Bacterial Foraging Algorithm for Chaotic Synchronization with PID control

Masoud Sursrafil, Halimeh Noormohamadi, Majid Mohamadi, Mehdi Jafari Shahbazi

Abstract— In this paper, a PID (proportional-integral-derivative) controller is used for chaos synchronization for delayed discrete systems. The gains of PID controller can be determined using optimization algorithm that is called Bacterial Foraging Algorithm (BFA). The algorithm is motivated from a bacterium is called E.Coli that is live in our intestines. The E.coli bacterium has a control system (guidance system) that enables it to search for food and try to avoid noxious substances. The simulation results show the effectiveness of this algorithm for chaos synchronization.

Index Terms— PID (proportional-integral-derivative) controller, BFA (bacterial foraging algorithm), chemotaxes, reproduction and disoersal

———————————— ◆ ————————————

1 INTRODUCTION

For the past decade, more and more researchers related about chaotic systems have been proposed and explored.. The chaotic system is well-known for its highly nonlinear behavior.

The problem of chaos synchronization has attracted a widw range of research activity in recent years. A chaotic system has complex dynamical behaviors that posses some special features, such as being extremely sensitive to tiny variations of initial conditions, having bounded trajectories in the phase space [1]. The concept of chaos synchronization is making two or more chaotic systems oscillate in a synchronized manner. There are several methods which can be used to achieve chaos synchronization such as adaptive control [2], active control [2], linear feedback method [3], impulsive control [4], etc. In these methods, the Lyapunov theorem is used for the stability of the closed loop chaotic systems. In this paper, two systems (drive systems and response systems) are synchronized. The output of drive system is used to control the output of response system, so that they oscillate in a synchronized manner.

Recently, search and optimal foraging of Bacteria have been used for solving optimization problems such as: in optimal control engineering, harmonic estimation [5], neural network fuzzy learning[6], robust tuning of modern power system[7], optimal power stabilizers design [8]….. This algorithm also has used in global optimization [9], [10]. Alive beings need food for eternity. One of the beings is Escherichia coli (E. coli). Bacterial foraging algorithm (BFA), is inspired by the behavior of this bacterium that searching for food in human intestines.

———————————————

- *Author: Masoud Suresrafil, Departmaent of Electrical Engineering, Sirjan Branch, Science and Research, Islamic Azad University,Sirjan, Iran. Email:masood.suresrafil5@gmail.com*
- *Co-Author1: Halimeh Noormohamadi, Departmaent of Electrical Engineering, Anar Branch, Islamic Azad University, Anar, Iran*
- *Co-Author2: Majid Mohamadi, Department of computer, Shahid Bahonar University, Kerman, Iran*
- *Co-Author3: Mehdi Jafari Shahbazi, Department of Electrical Engineering, Kerman Branch, Islamic Azad University, Kerman, Iran*

The E. coli bacteria also undergo this foraging strategy.

Natural selection tends to eliminate animals with poor foraging strategies and favor the propagation of genes of those animals that have successful foraging strategies [11].

The characteristics of this bacterium are: The diameter is 1μm, the length is 2μm and under appropriate conditions can reproduce (split) in 20min. In suggested algorithm, a limited space (neighborhood) is considered for each bacterium. Then instead of searching food in universal space, searching is done in limited space.

In this paper, BFA is used for chaos synchronization. In the other words, the BFA algorithm is suitably introduced into the determination of three PID control gains for chaotic synchronization. Chaos synchronization of two identical delayed discrete systems discrete systems will be achieved by using the resulted PID controller.

The reminder of this paper is organized as follows: In section 2, problem statements for PID control is dscribed, section 3 introduces BFA(Bacterial Foraging Algorithm) and describes the steps of this algorithm. In section 4 the flochat of the BFA algorithm is described (the pseudo code of BFA algorithm is described. Simulation results for delayed chaotic system synchron are given and finally conclusion part is given in section 5.

2 PROBLEM STATEMENTS

2.1 PID controller

The PID controller (proportional-integral-control) is a generic control loop feedback mechanism widely used in industrial control systems. A PID controller calculates an error value as the difference between a measured process value and the desired setpoint. The controller attempts to minimize the error by adjusting the process control inputs.

The control law of PID controller is :

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau)d\tau + k_d \frac{d}{dt}e(t) \qquad (1)$$

Where e is the error signal, as discussed in the previous section, u is the PID control force, and k_p, k_i, k_d are the proportional, integral and derivative gains the be updated using BFA algorithm.

Because the system is discrete, the control law must be discrete, so trapezoidal approximation is used to obtain the discrete control law [12]:

$$u(k) = u(k-1) + k_p[e(k) - e(k-1)] +$$

$$k_i \frac{T_s}{2}[e(k) - e(k-1)] + k_d \frac{1}{T_s}[e(k) - $$

$$2e(k-1) + e(k-2)] \qquad (2)$$

Where T_s is the sambling period.

2.2 Nonlinear delayed discrete chaotic systems

Two identical delayed discrete chaotic systems are considered To be synchronized using the proposed PID control. The master master system is given by the following difference equation:

$$x(k+1) = x(k) - \frac{\delta}{m}x(k-m) + \frac{\varepsilon}{m}x^3(k-m) \qquad (3)$$

Where δ and ε are positive constants, m is the delay term, and x is the master state. The delayed discrete system admits decaying, oscillatory, and chaotic behavior relying on setting of system parameters. On the other hand, the corresponding slave system is described by [12] :

$$y(k+1) = y(k) - \frac{\delta}{m}y(k-m) + \frac{\varepsilon}{m}y^3(k-m) + u(k) \qquad (4)$$

Where y is the slave state and u is the external control force that adopts the PID control of (2). For two identical delayed discrete chaotic systems (3) and (4) without control u, the state trajectories of these two chaotic systems will quickly separate each other if their initial condtions are not the same. However, the state trajectories can approach synchronization for any initial conditions if an appropriate controller is utilized. Hence the purpose of this paper is to apply the BFA algorithm to find out the optimal PID control gains such that chaos synchronization for two identical delayed chaotic systems is achieved.

3 BFA ALGORITHM

An E. coli bacterium is shown in figure (1). The move of the E. coli is done with flagellum. This part is located at the end of this bacterium. An E. coli bacterium alternates between running and tumbling. Each flagellum is a left-handed helix configured so that as the base of the flagellum (i.e., where it is

connected   to the cell) rotates counterclockwise, as viewed from the free end of the flagellum looking toward the cell, it produces a force against the bacterium so it pushes the cell. You may think of each flagellum as a type of propeller. If a flagellum rotates clockwise, it will pull at the cell [8].
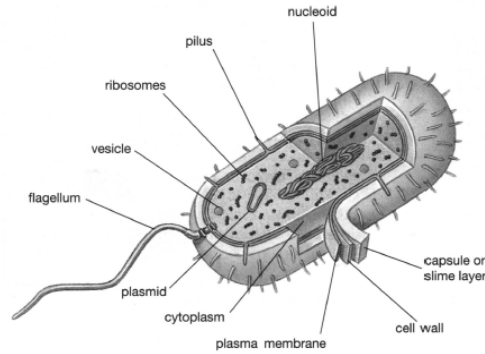

Fig.1 An E. coli Bacterium

Nutritious environments can separate in 3 parts: the neutral environment that the ratio of the nutrient substance is equal with the noxious substance, the bacterium alternatively tumble and run. In second environment the ratio of the nutrient substance is greater than the noxious substance, so E. coli more run and less tumble. In third environment the ratio the noxious substance is greater than the nutrient substance, so E. coli more tumble and less run. Figure (2) is shown the behavior of the chemotactic of E. coli bacterium. The optimization in BFA comprises the following process: chemotaxis, swarming, reproduction, elimination and dispersal. The chemotaxis is the activity that bacteria gathering to nutrient-rich area naturally [8]. A cell-to-cell communication mechanism is established to simulate the biological behavior of bacteria swarming. In reproduction step, the healthy bacteria are split in two parts, the reminder of bacteria are eliminated. In elimination and dispersal step, part of bacteria are killed and dispersed in random positions.

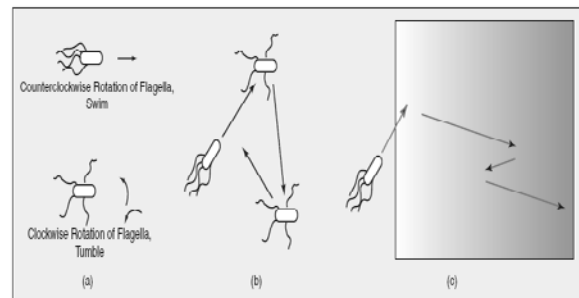
Fig. 2 The chemotactic behavior of E. coli

3.1. Chemotaxis Step

The process in the control system is achieved through Swimming and tumbling via flagellum [5]. To represent a tumble, a unit length random direction, Say $\varphi(j)$, this will be used to define the direction of movement after a tumble, then:

$$\theta^i(j+1,k,l)=\theta^i(j,k,l)+C(i)\phi(j) \tag{5}$$

Where $\theta^i(j,k,l)$ is the position of the ith bacterium at jth chemotaxis step, kth reproduction step and lth elimination and dispersal step.

$C(i)$ is the size of step taken in the random direction that specified by the tumble(run length unit). If the new position of bacterium $\theta^i(j+1,k,l)$ is better than the old position, then the bacterium will keep taking successive step in that direction. The number of iteration chemotaxis step is NC. And the maximum number of permissible successive steps is Ns [13].

3.2. Swarming Step

The bacteria in times of stresses release attractants to signal bacteria to swarm together. It however also releases a repellant to signal others to be at a minimum distance from it. Thus all of them will have a cell to cell attraction via attractant and cell to cell repulsion via repellant [9]. In BFA, each bacterium releases attractants to signal other, bacteria to swarm together and to keep a safe distance with each other. Such a mechanism to some extent suppresses the convergence because the bacteria in nutrient-poor areas attract the bacteria in nutrient-rich areas, which slows down the convergence speed of the entire group. The repelling effect among bacteria also prevents the group gathering together [8].

The mathematical representation for swarming can be represented by [9]:

$$J_{cc}(\theta)=\sum_{i=1}^{s} J_{cc}^i(\theta) \tag{6}$$

$$=\sum_{i=1}^{s}[-d_{attract} \ \exp(-w_{attract} \sum_{j=1}^{p} (\theta_j - \theta_j^i)^2 \ ]$$

$$+\sum_{i=1}^{s}[-d_{repellant} \ \exp(-w_{repellant} \sum_{j=1}^{p} (\theta_j - \theta_j^i)^2 \ ]$$

Where:

$d_{attract}$: depth of the attractant

$w_{attract}$: measure of the width of the attractant

$h_{repellant}$: height of the repellant effect

$w_{repellant}$: measure of the width of the repellant

p: number of parameters to be optimized

S: total number of bacteria

J_{cc}: the cost function to be added to the actual cost function to be minimized, to present a time varying cost function.

3.3. Reproduction

After NC chemotaxis step, a reproductive step is occurs. The fitness of bacteria is calculated, that is, during all chemotaxis steps:

$$J_{health} =\sum_{j=1}^{Nc} J(i,j,k,l) \tag{7}$$

Then this fitness is sorted in ascending order. The least healthy bacteria (with best cost function) die and the other bacteria split into two bacteria, thus the size of the population is constant. The number of this iteration of this step is Nre [13].

3.4. Elimination And Dispersal

The chemotaxis step provides a basis for local search, and the productive step speeds the convergence [9]. While to a large extent, only chemotaxis and reproduction are not enough for global optima searching. Then an elimination and dispersal event is necessary.

For each elimination and dispersal event each bacterium is eliminated with a probability P_{ed}, and dispersed them to a new environment. The selection of P_{ed}, play an important role in convergence of the algorithm.

If P_{ed} is large, the algorithm can degrade to random exhaustive search. If however, it is chosen appropriately, it can help the algorithm jump out of local optima and into a global optimum [13].

4. Pseudo Code For BFA

The algorithm is discussed here.

[step1]: Initialization

1. p: number of parameter that be optimized

2. S: the total number of bacteria

3. NC, Nre, Ned: the number of chemotaxis steps, the number of reproduction steps, the number of elimination and dispersal events, respectively.

4. Ns: the maximum number of permissible successive steps

5. The values of: $d_{attract}$, $w_{attract}$, $h_{repellant}$, $w_{repellant}$

6. P_{ed}: the probability of elimination and dispersal event

7. C (i) : the step size

8. NN: the number of bacteria that form neighborhood.

[step2]: Elimination and dispersal loop: l=l+1

[step3]: Reproduction loop: k=k+1

[step4]: chemotaxis loop: j=j+1

  [a]. for i = 1,2,…,N take a chemotaxis step for each bacterium i as follows:

  [b]. compute fitness function, $J(i,j,k,l)$

$$J(i,j,k,l)=J(i,j,k,l)+J_{cc} \qquad (8)$$

Let

J_{cc}, is calculated for the neighborhood that defined in section III.

[c]. let $J_{last} = J(i, j, k, l)$ to save this value since we may find a better cost via a run.

[d]. Tumble: generate a random vector $\Delta(i)$

$$\theta^i\ (j+1,k,l)=\theta^i\ (j,k,l)+c(i)\ \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \qquad (9)$$

that $1 \le \Delta(i) \le -1$

[e]. Move: Let

[f]. compute J (i, j+1, k, l) and let

$$J(i, j, k, l) = J(i, j, k, l) + J_{cc}$$

[g]. swim

i) Let m = 0(counter for swim length)

ii) While m<N_s

$$\theta^i\ (j+1,k,l)=\theta^i\ (j,k,l)+c(i)\ \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \qquad (9)$$

● Let m=m +1

● If J

(i, j+1, k, l) < J_{last} (if doing better), Let J_{last}=J (i, j+1, k, l) and

● Else let m = N_s

[h]. Go to next bacterium (i+1) if i ≠ N

[step5]. If j<NC, go to step 3.

[step6]. Reproduction

[a]. for the given k and l, and for each i = 1,2,…,N .

Let

$$J_{health} = \sum_{j=1}^{Nc} J(i, j, k, l)$$

Sort this fitness in order of ascending.

[b]. The Sr bacteria with the highest J_{health} values die, the remaining Sr bacteria with the best values split

[step7]. If k < Nre, go to step 3.

[step8]. Elimination and dispersal

For i=1,2,…,N with probability P_{ed}, eliminate and disperse each bacterium, and this result in keeping the number of bacteria in the population constant.

To do this, if a bacterium is eliminated, simply one to a random location on the optimization domain.

If l< Ned, then go to step 2; otherwise END [13].

5. Simulation results

This section demonstrates the effectiveness of BFA-based PID control design for synchronization of two identical delayed discrete chaotic systems. The Matlab software is used for simulation. Chaotic parameters in (3) , (4) are chosen as δ=0.3, ε=1, m=10 to exhibit chaotic behavior, d the initial condition are different as x(k)=0.5, y(k)=-0.5 for k=-m,-m+1,…,0. The sampling period T_s is set to 1 for simulating the non-linear delayed chaotic equations. In addition, the values assigned to the BFA algorithm are given by:

S=20   Number of Bacteria
D=3   Dimension
NC=50   Life time = Number of chemotaxis steps
Ns=4   Number of consecutive chemotaxis steps
NEL=0   Number of elemination steps
NRE=2   Number of reproduction steps

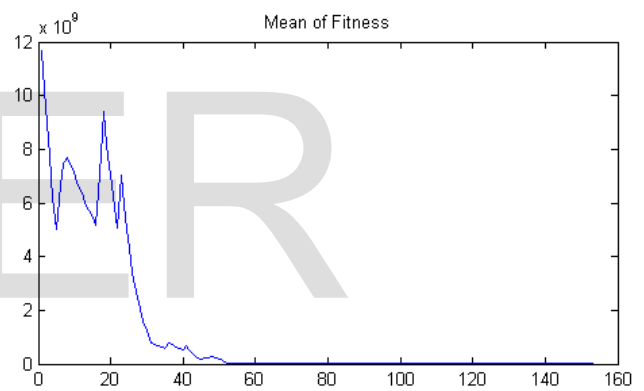The simulation results are shown by the following figures:
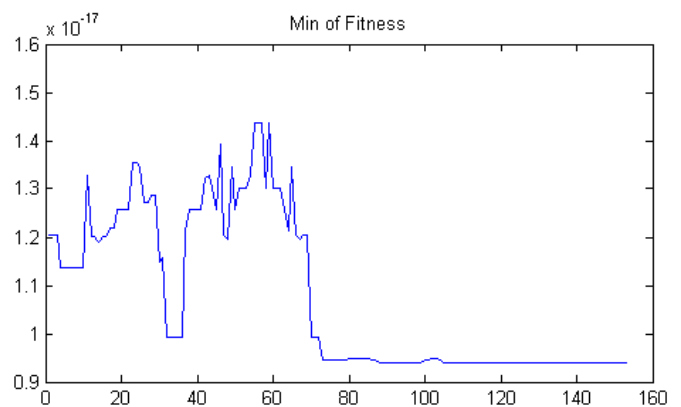

Fig.3 The mean of Fitness

Fig.4 The min of fitness

The above figures show that the mean and min of fitness are zero. In the other words, the figures reveal the effectiveness of

this algorithm, in the iteration less than 70 the cost function is zero. In figure 5 the error between two states x, y is brought. The figure shws the error is zero also.
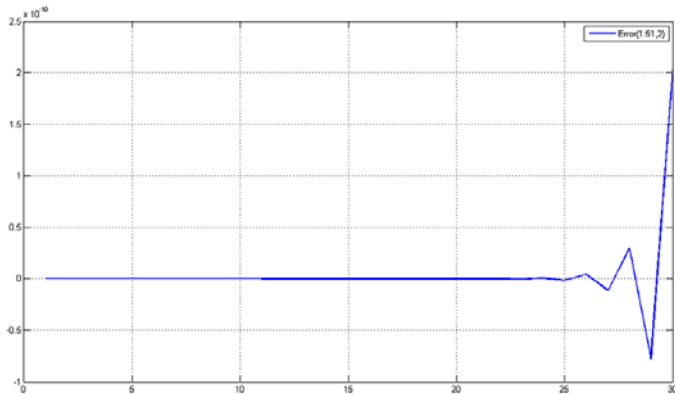

Fig. 5 The error Between states x, y

The parameters that are updates are given below:
K_p=0.8687, k_d=0.0483, k_i=0.3976.

6. Conclusion

In this paper a new algorithm for optimization problem is proposed. This algorithm is used for the optimal design of PID control for synchronization of two identical delayed discrete chaotic systems. Three PID control gains are updated by BFA algorithm. The simulation results reveal that this algorithm has good effect on this synchronization.

References

[1]   Thongchai Bomart, Piyapong Niamsup, Xinhi Liu, "Synchronization of Non-autonomous Chaotic Systems with Time-Varying Delay via Delayed Feed-back Control," *2010 11th Int. Conf. Control, Automation, Robotica and Vision Singapore*

[2]   T. Botmart and P.Niamsup, "*Adaptive control and synchronization perturbed Chuas systems,*" Math. Comput. Simulation, vol. 75, no.1-2, pp. 37-55,2007

[3]   T. William, Riccati Differntial Equations, Academic Press Inc., New York, 1972

[4]   J. Cao, D.W.C. Ho and Y. Yang, "Projective synchronization of a class of delayed chaotic systems via impulsive control," Phys. Lett. A, vol. 373,   no.35,   pp.3128-3133,   2009

[5]. S. Mishra, "A hybrid least square-fuzzy bacterial foraging strategy for harmonic estimation," *IEEE Transaction on Evolutionary Computation,*vol. 9, pp. 61–73, Feb. 2005.

[6]. Kim,D.H,Cho,C.H,"Bacterial Foraging Based Neural network fuzzy Learning".*IICAI* 2005,2030-2036

[7]. B. Sumanbabu, S. Mishra, B.K. Panigrahi, and G.K. Venayagamoorthy ", Robust Tuning of Modern Power System Stabilizers Using Bacterial Foraging Algorithm", IEEE2007

[8]. T. K. Das and G. K. Venayagamoorthy, "Bio-inspired algorithms for the design of multiple optimal power system stabilizers: SPPSO and BFA," in *IEEE Industry Application Society' s 41st Annual Meeting*, Tampa, USA, pp. 635–641, Oct. 2006

[9]. Arijit Biswas , Sambarta Dasgupta , Swagatam Das, and Ajith Abraham, " A Synergy of Differential Evolution and Bacterial Foraging Algorithm for Global Optimization",

[10]. W. J. Tang, Q. H. Wu,and J. R. Saunders," A Bacterial Swarming Algorithm For Global Optimization", IEEE2007

[11] K. M. Passino, "Bio mimicry of bacterial foraging for distributed optimizationand control", *IEEE Control Syst. Mag.*, vol. 22, no. 3, pp.52–67, Jun. 2002.

[12] Wie-Der Chang, "PID control for chaotic synchronization using particle swarm optimization", chaos, Solitons and fractals 39 (2009) 910-917

[13] H.Noormohamadi, A.A.Gharaveisi, M.Suresrafil, "A novel Bacterial Foraging algorithm for optimization problem", Advances in Computer Science and its Applications, Vol. 1, No. 1, March 2012