SIEMS: A Secure Intelligent Energy Management System for Industrial IoT Applications

Article *in* IEEE Transactions on Industrial Informatics · April 2022

DOI: 10.1109/TII.2022.3165890

5 authors, including:

Pedram Asef
University of Bath
35 PUBLICATIONS **111** CITATIONS
SEE PROFILE

Rahim Taheri
University of Portsmouth
21 PUBLICATIONS **251** CITATIONS
SEE PROFILE

Mohammad Shojafar
University of Surrey
171 PUBLICATIONS **4,960** CITATIONS
SEE PROFILE

Rahim Tafazolli
University of Surrey
1,017 PUBLICATIONS **14,456** CITATIONS
SEE PROFILE

SIEMS: A Secure Intelligent Energy Management System for Industrial IoT Applications

Pedram Asef, *Senior Member, IEEE,* Rahim Taheri, *Member, IEEE,* Mohammad Shojafar *Senior Member, IEEE*, Iosif Mporas, and Rahim Tafazolli, *Senior Member, IEEE*

***Abstract*—Microgrids are industrial technologies that can provide energy resources for the Internet of things (IoT) demands in smart grids. Hybrid microgrids supply quality power to the IoT devices and ensure high resiliency in supply and demand for PV-based grid-tied microgrids. In this system, the usage of predictive energy management systems (EMS) is essential to dispatch power from different resources, whilst the battery energy storage system (BESS) is feeding the loads. In this work, we deploy a one-day-ahead prediction algorithm using a deep neural network for a fast-response BESS in an intelligent energy management system (I-EMS) that is called *SIEMS*. The main role of the *SIEMS* is to maintain the state of charge at high rates based on the one-day-ahead information about solar power, which depends on meteorological conditions. The remaining power is supplied by the main grid for sustained power streaming between BESS and end-users. Considering the usage of information and communication technology components in the microgrids, the main objective of this paper is focused on the hybrid microgrid performance under cyber-physical security adversarial attacks. Fast gradient sign, basic iterative, and DeepFool methods, which are investigated for the first time in power systems e.g. smart grid and microgrids, in order to produce perturbation for training data. To secure the microgrid's *SIEMS*, we propose *two* Defence algorithms based on defensive distillation and adversarial training strategies for the first time in EMSs. We apply and evaluate these benchmark adversarial attack and Defence methods against the proposed machine learning models to increase the robustness of the models in the system against adversarial attacks.**

***Index Terms*—Internet of things (IoT), Machine Learning, Hybrid Microgrid, Adversarial Attacks, Cyber-Physical Security.**

I. Introduction

HIGH demand resiliency of both AC or DC microgrids relies on how low the risk of instability is. One popular microgrid architecture is the grid-tied or hybrid, in which the main energy sources are renewables (PV, wind) along with the main grid. Due to the intermittent nature of wind velocity, solar irradiation, and other environmental parameters such as relative humidity, partial shadings, and air temperature, the green power generation can meet uncertainties [1], [2]. Hybrid energy generation and distribution systems require management strategies for optimal power flow distribution, in which the centralized [3], [4], decentralized [5], [6], and hybrid energy management systems (EMSs) using coupled dynamic programming and model predictive control (MPC) algorithms [7], intelligent optimization algorithms, such as particle swarm optimization (PSO) [8] and mixed integer linear programming (MILP) [9] are among the most popular schemes. Energy storage system as a flexible grid asset offer resiliency and stability in hybrid microgrids [10]. The state of charge (SOC) in battery energy storage systems (BESS)s which are supplied by the hybrid energy resources, such as solar power and the main grid, highly reduce the risk of instability towards meteorological variables such as solar irradiance, temperature, humidity, optoelectronic anemometer and digital weather vane for any type of microgrids. In PV-tied microgrids, balancing the power supply and demand is a challenge because of frequent changes in the weather conditions. The power unbalance (supply and demand) results in lack of power quality (varying voltage, frequency instability, sudden blackouts, etc.) until the grid power dispatched by the system operators and fully streamed in the microgrid. During the power dispatching from the main grid streaming to the end-users, the operation time is approximately a few minutes, the network instability can critically damage the grid resiliency. The demand resiliency advantages of microgrids have widely been studied in the literature, however, the analytical and empirical modeling of microgrids for resiliency targets based on BESS is limited. There have been a few significant works on dynamic energy management systems for autonomous microgrids [11], [12] for enhancing the resiliency can be found in the bibliography. Due to time-dependent solar irradiation fluctuations, a self-supporting intelligent microgrid is needed to continuously support the I-EMS. Such smart microgrids use predictive algorithms, i.e. daily-ahead, weekly-ahead, etc., to ensure that the BESS is well supplied and can satisfy the non-critical and critical load connected. While this review focuses on I-EMSs and their corresponded demand resiliency effects, closely related works include: T. Pippia et al. [13], studied a single-level rule-based model predictive control (RBMPC) scheme for optimization of the EMS of a hybrid microgrid, considering a range of different sampling times to observe how fast the dynamic of the proposed RBMPC is, and also validated the RBMPC algorithm's performance with the popular MILP method. In another work [12], the authors investigated a coordinated energy dispatch based on a multi-level distributed model predictive control (DMPC) for an autonomous microgrid, where the upper level offers an optimal

P. Asef and I. Mporas are with the School of Physics, Engineering & Computer Science, University of Hertfordshire, Hatfield, Hertfordshire, AL10 9AB, UK (e-mail: {p.asef, i.mporas}@herts.ac.uk)

R. Taheri is with the Centre for Telecommunications Research, King's College London, UK (e-mail: rahim.taheri@kcl.ac.uk)

M. Shojafar and R. Tafazolli are with the 5G & 6G Innovation Centre (5GIC/6GIC), University of Surrey, Guildford, Surrey, GU2 7XH, UK (e-mail: {m.shojafar, r.tafazolli}@surrey.ac.uk)

Manuscript received 19 December 2021; revised 28 February 2022 and 25 March 2022; accepted 05 April 2022

scheduling for energy exchange between the distribution network operator and microgrids, while the lower level provides tracking between supply and demand. They presented how the proposed DMPC balanced the supply and end-user demand in a cost-effective approach.

Motivations. In this study, an intelligent EMS (I-EMS) is implemented using deep neural networks (DNN)s algorithm for daily-ahead prediction, which influences demand resiliency through tracking the predictive supply and usage of both energy resources, i.e. the solar photovoltaic (PV) and the main grid. For this reason, an algorithmic-based cost-effective, in terms of accuracy and reliability, *I-EMS* is used to study the behavior of the local non-critical and critical AC loads, where solar PV and the main grid are the main energy resources of the network. The predictive *I-EMS* has deployed DNN which balances the SOC of the BESS based on the daily-ahead weather conditions, thereby the power availability of the BESS always remains in a safe mode condition to serve the critical loads with a very fast lead-in time power streaming. As the developed *I-EMS* is highly depended on the smart communication components and sensors employed in the microgrid, therefore, the microgrid is found vulnerable toward cyber-physical attacks, i.e. adversarial attacks, where data poisoning can significantly damage the performance of the microgrid, its integrity and safety. The main objective of this study is focused on the hybrid microgrid performance under cyber-physical security [14] adversarial-related attacks, such as fast gradient sign, basic iterative, and DeepFool methods, in order to produce perturbation for training data. The performance of the adversarial-based Defence is evaluated using widely used in the bibliography error indexes. Recent studies highlighted the eminent importance of FDI attacks on the vulnerable smart microgrids [15], [16], which can castrate the role of physical units such as power control [17]. To deal with FDI attacks, most of the proposed detection systems envisage the spoofed signals at the actual power electronic and control units such as sensors and PMUs [18] which might exacerbate disruptions in the microgrids before the attacks are detected. Against FDI attacks, advanced detection systems reported in [19] have inspired the authors to develop *SIEMS* for securing predictive I-EMS for hybrid microgrids. Due to the potential for widespread use of DNN in a wide range of applications, ambiguities have been raised about the trustworthiness of this technique. One component that influences the trustworthiness of the system is the use of adversarial samples. In this paper, we study the effect of adversarial examples in DNN. Thus, in this study, the proposed detection system uses two layers of observation against the adversarial attacks, (1) one-day ahead prediction of the voltage and power in the system and (2) online observation at the time of operation in the intelligent electronic devices (IED)s. This way, the first layer continuously learns from the second layer to improve its prediction accuracy alongside security thresholds validations. The proposed *SIEMS* is validated under sophisticated adversarial-related attacks, such as fast gradient sign, basic iterative, and DeepFool methods, to produce perturbation for training data. The *SIEMS*'s adversarial-based Defence strategy relies on two AI-based methods known as defensive distillation and adversarial training strategies. The detection performance of the proposed *SIEMS* is evaluated using widely used in the bibliography error indexes and precision/ recall (or FPR) matrix.

Contribution of the paper. The main contributions of the paper are:

- We provide a full scheme of a *SIEMS* for a secured grid-tied, PV-based microgrid model of IoT devices. Building a supply-based prediction system using long short-term memory (LSTM) on real-time meteorological weather data acquired by a weather station installed in the location of a solar PV plant associated with IoT data traffic. In the first layer (one-day ahead prediction system), the LSTM is chosen because of its unique memory cells that can carry information for a longer time. Compared to the traditional recurrent neural network (RNN) and gated recurrent unit (GRU) architectures, which has been used in [17], the LSTM gates can enter, out or delete information and they do not suffer from vanishing and exploding gradient problems;
- We present the role of BESS and *SIEMS* for a hybrid microgrid under instantaneous boost and reduction of AC loads. The network is limited to 5kW controllable (plug-and-play) AC/DC loads, and critical load of 10kW. In the communication network of the studied hybrid microgrid, the I-DEMS and IEDs are determined as prime targets;
- We define adversarial algorithmic cyber-physical security attacks using fast gradient sign, basic iterative, and DeepFool methods on the *SIEMS*. Their data poising lead to failures of the LSTM's predictions and therefore, supply and demand imbalance. The deficiency of adversarial attacks on the *SIEMS* are presented;
- We develop two AI-based algorithmic Defence strategies, such as defensive distillation and adversarial training methods against the pre-defined random attacks. The performance of the proposed Defence algorithms is evaluated using fitness-based indexes and a precision/ recall (or FPR) matrix. The adversarial attacks and Defence software associated with the paper's achievements is available upon this publication;
- We validate the security features of the proposed security system for the *SIEMS*, including the efficiency of the Defense algorithms designed in this work.

Paper organization. In Section II, we give a survey of existing CPS energy management unit methods. The intelligent architecture devised for the CPS energy management hybrid microgrid is proposed in Section III. In Section IV, we present the attack mechanisms, and in Section V we present our Defence algorithms. The security model imposed for the machine learning (ML) algorithm is analyzed in Section VI. Finally, in Section VII we conclude the paper and present future plans.

II. Related work

The use of grid-connected microgrids is offering advantages such as backup during utility outages which bring resiliency and reliability, voltage sags' reduction, energy-saving via peak shaving, and sustainability [20]. Modern microgrids benefit

from new information and communication technologies (ICT)s for enabling the power system units, such as BEES, to work with the intelligent infrastructures (e.g., I-DEMS) for bi-directional power flows and balancing supply and power demands [21]. Such modern ICT-based networks (i.e., intelligent electronic devices (IED)s, BESSs, phasor measurement unit (PMU), and smart meters) satisfy the need of sharing information among all distributed energy resources (DER)s and loads (or end-users) [22], [23]. However, these DER units are well identified vulnerable to the cyber-attack in much related research papers [22], [24]–[26]. In another study [27], the researchers proposed data manipulation attack detection and mitigation techniques to increase the resilience of distributed control of microgrids concerning FDI attacks, based on Kullback-Liebler (KL) divergence to measure the discrepancy between the Gaussian distributions of the actual and expected local frequency/active power and voltage/reactive power neighborhood tracking errors. They also demonstrated an attack mitigation algorithm based on KL detectors, utilizing self-belief and trust values to modify distributed control protocols. Similarly, in [28], they employed FDI-based attacks and detection in cyber-physical DC microgrids. The detection of FDI in power electronics-intensive DC microgrids has been investigated, which involves spoofing a signal (sensor and/or communication network) via the attack vector to manipulate the microgrid's operation. To detect FDI-based attacks, the researchers have investigated various techniques such as machine- learning-based detectors [29], Kalman filters detectors [30], KL detectors [27], jamming [31], sparse algorithm detector [32], and generalized likelihood detectors [33]. The data corruptions made by FDI-related attacks may also result in denial-of-service (DoS) conditions [34], [35]. The result of any poisoning (or faulty) changes in the base parameters, such as voltage, current, frequency, etc., can create problems in most kinds of prediction algorithms for supply and demand power streaming and balancing in the microgrids, which are today working mostly based on ML methods. Among the AI-based attack defence systems, different methods to detect and defend the power systems against FDI attacks have been proposed in the literature [36], [37]. Defending methods focuses on the techniques, which make the use of mathematical tools like ML algorithms to detect the attacks in the system or make it harder for the attacker. However, there are limitations such as complex formulations, and non-linearity in the model. Following the recent findings, two new AI algorithms are selected to detect and defend the *SIEMS*,(1) defensive distillation and adversarial training strategies (2). Both algorithms are promising in detecting adversarial samples for smart microgrids, where a wide versatility of models is generated for information misclassifications in the system in a well-organized and competent way. They are also able to detect iteratively computed adversarial perturbations by linearizing the attacked model's decision boundaries near the input samples. Interestingly, the misclassified adversarial samples and their predictions have shown an incorrectly still highly confident rate in many cases. In this work, we have reported how our proposed detection system can deal and perform against such intelligent attackers.

III. I-DEMS: Intelligent Dynamic Energy Management Strategy

In this section, we explain our proposed method named Intelligent Dynamic Energy Management Strategy or *I-DEMS*.

A. The Grid-Tied Microgrid Operational Scenario

The grid-tied microgrid with hybrid energy sources is presented in Fig. 1, in which 3.2 kWp solar PV generation and a lead acid battery energy storage system, connected to both critical and non-critical controllable loads. The PV system consists of 18 modules Suntech STP175S-24/Ac. Modules individually wired to reconfigure the photovoltaic field from the laboratory. Two battery packs from Hewlett-Packard Power Trust II A1357A, each with 12 units of 12 V, 8 Ah batteries.

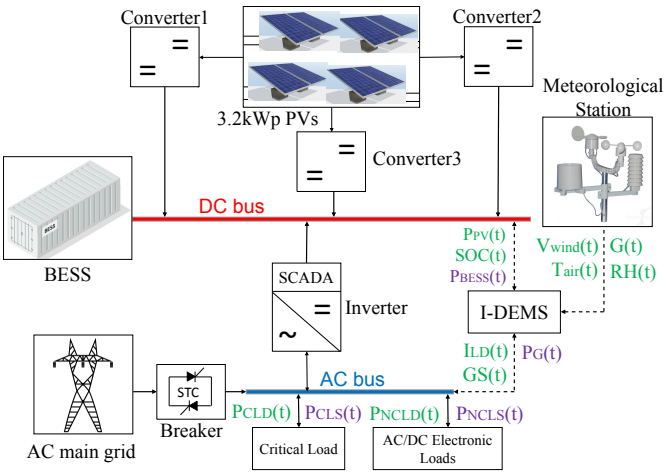


Fig. 1: SIEMS: A secure I-EMS-based architecture of the grid-tied microgrid connected to hybrid resources.

In Fig. 1 presents the I-DEMS-based architecture of the grid-tied microgrid connected to hybrid resources.

supervisory control and data acquisition (SCADA) unit detects faults, their diagnosis and absorption of them through structural redundancies. The server is a Data Logger Meteo-40 of Ammonit, ADC of 12 bit and 22 channels, with configuration to WEB interface and HTTPS connection, ethernet output through RS485, data encryption, and compatibility with SCADA system. The I-DEMS block communicates by receiving and sending states (shown in green) and dispatch (in purple) parameters from three sources, DC bus, AC bus, and the meteorological station, in which the green parameters such as output PV power $PPV(t)$, PBESS(t), and SOC(t) of the BESS (from the DC bus); critical load power demand $PCLD(t)$, non-critical load power demand $PNCLD(t)$, critical load total supplied power $PCLS(t)$, non-critical total supplied power $PNCLS(t)$, inverter demand current to load $ILD(t)$, the main grid power $PG(t)$, and grid statues $GS(t)$ are stated by the AC bus; solar irradiance $G(t)$, wind speed $Vwind(t)$, relative humidity $RH(t)$, and air temperature $Tair(t)$ are the data provided by meteorological station to the I-DEMS. All these twelve parameters are inputs of the DNN implemented in I-DEMS block. The logic operational scenario behind the I-DEMS is summarized as follows:

1) The BESS always dispatches power to the loads, where the $SOC(t)$ remains higher than half of the battery capacity. The PV power as primary source charges the BESS, and the main grid as secondary source ensures the SOC rate to be higher than the defined minimum rate over time. The equations (1) and (2) present how the power delivery process is done whether the BESS power $PBESS(t)$ is higher than $PL(t)$ or not.

$$P_{BESS}(t) = P_{PV}(t) - P_L(t),\ SOC(t) \geq SOC_{min} \quad (1)$$

$$P_{BESS}(t) = P_{PV}(t) + P_G(t) - P_L(t),\ SOC(t) \leq SOC_{min} \quad (2)$$

where the total power demand $PL(t) \leftarrow P_{CLD}(t) + P_{NCLD}(t)$.

2) Always the critical and non-critical total power supplies must satisfy the power demands over time, equations (3) and (4):

$$P_{CLS}(t) = P_{CLD}(t), \quad (3)$$

$$P_{NCLS}(t) = P_{NCLD}(t), \quad (4)$$

Otherwise the $GS(t)$ becomes on.

3) If the $SOC_i(t) \geq SOC_{min}(t)$, the breaker turns GS(t) to off which disconnects use of $PG(t)$, while $P_{PV}(t)$ is powering the BESS until the maximum thresholds rate of $SOC_{max}(t)$ has been satisfied.

4) If the $SOC_i(t) \leq SOC_{min}(t)$, and the following equations. 1 and 2 breach. Then, the $GS(t)$ is on, and thus, $PG(t)$ directly powers the loads by meaning that power dispatch rate of main grid.

5) The environmental parameters $G(t)$, $V_{wind}(t)$, $RH(t)$, and $T_{air}(t)$ are measured and stored in cloud at all time. Using predictive modelling updates $SOC(t)$ for the one-hour-ahead estimation of $SOC(t)$ and its related $P_{PV}(t)$.

6) Maximizing controllable load dispatch offers demand response capability by means of faster lead-in power streaming. In addition, the power dispatched to discharge the BESS should not exceed the total load demand (including critical and non-critical loads).

7) Harvesting maximum power available from the PV system and minimizing the use of main grid.

A dynamic programming derives the I-DEMS to minimize the cost function $J(t)$, its network and weight computation is presented in Fig. 1. As presented, there are fourteen inputs, ten green states and four purple as dispatch parameters. After a multiple three hidden layers, these dispatch parameters are updated to equilibrate other controllable parameters. For updating the weights, a standard back-propagation DDNN computes the Hamilton–Jacobi–Bellman equation of optimal control, as follow:

$$J(S) = \sum_{t=0}^{\infty}\Big(w_1(t) \times f\left(P_{CLS}(t)\right) + w_2(t) \times f\left(P_{NCLS}(t)\right)\Big) + \sum_{t=0}^{\infty}\Big(w_3(t) \times f\left(SOC(t)\right) + w_4(t) \times f\left(P_G(t)\right)\Big) + \sum_{t=0}^{\infty}\Big(w_5(t) \times f\left(P_{BESS}(t)\right) + w_6(t) \times f\left(N_{BESS}(t)\right)\Big) + \sigma \cdot J(t) - J(t-1) \quad (5)$$

where $NBESS(t)$ is the total charge and discharge continuous number of states in the BESS. The multiple-layer DNN has two sets of controllable parameters, the nonlinear and linear parameters in the basis of equation. (5) and its weights, where the supervised learning offers a continues optimization solver to balance all the dispatch parameters for the $J(t)$ minimization.

B. Intelligent Agent Architecture

In this paper, we present the following architecture for intelligent data classification, in which a classifier is used to classify the collected data.

There are three general parts to this proposed architecture. After collecting the data and performing prepossessing operations, an LSTM-based classification algorithm is used to predict the values in this work. Compared to the traditional RNN and gated GRU architectures, which has been used in [17], the LSTM gates can enter, out or delete information and they do not suffer from vanishing and exploding gradient problems. GRUs bags have two gates (reset, update) while LSTMs have three (input, output, forget). GRU has less training parameters than LSTM and thus is preferred in small datasets. However, microgrids work with big datasets where LSTM is usually performing better. Also, LSTMs are better than GRUs in modelling and remembering long duration patterns, but GRUs are trained faster. RNNs are trained faster than GRUs and LSTM but cannot model long-term dependencies. This part of the architecture, marked in black and numbered 1 in Fig. 2, belongs to the time when the goal was to predict values for unseen samples.

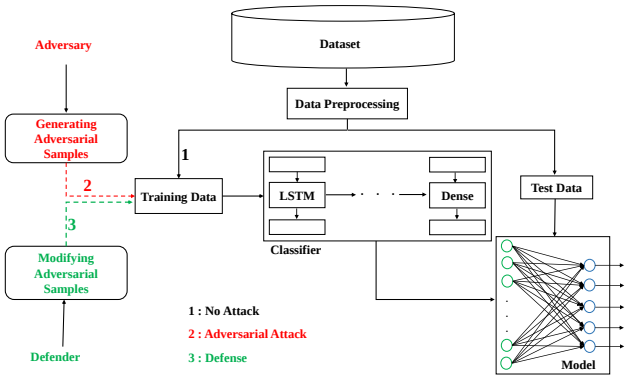


Fig. 2: The proposed LSTM-based architecture

The second part of the architecture is when the adversary tries to add perturbation to the training data by accessing the training data and therefore reducing the accuracy of the prediction algorithms. In this architecture, the production of adversarial examples is shown in red and numbered with 2. It is clear to expect that the accuracy of prediction operations will be reduced by using adversarial attacks.

The third part of the architecture, highlighted in green and numbered with the number 3, shows a Defence algorithm against adversarial attacks. In this part of the architecture, methods are proposed to modify the data attacked by adversarial methods. These methods try to replace perturbed samples with more appropriate values.

TABLE I: Settings and hyperparameters for both classifiers

Symbol	Value/Description
Learning Rate	0.005
Activation Function(non˙last layers)	Relu
Activation Function(last layer)	Softmax
Dropout Rate	0.2
Loss Function	Mean Square Error

1) LSTM3-Dense Classifier : In the first classifier, three LSTM structures are placed in sequential, with dropout layers used between them. The input to the first LSTM layer is from the data set, and the output of the last LSTM is given to a Dense layer responsible for predicting values.

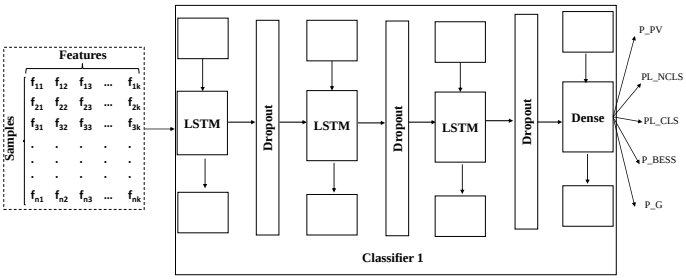


Fig. 3: LSTM3-Dense classifier architecture used to generate model

2) LSTM-Dense3 Classifier : In the second classifier, one LSTM layer and three sequential Dense layers are used. Similar to the first classifier, dropout layers are placed between the layers. The input from the data set is used as input to the LSTM layer, and the LSTM output enters the next Dense layers.

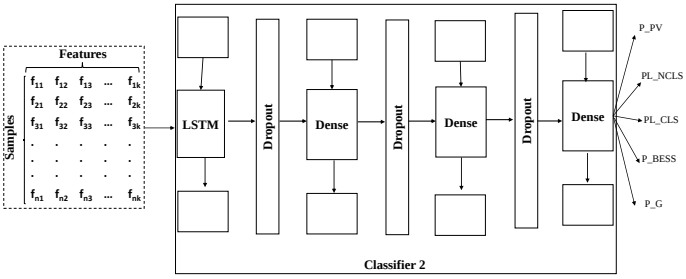


Fig. 4: LSTM-Dense3 classifier architecture used to generate model

Table I demonstrates some of the settings and hyperparameters used for both classifiers.The pseudocode of both classifiers are shown Alg. 1.

IV. Attack Algorithms

This section introduces *three* algorithms for generating adversarial samples, which are used in this work to add perturbation to training data before developing an ML model. In the attack methods, perturbations are generated based on changes in the data. In the detection system, both defense algorithms utilized adversarial training, the model is trained based on the changed data.

Algorithm 1 Classifiers: LSTM3-Dense , LSTM-Dense3

Input: Input_shape Classifier_ID
Output: $Model$

1: **if** (Classifier_ID==1) **then**
2: Model=Sequential(LSTM((Input_shape) + Dropout) + (LSTM + Dropout) + (LSTM + Dropout) +Dense)
3: **end if**
4: **if** (Classifier_ID==2) **then**
5: Model=Sequential(LSTM((Input_shape) + Dropout) + (Dense + Dropout) + (Dense + Dropout) +Dense)
6: **end if**
7: **return** $Model$

A. Attack 1:Fast Gradient Sign Method

The Fast Gradient Sign Method (FGSM) is a straightforward, effective method for generating adversarial samples. Goodfellow et al. [38] proposed FGSM, which works as follows:

In this method, a trained classifier is utilized to generate a prediction on each input sample. The prediction loss is then computed using the true class label. The gradients of the loss concerning the input sample are calculated next. Finally, the gradient's sign is derived, and the signed gradient is used to generate the output adversarial sample. Using the gradients of the neural network, the fast gradient sign method creates an adversarial example. Based on the gradients of the loss for the input sample for an input sample, the approach generates a new sample that maximizes the loss. This new sample has been given the term adversarial sample. The FGSM idea can be represented by equation (6):

$$adv_x = x + \epsilon * sign(\nabla_x J(\theta, x, y)) \tag{6}$$

In equation (6), x is the original input sample correctly classified , y is the ground truth label for x, θ represents the model parameters, and $J(\theta, x, y)$ is the loss that is used to train the network. The attack backpropagates the gradient back to the input data to calculate $\nabla_x J(\theta, x, y)$. Then, it adjusts the input data by a small step in the direction (i.e. $\text{sign}\left(\nabla_x J(\theta, x, y)\right)$) that will maximize the loss. The resulting perturbed sample, adv_x, is then misclassified by the target network as a different sample when it is still clearly same as first sample.

The fact that the gradients are taken with the input sample is an intriguing feature. The goal is to create a sample that minimizes loss. It is also one method for determining how each feature in the sample contributes to the loss value, thus adding a perturbation. This works very fast because it is simple to apply the chain rule and obtain the gradients required to determine how each input feature contributes to the loss. As a result, the gradients about the sample are computed. Furthermore, because the model is no longer trained, the model parameters remain consistent (and the gradient is no longer calculated compared to the trainable variables, i.e., the model parameters). The only goal is to deceive an already trained model.

Time Complexity of FGSM: To compute the time complexity of this method, we must note that equation (6) is repeated for all samples. If we consider the number of samples as n, the time complexity of this method will be $O(n)$. In

equation (6), there is a derivative part that must be considered. Therefore, we can say that the time complexity related to FGSM is: $\mathcal{O}(n) \times O(\text{Derivation part})$

B. Attack 2: Basic Iterative Method(BIM)

We use a simple method to extend the "fast" approach. We repeat the process with small step sizes several times, clipping feature values of intermediate results after each step to ensure that they are within a reasonable distance of the original sample. [39]:

The BIM from adversarial examples in the Physical World is a simple extension of the FGSM, where instead of taking one large step, it takes an iterative approach by applying FGSM multiple times to a sample with step size α, the change in sample value per iteration. The resulting adversary can then be clipped to limit the maximum perturbance for each sample.

All iterative methods like the BIM are slower, but generally produce more successful and subtle perturbation to samples. First, a clean sample X is used for initialization in iteration $n = 0$. Then, using this sample a step similar from the FGSM is performed as equations (7) and (8):

$$x_0^{adv} = x \tag{7}$$

$$x_{N+1}^{adv} = Clips_{x,\epsilon}\left(x_N^{adv} + \alpha sign(\nabla_x J(x_N^{adv}, y_{true}))\right) \tag{8}$$

The adversarial example is then clipped to ensure that all sample values are within the bounds of epsilon and the maximum and minimum sample intensities.

Repeat these steps for N iterations to get the final adversary. α is chosen to be one sample intensity value and the number of iterations is calculated to ensure enough steps to allow a sample to reach the maximum adversarial perturbance, ϵ.

Time Complexity of BIM: Similar to the FGSM method the time complexity of the BIM method, equation (7) is repeated for all samples. If we consider n samples, the time complexity of this method will be $O(n)$. Also, in equation (8), there is a derivative section and a Clips section that must be considered. So we can say that time complexity related to FGSM is $\mathcal{O}(n)\times$ $O(\text{Derivation part}) \times O(\text{Clips})$.

C. Attack 3: DeepFool

The main contributions of the DeepFool [40] method are listed below. First, a simple and accurate method is presented to calculate the strength of different classifiers for adverse perturbations. Then DeepFool calculates an adversarial disruption more effectively. Adversary training ultimately improves the robustness of the system. Suppose the input is X, each class is divided into a hyperplane. X is divided into a class based on its location in space. This algorithm locates and projects the next hyperplane and drives it a bit further, misclassifying it as little disturbance as possible.The formulation is illustrated in this closed-form equation (9):

$$\begin{aligned} \boldsymbol{r}_*(\boldsymbol{x}_0) &:= \arg\min \|\boldsymbol{r}\|_2 \\ &\text{subject to } \operatorname{sign}\left(f(\boldsymbol{x}_0 + \boldsymbol{r})\right) \neq \operatorname{sign}\left(f(\boldsymbol{x}_0)\right) \\ &= -\frac{f(\boldsymbol{x}_0)}{\|\boldsymbol{w}\|_2^2}\boldsymbol{w} \end{aligned} \tag{9}$$

Assuming now that f is a general differentiable classifier, we adopt an iterative procedure to estimate the robustness $\Delta(\boldsymbol{x}_0; f)$. Specifically, at each iteration, f is linearized around the current point $\boldsymbol{x}_i$ and the minimal perturbation of the linearized classifier is computed as equation (10):

$$\underset{\boldsymbol{r}_i}{arg\,min}\, \|\boldsymbol{r}_i\|_2 \text{ subject to } f(\boldsymbol{x}_i) + \nabla f(\boldsymbol{x}_i)^T \boldsymbol{r}_i = 0 \tag{10}$$

The perturbation $\boldsymbol{r}_i$ at iteration i of the algorithm is computed using this closed-form solution, and the next iterate $\boldsymbol{x}_{i+1}$ is updated.

Time Complexity of DeepFool: Regarding the time complexity of this method, we must consider equation (10), where the derivation and minimization are calculated. Given that the number of times to check the condition in minimization is equal to the number of samples,n, so time complexity is $\mathcal{O}(n) \times O(\text{Derivation part})$.

V. Defense Algorithms

This section proposes our *two* proposed defense algorithms, as a part of the proposed detection system, to mitigate attacks provided in the previous section.

A. Defence 1: Adversarial Training

Adversarial training [38] is a simple Defence against adversarial samples that attempts to improve the robustness of a neural network by training it with adversarial samples. The concept of adversarial training was initially introduced, in which neural networks are trained on a combination of adversarial and clean samples. Following that, other papers advocated using FGSM to generate adversarial samples during training. Their trained models, however, are still sensitive to repeated assaults since they used a linear function to approximate the loss function, which resulted in high curvature around data points on the decision surface of the associated deep models. The formulation is illustrated as equation (11):

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{\delta\in B(x,\varepsilon)} \mathcal{L}_{ce}(\theta, x+\delta, y)\right] \tag{11}$$

$(x, y) \sim \mathcal{D}$ represents training data sampled from distribution $\mathcal{D}$ and $B(x, \varepsilon)$ is the allowed perturbation set, expressed as $B(x, \varepsilon) := \left\{x + \delta \in \mathbb{R}^m \mid \|\delta\|_p \leq \varepsilon\right\}$.

Time Complexity of Adversarial Training: In this defense algorithm, in each step an operational set is made to select the best samples to train the model, and because n samples are examined, the time complexity is $\mathcal{O}(n)$.

B. Defence 2: Defensive Distillation

Defensive distillation [41] is an adversarial training approach that increases the flexibility of an algorithm's categorization process, making it less subject to abuse. To emphasize accuracy, one model is trained to predict the output probabilities of another model that was previously trained on a baseline standard during distillation training. It is trained in three steps: 1. Train a network (the teacher) using standard techniques. In this network, the output is given by equation (12):

$$F(\theta, x) = \operatorname{softmax}(Z(\theta, x)/T) \tag{12}$$

Algorithm 2 Scenarios 1-3: Prediction, Adversarial Attacks, Adversarial defenses.

Input: x, ClassifierID

Output: $accuracy$

`Scenario 1: Prediction of parameters`
Model ← Algorithm 1(x , ClassifierID)
1:
Accuracynoattack ← Use Model to Predict results before attack
`Scenario 2(Attacker side): Select one of adversarial attacks`
FGSM Attack: apply equation (3)
BIM Attack: apply equations (4) and (5)
DeepFool Attack: apply equation (6)
Model ← Algorithm 1(Perturbed x , ClassifierID)
2:
Accuracywithattack ← Use Model to Predict results after attack
`Scenario 3(Defender side): Select adversarial Defence`
Adversarial Training Defence: apply equation (7)
Defensive Distillation Defence: apply equation (8) and related Steps
Model ← Algorithm 1(Corrected x , ClassifierID)
3:
Accuracyafterdefense ← Use Model to Predict results after Defence
4:
5: **return** Accuracynoattack, Accuracywithattack, Accuracyafterdefense

for some temperature T. As $T \to \infty$ the distribution approaches uniform; as $T \to 0^+$the distribution approaches the hard maximum; standard softmax uses $T = 1$
2. Evaluate the teacher network on each instance of the training set to produce soft labels. These soft labels contain additional information. 3. Train a second network (the distilled network) on the soft labels again using temperature T. By training on the soft labels, the model should overfit the data less and try to be more regular.

Finally, to classify an input, run the distilled network using temperature $T = 1$. By training at temperature T the logits (the inputs to the softmax) become on average T times larger in absolute value to minimize the crossentropy loss. This causes the network to become significantly more confident in its predictions.

Time Complexity of Defensive Distillation: In this defense algorithm, similar to the adversarial training method, in each step an operational set is made to select the best samples to train the model, and because n samples are examined, the time complexity is also $\mathcal{O}(n)$. However, because each part involves several different operations, which are described by numbering above, the execution time is somewhat longer than standard adversarial training. The pseudocode of the proposed methods are shown in Fig. 2.

VI. Performance Evaluation

This section is devoted to comparing the proposed algorithm in various datasets. In this way, we shape the communication SG model that is presented in Fig. 5. Fig. 5 illustrates a simplified smart hybrid microgrid cyber-physical layout with both wired, in red, and wireless connectivity (shown in dashed blue). The studied hybrid microgrid has different domains and subdomains, where multiple smart components are employed, such as IEDs and smart power measurement units (PMU)s. These components transfer data via wireless communications, which is demonstrated by dashed blue, as presented Fig. 5. The data exchanges between themselves and all will be stored in the data acquisition unit of the PC for prediction and decision-making purposes. Both intelligent IED and PMU components are, the highlighted parts, vulnerable in cyber-physical security systems. These areas are targetable for cyber-physical attackers; this work has mainly focused on targets 1 and 2, in which target one manipulates the weather-related data and/ or the PC's data. While target 2 can be sub versioned by adding/ or removing the number of local loads and/ or the data exchange through feeders (or IEDs). The Python implementation of SIEMS is available in [42].

A. Simulation Setup

Our experiments have been performed on a Win10 64-bit OS server with Python 3.6.4, an eight-core Intel Core i7 4 GHz,

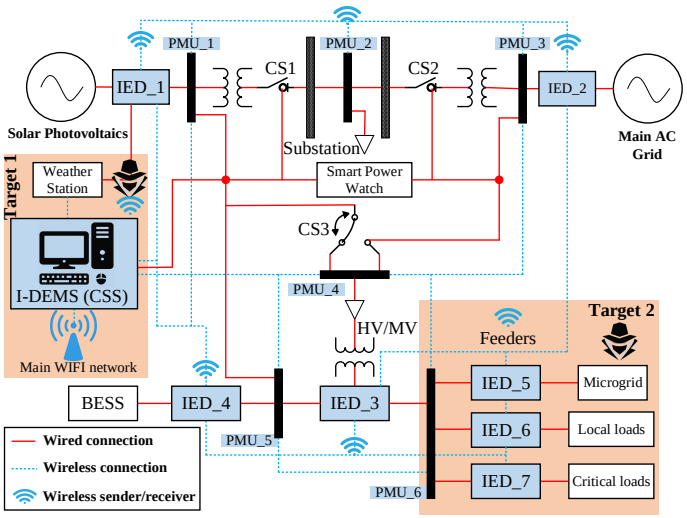


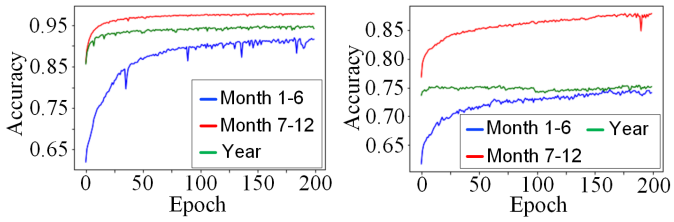
Fig. 5: Communications architecture of the studied grid-tied microgrid with wire (in red) and wireless (dashed blue) connections.

RAM 16 GB. We use Tensorflow 2 and Keras 2 to adopt our method.

B. Results

In this part we explain the results. Fig. 6 illustrates the accuracy of training models 1 and 2 on six-months and one-year data. This figure shows the better success of model 1 because the data volume is higher in six-months and one-year intervals, and it is expected that the model will be trained with higher accuracy, which according to the figure shows that model 1 has higher accuracy in the training phase.

The next part of the results is related to creating a model based on two classifiers designed to predict the parameters. In these figures, the training steps related to creating models from classifiers are shown. Fig. 7 shows the accuracy of the training step for generating model 1 from classifier 1 in different months of the year for 200 epochs. As can be seen, the accuracy of making model 1 is almost 96% in all months of the year after reaching a steady-state, which indicates that classifier 1 has high quality. The same results are shown in



(a) Model 1 (b) Model 2

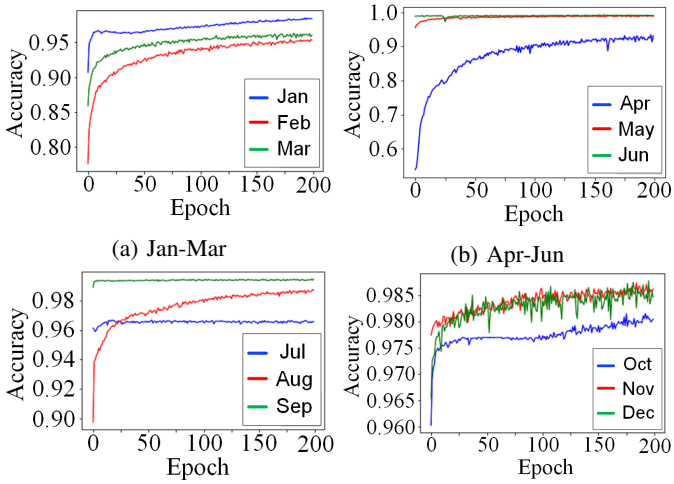
Fig. 6: Accuracy for both models in training (No attack-Compare for 6 months).

Fig. 8 for training Model 2 from Classifier 2, which shows that the accuracy of the Model from this method is lower, and in April, February and March, it could not reach an accuracy of more than 95%.

In the proposed model, the settings presented in Figs. 3 and 4 are used. In Fig. 3, we set three LSTM layers and one dense layer, and in Fig. 4, we consider one LSTM layer with three dense layers. In Table II, we present the accuracy of the model for some of the other settings. This table indicates that LSTM-Dense3 and LSTM3-Dense are reasonable options.

TABLE II: Accuracy results in different settings

	Period	LSTM3-Dense	LSTM2-Dense	LSTM-Dense	LSTM-Dense2	LSTM-Dense3
Months	January	99.14	97.98	95.39	96.93	96.68
	February	96.98	93.28	87.12	75.92	76.80
	March	98.20	92.17	80.57	85.11	84.53
	April	96.39	85.54	80.11	79.45	79.25
	May	99.47	94.29	96.41	98.11	97.22
	June	99.62	96.54	94.18	95.67	96.48
	July	96.29	93.49	93.87	95.12	96.29
	August	99.25	98.61	96.57	96.34	97.03
	September	99.25	96.29	97.01	98.32	99.08
	October	98.40	97.92	98.14	97.84	97.58
	November	98.76	99.21	99.02	97.98	98.06
	December	99.33	97.71	94.63	96.39	96.10

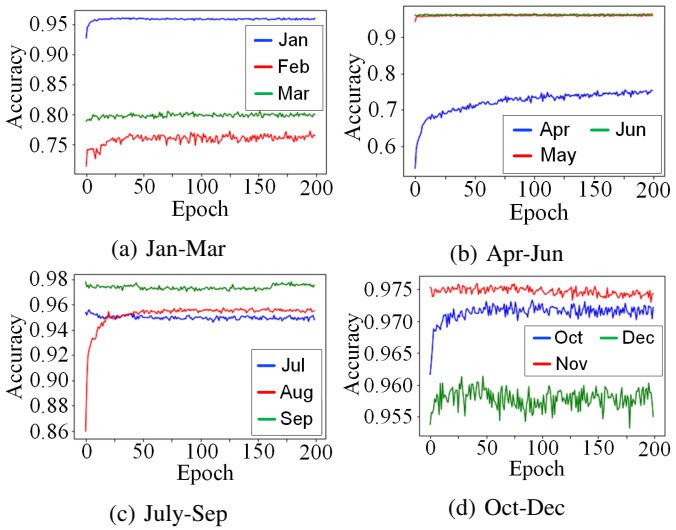


(a) Jan-Mar (b) Apr-Jun

(c) July-Sep (d) Oct-Dec

Fig. 7: Accuracy for model 1 in training for one year.

As can be seen in Figs. 7 and 8, in some months of the year, the accuracy of modeling based on the data used is significantly different from other months and has decreased. This can be for various reasons and is somewhat common because the user data are collected from real-world environments, and noise and climate change are possible, which can affect data.



(a) Jan-Mar (b) Apr-Jun

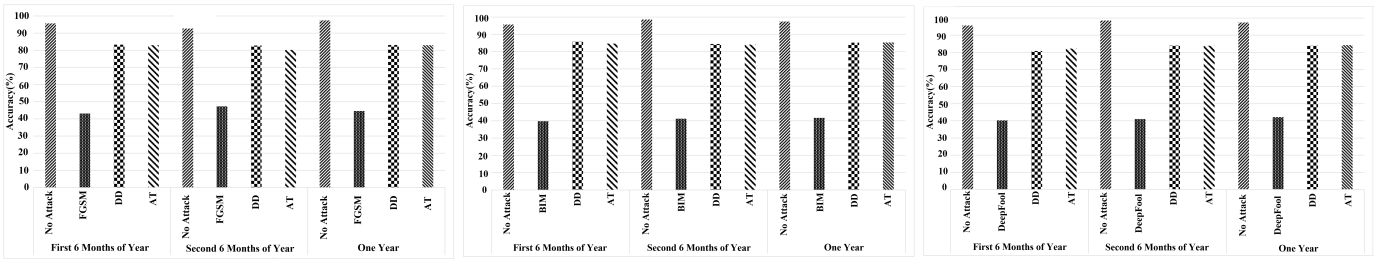
(c) July-Sep (d) Oct-Dec

Fig. 8: Accuracy for model 2 in training for one year.

1) Results related to 6 months and year: Fig. 9 presents the accuracy of the first proposed model before and after the use of three types of attack and two types of Defence used at six-month intervals. It is clear that in each of the six-month intervals before the attack, the ML model predicted the values with high accuracy, while after the attack, the accuracy was severely reduced. In the first six months, BIM was more successful, reducing accuracy by about 40%. In this scenario, it can be seen that BIM has been more successful in the first six months than the second six months and the whole year and has drastically reduced the accuracy. Among the Defence methods used, it can be seen that Defence number 1 has been able to have more robustness against all three types of attacks. After applying this Defence method against all three attacks, the prediction accuracy in model number 1 has increased. With this explanation, it can be said that in model number 1, from the attacker's point of view BIM, and the Defence system's point of view, defensive distillation had the best performance.

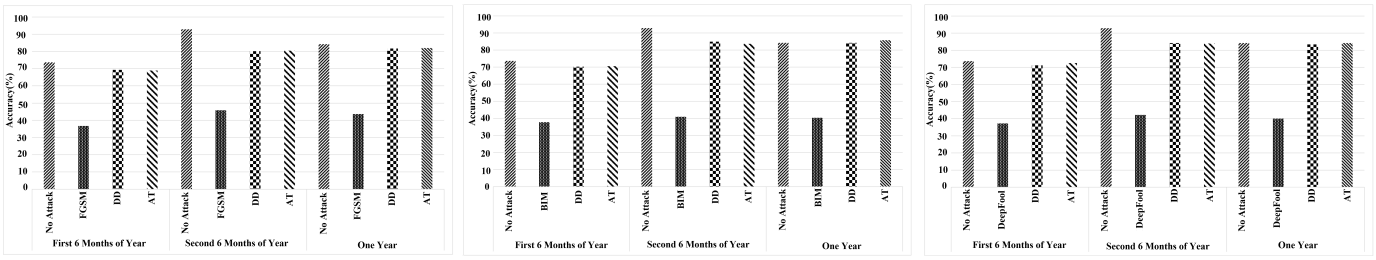
Similarly, Fig. 10 compares the accuracy of the second proposed model before and after the use of three types of attack and two types of Defence used at six-month intervals. As can be seen, in each of the six-month intervals before the attack, the ML model was able to predict the values with high accuracy, while after the attack, the accuracy was severely reduced. In the first six months, BIM was more successful, reducing accuracy by about 40%. In this scenario, it can be seen that BIM has been more successful in the first six months than the second six months and the whole year and has drastically reduced the accuracy. Among the Defence methods used, it can be seen that Defence number 1 has been able to have more robustness against all three types of attacks. After applying this Defence method against all three attacks, the prediction accuracy in model number 1 has increased. With this explanation, it can be said that in model number 1, from the attacker's point of view, BIM, and the Defence system's point of view, defensive distillation had the best performance.

Table III shows the results of using the Defensive Distillation method against all attacks1, BIM, and DeepFool. In scenario 0 in this table, the prediction results of the five-



(a) FGSM Attack (b) BIM Attack (c) DeepFool Attack

Fig. 9: Accuracy results for model 1 for the scale of 6 months (comparing no attack state, applying attacks and defenses against each attack)



(a) FGSM Attack (b) BIM Attack (c) DeepFool Attack

Fig. 10: Accuracy results for model 2 for the scale of 6 months(comparing no attack state, applying attacks and defenses against each attack)

goal variables are shown by Model 1 without an attacker's presence. As can be seen, Model 1 has been very successful, with a prediction accuracy of over 99% for most months of the year and a minimum accuracy of 96.29 in July. In scenario 1, the results of using FGSM and Defensive Distillation against this attack are shown. It can be seen that after FGSM, the accuracy of prediction has decreased drastically, reaching less than 38% accuracy in all months of the year. If we want to look at it from the attacker's point of view, the most successful attack has been made on the February data, in which the forecast accuracy is only 24.87%, and in fact, the accuracy has decreased by 72.11%. Table IV shows the accuracy results for using the proposed methods based on model 1. In this table, scenario 0 indicates the state in which an attack has not yet taken place. In this case, the prediction accuracy of the five values P_PV and PL_NCLS, PL_CLS, P_BESS, and P_G are shown. It can be seen that model 1 has been able to predict these five values with great accuracy for different months of the year. The other three scenarios are the combination of Defensive Distillation against FGSM, BIM, and DeepFool. As can be seen, after using each of the attacks, the accuracy of the prediction method is greatly reduced. Of course, the decrease in accuracy varies for different months of the year. Each of the three scenarios 1 to 3 is used against the defensive distillation method. By using the Defence method, the accuracy of the prediction process increases to an acceptable level.

Similarly, in Scenario 2, the results of BIM and Defensive Distillation against this attack are shown on Tables VIand V. It can be seen that after BIM, the accuracy of prediction has decreased drastically, reaching less than 40% accuracy in all months of the year. From the attacker's point of view, the most successful attack has been made on the March data, in which the forecast accuracy is only 29.53%, and in fact, the accuracy has been reduced by 68.67%.

TABLE III: Accuracy results for model 1, scenarios 1-3 (Using Defensive Distillation as defence method against all attacks)

		Scenario 0	Scenario 1		Scenario 2		Scenario 3	
	Period	No Attack	FGSM	DD	BIM	DD	DeepFool	DD
Model1	January	99.14	32.62	80.65	31.89	81.52	32.26	82.13
	February	96.98	24.87	67.6	29.23	77.36	25.39	78.98
	March	98.20	29.71	75.88	29.53	79.43	31.24	80.35
	April	96.39	28.05	74.11	29.45	71.59	27.96	75.54
	May	99.47	34.92	85.5	32.76	83.48	35.49	83.74
	June	99.62	36.45	88.81	35.18	88.95	36.14	89.06
	July	96.29	33.94	86.13	32.78	81.58	35.11	81.63
	August	99.25	32.23	79.75	39.17	80.07	34.39	80.79
	September	99.25	33.83	80.9	34.71	79.37	32.97	79.69
	October	98.40	37.29	81.03	37.61	82.7	34.59	82.66
	November	98.76	29.91	82.75	32.58	83.24	31.53	83.64
	December	99.33	30.47	81.56	31.91	83.48	30.29	83.01

TABLE IV: Accuracy results for model 1, scenarios 4-6 (using Adversarial Training as defense method against all attacks)

		Scenario 0	Scenario 4		Scenario 5		Scenario 6	
	Period	No Attack	FGSM	AT	BIM	AT	DeepFool	AT
Model1	January	99.14	32.62	83.11	31.89	81.25	32.26	81.96
	February	96.98	24.87	71.9	29.23	78.93	25.39	75.69
	March	98.20	29.71	75.31	29.53	78.68	31.24	81.32
	April	96.39	28.05	7371	29.45	78.82	27.96	76.26
	May	99.47	34.92	80.39	32.76	84.21	35.49	83.42
	June	99.62	36.45	89.45	35.18	87.59	36.14	88.61
	July	96.29	33.94	87.32	32.78	86.39	35.11	83.35
	August	99.25	32.23	80.32	39.17	81.19	84.39	79.86
	September	99.25	33.83	81.87	34.71	82.24	32.97	81.58
	October	98.40	37.29	80.96	37.61	81.39	34.59	80.91
	November	98.76	29.91	81.93	32.58	82.11	31.53	83.43
	December	99.33	30.47	81.09	31.91	83.26	30.29	82.74

One of the important metrics in comparing adversarial methods is FPR, which increases its value due to adversarial attacks and decreases after using defence methods [43]. In Table VII, the FPR values for the proposed attack and defence methods are presented, which confirms the results of the comparison based on accuracy. The results shown in Table VII for each scenario are based on when the attack took place, and this table does not report the results after applying the defense to keep the paper short. Given the above results and

TABLE V: Accuracy results for model 2, scenarios 1-3 (using Defensive Distillation as defense method against all attacks)

		Scenario 0	Scenario 1		Scenario 2		Scenario 3	
	Period	No Attack	FGSM	DD	BIM	DD	DeepFool	DD
Model2	January	96.68	32.47	82.8	35.11	81.03	34.16	81.07
	February	76.80	23.85	61.45	20.39	60.56	23.4	61.31
	March	84.53	29.45	68.92	26.74	64.95	21.94	65.08
	April	79.25	16.37	44.03	22.91	56.16	23.79	68.88
	May	97.22	39.34	84.3	37.29	85.95	40.02	81.11
	June	96.48	36.35	86.02	33.06	81.26	34.11	81.31
	July	96.29	35.53	85.36	31.71	80.09	29.92	80.01
	August	97.03	31.49	75.99	24.16	80.49	26.12	80.48
	September	99.08	35.61	87.77	31.49	82.34	32.16	82.35
	October	97.58	36.4	86.17	30.91	82.18	29.98	82.24
	November	98.06	32.55	87.53	33.12	82.43	32.01	82.54
	December	96.10	34.53	85.38	32.21	80.93	32.43	80.73

TABLE VI: Accuracy results for model 2, scenarios 4-6 (using Adversarial Training as defense method against all attacks)

		Scenario 0	Scenario 4		Scenario 5		Scenario 6	
	Period	No Attack	FGSM	AT	BIM	AT	DeepFool	AT
Model2	January	96.68	32.47	81.76	35.11	81.39	34.16	80.79
	February	76.80	23.85	63.01	20.39	62.21	23.4	63.24
	March	89.53	29.45	67.94	26.74	65.92	21.94	66.37
	April	79.25	16.37	43.29	22.91	54.36	23.79	70.06
	May	97.22	39.34	82.16	37.29	84.29	40.02	83.24
	June	96.48	36.35	85.11	33.06	79.98	34.11	82.91
	July	96.29	35.53	84.92	31.71	80.74	29.92	82.54
	August	97.03	31.49	77.35	24.16	81.35	26.12	81.19
	September	99.08	35.61	85.64	31.49	79.68	32.16	82.64
	October	97.58	36.4	86.23	30.91	80.93	29.98	82.41
	November	98.06	32.55	86.71	33.12	81.31	32.01	83.49
	December	96.10	34.53	84.93	32.21	82.57	32.43	81.02

TABLE VII: FPR, Precision and Recall values for different attacks; F:= FPR; P:= Precision; R:= Recall

		No Attack			FGSM			BIM			DeepFool		
	Month	F	P	R	F	P	R	F	P	R	F	P	R
Model 1	January	0.49	94.67	95.95	64.20	3.57	14.17	65.13	3.32	13.71	65.00	3.7	15.32
	February	2.44	75.00	89.55	70.64	2.54	7.26	66.98	2.35	8.63	69.92	1.76	5.42
	March	1.35	86.25	93.24	68.17	6.73	20.69	66.71	2.35	8.82	66.76	7.98	23.20
	April	2.56	72.00	83.08	68.85	4.91	14.94	65.73	0.60	2.24	68.34	2.51	8.97
	May	0.25	97.26	95.25	61.99	4.29	16.54	64.55	4.31	17.19	62.68	6.35	24.81
	June	0.25	97.22	98.59	60.71	4.74	19.01	61.80	3.46	14.78	61.10	3.25	15.53
	July	2.69	72.15	85.07	63.78	6.35	22.38	65.14	6.04	21.68	63.21	6.30	25.4
	August	0.37	95.65	94.29	64.72	3.75	14.62	57.88	4.27	18.18	65.59	8.82	34.06
	September	0.49	94.20	95.59	63.73	6.18	21.53	63.33	6.96	24.31	63.78	4.65	16.08
	October	0.62	93.15	88.31	60.10	5.18	20.83	60.38	6.28	25.00	63.47	7.39	25.00
	November	0.62	93.33	92.11	67.08	5.96	17.51	63.92	4.25	14.69	66.57	8.37	24.18
	December	0.25	97.22	94.59	66.71	6.35	18.75	66.03	7.84	23.30	67.30	6.21	19.41
Model 2	January	2.34	77.11	86.49	65.79	5.32	22.05	62.93	4.49	21.50	62.76	3.83	15.32
	February	22.87	15.86	72.00	71.89	2.49	7.14	75.79	2.23	6.32	72.51	2.68	7.69
	March	15.46	32.26	84.51	69.47	6.22	23.97	60.67	6.61	23.02	73.09	2.31	5.69
	April	20.60	19.34	77.36	79.51	1.08	2.82	72.96	2.67	7.57	72.41	2.45	7.65
	May	2.34	78.89	92.21	60.00	12.45	36.53	59.17	5.09	17.83	59.34	8.89	36.07
	June	2.68	72.15	86.36	62.98	8.33	32.82	62.97	3.09	11.19	62.92	2.82	13.21
	July	3.41	68.54	92.42	61.19	5.85	19.18	65.58	5.30	18.24	66.71	5.40	15.88
	August	2.85	76.53	96.15	65.32	3.95	14.29	71.92	3.87	10.26	73.92	6.34	26.62
	September	0.62	93.75	96.15	61.03	5.45	18.06	64.88	7.45	18.56	64.17	4.46	14.38
	October	0.99	89.19	83.54	60.76	4.54	18.33	67.57	5.32	22.40	68.17	7.10	21.89
	November	1.24	88.24	91.46	64.17	7.41	20.00	63.49	4.25	15.22	65.53	7.35	21.71
	December	2.60	75.00	81.82	61.45	6.67	18.08	65.80	7.82	23.84	65.13	4.33	17.69

according to the findings of this study, adversarial attacks increase adversarial samples in the system, and as discussed in [44] and [45], destroy trustworthiness in the system. It is anticipated that the application of the provided defensive mechanisms would boost trustworthiness.

VII. Conclusion and Future Directions

This paper designed a hybrid, grid-tied microgrid to introduce the role of active ICT-based components, such as IEM, PMU, PC, for information communication purposes. In this autonomous microgrid, the developed *SIEMS* receives the weather information and uses LSTM for one-day-ahead predictions. The problem with such *SIEMS* is its data streaming resources and nodes, which cyber-physical security threats like adversarial attacks can poison. We study the impact of several adversarial-based attacks, which have raised the vulnerability of the whole system's performance and safety. The paper also tests the targeted *SIEMS* under FGSM, BIM, and DeepFool attacks, where the microgrid's integrity was downgraded by these methods to about 30% in best cases from attacker view. The development of Defence algorithms, is the main contribution of this work, is successfully done using defensive distillation and Adversarial training algorithms. Considering all scenarios used in this paper, it shows that in almost all cases, the prediction accuracy of the proposed models is above 95% and often above 98%. Although after applying the proposed attacks on training data, the prediction accuracy of these models are reduced to about 30% to 40%, but Defence methods play a significant role in the robustness of the system and by applying them to the case data that attacks are taking place, the prediction accuracy of the models increases to over 80%. In fact, it can be said that by using the proposed models and Defence methods, the accuracy of prediction does not decrease to less than 80% even in the presence of an attacker. In future, we plan to extend the *SIEMS* and generate a real-time cyber-physical security software model by increasing the development of distributed systems. In this way, we will apply distributed ML methods such as federated learning to make a robust and reliable prediction model. We also plan to consider using generative methods such as GAN and Autoencoder to generate adversarial samples and the same methods to defend against attacks.

References

[1] J. Duan, Z. Yi, C. Lin, X. Lu, and Z. Wang, "Reinforcement-learning-based optimal control of hybrid energy storage systems in hybrid ac–dc microgrids," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5355–5364, 2019.

[2] M. Najafzadeh, R. Ahmadiahangar, O. Husev, I. Roasto, T. Jalakas, and A. Blinov, "Recent contributions, future prospects and limitations of interlinking converter control in hybrid ac/dc microgrids," *IEEE Access*, vol. 9, pp. 7960–7984, 2021.

[3] J. Arkhangelski, M. Abdou-Tankari, and G. Lefebvre, "Day-ahead optimal power flow for efficient energy management of urban microgrid," *IEEE Transactions on Industry Applications*, vol. 57, no. 2, pp. 1285–1293, 2021.

[4] Q. Sui, F. Wei, C. Wu, X. Lin, and Z. Li, "Day-ahead energy management for pelagic island microgrid groups considering non-integer-hour energy transmission," *IEEE Transactions on Smart Grid*, vol. 11, no. 6, pp. 5249–5259, 2020.

[5] R. Carli and M. Dotoli, "Decentralized control for residential energy management of a smart users' microgrid with renewable energy exchange," *IEEE/CAA Journal of Automatica Sinca*, vol. 6, no. 3, pp. 641–656, 2019.

[6] M. Nabatirad, R. Razzaghi, and B. Bahrani, "Decentralized voltage regulation and energy management of integrated dc microgrids into ac power systems," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 9, no. 2, pp. 1269–1279, 2021.

[7] L. Subramanian, V. Debusschere, H. B. Gooi, and N. Hadjsaid, "A distributed model predictive control framework for grid-friendly distributed energy resources," *IEEE Transactions on Sustainable Energy*, vol. 12, no. 1, pp. 727–738, 2021.

[8] M. Sato, Y. Fukuyama, T. Iizaka, and T. Matsui, "Total optimization of energy networks in a smart city by multi-swarm differential evolutionary particle swarm optimization," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 4, pp. 2186–2200, 2019.

[9] Q. Li, X. Zou, Y. Pu, and W. Chen, "A real-time energy management method for electric-hydrogen hybrid energy storage microgrid based on dp-mpc," *IEEE CSEE Journal of Power and Energy Systems*, vol. Early Access, pp. 1–13, 2021.

[10] Y. Jiang, M. Liu, H. Peng, and M. Z. A. Bhuiyan, "A reliable deep learning-based algorithm design for iot load identification in smart grid," *Ad Hoc Networks*, vol. 123, p. 102643, 2021.

[11] K. Thirugnanam, M. S. El Moursi, V. Khadkikar, H. H. Zeineldin, and M. Al Hosani, "Energy management of grid interconnected multi-microgrids based on p2p energy exchange: A data driven approach," *IEEE Transactions on Power Systems*, vol. 36, no. 2, pp. 1546–1562, 2021.

[12] Y. Du, J. Wu, S. Li, C. Long, and S. Onori, "Coordinated energy dispatch of autonomous microgrids with distributed mpc optimization," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5289–5298, 2019.

[13] T. Pippa, J. Sijs, and B. D. Schutter, "A single-level rule-based model predictive control approach for energy management of grid-connected microgrids," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2364–2376, 2020.

[14] K. Tang, R. Shi, H. Shi, M. Z. A. Bhuiyan, and E. Luo, "Secure beamforming for cognitive cyber-physical systems based on cognitive radio with wireless energy harvesting," *Ad Hoc Networks*, vol. 81, pp. 174–182, 2018.

[15] A. Cecilia, S. Sahoo, T. Dragicevic, R. Costa-Castello, and F. Blaabjerg, "On addressing the security and stability issues due to false data injection attacks in dc microgrids—an adaptive observer approach," *IEEE Transactions on Power Electronics*, vol. 37, no. 3, pp. 2801–2814, 2022.

[16] A. S. Mohamed, M. Arani, A. A. Jahromi, and D. Kundur, "False data injection attacks against synchronization systems in microgrids," *IEEE Transactions on Smart Grid*, vol. 12, no. 5, pp. 44–71, 2021.

[17] M. R. Habibi, H. R. Baghaee, T. Dragicevic, and F. Blaabjerg, "Detection of false data injection cyber-attacks in dc microgrids based on recurrent neural networks," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 9, no. 5, pp. 5294–5310, 2021.

[18] O. A. Begi, T. T. Johnson, and A. Davoudi, "Detection of false-data injection attacks in cyber-physical dc microgrids," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2693–2703, 2017.

[19] M. Jorjani, J. Seifi, and A. Y. Varjani, "A graph theory-based approach to detect false data injection attacks in power system ac state estimation," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2465–2475, 2021.

[20] B. Patnaik, M. Mishra, R. C. Bansal, and R. K. Jena, "Ac microgrid protection – a review: Current and future prospective," *Applied Energy*, vol. 271, no. 115210, pp. 1–28, 2020.

[21] X. Kang, C. E. K. Nuworklo, B. S. Tekpeti, and M. Kheshti, "Protection of micro-grid systems: a comprehensive survey," *The Journal of Engineering*, no. 13, pp. 1515–1518, 2017.

[22] Q. Zhou, M. Shahidehpour, A. Alabdulwahab, and A. Abusorrah, "A cyber-attack resilient distributed control strategy in islanded microgrids," *IEEE Transactions on Smart Grid*, vol. 11, no. 5, pp. 3690–3701, 2020.

[23] M. Ban, M. Shahidehpour, J. Yu, and Z. Li, "A cyber-physical energy management system for optimal sizing and operation of networked nanogrids with battery swapping stations," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 1, pp. 491–502, 2019.

[24] A. J. Gallo, M. S. Turan, F. Boem, T. Parisini, and G. Ferrari-Trecate, "A distributed cyber-attack detection scheme with application to dc microgrids," *IEEE Transactions on Automatic Control*, vol. 65, no. 9, pp. 3800–3815, 2020.

[25] A. Bidram, B. Poudel, L. Damodaran, R. Fierro, and J. M. Guerrero, "Resilient and cybersecure distributed control of inverter-based islanded microgrids," *IEEE Transactions on Industrial informatics*, vol. 16, no. 6, pp. 3881–3894, 2020.

[26] S. Sahoo, T. Dragicevic, and F. Blaabjerg, "Multilayer resilience paradigm against cyber attacks in dc microgrids," *IEEE Transactions on Power Electronics*, vol. 36, no. 3, pp. 2522–2532, 2021.

[27] A. Mustafa, B. Poudel, A. Bidram, and H. Modares, "Detection and mitigation of data manipulation attacks in ac microgrids," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2588–4649, 2020.

[28] O. Beg, A. Johnson, and A. Davoudi, "Detection of false-data injection attacks in cyber-physical dc microgrids," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2693–2603, 2017.

[29] B. M. Ruhul Amin, S. Taghizadeh, S. Maric, M. J. Hossain, and R. Abbas, "Smart grid security enhancement by using belief propagation," *IEEE Systems Journal*, vol. 15, no. 2, pp. 2046–2057, 2021.

[30] C. Fang, Y. Qi, J. Chen, R. Tan, and W. X. Zheng, "Stealthy actuator signal attacks in stochastic control systems: Performance and limitations," *IEEE Transactions on Automatic Control*, vol. 65, no. 9, pp. 3927–3934, 2020.

[31] K. Wang, L. Yuan, T. Miyazaki, Y. Chen, and Y. Zhang, "Jamming and eavesdropping defense in green cyber–physical transportation systems using a stackelberg game," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4232–4242, 2018.

[32] J. Hao, R. J. Piechocki, D. Kaleshi, W. H. Chin, and Z. Fan, "Sparse malicious false data injection attacks and defense mechanisms in smart grids," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 5, pp. 1198–1209, 2015.

[33] J. Zhao and L. Mili, "A robust generalized-maximum likelihood unscented kalman filter for power system dynamic state estimation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 4, pp. 578–592, 2018.

[34] L. Guo, B. Yang, J. Ye, H. Chen, F. Li, W. Song, L. Du, and L. Guan, "Systematic assessment of cyber-physical security of energy management system for connected and automated electric vehicles," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 3335–3347, 2021.

[35] J. Shi, S. Liu, B. Chen, and L. Yu, "Distributed data-driven intrusion detection for sparse stealthy fdi attacks in smart grids," *IEEE Transactions on Circuits and Systems-II*, vol. 68, no. 3, pp. 993–997, 2021.

[36] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Systems Journal*, vol. 11, no. 3, pp. 1644–1652, 2014.

[37] A. Sargolzaei, K. Yazdani, A. Abbaspour, C. D. Crane III, and W. E. Dixon, "Detection and mitigation of false data injection attacks in networked control systems," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4281–4292, 2019.

[38] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[39] A. Kurakin, I. Goodfellow, S. Bengio *et al.*, "Adversarial examples in the physical world," 2016.

[40] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2574–2582.

[41] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *2016 IEEE symposium on security and privacy (SP)*. IEEE, 2016, pp. 582–597.

[42] 2022, "SIEMS source code," https://github.com/mshojafar/sourcecodes/raw/master/SIEMS_Sourcecode.zip.

[43] R. Taheri, R. Javidan, and Z. Pooranian, "Adversarial android malware detection for mobile multimedia applications in iot environments," *Multimedia Tools and Applications*, vol. 80, no. 11, pp. 16 713–16 729, 2021.

[44] X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi, "A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability," *Computer Science Review*, vol. 37, p. 100270, 2020.

[45] X. Yan, Y. Xu, X. Xing, B. Cui, Z. Guo, and T. Guo, "Trustworthy network anomaly detection based on an adaptive learning rate and momentum in iiot," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6182–6192, 2020.

Pedram Asef (S'13-M'18-SM'20) is a Senior Lecturer in Automotive Engineering specialising in Automotive Electrical and Electromechanical Systems at the Department of Engineering and Technology, University of Hertfordshire. He received a Ph.D. degree in Electrical Engineering from the Universitat Politècnica de Catalunya-BarcelonaTech (UPC) with cum laude grade, Barcelona, Spain, in 2018. He is also Lead Representative of IEEE Power and Energy Society, Young Professionals, in Europe region, since 2018. His previous and current projects are funded by the European Space Agency (ESA), UKRI, European Commission, and Texas State Center for Port Management on Transportation Electrification, Electrical Machines and Intelligent Energy Systems. Dr. Pedram Asef is a PI of AutoTrust, a 750k euro 5G secure connected vehicular communication project supported by ESA in 2021. He is also a PI of AutoPower project, a £100k project on intelligent energy management system development for vehicles funded by European Commission. He received the Chinese Presidential Award and Shanghai Jiao Tong University Fellowship in 2013 and 2015, respectively. His Editorial activities are limited to IET Generation, Transmission & Distribution, Sustainability, and Distributed Generation & Alternative Energy Journals. In 2019, he was appointed as the Star Reviewer of IEEE Transactions on Energy Conversion. His fields of interest include Electrical Machinery and Drives, Intelligent Energy Systems, and Machine Learning. Recently, he has published +20 research papers at international conferences, journals, and registered patents.For more information: https://www.go.herts.ac.uk/pedram_asef

Rahim Taheri is a postdoctoral researcher working in the Centre for Telecommunications Research at the King College University, London, United Kingdom. He obtained his Ph.D. in Information Technology- Computer Networks from Shiraz University of Technology, Iran, in January 2020. He received his M.Sc. degree of Computer Networks from the same university and B.Sc. degree of Computer Engineering from Bahonar Technical and Engineering College of Shiraz in 2015 and 2007, respectively. He was a visiting Ph.D. student in the SPRITZ Security and Privacy Research Group at the University of Padua in 2018. His main research interests include Adversarial Machine Learning, Network Security and Differential Privacy, security of Cloud Storage, Software Defined Networks.

Mohammad Shojafar (M'17-SM'19) is a Senior Lecturer (Associate professor) is a Senior Lecturer (Associate Professor) in the network security and an Intel Innovator, Professional ACM member and ACM Distinguished Speaker, and a Marie Curie Alumni, working in the 5G & 6G Innovation Centre (5GIC & 6GIC), Institute for Communication Systems (ICS), at the University of Surrey, UK. Before joining 5GIC/6GIC, he was a Senior Researcher and a Marie Curie Fellow in the SPRITZ Security and Privacy Research group at the University of Padua, Italy. Dr. Mohammad is a PI of AutoTrust, a 750k euro 5G secure autonomous vehicular communication project supported by European Space Agency (ESA) in 2021 and was a PI of PRISENODE project, a 275k euro Horizon 2020 Marie Curie global fellowship project in the areas of Fog/Cloud security collaborating at the University of Padua, Italy. He also was a PI on an Italian SDN security and privacy (60k euro) supported by the University of Padua in 2018. He was contributed to some Italian projects in telecommunications like GAUChO, SAMMClouds, and SC2. He received his Ph.D. degree in ICT from Sapienza University of Rome, Rome, Italy, in 2016 with an "Excellent" degree. He is an Associate Editor in *IEEE Transactions on Network and Service Management*, *IEEE Transactions on Consumer Electronics*, *IEEE Systems Journal* and Computer Networks. For additional information: http://mshojafar.com

Rahim Tafazolli (SM'09) is a professor and the Director of the Institute for Communication Systems (ICS) and 5G Innovation Centre (5GIC), the University of Surrey in the UK. He has over 30 years of experience in digital communications research and teaching. He has published more than 500 research papers in refereed journals, international conferences and as invited speaker. He is the editor of two books on "Technologies for Wireless Future" published by Wiley Vol.1 in 2004 and Vol.2 2006. He is co-inventor on more than 30 granted patents, all in the field of digital communications. He was appointed as Fellow of WWRF (Wireless World Research Forum) in April 2011, in recognition of his personal contribution to the wireless world. As well as heading one of Europa's leading research groups. He is regularly invited by governments to advise on network and 5G technologies and was advisor to the Mayor of London with regard to the London Infrastructure Investment 2050 Plan during May and June 2014. For more information: https://www.surrey.ac.uk/people/rahim-tafazolli