# False Data Injection Attacks in Internet of Things and Deep Learning enabled Predictive Analytics

Gautam Raj Mode, Prasad Calyam, Khaza Anuarul Hoque
*Department of Electrical Engineering & Computer Science*
*University of Missouri, Columbia, MO, USA*
gmwyc@mail.missouri.edu, calyamp@missouri.edu, hoquek@missouri.edu

*Abstract*—**Industry 4.0 is the latest industrial revolution primarily merging automation with advanced manufacturing to reduce direct human effort and resources. Predictive maintenance (PdM) is an industry 4.0 solution, which facilitates predicting faults in a component or a system powered by state-of-the-art machine learning (ML) algorithms (especially deep learning algorithms) and the Internet-of-Things (IoT) sensors. However, IoT sensors and deep learning (DL) algorithms, both are known for their vulnerabilities to cyber-attacks. In the context of PdM systems, such attacks can have catastrophic consequences as they are hard to detect due to the nature of the attack. To date, the majority of the published literature focuses on the accuracy of the IoT and DL enabled PdM systems and often ignores the effect of such attacks. In this paper, we demonstrate the effect of IoT sensor attacks (in the form of false data injection attack) on a PdM system. At first, we use three state-of-the-art DL algorithms, specifically, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Convolutional Neural Network (CNN) for predicting the Remaining Useful Life (RUL) of a turbofan engine using NASA's C-MAPSS dataset. Our obtained results show that the GRU-based PdM model outperforms some of the recent literature on RUL prediction using the C-MAPSS dataset. Afterward, we model and apply two different types of false data injection attacks (FDIA), specifically, continuous and interim FDIAs on turbofan engine sensor data and evaluate their impact on CNN, LSTM, and GRU-based PdM systems. Our results demonstrate that attacks on even a small number of IoT sensors can strongly defect the RUL prediction in all cases. However, the GRU-based PdM model performs better in terms of accuracy and FDIA resiliency. Afterward, we perform a study on the GRU-based PdM model using four different GRU networks with different sequence lengths. Our experiments reveal an interesting relationship between the accuracy, resiliency and sequence length for the GRU-based PdM models.**

*Index Terms*—**deep learning, false data injection attack, LSTM, GRU, CNN, industry 4.0, Internet of things, machine learning**

## I. Introduction

Current advances in machine learning (ML) techniques and Internet-of-Things (IoT) sensors has enabled the emergence of predictive maintenance (PdM), which is a method of preventing asset failure by analyzing production data and identifying patterns to predict issues before they happen. State-of-the-art PdM techniques can help reduce downtime by 35%-45%, maintenance cost by 20%-25%, and can increase production by 20%-25% [1]. Due to these benefits, IoT and ML-enabled PdM solutions are reshaping automotive, aerospace, oil and gas, transportation, manufacturing industries and also reshaping the national defense. Specifically, deep learning (DL) algorithms have recently shown tremendous success in such PdM applications [2]. Unfortunately, IoT sensors and DL algorithms are both susceptible to attacks [3], which possess a significant threat to the overall PdM system. According to a recent report from the *Malwarebytes*, cyber-threats against businesses/factories have increased by more than 200% over the past year [4].

Specifically, it is very hard to detect stealthy attacks, such as False Data Injection Attack (FDIA) [5] on the PdM system due to the nature of the attack. In false data injection attack (FDIA) [5], an attacker stealthily compromises measurements from IoT sensors, such that the manipulated sensor measurements bypasses the sensor's basic 'faulty data' detection mechanism and propagates to the sensor output undetected. An FDI attack can be implemented by compromising physical sensors, sensor communication network, and data processing programs. Such attacks on a PdM system can act as a "time bomb" since FDIAs on a PdM system do not show their effect immediately, which also helps in bypassing basic anomaly detection mechanisms. Instead, the attack propagates from the sensor to the ML part of the PdM system and fools the system by predicting a delayed asset failure or maintenance interval. This might incur a significant cost by inducing an unplanned failure or loss of human lives in safety-critical applications [6]–[8]. FDI attacks have already caused many known disasterous incidents, such as the Northeast blackout of 2003 in the USA and the Ukrainian power grid attack affecting over 230,000 people, leaving them without electricity for several hours. Extensive research has been performed on the detection and mitigation of FDI attacks in cyber-physical systems (CPS) domain [9]–[11]. Unfortunately, the effect of FDIA on a PdM system is yet not explored which motivates our research. In the case of aircraft engine PdM systems, FDIAs may result in the delay of timely maintenance and lead to mid-air engine failures which are catastrophic. Current users of PdM systems for aircraft engine maintenance includes Pratt and Whitney, Rolls-Royce, Honeywell, General electronics and the US Air force [6], [12]–[14]. For example, Bombadiers new jetliner uses a Pratt and Whitney turbofan engine that boasted more than 5,000 sensors [15], [16]. Powered with the modern DL algorithms, this engine can predict the future demands

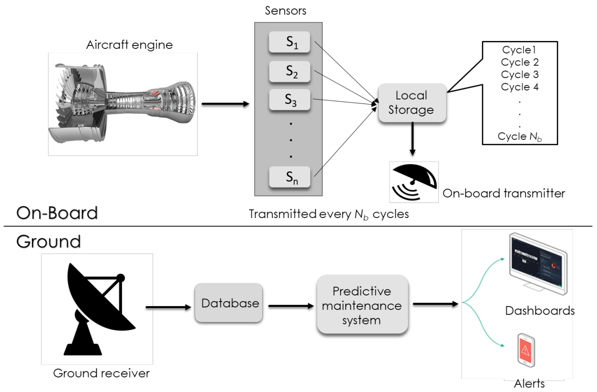


Fig. 1: Engine health monitoring (EHM) system architecture

of the engine, perform adjustments, and thus save 15% of fuel usage. However, the vulnerability of sensor-attacks against for such IoT and ML-based engines is considered a challenge [15], [17], [18]. The existing sensor attack detection solutions in the IoT and cyber-physical system domain is not sufficient to address this problem due to the fact that, when deployed individually to the thousands of sensors, most of the existing techniques suffer from scalability problems and resource overheads as many IoT sensors are power and resource-constrained.

*Contribution of this paper*: In this paper, we model continuous and interim FDIAs on IoT sensors and show their impact on a PdM model by performing a case study on aircraft Predictive Maintenance (PdM) system. We use the C-MAPSS [19] (Commercial Modular Aero-Propulsion System Simulation) dataset[1]. At first, to build an accurate predictive model, we train the Long Short-Term Memory (LSTM), Gated recurrent unit (GRU), and Convolutional neural network (CNN) algorithms using the C-MAPSS dataset. We evaluate these three predictive models, and the obtained results show that GRU-based model predicts the RUL[2] most accurately. The obtained results from the GRU-based model outperforms the recent works that uses DL for RUL prediction using the C-MAPSS dataset [20]–[22]. Afterward, we model two types of false data injection attack (FDIA) on the C-MAPSS dataset and evaluate their impact on CNN, LSTM, and GRU-based PdM models. To be more realistic, we model attack only on 3 sensors among the 21 sensors in the turbofan engine. The obtained results show that all the PdM models are greatly defected by the FDIA even if only 3 out of the 21 sensors are attacked. However, the GRU-based PdM model is comparatively more accurate and resilient to FDIA when compared to the other evaluated PdM models. In terms of sensitivity, we also explore that CNN is way more sensitive to FDIAs when compared to the LSTM and GRU. This is indeed an important observation since CNN-based techniques are quite popular in asset maintenance [23]–[25] and our results indicate that special measures should be taken for designing a CNN-based PdM. Afterward, we analyze the GRU-based PdM model using four different sequence lengths. The obtained results show an interesting relationship between the accuracy, the resiliency and the sequence length of the models. To the best of our knowledge, this is the *first work* that demonstrates the effects of IoT sensor attacks on a deep learning-enabled PdM system.

*Paper organization*: The rest of the paper is organized as follows. Section II briefly discusses the Engine Health Monitoring (EHM) systems and Predictive Maintenance (PdM). Section III introduces LSTM, GRU, CNN, and the NASA C-MAPSS dataset used for our experiment. Section IV describes in detail about an FDIA, ways of modeling it and also attack scenario. Section V compares the performance of CNN, LSTM and CNN in predicting RUL and analyses the impact on RUL prediction of CNN, LSTM, and CNN after the both continuous and interim FDIA. Section VI presents the observations from those obtained results, and Section VII concludes the paper.

## II. Engine health monitoring (EHM) system

### A. *Engine health monitoring (EHM) system*

An aircraft engine is a complex system, so it requires adequate monitoring to ensure safe operation and in-time maintenance [26]. Several displays and dials in the cockpit give different measurements like exhaust gas temperatures, engine pressure ratio, the pressure at fan inlet, rotational speeds, etc. All these parameters are crucial in indicating the health of the engine; they serve as early indicators of failure and prevent costly component damage. In order to accomplish the task of monitoring these parameters in an engine, Engine health monitoring (EHM) systems [27] have been in service for three decades. Fig. 1 shows a generic EHM architecture. An EHM system has several IoT (Internet of Things) sensors mounted inside and outside of an engine to monitor different parameters. All these IoT sensors are connected to a wireless network [28], which uses radio frequency for transmitting sensor output to central engine control [28]. These IoT sensors monitor different parameters of an aircraft engine and sends out alerts to the engine manufacturer if the Remaining Useful Life (RUL) [29] of the engine is approaching its end of life. [3] An EHM system employs PdM systems to predict the RUL using the data collected from the IoT sensors.

The sensors on-board the engine send time series data (cycles) every hour to the local storage on-board the airplane. After every $N_b$ cycles of data are captured, the data is transmitted to the ground station. At the ground station, the incoming live data is stored in the database and sent to PdM system to predict RUL of the engine. The PdM system sends out alerts if the predicted RUL is less than the permissible safe operation RUL of the engine.

[1] a popular turbofan engine degradation dataset published by NASA's Prognostics Center of Excellence (PCoE)

[2] Remaining useful life (RUL) is the length of time a machine is likely to operate before it requires repair or replacement.

[3] Remaining useful life (RUL) is the length of time a machine is likely to operate before it requires repair or replacement.

### *B. Predictive maintenance (PdM)*

In manufacturing supply chains, unexpected failures are considered as primary operational risk as it can hinder productivity and can incur huge losses. For example, in the modern automotive industry, an assembly line has several robots working on a car, and even if one of the robots fails, it will result in the total halt of the assembly line, causing loss of valuable production time and money. To overcome this problem, PdM strategies are employed. PdM is an industry 4.0 solution, which assists in predicting the future state of physical assets. It helps in better-informed maintenance decisions, to prevent unexpected delays. PdM systems are employed in major industries like Nuclear power plants, aviation industry, automotive industry, and health care services. PdM allows for convenient scheduling of corrective maintenance as parts for the equipment can be ordered beforehand to avoid the last-minute hassle, which saves a lot of valuable production time. PdM is well suited for making an informed decision when dealing with time-series data. A data-driven model of PdM employ some of the remarkable strategies like Random Forest algorithm [30], Artificial Neural Networks (ANN) [31], fuzzy models [32], Big data frameworks [33]. In this paper, three deep learning algorithms, specifically, LSTM, GRU, and CNN are employed in predicting RUL of an aircraft engine.

## III. DL ALGORITHMS FOR RUL PREDICTION

As mentioned earlier, RUL can be predicted using different ML algorithms. For this paper, we utilize LSTM, GRU, and CNN algorithms and compare their performance.

### *A. Long short-term memory model (LSTM)*

An LSTM [34] is a special kind of Recursive Neural Network (RNN), capable of learning long-term dependencies. LSTM is explicitly designed to avoid long term dependency problem, which is prevalent in RNN. It has achieved great praise in the field of machine learning and speech recognition. Some of the neural networks have a dependency problem, but an LSTM can overcome the problem of dependency by controlling the flow of information using input, output and forget gate. The input gate controls the flow of input activation into the memory cell. The output gate controls the output flow of cell activation into the rest of the network.

Suppose that training data has $N$ equipment of the same make and type that provide failure data, and each equipment provides set multivariate time-series data from the sensors of the equipment. Also, assume that there are $r$ sensors of the same type on each equipment. Then data collected from each equipment can be represented in a matrix form $X_n = [x_1, x_2, ..., x_t, ..., x_{T_n}] \in \mathbb{R}^{r \times T_n}$ $(n = 1, ..., N)$ where $T_n$ is time of the failure and at time $t$ the $r$-dimensional vector of sensor measurements is $x_t = [s_t^1, ..., s_t^r] \in \mathbb{R}^{r\times 1}, t = 1, 2, ..., T_n$. The data of each equipment in $X_n$ is fed to LSTM network and the network learns how to model the whole sequence with respect to target RUL. At time $t$, LSTM network takes $r$-dimensional sensor data $x_t$ and gives predicted $RUL_t$.

Let the LSTM cell has $q$ nodes, then $c_t \in \mathbb{R}^{q\times 1}$ is output of cell state, $h_t \in \mathbb{R}^{q\times 1}$ is output of LSTM cell, $o_t \in \mathbb{R}^{q\times 1}$ is output gate, $i_t \in \mathbb{R}^{q\times 1}$ is input gate, and $f_t \in \mathbb{R}^{q\times 1}$ is forget gate at time $t$. At time $t-1$, the output $h_{t-1}$, and hidden state $c_{t-1}$ will serve as input to LSTM cell at time $t$. The input $x_t$ is fed as input to the cell. In LSTM, the normalized data are calculated using the following equations:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \tag{1}$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f), \tag{2}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \tag{3}$$

$$\widetilde{c}_t = act(W_c \cdot [h_{t-1}, x_t] + b_c), \tag{4}$$

$$c_t = f_t * c_{t-1} + i_t * \widetilde{c}_t, \tag{5}$$

$$h_t = o_t * act(c_t), \tag{6}$$

Where $\sigma$ is the sigmoid layer. $c_t$ and $\widetilde{c}_t$ are each internal memory cell and temporary value to make a new internal memory cell at time t. $*$ is element-wise multiplication of two vectors.

### *B. Gated recurrent unit (GRU)*

The GRU was proposed by *Cho et al.* [35]. It operates using a reset gate and update gate. GRUs are improved version of standard recurrent neural network. Similar to the LSTM unit, the GRU has gating units that modulate the flow of information, however, without having a separate memory cell. GRU's performance on certain tasks of polyphonic music modeling and speech signal modeling was found to be similar to that of LSTM. GRUs have been shown to exhibit even better performance on certain smaller datasets [36]. The memory block of GRU is simpler than that of LSTM. The forget, input and output gates are replaced with an update and a reset gate. Also, GRU combines the hidden state and the internal memory cell. In GRU, the normalized data are calculated using the following equations:

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z), \tag{7}$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r), \tag{8}$$

$$\widetilde{h}_t = act(W \cdot [r_t * h_{t-1}, x_t] + b_h), \tag{9}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \widetilde{h}_t, \tag{10}$$

where $z_t$ and $r_t$ are the update gate and reset gate at time $t$, respectively. $\widetilde{h}_t$ is a temporary value to make new hidden state at time $t$.

### *C. Convolutional neural network (CNN)*

CNN a deep learning algorithm has achieved exceptional success in various research fields [37] because it has many advantages over traditional machine learning approaches such as MLP [38]. CNNs are fundamentally inspired from feed-forward ANNs. Like any other advanced DL algorithm, CNN also find their applications in different areas including CNN-based PdM system [23]–[25]. A CNN consists of one or more convolutional layers and then followed by one or more fully connected layers as in a standard multi-layer neural network. A 1D CNN model is utilized in this paper to predict the RUL of the engine. Details about CNN construction and network design are presented in detail in [39].

### D. C-MAPSS dataset

To evaluate the performance of the CNN, LSTM, and GRU DL algorithms, we use a well-known dataset, NASA's turbofan engine degradation simulation dataset C-MAPSS (Commercial Modular Aero-Propulsion System Simulation). This dataset includes 21 sensor data with different number of operating conditions and fault conditions[4]. In this dataset, there are four sub-datasets (FD001-04). Every subset has training data and test data. The test data has run to failure data from several engines of the same type. Each row in test data is a time cycle which can be defined as an hour of operation. A time cycle has 26 columns where the 1st column represents engine ID, and the 2nd column represents the current operational cycle number. The columns from 3 to 5 represent the three operational settings and columns from 6-26 represent the 21 sensor values. The time series data terminates only when a fault is encountered. For example, an engine with ID 1 has 192 time cycles of data, which means the engine has developed a fault at the 192nd time cycle. The test data contains data only for some time cycles as our goal is to estimate the remaining operational time cycles before a fault.

## IV. Modeling of FDIA

In this section, we describe in detail about an FDIA, ways of modeling it and also attack scenario.

**False data injection attack (FDIA):** As mentioned earlier, false data injection attack (FDIA) [5] can be injected into the system by compromising physical sensors, sensor data communication links, and data processing programs. Compromising physical sensors requires physical access to the sensors and hence is a tedious task. In contrast, hacking the sensor data communication links and data processing programs is an easier option for an attacker (explained in detail in the *attack surface* section). A successful FDIA can cause the engine sensors to output erroneous values to the central engine control, and thus make either physical or economic impact on the predictive maintenance model. For example, $X_i$ represents the information transmitted by the $i^{th}$ sensor. In an FDIA, the adversary contaminates the original vector with a vicious vector. Let $X_i = [x_1, x_2, ..., x_k]$ be the original vector data containing $k$ sensor reading for the $i^{th}$ sensor. The original vector could be contaminated by adding an FDIA vector with the same dimension as the original vector. Let the contaminated vector for the $i^{th}$ sensor be $F_i = [\lambda_1, \lambda_2, ..., \lambda_k]$, then the compromised vector is given by Eq. 11.

$$Z_i = X_i + F_i \tag{11}$$

An FDIA can be *constrained*, where the attacker has access to a limited number of sensors, and some part of the communication network, and an FDIA can also be *unconstrained*, where the attacker has access to all of the sensors and also has total control of the communication network. In this work, we consider the constrained attack since it is more practical that an attacker has access to only limited number of sensors (for the case study, the attack scenario considers only 3 sensors from a total of 21 sensors). We model two variations of FDIAs to explore and compare their impact, specifically, *continuous FDIA* and *interim FDIA*. In the case of continuous FDIA, the attack is continuous, which means, once the attack starts, from that point on-wards all the sensor reading are compromised. For instance, if the attack starts at the time instant $atck\_start = 3$ and ends at $atck\_end$ then $F_i$ can be expressed as $F_i = [\lambda_1, \lambda_2, \lambda_{atck\_start}, ..., \lambda_{atck\_end}]$, where $atck\_start \geq 1$ and $atck\_end = k$ . In the case of interim FDIA, the duration of attack is a short time interval, where $atck\_start > 1$ and $atck\_end < k$.

**Attacker's stealthiness:** An FDIA can be stealthy if it is not detected by the defense mechanism. In order to achieve that objective, the attack vector should remain in the boundary conditions of the sensor measurements. There exist constant vectors $Z_{min}$ and $Z_{max}$, such that for any FDIA vector $Z_i$, the compromised vector passes undetected through the defense if

$$Z_i = X_i + F_i \ and \ Z_{min} \leq Z_i \leq Z_{max} \tag{12}$$

We assume the attacker knows $Z_{min}$ and $Z_{max}$ to construct attack vectors satisfying Eq.12. Such information is easily available from the sensor data sheets provided by the vendor.

**Attacker's objective:** The attacker's objective is to cause delay in aircraft engine maintenance. This objective can be achieved by altering the IoT sensors readings that are fed to the PdM systems. Injecting false data to the sensor readings result in incorrect predictions from PdM systems which in turn results in delay of timely maintenance. As timely maintenance is a crucial factor of engine performance, lapse of maintenance may result in mid-air engine failures which are catastrophic.

One can argue that the attacker having access to the physical senors or the communication network of the sensors would directly attack the main systems (flight navigation and instrument landing systems) rather than just altering the sensor values for the PdM. However, their is a higher chance that a direct attack on the main system will easily get detected by the defence mechanisms. In contrast, introducing FDIA to sensors is an easier and safer option for an attacker since such attacks are more stealthy, hard to get detected as they are in sensor's acceptable range and also the impact on the aircraft does not show up immediately. Instead, it causes the erroneous calculation of RUL and delays the maintenance cycle leading to a catastrophic incident.

**Attack surface:** In this paper only the *constrained attacks* are considered. Note, one of the ways to launch an FDIA is using spoofing techniques. For instance, Tippenhauer *et al.* [41] showed a spoof attack scenario on GPS-enabled devices. In this attack scenario, a forged GPS signal is transmitted to the device to alter the location. In this way, the true location of

[4] More details about these 21 sensors can be found in [40]

the device is disguised and the attacker can perform a physical attack on the device. In another work, Giannetos *et al.* [42] introduced an app named *Spy-sense*, which monitors behaviors of several sensors in a device. The app can manipulate sensor data by deleting or modifying it. *Spy-sense* exploits the active memory region in a device and relays sensitive data covertly. These works show that FDI attacks can be performed even without gaining direct access to a system.

One of the recent articles [43] considers cyber-attacks as one of the reasons behind the two recent Boeing 737 Max 8 crashes. According to that article, a passenger, vehicle or drone carrying a sonic device capable of impacting the MCAS sensor controlling the plane could have been responsible for such an attack. Recently, ICS-CERT published an alert on certain controlled area network (CAN) bus systems aboard aircraft that might be vulnerable to hacking. It cited a report that an attacker with access to the aircraft could attach a device to an avionics CAN bus to inject false data, resulting in incorrect readings in an avionic equipment [44]. Using such a device attached to the bus could lead to incorrect engine telemetry readings, incorrect compass and attitude data, and incorrect altitude, airspeed, and angle of attack (AoA) data. Pilots might not be able to distinguish between false and legitimate readings. This alert explores the possibility of injecting false data into IoT sensor readings of aircraft engine which are transmitted on a CAN. In this work we consider FDIA using malicious device attached to a avionics CAN.

**Attack scenario:** As shown in Fig. 1 of the EHM architecture, the aircraft sends $N_b$ cycles of data at a time to the ground station/engine manufacturer. At the ground station, the PdM system performs data analytics on the received data and send out alerts if the RUL is close to the threshold $N_{th}$. The value of $N_{th}$ can vary from engine to engine, and it is manufacturer-dependant. An adversary having this knowledge can perform the attacks more effectively. In a more practical sense, the degradation of the engine is very negligible at the beginning, but as time proceeds, the degradation follows a linear trend, and it increases as the engine approaches the end of life. Assuming in an engine, the linear degradation initially starts at $N^d$ cycle. The value of $N^d$ is different for different engines, as the wear of the engines may be different. If the average of $N^d$ for all the engines in the dataset is taken, it is found to be $N^d_{avg}$. An adversary having the knowledge of $N^d_{avg}$ can perform the attacks after the degradation initiates, making the attack more destructive. To study the impact of FDIA on PdM systems, we consider an attack scenario where the attacker has access to the aircraft and could attach a device to an avionics CAN bus [44] as mentioned previously in section 4 (attack surface). The device attached to CAN bus can inject false data into engine sensor readings, resulting in incorrect predictions of RUL of the aircraft engine. Note, as mentioned in section (attack surface), it is also possible to launch an FDI attack without a direct access to the aircraft by using the sensor spoofing technique [45], or using a drone carrying a special device capable of interfering and impacting the on-board aircraft sensor measurements [43]. In this work, we consider two variations of FDIA which are continuous and interim FDIA. In continuous FDIA, the attack is initiated after $N^d$ and continues to the end of life of the engine. In Interim FDIA, the attack is initiated after $N^d$ and continues to the next 20 time cycles. In both the variations of FDIA, random and biased FDIAs are used to evaluate the PdM model's performance. Here, random FDIA means the noise added to the sensor output has a range (0.01% to 0.05%). Whereas, biased FDIA has constant amount of noise added to the sensor output.

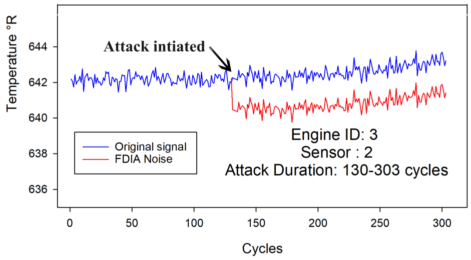


Fig. 2: Continuous FDIA signature

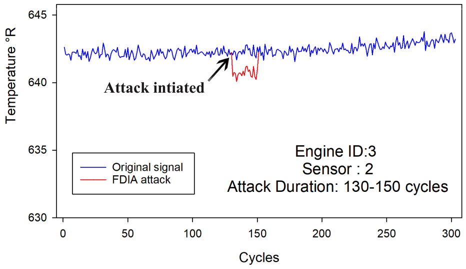


Fig. 3: Interim FDIA signature

TABLE I: Parameter settings

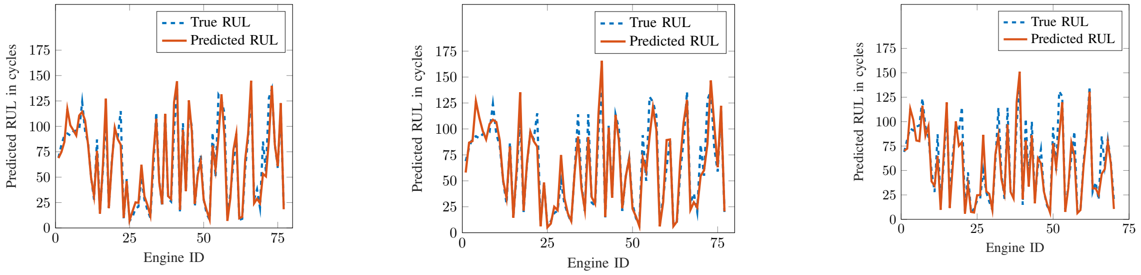
| Model | Hidden neuron | Dropout | Batch size | Epochs | Act. func. |
|---|---|---|---|---|---|
| CNN | 64 | 0.2 | 200 | 100 | relu |
| LSTM | 100 | 0.2 | 200 | 100 | tanh |
| GRU | 100 | 0.2 | 200 | 100 | tanh |

## V. Experimental Results

In Section V-A, we compare different DL algorithms on RUL prediction. In Section V-B, we present both continuous and interim FDIA signatures, and impact of attacks on the RUL prediction. Section V-C and V-D present piece-wise RUL prediction, and impact of sequence length on resiliency, respectively.

### *A. Comparison of deep learning algorithms*

In order to select the best machine learning algorithm for the PdM, we compare LSTM, GRU, and CNN algorithms for the C-MAPSS dataset. To evaluate the performance of the predictors, we utilize the root mean square error (RMSE) metric which is widely used as an evaluation metric in model evaluation studies. Fig. 4 and Table II represents the comparison of DL algorithms with architectures LSTM(100,100,100,100)



(a) GRU(100,100,100), lh(80), RMSE=7.26 (b) LSTM(100,100,100,100), lh(80), RMSE=8.76 (c) CNN(64,64,64,64), lh(100), RMSE=9.94

Fig. 4: Comparison of deep learning algorithms

lh(80), GRU(100,100,100) lh(80), and CNN(64,64,64,64) lh(100). The notation GRU(100,100,100) lh(80) refers to network that has 100 nodes in the hidden layers of the first GRU layer, 100 nodes in the hidden layers of the second GRU layer, 100 nodes in the hidden layers of the third GRU layer, and a sequence length of 80. At the end, there is a 1-dimensional output layer. Table I gives summary of all the parameters used for CNN, LSTM and GRU.

TABLE II: RMSE comparison for different DL algorithms

| Predictor architecture | RMSE |
|---|---|
| | Test |
| CNN(64,64,64.64) lh(100) | 9.94 |
| LSTM(100,100,100,100) lh(80) | 8.76 |
| GRU(100,100,100) lh(80) | 7.26 |

From Fig. 4 and Table II it is evident that the DL algorithm GRU(100, 100, 100) with a sequence length 80 has the least RMSE of 7.26. It means that GRU is very accurate in predicting accurate RUL for this dataset. Note, the obtained results in Table II show that the GRU performs better when compared to the recent works that uses deep learning for RUL prediction using C-MAPSS dataset [20], [22], [46]. In the next step, we model FDIA on CNN, LSTM, and GRU to evaluate their resiliency to the FDIA.

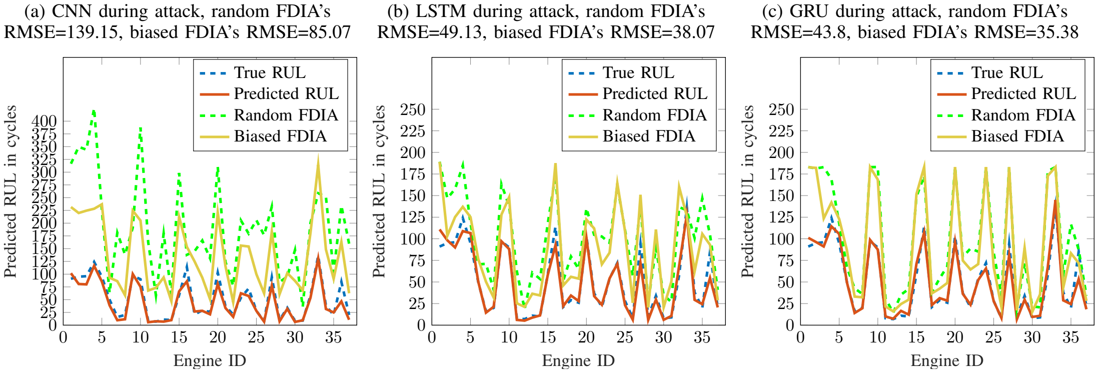
### B. Impact of attacks on a PdM system

The average degradation point of the engine $N^d_{avg}$ is considered as 130 for the FD001 dataset [47] [48] [49], and we assume that the Engine Health Monitoring (EHM) system of the aircraft sends 20 time cycles ($N_b$) of data to the ground at a time. The details of EHM, the parameter $N_b$ and how the data is sent to the ground is discussed in Fig. 1. The train and test dataset have 21 sensor data. The FDIA can be performed on 21 sensors, but to make the attack more realistic, we perform FDIA on only 3 sensors (specifically, T24, T50, and P30). In FDIA continuous scenario, the attacker has initiated the attacks after $N^d_{avg}$, which is 130 time cycles (one time cycle is equivalent of one flight hour), and the attack duration is until end of life of the engine. In FDIA interim scenario, the attacker has initiated the attacks after $N^d_{avg}$, which is 130 time cycles, and the attack duration is 20 hours (20 time cycles). Since the attack is initiated after 130 time cycles, we only consider the engines which has data for more than 130 cycles which gives us 37 engines in the FD001 dataset. The resultant dataset is re-evaluated using the LSTM, CNN and GRU-based PdM models and the obtained RMSEs are 6.09, 7.50, and 5.36, respectively.

**FDIA signature:** To model the FDIA on sensors, we add a vicious vector to the original vector, which modifies the sensor output by a very small margin (0.01% to 0.05%) for random FDIA and 0.02% for biased FDIA. Here, random FDIA means the noise added to the sensor output has a range (0.01% to 0.05%). Whereas, biased FDIA has constant amount of noise added to the sensor output. Fig. 2 shows the comparison between the original and FDIA attacked output signal of sensor 2 for engine ID 3 for continuous FDIA. In continuous FDIA, we attack the sensor output from time cycles 130 to the end of life of the engine. In the case of interim FDIA as shown in Fig. 3, the attack duration is only for 20 time cycles (130 to 150 time cycles). Note, in the constrained attack the adversary has limited access to sensors. As shown in Fig. 2 and 3, the attack signature is very similar to the original signal, making it stealthy and harder to detect even with common defense mechanisms in place.

**Impact of FDIA on CNN, LSTM and GRU:** To show the impact of an FDIA on the aircraft PdM system, we implement attack for the scenario mentioned previously in Section 4 (attack scenario). The FDIA is performed on three sensors (T24, T50, and P30) instead of attacking all the 21 sensors in the dataset. In FDIA continuous scenario, the adversary performs attacks from 130 time cycles to end of life of the engine. It is evident from Fig. 5 that LSTM, GRU, and CNN are greatly affected by the continuous FDI attack. In the case of random and biased FDIA, random FDIA showed considerable impact on all PdM models. The CNN based PdM model is the most affected by the continuous FDIA as random FDIA's RMSE is 139.15 and biased FDIA's RMSE is 85.07 (true RMSE is 7.50) which is almost 18 times and 11 times higher when compared to the true RMSE, respectively. In contrast, the GRU based PdM model is the least effected by the continuous FDIA as random FDIA's RMSE is 43.8 and biased FDIA's RMSE is 35.38 (true RMSE is 5.36). Eventhough, the GRU is least affected by both random and biased FDIA, their RMSE is 8 and 6 times higher than the true RMSE, respectively, making it also deadly for a PdM system.

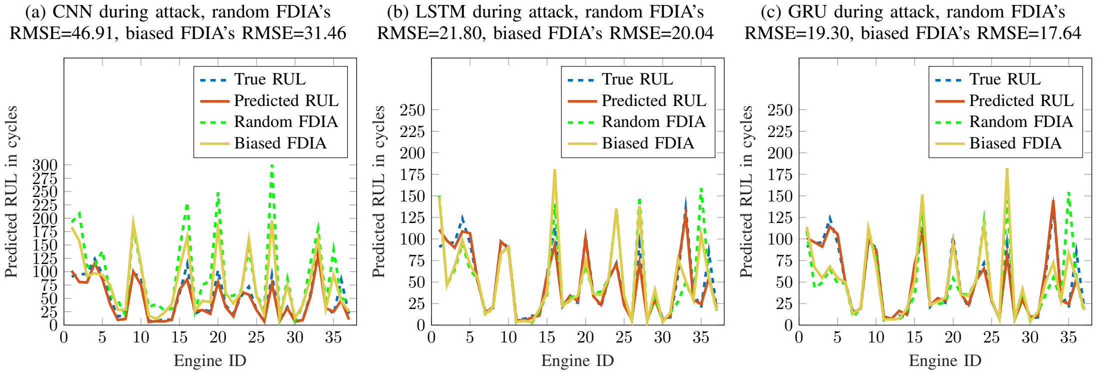
In FDIA interim scenario, the adversary performs attacks between 130 and 150 time cycles (20 time cycles). It is evident


(a) CNN during attack, random FDIA's RMSE=139.15, biased FDIA's RMSE=85.07

(b) LSTM during attack, random FDIA's RMSE=49.13, biased FDIA's RMSE=38.07

(c) GRU during attack, random FDIA's RMSE=43.8, biased FDIA's RMSE=35.38

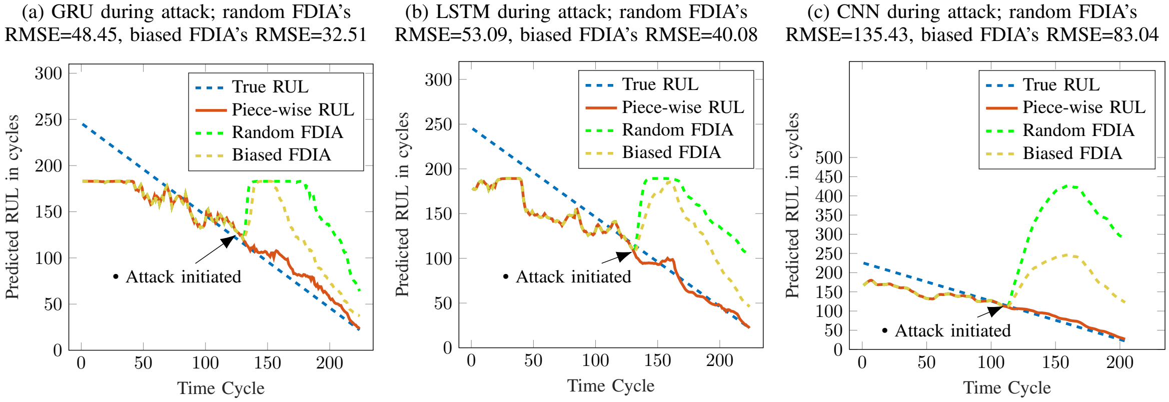
Fig. 5: FDI attack scenario for continuous period


(a) CNN during attack, random FDIA's RMSE=46.91, biased FDIA's RMSE=31.46

(b) LSTM during attack, random FDIA's RMSE=21.80, biased FDIA's RMSE=20.04

(c) GRU during attack, random FDIA's RMSE=19.30, biased FDIA's RMSE=17.64

Fig. 6: FDI attack scenario for interim period


(a) GRU during attack; random FDIA's RMSE=48.45, biased FDIA's RMSE=32.51

(b) LSTM during attack; random FDIA's RMSE=53.09, biased FDIA's RMSE=40.08

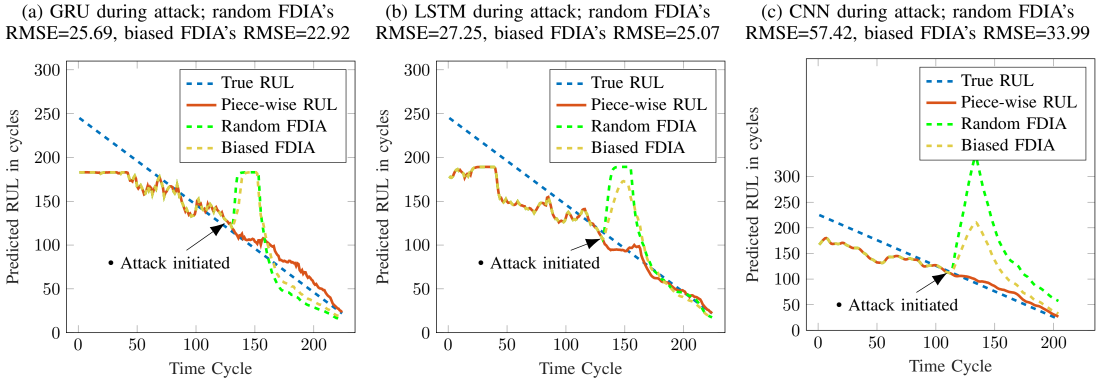
(c) CNN during attack; random FDIA's RMSE=135.43, biased FDIA's RMSE=83.04

Fig. 7: Piece-wise RUL prediction for continuous FDIA

from Fig. 6 that LSTM, GRU, and CNN are greatly affected by the interim FDI attack. Once again, the CNN based PdM model is the greatly affected by the continuous FDIA as random FDIA's RMSE is 46.91 and biased FDIA's RMSE is 31.46 (true RMSE is 7.50) which is almost 6 times and 4 times higher than the true RMSE, respectively. In contrast, the GRU based PdM model is the least effected by the interim FDIA as random FDIA's RMSE is 19.30 and biased FDIA's RMSE is 17.64 (true RMSE is 5.36). This indicates that GRU-based PdM models are comparatively resilient to both continuous and interim FDIA. Even though the GRU is least affected by both random and biased FDIA, their RMSE is still 4 times and 3 times higher than the true RMSE, respectively, making it deadly for a PdM system. When comparing both continuous and interim FDIA, it observed that continuous FDIA's RMSE is almost twice the interim FDIA's RMSE. Hence, continuous FDIAs are more potent than the interim FDIA.

### C. *Piece-wise RUL prediction*

In order to show the impact of FDIA attacks on a specific engine data, we apply the piece-wise RUL prediction. The piece-wise RUL prediction gives a better visual representation of degradation in an aircraft engine. Fig. 7(a) shows an example of an engine data from the dataset of 100 engines, and depicts the predicted RUL using GRU at each time step of that engine data. For example, if $X$ is the time series data of a particular engine, then $X_i = [x_1, x_2, x_3...x_{t-k}]$ represents time series data until time $t-k$. $RUL^p$ is predicted RUL at each time step in $X$, which is can be defined as $RUL_i^p = [RUL_1^p, RUL_2^p, RUL_3^p...RUL_{t-k}^p]$. From Fig. 7(a),



(a) GRU during attack; random FDIA's RMSE=25.69, biased FDIA's RMSE=22.92

(b) LSTM during attack; random FDIA's RMSE=27.25, biased FDIA's RMSE=25.07

(c) CNN during attack; random FDIA's RMSE=57.42, biased FDIA's RMSE=33.99

Fig. 8: Piece-wise RUL prediction for Interim FDIA

it is evident that as the time series approaches the end of life, the predicted RUL (red line) is close to the true RUL (blue dashes), because the DL model has more time series data to accurately predict the RUL.

In the case of piece-wise RUL prediction during continuous FDIA, it is observed from Fig. 7 that both random and biased FDIA are initiated from 130 time cycles to 242 time cycles for engine ID 17. Here, the green and yellow dashes in the figures are predicted RUL after random and biased FDIA, respectively. In the GRU, LSTM, and CNN based piece-wise RUL prediction (for both random and biased FDIA), the attacker initiates the FDIA after 130 time cycles. The impact of the attack is quite interesting as the RUL jumps upwards (around 200 for GRU and LSTM) with a possible indication to the engine maintenance operator that the engine is quite healthy. This may influence a 'no maintenance required' decision from the maintenance engineers point of view, however, in reality the RUL is decreasing continuously and going below the 100 time cycles which might require to schedule an urgent maintenance — leading to a catastrophic event. For CNN, the continuous FDIA causes a longer jump (even beyond the initial RUL value) when compared to the FDIA in LSTM and GRU. Of course their is a higher chance that this will be flagged as a potential fault either in the engine or in the PdM system, and will cause an unnecessary engine maintenance and will increase the aircraft downtown causing a financial loss to the flight operator.

In the case of piece-wise RUL prediction for engine ID 17 under interim FDIA, it is observed in Fig. 8 that the attack causes a similar jump as showed in the case of continuous FDIA in Fig. 7. However, the effect of the attack flushes away way sooner when compared to the continuous FDIA case. However, note that the attack duration was only 20 cycles, but it took more than 45 cycles to flush out the effect by the PdM system. Hence, if a maintenance is due around that period, it may lead to a catastrophic consequence. Once again, the piece-wise RUL prediction results indicates that employing CNN in PdM systems may result in systems that are very sensitive to the FDIA and hence special measures should be taken for designing a CNN-based PdM.

### D. Impact of sequence length on resiliency of GRU

Since GRU has performed best among the DL algorithms as shown in the experimental results in the previous subsections, in Fig. 9 we compare four different GRU networks under FDI attack. The GRU networks have structures GRU1(100,100,100) lh(90), GRU2(100,100,100) lh(80), GRU3(100,100,100) lh(70), and GRU4(100,100,100) lh(60). We observe that the GRU network with architecture GRU2(100,100,100) lh 80 has the least value of true RMSE (5.36), which means that it predicts RUL quite accurately, however, it is less resilient to both continuous and interim FDIA. In contrast, GRU with network architecture GRU3(100,100,100) lh(70) shows the second-best performance in predicting the RUL (RMSE of 6.89), however, in terms of resiliency, this network is the least affected by continuous and interim FDIA. This indeed shows an interesting insight that the sequence length affects not only the accuracy but also the resiliency of the model. It also indicates that accuracy should not be the only factor while designing a PdM system. For instance, in terms of accuracy GRU2 is the typical choice. However, if the accuracy and resiliency both factors are considered, GRU3 is can be an ideal choice (at the cost of losing some accuracy).

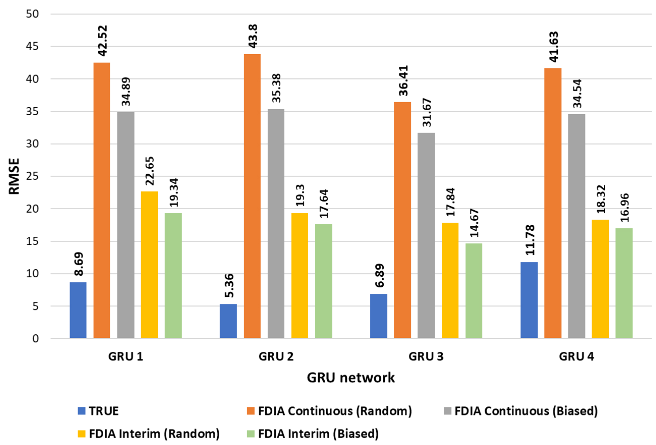


Fig. 9: RMSE comparison of different GRU networks

## VI. Discussion

In this work, we first evaluate different deep learning (DL) algorithms on C-MAPSS dataset and obtained results show a great prospect for deep learning in PdM. It is observed that sequence length and network architecture are crucial in predicting accurate RUL. Our work shows that the GRU performed better than some of the recent works that used deep learning on C-MAPSS dataset [20], [22], [46].

The impact analysis of FDIA on aircraft sensors in the C-MAPSS dataset provides some interesting insights. We observe that CNN based PdM model is greatly affected by both random and biased FDIA. In the case of interim FDIA, CNN's random and biased RMSE are 18 and 11 times higher than the true RMSE, respectively, and in the case of continuous, the random and biased RMSE are 6 and 4 times higher than the true RMSE, respectively. We also observe that GRU-based PdM model is more resilient to both random and biased in comparison with CNN and LSTM-based PdM models. Even though, the GRU is least affected by both random and biased FDIA, their RMSE is 8 and 6 times higher than the true RMSE in the case of continuous FDIA, respectively. In the case of interim FDIA, the random and biased RMSE are 4 and 3 times higher than the true RMSE, respectively, making it disastrous for the PdM system. This may result in the delay of timely maintenance for the aircraft engine and eventually result in engine failure at some point. Note, the attack signature of FDIA is very close to the original sensor output (within the boundary conditions of the sensor measurements) making it harder to be detected by common defense mechanisms in an engine health monitoring (EHM) system.

A piece-wise RUL predicting approach is used in visualizing the impact of attacks on the sensors, which clearly shows that PdM system is susceptible to sensor attacks. While designing of PdM systems, the engineers should take both continuous and interim FDI attacks into consideration. CNN based piece-wise RUL prediction results show that special measures should be taken when designing and adopting CNN-based PdM systems (such as the cases in [23]–[25], [50]) as they are very sensitive to the FDIA. Fig.9, gives an interesting insight into the relationship between accuracy and resiliency of the GRU network. It shows the need for considering the relationship between accuracy, resiliency and sequence length of a DL mode (such as GRU in our case) in the design phase. Indeed, such an analysis can serve as empirical guidance to the development of subsequent data-driven PdM systems.

All of these obtained results show that DL-based PdM systems have a great prospect for aircraft maintenance, however, they are very susceptible to sensor attacks. Hence it is required to investigate proper detection techniques to detect such stealthy attacks and special care should be taken when manufacturing IoT sensors for DL/AI applications. For the same reason, while designing a PdM system, the designer also must consider the resiliency of the DL algorithm instead of just emphasizing on the algorithm's accuracy, as we investigated in this paper.

## VII. Conclusions and Future Works

This paper compares the performance of LSTM, GRU, and CNN for RUL prediction using the C-MAPSS dataset, and explores the impacts of continuous an interim false data injection attacks on these deep learning algorithms. We observe that the GRU is a better suited DL technique when compared to LSTM and CNN in terms of accuracy. Our obtained results show that both continuous and interim FDIA have a substantial impact on the RUL prediction even if only a few of the IoT sensors are attacked. We also observed that the GRU-based PdM model is more resilient to FDIA, whereas CNN is dramatically sensitive to both continuous and interim FDIA. Finally, we explored that there exists a relationship between the accuracy and sequence length in the GRU-based PdM model which can serve as empirical guidance to the development of data-driven PdM systems. In the future, we plan to develop an end-to-end methodology for the detection and mitigation of sensor attacks in a PdM system.

## References

[1] "Predictive maintenance benefits for the freight logistics industr," Available: https://www.ibm.com/downloads/cas/AVNOLWQW.

[2] M. A. der Mauer, T. Behrens, M. Derakhshanmanesh, C. Hansen, and S. Muderack, "Applying sound-based analysis at porsche production: Towards predictive maintenance of production machines using deep learning and internet-of-things technology," in *Digitalization Cases*. Springer, 2019, pp. 79–97.

[3] A. K. Sikder, G. Petracca, H. Aksu, T. Jaeger, and A. S. Uluagac, "A survey on sensor-based threats to internet-of-things (iot) devices and applications," *arXiv preprint arXiv:1802.02041*, 2018.

[4] "Cyber-attacks On Smart Factories Are On The Rise." Available: https://smartmachinesandfactories.com/news/fullstory.php/aid/459/Cyber-attacks_on_smart_factories_are_on_the_rise.html.

[5] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using kalman filter," *IEEE transactions on control of network systems*, vol. 1, no. 4, pp. 370–379, 2014.

[6] "The Rolls-Royce Intelligent Engine — Driven by data," Available: https://www.rolls-royce.com/media/press-releases/2018/06-02-2018-rr-intelligentengine-driven-by-data.aspx.

[7] "Predictive digonostics, Bosch," Available: https://www.bosch-mobility-solutions.com/en/products-and-services/mobility-services/predictive-diagnostics/.

[8] "Transforming Railroad Asset Management: Going Smart with Predictive Maintenance," Available: https://www.tcs.com/content/dam/tcs/pdf/Industries/travel-and-hospitality/Transforming-Railroad-Asset-Management.pdf.

[9] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 4, pp. 1630–1638, 2016.

[10] B. Li, R. Lu, W. Wang, and K.-K. R. Choo, "Distributed host-based collaborative detection for false data injection attacks in smart grid cyber-physical system," *Journal of Parallel and Distributed Computing*, vol. 103, pp. 32–41, 2017.

[11] Y. Guan and X. Ge, "Distributed attack detection and secure estimation of networked cyber-physical systems against false data injection attacks and jamming attacks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 1, pp. 48–59, 2017.

[12] "Artificial intelligence for predictive maintenance, Aerospace Manufacturing and Design Magazine," Available: https://www.aerospacemanufacturinganddesign.com/article/artificial-intelligence-for-predictive-maintenance/.

[13] "USAF Launches Predictive Maintenance For Three Fleets," Available: https://aviationweek.com/defense/usaf-launches-predictive-maintenance-three-fleets.

[14] "The US Air Force Is Adding Algorithms to Predict When Planes Will Break, Defense One Magazine." Available: https://www.defenseone.com/business/2018/05/us-air-force-adding-algorithms-predict-when-planes-will-break/148234/.

[15] "IOT Use Cases and Innovation in IOT," Available: https://medium.com/@billsoftnet/iot-use-cases-and-innovation-in-iot-6b4e49fbc9dc.

[16] "Maintaining the data-rich Pratt Whitney GTF engine," Available: https://www.sae.org/news/2018/10/maintaining-the-data-rich-pratt--whitney-gtf-engine.

[17] Bruce Jackson, "Threat of a remote cyberattack on today's aircraft is real," 2019, [Online; accessed 01-07-2019]. [Online]. Available: https://www.darkreading.com/iot/threat-of-a-remote-cyberattack-on-todays-aircraft-is-real/a/d-id/1333551

[18] David Cenciotti, "Cybersecurity in the sky: Internet of things capabilities making aircraft more exposed to cyber threats than ever before," 2017, [Online; accessed 20-June-2017]. [Online]. Available: https://theaviationist.com/2017/06/20/cybersecurity-in-the-sky-internet-of-things-capabilities-to-make-aircraft-more-exposed-to-cyber-threats-than-ever-before/

[19] A. Saxena and K. Goebel, "C-mapss data set," *NASA Ames Prognostics Data Repository*, 2008.

[20] C. Zheng, W. Liu, B. Chen, D. Gao, Y. Cheng, Y. Yang, X. Zhang, S. Li, Z. Huang, and J. Peng, "A data-driven approach for remaining useful life prediction of aircraft engines," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 184–189.

[21] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*. IEEE, 2017, pp. 88–95.

[22] A. L. Ellefsen, E. Bjørlykhaug, V. Æsøy, S. Ushakov, and H. Zhang, "Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture," *Reliability Engineering & System Safety*, vol. 183, pp. 240–251, 2019.

[23] W. Silva, "Cnn-pdm: A convolutional neural network framework for assets predictive maintenance," 2019.

[24] T. Huuhtanen and A. Jung, "Predictive maintenance of photovoltaic panels via deep learning," in *2018 IEEE Data Science Workshop (DSW)*. IEEE, 2018, pp. 66–70.

[25] R. Caponetto, F. Rizzo, L. Russotti, and M. Xibilia, "Deep learning algorithm for predictive maintenance of rotating machines through the analysis of the orbits shape of the rotor shaft," in *International Conference on Smart Innovation, Ergonomics and Applied Human Factors*. Springer, 2019, pp. 245–250.

[26] K. Hünecke, *Jet engines: Fundamentals of theory, design and operation*. Airlife, 1997, no. BOOK.

[27] I. Tumer and A. Bajwa, "A survey of aircraft engine health monitoring systems," in *35th Joint Propulsion Conference and Exhibit*, 1999, p. 2528.

[28] H. Bai, M. Atiquzzaman, and D. Lilja, "Wireless sensor network for aircraft health monitoring," in *First International Conference on Broadband Networks*. IEEE, 2004, pp. 748–750.

[29] X.-S. Si, W. Wang, C.-H. Hu, and D.-H. Zhou, "Remaining useful life estimation–a review on the statistical data driven approaches," *European journal of operational research*, vol. 213, no. 1, pp. 1–14, 2011.

[30] A. P. Verma, "Performance monitoring of wind turbines: a data-mining approach," 2012.

[31] P. Bangalore and L. B. Tjernberg, "An approach for self evolving neural network based algorithm for fault prognosis in wind turbine," in *2013 IEEE Grenoble Conference*. IEEE, 2013, pp. 1–6.

[32] S. Simani, S. Farsoni, and P. Castaldi, "Fault tolerant control of an offshore wind turbine model via identified fuzzy prototypes," in *2014 UKACC International Conference on Control (CONTROL)*. IEEE, 2014, pp. 486–491.

[33] M. Canizo, E. Onieva, A. Conde, S. Charramendieta, and S. Trujillo, "Real-time predictive maintenance for wind turbines using big data frameworks," in *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*. IEEE, 2017, pp. 70–77.

[34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[35] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[36] R. Zhao, D. Wang, R. Yan, K. Mao, F. Shen, and J. Wang, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1539–1548, 2017.

[37] A. Bhandare, M. Bhide, P. Gokhale, and R. Chandavarkar, "Applications of convolutional neural networks," *International Journal of Computer Science and Information Technologies*, vol. 7, no. 5, pp. 2206–2215, 2016.

[38] R. R. Swain and P. M. Khilar, "A fuzzy mlp approach for fault diagnosis in wireless sensor networks," in *2016 IEEE region 10 conference (TENCON)*. IEEE, 2016, pp. 3183–3188.

[39] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, "Real-time motor fault detection by 1-d convolutional neural networks," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7067–7075, 2016.

[40] E. Ramasso and A. Saxena, "Performance benchmarking and analysis of prognostic methods for cmapss datasets." *International Journal of Prognostics and Health Management*, vol. 5, no. 2, pp. 1–15, 2014.

[41] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun, "On the requirements for successful gps spoofing attacks," in *Proceedings of the 18th ACM conference on Computer and communications security*. ACM, 2011, pp. 75–86.

[42] T. Giannetsos and T. Dimitriou, "Spy-sense: spyware tool for executing stealthy exploits against sensor networks," in *Proceedings of the 2nd ACM workshop on Hot topics on wireless network security and privacy*. ACM, 2013, pp. 7–12.

[43] Lee Neubecker, "Could a sonic weapon have caused the two recent boeing 737 max 8 crashes?" 2019, [Online; accessed 22-March-2019]. [Online]. Available: https://leeneubecker.com/sonic-weapon-attack-boeing/

[44] US Department of Homeland Security CISA Cyber + Infrastructure, "Can bus network implementation in avionics," 2019, [Online; accessed 13-September-2019]. [Online]. Available: https://www.us-cert.gov/ics/alerts/ics-alert-19-211-01

[45] A. J. Kerns, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, "Unmanned aircraft capture and control via gps spoofing," *Journal of Field Robotics*, vol. 31, no. 4, pp. 617–636, 2014.

[46] B. Wang, Y. Lei, N. Li, and T. Yan, "Deep separable convolutional network for remaining useful life prediction of machinery," *Mechanical Systems and Signal Processing*, vol. 134, p. 106330, 2019.

[47] F. O. Heimes, "Recurrent neural networks for remaining useful life estimation," in *2008 international conference on prognostics and health management*. IEEE, 2008, pp. 1–6.

[48] G. S. Babu, P. Zhao, and X.-L. Li, "Deep convolutional neural network based regression approach for estimation of remaining useful life," in *International conference on database systems for advanced applications*. Springer, 2016, pp. 214–228.

[49] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*. IEEE, 2017, pp. 88–95.

[50] N. Günnemann and J. Pfeffer, "Predicting defective engines using convolutional neural networks on temporal vibration signals," in *First International Workshop on Learning with Imbalanced Domains: Theory and Applications*, 2017, pp. 92–102.