FUZZY SLIDING MODE CONTROL OF A FLIGHT SIMULATOR MOTION BASE

Mauricio Becerra-Vargas, Eduardo M. Belo
Engineering School of Sao Carlos-University of Sao Paulo, Sao Carlos, Sao Paulo, 13560-970, Brazil

Keywords: *Sliding Mode Control, Fuzzy Logic, Lyapunov Stability, Flight Simulator, Stewart Platform*

Abstract

The purpose of this study is to apply sliding mode control integrated with fuzzy control for a six degree of freedom flight simulator motion system. Fuzzy logic control is included for reducing chattering phenomena. Forward and inverse kinematics and full dynamic model of six degrees of freedom motion base driven by electromechanical actuators are briefly presented. As a initial control design, the step acceleration response was used to evaluate the performance of the controllers. The Lyapunov method is used to prove the stability of control strategies.

Nomenclature

$\mathbf{b}_i$	Position vector of the i-th base point
$\mathbf{p}_i$	Position vector of the i-th platform point (in moving platform frame)
$\Re$	Rotation matrix of moving platform frame relative to base platform frame, (ϕ, θ, ψ)
$(\mathbf{q}_\text{p})_i$	$\Re\mathbf{p}_i$
ω	Angular velocity of the moving platform
α	Angular acceleration of the moving platform
$(\tilde{\mathbf{q}}_\text{p})_i$	Skew-symmetric matrix associated to $(\mathbf{q}_\text{p})_i$
$\mathbf{s}_i$	Unit vector along the direction of the i-th leg
M	Mass of the moving platform (including payload)
$\mathbf{g}$	Acceleration due to gravity
$\mathbf{a}$	Acceleration of the moving platform centre of gravity
$\mathbf{R}$	Position vector of the centre of gravity
$\mathbf{I}_\text{p}$	Moment of inertia of the moving platform
F_i	Driving force generated by the ith electromechanical actuator
L_i	Lenght of the ith leg
τ_i	Driving torque generated by the ith electromechanical actuator
$\mathbf{J}_{l,\omega}$	Jacobian matrix relating ω and $\dot{L}_i$
$\mathbf{J}_{l,q}$	Jacobian matrix relating $\dot{L}_i$ and $\dot{\mathbf{q}}$

1 Introduction

In most motion control schemes concerning flight simulators, focus on the washout-filter, forward and inverse kinematics and effects of the motion-base dynamics are ignored or a linearized model of motion-base dynamics is used [1]. However, in a high performance flight simulator the nonlinear dynamics of the motion-base strongly influences system performance, therefore, robust nonlinear control laws are necessary.

Inverse dynamics control is an approach to nonlinear control design of which the central idea is to construct an inner loop control based on the motion base dynamic model which, in the ideal case, exactly linearizes the nonlinear system and an outer loop control to drive tracking errors to zero. Nonetheless, this technique is based on the assumption of exact cancellation of nonlinear terms, therefore, parametric uncertainty, unmodeled dynamics and external disturbances may deteriorate the controller performance. In addition, a high computational burden is paid by computing on-line the complete dynamic model of the motion-base [2]. Robustness can be regained by applying robust control tecniques in the outer loop control structure as is shown in 3.

In this context, this work presents the application of a sliding mode control strategy combined with fuzzy logic [4] applied in the outer loop of the feedback linearized system for robust acceleration tracking in the presence of parametric uncertainty and unmodeled dynamics, which is intentionally introduced in the process of approximating the dynamic model in order to simplify the implementation of this approach.

Forward and inverse kinematics and dynamic model of six degrees of freedom motion base are briefly presented. Then, electromechanical actuator dynamics are included in order to obtain a full dynamic model.

The sliding model control scheme on cartesian-space of Stewart platform is presented and saturation control is applied for reducing chattering phenomena. Subsequently, chattering reduction is achieved by fuzzifying the sliding surface. Tuning of fuzzy membership function is analysed.

Finally, standard methods to characterize the perfomance of a flight simulator motion base are presented and, as an initial control design the step acceleration response is used to evaluate the performance of the controller. The two controllers, saturation control and fuzzy sliding mode control are compared.

About organization of the text, this paper is structured as follows: in Section II, the forward and inverse kinematics and dynamic model of six degrees of freedom motion base are briefly presented. Then, electromechanical actuator dynamics are included in order to obtain a full dynamic model; in Section III, the control structure in cartesian coordinates is presented and the control strategies are applied. Then, dynamic model matrices that will be use in the controller are defined; in Section IV, three methods to evaluate the controllers's performance are defined; in Section V the results obtained from simulation are shown and discussed; and finally, in Section VI, conclusions of the present work are discussed.

2 Motion base kinematics and dynamics

In this study, a six-degree-of-freedom mechanism called the Stewart platform [5] is considered. The UPS (Universal-Prismatic-Spherical) Stewart platform is composed of a moving platform linked to a fixed base through six extensible legs. Each leg is composed of a prismatic joint (i.e an electromechanical actuator), one passive universal joint and one passive spherical joint making connection with the base and the moving platform (Fig. 1), respectively.

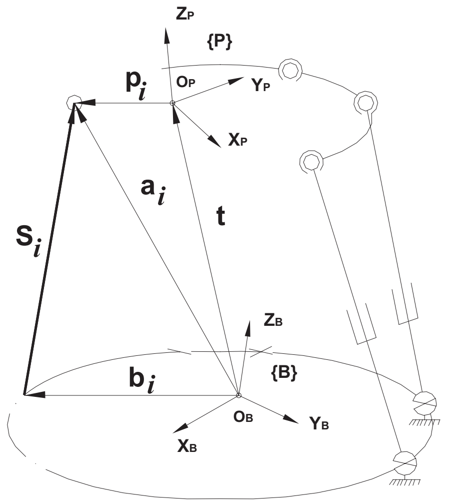

Fig. 1 UPS Stewart Platform

The Newton-Euler approach [6] was adopted to calculate the Stewart platform nonlinear dynamic model in cartesian coordinates. The cartesian space coordinates $\mathbf{q}$ are defined as

$$\mathbf{q} = \left[\mathbf{t}^{\mathrm{T}} \;\; \Theta^{\mathrm{T}}\right]^{\mathrm{T}},$$

where $\mathbf{t} = [x \;\; y \;\; z]^{\mathrm{T}}$, is the translation vector of the moving platform centroid and $\Theta = [\phi \;\; \theta \;\; \psi]^{\mathrm{T}}$ is the Euler angles vector definig its orientation. The leg vector with respect to the inertial base frame $\{\mathbf{B}\}$ (Fig. 1) can be denoted as

$$\mathbf{S}_i = \mathfrak{R}\mathbf{p}_i + \mathbf{t} - \mathbf{b}_i \tag{1}$$

Equation (1) represents the inverse kinematics problem in the sense one can compute the legs lengths, i.e., norms of $\mathbf{S}_i$, from the given position ($\mathbf{t}$) and orientation ($\mathfrak{R}$) of the platform.

The forward kinematics problem concerns with the determination of the position and orientation of the platform from the given actuator lengths. It is observed from Eq. (1) that the forward kinematics problem involves solving six simultaneous nonlinear equations for the values of the six unknowns variables representing the position and orientation of the platform. Consequently, iterative numerical methods are employed to solve the set of nonlinear equations [7].

Kinematic analysis of the legs can be derived from Eq. (1). Dynamic analysis of the legs can be derived taking force and moment balance of each legs (Fig. 2) and combining both equations. Thus, one gets

$$(\mathbf{F}_\mathrm{s})_i = \mathbf{Q}_i\ddot{\mathbf{t}} - \mathbf{Q}_i(\tilde{\mathbf{q}}_\mathrm{p})_i\alpha + \mathbf{V}_i - F_i\mathbf{s}_i \qquad (2)$$

where $(\mathbf{F}_\mathrm{s})_i$ is the force applied to the leg i by the upper platform and $\mathbf{Q}_i$ depends of the ith leg inertial properties and $\mathbf{V}_i$ depends of the ith leg dynamic properties.

Similarly, force and moment balance on the moving platform (Fig. 2) can be written as (with no external forces):

$$M\mathbf{a} = M\mathbf{g} - \sum_{i=1}^{6}(\mathbf{F}_\mathrm{s})_i \qquad (3)$$

and

$$M\mathbf{R}\times\mathbf{g} - \sum_{i=1}^{6}\left[(\mathbf{q}_\mathrm{p})_i \times (\mathbf{F}_s)_i\right] + \sum_{i=1}^{6}\mathbf{f}_i = \mathbf{I}_\mathrm{p}\alpha + \omega\times\mathbf{I}_\mathrm{p}\omega + M\mathbf{R}\times\mathbf{a}; \qquad (4)$$

where $\mathbf{f}_i$ is the moment of viscous friction at the i-th spherical joint.

Substituting the expression for $(\mathbf{F}_\mathrm{s})_i$ from Eq. (2) into Eqs. (3) and (4) and combining them, one gets

$$\mathbf{M}_\mathrm{t}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_\mathrm{t}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{B}_\mathrm{t}(\dot{\mathbf{q}}) + \mathbf{G}_\mathrm{t}(\mathbf{q}) = \mathbf{J}_{l\omega}\mathbf{F} \quad (5)$$

where:

$$\mathbf{M}_\mathrm{t} = \mathbf{M}_\mathbf{p} + \mathbf{M}_\mathbf{a}; \quad \mathbf{C}_\mathrm{t} = \mathbf{C}_\mathbf{p} + \mathbf{C}_\mathbf{a}; \quad \mathbf{G}_\mathrm{t} = \mathbf{G}_\mathbf{p} + \mathbf{G}_\mathbf{a};$$

and

$$\mathbf{F} = \begin{bmatrix} F_1 & F_2 & F_3 & F_4 & F_5 & F_6 \end{bmatrix}^\mathrm{T}.$$

The detailed elements of the above matrices are given in the Appendix. Information about derivation of the Stewart platform's dynamic model is not detalied here since it is not the scope of this paper.

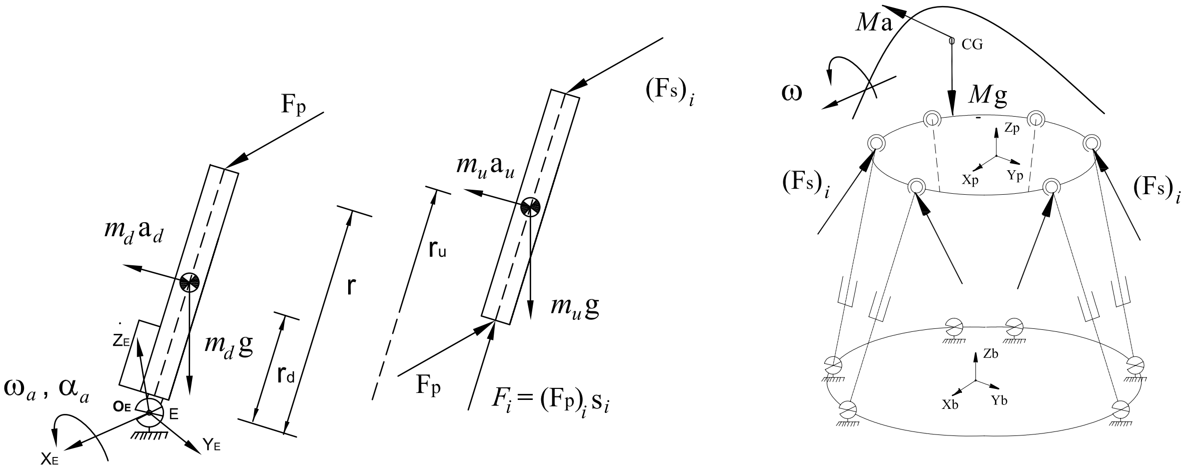

Fig. 2 Vector Analysis - Motion Base

2.1 Inclusion of actuator dynamics

The equation of motion of the electromechanical actuator (Fig. 3) can be written in matrix form:

$$\mathbf{F} = \mathbf{K}_a\mathbf{T}_m - \mathbf{D}_a\ddot{\mathbf{L}} - \mathbf{B}_a\dot{\mathbf{L}}; \tag{6}$$

where

$$\begin{array}{l} \mathbf{T}_m = \left[\begin{array}{cccccc} \tau_1 & \tau_2 & \tau_3 & \tau_4 & \tau_5 & \tau_6 \end{array}\right]^{\mathrm{T}}. \\ \mathbf{L} = \left[\begin{array}{cccccc} L_1 & L_2 & L_3 & L_4 & L_5 & L_6 \end{array}\right]^{\mathrm{T}}; \end{array}$$

and where $\mathbf{D}_a$ are the actuator inertia matrix, $\mathbf{B}_a$ is the actuator viscous damping coefficient matrix, and $\mathbf{K}_a$ is the actuator gain matrix and are detailed in the Appendix.

The relationship between the cartesian coordinates and joint coordinates can be written as

$$\begin{array}{l} \dot{\mathbf{L}} = \mathbf{J}_{l,q}\dot{\mathbf{q}} \\ \ddot{\mathbf{L}} = \mathbf{J}_{l,q}\ddot{\mathbf{q}} + \dot{\mathbf{J}}_{l,q}\dot{\mathbf{q}} \end{array} \tag{7}$$

Substituting Eq. (7) into Eq. (6), one gets

$$\mathbf{F} = \mathbf{K}_a\mathbf{T}_m - \mathbf{D}_a\mathbf{J}_{l,q}\ddot{\mathbf{q}} - \mathbf{D}_a\dot{\mathbf{J}}_{l,q} - \mathbf{B}_a\mathbf{J}_{l,q}\dot{\mathbf{q}} \tag{8}$$

And substituting Eq. (8) into Eq. (5), the full dynamic model in cartesian coordinates results as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{u}_T \tag{9}$$

where

$$\begin{array}{l} \mathbf{N} = \mathbf{C} + \mathbf{E} + \mathbf{G} \\ \mathbf{M} = \mathbf{K}_a^{-1}\left[\mathbf{J}_{l,\omega}^{-\mathrm{T}}\mathbf{M}_{\mathrm{t}} + \mathbf{D}_a\mathbf{J}_{l,q}\right] \\ \mathbf{C} = \mathbf{K}_a^{-1}\left[\mathbf{J}_{l,\omega}^{-\mathrm{T}}\mathbf{C}_{\mathrm{t}} + \mathbf{D}_a\dot{\mathbf{J}}_{l,q} + \mathbf{B}_a\mathbf{J}_{l,q}\dot{\mathbf{q}}\right] \\ \mathbf{E} = \mathbf{K}_a^{-1}\mathbf{J}_{l,\omega}^{-\mathrm{T}}\mathbf{B}_{\mathrm{t}} \\ \mathbf{G} = \mathbf{K}_a^{-1}\mathbf{J}_{l,\omega}^{-\mathrm{T}}\mathbf{G}_{\mathrm{t}} \end{array}$$

and $\mathbf{u}_T = \mathbf{T}_m$, which is proportional to vector voltage or current (not considering the actuator electrical dynamics) driving the servo-drive of the electromechanical actuator. The electrical dynamics can be ommitted considering that the closed loop bandwidth of the servo-drive is much higher than that of the motion base.

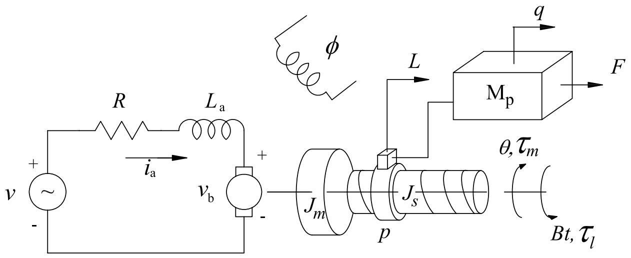

Fig. 3 Electromechanical Actuator

3 Motion base controller design

A flight simulator control system stems from two frameworks. The first one is based on the legs lenghts tracking control and is called joint space control. On the contrary, the second one is based on the position and orientation tracking control and is called cartesian space control.

Joint space control does not seem suitable in inverse dynamic control due to the fact that joint space dynamic equation are more complicated compared with the cartesian ones in Eq. (9), moreover, the terms of the joint space dynamic matrices will still depend on the cartesian coordinates.

Cartesian space control was adopted in this study as shown in Fig. 4. The pilot responds to simulator cues and tracking or disturbance tasks by driving the aircraft control surfaces, then aircraft's response is calculated through of an aircraft dynamic model. Because of the limited motion envelopes of the motion base, filtering (by washout filter [8]) is required between the calculated aircraft trajectories and the commanded motion base trajectories. Then, the controller attempts to null the cartesian coordinate error by commanding a torque signal (voltage or current) to the servo-drive of the electromechanical actuator.

Thus, the force driving the motion base is governed by the equation of motion of the electromechanical actuator in Eq. (8).

Cartesian space control needs information of a 6 degrees of freedom sensor to measure the position and orientation of the platform. However, when only legs lenghts measurements are avail-

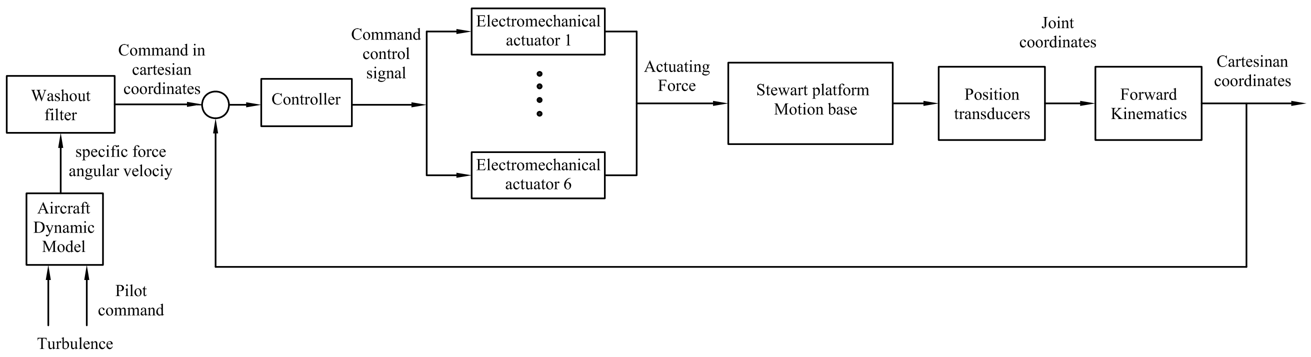

Fig. 4 Cartesian Space Control Framework

able, the forward kinematic problem must be resolved.

3.1 Imperfect compensation of the inverse dynamics control

As it is well knwon, inverse dynamics control is an approach to nonlinear control design of which the central idea is to construct an inner loop control which, in the ideal case, exactly linearizes the nonlinear system and an outer loop control to drive tracking errors to zero. However, from an implementation viewpoint, compensantion may be impefect, both for model uncertainties and for the approximation made of the dynamic model. Therefore, the law control can be expressed by (Fig. 5)

$$\mathbf{u}_T = \widehat{\mathbf{M}}(\mathbf{q})\mathbf{v} + \widehat{\mathbf{N}}(\mathbf{q}); \tag{10}$$

where $\widehat{\mathbf{M}}$, $\widehat{\mathbf{N}}$ represent simplified versions of $\mathbf{M}$, $\mathbf{N}$, and

$$\mathbf{v} = \ddot{\mathbf{q}}_{\mathrm{d}} + \mathbf{K}_{\mathrm{d}}\dot{\mathbf{e}} + \mathbf{K}_{\mathrm{p}}\mathbf{e} + \mathbf{u}; \tag{11}$$

and

$$\begin{aligned} \mathbf{K}_{\mathrm{p}} &= \mathrm{diag}\left\{\omega_1^2, \ldots, \omega_6^2\right\} \\ \mathbf{K}_{\mathrm{d}} &= \mathrm{diag}\left\{2\varsigma_1\omega_1, \ldots, 2\varsigma_6\omega_6\right\} \end{aligned} \tag{12}$$

and where the tracking error is defined as

$$\mathbf{e} = \mathbf{q}_{\mathrm{d}} - \mathbf{q}; \tag{13}$$

where $\mathbf{q}_{\mathrm{d}}$ is the desired cartesian space coordinates.

The term $\mathbf{u}$ in Eq. (11) is included to overcome imperfect compensation effects. Now, substituting Eq. (10) into Eq. (2) and simplifying it, one gets

$$\ddot{\mathbf{e}} + \mathbf{K}_{\mathrm{d}}\dot{\mathbf{e}} + \mathbf{K}_{\mathrm{p}}\mathbf{e} = \mathbf{w} - \mathbf{u}; \tag{14}$$

where

$$\begin{aligned} &\mathbf{w} = (\mathbf{I} - \mathbf{M}^{-1}\widehat{\mathbf{M}})\mathbf{v} - \mathbf{M}^{-1}\Delta\mathbf{N} \\ &\Delta\mathbf{N} = \mathbf{N} - \widehat{\mathbf{N}} \end{aligned} \tag{15}$$

In state space representation the system described by Eq. (14) becomes

$$\begin{aligned} &\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}(\mathbf{w} - \mathbf{u}) \\ &\mathbf{y} = \mathbf{x} \end{aligned} \tag{16}$$

where

$$\mathbf{A} = (\mathbf{H} - \mathbf{B}\mathbf{K}) \quad , \quad \mathbf{K} = \begin{bmatrix} \mathbf{K}_{\mathrm{p}} & \mathbf{K}_{\mathrm{d}} \end{bmatrix} \tag{17}$$

and

$$\mathbf{H} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} \mathbf{e} \\ \dot{\mathbf{e}} \end{bmatrix} \tag{18}$$

In this context, the term $\mathbf{u}$ must be designed to stabilize the nonlinear time-varying error system defined by Eq. (16) in the presence of the incertainty $\mathbf{w}$. In the next sections, two strategies will be designed in order to find this term.

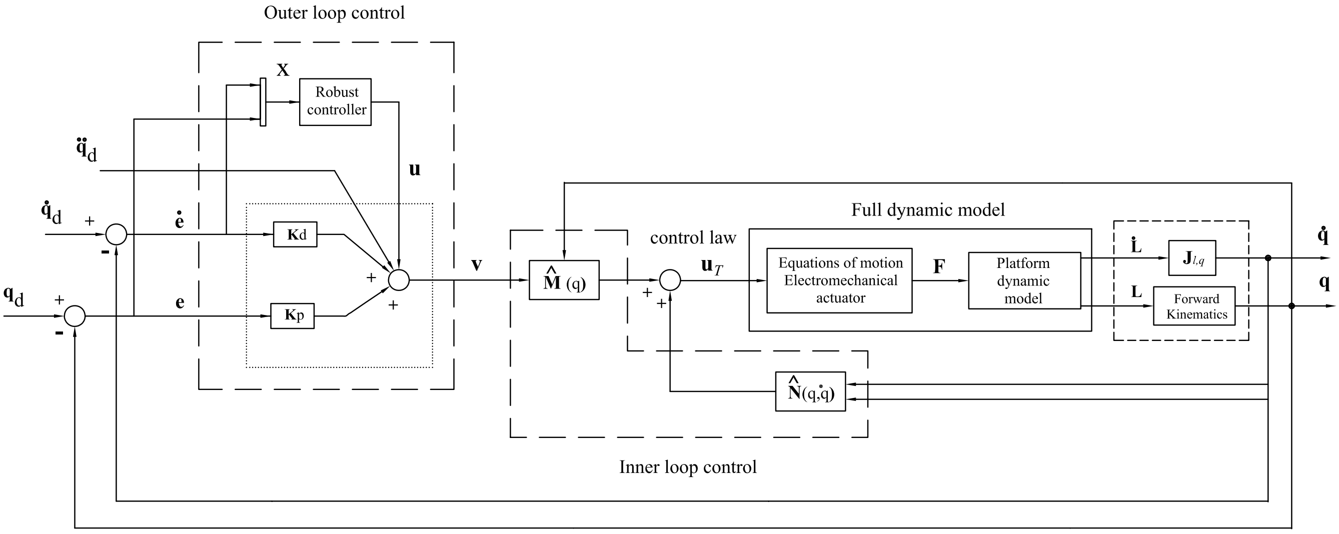

Fig. 5 Inverse Dynamic Control, Imperfect Compensantion

3.2 Robust outer loop design by Lyapunov's Second Method

To determine $\mathbf{u}$ one can consider the following positive definite quadratic form as Lyapunov function candidate:

$$\mathbf{V} = \mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{x} > 0 \qquad \forall \mathbf{x} \neq 0 \tag{19}$$

The time derivative of $\mathbf{V}$ is:

$$\frac{d\mathbf{V}}{dt} = -\mathbf{x}^{\mathrm{T}}\mathbf{T}\mathbf{x} + 2\mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{B}(\mathbf{w}-\mathbf{u}) \tag{20}$$

where

$$\mathbf{A}^{\mathrm{T}}\mathbf{P} + \mathbf{P}\mathbf{A} = -\mathbf{T} \tag{21}$$

for any symmetric positive definite matrix $\mathbf{T}$ and considering that $\mathbf{A}$ has eigenvalues with all real parts negative.

If $d\mathbf{V}/dt$ is negative then the system represented by the Eq. (16) converge to zero. Thus, the term $\mathbf{u}$ is chosen to render the second term in Eq. (20) less than or equal to zero, and it is given as [9]:

$$\mathbf{u} = \frac{\rho}{\|\mathbf{z}\|}\mathbf{z} \qquad \rho > 0 \quad ; \tag{22}$$

where

$$\mathbf{z} = \mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}; \tag{23}$$

and where

$$\rho \geq \frac{1}{1-\lambda}\left(\lambda Q_M + \lambda\,\|\mathbf{K}\|\,\|\mathbf{x}\| + B_M\Phi\right) \tag{24}$$

The scalar values Q_M, λ and Φ represent a worst case bound on the incertanty $\mathbf{w}$ in Eq. (15) and they are given as

$$\begin{gathered}\sup_{t\geq 0}\|\ddot{\mathbf{q}}_{\mathrm{d}}\| < Q_M < \infty \quad \forall \ddot{\mathbf{q}}_{\mathrm{d}} \\ \left\|\mathbf{I}-\mathbf{M}^{-1}\widehat{\mathbf{M}}\right\| \leq \lambda \leq 1 \quad \forall \mathbf{q} \\ \|\Delta\mathbf{N}\| \leq \Phi \leq \infty \quad \forall \mathbf{q}, \dot{\mathbf{q}}\end{gathered} \tag{25}$$

3.2.1 Sliding Surfaces

The robust control law in Eq. (22) guarantees existence of individual sliding modes in the sliding subspace $\mathbf{z} = 0$. A characterization of an error trajectory is given by considering the sliding subspace as:

$$\mathbf{z} = \mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x} = \begin{bmatrix} ae_1 & + & b\dot{e}_1 \\ \vdots & \vdots & \vdots \\ ae_6 & + & b\dot{e}_6 \end{bmatrix} = \mathbf{0}; \tag{26}$$

so, for all $e_i(0)$ and $\dot{e}_i(0) \notin$ null space of the sliding surfaces $ae_i + b\dot{e}_i = 0$, the error trajectory is attracted on the sliding surfaces and tends towards the origin.

The control law defined in Eq. (22) presents a phenomen kwown as chattering, which is often undesirable since the high frequency oscillations in the control can excite unmodeled dynamic effects.

Elimination of chattering can be achieved by inroducing a boudary layer that contains the sliding surface. The introduction of the boundary layer is equivalent to the replacement of the control law in Eq. (22) by a saturation function as

$$\mathbf{u} = \begin{cases} \frac{\rho}{\|\mathbf{z}\|}\mathbf{z} & \forall \|\mathbf{z}\| \geq \varepsilon \\ \\ \frac{\rho}{\varepsilon}\mathbf{z} & \forall \|\mathbf{z}\| < \varepsilon \end{cases} \tag{27}$$

Although the control law in Eq. (27) does not guarantee error convergence to zero, it ensures bounded-norm error given by ε.

The idea of the boundary layer is illustrated on Fig. 6, where the layer thickness, ε_t, the slope of the sliding line, λ_s, and the layer width, ε_w, are given as:

$$\varepsilon_t = \varepsilon/b; \quad \lambda_s = -a/b; \quad \varepsilon_w = \varepsilon_t/\lambda_s; \tag{28}$$

where a and b are defined in Eq. (26). The slope of the sliding line, λ_s, depends principally on matrix $\mathbf{P}$ in Eq. (21).

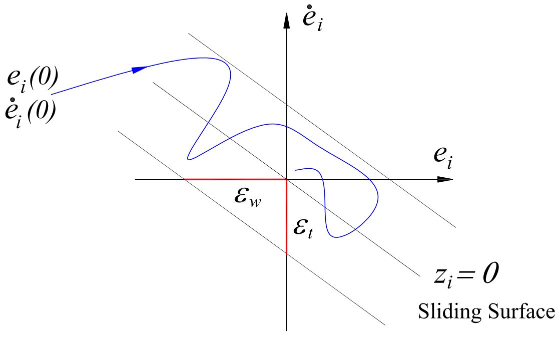

Fig. 6 Error Trajectory with Robust Control and Chattering Elimination

3.2.2 *Fuzzy Sliding Surfaces*

The idea behind the fuzzifying the sliding surface is that fuzzy logic control (FLC) is essentially a discontinuos switching control law like Eq. (22), i.e, can be observed a similarity between the discontinuos law and the rule-base of the FLC. For example, if the error is negative, push in the positive direction and conversely, in order to attract and keep the error trajectory on the sliding surface.

So, the chattering phenomenon brought by the conventional law in Eq. (22) can be eliminate by replacing the control law in Eq. (27) by an FLC structure.

For each degree-of-freedom of the Stewart platform a FLC is designed using the fuzzy phase plane which is defined by the fuzzy values of e_i and $\dot{e}_i$ (similar to Fig. 6).

The fuzzy rules produce a fuzzy control signal u_i employing the fuzzy values of e_i and $\dot{e}_i$ and are given as in Table 1 where P, N and Z are linguistic values standing for positive, negative and zero, respectively.

Table 1 Fuzzy Rules

$\dot{e}_i$ \ e_i	**N**	**Z**	**P**
P	Z	N	N
Z	P	Z	N
N	P	P	Z

In this paper, the triangular membership function, the minimum-maximum reasoning method, and the center-of-gravity (CoG) defuzzification method are used. The crisp output is obtained by calculating the CoG of the output fuzzy set.

The fuzzy values of the error, error derivative and control signal are defined on the normalized universes of discourse $X_e = [-a_e^*, a_e^*]$, $X_d = [-a_d^*, a_d^*]$ and $X_u = [-a_u^*, a_u^*]$, respectively, by the membership function in Fig. 7.

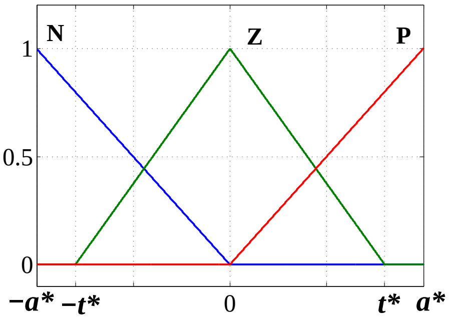

Fig. 7 Fuzzy Membership Functions

3.2.3 Tuning of Fuzzy Membership Functions

Tuning of fuzzy membership function consist in relating the normalized membership parameters t^* and a^* (Fig. 7), and boundary layer parameters defined in Eq. (28).

From Eq. (26) the equivalent normalized fuzzy sliding surfaces can be written as (without subscripting them by the degree of freedom index i):

$$\lambda^* e^* + \dot{e}^* = 0; \tag{29}$$

where e^* and $\dot{e}^*$ are the normalized values of the error and error derivative, respectively, and are given as $e^* = k_e e$ and $\dot{e}^* = k_d \dot{e}$, where k_e and k_d are the scaling factors.

Normalized fuzzy boundary layer parameters are defined as [4]:

$$\lambda^* = \frac{a_d^*}{a_e^*}; \tag{30}$$

and

$$\varepsilon_t^* = \lambda^* t_e^* + t_d^*; \tag{31}$$

where $t^* = t_e^*$ for the error membership functions and $t^* = t_d^*$ for the error derivative membership functions as illustrade in Fig. 7.

Now, the actual parameter $\tilde{\lambda}_s = \dot{e}/e$ is calculate by normalization of $\dot{e}$ and e, and is given as:

$$\tilde{\lambda}_s = \lambda^* \frac{k_e}{k_d} \tag{32}$$

And, the actual value $\tilde{\varepsilon}_t$ is calculated from Eq. (31) by denormalization of the parameters t_e^* and t_d^*:

$$\tilde{\varepsilon}_t = \frac{\varepsilon_t^*}{k_d}. \tag{33}$$

As can be see from Eq. (31) and Eq. (33), the layer thickness $\tilde{\varepsilon}_t$ can be controlled by modifying the dead zone parameters t_e^* and t_d^* of the membership function of the inputs to the FLC, or by modifying the scaling factor k_d.

With relation to the scaling factor of the control variable, k_u, and considering Eq. (27), it is calculated from the maximum value of ρ in Eq. (24) in a such way that:

$$k_u = \frac{\rho_{max}}{\rho_{max}^*} \quad ; \quad \rho_{max} \geq \rho; \tag{34}$$

and ρ_{max}^* is the maximal centroid of fuzzy sets **N** and **P** of the normalized control variable u^*.
So, the control variable u is calculated as:

$$u = u^* k_u \tag{35}$$

Therefore, control variable **u** defined in Eq. (11) is given as:

$$\mathbf{u} = \begin{bmatrix} u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \end{bmatrix}^{\mathrm{T}}.$$

3.3 Characteristics of the dynamic equations

Parallel manipulators motion bases have some drawback of relatively small workspace comparing to serial manipulators. In flight simulators motion bases, this is due mainly to the physical restriction in terms of position, velocity and acceleracion of the actuators, e.g, for low frequencies motion, the velocity and position constraints limit the maximal attainable acceleration. Moreover, the high pass wash-out filter characteristics keeps the motion system not to far away from the neutral positiion to prevent the actuators from running out of stroke. Thus, the coefficient matrices of Eq. (9) can be approximated to constant ones without introducing large modelling errors. Based on these constant matrices, calculation of the approximated inverse dynamics becomes much simpler reducing computation time significantly.

In this context, matrices $\widehat{\mathbf{M}}(\mathbf{q})$ and $\widehat{\mathbf{N}}(\mathbf{q})$ considered in the law control in Eq. (10), are defined at the neutral position as:

$$\begin{aligned}\widehat{\mathbf{M}}(\mathbf{q}_n) &= \mathbf{K}_a^{-1}\mathbf{J}_{l,\omega}^{-\mathrm{T}}(\mathbf{q}_n)\mathbf{M}_{\mathrm{p}}(\mathbf{q}_n)\\ \widehat{\mathbf{N}}(\mathbf{q}_n) = \widehat{\mathbf{G}}(\mathbf{q}_n) &= \mathbf{K}_a^{-1}\mathbf{J}_{l,\omega}^{-\mathrm{T}}(\mathbf{q}_n)\mathbf{G}_{\mathrm{p}}(\mathbf{q}_n)\end{aligned} \tag{36}$$

where $\mathbf{q}_n$ represents a neutral position and it was chosen to be at half stroke of all the actuators. Coriolis, centrifugal and leg effects are not considered.

4 Controller's performance evaluation

At present, only one standard method to characterize the performance of a motion system is known to exist [2]. This is described in the AGARD advisory Report [10].

As a initial control design, two methods defined in the report should be considered: Describing function as a frequency domain evaluation and the step acceleration response (dynamic threshold) in time domain. For each degree of freedom, six describing functions can be calculated. The primary describing function is the comparison of the response of the motion-base in the driven degree of freedom to the excitation signal and the other five describing functions (crosstalk describing functions) compare pure parasitic motion (motion in other than the degree of freedom excited) to the excitation signal [11]. Furthermore, to evaluate the system in its normal operating mode, some standard maneuvers should be evaluated as well.

In this study were only considered the step acceleration response, and the others methods will be evaluted in future works.

5 Numerical results and discussions

The performance of the proposed controllers is verified by numerical simulations, and results are presented only for sway degree of freedom (x direction). Similar results were presented in the others degree of freedom. In Table 2 are presented some motion base characteristic where M_{act} is the actuator mass and r_p and r_b are the moving and base platform radius, respectively.

Table 2 Motion Base Characteristics

Parameter	Value
M	2500 Kg
M_{act}	100 Kg
r_p	1.6 m
r_b	1.65 m
$\mathbf{q}_n$	2.154 m in z

Ref. 2 state the frequency, ω_i in Eq. (12) should not exceed human sensory thresholds and that it should ideally be sufficiently smooth and only requires limited bandwidth (well below 1 Hz). Therefore, a bandwidth, ω_i, of 1 Hz was chosen for a damping coefficient, ζ_i = 0.7. This values were chosen for both controllers.

In order to evaluate really the control strategies, the acceleration step amplitude was choosen to keep the motion base approximately 70% of the system limits in position, velocity and acceleration.

Acceleration response is shown in Fig. 8, where acceleration amplitude has been scaled to a desired final value of 1. Motor torques and angular velocities of the electromechanical actuators (for actuators 1, 2 and 3) are shown in Fig. 9

As can be seen from Fig. 8, both controllers presented similar responses. However, a smoother response was achieved by the conventional sliding mode controler. Smoother response can be achieved by increasing the levels of fuzzy partitioning of the input and output of the FLC, since it produces a more linear control [4].

From Figs. 8 and 9 can be observed chattering phenomen was eliminated and an smoothed control and torque was generated.

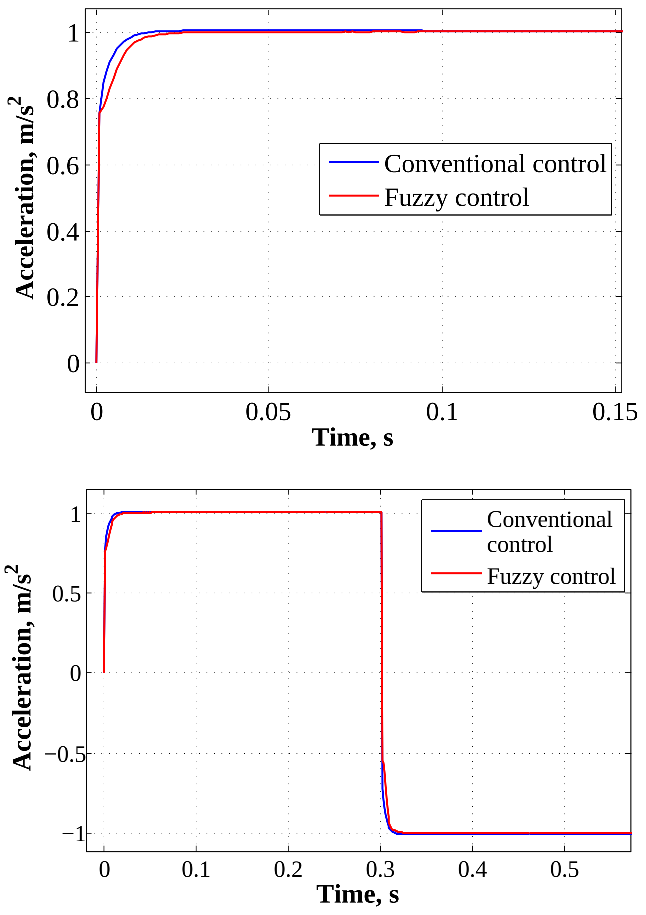

Fig. 8 Step Acceleration Response

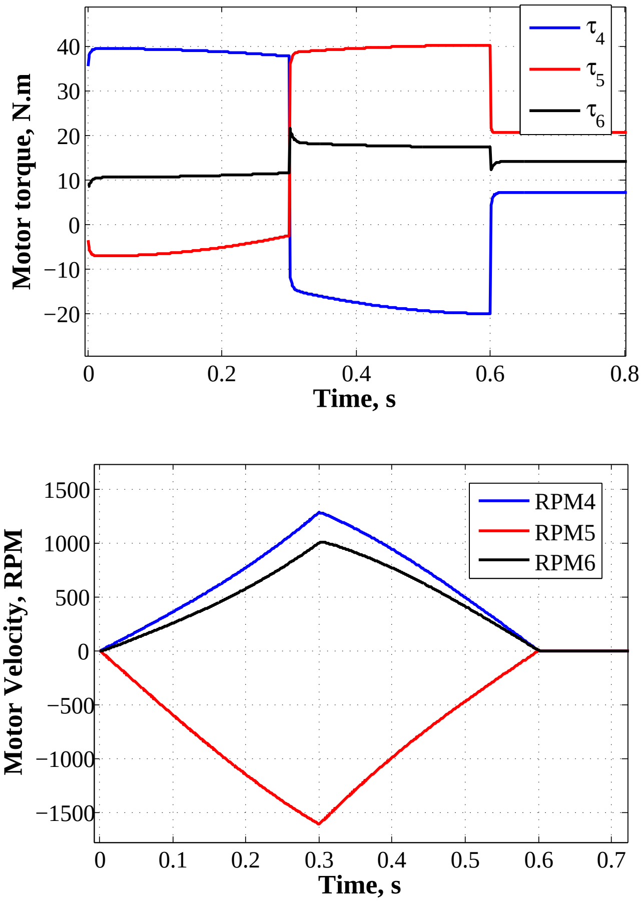

Fig. 9 Motor Torques and Velocity

6 Conclusions

In this paper, a control approach for the motion control of a flight simulator motion base was presented. The controller was implemented in the outerloop of the inverse dynamic control scheme in order to counteract imperfect compensation. Imperfect compensation was included intentionally by defining the motion base nominal dynamic matrices as constants. The approachs was designed via Lyapunov stability theory. Chattering reduction was achieved by saturarion control and by fuzzifying the sliding surface. Controller' performance evaluation was carried out through step acceleration input. The controller presented robustness to bounded modelling error and chattering phenomen was eliminated.

Appendix

The matrices of the motion base dynamic model are given by:

$$\mathbf{M}_{\mathrm{p}} = \begin{bmatrix} M\mathbf{I} & -M\widetilde{\mathbf{R}}\Re_{\omega} \\ M\widetilde{\mathbf{R}} & \left(\mathbf{I}_{\mathrm{p}} - M\widetilde{\mathbf{R}}\widetilde{\mathbf{R}}\right)\Re_{\omega} \end{bmatrix};$$

$$\mathbf{M}_{\mathrm{a}} = \begin{bmatrix} \displaystyle\sum_{i=1}^{6}\mathbf{Q}_i & -\left(\displaystyle\sum_{i=1}^{6}\mathbf{Q}_i(\widetilde{\mathbf{q}}_{\mathrm{p}})_i\right)\Re_{\omega} \\ \displaystyle\sum_{i=1}^{6}(\widetilde{\mathbf{q}}_{\mathrm{p}})_i\mathbf{Q}_i & -\left(\displaystyle\sum_{i=1}^{6}(\widetilde{\mathbf{q}}_{\mathrm{p}})_i\mathbf{Q}_i(\widetilde{\mathbf{q}}_{\mathrm{p}})_i\right)\Re_{\omega} \end{bmatrix}$$

$$\mathbf{C}_{\mathrm{p}} = \begin{bmatrix} M\omega \times (\omega \times \mathbf{R}) \\ \omega \times \mathbf{I}_{\mathrm{p}}\omega + M\mathbf{R} \times (\omega \cdot \mathbf{R})\omega \end{bmatrix} - \begin{bmatrix} M\widetilde{\mathbf{R}} \\ M\widetilde{\mathbf{R}}\widetilde{\mathbf{R}} - \mathbf{I}_{\mathrm{p}} \end{bmatrix} \dot{\Re}_{\omega}\dot{\Theta}$$

$$\mathbf{C}_{\mathrm{a}} = \begin{bmatrix} \sum_{i=1}^{6} (\mathbf{V}_c)_i \\ \sum_{i=1}^{6} ((\mathbf{q}_{\mathrm{p}})_i \times (\mathbf{V}_c)_i) \end{bmatrix} - \begin{bmatrix} \sum_{i=1}^{6} \mathbf{Q}_i(\widetilde{\mathbf{q}}_{\mathrm{p}})_i \\ \sum_{i=1}^{6} (\widetilde{\mathbf{q}}_{\mathrm{p}})_i \mathbf{Q}_i(\widetilde{\mathbf{q}}_{\mathrm{p}})_i \end{bmatrix} \dot{\Re}_{\omega}\dot{\Theta}$$

$$\mathbf{G}_{\mathrm{p}} = -\begin{bmatrix} M\mathbf{g} \\ M\mathbf{R} \times \mathbf{g} \end{bmatrix}; \quad \mathbf{G}_{\mathrm{a}} = \begin{bmatrix} \sum_{i=1}^{6} (\mathbf{V}_g)_i \\ \sum_{i=1}^{6} ((\mathbf{q}_{\mathrm{p}})_i \times (\mathbf{V}_g)_i) \end{bmatrix}$$

$$\mathbf{B}_{\mathrm{t}} = \begin{bmatrix} \sum_{i=1}^{6} (\mathbf{V}_f)_i \\ \sum_{i=1}^{6} ((\mathbf{q}_{\mathrm{p}})_i \times (\mathbf{V}_g)_i) - \mathbf{f}_i \end{bmatrix}$$

where $(\mathbf{V}_c)_i$ and $(\mathbf{V}_g)_i$ depend of the ith leg dynamic properties, and $(\mathbf{V}_f)_i$ is the viscous friction force vector at the ith leg joints.

The motion base Jacobian is given as

$$\mathbf{J}_{l,\omega} = \begin{bmatrix} \mathbf{s}_1^{\mathrm{T}} & (\mathbf{q}_{\mathrm{p}})_1^{\mathrm{T}} \\ \mathbf{s}_2^{\mathrm{T}} & (\mathbf{q}_{\mathrm{p}})_2^{\mathrm{T}} \\ \mathbf{s}_3^{\mathrm{T}} & (\mathbf{q}_{\mathrm{p}})_3^{\mathrm{T}} \\ \mathbf{s}_4^{\mathrm{T}} & (\mathbf{q}_{\mathrm{p}})_4^{\mathrm{T}} \\ \mathbf{s}_5^{\mathrm{T}} & (\mathbf{q}_{\mathrm{p}})_5^{\mathrm{T}} \\ \mathbf{s}_6^{\mathrm{T}} & (\mathbf{q}_{\mathrm{p}})_6^{\mathrm{T}} \end{bmatrix}$$

And the Jacobian that maps the cartesian coordinates into joint coordinates is given as

$$\mathbf{J}_{l,q} = \mathbf{J}_{l,\omega}\mathbf{J}_{\omega,q}$$

where

$$\mathbf{J}_{\omega,q} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \Re_{\omega} \end{bmatrix}$$

The matrices of the equation of motion of the electromechanical actuator are given as

$$\mathbf{K}_a = \begin{bmatrix} K_a & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & K_a \end{bmatrix}; \quad \mathbf{D}_a = \begin{bmatrix} D_a & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & D_a \end{bmatrix};$$

$$\mathbf{B}_a = \begin{bmatrix} B_a & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & B_a \end{bmatrix}$$

where

$$K_a = \tfrac{2\pi\eta}{p}; \; M_a = \tfrac{J_t 4\pi^2\eta}{p^2}; \; B_a = \tfrac{B_t 4\pi^2\eta}{p^2}$$

and where p and η are the lead and efficiency of the ballscrew, and J_t and B_t are the moment of inertia and viscous damping of the rotor and ballscrew, respectively.

Acknowledgments

The author gratefully acknowledge the financial support provided by FAPESP (Sao Paulo State Foundation for Research Support - Brazil) through contract number 2005/25486-6.

Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS2010 proceedings or as individual off-prints from the proceedings.

References

[1] M. Idan and D. Sahar, "Robust controller for a dynamic six degree of freedom flight simulator," in *AIAA Flight Simulation Technologies Conference*, (San Diego, CA), pp. 53–64, AIAA, 1996.

[2] S. Koekebakker, *Model based control of a flight simulator motion system*. PhD thesis, Delft University of Technology, Netherlands, 2001.

[3] M. Becerra-Vargas, E. Belo, and P. Grant, "Robust control of a flight simulator motion base," in *AIAA Modeling and Simulation Technologies Conference*, (Chicago, IL), pp. 1–13, AIAA, 2009.

[4] R. Yager and D. Filev, *Essentials of fuzzy modeling and control*. New York: John Wiley & Sons, Inc., 1 ed., 1994.

[5] D. Stewart, "A platform with 6 degrees of freedom," *Proceedings of the institution of mechanical engineers 1965-66*, vol. 180, no. 15, pp. 371–386, 1965.

[6] B. Dasgupta and T. Mruthyunjaya, "Closed form dynamic equations of the general Stewart platform through the Newton-Euler approach," *Mechanism and Machine Theory*, vol. 33, no. 7, pp. 993–1012, 1998.

[7] C. Nguyen, S. Antrazi, Z. Zhou, and C. Campbell, "Adaptive control of a Stewart platform-based manipulator," *Journal of Robotic systems*, vol. 10, no. 5, pp. 657–687, 1993.

[8] M. Nahon and L. D. Reid, "Simulator motion drive algorithms - a designer's perspective," *Journal of Guidance, Control, and Dynamics*, vol. 13, no. 2, pp. 702–709, 1990.

[9] L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators*. London: Springer, 6 ed., 2005.

[10] D. Lean, "Dynamic characteristics of flight simulation motion systems," tech. rep., France, 1979. AGARD Advisory Report No. 144.

[11] P. Grant, "Motion characteristics of the UTIAS flight research simulator motion-base," tech. rep., University of Toronto - UTIAS, Toronto-Canada, 1986. Technical Note No. 261.