Fractional Order Automatic Generation Controller For A Multi Area Interconnected System Using Evolutionary Algorithms

Ravi Kumar. Chekka  
Department of Electrical & Electronics Engineering, University College of Engineering & Technology, Acharya Nagarjuna University, Guntur, Andhra Pradesh, India.

Prof. P. V. Ramana Rao  
Department of Electrical & Electronics Engineering, University College of Engineering & Technology Acharya Nagarjuna University, Andhra Pradesh, Guntur, India.

Abstract
This paper presents the implementation of Fractional Order PID controller as a supplementary controller to improve the dynamics of Load Frequency Control in an interconnected Power System. FOPID is a PID controller whose integral and derivative orders are of non-integer rather than integer. The extension of integration and derivative order from integer to fractional order provides more flexibility in the design of the controller there by improving the dynamics of the system. Using integral square error as performance criteria to be optimized, the FOPID controller parameters are optimized using Cat Swarm Optimization (CSO). The performance obtained with CSO is compared with PSO. The simulation results demonstrate that CSO has much better performance than PSO. Simulations have been performed using MATLAB/Simulink on a Three area interconnected Power System.

Keywords: Integer Order PID controller (IOPID), Fractional Order PID Controller (FOPID); Particle Swarm Optimization (PSO), Cat Swarm Optimization (CSO), Automatic Generation Control (AGC), Integral square error (ISE).

List of Symbols
\[ \Delta f \] Deviation in frequency from nominal value  
\[ \Delta P_{12} \] Deviation in power flow of the tie-line connecting areas 1 and 2 from nominal value  
\[ \Delta P_{23} \] Deviation in power flow of the tie-line connecting areas 2 and 3 from nominal value  
\[ \Delta P_L \] Step load change  
\[ \Delta P_g \] Change in the active power output of generator  
\[ \Delta v \] Change in speed  
\[ \Delta P_{ref} \] Frequency Set-point (reference) signal  
\[ T_g \] Time constant of governor  
\[ T_t \] Time constant of turbine  
\[ H \] Generator inertia constant  
\[ D \] Frequency-sensitive load damping coefficient  
\[ B \] Frequency bias coefficient  
\[ R \] Speed regulation  
\[ K_p \] Proportional gain constant of controller  
\[ K_i \] Integral gain constant of controller  
\[ K_d \] Derivative gain constant of controller

Introduction
Load frequency control is a very important issue in power system operation and control for supplying sufficient and reliable electric power with good quality. The main goal of the LFC is to maintain zero steady state error for frequency deviation and tie line power deviations in a multi-area interconnected system [1-3]. The classical PID controller is the most widely used controller for industrial applications due to its simplicity in realization and tuning [4-6]. In recent years, the extension of integration and derivative order from integer to fractional order provides more flexibility in the design of the controller, there by controlling the wide range of dynamics of a system. Fractional calculus is a mathematical analysis which studies the ability of taking real number power of the differential operator and integration operator. It is an extension of the concept \( d^m y(t)/dt^m \) where \( m \) is an integer number to the concept \( d^n y(t)/dt^n \) where \( n \) is non-integer number with possibility to be complex [7]. The traditional IOPID controllers are certain cases of FOPID controllers. As the FOPID has two more extra fine-tuning knobs than the traditional IOPID controller, it provides more flexibility for the design of a control system and gives better opportunity to fine-tune the system dynamics [8-10].

In this paper a fractional PI^a D^b controller is designed for AGC of a three area power system. The parameters \( K_p, K_i, K_d, \lambda, \mu \) and \( B \) (frequency bias coefficient) are optimized using CSO. Simulation results showed that CSO based FOPID controller has better performance than PSO based FOPID controller.

Configuration of Three area Interconnected system
The three-area interconnected power system shown in Fig. 1 is taken as a test system in this study [30]. The conventional AGC has two control loops. The primary control refers to control action that are done locally (on the power plant level) based on the set points for frequency and power. The actual values of these can be measured locally, and deviations from the set values results in a signal that will influence the valves, gates, servos, etc in a primary-controlled power plant, such that the desired active power output is derived. In the primary frequency control, the control task of priority is to bring the frequency back to acceptable values. However, there remains an unavoidable frequency control error because the control...
law is purely proportional. The control task is shared by all generators participating in the primary frequency control irrespective of the location of the disturbance. 

In the secondary frequency control, also called load frequency control, the power set points of the generators are adjusted in order to compensate for the remaining frequency error after the primary control has acted. Apart from that another undesired effect has to be compensated by secondary control. Active power imbalances and primary control actions cause changes in the power flows in the tie lines to other areas, i.e. power exchanges not according to the scheduled transfers. The secondary control ensures by a special mechanism that this is remedied after a short period of time. Hence the control task is to reduce the deviations in system frequency and tie-line power in the three areas in case of any load disturbance in any area. This is attained conventionally with the help of a proper integral control action. The supplementary control of the p\textsuperscript{th} area with integral gain K\textsubscript{i} is therefore made to act on ACE\textsubscript{p} given by (1) which is an input signal to the controller.

\[
ACEp = \sum_{q=1}^{n} \Delta P_{\text{tie, pq}} + B_p\Delta P_f
\]  

Where

- \text{ACE}\textsubscript{p} is area control error of the p\textsuperscript{th} area
- \Delta P_f = Frequency error of p\textsuperscript{th} area
- \Delta P_{\text{tie, pq}} = Tie-line power flow error between p\textsuperscript{th} and q\textsuperscript{th} area
- B_p = frequency bias coefficient of p\textsuperscript{th} area

**Fractional Calculus**

Fractional calculus is a generalization of integration and differentiation to non-integer order fundamental operator \(D^\alpha\), where \(\alpha\) and \(t\) denote the limits of operation. The continuous integro-differential operator is defined as

\[
aD_t^\alpha f(t) = \begin{cases} 
\frac{d^\alpha}{dt^\alpha} f(t) & \text{if } R(\alpha) > 0 \\
1 & \text{if } R(\alpha) = 0 \\
\frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} f(s) \, ds & \text{if } R(\alpha) < 0
\end{cases}
\]  

Where \(\alpha\) is the order of the operation, generally \(\alpha \in \mathbb{R}\), but \(\alpha\) could also be a complex number.

**Definition of Fractional Differintegral**:

There are several definitions of fractional derivatives, Grunwald-Letnikov (GL) definition and Reiman-Liouville (RL) definition as shown in (3) & (4).

\[
aD_t^\alpha f(t) = \lim_{h \to 0} h^{-\alpha} \left[ \sum_{j=0}^{\infty} (-1)^j \binom{\alpha}{j} f(t-jh) \right], \quad (3)
\]

\[
aD_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \int_0^t \frac{f(\tau)}{(t-\tau)^{n-\alpha}} \, d\tau, \quad (4)
\]

For \(n-1 < \alpha < n\) and \(\Gamma(\cdot)\) is the Gamma function [7]. The Laplace transform method is routinely used for engineering problems. The formula for the Laplace transform of the RL fractional derivative (4) has the following form

\[
e^{-st} \left. D_t^\alpha f(t) \right|_{t=0} = s^\alpha F(s) - \sum_{k=0}^{n-1} s^k \int_0^t (t-\tau)^{n-k-1} f(\tau) \, d\tau, \quad (5)
\]

For \(n-1 < \alpha \leq n\), where \(s = j\omega\) denotes the Laplace transform variable.

**Properties of Fractional Calculus**

The main properties of fractional derivatives and integrals are the following:

1. If \(f(t)\) is an analytical function of \(t\), its fractional derivative \(0D_t^\alpha f(t)\) is an analytic function of \(z\) and \(\alpha\).
2. For \(\alpha = n\), where \(n\) is an integer, the operation \(0D_t^\alpha f(t)\) gives the same result as classical differentiation of integer order \(n\).
3. For \(\alpha = 0\) the operation \(0D_t^\alpha f(t)\) is identity operator.
4. Fractional differentiation and fractional integrations are linear operations.
5. The additive index law (semi group property) holds under some reasonable constraints on the function \(f(t)\).

The fractional order derivative commutes with integer order derivative

\[
\frac{d^n}{dt^n} aD_t^\alpha f(t) = aD_t^\alpha \left( \frac{d^n}{dt^n} f(t) \right), \quad \text{under the condition } t=\alpha \text{ we have } f^k(a) = 0, (k=0, 1, 2... n-I).
\]

The relationship above says the operators \(\frac{d^n}{dt^n}\) and \(aD_t^\alpha\) commute.
Fractional-Order PID Controllers

The differential equation of fractional order controller $\text{PPD}^q$ is described by eqn (6)

$$u(t) = K_p e(t) + K_i \int_0^t D^{-\lambda} e(t) + K_d D^{\mu} e(t)$$

(6)

Where $\lambda$ is the integral order, $\mu$ is the differential order $K_p$, $K_i$, $K_d$ are the parameters of PID controller. The continuous transfer function of FOPID [11] is obtained through Laplace transform, which is given by (7)

$$G_c(s) = \frac{K_p + \frac{K_i}{s^\lambda} + K_d s^\mu}{1}$$

(7)

The classical controllers are particular cases of the FOPID controller.

If $\lambda = \mu = 1$, the classical IOPID controller is obtained.

For $\lambda = \mu = 0$, the P controller is obtained.

And for $\lambda = 1$, $\mu = 0$, the PI controller is obtained.

As the FOPID has two added tuning knobs than the conventional integer-order PID controller, use of fractional controller ($\lambda$ and $\mu$ are non-integer) gives more flexibility for the design of a control system and provides better opportunity to alter system dynamics if the original system to be controlled is a fractional system.

Optimization of FOPID Controller’s parameters

Optimization of FOPID controller’s first needs the selection of optimization goal, and encoding the parameters to be searched. From eqn (6) $K_p$, $K_i$, $K_d$ and $\lambda$, $\mu$ are required to be designed, according to control objectives. Determination of these parameters based on experiences like traditional controller is not possible and so a fast and efficient way to optimize the parameters must be used [11].

CSO is a more recent swarm intelligence-based optimization algorithm developed in 2006. It is developed to solve various problems by mimicking the behaviour of cats. CSO has been proved to have a better performance in finding the global best solutions than other existing optimization algorithms.

Fitness Function

The design of FOPID controller is essentially a multi-dimensional optimization problem. The main objective of controller parameters optimization is to make the control error tend to zero and slither over shoot and faster response. In order to obtain satisfactory transition of the dynamic characteristic, the paper has used Integral squared error (ISE) as performance index for the parameter’s minimum objective function. ISE can be expressed as follows

$$\text{ISE} = \int_0^t (ACE(t))^2 \, dt$$

(8)

where $ACE_p = \sum_{q=1}^{n} \Delta P_{iq,pq} + B_p \Delta f_p$

Particle Swarm Optimization

Particle swarm optimization (PSO) is a kind of swarm intelligence algorithm, mimicking biological predation phenomenon in nature. PSO employs group intelligence, which comes from competition and cooperation between particles of a group, to guide the optimization search, with strong convergence, global optimization and computation-efficiency [11].

In PSO the ‘swarm’ is initialized with a population of arbitrary solutions. Each particle in the swarm is a different possible set of the unknown parameters to be optimized. Indicating a point in the solution space, each particle changes its flying experience and shares social information among particles. The goal is to efficiently search the solution space by swarming the particle towards the best fitting solution encountered in previous iterations with the intent of encountering better solutions through the course of process and eventually converging on a single minimum error solution. The following expressions for the particles velocity and position update are used.

$$v_i = \omega v_i + c_1 r_1 (p_i - x_i) + c_2 r_2 (p_{best} - x_i)$$

(9)

$$x_i = x_i + v_i$$

(10)

Where $\omega$ is the inertia weight factor, $c_1$ and $c_2$ are acceleration constant. $r_1$ and $r_2$ are random numbers between zero and one [12].

Cat Swarm Optimization

Cat Swarm Optimization algorithm (CSO) was first proposed in 2007 by Chu and Tsai to achieve better performance than Particle Swarm Optimization (PSO) [26-28]. The cat corresponding to a particle used in PSO is used as an agent. The behaviours of cats are modelled to solve the optimization problem employing a new learning rule. Cat swarm optimization is a novel algorithm based on two major behaviours of cats, termed as “seeking” and “tracing”. The first step to apply CSO in the optimization problem is to choose how many cats to use. Each cat has its own M-dimensional position, velocities for each dimension, a fitness value indicating the accommodation of the cat to the fitness function and a seeking/tracing flag to identify whether the cat is in seeking or tracing mode. The final solution would be the best position for one of the cats. CSO keeps the best solution until the end of all iterations.

Seeking mode:

This mode is used to model the behaviour of cat in the period in which it is relaxing but observing around and seeking the next position to move. Four important parameters are defined in the seeking mode as Seeking memory pool (SMP), Seeking range of the selected dimensional (SRD), Counts of dimension to change (CDC) and self-position consideration (SPC).

SMP is number of copies of each cat. All the copies of a cat are spread around their original cat based on the SRD range. SRD defines the distance between the original cat and its copies. CDC is the percent of dimension to be changed to the total number of dimensions that describes the location of each cat and SPC is a Boolean variable which resolves whether the point, on which the cat is already standing, can be one of the candidates to move to. The value of SMP cannot be influenced by SPC.

Seeking mode is described below

Step 1: Make j copies of the existing position of Catw, where
$j = \text{SMP}$ if the value of SPC is true. Let $j = (\text{SMP} - 1)$ then retain present position as one of the candidates.

Step 2: Based on CDC percentages, spread all the copied cats by randomly adding or subtracting SRD percentages to or from the present position of original cat.

Step 3: Evaluate the fitness values (FS) of all candidate points. If all FS are not exactly equal, estimate the selecting probability of each candidate point by eqn (11), otherwise set all the selecting probability of each candidate point be 1.

$$P_i = \frac{\left| FS_i - FS_b \right|}{FS_{\text{max}} - FS_{\text{min}}} \text{ where } 0 < i < j \quad (11)$$

If the objective of the fitness function is to find the minimum solution, $FS_b = FS_{\text{max}}$ otherwise $FS_b = FS_{\text{min}}$

Step 5: Arbitrarily pick the point to move to from the candidate points, and replace the position of cat

**Tracing mode:**

Tracing mode is the sub-model for demonstrating the case of the cat in tracing targets. Once a cat goes into tracing mode, it moves according to its individual velocities for each dimension. The tracing mode arises from the rapid movements of a cat. This mode relates to a global search. The tracing mode cat has a velocity and position in the direction of the global best cat, $g_{\text{best}}$. The action of tracing mode can be described as follows.

Step 1: Update the velocities for each dimension $v_i(t+1)$ according to equation (12)

Step 2: Check if the velocities are in the range of maximum velocity. In case the new velocity is over-range, it is set equal to the limit.

Step 3: Update the position of cat according to equation (13).

The velocity and position are updated according to the following equations.

$$v_i(t+1) = \omega \times v_i(t) + \text{acc} \times \text{rand} \times (g_{\text{best}} - P_i(t)) \quad (12)$$

$$P_i(t+1) = P_i(t) + v_i(t) \quad (13)$$

Where $\omega$ the inertia weight, acc is is the acceleration coefficients and rand is a random number uniformly distributed in the range between 0 and 1.

**CSO Algorithm Flow**

The computational procedure of the proposed algorithm is described in detail as follows.

Step 1: Create $N$ cats in the process.

Step 2: Randomly sprinkle the cats into the M-dimensional solution space and randomly give values, which are in-range of the maximum velocity, to the velocities of every cat. According to MR arbitrarily pick number of cats and set them into tracing mode, and the others set into seeking mode.

Step 3: Calculate the fitness value of individual cat by applying the positions of cats into the fitness function, which represents the performance criteria of our goal, and keep the best cat into memory. The position of the best cat ($g_{\text{best}}$) only need to be remembered as it represents the best solution so far.

Step 4: Move the cats according to their flags, if $\text{cat}_k$ is in seeking mode, apply the cat to the seeking mode process else apply it to the tracing mode process.

Step 5: Re-pick number of cats and set them into tracing mode according to MR, then set the other cats into the seeking mode.

Step 6: Check the termination condition, if satisfied, terminate the program, and otherwise repeat step3 to step5.

**Figure 2: Flow chart of Cat Swarm Optimization**

**Design of FOPID controller using CSO**

When the FOPID controller parameters are optimized using CSO, the five parameters of the fractional PID controller and B, the frequency bias coefficient of each area are viewed as a particle [25, 30] i.e $K = [K_p, K_i, K_d, \lambda, \mu, B]$. In order to limit the evaluation value of each individual of the population, feasible ranges are set for each parameter as given in Table 1 and Table 2.

**Table 1: Parameter Settings for CSO**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>Number of cats spread in solution area</td>
<td>10</td>
</tr>
<tr>
<td>SMP</td>
<td>Number of Cats each cat makes in seeking mode</td>
<td>5</td>
</tr>
<tr>
<td>SRD</td>
<td>Range of variation for each dimension in seeking mode</td>
<td>30%</td>
</tr>
<tr>
<td>CDC</td>
<td>Number of dimensions that will change in seeking mode for each copy</td>
<td>100%</td>
</tr>
<tr>
<td>MR</td>
<td>Percentage of cats in tracing mode vs seeking mode</td>
<td>20%</td>
</tr>
<tr>
<td>$f_1$</td>
<td>A random variable used in calculating velocities</td>
<td>$[0, 1]$</td>
</tr>
<tr>
<td>Acc</td>
<td>Acceleration coefficient used in calculating velocities</td>
<td>2</td>
</tr>
</tbody>
</table>
Simulation Results
In this study the optimal parameters of FOPID controllers for a Three area interconnected system obtained using PSO and CSO algorithms are,

Optimal parameters obtained using PSO:
$B_1=17$, $B_2=19$, $B_3=18$, $B_4=18$, $K_p=1.242$, $K_d=1.223$, $K_i=1.5438$, $K_d=1.3652$, $K_i=1.2936$, $K_i=1.5$, $K_d=1.1903$, $K_i=1.2249$, $K_d=1.0548$, $K_{i}=1.3288$, $K_{d}=1.5$, $K_{i}=1.4474$, $K_{d}=0.3554$, $K_{i}=0.1306$.

Optimal parameters obtained using CSO:
$B_1=16$, $B_2=18$, $B_3=17$, $B_4=18$, $K_p=1.48$, $K_d=1.33$, $K_i=1.26$, $K_d=1.4$, $K_d=1.14$, $K_i=1.4$, $K_{d}=1.1$, $K_{p}=1.35$, $K_{i}=1.5$, $K_{d}=1.45$, $K_{i}=0.38$, $K_{d}=0.15$.

In a complex Three area interconnected power system, parameter variations cannot be avoided during the operation. Therefore, robustness of the proposed controllers to the parameters’ changes are also to be verified. Figs 3 & 4 represent the plots of change in system frequency with PSO based and CSO based FOPID controller for 1% step load variations in area2. Figs 5 & 6 demonstrates the effectiveness of the proposed controller against large parametric uncertainties.

Table 2: Controller Parameters and the corresponding value ranges

<table>
<thead>
<tr>
<th>Controller Parameters</th>
<th>Description</th>
<th>Value ranges</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_p$</td>
<td>Proportional gain constant of controller</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>$K_i$</td>
<td>Integral gain constant of controller</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>$K_d$</td>
<td>Derivative gain constant of controller</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>$s$</td>
<td>Fractional order of the integrator</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Fractional order of the differentiator</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>$B_1$, $B_3$</td>
<td>Frequency biasing coefficient of area1, area3</td>
<td>[15, 35]</td>
</tr>
<tr>
<td>$B_2$</td>
<td>Frequency biasing coefficient of area2</td>
<td>[25, 35]</td>
</tr>
</tbody>
</table>

Figure 3. Change in Frequency Deviation with PSO based FOPID Controller for 1% Step load perturbation

Figure 4: Change in Frequency Deviation with CSO based FOPID Controller for 1% Step load perturbation

Figure 5: Change in Frequency Deviation with CSO based FOPID Controller considering (-50%) parameter variations.

Figure 6: Change in Frequency Deviation with CSO based FOPID Controller considering (+50%) parameter variations.

Conclusions
In this paper, an FOPID controller is proposed for LFC in a three-area power system. The parameters of controllers are tuned using PSO and CSO. Simulation results show that the proposed CSO based FOPID controller has faster response and less overshoot. Also the proposed controller gives good performance for wide range of variations in system parameters. Simulation results are compared with the PSO based FOPID controller in the Table 3.
Table 3. Comparative study of Settling time and Peak overshoot

<table>
<thead>
<tr>
<th>Controllers</th>
<th>Settling time (Sec)</th>
<th>Peak overshoot (p.u.) x10⁻⁴</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Δf1</td>
<td>Δf2</td>
</tr>
<tr>
<td>PSO based FOPID controller</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>CSO based FOPID controller</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>CSO based FOPID controller with (-50%) Parameter variations</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>CSO based FOPID controller with (+50%) Parameter variations</td>
<td>7</td>
<td>7</td>
</tr>
</tbody>
</table>

References

[28] John Paul T. Yusung. ‘Optimizing Artificial Neural Networks using Cat Swarm Optimization Algorithm’,

[30] Ch. Ravi Kumar, Dr. P. V. Ramana Rao ‘PSO based fractional order automatic generation controller for three area interconnected system’ International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO) – 2015
