Comparison of Quality Metrics between Motion Cueing Algorithms in a Virtual Test Environment

Kolff, M.J.C.; Venrooij, J.; Pool, D.M.; Mulder, Max

Publication date
2020

Document Version
Final published version

Citation (APA)
Kolff, M. J. C., Venrooij, J., Pool, D. M., & Mulder, M. (2020). *Comparison of Quality Metrics between Motion Cueing Algorithms in a Virtual Test Environment*. 93-99. Paper presented at Driving Simulation Conference Europe 2020 VR, Antibes, France.

Comparison of Quality Metrics between Motion Cueing Algorithms in a Virtual Test Environment

Maurice Kolff[1,2], Joost Venrooij[1], Daan M. Pool[2] and Max Mulder[2]

(1) BMW Group, Research, New Technologies, Innovations, 80788 Munich, e-mail : {Maurice.Kolff, Joost.Venrooij}@bmw.de
(2) Delft University of Technology, Faculty of Aerospace Engineering, section of Control & Simulation, 2629 HS Delft, e-mail : {m.j.c.kolff, d.m.pool, m.mulder}@tudelft.nl

Abstract - *Motion cueing algorithm design often involves a trade-off between priorities due to the limited workspace of the simulator. Such a trade-off requires a detailed understanding of human perception, which we do not yet have. For that reason, objective motion cueing quality metrics, based on the difference between vehicle and simulator signals, offer a fast and simple alternative. Next to motion cueing quality, we argue that the total motion cueing algorithm (MCA) quality is about more than only the quality of the motion, and can also entail implementation and operational aspects of an MCA for a specific use-case and simulator combination, i.e., it is a task-dependent issue. In this paper this idea is discussed by comparing three objective motion cueing quality metrics (absolute difference, delay and cross-correlation) from literature and two metrics regarding simulator operations (workspace management and energy consumption). Comparing such metrics is difficult, but is nevertheless useful to improve the process of simulator operations if various MCAs and/or simulators are available, to aid their selection process. As a first step towards such a method, a Virtual Test Environment (VTE) was developed as a versatile software environment to compare these metrics, as well as to visualize simulator motion and its characteristics in a 3D-animation. This aims at helping MCA designers in making choices between different MCA types, their configurations, simulators and use-cases, guiding them to select the best-suited motion cueing solution.*

Keywords: *Motion cueing; quality comparison; objective criteria; test environment.*

1. Introduction

The key function of a motion simulator is to provide drivers with similar motion as they would experience in a real vehicle. As the workspace of a simulator is by definition limited, a Motion Cueing Algorithm (MCA) is required, which typically limits the vehicle specific forces and rotational rates to fit the resulting motion inside the simulator workspace, while keeping differences between vehicle and simulator motion as small as possible. As perfect motion cueing is often not possible, the critical questions are which differences are acceptable, which are important to avoid and at what cost, such as investigated by [Cle20].

In recent years, these questions have become even more important due to two developments. Firstly, high-performance motion driving simulators, such as currently under construction at BMW, have the potential to improve the overall motion cueing quality compared to the classical hexapod structures. Larger workspaces allow for higher scaling factors, which not only lead to better tracking of the vehicle reference signal, but can also result in an amplification of motion cueing errors [Rom19]. In other words, the larger the simulator workspace is, the more important it becomes to focus on what the simulator is exactly doing wrong than what it is roughly doing right.
The second development comes from novel MCA types, such as model-predictive control (MPC) algorithms that currently find their way into motion simulators [Gar10]. First published by [Dag04, Dag09], an MPC algorithm typically provides a higher motion cueing quality compared to traditional filter-based classical washout algorithms (CWAs) [D. 17], as it optimizes the simulator movement based on the available workspace at each moment in time, compared to the overall, worst-case scenario tuning of CWAs. Nevertheless, MPC algorithms may not always offer the best practical solution, as they often put heavier constraints on other factors, such as inducing a higher computational load and being more complex to implement. For that reason, MCA designers require a comparison method looking at more than just motion cueing quality, which might depend on the available simulator(s) and/or use-case, as well as the wishes of designers, operators and users.

To the best of our knowledge, there is no such task-dependent approach to be found in literature. A task-oriented approach to compare driving simulators was made by [Fis15]. Although this work focused on specific simulator qualities based on the requirements of a use-case, a similar approach could eventually be of use for the trade-off between MCAs and simulators.

Thus, what is still missing in this context is a task-oriented approach for MCA tuning, testing and comparison that is able to help the trade-off between various MCAs. The goal of this paper is to describe a Virtual Test Environment (VTE) that can form the basis for such an approach within a single offline software environment. An offline analysis can also help the further design of MCAs [Qai12], as motion cueing quality can be used as a tool to quickly analyse a large set of MCAs. Although our future goal is to perform a deeper analysis of MCA quality metrics, in this paper, only three objective motion cueing quality metrics as defined by [Gro19] are implemented in the VTE for a single simulator as a demonstration. Many other objective metrics exist as well (such as those proposed by [Pou98, Cas15, Qai12]). Furthermore, two metrics regarding the operational aspects of motion cueing are implemented, being workspace management and energy consumption of the the simulator. The VTE can not only help trade-offs between MCAs, but also provides an insight in simulator behaviour due to differences in MCAs by showing simulator movement in 3D-animation. The VTE serves as a methodological preparation of the new simulators currently under construction at BMW in Munich.

The paper is structured as follows. First, an overview is given of objective metrics for MCA quality. Then the working principle of the VTE is explained, after which the simulator and use-case used in the analysis are discussed in Section 4. Results and discussion are given in Section 5, followed with the conclusions.

2. MCA Quality Metrics

In this section an overview is given of various example MCA quality metrics that can be used for trade-offs between MCAs. The goal of this analysis, and one of the main reasons the VTE was developed, is to be able to assess and compare specific characteristics of different types of MCAs, as well as different configurations of the same MCA, and assess their viability for a certain use-case.

An important component of such an analysis is to predict how satisfactory the motion cueing quality as perceived by the human driver will be. However, we suggest that other metrics regarding the implementation and operation of MCAs could also have a large effect on the MCA choice and that 'MCA quality' is therefore a broader term than only motion cueing quality. A trade-off between MCA quality properties therefore becomes a task-dependent approach, as motivations for a certain weighting between such quality metrics may depend on the priority determined by the use-case. For example, some testing scenarios in a simulator have a low focus on accurate motion, such that a simpler MCA with smaller platform excitations, and therefore with a lower energy consumption, is a viable option. Other use-cases might require the best motion cueing quality possible, regardless the cost.

Another reason for the importance of MCA quality analysis and comparison is the BMW Simulation Center currently under construction in Munich, Germany, which will operate multiple motion simulators. These simulators have different characteristics as they are being constructed for different purposes, including a simulator for highly dynamic maneuvers and a simulator for urban driving scenarios. The MCA quality metrics could give instructions on which use-case is best performed on which simulator and with which MCA. The development of the Virtual Test Environment therefore also serves the methodological preparation for enabling the best motion cueing solution across a simulator fleet.

2.1. Motion cueing quality

Even when regarding a variety of 'cueing quality metrics' that also look at the operations perspective of each MCA, the difference between expected and actual motion as experienced by humans drivers in the simulator is often the most critical part of MCA quality. It must be recognized that currently the models and tools to fully understand the human element are not available [Cas20], and therefore subjective ratings are still often used for motion cueing quality assessment, such as done by [Cle18]. Their main drawback is that subjective analyses are often too time-consuming to systematically assess the motion cueing quality for a large number of possible MCAs and their parameterizations, as they require experimental data for each of these variations.

However, even without fully understanding the human element in simulator studies, objective metrics can be used for MCA comparison by evaluating factors that drivers generally find important for their perception of good cueing quality. Here, 'objective' refers to numerical differences in vehicle reference (input) and MCA (output) signals. For example, [Cas15] introduced various objective metrics that were compared to subjective ratings, to see which metric would best predict human evaluations. Although not in the context of a comparisons between MCAs, their results showed the strongest dependence on delay and cross-correlation, rather than absolute differences, between the vehicle reference and MCA signals.

Similar to [Cas15], [Gro19] computed the absolute difference, a delay indicator and the correlation coefficient between the vehicle reference signal and the computed MCA signal between two variations of the same MPC controller with different configurations, although these were not experimentally compared to subjective ratings. We use the quality metrics of [Gro19], as these are the simplest to implement, without any perceptual thresholds as a basis and first example for the viability of the VTE.

2.1.1. Absolute difference

The first metric is the absolute difference (AD), as defined by [Gro19], which is based on the ratio of the area of the difference between the two signals divided by the area of the reference signal:

$$AD_d = \frac{\int |(f_d^v - f_d^s)| dt}{\int |f_d^v| dt}, \tag{1}$$

where f_d^v is the reference specific force signal with $d \in \{x, y, z\}$ for the degree-of-freedom. In this case the vehicle acceleration is taken as the reference signal. The signal f_d^s is the resulting simulator specific force resulting as output from the MCA. Similarly, the same equation can be used for the three rotational rates by substituting ω_d^v and ω_d^s with $d \in \{\phi, \theta, \psi\}$. A value closer to zero indicates a better reproduction of the vehicle cues. Arguably, this is one of the simplest and most direct comparison metrics for motion cueing quality, as it is directly based on the signal differences.

2.1.2. Cross-correlation

The correlation coefficient (CC) in direction d is defined as:

$$CC_d = \frac{\max R(f_d^v, f_d^s)(\tau)}{\max R(f_d^v, f_d^v)(\tau)}, \qquad (2)$$

where $R(f_d^v, f_d^s)(\tau)$ is the cross-correlation of the acceleration signals f_d^v and f_d^s (and similarly for the three rotational rates) as a function of the time shift τ between the two signals. The denominator term of the correlation coefficient represents the normalization by dividing by the auto-correlation of f_d^v. A value closer to 1 indicates a better reproduction of the vehicle cues, whereas a value of 0 indicates no correlation.

2.1.3. Time delay

The delay indicator (DI) in degree-of-freedom d between two signals can be found by calculating the time shift τ that maximizes the cross-correlation:

$$DI_d = \arg\max_{\tau \in \mathbb{R}} R(f_d^v(t-\tau), f_d^s(t)) \qquad (3)$$

It is expected that a clear difference can be seen in this indicator, as the type of MCA strongly affects the delay. A washout-algorithm inherently has phase shift as it makes use of filters, whereas model-predictive control algorithms can compensate for delays if the prediction horizon is large enough.

2.2. Operational quality

Besides the motion cueing quality metrics, two operational quality metrics are included in the comparison.

2.2.1. Workspace management

Workspace management aims to answer how much of the available workspace is used, and thus if the MCA is able to exploit all the space it is offered. As a first step, a convex hull similar to that by [Gro19] was calculated, which is the volume that spans around the outer most points the simulator has reached during a chosen time period, in this case a single maneuver, for the three positions vectors (x, y, z) as well as the three rotation vectors (ϕ, θ, ψ). In the future a similar volume metric per unit of cueing quality could be useful as well, in which the lower volume that is used for getting the same cueing quality thus means that the MCA is superior to an MCA that results in a higher volume. In other words, it makes more effective use of the workspace it is given.

2.2.2. Energy consumption

Energy consumption can be an important metric for MCA trade-off, especially if motion cueing quality does not have the highest priority. More accurate models of energy consumption based on the specific simulator characteristics can be included, although in this example a simple mass-normalized total kinetic energy metric summed over time as introduced by [Ven15] was used:

$$E = \sum_{i=1}^{6} \int \tfrac{1}{2} v_d^s(t)^2 dt, \qquad (4)$$

where $v_d^s(t)$ is the velocity along the d^{th}-axis, summed over the six axes. Although [Ven15] used this expression to calculate the amount of motion the simulator produces, this metric can be applied as a simplified energy consumption estimate by assuming that it is only caused by movement of the hardware.

3. Comparison environment

The core functionality of the developed Virtual Test Environment is the ability to simulate the output of different MCAs within a single environment, for a given simulator and input file, such as measured vehicle data from test runs. This not only results in simulated output data that can be used for MCA comparison, but also gives the ability to render the outputs (simulator motion) in a 3D visualization with live plotting the simulator output and corresponding quality metrics at the same time, resulting in an intuitive method of seeing differences between different MCA outcomes.

3.1. Simulink structure

MCAs can be expressed in various programs or programming languages (such as Simulink, C++ or python), although a core functionality of the VTE is the ability to simulate different MCAs at the same time. For that reason, the VTE was developed in Matlab/Simulink. Simulink accepts, besides MCAs developed in Simulink itself, models defined in other programming languages as well. This allows for a large flexibility in the amount of sources that can be used. The outputs of MCAs typically have the same form, including (but not limited to) platform and perceived dynamics and actuator deflections.

3.2. Working principle

The VTE requires three different user inputs:

1. The (measured or simulated) **vehicle data** for the considered use case, which are to be cued in the simulator using its motion cueing algorithm.
2. The **MCAs** that are to be compared. These can be either completely different MCAs in terms of structure (such as classical washout, model-predictive control or other) or variations of the same MCA, of which the parameters can be altered in the VTE as well. This can for example be used to visualize the difference between various cut-off frequencies for the distribution of lateral accelerations, whereas all other model parameters remain the same. Furthermore, for MCAs that require large computational times to be computed within the VTE (such as MPC-based algorithms), the option also exists to add pre-calculated data of that MCA.
3. The **simulator geometry, type, and workspace** parameters to be used in the simulation. The output of an MCA is typically limited by the simulator workspace, i.e., by limiting the excursions, velocities and accelerations that the simulator is allowed to make based on its hardware limits. These limits are specified per simulator in a separate file.

Matching each simulator's DoFs and limits, 3D models were made in the Simulink 3D world editor. The various components of each simulator (such as its projector dome, yaw table, hexapod plate, legs and base or platform/rail) can all be individually included by a simple 3D-representation. An example of the hexapod simulator used in the viability analysis of this paper is shown in Figure 1. The Simulink 3D workbench has the benefit of being able to directly communicate with Simulink and thus the translations and rotations of each individual components of the simulator in the 3D model, corresponding to what one would see for the real simulator.

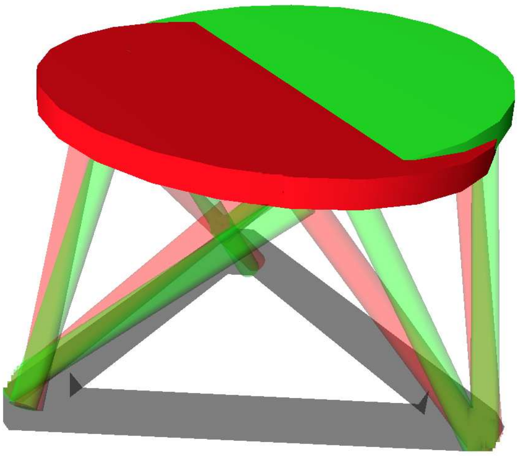

Figure 1: VTE screenshot of the CWA (green) and OPT (red) MCA plate and hexapod actuators for the PMS.

4. Application example

4.1. Simulator

In this example application of the VTE the Portable Motion Simulator (PMS) is used, which is a traditional hexapod configuration, shown in Figure 2. The position, velocity and acceleration limits are listed in Table 1 in all six degrees-of-freedom, whereas the maximum excursion of each of the actuators from the neutral state is $\delta_a = \pm 0.2$ m.

Figure 2: The Portable Motion Simulator as used in the analysis.

Table 1: Hexapod position, velocity and acceleration limits of the PMS.

	p	v	a
x	±0.36 m	0.6 m/s	11 m/s^2
y	±0.38 m	0.6 m/s	11 m/s^2
z	±0.26 m	0.5 m/s	12 m/s^2
ϕ	±23.00 deg	40 deg/s	500 deg/s^2
θ	±23.00 deg	40 deg/s	500 deg/s^2
ψ	±22.00 deg	40 deg/s	700 deg/s^2

4.2. Use-case

Vehicle data (specific forces and rotational rates) were collected on the PMS simulator for a rural road near Haimhausen, Bavaria, Germany. The corresponding vehicle data was computed based on the driver behaviour, which subsequently served as the inputs for the two MCAs under investigation. The road as driven by the driver is shown by its coordinates in Figure 3 and was divided into four maneuvers, which consisted of combined longitudinal and lateral specific forces, where for each maneuver the calculated metrics were determined separately:

M1: Acceleration up to 100 km/h with slight cornering.
M2: Slalom at at 100 km/h.
M3: Braking for a 50 km/h sign.
M4: Braking down to 20 km/h, roundabout, followed by an acceleration to 100 km/h.

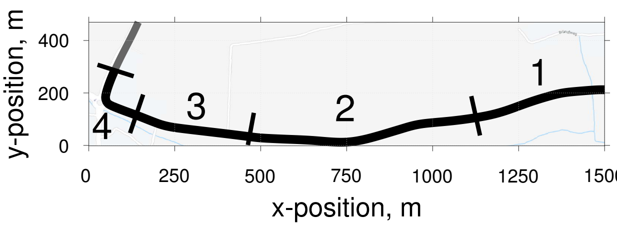

Figure 3: (x, y)-positions of the road signal used for analysis, with the separate maneuvers numbered.

4.3. Motion Cueing Algorithms

Two motion cueing algorithms were directly compared in the VTE. As both are common in the simulator industry, these are discussed briefly.

Classical washout algorithm: The first MCA was a typical classical washout algorithm. This filter-based approach, based on the work of [Rei85], uses a high-pass filter in the inertial frame for the translational as well as the rotational channels, to avoid the simulator reaching positions outside of its workspace and washout the simulator motion back to its neutral state. The low-frequency translational accelerations are reproduced by tilt-coordination. This makes use of the gravity vector to create a sustained acceleration in x and y due to rotations in θ and ϕ, respectively. Tilt-coordination in z-direction is not needed, as a rotation in ψ does not affect the gravity vector. The signal for tilt-coordination is low-pass filtered in the body frame complementary to the high-pass filtered simulator translational accelerations.

This MCA is denoted as '**CWA**'

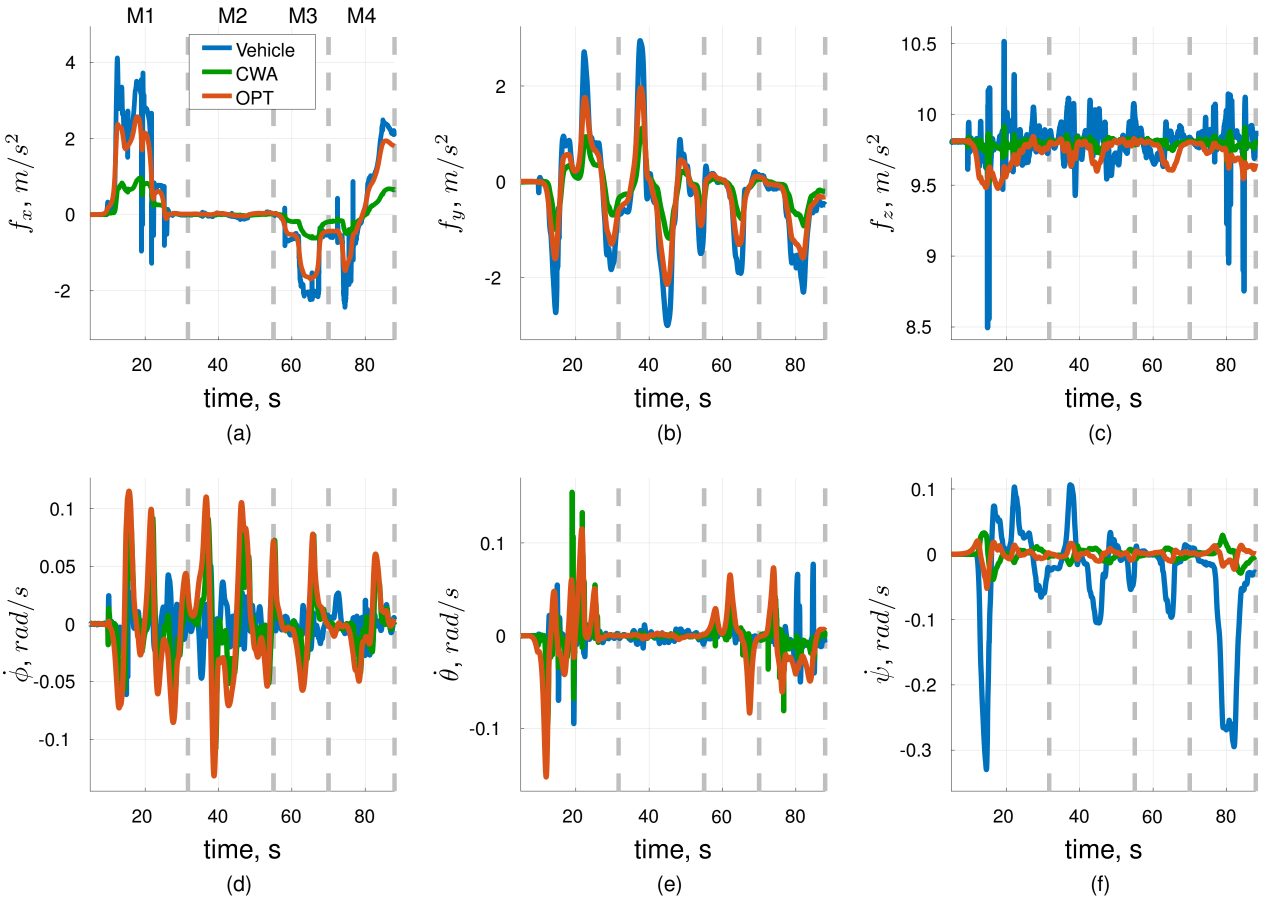

Figure 4: VTE output data of the perceived specific forces (a-c) and rotational rates (d-f) for the measured vehicle data and two MCA outputs.

Optimal model-predictive control: Instead of only responding to the simulator state using filters, an MPC algorithm uses predictions of future states to optimize the simulator motion to bring it as close as possible to a reference, such as the accelerations and rotations one is trying to reenact in the simulator. Here, knowledge of the kinematics of the system, in this case the motion simulator, is required. Although online MPC has become a viable option in recent years [Dro18, Beg12, Ell19a], the model-predictive control application used in this paper is a form of optimal control as it had perfect knowledge of future states, which is not possible in online applications. A non-perfect knowledge of the future, for example when using a finite prediction horizon, will result in a non-optimal solution [Kat15]. Details on the specific structure of this algorithm can be found in [Ell19b]. Similarly, the cueing error weights along the six degrees-of-freedom were set to $W = [1\ 1\ 1\ 10\ 10\ 10]$.

This MCA is denoted as '**OPT**'.

5. Results and Discussion

Figures 4a-f show the simulated outputs of the CWA (green) and OPT (red) algorithms, together with the vehicle data (blue) that they aim to reproduce. The calculated hexapod actuator deflections for both algorithms are shown in Figures 5a-f. As is clear from the pitch- and roll rates, the OPT algorithm makes strong use of tilt-coordination, resulting in a decent reproduction of the specific forces f_x and f_y. With this in mind, and by using the 3D-visualization option of the VTE, the parameters of the CWA were tuned to produce similar behaviour as the OPT. Nevertheless, the longitudinal- and lateral specific forces are clearly worse for the CWA. As this algorithm is not able to foresee upcoming maneuvers, it requires high rotational rates to provide effective tilt-coordination, which is typically sensed by human drivers [Rei85], meaning that less tilt-coordination was possible. The OPT algorithm can slowly build-up the tilt-coordination, resulting in higher specific forces. In the tuning process, the 3D-animation tool as part of the VTE drastically sped up the tuning process, as visualization of the motion platform, while at the same time looking at the actuators, showed where and why the limits of the simulator are reached. A video of this 3D-tool in action is provided at the end of this paper.

The five MCA quality metrics are shown in Table 2 and were calculated for each of the four maneuvers separately. [Gro19] only computed the absolute difference (AD) for the signals f_x, f_y and $\dot{\psi}$, as these vehicle signals have a relatively high power compared to $\dot{\phi}$, $\dot{\theta}$ and f_z. These latter three signals are also highly affected by the tilt-coordination and therefore also shown, for completion purposes, in Table 2.

As noted before, the OPT MCA uses the predic-

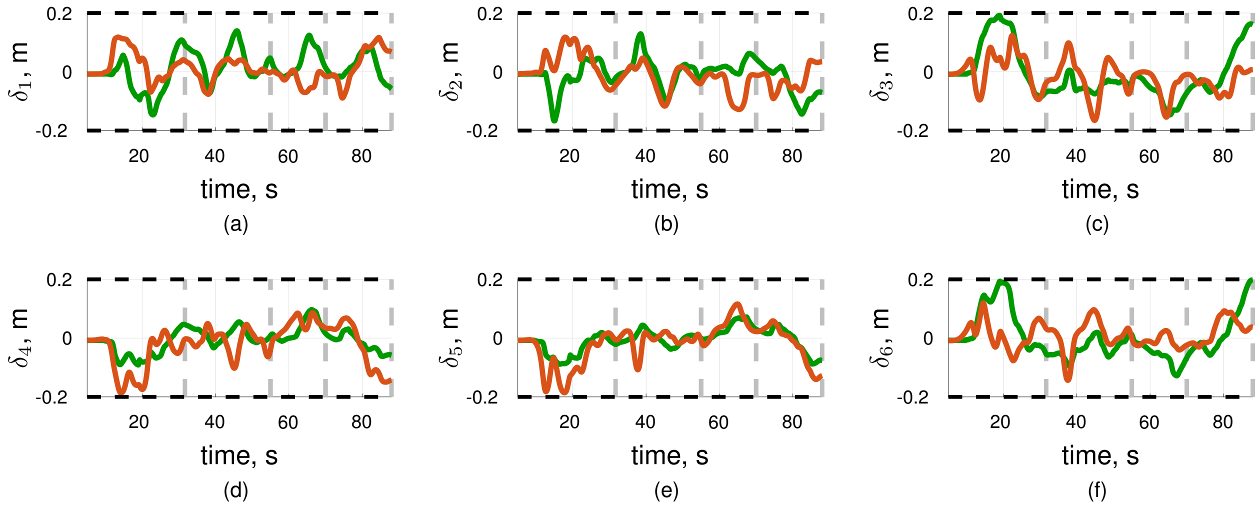

Figure 5: VTE output data of the hexapod deflections for both MCAs, black dashed lines indicate actuator limits.

Table 2: VTE output values for classical washout and optimal MCAs for the four maneuvers.

			M1		M2		M3		M4	
			CWA	OPT	CWA	OPT	CWA	OPT	CWA	OPT
AD	[-]	f_x	0.74	0.30	0.87	0.72	0.71	0.23	0.73	0.21
		f_y	0.62	0.29	0.64	0.30	0.65	0.32	0.59	0.24
		f_z	0.0064	0.014	0.01	0.012	0.0073	0.010	0.0081	0.014
		$\dot{\phi}$	1.96	3.21	2.15	3.40	2.54	3.65	1.50	2.17
		$\dot{\theta}$	1.50	3.72	0.99	1.14	1.99	5.04	1.01	2.23
		$\dot{\psi}$	2.69	2.74	2.58	2.53	1.22	1.23	1.02	1.01
DI	[s]	f_x	0.23	0.00	0.80	-0.080	0.35	0.00	0.02	0.00
CC	[-]	f_x	0.27	0.74	0.22	0.47	0.28	0.78	0.26	0.80
		f_y	0.39	0.71	0.38	0.71	0.39	0.70	0.41	0.76
		$\dot{\psi}$	0.11	0.12	0.10	0.10	0.090	0.090	0.040	0.040
WM	[cm^3]	p_x, p_y, p_z	320.05	248.71	98.34	40.81	39.50	40.88	141.78	42.32
	[rad^3]	ϕ, θ, ψ	56.22	359.42	0.94	5.15	4.19	24.76	20.07	132.00
EC	[J/kg]		75.93	23.80	40.70	10.03	51.49	32.45	648.98	275.02

tion function of the algorithm to consistently outperform the CWA in f_x and f_y by applying slower tilt-coordination, as its values are closer to zero for the AD and closer to 1 for the CC. Both the AD and CC also show a poor reproduction of the yaw cues for both MCAs, which simply could not be reproduced by the motion platform.

The delay indicator (DI) is only shown for f_x, which was in all cases the same as the delay of f_y. As the other signals show a poor correlation, these were not used for the DI. Clearly, the model-predictive control algorithm again performs better, as it does not have inherent phase shift due to the filtering. It is furthermore able to account for future states and is therefore able to compensate for any other delays.

Finally, the two operational metrics are shown at the bottom of Table 2. The workspace management (WM) is separately shown for the translations and the rotations as they have different units. As the CWA has a larger dependency on simulator translation rather than tilt-coordination, a clear dominance is seen in the translational WM, whereas the opposite is true for the rotational WM. For the energy consumption (EC) metric, which is based on a summation of translational velocities and rotational rates, the OPT algorithm again benefits from its large dependency on slow rotation of the platform. Its energy consumption is on average 2.8 times smaller than the CWA. Even when rotational rate thresholds would be disregarded, the energy consumption of the OPT algorithm is still considerably smaller due to the slow rotations, which is a clear advantage.

The main goal of this paper was to show how MCAs can be simulated in a single environment and how MCA quality metrics can be of use to trade-off between MCAs. The VTE allowed for quicker tuning due to its visualization options. The calculated metrics can subsequently be used to trade-off between two or more algorithms. Although beyond the scope of this paper, the next step is to develop a methodology on how these metrics can be combined in a single trade-off based on available MCAs and their configurations, the simulators and use-case.

6. Conclusion

Based on a realistic car driving use-case, an analysis of objective Motion Cueing Algorithms (MCAs) quality metrics was made within a newly developed Virtual Test Environment (VTE). This environment is able to simulate the simulator outputs for different MCA types, which are then compared using a 3D-animation as well as quality metrics. By looking at the absolute difference, cross-correlation, delay, workspace management and energy consumption, a comparison between a classical washout algorithm and a model-predictive control algorithm with a perfect prediction of the future states was made, which helped identifying the strengths and weaknesses of each algorithm. Although an overall analysis on how these metrics should be compared to each other for trade-off is still under development, the methodology applied in this paper is useful to trade-off between MCAs, simulators and use-cases.

VTE 3D-Animation

For a 3D-animation of the Virtual Test Environment in action, please visit:
`http://cs.lr.tudelft.nl/cybernetics/projects/driving-simulator-cueing/`.

References

A. Beghi, M. Bruschetta and F. Maran, **A real time implementation of MPC based Motion Cueing strategy for driving simulators**, in Proceedings of the 51st IEEE Conference on Decision and Control, Maui, Hawaii, 6340–6345, 2012.

S. Casas, C. Inmaculada, J. Vicente Riera and M. Fernández, **Motion-Cueing Algorithms: Characterization of Users' Perception**, *Human Factors*, vol. 57(1): 144–162, 2015.

S. Casas-Yrurzum, C. Portalés-Ricart, P. Morillo-Tena and C. Cruz-Neira, **On the Objective Evaluation of Motion Cueing in Vehicle Simulations**, *IEEE Transactions on Intelligent Transportation Systems*, 1–13, 2020.

D. Cleij, J. Venrooij, P. Pretto, D. M. Pool, M. Mulder and H. H. Bülthoff, **Continuous Subjective Rating of Perceived Motion Incongruence During Driving Simulation**, *IEEE Transactions on Human-Machine Systems*, vol. 48(1): 17–29, 2018.

D. Cleij, **Measuring, modelling and minimizing perceived motion incongruence**, Ph.D. thesis, Delft University of Technology, 2020.

D. Cleij, J. Venrooij, P. Pretto, M. Katliar, H. H. Bülthoff, D. Steffen, F. W. Hoffmeyer and H.-P. Schöner, **Comparison between filter- and optimization-based motion cueing algorithms for driving simulation**, *Transportation Research Part F*, vol. 61(2019): 53–68, 2017.

M. Dagdelen, G. Reymond, A. Kemeny, M. Bordier and N. Maïzi, **MPC Based Motion Cueing Algorithm: Development and Application to the ULTIMATE Driving Simulator**, in Proceedings of the Driving Simulation Conference 2004 Europe, Paris, 221–233, 2004.

M. Dagdelen, G. Reymond, A. Kemeny, M. Bordier and N. Maïzi, **Model-based predictive motion cueing strategy for vehicle driving simulators**, *Control Engineering Practice*, vol. 17(2009): 995–1003, 2009.

F. M. Drop, M. Olivari, M. Katliar and H. H. Bülthoff, **Model Predictive Motion Cueing: Online Prediction and Washout Tuning**, in Proceedings of the Driving Simulation Conference 2018 Europe, Antibes, 71–78, 2018.

F. Ellensohn, D. Hristakiev, M. Schwienbacher, J. Venrooij and D. Rixen, **Evaluation of an Optimization Based Motion Cueing Algorithm Suitable for Online Application**, in Proceedings of the Driving Simulation Conference 2019 Europe, Strasbourg, 1–8, 2019.

F. Ellensohn, M. Schwienbacher, J. Venrooij and D. Rixen, **Motion Cueing Algorithm for a 9 DoF Driving Simulator: MPC with Linearized Actuator Constraints**, *SAE International Journal of Connected and Automated Vehicles*, vol. 2(3): 145–155, 2019.

M. Fischer, A. Labusch, T. Bellmann and C. Seehof, **A Task-oriented Catalogue of Criteria for Driving Simulator Evaluation**, in Proceedings of the Driving Simulation Conference 2015 Europe, Tübingen, 139–150, 2015.

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

M. Grottoli, D. Cleij, P. Pretto, Y. Lemmens, R. Happee and H. H. Bülthoff, **Objective evaluation of prediction strategies for optimization-based motion cueing**, *Simulation*, vol. 95(8): 707–724, 2019.

M. Katliar, K. N. De Winkel, J. Venrooij, P. Pretto and H. H. Bülthoff, **Impact of MPC Prediction Horizon on Motion Cueing Fidelity**, in Proceedings of the Driving Simulation Conference 2015 Europe, Tübingen, 219–222, 2015.

N. A. Pouliot, C. M. Gosselin and M. A. Nahon, **Motion Simulation Capabilities of Three-Degree-of-Freedom Flight Simulators**, *Journal of Aircraft*, vol. 35(1): 9–17, 1998.

I. A. Qaisi and A. Trächtler, **Human in the loop: Optimal control of driving simulators and new motion quality criterion**, *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2235–2240, 2012.

L. D. Reid and M. A. Nahon, **Flight Simulation Motion-Base Drive Algorithms. Part 1: Developing and Testing the Equations**, Tech. Rep. UTIAS 296, University of Toronto, Institute for Aerospace Studies, 1985.

R. Romano, G. Markkula, E. R. Boer, H. Jamson, A. Bean, A. Tomlinson, A. Horrobin and E. Sadraei, **An objective assessment of the utility of a driving simulator for low mu testing**, *Transportation Research Part F*, 34–45, 2019.

J. Venrooij, P. Pretto, M. Katliar, S. Nooij, A. Nesti, M. Lächele, K. de Winkel, D. Cleij and H. H. Bülthoff, **Perception-based motion cueing: validation in driving simulation**, in Proceedings of the Driving Simulation Conference 2015 Europe, Tübingen, 139–150, 2015.