Numerical Optimization of Fractional Order PID Controller

Hassan N.A. Ismail\(^1\), I.K. Youssef\(^2\) and Tamer M. Rageh\(^3,\ast\)

\(^{1,3}\) Department of Basic Science Engineering, Faculty of Engineering in Benha, Benha University, Benha 13512, Egypt
\(^2\) Department of Mathematics, Faculty of Science, Ain Shams University, Cairo 11566, Egypt

ABSTRACT: The fractional order PID controller is the generalization of classical PID controller, many Researchers interest in tuning FOPID controller here we use the Pareto Optimum technique to estimate the controller parameter and compare our result with the classical model and with other Researchers result. We used both mathematica package and matlab for tuning and simulation.

KEYWORDS: Proportional Integral Derivative (PID) - fractional order PID - Optimization - Pareto Optimum

I. INTRODUCTION

The fractional order controllers are being the aim of many engineering and scientists in the recent few decay [1-5]. The fractional order Proportional-Integral-Derivative (FOPID) was first introduced by Podlubny [2] and it consider as the generalization case of classical PID controllers. The Proportional-Integral-Derivative (PID) controllers are still the most widely controller in engineering and industrial for process control applications. If the mathematical model of the plant can be derived, then it is possible to apply various design techniques for determining parameters of the controller that will meet the transient and steady state specifications of the closed loop system.

In the recent few decay due to the development of fractional calculus (FC) the modeling of engineering system can be appear in fractional order systems (FOS) that require much more than classical PID controller to meet both transient and steady state specifications.

There are many methods used to design FOPID, Deepyaman at. al. [4] using Particle Swarm Optimization Technique. Synthesis method which a modified root locus method for fractional-order systems and fractional order controllers was introduced in [8]. A state-space design method based on feedback poles placement can be viewed in [10].

The aim of design PID controller is achieve high performance including low percentage overshoot and small settling time. The performance of PID controllers can be further improved by appropriate settings of fractional-I and fractional-D actions.

\[ G_c(s) \]

\[ G(s) \]

\[ R(s) \]

\[ E(s) \]

\[ U(s) \]

\[ C(s) \]

Consider the simple unity feedback control system shown in fig. 1 where \( R(s) \) is an input, \( G(s) \) is the transfer function of controlled system, \( G_c(s) \) is the transfer of the controller, \( E(s) \) is an error. \( U(s) \) is the controller's output, and \( C(s) \) is the system's output.

II. FRACTIONAL ORDER CALCULUS [11-15]

Fractional calculus (FC) is a generalization of integration and differentiation to non-integer orders. FC provides a more powerful tool for modeling the real live phenomena, and this is actually a natural result of the fact that in FC the integer orders are just special cases.

Definition: Let \( \alpha \in \mathbb{R}^+ \). The operator \( J^\alpha_a \) defined on \( L^1[a, b] \) by

\[ J^\alpha_a \]

www.ijmsi.org 15 | Page
Numerical Optimization of Fractional Order PID Controller

for $a \leq t \leq b$ is called the Riemann-Liouville fractional integral operator of order $\alpha$

Definition: Let $\alpha \in \mathbb{R}^+$ and $n = \lceil \alpha \rceil$. The operator $D^\alpha_a$ defined as

$$D^\alpha_a f(t) = D^n J^\alpha_a f(t)$$

(2)

for $a \leq t \leq b$, is called the Caputo differential operator of order $\alpha$

Definition: Let $\alpha \in \mathbb{R}^+$. The operator $D^\alpha_a$ defined by

$$D^\alpha_a f(t) = \frac{1}{\Gamma(n - \alpha)} \int_a^t (t - \tau)^{n-\alpha-1} f(\tau) \, d\tau \quad n - 1 < \alpha < n$$

for $a \leq t \leq b$, is called the Grünwald-Letnikov fractional derivative of order $\alpha$

Definition: Let $\alpha \in \mathbb{R}^+$. The operator $G^\alpha_a$ defined by

$$G^\alpha_a = \lim_{h \to 0} \frac{\Delta^\alpha_f(t)}{h^\alpha} = \lim_{h \to 0} \frac{1}{h^n} \sum_{r=0}^{n} (-1)^r \binom{\alpha}{r} f(t - rh) \quad \alpha > 0$$

(4)

for $a \leq t \leq b$, is called the Riemann-Liouville fractional integral of order $\alpha$

From the Riemann-Liouville fractional integral, applying the Laplace transform of the convolution integral, Equations (1) and (2) will be:

$$L[D^\alpha_a f(t)] = \frac{1}{\Gamma(n - \alpha)} L[t^{n-\alpha-1}] L[f(t)] = \frac{1}{\Gamma(n - \alpha)} \frac{\Gamma(n)}{s^n} F(s) = \frac{F(s)}{s^n}$$

(5)

$$= s^\beta F(s) - \sum_{k=0}^{n-1} s^{\beta-k} f^{(k)}(0)$$

(6)

III. FRACTIONAL ORDER CONTROLLER [16-19]

Before we introduced the Fractional Order Controller we introduce the fractional-order transfer function (FOTF) given by the following expression:

$$G_\alpha(s) = \frac{1}{\alpha_0 s^{\beta_0} + \alpha_{-1} s^{\beta_{-1}} + \cdots + \alpha_1 s^{\beta_1} + \alpha_0 s^{\beta_0}}$$

(7)

where $\beta_k (k = 0, 1, \ldots, n)$ is an arbitrary real number, $\beta_n > \beta_{n-1} > \cdots > \beta_1 > \beta_0 > 0, \alpha_k (k = 0, 1, \ldots, n)$ is an arbitrary constant.
In the time domain, the FOTF corresponds to the n-term fractional-order differential equation (FDE)
\[ a_n D^{\alpha_n} y(t) + a_{n-1} D^{\alpha_{n-1}} y(t) + \cdots + a_1 D^{\alpha_1} y(t) + a_0 D^{\alpha_0} y(t) = u(t) \] (8)

where \( D^\gamma \equiv \frac{d^\gamma}{dt^\gamma} \) is Caputo’s fractional derivative of order \( \gamma \) with respect to the variable \( t \) and with the starting point at \( \tau = 0 \).

The transfer function for conventional PID controller is
\[ G_{PID}(s) = \frac{u(s)}{e(s)} = K_c \left( 1 + \frac{1}{\tau_i s} + \tau_d s \right) \] (9)

Transfer function for fractional order PID controller is
\[ G_{FOPID}(s) = \frac{u(s)}{e(s)} = K_c \left( 1 + \frac{1}{\tau_i s^\alpha} + \tau_d s^\beta \right) \] (10)

FO integro-differential equation
\[ u(t) = k_p e(t) + k_i D^{-\lambda} e(t) + k_d D^{\mu} e(t) \] (11)

Where \( k_p, k_i, k_d \in \mathbb{R} \) and \( \lambda, \mu \in \mathbb{R}^+ \) are the parameters of controller to be tuned, and \( D^{-\lambda} \) and \( D^{\mu} \) are the fractional integral and differential operator respectively, often defined by the Riemann-Liouville definition as the following:

\[ D^{-\lambda} f(t) = \frac{1}{\Gamma(\lambda)} \int_0^t (t-\tau)^{1-\lambda} f(\tau) \, d\tau \] (12)

\[ D^{\mu} f(t) = \frac{1}{\Gamma(m+\mu)} \left( \frac{d}{dt} \right)^m \int_0^t (t-\tau)^{m+\mu} f(\tau) \, d\tau \] (13)

<table>
<thead>
<tr>
<th>Table 1 Controller Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K_p )</td>
</tr>
<tr>
<td>( K_d )</td>
</tr>
<tr>
<td>( K_i )</td>
</tr>
<tr>
<td>( \lambda )</td>
</tr>
<tr>
<td>( \mu )</td>
</tr>
</tbody>
</table>

The fractional system is a system which could be better described by fractional order mathematical models, and its transfer function is at arbitrary real order instead of just integer order.

Podlubny (1999) introduced [1] as a generalization of the classical PID controller, namely the PI\(^{\lambda}\)D\(^{\mu}\) controller or FOPID controller with an integrator of order \( \lambda \) and a differentiator of order \( \mu \). He also proves the better response of FOPID controller compared by PID controller special in case of FOS.

![Fractional Order PID Controller](image)

The orders of integration and differentiation \((\lambda, \mu)\) must be positive real numbers. Taking \( \lambda = 1 \) and \( \mu = 1 \), we will have an integer order PID controller. Fig. 2 The classical PID controller has three parameters \((K_p, T_i, T_d)\) to be tuned, while the fractional order PID controller has five \((K_p, T_i, T_d, \lambda, \mu)\).

The interest of this kind of controller is justified by a better flexibility, since it exhibits fractional powers \((\lambda\) and \(\mu\)) of the integral and derivative parts, respectively. Thus, five parameters can be tuned in this structure \((\lambda, \mu, K_p, K_i, K_d)\), that is, two more parameters than in the case of a conventional PID controller \(\lambda\).
The fractional orders \( \lambda \) and \( \mu \) can be used to fulfill additional specifications of design or other interesting requirements for the controlled system.

**Figure 3** Types of controllers

From fig. 3 at the corners of square if \( \lambda = \mu = 1 \), then it is classical PID controller. If \( \lambda = 0 \) and \( \mu = 1 \), then it is classical PD controller. If \( \lambda = 1 \) and \( \mu = 0 \), then it is classical PI controller. If \( \lambda = \mu = 0 \), then it is classical P controller. But any point inside the square donates a fractional order PID controller.

**IV. OPTIMIZATION OF CONTROLLER PARAMETERS**

The aims of most interested in FOPID controller is to estimate the controller parameters so many methods are done for example self-tuning and auto-tuning which introduced by Monje CA at. al [20], rule base method [21-24] for which FOPID controller based on Ziegler Nichols-type rules. Analytical methods [25-27], finally numerical treatment for optimization fractional order controllers has been introduced by various authors, based on the genetic algorithm[28-30], based on particle swarm optimization (PSO) technique[4 and 31-33] has also been used for estimating the controllers parameters, A multi-objective optimization method was designed by I. Pan and S. Das [34].

As in the classical root locus method, based on our engineering requirements of the maximum peak overshoot \( M_p \) and rise time \( t_{rise} \) (or requirements of stability and damping levels) we find out the damping ratio \( \zeta \) and the undamped natural frequency \( \omega_n \). Using the values of \( \zeta \) and \( \omega_n \) we then find out the positions of the dominant poles of the closed loop system,

\[
P_{1,2} = -\xi \omega_n \pm j\omega_n \sqrt{1 - \xi^2}
\]

Let the closed loop transfer function of the system is:

\[
\frac{G(s)}{1 + G(s)H(s)}
\]

Here \( G(s) = G_c(s)G_p(s) \) where \( G_c(s) \) is the transfer function of the controller to be designed. \( G_c(s) \) is of the form

\[
G_c(s) = Kp + Ti s^{-\lambda} + Td s^\mu
\]

\( G_p(s) \) is the transfer function of the process we want to control. If the required closed loop dominant poles are located at \( s_{1,2} = p_{1,2} = -x \pm jy, -x - jy \), then at \( s = p_1 = -x + jy \), we must have

\[
1 + G(p1).H(p1) = 0
\]

we get:

\[
1 + (Kp + Ti s^{-\lambda} + Td s^\mu).G_p(p1).H(p1) = 0.
\]

Assuming \( H(s) = 1 \), and \( G_p(s) \) being known, (18) can be arranged as:

\[
1 + [Kp + Ti(-x + jy)^{-\lambda} + Td(-x + jy)^\mu]G_p(-x + jy) = 0.
\]
In this complex equation (19) we have five unknowns, namely \{K_p, T_i, T_d, \lambda, \mu\}. There are an infinite number of solution sets for \(s = p_1 = x + jy\). So the equation cannot be unambiguously solved.

Pereto optimization helps us the find the optimal solution set to the complex equation.
Let:
R=real part of the complex expression,
I=imaginary part of the complex expression,
P=phase \((=\tan^{-1}(I/R))\).

We define \(f = |R| + |I| + |P|\) and minimize ‘f’ using the pareto optimization. Our goal is to find out the optimum solution set \(\{K_p, T_i, T_d, \lambda, \mu\}\) for which \(f = 0\).

The solution space is five-dimensional, the five dimensions being \(K_p, T_i, T_d, \lambda\) and \(\mu\). The personal and global bests are also five-dimensional. The limits on the position vectors of the particles (i.e. the controller parameters) are set by us as follows. As a practical assumption, we allow \(K_p\) to vary between 1 and 1000, \(T_i\) and \(T_d\) between 1 and 500, and \(\lambda\) and \(\mu\) between 0 and 2.

V. NUMERICAL EXAMPLES
Consider the system of fractional order transfer function which need to be controlled as the following:

\[
G_p(s) = \frac{1}{a_2 s^\beta + a_1 s^\alpha + a_0}
\]
where \(a_2 = 0.8, a_1 = 0.5, a_0 = 1, \beta = 2.2\) and \(\alpha = 0.9\)
and consider the FOPID transfer function as:

\[
G_c(s) = k_p + \frac{T_i}{s^{1-\lambda}} + \frac{T_d}{s^{1+\mu}}
\]

Using mathematica package and apply the pareto optimal algorithm with some constraints on the controller parameters (\(k_p\) to vary between 1 and 1000, \(k_i\) and \(k_d\) between 1 and 500, \(\lambda\) and \(\mu\) between 0 and 2) we estimate the parameters values as \((k_p = 215.20, k_i = 2.04, k_d = 73.42, \lambda = 1.18, \mu = 1.37)\).

![Figure 4: Comparison between PID and FOPID](image)
Numerical Optimization of Fractional Order PID Controller

Fig (4) show comparison between the output response of open loop transfer function (red line) and the classical PID controller (green line) and the FOPID controller using pareto optimization (blue line) and it is clearly how worst the open loop system with long time response and large peak over shot, but using PID controller all system requirements improved but still need more improvement, after using pareto optimal to estimate the controller parameter which make the system response be better with less peak over shot (we can claim that no peak over shot) and very small time response.

Fig (5) show comparison between the output response of closed loop transfer function and fractional order PID by using Particle Swarm Optimization Technique [4] (red line) and our method by using pareto optimization (blue line)

VI. RESULTS AND CONCLUSION

Her we used pareto method for numerical optimization of the FOPID which give an estimation of the controller parameter to meet the engineering specification needs, our result compared by classical PID and POS method

REFERENCES