الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

École Nationale Polytechnique
Département d'Automatique

End of studies project thesis

Submitted in partial fulfillment of the requirements for the State Engineer Degree in Control Engineering

Fractional-order identification and control of a Twin Rotor MIMO System (TRMS)

Realized by:
Slimoune Mohamed El Amine
Publicly presented and defended on the 07th of July, 2022.

Jury members:

Mr. Tadjine	Mohamed	President
Mr. Boukhetala	Djamel	Supervisor
Mr. Boudjedir	Chems Eddine	Co-Supervisor
Mr. Achour	Hakim	Examiner
Mr. El Bey	Abdeldjalil	Invited

ENP 2022

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

École Nationale Polytechnique

Département d'Automatique

Laboratoire de Commande des Processus

End of studies project thesis

Submitted in partial fulfillment of the requirements for the State Engineer Degree in Control Engineering

Fractional-order identification and control of a Twin Rotor MIMO System (TRMS)

Realized by:
SLIMOUNE Mohamed El Amine
Publicly presented and defended on the 07th of July, 2022.

Jury members:

Mr. TADJINE	Mohamed	President
Mr. BOUKHETALA	Djamel	Supervisor
Mr. BOUDJEDIR	Chems Eddine	Co-Supervisor
Mr. ACHOUR	Hakim	Examiner
Mr. EL BEY	Abdeldjalil	Invited

ENP 2022

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

École Nationale Polytechnique
Département d'Automatique

Projet de fin d'études

pour l'obtention du diplôme d'Ingénieur d'État en Automatique

Identification et commande d'ordre fractionnaire d'un Twin Rotor MIMO System (TRMS)

Réalisé par :
Slimoune Mohamed El Amine
Présenté et soutenue publiquement le 07 juillet 2022.

Membres du jury :

M. Tadjine	Mohamed	Président
M. Boukhetala	Djamel	Encadreur
M. Boudjedir	Chems Eddine	Co-encadreur
M. Achour	Hakim	Examinateur
M. El Bey	Abdeldjalil	Invité

ENP 2022

ملخص

في هذا العمل يتم تقديم كيفية التحكم وتحديد النموذج الكسري لمقلد طائرة الهليكوبتر(TRMS) . بدءا بعرض مختلف النظريات الرياضية الكسرية وعلاقتها بنظرية التحكم حيث يتم مناقشة المفاهيم الضرورية في هذه الدراسة، ثم تقدير عدة نماذج رياضية لمقلد الطيران مع دراسة فيزيائية مفصلة للنظام. حددت ثلاث نماذج: نموذج خطي تام، نموذج خطي كسري ونموذج غير خطي تام. اضافة الى مناقشة كيفية تحديد نموذج رياضي غير خطي كسري للنظام. أخيرا تم تصميم متحكم من نوع(PID) تام وكسري للتحكم بسلوك النظام حيث استعملت خوارزمية استمثال عناصر السرب في هذا التصميم.

كلمات مفتاحية : تحديد النموذج الرياضي الكسري، التحكم الكسري، مقلد طائرة الهليكوبتر، زمية استمثال عناصر السرب.

Résumé

Le travail présenté dans ce mémoire concerne l'identification et la commande d'ordre fractionnaire d'un système Twin Rotor MIMO System (TRMS). Nous commençons par un rappel sur les concepts mathématiques liés à la théorie du calcul fractionnaire et son utilisation en commande des systèmes dynamiques. A travers une étude physique détaillée nous présentons le modèle mathématique analytique du système TRMS. Ensuite, nous avons proposé plusieurs modèles de représentation à base d'algorithmes d'identification, à savoir : un modèle linéaire d'ordre entier, un modèle linéaire d'ordre fractionnaire ainsi qu'un modèle non-linéaire d'ordre entier. Des commandes de type PID d'ordre entier et d'ordre fractionnaire sont, ensuite, synthétisées afin de stabilise le système. Afin d'améliorer les performances du système, les paramètres de l'ensemble des commande proposées sont optimisés par un algorithme méta heuristique de type (OEP).

Mots clés : Identification d'ordre fractionnaire, Commande d'ordre fractionnaire, simulateur de vol TRMS, optimisation par essaim de particules (OEP)

Abstract

In this project we present fractional-order identification and control of the Twin Rotor MIMO System. We start by introducing the necessary mathematical concepts of fractional calculus and then of fractional control theory in which we discuss all the necessary theoretical tools used in the study. Afterwards, we estimate several mathematical representations for our TRMS alongside with a detailed physical study of the system. Three different mathematical models are identified and compared: a linear integer-order model, a linear fractional-order model and a nonlinear integer-order model. We also discuss the possibility of establishing a nonlinear fractional-order model. Finally we design integer-order and fractional-order PID controllers to stabilize and control the TRMS. The design is made using particle swarm optimization to tune the controllers parameters.

Keywords : Fractional-order identification, Fractional-order control, TRMS, particle swarm optimization (PSO)

Dedication

“

To our dear parents, our families, and our friends.

”

- *Amine*

Acknowledgments

First of all, I thank God the Almighty for giving me the courage, the will and the patience to carry out this work.

I thank my parents, who supported me throughout my studies' journey.

I would like to express my gratitude to my supervisors, Mr. Djamel BOUKHETALA and Mr. Chems Eddine BOUDJEDIR, for their consistent support, guidance and clarifications during the running of this thesis.

I would also like to thank in advance Mr. Mohamed TADJINE and Mr. Hakim ACHOUR, for evaluating my project.

Contents

List of Figures

List of Tables

List of Abbreviations

TRMS	*Twin Rotor MIMO System*
MIMO	*Multiple Input Multiple Output*
PSO	*Particle Swarm Optimization*
PID	*Proportional Integral Derivative*
FOPID	*Fractional-Order Proportional Integral Derivative*
RLS	*Recursive Least squares*
TRR	*Trust region Reflective*
RSS	*Residual Sum of Squares*

Chapter 1

General Introduction

Fractional calculus is the branch of mathematics that considers the integrals and derivatives of any real positive order, meaning it generalizes the integro-differential operators for arbitrary orders.

The idea of fractional calculus is as old as traditional integer-order calculus, first mentioned by Leibnitz in the 17^{th} century [1] and later had its foundation built by Liouville in the 19^{th} century [2]. However, the applications of the theory remained limited until the 20^{th} century when it was introduced to different fields of science, more importantly control theory and signal processing where it had extended applications. In the last decades it became a subject of intensive works and interest due to its utility in these fields.

In control theory, fractional calculus bring so much value. First of all, real life systems show fractional behaviours that cannot be explained by integer-order dynamics. Secondly, some phenomena that did not have a good representation with integer-order models show perfect fitness with fractional-order models, such as the long memory effect and the hereditary phenomena. For instance, linear viscoelastic systems and heat transfer systems exhibit such phenomena [3]. Finally, fractional calculus controllers offer a huge flexibility in their design compared to integer-order ones and provide a good level of robustness.

A simple example where fractional behaviour appear in real life is shown when we represent the relation between the stress $\sigma(t)$ and the strain $\epsilon(t)$:
According to Newton's law:

$$\sigma(t) = \nu \frac{d\epsilon(t)}{dt} \tag{1.1}$$

And according to Hooke's law:

$$\sigma(t) = E\epsilon(t) \tag{1.2}$$

where:
ν: The viscosity
E: The modulus of elasticity
The stress is both an expression of the first derivative of the strain (1.1) and a direct function of the strain (1.2), this results in a transfer function between the strain and the stress with a behaviour between a proportional gain and a first order transfer function. This means that the true dynamics of the system are between the strain and its first derivative i.e. of a fractional-order of the strain.
Real life system usually have some degree of fractional-order dynamics. However, for most of them they are not shown explicitly through physical equations like in the case of the stress/strain dependency.

In this work, we presented a complete study of the fractional identification and control of a Twin Rotor MIMO System (TRMS) in which we identified mathematical representations of the TRMS both of fractional and integer orders. We then proceeded to control the system using both integer and fractional-order PID controllers.

This report consists of four chapters alongside with the general introduction and conclusion. The second chapter revolves around the fundamental notions in fractional calculus and their relation to control theory, a theoretical background necessary to proceed to application. The third chapter is about the study and identification of the TRMS, in which we estimated three different mathematical representations some of fractional orders and some of integer orders. The fourth chapter presents the process of designing integer and fractional-order PID controllers for the TRMS using metaheuristic algorithms.

Chapter 2

Fractional calculus theory

2.1 Fractional-order operator

We note by aD_t^α the fractional-order operator, where a and t are the limits of the operation, and α is a real number ($\alpha \in \mathbb{R}$) is the generalized fractional-order of the derivative and integral operations.
It is defined as [4]:

$$aD_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha}, \alpha > 0 \\ 1, \alpha = 0 \\ \int_a^t (d\tau^\alpha), \alpha < 0 \end{cases} \tag{2.1}$$

$\frac{d^\alpha}{dt^\alpha}$ and $\int_a^t (d\tau^\alpha)$ represent the α-order derivative and integral respectively.
Many definitions of the above operations were proposed in literature, we will mention in this paper the most used ones.

2.1.1 The Gamma function

Before we proceed to define the fractional-order integrals and derivatives we need to introduce the Gamma function, an essential function used in fractional calculus.
For any complex number z ($z \in \mathbb{C}$) the Gamma function is defined as followed [5]:

$$\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt, R(z) > 0 \tag{2.2}$$

This function clearly converges for any complex number z where $R(z) > 0$. Using integration by part we can prove that:

$$\Gamma(z+1) = z\Gamma(z) \tag{2.3}$$

So naturally if we consider z to be a natural number $z \in \mathbb{N}$, then we obtain:

$$\Gamma(z+1) = z\Gamma(z) = z(z-1)\Gamma(z-1) = z(z-1)(z-2)..1\Gamma(1) = z! \tag{2.4}$$

Furthermore, for any complex number z where $R(z) > 0$ the Gamma function preserves the following properties:

$$\begin{cases} \Gamma(z+1) = z\Gamma(z) \\ \Gamma(1) = 1 \end{cases} \tag{2.5}$$

(2.4) and (2.5) allow the Gamma function to generalize the factorial operator for non-integer values, a key-property to define non-integer integrals and derivatives.

2.1.2 Fractional integral

Let α be a complex number such that $R(\alpha) > 0$ and f a locally integrable function. We define the fractional α-order integral of f according to Riemann-Liouville as followed [6]:

$$aD_t^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_a^t (t-\tau)^{\alpha-1} f(\tau) d\tau \tag{2.6}$$

Where $\Gamma(\alpha)$ is the Gamma function. This definition is a direct result of the extension of Cauchy's formula for repeated integration [7] by generalizing the factorial operation using the Gamma function.

2.1.3 Fractional derivative

We can not simply define a fractional derivative by inverting the sign of α in Riemann-Liouville's integral (2.6), because the resulting formula will change the essential properties of the traditional integer-order derivatives, meaning the properties that define the concept of a derivative, as well as it may disturb the convergence of the integral.
To avoid those problems, many definitions of fractional-order derivatives were proposed in literature.
Let $\alpha \in \mathbb{R}^+$, $m \in \mathbb{N}$ such that $m-1 < \alpha < m$ and f an enough derivable function.

2.1.3.1 Riemann-Liouville's definition

One way to respect the above mentioned conditions is to derive the $(m-\alpha)^{th}$ fractional Riemann-Liouville's integral m times $(\frac{d^m}{dt^m}D^{m-\alpha})$. This will result in a $(-\alpha)$ fractional operation that agrees with the essential properties of derivatives and preserves the convergence of the integral.It represents the Riemann-Liouville's definition [4]:

$$aD_t^\alpha f(t) = \frac{1}{\Gamma(m-\alpha)}\frac{d^m}{dt^m}\int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-m+1}}d\tau \tag{2.7}$$

2.1.3.2 Caputo's definition

Another approach is to directly derive the function f m times instead of the $(m-\alpha)^{th}$ fractional Riemann-Liouville's integral. This corresponds to Caputo's definition [8]:

$$aD_t^\alpha f(t) = \frac{1}{\Gamma(m-\alpha)}\int_a^t \frac{f^{(m)}(\tau)}{(t-\tau)^{\alpha-m+1}}d\tau \tag{2.8}$$

One downward of this definition compared to the one proposed by Riemann-Liouville is that it requires the absolute integrability of the m^{th} derivative of the function $f(t)$.
However it is considered as the most suitable fractional derivative to model real life problems, because unlike Riemann-Louisville's derivative, when building a fractional-order differential equation Caputo's derivative brings the initial conditions $f^{(k)}(a)$, $k = 0, ..., m-1$ into the equation the same way integer-order derivatives do, so it brings much useful information about the system and its interactions with its past creating a memory effect.
By comparing Caputo's and Riemann-Liouville's derivatives we can extract the following relation:

$${}_a^{RL}D_t^\alpha f(t) = {}_a^{C}D_t^\alpha f(t) + \sum_{k=0}^{m-1}\frac{(t-a)^{k-\alpha}}{\Gamma(k-\alpha+1)}f^{(k)}(a) \tag{2.9}$$

This shows that the two definitions are equivalent for zero initial conditions, i.e. $f^{(k)}(a) = 0$ for $k = 0, ..., m-1$.

2.1.3.3 Gründwald-Leitnikov's definition

The Gründwald-Leitnikov's derivative is defined by extending the original definition of multiple derivatives [9] to fractional orders. It is defined as followed [6]:

$$aD_t^\alpha f(t) = \lim_{h\to 0} \frac{1}{h^\alpha} \sum_{j=0}^{k} (-1)^j \binom{\alpha}{j} f(kh - jh) \tag{2.10}$$

Where:

$$\binom{\alpha}{j} = \frac{\Gamma(\alpha+1)}{\Gamma(k+1)\Gamma(\alpha-k+1)} \tag{2.11}$$

Is the extension of the binomial coefficient for non-integer parameters, and h is the sampling step.

In addition, if f is a causal function and the step h is too small, we can approximate the α-order fractional derivitave in the instant $t_m = kh$:

$$aD_t^\alpha f(t) \approx \frac{1}{h^\alpha} \sum_{j=0}^{k} (-1)^j \binom{\alpha}{j} f(t_m - jh) \tag{2.12}$$

This derivative is important in control theory because it offers a simple form of the Laplace transform.

2.1.4 Laplace transform

The Laplace transform is an essential tool in control theory that allows us to commute between time and frequency domains. The Laplace transforms of the fractional-order operators are [5]:

- Fractional order integral:

$$L(aI_t^\alpha f(t)) = s^{-\alpha}.F(s) \tag{2.13}$$

- Riemann-Liouville's derivative:

$$L({}_a^{RL}D_t^\alpha f(t)) = s^\alpha.F(s) - \sum_{k=0}^{m-1} s^k [{}_a^{RL}D^{\alpha-k-1} f(t)]_{t=0} \tag{2.14}$$

- Caputo's derivative:

$$L({}_a^{C}D_t^\alpha f(t)) = s^\alpha.F(s) - \sum_{k=0}^{m-1} s^{\alpha-k-1} f^{(k)}(0) \tag{2.15}$$

- Gründwald-Leitnikov's derivative:

$$L({}_a^{GL}D_t^\alpha f(t)) = s^\alpha.F(s) \tag{2.16}$$

The advantages and disadvantages of the different definitions of the derivatives appear more clearly through the Laplace transforms, as we can observe that for the case of Riemann-Liouville's derivative the initial conditions are of a fractional-order and thus their physical interpretation is still considered a challenge in modern control theory, Caputo's derivative however brings physically well-known initial conditions. As for Gründwald-Leitnikov's derivative, its transform has a simple form, independent from initial conditions.

We can observe as well that for zero initial conditions Riemann-Liouville and Caputo's derivatives have the same simple Laplace transform as Gründwald-Leitnikov's.

2.1.5 Fractional order operator properties

Let $f(z)$ be a complex function of a general class that we can apply the fractional-order operator $D^{\alpha}f(z)$ on, and $\alpha \in \mathbb{C}$. D^{α} has the following properties [4]–[6], [10], [11]:

- If $f(z)$ is an analytical function of z, then $D^{\alpha}f(z)$ is an analytical function of z and α.
- if $\alpha \in \mathbb{N}$ then $D^{\alpha}f(z)$ ($D^{-\alpha}f(z)$ respectively) has the same result as the integer α-order derivative (integer α-order integral respectively).
- If $\alpha = 0$ then $D^{\alpha}f(z) = f(z)$.
- The operator $D^{\alpha}f(z)$ is linear, for f,g two complex functions and a,b two complex numbers:
$$D^{\alpha}(af(z) + bg(z)) = aD^{\alpha}f(z) + bD^{\alpha}g(z) \tag{2.17}$$
- The fractional order integrals hold the additive exponent law, for $\alpha,\beta \in \mathbb{R}^{+}$:
$$D^{-\alpha}(D^{-\beta}f(z)) = D^{-\beta}(D^{-\alpha}f(z)) = D^{-\alpha-\beta}f(z) \tag{2.18}$$
- The fractional-order derivatives hold the additive exponent law, for $\alpha_i \in \mathbb{R}^{+}$:
$$D^{-\alpha}f(z) = D^{-\alpha_1}D^{-\alpha_2}D^{-\alpha_3}...D^{-\alpha_n}f(z) \tag{2.19}$$
such that: $\alpha = \alpha_1 + \alpha_2 + ...\alpha_n$

2.2 Fractional-order linear time invariant-systems

Fractional-order linear time invariant-systems (FOLTIS) are linear systems that has a fractional-order mathematical representation that indirectly depends on time. In control theory the most used representations of the FOLTIS are the differential equations, transfer functions and state-space representations.
Let $u(t)$ and $y(t)$ represent the input and output of the system in the time domain respectively, and $U(s)$, $Y(s)$ their Laplace transforms. We define the following representations for fractional-order systems:

2.2.1 Fractional-order differential equations

Using the fractional-order operator we can establish fractional-order differential equations by generalizing the orders of the integer-order derivatives, this results in the following form:
$$a_n D^{\alpha_n}y(t) + ... + a_0 D^{\alpha_0}y(t) = b_m D^{\beta_m}y(t) + ... + b_0 D^{\beta_0}y(t) \tag{2.20}$$
If all the fractional orders are integer multiples of a base order $\alpha \in \mathbb{R}^{+}$ i.e. $\alpha_i,\beta_i = k\alpha$ where $k \in \mathbb{N}$,we call the system Commensurate-order system.
In addition if α is a positive rational number ,$\alpha \in \mathbb{Q}^{+}$ it is called a rational-order system.

2.2.2 Transfer function

Applying the Laplace transform on the fractional differential equation (2.20) allow us to extract the transfer function of the system:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^{\beta_m} + ... + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + ... + a_0 s^{\alpha_0}} \tag{2.21}$$

If the system is commensurable we can write [4]:

$$G(s) = \frac{\sum_{i=0}^{m} b_i s^{\alpha_i}}{\sum_{i=0}^{n} a_i s^{\alpha_i}} \tag{2.22}$$

2.2.3 State-space representation

The fractional-order state-space representation is defined as [11]:

$$\begin{cases} D^{\alpha}x(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \tag{2.23}$$

where: $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^r$ the input vector, $y \in \mathbb{R}^p$ the output vector and $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times r}$, $C \in \mathbb{R}^{n\times p}$ and $D \in \mathbb{R}^{p\times r}$ are the state, input output and feedthrough matrices respectively.
$\alpha = [\alpha_1, \alpha_2, ..., \alpha_n]$ represent the fractional orders.
The transfer function can be calculated from the state-space representation as followed [5]:

$$G(s) = C(s^{\alpha}I - A)^{-1}B + D \tag{2.24}$$

Where: I is the $n\times n$ identity matrix.

2.3 Stability

We consider the commensurable-order transfer function (2.22). We note by $p_i = \lambda_i^{\frac{1}{\alpha}}$ the poles of the system, such that λ_i are the eigenvalues of the matrix A in the state-space representation.
In order for linear invariant-time system to be stable, all the poles must have a negative real part, i.e. $|arg(p_i)| > \frac{\pi}{2}$, thus [12]:

$$|arg(\lambda_i)| > \alpha\frac{\pi}{2} \tag{2.25}$$

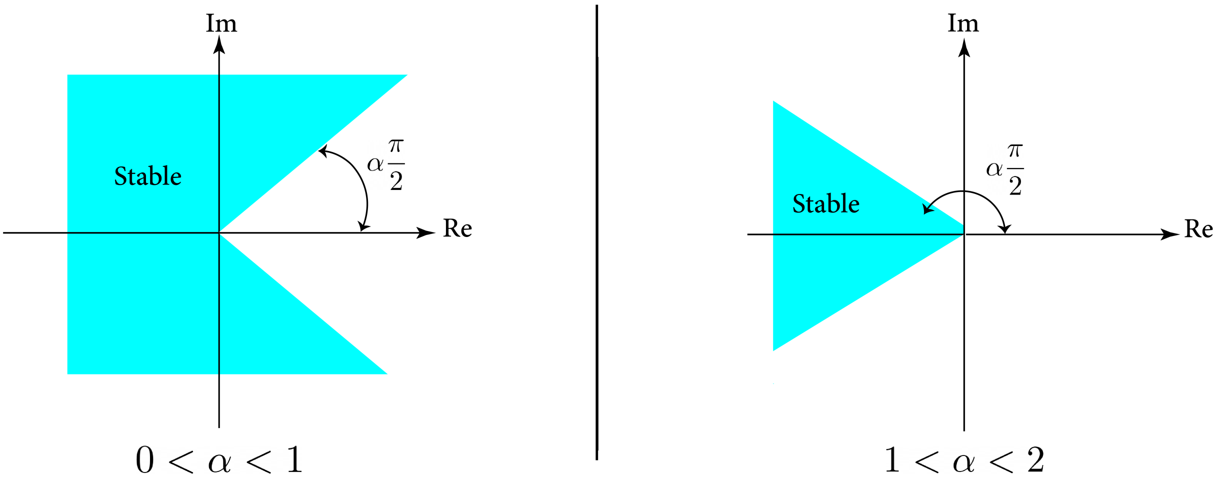

Figure 2.1: The stability regions in the complex plane for fractional-order systems

Note: The controllability and observability notions for fractional-order systems are the same as in integer-order systems. Therefore, the fractional-order system (2.23) is controllable (respectively observable) if the controllability matrix C (respectively the observability matrix O) has rank n. Where:

$$\begin{aligned} C &= [B\ AB\ A^2B\ ...\ A^{n-1}B] \\ O &= [C\ CA\ CA^2\ ...\ CA^{n-1}]^t \end{aligned} \tag{2.26}$$

2.4 Integer-order approximations of fractional-order systems

Control theory usually deals with integer-order systems and adapting the classical methods to fractional-order systems is a difficult process. This is due the irrational nature of their transfer functions and the infinite-dimensional state-space (they showed infinitely small and high time constants [13], [14]) which make the numerical approaches computationally hard to apply. To overcome this difficulty many methods to approximate the fractional-order systems by integer-order representations were proposed. The most famous of these methods are Charef approximation [15] and Oustaloup approximation [16]. In this paper we will use the Oustaloop approximation.

2.4.1 Oustaloup approximation

The Oustaloup method, also called the Oustaloup's recursive filter revolves around approximating the fractional operator s^α where $\alpha \in \mathbb{R}$ to an integer-order transfer function of a specific order in a limited frequency-band.
The $(2N+1)^{th}$-order Oustaloop approximation (where $N \in \mathbb{N}$) of the operator s^α in the a band of frequency of $[w_b, w_h]$ is defined as followed [16]:

$$s^\alpha \approx (\frac{w_u}{w_h})^\alpha \prod_{k=-N}^{k=N} \frac{(1+\frac{s}{w'_k})}{(1+\frac{s}{w_k})} \tag{2.27}$$

where:

$$\begin{aligned} w_u &= \sqrt{w_b.w_h} \\ w'_k &= w_b(\frac{w_h}{w_b})^{(k+N+0.5(1-\alpha))/(2N+1)} \\ w_k &= w_b(\frac{w_h}{w_b})^{(k+N+0.5(1+\alpha))/(2N+1)} \end{aligned} \tag{2.28}$$

2.4.2 Oustaloup filter's parameters

To understand how the parameters of the filter i.e. the order and the frequency-band influence the quality of the approximation, let us consider the following example of the fractional-order transfer function: $G(s) = \frac{1}{s^{0.5}}$, the different Oustaloup approximations of this function for different parameters are:

For an order of $N = 2$ in a frequency-band of $[0.01; 100]rad/s$:

$$\frac{1}{s^{0.5}} \approx \frac{0.1s^5 + 7.497s^4 + 76.85s^3 + 121.8s^2 + 29.85s + 1}{s^5 + 29.85s^4 + 121.8s^3 + 76.85s^2 + 7.497s + 0.1} \tag{2.29}$$

For an order of $N = 2$ in a frequency-band of $[0.001; 1000]rad/s$:

$$\frac{1}{s^{0.5}} \approx \frac{0.03162s^5 + 16.92s^4 + 537.1s^3 + 1072s^2 + 134.4s + 1}{s^5 + 134.4s^4 + 1072s^3 + 537.1s^2 + 16.92s + 0.03162} \tag{2.30}$$

For an order of $N = 3$ in a frequency-band of $[0.01; 100]rad/s$:

$$\frac{1}{s^{0.5}} \approx \frac{0.1s^7 + 9.834s^6 + 204.5s^5 + 1079s^4 + 1499s^3 + 548.7s^2 + 50.94s + 1}{s^7 + 50.94s^6 + 548.7s^5 + 1499s^4 + 1079s^3 + 204.5s^2 + 9.834s + 0.1} \tag{2.31}$$

The bode diagrams of $G(s)$ and its Oustaloop approximations are shown in figure 2.2.

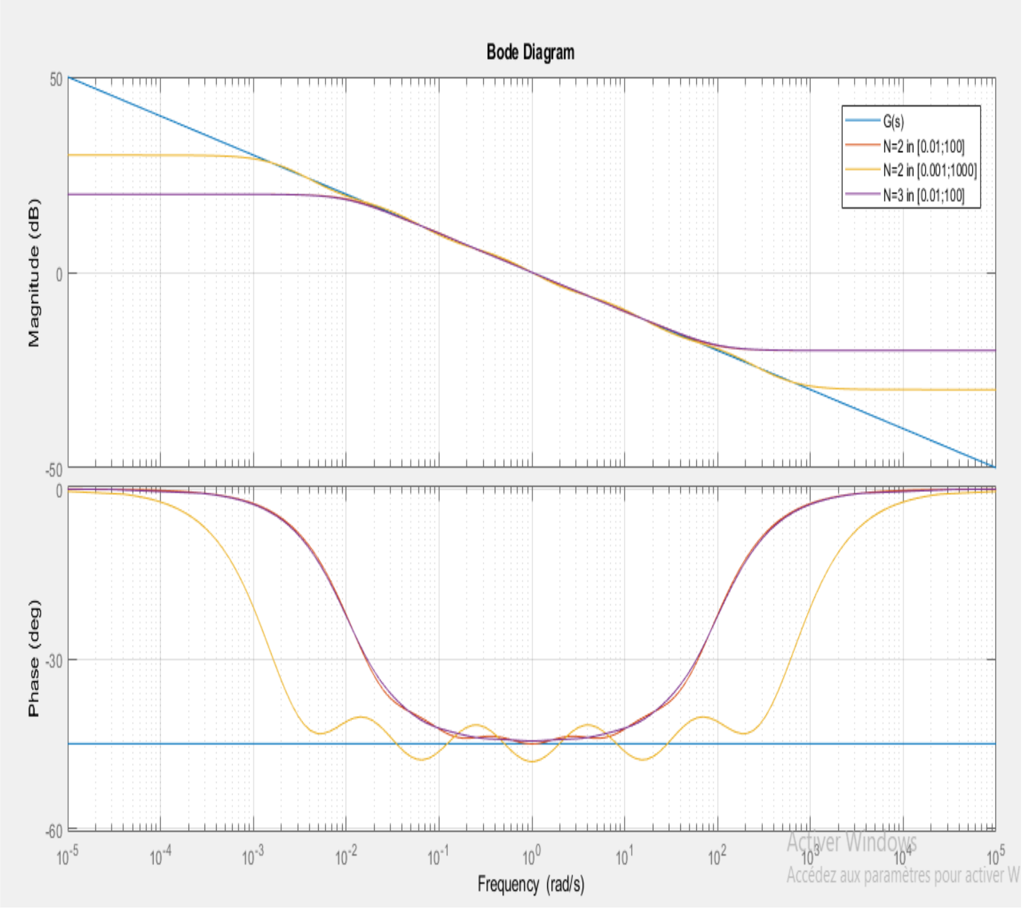

Figure 2.2: Bode diagrams of the function $G(s)$ and its Oustaloop approximations

From the above figure we can observe the following:

- In general, Oustaloup filter gives an accurate approximation of fractional-order transfer functions in the desired frequency-band.
- Outside Oustaloup frequency-band the approximated transfer function shows different magnitude and phase from the real function.
- The phase of the approximated function is more sensitive than the magnitude, where we observe that it does not correspond to the phase of the real function near the transition frequencies w_b and w_h.
- Enlarging the frequency-band causes approximation errors where the values of the phase and magnitude of the approximated function start oscillating around the values of the phase and magnitude of the real function.
- Increasing the order of the filter decreases the approximation errors in the frequency-band but increases computation time.

2.5 Fractional-order nonlinear systems

Fractional-order nonlinear systems can be described by a nonlinear version of the differential equation (2.20). The Laplace transform of such an equation does not result in a transfer function, therefore the state-space representation is the only remaining mathematical representation that can be deduced based on the physical properties. Such a state-space has the following form:

$$\begin{cases} D^{\alpha}x(t) = F(x(t), u(t)) \\ y(t) = G(x(t), u(t)) \end{cases} \tag{2.32}$$

where: F and G are nonlinear functions of $x(t)$ and $u(t)$.
Similarly to the integer-order case, the fractional-order linear control theory constitute the base tool of fractional-order nonlinear system study.

2.6 Conclusion

In this chapter, we provided the necessary mathematical background for fractional calculus which includes some fundamental definitions of the fractional-order integrals and derivatives with their main properties. We also briefly introduced fractional-order control theory and the necessary tools used in the study of fractional systems. Starting with the different mathematical representations of fractional-order linear time-invariant systems, then the analysis of the main characteristics of these systems like the stability. Finally, we introduced the Oustaloup recursive filter as an approximation method that helps to adapt classical control theory with fractional calculus.

Chapter 3

System identification

Introduction

We mean by system identification the process of determining a mathematical representation that describes the behaviour of the system. It is a tool to study it properly and does not necessarily need a physical interpretation.
In recent years, system identification developed drastically and came up with revolutionary models and techniques that made it faster, more accurate and richer in information. For instance, fractional-order models are currently one of the most promising representations. They proved to offer more insight on phenomena representation especially when it comes the hidden non-understandable effects and some particular properties like the memory effect [17], [18]. In this chapter we will discuss the estimation of some different integer and fractional-order models in both time and frequency domains for a highly non-linear MIMO system, the TRMS.

3.1 Control process

Our final goal in this study is to control our Twin Rotor MIMO system and get a desirable behavior, to do so, we need to pass through the following steps:

1. Technological study of our system.
2. Mathematical modeling.
3. Open-loop analysis (stability and dynamic behavior).
4. Command synthesis.
5. Closed-loop simulation.
6. Choice of the technology of command.
7. Practical tests.

To build the mathematical model of a system there are many approaches to determine the structure of the model:

3.1.1 White-box approach

This approach relies completely on the physical laws to model the system. Using physical properties we can find the mathematical expressions of the variables we want to control, or as in most cases, a differential equation that involves them. This method is a projection of real life phenomena so every single parameter has a physical interpretation, however it presents many limitations: first, most of the systems and phenomena are too complicated to be physically modeled, with a significant class of behaviors that cant be represented in the first place. Secondly, pure physical induction ignores many side-phenomena that affects the system's behavior as well as internal interactions, this causes a deficiency in the system's representation.

3.1.2 Black-box approach

In this approach we fix a mathematical model in which we include the input, output and the disturbances of the system with unknown parameters that will be identified using an optimization algorithm. Basically, the error between the real measured output value and the one predicted using the model is calculated, then an optimization criteria is fixed to minimize that error. A minimal error corresponds to a close predicted output to the real value, and thus a valid model. The disadvantages of this method consist in the absence of a systematic approach to fix the mathematical model which gives it some randomness, as well as the non-physical sense of the model's parameters.

3.1.3 Grey-box approach

This approach is a combination of the previous two. First, the structure of the mathematical model is built using physical laws, and then for any unknown parameter or complicated phenomena an optimization algorithm is used in the same way as in the black-box approach to numerically identify the remaining unknown parameters. This approach avoids a significant part of the inconveniences of the white and black-box approaches. The identification algorithm is:

Algorithm 1 Grey-box identification

```
1. Collect the measures (inputs and outputs)
2. Filter and normalize the collected data
3. Choose a model structure
4. Choose an optimization method
5. Identify the unknown parameters with the help of the optimization method
6. Validate the estimated model
```

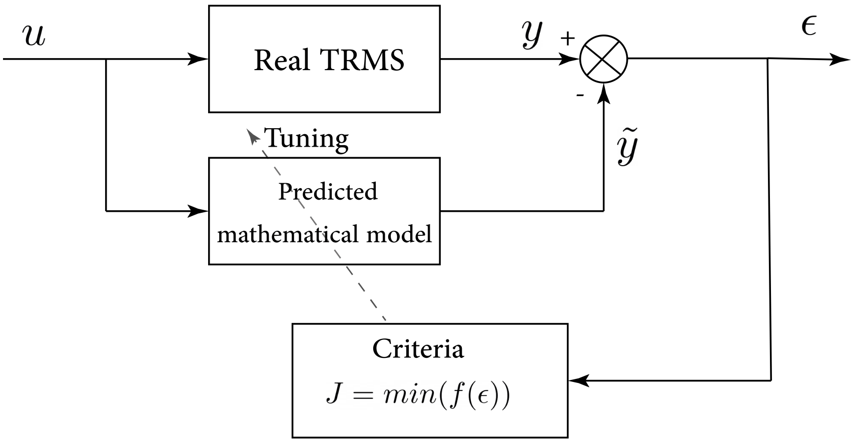

Figure 3.1: Estimation block

3.2 The Twin Rotor MIMO System:

The Twin Rotor MIMO System (TRMS) is a laboratory instrument developed by

Feedback Instruments Ltd. Its behavior is similar to that of a helicopter with some particular distinctions[19]: the location of the pivot in the TRMS is midway between the rotors instead of being in the main rotor head, and the lifting force comes from the rotational speed of the main rotor instead of the collective pitch control in the helicopter (constant rotational speed of the rotors with variable pitch angle of the tail rotor to generate lifting).
It consists of the following parts:

- A beam that rotates freely around the vertical and horizontal axis.
- A support tower fixed on a base that holds the rest of the parts. The base contains all the circuitry responsible for the processing of the I/O signals of the TRMS.
- A pivot that joints the beam and the support tower, allowing the beam to have the 2 degrees of freedom.
- Two rotors on the two extremities of the beam: A main rotor facing the horizontal plane and causes vertical movement, and a tail rotor facing the vertical plane and causes horizontal movement. Each of these rotors contain a DC-motor, a tachometer and propellers that are protected by a shield.
- a counterbalance beam attached to the pivot that holds a movable counterweight. Its purpose is to establish an equilibrium position and to diminish the vibrations of the system.
- An On and Off interface.

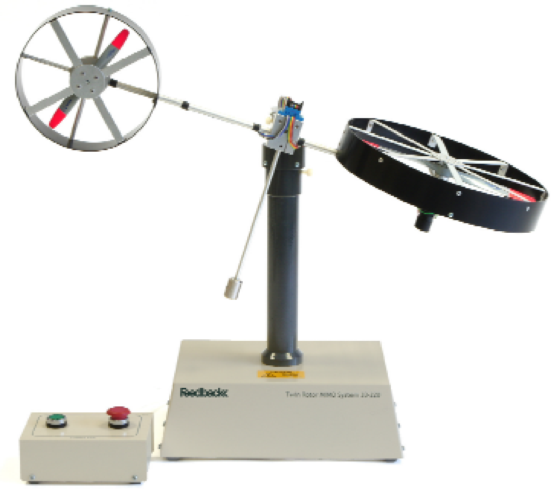

Figure 3.2: The Twin Rotor MIMO System [20]

In control, the TRMS represents a perfect example of a Multiple-Input Multiple-Output (MIMO) highly coupled system with non-linearities of superior order.
The measurable values of the TRMS (represent the outputs of the system) consist of the horizontal angle α_h called **azimuth** produced mainly by the propulsive force from the tail rotor, and the vertical angle α_v called **elevation** produced mainly by the propulsive force from the main rotor. These angles are measured by a sensor situated in the pivot part.
The controls of the system (i.e. the inputs of the system) are the voltage supplies of the motors u_v and u_h, they vary in a range of $[-2.5, +2.5]V$.

3.2.1 Physical study

Using Newton's second law of motion we can establish some differential equations that describe accurately the behaviour of our system.
Since the system has two degrees of freedom, we will consider studying the motion in each plane individually:

3.2.1.1 Vertical plane

Basically the majority of the dynamics in the vertical plane are caused by the main rotor and the gravity force pulling down the different parts of the setup. In addition, a centrifugal force is created when the beam moves vertically, accompanied with a friction force in the joint part. The torque of tail rotor has a slim effect on the dynamics of this plane, that we can simply neglect.
To make the process easier we can consider dividing the effect of the gravitational force on the different parts of the TRMS. The following figure illustrates all the forces that are applied in the vertical plane.

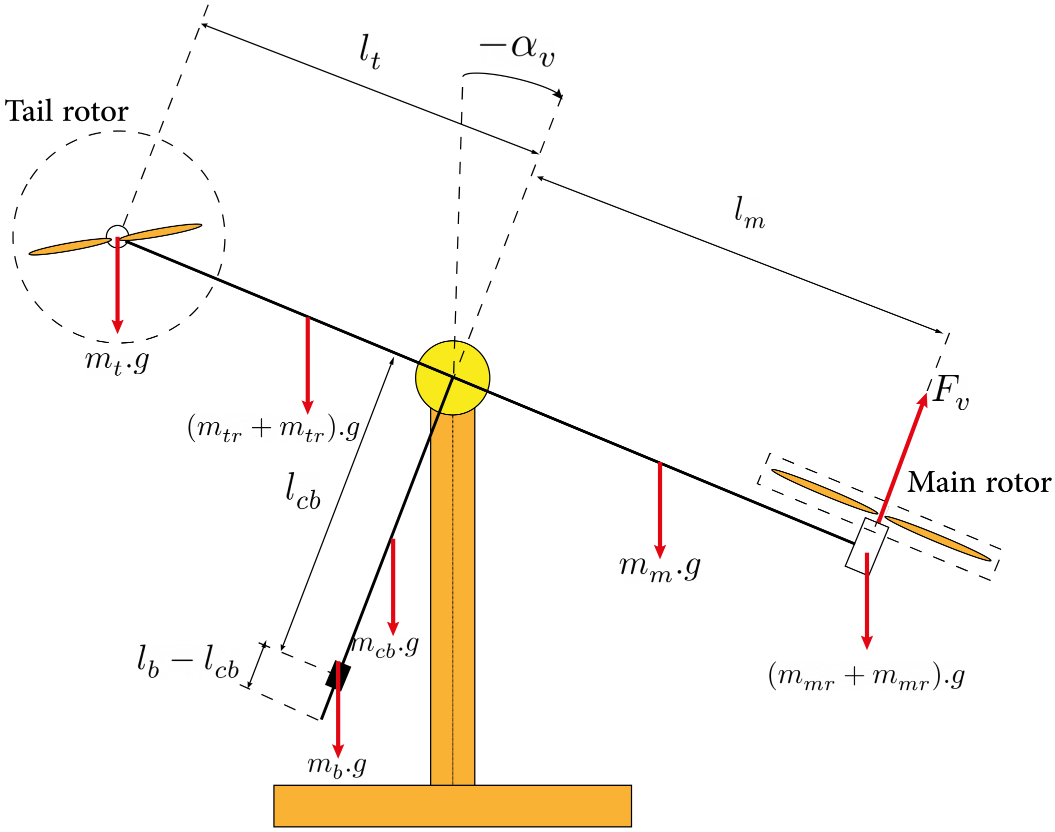

Figure 3.3: Forces in the vertical plane

Where:

F_v: The aerodynamic force created by the main motor
g: Gravitational acceleration of the earth
α_v: The angle of inclination from the vertical plane
m_m: The mass of the beam part holding the main rotor
m_{mr}: The mass of the main rotor
m_{ms}: The mass of the main shield
m_t: The mass of the beam part holding the tail rotor
m_{tr}: The mass of the tail rotor
m_{ts}: The mass of the tail shield
m_{cb}: The mass of the counterweight
m_b: The mass of the counterweight beam
l_m: The length of the main part of the beam
l_t: The length of the tail part of the beam
l_{cb}: The distance between the counterweight and the pivot
l_b: The length of the counterweight beam

According to Newton's second law of motion, we have:

$$\begin{cases} M = J\frac{d^2\alpha}{dt^2} \\ M = \sum_i M_i \\ J = \sum_i J_i \end{cases} \tag{3.1}$$

Where:
M: The torque caused by forces
J: The moment of inertia
We define as well:
$\Omega_{v/h}$: The angular velocity of the rotation in the vertical/horizontal plane
$K_{v/h}$: The friction coefficient in the vertical/horizontal plane

Feedback instrument Ltd in [20] and H.N. Nguyen in [21] simplified the calculation of the moments and the application of Newton's law (3.1) on the TRMS:

- The moment resulting from gravitational force:

$$M_{v1} = g[(A - B)cos(\alpha_v) - Csin(\alpha_v)] \tag{3.2}$$

where:

$$A = (\frac{m_t}{2} + m_{tr} + m_{ts})l_t, B = (\frac{m_m}{2} + m_{mr} + m_{ms})l_m, C = (\frac{m_b}{2} + m_{cb})l_{cb} \tag{3.3}$$

- The moment resulting from the aerodynamic force of the main rotor:

$$M_{v2} = l_m F_v \tag{3.4}$$

- The moment resulting from the centrifugal force:

$$M_{v3} = -\Omega_h^2(A + B + C)sin(\alpha_v)cos(\alpha_v) \tag{3.5}$$

- The moment resulting from the friction force existing in the joint:

$$M_{v4} = -\Omega_v K_v \tag{3.6}$$

The moments of inertia depend on the geometric distribution of the mass of the parts, we can calculate them :

$$\begin{aligned}
&J_{v1} = m_{mr} l_m^2 && J_{v2} = m_m \frac{l_{cb}^2}{3} && J_{v3} = m_{cb} l_{cb}^2 \\
&J_{v4} = m_b \frac{l_{cb}^2}{3} && J_{v5} = m_{tr} l_t^2 && J_{v6} = m_t \frac{l_t^2}{3} \\
&J_{v7} = J_{mr} = \frac{m_{ms}}{2} r_m^2 + m_{ms} l_m^2 && J_{v8} = J_{tr} = m_{ts} r_{ts}^2 + m_{ts} l_t^2
\end{aligned} \tag{3.7}$$

Replacing the above equations in Newton's law, we obtain:

$$\begin{cases}
J_v \frac{d^2\alpha_v}{dt^2} & = l_m F_v - \Omega_h^2 (A + B + C) sin(2\alpha_v) - \Omega_v K_v + g[(A - B) cos(\alpha_v) - C sin(\alpha_v)] \\
S_v & = \frac{d\alpha_v}{dt} \\
J_v S_v & = J_v \Omega_v - J_{tr} w_t
\end{cases} \tag{3.8}$$

Where:
S_v: The angular momentum of the beam in the vertical plane.
w_t: The angular velocity of the tail rotor.

3.2.1.2 Horizontal plane

In a similar way to the vertical plane, the tail rotor is responsible for most of the horizontal dynamics. Since gravity has no effect in this plane, the aerodynamic force is the major force causing the motion. This aerodynamic force is produced mainly by the tail rotor and significantly influenced by the main rotor, a cross-coupling effect that cannot be neglected . In addition, when moving frictions always exist in the joint and must be taken in consideration.

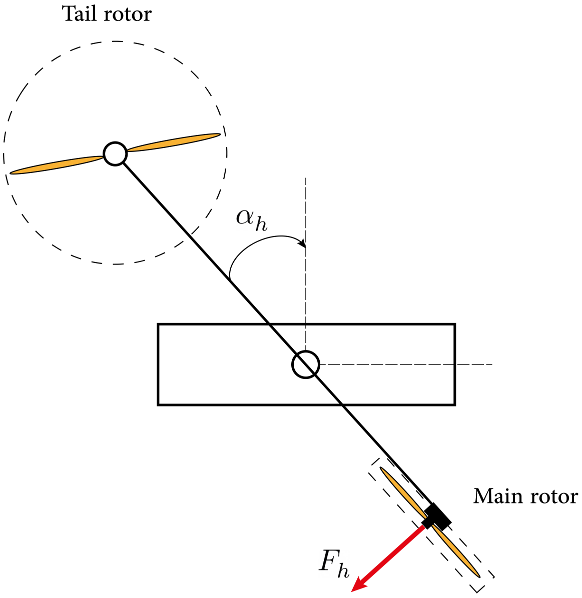

Figure 3.4: Forces in the horizontal plane

Where:
F_h: The aerodynamic force caused by the tail rotor.
α_h: The angle of inclination from the horizontal plane
In a similar way we can calculate:

- The moment caused by the aerodynamic force:

$$M_{h1} = l_t F_h cos(\alpha_v) \tag{3.9}$$

- The moment caused by the friction:

$$M_{h2} = -\Omega_h K_h \tag{3.10}$$

The horizontal moment of inertia was calculated according to the geometrical distribution of the mass of the system:

$$J_h = Dcos^2(\alpha_v) + Esin^2(\alpha_v) + F \tag{3.11}$$

where:

$$D = \frac{m_b}{3} l_b^2 + m_{cb} l_{cb}^2, E = (\frac{m_m}{3} + m_{mr} + m_{ms}) l_m^2 + (\frac{m_t}{3} + m_{ts} + m_{ms}) l_t^2, F = m_{ms} r_{ms}^2 + \frac{m_{ts}}{2} r_{ts}^2 \tag{3.12}$$

And thus:

$$\begin{cases} J_h \frac{d^2\alpha_h}{dt^2} & = l_t F_h cos(\alpha_v) - \Omega_h K_h \\ S_h & = \frac{d\alpha_h}{dt} \\ J_h S_h & = J_h \Omega_h - J_{mr} w_m sin(\alpha_v) \end{cases} \tag{3.13}$$

Where:
S_h: The angular momentum of the beam in the horizontal plane.
w_m: The angular velocity of the main rotor.

The equations (3.8) and (3.13) describe the behaviour of our outputs, the azimuth α_h and elevation α_v, and to produce an accurate mathematical representation of our system we need some relations that describe the behaviour of our controls u_v and u_h the control voltages of the rotors. Such a relation for the DC motors exists and combines the linear dynamics and the static nonlinearity of the rotors [20]:

$$\begin{cases} \frac{du_{vv}}{dt} = \frac{1}{T_{mr}}(-u_{vv} + u_v) \\ \frac{du_{hh}}{dt} = \frac{1}{T_{tr}}(-u_{hh} + u_h) \end{cases} \tag{3.14}$$

Where:
$u_{vv/hh}$: The input voltage of the main/tail rotor.
$T_{mr/tr}$: The time constant of the main/tail rotor.
We note that the angular velocity of each rotor is a static nonlinearity of its input voltage let us call it the function P.

$$\begin{cases} w_m = P_v(u_{vv}) \\ w_t = P_h(u_{hh}) \end{cases} \tag{3.15}$$

Note: We can physically measure some constants like the dimensions of the TRMS and the masses of its parts, their values are included in the Manuel of the TRMS [20].
The measurement of the constants is summarized in the following table:

measurements	
Constant	Value
m_m	0.0145 kg
m_{mr}	0.228 kg
m_{ms}	0.225 kg
m_t	0.0155 kg
m_{tr}	0.206 kg
m_{ts}	0.162 kg
m_{cb}	0.068 kg
m_b	0.022 kg
l_m	0.24 m
l_t	0.25 m
l_{cb}	0.13 m
l_b	0.13 m
r_{ms}	0.155 m
r_{ts}	0.10 m
K_v	0.006 N.s/rad
K_h	0.1 N.s/rad

3.2.1.3 Static nonlinearities and aerodynamic forces estimation

An interesting approach to deal with the non-linear functions in our representation $P_{v/h}$ and $F_{v/h}$ was introduced in [22] , where they proved that the approximated polynomials give valid results. This spares the trouble of dealing with the complex intertwined physical phenomena that produce them, as well as offering a simplicity of the modeling. Nevertheless, some attempts successfully modeled these functions using physical laws with high accuracy [23].
We will consider the polynomial expressions found in [22]:

$$P_v(u_{vv}) = 90.99u_{vv}^6 + 599.73u_{vv}^5 + 129.26u_{vv}^4 - 1238.64u_{vv}^3 + 63.45u_{vv}^2 + 1283.41u_{vv} \tag{3.16}$$

$$P_h(u_{hh}) = 2020u_{hh}^5 - 194.69u_{hh}^4 - 4283.15u_{hh}^3 + 262.27u_{hh}^2 + 3796.83u_{hh} \tag{3.17}$$

$$F_v(w_m) = 3187{\times}10^{-15}w_m^5 - 4096{\times}10^{-12}w_m^4 + 1385{\times}10^{-9}w_m^3 + 1234{\times}10^{-6}w_m^2 + 0.799w_m \tag{3.18}$$

$$F_h(w_t) = 9496{\times}10^{-16}w_t^5 - 9844{\times}10^{-16}w_t^4 + 2785{\times}10^{-10}w_t^3 + 1730{\times}10^{-7}w_t^2 + 0.729w_t \tag{3.19}$$

3.2.2 Block model simulation:

Due to equations (3.8),(3.13) and (3.14) we can build a MATLAB Simulink block model that simulates the attitude of the TRMS (figure 3.5).

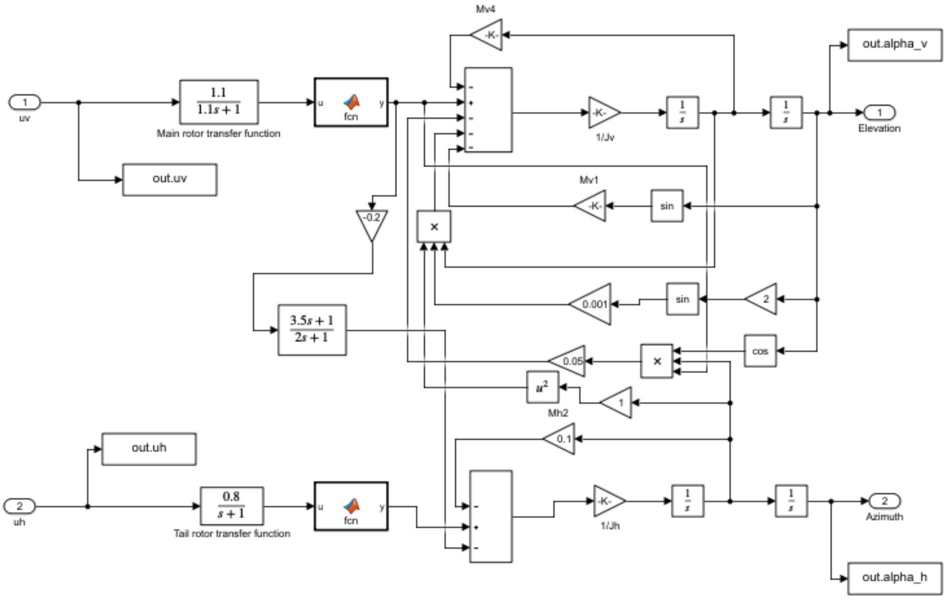

Figure 3.5: Block model of the TRMS

The simulations of a pulse response are presented in figure 3.6.

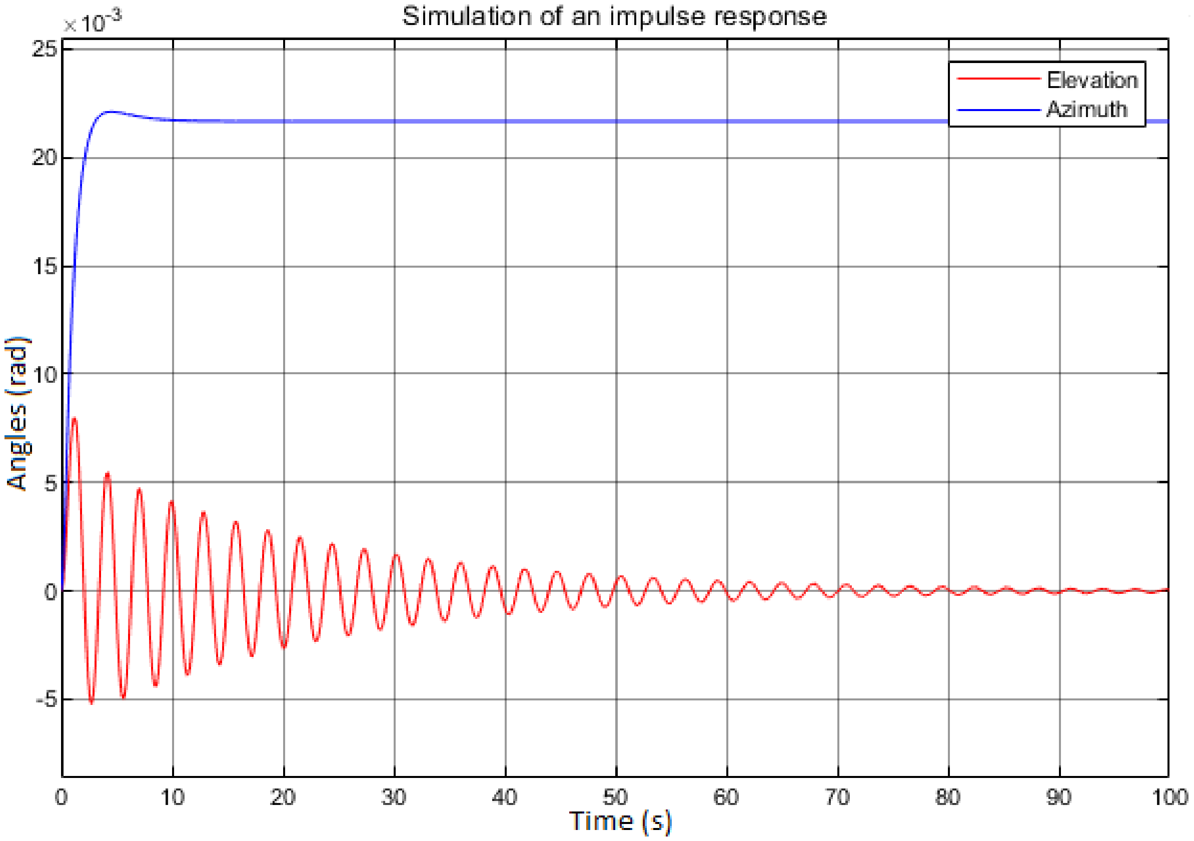

Figure 3.6: Simulation of a pulse response

3.3 The Linear representation

In general, it is interesting to asses a linear mathematical representation of a non-linear system because it offers an accessibility for a mathematical model of a certain hard-to-model non-linear class of systems, and a simplicity of analyzing and studying the system with insignificant errors. The frequency representation for instance, is rich in information and thus allows a good deduction when it comes to control and analysis. In our case however, there is a high order nonlinearity that prevents an accurate linear representation and results in a significant modeling error in the case of traditional transfer function representations. Nevertheless, introducing fractional-order in linear modeling proved to reduce the linear modeling error of non-linearities significantly when it comes to a certain class of non-linear systems [24]. Therefore, we will consider the traditional integer-order transfer function representation as well, for analyzing and comparing means.

3.3.1 Integer-order transfer function

We use in the estimation of our transfer function the data extracted from injecting an appropriate input into the simulation model in figure 3.5, which is a sequence that slightly disturbs the system, deterministic if possible and rich in frequency, such the Pseudo-Random Binary Sequence (PRBS). In our example we will use the PRBS as an input.
The principle of the estimation consists basically in determining the parameters of a fixed model using an optimization method, as followed:

Algorithm 2 Estimation of the model's unknown parameters

```
Result: The optimal model parameters
Initialize the parameters with random values
Excite the real system and the modeled one with a PRBS
Collect the output of the real system $y$ and the predicted output of the model $\tilde{y}$
Calculate the error between the two outputs $\epsilon = |y - \tilde{y}|$
Apply an optimization method to minimize a criteria of the error
if the optimization algorithm diverges then
    return to step 1 with different initial conditions
```

For the optimization criteria to minimize the error, we have many choices. In our example we will consider the least squares principle which aims to minimize the sum of the squares of the error [25]:

$$Min(J) = Min(\sum(\epsilon)^2) \tag{3.20}$$

As for the optimization method we will go with the recursive least squares RLS algorithm [26]. The linear frequency representation of the TRMS can be defined as followed:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

Where: $G(s) = \frac{Y(s)}{U(s)}$ represent the MIMO transfer function (MTF) of the TRMS, and the input and output parameters are:

$$u(t) = \begin{pmatrix} u_v & u_h \end{pmatrix}^t = \begin{pmatrix} u_1 & u_2 \end{pmatrix}^t \qquad y(t) = \begin{pmatrix} \alpha_v & \alpha_h \end{pmatrix}^t = \begin{pmatrix} y_1 & y_2 \end{pmatrix}^t$$

Thus the block model is:

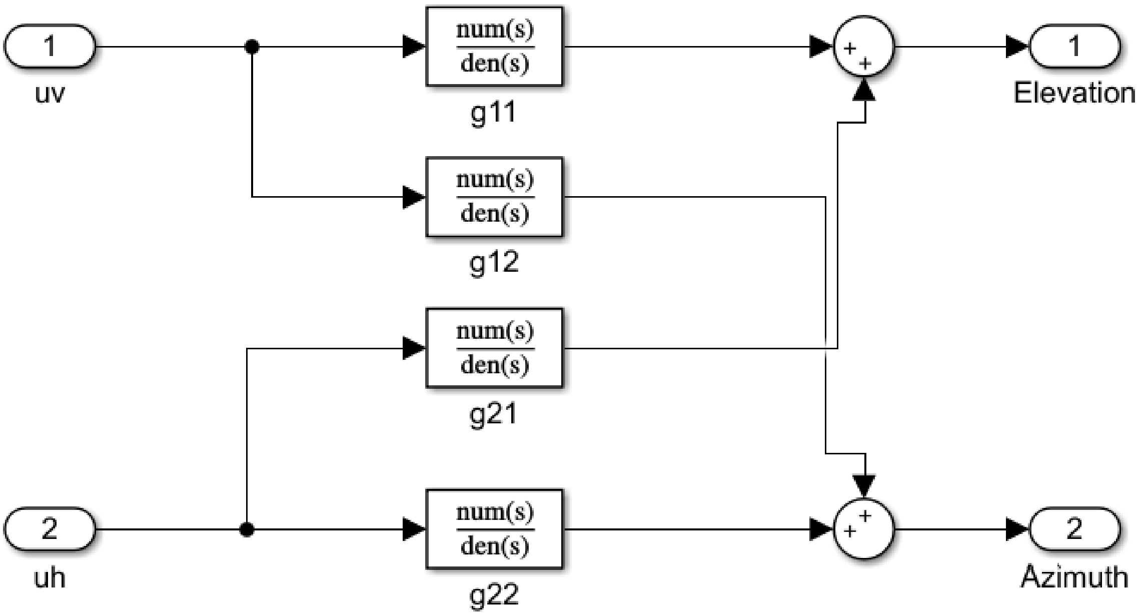

Figure 3.7: The block model of the transfer function representation of the TRMS

We fix the order of our transfer function g_{ii} to 5 poles for a better approximation and no zeros since the simulated response of the TRMS shown in figure 3.5 show no zero's dynamics and we apply **Algorithm 2**.

In our simulation we will rely on a two-steps method to identify the functions: In the first step only one input excites the system with a PRBS while the other input remains zero i.e. we will have $y_1^{step1} = g_{11}u_1^{step1}$ and $y_2^{step1} = g_{12}u_1^{step1}$ from which we can extract the functions g_{11} and g_{12} using the RLS algorithm. In the second step, in a similar way, the first input is switched to zero while the other one excites the system with a PRBS, resulting in $y_1^{step2} = g_{21}u_2^{step2}$ and $y_2^{step2} = g_{22}u_2^{step2}$, and therefore we can estimate the remaining functions.

In literature, many methods were proposed concerning the estimation of the transfer functions of a MIMO system, like the improved one-step method [24] in which the decoupling process was removed.

The estimation gives the following results:

$$\begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} = \begin{bmatrix} \frac{488.1}{s^5+5.856s^4+190.7s^3+478.6s^2+1124s+946.6} & \frac{816.9}{s^5+7.715s^4+219.4s^3+379.1s^2+132.6s+0.3898} \\ 0 & \frac{3777}{s^5+2.147s^4+899.7s^3+1231s^2+282.6s+0.486} \end{bmatrix}$$

The simulation of the identified functions for a random input signal are given in the following figures:

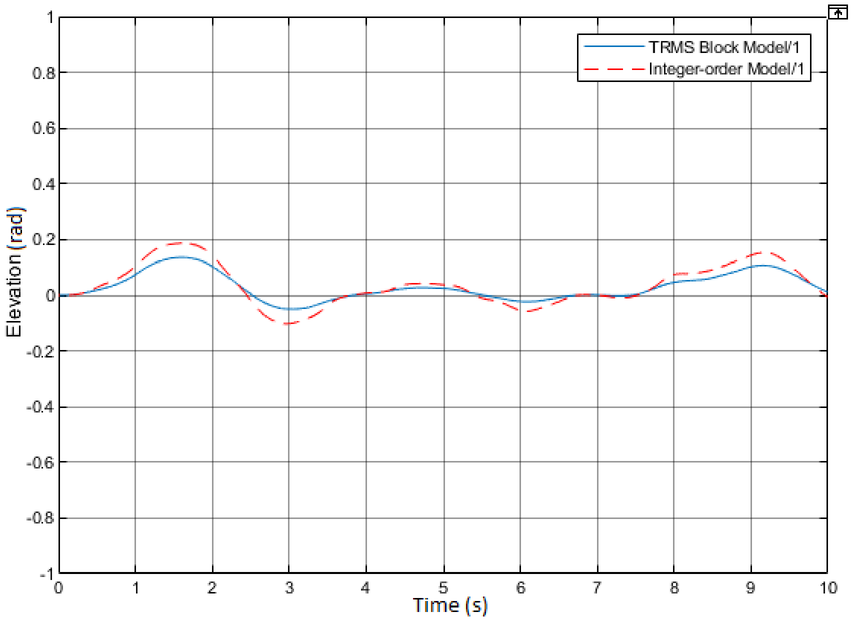

Figure 3.8: Elevation response of the estimated integer-order transfer function

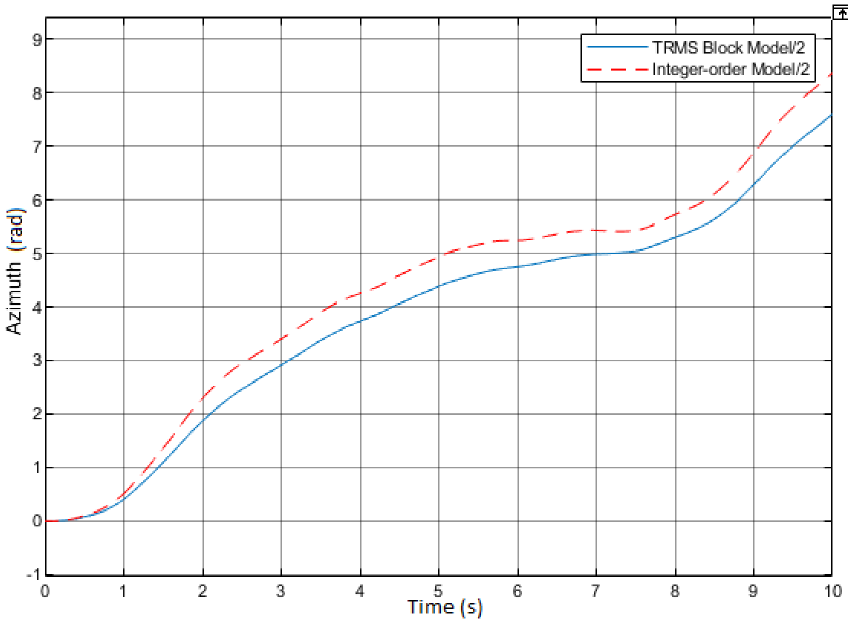

Figure 3.9: Azimuth response of the estimated integer-order transfer function

We can observe the resemblance between the block model and the identified integer-order responses, nevertheless, the identified model is clearly flawed and cannot be used as a valid model.
To note, $g_{21} = 0$ makes sens since the effect of tail rotor on the elevation is negligible, therefor the choice of parameters in the RLS algorithm led to approximate the function to zero.
The integer-order transfer function, being a linear mathematical representation of the TRMS does not model its behaviour well due the high-order non-linearities.

3.3.2 Fractional-order transfer function

The traditional identification methods like the RLS do not work directly in the case of fractional-order estimation, and this is for three main reasons:

1. Unlike the integer-order integro-differential operator, the fractional-order one does not disappear transforming the function into a past version or any other possible physical interpretation. For instance, the first derivative of an output $\dot{y}(t)$ can be interpreted as a past version of the present output i.e. $\dot{y}(t) = y(t-k)$ with $k \in \mathbb{Z}^+$, but for fractional-order operations this property does not exist, and therefore new mathematical solutions must take place.

2. This new transformation generates a non-linear relation between the measurements and the estimated parameters.

3. The fractional orders must be estimated as well and they have no direct relations with the other parameters.

To overcome those problems, the estimation methods were shifted and enhanced to suit the fractional-order systems. We will proceed and discuss how to apply the RLS algorithm for a fractional-order system [27].

3.3.2.1 Recursive least squares algorithm for fractional-order systems

For our TRMS we consider the following ARX system:

$$y(t) + \sum_{i=1}^{n} a_i D^{\alpha_i} y(t) = \sum_{j=0}^{m} b_j D^{\beta_j} u(t) \tag{3.21}$$

where: a_i,$b_j \in \mathbb{R}$ and α_i,$\beta_j \in \mathbb{R}^+$.
The fractional-order transfer function of the system is therefore:

$$G(s) = \frac{U(s)}{Y(s)} = \frac{\sum_{j=0}^{m} b_j s^{\beta_j}}{1 + \sum_{i=1}^{n} a_n s^{\alpha_i}} \tag{3.22}$$

Since it is highly difficult to directly deal with fractional systems we need to approximate the transfer function into an integer-order one. We consider the following version of the N^{th}-order Oustaloop approximation in a band of frequency $[w_b, w_h]$ [28]:

$$\frac{1}{s^m} \approx \frac{1}{(w_c)^m} \frac{\prod_{i=1}^{N-1}(1 + s/z_i)}{\prod_{i=1}^{N}(1 + s/p_i)} = \sum_{i=1}^{N} \frac{k_i}{1 + s/p_i} \tag{3.23}$$

where: $w_c = \gamma w_b$ for $(0.0001 \leq \gamma \leq 0.1)$, $w_{max} = \sigma w_h$ for $(10 \leq \sigma \leq 10000)$, and:

$$\begin{cases} p_1 = w_c 10^{\epsilon/m} \\ p_i = 10 p_{i-1}, i \geq 2 \\ \epsilon = \frac{m(1-m)}{2(N+(1-m)/2)}[log_{10}(w_{max}/w_c)] \end{cases} \tag{3.24}$$

$$\begin{cases} z_1 = p_1 10^{\frac{2\epsilon}{(1-m)}} \\ z_i = p_i 10^m, i \geq 2 \end{cases} \tag{3.25}$$

$$k_j = \frac{1}{(w_c)^m} \frac{\prod_{i=1}^{N}(1 - p_j/z_i)}{\prod_{i=1,i\neq j}^{N}(1 - p_j/p_i)}, j = 1, 2, ..., N \tag{3.26}$$

Now that we have an integer-order approximation of our fractional transfer function we can establish our traditional RLS algorithm, therefore we first calculate the discrete transfer function:

$$\frac{1}{s^m}|_{s=\frac{1-z^{-1}}{T}} = \sum_{i=0}^{N} \frac{k_i z}{(1 + \frac{1}{Tp_i})z - \frac{1}{Tp_i}} \tag{3.27}$$

Then the inverse Z-transform gives:

$$g(k) = \sum_{i=0}^{N} \frac{k_i p_i T}{p_i T + 1} (\frac{1}{p_i T + 1})^k u(k) \tag{3.28}$$

The function (3.28) allows us to define an algebraic expression of the fractional-order operator in the ARX model (4.11) and thus **solving the first problem** by creating a mathematical interpretation of $D^\alpha y(t)$ relating it to y. For M measures such that $M >> Q$ we have:

$$D^\alpha y(t) = \sum_{q=0}^{Q-1} g_\alpha(q) y(M - q) = \sum_{q=0}^{Q-1} \sum_{i=1}^{N} \frac{k_i p_i T}{p_i T + 1} (\frac{1}{p_i T + 1})^q u(q) y(M - q) \tag{3.29}$$

Replacing the results of (3.29) in the ARX model (4.11) we find:

$$y(M) = -\frac{\sum_{i=1}^{n} a_i \sum_{q=1}^{Q-1} g_\alpha(q) y(M - q)}{\sum_{i=1}^{n} a_i g_\alpha(0) + 1} + \frac{\sum_{j=0}^{m} b_j \sum_{q=0}^{Q-1} g_\beta(q) u(M - q)}{\sum_{i=1}^{n} a_i g_\alpha(0) + 1} \tag{3.30}$$

As we can observe, the above relation (3.30) involves the measurements and the parameters we want to estimate and it is non-linear . So to linearize them and as a consequence solve **the second problem** we can create the following variables:

$$\begin{aligned} a_i' &= \tfrac{a_i}{\sum_{i=1}^{n} a_i g_\alpha(0) + 1} \\ b_j' &= \tfrac{b_j}{\sum_{i=1}^{n} a_i g_\alpha(0) + 1} \end{aligned} \tag{3.31}$$

The predicted output becomes:

$$\tilde{y}(M) = -\sum_{i=1}^{n} a_i' Y_i(M) + \sum_{j=0}^{m} b_j' U_j(M) = \phi(M)\theta \tag{3.32}$$

where:

$$\begin{gathered} Y_i(M) = \textstyle\sum_{q=1}^{Q-1} g_\alpha(q) y(M - q) \\ U_j(M) = \textstyle\sum_{q=0}^{Q-1} g_\beta(q) u(M - q) \\ \theta^t = [a_1', a_2', ..., a_n', b_0', ..., b_m'] \\ \phi(M) = [Y_1(M), Y_2(M), ..., Y_n(M), U_0(M), ..., Y_m(M)] \end{gathered} \tag{3.33}$$

Now for the last problem, many solutions were proposed to determine the fractional orders. For instance, D.Idiou et al. in [27] and K.Kothari in [24] considered in their approaches a

commensurable system, which left them with only one fractional order to identify. Such an identification can be done for example, through a basic loop that goes over every possible order which is negligible since there are only 101 possibilities for a 2 decimal digits fractional-order $0 \leq \alpha \leq 1$. Or more efficiently, by using an optimization algorithm like the particle swarm optimization (PSO) to quickly estimate the best order. Others like O.Stark et al. in [29], developed techniques to estimate the non-commensurable fractional-order.
In our study, we will consider the 2 decimal digits non-commensurable fractional-order, in which it will be identified separately from the parameters a_i and b_j using the Trust-Region-Reflective (TRR) MATLAB function presented in [30] like follows:

```
**Algorithm 3** Identification of the fractional-order transfer function
1. Fix an ARX structure
2. Initialize the fractional-orders with random values
3. Identify the parameters' vector $\theta$ using RLS algorithm
4. Estimate the fractional-orders using the TRR algorithm in the new estimated system
5. Estimate the new parameters' vector related to the new order using the RLS algorithm
if the identified system converges to the real one then
    the estimation is finished
else
    return to step 4.
```

The TRR is an optimization algorithm used to calculate the minimum of an objective function, just like the RLS. We use it for the estimation of the fractional-orders because the regression principle used in the TRR makes the estimation of the exponents faster than the RLS .

3.3.2.2 Estimation and simulation

We fix the same structure as the integer-order transfer function with 5 poles and no zeros. As for the approximation of the fractional-order transfer function, we consider the 10^{th}-order Oustaloop approximation in the range of $[0.0001, 10000] rad/s$.
The estimation results are:

$$\begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} =$$

$$\begin{bmatrix} \frac{0.8}{0.66s^{2.75}+5.69s^{1.09}-2.65s^{1.01}-8.06s^{0.38}+9.12s^{0.29}+0.36s^{0.04}} & \frac{1}{0.04s^{3.67}+0.04s^{2.92}+0.45s^{2.56}+0.14s^{2.17}-0.44s^{1.65}+0.66s^{1.4}} \\ \frac{1}{433.53s^{1.15}+428.23s^{0.93}+437.97s^{0.67}+427.77s^{0.42}+434.6s^{0.18}+432.27} & \frac{0.85}{0.26s^{2.8}-0.01s^{2.77}+0.41s^{2.73}-0.66s^{2.15}+0.68s^{1.75}} \end{bmatrix}$$

The simulation of the identified functions for a random input signal are given in the following figures:

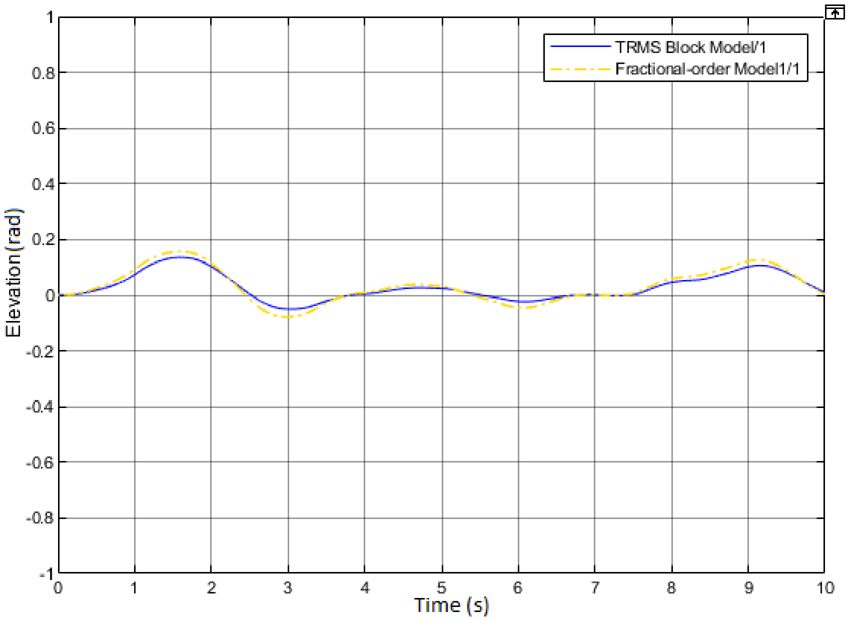

Figure 3.10: Elevation response of the estimated fractional-order transfer function

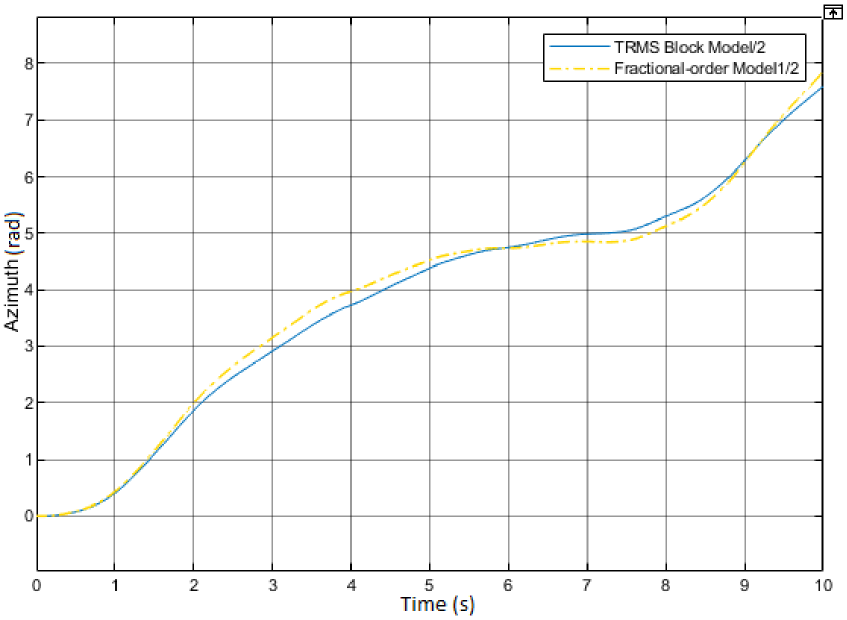

Figure 3.11: Azimuth response of the estimated fractional-order transfer function

The results of this estimation are quiet precise, with some little modeling errors. The model can be used as a mathematical representation of the TRMS to a certain degree. We must note that we cannot easily simulate the fractional transfer functions directly and we have to pass through an integer-order approximation, in which the type of approximation, order and the frequency range must be chosen carefully and correspond to those we estimated the function with.

3.3.3 Comparison

The simulation of our identified systems together is as followed:

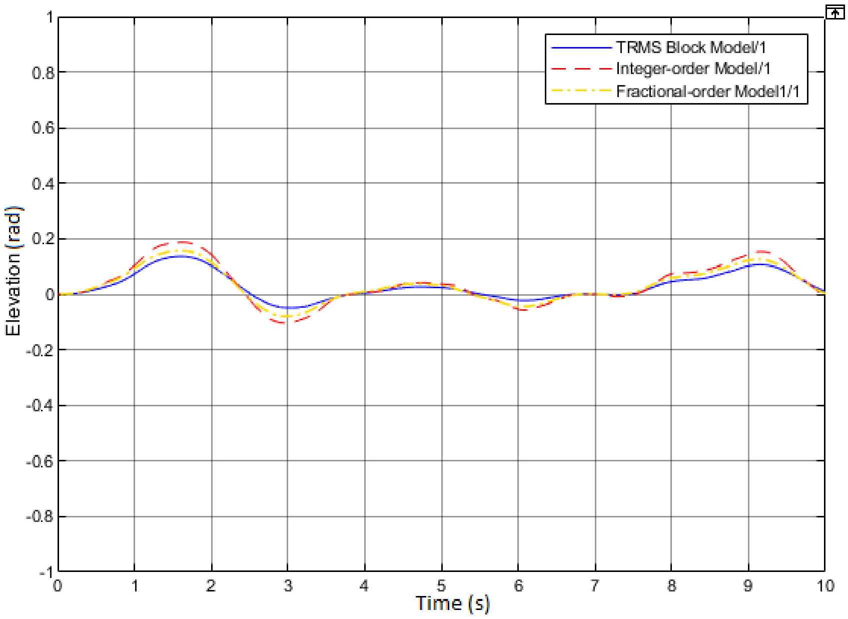

Figure 3.12: Elevation response of the estimated transfer functions

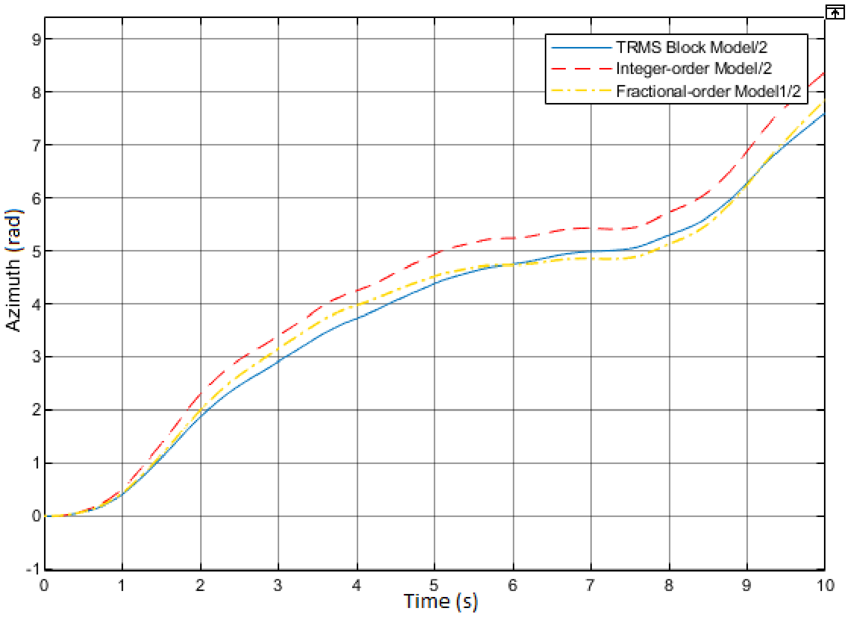

Figure 3.13: Azimuth response of the estimated transfer functions

For comparison purposes we consider the residual sum of squares (RSS) criteria, which evaluates the fitness of our estimated models, such that the smaller the RSS is, the better is the estimation. It is given by the following expression [31] :

$$RSS = \sum_{i=1}^{n} (y_i(t) - \tilde{y}_i(t))^2 \tag{3.34}$$

where n is the number of the measurements.
The results of the RSS criteria for the estimated responses in figures 3.12 and 3.13 are given in the following table, for a 100 measurements we have:

	TF estimation	FOTF estimation
RSS for elevation error	2.5138	0.6423
RSS for azimuth error	101.9981	69.1929

Table 3.1: Residual sum of squares of the estimation error

We can observe:

- The fractional-order transfer function is a more accurate representation than the integer-order one when it comes to the TRMS.
- Despite being a linear representation, the identified fractional-order transfer function represented the TRMS, a highly non-linear system with minimal modeling error.
- The modeling error in both cases is due to the limitation of the frequency representation for non-linear systems modeling, but we can see clearly through the RSS that the fractional-order estimation is significantly more accurate.

3.4 The non-linear representation

A more accurate representation of the TRMS must take in consideration the non-linear behaviour, like the state-space representation. In addition to modeling non-linearities, this representation is more adapted to the control problem because it allows us to directly asses the commands of the system in real time.

3.4.1 Integer-order state-space model

A precise state-space model is one that includes the controls, measures and any internal variable that has a significant relation with the previous ones and with the dynamics of the system like those that influence the stability, controllability and observability.
In our case a good choice of the variables is the following:
State vector $x(t)$: $\begin{pmatrix}\alpha_v & \Omega_v & u_{vv} & \alpha_h & \Omega_h & u_{hh}\end{pmatrix}^t$
Control vector $u(t)$: $\begin{pmatrix}u_v & u_h\end{pmatrix}^t$
Output vector $y(t)$: $\begin{pmatrix}\alpha_v & \alpha_h\end{pmatrix}^t$
To note, many good choices of the variables exist, like the one proposed by M. Ilyas et al. in [32]. They result in different mathematical representations but give similar precise results. Also, it is quiet difficult to deal directly with the coupling effect using the state-space representation since the decoupling in this representation generates model errors, and thus we will consider the coupling effect as an external disturbance when controlling our system.
Now that we defined our variables and we have all the necessary expressions and equations to create the state-space model for our TRMS:

From (3.8), (3.13) and (3.14) :

$$\begin{cases} \dot{x}_1 = \frac{1}{J_v}x_2 + \frac{J_{tr}}{J_v}P_h(x_6) \\ \dot{x}_2 = l_mF_v(P_v(x_3)) + g[(A-B)cos(x_1) - Csin(x_1)] - \frac{1}{2}[\frac{1}{J_h(x_1)}x_5 \\ +\frac{J_{mr}}{J_h(x_1)}P_v(x_3)cos(x_1)]^2(A+B+C)sin(2x_1) - K_v[\frac{1}{J_v}x_2 + \frac{J_{tr}}{J_v}P_h(x_6)] \\ \dot{x}_3 = \frac{1}{T_{mr}}(-x_3 + u_v) \\ \dot{x}_4 = \frac{1}{J_h(x_1)}x_5 + \frac{J_{mr}}{J_h(x_1)}P_v(x_3)cos(x_1) \\ \dot{x}_5 = l_tF_h(P_h(x_6))cos(x_1) - K_h[\frac{1}{J_h(x_1)}x_5 + \frac{J_{mr}}{J_h(x_1)}P_v(x_3)cos(x_1)] \\ \dot{x}_6 = \frac{1}{T_{tr}}(-x_6 + u_h) \end{cases} \tag{3.35}$$

As we can observe, the state-space model is highly nonlinear with the existence of sinusoidal behaviours, as well as significant cross-coupling effect between the vertical and horizontal subsystems.
It is a valid model since it was extracted from the equations without loss of information. Its simulation is identical to the block model in figure 3.5.

3.4.2 Fractional-order state-space modeling

It is interesting to introduce fractional calculus in the state-space representation of the TRMS, this could result in an enhanced model. Nevertheless, it is extremely complicated to find the expressions of the fractional derivatives of our states due to the high nonlinearities and complexity of the equations (3.8),(3.13) and (3.14) describing the behaviour of the system.
In literature, many methods were proposed to deal with nonlinearities for fractional-order state-space modeling, the most famous are Wiener, Hammerstein and Hammerstein-Wiener models [33]–[37]. For a better understanding of the process, let us discuss the principle of Wiener model identification, because it is suitable for most parts of our TRMS.
The Wiener model divides the system into two cascaded blocks: a dynamic linear block and a static nonlinear one (figure 3.14), which is perfect for some parts of our system. For instance the rotors present a nonlinear behaviour that can be described as a linear first-order system cascaded with a static nonlinearity of the input voltage $P(u)$.

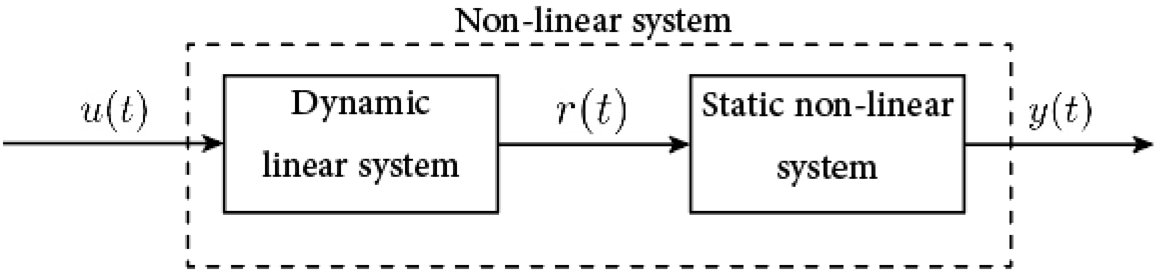

Figure 3.14: Wiener model

The linear part can be represented by a linear asymptotically stable fractional-order state-space model, for example:

$$\begin{cases} D^{\alpha}x(t) = Ax(t) + Bu(t) \\ r(t) = Cx(t) + Du(t) \end{cases} \tag{3.36}$$

While the nonlinear part is usually represented by any basic nonlinear function $f(r)$ such as a polynomial, same like some of our nonlinear functions of the TRMS.

$$f(r) = \sum_{i=0}^{n} a_i r^i, a_i \in \mathbb{R} \tag{3.37}$$

the combined system (3.38) represent a non-linear fractional-order model.

$$\begin{cases} D^{\alpha}x(t) = Ax(t) + Bu(t) \\ y(t) = f(r) = \sum_{i=0}^{n} a_i(Cx(t) + Du(t))^i \end{cases} \tag{3.38}$$

Hammerstein model is just a backward version of the Wiener model, in which the non-linear part comes first, and as consequence the Hammerstein-Wiener and Wiener-Hammerstein models are combinations of the two first models. Following this doctrine we can modelized a multitude of nonlinear system classes.

Back to our TRMS for example, the first-order linear behaviour of our main rotor can be modeled by a fractional-order state-space representation and the static nonlinearity by a polynomial (approximation in (3.16)). However, the discussed complexity of its intertwined relations makes it complicated to apply those methods on all of the parts. In addition, it is an unnecessary representation for the TRMS since it does not occasion any of the phenomena that have limited representations with traditional integer-order models like the memory effect in batteries for example [29]. The integer-order state-space (3.35) gives highly precise simulation results, and thus we will proceed with it to synthesize the controllers.

3.5 Conclusion

In this chapter, we went through a physical study of the TRMS to assess the mathematical relations that express its behaviour. Those expressions allowed us to build a block simulation model to use as a base to simulate the attitude of the system. In addition, we estimated three different mathematical representations to analyse the system with in the next chapters, which are:

- A linear integer-order transfer function that turned out to be incoherent due to its incapability to bypass the nonlinearities of the TRMS.
- A linear-fractional order transfer function that better approaches the real model and proved to be valid with some inaccuracies.
- A highly precise nonlinear integer-order state-space representation, purely a white-box estimated model.

Also we discussed the possibility to assess an enhanced non-linear fractional-order state-space representation of the TRMS.

Chapter 4

PID and FOPID control

Introduction

PID control is the most used controlling technique in industrial applications. It was first introduced by Elmer Sperry in 1911 [38] and developed since then.

The PID controllers operate in closed-loop feedback control, they target the error between the response of the system and the desired reference input to which they attribute a proportional gain k_p (controller P), a gain to the integral of the error k_i (controller I) and a gain to the derivative of the error k_d (controller D). These actions play on the characteristics of the closed-loop response which are the time delay, the overshoot, the oscillations and the steady-state error.

An interesting development of the PID controllers these last years consists in the introduction of fractional calculus into the effect of the integral and derivative actions. In this chapter we will assess a comparative study of the control of the TRMS using the integer and fractional-order PID controllers.

4.1 Problem statement

Designing a PID controller for non-linear systems is quiet different from the linear case, since we do not have a direct frequency representation like the transfer function that we can use to determine the PID parameters for a precise behaviour. This means that we cannot rely on the traditional methods for PID tuning, and this includes the FOPID as well.

An obvious solution is to try linearizing the system since a wide class of systems accept a linearization around the different operating points, and apply to each point the tradition PID controller. This approach however risks poor control qualities for high-order nonlinear systems, like the TRMS.

A more sophisticated and modern technique consists in estimating the parameters of the PID controller directly for the non-linear system using optimization algorithms like the Particle Swarm algorithm (PSO) [39], [40] and the Genetic Algorithm (GA) [41], [42]. These algorithms are used to minimise a cost function like the error between the reference input and the tuned closed-loop response by tuning the controller's parameters to achieve a better response.

In this study we will use the PSO algorithm to estimate the PID and FOPID parameters around the equilibrium point $(\alpha_v, \alpha_h) = (0, 0)$.

4.2 Particle swarm optimization

The particle swarm optimization is an iterative algorithm that was inspired from the movement of animal swarms like birds, insects and fishes, it mimics their navigation as a group to reach a objective location.

The principle of the method consists in the following:

- A populatiom of n homogeneous particles dedicated to solve the optimization problem are randomly arranged in a search-space D and each of the particles is a potential solution.

- Each particle has a position x_i and moves with a velocity of v_i.
- The particles have the ability to evaluate the quality of their positions using an objective function f, as well as keeping track of their best visited position $Pbest$.
- The particles communicate with the neighbouring particles and keep track of the global best position $Gbest$.
- Each particle adjusts its position and velocity in every iteration in order to converge to the global optimum.

In the beginning the positions and velocities are initialized randomly between the maximum and minimum candidate solutions, then each particle i tries to get to the best recorded performance according to the formula in figure 4.1.

$$\underbrace{v_{i,j}^{k+1}}_{\text{Next velocity}} = w\underbrace{v_{i,j}^{k}}_{\text{Current velocity}} + c_1 r_{1i,j}^{k}\underbrace{(\underbrace{Pbest_{i,j}^{k}}_{\text{Personal best Position}} - x_{i,j}^{k})}_{\text{Distance to the personal best}} + c_2 r_{2i,j}^{k}\underbrace{(\underbrace{Gbest_{j}^{k}}_{\text{Global best Position}} - x_{i,j}^{k})}_{\text{Distance to the global best}}, j \in \{1, ..., D\}$$

Figure 4.1: Velocity update in the PSO algorithm

w is called the weight of inertia, $c1$ and $c2$ are positive constants used to control the importance of the distance to the personal best (called cognitive behaviour) and of the distance to the global best (called social behaviour) respectively.
r_1 and r_2 are random numbers in the range of $[0, 1]$ that induce a stochastic character to the effect of the cognitive and social behaviours.
j represents the index of the dimension of the search-space D.
The position is therefore updated in every iteration like follows:

$$x_{i,j}^{k+1} = x_{i,j}^{k} + v_{i,j}^{k+1}, j \in \{1, ..., D\} \tag{4.1}$$

The algorithm also updates the best local and global positions in every iteration. In the case of a minimization problem for example the update is:

$$Pbest_i^{k+1} = \begin{cases} Pbest_i^k, & if \quad Pbest_i^k \leq f(x_i^k) \\ x_i^{k+1}, & if \quad Pbest_i^k > f(x_i^k) \end{cases} \tag{4.2}$$

$$Gbest^{k+1} = \underset{Pbest_i}{argmin} f(Pbest_i^{k+1}), 1 \leq i \leq n \tag{4.3}$$

where $argmin$ function returns the indices of the minimum values found.
We note that the algorithm needs a stopping criteria as well which defines the objective of the minimization and evaluates the results. In addition, this allows us to avoid going through all of the iterations when the objective can be attained for fewer attempts.
We can recap the PSO algorithm as follows:

Algorithm 4 Particle Swarm Optimization

```
1. Initialize the controlling parameters: The number of iterations, n, $c_1, c_2, w$.
2. Initialize the population of n particles randomly in the range of the potential solutions
3. for each particle do
      Evaluate the position of the particle using the objective function
      Calculate Pbest
      Calculate Gbest
4. while the stopping condition is not satisfied do
      Update the velocity
      Update the position
      return to step 3
5. Return Gbest the global optimum
```

Many variations of the PSO algorithm exist, and they all aim to improve some of its aspects for certain applications. We can mention for example the technique of setting up a variable weight of inertia w to enhance the quality of the results. Either by randomly varying it throughout the iterations or by adapting it accordingly to the solutions. In [43] results of adaptive weight inertia in the PSO algorithm proved the superiority of this technique in terms of speed convergence and accuracy.

It is worth mentioning that the size of the search-space and the number of iterations influence drastically the quality of the final results of the optimization, where a larger search-space and a bigger number of iterations give a better quality. Nevertheless, they induce additional computation time and thus it is more optimal to find a compromise between the time and the quality of the results.

Sometimes it happens that some particles exit the search-space, and to avoid this problem many solutions were proposed in literature. X. Cai and Y. Tan in [44] for instance, studied the effect of imposing in the algorithm some limitations concerning the velocity, where they confined it in a specific range. This will prevent the particles from getting high velocities that cause them to explode out of the search-space.

Another approach consists in establishing a relationship between w, $c1$ and $c2$ using a constriction factor χ, this was proven by M. Clerc and J. Kennedy in [45] to confine the particles inside the search-space. With this method the velocity update in figure 4.1 becomes:

$$v_{i,j}^{k+1} = \chi(v_i^k) + \Phi_1 r_{1i,j}^k (Pbest_{i,j}^k - x_{i,j}^k) + \Phi_2 r_{2i,j}^k (Gbest_{i,j}^k - x_{i,j}^k) \tag{4.4}$$

where:

$$\begin{cases} \chi = \frac{2}{|2-\Phi-\sqrt{\Phi^2-2\Phi}|} \\ \Phi = \Phi_1 + \Phi_2 > 4 \end{cases} \tag{4.5}$$

Compared to the velocity update in figure 4.1 we have the following equivalents:

$$\begin{gathered} w \equiv \chi \\ c1 \equiv \chi\Phi_1 \\ c2 \equiv \chi\Phi_2 \end{gathered} \tag{4.6}$$

Many tests were made by Clerc and Kennedy in [45] to find the optimal values of χ,Φ_1

and Φ_2:

$$\begin{cases} \chi = 0.7298844 \\ \Phi_1 = 2.05 \\ \Phi_2 = 2.05 \end{cases} \tag{4.7}$$

In this study we will use this approach to confine our particles.

4.3 PID controller using the PSO algorithm

For the TRMS, each output has a PID controller and they are defined as followed:

$$u_v = k_{p1}e_v(t) + k_{i1}\int e_v(t)dt + k_{d1}\frac{de_v(t)}{dt}, \quad e_v(t) = \alpha_{vref} - \alpha_v \tag{4.8}$$

$$u_h = k_{p2}e_h(t) + k_{i2}\int e_h(t)dt + k_{d2}\frac{de_h(t)}{dt}, \quad e_h(t) = \alpha_{href} - \alpha_h \tag{4.9}$$

The gains $k_{p1}, k_{i1}, k_{d1}, k_{p2}, k_{i2}, k_{d2}$ of the PID controllers are to be estimated with the PSO algorithm. The structure of this estimation process is shown in figure 4.2.

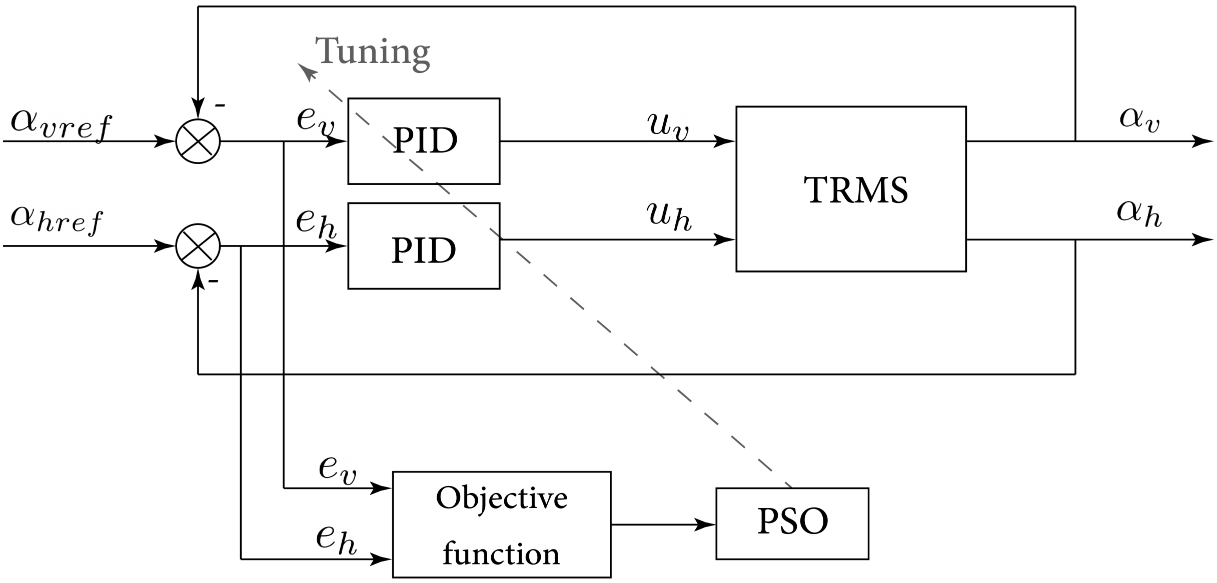

Figure 4.2: Tuning of the PID using the PSO

4.3.1 Open-loop response

In order to compare the closed-loop controlled response to a reference we need to simulate the open-loop response of the TRMS.
For a step input we get the following results:

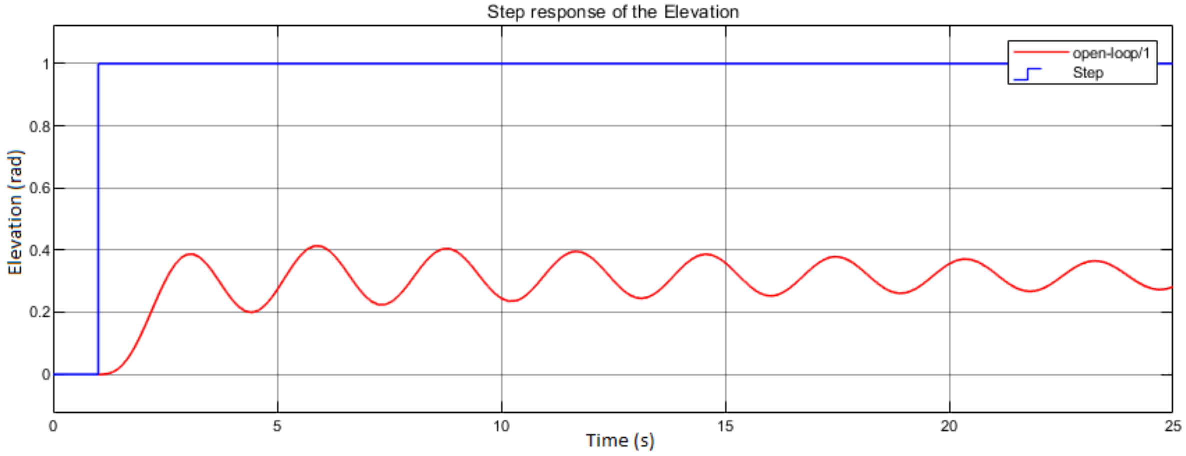

Figure 4.3: Step open-loop response of the elevation

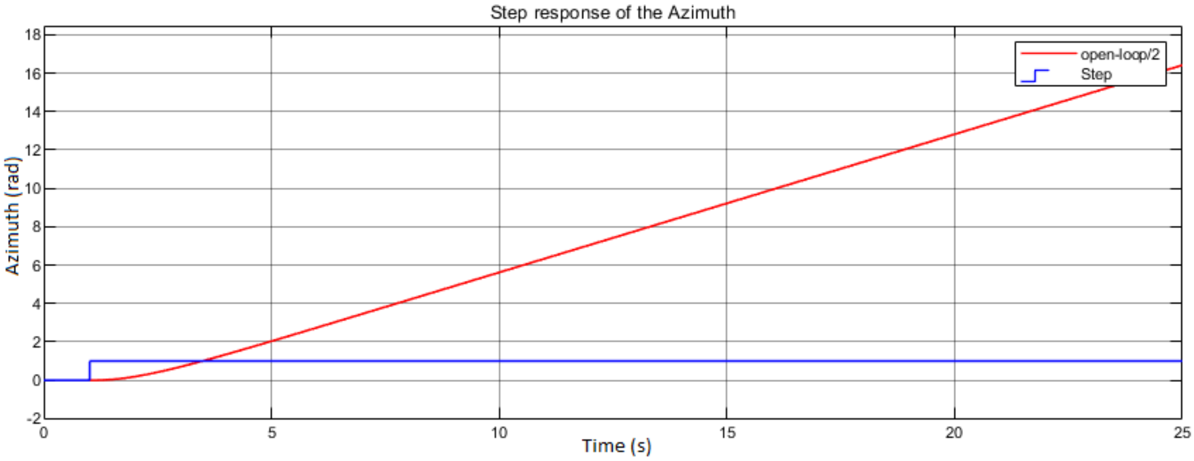

Figure 4.4: Step open-loop response of the azimuth

we observe in the open-loop response of the step that the elevation has a significant steady-state error and an oscillating behaviour of high order. As for the azimuth, it is smoothly divergent.

4.3.2 Setting the PSO parameters

We establish the structure in figure 4.2 on MATLAB Simulink to estimate the gains of the PID controllers. The best choice control-wise is to tune the entire vector $P = [k_{p1}, k_{i1}, k_{d1}, k_{p2}, k_{i2}, k_{d2}]$ because it takes in consideration the cross-coupling effect in its complete influence. However, tuning with metaheuristic algorithms like the PSO and GA is highly time-consuming, especially for big size estimation vectors. Thus, we will consider the cross-coupling effect as a disturbance and we will estimate each controller individually, this leave us with a 3-dimensional PSO problem for each controller.
We choose the objective function f as the sum of the squared errors $e(t)$ between the desired and actual inputs, for Q iterations:

$$f(gains) = \sum_{i=1}^{Q} (e(t))^2 \tag{4.10}$$

We fix the constants χ, Φ_1 and Φ_2 as in (4.7), the number of particle as 30 particles and the number of iterations as 100 iterations. In consideration of the compromise time/quality of the estimation we will also limit the search-space as follows: $0 \leq k_p, k_i, k_d \leq 10$.

4.3.3 Tuning results

The tuning process gives the following results:

Parameters	k$_p$	k$_i$	k$_d$
Elevation	2.5205	10	10
Azimuth	10	0	10

Table 4.1: PID parameters for the TRMS

The simulation of the step closed-loop response using the tuned PIDs is given in figures 4.5 and 4.6.

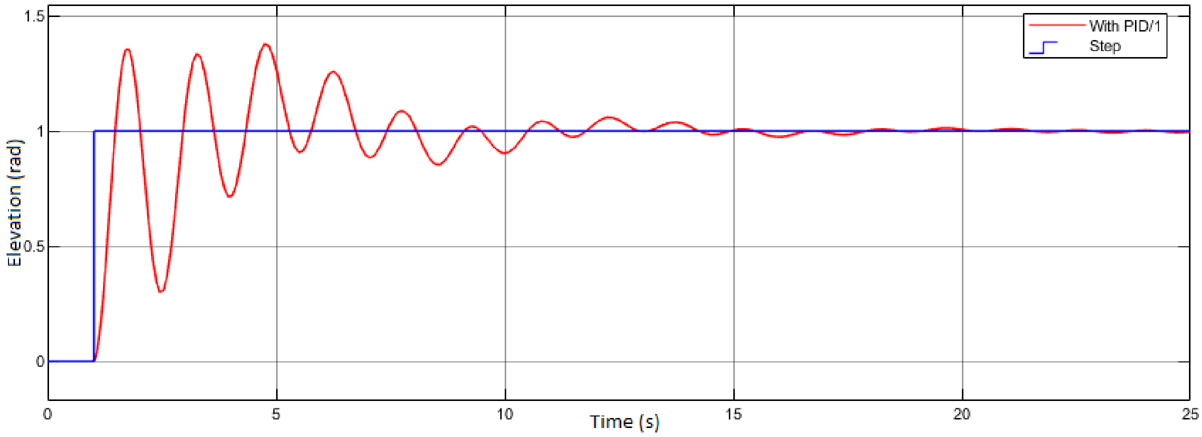

Figure 4.5: Step response of the elevation with PID

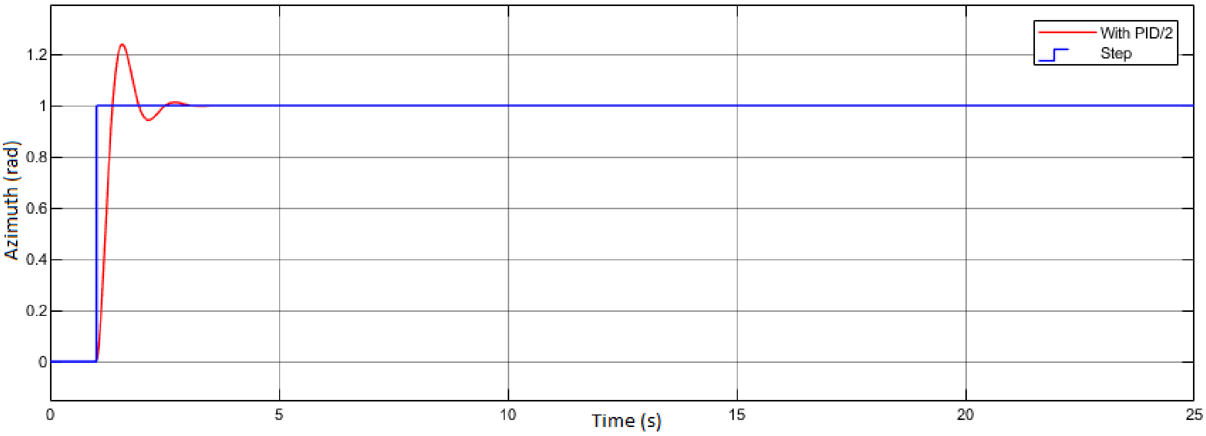

Figure 4.6: Step response of the azimuth with PID

The corresponding input signals are:

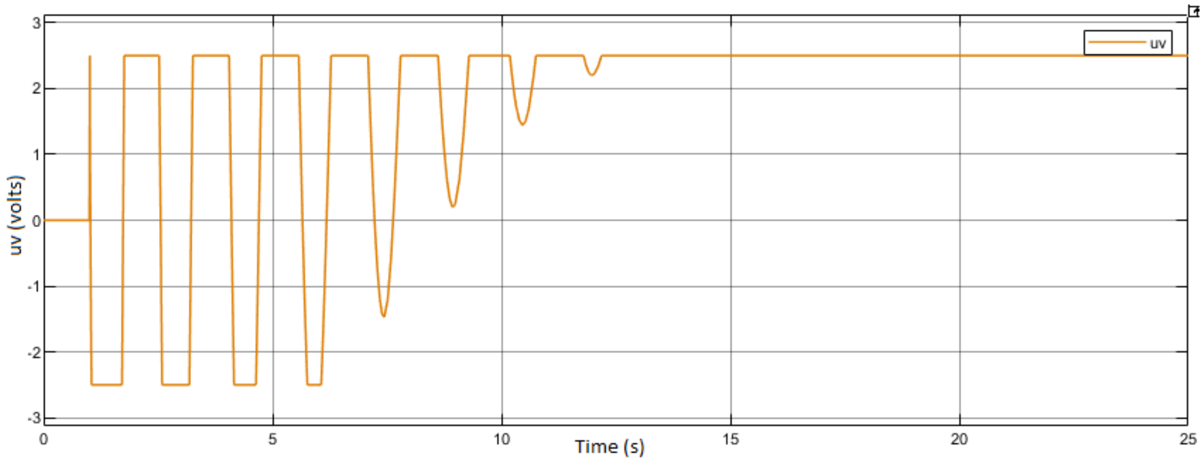

Figure 4.7: Input signal of the elevation subsystem with PID

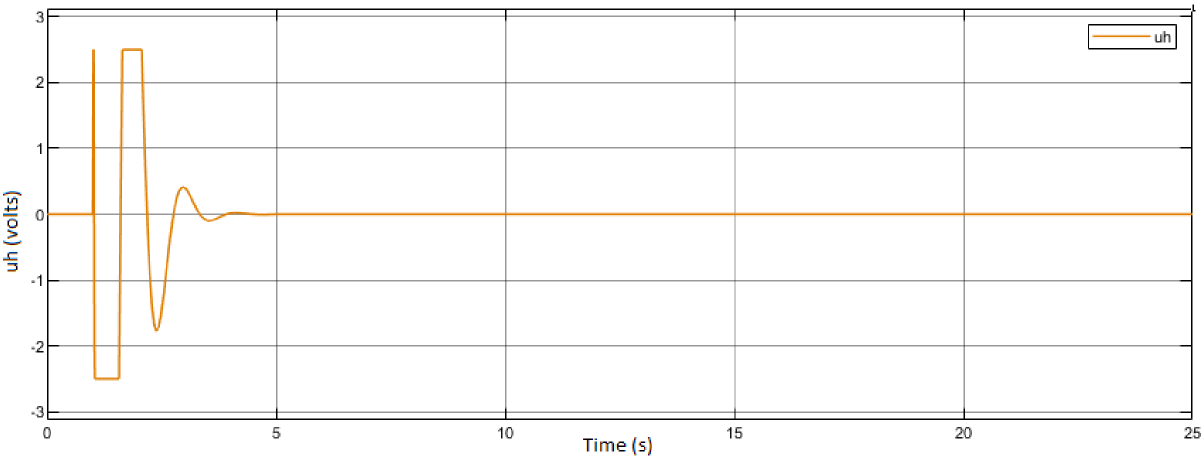

Figure 4.8: Input signal of the azimuth subsystem with PID

Now we change the parameters of our TRMS by adding an additional mass to the mathematical representation which represents a modeling error, and by varying the position of the counterbalance weight by $3cm$. The simulations of the step responses in this case are:

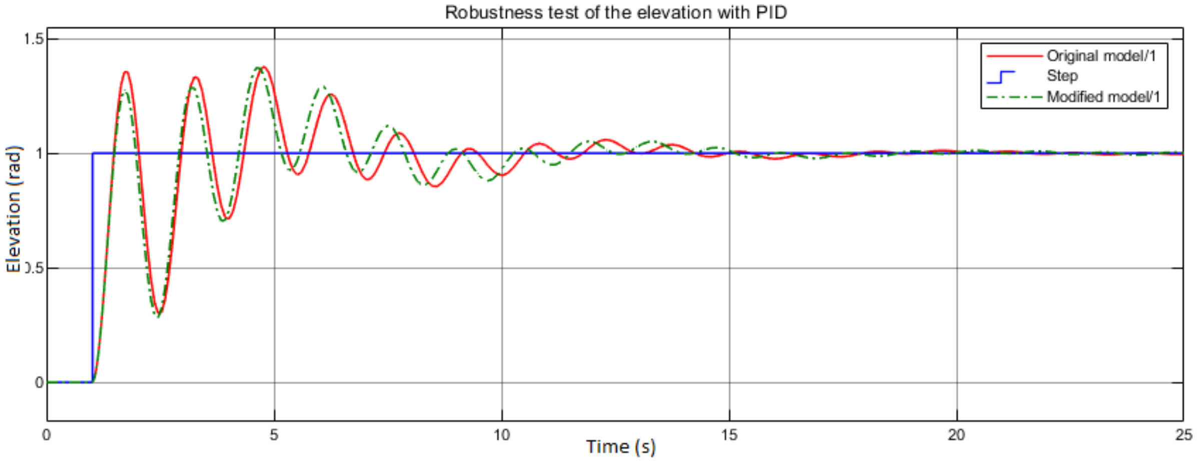

Figure 4.9: Robustness test of the PID for elevation subsystem

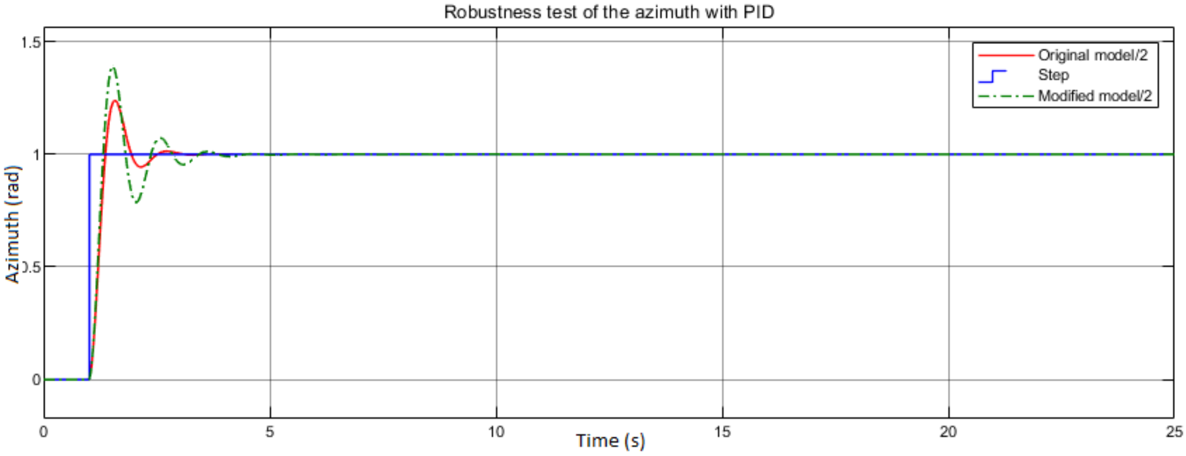

Figure 4.10: Robustness test of the PID for azimuth subsystem

These results are obtained for a maximum number of iterations (100). We can observe the following:

- Both the elevation and azimuth converges to the step signal.
- For the elevation the integral and derivative gains k_{i1} and k_{d1} are maxed out, same for the proportional and derivative gains k_{p2} and k_{d2} of the azimuth.
- Increasing the imposed limits of the gains increases drastically computation time but did not result in any significant enhancement of the quality of control, thus limiting the gains between 0 and 10 is a good choice.
- The elevation subsystem presents many oscillations in the process. But unlike in the open-loop response, the PID decreases the number and amplitude of the oscillations drastically over time.
- The azimuth subsystem has an integration and thus the PSO estimated the integral gain as 0. We can use this information and reduce the dimensions of the PSO to 2 dimensions for a faster optimization.
- The input signals frequently reach the physical limitations of $\pm 2.5v$, this implies that if the system tolerates a wider voltage range the inputs would be less limited and result in better control quality.

4.3.4 Feedforward control

An approach to enhance the poor quality of the PID control shown in figure 4.5 for the elevation subsystem, consists in using a feedforward control alongside with the PID feedback. S.F.Toha in [19] proved the efficiency of the method.
Its principle consists in feeding the control loop with the inverse model of the TRMS, however for nonlinear systems the inverse model must be estimated. The structure of a the PID with feedforward control in shown in figure 4.11 .

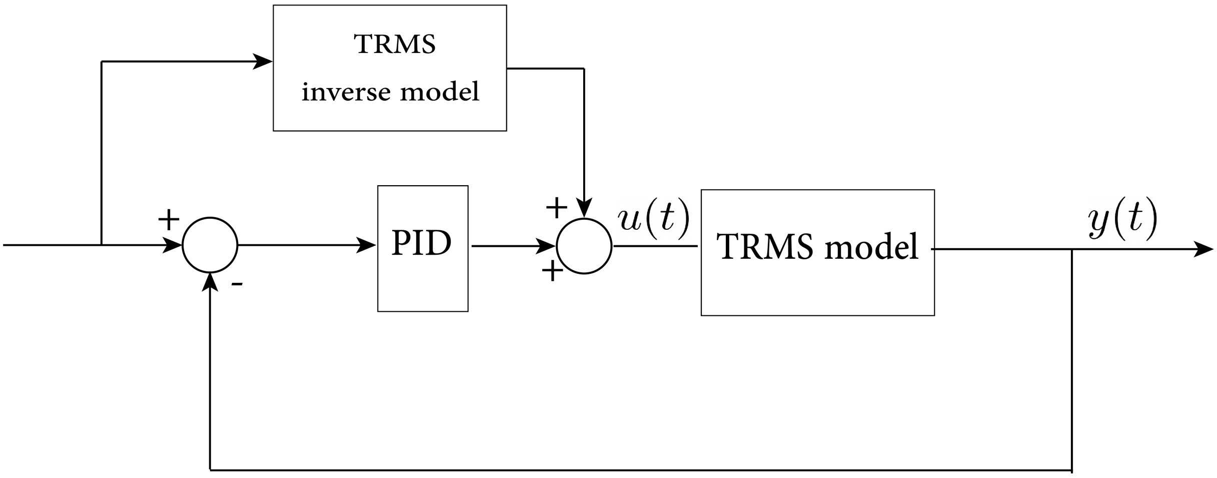

Figure 4.11: Feedforward control with feedback PID structure

The estimation structure of the inverse model is illustrated in figure 4.12.

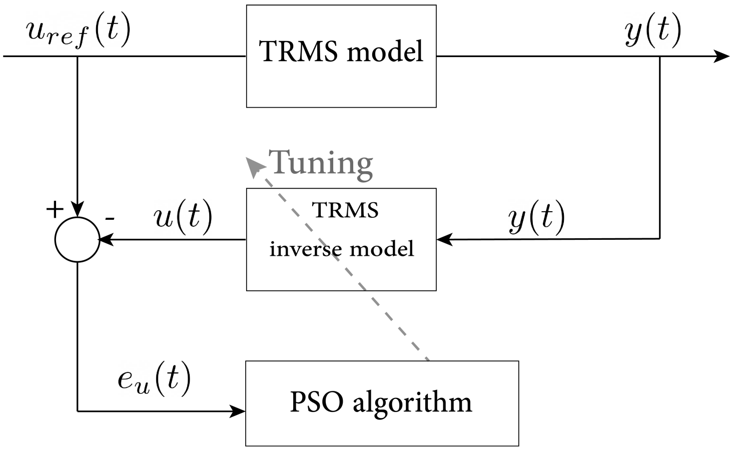

Figure 4.12: Estimation of the inverse model of the TRMS

An arbitrary mathematical model is chosen with random parameters then the error between the reference input signal and the generated input signal from the inverse model is calculated and used in an objective function by the PSO algorithm to tune the parameters of the inverse model. Finally the inverse model is used in the feedforward structure in figure 4.11 to tune the PID parameters. We fix a general ARX structure for the inverse model:

$$y(k) + \sum_{i=1}^{n} a_i y(k-i) = \sum_{j=0}^{m} b_j u(k-j) \tag{4.11}$$

where: a_i,$b_j \in \mathbb{R}$.
The work of Toha et al. in [46] showed through tests that an order of $n = 4$ and $m = 3$ gives good modeling results of the inverse model. Thus, the PSO algorithm tunes the parameters $[a_1, a_2, a_3, a_4, b_0, b_1, b_2, b_3]$.
The simulation results of the feedforward control with PID feedback is as followed:

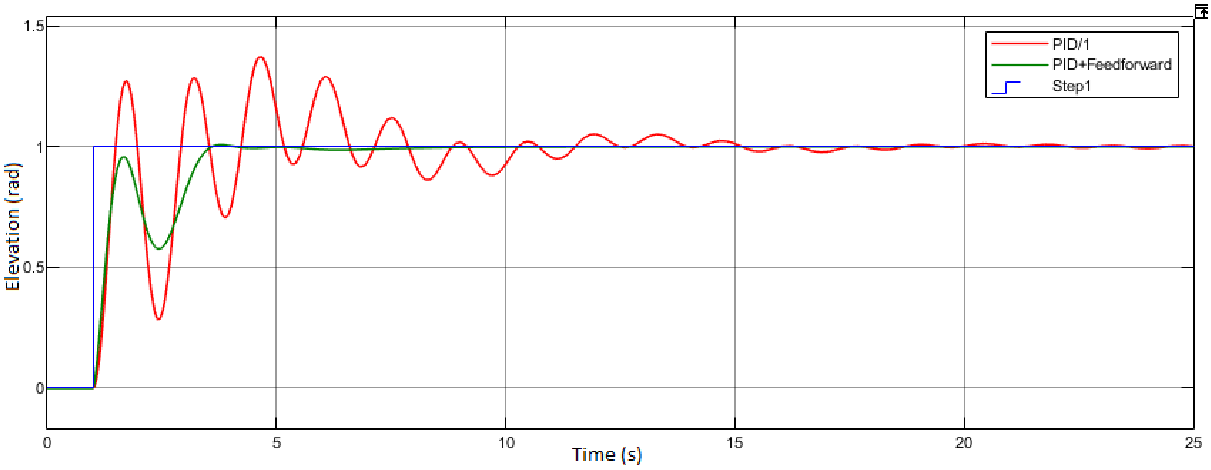

Figure 4.13: Step response of the elevation with PID+feedforward control

As we can observe the additional feedforward enhanced the quality of control, where the overshoot and oscillations got dampened.

4.4 FOPID controller using the PSO algorithm

The fractional-order PID (FOPID also noted as PI$^{\lambda}$D$^{\mu}$) is a generalization of the traditional PID controller where the derivative and integral actions are extended for non-integer orders.
We define the transfer function of the FOPID as [4]:

$$C(s) = k_p + k_i s^{-\lambda} + k_d s^{\mu}, \lambda, \mu \in \mathbb{R}^{+} \tag{4.12}$$

k_p,k_i and k_d are the gains of the proportional, integral and derivative actions respectively. And λ,μ are the real positive fractional-orders of the integral and derivative actions.
It allows us more flexibility in designing the controller compared to the integer-order PID, as shown in figure 4.14 it offers more possibilities of PID controllers design from 4 possible points into a plane of infinite points.

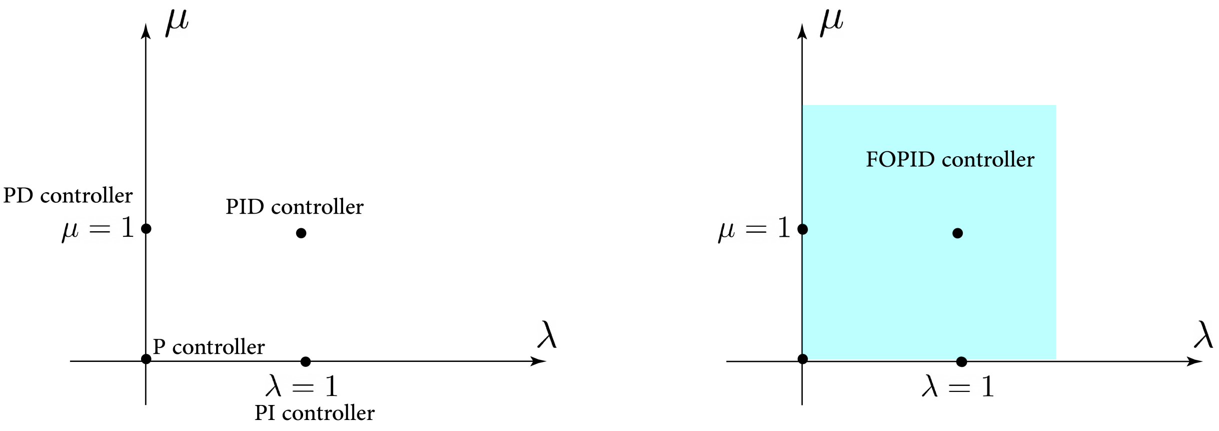

Figure 4.14: Extension of PID control from four points to a plane for FOPID

4.4.1 About FOPID tuning

In general, to estimate the parameters of the FOPID many methods were proposed in literature. A huge class of them are extensions of the traditional methods for integer-order PID tuning. They include:

- Imposing the crossover frequency w_c and the phase margin Φ_m like follows:

$$\begin{cases} |C(jw_c)G(jw_c)|_{dB} = 0dB \\ arg(C(jw_c)G(jw_c)) = -\pi + \Phi_m \end{cases} \tag{4.13}$$

- Robustness to the gain variation, by forcing an invariant open-loop phase for the crossover frequency [4]:

$$\frac{d(C(jw)G(wj))}{dw}|_{w=w_c} = 0 \tag{4.14}$$

- Assuring the pursuit of reference and disturbance rejection by imposing limit conditions for sensitivity and complementary sensitivity functions. Let A be the desired noise abatement for high frequencies and B the desired sensitivity for low frequencies, we have then:

$$\begin{array}{l} |T(s) = \frac{C(s)G(s)}{1+C(s)G(s)}|_{dB} \leq A \\ |S(s) = \frac{1}{1+C(s)G(s)}|_{dB} \leq B \end{array} \tag{4.15}$$

In our case, since we rely on the non-linear representation of the TRMS we do not have the frequency representation $G(s)$ and thus we cannot use the above methods in a direct way. We will proceed with the PSO algorithm to find the FOPID parameters.

4.4.2 Setting the PSO parameters

We will keep the same settings as the previous integer-order PID tuning, but in this case we have 2 more parameters to estimate for each controller (the fractional orders).
The FOPID for the TRMS is defined as:

$$u_v = k_{p1}e_v(t) + k_{i1}D_t^{-\lambda}e_v(t) + k_{d1}D_t^{\mu}e_v(t), \quad e_v(t) = \alpha_{vref} - \alpha_v \tag{4.16}$$

$$u_h = k_{p2}e_h(t) + k_{i2}D_t^{-\lambda}e_h(t) + k_{d2}D_t^{\mu}e_h(t), \quad e_h(t) = \alpha_{href} - \alpha_h \tag{4.17}$$

Similarly we will consider the tuning of each controller individually to decrease the time of estimation.
For the limitations of the search-space we take:

$$\begin{array}{c} 0 \leq k_p, k_i, k_d \leq 10 \\ 0 \leq \lambda, \mu \leq 2 \end{array} \tag{4.18}$$

4.4.3 Tuning results

The tuning process gives the following results:

Parameters	k_{p1}	k_{i1}	k_{d1}	λ	μ
Elevation	9.983	6.075	4.814	1	1.484
Azimuth	10	0	10	0	1.137

Table 4.2: FOPID parameters for the TRMS

The simulation of the step closed-loop response using the tuned FOPIDs is given in figures 4.15 and 4.16.

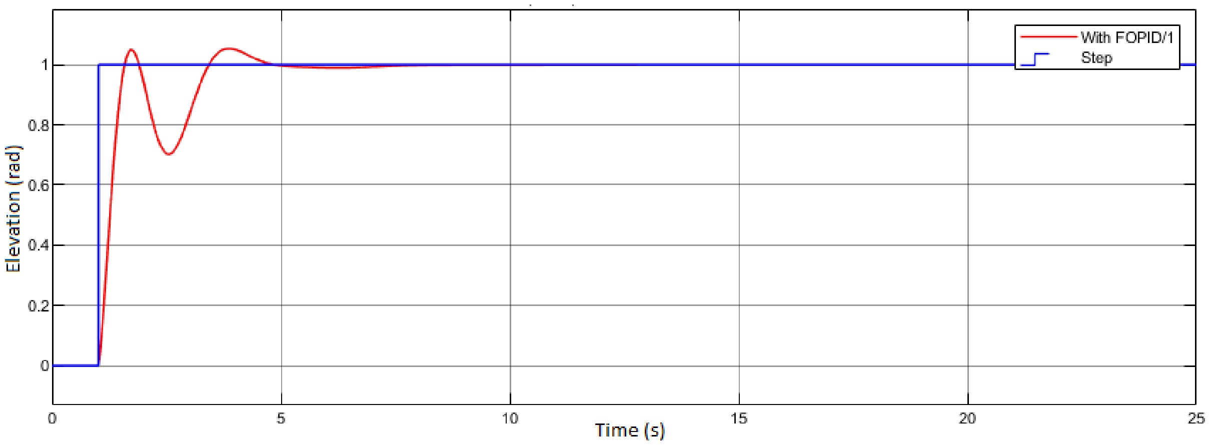

Figure 4.15: Step response of the elevation with FOPID

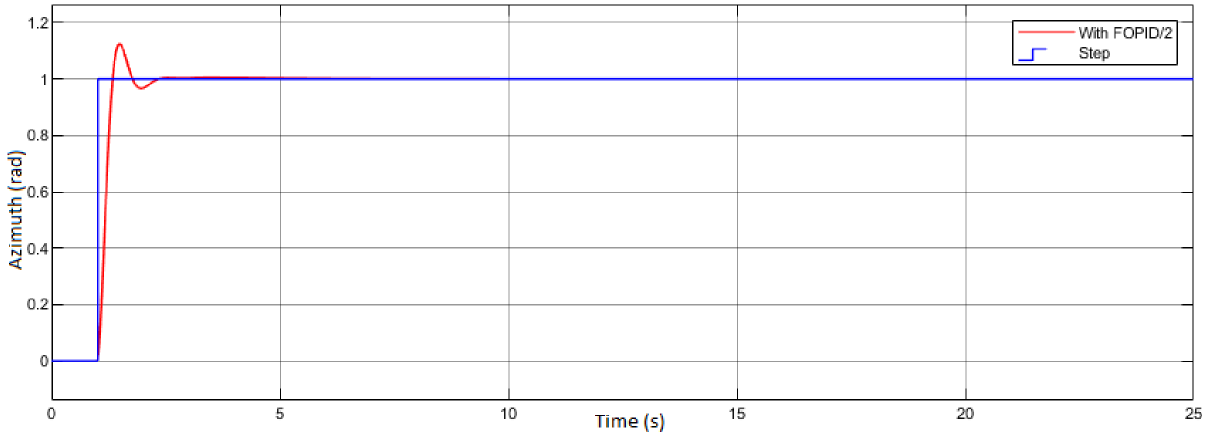

Figure 4.16: Step response of the azimuth with FOPID

The corresponding input signals are:

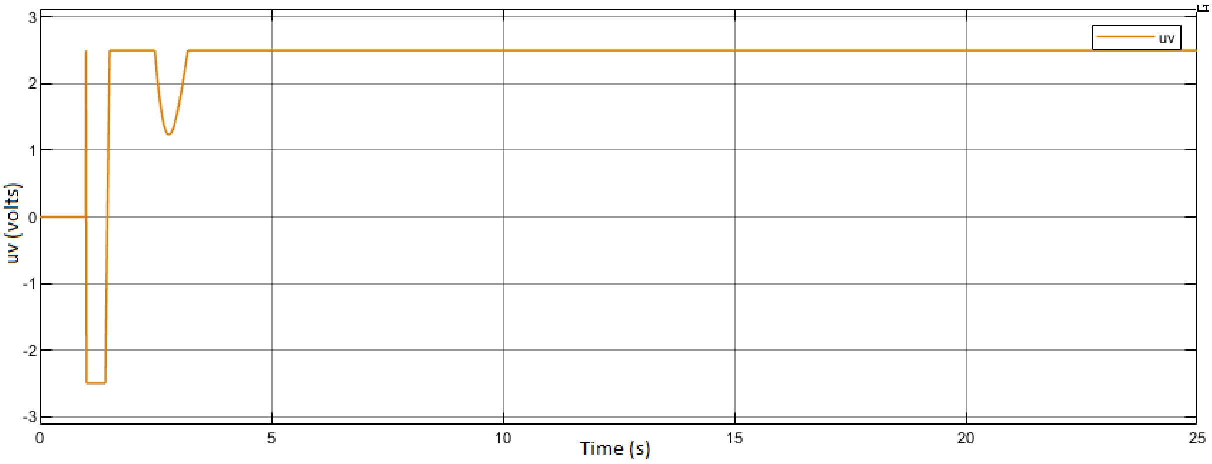

Figure 4.17: Input signal of the elevation subsystem with FOPID

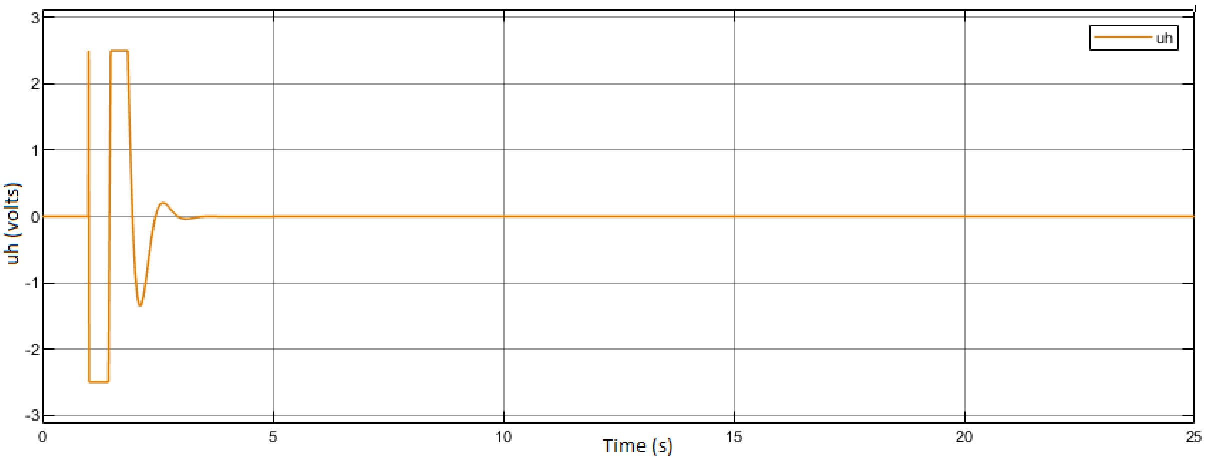

Figure 4.18: Input signal of the azimuth subsystem with FOPID

We make the same robustness tests as the ones we did for PID response. The simulation results are:

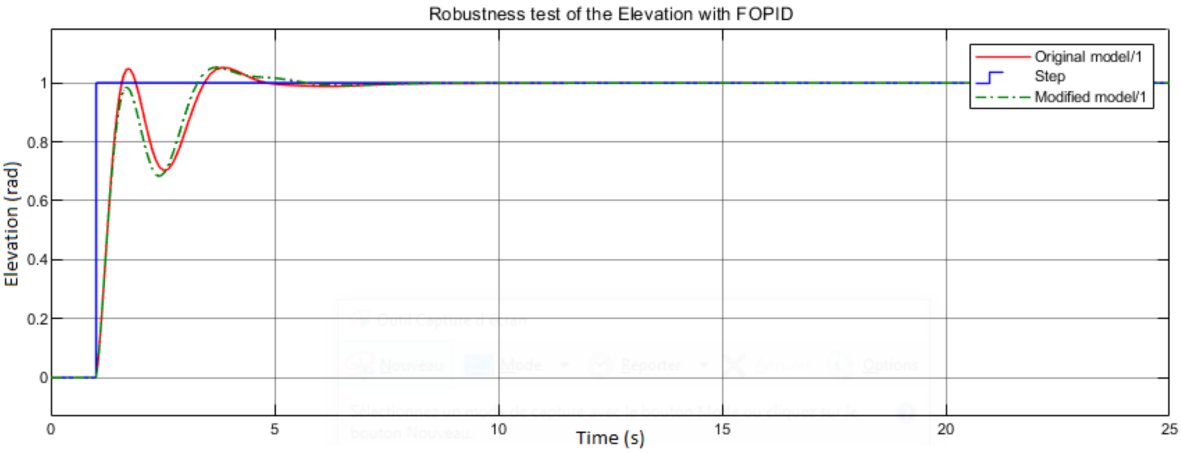

Figure 4.19: Robustness test of the FOPID for elevation subsystem

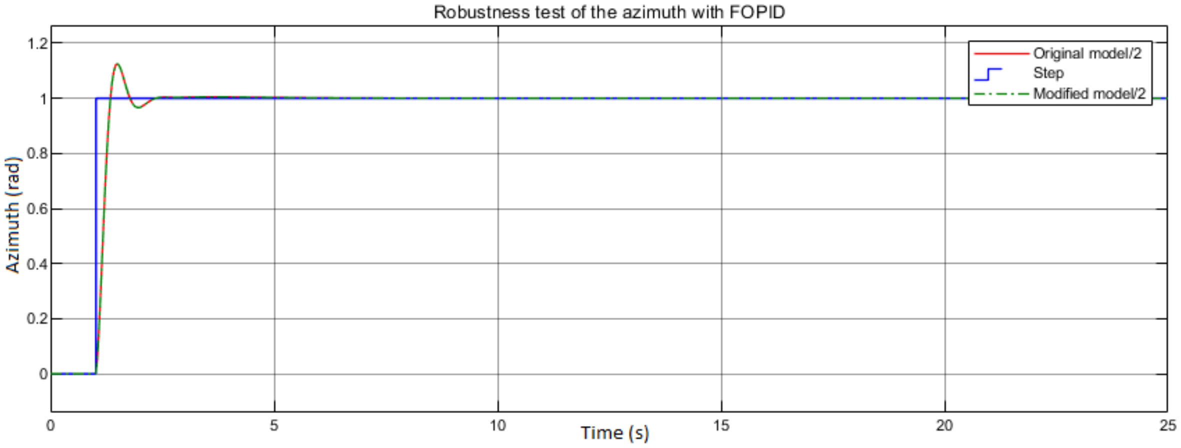

Figure 4.20: Robustness test of the FOPID for azimuth subsystem

Comparing to the open-loop step response we can observe:

- Both the elevation and azimuth converge.
- The optimal integral controller for the elevation is of integer-order since $\lambda = 1$.
- The proportional and derivative gains k_{p2} and k_{d2} for the azimuth are maxed out. Also the integral gain is zero since the subsystem already shown to have an integration.
- Increasing the imposed limits of the gains increases drastically computation time but did not result in any significant enhancement of the quality of control, thus limiting the gains between 0 and 10 is a good choice.
- The oscillations of the elevation have been dampened and the response becomes smoother.
- The input signals frequently reach the physical limitations of $\pm 2.5v$, this implies that if the system tolerates a wider voltage range the inputs would be less limited and result in better control quality.

4.5 comparison of the results

Now we put the PID control results side to side with the ones obtained using the FOPID in order to compare the performances of these two controllers for our TRMS.
In addition to figures 4.21 and 4.22 that show the step response of the system for the PID and FOPID control in the same graph, we consider the figures 4.23 and 4.24 that show a sneakier input reference, a random square signal, for a better comparison perspective we will also add a sinusoidal disturbance of amplitude $A = 0.1$ to these responses to study the disturbance rejection ability of both the PID and FOPID controllers.

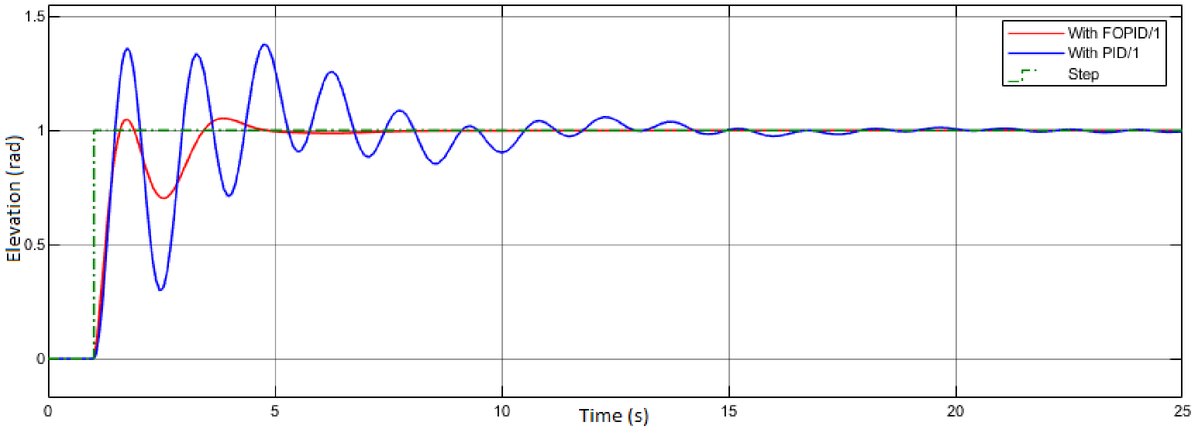

Figure 4.21: Comparative step response of the elevation with PID and FOPID controllers

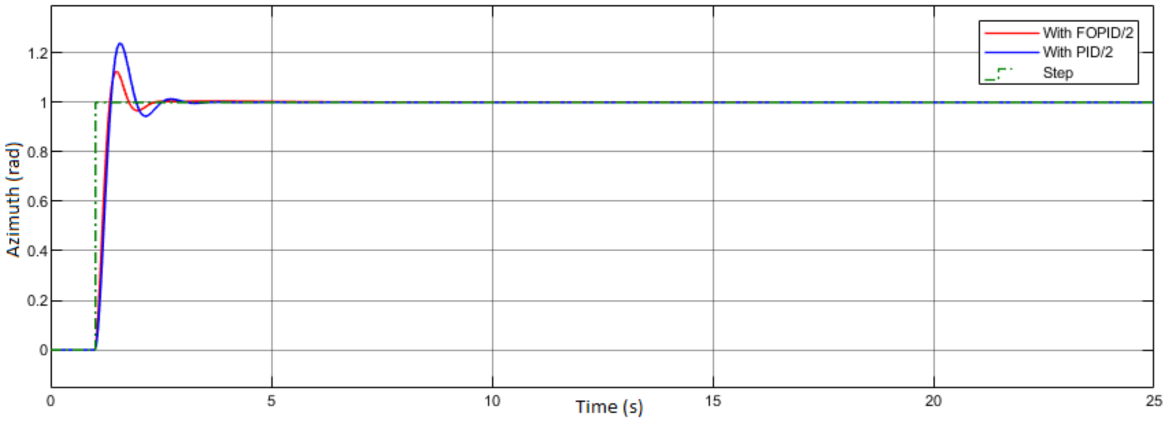

Figure 4.22: Comparative step response of the azimuth with PID and FOPID controllers

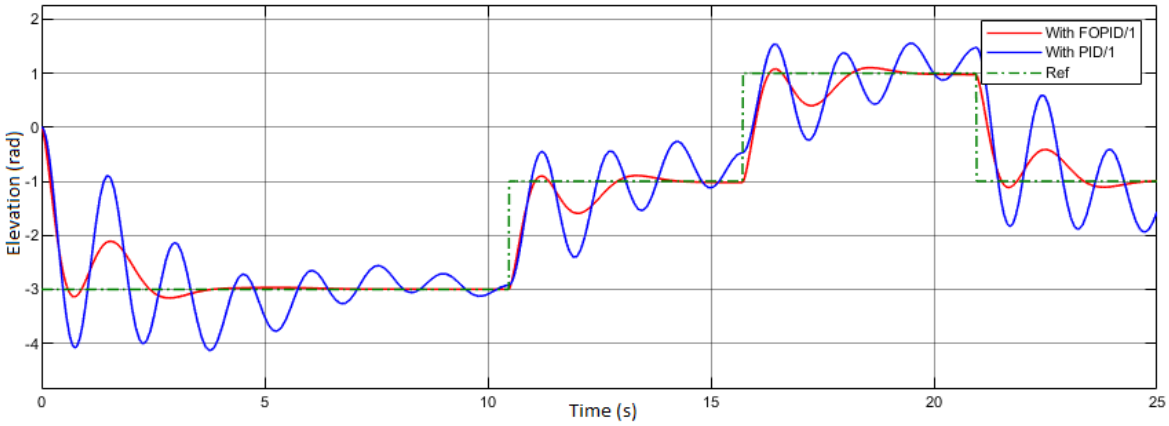

Figure 4.23: Comparative random signal response of the elevation with PID and FOPID controllers

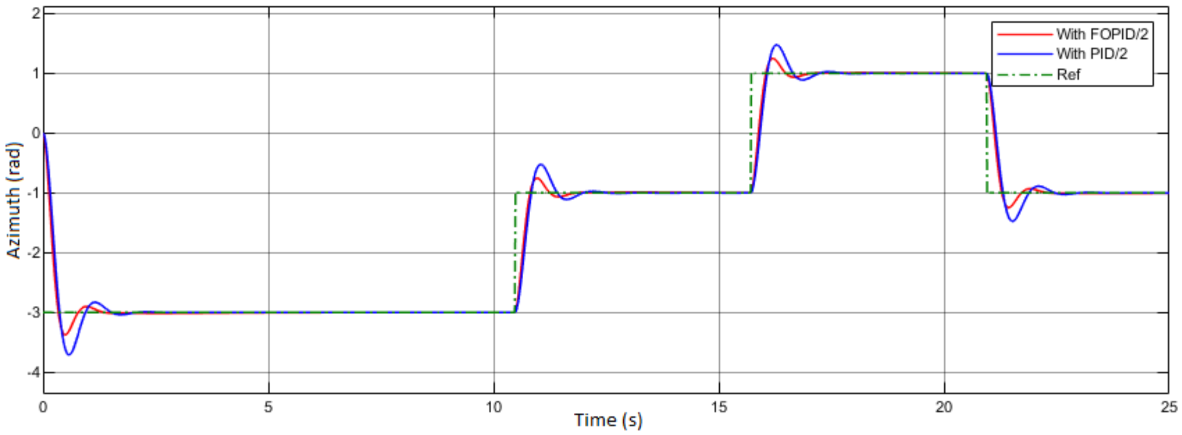

Figure 4.24: Comparative random signal response of the azimuth with PID and FOPID controllers

The control results obtained using the FOPID controllers are superior in quality to those obtained using PID controllers such that:

- The FOPID dampens the oscillations of the elevation subsystem instantly and entirely while the PID only reduced them partially and over time.
- The responses obtained using the FOPID are faster and smoother.
- The overshoot of the FOPID responses is minimal, especially in the elevation subsystem which is significantly smaller than that with PID controller.
- For a random variable reference, the FOPID control shows an acceptable reference tracking while the response with the PID controller cannot keep up with it and gives an altered result in the case of elevation.
- FOPID is more robust for modeling error.
- The PID controller needs a feedforward control to get a similar controlling quality to the FOPID in the case of the TRMS.
- When it comes to the external disturbance, both controllers reject it perfectly.
- The input signals in the case of the FOPID show less oscillations and sudden changes that cause damage to the physical system.

From these results, it is clear that the FOPID is a more accurate controller than the integer-order PID. However, it is worth mentioning that the estimation of the FOPID parameters using the PSO algorithm is significantly more time consuming than the PID estimation due to the additional parameters. Though, this is not an issue when we use powerful calculators for example, because the quality of the results overrides the additional computation delay.
We note as well that better controllers can be tuned for a larger search-space, a bigger number of iterations and particles but all in the expense of longer computation time.

Conclusion

In this chapter, we designed PID and FOPID controllers for the TRMS using the particle swarm optimization, a metaheuristic algorithm that estimates the gains and fractional orders of the controllers.
We also compared the results obtained from the closed-loop response in both PID and FOPID feedback controls of the TRMS, in which the FOPID showed a superior quality of control in most aspects: oscillation dampening, time response, overshoot reduction, robustness to modeling error and reference tracking.
The FOPID control of the TRMS, a highly nonlinear system gives promising results overall, nonetheless it has two major flaws in the case of this class of systems. The first consists in the time of computation spent by the metaheuristic algorithms like the PSO and GA to estimate the controller's parameters. And the second and most important, consists in the incapability of establishing online direct control for the TRMS since the optimization algorithms are time consuming, thus the present time control cannot synchronize with the present state of the system.

Chapter 5

General Conclusion

Fractional calculus generalizes the integro-differential operator for any real positive order which opens up a wide range of applications. In this project, we presented some fundamental definitions of the theory which include Riemann-Liouville's integral and three distinct definitions of derivatives: Riemann-Liouville's derivative, Caputo's derivative and Gründwald-Leitnikov's derivative, that is because each one of these latter definitions has its own advantages and disadvantages in applications. This type of calculus brings many benefits to several scientific fields like control theory, our main interest. Therefore, we defined the necessary properties and tools that allow us to make a transition from classical integer-order control theory to a fractional-order version, some of these tools are Laplace transform of the integro-differential operator and Oustaloop approximation. The theoretical part covers as well some basic concepts of control theory for fractional-order systems like mathematical representations and stability.
As for the application section of this project, it includes two major parts: Identification and Control of a Twin Rotor MIMO System (TRMS).
To identify our system, the TRMS, we had to go through a detailed physical study of its components and their behaviour to establish some mathematical relations describing the latter. Those relations were used as a base to build a simulation model that proved to be valid when compared to the one given by the manufacturer [20]. Afterwards, three mathematical models were identified: a linear integer-order representation which consists of a traditional transfer function, a linear fractional-order representation which is a fractional-order transfer function and a nonlinear integer-order representation by a state-space model. Several challenges were faced when trying to apply traditional identification techniques for fractional estimation, where they call for some adjustments of the algorithms' structure. Simulations of the identified models show that the fractional-order transfer function approximated the real system more accurately than its integer-order representation. However, both linear representations were flawed due the high-order nonlinearity of our system, therefore we proceeded using the state-space model in the rest of the project. Methods for nonlinear fractional-order modeling were discussed, but not applied due the complexity of their application in the TRMS and satisfying results offered by the integer-order state-space.
In the control part, we focused on PID control. Two versions of PID controllers were designed: a traditional integer-order PID and a fractional-order PID (FOPID). Due to the nonlinear nature of the system, we used the Particle Swarm Optimization (PSO) algorithm in order to find the optimal gains of our controllers (and fractional-orders as well for the FOPID) that result in the best control. Both controllers achieved their goal where the two subsystem converges to the desired reference, but with different control quality where the FOPID proved a superiority in control quality in the following aspects: oscillation and overshoot dampening, reference tracking, robustness to modeling errors, input signal smoothness and flexibility of design. Results of simulation showed as well that the PID controller needed a feedforward to get a control quality similar to that using a FOPID for the TRMS.

Throughout this project, fractional-order systems proved their efficiency in modeling and controlling nonlinear systems like the TRMS and showed huge flexibility when it comes to controller's design.

Bibliography

[1] U. N. Katugampola, “New approach to a generalized fractional integral,” *Applied mathematics and computation*, vol. 218, no. 3, pp. 860–865, 2011.

[2] J. Liouville, *Mémoire sur quelques questions de géométrie et de mécanique, et sur un nouveau genre de calcul pour résoudre ces questions.* 1832.

[3] M. A. Matlob and Y. Jamali, “The concepts and applications of fractional order differential calculus in modeling of viscoelastic systems: A primer,” *Critical Reviews™ in Biomedical Engineering*, vol. 47, no. 4, 2019.

[4] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu-Batlle, *Fractional-order systems and controls: fundamentals and applications.* Springer Science & Business Media, 2010.

[5] S. Das, *Functional fractional calculus.* Springer, 2011, vol. 1.

[6] I. Podlubny, A. Chechkin, T. Skovranek, Y. Chen, and B. M. V. Jara, “Matrix approach to discrete fractional calculus ii: Partial fractional differential equations,” *Journal of Computational Physics*, vol. 228, no. 8, pp. 3137–3153, 2009.

[7] A.-L. Cauchy, *Résumé des leçons données à l’école royale polytechnique sur le calcul infinitésimal.* Imprimerie royale, 1823.

[8] M. Caputo, “Linear models of dissipation whose q is almost frequency independent—ii,” *Geophysical Journal International*, vol. 13, no. 5, pp. 529–539, 1967.

[9] M. D. Ortigueira and F. Coito, “From differences to derivatives,” *Fractional Calculus and Applied Analysis*, vol. 7, no. 4, p. 459, 2004.

[10] C. Li and W. Deng, “Remarks on fractional derivatives,” *Applied mathematics and computation*, vol. 187, no. 2, pp. 777–784, 2007.

[11] I. Petráš, *Fractional-order nonlinear systems: modeling, analysis and simulation.* Springer Science & Business Media, 2011.

[12] D. Matignon, “Stability results for fractional differential equations with applications to control processing,” in *Computational engineering in systems applications*, Citeseer, vol. 2, 1996, pp. 963–968.

[13] B. Ross, *Fractional calculus and its applications: proceedings of the international conference held at the University of New Haven, June 1974.* Springer, 2006, vol. 457.

[14] J. Sabatier, “Fractional order models are doubly infinite dimensional models and thus of infinite memory: Consequences on initialization and some solutions,” *Symmetry*, vol. 13, no. 6, p. 1099, 2021.

[15] A. Djouambi, A. Charef, and A. V. Besancon, “Optimal approximation, simulation and analog realization of the fundamental fractional order transfer function.,” *International Journal of Applied Mathematics & Computer Science*, vol. 17, no. 4, 2007.

[16] A. Oustaloup, F. Levron, B. Mathieu, and F. M. Nanot, “Frequency-band complex noninteger differentiator: Characterization and synthesis,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, no. 1, pp. 25–39, 2000.

[17] L. Chen, B. Basu, and D. McCabe, “Fractional order models for system identification of thermal dynamics of buildings,” *Energy and Buildings*, vol. 133, pp. 381–388, 2016.

[18] R. L. Bagley and R. Calico, “Fractional order state equations for the control of viscoelasticallydamped structures,” *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pp. 304–311, 1991.

[19] S. Toha, M. Tokhi, and Z. Hussain, “Anfis modelling of a twin rotor system,” in *UKACC International Conference on Control*, 2008, P125.

[20] “Feedback, twin rotor mimo system,” *Advanced Teaching Manual 1*, vol. 33-007-4M5,

[21] H. N. Nguyen, “Modelling, simulation, and calibration of twin rotor mimo system,” M.S. thesis, Universitat Politècnica de Catalunya, 2007.

[22] P. Chalupa, J. Přikryl, and J. Novák, “Modelling of twin rotor mimo system,” *Procedia Engineering*, vol. 100, pp. 249–258, 2015.

[23] A. Rahideh, M. Shaheed, and H. Huijberts, “Dynamic modelling of a trms using analytical and empirical approaches,” *Control Engineering Practice*, vol. 16, no. 3, pp. 241–259, 2008.

[24] K. Kothari, “System identification based on fractional order models,” 2020.

[25] K. Levenberg, “A method for the solution of certain non-linear problems in least squares,” *Quarterly of applied mathematics*, vol. 2, no. 2, pp. 164–168, 1944.

[26] Y. Engel, S. Mannor, and R. Meir, “The kernel recursive least-squares algorithm,” *IEEE Transactions on signal processing*, vol. 52, no. 8, pp. 2275–2285, 2004.

[27] D. Idiou, A. Charef, and A. Djouambi, “Identification récursive des systèmes à dérivée fractionnaire,” in *3rd International Conference On Systems And Processing Information*, 2013.

[28] A. Djouambi, A. Charef, and T. Bouktir, “Fractional order robust control and pî 1̂ â m̂ b̂ d̂ ad̂ m̂ u controllers,” *WSEAS transactions on circuits and systems*, vol. 4, no. 8, p. 850, 2005.

[29] O. Stark, M. Pfeifer, and S. Hohmann, “Parameter and order identification of fractional systems with application to a lithium-ion battery,” *Mathematics*, vol. 9, no. 14, p. 1607, 2021.

[30] A. Tepljakov, E. Petlenkov, and J. Belikov, “Fomcom: A matlab toolbox for fractional-order system identification and control,” *International Journal of Microelectronics and computer science*, vol. 2, no. 2, pp. 51–62, 2011.

[31] T. J. Archdeacon, *Correlation and regression analysis: a historian's guide*. Univ of Wisconsin Press, 1994.

[32] M. Ilyas, N. Abbas, M. UbaidUllah, W. A. Imtiaz, M. Shah, and K. Mahmood, "Control law design for twin rotor mimo system with nonlinear control strategy," *Discrete Dynamics in Nature and Society*, vol. 2016, 2016.

[33] R. Stanisławski, K. J. Latawiec, M. Gałek, and M. Łukaniszyn, "Modeling and identification of a fractional-order discrete-time siso laguerre-wiener system," in *2014 19th International Conference on Methods and Models in Automation and Robotics (MMAR)*, IEEE, 2014, pp. 165–168.

[34] L. Sersour, T. Djamah, and M. Bettayeb, "Nonlinear system identification of fractional wiener models," *Nonlinear Dynamics*, vol. 92, no. 4, pp. 1493–1505, 2018.

[35] K. Kothari, U. Mehta, V. Prasad, and J. Vanualailai, "Identification scheme for fractional hammerstein models with the delayed haar wavelet," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 3, pp. 882–891, 2020.

[36] W. Allafi, I. Zajic, K. Uddin, and K. J. Burnham, "Parameter estimation of the fractional-order hammerstein–wiener model using simplified refined instrumental variable fractional-order continuous time," *IET Control Theory & Applications*, vol. 11, no. 15, pp. 2591–2598, 2017.

[37] L. Sersour, T. Djamah, and M. Bettayeb, "Nonlinear system identification of fractional wiener models," *Nonlinear Dynamics*, vol. 92, no. 4, pp. 1493–1505, 2018.

[38] S. Bennett, "The past of pid controllers," *Annual Reviews in Control*, vol. 25, pp. 43–53, 2001.

[39] G. Štimac, S. Braut, and R. Žigulić, "Comparative analysis of pso algorithms for pid controller tuning," *Chinese Journal of Mechanical Engineering*, vol. 27, no. 5, pp. 928–936, 2014.

[40] W.-D. Chang and S.-P. Shih, "Pid controller design of nonlinear systems using an improved particle swarm optimization approach," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 11, pp. 3632–3639, 2010.

[41] J.-G. Juang, M.-T. Huang, and W.-K. Liu, "Pid control using presearched genetic algorithms for a mimo system," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 5, pp. 716–727, 2008.

[42] W.-D. Chang, "Nonlinear system identification and control using a real-coded genetic algorithm," *Applied Mathematical Modelling*, vol. 31, no. 3, pp. 541–550, 2007.

[43] A. Nickabadi, M. M. Ebadzadeh, and R. Safabakhsh, "A novel particle swarm optimization algorithm with adaptive inertia weight," *Applied soft computing*, vol. 11, no. 4, pp. 3658–3670, 2011.

[44] X. Cai and Y. Tan, "A study on the effect of vmax in particle swarm optimisation with high dimension," *International Journal of Bio-Inspired Computation*, vol. 1, no. 3, pp. 210–216, 2009.

[45] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[46] S. F. Toha, I. Abd Latiff, M. Mohamad, and M. O. Tokhi, “Parametric modelling of a trms using dynamic spread factor particle swarm optimisation,” in *2009 11th International Conference on Computer Modelling and Simulation*, IEEE, 2009, pp. 95–100.