

# A Comparative Study of PID, PID with Tracking, and FPID Controller for Missile Canard with an Optimized Genetic Tuning Method Using Simscape Modelling

Saif ALAMERI, Dragan LAZIC, Milan RISTANOVIC

**Abstract:** In this paper, a comparative study between PID, PID with tracking feedback, and Fractional Order PID for a missile canard is presented. The missile canard is modelled using computer aided design (CAD) software and imported into the Simscape, a physical system-modelling tool within Simulink environment. Since a mathematical model is not derived, the genetic tuning algorithm is used to tune the controllers. In addition, the genetic algorithm tuning is refined by improving the fitness function.

**Keywords:** fractional order PID; genetic algorithm; missile canard; PID; tracking feedback

## 1 INTRODUCTION

The PID controller is one of the most popular controllers used in a wide range of industries [1]. Modern missiles also use PID controllers for the actuation of their control surfaces. The use of PID controller for DC motors in aerospace applications is reviewed in [2]. Due to technological progress, several optimizations to the PID controllers were introduced. A study of the performance differences between PID and FPID have been presented in [3]. Hence, in order to evaluate the optimization to the PID controller for missile canard, a comparative study is performed. The comparison of different controllers is presented in section 2.

In addition, the evaluation of the controllers requires a model to simulate the behavior of the missile canard. Traditional methods use higher order transfer function to estimate the canard behavior [4]. However, since transfer functions are linear, a more accurate estimation is needed. There are several papers attempting to model a dynamic system. The use of Takagi-Sugeno-Kang Fuzzy method, which is an iterative of modelling, is given in [5]. Moreover, [6] proposes the use of artificial neural network to model the dynamic system, which requires the input and outputs of the system to be estimated. However, since the iteration and training of neural network can be time consuming, this paper proposes an alternative solution. The missile canard can be drawn in CAD software then imported into Simscape for simulation. The simulated model will be compared to a physical canard to determine its accuracy. The modelling approaches are discussed in section 3.

Since the proposed modelling method does not have a mathematical model, traditional method of tuning cannot be used. There are several papers that address this issue. For example, [7] proposes using online controller tuning and recursive least square method. In addition, [8] proposes the use of genetic algorithm as a technique to tune the controllers. This paper improves on the genetic algorithm tuning technique by refining the fitness function as shown in section 4.

Section 5 compares the performance of all the controllers for both missile canards modelled in Simscape and actuated physically. Furthermore, it shows the results of the refined genetic algorithm tuning by illustrating the errors corresponding to each controller. The novelty of this

paper is in the fact that it provides an overall study of different PID controllers, genetic tuning algorithm refinement, and a new modelling approach for the missile canard.

## 2 CONTROLLERS

There are three controllers used in this comparison: the PID controller, the PID controller with tracking feedback, and Fractional Order PID.

### 2.1 PID Controller

The PID controller's mathematical model is shown in Eq. (1) [9]. Here, it can be seen that the PID controller calculates the control signal  $u(t)$  by adjusting the error  $e(t)$ , the derivative of error, and integral of error with their corresponding gains. Simulink's implementation of the PID block diagram is illustrated in Fig. 1 [10]. It is important to note that the derivative part of the controller's implementation is done by a filter with a gain  $N$ . In order to maintain consistency, the Simulink PID algorithm is also used in the microcontroller for the physical missile canard actuator.

$$u(t) = K_p e(t) + K_i \int_0^{\tau} e(t) dt + K_d \frac{de(t)}{dt} \quad (1)$$

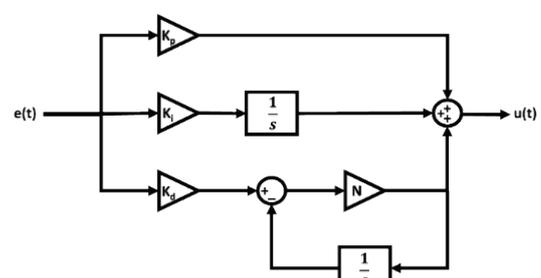


Figure 1 PID block diagram in Simulink

### 2.2 PID Controller with Tracking Feedback

One optimization for the PID controller is the inclusion of a feedback for the control signal  $u(t_{n-1})$ . This allows for the controller to consider the magnitude of the control signal in the calculation of the control signal  $u(t)$ . Eq. (2)

shows the mathematical model of the PID controller with tracking feedback. In addition, Fig. 2 illustrates the Simulink implementation of the block diagram [11]. Similarly, the Simulink PID with tracking algorithm is used in the microcontroller of the physical missile canard.

$$u(t) = \int_0^{\tau} K_i e(t) + K_t(u(t_{n-1}) - u(t)) dt + K_p e(t) + K_d \frac{de(t)}{dt} \quad (2)$$

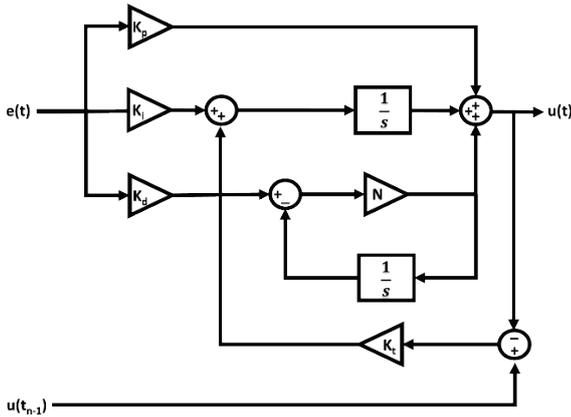


Figure 2 PID with tracking block diagram in Simulink

2.3 Fractional Order PID Controller

Fractional Order PID controller implies that the derivative and the integration of the PID controller are of fractional order. This allows the finer tuning of the control signal  $u(t)$  that is not possible with traditional PID controller. Eq. (3) shows the mathematical model of the controller [12]. Simulink does not have a library for the FPID. Therefore, the controller from the FOMCON library, which allows for the computation of the fractional order derivative and integration, is used. The block diagram is illustrated in Fig. 3 [13]. The FOMCON evaluation of fractional order is also used in the microcontroller of the physical missile canard.

$$u(t) = K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^{\mu} e(t) \quad (3)$$

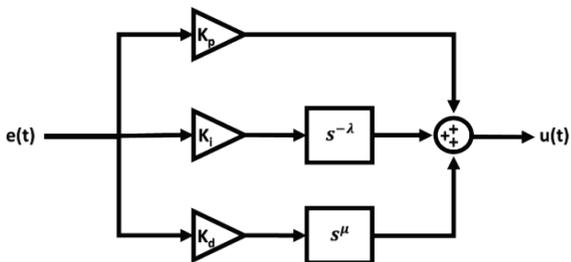


Figure 3 FPID block diagram in Simulink

3 MISSILE CANARD MODELLING

Transfer functions have been used traditionally to represent systems but they are not accurate because of their linearity. There are alternative modelling techniques used in the scientific community that are discussed in section 3.2. However, this paper uses Simscape in Simulink as the

modelling tool. That is because the Simscape allows for full CAD models to be imported into the Simulink [14]. This means that the missile canard structure and joints are imported directly and ready for simulation. Moreover, Simscape has advanced numerical model for mechanical and electrical system, which allows the controller and motor to be easily simulated. Simscape is divided into several categories each severing an integral function of the modelling tool. The categories used in this paper are multibody, mechanical, and electrical. In addition, traditional Simulink blocks are used to simulation controller software as it connects directly to the Simscape model.

3.1 Physical Canard

The canard used in this paper is shown in Figure 4. Here, the DC motor's torque is multiplied by the mechanical gearing design. In addition, the controller for the physical canard is shown in Figure 5.

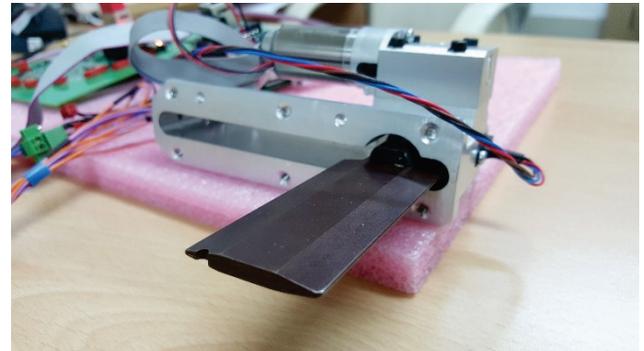


Figure 4 Physical missile canard



Figure 5 Physical missile canard controller

3.2 Alternative Modelling Methods in the Field

There are two alternative modelling techniques used in the scientific community. The Takagi-Sugeno-Kang (TSK) Fuzzy modelling is an online training method. The algorithm of the TSK technique is shown in Fig. 6. Here, the model parameters are iteratively adjusted in order to match the behavior of the dynamic model [5].

Another popular modelling method is the neural network presented shown in Fig. 7. The forces of the object are set as input while the output is set as displacement. The neural network is then trained and used to simulate the behavior of the dynamic system [6].

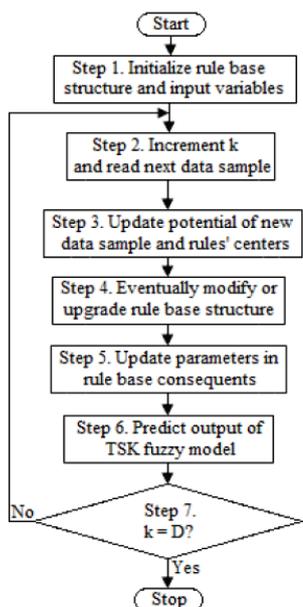


Figure 6 Takagi-Sugeno-Kang (TSK) Fuzzy modelling algorithm [5]

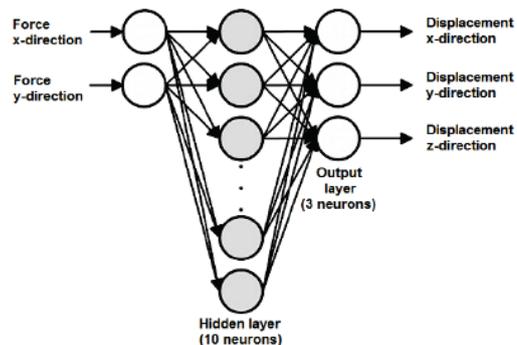


Figure 7 Neural Network modelling [6]

### 3.3 Simscape Multibody

Simscape Multibody allows for CAD models to be imported directly into Simulink as shown in Fig. 8. The blocks correspond to the pieces of the canard and the relationship between its joints [14]. It is important to note the CAD in Multibody only models the canard and the rod. Hence, the inertia has been adjusted to more accurately represent the physical system. The DC motor as well as the controller circuits is modelled using Simscape Electrical and Mechanical. This allows for simplicity as well as the evaluation of the rest of the Simscape categories.

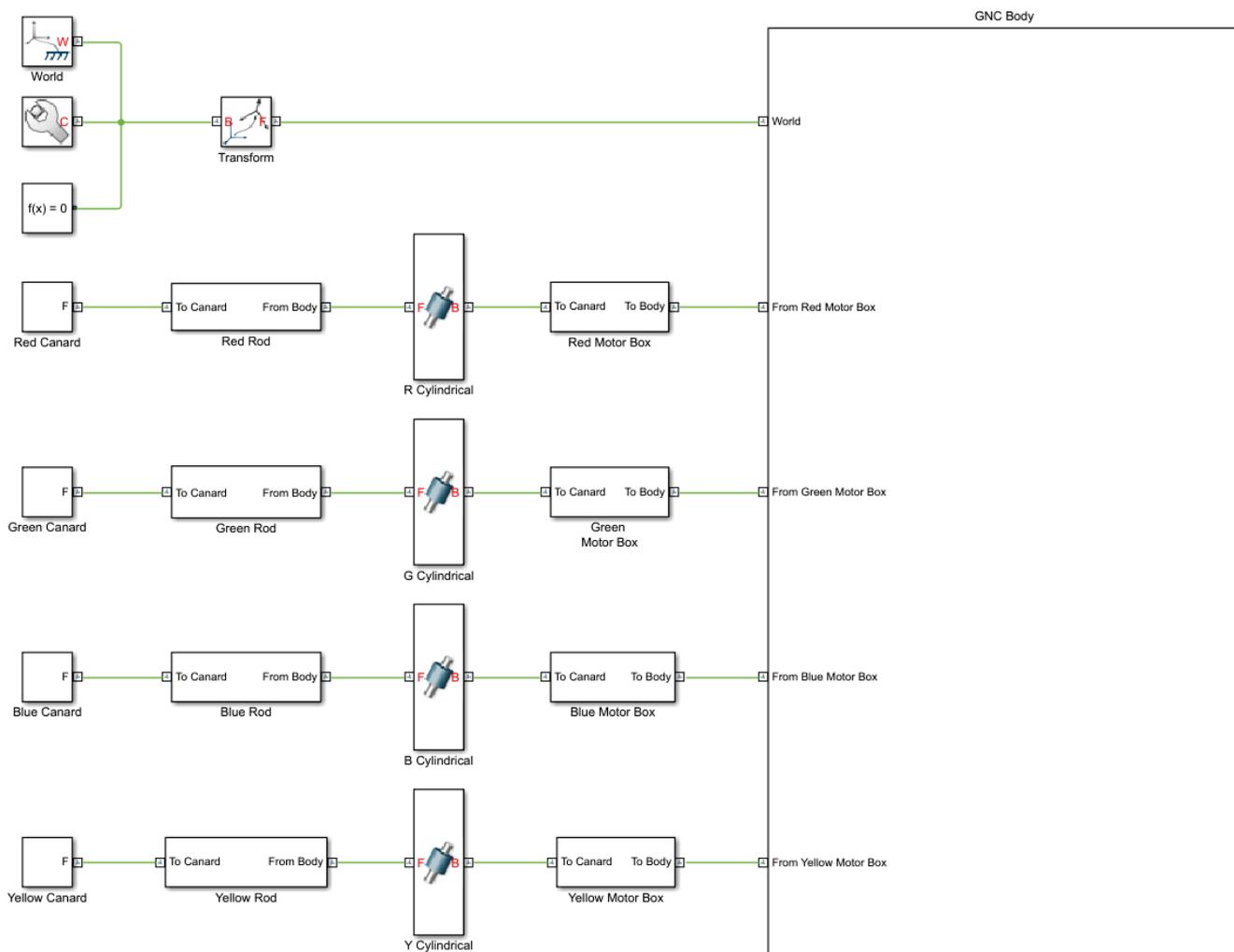


Figure 8 Simscape Multibody block of the missile canard

In addition, the missile canard model can be viewed using the Mechanics Explorer. This allows for the behavior of the canards to be visualized as shown in Fig. 9.

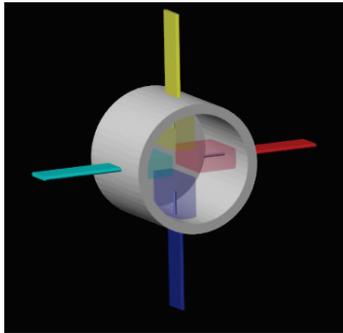


Figure 9 Simscape multibody mechanics explorer

### 3.3.1 Simscape Mechanical

Simscape Mechanical allows for mechanical parts to be modelled in Simulink. Here, the DC motor as well as the gear box are modelled as shown in Fig. 10 [15].

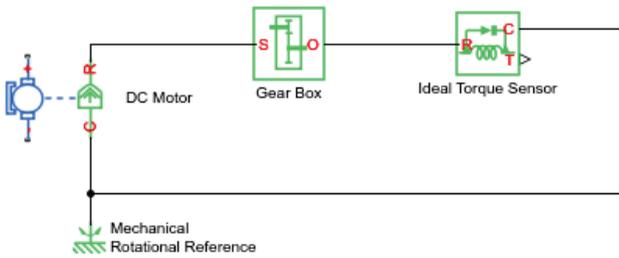


Figure 10 Simscape mechanics DC motor

### 3.3.2 Simscape Electrical

Simscape Electrical allows for electrical circuits to be modelled in Simulink. Here, the DC motor control circuit is modelled as shown in Fig. 11 [16].

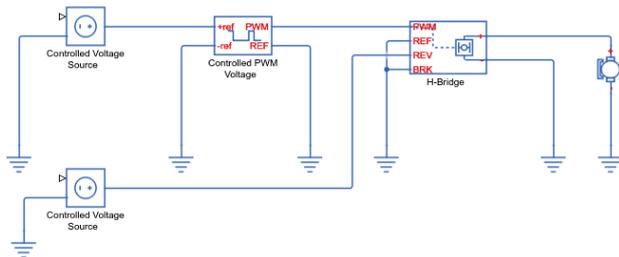


Figure 11 Simscape electrical DC motor control circuit

## 3.4 Controller Software Simulation

The software portion of the canard is modelled in the Simulink. Fig. 12 shows the implementation of the PID controller in Simulink.

## 4 TUNING METHODS WITHOUT TRANSFER FUNCTIONS

### 4.1 Iterative Feedback Tuning Method

One method of tuning in the scientific community is the use of iterative feedback tuning (IFT) with recursive least square. Here, the closed loop input and output parameter are obtained. Then, the arbitrary reference signal

is calculated. The reference signal is used to tune the controller for optimal performance index [7].

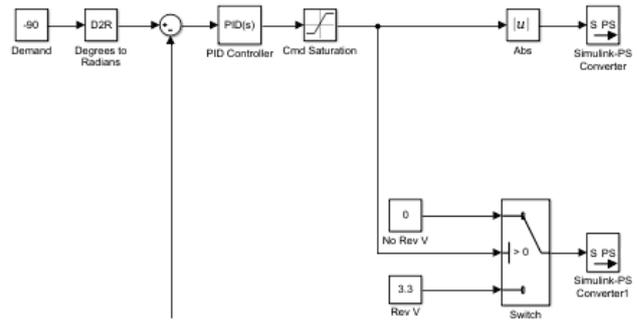


Figure 12 PID controller in Simulink

### 4.2 Genetic Tuning Method

Another method is the use of Genetic algorithm, which is an advanced multidimensional random search technique based on biological models such as natural selection. The genetic algorithm evaluates the performance of the controller by a fitness function. Cao's paper proposes the term  $J_1$  and  $J_2$  as a measurement of control performance quality. The  $J_1$  term quickly tracks error but is prone to oscillation. The  $J_2$  has good response but bad selection performance. Therefore, Cao proposes an optimized performance quality  $J$  which weightage values  $w_1$  and  $w_2$ . Once  $J$  is obtained, the fitness function  $F$  can be evaluated [8].

$$J_1 = \int_0^{\infty} e^2(t) dt \tag{4}$$

$$J_2 = \int_0^{\infty} |e(t)| dt \tag{5}$$

$$J = \int_0^{\infty} (w_1 |e(t)| + w_2 u^2(t)) dt \tag{6}$$

$$F = \frac{1}{J} \tag{7}$$

It is important to keep in mind the stop criteria for the genetic algorithm. There is a generation limit that sets the boundary for the number of iterations for the genetic algorithm. In addition, there is a time limit which controls the maximum time spent finding the optimal solution. Furthermore, there is a range limit, which sets the boundaries for the multidimensional search. There is no unique solution for the genetic algorithm. Hence, it is essential to choose the correct limits. The stop criteria for the range limits in this paper are  $K_p = [0.001, 20]$ ,  $K_i = [0.001, 20]$ ,  $K_d = [0.001, 20]$ ,  $\lambda = [0.001, 1]$ , and  $\mu = [0.001, 1]$ . These limits are chosen because of the canard's optimal performance range. In addition, a time limit of twenty-four hours was chosen to allow enough time for the optimal solution to be found. Moreover, a generation limit of 500 was selected. Lastly, a stop criterion of an average change of less than  $1 \times 10^{-6}$  was implemented.

### 4.3 Fitness Function Refinement

This paper proposes an alternative to Cao's fitness function to address the shortcomings of the genetic algorithm. Eq. (8) takes the Means Square Error (MSE) of the response  $Y_i$  and controller command  $Cmd$ . However, by only using  $F_1$  in the fitness function, the response will be prone to overshooting. Hence, an additional term  $F_2$  is introduced to limit the overshooting. This is done by taking the square of the error between maximum  $Y_{Overshoot}$  and the controller command. Lastly, the two terms are added in equation 10 to form the fitness function  $F$ .

$$F_1 = MSE(Y_i, Cmd) = \frac{1}{n} \sum_{i=1}^n (Y_i - Cmd)^2 \quad (8)$$

$$F_2 = (Y_{Overshoot} - Cmd)^2 \quad (9)$$

$$F = F_1 + F_2 \quad (10)$$

It is important to note that Cao's fitness function focuses on lowering the control signal. While some applications require limited control signal, this comes at the cost of performance. Nevertheless, the refined fitness function focuses on minimizing the error of the controller output, hence, producing results with less error. However, it does not consider control signal magnitude, which could be troublesome for some applications.

## 5 CONTROLLER PERFORMANCES

The three controllers' performances are compared to determine the most optimal controller for the missile canard. The controllers' comparison will be performed on Simscape. In addition, it will also be performed on the physical actuator. This allows for the accuracy of the Simscape model to be evaluated. In addition, it allows for the difference in performances between the physical canard controllers and the simulated canard controllers to be evaluated and compared. In addition, the genetic algorithm with refined fitness function will be used to tune the simulated controller. Then, the same gains will be used in the physical microcontroller. Moreover, the time step of 0.2 ms is used for uniformity. It is important to note that section 5.3 compares Cao's genetic algorithm tuning to the one proposed in this paper.

### 5.1 Step Response

The step response is used to evaluate the performance of the controller to the change of the command. The command steps from 0 to 10°.

#### 5.1.1 PID step response

Fig. 13 shows the step response of the PID controller for both simulation and physical canard. The simulation overshoots 0.01° but slowly corrects it. The physical actuator overshoots 0.02° and also slowly corrects it. Both simulated and physical canards behave similarly.

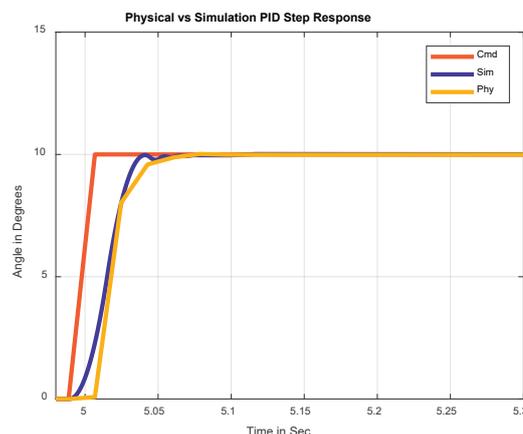


Figure 13 Physical vs simulation PID step response

#### 5.1.2 PID with Tracking Step Response

Fig. 14 shows the step response of the PID with tracking controller for both simulation and physical canard. The simulation overshoots 0.05° but slowly increases the error. The physical actuator overshoots 0.02° and maintains a static error of 0.01°. Both simulated and physical canards are relatively similar.

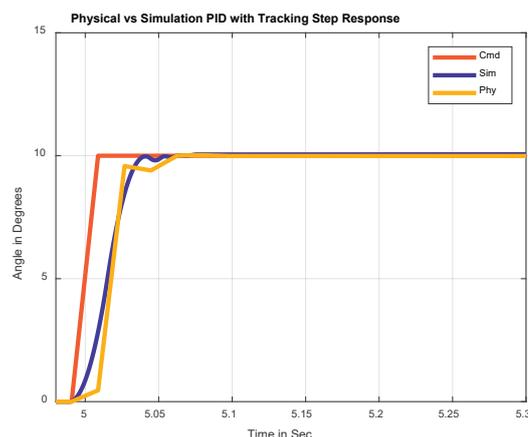


Figure 14 Physical vs simulation PID with tracking step response

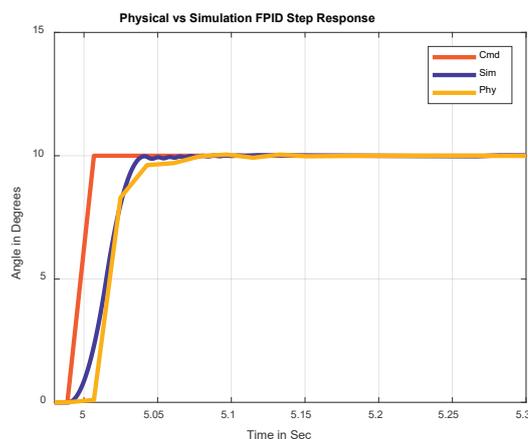


Figure 15 Physical vs simulation FPID step response

#### 5.1.3 FPID Step Response

Fig. 15 shows the step response of the FPID controller for both simulation and physical canard. The simulation does not overshoot initially but oscillates between  $\pm 0.024^\circ$ .

The physical actuator also does not overshoot initially, but it oscillates between  $\pm 0.05^\circ$  for 10 ms, then settles to with  $0^\circ$  error. Both simulated and physical canards behave relatively similar.

### 5.1.4 All Simulated Controllers' Step Response

Fig. 16 shows the comparative response of all the simulated controllers. Fig. 17 shows the comparative error of all the controllers. As it can be deduced from both figures, the FPID controller has the least error initially, however, the PID controllers managed to outperform it. The PID controller with tracking feedback has the largest error.

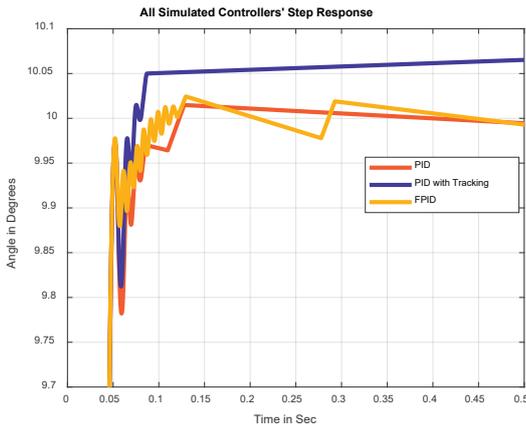


Figure 16 Comparison of all simulated controllers' step response

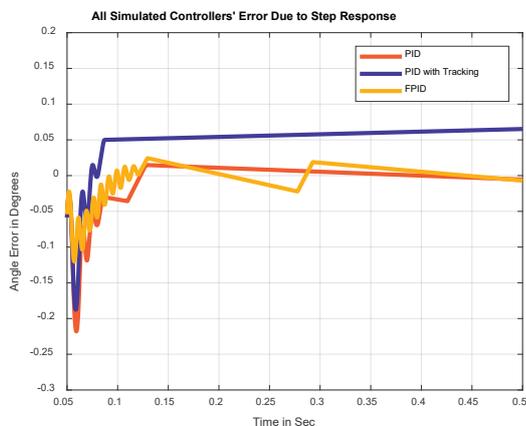


Figure 17 Comparison of all simulated controllers' error due to step response

### 5.1.5 All Physical Controllers' Step Response

Fig. 18 shows the comparative response of all the physical controllers. Fig. 19 shows the comparative error of all the controllers. As it can be deduced from both figures, the PID tracking controller reaches the demand fastest but has an odd shape. The FPID controller reaches slowest, oscillates, but then maintains zero error. The PID controller is the most stable.

## 5.2 Response to Sine Wave Input

The response to sine wave is used to evaluate the performance of the controllers to the constant change of the command. The command varies from  $-10^\circ$  to  $10^\circ$  with a frequency of 5 Hz.

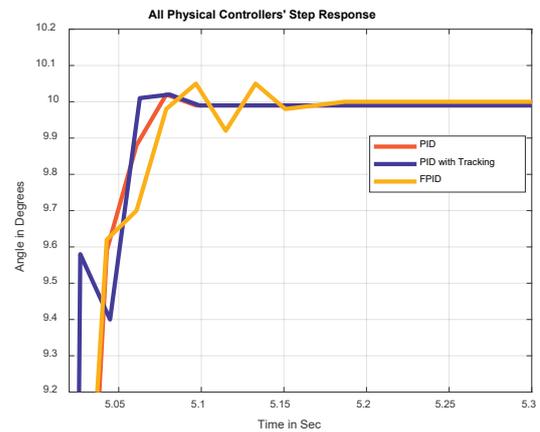


Figure 18 Comparison of all physical controllers' step response

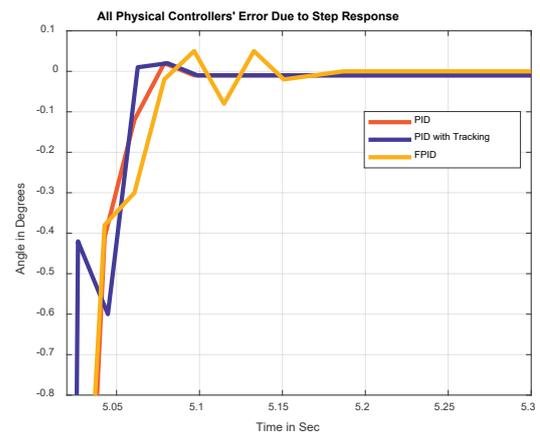


Figure 19 Comparison of all physical controllers' error due to step response

### 5.2.1 PID Response to Sine Wave Input

Fig. 20 shows the response to sine wave of the PID controller for both simulation and physical canard. The simulation overshoots  $0.7^\circ$  at maximum. The physical actuator undershoots by  $0.27^\circ$  and at minimum. The simulated and physical canards differ mostly in overshooting.

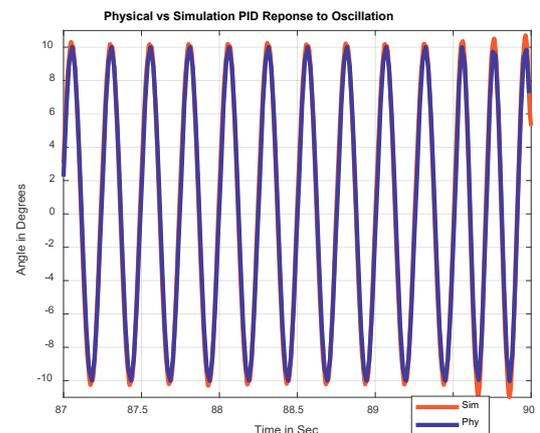


Figure 20 Physical vs simulation PID response to sine wave

### 5.2.2 PID with Tracking Response to Sine Wave Input

Fig. 21 shows the response to sine wave of the PID with tracking controller for both simulation and physical canard. The simulation overshoots  $0.7^\circ$  at maximum. The physical actuator undershoots by  $0.29^\circ$  and at minimum. The

simulated and physical canards differ mostly in overshooting.

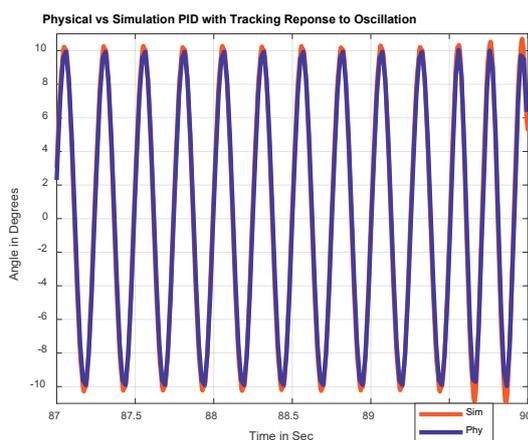


Figure 21 Physical vs simulation PID with tracking response to sine wave

### 5.2.3 FPID Response to Sine Wave

Fig. 22 shows the response to sine wave input of the FPID controller for both simulation and physical canard. The simulation overshoots  $0.83^\circ$  at maximum. The physical actuator undershoots by  $0.19^\circ$  and at minimum. The simulated and physical canards differ mostly in overshooting.

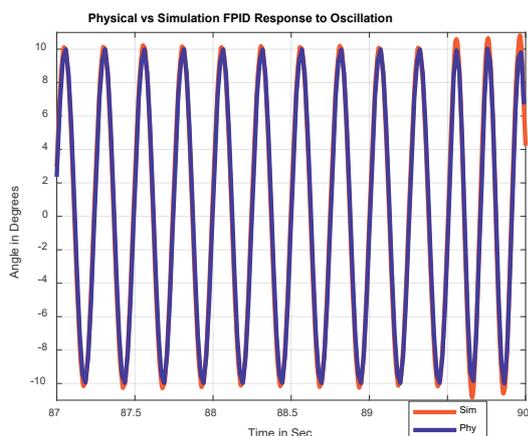


Figure 22 Physical vs simulation FPID response to sine wave

### 5.2.4 All Simulated Controllers' Response to Sine Wave

Fig. 23 shows the comparative response of all the simulated controllers. Fig. 24 shows the comparative error of all the controllers. As it can be deduced from both figures, the FPID controller has the least error and continues to maintain it. Nonetheless, the error starts increasing towards the end. The PID with tracking controller has the higher error throughout the simulation. The PID controller's performance is very similar to the PID with tracking. However, it has slightly less error. This result is significant because while the FPID controller did not outperform the other two controllers in the step response, it manages to perform very well to the sin wave.

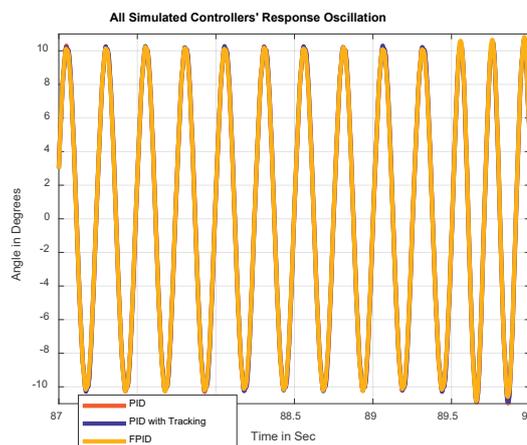


Figure 23 Comparison of all simulated controllers' response on sine wave

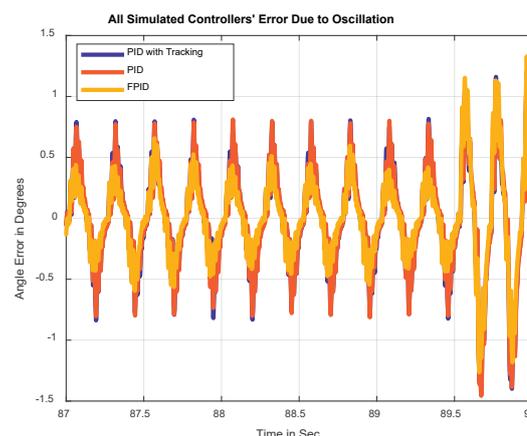


Figure 24 Comparison of simulated controllers' error on sine wave

### 5.2.5 All Physical Controllers' Response to Sine Wave

Fig. 25 shows the comparative response of all the physical controllers. Fig. 26 shows the comparative error of all the controllers. As it can be deduced from both figures, similar to the simulation, the FPID controller has the least error and manages to maintain it. On the other hand, the PID with tracking controller has second least error unlike the simulated. The PID controller has the highest error despite performing very well in step response.

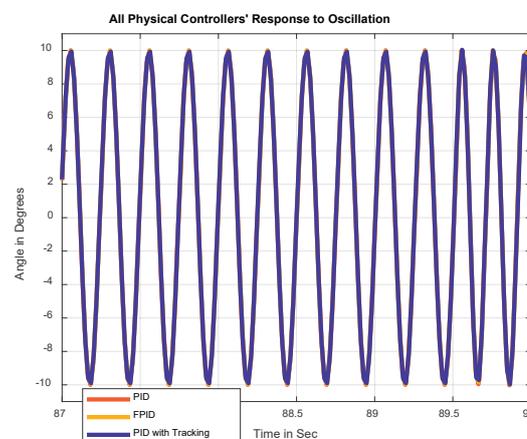


Figure 25 Comparison of all physical controllers' response on sine wave

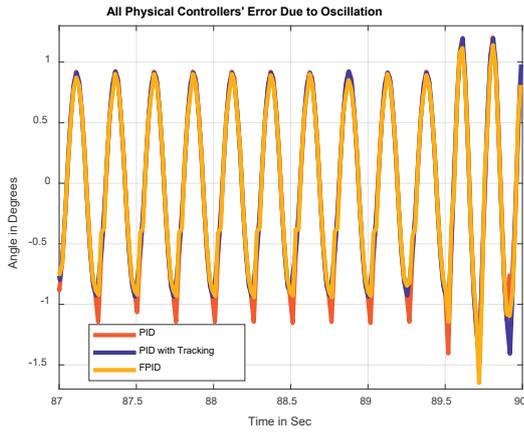


Figure 26 Comparison of all physical controllers' error on sine wave

### 5.3 Simulated vs Physical

As it can be seen from the result of section 5.1.1 through section 5.2.5, the performance of the physical canard is very similar to the performance of the simulated canard. This leads to the conclusion that the Simscape modelling technique is accurate.

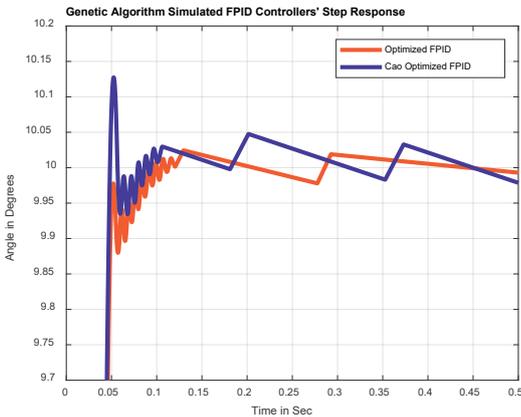


Figure 27 Genetic algorithm simulated controllers' step response

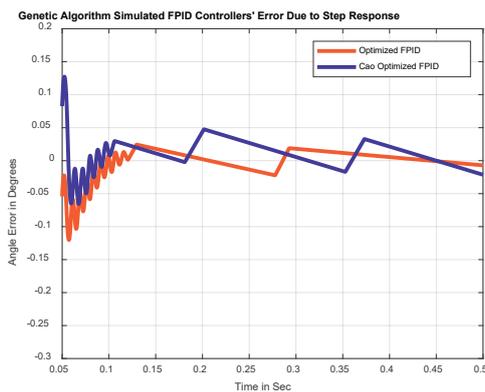


Figure 28 Genetic algorithm simulated controllers' error due to step response

### 5.4 Genetic Algorithm Optimization Comparison

The genetic algorithm optimization comparison is used to validate the improvements of the proposed optimization. The resultant gain from the proposed optimization is  $K_p = 19.3$ ,  $K_i = 2.5$ ,  $K_d = 0.4$ ,  $\lambda = 0.33$ ,  $\mu = 0.939$  and from Cao's method is  $K_p = 9.98$ ,  $K_i = 10.4$ ,  $K_d = 0.564$ ,  $\lambda = 0.11$ ,  $\mu = 0.877$ . The reason for the difference can be the genetic

algorithm, as there is no unique solution, and the differences in the optimization method.

#### 5.4.1 Genetic Algorithm Simulated Step Response

By evaluating Fig. 27 and Fig. 28, it can be seen that the proposed simulated optimization's FPID has less overshoot and error than Cao's optimized FPID.

#### 5.4.2 Genetic Algorithm Physical Step Response

By evaluating Fig. 29 and Fig. 30, it can be seen that the proposed physical optimization's FPID is faster. However, Cao's optimized FPID is smoother and has less overshoot.

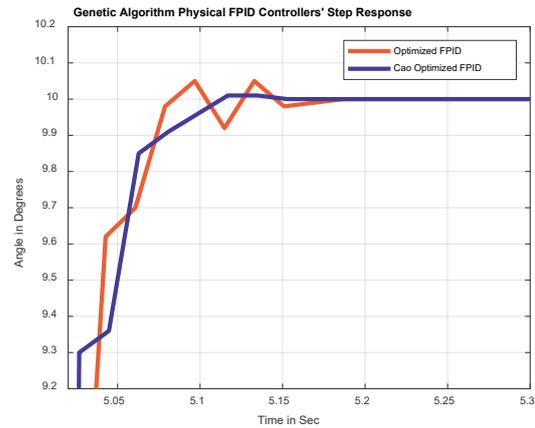


Figure 29 Genetic algorithm physical controllers' step response

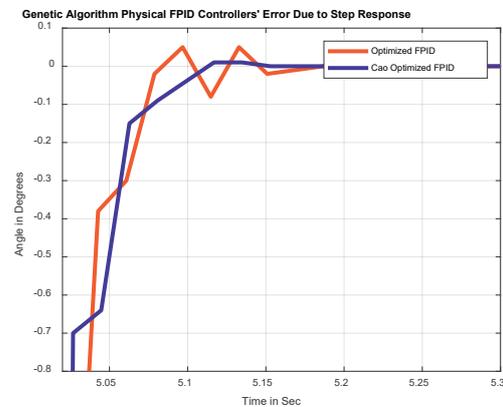


Figure 30 Genetic algorithm physical controllers' error due to step response

#### 5.4.3 Genetic Algorithm Simulated Error Due to Sine Wave

By evaluating Fig. 31, it can be seen that the proposed simulated optimization's FPID has less error than Cao's optimized FPID.

#### 5.4.4 Genetic Algorithm Physical Error Due to Sine Wave

By evaluating Fig. 32, it can be seen that the proposed physical optimization's FPID has less error than Cao's optimized FPID.

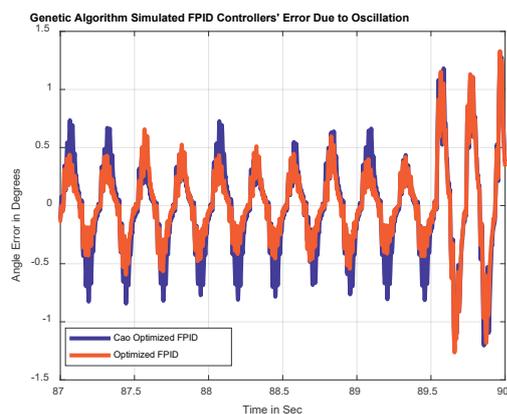


Figure 31 Genetic algorithm simulated controllers' error due to response to sine wave

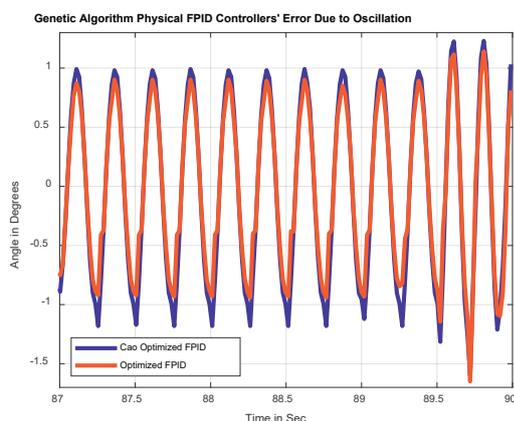


Figure 32 Genetic algorithm physical controllers' error due to response to sine wave

## 6 CONCLUSION

The three PID controllers successfully controlled the missile canard both in simulation and physically. The traditional PID controller resulted in the most stable performance. This is because it had the least oscillation. In addition, the PID with tracking feedback resulted in the fastest performance. That is because it reached the steady state in the least amount of time. Furthermore, the FPID controller has the best performance for oscillating dynamics. That is because it had the least overshooting and error during the response to sine wave. The results indicate that the right controller depends on the design criteria of the dynamic system.

In addition, the modelling technique proposed in this paper was successful. That is because the behavior of both simulated and physical missile canard was similar. In addition, the gain that was tuned using the simulation successfully controlled the physical canard. The accuracy of the modelling technique can be improved by a more accurate drawing of the missile canard using CAD software. Moreover, the mechanical tolerances and noise can be incorporated in future models to improve accuracy. Nevertheless, the Simscape model technique can be used as a viable modelling solution.

The refinement to the fitness function for the genetic algorithm was successful. That is because the gains obtained from tuning all three controllers resulted in performance with minimum error. In addition, the performance of the refined fitness tuning produces results

with less error and overshoot than Cao's genetic tuning algorithm. Hence, the genetic algorithm with refined fitness function is a feasible tuning solution.

## Acknowledgements

The professors of the Faculty of Mechanical Engineer in the University of Belgrade provided the technical guidance and hardware needed to complete the study. In addition, Emirates Advanced Research and Technology Holdings provided the financial support. Together, they made this study possible.

## 7 REFERENCES

- [1] Astrom, K. J. & Murray, R. M. (2009). *Feedback Systems*. Princeton University Press, New Jersey.
- [2] Khan, M. A., Todić, I., Miloš, M., Stefanović, Z., & Blagojević, Đ. (2010). Control of Electro-Mechanical Actuator for Aerospace Applications. *Strojarstvo*, 52(3), 303-313.
- [3] Bayat, F. & Jamshidi, A. (2012). Comparing the Performace of Optimal PID and Optimal Fractional-Order PID Controllers Applied to the Nonlinear Boost Converter. *The Fifth IFAC Symposium on Fractional Differentiation and Its Applications (FDA12)*, 14-17 May 2012, Hohai University, Nanjing, China.
- [4] Buzantowicz, W., Miernik, J., & Pietrasinski, J. (2015). Fin Actuator Transfer Function of the Anti-Airfract Missile. *XIX Międzynarodowa Szkoła Komputerowego Wspomagania Projektowania, Wytwarzania i Eksploatacji*, pp.83-92.
- [5] Precup, R. E. Preitl, S., Dragos, C. A., Radac, M. B., Stinean, A. I., Hedrea, E. L., & Roman, R. C. (2017). Automotive Applications of Evolving Takagi-Sugeno-Kang Fuzzy Models. *Facta Universitatis Mechanical Engineering*, 15(2), 231-244. <https://doi.org/10.22190/FUME170505011P>
- [6] Cojbasic, Z., Nikolic, V., Petrovic, E., Pavlovic, V., Tomic, M., Pavlovic, I., & Ciric, I. (2014). A Real Time Neural Network Based Finite Element Analysis Of Shell Structure. *Facta Universtitatis Mechanical Engineering*, 12(2), 149-155.
- [7] Wakasa, Y., Tanaka, K., & Nishimura, Y. (2012). Online Controller Tuning via FRIT and Recursive Least-Square. *IFAC Conference on Advances in PID Control*, Brescia, March 28-30.
- [8] Cao, J., Liang, J., & Cao, B. (2005). Optimization of Fractional Order PID Controller Based on Genetic Algorithm. *Proceedings of Fourth Int. Conference on Machine Learning and Cybernetics*. Guangzhou, 5688-5689.
- [9] Ogata, K. (2010). *Modern Control Engineering*. 5<sup>th</sup> ed. Prentice Hall, New Jersey.
- [10] PID Controller, Discrete PID Controller. <https://www.mathworks.com/help/simulink/slref/pidcontroller.html>
- [11] Bumpless Control Transfer between Manual and PID Control. <https://www.mathworks.com/help/simulink/examples/bumpless-control-transfer-between-manual-and-pid-control.html>.
- [12] Valerio, D. & Costa, J. S. (2013). *An Introduction to Fractional Control*. The Institution of Engineering and Technology, London.
- [13] Tepljakov, A., Petlenkov, E., & Belikov, J. (2011). FOMCON: Fractional-Order Modelling and Control Toolbox for MATLAB, Mixed Design of Integrated Circuits and Systems. *Proceedings of the 18<sup>th</sup> Int. Conference*, Gliwice.
- [14] Simscape Multibody User's Guide. The MathWorks, Inc., Massachusetts, 2017.

- [15] Mechanical Models. <https://www.mathworks.com/help/physmod/simscape/mechanical-models.html>  
[16] Electrical Models. <https://www.mathworks.com/help/physmod/simscape/electrical-models.html>

**Contact information:**

**Saif ALAMERI**, Eng. Msc.  
Faculty of Mechanical Engineering  
University of Belgrade  
Kraljice Marije 16, 11120 Belgrade, Serbia  
alameriss@gmail.com

**Dragan LAZIC**, PhD, Full Professor  
Faculty of Mechanical Engineering  
University of Belgrade  
Kraljice Marije 16, 11120 Belgrade, Serbia  
dlazic@mas.bg.ac.rs

**Milan RISTANOVIC**, PhD, Associate Professor  
Faculty of Mechanical Engineering  
University of Belgrade  
Kraljice Marije 16, 11120 Belgrade, Serbia  
mristanovic@mas.bg.ac.rs