in white contours.

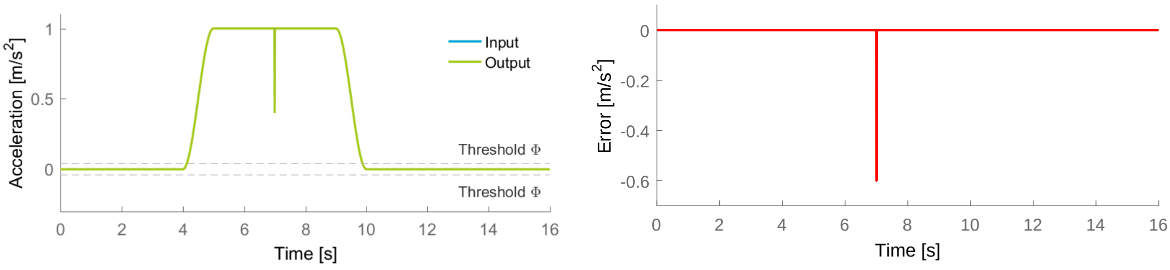
(d) Wavelet coherence of the input and output signal of a Type II false cue. Significant regions are enclosed in white contours.



(e) Power ratio (at times required) for a Type II false cue, with the coherence zone indicated. The dashed line at PR = 2 represents PR-values larger than 2.

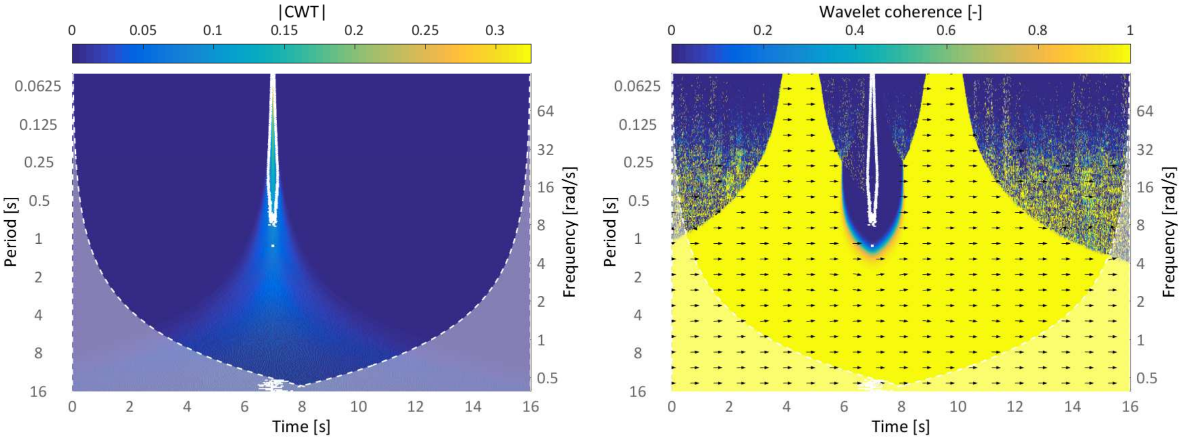
(f) Algorithm outcome, highlighting the presence of a Type II false cue between $t = 9.5$s and $t = 12$s.

Figure 5-6: Graphical output of the motion cueing error type detection algorithm for an idealised Type II false cue.



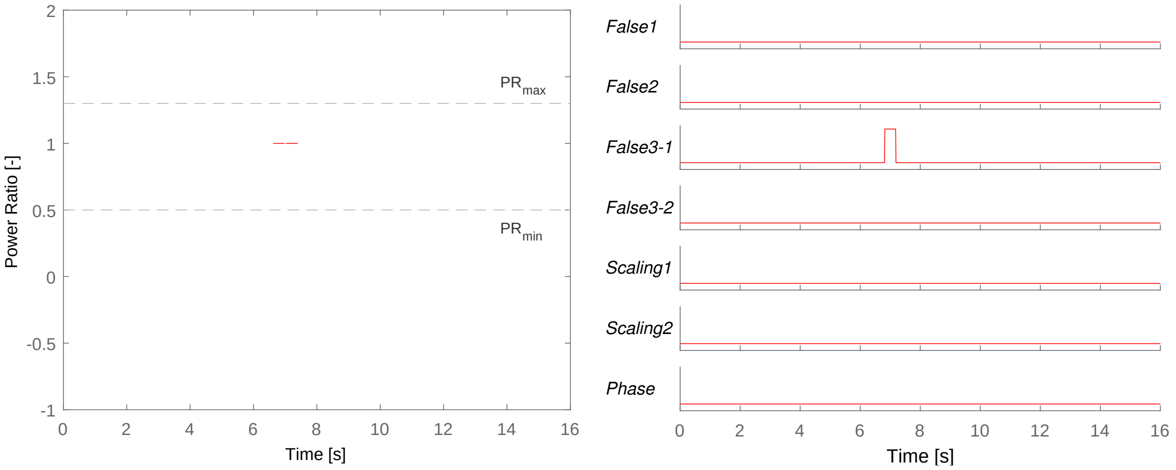
(a) Input & Output signals of a Type III false cue caused by a simulator displacement limit hit.

(b) Error signal of a typical Type III false cue caused by a simulator displacement limit hit.



(c) CWT (using the time-based complex Gaussian ($n = 2$) of a Type III false cue. Significant regions are enclosed in white contours.

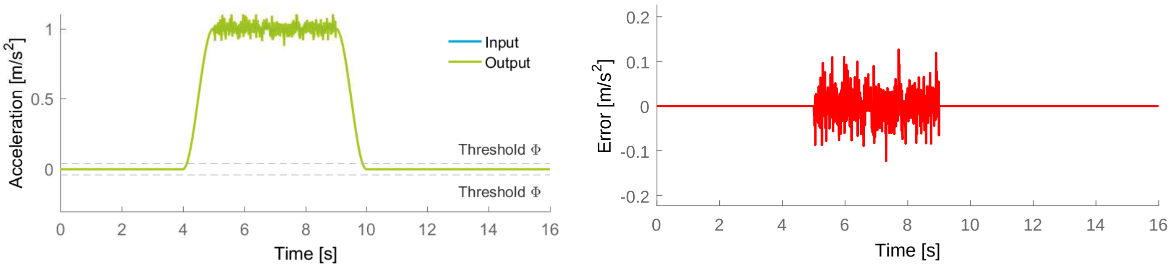
(d) Wavelet coherence of the input and output signal of a Type III false cue. Significant regions are enclosed in white contours.



(e) Power ratio (at time points required) for the Type III false cue caused by a simulator hitting its displacement limit, with the coherence zone indicated.

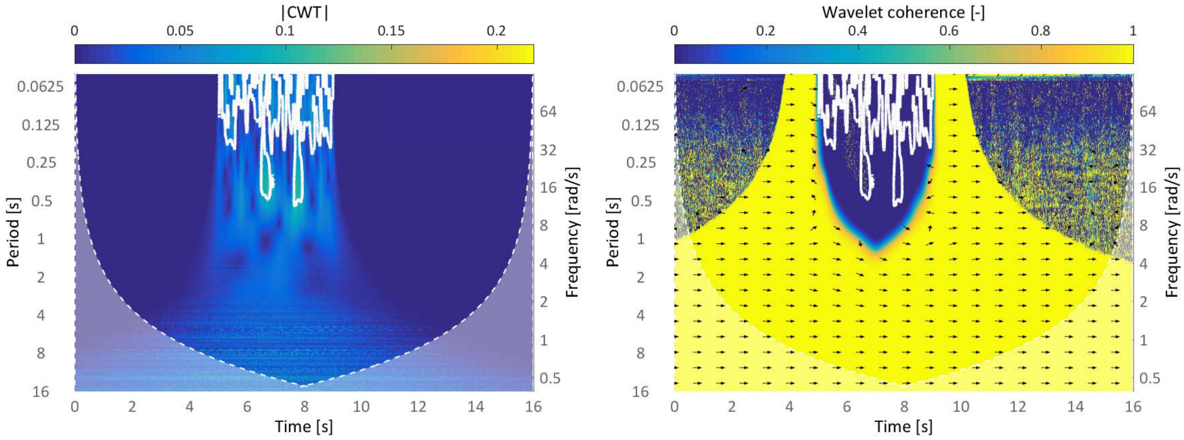
(f) Algorithm outcome, highlighting the presence of a Type III false cue caused by a simulator hitting its displacement limit.

Figure 5-7: Graphical output of the motion cueing error type detection algorithm for an idealised Type III-1 false cue caused by a simulator displacement limit hit.



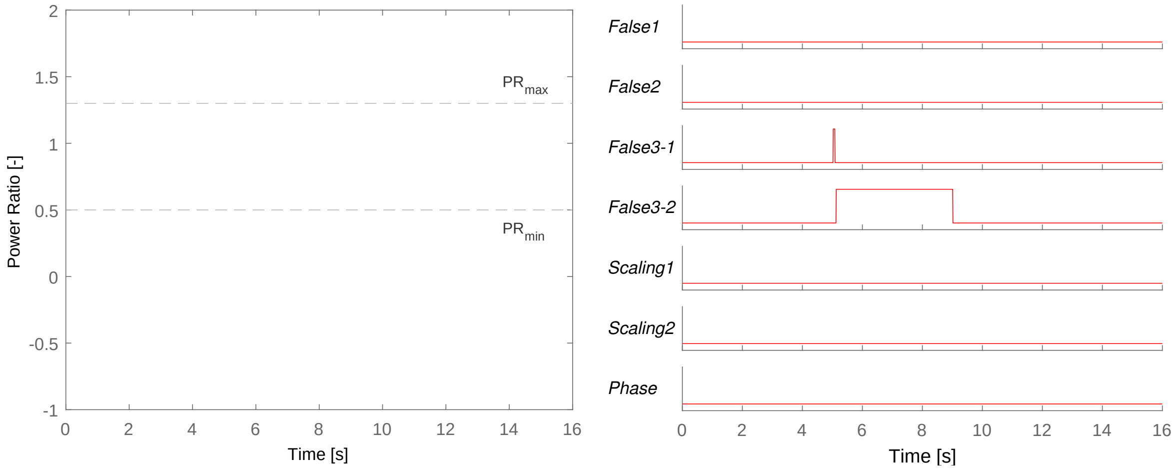
(a) Input & Output signals of a Type III false cue caused by simulator vibrations.

(b) Error signal of a typical Type III false cue caused by simulator vibrations.



(c) CWT (using the time-based complex Gaussian $(n = 2)$ of a Type III false cue. Significant regions are enclosed in white contours.

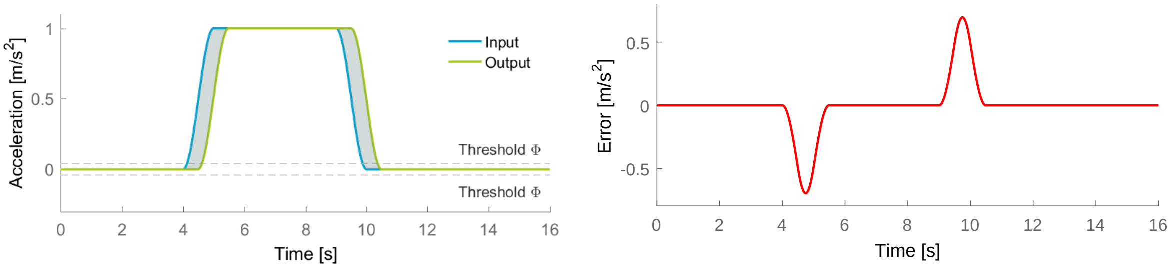
(d) Wavelet coherence of the input and output signal of a Type III false cue. Significant regions are enclosed in white contours.



(e) Power ratio (at time points required) for the Type III-2 false cue, with the coherence zone indicated.

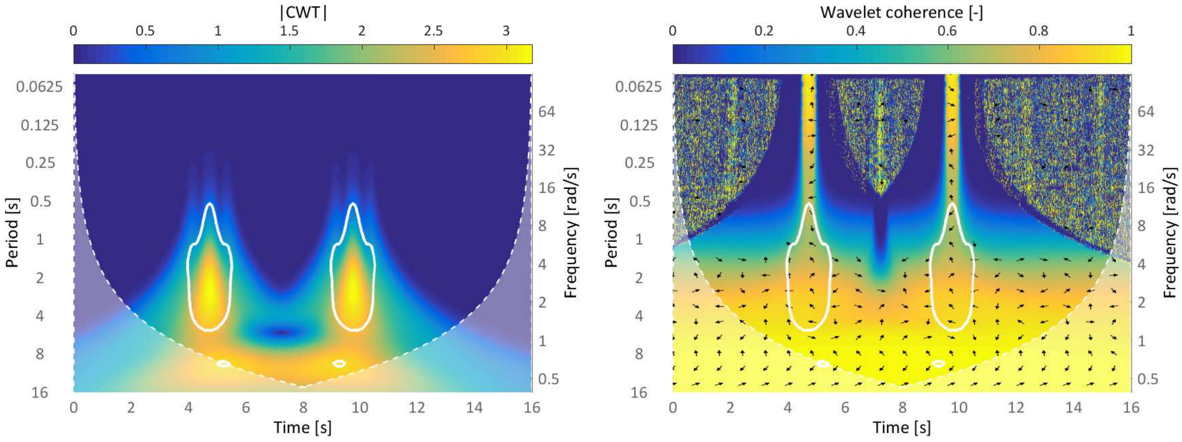
(f) Algorithm outcome, highlighting the presence of a Type III false cue due to simulator vibrations.

Figure 5-8: Graphical output of the motion cueing error type detection algorithm for an idealised Type III-2 false cue due to simulator vibrations.



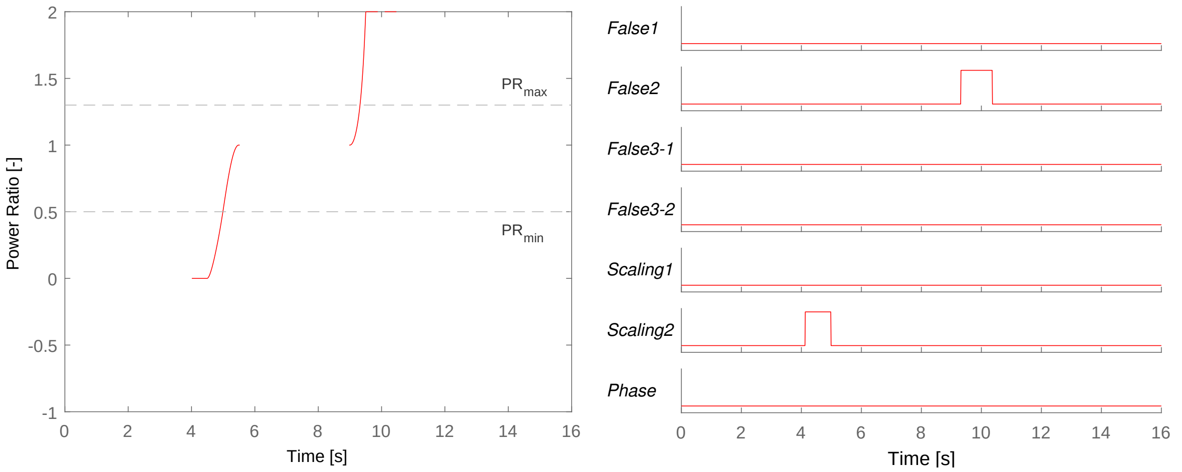
(a) Input & Output signals of a typical phase error.

(b) Error signal of a typical phase error.



(c) CWT (using the time-based complex Gaussian ($n = 2$) of a phase error. Significant regions are enclosed in white contours.

(d) Wavelet coherence of the input and output signal of a phase error. Significant regions are enclosed in white contours.



(e) Power ratio (at times required) for a phase error, with the coherence zone indicated. The dashed line at PR = 2 represents PR-values larger than 2.

(f) Algorithm outcome, highlighting the presence of a Type II scaling error and Type II false cue, rather than a phase error.

Figure 5-9: Graphical output of the motion cueing error type detection algorithm for an idealised phase error.

5-3 Chapter conclusions

In this chapter the preparatory work of all chapters so far has been combined to explain a process capable of detecting various types of motion cueing errors. In the process the following answers have been provided to research questions posed in chapter 1:

1. *Can we come up with a set of mutually exclusive categories of motion cueing errors?*

  1c. *What mathematical technique(s) allow(s) for identifying the properties of and differences between these error types?*
  It has previously been determined that motion cueing errors can be discriminated based on their frequency content, duration, phase different, signal coherence and input/output ratio. In this chapter a variety of mathematical techniques has been applied to categorise errors based on these properties.

  The timing and frequency content of motion cueing errors was determined using the spatio-temporal wavelet analysis technique using significance checking. Using this same technique signal coherence and signal phase difference were determined. The relevant input/output ratios of simulator signals were determined by dividing the two signal values at time points of interest.

3. *Which wavelet types are best suited for motion cueing error detection?*
For accurate identification of the timing and frequency content of motion cueing errors only complex wavelets can be used, with wavelets with a higher time resolution leading to better time-identification of errors. As such the complex Morlet (both time- and frequency-based), complex Gaussian and Paul wavelet are suitable for use.

4. *How do you compare two wavelet-transformed signals?*

  4c. *What method is best suited for the comparison of motion simulator input and output?*
  In this chapter it has been shown that a combination of methods is required to identify different types of motion cueing error from a simulator input and output trace. The combination of using the wavelet transform on the simulator error signal and then determining the coherence of the input and output signal using the wavelet coherence method allows for first identifying (time-frequency)-regions containing a motion cueing error, and then determining the type of error present.

5. *How can we verify and validate that the algorithm functions correctly?*
Verification of the algorithm designed has been performed by constructing individual idealised motion cueing errors, as shown in the top left of Figures 5-3 through 5-9. The algorithm has been given these input and output signals as input and it has been verified that the algorithm identifies the correct type of error at the correct points in time, without falsely identifying other errors.

Chapter 6

Conclusions & Discussion

In this preliminary thesis the suitability of wavelet analysis for the detection of motion cueing error types from simulator input and output signals has been investigated. This chapter serves as a round-up of this report, drawing conclusions from the work done in section 6-1. Following this summary results of this report are discussed in section 6-2, with shortcomings of the work done so far highlighted. These shortcomings serve as input for the discussion of future work in section 6-3, which is followed by an experiment proposal for validation of the algorithm designed in section 6-4.

6-1 Conclusions

In section 5-2 it has been shown that individual idealised motion cueing errors can be correctly identified using the wavelet-based identification scheme proposed in section 5-1, without other motion cueing error types being falsely identified. Additionally the time of occurrence of these errors is identified correctly. Because of this finding it is concluded that wavelet analysis holds promise for the identification of scaling errors and false cues in simulator input and output traces.

It has been found the algorithm is unable to identify a typical simulator phase error, owing to the ambiguity of the definition of this error and the definitions of Type II scaling errors and Type II false cues. As a result the algorithm identifies a combination of the latter two rather than a phase error.

Detection of motion cueing error types is best done using complex wavelets with relatively good time resolution, such as the complex Morlet wavelet (in either the time or frequency domain), the time-based complex Gaussian wavelet or the frequency-based Paul wavelet.

6-2 Discussion

The positive findings of this preliminary thesis should be accompanied by a number of considerations, which are gone through in this discussion section. These considerations include assumptions and simplifications made, and limitations of the current results.

Phase error detection - As noted in section 2-2 false cues are most detrimental to the simulator fidelity perceived by humans, followed by scaling errors. It is for this reason that the algorithm's current inability to detect a short term phase error (instead detecting a combination of a Type II scaling error and Type II false cue due to the fact that phase errors which don't span multiple frequency cycles cause signal coherence to drop) is not deemed too much of a problem, since this algorithm behaviour does correspond to the definitions of the latter two error types.

***dt*- and α-dependency** - As noted in subsection 3-4-2 the outcome of the wavelet significance test proposed by Torrence & Compo depends on the sampling frequency of the simulator input and output signal (with a higher sample rate leading to larger regions being marked as statistically significant). This dependency can be partly covered by adjusting the α-value of the significance test, this lag-1 autocorrelation coefficient in itself however is arbitrarily picked at present.

For geophysical processes (for which this significance test was developed) determining a suitable α-value is possible from signal autocorrelation (Torrence & Compo, 1998), for simulator signals (which in general are non-periodic) this method is unsuitable. Trial and error will thus be needed to determine proper α-values for the detection of motion cueing errors in actual simulator signals.

Use of idealised errors - The idealised individual motion cueing errors used in chapter 5 represent a very limited case of the errors to be encountered in true simulator input/output signals. In reality multiple errors could be combined (with Type I scaling errors and Type III-2 false cues being present most of the time), meaning the algorithm should be capable of discriminating all error types simultaneously. Looking forward to true simulator signals, a number of problems are foreseen:

1. In case a true simulator signal features a Type III-1 false cue amidst a continuous Type III-2 false cue due to simulator vibrations, the algorithm as currently implemented won't be able to discriminate the Type III-1 false cue from the Type III-2 false cue (since high-frequency content is continuously present). Alternatively (as shown in Figure 5-8) a small significant region of a Type III-2 false cue will show up as a Type III-1 false cue in the algorithm output.
2. All idealised errors (apart from the Type I scaling error) presented in section 5-2 featured no overall signal scaling. It is known that in most simulators such overall scaling is applied, leading to errors being present to some extent most of the time. It remains to be seen how well the wavelet significance test is able to identify individual error regions in case multiple errors are present in the time signal.
3. From CMS experiments at the MPI it is known that real simulator output signals feature significant noise due to simulator vibrations. It is expected that this noise

will at times (in case of small output values) lead to the simulator output signal falsely switching sign (causing the power ratio to switch sign) or exceeding perceptual thresholds. Both effects will lead to false identification of a Type I or Type II false cue, when in reality only a Type III-2 error should be detected.

4. Simulating idealised motion cueing errors has shown that the coherence between a simulator's input and output signal drops quickly in case any error other than a Type I scaling error is present. It remains to be seen therefore whether Type I scaling errors are still correctly identified for noisy simulator output.

Number of sensory channels - So far, only a single pair of motion cueing signals has been analysed at a time to identify the error(s) between the two. In a simulator however subjects experience six motion signals simultaneously (a_x, a_y, a_z, ω_x, ω_y and ω_z), with each sensory channel possible containing errors.

Use of binary thresholds - In order to prevent false-positive error detections by the detection algorithm proposed in this preliminary thesis human perceptual threshold levels have been implemented in the algorithm. The use of these thresholds should mimic the tendency of humans not to detect false cues when the output is at a sub-threshold level (although it is substantially larger than the input) or to detect a Type II scaling error in case an above-threshold input is accompanied by a sub-threshold output (even though the two may be within the coherence zone).

It is known that in reality perceptual thresholds are not so much a binary concept, but are a Bayesian process instead (depending on the stimulus the chance of detecting it increases) (Nash et al., 2016, p100). As a result the short Type II scaling error indicated in Figure 5-4 (which is the result of the input surpassing the threshold level slightly earlier than the output value) will not show up as such in Continuous Ratings provided by subjects. Additionally this human behaviour will cause the timing at which a subject indicates the onset of an error to fluctuate.

6-3 Future work

Because of the limitations of the work done so far (as discussed in the previous section) a number of amendments to and checks on the current algorithm are required before a simulator validation experiment (as described in section 6-4) can be carried out. Of particular importance is to test the current algorithm using realistic simulator signals (possibly using data of both the CMS at the Max Planck Institute for Biological Cybernetics and the SRS at the Delft University of Technology, which represent two different kinds of simulator) in order to see to what extent the problems foreseen in the previous section (and other unforeseen problems) are encountered.

Besides these checks a number of modifications are intended to be made:

1. In order to deal with the overall scaling of motion input signals (see point 2 in section 6-2) the lag-1 autocorrelation coefficient α needs to be tuned such that all regions of significant error due to Type I scaling errors and Type III false cues (which both can be ever-present) are identified separately from regions of significant other errors (since the

distinction between these is based on the identification of separate regions of significant error).

2. In order to prevent point 3 of section 6-2 from taking place output signal denoising using the inverse CWT is proposed. This will be done by removing the high-frequency components identified as Type III-2 errors from the output CWT spectrum and then inversely transforming this spectrum to obtain the denoised output signal.

3. In order to avoid short-term error indications like the Type II scaling error indication in Figure 5-4 at $t = 4.1$s a minimum duration of indicated errors could be imposed (to account for humans also requiring a certain exposure to an error). Such an adjustment will only be made in case it is beneficial for future uses of this algorithm.

4. To deal with multiple sensory channels an easy-to-interpret visual representation of errors is to be made which clearly shows what errors are present at what time and in what sensory channel. This representation will be made by extending the bottom-right plots in Figures 5-3 through 5-9, using colours indicating the different motion channels.

Following verification of the correct functioning of the error identification algorithm using real simulator signals, the algorithm will then be validated (showing that the algorithm indicates errors at points in time at which subjects also indicate errors) through a simulator experiment, the set-up of which is discussed in the next section.

6-4 Validation experiment proposal

In this section the set-up of a proposed validation experiment for the motion cueing error detection algorithm is discussed. Following the experiment's objectives discussed in subsection 6-4-1 and the hypotheses provided in subsection 6-4-2, the experimental set-up is discussed in subsection 6-4-3. Finally the proposed data analysis is discussed in subsection 6-4-4.

6-4-1 Experiment objectives

This experiment has two objectives, the first of which is to validate that the motion cueing error detection algorithm presented in this preliminary thesis detects errors at (or shortly before) instances in time at which subjects also indicate a motion cueing error to be present. It should also be validated that the algorithm does not signal false positives. At the very least it should be validated that the algorithm does not miss any errors reported by human subjects (and thus avoids false negatives).

The second objective of this experiment is to investigate whether the different types of errors are rated significantly different for an equal error magnitude (which would confirm the need to be able to discern them).

6-4-2 Hypothesis

Prior to the experiment the following hypotheses are set up, which will be tested during the experiment:

1. The motion cueing error detection algorithm designed will detect an error at instances in time at which subjects indicate an error, and will not detect errors when subjects indicate no error to be present.
2. The motion cueing error detection algorithm will detect errors earlier than subjects.
3. The motion cueing error types will, from worst to least bad, be ranked Type III-1 false cues - Type I false cues - Type II false cues - Phase errors - Type III-2 false cues - Type II scaling errors - Type I scaling errors.

The first hypothesis is set up to test the performance of the detection algorithm in terms of the presence of false positive and false negative detections. The second hypothesis serves to comment on the average time delay between an error taking place (as detected by the algorithm) and a human indicating this error. This time delay can then be used when analysing Continuous Ratings provided by test subjects, as described in section 1-1.

The third hypothesis finally serves to fulfill the second experimental objective and comment on which motion cueing error types should be treated separately in the evaluation of a simulator's motion cueing quality.

6-4-3 Experimental set-up

The hypotheses described above will be tested through a simulator experiment performed at the MPI, details of this experiment are described below:

General description - In this validation experiment subjects will experience a passive car drive in a simulator and will be asked to rate the incoherence between the visual images they are presented and the motion they feel. This incoherence rating will then be compared to the detection algorithm output. The car drive features motion cueing errors of different types, allowing to evaluate the influence of each motion cueing error type.

Simulator - The experiment will be carried out in the CyberMotion Simulator at the Max Planck Institute for Biological Cybernetics. The occupant cabin of this simulator is equipped with two overlaying projectors to display visual images and will be equipped with a SensoDrive rotary electrical knob to allow subjects to provide an incoherence rating. This rating will also be displayed by the projectors to indicate to subjects their current rating.

Manoeuvres - In order to simplify the comparison between the six-dimensional algorithm output (which can detect motion cueing errors per motion channel) and the one-dimensional output provided by subjects (who integrate the errors in all six motion channels into one overall rating) a car manoeuvre is chosen which features motion on only a selection of motion channels. For this reason a car drive through alternating right and left corners (with pieces of straights in between) at constant velocity (a track made using the vehicle simulation programme CarSim provided by Mechanical Simulation, US) is chosen as simulated manoeuvre.

The resulting input motion for each corner into the MCA is a lateral acceleration and yaw rate with a profile similar to the input signal shown in Figures 5-3 through 5-9. Choosing this input profile allows for modifying the simulator motion such that outputs such as those shown in Figures 5-3 through 5-9 are presented to the subject. The choice of a curve-driving manoeuvre also allows for relating the results of this research to previous research on the perceived fidelity during curve driving in a simulator (Grant et al., 2003; Valente Pais, Wentink, van Paassen & Mulder, 2009; Feenstra, Wentink, Correia Grácio & Bles, 2009; Damveld, Wentink, van Leeuwen & Happee, 2012; Lakerveld et al., 2016).

Independent variable - During each corner either motion perfectly coherent to the visuals is presented, or simulator motion containing a motion cueing error is presented to the subject. By varying this motion cueing error type the effect of each error type on perceived simulator fidelity can be investigated.

Measurements/Dependent variable - During each car manoeuvre a subject indicates his or her rating of the perceived instantaneous incoherence between the presented visuals and the simulator motion by turning the rotary knob installed in the cabin (a process described by Cleij et al. (Cleij et al., 2015)), producing a rating between zero ('fully coherent') and one ('fully incoherent'). Measuring this rating allows for commenting on the effect of the error type on a subject's perceived simulator fidelity. The straight road sections in between curves (in which no acceleration/rotation input is given) allow subjects to decrease their rating back to zero, in turn allowing for analysing all error types individually.

Additionally subjects will be asked to comment on any experienced simulator sickness after their simulator run, allowing for exclusion of data in case subjects indicate to have suffered from severe simulator sickness.

Subjects - In this experiment 12 subjects with car driving experience and at least some motion simulation experience in the CMS will be selected. Previous experience with the Continuous Rating method is not necessarily required as this can be acquired during the training phase.

Training & Experiment phase - In order to get acquainted with the type of motion provided by the CMS and the method of continuous rating subjects will first do a training run through a number of corners. This training run is especially important for the calibration of a subject's internal incoherence grading scale, as subjects will be instructed to assign a rating of 1 to the worst incoherence encountered.

To allow subjects to calibrate their own scale all motion cueing error types (and the one-to-one motion) will feature twice during the training phase, once in a left turn and once in a right turn. The order of these errors will be randomised.

Following this training phase the subjects will experience all motion cueing error types four times during the experiment (twice in a left turn and twice in a right turn), again in randomised order. As mentioned previously all turns feature one after the other, ensuring subjects don't have to re-calibrate their internal incoherence scale.

6-4-4 Data analysis

Following the experiment the continuous ratings given by all subjects (during all repetitions of the experiment) will be averaged per car manoeuvre and MCA setting. This average incoherence rating will then be compared to the output of the error detection algorithm to assess the algorithm's performance and comment on hypotheses 1 and 2.

Additionally the distributions of maximum continuous ratings during a corner will be investigated to determine what motion cueing error types differ significantly in terms of experience incoherence, allowing for commenting on hypothesis 3.

Appendix A

Continuous Wavelet Transform Plots

In this appendix the Continuous Wavelet Transform of a test signal (generated with $\Delta t = 0.01$s) for the various wavelet types discussed in subsection 3-3-2 is presented. The test signal consists of various typical signal contents: Two peaks, a constant sine superimposed with a higher-frequency sine wave, a step input and a chirp signal with increasing frequency. For all wavelet types the effect of different parameters is illustrated.

All plots are shown with a reversed y-axis (in order to have high-frequency CWT content 'point' at the signal in time) and a Region of Influence (as discussed in subsection 3-4-1) has been included with all figures.

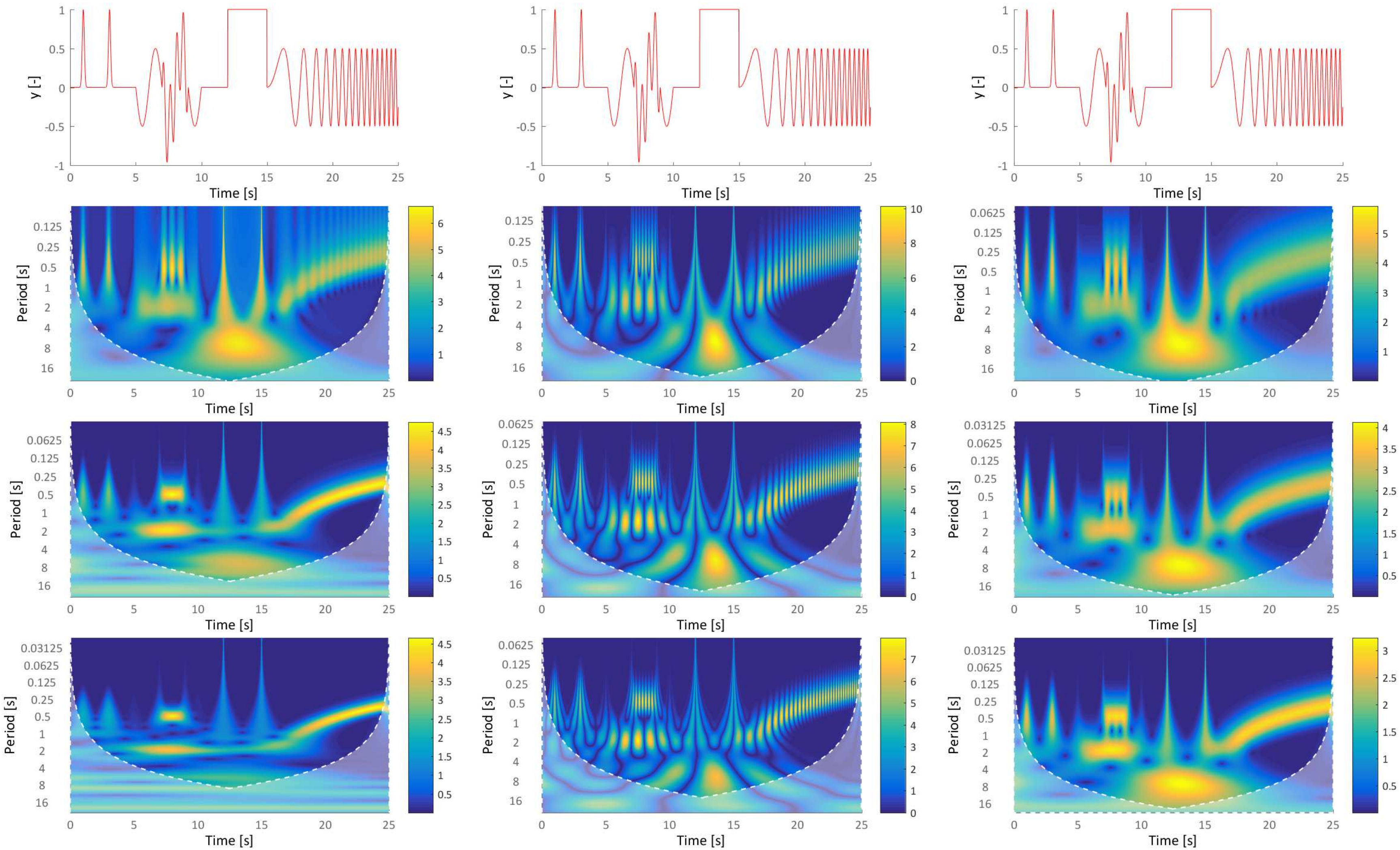


Figure A-1: Wavelet transforms of a test signal using frequency-based complex Morlet wavelets (left column, $\omega_0 = 2rad/s, \omega_0 = 4rad/s, \omega_0 = 6rad/s$ from top to bottom), real DOG wavelets (middle column, $n = 2, n = 4, n = 6$ from top to bottom) and complex Paul wavelets (right column, $m = 2, m = 4, m = 6$ from top to bottom). Shown is the absolute value of the CWT, the Regions of Influence (ROI) are shaded white.

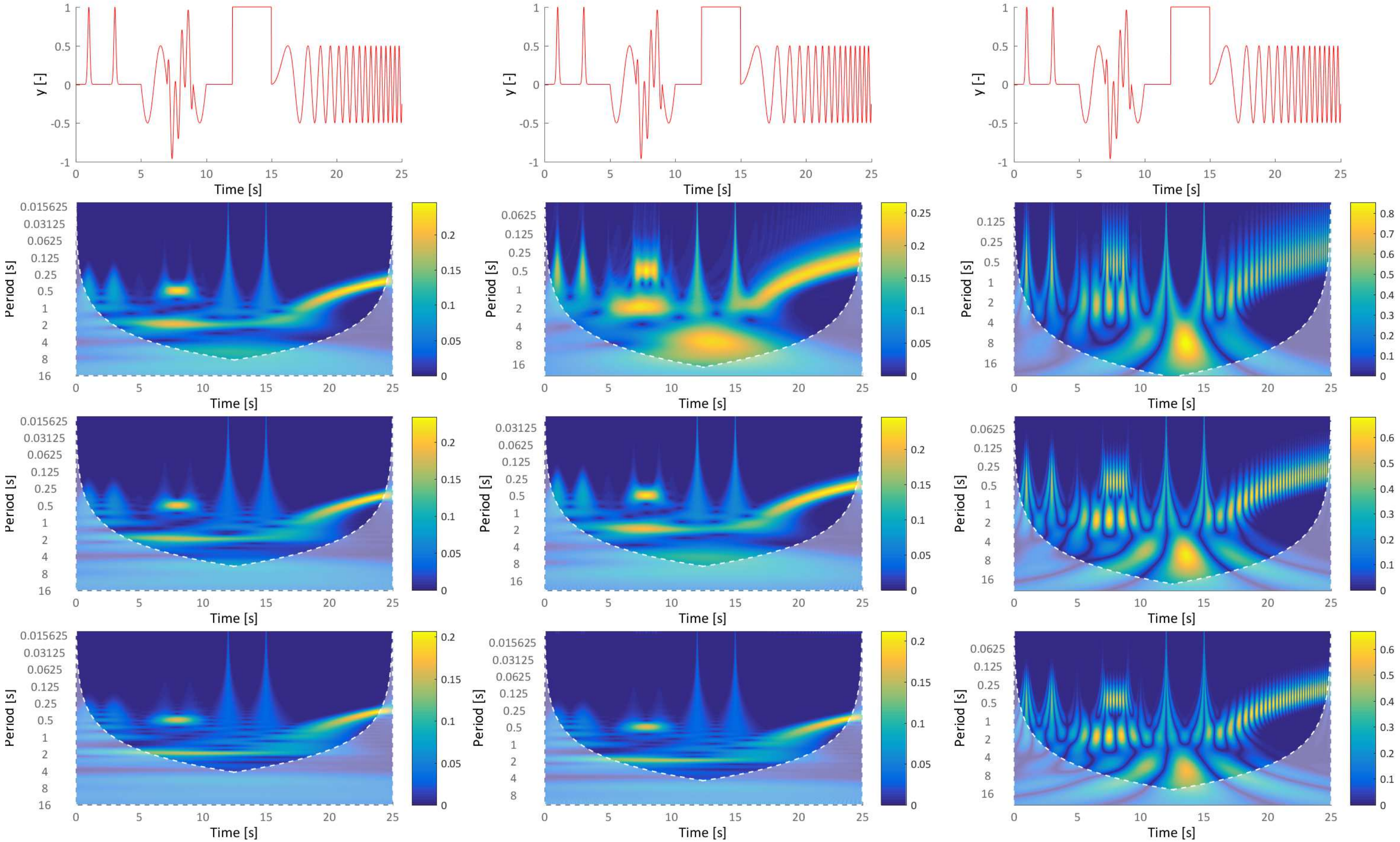


Figure A-2: Wavelet transforms of a test signal using complex time-based Morlet wavelets (left column ($f_c = 1.5Hz$ and $f_b = 1, f_b = 2, f_b = 4$ from top to bottom) and middle column ($f_b = 2$ and $f_c = 0.5Hz, f_c = 1Hz, f_c = 2Hz$ from top to bottom)) and time-based real Gaussian wavelets (right column, $n = 2, n = 4, n = 6$ from top to bottom). Shown is the absolute value of the CWT, ROIs are shaded white.

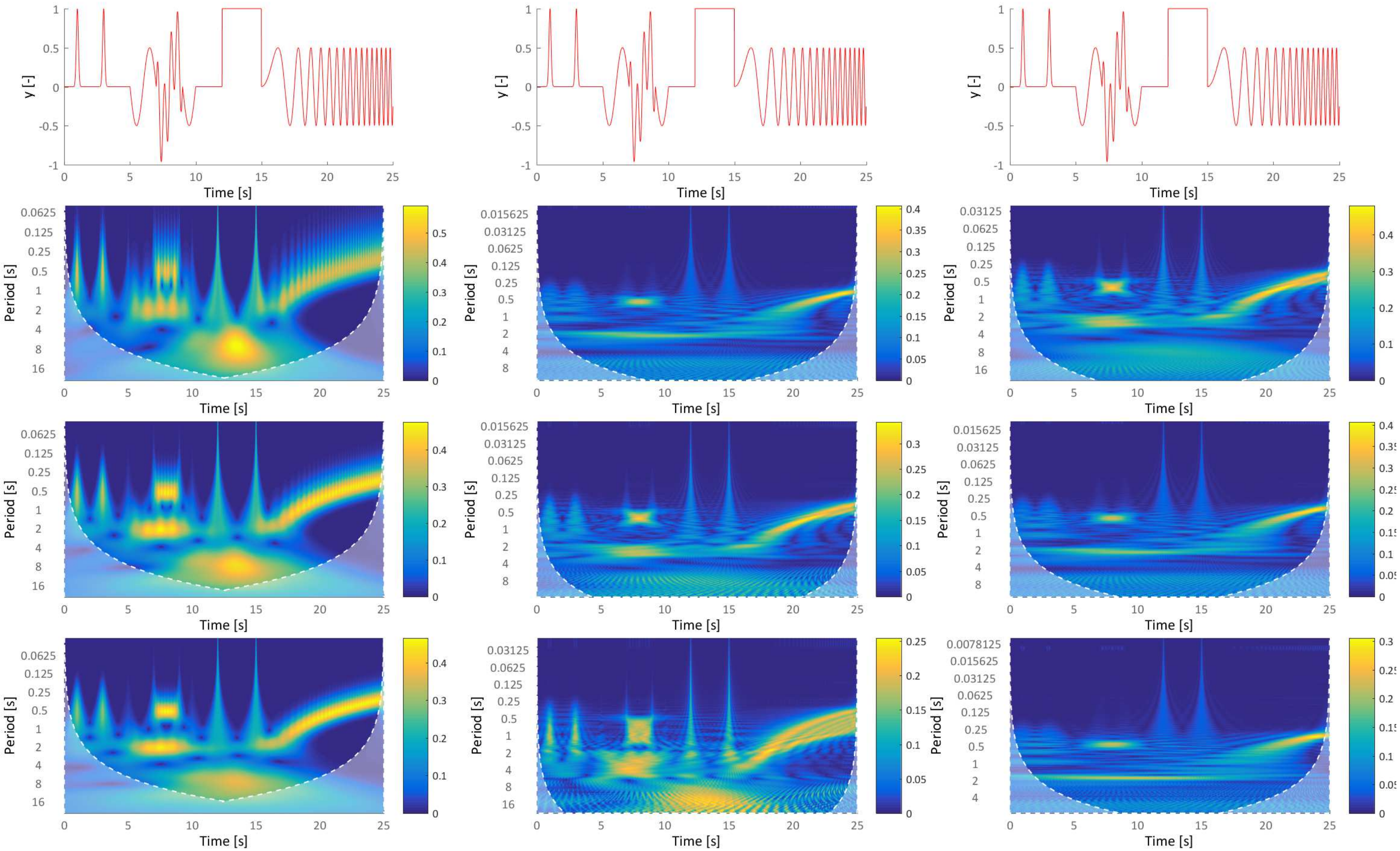


Figure A-3: Wavelet transforms of a test signal using time-based complex Gaussian wavelets (left column, $n = 2, n = 4, n = 6$ from top to bottom) and complex Shannon wavelets (middle column ($f_c = 2Hz$ and $f_b = 0.5Hz, f_b = 1Hz, f_b = 2Hz$ from top to bottom) and right column ($f_b = 0.5Hz$ and $f_c = 1Hz, f_c = 2Hz, f_c = 3Hz$ from top to bottom)). Shown is the absolute value of the CWT, ROIs are shaded white.

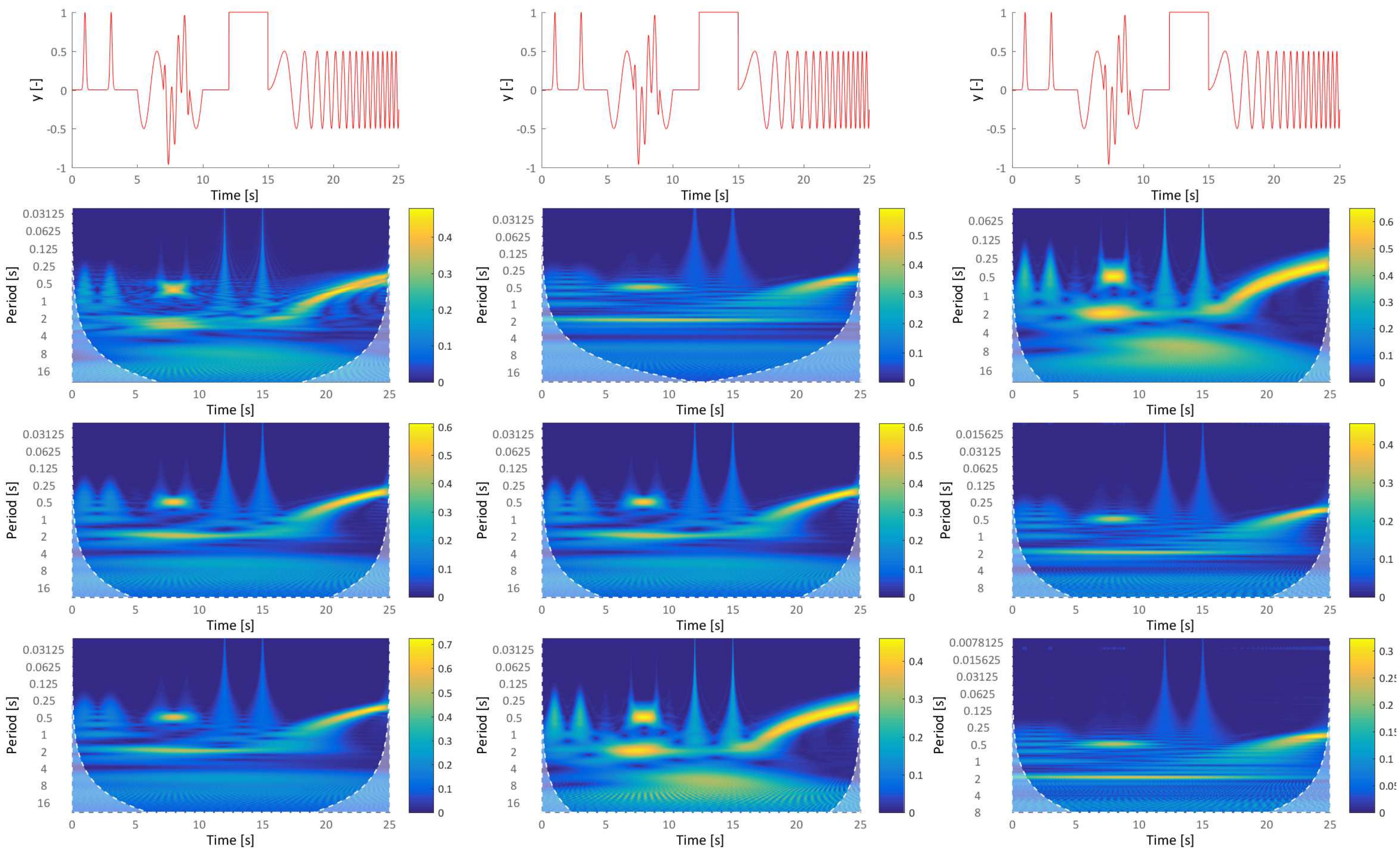


Figure A-4: Wavelet tranforms of a test signal using time-based complex Frequency B-spline wavelets. In the left column $f_b = 0.5Hz, f_c = 1Hz$ and $M = 1, M = 2, M = 3$ (top to bottom), in the middle column $M = 2, f_c = 1Hz$ and $f_b = 0.2Hz, f_b = 0.5Hz, f_b = 1Hz$ (top to bottom) and in the right column $M = 2, f_b = 0.5Hz$ and $f_c = 1Hz, f_c = 2Hz, f_c = 3Hz$ (top to bottom). Shown is the absolute value of the CWT, ROIs are shaded white.

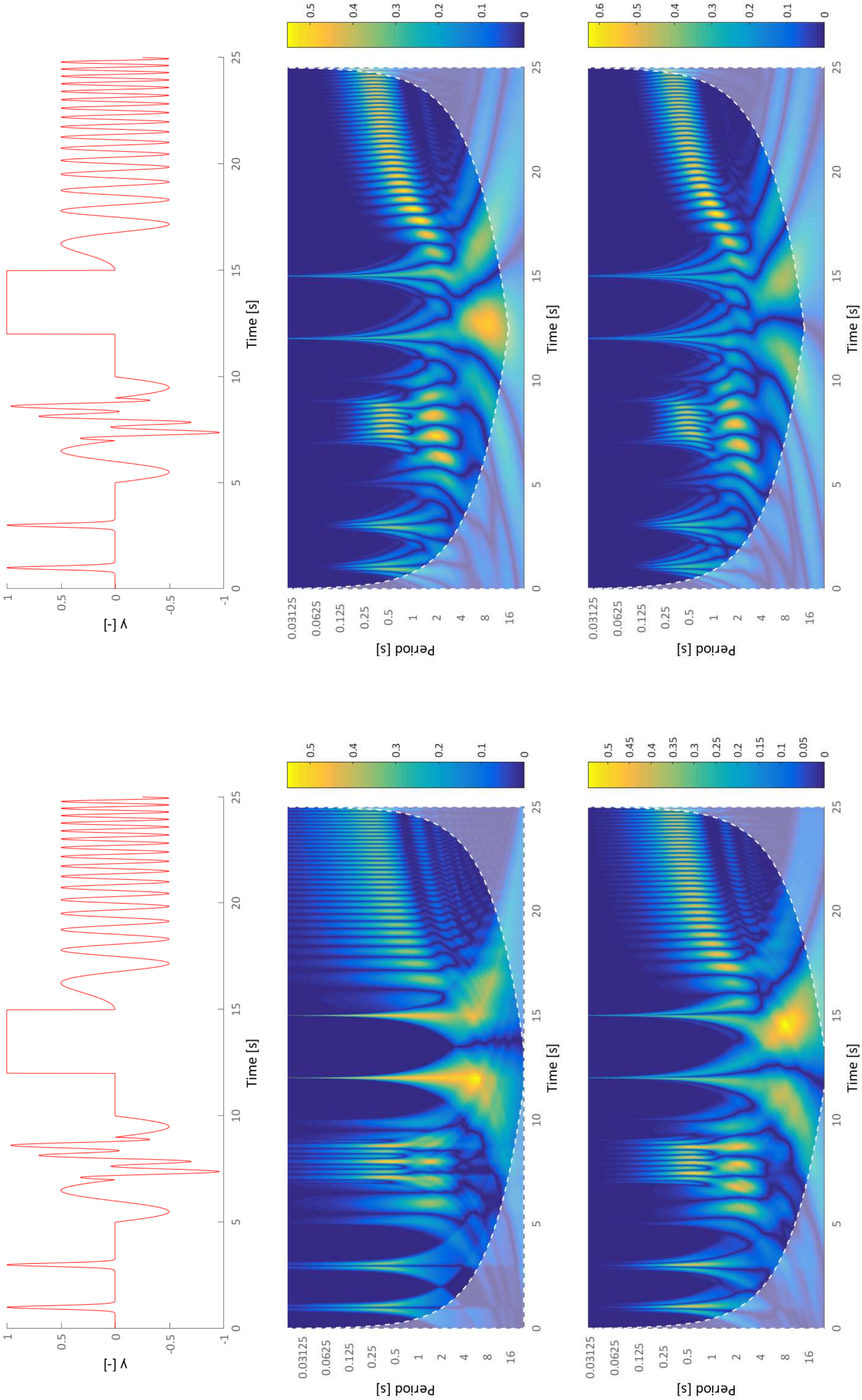


Figure A-5: Wavelet transforms of a test signal using the real time-based Haar wavelet (top left), Daubechies2, Daubechies4 and Daubechies6 wavelet (bottom left, top right and bottom right respectively). Shown is the absolute value of the CWT, ROIs are shaded white.

References

Advani, S. & Hosman, R. (2006). Towards standardising high-fidelity cost-effective motion cueing in flight simulation. In *Royal aeronautical society conference on: Cutting costs in flight simulation. balancing quality and capability. london, nov. 7-8, 2006.*

Advisory Group for Aerospace Research & Development. (1980). *Fidelity of simulation for pilot training.* Harford House, 7-9 Charlotte Street, London: Technical Editing and Reproduction Ltd.

Baarspul, M. (1990). A review of flight simulation techniques. *Progress in Aerospace Sciences*, *27*(1), 1-120. doi: 10.1016/0376-0421(90)90006-6

Beckers, N., Pool, D., Valente Pais, A., van Paassen, M. & Mulder, M. (2012). Perception and behavioral phase coherence zones in passive and active control tasks in yaw. In *Modeling and simulation technologies conference, Minneapolis, Minnesota, Aug. 13-16, 2012.* doi: 10.2514/6.2012-4794

Berthoz, A., Bles, W., Bülthoff, H., Correia Grácio, B., Feenstra, P., Filliard, N., ... Wentink, M. (2013). Motion Scaling for High-Performance Driving Simulators. *IEEE Transactions on Human-Machine Systems*, *43*(3), 265-276. doi: 10.1109/TSMC.2013.2242885

Casas, S., Inmaculada, C., Riera, J. & Fernández, M. (2015). Motion-cuing algorithms: Characterization of users perception. *Human Factors*, *57*(1), 144-162. doi: 10.1177/0018720814538281

Chang, C. & Glover, G. (2010). Time-frequency dynamics of resting-state brain connectivity measured with fMRI. *Neuroimage*, *50*(1), 81-98. doi: 10.1016/j.neuroimage.2009.12.011

Cleij, D., Venrooij, J., Pretto, P., Pool, D., Mulder, M. & Bülthoff, H. (n.d.). Continuous subjective rating of perceived motion incoherence during driving simulation. *IEEE Transactions on Human-Machine Systems*, *Submitted for publication.*

Cleij, D., Venrooij, J., Pretto, P., Pool, D., Mulder, M. & Bülthoff, H. (2015). Continuous rating of perceived visual-inertial motion incoherence during driving simulation. In *Driving simulation conference & exhibition, Tübingen, Germany, Sep. 16-18, 2015.*

Cooper, G. & Cowan, D. (2008). Comparing time series using wavelet-based semblance analysis. *Computers & Geosciences*, *34*(1), 95-102. doi: 10.1016/j.cageo.2007.03.009

Correia Grácio, B., Valente Pais, A., van Paassen, M., Mulder, M., Kelly, L. & Houck, J. (2013). Optimal and coherence zone comparison within and between flight simulators. *Journal of Aircraft*, *50*(2), 493-507. doi: 10.2514/1.C031870

Damveld, H., Wentink, M., van Leeuwen, P. & Happee, R. (2012). Effects of motion cueing on curve driving. In *DSC 2012 Europe: Driving Simulation Conference & Exhibition, Paris, France, Sep. 6-7, 2012.*

Daubechies, I. (1992). *Ten lectures on wavelets.* Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.

de Winter, J., Dodou, D. & Mulder, M. (2012). Training effectiveness of whole body flight simulator motion: A comprehensive meta-analysis. *The International Journal of Aviation Psychology*, *22*(2), 164-183. doi: 10.1080/10508414.2012.663247

Feenstra, P., Wentink, M., Correia Grácio, B. & Bles, W. (2009). Effect of simulator motion space on realism in the Desdemona simulator. In *DSC 2009 Europe: Driving Simulation Conference & Exhibition, Monaco, Feb. 4-6, 2009.*

Fischer, M. (2007). A survey of state-of-the-art motion platform technology and motion cueing algorithms. In *2nd motion simulator conference, braunschweig, germany, august, 2007.*

Fischer, M. (2009). *Motion-cueing-algorithmen für eine realitätsnahe bewegungssimulation* (PhD Thesis). Technische Universität Braunschweig - Fakultät für Maschinenbau, Germany.

Gilman, D., Fuglister, F. & Mitchell, J. (1962). On the power spectrum of red noise. *Journal of the Atmospheric Sciences*, *20*, 182-184. doi: 10.1175/1520-0469(1963)020⟨0182:OTPSON⟩2.0.CO;2

Grant, P. (1995). *The development of a tuning paradigm for flight simulator motion drive algorithms* (PhD Thesis). University of Toronto - Department of Aerospace Science and Engineering, Canada.

Grant, P., Blommer, M., Artz, B. & Greenberg, J. (2003). Analysing classes of motion drive algorithms based on paired comparison techniques. In *Proceedings of the Driving Simulation Conference (DSC) North America, Dearborn, USA, Oct. 8-10, 2003.*

Grant, P. & Reid, L. (1997). Motion washout filter tuning: Rules and requirements. *Journal of Aircraft*, *34*(2), 145-151. doi: 10.2514/2.2158

Grinsted, A., Moore, J. & Jevrejeva, S. (2004). Application of the cross wavelet transform and wavelet coherence to geophysical time series. *Nonlinear processes in geophysics*, *11*, 561-566. doi: 10.5194/npg-11-561-2004

Groen, E. & Bles, W. (2004). How to use body tilt for the simulation of linear self motion. *Journal of Vestibular Research*, *14*(5), 375-385.

Heerspink, H., Berkouwer, W., Stroosma, O., van Paassen, M., Mulder, M. & Mulder, J. (2005). Evaluation of vestibular thresholds for motion detection in the Simona Research Simulator. In *AIAA Modeling and Simulation Technologies Conference and Exhibit, San Francisco, California, Aug. 15-18, 2005.* doi: 10.2514/6.2005-6502

Hosman, R. & Advani, S. (2016). Design and evaluation of the objective motion cueing test and criterion. *The Aeronautical Journal*, *120*(1227), 873891. doi: 10.1017/aer.2016.35.

Jammazi, R. & Aloui, C. (2015). On the interplay between energy consumption, economic growth and CO_2 emission nexus in the GCC countries: A comparative analysis through wavelet approaches. *Renewable and Sustainable Energy Reviews*, *51*(1), 1737-1751. doi: 10.1016/j.rser.2015.07.073

Jonah, O., de Paula, E., Muella, M., Dutra, S., Kherani, E., Negreti, P. & Otsuka, Y. (2015). TEC variation during high and low solar activities over South American sector.

Jounal of Atmospheric and Solar-Terrestrial Physics, *135*(1), 22-35. doi: 10.1016/j.jastp.2015.10.005

Jonik, P., Valente Pais, A., van Paassen, M. & Mulder, M. (2011). Phase coherence zones in flight simulation. In *Modeling and simulation technologies conference, Portland, Oregon, Aug. 8-11, 2011.* doi: 10.2514/6.2011-6555

Kennedy, R., Lane, N., Berbaum, K. & Lilienthal, M. (1993). Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The International Journal of Aviation Psychology*, *3*(3), 203-220. doi: 10.1207/s15327108ijap0303_3

Ko, S. (2012). *Investigation of simulator motion drive algorithms for airplane upset simulation* (Masters Thesis). University of Toronto, Canada.

Kristoufek, L. (2015). What are the main drivers of the bitcoin price? Evidence from wavelet coherence analysis. *PLoS ONE*, *10*(4), e0123923. (Note: Published online, open access) doi: 10.1371/journal.pone.0123923

Lakerveld, P., Damveld, H., Pool, D., van der El, K., van Paassen, M. & Mulder, M. (2016). The effects of yaw and sway motion cues in curve driving simulation. In *13th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine Systems, Kyoto, Japan, Aug. 30 - Sep. 2, 2016.*

Naseri, A. & Grant, P. (2005). An improved adaptive motion drive algorithm. In *AIAA Modelling and Simulation Technologies Conference and Exhibit 2005, San Francisco, USA, Aug. 15-18, 2005.*

Nash, C., Cole, D. & Bigler, R. (2016). A review of human sensory dynamics for application to models of driver steering and speed control. *Biological Cybernetics*, *110*(2), 91-116. doi: 10.1007/s00422-016-0682-x

Papaioannou, G., Dikaiakos, C., Evangelidis, G., Papaioannou, P. & Georgiadis, D. (2015). Co-movement analysis of Italian and Greek electricity market wholesale prices by using a wavelet approach. *Energies*, *8*(10), 11770-11799. doi: 10.3390/en81011770

Pei, Q., Zhang, D., Li, G. & Lee, H. (2015). Climate change and the macroeconomic structure in pre-industrial Europe: New evidence from wavelet analysis. *PLoS ONE*, *10*(6), e0126480. (Note: Published online, open access) doi: 10.1371/journal.pone.0126480

Pool, D., Harder, G. & van Paassen, M. (2016). Effects of simulator motion feedback on training of skill-based control behavior. *Journal of Guidance, Control and Dynamics*, *39*(4), 889-902. doi: 10.2514/1.G001603

Reid, L. & Nahon, M. (1986). *Flight simulation motion-base drive algorithms: Part 1 - Developing and testing the equations* (UTIAS Report, no. 296). University of Toronto Institute for Aerospace Studies.

Reid, L. & Nahon, M. (1987). *Flight simulation motion-base drive algorithms: Part 3 - Pilot evaluations* (UTIAS Report, no. 319). University of Toronto Institute for Aerospace Studies.

Robuffo Giordano, P., Masone, C., Tesch, J., Breidt, M., Pollini, M. & Bülthoff, H. (2010). A novel framework for closed-loop robotic motion simulation - Part I: Inverse kinematics design. In *2010 IEEE International Conference on Robotics and Automation, Anchorage, Alaska, May 3-7, 2010.* doi: 10.1109/ROBOT.2010.5509647

Rolfe, J. & Caro, P. (1982). Determining the training effectiveness of flight simulators: Some basic issues and practical developments. *Applied Ergonomics*, *13*(4), 243-250. doi: 10.1016/0003-6870(82)90063-1

Salisbury, I. & Limebeer, D. (2016). Optimal motion cueing for racing cars. *IEEE Transactions on Control Systems Technology*, *24*(1), 200-215. doi: 10.1109/TCST.2015

.2424161

Sanders, A. (1991). Simulation as a tool in the measurement of human performance. *Ergonomics*, *34*(8), 995-1025. doi: 10.1080/00140139108964844

Schmidt, S. & Conrad, B. (1970). *Motion drive signals for piloted flight simulators* (Technical report contract NASA CR-1601). National Aeronautics and Space Administration, Ames Research Center.

Schroeder, J. (1999, July). *Helicopter flight simulation motion platform requirements* (Tech. Rep. No. NASA/TP-1999-208766). Stanford University. Dept. of Aeronautics and Astronautics.

Sinacori, J. (1977). *The determination of some requirements for a helicopter flight simulation facility* (Tech. Rep. No. CR-152066). NASA.

Stahl, K., Abdulsamad, G., Leimbach, K. D. & Vershinin, Y. A. (2014, Oct). State of the art and simulation of motion cueing algorithms for a six degree of freedom driving simulator. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (p. 537-541). doi: 10.1109/ITSC.2014.6957745

Teolis, A. (1999). *Computational signal processing with wavelets.* Cambridge, MA 02139, USA: Birkhäuser Boston. doi: 10.1007/978-1-4612-4142-3

Teufel, H., Nusseck, H.-G., Beykirch, K., Butler, J., Berger, M. & Bülthoff, H. H. (2007). MPI Motion Simulator: Development and Analysis of a Novel Motion Simulator. In *Proceedings of the AIAA Modeling and Simulation Technologies Conference and Exhibit, Hilton Head, South Carolina, Aug. 20-23, 2007.* doi: 10.2514/6.2007-6476

Torrence, C. & Compo, G. (1998). A practical guide to wavelet analysis. *Bulletin of the American Meteorological Society*, *79*(1), 61-78. doi: 10.1175/1520-0477(1998)079⟨0061:APGTWA⟩2.0.CO;2

Torrence, C. & Webster, P. (1999). Interdecadal changes in the ensomonsoon system. *Journal of Climate*, *12*(8), 2679-2690. doi: 10.1175/1520-0442(1999)012⟨2679:ICITEM⟩2.0.CO;2

Valente Pais, A., van Paassen, M. & Mulder, M. (2009). Perception coherence zones in flight simulation. In *Proceedings of the AIAA Modeling and Simulation Technologies Conference, Chicago, Illinois, Aug. 10-13, 2009.* doi: 10.2514/6.2009-6242

Valente Pais, A., Wentink, M., van Paassen, M. & Mulder, M. (2009). Comparison of three motion cueing algorithms for curve driving in an urban environment. *Presence*, *18*(3), 200-221. doi: 10.1162/pres.18.3.200

Valle-Levinson, A., Castro, L., Cáceres, M. & Pizarro, O. (2014). Twilight vertical migrations of zooplankton in a Chilean fjord. *Progress in Oceanography*, *129*, 114-124. doi: 10.1016/j.pocean.2014.03.008

Venrooij, J., Pretto, P., Katliar, M., Nooij, S., Nesti, A., Lächele, M., … Bülthoff, H. (2015). Perception-based motion cueing: Validation in driving simulation. In *DSC 2015 Europe: Driving Simulation Conference & Exhibition, Tübingen, Germany, Sep. 16-18, 2015.*

Verspecht, T. (2011). *Identification of Time Variant Neuromuscular Admittance* (Masters Thesis). Delft University of Technology - Faculty of Aerospace Engineering, Kluyverweg 1, Delft, The Netherlands.

Yochum, M., Renaud, C. & Jacquir, S. (2016). Automatic detection of P, QRS and T patterns in 12 leads ECG signal based on CWT. *Biomedical Signal Processing and Control*, *25*, 46-52. doi: 10.1016/j.bspc.2015.10.011

Zaichik, L., Rodchenko, V., Rufov, I., Yahin, Y. & White, A. (1999). Acceleration perception. In *Modeling and simulation technologies conference and exhibit, Portland, Oregon, 1999*

(p. 512-520). doi: 10.2514/6.1999-4334