Motion cueing algorithm development for X-Y linear motion and yaw table based driving simulator

Zhou FANG\textsuperscript{1,2}, Masashi TSUSHIMA\textsuperscript{1}, Eiichi KITAHARA\textsuperscript{1}, Naoya MACHIDA\textsuperscript{1}, Andras KEMENY\textsuperscript{2}

(1): Nissan Motor Co., Ltd. CAE & Testing Division 1, NTC 560-2, Okatsukoku, Atsugi-shi, Kanagawa 243-0192, Japan, e-mail: {zhou_fang, masashi-tsushima, e-kitahara, nao-machida}@mail.nissan.co.jp
(2): RENAULT, Virtual Reality and Immersive Simulation Center, TCR AVA 0 13, 1 avenue du Golf, 78288 Guyancourt, France, e-mail: {zhou.fang, andras.kemeny}@renault.com

Abstract - The double lane change and the driving in curves with or without sudden maneuver are some typical scenarios commonly investigated in driving simulator. The high acceleration at large yaw angles is a typical characteristic in such kind of scenarios which makes the reproduction of motion cues more difficult. In fact, the resultant longitudinal and lateral motions in driving simulator can be no longer considered as two independent variables. However, the most of the developed motion cueing algorithms are generally based on the hypothesis of independence of the x-y translation motions in the driving simulator. A literature review shows only nonlinear MPC algorithm could be useful to solve out the x-y coupled motion cues system [Bru14]. This paper describes a method to deal with the x-y coupled motion cueing algorithm which is particularly useful to analyze the motion cueing results from high advanced driving simulator disposed of a yaw table.

Keywords: motion cueing, NMPC based motion cueing algorithm, washout, driving simulation

Introduction

In a driving simulator, considering the seat, cockpit and the hexapod as a rigid body, a vector expressed in the driving simulator (DS) ground coordinate system (inertial frame) can be related to the rotational cockpit reference system (body-fixed frame) by using the Euler’s rotation matrix:

\[
R_{\text{inertial to cockpit}}(\psi, \theta, \phi) =
\begin{bmatrix}
\cos \theta \cos \psi \sin \phi - \sin \phi \sin \theta \\
\sin \psi \sin \phi + \cos \phi \sin \theta \\
\cos \psi \sin \phi + \sin \phi \cos \theta
\end{bmatrix}
\]

which defines a sequence of rotations, yaw $\psi$, pitch $\theta$ and roll $\phi$, carrying the inertial frame to the body-fixed frame [Tel05].

One of the applications of Euler’s rotation matrix is to estimate the resultant longitudinal (x) and lateral (y) acceleration on the body-fixed frame by ignoring the high order of small value terms:

\[
\begin{bmatrix}
a_{\text{longi}} \\
a_{\text{lat}}
\end{bmatrix} \approx
\begin{bmatrix}
cos \theta \cos \psi \cos \phi \sin \psi \sin \phi - \sin \phi \sin \theta \\
- \cos \phi \sin \psi + \sin \psi \cos \phi \sin \phi \cos \psi \sin \phi - \sin \phi \cos \phi \cos \psi \sin \phi + \sin \phi \cos \theta
\end{bmatrix}
\begin{bmatrix}
a_x \\
a_y \\
\theta
\end{bmatrix}
\]

(1)

where the acceleration, $a_{\text{longi}}$ and $a_{\text{lat}}$ are called specific force which approximates the stimulus that the driver would experience in a real car[Tel05]. With the help of other coordinate cues, such as visual and seat feedback force etc. the specific force in DS, composed of linear acceleration and the gravity projection component due to the tilt angle, are used to reproduce the sustainable acceleration.

In conventional linear model approach, the eq-1 is simplified into:

\[
\begin{bmatrix}
a_{\text{longi}} \\
a_{\text{lat}}
\end{bmatrix} \approx
\begin{bmatrix}
a_x + g \cdot \theta \\
a_y - g \cdot \phi
\end{bmatrix}
\]

However, for a scenario with strong translation at large yaw angles, it is shown from eq-1 that the resultant longitudinal and lateral specific forces are strongly coupled. Thus, it is necessary to develop a new motion cueing algorithm for satisfying such kind of DS application. Otherwise, for the DS disposed of only 1DOF translation and yaw motion systems, the translation or the yaw angle signals must be restricted to some relative low values in order to reduce the unexpected motion in other axis. Concerning a 2DOF translation and yaw motion system based DS, it is possible by tuning separately the linear motion cueing algorithm to achieve an optimal motion strategy. Here we develop a nonlinear MPC based motion cueing algorithm aiming to simplify the tuning task and to improve the optimal motion cueing strategy.

MPC based motion cueing algorithm

QP problem formulation

The MPC based motion cueing algorithm is a time-variant reference signal tracking problem. Based on
perfect non-disturbing plant model which is described by:
\[
\begin{align*}
\dot{x} &= Ax + Bu \\
y &= Cx
\end{align*}
\]  
(2)

the corresponding discrete MPC system model can be written with the system constraints and \( \Delta u \) as input:
\[
\begin{align*}
\hat{x}_{k+1} &= A_m \hat{x}_k + B_m \Delta u_k \\
y_k &= C_m \hat{x}_k \\
K_{\text{min}} \leq [H_x H_u] \hat{x}_k \leq K_{\text{max}}
\end{align*}
\]  
(3)

A cost function associated with the system’s output, \( y \), input, \( u \), and the state variables, \( x \), is defined for the purpose of finding the optimal strategy. In motion cueing algorithm, the cost function takes into account the error of tracking signals and the efforts used, such as DS workspace, velocity and acceleration to achieve the tracking task. It can be written by:
\[
J^*_k(X_k) = \min_{\Delta u_{k-1}} \sum_{i=0}^{N} \| \hat{x}_i - q_i \|^2 + \sum_{i=0}^{N} \| \Delta u_i \|^2 + \sum_{i=0}^{N} \| \hat{y}_i \|^2 
\]
(4)

where \( q_i \) is the current reference signal, \( q_i > 0 \), weighting coefficient for tracking error, \( R_m > 0 \), \( Q_m \geq 0 \), \( Q_u > 0 \), weighting matrices for input and state respectively. They are used to balance between the tracking error and the efforts used.

We will express (3) and (4) in a quadratic programming standard form. Towards this end, we define an \( M \) manipulated steps input moving block in order to reduce the unknown variables to be solved, denoted by \( U \), then the \( N \) steps input vector is given by:
\[
\begin{bmatrix}
\Delta u_k \\
\vdots \\
\Delta u_{M+k-1}
\end{bmatrix} = KU \quad \text{with} \quad U = \begin{bmatrix}
I_k \\
\vdots \\
I_k
\end{bmatrix}
\]

By applying the recurrent formulation (3) with index from \( k+1 \) to \( k+N \), the \( N \) steps predictive states are obtained as below by using the current state, \( \hat{x}_k \), and \( M \) steps input vectors:
\[
X = F\hat{x}_k + SU
\]
(5)

where:
\[
\begin{bmatrix}
\hat{x}_{k+1} \\
\hat{x}_{k+2} \\
\vdots \\
\hat{x}_{k+N}
\end{bmatrix} = \begin{bmatrix}
A_m & 0 & 0 \\
A_m & B_m & 0 \\
\vdots & \vdots & \vdots \\
A_m^{N-1} & B_m & \cdots & 0
\end{bmatrix}
\]
\[
S = \begin{bmatrix}
B_m \\
A_mB_m & B_m \\
\vdots & \vdots & \vdots \\
A_m^{N-1}B_m & A_m^{N-2}B_m & \cdots & B_m
\end{bmatrix}
\]
\[
K = \begin{bmatrix}
I_k \\
I_k \\
\vdots \\
I_k
\end{bmatrix}
\]

Considering the cost function (4) and using the following expression for \( N \) steps tracking error vector:
\[
Y = \Phi X - R(k)
\]
(6)

where: \( \Phi = \text{diag}(C_m, C_m, \ldots, C_m, 0) \), \( R(k) = [1, \ldots, 1, 0]^T r(k) \), we can deduce [Fan17]:
\[
J_k(U, X_k) = J_k(\hat{x}_m, R_i(k), \ldots, U^T Q_u U + 2 Q_u U)
\]
(7)

where: \( Q_u \) and \( Q_m \) are constant, function of current state and system matrices, weight coefficients etc. Using equations (3) and (5), we can transform the state constraints into input constraints:
\[
[P \ \ H_u]^T U \leq [b \ U_m] 
\]

with \( P = H_x S, \ b = [K \ \ -H_x F \hat{x}_k] \)
(8)

Equations (7) and (8) consist of standard QP problem formulation.

Nonlinear MPC based motion cueing model

Motion system model

In DS motion system, generally, the electro-mechanical motion system control and the motion cueing control are two separate control systems. The former is the role of the DS motion system provider and the second, the DS exploiters. Hence, in motion cueing algorithm, the motion system can be considered as an ideal DS, i.e., a double integrator described by the system bellow:
\[
\frac{dx_i}{dt} = Ax_i + B_i u_i
\]
(9)

where the index \( i \) denotes, in DS frame, the axis \( x, y, z \) and also the corresponding pitch, roll, yaw rotational axis.

\[
X_i = [x_i \ \dot{x}_i]^T, A_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, B_i = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{with} \quad u_i = a_i
\]
or
\[
X_i = [x_i \ \dot{x}_i \ \ddot{x}_i]^T, A_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, B_i = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \text{with} \quad u_i = \frac{da_i}{dt}
\]

For the motion cueing algorithm, the Real-Time (RT) performance requires the motion cueing system’s dimension as low as possible. If the car’s yaw motion can be reproduced by DS yaw motion based on yaw table with a reasonable scaling factor or the scale 1:1, it could be better to consider it as an independent motion cueing system in order to reduce the global system dimension and also to keep a smooth yaw motion. Based on the ideal DS hypothesis and the separate yaw motion cueing strategy, without integrating the vestibular model, the DS x-y coupled motion system can be described by a 12 state variables using \( du/dt \) as input:
\[
x_{DS} = [p_y \ p_y \ a_y \ \dot{a}_y \ \psi \ \dot{\psi} \ \phi \ \dot{\phi}]^T, \\
U_{DS} = d[a_x \ a_y \ \dot{\theta} \ \dot{\phi}]/dt
\]
and \( x_{yaw} = [\psi \; \dot{\psi}]^T \), \( u_{yaw} = d\dot{\psi}/dt \)

The use of \( du/dt \) as system’s input aims at improving the quality of motion cueing result by introducing jerk constraint on the system control signals.

As these state variables are independent, the global system matrices are then given:

\[
A = \text{diag}(A_x, A_y, A_{\phi_1}, A_{\phi_2}), \quad B = \begin{bmatrix} B_x & 0 & 0 & B_y & 0 & 0 & 0 & B_\phi \end{bmatrix}
\]

The cost function in discrete form is given:

\[
J(U) = \sum_{i=0}^{N-1} \left[ a_{DS_{\text{long}i}} - a_{\text{veh}_{\text{long}i}} \right]^2 + \sum_{i=0}^{N-1} \left[ a_{DS_{\text{lat}i}} - a_{\text{veh}_{\text{lat}i}} \right]^2
+ \sum_{i=0}^{N-1} \left[ \phi_{DS_{\gamma \phi},i} - \phi_{\gamma \phi,i} \right]^2 + \sum_{i=0}^{N-1} \left[ \phi_{DS_{\gamma},i} - \phi_{\gamma,i} \right]^2
+ X^TQ_vX + U^TR_uU
\]

\[ \text{subject to:} \]

\[
|p_1 + c_T \dot{x}_j + \frac{T^2}{2}a_1| < L_{x,\text{lim}}/2
\]

\[
|p_1| \leq x_{\text{max}} \quad \left| \dot{p}_1 \right| \leq \dot{x}_{\text{max}}
\]

\[
\dot{\theta}_1 + c_{T_{\phi}} \omega_1 + \frac{T^2}{2} \dot{\theta}_{\phi,nollim}/2 \leq \theta_{\phi,\text{nollim}}/2
\]

\[
|\phi_1| \leq \phi_{\text{max}} \quad \left| \dot{\phi}_1 \right| \leq \dot{\phi}_{\text{max}}
\]

where the \( a_{DS_{\text{long}}} \) and the \( a_{DS_{\text{lat}}} \) are given by eq.1.

The model built in this manner keeps always the linear dynamic characteristics for the motion system model. The nonlinearity of the motion cueing problem comes from the resultant \( a_{DS_{\text{long}}} \) and \( a_{DS_{\text{lat}}} \) in the cost function. The process to find out the optimal cueing strategy is illustrated in figure 1.

If the stability issue can be guaranteed, this nonlinear algorithm is close to the linear approach. The introduced linear combination constraints in above system is to restrict DS motion trajectory to stay within the workspace bounds and to limit the brake signal threshold when the motion is in opposite direction [Fan15]. Therefore the algorithm stability is guaranteed without application of MPC standard stability condition. Consequently, in each time step, we just need to update the cost function with yaw angle value from the independent yaw motion cueing algorithm.

The first experience for the developed \( x \cdot y \) coupled motion cueing algorithm shows that the CPU consumption is very high due to the necessary high predictive horizon, thereby the large matrices’ mathematical operation time. After some optimization on these matrices operation and some setting on the parameter \( T \) to reduce the predictive horizon, the CPU time is greatly reduced and the RT condition is possible.

**Vestibular model**

The vestibular system, situated in the inner ear, consists of two important parts. One contains the semi-circular canals that sense rotational motion and the other, the otoliths that sense linear motion. Both can be theoretically modelled by a spring-damping second order system [Van76, Rey00]. A general physical model [Nah86] is illustrated in figure 2 with a dead-detection area, which is generally ignored in MPC based motion model:

\[
\frac{f}{g} \Rightarrow \frac{k_{fy}}{(\tau_f s + 1)} \Rightarrow \frac{d}{d_s} \Rightarrow s + \tau_{s}^{-1} \Rightarrow \dot{\hat{f}}
\]

where \( \frac{f}{g} \) corresponds to the \( a_{\text{longi}} \) and \( a_{\text{lat}} \) accelerations in DS or in the car.

For describing the vestibular system, One of the typical transfer functions proposed by Young and Meiry(1968)[Tel05, Rey00] describing the relationship between inertial force, \( f \), and perceived inertial force, \( \hat{f} \), is:

\[
G_{m}(s) = \frac{\hat{f}}{f} = \frac{k(\tau_f s + 1)}{(\tau_f s + 1)(\tau_f s + 1)}
\]

or in the continuous-state equation form [Che10]:

\[
\dot{x}_{oto} = A_{oto}x_{oto} + B_{oto}u_{oto}
\]

\[
x_{oto} = [f_{tx} \; f_{tx} \; f_{tx} \; f_{tx}]^T, \quad u_{oto} = [a_x \; \dot{\phi}]^T
\]

\[
A_{oto} = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
-k_{f_y} & 0 & k_{f_y} & 0 \\
0 & 0 & 0 & k_{\phi_{\text{f}}}
\end{bmatrix}
\]

where \( f_{tx} = f \), is the perceived specific force.
The semi-circular canal (SCC) sensation model is given by Young & Oman [Tel05, Rey00]:

$$\dot{\omega} = \frac{k_{scc} T_s T_a \omega^2}{(T_s + 1)(T_a + 1).\omega}$$

which can be also written by:

$$\dot{\omega} = \frac{T_s \omega^2}{s^2 + T_s^2 + T_s + 1).\omega}$$

where:

$$T_s = \frac{k_{scc} T_L T_a}{T_L + T_a + 1), T_L = \frac{T_L T_s + T_L T_a + T_s T_a}{T_L + T_a + T_s}, T_0 = \frac{1}{T_L + T_a + T_s}$$

or:

$$x_{scc} = A_{scc} x_{scc} + B_{scc}.\omega$$

where:

$$x_{scc} = [x_{scc1} x_{scc2} x_{scc3}]^T$$

where $x_{scc1} = \dot{\omega}$ is the perceived rotational velocity.

The vestibular model was first time used in the flight simulator’s motion cueing algorithm by Sivan et al. [Siv82] and then others [Nah86, Tel00]. Generally, the otoliths’ model which plays the essential role in the tilt-coordination technique has a low-pass filter behaviour. By using such kind of model, the motion strategy will optimize only the perceived frequencies motion signal. It relaxes the high frequency components’ signal tracking condition which could be helpful for optimizing the linear motion system. However, the reported transfer functions reveals the very different bandwidths for the vestibular model (cf. table 1). Notice that in the classical optimal filter design [Siv82, Nah86, Tel05, Che10], only the tilt angle or the tilt velocity is used as the system’s input. In consequence, the tilt acceleration is not subject to any constraint. Fortunately, in MPC based MCA model, the linear and the angular accelerations, e.g., $u_s = [a_s \ \ \dot{\omega}]^T$, are used as system input. Our experience shows using jerk variable (du/dt) as input, the command signal quality can be improved again.

Regarding the three tilt velocity thresholds, it seems likely that they could be much higher in the driving simulator than in the flight simulator [Nes12, Tel05].

### Table 1. Human vestibular model parameters [Rey00, Tel05]

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>τc</td>
<td>5.3</td>
<td>7.5</td>
<td>7.5</td>
<td>5.0</td>
</tr>
<tr>
<td>τr</td>
<td>0.66</td>
<td>0.51</td>
<td>0.0</td>
<td>0.016</td>
</tr>
<tr>
<td>τs</td>
<td>13.2</td>
<td>10.1</td>
<td>20.0</td>
<td>10</td>
</tr>
<tr>
<td>k</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Threshold, dnu (m/s²):

Sway = 0.17 Surge = 0.17 Heavy = 0.28

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TL</td>
<td>18</td>
<td>6.1 (Roll)</td>
<td>5.3 (Pitch)</td>
<td>5.73</td>
</tr>
<tr>
<td>Tc</td>
<td>0</td>
<td>0.1</td>
<td>0.1</td>
<td>0.005</td>
</tr>
<tr>
<td>Tc</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>80</td>
</tr>
<tr>
<td>Tc</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.06</td>
</tr>
<tr>
<td>k_scc</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>28.65</td>
</tr>
<tr>
<td>Threshold (°/s)</td>
<td>Pitch=2.0</td>
<td>Roll = 2.0</td>
<td>Yaw = 1.6</td>
<td></td>
</tr>
</tbody>
</table>

The integration of vestibular model in the motion cueing algorithm has some important impacts on the CPU computational time, essentially for the model’s update and for its transformation to the standard OP model. In fact, the CPU computation time increases quickly as the model dimension increases. We present in table 2, the computational time in 1DOF and 2DOF motion cueing algorithm in function of model’s dimension.

### Table 2. CPU time in function of model’s dimension

<table>
<thead>
<tr>
<th>Model dimension</th>
<th>QP solver</th>
<th>Model update and transformation</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1.10</td>
<td>3.05</td>
<td>4.15</td>
</tr>
<tr>
<td>6</td>
<td>4.11</td>
<td>11.85</td>
<td>15.96</td>
</tr>
<tr>
<td>9</td>
<td>5.30</td>
<td>22.48</td>
<td>27.76</td>
</tr>
<tr>
<td>11</td>
<td>10.14</td>
<td>31.95</td>
<td>42.09</td>
</tr>
</tbody>
</table>

We can observe, from the table 2, that when the vestibular model is used, the computational time increases quickly. If we recheck the otolith model (eq-12). We can find that the third state variable is related to the tilt angle. A slight modification of the system is presented as bellow which allows to use directly the tilt angle as the last state variable.

Reformulated otolith’s model:

$$\dot{x}_{oto} = A_{oto} x_{oto} + B_{oto} u_{oto}$$

- **Paris, 7 – 9 Sep 2016**

**Motion cueing algorithm development for X-Y linear motion and yaw table based driving simulator**
As the tilt angle is already presented in DS system’s state. Thus we can move the tilt angle in the otolith model.

Therefore, the following 9 state variables are sufficient to describe the x or y decoupled linear approach system:

\[
x = \begin{bmatrix} f_{x1} & f_{x2} & x_{act} & x_{mov2} & x_{mov3} & p_y & v_y & \theta & \omega \end{bmatrix}^T
\]

Generally, the global system is formed by:

\[
x = \begin{bmatrix} x_{oto} & x_{sc} & x_{DSC} \end{bmatrix}^T
\]

with \( A = \text{diag}(A_{oto}, A_{sc}, A_{DSC}) \)

As the tilt angle in eq-12 is moved, a modification of the global matrix \( A \) is necessary. The corresponding system is a 9 dimensions model using \( u \) as input and 11 using \( du/dt \) as input.

Notice that for the linear acceleration cues using tilt coordination technique, the DS tilt rate and the vehicle’s rotational motion cue are often in conflict. The minimization of the difference between the simulated car’s tilt rate and the DS one by means of the cost function is basically sufficient to reduce the false cues in rotational sensation. With this hypothesis, the 2DOF linear – tilt model can be simplified into a reduced 6 state variables system:

\[
x = \begin{bmatrix} f_{x1} & f_{x2} & p_y & v_y & \theta & \omega \end{bmatrix}^T
\]

MPC predicted horizon and the reference signal

In the MPC based MCA, the reference signal (predicted signal) and the associated horizon are particularly important for the motion cueing algorithm performances. On one hand, to have a real optimal strategy, the best case is to use a reference signal as close as possible to the real test one, on the other hand, the algorithm needs to have some minimal horizon to anticipate the future action by using the tilt-coordination technique. For example, to compensate the tilt motion which brings the cockpit form 5.8° (corresponds about to a 1m/s^2 linear motion) to zero, it needs about 1sec. due to the tilt velocity limitation. The rail linear motion system needs more additional times to place the cockpit into an optimal position. By using the look ahead technique [Bru15], i.e., from \( N_c \) to \( N \), the reference signal was supposed as the same to the driver’s future manoeuvre. An interesting pre-tilt result can be obtained with \( N_c = 10 \) and \( N = 300 \) (\( Ts=8\text{ms} \), see fig. 3). Unfortunately, if the reference signal is delayed for 1s. The resultant motion cues present also some delay for the same algorithm (see figure 4). The figure 5 illustrates the general case where a constant predictive signal is applied.

![Fig.3: Motion cueing result obtained with look-ahead technique and using vestibular model (dimension = 9)](image)

![Fig.4: Motion cueing result obtained with look-ahead signal delayed by 1 s and using vestibular model (dimension = 9)](image)
Motion cueing algorithm development for X-Y linear motion and yaw table based driving simulator

From Fig.3 to Fig.5, it is shown that the constant reference signal can provide a relative efficient solution and the predictive horizon must be high (Nc = 20, N = 250 ~ 500) in order to have a good quality motion cues. However, in x-y coupled system, the model update procedure changes simultaneously Q_u1 and Q_u2 in (eq-7). It is very different to the linear 2DOF model where only Q_u1 needs to be updated at each step time. The more important model update time limits greatly the predictive horizon for the current developed motion cueing algorithm in RT system. In this case, we need to pay more attention for the parameter T presented in the system constraints (eq.-10) in order to reduce the false cues.

Results

In the following motion cueing result, we suppose the DS has enough capacity to reproduce the scenarios' yaw motion by using yaw table [Yan14]. So the reference yaw angle is directly used as DS yaw angle. Figure 6 illustrates a double lane change motion rendering result for the lateral acceleration. In such fast change in acceleration signal, the tilt coordination technique brings rather inadequate performance due to the existent phase delay, so a 1DOF or 2DOF (rail-hexapod) linear motion gives generally a better motion rendering result than that with tilt technique. Moreover, we can use the whole hexapod tilt capacity to reproduce the simulated car's naturel rotational motions. From figure 6, we can observe that the influence of a_y to a_long_DS is particularly important due to its high motion in y axis. Notice that even for a normal DS with the maximal acceleration amplitude limited to 3m/s^2 and a scale 1:1 yaw motion, the induced maximal longitudinal motion is around 1m/s^2 which is non-negligible. In this case, it could be better to apply some scaling factor to reduce the unexpected longitudinal motion. For this scenario, the developed motion cueing algorithm can be tuned as pure x-y coupled linear motion without tilt angle. Compared with a real car's measurement, the motion rendering results demonstrate that the DS can cover the lateral motion with scaling factor 1:1 and regarding the longitudinal acceleration, using some scaling factor, it could be better to reproduce the longitudinal motion stimuli.

Fig. 5: Motion cueing result obtained with N constant signal and using vestibular model (dimension = 9)

Fig. 6: Motion rendering result for lateral acceleration in double lane change scenario (low influence of a_x to a_long_DS)

Fig. 7: Motion rendering result for longitudinal acceleration in double lane change scenario (high influence of a_y to a_long_DS)
With constant oping and testing the model, we have reformulated the otoliths model which allows to reduce 1 dimension in each axis. We have validated the proposed vestibular model for a 2DOF x or y independent motion cueing strategy. It can be extended in future to the x-y coupled motion cueing algorithm.

References

Bruschetta M., Maran F., Beghi A., Minen D. An MPC approach to the design of motion cueing algorithms for a high performance 9 DOFs driving simulator, DSC Europe 2014 proceedings, Paris, France

Fabio Maran, Mattia Bruschetta, Alessandro Beghi, Diego Minen, Improvement of an MPC-based Motion Cueing Algorithm with Time-Varying Prediction and Driver Behaviour Estimation, DSC Europe 2015 proceedings, Tübingen, Germany

Chen S.H., and Li-Chen Fu, An Optimal Washout Filter Design for a Motion Platform with Senseless and Angular Scaling Maneuvers, American Control Conference (ACC), 2010: Baltimore, Maryland, USA, 30 June - 2 July 2010

Fang Z., A. Kemeny, An efficient MPC based motion cueing delay compensation algorithm for driving simulator, DSC Europe 2015 proceedings, Tübingen, Germany

Fang Z., A. Kemeny, An efficient MPC based motion cueing algorithm to driving simulator, AIME special issue “Modeling, Scheduling, and Control in Advanced Production Systems”, April, 2017


Nesti Alessandro, Carlo Masone, Michael Barnett-Cowan, Paolo Robuffo Giordano, Heinrich H. Bülthoff, Paolo Pretto, Roll rate thresholds and perceived realism in driving simulation, DSC Europe 2012 proceedings, Paris, France


Telban Robert J. and Frank M. Cardullo, Motion cueing algorithm development: Human-centered linear and nonlinear approaches, NASA /CR-2005-213747, 2005


Conclusion

In the present work, it is shown that for the scenario with fast change in acceleration at large yaw angles, the x-y motion is strongly coupled for the type of x-y rail and yaw motion system based DS. Generally, for such kind of scenarios, a nonlinear motion cueing algorithm is necessary. However, based on our investigation with a special stability condition, the motion cueing algorithm could be a quasi-linear one. Thus, the RT performance is ensured. The achieved motion cueing results without integrating vestibular model demonstrate the efficiency of the algorithm. Furthermore, in order to integrate the vestibular model, we have reformulated the otoliths model which allows to reduce 1 dimension in each axis. We have validated the proposed vestibular model for a 2DOF x or y independent motion cueing strategy. It can be extended in future to the x-y coupled motion cueing algorithm.

For the scenario of driving in curves with constant speed, finally, the coupled effect of x-y motion is small. For such kind of scenario, the separate x-y 2DOF motion algorithm is sufficient. However, in other car’s validation scenarios, e.g. the ESC tuning task, one of the scenarios is to make a sudden brake during the driving in curves, in this case, the x-y motion in DS is strongly coupled. The current investigated motion cueing algorithm will be tested in future when the scenario is developed.