Control and Cybernetics

vol. **47** (2018) No. 4

Self-adaptive grey wolf optimization based adaptive fuzzy aided sliding mode control for robotic manipulator

by

Sunil C. Kalshetti and S. K. Dixit

Walchand Institute of Technology
Electronics Engineering Department, Solapur, Maharashtra, India
sunilkalshettic@gmail.com
dixitsk1@yahoo.com

Abstract: As the robotic manipulators are highly nonlinear, it is a challenging task to design, in particular, the PUMA 560 robotic arm with acceptable performance. This paper intends to show the design and development of an adaptive sliding mode controller (SMC) for a robotic manipulator. Since it is not realistic to match the SMC operations with the system model at every time instant, this paper adopts fuzzy inference to replace the system model. This approach successfully achieves the objectives of the experiment, carried out in two stages. In the first stage, it acquires the precise characteristics of the system model for the diverse samples and adequately represents the robotic manipulator. Subsequently, we derive the acquired characteristics in the form of fuzzy rules. In the second stage, we represent the derived fuzzy rules on the basis on adaptive fuzzy membership functions. Further, the approach introduces the self-adaptiveness into a recent algorithm called Grey Wolf Optimization (GWO) in order to establish the adaptive fuzzy membership functions. We then compare the effectiveness of the proposed method with the identified experimental model and the known methods, like SMC, Fuzzy SMC (FSMC), and GWO-SMC. Finally, the comparison with the known methods establishes the effectiveness of the proposed SAGWO-FSMC technique.

Keywords: PUMA 560 robotic arm, robotic manipulator, joint angles, fuzzy model, SAGWO

1. Introduction

In recent times, robots have been playing the fundamental role in the advancement of the industrial automation. Regarding the development of effective adaptive controllers, it is one of the truly challenging tasks to manage the robotic manipulators. Thus, the control engineers consider as highly complex handling the robotic manipulators in the conditions of fast operation in response to the disturbances from the external environment (see Lin and Lian, 2011; Rongcheng, Wenzhan and Sen, 2008; Lian, 2012; Vinay et al., 2016). The robotic manipulator can be considered as an example of a complex system, which acquires its complexity from the unmodelled dynamics and the properties of friction, pliability and the impacts of cross coupling, involving also the uncertainties and nonlinearities. Since most of the industrial automation systems gradually increasingly depend on the robotic manipulators, the sophisticated tasks, like welding and painting, are expected to produce precise trajectory tracking (Whitcomb et al., 1997), performed by the robust robotic system. Therefore, the recent stream of research often involves the task of motion control of robotic manipulators.

Under such circumstances, it is essential to develop the innovative control schemes, which should be appropriately insensitive with respect to the uncertainties of the parameters. These control schemes should be able to achieve exact trajectory tracking over the extensive ranges of motions with highly differentiated payloads. In the domain of robotics, the developments, concerning motion control, have been realised by the use of diverse schemes, like feedback linearization (Lian, 2012; Vijay and Jena, 2016), decentralized control (Lian, 2013; Mondal and Mahanta, 2014), model predictive control (Lian, 2014), adaptive control (Ekemezie and Osuagwu, 2001; Lin and Lian, 2011), and sliding mode control (Wong and Chen, 1998; Rongcheng, Wenzhan and Sen, 2008; Li, Ling and Chen, 2015). Due to its dimension-wise stability and environmental resistance, the sliding mode control (SMC) (Vascak and Madarasz, 2005; Kuo, Huang and Hong, 2011) gained extensive acceptance and drew a lot of attention from the research community. Basically, SMC is considered to provide the effective controlling scheme, as it ensures adequate performance also in the presence of external disturbances and uncertainties of the system (Salas, Llama and Santibanez, 2013; Shafiei, 2010; Piltan and Sulaiman, 2012). Furthermore, the state variables of the system are set to evenness using the rule of discontinuous feedback control, by the SMC technique. Thus, the uncertain nonlinear system is appropriately controlled by the SMC control system, as it possesses the valuable invariance properties. Even more, the SMC-based systems ensure better control performance for the multi-input multi-output (MIMO), nonlinear and discrete time signals.

Currently, diverse applications, like process control, robotics (Li, Ling and Chen, 2015; Vascak and Madarasz, 2005; Kuo, Huang and Hong, 2011; Robinson et al., 2016; Incremona et al., 2015; Corradini et al., 2012; Lakhekar and Roy, 2014), aerospace and power electronics (Han and Lee, 2011; Hu and Woo, 2006), do largely depend on the sliding mode controllers. Due to the existence of a linear category of switching surfaces, the asymptotic pattern of convergence is secured in the existing SMC. Nevertheless, some drawbacks have also arisen in connection with the conventional SMC technique. The primary limitation is the problem of chattering, which amounts to generation of high-frequency oscillations in the output of the controller, while the second drawback is the high sensitivity with respect to the external disturbances and the parameter uncertainties. In order to overcome certain pertinent limitations, various researchers have developed different methods, which were collectively classified into two categories, namely the estimated "uncertain methods" and the boundary layer saturation method.

Nowadays, SMC is applied in conjunction with the methodologies of artificial intelligence. These methodologies include neural networks (NNs), and fuzzy logic and neuro-fuzzy techniques, which are being associated with the SMC to promote enhanced performance. They can be applied in time-variant, nonlinear and uncertain types of plants (Lin and Chen, 2002; Leung, Zhou and Su, 1991; Islam and Liu, 2011). In addition, the chattering can be reduced by applying the fuzzy logic approach to the SMC, and the methods of SMC to the fuzzy logic controller (FLC) (Siradjuddin et al., 2014; Iyapparaja and Sureshkumar, 2012), which, in turn, enhances also the stability of the system (Efe, 2008; Singh, 1985; Visioli and Legnani, 2002; Hsu, Chen and Li, 2001; Whitcomb et al., 1997; Kim and Gibson, 1991). Besides, some authors have noted the robust trajectory tracking problem arising in the presence of external disturbances and uncertainties in the robotic manipulators. The effective trajectory tracking of the robotic manipulator was attained by the neural network based sliding mode adaptive control (NNSMAC), that is – the linkage of adaptive techniques, an approximation by the neural network and the SMC technique.

This paper contributes the SAGWO-FSMC methodology to promote the optimum tuning of the joint angle in a robotic manipulator like PUMA 560 robotic arm. The methodology is used for this purpose by proceeding through two stages: (a) extraction of precise characteristics of the system model using fuzzy inference system, and (b) adaptive modification of the acquired characteristics of the system, using SAGWO algorithm, which thus controls the overall performance of the manipulator. The second section of the paper describes the related studies, along with the problem definition, while the third section of the paper presents the system model of PUMA 360 robotic arm. Further, the adaptive fuzzy based SMC controlling scheme is described in the fourth section. Then, the fifth section of the paper is devoted to the comparative analysis of the proposed SAGWO-FSMC with the desired experimental model and conventional SMC, FSMC and GWO-SMC models regarding the joint angles, the displacement and the mean displacement error. The final section provides the summarising remarks on the proposed method.

2. Literature review

2.1. Related work

In 2013, Ruey-Jing Lian (2013) developed an enhanced adaptive grey-prediction self-organizing fuzzy sliding-mode controller (EAGSFSC) to assure the positive steadiness of the robotic systems. In the implemented grey-prediction self-organizing fuzzy controller (GPSOFC), the proposed method was used to get rid of the entire problem, related to stability. In addition, the approach proposed supported the self-organizing fuzzy controller (SOFC), in the removing the issues, related to the parameter values, resulting from the choice of unstable parameters, while the method proposed also improved the impact of dynamic coupling among the degrees of freedom (DOF), here referring to the number of joints in a robotic arm in a dynamic system. Moreover, the same method made it possible to achieve several benefits, such as the reduction of tracking errors, related to the joint-space trajectory planning and point-to-point control of the robot. Furthermore, this method assisted the rhombus-path control of the robot with reduced tracking errors of the joint-space trajectory. Next, it reduced the variation of control command to enhance the duration of the service life of the system. The implemented GPSOFC method reduces the computational effort of the SOFC technique by bringing the number of learning cycles from four to two. Thus, the effective performance of the proposed EAGSFSC has been validated through the valuable experimental results, this performance being obviously better than that of the controllers like GPSOFC and SOFC.

Then, Sanjoy Mondal and Chitralekha Mahanta (2014) controlled robotic manipulators using the controller called adaptive second order terminal sliding mode (SOTSM) controller. Instead of the normal control input, its time derivative was used in the SOTSM controller. In the derivative based control, the discontinuous sign function appeared. In order to obtain the continuous control, the integration process was performed, and so the control becomes chatterless. Moreover, for dealing with uncertainties an adaptive tuning method was utilized. Since he proposed SOTSM technique removed the problem of inappropriate chattering of the controller, the successful tracking performance could be achieved.

Yet before, in 2012, Ruey-Jing Lian (2012) presented the enhanced self-organizing fuzzy sliding-mode controller (EASFSC) meant to overcome the problem of stability in the robotic system, this controller having originated from SOFC. In this case, the proper selection of the membership function and the extraction of appropriate fuzzy rules, adapted for the Fuzzy Logic Controller (FLC) design, were not the sole characteristics of the proposed method. The method, namely, enhanced the performance, related to the control of the robotic system, by assuring the balance associated with the overall operation of the system. In the respective experiment, an effective algorithm, based on Lyapunov stability theorem was adopted to improve the stability of the system using the proposed method. In addition, the control performance of the proposed method was validated by applying it to the 2-link robot. The most advantageous part of the proposed method was the reduction of the total count of learning cycles, as demonstrated by the simulation results. Consequently, the life expectancy of the actuator was also increased by the proposed EASFSC method. Moreover, the EASFSC was highly effective in diminishing both RMS and maximum errors, during the trajectory tracking for the purpose of robotic manipulator control.

Vijay and Jena (2016) suggested a hybrid system, which was a combination of the SMC with artificial neuro-fuzzy inference system (ANFIS), for use in the robotic manipulator. They have contributed, thereby, a new system for the robotic manipulator with the control strategy of two DOFs. First, a controlling technique of the robotic manipulator was accomplished by incorporating the proportional integral derivative (PID) sliding surface to the SMC. Further, a meta-heuristic algorithm of Particle Swarm Optimization (PSO) was used to optimize the parameters of the controller. Moreover, the same procedure was performed for other SMC techniques, including the boundary sliding mode control (BSMC), and the boundary sliding mode control with PID sliding surface (PIDBSMC). However, the conventional method used was not convenient for practical purposes, and the PSO tuning was used to identify the proper parameters of PIDBSMC. Apart from this, the proposed ANFIS based PIDBSMC method turned out to be suitable for experimenting with a real-time system, since it could adaptively alter the parameters of SMC, which further enhanced the performance of the system, even in the presence of diverse disturbances at the input.

The parameters of the SOFC are managed by the addition of the radial-basis-function neural-network (RBFN) to the self-organizing fuzzy radial-basis-function neural network controller (SFRBNC) in the real time systems. However, the complication in determining the stability of the system existed as the challenging problem for the systems in question. Accordingly, Ruey-Jing Lian (2014) implemented the robotic system with self-organizing fuzzy sliding-mode radial-basis-function neural-network controller (ASFSRBNC). Application of the ASFSRBNC ensured the accurate attainment of stability of the system, meaning that in this manner the problems of SFRBNC have been resolved. Further, the control performance of the FLC was enhanced by employing the adaptive law that involved optimization of the parameters of the fuzzy system for controlling the robotic system. Ultimately, the Lyapunov stability theorem was employed in order to improve the stability of the ASFSRBNC method.

2.2. Assessment

Over the previous half-century, diverse sets of robotic manipulators have been designed (Lian, 2013). The development of the robotic systems with effective controllers has been growing tremendously. The controllers applied included the techniques of SMC, FLC, Neural Networks, feedback control based on PD output, finite-time control etc. (see, e.g., Mondal and Mahanta, 2014; or Wong and Chen, 1998), and were meant to bring the robotic systems to perfection. This is insofar a challenging task as, in general, the nonlinearities may occur in the dynamic model robotic manipulators. Furthermore, the dynamic models may also suffer from uncertainties, related to stability, friction, as well as alteration of the load. Those issues highly affect the overall performance of the manipulators, when a simple control algorithm regulates the imprecise model of the plant. Under such circumstances, SMC has attained much of attention, as it both provides effective performance and improved stability. In fact, SMC is less sensitive with regard to various external disturbances and the deviations of the poorly known and undefined parameters (Rongcheng, Wenzhan and Sen, 2008; Li, Ling and Chen, 2015; Vascak and Madarasz, 2005; Kuo, Huang and Hong, 2011). In this context, though, generation of chattering has to be considered as another challenge inherent to the nature of the respective control law. This kind of challenge can be removed by the boundary-layer method.

Recently, diverse self-tuning mechanisms have been proposed to enhance the performance of the controllers. Among the different methods, SOFC was considered to be a successful method, constituting a realistic approach to robot control in terms of varying the parameters of the PS controllers and of other existing controllers (see Mondal and Mahanta, 2014; Lian, 2012; or Vijay and Jena, 2016), other applications, including naval vehicles (Wong and Chen, 1998), process control (Lian, 2014; Ekemezie and Osuagwu, 2001; Lin and Lian, 2011), as well as other kinds of processes (Rongcheng, Wenzhan and Sen, 2008; Li, Ling and Chen, 2015; Vascak and Madarasz, 2005).

First of all, the SOFC (Lian, 2012) is meant to reduce the usage of human experts in implementing the FLC. However, the design parameters should be chosen prudently for the application of this method, since the imprecise selection may lead to the instability of the system. Even more, the GPSOFC technique (Lian, 2013) provides complete robustness and self-adaptive properties to the system, yet the system is highly sensitive to the initial value, which results in generation of wider fluctuations in the output. Then, high precision and robust characteristics are the beneficial features of the TSM controller (see Modal and Mahanta, 2014). However, the convergence of this method is slow, and it requires complex manipulations to control the speed of the joints. Further, the PSO, used by Vijay and Jena (2016) is easy to implement, which leads to high efficiency, but it may in some cases get trapped in the local area and thus bring out a lower convergence rate. Finally, RBFN (Lian, 2014) provides for a faster convergence rate and a high level of tolerance, also with respect to the noisy inputs.

3. Modelling of PUMA 560 robot

3.1. Robotic variables and the coordinate system

Figure 1 shows the structure of the PUMA 560 robot, which comprises the arrangement of six revolute joints. The middle line of the trunk line L_1 concurs with the axis of the joint 1. Accordingly, the measurement regarding the angle of the joint 1, θ_1, begins from the positive y-axis, that is – in the counter clockwise direction. Regarding the joint 2, the respective axis is assigned to begin from the positive x-axis, move perpendicularly to it, and converge with the axis of joint 1. In addition, it concurs with the middle line of the shoulder. In general, the shoulder is considered to be an offset, with length b_1. The measurement of the particular lengths is made between the upper arm and the trunk. When θ_1 is equal to zero, the respective offset is parallel to the $x-y$ plane and is in the direction of the negative x-axis. Around the joint 2, the upper arm and link L_2 get revolved, at an angle of θ_2. As per Fig. 2, when the link L_2 is parallel to the z-axis, the angle θ_2 is equal to zero.

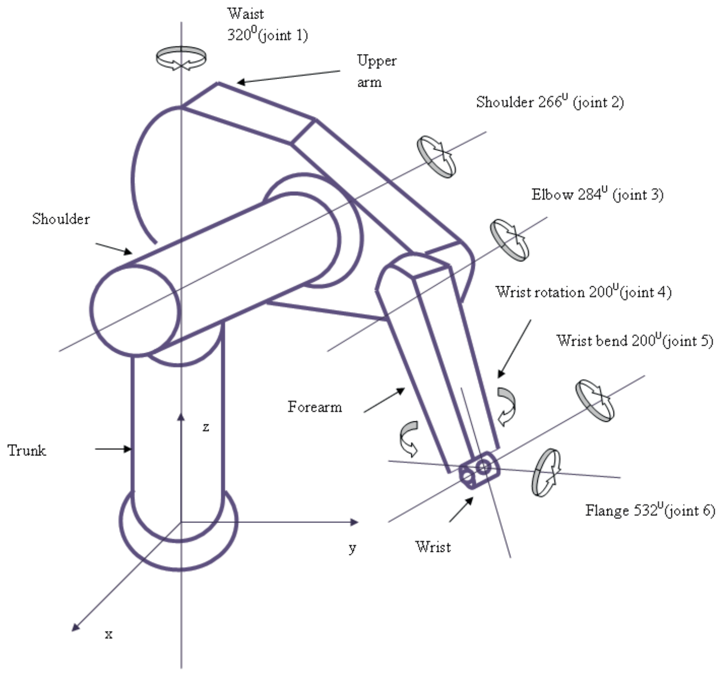

Figure 1. The structure of PUMA 560 robot arm

Furthermore, the axis of joint 3 and the elbow are placed in parallel to the axis of joint 2. As shown in Fig. 3, the two-part link is formed by the link L_3 and the forearm, termed m and n, respectively. The vector sum of m and n defines the link L_3. This vector sum corresponds to the distance between the axis of joint 3 and the middle of the spherical wrist. The distance b_2, parallel to the $x - y$ plane, provides the offset of the link L_3 from the link L_2. Moreover, b_2 is situated along or parallel to the positive x-direction, when the angle θ_1 is equal to zero. As per Fig. 2, L_2 is parallel to the z-axis and L_3 forms the angle δ with the vertical direction, this angle being considered to be known, when pointing up the arm at the reference position, explained below. The resultant angle is a function, associated with the dimension of the arm. Therefore, the reference position is as defined in Eq. (1), where $\delta = \sin(L_0/L_3)$, and L_3 indicates the offset from the middle line of the two parts, making the forearm:

$$\theta_{3r} = \delta. \tag{1}$$

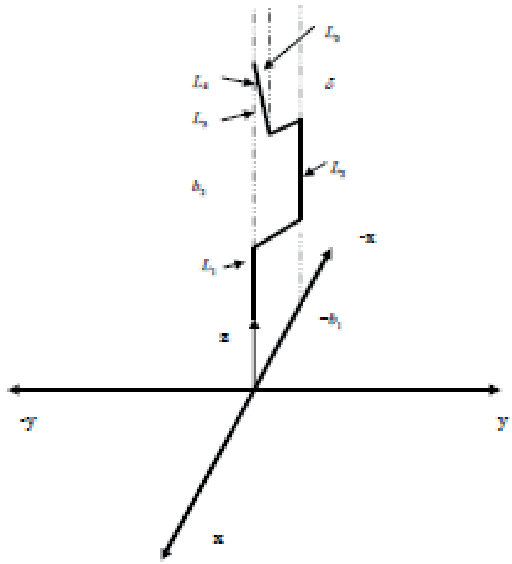

Figure 2. Initial state of the PUMA 360 robot arm

Now, the spherical wrist is formed by the joints 4, 5 and 6. Here, the axis of the joint 4 is perpendicular to and converges with the axis of the joint 5. The middle of the wrist to the flange forms the link L_4. In addition, around the axis of joint 4, its angle θ_4 gets rotated. On the other hand, the axis of the joint 5 is parallel to the axes of joints 2 and 3. The link L_4 forms the angle of rotation θ_5. The measurement of the particular angle is on the z-axis in the coordinate of L_4. This results in the rotation of the base coordinates through $\theta_1 k$ (first), -$(\theta_2 + \theta_3)i$ (second), and $\theta_4 k$ (final). Furthermore, the axis of the joint 6 is perpendicular to and crosses the axis of joint 5. Besides, it coincides with the middle line of the gripper mounting flange.

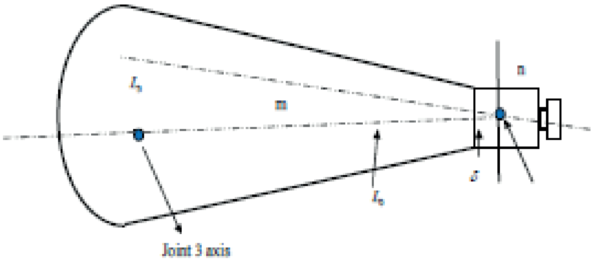

Figure 3. Demonstration of link L_3

Conform to this specification, Fig. 4 shows the link representation regarding the arm of the robot at any random positions. The joint coordinates ($\theta = \theta_1$, θ_2, θ_3, θ_4, θ_5, θ_6), and the Cartesian coordinates $R = (r_x,\ r_y,\ r_z,\ r_\rho,\ r_\theta,\ r_\psi)^T$ represent the end effector positions, with T denoting the transpose. In this representation, the position vector is constituted by $r = (r_x,\ r_y,\ r_z)^T$, and the rotations of the z-axis, the new x-axis, and new z-axis are denoted as, respectively, r_ρ, r_θ, r_ψ. Thus, a simplified solution is obtained by selecting the parameters of rotation, by the joint arrangements at the wrist.

3.2. Robotic model

The representation of dynamics of a serial n-link robot is given in Eq. (2), where u denotes the joint displacements, represented by an n x 1 vector, $\dot{u}$ denotes the joint velocities as an $nx1$ vector, τ denotes the torque of the actuators, again as an n x 1 vector, $M(u)$ denotes the symmetric positive definite n x n inertia matrix, $c(u, \dot{u})$ denotes the torques of centripetal and Coriolis forces as an n x 1 vector and $g(u)$ denotes the torque of the gravitation, being also an n x 1 vector. Moreover, the value of $g(u)$ is calculated as the gradient of the potential energy $U(u)$, due to gravity:

$$M(u)\ddot{u} + c(u, \dot{u}) + g(u) = \tau. \tag{2}$$

Let us consider that the joints of the robot are linked together with the revolute joints. Here, u_d represents the required joint positions. It is assumed to be a double differentiable vector function. To attain the control aim, Eq. (3), below, needs to be satisfied, which, in turn, provides the estimation of the torque of the actuator.

$$\lim_{t\to\infty} u(t) = u_d(t). \tag{3}$$

The current simulation considers the DOF PUMA-560 robot, with the arrangement of six joints. In addition, based on Armstrong, Khatib and Burdick (1986),

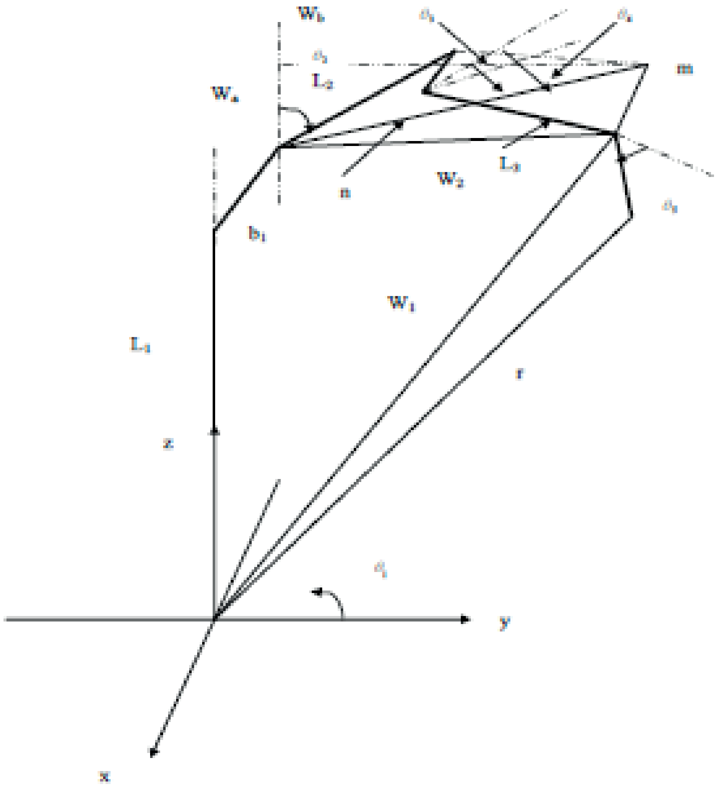

Figure 4. Link representation of PUMA 560 arm

the kinematical and dynamical properties of the arm are established. The motors of PUMA are the commercially available and applicable DC motors.

4. Adaptive fuzzy based SMC control scheme

4.1. The proposed control scheme

The architecture of the control scheme, based on the adaptive fuzzy system, is shown in Fig. 5. The proposed simulation model is developed to tune the joint angles of the PUMA 560 robot arm. Here, the actual feedback is generated from the real PUMA 560 system, which is connected to the equivalent control law generator. Further, the desired trajectory and the actual feedback are used to compute the error function (E) and the differential error function (DE). Then, the sliding surface generator generates the activating signal, based on the computed error function value. The sliding mode constants are adjusted by the proposed adaptive fuzzy system with a metaheuristic SAGWO algorithm, which results in the minimized joint angle error.

To the fuzzy system, two inputs, namely E and DE, are applied. As per the fuzzy values of these inputs, these values are referred to as Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), Negative Small (NS), Negative Medium (NM) and Negative Big (NB). The values of Z, PS, PM, NS and NM are represented by the triangular membership functions and the values of PB and NB are represented by the trapezoidal membership functions.

With the so defined input values, the fuzzy system generates the corresponding rules, considered to produce the sliding mode constants. In this manner, the generated sliding mode constants are completely based on the values of E and DE. Accordingly, Table 1 depicts the rules or the sliding mode constants generated by the fuzzy system.

Table 1. Rules or SMC constants generated by the fuzzy system

E/DE	*NB*	*NM*	*NS*	*Z*	*PS*	*PM*	*PB*
NB	NB	NB	NB	NB	NM	NS	Z
NM	NB	NB	NB	NM	NS	Z	PS
NS	NB	NB	NM	NS	Z	PS	PM
Z	NB	NM	NS	Z	PS	PM	PB
PS	NM	NS	Z	PS	PM	PB	PB
PM	NS	Z	PS	PM	PB	PB	PB
PB	Z	PS	PM	PB	PB	PB	PB

The shapes of the fuzzy membership functions used are shown in Fig. 6.

The expression for the triangular membership function is given in Eq. (4), in which r refers to the lower limit, s to the upper limit, t is some value between r and s, and x is the input variable, representing either error (E) or the differential error (DE). Likewise, the representation of the trapezoidal membership function

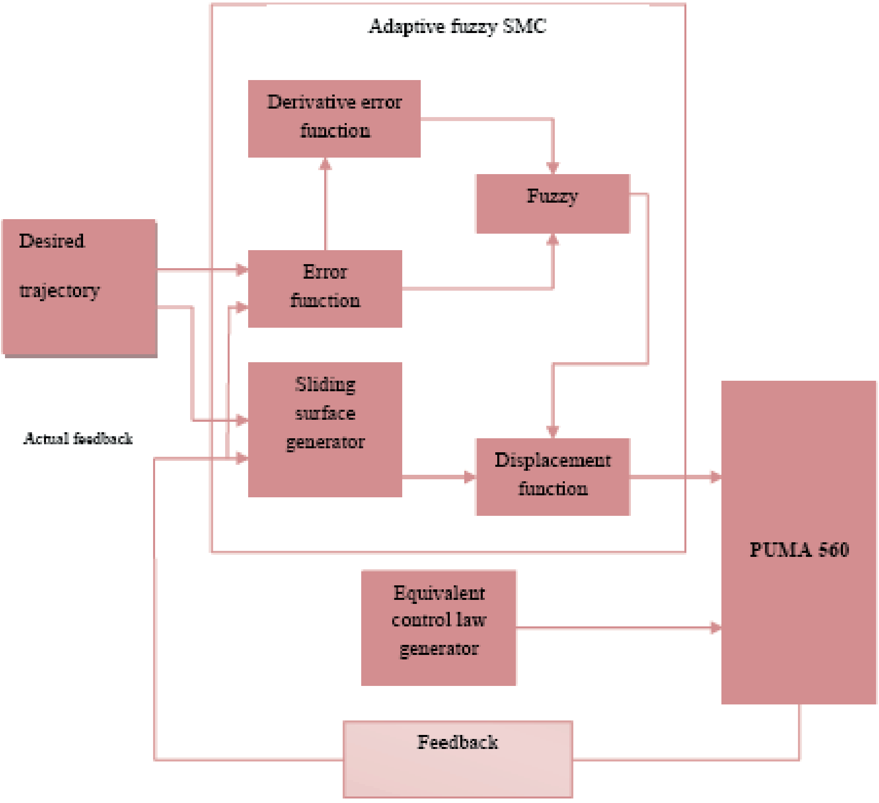

Figure 5. Architecture of the adaptive fuzzy based SMC control scheme

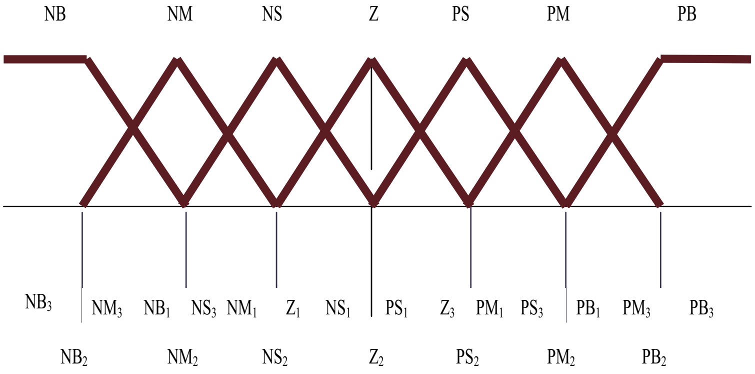

Figure 6. Shapes of the fuzzy membership functions used

is provided in Eq. (5), where u and v denote the lower and upper support limits at the membership of 1, and where $r < u < v < s$. The values of the parameters of the memberships function for SMC are shown in Table 2.

$$\mu_1(x) = \begin{cases} 0, \quad x \leq r \\ \frac{x-r}{t-r}, \quad r < x \leq t \\ \frac{s-x}{s-t}, \quad t < x < s \\ 0, \quad x \geq s \end{cases} \tag{4}$$

$$\mu_2(x) = \begin{cases} 0, \quad (x < r) \ or \ (x > s) \\ \frac{x-r}{u-r}, \quad r \leq x \leq u \\ 1, \quad u \leq x \leq v \\ \frac{s-x}{s-v}, \quad v \leq x \leq s \end{cases} \tag{5}$$

Table 2. Parameters of the membership functions used in SMC

x	r	t	s	u	v
NM	NB$_1$	(NB$_2$-NB$_1$)/2	NB$_2$	-	-
NS	NM$_1$	(NM$_2$-NM$_1$)/2	NM$_2$	-	-
PS	PM$_1$	(PM$_2$-PM$_1$)/2	PM$_2$	-	-
PM	PB$_1$	(PB$_2$-PB$_1$)/2	PB$_2$	-	-
NB	-	-	-	NB$_1$	NM$_2$
PB	_	_	-	PM$_2$	PB$_1$

4.2. Adaptive membership function

Basically, a membership function is defined as a *"curve that defines how each point in the input space is mapped to a membership value between 0 and 1"*. The output of a fuzzy system generates the particular membership function with the rules it extracts, see Alavandar and Nigam (2008). Therefore, the SAGWO algorithm is made use of here to adaptively alter the membership functions, which should minimize the error between the actual and the desired value.

The GWO algorithm (Mirjalili, Mirjalili and Lewis, 2014) is a recently introduced meta-heuristic algorithm, which operates on the basis of the principles standing behind the hunting behaviour of grey wolves, when they catch the prey. In our case, three wolves, designated as α, β, and δ, proceed with the hunting for the respective preys. The hunting pattern is based on three phases, which include (a) tracking, following, and catching the prey, (b) until the movement of the prey, continuing pursuing, surrounding and disturbing, and (c) attacking the prey. Among the three wolves, α is assigned the role of the leader of all wolves, taking the decisions, regarding sleeping, hunting and resting time of wolves. The second and third wolves, i.e. β and δ assist the leader in taking the necessary decisions. There is also a fourth wolf present, called ω, this wolf being only allowed to eat, and so it does not have importance during hunting. Figure 7 shows the solution encoding for adaptive membership functions, treated by the SAGWO algorithm. Here, each solution element is the position of each wolf, and the best solution has to be determined from a set of solutions using the proposed algorithm.

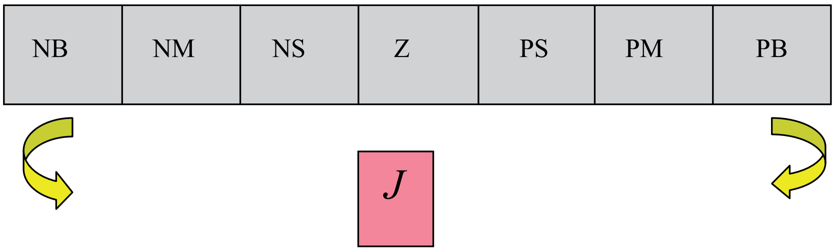

Figure 7. Solution encoding pattern of theSAGWO algorithm

Thus, the membership functions, after integrating the fuzzy system with the GWO algorithm, are collectively represented as in Eq. (6), where J indicates the solution vector to the SAGWO algorithm, as shown in Fig. 7.Eq. (6), namely, represents the update of the fuzzy membership functions, relative to the improvement, introduced through the solution found.

$$\mu = \mu(J). \tag{6}$$

The objective function of the proposed SAGWO based SMC in PUMA 560 robotic arm is expressed through Eq. (7), and it amounts to the error between the actual and the desired joint angles. In Eq. (7), θ_i^D denotes the desired joint angle and θ_i^A denotes the measured joint angle. The formulation for the desired joint angle is provided in Eq. (8), in which ε denotes the control signals from SMC and K^{fuzzy} denotes the output from the fuzzy system. Here, the desired joint angle is determined on the basis of the generated control signals from the SMC, which change. The computation of the control signal is done according to Eq. (9), in which ε_{eq} and ε_{sw} denote the system model and the continuous part of SMC, respectively, the latter magnitude being defined as in Eq. (10), followed by Eq. (11), with s_i denoting the switching boundary and f referring to the fuzzy representation of x_i (the membership function).

$$E|_{t=t^{\max}} = \sum_{i=1}^{3} \left|\theta_i^D(t) - \theta^A(t)\right| \tag{7}$$

$$\theta^D = \varepsilon\left(K^{fuzzy}\right) \tag{8}$$

$$\varepsilon = \varepsilon_{eq} + \varepsilon_{sw} \tag{9}$$

$$\varepsilon_{sw} = -K^{fuzzy} satf(x_i) \tag{10}$$

$$satf(x_i) = \begin{cases} +1, \ if \ \ f(x_i) > s_i \\ \frac{f_i}{s_i} \ if \ \ f(x_i) \leq s_i \\ -1, \ if \ \ f(x_i) < s_i \end{cases} . \tag{11}$$

Thus, the control signal from the SMC is generated on the basis of the fuzzy rules, as illustrated in Table 1. From there, the desired joint angles are computed. Next, there follows the adaptive adjustment of the determined joint angles by the SAGWO algorithm.

The formulations, associated with the encircling pattern of grey wolves are represented in Eqs. (12) through (15), where C and H specify the coefficient vectors specifying the control combinations, J denotes the position(s) of the grey wolves, J_p denotes the position vector of the prey and t is the current iteration number. The expressions for the determination of vectors C and H are provided in Eqs. (14) and (15), respectively, in which a denotes a parameter, which decreases from the value of 2 to 0, as explained later on, while r_1 and r_2 denote the random vectors, whose values are uniformly distributed between [0, 1].

$$K = |H \cdot J_p(t) - J(t)| \tag{12}$$

$$J(t+1) = J_p(t) - C \cdot K \tag{13}$$

$$C = 2a.r_1 - a \tag{14}$$

$$H = 2 \cdot r_2. \tag{15}$$

The SAGWO algorithm adapts the parameter a as in Eq. (16), where τ denotes the change in the fitness function, which, in turn, is given in Eq.(17). In this equation, $f(t\text{-}1)$ refers to the preceding iteration and $f(t)$ to the current one. In the first iteration, $\tau = 1$.

$$a = \left(2 - 2 \times \frac{1}{Maximum \quad iteration}\right) \times \tau \tag{16}$$

$$\tau = \frac{f(t-1) - f(t)}{f(t-1)}. \tag{17}$$

Finally, the hunting behaviour of the grey wolves is formulated in Eqs. (18) through (23), these equations determining the respective positions of each wolf. Ultimately, Eq. (24) gives the rule for updating the position, based on the positions of all the involved wolves.

$$K_\alpha = |C_1 \cdot J_\alpha - J| \tag{18}$$

$$K_\beta = |C_2 \cdot J_\beta - J| \tag{19}$$

$$K_\delta = |C_3 \cdot J_\delta - J| \tag{20}$$

$$J_1 = J_\alpha - C_1 \cdot (K_\alpha) \tag{21}$$

$$J_2 = J_\beta - C_2 \cdot (K_\beta) \tag{22}$$

$$J_3 = J_\delta - C_3 \cdot (K_\delta) \tag{23}$$

$$J(t+1) = \frac{J_1 + J_2 + J_3}{3}. \tag{24}$$

The pseudocode of the proposed SAGWO based SMC for the PUMA 560 robotic arm is provided in the form of Algorithm 1. The input of the SAGWO algorithm consists in establishing the bounds of the triangular membership functions (the maximum and minimum limits), corresponding to E and *DE*. In the SAGWO algorithm, the position of the α wolf is assigned to represent the best solution.

A more verbal description of functioning of Algorithm 1 is provided below:

```
ALGORITHM 1: PSEUDOCODE OF THE PROPOSED SAGWO BASED SMC
Input: Bounds of membership functions and population size
Output: $J_a$
Initialize the population of grey wolves $J_n, n = 1, 2, \ldots, n_{max}$
Initialize the vectors $C$ and $H$
Assign $J_\alpha, J_\beta$ and $J_\delta$, as three best search agents
While ($t < Maximum\ \ iteration$)
    Set $\tau = 1$ and assign $a$ as in Eq. (16)
    For all positions of wolves
        Update each best position as per Eq. (24)
    End for
    Update $a$, $C$ and $H$
    Compute the fitness of all positions of wolves
    $t = t + 1$
    Set $\tau$ as per Eq. (17) and assign $a$
End while
Return $J_a$
```

1. At first, the population of the grey wolves is initialized as J_n, where $n = 1, 2, \ldots, n_{max}$.
2. Then, the vectors C and H are also initialized.
3. Subsequently, the three best search agents are considered as J_α, J_β and J_δ.
4. It is also necessary to initialize the value of a as per Eq. (16).

5. At the first iteration, the value of τ is equal to one, and following the first iteration, the value of τ is assigned as per Eq. (17).
6. Then, as per Eq. (24), the position of the entire set of wolves or the best search agents is updated.
7. Accordingly, the update of the vectors C and H and of the parameter a is performed.
8. Then, the fitness of all the positions of wolves is calculated.
9. The same process is repeated for consecutive iterations until the best search agent is obtained.

5. Simulation results

5.1. The procedure

The basic Simulink model of the SAGWO-FSMC is shown in Fig. 8, with the SAGWO block being broadly modelled as in Fig. 9. The respective experiment is simulated in MATLAB. During the optimization procedure, the number of iterations required to realise the SAGWO-FSMC is assigned as 100. The required parameter values are set as mentioned in the algorithm. Following the here described experiment implementation, its performance is compared with those of the conventional techniques, such as SMC, FSMC, and GWO-SMC, in order to validate the effectiveness of the new approach.

5.2. Results and discussion

We now turn to the discussion of results obtained from the experiments, concerning the proposed robotic controller. Thus, Fig. 10 shows the convergence analysis. As mentioned earlier, the objective function or the cost function of this experiment is the error between the actual and the desired value, which is to be minimised. The analysis compares the converging performance between the conventional GWO and the proposed SAGWO based FSMC approach. Here, the conventional GWO-FSMC starts from the value of 8×10^{-3} and ends at 4.4×10^{-3}. On the other hand, the proposed SAGWO-FSMC, which also starts from 8×10^{-3}, ends at 4.2×10^{-3}, i.e. performs better than the GWO-FSMC.

Further, the performance analysis in terms of three joint angles, namely θ_1, θ_2 and θ_3 has been conducted and is illustrated in Fig. 11.

The analysis was limited to 10 ms and the movement of the joint angles subject to control was observed. The comparison of the actual joint angle θ_1 with the desired value of it, as provided in Fig. 11(a), shows that the performance of SAGWO-FSMC in terms of the error is 1%, of SMC – 3%, of FSMC and GWO-FSMC – 1.9%. Now, regarding a similar analysis with respect to the joint angle θ_2, which is shown in Fig. 11(b), the measured θ_2 of SAGWO-FSMC is 4.41% away from the actual value, which is again better than for the other models. Finally, the measured θ_3 of SAGWO-FSMC is 6.15% away from the actual value, as shown in Fig. 11(c). Therefore, the proposed SAGWO-FSMC controls the

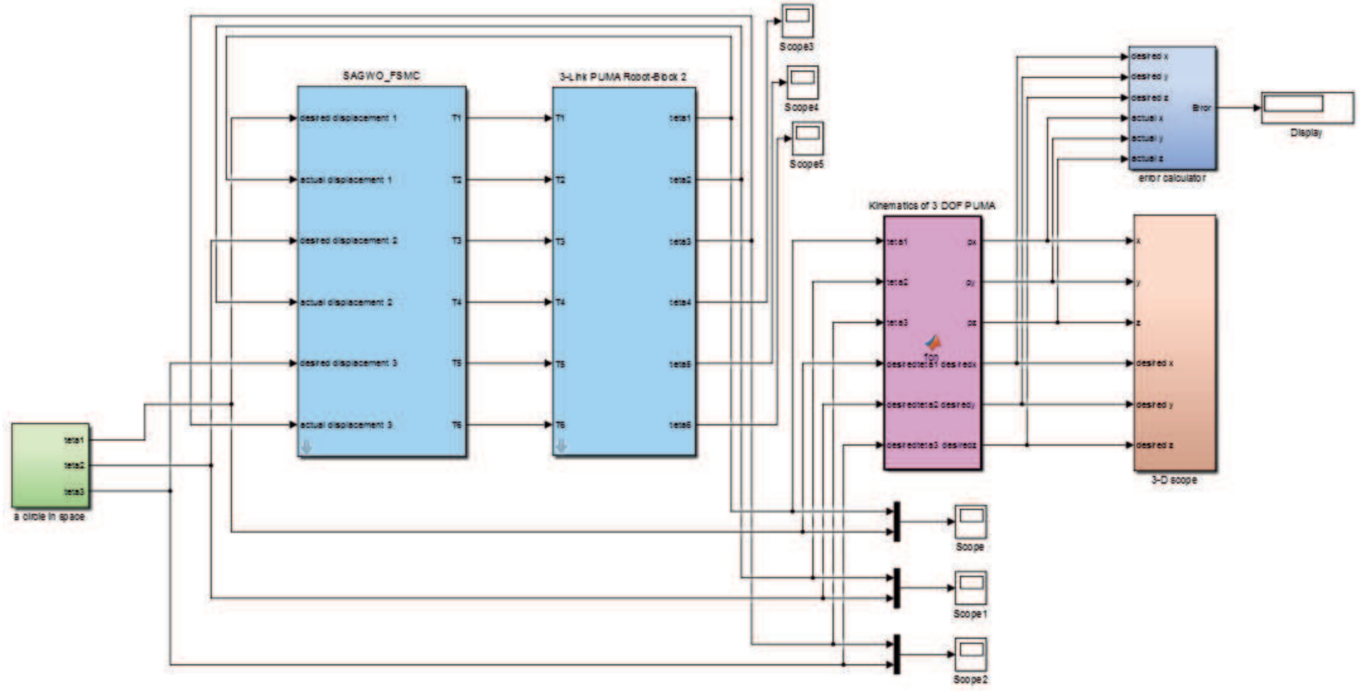

Figure 8. Simulink model of SAGWO-FSMC

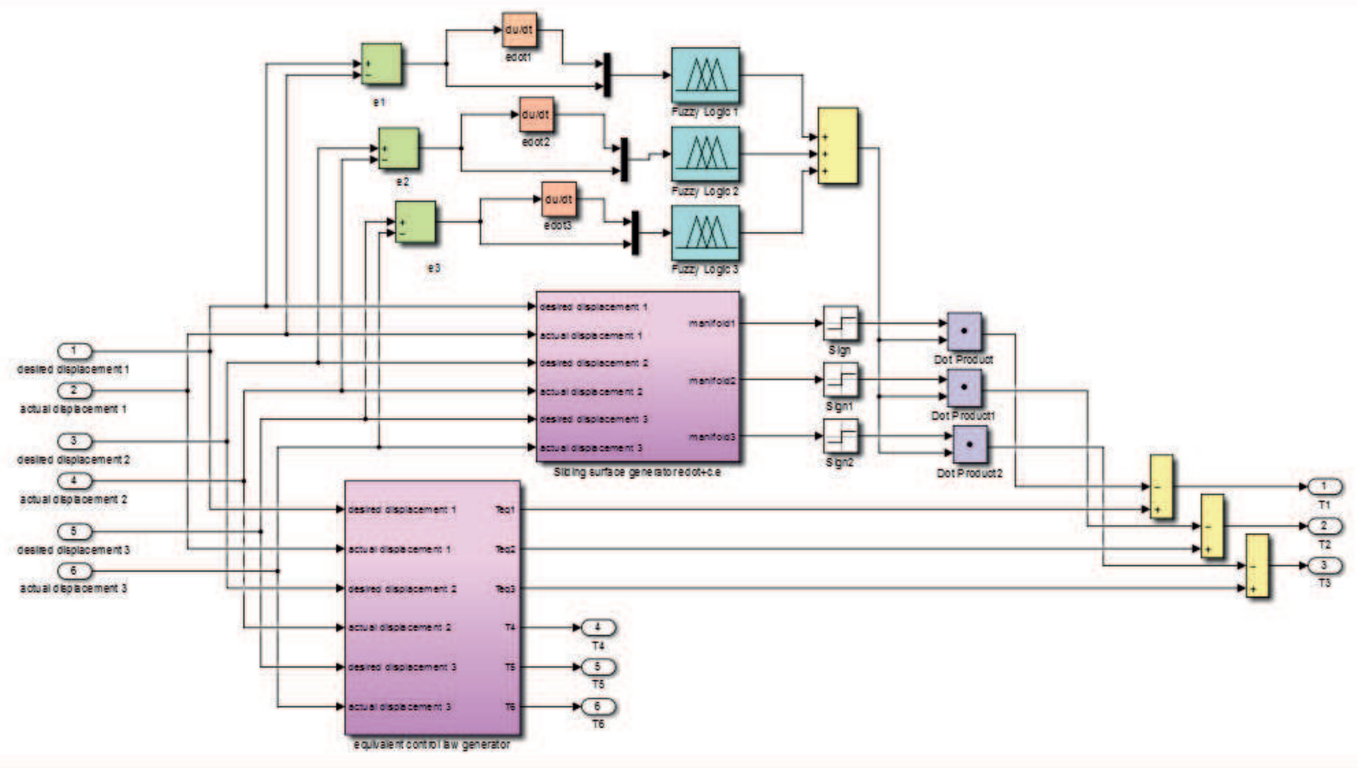

Figure 9. Simulink model of the SAGWO block

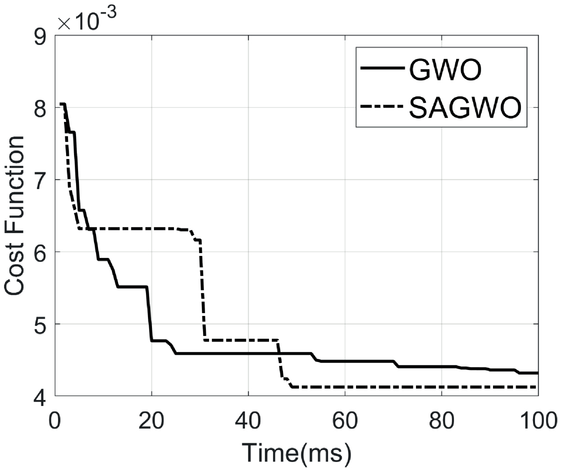

Figure 10. Convergence analysis

joint angles, concerning movement precision, in the way showing superiority over the conventional SMC methods.

The graphical representation of the analysis concerning three displacements, namely x, y, and z, over time is shown in Fig. 12. The performance with respect to each of displacement dimensions by the proposed SAGWO-FSMC is compared with the desired experimental model and the results for the conventional techniques, such as SMC, FSMC, and GWO-FSMC. The measured displacement should be possibly close to the displacement of the desired model. In the case of displacement x, as shown in Fig. 12(a), the error-wise performance of SMC is 9.52% and of FSMC and SAGWO-FSMC it is 2.56%. Thus, the SMC technique produces movement far away from the actual one, whereas the other methods follow the desired movement much more closely in terms of x. Then, the displacement y shows for SMC the deviation of 1%, for FSMC and GWO-FSMC and for SAGWO-FSMC - only 0.04%, as illustrated in Fig. 12(b). Finally, as shown in Fig. 12(c), the displacement z, produced by SAGWO-FSMC is by 14.49% away from the desired displacement, meaning better performance than for other compared techniques.

Fig. 13 shows the graphical representation of changes in mean displacement error with time. As per the graphical representation, it can be seen that the mean displacement error of the SMC is quite high, being equal 0.05, while that of GWO-FSMC is at 0.01. However, the error is reduced, virtually to 0 by the proposed SAGWO-FSMC. Thus, it is proved that the error is perfectly [?????] reduced by the proposed SAGWO-FSMC method.

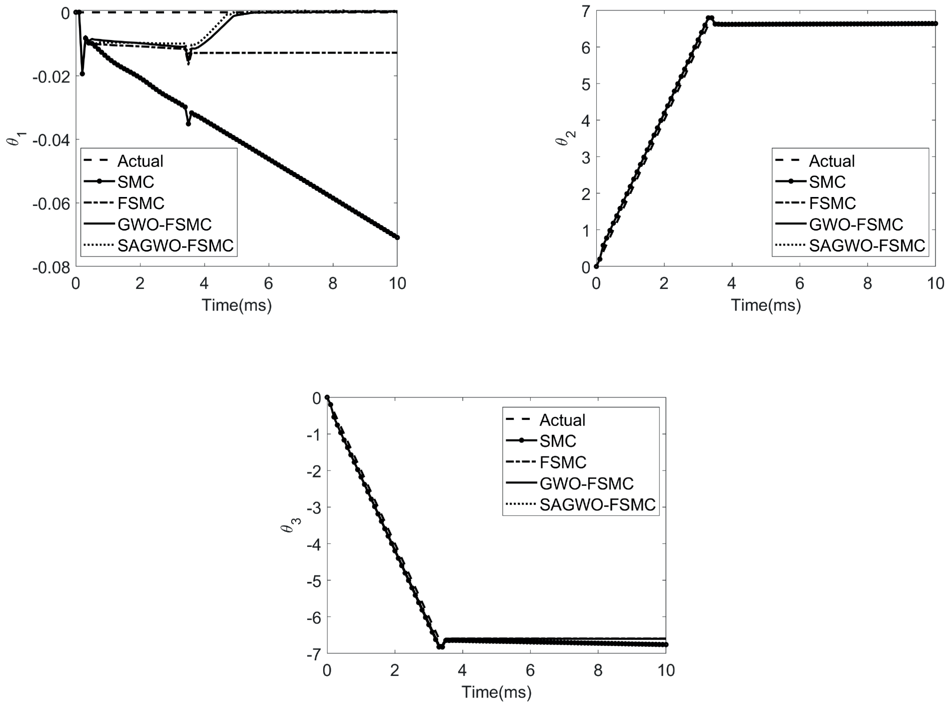

Figure 11. Performance with respect to three angles of joints: (a) θ_1, (b) θ_2 and (c) θ_3 in time

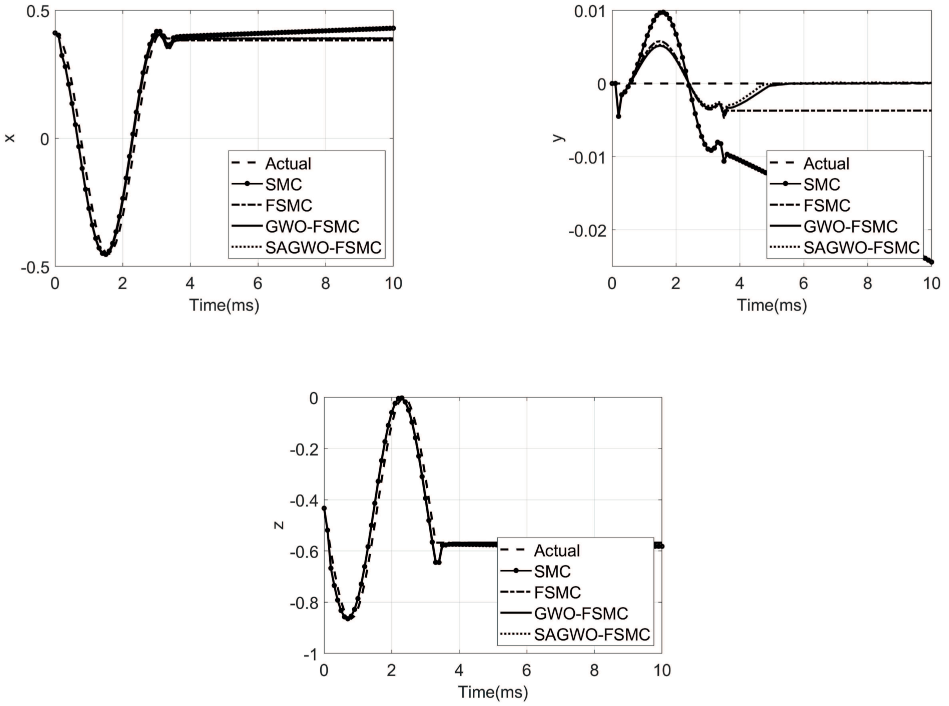

Figure 12. Performance regarding three displacements, namely (a) X (b) Y and (c) Z over time

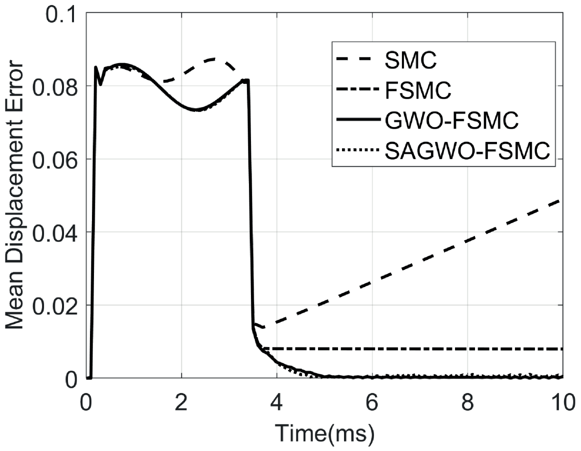

Figure 13. Representation of mean displacement error with time

6. Error analysis

Table 3. presents error analysis of the proposed SAGWO-FSMC in comparison with other considered methods, like SMC, FSMC, and GWO-FSMC. The error values, provided in this table, imply that indeed, the proposed SAGWO-FSMC algorithm performs well compared to other techniques.

6.1. Optimized plots

In this research work, an adaptive SMC controller for robotic manipulator has been developed. Further, in this context, fuzzy rules have been generated with two inputs, Error (E) and Differential Error (DE). These rules can serve for an easy determination of the SMC constants, which can further be used to control the system properly. The method uses the optimized membership functions for the fuzzy part of the system. These optimized triangular membership plots for two inputs, and for output are shown in Figs. 14, 15 and 16, respectively.

7. Conclusions

This paper presented the development of an effectively performing controller for the robotic manipulator like PUMA 560 robotic arm. An adaptive SMC was implemented for this robotic manipulator. The general difficulty consisted in that a system model could not be ensured to match the operation of SMC at every time instant. Hence, fuzzy inference system was employed to replace

Table 3. Error analysis of proposed SAGWO-FSMC compared to other techniques. Error values are computed according to Eq. (7)

Time (ms)	SMC	FSMC	GWO-FSMC	SAGWO-FSMC
1	0.083907	0.08509	0.085151	0.084797
2	0.082915	0.074572	0.074599	0.074599
3	0.085664	0.078341	0.07845	0.078137
4	0.015349	0.0080468	0.0043225	0.0041288
5	0.020738	0.0080224	0.00036686	0.0005326
6	0.02631	0.0080041	0.0001145	0.00003908
7	0.031954	0.0079878	0.00015332	0.00046102
8	0.037628	0.0079734	0.0002266	0.00039751
9	0.043308	0.0079607	0.00034445	0.00043008
10	0.048986	0.0079494	0.00042725	0.00036425
Mean	0.047676	0.029395	0.024416	0.024389

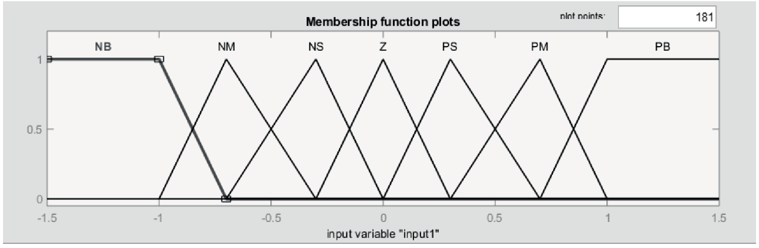

Figure 14. Triangular membership functions (after optimization) plot for input 1 (Error)

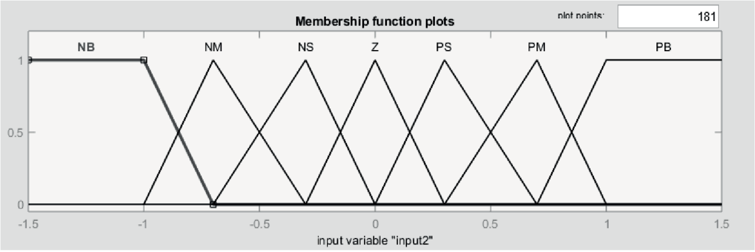

Figure 15. Triangular membership functions (after optimization) plot for input 2 (Differential Error)

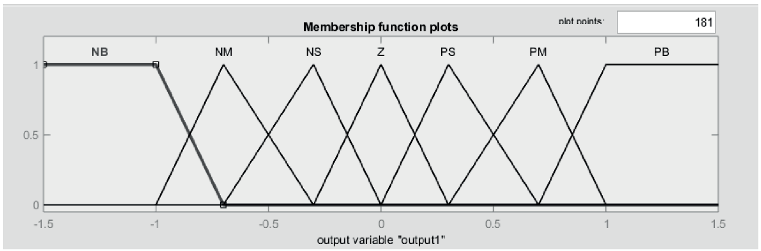

Figure 16. Triangular membership functions plot for output error

the system model. The adaptive fuzzy membership function was used to represent the derived fuzzy rules in the second stage, using the SAGWO algorithm. Next, the performance of the thus elaborated SAGWO-FSMC algorithm was compared with the desired behaviour and the methods like SMC, Fuzzy SMC (FSMC) and GWO-SMC. The experimental analysis demonstrated the superior performance of SAGWO-FSMC in tuning the optimum joint angles of the robotic manipulator.

References

Alavandar, S. and Nigam, M.J. (2008) Fuzzy PD+I control of a six DOF robot manipulator. *Industrial Robot: An International Journal,* **35**(2): 125–132.

Armstrong, B., Khatib, O. and Burdick, J. (1986) The explicit dynamic model and inertial parameters of the PUMA 560 arm. *Proceedings. 1986 IEEE International Conference on Robotics and Automation,* 510-518.

Corradini, M.L., Fossi, V., Giantomassi, A., Ippoliti, G., Longhi, S. and Orlando, G. (2012) Minimal Resource Allocating Networks for Discrete Time Sliding Mode Control of Robotic Manipulators. *IEEE Transactions on Industrial Informatics,* **8**(4): 733-745.

Efe, M.O. (2008) Fractional Fuzzy Adaptive Sliding-Mode Control of a 2-DOF Direct-Drive Robot Arm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* **38**(6): 1561-1570.

Ekemezie, P.N., Osuagwu, C.C. (2001) A Self-Organising Fuzzy Logic Controller. *Nigerian Journal of Technology,* 20(1).

Han, S. I. and Lee, J.M. (2011) Friction and uncertainty compensation of robot manipulator using optimal recurrent cerebellar model articulation controller and elasto-plastic friction observer. *IET Control Theory & Applications,* **5**(18): 2120-2141.

Hsu, Y., Chen, G. and Li, H. (2001) A fuzzy adaptive variable structure controller with applications to robot manipulators. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* **31**(3): 331-340.

Hu, H. and Woo, P. (2006) Fuzzy supervisory sliding-mode and neural-network control for robotic manipulators. *IEEE Transactions on Indus-*

trial Electronics, **53**(3): 929-940.

Incremona, G.P., Felici, G.D., Ferrara, A. and Bassi, E. (2015) A Supervisory Sliding Mode Control Approach for Cooperative Robotic System of Systems. *IEEE Systems Journal,* **9**(1): 263-272.

Islam, S. and Liu, X.P. (2011) Robust Sliding Mode Control for Robot Manipulators. *IEEE Transactions on Industrial Electronics,* **58**(6): 2444-2453.

Iyapparaja, M. and Sureshkumar, S. (2012) Coupling and cohesion metrics in Java for adaptive reusability risk reduction. *IET Digital Library,* 52-57.

Kim, Y.K. and Gibson, J.S. (1991) A variable-order adaptive controller for a manipulator with a sliding flexible link. *IEEE Transactions on Robotics and Automation,* **7**(6): 818-827.

Kuo, T.C., Huang, Y.J. and Hong, B.W. (2011) Design of Adaptive Sliding Mode Controller for Robotic Manipulators Tracking Control. *International Journal of Computer, Electrical, Automation, Control and Information Engineering,* **5**(5): 453-457.

Lakhekar, G.V. and Roy, R.G. (2014) Heading control of an underwater vehicle using dynamic fuzzy sliding mode controller. *2014 International Conference on Circuits, Power nad Computing Technologies (ICCPCT-2014), Nagercoil*, 1448-1454.

Leung, T., Zhou, Q. and Su, C. (1991) An adaptive variable structure model following control design for robot manipulators. *IEEE Transactions on Automatic Control,* **36**(3): 347-353.

Li, Y., Ling, L. and Chen, J. (2015) Combined grey prediction fuzzy control law with application to road tunnel ventilation system. *Journal of Applied Research and Technology,* **13**(2): 313-320.

Lian, R. (2012) Design of an enhanced adaptive self-organizing fuzzy sliding-mode controller for robotic systems. *Expert Systems with Applications,* **39**(1): 1545-1554.

Lian, R. (2013) Enhanced adaptive grey-prediction self-organizing fuzzy sliding-mode controller for robotic systems. *Information Sciences,* 236, 186-204.

Lian, R.J. (2014) Adaptive Self-Organizing Fuzzy Sliding-Mode Radial Basis-Function Neural-Network Controller for Robotic Systems. *IEEE Transactions on Industrial Electronics,* **61**(3): 1493-1503.

Lin, J. and Lian, R. (2011) Hybrid Self-Organizing Fuzzy and Radial Basis-Function Neural-Network Controller for Active Suspension Systems. *International Journal of Innovative Computing, Information and Control,* **7**(6).

Lin, W.S. and Chen, C.S. (2002) Robust adaptive sliding mode control using fuzzy modelling for a class of uncertain MIMO nonlinear systems. *IEEE Proceedings - Control Theory and Applications,* **149**(3): 193-201.

Mirjalili, S., Mirjalili, S.M. and Lewis, A. (2014) Grey Wolf Optimizer. *Advances in Engineering Software,* 69, 46–61.

Mondal, S., Mahanta, C. (2014) Adaptive second order terminal sliding mode controller for robotic manipulators. *Journal of the Franklin Institute,* **351**(4): 2356-2377.

Piltan, F. and Sulaiman, N.B. (2012) Review of Sliding Mode Control of Robotic Manipulator. *World Applied Sciences Journal,* **18**(12): 1855-1869.

Robinson, R.M., Kothera, C.S., Sanner, R.M. and Wereley, N.M. (2016) Nonlinear Control of Robotic Manipulators Driven by Pneumatic Artificial Muscles. *IEEE/ASME Transactions on Mechatronics*, **21**(1): 55-68.

Rongcheng, L., Wenzhan, V. and Sen, X. (2008) An adaptive grey fuzzy prediction controller design based on the improving residual error model. *2008 IEEE Conference on Cybernetics and Intelligent Systems, Chengdu,* 652-656.

Salas, F., Llama, M. and Santibanez, V. (2013) A Stable Self-Organizing Fuzzy PD Control for Robot Manipulators. *International Journal of Innovative Computing, Information and Control,* **9**(5): 2065-2086.

Shafiei, S.E. (2010) Sliding Mode Control of Robot Manipulators via Intelligent Approaches. *Advanced Strategies for Robot Manipulators,* 135-173.

Singh, S. (1985) Adaptive model following control of nonlinear robotic systems. *IEEE Transactions on Automatic Control,* **30**(11): 1099-1100.

Siradjuddin, I., Behera, L., Ginnity, T. M. M. and Coleman, S. (2014) Image-Based Visual Servoing of a 7-DOF Robot Manipulator Using an Adaptive Distributed Fuzzy PD Controller. *IEEE/ASME Transactions on Mechatronics,* **19**(2): 512-523.

Vascak, J. and Madarasz, L. (2005) Automatic Adaptation of Fuzzy Controllers. *Journal of Cybernetics and Informatics,* **2**(2): 5-18.

Vijay, M. and Jena, D. (2016) PSO based neuro fuzzy sliding mode control for a robot manipulator. *Journal of Electrical Systems and Information Technology*, **4**(1). 243–256.

Vinay, R., Naveen, S. B. K., Kumar, S. and Monica, B.V. (2016) Indoor Surveillance Robot with Live Video Transmission. *Computation and Communication Technologies,* 99–108.

Visioli, A. and Legnani, G. (2002) On the trajectory tracking control of industrial SCARA robot manipulators. *IEEE Transactions on Industrial Electronics,* **49**(1): 224-232.

Whitcomb, L. L., Arimoto, S., Naniwa, T. and Ozaki, F. (1997) Adaptive model-based hybrid control of geometrically constrained robot arms. *IEEE Transactions on Robotics and Automation,* **13**(1): 105-116.

Wong, C. and Chen, C. (1998) Design of Fuzzy Control Systems with a Switching Grey Prediction. *Fuzzy Systems Proceedings, 1998. IEEE World Congress on Computational Intelligence, The 1998 IEEE International Conference on, Anchorage, AK,* 1, 567-571.