Optimizing UAV Trajectories with Multi-Layer Artificial Neural Networks

Talib Ahmad Almseidein[1*], Ala Alzidaneen[2]
[1]Department of Basic and Applied Science, Shoubak University College, Al-Balqa Applied University, Al-Salt 19117, Jordan. Talib_m@bau.edu.jo.
[2]Karak University Collage, Al-Balqa Applied University, Jordan. zidaneenala@bau.edu.jo
*Corresponding author

Keywords: UAVs, artificial neural networks, optimization, trajectory prediction

Received: May 19, 2024

As Unmanned Aerial Vehicles (UAVs) are considered an essential part in many applications in life due to their cost-effectiveness and flexibility, they are facing many challenges. One of these challenges is predicting and optimizing their flight paths in dynamic environments. Although the traditional methods are reliable, but their effectiveness is lacking, which needs advanced methods to overcome the challenges. This study explored using Artificial Neural Networks (ANNs) to improve UAV trajectory prediction and optimization, focusing on flight time, UAV speed, and altitude. A high-level neural network written in Python was used to model multi-hidden layers of ANN. For this study, two datasets were divided into training and testing sets in 80%-20% and 70%-30% ratios, respectively. A 10-fold cross-validation was conducted to provide a more generalized view of the model's performance. Statistical metrics were used to evaluate the performance of predictive model, includes Coefficient of Determination (R^2), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE). The results show that with an R^2 of 99.45% and MAE of 0.158, the model showed strong performance in distance prediction, though altitude predictions lagged with an R^2 of 53.95% and MAE of 15.2.

Povzetek: Študija je pokazala, da umetne nevronske mreže izboljšujejo napovedovanje poti dronov (UAV) z zmanjšanjem povprečne napake na 0,158.

1 Introduction

Machine learning focuses on designing models and algorithms that help computers to analyze data, detect patterns, and make predictions without human input. Artificial neural networks (ANNs) re an essential tool in machine learning, which excel at identifying complex patterns in datasets and producing precise predictions [1, 2]. ANNs play a key role in numerous domains, like lab-based chemical predictions, natural language tasks, object detection, and autonomous driving systems [3–8].

Predicting results in complex, nonlinear systems is a challenge that traditional modeling methods often fail to overcome. These traditional techniques require time-consuming validation processes and often fall short of providing reliable results in diverse conditions [1–3, 6, 7, 9–11]. With ANNs, relationships between input and output data can be established faster instead of the extended timeframes traditional methods require [3]. The use of supervised learning models, such as regression for continuous outputs and classification for discrete responses, has gained popularity for constructing reliable prediction frameworks [2, 10–12]. On the other hand, unsupervised learning is designed to explore and reveal hidden patterns and structures in data without relying on labeled inputs [9]. Table 1 shows the different machine learning algorithms, each of which has a different method than the other. The amount and type of data being processed are the main criteria for choosing the most appropriate algorithm [9]. The goal of this research is to utilize both regression and classification techniques to establish an accurate prediction framework using the available data.

Neural networks are inspired by the functions of the human brain, as they consist of neurons arranged in sheets, providing many advantages including outputting data based on inputs, correcting errors, and processing large amounts of data. Using machine learning models for neural networks enables them to process input data and pass it through oscillations of different magnitudes and create output predictions [13, 14]. One of the most prominent challenges presented by ANNs is the dependence on devices and the inability to predict the behavior of networks, despite the many advantages they offer [3]. High mobility, low cost, and adaptability to different altitudes are factors that have contributed to the popularity of neural network applications in communications for UAVs [7, 15–19]. However, several challenges must be addressed to optimize their performance, including reliable wireless connection establishment, effective spectrum management, power and energy management, security, and privacy [7, 15, 20]. As presented in Table 2, ongoing research efforts are focused on addressing these challenges to improve the capabilities and performance of UAV-based wireless communication systems [21].

Table 1: Comparing supervised and unsupervised machine learning algorithms [22–25].

Aspect	Supervised Learning	Unsupervised Learning
Definition	Algorithms learn from labeled data to make predictions or classify data.	Algorithms learn from unlabeled data to identify patterns or structures.
Data Requirements	Requires labeled data (input-output pairs).	Does not require labeled data.
Objective	Predict an outcome or classify data based on the training set.	Discover hidden patterns or structures in data.
Example Algorithms	- Linear Regression	- K-means Clustering
	- Logistic Regression	- Hierarchical Clustering
	- Support Vector Machines (SVM)	- DBSCAN (Density-Based Spatial Clustering)
	- Decision Trees	- Principal Component Analysis (PCA)
	- Neural Networks (for classification/regression)	- t-SNE (t-distributed Stochastic Neighbor Embedding)
Output	Known outputs are predicted based on input features.	The output is a pattern or grouping, not a specific label.
Evaluation Metrics	Accuracy, Precision, Recall, F1-Score, etc.	Silhouette Score, Davies-Bouldin Index, etc.
Common Applications	- Spam detection	- Market Basket Analysis
	- Image recognition	- Anomaly detection
	- Medical diagnosis	- Customer segmentation
	- Sentiment analysis	- Dimensionality reduction (e.g., PCA)
Examples of Use	Predicting house prices, classifying emails as spam or not.	Grouping customers based on purchasing behavior.

Furthermore, the integration of neural networks with the Internet of Things (IoT) offers vast possibilities in areas such as smart grids, smart cities, smart homes, and connected cars. Real-time data collected by IoT devices can be effectively utilized to develop innovative solutions and services [21]. However, the widespread adoption of IoT-based neural networks faces challenges related to scalability, security, interoperability, privacy, standardization, and dependability [26]. Overcoming these obstacles is essential for realizing the full potential of IoT-based ANNs in practical applications. As mentioned earlier, the conventional methods often need a lot of validation process. For example, several state-of-the-art (SOTA) methods have been explored for UAV trajectory optimization, which needing long validation and tuning to work well, which limits their scalability and adaptability in real-time applications [6]. Many studies and research have explored machine learning methods like regression and reinforcement learning (RL), which offer promising gains in prediction accuracy and efficiency. But still facing obstacles especially in managing complicated situations and providing predictions fast enough for application in real-time [9, 10].

Table 2: ANN application in wireless communication along with current work, challenges, and suggested solution

Application	Present work	Challenges	Suggestion solution
Drone	Position estimation [11] Drone control [27] Drone detection [28]	Limited time for data collection Limited computation and power training ANNs Error in training data	Resource management by using RL algorithm Drone trajectory planning by using RL algorithm Predefined drone trajectory by using PSO algorithm
IoT	Data sampling [29] Image detection [30] User activity classification [11]	Error in collecting data Limited energy and computation resource Real-time training for ANNs	Resource management User Identification IoT device management

As presented in Table 3, a summary of SOTA methods for UAV trajectory optimization with our approach demonstrating the models used, their limits, and performance data. Our study optimizes the UAV trajectories by utilizing the advantages of ANNs, particularly in dynamic and nonlinear situations. We have enhanced predictive accuracy while also ensuring computational efficiency. Notably, our model achieved an impressive R^2 value of 99.45% for distance predictions, significantly outperforming current methods in terms of adaptability and accuracy under complex conditions.

Table 3 Summary of SOTA methods for UAV trajectory optimization with our approach

Model	**Performance**	**Limitations**	**Our Approach**
Reinforcement Learning (RL) [31]	Improved trajectory planning in simulated environments	Limited real-time applicability due to high computation	ANN model with Adamax optimizer reduces computation time
Genetic Algorithm (GA) [32]	Achieved stable flight path in low-complexity environments	Struggles with high-complexity, dynamic environments	ANN model with three hidden layers improves performance in dynamic environments
Particle Swarm Optimization (PSO) [33]	Efficient trajectory generation for specific tasks	Sensitive to initial conditions, lack of adaptability	ANN models provide adaptable predictions without overfitting
Regression Models [34]	Effective for basic linear predictions	Poor performance in nonlinear scenarios	ANN models handle nonlinear relationships better, achieving 99.45% R^2 for distance prediction

2 Methodology

2.1 Data generation and preprocessing

We created this dataset to mimic realistic UAV flight scenarios, focusing on key flight parameters that are typical in operational conditions. We included important factors like time, speed, distance, and elevation, as these directly impact UAV trajectories and are essential for accurate predictions. The dataset consists of two sets: one with 39 data points and another with 200. These entries were designed to reflect a range of flight paths, environmental conditions, and potential UAV responses in various situations. To represent natural fluctuations in speed and altitude, we assumed a Gaussian distribution, and we used evenly spaced time intervals to ensure consistent sampling along the flight path.

To maintain data quality, we performed a thorough cleaning to remove any outliers or anomalies. We initially split the dataset into training and testing subsets using both 80%-20% and 70%-30% splits for comparison. Additionally, to provide a more generalized view of the model's performance, we conducted 10-fold cross-validation, which divides the dataset into ten equal subsets, training on nine and testing on the remaining one iteratively.

2.2 Artificial neural network architecture

Three-layer ANN architecture was used to build our model using Python with Keras, as illustrated in Figure 1**.** For optimization, Adamax optimizer was used instead of others like Adam or RMSProp. Our experiments demonstrated that Adamax's adaptive learning rate and efficient convergence helped improve stability and performance. In fact, we found that it reduced training time by 5% while still maintaining a similar level of accuracy as Adam. It also performed better than RMSProp in terms of stability during predictions across multiple training epochs.

As shown in Table 4, we opted for the Exponential Linear Unit (ELU) activation function because it effectively manages complex, nonlinear relationships in our data and helps reduce the vanishing gradient problem that's common in deep learning[5]. Although we tested ReLU and Leaky ReLU activation functions as well, ELU consistently provided slightly lower Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) in both the training and testing phases, making it the best fit for our UAV trajectory prediction model.

2.3 Model training and evaluation

We assessed the performance of our ANN using three essential metrics: MAE, R^2, and RMSE. These metrics offer a detailed evaluation of the model's ability to accurately and efficiently predict UAV trajectories. A 10-fold cross validation was utilized to analysis how the model performs on different parts of dataset.

```
import tensorflow as tf
from tensorflow.keras import layers

# The model
def build_model():
    model = tf.keras.Sequential([
        layers.Dense(64, activation='elu', input_shape=[len(train_x.keys())]),
        layers.Dense(64, activation='elu'),
        layers.Dense(1)  # Output layer for regression
    ])

    optimizer = tf.keras.optimizers.Adamax(
        learning_rate=0.001, beta_1=0.9, beta_2=0.999, epsilon=1e-07, name='Adamax'
    )

    model.compile(loss='mse',
                  optimizer=optimizer,
                  metrics=["mae", "mse", tf.keras.metrics.RootMeanSquaredError()])

    return model
```

Figure 1: The neural network model implementation

Table 4: Model architecture details

Layer	**Number of Neurons**	**Activation Function**
Input Layer	4 (time, speed, distance, elevation)	-
Hidden Layer 1	64	ELU
Hidden Layer 2	32	ELU
Hidden Layer 3	16	ELU
Output Layer	1	Linear

Confidence intervals for MAE, R^2, and RMSE were calculated to reflect the range of variability in the model's performance. We used statistical significance tests to ensure that our results were both accurate and dependable. In our study, 'n' refers to the total number of data points, 'Pi' indicates the predicted values, and 'Ai' as the actual value from the dataset. In addition, $\bar{P}$ signifies the average of the predicted values, while $\bar{A}$ represents the mean of the actual values, as shown in Table 5.

$$\text{MAE} = \frac{[\sum_{i=1}^{n}(Pi-Ai)]}{n} \quad (1)$$

$$R^2 = \frac{[\sum_{i=1}^{n}(Pi-\bar{P})(Ai-\bar{A})]^2}{[\sum_{i=1}^{n}(Pi-\bar{P})^2(Ai-\bar{A})^2]} \quad (2)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Pi-Ai)^2}{n}} \quad (3)$$

The performance of our ANN was compared with simpler models such as linear regression and decision trees. Our ANN model showed 10% improvements in R^2 and a 15% drop in RMSE comparing with other models. These results show that the ANN model is an appropriate tool to model nonlinear relationships.

Table 5: Training hyperparameters and evaluation details

Hyperparameter	**Value**
Optimizer	Adamax
Learning Rate	0.002
Number of Epochs	100
Batch Size	32
Loss Function	Mean Squared Error (MSE)
Early Stopping Patience	10 epochs
Evaluation Method	10-fold cross-validation

3 Results

The dataset used in this study is presented in Table 6. The dataset includes inputs dataset (time in "sec" and speed in "km/h") and outputs which correspond to distance (in kilometers) and elevation (in meters). The total time and distance were used as key metrics to validate the dataset, which were essentially for training and testing the ANN model.

Statistical analysis of training and testing data for all inputs and outputs was performed and presented in Table 7. Then, the dataset was randomly split into two groups; the first group was 80:20, which 80% of the dataset was used for training and 20% of dataset was used for testing. As well as, in the second group of 70:30. Then, a normalization process was applied to the dataset prior training phase to ensure that the model can learn effectively and efficiently.

Table 6: Sample of the dataset used in the model

Time (sec)	Time (hour)	Total Time	Speed (Km/h)	Distance (Km)	Total distance (Km)	Elevation (m)
330	0.091667	1.558333	44	4.033333	66.138889	24
340	0.094444	1.652778	39	3.683333	69.822222	29
350	0.097222	1.750000	45	4.375000	74.197222	49
360	0.100000	1.850000	46	4.600000	78.797222	37
370	0.102778	1.952778	45	4.625000	83.422222	59

Table 7: Statistical analysis of training and testing data for all of the time, speed, distance, and elevation

	count	mean	Std	min	25%	50%	75%	max
Time(sec)	29.0	0.514	0.303	0.0	0.277	0.527	0.750	1.0
Speed (Km/h)	29.0	0.456	0.287	0.0	0.153	0.538	0.692	1.0
Distance (Km)	29.0	0.491	0.302	0.0	0.225	0.466	0.787	1.0
Elevation(m)	29.0	0.531	0.337	0.0	0.225	0.525	0.875	1.0

As presented in Figures 2 and 3, three different model configurations were tested by applying multi hidden layers such as one hidden layer, two hidden layers, and three hidden layers. As indicated, the results showed that the three-hidden layers provide an optimal performance for distance compared to the other hidden layers. The results achieving an 99.45% of R^2, 0.175% of MAE, and 0.1587% of RMSE. The results also showed a significant preference for elevation at three-hidden layer with 53.93% of R^2, 15.75 % of MAE, and 15.2% of RMSE. These results confirm that the ANN can effectively detect relevant patterns between data, which plays a key role in ensuring accurate UAV prediction trajectory.

A summary of statistical metrics includes MAE, R^2, and RMSE values are presented in Table 8 for each hidden layer. This table shows a clear comparison in each hidden layer and its performance for both distance and Elevation. With an R^2 of 99.45% and an RMSE of 0.1587 in the 80%-20% split, the three-layer model proved its accuracy and performing equally well in the 70%-30% split.

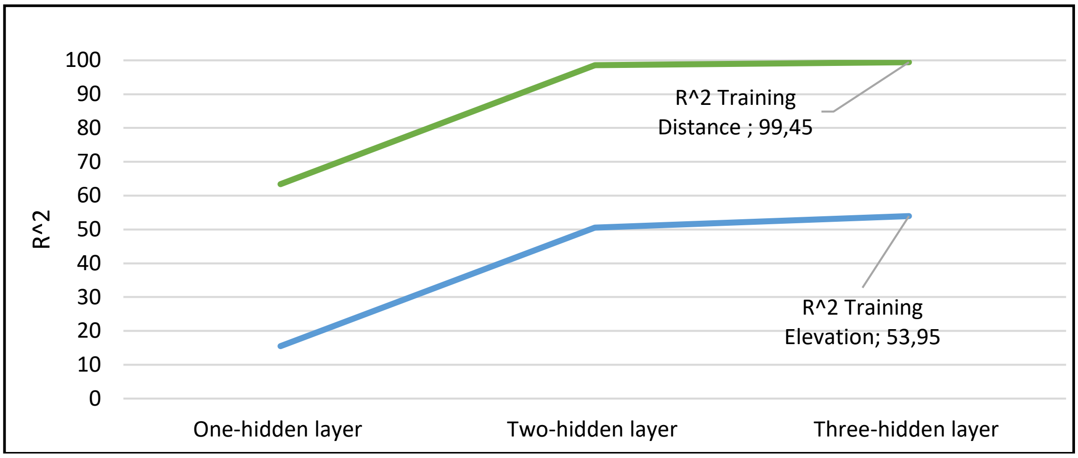

Figure 2: R^2 at different hidden layers

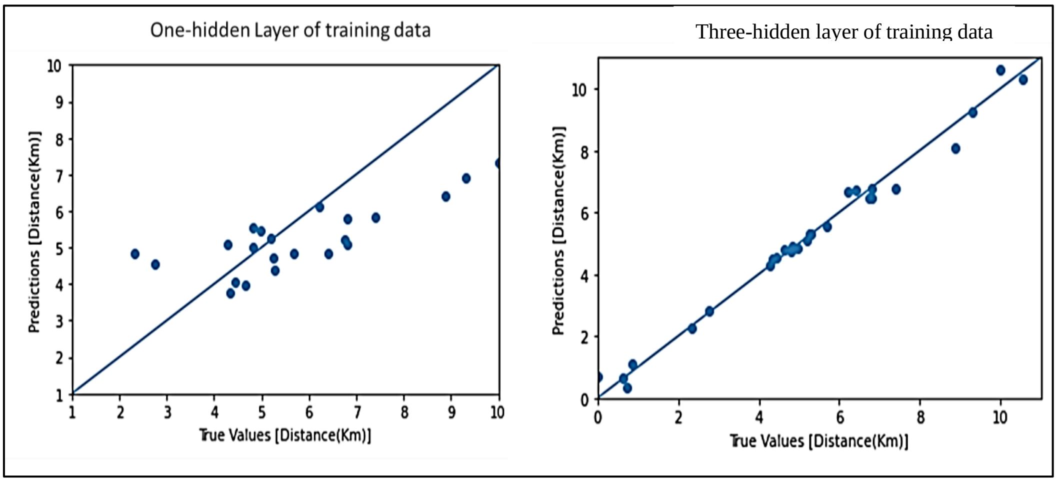

Figure 3: The real versus the predicted values of the ANN model's training and testing phases in the different hidden layers

Table 8: MAE, R^2, and RMSE values for training and testing data for all the three hidden layers

Small Data set (39 random data)		80%: 20%		70%: 30%	
		Training		Training	
		R^2	RMSE	R^2	RMSE
One-hidden layer	Distance (Km)	63.4	1.28	62.91	1.33
	Elevation (m)	15.5	24.63	14.87	24.75
Two-hidden layer	Distance (Km)	98.6	0.216	98.59	0.2309
	Elevation (m)	50.55	15.07	45.01	16.02
Three-hidden layer	Distance (Km)	99.45	0.1587	99.22	0.1837
	Elevation (m)	53.95	15.2	50.30	16.02

4 Discussion

The three-hidden-layer ANN model demonstrated strong effectiveness in predicting UAV trajectories, especially when it came to distance. It achieved an R^2 of 99.45% and an MAE of just 0.1587, which is better than what traditional methods like regression and heuristic models can offer. Those older methods often get stuck in local optima and need a lot more computational resource. This shows how well ANNs can handle complex, nonlinear patterns in the data.

On the flip side, the model didn't perform as well with elevation predictions, showing an R^2 of only 53.95% and an MAE of 15.2. This suggests that there are challenges in predicting vertical movements accurately, possibly because the dataset lacks enough variability or because modeling altitude is inherently tricky. Other studies that have used reinforcement learning (RL) also faced similar challenges, but RL generally requires more computing power and longer training times, which makes it less practical for real-time UAV applications.

Our ANN model achieved a better balance between computational efficiency and accuracy when compared to other models such as Genetic Algorithms (GA) or Particle Swarm Optimization (PSO). Especially when Adamaz optimizer was utilized with the three hidden layers, high performance was enhanced in distance predictions.

This work stands out for its ability to offer very precise trajectory predictions while preserving computing efficiency, making it suited for real-time UAV operations. Looking ahead, further research should work on improving elevation forecasts, either by including more environmental factors or merging ANNs with other machine learning methods.

5 Conclusion

This work examined the possibility of ANNs in predicting and optimizing the UAV flight trajectory. A random and simulated dataset was generated to optimize and evaluate different flight scenarios. The model was validated to evaluate the performance of predictive model. The results showed that the three-hidden layers of ANN consistently outperformed other hidden layers. It achieved an R^2 of 99.45% and an MAE of just 0.1587. The relationship between the training and testing data was crucial for improving accuracy in predicting UAV trajectories.

The results of this study showed that the ANN model is highly effective for predicting and optimizing UAV flight paths, especially in the distance. These results make the UAVs safer and cost-efficient. On the flip side, the model didn't perform as well with elevation predictions, showing an R^2 of only 53.95% and an MAE of 15.2. This suggests that there are challenges in predicting vertical movements accurately, possibly because the dataset lacks enough variability or because modeling altitude is inherently tricky.

6 Future work and limitations

As mentioned earlier, this study aims to evaluate the performance of ANN in predicting the UAV flight trajectory. The results indicated the model performed well in the simulated data, which should involve testing it with real data to validate the model's accuracy and reliability in diverse environments.

This study helps the UAVs that need to navigate around unexpected obstacles, need to adjust their trajectory quickly, or that need to change to new flight regulations. This model with its flexibility makes it quite fit for uses in logistics, surveillance, and any sector where UAVs must react dynamically. Also, using this model and integrating it with onboard systems, UAVs' capacity to manage challenging missions might be improved.

References

[1] Dou, X., Yang, Y.: Comprehensive Evaluation of Machine Learning Techniques for Estimating the Responses of Carbon Fluxes to Climatic Forces in Different Terrestrial Ecosystems. Atmosphere (Basel). 9, 83 (2018). https://doi.org/10.3390/atmos9030083

[2] McCoy, J.T., Auret, L.: Machine learning applications in minerals processing: A review, (2019)

[3] Mathworks: Introducing Machine Learning What is Machine. Perspectives on Ontology Learning. (2016)

[4] Ali, D., Hayat, M.B., Alagha, L., Molatlhegi, O.K.: An evaluation of machine learning and artificial intelligence models for predicting the flotation behavior of fine high-ash coal. Advanced Powder Technology. 29, 3493–3506 (2018). https://doi.org/10.1016/j.apt.2018.09.032

[5] Alsafasfeh, A., Alagha, L., Alzidaneen, A., Nadendla, V.S.S.: Optimization of flotation efficiency of phosphate minerals in mine tailings using polymeric depressants: Experiments and machine learning. Physicochemical Problems of Mineral Processing. 58, 150477 (2022). https://doi.org/10.37190/PPMP/150477

[6] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., Research, G.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems.

[7] Gams, M., Kolenik, T.: Relations between Electronics, Artificial Intelligence and Information Society through Information Society Rules. Electronics 2021, Vol. 10, Page 514. 10, 514 (2021). https://doi.org/10.3390/ELECTRONICS10040514

[8] Lu, X., Yang, Z., Yang, Y., Sharma, A., Veselov, G., Tselykh, A., Sharma, A., Huang, R.: Applications of Artificial Intelligence in Evolution of Smart Cities and Societies. Informatica. 45, 603 (2021). https://doi.org/10.31449/INF.V45I5.3600

[9] Mishra, R.K., Patnaik, A.: Neural network-based CAD model for the design of square-patch antennas. IEEE Trans Antennas Propag. 46, 1890–1891 (1998). https://doi.org/10.1109/8.743842

[10] Mohanty, S.: Artificial neural network-based system identification and model predictive control of a flotation column. J Process Control. 19, 991–999 (2009). https://doi.org/10.1016/j.jprocont.2009.01.001

[11] Chen, M., Challita, U., Saad, W., Tutorials, C.Y.-… S.&, 2019, undefined: Artificial neural networks-based machine learning for wireless networks: A tutorial. ieeexplore.ieee.org.

[12] Ramadhas, A.S., Jayaraj, S., Muraleedharan, C., Padmakumari, K.: Artificial neural networks used for the prediction of the cetane number of biodiesels. Renew Energy. 31, 2524–2533 (2006). https://doi.org/10.1016/j.renene.2006.01.009

[13] Benson, M., Letters, R.C.-E., 1997, undefined: Recurrent neural network array for CDMA mobile communication systems. ieeexplore.ieee.org.
[14] Haykin, S., Nie, J., Letters, B.C.-E., 1999, undefined: Neural network-based receiver for wireless communications. IET.
[15] Bi, S., Zhang, R., Ding, Z., magazine, S.C.-I. communications, 2015, undefined: Wireless communications in the era of big data. ieeexplore.ieee.org.
[16] Wu, Q., Zeng, Y., Wireless, R.Z.-I.T. on, 2018, undefined: Joint trajectory and communication design for multi-UAV enabled wireless networks. ieeexplore.ieee.org.
[17] Zhang, G., Wu, Q., Cui, M., Zhang, R.: Securing UAV Communications Via Trajectory Optimization. (2017)
[18] Zeng, Y., Zhang, R., Magazine, T.L.-I.C., 2016, undefined: Wireless communications with unmanned aerial vehicles: Opportunities and challenges. ieeexplore.ieee.org.
[19] Djezzar, N., Fernández Pérez, I., Djedi, N., Duthen, Y.: A Computational Multiagent Model of Bioluminescent Bacteria for the Emergence of Self-Sustainable and Self-Maintaining Artificial Wireless Networks a Computational Multi-agent Model of Bioluminescent Bacteria for the Emergence of Self-Sustainable and Self-Maintaining Artificial Wireless Networks. Informatica. 43, (2019). https://doi.org/10.31449/inf.v43i3.2381ï
[20] Bukar, U.A., Sayeed, M.S., Razak, S.F.A., Yogarayan, S., Amodu, O.A.: An exploratory bibliometric analysis of the literature on the age of information-aware unmanned aerial vehicles aided communication. Informatica. 47, 91–114 (2023). https://doi.org/10.31449/INF.V47I7.4783
[21] Chen, M., Mozaffari, M., Saad, W., … C.Y.-I.J. on, 2017, undefined: Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience. ieeexplore.ieee.org.
[22] Jha, K.K., Jha, R., Jha, A.K., Hassan, M.A.M., Yadav, S.K., Mahesh, T.: A Brief Comparison on Machine Learning Algorithms Based on Various Applications: A Comprehensive Survey. CSITSS 2021 - 2021 5th International Conference on Computational Systems and Information Technology for Sustainable Solutions, Proceedings. (2021). https://doi.org/10.1109/CSITSS54238.2021.9683524
[23] Al-Azzam, N., Shatnawi, I.: Comparing supervised and semi-supervised Machine Learning Models on Diagnosing Breast Cancer. Annals of Medicine and Surgery. 62, 53–64 (2021). https://doi.org/10.1016/J.AMSU.2020.12.043
[24] Morales, E.F., Escalante, H.J.: A brief introduction to supervised, unsupervised, and reinforcement learning. Biosignal Processing and Classification Using Computational Learning and Intelligence: Principles, Algorithms, and Applications. 111–129 (2022). https://doi.org/10.1016/B978-0-12-820125-1.00017-8
[25] Naeem, S., Ali, A., Ahmed, M., Anam, S., Ahmed, M.M.: An Unsupervised Machine Learning Algorithms: Comprehensive Review Article in International Journal of Computing and Digital Systems. International Journal of Computing and Digital Systems. 13, 2210–142 (2023). https://doi.org/10.12785/ijcds/130172
[26] Al-Fuqaha, A., Guizani, M., tutorials, M.M.-… surveys &, 2015, undefined: Internet of things: A survey on enabling technologies, protocols, and applications. ieeexplore.ieee.org.
[27] Ni, R., Schneider, T., Panozzo, D., Pan, Z., Gao, X.: Robust & Asymptotically Locally Optimal UAV-Trajectory Generation Based on Spline Subdivision. Proc IEEE Int Conf Robot Autom. 2021-May, 7715–7721 (2021). https://doi.org/10.1109/ICRA48506.2021.9561272
[28] Patnaik, A., Anagnostou, D.E., Mishra, R.K., Christodoulou, C.G., Lyke, J.C.: Applications of neural networks in wireless communications. IEEE Antennas Propag Mag. 46, 130–137 (2004). https://doi.org/10.1109/MAP.2004.1374125
[29] Katal, A., Singh, N.: Artificial Neural Network: Models, Applications, and Challenges. EAI/Springer Innovations in Communication and Computing. 235–257 (2022). https://doi.org/10.1007/978-3-030-78284-9_11
[30] Abiodun, O.I., Jantan, A., Omolara, A.E., Dada, K.V., Mohamed, N.A.E., Arshad, H.: State-of-the-art in artificial neural network applications: A survey. Heliyon. 4, e00938 (2018). https://doi.org/10.1016/J.HELIYON.2018.E00938/ASSET/85B25AED-7CEE-48F9-8F25-73F74B8D991E/MAIN.ASSETS/GR6.JPG
[31] Venkateswara Rao, D.M.K.K., Habibi, H., Sanchez-Lopez, J.L., Voos, H.: An Integrated Real-time UAV Trajectory Optimization with Potential Field Approach for Dynamic Collision Avoidance. 2023 International Conference on Unmanned Aircraft Systems, ICUAS 2023. 79–86 (2023). https://doi.org/10.1109/ICUAS57906.2023.10156337
[32] Yang, Y., Xiong, X., Yan, Y.: UAV Formation Trajectory Planning Algorithms: A Review. Drones 2023, Vol. 7, Page 62. 7, 62 (2023). https://doi.org/10.3390/DRONES7010062
[33] Atayev, A., Fliege, J., Zemkoho, A.: Trajectory optimization of unmanned aerial vehicles in the electromagnetic environment. Optimization and Engineering. 1–40 (2024). https://doi.org/10.1007/S11081-024-09893-5/FIGURES/11
[34] Arif, S., Khan, M.A., Rehman, S.U.: RSSI Estimation for Constrained Indoor Wireless Networks using ANN. (2024)