# Control of Leading-Edge Shock of Train Using Deep Neural Network to Prevent Unstart

Stefalo Acha [1], Sun Yi [2], Frederick Ferguson [3]
[1,2,3] Department of Mechanical Engineering, North Carolina A&T State University
Email: [1] sacha@aggies.ncat.edu, [2] syi@ncat.edu, [3] fferguso@ncat.edu
*Corresponding Author

***Abstract*—The primary aim of this research is to create a comprehensive neural network model that can effectively regulate the position of the leading-edge shock in a scramjet by manipulating the required backpressure, thereby achieving, and maintaining hypersonic speeds. By utilizing computational fluid dynamic data, a dynamic model is constructed using a neural network-based approach to control the positions of the leading-edge shock train. The scramjet isolator, which is a duct where pressure increases from the inlet to the combustor via a series of shock waves, necessitates precise control of the leading-edge shock locations during scramjet operation. The model employed in this research project is a neural network adaptive controller implemented in MATLAB/Simulink software, which accounts for the nonlinear characteristics of the plant and predicts its future behavior. To enhance control performance, a robust controller is employed, integrating a learning rule that reduces the error percentage throughout the system's lifespan. The neural network is trained using flight behavior datasets, enabling it to learn from a set of training patterns. Plant identification is achieved through a neural network, capturing the system dynamics, and enabling the neural network to function as a controller. Additionally, the controller's performance is validated through simulations and optimization analyses. This research presents an adaptable, robust, and effective control system that provides added reliability and reduces disturbances.**

***Keywords*—Component, Leading Edge Shock Train, Robust Control, Reference Model, Neural Network, PID Controller, Time Series**

## I. Introduction

The robust control system is developed for hyper-sonic vehicle systems in the presence of unknown modeling errors, external disturbance, and actuators' limitation (bandwidth/saturation). First, deep learning techniques based on neural network (NN) architecture are used to developed prediction and, subsequently, control implementation. The use of neural networks model for analysis large and highly nonlinear system such as hyper-sonic vehicles is preferred as an alternative to standard nonlinear regression or cluster analysis techniques. In the identification and control of dynamic systems, neural networks have been implemented very successfully by the research group at North Carolina A&T State University for flight control of multiple aerial vehicles. When using neural networks for control, there are usually two steps involved:
[1] System identification
[2] Control design.

In the system identification phase, a NN model of the open-loop plant is developed and set for creation.

The neural network process model is needed to estimate the plant's sensitivity regarding its inputs and provides the information required for the plant. Therefore, a neural network is designed to identify the plant and learn the plant behavior through training. Then, the plant's neural network model is used to design the controller to direct the plant to produce a healthy, stable, and functioning system response. The control design phase uses the NN plant model to design or train the controller. The neural network acts as the controller by understanding the plant behavior. Hence, the plant of the system feedback traces the reference model. This controller predicts the output of the plant/system by learning and observing the system's behavior. The model reference architecture requires that a separate neural network controller be trained off-line, in addition to the neural network plant model. The model reference control applies to a larger plant class and the overall concept is the controller directs the plant's outputs to robustly follow the reference model's ones in presence uncertainties. The output leads the aircraft's behavior to a desired response. Training and learning processes are designed to reproduce the system's efficient output responses. The project's focus is to provide the model reference neural network controller, which will be trained to control a plant to follow a reference model.

The objective is to develop a new control strategy for the leading-edge shock position for a scramjet that closes the existing research gaps between the undesired uncertainties and current control systems. The leading-edge shock position is controlled to be at a certain position for every given moment to achieve a required backpressure suitable for flight operation. Due to external disturbances and system instability, such has not been an easy task to accomplish [1].

The disturbance cannot be avoided, and the neural network based adaptive controller for high-performance tracking reduces the effect of these disturbances significantly and is one of the best models to tackle this problem Fig. 1 below shows the schematic of the flow physics within isolator using artificial neural network.

The hypothesis of this research was to control the location of a leading-edge shock train position. Our implemented Simulink base model is a Comprehensive deep learning Neural Network Control. The controller in this case can place and hold the steady-state shock train leading edge location to within one duct away from the desired shock train leading edge location [2]. External disturbance can lead to instability and unpredictable behaviors of the system. The disturbance cannot be avoided, and the neural network based adaptive controllers for high performance tracking reduces the effect of these disturbances significantly. The control system will modify, (update real time) its behavior in response to changes in the dynamic of the process and the disturbances.

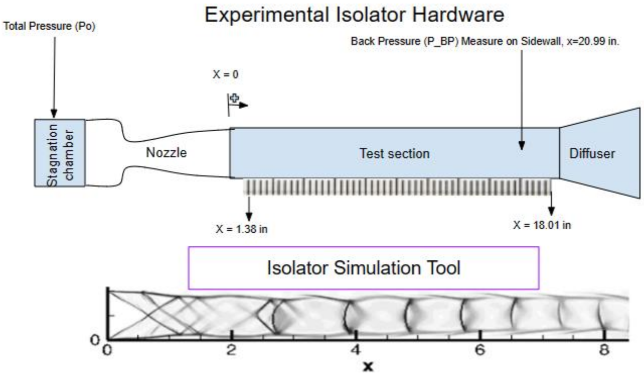


Fig. 1. Controlling the flow physics within isolators using Artificial Intelligence [2]

## II. Methodology

### A. Preliminary Studies

This research study will employ software tools like MATLAB Simulink to facilitate the design and computer simulation of the proposed system. The investigation will commence by examining the currently available products in the field, followed by a comprehensive analysis of their advantages, disadvantages, and potential areas for enhancement. This meticulous approach involves conducting thorough background research on specific types of neural network deep learning tools that are essential for achieving optimal performance in the flight system. Additionally, the study aims to develop adaptive algorithms that can adapt and evolve with changing conditions and requirements. By employing these advanced techniques and methodologies, the research endeavors to maximize the efficiency and effectiveness of the flight system design, resulting in improved performance and enhanced capabilities.

The standard neural network architecture is well-suited to high-dimensional and spatially distributed data like the one used in most engineering control [3]. This is due to the local approach of convolutional layers which enables them to exploit spatial correlations and extract low-level features of the input to carry out the task. Our research is based on training back-pressure/ shock location vs time data to achieve robust flight control of a hyper-sonic aircraft. The design architecture as shown in Fig. 2 below provides a simple view, yet each block model has tremendous engineering calculations and fixtures to achieve every given task.

The proposed design of adaptive algorithms is to control an actuator which uses backpressure data to control the scramjet leading edge shock train. The adaptive algorithms are set to operate a dynamic system “actuator” on the leading-edge shock train. The backpressure is modulated to adjust the shock-train location. Deflecting the mechanical flap or altering the mass flow rate of the jets could be used to control the backpressure [4]. Once the backpressure/shock location vs time data has been achieved during testing, data collected from the testing is being trained using our proposed design adaptive algorithms for robust flight control to achieve a quick system response during actual flight.

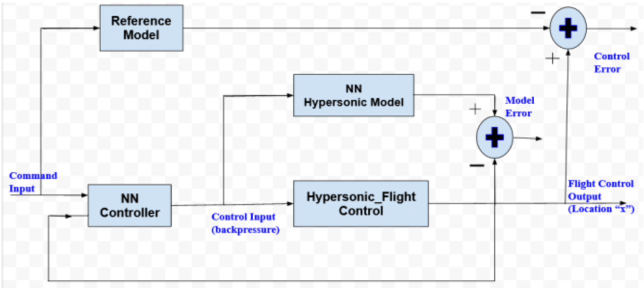


Fig. 2. Schematic of the overall neural network system to be implemented

### B. Modeling

Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of profound learning calculations [5]. Their title and structure are motivated by the human brain, imitating the way that organic neurons signal to one another. These artificial neural (neural network) are comprised of a node layer, containing an input layer, one or more hidden layers, and one out. Each node interfaces to another and has a related weight and edge. In case the output of any individual node is over the desired edge esteem, that node is enacted, sending information to the next layer of the network. Table 1 shows one of the sample datasets from computational fluid dynamics analysis used in this research. The below dataset point has a continuation for up to 590 data points and even more applicable dataset in other simulations.

Table 1. Computational fluid dynamics data [4]

| Time | Realtime (s) | Back Pressure | x | Real pressure (pa) | Real Location (inches) |
|---|---|---|---|---|---|
| 0 | 0 | 2.52 | 2.66 | 74200.94 | 6.65 |
| 1 | 0 | 2.52 | 2.65 | 74200.94 | 6.63 |
| 2 | 0 | 2.52 | 2.67 | 74200.94 | 6.67 |
| 3 | 0 | 2.52 | 2.66 | 74200.94 | 6.65 |
| 4 | 0 | 2.52 | 2.66 | 74200.94 | 6.64 |
| 5 | 0 | 2.52 | 2.66 | 74200.94 | 6.64 |
| 6 | 0 | 2.52 | 2.66 | 74200.94 | 6.64 |
| 7 | 0 | 2.52 | 2.66 | 74200.94 | 6.64 |
| 8 | 0 | 2.52 | 2.66 | 74200.9 | 6.64 |
| 9 | 0 | 2.52 | 2.66 | 74200.9 | 6.65 |

Neural networks depend on preparing information to memorize and make strides their exactness over time. Once these learning algorithms are fine-tuned for accuracy, they are effective devices in artificial intelligence, allowing us to classify, cluster data and control components at a high velocity.

## III. Design of Controller

In this study, software such as MATLAB/Simulink is used for designing and a computer simulation of the proposed system [6]. Research is conducted for the existing available products followed by studying pros and cons and scope for improvement. This procedure is engaging in performing background research of specific types of neural network deep learning tools needed to produce the best outcome for the flight system [7]. Design adaptive algorithms. The standard neural network architecture is well-suited to high-dimensional and spatially distributed data like the one used in most engineering control. This is due to the local approach of convolutional layers which enables them to exploit spatial correlations and extract low-level features of the input to carry out the task. Our research is based on training back-pressure/shock location vs time data to achieve robust flight control of a hyper-sonic aircraft [8]. The design architecture as shown in Fig. 2 provides a simple view, yet each block model has tremendous engineering calculations and fixtures to achieve every given task.

The proposed design of adaptive algorithms is to control an actuator which uses backpressure data to control the scramjet leading edge shock train. The adaptive algorithms are set to operate a system very similar to as indicated in Fig. 6. The backpressure is modulated to adjust the shock-train location. Deflecting the mechanical flap or altering the mass flow rate of the jets could be used to control the backpressure. Once the backpressure/shock location vs time data has been achieved during testing, data collected from the testing is being trained using our proposed design adaptive algorithms for robust flight control to achieve a quick system response during actual flight.

The application of neural networks for control incorporates two processes: modeling the system and control design. System modeling involves the development of a model of the system or plant needed to be controlled based on artificial neural networks. In the control process, the developed neural network model facilitates learning and training to the controller to enhance smooth and efficient control of the plant operations. A neural network is trained by representing forward dynamics to the system to attain neural control of a system. Various neural model control methods include predictive model control, NARMA-L2 control, PID/ neural network control and model reference control was studied for the control application of this research.

Artificial neural networks have enhanced the modeling and control of complex and varying plants. This enhancement has been achieved by developing algorithms capable of learning past experiences, creating an environment for adaptability and learning systems that contribute to the development of modeling and controlling systems operating in dynamic environments [9].

### A. PID Control Theory

The PID control conspire is named after its three adjusting terms, whose entirety constitutes the manipulated variable (MV). The proportional, integral, and derivative terms are summed to calculate the yield of the PID controller (Fig. 3). Characterizing u(t) as the controller output, Controller producers orchestrate the proportional, integral, and derivative modes into three different controller calculations or controller structures. These are called: interactive, Noninteractive, and Parallel algorithms. 1) Interactive Algorithm, 2) Noninteractive Algorithm, and 3) Parallel Algorithm. For the sake of this research, emphasis on based control is limited in exploring the Parallel Algorithm.

The output of a PID controller, which is equal to the control input to the plant, is calculated in the time domain from the feedback error as follows:

$$u(t) = K_{p^e}(t) + K_i \int_0^t e\,(\tau)d\tau + K_d \frac{d}{dt} e(t) \tag{1}$$

Where:

$K_p = K_c$ is the Proportional Gain, $K_i = \frac{K_c}{T_i}$ is the Integral Gain, and $K_d = K_c\, T_d$ is Derivative Gain.

$$e(t) = r(t) - y(t) \tag{2}$$

$t$ is the time (sec), $r$ is the reference desired signal input.

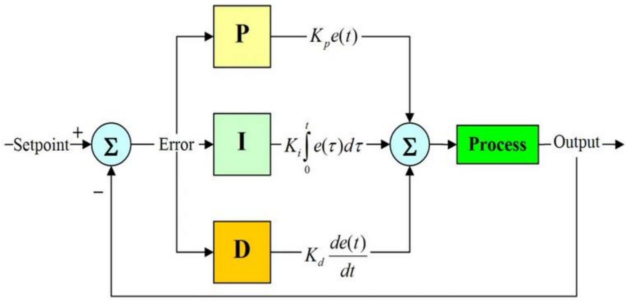


Fig. 3. PID controller

**Overview of methods:** There are a few strategies for tuning a PID loop. The foremost successful strategies by and large involve the advancement of a few processes model, at that point choosing P, I, and D based on the dynamic model parameters. Manual tuning strategies can be relatively efficient, especially in case the loops have a response time in the order of minutes or longer. The choice of strategy will depend to a great extent on whether the circle can be taken "offline" for tuning, and on the reaction time of the system. If the system can be taken offline, the most excellent tuning method frequently includes subjecting the system to a step alter in input, measuring the yield as a function of time, and utilizing this reaction to decide the control parameters. The PID controller design development has advanced from mechanical devices to digital devices, but the control algorithm is almost the same [9], [10]. In the time domain, the Input and Output equation of the PID controller, the transfer function is found by taking the Laplace transform of equation (6).

$$U(s) = K_p E(s) + K_i \frac{1}{s} E(s) + K_d s E(s) \tag{3}$$

In Addition, $E(s) = Y(s) - U(s)$ and the controller is equivalent to the equation given as:

$$U(s) = C(s)E(s) \tag{4}$$

Where $G(s)$ is the transfer function given as:

$$G(s) = \frac{x_\theta(s)}{\delta_e(s)} = \frac{18.2386}{s^2 + 0.4988s + 13.8636} \tag{5}$$

$$c(s) = \frac{\left(k_d s^2 + k_p s + k_i\right)}{s} \tag{6}$$

The combined system of $G(s)$ and $C(s)$ are regular functions that act on the tracking error $E(s)$ to obtain the output $Y(s)$. The closed-loop arrangement of the transfer function summarizes as:

$$Y(s) = \frac{c(s) * G(s)}{1 + c(s) * G(s)} R(s) \tag{7}$$

Substitute $G(s)$ and $u(t)$ “Equation 10” into Equation (12), the closed-loop transfer function from $Y(s)$ to $R(s)$ is expressed as:

$$\frac{Y(s)}{R(s)} = \frac{N(s)}{D(s)} \tag{8}$$

The PID Controller in MATLAB/Simulink uses a transfer function base model where $K_p$, $K_i$ and $K_d$ can be tune directly, for example:
$K_p$ = 1; $K_i$ = 1; $K_d$ = 1; S = tf ('s'); C = pid($K_p$, $K_i$, $K_d$).

Will generate a continuous-time PID controller in parallel form, like the one used in this simulation. Increasing the proportional gain ($K_p$ will proportionally increase the control signal for the same level of error. The fact that the controller will "push" harder for a given level of error tends to cause the closed-loop system to react more quickly, but also to overshoot more. Another effect of increasing $K_p$ is that it tends to reduce, but not eliminate, the steady-state error.

The derivative term $K_d$, adds the ability of the controller to anticipate error. With derivative control, the control signal can become large if the error begins sloping upward, even while the magnitude of the error is still relatively small. The addition of a derivative term, however, has no effect on the steady-state error.

## B. Adaptive Control Theory

After a vigorous study and analysis between adaptive and PID controller. It was clear that adaptive control was a more suitable and accurate controller for this project. Adaptive control is a branch of control theory that focuses on developing control strategies that can adjust and adapt to changes in the system dynamics or operating conditions. It aims to improve the performance of control systems by continuously updating controller parameters based on observed system behavior.

The basic principle behind adaptive control involves the following steps:

a) System Identification: Adaptive control starts with system identification, where the dynamics of the system are characterized and estimated. This typically involves collecting data from the system or conducting experiments to determine the system's mathematical model or its key parameters.

b) Parameter Estimation: In this step, the adaptive controller estimates or updates the parameters of the control system based on the collected data. This is often achieved using various estimation techniques, such as least squares estimation, recursive estimation, or Kalman filtering. The estimation process helps capture the dynamic variations and uncertainties in the system.

c) Controller Design and Adaptation: Once the system parameters are estimated, the adaptive controller adjusts its control action to achieve the desired performance. The controller continuously updates its parameters or gains based on the estimated system parameters and observed system behavior. This adaptation process allows the controller to respond and compensate for changes in the system, such as variations in system dynamics, external disturbances, or parameter uncertainties.

d) Feedback Loop: Adaptive control involves a feedback loop, where the output of the system is measured, compared to the desired output, and fed back to the controller. The difference between the desired output and the actual output, known as the error or the control signal, is used to update the controller parameters and adjust the control action.

e) Adaptation Laws and Algorithms: Adaptive control relies on adaptation laws or algorithms that govern how the controller parameters are updated. These laws are typically designed based on control theory principles, optimization techniques, or learning algorithms. The adaptation laws aim to minimize the difference between the desired and actual system performance over time.

f) Stability Analysis: Stability analysis is a crucial aspect of adaptive control. It ensures that the adaptive controller's parameter updates do not lead to instability or oscillations in the system. Stability analysis techniques, such as Lyapunov stability analysis, are used to verify the stability of the adaptive control system.

## C. Mathematical Modeling (Equation)

The equation used for the training of the controller so that the plant tracks the reference model is:

$$\frac{d^2 y_r}{dt^2} = -9y_r - 6\frac{dy_r}{dt} + 9r \tag{9}$$

The Neural Network controller has three inputs.

$$\dot{x} = A_x + B_u + f(x) \tag{10}$$

With Adaptive control the gain parameters of the controller are not allowed to be determined ahead of time. Instead, a learning mechanism is built into the controller, allowing the controller to constantly optimize the controller parameters to adapt to variations.
Discrete state space equation

$$x_{n+1} = Ax_n + Bu_n \tag{11}$$

$$y_n = Cx_n + Du_n \tag{12}$$

The general expression block $f(x)$: Parameters Expression:

$$(u(1) - mint) * \frac{2}{maxt - mint} - 1$$

$$(K * u) \quad is\ the\ Matrix\ \ gain$$

The Deep learning toolbox in MATLAB/Simulink has a block for control system, the model reference control block can be obtained from there. The Model reference control is an advanced built block from the basic model reference controller. For the basic adaptive control, the Reference model equation is:

$$\dot{x}_m = A_m X_m + B_m r \quad \rightarrow X_m \tag{13}$$

Plant:

$$\dot{x} = A_x + B_u \tag{14}$$

Real plant is:

$$u = k_r - k_x \tag{15}$$

$$\dot{x} = [A - Bk_x]x + Bk_r r \tag{16}$$

Solving for the state equation:

$$\dot{x}_m = A_m X_m + B_m r \quad set - equal\ state\ \&\ input\ matrices\ \dot{x} = [A - Bk_x]x + Bk_r r$$

Model Matching conditions

$$Am = A - Bk_x \quad (17)$$

$$Bm = Bk_r \quad (18)$$

All we need to do is to solve for the $Kx$ & $Kr$ that makes these two conditions true.

$$e = X - Xm \quad (19)$$

$$\dot{e} = \dot{X} - \dot{X}m = [A - Bk_x]x + Bk_r r - A_m X_m + B_m r \quad (20)$$

If we choose an eigenvalue that is stable, then the error will go to zero over time. The above equation is reduced to:

$$\dot{e} = Am\left(\dot{X} - \dot{X}m\right) = Am * e \text{ if Am} < \text{o then error} \rightarrow o\ over\ time$$

If error matching condition is meet, then the error will eventually go to zero. That means if we initialize each of these systems with different initial conditions, the error will eventually go to zero and the outputs will be the same. Since the are some uncertainties our equation of $f(x)$ becomes:

$$\dot{x} = A_x + B\big(u + f(x)\big) \quad (21)$$

Where the function $f(x)\ represent\ the\ uncertainty$ Which renders the conditions to be meet almost impossible? The uncertainty $f(x)$ makes picking the right value of $Kr$ and $Kx$ very difficult to closely match the close loop to the system. If we can cancel out that uncertainty, then we can be left back with our model. To cancel $f(x)$ we can add another loop.

$$w^T \emptyset(x) \quad (22)$$

$w^T$=is a vector of adaptive control weight, $\emptyset$=is a set of uncertainty model features.
If

$$w^T \emptyset(x)$$

Perfectly match f(x) in the following conditions is meet.

$$u = k_r r - k_x x - w^T \emptyset(x) \quad (23)$$

$$\dot{x} = Ax + B(B_r r - k_x x - w^T \emptyset(x) + f(x) \quad (24)$$

$$-w^T \emptyset(x) + f(x)\ will\ cancel\ if\ perfectly\ match$$

*Phi* $\emptyset(x)$ *is a set of bias functions or features, which can then be combine using the weighting vector* $w^T$. Once, the error matching condition is meet, the uncertainty will be cancel out, which will render the model suitable for application.

### D. Simulation

To develop a simulation of complex neural network, the principles of incorrect results encompass from insufficient model implementations and data examination strategies, deficiencies in workmanship. For us to reduce actuated equipment errors, simulation planning, setup, and execution strategies must be followed. The credibility, method such as verification and validation of the system is based on the synchronized graph of our final result with the result obtained from the CFD report. Our layout a thorough workflow determined from demonstrate confirmation and approved techniques for the conceivable Neuron network-controlled system. To demonstrate a formalized process, numerical precision is fundamentally approved against dynamics at network level. The choice of organized generation executed on the Neural network control for the leading-edge shock position targets the comparison between the CFD simulation to our maybe distinctive imperatives model of fixed-point control. Illustrating the esteem of computer program building strategies, such as refactoring, for confirmation task. Emphasis on comparing the neurons and arranging elements between the hidden layers and the execution outcomes, making some mathematical corrections within the model to fit numerical exactness is a vital strategy to increase system performance [10]-[13].

## IV. Result

The basic question for all modeling simulation is whether the demonstrated result gives an adequate exact representation of the system being studied. Assessing the outcomes of a neural network system is a non-trivial exercise which requires a thorough approval process. The term approval, or more by and large confirmation and validation also require an exact definition, as they have diverse implications in completely different settings. In program building, for case, confirmation and approval is the objective appraisal of items and forms all through the life cycle [14].

In specific, due to the observational challenges of neurobiology, spiking neural models are regularly not based on a particular natural network that might be considered "reality" and from which ground truth behavior can be recorded, in contrast to, for illustration, the airflow around a wing. The term "system of interest" recognizes that the method of confirmation and approval can moreover be connected to systems without concrete physical counterparts.

### A. Verification

Depicts the method of guaranteeing that the numerical demonstrate is fittingly represented by the executable demonstrate model. Model confirmation is the evaluation of a show execution. Neural network models are scientific models that are composed down in source code as numerical algorithms. The consistency and correlation of the output from the predictive simulation to that of the Computation Fluid Dynamic (CFD) report validated the interest of a best result. These validation results are based on the agreement between test information that characterizes the system of intrigued and the reenactment results. This assessment must take into thought the domain of intended application of the numerical demonstrate as well as its anticipated level of interest, since any model reflects the system of intrigued and as it were planning to coordinate to a certain degree and for certain endorsed conditions [15], [16].

### B. PID Simulated Result

A PID (integral-proportional-derivative) model is a well-established way of driving a system towards a target position or level. Objective is that the controller ensures that the process remains closer to its required or desired value regardless of various disruptions. The controller compares the process transmitter value (PV signal) values in the setpoint, based on that comparison the controller produces an output signal to operate the final control element.

The idea of using a PID as an alternative method is because, a similar procedure has been performed in controlling an industrial temperature according to Liu et al. Who state that the erroneous signals between the standard model results and the actual system outputs drove the system. Fig. 4 shows an overview of the model used in this simulation. The PID controller was applied with and without the adaptive controller for the LEST (leading edge shock train) control system. However, the result on the PID without adaptive control was out of speak, as a result comparison of the PID Controller with adaptive control and a Neural Network controller is shown below. The PID control response displays almost a strait horizontal line in comparison to the NN adaptive controller.

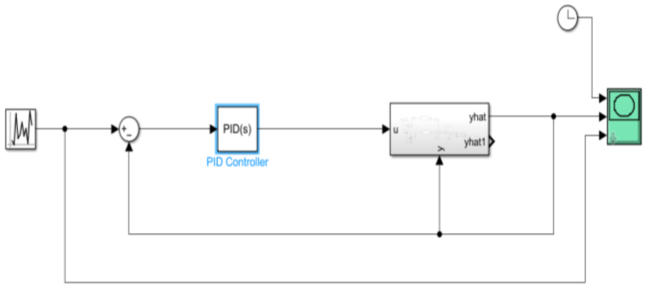


Fig. 4. PID overview

The above PID controller results without disturbance in the system (Fig. 5). To properly analyze the behavior of the controller, a disturbance with the following parameters was introduced in the system. Parameter; Amplitude: 0.001 to 0.05, Frequency: 1 to 5.

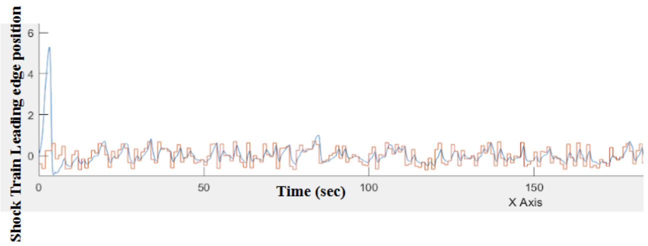


Fig. 5. Simulated result of a PID Controller without disturbance. Red: Desired leading edge shock position. Blue: Actual leading edge shock position

Fig. 6 illustrates a detailed Simulink block diagram that incorporates the introduction of a disturbance into the system. This block diagram provides a visual representation of the various components and their interconnections, enabling a comprehensive understanding of the system's structure and functionality. The deliberate inclusion of the disturbance allows for an in-depth analysis of how the system behaves in the presence of external factors.

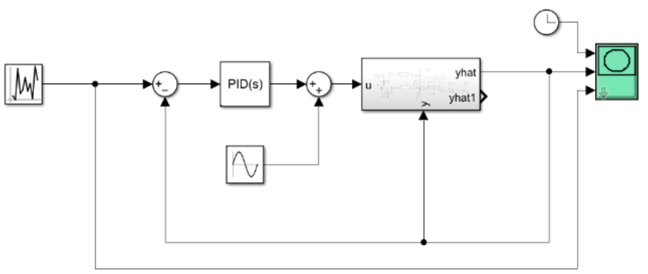


Fig. 6. PID overview with disturbance in the system

In Fig. 7, the simulated results of the system's response to the introduced disturbance are presented. The outcome reveals that the PID controller struggles to accurately track the required trajectory of the leading-edge shock trend. The control performance is notably compromised when compared to a PID system operating without any disturbances. This discrepancy in performance can be attributed to the disruptive influence of the introduced disturbance, which poses challenges for the controller to maintain the desired trajectory.

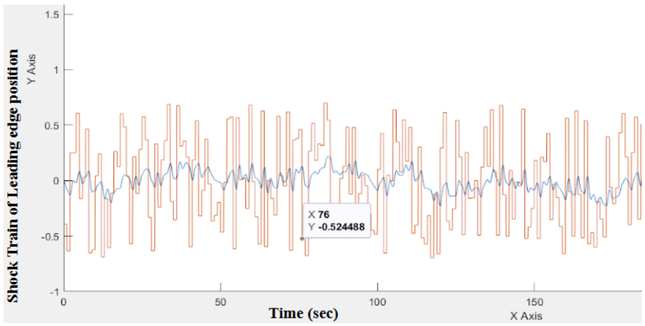


Fig. 7. Simulated result of a PID controller with disturbance. Red: Desired leading edge shock position. Blue: Actual leading edge shock position

The simulated result serves as a valuable insight into the system's limitations and the impact of disturbances on its performance. It highlights the need for further improvement in the control strategy or the incorporation of additional compensatory measures to enhance the system's ability to mitigate disturbances effectively. This observation underscores the importance of developing robust control mechanisms that can adapt to external disturbances and maintain stable operation, ensuring better overall system performance.

## C. Adaptive Control

The first step in the neural network process is to approximate the neural network's plant identification behavior. The goal is to apply the plant identification process to allow the neural network to train itself to model the plant's output. Secondly, it is necessary to identify the plant before the controller is trained. The three layers with 10 neurons in the hidden layer are utilized as NN architecture. Two delayed inputs $u(k-1)$, $u(k-2)$ and two delayed outputs $y(k-1)$, $y(k-2)$ with a sampling period of 0.05 of the actual plant are used as input to capture the system dynamics [12]. The simulation result shown in Fig. 8 is based on data-set case-2 of the CFD report, with focus on data set just on the transient state 293 samples of input [17].

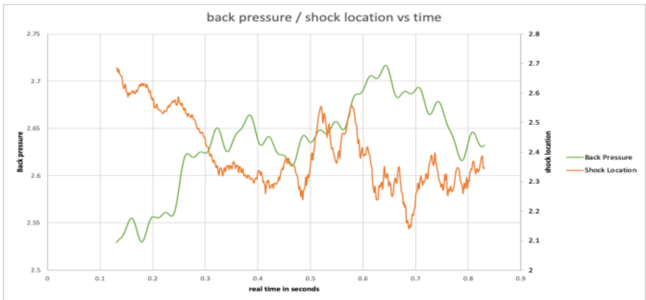


Fig. 8. CFD simulation data obtained

Fig. 8 above shows the graph of the data received from a computational fluid dynamic needed for this simulation. The first part of the simulation is to train the reference model. In this model, about half of the dataset, 293 to be precise, was used to determine system accuracy. Focus will be only on the transient part of the data shown in the simulation. Maximum/minimum with the size of 0.7/-0.7 radian are used for the neural network plant model training [10].

The remarkable performance of the Neural Network (NN) controller is evidenced by the minute degree of error observed in both the Plant output and the Neural Network output (shown in Fig. 9 for detail Simulink block, Fig. 10 for testing data, Fig. 11 for validation data, Fig. 12 for training data, and Fig. 13 for training validation of reference control model). The range of error, spanning from 0.03 to -0.02, highlights the exceptional accuracy with which the NN controller has learned and replicated the dynamic behavior of the system.

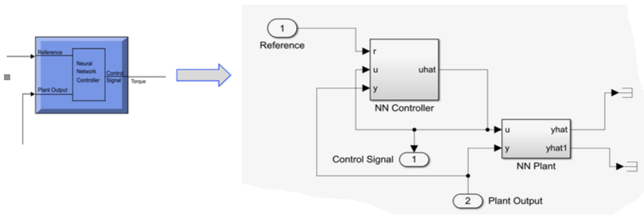


Fig. 9. Detail of simulink block

Analysis the error, difference between the CFD data and the neural network model output.

The neural network (NN) controller represents a remarkable advancement in control systems, enabling the leading-edge shock train to reach arbitrary objectives (Fig. 14, Fig. 15, and Fig. 17). This sophisticated controller possesses the unique capability to adapt and adjust dynamically in the presence of various disturbances. The degree of error, when comparing the plant out and the NN output, is very minimal. In a study conducted by Gao et al. [13], it was demonstrated that NN controllers are specifically designed to deliver robust closed-loop efficiency. These controllers excel in achieving precise error tracking, minimizing deviations between the desired trajectory and the actual response, while also imposing restrictions on the control inputs.

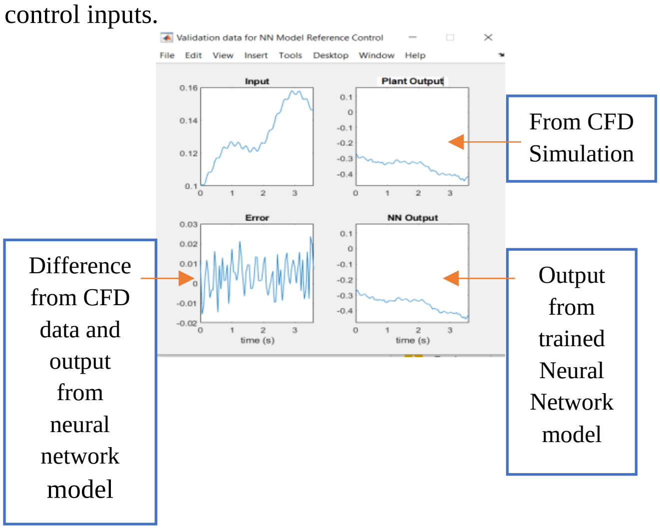


Fig. 10. Testing data

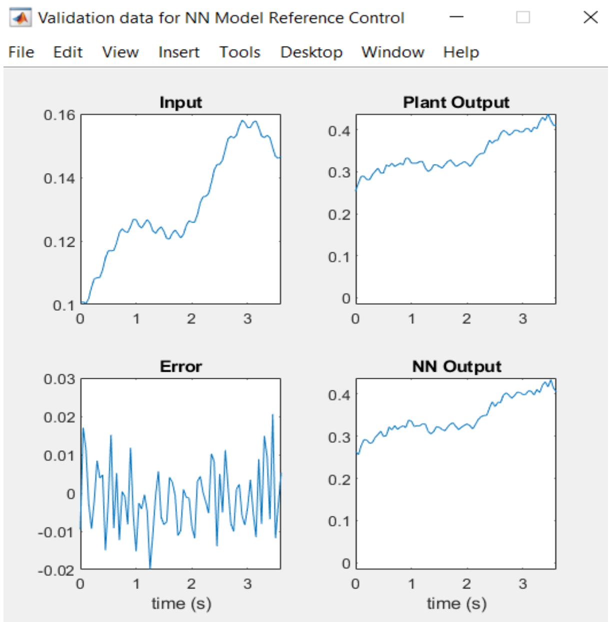


Fig. 11. Validation data

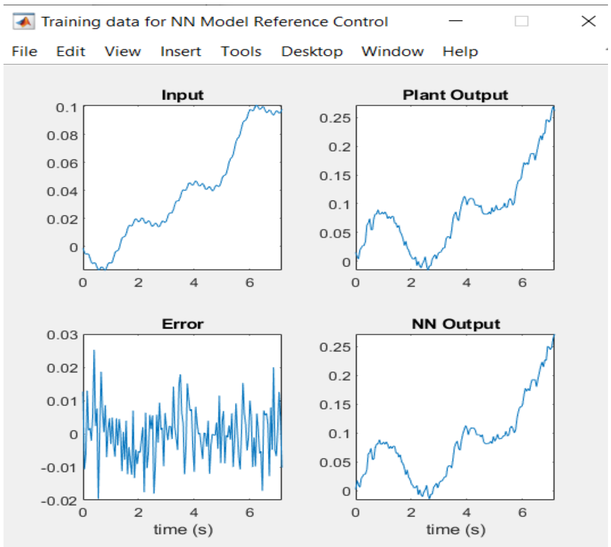


Fig. 12. Training data

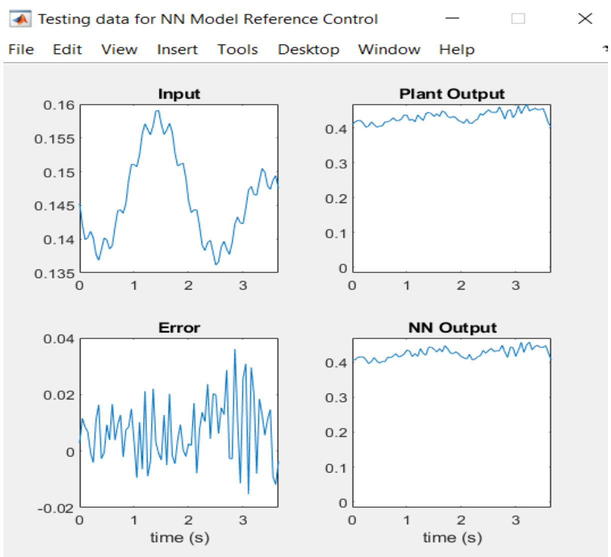


Fig. 13. Training validation of reference control model

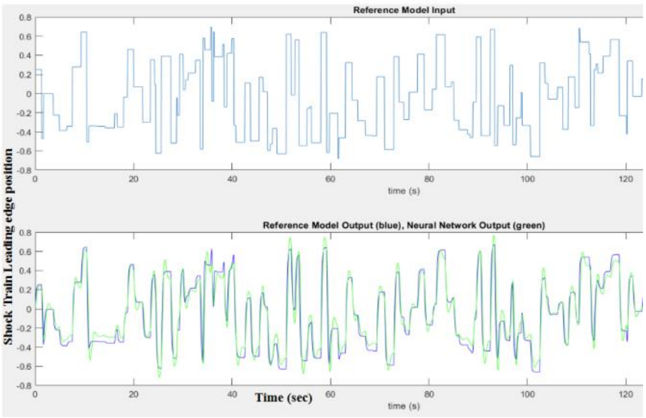


Fig. 14. Neural network simulated test result

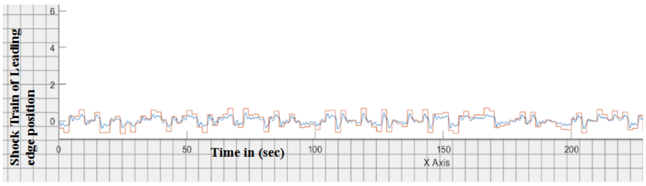


Fig. 15. Neural network adaptive plant result without disturbance

The ability to maintain such tight control allows for enhanced precision and stability in diverse applications.

The Neural Network output, indicated by the green line, exhibits a significant correlation with the reference model output represented by the blue line (Fig. 14). This finding not only validates the effectiveness of the neural network but also provides promising evidence that the system can operate in accordance with the desired specifications. The strong alignment between the Neural Network output and the reference model output underscores the accuracy and reliability of the proposed system. This encouraging result signifies that the neural network has effectively learned and captured the underlying patterns and dynamics of the system, enabling it to generate outputs that closely align with the expected behavior. Such a positive outcome instills confidence in the potential of the system, suggesting that it has the capability to perform as intended and meet the desired objectives.

To conduct a comprehensive analysis of the controller's behavior, a deliberate introduction of a disturbance with specific parameters was implemented in the system (Fig. 16). This strategic approach aims to evaluate how the controller responds and adapts in the presence of external influences or perturbations. By introducing a controlled disturbance, the research team can gain valuable insights into the system's stability, robustness, and overall performance under challenging conditions.

The selected disturbance parameters were carefully chosen to represent real-world scenarios and encompass a range of potential challenges that the controller might encounter during operation. This meticulous examination enables a thorough understanding of the controller's capability to mitigate disturbances and maintain optimal performance, ultimately contributing to the refinement and enhancement of the system's overall design and functionality. Parameter; Amplitude: 0.001 to 0.05, Frequency: 1 to 5.

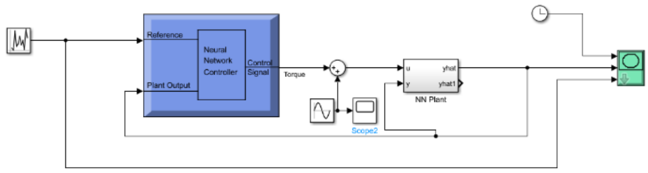


Fig. 16. Simulink block with disturbance introduce in the system

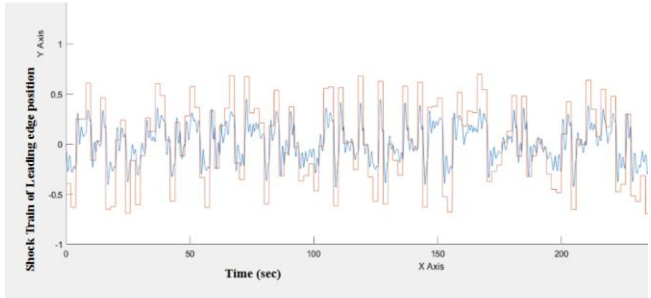


Fig. 17. Neural network adaptive plant result with disturbance

An adaptive controller and a PID (Proportional-Integral-Derivative) controller are both widely used control strategies in engineering and industrial automation (Table 2). While they both aim to regulate a system's output, they differ in their approaches and capabilities. The controllers' advantages and disadvantages listed below, with the simulated result obtain help in selecting the right controller for this research.

Robustness: PID Controller: While PID controllers can provide satisfactory performance for many systems, they may struggle to handle significant parameter variations or disturbances. They are not inherently robust to changes in system dynamics.

Adaptive Controller: Adaptive controllers are designed to handle parameter variations and uncertainties. They can identify changes in the system's dynamics and adjust their control strategy accordingly. This adaptability enables them to maintain performance even in the presence of unknown or varying system parameters.

Implementation Complexity: PID Controller: PID controllers are relatively straightforward to implement and require minimal computational resources. The main complexity lies in tuning the controller parameters correctly.

Adaptive Controller: Adaptive controllers can have higher implementation complexity, especially when advanced identification or adaptation algorithms are used. They may require more computational resources and expertise in system identification techniques.

Performance: PID Controller: PID controllers can provide satisfactory control performance for many systems, especially when the system dynamics are well-understood and relatively constant. However, they may struggle to achieve optimal performance in the face of varying or uncertain system parameters.

Adaptive Controller: Adaptive controllers excel in situations where system dynamics change over time or have uncertain parameters. They can adapt their control strategy to optimize performance and maintain stability, even in the presence of disturbances or parameter variations.

Table 2. Conclusion and future work

| Objectives | Conclusion | Limitation |
|---|---|---|
| Model Neural Network Control | • The Adaptive controller was the main selected controller for this research.<br>• The PID controller was the alternative method of control. | • Due to its robust features, adaptive controller requires more time to optimize than a PID controller. |
| Design and implement the model. | • The approach of Modeling and manual turning of the PID controller was very efficient.<br>• The adaptive controller was model based on the optimized system which is an upgrade of the basic adaptive control. **Design AC:** Reference Model, NN Plant model, NN Controller, and Plant | • More sophisticated algorithms might also be used to bust the performance of the Adaptive control.<br>• The PID Controller was able to closely control the shock train to the desired location. But system optimization is limited. |
| Analyses and optimize the model to work accordingly | • The Adaptive Controller is efficient in tackling the system uncertainty and adapting to the system control strategy. | • In the result obtained, the Adaptive controller was able to track the desired leading-edge location, even with disturbance introduced in the system. |

## V. CONCLUSION

Many approaches for boosting the effectiveness of the controller, including PID controller and IMC (internal model control), have been developed to assure the controller's resilience and stability. Various decoupling control schemes, including INA (inverse Nyquist array), feedforward decoupled control, and reverse-biased decoupling regulation, have been tested to cancel the coupling effects between each stream of the systems.

A model-based NN control approach paired with the PID (integral-proportional-derivative) controller was used as an alternative to control the Leading-edge shock train but the chosen parameters for the model did not provide better signals between the standard model results and the actual system outputs derived. The approach used involved applying the NN controller to the end of the PID controller, changing the control input of each channel suitably. The NN control approach both enhances and proves to be the best fit for the project without the incorporation of the PID, reprograms dynamic response and contents of distinct channels in a steady and transient state. The method's controlling effectiveness is effectively verified. The combination of input and instructional signals for NN control training in the system is necessary for future research. Furthermore, with some system optimization, the simulated results could be improved to a more accurate and precise fit between the required and the actual position of the LEST [18]-[20].

## ACKNOWLEDGMENT

This research is funded by the ONR (Award No. N00014-22-1-2724). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. I would like to extend my sincere appreciation to everyone involved in the research projects, as well as the entire team at North Carolina A&T State University (NCAT) and TCASA. The knowledge and skills I have acquired during my time at the lab will undoubtedly have a lasting impact on my career. The dedication and expertise exhibited by the lab's personnel have inspired me and further fueled my passion for research and development. To the Air Force research Lab, thank you for the invaluable experience and support.

## REFERENCES

[1] L. Vanstone, A. Bosco, Y. Saleh, M. Akella, N. T. Clemens, and S. Gogineni, "Closed-loop control of unstart in a Mach 1.8 isolator," *Journal of Propulsion and Power*, vol. 36, no. 1, pp. 153-157, 2020, https://doi.org/10.2514/1.B37444.

[2] J. R. Hutzel. *Scramjet isolator modeling and control*. Air Force Institute of Technology, 2011, https://scholar.afit.edu/etd/1050/.

[3] O. Boucher and M. Weeks. *Introduction to MATLAB & SIMULINK (A Project Approach)*. Laxmi Publications Ltd., 2008, https://cir.nii.ac.jp/crid/1130000794495812096.

[4] F. Ferguson, Y. Gao, D. Feng, M. D. Atkinson, and J. C. Mendez, "Investigating the Unsteady Fluid Physics within Complex Flows Fields," in *AIAA Scitech 2021 Forum*, p. 0750, 2021, https://doi.org/10.2514/6.2021-0750.

[5] M. A. Muktadir and S. Yi, "Machine Vision-Based Detection of Surface Defects of 3D-Printed Objects," in *2021 ASEE Virtual Annual Conference Content Access*, 2021, https://doi.org/10.18260/1-2--37471.

[6] M. T. Mathews. *Adaptive and Neural Network Based Control of Unmanned Aerial Vehicles*. Doctoral dissertation, North Carolina Agricultural and Technical State University, 2021, https://www.proquest.com/openview/b38cf1dc8d15f5f7d1adad663076e6e0/1?pq-origsite=gscholar&cbl=18750&diss=y.

[7] M. Chen, U. Challita, W. Saad, C. Yin and M. Debbah, "Artificial Neural Networks-Based Machine Learning for Wireless Networks: A Tutorial," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3039-3071, 2019, https://doi.org/10.1109/COMST.2019.2926625.

[8] S. H. Hein, G. Lee, Y. Tashima, A. M. A. Elnaiem, and S. Nagatomo, "Feedback Control for Tillage Tool Angle of Home-Garden Robot," *2018 IEEE 7th Global Conference on Consumer Electronics (GCCE)*, pp. 293-294, 2018, https://doi.org/10.1109/GCCE.2018.8574606.

[9] W. H. Heiser and D. T. Pratt. *Hypersonic Airbreathing Propulsion*. American Institute of Aeronautics and Astronautics, 1994, https://books.google.co.id/books?id=d1sQvT2_kMsC&.

[10] A. Afram, F. Janabi-Sharifi, A. S. Fung, and K. Raahemifar, "Artificial neural network (ANN) based model predictive control (MPC) and optimization of HVAC systems: A state of the art review and case study of a residential HVAC system," *Energy and Buildings*, vol. 141, pp. 96-113, 2017, https://doi.org/10.1016/j.enbuild.2017.02.012.

[11] H. S. Dakheel, Z. B. Abdullah, N. S. Jasim, and S. W. Shneen, "Simulation model of ANN and PID controller for direct current servo motor by using Matlab/Simulink," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 20, no. 4, pp. 922-932, 2022, http://doi.org/10.12928/telkomnika.v20i4.23248.

[12] J. Verberne. *Development of Robust Control Laws for Disturbance Rejection in Rotorcraft UAVs*. Master's Thesis, Embry-Riddle Aeronautical University Scholarly Commons, 2019, https://commons.erau.edu/edt/447/.

[13] H. Alimohammadi, B. B. Alagoz, A. Tepljakov, K. Vassiljeva, and E. Petlenkov, "A NARX model reference adaptive control scheme:

improved disturbance rejection fractional-order PID control of an experimental magnetic levitation system," *Algorithms*, vol. 13, no. 8, p. 201, 2020, https://doi.org/10.3390/a13080201.

[14] H. Gao, W. He, C. Zhou and C. Sun, "Neural Network Control of a Two-Link Flexible Robotic Manipulator Using Assumed Mode Method," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 755-765, 2019, https://doi.org/10.1109/TII.2018.2818120.

[15] M. Muktadir *et al*., "Uncrewed Ground Vehicles (UGVs) and Nature-Inspired Designed Robot DIGIT and SPOT: A Review," *American Journal of Engineering and Applied Sciences*, vol. 15, no. 4, pp. 274-287, 2022, https://doi.org/10.3844/ajeassp.2022.274.287.

[16] S. C. Dekkata, S. Yi, M. A. Muktadir, S. Garfo, X. Li, and A. A. Tereda, "Improved Model Predictive Control System Design and Implementation for Unmanned Ground Vehicles," *Journal of Mechatronics and Robotics*, vol. 6, pp. 90-105, 2022, https://doi.org/10.3844/jmrsp.2022.90.105.

[17] S. Garfo, M. A. Muktadir, and S. Yi, "Defect detection on 3D print products and in concrete structures using image processing and convolution neural network," *Journal of Mechatronics and Robotics*, vol. 4, no. 1, pp. 74-84, 2020, https://doi.org/10.3844/jmrsp.2020.74.84.

[18] R. Karagoz and K. Batselier, "Nonlinear system identification with regularized Tensor Network B-splines," *Automatica*, vol. 122, p. 109300, 2020, https://doi.org/10.1016/j.automatica.2020.109300.

[19] S. C. Dekkata, S. Yi, M. A. Muktadir, and S. Garfo, "LiDAR-Based Obstacle Detection and Avoidance for Navigation and Control of an Unmanned Ground Robot Using Model Predictive Control," *Journal of Mechatronics and Robotics*, vol. 7, no. 1, pp. 27-41, 2023, https://doi.org/10.3844/jmrsp.2023.27.41.

[20] R. Ivanov, T. J. Carpenter, J. Weimer, R. Alur, G. J. Pappas, and I. Lee, "Case study: verifying the safety of an autonomous racing car with a neural network controller," *Proceedings of the 23rd International Conference on Hybrid Systems: Computation and Control*, pp. 1-7, 2020, https://doi.org/10.1145/3365365.3382216.