World Journal of Advanced Engineering Technology and Sciences
eISSN: 2582-8266
Cross Ref DOI: 10.30574/wjaets
Journal homepage: https://wjaets.com/

(RESEARCH ARTICLE)

Development of model predictive controller in avoidance design of Unmanned Aerial Vehicle (UAV)

Kelechi Kingsley Igbokwe [1,*], James Eke [2] and Patrick Uche Okafor [2]

[1] *Department of Physics, Electronics, Abia State Polytechnic, Aba, Abia State, Nigeria.*
[2] *Department of Electrical, Electronics, Enugu State University of Science and Technology, Enugu State, Nigeria.*

World Journal of Advanced Engineering Technology and Sciences, 2023, 09(01), 076–083

Publication history: Received on 27 March 2023; revised on 09 May 2023; accepted on 12 May 2023

Article DOI: https://doi.org/10.30574/wjaets.2023.9.1.0137

Abstract

This research was accomplished by employing already developed mathematical model of a 6DoF UAV in free space through the motion of the 6DoF aircraft (unmanned aerial vehicle) determined by coordinate systems which allow an aircraft's position and orientation in space to be kept tracked. The discrete method of the model developed for simulation in Simulink was realized with a suitable transfer function. Then, an artificial neural network model predictive adaptive controller that can handle the nonlinearities associated with the UAV was developed based on state space technique and the model predictive controller (MPC) utilizes a neural network model to envisage future plant (aircraft) responses to potential control signals. The developed model predictive controller network was successfully trained offline using Feed-forward Back-propagation algorithm with speed and position as inputs since the underlying objective of this work is to improve the speed and position in order to advance the safety collision distance of the UAV because these parameters are mostly considered in the avoidance maneuver performance. Also, the results reveal that the speed of the generated UAV is approximately the input speed of 75 ms^{-1} during the time the UAV is affecting maneuvering. Significance of the result is that the proposed algorithm is capable of generating collision-free trajectory for different waypoint cases.

Keywords: UAV; Model Predictive Controller; Avoidance Strategies; Avoidance Design; Maneuver

1. Introduction

It has been established that Unmanned aerial vehicles (UAV) in a simple term is seen as an automated aircraft or an aircraft without a human pilot onboard to control it. According to Matthew [1], UAV is a flight that has no pilot, which include all kinds of airplanes, translational inflatable, helicopters, and airships. The unmanned vehicle together with the taking charge of control station on ground, announcement connections and launch recovery systems are used in categorization of unmanned aerial systems [1, 2]. For a UAV to be employed, it must avoid colliding with moving objects and stationary. Nevertheless, there are some fundamental similarities between UAV collision avoidance; air traffic and mobile robotics. UAVs undoubtedly have many unique characteristics to take into account, making this a fascinating research area [3].

The Model Predictive Control (MPC) method of control based on optimization guarantees that the selection of control inputs are achievable goals through minimization. The heart of MPC serves as procedural model and it give models a beauty of been categorized by a variety of attributes. However, the nature of model chosen plays a huge role on one's capacity to employ MPC. The Model Predictive Control system applies a neural network model to forecast the output of plant to a particular input signal [4]. This controller has an optimization algorithm that calculates the input or reference signals that optimize the plant's future performance. There are numerous modeling strategies that have been used to manage extremely complex and dynamic systems. The advanced process control (APC) technology known to have been

* Corresponding author: Kelechi Kingsley Igbokwe

effectively used is model predictive control (MPC), which is depicted in the block diagram of Figure 1.1. According to Lee and Marcus' definition [5], MPC is one method for developing a feedback controller synthesis from an understanding of open loop controllers. It gauges the condition of the operation right now and then quickly computes the open-loop control function. Following a brief period of use of the first component of this function, a new value of the function is computed for this measurement.

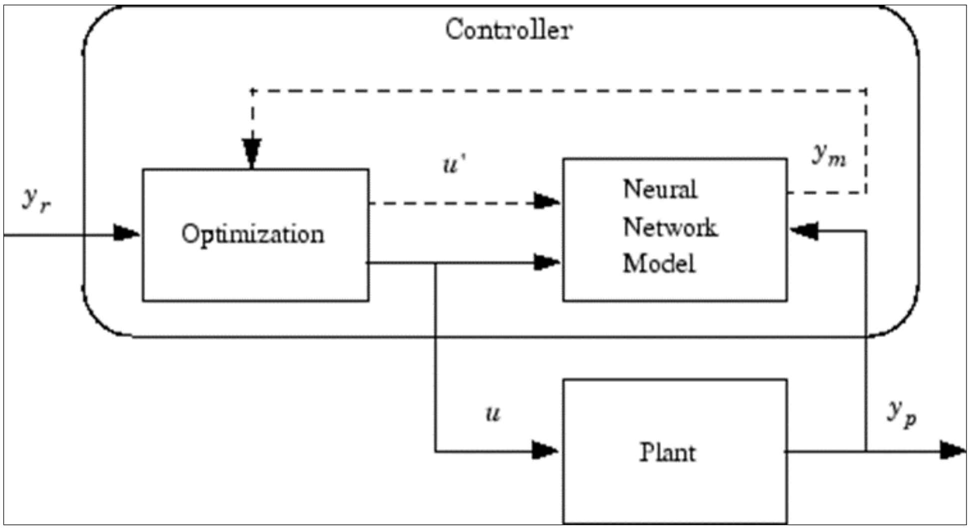

Figure 1 Block Diagram of Model Predictive Control [4]

The objective of this research is to develop model predictive control (MPC) strategies in the collision avoidance system of UAV of an unmanned aerial vehicle (UAV).

2. Material and methods

The materials are of software and hardware requirements. In the software requirements, UAV specification was considered which is typically a lightweight aircraft as a result, MathsWork MatLab, Simulink and Aerospace toolboxes after not later than 2016 release is required though, and the 2018 release was used in this work. Achieving real-time communication requires additional software requirements and such software includes Microsoft Windows Software Development Kit (SDK) 7 Compiler and .NET Framework 4. The Windows 7 SDK provides the latest headers, libraries, metadata, and tools for building Windows 7 applications [6]. The Windows 7 SDK, when used in conjunction with Visual Studio 2010, provides the best experience for instigating various applications.

The hardware requirements comprised of Laptop Computer which is an Intel core i7 CPU running on Windows 8 operating system. A main prerequisite for a computer system is to have a good video graphics array (VGA) for enabling enlarged visual display because of the expected animation output from the proposed system.

2.1. Method of Developing the Model Predictive Controller (MPC)

The crucial initiative depends on the application of the nonlinear dynamics state which depends on factorization coefficient [7,3]. An UAV state space representation is obtained where each of its process matrices is represented as current state functions. Considering a system that is dynamic as presented in Equation (2). The formulation of the MPC involves the transformation of the nonlinearity of a system to the subsequent space order:

$$\dot{x} = A_c(x)x + B_c(x)u \text{.....................(1)}$$

$$y = C_c(x)x + D_c(x)u \text{.....................(2)}$$

But $x \in \Re^n$ is described as state vector; $u \in \Re^m$ is described as initial vector; $y \in \Re^p$ is described as resulting vector and the $A_c(x) \in \Re^{n\times n}$, $B_c(x) \in \Re^{n\times m}, C_c(x) \in \Re^{p\times n}$, $D_c(x) \in \Re^{p\times m}$ respectively are described as pseudo-linear system in the order of the matrices, same as the pairs $\left(A_c(x), B_c(x)\right)$ and $\left(A_c(x), C_c(x)\right)$ that are able to be stabilized and detected as $\forall x \in \Re^n$ accordingly. It is important to make a note of the fact that except for the scalar system, this representation of $A_c(x), B_c(x), C_c(x)$ and $D_c(x)$ is generally indistinctive. Various matrices of state-dependent coefficient are obtained from equations of the motion which probably may or may not be a solution to the problem of optimization. Nonetheless, here is derived a common factorization. Equation's discrete-time equivalent (2) can be achieved through the use of a specified sample time zero-order-hold (ZOH). Let the system's discrete time counterpart be as follows:

$$x_{k+1} = A(x_k)x_k + B(x_k)u_k \text{.......................}(3)$$

$$y_k = C_c(x_k)x_k \quad \text{.......................} (4)$$

Where $A(x_k)$ and $B(x_k)$ are distinct estimations of uninterrupted $A_c(x)$ and $B_c(x)$ in that order. According to explanation of UAV, it is able to be observed that $D(x_k) = 0$; and $B(x_k) = B$ (the steady matrix). Because of Equation (4) is in the order of pseudo-linear, then the matrices of the system is regard as being variables for every time of illustration, that is, t_k and t_{k+1}) while t_{k+1} - t_k = Δt . It ought to be made a note of Equation (2) in which the first term of $A_c(x)x$ disappears when $x = 0$. For the UAVcase, $f(x)$ does not completely vanish when $x = 0$. Hence, a slim transform is highly required in Equation (2) to report it. The system of the UAV structure is then change to the expression given in equation (5) below;

$$\dot{x} = A_c(x)x + B_c(x)(u_0 + \delta u) \text{.......................} (5)$$

where u_0 is the steady strength expression necessary to put in equilibrium the UAV's heaviness. The UAV is presumed to operate from an initial state of equilibrium and then the new design of control centers on production of δu. Obviously, for $\delta u = 0$, in the "hover" balance condition, $\dot{x} = 0$, given that $A_c(x)x + B_c(x)u_0 = 0$. Equation (6) provides a probable means of factorizing the expression of motion to the form of the SDC.

$$A_c(x) = \begin{bmatrix} 0_{3\times3} & R_{BI}^T & 0_{3\times3}0_{3\times3} \\ 0_{3\times3} & A_{22} & A_{23}\ A_{24} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3}\ W \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3}\quad A_{44} \end{bmatrix} \text{.......................} (6)$$

Where

$$A_{22} = \begin{bmatrix} 0 & \frac{r}{2} & -\frac{q}{2} \\ -\frac{r}{2} & 0 & \frac{p}{2} \\ \frac{q}{2} & -\frac{p}{2} & 0 \end{bmatrix}, A_{23} = \begin{bmatrix} 0 & -g\frac{sin\theta}{\theta} & 0 \\ g\frac{cos\theta\ sin\varphi}{\varphi} & 0 & 0 \\ g\frac{(cos\theta+1)\ (cos\varphi-1)}{2\varphi} & g\frac{(cos\varphi+1)\ (cos\theta-1)}{2\theta} & 0 \end{bmatrix}$$

$$A_{24} = \begin{bmatrix} 0 & -\frac{w}{2} & \frac{v}{2} \\ \frac{w}{2} & 0 & -\frac{u}{2} \\ -\frac{v}{2} & \frac{u}{2} & 0 \end{bmatrix}, A_{44} = \begin{bmatrix} 0 & \frac{(J_y - J_z)r}{2J_x} & \frac{(J_y - J_z)q}{2J_x} \\ \frac{(J_z - J_x)r}{2J_y} & 0 & \frac{(J_z - J_x)p}{2J_y} \\ \frac{(J_x - J_y)q}{2J_z} & \frac{(J_x - J_y)p}{2J_z} & 0 \end{bmatrix}$$

and B_c is similar to Equation in (6). A matter which occurs as a result this type of SDC is set to be when $q = 0$, the expression similar to $\frac{sin\theta}{\theta}$ can never occur. In order to stop its occurrence, the Taylor series expansions' first three terms $sin\theta.sin\varphi$, $(cos\theta - 1)$ and $cos\varphi - 1$ become useful to create a nearly true value. In actual sense, the nearly true (approximation) values differ barely from their accurate values by a maximum of four (4) percent in the variety of $\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$.

This can be used to express A_{23} as shown in Equation (7)

$$A_{23} = \begin{bmatrix} 0 & -g\left(1 - \frac{\varphi^2}{3!} + \frac{\varphi^4}{5!}\right) & 0 \\ g\cos\theta\left(1 - \frac{\varphi^2}{3!} + \frac{\varphi^4}{3!}\right) & 0 & 0 \\ g\frac{\cos\theta+1}{2}\left(-\frac{\theta}{2} + \frac{\varphi^3}{4!} + \frac{\varphi^5}{6!}\right) & g\frac{\cos\theta+1}{2}\left(-\frac{\theta}{2} + \frac{\varphi^3}{4!} + \frac{\varphi^5}{6!}\right) & 0 \end{bmatrix} \ldots\ldots\ldots\ldots\ldots (7)$$

The discrete method for deriving the MPC is outlined below, taking into account the specific aspects of the UAV as described above.

$$x_{k+1} = A(x_k)x_k + B(u_T + \delta u_k) \ldots\ldots\ldots\ldots\ldots (8)$$

$$y_k = C_c(x_k)x_k \ldots\ldots\ldots\ldots\ldots (9)$$

Where $B(x_k) = B$ (a constant matrix) is used for this system. Following the same approach as the linear MPC, the equations of the N step state prediction were arrived at;

$$X_{k+1} = F(x_k)x_k + H(x_k)(U_T + \Delta U_k) \ldots\ldots\ldots\ldots\ldots (10)$$

$$Y_k = C_c(x_k)X_k \ldots\ldots\ldots\ldots\ldots (11)$$

where,

$$x_k = \begin{bmatrix} x_k \\ x_{k+1} \\ x_{k+2} \\ \vdots \\ x_{k+N-1} \end{bmatrix}, U_T = \begin{bmatrix} u_T \\ u_T \\ u_T \\ \vdots \\ u_T \end{bmatrix}, \Delta U_k = \begin{bmatrix} \delta u_k \\ \delta u_{k+1} \\ \delta u_{k+2} \\ \vdots \\ \delta u_{k+N-1} \end{bmatrix}, Y_k = \begin{bmatrix} y_k \\ y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+N-1} \end{bmatrix} \text{ and } (x_k) = \begin{bmatrix} I \\ A(x_k) \\ A(x_k)^2 \\ \vdots \\ A(x_k)^{N-1} \end{bmatrix}$$

Correspondingly, the input state and output is expressed below:

$$x_{k+N} = A(x_k)^N x_k + B(x_k)(U_T + \Delta U_T)^N,\ y_{k+N} = A(x_k)^N x_{k+N}$$

where:

$$B(x_k) = [A(x_k)^{N-1}B\ A(x_k)^{N-2}B\ \cdots\ A(x_k)B\ B]$$

Using similar algorithm to linear MPC, the nonlinear MPC calculates solution to minimize the charge task (cost function). The key dissimilarity that exist MPC's linear and nonlinear edition is that the objective variable at the present depends on the system's current state and has to be determined in the beginning of every time of illustration.

3. Results and discussion

Model Predictive controller uses a neural network model to forecast future plant (aircraft) responses to possible control signals. The Rolling optimization algorithm then computes the control signals that optimize future aircraft performance. The neural network aircraft model is trained offline, in batch form. According to Beale and the co-authors [8, 2], the controller requires a large amount of online computation as the optimization algorithm runs at each sample time to compute the optimal control inputs. The appropriate mathematical model of Equations (5) was used to derive the transfer function deployed in the development of the Simulink model of the predictive controller as shown in Figure 2.

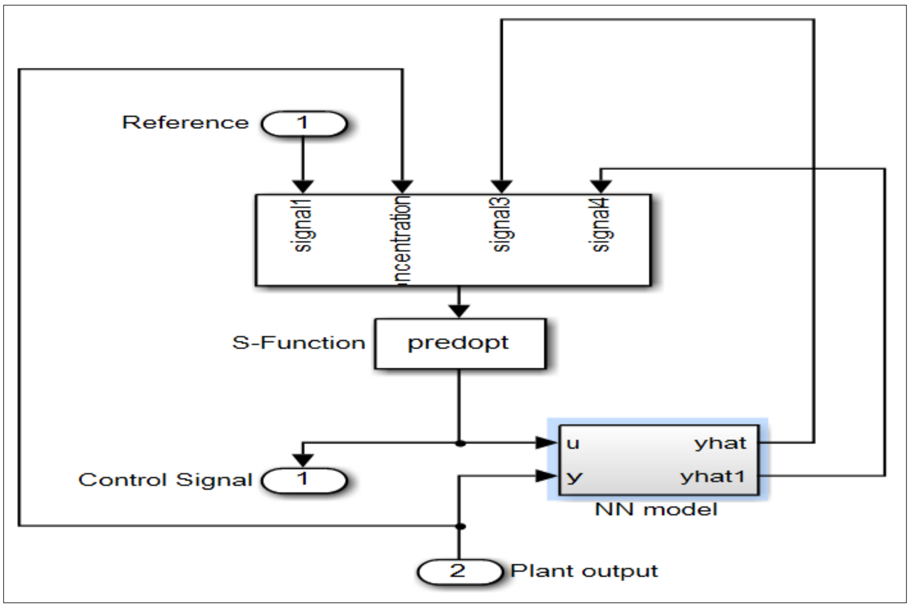

Figure 2 Simulink Predictive Controller Model

Training of the model predictive controller and the plant is a straight forward process as shown in Figure 3.

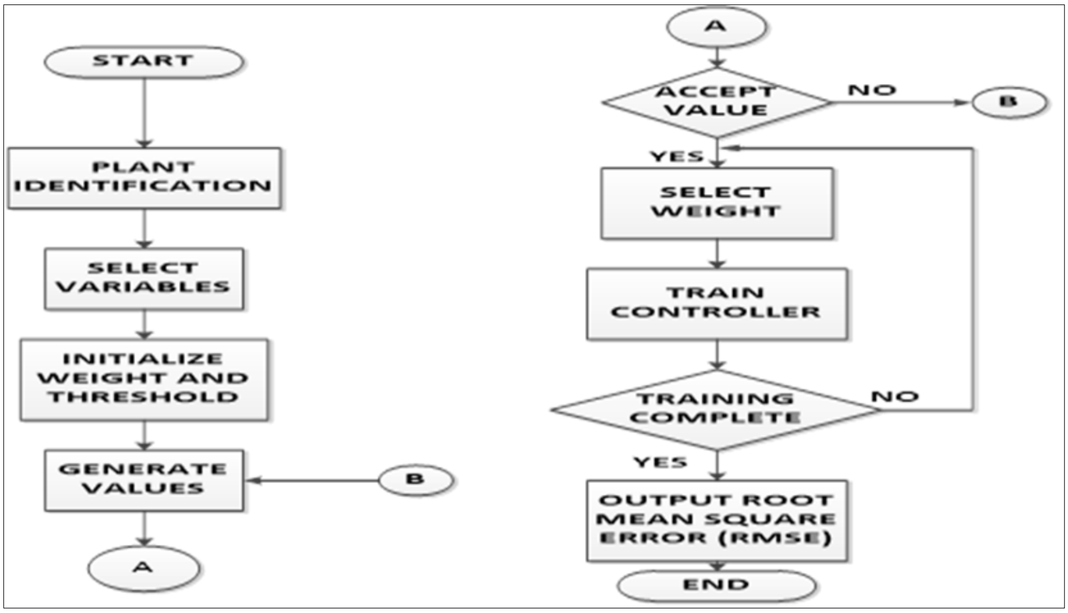

Figure 3 Flowchart of the Predictive Controller Training

The figure clearly shows that the controller and plant training are achieved according to [11], hence, call an already designed model, possibly in .slx file format (The system to be controlled must be identified before it will be controlled and the system to train is an aircraft.); generate the training data by applying a set of velocity patterns containing random step inputs to a Simulink plant model and once the generated data is accepted, aircraft training will begin (the training is performed according to the training algorithm. The error is the difference between the plant output and the predictive model output.)

The basic equation describing the weight update by the error signal at the output of neuron k is given as; [5]

$$e_k(n) = d_k(n) - O_k^o(n) \quad (12).$$

After training the aircraft, it's time to return to the control system window and start the simulation. H. Pilot training. The training program introduces data segments to the network and trains the network for a specified number of iterations. This process continues segment by segment until the entire training set is presented to the network. Training a controller can obviously take longer than training a plant model [10]. This is because the controller must be trained using dynamic back-propagation. A flowchart describing the steps to deploy the feed-forward back-propagation algorithm is shown in Figure 4 below.

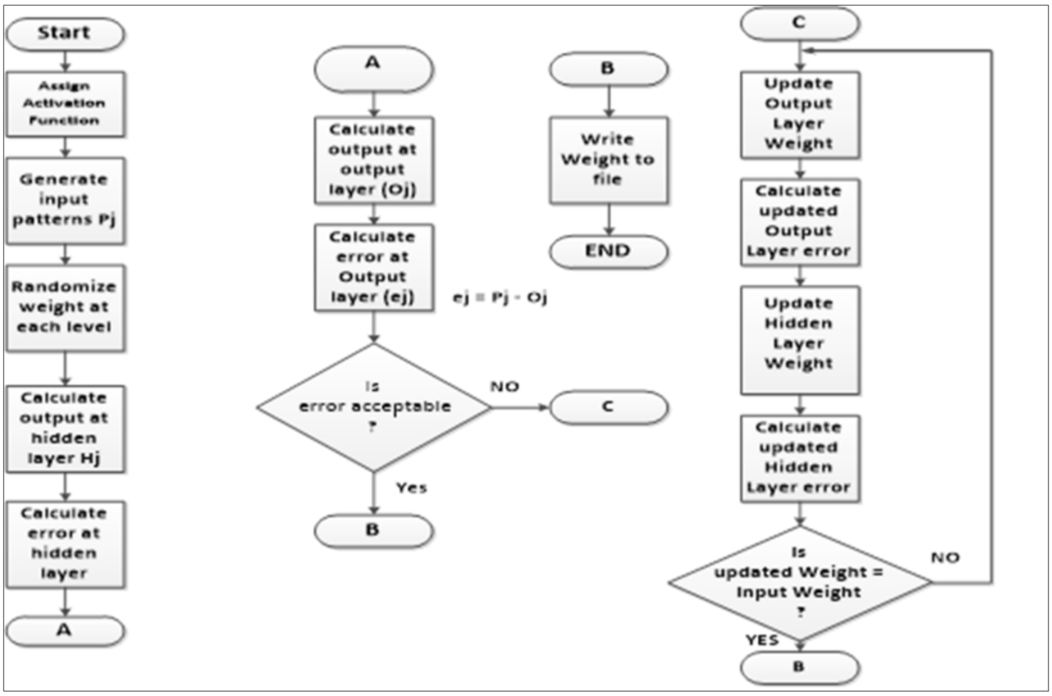

Figure 4 Flowchart for Back-propagation Algorithm

From figure 5, the NN training tool makes it easy to graphically determine the performance of control systems.

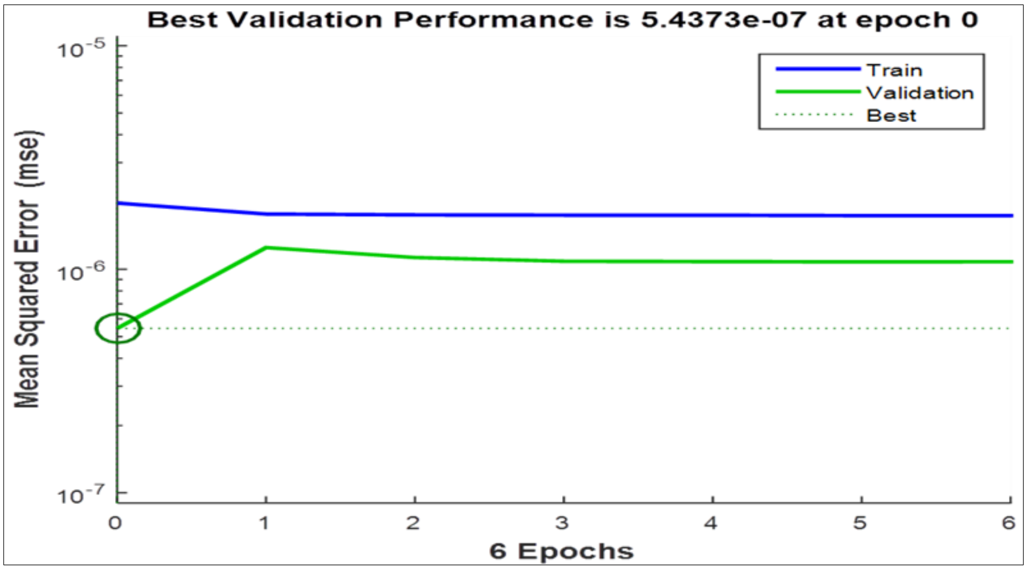

Figure 5 NN Training Performance

When the aircraft is excited by an input signal, the aircraft's output (velocity versus voltage) is fed as an input to the predictive controller. The terminal voltage v(k) is linked to the actual aircraft output ei(k) for a common excitation signal. Then the mean square value of the error ei(k) between the actual aircraft input and the estimated output voltage yields the performance error of the controller. Figure 5 shows the most training performance of $5.4347 * 10^{-7}$ the root mean squared error (RSME) is reached at epoch 0.

Since the training of the controller had, at this stage, being complete, next was to load the weight into Simulink model. Once the weight loading is done then, simulation of the model begins.

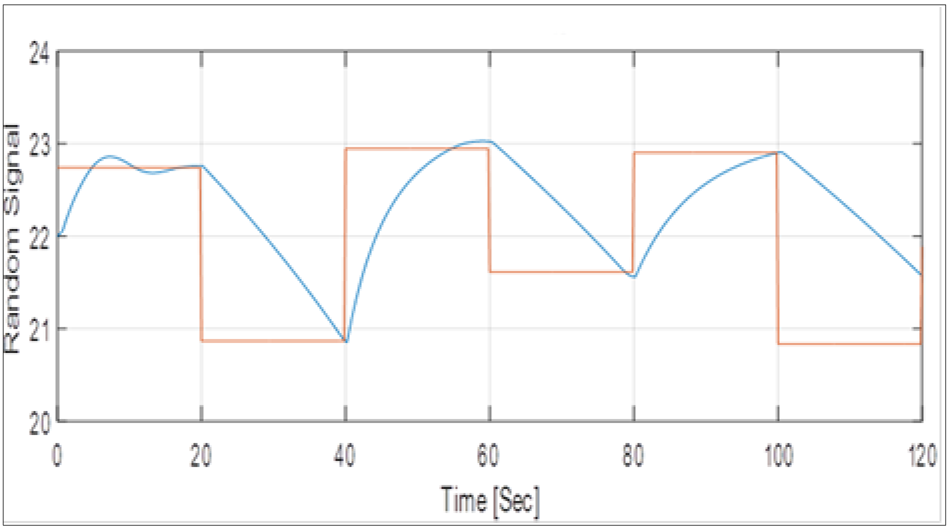

Figure 6 The Predictive Controller Response Using Random Signal

The figure 6 above is plant trajectory (output), followed the reference input for training network. The significance of this simulation result is that the controller is capable of generating accurate and precise values of parameters needed in controlling the trajectory of the aircraft in line with the Rolling Optimizing algorithm that produces a trajectory free from collisions for the next several intervals during UAV flight by anticipating and preventing potential collisions.

4. Conclusion

In a Collision Avoidance System of UAVs in free space model predictive controller is very crucial, especially for 6DoF UAV. This work was realized by employing already developed mathematical model of a 6DoF UAV in free space through the motion of the 6DoF aircraft determined by coordinate systems which allow an aircraft's position and orientation in space to be kept tracked. The developed model predictive controller network was successfully trained offline using Feed-forward Back-propagation algorithm with speed and position as inputs since the underlying objective of this work is to improve the speed and position in order to improve the safety collision distance of the UAV because these parameters are mostly considered in the avoidance maneuver performance. The results show that for effective avoidance maneuver performance in collision avoidance design of unmanned aerial vehicle in free space, the model predictive controller is essential.

Recommendations

It is recommended that a MatLab executable programme be written in C^{++} to animate the cooperative collision avoidance of the designed and simulated work, to visualize the extent of the safety distance of the aircraft in real time.

Compliance with ethical standards

Acknowledgments

The authors acknowledge the efforts of all the contribuors to this research.

Disclosure of conflict of interest

The authors share same level of interest in this research.

References

[1] Angelov, P. P. (2012). Sense and Avoid in UAS: Research and Applications, New Jersey: John Wiley & Sons.

[2] Beale M. H.; Hagan M. T. and Demuth H. B. (2015). Neural Network Toolbox, The User's Guide.The MathWorks, Inc.

[3] Camacho, E. F. and Bordons, C. (1999). Model Predictive Control London: Springer, pp. 18-20.

[4] Craig, A. K. (2020). Dynamic Systems: Modeling, Simulation and Control,'' Colorado: Wiley publishers.

[5] Haykin, S. (1999).Neural networks: A comprehensive foundation, New Jersey: Prentice Hall Inc.

[6] Venugopal, K. R.; Srinivasa, K. G and Patnaik, L. M. (2009). Soft Computing for Data Mining Applications,Studies in Computational Intelligence, DOI:10.1007/978-3-642-00193-2.

[7] Kato, K.; Ishiguro, H. and Barth, M. (2002). Identifying and localizing robots in a multi-robot system environment, Technical report, California Institute of Technology.

[8] Lee, E. B. and Markus, L. (1967). Foundations of optimal control theory, The SIAM Series in Applied Mathematics. New York-London-Sydney: John Wiley and Sons, Inc.

[9] MathsWork.com. (2017). Design Neural Network Predictive Controller in Simulink, Retrieved from https://www.mathworks.com/help/nnet/ug/design-neural-networkpredictive-controller-in-simulink.html.

[10] Matthew, J. C. (2017). Collision Avoidance And Navigation Of UAV Using Vision-Based Proportional Navigation, Student Thesis, Embry-Riddle Aeronautical University Daytona Beach, Florida.

[11] Siddique, N. and Adeli, H. (2013). Computational Intelligence Synergies of Fuzzy Logic, Neural Networks and Evolutionary Computing, West Sussex: John Wiley & Sons, Ltd.