Simulation design of trajectory planning robot manipulator

Wahyu S. Pambudi¹, Enggar Alfianto², Andy Rachman³, Dian Puspita Hapsari⁴
¹ Department Electrical Engineering, Institut Teknologi Adhi Tama Surabaya, Jl. Arief Rahman Hakim 100 Surabaya, Indonesia
² Department Computer System, Institut Teknologi Adhi Tama Surabaya, Jl. Arief Rahman Hakim 100 Surabaya, Indonesia
³,⁴ Department Informatics Engineering, Institut Teknologi Adhi Tama Surabaya, Jl. Arief Rahman Hakim 100 Surabaya, Indonesia

ABSTRACT

Robots can be mathematically modeled with computer programs where the results can be displayed visually, so it can be used to determine the input, gain, attenuate and error parameters of the control system. In addition to the robot motion control system, to achieve the target points should need a research to get the best trajectory, so the movement of robots can be more efficient. One method that can be used to get the best path is the SOM (Self Organizing Maps) neural network. This research proposes the usage of SOM in combination with PID and Fuzzy-PD control for finding an optimal path between source and destination. SOM Neural network process is able to guide the robot manipulator through the target points. The results presented emphasize that a satisfactory trajectory tracking precision and stability could be achieved using SOM Neural networking combination with PID and Fuzzy-PD controller. The obtained average error to reach the target point when using Fuzzy-PD=2.225% and when using PID=1.965%.

1. INTRODUCTION

Optimal manipulator robot control to achieve the desired trajectory points requires sustainable research. Optimal control in this research is the handling of objects or workpieces in specified positions, where the robot manipulator can determine the angle with a small error to achieve the targets that have been determined [1, 2]. The addition of fuzzy control system to robot manipulatoris able to move the robot to reach one target point but it still have an error. During the process of movement of the robot manipulator by using forward and inverse kinematic, so it can be transformed from the angle of the arm to the coordinate position in two-dimensional plane or vice versa [2-4]. In the realization of robot manipulators to solve a problem or just applyinga method, it requires a number of hardware with expensive cost. Simulation is one of the solutions to represent robot performance research without using hardware. Problems about robots can be modeled mathematically with computer programs where the results can be displayed visually, so it can be used to determine the input, gain, attenuate and error parameters of the control system. Manipulator robot simulation can be included force element by using dynamic equation, so the effect of load change has influence the movement of robot manipulator [1-5].

The use of this simulation can help researchers in the field of robot movement technology, so it can approach the problems to be solved [6, 7]. In addition to the robot motion control system, to achieve the target position and get the best trajectory/track the robots needs to be controlled, so the movement of robots
can be more efficient. One method that can be used to get the best path is using Genetic Algorithm (GA), where GA is an optimization method of a mathematical function [7]. Another method that can be used to get the best path is one of the types of neural network that usually called SOM (Self Organizing Maps) [8-11].

This paper is organized as follows, applied SOM neural network to get the best path from the point that spread randomly on the robot manipulator. This robot manipulator is also given a control using Proportional Integral Derivative (PID) and Fuzzy Proportional Derivative (Fuzzy-PD) [12-16]. The reason to combine both of SOM neural network and Fuzzy-PD control to simulate robot manipulator in this research because it is still rarely used. The advantage of SOM's neural network for trajectory planning robot arm manipulator is the learning process without guidance, unlike the neural network back propagation that should always be guided to learn trajectory planning [17-21]. This robot manipulator simulation is using Matlab Robotics Toolbox, where the simulation is not built from beginning [5, 6]. The simulation was made by using C# programming languages and the results are displayed with images in 2 dimensional fields. The purpose of this joint implementation research between SOM neural network and Fuzzy-PD control can be the basic research in the realization process later.

2. RESEARCH METHOD

2.1. SOM training algorithm

Stages for the SOM training algorithm is described below [18]:

a. Initialize all weight, learning rate parameter μ (0), set iteration value (epoch) k=0 and determine the topology of the neighbourhood function.

b. Check the conditions that meet to stop the iteration.

c. For each training value of vector xi, do steps a through c

1) Calculate the best pair of weights and input values.

\[ q(x) = \min_{i \in \Gamma_{\text{vis}}} \| x - w_i \|_2 \]  \hspace{1cm} (1)

2) Update the weighted value by using the function as in (2):

\[ w_i(k+1) = \begin{cases} w_i(k) + \mu(k) & \text{if } i \in N_q(k) \\ w_i(k) & \text{if } i \notin N_q(k) \end{cases} \]  \hspace{1cm} (2)

3) Set the learning rate parameter.

Set k=k+1, then subtract the value of learning rate and proceed to step no. 3.

d. Simulation of 3 DOF robot manipulator

Figure 1 shows the 3 DOF three-link planar physical structure, its consists of link 1 (shoulder), link 2 (elbow) and link 3 (end of effector (EE)), which in this study link 3 is designed passive. EE is the end of the robot manipulator which had a special device. This device has a function like human-hand that directly relate with the object and the environment. This simulation is using a computer program where the movement of this robot manipulator as visually. Things that need to be understood during the making of this simulation program are programming, mathematical concepts about geometry, kinematic and dynamics of robot manipulator.

The control system in this research is using proportional integral derivative (PID) and Fuzzy-PD as in Figure 2, where for Fuzzy-PD input variables are using error θ₁, error θ₂, Δerror θ₁ and Δerror θ₂ while the output variables are θ₀₁ and θ₀₂ [12-16]. The distribution of membership function for each input error and Δerror is represented in Figure 3(a) and 3(b), for the output as in Figure 4, while for rule evaluation is described in Table 1. This tho output is included in dynamic equation, so the kinetic and potential energy can be entered. The result of this dynamic process were produce an angular acceleration of θ₁ and θ₂. Both angular accelerations are incorporated in the kinematic forward and the integral of this kinematic forward is used to obtain the angle of each arm. Center of Area (COA) is chosen as defuzzification method, which calculates the area center of the fuzzy set membership function curve surrounded by the horizontal coordinate.
Figure 1. 3 DOF three-link planar physical structure [2]

Figure 2. Block diagram control system on simulation of 3 DOF robot manipulator.

Figure 3. Fuzzification of input variable (a) error input (b) delta error input.
Simulation design of trajectory planning robot manipulator (Wahyu S. Pambudi)
3. RESULTS AND ANALYSIS

Simulation was performed in Visual Studio C# software to investigate the performance of Fuzzy-PD and PID controller for finding an optimal path between source and destination. The simulation results are demonstrated in this section. Travelling salesman problem (TSP) implementation program with SOM on simulation of robot manipulator is using Fuzzy-PD control system and follows the block diagram in Figure 2. The used scale for this simulation is 1 pixel=1 cm, while speed in robot motion unit is 10 cm/sec. The gravity value is equal to 9.8 cm/s^2, shoulder length of 0.75 m and elbow of 0.85 m. At the first run, the point distributed randomly around the EE, the number of points is 10, the plot results from this point as in Figure 7(a) when it was executed by using SOM algorithm then the result as shown in Figure 7(b)-(d). Based on Figure 7(b)-(d) it can be described that during SOM process, the network were formed following the specified target point. This network is created as a trajectory robot manipulator. The next testing stage is to provide new target points, the result can be seen in Figure 8(a)-(h), where the red (large) point is the target and the green dot (small) is the SOM process. SOM was used in the study because it was better than back propagation in the previous studies [17], this is due to the learning process of SOM that doesn’t need to be guided to make trajectory. Trajectory Planning Robot Manipulator simulation test in this research is using 20 target points where in the previous research is less than 20 target points [19].

![Figure 6. Manipulator robot simulation program](image)

![Figure 7(a)-(d)](image)

(a) the initial process  (b) SOM process on the 57th epoch

![Figure 8(a)-(h)](image)
Simulation design of trajectory planning robot manipulator (Wahyu S. Pambudi)
Based on Figure 8(a)-(h), when epoch is less than 246 SOM the process has not achieved the target points optimally. After entering epoch above 354, SOM process has achieved the target points and kept stable until 1999th epoch. Result of trajectory planning can be seen in Figure 9, where the result of trajectory robot manipulator is able to follow trajectory that formed by neural network. Figure 10 and Figure 11, represents a change in angle $q_1$ and $q_2$ during the arm manipulator process along the planned targets using the Fuzzy-PD control, while Figure 11 when using PID controls. The testing process of arm manipulator is using random points between 9 to 182 on x-axis and 52-186 on y-axis. The result of this arm manipulator simulation test has an average error of 3.57% on x-axis and 0.88% on y-axis for PID control, whereas The Fuzzy-PD has an average error of 2.62% on x-axis and 1.31% in y-axis. The average error on x-axis and y-axis for the Fuzzy-PD control is 2.225%, when using the PID control the error is 1.965%. This result is better than previous studies that have an average error of 3% [20]. The complete result during the process of movement of this manipulator arm is represented in Table 2 and Table 3.
### Table 2. Target vs SOM results for (x,y) coordinates using fuzzy-PD

| Target x | SOM result x | Error (%) | Target y | SOM result y | Error (%) |
|----------|--------------|-----------|----------|--------------|-----------|
| 175      | 175          | 0.00%     | 103      | 103          | 0.00%     |
| 182      | 181          | 0.55%     | 124      | 124          | 0.00%     |
| 167      | 169          | 1.20%     | 142      | 140          | 1.41%     |
| 153      | 154          | 0.65%     | 163      | 161          | 1.23%     |
| 147      | 148          | 0.68%     | 167      | 166          | 0.60%     |
| 118      | 120          | 1.69%     | 181      | 180          | 0.55%     |
| 112      | 112          | 0.00%     | 185      | 183          | 1.08%     |
| 96       | 94           | 2.08%     | 185      | 187          | 1.08%     |
| 26       | 27           | 3.85%     | 184      | 184          | 0.00%     |
| 10       | 12           | 20.00%    | 186      | 185          | 0.54%     |
| 9        | 7            | 22.22%    | 142      | 143          | 0.70%     |
| 33       | 31           | 6.06%     | 148      | 146          | 1.35%     |
| 55       | 54           | 1.82%     | 149      | 149          | 0.00%     |
| 62       | 61           | 1.61%     | 123      | 125          | 1.63%     |
| 66       | 66           | 0.00%     | 101      | 102          | 0.99%     |
| 82       | 80           | 2.44%     | 97       | 95           | 2.06%     |
| 96       | 94           | 2.08%     | 93       | 95           | 2.15%     |
| 116      | 114          | 1.72%     | 75       | 75           | 0.00%     |
| 137      | 135          | 1.46%     | 54       | 54           | 0.00%     |
| 170      | 168          | 1.18%     | 86       | 84           | 2.33%     |
| **Average** |            | **3.57%** | **Average** |            | **0.88%** |

### Table 3. Target vs SOM results for (x,y) coordinates using PID

| Target x | SOM result x | Error (%) | Target y | SOM result y | Error (%) |
|----------|--------------|-----------|----------|--------------|-----------|
| 152      | 151          | 0.66%     | 69       | 67           | 2.90%     |
| 135      | 136          | 0.74%     | 52       | 54           | 3.85%     |
| 112      | 111          | 0.89%     | 76       | 74           | 2.63%     |
| 96       | 96           | 0.00%     | 93       | 91           | 2.15%     |
| 83       | 84           | 1.20%     | 96       | 94           | 2.08%     |
| 66       | 68           | 3.03%     | 101      | 100          | 0.99%     |
| 39       | 40           | 2.56%     | 119      | 117          | 1.68%     |
| 9        | 8            | 11.11%    | 138      | 136          | 1.45%     |
| 10       | 9            | 10.00%    | 160      | 158          | 1.25%     |
| 11       | 12           | 9.09%     | 187      | 186          | 0.53%     |
| 35       | 34           | 2.86%     | 167      | 166          | 0.60%     |
| 55       | 53           | 3.64%     | 151      | 151          | 0.00%     |
| 79       | 79           | 0.00%     | 165      | 163          | 1.21%     |
| 112      | 111          | 0.89%     | 184      | 182          | 1.09%     |
| 133      | 131          | 1.50%     | 173      | 174          | 0.58%     |
| 153      | 151          | 1.31%     | 164      | 164          | 0.00%     |
| 164      | 165          | 0.61%     | 147      | 149          | 1.36%     |
| 181      | 181          | 0.00%     | 124      | 125          | 0.81%     |
| 175      | 177          | 1.14%     | 103      | 102          | 0.97%     |
| 171      | 173          | 1.17%     | 87       | 87           | 0.00%     |
| **Average** |            | **2.62%** | **Average** |            | **1.31%** |

Simulation design of trajectory planning robot manipulator (Wahyu S. Pambudi)
Figure 10. The angle changes of the arm manipulator using Fuzzy-PD

Figure 11. The angle changes of the arm manipulator using PID

The novel of this research is the use of Fuzzy-PD on SOM application. In most cases, SOM applications are using heuristic-based methods such as Back Propagation Neural Network and Extreme Learning Machine [21] to model the movement of robotic arms, this heuristic method has limitations to the generated response if the used dataset and sensor readings are not suitable with the field conditions. Therefore this research concludes that kinematic inverse modeling will be more efficient by using an error-based method through the use of PID and with a little additional logic rule (Fuzzy-PD). The result shows the error-based method by using either PID or Fuzzy-PD has been able to produce a smaller error deviation than the ANN and Genetic Algorithm methods [20].

4. CONCLUSION

Trajectory planning on robot manipulator by using SOM neural network and Fuzzy logic control is able to guide robot manipulator to reach the target points. The obtained results from this simulation to reach the target point is when using the Fuzzy-PD control, errors are in x coordinates is 3.57% and y coordinates is 0.88% or the average of both is 2.225%. When using PID, errors are in x coordinates is 2.62% and y coordinates is 1.31% or the average is 1.965%.

REFERENCES

[1] HalaBezine, NabilDerbel, Adel M Alimi. Fuzzy control of robot manipulators: some issues on design and rule base size reduction. Science Direct Engineering Applications of Artificial Intelligence. 2002; 15(5): 401–416.
[2] W S Pambudi, N M A Sumanang. Implementation of FuzzyPD to Determine Object Position on Simulation Model of Robot Arm Manipulator 3 DOF (Degree of Freedom) in 2D. Mikrotek Journal, Universitas Trunojoyo Madura. 2014; 1(2): 19-29 (original version in Bahasa Indonesia).
Simulation design of trajectory planning robot manipulator (Wahyu S. Pambudi)

[3] Jinho K, Andrew S. Lee, Kevin C, Brian S, S A Gadsden. M Al-Shabi. Dynamic Modeling and Motion Control of a Three-Link Robotic Manipulator. The 2017 International Conference on Artificial Life and Robotics (ICAROB 2017). Miyazaki. 2017; P-380 – P-381.

[4] I. Pajak, "Trajectory planning of cooperative mobile manipulators subject to control constraints," 2017 11th International Workshop on Robot Motion and Control (RoMoCo), Wasowo, 2017, pp. 123-128.

[5] Adri’a Colom e. Smooth Inverse Kinematics Algorithms for Serial Redundant Robots. Master Thesis. Barcelona: Institute de Robòtica Informàtica Industrial (IRI). 2011.

[6] Li Jingwei, Tong Yifei, Wu Shaoqeng, Tan Qingmeng, Li Dongbo. Welding Robot Kinematics Analysis and Trajectory Planning. TELKOMNIKA (Telecommunication, Computing, Electronics and Control) Indonesian Journal of Electrical Engineering, 2016; 14(2A): 92-100.

[7] V. K. Banga, Y. Singh, and R. Kumar. Simulation of Robotic Arm using Genetic Algorithm & AHP. World Academy of Science, Engineering and Technology 25 2007; 25: 95-101.

[8] Yanping Bai, Wendong Zhang, Zhen Jin. An new self-organizing maps strategy for solving the traveling salesman problem. Science Direct Elsevier, Chaos, Solitons and Fractals 28 (2006). 2006; 1082–1089.