ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

A NEW ANTI-WINDUP STRATEGY FOR FRACTIONAL ORDER PI CONTROLLERS

M.Sc. THESIS

Muhammed Ali ELAYDIN

Department of Control and Automation Engineering

Control and Automation Engineering Programme

JUNE 2025

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

A NEW ANTI-WINDUP STRATEGY FOR FRACTIONAL ORDER PI CONTROLLERS

M.Sc. THESIS

Muhammed Ali ELAYDIN
(504221114)

Department of Control and Automation Engineering

Control and Automation Engineering Programme

Thesis Advisor: Prof. Dr. Müjde GÜZELKAYA

JUNE 2025

ISTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

KESİRLİ MERTEBE PI KONTROLÖRLER İÇİN YENİ BİR İNTEGRAL YIĞILMASI KARŞITI STRATEJİ

YÜKSEK LİSANS TEZİ

Muhammed Ali ELAYDIN
(504221114)

Kontrol ve Otomasyon Mühendisliği Anabilim Dalı

Kontrol ve Otomasyon Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Müjde GÜZELKAYA

HAZİRAN 2025

Muhammed Ali Elaydın, a M.Sc. student of İTU Graduate School student ID 504221114, successfully defended the thesis/dissertation entitled “A New Anti-Windup Strategy For Fractional Order PI Controllers”, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Prof. Dr. Müjde GÜZELKAYA** Istanbul Technical University	.....
Jury Members :	**Assis. Prof. Dr. Erhan YUMUK** Istanbul Technical University	.....
	Prof. Dr. Cenk ULU Yıldız Technical University	.....

Date of Submission : 28 May 2025
Date of Defense : 19 June 2025

To those who inspire me,

FOREWORD

I would like to express my sincere gratitude to Prof. Dr. Müjde GÜZELKAYA for her invaluable contributions to my academic journey since the day I was first introduced to control engineering. It has also been a privilege to work alongside Dr. Erhan YUMUK while exploring the intersection of fractional calculus and control.

I extend my heartfelt thanks to my seniors at the TÜBİTAK RUTE Robotics and Intelligent Systems Department, especially Yekta YEŞİLYURT, whose unwavering moral support has sustained me throughout this challenging endeavor.

Finally, I am deeply grateful to my family and friends, who have stood by me unconditionally throughout my life.

May 2025

Muhammed Ali ELAYDIN
Control and Automation Engineer

TABLE OF CONTENTS

Page

ABBREVIATIONS

FOPDT **:** First Order Plus Time Delay

FOPI **:** Fractional Order Proportional Integrator

FOPD **:** Fractional Order Proportional Derivative

FOPID **:** Fractional Order Proportional Integrator Derivative

IMC **:** Internal Model Control

LIST OF TABLES

Page

LIST OF FIGURES

Page

A NEW ANTI-WINDUP STRATEGY FOR FRACTIONAL ORDER PI CONTROLLERS

SUMMARY

Fractional calculus originated in 1695 and was later expanded by various mathematicians. Among the most commonly used fractional-order definitions are the operators proposed by Riemann-Liouville, Grünwald-Letnikov, and Caputo. These formulations extend classical differentiation and integration to non-integer orders, allowing for more flexible and detailed mathematical modeling. In control engineering, fractional-order controllers offer enhanced design flexibility, enabling the development of controllers with improved performance and robustness characteristics. Implementing fractional-order controllers requires certain approximations. One of the most widely adopted methods is proposed by Oustaloup. This aims to construct an integer-order transfer function that closely emulates the frequency response of a fractional-order operator.

In control system implementations, various physical constraints must be considered, one of the most significant being actuator limits. These limitations can lead to a phenomenon known as integral windup, which occurs when the control signal produced by the controller exceeds the actuator's physical capabilities. The integrator continues accumulating error, resulting in an excessively large control signal. This mismatch degrades the system response by increasing overshoot, prolonging settling time, and potentially leading to instability.

To address this issue, numerous anti-windup techniques have been proposed in the literature. This study focuses on a comparison of four leading and widely used methods: back calculation, automatic reset configuration (ARC), ARC with fractional filter and fractional integral order reset. While most of these techniques are formulated based on integer-order PID structures, some have been adapted or developed for fractional-order controllers. However, these often fall short of leveraging the flexibility offered by fractional calculus.

This thesis, firstly, focused on key factors contributing to integral windup. Among these, saturation limits define the physical constraints of the system, while controller parameters, including the proportional gain, integral gain, and integral order, also play a critical role. Each factor influencing the growth of the control signal can worsen windup; however, the effect of the integrator, particularly the order of integrator, is found to be especially significant and is thoroughly investigated in this thesis.

To evaluate anti-windup performance, existing methods are tested on first-order plus dead time (FOPDT) systems, which are widely used in control applications due to their ability to capture the dominant behavior of various dynamic systems, facilitate system identification, and approximate higher-order models. The proposed stategy is designed and generalized specifically for such systems, using their characteristic parameters.

Two key performance criteria are defined for windup mitigation: the tracking error between the reference and system output, and the magnitude of saturation error. Based on these, a method is proposed in which the fractional integral order is adapted online. By dynamically updating the integral order, the proposed structure improves system performance while suppressing integral windup.

The update rule for the integral order relies on two criteria: the tracking error and the saturation error. The variation of these metrics is analyzed in relation to the FOPDT system parameters: the process gain, time constant, and time delay. These relationships are then incorporated into the update rule, allowing the proposed strategy to be expressed as a generalized formulation for FOPDT systems.

In conclusion, the proposed strategy has been tested alongside existing approaches on various FOPDT systems. The comparison is based on four performance metrics: overshoot, rise time, settling time, and peak time. The results demonstrate that the proposed approach provides a notable advantage in overshoot suppression and generally yields a more stable system response compared to the other methods.

KESİRLİ MERTEBE PI KONTROLÖRLER İÇİN YENİ BİR İNTEGRAL YIĞILMASI KARŞITI STRATEJİ

ÖZET

Kesirli mertebe hesabı, ilk olarak 1695 yılında Leibniz ile L'Hopital arasındaki mektuplaşmalarda ortaya çıkmış ve zamanla matematikçilerin yoğun ilgisini çeken bir araştırma alanı haline gelmiştir. Riemann-Liouville, Grünwald-Letnikov ve Caputo tarafından tanımlanan operatörler, bu alanda en yaygın kullanılan kesirli mertebe tanımlarıdır. Bu tanımlar sayesinde, klasik türev ve integral işlemleri kesirli mertebelere genellenebilmiş, böylece matematiksel tanımların daha esnek ve ayrıntılı ifade edilebilmesine olanak tanınmıştır. 20. yüzyıldan itibaren, kesirli mertebe hesabının mühendislik disiplinlerinde uygulamaları görülmeye başlanmıştır. Başta kontrol, sinyal işleme ve modelleme olmak üzere birçok alanda kesirli mertebe hesabından faydalanılmıştır.

Klasik PID kontrolör yapısı, oransal, integral ve türev katsayılarının ayarlanmasına dayanır. Ancak kesirli mertebe kontrolörlerde integral ve türev işlemleri, klasik anlamlarının yanı sıra kesirli mertebelerle tanımlanabilir hale gelmiştir. Bu durum, kontrolör tasarımına iki ek parametre daha kazandırmıştır: integral mertebesi ve türev mertebesi. Örneğin, bir PI kontrolör yapısında klasik olarak iki parametre (oransal ve integral kazanç) ayarlanabilirken, kesirli mertebe PI kontrolörlerde bu sayı üçe çıkar. Bu sayede, kontrolör tasarımında esneklik kazanılmış olur ve özellikle performans ve gürbüzlük açısından daha iyi kontrolörler tasarlanabilir.

Kesirli mertebeli kontrolörlerin gerçeklenmesi, belirli yaklaşımlar ile mümkündür. Bu amaçla geliştirilen yaklaşımlar arasında en yaygın olanı, Oustaloup tarafından sunulan yöntemdir. Bu yaklaşım, kesirli mertebeli bir operatörün frekans cevabını en iyi şekilde taklit eden tam sayı mertebeli bir transfer fonksiyonu elde etmeyi hedefler.

Kontrol sistemleri çoğunlukla lineer zamanla değişmeyen (LTI) modellerle ifade edilebilir. Ancak bu modeller fiziksel sistemlere uygulanmak istendiğinde, doğrusal olmayan birtakım kısıtlar kendini gösterir. Bu doğrusal olmayan etkilerden biri, aktüatör doygunluğu (saturasyon) olup, kontrol sinyallerinin belirli fiziksel sınırların ötesine geçmesini engeller. Bu durum, kontrol teorisinde "integral yığılması" (integral windup) olarak bilinen probleme yol açar. İntegral yığılması, kontrolörün ürettiği kontrol sinyali ile sistemin fiziksel olarak uygulayabileceği giriş arasında uyuşmazlığı tanımlar. Bu, kontrolörün integral terimi hata sinyalini biriktirerek kontrol sinyalinin aşırı büyümesine sebep olur. Sonuç olarak, sistem cevabı bozulur, aşım artar, yerleşme süresi uzar ve hatta sistem kararsızlığa gidebilir.

Bu problemi önlemek için literatürde çeşitli integral yığılması önleyici yöntemler geliştirilmiştir. Bu yöntemlerin büyük bir bölümü tamsayı mertebeli PID kontrolör yapısı esas alınarak formülize edilmiştir. Bazı yöntemler ise kesirli mertebeli kontrolörler için uyarlanmış ya da geliştirilmiştir yötemlerdir. Son dönemde, kesirli mertebeli yapıların karakteristik özelliklerinden faydalanarak, yığılma problemini daha etkin şekilde bastırmayı hedefleyen yöntemler de görülebilmektedir. Fakat bu

yaklaşımlar da kesirli mertebe hesabının sağladığı esnekliği yeterince kullanmaktan uzaktır.

İntegral yığılması sorununun temelinde, kontrolörün integral bileşenleri yer alır. Tamsayı mertebeli kontrolörlerde bu durum yalnızca integral kazancının manipüle edilebilmesi ile sınırlı biçimde yönetilebilirken, kesirli mertebeli kontrolörlerde integral mertebesi de tasarım parametresi olarak değerlendirilerek daha etkili çözümler geliştirilebilir. Böylece, kontrol performansı ve integral yığılması probleminin çözümü bir arada değerlendirilmiş olunur.

Bu çalışmada, integral yığılmasına etki eden başlıca faktörler ele alınmıştır. Bunlardan biri, sistemin fiziksel kısıtlarını belirleyen saturasyon limitleridir. Diğerleri ise kontrolör parametreleridir: oransal katsayı, integral kazanç ve integral mertebesi. Kontrol sinyalinin aşırı büyümesine neden olan her unsur, yığılma problemine katkı sağlar. Fakat kontrolörün integral katkısından gelen etkiler, özellikle önem arz eder. Özellikle integral mertebesinin kritik etkisi çalışmada değerlendirilmiştir.

Bu kapsamda, literatürden karşılaştırmak üzere beş yöntem seçilmiştir. Birincisi, koşullu entegrasyon yöntemidir. Koşullu entegrasyon temelde, saturasyon varlığında integral etkisini sıfırlayarak kontrolörün oransal kontrolör olarak davranmasını önerir. İkinci yöntem, geri hesaplama yöntemidir. Geri hesaplama yaklaşımında, saturasyon hatası hesaplanır ve bu nisbette integral kazancı etkisi zayıflatılır. Üçüncü yöntem, otomatik sıfırlama yapılandırması (automatic reset configuration) yöntemidir. Bu yöntem ise, kontrol çevrimi içerisine bir model saturasyon bloğu eklemeyi ve integral etkisini geri besleme yolu ile saturasyonu denetleyecek hale getirmeyi önerir. Dördüncü yöntem, otomatik sıfırlama yapılandırması (automatic reset configuration) yönteminde önerilen konfigürasyonun önüne bir kesirli mertebeli filtre yerleştirir. Filtre, parametrelerini kontrolörden ve tasarım kriterlerinden alır. Beşinci ve son yöntem, kesirli integratör sıfırlama yöntemidir. Bu yöntem, integral mertebesini saturasyon varlığında sıfırlama; saturasyonun olmadığı durumlarda ilk tasarım değerine getirme üzerine kuruludur. Bu beş yöntem arasında integral mertebesinin önemi ve etkisini göz ardı etmeyen yegane yöntem beşincisidir.

Çalışmada, integral yığılmasını önlemek üzere literatürde mevcut olan yöntemler birinci mertebeden ölü zamanlı sistemler üzerinde karşılaştırılmış; önerilen yöntem ise bu sistemlerin karakteristik parametreleriyle genelleştirilmiştir. Birinci mertebeden ölü zamanlı sistem modeli, kontrolde birçok açıdan önemli ve yaygın kullanıma sahiptir. Bunun nedenleri arasında; dinamik sistem davranışlarının bu tür bir modelle temsil edilebilme kabiliyeti, sistem tanıma için kullanılabilme kolaylığı, yüksek mertebeden sistem modellerinin birinci mertebeden ölü zamanlı sistem modeli ile temsil edilebilmesi gibi nedenler vardır.

Yığılmanın baskılanması açısından önemli iki performans ölçütü belirlenmiştir: referans ile sistem yanıtı arasındaki fark ve saturasyon hatasının büyüklüğü. Bu ölçütler temel alınarak, integral mertebesinin çevrim içi olarak uyarlanabildiği yeni bir yöntem geliştirilmiştir. Önerilen yöntemde, kontrolör integral mertebesini gerçek zamanlı olarak değiştirerek hem sistem performansını artırmakta hem de integral yığılmasını bastırmaktadır.

İntegral mertebesini güncelleme kuralı iki kritere bağlanmıştır: referans takip hatası ve saturasyon hatası. Bu iki değerin değişiminin birinci mertebeden ölü zamanlı sistem parametreleri olan sistem kazancı, zaman sabiti ve ölü zaman değerleri ile ilişkileri analiz edilmiştir. Bu ilişkiler nisbetinde, güncelleme kuralına sistem parametreleri

dahil edilmiştir. Böylece önerilen yöntem, birinci mertebeden ölü zamanlı sistemler için genelleştirilmiş bir ifade halinde sunulmuştur.

Sonuç olarak, önerilen yöntem literatürdeki diğer yöntemlerle birlikte çeşitli birinci mertebeden ölü zamanlı sistemler üzerinde test edilmiştir. Karşılaştırmalar; aşım, yükselme zamanı, yerleşme zamanı ve tepe zamanı olmak üzere dört klasik performans ölçütü üzerinden gerçekleştirilmiştir. Elde edilen bulgular, önerilen yöntemin özellikle aşım kontrolü açısından belirgin bir üstünlük sağladığını ve genel anlamda daha stabil sistem yanıtı sunduğunu göstermektedir.

1. INTRODUCTION

The application of fractional calculus to control engineering has opened up a new field of study. Compared to traditional integer-order PID controller design, fractional-order controller design offers greater flexibility, which in turn enables the development of more robust and higher-performing control systems [1].

The integral windup problem is a common issue that emerges during implementation, particularly due to the nonlinear dynamics of physical systems [2]. Various anti-windup strategies have been proposed to mitigate this problem. However, most of these approaches do not take full advantage of the flexibility provided by fractional-order controllers.

One of the key advantages of fractional-order PID/PI controllers in the context of the windup problem is the ability to tune the integral order. Windup typically occurs in the presence of saturation and is directly related to the saturation error. The integral order is a critical parameter for ensuring that the system output reaches the desired reference value. From this perspective, the reference tracking error becomes the second important factor influencing controller behavior.

First-order plus dead time (FOPDT) systems are widely used mathematical models in controls. They are considered effective representations of dynamic system behavior [3]. Therefore, proposing a design method that is generalized for FOPDT systems can be highly beneficial in terms of broad applicability.

In this thesis, the physical constraints and controller parameters that influence the windup problem are thoroughly analyzed. Subsequently, the performance of existing anti-windup methods from the literature is examined in the context of FOPI-controlled FOPDT systems. The shortcomings of these methods are discussed, and a new approach is proposed that leverages the benefits offered by the fractional controller perspective.

Chapter 2 provides an overview of fractional calculus and fractional-order controllers. It discusses the historical development of fractional calculus in mathematics, presents

general definitions, and outlines common approximation methods. It also introduces the general structure and characteristics of fractional-order controllers.

Chapter 3 defines the integral windup problem and investigates two key factors that influence it: the saturation limit and controller parameters. Since this thesis focuses on FOPI controllers, three control parameters are considered: proportional gain, integral gain, and integral order.

Chapter 4 presents an extensive literature review of anti-windup methods. It explores how existing methods have been implemented in conjunction with FOPI controllers. Within this context, four prominent methods from the literature are examined: back calculation, automatic reset configuration, fractional filtered automatic reset configuration, and the fractional integral order reset methods.

Chapter 5 identifies the limitations of existing methods and leverages the advantages of the fractional-order perspective by incorporating the effect of the integral order. A novel approach is proposed in which the integral order is expressed as a function and tuned online. This method addresses the windup problem while also improving system performance by introducing a more adaptive control structure. The proposal is supported by a detailed analysis of how FOPDT system dynamics are affected by saturation and reference tracking.

Chapter 6 presents various FOPDT system scenarios along with corresponding FOPI controller configurations. The behavior of these systems under saturation is examined. Both the existing methods and the proposed strategy are evaluated independently. System responses and control signals are compared using unit step inputs, and the analysis is further supported with performance metrics such as rise time, settling time, peak time, and overshoot.

2. FRACTIONAL CALCULUS

2.1 Background

Fractional calculus has a long history and has been studied in both theoretical and applied mathematics. It is also used in engineering, especially in modeling and control problems.

The roots of fractional calculus go back to the work of Leibniz in 1695. However, systematic studies in this area started in the 19th century. In 1823, Abel tried to solve the well-known tautochrone problem using fractional derivatives. In 1832, Liouville used exponential series and extended the derivative operator to fractional orders. In 1847, Riemann defined the fractional-order integral and generalized fractional derivatives. In 1867, Grünwald and Letnikov introduced the discrete fractional calculus by combining two earlier approaches [5].

The use of fractional calculus in engineering became more common in the 20th century. It found applications in system theory, control theory, and signal processing. In control theory, Manabe used fractional calculus in frequency domain calculations in 1961 [4].

Oustaloup proposed a robust controller of non-integer order, which he called "Commande Robuste d'Ordre Non Entier". Later, Podlubny generalized classical PID controllers using fractional calculus and introduced the $PI^{\alpha}D^{\mu}$ controllers [6].

In recent years, fractional calculus has become increasingly popular in control engineering. Various methods have been developed for system modeling and controller design using fractional calculus. For instanse, Yumuk conducted a comprehensive study and proposed a fractional-order controller design method based on analytical computation techniques [7].

Fractional controllers are also implemented in different practical applications such as electric motor control [8], process control [9], aviation areas [10].

2.2 Definitions

Let the generalized fractional-order operator be given in (2.1).

$$ {}_rD_t^{\alpha} = \begin{cases} \frac{d^{\alpha}}{dt^{\alpha}} & Re(\alpha) > 0 \\ 1 & Re(\alpha) = 0 \\ \int_{\alpha}^{t}((d\tau)^{\alpha} & Re(\alpha) < 0 \end{cases} \tag{2.1} $$

There are three commonly accepted definitions for the generalized fractional-order operator. Two of them are the Grunwald-Letnikov and Riemann-Liouville. The third one, the Caputo method, focuses on solving differential equations using fractional calculus with a definition similar to that of Riemann-Liouville [1].

2.2.1 Grunwald-Letnikov definition

For $\alpha \epsilon R$, the definition of Grunwald-Letnikov is given in (2.2).

$$ {}_rD_t^{\alpha} g(t) = \lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{j=0}^{\left[\frac{t-r}{h}\right]} (-1)^j \binom{\alpha}{j} g(t - jh) \tag{2.2} $$

Where α is the order of the fractional derivative, r is the initial point that used as starting point of differentiation, h is the step size used in discrete approximation of fractinoal derivative.

2.2.2 Riemann-Liouville definition

For $\alpha \epsilon R$, the definition of Riemann-Liouville is given in (2.3).

$$ {}_rD_t^{\alpha} g(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_r^t \frac{g(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \, , \; n-1 < \alpha < n \tag{2.3} $$

Where r is the lower limit, τ is the integration variable and n is the smallest integer greater than α, used to express the derivative.

The definition of gamma function is given in (2.4).

$$ \Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt \tag{2.4} $$

Laplace transform of this definition is given in (2.5).

$$L[\,{}_0D_t^{\alpha} g(t)] = s^{\alpha} G(s) - \sum_{k=0}^{[\alpha]-1} s^{k}\ {}_0D_t^{\alpha-k-1} g(0) \tag{2.5}$$

Where $G(s)$ is the Laplace transform of $g(t)$ and ${}_0D_t^{\alpha-k-1} g(0)$ is the initial value of the function.

2.2.3 Caputo definition

For $\alpha \in R$, the definition of Caputo is given in (2.6.)

$${}_rD_t^{\alpha} g(t) = \frac{1}{\Gamma(n-\alpha)} \int_r^t \frac{d^n}{dt^n} \frac{g(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau\ ,\ n-1 < \alpha < n \tag{2.6}$$

Laplace transform of this definition is given in (2.7).

$$L[\,{}_0D_t^{\alpha} g(t)] = s^{\alpha} G(s) - \sum_{k=0}^{[\alpha]-1} s^{\alpha-k-1}\ {}_0D_t^{k} g(0) \tag{2.7}$$

As it is seen that the Caputo definition uses integer order derivative in order to transform to Laplace for initial conditions. That's why Caputo definition is more applicable than Riemann-Liouville definition [4].

2.3 Oustaloup Approximation Method

To realize fractional transfer functions, various approximation methods are needed. There are several approximation methods. Among them, the most commonly used is Oustaloup method.

It generates a integer-order transfer function that shows similar behavior to the fractional one in the frequency domain. The Oustaloup approximation method is given in (2.8).

$$s^{\alpha} = K \prod_{i=1}^{N} \frac{s+\widetilde{\omega}_i}{s+\omega_i} \tag{2.8}$$

Where K is DC gain of serial filter bank, ω_i and $\widetilde{\omega}_i$ are pole and zero frequencies.

Finally, (2.9) shows the mathematical formulations of K, ω_i and $\widetilde{\omega}_i$.

$$K = \omega_h^{\alpha} \quad , \omega_i = \omega_l \omega_u^{\frac{2i-1+\alpha}{N}} \quad , \widetilde{\omega}_i = \omega_l \omega_u^{\frac{2i-1-\alpha}{N}} \tag{2.9}$$

Where ω_l and ω_u are upper and lower bounds of the frequencies [4].

2.4 Fractional Order Controllers

The general formulation for classical PID controllers is given in (2.10).

$$C(s) = K_p + \frac{K_i}{s} + K_d s \tag{2.10}$$

The generalized form of this expression using fractional-order calculation is given in (2.11).

$$C(s) = K_p + \frac{K_i}{s^\alpha} + K_d s^\mu \tag{2.11}$$

Where K_p, K_i and K_d are proportional, integral and derivative gains of the controller, seperately. Also s is the derivative and 1/s is the integral terms in Laplace domain.

Figure 2.1. shows the general form of the fractional order PID controllers. In general, α and μ parameters are chosen in the interval $0 \leq \alpha, \mu \leq 2$.

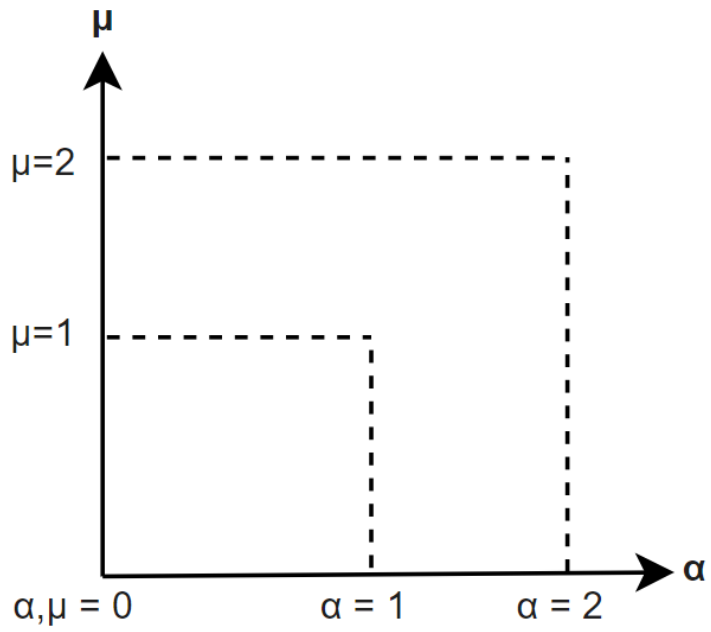

Figure 2.1 : The range of fractional parameters.

This form of the FOPID controllers is also contains classical PID controller types.

If both α and μ are set to 1; then the controller becomes the classical PID controller. If $\alpha = 1$ and $\mu = 0$; then the controller is named PI controller. Similarly, the

condition of $\alpha = 0$ and $\mu = 1$ is PD controller. Besides, $\alpha = 0$ and $\mu = 0$ means classical proportional controller.

When the controller is in fractional form, the definitions can be made in a similar way.

If $\alpha = 0$ and μ is in fractional form; then the controller is named as fractional order PD (FOPD) controller.

If $\mu = 0$ and α is in fractional form; then the controller is named as fractional order PI (FOPI) controller.

Finnaly if both α and μ are in fractional form, the controller takes fractional order PID (FOPID) controller shape. Figure 2.2. illustrates the block diagram of the FOPI controller. Throughout the studies conducted in this thesis, FOPI controllers are used.

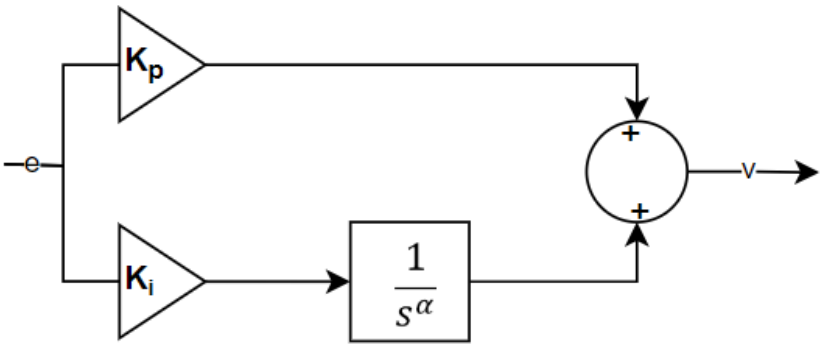

Figure 2.2 : Block diagram of FOPI controller.

3. INTEGRAL WINDUP

In this section, the integral windup problem will be explained. First, a problem definition of the windup problem will be given. Next, the effect of the saturation limit, which is one of the key factors influencing windup, will be analyzed. Finally, the impact of controller parameters on windup will be examined.

3.1 Problem Definition

In control systems, linear time-invariant (LTI) models are commonly used for modeling and controller design. However, when these models are applied to real physical systems, certain nonlinear behaviors often appear. These nonlinearities usually arise from physical constraints of the system.

For example, consider the control of an electric motor. Physical limitations include the maximum achievable speed, the nominal input voltage required for operation, and the maximum current the motor can draw. Similarly, in a process control application, constraints may involve pressure, mass flow rate, or the maximum flow rate that a valve can allow [11].

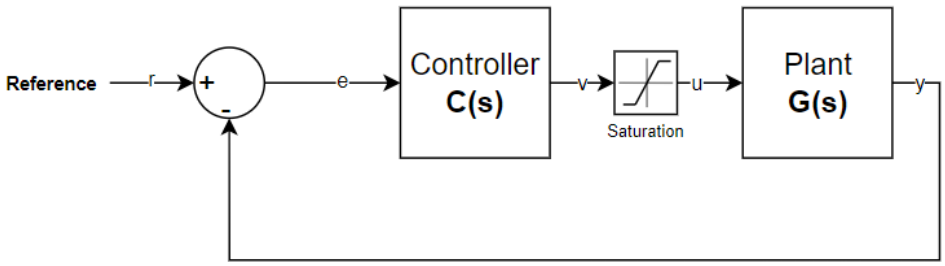

Figure 3.1 : Block diagram of closed loop system with saturation.

Figure 3.1. shows the block diagram of closed loop system with saturation. When the control signal exceeds the actuator limits, a difference occurs between the "u" signal and the "v" signal. During the time this difference exists, the system does not receive the control signal proposed by the controller. Instead, the system receives the maximum signal allowed by the saturation limit. This situation indicates that the

system is no longer operating within the feedback structure. The difference between the system response and the reference signal becomes irrelevant. In other words, the control system now operates in open-loop instead closed-loop.

Meanwhile, the controller continues to calculate the tracking error. The integral term in the controller keeps accumulating this growing error over time and adds it to the "v" signal. As a result, the "v" signal increases significantly. This leads to a situation where the controller can no longer drive the system as intended.

As a benchmark problem, let us consider an example FOPDT system. The parameters of this system are K=1, T=1, and L=0.5.

$$G(s) = \frac{1}{s+1} e^{-0.5s} \tag{3.1}$$

The transfer function of the example system is given in (3.1). For this system, the saturation limit is set to 1.2.

A FOPI controller is defined to control this system. The controller is selected to ensure that the system settles to the reference, with a response that includes some overshoot.

$$C(s) = 1 + \frac{1}{s} \tag{3.2}$$

Equation (3.2) is the transfer function of the controller. α value is set to 1 first, thus the controller became a classical PI controller.

The system responses under unit step reference input are analyzed for both saturated and non-saturated conditions. Figure 3.2 shows the unit step responses of the system under both with saturation and without saturation conditions. It can be observed that saturation causes three major changes in the system response. First, the system under saturation exhibits a higher overshoot. Second and third, both the rise time and the settling time increase. This behavior reflects the typical effects of saturation.

In Figure 3.3, the control signals before and after the saturation block are compared for the system with saturation. Between approximately 0.2 seconds and 2.2 seconds, the controller produces a control signal larger than the system input. During this time, the system is driven not by the generated control signal v, but by the saturated signal umax. This is the situation where the integral windup problem occurs.

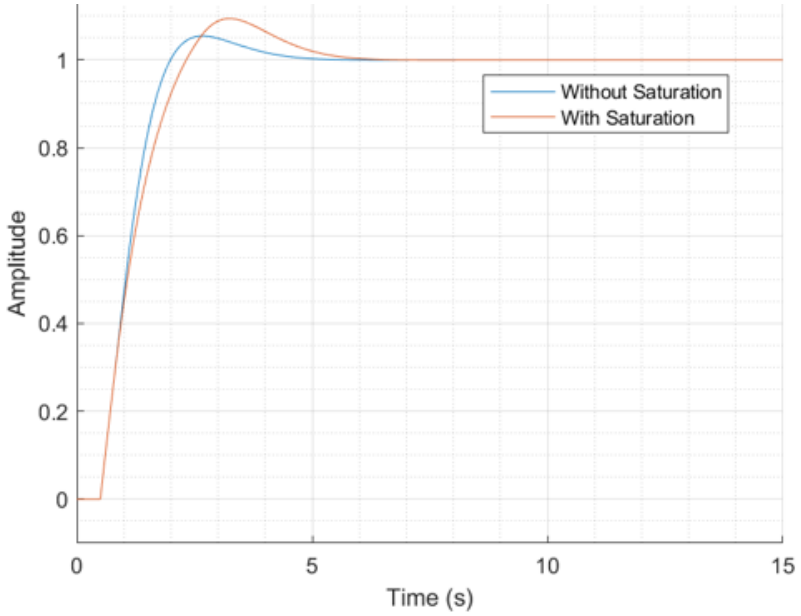

Figure 3.2 : The comparison of step responses with and without saturation effect.

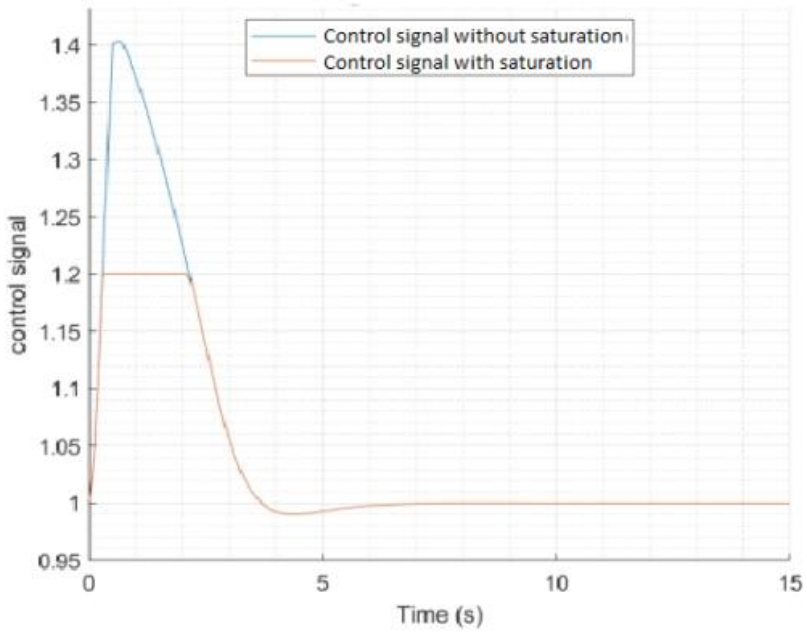

Figure 3.3 : The comparison of control signals with and without saturation effect.

3.2 Effect of Saturation Limits

The previous section explained how the presence of saturation in a control system makes it harder to control. In this section, the effect of different saturation limits on the system response is analyzed, while keeping all other conditions the same.

If the system does not saturate, the maximum control signal generated will be around 1.5. Figure 3.4 shows the unit step responses for different saturation limits. As the

saturation limit increases, the constraint on the control input is reduced. This allows the controller to produce results closer to the desired response.

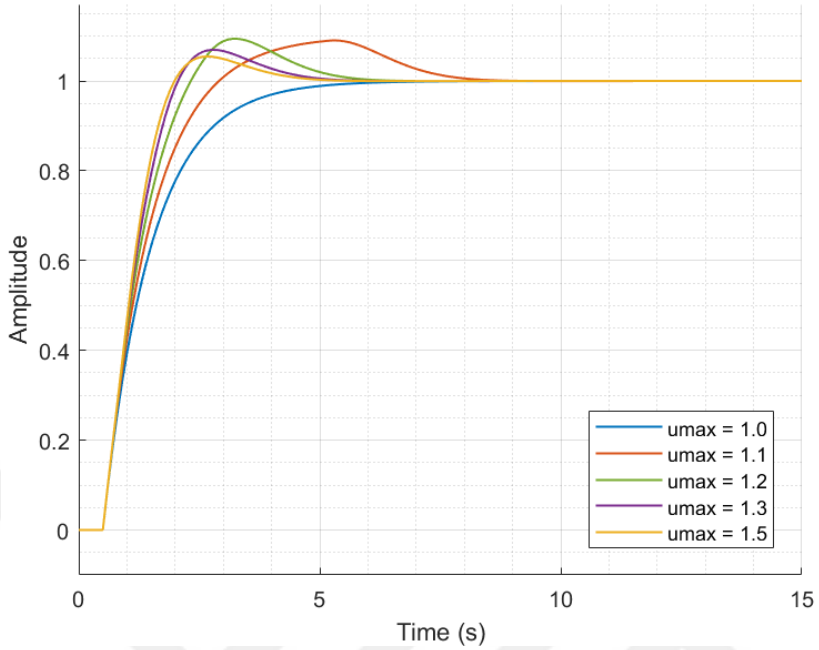

Figure 3.4 : The effect of saturation limit on unit step response.

When umax limit becomes high enuogh, the system no longer experiences saturation, and the control signal generated by the controller (v) is directly applied to the system. As the saturation limit value decreases, it is observed that the overshoot initially increases and then decreases. On the other hand, the rise time decreases as the saturation limit increases, indicating an acceleration in the system. The settling time, except for the limit value of 1.1, remains at similar levels.

Figure 3.5 shows how the control signal changes for different umax limits. From this graph, it is observed that when umax is greater than a certain value (1.1 for this system), the control signal can reach a steady value within a certain time. However, as the umax value decreases, the control signal reaches saturation earlier. Although the controller attempts to apply more correction, the saturation limit prevents it. As this period extends, the control signal before the saturation block (v) continues to increase linearly.

As a result, the lower the saturation limit, the less capable the controller becomes in controllability for the system. The system response slows down, and below a certain value, the system starts to show steady-state error. Increasing the umax value leads to a faster system response, but it also increases the risk of overshoot. These results

highlight the importance of considering the saturation limit as a physical constraint while designing a controller.

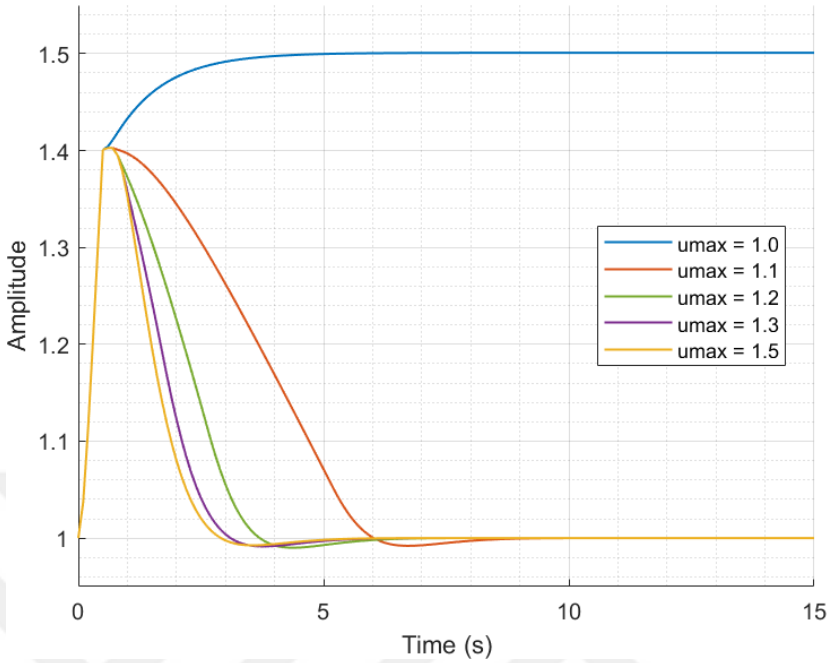

Figure 3.5 : The effect of saturation limit on control signals.

3.3 Effect of Controller Parameters

In this section, the effect of controller parameters on the windup problem will be investigated when a system is controlled using fractional-order PI controllers. In FOPI controllers, there are three parameters: Kp, Ki and α.

3.3.1 Effect of proportional gain

In this analysis, other parameters are kept constant as in the initial condition:

$u_{max} = 1.2, K_i = 1, \alpha = 1.$

Figure 3.6 shows the unit step responses of the system for different Kp values. At relatively low Kp values (as seen in the case of 0.7), rise time and settling time increase dramatically, and the system exhibits the highest overshoot. In general, as the Kp value increases, the overshoot tends to decrease up to a certain point. However, the closed-loop system gradually approaches an underdamped behavior.

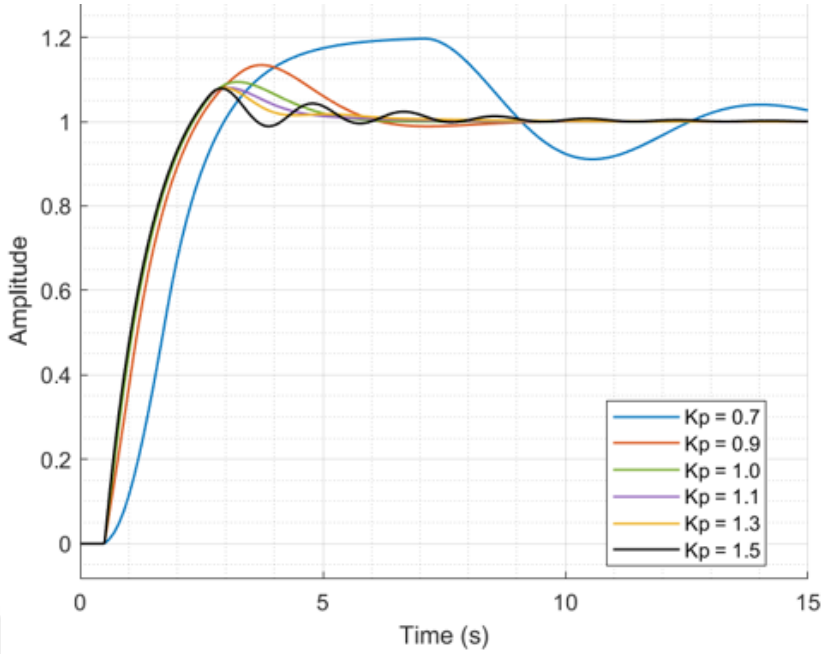

Figure 3.6 : The effect of K_p on unit step response.

It should be noted that in this analysis, umax value is fixed at 1.2. Figure 3.7 shows the amplitude of the control signal observed before the saturation block, as produced by the controller. As Kp increases, the amplitude of the control signal also increases. In contrast, lower Kp values result in smaller control signals generated by the controller.

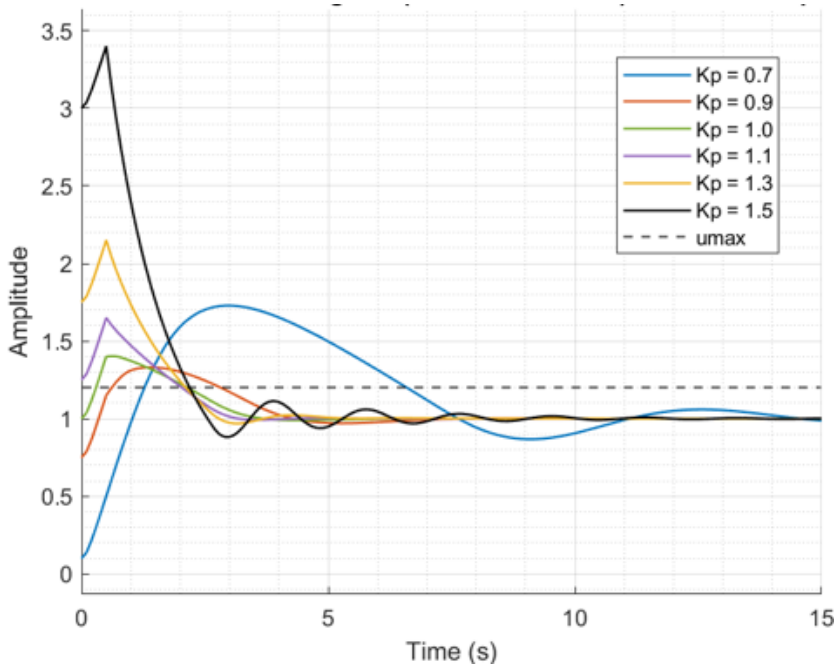

Figure 3.7 : The effect of $K_p$on control signals without saturation.

In general, the effect of Kp is directly related to the aggressiveness of the system. Very high Kp values lead to large control signals, which increases the risk of hitting actuator limits. On the other hand, low Kp values are indirectly helpful for avoiding saturation. However, in this case, the controller performance is negatively affected. Finally, this analysis shows that the effect of windup should also be considered when tuning the proportional gain during controller design.

3.3.2 Effect of integral gain

In this analysis, other parameters are kept constant as in the initial condition: $u_{max} = 1.2$, $K_p = 1$, $\alpha = 1$. Figure 3.8 shows the effect of increasing integral gain values on the unit step response. As Ki increases, the overshoot increases. This is because a high integral gain tries to reduce the error more aggressively. At low integral gain values, the system responds with less overshoot.

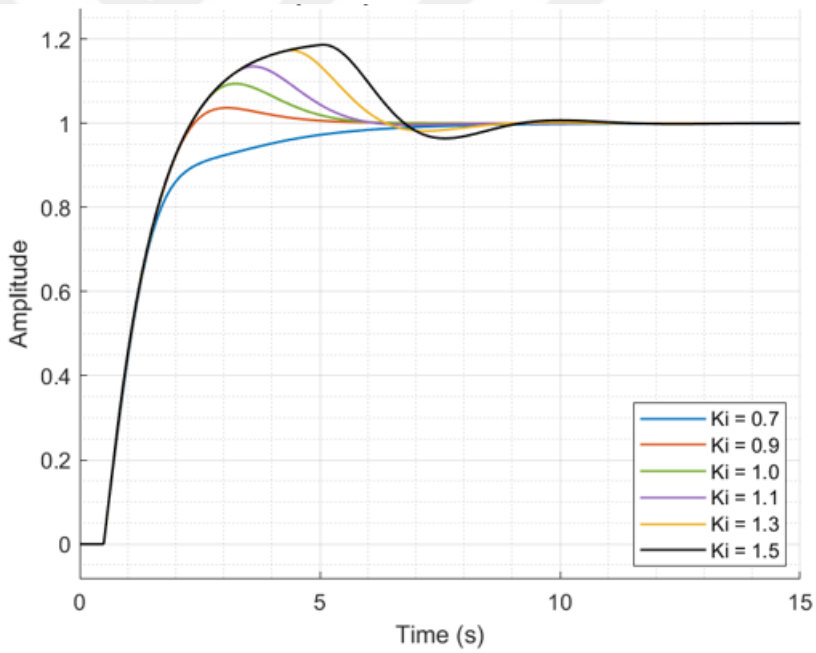

Figure 3.8 : The effect of K_i on unit step response.

Figure 3.9 shows how the control signal generated by the controller changes for different integral gain values, under the condition where the saturation limit is fixed at 1.2. As Ki increases, the first peak of the control signal rises significantly. When Ki is too high, the control signal increases rapidly, resulting in an oscillatory transient response. This becomes a factor that negatively affects the stability of the system. If

the integral gain is reduced sufficiently, the system may not enter saturation. However, this may also cause the system response to deviate from the desired behavior.

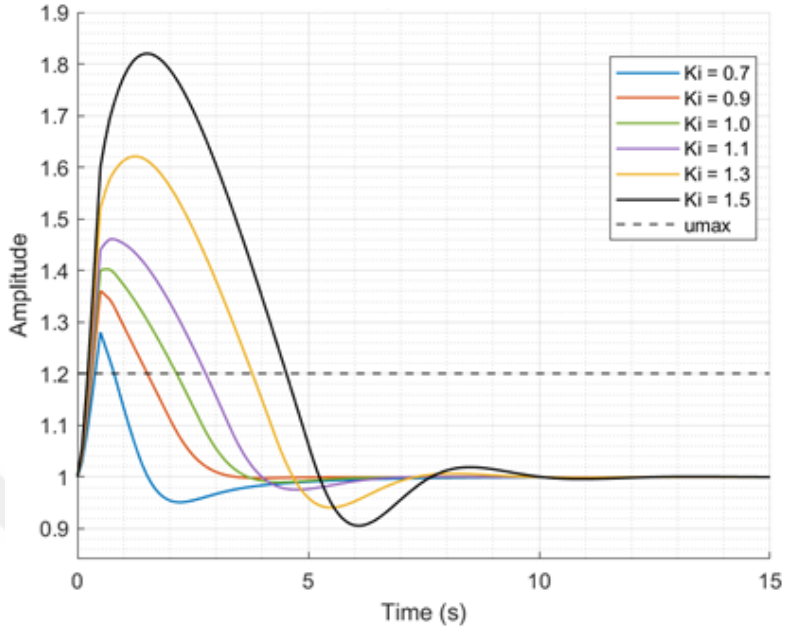

Figure 3.9 : The effect of K_i on control signals without saturation.

Integral gain represents the system's integrated response to error. Therefore, increasing the integral gain makes the response more aggressive. An aggressive response means a higher control signal. In a system with saturation, this becomes the primary cause of the windup problem. The integral term significantly increases the control signal produced by the controller. This directly leads to overshoot issues. Overshoot is one of the main pointers of integral windup problem.

3.3.3 Effect of integral order

In this section, the effect of only the variation in the fractional integral order is investigated when using a fractional-order PI controller structure. Specifically, the behavior of integral windup under saturation is examined. For this analysis, other parameters are kept constant as in the initial condition:

$u_{max} = 1.2, K_p = 1, K_i = 1.$

Figure 3.10 illustrates how the system response changes under a unit step input for different values of the integral order. For lower values of α, the system exhibits a higher overshoot, while increasing α leads to a reduction in overshoot. However, as α

increases, the system slows down, and the time required to reach steady-state becomes longer.

It is also observed that both very low and very high values of α increase oscillations. In particular, for α=1.3, the system's response to saturation is found to be very aggressive.

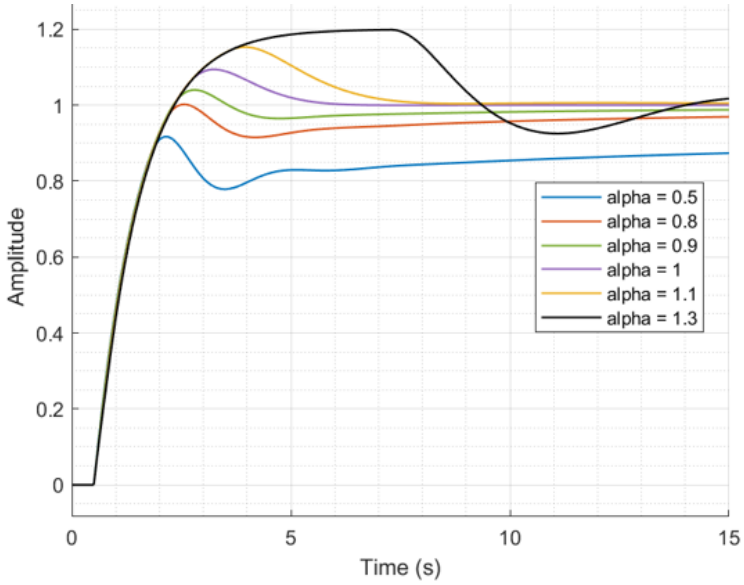

Figure 3.10 : The effect of α on unit step response.

Figure 3.11 compares the control signals of the systems for different α values. As the value of α decreases, the initial peak of the control signal increases.

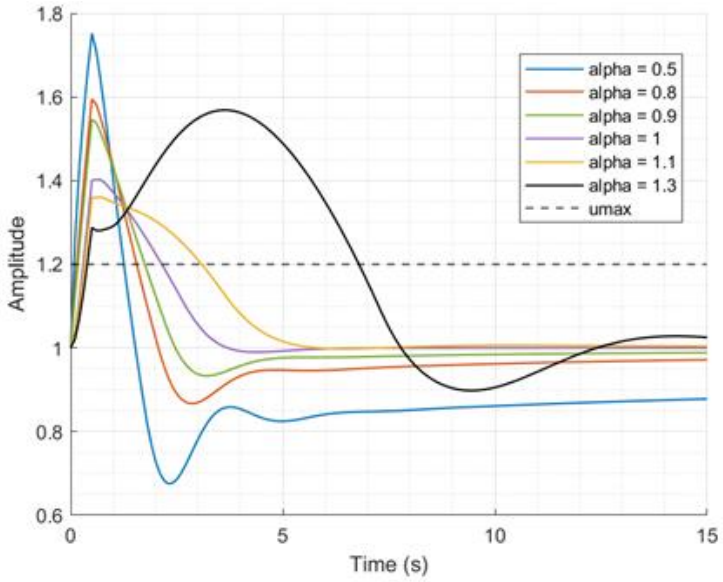

Figure 3.11 : The effect of α on control signals without saturation.

However, the signal quickly settles and drops below the saturation limit of 1.2. As α increases, the peak occurs later. As a result of this behavior, the control signal remains above the saturation limit for a longer duration. This situation significantly increases the risk of integral windup, as the magnitude of the integrator signal continues to accumulate error during saturation.

These results demonstrate that the integral order has a significant impact on the system behavior under saturation. Therefore, during controller design, taking the fractional integral order into account may also play an important role in preventing or mitigating integral windup.

3.3.4 Discussions

The windup problem arises due to a mismatch between the control signal generated by the controller and the maximum or minimum input that the system can physically accept. Fundamentally, this mismatch can result from two main reasons:

- a low saturation limit, and
- a large control signal generated by the controller.

Any parameter that increases the magnitude of the control signal contributes to the risk of windup. However, the problem becomes particularly critical when the error is accumulated through integration, which amplifies the signal over time. In classical PID/PI controllers, this accumulation is governed only by the integral gain. In fractional-order PI/PID (FOPI/FOPID) controllers, however, an additional influential factor is the fractional integral order itself.

4. ANTI-WINDUP METHODS FOR FOPI CONTROLLERS

In Chapter 3, the problem of integral windup was thoroughly examined. The causes of this issue and the contributing factors were analyzed in detail. This chapter focuses on the methods that have been developed to address the integral windup problem.

The significance of the windup issue mainly arises from the integral action of the controller. Methods proposed in the literature have aimed to limit or suppress this integral contribution in order to mitigate the impact of saturation on control performance. Some of these methods were initially designed for classical PID controllers and were later adapted or applied to fractional-order controllers. Others were specifically developed for fractional-order control systems.

Four fundamental methods that are commonly used in the literature for FOPI controllers will be presented: back calculation, automatic reset configuration (ARC), ARC with a fractional filter and fractional integral order reset methods.

4.1 Back Calculation

One of the most commonly used methods developed to solve the wind-up problem is the back calculation method. This method is based on the principle of recalculating the integrator term of the controller when saturation occurs.

The back calculation method was first introduced by Fertik and Ross in 1967 as a technique to prevent the integrator term of the controller from growing excessively due to saturation [12]. Later studies by Schaufelberger and Glattfelder in 1983 [13], as well as by Åström and Wittenmark in 1987 [14], contributed to the widespread adoption of this method.

Back calculation is a widely used anti-windup technique in both classical PID and fractional-order PID controllers. It is applied in various areas, including electric motor drive systems [18] and industrial process control applications [19].

The use of the back calculation method in classical PID/PI controllers is illustrated in Figure 4.1. In this diagram, the actuator block represents saturation. Kp is the proportional gain of the controller, while Kp/Ti represents the integral contribution. Tt is the term where the anti-windup action, known as the ‘tracking time constant’.

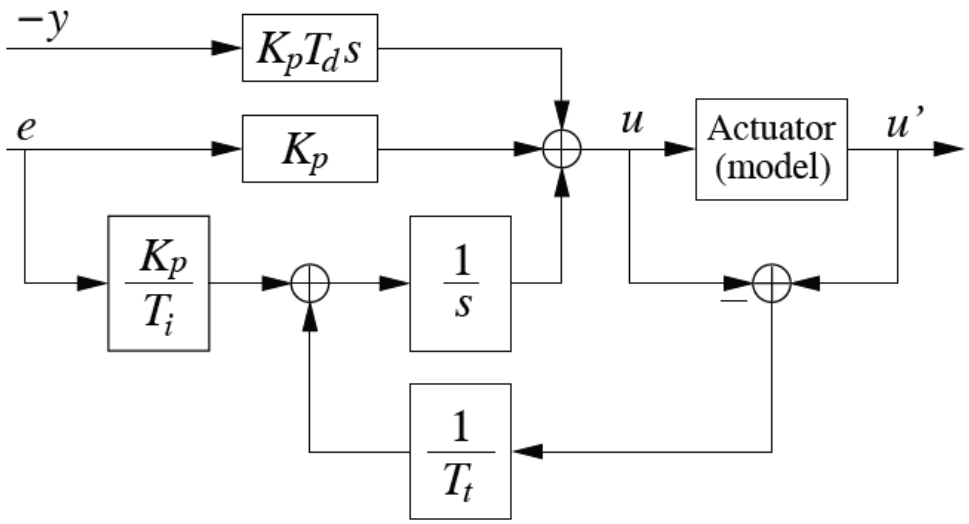

Figure 4.1 : General scheme of back calculation anti wind-up method [14].

There are several methods in the literature for calculating the tracking time constant based on studies conducted on this topic.

$$T_t = \sqrt{T_i \cdot T_d} \tag{4.1}$$

In the study conducted by Åström and Hägglund in 1995, it was recommended that the value of the tracking time constant be selected as expressed in (4.1) when using a PID controller [15]. However, this formulation does not provide any recommendation for the case of a PI controller.

$$T_t = T_i \tag{4.2}$$

Equation (4.2) presents a formulation proposed by Bohn and Atherton, which defines how the tracking time constant should be selected when a PI controller is used [16].

$$T_t = K_p \tag{4.3}$$

Finally, (4.3) describes the method proposed by Hanus in his 1987 study for determining the tracking time constant [17].

In this thesis, anti-windup mechanisms are implemented together with fractional-order PI controllers. In line with the overall structure and notation of the study, the block diagram of the back calculation method has been redesigned specifically for FOPI controllers.

Figure 4.2 presents the block diagram illustrating the use of the back calculation anti-windup method in FOPI controllers. The signals shown in the diagram are listed as follows:

- e: error signal
- Kp: proportional gain
- Ki: integral gain
- $\frac{1}{s^{-\alpha}}$: fractional order integrator
- v: sum of integral and proportional controller parts
- u: control signal to be applied to the system after saturation block
- e_{sat}: saturation error
- K_{aw}: back calculation parameter (1/tracking time constant)

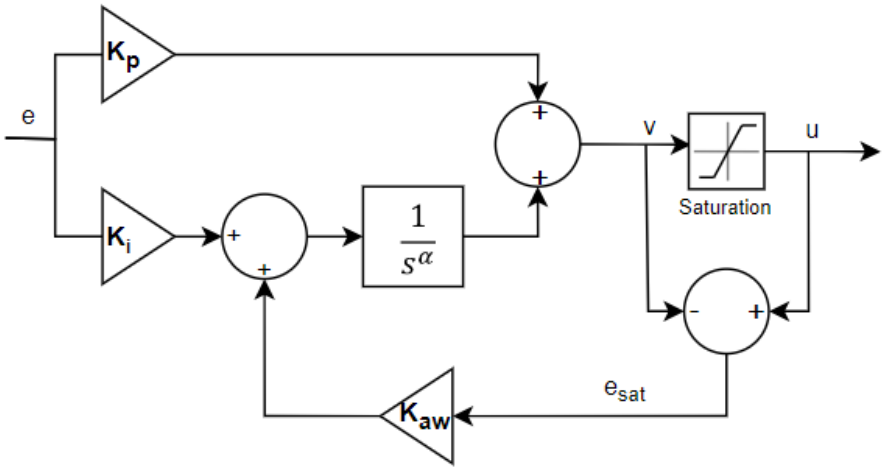

Figure 4.2 : Block diagram of back calculation method.

In this study, the value of Kaw is given as a modified form of (4.2).

$$K_{aw} = \frac{K_i}{K_p} \tag{4.4}$$

Equation (4.4) provides the formulation used to determine the K_{aw} value throughout the study when applying the back calculation method.

4.2 Automatic Reset Configuration

One other method developed to address the integral windup problem is the Automatic Reset Configuration (ARC). The ARC method aims to regulate the accumulation of the integral term by explicitly incorporating the effect of saturation into the control structure. It was initially proposed by Åström and Hägglund in the context of classical PID control, as described in their work 'Advanced PID Control' [20].

Several studies in the literature have investigated the ARC method. Wang, Gambier, and Vinagre [21] applied this method in a wind turbine pitch control system using a FOPID controller and successfully mitigated the adverse effects of saturation on system performance. Murad, Tzounas, and Milano [22] demonstrated the positive impact of integrating saturation limiters into FOPI controllers in power systems. Additionally, Erik Torstensson referred to this method as the 'Foxboro method' in his master's thesis submitted to Lund University [23].

The key feature of the ARC method is the inclusion of a model saturation block within the controller structure. The proportional and integral components of the control signal are independently passed through this 'model saturation' block. This block simulates the actuator's physical limits and acts as if the system is already in saturation. In this way, it models the effect of saturation in advance. The signal output from this block is used both as an input to the integral block and as the control signal to apply to saturation block.

Figure 4.3. illustrates the implementation of the ARC method used in the context of this thesis. The signals represents followings:

- e: error signal
- Kp: proportional gain
- $\left(\frac{K_i}{K_i + s^{\alpha}}\right)$: integral part of the controller as a transfer function
- v : sum of integral and proportional controller parts
- u: control signal to be applied to the system

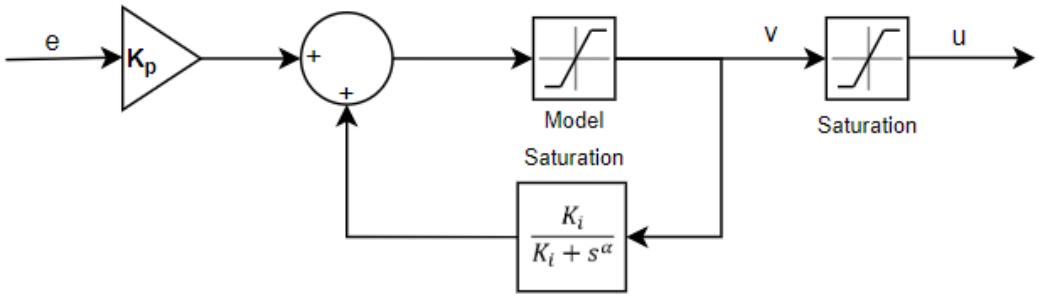

Figure 4.3 : Block diagram of automatic reset configuration method.

In the combination of FOPI controller and ARC anti windup method structure, the integral term is represented not by the transfer function of $\left(\frac{K_i}{K_i + s^{\alpha}}\right)$.

The fractional-order structure of the controller allows for greater flexibility in adapting the integrated error signal to system dynamics. In the ARC method, the signal entering the integral block is not the error signal alone but also includes the effect of the modeled saturation. This ensures that the integral action slows down or is limited as the system approaches the saturation region. In this way, this method aims to prevent the occurrence of windup.

This configuration limits how the integral term grows when saturation occurs. Even if the error increases, the integral action is reduced as the control signal nears the actuator limits. This helps the controller stay within the system's physical boundaries and prevents unrealistic control signals from harming the plant.

4.3 Automatic Reset Calculation with Fractional Filter

Another method proposed to address the wind-up problem can be considered as a customized version of the automatic reset configuration (ARC) method for fractional-order controllers. In this approach, all the standard steps of the ARC method remain valid. In addition, a fractional filter block is placed in front of the controller. This approach for automatic reset configuration method was proposed and demonstrated in the study conducted by Meena and colleagues in 2024 [24].

Figure 4.4. shows the block diagram of the ARC method modified with a fractional filter. The signals and blocks can be defined in a similar manner to those in the classical ARC method.

- e: error signal
- Kp: proportional gain
- $\left(\frac{K_i}{K_i + s^{\alpha}}\right)$: integral part of the controller as a transfer function
- v : sum of integral and proportional controller parts
- u: control signal to be applied to the system after saturation block

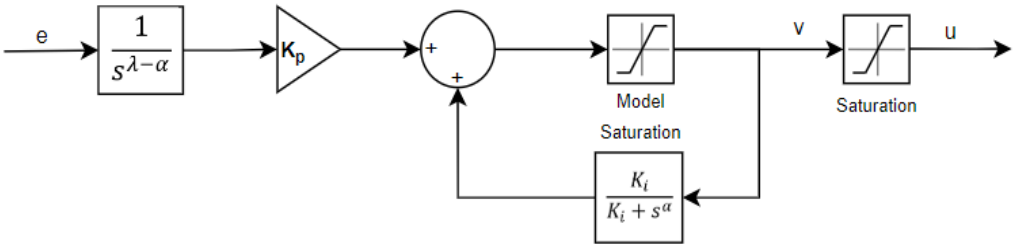

Figure 4.4 : Block diagram of ARC with fractional filter.

The fractional filter added before the controller is defined as given in (4.5). In this equation, α represents the integral order of the FOPI controller. λ is a parameter determined in relation to the target crossover frequency based on the proposed design criteria.

$$Fractional\ filter: \frac{1}{s^{\lambda-\alpha}} \tag{4.5}$$

λ is defined during the controller design process based on Bode's ideal transfer function. Equation (4.6) presents the target loop transfer function. In this equation, L represents the time delay of the original process, λ corresponds to the slope of the magnitude curve, and ω_c is defined as the target crossover frequency.

$$T(s) = \left(\frac{\omega_c}{s}\right)^{\lambda} e^{-Ls} \tag{4.6}$$

The value of λ is directly related to the controller parameters when designing a fractional-order PI controller.

$$\lambda = \log_{\omega_c} \frac{K_p \cdot K}{T} \tag{4.7}$$

Equations (4.7) and (4.8) reveal the relationship between the λ parameter and both the controller and system parameters. Here, K_p is the proportional gain of the controller, while K_i is the integral gain. Additionally, K represents the gain of the first-order plus time delay (FOPTD) system.

$$\lambda = \log_{\omega_c} K_i \cdot K \tag{4.8}$$

4.4 Fractional Integrator Order Reset

Another method proposed to address the integral wind-up problem has been developed specifically for fractional-order controllers. This approach aims to reduce the effect of integration by setting the integrator order to zero in the presence of saturation [25].

Figure 4.5. presents the block diagram of the method that aims to mitigate the wind-up problem by resetting the integral order. The method can be explained as follows:

- e: error signal
- Kp: proportional gain
- Ki: integral gain
- $\frac{1}{s^{-\alpha}}$: fractional order integrator
- v: sum of integral and proportional controller parts
- u: control signal to be applied to the system
- e_{sat}: saturation error

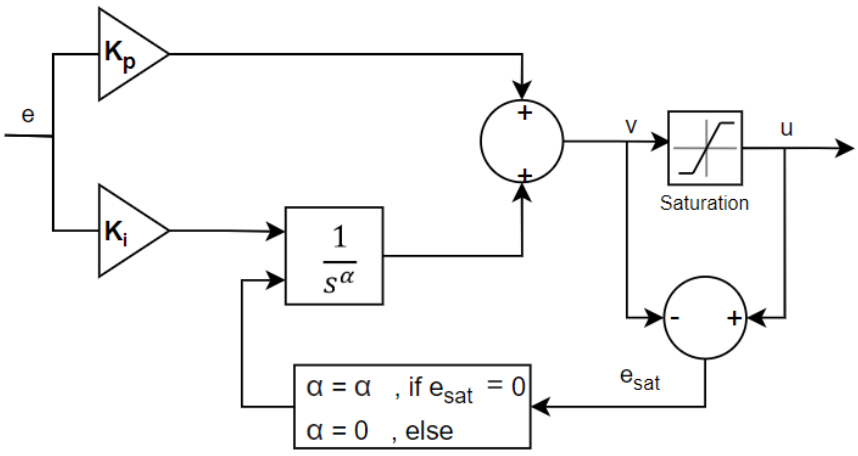

Figure 4.5 : Block diagram of fractional integral order reset method.

The difference between the control signals before and after saturation is referred to as the saturation error. If the saturation error is zero, it indicates that the system has not reached saturation, and the integral order remains at its designed value to influence the system.

However, if the saturation error takes a nonzero value, it means the system has entered saturation. In this case, the integral order is reset to zero. As a result, the integral action of the controller is effectively removed, and the controller begins to behave like a proportional controller.

5. PROPOSED ANTI-WINDUP STRATEGY

This section will present the details of the proposed anti-windup strategy. First, existing algorithms in the literature will be reviewed to explain the motivation behind the proposed approach. Then, the focus will shift to FOPDT systems, which are central to the method. The mathematical background of the method will be provided, and the selection of its parameters will be discussed. Finally, the complete mathematical formulation of the proposed strategy will be presented.

5.1 Motivation

Various methods have been proposed in the literature to address the windup problem. Some of these are based on the classical PID control approach, while others are designed or adapted specifically for fractional-order controllers.

Fundamentally, all methods focus on mitigating the effect of the integral action under saturation to overcome the windup issue.

The first method, back calculation, adopts a comprehensive perspective. Instead of completely disabling the integral action, it attenuates it. This helps prevent steady-state errors that may occur under sustained saturation. This method was originally developed for PID controllers and is one of the earlier approaches. The second method, Automatic Reset Configuration (ARC), integrates the saturation behavior directly into the controller structure. And the integral term is feeding back into the controller. Both of these methods were originally developed for PID controllers. These approaches do not consider the flexibility offered by fractional-order controllers. In fractional-order control, the integral term is defined not only by the integral gain but also by the integral order, which should be treated as an additional design parameter. The approach that takes fractional parameters into account involves placing a fractional filter in front of the ARC method.

The final method sets the integral order to zero in the presence of saturation and restores it to its designed value when saturation is not present. Simulations show that,

under many conditions, this final approach can outperform the others in terms of performance.

The motivation of this study stems from the following question: if the controller is a fractional-order PI controller, and we have the flexibility to adjust the integral order, what advantages can be gained by updating this value online?

As will be discussed in detail later, many systems can be modeled as First-Order Plus Dead Time (FOPDT) systems. Therefore, the proposed approach could be parameterized using the parameters of FOPDT models. In addition to the presence and severity of saturation, the deviation of the system response from the reference is also a critical factor. Incorporating both of these criteria into the online update of the controller can significantly enhance performance. Accordingly, an anti-windup approach is proposed that updates the fractional integral order online based on both the reference tracking error and the saturation error.

5.2 First Order Plus Dead Time Systems

In this section, the First-Order Plus Dead Time (FOPDT) system model will be introduced. The reason we focus on this system model is that; within the simulations, the anti-windup methods discussed are implemented on FOPDT systems. Also and more importantly, proposed strategy is generalized for FOPDT systems.

The main reason for choosing FOPDT systems is that FOPDT models are commonly encountered in control system modeling. One of the main reasons for preferring FOPDT structures in system modeling is that the behavior of many real-world systems follows this form. For example, many industrial processes, such as chemical, thermal, and mechanical systems, exhibit first-order plus time delay dynamics. FOPDT models are successful in representing the dynamic responses of such systems [3].

Second reason is the modeling and parameter estimation advantages offered by FOPDT systems. FOPDT systems are defined by three parameters: gain, time constant, and time delay. For instance, for modeling black-box systems, these parameters can often be easily obtained from even a unit step response. The explicit inclusion of time delay within the model significantly contributes to accurately representing system behavior. The presence of this parameter is particularly important in both system identification and control design [26].

Furthermore, classical control approaches have a rich body of literature and practical applications specifically developed for FOPDT models. For example, widely known PID tuning methods such as Ziegler-Nichols, Cohen-Coon, and Internal Model Control (IMC) are designed for controlling FOPDT systems [27].

Another advantage of the FOPDT model is its ability to approximate high-order systems. Complex and high-order systems can often be represented using a simplified FOPDT model [29].

The general form of an FOPDT system is given in (5.1).

$$G(s) = \frac{K}{Ts + 1} e^{-Ls} \tag{5.1}$$

Where, K represents the system gain, T the time constant, and L the time delay.

The system's gain (K) is the coefficient that indicates the amount of change in the output signal resulting from a unit change in the input signal. It can be considered a measure of the system's sensitivity. A higher gain leads to a larger output response.

The time constant (T) represents the time it takes for the system output to reach approximately 63.2% of its final steady-state value. It indicates how quickly the system responds to input changes. A small T means a fast response, while a large T implies a slower response.

The time delay (L) indicates how long it takes for a change in the input to begin affecting the output. A large time delay negatively impacts the system's stability and controllability [3].

5.3 Mathematical Background of Strategy

In this section, the proposed strategy is presented. The method adresses the windup problem that occures when controlling FOPDT systems using FOPI controllers.

In fractional-order control systems, especially in the FOPI controller structure, the parameter α, which defines the order of the integrator, plays a critical role in shaping the system's dynamic response. Compared to classical PI controllers, fractional-order structures provide more flexibility in controller design and offer the potential to improve system performance. However, achieving optimal performance across the

entire operating range with a fixed α parameter is sometimes not possible especially considering the windup problem.

Therefore, this study proposes an approach in which the α parameter is updated online and adaptively based on the system's error dynamics and saturation condition.

Figure 5.1. shows the block diagram designed for the proposed controller.

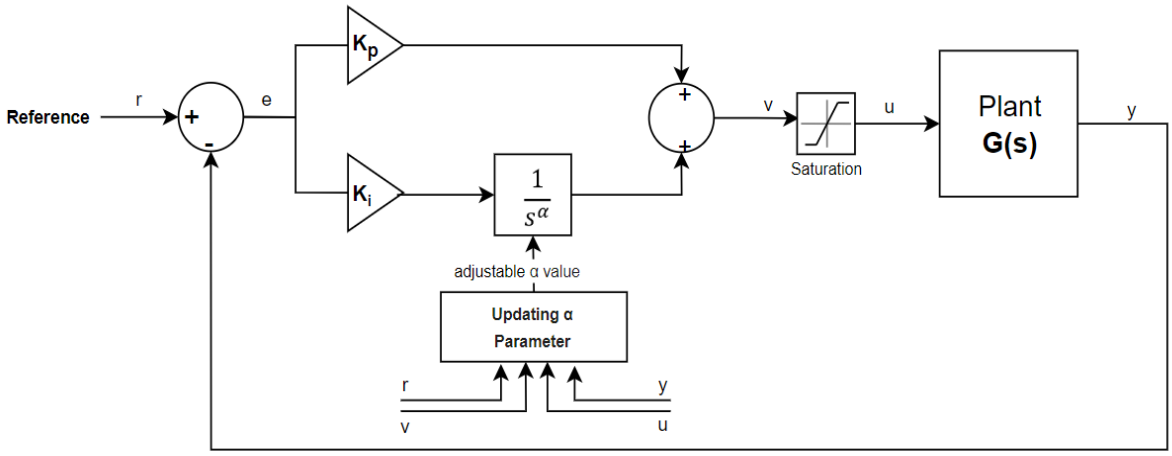

Figure 5.1 : Block diagram of the proposed strategy.

- The reference input (r) is the desired target value that the system aims to reach.
- The system output (y) is the actual output signal produced by the system G(s).
- The error signal (e) is the main input to the controller, indicating the deviation of the output from the reference.
- The pre-saturation control signal (v) is the sum of the proportional and integral terms of the controller. It is not directly applied to the system but is instead fed into the saturation model.
- The post-saturation control signal (u) is the output of the saturation block and input signal that is applied to the system G(s).
- The saturation error (e_{sat}) is the difference between the applied control signal and the calculated (pre-saturation) control signal. If this value is zero, it means the system is operating without saturation.

In the proposed anti wind-up scheme, the α term of the FOPI controller, expressed in is not kept constant as defined during the controller design prosedure. Instead, α is

continuously adjusted based on both the system's tracking error and the saturation error.

The error defined in (5.2). represents the deviation of the systems output signal from the reference signal. Where e is the reference tracking error, y is system output and r is the reference signal.

$$e = r - y \tag{5.2}$$

For the formulation, the reference tracking error value is normalized using the formula given in (5.3).

$$e_{norm} = \frac{|e|}{|r|} \tag{5.3}$$

Tracking error and saturation error are defined as e and e_{sat}, respectively. The normalization of these values is fundamentally required to ensure the generalizability of the proposed strategy, based on the following two needs: First, normalization eliminates scale dependency. Reference signals and saturation limits may vary in amplitude. Depending on the system and the controller, the system output signal (e.g., in a system exhibiting large overshoot) and the control signal generated by the controller (e.g., in a system that remains in saturation for a prolonged period) can produce different magnitudes. Normalizing with respect to the reference signal (r) and the control signal (r) provides a basis for generalization. Second, the error signals are rendered unsigned. For the tracking error, what matters is not whether the system output is above or below the reference, but rather the magnitude of the deviation from the reference. For the saturation error, the update rule should treat v > umax and v < umin equivalently, making the direction of the violation irrelevant. Finally, with normalization it is provided a scale-independent performance metric, allowing for generalization across reference signals of varying magnitudes.

Similar to the reference tracking error, e_{sat}, which is the difference between the theoretical control signal and the output of the saturation block, is also an indicator that the system is operating under physical constraints. Equation (5.4) shows how the saturation error value is obtained.

$$e_{sat} = v - u \tag{5.4}$$

Similar to e signal, the saturation error is also normalized. Equation (5.5) provides the expression used to normalize the saturation error.

$$e_{sat_{norm}} = \frac{|e_{sat}|}{|u|} \tag{5.5}$$

At the core of the function lies the α update rule, which has a two-term structure defined based on the tracking error and saturation error.

The general form of the α update rule is given in (5.6). Here, the coefficient η_1 is used to reduce the α value in response to saturation error, while η_2 defines the influence of the tracking error in increasing the α value.

$$\Delta\alpha = -\eta_1 e_{sat_{norm}} + \eta_2 e_{norm} \tag{5.6}$$

Through this design, when the control signal is saturated, the integral effect is suppressed to reduce the windup problem. On the other hand, when the system output is far from the reference, the integral effect is increased to produce a more aggressive response. The proposed anti-windup strategy is based on adjusting the parameter α according to the saturation error and the reference tracking error. In this context, the main objective is as follows:

As expected, the equation includes the saturation error, which is a commonly used parameter in anti-windup methods found in the literature. This term has a negative sign and aims to reduce the effect of the integrator by lowering the integral gain in the presence of saturation and in proportion to its severity.

The integral gain becomes a value that is adjusted online through the defined Δα equation. Introducing an adjustable alpha parameter creates an additional benefit. By incorporating the reference tracking error into the equation, the aim is to improve the performance of the closed-loop system. In particular, controlling the deviation from the reference is expected to contribute to the reduction of overshoot.

The coefficients η_1 and η_2 used in updating the α value are not chosen as fixed constants. Instead, they are related to the system parameters to ensure that the method

can be applied to all FOPTD system models. As a result of the studies conducted within this scope, a set of parametric equations has been proposed.

In high-gain systems, even small control signals can cause significant changes in the output. Whereas low-gain systems respond more slowly. The system response tends to behave more cautiously. By placing the parameter K in the denominator of the alpha update equation's this part, the aim is to enable systems with low gain to exhibit more aggressive responses through the controller, while adopting a more cautious approach for systems with high gain that may already produce aggressive responses.

The time constant T indicates how quickly the system reacts. Large T values correspond to slower systems, while smaller values indicate faster responses. For slow systems (large T), the integral action should be applied more cautiously. This means the increase in α should be more gradual. This is achieved by placing T in the denominator of the η_2 formulation.

The delay time L is one of the most critical and challenging parameters in this design. As dead time increases, the effect of the controller's command is delayed, and the system becomes harder to observe and respond to. This can lead to instability, especially when aggressive integral action is present.

In this formulation, L is placed in the numerator rather than the denominator. As the time delay increases, the duration during which the system deviates from the reference also increases. The parameter L is placed in the numerator to allow the controller to generate more aggressive control signals when the system is far from the reference.

To determine the parameters of the α update equation, simulations were carried out using an example FOPDT system. The selected system parameters are given as below:

- K = 1.5
- T = 3.5,
- L = 0.5

As a result, the FOPDT system model is conducted as shown in (5.7).

$$G(s) = \frac{1.5}{3.5s + 1} e^{-0.5s} \tag{5.7}$$

Saturation limit for $G(s)$ is chosen in $\pm$ 1.1 bound.

The parameters of the fractional-order PI controller were determined using the method proposed by Bhambhani et al. [2] and applied by Ranganayakulu et al. [3]. This tuning rule was specifically designed and implemented for FOPDT systems, using the jitter margin and ITAE criteria. One of the reasons for its use in this study is its ease of implementation. The primary objective of this work is not to design a FOPI controller, but rather to develop an anti-windup scheme. This tuning rule was chosen because it is tailored for FOPDT systems and is straightforward to implement. The tuning rules recommended by this method are provided in Table 5.1.

Table 5.1 : FOPI controller tuning rule.

K_p	K_i	α
$\frac{0.2 \cdot T}{L} + 0.16$	$\frac{0.25}{T \cdot L} + \frac{0.19833}{L} + 0.09$	$\frac{L}{L+T} - 0.04 \cdot L + 1.2399$

When this tuning rule is applied, the controller parameters are found as below.

- $K_p = 1.56$
- $K_i = 0.6295$
- $\alpha = 1.3449$

As a result, the controller equation is formulated as given in (5.8).

$$C(s) = 1.56 + \frac{0.6295}{s^{1.3449}} \tag{5.8}$$

5.4 Effect of System's Gain to Tracking Error Contribution

In this section, the effect of the system's gain K on the η_2 coefficient in the α update rule has been analyzed. All other parameters in the update rule were kept constant, and only the position of K within the η_2 expression was modified.

Table 5.2 presents the results showing how placing the system gain K in the numerator or denominator of the equation affects the time-domain responses. The results indicate that when K is placed in the numerator, the system exhibits a larger overshoot (7.20%) and a slower settling time (14.15 s). In contrast, when K is placed in the denominator,

the overshoot significantly decreases (4.09%) and the settling time improves notably (10.50 s).

Table 5.2 : Effect of K on η_2.

Time Domain Specs.	K in the denominator	K in the numerator
Overshoot	4.0928	7.1951
Settling Time	10.5034	14.1549

Figure 5.2. compares the effect of the K value on η_2 using step responses. The difference is clearly visible in the graph: the curve with K in the denominator reaches steady state faster with a smoother initial peak, while the curve with K in the numerator exhibits a sharper peak and longer oscillations.

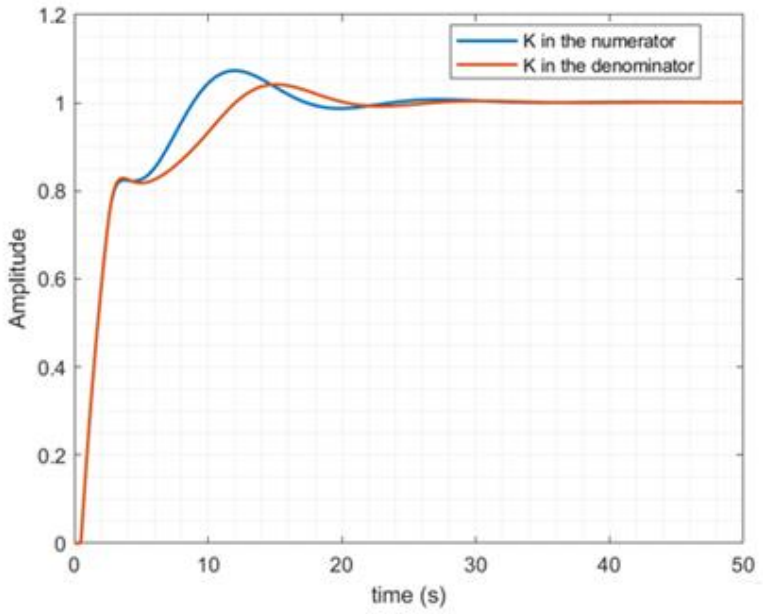

Figure 5.2 : The effect of K on the step response depending on its position in η_2.

These results prove placing K in the denominator of the η_2 formulation in α update rate. It is clear that placing K in the denominator rather than the numerator of the η_2 coefficient yields more balanced control performance.

5.5 Effect of Time Constant to Tracking Error Contribution

Table 5.3 presents the time-domain specifications for the effect of T on η_2. According to the results, when T is placed in the numerator, the system overshoot increases to

9.05%, and the settling time rises to 12.87 seconds. In contrast, when T is placed in the denominator, the overshoot decreases to 4.09%, and the system reaches steady state in a shorter time of 10.50 seconds.

Table 5.3 : Effect of T on η_2.

Time Domain Specs.	T in the denominator	T in the numerator
Overshoot	4.0928	9.0484
Settling Time	10.5034	12.8703

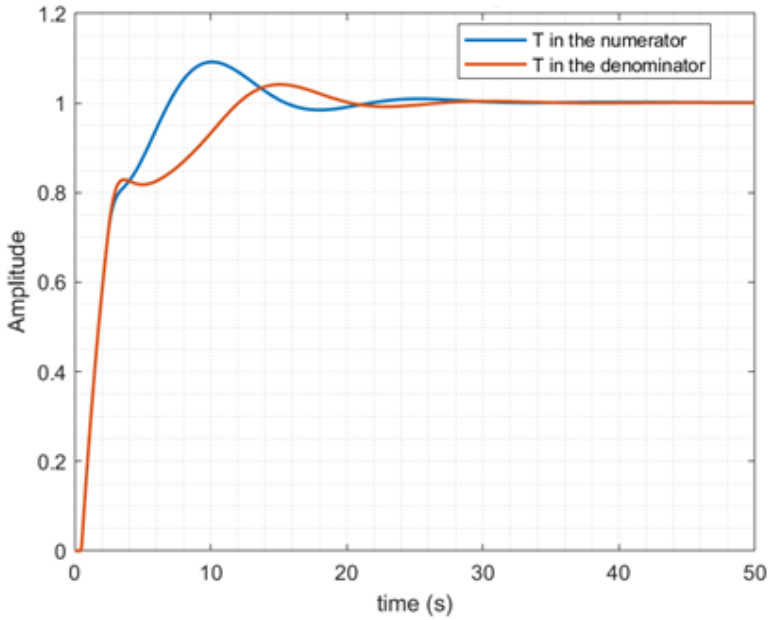

Figure 5.3 : The effect of T on the step response depending on its position in η_2.

As shown in the Figure 5.3, when the parameter T is placed in the denominator, the closed-loop system exhibits lower amplitude and shorter oscillations, resulting in a more stable response.

In contrast, placing T in the numerator leads to a bigger overshoot and longer settling behavior. These differences indicate that including T in the numerator of the η_2 formulation increases the integrative effect of the controller, which in turn leads to higher overshoot and damping.

The results support that defining η_2 as inversely proportional to T is a more balanced and safer choice for achieving better control performance.

5.6 Effect of Dead Time to Tracking Error Contribution

Table 5.4 presents the analysis of the effect of dead time on the formulation based on time-domain responses. The results show that when L is placed in the denominator, the system's overshoot increases to 8.10%, and the settling time extends to 13.54 seconds. In contrast, when L is placed in the numerator, the overshoot is more limited at 4.09%, and the system settles in 10.50 seconds. This indicates that the system exhibits a more balanced transient response.

Table 5.4 : Effect of L on η_2.

Time Domain Specs.	L in the numerator	L in the denominator
Overshoot	4.0928	8.1000
Settling Time	10.5034	13.5433

The step response shown in Figure 5.4., titled "Step response for effect of L on η_2", supports these findings. The curve with L in the numerator exhibits lower overshoot and faster damping. On the other hand, the curve with L in the denominator tends to show a higher amplitude overshoot and takes longer to reach steady state.

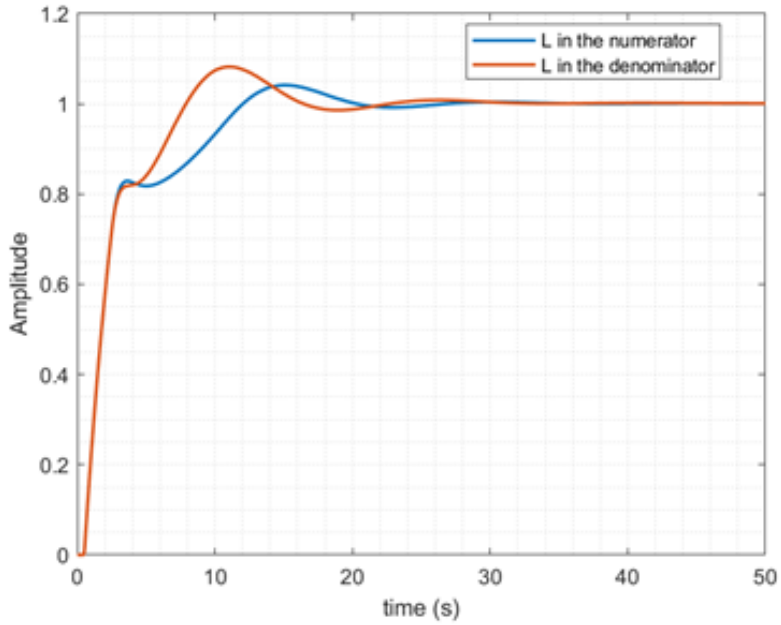

Figure 5.4 : The effect of T on the step response depending on its position in η_2.

This difference indicates that placing L in the denominator of the η_2 definition does not mitigate the effect of delay but instead causes α to increase more rapidly, leading to an uncontrolled rise in the integrative action of the controller. Since delay is a critical parameter for system stability, such an increase results in higher overshoot and slower recovery.

5.7 Finalized Boundaries, Parameter Set and Update Rule

For η_1 contribution of the Equation 5.5, FOPDT system parameters are not used. The main reason of that the equation for η_1 represents the part where the integral gain is updated based on the saturation error. The saturation error is not directly related to the system itself; rather, it is associated with the controller (i.e., the generated control signal) and the saturation limits (i.e., the allowable control signal). For η_2 contribution of the Equation 5.5; the situation differs. The effects of the system parameters are examined in detail.

The proposed expression for η_1 is given in (5.9). The saturation error varies proportionally with a coefficient k1, which determines the rate of change of the delta alpha value. Because the magnitude of the k1 value is directly related to the variation range of the delta alpha value.

$$\eta_1 = k_1 \tag{5.9}$$

The proposed expression for η_2 is given in (5.10). As previously discussed in detail; K, L, and T represent the system gain, delay time, and time constant, respectively. The parameter k_2 is a coefficient that determines the rate of change of the delta alpha value.

$$\eta_2 = k_2 \frac{L}{KT} \tag{5.10}$$

In the proposed update rule, it is said that the α parameter is updated online based on the system's tracking and saturation behaviors. However, allowing this update to occur without constraints may lead to instability or undesirable transient performance due to uncontrolled growth or reduction in α.

To prevent this, the parameter is constrained within specified bounds. In other words, the update rule is designed to ensure that the computed value remains within a defined

range. For instance, if the updated value exceeds the upper limit, the maximum allowable value is applied to the controller. Similarly, if it falls below the lower limit, the controller is updated using the minimum specified value. Equation (5.11) defines the boundaries for the α value.

$$\alpha_{t_{calc}} \epsilon [\alpha_{min} , \alpha_{max}] \tag{5.11}$$

The term $\alpha_{updated}$ represents the integral gain value of the controller at the moment of update. The bound is defined as $[0.5 \cdot \alpha_{init} , \alpha_{init}]$ after many simulations and reasons. α_{init} denotes the integral gain of the controller designed without considering saturation limits at first. The reasons are several why this bounds are chosen in this way.

During the control design phase, before considering constraints such as saturation, a nominal value α_{init} is assumed. This value is expected as already chosen based on the system's transfer function, error dynamics, and desired reference tracking performance etc. Allowing α to exceed this designed value may lead to undesired system behavior. Therefore, the upper bound for α is introduced to prevent the controller from deviating beyond its nominal behavior and to maintain the validity of the original design. The reason for selecting the lower bound in this manner is that, in numerous simulations, this constraint was observed to yield the most favorable results.

Additionally, when designing fractional-order controllers, the parameter α is mostly defined in the range $0 < \alpha < 2$. This range is important both for modeling physically meaningful system behaviors and for maintaining analytically valid operations in the Laplace domain. For example, when $\alpha < 0$, the integration loses its meaning, while $\alpha > 2$ can make the controller overly aggressive, seriously threatening system stability. In the literature, particularly in $PI^{\lambda}D^{\mu}$ controllers, the parameters λ and μ are also kept within this range.

The alpha change rate should be values that will prevent sudden changes in the control signal but will not reduce responsiveness. As a result of the analyses made for different systems, k values were determined as (5.12).

$$\mathrm{k}_1 = 10^{-2} , k_2 = 10^{-4} \tag{5.12}$$

As a result, a comprehensive update rule has been derived. When all parameters are incorporated, the resulting update equation takes the form presented in (5.13).

$$\Delta\alpha = -10^{-2} e_{sat_{norm}} + 10^{-4} \frac{L}{KT} e_{norm} \tag{5.13}$$

The calculated alpha value, obtained through this calculation, is presented in (5.14).

$$\alpha_{t_{calc}} = \alpha_{t-1} + \Delta\alpha \tag{5.14}$$

When the boundaries are also taken into account, the updated alpha value is governed by the rule expressed in (5.15).

$$\alpha_t = \begin{cases} \alpha_{min} & if\ \alpha_{t_{calc}} < \alpha_{min} \\ \alpha_{t_{calc}} & if\ \alpha_{min} < \alpha_{t_{calc}} < \alpha_{max} \\ \alpha_{max} & if\ \alpha_{max} < \alpha_{t_{calc}} \end{cases} \tag{5.15}$$

6. SIMULATIONS

In this section, the proposed strategy and existing methods from the literature, as discussed in the previous sections, are compared for various FOPDT systems controlled by FOPI controllers.

The simulations were conducted using MATLAB and SIMULINK environments. In the SIMULINK models, the Runge-Kutta method was employed as the fixed-step solver, with a step size set to 10^{-5}.

During model development, features from the Control Toolbox were utilized. For the fractional integrator, the FOMCON toolbox was implemented in the models. Frequency boundaries were defined as $[10^{-3}, 10^{3}]$, in accordance with common practice. The approximation order was set to 5 to ensure a balance between computational efficiency and accurate system behavior [29].

For the Fractional Integral Order Reset and Proposed Strategy, the "Fractional Variable Order Derivative Simulink Toolkit" was used. In this case, the sampling time was set to 10^{-1}, and the N_{buffer} parameter was determined based on the formulation recommended by the toolkit's developers, defined as "$simulation\_time$" divided by "$sampling\_time$" [30].

The saturation limit for the systems has been standardized as 15% less than the maximum control signal produced by the system without saturation. The maximum control signal was determined to be 1.835 for "System 1" and 1.732 for "System 2".

6.1 Comparison for "System 1"

The first FOPDT system and the corresponding FOPI controller selected for comparison are presented in (6.1) and (6.2), respectively.

$$G_1(s) = \frac{1}{0.8s + 1}\, e^{-0.2s} \tag{6.1}$$

According to the defined standard, the saturation limit for this system has been set to 1.596, which is 15% less than the maximum control signal produced by the system without saturation.

The system has a medium-level gain, a low time constant, and a time delay. The low time constant results in a relatively aggressive system behavior. The integral gain and integral order of the controller have been selected using the tuning rule discussed in the previous section.

$$C_1(s) = 0.96 + \frac{2.6442}{s^{1.4319}} \tag{6.1}$$

This system–controller pair yields a response that is difficult to control and exhibits excessive oscillations. The behavior of the methods under this scenario has been examined.

The effects of different anti-windup methods on system responses and control signals under a unit step input are compared. Figure 6.1 shows the system outputs. Figure 6.2 and Figure 6.3 illustrate the control signals before and after saturation block, respectively. In addition, Table 6.1 presents the step response metrics corresponding to each method.

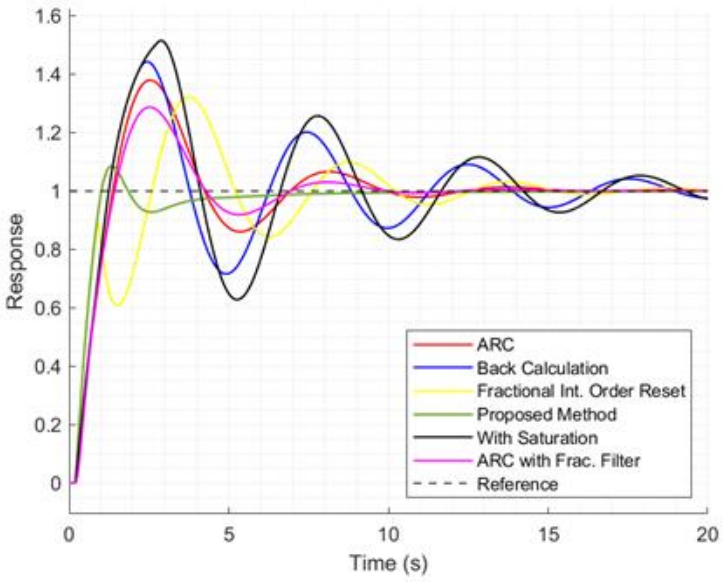

Figure 6.1 : Comparison of step responses for System 1.

The label "With Saturation" represents the case where no anti-windup method is applied, and only a saturation limit is imposed on the control signal. This case serves

as a reference, as it reveals the system's natural behavior and the negative effects caused solely by saturation. Observations show that in this scenario, the system exhibits a high overshoot of approximately 52.13% and a long settling time of around 18 seconds, along with pronounced oscillations and delays, indicating an unstable response.

Table 6.1 : Comparison of step response performances for System 1.

Anti Windup Method	Overshoot (%)	Settling Time (s)	Rise Time (s)	Peak Time (s)
With Saturation	52.1274	18.3160	0.8255	2.8791
Back Calculation	44.8334	15.2515	0.8257	2.4409
ARC	37.7816	8.7821	0.9284	2.5390
ARC with Fractional Filter	28.6280	6.1567	0.9024	2.5222
Fractional Int. Order Reset	32.0280	9.6360	2.1347	3.7409
Proposed Strategy	8.5773	3.2515	0.6224	1.3000

When anti-windup methods are applied, significant improvements in system behavior are observed. For instance, classical methods such as back calculation and ARC reduce the overshoot to the 35–45% range, yet still fail to fully suppress oscillations and achieve adequate settling times, thus not ensuring complete system stability. The fractional integral order reset and fractional filtered ARC methods alleviate these issues to some extent but cannot completely eliminate them.

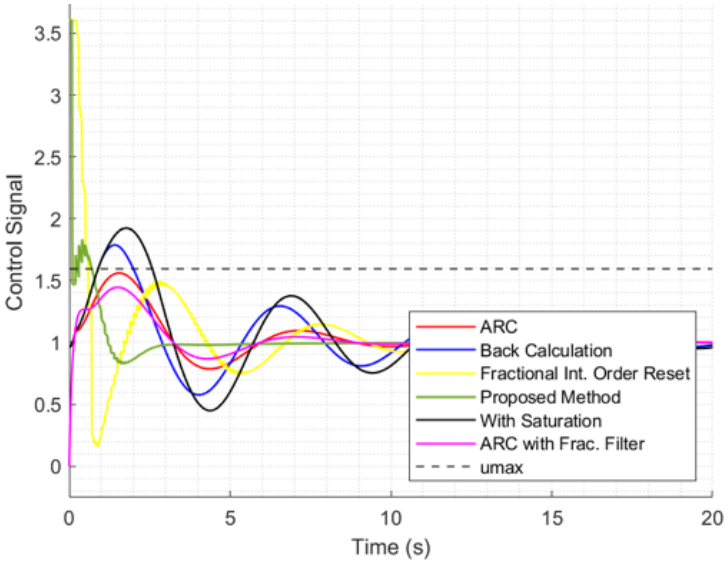

Figure 6.2 : Comparison of control signals (without saturation) for System 1.

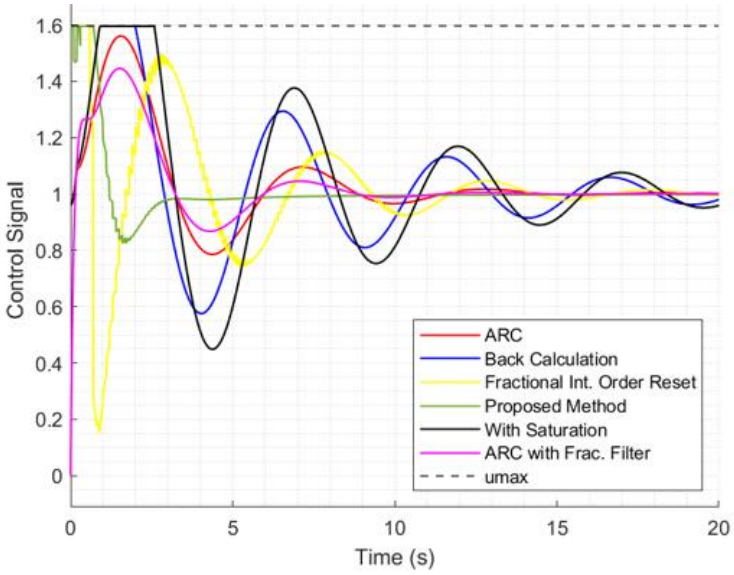

Figure 6.3 : Comparison of control signals (with saturation) for System 1.

When the control signals are examined, it is observed that, although the proposed strategy initially produces a high control input, it quickly brings the signal below the saturation limit.

Figure 6.4 illustrates the values taken by the integral order in the proposed strategy. Starting from an initially assigned value, the integral order shows a rapid decreasing trend. As the system approaches the reference and the produced control signal falls below the saturation limit, the integral order begins to stabilize, exhibiting less abrupt changes. Over time, it is observed to converge toward a value close to 1.1.

Among all the methods, the best performance is achieved by the proposed anti-windup strategy. This method results in only 8.58% overshoot, a fast rise time of 0.62 seconds, and a settling time of 3.25 seconds. It produces a response that is both fast and well-controlled.

6.2 Comparison for "System 2"

As the second example system, a model with a high time constant, which avoids relatively aggressive behavior, was selected. In contrast, the time delay was set to a longer value. The controller was tuned using the previously discussed tuning method.

The FOPDT system's and the corresponding FOPI controller selected for comparison are presented in (6.3) and (6.4), respectively.

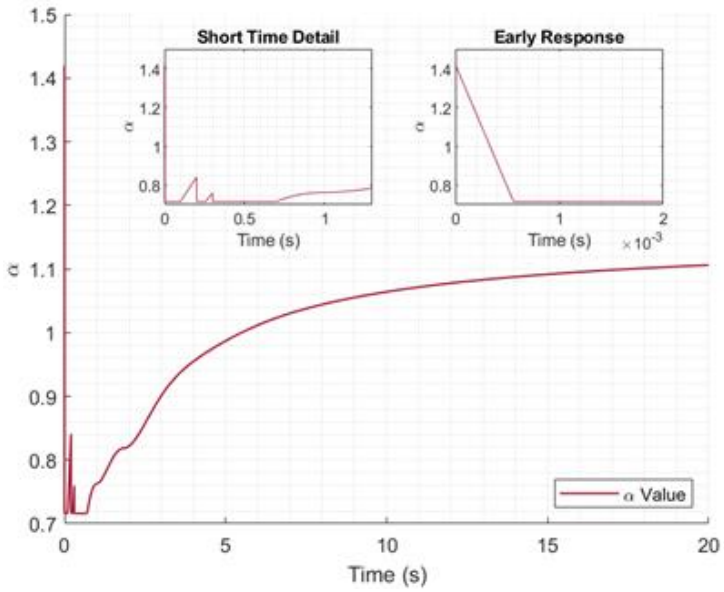

Figure 6.4 : The variation of the integral order with respect to time for System 1.

$$G_1(s) = \frac{1.5}{3.5s + 1}\, e^{-0.5s} \tag{6.3}$$

According to the defined standard, the saturation limit for this system has been set to 1.506, which is 15% less than the maximum control signal produced by the system without saturation.

$$C_1(s) = 1.56 + \frac{0.62}{s^{1.3449}} \tag{6.4}$$

Table 6.2 : Comparison of step response performances for System 2.

Anti Windup Method	Overshoot (%)	Settling Time (s)	Rise Time (s)	Peak Time (s)
With Saturation	32.7853	15.3065	1.6674	5.4430
Back Calculation	30.6774	15.2523	1.7120	5.5452
ARC	40.0102	17.5478	1.6299	4.7723
ARC with Fractional Filter	22.0653	13.3853	1.9221	5.4501
Fractional Int. Order Reset	13.8134	11.5542	3.4459	8.1000
Proposed Strategy	0.8885	7.2823	1.8428	15.1000

In this part, the effects of different anti-windup methods on the Example 2 system are evaluated. While Figure 6.5 shows the system outputs, Figure 6.6 and Figure 6.7

illustrate the control signals before and after saturation block, respectively. In addition, Table 6.2 presents the step response metrics corresponding to each method for Example 2 system & controller pair.

With saturation case represents the baseline system behavior, where only a saturation limit is imposed on the control signal without any anti-windup method applied. In this scenario, the system exhibits a high overshoot of approximately 32% and reaches steady state in approximately 15 seconds. Notable oscillations and prolonged transient behavior are also observed in the system output.

Although one of the classical methods back calculation reduces the overshoot to the 30% range, it offers only limited improvement due to prolonged settling times and transient saturation observed in the control signals.

The ARC with fractional filter method further reduces the overshoot to 22%, yet still fails to sufficiently suppress oscillations. The fractional integral order reset method yields a 14% overshoot and a significantly delayed peak time as 8 seconds, producing a transient response that is oscillatory but controlled.

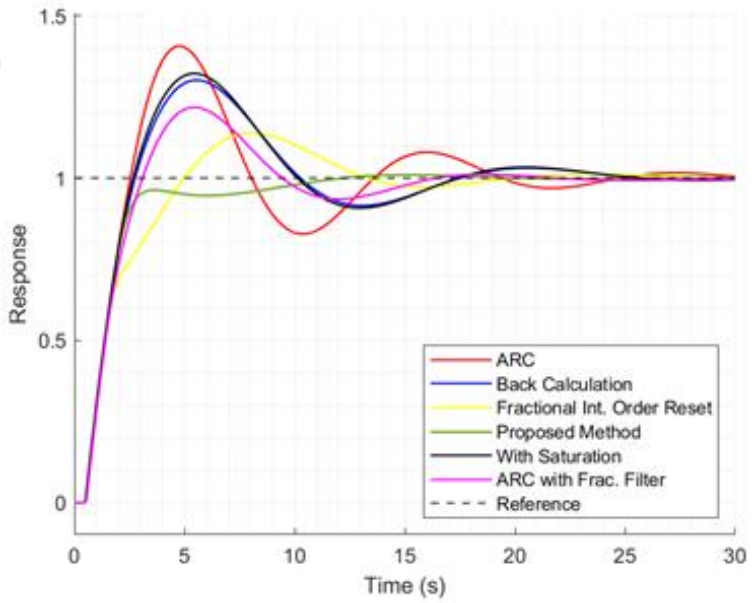

Figure 6.5 : Comparison of step responses for System 2.

In contrast, the proposed strategy achieves a negligible overshoot of only 0.89% and reaches steady state in just 7.28 seconds. With a rise time of 1.84 seconds, the system demonstrates a response that is both fast and stable.

These results indicate that the proposed strategy offers the most successful solution in terms of both overshoot suppression and rapid reference tracking then other classical and fractional-order controller based methods.

An analysis of the control signals reveals that the fractional integral order reset and proposed strategy methods start with relatively large signals compared to the other methods. The ARC and ARC with fractional filter methods are able to inherently limit the control signal due to the presence of a model-based saturation block within the controller.

Notably, the proposed strategy, in particular, quickly falls below the saturation limit and settles at a stable value, thereby avoiding unnecessary stress on the system. This behavior provides a significant advantage in terms of both system safety and actuator durability.

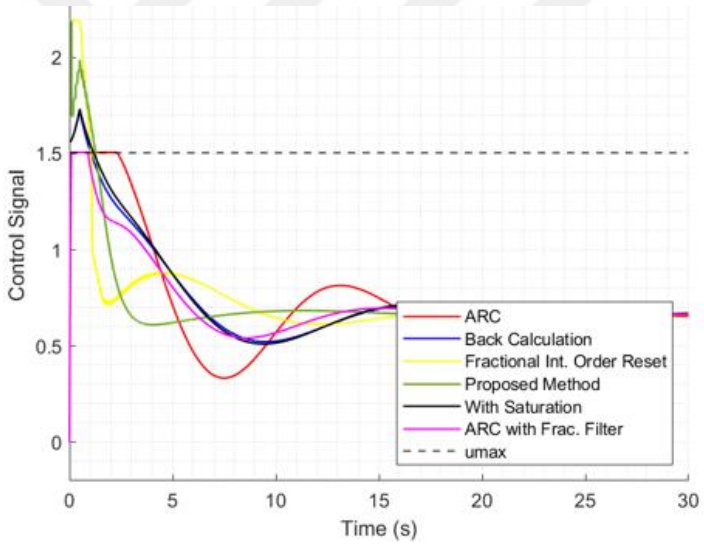

Figure 6.6 : Comparison of control signals (without saturation) for System 2.

Figure 6.8 presents the graph of integral order by time. The alpha value initially drops rapidly to its minimum. As the control signal decreases and the system approaches the reference, it begins to increase again.

Around the 11th second, the integral order returns to its initial value and continues with the predefined parameters set in the controller design. As mentioned before, the initial value of alpha is the best value chosen for the controller in the ideal case where no windup problem is encountered.

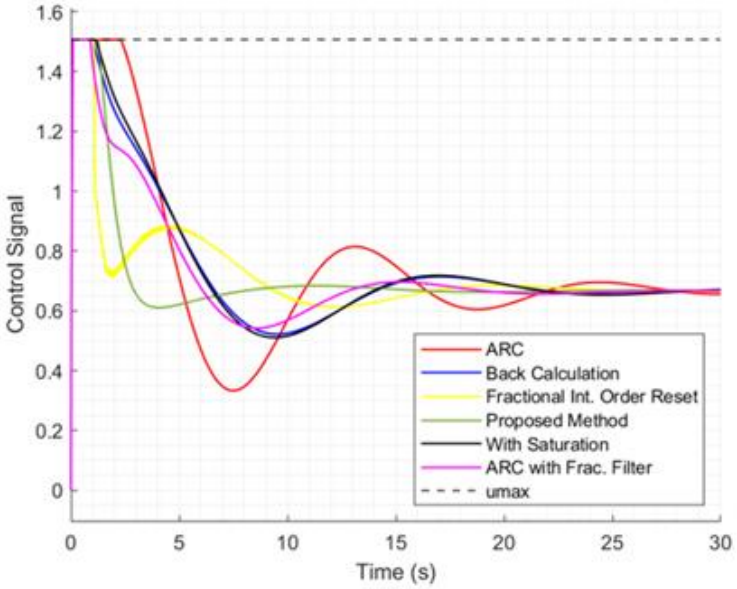

Figure 6.7 : Comparison of control signals (with saturation) for System 2.

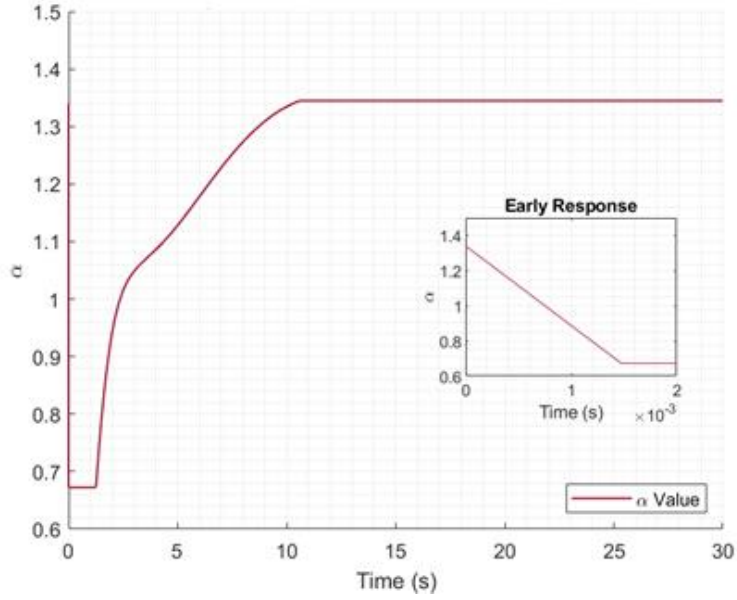

Figure 6.8 : The variation of the integral order with respect to time for System 2.

7. CONCLUSION

Throughout this thesis, the concept of integral windup has been thoroughly examined. The underlying causes of windup have been emphasized, and the effects of different saturation limit values on system response have been demonstrated through simulation studies. The contributions of controller parameters to the windup problem have been illustrated, with particular focus on how the integral order in fractional-order controllers influences this issue.

Well-established anti-windup methods from the literature have been reviewed, and their implementation in systems controlled by FOPI controllers has been described. However, the existing literature largely overlooks the potential advantages offered by the tunable integral order in fractional-order controllers.

An anti-windup method that adaptively adjusts the integral order online has been proposed. This approach utilizes both the saturation error and the reference tracking error as key parameters in determining the value of the integral order (α). The influence of FOPDT system parameters on these two criteria has been analyzed in detail, and the update rule has been generalized using the characteristics of FOPDT systems.

The proposed strategy and existing approaches have been comparatively evaluated for different FOPDT system dynamics and FOPI controller configurations. Four performance metrics have been used to support the analysis: rise time, settling time, overshoot, and peak time. The results demonstrate that the proposed strategy significantly improves system performance and offers an effective solution to the windup problem.

REFERENCES

[1] **Yumuk, E.** (2015). Kesirli Mertebe Tek Kutuplu Sistem Modeli İçin Tam Sayı Mertebe PID Kontrolör Tasarımı. M.S. thesis. Control and Automation Engineering Department, Istanbul Technical University, Istanbul.

[2] **P. Hippe**, (2006). Its Effects and Their Prevention. *Windup in Control, 1 ed. Advances in Industrial Control*. Springer. London pp. 3-6.

[3] **Muresan, C. I., & Ionescu, C. M**. (2020). Generalization of the FOPDT Model for Identification and Control Purposes. *Processes*, 8(6), 682. https://doi.org/10.3390/pr8060682

[4] **Xue, D., & Bai, L.** (2024). Fractional Calculus. *Springer Nature Singapore*. https://doi.org/10.1007/978-981-99-2070-9

[5] **Aburakhis, M., & Raúl Ordóñez.** (2024). Generalization of Direct Adaptive Control Using Fractional Calculus Applied to Nonlinear Systems. *Journal of Control Automation and Electrical Systems*, 35(3), 428–439. https://doi.org/10.1007/s40313-024-01082-0

[6] **La Oustaloup A.** (1991). *Commande CRONE*. Hermès, Paris.

[7] **Yumuk, E., Güzelkaya, M., & Eksin, İ.** (2019). Analytical fractional PID controller design based on Bode's ideal transfer function plus time delay. *ISA Transactions*, 91, 196–206. https://doi.org/10.1016/j.isatra.2019.01.034

[8] **Alatawi, K. S., Zaid, S. A., & El-Shimy, M. E.** (2024). Optimal Fractional-Order Controller for Fast Torque Response of an Asynchronous Motor. *Processes*, 12(12), 2914–2914. https://doi.org/10.3390/pr12122914

[9] **Parmanand Maurya, Paul, N., Prasad, D., & Singh, R. S.** (2024). Modified fractional order PID structure for non-integer model bioreactor control. *The Canadian Journal of Chemical Engineering*. 102(9), 3173–3191. https://doi.org/10.1002/cjce.25254

[10] **Sir Elkhatem, A., Engin, S. N., & Malik, Z.** (2025). New FOPID Control Design for Flight Dynamics With Special Phenomena. *Optimal Control Applications and Methods*. https://doi.org/10.1002/oca.3293

[11] **Markaroglu, H., Guzelkaya, M., Eksin, I., & Yesil, E.** (2006). Tracking Time Adjustment in Back Calculation Anti-Windup Scheme. *ECMS 2006 Proceedings* edited by: W. Borutzky, A. Orsoni, R. Zobel (pp. 613-618). European Council for Modeling and Simulation. doi:10.7148/2006-0613

[12] **H. A. Fertik and C. W. Ross,** Direct digital control algorithm with anti-windup feature. *ISA Transactions*, Vol. 6, No. 4, pp.317-328, 1967.

[13] **Glattfelder, A., & Schaufelberger, W.** (1983). Stability analysis of single loop control systems with saturation and antireset-windup circuits. *IEEE Transactions on Automatic Control*, 28(12), 1074–1081. https://doi.org/10.1109/tac.1983.1103180

[14] **Wittenmark, B., & Åström, K. J.** (1987). Self-Tuning Control. In M. G. Singh (Ed.), *Systems and Control Encyclopedia Pergamon Press Ltd*.

[15] **Astrom, K.J. and Hagglund, T.** (1995) PID Controllers: Theory, Design and Tuning. 2nd Edition. *Instrument Society of America, Research Triangle Park*.

[16] **Bohn C., Atherton D.P.** (1995). An analysis package comparing PID anti-windup strategies. *IEEE Control Systems*, 15(2), 34–40. https://doi.org/10.1109/37.375281

[17] **Hanus, R., Kinnaert. M., And Henrotte, S.-L.** (1987). Conditioning technique a general anti-windup and bumpless transfer method. *Automatica.*, pp. 729-739.

[18] **S. Pandey, P. Dwivedi, and Anjali Junghare.** (2017). A newborn anti-windup scheme based on state prediction of fractional integrator for variable speed motor. *International Conference on Control, Automation and Systems (ICCAS)*, pp. 663–668, doi: https://doi.org/10.23919/iccas.2017.8204312.

[19] **Berno e Silva, V., Resende Arce, B., de Souza Vieira, P. A., Ferreira Dos Santos, M., Santos Neto, A. F., & Mercorelli, P.** (2023). The Use of Anti-Windup Techniques in Didactic Level Systems: An Interesting Application. *2023 24th International Carpathian Control Conference (ICCC)*, 49–53. https://doi.org/10.1109/iccc57093.2023.10178982

[20] **Åström, K. J., & Tore Hägglund.** (2006). Advanced PID Control. *ISA*.

[21] **Wang, X., Gambier, A., & Vinagre, B. M.** (2020). Fractional Order PID Control with Rate-limited Anti-windup for the Pitch System of Wind Turbines. *2020 IEEE Conference on Control Technology and Applications (CCTA)*, 933–938. https://doi.org/10.1109/ccta41146.2020.9206341

[22] **Ahsan, M., Georgios Tzounas, & Milano, F.** (2020). Modeling and Simulation of Fractional Order PI Control Limiters for Power Systems. *IFAC-PapersOnLine*, 53(2), 13107–13112. https://doi.org/10.1016/j.ifacol.2020.12.2273

[23] **E. Torstensson** (2013). Comparison of Schemes for Windup Protection, M.Sc. thesis, Dept. of Automatic Control, Lund University, Lund, Sweden.

[24] **Meena, R., Chakraborty, S., Pal, V. C., & Lala, H.** (2024). Experimentally validated fractional-order PI with anti-windup for fractional-order plus time delay processes. *International Journal of Dynamics and Control*. https://doi.org/10.1007/s40435-024-01483-8

[25] **W. Malesza, & D. Sierociuk.** (2018). Fractional variable order anti-windup control strategy. *Bulletin of The Polish Academy Of Sciences Technical Sciences*, Vol. 66, No. 4. https://doi.org/10.24425/124258

[26] **Pathiran, A.** (2020). Effect of dead-time approximation on controller performance/robustness designed for a first order plus dead-time model. *Journal of Electrical Engineering*.

[27] **Enwerem, C., & Okoro, I. S**. (2022). Optimal controller tuning technique for a first-order process with time delay (arXiv:2210.08187). *arXiv*. https://arxiv.org/abs/2210.08187

[28] **Neha Belwal, Pradeep Kumar Juneja, Sandeep Kumar Sunori, Govind Singh Jethi, & Sudhanshu Maurya**. (2022). Modeling and Control of FOPDT Modeled Processes—A Review. *Lecture Notes in Networks and Systems*, 255–260. https://doi.org/10.1007/978-981-19-2538-2_25

[29] **Tepljakov, A., Petlenkov, E. and Belikov, J.** (2011). FOMCOM: a MATLAB toolbox for fractional-order system identification and control, *International Journal of Microelectronics and computer science*, 2(2), 51–62.

[30] **Sierociuk, D**., Fractional Variable Order Derivative Simulink Toolkit, https://www.mathworks.com/matlabcentral/fileexchange/38801-fractional-variable-order-derivative-simulink-toolkit.

CURRICULUM VITAE

Name Surname : Muhammed Ali ELAYDIN

EDUCATION :

- **B.Sc. :** 2022, Istanbul Technical University, Faculty of Electrical and Electronics Engineering, Control and Automation Engineering Department

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2022 – Present, Researcher, TÜBİTAK RUTE.

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Elaydın M.A.,** Yumuk E. and Güzelkaya, M. 2025: Comparison of Anti Windup Methods for Fractional Order PI Controllers. International Graduate Reserch Symposium , May 12-14, 2025 İstanbul, Türkiye.

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

- **Elaydın M.A.** and Sonkaya F**.** 2024: Boru İçi Denetleme Robotu İçin Bir XYZ Haritalama Algoritması Tasarımı ve Uygulanması, 25. Otomatik Kontrol Ulusal Konferansı, September 12-14, 2024 Konya, Türkiye.