International Journal of Engineering & Technology, 7 (3.13) (2018) 27-32

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Adaptive Backsteeping SMC with Cuckoo Search Algorithm for Two Wheeled Self Balancing Robot

Asst.Pro.Dr.Ekhlas H.Karam [1]*, Noor.M.Mjeed [2]

[1] *Dept. of Computer Engineering, Mustansiriayh University; Iraq.*
[2] *Dept. of Computer Engineering, Mustansiriayh University; Iraq. Email: noormjeed@yahoo..com*
**Corresponding author E-mail: ek_karam@yahoo.com*

Abstract

In this paper, a robust Radial Basis Function (RBF) Backstepping Sliding Mode controller (BS-SMC) is successfully developed for the attitude stabilization and tracking the trajectory of two wheeled self-balancing mobile robot under the external disturbance and uncertainty. The design of BS control is derived based on Lyapunov function to ensure the stability of the robot system and the SMC is designed with a switching function in order to attenuate the effects of the disturbances, the auto-adjustable RBF inference system is suggested to estimate the equivalent component of the BS-SMC to treat the model dependency problem and robustness improvement. Also a cuckoo search (CS) optimization algorithm is used to determine the optimal values of the backsteeping sliding mode controller. Numerical simulations show the efficiency of the suggested controller in handling the balance and tracking problems of the two wheeled self-balancing mobile robot

Keywords: *BS; BS-SMC;CS; RBFNN; Two wheeled self balancing robot.*

1. Introduction

Two wheeled self-balancing robots (TWSBR) is kind of mobile robot that can move from one place to another autonomously, this robot has the special feature of moving around freely within a predefined workspace to achieve their desired goals. The TWSBR is one of the most vastly researched robots because of it unstable, nonlinear, multi-output and suffers from balancing problem. On the other hand its small size, flexibility, low cost and so on compared with other mobile robots [1, 2], the TWSBR can be used in many applications, and it can be access to a hazardous or confined environment which is difficult to maneuver around a multi-wheeled vehicle or track. It can be controlled remotely or independently [3]. Robot control deals with the problem of balancing that must be developed by the robotic action in order for the robot to go at a wanted position, and tracking the desired trajectory problem, and, in general, to perform some task with wanted performance requirements. Various controllers were proposed to address these problems. The sliding Mode Controller and backstepping (SMC, BS) are considered as one of the most common controllers that are often used to control the nonlinear and high order and complex systems under certain uncertainties of parameters changes and external disturbance [4]. A wide range of SMC [5, 6] with BS controllers type have been proposed in the literature to balance and trajectory tracking of the TWSBR. Here are some of the past works, Nguyen and Son [7, 8] design BS-SMC for TWSBR trajectory tracking. A hybrid PD and adaptive BS controller is suggested by [9], while [10, 11] design an adaptive PID, PD backstepping controller.
In this paper, a two vigorous BS-SMC for self-adjusting robot is designed. Since the BS-SMC controller has the best execution to track the desired trajectory and reject the disturbance. Also in this paper, a radial basis function neural network (RBFNN) is combined with the BS-SMC in order to enhance the performance of it. Finally, the cuckoo search (CS) algorithm has been used to tune the BS-SMC parameters.
The organization of this paper is as follows. In section 2, the TWSBR dynamic model is presented, BS-SMC design is explained in section 3. Selection parameters based on the CS algorithm are explained in section 4. Section 5 explains the integration of the radial basis function with the BS-SMC. Section 6 shows the proposed controller's efficiency with simulation results. Finally, conclusions are provided in section 7.

2. Two Wheeled Self Balancing Dynamic Robot Model

The body of two-wheel self-balancing robot consists of two part: the first part is two wheels of the same type, moment of inertia and mass, the second part is rigid chassis which contains the power and control hardware as well as a type of battery as shown in Fig. 1

Fig. 1: Diagram of the chassis two wheeled self-balancing robot [12].

The dynamic model of the two wheeled self-balancing robot can be described by introducing the prevailing differential equations representing the dynamics of the two-wheeled robotic system. This model for balancing robot is derived using Newton's equations of motion. In general, the equations of the dynamic nonlinear model of TWSBR can be written as [13]:

$$\ddot{x} = f_x(x,\theta,t) + b_x u(t) + d(t) \tag{1}$$

$$\ddot{x} = -\frac{M_p l}{(2M_W + \frac{2I_o}{r^2} + M_p)}\ddot{\theta}\cos\theta - \frac{2K_m K_a}{Rr^2(2M_W + \frac{2I_o}{r^2} + M_p)}\dot{x} + \frac{2K_m}{R(2M_W + \frac{2I_o}{r^2} + M_p)}V_a + \frac{M_p l}{(2M_W + \frac{2I_o}{r^2} + M_p)}\dot{\theta}^2\sin\theta \tag{2}$$

And

$$\ddot{\theta} = f_\theta(x,\theta,t) + b_\theta u(t) \tag{3}$$

$$\ddot{\theta} = \frac{M_p l}{(I_p + M_p l^2)}\ddot{x}\cos\theta + \frac{2K_a K_m}{Rr(I_p + M_p l^2)}\dot{x} - \frac{M_p l}{(I_p + M_p l^2)}\sin\theta - \frac{2K_m}{R(I_p + M_p l^2)}V_a \tag{4}$$

Where x is the linear position, θ is the lilt angle of the TWSBR, $u(t) \in R$ is the control action, $f_x(.)$, $f_\theta(.)$ are the unknown nonlinear function, for robot position and angle respectively, b_x, b_θ are the control gain of the position and angle respectively, and $d(t)$ is the unknown external disturbance. The parameters of these equations for chassis robot are given by Table1.

Table 1: The parameter values of the two wheeled self-balancing mobile robot [13, 14].

Symbol	Definition	Parameter
M_p	Mass of body	6 kg
M_W	Mass of wheel	0.3 kg
K_a	Back EMF constant	0.0458 Vs/rad
K_m	Motor torque constant	0.0458 Nm/Amp
l	Length to the body's center of mass	0.2 m
R	Nominal Terminal Resistance	2.49 Ω
r	Radius of wheel	0.077 m
g	Gravity	9.81 m/s^2
I_o	Inertia of the wheel	0.0017 $kg.m^2$
I_p	Inertia of the body	0.29 $kg.m^2$
x	Position of the chassis	m
θ	angle of the chassis	rad

3. Backsteeping Sliding Mode Controller Design

The backsteeping control (BS) and sliding mode control (SMC) are integrated to confirmation (BS-SMC) in order to obtain the characteristics of the controllers together, and hence realizes the robust control for uncertain systems.

In this paper, the main purpose of designing the BS-SMC is to handle the problems of the two wheeled self balancing mobile robot in balancing itself and in tracking the trajectories.

The procedures for design of the BS-SMC can be described by the following steps:

Step 1: Define the error and the derivative of tracking error by:

$$e_1 = x_d - x \tag{5}$$

$$\dot{e}_1 = \dot{x}_d - \dot{x} \tag{6}$$

where x_d is the desired trajectory, then select the first Lyapunov function as:

$$V_1 = \frac{1}{2}e_1^T e_1 \tag{7}$$

Therefore,

$$\dot{V}_1 = e_1^T \dot{e}_1 = e_1^T(\dot{x}_d - \dot{x}) \tag{8}$$

$$\tau = \dot{x}_d + \lambda_1 e_1 \tag{9}$$

Where is λ_1 a positive constant, by replacing the virtual control by the desired value, Eq. (8) becomes:

$$\dot{V}_1 = -\lambda_1 e_1^T e_1 \le 0 \tag{10}$$

$$e_2 = \dot{x} - \tau = \dot{x} - \dot{x}_d - \lambda_1 e_1 \tag{11}$$

Define a sliding surface in terms of the error such as:

$$s = e_2 = \dot{x} - \tau = \dot{x} - \dot{x}_d - \lambda_1 e_1 \tag{12}$$

Step 2: Select the second Lyapunov function as:

$$V_2 = V_1 + \frac{1}{2}s^T s \tag{13}$$

$$\dot{V}_2 = V_1 + \frac{1}{2}s^T\dot{s}$$
$$= e_1^T(-e_2 - \lambda_1 e_1) + s^T(f(x,\theta,t) + b_x u(t) + d - \ddot{x}_d - \lambda_1\dot{e}_1)$$
$$= -\lambda_1 e_1^T e_1 + s^T(-e_1 + f(x,\theta,t) + b_x u(t) + d - \ddot{x}_d - \lambda_1\dot{e}_1) \tag{14}$$

Step 3: Since the external disturbance is un known, a backstepping control can be suggested as:

$$u_{x_{eq}} = \frac{1}{b_x}(-f_x(x,\theta,t) - \lambda_2 s - e_1 + \lambda_1\dot{e}_1 - \ddot{x}_d) \tag{15}$$

$$u_x = u_{x_{eq}} + u_{s_x} \tag{16}$$

Where $u_{x_{eq}}$ the equivalent part and u_{s_x} is the discontinuous part $= ksat(s))$,

$$u_x = \frac{1}{b_x}(-f_x(x,\theta,t) - \lambda_2 s - e_1 + \lambda_1\dot{e}_1 - \ddot{x}_d + ksat(s)) \tag{17}$$

The same previous above steps () are used to derive the control action u_θ for θ and final equations for both the equivalent will be as [13]:

$$u_{\theta_{eq}} = \frac{1}{b_x}(-f_\theta(x,\theta,t) - \lambda_2 s - e_1 + \lambda_1 \dot{e}_1 - \ddot{\theta}_d) \quad (18)$$

$$u_\theta = \frac{1}{b_\theta}(-f_\theta(x,\theta,t) - \lambda_2 s - e_1 + \lambda_1 \dot{e}_1 - \ddot{\theta}_d + k sat(s)) \quad (19)$$

The final TWSBR control signal is given by:

$$u = u_x + u_\theta \quad (20)$$

The parameters λ_1, λ_2, k for robot position and angle needed to be assigned. These parameters effect on the performance and even the stability of the TWSBR system.

4. Parameters Selection Based on CS Algorithm

Cuckoo Search (CS) algorithm is adopted to obtain the appropriate parameters values that ensure the best performance of the BS-SMC and hence improve robotic system performance by (decrease the settling time, cancel the effective of the external disturbance and parameters uncertainty). The CS algorithm is a nature-inspired metaheuristic algorithm and it is essentially based on some cuckoo types brood parasitism and enhanced by levy flights rather than by simple isotropic random walks. In CS algorithm, each cuckoo lays only one egg at a time and put it in a random nest; best nests with higher eggs fitness approved over to the next eggs generations. The cuckoo's egg discovered according to probability $p_a \in [0,1]$; if the host bird discovers the strange egg it will get rid of the cuckoo egg, or abandon the original nest and build another new nest. In CS algorithm, each cuckoo egg represents a candidate solution while generating new solutions performed as in Eq.(21) [15,16].

$$nest^{t+1}{}_i = nest^t{}_i + \alpha \oplus levy(\tau) \quad (21)$$

Where $nest_i^t$ is the location of the t generation of the i-th parasitic nest, and $\alpha = 0.01$ represents the step size vector that is associated to the scales of the optimization problem, $\oplus$ is an entry wise multiplications Levy function *randn()* is the random function which satisfies Gauss distribution,Levy distribution is difficult; the equivalent calculation can be realized by Mantegana algorithm , which is given by

$$nest^{t+1}{}_i = nest^t{}_i + stepsize \oplus rand() \quad (22)$$
$$i = 1,2,3...,N$$

$$stepsize = \alpha.step \oplus (nest^t{}_i - nest^t{}_{best})$$

$$step = \frac{z}{|\psi|^{1/q}}, z \approx N(0,\sigma^2{}_z), \psi \approx N(0,\sigma^2{}_\psi) \quad (24)$$

Where $\sigma_\psi - 1, q = 3/2$,and σ_z and can be written as:

$$\sigma_z = \left\{ \frac{\Gamma(1+q).\sin \Pi q/2)}{\Gamma[(1+q)/2]q.2^{(q-1)/2}} \right\}^{1/q} \quad (25)$$

The flowchart which is shown in Fig.2 explains the steps for obtaining the optimal control parameters by the CS algorithm.

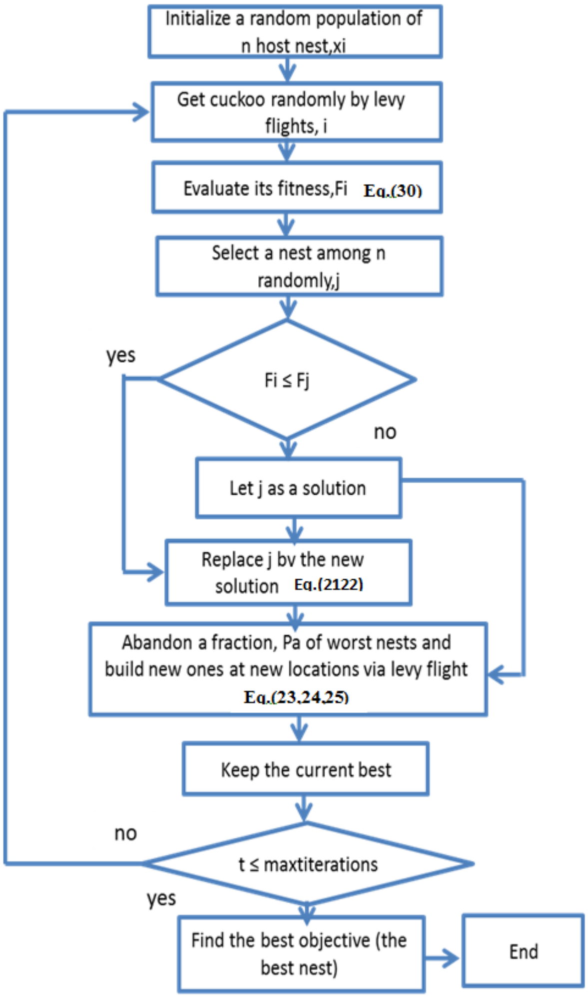


Fig.2: Flow chart of cuckoo optimization algorithm.

5. Backsteeping Sliding Mode Controller with Radial Basis Function

In order to improve the performance of BS-SMC, Radial Basis Function Neural Network (RBFNN) is used. This neural network consists of the nodes of input layer, hidden layer and output layer. The input nodes are just buffer units, which pass the signal without changeable them. The output units are linear units which add the signals fed to them. The hidden units are nonlinear activation functions (it's called Gaussian function) as shown in Fig.3. The number of the hidden layer nodes can be changed based on the required efficiency of the RBF neural network. The input of the Gaussian functions, are created by the distance between the error input value $x,\theta = [e_1, \dot{e}_1]$ and the Prototype input c_{ij} [17, 18].

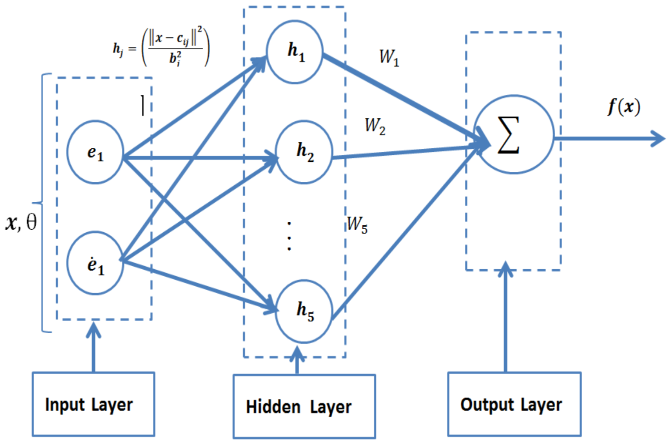


Fig .3. The structure of the three-layer RBF neural network.

The RBF neural network is used to estimate the unknown functions f_x, $f_\theta(.)$ by the following equations:

$$\hat{f}_x(x) = \hat{W}^T h(x) \tag{26-a}$$

And for angle

$$\hat{f}_\theta(x) = \hat{W}^T h(x) \tag{26-b}$$

Where $h(x)$ is the Gaussian function of the neural network, w weight vector in connect between hidden-layer and output layer are specified as constant gains 1.0 is used to adjust the weightings for searching the optimal weighting values and obtaining the stable convergence property [19].The $h(x)$ is given by:

$$h_j = \exp\left[\frac{\left\|x - c_{ij}\right\|^2}{b^2{}_j}\right] \tag{27}$$

Where $c_{ij} = 0.5, b_j = 1$ center vector, spread factor, i is the input number of the network $x, \theta = [e_1, \dot{e}_1]$ is the input state of the network, j is the number of hidden layer nodes in the network, $h = [h_1, h_2, h_3, h_4, h_5]$ is the output of Gaussian function.

Based on gradient descent method, the weight adjustment algorithm is given by:

$$\nabla w = w(t-1) + \nabla w(t) + \rho(w(t) - w(t-1)) \tag{28}$$

$$\nabla w = \gamma(y - y_d) * f$$

γ is the learning rate, ρ the momentum factor, $\gamma, \rho \in$ [0, 1], used in the work $\gamma, \rho = 0.1, 0.7$.

With replacing the ($f_x(.)$, $f_\theta(.)$) by neural function $\hat{f}_x(.)$, $\hat{f}_\theta(.)$ then the position and angle control signal Eqs.(17&19) becomes as:

$$u_x = \frac{1}{b_x}(-\hat{f}_x(x,\theta,t) - \lambda_2 s - e_1 + \lambda_1 \dot{e}_1 + \ddot{x}_d - ksat(s))$$

(29-a)

$$u_\theta = \frac{1}{b_\theta}(-\hat{f}_\theta(x,\theta,t) - \lambda_2 s - e_1 + \lambda_1 \dot{e}_1 + \ddot{\theta}_d - ksat(s))$$

(29-b)

The complete block diagram for the TWSBR with the proposed controller (RBF-BS-SMC) and based on CS algorithm is shown in Fig.4.

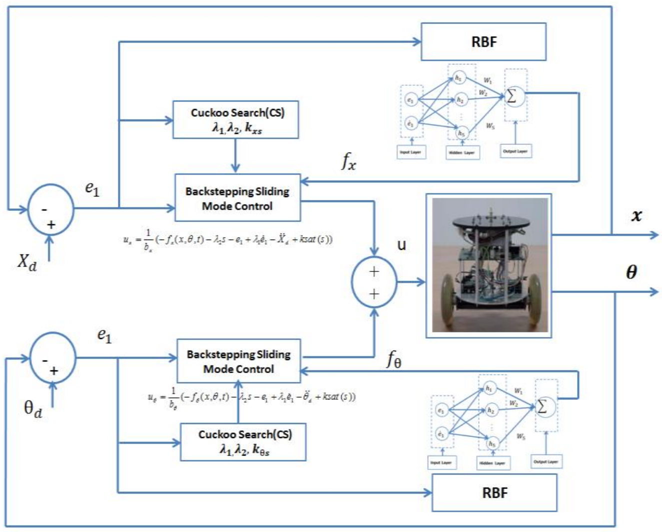


Fig. 4: The complete block diagram for the self-balancing mobile robot with the suggested controller (RBF-BS-SMC) with CS algorithm.

6. Simulation Result and Dissection

With the facility available in the Matlab software version (R2016a) and to illustrate the efficiency of the proposed (RBF-BS-SMC with CS) as compared with (SMC, BS, BS-SMC with CS, two simulation cases are carried out to test the performance of the TWSBR by linear and nonlinear trajectories with external disturbance and with 20% uncertainty in $\hat{f}_{x,\theta}(.)$. The parameters of the CS algorithm are given in Table 2, the CS fitness function integral square error (ISE) is given by:

$$F = ISE = \int_0^\infty e^2 d(t) \tag{30}$$

final optimal BS-SMC parameters obtain by CS algorithm are given in Table 3

Table 2: The CS Parameters.

No.of iterations	No.of nests	Dimension	Discovery rate p_a
100	25	6	0.25

Table 3: Final BS-SMC parameters values.

Parameters	BS-SMC		
	λ_1	λ_2	k
position (x)	4	2	44
angle (θ)	2	1.2	1.78

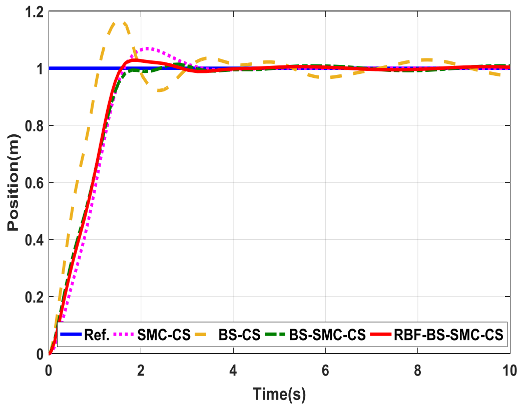
The two simulation cases are:

- **Simulation results for linear step trajectory**

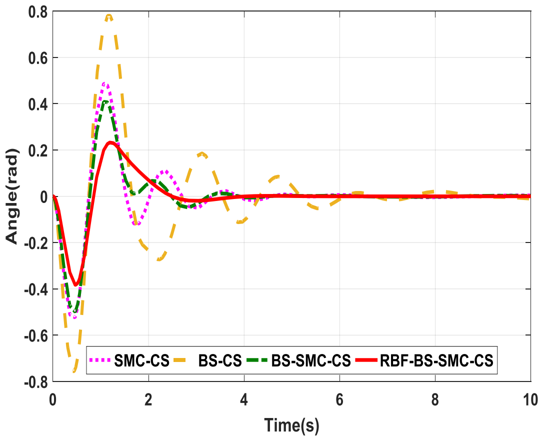
The result of this simulation (position, angle, and control signal) are illustrated in Fig.5, these result show that the performance of robot with RBF-BS-SMC and CS is more efficient than with (SMC-CS, BS-CS, BS-SMC-CS), where the TWSBR with this controller flow the designed trajectory very fast with very small overshoot and zero steady state error (Fig.5-a), small pitching angle (Fig.5-b), and very smooth control signal (Fig.5-c)

- **Simulation result for nonlinear trajectory**

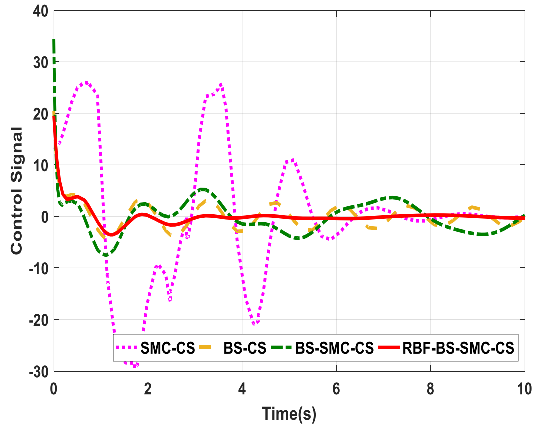
The nonlinear signal $(x_d = \sin(\pi t/2))$ is used to test controlled TWSBR system and the result for this simulation are given by Fig.6 which is show that the (TWSBR with RBF-BS-SMC and CS) is remaining stable even with 20% uncertainty and external disturbance and it is flow the desired input signal very fast with no error (Fig.6-a), small pitching angle (Fig.6-b), and smooth control signal (Fig.6-c) than the other controller (SMC- CS, BS- CS, BS-SMC-CS, RBF-BS-SMC).



(a)

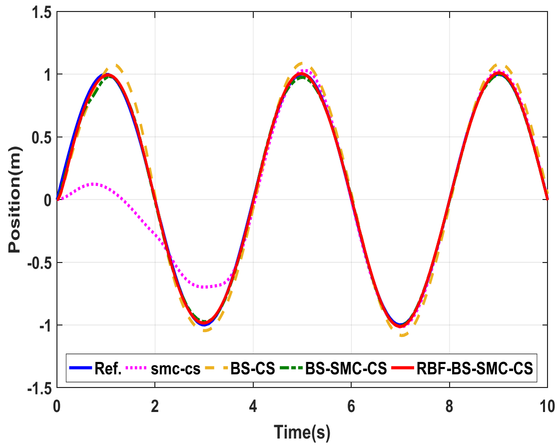


(b)

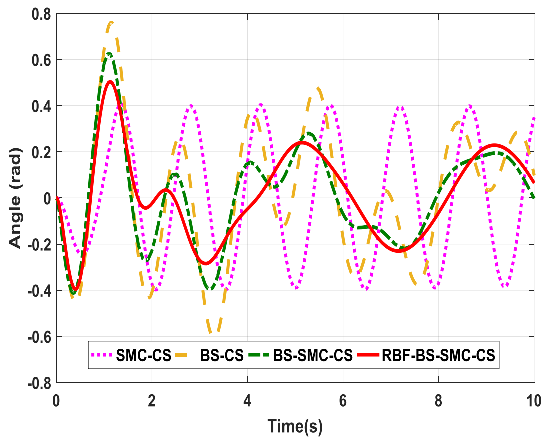


(c)

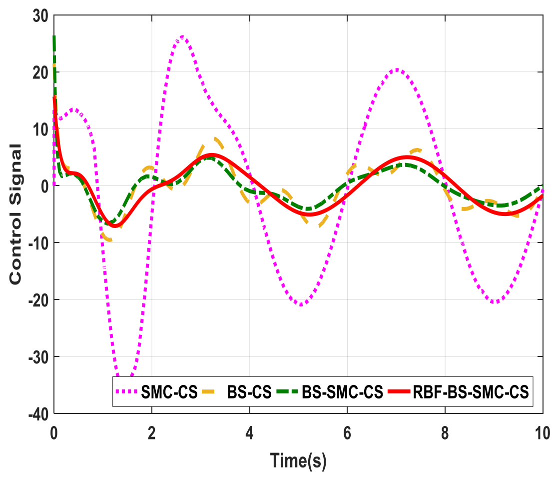
Fig.5: Simulation result of the TWSBR with (SMC-CS, BS-CS, BS-SMC-CS, RBF-BS-SMC-CS) for linear input with (uncertainty and external disturbance). (a): Position, (b): Angle, (c): control signal.



(a)



(b)



(c)

Fig. 6: Simulation result of the TWSBR with (SMC-CS, BS-CS, BS-SMC-CS, RBF-BS-SMC-CS) for nonlinear input signal with (uncertainty and external disturbance). (a): Position, (b): Angle, (c): control signal.

7. Conclusions

In this paper, a BS-SMC controller is designed for controlling the (position and pitching angle) of the TWSBR. The parameters of this controller are determine based on CS algorithm. Also, the RBF neural network is used to enhance the efficiency of the BS-SMC and hence the performance of the self-balancing mobile robot will be improve. Matlab simulation results showed the efficiency of the proposed controllers (BS-SMC, and RBF-BS-SMC) with CS in handling the tracking and balancing problems under external disturbance and uncertainty especially the RBF-BS-SMC which gave a high response speed as compared to (SMC- CS, BS- CS, BS-SMC-CS,BS-SMC) with CS only.

Acknowledgement

All thanks and praise be to Allah, the Compassionate, and the Merciful for all His Blessings.

I would like to express my deep sense of gratitude and respect to my supervisor, Asst. Prof. Dr. Ekhlas H. Karam, for the patient, excellent guidance, encouragement, and support she provided throughout my time as his student.

References

[1] Liu, K., Bai, M., & Ni, Y., (2011), Two-wheel self-balanced car based on Kalman filtering and PID algorithm, IEEE 18th International Conference on Industrial Engineering and Engineering Management, *Changchun*, pp. 281-285.
[2] Sun, L. & GAN J. (2010), Researching of Two-Wheeled Self-Balancing Robot Base on LQR Combined with PID,pp. 1-5.
[3] Memarbashi, Hamid R, & Jen-YuanChang (2011), Design and Parametric Control of Co-axes Driven Two-Wheeled Balancing Robot. New Zealand, the Massey University,MSc Thesis June 01.
[4] Kachroo, P., & Tomizuka, M.. (1996), Chattering Reduction and Error Convergence in the Sliding-Mode Control of a Class of Nonlinear Systems. IEEE TRANSACTIONS ON AUTOMATIC CONTROL AC,41 (7): 1063-1067.
[5] Fuquan, D., Fangxing, L., Yang, B., Wenzeng, G., Chengguo, Z., Xueshan, G.,(2012), Development of a Coaxial Self-Balancing Robot Based on Sliding Mode Control. IEEE International Conference on Mechatronics and Automation (ICMA),1241-1246.
[6] Song, Y., Guo, L., Xing, B., (2016), Cascade sliding mode and RBF network self-adaption controller design for a two-wheeled robot. 1659-1663.
[7] Nguyen Gia Minh Thao; Duong Hoai Nghia, Nguyen Huu Phuc (2010), A PID Backstepping Controller for Two-Wheeled Self-Balancing Robot.78-81.
[8] Son, N. N., & Anh, H. P. H. (2014), Adaptive Backstepping Self-balancing Control of a Two-wheel Electric Scooter. International Journal of Advanced Robotic Systems,11(10):1-11.
[9] Anh, H. P. H., & Son, N. N., (2015), Hybrid PD and adaptive backstepping control for self-balancing two-wheel electric scooter. Journal of Computer Science and Cybernetics, 30(4): 347–360.
[10] Tsai, C.-C., Shang-Yu, J., & Shih-Min, H., (2010), Trajectory tracking of a self-balancing two-wheeled robot using backstepping sliding-mode control and fuzzy basis function networks. IEEE/RSJ International Conference on Intelligent Robots and Systems ,3943-3948.
[11] Esmaeili, N., Alfi, A., & Khosravi, H.(2017), Balancing and Trajectory Tracking of Two-Wheeled Mobile Robot Using Backstepping Sliding Mode Control: Design and Experiments. journal of Intelligent and Robotic Systems: Theory and Applications, 87(3): 601-613.
[12] Qin, Y., Liu, Y., Zang, X., & Liu, J., (2011),Balance control of two-wheeled self-balancing mobile robot based on TS fuzzy model. International Forum on Strategic Technology (IFOST), 406-409.
[13] Prakash, K., &Thomas, K., (2016), Study of Controllers for A Two wheeled Self-Balancing Robot. International Conference on Next Generation Intelligent Systems (ICNGIS),1-7.
[14] Jamil, O., Jamil, M., Ayaz, Y., Ahmad, K., (2014), Modeling, Control of a Two-Wheeled Self- Balancing Robot. International Conference on Robotics and Emerging Allied Technologies in Engineering (iCREATE), 191-199.
[15] Qu, C., & He, W., (2015), A double Mutation Cuckoo Search Algorithm for Solving Systems of Nonlinear Equations. International Journal of Hybrid Information Technology,8(12): 433-448.
[16] Yang, Xin-She, & Deb, Suash.,(2009), Cuckoo search via Lévy Flights. IEEE Publications, 210-214.
[17] Wang, R., Zhao, Y.,& Sun, J.(2016), Application of optimized RBF neural network in ship's autopilot design. IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC),1642-1646.
[18] Yong Tao, Jiaqi Zheng, &Yuanchang Lin (2016), A Sliding Mode Control-based on a RBF Neural Network for Deburring Industry Robotic Systems. International Journal of Advanced Robotic Systems, 13(1): 1-10.
[19] Hung-Ching, L., Cheng-Hung, T.,& Ming-Hung, C., (2010), Radial Basis Function Neural Network with Sliding Mode Control for Robotic Manipulators. IEEE International Conference on Systems, Man and Cybernetics-SMC, 1209-1215.