Designing Path for Robot Arm Extensions Series with the Aim of Avoiding Obstruction with Recurring Neural Network

Ramin Khodaie , Mohammad Esmaeil Akbari*
Department of Electrical Engineering, Ahar Branch, Islamic Azad University, Ahar, Iran
Email: ramin36kh@yahoo.com , makbari@tabrizu.ac.ir

Abstract:

In this paper, recurrent neural network is used for path planning in the joint space of the robot with obstacle in the workspace of the robot. To design the neural network, first a performance index has been defined as sum of square of error tracking of final executor. Then, obstacle avoidance scheme is presented based on its space coordinate and its minimum distance between the obstacle and each of robot links and proper inequality equations have been derived which describe the qualification of obstacle avoidance. Moreover, nonlinear optimization problem with nonlinear constraint functions has been derived by considering the joint physical limits. In order to design the neural network, equivalent problem of projection theorem has been converted to quadratic programming by using Kuhn-Tucker 1optimality conditions. Based on the projection theorem, model of recurrent neural network has been determined which is as a first order differential equation. Simulation results show the good performance of the proposed method by applying the proposed algorithm on the seven-degree freedom PA-10 robot.

Keywords: Joint path planning, extended Robot arm, Obstacle avoidance, Recurrent neural network.

1- Introduction

The optimal path design in the joint space corresponds to determining the position of are realized the robot joints angles in minimum error tracking way and the specific optimization criteria. Among these criteria we can mention the least Norm2 joint velocity or minimum energy, the joints motion repeatability, the minimum torque, singularity, Avoidingjoints physics limitation and obstacle avoidance. It should be noted that the realization of these criteria is in the second order and the first goal should be the correct tracking of the robot's final implementer to the target position. For extension robots, having the final operator position, the position of the joints cannot be analytically determined.

The use of numerical approaches to solve the problem is necessary in these Conditions. Among the most prominent and most widely used methods in the kinetics inversion field, methods are based on the Jacobi inversion matrix. In scheme [1] Jacobi pseudo-inversion matrix is used along with obstacles avoidance program. To avoid obstacles, a developed form of Jacobi pseudo-inversion has been proposed. The redundancy decomposition of the gradient image of the cost function into the empty space of the matrix uses this approach and uses Jacobi. In [2] by using this approach and in the closed loop and with a feedback from the Cartesian error, the seven-degree freedom Robotic kinematics of inverse is carried out. In [3] a combination of the dissipated least squares, and genetic algorithms is used. At [4] a duplicate approach is used based on geometric information of robot configuration.

In the first phase, a repetitive process is the passing of final implementer to the robot base and then from the base of the robot to the final implementer. At [5] the geometric method is proposed thatby assuming constant angles to the next and considering two distinct angles in direct kinematic equations, the proper configuration of the joints is determined. At [6] inverse kinematic solution is done by combination of numerical analytic methods. Also some researches have been focused on solving analytic inverse kinematics and for example in [7] by using geometric-space theory, relations between the Angles of robots' space rotation with space vectors' value is analyzed. In [8] there is an overview of the most important closed loop approaches and their features are analyzed fromthe Convergence, numerical error, perspectives and other cases.

Neural networks are used inverted kinematics field in accordance with distributed calculations and parallel, linear and nonlinear mapping capabilities. At [9] MLP1multi-layered proton neural network is used to determine the angle of the robots' joints with three-degree freedom with a work barrier. In [10]two MLP neural Radial Basis Function network have been used for robot with six-degree freedom. At [11] real robot is divided into two virtual robots. The joints variables are determined with the proper correlation between their direct kinematic equations and solution of the resulting equation. At [12] SOMnetwork is used to find the variable of the seven-degree freedom robot joints. In [13], dual network is used .The purpose of the problem is to use a combination of norm2 with infinite joint velocity. In [14], the inverse approach of Neural network has been used to determine the inverse kinetic response of seven-degree freedom robot. In [15] a frequent method called the association machine which has a structure involving several neural networks, is the proposed for determine the inverse kinematic response

The overall structure of this paper is that in the second part, the joint path is designed the form of an optimization problem. Then, the avoidance barrier program is presented along with avoiding the physical limitation of joints. In the next section, the recursive neural network model is extracted and simulation and analysis of the results are presented. Conclusions will also be made in the end section.

2- The problem of designing the path of the joints of the robot arm

The purpose of the joints design is to determine the position of the joints in such a way that the final executor of the robot can accurately track the desired path in Cartesian space. For this purpose, the problem of design of path of the joints the joints is defined in the cost function form in the following form:

$$f = \frac{1}{2}\left[(x_d - x)^2 + (y_d - y)^2 + (z_d - z)^2 \right] \quad (1)$$

Where , $\begin{bmatrix} x & y & z \end{bmatrix}$ and $\begin{bmatrix} x_d & y_d & z_d \end{bmatrix}$ □ present the momentary position of final executor and the Cartesian ideal position. The angular positions of the joints are determined indirectly and corresponding to the minimization of the cost function.

2-1. constraint of avoiding the physical limitations of the joints

Joint's Physical constraints of PA-10 7-degree freedom arm robot are as below:

$$\theta_{i\min} \le \theta_i \le \theta_{i\max} \ ; \quad i = 1, 2, \ldots, 7 \tag{2}$$

Where $\theta_{i\min}$ and $\theta_{i\max}$ are respectively minimum and maximum values of i^{th} .In order to apply these constraints to the structure of the recurring neural network, first they must be converted into negative inequalities. So you can write:

$$\theta_{i\min} - \theta_i \le 0; \ \theta_i - \theta_{i\max} \le 0; \ \ i = 1, 2, \ldots, 7 \tag{3}$$

Given that in the PA-10 robot, each the minimum joint position equals with the negative of maximum value of the

$$\begin{cases} -\theta_i - \theta_{i\max} \le 0 \\ \theta_i - \theta_{i\max} \le 0 \end{cases} \rightarrow |\theta_i| - \theta_{i\max} \le 0; \quad i = 1, 2, \ldots, 7 \tag{4}$$

2-2 obstacle Avoidance plan

To formulate of the avoidance obstacle problem, assume that obstacle in the workspace is enclosed with a circle radius r0 and $\mathbf{n}_0 = [x_0 \ y_0 \ z_0]^T$. Also, the coordinates of beginning and end of the arm of the robot for i= 1, 2, 3 are described $\mathbf{m}_i = [x_i \ y_i \ z_i]^T$ and $\mathbf{n}_i = [x_i \ y_i \ z_i]^T$.

The coordinates of each point of i^{th} arm can be determined in a parametric manner and with the information on the Cartesian form of the start and end the joints of the arm. Suppose that $\mathbf{m}_{ni} = [x_t \quad y_t \quad z_t]^T$ represents coordinates of the desired point in i^{th} arm . The coordinates of the above point can be written as on the parametric and according to the coordinates of the start and end of the arm. In the other words

$$\mathbf{m}_{ni} = \mathbf{m}_i + \alpha \mathbf{m}_i \mathbf{n}_i \ ; \ i = 1, 2, 3 \tag{5}$$

In which i is its arm number and α is a Parameter dependent on arm coordinates $0 \le \alpha \le 1$. By calculating the distance of desired point $\mathbf{m}_{ni}$ to avoidance of center of $\mathbf{n}_0$ the **dist** analytic expression which represents the distance between the arm and the obstacle, is obtained as a function of the parameter α as follows:

$$\begin{aligned} dist &= (\mathbf{m}_{ni} - \mathbf{n}_0)^T (\mathbf{m}_{ni} - \mathbf{n}_0) \\ &= (\mathbf{m}_i + \alpha \mathbf{m}_i \mathbf{n}_i - \mathbf{n}_0)^T (\mathbf{m}_i + \alpha \mathbf{m}_i \mathbf{n}_i - \mathbf{n}_0) \end{aligned} \tag{6}$$

To find the optimal parameter of α in a way that the **dist** distance is minimized, is derivate from (6) to α in the above expression:

$$\alpha (\mathbf{m}_i \mathbf{n}_i)^T (\mathbf{m}_i \mathbf{n}_i) - (\mathbf{m}_i \mathbf{n}_0)^T (\mathbf{m}_i \mathbf{n}_i) = 0 \tag{7}$$

By specifying the optimal parameter value, the coordinates of the critical point $\mathbf{m}^*_{ni}$ will be estimated as follows:

$$\mathbf{m}^*_{ni} = \mathbf{m}_i + \left(\frac{(\mathbf{m}_i \mathbf{n}_0)^T (\mathbf{m}_i \mathbf{n}_i)}{(\mathbf{m}_i \mathbf{n}_i)^T (\mathbf{m}_i \mathbf{n}_i)} \right) \mathbf{m}_i \mathbf{n}_i \ , i = 1, 2, 3 \tag{8}$$

As a result, minimum distance i^{th} arm to the center of the barrier is equal to:

$$\left\| \mathbf{m}^*_{ni} \mathbf{n}_0 \right\| = \frac{\left\| \mathbf{m}_i \mathbf{n}_0 \right\| \left\| \mathbf{m}_i \mathbf{n}_i \right\| - ((\mathbf{m}_i \mathbf{n}_0)^T (\mathbf{m}_i \mathbf{n}_i))^2}{\left\| \mathbf{m}_i \mathbf{n}_i \right\|} \tag{9}$$

To avoid any accident between the arm and the obstacle, a security zone is considered around the enclosing sphere obstacle with a radius d_s. The distance between the critical point and the center of the obstacle must be greater than the total radius of the sphere and the security zone. On the other hand:

$$r_0 + d_s - \left\| \mathbf{m}^*_{ni} - \mathbf{n}_0 \right\| \le 0;\ i = 1,2,3 \tag{10}$$

Therefore, designing the path of the joints of the robot is described in the form of the Optimization problem as follows:

$$\begin{aligned} &\min_{\boldsymbol{\theta}} \frac{1}{2}\left[(x_d - x)^2 + (y_d - y)^2 + (z_d - z)^2\right] \\ &\text{s.t. } \begin{bmatrix} |\theta_i| - \theta_{1\max} \\ r_0 + d_s - \left\| \mathbf{m}^*_{nj} - \mathbf{n}_0 \right\| \end{bmatrix} \le 0,\ i = 1,...,7,\ j = 1,2,3 \end{aligned} \tag{11}$$

3- Recursive Neural network model

Based on the principles of network design for a Convex nonlinear optimization problem in [16] recursive neural network model is extracted and is developed to find the angular position of the joints of the arm robot. Consider the below nonlinear convex optimization problem:

$$\begin{aligned} &\min \quad f(\mathbf{x}) \\ &\text{subject to } \ \mathbf{g}(\mathbf{x}) \le 0,\ \ \mathbf{x} \ge 0 \end{aligned} \tag{12}$$

Where $\mathbf{x} \in R^n$, $f : R^n \to R$ and $\mathbf{g}(\mathbf{x}) = [g_1(\mathbf{x}) \ \cdots \ g_m(\mathbf{x})]^T$ are optimization vector, cost function and constraint function Respectively. optimization conditions KKT[3] Is defined for Optimization problem [12] as follows:

$$\begin{aligned} &\mathbf{y} \ge 0,\ \ \mathbf{g}(\mathbf{x}) \le 0,\ \ \mathbf{x} \ge 0 \\ &\nabla f(\mathbf{x}) + \nabla \mathbf{g}(\mathbf{x})\mathbf{y} \ge 0,\ \ \mathbf{y}^T \mathbf{g}(\mathbf{x}) = 0 \end{aligned} \tag{13}$$

In which, $\nabla f(\mathbf{x}), \nabla \mathbf{g}(\mathbf{x})$ represent cost function gradients and constraints gradient. Using the image optimization technique, we can find the relationship between the solution of the convex nonlinear optimization problem and the solution of an image problem as follows:

$$\begin{cases} \left(\mathbf{x} - \alpha\left(\nabla f(\mathbf{x}) + \nabla \mathbf{g}(\mathbf{x})\mathbf{y}\right)\right)^+ - \mathbf{x} = 0 \\ \left(\mathbf{y} + \alpha\, \mathbf{g}(\mathbf{x})\right)^+ - \mathbf{y} = 0 \end{cases} \tag{14}$$

In which, $[x_i]^+ = \max\{0, x_i\}$, $[y_i]^+ = \max\{0, y_i\}$ and α is a positive integer. In accordance with the zeinee dot's theorem (4), if $\mathbf{x}^*$ represents the optimal response and $\phi(\mathbf{x}) = f(\mathbf{x}) + (\mathbf{y}^*)^T \mathbf{g}(\mathbf{x})$, this can be written:

$$\phi(\mathbf{x}) \ge \phi(\mathbf{x}^*),\ \ \forall \mathbf{x} \ge 0 \tag{15}$$

It can be seen from the above relation

$$\left(\mathbf{x} - \mathbf{x}^*\right)^T \nabla \phi(\mathbf{x}^*) \ge,\ \ \forall \mathbf{x} \ge 0 \tag{16}$$

Since $\phi(\mathbf{x})$ is a convex function. Therefore, $\mathbf{x}^*$ is a response from nonlinear optimization problem(12), if and only if there is an optimal point of $\mathbf{y}^* \in R^m_+$

$$\begin{cases} \left(\mathbf{x} - \mathbf{x}^*\right)^T \left(\nabla f(\mathbf{x}^*) + \nabla \mathbf{g}(\mathbf{x}^*)\mathbf{y}^*\right) \ge 0, & \forall \mathbf{x} \ge 0 \\ \left(\mathbf{y} - \mathbf{y}^*\right)^T \left(-\mathbf{g}(\mathbf{x}^*)\right) \ge 0, & \forall \mathbf{y} \ge 0 \end{cases} \tag{17}$$

Based on the image formulation in (17), neural network model is for solving the convex nonlinear optimization problem as follows

$$\frac{d}{dt}\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix}=\begin{pmatrix}\left(\mathbf{x}-\alpha\left(\nabla f(\mathbf{x})+\nabla\,\mathbf{g}(\mathbf{x})\mathbf{y}\right)\right)^{+}-\mathbf{x}\\ \left(\mathbf{y}+\alpha\,\mathbf{g}(\mathbf{x})\right)^{+}-\mathbf{y}\end{pmatrix} \qquad (18)$$

4- Results and discussion

To evaluate the performance of the proposed algorithm to determine the degree of joining angle, simulations are done on PA-10 seven-degree freedom robot. Physical limitation of joints of this robot is provided on table10.

Table11. PA-10 Position Range and Speed Arm Robot – 1

Speed of joints per radians per second	Range of joints in degrees	Joint No
± 1	± 177	1
± 1	± 91	2
± 2	± 174	3
± 2	± 137	4
$\pm 2\pi$	± 255	5
$\pm 2\pi$	± 165	6
$\pm 2\pi$	± 360	7

First, consider the circular path to the center (0.6, 0.0, 0.0) and radius 0.5^{m}. The obstacle in the workspace also has a center (0.6, 0.0, 0.0) and radius 0.5^{m} . The radius of security zone is considered 0.01^{m} . As shown in Figure 1, open robot arms have avoided obstacles. By considering fig.2 the position of the arm joints are also located in the allowed range. The final error tracking is very small in final executor. In other words, track tracing has been carried out with great precision. In the following, a rectangular path is taken, latitude point starts -0.2 and ends at 0.2. Also, the altitude of the route is 0.6. there is an obstacle with a (0.3,0.1,0.1) and a radius of 0.1^{m} . For the rectangular path, also joint movementis very far for tracking the optimal routes. (fig.4). Also, the fig.5 Shows that the physical limitations of the joints are in the range of the optimal shape of the arm robot. The tracking error of the executor is also negligible according to Figure 6.

5- Conclusion

In this paper, the design of the movement direction of the joints of the arm of the PA-10 seven-degree freedom robot was formulated with the based on a recursive neural network, and simulations and results analysis were presented. In this regard, the avoidance constraints of joints physical limitation and obstacles avoidance in the robot's work space were presented in an analytical manner. Then, the neural network structure was extracted to solve the optimized optimization problem based on optimality conditions KKT and its relation with image theory. According to, the results presented for two circular and rectangular paths, in addition to tracking is done with desirable precision, the angular position of the joints is within its range and the arms avoid collision with the obstacle.

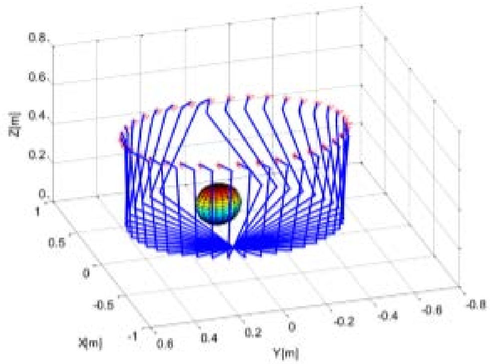

Fig.1. Circular tracking using recursive neural network

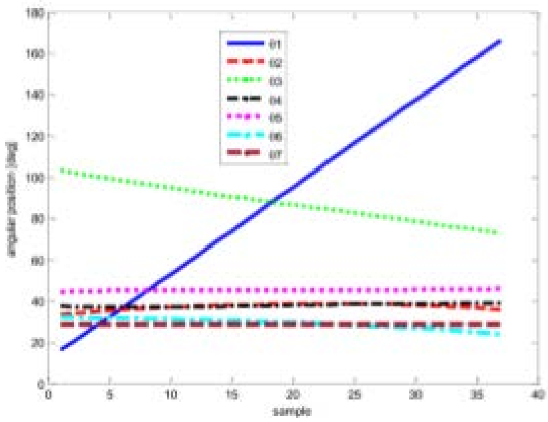

Fig.2. Angle position of the joints corresponding to the circular path

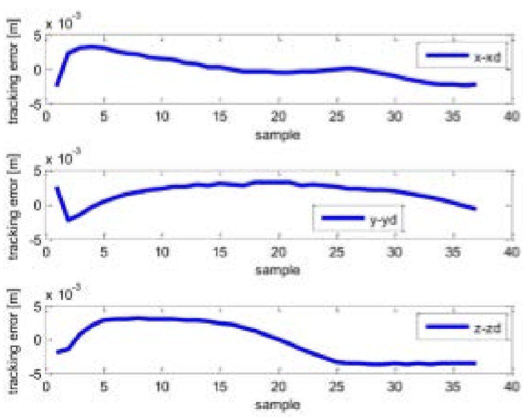

Fig.3. tracking Error the final executor corresponding to the circular path

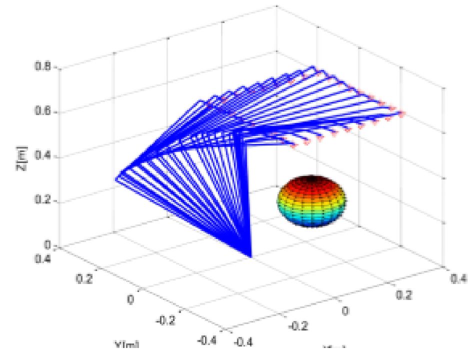

Fig.4. Rectangular path tracking using recursive neural network

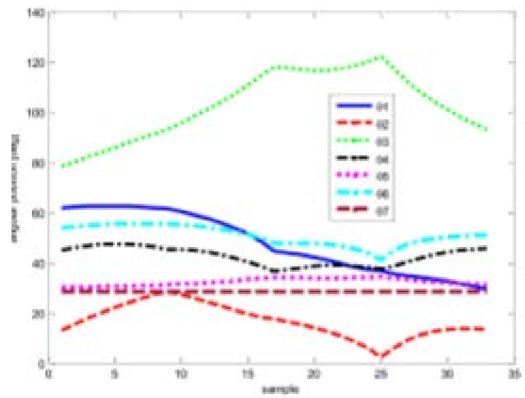

Fig.5. Angle position of the joints corresponding to the rectangular path

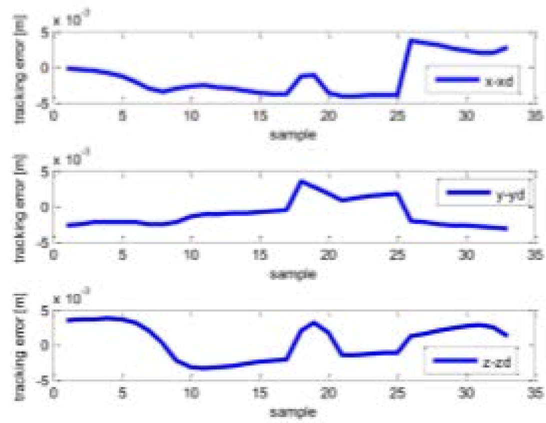

Fig.6.tracking Error the final executor corresponding to the rectangular path

References

[1] Benzaoui M., Redundant robot manipulator control with obstacle avoidance using extended jacobian method, 18th Mediterranean Conf. Cont. & Auto.(MED), Morocco, Marrkech, 371-376(2010).

[2] Wang J., Lee Y., Zhao X., Inverse kinematics and control of a 7-DOF redundant manipulator based on the closed-loop algorithm, Advan. Robo. Syst, 7: 1-10(2011).

[3] Phuoc L., et al., Damped least square based genetic algorithm with Gaussian distribution of damping factor for singularity-robust inverse kinematics, J. Mech. Scie. & Tech, 22: 1330-1338(2008).

[4] Aristidou A., Lasenby J., FABRIK: A fast, iterative solver for the Inverse Kinematics problem, Graph. Models, 73: 243-260(2011).

[5] Yahya S., Moghavvemi M., Mohamed M., Geometrical approach of planar hyper-redundant manipulators: Inverse kinematics, path planning and workspace, Simul. Model. Prac. & Theo, 19: 406-422(2011).

[6] Kucuk S., Bingul Z., Inverse kinematics solutions for industrial robot manipulators with offset wrists, Appl. Math. Model, 38: 1983-1999(2014).

[7] Wei Y., Jian S., He S., Wang Z., General approach for inverse kinematics of nR robots, Mech. & Mach. Theo, 75: 97-106(2014).

[8] Colom A., Torras C., Closed-Loop Inverse Kinematics for Redundant Robots: Comparative Assessment and Two Enhancements, IEEE/ASME Trans. Mecha, 20: 944-955(2015).

[9] Bhattacharjee T., Bhattacharjee A., A study of neural network based inverse kinematics solution for a planar three joint robot with obstacle avoidance, Ass. Univer. J. Scie. & Tech, 5: 1-7(2010).

[10] Chiddarwar S.S., Ramesh Babu N., Comparison of RBF and MLP neural networks to solve inverse kinematic problem for 6R serial robot by a fusion approach, Engin. App. Artif. Intell, 23: 1083-1092(2010).

[11] Angulo V.R., Torras C., Learning inverse kinematics: reduced sampling through decomposition into virtual robots, IEEE Trans. Syst., Man, & Cyber., Part B: Cyber, 38: 1571-1577(2008).

[12] Kumar S., Patel N., Behera L., Visual motor control of a 7 DOF robot manipulator using function decomposition and sub-clustering in configuration space, Neur. Proc. Lett, 28: 17-33(2008).

[13] Zhang Y., Wang J., Xu Y., A dual neural network for bi-criteria kinematic control of redundant manipulators, IEEE Trans. Robot. & Autom, 18: 923-931(2002).

[14] Kumar S., Behara L., McGinnity, T., Kinematic control of a redundant manipulator using an inverse-forward adaptive scheme with a KSOM based hint generator, Robot. & Auton. Syst, 58: 622-633(2010).

[15] Koker R., Cakar T., Sari Y., A neural-network committee machine approach to the inverse kinematics problem solution of robotic manipulators, Engin. with Comp, 30: 641-649(2014).

[16] Xia Y., Wang J., A recurrent neural network for nonlinear convex optimization subject to nonlinear inequality constraints, IEEE Trans. Circ. & Sys. I, Regular Papers, 51: 1385-1394(2004).