Driving Simulator Motion Cueing Development for the Non-linear Handling Regime

Nikhil J.I. Garrett and Matthew C. Best
Loughborough University, UK

Dept. Aeronautical and Automotive Engineering, Stewart Miller Building, Loughborough University
Loughborough, Leicestershire, LE11 3TU, UK
Phone: +44 1509 227264
Fax: +44 1509 227275
E-mail:N.J.I.Garrett@lboro.ac.uk

This paper presents the results of a study to evaluate the suitability of two new driving simulator motion cueing algorithms for driving in the non-linear region of vehicle behaviour. The new algorithms are a Model Predictive Control (MPC)-based algorithm with constraints based on actuator states, and an algorithm based around the use of the vehicle sideslip angle as the demanded platform yaw angle. The results indicate that the body sideslip algorithm is preferred to the MPC and standard filter-based algorithms, with the more experienced participants also expressing a liking for the MPC algorithm.

Topics / Driver-vehicle Systems & Driving Simulator

1. INTRODUCTION

Motion-based driving simulators generally have a motion workspace far smaller than that required for full restitution of vehicle motions; this is especially true of those based on six degrees-of-freedom (DOF) Stewart platforms where maximum translation is of the order of 0.5m. It is therefore not possible to reproduce the large low-frequency motion that is characteristic of road vehicle manoeuvres and some transformation from the vehicle motion to the desired platform motion is required; this transformation is known as the motion cueing algorithm. Several approaches to motion cueing have been suggested, from high-pass filtering to optimal control formulations. Previous work by Newton and Best [1] has shown that the presence of motion improves the perceived simulator fidelity; Siegler et al [2] show that motion prompts drivers to exhibit behaviour closer to that in the real vehicle.

However, there is still a disparity between the way people drive the simulator and the way they drive a real vehicle; or perhaps more appropriately drivers are unable to drive the simulator as they would a real vehicle due to differences in sensory feedback. Since visual, audio and steering feedback can be provided with a high level of fidelity, the platform motion remains the major stumbling block in reducing the gap between simulation and reality. In particular, the lack of information from the platform motion seems to be a problem for driving in the non-linear region of vehicle behaviour.

The ongoing research aims to find an improved motion cueing algorithm that presents better information about the vehicle state to the driver and brings their control strategy closer to that in the real vehicle. In addition the authors intend to address the lack of published results of comprehensive tests with human drivers. Previous work by the authors [3] compares the effectiveness of different motion cueing algorithms for driving in the linear handling regime using a broad range of test subjects. This paper presents the results of a study comparing three simulator cueing algorithms for driving in the non-linear handling regime with more focus on experienced drivers. The first cueing algorithm is based on the classical linear filter-based method with adaptive gains and is the standard cueing supplied with the simulator. The other two algorithms have been developed as part of the research; one based on Model Predictive Control (MPC) which, unlike other MPC algorithms, uses actuator states as constraints, and a novel algorithm based around using the vehicle sideslip angle as the platform yaw demand.

The cueing algorithms to be evaluated are presented in section 2. The details of the experiments are provided in section 3, and the results are presented and discussed in section 4.

2. CUEING ALGORITHMS

2.1 Classical algorithm

The well-established classical cueing algorithm (see e.g. Schmidt and Conrad, [4]) attenuates and high-pass filters the vehicle accelerations in order to obtain suitable platform acceleration commands. Additionally, the lateral and longitudinal accelerations are low-pass filtered and added to the roll and pitch DOFs, such that the orientation of the gravity vector relative to the driver gives an illusion of sustained accelerations; this process is known as tilt coordination.

A variant on the method introduces adaptive gains in series with the high-pass filters; these gains are adapted on-line to minimise a cost on motion perception error and platform excursion (see e.g. Nahon et al, [5]). The effect of this is a reduction in the attenuation of low magnitude accelerations and also a reduction in false cues, i.e. platform accelerations that are of opposite sign to the current vehicle acceleration.

The tuning of the classical algorithm is not a straightforward task, as there is no obvious relationship between the choice of filter cut-off frequencies and damping ratios and the perceived motion quality. The addition of adaptive gains makes the problem even worse with cost weightings and descent algorithm parameters to be chosen as well.

2.2 Model Predictive Control (MPC) algorithm

Optimal control approaches to motion cueing, the most popular of which is based on Linear Quadratic Regulator (LQR) control as proposed by Sivan et al [6], define motion cueing as a tracking problem where the motion perceived in the simulator should track the motion perceived in the real vehicle. Platform washout is achieved by applying a cost to platform excursion. Calculation of the perceived accelerations from actual accelerations is achieved by including dynamic models of the human vestibular system. Although several complex vestibular models have been proposed, it is possible to simplify these to obtain second- or third-order systems that capture the large scale behaviour as discussed by Telban and Cardullo [7].

Recently, motion cueing using model-based predictive control has been proposed. Dagdelen [8] proposes an unusual variant on the tracking formulation, the result of which is an algorithm that matches vehicle acceleration until a platform limit is imminent, then returns to centre below the human perception threshold. Augusto [9] describes an MPC-based algorithm that follows the normal tracking formulation and it is shown that the algorithm makes better use of the platform workspace than the classical algorithm. Neither present results of tests with human drivers.

Both of the MPC-based algorithms described above use Cartesian workspace boundaries (which must necessarily be more restrictive than the actual motion envelope due to the coupling of Cartesian limits in the different DOFs) and make use of commercially available MATLAB add-ons to solve the MPC optimization problem. The MPC algorithm developed in this research is, like Augusto [9], formulated as a tracking problem, however the spatial limits are based on actuator states instead of Cartesian platform states, and thus better use of the workspace is possible. Additionally, a solver for the MPC optimization has been derived that requires only simple linear algebra and can be implemented using standard MATLAB functions.

In order to reduce the computational complexity, the MPC control for motion in 6DOF is split into two dual-input dual-output controllers for the lateral-roll and longitudinal-pitch pairs (these are paired to allow for tilt coordination, if required) and two single-input single-output controllers for the vertical and yaw DOFs. For each of these controllers a linear model containing the human motion perception (vestibular system) dynamics and the platform kinematics is created; taking the lateral (y) and roll (ϕ) controller as an example, the model in state-space form is

$$\begin{bmatrix} \boldsymbol{x}_{vest}(k+1) \\ \boldsymbol{x}_{plat}(k+1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_{vest} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}_{plat} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{vest}(k) \\ \boldsymbol{x}_{plat}(k) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_{vest} \\ \boldsymbol{B}_{plat} \end{bmatrix} \begin{bmatrix} \phi_{plat}(k) \\ y_{plat}(k) \end{bmatrix}$$

$$\begin{bmatrix} \hat{\boldsymbol{\phi}}(k) \\ \hat{\boldsymbol{y}}(k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{C}_{vest} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{vest}(k) \\ \boldsymbol{x}_{plat}(k) \end{bmatrix}$$

where $\boldsymbol{x}_{vest}$ and $\boldsymbol{x}_{plat}$ are the vestibular and platform state vectors respectively, ϕ, y_{plat} are the control inputs to the platform, and $\hat{\boldsymbol{\phi}}, \hat{\boldsymbol{y}}$ are the perceived motions. If tilt coordination is required then the effect of the roll input on perceived lateral motion is included in the vestibular dynamics model. Note that the platform dynamics are not modelled; in order to ensure real-time operation of the algorithm the number of states was kept to a minimum, and thus it is assumed that the platform demand is met (this is true up to around 20Hz).

The platform state vector comprises the platform velocities and actuator lengths. In order to calculate the effect of the control inputs on the actuator lengths, expressions for the actuator length derivatives are found; these are of the form

$$\frac{d}{dt}(l_i) = \frac{k_A\dot{x} + k_B\dot{y} + k_C\dot{z} + k_D\dot{\phi} + k_E\dot{\theta} + k_F\dot{\psi}}{l_i}$$

The $k_{A...F}$ are functions of the platform position and orientation and their derivatives and are therefore time-varying, however they are assumed constant over the prediction horizon and recalculated at each time step.

The MPC optimization problem is to find, at each time step, the optimal control sequence over a horizon H_u that minimizes the following cost function over a prediction horizon H_p:

$$J = \sum_{i=0}^{H_p-1} \{\boldsymbol{e}(t+i)^T\boldsymbol{Q}_{perc}\boldsymbol{e}(t+i) + \boldsymbol{x}_{plat}^T(t+i)\boldsymbol{Q}_{plat}\boldsymbol{x}_{plat}(t+i) + \boldsymbol{u}(t+i)^T\boldsymbol{R}\boldsymbol{u}(t+i)\}$$

where $\boldsymbol{e}(t) = \hat{\boldsymbol{y}}_{car}(t) - \hat{\boldsymbol{y}}_{sim}(t)$ is the perception error (difference between perceived motion in vehicle and perceived motion on simulator), $\boldsymbol{x}_{plat}(t)$ are the actuator positions and velocities, $\boldsymbol{u}(t)$ is the control input applied to the platform, and the $\boldsymbol{Q}$ and $\boldsymbol{R}$ matrices contain weightings for the cost terms. Tuning of the cost weightings relative to one another determines the strategy employed by the controller.

The solver for this optimization follows the method of Boyd and Vandenberghe [10]. Firstly, the MPC problem is reformulated as a quadratic programming (QP) problem. This is then solved using a primal barrier infeasible start method. This can find a step which is close to the optimizing step within a small number of iterations – in this case up to five. Testing of the MPC algorithm revealed that good performance was possible with a control horizon of a single time step and a prediction horizon as low as three time steps.

2.4 Body Sideslip Algorithm

The various cueing algorithms currently in use all follow the same basic idea, namely to get the platform accelerations as close to the vehicle accelerations as possible within the available workspace. However, it seems sensible instead to concentrate on the information given to the driver by the motion. As with all the driver's sensory inputs, the motion provides information about the vehicle state; this is particularly true in the non-linear region of vehicle response, where the driver's control strategy will involve more closed-loop behaviour than in the linear region.

When evaluating vehicle stability the body sideslip angle β is of interest; it was therefore proposed that the vehicle sideslip angle be used to directly drive the platform yaw angle. As the velocity and acceleration are also required, the first and second derivatives β must be found – in terms of the body reference longitudinal and lateral velocities u and v, and using the small angle approximation $\beta = v/u$, these are

$$\dot{\beta} = \frac{\dot{v}}{u} - \frac{\dot{u}v}{u^2} \qquad \ddot{\beta} = \frac{\ddot{v}}{u} - \frac{\ddot{u}v}{u^2} - 2\frac{\dot{u}\dot{v}}{u^2} + 2\frac{\dot{u}^2 v}{u^3}$$

The first derivative is straightforward as all the quantities are available in the vehicle dynamics. However, the second derivative contains jerk terms $\ddot{u}$ and $\ddot{v}$ which are not available. This is overcome by considering the derivatives of the expressions for longitudinal and lateral accelerations a_x and a_y; considering a_y, this gives

$$a_y = \frac{\Sigma F_y}{m} = \dot{v} + ur + wp$$

$$\xrightarrow{d/dt} \quad \ddot{v} = \frac{\Sigma \dot{F}_y}{m} - \dot{u}r - u\dot{r} - \dot{w}p - w\dot{p}$$

where the F_y are the lateral tyre forces, m is the vehicle mass, w is the vertical velocity and p and r are the roll and yaw velocities. All the terms in the expression for $\ddot{v}$ are available apart from the tyre force derivatives. However, the tyre model contains an expression for relaxation lag with time constant τ, $\dot{\hat{F}}_{xi,yi} = \tau^{-1}(F_{xi,yi} - \hat{F}_{xi,yi})$, so the tyre force derivatives are in fact available. The derivation of the expression for $\ddot{u}$ follows in the same way from a_x.

To complement the sideslip-based yaw motion, the lateral and longitudinal platform motion is calculated to provide an instantaneous centre (IC) on the platform related to the IC of the vehicle, the idea being that this provides some more information about the vehicle state. The distance of the IC from the vehicle centre of mass would generally result in unachievable longitudinal/lateral platform motion, so in order to avoid saturation and maintain fidelity of the cue, the IC is calculated in polar coordinates and a limiting non-linear gain is applied to the magnitude.

The platform roll angle is driven by the vehicle roll angle, attenuated to avoid the perception of excessive roll (one-to-one restitution is perceived as being excessive; this is perhaps due to the mismatch in magnitude between full restitution and low restitution in different DOFs).

3. EXPERIMENTAL METHOD

The Loughborough simulator, produced by Cruden [11], works with vehicle models in MATLAB/Simulink that run with a fixed integrator time step of 1ms. The vehicle model used here has 6 body DOFs and uses a Pacejka combined-slip tyre model. Motion cueing is also done within the Simulink model, and the operator can switch between cueing algorithms whilst the simulation is running.

In order to focus on the effect of motion related to vehicle lateral stability, the standard classical algorithm provided motion in the longitudinal, pitch and vertical DOFs during all test cases; only the lateral, roll and yaw motion was modified. Note that the longitudinal motion used to place the rotation centre in the body sideslip algorithm was added to the that demanded by the classical algorithm.

Two vehicle parameter sets were used in the tests; the first is based on a mid-sized front wheel drive saloon car, with parameters from the manufacturer and tyre parameters based on vehicle test data. This vehicle was chosen to be quite easy to drive with a tendency towards limit understeer. The second test vehicle is intended to be one which is difficult to control, particularly near the limit of grip with a tendency towards limit oversteer; the vehicle parameters are based on those of a small single-seater rear wheel drive racing car. The tyre parameters are based on those used in the first model with increased road-tyre friction coefficient μ, a shorter relaxation length, and tyre curves modified to give the desired limit behavior.

Three groups of drivers were used as participants – 'expert' drivers experienced in driving consistently near the performance limits of a vehicle and with some knowledge of vehicle dynamics; 'advanced' drivers with some experience of driving a vehicle near the limit; and 'normal' drivers, those who have only driven on the public road and have little or no experience of vehicle behaviour beyond the linear region. The expert group (and to a lesser extent the advanced group) are key to these tests as experience of non-linear vehicle behaviour is necessary in order to evaluate how realistic the various cueing algorithms are in these tests. The group of normal drivers was included to see how they rate the motion when spending an extended period of time in the non-linear regime for the first time, and to compare the results of expert drivers with those from a wider range of people.

The course chosen was a 2.1km race circuit with a mixture of low- and high-speed corners, figure 1. A race circuit was chosen in order to prompt drivers to push the vehicle towards the limits of its capabilities.

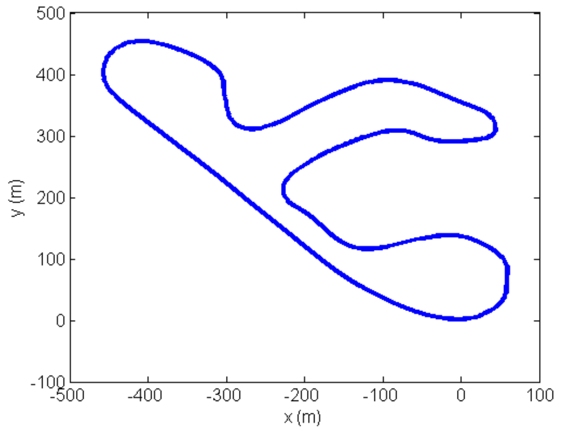

Fig. 1 Circuit layout

The initial intention was for the drivers to spend some time getting used to the vehicle behavior and track layout, before proceeding to do pair-wise comparison tests between the three algorithms. However, pre-tests indicated that the choice of algorithm in the acclimatization laps had a significant effect on the driver's learning process. It was therefore decided to modify the experimental process such that, after 3 laps to become accustomed to the general vehicle behaviour and track layout, a set of pair-wise comparison tests was carried out (i.e. while the learning process was ongoing). After this the subjects were asked to drive 3 further laps to give them more time to learn the vehicle's handling characteristics and then another paired comparison test was carried out, with the intention that this would take place after most learning had been completed. This process was done for both vehicle models; the more easily-controlled saloon car was used first as it was judged that this was easier to learn and therefore gave the driver time to learn the track at the same time.

Each set of pair-wise comparisons comprised seven laps; this is the minimum number of laps required to evaluate all possible algorithm pairings in both directions. The pair order was varied for each of the four pair-wise sets that each participant carried out and also varied between participants, in order to eliminate order effects as far as possible. The participants were not aware of which cueing algorithm they were driving at any point, nor in fact were they told how many cueing algorithms were being compared.

After each lap in the set of seven (except the first), the participants were asked to evaluate how realistic the simulator was on the lap just completed compared to the one before; they were asked for responses on a Likert scale, table 1.

Much worse	Worse	About the same	Better	Much better
-2	-1	0	+1	+2

Table 1 Likert scale for paired comparisons

If the participants had any additional comments these were also recorded.

4. RESULTS

The first analysis of the pairwise results calculates a total score for each algorithm based on the Likert scale values for each paired comparison. First, each participant's results are normalized by the mean absolute value of their non-zero scores; this is done to eliminate the effect of the participants' differing interpretation of the Likert scale levels. For example, if a participant scores mostly ±1 but on a few occasions scores ±2 then, after normalizing, these will have a greater significance than they would for a participant who consistently scores ±2. Once the results have been normalized, the total scores for the three algorithms are calculated; for each paired comparison result, the winning algorithm's score is increased by the normalized rating and the losing algorithm's score is decreased by the same amount. These total scores are calculated for the two vehicles separately; the scores for the complete set of participants are shown in figure 2.

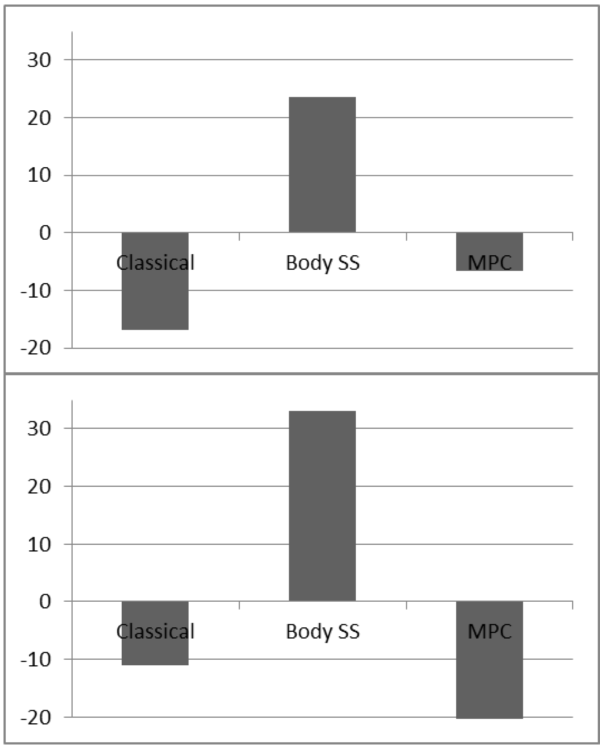

Fig. 2 Total normalized scores (top: saloon, bottom: race car) for all participants

For both vehicles the body sideslip algorithm is the clear 'winner'; for the saloon car the MPC algorithm comes in second and the classical algorithm is worst, although both have a negative overall score. For the race car the MPC algorithm comes out worst, again with both it and the classical algorithm having a negative score.

The results for the five expert drivers are shown in figure 3. As with the combined results the body sideslip algorithm performs well with a positive score and the classical algorithm has a large negative score; the main difference is in the MPC results. Here, the MPC algorithm has a score close to that of the sideslip algorithm for both vehicles and, for the saloon car, there is even a slight preference for the MPC algorithm. In general, the expert drivers commented that the MPC algorithm gave good feedback about the behaviour of the vehicle up to a point near the limit of grip, whereas the body sideslip algorithm gave very good yaw feedback on turn-in and on the transition into limit oversteer. This appeared to be particularly true in the race car; a few of the expert drivers commented that the body sideslip algorithm gave them fast enough feedback to maintain good control of the race car with its higher-frequency dynamics. Some also said they felt like the body sideslip algorithm allowed them to drive in a closed-loop manner, rather than relying on open-loop inputs based on experience from previous laps. Note that all these comments were made during the paired comparisons, i.e. the participants were unaware of how many algorithms there were and which they had just driven. These results suggest that a composite MPC and body sideslip cue (with the MPC cue providing feedback away from the limit and body sideslip providing extra information near/beyond the limit) would perhaps be a good all-round solution and it is intended that this be investigated in future work.

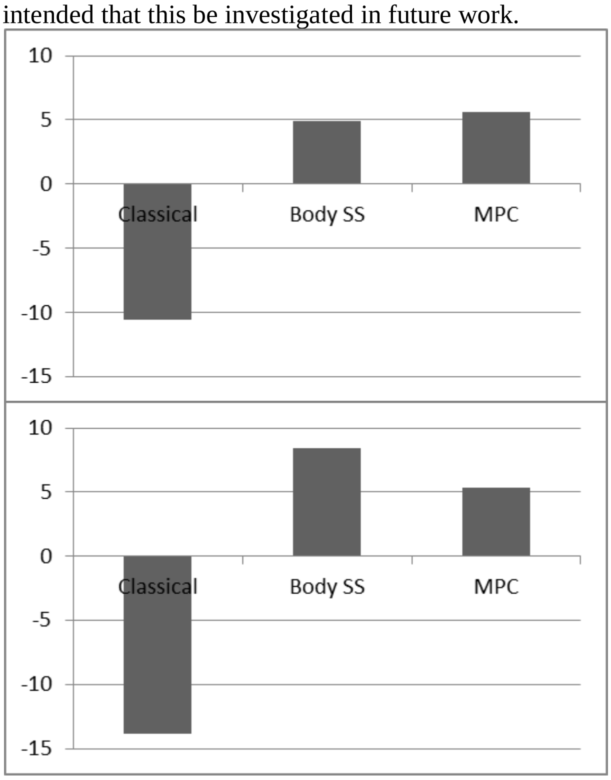

Fig. 3 Total normalized scores (top: saloon, bottom: race car) for expert drivers

An alternative analysis uses a least squares (LS) regression to fit a model to the data. The problem is formulated as $\boldsymbol{U\theta} = \boldsymbol{y}$ where the matrix $\boldsymbol{U}$ describes the permutations of the various tests (for example the row for the first algorithm followed by the second algorithm is [-1 1 0]), the vector $\boldsymbol{y}$ contains the results of the tests, and the vector $\boldsymbol{\theta}$ is calculated as the scores for the three algorithms that fit the data best. The results, again calculated for all participants and separately for the expert group, are shown in figures 4 and 5 (note that in each case the classical algorithm is set as the baseline with a score of zero).

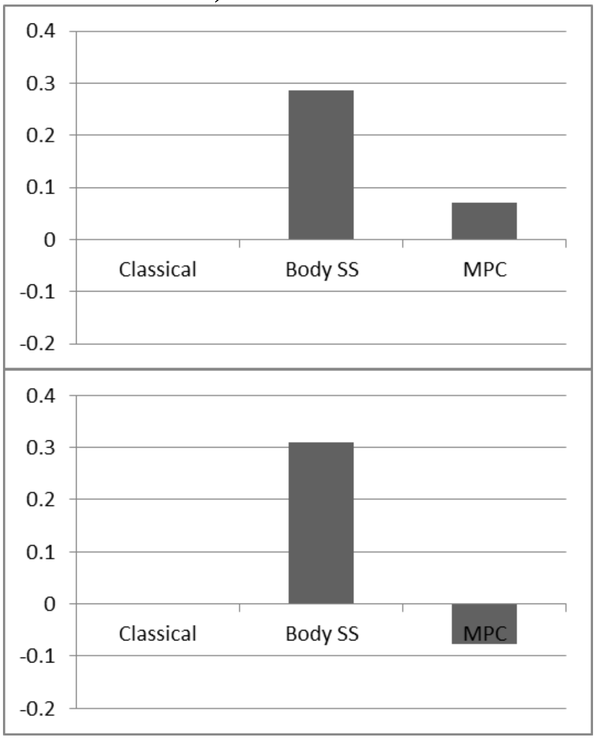

Fig. 4 LS regression results (top:saloon, bottom: race car) for all participants

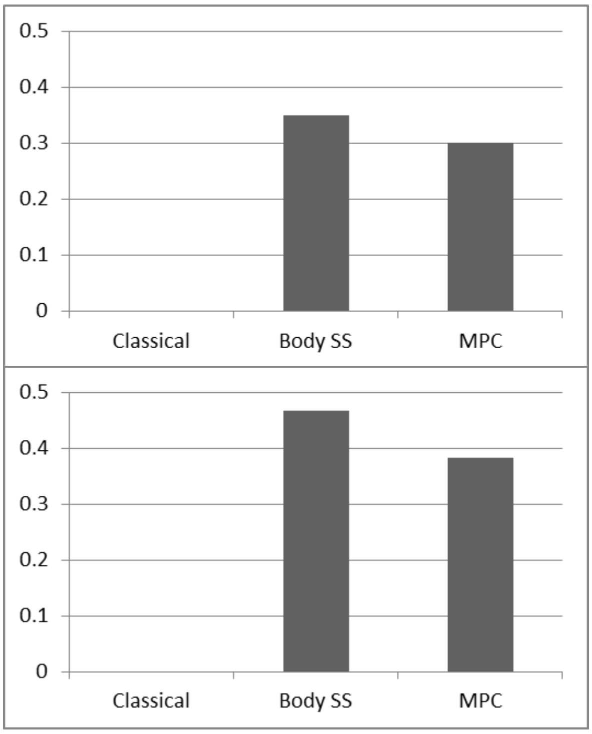

Fig. 5 LS regression results (top:saloon, bottom: race car) for expert drivers

The regression results largely agree with the results of figures 2 and 3, the body sideslip algorithm appearing to be the best algorithm overall with the MPC algorithm being a close second for the expert drivers but performing badly in the combined results.

Applying the Friedman test (see e.g. David [12]) to the results gives an indication of whether the null hypothesis (in this case that the cueing algorithm has no effect on the results) is true; in other words it is a statistical measure of how likely it is that the observations are due to significant differences in the effectiveness of the three algorithms and not just a result of noise in the results. The test was carried out on two permutations of the normalized score data; one with the results grouped into blocks by participant, and the other with the grouping done by run. The p-values calculated for the two cases are, respectively, 0.0008 and 0.0011 and thus the null hypothesis can be rejected with some confidence (a threshold of 0.05 is often quoted as a reasonable cut-off for rejection of the null hypothesis, see e.g. Noether [13]).

5. CONCLUSION

Two new cueing algorithms have been compared with a commercially-supplied algorithm for driving in the non-linear region of vehicle behaviour. The two algorithms represent, in the form of the MPC algorithm, the best possible algorithm that follows the 'standard' motion-tracking approach to cueing and, in the body sideslip algorithm, a new approach to cueing that is designed to give the best information about the vehicle state to the driver.

The results of the bidirectional pair-wise comparisons indicate that the body sideslip algorithm is best, with the MPC algorithm a close second for the expert drivers. Many drivers made comments along the lines that the body sideslip angle provided them with the feedback necessary to pick up the current vehicle state quickly enough to maintain control of the vehicle near its limits.

It is anticipated that a composite algorithm that combines the MPC and body sideslip algorithm would provide good feedback for all driving scenarios; it is intended that such an algorithm be developed and tested in future.

REFERENCES

[1] Newton, A.P. and Best, M.C. "The influence of motion on handling dynamics analysis in full vehicle simulators", Proc. of AVEC '06, 2006, pp 303-308

[2] Siegler, I., Reymond, G., Kemeny, A. & Berthoz, A. "Sensorimotor integration in a driving simulator: contributions of motion cueing in elementary driving tasks." Paper presented at DSC2001, Nice, France, 2001

[3] Garrett, N.J.I and Best, M.C. "Evaluation of a new body-sideslip-based driving simulator motion cueing algorithm", IMechE Part D, in press

[4] Schmidt, S. F., & Conrad, B. "Motion Drive Signals for Piloted Flight Simulators" NASA Report CR-1601, 1970

[5] Nahon, M.A., Reid, L.D., and Kirdeikis, J. "Adaptive Simulator Motion Software with Supervisory Control", J. Guidance, Control and Dynamics, March-April 1992, 15(2), pp. 376-383

[6] Sivan, R., Ish-Shalom, J., & Huang, J. K. "An Optimal Control Approach to the Design of Moving Flight Simulators" IEEE Transactions on Man, Systems and Cybernetics, 1982, 12(6), pp. 818-827

[7] Telban, R.J. and Cardullo, F.M. " Motion Cueing Algorithm Development: Human-Centred Linear and Nonlinear Approaches", NASA report CR-2005-213747, 2005

[8] Dagdelen, M., Reymond, G., Kemeny, A., Bordier, M., Maïzi, N. "Model-based predictive motion cueing strategy for vehicle driving simulators" Control Engineering Practice, 2009, 17(9), 995-1003

[9] Augusto, B. D. C. "Motion Cueing in the Chalmers Driving Simulator: An Optimization-Based Control Approach" MSc Thesis, Chalmers University of Technology, Sweden, 2009

[10] Boyd, S., & Vandenberghe, L. "Convex Optimization", Cambridge University Press, 2004

[11] http://www.cruden.com, accessed June 2012

[12] David, H.A. "The method of paired comparisons", Griffin, 1988

[13] Noether, G.E. "Introduction to Statistics - The Nonparametric Way", Springer-Verlag, 1991