Collaborative Intrusion Detection System for Unmanned Aerial Vehicles Swarm Security

Leandro Marcos da Silva[1], **Kalinka R. L. J. C. Branco**[1]

[1]Institute of Mathematics and Computer Sciences – University of São Paulo (USP)
Ave. Trabalhador São-Carlense, 400, São Carlos 13564-002, São Paulo, Brazil

leandro.marcos@usp.br, kalinka@icmc.usp.br

Abstract. *Unmanned Aerial Vehicles (UAVs) are increasingly used in military, civil, and commercial applications, but they are vulnerable to cyberattacks that threaten security and privacy. These threats include signal interception, unauthorized access, data theft, and remote control of UAVs. An Intrusion Detection System (IDS) is used to mitigate these risks. However, most IDS focus on individual data sources, failing to detect swarm-specific threats. This work presents REMY, a collaboRative intrusion dEtection system for unManned aerial vehicles swarm SecuritY that detects network and in-flight anomalies using machine learning techniques. REMY identifies network attacks such as blackhole, grayhole, and flooding, as well as in-flight threats like GPS spoofing and jamming. Federated learning is applied in model training to ensure data privacy and collaboration. The system is designed to be hardware-independent and lightweight, with low energy consumption and efficient use of resources.*

1. Introduction

The widespread adoption of Unmanned Aerial Vehicles (UAVs), or drones, across sectors like communications, disaster management, environmental monitoring, and smart cities is driven by their reconfigurability, rapid response, and ease of deployment [Zaidi et al. 2021]. UAVs also contribute to public safety, surveillance, medical services, and military operations. When integrated with the Internet of Things (IoT), forming the Internet of Flying Things (IoFT), UAVs extend IoT services, especially in areas lacking infrastructure, enhancing mobility and flexibility [Xue et al. 2018]. The Urban Air Mobility (UAM) market, including air taxis and UAV deliveries, is projected to contribute 11% to global GDP by 2050, with UAVs playing a key role in future 6G connectivity. However, UAVs face challenges like collision risks, interference, deployment issues, power consumption, and security vulnerabilities due to their wireless transmission nature [Zaidi et al. 2021]. Threats like GPS spoofing and jamming, often executed with low-cost Software-Defined Radio (SDR) tools, pose significant risks, such as a GPS jamming attack in Hong Kong that caused considerable damage [Whelan et al. 2022]. Strategies such as secure routing protocols, authentication, and Intrusion Detection Systems (IDS) are recommended to mitigate these risks, where IDS monitor system settings and network traffic to identify abnormal behavior and alert the Ground Control Station (GCS) to potential threats [Choudhary et al. 2018].

UAV IDS can be based on signature, anomaly, specification, or hybrid models, with anomaly-based systems being the most common, often utilizing Deep Learning (DL) to detect threats like flooding and GPS spoofing. These systems typically rely on

information from communication links or sensors [Choudhary et al. 2018], but current IDS only detect attacks from a single source, lacking sufficient security for the entire UAV system. Moreover, conventional IDS datasets, such as CIC-IDS2017, do not represent UAV communication, limiting their real-world applicability [Ahmed et al. 2022]. Few studies address IDS for UAV swarms, often ignoring swarm-specific threats like blackhole, grayhole, and flooding attacks, while focusing on routing protocols and battery life [Basan et al. 2021]. Federated Learning (FL) is a promising solution for collaborative attack detection, enabling UAVs to share threat information within the swarm [He et al. 2022]. Given the limited hardware in UAVs, lightweight IDS development is essential [Whelan et al. 2022]. The proposed IDS in this work aims to address these gaps, providing enhanced security for UAV swarms with collaborative and hardware-independent features.

Therefore, the main goal of this work is to develop an anomaly-based IDS for detecting threats in UAVs, specifically in swarms, called REMY. The objectives include detecting UAV swarm network attacks like blackhole, grayhole, and flooding, as well as flight anomalies such as GPS spoofing and jamming. The system applies Machine Learning (ML) techniques, both supervised and unsupervised, based on the data source and attack type. It also utilizes the geographic and physical features of the UAV to ensure hardware independence. Additionally, the system incorporates FL for collaborative model building and data privacy, while focusing on creating a lightweight IDS with low power consumption, efficient resource usage, and low-latency attack detection.

2. Related Works

Several techniques for implementing IDS in UAVs are being developed, including using physical sensor data to detect attacks and analyzing network traffic for suspicious patterns [Choudhary et al. 2018]. ML algorithms are increasingly utilized to enhance detection accuracy and improve system efficiency. Additionally, FL is being explored to ensure privacy, integrity, and collaboration in model training [He et al. 2022]. Recent research has focused on key aspects such as flight anomaly and network attack detection, FL, and swarm exclusivity. REMY integrates these features into a single system, emphasizing collaboration between UAVs.

The IDS proposed by [Park et al. 2020] use an unsupervised approach to detect unknown threats without needing labeled data, training an autoencoder with benign flight data to identify anomalies like GPS spoofing and jamming based on reconstruction loss. The study also emphasizes using geographic and physical features to ensure hardware independence. [Basan et al. 2021] discusses anomaly detection in UAV swarms, detecting attacks like SYN-flood in different network topologies with an Multi-Layer Perceptron (MLP), and monitoring changes in traffic, CPU usage, and device temperature. [Zhang et al. 2022] introduces the AMDES system, using multi-fractal spectral analysis and neural networks to detect Man-in-the-Middle (MITM) attacks with high accuracy and low false positives. [Whelan et al. 2022] employs Principal Component Analysis (PCA) and one-class classifiers to detect flight anomalies like GPS spoofing, working effectively on resource-constrained vehicles such as the Raspberry Pi Zero. Lastly, [He et al. 2022] uses Long Short-Term Memory (LSTM) and Conditional Generative Adversarial Network (CTGAN) for detecting network intrusions, enhancing training data via blockchain-powered FL, which improves detection performance and sample generalization.

The presented IDS apply ML techniques to detect network intrusions [Basan et al. 2021, Zhang et al. 2022, He et al. 2022] or flight anomalies [Park et al. 2020, Whelan et al. 2022]. Only [Basan et al. 2021] study mentions exclusivity for swarms, with changes in network traffic when a neighboring UAV is attacked. FL is applied only in [He et al. 2022], with the advantage of ensuring data security and privacy and collaborative training. Finally, none of the IDS focuses on detecting flight anomalies and network attacks, only working with one or the other, and not leveraging flight data for FL.

3. Material and Methods

This section will present the methodology applied to the development of REMY. The IDS is divided into two detection subsystems: (1) Network Attack Detection and (2) Flight Anomaly Detection. The division is illustrated in Figure 1, which shows the IDS acting when an attack happens. The system allows monitoring of the mesh network traffic data and individual flight data. When an intrusion is detected, the GCS is notified. The novelty of this work is that it detects threats both in the network and in the UAV sensors, something that has still not been addressed in the literature. UAV swarm exclusivity is also differential, allowing collaboration to generate a generic model.

Figure 1. REMY in Operation

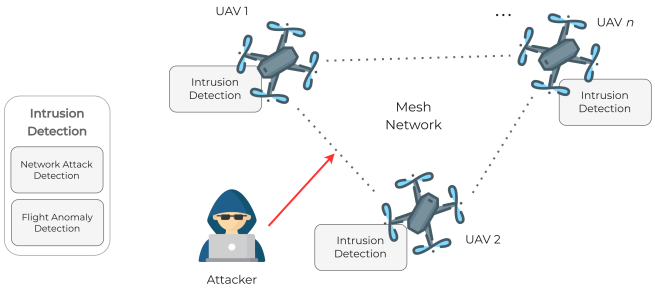


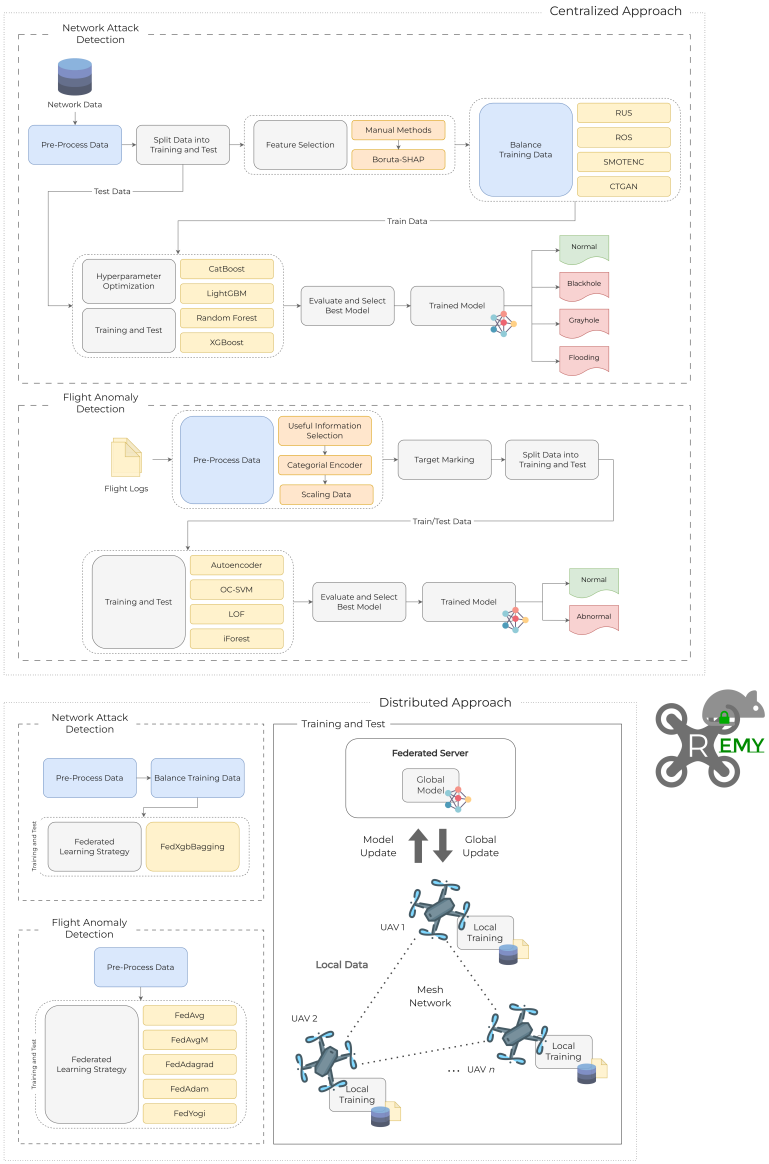
Figure 2 provides an overview of REMY. The development process of the subsystems, which together make up the REMY, is detailed below. Each uses a different ML method: supervised learning for detecting network attacks, such as the XGBoost algorithm, and unsupervised learning for in-flight anomalies, such as the stacked autoencoder. Furthermore, experimentation is divided into centralized and distributed approaches, with the distributed one being applied to FL, corresponding to the final operation of REMY.

3.1. Network Attack Detection

Network data for detecting attacks on the UAV network is extracted from the WSN-DS [Almomani et al. 2016] dataset. Thus, network behavior is classified as normal or under blackhole, grayhole, and flooding attacks in REMY. The data are ordered by the time column and divided into 80% for training and validation, and 20% for testing, due to a large number of samples in the dataset and this proportion being standardized in both subsystems developed. After this, the number of samples for each dataset is as follows:

- **Training and Validation:** 272,053 benign; 11,677 grayhole; 8,039 blackhole; and 2,650 flooding data;
- **Test:** 68,013 benign; 2,919 grayhole; 2,010 blackhole; and 662 flooding data.

Figure 2. System Overview



In data pre-processing, non-useful columns were excluded, scheduling attack samples were removed from the dataset, and labels were converted to numeric. Regarding the features used in the WSN-DS, the 16 already present were selected with manual methods, dropping (1) features with Spearman correlation greater than 0.80 or (2) with Information Value (IV) less than 0.02. This process resulted in 12 features. Also, Boruta-SHAP was applied to select a subset of features. As a result, eight features were chosen from the previous 12. With this subset of features, obtaining a simple model with good results was possible, as will be presented in Section 4.

As the dataset presents the problem of data imbalance, having more data from the normal label than the rest, data balancing is used. In this work, undersampling and oversampling methods are used. Random Undersampling (RUS) is applied to the benign label data, reducing sample sizes to the total number of minority label samples, encompassing

the three attacks. Then, when reduced, the number of benign label samples is 22,366. Next, four oversampling methods are compared to generate data in the minority labels, increasing the data by 50%. The applied methods are Random Oversampling (ROS), SMOTE for Nominal and Continuous (SMOTENC), and CTGAN. At the end of the process, the following number of samples for each minority label for training and validation was obtained: 17,515 for grayhole, 12,058 for blackhole, and 3,975 for flooding label.

Supervised algorithms, Random Forest (RF), XGBoost, CatBoost, and LightGBM, are applied for model training to compare which algorithm obtains a better result. Algorithms based on the gradient boosting method are focused on because they have faster training speeds and higher efficiency, especially XGBoost, which supports FL in the Flower framework. The relevance of XGBoost is its low memory consumption, which is crucial when running on embedded systems.

The hyper-parameters were optimized with Bayesian Optimization in the Optuna framework to improve the result and avoid overfitting, being optimized *learning_rate*, *n_estimators*, *max_depth*, *reg_alpha*, and *reg_lambda*, when existing in the algorithm. The stratified k-fold cross-validation technique with $k = 5$ is applied during training and validation. The evaluation metrics used to evaluate the algorithms are the macro F1-score and Area Under the Curve ROC (AUC ROC). The One-vs-Rest (OvR) approach is used to calculate the AUC ROC due to the multiclass classification. After that, the model is generated and is ready to be tested with new data.

3.2. Flight Anomaly Detection

Flight logs are acquired from the Parrot Bebop 2 UAV, collected with the PyParrot package. PyParrot is necessary for UAV control and data collection. Data is collected at a frequency of 100 ms and saved in CSV format using a script developed in Python. Flight data was collected in normal conditions and under anomalies. The HackRF device was used to perform the attacks.

In these logs, specific useful flight information are selected based on previous studies [Park et al. 2020, Whelan et al. 2022]. As the system is intended to work on any UAV, features related to physical and geographic properties are selected regardless of hardware. They are based on internal measurements, location, position and orientation, and system status [Park et al. 2020]. Thus, 24 features are selected for this subsystem, divided into the abovementioned categories. The categorical encoder is performed on categorical features, transforming each category into a new column. The final result is 37 features after the encoder.

As the autoencoder is used, features must be scaled to speed up the optimization process; feature scaling is necessary to work with the data at the same scale. The min-max normalization is employed. In the target marking stage, the time of the flight, when the intrusion started, is defined, making it possible to evaluate the output of the trained model later in the test. To accomplish this, the time the abnormal change in latitude and longitude began is checked in the flight scenarios with the GPS threat. Only the data without abnormalities is used for training.

The purpose of using one-class classifiers is to generate a normality model from the data of benign samples and identify data that are considered outliers [Whelan et al. 2022]. The algorithms for this work are stacked autoencoder, One-

Class SVM (OC-SVM), Local Outlier Factor (LOF), and Isolation Forest (iForest). The hyper-parameters used in OC-SVM, LOF, and iForest are the defaults from scikit-learn. The autoencoder is the focus because it was highlighted in previous studies [Park et al. 2020, Whelan et al. 2022] in-flight anomaly detection, particularly GPS spoofing and jamming. The F1-score, AUC ROC, and log loss are considered evaluation metrics in these classifiers.

The autoencoder compresses the data into a reduced dimension in the encoder step. Then, the decoder performs the decompression, having as output the same number of features as the input. In this process, the reconstruction error is generated. Therefore, if the data reconstruction error is too high, it can be classified as an anomaly. A threshold is set for benign samples. When something is above that threshold, it can be classified as a novelty.

To get a metric to measure the performance of the unsupervised method, 20% of the labeled dataset is used as a test. In training, only benign flight data are employed because it is challenging to sample anomalies in practice, in addition to the excellent performance of one-class classifiers in detecting outliers [Park et al. 2020]. Of the benign training data, 10% is used to calculate the benign threshold in the autoencoder. 80% of the attack data is applied to estimate the anomaly threshold, and the rest is for testing. With new data, it is verified which threshold the reconstruction error is closest to from Equation 1, where it is classified as an anomaly if it meets the condition.

$$anomaly = |loss - threshold_{normal}| > |loss - threshold_{abnormal}| \quad (1)$$

3.3. Federated Learning

In FL, decentralized learning is promoted, which preserves data privacy and security. It is used in this system to bring the collaborative feature of the UAV swarm and privacy in sensitive data. This is interesting to guarantee the privacy of the network and flight information, preventing an attack from intercepting the communication between a UAV and a server, and thus obtaining the data.

Data pre-processing is applied to network attack and flight anomaly detection. Furthermore, balancing is applied to the network data with CTGAN. The training is conducted locally in each vehicle, and the parameters obtained are sent to the central server. The parameters are aggregated on this server, generating the global model. The Flower framework is used to apply FL, where different parameter aggregation strategies are available in the framework. FedXgbBagging strategy is used for network attack detection. In turn, strategies FedAvg, FedAvgM, FedAdagrad, FedAdam, and FedYogi are compared in the subsystem for flight anomaly detection. Finally, FL is only applied with the stacked autoencoder classifier to detect flight anomaly, which is expected to perform better than the others [Whelan et al. 2022] and its ease with the Flower. In turn, only XGBoost is used in the network system, due to its support in the framework and good results in centralized training and testing, which will be presented in Section 4.1.

4. Results and Discussion

This section presents the results obtained in this work. The network attack and in-flight anomaly detection results were based on the WSN-DS dataset and flight data collected

during the experiments, respectively. Thus, supervised algorithms and data balancing techniques are compared in network attack detection on test data. In in-flight anomaly detection, one-class classifiers are evaluated. Finally, the FL evaluation strategies are compared in both subsystems.

4.1. Network Attack Detection

The evaluation of the supervised approach for network attack detection is presented below, covering the comparison of ML algorithms, data balancing methods, and test data classification. The hyper-parameters of the algorithms were optimized with the Optuna framework in 25 trials. Table 1 shows the evaluation of oversampling methods and algorithms about F1-score and AUC ROC in training and testing. The acronym (BF) indicates Boruta Features.

Table 1. Oversampling Methods and ML Algorithms Evaluation

Oversampling Method	Algorithm	F1-Score	AUC ROC	Test F1-Score	Test AUC ROC
Only RUS	CatBoost	97.12 ± 0.18	99.90 ± 0.01	88.02	99.91
	LightGBM	97.82 ± 0.19	99.94 ± 0.01	91.91	99.94
	RF	97.29 ± 0.16	99.87 ± 0.04	90.11	99.60
	XGBoost	97.69 ± 0.23	99.93 ± 0.01	92.31	99.94
	XGBoost (BF)	97.42 ± 0.18	99.92 ± 0.01	91.07	99.93
ROS	CatBoost	97.95 ± 0.11	99.93 ± 0.01	87.89	99.90
	LightGBM	98.58 ± 0.15	99.95 ± 0.01	91.71	99.94
	RF	98.18 ± 0.18	99.91 ± 0.02	89.55	99.58
	XGBoost	98.54 ± 0.18	99.95 ± 0.01	90.30	99.94
	XGBoost (BF)	98.43 ± 0.21	99.94 ± 0.01	90.64	99.93
SMOTENC	CatBoost	97.57 ± 0.17	99.91 ± 0.01	85.52	99.88
	LightGBM	98.13 ± 0.12	99.93 ± 0.01	90.28	99.89
	RF	97.78 ± 0.08	99.88 ± 0.02	86.43	99.53
	XGBoost	98.01 ± 0.14	99.93 ± 0.01	89.63	99.87
	XGBoost (BF)	97.97 ± 0.15	99.92 ± 0.01	90.17	99.87
CTGAN	CatBoost	97.43 ± 0.18	99.91 ± 0.01	86.97	99.90
	LightGBM	98.00 ± 0.13	99.94 ± 0.01	91.23	99.94
	RF	97.29 ± 0.19	99.87 ± 0.02	88.98	99.70
	XGBoost	97.83 ± 0.16	99.93 ± 0.01	91.72	99.94
	XGBoost (BF)	97.45 ± 0.11	99.91 ± 0.01	89.63	99.92

Of the results in the table, the oversampling methods had similar results. However, the highlight is CTGAN, a recent method for generating synthetic data, being selected as the default. The GAN method has already been applied for data augmentation in the UAV scenario in [He et al. 2022]. Regarding the ML algorithms, the worst results were for CatBoost and RF in all scenarios, both in the F1-score and AUC ROC. In addition, CatBoost and LightGBM algorithms took hours to optimize hyper-parameters, while the other algorithms took minutes. Good results were obtained in gradient boosting algorithms, with similar metrics in CatBoost, LightGBM, and XGBoost. The result mostly stayed the same with only the subset of features, i.e., selected with the Boruta-SHAP method. The high values obtained in the metrics are due to working with a simulated dataset, and there may be a drop in the real environment. Therefore, due to the low variation in the results and the support for FL, XGBoost (BF) with CTGAN was chosen as a simpler model due to the smaller number of features.

In the IDS developed by [Ramadan et al. 2021], the WSN-DS dataset is also used. Because of this, it becomes possible to compare with this work, where even applying the LSTM, the results of both works were similar, with an F1-score close to 90.00. The authors also test on other datasets and compare the results, such as KDD-99, NSL-KDD, UNSW-NB15, CIC-IDS2017, and TON_IoT, reference datasets for IDS.

4.2. Flight Anomaly Detection

Next, the results of the unsupervised approach are presented, covering the evaluation of one-class classifiers. The evaluation of the four classifiers for detecting flight anomalies is represented in Table 2, where the algorithms were trained only with benign data. The results indicated that the stacked autoencoder performed best regarding the test F1-score, AUC ROC, and log loss metrics, followed by LOF and iForest. The OC-SVM obtained the worst results, presenting lower values in all evaluated metrics.

Table 2. Anomaly Detection Evaluation

Algorithm	Test F1-Score	Test AUC ROC	Test Log Loss
Stacked Autoencoder	95.73	95.90	1.73
OC-SVM	83.75	72.69	8.18
LOF	94.57	91.92	2.42
iForest	92.66	88.85	3.34

In the [Whelan et al. 2022] study, OC-SVM, LOF, and autoencoder algorithms were also used for an unsupervised approach to detecting GPS spoofing and jamming attacks in the UAV Attack Dataset. However, attacks are tested separately. The results indicated that the detection of these attacks obtained an average F1-score of 90.57 and 94.30 in the autoencoder, respectively. The metrics achieved in this study showed similar efficiency, reinforcing the effectiveness of the methods employed and the possibility of improving with more data during the training step.

4.3. Distributed Approach

The metrics resulting from the decentralized approach, FL, will be presented below. The Flower framework, with scripts for the server and client, was used to implement FL. It should be noted that the selected hyper-parameters are framework defaults, changing only the aggregation strategies during the experiments.

For the network attack detection subsystem, the FedXgbBagging aggregation strategy was used. Table 3 shows the result in terms of average F1-score and AUC ROC. The metrics obtained are close to the centralized one in ten rounds, validating the efficiency with FL – the metric drops over the rounds, especially due to the dataset split.

Table 3. Average Metrics in FL Strategies with Network Attack Detection

Aggregate Method	Average F1-Score	Average AUC ROC
FedXgbBagging	88.04	99.71

In the flight anomaly detection, the following strategies were tested: FedAvg, FedAvgM, FedAdagrad, FedYogi, and FedAdam. Table 4 shows the evaluation of the aggregation strategies. The results are the average F1-score and AUC ROC over ten rounds.

The analysis of the methods indicates that FedYogi and FedAdagrad performed the best. FedAvg was another method that obtained interesting results, with a 4% difference compared to the best result. Thus, it can be considered viable, being the default used in the framework. Based on the metrics and variation of the rounds, the selected strategy was FedYogi, applying adaptive federated optimization algorithms.

Table 4. Average Metrics in FL Strategies with Flight Anomaly Detection

Aggregate Method	Average F1-Score	Average AUC ROC
FedAvg	78.84	81.28
FedAvgM	79.69	84.69
FedAdagrad	80.69	85.86
FedAdam	79.36	84.92
FedYogi	82.90	86.36

5. Conclusions

This work presents a new collaborative IDS for UAV security called REMY. In this way, REMY aims to detect in-flight anomalies, such as GPS spoofing and jamming, and UAV network attacks, such as blackhole, grayhole, and flooding, ensuring a wider range of identified threats. For this, supervised and unsupervised learning techniques are applied in the detection, together with the FL, which guarantees the privacy of the data and the collaboration of the UAVs to build the global model. In addition, the IDS uses geographic and physical flight features to bring hardware independence. Thus, the IDS has the advantage of being aimed at a swarm of UAVs due to the threats detected on the network and the collaborative resource.

The WSN-DS dataset, which contains mesh network traffic with DoS attacks, was used to build the system. Furthermore, flight data was collected with Parrot Bebop 2 under normal conditions and anomalies, using the HackRF device to make GPS threats. From the results, the subsystem for network attack detection achieved an F1-score of 89.63 in the centralized approach with supervised algorithm XGBoost, with Boruta features, and applying CTGAN to balance the data and the hyper-parameters optimized with Bayesian optimization. FL application, with Flower framework, obtained an F1-score of 88.04 in ten rounds with FedXgbBagging. The focus was primarily on XGBoost with fewer features to develop a lightweight IDS, as it is a simpler model and takes less time to optimize hyper-parameters with Optuna.

The unsupervised stacked autoencoder algorithm stood out to identify flight anomalies, achieving an F1-score of 95.73 in the test flight data. The federated configuration presented an F1-score of 82.90 in ten rounds with the FedYogi aggregation strategy, the one with the best performance among those compared. The results highlight the relevance of the distributed approach, even with a reduction in the metrics, ensuring scalability, privacy preservation, and collaboration between UAVs in the swarm.

All code developed in this work is available in a GitHub Repository: `https://github.com/silvamleandro/remy-project`.

References

Ahmed, M., Cox, D., Simpson, B., and Aloufi, A. (2022). Ecu-ioft: A dataset for analysing cyber-attacks on internet of flying things. *Applied Sciences*, 12(4):1990.

Almomani, I., Al-Kasasbeh, B., and Al-Akhras, M. (2016). Wsn-ds: A dataset for intrusion detection systems in wireless sensor networks. *Journal of Sensors*, 2016.

Basan, E., Lapina, M., Mudruk, N., and Abramov, E. (2021). Intelligent intrusion detection system for a group of uavs. In *Advances in Swarm Intelligence: 12th International Conference, ICSI 2021, Qingdao, China, July 17–21, 2021, Proceedings, Part II 12*, pages 230–240. Springer.

Choudhary, G., Sharma, V., You, I., Yim, K., Chen, R., and Cho, J.-H. (2018). Intrusion detection systems for networked unmanned aerial vehicles: A survey. In *2018 14th International Wireless Communications & Mobile Computing Conference*, pages 560–565. IEEE.

He, X., Chen, Q., Tang, L., Wang, W., and Liu, T. (2022). Cgan-based collaborative intrusion detection for uav networks: A blockchain-empowered distributed federated learning approach. *IEEE Internet of Things Journal*, 10(1):120–132.

Park, K. H., Park, E., and Kim, H. K. (2020). Unsupervised intrusion detection system for unmanned aerial vehicle with less labeling effort. In *International Conference on Information Security Applications*, pages 45–58. Springer.

Ramadan, R. A., Emara, A.-H., Al-Sarem, M., and Elhamahmy, M. (2021). Internet of drones intrusion detection using deep learning. *Electronics*, 10(21):2633.

Whelan, J., Almehmadi, A., and El-Khatib, K. (2022). Artificial intelligence for intrusion detection systems in unmanned aerial vehicles. *Computers and Electrical Engineering*, 99:107784.

Xue, Z., Wang, J., Ding, G., Zhou, H., and Wu, Q. (2018). Maximization of data dissemination in uav-supported internet of things. *IEEE Wireless Communications Letters*, 8(1):185–188.

Zaidi, S., Atiquzzaman, M., and Calafate, C. T. (2021). Internet of flying things (ioft): A survey. *Computer Communications*, 165:53–74.

Zhang, R., Condomines, J.-P., and Lochin, E. (2022). A multifractal analysis and machine learning based intrusion detection system with an application in a uas/radar system. *Drones*, 6(1):21.