PSO Based Optimal Design of Fractional Order Controller for Industrial Application

Rohit Gupta, Ruchika

Abstract—In this paper, a PSO based fractional order PID (FOPID) controller is proposed for concentration control of an isothermal Continuous Stirred Tank Reactor (CSTR) problem. CSTR is used to carry out chemical reactions in industries, which possesses complex nonlinear dynamic characteristics. Particle Swarm Optimization algorithm technique, which is an evolutionary optimization technique based on the movement and intelligence of swarm is proposed for tuning of the controller for this system. Comparisons of proposed controller with conventional and fuzzy based controller illustrate the superiority of proposed PSO-FOPID controller.

Keywords—CSTR, Fractional Order PID Controller, Particle Swarm Optimization.

I. INTRODUCTION

There is an increasing interest in dynamic systems of noninteger orders which are employed in extending the order of derivatives and integrals from integer to noninteger order & has a firm and long standing theoretical foundation.

In 1974, K. B. Oldham and J. Spanier presented theoretical and practical aspects of computing methods for mathematical modeling of nonlinear systems by considering number of computing techniques such as methods of operator approximation, operator interpolation techniques including a non-Lagrange interpolation [1]. The possible advantages of fractional order control in modeling and control design have motivated renewed interest in various applications of fractional order control [2], [3]. For the fractional order dynamic system modeling, control and filtering purpose, MATLAB is used [4]. A fractional order PID controller designing method is proposed by minimizing the integral of the squared errors [5]. Some examples of the fractional order PID control are seen in [6], [7]. In [8], a fractional controller was proposed to ensure that the closed-loop system is robust to gain variations and the step response exhibit an iso-damping property. A comparative introduction of four fractional order controllers can be found in [9]. Many researchers proposed various methods of tuning of fractional order controller like pole distribution of the characteristic equation in the complex plane [10], state space approach based on feedback pole placement [11], method using differential evolution (DE) technique [12] and using optimization method [13]. Some researchers also proposed optimal tuning of classical PID controller with various fitness functions but classical PID controller cannot achieve a high quality solution that effectively improve the transient response of the system. Genetic algorithm and fuzzy logic is also used to optimize the parameters of controller [14]. Some group of researchers also proposed nature inspired optimization techniques to tune the controller, which are Bee Colony algorithms, Ant Colony algorithms and Particle Swarm Optimization technique [15]-[18]. In fractional order controllers, integral and derivative operations are usually of fractional order, therefore besides tuning the proportional ($k_p$), derivative ($k_d$) and integral ($k_i$) constants we have two more parameters: the power of $s$ in integral and derivative actions $\lambda$ and $\mu$ respectively. This adds flexibility and makes the system more robust, thus, enhancing its dynamic performance compared to its integer counterpart. Finding an optimal set of values for $k_p$, $k_d$, $k_i$, $\lambda$ and $\mu$ to meet the specifications of user for a given process plant calls for real parameter optimization in five-dimensional hyperspace [19].

This paper is organized as follows. In Section II, the problem formulation is done, i.e. mathematical modeling of CSTR. Section III discusses a brief overview of fractional calculus especially fractional order PID controller. In Section IV, brief introduction to particle swarm optimization algorithm and proposed fitness function is discussed. In Section V, simulation results are presented and Section VI deals with the conclusion.

II. PROBLEM FORMULATION

Chemical reactors are the most important unit of chemical plants in the industries used for unit operations. Basically, a chemical reactor is a device in which chemical reaction takes place.

Fig. 1 Isothermal Stirred Tank Reactor

Rohit Gupta and Ruchika are with the Department of Electrical and Instrumentation Engineering at Thapar University, Patiala, Punjab, India. (e-mail: rohit.gupta@thapar.edu, ruchika.mehta@thapar.edu)
While designing a chemical reactor following factors have to be considered, (i) Overall size of reactor, (ii) Products emerging from reactor, (iii) Temperature inside the reactor (iv) Pressure inside the reactor (v) Rate of reaction (vi) Activity and mode of catalyst (vii) Stability and controllability of reactor.

Isothermal CSTR is a type of CSTR which is operating at a constant temperature. Here a constant volume, series-parallel reaction is assumed.

Consider an isothermal CSTR shown in Fig. 1. The following reaction scheme consists of the irreversible reactions, which is called Van de Vusse reaction:

\[ \begin{align*}
A & \rightarrow B \rightarrow C \\
2A & \rightarrow D
\end{align*} \]

The feed stream contains only component A which is cyclopetanol, the desired component is B which is cyclopentenol, the intermediate components are cyclopetanediol and dicyclopentadiene denoted by C and D respectively. For the above reaction the values of rate constant are

\[ k_1 = 50h^{-1} = 0.83\text{min}^{-1} \]

\[ k_2 = 100h^{-1} = 1.66\text{min}^{-1} \]

\[ k_3 = 10\text{mol}^{-1}\text{h}^{-1} = 0.166\text{mol}^{-1}\text{min}^{-1} \]

Steady state feed concentration is \( C_{\text{feed}} = 10\text{gmoll}^{-1} \). Overall material balance is given by (1)

\[ \frac{d(V\rho)}{dt} = F_0\rho_0 - F\rho \]  

(1)

Here assumption is made that the liquid phase density \( \rho \) is not a function of concentration \( \rho = \rho_0 \). So,

\[ \frac{dV}{dt} = F_0 - F \]  

(2)

Let \( C_A \) and \( C_B \) represent the molar concentration A and B in moles/volume, component material balance can be written as (3):

\[ \frac{d(VC_A)}{dt} = F(C_{\text{A,feed}} - C_A) - Vk_1C_A - Vk_2C_A^2 \]  

(3)

Simplifying (3) we obtain (4)

\[ \frac{dC_A}{dt} = \frac{F}{V}(C_{\text{A,feed}} - C_A) - r_A \]  

(4)

Similarly, for B, C and D

\[ \frac{dC_B}{dt} = \frac{F}{V}C_A + r_B \]  

(5)

\[ \frac{dC_C}{dt} = \frac{F}{V}C_B + r_C \]  

(6)

\[ \frac{dC_D}{dt} = \frac{F}{V}C_D + r_D \]  

(7)

For each component the molar rate of formation (per unit volume) is given as:

\[ r_A = -k_1C_A - k_2C_A^2 \]  

(8)

\[ r_B = k_1C_A - k_2C_B \]  

(9)

\[ r_C = k_2C_B \]  

(10)

\[ r_D = \frac{1}{2}k_3C_A^3 \]  

(11)

It is noticed that (4) and (5) do not depends on the concentration of components C or D, since for this particular problem component B is our concerned hence only (4) and (5) are considered hence the two dynamic functional equations can be represented as:

\[ \frac{dC_A}{dt} = f_1(C_A, C_B, \frac{F}{V}) = \frac{F}{V}(C_{\text{A,feed}} - C_A) - k_1C_A - k_2C_A^2 \]  

(12)

\[ \frac{dC_B}{dt} = f_2(C_A, C_B, \frac{F}{V}) = -\frac{F}{V}C_A + k_1C_A - k_2C_B \]  

(13)

Using (4) and (5) at steady state, here subscript s is used to indicate the steady state value represented by (12) and (13):

\[ -k_1C_A^s + \left( -k_2 - \frac{F}{V} \right) C_A^s + \frac{F}{V}C_{\text{feed}} = 0 \]  

(12)

\[ -\frac{F}{V}C_A^s + k_1C_A^s - k_2C_B^s = 0 \]  

(13)

Steady state concentration of A and B is defined as (14) and (15) respectively using (12) and (13), only positive roots are considered as there cannot be negative concentrations.

\[ C_A^s = \frac{-k_1 + \frac{F}{V} + \sqrt{\left( k_1 + \frac{F}{V} \right)^2 + 4k_2k_3\frac{F}{V}C_{\text{feed}}}}{2k_3} \]  

(14)

\[ C_B^s = \frac{k_1C_{\text{feed}}}{\frac{F}{V} + k_2} \]  

(15)

The linear state space model is represented as
\[
\dot{x} = Ax + Bu \\
y = Cx + Du
\]
The state variable is represented as
\[
x = \begin{bmatrix} C_A - C_A' \\ C_B - C_B' \end{bmatrix}
\]
The output variable is represented as
\[
y = \begin{bmatrix} C_B - C_B' \end{bmatrix}
\]
The input variable is represented as
\[
u = \begin{bmatrix} F_x \\ F_z \\ V \\ C_A' - C_A' \end{bmatrix}
\]
The elements of state space A matrix are found by
\[
A = \left. \frac{\partial f}{\partial x} \right|_{x,v,v,v}
\]
The elements of state space B matrix are found by
\[
B = \left. \frac{\partial f}{\partial u} \right|_{x,v,v,v}
\]
The state space model is represented as [20].
\[
A = \begin{bmatrix} -\frac{F_z}{V} - k_1 - 2k_2C_d & 0 \\ k_1 & -\frac{F_z}{V} - k_2 \end{bmatrix}
\]
\[
B = \begin{bmatrix} C_A' - C_A' & \frac{F_z}{V} \\ -C_B' & 0 \end{bmatrix}
\]
\[
C = \begin{bmatrix} 0 & 1 \end{bmatrix}
\]
\[
D = \begin{bmatrix} 0 & 0 \end{bmatrix}
\]
Based on steady state operating point
\[
C_d = 3g\text{mol}^{-1}, C_B = 1.117g\text{mol}^{-1}, \\
\frac{F_z}{V} = 0.5714\text{ min}^{-1}
\]
\[
A = \begin{bmatrix} -2.4 & 0 \\ 0.83 & -2.23 \end{bmatrix}
\]
\[
B = \begin{bmatrix} 7 & 0.57 \\ -1.117 & 0 \end{bmatrix}
\]
\[
C = \begin{bmatrix} 0 & 1 \end{bmatrix}
\]
\[
D = \begin{bmatrix} 0 & 0 \end{bmatrix}
\]
Converting the state space model to transfer function
\[
G(s) = C(sI - A)^{-1}B
\]
\[
g_p(s) = \frac{-1.117s + 3.1472}{s^2 + 4.6429s + 5.3821}
\]
Equation (16) represents the process transfer function which is a non-minimum phase type system as there is a right half zero which act as a delay in the system.

Fig. 2 shows the block diagram of CSTR with controller, the primary objective of the control mechanism developed for the isothermal CSTR is that the product concentration should be controlled.

![Fig. 2 Feedback Control Strategy for Isothermal CSTR](image)

III. FRACTIONAL ORDER CONTROLLER

In the past decade, fractional order calculus attracted many researchers and also the use of fractional calculus in control system widely increased as well as accepted. For a control loop perspective, four situations may arise:
1. Integer order plant with integer order controller.
2. Integer order plant with fractional order controller.
3. Fractional order plant with integer order controller.
4. Fractional order plant with fractional order controller.

One of the primary controllers is PID controller, which is widely used. Fractional controller is denoted by $P^\lambda D^\mu$ was proposed by Igor Podlubny in 1997 [21], here $\lambda$ and $\mu$ have non-integer values. Fig. 3 shows the block diagram of fractional order PID controller.
Fig. 3 Fractional order PID controller

Equation (17) represents the transfer function for conventional PID controller:

$$ G_{PID}(s) = \frac{U(s)}{E(s)} = \left( K_p + \frac{1}{T_i s} + T_d s \right) $$

(17)

Transfer function for fractional order PID controller can be written as (18):

$$ G_{FOPID}(s) = \frac{U(s)}{E(s)} = \left( K_p + \frac{1}{T_i s^\lambda} + T_d s^\mu \right) $$

(18)

where $\lambda$ and $\mu$ are arbitrary real numbers, $K_p$ is amplification (gain), $T_i$ is integral constant and $T_d$ is differentiation constant. Taking $\lambda=1$ and $\mu=1$, a classical PID controller is obtained.

The $\mathcal{P}^1\mathcal{D}^\mu$ controller is more flexible and gives an opportunity to better adjustment of the dynamics of control system. It is compact and simple but the analog realization of fractional order system is very difficult [22].

IV. PARTIAL SWARM OPTIMIZATION (PSO)

In this paper, Particle Swarm Optimization (PSO) technique is used. PSO algorithm adopts a strategy based on particle swarm and parallel global random search. PSO algorithm determines search path according to the velocity and current position of particle without more complicated evolution operation. It has better performance than early intelligent algorithms on calculation speed and memory occupation, and has less parameters and is easier to realize [23].

Particle Swarm has two primary operators: Velocity update and Position update. During each generation, each particle is accelerated toward the particles previous best position and the global best position. At each iteration a new velocity value for each particle is calculated based on its current velocity; the distance from its previous best position, and the distance from the global best position. The new velocity value is then used to calculate the next position of the particle in the search space. This process is then iterated a set number of times or until a minimum error is achieved [24].

Let in PSO, a swarm has N number of particles moving around in a D dimensional search space. The ith particle denoted as (19).

$$ X_i = (x_{i1}, x_{i2}, \ldots, x_{iD}) $$

(19)

whose previous best solution ($P_{best}$) is represented as (20):

$$ P_i = (p_{i1}, p_{i2}, \ldots, p_{iD}) $$

(20)

and the current velocity is described by (21):

$$ V_i = (v_{i1}, v_{i2}, \ldots, v_{iD}) $$

(21)

Finally, the best solution of whole swarm ($g_{best}$) also called global best is represented as (22):

$$ P_g = (p_{g1}, p_{g2}, \ldots, p_{gD}) $$

(22)

At each time step, each particle moves towards ($P_{best}$) and ($g_{best}$) locations. The fitness function evaluates the performance of particles to determine whether the best fitting solution is achieved [24]. The detailed operation of particle swarm optimization is given as

Step 1: Initialization The velocity and position of all particles are randomly set within pre-defined ranges.

Step 2: Fitness Function Evaluate the fitness of each particle of the swarm.

Step 3: Velocity Updating: At each iteration, the velocities of all particles are updated according to (23):

$$ v_{ia} = w \cdot v_{ia} + c_1 \cdot rand() \cdot (p_{ia} - x_{ia}) + c_2 \cdot rand() \cdot (g_{ia} - x_{ia}) $$

(23)

where $w$ is inertia factor which balances the global wide-range exploitation and the local nearby exploration abilities of the swarm given by (24). $c_1$ and $c_2$ are two positive constants, called cognitive learning rate and social learning rate respectively; rand() is a random function in the range [0,1].

$$ w = \frac{w_{max} - w_{min}}{iter_{max} - iter_{max}} \cdot iter_{max} + w_{min} $$

(24)

Step 4: Position Updating Assuming a unit time interval between successive iterations, the positions of all particles are updated according to (25):

$$ x_{ia}(t+1) = x_{ia}(t) + v_{ia}(t+1) $$

(25)

After updating, position should be checked and limited to the allowed range.

Step 5: Memory Updating Update $P_{best}$ and $g_{best}$ as per (26) and (27):

$$ P_{best} = p_i \quad \text{if} \quad f(p_i) < f(P_{best}) $$

(26)

$$ g_{best} = p_g \quad \text{if} \quad f(p_g) < f(g_{best}) $$

(27)

where $f(x)$ is the objective function subject to minimize.
Step 6: **Stop Criteria** The algorithm repeats from Steps 2 to 5 until certain stop conditions are met, such as a predefined number of iterations or a failure to make progress for a certain number of iterations. Once terminated, the algorithm reports the values of $P_{best}$ and $g_{best}$ as its solution [24].

The performance of the combination of FOPID parameters is determined by a fitness function. For CSTR, which is non-minimum phase type problem four components are considered to define fitness function. There are steady state error, peak overshoot, rise time and settling time. However, the contribution of these component functions towards the original fitness function is determined by a scale factor that depends upon the choice of the designer. For this particular problem the best point is the point where the fitness function has the minimum value. The chosen fitness function is given as (29):

$$F = 1 - e^{-\beta(M_p + e_{ss})} - e^{-\beta(T_r - T_s)}$$  

(29)

where $F$: Fitness function; $M_p$: Peak Overshoot; $T_s$: Settling Time; $T_r$: Rise Time; $e_{ss}$: Steady State Error; $\beta$: Scaling Factor (Depends upon the choice of designer) For our case of design we have taken the scaling factor $\beta = 1$.

V. **Simulation Results**

This section deals with the Simulink models used in this particular problem and simulation results obtained from those models. Figs. 5-7 show Simulink models of process with conventional PID controller, with fuzzy controller and with fractional order controller respectively. In all the Simulink models the disturbance is considered at the output. The transfer function of CSTR is given in (16).
For tuning of conventional PID controller, Ziegler Nicholas method is used. The values of $K_p$, $K_d$ and $K_i$ obtained from it are 0.56, 0.012 and 3.02 respectively. The fuzzy controller is implemented using FIS toolbox in MATLAB with two inputs and one output variable having seven membership functions each. PSO algorithm is used to tune the FOPID controller.

Table I gives a comparative analysis for different controllers designed to control the product concentration of CSTR. It can be seen that the FOPID controller shows minimum overshoot and minimum negative peak; the fuzzy based controller shows infinite settling time as its output oscillates and never get settled.

<table>
<thead>
<tr>
<th>Parameters/Controller</th>
<th>Rise Time (Sec.)</th>
<th>Peak Time (Sec.)</th>
<th>Settling Time (Sec.)</th>
<th>Peak Overshoot (%)</th>
<th>--ve Peak (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional PID</td>
<td>2.44</td>
<td>3.202</td>
<td>18.74</td>
<td>68</td>
<td>11</td>
</tr>
<tr>
<td>Fuzzy Controller</td>
<td>3.94</td>
<td>5.41</td>
<td>Inf</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td>FOPID Controller</td>
<td>7.92</td>
<td>7.92</td>
<td>7.92</td>
<td>00</td>
<td>03</td>
</tr>
</tbody>
</table>

![Simulink model with FOPID controller](image)

Fig. 8 shows the comparative step response for all three types of controllers, which are conventional PID, fuzzy based and FOPID. The oscillations and highest negative peak in output of fuzzy based controller is clearly visible. The negative peak in the step response is because of RHP zero of the system which makes it a non-minimum phase type system.

Table II compares the various performance indices for the different type of controllers. Among all the performance indices ITAE is the main concerned performance index, as it tell about the speed and accuracy of the response. In terms of performance indices, FOPID controller performs fairly well.

<table>
<thead>
<tr>
<th>Error/Controller</th>
<th>IE</th>
<th>IAE</th>
<th>ISE</th>
<th>ITAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional PID</td>
<td>0.566</td>
<td>3.682</td>
<td>1.931</td>
<td>19.590</td>
</tr>
<tr>
<td>Fuzzy Controller</td>
<td>1.324</td>
<td>2.657</td>
<td>1.280</td>
<td>25.300</td>
</tr>
<tr>
<td>FOPID Controller</td>
<td>2.348</td>
<td>1.136</td>
<td>1.010</td>
<td>1.0460</td>
</tr>
</tbody>
</table>

![Step responses of various controllers](image)

Table III shows the parameters of fractional order PID controller with respect to number of iterations taken by the PSO algorithm which is used to optimize the parameters of the controller. The time taken by the PSO algorithm is also mentioned in seconds with respect to number of iterations. This time might vary with control problem for which it is used and the PSO parameters explained in section IV. It is on higher side for his CSTR problem due to the non-minimum phase system and use of fractional order controller.

<table>
<thead>
<tr>
<th>Parameters/ No. Iterations</th>
<th>$K_p$</th>
<th>$K_d$</th>
<th>$K_i$</th>
<th>$\lambda$</th>
<th>$M$</th>
<th>Run time in sec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.0500</td>
<td>0.0502</td>
<td>0.980</td>
<td>0.3000</td>
<td>0.8600</td>
<td>18.39</td>
</tr>
<tr>
<td>40</td>
<td>0.2510</td>
<td>0.0245</td>
<td>0.499</td>
<td>0.5968</td>
<td>0.0706</td>
<td>34.59</td>
</tr>
<tr>
<td>60</td>
<td>0.0027</td>
<td>0.3088</td>
<td>0.980</td>
<td>0.4006</td>
<td>0.6101</td>
<td>53.37</td>
</tr>
<tr>
<td>80</td>
<td>1.1930</td>
<td>0.0545</td>
<td>0.334</td>
<td>0.6046</td>
<td>0.4710</td>
<td>60.62</td>
</tr>
<tr>
<td>100</td>
<td>0.2000</td>
<td>0.0600</td>
<td>0.800</td>
<td>0.8900</td>
<td>0.7500</td>
<td>70.92</td>
</tr>
</tbody>
</table>

![Value of fitness function](image)

Fig. 9 shows the value of fitness function defined by (29) with each iteration. As seen from Fig. 9, there is no change in the value of fitness function after 100 iterations. So 100
iterations is considered as maximum iteration for this problem and the best solution is given by the set of variables for which the fitness function will be minimum.

![Fig. 9 Value of fitness function w.r.to no. of iteration](image)

![Fig. 10 Step responses of FOPID for different no. of iterations](image)

The time domain parameters with respect to number of iterations are tabulated in Table IV, through which it can be concluded that for 100th iteration the overall performance of the system improves in terms of settling time, peak overshoot, and negative peak, which also reflects in Fig. 10.

**TABLE IV**

<table>
<thead>
<tr>
<th>Parameters/No. Iterations</th>
<th>Rise Time (Sec.)</th>
<th>Peak Time (Sec.)</th>
<th>Settling Time (Sec.)</th>
<th>Peak (Sec.)</th>
<th>Overshoot (%)</th>
<th>-ve Peak (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>3.85</td>
<td>5.07</td>
<td>13.0</td>
<td>0.50</td>
<td>2.0</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>3.65</td>
<td>4.76</td>
<td>14.0</td>
<td>4.0</td>
<td>8.0</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>3.16</td>
<td>3.81</td>
<td>9.0</td>
<td>5.0</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>4.54</td>
<td>5.71</td>
<td>9.76</td>
<td>0.0</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>7.92</td>
<td>7.92</td>
<td>7.92</td>
<td>0.0</td>
<td>3.0</td>
<td></td>
</tr>
</tbody>
</table>

![Fig. 10 shows the step response of fractional order PID controller for different number of iterations. It shows that as the number of iterations increases, the overshoot, negative peak and oscillations in output decreases.](image)

Table V compares the various performance indices of the step response with respect to number of iterations. The best results are obtained for 100 iterations which stands in support with Figs. 8 and 9.

**TABLE V**

<table>
<thead>
<tr>
<th>Errors/No. Iterations</th>
<th>IE</th>
<th>IAE</th>
<th>ISE</th>
<th>ITAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>1.211</td>
<td>3.359</td>
<td>1.937</td>
<td>15.06</td>
</tr>
<tr>
<td>40</td>
<td>1.247</td>
<td>1.787</td>
<td>1.217</td>
<td>5.902</td>
</tr>
<tr>
<td>60</td>
<td>1.487</td>
<td>1.226</td>
<td>1.156</td>
<td>5.740</td>
</tr>
<tr>
<td>80</td>
<td>1.167</td>
<td>1.289</td>
<td>1.114</td>
<td>2.528</td>
</tr>
<tr>
<td>100</td>
<td>2.348</td>
<td>1.137</td>
<td>1.010</td>
<td>1.046</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

This paper dealt with the analysis of CSTR, which is a typical non-minimum phase type nonlinear system with lumped parameters, having swarm intelligent based optimized fractional order PID controller. A comparative study is made with conventional PID controller and fuzzy controller. The comparison of controllers is done on the basis of time domain parameters and performance indices of the step response of CSTR model.

From the results explained in previous section, it is seen that the fractional order controller perform well in comparison to other controllers. The optimized PSO based tuning method is found most suitable for fractional order controller.

REFERENCES


