

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

Detailed Study of Fractional Order PID Controllers using different Optimization Techniques with one Case Study

Omar Hanif¹, Dr. G.Uday Bhaskar Babu², Mullapudi Siva³

M.Tech Scholar, Process Control, National Institute of Technology, Warangal, Telangana, India¹

Assistant Professor, Process Control, National Institute of Technology, Warangal, Telangana, India²

M.Tech Scholar, Process Control, National Institute of Technology, Warangal, Telangana, India³

ABSTRACT: Fractional system analysis has received a great attention recently, both from the academic and industrial automations because of their increased flexibility with respect to integer-order systems. Recent efforts have been mostly focused on the synthesis of variants of Proportional Integral Derivative Controllers (PIDs) such as Tilt Integral Controller (TIDs), Fractional order PIDs (FOPIDs), and Multi-Fractional order PIDs (MFOPIDs) using different optimizing techniques. This paper highlights designing of PID (PID control) and its variants through Linear Matrix Inequality (LMI) optimizing technique and Fmincon (interior point algorithm) technique and compares the results of both. It further presents a temperature control case study demonstrating a lot of improvement in the responses of Fractional Controllers when compared with their Integer counterpart.

KEYWORDS: Fractional Order System, Linear Matrix Inequality, PID, FOPID, MFOPID, Interior Point Optimization algorithm

I. INTRODUCTION

1.1. *Fractional Calculus* deals with derivatives and integrals from non-integer orders. There are different definitions of Fractional Order differentiations and integrations. The well-established definitions include the Grünwald-Letnikov definition, the Cauchy integral formula, the Caputo definition and the Riemann-Liouville definition^{[1][2]}. The general fractional order differentio-integral operator can be summed as^[3] shown in (1):

$${}_a \mathcal{D}_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \Re(\alpha) > 0, \\ 1 & \Re(\alpha) = 0, \\ \int_a^t (dt)^{-\alpha} & \Re(\alpha) < 0, \end{cases} \quad (1)$$

Some of the definitions of fractional order calculus are:

Riemann-Liouville Fractional Integral and Derivative Operator^[1] as given in (2):

$${}^\beta D_t f(t) = \frac{1}{\Gamma(n-\beta)} \frac{d^n}{dt^n} \int_\alpha^t \frac{f(\tau)}{(t-\tau)^{\beta-n+1}} d\tau \quad (2)$$

where $n-1 < \beta < n$ $\Gamma(\cdot)$ is the Gamma function. The integrals (or derivatives) are of complex order β ($\Re(\beta) \neq 0$) representing an analytic continuation. *Grünwald-Letnikov Fractional Integral and Derivative Operator* given in (3) represents the derivative of order n

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

$${}_a D_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{r=0}^{\lfloor \frac{t-a}{h} \rfloor} (-1)^r \binom{\alpha}{r} f(t-rh) \tag{3}$$

and the n-fold integral of the function f(t) if $\alpha = n$ and $t = -n$ respectively where ${}^{[5]} \binom{\alpha}{r} = \frac{\alpha(\alpha-1)(\alpha-2)\dots(n-r+1)}{r!}$

The Caputo Fractional Differential Operator: When $\alpha > 0$, $t > a$, $[a, t] \in \mathbb{R}$, the Caputo fractional derivatives are defined ^{[1][4]} as shown in (4):

$${}^\beta_\alpha D_t f(t) = \begin{cases} \frac{1}{\Gamma(n-\alpha)} \int_a^t \frac{f^n(\tau)}{(t-\tau)^{\alpha+1-n}} d\tau, & n-1 < \alpha < n \in \mathbb{N} \\ \frac{d^n}{dt^n} f(t), & \alpha = n \in \mathbb{N} \end{cases} \tag{4}$$

1.2. *Fractional-order system* is obtained from the fractional-order differential equations. In general, the fractional-order transfer function (FOTF) of a single variable dynamic system can be defined in (5) ^{[4][5]}:

$$G(s) = \frac{b_0 s^{\gamma_0} + b_1 s^{\gamma_1} + \dots + b_m s^{\gamma_m}}{a_0 s^{\beta_0} + a_1 s^{\beta_1} + \dots + a_n s^{\beta_n}} \tag{5}$$

where $b_i (i = 0; 1 \dots m)$; $a_i (i = 0; 1 \dots n)$ are constants and $\gamma_i (i = 0; 1 \dots m)$; $\beta_i (i = 0; 1 \dots n)$ are arbitrary real or rational.

1.3. *PID controllers and their variants:* (a) PID algorithm consists of three basic coefficients: proportional (kp) integral (ki) and derivative (kd) which are varied to get optimal response. Fig. 1 shows the PID controller with process and sensor feedback ^[6]. The transfer function of PID controller is given in (6):

$$C(s) = kp + ki/s + kds \tag{6}$$

(b) Fractional-order PID Controller: The transfer function of $PI^\lambda D^\mu$ controller is given in (7) ^[7].

$$C(s) = kp + kis^{-\lambda} + kds^\mu \tag{7}$$

where λ and μ are positive real numbers; other constants are the same meaning as in PID.

(c) Tilt Integral Derivative Controller: The transfer function of TID controller is given in (8)

$$C(s) = ki \cdot s^{-1/n} + kt/s + kd \cdot s \tag{8}$$

This controller uses the exponent coefficient of $-1/n$. A tilt compensator gives an overall response which is closer to the theoretical optimal response. ^[7]

(d) Multiorder-Fractional-order PID Controller (MFOPID): An extension of FOPID, it demonstrates greater robustness and better response than FOPID controller. The transfer function of MFOPID controller is shown below in (14) ^[8]:

$$C(s) = kp + kis^\lambda + kd_1 s + kd_2 s^\mu + kd_3 s^\delta + \dots + kd_n s^n \tag{9}$$

where $\lambda, \mu, \delta, \dots, n$ are the powers of the compensated term.

1.4. *Linear Matrix Inequality (LMI)* is a very useful optimization tool which involves linear equality and inequality by solving in Matrix domain. A linear matrix inequality (LMI) has the form (10).

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

$$F(x) = F_0 + \sum_{i=1}^m x_i F_i > 0 \quad (10)$$

where $x \in \mathbb{R}^m$ is the variable and the symmetric matrices $F_i = F_i^T \in \mathbb{R}^{n \times n}$ ($i = 0, \dots, m$) are given. The inequality symbol in (10) means that $F(x)$ is positive-definite, i.e.

$u^T F(x) u > 0$ for all nonzero $u \in \mathbb{R}^n$. Syntax to define variable
 $[X, n, sX] = \text{lmivar}(\text{type}, \text{struct})$ (A)

Syntax for specifying the term content of an LMI ^[10]: $\text{lmiterm}(\text{termID}, A, B, \text{flag})$ (B)

Syntax of 3 solvers ^[10]: $[\text{tmin}, \text{xfeas}] = \text{feasp}(\text{lmisys}, \text{options}, \text{target})$
 $[\text{llopt}, \text{xopt}] = \text{gevp}(\text{lmisys}, \text{nlf}, \text{options}, \text{limit}, \text{xinit}, \text{target})$
 (C)

1.5. Nonlinear Optimization: *fmincon* is nonlinear constrained optimization method used in the paper based on three algorithms: Trust-region reflective, active set and interior Point method ^[12]. The general *fmincon* optimization problem can be written as (11) ^[12].

$$\min_x f(x) \text{ such that } \begin{cases} c(x) \leq 0 \\ \text{ceq}(x) = 0 \\ A \cdot x \leq b \\ \text{Aeq} \cdot x = \text{beq} \\ lb \leq x \leq ub, \end{cases} \quad (11)$$

where b and beq are vectors, A and Aeq are matrices, $c(x)$ and $\text{ceq}(x)$ are functions that return vectors, and $f(x)$ is a function that returns a scalar. $f(x)$, $c(x)$, and $\text{ceq}(x)$ can be nonlinear functions. x , lb , and ub can be passed as vectors or matrices. ^[12]

Syntax: ^[12] : $x = \text{fmincon}(\text{fun}, A, b, \text{Aeq}, \text{beq}, lb, ub)$ (D)

1.6. FOMCON tool box: It is one of those successful add-on software in MATLAB (as MATLAB does not support fractional analysis) which has been used in implementation of this paper. ^[5]

1.7. Rules in LMI optimization: Step1: The writing format while solving LMI optimization problem in MATLAB is by defining controller transfer function in the form of $C(s) = W(s)X$ and $L(s) = C(s)P(s)$ where $W(s)$ and X are the known weights and the unknown parameter vectors as shown in the Fig.1 ^{[8][9]}.

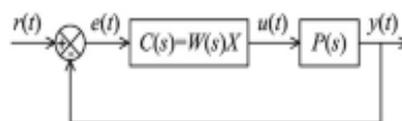


Fig.1 Closed Loop plant with Controller ^[9]

(a). PID Controller: PID controller is linear in its parameters. $X_{\text{PID}} = [k_p \ k_i \ k_d]^T$, $W_{\text{PID}}(s) = [1 \ s^{-1} \ s]$, $C_{\text{PID}}(s) = k_p + k_i/s + k_d s = [1 \ s^{-1} \ s]X = W_{\text{PID}}(s)X$, which is linear in X .

(b). TID Controller is also linear in its parameters $X_{\text{TID}} = [k_t \ k_i \ k_d]^T$, $W_{\text{TID}} = [s^{-1/n} \ s^{-1} \ s]$, $C_{\text{TID}}(s) = k_t/s^{1/n} + k_i/s + k_d s = [s^{-1/n} \ s^{-1} \ s]X = W_{\text{TID}}(s)X$

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

(c). MFOPID Controller: The MFOPID Controller can be written as : $X = [k_1, k_2, \dots, k_n]$, $W_{MFOPID} = [s^{\alpha_1} s^{\alpha_2} \dots s^{\alpha_n}]$
 $C_{MFOPID}(s) = [s^{\alpha_1} s^{\alpha_2} \dots s^{\alpha_n}]X = W_{MFOPID}(s)X$ [8][9]

Step2. Tuning in frequency domain: *If plant is stable*: LMI optimization in closed loop is applied after checking the Nyquist Stability criteria in the open loop, i.e. $L(s)$ (controller cum plant) = $C(s)P(s) = W(s)[X]P(s)$ should not encircle point $(-1,0)$ [8] as shown in the Fig.2 and given in (12) (13) and (14):

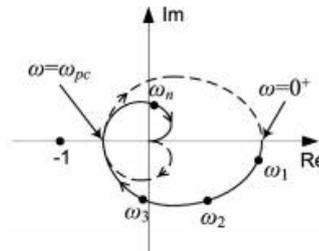


Fig.2 Assuming that $P(s)$ is stable, the Nyquist plot of $L(s) = W(s)XP(s)$ [8]

For closed-loop stability from Nyquist stability criterion:

$$\text{Im}\{L(j\omega)\} < 0, \quad 0 \leq \omega \leq \omega_{pc} \quad (12)$$

$$\text{Re}\{L(j\omega_{pc})\} > -1 \quad \wedge \quad \text{Im}\{L(j\omega_{pc})\} = 0 \quad (13)$$

$$\text{Im}\{L(j\omega_{pc})\} > 0 \quad \omega_{pc} < \omega \quad (14)$$

where ω_{pc} is phase crossover frequency [8]. On decomposing $W(j\omega)$ and $P(j\omega)$ into real and imaginary following (15) (16) (17) & (18) are obtained. [8]

$$W_i(j\omega)X P_r(j\omega) + W_r(j\omega)X P_i(j\omega) < 0, \quad 0 \leq \omega < \omega_{pc}, \quad (15)$$

$$W_r(j\omega_{pc})X P_r(j\omega_{pc}) - W_i(j\omega_{pc})X P_i(j\omega_{pc}) > -1, \quad (16)$$

$$-\epsilon < W_i(j\omega_{pc})X P_r(j\omega_{pc}) + W_r(j\omega_{pc})X P_i(j\omega_{pc}) < \epsilon, \quad (17)$$

$$W_i(j\omega)X P_r(j\omega) + W_r(j\omega)X P_i(j\omega) > 0, \quad \omega_{pc} < \omega. \quad (18)$$

where ϵ is a very small positive real constant, e.g., $\epsilon = 0.01$. In order to adjust the Gain Margin (GM) the following (19)

LMI equation can be applied [8]. $W_r(j\omega_{pc})X P_r(j\omega_{pc}) - W_i(j\omega_{pc})X P_i(j\omega_{pc}) > -1/\overline{GM}$. (19)

If plant is unstable: According to the Nyquist stability theorem the closed-loop system is stable if and only if the Nyquist plot of $L(s)$ (open loop) encircles the point $(-1,0)$ exactly once anticlockwise (in case of one unstable pole).

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

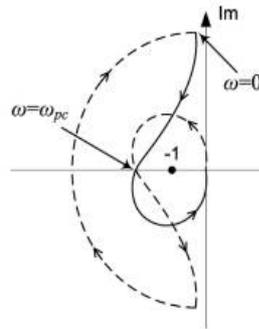


Fig.3 Assuming that $f(s)$ has one unstable pole. [8]

For closed-loop stability from Nyquist stability criterion, the relationship for getting stable system (20), (21) and (22) can be considered :

$$\text{Im}\{L(j\omega)\} > 0, \quad 0 \leq \omega \leq \omega_{pc} \quad (20)$$

$$\text{Re}\{L(j\omega_{pc})\} < -1 \quad \wedge \quad \text{Im}\{L(j\omega_{pc})\} = 0 \quad (21)$$

$$\text{Im}\{L(j\omega_{pc})\} < 0 \quad \omega_{pc} < \omega \quad (22)$$

where ω_{pc} is phase crossover frequency [8]. Decomposing $W(j\omega)$ and $P(j\omega)$ into real and imaginary parts the LMI equation for tuning X is given in (23), (24), (25) and (26) [8]:

$$W_i(j\omega)XP_r(j\omega) + W_r(j\omega)XP_i(j\omega) > 0, \quad 0 \leq \omega < \omega_{pc} \quad (23)$$

$$W_r(j\omega_{pc})XP_r(j\omega_{pc}) - W_i(j\omega_{pc})XP_i(j\omega_{pc}) < -1, \quad (24)$$

$$-\epsilon < W_i(j\omega_{pc})XP_r(j\omega_{pc}) + W_r(j\omega_{pc})XP_i(j\omega_{pc}) < \epsilon, \quad (25)$$

$$W_i(j\omega)XP_r(j\omega) + W_r(j\omega)XP_i(j\omega) < 0, \quad \omega_{pc} < \omega, \quad (26)$$

where ϵ is a small positive real constant. In order to guarantee the closed-loop stability with the minimum GM ($0 < GM < 1$) is given (27)

$$W_r(j\omega_{pc})XP_r(j\omega_{pc}) - W_i(j\omega_{pc})XP_i(j\omega_{pc}) < -1/GM. \quad (27)$$

Adjusting Phase Margin (PM) of the plant: The desired GM and PM is very necessary for the feedback system, given in (28) [8]:

$$\min_x \|W(j\omega_{gc})XP(j\omega_{gc}) - e^{j\phi^m}\| \quad (28)$$

where ω_{gc} is the desired gain crossover frequency, and

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

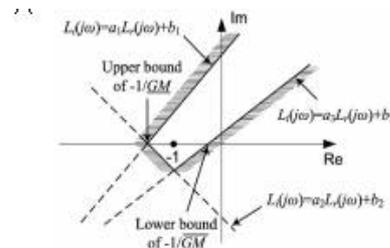
$\varphi_m = \pi + PM$ is the desired value for open-loop phase at ω_{gc} . Since (35) is not LMI, using Schur complement lemma^[13] the optimization problem is equivalent to the generalized eigenvalue problem given in (29)^[8]:

$$\begin{aligned} & \text{minimize } \gamma, \\ & \text{subject to} \\ & \begin{pmatrix} 0 & e^{j\varphi_m} - W(j\omega_{gc})XP(j\omega_{gc}) \\ (e^{j\varphi_m} - W(j\omega_{gc})XP(j\omega_{gc}))^H & 0 \end{pmatrix} < \gamma I. \end{aligned} \quad (29)$$

Converting the complex matrix into LMI compatible real matrix. solving the matrix for the real part results (37)

$$\begin{pmatrix} 0 & \cos\varphi_m - W_rXP_r + W_iXP_i & 0 & \sin\varphi_m - W_rXP_i - W_iXP_r \\ * & 0 & W_rXP_i + W_iXP_r - \sin\varphi_m & 0 \\ * & * & 0 & \cos\varphi_m - W_rXP_r + W_iXP_i \\ * & * & * & 0 \end{pmatrix} < \gamma I. \quad (30)$$

Higher order and time-delayed processes with one unstable pole: Plant $P(s)$ has exactly one unstable pole with time delay, the closed-loop system is stable if the Nyquist plot of $L(j\omega)$ remains in the hatched region in Fig. 4 for all $\omega \geq 0$ ^[8]:



The hatched region shows the location of the Nyquist plot of $L(j\omega)$ to achieve loop stability when $P(s)$ has exactly one unstable pole. ^[8]

The border of the hatched region in Fig.4 is marked by three lines in which the origin belongs to the hatched region (i.e., $b_3 > 0$) and secondly the point -1 stays out of it. Moreover, it is done $0 < a_3 < a_1$ to make sure that it is impossible to draw a simple closed contour in the hatched region which encircles -1. GM can be adjusted by appropriate choice of a_1, b_1, a_3, b_3 . The frequency at which the Nyquist plot intersects the negative real axis for the first time as ω_{gml} . The speed of the closed system response can be adjusted by making ω_{gml} large. The following LMI equation is formed in process with time delays given to keep $L(j\omega)$ in the hatched region. It is shown in (31) (32) (33) (34) and (35)^[8]:

$$W_i(j\omega_k)XP_i(j\omega_k) + W_r(j\omega_k)XP_r(j\omega_k) > a_1(W_r(j\omega_k)XP_r(j\omega_k) - W_i(j\omega_k)XP_i(j\omega_k)) + b_1, \quad 0 \leq \omega_k \leq \omega_1 \quad (31)$$

$$W_i(j\omega_k)XP_i(j\omega_k) + W_r(j\omega_k)XP_r(j\omega_k) < a_2(W_r(j\omega_k)XP_r(j\omega_k) - W_i(j\omega_k)XP_i(j\omega_k)) + b_2, \quad \omega_1 \leq \omega_k \leq \omega_2 \quad (32)$$

$$W_i(j\omega_k)XP_i(j\omega_k) + W_r(j\omega_k)XP_r(j\omega_k) < a_3(W_r(j\omega_k)XP_r(j\omega_k) - W_i(j\omega_k)XP_i(j\omega_k)) + b_3, \quad \omega_2 \leq \omega_k \leq \infty \quad (33)$$

$$-\epsilon < W_i(j\omega_{gml})XP_i(j\omega_{gml}) + W_r(j\omega_{gml})XP_r(j\omega_{gml}) < \epsilon \quad (34)$$

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

$$W_r(j\omega_{gml})XP_r(j\omega_{gml}) - W_i(j\omega_{gml})XP_i(j\omega_{gml}) < -1/GM(35)$$

1.8. Illustrative examples:(For Comparison between two techniques). **The transfer function of a process (unstable) is [8]:**

$$P(s) = \frac{e^{-0.05s}}{s^{2.5} + s^2 - 1}$$

Designing specifications taken: $a_1=b_1=1, a_2=b_2=-0.3, a_3=0.3, b_3=0, \omega_{gml} = 1 \text{ rad/s}, \epsilon = 0.05, 1/GM \text{ (in dB)}, PM=65^\circ, \omega_1 = 10^{-3}, 10^{-2}, 10^{-1} \text{ rad/s}, \omega_2 = 30, \dots, 510 \text{ rad/s}$ (36)

Forming LMI equations from (A),(B),(C),(31),(32),(33),(34), (35) and (36) .

(i). PID Controller tuning: $X_{PID} = [k_p \ k_i \ k_d]^T = [29.5680 \ 40.2836 \ 49.6060]^T, W_{PID}(s) = [1 \ s^{-1} \ s], C_{PID}(s) = 29.5680 + 40.2836/s + 49.6060s$

(ii). FOPID Controller tuning at $\lambda = 0.01$ and $\mu = 1.0069$: $X = [k_p \ k_i \ k_d] = [-1.8406 \times 10^3 \ 1.8739 \times 10^3 \ 39.7474], W_{FOPID} = [1 \ s^{-1} \ s^\mu] = [1 \ s^{-0.01} \ s^{1.0069}], C_{FOPID} = -1.8406 \times 10^3 + 1.8739 \times 10^3/s^{0.01} + 39.7474s^{1.0069}$

Using *Fmincon (Interior Point Optimization) Algorithm*: Forming *Fmincon* from (D), and (36)

(i). PID Controller tuning: $K_p = 20.00, K_i = 40.00, K_d = 50.1437, C_{PID}(s) = 20.00 + 40.00/s + 50.1437s$

(ii). FOPID Controller tuning (taking exponents in case of LMI): $K_p = 70.00, K_i = 29.00, K_d = 30.00, \lambda = 0.01, \mu = 1.0069, C_{FOPID} = 70.00 + 29.00/s^{0.01} + 30.00s^{1.0069}$. Step responses are shown in Fig. 5 and Fig. 6:

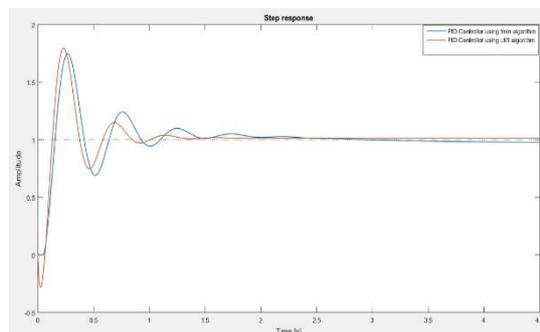


Fig. 5: Step responses of PID Plant tuned by LMI (red) and Fmincon (blue) techniques respectively

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

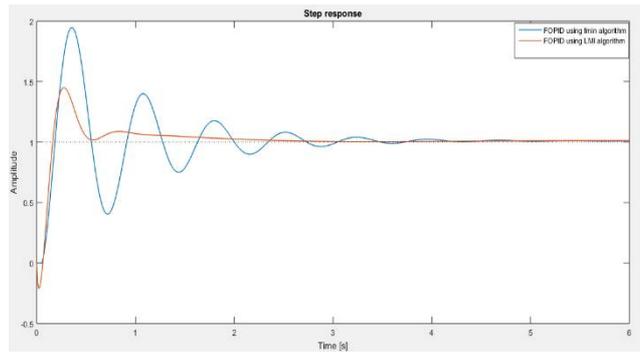


Fig.6: Step responses of FOPID Plant tuned by LMI (red) and Fmincon (blue) techniques respectively

	Control lers	ISE	IAE	Rise time (<i>t_r</i>) sec	Settling time (<i>t_s</i>) sec	O%
F m i n c o n	PID	0.173	0.336	0.053	2.854	69.1
	FOPID	0.297	0.544	0.081	2.127	84.6
L M I	PID	0.162	0.320	0.052	1.18	76.8
	FOPID	0.154	0.331	0.078	1.685	43.1

Table.1: Comparative analysis of LMI tuning and Fmincon tuning algorithm

Conclusion: As shown in the Table 1 the LMI tuned PID and FOPID shows faster time of response and error performance specification is less. This is so because LMI technique is converges faster than Fmincon algorithm technique.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

1.9. CASE STUDY: Temperature Control by changing the valve opening as shown in the Fig.7:

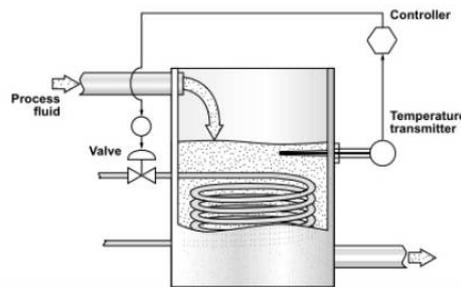


Fig.7 : Closed loop Temperature control**

Process Instrumentation and Control (ICE 401)

** Dr. S.Meenatchisundaram, MIT, Manipal, Jan – May 2015

Identification of raw data as Integer and fractional transfer function from the Table.2:

Y (temp in °C)	U(% of valve opening)
32.496	71.385
32.497	71.583
.....
1423 readings	1423 readings

Table.2: Experimental data of % valve opening versus (Temperature °C)

Step1: Open System identification tool box: Type `ident` in command window (MATLAB) which opens a tool box as shown in Fig.8^[15]:

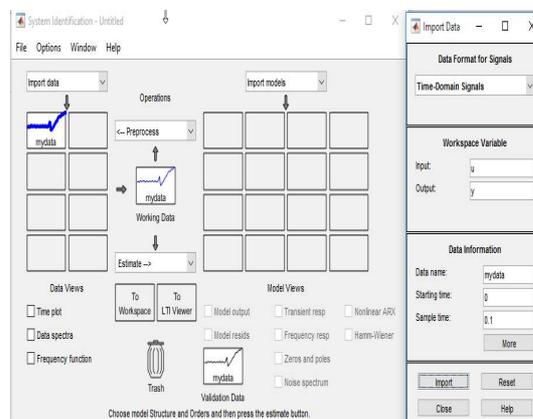


Fig.8: System Identification toolbox for identifying integer transfer function ^[14]

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

2. Select the estimate column >> Process Models. Choosing the selection option as shown in Fig. 9^[14].

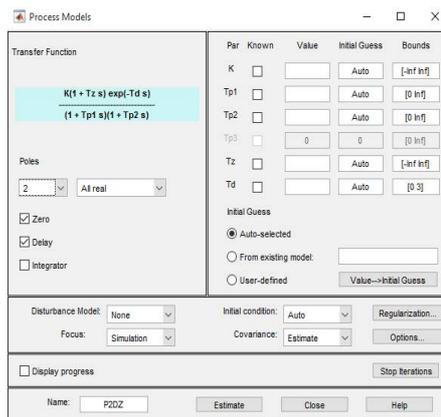


Fig. 9 Process Models dialog box in SI toolbox ^[14]

The Integer order transfer function from the raw input and output data is identified as in (37):

```
>> ITFplant
ITF =
exp(-3*s) * (1.546e05 s + 47226) / (1.066e08 s^2 + 1.964e07 s + 1)
```

(37)

(b). Identification of fractional order transfer function:

Step1: Opening FOTF Time domain Identification tool: Type `fotfid` which opens a tool box as shown in Fig.10^[5]:

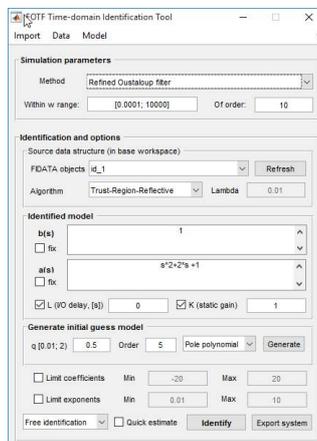


Fig.10: FOTF identification tool box^[5]

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

Step2: Click on Import option choose the options of identification method and guessed model type to get (38):

```
>> FOTFplant
Fractional-order transfer function:
          1799.4
----- exp(-7.5735e-05s)
4723.3s^{2.7896}+28992s^{0.87573}+1
```

(38)

Tuning conventional PID and FOPID for Integer order transfer function: Design specifications used $\omega_{gc} = 5\text{rad/s}$, $PM=65^\circ$; $1/GM = 20\text{ dB}$, $\omega_{ph}=3\text{rad/s}$. (39)

For PID Controller we take fmincon algorithm from (D) and (39) to tune and we get the following controller: $C_{PID} = WX = [1\text{ s}^{-1}\text{ s}][k_p\ k_i\ k_d] = [1\text{ s}^{-1}\text{ s}][5.000\ 9.1340\ 9.5156]$

For FOPID Controller we take fmincon algorithm from from (D) and (39) to tune and we get the following controller: $C_{FOPID} = WX = [1\text{ s}^{-\lambda}\text{ s}^{\mu}][k_p\ k_i\ k_d] = [1\text{ s}^{-0.2000}\text{ s}^{0.2001}][4.9987\ 9.4762\ 9.9988]$. The step responses of PID and FO-PID controlled plants are shown in Fig.11. Table.3 describes the performance characteristics.

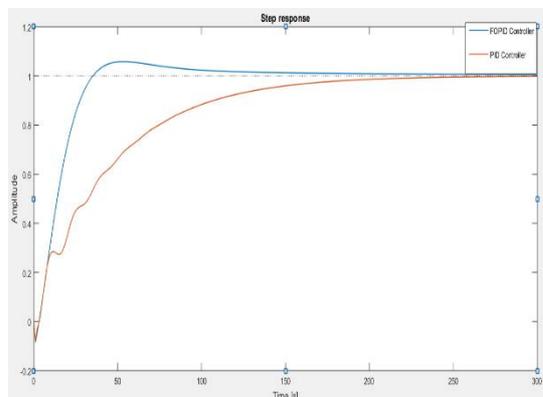


Fig.11:Step responses of the closed loop plants with FOPID (blue) and PID (red) controllers

Controller	Rise Time (t_r) seconds	Settling Time (t_s) seconds	O%
PID	101.808	175.295	0.000
FOPID	22.237	92.854	5.244

Table.3: Performance specifications of closed loop plant

Tuning conventional PID and FOPID for Fractional Order transfer function: Design specifications used $\omega_{gc} = 5\text{rad/s}$, $PM=65^\circ$; $1/GM = 20\text{ db}$, $\omega_{ph}=3\text{rad/s}$. (40)

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

For PID Controller the fmincon algorithm is taken from (40) to tune to get the following controller: $C_{PID} = WX = [1 \text{ s}^{-1} \text{ s}][k_p \text{ } k_i \text{ } k_d] = [1 \text{ s}^{-1} \text{ s}][4.8246 \text{ } 10.000 \text{ } 1.1358]$

For FOPID Controller fmincon algorithm is taken from (40) and tuned to get the following controller: $C_{FOPID} = WX = [1 \text{ s}^{-\lambda} \text{ s}^{\mu}][k_p \text{ } k_i \text{ } k_d] = [1 \text{ s}^{-0.8913} \text{ s}^{1.3709}][2.8306 \text{ } 9.9912 \text{ } 0.1707]$. The step response of FOPID and PID controlled plants are shown in Fig.12. The detailed performance specification is given in Table.4.

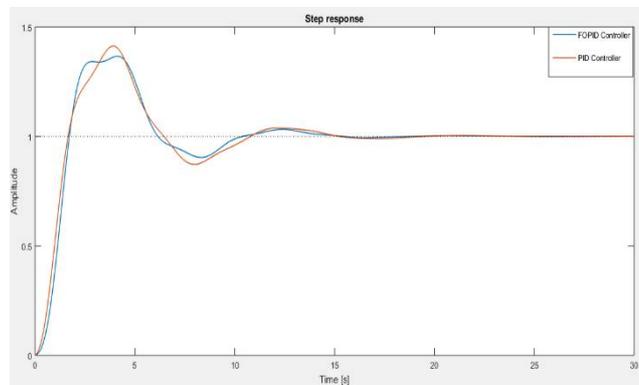


Fig.12: Step responses of FOPID (blue) and PID (blue) controlled plant

Types of Controller	Rise Time (t_r) seconds	Settling Time (t_s) seconds	O%
PID	1.0866	14.1337	41.214
FOPID	1.0512	13.354	35.431

Table.4: Performance specifications of closed loop plant

Conclusion: From Table 3 and Table 4 Fractional Order PID controller (FOPID) always shows better time of response and lesser overshoot.

II. CONCLUSIONS

(1). Tuning techniques: LMI is better as compared to Fmincon as it has faster tendency to converge and more accuracy, than Fmincon (2).FOPIDs have more degrees of freedom more parameters to tune. Stability region is large due to fractional part, hence more flexibility to design controllers. FOPID provides robustness to the system.They track the set point more accurately than their conventional counterpart. A fractional-order controller will always provide better response than integer-controller if it is properly tuned whatever may be the type of plant.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

REFERENCES

- [1]. Poldubny, I. (1999). *Fractional Differential Equations. An Introduction to Fractional Derivatives, Fractional Differential Equations, Some Methods of Their Solution and Some of Their Applications*. Academic Press, San Diego-Boston-New York-London-Tokyo-Toronto, 368p.
- [2]. Miller, K.S., and Ross, B. (1993). *An Introduction to the Fractional Calculus and Fractional Differential Equations*. John Wiley and Sons, Inc., 384p.
- [3]. Chen, Y.Q., Petráš, I., and Xue, D. (2009). *Fractional order control—a tutorial*. In: *Proceedings of the 2009 American Control Conference (ACC'09)*, pp. 1397-1411.
- [4]. Monje, C.A., Chen, Y.Q., Vinagre, B., Xue, D., and Fileu, V. (2010). *Fractional Order Systems and Controls - Fundamentals and Applications*. Advances in Industrial Control Series. 415 p., Springer-Verlag London.
- [5]. Tepljakov, A., Petlenkov, E., and Belikov, J. (2011). "FOMCON : a MATLAB Toolbox for Fractional-order System Identification and Control". *International Journal of Microelectronics and Computer Science*, 2(2), 51-62.
- [6]. Tangirala, A.K. (2014). *Principles of System Identification: Theory and Practice*. CRC Press, Taylor & Francis Group, 908 p.
- [7]. Sandhya, A., Sandhya, R., and Prameela, M. (2016). An overview of Fractional order PID Controllers and its Industrial Applications. *International Journal of Innovations in Engineering and Technology (IJJET)*, 6(4), 534-546.
- [8]. Merrikh-Bayat, F. (2017). A uniform LMI formulation for tuning PID, multi-term fractional-order PID, and Tilt-Integral-Derivative (TID) for integer and fractional-order processes. *ISA Transactions*, 68, 99-108.
- [9]. Boyd, S., El Ghaoui, L., Feron, E. and Balakrishnan V. (1994). "*Linear Matrix Inequalities in System and Control Theory*". *Studies in Applied Mathematics (SIAM) Series*, 15, University City Science Center, Philadelphia, Pennsylvania.
- [10]. Erkus, B., and Lee, Y.J. (2004). "Linear Matrix Inequalities and Matlab LMI Toolbox". *University of Southern California Group Meeting Report*, Los Angeles, California
- [11]. Coleman, T.F., and Li, Y. (1996). An Interior, Trust Region Approach for Nonlinear Minimization subject to Bounds. *SIAM Journal on Optimization*, 6, 418-445.
- [12]. Grace, A.C.W. (1992). "*An Optimization Toolbox for use with MATLAB: User's Guide*", The Math Works Inc.
- [13]. Gahinet, P., Nemirovshi, A., Laub, A., and Chilali, M. (1995). *MATLAB LMI Control Toolbox*. Natick, M.A.: MathWorks.
- [14]. Ljung, L. (2002). "*System Identification Toolbox for Use with MATLAB*", Natick, MA: The MathWorks, Inc., User's Guide Version 5.0.2