The 2013 World Congress on
Advances in Nano, Biomechanics, Robotics, and Energy Research (ANBRE13)
Seoul, Korea, August 25-28, 2013

Path planning for robotic manipulators using a new hybrid neuro-fuzzy technique

*Rodolfo Ponce[1)], Emmanuel Alejandro Merchán[2)] and Luis Héctor Hernández[3)]

[1), 2) ,3)] *Department of Mechanical Engineering, IPN (National Polytechnic Institute), Mexico City, Mexico*
[1)] ponce.reynoso@gmail.com

ABSTRACT

A new neuro-fuzzy technique is explained in this work to deal with the path planning problem for robotic manipulators, taking into account obstacles avoidance. First, a neuro-fuzzy system is employed to solve the inverse kinematics problem of a two and three degree of freedom manipulator. Then, cycloidal motion functions are used to generate minimum-time trajectories (position, velocity, acceleration and jerk), taking into account the admissible capabilities of the actuators of the robot. Finally, it is explored the artificial potential field (APF) method in order to generate obstacle-free paths for the robotic manipulators.

1. INTRODUCTION

The motions of a robotic manipulator should be, as a rule, as smooth as possible; *i.e.*, abrupt changes in position, velocity and acceleration should be avoided. Furthermore, abrupt motion changes arise when the robot collides with an object, a situation that should also be avoided. Due to the actual complex robotic manipulator interaction for executing specific tasks (among them or with its work environment), unpredictable situations should thus be taken into account when designing a robotic system controller, which can be done by supplying the system with sensors for the automatic detection of unexpected events or by providing for human monitoring. Two typical tasks call for trajectory planning techniques, namely,

- Pick-and-place operations (PPO), and
- Continuous paths (CP).

The path planning of robotic manipulators consists of finding a feasible path to be tracked by its end effector, while satisfying some imposed constraints, e.g. joint's angle limits. On the other hand, the trajectory planning deals with the time

[1)] Ph.D. Student
[2)] Professor

history of the joint variables and its derivatives, taking into account different important parameters such as total travelling time and joint's torque limits. There are several approaches and methods to deal with all these issues. Some of them are focused on finding a path in a workspace with obstacles that can be static or dynamic and have to be avoided in order not to collide with them.

Other methodologies are intended to generate minimum travelling time for the joints (cycloidal functions) and some others are employed to explode the redundancy property presented in some manipulators. Several research works related with the use of soft computing methodologies have been published to deal with these issues, among them (Khoukhi *et al* 2007).

As mentioned in many other publications, artificial potential fields (APF) are popular and useful in solving path planning problems with collision avoidance. There is the possibility, however, to get stuck in a local minimum, which not represent a global minimum, i. e., the goal configuration.

1.1 Adaptive Network based Fuzzy Inference System (ANFIS)

This kind of neural network was originally introduced by (Jang 1993). It combines the advantages of a neural network (learning and generalization capability) and of a fuzzy inference system (ambiguous and imprecision reasoning). It uses a hybrid training algorithm, one for adjusting the consequent parameters of Takagi-Sugeno-type rules (recursive least squares), and one for the antecedent parameters of the membership functions (gradient descent) (Jang *et al* 1997). ANFIS architecture is shown in Fig 1.

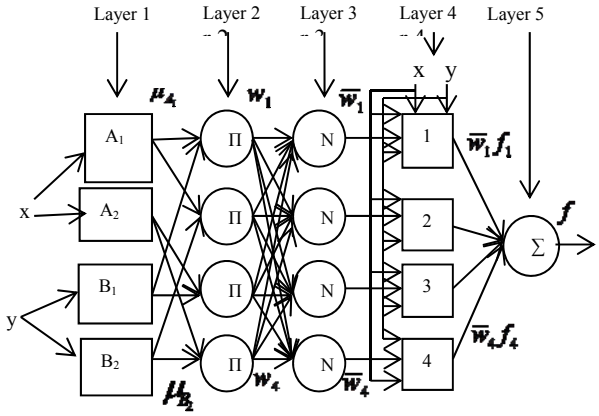


Fig. 1 ANFIS architecture (Jang 1993)

1.2 Cycloidal motion

This is an alternative motion that produces zero velocity and acceleration at the ends of a finite interval. A cycloidal function models the trajectory time t from *0* to T with the normalized time τ defined as:

$$\tau = \frac{t}{T} \qquad 0 \leq t \leq T\,,\ 0 \leq \tau \leq 1 \tag{1}$$

In normal form, this motion is given by (Angeles 2007):

$$s(\tau) = \tau - \frac{1}{2\pi}\sin 2\pi\tau \quad (2)$$

And its derivatives are:

$$s'(\tau) = 1 - \cos 2\pi\tau \quad (3)$$

$$s''(\tau) = 2\pi \sin 2\pi\tau \quad (4)$$

$$s'''(\tau) = 4\pi^2 \cos 2\pi\tau \quad (5)$$

When implementing the cycloidal motion in pick and place operations (PPO), we have, for the *jth* joint:

$$q_j(t) = q_j^I + \left(q_j^F - q_j^I\right)s(\tau) \quad (6)$$

$$\dot{q}_j(t) = \frac{\left(q_j^F - q_j^I\right)}{T}\, s'(\tau) \quad (7)$$

$$\ddot{q}_j(t) = \frac{\left(q_j^F - q_j^I\right)}{T^2}\, s''(\tau) \quad (8)$$

$$\dddot{q}_j(t) = \frac{\left(q_j^F - q_j^I\right)}{T^3}\, s'''(\tau) \quad (9)$$

Whence,

$\dot{q}_j(t), \ddot{q}_j(t), \dddot{q}_j(t) =$ Angular velocity, acceleration and jerk of the *jth* joint in rad/s, rad/s^2 and rad/s^3, respectively.

q_j^F, q_j^I= Final and initial angular position of the *jth* joint

1.3 Minimum-time trajectory generation

It can be seen in Fig. 2 that this kind of motion attains its maximum velocity at the center of the interval, *i. e.*, at $\tau = 0.5$, being the maximum:

$$s'_{max} = s'(0.5) = 2 \quad (10)$$

And hence,

$$\left(\dot{q}_j\right)_{max} = \frac{2}{T}\left(q_j^F - q_j^I\right) \quad (11)$$

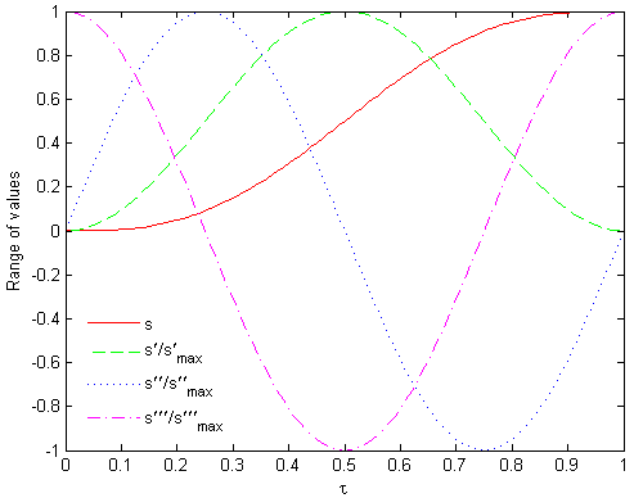


Fig. 2 The normal cycloidal motion and its time derivatives

Likewise, the *jth* joint acceleration attains its maximum and minimum values at $\tau = 0.25$ and $\tau = 0.75$, respectively, *i. e.*,

$$s''_{max} = s''(0.25) = s''(0.75) = 2\pi \tag{12}$$

And hence,

$$\left(\ddot{q}_j\right)_{max} = \frac{2\pi}{T^2}\left(q_j^F - q_j^I\right) \tag{13a}$$

$$\left(\ddot{q}_j\right)_{min} = -\frac{2\pi}{T^2}\left(q_j^F - q_j^I\right) \tag{13b}$$

Moreover, $s'''(\tau)$ attains its extrema at the ends of the interval, *i. e.*,

$$s'''_{max} = s''(0) = s''(1) = 4\pi^2 \tag{14}$$

And hence,

$$\left(\dddot{q}_j\right)_{max} = \frac{4\pi^2}{T^3}\left(q_j^F - q_j^I\right) \tag{15}$$

Thus, if motion is constrained by the maximum speed, acceleration and jerk delivered by the motors, the minimum time T_j for the *jth* joint to produce the desired pick and place operation can be easily determined as:

$$T_j = \frac{2\left(q_j^F - q_j^I\right)}{\left(\dot{q}_j\right)_{max}} \tag{16}$$

$$T_j = \sqrt{\frac{2\pi\left(q_j^F - q_j^I\right)}{\left(\ddot{q}_j\right)_{max}}} \tag{17}$$

$$T_j = \left(\frac{4\pi^2\left(q_j^F - q_j^I\right)}{\left(\ddot{q}_j\right)_{max}}\right)^{1/3} \tag{18}$$

And hence, the minimum time in which the operation can take place is found readily as:

$$T_{min} = \max_j\{T_j\} \tag{19}$$

And substituting the above expression in the equations of cycloidal motion, and inserting them into the equations of the minimum-time trajectory yields the resulting minimum-time trajectory characterized by joint position, velocity, acceleration and jerk given as:

$$q_j(t) = q_j^I + \left(q_j^F - q_j^I\right)\left(\frac{t}{T_{min}} - \frac{1}{2\pi}\sin\left(2\pi\frac{t}{T_{min}}\right)\right) \tag{20}$$

$$\dot{q}_j(t) = \frac{\left(q_j^F - q_j^I\right)}{T_{min}}\left(1 - \cos\left(2\pi\frac{t}{T_{min}}\right)\right) \tag{21}$$

$$\ddot{q}_j(t) = \frac{\left(q_j^F - q_j^I\right)}{T_{min}^2}\left(2\pi\sin\left(2\pi\frac{t}{T_{min}}\right)\right) \tag{22}$$

$$\dddot{q}_j(t) = \frac{\left(q_j^F - q_j^I\right)}{T^3}\left(4\pi^2\cos\left(2\pi\frac{t}{T_{min}}\right)\right) \tag{23}$$

1.4 Artificial potential fields (APF)

A potential function is a differentiable real-valued function $U: \mathbb{R}^m \to \mathbb{R}$. The value of a potential function can be viewed as energy and hence the gradient of the potential is force (Spong *et al* 2008). The potential field approach creates an artificial potential field (*U*) in space that reflects the structure of space. It has two components:

- An attractive potential (U_{att}) pulling a particle toward the goal.
- A repulsive potential (U_{rep}) pushing the particle away from obstacles.

A particle in space, at point q, will be under the influence of *U*. The artificial force acting on it is the gradient of *U*, which is:

$$\mathcal{F}(q) = \mathcal{F}_{att}(q) + \mathcal{F}_{rep}(q) = -\nabla \mathrm{U}(q) \tag{24}$$

Attractive potential field The attractive potential is felt over the whole region and is defined by the bounded expressions:

$$U_{att}(q) = \begin{cases} \frac{1}{2}\varepsilon\rho_f^2(q) & : \rho_f(q) \le d \\ d\varepsilon\rho_f(q) - \frac{1}{2}\varepsilon d^2 & : \rho_f(q) > d \end{cases} \tag{25}$$

Where ε is a positive scaling factor, d is the threshold distance from the goal where the planner switches between conic and quadratic potentials, and the Euclidian distance to the goal, ρ_f, is determined as:

$$\rho_f(q) = \|q - q_{goal}\| \tag{26}$$

The above equations combines the properties of quadratic and conic potentials functions, so that the conic potential attracts the robot when it is very distant from q_{goal}, and the quadratic potential attracts the robot when it is near q_{goal}.

Then, the attractive force is a vector directed toward q_{goal} with magnitude linearly related to the distance from q to q_{goal}, and is calculated by the gradient of U_{att} as:

$$\mathcal{F}_{att}(q) = -\nabla U_{att}(q) = \begin{cases} -\varepsilon(q - q_{goal}) & : \rho_f(q) \le d \\ -\frac{d\varepsilon(q - q_{goal})}{\rho_f(q)} & : \rho_f(q) > d \end{cases} \tag{27}$$

Repulsive potential field The repulsive potential keeps the robot away from an obstacle and is only felt in the vicinity of an obstacle, that is, within a distance of influence ρ_o that allows the robot to ignore obstacles sufficiently far away from it) and is:

$$U_{rep}(q) = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{\rho(q)} - \frac{1}{\rho_0}\right)^2 & : \rho(q) \le \rho_0 \\ 0 & : \rho(q) > \rho_0 \end{cases} \tag{28}$$

Where η is a positive scaling coefficient that determines the strength of the repulsive field, and $\rho(q)$ is the shortest distance from q to an obstacle (at the point b on the object closest to q), calculated as:

$$\rho(q) = \min_{b \in \text{object}} \|q - b\| \tag{29}$$

Therefore, the repulsive force from an obstacle is:

$$\mathcal{F}_{rep}(q) = -\nabla U_{rep}(q) = \begin{cases} \eta\left(\frac{1}{\rho(q)} - \frac{1}{\rho_0}\right)\frac{1}{\rho^2(q)}\nabla\rho(q) & : \rho(q) \le \rho_o \\ 0 & : \rho(q) > \rho_o \end{cases} \tag{30}$$

The gradient of the distance to the nearest obstacle at the point b is given by:

$$\nabla\rho(q) = \frac{q - b}{\|q - b\|} \tag{31}$$

An obstacle's repulsive force is calculated if q is within its distance of influence. The total repulsive force is the sum of each individual repulsive force. If the particle at q collides with the obstacle (that is, the distance from the obstacle is zero), the repulsive force from that obstacle will be infinite.

Gradient descent algorithm Gradient descent is a well-known approach to optimization problems. Starting at the initial configuration, take a small step in

the direction opposite the gradient. This gives a new configuration, and the process is repeated until the gradient is zero.

The gradient descent algorithm can be defined as follows:

1. $q^0 \leftarrow q_{init}, \ i \leftarrow 0$
2. IF $q^i \neq q_{final}$

   $q^{i+1} \leftarrow q^i + \alpha^i \frac{\mathcal{F}(q^i)}{\|\mathcal{F}(q^i)\|}$

   $i \leftarrow i + 1$

   ELSE return $< q^0, q^1, \cdots q^i >$
3. GO TO 2

The notation q^i is used to denote the value of q at the ith iteration and the final path consists of the sequence of iterations $\{q^0, q^1, \cdots q^i\}$. The value of the scalar α^i determines the step size at the *ith* iteration. It is important that α^i be small enough that the robot is not allowed to "jump into" obstacles, while being large enough that the algorithm does not require excessive computation time.

2. TRAJECTORY PLANNING

When using ANFIS for generating the robot path, the first step is to gather a suitable, enough and accurate amount of data which represent, as close as possible, the dynamic behavior of the system to be learnt.

In this work, a training table is generated by means of the direct kinematic equations of the manipulator. For the case of a two-degree of freedom planar manipulator, this input (x, y)-output (θ_1, θ_2) relationship, represents the domain and range of nonlinear functions (Denai *et al* 2004). ANFIS realizes a nonlinear mapping and learns the behavior of the data presented. Fig. 3 shows the proposed methodology. For only position of a three-degree of freedom spatial manipulator, these equations are defined by:

$$x = \cos q_1[l_3 \cos(q_2 + q_3) + l_2 \cos q_2] \tag{32}$$

$$y = \sin q_1[l_3 \cos(q_2 + q_3) + l_2 \cos q_2] \tag{33}$$

$$z = l_1 + l_3 \sin(q_2 + q_3) + l_2 \sin q_2 \tag{34}$$

Whence q_1, q_2, q_3 are the angular positions of the joints (in radians), l_1, l_2, l_3 are the link lengths in meters and x, y, z are the respective end effector positions.

Eq. (32-34) permit to calculate all possible situations of positions of the end-effector in its workspace, according to the motion range of each joint and the different length of links. The same procedure is used to generate the training table of a two-degree of freedom planar robotic manipulator.

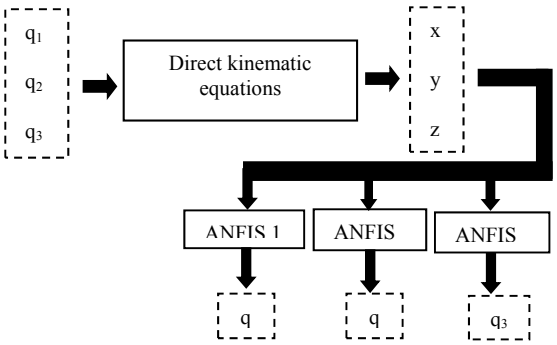


Fig. 3 Methodology to gather data and train the ANFIS systems (Inverse kinematics solution)

In this way, once the ANFIS system is trained and validated, it is capable of yielding an accurate output (solution) and therefore the complicated and time-consuming inverse kinematic equations are avoided (Spong *et al* 2008).

Once the appropriate values of the joint positions have been obtained, the next step is to use the cycloidal minimum-time functions to yield the complete trajectory profile of each joint, *i. e.*, position, velocity, acceleration and jerk. If there were obstacles inside the robot's workspace, then the APF method is a suitable approach to deal with this problem (Kathib 1986).

3. SIMULATION RESULTS

All the simulations were carried out by using Matlab® R2011a software. In the first study case 6 generalized bell type membership functions were employed for the path generation using ANFIS, 60 epochs were enough to learn the nonlinear behavior of the system, and to prevent over fitting. Likewise, for the second study case were used 4 generalized bell type membership functions for each input, and 80 epochs of training. Fig. 4 illustrates how is realized the mapping of the two-degree of freedom planar manipulator (study case I), by means of the corresponding direct kinematic equations.

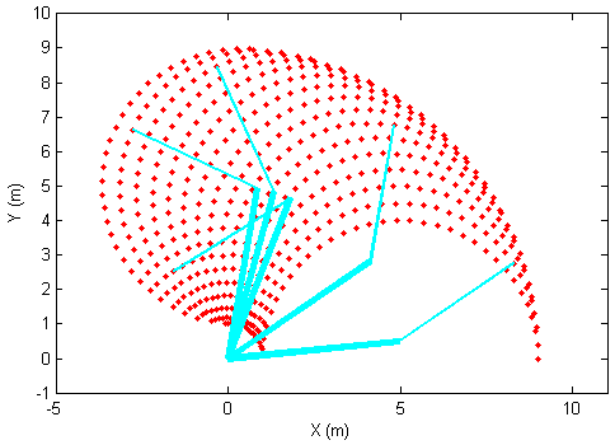


Fig. 4 Mapping of the workspace of the planar manipulator

Fig. 5 shows the resulting identification of ANFIS, when is required to yield the inverse kinematic solution to the presented end effector's path. As it is observed, the reference and the resulting ANFIS path are almost the same. Fig.

5 shows also how ANFIS is able to yield accurate results when a highly nonlinear desired path has been presented to the system.

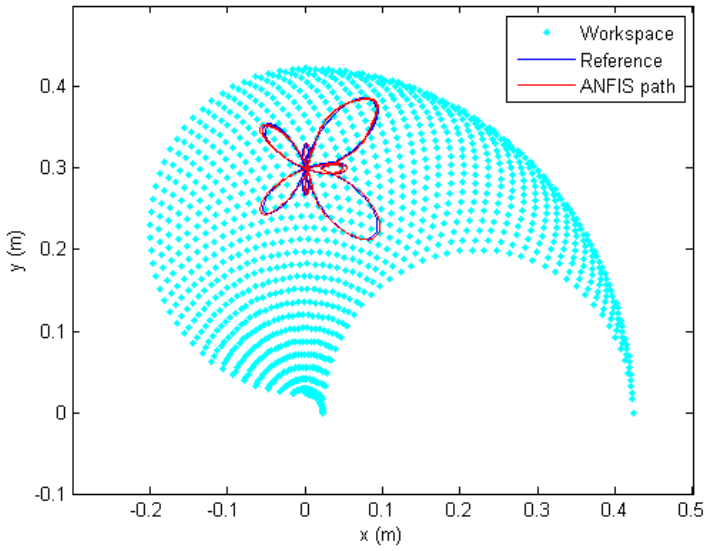


Fig. 5 Comparison among desired path and ANFIS path of the end effector in its workspace.

For the study case II, the workspace envelope is obtained by employing the direct kinematic equations, for all possible combinations of the joint's allowable angular limits. This workspace mapping is presented in Fig. 6:

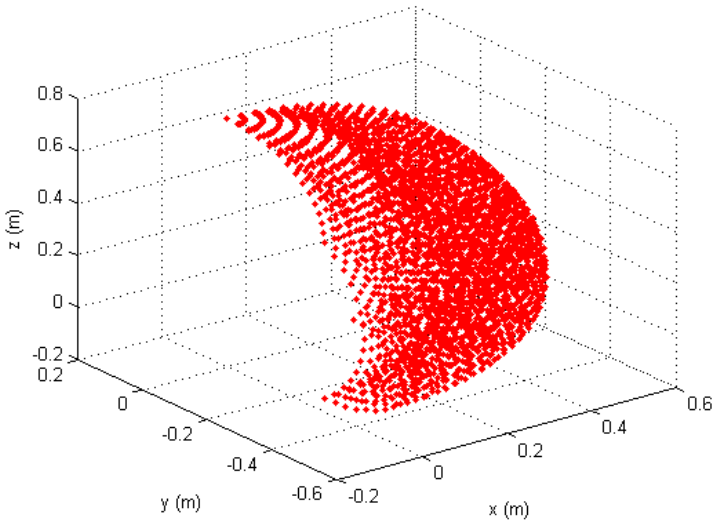


Fig. 6 Mapping of the workspace of the three-degree of freedom spatial robotic manipulator

A pick and place operation (PPO) is to be realized by the end effector of the robot, the corresponding cartesian path is obtained by using the trained ANFIS. Fig. 7 depicts a very accurate linear cartesian path.

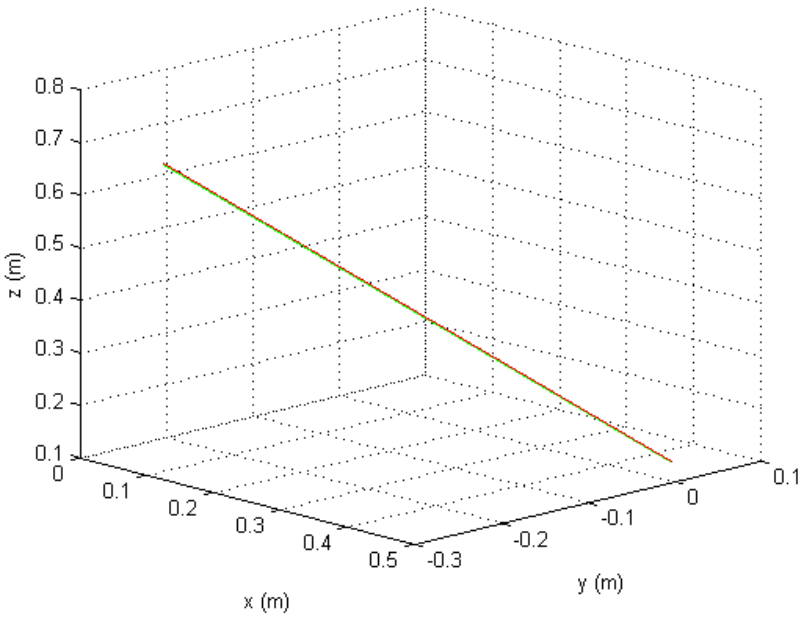
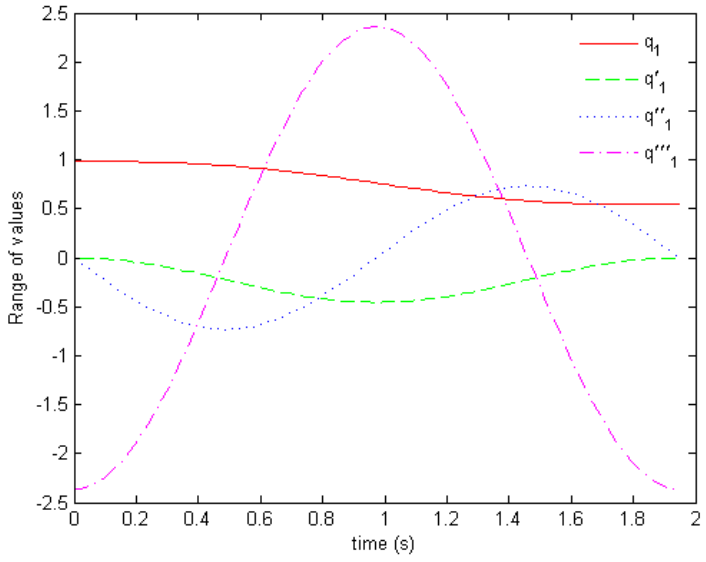
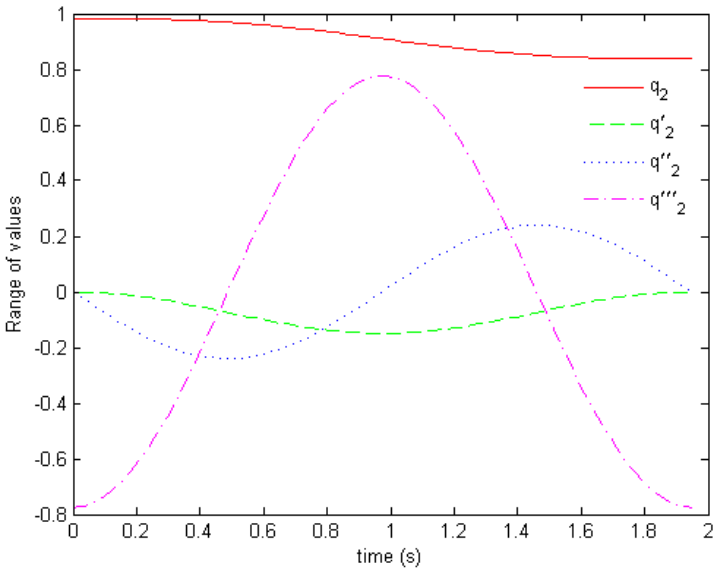


Fig. 7 Desired path (red) and ANFIS path (green), in a 3D workspace

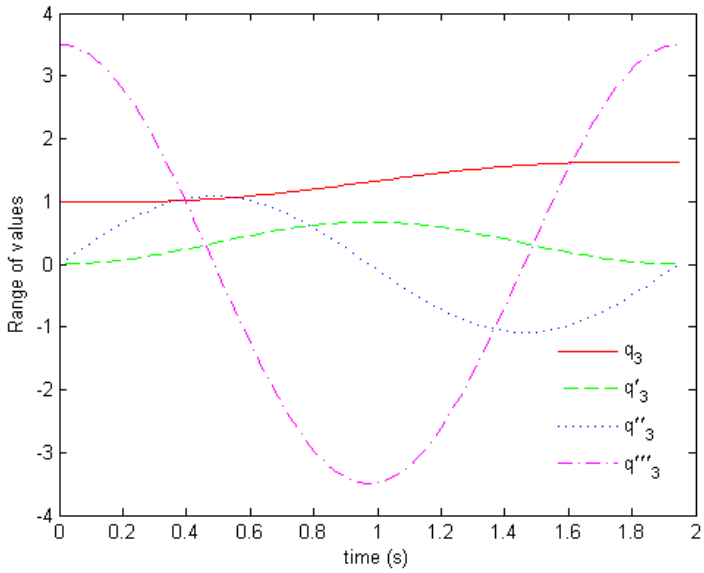
On the other hand, Fig. 8 presents the minimum-time trajectories for the robot's joints. Cycloidal motion functions were employed to produce the required trajectories. As it can be observed in Fig. 8, the trajectory profiles for the three joints are smooth and continuous, along with the minimum time necessary for generating them, and taking into account the admissible values of the actuators of the robot. However, it is necessary to mention that the jerk's profile does not start at a zero value, as it is the case for angular velocity and acceleration profiles.



(a)



(b)



(c)

Fig. 8 Minimum-time trajectory profile of joints 1 (a), 2 (b) and 3 (c). Angular position, velocity, acceleration and jerk.

In the case of present obstacles, Fig. 9 shows the free-obstacle generated path of a robot, by using the artificial potential field method. There are 6 randomly distributed point- type obstacles along its workspace. It is observed the absence of local minima in the obtained path, the proposed algorithm does not get stuck in them.

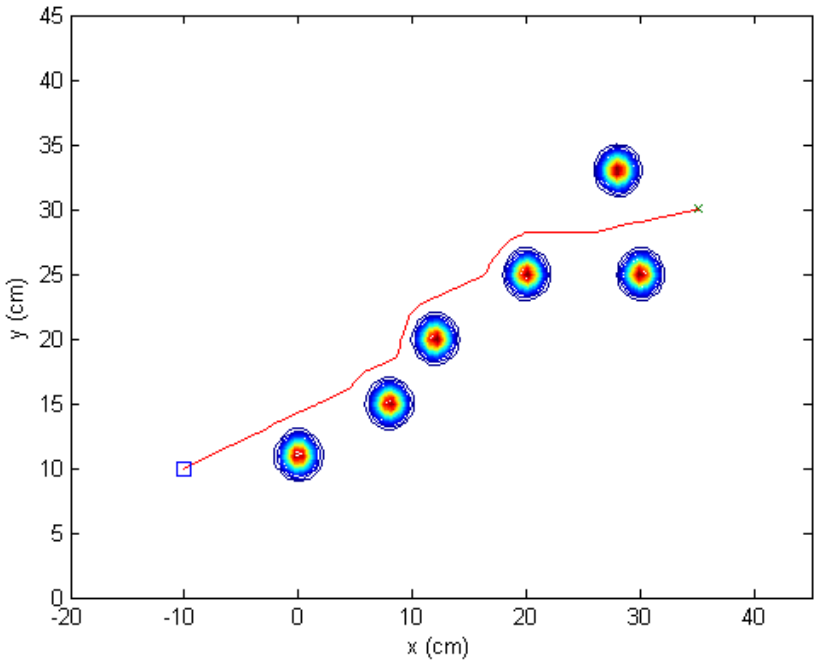


Fig. 9 Robot's path generated by using the *APF* approach. Initial (square) and goal (cross) positions.

CONCLUSIONS

The neuro-fuzzy system possess the advantage that once it has been trained, it can yield suitable and fast outputs when is required in high speed applications. Any complicated trajectory can be predicted due to the nonlinear nature of this kind of approximators. It is also possible to determine the minimum time during which can be performed a given PPO, while observing the physical limitations of the motors. This can be accomplished by using the cycloidal motion functions. Finally, the artificial potential field method demonstrates its great capability to yield a free-obstacle Cartesian path, which can be later presented to the ANFIS system and predict the necessary joint positions values, without the need of the inverse kinematics equations.

Acknowledgments

The authors would like to thank the support of CONACyT, for the fulfillment of this research work.

REFERENCES

Angeles, J. (2007), *Fundamentals of Robotic Mechanical Systems,* Springer, 3rd Ed., New York.

Denai, M. A., Palis, F. and Zeghbib, A. (2004) "ANFIS based Modelling and Control of non-linear Systems: a Tutorial", *IEEE International Conference on Systems, Man and Cybernetics*, Vol. **4**, 3433-3438.

Jang, R. (1993), "ANFIS: Adaptive Network-based Fuzzy Inference System" *IEEE Transactions on Systems, Man and Cybernetics*, Vol. **23**(3), 665-685.

Jang, R., Sun, C. T. and Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence.* Prentice Hall. Upper Saddle River, NJ.

Khatib, O. (1986), “Real-Time Obstacle Avoidance for Manipulators and Mobile Robots”. *The International Journal of Robotics*, Vol. **5**(1), 90-98.

Khoukhi, A., Demirli, K., Baron, L. and Balazinski, M. (2007) ”Neuro-fuzzy multi-objective trajectory planning of redundant manipulators”, *IEEE International Conference on Systems, Man and Cybernetics*, 2831-2836, Montreal, Que.

Spong, M, Hutchinson S. and Vidyasagar, M. (2008), *Robot Modeling and Control*, John Wiley & Sons, New York.