

Design Robust PID Controller for Hydro-turbine governing with ABC Algorithm

A. Yosefi  
Technical Eng. Department  
University of Mohaghegh Ardabili,  
Ardabi, Iran

A. Ghasemi  
Young Researcher Club, Ardabil Branch, Islamic Azad University,  
Ardabil, Iran

R. Bazyar  
East Azarbayjan Power Generation Management Company  
Ardabil, Iran

H. A. Shayyanfar*  
Electrical Eng. Department, Islamic Azad University, South Tehran Branch, Tehran, Iran

O. Abedinia  
Young Researcher Club, Ardabil Branch, Islamic Azad University,  
Ardabil, Iran

H. GholamalitabarFiroozjaee  
Technical Eng. Department  
University of Mohaghegh Ardabili,  
Ardabi, Iran

ali.yousefi1365@gmail.com, ghasemi.agnost@gmail.com, R.Bazyar@yahoo.com, hashayanfar@yahoo.com,  
oveis.abedinia@gmail.com, h.gholamalitabar@yahoo.com

Abstract— In this paper, the parameters of PID controller are tuned by Gravitational Search Algorithm (GSA). The key idea of the proposed method is to use a new design PID controller of Hydro-turbine governing. The result of simulation are shown effectiveness of the proposed method and GSA algorithm for solve Hydro-turbine governing problem in different load condition of power system. The achieved result of GSA algorithm compare with CEP, FEP, MFEP and DCMEP algorithm, the table and figure shown transcendent GSA algorithm for optimizations problem.

Keywords— Hydro-turbine governing, GSA, robust PID controller.

I. INTRODUCTION

Energy is the basic need for economic development; every sector of country’s economy (industry, agricultural, transport, commercial and domestic) needs input of energy. The Hydro-turbine governing systems contain many parts, high dimension complex systems, time-variant and multi-parameters. With attention to increases demand, the generating of energy is near to demand of energy, so need a best controller to guarantee systems for different condition. In most study used ideal PID controller for model of Hydro-turbine governing but in industrial production of ideal PID controller is not possible, so in this paper used a robust PID (RPID) controller for control Hydro turbine governing system with disturbance. PID controller is a best controller for us decide, and use many artificial algorithm for getting best answer for PID controllers (Kp, Ki, Kd) to guarantee system in best condition of working. hithero used Genetic Algorithm (GA) [1], Simulated annealing (SA) algorithm [2], Evolutionary Programming (EP) [3], Conventional Evolutionary Programming (CEP) [4], Fast Evolutionary Programming (FEP) [4,5], Deterministic Chaotic Mutation Evolutionary Programming (DCMEP) [4,6], but the above algorithms cannot best solution for optimization of Hydro-turbine governing, so this paper a new optimization algorithm based on the law of gravity, namely Gravitational Search Algorithm (GSA) for problem solving is proposed [8]. This algorithm is based on the Newtonian gravity: “every particle in the universe attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them”. In Sect. II, the proposed GSA algorithm is described, in Sect. IV experiments and results are presented.

Nomenclature:

N: population size  
Tg: inertial time constant of generator  
Te: current inertial time constant  
Tζ: adjust time in seconds  
Δ: overshoot level  
Tγ: Engagers relay time constant  
eγ: hydro-turbine torque vine opening level transfer coefficient  
eq: hydro-turbine torque pressure transfer coefficient  
eφ: hydro-turbine hydraulic flux vine opening level transfer coefficient  
eφ: hydro-turbine hydraulic flux pressure transfer coefficient  
Kp: proportional adjustment coefficient  
Ki: integral adjustment coefficient  
Kd: deferential adjustment coefficient  
e: generator’s self adjustment coefficient

*Corresponding Author. E-Mail Address: hashtyanfar@yahoo.com (H. A. Shayyanfar)
II. GRAVITATIONAL SEARCH ALGORITHM (GSA)

The gravitational search algorithm is constructed based on the law of gravity and the notion of mass interactions. The GSA algorithm uses the theory of Newtonian physics and its searcher agents are the collection of masses. In GSA, we have an isolated system of masses. Using the gravitational force, every mass in the system can see the situation of other masses. The gravitational force is therefore a way of transferring information between different masses. GSA was introduced by E. Rashedi et all, 2009. In GSA, agents are considered as objects and their performance is measured by their masses. All these objects attract each other by the gravity force, and this force causes a global movement of all objects towards the objects with heavier masses. Hence, masses cooperate using a direct form of communication, through gravitational force. The heavy masses - which correspond to good solutions - move more slowly than lighter ones, this guarantees the exploitation step of the algorithm [7,8]. In GSA, each mass (agent) has four specifications: position, inertial mass, active gravitational mass, and passive gravitational mass. The position of the mass corresponds to a solution of the problem, and its gravitational and inertial masses are determined using a fitness function. In other words, each mass presents a solution, and the algorithm is navigated by properly adjusting the gravitational and inertial mass of each mass. By lapse of time, we expect that masses be attracted by the heaviest mass. This mass will present an optimum solution in the search space. The GSA could be considered as an isolated system of masses. It is like a small artificial world of masses obeying the Newtonian laws of gravitation and motion. More precisely, masses obey the following laws: Law of gravitation: each particle attracts every other particle and the gravitational force between two particles is directly proportional to the product of their masses and inversely proportional to the distance between them, R. Law of motion: the current velocity of any mass is equal to the sum of the fraction of its previous velocity and the variation in the velocity. Variation in the velocity or acceleration of any mass is equal to the force acted on the system divided by the mass of the object [8, 9].

In the GSA algorithm particle researcher, is sum of all mass. We define the position of the \textit{i}th agent by:

\[ X_i = (x_{1,i}, x_{2,i}, \ldots, x_{N,i}) \text{ for } i = 1, 2, 3, \ldots, N \]  

(1)

In the Eqs.1 \textit{x}_{d,i} is the position of \textit{i}th agent in the \textit{d}th dimension. 

\textit{N} is total of agent. At a particular time \textit{t}, we have define the force acting on mass \textit{i} (from mass \textit{j}) as pursuing:

\[ F_{ij}(t) = G(t) \left( (M_{ai}(t) - M_{aj}(t)) \times (X_{ij}(t) - X_{id}(t)) \right) \]  

(2)

\textit{M}_{ai}(t) is the active gravitational mass related to agent \textit{j}, \textit{M}_{aj}(t) is the passive gravitational mass related to agent \textit{i}, \textit{G} (t) is gravitational constant at time \textit{t}, \textit{ɛ} is a small constant, and \textit{R}_{ij}(t) is the Euclidian distance between two agents \textit{i} and \textit{j}:

\[ R_{ij} = \sqrt{(X_{i}(t) - X_{j}(t))^2} \]  

(3)

To give a chromatic characteristic to GSA algorithm, suppose that the total force that acts on agent \textit{i} in a dimension \textit{d} be a randomly weighted sum of \textit{d}th components of the forces exerted from other agents:

\[ F_{it} = \sum_{j \neq i} \text{rand}_d \times F_{ij} \]  

(4)

Where \text{rand}_d is generate random in the interval [0, 1], by the law of motion, the acceleration of the agent \textit{i} at time \textit{t}, and in direction dth, \textit{a}_{it}(t), give from equation 5 is equal:

\[ a_{it}(t) = \frac{F_{it}}{M_{ai}(t)} \]  

(5)

In the equation (4) \textit{M}_i is the inertial mass of \textit{i}th agent. Velocity for next step will update from equation 5 it is similar to PSO algorithm, because any particle get a new vector of velocity for generate new population. After update velocity vector for agents the position of any agent get from equation 6. The equation for give new velocity and new position following:

\[ v_{it}(t + 1) = \text{rand}_d \times v_{it}(t) + a_{it}(t) \]  

(6)

With attention to Eqs.5, the next velocity of an agent is considered as a fraction of its current velocity added to its acceleration. \text{rand}_d is a uniform random variable in the interval [0, 1]. For give a randomized characteristic to the search used this random number.

The gravitational constant, \textit{G}, is initialized at the beginning and will be reduced with time to control the search accuracy. In other words, \textit{G} is a function of the initial value (\textit{G}0) and time \textit{t}:

\[ G(t) = G(G0, t) \]  

(7)

Gravitational and inertia masses are simply calculated by the fitness evaluation. A heavier mass means a more efficient agent. This means that better agents have higher attractions and walk more slowly. Assuming the equality of the gravitational and inertia mass, the values of masses are calculated using the map of fitness. We update the gravitational and inertial masses by the following Eqs.8-10:

\[ M_w = M_p = M_i, i = 1, 2, \ldots, N \]  

(8)

\[ m_i(t) = \frac{\text{fit}_i(t) - \text{worst}(t)}{\text{best}(t) - \text{worst}(t)} \]  

(9)
\[ M_i(t) = \frac{m_i(t)}{\sum_j m_j(t)} \tag{10} \]

Fitness value of the agent \( i \) at time \( t \) shown with \( \text{fit}_i(t) \) and, worst \( (t) \) and best \( (t) \) are defined as follows (for a minimization problem):

\[ \text{best}(t) = \min_{j \in \{1,2,...,N\}} \text{fit}_j(t) \tag{11} \]

\[ \text{worst}(t) = \max_{j \in \{1,2,...,N\}} \text{fit}_j(t) \tag{12} \]

It is to be noted that for a maximization problem, Eqs. (11) and (12) are changed to Eqs. (13) and (14), respectively:

\[ \text{best}(t) = \max_{j \in \{1,2,...,N\}} \text{fit}_j(t) \tag{13} \]

\[ \text{worst}(t) = \min_{j \in \{1,2,...,N\}} \text{fit}_j(t) \tag{14} \]

For getting best performed with desirable compromise between exploration and exploitation, one way is to reduce the number of agents with lapse of time in Eq. (3). For getting that target, suggest set an agent with bigger mass apply their force to the other. However, we should be careful of using this policy because it may reduce the exploration power and increase the exploitation capability. We remind that in order to avoid trapping in a local optimum the algorithm must use the exploration at beginning.

By lapse of iterations, exploration must fade out and exploitation must fade in. To improve the performance of GSA by controlling exploration and exploitation only the \( K_{\text{best}} \) agents will attract the others. \( K_{\text{best}} \) is a function of time, with the initial value \( K0 \) at the beginning and decreasing with time. In such a way, at the beginning, all agents apply the force, and as time passes, \( K_{\text{best}} \) is decreased linearly and at the end there will be just one agent applying force to the others. Therefore, Eq. (3) could be modified as:

\[ F_i^a(t) = \sum_{j \in K_{\text{best}}} \text{rand} \cdot F_i^a(t) \tag{15} \]

Where \( K_{\text{best}} \) is the set of first \( K \) agents with the best fitness value and biggest mass. The principle of GSA is shown in flowchart of Fig. 1. The flowcharts show how the proposed algorithm is efficient some remarks are noted:

III. MODEL FOR PID CONTROLLER IN SYSTEM STUDY

The system structure based on the GSA algorithm shown in Fig. 2 is targeted for optimizing the on line RPID parameters in the hydro-turbine governing system. The gains of the RPID controller are tuned online in terms of the knowledge base and GSA inference, and then, the RPID controller generates the control signal.

![Flowchart of GSA.](image)

![The proposed RPID controller design problem](image)
\[ \text{PID} = k_p + \frac{K_s}{S} + k_d \]  \hspace{1cm} (16)

But in design industrial or robust PID controller use a low filter for delete the noise of high frequencies, so this paper the function of derive represent Eqs.18:

\[ \frac{k_p S}{1 + T_s S} T_s \gg k_p \]  \hspace{1cm} (17)

\begin{align*}
\sum G(s) = \frac{1}{T_s S + e_v} \\
G_1(s) = \frac{1}{T_s S + e_v} \\
G_2(s) = \frac{G_1(s)}{1 + T_s S + e_v} \\
G_3(s) = G_2(s) \\
K_p e(t) \\
K_p \int e(t) dt \\
K_p \frac{d e(t)}{dt} \\
1 + T_s \frac{d e(t)}{dt} \\
G_4(s) \\
G_5(s) \\
Mg \\
\text{outlet}
\end{align*}

Figure 3. The system structure of Hydro-turbine governing system

The transfer function of the hydraulic pressure servo system is:

\[ G_1(S) = \frac{1}{1 + T_s S} \]  \hspace{1cm} (18)

The transfer function of the hydro-turbine system is

\[ G_2(S) = \frac{e_v - e_v e_v - e_v e_v T_s S}{1 + e_v T_s S} \]  \hspace{1cm} (19)

The transfer function of the generator and load is

\[ G_3(S) = \frac{1}{T_s S + e_v} \]  \hspace{1cm} (20)

The transfer function of the PID governing system is

\[ G_4(Z) = K_p + \frac{K_s}{1 - Z^{-1}} + K_d (1 - Z^{-1}) \]  \hspace{1cm} (21)

Its incremental expression is

\[ \Delta U(k) = K_p (e(k) - e(k - 1)) + K_v e(k) + K_p (e(k) - 2e(k - 1) + e(k - 2)) \]  \hspace{1cm} (22)

where \( U(k) \) is the governor output and \( e(k) \) is the frequency error of the kth sampling. The GSA algorithm optimizes the three PID parameters to improve the static and dynamic performances of the governed object.

IV. RESULT OF RPID DESIGN USING GSA ALGORITHM

The proposed method was applied for RPID design for hydropower station in two scenarios pursues and any scenarios contain two cases. For show proficiency of GSA algorithm in solve intricate problem with many parameters, compare the result of GSA algorithm with CEP, FEP, MFEP and DCEMP [4,5]. The object function for optimization is:

\[ \text{Min } J = \frac{1}{4} e^T e \]  \hspace{1cm} (23)

s. t.

\[ K_{P_{\text{min}}} \leq K_p \leq K_{P_{\text{max}}} \]
\[ K_{I_{\text{min}}} \leq K_i \leq K_{I_{\text{max}}} \]
\[ K_{D_{\text{min}}} \leq K_d \leq K_{D_{\text{max}}} \]

\[ \text{min } j = e^T e \]

Where \( K_{P_{\text{min}}}, K_{I_{\text{min}}} \text{ and } K_{D_{\text{min}}} \) and \( K_{P_{\text{max}}}, K_{I_{\text{max}}} \text{ and } K_{D_{\text{max}}} \) are the upper and lower bounds of \( K_p, K_i, K_d \), respectively.

Scenario 1: the model of Hydro-turbine is HL638-WJ-60, winding speed \( n = 1000 \text{ r/min}, \) power \( P_t = 1612 \text{ kW}, \) pipeline length \( L = 1956 \text{ m}, \) cross area \( 4.22 \text{ m}^2, \) inertial time constant \( T_i = 3.9 \text{ s}, \) current inertial time constant \( T_u = 0.365 \text{ s}. \) The result simulation from GSA, DCEMP, FEP, MFEP and CEP of scenario 1 is presented in Table I.

Case 1: for vane opening level = 60\%, the transfer coefficients are:
\[ e_v = -0.728, \ e_\gamma = 1.28, \ e_h = 0.95, \ e_q = -0.075, \ e_q = 0.956, \ e_\phi = 0.618 \]

Case 2: for vane opening level = 80\%, the transfer coefficients are:
\[ e_v = -0.860, \ e_\gamma = 0.948, \ e_h = 1.31, \ e_q = -0.294, \ e_q = 0.868, \ e_\phi = 0.830 \]

<table>
<thead>
<tr>
<th>Method</th>
<th>Vane Opening (%)</th>
<th>Optimization Parameters</th>
<th>( T_d(s) )</th>
<th>( \delta (%) )</th>
<th>( N )</th>
</tr>
</thead>
<tbody>
<tr>
<td>CEP</td>
<td>60</td>
<td>2.6 0.25 1.0</td>
<td>5.4</td>
<td>1.9</td>
<td>118</td>
</tr>
<tr>
<td>80</td>
<td>4.0 0.23 0.2</td>
<td>6.0 1.77 97</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FEP</td>
<td>60</td>
<td>2.7 0.23 1.5</td>
<td>5.2</td>
<td>1.85</td>
<td>101</td>
</tr>
<tr>
<td>80</td>
<td>4.0 0.22 0.2</td>
<td>5.8 1.74 86</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MFEP</td>
<td>60</td>
<td>3.0 0.25 1.8</td>
<td>4.8</td>
<td>1.35</td>
<td>57</td>
</tr>
<tr>
<td>80</td>
<td>4.0 0.21 0.2</td>
<td>5.2 1.74 85</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCEMP</td>
<td>60</td>
<td>3.0 0.25 1.8</td>
<td>4.8</td>
<td>1.35</td>
<td>49</td>
</tr>
<tr>
<td>80</td>
<td>4.0 0.21 0.2</td>
<td>5.2 1.74 85</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSA</td>
<td>80</td>
<td>3.74 0.22 1.22</td>
<td>5.02</td>
<td>1.67</td>
<td>50</td>
</tr>
</tbody>
</table>

TABLE I. THE RESULT SIMULATION FROM GSA, DCEMP, FEP, MFEP AND CEP OF SCENARIO 1
Figure 4. outing of system for all algorithms for vane opening level = 60%, CEP (...), FEP (+++), MFEP (-.-), DCMEP (***), GSA(____)

Figure 5. outing of system for all algorithms for vane opening level = 80%, CEP (...), FEP (+++), MFEP (-.-), DCMEP (***), GSA(____)

Scenario 2: the model of Hydro-turbine is HL220, winding speed \( n = 136.4 \text{ r/min} \), power \( P_T = 75000 \text{ kW} \), pipeline length \( L = 700 \text{ m} \), cross area \( 4.22 \text{ m}^2 \), inertial time constant \( T_a = 9.42 \text{ s} \), current inertial time constant \( T_w = 1.32 \text{ s} \). the simulation result from GSA, DCEMP, FEP, MFEP and CEP of scenario 2 is presented in Table 2.

Case 1: for vane opening level = 60%, the transfer coefficients are:
\[
e_x = -0.898, e_y = 1.205, e_h = 0.9298, e_{q_x} = -0.197, e_{q_y} = 0.946, e_{q_h} = 0.3457
\]

Case 2: for vane opening level = 80%, the transfer coefficients are:
\[
e_x = -1.248, e_y = 1.313, e_h = 1.3028, e_{q_x} = -0.1035, e_{q_y} = 1.0045, e_{q_h} = 0.3843
\]

V. CONCLUSIONS

Actually, conventional PID control has been largely applied to hydro-turbine governors and has achieved valuable results. The main reason is due to their simplicity of operation, inexpensive maintenance and low cost. But many researches were shown that classical PID control was unable to perform optimally over the full

### Table II. The Result Simulation from GSA, DCEMP, FEP, MFEP and CEP of Scenario 2

<table>
<thead>
<tr>
<th>Method</th>
<th>Vane Opening (%)</th>
<th>Optimization Parameters</th>
<th>( T_a ) (s)</th>
<th>( \delta ) (%)</th>
<th>( N )</th>
</tr>
</thead>
<tbody>
<tr>
<td>CEP</td>
<td>60</td>
<td>2.83, 0.12, 1.41</td>
<td>3.5</td>
<td>1.60</td>
<td>139</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>4.61, 0.23, 0.55</td>
<td>5.6</td>
<td>1.73</td>
<td>103</td>
</tr>
<tr>
<td>FEP</td>
<td>60</td>
<td>2.83, 0.13, 1.38</td>
<td>3.2</td>
<td>1.58</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>4.60, 0.22, 0.54</td>
<td>5.4</td>
<td>1.71</td>
<td>81</td>
</tr>
<tr>
<td>MFEP</td>
<td>60</td>
<td>2.81, 0.12, 1.30</td>
<td>3.1</td>
<td>1.55</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>4.60, 0.21, 0.51</td>
<td>5.1</td>
<td>1.68</td>
<td>55</td>
</tr>
<tr>
<td>DCMEP</td>
<td>60</td>
<td>2.81, 0.12, 1.30</td>
<td>3.1</td>
<td>1.55</td>
<td>52</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>4.60, 0.21, 0.51</td>
<td>5.1</td>
<td>1.68</td>
<td>43</td>
</tr>
<tr>
<td>GSA</td>
<td>60</td>
<td>4.09, 0.14, 4.9</td>
<td>2.9</td>
<td>1.43</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>3.74, 0.22, 0.21</td>
<td>4.8</td>
<td>1.55</td>
<td>50</td>
</tr>
</tbody>
</table>

Figure 6. outing of system for all algorithms for vane opening level = 60%, CEP (...), FEP (+++), MFEP (-.-), DCMEP (***), GSA(____)

Figure 7. outing of system for all algorithms for vane opening level = 80%, CEP (...), FEP (+++), MFEP (-.-), DCMEP (***), GSA(____)
Alireza Ghasemi received the B.S., M.Sc., and Ph.D. degrees in electrical engineering from the University of Tehran in 1981, 1986, and 1990, respectively. In Fall 1989, he received a Fulbright fellowship and joined the University of Arizona. He also received the Ministry of Education fellowship to conduct research at the University of Southern California, Los Angeles, in 1990. From 2001 to 2006, he was a Full Professor at the University of Tehran. He is currently an Engineer of the East Azarbayjan Power Plant, Iran. His research interests include power electronics, power system operation and control, and power system planning and operation.
Oveis Abedinia received the B.S. and M.Sc. degrees in Electrical Engineering from Azad University, Ardabil and Science and Technology Research Branch, Tehran, Iran in 2005 and 2009, respectively. Currently, he is a Ph. D. student in Electrical Eng. Department, Semnan University, Semnan, Iran. His areas of interest in research are Application of Artificial Intelligence to Power System and Control Design, Load and Price Forecasting, Distribution Generation, Restructuring in Power Systems, Congestion Management, Optimization.

Hossein Gholamalitabar Firoozjaee was born in Babol, Mazandaran, Iran, in 1988. He received the B.S. degree in electrical engineering from the University of Nooshirvanibabol, mazandaran, iran, in 2010. He is currently M.S. degree student in electrical engineering group from Mohaghegh Ardebili University, Ardebil, Iran. His current research interests are in power market, renewable energy and optimization with artificial algorithms.