Motion cueing algorithms for small driving simulator — **Source link** ↗

Lamri Nehaoua, Hichem Arioui, S. Espie, Hakim Mohellebi

Related papers:

- Motion Washout Filter Tuning: Rules and Requirements

- Simulator motion-drive algorithms - A designer's perspective

- Coordinated Adaptive Washout for Motion Simulators

- An Optimal Control Approach to the Design of Moving Flight Simulators

- Motion drive signals for piloted flight simulators

Motion Cueing Algorithms for Small Driving Simulator

L. Nehaoua
INRETS-LSC
Email: nehaoua@inrets.fr

H. Arioui
LSC
40, Rue Pelvux 91020
Evry, FRANCE
Email: Arioui@iup.univ-evry.fr

S. Espie and H. Mohellebi
INRETS
2 Av General Marellet-Joinville 91114
Arceuil, FRANCE
Email: espie,mohellebi@inrets.fr

Abstract— This paper deals with motion control problem for a 2 DOF small driving simulator. The main idea is to test and compare performances of different Washout Algorithms applied to such platform category. The experimentations allow us to have the best compromise between quality of the perception (sensation), implementation complexity and platform architecture.

Implementation of different Washout Algorithms (optimal, adaptive and classical one) will be discussed. Only the longitudinal restitution will be studied. The results show that there is not significant differences between these approaches using with platform type. The lack of pitch DOF in our simulator does not allow a restitution of the sustained acceleration and no coordination between longitudinal and pitch channels may be done.

I. Introduction

In fixed-base driving simulators, the driver manipulates a set of driving controls such accelerating, braking, steering to receive visual cues corresponding to the actual driving situation. Although for some types of driving, it is desirable to provide a motion and haptic restitution to improve the simulation fidelity. Therefore, the driving simulators use a moving platform to restitute in a limited and constrained workspace a sufficient sensation of movement as closely as the one sensed in a real vehicle [1].

Vehicle acceleration can not be reproduced totally, a compromise must be realized between the restitution of inertial indices and the maintain of the platform in its workspace limits. Thus, many command strategies were developed first for flight simulators area [2], [3], [4], [5], [6], [7]. Since that problematic is similar to the driving simulator, the application of these algorithms is direct [8]. However, some characteristics of driving must be taken into consideration. The dynamics of a vehicle are indeed different from those of an airplane, and the 6 DOF acceleration variations in a vehicle are more frequent and sometimes more brutal than those observed in airplane (in particular in bends, when changing lanes or braking). Driving a vehicle takes place in traffic that can sometimes create very complex situations. The driver is thus more solicited for the control of his vehicle than is an airplane pilot. The sensory informations used for driving a vehicle are greater and sometimes different that the ones used for flying an airplane.

This algorithms are based on two main principles:

- The first so called 'Washout', the platform is linearly moved in the same acceleration direction to reproduce the transient longitudinal and lateral accelerations and return back to its neutral position. This last movement should not be detected by the driver (with respect to the sensation threshold) to be not interpreted as a false cue [9],

- The second so called 'tilt-coordination', allows the restitution sustained longitudinal and lateral accelerations by tilting the cabin forward or backward to gain a component of gravity vector. Such tilt can be interpreted by his/her vestibular system as either a positive or negative acceleration, depending on the direction of the tilt. The rate of tilting must be done under the detectable threshold of semicircular channels [9].

The design of washout filters is quite complicated. It depends on simulator architecture and the type of maneuver we are looking to reproduce. Many algorithms were proposed for motion generation in flight simulators, beginning with variations on the classical algorithms [2], [3], [4], [7], followed by variations on the adaptive algorithms [10], [11], [12] and the optimal control approach [13], [14], [15]. Recently, a method based on predictive theory was developed to generate motion cues for Renault Ultimate driving simulator [16]. All this algorithms suppose that the dynamic model of platform can reproduce exactly the reference signals from the output of washout filters. Otherwise, incertitude of modelisation must be taken in account to design a robust controller.

For the current study, a low-cost motion platform equipped with two degrees of freedom have been designed and built. The choice of this architecture is based on the simplicity of design, type of phsycophysic studies to be carried out and the global cost. It is possible to move the simulator's cab with a longitudinal movement, with a small rotation movement of the driver seat.

II. Platform Modeling

To model the driving simulator motion, the overall system is considered as two independent systems mechanically linked: the rotating driving seat and the longitudinal motion platform (cabin). Each of them is driven by a single actuator and a screw/nut device. The motion platform undergoes translational

motions according to one direction (front and back) which correspond to driver's acceleration and deceleration. The overall system's design allows having a simple linear model of the motion.

The choice of the types of motors and screw/nut tread device is taken according to many parameters, that is, accelerations to reproduce, delivered torque, nominal rotational rates and thermic dissipation.

A. The linear motion platform

The motion base supports the cabin which consists of the seat, the vehicle chassis and the driver. Because the rotations of the seat are slow and low amplitude, its induced inertia is negligible comparing to the total mass of the cabin's set. The linear motion of the cabin's set is made thanks to a ball screw/nut transmission mechanism driven by a DC actuator. The technological design was made in order to reduce, mechanical flaws, static and dynamic friction, and to facilitate the design of simple controllers. The overall modelisation was detailed in previous papers [17], [18], we remember here only the dynamic model of the cabin's position $X(s)$ and the voltage command signal $U(s)$:

$$\frac{X}{U} = \frac{1}{s} \frac{K_{t1}}{(J_1 s + f_1)(L_1 s + R_1) + \frac{2\pi N_1}{p_1} K_{e1} K_{t1}} \quad (1)$$

where: s is the Laplacien operator. K_{t1}, K_{e1} are electrical constant of platform DC motor. J_1, f_1 global rotational inertia and friction of platform. R_1, L_1 are platform motor armature resistance and inductance. N_1, p_1 are reduction factor and screw thread of screw/nut device.

B. The rotating seat model

As previously stated, the driver seat can perform two kinds of small rotational motions, the rotation of only the seat's back or the entire seat rotation. A single actuator with a manual switch performs either the first or the second functionality but not both at time. This motion can be coupled to the linear one giving five possible combinations for experimental investigations of motion cue strategy. Using a modeling approach similar to that of cabin supporting platform, we obtain dynamic equation model, of the seat, as follows:

$$
\begin{aligned}
K_{t2} i &= \left(\frac{2\pi l N_2}{p_2} f_{a2} + \frac{2\pi l}{p_2 N_2} f_{s2} \right) \dot{\theta} \quad (2) \\
&- \frac{p_2}{2\pi \eta l N_2} m_t g \rho \sin(\phi + \theta) + \frac{p_2}{2\pi \eta l N_2} m_t K(\theta, \ddot{x}) \\
&+ \left(\frac{2\pi l N_2}{p_2} J_{a2} + \frac{2\pi l}{p_2 N_2} J_{s2} + \frac{p_2}{2\pi \eta l N_2} (J_{t2} + m_t \rho^2) \right) \ddot{\theta}
\end{aligned}
$$

where: θ is seat rotational angle. i is armature current. K_{t2} electrical constant of the seat motor. N_2, p_2 are reduction factor and screw thread of screw/nut device. f_{a2}, f_{s2} are seat motor and screw/nut friction. J_{a2}, J_{s2} and J_{t2} are seat motor, screw/nut and seat/driver rotational inertia. m_t is the estimated seat/driver mass. g is the gravitational vector. ρ is

estimated distance between gravity center and rotation axis. $K(\theta, \ddot{x}) = \ddot{x}\rho \cos(\theta + \phi)$ is the nonlinear term.

Some considerations concerning the rotation angle of the seat are taken in account. We want to generate a platform motion which give the more close sensation as in a real vehicle, without exceeding the small available physical workspace, we will be using the washout and tilt techniques cited above. However, restitution of sustained acceleration requires tilting the seat in a way such that the longitudinal component of vector gravity will be sensed by the operator's otoliths. Nevertheless the tilting angle and rate must be maintained under a certain threshold, otherwise the operator is aware of the seat tilting and an inertial conflict is generated. Consequently, the tilting angle must be kept small ($\prec 4°$), one can make the well known approximations: $\sin\theta \approx \theta$ and $\cos\theta \approx 1$, then:

$$K(\theta, \ddot{x}) = \rho \ddot{x} \cos(\phi) \theta \quad (3)$$

If there is no motion then $\ddot{x} = 0$, the overall equation (2) is linearized in the neighborhood of $-4° \prec \theta \prec 4°$. Otherwise the equation is still nonlinear because F_{x2} varies according to time and the nonlinear term $\ddot{x}.\theta$ can be linearized by dynamic state feedback approach (other appropriate approaches can be developed).

III. WASHOUT FILTER

As we state previously, the platform has 2 DOF the longitudinal movement and the seat rotations. These seat rotations are made to improve the movement perception, but using it for tilting instead the all platform tilting is objectively not proved. For this reason, we only discuss the longitudinal case.

A. Classical Algorithm

This algorithm use a linear high-pass filters to reproduce the transient accelerations of platform. The acceleration of the simulated vehicle is passed though this filter to remove the sustained components which take the platform over its physical limits. The resulting signal is integrated twice to produce the position reference for the actuators as shown in figure 1.
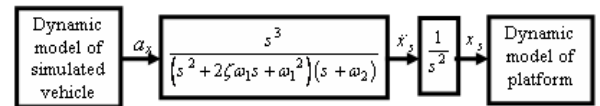

Fig. 1.   Classical Washout Algorithm.

The choice of this filters (order and parameters) depend on the architecture of the driving simulator and the types of maneuvers executed by the driver. Generally, a second order filter can bound the resulted displacement but a three order one is required to realize washout (see figure 2 and figure 3 : simulation results).

Consequently, we consider a washout as:

$$\frac{\ddot{x}_s(s)}{a_x(s)} = \frac{s^3}{(s^2 + 2\zeta.\omega_1 s + \omega_1^2)(s + \omega_2)} \quad (4)$$

where: $\ddot{x}_s$ and a_x are platform and virtual vehicle accelerations. ζ is damping coefficient. ω_1 and ω_2 are break frequencies.
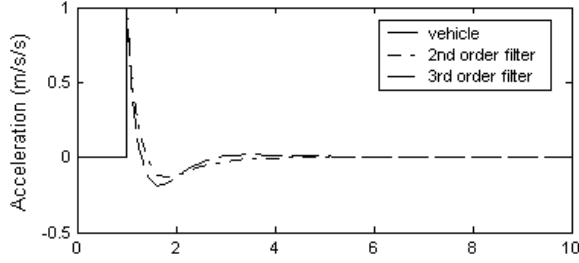

Fig. 2.   Simulated acceleration signal using classical washout filters
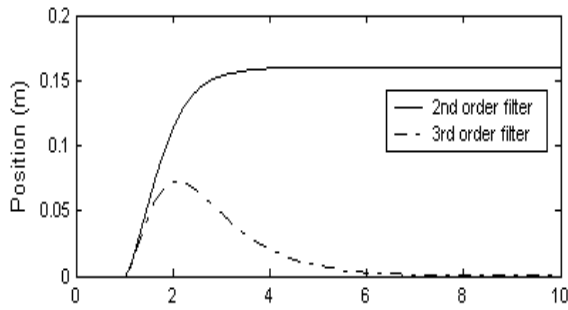

Fig. 3.   Simulated position signal using classical washout filters for longitudinal displacement

The selection of the filter parameters is a trade-off between the restitution fidelity and the physical limits of platform. The filter is configured for the worse case, supposing that the acceleration of simulated vehicle is a step signal of amplitude A_{max}. The pulsation ω_1 determine the acceleration frequency components to be rejected. While the pulsation ω_2 control the rapidity that the platform return back at its neutral position. This process is realized by a trial-error experimentation, in which a set of parameters is fixed, and to obtained results a correction is made up or by resolving an optimisation problem. Nevertheless, the optimal parameters obtained by this last method are not necessarily optimal for other maneuvers.

Classical algorithm is a quite simple one which provide sufficient results for some accelerations maneuvers. Nevertheless, since parameters are configured from the worst case, the exploited workspace is very small comparing to the available one. Other disadvantage is the linear characteristic of the high-pass filters which produce a false cue that can alter the driver perception.

B. Adaptive Algorithm

Proposed by Parrish and *al* [10] to provide motion cues for the Langley flight simulator. This algorithm can be seen as a classical one where parameters are variable and calculated at each step of simulation time. Various schemes were proposed to improve the stability of algorithm [11]. Ariel and Sivan

[12] include the vestibular system for the lateral false cues reduction.

It is based on the minimization of a cost function containing the acceleration error and constraints on the platform displacement. The adaptation is carried out using the steepest descent method to resolve the sensitivity equations [19]. The resulting filter is then nonlinear.
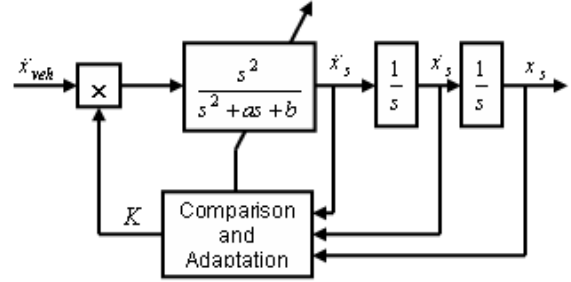

Fig. 4.   Adaptive Washout Algorithm for longitudinal displacement

The filter equation is given by:

$$\ddot{x}_s = K\ddot{x}_{veh} - a\dot{x}_s - bx_s \qquad (5)$$

where: $\ddot{x}_{veh}$ is the virtual vehicle acceleration, $\ddot{x}_s$, $\dot{x}_s$, x_s is the platform acceleration, velocity and position respectively. K, a and b are adaptive parameters of the Washout filter.

The cost function to be minimized is:

$$J = \frac{1}{2}\left[w_a\left(\ddot{x}_{veh} - \ddot{x}_s\right)^2 + w_v\dot{x}_s^2 + w_px_s^2 + w_{p_i}(P_i - P_{i0})\right] \qquad (6)$$

where: wi are weighting coefficients, $P_i \ with \ i = 1, 2, 3$ are to be the adaptive parameters K, a, b, and P_{i0}, $i = 1, 2, 3$ are its initial values.

Optimization is processed by the steepest descent method, that:

$$\dot{P}_i = -\gamma_i \cdot \frac{\partial J}{\partial P_i} \qquad (7)$$

Once the weighting of the function cost w_i and initial conditions P_{i0} are determined, the resolution of sensitivity equation permits to provide acceleration and position signals to drive the platform.

One problem of this algorithm, is the stability of the gradient descent method. This is depends strongly on the adaptation parameter γ_i, which defines the convergence speed of algorithm.

Figure 5 shows a simulation of longitudinal adaptive algorithm and in figure 6 we assume that a tilt coordination exist.

C. Optimal Algorithm

First proposed by Sivan and *al* [13], and developed by others [14], [15]. This algorithm uses higher order filters with optimal control methods. This method incorporate a mathematical model of the human vestibular system, constraining the sensation error between the simulated vehicle and motion platform dynamics.
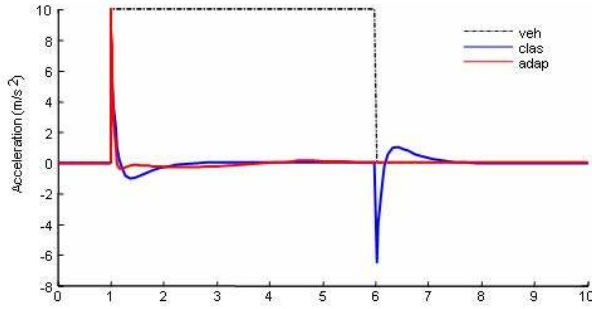
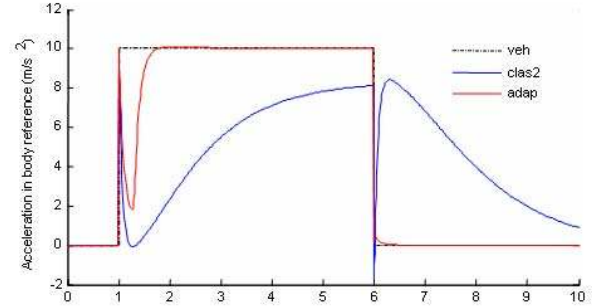Fig. 5. Classical and Adaptive algorithm comparison for 1 DOF

Fig. 6. Classical and Adaptive algorithm comparison for 2 DOF

The goal is to calculate a transfer function $W(s)$ linking the vehicle and platform motion dynamics such:
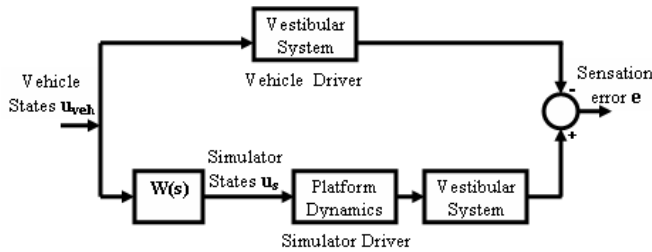
$$U_s(s) = W(s).U_{veh}(s) \qquad (8)$$

Fig. 7. Optimal Washout Algorithm scheme

The optimal command strategy, determine the acceleration u_s by minimizing the following cost function:

$$J(u_s) = E\left\{\int_0^\infty \left(e^T Q e + x_d^T R_d x_d + u_s^T R u_s\right) dt\right\} \qquad (9)$$

While e is the error sensation between driver in the simulator platform and one on the real vehicle. x_d position and velocity states, u_s platform longitudinal acceleration. Q, R_d and R are weighting matrices positifs definite, they define the compromise between the sensation error minimization and the respect of physical limits of the platform.

Fig. 8. The experimented platform

IV. IMPLEMENTATION

In order to compare the performances of previous described algorithms, experimentations are carried out on the present driving simulator (figure 8). Virtual scenes are projected by a two Barco projectors on a fixed wide screen. Traffic simulation, sound rendering and scenarios administration are computed by ARCHISIM Software [20].

First, a scenario consisting in a set of accelerations, decelerations and braking is accomplished. The resulting acceleration from the virtual vehicle dynamic model is saved to be used later for the classic, adaptive and optimal algorithms. This is done to compare the different algorithms for the same maneuver. The parameters of each algorithm are adjusted to respect the physical constraints of the platform ($\pm 0.6\,[m]$, $\pm 1.3g\,[m/s^2]$). For the Optimal method, we use an otolith model of second order proposed by Young and Meiry [21].

The longitudinal acceleration and position of the platform issued from each algorithm is saved and plotted using MATLAB/SIMULINK Software to be analyzed. For a reason of figures clarity, the virtual vehicle acceleration is multiplied by a factor of 0.2 for plotting.

In figures (9, 10) show accelerations (virtual vehicle is the dark one) of classical, adaptive and optimal algorithms (see legend). Due to the limited workspace of the platform, the restituted accelerations is so small regarding the virtual vehicle acceleration. The classical algorithm provide more transient acceleration restitution comparing with the two remaining algorithms, but it shows many false cues due to the linear characteristic of the high pass filter. In fact, when a braking maneuver is executed, visually, the vehicle is stopped. Therefore, the classical algorithm provide a forward displacement to the platform (see figures 11 and 12) corresponding to a generation of an inertial conflict. The adaptive algorithm (figure 9) reduce false cues in this situation, which make an important perception advantage comparing to classic one. Nevertheless, returning back the platform is more slow then the
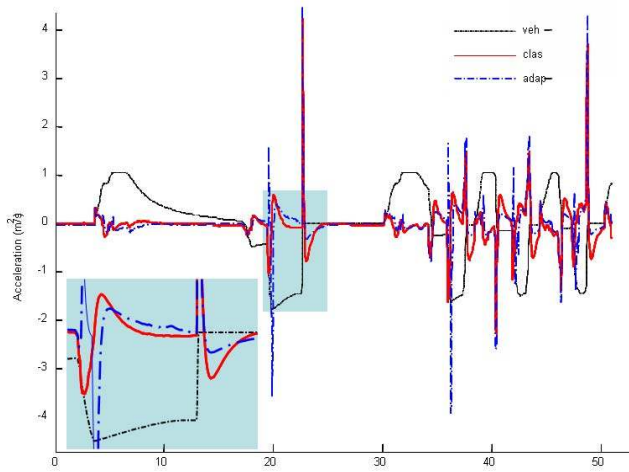
Fig. 9. Cabin's accelerations using adaptive and classical algorithms

Fig. 11. Cabin's positions using adaptive and classical algorithms
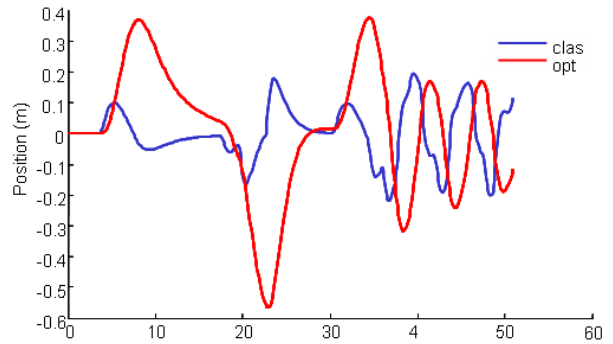
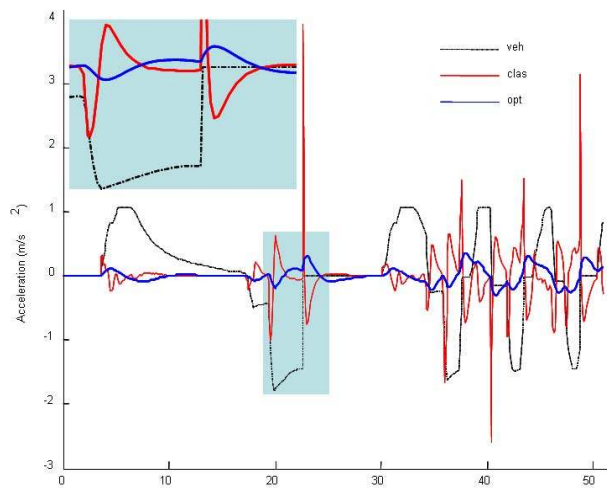Fig. 12. Cabin's positions using optimal and classical algorithms

Fig. 10. Cabin's accelerations using optimal and classical algorithms

classical algorithm. Optimal algorithm (figure 10) has provide the best signal profile acceleration but reduce very much the amplitude of the restituted acceleration. This is due to platform limitations.

In experimentation, it is stated that classical algorithm is the more efficient. Accompanied with other artifacts as backlash algorithm [9] (examlpe: using a non-linear filter), it gives a sufficient results during drive operation. Adaptive algorithm reduce the previous cited false cues, but it is more soft to provide a good acceleration sensation comparing to the classical algorithm. The optimal one provide a non sufficient perception and it is classed as the more bad one applying to our simulator.
We can notice also that the limited available displacement of the platform had strictly constrained the motion restitution, figures (11 and 12). The only longitudinal displacement is then not sufficient to have a good perception.
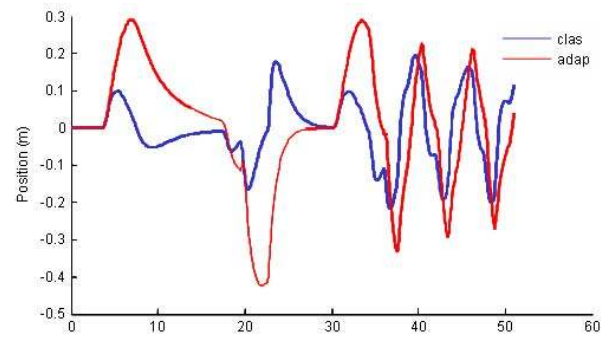
V. CONCLUSION

In this paper, three algorithms (Classical, Adaptive and Optimal) for motion cueing are exposed and experimented on the our low cost platform. The implementation concerns only the longitudinal acceleration restitution. The aim of this study is to compare the performances of each algorithm, and its impact on the driver perception.

The classical washout is the more appropriate for our case in term of human perception and design simplicity. The parameters adjustment is too easy regarding the remaining algorithms. Once the form of the classical filter is defined, a trial-error experimentation are done to establish the value of the different filter parameters. The inconvenience of this method is that some false cues are induced for the brutal changes in acceleration like braking, due to the linear characteristic of the high-pass filter. This can be corrected by inducing other algorithms and artifacts to reduce the backlash. Finally, classical filter are adjusted for the worst case which reduce considerably the displacement of the platform for others maneuvers.

Secondly, adaptive algorithm allows the adjustment of the filter parameters at each time of the driving simulation. This can reduce some false cues generated by the classical algorithm. Nevertheless, we have found that returning back to the neutral position is more slow then the classical algorithm, the perception is more soft mainly at the the beginning of acceleration. The inconvenience of this method is the difficulty to find the most relevant weighting of cost function, and initial values of the different parameters, that gives the best results

while assuring stability.

Optimal algorithm present different lacks. Despite, it minimizes the sensation error between the driver on the virtual vehicle, and the one on the simulator platform. The obtained results are not sufficient to cue a good perception. This due to the physical constraints of our simulator.

Therefore, the use of a tilt-coordination can improve the fidelity of motion. This is not implemented on the platform regarding the financial cost of such implementation on the present simulator. The platform's seat can rotate by small angles to study its impact on the driver behavior, and so to its possibility to replace the whole platform tilt. Subjectively, it is found during psychophysics experimentation previously conducted, that the seat rotation can improve the driver perception for some driving situation,but no objectively explication was found, we continue to work upon.

REFERENCES

[1] A. Kemeny, "Simulation et perception du mouvement," in *Driving Simulation Conference(DSC99)*, Paris, France, July 1999, pp. 33–55.

[2] S. F. Schmidt and B. Conrad, "Motion drive signals for piloted flight simulators," in *Contractor Report NASA CR-1601*, Washington, USA, 1970.

[3] W. R. Sturgeon, "Controllers for aircraft motion simulators," *Journal of Guidance and Dynamics*, vol. 4, pp. 184–191, Mar.-Apr. 1981.

[4] P. R. Grant and L. D. Reid, "Motion washout filter tuning: Rules and requirements," *Journal Of Aircraft*, vol. 34, pp. 145–151, Mar.-Apr. 1997.

[5] M. Idan and M. A. Nahon, "Off-line comparison of classical and robust flight simulator motion control," *Journal of Guidance and Dynamics*, vol. 22, pp. 702–709, Sept.-Oct. 1999.

[6] L. D. Reid and M. A. Nahon, "Response of airline pilots to variations in flight simulator motion algorithms," *Journal Of Aircraft*, vol. 7, pp. 639–646, July 1988.

[7] M. A. Nahon and L. D. Reid, "Simulator motion-drive algorithms: A designer's perspective," *Journal of Guidance and Dynamics*, vol. 13, pp. 356–362, July 1989.

[8] G. Reymond, "Contribution respective des stimuli visuels, vestibulaires et proprioceptifs dans la perception du mouvement du conducteur," in *Paris6 PhD Thesis)*, Paris, France, Dec. 2000.

[9] G. Reymond and A.Kemeny, "Motion cueing in the renault driving simulator," in *Vehicule System Dynamic*, vol. 34, Paris, France, Oct. 2000, pp. 249–259.

[10] R. V. Parrish, J. E. Dieudonne, R. L. Bowles, and D. J. Martin, "Coordinated adaptive washout for motion simulators," *Journal of Aircraft*, vol. 12, pp. 44–50, July 1975.

[11] M. A. Nahon, L. D. Reid, and J. Kirdeikist, "Adaptive simulator motion software with supervisory control," *Journal of Guidance and Dynamics*, vol. 15, pp. 376–383, Mar.-Apr. 1992.

[12] D. Ariel and R. Sivan, "False cue reduction in moving flight simulators," *IEEE Transactions on Systems, Man and Cybernetic*, vol. 14, pp. 665–671, July-Aug. 1984.

[13] R. Sivan, J. Ish-shalom, and J. . Huang., "An optimal control approach to the design of moving flight simulators," *IEEE Transactions on Systems, Man and Cybernetic*, vol. 12, pp. 818–827, July-Aug. 1982.

[14] F. Cardullo, R. Telban, and J. Houck, "Motion cueing algorithms: A human centered appraoch," in *5th International Symposium on Aeronautical Sciences*, Zhukovsky, Russia, 1999.

[15] R. Telban and F. Cardullo, "A nonlinear human centred approach to motion cueing with neurocomputing solver," in *AIAA Modeling and Simulation Technologies Conference and exhibit*, Monterey, California, Aug. 2002.

[16] M. Dagdelen, G. Reymond, A. Kemeny, M. Bordier, and N. Mazi, "Mpc based motion cueing algorithm: development and application to the ultimate driving simulator," in *Driving Simulation Conference (DSC04))*, Paris, France, Sept. 2004.

[17] H. Mohellebi, S. Espie, H. Arioui, A. Amouri, and A. Kheddar, "Low cost motion platform for driving simulator," in *5th International Conference on Machine Automation (ICMA04)*, Osaka, Japan, Nov. 2004.

[18] L. Nehaoua, A. Amouri, and H.Arioui, "Classic and adaptive washout comparaison for a low cost driving simulator," in *Proceedings of the 13th Mediterranean Conference on Control and Automation (MED05)*.

[19] P. A. Ioannou and J. Sun, *Robust adaptive control*. Prentice-Hall Inc, 1995.

[20] S. Espie, "Vehicle-driven simulator versus traffic-driven simulator: the inrets approach-driving simulation conference," in *Driving Simulation Conference (DSC99)*, Paris, France, 1999.

[21] L. Young and J. L. Meiry, "A revised dynamic otolith model," in *Aerospace Medicine*, vol. 39, June 1968, pp. 606–608.