CIT. Journal of Computing and Information Technology, Vol. 28, No. 1, 2020, 1-10
doi: 10.20532/cit.2020.10048872

Motion Planning for Robotic Guidance System of Dental Surgery using Particle Swarm Optimization

Ali Muhssen Abdul-Sadah[1], Kamal .M.H. Raheem[2], Nasseer K. Bachache[3],
[1] Computer Technical Engineering Dept., Technical Engineering College, University of Alkafeel, Najaf, Iraq
[2] Computer Technical Engineering Dept., Technical Engineering College, University of Alkafeel, Najaf, Iraq
[3] Computer Technical Engineering Dept., Technical Engineering College, University of Alkafeel, Najaf, Iraq
E-mail: ali.muhssen@alkafeel.edu.iq[1], Tech_n2008@yahoo.com[2], kamal.raheem@alkafeel.edu.iq[3]

Abstract: Dental Robotics is one of the promised treatment and therapy options in dental surgery nowadays. Recently a dental robotic guidance system was proposed, where a novel Particle Swarm optimization PSO is used to control the needle tip position and orientation as a function of the knots along the path between the end-effector segments. The new approach named “Numeric Alphabet Flow” NAF can direct the arbitrary point to point endodontic trajectories for a 3-link robotic guidance system. The new NAF method uses the means of flowing numbers and alphabets in the rule base that employ the particle swarm optimization to generate a new form named as NAF-PSO technique. In this work, we apply the new NAF-PSO to optimize the robot trajectory planning. A novel path planner is introduced in this paper which defines the proposed NAF-PSO objective function as to minimize space and travelling time without exceeding a maximum predefined torque, and by avoiding the collision with an obstacle.

***Keywords**: Medical Robot and system, Nonholonomic motion planning, Obstacle avoiding, Needle steering.*

Introduction

Dental Robotics could be one of the promised treatment and therapy options in dental surgery nowadays, we use robotic in surgery because of its high level of quality, precision and safety. Since their new advancement, high-technologies like robotics laser and dental laser, such technologies are currently employed in root canal therapy and photo bio-stimulate chemically damaged cells. In this paper we address our proposed system of robotics, the online motion planning for obstacle avoidance for robotics guidance system. The dental robotic could support the dental surgeons in their interventions both physically and visually, since it reduce surgery time, that’s what our new approach proves, and to improve accuracy and quality through the new version of NAF-BBO approach. The surgeon\s procedure is programmed basically in the software and by employing these algorithms the robot will guide the doctor through showing everything on a screen. And if there is any change in the procedure, the doctor can update the software easily.

Dentifrice delivered by toothpaste or mouthwash could patrol all sub and supra gingival surfaces at least one time or more a daily, metabolizing trapped organic matter into odorless vapors and harmless and performing continuous debridement. By these dental procedures, gingival diseases and tooth decay can be prevented [1-5].

Robots are basically used for industrial purposes however they can be employed for the benefits of the medical field. The manipulator is a group of links that are connected by joints, these joint can be either prismatic or revolute, whereas and prismatic joint has linear motion and revolute joint has rotary motion.

Each joint provides one Degree Of Freedom (DOF) [6-7].

Minimally, invasive, surgery (MIS) for therapy and treatment is considered to be one of the promising choices in many clinical applications [8]. Particularly, percutaneous involvement is concerned more interest, because of the insignificant access requirements for this kind of process [9].

It's basically applied with a conventional needle because surgeons can assume that its performance is applicable in the tissue automatically through the insertion procedure. The efforts to use flexible needle to reach a target while obstacle avoiding areas can be seen in [10-12]. Some examples that can be considered are the magnetically driven needle navigation system is proposed in [13], at which the needle tip with the existence of an obstacle is controlled via a magnetic force is. Additionally, in [14] a lung biopsy is described which is supported by a robotic application.

In this line of thorough ideas, our work proposes a dental robotics guidance system in which we apply the direct kinematics and inverse dynamics. The optimum path results in minimizing both time and space, provided that not to exceed a maximum predefined torque without collision with the 'no-go' areas.

Bearing these new approaches in mind, the structure of our paper is as follow:

Section (II) presents the motion strategy; section (III) presents the NAF-Particle swarm optimization scheme; section (V) presents the path planner simulation results; and section (IV) presents the conclusion and future work.

2. Motion Strategy for the path planner

Several segments connect our suggested trajectory with continuous acceleration. This is done via intermediate points as shown in Figure (1). The points that should be passed through are the intermediate points.

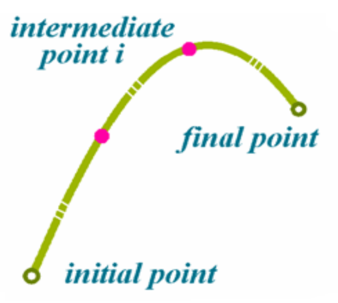


Figure (1) point to point motion planning

For the robotic guidance system, we consider that the number of DOF is *n and m respectively*. A quadrinomial and quantic polynomial can be considered in order to reach a certain position, velocity, and acceleration at each segments ends. Assume that there are m_p's points between both start point (Base) and final points (Tip).

Considering that a quadrinomial polynomial is used to describe the segments between that start point and m_p midpoint as in equations [1-4]:

$$\theta_{i,i+1}(t) = a_{i0} + a_{i1}t_i + a_{i2}t_i^2 + a_{i3}t_i^3 + a_{i4}t_i^4 \quad (i=0,\ 1,\ 2\ldots,\ mp-1) \quad \ldots\ldots (1)$$

Where (a_{i0} to a_{i4}) are constants, and let angle constraints as below:

$$\theta_i = a_{i0} \quad \ldots\ldots (2)$$

$$\theta_{i+1} = a_{i0} + a_{i1}T_i + a_{i2}T_i^3 + a_{i4}T_i^4 \quad \ldots\ldots (3)$$

$$\dot{\theta}_i = a_{i1} \quad \ldots\ldots (4)$$

$$\dot{\theta}_{i+1} = a_{i1} + 2a_{i2}T_i + 3a_{i3}T_i^2 + 4a_{i4}T_i^3 \quad \ldots\ldots (5)$$

$$\ddot{\theta}_i = 2a_{i2} \quad \ldots\ldots (6)$$

T_i is the total time from points i to next point. Therefore we can solve for the constants:

$$a_{i0} = \theta_i \quad \ldots\ldots (7)$$

$$a_{i1} = \dot{\theta}_i \quad \ldots\ldots (8)$$

$$a_{i2} = \ddot{\theta}_i / 2 \quad \ldots\ldots (9)$$

$$a_{i3} = (4\theta_{i+1} - \dot{\theta}_{i+1}T_i - 4\theta_i - 3\ddot{\theta}_i T_i^2)/T_i^3 \quad \ldots\ldots (10)$$

$$a_{i4} = (\dot{\theta}_{i+1}T_i - 3\theta_{i+1} + 3\theta_i + 2\dot{\theta}_i T_i + \ddot{\theta}_i T_i^2/2)/T_i^4 \quad \ldots\ldots (11)$$

The acceleration of the mid-point *(i+1)* will be as:

$$\ddot{\theta}_{i+1} = 2a_{12} + 6a_{i3}T_i + 12a_{i4}T_i^2 \quad \ldots\ldots (12)$$

We consider that a quantic polynomial is used to describe the segment between m_p mid and the final point as in equation as below:

$$\theta_{i,i+1}(t) = b_{i0} + b_{i1}t_i + b_{i2}t_i^2 + b_{i3}t_i^3 + b_{i4}t_i^4 + b_{i5}t_i^5 \quad \ldots\ldots(13)$$

Where constants are solved as :

$$\theta_i = b_{i0} \quad \text{……..} \quad (14)$$

$$\theta_{i+1} = b_{i0} + b_{i1}T_i + b_{i2}T_i^2 + b_{i3}T_i^3 + b_{i4}T_i^4 + b_{i5}T_i^5 \quad \text{……..} \quad (15)$$

$$\dot{\theta}_i = b_{i1} \quad \text{……..} \quad (16)$$

$$\dot{\theta}_{i+1} = b_{i1} + 2b_{i2}T_i + 3b_{i3}T_i^2 + 4b_{i4}T_i^3 + 5b_{i5}T_i^4 \quad \text{……..} \quad (17)$$

$$\ddot{\theta}_i = 2b_{i2} \quad \text{……..} \quad (18)$$

$$\ddot{\theta}_{i+1} = 2b_{i2} + 6b_{i3}T_i + 12b_{i4}T_i^2 + 20b_{i5}T_i^3 \quad \text{……..} \quad (19)$$

And the solutions to these constraints are:

$$b_{i0} = \theta_i \quad \text{……..} \quad (20)$$

$$b_{i1} = \dot{\theta}_i \quad \text{………} \quad (21)$$

$$b_{i2} = \ddot{\theta}_i/2 \quad \text{………} \quad (22)$$

$$b_{i3} = (20\theta_{i+1} - 20\theta_i - (8\dot{\theta}_{i+1} + 12\dot{\theta}_i)T_i - (3\ddot{\theta}_i - \ddot{\theta}_{i+1})T_i^2)/2T_i^3 \quad \text{….} \quad (23)$$

$$b_{i4} = (12\theta_{i+1} - 12\theta_i - (6\dot{\theta}_{i+1} + 6\dot{\theta}_i)T_i - (\ddot{\theta}_i - \ddot{\theta}_{i+1})T_i^2)/(2T_i^5) \quad \text{…..} \quad (24)$$

As formed and considered above; the angles, angle velocities, the total time for each segment, and the final configuration (n-m) for each midpoint are obtained.
Knowing that the solving nine parameters for three links robot case, by assuming that n =3, and m_p= 1.

3. NAF-Particle Swarm Optimization Scheme

3.1 Fuzzy Logic (FL)

Since fuzzy logic (FL) was invented, it had many successful applications mostly in control. One of the main advantages of fuzzy logic system is the design on the basis of incomplete and approximate information, thus providing simple and fast approximations of the unknown or too complicated models [15, 16].

The main idea of the Fuzzy Logic control, which had proved a very successful method, is build a human control expert model who is able to control the plant without thinking in terms of mathematical model. Usually the Mamdani method is used in adaptive fuzzy logic controller system. For example: if X & Y are the inputs of the fuzzy system, and "F" is the output signal:

IF X is A_1 *AND* Y is B_1 *THEN* $z=f_1$
IF X is A_2 *AND* Y is B_2 *THEN* $z=f_2$
The output "F" can be constructed as:

$$F = \frac{W_1}{W_1+W_2} f_1 + \frac{W_2}{W_1+W_2} f_2 \quad \text{……..} \quad (25)$$

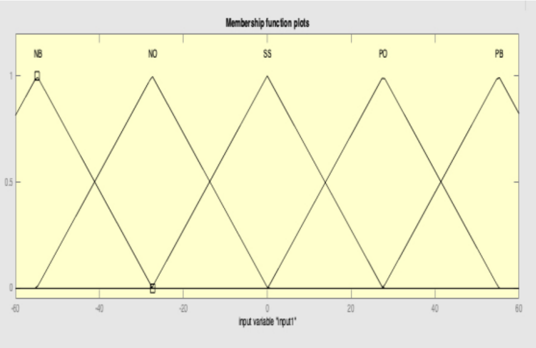
Where: A_1, A2, B1, B2 are the input membership functions, f_1 and f_2 are the output singleton membership functions, and W_1 and W_2 are the Degree Of Fulfillments (DOF) of rule 1 & 2, which can be adaptive to satisfied the input/output data [15, 16]. As mentioned before the fuzzy rules and Membership Functions (MFs) selection for variables of the input and output are not always easy. There is no formal framework to choose Fuzzy Logic Control (FLC) parameters therefore the tuning them and learning models becomes an important subject of fuzzy control.

3.2 Design of FL Controller

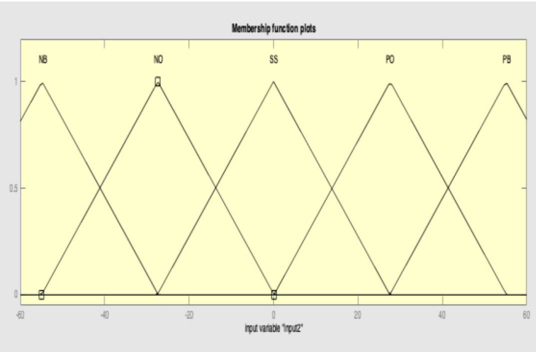
The error signal of the speed loop has a pattern which observed by the function of the fuzzy controller and compatibly updates the control signal so that the real speed (w_r) matches the command speed (w_r^*).
The fuzzy controller has two input signals.

The error $e = w_r^* - w_r$

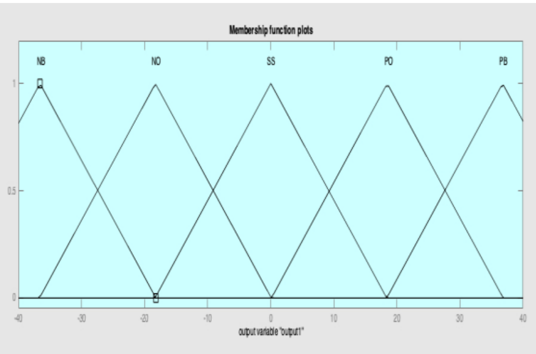
And the change of error *CE* (also it known as the future of error), which is related to the derivative of error *dE/dt*. A simple fuzzy controller with two inputs and one output can be used in this application, for each input three of triangle memberships was used, and five triangle memberships for the output. For such FLC nine (3*3) of *"If"* statement rules was used. Adapting of the input/output memberships positions are very difficult, for anyone doesn't have required experience in system behavior. In such situations, trial and error method can be used, in this work the final suitable distribution of the input/output memberships after several trial and error iterations can be shown in Figure (2). And system performance of different command speed, with fuzzy logic controller, under full-load condition is shown in the Figure (3).



(a)



(b)



(c)

Figure (2)(a),(b), (c) *Ordinary Adapting of Input/Output MFs*
(c)

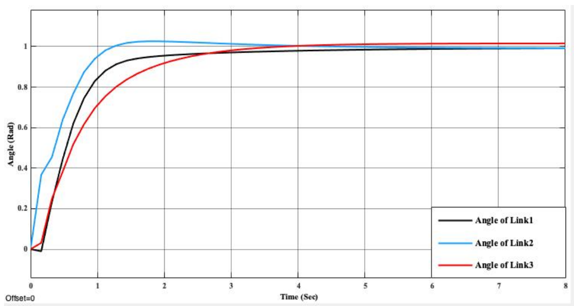


Figure (3) *System Performance of Ordinary Adapting FLC*

3.3 Particle Swarm Optimization (PSO)

Eberhart and his Colleagues are the first who proposed the optimization technique of Particle Swarm Optimization (PSO) [17, 18, and 19]. According to the social-psychological theory, the PSO has been found to solve, robustly a non-differentiability and non-linearity problem.

. This technique is derivative from many researches on swarm such as bird flocking and fish schooling. To manipulate PSO algorithm, the population dynamics simulates a bird flock's behavior instead of using operators of evolutionary like mutation and crossover. The sharing of social information is Bird flock's behavior, the swarm represents the population and the particle represents each companion. The Swarm is supposed to fly in the area space of many directions according to fitness function (F.F) [18, 20, 21, and 22].

For optimization n-variables problem, a group of particles are put into n-dimensional searching area space with positions and velocities which randomly chosen. So the best values of positions and velocities (P_{best}) is the best position which it's the optimal value obtained of the same particle [17, 22] are known. According to own particle and other particles flying experience, the velocity of flying can be adjusted. For example, the i_{th} particle is represented, as:,

$$x_i = (x_{i,1}, x_{i,2}, x_{i,3}, \dots \dots .. x_{i,n}) \qquad \dots\dots\dots(26)$$

In n-dimensional space, the best previous position of the i_{th} particle is recorded as:

$$P_{best\,i} = (P_{best\,i,1}, P_{best\,i,2}, P_{best\,i,3} \dots \dots . P_{best\,i,n}) \quad \dots..(27)$$

For each particle, the modified position and velocity can be calculated by using the distance and current velocity from ($P_{best\,i,d}$) to ($g_{best\,d}$) as shown in the formula [17, 18, 19, and 21]:

$$V_{i,m}^{(It.+1)} = W * V_{i,m}^{(It.)} + c1 * rand * \left(P_{best\,i,m} - x_{i,m}^{(It.)}\right) + c2 * rand * (g_{best\,m} - x_{i,m}^{(It.)}) \;\dots.\,(28)$$

$$x_{i,m}^{(It.+1)} = x_{i,m}^{(It.)} + v_{i,m}^{(It.)} \qquad \dots\dots\dots.(29)$$

i=1, 2, 3….., n

m=1, 2, 3….., d

Where;
n = Particles No.

d = Dimension of area space.

It. = No. of Iterations.

$V_{i,m}^{(It.)}$ = Velocity of (i) particle at iteration It.

W = factor of Inertia weight.

c1,c2 = Constant of Acceleration.

rand = Random Fractional number from 0 to 1.

$x_{i,m}^{(It.)}$ = The Current position of particle *i* at iteration It.

P_{best_i} = Best previous position of i_{th} particle.

g_{best_m} = Global best particle in the population.

3.4 Implementing PSO for adapting FL controller

The implementation of PSO is complex, because of the system performance have to be examined in particles position and each iteration according to PSO algorithm mentioned below. MATLAB is used to check the performance of the system. The problem of this work is to optimize each triangle memberships by recognize three parameters: left corner, right corner, and center. Therefore, for six memberships (of the two inputs) eighteen (6*3) parameters must be adapted. Thus, the particles (birds) have eighteen dimensions, or in other words particles must 'fly' in eighteen dimensional spaces. A random of 15 particles positions is assumed (for each dimension 15 birds), and optimization algorithm of 20 iteration is used to estimate the optimal positions of the two inputs memberships parameters.

In most intelligent optimization algorithms, there are commonly performance criteria such as: Integrated Time weight Square Error (ITSE), Integrated of Absolute Error (IAE), and the Integrated Square Error (ISE).That can be analytically estimated in the frequency domain [17, 19].

The main disadvantage of this criterion is IAE and ISE criterion which can overcome them by ITSE performance criterion. The disadvantage of the minimization of IAE and ISE can result in a response with long settling time and small overshot, because of the performance of ISE criteria dals with all errors equally Regardless of time.

. The IAE, ISE, and ITSE criterion formulas are as below:

$$IAE = \int_0^t |r(t) - y(t)|dt = \int_0^\infty |e(t)|dt$$
……..(30)

$$ISE = \int_0^t e^2(t)dt$$
…………(31)

$$ITSE = \int_0^t t * e^2(t)dt$$
………(32)

In this paper the optimization category is (ITSE) integrated time square error therefore accuracy performance of the fuzzy logic controller. Fitness Function (F.F) can evaluated to minimize the total performance using a set parameters of good control which can produce the best path. Fitness Function (F.F) can be as below:

$FF=\beta_1 f_{ot}+\beta_2 f_q+\beta_3 f_c+\beta_4 tim$ ……….(33)

The optimization aim contains to find the set of parameters that reduce f_f according to the given by the weight factors β_i (i = 1,2, .., 4).

The steady state error, Max. Overshot, rise time and settling time values depending on the value of β which if it's greater than 0.7, the steady state error, Max. Overshot are decrease. If β less than 0.7, the rise and settling time will decrease.

The flowchart in figure (4) shows the PSO algorithm also the objective function is to minimize the fitness function FF. And the final obtained positions of the membership functions from the particle swarm optimization algorithm are illustrated in Figure (5). System performance of PSO-based for different command speed is shown in figure (6).

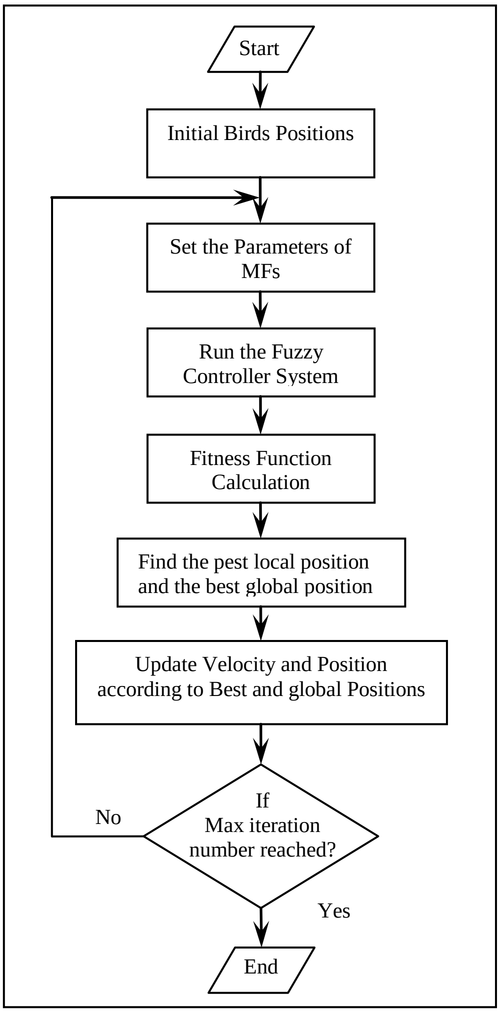
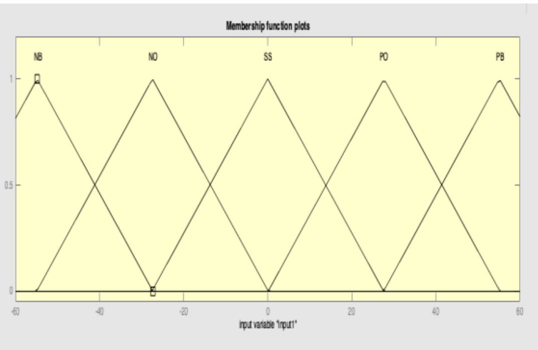
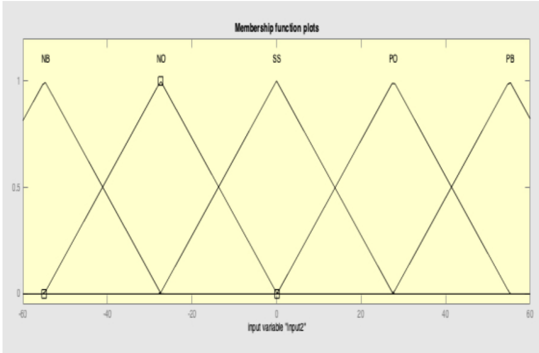


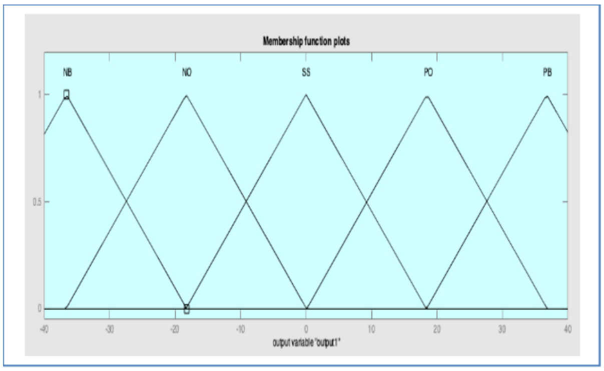
Figure (4) *Flowchart of NAF-PSO Algorithm*



(a)



(b)



(c)

Figure (5) *PSO Adapting of Input MFs*

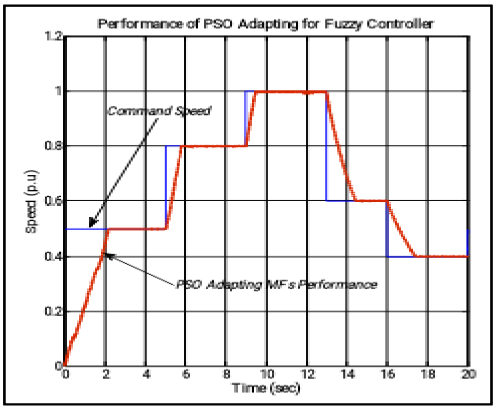


Figure (6) *System Performance of PSO Adapting FLC*

For simplicity, all mass located at the distal end of each link as a point mass as shown in figure (7).

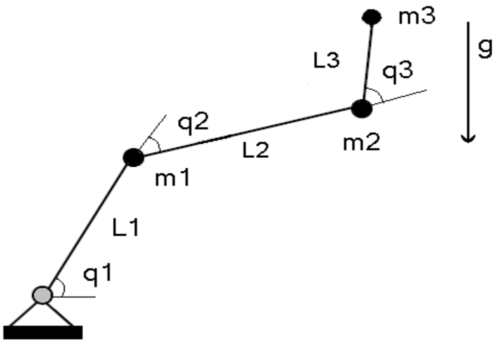


Figure (7) The3R robot

4. Path Planar Simulation Results

This section shows the results of the simulation by using MATLAB/Simulink of the 3-DOF robot manipulator with PSO case. It is illustrated on applying 3-link robot which starts and end point coordinates will be as in the table (1)

Table (1) starts and end point coordinate

	X (m)	Y (m)	$\theta_{g\ (Deg.)}$
Start Point	0	2.3	80
End Point	-2	0	

The robot links parameters are shown in the table (2)

Table (2) robot links parameters

	Length (m)	Mass (kg)	Max. Torque (N.m)
Link1	1	1	45
Link2	1	1	20
Link3	0.5	0.5	5

Whereas, the velocity and acceleration of the start-, end-configuration are considered to be zeros. Additionally, all joints in the robot rotate 2π. The obstacle has a shape of a circle with radius 0.45 m.

PSO generates the rule base for the Fuzzy Controller with the following range:

$-\pi \le q_i \le +\pi$ Rad (i=1, 2, 3) .... (34)

$-\pi \le q_g \le +\pi$ Rad ...... (35)

$-\pi/4 \le \dot{q}_i \le +\pi/4$ Rad/ sec (i=1, 2, 3) ....(36)

$0.1 \le t_i \le 8$ Sec (i=1, 2) ....(37)

Figures (8 to11) demonstrate the results of optimization for free areas.

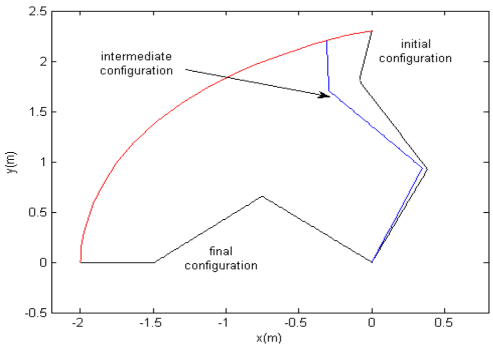


Figure (8) Cartesian path for the robotic guidance system

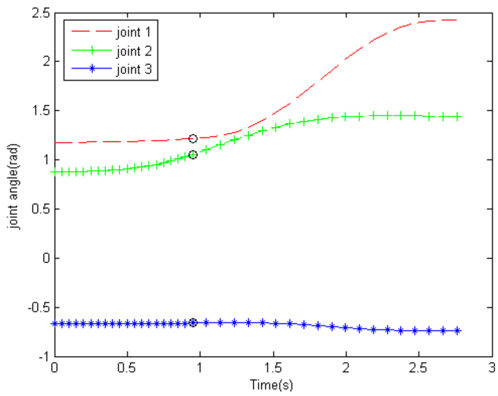


Figure (9) Joint angle in free areas

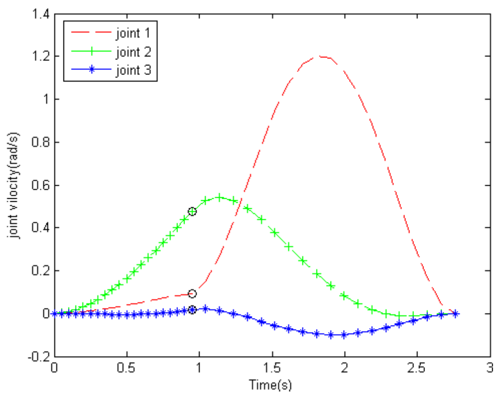


Figure (10) Joints velocity in free areas

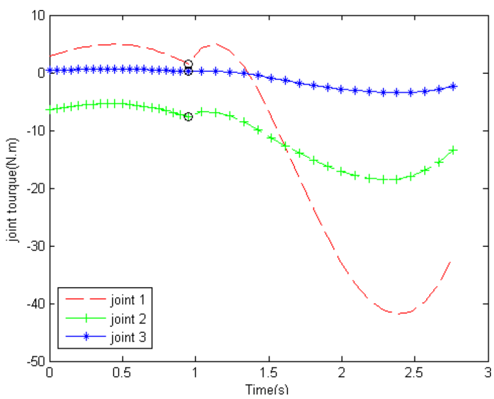


Figure (11) joints torque versus time in free workspace

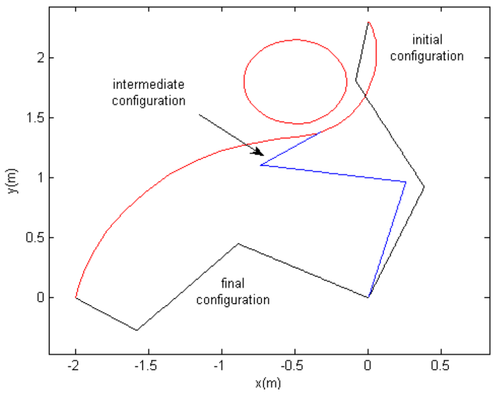


Figure (12) Obstacle existence through the Cartesian path

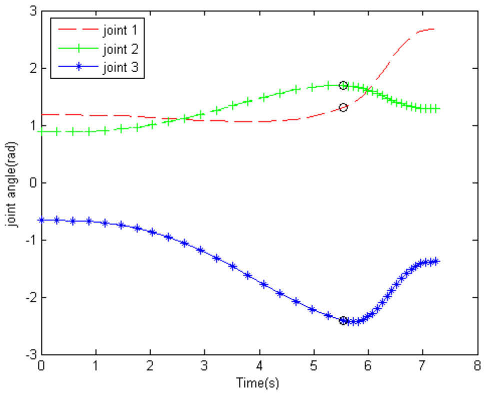


Figure (13) Joint angle in the existence of an obstacle

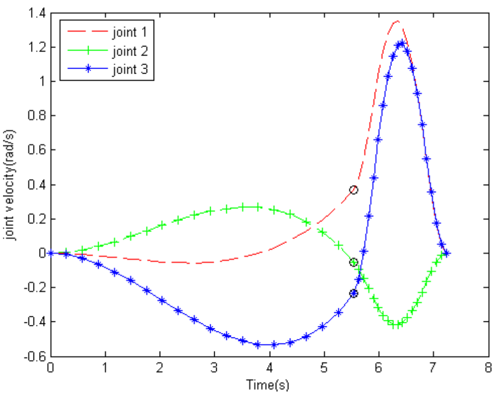


Figure (14) Joint velocity in the existence of an obstacle

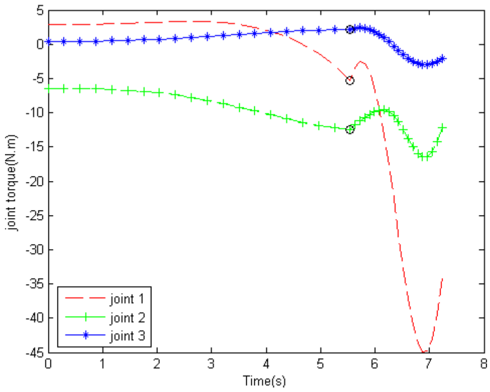


Figure (15)Joint torque in the existence of an obstacle

In the existence of an obstacle with the dimensions (x=-0.5, y=1.8), the results of PSO optimization are shown in Figures12-15.

Fig. 8 defines the path in the Cartesian coordinate. Whereas, the shortest line is the one from the start to end point, but is different from the best one results from the PSO optimization. The black dots in Figures (9 and 12), and Figures (10 and 13) show the mid-joint angle and velocity that are optimized, respectively. The black dots in Figures 10 and 12show the joints torque versus time in free workspace. In Figures11 and 13 the torque that obtained with obstacle existence is greater than the joint trajectory in free workspace. And in both cases, the not exceeding the predefined torque is achieved.

Figure 12 shows the application of NAF-PSO and the obstacle existence through the Cartesian path.

Table.3 shows the value of overall time, overall traveling distance and Overall trajectory length.

Table.3 simulation results

Result value	Free workspace	Obstacle workspace
Overall time (sec)	2.75	7.24
Overall distance (rad)	1.92	5.79
Overall trajectory length (m)	3.25	3.41

5. Conclusion and Future Work

A comparison performance between the proposed PSO method and the ordinary adapting method of the fuzzy controller membership functions is illustrated in figure (16), from which the following tips can be concluded:

i. The proposed PSO adapting method can be considered as: very computationally efficient in eliminate computing time, easy to implement, and simple concept. Unlike, the ordinary adapting method which needs long adapting time and complex procedure.

ii. In the ordinary adapting method, the fuzzy controller gives a satisfactory system performance: max. overshot = 4% in low seed & 0% in high speed, rise time = 6 sec, settling time = 6.5 sec, and steady state error = 0%.

iii. The proposed PSO adapting method gives a superior performance, in which: max. overshot = 0% in low and high speed variation, rise time = 3 sec, settling time = 3 sec, and steady state error = 0%.

The proposed PSO adapting method gives a perfect speed tracking with non-sluggish performance and high robust controller than those obtained by the ordinary adapting of fuzzy logic controllers

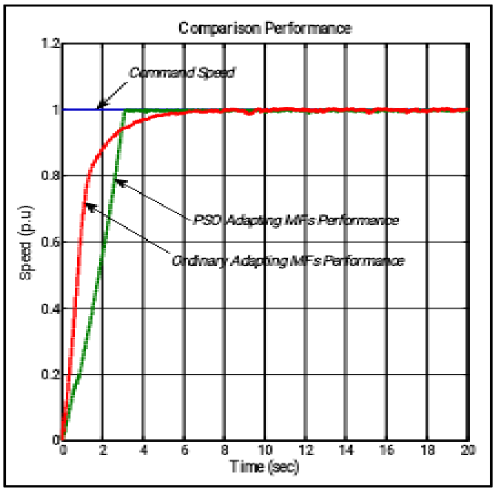


Figure (16) *Comparison Performance*

References

Freitas RA Jr. Nanodentistry. Journal of American Dental Association. 2000;131(11):1559-65.

West JL, Halas NJ. Applications of nanotechnology to biotechnology commentary. Curr OpinBiotechnol.2000;11:215-7.

Shi H, Tsai WB, Garrison MD, Ferrari S, Ratner BD. Templateimprinted nanostructured surfaces for protein recognition. Nature. 1999;398:593-7.

Sims MR. Brackets, epitopes and flash memory cards: a futuristic view of clinical orthodontics. AustOrthod J. 1999; 15: 260-8.

Slavkin HC. Entering the era of molecular dentistry. J Am Dent Assoc. 1999 Mar;130:413-7.

T. R. Kurfess, Robotics and automation handbook: CRC, 2005.

L. Cheng, Z. G. Hou, M. Tan, D. Liu and A. M. Zou, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

G. Dogangil, B. L. Davies, and F. Rodriguez y Baena, "A review of medical robotics for minimally invasive soft tissue surgery," Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine, vol. 224, pp. 653-679, 2010.

N. Abolhassani, R. Patel, and M. Moallem, "Needle insertion into soft tissue: A survey," Medical Engineering & Physics, vol. 29, pp. 413–431, 2007.

L. Frasson, S. Y. Ko, A. Turner, et al., "STING: a soft-tissue intervention and neurosurgical guide to access deep brain lesions through curved trajectories," Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine, vol. 224, pp. 775-788, 2010.

S. Y. Ko, L. Frasson, and F. Rodriguez y Baena, "Closed-Loop Planar Motion Control of a Steerable Probe with a "Programmable Bevel" Inspired by Nature," IEEE Transactions on Robotics, vol. 27, pp. 970 - 983, Oct 2011.

S. Y. Ko and F. Rodriguez y Baena, "Trajectory Following for a Flexible Probe with State/Input Constraints: an Approach based on Model Predictive Control," Robotics and Autonomous Systems, vol. 60, pp. 509-521, April 2012.

L. Tang, Y. Chen, and X. He, "Magnetic Force Aided Compliant Needle Navigation and Needle Performance Analysis," in IEEE International Conference on Robotics and Biomimetics, Sanya, China, 2007, pp. 612-616.

L. B. Kratchman, M. M. Rahman, J. R. Saunders, et al., "Toward Robotic Needle Steering in Lung Biopsy: A Tendon-Actuated Approach," in Proc. of SPIE, 2011.

Bimal K. Bose, " Modern Power Electronic and AC Drives", Prentice Hall, 2002.

K. M. Passino and S. Yurkovich, "Fuzzy control", Addison – Wesley Longman, 1998.

B. Allaoua, B. Gasbaoui, B. Mebarki, "Setting Up PID DC Motor Speed Control Alteration Parameters Using Particle Swarm Optimization Strategy", Leonardo Electronic Journal of Practices and Technologies, Issue 14, P 19-32, 2009.

B. Nagaraj, N. Murugananth, "Soft Computing-Based Optimum Design of PID Controller for a Position Control of DC Motor", Mediamira Science Publisher, Vol. 51, No. 1, 2010

Lucas Alves Dias, Roger William de Oliveira Silva, Paulo César da Silva Emanuel, André Ferrus Filho, and Rodrigo Teixeira Bento, "Application of the Fuzzy Logic for the Development of Automnomous Robot with Obstacles Deviation" International Journal of Control, Automation, and Systems , vol. 16, no. 2, pp.823-833, 2018

Soundarrajan, Dr.S.Sumathi, C.Sundar"Particle Swarm Optimization Based LFC and AVR of Autonomous Power Generating System", IAENG International Journal of Computer Science, 37:1, February, 2010.

Thanga, S.P. Snvastava, and P. Agarwal, "Particle Swarm and Fuzzy Logic Based Optimal Energy Control of Induction Motor for a Mine Hoist Load Diagram", IAENG International Journal of Computer Science, 36:1, 2009.

M. Nasri, H. Nezamabadi, M. Maghfoori, "A PSO Based Optimum Design of PID Controller for a Linear Brushless DC Motor", World Academy of Science, Engineering and Technology 26, 2007.

Received : June 2020
Revised :July 2020
Accepted : August 2020