All Sciences Proceedings
http://as-proceeding.com/

1st International Conference on Modern and Advanced Research
July 29-31, 2023 : Konya, Turkey

https://as-proceeding.com/index.php/icmar

Position Analysis of the Slider-Crank (R-RRT) Mechanism Using Artificial Neural Networks

Onur Denizhan[1*]

[1]Department of Electronics and Automation, Batman University, Türkiye

*(onur.denizhan@batman.edu.tr)

Abstract – The slider-crank mechanism is a common mechanical linkage that converts rotary motion into reciprocating motion. It finds wide applications in various fields, including internal combustion engines, pumps, compressors, presses, robotics, and human-powered vehicles. Due to its widespread use, several textbooks have covered its position, velocity, and acceleration analyses, with different researchers proposing various analysis solutions. Recently, artificial neural networks (ANN) have been utilized in diverse research areas, including inverse and forward kinematic analysis. However, there has not been a specific use of ANN for position analysis of slider-crank mechanisms. This study aims to addresses that gap by presenting the position analysis of the in-line type slider-crank (R-RRT) mechanism using the ANN algorithm. For this purpose, the Levenberg-Marquardt backpropagation algorithm is selected due to its advantages, such as speed, stable convergence of training error, and the combination of Gauss-Newton training algorithm and steepest descent method. To train the algorithm effectively, 50 data sets are carefully chosen and randomly split for training, validation, and testing. Moreover, an additional 200 data sets are reserved for testing the trained algorithm to evaluate its performance. This study presents the result of the neural network algorithm training, as well as the outcomes of additional testing of the trained algorithm. These results are thoroughly discussed and analyzed.

Keywords – Slider-Crank Mechanism, Position Analysis, Neural Networks, Regression, Levenberg-Marquardt Backpropagation.

I. Introduction

The slider-crank mechanism is a four-link mechanism with three revolute joints and one prismatic or sliding joint. Its purpose is to convert straight-line motion to rotary motion, as seen in a reciprocating piston engine, or to convert rotary motion to straight-line motion as seen in a reciprocating piston pump. The slider-crank mechanism comprises three primary components: a crank, a connecting rod and a slider. The crank rotates, the slider slides inside the tube, and the connection rod links these parts together.

Artificial neural network algorithms find applications in various research fields and one of the is mechanism kinematic analysis and synthesis. Some research articles have been presented in this area, a few of them are as follows: A neural network algorithm solution approach for the robotic arm inverse kinematics problem is presented by Duka [1]. In this article, inverse kinematic problem solution of the planar three-link manipulator is reported by using artificial neural network algorithm. Another inverse kinematic problem solution approach is introduced by Lu et. al. [2]. In this study, an inverse kinematic problem solution approach for general six-axis robots is presented based on multilayer perception artificial neural networks. An artificial neural networks solution for the forward kinematic is presented by Prado et. al. [3]. In this study, forward kinematics of a wearable parallel robot with semi-rigid links are solved by using neural network algorithm.

Previously, Denizhan [4] reported the application of the Levenberg-Marquardt backpropagation neural network algorithm to the four-bar linkage mechanism. The article focused on solving the two-position kinematic synthesis problem of the four-bar planar linkage mechanism using a neural network approach. In this current study, the position analysis of the in-line type slider-crank (R-RRT) mechanism utilizing the Levenberg-Marquardt backpropagation artificial neural network algorithm is investigated. To ensure the accuracy and effectiveness of the algorithm, a total of 50 data sets are employed for training, validation, and testing. Upon completing the training, validation, and testing phases of the algorithm, an additional test of the trained model is also presented. For this purpose, a different set of 200 data sets is determined to evaluate the performance of the trained algorithm.

II. Position Analysis of the Slider-Crank Mechanism

The slider-crank mechanism can be classified into two-types: in-line and offset. In the in-line type, the pivot point of the crank aligns with the axis of the linear movement, whereas in the offset type, the line of the hinged joint of the slider does not pass through the base pivot of the crank. In this study, an in-line type of the slider-crank mechanism is considered.

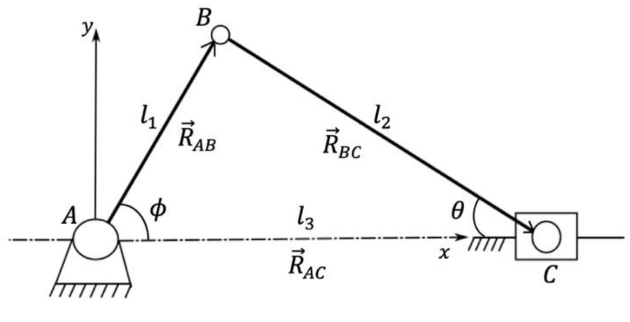

Fig. 1 In-line type slider-crank (R-RRT) mechanism

Figure 1 shows the in-line type slider-crank mechanism along with its position vectors, angles, and lengths. As shown in Fig. 1, vectors $\vec{R}_{AB}$, $\vec{R}_{BC}$ and $\vec{R}_{AC}$ represent the position vectors of Links [AB], [BC] and [AC], respectively. The angles ϕ and θ correspond to the Links [AB] and [BC] angles relative to the horizontal axis, respectively. The parameters l_1, l_2 and l_3 denote the lengths of Links [AB], [BC] and [AC] respectively. It is important to note that l_1 and l_2 remain constant, while l_3 changes during the motion of the mechanism.

In this study, the location of Joint A is fixed at the origin (0,0), and Link [AB] is capable of completing a full rotation. Joint C exhibits reciprocating motion along the *x*-axis during the motion of mechanism. As previously mentioned, the slider-crank mechanism is of the in-line type, resulting in Joint C having reciprocating motion solely along the *x*-axis. The position analysis of the slider-crank mechanism can be expressed using the following vector loop equation:

$$\vec{R}_{AB} + \vec{R}_{BC} - \vec{R}_{AC} = 0 \quad (1)$$

where $\vec{R}_{AB}$, $\vec{R}_{BC}$ and $\vec{R}_{AC}$ are the position vectors of the Link [AB], Link [BC] and Link [AC], respectively. The complex form of Eq. (1) can be expressed by following equation:

$$l_1 e^{i\phi} + l_2 e^{i(2\pi-\theta)} - l_3 e^{i0} = 0 \quad (2)$$

where the parameters l_1, l_2 and l_3 are lengths of the Links [AB], [BC] and [AC], respectively and the angles ϕ and θ are the Links [AB] and [BC] angles relative to the horizontal axis, respectively. In Eq. (2), there are 2 unknown parameters: l_3 and θ. After writing imaginary and real parts of the Eq. (2), this system of equations can be solved for these 2 unknown parameters. The position analysis of the slider-crank mechanism has been extensively covered in various textbooks [5-7]. Hence, there is no need to provide a detailed solution procedure for the position analysis using the vector loop method in this study.

The slider-crank mechanism is characterized by the following parameters: $l_1 = 0.5$ m, $l_2 = 1$ m and angle $\phi = 45$ deg for the mechanism initial position. The motion investigated in this study involves the Link [AB] completing one full rotation. During the position analysis using the vector loop method, the unknown parameters (l_3 and θ) are determined as the mechanism undergoes motion.

III. ARTIFICIAL NEURAL NETWORK APPROACH

In this study, a two-layer feedforward artificial neural network is designed for the position analysis of the slider-crank mechanism. The Levenberg-Marquardt backpropagation algorithm is employed for this purpose due to its speed, ability to combine the Gauss-Newton training algorithm and steepest descend method, and its capability to ensure stable convergence of training error [8-10]. Figure 2 illustrates the structure of the neural network design. In Fig. 2, b refers to the neural network bias, and w denotes the neural network weights. The algorithm is composed of a hidden layer with 100 neurons and an output layer with 2 neurons. For the hidden layer, the sigmoid activation function is used, while the output layer utilizes the linear activation function, as shown in Fig. 2.

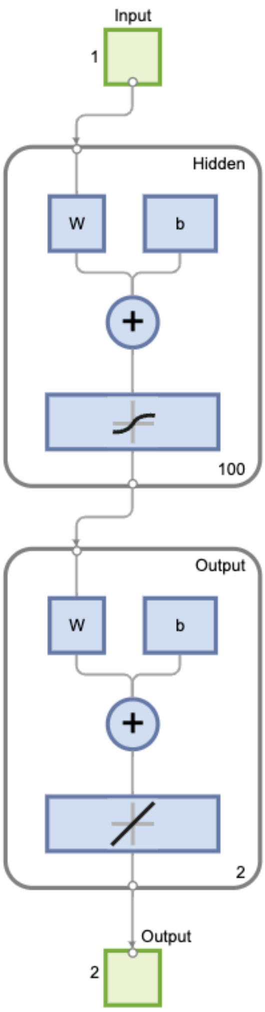

Fig. 2 Structure of the artificial feedforward neural network algorithm

In the neural network algorithm designed for this study, there is a single input, which is the angle of Link [AB] (ϕ), and two outputs, representing the position coordinate of Joint C (x_C) and the angle of Link [BC] (θ). It should be noted that y-coordinate of Joint C (y_C) remains constant at zero throughout the motion. To account for one full rotation of Link [AB], a total of 50 data sets are collected for training, validation, and testing purposes. To prevent overfitting issues, only 50 data sets for the parameters ϕ, θ and x_C are determined to represent the entire motion of the mechanism. These 50 data sets are randomly split into 80% for training (40 data sets), 10% for validation (5 data sets), and 10% for testing (5 data sets).

An additional test is also conducted to evaluate the overall performance of the trained algorithm. For the full rotation of Link [AB], a separate set of 200 data sets is determined for this additional testing. These 200 data sets are entirely distinct from the initial 50 data sets used for training, validation, and testing of the algorithm. The MATLAB Neural Net Fitting Tool is employed for the training, validation and testing in this study.

IV. RESULTS

Based on the results obtained from the position analysis using the vector loop method, the minimum and maximum positions of Joint C are found to be $x_{C_{min}} = 0.5$ m and $x_{C_{max}} = 1.5$ m during the motion considering constant lengths of Links [AB] and [BC]. As previously mentioned, the y-component of the Joint C position coordinates remains constant at zero throughout the motion.

Table 1. Levenberg-Marquardt Backpropagation algorithm training progress

Unit	Initial Value	Stopped Value	Target Value
Epoch	0	4	1000
Elapsed Time	---	00:00:00	---
Performance	7.91	1.43e-19	0
Gradient	12.2	5.17e-11	1e-07
Mu	0.001	1e-07	1e+10
Validation Checks	0	3	6

Table 1 shows summary of the Levenberg-Marquardt backpropagation algorithm training process. According to the Table 1, algorithm training is completed after 4 epochs because training reached minimum gradient (5.17e-11). Table 1 shows that performance value is close to the target value and elapsed time is zero.

Table 2. Levenberg-Marquardt Backpropagation algorithm training results

	Observations	MSE	R
Training	40	0.0024	0.9967
Validation	5	0.5298	0.8229
Test	5	0.9237	0.5341
Additional Test	200	0.1645	0.8410

Table 2 shows the number of data sets (observations), mean squared error (MSE) and regression (R) values of the neural network algorithm. According to the Table 2, a total of 40 data sets for the training, 5 data sets for validation, 5 data sets for testing and a total of 200 data sets for additional test are employed. The mean squared error value is almost zero (0.0024) for the training; therefore, regression value is almost 1 (0.9967). On the other hand, the mean squared error value is 0.5298 for validation, 0.9237 for testing and 0.1645 for the additional test. Based on the additional test results presented in Table 2, the overall performance of the trained algorithm is not perfect, with a regression value of 0.8410. However, it is still considered good.

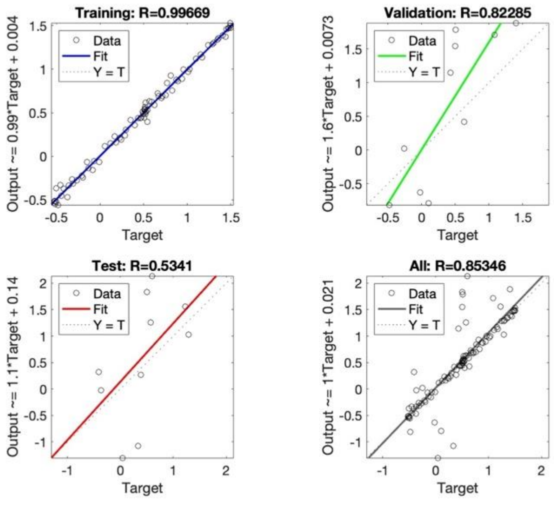

Fig. 3. Regression plots of the Levenberg-Marquardt backpropagation algorithm

Figure 3 shows regression plots of the feedforward neural network algorithm. In Fig. 3, the regression plots display predictions (outputs) and responses (targets) of neural network algorithm. According to the Fig. 3, training data sets are perfectly fit and regression value is 0.99669. As seen in Fig. 3, the regression value for testing is 0.5341, indicating a moderate fit. However, the plot demonstrates a good fit of the testing data. Conversely, the regression value for validation (0.82285) is higher than the testing regression value, but the validation regression plot shows that the validation data does not fit well.

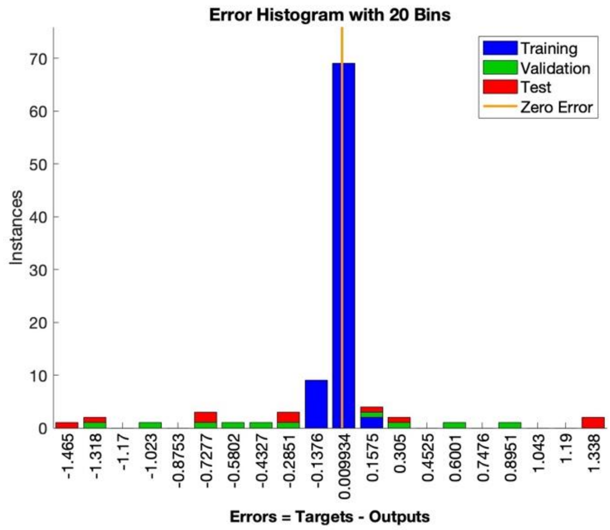

Fig. 4. Error histogram plot of the Levenberg-Marquardt backpropagation algorithm

Figure 4 shows error histogram plot of the Levenberg-Marquardt backpropagation algorithm after training. In Fig. 4, errors between prediction values and target values can be seen clearly. According to the Fig. 4, validation data sets have various error values and zero error value is 0.009934. As seen in Fig. 4, neither validation data sets nor test data sets have zero error.

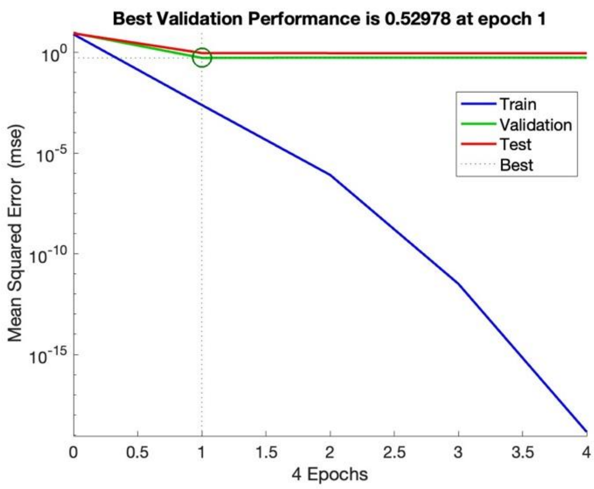

Fig. 5. Best validation performance plot of the Levenberg-Marquardt backpropagation algorithm

Figure 5 shows the best validation performance graph for the Levenberg-Marquardt algorithm. According to the Fig. 5, the best validation performance of the algorithm is 0.52978 at epoch 1. As seen in Fig. 5, mean squared error is almost zero at epoch 4.

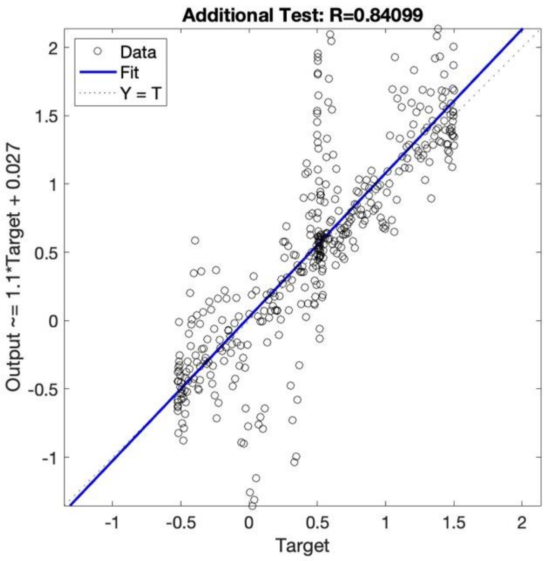

Fig. 6. Trained Levenberg-Marquardt backpropagation algorithm additional test regression plot

As mentioned previously, additional test of the trained algorithm is performed with 200 data sets and these 200 data sets are different than previously used 50 data sets for training, validation and test of the neural network algorithm. Figure 6 shows regression plot for the trained algorithm additional test. According to the Fig. 6, additional test regression value is 0.84099 and data sets are not perfectly fit but overall fit is reasonably good.

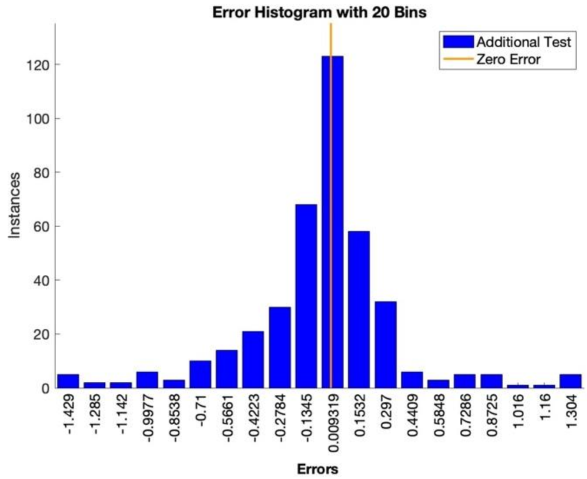

Fig. 7. Trained Levenberg-Marquardt backpropagation algorithm additional test error histogram plot

Figure 7 shows trained algorithm additional test error histogram graph. As seen in Fig. 7, there is are not any validation or training data and all 200 data sets are employed for testing. According to the Fig. 7, the zero-error value is 0.009319 but some data sets also have different error values. Zero error value for the additional test data sets is smaller than zero error value for the algorithm training data sets.

V. Discussion

The additional test results play a crucial role in comprehending the overall performance of the trained algorithm. As mentioned earlier, a total of 200 data sets are used for this additional test. It is worth noting that incorporating more data sets for the additional test could contribute to a better understanding of the trained algorithm's performance. Thus, a possible future research direction involves performing the trained algorithm with an even larger set of additional test data.

The utilization of more data sets benefits the neural network algorithm in terms of improved training, validation, and testing; however, it also raises concerns about potential overfitting. To achieve a better-trained algorithm, a future research direction could investigate the optimal number of data sets while considering constraints related to overfitting.

Furthermore, there exist other neural network algorithms, such as Bayesian regularization and Scaled conjugate gradient backpropagation algorithms. Exploring different neural network algorithms may lead to finding the best-suited solution for the slider-crank (R-RRT) mechanism. A promising avenue for future research is to investigate various neural network algorithm applications on the same slider-crank mechanism and compare their performances too identify the most effective approach.

VI. CONCLUSION

This study introduces the position analysis of the slider-crank mechanism (R-RRT) using an artificial neural network approach. The Levenberg-Marquardt backpropagation algorithm is utilized for training, validation and testing. An additional test is also conducted to evaluate the overall performance of the trained algorithm. While the data sets do not achieve a perfect fit, the trained algorithm exhibits a reasonably good fit overall.

ACKNOWLEDGMENT

Author wishes to express his gratitude to the Ministry of National Education of the Republic of Türkiye which indirectly made this work possible.

REFERENCES

[1] A. V. Duka, "Neural network based inverse kinematics solution for trajectory tracking of a robotic arm," *Procedia Technology*, vol. 12, pp. 20-27, 2014.

[2] J. Lu, T. Zou, X. Jiang, "A neural network based approach to inverse kinematics problem for general six-axis robots," *Sensors*, 22, 8909. 2022.

[3] A. Prado, H. Zhang and S. K. Agrawal, "Artificial neural networks to solve forward kinematics of a wearable parallel robot with semi-rigid links," *2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, 2021, pp. 14524-14530.

[4] O. Denizhan, "Two-position synthesis of the four-bar planar linkage mechanisms using artificial neural networks," *In 5th International Conference on Applied Engineering and Natural Science*, July 10-12, 2023 Konya, Turkey. pp. 964-970.

[5] R. L. Norton, *Design of Machinery: An Introduction to the Synthesis and Analysis of Mechanisms and Machines*, 6th ed.: McGraw Hill, 2020.

[6] K. J. Waldron, G. L. Kinzel, and S. K. Agrawal, *Kinematics, Dynamics, and Design of Machinery*, 3rd ed.: Wiley, 2016.

[7] J. J. Uicker Jr., G. R. Pennock, and J. E. Shigley, *Theory of Machines and Mechanisms*, 5th ed.: Oxford University Press, 2017.

[8] H. Yu and B. M. Wilamowski, *Intelligent systems*, ed.: CRC Press, 2018.

[9] J. J. More, The Levenberg-marquardt algorithm: Implementation and theory, Numerical Analysis, ed: Springer, 1978.

[10] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std. 802.11, 1997.