Comparison between filter- and optimization-based motion cueing in the Daimler Driving Simulator

Venrooij J [1], Cleij D [1], Katliar M [1], Pretto P [1], Bülthoff HH [1], Steffen D [2], Hoffmeyer FW [2], Schöner H-P [2]

(1) Max Planck Institute for Biological Cybernetics, Tübingen, Germany,
e-mail: {joost.venrooij,diane.cleij,mikhail.katliar,paolo.pretto,heinrich.buelthoff}@tuebingen.mpg.de
(2) Daimler AG, Driving Simulators Dept. RD/FFS, Sindelfingen, Germany,
e-mail: {dennis.steffen,friedrich.hoffmeyer,hans-peter.schoener}@daimler.com

Abstract *– This paper describes a driving simulation experiment, executed on the Daimler Driving Simulator (DDS), in which a filter-based and an optimization-based motion cueing algorithm (MCA) were compared using a newly developed motion cueing quality rating method. The goal of the comparison was to investigate whether optimization-based MCAs have, compared to filter-based approaches, the potential to improve the quality of motion simulations. The paper describes the two algorithms, discusses their strengths and weaknesses and describes the experimental methods and results. The MCAs were compared in an experiment where 18 participants rated the perceived motion mismatch, i.e., the perceived mismatch between the motion felt in the simulator and the motion one would expect from a drive in a real car. The results show that the quality of the motion cueing was rated better for the optimization-based MCA than for the filter-based MCA, indicating that there exists a potential to improve the quality of the motion simulation with optimization-based methods. Furthermore, it was shown that the rating method provides reliable and repeatable results within and between participants, which further establishes the utility of the method.*

Keywords: *motion cueing; optimization-based; filter-based; continuous rating;*

Introduction

Motion cueing is the process of converting a desired physical motion, obtained from, e.g., a vehicle model, into motion simulator input commands. This conversion is done by a motion cueing algorithm (MCA). In past decades, many different types of MCAs have been introduced [Gar10]. The vast majority of them are variations of the filter-based approach, which relies mainly on scaling down and filtering the physical motions such that the commanded motion lies within the limited motion envelope of a simulator.

Recently, several optimization-based MCAs have been developed, e.g. [May07, Beg12, Gar13, Ven15]. The most important difference with filter-based approaches is that an optimization-based MCA produces an optimized output, in which simulator constraints are explicitly accounted for, instead of a filtered output for which it is not guaranteed it lies within the simulator's operational capabilities. In some optimization-based MCAs the motions of the simulator platform are optimized which are then converted to simulator control commands by low-level controllers, e.g., [May07, Beg12]. In others, the simulator control commands are optimized directly, e.g., [Gar13, Ven15].

It is clear that filter-based and optimization-based MCAs are fundamentally *different* algorithms, but it is not readily apparent which provides *better* motion cueing, if at all, and under which conditions. One can compare the algorithms' output, i.e., the commanded simulator motions, but that does not provide a direct answer to the question how the motion cueing quality of the algorithms is actually *perceived* by simulator occupants. The study presented in this paper aimed at providing some answers to that question by performing an experimental comparison.

The filter-based and optimization-based motion cueing approaches were compared in a driving simulation experiment, executed on the Daimler Driving Simulator (DDS) of Daimler AG in Sindelfingen, Germany. In the experiment, an optimization-based algorithm, developed by the Max Planck Institute for Biological Cybernetics in Tübingen (MPI), was compared against Daimler's filter-based MCA, using a newly developed motion cueing quality rating method [Cle15]. The two algorithms will be referred to, in this paper, as the MPI-MCA and Daimler-MCA respectively. The goal of the comparison is to investigate whether optimization-based MCAs have, compared to filter-based approaches, the potential to further improve the quality of motion simulations.

Motion cueing algorithms

Filter-based motion cueing

Filter-based motion cueing algorithms consist of a combination of gains and filters, which transform (desired) vehicle motion into simulator set-points in real-time. Typically, the filters have a high-pass characteristic to prevent low-frequency accelerations from consuming a considerable part of the simulator's motion space. The gain functions can be linear or nonlinear and are adjustable, with the aim of reaching a good motion representation in a wide range of manoeuvers, preferably with a constant set of parameters. Special requirements of the manoeuvre that is to be simulated, like tight curves or turns, might be considered separately by the algorithm in order to provide good motion cueing quality while keeping the simulator within its operational limits. Well-known characteristics of the filter-based approach are tilt-coordination (where low-frequency components of the linear acceleration are reproduced by tilting the simulator platform) and motion washout (the ever-present push to return to the initial position). Such MCAs are commonly referred to as washout filters.

The Daimler-MCA for non-professional driver applications is based on a classical washout algorithm. Scaling factors and filters are used in all six degrees of freedom (DOF) to calculate the motion cues. Lateral, vertical and yaw excitations are dynamically limited by high-pass filters. A modified tilt-coordination algorithm provides an impression of steady-state acceleration in longitudinal direction and maximizes the use of the linear rail.

The main goal of the Daimler-MCA is to provide linear motion cues within the envelope of the motion system even during worst-case manoeuvres. The algorithm takes the outputs of the vehicle simulation (accelerations and rotation angles and rates in 6 DOF) and calculates the commands for the motion system at 500 Hz. When used in driver-in-the-loop studies, the algorithm operates in real-time such that the driver is free to choose velocity, acceleration, deceleration and manoeuvers like lane change or overtaking other cars.

Optimization-based motion cueing

Optimization-based motion cueing optimizes simulator motions or control commands through an optimization algorithm. An often-used approach is Model Predictive Control (MPC). MPC is a control methodology that optimizes the current control signal based on a process model and a future reference trajectory of finite length, while taking constraints into account [Raw15]. The optimization is governed by an objective function which quantifies the difference between (desired) vehicle motion and simulator motion. The optimization is constrained by the simulator's actuator limits. As a result, the optimized simulator control inputs and states always lie within the simulator's operational capabilities. MPC-based algorithms utilize predictions of future reference signals, using a 'prediction horizon' of a certain length, to compute the current control action. The advantage of this is that the current control action is optimized while taking future simulator states and control actions into account [Mar15].

In general, an MPC-based MCA finds a sequence of controls $\boldsymbol{u}$ and states $\boldsymbol{x}$ which minimizes the following objective function:

$$\begin{aligned} J(\boldsymbol{x},\boldsymbol{u}) = \sum_{k=1}^{N} \Bigg(\|\boldsymbol{u}_k\|_P^2 & \\ &+ \frac{1}{2}\left(\|\boldsymbol{y}(\boldsymbol{x}_k,\boldsymbol{u}_k) - \hat{\boldsymbol{y}}_k\|_R^2\right. \\ &\left. + \|\boldsymbol{y}(\boldsymbol{x}_{k+1},\boldsymbol{u}_k) - \hat{\boldsymbol{y}}_{k+1}\|_R^2\right)\Bigg) \\ &+ \sum_{k=1}^{N+1} \|\boldsymbol{x}_k - \boldsymbol{x}_0\|_Q^2 \end{aligned} \tag{1}$$

subject to the constraints:

$$\begin{gathered} \boldsymbol{x}_{k+1} - F(\boldsymbol{x}_k,\boldsymbol{u}_k) = \mathbf{0} \\ \boldsymbol{u}_{min} \le \boldsymbol{u}_k \le \boldsymbol{u}_{max} \\ \boldsymbol{x}_{min} \le \boldsymbol{x}_k \le \boldsymbol{x}_{max} \end{gathered} \tag{2}$$

where N – number of time steps, $\boldsymbol{x}_0$ – the "neutral" state of the simulator, $\boldsymbol{y}(\boldsymbol{x}_k,\boldsymbol{u}_k)$ – inertial signal at the head point in the simulator as a function of its state and input, $\hat{\boldsymbol{y}}_k$ – inertial signal at the head point in the vehicle (reference value), P,Q,R – symmetric positive-definite weighting matrices for penalizing control input, deviation from the neutral state and error in the inertial signal, respectively. F is the function that describes discrete-time dynamics of the system, $\boldsymbol{u}_{min}$, $\boldsymbol{u}_{max}$, $\boldsymbol{x}_{min}$, $\boldsymbol{x}_{max}$ are the lower and upper bounds of the inputs and states. The inertial signal is defined as

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{\omega} \end{bmatrix} \tag{3}$$

where $\boldsymbol{f}$ – specific force, $\boldsymbol{\omega}$ – rotational velocity at the head point.

The MPI-MCA is described in more detail in [Ven15, Pre15]. In the current paper, a *trajectory-based* optimization was performed, which means that the information of the entire trajectory was provided to the algorithm at the start of the optimization: i.e. in Eq. 1, N is the total number of trajectory samples and $\hat{\boldsymbol{y}}$ is obtained from a recording of the manoeuvre that was to be simulated (instead of a prediction of the future reference). In theory, this should lead to the best cueing quality, as the maximum amount of available information (i.e., a 'perfect prediction') is provided to the optimization algorithm. A clear disadvantage of this approach is that it makes the algorithm only suitable for simulation of pre-recorded manoeuvres. It is possible to use prediction methods to obtain real-time predictions of the future reference signal, which makes the algorithm suitable for driver-in-the-loop simulations, e.g., [Beg12]. It is to be expected that this would result in a lower simulation quality compared to the trajectory-based optimization approach used in the current study [Kat15].

The weighting matrices used in the objective function are: $P = 0.1$, $Q = 0$, $R = \mathrm{diag}(1,1,1,10,10,10)^2$. As the value for Q was zero, there was no penalty for deviation from the neutral state, which implies that the algorithm did not exhibit washout behaviour. The weighting factor 10 for the rotational velocities is chosen as an approximate ratio of standard deviation of specific force components in m/s^2 and standard deviation of rotational velocity components in rad/s for typical car manoeuvres.

The optimization was constrained by the actuator limits. For safety reasons, the bounds were set at 95% of the actual position, velocity and acceleration limits of each actuator. In addition, the simulator state was constrained by the condition that the initial and final position of the simulator should be upright (zero degrees of roll, pitch and yaw) and the initial and final velocity of simulator should be zero. The optimization was performed using CasADi toolbox [And13] and Ipopt solver [Wäc06].

Algorithm comparison

Table 1: comparison of algorithm characteristics

	Daimler-MCA	**MPI-MCA**
Type	Filter-based	Optimization-based
Real-time capable	Yes	No
Driver-in-the-loop applications	Suitable	Not suitable
Sampling rate	500 Hz	50 Hz
Future Reference	Not applicable	Entire trajectory
Accounting for simulator limits	Through manual tuning	Through constrained optimization
Tuned	Yes	No

It is important to note that the two MCAs described above are very different algorithms, each with their own characteristics, see Table 1.

The Daimler-MCA is a robust, real-time algorithm, suitable for driver-in-the-loop simulations. The MPI-MCA performs its optimization based on perfect knowledge of the entire driving manoeuvre, making it unsuitable for real-time driver-in-the-loop applications, but suitable for passive simulations.

The MPI-MCA does not run in real-time due to the high computational load associated with the optimization. The optimization of the trajectory used in this study – with a duration of approximately 5 minutes – took a few hours on a regular PC. After the optimization, the output of the MPI-MCA, which provided data at a sampling rate of 50 Hz, was resampled (interpolated) to 500 Hz, in order to run synchronously with the output of the Daimler-MCA.

During the optimization, the MPI-MCA utilized exact knowledge on the desired motion for all future time steps (trajectory-based optimization). Such an optimization would not be possible if the knowledge about the future is limited, as is the case in real-time driving scenarios with a driver in the loop. In that case, prediction algorithms would be required to obtain an estimate of the future reference trajectory. The effect of using (different approaches to) real-time prediction on simulation quality remains a topic to be addressed in future studies.

Furthermore, the optimization of the MPI-MCA is constrained by the simulator's actuator limits. As a result, the optimized simulator control inputs and states always lie within the simulator's operational capabilities. This is not guaranteed for the output of the Daimler-MCA, where simulator limits are typically accounted for by tuning the algorithm's parameters. The MPI-MCA did not need tuning for the experiment described in this paper. The implementation of the MPI-MCA used in the current study did not account for perceptual factors like drivers' motion sensitivity and thresholds [Pre15]. It is to be assumed that the implementation of such features will further improve the cueing quality of the MPI-MCA.

Due to the above differences, this study is not to be considered as a competitive comparison between MCA alternatives, but rather as an attempt to gain insight in the potential that an optimization-based approach has to offer with respect to well-established filter-based approaches.

Methods

Research questions

The primary research question of this study is whether optimization-based MCAs have the potential to further improve the quality of motion simulations compared to filter-based approaches. At the start of the study it was unknown whether there would be any measurable differences between the two algorithms, and if so, what can be learnt from these differences to further improve motion cueing.

In order to measure the quality of the motion cueing, a quality rating method developed at the MPI was utilized. The method was described and evaluated in [Cle15]. As the rating method was only recently developed, a secondary research question was whether the method provides reliable and repeatable results within and between participants.

Apparatus

The experiment was conducted in the Daimler Driving Simulator (DDS), an electrical hexapod platform mounted on a 12 m long linear axis [Zee10], see Fig. 1. For this experiment, the car's longitudinal

Figure 1. Exterior of the Daimler Driving Simulator (DDS).

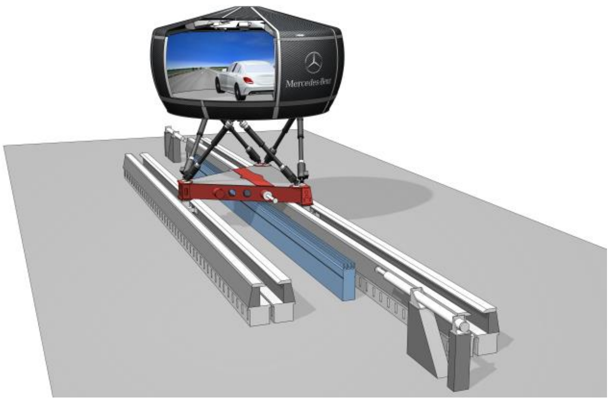

Figure 2. Car orientation during the experiment.

axis was aligned with the simulator's linear axis by rotating the cabin in the dome, see Fig. 2. This adjustment provides a relatively large motion space for the reproduction of longitudinal accelerations and deceleration with the disadvantage that the space for lateral motion is limited.

The driver's cabin was a standard Mercedes-Benz C-Class model (W204) equipped with an additional display showing the rating bar (described below).

Participants

In total 18 participants, 9 females, aged between 21 and 40 (mean = 29.3; std = 5.7) took part in the experiment. All had previous experience in driving simulators, but no or limited knowledge of motion cueing. They were expert drivers with a minimum mileage of 10,000 km per year (mean = 17,222; std = 7,496). Two participants did not complete the experiment due to motion sickness symptoms and their data were excluded from the analysis.

Experimental procedure

In the experiment, participants were presented with four pairs of evaluation trials, of which the first pair was used for training purposes. Each trial consisted of the playback of an identical pre-recorded simulated drive. While the visuals remained unaltered, the vehicle motions of the simulated drive were processed by either the Daimler-MCA or the MPI-MCA, generating two different simulator trajectories. These trajectories were repeatedly presented in random order at each trial pair. In total, participants rated each trajectory four times, of which the last three were included in the data analysis. During the playback, the participants did not need to take any actions on the steering wheel or pedals. Instead, they were asked to concentrate on the movements of the simulator and rate the perceived motion mismatch, i.e., the perceived mismatch between the motion felt in the simulator and the motion one would expect from a drive in a real car.

The experiment lasted approximately 1 hour, of which 45 minutes in the DDS (Table 2).

Table 2. Overview of experiment procedure.

TRAINING (1 trial pair, 10 min)	Familiarization with rating device and procedure.
EXPERIMENT PART 1 (1 trial pair, 10 min)	Motion mismatch rating: CR followed by OR for each trial
BREAK (5 min)	
EXPERIMENT PART 2 (2 trial pairs, 20 min)	Motion mismatch rating: CR followed by OR for each trial

Rating procedure

The ratings were provided using the built-in rotary COMAND-knob of a Mercedes C-Class. By rotating it, participants controlled a rating bar (Fig. 3) with 15 coloured markers, visible on a small screen located to the left of the steering wheel (Fig. 4). A rotation to the left reduced the number of visible marks (lower motion mismatch); a rotation to the right increased the number of visible marks (higher motion mismatch). There was always at least one green mark visible. The quality of the motion cueing was rated in two ways:

- A continuous rating (CR) method was used to measure time-varying aspects of the perceived mismatch between real and simulated drive. For the CR, participants were asked to continuously assign a value (magnitude) to the instantaneous perceived motion mismatch via the rotary knob during the playback. If no mismatch was perceived they were asked to provide a rating of zero.
- After each trial the participants were asked to provide an overall rating (OR), by indicating the perceived motion mismatch of the entire playback. The OR resulted in a single rating for each trial.

The rating method is described in more detail in [Cle15].

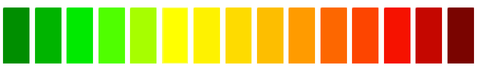

Figure 3. Rating bar with 15 coloured marks. Green/red marks indicated low/high perceived mismatch.

Figure 4. Location of rating bar and rating knob in car cabin.

Stimuli

The recorded simulated drive that was used in this experiment was performed by a human driver. The drive consisted of an initial mild acceleration along a rural road up to the speed of 100 km/h. The rural road consisted of a large-radius left, right and left curve, during which a constant speed was maintained. Shortly after exiting the last curve, the car performed a double lane change manoeuvre at constant speed to avoid a car parked on the right-hand side of the road. The speed was then decreased to 70 km/h and then to 50 km/h, upon entering an urban area. After a few gentle curves the car came to a full stop at a red traffic light. After the light turned green, the car accelerated to a cruising speed of 50 km/h. After a few gentle curves the car entered a four-exit roundabout and exited at the second exit. After a few gentle curves the car took a 90-degrees left turn and reached finally a full stop at another traffic light. Overall the simulated drive lasted about 5 minutes.

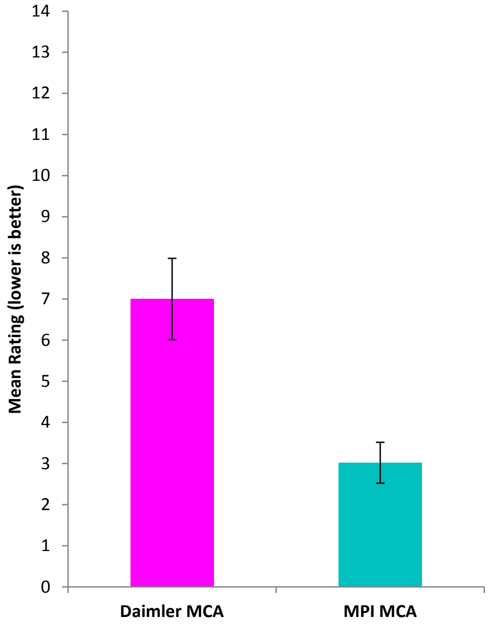

Figure 5. Mean overall rating across three evaluation trials. Error bars indicate 95% confidence interval.

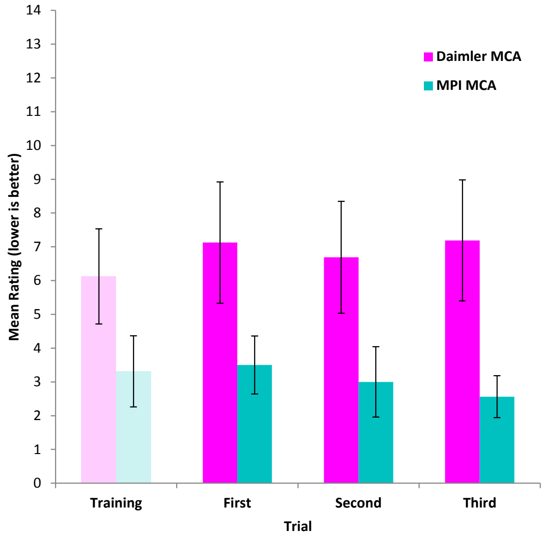

Figure 6. Mean overall rating per trial. Error bars indicate 95% confidence interval.

Results

Using the method described above, participants rated the perceived motion mismatch between 0 (no motion mismatch) and 14 (strong motion mismatch). Note that a higher rating value implies a lower cueing quality.

The average results obtained for the overall rating (OR) are shown in Fig. 5, showing the mean overall rating (Daimler = 7.00, MPI = 3.02) across all participants. The difference between MCAs is significant: $t(47) = 8.296$, $p < 0.05$. This indicates that participants felt less motion mismatch with the MPI-MCA than with the Daimler-MCA.

Fig. 6 shows that the overall rating does not change significantly over the three evaluation trials. This result indicates that learning, habituation or fatigue effects did not impact the overall rating, and participants were able to provide consistent estimates during the whole experiment.

The average results obtained for the continuous rating (CR) are shown in Fig. 7, showing the mean continuous rating across all participants. Before computing the means, the CR raw data were standardized per trial pair by subtracting the minimum rating and dividing by the rating range.

The results showed also that participants were consistent when rating perceived motion mismatch continuously. From the 16 participants only two did not pass the statistical test for consistency (Cronbach's Alpha < 0.7) [Hai09], as shown in Table 3. This result is in line with previous findings in which the same method was used to determine MCA perceived quality [Cle15].

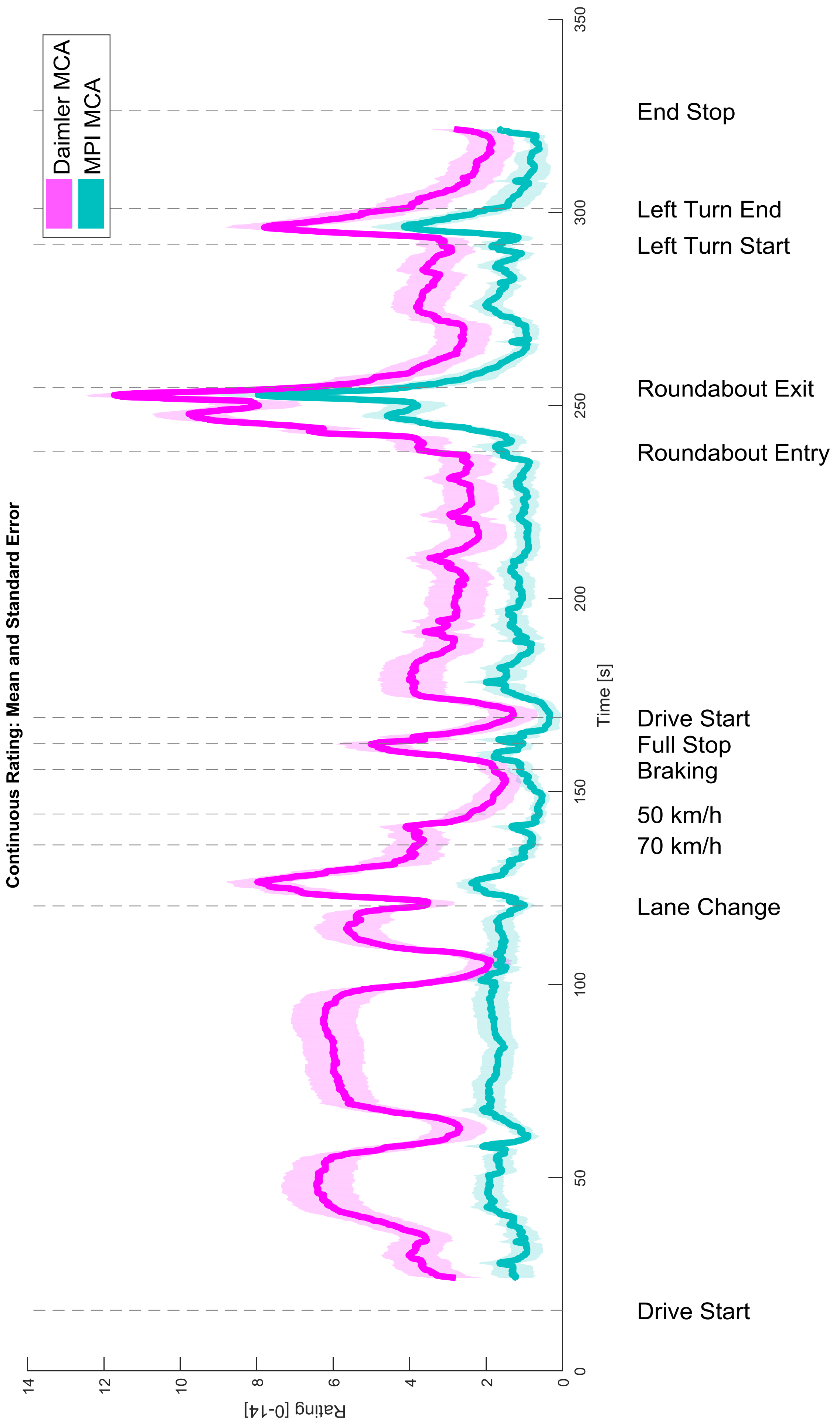

Figure 7. Mean continuous rating. Shaded areas indicate standard error.

Table 3 Consistency test (Cronbach's Alpha) for participants' continuous rating. [* Two participants did not reach 0.7].

Participant	Cronbach's Alpha
0	0.894
1	0.828
2	0.883
3	0.868
8	0.822
9	0.897
10	0.884
11	0.843
12	0.888
13	0.84
14	0.683 *
15	0.875
16	0.671 *
17	0.886
18	0.727
19	0.854

The presented results show that the motion mismatch, which can be assumed to be a metric for motion cueing quality, was rated very differently for the two MCAs. In the following, some insight is provided into several of the dominant mechanism that contribute to the measured differences.

First of all, the MCAs utilize very different approaches to deal with the limited motion space of the simulator. The Daimler-MCA relies to a large extent on globally scaling down all motions before applying filtering. The scaling keeps the visual and vestibular motions coherent, but it also significantly reduces the overall strength of the vestibular motions that is felt. Because the MPI-MCA uses an optimization at each time step, motion scaling is only applied at those time steps where this is required (i.e., local scaling), resulting in virtually no global scaling.

Second, the MPI-MCA is capable of making more efficient use of the simulator's actuators, such as the linear axis. Where the Daimler-MCA only uses the linear axis to reproduce longitudinal motions, the MPI-MCA uses this axis for both longitudinal and lateral motions by utilizing the current yaw angle of the hexapod. For example, the MPI-MCA reproduced about 30% of the lateral acceleration during one of the curves through the linear axis by exploiting the fact that the hexapod was yawed by 28 degrees. The remaining lateral acceleration was reproduced by tilt-coordination, i.e. by rolling the cabin.

The above example illustrates that the MPI-MCA utilizes tilt-coordination, just as the Daimler-MCA. It should be noted, however, that the angular rates produced by the MPI-MCA are much larger than those produced by the Daimler-MCA (mean of 0.47 deg/s versus 0.14 deg/s). This difference can mainly be attributed to the different trade-offs made for each MCA between reproducing sustained accelerations through tilt-coordination and the tilt rate errors that tilt-coordination produces. The superior ratings obtained for the MPI-MCA in this experiment prompt the question whether the improved cueing of sustained acceleration can outweigh the cost of introducing larger tilt rate errors.

Another noteworthy difference is that the Daimler-MCA uses motion washout to conserve motion space, while the MPI-MCA had no washout mechanism implemented (as $Q = 0$ in Eq. 1). Instead, the MPI-MCA makes use of prepositioning based on a reference signal containing all future vehicle motions. For example, several seconds before a right-to-left lane change maneuver, the hexapod is slowly moved to a large lateral offset to the right (~1.1m), which maximizes the available leftward excursion. A similar prepositioning strategy can be identified for the roundabout maneuver. Here, the available clockwise yaw excursion of the hexapod is maximized by slowly prepositioning the hexapod to a counterclockwise yaw angle of 21 degrees before entering the roundabout.

A final mechanism that will be discussed here is 'velocity buffering', which is utilized by the MPI-MCA but not by the Daimler-MCA. In velocity buffering the simulator's velocity is utilized to maximize the simulator's future acceleration capabilities. It can be interpreted as the velocity equivalent of the prepositioning mechanism. By giving the simulator a constant velocity in one direction, the duration for which the simulator can then accelerate in the opposite direction is increased. In the experiment, velocity buffering was most clearly observed when accelerating from standstill after the full stop at the traffic light. For the Daimler-MCA, the simulator was not moving during the vehicle's standstill, but the MPI-MCA generated a backwards simulator motion at constant velocity during the vehicle's standstill. As this velocity was constant it was not perceived by the participant. Upon accelerating, the initial forward vehicle acceleration was simulated by first *slowing down* the backwards motion before the simulator obtained forward velocity, effectively extending the duration at which the forward acceleration could be sustained. Effective velocity buffering requires accurate knowledge on future accelerations in order not to exceed actuator position limits.

There are several other mechanism that account for the differences observed in the simulator trajectories obtained from the two MCAs. A detailed discussion, supported by a quantitative analysis, would take up more space than is available here. Such a discussion will be provided in future publications.

Conclusions and discussion

The results of the experiment lead to the following conclusions:

- The results show that participants were able to rate the perceived mismatch consistently over the various repetitions. This holds for both the overall rating (Fig. 6) as for the continuous rating (Table 3).
- The overall rating (Fig. 5) shows that participants felt less motion mismatch with the MPI-MCA than with the Daimler-MCA.
- The continuous rating (Fig. 7) shows that, on average, the perceived mismatch for the MPI-MCA was rated lower than the perceived mismatch for the Daimler-MCA for every point in time.
- Several mechanisms were identified that contributed to the differences observed between the two algorithms, amongst which scaling, actuator utilization, tilt coordination, prepositioning and velocity buffering.

These conclusions indicate that the quality of the motion cueing was rated better for the MPI-MCA than for the Daimler-MCA. It should be noted however, that the two MCAs are very different algorithms, each with their own characteristics. This study is therefore not to be considered as a competitive comparison between MCA alternatives, but rather as an attempt to gain insight in the potential that an optimization-based approach has to offer. The results show that there exists a potential to further improve the quality of the motion simulation with optimization-based methods, deserving of further research.

Regarding the rating method, the results show that the rating method provides reliable and repeatable results within and between participants, which further confirms the reliability and utility of the method. Ongoing research investigates the dynamics and limitations of the rating behavior.

In future experiments it could be investigated how the quality of the MPI-MCA degrades if the prediction horizon is decreased (i.e., no longer using trajectory-based optimization) or if the prediction is imperfect (i.e., no longer using the pre-recorded trajectory but a predicted trajectory). Also, it could be studied how the objective function can be adapted to further improve the quality of the MPI-MCA. Finally, it would be interesting to investigate whether the tuning of the Daimler-MCA can be further improved based on the more detailed analysis of the results.

References

Anderson, J. **A General-Purpose Software Framework for Dynamic Optimization**. *PhD Thesis, Arenberg Doctoral School, KU Leuven,* 2013.

Beghi, A., Bruschetta, M., & Maran, F. **A real time implementation of MPC based Motion Cueing strategy for driving simulators**. *51st IEEE Conference on Decision and Control*, *Maui, HI, USA,* 2012, pp. 6340-6345.

Cleij, D., Venrooij, J., Pretto, P., Pool, D. M., Mulder, M. & Bülthoff, H. H. **Continuous rating of perceived visual-inertial motion incoherence during driving simulation.** *DSC 2015 Europe: Driving Simulation Conference & Exhibition, Tübingen, Germany,* 2015, pp. 191-198.

Garrett, N. J. I., & Best, M. C. **Driving simulator motion cueing algorithms – a survey of the state of the art**. *Proceedings of the 10th International Symposium on Advanced Vehicle Control (AVEC), Loughborough, UK,* 2010, pp. 183-188.

Garrett, N. J. I. & Best, M. C., **Model predictive driving simulator motion cueing algorithm with actuator-based constraints**. *Vehicle System Dynamics*, 2013, 51(8), pp. 1151-1172.

Hair, J. F., Black, W. C., Babin, B. J. & Anderson, R. E., **Multivariate Data Analysis**, 7th ed. Prentice Hall, 2009.

Katliar, M., De Winkel, K. N., Venrooij, J., Pretto, P. & Bülthoff, H. H. **Impact of MPC prediction horizon on motion cueing fidelity** *DSC 2015 Europe: Driving Simulation Conference & Exhibition, Tübingen, Germany,* 2015, pp. 219-222.

Maran, F., Bruschetta, M., Beghi, A. & Minen, D., **Improvement of an MPC-based motion cueing algorithm with time-varying prediction and driver behaviour estimation**, *DSC 2015 Europe: Driving Simulation Conference & Exhibition, Tübingen, Germany*, 2015, pp. 183-190.

Mayrhofer, M., Langwallner, B., Tischer, W., Schlüsselberger, R., Bles, W. & Wentink, M., **An innovative optimal control approach for the next generation motion platform DESDEMONA**. *AIAA Modeling and Simulation Technologies Conference, Hilton Head, SC, USA,* 2007, pp. 1-14

Pretto P., Venrooij J., Nesti A. & Bülthoff H. H., **Perception-Based Motion Cueing: A Cybernetics Approach to Motion Simulation**, *Recent Progress in Brain and Cognitive Engineering*, (Ed) S.-W. Lee, Springer, Dordrecht, The Netherlands, 2015, pp. 131-152.

Rawlings J. B., Mayne, D. Q., **Model Predictive Control: Theory and Design**, 5th ed. Nob Hill Pub., 2015.

Venrooij, J., Pretto, P., Katliar, M., Nooij, S. A. E., Nesti, A., Lächele, M., de Winkel, K.N., Cleij, D. & Bülthoff, H. H. **Perception-based motion cueing: validation in driving simulation.** *DSC 2015 Europe: Driving Simulation Conference & Exhibition, Tübingen, Germany,* 2015, pp. 153-161.

Wächter, A. & Biegler, L. T., **On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming**, Mathematical Programming 106(1), pp. 25-57, 2006

Zeeb E., **Daimler's New Full-Scale, High-dynamic Driving Simulator - A Technical Overview**. *DSC 2010: Driving Simulation Conference 2010*, Paris, France, 2012, pp. 157–165.

