Fuzzy/PSO Based Washout Filter for Inertial Stimuli Restitution in Flight Simulation

Khaled Fellah, Mohamed Guiatni and Yacine Morsly
Control Laboratory
Ecole Militaire Polytechnique
Bordj El Bahri, Algiers, 16111, Algeria
Email: khaled.fellah@gmail.com, mohamed.guiatni@gmail.com, ymorsly@yahoo.fr

***Abstract*—The aim of this study is to present a new approach using Particle Swarm Optimization (PSO) algorithm and fuzzy logic for motion cueing considering both the physical limits of restitution platforms and realistic sensations. We added the necessary software in order to restitute the specific force based on a virtual aircraft. We used Microsoft Flight Simulator software (MSFS) and built-in data structure and methods. Results using the overall simulation are presented and evaluated on a motion platform. Interesting sensations have been recorded, which enhance the realism of the simulation. The obtained results indicate that the proposed PSO/Fuzzy approach improves the performance of the classical washout filter based motion cueing algorithm.**

Keywords*—Inertial stimuli; flight simulation; washout filter; fuzzy logic; particle swarm optimization.

I. INTRODUCTION

In the last decades, the use of driving simulation for traffic safety, vehicle design and driver perception studies has expanded rapidly [1] [2] [3]. This is largely because simulation saves engineering time and costs, and can be used for studies of road and traffic safety. Simulation is also a useful and indispensable tool for aviation research and training. It has evolved and matured over the last forty years in equal pace with developments in the aerospace industry. Flight simulation allows pilots to fly in simulated conditions, without the costs and safety issues that go with performing real flight.

In addition, recent psychophysical studies have revealed an unexpectedly important contribution of vestibular cues in distance perception and steering, prompting a re-evaluation of the role of visuo-vestibular interaction in driving simulation studies.

When flight simulation and research are combined, the objective is to measure the human performance in the simulated environment [4]. Research will pose certain requirements on the use of simulation hardware and software. It requires generic tools that can be adjusted to the evolving insight in topics. This implies that flight simulators (hardware) and simulation models (software) used for research will often be a trade-off between realism and flexibility [5]. Therefore, a flight simulator must include an aircraft model, a display capability and control hardware [6]. The aircraft model is implemented as software [7]. Three examples of commonly used flight simulator packages are Microsoft Flight Simulator (sometimes abbreviated to MSFS or FS), FlightGear and XPlane.

The flight controls are hardware providing input to the aircraft model. In most cases they are used along with an aircraft-like control input such as a joystick, yoke and rudder pedals.

While a visual system alone can provide motion cues at low frequency, physical motion stimuli are necessary to provide higher frequency cues in the range sensitive to the vestibular and somatosensory systems.

The adjunction of high fidelity motion cues from a moving platform in conjunction with visual motion cues have been shown to produce a rapid onset of vection, or the illusion of motion, thus reducing the delay associated with visual motion alone.

A key element in providing physical stimuli in flight simulators is the cueing algorithm konwn as washout filter that produces the drive signals used to control the motion system hardware.

In this paper, we propose a new approach for motion cueing considering both the physical limits of restitution platforms and realistic sensations objective. The proposed approach integrates fuzzy logic and Particle Swarm Optimization (PSO) algorithm to the classical washout filter. This approach is an improvement of those presented in the litterature. We added the necessary software in order to restitute the specific force based on a virtual aircraft. We used Microsoft Flight Simulator software (MSFS) and built-in data structure and methods. Results using the overall simulation are presented and evaluated on a motion platform.

This paper is organised as follows: Section II gives a background about the evolution of researches on the washout filter. Section III presents the different frames and a brief overview on the classical washout filter priciple and algorithm. Section IV presents the coordinate transformation of the orientation between the body-fixed simulator reference frame and the inertial reference frame. In Section V, we present the fuzzy logic washout filter design and the optmization algorithm. Section VI presents the Microsoft Flight Simulator software which is used to perform the simulation. Finally, we present the implementation details and some results about the fuzzy logic based motion cueing algorithm for flight simulation.

II. BACKGROUND ON MOTION CUEING ALGORITHMS

Many related researches on washout filter have been presented in the last three decades. Classical washout filter is

 ISBN: 978-1-61208-296-7

the first scheme that has been proposed, which is composed of linear low-pass and high-pass filters. Its advantages are simplicity and easy to adjust. The fixed scheme and parameters of the classical washout filter cause inflexibility of the scheme and make the resulting simulator fail to suit all circumstances. In [8] [9], the authors proposed coordinated adaptive filters, used to coordinate translational and rotational motion.

Nahon and Reid [10] suggested an adaptive washout algorithm with the same scheme as that for classical washout filter and with self-tuning of the filter parameters. Sivan et al. [11] proposed an optimal washout algorithm which takes into account the vestibular system models. This algorithm uses techniques of optimal control and employs a cost function that depends on both sensation error and platform motion. The optimal algorithm designs an optimal structure and a set of optimal parameters subject to the assumptions of human vestibular models and platform limitation by solving the Riccati equation [11]. Telban et al. [12] formulated a linear optimal control problem similar to [11] and solved the Riccati equation in real time so that a scalar coefficient that increases control action can be tuned online. The magnitude of the scalar coefficient depends upon the platform motion.

With large platform motion, the large coefficient increases and results in faster control action. Nehaoua et al. [13] applied classical, adaptive, and optimal algorithms and compared performances of these algorithms in their driving simulator. Authors in [14] propose the use of a linear optimal controller synthesized with a quadratic performance index, which has been applied to a planar model of the Vertical Motion Simulator. Idan and Nahon [15] proposed the use of a robust controller design based on the model of the motion-base dynamics and control in order to compensate for changes or uncertainties in the motion-base dynamics, particularly electrically driven ones with limited bandwidth and power. In [16], a variation of the optimal algorithm was formulated, which incorporate the models of the human vestibular sensation system, i.e., the semicircular canals and otoliths. No matter what kind of platform is used as the simulator, the limited workspace is an important issue in designing the motion cueing algorithm. Several works that increase the efficiency of the platform workspace have been presented.

Huang and Fu [17] proposed a senseless manoeuvre that moves the operator with the acceleration under the threshold value of human perception to conserve the workspace. Liao et al. [18] combined the classical washout filter with an adjustable scaling filter, a yawing washout filter, a dead zone washout filter, and an adaptive washout filter in order to complete the motion planning of the simulator in restricted workspace. A common method to this problem is the fuzzy control-based washout filter proposed by [19] [20], which propose to tune the washout filter parameters based on fuzzy logic in real-time. The cut-off frequencies of the filters are adjusted according to the workspace margins and driving conditions. That algorithm combined characters of motion perception and actual movement of a train to design the fuzzy control rules. However, no optimization process is included in these works. In [21], an optimal washout filter is developed by applying the algorithm of Sivan et al. [11] to the human vestibular system. In this later, the fuzzy controller is used to compensate the filtred signals.

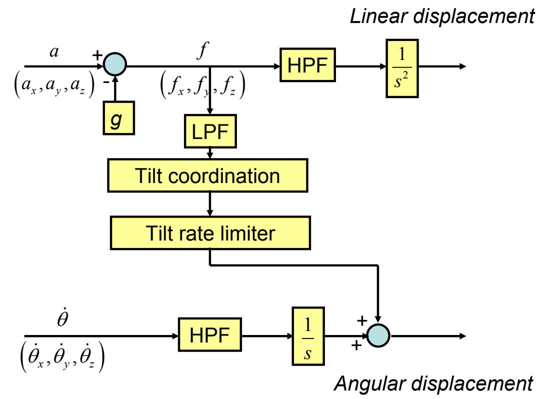

Fig. 1: Classical washout filter flowchart.

The aim of this work is minimize the sensation error produced from the comparison of the human vestibular signals about actual vehicle and simulator. The optimal motion cueing algorithm is then featured by systematic integration of linear filters that are determined through an off-line design process. For the drawback with fixed parameters of the optimal algorithm, the fuzzy control rules are designed to reduce the sensation errors. In order to further eliminate the sensation errors, we propose a new motion cueing algorithm to be used in a research flight simulator using the Particle Swarm Optimization algorithm.

III. CLASSICAL WASHOUT FILTER

A vestibular system, consisting of semicircular canals and otoliths, has a great role in sensing physical motion cues in a motion simulator. It is known that the otoliths are able to detect the high-frequency components of the linearly accelerated motion via a specific force, while the semicircular canals can sense the high-frequency components of the angular motion. Due to the structural limit of the Stewart-Gough platform [22] [8], a motion base has relatively small workspace compared with the operating range of a real aircraft. Because of this limited motion envelope of typical motion bases, filtering is required between the aircraft motion computed from aircraft dynamics and the simulated motion commanded to the motion base. The filter used for this purpose is called a 'washout filter'.

Fig. 1 shows the classical washout algorithm which consists of a High-Pass Filter (HPF) and a Low-Pass Filter (LPF) [23]. The linear acceleration (i.e., specific force) and the angular velocity from the vehicle dynamics are sent to the washout algorithm. The details on how the linear and angular displacements are computed in the washout filter are discussed below. The specific force f_x, f_y, and f_z are computed by subtracting g from the given linear acceleration a. The high-frequency components of the specific force f_x are then obtained by the following 2^{nd}-order HPF:

$$\frac{f_{Hx}}{f_x} = \frac{s^2}{s^2 + 2\zeta_{Hx}\omega_{Hx}s + \omega_{Hx}^2} \tag{1}$$

where ζ and ω represent the damping ratio and the cutoff frequency of the HPF, and the subscript H denotes the high-frequency. Note that f_{Hy} and f_{Hz} in the y and z axes are similarly obtained. This HPF filters out low-frequency motion signals, which tend to lead to large linear displacements that

 ISBN: 978-1-61208-296-7

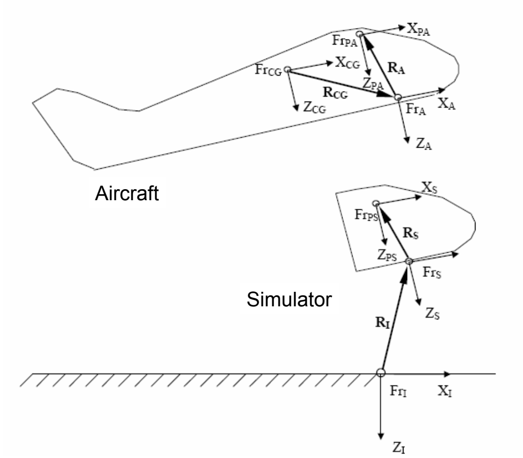

Fig. 2: Frames definition.

cause the motion base to reach its motion limits. The high-frequency motion signals are realized by linear motion of the platform of the motion base by double integration of the filtered signals. Low-frequency components of the specific force in the x axis can be obtained by the following 2nd-order LPF:

$$\frac{f_{Lx}}{f_x} = \frac{\omega_{Lx}^2}{s^2 + 2\zeta_{Lx}\omega_{Lx}s + \omega_{Lx}^2} \tag{2}$$

where the subscript L denotes the low-frequency.

Note that fly is similarly obtained. This low frequency portion is realized by tilting the platform (i.e., angular motion about the $x-$ and $y-$ axes), since a human perceives as if he experienced a specific force when his body tilts. This is referred to as *tilt coordination*. Usually, a pitch angle is set to $\theta_{ytilt} = -f_{Lx}/g$ and a roll angle to $\theta_{xtilt} = -f_{Ly}/g$ for relatively small angles. The rate of tilting is limited to $l - 5°$/s to prevent the semicircular canal from perceiving this signal for tilt coordination as the angular motion cue [10]. In the meantime, the high-frequency components of angular velocity about the $x-$ axis are obtained by the following first-order HPF :

$$\frac{\dot{\theta}_{Hx}}{\dot{\theta}_x} = \frac{s}{s + \omega_{H\theta x}} \tag{3}$$

Note that $\dot{\theta}_{Hx}$ and $\dot{\theta}_{Hx}$ about the y and z axes are similarly obtained. The high-frequency components are integrated to give the angular displacement $(\theta_{Hx}, \theta_{Hy}, \theta_{Hz})$ of the platform. The orientation angle (θ_x, θ_y) about the x and y axes are then obtained by adding the tilt angles $(\theta_{xtilt}, \theta_{ytilt})$ to the angles $(\theta_{Hx}, \theta_{Hy})$ due to angular motion, while θ_z is given as θ_z since no tilt coordination is performed about this axis. It should be noted that the cross feed channel in Fig. 1 includes a rate-limiting element that keeps the tilt rates below 3 °/s.

A. Reference Frames

A series of reference frames are used in the definition of the motion cueing algorithms. These reference frames are adopted from [24], defined below and shown in Fig. 2.

B. Aircraft Center of Gravity

The aircraft center of gravity reference frame Fr_{CG} has its origin at the center of gravity of the aircraft. Frame Fr_CG has an orientation for X_{CG}, Y_{CG}, and Z_{CG} that is parallel to reference frames Fr_S and Fr_A.

C. Simulator Frame

The simulator reference frame Fr_S has its origin at the centroid of the simulator payload platform, i.e., the centroid of the upper bearing attachment points. The origin is fixed with respect to the simulator payload platform. X_S points forward and Z_S points downward with respect to the simulator cockpit, and Y_S points toward the pilot's right hand side. The $x - y$ plane is parallel to the floor of the cockpit.

D. Aircraft Frame

The aircraft reference frame Fr_A has its origin at the same relative cockpit location as the simulator reference frame Fr_S. Fr_A has the same orientation for X_A, Y_A, and Z_A with respect to the cockpit as the simulator frame Fr_S.

E. Inertial Frame

The inertial reference frame Fr_I is earth-fixed with Z_I aligned with the gravity vector g. Its origin is located at the center of the fixed platform motion base. X_I points forward and Y_I points to the right hand side with respect to the simulator pilot.

F. Reference Frame Locations

In Fig. 2, four vectors are illustrated, which define the relative location of the reference frames. R_I defines the location of Fr_S with respect to Fr_I. R_S defines the location of Fr_{PS} with respect to Fr_S. Similarly, R_A defines the location of Fr_{PA} with respect to Fr_A, where $R_A = R_S$. R_{CG} defines the location of Fr_A with respect to Fr_{CG}.

IV. Coordinate Transformations

The orientation between the body-fixed simulator reference frame Fr_S and the inertial reference frame Fr_I can be specified by three Euler angles: $\beta = [\varphi, \theta, \psi]$ that define a sequence of rotations that carry Fr_S into Fr_I. A vector V expressed in the two frames can be related by the transformation matrix L_{SI} (Fr_S to Fr_I), $V^I = L_{SI}V^S$ with:

$$L_{SI} = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \tag{4}$$

The angular velocity of Fr_S with respect to Fr_I can be related to the Euler angle rates $\dot{\beta}$ by the following expression. Let ω_s^s represent the components of this angular velocity in frame Fr_S, then $\dot{\beta} = T_s\omega_s^s$, where :

$$T_s = \begin{bmatrix} 1 & sin\phi tan\theta & cos\phi tan\theta \\ 0 & cos\phi & -sin\phi \\ 0 & sin\phi/cos\theta & cos\phi/cos\theta \end{bmatrix} \tag{5}$$

The specific force is defined as:

$$f_{pS}^S = a_{pS}^s - g^s \tag{6}$$

where a_{pS}^{s} is the linear acceleration and g^s is the gravitational acceleration. It is noted that human sense of motion is more sensitive to high-frequency components of the motion than low-frequency ones.

A. Nonlinear Input Scaling

Limiting and scaling are applied to both aircraft translational input signals a_s^s and rotational input signals ω_s^s. Limiting and scaling modify the amplitude of the input uniformly across all frequencies. Limiting is a nonlinear process that clips the signal so that it is limited to be less than a given magnitude. Limiting and scaling can be used to reduce the motion response of a flight simulator. A third-order polynomial scaling has been implemented in the simulator motion cueing algorithms.

When the magnitude of the input to the simulator motion system is small, the gain is desired to be relatively high, or the output will be below the pilot's perception threshold. When the magnitude of input is high, the gain is desired to be relatively low or the simulator may attempt to go beyond the hardware limits. Let us define the input as x and the output as y. Now, define x_{max} as the expected maximum input and y_{max} as the maximum output, and s_0 and s_1 as the slopes at $x = 0$ and $x = x_{max}$, respectively. Four desired characteristics for the nonlinear scaling are expressed as:

$$\begin{cases} x = 0 \Rightarrow y = 0 \\ x = x_{\max} \Rightarrow y = y_{\max} \\ \dot{y}|_{x=0} = s_0 \\ \dot{y}|_{x=x_{\max}} = s_1 \end{cases} \tag{7}$$

A third-order polynomial is then employed to provide functions with all the desired characteristics. This polynomial will be of the form:

$$y = c_0 + c_1 x + c_2 x^2 + c_3 x^3 \tag{8}$$

where :

$$\begin{cases} c_0 = 0 \\ c_1 = s_0 \\ c_2 = x_{\max}^{-2} \left(3y_{\max} - 2s_0 x_{\max} - s_1 x_{\max}\right) \\ c_3 = x_{\max}^{-3} \left(s_0 y_{\max} - 2y_{\max} + s_1 x_{\max}\right) \end{cases} \tag{9}$$

V. Proposed Washout Filter Design

Washout filters have parameters that can be adjusted to alter the motion responses of a simulator. Since the parameters of most classical washout filters remain fixed during their operation, they cannot cope with various flight conditions efficiently. On-line tuning of the washout filters can enable the simulator to offer better simulator motions within the limited motion range. As part of the design of the experiment, the underlying factor for the entire project was the sensed specific force that results from the aircraft motion. This section focuses on design of a fuzzy model, which shall generate the adequate parameters for the eight filters.

A. Fuzzy Logic Washout Filter Design

1) Fuzzification: In this research, fuzzy logic [25] is suggested to tune the filter parameters, since it is known that the fuzzy logic is effectively applicable to a system in which the input-output relations are not clearly identified. To this end, the fuzzy logic is used which takes the Displacement Limit (DL), the Angle Limit (AL) and the Low-Frequency Specific Force (LFSF) as inputs and then provides the proper parameters of the six HPFs and the two LPFs as outputs. Fig. 3 illustrates the flowchart showing this fuzzy logic system. Triangular functions are used for each variable. There were 5 linguistic terms for each input variable. These terms are:

- Displacement limit (DL) & angle limit (AL): VF (Very Far), F(Far), M(Medium), N(Near) and (VN) Very Near.
- Low-frequency specific force (LFSF): VS (Very Small), S (Small), M (Medium), B (Big) and VB (Very Big).

2) Rule-based system: Once the proper inputs were created, the next step is to create the output's characteristic behavior. The output variables was classified as :

- HPF cutt-off frequencies: VS (Very Small), S (Small), M (Medium), B (Big) and VB (Very Big).
- LPF cutt-off frequencies: VSF (Very Small with Far Limit), VSM (Very Small with Medium Limit), VSN (Very Small with Near Limit), SF (Small with Far Limit), SM (Small with Medium Limit), SN (Small with Near Limit), MF (Medium with Far Limit), MM (Medium with Medium Limit), MN (Medium with Near Limit), BF (Big with Far Limit), BM (Big with Medium Limit), BN (Big with Near Limit), VBF (Very Big with Far Limit), VBM (Very Big with Medium Limit), VBN (Very Big with Near Limit).

The next step is to create a rule base that would govern the operation of the fuzzy system. The proper conditions must be created to implement a system that will allow for perceptible specific force reproduction while taking into account the motion limits of restitution platform. The fuzzy rules are shown in Tables I and II.

In order to select the optimal fuzzy membership parameters, we use the Particle Swarm Optimization algorithm (PSO). Some of the attractive features of the PSO include the ease of

TABLE I. Rule based system for the HPF cutt-off frequencies selection.

Displacement limit (DL)	VF	F	M	N	VN
or angle limit (AL)	VF	F	M	N	VN
HPF cutt-off frequencies	VS	S	M	B	VB

TABLE II. Rule based system for the LPF cutt-off frequencies selection.

Angle limit (AL) / Low-frequency specific force (LFSF)	VF	F	M	N	VN
VS	VBN	VBF	BN	BM	BF
S	MN	MM	MF	SN	SM
M	VBF	BN	BM	BF	SF
B	MM	MF	SN	SM	VSN
VB	BN	BM	BF	VBM	VBF

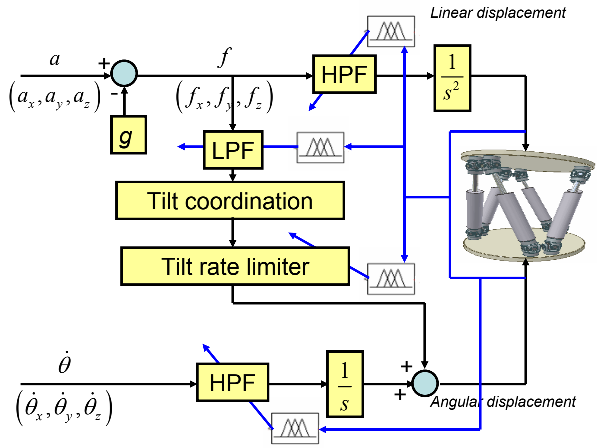

Fig. 3: Fuzzy/washout filter architecture.

implementation and the fact that no gradient information is required. In addition, PSO has the same effectiveness (finding the true global optimal solution) as the Genetic Algorithms (GA) but with significantly better computational efficiency (less function evaluations).

B. Particle Swarm Optimization Algorithm (PSO)

PSO is a stochastic optimization algorithm [26] [27] [28]. The main idea of the PSO is the mathematical modeling and simulation of the food searching activities of a flock of birds. In the multidimensional space, each particle is moved toward the optimal point by changing its position according to a velocity. The velocity of a particle is calculated by three components: inertia, cognitive, and social. The inertial component simulates the inertial performance of the bird to fly in the previous direction. The cognitive component models the memory of the bird about its previous best position. The social component models the memory of the bird about the best position among the particles. The particles move around the multidimensional search space until they find the optimal solution. Based on the above discussion, the mathematical model for PSO is given as follows:

$$\begin{cases} V_i^{t+1} = \omega.V_i^{t+1} + c_1.rand_1\,(.)\,.\,(Pbest_i - X_i^t) \\ \qquad +c_2.rand_2\,(.)\,.\,(Gbest - X_i^t) \\ X_i^{t+1} = X_i^t + V_i^{t+1}, \quad i = 1, 2, 3, ..., N_{swarm} \end{cases} \tag{10}$$

where, i is the index of each particle, t is the current iteration number, $rand_1$ and $rand_2$ are random numbers between 0 and 1. $Pbest_i$ is the best previous experience of the i^{th} particle that is recorded. $Gbest$ is the best particle among the entire population. N_{Swarm} is the number of the swarms. Constants c_1 and c_2 are the weighting factors of the stochastic acceleration terms, which pull each particle towards the $Pbest_i$ and $Gbest$. ω is the inertia weight. As indicated in (10), there are three tuning parameters; ω, $c1$, and c_2 that each of them has a great impact on the algorithm performance. The inertia weight ω controls the exploration properties of the algorithm. The learning factors c_1 and c_2 determine the impact of the personal best $Pbest_i$ and the global best $Gbest$, respectively.

If $c_1 > c_2$, the particle has the tendency to converge to the best position found by itself $Pbest_i$ rather than the best position found by the population $Gbest$, and vice versa. Most implementations use a setting with $c_1 = c_2 = 2$. To implement the PSO algorithm to solve our problem, the following steps were taken:

- *Step 1:* The initial population and initial velocity for each particle should be generated randomly.
- *Step 2:* The objective function is to be evaluated for each individual.
- *Step 3:* The individual that has the minimum objective function should be selected as the global position.
- *Step 4:* The i^th individual is selected.
- *Step 5:* The best local position ($Pbest$) is selected for the i^th individual.
- *Step 6:* The modified velocity for the i^th individual needs to be calculated based on the local and global positions and (10)).
- *Step 7:* The modified position for the i^th individual should be calculated based on (10) and then checked with its limit.
- *Step 8:* If all individuals are selected, go to the next step, otherwise $i = i + 1$ and go to step 4.
- *Step 9:* If the current iteration number reaches the predetermined maximum iteration number, the search procedure is stopped, otherwise go to step 2.

The objective function is to minimize the sensation error between the spectific force estimated on the aircraft and the one restituted on the simulator based on the proposed fuzzy washout algorithm with respect to the following constraints:

- The negative acceleration is limited to $0.17m/s^2$ after high pass filtering.
- The washout rate is limited to $0.048m/s$.
- The output of the low pass filter after tilt coordination is limited to $5°/s$.

The last $Gbest$ is the solution of the problem.

VI. Experiments and Implementation Details

A. Virtual aircraft model and instrumentation

The use of non-flight-certified Commercial-Off-The-Shelf (COTS) solutions has proved its fidelity and coordination characters as a flight training device [29]. MSFS is a flight simulator program, marketed and often seen as a video game. However, it is less a game than an immersive virtual environment since it is very realistic, (see Fig. 4). Its first version appeared in 1982, whereas, its most recent versions, Century of Flight and Flight Simulator X appear respectively in 2003 and 2006 [30].

The long history, the consistent popularity and the open nature of flight simulator structure have encouraged a very large body of freeware and payware add-on packages to be developed. These add-ons, widely available over the internet, are very helpful because, they, not only, permit to change internal aspects of the simulator (airplanes, scenery...), but also to interface it with external software and hardware such as

Fig. 4: Virtual Mi24 helicopter under the MSFS environment.

Fig. 5: The *MOOG 6DOF 2000E 170 E131* motion platform.

homebuilt cockpits. Among the most famous add-ons, we cite a dynamic link library add-on made by Pete Dowson [31], called Flight Simulator Universal Inter-Process Communication (FSUIPC). This module is designed to allow external (i.e., separate) programs to communicate with and control MSFS in real time. In other words, they permit to read from and write in MSFS while it's running and place it in a 64 Kb buffer. So to achieve a variable we have only to know its offset (address) in this buffer. However, before writing or reading data from FSUIPC, we have to scale it to the desirable unit.

Our algorithm was developed to achieve the desired motion cues at an update rate of 60 Hz. Since the computer image generator, which provides the out-the window visual imagery to the simulator pilot, also runs at 60 Hz, the motion cues would be synchronous with the visual cues. The HPF and PHF filters are implemented using the tustin operator which consists to replace the Laplace operator by $s = \frac{2}{Ts}\frac{1-z^{-1}}{1+z^{-1}}$, where z is the discret operator, and Ts is the sample time. The proposed algorithm was successfully implemented on the *MOOG 6DOF 2000E 170 E131* motion platform (see Fig.5).

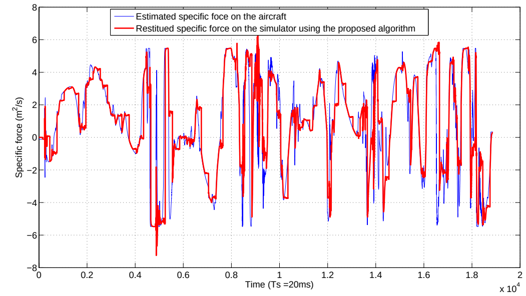

Fig. 6: Real specific force *vs* restituted specific force using the proposed algorithm.

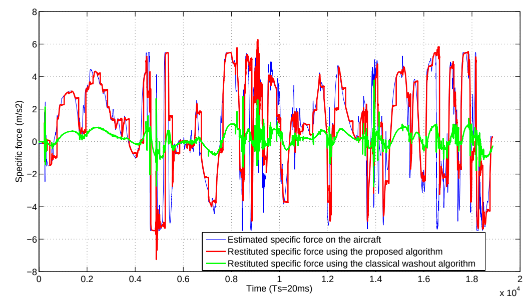

Fig. 7: Restituted specific force using the classical washout algorithm *vs* restituted specific force using the proposed algorithm.

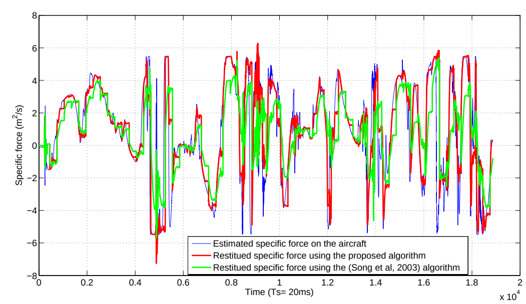

Fig. 8: Restituted specific force using the [19] algorithm *vs* restituted specific force using the proposed algorithm.

B. Results and Discussion

Figs. 6 to 9 show an example of simulated manoeuvres for a MI24 helicopter. Fig. 6 presents comparison between the specific force estimated on the aircraft and that restituted on the simulator based on the proposed algorithm. The specific force is estimated using the vestibular model as explained above.

Fig. 7 shows a comparison between the produced specific force using the classical washout algorithm and that produced using the proposed algorithm. We remark from this figure that the specific force signal produced using our algorithm is highly important and closer to the real signal in comparison with the result obtained using the classical algorithm.

While comparing the obtained results in Fig. 8, we can see that the proposed algorithm allows to get a specific force closer to the one estimated on the aircraft in comparison with the results obtained using the approach proposed in [19].

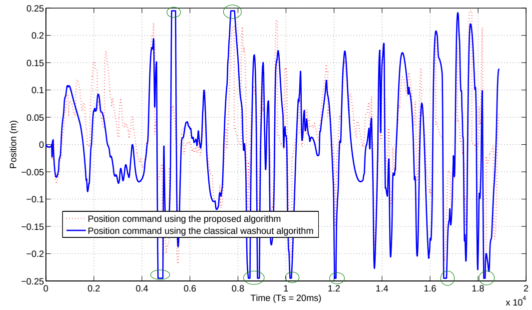

Fig. 9: Comparison of the position cammands using the classical washout algorithm *vs* restituted the position cammands using the proposed algorithm.

Therefore, it allows to the pilot to get a realistic sensation on the simulator. In addition, the magnitude of position commands sent to the motion platform are always within its displacement limits, as seen from Fig. 9. In this way, we can preserve the motion platform from damage and maintain its functionality.

VII. CONCLUSION

This paper reported a successful achievement of a new fuzzy washout algorithm implementation for inertial stimuli restitution in Flight Simulation. The integration of the PSO algorithm has allowed to select the optimal parameters of the fuzzy models. The related built-in software have been integrated successfully with COTS software that provides the aircraft dynamics in a virtual environment. The different simulations proved the effectiveness of the proposed algorithm.

REFERENCES

[1] A. Kemeny and F. Panerai, “Evaluating perception in driving simulation experiments,” in *Trends in Cognitive Sciences*, vol. 7, no. 1, 2003, pp. 31–37.

[2] D. Katzourakis, J. de Winter, S. de Groot, and R. Happee, “Driving simulator parameterization using double-lane change steering metrics as recorded on five modern cars,” in *Simulation Modeling Practice and Theory*, vol. 26, 2012, pp. 96–112.

[3] M. Hoedemaeker and K. Brookhuis, “Behavioural adaptation to driving with an adaptive cruise control (acc),” in *Transportation Research Part F*, vol. 1, no. 2, 1998, pp. 95–106.

[4] S. Valenti Clari, R. C. Ruigrok, B. W. Heesbeen, and J. Groeneweg, “Research flight simulation of future autonomous aircraft operations,” in *Proceedings of the Conference of European Aerospace Societies (CEAS)*, vol. 2, 8-11 Dec 2002, pp. 1126–1134.

[5] J. Hoekstra, “The smart software-simple hardware concept for maximum flexibility in research flight simulation,” *Proceedings of the Conference of European Aerospace Societies (CEAS)*, vol. NLR-TP-96215, p. 16, Amsterdam, the Netherlands 1995.

[6] H. Teufel, H. Nusseck, K. Beykirch, J. Butler, M. Kerger, and H. Blthoff, “MPI Motion Simulator: Development and Analysis of a Novel Motion Simulator,” in *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Hilton Head, South Carolina, 20-23 August 2007, pp. 1–11.

[7] F. Holzapfel, I. Sturhan, , and G. Sachs, “Low-cost pc based flight simulator for education and research,” *AIAA Modeling and Simulation Technologies Conference and Exhibit*, no. 1, p. 1540, 5-8 August 2002.

[8] R. Parrish, J. Dieudonne, D. Martin, and R. Bowles, “Coordinated adaptive filters for motion simulators,” in *Summer Computer Simulation Conference*, 1973, pp. 295–300.

[9] R. Parrish, J. Dieudonne, and R. Bowles, “Coordinated adaptive washout for motion simulators,” in *J. Aircraft*, vol. 12, no. 1, 1975, pp. 44–50.

[10] M. A. Nahon, L. D. Reid, and J. Kirdeikis, “Adaptive Simulator Motion Software with Supervisory Control,” in *Journal of Guidance, Control and Dynamics*, vol. 15, no. 2, 1992, pp. 376–383.

[11] R. Sivan, J. Ish-Shalom, and J. Huang, “An optimal control approach to the design of moving flight simulators,” in *IEEE Trans. On Systems, Man, and Cybernetics*, vol. 12, no. 6, 1982, pp. 818 – 827.

[12] R. Telban, F. Cardullo, and J. Houck, “A nonlinear, human-centered approach to motion cueing with a neurocomputing solver,” in *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Monterey, California, 2002.

[13] L. Nehaoua, H. Mohellebi, A. Amouri, H. Arioui, S. Espié, and A. Kheddar, “Design and control of a small-clearance driving simulator,” in *IEEE Transactions on Vehicular Technology*, vol. 57, no. 2, 2008, pp. 736 – 746.

[14] W. Sturgeon, “Controllers for aircraft motion simulators,” in *Journal of Guidance and Control*, vol. 4, no. 2, 1981, pp. 184–191.

[15] M. Idan and M. Nahon, “Off-line comparison of classical and robust flight simulator motion control,” in *J. Guidance, Control and Dynamics*, vol. 22, no. 5, 1999, pp. 184–191.

[16] R. Telban, F. Cardullo, and J. Houck, “Developments in human centered cueing algorithms for control of flight simulator motion systems,” in *Modeling and Simulation Technologies Conference and Exhibit*, 1999, pp. 1–10.

[17] C. Huang and L. Fu, “Human vestibular based (HVB) senseless maneuver optimal washout filter design for VR-based motion simulator,” in *IEEE International Conference on Systems, Man and Cybernetics*, 2006, pp. 4451–4458.

[18] C. Liao, C. Huang, and W. Chieng, “A novel washout filter design for a six degree-of-freedom motion simulator,” in *JSME International Journal, Series C*, vol. 47, no. 2, 2004, pp. 626 – 636.

[19] J. Song, U. Jung, and H. Ko, “Washout algorithm with fuzzy-based tuning for a motion simulator,” in *KSME International Journal*, vol. 17, no. 2, 2003, pp. 221–229.

[20] X. Wang, L. Li, and W. Zhang, “Research on fuzzy control washout algorithm of locomotive driving simulator,” in *Proceedings of the 7th World Congress on Intelligent Control and Automation*, Chongqing, 2008, pp. 3737 – 3741.

[21] S. Chen and L. Fu, “An optimal washout filter design with fuzzy compensation for a motion platform,” in *18th IFAC World Congress*, Milano, Italy, 2011, pp. 8433–8438.

[22] D. Stewart, “A platform with six-degrees-of-freedom,” in *Proc. Inst. of Mechanical Engineers, part 1*, vol. 180, no. 5, 1966, pp. 371–386.

[23] P. Grant and L. Reid, “Motion washout filter tuning: Rules and requirements,” in *AIAA Flight Simulation Technologies*, vol. 34, no. 2, 1997, pp. 145–151.

[24] W. Wu, “Development of cueing algorithm for the control of simulator motion systems,” Ph.D. dissertation, State University of New York at Binghamton, Binghamton, NY, 1997.

[25] R. Precup and H. Hellendoorn, “A survey on industrial applications of fuzzy control,” in *Computers in Industry*, vol. 62, no. 3, 2011, pp. 213–226.

[26] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *IEEE Int Conf Neural Networks*, 1995, pp. 1942 – 1948.

[27] P. Bajpai and S. Singh, “Fuzzy adaptive particle swarm optimization for bidding strategy in uniform price spot market,” in *IEEE Trans Power Syst*, vol. 22, no. 4, 2007, pp. 2152–2160.

[28] T. Niknam, “A new fuzzy adaptive hybrid particle swarm optimization algorithm for non-linear, non-smooth and non-convex economic dispatch problem,” in *Journal of Applied Energy*, vol. 87, 2010, pp. 327–339.

[29] S. Zheng, S. Zheng, and J. Han, “COTS and design pattern based high fidelity flight simulator prototype system,” in *Journal of Computers*, vol. 6, no. 1, January 2011, pp. 28–35.

[30] “http://www.microsoft.com/games/flightsimulator/,” *Microsoft Flight Simulator 2004*, [retrieved: July, 2013].

[31] “http://www.schiratti.com/dowson,” [retrieved: July, 2013].

 ISBN: 978-1-61208-296-7