Evolutionary Technique Based Design of Reduced Order Feedforward Controller

Ashish Kumar Jain
Department of Electronics & Instrumentation Engineering, MJPRU, Bareilly
akjain00@gmail.com

Keyword: Feed forward Control, Ziegler Nichols Tuning, Particle Swarm Optimization, Model Order Reduction, Integral Square Error

Abstract: This paper shows the importance of the model order reduction using one of the evolutionary technique namely Particle Swarm Optimization (PSO) by designing a combined feedback – feedforward controller for a given plant by minimizing the actuating error signal. The feed forward controller is designed to eliminate the undesirable effects of measurable disturbances. There are two approaches for the controller design. The first is developed to design the PID feedback controller excluding feedforward loop while the second takes care of the design of feedforward controller using model order reduction technique.

INTRODUCTION

In a closed loop or feedback control system an actuating error signal $E(s)$ in fig.1 is fed to the controller $G_c(s)$ so as to reduce the error and bring the output of the system to a desired value, so as to make the system response relatively insensitive to external disturbances and internal variations in system parameters.

The Proportional-Integral-Derivative (PID) controllers are widely being used in industries for process control applications [1]. The merit of using PID controllers lie in its simplicity of design and good performance indicated by controller parameters such as percentage overshoot, settling time (t_s), rise time (t_r) and peak time (t_p) for any industrial process. It improves both transient response and steady-state error performance. The PID controller has the following transfer function:

$$G_c(s) = K_p + K_d \times s + \frac{K_i}{s} \quad (1)$$

Where K_p = Proportional Gain

K_d = Derivative Gain

K_i = Integral Gain

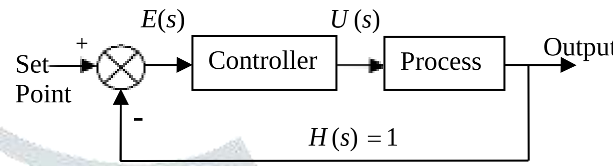

Fig.1 General Closed Loop System

Since in a usual feedback control scheme as shown in Fig.(2), the corrective action starts only after the output has been affected and further the limitations posed by feedback control such as instability, large/significant dead time (i.e. unsatisfactory performances for slow processes) [2]. To overcome these, the feedforward control scheme shown in Fig.(3) becomes a more effective and attractive method of eliminating the undesirable effects (on the system outputs) of the measurable disturbances by approximately compensating for them before they materialized [3]. Because of the disadvantages associated with the feedforward control i.e. more sensitive to process parameters variations, requirement of prior knowledge of the process model and measurement of all possible disturbances, there is need of the proposed combined controller of feedback and feedforward controls using evolutionary technique viz. PSO is of great advantages as confirmed by the results obtained in this work explained in design procedure.

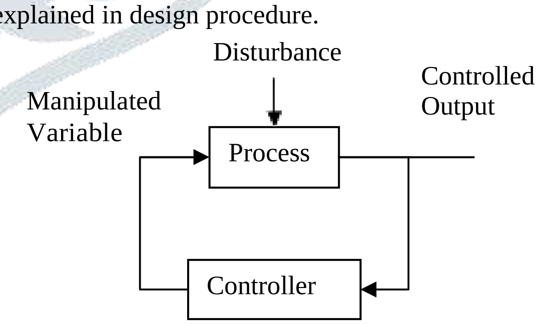

Fig.2 Feedback Control Scheme

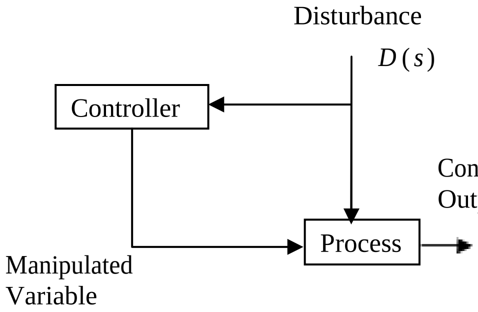

Fig.3 Feedforward Control Scheme

PARTICLE SWARM OPTIMIZATION

In PSO, the 'swarm' is initialized with a population of random solutions. Each particle in the swarm is a different possible set of the unknown parameters to be optimized. Representing a point in the solution space, each particle adjusts its flying toward a potential area according to its own flying experience and shares social information among particles. The goal is to efficiently search the solution space by swarming the particles toward the best fitting solution encountered in previous iterations with the intent of encountering better solutions through the course of the process and eventually converging on a single minimum error solution. The position corresponding to the best fitness is known as $pbest$ and the overall best out of all the particles in the population is called $gbest$. The velocity update in a PSO consists of three parts; namely momentum, cognitive and social parts. The balance among these parts determines the performance of a PSO algorithm. The parameters c_1 and c_2 determine the relative pull of $pbest$ and $gbest$ and the parameters r_1 and r_2 help in stochastically varying these pulls. The modified velocity and position of each particle can be calculated using the current velocity and the distances from the $pbest_{j,g}$ to $gbest_g$ as shown in the following formulae [4,5,6].

$$v_{j,g}^{(t+1)} = w * v_{j,g}^{(t)} + c_1 * r_1(\ ) * (pbest_{j,g} - x_{j,g}^{(t)}) + c_2 * r_2(\ ) * (gbest_g - x_{j,g}^{(t)}) \quad (2)$$

$$x_{j,g}^{(t+1)} = x_{j,g}^{(t)} + v_{j,g}^{(t+1)} \quad (3)$$

Where w is inertia factor which balances the global wide-rang exploitation and the local nearby exploration abilities of the swarm. The computational flowchart of the PSO process is given in Fig.4.

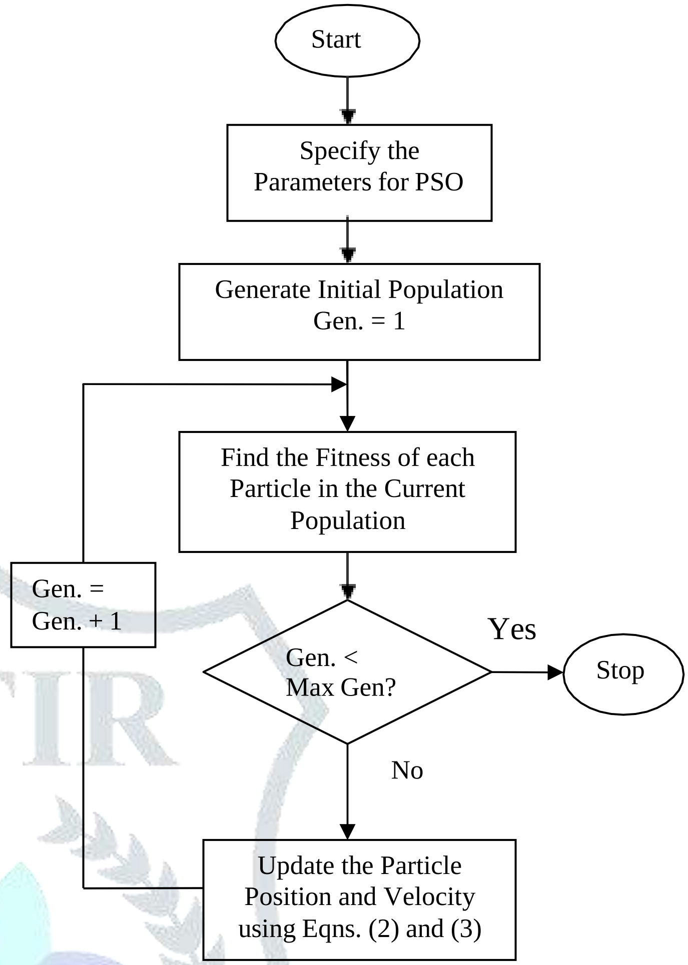

Fig.4 Flow Chart of PSO Algorithm

DESIGN PROCEDURE

Step1:To design a PID controller $G_c(s)$ as feedback controller for a given plant $G_p(s)$ without considering the feedforward loop as shown in Fig.5 by using the Ziegler Nichols rules of tuning PID controller [7].

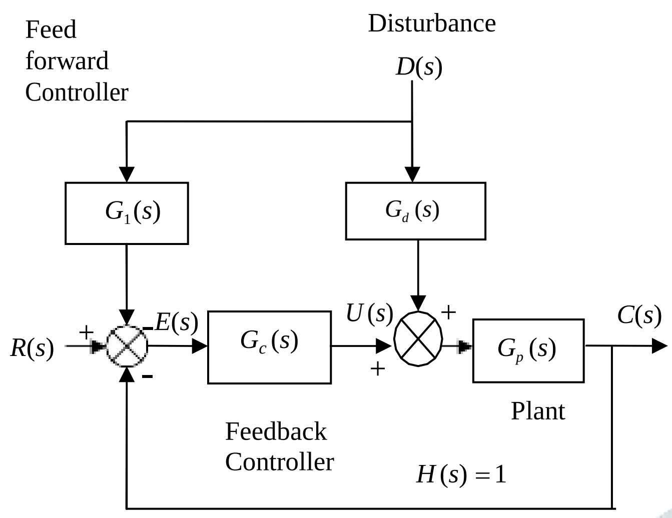

Fig.5 Block Diagram for Feedforward – Feedback Control

Step 2: By combining Feedforward loop

$$C(s) = [\{R(s) - C(s) - D(s)G_1(s)\} \times G_c(s) + D(s)G_d(s)] \times G_p(s) \quad (4)$$

$$C(s) = \{R(s) - C(s)\} \times G_c(s)G_p(s) + D(s)\{-G_1(s)G_c(s) G_p(s) + G_d(s)G_p(s)\} \quad (5)$$

To nullify the effect of disturbance $D(s)$

$$-G_1(s)G_c(s)G_p(s) + G_d(s)G_p(s) = 0 \quad (6)$$

Where $G_p(s)$ cannot be zero, hence

$$G_1(s) = \frac{G_d(s)}{G_c(s)} \quad (7)$$

Step 3: The higher order feedforward controller $G_1(s)$ is reduced in low order by using proposed algorithm of order reduction named as PSO. The fitness function J in the proposed algorithm is taken as an integral squared error of difference between the responses given by the expression:

$$J = \int_0^{t_\infty} [g_{1zn}(t) - g_{1znl}(t)]^2 \, dt \quad (8)$$

Where $g_{1zn}(t)$ and $g_{1znl}(t)$ are the unit step responses of original and reduced order systems. The values of parameters e.g, swarm size, number of iterations, w, c_1 and c_2 used for implementation of this algorithm are 25, 10, 0.4, 1.5 and 1.5 respectively.

The procedure explained above is illustrated for the following plant $G_p(s)$ [8].

$$G_p(s) = \frac{s+1}{(s+2)(2s+3)} \quad \text{and} \quad G_d(s) = \frac{5}{(s+2)}$$

The following PID controller is obtained by using step1.

$$G_{czn}(s) = 2.38 + 2.64s + \frac{0.537}{s}$$

The corresponding close loop transfer function of system without considering disturbance is

$$G_{clzn}(s) = \frac{C(s)}{R(s)} = \frac{2.64s^3 + 5.02\,s^2 + 2.917\,s + 0.537}{4.64s^3 + 12.02\,s^2 + 8.917\,s + 0.537}$$

The step responses of $G_{clzn}(s)$ is shown in Fig.6.

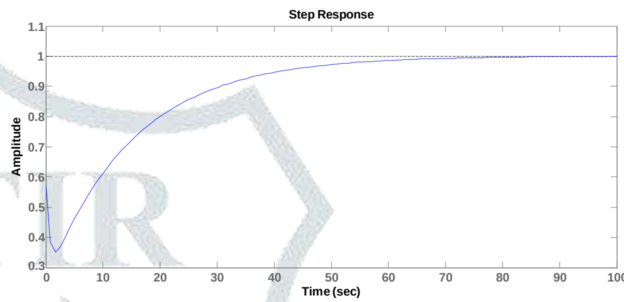

Fig.6 Step response of close loop system with only feedback loop (without considering disturbance)

The derived feed forward controller by using step 2 is following as:

$$G_{1zn}(s) = \frac{5s}{2.64s^3 + 7.66\,s^2 + 5.297\,s + 1.074}$$

Corresponding transfer function of lower order feedforward controller is

$$G_{1lzn}(s) = \frac{0.864s}{s^2 + 0.883s + 0.18}$$

The comparison of step responses of original (3rd) and reduced (2nd) order feedforward controller is shown in Fig.7

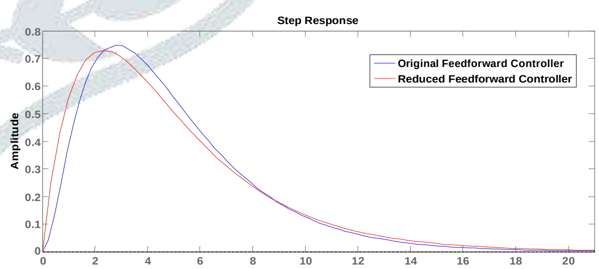

Fig.7 Comparison between step responses of original and reduced order feedforward controller.

The combined effect of feedback-feedforward controller is shown in Fig.8 by comparing the step responses of only feedback and combined feedback-feedforward controller.

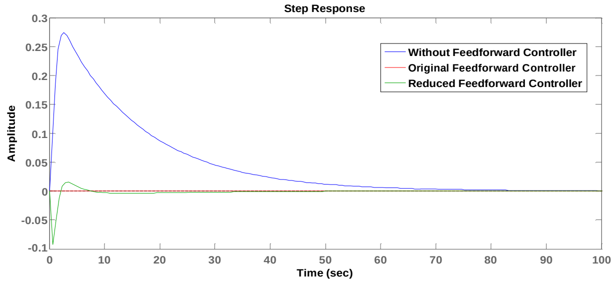

Fig.8 Comparison of step responses of close loop transfer function ($C(s)/D(s)$) with and without feedforward controller.

References

1. Jun-Yi Cao, Bing-Gang Cao, "Design of Fractional Order Controller Based on Particle Swarm Optimization" International Journal of Control, Automation, and Systems, Vol. 4, No. 6, pp.775-781, December, 2006.
2. Norman S Nise "Control System Engineering" John Wiley & Sons, United States of America. Pp 256-258, 2000.
3. Ogata K., "Modern Control Engineering" Prentice - Hall of India Private Limited, pp 608-610, 1996.
4. Kennedy J., Eberhart R., "Particle Swarm Optimization", Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, pp. 1942-1945, 1995
5. Sivanandam S.N., Deepa S.N. "A Comparative Study Using Genetic Algorithm and Particle Swarm Optimization for Lower Order System Modelling" International Journal of the Computer, the Internet and Management Vol. 17, No.3, pp 1-10 December, 2009
6. Tomar S.K., Jain A.K. "Particle Swarm Optimization Based Reduced Order Models for Controller Design", XVII International Conference on System Science, Poland, pp-43, September, 2010
7. Stephenopolous G., "Chemical Process Control- An Introduction to Theory and Practice", Pearson Prentice Hall, pp-455, 2007