UKACC International Conference on Control 2012
Cardiff, UK, 3-5 September 2012

A Novel Hybrid Spiral Dynamics Bacterial Chemotaxis Algorithm for Global Optimization with Application to Controller Design

A. N. K. Nasir, M. O. Tokhi and N. M. Abd Ghani
Department of Automatic Control & Systems Engineering, University of Sheffield, Sheffield, UK.
cop11ann@sheffield.ac.uk

M. A. Ahmad
Faculty of Electrical & Electronics Engineering, Universiti Malaysia Pahang (UMP), 26600 Pekan, Pahang, Malaysia.

Abstract—**This paper presents a hybrid optimization algorithm, referred to as hybrid spiral dynamics bacterial chemotaxis (HSDBC) algorithm. HSDBC synergizes bacterial foraging algorithm (BFA) chemotaxis strategy and spiral dynamics algorithm (SDA). The original BFA has higher convergence speed while SDA has better accuracy and stable convergence when approaching the optimum value. This hybrid approach preserves the strengths of BFA and SDA and thus has the capability of producing better results. Moreover, it has simple structure, hence reduced computational cost. Several unimodal and multimodal benchmark functions are employed to test the algorithm in finding the global optimum point. Furthermore, the proposed algorithm is tested in the design of PD controller for a flexible manipulator system. The results show that the HSDBC outperforms SDA and BFA in all test functions and successfully optimizes the PD controller.**

Keywords-Spiral dynamics; bacterial chemotaxis; optimization algorithm; PD control; flexible manipulator.

I. INTRODUCTION

Metaheuristic optimization algorithms have gained a lot of interest by many researchers worldwide. These algorithms are inspired by biological phenomena or natural phenomena. Some of the newly introduced algorithms include biogeography-based optimization (BBO) [1], firefly optimization algorithm [2], cuckoo search optimization [3], galaxy-based search algorithm [4], and spiral dynamics inspired optimization (SDA) [5]. All these algorithms have gained attention due to their simplicity to program, fast computing time, easy to implement, and possibility to apply to various applications. Each of these algorithms has its own unique features, advantages and also disadvantages. Therefore, there are a lot of possibilities to improve the algorithms from various aspects. Many attempts have been made to improve performances of the algorithms such as developing adaptive approaches or incorporating powerful mathematical functions into the algorithms and mostly hybridizing two or more algorithms.

Hybridisation is a common approach used in metaheuristic to enhance capability of optimization algorithms. It may reduce computational cost by making a simple and better structure to lead to higher performance. Moreover, with the rapidly emerging computing tools and efficiency in current technology, hybrid approaches have become increasingly popular to explore. Various combinations of optimization algorithms have been considered by researchers with the aim to increase system performance. [6] developed a hybrid optimization algorithm combining bacterial foraging optimisation algorithm (BFA) with BBO, and referred to it as intelligent biogeography-based optimization. In the algorithm, chemotaxis behaviour of bacteria is adopted into BBO migration process to determine a valid emigration of an individual from one place to another. This ensures the island that receives the emigrated solution preserves its fitness level by only accepting individuals that contribute to a better fitness value. [7] introduced hybrid version of BFA with differential evolution (DE) algorithm called chemotaxis differential evolution. In the algorithm, chemotaxis strategy of bacteria is combined with the mutation process in DE. [8], [9] and [10] introduced hybrid GA-BF algorithm employing modified mutation and crossover operation in GA while applying variation bacterial chemotaxis step size in BFA. [11] developed cooperative (BF-TS) by combining adaptive bacterial foraging optimization algorithm (ABFA) and adaptive tabu search (ATS). With limited exploration capability of ATS in the search space and complexity of ABFA, the chemotaxis strategy of ABFA is incorporated into ATS to provide suitable exploration at the early stage. On the other hand, [12] used hybrid ABFA and ATS called BTSO, to analyze Lyapunov's stability of linear and nonlinear systems. [13] introduced a hybrid algorithm namely BPSO-DE synergizing BFA, particle swarm optimization (PSO), and DE to solve dynamic economic dispatch problem with valve-points effect. Bacterial chemotaxis strategy with adaptive step-size in BFA is used to perform local search to enhance exploitation while PSO-DE features containing evolutionary operators and velocity update equation are used to perform exploration search over the entire search space. Hybrid BFA and PSO on the other hand, has received the most attention. [14], [15], [16], [17] and [18] employed velocity and position update equation in PSO to act as global search method while utilizing chemotaxis strategy in BFA to serve as local search method. [19] introduced simplified version of BFA employing bacterial chemotaxis strategy and PSO velocity update equation to solve parameter identification problem of heavy oil thermal cracking model. Reproduction and elimination stages were omitted to reduce computational time.

This paper presents hybrid version of bacterial foraging algorithm (BFA) chemotaxis strategy and spiral dynamics

978-1-4673-1558-6/12/$31.00 ©2012 IEEE

algorithm (SDA). The rest of the paper is organized as follows. Section II provides a brief literature review of the original BFA and spiral dynamics inspired optimization. The proposed HSDBC is described in section III. Validation of the proposed HSDBC in comparison to SDA and BFA with uni-modal and multi-modal test functions as well as application of the algorithm in optimizing a PD controller is presented in section IV. Section V presents concluding remarks.

II. BFA AND SDA

The original versions of BFA and SDA are briefly described in this section.

A. *Bacterial foraging optimization algorithm*

The BFA is a biologically inspired algorithm introduced in [20]. It is based on adaptation technique of Escherichia Coli (E. Coli) bacteria to find nutrient or food source during their lifetime or alternatively the technique might be called bacterial foraging strategy. Furthermore, E. Coli bacteria use saltatory search technique, which is the combination of cruise and ambush movement. One of the exceptional features of E. Coli is that it has very high growth rate, which is normally exponential. This extraordinary capability of E. Coli has motivated researchers to adopt the strategy as optimization technique. Bacterial foraging strategy consists of three basic cycles namely chemotaxis, reproduction and elimination & dispersal. These cycles are continuing processes and very effective for optimization purposes [21]. Moreover, it offers flexibility for researchers to manipulate the strategy according to a specific application area. When searching for food or nutrient, tumbling and swimming will take place. Tumbling is similar to cruise and it happens when the E. Coli navigates in the search area and once the food source is found, it swims like ambushing a target area with great speed, up to 20μm/s or faster in a rich nutrient medium. This unique movement is called chemotaxis. Reproduction, elimination and dispersal events then happen to bacteria with high fitness or healthier that has capability to reach food source accurately and quickly. The strength of BFA lies in the bacterial chemotaxis strategy adopted by many researchers to improve the optimization algorithm. The details of the original algorithm and pseudocode of BFA can be found in [20]. In this paper, number of bacteria, number of chemotaxis, chemotactic step size, number of swims, number of reproduction, number of elimination & dispersal are represented as *S, Nc, C, Ns, Nre* and *Ned* respectively. The probability that each bacterium will be eliminated and dispersed is defined as *ped* = 0.25 for the problems considered.

B. *Spiral dynamics inspired optimization algorithm*

The SDA is another metaheuristic algorithm adopted from spiral phenomena in nature [5]. 2-dimensional and N-dimensional [5] logarithmic spiral discrete models have been tested on several benchmark functions. Moreover, comparisons with other optimization algorithms such as PSO and DE have shown that SDA performance is either better or the same as those [5]. This simple and effective strategy retains the diversification and intensification at the early phase and later phase of the trajectory as diversification and intensification are important features of the optimization algorithm. At the early stage, the spiral trajectory explores a wider search space and it continuously converges with a smaller radius providing dynamics step size when approaching the final point, which is the best solution, located at the centre. The distance between a point in a path trajectory and the centre point is varied constantly if the radius of the trajectory is changing at constant rate thus making the radius an important converging parameter for the algorithm. The strength of SDA lies in its spiral dynamics model. An n-dimensional spiral mathematical model that is derived using composition of rotational matrix based on combination of all 2 axes is given as:

$$x(k+1) = S_n(r,\theta)x(k) - (S_n(r,\theta) - I_n)x^* \qquad (1)$$

where

$$S_n(r,\theta)x(k) = rR^n(\theta_{1,2},\theta_{1,3},\ldots,\theta_{n-1,n})x(k).$$

or

$$S_n(r,\theta)x(k) = \prod_{i-1}^{n-1}\left(\prod_{j=1}^{i}(R^n_{n-i,n+1-j}(\theta_{n-i,n+1-j}))\right)$$

and $R^n_{i,j}(\theta_{i,j}) :=$

$$\begin{array}{cc}
 & \begin{array}{ccccccccccc} & & & i & & & & j & & & \end{array} \\
\begin{array}{c} \\ \\ \\ i \\ \\ \\ \\ j \\ \\ \\ \\ \end{array} &
\left[\begin{array}{ccccccccccc}
1 & & & & & & & & & & \\
 & \ddots & & & & & & & & & \\
 & & 1 & & & & & & & & \\
 & & & \cos\theta_{i,j} & & \cdots & & -\sin\theta_{i,j} & & & \\
 & & & & 1 & & & & & & \\
 & & & \vdots & & \ddots & & \vdots & & & \\
 & & & & & & 1 & & & & \\
 & & & \sin\theta_{i,j} & & \cdots & & \cos\theta_{i,j} & & & \\
 & & & & & & & & 1 & & \\
 & & & & & & & & & \ddots & \\
 & & & & & & & & & & 1
\end{array}\right]
\end{array}$$

Parameters and descriptions used in equation (1) are similar to those used in HSDBC optimization algorithm, which are shown in Table 1. Since SDA is relatively new, not much work in the literature involving the algorithm has been reported. The details of the original SDA algorithm for 2-dimension and n-dimension can be found in [5]. The hybrid approach of this algorithm and its details are provided in the next section.

III. HYBRID SPIRAL DYNAMICS BACTERIAL CHEMOTAXIS ALGORITHM

The HSDBC is a combination of bacterial chemotaxis strategy used in BFA and SDA. BFA has faster convergence speed due to the chemotaxis approach but suffers from oscillation problem towards the end of its search process. On the other hand, SDA provides better stability when approaching optimum point due to dynamic spiral step in its trajectory motion but has slower convergence speed. HSDBC algorithm preserves the strengths possessed by BFA and SDA. Moreover, by incorporating only chemotaxis part of BFA simple structure of SDA can be retained, thus reducing computational time and enhancing performance of the algorithm. The parameters and description used in n-dimensional HSDBC optimization algorithm are presented in Table 1 and the algorithm is shown in Fig. 1.

TABLE I. PARAMETERS FOR HSDBC OPTIMIZATION ALGORITHM

Symbols	Description
$\theta_{i,j}$	Bacteria angular displacement on $x_i - x_j$ plane around the origin
r	Spiral radius
m	Number of search points
$k_{\max}$	Maximum iteration number
N_s	Maximum number of swim
$x_i(k)$	Bacteria position
R^n	n x n matrix

An n-dimensional hybrid spiral dynamics bacteria chemotaxis optimization algorithm.

Step 0: Preparation

Select the number of search points (bacteria) $m \geq 2$, parameters $0 \leq \theta < 2\pi, 0 < r < 1$ of $S_n(r,\theta)$, maximum iteration number, $k_{\max}$ and maximum number of swim, N_s for bacteria chemotaxis. Set $k = 0,\ s = 0$.

Step 1: Initialization

Set initial points $x_i(0) \in R^n$, $i = 1,2,...m$ in the feasible region at random and center x^* as $x^* = x_{i_g}(0)$, $i_g = \arg\min_i\ f(x_i(0))$, $i = 1,2,...,m$.

Step 2: Applying bacteria chemotaxis

(i) Update x_i

$$x_i(k+1) = S_n(r,\theta)x_i(k) - (S_n(r,\theta) - I_n)x^*$$
$$i = 1,2,...,m.$$

(ii) Bacteria swim

(a) Check number swim for bacteria *i*.

If $s < N_s$, then check fitness,

Otherwise set $i = i+1$, and return to step (i).

(b) Check fitness

If $f(x_i(k+1)) < f(x_i(k)$, then update x_i,

Otherwise set $s = N_s$, and return to step (i).

(c) Update x_i

$$x_i(k+1) = S_n(r,\theta)x_i(k) - (S_n(r,\theta) - I_n)x^*$$
$$i = 1,2,...,m.$$

Step 3: Updating x^*

$x^* = x_{i_g}(k+1)$,

$i_g = \arg\min_i\ f(x_i(k+1))$, $i = 1,2,...,m$.

Step 4: Checking termination criterion

If $k = k_{\max}$ then terminate. Otherwise set $k = k+1$, and return to step 2.

Figure 1. HSDBC optimization algorithm.

In the proposed hybrid approach, bacterial chemotaxis strategy is employed in step 2 to balance and enhance exploration and exploitation of the search space. The bacteria move from low nutrient location towards higher nutrient location, placed at the centre of a spiral. The most important factor of HSDBC algorithm is the respective diversification and intensification at the early phase and later phase of the spiral motion. In the diversification phase, bacteria are located at low nutrient location and move with larger step size thus producing faster convergence. On the other hand, in the intensification phase, bacteria are approaching rich nutrient location and move with smaller step size hence avoiding oscillation around the optimum point. Another factor contributing to better performance of the algorithm is the swimming action in bacterial chemotaxis. Bacteria continuously swim towards optimum point if the next location has higher nutrient value compared to previous location until the maximum number of swim is reached.

IV. VALIDATION TEST AND RESULTS

In this section, the proposed algorithm is validated through simulation tests on two 3-dimensional uni-modal and two 2-dimensional multi-modal benchmark functions. Moreover, the HSDBC algorithm is tested in optimizing PD controller of a flexible manipulator system. Comparison with the original version of SDA and BFA tested on the four benchmark functions is also given to show the improved performance of HSDBC. The parameters used in the simulation are chosen heuristically for all test functions.

A. *Uni-modal sphere function*

The sphere function is defined as:

$$f(x) = \sum_{i=1}^{n} x_i^2 \tag{2}$$

The function has a global minimum at x_i = [0, 0, 0] with fitness *f(x)* = 0. In this simulation, the sphere function is considered to have dimension n = 3 and variable x_i is in the range [–5.12, 5.12]. Number of search points, m = 30, iteration number, 80, angular displacement, θ = π/4, and spiral radius, *r* = 0.96 were used for both algorithms. Number of swims with for HSDBC was defined as N_s = 5. BFA parameters for this function were *S* = 30, *Nc* = 30, *C*=0.01, *Ns* = 4, *Nre* = 4 and *Ned* = 2. The convergence plot for 3 dimensional sphere function thus achieved is shown in Fig 2.

B. *Uni-modal Ackley function*

The Ackley function is mathematically defined as:

$$f(x) = -20\exp(-0.2\sqrt{(\frac{1}{n}\sum_{i=1}^{n} x_i^2)} - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i) + 20 + e \tag{3}$$

The function has a global minimum at x_i = [0, 0, 0] with fitness *f(x)* = 0. The Ackley function is considered with dimension n = 3 and variable x_i in the range [–32.768, 32.768]. Number of search points, m = 30, iteration number

200, angular displacement, θ = π/4, and spiral radius, r = 0.96 were used in both algorithms. Number of swims for HSDBC with swim radius, r = 0.6 was defined as N_s = 1. BFA parameters for this function were S = 20, Nc = 20, C = 0.02, Ns = 4, Nre = 4 and Ned = 2.The resulting convergence plot for 3-dimension Ackley function is shown in Fig 3.

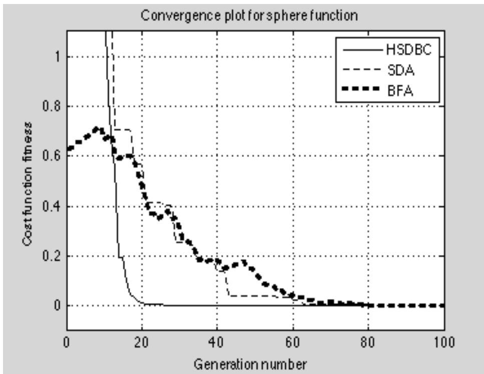

Figure 2. Convergence plot for 3D sphere function.

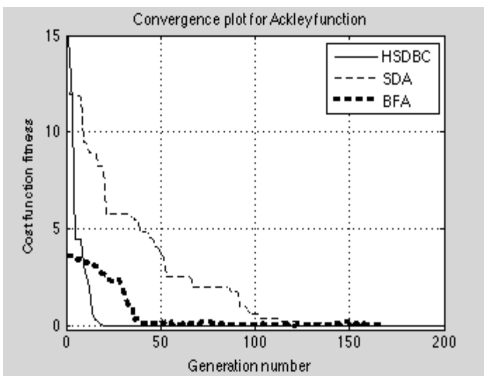

Figure 3. Convergence plot for 3D Ackley function.

C. *Multi-modal Rastrigin function*

The Rastrigin function is defined as:

$$f(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10] \quad (4)$$

The function has a global minimum at x_i = [0, 0] with fitness $f(x)$ = 0. The Rastrigin function is considered with dimension n = 2 and variable x_i in the range [–5.12, 5.12]. The number of search points, m = 50, iteration number 120, angular displacement, θ = π/4, and spiral radius, r = 0.96 were used in both algorithms. Number of swims for HSDBC with swim radius, r = 0.65 was defined as N_s = 2. BFA parameters for this function were S = 30, Nc = 20, C = 0.01, Ns = 4, Nre = 4 and Ned = 2. The resulting convergence plot for the 2-dimensional Rastrigin function is shown in Fig 4.

D. *Multi-modal Griewank function*

The Griewank function is defined as:

$$f(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (5)$$

The function has a global minimum at x_i = [0, 0] with fitness $f(x)$ = 0. The Griewank function was considered with dimension n = 2 and variable x_i in the range [–600, 600]. The number of search points, m = 50, iteration number 200, angular displacement, θ = π/4, and spiral radius, r = 0.96 were used for both algorithms. Number of swims for HSDBC with swim radius, r = 0.55 was defined as N_s = 1. BFA parameters for Griewank function were S = 30, Nc = 10, C = 0.1, Ns = 4, Nre = 4 and Ned = 2. The resulting convergence plot for the 2-dimensional Griewank function is shown in Fig 5.

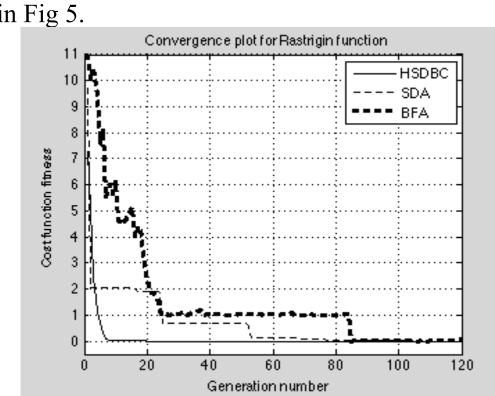

Figure 4. Convergence plot for 2D Rastrigin function.

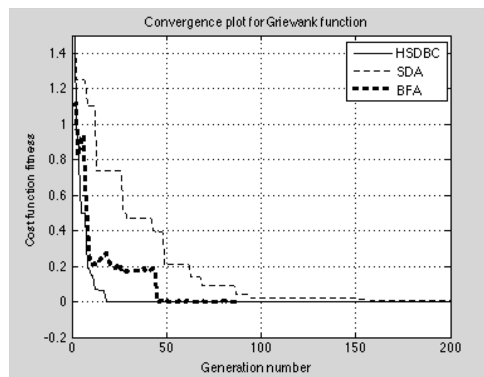

Figure 5. Convergence plot for 2D Griewank function.

It can be clearly seen in the plots, in Figures 2-5 that the HSDBC outperformed SDA and BFA in terms of convergence speed and improved accuracy. Numerical results of HSDBC, SDA and BFA performance tests with the benchmark functions are shown in Tables II, III and IV respectively. It is noted that HSDBC has achieved better performance than SDA and BFA with the test functions in terms of convergence speed and accuracy.

TABLE II. HSDBC PERFORMANCE ON BENCHMARK FUNCTIONS

Cost Function Name	Performance				
	Best fitness	*Converge time (iter)*	X_1	X_2	X_3
Sphere	6x10^{-7}	26	2x10^{-4}	6x10^{-7}	-7x10^{-4}
Ackley	3x10^{-7}	20	1x10^{-7}	2x10^{-8}	-3x10^{-8}
Rastrigin	0	15	-2x10^{-9}	4x10^{-10}	-
Griewank	2x10^{-11}	18	-3x10^{-6}	6x10^{-6}	-

TABLE III. SDA PERFORMANCE ON BENCHMARK FUNCTIONS

Cost Function Name	Performance				
	Best fitness	*Converge time (iter)*	X_1	X_2	X_3
Sphere	$5x10^{-3}$	63	$-4x10^{-2}$	$-5x10^{-2}$	$-7x10^{-3}$
Ackley	$6x10^{-3}$	159	$9x10^{-4}$	$2x10^{-5}$	$-2x10^{-3}$
Rastrigin	$1x10^{-6}$	84	$-8x10^{-5}$	$-2x10^{-5}$	-
Griewank	$7x10^{-5}$	91	$-6x10^{-3}$	$-1x10^{-2}$	-

TABLE IV. BFA PERFORMANCE ON BENCHMARK FUNCTIONS

Cost Function Name	Performance				
	Best fitness	*Converge time (iter)*	X_1	X_2	X_3
Sphere	$5x10^{-5}$	84	$4x10^{-3}$	$-4x10^{-3}$	$3x10^{-3}$
Ackley	$2x10^{-2}$	40	$-5x10^{-3}$	$-9x10^{-3}$	$-1x10^{-3}$
Rastrigin	$5x10^{-4}$	85	$-8x10^{-4}$	$-1x10^{-3}$	-
Griewank	$7x10^{-4}$	45	$-1x10^{-2}$	$4x10^{-2}$	-

E. Controller design optimization

The HSDBC algorithm is employed here to optimize PD controller of a flexible manipulator system (FMS). Schematic diagram of the flexible manipulator system is shown in Fig. 6. X_oOY_o and XOY represent the stationary and moving coordinate frames respectively. τ represents the applied torque at the hub. Young modulus, area moment of inertia, mass density per unit volume, cross-sectional area, hub inertia, displacement and hub angle of the manipulator are represented by E, I, ρ, A, I_h, $v(x,t)$ and $\theta(t)$ respectively [23].

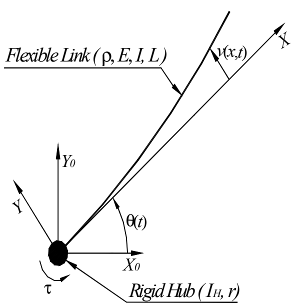

Figure 6. Schematic diagram of flexible manipulator system.

Mathematical model of FMS adopted here is that derived using Lagrange method in [22]. The FMS model has been used by many researchers in testing various types of controller for flexible systems [23], [24]. The dynamic equation of motion of FMS can be represented as:

$$M\ddot{Q}(t) + D\dot{Q}(t) + KQ(t) = F(t) \quad (6)$$

where M, D and K are mass, damping and stiffness matrices respectively. $F(t)$ and $Q(t)$ are vectors of external forces and modal displacement respectively;

$$F(t) = [\ \tau \quad 0 \quad 0 \quad \cdots \quad 0\ ]^T \quad (7)$$

$$Q(t) = [\ \theta \quad q_1 \quad q_2 \quad \cdots \quad q_n\ ]^T = [\ \theta \quad q^T\ ]^T \quad (8)$$

More details of the derivation and parameters of FMS can be found in [22], [23]. A state-space model of FMS is obtained by linearizing (6) and it is used to design PD controller through HSDBC. The control strategy of FMS is adopted from [23] and [24] where PD feedback of collocated sensor signals is employed. A block diagram of the control structure is shown in Fig. 7, where K_p, K_v and A_c are the proportional, derivative and motor amplifier gains respectively. The input of the system is reference hub angle, R_f and the outputs of the system are hub angle, θ and hub angle velocity, $\dot{\theta}$. In this simulation, number of search points, m = 30, iteration number 100, angular displacement, θ = π/4, and spiral radius, r = 0.96, and number of swim N_s = 3 were used to optimize the PD controller.

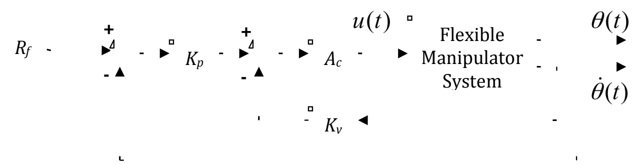

Figure 7. Collocated PD control structure of FMS.

Integral square error (ISE) of hub angle was chosen as cost function for the optimization algorithm. As a means of examining the proposed algorithm, this paper is only dealing with step input tracking capability of FMS. Step input was defined to have final value at 0.8 radians, which is the final location of hub angle. Graphical plot of the hub angle achieved is shown in Fig. 8.

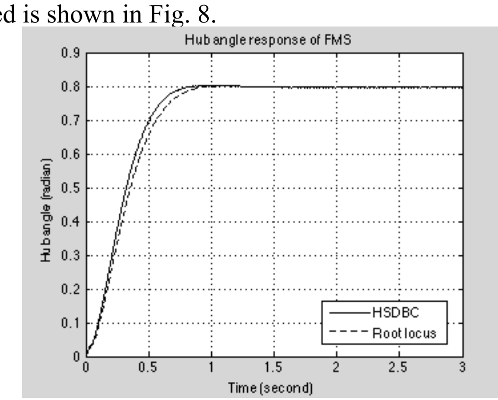

Figure 8. Hub angle response of FMS.

In this plot, for the purpose of comparison, the hub angle response with PD controller designed using root locus approach from [24] is also shown. Simulation with HSDBC optimization algorithm on the FMS gave K_p =72.3459 and K_v =20.6227 while K_p =60 and K_v =19 using root locus technique [24]. It is clear from Fig. 8, that hub angle response of FMS using HSDBC was better than hub angle response using root locus technique in terms of speed of response. Numerical results of the hub angle response are shown in Table V. It is noted that the HSDBC approach resulted slightly larger overshoot within acceptable range. However, the response rise time with HSDBC was better, which indicates that the algorithm can perform faster with satisfactory response overshoot and no error at steady state.

TABLE V. PERFORMANCE SPECIFICATION OF HUB ANGLE RESPONSE

Tuning Method	Performance Specification			
	Overshoot, %os (%)	*Settling time, ts (s)*	*Rise time, tr (s)*	*Steady state error, ess*
HSDBC	0.84	1.47	0.44	0
Root locus	0.53	1.47	0.50	0

V. CONCLUSION

A novel hybrid spiral dynamics bacterial chemotaxis optimization algorithm has been proposed. Validation with uni-modal and multi-modal benchmark functions and comparison with standard SDA and BFA have been carried out. Moreover, the HSDBC has been used in controller design of a flexible manipulator in comparison with root locus design approach. Simulation results have shown that the proposed algorithm outperformed its counterpart in all test functions and it successfully optimized PD controller of flexible manipulator system in terms of convergence speed and accuracy.

ACKNOWLEDGMENT

Ahmad Nor Kasruddin Nasir is on study leave from Faculty of Electrical & Electronics Engineering, Universiti Malaysia Pahang (UMP) and is sponsored by Ministry of Higher Education Malaysia.

REFERENCES

[1] D. Simon, Biogeography-based optimization, IEEE Transaction on Evolutionary Computation, 12(16), December 2008, pp. 702-713.

[2] X. S. Yang, "Firefly algorithm, levy flight and global optimization," Research and Development in Intelligent systems XXVI, Springer London, 2010, pp. 209-218.

[3] X. S. Yang, and S. Deb, “Cuckoo search via levy flights,” Proceeding of World Congress on Nature & Biologically Inspired Computing (Nabic 2009), December 2009, India. IEEE Publication, USA, pp. 210 - 214.

[4] H. S. Hosseini, “Principal components analysis by the galaxy-based search algorithm, a novel metaheuristic for continuous optimization,” International Journal of Computational Science and Engineering, **6**(1/2), 2011, pp. 132 - 140.

[5] K. Tamura, and K. Yasuda, “Primary study of spiral dynamics inspired optimization,” IEEJ Transactions on Electrical and Electronic Engineering, **6**(S1), 2011, pp. 98 – 100.

[6] M. R. Lohokare, S. S. Pattnaik, S. Devi, B. K. Panigrahi, S. Das, K. M. Bakwad, “Intelligent biogeography-based optimization for discrete variables,” Proceeding of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), Coimbatore, India, 9 – 11 December, 2009, pp. 1088 – 1093.

[7] A. Biswas, S. Dasgupta, S. Das, A. Abraham, “A synergy of differential evolution and bacterial foraging optimization for global optimization,” Journal of Neural Network World, **17**(6), 2007, pp. 607 – 626.

[8] D. H. Kim, A. Abraham, J. H. Cho, “A hybrid genetic algorithm and bacterial foraging approach for global optimization,” Journal of Information Sciences, **177**, 2007, pp. 3918 – 3937.

[9] T. C. Chen, P. W. Tsai, S. C. Chu, J. S. Pan, “A novel optimization approach: bacterial-ga foraging,” Proceedings of the Second International Conference on Innovative Computing, Information and Control ICICIC '07, Kumamoto, Japan, September 5 - 7, 2007, pp. 391 – 394.

[10] D. G. Jadhav, S. S. Pattnaik, S. Devi, M. R. Lohokare, K. M. Bakwad, “Approximate memetic algorithm for consistent convergence,” Proceeding of National Conference on Computational Instrumentation (NCCI 2010), Chandigarh, India, 19-20 March 2010, pp. 118 – 122.

[11] N. Sarasiri, S. Sujitjorn, “Bacterial foraging optimization and tabu search: performance issues and cooperative algorithms,” Proceedings of the 10th WSEAS International Conference on Systems Theory and Scientific Computation, Taipei Taiwan, August 20-22, 2010, pp. 186 – 191.

[12] S. Panikhom, N. Sarasiri, S. Sujitjorn, “Hybrid bacterial foraging and tabu search optimization (BTSO) algorithms for Lyapunov’s stability analysis of nonlinear systems,” International Journal of Mathematics and Computers in Simulation, **3**(4), 2010, pp. 81 – 89.

[13] P. Praveena, K. Vaisakh, S.R.M. Rao, “A bacterial foraging and pso-de algorithm for solving dynamic economic dispatch problem with valve-point effects,” First International Conference on Integrated Intelligent Computing (ICIIC), Bangalore, India, 5-7 Aug. 2010, pp. 227 – 232.

[14] A. Biswas, S. Dasgupta, S. Das, A. Abraham, “Synergy of pso and bacterial foraging optimization – A comparative study on numerical benchmarks. Springer-Verlag Berlin Heidelberg, 2007, pp. 255 – 263.

[15] Y. Chu, H. Mi, H. Liao, Z. Ji, Q. H. Wu, “A fast bacterial swarming algorithm for high-dimensional function optimization,” Proceeding of 2008 IEEE Congress on Evolutionary Computation (CEC 2008), 1-6 June 2008, Hong Kong, pp. 3135 – 3140.

[16] W. M. Korani, “Bacterial foraging oriented by particle swarm optimization strategy for PID tuning,” Proceedings of the 2008 GECCO conference companion on Genetic and evolutionary computation, Atlanta, GA, USA, July 12-16, 2008, pp. 1823 – 1826.

[17] H. Shen, Y. Zhu, X. Zhou, H. Guo, C. Chang, “Bacterial foraging optimization algorithm with particle swarm optimization strategy for global numerical optimization,” Proceedings of 2009 World Summit on Genetic and Evolutionary Computation, Shanghai, China, June 12 – 14, 2009, pp 497 – 504.

[18] R. A. Hooshmand, M. E. Pour, “Corrective action planning considering FACTS allocation and optimal load shedding using bacterial foraging oriented by particle swarm optimization algorithm,” Turkish Journal of Electrical Engineering and Computer Sciences, **18**(4), 2010, pp. 597 – 612.

[19] Y. Fang, Y. Liu, J. Liu, “A novel simplified foraging optimization algorithm for parameter identification of nonlinear system model,” Proceedings of the IEEE International Conference on Automation and Logistics, Jinan, China, August 18 - 21, 2007, pp. 798 – 802.

[20] K. M. Passino, “Biomimicry of bacterial foraging for distributed optimization and control,” IEEE Control System Magazine, June 2002, pp. 52 - 67.

[21] B. Niu, H. Wang, L. J. Tan, J. Xu, “Multi-objective optimization using BFO algorithm,” Springer-Verlag Berlin Heidelberg, 2012, pp. 582 - 587.

[22] J. M. Martins, Z. Mohamed, M. O. Tokhi, J. Sa da Costa and M. A. Botto, “Approaches for dynamic modeling of flexible manipulator system,” IEE Proceedings-Control theory and Application, vol. **150**(4), 2003, pp. 401 - 411.

[23] M.A. Ahmad, A. N. K. Nasir, R.M.T Raja Ismail, and M. S. Ramli, “Comparison of hybrid control schemes for vibration suppression of flexible robot manipulator,” International Conference on Computer Modeling and Simulation 2009 (ICCMS ‘09), Macau, China, February 20- 22, 2009, pp. 356 – 360.

[24] Z. Mohamed and M. A. Ahmad, “Hybrid input shaping and feedback control scheme of a flexible robot manipulator,” Proceeding of the 17th World Congress The International Federation of Automatic Control, Seoul, Korea, July 6-11, 2008, pp. 11714 – 11719.