ARTICLE INFO
Article ID: 01-16-03-0019
© 2023 Seyed Hashemi and Ruxandra Botez
doi:10.4271/01-16-03-0019

A Novel Flight Dynamics Modeling Using Robust Support Vector Regression against Adversarial Attacks

Seyed Mohammad Hashemi[1] and Ruxandra Mihaela Botez[1]

[1]University of Quebec, École de technologie supérieure, Canada

Abstract

An accurate Unmanned Aerial System (UAS) Flight Dynamics Model (FDM) allows us to design its efficient controller in early development phases and to increase safety while reducing costs. Flight tests are normally conducted for a pre-established number of flight conditions, and then mathematical methods are used to obtain the FDM for the entire flight envelope. For our UAS-S4 Ehecatl, 216 local FDMs corresponding to different flight conditions were utilized to create its Local Linear Scheduled Flight Dynamics Model (LLS-FDM). The initial flight envelope data containing 216 local FDMs was further augmented using interpolation and extrapolation methodologies, thus increasing the number of trimmed local FDMs of up to 3,642. Relying on this augmented dataset, the Support Vector Machine (SVM) methodology was used as a benchmarking regression algorithm due to its excellent performance when training samples could not be separated linearly. The trained Support Vector Regression (SVR) predicted the FDM for the entire flight envelope. Although the SVR-FDM showed excellent performance, it remained vulnerable to adversarial attacks. Hence, we modified it using an adversarial retraining defense algorithm by transforming it into a Robust SVR-FDM. For validation studies, the quality of predicted UAS-S4 FDM was evaluated based on the Root Locus diagram. The closeness of predicted eigenvalues to the original eigenvalues confirmed the high accuracy of the UAS-S4 SVR-FDM. The SVR prediction accuracy was evaluated at 216 flight conditions, for different numbers of neighbors, and a variety of kernel functions were also considered. In addition, the regression performance was analyzed based on the step response of state variables in the closed-loop control architecture. The SVR-FDM provided the shortest rise time and settling time, but it failed when adversarial attacks were imposed on the SVR. The Robust-SVR-FDM step response properties showed that it could provide more accurate results than the LLS-FDM approach while protecting the controller from adversarial attacks.

History

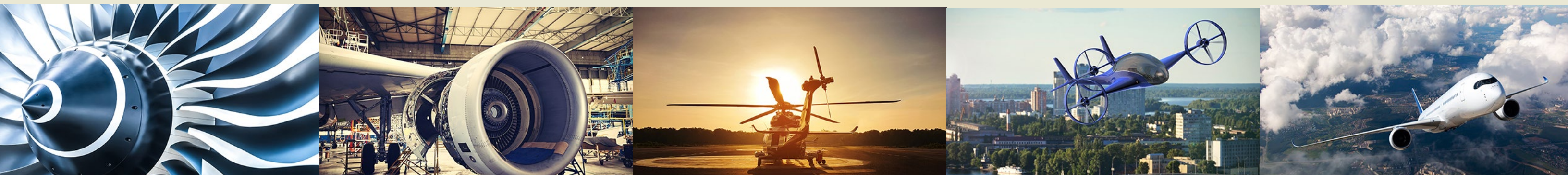
Received: 29 Nov 2021
Revised: 29 Oct 2022
Accepted: 01 Mar 2023
e-Available: 24 Mar 2023

Citation

Hashemi, S. and Botez, R., "A Novel Flight Dynamics Modeling Using Robust Support Vector Regression against Adversarial Attacks," *SAE Int. J. Aerosp.* 16(3):305-323, 2023, doi:10.4271/01-16-03-0019.

ISSN: 1946-3855
e-ISSN: 1946-3901

This article is part of a special Issue on Uncrewed Aircraft Systems (UAS) and Autonomy.



1. Introduction

Unmanned Aerial Systems (UASs) have been successfully utilized mainly for surveillance and reconnaissance [1, 2]. The fast-growing demand for UASs highlights the need for special attention to the safety and efficiency of such critical systems [3]. According to the aviation transportation industry, the advancement of Flight Dynamics Modeling (FDM) and control is one of the most important factors needed to improve the safety and efficiency of UASs [4, 5]. Hence, our goal is to design such an intelligent algorithm for accurate FDM that improves the corresponding controller performance.

The UAS-S4 Ehecatl (designed and manufactured by the company Hydra Technologies) was utilized to evaluate and experimentally validate the developed FDM algorithm and controller [6, 7]. This UAS is equipped with elevators, ailerons, and rudders that control moments around the pitch, roll, and yaw axes. Figure 1 shows Hydra Technologies' UAS-S4 Ehecatl, and Table 1 lists its main specifications.

An Flight Dynamics Model (FDM) is a group of mathematical equations that are used for calculating the influence of the main loads (forces and moments) on an aircraft, such as lift, drag, and thrust. The three main flight dynamics parameters are the pitch, roll, and yaw angles of rotation around the UAS center of gravity. In this context, the "model" refers to the mathematical representation of the UAS-S4 FDM. UAS modeling is crucial, as its FDM is used for the design and development of a flight dynamics controller. Flight dynamics control is a process that aim to stabilize the aircraft flight dynamics. Having access to a UAS-S4 FDM enhances our ability to evaluate its controller performance in the early stage of its development which allows us to increase its flight safety while reducing flight costs [8, 9, 10, 11].

The UAS-S4 FDM was obtained based on 216 flight cases [12, 13, 14, 15] using the equipment including a UAS-S4, a wind tunnel [6], and a Level D research aircraft flight simulator [16]. The flight envelope consisted of 216 local FDMs, where each FDM was designed for a specific range of altitude, speed, and mass. The trimmed local FDMs were represented based on the state-space system, linearized around their corresponding equilibrium points. Given that the accuracies of local FDMs decrease with the increasing distance with respect to their equilibrium points, a proper regression can thus provide a more precise FDM for the entire flight envelope [17].

FIGURE 1 Hydra Technologies UAS-S4 Ehecatl.

TABLE 1 The UAS-S4 geometrical and flight data specification.

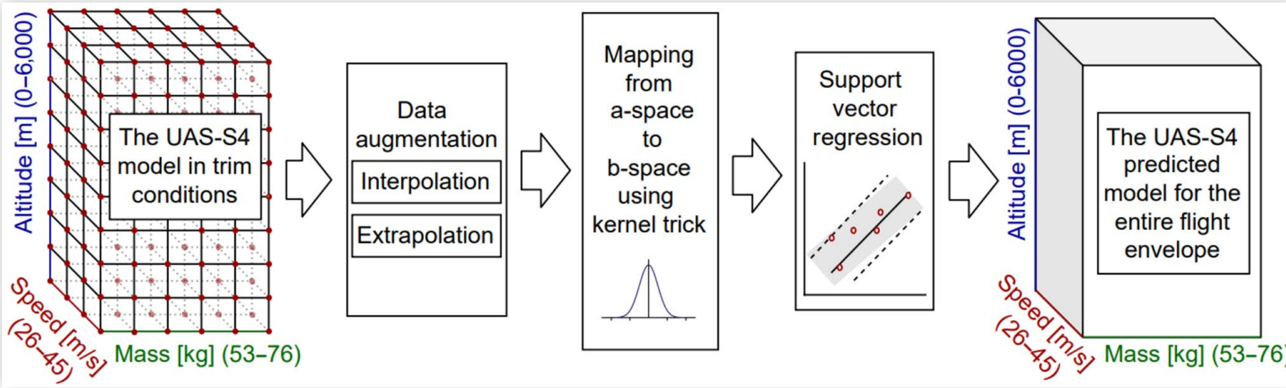
Specification	Value
Wingspan	4.2 m
Wing area	2.3 m^2
Total length	2.5 m
Mean aerodynamic chord	0.57 m
Empty weight	50 kg
Maximum take-off weight	80 kg
Loitering airspeed	35 knots
Maximum speed	135 knots
Service ceiling	15,000 ft
Operational range	120 km

For regression problems, data-driven algorithms have shown great efficiency in a variety of applications [18, 19, 20, 21]. Many studies formulated the FDM as a regression problem in order to improve FDM accuracy. State-of-the-art algorithms with deep learning architectures such as Recurrent Neural Network [22] (RNN), Long Short-Term Memory [23] (LSTM), and Convolutional Neural Network [24] (CNN) were utilized for FDM. The CNN can automatically extract the important features of the FDM without learning supervision. The RNN and LSTM can perform accurately if there exists a correlation among FDM samples. Despite their excellent performance in terms of regression accuracy, none of them can outperform the Support Vector Regression (SVR) while approaching adversarial FDM samples.

Data-driven algorithms have disadvantages related to their high dependency on a large quantity of training data. For data-driven regression using small-sized datasets, training data augmentation is required [25, 26]. With this aim, "interpolation" and "extrapolation" are the most well-known practical methods for data augmentation, and both of them operate based on the k-nearest neighbors principle [27]. Relying on an augmented dataset, data-driven algorithms can safely be employed for FDM regression. In this study, we utilized the SVR methodology as the benchmarking algorithm [28] because it has excellent ability when training samples cannot be separated linearly [29]. Figure 2 shows this procedure for developing the UAS-S4 FDM.

As shown in Figure 2, A flight envelope containing 216 local state-space FDMs corresponding to various trim flight conditions (red nodes) for 9 altitudes (0–6,000 m), 6 masses (53–76 kg), and 4 speeds (26–45 m/s) was constructed, and then the number of local FDMs was increased from 216 to 3,642 using interpolation and extrapolation methodologies in the data augmentation block. Next, kernel functions were used to map the data from a low-dimensional space into a high-dimensional space in which the data were linearly

FIGURE 2 The procedure to obtain the UAS-S4 FDM for its entire flight envelope.



separated using support vectors. Finally, the trained SVR predicted the UAS-S4 FDM for the entire flight envelope.

Although the need for a large-scale dataset (a minor challenge) can be solved through data augmentation, security attacks (a major challenge) are potential threats to data-driven regression algorithms [30]. In fact, it is possible to mislead a neural network–based FDM using adversarial data, generated by carefully manipulating original data. This deception operation is known as an "Adversarial Attack" in the Artificial Intelligence (AI) community [31]. In our work, in the case of adversarial attacks, the SVR provided a wrong local FDM for the controller in the supposed flight conditions. The controller then generates incorrect commands (for the control surfaces) which result in the UAS instability and collapse [32]. Development of an effective defense algorithm against adversarial attacks on SVR-based FDMs becomes therefore a priority for reliable UASs.

There are four main original contributions of our research article:

1. The design of a data augmentation procedure to prepare a large-scale dataset for FDM using *k*-nearest-based interpolation and extrapolation.
2. The design and testing of an SVR algorithm for accurate FDM when training samples could not be linearly separated.
3. A realistic hacking of a UAS controller was obtained by imposing adversarial attacks on its neural network–based FDM, in which a small perturbation was injected into the SVR output layer, that resulted in wrong FDM prediction, thus in wrong control commands computation.
4. The design of an effective defense algorithm as a countermeasure in order to make the FDM and controller robust against adversarial attacks.

This article is composed of five sections beginning with the introduction. The UAS-S4 is modeled in Section 2 using scheduled local FDMs (under different flight conditions), which are then augmented through interpolation and extrapolation procedures. The SVR model was designed, fine-tuned, and trained for the UAS-S4 FDM regression in order to predict its FDM in any flight condition. Section 3 introduces the concept of adversarial attacks and discusses it from both white-box and black-box aspects. After imposing an adversarial attack on the UAS-S4 SVR-based FDM, we show how to render it robust by designing a defense algorithm using adversarial samples. Section 4 presents the results and shows the improved UAS-S4 FDM accuracy obtained by relying on the SVR and details the SVR sensitivity to adversarial attacks. This section also evaluates the efficiency of the defense algorithm against adversarial attacks based on the performance of a controller for the UAS-S4 FDM. Finally, a comprehensive conclusion and the opportunities for future works are discussed in Section 5.

2. UAS-S4 FDM

A. Local Linear Scheduling Flight Dynamics Model

The Finite Element Method is a common and well-known methodology for solving differential equations in engineering [33]. This methodology has been utilized for various mathematical modeling [34, 35], including FDM. By considering the UAS-S4 differential equations of lateral and longitudinal motion [36], its flight dynamics can be linearly modeled around its equilibrium points for 216 flight conditions. The state variables including the axial velocity u, vertical velocity w, pitch rate q, and pitch angle θ constitute the longitudinal state vector $X_{lon} = [u\ w\ q\ \theta]^T$, where X_{lon} is controlled by the longitudinal input vector $\delta_{lon} = [\delta_e\ \delta_T]^T$, which consists of the elevator deflection δ_e and thrust δ_T. The state variables including the lateral velocity v, roll rate p, yaw rate r, and roll angle φ constitute the lateral state vector $X_{lat} = [v\ p\ r\ \varphi]^T$, which is controlled by the lateral input vector $\delta_{lat} = [\delta_a\ \delta_r]^T$, composed of the aileron and rudder deflections, respectively.

For any trim condition, the UAS-S4 FDM can be represented based on the following state-space system by utilizing the state vector X, input vector δ, and output vector Y [37]:

$$\dot{X}(t) = AX(t) + B\delta(t)$$
$$Y(t) = CX(t) + D\delta(t) \quad \text{Eq. (1)}$$

where the longitudinal state-space matrices are

$$A_{lon} = \begin{bmatrix} G_u & G_w & 0 & -g\ \cos\ \theta_0 \\ H_u & H_w & u_0 & -g\ \sin\ \theta_0 \\ M_u + M_{\dot{w}} H_u & M_w + M_{\dot{w}} H_w & M_q + u_0 M_{\dot{w}} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$
$$B_{lon} = \begin{bmatrix} G_{\delta_e} & G_{\delta_T} \\ H_{\delta_e} & H_{\delta_T} \\ M_{\delta_e} + M_{\dot{w}} H_{\delta_e} & M_{\delta_T} + M_{\dot{w}} H_{\delta_T} \\ 0 & 0 \end{bmatrix}, \quad C_{lon} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}^T, \quad D_{lon} = 0 \quad \text{Eq. (2)}$$

and the lateral state-space matrices are

$$A_{lat} = \begin{bmatrix} Y_v & Y_p & -(u_0 - Y_r) & g\ \cos\ \theta_0 \\ L_v & L_p & L_r & 0 \\ N_v & N_p & N_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$
$$B_{lat} = \begin{bmatrix} 0 & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \end{bmatrix}, \quad C_{lat} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}^T, \quad D_{lat} = 0 \quad \text{Eq. (3)}$$

Dimensional stability derivatives were explained [13, 14] based on a variety of aerodynamics [38, 39, 40] and propulsion [12, 41] analyses. Based on the above modeling approach, the UAS-S4 FDM was obtained using 216 local linear state-space representations that were named the Local Linear Scheduling (LLS) FDM [42, 43]. Each state-space FDM expressed the linearized state variables around a specific equilibrium point corresponding to a certain combination of altitudes, speeds, and masses. To obtain a large number of local linear FDMs, the use of data augmentation methodologies is required.

B. Data Augmentation

In AI, data augmentation refers to a procedure to generate synthetic data that are used for training data-driven algorithms to improve their performance [44]. While there is a high number of data-augmentation techniques, interpolation and extrapolation are considered the most well-known practical methods for solving regression problems.

Figure 3 shows the process of augmentation of local FDMs using interpolation methodology based on the k-nearest neighbors.

Based on its three nearest neighbors, the centroid of the three local FDMs associated with the closest operating points embedding is computed. The new local FDM can then be generated by interpolation between the centroid and the FDM corresponding to its original operating point. Equation 4 represents this generation process [27]:

$$\bar{Z}_j = (Z_k - Z_j)\lambda_{int} + Z_j \quad \text{Eq. (4)}$$

where Z_k, Z_j, and $\bar{Z}_j$ denote the computed centroid, the original, and the new local FDM embedding, respectively. λ_{int} is an adjustable factor for tuning the interpolation degree.

Following a similar methodology, data can be augmented using extrapolation. Figure 4 demonstrates the process for augmenting local FDMs using an extrapolation methodology based on the k-nearest neighbors.

Using the three nearest neighbors, the centroid of the three local FDMs related to the closest operating points embedding is computed. The new local FDM can then

FIGURE 3 Augmenting local FDMs using an interpolation methodology based on the k-nearest neighbors (data taken from [27]).

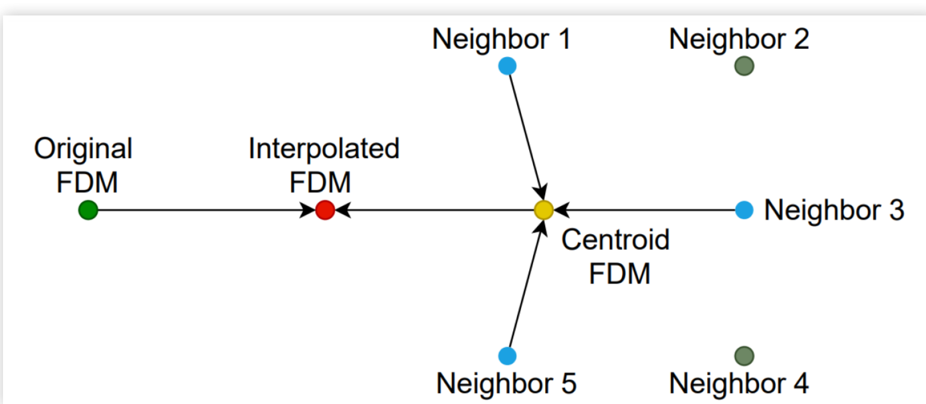
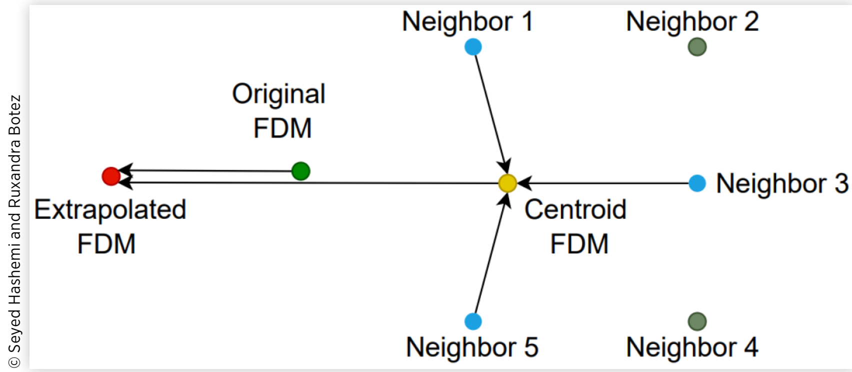


FIGURE 4 Augmenting local FDMs using an extrapolation methodology based on the k-nearest neighbors (data taken from [27]).



be generated via an extrapolation between the archived centroid value and the FDM related to the original operating point, as represented in Equation 5 [27].

$$\bar{Z}_j = \left(Z_j - Z_k\right)\lambda_{ext} + Z_j \qquad \text{Eq. (5)}$$

It should be mentioned that λ needs to be selected carefully within its valid interpolation and extrapolation ranges, as $\lambda_{in} \in [0, 1]$ and $\lambda_{ex} \in [0, \infty)$.

By use of the abovementioned data augmentation techniques, the number of training datasets was increased from 216 to 3,642. Based on the new augmented dataset, an improved regression algorithm can provide a more accurate (compared to the use of a smaller dataset) FDM for any flight condition in the flight envelope.

C. Support Vector Regression of a Flight Dynamics Model

There is no doubt regarding the excellent effectiveness of data-driven predictors in regression tasks when a large dataset is provided [45]. However, small-sized datasets can be successfully enlarged using data augmentation methodologies. Relying on the augmented dataset, we utilized SVR as the benchmarking algorithm. The SVM is one of the most powerful classifications and regression algorithms. Regarding its attack robustness, skillful generalization capability, and user-friendly decision model modification, the SVR is considered an outstanding benchmark. It is worth noting that while the performance of an SVR depends on the dataset characteristics and its sample distributions, its computational complexity is not affected by the number of input variables or features for a dataset (input dimensionality) [46]. Besides, the SVR performs the regression task favorably when training samples cannot be separated linearly [47]. Hence, the SVR is employed for capturing data distribution.

This conventional regressor is based on the Support Vector Machine (SVM) principle, which is known as deserving "high-dimensional space" learning algorithm [48]. Basically, the SVR is one of the nonlinear regression algorithms, as its inherent kernel function is nonlinear. Following the "kernel trick" procedure, the SVR maps the training data into desired higher dimension space, and then the SVR can perform the regression task linearly in higher dimension space and learn the regression decision boundaries. A variety of kernel functions (e.g., polynomial, tanh, Gaussian) can be utilized for the mapping task [49]. We carried out several experiments using these kernels and adopted the most effective one for our training data. The SVR optimization process is represented by Equation 6 [50]:

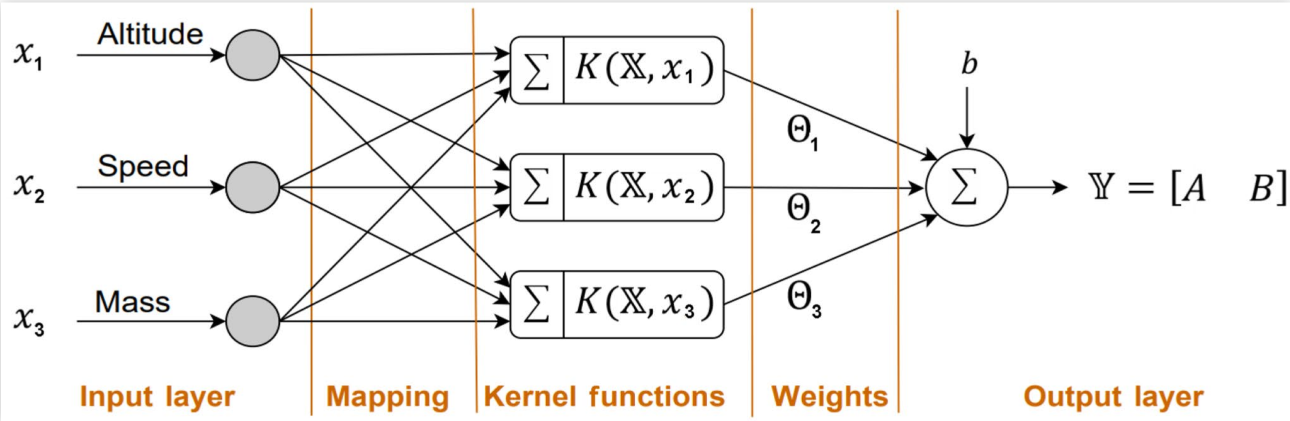
$$\min \frac{1}{2}\|\Theta\|^2 \;\; s.\ t. \begin{cases} \Theta_i x_i + b - y_i \le \epsilon \\ y_i - \Theta_i x_i - b \le \epsilon \end{cases} \qquad \text{Eq. (6)}$$

where $x_i \in \mathbb{X}$ is the given input, $y_i \in \mathbb{Y}$ is the output, and Θ denotes weighting vectors. Moreover, b is the bias term and ϵ adjusts the regression decision boundary, which needs to be tuned carefully. Weighting vectors are obtained during an iterative learning process, and their fine-tuning decreases the chance of overfitting (in order to avoid exactly fitting on a particular dataset and loss of generalization for other input data).

Figure 5 shows the trained neural network architecture for the UAS-S4 FDM prediction. The designed SVR learns relying on training data applied to the input layer (altitude, speed, and mass) and expected in the output containing the state matrix A and control matrix B. In detail, the input neurons fully propagate the input vector into all the kernel functions that are designated for dimensional mapping. Next, the outputs of kernel functions directly update the SVR weight vector. Finally, the output layer computes A and B based on the input vector, weight vector, biased term, and decision boundaries.

Although neural network–based models can accurately perform the FDM regression task, they remain always vulnerable to adversarial attacks as they are considered a potential threat. The following section describes the concept of

FIGURE 5 Architecture of the trained SVR for the UAS-S4 FDM.



adversarial attack, its effects on the FDM regression, and its proper countermeasure for securing the UAS-S4 data-driven FDM.

3. Adversarial Attacks

Basically, an "adversarial attack" is an optimization problem that is solved to produce "crafted" samples similar to the original samples, which can deliberately mislead a neural network prediction model [51]. The concept of adversarial attack can be mathematically determined as the following optimization problem [18], which generates adversarial FDM samples $\tilde{x}$ by the use of training FDM samples $x_i \in \mathbb{X}$.

$$\min\left\|x_i - \tilde{x}\right\| < \varepsilon \quad s.t. \quad \min\left\|f\left(x_i\right) - f\left(\tilde{x}_i\right)\right\| > \gamma \qquad \text{Eq. (7)}$$

where ε is the injected perturbation that back-propagates through the deep network and γ denotes the threshold for surpassing the eligible regression boundary.

Regarding the aim of the adversarial attack optimization problem (based on Equation 7), even though the absolute value of the added perturbation is less than the random noise value, the error rate associated with the perturbation is significant as injected perturbation exactly crafts the samples by exploiting the model vulnerabilities [52]. In other words, the predicted FDM samples relying on the training FDM samples $x_i \in \mathbb{X}$ are predicted wrong in that they are located outside of the regression boundary.

Adversarial attacks can be classified based on their learning phases into two groups, known as "poisoning attacks" and "evasion attacks" [53]. Poisoning (or causative) attacks occur during the training phase. If the dataset for training a target model is accessible, a model may be attacked by the injection of adversarial vulnerabilities. In other words, the targeted model is trained while it is sensitive to certain perturbations and surpasses the threshold in the regression problem. Other attacks may occur during the test phase; these are called evasion (or exploratory) attacks. Instead of manipulating the parameters or architecture of a model, the evasion technique guides a model toward generating selected adversarial outputs. Since evasion-based attacks require less time and less computational effort than poisoning attacks to generate adversarial samples, they have been more common and more successful [54]. Hence, our FDM attack algorithm is designed based on the evasion procedure.

Evasion adversarial attacks can be generated using either white-box or black-box methodologies [55]. White-box attacks use methods that require transparency of the data-driven model and its training data, while black-box attacks are used when the training set and model parameters are not available. In this way, white-box attackers generate adversarial samples by relying on their knowledge of parameters, hyperparameters, and architecture of the prediction model [56]. In this study, given our access to the training data and the SVR model details, a white-box adversarial attack methodology is used. Preparing for this type of attack is more practical for the prediction model development, as the developer can better identify model weaknesses, and can improve its robustness during the training phase. Several different white-box attack techniques have been developed for a variety of applications [57, 58, 59]. The Fast Gradient Sign Method (FGSM) as the baseline successful algorithm has been reformulated for the regression problem that can mislead FDMs.

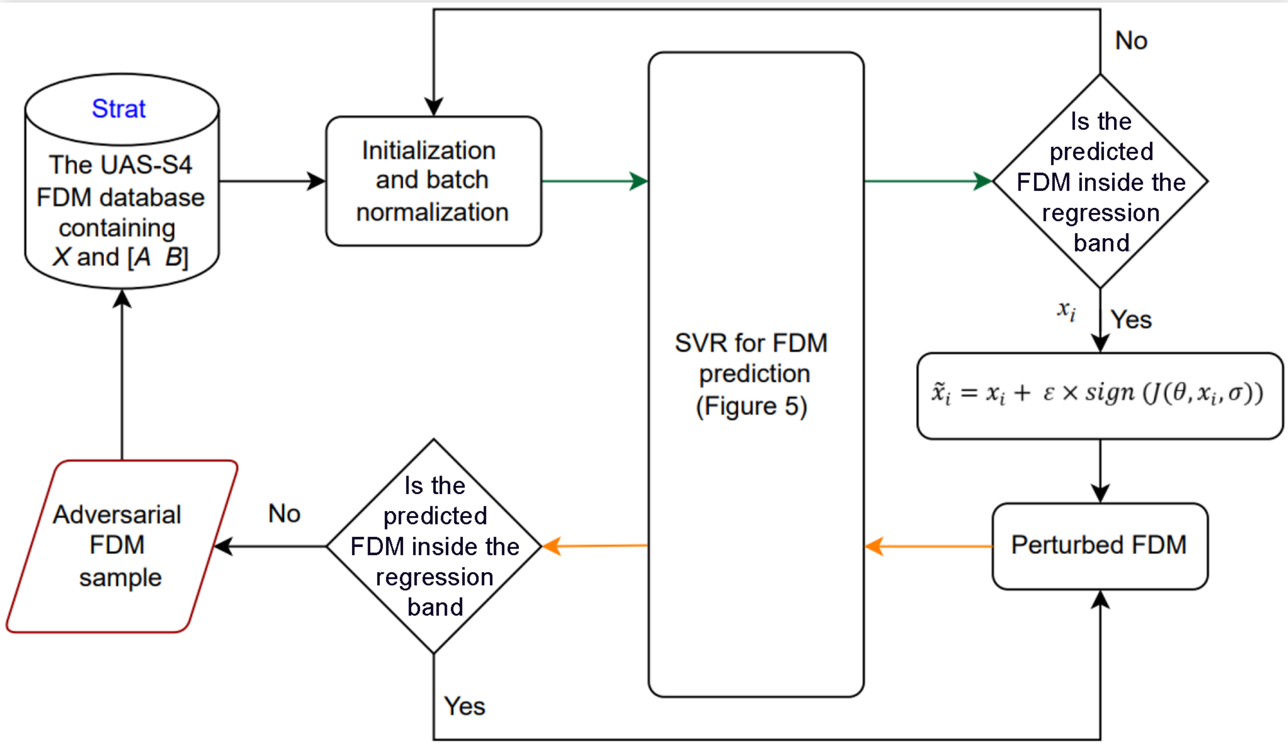
A. Adapted Fast Gradient Sign Method

The FGSM is a nontargeted attack that operates based on gradient information [60]. Formulation of the Adapted Fast Gradient Sign Method (AFSGM) for solving a regression problem can be represented as Equation 8 [18]:

$$\tilde{x}_i = x_i + \varepsilon \times sign\left(J\left(\theta, x_i, \sigma\right)\right) \qquad \text{Eq. (8)}$$

where J is the cost function relying on the weight vector θ, input x_i, and output σ, in which $\sigma \notin [a - \mu, a + \mu]$. In detail, a float scalar perturbation ε is injected into the SVR-FDM

FIGURE 6 The step-wise procedure followed for generating adversarial FDM samples in order to use for the SVR retraining.



output layer and backpropagated through it. Relying on both optimized ε and μ, the adversarial FDM sample $\tilde{x}_i$ will be generated in the input using gradient information as shown in Equation 8. A flowchart shown in Figure 6 is the procedure followed for generating adversarial FDM samples in order to use it for the SVR retraining.

Consider the UAS-S4 FDM database containing 3,642 local linear models. The database is composed of a given input $x_i \in \mathbb{X}$ (altitude, speed, and mass) and the associate state-space matrices $y_i \in \mathbb{Y}$ (A and B). Initialization is done to set the weighting vectors, and batch normalization is done for recentering and rescaling input data. Then the SVR is trained using the prepared UAS-S4 FDM dataset. If the predicted FDM is outside of the regression band, it should be initialized again. Otherwise, it can be used for generating adversarial FDM. The predicted FDM sample is transmitted into the optimization recursive gradient-based algorithm (Equation 7) to create a perturbed FDM sample that is similar to the original FDM. The perturbed FDM is generated through the following algorithm. Finally, the perturbed FDM sample is applied to the SVR in the test phase as the new sample. If the predicted FDM is located outside of the regression band, that perturbed FDM sample can be considered adversarial. The adversarial FDM samples can be added to the database in order to retrain the SVR (with the aim of adversarial attack robustness). Now we need to secure the regression model against adversarial samples through a proper defense strategy.

B. Defense against Adversarial Attacks

Generally, defense algorithms can be categorized into two procedures based on their operation mechanisms [51]. As explained above, adversarial FDM samples can be generated using white-box or black-box methodologies. In fact, the defense strategy must work fine regardless of attack choice approach. Generative Adversarial Network (GAN) strategy can make the FDM model robust even in the absence of knowledge about the corresponding neural network architecture and hyperparameters. However, the GAN requires high computational efforts for providing robustness. In this study, since the structure of the SVR and its hyperparameters are known, the GAN strategy is not affordable. With knowledge of the FDM prediction model, we adopted the typical and most known countermeasures called adversarial retraining defense [61], in which the targeted model is trained using Equation 9 [18].

$$\tilde{J}(x_i, y_i, \theta) = cJ(x_i, y_i, \theta) + (1-c)cJ(\tilde{x}_i, y_i, \theta) \quad \text{Eq. (9)}$$

where $c \in [0, 1]$ is a constant value in the retraining process.

"Adversarial Retraining" methodology functions based on retraining the neural network prediction models using a combination of original and adversarial samples. By use of this technique, the combination of adversarial and original samples is applied to the SVR-FDM within the training phase. Finally, the retrained FDM avoids mistake prediction in the test phase, and in real-time operations. It is worth mentioning that high values for $c \in [0, 1]$ improve the SVR robustness while reducing prediction accuracy and vice versa. The optimization of the c value is not in the scope of this article, and we considered c = 0.5, which is a fair value on average. The optimized c value can be obtained through well-known methodologies, such as Genetic Algorithm and Particle Swarm Optimization [62, 63].

4. Results

The SVR performance is analyzed using different tools. As explained in Section 2, the SVR output gives the UAS-S4 FDM through the prediction of state and control matrices for any flight condition in the flight envelope. Since there are an infinite number of state-space representations for an FDM, the numerical differences between the original and predicted elements of state-space matrices cannot be used for evaluating the SVR prediction accuracy. In fact, the location of closed-loop poles can represent a dynamic model behavior. Therefore, that infinite number of state-space representations for a specific dynamics model can be demonstrated by a unique "Root Locus" diagram, a tool for prediction accuracy assessment. Another criterion that needs to be evaluated is the SVR regression precision. Using this criterion, the performance of the controlled UAS-S4 state variables based on the predicted FDM (provided by the SVR and Robust-SVR) can be compared to those based on the original FDM (provided by the LLS).

A. UAS-S4 FDM Root-Locus Diagram

Assume that the UAS-S4 FDM in the trim flight condition is expressed by the speed = 43 m/s, altitude = 6,000 m, mass = 53 kg; it is important to note that this trimmed local FDM has not been used for data augmentation or as a training sample.

The original longitudinal state-space matrices under the elevator angle actuation are

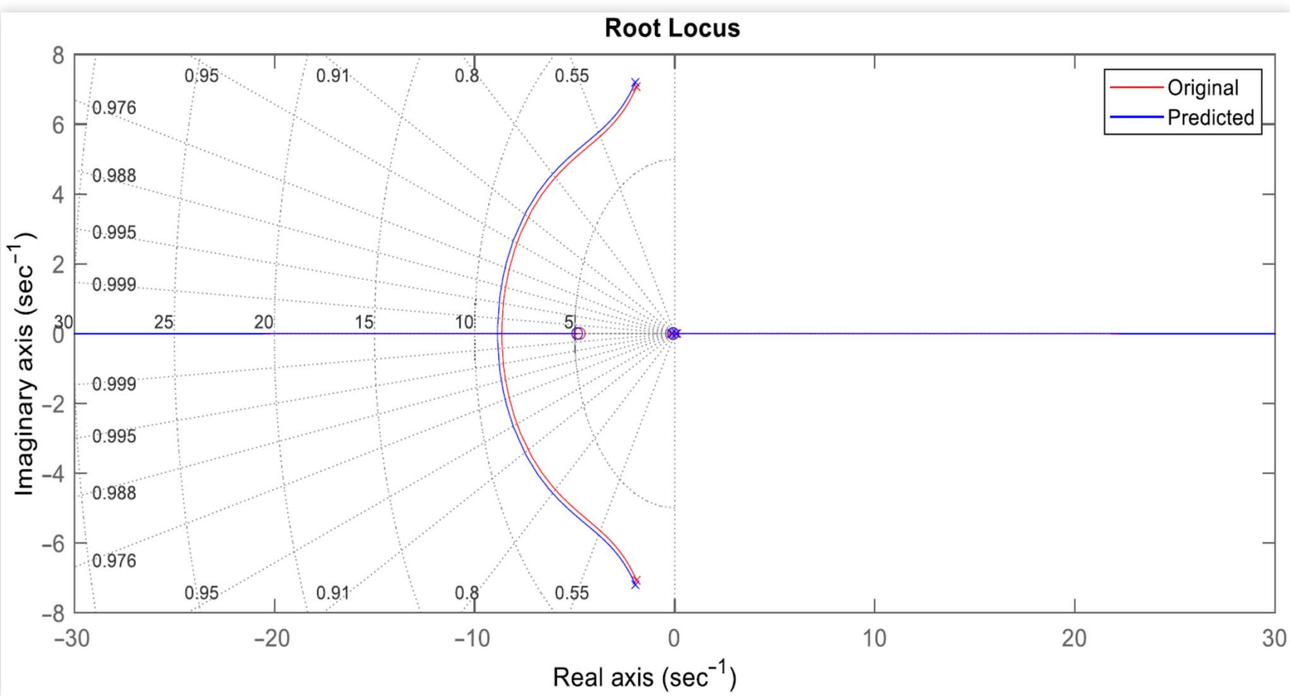
$$A_{lon} = \begin{bmatrix} -0.064 & 0.2434 & -1.087 & -9.784 \\ -0.361 & -4.261 & 43.826 & -0.251 \\ -0.136 & -1.268 & 0.4455 & -0.012 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad B_{lon_E} = \begin{bmatrix} -0.012 \\ 0.0592 \\ -0.145 \\ 0 \end{bmatrix}$$

and the predicted matrices are

$$A_{lon} = \begin{bmatrix} -0.063 & 0.2501 & -1.095 & -9.876 \\ -0.376 & -4.362 & 41.991 & -0.269 \\ -0.137 & -1.271 & 0.4531 & -0.013 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad B_{lon_E} = \begin{bmatrix} -0.012 \\ 0.0580 \\ -0.148 \\ 0 \end{bmatrix}$$

The Root Locus diagram is used to compare the original with the predicted state-space matrices and their stability. As seen in Figure 7, the SVR accurately predicts the UAS-S4 trimmed FDM eigenvalues.

FIGURE 7 Root Locus diagram of the UAS-S4 longitudinal FDM under its elevator angle actuation.



The original lateral state-space matrices under the aileron angles actuation are

$$A_{lat} = \begin{bmatrix} -0.234 & 0.0617 & -55.24 & -9.591 \\ -0.058 & -13.99 & 0.8235 & 0 \\ 0.0943 & -0.176 & -0.642 & 0 \\ 0 & 1 & 0.0011 & 0 \end{bmatrix}, \quad B_{lat_A} = \begin{bmatrix} 0 \\ 0.8058 \\ -0.008 \\ 0 \end{bmatrix}$$

and the predicted matrices are

$$A_{lat} = \begin{bmatrix} -0.241 & 0.0611 & -56.05 & -9.591 \\ -0.060 & -14.23 & 0.8122 & 0 \\ 0.0955 & -0.182 & -0.649 & 0 \\ 0 & 1 & 0.0012 & 0 \end{bmatrix}, \quad B_{lat_A} = \begin{bmatrix} 0 \\ 0.7889 \\ -0.007 \\ 0 \end{bmatrix}$$

The following Root Locus diagram shows the SVR performance in terms of the UAS-S4 lateral FDM prediction accuracy under the aileron angle actuation. As seen in Figure 8, the predicted eigenvalues are very close to the original ones.

For the lateral FDM under rudder angle actuation, the original state-space matrices are

$$A_{lat} = \begin{bmatrix} -0.234 & 0.0617 & -55.24 & -9.591 \\ -0.058 & -13.99 & 0.8235 & 0 \\ 0.0943 & -0.176 & -0.642 & 0 \\ 0 & 1 & 0.0011 & 0 \end{bmatrix}, \quad B_{lat_R} = \begin{bmatrix} -0.044 \\ 0.0091 \\ -0.199 \\ 0 \end{bmatrix}$$

and the predicted matrices are

$$A_{lat} = \begin{bmatrix} -0.241 & 0.0611 & -56.05 & -9.591 \\ -0.060 & -14.23 & 0.8122 & 0 \\ 0.0955 & -0.182 & -0.649 & 0 \\ 0 & 1 & 0.0012 & 0 \end{bmatrix}, \quad B_{lat_R} = \begin{bmatrix} -0.043 \\ 0.0082 \\ -0.196 \\ 0 \end{bmatrix}$$

Figure 9 shows the Root Locus diagram of the UAS-S4 lateral FDM under rudder angle actuation.

The Root Locus diagrams in Figures 7, 8, and 9 graphically show the SVR success in accurate UAS-S4 FDM eigenvalues prediction, in which the predicted eigenvalues were found to be very close to their original values. These diagrams allowed the SVR prediction accuracy to be evaluated for different flight conditions. These validation studies were performed for five flight conditions, as their five corresponding UAS-S4 FDMs have not been used for data augmentation or as training samples. As previously mentioned, the UAS-S4 FDM prediction accuracy was evaluated separately for the elevator, aileron, and rudder actuation. The Mean Absolute Error (MAE in %) was considered as a performance index calculated for the actual and predicted eigenvalues. Table 2 shows the SVR prediction errors for the UAS-S4 FDM for different flight conditions and control actuators.

As shown in Table 2, the SVR could accurately predict the UAS-S4 FDM for different flight conditions. Its performance was found to be slightly better for the first flight condition, where the UAV operated at the lowest altitude, speed, and mass. The low percentages of the MAE confirm the

FIGURE 8 Root Locus diagram of the UAS-S4 lateral FDM under aileron angle actuation.

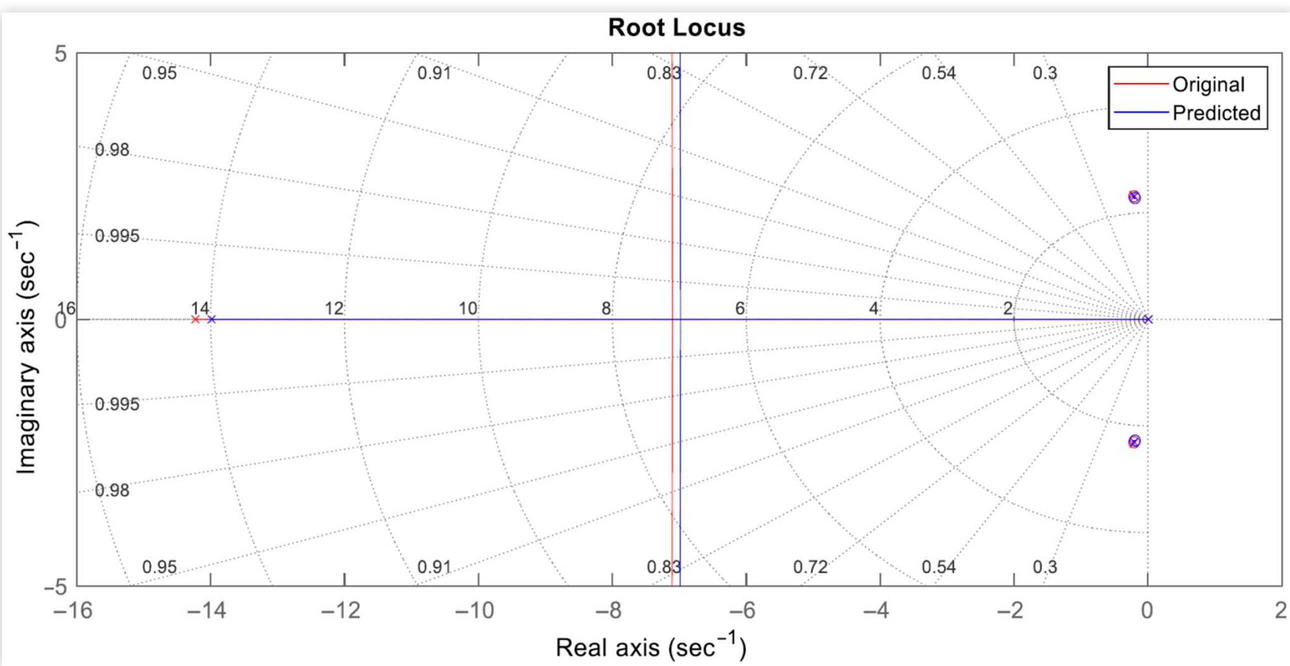
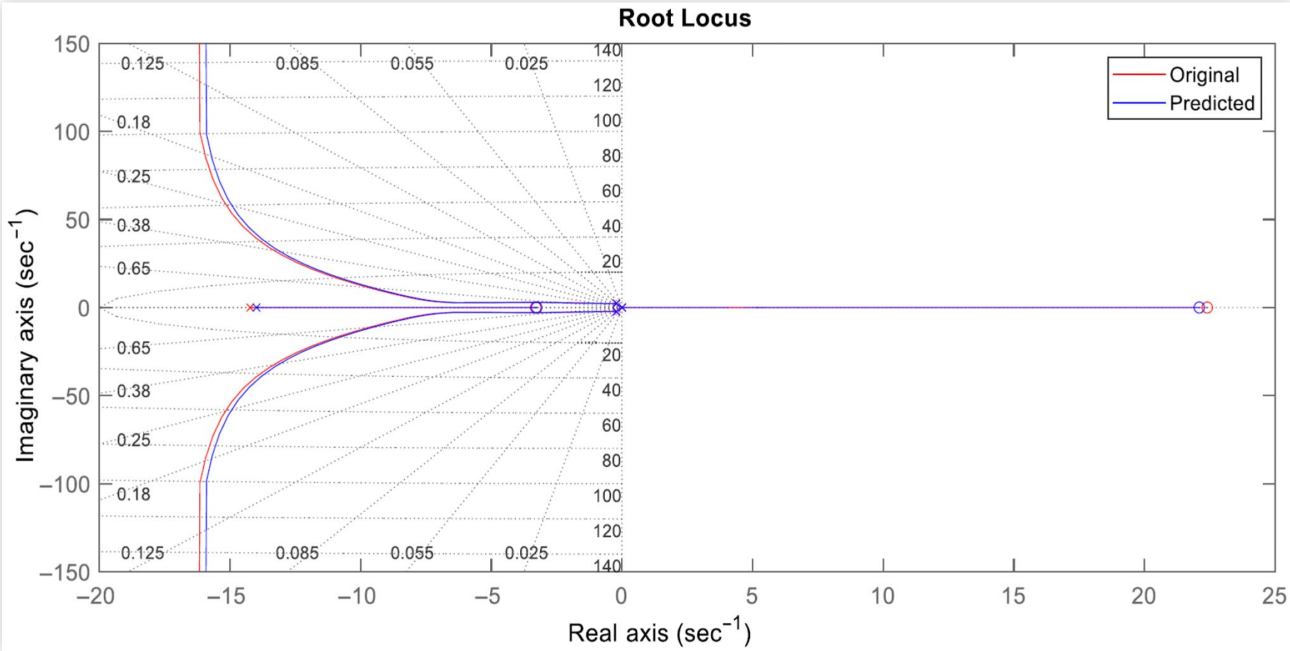


FIGURE 9 Root Locus diagram of the UAS-S4 lateral FDM under rudder angle actuation.



excellent SVR prediction accuracy for a variety of flight conditions, as the MAE never exceeded 2.38%.

Since the data augmentation process used in a neighbor-based algorithm was the fundamental process for the SVR, we conducted another study to evaluate the different numbers of neighbors used for data augmentation. Note that the UAS-S4 FDM was considered in the trim condition, where v = 43 m/s, alt = 6,000 m, and M = 53 kg. Table 3 shows the SVR accuracy obtained for the UAS-S4 FDM prediction when different numbers of neighbors are used in the data augmentation process.

TABLE 2 The UAS-S4 FDM prediction accuracy using the SVR at five flight conditions.

Flight condition		Control surface used for actuation	Mean absolute error %
1	Altitude = 100 m	Elevator	**1.71**
	Speed = 26 m/s	Aileron	**2.29**
	Mass = 53 kg	Rudder	**2.07**
2	Altitude = 2,250 m	Elevator	1.74
	Speed = 32.3 m/s	Aileron	2.33
	Mass = 57.6 kg	Rudder	2.11
3	Altitude = 4,500 m	Elevator	1.78
	Speed = 38.6 m/s	Aileron	2.36
	Mass = 66.8 kg	Rudder	2.14
4	Altitude = 6,000 m	Elevator	1.81
	Speed = 43 m/s	Aileron	2.38
	Mass = 76 kg	Rudder	2.17
5	Altitude = 6,000 m	Elevator	1.75
	Speed = 43 m/s	Aileron	2.33
	Mass = 53 kg	Rudder	2.12

Table 3 represents the UAS-S4 FDM prediction accuracy for four different numbers of neighbors under elevator, aileron, and rudder angle actuation, separately. According to the MAE values considered as the performance index, it was found that a high number of neighbors (5 neighbors compared to 4, 3, and 2 neighbors) can provide better data augmentation, and consequently more accurate FDM prediction. However, a high number of neighbors cause computational complexity in the data augmentation process. Adopting three nearest local FDMs in the neighborhood of the original FDM that is supposed to be augmented is a satisfactory trade-off for the interpolation and extrapolation methods.

TABLE 3 The UAS-S4 FDM prediction accuracy using SVR with different numbers of neighbors used for data augmentation.

Number of neighbors for data augmentation	Control surface used for actuation	Mean absolute error %
2	Elevator	1.86
	Aileron	2.45
	Rudder	2.21
3	Elevator	1.75
	Aileron	2.33
	Rudder	2.12
4	Elevator	1.71
	Aileron	2.30
	Rudder	2.07
5	Elevator	**1.69**
	Aileron	**2.28**
	Rudder	**2.05**

TABLE 4 The UAS-S4 FDM prediction accuracy while considering a range of kernel functions for the SVR.

Kernel functions utilized for the SVR	Control surface used for actuation	Mean absolute error %
Polynomial μ = 0, γ = 1	Elevator	1.89
	Aileron	2.47
	Rudder	2.28
Gaussian μ = 0, σ^2 = 0.5	Elevator	**1.75**
	Aileron	**2.33**
	Rudder	**2.12**
Radial Basis Function (RBF) μ = 0, σ^2 = 0.5	Elevator	1.81
	Aileron	2.39
	Rudder	2.16
Sigmoidal h = 0, C = 1	Elevator	1.83
	Aileron	2.41
	Rudder	2.17

The other factor that affects the SVR performance is the type of kernel function in the hidden layer. Table 4 shows the UAS-S4 FDM prediction accuracy while considering a range of kernel functions in the SVR.

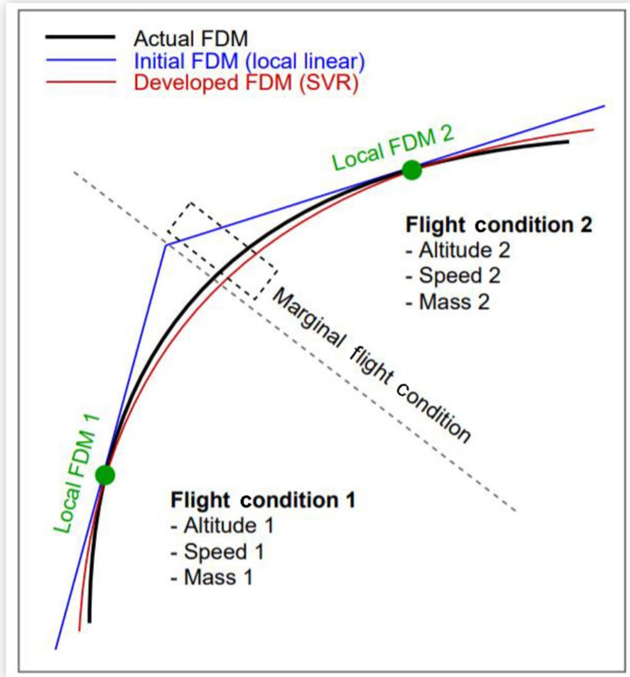
Four types of kernel functions were employed: polynomial, gaussian, radial basis, and sigmoidal. Their fine-tuned parameters are described in Table 4. According to the calculated MAE, the SVR based on the Gaussian function gave the best performance for the UAS-S4 FDM among all the kernel functions utilized, as shown in Table 4. Therefore, the Gaussian-based SVR augmented data relying on three neighbors are considered for further studies on the regression effectiveness in the control loop.

B. Regression Effectiveness under the Operation of a Unique LQR

In order to do a comparison study, we considered the Local Linear Scheduled [64] (LLS) modeling methodology as the baseline methodology. According to the UAS-S4 LLS-FDM, the flight envelope contains 216 local linearized state-space representations, where each one corresponds to a specific case of altitude, speed, and mass. The accuracies of the local linear FDMs degrade when operating points move away from equilibrium points, but the FDM accuracy can be improved using nonlinear regression as shown in Figure 10.

In Figure 10, both the "LLS" and the "SVR" approaches can provide almost the same FDM around the equilibrium points. The effectiveness of the SVR approach is most notable in the marginal flight condition, where it can provide a significantly more accurate FDM compared to the local linear approach. Following this observation, it is clear that a more accurate FDM can improve model-based controller performance. To assess the extent of these improvements, the

FIGURE 10 FDM design based on "Local Linear Scheduling" and "SVR" approaches.



accuracies of the initial (LLS) and developed (SVR) FDMs were compared in the marginal flight condition while they were used in the same Linear Quadratic Regulator (LQR) control loop.

Assume that the UAS-S4 FDM is in the trim condition, where the speed = 43 m/s, altitude = 6,000 m, and mass = 53 kg; its initial longitudinal LLS-FDM is represented by state-space matrices:

$$A_{LLS} = \begin{bmatrix} -0.064 & 0.2434 & -1.087 & -9.784 \\ -0.361 & -4.261 & 43.826 & -0.251 \\ -0.136 & -1.268 & 0.4455 & -0.012 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad B_{LLS} = \begin{bmatrix} -0.012 \\ 0.0592 \\ -0.145 \\ 0 \end{bmatrix}$$

In the initial (LLS approach) flight envelope, this linearized state-space FDM was allocated to the operation range of speed = 41.8 – 43 m/s, altitude = 5,625–6,000 m, and mass = 53–55 kg. We assume that the UAS-S4 is operating within the flight conditions that are at the margin of the mentioned operation range, where the speed = 41.8 m/s, altitude = 5,625 m, and mass = 55 kg. When the UAS-S4 operates in this marginal flight condition, the initial state-space representation in the trim condition cannot accurately determine the flight dynamics behavior. In order to solve this problem, our designed SVR algorithm predicts the UAS-S4 FDM for any flight conditions away from the trim conditions in the flight envelope. For instance, the state-space matrices of the local

UAS-S4 FDM predicted under the elevator actuation for the abovementioned marginal flight condition are

$$A_{SVR} = \begin{bmatrix} -0.0726 & 0.2346 & -0.9547 & -9.7830 \\ -0.3729 & -4.5992 & 43.3325 & -0.2240 \\ -0.1308 & -1.3599 & 0.4664 & -0.0118 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$$B_{SVR} = \begin{bmatrix} -0.0133 \\ 0.0631 \\ -0.1525 \\ 0 \end{bmatrix}$$

The developed FDM (predicted by the SVR) efficiency was evaluated in a control loop using the same controller as the one that was used for the UAS-S4 trimmed FDM [64]. The Linear LQR is employed as the controller in the closed-loop architecture [65, 66]. Figure 11 shows the LQR controller time-domain performance during the above marginal flight conditions.

In this figure the controlled pitch angle step responses for the initial and the developed UAS-S4 FDMs are plotted in the marginal flight conditions. It can be observed that our designed SVR-FDM (developed model) can provide a more accurate FDM as the same LQR controller gives a better time-domain response compared to the LLS-FDM (initial model) response. Table 5 represents the pitch angle step response results using time-domain performance indexes for the "rise time," the "settling time," and the "overshoot."

According to Table 5, the SVR can provide a more accurate FDM in the marginal flight condition as the SVR-based predicted FDM allows the same LQR controller

FIGURE 11 The controlled pitch angle step responses of the developed (SVR) and the initial (LLS) UAS-S4 FDMs.

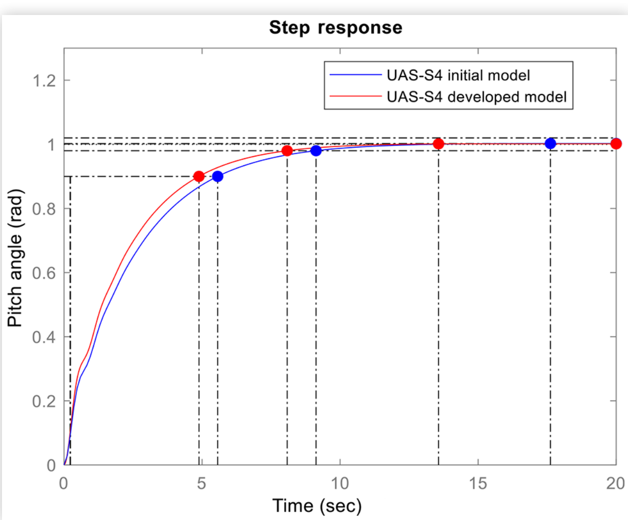


TABLE 5 The controlled step response properties of the developed and the original UAS-S4 FDM.

	Developed UAS-S4 FDM	Initial UAS-S4 FDM
Rise time	4.87 sec	5.63 sec
Settling time	8.08 sec	9.13 sec
Overshoot	0.14%	0.25%

to give a faster pitch angle step response in the marginal flight condition. For the pitch angle step response, the rise time, settling time, and the overshoot decreased by 0.76 sec, 1.05 sec, and 0.105%, respectively, as seen in Table 5.

C. Robustness against Adversarial Attacks

Security issues are always a huge concern for AI-based control systems. By imposing an adversarial attack on the FDM, the controller that works based on that model can be hacked. We generated an adversarial sample using the AFGSM, and we further applied it to the UAS-S4 SVR-FDM. The adversarial sample could mislead the UAS-S4 SVR-FDM prediction. Figure 12 shows the effect of an imposed adversarial attack on the UAS-S4 longitudinal FDM prediction.

According to Figure 12, before imposing such an adversarial attack on the SVR-FDM, the original (blue) and predicted (red) eigenvalues are very close together. However, the hacked eigenvalues (green) in the Root Locus diagram show that an adversarial sample can mislead the data-driven model so that the SVR predicts the UAS-S4 FDM far from its original values. This incorrect prediction directly affects the FDM controller and causes it to generate the wrong commands for the control surfaces.

Assume that the UAS-S4 operates in the marginal flight condition speed = 42 m/s, altitude = 5,700 m, and mass = 55 kg, and the LQR controls its state variables. Figure 13 shows the UAS-S4 pitch angle step response under LQR controller operation when different types of FDMs are used.

The model-based controller shows the best performance in terms of time-domain properties when it relies on the SVR-FDM (green). However, it approached failure when an adversarial attack was imposed on the SVR-FDM (red). As shown in Figure 13, the Robust SVR-FDM (black) offered a better step response compared to the conventional LLS-FDM (Blue) under an imposed adversarial attack. The time-domain properties in terms of "rise" and "settling" times are given in Table 6.

The SVR-FDM provided the fastest responses (4.87 sec for rise time and 8.08 for settling time). The SVR FDM is more accurate than the LLS FDM.

However, the SVR failed when it approached an adversarial sample. The Robust SVR-FDM could provide a trade-off between regression effectiveness and adversarial attack robustness. The Robust SVR-FDM regression accuracy was lower than that of the SVR-FDM in which the corresponding controller showed a 0.42 sec slower step response for the rise

FIGURE 12 The effect of an imposed adversarial attack on the UAS-S4 longitudinal FDM prediction.

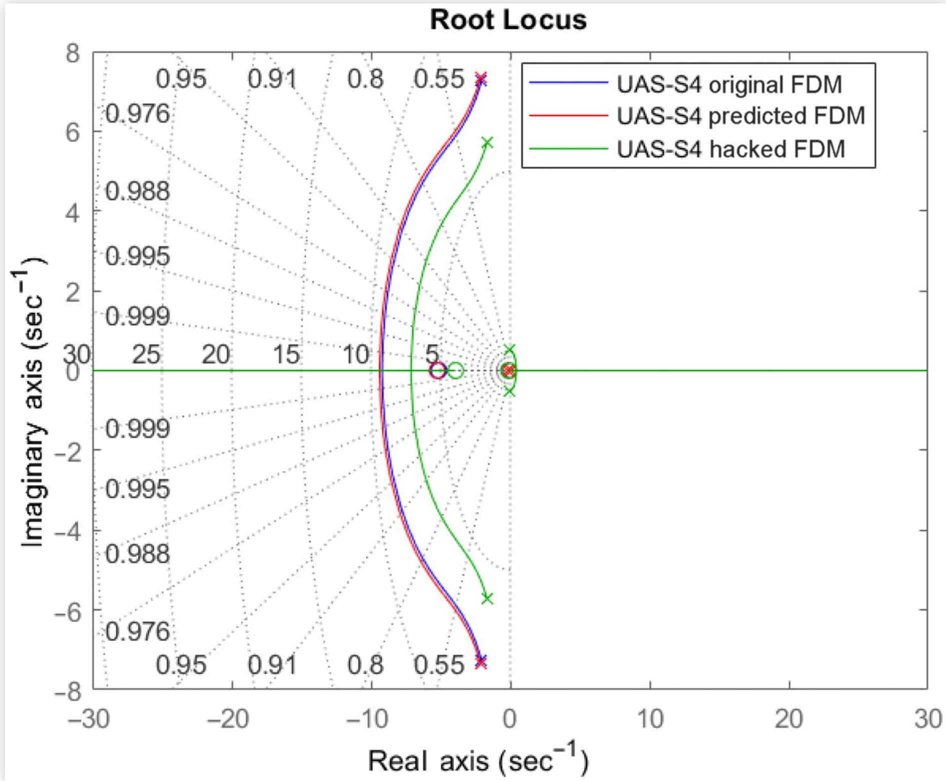


FIGURE 13 The UAS-S4 pitch angle step response using an LQR controller when different types of FDMs are used while an adversarial attack is imposed.

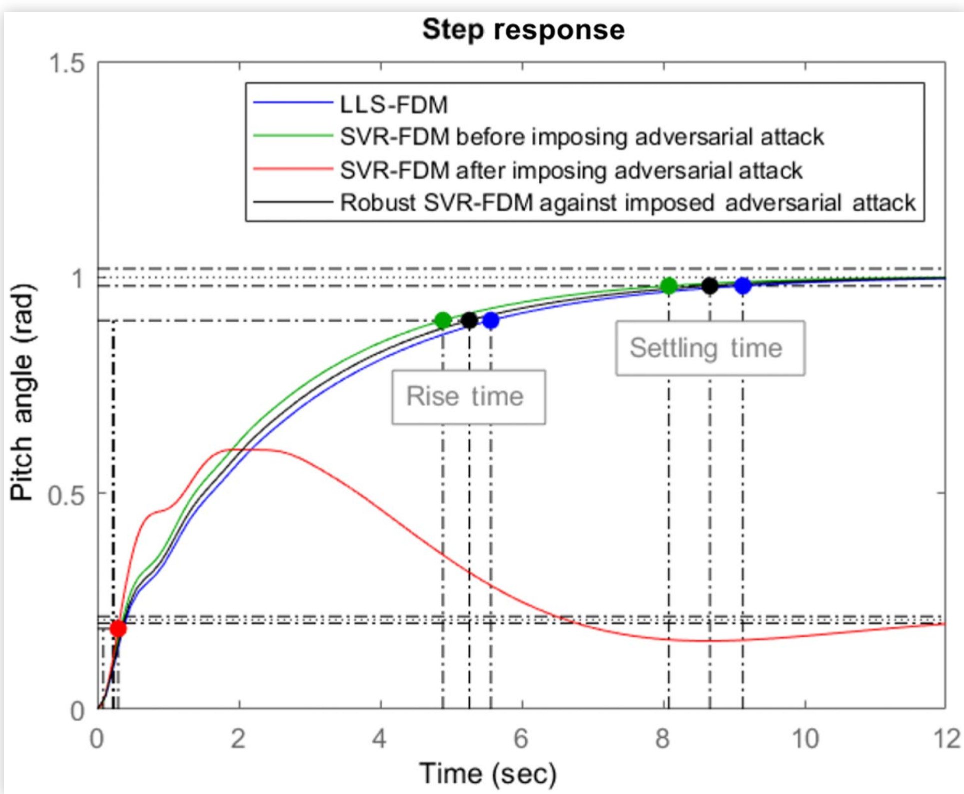


TABLE 6 The UAS-S4 pitch angle step response time-domain characteristics under the LQR controller when different types of FDM are used.

	Rise time (sec)	Settling time (sec)	Adversarial attack examination
LLS-FDM	5.63	9.13	Pass
Robust SVR-FDM	5.29	8.64	Pass
SVR-FDM	4.87	8.08	Fail

TABLE 7 The UAS-S4 roll angle step response time-domain characteristics under the LQR controller when different types of FDM are used.

	Rise time (sec)	Settling time (sec)	Adversarial attack examination
LLS-FDM	4.29	7.21	Pass
Robust SVR-FDM	4.05	6.76	Pass
SVR-FDM	3.74	6.22	Fail

time and 0.56 sec for the settling time. On the contrary, the Robust SVR-FDM was better than the LLS-FDM (0.34 faster step response for the rise time and 0.49 sec for the settling time) while remaining robust against adversarial attacks.

A similar type of analysis was done for the lateral control surfaces. The UAS-S4 roll angle step responses with the LQR controller when different types of FDMs are used are presented in Figure 14.

Using the SVR-FDM, the LQR controller gives the best step response for the roll angle (green). However, it fails when is submitted to an adversarial FDM sample (red). As shown in Figure 14, the Robust SVR-FDM (black) provides a faster response compared to the LLS-FDMs (blue), under an imposed adversarial attack. The time-domain properties are given in terms of rise time and settling time in Table 7.

As indicated in Table 7, the SVR-FDM provided the fastest roll angle step response (3.74 sec for rise time and 6.22 for settling time); however, its failure in case of adversarial attacks convinced us to improve it. The Robust SVR-FDM effectiveness was less than that of the SVR-FDM (0.31 sec slower step response for the rise time and 0.54 sec for the settling time). On the contrary, the Robust SVR-FDM was more accurate than the LLS-FDM, in which its step response was 0.24 and 0.45 sec faster for the rise time and settling time, respectively. Additionally, it remained robust while approaching adversarial FDM.

FIGURE 14 The UAS-S4 roll angle step responses using the LQR controller when different types of FDMs are used while an adversarial attack is imposed.

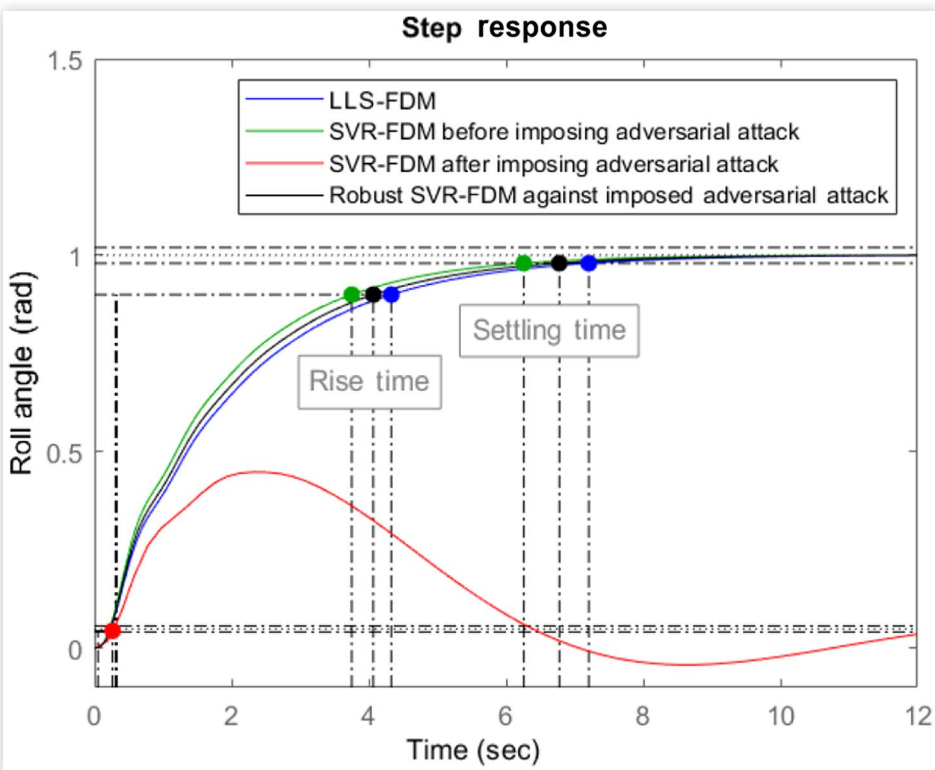
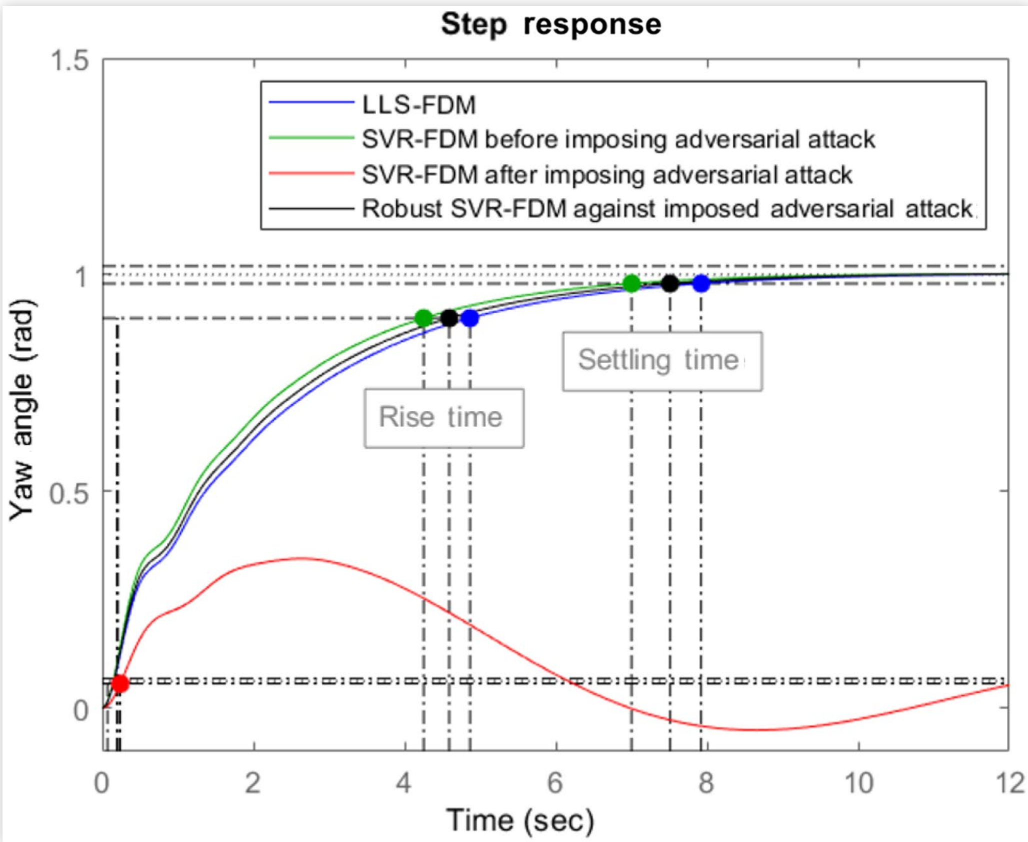


FIGURE 15 The UAS-S4 yaw angle step response with an LQR controller when different types of FDMs are used while an adversarial attack is imposed.



Regarding the lateral motion control studies, the UAS-S4 yaw angle step response is presented in Figure 15 revealing that in using an SVR-FDM, the LQR controller performed the fastest step response for the yaw angle (green). However, the LQR step response collapsed when it approached adversarial FDM samples. On the contrary, the Robust SVR-FDM (black) gave a faster step response compared to the LLS-FDMs (blue) under an imposed adversarial attack.

The time-domain specifications in terms of rise time and settling time are listed in Table 8. Moreover, adversarial attack examinations were conducted in order to evaluate various FDM methodologies results.

As seen in Table 8, it is clear that using the SVR-FDM for a model-based controller resulted in the fastest yaw angle step response (4.24 sec for rise time and 6.98 for settling time). However, its failure to perform under adversarial attacks motivated us to make more improvements. The effectiveness of our Robust SVR-FDM for the yaw angle control was less than that obtained for the SVR-FDM (0.32 sec slower rise time and 0.53 sec slower settling time). On the other hand, the Robust SVR-FDM confirmed its superiority over the LLS-FDM (0.25 sec faster rise time and 0.39 faster settling time), while it proved its robustness against adversarial attacks.

Following the lateral and longitudinal step response analyses described above, the study was performed for all marginal flight conditions, and the average time-domain step response improvements of the UAS-S4 state variables are given in Table 9.

Table 9 shows that the SVR-FDM improved time-domain step response properties much better than the Robust SVR-FDM. However, the Robust SVR-FDM could preserve the controller against adversarial attacks as the SVR-FDM failed due to its weakness against adversarial samples.

According to the tables and figures presented above, our designed Robust-SVR outperformed the conventional FDM approach (LLS) in terms of prediction accuracy, and also

TABLE 8 The UAS-S4 yaw angle step response time-domain characteristics with an LQR controller when different types of FDM are used.

	Rise time (sec)	Settling time (sec)	Adversarial attack examination
LLS-FDM	4.81	7.90	Pass
Robust SVR-FDM	4.56	7.51	Pass
SVR-FDM	4.24	6.98	Fail

TABLE 9 The SVR and Robust-SVR step response improvements compared to the LLS-FDM performance for 120 marginal flight conditions.

Control surfaces angle	Average rise time improvement (%)			Average settling time improvement (%)		
	Pitch	Roll	Yaw	Pitch	Roll	Yaw
Robust SVR-FDM	6.1	5.6	5.2	5.3	6.2	4.9
SVR-FDM	13.5	12.9	11.9	11.5	13.7	11.6

outperformed the modern data-driven FDM approach (SVR) in terms of robustness against adversarial attacks.

5. Conclusions

In this study, 216 LLS-FDMs corresponding to 216 trim flight conditions were used. The flight envelope data were then augmented using interpolation and extrapolation methodologies. These methodologies gave the three closest neighbors of the original operating point supposed in the flight envelope. Next, the centroid of the embedding local FDMs was computed. Then, the new FDM was generated through interpolation and extrapolation methodologies between the centroid and the original operating point. Following this procedure, the number of trimmed local FDMs was increased by up to 3,642. Relying on the augmented dataset, the SVR methodology was used as the benchmarking algorithm. The designed SVR predicted very well the UAS-S4 FDM for the entire flight envelope.

For validation studies, the initial (LLS) and the developed (SVR) UAS-S4 FDM were evaluated based on the Root Locus diagram for elevator, rudder, and aileron angle actuation. The closeness of the predicted eigenvalues to the actual eigenvalues confirmed the high accuracy of the SVR-based UAS-S4 FDM. According to the MAE Performance Index, the excellent SVR prediction accuracy for a variety of flight conditions was confirmed as the MAE never exceeded 2.38%. In addition, a high number of neighbors (5) compared to a lower number of neighbors (4, 3, and 2) could provide better data augmentation, and consequently higher accurate FDM prediction. In the case of considering five FDM neighbors, the MAEs reached their lowest values for the UAS-S4 elevator (1.69%), aileron (2.28%), and rudder (2.05%). However, a high number of neighbors cause computational complexity for data augmentation. The SVR based on a Gaussian kernel function (with 3 neighbors) showed the best performance for the UAS-S4 FDM compared to the conditions when the kernel functions such as the RBF, Sigmoidal, and Polynomial were used, where the highest MAEs were 2.33% for the aileron and 1.75 for the elevator.

The regression performance was then analyzed based on the pitch, roll, and yaw angle step response in closed-loop LQR control architecture. Compared to the initial UAS-S4 LLS-FDM, our newly developed UAS-S4 SVR-FDM provided more accurate local FDMs in marginal flight conditions. By using the newly developed UAS-S4 SVR-FDM, the controller could improve the time-domain response properties for the pitch angle step response: 0.76 sec faster rise time, 1.05 sec faster settling time, and 0.105% less overshoot confirmed the better performance of the SVR-FDM over the LLS-FDM.

It was shown that the SVR architecture was vulnerable against adversarial attacks. The concept of adversarial attack was clarified as the main threat for our designed SVR-FDM. AFGSM was utilized in order to generate the adversarial FDM sample in order to show the possibility of fooling the FDM. Wrong control commands that were generated by the SVR-FDM under adversarial attacks caused misled step responses. The adversarial retraining methodology was designed in order to provide our UAS-S4 SVR-FDM robustness. The robust version of our UAS-S4 SVR-FDM could provide a trade-off between regression accuracy and attack resiliency. In detail, average rise time and settling time improvement for 120 marginal flight conditions were analyzed. Results showed that the designed Robust SVR-FDM was more accurate than LLS-FDM, could detect adversarial FDM samples, and saved the LQR controller against adversarial attacks.

Acknowledgment

Special thanks are due to the Natural Sciences and Engineering Research Council of Canada (NSERC) and to Mrs Carlos Ruiz, Eduardo Yakin, and Alvaro Gutierrez Prado from Hydra Technologies in Mexico, as well as to Ministry of Economy and Innovation (MEI) of the Government of Canada.

Contact Information

Ruxandra Mihaela Botez, PhD, Eng., Full Professor and Canada Research Chair Tier 1 in Aircraft Modeling and Simulation Technologies, Head of the Laboratory of Applied Research in Active Controls, Avionics and AeroServoElasticity LARCASE, ÉTS-University of Quebec, Montreal, Canada, H3C-1K3, ruxandra.botez@etsmtl.ca

Nomenclature

Symbols

A_{lon}, A_{lat} - Longitudinal and lateral state matrices

B_{lon}, A_{lat} - Longitudinal and lateral control matrices

C_{lon}, C_{lat} - Longitudinal and lateral output matrices

D_{lon}, D_{lat} - Longitudinal and lateral feedforward matrices

$G_u, G_w, H_u, H_w, M_u, M_w, M_q$ - Longitudinal state matrix dimensional stability derivatives

G_δ, H_δ, M_δ - Longitudinal control matrix dimensional stability derivatives

p - Roll rate

q - Pitch rate

r - Yaw rate

u - Axial velocity

v - Sideway velocity

w - Vertical velocity

x - Original sample

$\tilde{x}$ - Adversarial sample

X - State variables vector

$\mathbb{X}$ - Support vector regression input

$\mathbb{Y}$ - Support vector regression output

$Y_v, Y_p, Y_r, L_v, L_p, L_r, N_v, N_p, N_r$ - Lateral state matrix dimensional stability derivatives

Y_δ, L_δ, N_δ - Lateral control matrix dimensional stability derivatives

Z_j - Original local flight dynamics model

$\bar{Z}_j$ - New embedding local flight dynamics model

Z_k - Computed centroid flight dynamics model

Greek Letters

γ - Threshold for surpassing the regression boundary

δ - Control vector

δ_a - Aileron angle

δ_e - Elevator angle

δ_r - Rudder angle

ε - Injected perturbation to the SVR output

Θ - Support Vector Regression weighting vector

θ - Pitch angle

λ_{ext} - Extrapolation tuning factor

λ_{int} - Interpolation tuning factor

μ - Decision boundary toward generating adversarial sample

σ - Utilized output value for generating adversarial sample

φ - Roll angle

ϵ - Support Vector Regression decision boundary

Abbreviation

AFGSM - Adapted Fast Gradient Sign Method

AI - Artificial Intelligence

FDM - Flight Dynamics Model

LLS - Local Linear Scheduling

MAE - Mean Absolute Error

SVM - Support Vector Machine

SVR - Support Vector Regression

UAS - Unmanned Aerial System

References

1. Ebeid, E., Skriver, M., Terkildsen, K.H., Jensen, K. et al., "A Survey of Open-Source UAV Flight Controllers and Flight Simulators," *Microprocessors and Microsystems* 61 (2018): 11-20.
2. Saggiani, G., Persiani, F., Ceruti, A., Tortora, P. et al., "A UAV System for Observing Volcanoes and Natural Hazards," in *AGU Fall Meeting Abstracts*, San Francisco, CA, 2007, GC11B-05.
3. Alzahrani, B., Oubbati, O.S., Barnawi, A., Atiquzzaman, M. et al., "UAV Assistance Paradigm: State-of-the-Art in Applications and Challenges," *Journal of Network and Computer Applications* 166 (2020): 102706.
4. Stengel, R.F., *Flight Dynamics* (Princeton, NJ: Princeton University Press, 2015)
5. Vega, L.F.L., Lopez-Neri, E., Arellano-Muro, C.A., González-Jiménez, L.E. et al., "UAV Flight Instructional Design for Industry 4.0 Based on the Framework of Educational Mechatronics," in *IECON 2020 the 46th Annual Conference of the IEEE Industrial Electronics Society*, Singapore, IEEE, 2020, 2313-2318.
6. Botez, R., "Morphing Wing, UAV and Aircraft Multidisciplinary Studies at the Laboratory of Applied Research in Active Controls, Avionics and AeroServoElasticity LARCASE," Aerospace Lab, 2018, Issue 14, 1-11.
7. Hashemi, S. and Botez, R., "Lyapunov-Based Robust Adaptive Configuration of the UAS-S4 Flight Dynamics Fuzzy Controller," *The Aeronautical Journal* 126 (2022): 1187-1209.
8. Wang, Q. and Stengel, R.F., "Robust Nonlinear Flight Control of A High-Performance Aircraft," *IEEE Transactions on Control Systems Technology* 13, no. 1 (2004): 15-26.
9. Ghazi, G., Botez, R., and Messi Achigui, J., "Cessna Citation X Engine Model Identification from Flight Tests," *SAE Int. J. Aerosp.* 8, no. 2 (2015): 203-213, doi:https://doi.org/10.4271/2015-01-2390.
10. Zhou, X., Yu, X., Guo, K., Zhou, S. et al., "Safety Flight Control Design of a Quadrotor UAV with Capability Analysis," *IEEE Transactions on Cybernetics* 53, no. 3 (2021): 1738-1751.
11. Chabir, A., Boukhnifer, M., Bouteraa, Y., Chaibet, A. et al., "Modelling and Fixed Order Robust H∞ Control of Aerial Vehicle: Simulation and Experimental Results," *Compel: International Journal for Computation and Mathematics in Electrical and Electronic Engineering* 35, no. 3 (2016): 1064-1085.
12. Kuitche, M.A.J. and Botez, R.M., "Modeling Novel Methodologies for Unmanned Aerial Systems—Applications

to the UAS-S4 Ehecatl and the UAS-S45 Bálaam," *Chinese Journal of Aeronautics* 32, no. 1 (2019): 58-77.

13. Kuitche, M., Segui, M., Botez, R.M., and Ghazi, G., "New Methodology for Longitudinal Flight Dynamics Modelling of the UAS-S4 Ehecatl towards Its Aerodynamics Estimation Modelling," in *AIAA Modeling and Simulation Technologies Conference*, Grapevine, TX, 2017, 0807.
14. Kuitche, M.A.J., Botez, R.M., Guillemin, A., and Communier, D., "Aerodynamic Modelling of Unmanned Aerial System through Nonlinear Vortex Lattice Method, Computational Fluid Dynamics and Experimental Validation—Application to the UAS-S45 Bàlaam: Part 1," *INCAS Bulletin* 12, no. 1 (2020): 91-103.
15. Kuitche, M.A.J., Botez, R.M., Guillemin, A., and Communier, D., "Aerodynamic Modelling of Unmanned Aerial System through Nonlinear Vortex Lattice Method, Computational Fluid Dynamics and Experimental Validation—Application to the UAS-S45 Bàlaam: Part 2," *INCAS Bulletin* 12, no. 2 (2020): 99-115.
16. Botez, R.M., Hamel, C., Ghazi, G., Boughari, Y. et al., "Level D Research Aircraft Flight Simulator Use for Novel Methodologies in Aircraft Modeling and Simulation," in *Third International Workshop on Numerical Modelling in Aerospace Sciences NMAS*, Bucharest, Romania, 2015.
17. Mosbah, A.B., Botez, R.M., and Dao, T.M., "A Hybrid Original Approach for Prediction of the Aerodynamic Coefficients of an ATR-42 Scaled Wing Model," *Chinese Journal of Aeronautics* 29, no. 1 (2016): 41-52.
18. Hashemi, S.M., Botez, R.M., and Grigorie, T.L., "New Reliability Studies of Data-Driven Aircraft Trajectory Prediction," *Aerospace* 7, no. 10 (2020): 145.
19. Andrianantara, R.P., Ghazi, G., and Botez, R.M., "Aircraft Engine Performance Model Identification Using Artificial Neural Networks," in *AIAA Propulsion and Energy 2021 Forum*, 2021, 3247.
20. Segui, M. and Botez, R.M., "Electric Motor Modeling using Artificial Neural Networks: Application for Drones," in *AIAA AVIATION 2021 FORUM*, 2021, 3003.
21. van Iersel, Q.G., Murrieta Mendoza, A., Felix Patron, R.S., Hashemi, S.M. et al., "Attack and Defense on Aircraft Trajectory Prediction Algorithms," in *AIAA AVIATION 2022 Forum*, Chicago, IL, 2022, 4027.
22. Yu, Y., Yao, H., and Liu, Y., "Physics-Based Learning for Aircraft Dynamics Simulation," in *PHM Society Conference*, Philadelphia, PA, 2018.
23. Yu, Y., Yao, H., and Liu, Y., "Aircraft Dynamics Simulation Using a Novel Physics-Based Learning Method," *Aerospace Science and Technology* 87 (2019): 254-264.
24. Chen, S., Cao, Y., Kang, Y., Zhu, R. et al., "Deep CNN Identifier for Dynamic Modelling of Unmanned Helicopter," in Liu, D., Xie, S., Li, Y., Zhao, D. et al. (eds), *International Conference on Neural Information Processing* (Cham: Springer, 2017), 51-60.
25. Van Dyk, D.A. and Meng, X.-L., "The Art of Data Augmentation," *Journal of Computational and Graphical Statistics* 10, no. 1 (2001): 1-50.
26. Santi, G.M., Ceruti, A., Liverani, A., and Osti, F., "Augmented Reality in Industry 4.0 and Future Innovation Programs," *Technologies* 9, no. 2 (2021): 33.
27. Giridhara, P.K.B., Mishra, C., Venkataramana, R.K.M., Bukhari, S.S. et al., "A Study of Various Text Augmentation Techniques for Relation Classification in Free Text," in *ICPRAM 2019*, Prague, Czech Republic, 2019, 360-367.
28. Hashemi, S.M. and Botez, R.M., "Support Vector Regression Application for the Flight Dynamics New Modelling of the UAS-S4," in *AIAA SCITECH 2022 Forum*, San Diego, CA, 2022, 2576.
29. Awad, M. and Khanna, R., "Support Vector Regression," in *Efficient Learning Machines* (Springer, 2015), Apress, Berkeley, CA, 67-80.
30. Liu, Q., Li, P., Zhao, W., Cai, W. et al., "A Survey on Security Threats and Defensive Techniques of Machine Learning: A Data Driven View," *IEEE Access* 6 (2018): 12103-12117.
31. Huang, S., Papernot, N., Goodfellow, I., Duan, Y. et al., "Adversarial Attacks on Neural Network Policies," arXiv preprint arXiv:1702.02284, 2017.
32. Tuzcu, I., Marzocca, P., Cestino, E., Romeo, G. et al., "Stability and Control of a High-Altitude, Long-Endurance UAV," *Journal of Guidance, Control, and Dynamics* 30, no. 3 (2007): 713-721.
33. Zienkiewicz, O.C., Taylor, R.L., and Zhu, J.Z., *The Finite Element Method: Its Basis and Fundamentals* (Amsterdam, the Netherlands: Elsevier, 2005)
34. Belhocine, A. and Ghazaly, N.M., "Effects of Material Properties on Generation of Brake Squeal Noise Using Finite Element Method," *Latin American Journal of Solids and Structures* 12 (2015): 1432-1447.
35. Belhocine, A. and Abdullah, O.I., "Thermomechanical Model for the Analysis of Disc Brake Using the Finite Element Method in Frictional Contact," *Multiscale Science and Engineering* 2, no. 1 (2020): 27-41.
36. Caughey, D.A., "Introduction to Aircraft Stability and Control Course Notes for M&AE 5070," Sibley School of Mechanical & Aerospace Engineering Cornell University, Ithaca, NY, 2011.
37. Nelson, R.C., *Flight Stability and Automatic Control*. Vol. 2 (New York: WCB/McGraw Hill, 1998)
38. Belhocine, A., Stojanovic, N., and Abdullah, O.I., "Numerical Simulation of Laminar Boundary Layer Flow over a Horizontal Flat Plate in External Incompressible Viscous Fluid," *European Journal of Computational Mechanics* 30, no. 4-6 (2021): 337-386.
39. Belhocine, A., "Numerical Study of Heat Transfer in Fully Developed Laminar Flow inside a Circular Tube," *The International Journal of Advanced Manufacturing Technology* 85, no. 9 (2016): 2681-2692.
40. Belhocine, A. and Abdullah, O.I., "Numerical Simulation of Thermally Developing Turbulent Flow through a Cylindrical Tube," *The International Journal of Advanced Manufacturing Technology* 102, no. 5 (2019): 2001-2012.
41. Afzal, A., Soudagar, M.E.M., Belhocine, A., Kareemullah, M. et al., "Thermal Performance of Compression Ignition

Engine Using High Content Biodiesels: A Comparative Study with Diesel Fuel," *Sustainability* 13, no. 14 (2021): 7688.

42. Sadeghzadeh, I., Mehta, A., and Zhang, Y., "Fault/Damage Tolerant Control of a Quadrotor Helicopter UAV Using Model Reference Adaptive Control and Gain-Scheduled PID," in *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, 2011, 6716.
43. Lawrence, D.A. and Rugh, W.J., "Gain Scheduling Dynamic Linear Controllers for a Nonlinear Plant," *Automatica* 31, no. 3 (1995): 381-390.
44. Dubost, F., Bortsova, G., Adams, H., Ikram, M.A. et al., "Hydranet: Data Augmentation for Regression Neural Networks," in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Cham: Springer, 2019), 438-446.
45. Dijkstra, T. and Veldkamp, J.H., "Data-Driven Selection of Regressors and the Bootstrap," in Dijkstra, T.K. (ed), *On Model Uncertainty and Its Statistical Implications* (Berlin: Springer, 1988), 17-38.
46. Cherkassky, V. and Ma, Y., "Practical Selection of SVM Parameters and Noise Estimation for SVM Regression," *Neural Networks* 17, no. 1 (2004): 113-126.
47. Sebald, D.J. and Bucklew, J.A., "Support Vector Machine Techniques for Nonlinear Equalization," *IEEE Transactions on Signal Processing* 48, no. 11 (2000): 3217-3226.
48. Gunn, S.R., "Support Vector Machines for Classification and Regression," *ISIS Technical Report* 14, no. 1 (1998): 5-16.
49. Lorenzi, L., Mercier, G., and Melgani, F., "Support Vector Regression with Kernel Combination for Missing Data Reconstruction," *IEEE Geoscience and Remote Sensing Letters* 10, no. 2 (2012): 367-371.
50. Chen, D.-R., Wu, Q., Ying, Y., and Zhou, D.-X., "Support Vector Machine Soft Margin Classifiers: Error Analysis," *The Journal of Machine Learning Research* 5 (2004): 1143-1175.
51. Huang, X., Kroening, D., Ruan, W., Sharp, J. et al., "A Survey of Safety and Trustworthiness of Deep Neural Networks: Verification, Testing, Adversarial Attack and Defence, and Interpretability," *Computer Science Review* 37 (2020): 100270.
52. Fan, Y., Wu, B., Li, T., Zhang, Y. et al., "Sparse Adversarial Attack via Perturbation Factorization," in *Computer Vision–ECCV 2020: 16th European Conference*, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16, Springer, Cham, 2020, 35-50.
53. Jiang, W., Li, H., Liu, S., Luo, X. et al., "Poisoning and Evasion Attacks against Deep Learning Algorithms in Autonomous Vehicles," *IEEE Transactions on Vehicular Technology* 69, no. 4 (2020): 4439-4449.
54. Biggio, B., Corona, I., Maiorca, D., Nelson, B. et al., "Evasion Attacks against Machine Learning at Test Time," in Blockeel, H., Kersting, K., Nijssen, S., and Železný, F. (eds), *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (Berlin: Springer, 2013), 387-402.
55. Gil, Y., Chai, Y., Gorodissky, O., and Berant, J., "White-to-Black: Efficient Distillation of Black-Box Adversarial Attacks," arXiv preprint arXiv:1904.02405, 2019.
56. Meng, L., Lin, C.-T., Jung, T.-P., and Wu, D., "White-Box Target Attack for EEG-Based BCI Regression Problems," in Gedeon, T., Wong, K., and Lee, M. (eds), *International Conference on Neural Information Processing* (Cham: Springer, 2019), 476-488.
57. Sun, L., Tan, M., and Zhou, Z., "A Survey of Practical Adversarial Example Attacks," *Cybersecurity* 1, no. 1 (2018): 1-9.
58. Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A. et al., "Adversarial Attacks and Defences: A Survey," arXiv preprint arXiv:1810.00069, 2018.
59. Wang, Q., Liu, H., Xie, G., and Zhang, Y., "Image Denoising Using an Improved Generative Adversarial Network with Wasserstein Distance," in *2021 40th Chinese Control Conference (CCC)*, Shanghai, China, IEEE, 2021, 7027-7032.
60. Goodfellow, I.J., Shlens, J., and Szegedy, C., "Explaining and Harnessing Adversarial Examples," arXiv preprint arXiv:1412.6572, 2014.
61. Mani, N., Moh, M., AND Moh, T.-S., "Towards Robust Ensemble Defense against Adversarial Examples Attack," in *2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, IEEE, 2019, 1-6.
62. Belhocine, A., Shinde, D., and Patil, R., "Thermo-Mechanical Coupled Analysis Based Design of Ventilated Brake Disc Using Genetic Algorithm and Particle Swarm Optimization," *JMST Advances* 3, no. 3 (2021): 41-54.
63. Hashemi, S.M., Barati, S., Talati, S., and Noori, H., "A Genetic Algorithm Approach to Optimal Placement of Switching and Protective Equipment on a Distribution Network," *Journal of Engineering and Applied Sciences* 11 (2016): 1395-1400.
64. Kuitche, M., Yañez-Badillo, H., Botez, R., and Hashemi, S., "Stabilisation, Tracking and Disturbance Rejection Control Design for the UAS-S45 Bálaam," *The Aeronautical Journal* 126, no. 1303 (2022): 1474-1496.
65. Boughari, Y., Ghazi, G., Botez, R.M., and Theel, F., "New Methodology for Optimal Flight Control Using Differential Evolution Algorithms Applied on the Cessna Citation X Business Aircraft—Part 1. Design and Optimization," *INCAS Bulletin* 9, no. 2 (2017): 31.
66. Boughari, Y., Ghazi, G., Botez, R.M., and Theel, F., "New Methodology for Optimal Flight Control using Differential Evolution Algorithms applied on the Cessna Citation X Business Aircraft—Part 2. Validation on Aircraft Research Flight Level D Simulator," *INCAS Bulletin* 9, no. 2 (2017): 45.