

RESEARCH

Open Access



Hybrid spiral-dynamic bacteria-chemotaxis algorithm with application to control two-wheeled machines

K. M. Goher^{1*}, A. M. Almeshal³, S. A. Agouri², A. N. K. Nasir², M. O. Tokhi², M. R. Alenezi³, T. Al Zanki³ and S. O. Fadlallah⁴

Abstract

This paper presents the implementation of the hybrid spiral-dynamic bacteria-chemotaxis (HSDBC) approach to control two different configurations of a two-wheeled vehicle. The HSDBC is a combination of bacterial chemotaxis used in bacterial foraging algorithm (BFA) and the spiral-dynamic algorithm (SDA). BFA provides a good exploration strategy due to the chemotaxis approach. However, it endures an oscillation problem near the end of the search process when using a large step size. Conversely; for a small step size, it affords better exploitation and accuracy with slower convergence. SDA provides better stability when approaching an optimum point and has faster convergence speed. This may cause the search agents to get trapped into local optima which results in low accurate solution. HSDBC exploits the chemotactic strategy of BFA and fitness accuracy and convergence speed of SDA so as to overcome the problems associated with both the SDA and BFA algorithms alone. The HSDBC thus developed is evaluated in optimizing the performance and energy consumption of two highly nonlinear platforms, namely single and double inverted pendulum-like vehicles with an extended rod. Comparative results with BFA and SDA show that the proposed algorithm is able to result in better performance of the highly nonlinear systems.

Keywords: Spiral dynamics, Bacteria chemotaxis, Two-wheeled inverted pendulum with new configuration, PD-like fuzzy logic control, Hybrid fuzzy logic control

Background

Optimization algorithms play a dominant role in solving real problems [38, 58]. Bacterial foraging algorithm (BFA) [42] and spiral-dynamics algorithm (SDA) [50, 51] are well-known optimization techniques in solving real-world problems. Evolutionary algorithms (EA) have been used extensively in literature: soft computing techniques [46], particle swarm optimization [53, 55], incremental encoding [13], neural stochastic multi-scale optimization [9], multi-objective optimization [12, 23], multi-criteria optimization [43] and fuzzy logic and genetic programming [48].

Nasir et al. [33, 34, 36] proposed linear and nonlinear adaptive BFA where the bacteria step size is varied based on the combination of bacteria and iteration index. Chen and Lin [14], Farhat and El-Hawary [18] and Huang and Lin [22] utilized index and total number of chemotaxis to vary bacteria step size within a specified range. Niu et al. [39], Yan et al. [57] and Xu et al. [56] varied the step size within a user-defined range using combination of index and total number of iterations. Supriyono and Tokhi [49] developed various versions of BFA based on linear and nonlinear mathematical formulations to establish relationship between bacteria step size and their current fitness value. This relationship enables bacteria to have different step sizes in similar iteration as well as through the whole operation. There are other adaptive approaches considered the variation of the step size based on fitness value [16, 28, 29, 44, 45, 54]. Nasir et al. [30–32] proposed adaptive spiral-dynamic

*Correspondence: kgoher@clawar.org; k.m.goher@gmail.com

¹ Department of Informatics and Enabling Technologies, Lincoln University, Lincoln, New Zealand

Full list of author information is available at the end of the article

algorithm (ASDA) to establish relationship between spiral radius (r) and fitness value of each search point. They introduced schemes to make variation in the spiral radius within a specific range, enabling each search point to have different spiral radius in moving from one location to another location. Moreover, the movement step of each search agent was made with respect to its fitness value at the current location. As a result of the variation, there was improvement to the performance mainly on the accuracy of the final solution.

Hybrid optimization techniques

Hybrid approach is the combination of two or more algorithms aimed to retain the advantages and eliminate the weaknesses of the original algorithms. This includes the synergization between different groups such as bio-inspired, nature-inspired, etc. Biswas et al. [10, 11] proposed hybrid BFA-PSO where a chemotactic strategy of bacteria was designed to represent exploitation part of the algorithm, while the exploration of optimum location was accomplished by PSO. The same approach using a constant step size was implemented by Korani [26], where the PSO operator was used to determine new direction of bacteria motion. Ghaffar et al. [19] adopted a modified PSO operator to determine new direction of bacteria to avoid local optima solution. Biswas et al. [11] proposed chemotactic differential evolution algorithm where adaptive chemotactic strategy of bacteria has been used to improve fitness accuracy of classical differential evolution (DE). Sinha et al. [47] implemented the same approach on an electric power system. Kim et al. [24] and Kim [25] used GA and BFA to tune a PID controller for automatic voltage regulation. Panigrahi and Ravikumar [40] and Hooshmand et al. [21] incorporated Nelder–Mead method into bacteria chemotaxis phase to enhance the search strategy and improve bacteria location. Other hybrid approaches involving BFA [41, 59] used bee colony algorithm and Tabu search.

Limitations of BFA and SDA

BFA is a well-known bio-inspired algorithm. It has a comparable or better performance compared to other types of optimization algorithm [17]. Therefore, it has been adopted by many researchers worldwide to solve real-world problems in many areas [52]. However, BFA has a slow convergence speed and longer computation time. Due to this issue, the application of original BFA in online and offline tuning for solving a complex real-world problem is unsatisfactory [15]. On the other hand, SDA is a relatively new and a simple algorithm developed inspired from natural spiral phenomena on earth. It has a relatively fast convergence speed which can complement the

drawback of BFA performance. Previous study showed that SDA has a similar or better performance compared to other differential evolutionary (DE) and particle swarm optimization (PSO) algorithms [50, 51]. However, SDA has a premature convergence issue where it hardly provides an optimal solution for complex problems.

Hybrid spiral-dynamic bacteria-chemotaxis

A hybrid bacteria-chemotaxis spiral-dynamic algorithm (HSDBC) has been proposed by Nasir et al. [30–32] to synergize the chemotactic strategy of bacteria and ASDA. The chemotaxis phase in BFA was designed such that it represents exploration stage and placed at the first phase of the algorithm, while the ASDA as the exploitation stage and was placed at the second phase of the algorithm. The combination simplified the BFA algorithm and greatly reduced the total computation time of BFA. Moreover, comparison with original algorithms concluded that it improved the accuracy of the final solution and had the capability to avoid the local optima problem. HSDBC is a new variant of hybrid-type BFA-SDA algorithm developed to solve the issues aforementioned above. Our previous study showed that the algorithm outperformed both BFA and SDA algorithms in terms of accuracy in finding a global optima solution. Compared to BFA, the total computation time has been significantly reduced and its convergence speed has been considerably increased [31, 37].

Full description of the HSDBC algorithm for n -dimensional optimization is shown in Fig. 1. The description of the associated parameters used in the algorithm is shown in Table 1, and the corresponding flow chart is given in Fig. 2. The HSDBC algorithm has been tested to model and control nonlinear systems including flexible robot manipulator and a twin rotor system using a PD-like FLC [35, 37].

Contribution overview and paper organization

Establishing the optimal control strategy for nonlinear dynamic systems, specifically inverted pendulum-based systems, has been and still remains a field of interest for a countless number of research studies due to their various promising real-life applications including personal transport systems and wheelchairs. This paper presents an extended study of the proposed algorithm in solving complex problem of two-wheeled inverted pendulum systems. We will implement HSDBC algorithm to control two different configurations of two-wheeled machines. A detailed simulation study of the HSDBC algorithm using several unimodal and multimodal benchmark functions can be found in the work of Nasir and Tokhi [37]. A hybrid fuzzy-like PD and I controller is designed and implemented on the two systems.

Step 0: Preparation

Select the number of search points (bacteria)
 $m \geq 2$, parameters
 $0 \leq \theta_{tumble}, \theta_{swim} < 2\pi, 0 < r_{tumble}, r_{swim} < 1$ 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 $x_i(0) \in R^n, i = 1, 2, \dots, m$ in the
 feasible region at random and center x^* as
 $x^* = x_{i_g}(0), i_g = \arg \min_i f(x_i(0)), i = 1, 2, \dots, m$.

Step 2: Applying bacteria chemotaxis

i. Bacteria tumble

(a) Update x_i

$$x_i(k+1) = S_n(r_{tumble}, \theta_{tumble})x_i(k) - (S_n(r_{tumble}, \theta_{swim}) - I_n)x^*$$

$$i = 1, 2, \dots, m.$$

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

$$x_i(k+1) = S_n(r_{swim}, \theta_{swim})x_i(k) - (S_n(r_{swim}, \theta_{swim}) - I_n)x^*$$

$$i = 1, 2, \dots, m.$$

Step 3: Updating x^*

$x^* = x_{i_g}(k+1),$
 $i_g = \arg \min_i f(x_i(k+1)), i = 1, 2, \dots, m.$

Step 4: Checking termination criterion

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

Fig. 1 HSDBC algorithm for n -dimensional optimization

Table 1 HSDBC algorithm parameters

θ_{tumble}	Bacteria angular displacement on $x_i - x_j$ plane around the origin for tumbling
θ_{swim}	Bacteria angular displacement on $x_i - x_j$ plane around the origin for swimming
r_{tumble}	Spiral radius from bacteria tumble
r_{swim}	Spiral radius for bacteria swim
m	Number of search points
k_{max}	Maximum iteration number
N_{sw}	Maximum number of swim
$x_i(k)$	Bacteria position
R^n	$n \times n$ matrix

This paper is organized as follows: “**Background**” section introduces both ASDA and ABFA optimization algorithms, along with an explanation of the principle of HSDBC algorithm. In order to test and validate the proposed HSDBC algorithm on real dynamic systems, two case studies are considered in the study and are introduced in “**Methods**” section. “**Case study I: single IP with an extended rod**” section describes in details the first case study that involves a single inverted pendulum (IP) system. A double IP system with an extended rod is considered as the second case study and is presented in “**Case study II: double IP with an extended rod**” section. The results of the investigation are presented at the end of each of the previously mentioned sections, sections “**Case study I: single IP with an extended rod**” and “**Case study II: double IP with an extended rod**”. At last, the paper is concluded in “**Conclusion**” section.

Methods

An inverted pendulum as a typical multi-input multi-output system has the characteristics of nonlinear, multivariable and close coupling Luo et al. [27]. The uniqueness and wide application of technology derived from this unstable system has drawn interest of many researchers including Akesson et al. [2], Askari et al. [5] and Balan et al. [6, 7]. There are various applications of IP configuration including design of walking gaits, wheelchairs, and personal transport systems.

The system considered in this paper is a two-wheeled machine (TWM) with an extendable rod as described by Goher et al. [20] and verified by Almeshal et al. [3, 4]. This system stabilizes it extendable intermediate body (IB) by controlling the wheel movements in a desired manner. A TWM is designed such that either

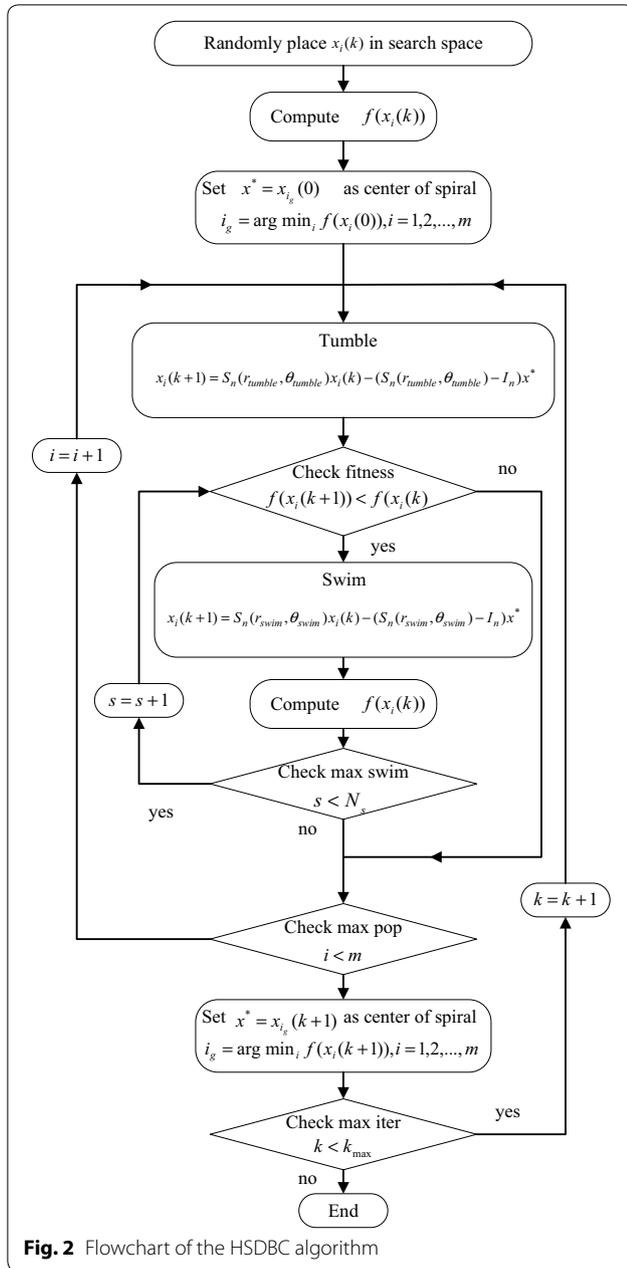


Fig. 2 Flowchart of the HSDBC algorithm

the center of mass of the robot is above or below the axle joining two wheels. Statically unstable TWM have evoked a lot of interest in present decade [8]. Two case studies are used to test and validate the developed algorithm; single IP and double IP with an extended rod. For consistency, the two systems are considered to move along an inclined surface. The results of the simulation are shown in a comparative manner with three different optimization algorithms; BFA, SDA and HSDBC.

Case study I: single IP with an extended rod

System description

The system comprises a rod on an axle incorporating two wheels as shown in Figs. 3 and 4. The numerical parameters of the system are described in “Appendix 1”. Full details on the system description are available in Almeshal et al. [3, 4].

Mathematical modeling of the single IP with an extended rod

Lagrange-Euler formulation is used to derive the system dynamic model using the following n-coordinates dynamic equations:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = Q_i, \quad i = 1, 2, \dots, n \quad (1)$$

where Q_i is generalized force vector and q_i is generalized coordinate vector. The coordinate vector is selected as:

$$q_i = [\gamma \ \theta \ Q]^T \quad (2)$$

and the force vector as:

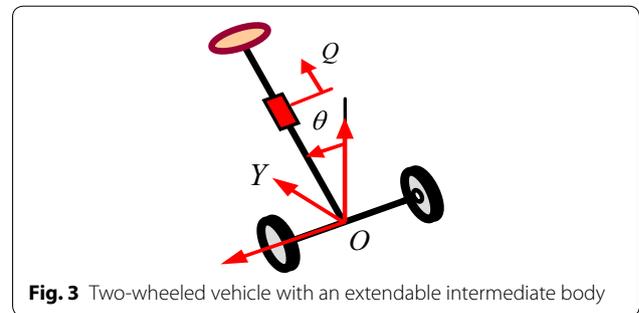


Fig. 3 Two-wheeled vehicle with an extendable intermediate body

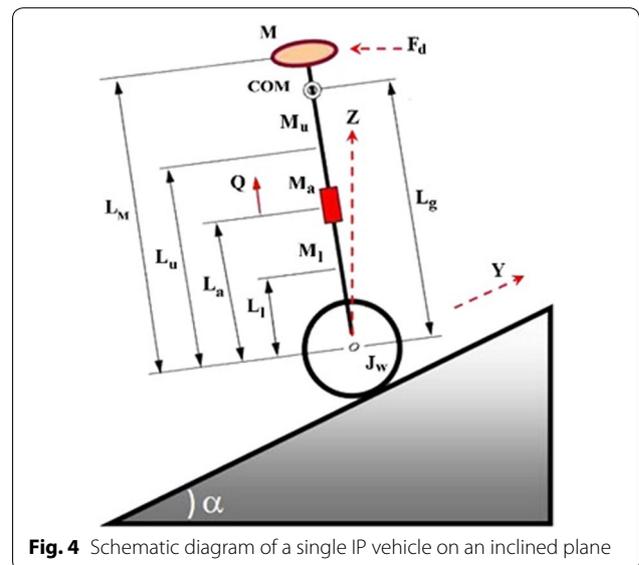


Fig. 4 Schematic diagram of a single IP vehicle on an inclined plane

$$Q_i = [F_c F_d F_a]^T \tag{3}$$

The system equations of motion of the model can be written as:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{y}} \right) - \frac{\partial L}{\partial y} = F_c \tag{4}$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = F_d \tag{5}$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{Q}} \right) - \frac{\partial L}{\partial Q} = F_a \tag{6}$$

Driving further the above equations yields the following nonlinear equations of motion of the system:

$$C_7 \ddot{y} + (C_{15} + C_{16}Q) \ddot{\theta} \cos(\theta + \alpha) - (C_{15} + C_{16}Q) \dot{\theta}^2 \sin(\theta + \alpha) + C_{16} \dot{Q} \dot{\theta} \cos(\theta + \alpha) + C_{11} \sin \alpha = F_c \tag{7}$$

$$(C_{18} + QC_{16}) \ddot{y} \cos(\theta + \alpha) - (C_{18} + QC_{16}) \dot{y} \dot{\theta} \sin(\theta + \alpha) + C_{16} \dot{Q} \dot{y} \cos(\theta + \alpha) + 2\ddot{\theta} (C_{12}Q^2 + C_{13}Q + C_{14}) + \dot{\theta} (4C_{12}Q + 2C_{13}\dot{Q}) + \dot{y} \dot{\theta}^2 \sin(\theta + \alpha) (C_{10} + M_u(C_5 + Q) + M_m(C_6 + Q)) - \dot{\theta} \sin \theta (C_{10}g + M_u g(C_5 + Q) + M_m g(C_6 + Q)) = F_d \tag{8}$$

$$2C_8 \ddot{Q} - C_{16} \dot{\theta} \dot{y} \cos(\theta + \alpha) - 2C_{12} Q \dot{\theta}^2 - C_{13} \dot{\theta}^2 - C_{16} g \cos \theta = F_a \tag{9}$$

Detailed explanations of the constant parameters appearing in Eqs. (4)–(9) are formulated in ‘‘Appendix 2’’.

Control strategy

Three independent control loops, shown in Fig. 5, are implemented on the system. Fuzzy PD-like combined with conventional integrator is designed as shown in Fig. 6. The three control loops are working to: stabilize the

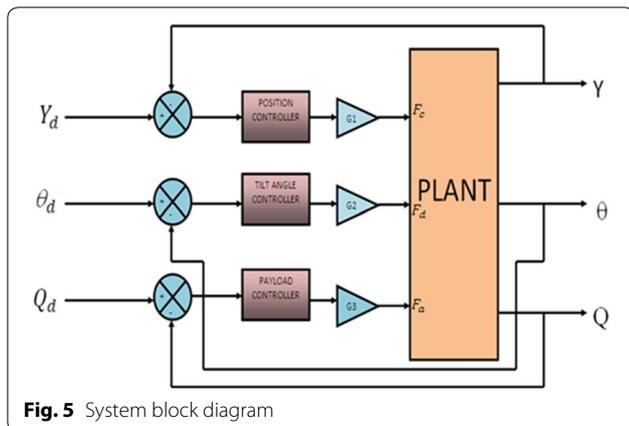


Fig. 5 System block diagram

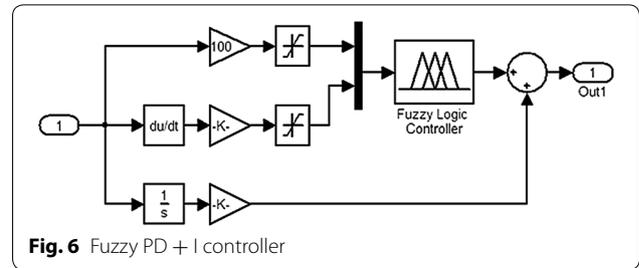


Fig. 6 Fuzzy PD + I controller

IB at the vertical upright position, keep the cart wheels within a specified linear position from a predefined reference while moving on an inclined surface, and to control the linear displacement of the payload along the IB.

The inputs to the three control loops are the error signal, change of error and the sum of previous errors. The system inputs are the driving force \$F_c\$, the linear actuator force \$F_a\$ and the disturbance force \$F_d\$. FLC controllers are developed based on Mamdani-type fuzzy inference engine with (25) fuzzy rules shown in Table 2.

Constrained optimization

The optimization process is constrained within the stability region of the system. Each parameter has a feasible interval that guarantees the stability of the system within the defined gain limits. Table 3 presents the limits of each parameter which represent the search space of each of

Table 2 Fuzzy rule base

\dot{e}	e				
	NB	NS	Z	PS	PB
NB	NB	NB	NB	NS	Z
NS	NB	NB	NS	Z	PS
Z	NB	NS	Z	PS	PB
PS	NS	Z	PS	PB	PB
PB	Z	PS	PB	PB	PB

Table 3 Boundary limits of the controller gain parameters

Gain parameters	Minimum value	Maximum value
Kp_1	4	5
Kd_1	3	4
Ki_1	0.4	0.8
Kp_2	4	5
Kd_2	3	4
Ki_2	1	1.3
Kp_3	10	13
Kd_3	15	20
Ki_3	2	3

the three addressed algorithms. Those parameters were obtained through a manual tuning exercise of the system.

Objective functions

The performance index of the system is chosen as the minimum mean squared error (MSE) of each control loop. The MSE is calculated for each control loop of the vehicle system using the following equations:

$$\begin{aligned}
 \text{Objective_Function1} &= \min \left[\frac{1}{N} \sum_{i=1}^N (Y_d - Y_m)^2 \right] \\
 \text{Objective_Function2} &= \min \left[\frac{1}{N} \sum_{i=1}^N (\theta_d - \theta_m)^2 \right] \\
 \text{Objective_Function3} &= \min \left[\frac{1}{N} \sum_{i=1}^N (Q_d - Q_m)^2 \right]
 \end{aligned} \tag{10}$$

The objective function of the system is calculated based on the total MSE which can be expressed as:

$$J = \sum_{i=1}^3 \text{Objective_function}(i) \tag{11}$$

The parameters used to implement the three optimization algorithms are shown in Tables 4, 5 and 6 and the calculated optimized parameters are shown in Table 7. The data shown in Table 8 gives the minimum cost functions due to the implementation of the three optimization algorithms where the HSDBC algorithm was able to give the minimum cost function compared to the BFA and SDA optimizations.

Simulation results

Four consecutive simulation runs of the system model yielded the performance of the system as shown in Fig. 7.

Table 4 BFA parameters

P	S	Nc	Ns	Nre	Ned	Ped	Sr
9	40	10	6	2	2	0.25	S/2

Table 5 SDA parameters

P	R	Theta	Initial points	Iterations
9	0.9	$\pi/4$	5	90

Table 6 HSDBC parameters

P	R	Rzw	Ns	Theta	Initial points	Iterations
9	0.95	0.55	2	$\pi/4$	5	90

Table 7 Optimized gain values

	Parameter	BFA	SDA	HSDBC
Loop 1	Kp_1	4.2287	4.0000	4.0003
	Kd_1	3.0064	3.1065	3.0089
	Ki_1	0.7380	0.6773	0.7267
Loop 2	Kp_2	4.5638	4.3183	4.7770
	Kd_2	3.2615	3.6085	3.4461
	Ki_2	1.0322	1.2380	1.1306
Loop 3	Kp_3	11.4488	10.7368	11.3992
	Kd_3	19.3417	17.1030	18.0021
	Ki_3	2.6508	2.0113	2.6529

Table 8 Cost functions

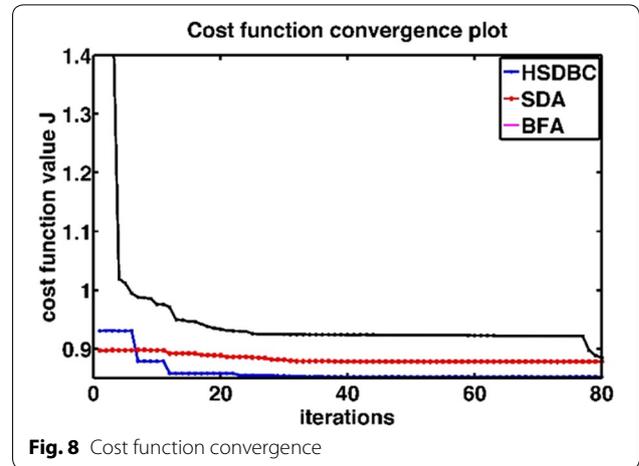
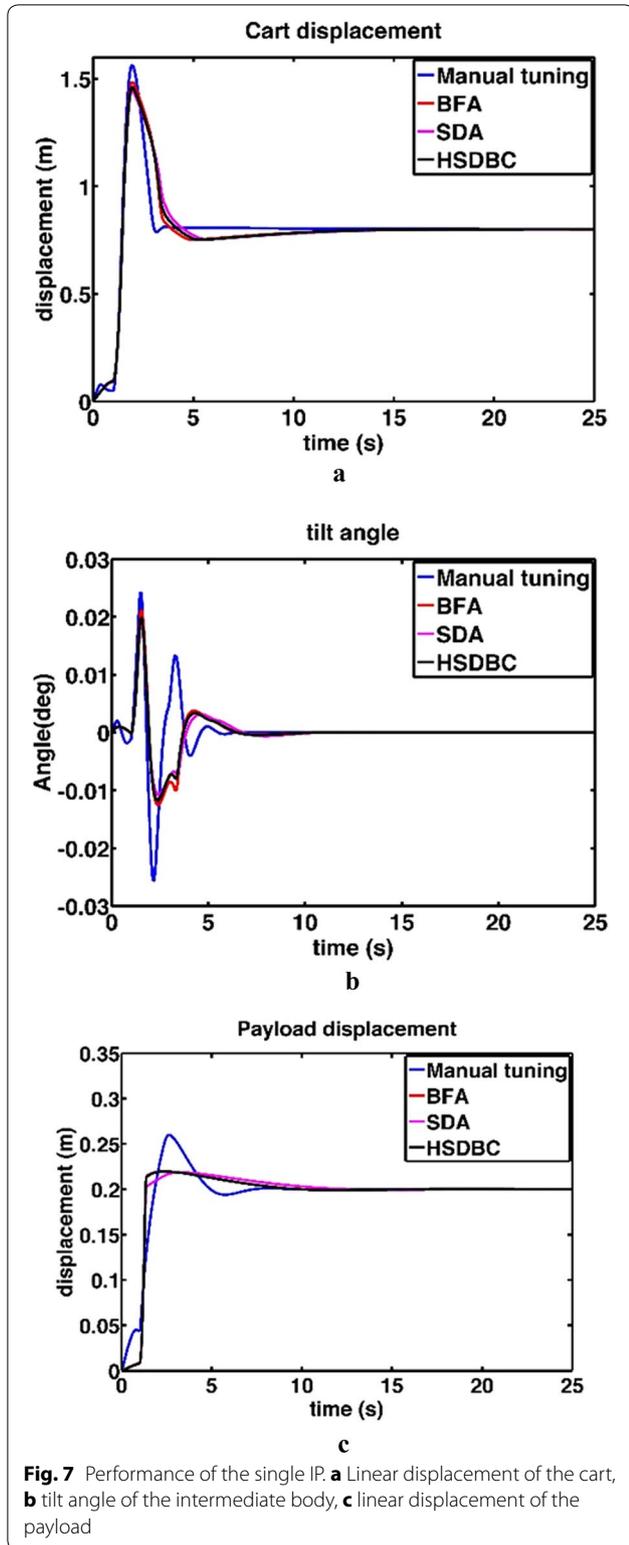
Minimum cost function value	BFA	SDA	HSDBC
J	0.922	0.8804	0.8517

As noted from Fig. 6; the three optimization algorithms; BFA, SDA and HSDBC, resulted generally in a satisfactory performance of the system. However, HSDBC algorithm showed a superior performance in minimizing the percentage overshoot in the payload displacement as appeared in Fig. 7c. As per the tilt angle shown in Fig. 7b, all the three algorithms behaved the same in terms of minimizing the level and period of oscillations. As per convergence graph shown in Fig. 8 show that the three algorithms resulted in similar convergence of the cost function within around 25 iterations. However, the HSDBC was faster in convergence of the cost function too early if compared to the BFA and SDA algorithms.

Attention has been focused on energy consumption in this investigation. The control effort components as a measure of energy consumption are shown in Fig. 9. It is noted that the control effort required in the transient range; the three algorithms yielded nearly close results. However, the HSDBC was more robust as it resulted less oscillation of the control effort components in the magnified areas of the plots. Significant amount of energy saving has been achieved specifically in the cart and tilt angle control efforts as appeared in Fig. 9a, b. Furthermore, the HSDBC resulted in a great improvement in the control effort for the payload; this can be demonstrated by the significant improvement shown in Fig. 9c in terms of less oscillations and the short time taken by the control signal to stabilize.

Case study II: double IP with an extended rod

In this case study, an additional link is added and hence increasing the degrees of the freedom (DOF) and the complexity of the structure. The double IP with such configuration shown in Fig. 10 is mimicking the scenario of

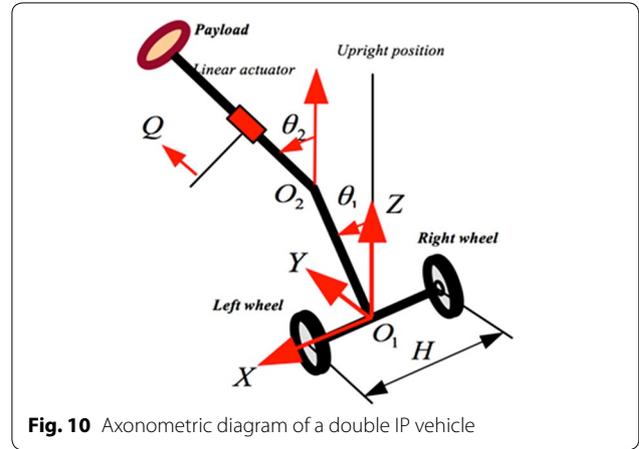
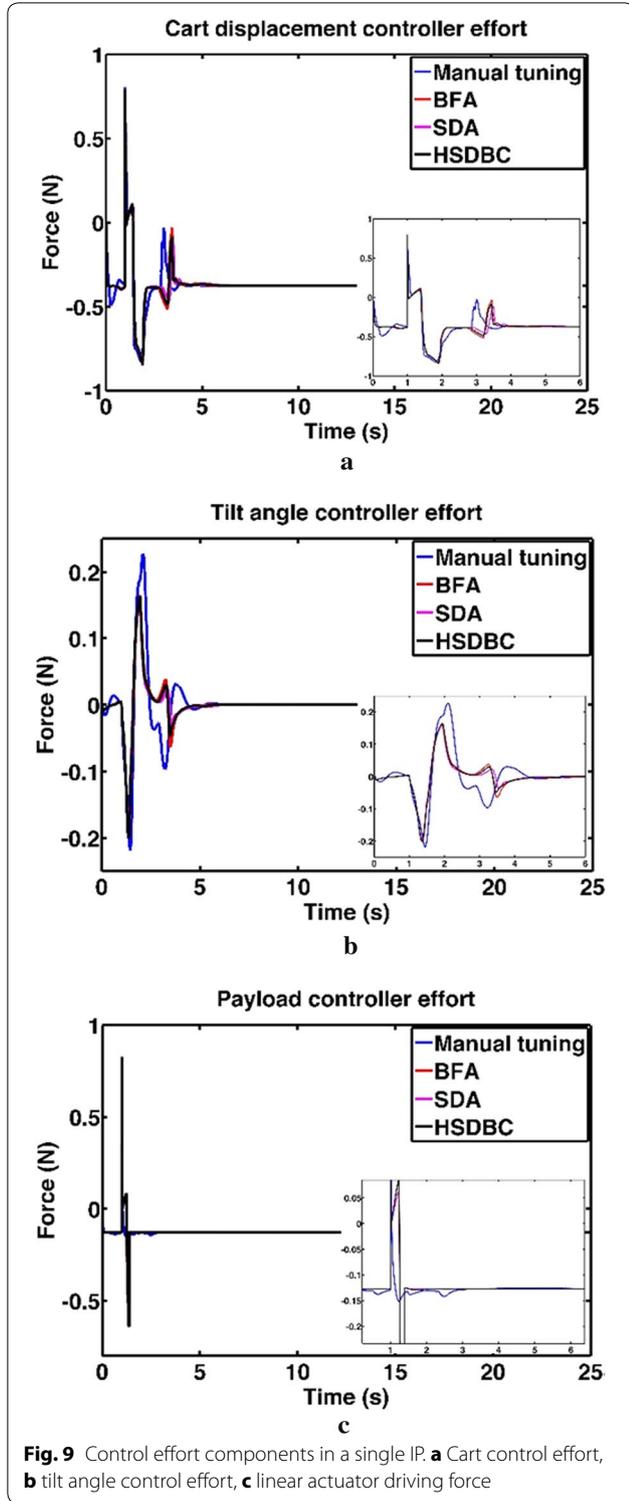


a wheelchair on only two wheels which has been studied significantly by Ahmad and Tokhi [1].

The design of the two-wheeled robotic vehicle is based on double inverted pendulum system with a movable payload moving on an inclined surface with five DOF. The increased DOFs will enable the vehicle to maneuver freely in all directions and in different environments. Moreover, the second link provides an extended height to lift up the payload to a demanded height. The system equations of motion are presented with five highly coupled differential equations as follows:

$$\begin{aligned}
 &2C_{27}\ddot{\delta}_L + 2C_1\ddot{\delta}_R + C_6\ddot{\theta}_1 \cos(\theta_1 + \alpha) \\
 &\quad - C_6\dot{\theta}_1^2 \sin(\theta_1 + \alpha) \\
 &\quad + 0.5(C_{25} + C_{26}Q) \\
 &\quad \left(\ddot{\theta}_2 \cos(\theta_2 + \alpha) - \dot{\theta}_2^2 \sin(\theta_2 + \alpha) \right) \\
 &\quad + 0.5C_{28}\dot{Q}\dot{\theta}_2 \cos(\theta_2 + \alpha) \\
 &\quad + C_{16}g \sin \alpha = T_L - T_{fl}
 \end{aligned} \tag{12}$$

$$\begin{aligned}
 &2C_{27}\ddot{\delta}_R + 2C_1\ddot{\delta}_L + C_6\ddot{\theta}_1 \cos(\theta_1 + \alpha) \\
 &\quad - C_6\dot{\theta}_1^2 \sin(\theta_1 + \alpha) \\
 &\quad + 0.5(C_{25} + C_{26}Q) \\
 &\quad \left(\ddot{\theta}_2 \cos(\theta_2 + \alpha) - \dot{\theta}_2^2 \sin(\theta_2 + \alpha) \right) \\
 &\quad + 0.5C_{28}\dot{Q}\dot{\theta}_2 \cos(\theta_2 + \alpha) \\
 &\quad + C_{16}g \sin \alpha = T_R - T_{fr}
 \end{aligned} \tag{13}$$



$$\begin{aligned}
 & 2C_2\ddot{\theta}_1 + (C_5 + M_{2u}L_1(C_8 + Q) + ML_1(C_9 + Q)) \\
 & (\ddot{\theta}_2 \cos(\theta_1 - \theta_2) - \dot{\theta}_2(\dot{\theta}_1 - \dot{\theta}_2) \sin(\theta_1 - \theta_2)) \\
 & + \dot{\theta}_2 \cos(\theta_1 - \theta_2) (\dot{Q}(M_{2u}L_1 + ML_1)) \\
 & + C_6(\ddot{\delta}_L + \ddot{\delta}_R) \cos(\theta_1 + \alpha) \\
 & + C_6(\dot{\delta}_L + \dot{\delta}_R) \sin(\theta_1 + \alpha) (\dot{\theta}_1^2 - \dot{\theta}_1) \\
 & + \dot{\theta}_1^2 \dot{\theta}_2 \sin(\theta_1 - \theta_2) \\
 & (C_5 + M_{2u}L_1(C_8 + Q) + ML_1(C_9 + Q)) \\
 & - gC_{14}\dot{\theta}_1 \sin \theta_1 = 0.5(T_R + T_L) \tag{14}
 \end{aligned}$$

$$\begin{aligned}
 & C_{19}\ddot{Q} - 0.5\dot{\theta}_2^2(2C_{19}Q + C_{22}) \\
 & - C_{23}\dot{\theta}_1\dot{\theta}_2 \cos(\theta_1 - \theta_2) \\
 & - 0.5C_{25}\dot{\theta}_2(\dot{\delta}_L + \dot{\delta}_R) \cos(\theta_2 + \alpha) \\
 & + gC_{18} \cos \theta_2 = F_a - F_{fa} \tag{15}
 \end{aligned}$$

$$\begin{aligned}
 & \ddot{\theta}_2(C_{19}Q^2 + C_{20}Q + C_{21}) + \dot{\theta}_2(2C_{19}Q^2 + C_{22}) \\
 & + \ddot{\theta}_1 \cos(\theta_1 - \theta_2)(C_{23}Q + C_{24}) \\
 & - \dot{\theta}_1(\dot{\theta}_1 - \dot{\theta}_2) \sin(\theta_1 - \theta_2)(C_{23}Q + C_{24}) \\
 & + C_{23}\dot{\theta}_1 \cos(\theta_1 - \theta_2) \\
 & + 0.5(\dot{\delta}_L + \dot{\delta}_R) \cos(\theta_2 + \alpha)(C_{25}Q + C_{26}) \\
 & - 0.5(\dot{\delta}_L + \dot{\delta}_R)\dot{\theta}_2 \sin(\theta_2 + \alpha)(C_{25}Q + C_{26}) \\
 & + 0.5C_{25}(\dot{\delta}_L + \dot{\delta}_R) \cos(\theta_2 + \alpha) \\
 & - \dot{\theta}_1 \dot{\theta}_2 \sin(\theta_1 - \theta_2)(C_{23}Q + C_{24}) \\
 & + 0.5\dot{\theta}_2^2(\dot{\delta}_L + \dot{\delta}_R) \sin(\theta_2 + \alpha)(C_{25}Q + C_{26}) \\
 & - g\dot{\theta}_2 \sin \theta_2(C_{17} + C_{18}Q) = T_M - T_{fM} - L_d F_d \tag{16}
 \end{aligned}$$

Control strategy

A robust hybrid fuzzy logic control strategy (FLC) with five control loops is developed. The control strategy block diagram is presented in Fig. 11, to control the vehicle and to counteract the disturbances occurring due to different movement scenarios.

The control system of the vehicle consists of five hybrid FLC controllers with a total of 15 gain parameters. The gain parameters were first tuned heuristically in order to test the controller as well as to find the boundaries of the search space of those gain parameters. The same optimization algorithms, SDA, BFA and HSDBC, are implemented in order to optimize the vehicle control system parameters. The performance index of the system is chosen as the minimum mean squared error (MSE) for each control loop and defined as:

$$\begin{aligned}
 \text{MSE 1} &= \min \left\{ \frac{1}{N} \sum_{i=1}^N (\delta_{Ld} - \delta_{Lm})^2 \right\} \\
 \text{MSE 2} &= \min \left\{ \frac{1}{N} \sum_{i=1}^N (\delta_{Rd} - \delta_{Rm})^2 \right\} \\
 \text{MSE 3} &= \min \left\{ \frac{1}{N} \sum_{i=1}^N (\theta_{1d} - \theta_{1m})^2 \right\} \\
 \text{MSE 4} &= \min \left\{ \frac{1}{N} \sum_{i=1}^N (\theta_{2d} - \theta_{2m})^2 \right\} \\
 \text{MSE 5} &= \min \left\{ \frac{1}{N} \sum_{i=1}^N (Q_d - Q_m)^2 \right\}
 \end{aligned} \tag{17}$$

The objective function is chosen as the summation of the MSE of the system expressed as:

$$J = \text{MSE}_1 + \text{MSE}_2 + \text{MSE}_3 + \text{MSE}_4 + \text{MSE}_5 \tag{18}$$

Minimization of the objective function J is used to find the optimal controller gain parameters that result in the minimum control loop errors in the stability region of the system.

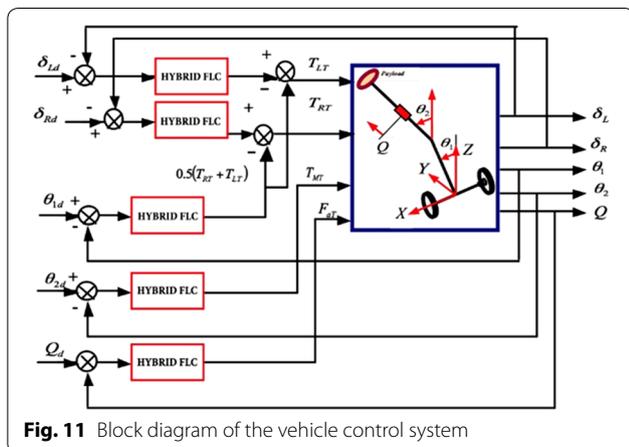


Fig. 11 Block diagram of the vehicle control system

Constrained optimization

With the complexity of the model, slight changes in the control gain parameters will result in oscillations in the system response and may lead to instability of the vehicle. Constrained optimization techniques are used to avoid this problem occurring while optimizing the control system parameters. The optimization process is constrained within the stability region of the system. This is achieved by defining a feasible interval for each control parameter shown in Table 9, which assures the stability of the system.

Results and discussion

This simulation scenario allows comparing the performance of the HSDBC with other similar optimization algorithms. Tables 10, 11 and 12 provide the simulation

Table 9 Boundary limits of the controller gain parameters

	Parameter	Lower	Upper
Loop 1	Kp_1	1.5	2.4
	Kd_1	0.5	1
	Ki_1	0.9	1.4
Loop 2	Kp_2	5	6.5
	Kd_2	2.5	4
	Ki_2	1.5	2
Loop 3	Kp_3	8	12
	Kd_3	7.5	9
	Ki_3	0	0.5
Loop 4	Kp_4	8	10
	Kd_4	5	8
	Ki_4	0	0.5
Loop 5	Kp_5	30	50
	Kd_5	10	20
	Ki_5	1	10

Table 10 BFA parameters

P	S	Nc	Ns	Nre	Ned	Ped	Sr
15	20	14	6	2	2	0.25	S/2

Table 11 SDA parameters

P	R	Theta	Initial points	Iterations
15	0.95	$\pi/4$	10	150

Table 12 HSDBC parameters

P	R	Rzw	Ns	Theta	Initial points	Iterations
15	0.95	0.55	2	$\pi/4$	10	150

parameters used for BFA, SDA and HSDBC algorithms, respectively. The optimized control gain parameters reported by each optimization algorithm are presented in Table 13, whereas Table 14 provides the minimum cost function calculated by each of the optimization algorithms. Clearly, the HSDBC algorithm has found the minimum cost function value of 0.3682.

Figure 13 shows the system response based on the optimized control parameters obtained by the implementation of the BFA, SDA and HSDBC algorithms in comparison to the manual-tuned gain parameters. It can be noted that BFA, SDA and HSDBC are of much similar effect on the system response by finding stable solutions, lowering the overshoots and improved steady-state error. However, HSDBC algorithm has a superior performance in minimizing the percentage overshoot and the settling time for the linear displacement of the left and right wheel as shown in Fig. 13a,

Table 13 Optimized gain values

	Parameter	BFA	SDA	HSDBC
Loop 1	Kp_1	2.0729	2.3452	2.1566
	Kd_1	0.8572	0.8714	0.8095
	Ki_1	1.3925	1.2778	1.2026
Loop 2	Kp_2	6.0155	5.1504	5.1530
	Kd_2	2.8185	3.1264	2.6917
	Ki_2	1.6390	1.9794	1.8754
Loop 3	Kp_3	8.7514	11.3330	11.4514
	Kd_3	8.1889	8.3229	8.9946
	Ki_3	0.2449	0.2731	0.3771
Loop 4	Kp_4	8.9718	9.8522	9.9903
	Kd_4	5.1315	6.7829	6.6239
	Ki_4	0.0071	0.0532	0.0410
Loop 5	Kp_5	49.9646	36.5230	36.6753
	Kd_5	13.6834	14.2519	14.3583
	Ki_5	4.0408	5.3567	5.4203

Table 14 Cost functions

Minimum cost function	BFA	SDA	HSDBC
J	0.3684	0.3685	0.3682

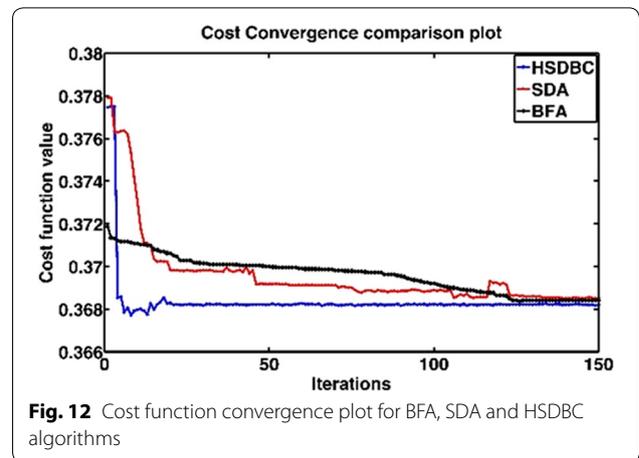
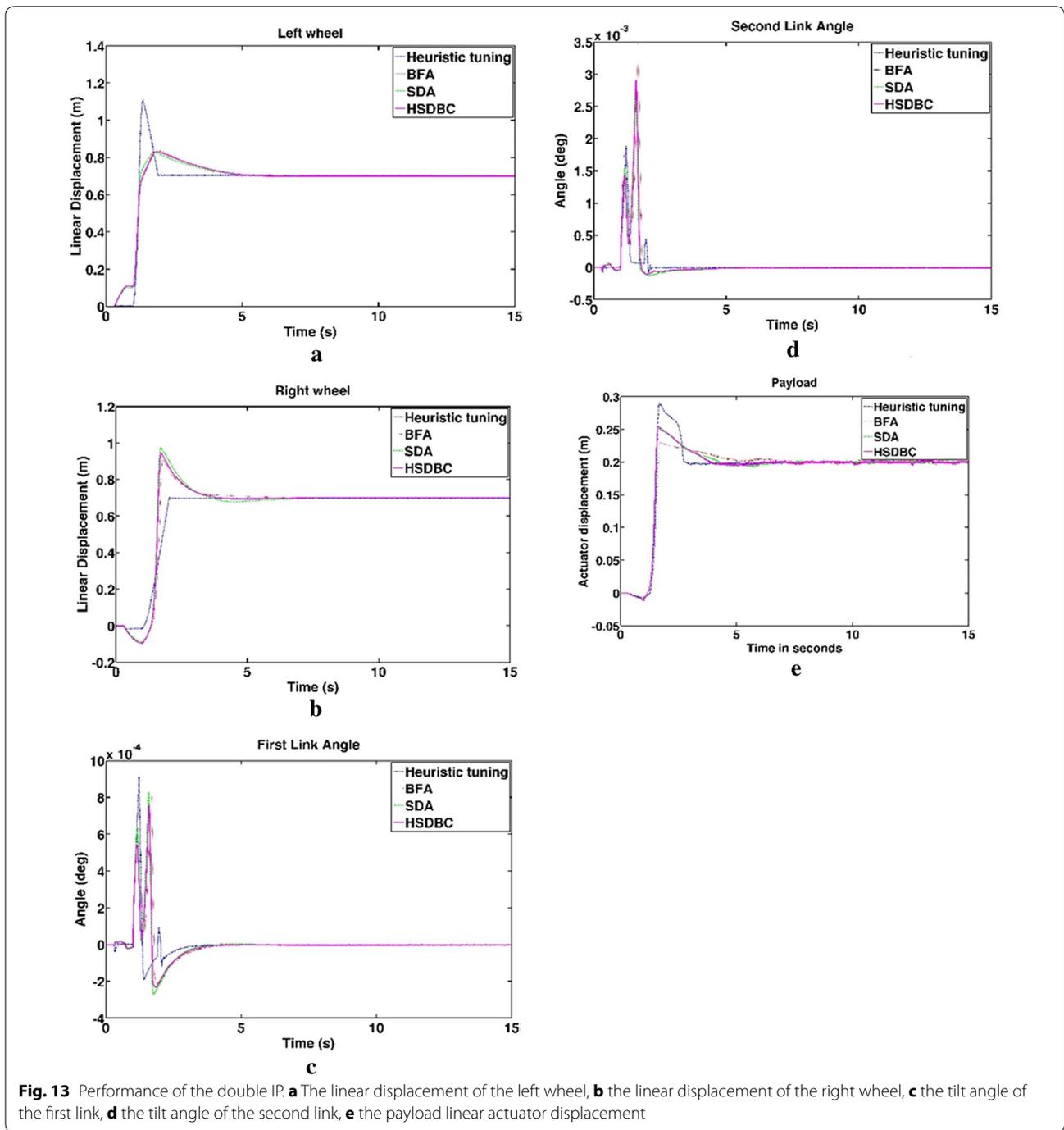


Fig. 12 Cost function convergence plot for BFA, SDA and HSDBC algorithms

b and the tilt angles of the two pendula as shown in Fig. 13c, d. Furthermore, HSDBC-optimized gain parameters clearly improved the settling time of the payload actuator displacement as depicted in Fig. 13e. As can be noticed from the cost function convergence plots shown in Fig. 12, the HSDBC algorithm cost function has converged into the minimum value within approximately 25 iterations. However, the BFA and SDA algorithms seem to need more iterations to settle into their best-found minimum values presented in Table 14. HSDBC has successfully found the minimum cost function and proved its speed in convergence. In terms of the control output components shown in Fig. 14, the control efforts was minimized by the implementation of HSDBC algorithm for the left wheel, first link and the payload linear actuator. However, the heuristic tuning yields better results in case of the right wheel and the second link. This seems to be accompanied with a poor response of the system, in terms of increased disturbance period and higher gain values, if compared to the results obtained by the HSDBC algorithm.

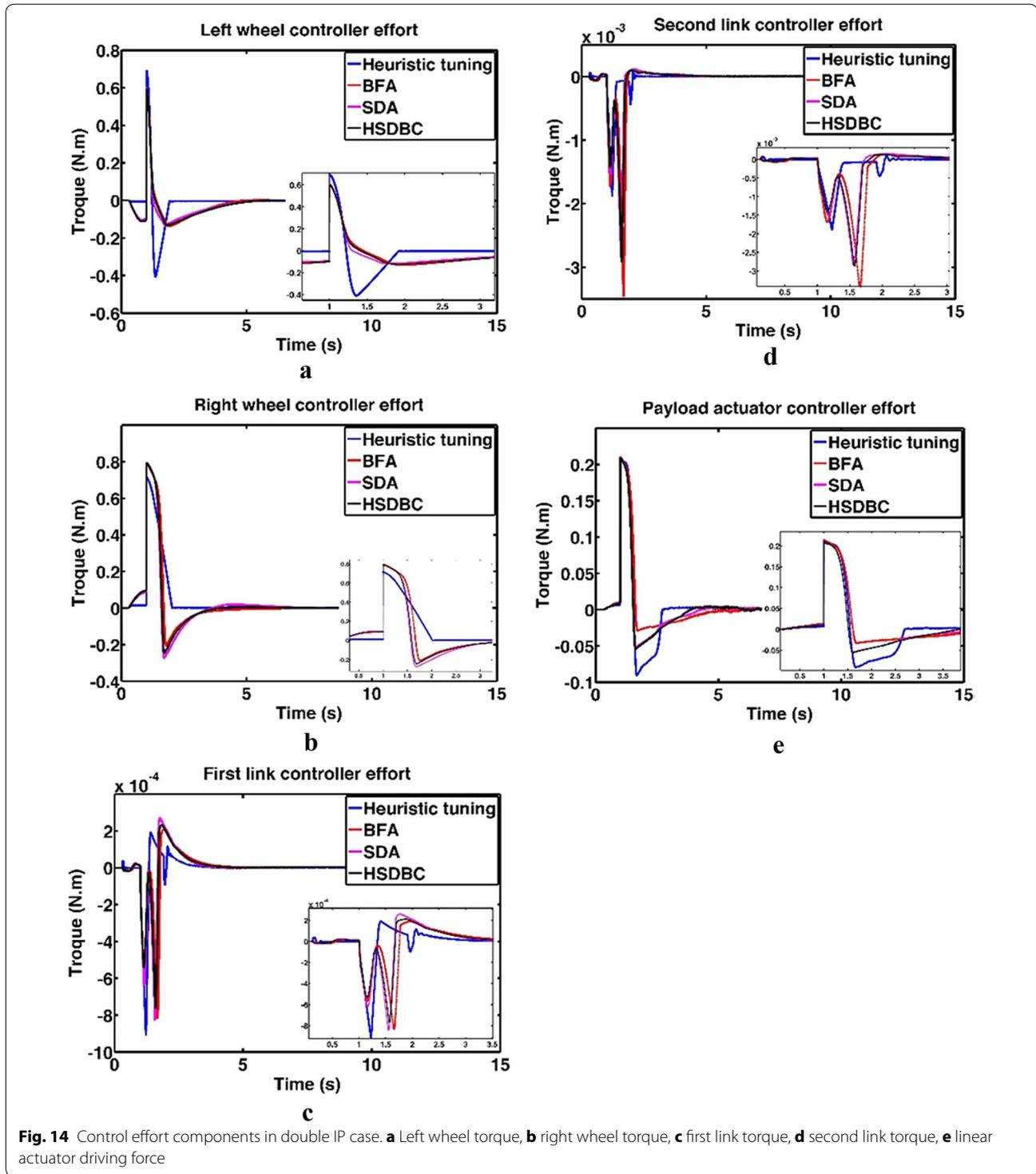
Conclusions

A novel hybrid spiral-dynamics bacteria-chemotaxis (HSDBC) optimization algorithm has been proposed. Chemotactic strategy of bacteria through spiral tumble and swim actions of bacteria is adopted to improve exploration strategy of SDA. Moreover, spiral radius



and angular displacement of spiral model is made adaptive to enhance the movement of bacteria within feasible region. Incorporating these two schemes have successfully saved the SDA from getting trapped into local optima point and provides faster convergence. The proposed algorithm has been utilized to optimize the performance of two different IP platforms; single

and double IP with a new configuration of an extended intermediate body. Simulation results have shown that the proposed hybrid algorithm outperformed its predecessor algorithms (BFA and SDA) in terms of increased convergence speed and better fitness accuracy. Furthermore, implementation of the HSDBC yielded significant



saving in the energy consumption of the two tested platforms.

Future work will consider investigating standard PID tuning methods, such as Ziegler–Nichols method, and evaluating and comparing their performance with the HSDBC algorithm.

Authors' contributions

KG initiated the concept and developed the system of two-wheeled machine with extended rods. AN developed the HSDBC algorithm. AA and SA contributed to the modeling and simulation of the system. OT was overseeing the entire research and he gave technical insights in the control part. MA and TA helped with the editing of the final draft. Final editing of the manuscript is managed by KG. All authors read and approved the final manuscript.

Author details

¹ Department of Informatics and Enabling Technologies, Lincoln University, Lincoln, New Zealand. ² Department of Automatic Control and Systems Engineering, The University of Sheffield, Sheffield, UK. ³ Electronics Engineering Technology Department, College of Technological Studies, Public Authority for Applied Education and Training (PAAET), Adailiya, Kuwait. ⁴ Mechanical Engineering Department, Auckland University of Technology, Auckland, New Zealand.

Acknowledgements

The authors of this paper would like to thank Lincoln University in New Zealand by offering the funding support for this publication.

Competing interests

The authors declare that they have no competing interests.

Funding

This research is originally funded by a governmental PhD scholarships and research grants from various countries including New Zealand, Malaysia, Kuwait and the UK.

Appendices**Appendix 1: Single IP parameters**

	Description	Unit
L_P	Position of the COM of the payload	m
L_u	Position of the COM of the rod's upper part	m
L_a	Position of COM of the linear actuator	m
L_l	Position of COM of the lower part of the rod	m
Q	Displacement of the linear actuator	m
Y	Linear displacement of the vehicle	m
θ	The angular displacement of the IB	rad
\dot{y}	Linear velocity of the IB	m/s
$\dot{\theta}$	Angular velocity of the IB	rad/s
\ddot{y}	Linear acceleration of the IB	m/s ²
$\ddot{\theta}$	Angular acceleration	rad/s ²
\ddot{Q}	Linear acceleration of the attached payload	m/s ²
M_L	Mass of the lower part of the rod	kg
M_a	Mass of the linear actuator	kg
M_u	Mass of the upper part of the rod	kg
M	Payload mass	kg
M_W	Wheel mass	kg
M_c	Cart mass	kg
R_W	Radius of the wheel	m
g	Gravitational acceleration	m/s ²
T_c	Kinetic energy of the cart	N m
T_L	Kinetic energy of the lower part of the rod	N m
T_a	Kinetic energy of the linear actuator	N m
T_u	Kinetic energy of the upper part of the rod	N m
T_p	Kinetic energy of the payload	N m
V_L	Potential energy of the lower part of the rod	N m
V_a	Potential energy of the linear actuator	N m
V_u	Potential energy of the upper part of the rod	N m
V_p	Potential energy of the payload	N m
V_c	Potential energy of the cart	N m
α	Inclination angle	rad

Appendix 2: Single IP constants

$$C_1 = 4L_1 + L_1^2 + 4L_1L_u$$

$$C_2 = 4L_1 + 2L_u$$

$$C_3 = 4L_1^2 + 4L_u^2 + 8L_1L_u$$

$$C_4 = 4L_1 + 4L_u$$

$$C_5 = 2L_1 + L_u$$

$$C_6 = 2L_1 + 2L_u$$

$$C_7 = M_c + M_1 + M_a + M_u + M_m$$

$$C_8 = M_c + M_1 + M_a + M_u + M_m$$

$$C_9 = \frac{1}{2}M_1L_1^2 + \frac{1}{2}J_1 + \frac{1}{2}M_aL_a^2 + \frac{1}{2}J_a$$

$$C_{10} = M_1L_1 + M_aL_a$$

$$C_{11} = (M_c + M_1 + M_a + M_u + M_m)g$$

$$C_{12} = \frac{1}{2}M_u + \frac{1}{2}M_m + \frac{1}{2}M_u + \frac{1}{2}M_m$$

$$C_{13} = \frac{1}{2}(C_2 + C_4)M_u + \frac{1}{2}M_u(C_2 - 2L_g) + \frac{1}{2}M_m(C_4 - 2L_g)$$

$$C_{14} = C_9 + \frac{1}{2}C_1M_u + C_3M_m + \frac{1}{2}(M_u + M_m)L_g^2 - C_5M_uL_g + \frac{1}{2}C_1M_u + \frac{1}{2}M_mL_g^2 - C_6M_mL_g$$

$$C_{15} = C_{10} + C_5M_u + C_6M_m$$

$$C_{16} = M_u + M_m$$

$$C_{17} = C_5M_u + C_6M_m$$

$$C_{18} = C_{10} + C_{17}$$

$$C_{19} = C_{10} + M_u(C_5 + Q) + M_m(C_6 + Q)$$

$$C_{20} = C_{10}g + M_u g(C_5 + Q) + M_m(C_6 + Q)$$

$$L_u = C_5 + Q$$

$$L_m = C_6 + Q$$

$$J_m = M_m(Q^2 + (C_4 - 2L_g)Q + L_g^2 - 2L_gC_6 + C_3)$$

$$J_u = M_u(Q^2 + (C_2 - 2L_g)Q + L_g^2 - 2L_gC_5 + C_1)$$

Appendix 3: Double IP constants

$$C_1 = 0.125R_w(M_m + M_1 + M_{2l} + M_a + M_{2u} + M)$$

$$C_2 = 0.5(M_mL_1^2 + M_1L_{c1}^2 + M_{2l}L_1^2 + M_aL_1^2 + M_{2u}L_1^2 + ML_1^2)$$

$$C_3 = M_{2l}L_{c2}^2 + M_aL_a^2$$

$$C_4 = 0.5M_m(\cos \alpha + \sin \alpha) + 0.5M_1(\cos \alpha + \sin \alpha)$$

$$C_5 = M_{2l}L_1L_{c2} + M_aL_1L_a$$

$$C_6 = 0.5R_w(M_{2l}L_1 + M_aL_1 + M_{2u}L_1 + ML_1)$$

$$C_7 = R_w(M_{2l}L_{c1} + M_aL_a)$$

$$C_8 = 2L_{c2} + L_{2u}$$

$$C_9 = 2L_{c2} + 2L_{2u}$$

$$C_{10} = 4L_{c2}^2 + L_{2u}^2 + 4L_{2u}L_{c2}$$

$$C_{11} = 4L_{c2}^2 + 4L_{2u}^2 + 8L_{2u}L_{c2}$$

$$C_{12} = M_wR_w^2 + J_w$$

$$C_{13} = 2J_w + J_{IB}$$

$$\begin{aligned}
C_{14} &= M_1 L_{c1} + M_m L_1 + M_{2l} L_1 + M_a L_1 + M_{2u} L_1 + M L_1 \\
C_{15} &= M_{2l} L_{c2} + M_a L_a \\
C_{16} &= M_1 + M_m + M_{2l} + M_a + M_{2u} + M \\
C_{17} &= C_{15} + M_{2u} C_8 + M C_9 \\
C_{18} &= C_8 + C_9 \\
C_{19} &= M_{2u} + M \\
C_{20} &= 2C_8 M_{2u} + 2C_9 M \\
C_{21} &= C_3 + M_{2u} C_{10} + M C_{11} \\
C_{22} &= 2M_{2u} C_8 + 2M C_9 \\
C_{23} &= M_{2u} L_1 + M L_1 \\
C_{24} &= C_5 + M_{2u} L_1 C_8 + M L_1 C_9 \\
C_{25} &= M_{2u} R_w + M R_w \\
C_{26} &= C_7 + M_{2u} R_w C_8 + M R_w C_9 \\
C_{27} &= C_1 + C_{12} \\
C_{28} &= R_w (M_{2u} + M)
\end{aligned}$$

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 27 October 2016 Accepted: 5 May 2017

Published online: 16 June 2017

References

- Ahmad S, Tokhi MO. Modelling and control of a wheelchair on two wheels. In: Proceedings of the 2nd Asia International Conference on Modelling & Simulation (AICMS 08), Kuala Lumpur, Malaysia; 2008.
- Akesson J, Blomdell A, Braun R. Design and control of YAIP an inverted pendulum on two wheels robot, computer aided control system design. In: IEEE international conference on control applications. 2011. p. 2178–83.
- Almeshal AM, Goher KM, Tokhi MO. Modelling of two-wheeled robotic wheelchair with moving payload. In: Proceedings of the 14th international conference on climbing and walking robots and the support technologies for mobile machines (CLAWAR 2011), Paris, France. 2011. p. 731–41.
- Almeshal AM, Goher KM, Nasir AN, Tokhi MO. Hybrid spiral dynamic bacterial chemotaxis optimisation for hybrid fuzzy logic control of a novel two-wheeled robotic vehicle. In: Proceedings of the 15th international conference on climbing and walking robots and the support technologies for mobile machines (CLAWAR 2013), Sydney, Australia. 2011. p. 179–88.
- Askari M, Mohamed HAF, Moghavvemi M, Yang SS. Model predictive control of an inverted pendulum. In: Proceedings of the international conference for technical postgraduates (TECHPOS). 2009. p. 1–4.
- Balan R, Maties V, Hancu O, Stan S. A predictive control approach for the inverse pendulum on a cart problem. In: Proceedings of the IEEE international conference mechatronics and automation, vol. 2024. 2005. p. 2026–31.
- Balan R, Maties V, Stan S. A solution of the inverse pendulum on a cart problem using predictive control. In: Proceedings of the IEEE ISIE, Dubrovnik, Croatia. (2005). p. 63–8.
- Baloch M, Parent M (2003) Modelling and model verification of an intelligent self-balancing two wheeled vehicle for an autonomous urban transportation system. In: Proceedings of conference on computer intelligence, robot and autonomous system, Singapore.
- Banchs RE, Klie H, Rodriguez A, Thomas SG, Wheeler MF. A neural stochastic multiscale optimization framework for sensor-based parameter estimation. *Integr Comput Aided Eng.* 2007;14(3):213–23.
- Biswas A, Dasgupta S, Das S, Abraham A. Synergy of PSO and bacterial foraging optimization: a comparative study on numerical benchmarks. *Adv Soft Comput.* 2007;44:255–63.
- Biswas A, Dasgupta S, Das S, Abraham A. A synergy of differential evolution and bacterial foraging optimization for global optimization. *Neural Netw World.* 2007;17(6):607–26.
- Campomanes-Álvarez BR, Cordon O, Damas S. Evolutionary multi-objective optimization for mesh simplification of 3D open models. *Integr Comput Aided Eng.* 2013;20(4):375–90.
- Carro-Calvo L, Salcedo-Sanz S, Ortiz-García EG, Portilla-Figueras A. An incremental-encoding evolutionary algorithm for color reduction in images. *Integr Comput Aided Eng.* 2010;17(3):261–9.
- Chen Y, Lin W. An improved bacterial foraging optimization. In: Paper presented at the 2009 IEEE international conference on robotics and biomimetics, ROBIO 2009, pp. 2057–2062.
- Das TK, Venayagamoorthy GK, Aliyu UO. Bio-inspired algorithms for the design of multiple optimal power system stabilizers: SPPSO and BFA. *IEEE Trans Ind Appl.* 2008;44(5):1445–57.
- Dasgupta S, Das S, Abraham A, Biswas A. Adaptive computational chemotaxis in bacterial foraging optimization: An analysis. *IEEE Trans Evol Comput.* 2009;13(4):919–41.
- Dwibhashya SP, Rambabu C. Optimal bidding strategy in competitive market using GA & BFA: a comparative study. *Int J Eng Res Dev.* 2012;3(6):83–94.
- Farhat IA, El-Hawary ME. Short-term hydro-thermal scheduling using an improved bacterial foraging algorithm. In: Paper presented at the 2009 IEEE electrical power and energy conference (EPEC 2009), 2009. pp. 1–6.
- Ghaffar HIA, Ebrahim EA, Azzam M. Design of PID controller for power system stabilization using hybrid particle swarm-bacteria foraging optimization. *WSEAS Trans Power Syst.* 2013;8(1):12–23.
- Goher KM, Tokhi MO, Ahmad S. A new configuration of two-wheeled vehicles: towards a more workspace and motion flexibility. In: Proceedings of the 4th IEEE international systems conference, San Diego, CA, USA, April 5–8, 2010.
- Hooshmand RA, Parastegari M, Morshed MJ. Emission, reserve and economic load dispatch problem with non-smooth and non-convex cost functions using the hybrid bacterial foraging-Nelder–Mead algorithm. *Appl Energy.* 2012;89(1):443–53.
- Huang W, Lin W. Parameter estimation of wiener model based on improved bacterial foraging optimization. In: Paper presented at the 2010 international conference on artificial intelligence and computational intelligence (AICI 2010), 1, 2010. pp. 174–178.
- Jia L, Wang Y, Fan L. Multiobjective bilevel optimization for production–distribution planning problems using hybrid genetic algorithm. *Integr Comput Aided Eng.* 2014;21(1):77–90.
- Kim DH, Abraham A, Cho JH. A hybrid genetic algorithm and bacterial foraging approach for global optimization. *Inf Sci.* 2007;177(18):3918–37.
- Kim DH. Hybrid GA–BF based intelligent PID controller tuning for AVR system. *Appl Soft Comput.* 2011;11(1):11–22.
- Korani WM. Bacterial foraging oriented by particle swarm optimization strategy for PID tuning. In: Proceedings of the 2008 GECCO conference companion on genetic and evolutionary computation (GECCO'08), Atlanta, 12–16 July 2008. p. 1823–6.
- Luo Z, Shi Z, Hu P, Hao L, Liu H. Error analysis of a inverted pendulum on an inclined rail. In: Proceedings of the 2nd IEEE international conference on advanced computer control (ICACC 2010), 3, 2010. pp. 52–55.
- Majhi B, Panda G, Choubey A. On the development of a new adaptive channel equalizer using bacterial foraging optimization technique. In: Proceedings of 2006 annual IEEE india conference, INDICON, New Delhi, 2006. p. 1–6.
- Majhi R, Panda G, Majhi B, Sahoo G. Efficient prediction of stock market indices using adaptive bacterial foraging optimization (ABFO) and BFO based techniques. *Expert Syst Appl.* 2009;36(6):10097–104.
- Nasir ANK, Tokhi MO, Abd Ghani NM, Ahmad MA. A novel hybrid spiral dynamics bacterial-foraging algorithm for global optimization with application to control design. In: Proceedings of 12th annual workshop on computational intelligence (UKCI2012), Edinburgh, UK, 2012. p. 1–7.
- Nasir ANK, Tokhi MO, Abd Ghani NM, Ahmad MA. A novel hybrid spiral dynamics bacterial-chemotaxis algorithm for global optimization with application to controller design. In: Proceedings of 2012 UKACC international conference on control, Cardiff, UK, 3–5 September 2012. p. 753–8.

32. Nasir ANK, Tokhi MO, Abd Ghani NM, Raja Ismail RMT. Novel adaptive spiral dynamics algorithms for global optimization. In: Proceeding of 11th IEEE international conference on cybernetic intelligent system 2012, Limerick, Ireland, 23–24 August 2012.
33. Nasir ANK, Tokhi MO. A novel hybrid bacteria-chemotaxis spiral-dynamics algorithm with application to modelling of a flexible system. In: Proceedings of 2013 5th international conference on modeling, simulation and applied optimization (ICMSAO'13), Hammamet, Tunisia, April 28–30, 2013.
34. Nasir ANK, Tokhi MO. Novel adaptive bacteria foraging algorithms for global optimization. In: Proceedings of 2013 5th international conference on modeling, simulation and applied optimization (ICMSAO'13), Hammamet, Tunisia, April 28–30, 2013.
35. Nasir ANK, Tokhi MO. A novel hybrid bacteria-chemotaxis spiral-dynamic algorithm with application to modelling of flexible systems. *Eng Appl Artif Intell.* 2014;33:31–46.
36. Nasir ANK, Tokhi MO, Ghani NMA. Novel adaptive bacterial foraging algorithms for global optimization with application to modelling of a TRS. *Expert Syst Appl.* 2015;42:1513–30.
37. Nasir ANK, Tokhi MO. Novel metaheuristic hybrid spiral-dynamic bacteria-chemotaxis algorithms for global optimisation. *Appl Soft Comput.* 2015;27:357–75.
38. Neri F, Cotta C. Memetic algorithms and memetic computing optimization: a literature review. *Swarm Evol Comput.* 2012;2:1–14.
39. Niu B, Fan Y, Zhao P, Xue B, Li L, Chai Y. A novel bacterial foraging optimizer with linear decreasing chemotaxis step. In: Paper presented at the 2010 2nd international workshop on intelligent systems and applications (ISA 2010); 2010.
40. Panigrahi BK, Ravikumar PV. Bacterial foraging optimisation: Nelder–Mead hybrid algorithm for economic load dispatch. *IET Gener Transm Distrib.* 2008;2(4):556–65.
41. Panikhom S, Sarasiri N, Sujitjorn S. Hybrid bacterial foraging and tabu search optimization (BTSSO) algorithms for Lyapunov's stability analysis of nonlinear systems. *Int J Math Comput Simul.* 2010;4(3):81–9.
42. Passino KM. Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Syst Mag.* 2002;22:52–67.
43. Rokni S, Fayek AR. A multi-criteria optimization framework for industrial shop scheduling using fuzzy set theory. *Integr Comput Aided Eng.* 2010;17:175–1963.
44. Sanyal N, Chatterjee A, Munshi S. An adaptive bacterial foraging algorithm for fuzzy entropy based image segmentation. *Expert Syst Appl.* 2011;38(12):15489–98.
45. Sathya PD, Kayalvizhi R. Optimal segmentation of brain MRI based on adaptive bacterial foraging algorithm. *Neurocomputing.* 2011;74(14–15):2299–313.
46. Sedano J, Berzosa A, Villar JR, Corchado E, de la Cal E. Optimising operational costs using Soft Computing techniques. *Integr Comput Aided Eng.* 2011;18(4):313–25.
47. Sinha N, Lai LL, Saikia LC, Malakar T. Hybrid bacteria foraging-de based algorithm for economic load dispatch with non-convex loads. In: Proceedings of 2010 international conference on machine learning and cybernetics (ICMLC), Qingdao, 11–14 July 2010, p. 3206–12.
48. Smith JF III, Nguyen TVH. Autonomous and cooperative robotic behavior based on fuzzy logic and genetic programming. *Integr Comput Aided Eng.* 2007;14(2):141–59.
49. Supriyono H, Tokhi MO. Parametric modeling approach using bacterial foraging algorithms for modeling of flexible manipulator systems. *Eng Appl Artif Intell Syst.* 2012;25(5):898–916.
50. Tamura K, Yasuda K. Spiral multipoint search for global optimization. In: Proceedings of 10th international conference on machine learning and applications, 2011. p. 470–5.
51. Tamura K, Yasuda K. Primary study of spiral dynamics inspired optimization. *IEE J Trans Electr Electron Eng.* 2011;6(51):98–100.
52. Tang WJ, Li MS, Wu QH, Saunders JR. Bacterial foraging algorithm for optimal power flow in dynamic environments. *IEEE Trans Circuits Syst I.* 2008;55(8):2433–42.
53. Tao H, Mohamad ZJ, Ahmed MM, Abdalla AN, Jing W. A wavelet-based particle swarm optimization algorithm for digital image watermarking. *Integr Comput Aided Eng.* 2012;19(1):81–91.
54. Venkaiah C, Vinod Kumar DM. Fuzzy adaptive bacterial foraging congestion management using sensitivity based optimal active power re-scheduling of generators. *Appl Soft Comput.* 2011;11(8):4921–30.
55. Wu J-W, Tseng JCR, Tsai W-N. A hybrid linear text segmentation algorithm using hierarchical agglomerative clustering and discrete particle swarm optimization. *Integr Comput Aided Eng.* 2014;21(1):35–46.
56. Xu X, Liu YH, Wang AM, Wang G, Chen HL. A new adaptive bacterial swarm algorithm. In: Paper presented at the 2012 8th international conference on natural computing (ICNC 2012), Chongqing China; 2012. pp. 991–995.
57. Yan X, Zhu Y, Chen H, Zhang H. Improved bacterial foraging optimization with social cooperation and adaptive step size. *Intelligent Computing Technology, Lecture notes in computer science, vol. 7389.* 2012. pp. 634–640.
58. Zang H, Zhang S, Hapeshi K. A review of nature-inspired algorithms. *J Bionic Eng.* 2010;7:5232–7.
59. Zhong Y, Lin J, Ning J, Lin X. Hybrid artificial bee colony algorithm with chemotaxis behavior of bacterial foraging optimization algorithm. In: Proceeding of seventh international conference on natural computation, Shanghai, 2011, p. 1171–4.

Submit your manuscript to a SpringerOpen® journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com