Decentralized PID Controller Design for 3x3 Multivariable System using Heuristic Algorithms

A. H. Devikumari[*] and V. Vijayan

Department of Electronics and Instrumentation Engineering, St. Joseph's College of Engineering, Chennai - 600119, Tamil Nadu, India; devijyostnakiran@gmail.com, vinvpn@gmail.com

Abstract

Background/Objectives: The objective is to design decentralised PID Controller for 3x3 multivariable system using heuristic algorithms. **Methods/Statistical analysis:** It is very difficult to control Multivariable systems in the process industries due to the interaction between the process variables. It needs a decentrlised PID controller scheme for the effective control. Here, Ogunnaike and Ray distillation column is considered as an example for simulation. The PID controllers are tuned using using Firefly (FA) and Particle Swarm Optimization (PSO) algorithm for diagonal elements of the transfer function. **Findings:** From the Simulation it is observed that the performance of Firefly is better than Particle Swarm Optimization for the chosen process model in terms of integral square (ISE) and peak overshoot. **Application/Improvements:** This method of controller design can be used for other distillation column (Example: Wood-berry distillation column). Different optimization algorithm can be used for controller design (Ant Colony optimization algorithm).

Keywords: Decentralised Control, Distillation Column, Firefly, Multivariable System, Particle Swarm Optimization, PID Controller

1. Introduction

Almost all industrial processes are considered as multivariable systems. It is very difficult to control the multivariable process due to the interaction between the variables. In order to meet the high product quality and energy consumption requirements of the process industries, multivariable system has to be controlled effectively. Decentralized PI controllers are used to control the multivariable system. Various methods for designing decentralized PI controller available in the literature. The multivariable controller is designed based on maximum closed-loop log modulus[1]. Because it measures the robustness of the control systems and thereby it can indicate the appropriateness of controller design with respect to the current operating conditions. The Maximum log modulus is identified via relay feedback experiments. In another method Fast Fourier Transform (FFT) technique is used to identify the maximum closed-loop log modulus[2]. A multi-loop control system is designed by adjusting single parameter until is satisfy stability and performance bounds[3]. Synthesis method of Controller design is very simple for designing

decentralized controller. Only the information needed for controller tuning is the dynamic model parameters of the diagonal elements[4]. An auto tuning with least square method can be used to design decentralised controller due to the significant reduction in the duration of relay tests[5]. Effective Relative Gain Array (ERGA) method can be used to design a decentralised controller design. In this method, the interaction effects for a particular loop from all other closed loops are analyzed through both steady-state gain and critical frequency variations[6]. Equivalent transfer functions method is used for designing controller for a selected loop. In this method a best loop pairing is obtained using the RGA (Relative Gain Array), Nederlinski Index (NI) and Relative Normalized Gain Array (RNGA) rules[7,8]. Double feedback and set point filter method are used for improving speed of response and also used to reduce the peak overshoot to a desired value[9,10]. PID controller design based on transfer function estimation[24] and gain margin and phase margin[25–27] also discussed in literature. Decoupler with decentralised controller can be used to reduce the interaction between the loop variables[11]. In recent year, Heuristic methods are used in various engineering optimization problems[28–31].

Ogunnaike and Ray distillation column is considered as example for simulation purpose. It is an example for 3x3 multivariable systems. Both firefly and PSO algorithms are implemented in distillation column and performances in terms of Integral Square Error are compared.

2. Decentralised Controller Design for a 3x3 System

In multivariable system, more manipulated variables will affect the controlled variables in a specific loop or all other control loops. Hence, Multi-Input and Multi-Output (MIMO) process are decomposed into a set of Single-Input and Single-Output processes and then decentralised PID controllers can be designed to stabilize these SISO loops independently.

A 3x3 system is defined by the following equation (1).

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} g11 & g12 & g13 \\ g21 & g22 & g23 \\ g31 & g32 & g33 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \tag{1}$$

The decentralized controller is described by equation (2)

$$G_c(s) = \begin{bmatrix} G_{c1} & 0 & 0 \\ 0 & G_{c2} & 0 \\ 0 & 0 & G_{c3} \end{bmatrix} \tag{2}$$

A controller is placed between output 'y' and input 'u'. A 3x3 system with three decentralised controllers is shown in Figure 1.

A well-known Ogunnaike and Ray 3- by-3 distillation column[12] is considered in this paper. Ogunnaike and Ray transfer function is given in the equation (3).

$$G_p(s) = \begin{bmatrix} \dfrac{0.66e^{-2.6s}}{6.7s+1} & \dfrac{-0.61e^{-3.5s}}{8.64s+1} & \dfrac{-0.0049e^{-s}}{9.06s+1} \\ \dfrac{1.11e^{-6.5s}}{3.25s+1} & \dfrac{-2.36e^{-3s}}{5s+1} & \dfrac{-0.01e^{-1.2s}}{7.09s+1} \\ \dfrac{-34.68e^{-9.2s}}{8.15s+1} & \dfrac{46.2e^{-9.4s}}{10.9s+1} & \dfrac{0.87(11.61s+1)e^{-s}}{(3.89s+1)(18.8s+1)} \end{bmatrix} \tag{3}$$

The Relative Gain Array (RGA) of the above system is given by

$$RGA = \begin{bmatrix} 2.0084 & -0.7220 & -0.2864 \\ -0.646 & 1.8246 & -0.1786 \\ -0.3624 & -0.1026 & 1.465 \end{bmatrix} \tag{4}$$

So, the paring of 3-by-3 MIMO system should be between $(y_1\text{-}u_1)$, $(y_2\text{-}u_2)$ and $(y_3\text{-}u_3)$ under decentralized PID control.
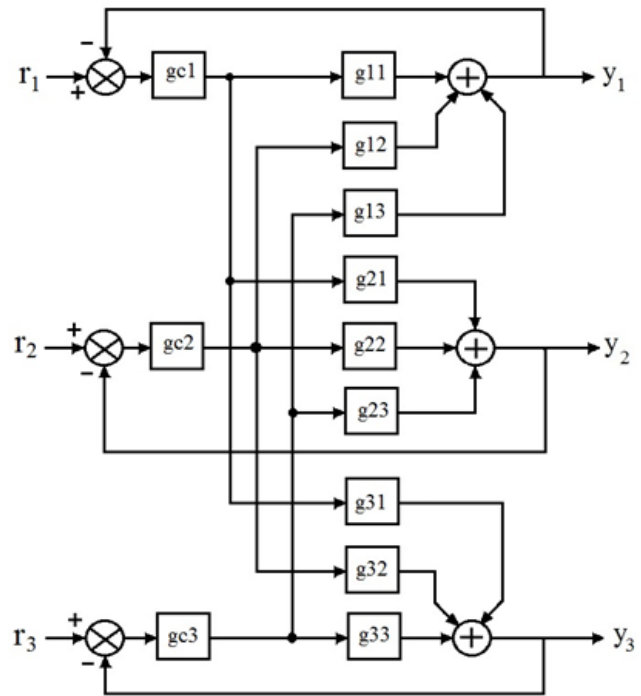

Figure 1. A 3x3 system with three decentralised controllers.

3. Decentralised PID Controller using PSO Algorithm

3.1 Overview of PSO Algorithm

PSO is optimization algorithm based on evolutionary computation technique. The basic PSO is developed from research on swarm such as fish schooling and bird flocking[13–16]. The PSO was introduced in 1995; a modified PSO where inertia weight[15] is added was introduced in 1998 to improve the performance of the original PSO. In this PSO inertia weight is linearly decreasing during iteration. Another type of PSO was reported by Clerc[15]. In PSO, particles compete among themselves. Particles are "evolved" to next generation by competition. Each particle is considered as a point in a D-dimensional space. The particle adjusts its flying comparing its own flying experience with its companion flying experience. Updating particles is based on the following equations:

$$v_{id}^{n+1} = w \cdot v_{id}^{n} + c_1 \cdot r \text{ and}() \cdot \left(p_{id}^{n} - x_{id}^{n} \right) + c_2 \cdot r \text{ and}() \cdot \left(p_{gd}^{n} - x_{id}^{n} \right) \tag{8}$$

$$x_{id}^{n+1} = x_{id}^{n} + v_{id}^{n+1} \qquad (9)$$

The ith particle is represented as $X_I = (x_{i1}, x_{i2},…,x_{iD})$. The best previous position of a particle is represented as $P_I = (p_{i1}, p_{i2},…,p_{iD})$, is called pbest. The index of the best particle among all particles in the population is represented as g, called g-best. The velocity of the ith particle is represented as $V_I = (v_{i1},v_{i2},…,v_{iD})$. c_1 and c_2 are two positive constant.

3.2 Selection of PSO Parameters and Implementation of PSO-based PID Tuning

Eq.8 is used to calculate particle's new velocity according to its previous velocity and the distances of its current position from its best position and the group's best experience. Then the particle flies toward a new position according to Eq 9. The performance of each particle is measured according to a pre-defined performance index, which is related to the problem to be solved. We need to identify certain parameters to start working with PSO. Selection of these parameters decides to a great extent the ability of global minimization. They are,

The maximum velocity: affects the ability of the particles escaping from local optimization and refining global optimization.

The size of swarm: balances the requirement of global optimization and computational cost. rand (): is random function between 0 and 1. n: represents iteration.

Inertia weight (w): is brought into the equation to balance between the global search and local search capability. It can be a positive constant or even positive linear or nonlinear function of time. As recommended in Clerc's PSO, the constants are $c_1=c_2=1.494$. A guaranteed convergence of PSO proposed by Clerc, set w=0.729. Swarm size of 15-50 is usually selected. Here, we set 40.

4. Decentralised PID Controller using Firefly Algorithm

4.1 Overview of Firefly Algorithm

The Firefly Algorithm (FA) is a heuristic, nature-inspired, optimization algorithm which is based on the social flashing behaviour of fireflies or lighting bugs, in the summer sky in the tropical temperature regions[17–23]. It was developed by Dr. Xin-She Yang at Cambridge University in

2007, and it is based on the swarm behaviour such as fish, insects or bird schooling in nature. In recent literature, we can see Firefly algorithm outperform other conventional algorithms and is efficient, for solving optimization problems. The main advantage of firefly algorithm is it uses real random numbers, and global communication among the swarming particles takes place in it. So, it becomes more effective in multi-objective optimization. Firefly algorithm has many similarities with other swarm intelligence algorithms such as Artificial Bee Colony optimization, Particle Swarm Optimization and Bacterial Foraging algorithms, but it is simpler both in concept and implementation[17,18]. FA achieves the global optimization by continuously updating firefly's position based on the brightness and attraction.

When the medium is known, the light intensity of one firefly can be specified by the following equation.

$$L(r) = L_0\, e^{-\gamma r^2} \qquad (10)$$

Where, γ is the absorption coefficient, and L_0 is its initial brightness, its brightness at r = 0.

The attractiveness β of a firefly is determined by equation (11) where β_0 is the attractiveness at

r = 0. As the light intensity decreases with the distance from its source, the attractiveness changes with the distance r_{ij} between firefly i and firefly j. Light is also absorbed by the media.

$$\beta = \beta_0 e^{-\gamma r^m}, (m \geq 1) \qquad (11)$$

The distance between any two fireflies i and j at x_i and x_j respectively, the Cartesian distance is determined by equation (12) where $x_{i,k}$ is the kth component of the spatial coordinate x_i of the ith

Firefly and d is the number of dimensions.

$$r_{ij} = \sqrt{\sum_{k=1}^{d}\left(x_{i,k} - x_{j,k}\right)^2} \qquad (12)$$

4.2 Position Update

When firefly 'i' is attracted to another more attractive firefly 'j', the position update is determined by the following equation.

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \alpha\varepsilon \qquad (13)$$

Where, the second term is due to the attraction while the third term is randomization. α is the randomization parameter and ε is the vector of random numbers drawn

from a Gaussian distribution. The parameter γ defines the contrast of the attractiveness and its value varies from 0.1 to 10. It determines the convergence speed of the FA.

The firefly algorithm has three particular idealized rules which are based on some of the major flashing characteristics of real fireflies [17-23]. These are the following:

(a) All fireflies are unisex, and they will move towards more attractive and brighter ones regardless their sex. (b) The degree of attractiveness of a firefly is proportional to its brightness which decreases as the distance from the other firefly increases due to the fact that the air absorbs light. If there is not a brighter or more attractive firefly than a particular one, it will then move randomly. (c) The brightness or light intensity of a firefly is determined by the value of the objective function of a given problem. For maximization problems, the light intensity is proportional to the value of the objective function.

5. Results and Discussion

Simulation results show the performance of the 3x3 system in time domain specifications and error index for a servo response. The PID gain values are obtained by Firefly and PSO algorithms and the output responses are compared. All the simulations were implemented using MATLAB. Optimized PID parameters using Firefly Tuning is shown in Table 1. Optimized PID parameters using PSO Tuning is shown in Table 2.

Table 1.   Optimized PID parameters using Firefly Tuning

Firefly Tuning	K_p	K_i	K_d
Y1	0.45865	0.1546	0.2380
Y2	-0.4622	-0.0725	-0.0677
Y3	0.5438	0.2583	0.4688

Table 2.   Optimized PID parameters using PSO Tuning

PSO Tuning	K_p	K_i	K_d
Y1	0.7739	0.2465	0.2055
Y2	-0.7235	-0.0890	-1.0059
Y3	0.5098	0.4382	0.1028

Servo output response of 3x3 Ogunnaike and Ray distillation column using Firefly and PSO tuning methods are shown in Figures 2, 3 and 4 respectively.
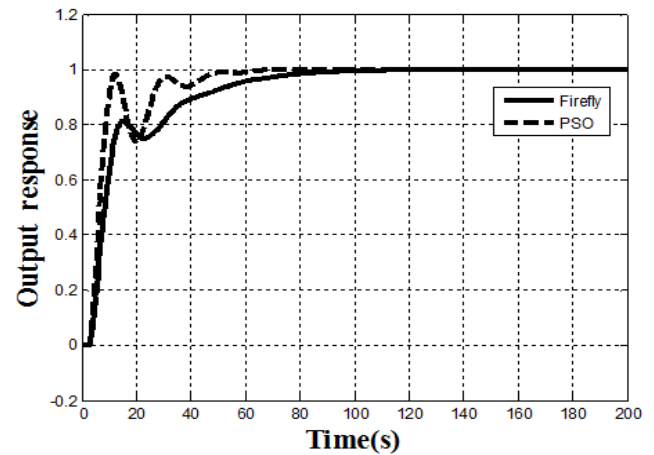

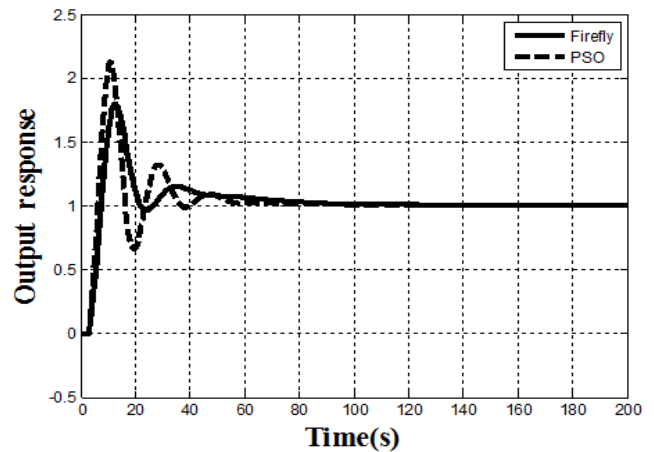Figure 2.   PID Controller response of output Y1.

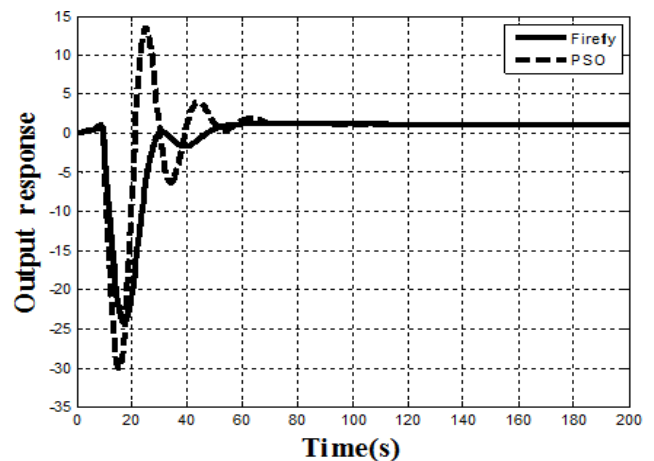Figure 3.   PID Controller response of output Y2.

Figure 4.   PID Controller response of output Y3.

The servo output response for the PID controllers tuned based on Firefly and PSO methods are compared

with time domain specification i.e, peak overshoot, major error criterion techniques such as Integral of Absolute Error (IAE) and Integral Square of Error (ISE). The comparison of the results obtained for servo response of Ogunnaike and Ray system is shown in Table3. It is clear that peak overshoot and error indices are less in Firefly tuning method.

Table 3. Step response performance of PID controllers

Firefly Tuning	PO	IAE	ISE
Y1	1.0000	15.99	7.87
Y2	1.7922	16.72	8.869
Y3	1.4067	337.5	5291
PSO Tuning	**PO**	**IAE**	**ISE**
Y1	1.0011	10.08	5.834
Y2	2.1263	17.07	11.1
Y3	13.4915	384	6468

In the PSO tuned PID controller, the plant response produces slightly high overshoot and high error index but a better performance was obtained with the implementation of Firefly-based PID controller.

6. Conclusion

In this paper, heuristic algorithm based design methods aiming at enhancing PID control for complex MIMO process is implemented. It is shown graphically that there is a substantial improvement in time domain specification in terms of lower overshoot and less error index in Firefly based PID controller. From the results, the designed PID controllers using Firefly based optimization have less overshoot compared to that of the PSO optimization. Furthermore, the Firefly-based PID controllers which are optimized with different performance indices like ISE and IAE have better performances, than PSO controllers.

Therefore the benefit of using a heuristic optimization approach is observed as a complement solution to improve the performance of the PID controller. Of course there are many techniques can be used as the optimization tools and Firefly is one of the recent and efficient optimization tools.

7. References

1. Ju J, Chiu MS. Relay-based on-line monitoring procedures for 2 × 2 and 3 × 3 multiloop control systems. Ind Eng Chem Res. 1997; 3:2225–30.
2. Ju J, Chiu MS. A fast fourier transform approach for on-line monitoring of the maximum closed-loop log modulus. Ind Eng Chem Res. 1998; 37:1045–50.
3. Jung J, Choi JY, Lee J. One-Parameter method for a multiloop control system design. Ind Eng Chem Res. 1999; 38:1580–8.
4. Chien IL, Huang HP, Yang JC. A Simple multiloop tuning method for pid controllers with no proportional kick. Ind Eng Chem Res. 1999; 38:1456–68.
5. Toh WK, Rangaiah GP. A methodology for autotuning of multivariable systems. Ind Eng Chem Res. 2002; 41:4605–15.
6. Xiong Q, Cai WJ, He MJ, He M. Decentralized control system design for multivariable processes-a novel method based on effective relative gain array. Ind Eng Chem Res. 2006; 45:2769–76.
7. Shen Y, Cai WJ, Li S. Multivariable process control: decentralized, decoupling, or sparse? Ind Eng Chem Res. 2010; 49:761–71.
8. Shen Y, Li S, Li N, Cai WJ. Partial decoupling control for multivariable processes. Ind Eng Chem Res. 2011; 50:7380–7.
9. Vijayan V, Panda RC. Design of pid controllers in double feedback loops for siso systems with set-point filters. ISA Transactions. 2012; 51:514–21.
10. Vijayan V, Panda RC. Design of set point filter for minimizing overshoot for low order process. ISA Transactions. 2012; 51:271–6.
11. Ramadevi C, Vijayan V. Design of decoupled pi controller for quadruple tank system. International Journal of Science and Research. 2014; 3(5):318–23.
12. Luyben WL. Simple method for tuning siso controllers in multivariable systems. Ind Eng Chem Process Des Dev. 1986; 25:654–60.
13. Kennedy J, Eberhart RC. Particle swarm optimization. Proceedings of IEEE International Conference on Neural Networks. 1995; 4:1942–8.
14. Ou C, Lin W. Comparison between pso and ga for parameters optimization of pid controller. 2006 International Conference on Mechatronics and Automation; IEEE; 2006. p. 2471–5.
15. Shi Y, Eberhart, R. A modified particle swarm optimizer. IEEE International Conference Evolutionary Computation; 1998. p. 69–73.
16. Clerc M. The Swarm and the queen: towards a deterministic and adaptive particle swarm optimization. Proceedings of the Conference on Evolutionary Computation; 1999. p. 1951–7.
17. Yang XS. Firefly algorithm, stochastic test functions and design optimisation. International Journal of Bio-Inspired Computation. 2015 Mar 20; 2(2):78–82.
18. Yang XS. Firefly algorithm, stochastic test functions and design optimisation. International Journal of Bio-Inspired Computation. 2010; 2:78–84.
19. Yang XS. Firefly algorithm, levy flights and global optimization. Research and Development in Intelligent Systems

XXVI. London, UK: Springer; 2010. p. 209–18

20. Rajinikanth V, Latha K. PSO-Based PID controller design for a class of stable and unstable system. ISRN Artificial Intelligence. 2013; 543607.

21. Raja SMN, Rajinikanth V, Latha K. Otsu based optimal multilevel image thresholding using firefly algorithm. Modelling and Simulation in Engineering. 2014; 794574.

22. Saranya R, Vijayan V. Design of PI Controllers for Unstable MIMO System Using Firefly Algorithm. International Journal of Science and Research. 2015; 4(5):294–9.

23. Lukasik S, Zak S. Firefly algorithm for continuous constrained optimization tasks. International Conference on Computer and Computational Intelligence (ICCCI'09);, Wroclaw, Poland: Springer; 2009. 5796:97–106.

24. Meena S, Chitra K. A New approach of pid tuning for nonlinear siso system based on particle swarm optimization technique. International Journal of Applied Engineering Research. 2015; 9(23):21701–11.

25. Schei TS. Automatic tuning of pid controllers based on transfer function estimation. Automatica. 1994; 30:1983–9.

26. Astrom KJ, Hagglund T. Automatic tuning of simple regulators with specifications on phase and amplitude margins. Automatica. 1984; 20:645–51.

27. Ho WK, Lim KW, Xu W. Optimal gain and phase margin tuning for pid controllers. Automatica. 1998; 34:1009–14.

28. Sethuramalingam TK, Nagaraj B. PID controller tuning using soft computing methodologies for industrial process: a comparative approach. Indian Journal of Science and Technology. 2014; 7(S7):140–5.

29. Kaliappan V, Thathan M. Enhanced abc based pid controller for nonlinear control systems. Indian Journal of Science and Technology. 2015; 8(S7):48–56.

30. Rajinikanth V, Couceiro MS. rgb histogram based color image segmentation using firefly algorithm. Procedia Computer Science. 2015; 46:1449–57.

31. Singh R, Kuchhal P, Choudhury S, Gehlot A. Implementation and Evaluation of Heating System using PID with Genetic Algorithm. Indian Journal of Science and Technology. 2015; 8(5):413–8.