Designing PID Controller for DC Motor by Means of Enhanced PSO Algorithm with Dissipative Chaotic Map

Michal Pluhacek[1], Roman Senkerik[1], Donald Davendra[2], and Ivan Zelinka[1]

[1] Tomas Bata University in Zlin, Faculty of Applied Informatics,
T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
{pluhacek,senkerik,zelinka}@fai.utb.cz
[2] VŠB-Technical University of Ostrava,
Faculty of Electrical Engineering and Computer Science,
Department of Computer Science, 17. Listopadu 15,
708 33 Ostrava-Poruba, Czech Republic
donald.davendra@vsb.cz

Abstract. In this paper, it is proposed the utilization of chaotic dissipative map based chaos number generator to enhance the performance of PSO algorithm. This paper presents results of using chaos enhanced PSO algorithm to design a PID controller for DC motor system. Results are compared with other heuristic and non-heuristic methods.

1 Introduction

Complex problems of optimization tasks emerged with the spread of computer technology which made possible solving many previously unsolvable problems using their enormous computing power in "brutal force attacks" (trying all possibilities) on these tasks. However it became soon clear that some problems will probably be unsolvable by "brutal force" even in distant future because of their enormous complexity and physical limitations of computer technology.

Genetic based algorithms [1-5] were discovered capable of finding very good solutions for these problems in a short time. Main thought is divided from the discoveries of Charles Darwin about evolution of species. New generation of individuals (representing possible solutions of the problem) is created via transforming the old population following a set of rules. This set of transformation rules is unique for each algorithm.

More recently the soft-computing methods which include neural networks, evolutionary algorithms, genetic programming and fuzzy logic extended their applications in almost every computer-science and engineering discipline.

Good examples of soft-computing possibilities are recent studies of using these methods for designing PID controllers [6], which represents very complex optimization task that can be solved by non-heuristic methods only with limitations. A recent study [7] suggests that using chaos based number generators instead of common computer random number generators can lead to increasing of the performance of evolutionary algorithms in the task of PID controller design.

V. Snasel et al. (Eds.): SOCO Models in Industrial & Environmental Appl., AISC 188, pp. 475–483.
springerlink.com © Springer-Verlag Berlin Heidelberg 2013

This research presents using of Dissipative standard map as discrete chaotic system for the chaotic number generator and implementation of this chaotic generator into PSO algorithm, which is modified by using the inertia weight factor w [8]. This enhanced PSO algorithm is applied on the PID controller design problem.

The main idea and motivation for combining evolutionary algorithms and deterministic chaos system is that both these are originally inspired in nature. So for nature-based algorithm such as PSO it should be much more natural to use chaos number generator such as the dissipative standard map. And the hope is that it might improve the performance of PSO algorithm.

2 Particle Swarm Optimization Algorithm

PSO (Particle swarm optimization) algorithm is based on the natural behavior of birds and fishes and was firstly introduced by R. Eberhart and J. Kennedy in 1995 [1,2]. As an alternative to genetic algorithms [4] and differential evolution [5], PSO proved itself to be able to find better solutions for many optimization problems. Term "swarm intelligence" [2,3] refers to the capability of particle swarms to exhibit surprising intelligent behavior assuming that some form of communication (even very primitive) can occur among the swarm particles (individuals).

Basic PSO algorithm disadvantage is the rapid acceleration of particles which causes abandoning the defined area of interest. In each generation, a new location of a particle is calculated based on its previous location and velocity (or "velocity vector"). For this reason, several modifications of PSO were introduced to handle with this problem. Main principles of PSO algorithm and its modifications are well described in [1-3].

Within this research, chaos driven PSO strategy with inertia weight was used. The selection of inertia weight modification of PSO was based on numerous previous experiments. Default values of all PSO parameters were chosen according to the recommendations given in [2,3]. Inertia weight is designed to influence the velocity of each particle differently over the time [8]. In the beginning of the optimization process, the influence of inertia weight factor w is minimal. As the optimization continues, the value of w is decreasing, thus the velocity of each particle is decreasing, since w is always the number < 1 and it multiplies previous velocity of particle in the process of new velocity value calculation. Inertia weight modification PSO strategy has two control parameters w_{start} and w_{end}. New w for each generation is then given by Eq. 1, where i stand for current generation number and n for total number of generations.

$$w = w_{start} - \frac{\left(\left(w_{start} - w_{end}\right) * i\right)}{n} \tag{1}$$

Chaos driven number generator is used in the main PSO definition (Eq. 2) which determines a new "velocity", thus the position of each particle in the next generation (or migration cycle).

$$\begin{aligned} v(t+1) = v(t) + c_1 \cdot Rand \cdot (pBest - x(t)) + \\ c_2 \cdot Rand \cdot (gBest - x(t)) \end{aligned} \tag{2}$$

Where:

$v(t+1)$ – New velocity of particle.
$v(t)$ – Current velocity of particle.
c_1,c_2 – Priority factors.
pBest – Best solution found by particle.
gBest – Best solution found in population.
$x(t)$ – Current position of particle.
Rand – Random number, from the interval <0,1>. Within Chaos PSO algorithm, the basic inbuilt computer (simulation software) random generator is replaced with chaotic generator (in this case, by using of Dissipative standard map).

New position of a particle is then given by Eq. 3, where x(t+1) represents the new position:

$$x(t+1) = x(t) + v(t+1) \tag{3}$$

3 Dissipative Standard Map

The Dissipative Standard map is a two-dimensional chaotic map. The parameters used in this work are $b = 0.1$ and $k = 8.8$. For these values, the system exhibits typical chaotic behaviour and with this parameter setting it is used in the most research papers and other literature sources [9]. The Dissipative standard map is given in Fig. 1. The map equations are given in Eq. 4 and 5.

$$X_{n+1} = X_n + Y_{n+1} (\mathrm{mod}\, 2\pi) \tag{4}$$

$$Y_{n+1} = bY_n + k \sin X_n (\mathrm{mod}\, 2\pi) \tag{5}$$

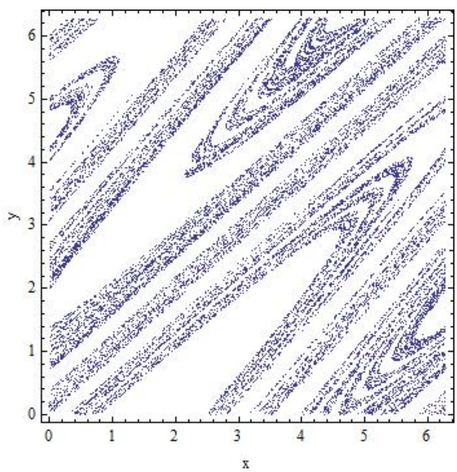

Fig. 1. Dissipative standard map

4 Problem Design

This section contains the description of the PID controller, used model of DC motor as well as the overview of the most important results.

4.1 PID Controller and DC Motor System

The PID controller contains three unique parts; proportional, integral and derivative controller [10]. A simplified form in Laplace domain is given in Eq 6.

$$G(s) = K\left(1 + \frac{1}{sT_i} + sT_d\right) \tag{6}$$

Where K is controller gain, T_i is the adjustable integral time parameter, T_D is the rate time.

The PID form most suitable for analytical calculations is given in Eq 7.

$$G(s) = k_p + \frac{k_i}{s} + k_d s \tag{7}$$

The parameters are related to the standard form through: $k_p = K$, $k_i = K/T_i$ and $k_d = KT_d$. Estimation of the combination of these three parameters that gives the lowest value of the four test criterions was the objective of this research.

The test criterion measures properties of output transfer function and can indicate quality of regulation. Following four different integral criterions were used for the test and comparison purposes: IAE (Integral Absolute Error), ITAE (Integral Time Absolute Error), ISE (Integral Square Error) and MSE (Mean Square Error). For further details see [6,7]. These test criterions were minimized within the cost functions for the enhanced PSO algorithm.

The transfer function of used DC motor is given by Eq. 8. [6,7]

$$G(s) = \frac{0.9}{0.00105s^3 + 0.2104s^2 + 0.8913s} \tag{8}$$

4.2 Cost Function

Test criterion measures properties of output transfer function and can indicate quality of regulation. Following four different integral criterions were used for the test and comparison purposes: IAE (Integral Absolute Error), ITAE (Integral Time Absolute Error), ISE (Integral Square Error) and MSE (Mean Square Error). These test criterions (given by Eq. 9–12) were minimized within the cost functions for the enhanced PSO algorithm.

Integral of Time multiplied by Absolute Error (ITAE)

$$I_{ITAE} = \int_0^T t\left|e(t)\right| dt \tag{9}$$

Integral of Absolute Magnitude of the Error (IAE)

$$I_{IAE} = \int_0^T |e(t)|\, dt \tag{10}$$

Integral of the Square of the Error (ISE)

$$I_{ISE} = \int_0^T e^2(t)\, dt \tag{11}$$

Mean of the Square of the Error (MSE)

$$I_{MSE} = \frac{1}{n}\sum_{i=1}^{n} (e(t))^2 \tag{12}$$

5 Results

The experiments were focused on the optimization of the four different specification functions as given in Section 4.1. The best results of the optimization with corresponding values of k_p, k_i and k_d together with selected response profile parameters are presented in Table 1.

When tuning a PID controller, generally the aim is to match some preconceived 'ideal' response profile for the closed loop system. The following response profiles are typical [11]:

Overshoot: this is the magnitude by which the controlled 'variable swings' past the setpoint. 5-10% overshoot is normally acceptable for most loops.

Rise time: the time it takes for the process output to achieve the new desired value. One- third the dominant process time constant would be typical.

Settling time: the time it takes for the process output to die to between, say +/- 5% of setpoint.

From the statistical reasons, optimization for each criterion was repeated 30 times. Results of the simple statistical comparison for the optimizations by means of chaos driven PSO algorithm are given in tables 2 and 3.

Furthermore obtained results are compared with previously published result [6] given by other heuristic and non-heuristic methods (See table 4).

Optimized system responses are depicted in figures. 2 – 5.

Table 1. The best results for DC motor system

Criterion	CF	Kp	Ki	Kd	Overshoot	Rise Time	Settling time
IAE	0.223055	241.917000	2.557960	58.577400	0.215717	0.010100	0.023300
ITAE	0.008617	297.775000	0.252631	71.815200	0.257957	0.008700	0.030800
ISE	0.018387	144.622000	28.458700	64.141000	0.223852	0.009500	0.032400
MSE	0.000919	147.209000	29.574200	64.035400	0.223748	0.009500	0.032400

Table 2. Average response profiles for DC motor system

Criterion	Avg overshoot	Avg rise time	Avg settling time
IAE	0.223677	0.009847	0.027067
ITAE	0.225357	0.010020	0.028427
ISE	0.224081	0.009507	0.032347
MSE	0.223918	0.009520	0.032360

Table 3. Statistical overview of the cost function (criterion) values for DC motor system

Criterion	Max CF	Min CF	Avg CF	Median
IAE	0.275357	0.223055	0.243871	0.239859
ITAE	0.040698	0.008617	0.020648	0.019312
ISE	0.018448	0.018387	0.018405	0.018399
MSE	0.000924	0.000919	0.000921	0.000920

Table 4. Comparison of other methods and proposed enhanced PSO

Criterion	Z-N (step response)	Kappa-Tau	Continuous cycling	EP	GA	PSO	Chaos PSO
IAE	0.517600	0.518800	0.560000	0.489100	0.771200	0.916100	0.223055
ITAE	3.380500	3.311300	7.820000	0.072100	0.378100	0.022900	0.008617
ISE	2.346700	2.250300	3.200000	1.027700	1.043500	1.001600	0.018387
MSE	0.011700	0.077778	0.016000	0.005100	0.005200	0.005000	0.000919

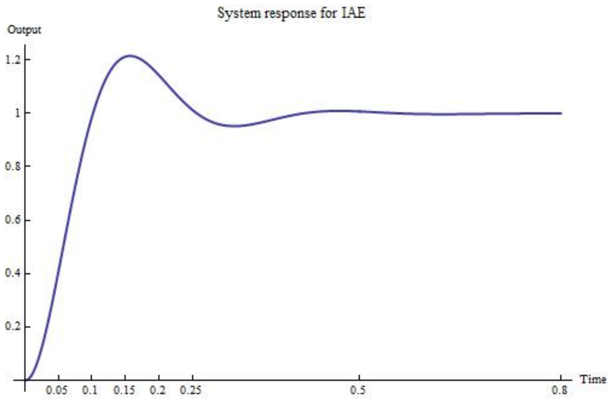

Fig. 2. System response for IAE

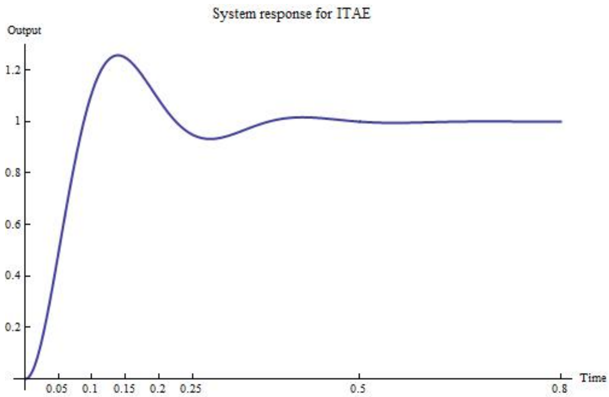

Fig. 3. System response for ITAE

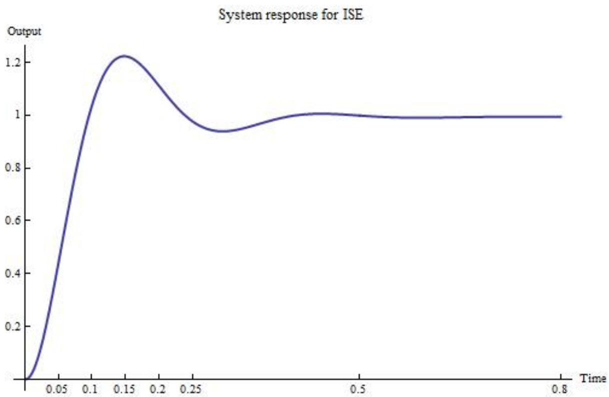

Fig. 4. System response for ISE

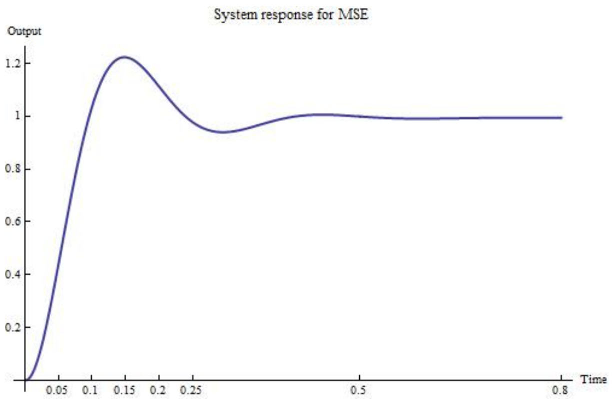

Fig. 5. System response for MSE

6 Brief Analyses of the Results

From Table 4, it follows that all heuristic methods has given better results than non-heuristic methods (e.g. Ziegler-Nichols and Continuous cycling) for the most of test criterions (exception was the IAE criterion). Proposed enhanced PSO with chaos based number generator has given better results for all test criterions than any other heuristic or non-heuristic method compared within this paper. Obtained control parameters (see Table 1) were used to obtain system responses (Figures. 2 - 5) which were afterwards analyzed for detailed response profiles: overshot, rise time and settling time (see Tables 1 and 2).

Given the presented data the PSO algorithm driven by dissipative standard chaos map seems to be a valid tool for PID controller design for such systems as the DC motor system presented in this paper. It also seems so, that this proposed modification outperformed previously presented stochastic methods on this task. However more research and further analyses are still needed to determine the impact of using chaos number generator on the inner dynamic of PSO algorithm.

7 Conclusion

In this paper chaos driven PSO were used to find optimal settings for PID controller for DC motor system. From the presented data, it follows that implementation of chaotic dissipative standard map as a random number generator into PSO algorithm led to improving its performance over other heuristic or non-heuristic methods for solving the PID controller design for DC motor. Further research will be focused on the possibilities of the development and improvement of the enhanced chaos driven PSO algorithm to achieve better results and explore more possible applications for this promising method.

Acknowledgements. This work was supported by European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089, and by Internal Grant Agency of Tomas Bata University under the project No. IGA/FAI/2012/037.

References

1. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. In: Proceedings of IEEE International Conference on Neural Networks. IV, pp. 1942–1948 (1995)
2. Dorigo, M.: Ant Colony Optimization and Swarm Intelligence. Springer (2006)
3. Eberhart, R., Kennedy, J.: Swarm Intelligence, The Morgan Kaufmann Series in Artificial Intelligence. Morgan Kaufmann (2001)
4. Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization 11, 341–359 (1997)
5. Goldberg, D.E.: Genetic Algorithms in Search Optimization and Machine Learning, p. 41. Addison Wesley (1989) ISBN 0201157675
6. Nagraj, B., Subha, S., Rampriya, B.: Tuning algorithms for pid controller using soft computing techniques. International Journal of Computer Science and Network Security 8, 278–281.3 (2008)
7. Davendra, D., Zelinka, I., Senkerik, R.: Chaos driven evolutionary algorithms for the task of PID control. Computers & Mathematics with Applications 60(4), 1088–1104 (2010) ISSN 0898-1221
8. Nickabadi, A., Ebadzadeh, M.M., Safabakhsh, R.: A novel particle swarm optimization algorithm with adaptive inertia weight. Applied Soft Computing 11(4), 3658–3670 (2011) ISSN 1568-4946
9. Sprott, J.C.: Chaos and Time-Series Analysis. Oxford University Press (2003)
10. Astrom, K.: Control System Design. University of California, California (2002)
11. Landau, Y.: Digital Control Systems. Springer, London (2006)