ABSTRACT: This paper presents a simulation of a 6 degrees-of-freedom (6R) articulated robot arm using backpropagation neural network to solve the problem regarding inverse kinematics for the industrial articulated robot. The Denavit – Hartenberg model is used to analyze the robot arm movement. Next, the forward kinematics is used to identify the relationships for each joint of the robot arm and to determine various parameters for learning system of random neural network for 5,000 data points. The simulation results show that the robot arm can move to target positions with precision, and the average error for the entire 6 joints is at approximately 4.03 degrees.

Keywords: neural network, articulated robot arm, inverse kinematics

1. INTRODUCTION

Nowadays, technologies have greatly influenced on our lives. Many factories in the industrial sphere, such as car assembly industry, use robot arms as their important tools for manufacturing. In order to control the movement of robot arms according to target positions, the robot arm movement needs to be designed for stability and high precision. From the past study for the movement of robot arm, it found those forward and inverse kinematics are used to design the movement system, as shown in Figure 1. These approaches have caused some difficulties in executive of arm movement. To allow the robot arm moving more freely and efficiently, numerous techniques, such as fuzzy logic [2-3], are applied to boost operational efficiency. In this regard, neural network (NN) is an alternative approach chosen for reducing the complexity of inverse kinematics [4-8].

Figure 1 The schematic representation of forward and inverse kinematics [9].

Many research papers have been reported in various types of robot arms such as 2-joint robot arms, 3-joint robot arms, and even 6-joint robot arms, which are articulated by different systems and have different advantages and disadvantages. For example, 2-joint robot arms [9], 3-joint robot arms [10], and 4-joint robot arms [11] cannot grip objects located in remote and difficult-to-access locations. However, they are popular for simple tasks because of their easy system. In contrast, 5-joint robot arms [12-13] and 6-joint robot arms [14] have flexible patterns, depending mainly on the nature of tasks, such as automobile industrial factories, etc. The robot arms with multiple joints can work in a narrow space and complicated environment better than 2-joint or 3-joint robot arms. The paper [14] has presented the manipulator robot arm with offset wrist using neural network for moving it to accurately target positions. However, the robot arm cannot reach to objects in a narrow and complicated space, and it also has a short range of gripping. In addition, the results show that the robot arm can move to the target positions, but there are still high sum of errors to cause the robot arm to a lack of precision.

In this paper, the simulation of a 6R articulated robot arm is presented through the use of backpropagation neural network to solve the problem regarding inverse kinematics utilized for industrial factories, because the robot arm will be similar to human arm. It can move freely and access a narrow and multi-leveled space. There are 5,000 data points inputted to NN with cross validation for testing movement of the robot arm in its learning. The simulation results show that the robot arm can move to the target positions and has more precision.

2. Proposed Method

2.1. Denavit – hartenberg model

D – H or Denavit – hartenberg [1] is a standard model used in determining variables of robot arm for operational analysis of the robot arm. This model is used to find value of the variants $a_i$, $d_i$, $\theta_i$, $\alpha_i$, $\beta_i$, which are lengths of each joint, distance between joints, close angle and each angle of each joint, respectively, in order to replace the value in the matrix to find relationship between each joint adherent with the robot arm. In the matrix model, degree of movement between frame $i$ and frame $i-1$ is as follow:

$$ A_{i-1}^{-1}(\theta_i) = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\beta_i \\ s\theta_i c\beta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\beta_i \\ 0 & s\alpha_i & c\alpha_i & 0 \end{bmatrix}$$

From equation (1), to determine $c\theta_i = \cos \theta_i$, $s\theta_i = \sin \theta_i$, $c\alpha_i = \cos \alpha_i$ and $s\alpha_i = \sin \alpha_i$ by each value in the matrix will
be replaced by the parameters derived from the D-H model which is found from the robot arm analyzed in Fig. 2. The robot arm with 6 degrees is dependent. We can analyze the relationship between each joint, as shown in Table 1.

![Figure 2](image)

**Figure 2** The 6R articulated robot arm analyzed by D – H.

<table>
<thead>
<tr>
<th>Link</th>
<th>a_i</th>
<th>d_i</th>
<th>α_i</th>
<th>θ_i</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link 1</td>
<td>0</td>
<td>0</td>
<td>90</td>
<td>θ_1</td>
</tr>
<tr>
<td>Link 2</td>
<td>a_2</td>
<td>0</td>
<td>0</td>
<td>θ_2</td>
</tr>
<tr>
<td>Link 3</td>
<td>a_3</td>
<td>0</td>
<td>0</td>
<td>θ_3</td>
</tr>
<tr>
<td>Link 4</td>
<td>a_4</td>
<td>0</td>
<td>-90</td>
<td>θ_4</td>
</tr>
<tr>
<td>Link 5</td>
<td>0</td>
<td>0</td>
<td>90</td>
<td>θ_5</td>
</tr>
<tr>
<td>Link 6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>θ_6</td>
</tr>
</tbody>
</table>

According to the analysis in Table 1, various values can be replaced in the transformation matrix $A_1^0$ to $A_6^0$, as shown in Fig. 3. For example, $A_2^0$ represents a movement from frame 0 to frame 1 (designating $C_i$ as $\cos\theta_i$ and $S_i$ as $\sin\theta_i$ etc).

![Figure 3](image)

**Figure 3** Transformation matrix of the robot arm.

### 2.2. Forward kinematics

Calculating the movement of robot arm to the target point using joint angles is called as forward kinematics analysis. These equations are created by transformation matrixes, as shown in Fig. 3, in which the configurations of each model of robot arms will has different equations in following the patterns to particular models of such robot arms. In this research, the equations of the robot arms are 6-joint robot arms with the model equation. Combining each equation and each joint together as follows:

$$T_6^0 = A_6^0 A_5^1 A_4^2 A_3^3 A_2^4 A_1^5$$

which $(n, o, a)$ are rotation elements of transformation matrix and $p_x, p_y, p_z$ are elements of position vector used in analyzing the movement of robot arm toward the target points.

### 2.3. Model of Neural Network

Artificial neutral network (ANN) is an attempt to imitate human brain functioning through learning mechanism utilizing the various past samplers in training. The ANN can be applied to the problems with no algorithmic solutions or with too complex algorithmic solutions to be found. Owing to the learning ability from these samplers, it causes the ANN to be more flexible and powerful than the parametric approaches [16]. The component of the ANN is comprised of various nodes which interlock each other. Each node $(n)$ receives the value of input and weight for analyzing with the following equation:

$$n = (\sum_{i=1}^{P} w_i x_i) + b$$

where $P$ is a total node number, $w_i$ is the weight value of input, $x_i$ is input which taken into and $b$ is bias value added into the sum amount of that node. Then the sum amount of input which its weight value is already adjusted will be assessed through activation function to achieve the output and then transmit to the node in the next layer. The equation model is as follows:

$$y = f(n) = f\left(\sum_{i=1}^{P} w_i x_i + b\right)$$

when $y$ is output, $n$ is the sum of the weighted inputs and $f$ is activation function. Normally, the activation function has various types, both linear and non-linear. We can select types for appropriate to its function such as sigmoid function, threshold function, hyperbolic tangent function etc.
The ANN has been instructed for capability to analyze outcome by adjusted/trained the weight. During the operating process in each nodes and go through such appropriate activation function in order to make the required output to be consistent to the output acquired from training as much as possible. The less error has occurred by changing the value of the weights and biases. The ANN will be divided into three parts which are an input layer, a hidden layer(s) and an output layer. The input layer is responsible for receiving data through the process of hidden layer(s) and then showing the outcome through output layer. The outcome from each node of each layer will be sent to the node of the next layer respectively. The amount of node input and node output is equal to the amount of input parameter and output parameters which we required.

2.4. The Structure of Back Propagation Neural Network

In this paper, the 12 parameters used in this experiment are derived from the process of forward kinematic. These 12 parameters are divided into two parts as follows. Nine parameters are rotation matrix and three parameters are position vector. From training data, we will get 6 outputs which are various angles of the robot arms that will move the imitated robot arms to coordinated target. The training data derived from the equation (2) will have 5000 data. This data is chosen by random. Seventy percents (70%) are the data used for training while another thirty percents (30%) is used for testing and validation.

Figure 4 A proposed model of neural network.

After the trial and errors for many times, we finally get the multilayer model of backpropagation neural network which is comprised of 1 hidden layer 40 nodes and an output layer 6 nodes where with the input layer is equal to the amount of parameters as mentioned before (12 parameters or 12 nodes), as shown in Fig. 4. The nodes between the input and hidden layer will have activation function as sigmoid function. But the output layer will have the activation function as linear function. Therefore, the answer will be above the value -1 to +1. For the training process, the weight value will be adjusted by Levenberg-Marquardt algorithm due to fastest training algorithm for the networks of moderate sizes.

3. Simulation Results

From the simulation results, we use randomly 5,000 data points inputted to NN with cross validation for testing movement of the robot arm in its learning. The simulation results show that the robot arm can move to target positions with precision, and the average error for the entire 6 joints is at approximately 4.03 degrees, as shown in Figure 5.

Table 1 Comparisons of the 6R robot arms.

<table>
<thead>
<tr>
<th>Titles</th>
<th>This paper</th>
<th>[14]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Nodes</td>
<td>5,000</td>
<td>8,000</td>
</tr>
<tr>
<td>Hidden layers/Number of Nodes</td>
<td>1 / 40</td>
<td>2 / 12,24</td>
</tr>
<tr>
<td>Sum of Errors</td>
<td>4.03 degrees</td>
<td>8.78 degrees</td>
</tr>
<tr>
<td>Arm Models</td>
<td>Articulated Robot</td>
<td>Manipulator Robot with offset wrist</td>
</tr>
<tr>
<td>Reach to Areas</td>
<td>Complex</td>
<td>Simple</td>
</tr>
<tr>
<td>Distances of Movements</td>
<td>Long</td>
<td>Short</td>
</tr>
</tbody>
</table>

Table 1 shows comparisons of the 6R robot arms. It can be seen from Table 1 that input nodes, hidden layers, sum of errors, reach to areas and distances of movements in this paper are better than those of the paper [14].

Figure 6 shows examples of six graphs of the 6-joints robot arm for comparing the error results between the training output (dotted line) and target output (solid line).
The ANN is a tool for researchers who likely use to solve such complicated and difficult problems. In this paper, the ANN has been applied to solve problems of inverse kinematics problem of the robot arm with 6-joint robot arms using backpropagation algorithm in ANN training. Structural pattern like the inverse kinematics of 6-joint robot arms is highly complicated. Therefore, ANN method is a good alternative for this problem solving. However, the proper usage of ANN method with multilayer is not so easy. Roughly experiment is easily done. But making the outcome from the trail be accurately fit to the target’s position is very difficult due to huge neural network’s structure needs data with high resolution and the data’s width should not too large. As a result of experimental outcome, the analysis shows that motion degree prediction of robot arms is at the satisfied accuracy level while the error is at approximately 5 degrees.

REFERENCES


