IJIREEICE
ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 4, Issue 6, June 2016

PID Controller Design Using Exact Model Matching and Multiobjective Optimization with GA

Rahul Kumar[1], Shiv Narayan[2]

M.E. Scholar, Electrical Engineering Department, P.E.C. University of Technology, Chandigarh, India [1]

Professor, Electrical Engineering Department, P.E.C. University of Technology, Chandigarh, India [2]

Abstract: This paper presents design methodology for PID controller using Exact Model Matching (EMM) approach and multiobjective optimization with GA. The main objective is to design such a controller that can mimic both static and dynamic behaviour of user defined linear model. Here, we use software based approach in MATLAB environment for designing linear PID and also optimizing it using multiobjective optimization based on Genetic Algorithm (GA). Simulation results confirm the usefulness of multi-objective optimization based GA approach for designing linear PID controller to have better servo-regulator trade-off.

Keyword: Linear PID controller, Genetic Algorithm, exact model matching, Multi-objective Optimization.

I. INTRODUCTION

Model matching control is an important part of robust control system and first proposed in [9]. The main aim of the EMM controller is to design a closed loop transfer function model that exactly follows the reference plant model [7-8]. In, EMM, the desired closed loop plant has same static and dynamic behaviour as that of reference closed loop plant. The procedure of model matching controller design for SISO delay system is given in [13]. The model matching with delay system is presented in [11].

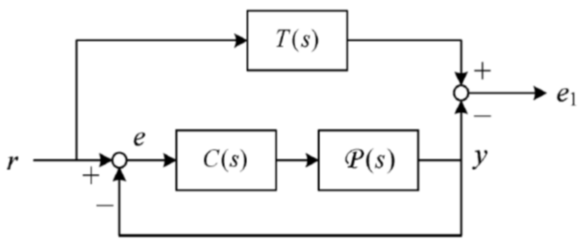

Fig 1 Block diagram of the model matching problem

In today's industrial process, the mostly used controller is PID controller. They are very simple and reliable but they can be applied only on linear plant model. Since, PID controllers have many advantages; researchers have developed design technique for non-linear controller also [12].

In this paper, linear closed loop PID controller is designed with EMM approach. The EMM approach imitates the desired plant transfer function with the reference plant model. A MATLAB function is created which automates to get the PID controller gain in MATLAB environment within the desired frequency range.

In the approach presented in this paper the PID controller design using EMM is carried out first, than the performance as regards the time domain specification, including robustness are compared with another design using multiobjective optimization with GA.

The remaining paper is organised as follows: Problem statement is described in Section-II, section-III gives overview on PID controller, exact model matching design equations are given in section-IV, section-V gives overview on genetic algorithm and its process, section-VI provides simulation and comparison and finally, section-VII gives conclusion of the paper.

II. PROBLEM STATEMENT

In this paper, a process control with time delay i.e. First Order plus Time Delay (FOPTD) plant transfer function model is considered.

General form of FOPTD plant model is given as:

$$P(s) = \frac{K_{pg}\, e^{-hs}}{1 + \tau_c s} \qquad (1)$$

where, K_{pg} = process gain,
h = time delay,
τ_c = time constant.

Process control plant model must be expressed as rational function; and for this reason we are using Padé approx. method to approximate FOPTD plant model. The approximation of the delay element e^{-hs} is given as:

$$e^{-hs} \approx \frac{1 - \frac{h}{2}s}{1 + \frac{h}{2}s} \qquad (2)$$

The above approximation is made to make the controller proper. Now, by putting the approximation of delay term in plant model, the transfer function becomes:

IJIREEICE
ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 4, Issue 6, June 2016

$$P(s) \approx \frac{K}{1+\tau s} \times \frac{1-\frac{h}{2}s}{1+\frac{h}{2}s} \qquad (3)$$

For above configuration, the unity feedback controller is having following form:

$$K_{PID}(s) = \frac{K_D s^2 + K_P s^1 + K_I}{s} \qquad (4)$$

where, K_P, K_I and K_D are controller gains. Thus, above controller contains three degree of freedom and considered as parallel PID controller form.

In this paper, software based on the MATLAB environment is developed to obtain a linear PID controller which would mimic user defined static and dynamic performance. Also, we are using multi-objective optimization based on GA technique to minimize the fitness function. Here, fitness function is considered as a multi-objective function, since we have to deal with the trade off between servo-regulator modes. To maintain a servo and regulator trade-off, we are minimizing Integral Squared Error (ISE), to get a better set-point tracking and peak of the sensitivity function (M_s), which results in better robustness.

III. PID CONTROLLER

A PID closed loop feedback controller is mainly employed in industrial process control system. PID controller commonly calculates the error between process model value and desired set-point value. PID controller is mainly used to reduce this error value over time by using adjustable PID constants i.e. K_P, K_I and K_D.

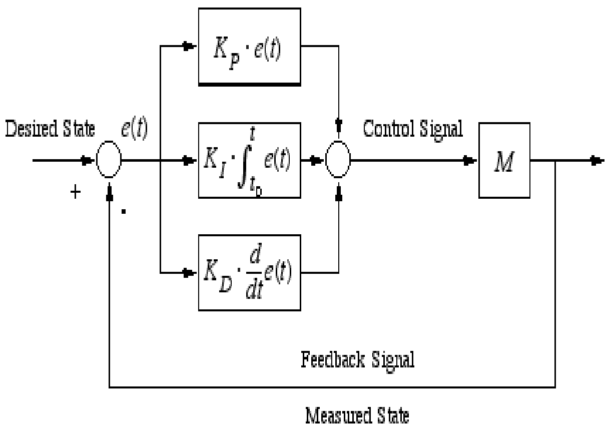

Fig 2. Block diagram of PID controller

The parallel form of PID controller in Laplace from is given as follows:

$$G_{PID}(s) = \frac{K_D s^2 + K_P s^1 + K_I}{s} \qquad (5)$$

where,
K_P = Proportional gain
K_I = Integral gain
K_D = Derivative gain

IV. EXACT MODEL MATCHING DESIGN EQUATIONS

Consider a closed loop control system having PID controller, $K_{PID}(s)$ with plant G(s) in series path and having unity feedback [1]. Design equations are as follows:

The transfer function of plant model is given as follows:

$$G(j\omega) = \frac{G_1 + G_2 j}{G_3 + G_4 j} \qquad (6)$$

The PID controller is given by:

$$G_{PID}(j\omega) = \frac{K_D (j\omega)^2 + K_P (j\omega)^1 + K_I}{(j\omega)} \qquad (7)$$

The desired user defined reference plant model is given as:

$$H(j\omega) = \frac{H_1 + H_2 j}{H_3 + H_4 j} = \frac{Y_1}{Y_2} \qquad (8)$$

And the designed closed loop system

$$G_P = \frac{G * G_{PID}}{1 + G * G_{PID}} = \frac{X_1}{X_2} \qquad (9)$$

The EMM objective function is given by:

$$J_C = \int_{\omega_1}^{\omega_2} |G_P - H|^2 d\omega \qquad (10)$$

Now, put Eqs. (8) and (9) in Eq. (10), we get:

$$J_C = \int_{\omega_1}^{\omega_2} \left|\frac{X_1}{X_2} - \frac{Y_1}{Y_2}\right|^2 d\omega \qquad (11)$$

Under optimal condition, $\frac{X_1}{X_2} = \frac{Y_1}{Y_2}$ Or, $X_1Y_2 - X_2Y_1=0$ but our objective is to find a closed loop solution so instead of minimizing Eq. (11) we are minimizing below objective function.

$$J_C = \int_{\omega_1}^{\omega_2} |X_1Y_2 - Y_1X_2|^2 d\omega \qquad (12)$$

Putting Eqs. (6)-(9) in eq. (12) we get,

$$J_C = \int_{\omega_1}^{\omega_2} \{(-K_P\omega Q + K_I P - K_D\omega^2 P + R)^2 + (K_P\omega P + K_I Q - K_D\omega^2 Q + S)^2\} d\omega \qquad (13)$$

where,

$P = G_1H_3 - G_2H_4 - G_1H_1 + G_2H_2$ (14)
$Q = G_2H_3 + G_1H_4 - G_2H_1 - G_1H_2$ (15)
$R = \omega*(G_4H_1 + G_3H_2)$ (16)
$S = \omega*(G_4 H_2 - G_3H_1)$ (17)

Now taking derivative of J_C with respect to PID controller gain, and equating the results to 0, the results obtained are as follows:

$$\begin{bmatrix} A_0 & -A_2 \\ -A_2 & A_4 \end{bmatrix} \begin{bmatrix} K_I \\ K_D \end{bmatrix} = \begin{bmatrix} -B \\ C \end{bmatrix} \qquad (18)$$

$$K_P = \frac{D}{A_2} \qquad (19)$$

where,

$$A_x = \int_{\omega_1}^{\omega_2} \omega^x * (P^2 + Q^2) d\omega, \quad = 0,2,4 \qquad (20)$$

$$B = \int_{\omega_1}^{\omega_2} (PR + QS) d\omega \qquad (21)$$

$$C = \int_{\omega_1}^{\omega_2} \omega^2 * (PR + QS) d\omega \qquad (22)$$

$$D = \int_{\omega_1}^{\omega_2} \omega * (PR + QS) d\omega \qquad (23)$$

Therefore, by solving above equations we can directly get PID controller gains in minimum time as compared to other tuning techniques.

V. GENETIC ALGORITHMS (GAS)

A. Introduction

Genetic Algorithm (GA) is a computational method that uses random search technique that imitates the process of natural selection and evolution. It is very useful optimization technique used to minimize or maximize the objective function.

GA is generally used as search and optimization technique in which some ideas are hired which are used artificially to develop algorithms that are robust in nature and require minimum problem information. The GA algorithm is motivated by natural evolution and selection and is having many advantages over classical techniques like broad applicability and ease of use [14].

B. GAs Process

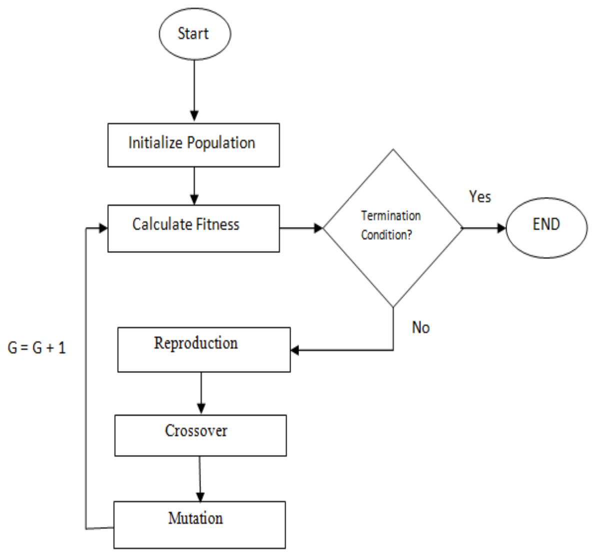

Fig. 3 Genetic Algorithm Outline

Genetic Algorithm (GA) process include following steps:

Step1: Initialize population.
Step2: Calculate the function to be minimized /maximised.
Step3: If terminations conditions are satisfied then stop the program else go to step 4.
Step4: In this step good strings are selected from the population and eliminate bad strings.
Step5: Implement the crossover over the reproduced strings. Select two parent strings from good string and crossover to get new strings.
Step6: Execute mutation on the string produced in above step to create a better string.
Step7: Increase the generation size by 1 then go to step-2 and repeat till optimal solutions are not obtained or generation size not exceeded.

C. GA Operators

There are three main operator of GAs and these operators are usually run for a pre-specified no of generation or until a best fitness value is found [14].

These operators are:

a) Reproduction operators: reproduction or selection basically select good strings and remove bad string from the population. It makes duplicates of good string so that there are plenty of good strings available for crossover operation.

b) Crossover operators: this operator creates new strings in the population. In this two strings are picked which commonly known as parent string at random from the mating pool and it exchange some portion of string from one another to create new string from parent string.

c) Mutation operator: This operator also creates new strings in the population by altering it locally to create better string than the previous one. This is done so that the algorithm converges to the optimum value.

VI. TUNING OF LINEAR PID CONTROLLER

A. Software Based Approach

The EMM technique described above is automated by using a software based function which is developed in the MATLAB software [1]. The new MATLAB function which will automate the above mentioned design equations is given as:

$$[K_P, K_I, K_D] = PID\_GAIN(W_1, W_2, S_S, G_P, H) \qquad (24)$$

where, K_P, K_I and K_D are PID controller gains. The inputs of the MATLAB function are: (1) W_{g1} is lower frequency of interest; (2) W_{g2} is upper frequency of interest; (3) S_h is the step size for the integration; (4) G_P is the given plant's transfer function; (5) H is the reference plant model whose static and dynamic behavior is to be matched [1].

B. IMC Tuning

Tuned parameter values of IMC based PID controller for the FOPDT plant/process model are given in Table I.

Table I tuning parameter values of IMC based PID controller

Tuning parameter	K_p	K_I	K_D
λ	$\frac{2\tau + \theta}{k(\theta + 2\lambda)}$	$\frac{K_C}{\left(\tau + \frac{\theta}{2}\right)}$	$K_C * (\frac{\tau\theta}{\theta + 2\tau})$

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 4, Issue 6, June 2016

As shown in Table I, K_p, K_I and K_D are depending upon the parameter 'λ'. Hence, tuning of IMC based PID controller is easier as we have to tune only one parameter to get the desired output [15].

C. Proposed Tuning Method: Multi-objective GA based Linear PID controller Design

In this proposed tuning method we are using multi-objective optimization based on GA approach to find out the various gains of the PID controller like K_P which is proportional gain, K_I which is Integral gain and K_D which is derivative gain. There are various Integral error criteria like ISE, IAE, IATE, etc. which can be used to minimize the fitness function to get the desired response.

Here, in present approach we are using two factors in minimizing the fitness function. Firstly to have a better performance we are minimizing ISE and secondly for better robustness we are minimizing M_S which is the peak of the sensitivity function. Both these factors are simultaneously minimized so that we can have better performance and robustness of the system. And after minimizing the multiobjective fitness function we get the optimal value of the various gains of the PID controller which make the unity feedback control system to have better trade-off between robustness and performances.

VII. SIMULATION AND COMPARISON

In this section, the simulation study of software based approach and proposed multiobjective optimization technique based on GA for the FOPTD process model, which is widely used in process control industries/plants. The simulations are carried out in MATLAB/SIMULINK and different performance/robustness specification and time domain specification matrices have been calculated and compared.

In this paper we are using FOPTD plant process model which is taken from [13].

$$G_P(s) = \frac{e^{-5s}}{1+10s} \quad (25)$$

And for desired transfer function following form is taken:

$$H(s) = \frac{e^{-5s}}{1+20s} \quad (26)$$

Now, the delay is approximated using padé approximation which is given by:

$$e^{-5s} \approx \frac{1-\frac{5}{2}s}{1+\frac{5}{2}s} \quad (27)$$

By putting Eq. (27) in Eqs. (25) and (26) we get:

$$G_P(s) \approx \frac{1}{1+10s} \times \frac{1-\frac{5}{2}s}{1+\frac{5}{2}s} \quad (28)$$

and,

$$H(s) \approx \frac{1}{1+20s} \times \frac{1-\frac{5}{2}s}{1+\frac{5}{2}s} \quad (29)$$

The developed MATLAB function in Eq. (24) is run with Wg1 = 0, Wg2 = 0.025 and Sh = 0.001. Results of tuning rules is calculated and compared and are tabulated in Table I, Table II and Table III.

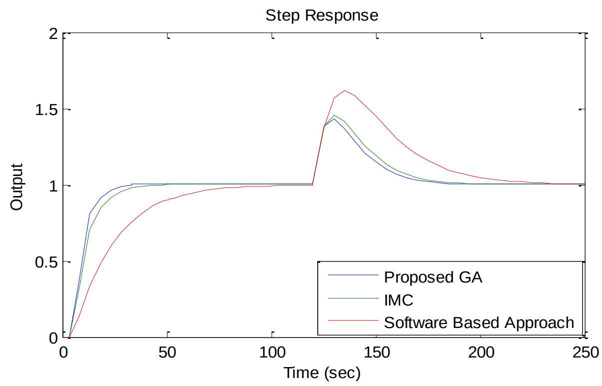

Fig 4 Output step responses of Plant Model for different tuning rules with disturbance at Input of the Plant

Table I Comparisons of various gains of PID controller

Model	Tuning	K_P	K_I	K_D
$\frac{e^{-5s}}{1+10s}$	Proposed GA	1.57407	0.0925	2.3148
	IMC	1	0.08	2
	Software Based	0.42	0.04	0.30

Table II Comparison of Time Domain Specification

Model	Tuning	Rise Time (sec)	Settling Time (sec)
$\frac{e^{-5s}}{1+10s}$	Proposed GA	12.8461	174.5153
	IMC	17.1833	179.7673
	Software Based	44.1984	213.3376

Table III Comparisons of various types of Integral Error Performance

Model	Tuning	ISE		IAE		ITAE	
		SP	LD	SP	LD	SP	LD
$\frac{e^{-5s}}{1+10s}$	Proposed GA	8.357	3.133	11.45	10.81	91.81	1501
	IMC	8.931	3.84	12.63	12.5	120.7	1757
	Software Based	15.23	10.3	25	24.93	528.5	3748

DOI 10.17148/IJIREEICE.2016.4630

VIII. CONCLUSION

In this paper, multiobjective based GA optimization has been used for designing of an closed loop linear PID controller for FOPTD plant model and the result obtained have been compared with the controller design based on model matching.

The design using multiobjective with GA gives better results in terms of the time domain specification and robustness but the value of PID gains are better in software based approach. Both these are compared and tabulated for both tuning methodology and it is seen that the multiobjective based GA approach gives better performance and robustness as compared to software based approach using exact model matching.

REFERENCES

[1] Nassirharand A., Karimi H., Dadfarnia M., "A new software tool for synthesis of linear PID controllers", Advances in Engineering Software 34th, 2003, pp 551–557.
[2] Goldberg D., "Genetic algorithms in searching, optimization and machine learning". Reading, MA: Addison-Wesley, 1989.
[3] Fonseca C. M., Fleming P. J., “Genetic algorithm for multiobjective optimization: Formulation, discussion and generalization”, in Proc. 5st Int. Conf. Genetic Algorithms, 1993, p.p. 416-423.
[4] Astrom K., Hagglund T., “PID Controller: theory, design and tuning”, Instrument Society of America, 1995.
[5] Arrieta O., Vilanova R,“Servo/regulation trade-off tuning of PID controllers with a robustness consideration”. Proc. of the 46th Conference on Decision and Control, 2007.
[6] Meng, X., Song B., “Fast genetic algorithm used for PID parameter optimization”, Proceeding of the IEEE International conference on Automation and Logistics, China, Aug 2007, p.p. 2144-2148.
[7] Wang S. H., Desoer C. A., “The exact model matching problem of linear multivariable systems,” IEEE Trans. Automat. Contr., vol. AC-17, 1972 p.p. 347- 349.
[8] Moore B. C., Silverman L. M., “Model-matching by state feedback and dynamic compensation,” IEEE Trans. Automat. Contr., vol. AC-17, 1972, p.p. 491- 497.
[9] WOLOVICH W.A., “Linear multivariable systems”, Springer-Verlag, New York, 1974.
[10] ICHIKAWA K., “Frequency-domain pole assignment and exact model matching for delay systems”, Int. J. Control, 1985, vol. 41, issue- 4, p.p. 1015–1024
[11] Picard P., Lafay J. F. and Kucera V., "Model Matching for linear systems with delay" Proc. 13th IFAC World Congress, San Fransisco, USA, Vol D, 1996, p.p. 183-188.
[12] Taylor J.H., Strobel K.L., “Nonlinear control system design based on quasilinear system models”. Proceedings of the American Control Conference, Boston, MA, 1985. p.p. 1242–1247.
[13] Wang L, Barnes T.J.D., Cluett W.R., “New frequency-domain design method for PID controllers”. IEE Proc Control Theory Appl 1995, 142(4):265–71. July 1995.
[14] Deb. K., “Multi-objective optimization using evolutionary algorithms”. Chichester, UK, Wiley, 2001.
[15] Rivera D.E., Morari M., Skogestad S., “Internal mode control. 4. PID controller design” Ind. Eng. Chem. Process Des. Dev., 1986, p.p. 252–65.