Stability region of a simplified multirotor motor–rotor model with time delay and fractional-order PD controller

Talar Sadalla, Dariusz Horla & Wojciech Giernacki

To cite this article: Talar Sadalla, Dariusz Horla & Wojciech Giernacki (2017) Stability region of a simplified multirotor motor–rotor model with time delay and fractional-order PD controller, Automatika, 58:4, 384-390, DOI: 10.1080/00051144.2018.1456098

To link to this article: https://doi.org/10.1080/00051144.2018.1456098

© 2018 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group

Published online: 04 May 2018.

Submit your article to this journal

Article views: 70

View Crossmark data
Stability region of a simplified multirotor motor–rotor model with time delay and fractional-order PD controller

Talar Sadalla*, Dariusz Horla and Wojciech Giernacki

Institute of Control, Robotics and Information Engineering, Faculty of Electrical Engineering, Poznan University of Technology, Poznan, Poland

ABSTRACT
The main aim of this paper is to present stability region analysis for a closed-loop system with the second-order model with a time delay and continuous-time fractional-order proportional-derivative (PD) controller. The model of the plant used in the paper approximates the dynamics of a simplified motor–rotor model of multirotor’s propulsion system. The controller tuning method is based on Hermite–Biehler and Pontryagin theorems. The tracking performance is also analysed in the paper by observing the integral of absolute error and integral of squared error indices. The presented results are expected to be useful in future when comparing simulation with experimental results.

1. Introduction
Fractional-order systems, can be considered as a generalization of integer-order ones [1]. In the last years, increased interest in fractional-order (FO) systems can be noted. From the mathematical point of view, research in this subject can still present new results, and one can easily present the advantages of using FO systems or non-integer systems in many specific areas, e.g. chemical analysis of aqueous solutions or quantum mechanical calculations. They also appear in control theory of dynamical systems where controlled plants can be of integer or non-integer order and the controller can also be of integer or non-integer order. This leads to FO differential equations and the necessity to solve these type of equations.

A non-integer order system is used where appropriate classical mathematical methods fail. It is possible to apply a FO system in control engineering theory [2,3] and in practical implementations [4–6], approximating it with appropriately high-order discrete-time systems. From the authors’ point of view, it is worth noting that to design e.g. a controller for a system with a time delay, a FO controller can be used, instead of applying classical methods, such as approximating the plant with a first-order inertia, see e.g. [7]. Classification of dynamic systems can be performed with respect to their orders, and the similar holds for the controllers, giving four possible combinations of integer- and FO-components in the closed-loop system [8]. Many control plants are fractional by nature, thus this approach seems most appealing and, in addition, it is often cited that a FO controller can always outperform an integer-order controller [8–10]. Of course, there is a problem with tuning these controllers, as well as with their implementation. Tuning can be performed offline, even by genetic or particle swarm optimization (PSO) algorithms [11], and implementation usually requires a number of past samples of signals stored in the controller’s memory, to perform suitable approximations of the FO controller operations.

In this paper, the FO calculus is used to analyse the stability region of a second-order plant with time delay and the use of fractional-order proportional-derivative (FOPD) controller. FO control is being currently and increasingly used in order to achieve better performance of control systems in comparison with classical control. Tuning of FO controllers is challenging, since in, e.g. the general framework of an FOPD controller there are three parameters to tune. Many approaches to tune FO controllers are available in the literature, nevertheless they present either simplifying approaches, such as the Ziegler–Nichols methods, or require an in-depth knowledge of the model of the plant, such as in the Bode approach. Other methods may require, e.g. genetic algorithms to be used, which cannot be easily applied due to their computational complexity in real-time regime.

According to Shah and Agashe [12], three main categories of approaches to controller-tuning can be distinguished, i.e. rule-based methods, analytical methods, numerical methods and, in addition, classical adaptive control methods. An increasing number of
papers report the applications of FO controllers. In [13], a FO controller design for a mini DC motor that gives a step response of the closed-loop system with overshoot independent of payload changes, considering analogue implementation of FO operators, using operational amplifiers is given. The paper [14] introduces two fractional-order PI (FOPD) controllers for a class of FO systems, with tuning constraints imposed on closed-loop systems, taking care of performance and robustness. Simulation and experimental results are presented in these papers. Experimental results are carried out in a hardware-in-the-loop system in dynamometer control system. The authors of [15] propose the use of FO controllers to reduce the sensitivity of the closed-loop system to parameter variations, with the performance compared by means of Matlab simulations with a system with a well-tuned proportional–integral (PI) controller for an AC motor system.

In this paper, the stabilizing solution to the control system is found (with stability areas described) by the use of FO calculus, FOPD controller and continuous-time plant model. The plant model originates from the model of a propulsion unit of a quadrotor, namely with AXI 2814/12 GOLD LINE brushless direct-current (BLDC) motor from Model Motors company with three-bladed propeller GWS-HD9050x3-SW 9x5” [16]. In our next work, it will enable us to obtain a stabilizing FOPD controller in a real-world system. The test stand presented in Figure 1 has been used to obtain the time characteristics presented in the further part of the text.

Electrical drive units of multirotor unmanned aerial vehicles (UAVs) have simple mechanical construction and possibilities hidden in their control. In addition, by using appropriate drive units [17] with proper dimensions, placement, orientation, and ensuring appropriate torque and thrust control, by controlling the rotational speeds of the UAV’s rotors, this hidden potential can be exploited [18,19]. The problem of rotational speed control of propellers is not frequently addressed, but by keeping the real rotational speed of a particular propeller in accordance with a reference value of this speed, full stability and complete control over the multirotor at minimum energy consumption can be achieved [20].

The need to control rotational speed originates from the fact that the true thrust forces and torques generated by the blades of the driving units can, in general, be different or can be other as specified by the manufacturer. This usually results from the precision of quality of the production process, improper fitting (backlash), natural wear and ageing of high-speed drives or due to possible damages caused to the structure of the blades due to contact with obstacles (falls, scratches, etc.). These differences are observed especially during stabilization phase of the UAV in flight, when all the units should theoretically have an equal rotational speed, and as a result, equal thrust forces and torques, to keep the robot in the air, holding it still, despite the weather conditions. Due to the characteristics of controlling a UAV, every additional correction of its position or orientation, resulting from lack of balance between the forces, is connected with an additional energy expense, limited by the battery capacity, and may result in reducing the time of flight. This is the reason the authors propose to introduce an additional feedback loop to track the rotational speed in a multirotor UAV control loop, to ensure the control system can reserve a set of rotational speed and thrust force values of the output of the driving units [21].

Nowadays, in increasingly, widely applicable and professional multirotor flying robots [22,23], there dominates a multilayer control architecture (see Figure 2). The first layer is an outer control loop of the robot position, and together with the second layer (middle control loop of robot orientation) is commonly used in various configurations, e.g. when using quaternion-based logic [24], which aims at eliminating the gimbal lock phenomenon. Every such implementation of a multilayer control system is aimed at increasing the precision of positioning and maintaining high precision of orientation of the robot in 3D space.

Rotational speed control of a propulsion unit, the innermost control loop, in multirotor flying robots is a part of the majority of small-sized UAVs. This control problem is addressed, e.g. [25]. The sample advantages of a rotational speed control have already been mentioned in the previous section. It is not a demanding problem, since the mathematical model of the system is fairly simple, and the electronic speed control module, responsible for driving the DC motor by appropriate pulse-width modulation (PWM) modulation does the task.

Introduction of the fastest internal control loop of rotational speed that would be easily implementable with the use of speed measurements gained from counting of impulses generated by the marker placed at the BLDC motor propeller, for each of the motors is appealing. The latter is done, as mentioned above, in order to improve the most important feature of the

Figure 1. Test stand of a real propulsion unit.
control system, i.e. stabilization of the robot by forcing counterbalance of thrust and rotational torque of the motors. To achieve this improvement, it is expected to propose an efficient control algorithm that is tuned to an adequate dynamics model (adequate model of the motor).

2. Statement of the problem

2.1. Model of the propulsion unit, simplified linear model of the plant

Models of propulsion units with BLDC motors (used in driving systems of multirotor flying robots, UAVs), as remarked, have been developed in great variety. Linear models with time delay are described in [16], and non-linear models in [26]. The linear model is used in the paper as it has been already proven that one can propose a FO controller (in general, in FO proportional–integral–derivative (PID) structure) that would result in better control quality [27]. At the Institute of Control, Robotics and Information Engineering, the designed multirotor flying robots have the following motor in the speed control loops: AXI 2814/12 GOLD LINE BLDC (see the Introduction). From the tests of a real motor–rotor on a test bench [27], it has been found that maximum thrust (approx. 19.1 N) of the tested propulsion unit is observed at 77% PWM, with 8893 rotations per minute (RPM), and with a maximum speed of 9039 RPM with 75% PWM. In Figure 3 (from http://part.put.poznan.pl), the installation place of the real propulsion unit is presented.

The step response of the real propulsion unit has been presented in Figure 4 together with a proposition of a simplified model, namely the second-order inertia model with a time delay, where the unity gain corresponds to maximum useful rotational speed of the propulsion unit, namely approx. 9550 RPM.

Since mismodelling errors are inherently connected with assuming any models, or the dynamics of the systems considered might be more complex that it is thought of, a FO controller is proposed as an alternative to integer-order controller, to verify its applicability in this field. From previous research carried out by the authors, concerning velocity control of a servo system, it has turned out that the FOPI controller enabled obtaining better control quality in comparison with an integer-order controller [28]. The expected improvement in control quality should be a good reason to take a possible increase in complexity of the controller into account. This increase is connected to the approximation of FO operations by their integer-order discrete-time representatives, which is not a problem, as this is not a computationally demanding task, and can be easily encoded into the microcontroller driving the UAV.

In a further part of the paper, a simplified model of the plant is described by the transfer function:

\[
G(s) = \frac{b_0}{s^2 + a_1 s + a_0} e^{-sT_0},
\]  

where

\[
b_0 = K, \quad a_1 = 2T, \quad a_0 = T^2.
\]
It is assumed that all coefficients are known, where $K$ is a gain of the model, $T$ is a time constant and $T_0$ is a time delay [7]. The selected approximation results from the previous research of the authors [29,30] and rises naturally from observation of second-order-like behaviour of the step response of a propulsion unit. The model is simple enough to enable conducting the calculations, and complex enough to approximate the above-mentioned response.

Since the values of inertia time and delay time are only estimates, and the selected model has been shown by time analysis to be valid, the rest of the paper will be based on the parameters’ values rounded to the nearest integer. This part of the paper is devoted not to experimental validation now, but to presenting the possibility to both evaluate the performance of the system and to describe the range of controller parameters that yields a stable closed-loop system. The same approach could be applied to the obtained estimates of model parameters in the proximity of the rounded values, but would not change the presented methodology.

2.2. Fractional-order PD controller

The fractional PD$^\mu$ controller is described by the transfer function:

$$ C(s) = K_p + K_d s^\mu. \quad (3) $$

The block diagram of such a control system is presented in Figure 5.

3. Shaping the desired closed-loop response

3.1. Time delay vs. stability

The main problem in time-delayed systems is that the transport delay may cause instability. Time delays exist in every physical system which make control a difficult task. The problem is the inability to describe the system with a characteristic polynomial for the pole-placement method due to the presence of exponential function. Nevertheless, the time delay itself for small transport delays can by approximated with a exponential function. Nevertheless, the time delay itself for pole-placement method due to the presence of exponential function. Nevertheless, the time delay itself for pole-placement method due to the presence of exponential function. Nevertheless, the time delay itself for pole-placement method due to the presence of exponential function.

egin{equation}
\delta(s) = (b_0 K_p s^2 + b_0 K_d s^{\mu+\frac{\mu}{2}}) e^{-LT} + (s^2 + a_1 s + a_0). \quad (4)
\end{equation}

It is necessary to rewrite the $\delta(s)$ quasi-polynomial as

\begin{equation}
\delta^*(s) = b_0 K_p + b_0 K_d \delta^\mu + (s^2 + a_1 s + a_0) e^{Ls} = n(s) + d(s) e^{Ls}, \quad (5)
\end{equation}

where $\mu$ implies $\mu \leq 1$ and $L > 0$ must be verified.

Hermite–Biehler theorem

Let $\delta$ be a complex function of $\omega$ and be described by the equation:

\begin{equation}
\delta^*(j\omega) = \delta^*_r(\omega) + j\delta^*_i(\omega), \quad (6)
\end{equation}

where $\delta^*_r(\omega)$ and $\delta^*_i(\omega)$ represent the real- and imaginary-parts of $\delta^*(j\omega)$. The $\delta^*(j\omega)$ is stable when

1. $\delta^*_r(\omega)$ and $\delta^*_i(\omega)$ have only simple real and interlaced roots;
2. $\delta^*_r(\omega) - \delta^*_i(\omega) > 0$, for some $\omega = \sigma$ in $(-\infty, +\infty)$,

where $\delta^*_r(\omega)$ and $\delta^*_i(\omega)$ are the derivatives of $\delta^*_r(\omega)$ and $\delta^*_i(\omega)$ with respect to $\omega$.

An important step is to ensure that $\delta^*_r(\omega)$ and $\delta^*_i(\omega)$ have only real roots. This one can be achieved by applying the Pontryagin theorem.

Pontryagin theorem

Let $\delta^*(s)$ be described by the following equation assuming that $s = j\omega$ holds:

\begin{equation}
\delta^*(j\omega) = \delta^*_r(\omega) + j\delta^*_i(\omega). \quad (7)
\end{equation}

To assure that $\delta^*_r(\omega) = 0$ and $\delta^*_i(\omega) = 0$ have only real roots, it must be assured that in intervals

\begin{equation}
-2\pi l + \eta \leq \omega \leq 2\pi l + \eta l = 1, 2, 3, \ldots \quad (8)
\end{equation}

$\delta^*_r(\omega)$ and $\delta^*_i(\omega)$ have exactly $4N + M$ roots. For situations where the characteristic equation is of FO, the $\delta^*_r(\omega)$ and $\delta^*_i(\omega)$ must have $4\lceil |N| + 1 \rceil + |M| + 1$ roots, where $\lceil \cdot \rceil$ denotes the integer part, and $N$ and $M$ are taken as degrees of the numerator and denominator polynomials of the integer part, respectively. Proofs can be found in [1,31].

![Figure 5. Block diagram of the considered control system (second-order model with time delay) and fractional-order PD controller.](image-url)
Assuming that \( \mu = c/b \), Equation (5) can be rewritten as
\[
\delta^*(s) = \left[ b_0K_p\bar{s}^2 + b_0K_p + (s^2 + a_1s + a_0), e^{j\theta} \right].
\] (9)

By putting \( f = Ls \), the quasi-polynomial takes the form:
\[
\delta^*(f) = \left( b_0K_d\left( \frac{f}{L} \right)^2 + b_0K_p + \left( \frac{f}{L} \right)^2 + a_1\left( \frac{f}{L} \right) + a_0 \right) e^{j\theta}.
\] (10)

Then, for \( f = j\omega \), \( \delta^*(j\omega) \) becomes
\[
\delta^*(j\omega) = \left( b_0K_d\left( \frac{j\omega}{L} \right)^2 + b_0K_p + \left( \frac{j\omega}{L} \right)^2 + a_1\left( \frac{j\omega}{L} \right) + a_0 \right) e^{j\theta}.
\] (11)

Replacing \( e^{j\omega} \) with \( \cos(\omega) + j\sin(\omega) \), the real \( \delta^r(\omega) \) and imaginary \( \delta^i(\omega) \) parts become
\[
\delta^r(\omega) = \frac{b_0K_d}{L^2} \left| \text{Re}(j)\bar{\theta} \right|/|\omega\theta| + b_0K_p - \frac{\omega^2}{L^2}\cos(\omega) - \frac{\omega}{L}a_1\sin(\omega) + a_0\cos(\omega),
\] (12)
\[
\delta^i(\omega) = \frac{b_0K_d}{L^2} \left| \text{Im}(j)\bar{\theta} \right|/|\omega\theta| \text{sign}(\omega) - \frac{\omega^2}{L^2}\sin(\omega) - \frac{\omega}{L}a_1\cos(\omega) + a_0\sin(\omega).
\] (13)

The factor \( j^2 \) has \( b \) complex solutions, but the solution which must be considered in the characteristic equation is the common one with the smallest positive phase.

According to Pontryagin theorem, \( \delta^r(\omega) = 0 \) and \( \delta^i(\omega) = 0 \), the set of parameters \( K_p \) and \( K_d \) can be found and described by the equations:
\[
K_d = \left( \frac{\left( \frac{\omega^2}{L^2}\sin(\omega) + \frac{\omega}{L}a_1\cos(\omega) - a_0\sin(\omega) \right)\bar{\theta}^2/b}{\text{Im}(j)\bar{\theta}^2/|\omega\theta|\text{sign}(\omega)b_0} \right),
\] (14)
\[
K_p = -\frac{b_0K_d}{L^2}\left| \text{Re}(j)\bar{\theta} \right|/|\omega\theta| + \frac{\omega^2}{L^2}\cos(\omega) + \frac{\omega}{L}a_1\sin(\omega) - a_0\cos(\omega).
\] (15)

Since \( \delta^r(\omega) \) is an odd function, it always has a root at \( \omega = 0 \); thus, for \( \omega = \omega_0 = 0 \),
\[
\delta^r(\omega) = b_0K_p + a_0.
\] (16)

In order to verify the interlace property between the roots of \( \delta^r(\omega) \) and \( \delta^i(\omega) \), one must impose
\[
\delta^r(\omega) > 0 \Rightarrow b_0K_p + a_0 > 0 \Rightarrow K_p > \frac{-a_0}{b_0}.
\] (17)

The range of values of the parameters \( K_p \) and \( K_d \) that fulfil the conditions is shown in Section 5.4.

4. Stability criteria and tracking performance

4.1. Preliminaries

The stability criterion of the considered closed-loop system are based on two bounded-input, bounded-output (BIBO) conditions. The first one depends on the simulation process time (the algorithm verifies that the simulation time is the same as the desired time stated prior to simulation), and the second verifies for simulations terminated successfully if the consecutive peaks of the output signal \( y(t) \) are diverging, by calculating the differences between them. If the difference is increasing, the system is considered to be unstable, and in other situations the described system is assumed to be stable. To analyse the tracking performance, the authors used two standard quality criteria: integral of absolute error (IAE) and integral of squared error (ISE) [7].

4.2. Stability and performance analysis

The stability analysis is performed for range of \( \omega \) from \(-5 \) to \( 5 \) with step of 0.01. The ranges for parameters of the FOPD controller, namely \( K_d \) and \( K_p \), were calculated by using Equations (14) and (15). The \( \mu \) parameter varies from 0.01 to 1, and, in accordance with the condition mentioned above, \( \mu \leq 1 \). Simulation time was set to 100 sec and the input signal \( r(t) = 1(t) \).

Second-order plant model parameter is put to \( T = 1 \) and according to (2), \( a_0 = 1 \), and \( a_1 = 2 \). The time delay \( T_0 = 1 \) sec.

Figures 6–8, are examples of stability surface plots. The filled surface represents the region that has been obtained as the result of simulations. In addition, it has been truncated to obtain a stabilizing FOPD controller using Matlab curve-fitting tool and this area is limited by black dotted lines.

It can be noted that the range of parameters is relatively wide, but not all combinations assure a stable
closed-loop system. The stability region for the considered ranges is marked with black dotted line. It should be stressed that with increasing non-integer order of \( m \), the stability region expands too. In Figure 8, the classical proportional-derivative (PD) controller is presented, because \( m = 1 \), and this choice provides the widest range for parameters \( K_p \) and \( K_d \) that ensures stability of the closed-loop system. To verify the stability, the authors implemented the stability criteria mentioned in Section 4. The stability surface for the whole range of \( m \) and \( v \) is shown in Figure 9.

Apart from the stability, the tracking performance measured by IAE and ISE indexes is also presented, and analysed by means of simulations. In Figure 10, two performance index surfaces for IAE (left) and ISE (right) are presented, for stable closed-loop systems, with the white area corresponding to unstable closed-loop representation based on the stability criteria mentioned in Section 4.

5. Conclusions

The results presented in the paper are only few examples of the performance and stability analyses for FO systems with time delay, where the plant model mimics the behaviour of the propulsion unit model. It was observed during the analysis that with increasing non-integer order \( \mu \) of the PD controller, the stability region also increased. The FOPD controller gives the possibility to analyse the time-delayed systems without introducing approximations of the time-delay which reflect the real physical systems. Further research will be conducted on real machine, either as hardware-in-the-loop experiment or by logging data from real-time control systems to extend the results of this paper, as this paper is the introductory work. Nevertheless, it would also be interesting to use even the simulation result only to verify the impact of changes in parameters of the second-order model and its time delay on the closed-loop stability and performance. Similar results could also be obtained for controllers with \( I \) term, with applied anti-windup compensators in the FO controller.

Disclosure statement

No potential conflict of interest was reported by the authors.

References


[27] Giernacki W. Near to optimal design of $P^{f}D^{m}$ fractional-order speed controller (FOPID) for multirotor–motor–rotor simplified model. Arlington (TX); 2016 Jun 7–10.


