Autoscaling: Minimising Immersion Disruption in Motion Cueing Using Model Predictive Control

Jain, V.; Lazcano, Andrea Michelle Rios; Happee, R.; Shyrokau, B.

DOI
10.82157/dsa/2025/18

Publication date
2025

Document Version
Final published version

Published in
Proceedings of the Driving Simulation Conferences

Autoscaling: Minimising Immersion Disruption in Motion Cueing Using Model Predictive Control

Vishrut Jain[1], Andrea Lazcano[2], Riender Happee[1], Barys Shyrokau[1]

(1) Cognitive Robotics Department (CoR), Faculty of Mechanical Engineering, Delft University of Technology, 2628 CD, Delft, The Netherlands, e-mail: {V.J.Jain, R.Happee, B.Shyrokau}@tudelft.nl
(2) Toyota Motor Europe, Zaventem, Belgium, e-mail: Andrea.Lazcano@toyota-europe.com

Abstract - *Driving simulators aim to replicate real-world vehicle experiences by recreating accelerations acting on occupants using a combination of translational accelerations and tilt-coordination. Due to space constraints, translational accelerations alone are insufficient, and platform tilting generates additional gravitational forces to enhance realism. However, ensuring the tilt motion remains imperceptible is critical to maintaining immersion.*
Model Predictive Control-based motion cueing algorithms demonstrate superior specific force tracking and platform workspace utilization. Despite these benefits, MPC algorithms can exhibit pre-positioning, a phenomenon where the platform tilts prematurely in anticipation of future motion, causing perceptible false cues that disrupt immersion. This phenomenon is particularly noticeable in tilt-coordination due to sustained specific forces.
This work proposes a solution to mitigate pre-positioning by introducing a dynamic scaling factor for tilt-coordination. By scaling down the reference signal for tilt coordination, it stays within the simulator's tilt angle and tilt-rate capabilities, and platform tilt rates are kept below human perception thresholds. The scaling factor is derived from two key parameters: the maximum specific force generated by platform tilt and the tilt rate perception threshold. The reference for specific force is unscaled to optimally use the translational workspace.
This approach enhances driving simulator realism by minimizing the perceptibility of pre-positioning while optimizing specific force recreation. Subjective evaluations also indicate improved immersion, illustrating the effectiveness of the scenario-adaptive Autoscaling MCA.

Keywords: *Motion cueing algorithm, Human-in-the-loop assessment, Pre-positioning, Model predictive control, Automated driving*

https://doi.org/10.82157/dsa/2025/18

Introduction

Driving simulators play a crucial role in replicating real-world driving experiences by providing visual, auditory, vestibular, and haptic cues (Shyrokau, et al., 2018). A key aspect of realism in driving simulation is motion cueing, which is achieved through reproduction of specific forces. Motion cueing algorithms (MCAs) synchronize platform motion with visual stimuli, ensuring a coherent, immersive experience.

MCAs simulate the vehicle dynamics by replicating the accelerations and forces experienced by occupants. However, due to the physical constraints of simulators such as limited motion space, realistic accelerations cannot be generated through translational movement alone. Thus, simulators employ a combination of translational accelerations and tilt-coordination, where platform tilting induces additional gravitational forces. This technique helps recreate the specific forces acting on an occupant's head, mimicking the perception of real-world driving. The classical approach uses tilt-coordination for low-frequency accelerations and platform translation for high-frequency accelerations (Seehof, Durak, and Duda, 2014; Stratulat, et al., 2011).

A key challenge is keeping platform rotations imperceptible, as exceeding human perception thresholds disrupts immersion and makes specific forces feel unnatural. Thus, tilt rates must remain below perceptible limits.

Recent advancements in motion cueing, particularly through model predictive control (MPC)-based algorithms, have improved the fidelity of driving simulations (Bruschetta, Maran, and Beghi, 2016; Khusro, et al., 2020; Lamprecht, et al., 2021). These algorithms predict future platform motion and optimise specific force tracking while accounting for platform workspace constraints. Despite their advantages, MPC-based algorithms are not flawless. The predictive nature of these algorithms can lead to pre-positioning, where the platform moves prematurely in anticipation of future motion. This is particularly disturbing in transitions from steady state driving to dynamic manoeuvres where in steady state the slightest motion is noticeable. While the specific force can

remain zero, the constituent components of the specific force arising from translational acceleration and tilt-coordination can be non-zero, resulting in perceptible platform motion. In a subjective assessment of a frequency-splitting MPC-based cueing algorithm (Jain, et al., 2023), 7 out of 38 participants reported these premature movements as false cues. This phenomenon is also reported in some other works (described as "velocity buffering") (Biemelt, et al., 2021; Grottoli, et al., 2019). However, due to the objective nature of these studies, no comment on subjective feedback is made in these works.

In moving-base driving simulators, it is a common trend to scale down the reference accelerations by a factor of 0.2 to 0.6 to better fit the available workspace (Bellem, et al., 2017; Lamprecht, et al., 2021). Scenario-specific scaling factors are typically used to precondition the reference signal and optimise tracking within the simulator's physical limits. The quality of the driving simulation heavily depends on this factor. In MPC-based MCAs, the scaling factor is generally kept constant throughout the simulation, requiring a conservative choice that suits the entire scenario rather than optimizing for different sections. This paper addresses the challenge of prepositioning by proposing a method to dynamically adjust the scaling factor for tilt-coordination based on the simulator capabilities. By ensuring that platform tilt remains within human perception thresholds, prepositioning is minimized, enhancing the realism and immersion of the driving experience, while eliminating the need for scaling factor tuning for every scenario.

Methodology

Figure 1 presents the proposed algorithm's structure, where the MPC receives two separate references: for tilt coordination and for total specific force. This ensures that a reproducible component of the specific force is provided to tilt coordination as a reference, preventing premature platform movements.

The MPC controls four degrees of freedom (DoFs) of the platform motion, with vehicular roll and pitch assumed to be negligible for the vehicle dynamics. Hence the applied platform pitch and roll serve only to recreate vehicle acceleration through tilt coordination. However, if available, vehicle pitch and roll data can be directly incorporated into the rotational channels as additional input. Additionally, since yaw does not affect the specific force generation, the yaw controller can be decoupled from the MPC. Thus, the yaw motion is managed separately using a simple washout filter to avoid additional computational expense.

References for the Algorithm

Reference for Tilt Coordination

The algorithm aims to prevent platform prepositioning by addressing its root cause in MPC: anticipatory motion triggered when required accelerations exceed what can be achieved within tilt rate limits. Unable to generate the desired force in time, the platform moves early, resulting in unrealistic specific force cues. This issue is mitigated by providing the tilt coordination with a reference signal that can be accurately followed while staying within the tilt-rate limit. Thus, the tilt-coordination receives a scaled-down low-pass vehicle acceleration data as a reference.

To enforce strict reference tracking, a high penalty is applied to the tilt coordination term in the objective function, preventing tilt components from compromising the accurate recreation of the total specific force.

Dynamic Scaling Factor Design

To ensure accurate reference signal recreation, the required specific force and its rate of change must be less than or equal to the simulator's potential. Two scaling factors are derived, based on maximum achievable tilt angle and maximum rate change of tilt angle.

The first scaling factor, k_θ, is based on the maximum tilt angle, ensuring the reference specific force remains within the platform's tilt coordination capability:

$$\max|f_{spec,ref}| \leq max|g\ sin(\theta_{tilt})| \quad (1)$$

$$k_\theta = |\frac{g\ sin(\theta_{lim}|)}{\max|f_{spec,ref}}| \quad (2)$$

where 'θ_{lim}' is the maximum platform tilt angle.

The second scaling factor, k_ω , is derived from the tilt rate constraint to prevent false cues.

$$\max|\dot{f}spec, ref| \leq \max|\omega_{tilt} g\cos(\theta_{tilt})| \quad (3)$$

here, the maximum value of '$cos(\theta_{tilt})$' is 1, and the tilt rate is limited to the perception threshold, ω_{thd}, $(3^\circ/s)$. The scaling factor is thus given by:

$$k_\omega \leq \omega_{thd}\frac{g}{\max \dot{f}_{spec,ref}} \quad (4)$$

$$k = min(k_\theta, k_\omega, 1) \quad (5)$$

where k is the dynamic scaling factor that scales down the reference signal for the tilt coordination. In this work, for the choice of scaling factor the reference signal within the MPC prediction horizon is considered.

Reference for Total Specific Force

In this work, unscaled vehicular accelerations are used as the reference for total specific force. Given that tilt coordination has a separate reference, the MPC reproduces the remaining specific force solely through translational accelerations. This ensures optimal use of translational space while employing tilt coordination to enhance realism by complementing specific force reproduction through translational motion.

Hexapod dynamics

The motion of the hexapod platform is defined in a state space form to facilitate implementation in the MPC. The base states include hexapod position (s_{hex}) and angular orientation (θ_{hex}). These base states are added to the state-space model with the relation

$$\dot{x}_{hex} = A_{hex}x_{hex} + B_{hex}u_{hex} \quad (6)$$

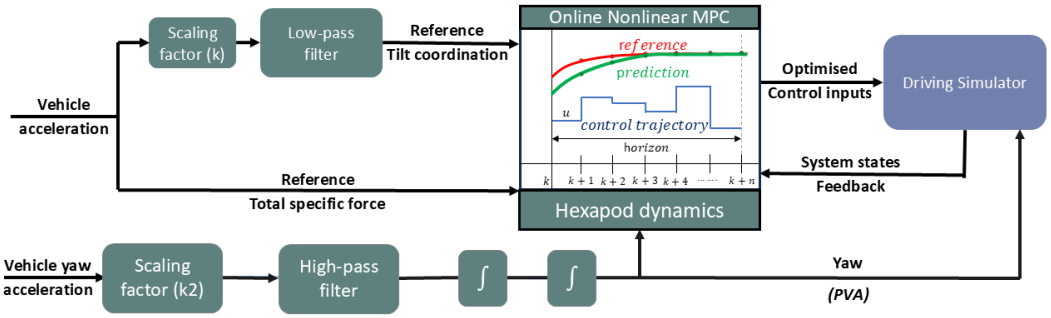

Figure 1: Structure of the five-DoF Autoscaling MCA

where the state vector, x_{hex}, comprises of the position, s_{hex}, translational velocity, v_{hex}, angular orientation, θ_{hex}, and angular velocity, ω_{hex}, of the hexapod and the input vector, u_{hex}, comprises of translational acceleration, a_{hex} and angular acceleration, α_{hex}. The matrix A_{hex} and B_{hex} represent the double integrator system of the inputs, adapted from (Qazani, et al., 2019).

As the algorithm is designed for both longitudinal and lateral degrees of freedom, each state comprises of components in x and y directions (roll and pitch for orientation). In this study, positive values correspond to forward, left, and upward orientations along the x, y, and z axes, with counterclockwise rotations indicated as positive.

Objective Function

The objective function structure is similar to that of the frequency-splitting algorithm in (Jain, et al., 2023), but with changes in output terms and references. Instead of using total specific force, translational motion and tilt coordination as output terms, this algorithm includes only tilt coordination and total specific force.

$$\begin{aligned} J_c = \quad & [\underbrace{(Y(x_k,u_k)-\hat{Y}_k)^T W_Y (Y(x_k,u_k)-\hat{Y}_k)}_{output\ terms} \\ & + \underbrace{(X_k-\hat{X}_k)^T W_X (X_k-\hat{X}_k)}_{state\ terms} + \underbrace{J_k^T W_J J_k}_{jerk\ terms} \\ & + \underbrace{(U_k)^T W_U (U_k)}_{input\ terms} + \underbrace{\delta^T w_\delta \delta}_{slack\ term} \ ] \end{aligned} \tag{7}$$

$$Y(x_k,u_k) = [f_{spec}\ G_{loc}] \tag{8}$$

$$W_Y = [w_{f,spec}\ w_{G,loc}] \tag{9}$$

$$J_k = [j_{trans}\ j_{ang}]\ W_J = [w_{j,trans}\ w_{j,ang}] \tag{10}$$

Penalising jerk in the cost function of an MPC-based MCA is a well established practice to reduce oscillations in the specific force. We approximate jerk using the acceleration change over time-steps divided by the time-step $j(k) = \frac{a(k)-a(k-1)}{Ts}$. This approach avoids the need to add jerk as a system state, reducing computational complexity. The state term $(x_k - \hat{x}_k)$ introduces 'washout' by bringing the platform back to its neutral position $\hat{x}_k$. The input term (U_k) penalises high input values.

In Equation 7, the user defined (tunable) parameters are the weighting matrices W_Y, W_U, W_X and w_δ. W_X is a diagonal matrix with weights corresponding to the states being its diagonal elements. The washout weights w_s and w_θ are time varying and are defined in a later subsection.

The slack variable, δ, represents the deviation of the tilt-rate from the soft constraint limit (perception threshold).

Table 1: Limits for the simulator and the used MPC limits

Quantity	Platform physical limit	Defined MPC limit
θ_{hex}	$\pm 30deg$	$\pm 20deg$
v_{hex}	$\pm 7.2m/s$	$\pm 7.2m/s$
a_{hex}	$\pm 9.81m/s^2$	$\pm 9.81m/s^2$
s_{hex}	$\pm 0.5m$	$\pm 0.3m$

Platform Constraints

Unlike a real vehicle, a driving simulator is restricted to a maximum displacement and maximum tilt angle, specific to the simulator used. This work uses the workspace limitations of Delft Advanced Vehicle Simulator (DAVSi).

DAVSi is a 6-DoF moving-based driving simulator (Khusro, et al., 2020), capable of generating acceleration up to 1 g in all directions and can simulate motions up to the frequency of 10 Hz. The considered limits of the platform motion are given in Table 1. As the algorithm includes yaw washout as a separate controller, conservative limits are chosen for translational and rotational displacement. Additionally, the soft constraint used to define the tilt-rate perception limit is formulated as

$$\begin{aligned} -\omega_{thd} &\leq \omega_{hex} + \delta \\ \omega_{hex} - \delta &\leq \omega_{thd} \\ 0 &\leq \delta \end{aligned} \tag{11}$$

where ω_{thd} is the perception threshold limit (3 deg/s). δ is a positive slack variable which is penalised in the cost function to keep its value low. Hence, the tilt-rate soft constraint limit is allowed to be violated, however, any violation of the constraint is penalised, to minimise it.

Workspace management

The MPC considers these constraints over the prediction horizon to optimally use the workspace and generate realistic motion. Two additional strategies

are employed here for effective workspace management: washout and dynamic constraints.

Washout: The simulator platform has the maximum potential of recreating the specific forces at its neutral position. To ensure the platform remains near its neutral position, we penalize its states in the cost function. In this work, we use non-linear weights (based on the platform orientation and position) for the washout instead of constant weights. This allows a single non-linear setting for all scenarios rather than tuning the washout weights for each scenario. The non-linear weights are defined as

$$w_s = \frac{k_1}{k_2 * (|s_{hex}| - s_{lim})^2 + \Delta} \quad (12)$$

$$w_\theta = \frac{k_3}{k_2 * (|\theta_{hex}| - \theta_{lim})^2 + \Delta} \quad (13)$$

where k_1, k_2 and k_3 define the shape of the weight function, s_{lim} and θ_{lim} are the defined limits for the platform for displacement and tilt angle. Δ (here 0.01) is a small value added to the denominator to avoid singularity. The selected values are $k_1 = 1$, $k_2 = 50$ and $k_3 = 0.1$, these values were manually tuned to ensure that the penalisation is low near the neutral position, while high, close to the platform limits.

Dynamic constraints: In this study, we incorporate dynamic bounds on the platform position and orientation via the constraints proposed in (Fang and Kemeny, 2012), as 'braking constraints'. The formulation of the constraints is

$$s_{hex,min} \leq s_{dy} \leq s_{hex,max} \quad (14)$$

$$\theta_{hex,min} \leq \theta_{dy} \leq \theta_{hex,max} \quad (15)$$

$$s_{dy} = s_{hex} + c_v v_{hex} T_{dy,s} + 0.5 c_u a_{hex,tran} T_{dy,s}^2 \quad (16)$$

$$\theta_{dy} = \theta_{hex} + c_w \omega_{hex} T_{dy,\theta} + 0.5 c_u a_{hex,rot} T_{dy,\theta}^2 \quad (17)$$

where, $c_v = 1, c_w = 1, c_u = 0.45, T_{dy,\theta} = 0.5$, $T_{dy,p} = 2.5$ and s_p, θ_p limits are $0.3\ m$ and $20\ deg$ respectively. These values were adopted from (Munir, et al., 2017).

Yaw Channel

The fifth DoF, yaw, is controlled separately using a parallel washout channel, ensuring reduced computational complexity. The first-order high-pass filter used for this purpose is given as

$$HP = \frac{s}{s + 2\pi\nu_{yaw}} \quad (18)$$

where ν_{yaw} is the cutoff frequency for the high pass filter. In this work, we use the value of 0.0159 Hz for this cutoff frequency.

Weight settings

To ensure a fair comparison the penalisation weights on tilt angle, translational displacement, translational jerk, angular jerk and slack variable are kept the same as used in (Jain, et al., 2023). As the output terms vary in the cost terms of the two MCAs, the weights for these terms also change. In the Autoscaling MCA priority is given to the tilt-coordination to follow its reference accurately, hence the weight for tilt-coordination reference tracking is selected to be 5 times higher than for specific force tracking (the weight used for specific force tracking is unity). However, as the tilt-coordination is forced to begin at the onset of the manoeuvres, pre-positioning in the translational workspace can be observed by the occupant of the simulator. To resolve this, the translational velocity of the platform is penalised. Varying the velocity penalty from 0 to 1 shows an insignificant difference. However, a penalty of 10 resulted in an improved solution, with reduced pre-positioning and limited adverse effect on the rendered profile.

Yaw washout

The tuning parameter for the yaw washout was the cut-off frequency of the washout filter. The cut-off frequency for the yaw washout was varied between $0.0159\ Hz$ (0.1 rad/s) to $0.1592\ Hz$ (1 rad/s).

Since human perception is primarily sensitive to rotational velocities, the selection of the cut-off frequency is chosen based on desirable rotational velocity tracking. It is also essential to ensure that the yaw angle remains within acceptable limits ($20\ deg$ for this work). Based on these considerations, a cut-off frequency of $0.0159\ Hz$ is chosen for this study.

Benchmarking

As pre-positioning was identified to occur with the subjective validation of the frequency splitting algorithm (Jain, et al., 2023), it is chosen as the benchmark for this study.

Scenario description

To compare the realism and immersion of the algorithm in the driving simulator, real-vehicle driving data from an experiment conducted by our group at the Valkenburg track was utilised (Papaioannou, et al., 2023). Given the experiment's duration, specific sections were chosen for this study, including multi-turns with acceleration and braking, a slalom maneuver, and a lane-change maneuver, capturing a variety of naturalistic driving scenarios.

The total scenario duration was deliberately limited to 110 seconds to minimize the risk of simulator sickness and potential bias in the study.

Fidelity criteria

In this paper, the algorithm is evaluated both subjectively and objectively. For the objective comparison, the shape similarity, root mean squared of the specific force and the timing of the initiation of the tilt-coordination compared to the onset of the vehicle maneuvers is considered. On the other hand, for subjective assessment the participants were asked to rate the realism of the ride based on realism of cornering and braking, the amount of unnatural instances/false cues observed during the ride and are asked for any instance of pre-positioning observed.

Human-in-the-loop evaluation

This section describes the human-in-the-loop driving simulator experiment and its subjective evalua-

tion. Perceived driving simulator fidelity was evaluated from the perspective of passive users, representative of users of automated vehicles. In pilot studies, we even aimed to evaluate perceived fidelity with the eyes off the road using a tablet for a non-driving task. However, with vision limited to the vehicle interior, the MCAs were not perceived as representative of driving. This was resolved by adding exterior vision and instructing participants to observe the road.

Experimental procedure

All participants gave their informed consent prior to participating. The Human Research Ethics Committee of TU Delft, The Netherlands, approved the experiment protocol under application number 3965.

In total, 8 participants from the pool of students and employees of TU Delft participated in the study (mean age: 25.5, std: 2.5 years, 3 females, 5 males). All participants were subjected to both the proposed algorithm and frequency splitting drives sequentially within one session, to compare the subjective evaluation of the realism and feel of the MCAs.

Before the experiment, the participants underwent a concise briefing session to familiarise themselves with the questionnaire and to understand the objective of the experiment. During the experiment, two-way communication was established between the experimenter and the participant via bluetooth headphones and microphones.

The visualisation was projected onto a screen in front of the simulator, with side windows and windshield partially covered to block peripheral views and eliminate cues revealing platform motion. Dynamic visual compensation adjusted the projection to match the platform's movements, ensuring the visuals stayed aligned with the motion felt by the participants despite the screen being outside the cockpit.

Following the briefing, participants underwent a series of six two-minute drives in fully automated mode with minimal time between drives to facilitate a clear comparison. After each drive, participants completed an absolute grading questionnaire to assess various aspects of the driving experience. These questions can be found in Table 4. To ensure fair comparison, the sequence of algorithms was varied for different participants, maintaining either the pattern (A B A B A B) or (B A B A B A).

The first round for each algorithm aimed to immerse participants in the simulation environment and was not graded. Participants provided ratings in a 5-point scale for the second and third rounds based on the provided questionnaire. At the end of the experiment, the participants were asked to fill in a comparative questionnaire.

Results and discussion

This section presents the results of both objective and subjective evaluations of the two algorithms, followed by an analysis of the findings.

Objective evaluation

Specific force tracking

Figure 2 and Figure 3 show the comparison between the profiles rendered through the Autoscaling MCA and the benchmarking frequency-splitting algorithm for real driving data. It is worth mentioning that the Autoscaling MCA receives unscaled vehicle accelerations, whereas the frequency-splitting algorithm used a reference scaled down by a factor of 0.3 for longitudinal acceleration and 0.4 for lateral acceleration. The reference signal presented in Figure 2 and Figure 3 corresponds to the reference chosen for the frequency splitting algorithm, where coincidentally, the rendered profiles via the two algorithms have a similar magnitude.

For the frequency-splitting algorithm, the RMSE of the longitudinal specific force tracking is 0.0466 while for the lateral tracking it is 0.1943. On the other hand, for the Autoscaling MCA the RMSEs for longitudinal and lateral specific force tracking are 0.1210 and 0.2459 respectively.

Additionally, the shape similarity factors for the rendered profiles were analysed. For the longitudinal direction, the shape similarity factor is 0.9787 for the frequency-splitting algorithm and 0.9312 for the Autoscaling MCA. Similarly, for the lateral direction, the frequency-splitting algorithm achieves a shape similarity coefficient of 0.9519, whereas the Autoscaling MCA yields 0.9208.

Workspace utilisation

A comparison of workspace utilisation, based on profiles generated by the Autoscaling MCA and the benchmarking frequency-splitting algorithm, is presented in Table 3. The Autoscaling MCA utilises the translational workspace more, while the frequency-splitting algorithm exhibits higher usage of the rotational workspace. Although both algorithms respect the tilt-rate perception threshold, the FS MCA achieves higher platform tilts by initiating tilt earlier, causing pre-positioning, whereas Autoscaling MCA avoids this by reducing the reference specific force, eliminating the need for early motion.

Pre-positioning

In Figure 2, it can be observed that, for the Autoscaling MCA the translational accelerations generate a higher component of specific force compared to the frequency splitting MCA. Additionally, the onsets of the manoeuvres are also at the correct instances, however, due to the gradually varying and continuous nature of longitudinal acceleration, this difference is not distinctly observed. On the other hand, it can be observed in Figure 3, that for lateral motion, the frequency-splitting algorithm exhibits pre-positioning within the rotational workspace at various instances to prepare for future motion. In contrast, the Autoscaling MCA significantly reduces pre-positioning. The motion onset in the Autoscaling MCA occurs at the correct instances, with a higher contribution from translational motion compared to the frequency-splitting algorithm.

Subjective evaluation

This subsection compares the two algorithms based on the conducted experiment. Table 4 presents the

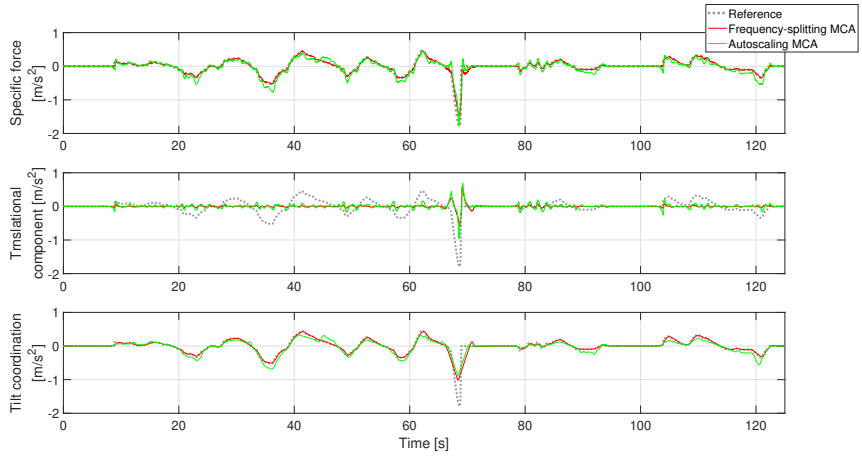

Figure 2: Comparison of the response of Autoscaling and Frequency-splitting MCAs: longitudinal acceleration.

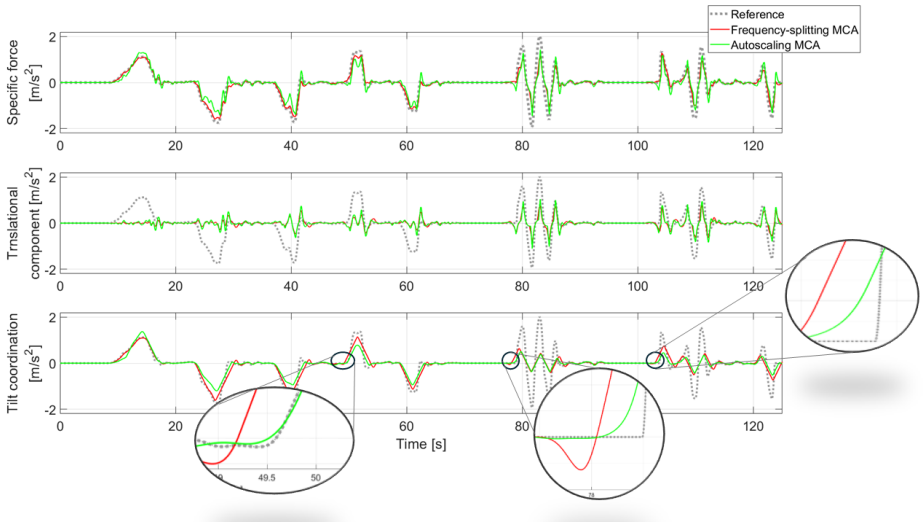

Figure 3: Comparison of the response of Autoscaling and Frequency-splitting MCAs: lateral acceleration.

obtained responses in detail, including their statistical significance.

Coherence of the motion with the video

The Autoscaling MCA received an average rating of 3.75 out of 5 (75% realism), while the frequency-splitting algorithm scored 3.25 (65% realism). Both algorithms provided a coherent ride experience relative to the video, with no statistically significant difference between them.

Post-experiment verbal feedback indicated that the limiting factor in perceived coherence was the video quality, rather than the platform's motion.

Cornering realism

The Autoscaling MCA received an average rating of 4.0 out of 5, with a median of 4.0, whereas the frequency-splitting algorithm was rated 3.0 on average, with a median of 3.0. A statistically significant difference was observed between the ratings, indicating a preference for the Autoscaling MCA in terms of cornering realism.

Participants reported instances of pre-positioning in the frequency-splitting algorithm during post-experiment verbal feedback, which may have influenced its lower realism rating.

Table 2: Tracking Performance for the two algorithms

Algorithm	Performance				
	Specific force tracking			Shape similarity factor	
	RMSE long. [m/s^2]	RMSE lat. [m/s^2]	RMSE tot. [m/s^2]	long. [-]	lat. [-]
Autoscaling MCA	0.1210	0.2459	0.2740	0.9312	0.9208
Frequency-splitting MCA	0.0466	0.1943	0.1998	0.9787	0.9519

Table 3: Workspace utilisation for the two algorithms

Algorithm	Workspace utilisation			
	RMS displacement [m]	RMS velocity [m/s]	RMS angular displacement [deg]	RMS angular velocity [deg/s]
Autoscaling MCA	0.0423	0.0770	2.2339	1.8113
Frequency-splitting MCA	0.0334	0.0682	2.6181	2.0091

Table 4: Subjective evaluation and statistical analysis (on a scale from 0-5)

Criterion	Algorithm	All data				
		mean	std.	median	p-value	significance
How closely did the ride's motion correspond to the video?	AU	3.75	0.71	4	0.0796	No
	FS	3.38	0.52	3		
How close did the cornering feel compared to a real car?	AU	4	0.76	4	0.0072	Yes
	FS	3	0.53	3		
How realistic did the deceleration feel compared to a real vehicle drive?	AU	3.5	0.53	3.5	0.1705	No
	FS	3.75	0.46	4		
Aggressiveness of Section 1 : multi-turn	AU	3.75	0.46	4	0.0750	No
	FS	3.125	0.64	3		
Aggressiveness of Section 2 : slalom	AU	2.5	0.76	2	0.5983	No
	FS	2.625	0.52	3		
Aggressiveness of Section 3 : lane change	AU	2.75	0.46	3	0.0112	Yes
	FS	2.125	0.35	2		
Was the ride disorienting or sickening?	AU	2	0.89	2	0.3506	No
	FS	1.75	0.76	2		
Were there any unnatural motions that did not match real driving?	AU	2.25	0.46	2	0.0025	Yes
	FS	3	0.76	3		

Realism of braking

The Autoscaling MCA received an average rating of 3.5 (median: 3.5), while the frequency-splitting algorithm scored 3.75 (median: 4). However, the difference was not statistically significant. In post-experiment feedback, participants found the frequency-splitting algorithm more comfortable, whereas the Autoscaling MCA felt abrupt during acceleration and braking.

Overall realism

In the comparative questionnaire, 7 out of 8 participants found the Autoscaling MCA more realistic overall, showing a clear preference for its motion cueing.

Aggressiveness

The Autoscaling MCA was rated more aggressive in the first and third sections, while the second section scored higher for the frequency-splitting algorithm. However, statistical significance was found only in the third section, preventing a general conclusion on the Autoscaling MCA's aggressiveness, especially given its dynamic scaling. Detailed statistics are provided in Table 4.

Despite this, 6 out of 8 participants in the comparative questionnaire perceived the Autoscaling MCA as more aggressive overall.

Sickness

After the 2-minute ride, participants rated the algorithms on their sickening or disorienting effects. The Autoscaling MCA received an average rating of 2, while the frequency-splitting MCA scored 1.75, with both medians at 2. This indicates no significant difference in the motion profiles' sickening effects.

Unnatural motion/ false cues

The Autoscaling MCA received an average rating of 2.25 (median: 2), while the frequency-splitting MCA scored 3.0 (median: 3). This statistically significant difference suggests that the Autoscaling MCA generates fewer false cues.

In the comparative questionnaire, seven participants reported more false cues with the frequency-splitting MCA, while one found the Autoscaling MCA to produce higher unnatural motions.

Additionally, six out of eight participants observed pre-positioning in the frequency-splitting algorithm, whereas only one reported it in the Autoscaling MCA, occurring during a single corner.

Discussion

The Autoscaling MCA aims at eliminating two major issues encountered in MPC-based MCAs, one being pre-positioning, where the platform prepares itself for the future motion, moving prematurely. The second issue being the necessity to scale down the reference signal to precondition it for desirable recreation of specific scenarios. The performance of traditional MCAs depends highly on the preconditioning (scaling) parameters. In this work, we present the Autoscaling MCA which automatically derives a time varying scaling factor for the tilt coordination reference.

As shown in Figure 3, the Autoscaling MCA reduces roll pre-positioning in lateral motion, though it does not eliminate it entirely. This is due to total specific force error minimization in the objective function. A higher penalty on tilt coordination improves adherence to the reference, leading to reduced pre-positioning, as the platform produces dynamically scaled-down accelerations via tilt coordination.

The scaling factor is determined based on the platform's capability to reproduce the maximum specific force via tilt coordination, while ensuring that the resulting rotational velocity remains within human perception thresholds. In this context, the maximum allowable tilt rate governs the maximum rate of change of specific force, thereby playing a critical role in

defining the scaling factor. This ensures the simulator always has the potential to generate the scaled-down reference for tilt coordination, maximising the simulator's potential in recreating specific forces. Since the platform can always achieve its reference tilt coordination, premature tilting is avoided, reducing pre-positioning.

Referencing tilt coordination and total specific force ensures that the algorithm tracks the scaled-down tilt reference while using translational motion to recreate the remaining specific force. This is evident in workspace utilization, where the Autoscaling MCA relies less on tilt workspace and more on translational workspace compared to the frequency-splitting algorithm. As a result, tilt coordination serves as a supporting mechanism for generating higher specific forces rather than being the primary contributor. Studies have shown that translational motion has a greater impact on realism than tilt coordination (Hettinger and Riccio, 1992; Jamson, 2010).

Based on the objective performance indicators, however close, the frequency splitting algorithm showed a better performance, with higher shape similarity and lower RMSE for the specific force tracking. Both the algorithms achieved high values for their performance, however the objective KPIs indicated frequency splitting algorithm to perform better. However, these KPIs are based on the total specific force, combining tilt coordination and translational accelerations. Thus, conflicting motions in translational workspace and tilt coordination may create a motion that does not correspond to the actual specific force. One such case is when the tilt and translational components cancel each other to create a zero net specific force. This corresponds to the no motion case, however, the opposite motions of tilt and translation can still be picked up by the participant.

Hence, in this work, the tilt coordination is treated as a separate reference with higher penalisation, and the rest of the specific force is left for the translational motion to generate. The reference on tilt coordination is scaled down rather than that on the translation motion, as tilt produces sustained forces on the occupant's head, thus a premature motion creates a sustained acceleration perception which is probable to be perceived more than a pre-positioning motion in translational workspace.

Conclusion

In this work, an Autoscaling MCA framework for an MPC-based MCA is developed, enabling automatic and adaptive scaling of vehicular accelerations or reference specific force. This approach minimizes the need for aggressive downscaling, thereby preserving motion fidelity.

The dynamic scaling factor ensures reference tracking without prematurely tilting the platform, generating the demanded specific force, reducing pre-positioning.

Human-in-the-loop evaluation demonstrated that 87.5% of participants (7 out of 8) perceived the Autoscaling MCA as more realistic, with only one participant noticing platform pre-positioning compared to six participants for the frequency-splitting algorithm. Interestingly, this preference occurred despite the frequency-splitting approach exhibiting lower RMSE in specific force and higher shape correlation factors. This highlights the influence of additional perceptual aspects-such as the timing of platform motion onset, which are not adequately captured by conventional objective metrics.

The Autoscaling MCA also utilizes translational workspace more effectively and relies less on tilt coordination, enhancing realism by replicating vehicle-like translational accelerations.

References

Bellem, H., Klüver, M., Schrauf, M., Schöner, H.-P., Hecht, H., and Krems, J. F., 2017. Can we study autonomous driving comfort in moving-base driving simulators? A validation study. *Human factors*, 59(3), pp. 442–456.

Biemelt, P., Böhm, S., Gausemeier, S., and Trächtler, A., 2021. Subjective Evaluation of Filter- and Optimization-Based Motion Cueing Algorithms for a Hybrid Kinematics Driving Simulator. In: *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1619–1626. https://doi.org/10.1109/SMC52423.2021.9658974.

Bruschetta, M., Maran, F., and Beghi, A., 2016. A nonlinear, mpc-based motion cueing algorithm for a high-performance, nine-dof dynamic simulator platform. *IEEE Transactions on Control Systems Technology*, 25(2), pp. 686–694.

Fang, Z. and Kemeny, A., 2012. Explicit MPC motion cueing algorithm for real-time driving simulator. In: *7th International Power Electronics and Motion Control Conference*. Vol. 2. IEEE, pp. 874–878.

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

Hettinger, L. J. and Riccio, G. E., 1992. Visually induced motion sickness in virtual environments. *Presence: Teleoperators and Virtual Environments*. MIT Press, One Rogers Street, Cambridge, MA 02142-1209, USA; journals-info@mit.edu.

Jain, V., Lazcano, A., Happee, R., and Shyrokau, B., 2023. Motion Cueing Algorithm for Effective Motion Perception: A frequency-splitting MPC Approach. In: *Proceedings of the Driving Simulation Conference 2023 Europe, Antibes, France*, pp. –.

Jamson, A. H. J., 2010. *Motion cueing in driving simulators for research applications*. University of Leeds.

Khusro, Y. R., Zheng, Y., Grottoli, M., and Shyrokau, B., 2020. MPC-based motion-cueing algorithm for a 6-DOF driving simulator with actuator constraints. *Vehicles*, 2(4), pp. 625–647.

Lamprecht, A., Steffen, D., Nagel, K., Haecker, J., and Graichen, K., 2021. Online model predictive motion cueing with real-time driver prediction. *IEEE Transactions on Intelligent Transportation Systems*, 23(8), pp. 12414–12428.

Munir, S., Hovd, M., Fang, Z., Olaru, S., and Kemeny, A., 2017. Complexity reduction in motion cueing algorithm for the ULTIMATE driving simulator. *IFAC-PapersOnLine*, 50(1), pp. 10729–10734.

Papaioannou, G., Cvetkovic, M., Messiou, C., Kotian, V., Shyrokau, B., and Happee, R., 2023. A novel experiment to unravel fundamental questions about postural stability and motion comfort in automated vehicles. In: *Comfort Congress*, p. 123.

Qazani, M. R. C., Asadi, H., Khoo, S., and Nahavandi, S., 2019. A linear time-varying model predictive control-based motion cueing algorithm for hexapod simulation-based motion platform. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(10), pp. 6096–6110.

Seehof, C., Durak, U., and Duda, H., 2014. Objective motion cueing test-experiences of a new user. In: *AIAA Modeling and Simulation Technologies Conference*, p. 2205.

Shyrokau, B., De Winter, J., Stroosma, O., Dijksterhuis, C., Loof, J., Paassen, R. van, and Happee, R., 2018. The effect of steering-system linearity, simulator motion, and truck driving experience on steering of an articulated tractor-semitrailer combination. *Applied Ergonomics*, 71, pp. 17–28.

Stratulat, A., Roussarie, V., Vercher, J.-L., and Bourdin, C., 2011. Improving the realism in motion-based driving simulators by adapting tilt-translation technique to human perception. In: *IEEE Virtual Reality Conference*, pp. 47–50.