Control of a Manipulator Using Shuffled Complex Evolution

Iraj Hassanzadeh[1], Mohsen Amiri[2], and Atabak M. Kashtiban[1]

[1] Control Engineering Department, Faculty of Electrical & Computer Engineering, University of Tabriz, Iran
Izadeh@tabrizu.ac.ir, mohsen.amiri313@gmail.com and Nirvana_amk@yahoo.com

Abstract

Many Proportional-Integral-Derivative (PID) controller design methods on nonlinear systems are reported in recent years due to its simple structure and powerful performance. This paper presents Shuffled Complex Evolution (SCE) method for PID design process that yeilds more optimum parameters. By adding saturation constraint to the system, our model is more precise than other reported researches. Also we have improved the cost function formulation of the optimization problem. The PID controller has been applied on a 5-bar-linkage manipulator.

1. Introduction

Proportional-Integral-Derivative (PID) controller is one of the most common controllers which has been used in industry during recent years. This controller has three parameters; proportional gain k_p , integral gain k_i, and derivative gain k_d. There are some definite approaches to design PID controller for linear systems. So these powerful advantages can be used in linear systems readily. But it's interesting and helpful to find a way which makes it possible to use this controller in nonlinear systems control, too. As we know, there are no systematic methods to design a PID controller with near optimal parameters. Traditionally trial and error method has been a common method to determine PID controller parameters that usually takes considerable time and does not lead to such accurate results. To overcome these problems, several methods have been developed that give more optimum results with respect to traditional method. Ant colony optimization [1], genetic algorithm [2], fuzzy logic [3], simulated annealing [4], and pattern recognition [5] are new methods. In most recent researches the Particle Swarm Optimization (PSO) [6], and Binary Genetic Algorithm (BGA) [7] algorithms have been tried. In this paper we want to design tow controllers for a 5-bar-linkage manipulator robot. The model of this robot is nonlinear. In comparison with other recent researches, we have added another nonlinear block to the model in order to make it more precise and have considered input signal limits due to saturation constraints. This nonlinearity increases problem complexity and needs some methods to solve optimization problem effectively. we have also changed the cost function of the optimization problem to lead the algorithm to more reasonable solutions. Because of saturation constraint that has been added to model, it's possible to expand the search space of the optimization problem. Eventually we have applied another new algorithm called as Shuffled Complex Evolution (SCE) that is more robust and more precise than other reported methods.

2. Manipulator dynamic equations

In many industrial robotic applications such as spray-painting, arc welding and adhesive or sealant applications a 5-bar-linkage manipulator is a suitable and prevalent robot. Figure 1 shows the 5-bar-linkage manipulator made in robotics research lab in our department. In this section, the dynamic equations of a 5-bar-linkage manipulator are presented. Let q_i ,T_i and I_h^i be the joint variable, torque and hub inertia of the i^{th} motor, respectively. Also, let I_i, l_i, l_{ci} and m_i be the inertia matrix, length, distance to the center of gravity and mass of the i^{th} link, respectively. The measured mass, length and center of gravity of links are reported in Table 1.

Table 1. 5-bar-linkage manipulator data

Link	Mass(Kg)	Length(m)	C of G
1	0.288	0.33	0.166
2	0.0324	0.12	0.06
3	0.3702	0.33	0.166
4	0.2981	0.45	0.075

Fig. 1. Five-bar-linkage manipulator

Figure 2 depicts the 5-bar-linkage manipulator schematic where the links form a parallelogram[8].

2.1. Equations

Equations (1-5) present the dynamic equation of the manipulator [11].

$$T_1 = (M_{11} + I_h^1)\ddot{q_1} + M_{12}\ddot{q_2} + \frac{\partial M_{12}}{\partial q_2}\dot{q_2}^2$$
$$+g(m_1 l_{c1} + m_3 l_{c3} + m_4 l_1)cosq_1 \quad (1)$$

$$T_2 = \left(M_{22} + I_h^2\right)\ddot{q_2} + M_{12}\ddot{q_1} + \frac{\partial M_{12}}{\partial q_1}\dot{q_2}^2$$
$$+g(m_2 l_{c2} + m_3 l_2 - m_4 l_{c4})cosq_2 \quad (2)$$

where g is the gravitational constant and

$$M_{11} = I_{11}^1 + I_{11}^3 + m_1 l_{c1}^2 + m_3 l_{c3}^2 + m_4 l_1^2 \quad (3)$$

$$M_{22} = I_{11}^2 + I_{11}^4 + m_2 l_{c2}^2 + m_3 l_2^2 + m_4 l_{c4}^2 \quad (4)$$

$$M_{12} = M_{21} = (m_3 l_{c2} l_2 - m_4 l_{c4} l_1)\cos(q_1 - q_2) \quad (5)$$

Now, we note from the above equations that if

$$m_3 l_2 l_{c3} = m_4 l_1 l_{c4} \quad (6)$$

then M_{12} and M_{21} are zero, that is, the inertia matrix is diagonal and constant. Hence the dynamic equations of this manipulator will be as below:

$$T_1 = (M_{11} + I_h^1)\ddot{q_1} + g(m_1 l_{c1} + m_3 l_{c3} + m_4 l_1)cosq_1 \quad (7)$$
$$T_2 = \left(M_{22} + I_h^2\right)\ddot{q_2} + g(m_2 l_{c2} + m_3 l_2 - m_4 l_{c4})cosq_2 \quad (8)$$

Notice that T_1 depends only on q_1 but not on q_2. But on the other hand T_2 depends only on q_2 but not on q_1. If the condition in (6) is satisfied, we can adjust the two angles q_1 & q_2 independently, without worrying about interactions between them. Using equations (7) and (8), 5-bar-linkage manipulator robot is easily simulated using Matlab® and Simulink® blocks.

3. PSO and SCE Review

Particle Swarm Optimization (PSO) is a simple search approach formulated by Eberhart and Kenedy in 1995 [9]. In this algorithm each solution to the problem is considered as a bird flying in the problem search space. The flying birds are called as particles. In each iteration the best solution is determined from the following formula:

$$X_{new} = X_{old} + V_{new} \quad (9)$$

where X is the solution vector and V is the velocity vector that refreshes the solution.

$$V_{i,new} = w \times V_{i,old} + c_1 \times rand \times (X_{PB} - X_i) + c_2 \times Rand \times (X_{GB} - X_i) \quad (10)$$

That X_{PB} and X_{GB} are the personal best location and global best location of the i^{th} particle, respectively; rand() and Rand() are two random numbers in the range [0,1]; c_1 and c_2 are two constants that ($c_1 + c_2 \geq 4$) and w is the inertia weight factor. w helps algorithm to climb from local minimums. If the local best solution has a cost less than the cost of the current global best solution, then the local best solution replaces the global best solution. Equation (11) gives the usual formula for w:

$$w = 0.5 \times (1 + rand) \quad (11)$$

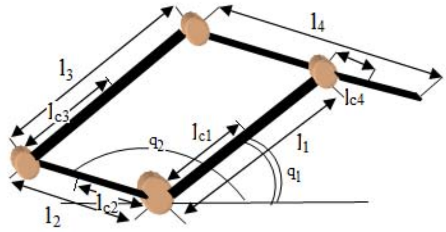

Fig. 2. 5-bar-linkage manipulator plannar presentation

The other algorithm used in this research is a novel one. Shuffled Complex Evolution (SCE) [10] enjoys a special version of a local search algorithm called as Nelder-Mead (NM). By choose of $N(Npop)$ random points in search space and sort them according to their costs. Now they should be divided into P complexes ($N = P \times m$). Then improve these P complexes by apply of a special version of NM on all of them. Eventually shuffle the complexes, sort them again and repeat the steps just like mentioned in above. The presented steps in below, describe more about what should be done in SCE algorithm.

a. select q points randomly from each complex, the points with lower cost must have better chances in selection process.

b. Apply the improved NM on these q points as follow:

$\boldsymbol{i}$. Calculate mean of $(q-1)$ best points $(\bar{x})$.
$\boldsymbol{ii}$. Reflect the worst point (x_w) with respect to $(\bar{x})$ to reach a new point in search space called as x_r (12).

$$x_r = x_w + 2 \times (\bar{x} - x_w) \quad (12)$$

$\boldsymbol{iii}$. If cost of x_r is better than cost of x_w, replace x_w with x_r.
$\boldsymbol{iv}$. If x_r was not better than x_w, x_c is next point in search space(13).

$$x_c = {(\bar{x} + x_w)}/{2} \quad (13)$$

$\boldsymbol{v}$. Now if x_c is better than x_w, replace them. Else do as below:
$\boldsymbol{vi}$. Select a random new point in search space and replace it with x_w.
Repeat the **a** and **b**, s and r times respectively ($r, s \geq 1$).
The mentioned parameters for SCE in our simulations are as below:

$$\text{Npop} = 30, \text{P} = 3, m = 10, \text{q} = 5, \text{r} = 1, \text{s} = 15$$

For PSO, we have used the same parameters in [6] to do a comparsion with our results:

$$\text{Npop} = 40, \text{c}_1 = \text{c}_2 = 2,\ k_p, k_i, k_d \in (0, 30)$$

4. PID controller design and problem formulation

The PID controller is used to improve the dynamic response and reduce the steady-state error. The transfer function of a PID controller is described as (16) where k_p, k_i and k_d are the proportional, integral and derivative gains, respectively. A performance criterion in the time domain includes the overshoot

Table 2. The best PID parameters. Tow first rows are related to motor 2

	k_p	k_i	k_d	$M_p(\%)$	E_{ss}	$t_r(s)$	$t_s(s)$	cost
PSO [6]	26.4326	0.6244	2.3598	0	$7.3328E-005$	0.1500	0.2072	0.0798
SCE *motor 2*	56.5761	0.7303	3.2022	0	$5.9360E-006$	0.1130	0.1330	0.0548
SCE *motor 1*	57.3645	0.8573	3.1626	0	$6.0172E-006$	0.1094	0.1294	0.0532

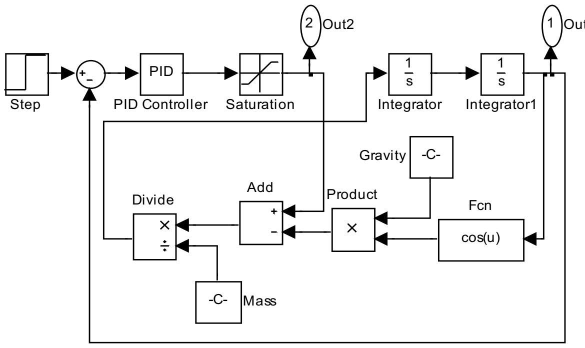

Fig. 3. Block diagram of manipulator with PID controller

M_p , rise time t_r, settling time t_s, and steady-state error E_{ss}. A good set of control parameters k_p, , k_i and k_d can make a good step response that will result in performance criteria minimization. To find the optimal PID controller parameters that minimize the performance indices on the load disturbance and transient responses a cost function has been defined as follows [12]:

$$f(K) = (1 - e^{-\beta})(M_p + E_{ss}) + e^{-\beta}(t_s - t_r) \quad (14)$$

that K is $[k_p, k_i, k_d]$ and β is the weighting factor that if set to be smaller than 0.7, reduces the rise time and settling time and if set to be larger than 0.7 reduces the overshoot and steady state error. It is so clear that minimizing each of four mentioned indices should result in minimizing the problem cost function. In function proposed in (14), the minus sign between t_s and t_r should be corrected. Because, an increase in t_r, that isn't desirable, results in cost function decrement that is desirable. This fault can mislead the algorithm. The reasonable way is to change cost function as below:

$$f(K) = (1 - e^{-\beta})(M_p + E_{ss}) + e^{-\beta}(t_s + t_r) \quad (15)$$

subject to:

- $k_p, k_i, k_d \in (-10,60)$.
- plant input signal swing $\in (-12, 12)$.

It is remarkable that in industrial or other applications that robots and manipulators are used, because of safety problems, the transient response of these devices must be so smooth. Therefore it is essential to choose relatively a large amount for weight of first term in the cost function to more emphasize on overshoot index. So in this paper we let $\beta = 1.5$.
PID controller transfer function is as follows:

$$G_c(s) = k_p + {}^{k_i}/_{s} + k_d s \quad (16)$$

With generating $Npop$ random amounts for three controller parameters in predefined search space as initial population, each point K (controller parameters) is sent to Matlab Simulink block and the values of four performance criteria in the time domain namely M_p, E_{ss}, t_r, and t_s, are calculated iteratively. Then cost function is evaluated for each point according to these performance criteria.

5. Simulation results

In this paper we have performed a comparsion between the simulation results of this research, which is based on SCE, and the approach proposed in [6]. The idea of this research is to use SCE to find a more optimum set of PID parameters in a wider search space and in the presence of the input signal saturation, which is a hard nonlinear constraint. Also the common cost function of the problem has been verified as (15). A typical block diagram of a 5-bar-linkage manipulator robot with PID controller is shown in Fig. 3, which seems drastically nonlinear. Also Fig. 4 plots the step response without the controller applied, and verifies that the system is in the boundary stability state. Table 2 contains PID controller parameters and step response indices of two methods, presented in [6] and in this paper. Studying this table indicates that our approach presents more optimum solutions with respect to [6]. For example in motor 2 the cost has decreased from 0.0798 to 0.0548 and also all of step response indices have been decreased. To have an intuitive view from the results, the step responses of motor 2 with implementation of both PSO and SCE algorithms are shown in Fig.5. Because of nonlinearity of the system it is not possible to extract reasonable rules about behavior of the controller parameters variations and step response indices, as we do in linear systems. This is the main reason that justifies use of artificial intelligence methods in finding near optimal solutions for such problems. Fig. 6 includes two plots that are related to plant input signals with and without saturation constraint. In unsaturated state response is faster but it has no good performance practically. All simulations have been done on

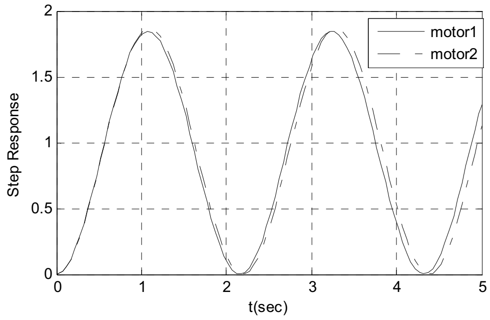

Fig. 4. Step response of the motors without PID controller

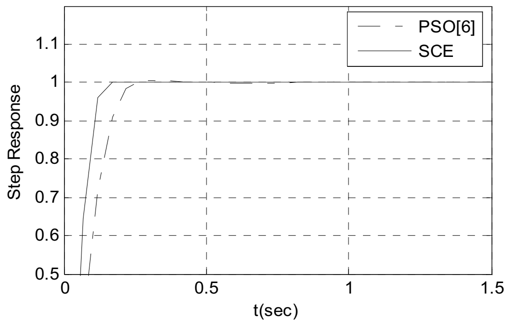

Fig. 5. Step response of motor 2 related to PSO and SCE

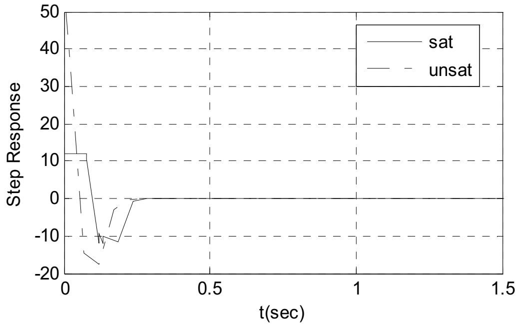

Fig. 6. Plant input signals in saturated and unsaturated states

motor 2 and the results for motor 1 are similar to motor 2. The real output of system directly depends on two mentioned angles that we were able to control them as the outputs of a decoupled MIMO system.

6. Conclusions

In this research we have defined a more precise model for a 5-bar linkage manipulator by adding a saturation block on its input signal. Also the common cost function of the optimization problem has been modified to lead the solutions toward lower costs better. Eventually by applying Shuffled Complex Evolution (SCE) algorithm to the optimization problem, we have gained much better PID controller parameters with respect to other recent methods. Design of a MIMO controller for the 5-bar-linkage manipulator without decoupling the dynamic equations can be the subject of the next researches.

7. References

[1] Y. T. Hsiao, C. L. Chuang, "Ant colony optimization for designing of PID controllers", in *IEEE International Symposium on comp. aided control systems design.*, Taipei., Taiwan., 2004.

[2] R.A. Krohling and J.P. Rey, "Design of optimal disturbance rejection PID controllers using genetic algorithms ", *IEEE Trans. Evol. Computer*, Vol.5, pp.78-82, Feb, 2001.

[3] A.Visioli, "Tuning of PID controllers with fuzzy logic", *IEE proc. Control theory appl.*, vol, 148, no. 1, pp.1-8, January, 2001.

[4] G.Zhou and J.D. Birdwell, "Fuzzy logic-based PID autotuner design using simulated annealing", in *proceedings of the IEEE/IFAC joint symposium, computer aided control system design*, 1994, pp.67-72.

[5] T.W. Kraus and T.J. Myron, "Self-tuning PID controller uses pattern recognition approach", in *Control Engineering*, 1984, pp.106-111.

[6] I.Hasanzadeh and S.Mobayen, "Optimum Design of PID Controller for 5-bar-linkage Manipulator Using Particle Swarm Optimization", In *proceeding of the 4th t International Symposium on Mechatronics and its Applications,* ISMA07, Sharjah, U.A.E., 2007.

[7] I. Hasanzadeh and S. Mobayen, "Tuning of PID controllers for a robot using binary genetic algorithm", *The 5th IFAC Intl. WS DECOM-TT*, Cesme, Turkey, 2007.

[8] J.P. Huissoon and D. Wang, "On the design of a 5-bar-linkage manipulator", *Robotica*, vol. 9, pp. 441-446, 1991.

[9] Kennedy, J. and Eberhart, R.C. "particle swarm optimization", in *proceedings of IEEE International Conference on Neural Networks.* Piscataway, NJ, IEEE service center, 1995, pp.1942-1948.

[10] Q. Y. Duan, V. K. Gupta, and S. Sorooshian, "Shuffled Complex Evolution approach for effective global minimization", *journal of optimization theory and application*: Vol. 76,No 3, March 1993.

[11] D.Wang, J.P.Huissoon and K.Luscott, "A teaching robot for demonstrating robot control strategies", *manufacturing research corporation of Ontario*, February 1993.

[12] Z. L. Gaing, "A Particle Swarm Optimization Approach For Optimum Design of PID controller in AVR system" , *IEEE Transactions on Energy Conversion*, vol. 19, no. 2, pp. 384-391, June 2004.