

Application of fractional algorithms in the control of a robotic bird

Micael S. Couceiro, N.M. Fonseca Ferreira, J.A. Tenreiro Machado

ABSTRACT

In this paper, it is studied the dynamics of the robotic bird in terms of time response and robustness. It is analyzed the wing angle of attack and the velocity of the bird, the tail influence, the gliding flight and the flapping flight. The results are positive for the construction of flying robots. The development of computational simulation based on the dynamic of the robotic bird should allow testing strategies and different algorithms of control such as integer and fractional controllers.

Keywords: Robotic, Bird, Dynamic, Control, Fractional algorithms, Aerodynamics

1. Introduction

Life on Earth has appeared approximately at 2.5 billions of years, as a result of physicals and chemicals processes, and evolved towards today's existing species. One of those species is the bird that derived from a trunk of reptiles about 150 million years ago, at the same time of the first mammals. In the group of vertebrate animals, the birds are one of the most numerous, being found almost everywhere in the world. Birds have many similar characteristics to the reptiles but they are different from all the other animals because of their feathers [1] and unique flying characteristics [2]. A robot can be defined as a programmable, self-controlled device consisting of electronic, electrical, or mechanical units. More generally, it is a machine that functions in place of a living agent. Having as inspiration the behaviors of the animals, some work have been developed with the purpose of implementing similar robotic behaviors. Examples of some robot-animals already built are spiders [3] and snakes [4]. Other interesting works focus specific characteristics of animals such as the application of morphing materials [5]. Based on the knowledge stemming from the state of the art, our objective is to simulate and to construct a robot that resembles to a bird. Bearing these ideas in mind the paper is organized as follow. Section two presents the state of the art in the area. Sections three and four provide an overview of the physical structure and the kinematics of the bird, respectively. Sections five and six describe the bird dynamics and the flight process implemented through the proposed model. In section seven and eight it is developed a dynamical analysis and the action of control algorithms, respectively. Finally, section nine outlines the main conclusions.

2. State of the art

This work implements a system that includes a physical and dynamic model of a bird, in the perspective of using a real-time animated model taking aerodynamics into consideration [6]. In this model, a bird flies by the wing beat motion, while using its tail feathers. The trajectory is established by determined points in the space and by adjusting the bird's orientation and flapping such that it passes through these points in sequence. This allows the bird to fly along an arbitrary path. A method for producing realistic animations from numerical solutions is given for generic bird models with various levels of complexity [7]. The study describes the development of models, implemented in the analysis of flapping flight, balancing the scientific analysis and model-based animation. The results show numerical data and visual simulations able to produce realistic flapping flight with physical strong foundations. The aerodynamic coefficients of lift and drag are based on the blade-element theory. This method requires an input of the angle of attack, along with the key aerodynamic properties of the wing, in order to determine the lift and drag coefficient. The birds tail also plays an important aerodynamic role in mechanical flight power and flight performance [8]. Theory provides a conventional explanation for how bird's tail works. In [6] the influence of the tail and feathers is taken in consideration.

3. Physical structure

In order to visualize the behavior of the bird during the simulation we developed a 3D model in *AutoCAD* inspired in a seagull as can be seen in Fig. 1. Each adjacent part (with different colors) corresponds to individual elements connected through joints. For simplicity, the structure of the wings is defined in the sense of a human arm, using the terms arm and hand accordingly. The corresponding wing joints will be denoted the shoulder and wrist.

We can subdivide flight in two distinct types: quasi-steady and unsteady states. For large birds the flight can be approximated using quasi-steady state assumptions because their wings flap at lower frequency during cruising. This means that the wing tip speed is low compared to the flight speed. Thus large birds, such as eagles and seagulls, tend to have a soaring flight. Their wings behave closely to fixed wings. On the other hand, small birds and insects fly in an unsteady state regime[9] in which their wing tip speed is faster than their flight speed. The forces and flows around a flapping wing are still a challenge in fluid dynamics [10]. Some characteristics of real birds when considering a bird flight simulation can be found in Table 1.

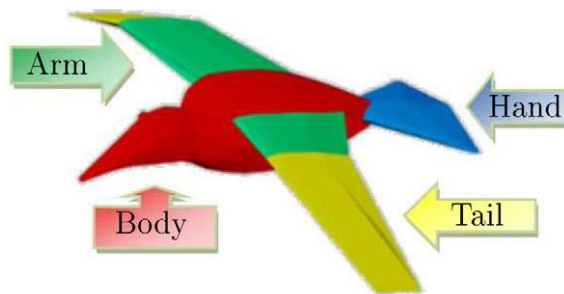


Fig. 1. 3D model of the robotic bird.

Table 1

Some characteristics of different

	Weight (kg)	Wing area (m ²)	Maximum velocity
Common tern	0.117	0.05	7.8
Black-headed gull	0.235	0.075	9
Seagull	0.374	0.115	9.2
Royal tern	0.480	0.108	10.7
Herring gull	0.960	0.181	11.7
Great skua	1.378	0.214	12.9
Great backed-backed gull	1.959	0.272	13.6
Sooty albatross	2.857	0.34	14.7
Wandering albatross	8.878	0.62	19.2

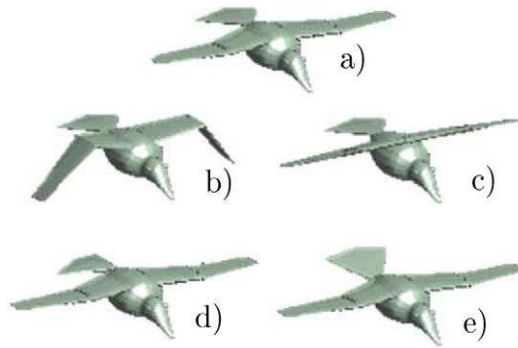


Fig. 2. (a) Bird geometry, (b) wing flapping, (c) wing twisting, (d) tail twisting, (e) tail bending.

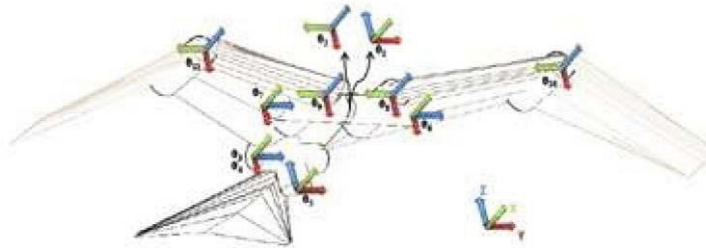


Fig. 3. Kinematic structure of the system.

4. Kinematics

With the kinematic model, we analyzed the bird flight movement and its behavior in different states such as taking off, flying with twists and turns, and others. Through this study, we obtained valuable specifications which helped choosing the initial mechanical design (Fig. 2).

The multi-link model is shown in Fig. 3. The number of joints has been reduced when compared with a real bird, but this mechanical structure gives a good mobility. The joints are distributed as follows: two in the shoulder, one in the wrist and two in the tail. Differently from all the others, the wrist joint is not controlled. It is a mechanical spring mechanism that allows a movement of the wing similar to real birds. This structure provides a good mobility having a total of six controlled joints.

In order to implement the animation of the bird in MatLab it is followed the Denavit–Hartenberg (D–H) notation [11] to represent frame (joint) coordinates in the robot kinematic chain. The next equation represents the homogeneous transformation A_i , namely a matrix constituted by a product of four fundamental transformations $A_i \propto R_{hi} T_{di} R_{ai} T_{ai}$ where R_{hi} and T_{di} are, respectively, the matrix rotation and matrix translation in x -axis and R_{ai} and T_{ai} are, respectively, the matrix rotation and the matrix translation in z -axis. With a series of D–H matrix multiplications, the final result is a transformation matrix from a given frame to the initial frame.

5. Bird dynamics

The relative wind acting on a wing produces a certain amount of force which is called the total aerodynamic force. This force can be resolved into two components, called lift and drag (Fig. 4).

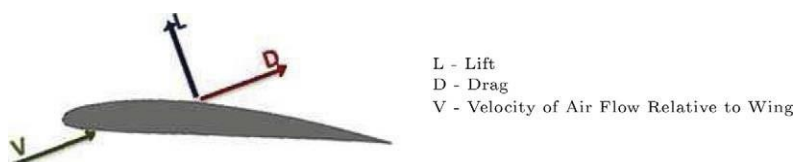


Fig. 4. Force acting on the wing.

The lift (L) is the component of aerodynamic force perpendicular to the relative wind and the drag (D) is the component of aerodynamic force parallel to the relative wind. Those components can be expressed by the following formulae:

$$L = \frac{1}{2} \rho v_{\infty}^2 S C_l(\alpha) \quad (1)$$

$$D = \frac{1}{2} \rho v_{\infty}^2 S C_d(\alpha) \quad (2)$$

The lift and drag on the wing depends on the wing area S , the density of air ρ , the velocity of the air flow relative to the wing v_{∞}^2 and the lift and drag coefficients C_l and C_d , respectively, expressed as functions of the angle of attack α . The lift and drag coefficients depend on the shape of the airfoil and will vary with changes in the angle of attack and other wing

trimmings. The characteristics of any particular airfoil section can conveniently be represented by graphs showing the amount of lift and drag obtained at various angles of attack, the lift–drag ratio, and the movement of the center of pressure. Similarly to [7] we adopted the blade-element theory representing the lift and drag coefficients C_l and C_d as functions of the angle of attack of the local wind.

$$C_l = C_{l_{\max}} \sin(2\alpha) \quad (3)$$

$$C_d = C_{d0} + C_{d_{\max}} \sin^2(\alpha) \quad (4)$$

We considered the following wing aerodynamics properties of maximum lift $C_{l_{\max}}$ and drag $C_{d_{\max}}$ coefficients and zero drag C_{d0} coefficient.

$$C_{l_{\max}} = 2 \quad (5)$$

$$C_{d0} = 0.05 \quad (6)$$

$$C_{d_{\max}} = 1 \quad (7)$$

6. Studying the flight

6.1. Gliding flight

Some birds take advantage of the air currents to remain aloft for long periods without flapping their wings. The bird's aerodynamic characteristics determine how far and for how long it can glide, and how successfully it can soar in moving air. Those aerodynamic characteristics can be optimized by the bird during the flight by changing the wing spans and wing areas. For optimal gliding a bird's wing must maximize lift and minimize drag. As a rule, the smaller the bird, the shorter the distance it can glide and the faster it sinks. During gliding the wings are stretched out stiffly. A good glider travels a long way horizontally with minimum loss of height, but eventually loses altitude due to the pull of gravity. The efficiency of a glider can be measured by calculating the angle between the track of its motion and the horizon. This angle depends not upon the weight of the bird but rather upon the forces of lift and drag, through wing shape has some influence.

6.2. Flapping flight

Aerodynamics involving flapping wings differs in many ways from conventional aerodynamics, but some conventional rules still apply. A conventional airplane uses a propeller for thrust and fixed wings for lift. An ornithopter's wing must provide both of these forces. The forces on the wing vary throughout the flapping cycle as we will see in the dynamical analysis. On the down stroke air is displaced in a downward and backward direction. On the upstroke, the situation is reversed being the area of the wing smaller than before in order to make a positive global lift. Therefore, for making the area of the wing smaller, birds uses different techniques such as manipulating the wings. To simplify we considered the area in the upstroke half the area in the down stroke. The science behind flapping flight is complex and research is still ongoing [12].

6.3. Tail influence

The precise use of the tail in flying birds has not been thoroughly documented [6]. The tail feathers are instrumental in stabilizing the flight, changing the direction of the forward movement, compensating for the lift force and acting as a brake

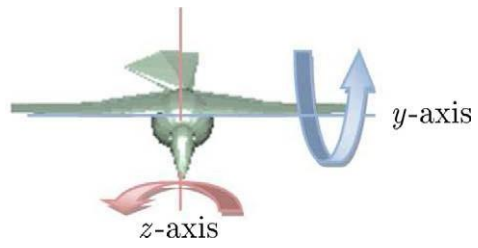


Fig. 5. Action of bird tail.

when the bird lands. We are using the tail in order to cause a drag force changing the moment of the bird and, consequently, producing a rotation around an axis equal to the axis of rotation of the tail. That is, if the tail is bending up, the bird will rotate around the same joint bending up too. If the tail bends up and twists right, for example (Fig. 5), the bird will then rotate around both the joints of the tail up and right. The angle of rotation of the tail is always relative to the movement of the bird.

7. Dynamical analysis

We have undertaken a dynamical analysis to test the validity of the system model. In order to easily change the parameters (e.g., wing area, weight) we built a computer program highlighting the fundamentals of robot mechanics and control.

The computer programs emphasize capabilities such as the 3D graphical simulation and the programming language giving some importance to mathematical aspects of modeling and control [13].

The project lead to the SIRB – simulation and implementation of a robotic bird (Fig. 6) program which was adopted as an educational tool in flying robotics birds [14].

We will start by showing some results of the bird dynamics during the gliding flight. In each simulation the wind has a constant velocity of $v \approx 5.0 \text{ m/s}$ against the movement of the bird that has an initial velocity of $v_0 \approx 4.0 \text{ m/s}$. We will change the weight m and the area of the wing S in order to analyze the bird dynamics. The initial parameters are a total weight of $m \approx 0.4 \text{ kg}$ and the wing an area of $S \approx 10^{-3} \text{ m}^2$.

7.1. Analyzing the gliding flight

Figs. 7–10 show the relation between the angle of attack α and the velocity of the bird v_x while gliding in a straight line. We verify that, for the bird to fly in a straight line without flapping its wings, he needs to change continuously the angle of attack to keep a vertical resulting force equal to zero. The angle of attack will then increase increasing the lift and the drag forces. An higher drag force results in the reduction of the velocity. This process stops when the velocity reaches zero, since we do not want the bird to be dragged by the wind. In Figs. 7 and 8, we increased the weight of the bird by $\Delta m \approx 0.1 \text{ kg}$ in each experiment. As can be seen, increasing the weight will requires a higher angle of attack in order to fly straight. The velocity does not change dramatically since the drag force does not increases significantly for angles of attack lower than 20° . Moreover, we verify that if the angle of attack increases the velocity decreases less than previously.

In Figs. 9 and 10, we change the area of the bird wings with increments $\Delta S \approx 0.1 \times 10^{-3} \text{ m}^2$ for each experiment. As expected, when increasing the area of the wing the bird is able to glide in a straight line with a smaller angle of attack. The velocity does not change significantly.

The second experiment (Figs. 11–15) show the horizontal v_x and vertical v_z velocities of the bird as well as the vertical distance obtained when the bird is gliding down a vertical distance of 5.0 m considering an angle of attack of 5.0° in both wings. Fig. 11 reveals that the bird oscillates when gliding down, without changing the angle of attack. As the bird goes down the horizontal velocity increases as well as the lift force. There is an instant where the resulting vertical force is zero and the

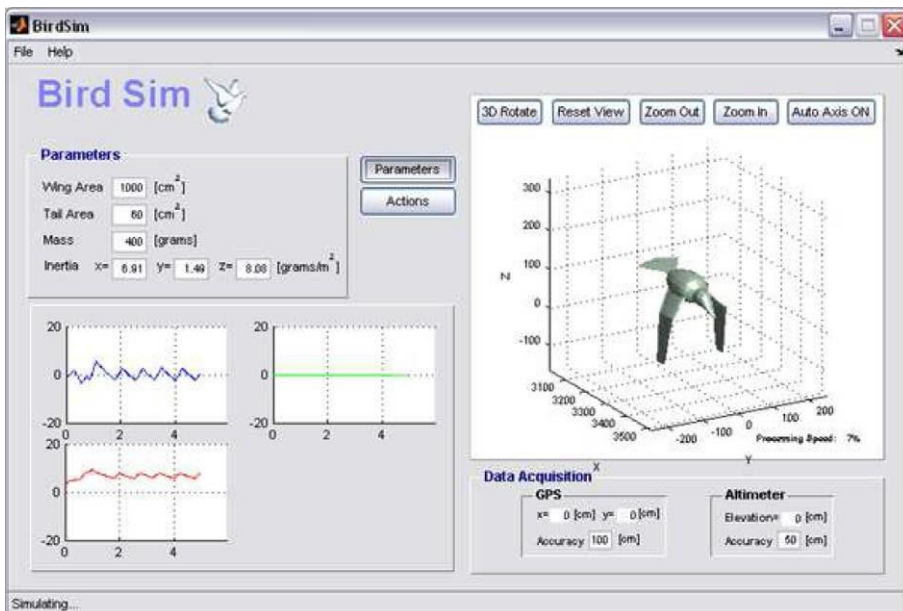


Fig. 6. Simulator SIRB.

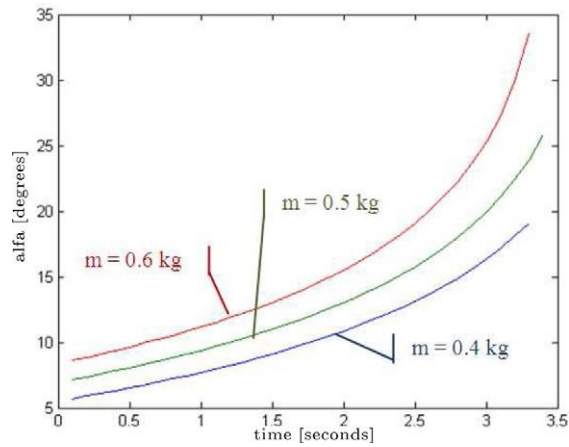


Fig. 7. Bird gliding straight – angle of attack versus time for different weights.

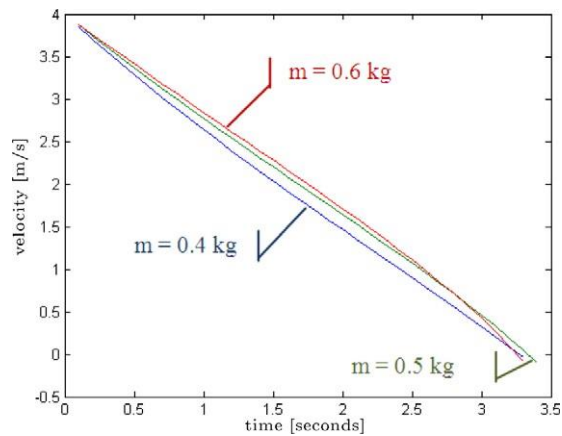


Fig. 8. Bird gliding straight – velocity versus time for different weights.

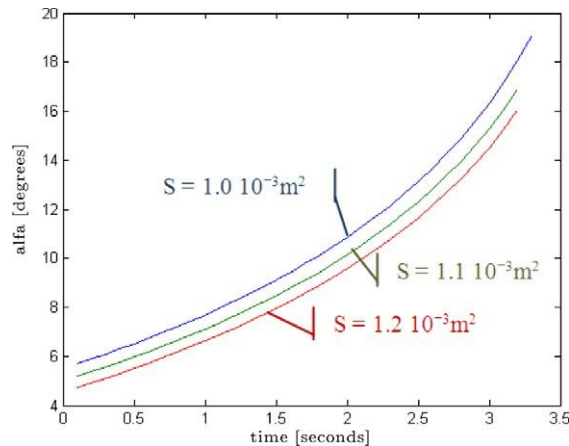


Fig. 9. Bird gliding straight – angle of attack versus time for different wing areas.

bird moves horizontally until it as not enough horizontal velocity. When the horizontal velocity decreases there is no more balance between the vertical forces and the bird goes down again.

Similarly to what we have done previously, we study the effect of varying the weight of the bird (Figs. 12 and 13). Since we are not changing the angle of attack, if we increase the weight the velocity will increase and the bird will reach its target

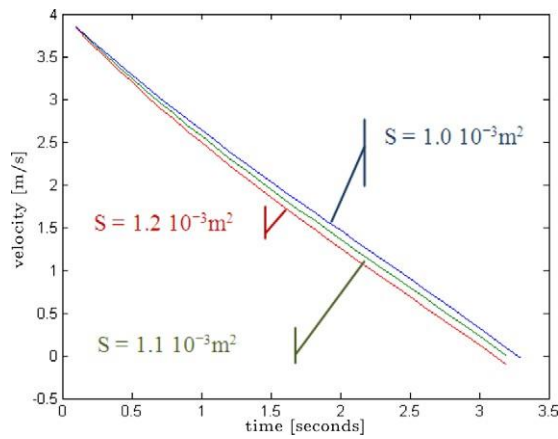


Fig. 10. Bird gliding straight – velocity versus time for different wing areas.

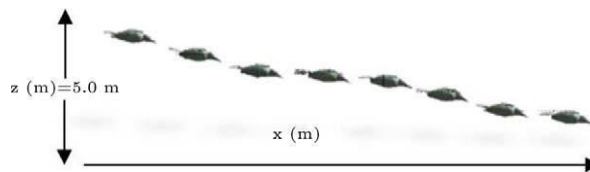


Fig. 11. Bird gliding down – sequence of images in the xz plane.

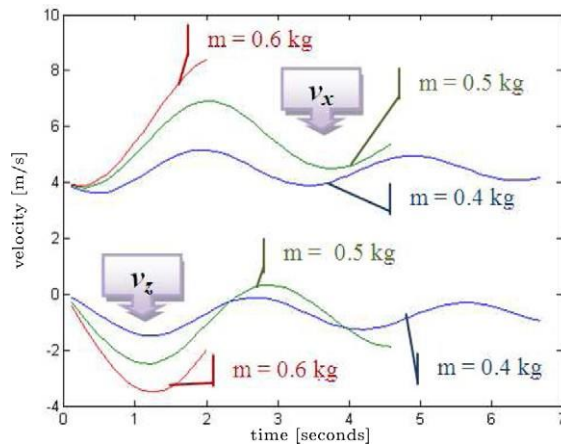


Fig. 12. Bird gliding down – velocities versus time for different weights.

faster (approximately 2.5 s faster for each increment $\Delta m \approx 0.1$ kg of weight). Repeating the flight experiment while increasing the wing area, the bird takes a longer time, traveling a longer distance to reach the 5.0 m, as one should expect (Fig. 14). Both velocities, particularly the horizontal one, have smaller oscillations, but where we can truly see the difference is in the vertical movement (Fig. 15) of the bird.

Increasing the wing area, the movement will be more close to a linear one flying for a long time without flapping the wings.

7.2. Analyzing the flapping flight

The analysis of the flapping flight is not as simple as for the case of the gliding flight. We have implemented three *PID* controllers in order to control the flapping velocity of the wings, based on the error of the horizontal and vertical velocities relative to a constant reference. Using the *PID* parameters tuned by trial and error and ignoring the *y*-axis velocity controller for now we obtained good results with the gains shown in Table 2.

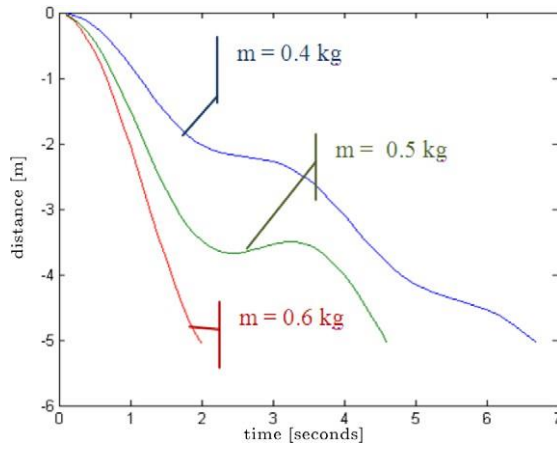


Fig. 13. Bird gliding down – vertical distance versus time for different weights.

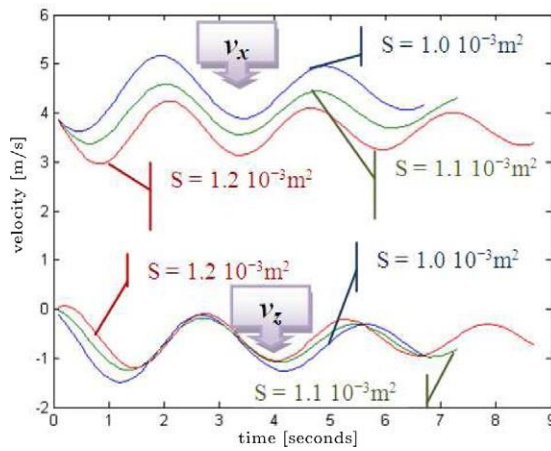


Fig. 14. Bird gliding down – velocities versus time for different wing areas.

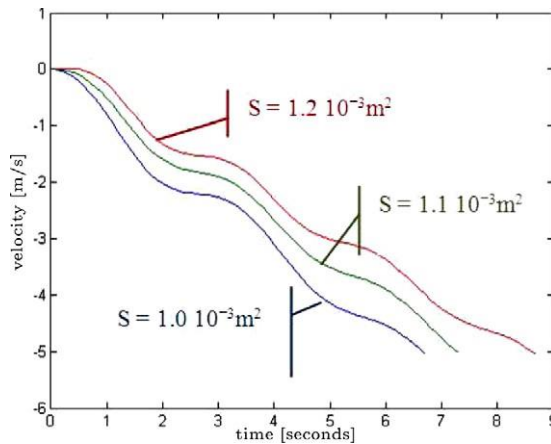


Fig. 15. Bird gliding down – vertical distance versus time for different wing areas.

We must note that our first priority is to fly in a straight line. The horizontal velocity may have some changes, if needed, in

order to keep the bird flying in a straight line. The *PID* controllers will be used in the following experiments concerning the analysis of the flapping flight.

Table 2

Controller's parameters obtained by trial.

	K_p	K_i	K_d
Horizontal velocity	1/6	1/6	0
Vertical velocity	6	6	0

For easing the comparison with the previous experiments, we will have a reference of $v_x \approx 4$ m/s in the horizontal velocity and $v_z \approx 0$ m/s in the vertical velocity (to fly straight). Following a similar line of thought of the gliding flight we change the weight and the wing area. Figs. 16 and 17 show how the velocities and vertical distance react while changing the bird weight.

With the PID controllers the bird flies in a quasi-steady state having a good response. Unfortunately, with $m \approx 0.6$ kg the bird is not able to keep the same altitude with the velocity of $v_x = 4.0$ m/s and, therefore, he needs to gain more horizontal velocity.

We can clearly see in Fig. 17 that the bird will loose altitude, something like 10^{-3} meters in 6.0 s, for a weight of $m \approx 0.5$ kg. In order to keep the same altitude the bird could be corrected after some time by continuously changing the angle of attack.

The next experiment shows what happens if the bird has larger wings. In Fig. 18 we can see that the bird does not need to travel with a velocity of $v_x \approx 4.0$ m/s to fly in a straight line. The bird reduces the wing beat velocity, requiring less energy, to keep the same altitude. In Figs. 19 and 20 we can see the bird trajectory.

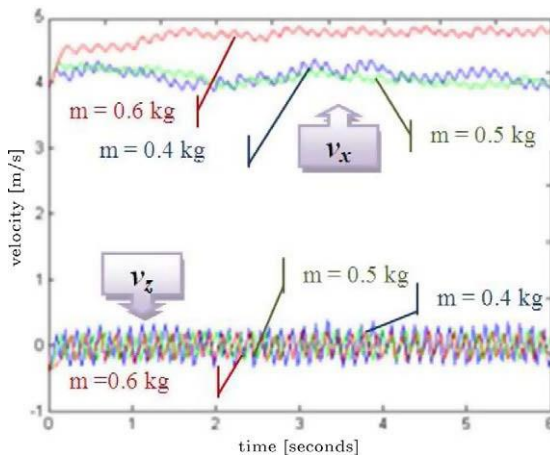
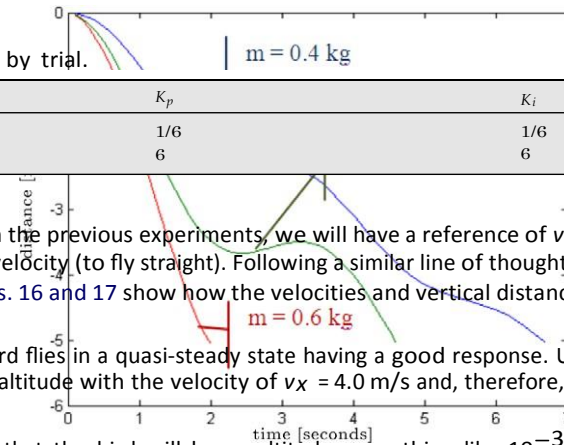


Fig. 16. Bird flapping straight – velocities versus time for different weights.

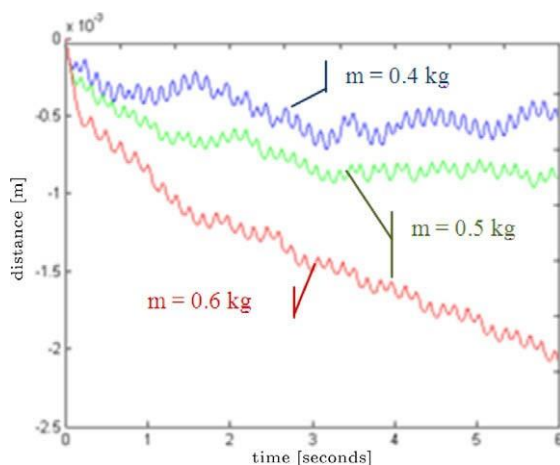


Fig. 17. Bird flapping straight – vertical distance versus time for different weights.

Fig. 18. Bird flapping straight – velocities versus time for different wing areas.

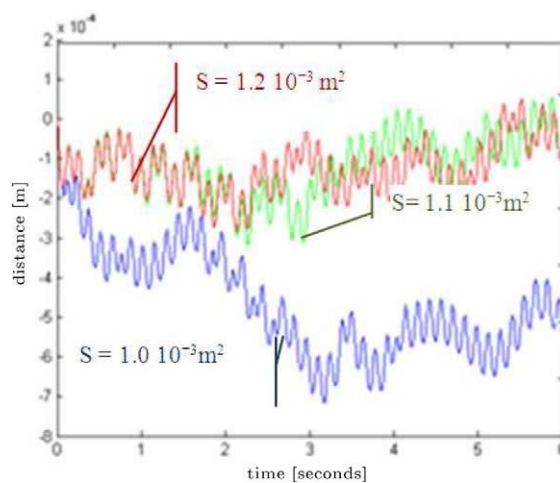


Fig. 19. Bird flapping straight – vertical distance versus time for different wing areas.



Fig. 20. Bird flapping straight – sequence of images in the xz plane.

We verify that, for a wing area larger than $S \geq 1.1 \times 10^3 \text{ m}^2$ the bird does not lose altitude even flying with a velocity lower than the reference. This means that larger birds are able to travel for greater distances with a quasi-steady state flight requiring less energy than little birds.

8. Controller performances

The previous analysis adopted classical *PID* controllers tuned by trial. In this section, we develop several experiments for comparing the performances of the *FO* (fractional order) *PID* algorithms [15]. *FO* controllers are algorithms whose dynamic behavior is described through differential equations of non integer order. Contrary to the classical *PID*, where we have three gains to adjust, the *FO PID*, also known as $PI^{\delta}D^{\beta}$, $0 < \delta < 1, 0 < \beta < 1$, has five tuning parameters, including the derivative and the integral orders to improve the design flexibility. The mathematical definition of a derivative of fractional order α

has been the subject of several different approaches. The Laplace Grünwald–Letnikov definition are perhaps the best suited for designing control algorithms.

$$D^\alpha[x(t)] = L^{-1}\{s^\alpha X(s)\} \quad (8)$$

$$D^\alpha[x(t)] = \lim_{k \rightarrow 0} \left[\frac{1}{h^\alpha} \sum_{k=1}^{\infty} \frac{(-1)^k \Gamma(\alpha+1)}{\Gamma(k+1)\Gamma(\alpha-k+1)} x(t-kh) \right] \quad (9)$$

where C is the gamma function and h is the time increment. The implementation of the $PI^K D^\alpha$ is given by:

$$G_c(s) = K \left(1 + \frac{1}{T_i s^i} + T_d s^d \right) \quad (10)$$

We adopt a 4th-order discrete-time Padé approximation ($a_j; b_j; c_j; d_j \in \mathbb{R}; k \in \mathbb{N}$):

$$G_p = KP \frac{a_0 z^k + a_1 z^{k-1} + \dots + a_k}{b_0 z^k + b_1 z^{k-1} + \dots + b_k} \quad (11)$$

where KP is the gain.

To tune the controllers' parameters we used a medium scale *Gradient Descent* method with 200 maximum iterations. To find a local minimum of a function of the position error using gradient descent, one takes steps proportional to the negative of the gradient (or the approximate gradient) of the function at the current point. Gradient descent is based on the observation that if the real-valued function $F(x)$ is defined and differentiable in a neighborhood of a point a , then $F(x)$ decreases fastest if one goes from a in the direction of the negative gradient of F at a . The first attempt to control our system will be changing the wing speed velocity, angle of attack and tail rotations accordingly with the cartesian position error. In order to

study the system dynamics, during the contact we apply, separately, rectangular pulses, at the references. The trajectory used to optimize the controllers is then a straight line flight with a velocity of $v_x \approx 3$ m/s during the first 20 s. The bird will then need to instantaneously achieve a velocity of $v_x \approx 5$ m/s. Finally, 20 s later, he will instantaneously reduce his velocity to $v_x \approx 3$ m/s again. We will repeat the same experiment going up and down with a vertical velocity $v_{zj} \approx 0.05$ m/s. In this kind of optimization it is unnecessary the use of a controller in the y -axis since there will be no movement in this axis; therefore we will ignore it for now. Let us first compare the PD and PD^α controllers. Under the last conditions we obtained the PD and PD^α controller parameters from Tables 3 and 4.

To analyze more clearly the dynamical response to the step perturbation we subtract the dynamic response without perturbation. Figs. 21–26 depict the system response under the action of the PD and PD^α algorithms.

Table 5 compares the time response characteristics of the integer and the fractional PD controllers, namely the percent overshoot PO , the rise time t_r , the peak time t_p and the settling time t_s .

We can clearly see the advantages of the FO algorithm that leads to a lower overshoot and a smaller settling time.

Similarly to the analysis made for the PD and PD^α controllers we will now compare the PID and $PI^K D^\alpha$ controllers (Tables 6 and 7).

Developing an identical analysis, we show in Figs. 27–32, Table 8 the system time response.

9. Conclusion

The functionalities presented in this work are implemented on the simulator. In the other hand, we obtained some results that are satisfactory proving the development of the kinematical and dynamic model can show us the behavior of the bird. The design methodology and the implementation can be deemed successful in this project. By obtaining a balance between physical modeling and the objective of animation, a strong advance in the system sophistication has been achieved. Despite all simplifications, the model is still quite complex, and further research needs to be conducted to explore additional abstractions. It is

Table 3
 PD and PD^α controller parameters for x -axis.

	K_{pX}	K_{dX}	I_x
PD	5	7	—
PD^α	8	3	0.9

Table 4
PD and *PD'* controller parameters for z-axis.

	K_{pz}	K_{dz}	I_z
<i>PD</i>	2	10	–
<i>PD'</i>	15	5	0.9

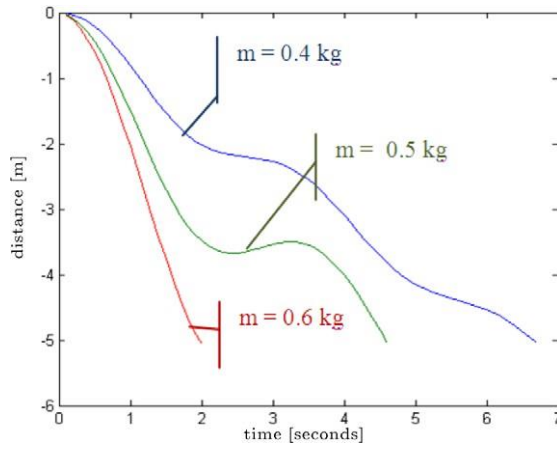


Fig. 21. Time response of the system under the action of the PD controller flying in a straight line.

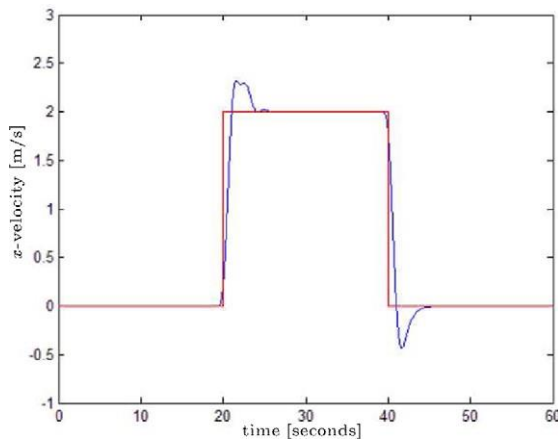


Fig. 22. Time response of the system under the action of the PD controller flying in a straight line.

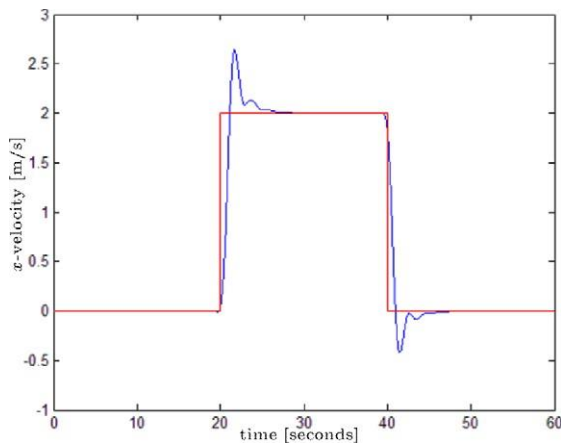


Fig. 23. Time response of the system under the action of the *PD* controller flying up.

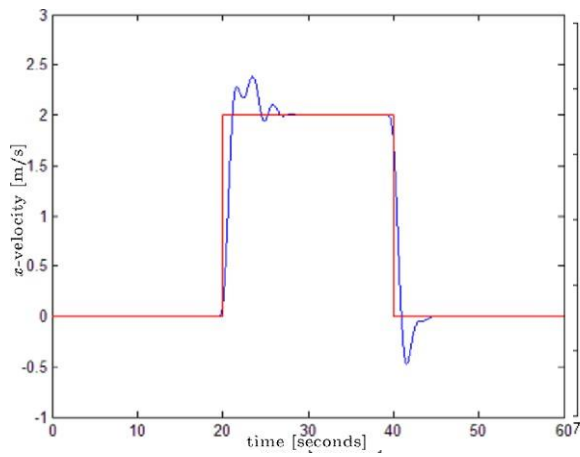


Fig. 24. Time response of the system under the action of the PD^I controller flying up.

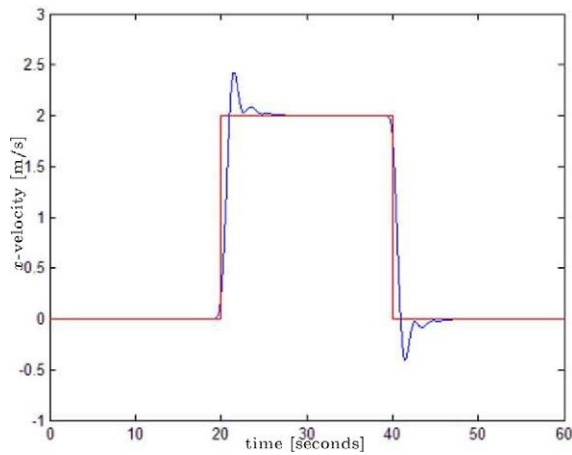


Fig. 25. Time response of the system under the action of the PD controller flying down.

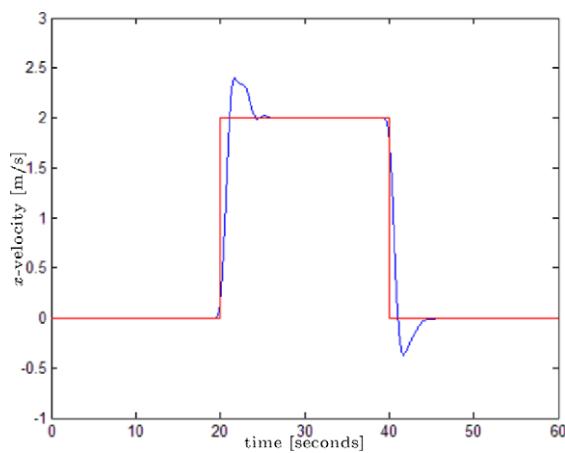


Fig. 26. Time response of the system under the action of the PD^I controller flying down.

Table 5
Time response parameters of the system under the action of the PD and PD^I controllers.

	PO (%)	t_r	t_D	t_s	
Straight	PD	27.52	21.08	21.64	28.6
	PD^I	16.01	21.04	21.60	25.7
Up	PD	31.88	21.12	21.70	28.9
	PD^I	19.19	21.16	23.52	28.5
Down	PD	21.43	21.00	21.52	28.0
	PD^I	19.67	21.12	21.74	25.9

Table 6
 PID and PI^kD^l controller parameters for x -axis.

	K_{pX}	K_{iX}	K_{dX}	l_X	k_X
PID	5	0.1	7	—	—
PI^kD^l	13	0.1	3	0.9	1

Table 7
 PID and PI^kD^l controller parameters for z -axis.

	K_{pZ}	K_{iZ}	K_{dZ}	l_Z	k_Z
PID	100	80	12	—	—
PI^kD^l	30	10	6	0.9	0.4

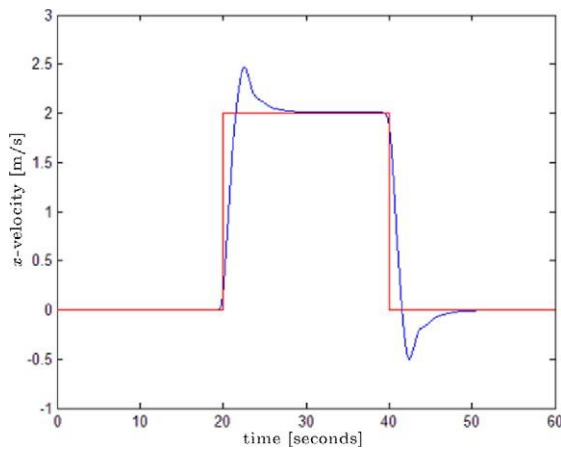


Fig. 27. Time response of the system under the action of the PID controller flying in a straight line.

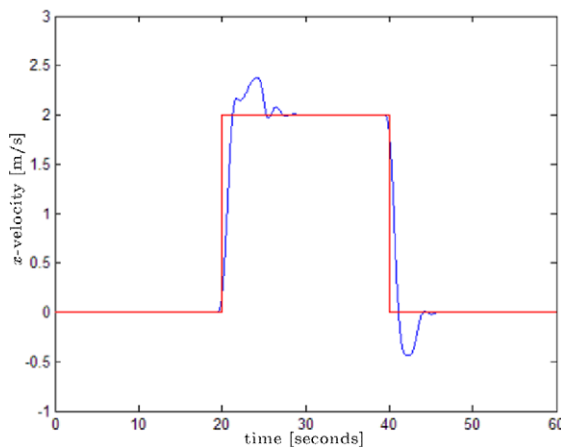


Fig. 28. Time response of the system under the action of the PI^kD^l controller flying in a straight line.

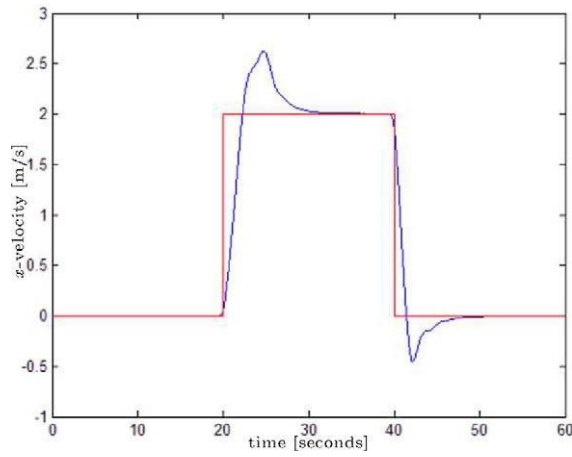


Fig. 29. Time response of the system under the action of the PID controller flying up.

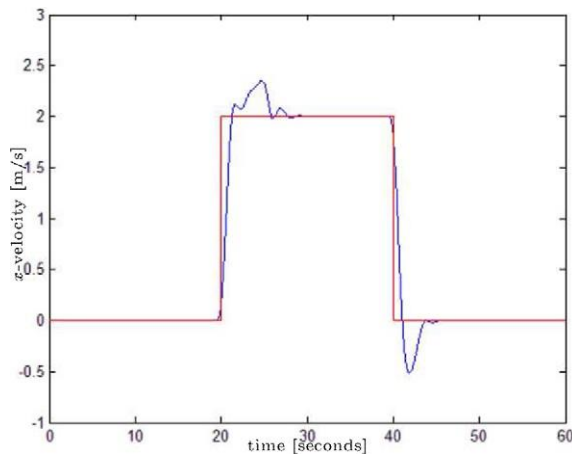


Fig. 30. Time response of the system under the action of the $PI^k D^l$ controller flying up.

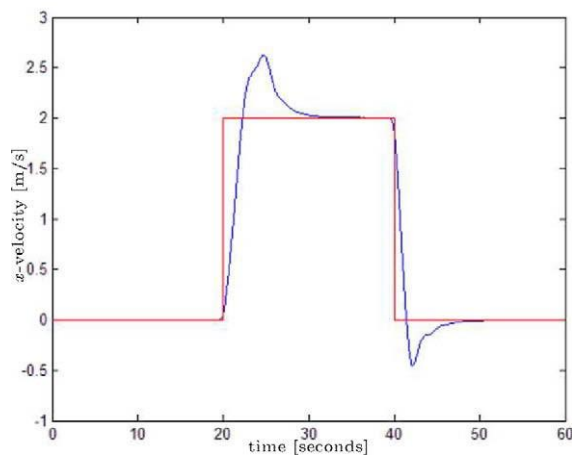


Fig. 31. Time response of the system under the action of the PID controller flying down.

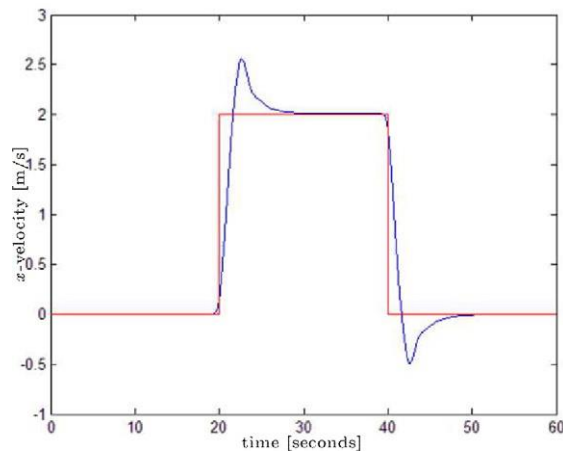


Fig. 32. Time response of the system under the action of the $PI^k D^l$ controller flying down.

Table 8

Time response parameters of the system under the action of the PID and $PI^k D^l$ controllers.

	PO (%)	t_r	t_n	t_s	
Straight	PID	22.29	21.58	22.58	21.0
	$PI^k D^l$	19.04	21.28	24.26	29.0
In	PID	21.12	22.26	24.74	29
	$PI^k D^l$	17.54	21.26	24.74	29.4
Down	PID	28.01	21.66	22.68	29.2
	$PI^k D^l$	21.94	21.24	23.98	26.9

possible to simulate all kind of closed loop actions like gliding, flapping wings, taking off, landing, following trajectories and others in a closed loop.

References

- [1] Cianchi M. Leonardo da Vinci's machines. Becocci Editore: Florence; 1988.
- [2] Dial Kenneth P, Randall Ross J, Dial Terry R. BioScience – what use is half a wing in the ecology and evolution of birds; 2006.
- [3] Vallidis Nick. A hexapod robot and novel training approach for artificial neural networks; 2001.
- [4] Spranklin Brent William. Design, analysis, and fabrication of a snake-inspired robot with a rectilinear gait; 2006. [5] Manzo Justin Edward. Analysis and design of a hyper-elliptical cambered span morphing aircraft wing; 2006.
- [6] Zhu Chaojiang, Muraoka Kazunobu, Kawabata Takeyuki, Cao Can, Fujimoto Tadahiro, Chiba Norishige. Real-time animation of bird flight based on aerodynamics; 2006.
- [7] Parslew Ben. Low order modelling of flapping wing aerodynamics for real-time model based animation of flapping flight; 2005.
- [8] Evans Matthew R, Rosén Mikael, Park Kirsty J, Hedenström Anders. How do birds' tails work? Delta-wing theory fails to predict tail shape during flight; 2001.
- [9] Colozza A. Fly like a bird. IEEE Spectr 2007;44(5):38–43.
- [10] Wang ZJ. Dissecting insect flight. Ann Rev Fluid Dyn 2005;37(1):183–210.
- [11] Denavit J, Hertenberg RS. Kinematics notation for lower-pair mechanism based on matrices; 1955.
- [12] Van Den Berg Coen, Rayner Jeremy MV. The moment of inertia of bird wings and the inertial power requirement for flapping flight; 1995.
- [13] Fonseca Ferreira NM, Tenreiro Machado JA. RobLib: an educational program for robotics SYROCO'00. In: IFAC symposium on robot control, vol. 1; 2000. p. 163–8.
- [14] Couceiro Micael S, Figueiredo Carlos M, Fonseca Ferreira NM, Tenreiro Machado JA. Simulation of a robotic bird. In: 3rd IFAC workshop on fractional differentiation and its applications, Ankara, Turkey, 05–07 November; 2008.
- [15] Fonseca Ferreira NM, Duarte Fernando B, Lima Miguel FM, Marcos Maria G, Tenreiro Machado JA. Application of fractional calculus in the dynamical analysis and control of mechanical manipulators. Fract Calc Appl Anal

