Washout Filter Design for a Motorcycle Simulator

Federico Barbagli, Diego Ferrazzin, Carlo Alberto Avizzano, Massimo Bergamasco
PERCRO
Scuola Superiore S.Anna
via Carducci 40, 56127, Pisa, Italy
carlo, fed, padi, bergamasco@sssup.it

Abstract

Many motion base simulators have been developed in the last thirty years for many different types of vehicles. In order to make a simulation more realistic, linear accelerations and angular rates are exerted on the pilot by moving the platform on which the mock-up vehicle is located. This has to be accomplished without driving the simulator out of its workspace. The software component that is in charge of this is commonly referred to as washout filter.

Washout filters have been widely investigated in the past, mainly in the field of flight simulators. In this article we present a washout filter designed for a motorcycle simulator. The solution is preliminary and follows, as a reference point, techniques previously adopted for large aircraft simulators. Differences between motorcycle and aircraft simulation are analyzed and a preliminary customized solution is proposed. The washout filter, which will be used to drive a motorcycle simulator, currently being built at PERCRO, has been tested off-line showing good results and will soon be tested on real riders.

1 Introduction

Flight simulators have been the reference point in the field of vehicle simulation for the last 30 years. This has been due to the high costs of aircrafts, if compared to other vehicles as cars or motorcycles. Flight simulators have always been less expensive than the actual aircraft they were trying to reproduce, thus allowing pilots and crews to be trained at lower costs and lower risks. The same cannot be said for car and motorcycles and here lays one of the basic differences between such types of simulators. Land vehicle simulators have been developed with different purposes, most often as a tool for designers to test new prototypes before actually building them or to study human behavior in specific situations.

Two main types of simulators can be distinguished: fixed and motion base. The former are based solely on visual and instrument cues while the latter also provide the pilot with realistic motion cues. Which type of simulator is capable of giving the most realistic feeling to its users has been matter of much debate. It has been often acknowledged that a good motion base can significantly enhance simulation realism. On the other side, poorly controlled motion base, which results in erroneous or delayed motion cues, can have an extremely negative effect on the pilot, resulting in an unrealistic simulation. Moreover the advent of high-performance image generation (IG) systems has lead the way to a new class of cheaper fixed base commercial simulators. Nonetheless visual systems alone can provide motion cues only at low frequency and therefore motion base is still considered necessary, especially in more expensive research simulators.

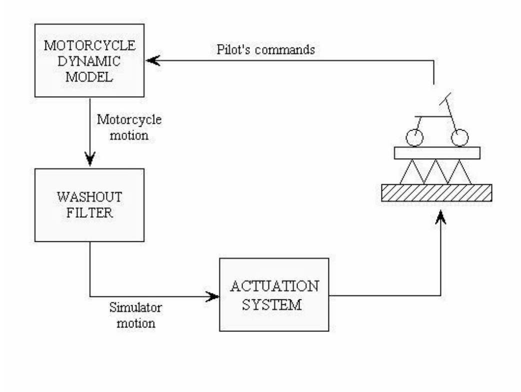

Figure 1. The motorcycle simulator scheme

A simplified structure of a motion base vehicle simulator is sketched in fig. 1. Typically the pilot's commands are sampled and fed to a vehicle *dynamical model unit* (DM) that computes the vehicle's response. The vehicle motion is then processed by a *washout filter* (WF) to produce a desired simulator motion. The trajectories computed by the WF, usually in the form of actuators lengths, are used to command the motion base.

One of the keys to obtain realistic motion cues on the pilot is the WF (whose name originates from the fact that one of its functions is to "wash out" the position of the simulator back to its neutral position [12]). The purpose of the WF is to transform the trajectories generated by the DM, which include very large displacements, into actuators commands capable of providing the pilot with realistic motion cues while remaining within the simulator's limits.

The design of efficient WF is a complex problem. These filters are first of all complex control systems whose robustness and stability must be ensured in order to avoid mechanical damage to the simulator. Moreover washout filters must take into account the nondeterministic nature of pilots which makes it hard to define what "realistic" means and making this a complex design problem in the field of human factors and human-machine interaction. Many different schemes have been proposed in the last twenty years. Classical washout filters ([6],[14]) were the first to be developed, followed by adaptive algorithms ([13], [1]), optimal control filters ([15],[16]), hybrid classical-adaptive filters ([12]) and robust filters ([8], [11]). It is important to note that even though such literature is very extensive, it has been developed specifically for flight simulators. To the authors' knowledge much less has been proposed specifically for the case of land vehicles simulators and almost nothing for 6 DOF motorcycle simulators [2],[17].

The purpose of this article is to present a design procedure for the WF software to be used with a motion base motorcycle simulator. The solution proposed has been tested off-line showing good results and it will soon be tested on the MORIS motion base motorcycle simulator which is currently being built at PERCRO, Pisa, Italy.

The MORIS Esprit project[1] started in 1995 with the aim of developing a two-wheeled motorcycle simulator conceived as a tool for the designer to acquire data on motorcycle maneuverability at the design stage as well as to collect data about rider control behavior implications in motorcycle performances. The MORIS project is being developed by a consortium composed by industrial partners (Piaggio and Humanware from Italy and HEAD acoustics from Germany) and by academic partners (Scuola Superiore S. Anna from Italy, Halmstad University from Sweden and University of Bochum from Germany)

The reminder of this paper is organized as follows. In section 2 the differences between flight and motorcycle simulators are pointed out and consequently the basic structure of the WF is proposed. In section 3 the equations on which the washout algorithm is based are presented. In section 4 a possible simulink implementation of the filter is presented and analyzed. In section 5 a testing procedure is presented and used to evaluate the system's performances. Finally in section 6 some conclusion are drawn and some possible future developments are pointed out.

2 Washout filter

The purpose of the washout filter is to reproduce the angular rates and forces that a pilot would feel if the simulation was real, using a 6 DOF *Stewart Platform* (SP), which has limited workspace. In order to do this the filter's inputs are the linear accelerations and angular rates at the rider's head, since that is where the human vestibular system is located. Such input is then reproduced, in the most realistic way, on the rider's head by moving the platform on which the mock-up is mounted.

Many techniques have been used to implement WF for flight simulators. The most well-known of such techniques, commonly referred to as classic WF, has been chosen as a reference point for the design of a WF that would suit, at best, the specific characteristics of motorcycle dynamics. This is because classical washout filters are relatively simple and transparent to the designer [10] and they usually have good performances. In order to customize a classical WF for our specific motion base simulator the following considerations were made.

1. The dynamics of a land vehicle are very different from the ones of an aircraft. Land vehicles dynamics are usually much faster if compared to the ones of a large aircraft. This is due to a higher power to mass ratio and to the specific nature of moving on the ground, where higher friction is present. Consider, for instance, a car moving longitudinally at a constant speed of $30m/s$. Arresting the car means applying a negative acceleration step signal to the vehicle. Moreover, when the car stops the acceleration drops from a almost constant value, $-7m/s^2$ for instance, to 0 almost instantly. This type of very fast motions are hard to reproduce on the rider using a simulator with limited workspace, and will obviously lead to some errors that are not usually present in aircraft simulators.

2. The dynamics of a motorcycle are different from the ones of a car or an aircraft because of the limited mass of the vehicle. Being the rider's mass comparable to that of the vehicle, the position of the pilot on the motorcycle strongly effects its dynamics and therefore should be considered at all times. An example of this is the fact that a pilot can approach a turn maneuver, on a motorcycle, using almost solely his body lean. Moreover, the rider's head position with respect to the motorcycle, varies much more, during a normal run, if compared to what happens for other larger vehicles. The MORIS simulator uses an encoder sensor, fixed through a mechanical structure to the rider's back, to

[1]This work has been supported in part by EEC ESPRIT project ♯20521.

evaluate the rider's lateral lean angle (see fig. 2). Such measurment is used in many different ways. First of all it is used by the motorcycle DM to compute the effect of lateral lean on the motorcycle lateral and longitudinal motion. Secondly it can be used to estimate the rider's head position in a more precise way. By doing this off-line tests can resemble what the user would really feel in a more precise way. Moreover such information can also be used by the WF, as it will be pointed in the following.

3. Washout filters performances are highly influenced by how their parameters are tuned. As Nahon and Reid point out in [10], "the utility of a given scheme can be vastly improved or degraded by the choice of parameters used". Flight simulators are usually used to train pilots during a specific maneuver. The WF parameters are therefore optimized for the particular dynamics that occur during such operation. The MORIS motorcycle simulator, on the other side, has been created as a tool for motorbike designers to acquire data on prototypes maneuverability before they have actually been built. The simulator must be able, therefore, to reproduce a whole run and thus a wider range of dynamics without being optimized for any specific one. This will lead, in general, to poorer performances for the WF.

The washout filter presented hereafter has been designed taking into account these considerations.

3 General mathematical setting

3.1 Nomenclature

$I_{3\times3}$	$n \times n$ Identity matrix
Σ_B	Base inertial reference frame
Σ_P	Mobile reference frame fixed to the platform
Σ_H	Mobile reference fixed to the user's head
R_j^i	Rotation matrix from Σ_j to Σ_i
T_j^i	Transformation matrix from Σ_j to Σ_i
$\mathbf{v}$	Generic vectorial quantity v
$\mathbf{p}_h^i = [x_h^i, y_h^i, z_h^i]^T$	Head position w.r.t. Σ_i
$\mathbf{a}_F^i$	Linear head acceleration with respect to Σ_i
$\mathbf{a}_D^i$	Desired linear head acceleration with respect to Σ_i
ω_h	Desired angular velocity of the rider's head
h	Distance between Σ_H and Σ_P
b	Distance between Σ_H and the mock-up seat
c_α	$\cos(\alpha)$
s_α	$\sin(\alpha)$
θ, ϕ, ψ	Euler angles describing the orientation of Σ_P w.r.t. Σ_B
$\mathbf{g}$	Gravity vector $[\ 0 \quad 0 \quad g\ ]^T$
g	Gravity acceleration

Figure 2. The encoder measuring the rider's lateral lean angle

3.2 Frames of reference

The human vestibular system, which plays a dominant role in motion sensing, is located inside the bony labyrinth of the ear. It is therefore important to know the linear accelerations and angular velocities to which the pilot's head is subjected during the simulation. In order to do so, several reference frames are associated with the motion base, as shown in fig.3. The following notation is used: Σ_B is an inertial frame fixed to the base of the simulator; Σ_P is a frame of reference fixed to the platform moved by the simulator to recreate the appropriate motion cues on the operator; Σ_H is a frame of reference fixed to the user's head.

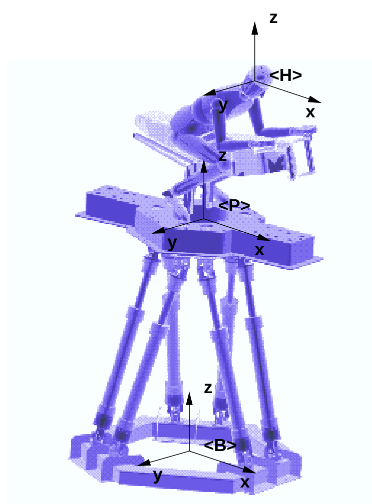

Figure 3. The reference frame used for the simulator

The origin of Σ_B has been chosen to coincide with the centroid of the lower exagon of the SP. By convention the x-axisis parallel to the active runway while the z-axis is vertical pointing upwards, opposite to the gravity force.

The origin of Σ_P has been chosen to coincide with the centroid of the upper exagon of the SP, where the mock up is mounted. By convention the x-axis points forward from the back to the front wheel of the mock-up, the z-axis points upward opposite to the direction of gravity. As a result, the y-axis points to the left side of the motorcycle rider.

Furthermore the origin of Σ_H has been chosen in the geometrical center of the rider's head, to be aligned to those of Σ_P when the simulator is not in use. In normal operating conditions, i.e. when the platform moves, the position of references Σ_H and Σ_P with respect to Σ_B can be computed through the use of appropriate 4×4 matrices representing homogeneous transformation.

3.3 Platform position and orientation with respect to Σ_B

The position and orientation of reference Σ_P with respect to the base reference Σ_B is fully given by matrix

$$T_P^B = \begin{bmatrix} R_P^B & \mathbf{v}_p \\ \mathbf{O} & 1 \end{bmatrix} \quad (1)$$

where $\mathbf{O} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$, $\mathbf{v}_p = \begin{bmatrix} x_p & y_p & z_p \end{bmatrix}^T$ is the position of the origin of Σ_P with respect to Σ_B, and $x_p = 0$, $y_p = 0$ and $z_p = z_{p0}$ is the position of the platform when the simulator is not in use. Moreover R_P^B is the rotation matrix from Σ_P to Σ_B. The standard minimal representations of orientation for a motorcycle [5] is represented by a sequence of three rotations of angles ψ, ϕ and θ (yaw, roll and pitch angles) around the z, y and x-axis of the current frame of reference. Composing such rotations we obtain matrix R_P^B given by

$$R_P^B = \begin{bmatrix} c_\psi c_\phi & c_\psi s_\phi s_\theta - s_\psi c_\theta & c_\psi s_\phi c_\theta + s_\psi s_\theta \\ s_\psi c_\phi & s_\psi s_\phi s_\theta + c_\psi c_\theta & s_\psi s_\phi c_\theta - c_\psi s_\theta \\ -s_\phi & c_\phi s_\theta & c_\phi c_\theta \end{bmatrix} \quad (2)$$

Such angles are used to specify

1. the orientation of the virtual motorcycle w.r.t. a inertial frame of refence, in the DM unit
2. the orientation of Σ_P w.r.t. Σ_B in the WF.

It is important to note that the rotations are always performed w.r.t. the current frame of reference. This is important since given the angular rates $\begin{bmatrix} \dot{\theta} & \dot{\phi} & \dot{\psi} \end{bmatrix}^T$ its integral gives $\begin{bmatrix} \theta & \phi & \psi \end{bmatrix}^T$ while the same would not be true if the rotations were performed around fixed axis.

3.4 Head position and orientation with respect to Σ_P and Σ_B

The position and orientation of reference Σ_H with respect to the base reference Σ_P is fully given by matrix

$$T_H^P = \begin{bmatrix} R_H^P & \mathbf{p}_h^P \\ \mathbf{O} & 1 \end{bmatrix} \quad (3)$$

Such matrix strongly depends on the rider's position and posture while sitting on the mock-up. Two cases can be considered:

1. a simplified approach in which the head position is vertically fixed above the mock-up, i.e. $R_H^P = I_{3\times 3}$ and $p_h^P = \begin{bmatrix} 0 & 0 & h \end{bmatrix}^T$;
2. a more generic approach in which the head has a lateral degree of freedom. If the rider can laterally lean his torso of an angle δ, we obtain

$$R_H^P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\delta & -s_\delta \\ 0 & s_\delta & c_\delta \end{bmatrix} \quad p_h^P = \begin{bmatrix} 0 \\ (h-b) + bs_\delta \\ (h-b) + bc_\delta \end{bmatrix} \quad (4)$$

   As mentioned before angle δ is measured by an encoder through a mechanical structure fixed on the pilot's back.

Moreover being $R_H^B = R_P^B R_H^P$ we obtain $R_H^B = R_P^B$ in the first simpler case. In the following the first simpler case will be proposed.

3.5 Washout location

Accelerations $\mathbf{a}$ and angular rates ω are calculated, by the motorcycle DM software, at a certain location, i.e. referred to a specific frame of reference fixed with a point of the vehicle. The scope of the WF is to compute a platform trajectory, laying completly in the system's workspace, reproducing, as closely as possible, vector $\begin{bmatrix} \boldsymbol{a}^T & \boldsymbol{\omega}^T \end{bmatrix}^T$ at the same physical location, referred to as *washout location*. In order to avoid spurious linear accelerations, the washout location must be coincide for the DM and for the WF. The most typical washout location for flight simulators is the point of the aircraft corresponding to the centroid of the motion base. Anyway, other washout locations have been adopted in the past [3]. In this work the washout location has been chosen in the rider's head. In the future, anyway, different washout locations will be tested on pilots in order to evaluate which may lead to the most realistic effect.

3.6 Motorcycle Dynamical Model

The dynamical model of the motorcycle is based on three frames of reference. One is inertial and it is used to compute an absolute position of the motorbike in the virtual world; one is fixed with the motorcycle with its origin positioned at the projection of the motorcycle's center of mass on the ground along the plane of symmetry of the motorcycle; one is fixed to the pilot's head. Further details can be found in [4]. However it is important to note that the washout location, for both the DM and the WF, is the rider's head. This means that the WF performs calculations using values, computed by the DM, of specific accelerations existing at the rider's head.

3.7 Basic functions of the washout algorithm

The WF implemented follows closely the so called classical scheme. Such scheme is based on three channels. Two are used to reproduce, on the rider, linear accelerations while one is used to reproduce angular rates.

In the following let us focus on the linear accelerations that the WF must reproduce on the rider. Referring to fig. 3, the linear accelerations felt by the simulator rider at his head are given by

$$\mathbf{a}_F^H = -R_B^H\,\mathbf{g} + R_B^H\,\ddot{\mathbf{p}}_h^B \tag{5}$$

In order to reproduce a given acceleration a_D^H (given by DM) on the pilot's head it is possible to divide such target vector into two components, using a high pass and a low pass filter, i.e.

$$\mathbf{a}_D^H = \mathbf{a}_{grav} + \mathbf{a}_{mov}. \tag{6}$$

Such components can be reproduced using two separate channels. The low-frequency components, which would drive the platform out of its workspace, are reproduced by tilting the platform. By doing this it is possible to reproduce a constant acceleration on the rider's head with zero steady-state error. In fact, at steady state, i.e. when $\ddot{p}_h^B = 0$, equation (5) becomes

$$\mathbf{a}_{grav} = -R_B^H\,\mathbf{g} \tag{7}$$

which is satisfied by tilting the platform of the following pitch and roll angles

$$\phi = -\arcsin(\frac{a_{Dx}^B}{g})\ ,\ \theta = \arcsin(\frac{a_{Dy}^B}{g\,c_\phi}) \tag{8}$$

In other words it is always possible to use gravity to reproduce constant longitudinal and lateral accelerations on the rider, while the visual display continues to show horizontal riding conditions. By doing this a certain error is introduced. First of all the acceleration felt by the user along the z-axis of Σ_H, referred to as g', is not exactly equal to g. Such error is very small being $a \ll g$ and therefore $g \approx g'$. Moreover no vertical low frequency components of linear acceleration can be reproduced on the rider using this technique. Finally, since the platform tilting is an artifact generated to trick the rider's senses, it should not be perceived by the rider. Therefore the maximum roll and pitch tilt rate is set to 3 deg/s [7]. This limits the performances of the motion controller unit while attempting to track the desired low frequency accelerations, and introduces a non-linear element that has to be considered when analyzing the overall stability of the WF.

On the other side it is possible to linearly move the platform in order to reproduce $\mathbf{a}_{mov}$ on the rider. Referring to equation (5), matrix R_B^H can now be considered known at any time since it is a function of time set to track $\mathbf{a}_{grav}$. Therefore we obtain

$$\ddot{\mathbf{p}}_h^B = R_H^B\,\mathbf{a}_{mov} \tag{9}$$

which can be used to directly drive the platform. This is because vector $\ddot{\mathbf{p}}_h^B$ is referred to Σ_B and therefore by driving Σ_P linearly, the same acceleration is exerted on Σ_H. It is important to note that $\mathbf{a}_{grav}$ is not tracked instantly by the system due to the limits imposed on the maximum platform tilt rate. One of the side effects of such tilting is to create an undesired acceleration on the rider's head that lasts until conditions (8) are met. This means that equation (9) is not exact and a certain term a_{drift} should be considered in the computation of the linear accelerations driving the platform. However, due to the limits imposed on the tilt rates, such error is usually neglectable.

The same principle with which the WF tracks $\mathbf{a}_{mov}$ can be applied to track the system's high-pass filtered angular rates. It is important to note that the input of the WF is usually given as an angular velocity vector ω_h with respect to the fixed axis of an inertial frame. Such angular rates must be transformed into the derivatives of angles ψ, ϕ and θ in order to be integrated. This is accomplished using equation

$$\omega_h = \begin{bmatrix} 0 & -s_\psi & c_\psi c_\phi \\ 0 & c_\psi & s_\psi c_\phi \\ 1 & 0 & -s_\phi \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \dot{\phi} \\ \dot{\theta} \end{bmatrix} \tag{10}$$

where matrix $R(\psi,\phi)$ is non singular in the range of phisically admissible values of ϕ.

4 Washout filter implementation

The solution adopted for the MORIS simulator is sketched in fig.4.

The upper channel receives, as input, the linear accelerations felt by the rider at his/her head, expressed with respect to Σ_H. The first element of such channel, the Strategy Splitter unit (block (1) in fig. 4), has multiple purposes. First of all $R_B^H\mathbf{g}$ is added to the z_H-axis component of a_D^H. This is

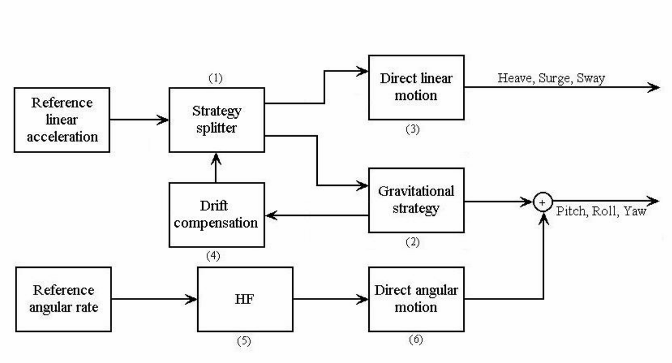

Figure 4. The washout filter scheme

because the DM output along z_H also considers the gravity effect which should not be reproduced by the SM. The resulting vector is then transformed into Σ_B. Finally the resulting vector is divided into high frequency and low frequency components, i.e. into $\mathbf{a}_{grav}$ and $\mathbf{a}_{mov}$ introduced in equation (6). A fourth-order filter has been chosen in order to ensure that the platform is "washed out" back to its zero position after some time, i.e. that $\mathbf{x_{mov}} = \int\int \mathbf{a}_{mov} = 0$ at steady state, for step and ramp inputs. Twelve parameters can be regulated in order to vary the threshold between high and low frequencies to be replicated using different strategies.

The low pass filtered desidered accelerations are fed to the Gravitational Strategy unit (block (2) in fig. 4). The purpose of this block is to track $\mathbf{a}_{grav}$ as presented in equation 7 by tilting the platform around its pitch (ϕ) and roll (θ) angles. As mentioned before this will introduce a certain error along the acceleration felt along the z_H-axis. Moreover saturation modules have been inserted in order to limit roll and pitch rates, since the tilting should not be perceived by the rider, and pitch and roll accelerations since the platform has physical limits to its maximum angular accelerations.

The Gravitational Strategy unit is non-linear due to the saturation blocks and to the coupling effects introduced by matrix R_H^B. As far as the stability is concerned it is possible to show, using Lyapunov's theory, that this unit results asymptotically stable in a ball centered on the system's equilibrium points.

The high pass filtered desidered accelerations are fed to the Direct Linear Motion Unit (block (3) in fig. 4). Its purpose is to integrate twice $\mathbf{a}_{mov}$ and use such information to linearly drive the platform. Since $\mathbf{a}_{mov}$ represents the high frequency components of the accelerations felt on the rider's head, and because of the structure of the Strategy Splitter, such commands will drive the platform for short displacements and will drive it back to its zero position thus making sure that the workspace limits won't be met. However, safety units can be used in order to eventually decrease the linear velocities of the platform in order for it to reach a pre-set safety limit with zero speed. Such unit reproduces unrealistic motion cues on the rider but is not intended to normally operate when the system is tuned properly by the operator. The safety units consider the workspaces along x_B, y_B and z_B-axis as decoupled. This is an approximation but these safety units are nonetheless very useful if the workspace limits are chosen appropriatly.

The angular rates are reproduced on the rider's head in a similar way. First of all the desired angular velocity vector ω_h is transformed into $\begin{bmatrix} \dot\theta & \dot\phi & \dot\psi \end{bmatrix}^T$ using relation (10). Such angles are then high passed filtered and integrated. The HP filter, similar to the one used in block (3) and therefore the angles are washed out at zero position at steady state. Finally such angles are added to the ones computed by the Gravitational Strategy Unit. Strictly speaking the sum of such angles will introduce some approximation since $\begin{bmatrix} \theta & \phi & \psi \end{bmatrix}^T$ is not strictly a vector but an ordered triple. However this operation has been shown to be effective [9] and therefore used in the Moris WF structure.

Block (4) is used to compute spurious accelerations on the rider's head due to the platform tilting introduced by blocks (2) and (6). The Coriolis acceleration $\omega \times (\omega \times p_h^P)$ is computed and fed forward to block (3) in order to be "deleted" using the linear direct strategy.

5 Methodology

The WF and motorcycle DM presented in this paper have been tested offline, since the MORIS simulator is not yet fully operative. An evaluation block computes the accelerations felt by the rider at his head, w.r.t. Σ_H, given a certain platform trajectory in Σ_B and $\begin{bmatrix} \psi & \phi & \theta \end{bmatrix}$. This block considers the linear accelerations computed by the Direct Linear Acceleration block, the gravity effect on the rider and the Coriolis and drift accelerations on the rider's head due to the tilting of the platform. The dynamics of the platform are not included, introducing some approximation in the off-line tests.

The WF has been extensively tested. In order to do so some basic maneuvers have been selected [2]. The outputs of the DM are fed to the WF and compared to its outputs.

The first simulation lasts for 55 seconds. The motorcycle is longitudinally accelerated and then decelerated. No lateral motion is present. The WF tracks the input along the x-axis given by the DM with a very small delay (fig. 5 (a)). Note that the acceleration along the z-axis of Σ_H (fig. 5 (b)) and the pitch rate (fig. 6 (a)) are not very precise, as expected. This is due to the limits introduced by the tilting strategy. Note also that the actuators' lengths always belong to the interval $[1m, 1.6m]$ (fig. 6 (b)), which are the simulators' physical limits, as introduced in [2].

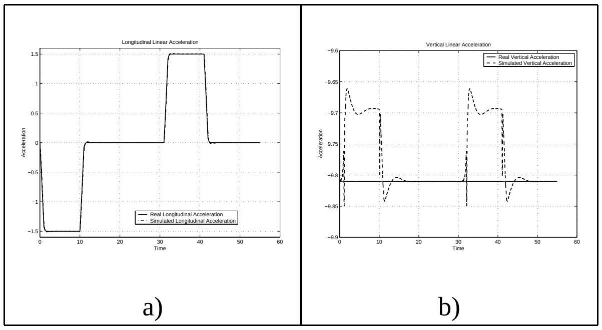

Figure 5. Linear acceleration (along the x_H-axis (a) and z_H axis (b))

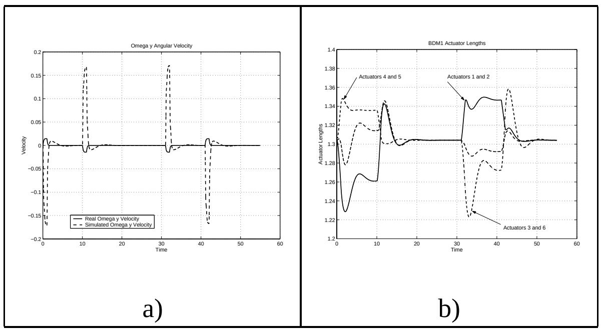

Figure 6. ω_y on the pilot's head (a) and actuators' lengths (b))

The second simulation lasts for 10 seconds. The motorcycle is longitudinally moving at constant velocity behind a truck and is influenced by its wave. The WF tracks the input along the x-axis (fig. 7 (a)) and the input along the y-axis (fig. 7 (b)) with a very small delay, while the input along the z-axis (fig. 8 (a)) as well as the angular velocity components (see fig. 8 (b) for the ω_x) are not very precise, as expected, due to the limits introduced by the tilting strategy.

Finally note that brusque stops are hardly simulated by the WF. Consider the simulation in figure 9, which lasts for 50 seconds. The motorcycle is longitudinally accelerated and then decelerated until a final brusque stop. The WF tracks the input given by the DM with a very small delay. Problems occur, as expected, when the motorcycle stops, in which case the the acceleration drops from $-3.5m/s^2$ to 0 istantly (see fig. 9). Better results can be obtained scaling the input to the WF or low pass filtering it. The steepness of such signal is impossible to be reproduced using a simulator with limited workspace.

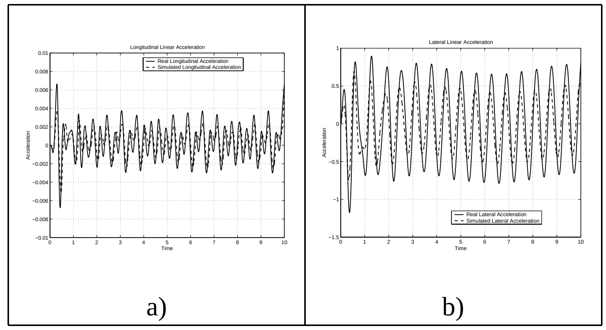

Figure 7. Linear acceleration (along the x_H-axis (a) and y_H axis (b))

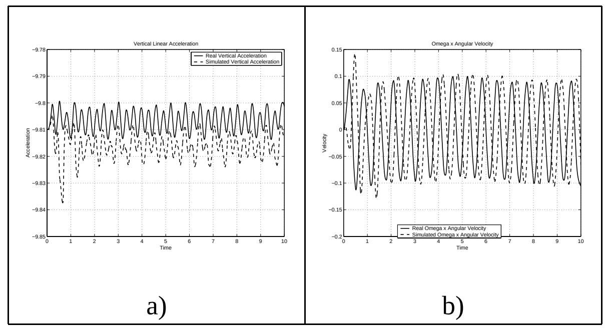

Figure 8. Linear acceleration (along the z_H-axis (a)) and ω_x (b))

6 Conclusions and future developments

The classic washout filter has been implemented for the MORIS motorcycle simulator and tested offline. The filter has been customized for the specific case of a motorcycle. The washout location is always in the pilot's head. In this preliminary version the position of the head has been considered fixed at a constant height above the motorcycle seat.

Future versions of the washout filter will consider the rider's head position with more precision and different washout locations. Such different versions will be tested on real rider's in order to determine which give more realistic feelings.

Moreover preaction units are being studied in order to perfectly simulate a brusque stop, which is reproduced on the rider's head, using the current version of the WF, very poorly. Furthermore a new evaluator unit is being studied which includes a dynamical model of the platform and of its actuators. This should lead to off-line tests evaluating more closely the accelerations felt by the rider due to a given trajectory produced by the WF.

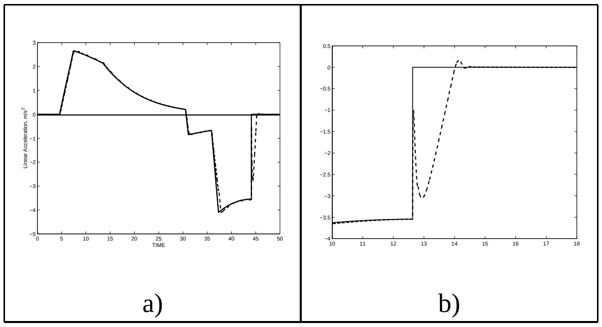

Figure 9. Linear acceleration along the x_H-axis

References

[1] A. D. and S. R. False cue reduction in moving flight simulators. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-14(4):665–671, 1984.

[2] F. et al. Strategy manager implementation in a motion based two wheeled vehicle simulator. In *32nd ISATA*, pages 281–288, June 1999.

[3] P. N. et al. Motion simulation capabilities of three-degree-of-freedom flight simulators. *Journal of Aircraft*, 35(1):9–17, January-February 1998.

[4] B. F. Ferrazzin D. and S. F. A simplified dynamic model for a motorcycle. Technical Report MO-SS-ME-D-DM01, PERCRO, Scuola Superiore S.Anna, Pisa, Italy, September 1999.

[5] I. O. for Standardization. Motorcycle and motorcycle-rider kinematics - vocabulary. Technical Report ISO/DIS 11838, ISO, 1994.

[6] S. J.B. A practical approach to motion simulation. In *AIAA Visual and Motion Simulation Conference*, Palo Alto (CA), September 1973. AIAA paper number 73-931.

[7] R. L.D. and N. M.A. Fligt simulator motion-base drive algorithms: Part 3 - pilot evaluations. Technical Report UTIAS Report 319, Univ. of Toronto, Toronto - Canada, December 1986.

[8] I. M. and S. D. A robust controller for a dynamic six degree of freedom flight simulator. In *AIAA Proc. Of Conf. On Flight Simulator Technologies*, pages 53–60, 1996. paper n. AIAA-99-4238.

[9] N. M.A. and R. L.D. Response of airline pilots to variations in flight simulator motion algorithms. *Journal of Aircraft*, 25(7):639–646, July 1988.

[10] N. M.A. and R. L.D. Simulator motion-drive algorithms: a designer perspective. *Journal of Guidance, Control, and Dynamics*, 13(2):356–362, March-April 1990.

[11] I. Moshe and N. M. A. Off-line comparison of classical and robust flight simulator motion control. *Journal of Guidance, Control, and Dynamics*, 22(5):702–709, September-October 1999.

[12] R. L. Nahon M.A. and K. J. Adaptive simulator motion software with supervisory control. *Journal of Guidance, Control, and Dynamics*, 15(2):376–383, March-April 1992.

[13] B. R. Parrish R.V., Dieudonne J.E. Coordinated adaptive washout for motion simulators. *Journal of Aircraft*, vol 12(1):44–50, 1975.

[14] S. S.F., C. B., and C. B. Motion drive signals for piloted flight simulators. Technical Report Contract NAS2-4869, Analytical Mechanics Associated, May 1970.

[15] I.-S. J. Sivan R. and H. J.K. An optimal control approach to the design of moving flight simulators. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-12(6):818–827, 1982.

[16] W. W. and C. M. Is there an optimum motion cueing algorithm? pilot and aircraft simulators. In *AIAA, Modeling and Simulation Technologies Conference*, New Orleans (LA), August 1997.

[17] M. Y. Yoshimoto K., Kawasaki D. and F. D. Development of a motorcycle simulator using a parallel manipulator and a head mounted display. the ynl bike simulator. In *Accepted at DCS2000*, September 2000.