# ATTITUDE CONTROLLER DESIGN OF THE BRAZILIAN SATELLITE LAUNCHER VIA HYBRID NEURAL-GENETIC APPROACH

**Paulo R. Silva**
**Ivanildo S. Abreu**
**Henrique M. C. do Amaral**
*paulosilva14@aluno.uema.br*
*ivanildoabreu@professor.uema.br*
*henriqueamaral@professor.uema.br*
*Dept. of Computer Engineering, State University of Maranhão, 65055-310, Maranhão, Brazil*

**Abstract.** This work proposes the design of an attitude controller for the Brazilian launching vehicle via mode-selection using a hybrid neural-genetic method. Given the high complexity of the rocket dynamic equations, the model was linearised and minimized with a model order reduction technique, in particular mode-selection. The hybrid approach performs the weighting matrices search of the linear quadratic (LQ) method and the solution of the Algebraic Riccati Equation (ARE) that leads to the attitude controller gains. The performance analysis of the reduced order model and the designed controller was performed in the frequency and time domain, while the hybrid neural-genetic approach was evaluated through fitness function and energy and infinity norms, respectively. The proposed controller reached the time domain specifications, i.e. rise time, settling time and overshoot for the maximum dynamic pressure instant. The results suggest that the hybrid approach could speed up the attitude controller design process of Brazilian launchers, reducing costs and re-design possibility.

**Keywords:** Genetic Algorithm, Neural Networks, Reduced Order Model, Attitude Control

# 1 Introduction

Aerospace engineering has carried notable improvements in navigation, communications, space and earth observation, bringing progress to human-being. In order to travel safely through space, the non-linear space vehicles demand superb navigation and guidance modules with embedded digital controllers to command attitude angles and velocities, Markley and Crassidis [1], Tiwari et al. [2, 3], Han et al. [4].

The Brazilian Satellite Launcher (*Veículo Lançador de Satélites* - in portuguese), defined by Palmerio [5] and illustrated in Fig. 1, is designed by the Instituto de Aeronáutica e Espaço (IAE) and can place satellites of $115\ kg$ in $700\ km$ height circular orbits, with maximum $25^o$ of inclination. For this task the launcher keeps four solid propulsion stages with Thrust Vector Control (TVC) for attitude tracking in the first three stages.

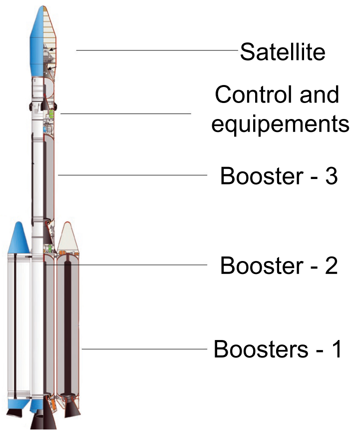



Figure 1. *VLS* architecture.

Due to instability, non-linearities and load specifications, the design of attitude controllers has been stated as a challenge task, Daitx and Kienitz [6]. For linear time invariant systems, the controller design adopting optimal control strategies have to minimize a quadratic performance index to satisfy desired specifications, Das et al. [7]. This performance index is directly associated with the control outcome, as it includes state and control vectors that must be weighted by user defined matrices. Selection of these matrices is not straightforward since it requires familiarity with the subject and a vast number of simulations for refinement, Wongsathan and Sirima [8].

Usage of Evolutionary approaches to regulate optimal controllers has been stated in the literature in a variety of areas, producing notable results and reducing the time spent in the design process. In Vishal and Ohri [10], the authors successfully tuned a Linear Quadratic Regulator (LQR) and Proportional-Integral-Derivative (PID) controller using a Genetic Algorithm (GA) approach for the aircraft pitch control problem. Kukreti et al. [11] concluded that a GA-tuned LQR controller for the magnetically actuated attitude control of CubeSats was superior to the simple LQR and Proportional-Derivative controller, resulting in smaller steady state error and faster response. Sangdani et al. [12] obtained optimal control gains via Genetic Algorithms, the GA-based controller exhibited improved results since the conventional tuning techniques were not effective due to unseen non-linearities of the tracker robot. Dracopoulos and Jones [13] proposed a neural-genetic controller for the attitude control problem of a non-linear satellite in chaotic motion due to large external motions without any previous knowledge of the system dynamics.

This work is a extension of Silva et al. [14] and proposes a hybrid neural-genetic approach for the weighting matrices search and solution of the Algebraic Riccati Equation (ARE) that will result in the attitude controller gains.

The text is organized as follows. Section 2 introduces the neural-genetic method presented in this work along with simplified launcher model, control structure, Genetic Algorithm (GA) and Recurrent Neural Network (RNN) features. The simulation results are presented in Section 3. Finally, conclusions are given in Section 4.

# 2 Hybrid Neural-Genetic Approach

This section addresses the main features of the proposed approach. Since the launcher presents a non-linear set of equations and demonstration of the linearisation process is extensive, a simplified rigid-body linear model is presented. Next, the control structure is introduced. From there on, the evolutionary approaches are exposed.

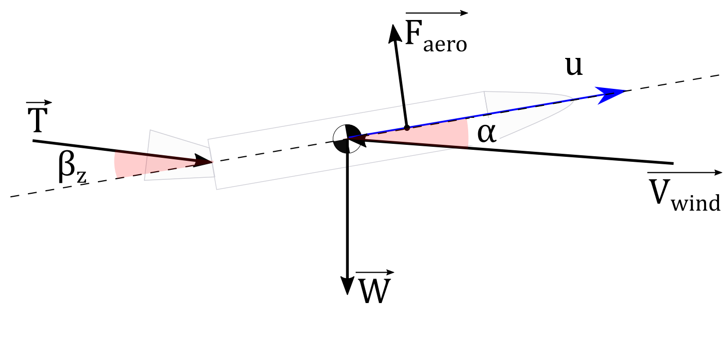
## 2.1 Rigid-Body Linear Model

This work considers a simplified linear dynamical model of a launch vehicle, given by Silva et al. [14], illustrated in Fig. 2, as follows:

$$\begin{bmatrix} \dot{\theta} \\ \dot{q} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\mu_q & \mu_\alpha/u \\ -g\cos\theta_N & u & -Z_\alpha/u \end{bmatrix} x + \begin{bmatrix} 0 \\ \mu_{\beta z} \\ Z_{\beta z} \end{bmatrix} \beta_z \tag{1}$$

where:

$\theta$ $\rightarrow$ pitch attitude angle;
$\beta_z$ $\rightarrow$ thrust deflection angle due to actuator deflection;
$\mu_\alpha$ $\rightarrow$ moment dimensional due to angle of attack, $\alpha$;
$\mu_q$ $\rightarrow$ moment dimensional due to pitch rate, $q$;
$\mu_{\beta z}$ $\rightarrow$ moment dimensional due to $\beta_z$;
$Z_\alpha$ $\rightarrow$ force dimensional due to $\alpha$;
$Z_{\beta z}$ $\rightarrow$ force dimensional due to $\beta_z$;
$g$ $\rightarrow$ gravity acceleration;
$u$ $\rightarrow$ velocity in the longitudinal axis.




Longitudinal plane

Figure 2. Simplified model of satellite launcher (longitudinal plane).

## 2.2 Control Structure

Figure 3 illustrates the control structure adopted to evaluate the resulting controller. In this structure, the control input, $\beta_z$, is the outcome of a proportional-integral controller based on the error, $(\theta_{ref} - \theta)$, and the weighted angular velocity feedback, $(d\theta/dt)$. According to Carmona and Leite Filho [15], this structure performs better tracking to reference commands, good robustness and time performance.

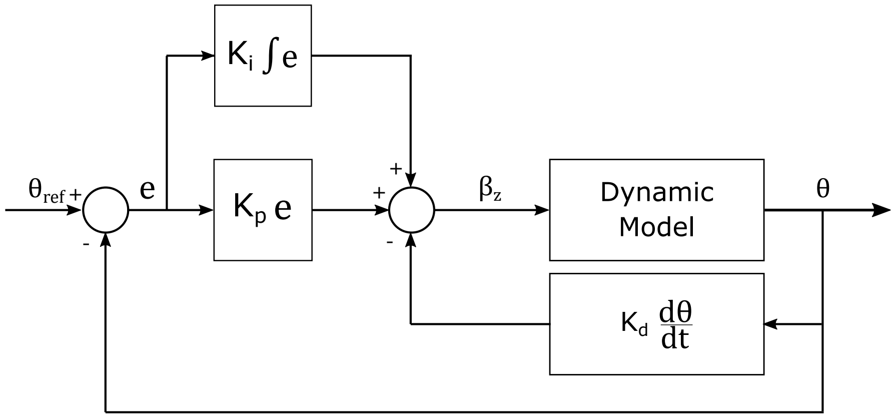



Figure 3. Control structure for the controller design.

**Closed-loop model** The closed loop state-space model for the control structure is given by

$$\begin{bmatrix} \dot{x}_{2\times1} \\ \hline \dot{\tau} \end{bmatrix} = \left[\begin{array}{cc|c} \multicolumn{2}{c|}{A_{2\times2}} & 0_{2\times1} \\ \hline -1 & 0 & 0 \end{array}\right] \begin{bmatrix} x_{2\times1} \\ \hline \tau \end{bmatrix} + \begin{bmatrix} B_{2\times1} \\ \hline 0 \end{bmatrix} \beta_z + \begin{bmatrix} 0_{2\times1} \\ \hline 1 \end{bmatrix} \theta_{ref} \tag{2}$$

where $\tau$ is the error integral

$$\tau = \int \theta_{ref}(t) - \theta(t)dt \tag{3}$$

and the control input, $\beta_z$, is given by

$$\beta_z = [-Kp \quad -Kd \quad Ki] \begin{bmatrix} x_{2\times1} \\ \hline \tau \end{bmatrix} + K_p\theta_{ref} \tag{4}$$

Once the system is in the form $\dot{x} = Ax + Bu$, the linear quadratic method can be used to find the $K_p$, $K_i$ and $K_d$ control gains in Eq. 4. The concern now is how the control problem will be encapsulated in Genetic Algorithms and how the GA will converge to good weighting matrices, $Q$ and $R$, that lead optimal control gains.

### 2.3 Genetic Algorithm

The GA-based weighting matrix search was based on Silva et al. [14]. On their work, the authors encapsulated the LQ method on a Genetic Algorithm as follows.

**Chromosome Model** Since $Q_{n\times n}$ and $R_{m\times m}$ are symmetric positive-definite matrices satisfying the linear quadratic requirements, the chromosome model can be given as the diagonal elements of $Q$ and $R$ matrices, where the total genes is

$$g = n + m \tag{5}$$

The resulting chromosome is then

$$QR_z = [q_{11}\ q_{22}\ \dots\ q_{nn}\ \ r_{11}\ r_{12}\ \dots\ r_{nn}] \tag{6}$$

**Population Model** A population is defined by a set of chromosomes. If a individual with $g$ genes contains $Q$ e $R$, then a population is represented by $QR_{n_{indiv}\times g}$, where $n_{indiv}$ is the number of chromosomes in the population.

**Fitness Model** The fitness function evaluates each individual in a population to ensure GA's convergence to a optimal solution. The model is given by

$$\begin{aligned}
K_z &= LQR_z(A, B, Q_z, R_z) \\
A_z &= (A - BK_z) \\
S_z &= \frac{||V_z||^2||W_z||^2}{< V_z, W_z >} \\
F_{S_z} &= \sum S_z \\
R_{S_z} &= rank(S_z, F_{S_z})
\end{aligned} \tag{7}$$

where $z = 1, ..., n_{indiv}$, $A_z$ is the closed-loop matrix for the gain vector $K_z$. $S_z$ is the sensibility, $V_z$ and $W_z$ are eigenvectors of $A_z$. $F_{S_z}$ is the fitness and $R_{S_z}$ represent each individual fitness. The fitness model in Eq. 7 scores each individual based on its current location in the $s$-plane and user-defined control goals.

**Elite Selection:** The elite selection ensures that the best individuals (highest fitness) of a given population will survive in the next generation. This operator avoids the fittest individuals being lost in crossover and mutation operations.

**Roulette Selection:** This operator is based on a random experiment that performs individuals selection based on their fitness.

**Crossover:** The crossover operator combines two individuals randomly in order to generate another two chromosomes.

**Mutation:** This operator is essential as it avoids premature convergence, Sastry et al. [16]. It randomly changes a gene of a given individual based on the probability of mutation, $p_m$.

### 2.4 Recurrent Neural Network

The neural network scheme followed in this work was first introduced by Wang and Wu [17]. In their work, the authors defined a new performance index based on the Algebraic Riccati Equation, adding the Cholesky factor, resulting in:

$$A^T P + PA - PBR^{-1}B^T P + Q + LL^T - P = 0 \tag{8}$$

Minimization of Eq. 8 can be achieved by the neural network set given by:

$$\begin{aligned}
\frac{dP(t)}{dt} &= -\eta_v[P(t)SU(t) + U(t)SP(t) - AU(t) \\
&\quad -U(t)A^T - Y(t)] \\
\frac{dL(t)}{dt} &= -\eta_z Y(t)L(t) \\
U(t) &= F[P(t)SP(t) - A^T P(t) - P(t)A + Q] \\
Y(t) &= F[L(t)L(t)^T - P(t)]
\end{aligned} \tag{9}$$

where $U$, $P$ and $Y$ are square matrices, $\eta_v$ and $\eta_z$ are design parameters and $L$ is a inferior triangular matrix. The RNAR structure is represented on Fig. 4, where four connected layers can be noted - the output layer representing the ARE solution, $P$, the input layer, $U$, and two hidden channels, $L$ and $Y$.

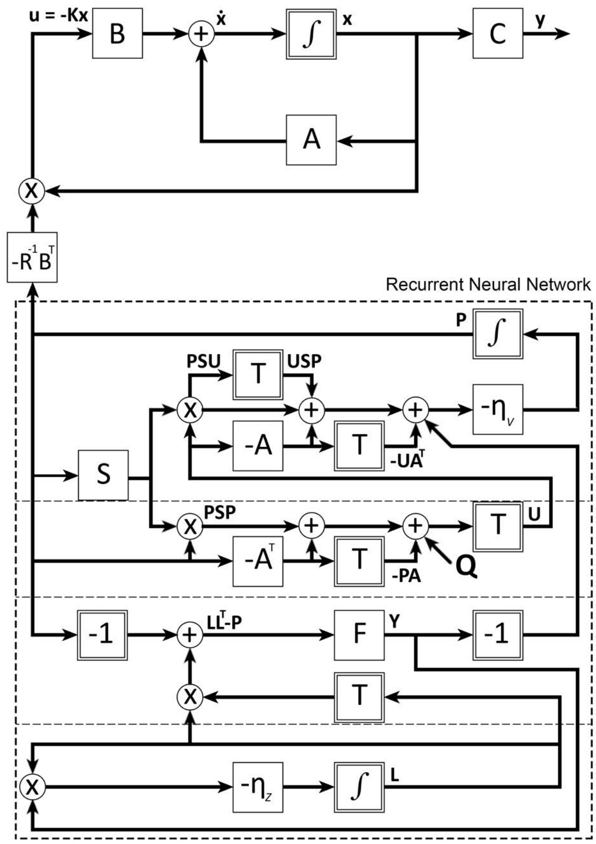



Figure 4. Recurrent Neural Network structure. From Wang and Wu [17]

## 3 Simulation Results

This section aims to present the simulation performance results of the Hybrid Neural-Genetic approach and a time-domain analysis of the control gains found by the proposed method.

### 3.1 Reduced Model

Considering the simplified launch vehicle rigid-body model, in Eq. 1, its basic parameters at the maximum dynamic pressure instant are:

$$\begin{aligned} \mu_\alpha &= 4.1600\, rad/s^2/rad & \qquad Z_{\beta z} &= 19.93\, m/s^2/rad \\ \mu_{\beta z} &= 7.2100\, rad/s^2/rad & Z_\alpha &= 48.90\, m/s^2/rad \\ \mu_q &= 0.0112\, rad/s^2/rad & g &= 9.810\, m/s^2 \\ u &= 596.9\, m/s & & \end{aligned} \tag{10}$$

The reduced model with the mode-selection approach, presented by Safonov et al. [18], is given as:

$$\dot{x} = \begin{bmatrix} -2.094 & 0.5478 \\ 0 & 1.985 \end{bmatrix} x + \begin{bmatrix} -1.163 \\ 1.377 \end{bmatrix} u \tag{11}$$

$$y = \begin{bmatrix} 1.238 & 1.194 \\ -2.594 & 3.048 \end{bmatrix} x$$

In order to validate the reduced model, a frequency domain analysis was performed, Fig. 5. As can be noted, the reduced order model is similar to the full order model for $\omega > 0.3\ rad/s$. Also, the mode-selection result is similar to the reduced model stated in Carmona and Leite Filho [15], where the model was contracted based on the longitudinal velocity, $u$.

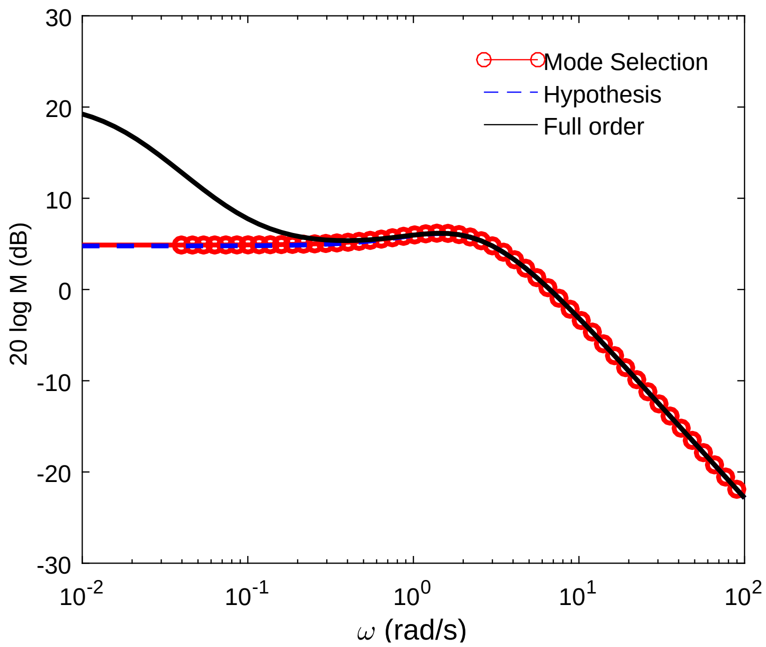



Figure 5. Reduced model frequency domain analysis

## 3.2 Genetic Algorithm

For the proposed work, the set of parameters used in the GA initialization are presented in Table 1. The population is composed by 20 individuals of each presented operator, except for mutation - 40 individuals. In addition, 20 new random individuals are generated on each iteration, in order to avoid premature convergence.

Table 1. Genetic Algorithm parameters.

| Parameter | Quantity |
|---|---|
| Chromosome dimension | 4 |
| Population size | 120 |
| Mutation probability | 5 |
| Mutation factor | 0.1 |

As the controller needs to be evaluated based on time domains specifications and these are directly linked with eigenvalues position on the s-plane, there is a need on the definition of control goals and constraints for the weighting matrices design process, Table 2. Finally, individuals fitnesses will be weighted based on each of these achieved features.

Table 2. Control goals and constraints.

| Specification | Range | Weight |
|---|---|---|
| Eigenstructure | $-8 \pm 0.1j < \lambda_i < 0 \pm 0.1j$ | 1.2 |
| Settling time | $8$ s $< t_s < 10$ s | 1.2 |
| Rise time | $0.5$ s $< t_r < 1$ s | 1.5 |
| Overshoot | $\%OS < 50\%$ | 1.5 |

**Initial population** Initial population was randomly created with $120$ individuals. Mean fitness of this population was $3.27$ and the best individual presented fitness of $6.5$. As can be noted in Fig. 6, initial population presented good diversity, that is essential to avoid local maxima or minima.

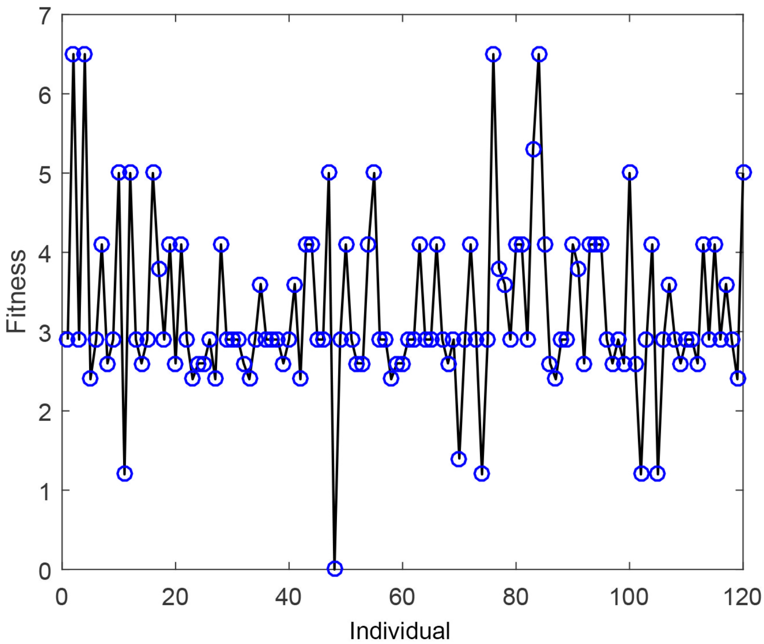


Figure 6. Reduced model frequency domain analysis

**Final population** As can be noted in Fig. 7, $40\%$ of the individuals reached maximum fitness $(8)$. Also, diversity was reduced suggesting that GA is close to the stop criterion. Last individuals of this population presented poor fitness as they were recently created.

**Fitness Evolution** Figure 8 highlights the evolution of the mean fitness of populations. It can be noted that GA met the stop criteria with $50$ iterations.

**Comments** The effect of each GA parameter was evaluated. Mutation probability and mutation factor represent the key parameters since they directly affect the speed of convergence. Very high or very low values lead to divergence from optimal solution. Also, eigenstructure size can also culminate the convergence.

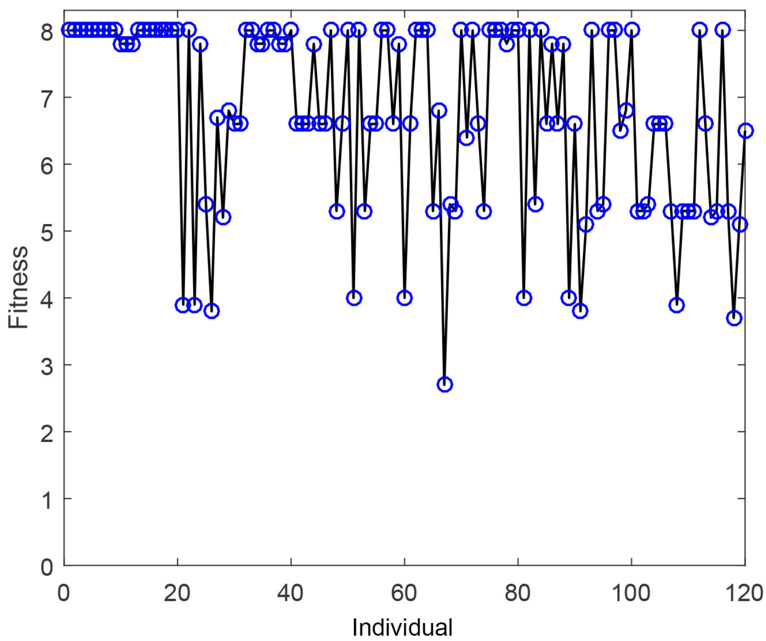



Figure 7. Reduced model frequency domain analysis

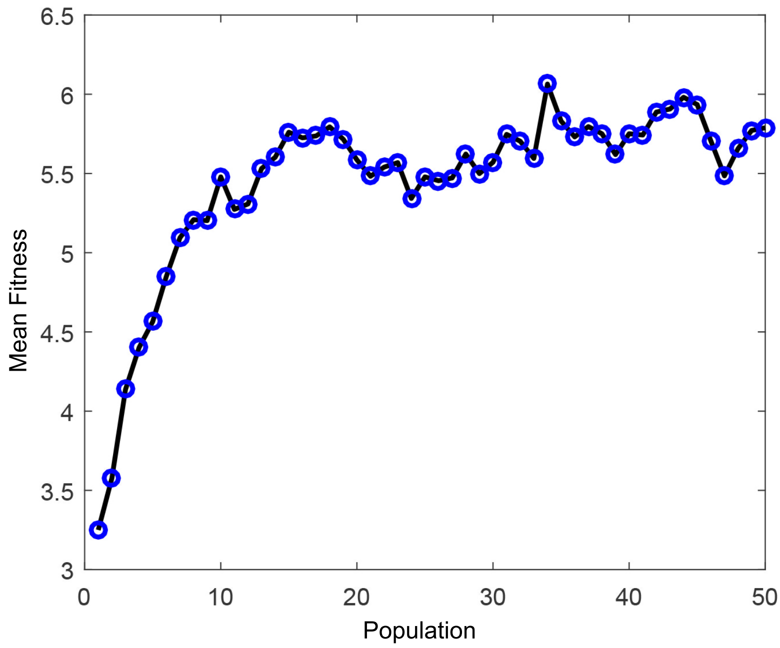



Figure 8. Reduced model frequency domain analysis

**Weighting matrices** The GA's best individual produced the following weighting matrices. This chromosome was chosen to evaluate the controller gains with the RNN in the next section.

$$Q = \begin{bmatrix} 0.6525 & 0 & 0 \\ 0 & 0.2615 & 0 \\ 0 & 0 & 0.1713 \end{bmatrix} \quad \text{and} \quad R = 0.5707 \tag{12}$$

### 3.3 Recurrent Neural Network

This section aims to present the Recurrent Neural Network analysis. The RNN performance is directly linked with $\eta_v$ and $\eta_z$ parameters, that are evaluated based on the infinity norm and energy surfaces. These parameters must be initialized to improve stability, convergence and solvability of the RNN, da Fonseca Neto et al. [9].

Figures 9 and 10 show that the range $4 < \eta_v < 9$ and $50 < \eta_z < 100$ minimizes both the infinity norm and energy surface. Since there is not a mismatch between the surfaces (i.e. infinity norm is minimized while energy is not, or vice versa), $\eta_v = 8$ and $\eta_z = 100$ were selected to proceed.

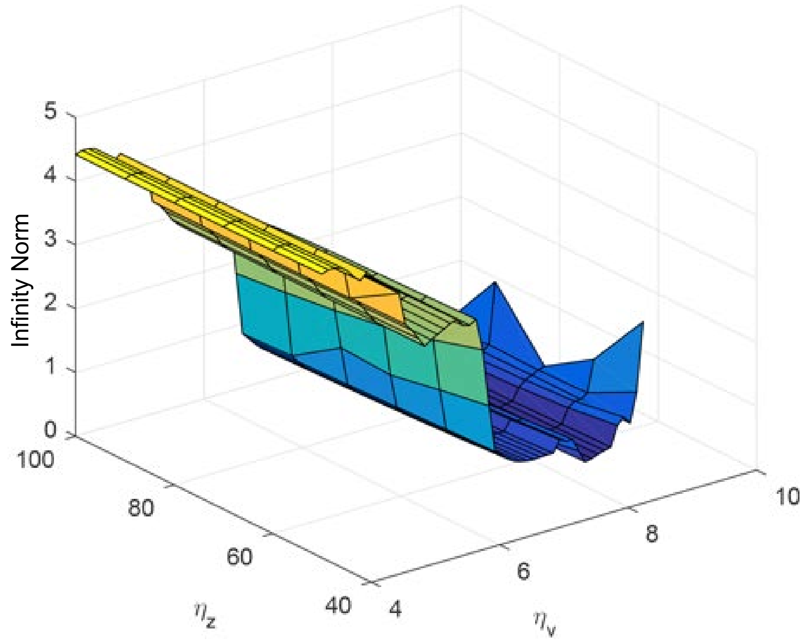



Figure 9. Reduced model frequency domain analysis

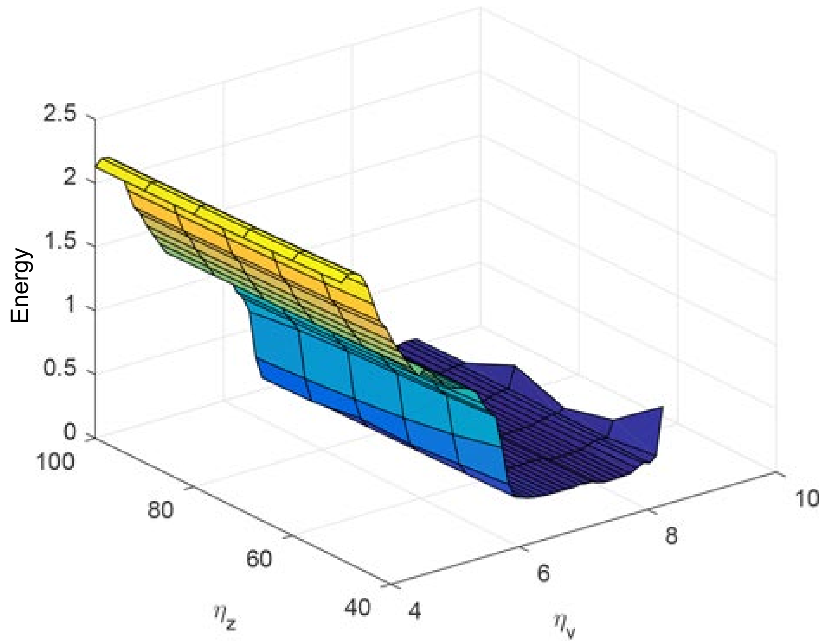



Figure 10. Reduced model frequency domain analysis

**Neural Solution to ARE** The RNN output layer (i.e. ARE solution) solves the Eq. 9, that minimizes the performance index given in Eq. 8. Final ARE solution is given by:

$$P = \begin{bmatrix} 0.9082 & 0.1735 & -0.3228 \\ 0.1735 & 0.0817 & -0.0434 \\ -0.3228 & -0.0434 & 0.0607 \end{bmatrix} \tag{13}$$

The controller gains calculated with the GA-based weighting matrices, $Q$ and $R$, and the RNN solution, are then given by

$$K = [2.1917 \quad 1.0325 \quad -0.5478] \tag{14}$$

### 3.4 Time Domain Analysis

As the final intention of controllers is to track a reference command with good robustness and time performance, this subsection aims to perform a step response analysis. The closed-loop step response for $K$ in Eq. 14 is represented in Fig. 11. As can be noted in the illustration, the proposed controller based on a reduced order model can fully represents the full order system. The time domains specifications for this closed-loop step response follows: $t_r = 0.568\ s$, $t_s = 9\ s$, $\%OS = 39.9\%$ and $e_\infty < 1$.

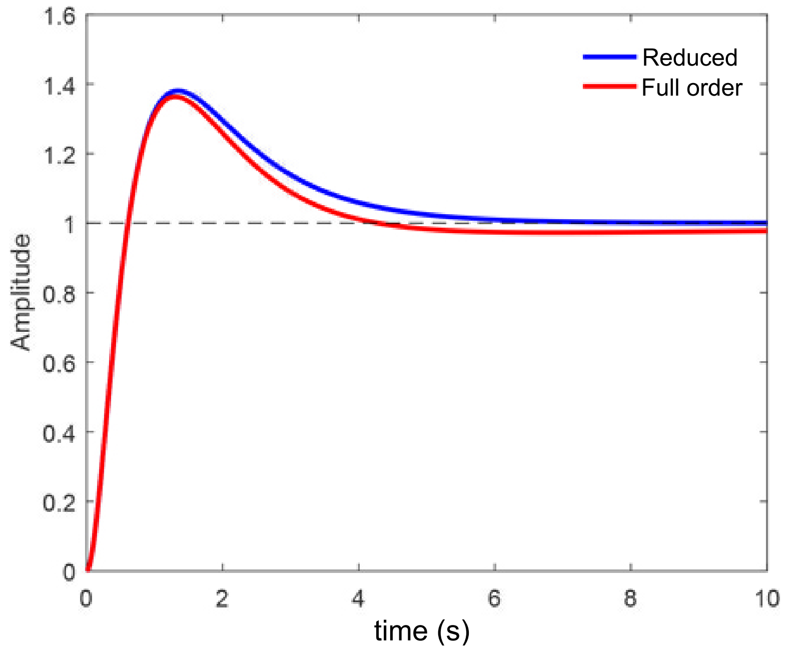



Figure 11. Reduced model frequency domain analysis

**Comments** Although all the time domain specifications have been satisfied in the step response, this one time domain simulation is lack of reality. First, flex modes were not considered in this work. Second, the rocket undergoes aerodynamic force effects due to wind. Third and most importantly, the real flight is more than 70 seconds longer, that means, the controller gains must be computed for each second (gain scheduling). Furthermore, other outputs such as angle of attack, actuator deflection, linear and angular velocities must be analysed since they are usually constrained. Analysis of these parameters is crucial, yet it is far out of this article focus.

## 4 Conclusion

In this text, a hybrid neural-genetic approach for the satellite launcher attitude controller design was proposed since currently techniques require prior experience about the problem and often result in inefficient controller.

Difficulty in find the weighting matrices was overcame by using the GA-based search. Overall results show that the proposed method reaches the design specifications with $30 - 50$ iterations along with a population of $120$ elements. Additionally, to ensure unique ARE solution a RNN approach was implemented. This process was carried out with parameters tuning based on the minimization of infinity norm and energy surfaces.

The step response represented the time domain analysis. Although this only response is not sufficient to ensure a perfect real flight, it is enough to prove convergence of the hybrid neural-genetic approach to a solution.

Indeed, usage of evolutionary techniques speeds up the controller design process and reduce costs. Consequently, it is believed that the proposed approach can be used instead analytical methods.

For future work, authors will propose new fitness model approaches, usage of other evolutionary algorithms, such as fuzzy logic, in the search problem and refine the control problem to a more realistic one.

## Acknowledgements

The authors would like to thank Professor Alain Giacobini, Instituto Tecnológico de Aeronática, for his valuable comments on this work and Fundação de Amparo à Pesquisa e ao Desenvolvimento Científico e Tecnológico do Maranhão for the great opportunity to obtain a Master Degree.

## References

[1] Markley, F. L. & Crassidis, J. L., 2014. *Fundamentals of spacecraft attitude determination and control*, volume 33. Springer.

[2] Tiwari, P. M., Janardhanan, S., & un Nabi, M., 2016. Attitude control using higher order sliding mode. *Aerospace Science and Technology*, vol. 54, pp. 108 – 113.

[3] Tiwari, P. M., Janardhanan, S., & un Nabi, M., 2015. Rigid spacecraft attitude control using adaptive integral second order sliding mode. *Aerospace Science and Technology*, vol. 42, pp. 50 – 57.

[4] Han, Y., Biggs, J. D., & Cui, N., 2015. Adaptive fault-tolerant control of spacecraft attitude dynamics with actuator failures. *Journal of Guidance, Control, and Dynamics*, vol. 38, n. 10, pp. 2033–2042.

[5] Palmerio, A. F., 2016. *Introdução à Tecnologia de Foguetes*. SindCT, 1 edition.

[6] Daitx, H. & Kienitz, K. H., 2014. Desenvolvimento de algoritmo de controle para o veículo lançador de microssatélites (vlm). *Anais do XX Congresso Brasileiro de Automática*, pp. 4107–4114.

[7] Das, S., Pan, I., Halder, K., Das, S., & Gupta, A., 2013. Optimum weight selection based lqr formulation for the design of fractional order pi$\lambda$d$\mu$ controllers to handle a class of fractional order systems. In *2013 International Conference on Computer Communication and Informatics*, pp. 1–6.

[8] Wongsathan, C. & Sirima, C., 2009. Application of ga to design lqr controller for an inverted pendulum system. In *2008 IEEE International Conference on Robotics and Biomimetics*, pp. 951–954.

[9] da Fonseca Neto, J. V., Abreu, I. S., & da Silva, F. N., 2010. Neural-genetic synthesis for state-

space controllers based on linear quadratic regulator design for eigenstructure assignment. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, n. 2, pp. 266–285.

[10] Vishal & Ohri, J., 2014. Ga tuned lqr and pid controller for aircraft pitch control. In *2014 IEEE 6th India International Conference on Power Electronics (IICPE)*, pp. 1–6.

[11] Kukreti, S., Walker, A., Putman, P., & Cohen, K., 2015. Genetic algorithm based lqr for attitude control of a magnetically actuated cubesat. In *AIAA Infotech @ Aerospace*, pp. 0886.

[12] Sangdani, M., Tavakolpour-Saleh, A., & Lotfavar, A., 2018. Genetic algorithm-based optimal computed torque control of a vision-based tracker robot: Simulation and experiment. *Engineering Applications of Artificial Intelligence*, vol. 67, pp. 24 – 38.

[13] Dracopoulos, D. C. & Jones, A. J., 1997. Adaptive neuro-genetic control of chaos applied to the attitude control problem. *Neural Computing & Applications*, vol. 6, n. 2, pp. 102–115.

[14] Silva, P., Silva Abreu, I., Almeida Forte, P., & Costa do Amaral, H. M., 2019. Genetic algorithms for satellite launcher attitude controller design. *Inteligencia Artificial*, vol. 22, n. 63, pp. 150–161.

[15] Carmona, D. & Leite Filho, W., 2004. Analytical method for computing controller gains of a satellite launcher. *IFAC Proceedings Volumes*, vol. 37, n. 6, pp. 1019–1023.

[16] Sastry, K., Goldberg, D. E., & Kendall, G., 2014. *Genetic Algorithms*, pp. 93–117. Springer US, Boston, MA.

[17] Wang, J. & Wu, G., 1998. A multilayer recurrent neural network for solving continuous-time algebraic riccati equations. *Neural Networks*, vol. 11, n. 5, pp. 939 – 950.

[18] Safonov, M. G., JONCKHEERE, E. A., Vermaj, M., & LIMEBEER, D. J., 1987. Synthesis of positive real multivariable feedback systems. *International Journal of Control*, vol. 45, n. 3, pp. 817–842.