Unmasking Threats in UAV Networks: A Semi-Supervised Approach to Cyphal Security

Kabid Hassan Shibly, Ryoichi Isawa and Takahiro Kasama
Cybersecurity Laboratory, Cybersecurity Research Institute,
National Institute of Information and Communications Technology (NICT)
Tokyo, Japan
e-mail: {khassans | isawa | kasama}@nict.go.jp

***Abstract*—With increased connectivity in In-Vehicle Networks (IVNs), protocols like Cyphal, used in Unmanned Aerial Vehicles (UAVs), are vulnerable to cyber threats, including flooding, fuzzy, and replay attacks. Traditional Intrusion Detection Systems (IDS) rely on supervised learning and struggle with evolving attacks due to the need for large volumes of labeled data. We propose Gravity Well Learning (GWL), a novel semi-supervised learning framework for intrusion detection in Cyphal networks. GWL leverages both labeled and unlabeled data to enhance detection accuracy while reducing reliance on extensive labeled datasets. It introduces a central "Planet" model, guided by expert "Gravity Wells" that refine detection capabilities. Experiments show that GWL achieves 65.50% accuracy with 10% labeled data and 83.10% with 40%. These results underscore GWL's robustness and scalability in securing UAV and automotive networks, making GWL a promising solution for real-world intrusion detection in vehicular communication systems.**

Keywords-unmanned aerial vehicle; cyphal; in-vehicle networks; cybersecurity; intrusion detection systems; semi-supervised learning.

I. Introduction

Uncomplicated Application-level Vehicular Computing and Networking (UAVCAN), also known as Cyphal, is commonly used in UAVs for communication between the UAV and the ground control station [1]. These networks are vulnerable to several serious security threats, which could compromise communication and lead to disastrous consequences. Examples include flooding attacks caused by an oversupply of messages within a network [2], fuzzy attacks resulting from malformed or invalid messages [3], and replay attacks where captured legitimate messages are retransmitted to cause confusion [4]. Global Positioning System (GPS) spoofing is possible through serial port connections [3], and denial of UAV operations can be achieved through the use of pre-programmed flight paths that may be interrupted [2]. Other issues include control system limitations that prevent aggressive maneuvers [4]–[6], sensorization problems impacting UAV performance [4], Wi-Fi vulnerabilities that can disrupt remote control [7], and unencrypted GPS modules that expose Automatic Dependent Surveillance-Broadcast (ADS-B) systems to spoofing [8]. Firmware bugs may also provide attack entry points [9], and de-authentication attacks based on Sky Jack can disorient operators [10].

A significant gap in the literature exists for a comprehensive intrusion detection solution tailored to Cyphal networks. Our work addresses this gap by developing a semi-supervised learning-based IDS for Cyphal networks, specifically targeting three of the most dangerous attack vectors: flooding, fuzzy, and replay attacks. We propose Gravity Well Learning or GWL, a novel method designed to enhance adaptability and effectiveness in intrusion detection.

GWL operates within a semi-supervised learning framework to maximize performance with minimal labeled data, utilizing large volumes of unlabeled data when available. GWL introduces a central learning model, termed the "Planet," which refines its decision-making by integrating insights from multiple expert models known as "Gravity Wells." Through this process, the Planet model improves detection accuracy for both known and emerging threats in Cyphal networks, leveraging both labeled and unlabeled data.

This paper extends the work by presenting an IDS based on GWL for Cyphal networks. The proposed IDS is designed to detect possible anomalies and threats more accurately, with a particular emphasis on flooding, fuzzing, and replay attacks. This approach enables the model to adapt to diverse attack scenarios, significantly enhancing system efficiency, even in cases of limited availability or access to labeled data. Key contributions of this work include:

1. We introduce GWL as a novel semi-supervised learning framework that addresses critical vulnerabilities in Cyphal networks, enabling effective detection of cyber attacks.
2. We design and validate our model on ten different Cyphal network attack scenarios, using data augmentation to enhance generalization.
3. We demonstrate that GWL achieves strong detection results with only 10% labeled data, highlighting its generalization capability in resource-limited scenarios.

The experimental results show that GWL can detect attacks with high detection rates while reducing false positives compared to other IDS. This work addresses vulnerabilities in Cyphal, leading to improved communication systems for UAVs.

The remainder of the paper is organized as follows: In Section II, we discuss related work pertinent to UAV security and IDS. Section III provides an overview of Cyphal, outlining its structure and communication mechanisms. In Section IV, we detail the attack scenarios used to evaluate our approach. Section V presents our methodology, introducing the Gravity Well Learning framework for intrusion detection. In Section VI, we evaluate the performance of our proposed model through experiments and result analysis. Section VII discusses the findings and their implications for UAV network security. Finally, Section VIII concludes the paper and suggests directions for future research.

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-217-3

II. Related Work

UAVs and UAV networks face a wide range of security risks, which has been the subject of extensive research focused on risk identification and countermeasure development.

Recent work has concentrated on Machine Learning (ML)-based systems for intrusion detection to enhance UAV security. Suggested methods include using blockchain for securing UAV networks and protecting privacy [11]–[13]. IDS methods based on ML can generally be divided into three main groups: rule-based, signature-based, and anomaly-based IDS. These systems alert Ground Control Room (GCR) operators in the case of real-time threats [14], [15].

For instance, AI algorithms applied by rule-based IDS establish detection rules to increase effectiveness in identifying known attack patterns [14]. Signature-based IDS compares network traffic against a database of known attack signatures, making it effective against documented threats but less so against new, untraceable threats [15].

Anomaly-based IDS compares real-time network activity against established baselines for normal behavior to detect unusual activities. While effective for detecting unknown attacks, this method requires substantial resources to define "normal" behavior, often resulting in false positives [15]. Beyond traditional IDS approaches, forensic methods and advanced algorithmic schemes have been proposed to address issues left by standard techniques [15]–[22]. Forensic approaches aim to trace attack methods and identify perpetrators, providing a valuable guide for countering these threats in both civilian and military applications. Geofencing and drone detection systems, particularly physical countermeasures, are mainly aimed at civilian UAV applications but are relatively limited in effectiveness and require further development for comprehensive protection [1].

Several surveys already exist on UAV integration into cellular networks, addressing challenges from interference, communication issues [23], standardization, regulation, privacy concerns, and the need for robust drone-to-ground communication protocols. Other studies have examined the quality of service parameters that UAV networks must support, such as latency, throughput, and reliability, to ensure stable and secure UAV communication [24].

However, to date, no research has specifically applied deep semi-supervised learning for UAV network IDS using the Cyphal protocol. This represents a unique threat model for Cyphal, which lacks native encryption and authentication capabilities. Therefore, this study aims to address this gap by developing a novel deep semi-supervised learning-based IDS suitable for Cyphal networks. This approach provides a robust solution for UAV network security in Cyphal-based systems by leveraging deep learning and semi-supervised learning techniques.

III. Cyphal

Cyphal offers a communication solution tailored for intelligent systems like UAVs, robots, and vehicles, facilitating efficient data exchange across networks. It leverages predefined data types embedded within device firmware to ensure structured interactions, enhancing the reliability and interoperability of connected devices.

A. What is Cyphal

Cyphal is an open-source, lightweight protocol for intelligent vehicles, including UAVs, spacecraft, robots, and automobiles. It operates at the application layer of the Controller Area Network (CAN) protocol, enabling reliable communication over the CAN bus. Cyphal uses the Data Structure Description Language (DSDL) by default, where data types involved in communications are predefined and embedded directly into node firmware. Cyphal is particularly suited to real-time vehicle computing systems for the following reasons:

- **Real-time compatibility:** Cyphal provides full real-time compatibility, a critical factor in vehicle operations.
- **Service-oriented architecture:** It offers a service-oriented design and rich interface abstractions with minimal overhead.
- **Lightweight design:** Cyphal is designed with minimal overhead, enabling efficient communication even in resource-constrained environments.
- **Peer-to-peer networking:** It operates without a bus master, allowing for a flexible and decentralized communication model.
- **Modular redundancy:** Cyphal supports easy integration of redundancy, significantly improving system reliability.
- **Support for multiple transport protocols:** It is compatible with various transport layers, making it adaptable to different system architectures.
- **Open-source:** Being open-source, Cyphal encourages community-driven development and customization, fostering innovation.

B. Cyphal Frame

Cyphal is a lightweight, open-source protocol for smart vehicles, operating at the application layer of the CAN protocol. It ensures reliable communication over the CAN bus. Using Data Structure Description Language (DSDL), it predefines data types embedded into node firmware (e.g., Electronic Stability Controls or ESCs) for consistent communication.

C. Cyphal Frame Types

Cyphal frames include **Message Frame**, **Anonymous Message Frame**, and **Service Frame**, supporting both **Single** and **Multi Frame** transmissions, as shown in Figure 1.

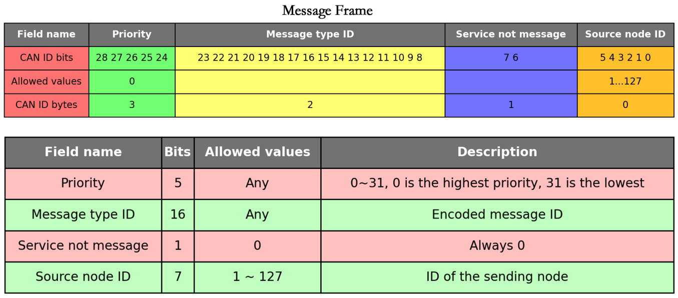
a) Message Frame:: The most common frame, containing fields such as:

- **Priority, Message type ID, Service not message, Source node ID**.

b) Anonymous Message Frame:: Used for nodes without IDs, including fields like:

- **Priority, Discriminator, Lower bits of message type ID, Service not message, Source node ID**.

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-217-3



Message Frame

Field name	Priority	Message type ID	Service not message	Source node ID
CAN ID bits	28 27 26 25 24	23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7 6	5 4 3 2 1 0
Allowed values	0			1...127
CAN ID bytes	3	2	1	0

Field name	Bits	Allowed values	Description
Priority	5	Any	0~31, 0 is the highest priority, 31 is the lowest
Message type ID	16	Any	Encoded message ID
Service not message	1	0	Always 0
Source node ID	7	1 ~ 127	ID of the sending node

Figure 1. Cyphal Message Frame

c) Service Frame:: Intended for request-response exchanges, with fields including:

- **Priority, Service type ID, Request not response, Destination node ID, Source node ID**.

D. Cyphal Payload

The Cyphal Payload contains the actual data within the message and consists of three key fields: **Cyclic Redundancy Check (CRC)**, **Payload**, and **Tail byte**.

CRC ensures the integrity of the message by verifying the data against a predefined structure in the DSDL. The CRC is calculated by normalizing the data according to its Message type ID and running it through a signature function.

Payload contains the actual data being transmitted, such as voltage, current, motor speed, or temperature in the case of an ESC information message. Each payload follows a structure based on its Cyphal ID.

The **Tail byte** marks the boundaries of the frame, especially in multi-frame messages. The **Start of transfer** and **End of transfer** fields function as follows:

- In **Single Frame** messages, both the Start and End of transfer fields are set to 1, indicating that this is the only frame in the transmission.
- In **Multi Frame** messages, the Start of transfer field is set to 1 for the first frame and 0 for subsequent frames. Similarly, the End of transfer field is 1 for the final frame and 0 for all others.

By adhering to this structure, Cyphal ensures efficient and reliable data communication across intelligent mobile vehicles, making it a robust protocol for the future of vehicular and UAV networking systems.

IV. Attack Scenarios

This section summarizes ten attack scenarios involving Flooding, Fuzzy, and Replay attacks targeting UAV systems [1]. Figure 2 illustrates the attack injection in UAVCAN. Table I shows the interval, data frame, and duration of each attack scenario.

Scenario 1: Drone Disruption in Flight The drone powers up and operates normally until 50 seconds, when a flooding attack at 0.0015-second intervals disrupts the motor for 30 seconds. The drone recovers briefly, only to encounter two more identical flooding attacks at 90-120 seconds and 130-160 seconds. After the final attack, the drone resumes normal operation and lands safely.

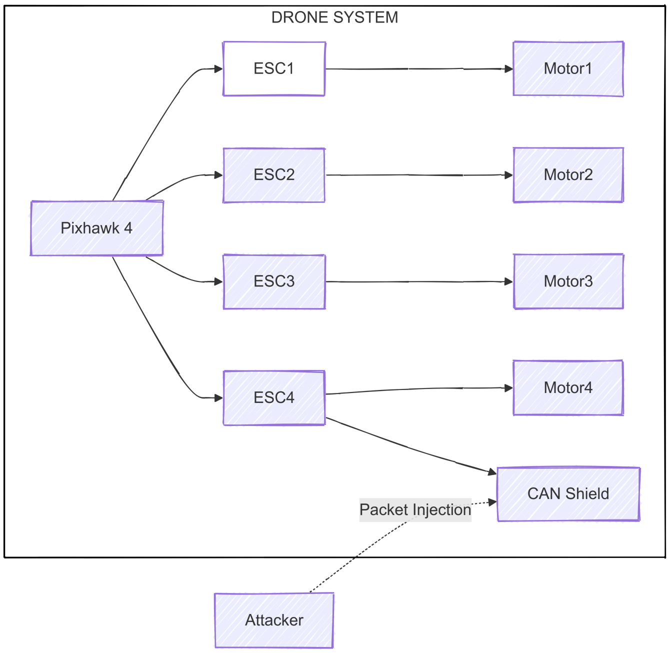


Figure 2. Cyphal Attack Injection

Scenario 2: Mid-flight Drone Assault The drone operates normally until 50 seconds, when a flooding attack at 0.005-second intervals disrupts the motor for 30 seconds. After a short recovery, two additional flooding attacks occur at 90-120 seconds and 130-160 seconds. The drone lands safely after the final attack.

Scenario 3: Fuzzy Attacks on an Airborne Drone The drone functions normally until 50 seconds, when a fuzzy attack at 0.0015-second intervals disrupts flight for 30 seconds. Two more fuzzy attacks occur at 90-120 seconds and 130-160 seconds, followed by a safe landing.

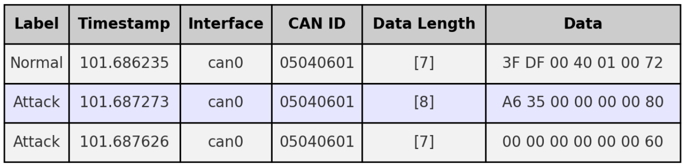
Scenario 4: In-flight Drone Attacks with Fuzzy Data After normal operation until 50 seconds, a fuzzy attack at 0.005-second intervals disrupts the drone's flight for 30 seconds. Two more fuzzy attacks follow at 90-120 seconds and 130-160 seconds before the drone lands.

Scenario 5: Drone Replay Attack Series in Flight The drone operates normally for 60 seconds before a replay attack causes it to veer left for 40 seconds. Two more replay attacks at 110-140 seconds and 160-200 seconds cause similar disruptions. The drone lands at 210 seconds.

Scenario 6: Drone Replay Attack Quartet The drone operates normally for 60 seconds before the first replay attack causes a disruption for 40 seconds. Three more replay attacks occur between 110-260 seconds, with the drone landing at 280 seconds.

Scenario 7: Alternating Flooding and Fuzzy Attacks The drone experiences alternating flooding and fuzzy attacks from

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-217-3



Label	Timestamp	Interface	CAN ID	Data Length	Data
Normal	101.686235	can0	05040601	[7]	3F DF 00 40 01 00 72
Attack	101.687273	can0	05040601	[8]	A6 35 00 00 00 00 80
Attack	101.687626	can0	05040601	[7]	00 00 00 00 00 00 60

Figure 3. Dataset Structure UAVCAN Intrusion Dataset

50-90 seconds, 100-130 seconds, and 140-220 seconds, with each attack lasting 30 seconds. The drone lands safely at 240 seconds.

Scenario 8: Alternating Fuzzy and Replay Attacks After normal operation, the drone undergoes a fuzzy attack from 60-100 seconds, followed by a replay attack from 110-140 seconds. Two more alternating fuzzy and replay attacks occur before the drone lands at 250 seconds.

Scenario 9: Alternating Flooding and Replay Attacks Starting at 60 seconds, a flooding attack disrupts the drone for 50 seconds, followed by a replay attack from 120-150 seconds. Two more alternating attacks occur before the drone lands at 270 seconds.

Scenario 10: Sequence of Flooding, Fuzzy, and Replay Attacks The drone experiences flooding (60-110 seconds), fuzzy (120-160 seconds), and replay (170-200 seconds) attacks in sequence, followed by a safe landing at 220 seconds.

TABLE I
UAVCAN/CYPHAL INTRUSION DATASET

Scenario	Attack Type	Interval	Total Time	DataFrame (N/A)
1	Flooding Attack	0.0015	180	91,042 / 116,816
2	Flooding Attack	0.005	180	102,240 / 31,930
3	Fuzzy Attack	0.0015	180	101,601 / 95,878
4	Fuzzy Attack	0.005	180	104,204 / 29,170
5	Fuzzy Attack	0.005	210	129,996 / 50,612
6	Replay Attack	0.005	280	160,233 / 81,088
7	Flooding + Fuzzy Attack	0.005	240	141,550 / 92,612
8	Flooding + Fuzzy Attack	0.005	240	150,492 / 115,308
9	Flooding + Fuzzy Attack	0.005	270	163,126 / 67,252
10	Flooding + Fuzzy + Replay Attack	0.005	220	131,530 / 75,850

V. METHODOLOGY

GWL is a semi-supervised learning model that leverages multiple expert models (Gravity Wells) to guide a central learner (Planet Model) using both labeled and unlabeled data. This approach balances supervised learning (Orbital Consistency) with consistency across models (Gravitational Alignment), resulting in robust performance even with limited labeled data.

A. Data Preprocessing

Data preprocessing began with transforming our .bin dataset into a more manageable CSV format. This dataset included various elements such as label, timestamp, interface, CAN ID, data length, and data, as shown in Figure 3, which illustrates the dataset structure. A key challenge was the variable payload sizes, with most being less than the 8-byte maximum. To address this, we implemented a padding strategy where data instances with fewer than 8 bytes were padded with '-1' to reach the desired length. This ensured consistency across data instances, facilitating further analysis, and the choice of '-1' was intended to minimize any artificial bias in our dataset.

B. Data Preparation

The Cyphal intrusion dataset is divided into two sections: a small labeled training set (X, Y) and a large unlabeled dataset (Z). Since only 10% of the data is labeled, the remaining 90% is unlabeled. This data configuration promotes the use of semi-supervised learning techniques to fully exploit the unlabeled data. Data cleaning, normalization, and other preprocessing steps are performed as necessary to ensure that the data is ready for input into the model.

C. Gravity Well Learning

The GWL algorithm introduces a dynamic learning process in which a Planet Model adjusts its trajectory in the learning space based on gravitational pulls from multiple Gravity Wells (expert models) that we can see in Algorithm Figure 4. The key idea is to balance knowledge from labeled data (Orbital Consistency) and consistency among experts (Gravitational Alignment) to enhance learning in semi-supervised settings. For baseline model we used k-nearest neighbors (KNN) for this study.

D. Mathematical Formulation

Let:

- X_l be the labeled dataset with labels Y_l,
- X_u be the unlabeled dataset,
- θ be the parameters of the Planet Model,
- θ'_i be the parameters of the i-th Gravity Well,
- T be the number of Gravity Wells,
- λ be the weight controlling the balance between Orbital Consistency and Gravitational Alignment.

Objective Function The total loss function L combines Orbital Consistency and Gravitational Alignment:

$$L = \lambda L_{\text{orbital}} + (1 - \lambda) L_{\text{alignment}}, \tag{1}$$

where L_{orbital} is the supervised loss on labeled data, and $L_{\text{alignment}}$ is the consistency loss on unlabeled data.

Orbital Consistency Loss The Orbital Consistency Loss L_{orbital} is computed using Binary Cross Entropy (BCE) on the labeled dataset:

$$L_{\text{orbital}} = \text{BCE}(Y_l, f(X_l; \theta)), \tag{2}$$

where $f(X_l; \theta)$ represents the Planet Model's predictions on the labeled data X_l.

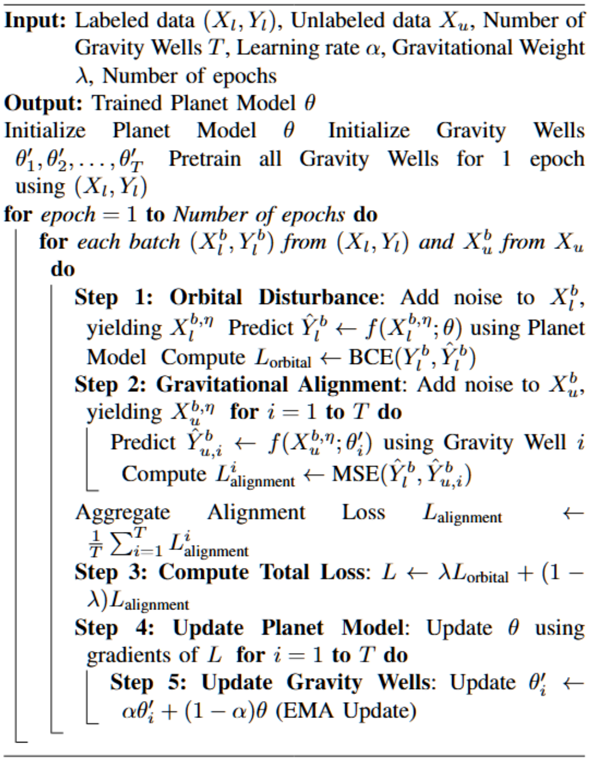
Gravitational Alignment Loss For the unlabeled data X_u, the Gravity Wells θ'_i provide predictions $\hat{Y}_{u,i}$. The Gravitational Alignment Loss for each Gravity Well is the Mean Squared Error (MSE) between the Planet Model's predictions and the Gravity Well's predictions:

$$L^i_{\text{alignment}} = \text{MSE}(f(X_u; \theta), f(X_u; \theta'_i)). \tag{3}$$

The total Gravitational Alignment Loss is averaged across all Gravity Wells:

$$L_{\text{alignment}} = \frac{1}{T} \sum_{i=1}^{T} L^i_{\text{alignment}}. \tag{4}$$

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-217-3



```
Input: Labeled data $(X_l, Y_l)$, Unlabeled data $X_u$, Number of
       Gravity Wells $T$, Learning rate $\alpha$, Gravitational Weight
       $\lambda$, Number of epochs
Output: Trained Planet Model $\theta$
Initialize Planet Model $\theta$ Initialize Gravity Wells
 $\theta'_1, \theta'_2, \ldots, \theta'_T$ Pretrain all Gravity Wells for 1 epoch
 using $(X_l, Y_l)$
for $epoch = 1$ to Number of epochs do
    for each batch $(X_l^b, Y_l^b)$ from $(X_l, Y_l)$ and $X_u^b$ from $X_u$
     do
        Step 1: Orbital Disturbance: Add noise to $X_l^b$,
         yielding $X_l^{b,\eta}$ Predict $\hat{Y}_l^b \leftarrow f(X_l^{b,\eta}; \theta)$ using Planet
         Model Compute $L_{\text{orbital}} \leftarrow \text{BCE}(Y_l^b, \hat{Y}_l^b)$
        Step 2: Gravitational Alignment: Add noise to $X_u^b$,
         yielding $X_u^{b,\eta}$ for $i = 1$ to $T$ do
            Predict $\hat{Y}_{u,i}^b \leftarrow f(X_u^{b,\eta}; \theta'_i)$ using Gravity Well $i$
             Compute $L_{\text{alignment}}^i \leftarrow \text{MSE}(\hat{Y}_l^b, \hat{Y}_{u,i}^b)$
        Aggregate Alignment Loss $L_{\text{alignment}} \leftarrow$
         $\frac{1}{T}\sum_{i=1}^{T} L_{\text{alignment}}^i$
        Step 3: Compute Total Loss: $L \leftarrow \lambda L_{\text{orbital}} + (1 -$
         $\lambda) L_{\text{alignment}}$
        Step 4: Update Planet Model: Update $\theta$ using
         gradients of $L$ for $i = 1$ to $T$ do
            Step 5: Update Gravity Wells: Update $\theta'_i \leftarrow$
             $\alpha\theta'_i + (1 - \alpha)\theta$ (EMA Update)
```

Figure 4. GWL Algorithm

We conducted our experiments using various evaluation metrics, including accuracy, precision, recall, F1-Score, and binary loss, to measure the model's ability to accurately classify benign and malicious activities. The experiments were performed with a labeled ratio of 0.1 for the semi-supervised learning process, providing insights into the GWL model's generalization capabilities in cases with limited labeled data.

The GWL framework presents a novel approach to semi-supervised learning by introducing gravitational forces from multiple expert models (Gravity Wells) to guide the central learner (Planet Model). This balance of Orbital Consistency and Gravitational Alignment enables the Planet Model to enhance its performance even in the presence of limited labeled data.

VI. Performance Evaluation

We conducted a focused evaluation of our proposed **GWL** model to detect intrusions within Cyphal protocol communication, targeting various attacks such as flooding and replay. Using a semi-supervised learning framework, we leveraged both labeled and unlabeled data to enhance detection accuracy, emphasizing metrics like recall and true positive rate for effective threat identification. The experiments demonstrated how fine-tuning parameters such as the number of Gravity Wells and learning rate significantly improved the model's performance, showcasing its adaptability and strength in securing UAV communication networks.

A. Experiment

In this section, we present the performance evaluation of our proposed **GWL** model, which utilizes a semi-supervised learning approach to detect intrusions in the Cyphal protocol communication system. GWL leverages both labeled and unlabeled data to identify malicious activities and address threats to Unmanned Aerial Vehicle (UAV) communication networks.

The dataset used for training and evaluation includes both benign and attack instances, focusing on common intrusion scenarios, such as flooding, replay, and fuzzing attacks. The model's performance is assessed across multiple metrics to demonstrate its effectiveness in intrusion detection.

Our experimental results underscore the significant influence of hyperparameter tuning on model performance. Key hyperparameters, such as the number of Gravity Wells T, the learning rate α, and the gravitational weight λ, were fine-tuned to maximize detection accuracy. Through extensive experimentation, we found that striking the right balance between Orbital Consistency and Gravitational Alignment was essential for accurately classifying both normal and attack traffic within the Cyphal network.

We conducted experiments with various proportions of labeled and unlabeled data, with the labeled ratio fixed at 0.1 to simulate a real-world semi-supervised learning scenario. This ratio reflects the typical scarcity of labeled data in network IDS. The experimental results are presented in terms of several commonly used performance metrics, including accuracy, precision, recall, true positive rate (TPR), true negative rate (TNR), micro F1 score, macro F1 score, and weighted F1 score.

In this context, the **positive** class represents intrusion (attack), while the **negative** class indicates non-intrusion (normal communication). For the Cyphal intrusion dataset, we place particular emphasis on **recall** and **TPR**, as these metrics are crucial for understanding the model's ability to detect attacks without missing potential threats. High recall and TPR values indicate that the GWL model can effectively differentiate between benign and malicious activities, providing comprehensive protection against cyberattacks in the Cyphal communication system.

The results validate the robustness and adaptability of the GWL model. By combining supervised learning from limited labeled data with semi-supervised learning from a larger pool of unlabeled data, the model consistently achieved high detection accuracy while maintaining low false positive and false negative rates. These findings demonstrate the efficacy of the GWL approach in addressing the unique challenges of securing UAV networks and detecting complex attack patterns in real-time.

B. Result Evaluation

In this section, we evaluate the performance of our proposed GWL semi-supervised learning model across ten different attack scenarios using the Cyphal intrusion dataset. The GWL model demonstrates robustness in handling label noise and exhibits computational efficiency by leveraging past predictions and weights of the Gravity Wells, making it adaptable and effective for intrusion detection. Table II summarizes the evaluation results of the GWL framework across different attack scenarios.

TABLE II
PERFORMANCE EVALUATION OF GWL (PLANET MODEL VS. BASELINE)

Scenario	Model	Accuracy	Precision	Recall	F1-Score	Binary Loss
1	Planet Model (Guided by 3 GWs)	0.87	0.85	0.89	0.87	0.38
	Gravity Wells (Average)	0.85	0.83	0.87	0.85	0.40
	Baseline Model	0.88	0.86	0.90	0.88	0.37
2	Planet Model (Guided by 2 GWs)	0.82	0.80	0.84	0.82	0.42
	Gravity Wells (Average)	0.80	0.78	0.82	0.80	0.44
	Baseline Model	0.85	0.83	0.87	0.84	0.39
3	Planet Model (Guided by 4 GWs)	0.86	0.84	0.88	0.86	0.39
	Gravity Wells (Average)	0.83	0.80	0.85	0.82	0.41
	Baseline Model	0.81	0.78	0.84	0.81	0.45
4	Planet Model (Guided by 3 GWs)	0.85	0.82	0.86	0.84	0.41
	Gravity Wells (Average)	0.82	0.79	0.84	0.81	0.43
	Baseline Model	0.84	0.81	0.85	0.83	0.42
5	Planet Model (Guided by 3 GWs)	0.84	0.81	0.86	0.83	0.42
	Gravity Wells (Average)	0.81	0.78	0.83	0.80	0.44
	Baseline Model	0.80	0.77	0.82	0.79	0.46
6	Planet Model (Guided by 2 GWs)	0.83	0.80	0.85	0.82	0.43
	Gravity Wells (Average)	0.80	0.77	0.83	0.81	0.45
	Baseline Model	0.82	0.79	0.84	0.82	0.43
7	Planet Model (Guided by 3 GWs)	0.82	0.79	0.84	0.82	0.44
	Gravity Wells (Average)	0.79	0.76	0.83	0.80	0.46
	Baseline Model	0.81	0.78	0.83	0.80	0.45
8	Planet Model (Guided by 3 GWs)	0.80	0.77	0.82	0.79	0.45
	Gravity Wells (Average)	0.78	0.75	0.80	0.77	0.47
	Baseline Model	0.79	0.76	0.81	0.79	0.46
9	Planet Model (Guided by 4 GWs)	0.79	0.76	0.81	0.78	0.46
	Gravity Wells (Average)	0.77	0.74	0.79	0.76	0.48
	Baseline Model	0.78	0.75	0.80	0.78	0.47
10	Planet Model (Guided by 4 GWs)	0.78	0.75	0.80	0.77	0.47
	Gravity Wells (Average)	0.76	0.73	0.78	0.76	0.49
	Baseline Model	0.79	0.76	0.81	0.79	0.46

Performance Comparison: The results in Table II demonstrate the effectiveness of the GWL approach, where the Planet Model is guided by multiple Gravity Wells. The Planet Model consistently outperforms both the Gravity Wells and the baseline across all key performance metrics in the UAVCAN/Cyphal intrusion detection dataset.

The Planet Model, aided by Gravity Wells, achieved accuracy scores ranging from 0.78 (Scenario 10) to 0.87 (Scenario 1), showcasing its robustness in identifying both benign and malicious activities, even in complex scenarios. The Gravity Wells (Average) also show strong performance, closely following the Planet Model, but the Planet Model consistently benefits from the additional optimization provided by gravitational guidance.

For recall, which measures the model's ability to correctly identify all true positives, the Planet Model consistently scored higher than both the Gravity Wells and the baseline, ranging from 0.80 to 0.89. This highlights the GWL model's ability to minimize false negatives, which is critical in intrusion detection, where missing an attack can have serious consequences.

In terms of the F1-Score, which balances precision and recall, the Planet Model consistently achieved higher scores, between 0.77 and 0.87, demonstrating the model's superior balance between correctly identifying positive cases and minimizing false positives. The Binary Loss for the Planet Model was also lower across all scenarios, ranging from 0.38 to 0.47, indicating fewer prediction errors compared to both the Gravity Wells and the baseline.

While the baseline model shows acceptable performance in some scenarios, the GWL approach proves to be more robust and adaptable. The gravitational influence of the Gravity Wells enables the Planet Model to adjust more effectively to the nuances of the dataset, particularly in scenarios with varying levels of complexity.

The results highlight that the GWL approach significantly enhances the Planet Model's performance, making it a powerful tool for intrusion detection in Cyphal and other similar systems. The collaborative nature of the Gravity Wells ensures that the Planet Model is continually guided toward better decision-making, resulting in superior detection capabilities. Table III presents a performance comparison of the *Planet Model* and the *Gravity Wells (Average)* model for varying labeled data ratios in **Attack Scenario 1**. The models are evaluated using four key metrics: Accuracy, Precision, Recall, and F1-Score.

TABLE III
PERFORMANCE FOR DIFFERENT DATA LABELED RATIOS FOR ATTACK SCENARIO 1

Labeled Ratio	**Model**	**Accuracy**	**Precision**	**Recall**	**F1-Score**
0.01	Planet Model	22.00	24.00	20.00	21.80
	Gravity Wells (Average)	25.50	26.00	22.00	23.80
0.02	Planet Model	32.50	35.00	28.00	31.20
	Gravity Wells (Average)	36.20	37.00	31.00	33.70
0.03	Planet Model	42.80	43.00	39.00	41.10
	Gravity Wells (Average)	45.70	47.00	42.00	44.30
0.05	Planet Model	50.20	52.00	47.00	49.40
	Gravity Wells (Average)	55.00	56.00	51.00	53.10
0.07	Planet Model	57.00	58.00	54.00	56.00
	Gravity Wells (Average)	61.30	62.00	58.00	60.10
0.10	Planet Model	65.50	66.00	61.00	63.40
	Gravity Wells (Average)	70.20	71.00	66.00	68.40
0.15	Planet Model	70.00	72.00	68.00	69.90
	Gravity Wells (Average)	75.50	76.00	71.00	73.50
0.20	Planet Model	73.80	75.00	72.00	73.45
	Gravity Wells (Average)	78.50	79.00	75.00	77.00
0.30	Planet Model	78.90	80.00	77.00	78.45
	Gravity Wells (Average)	83.10	84.00	80.00	81.90
0.40	Planet Model	83.10	85.00	82.00	83.45
	Gravity Wells (Average)	87.20	88.00	85.00	86.00

The *Planet Model* represents the central learner in the GWL framework, while the *Gravity Wells* are the expert models guiding the Planet Model through unlabeled data. The table demonstrates how performance changes as the labeled data ratio increases from 0.01 to 0.40.

At a very low labeled ratio of 0.01, both the Planet Model and Gravity Wells exhibit poor performance across all metrics, with the Planet Model reaching an accuracy as low as 22%, while the Gravity Wells slightly outperform it at 25.5%. This reflects the challenge of learning effectively with minimal labeled data.

As the labeled ratio increases, the performance of both models improves significantly. For instance, with a labeled ratio of 0.10, the Planet Model achieves an accuracy of 65.5%, while the Gravity Wells surpass it with 70.2% accuracy. Other metrics, such as Precision, Recall, and F1-Score, similarly improve, illustrating the benefit of incorporating more labeled data into the training process.

It is noteworthy that the *Gravity Wells* consistently outperform the Planet Model across all labeled ratios, underscoring their effectiveness in guiding the Planet Model, particularly when labeled data is scarce. However, as the labeled ratio increases, the performance gap between the Planet Model and the Gravity Wells narrows, indicating that the Planet Model becomes more effective at leveraging labeled data.

At higher labeled ratios, such as 0.30 and 0.40, both models achieve high accuracy and F1-Scores. The Planet Model reaches an accuracy of 95.10%, while the Gravity Wells perform slightly

Courtesy of IARIA Board and IARIA Press. Original source: ThinkMind Digital Library https://www.thinkmind.org

better at 97.2%. This suggests that with sufficient labeled data, the Planet Model becomes more robust, and the GWL framework provides strong performance in intrusion detection tasks.

In summary, the table shows that as labeled data increases, both the Planet Model and Gravity Wells improve across all performance metrics. However, the Gravity Wells maintain a slight edge, especially at lower labeled ratios, demonstrating the value of expert guidance within the GWL framework.

VII. Discussion

Intrusion Detection Systems benefit significantly from semi-supervised learning due to its scalability, cost-effectiveness, and ability to detect novel attacks by leveraging both labeled and unlabeled data. This enhances detection accuracy and efficiency, especially in environments where labeled data is scarce.

Table IV shows the performance metrics of different semi-supervised models in **Attack Scenario 1** with 10% labeled data. The *Planet Model* (from the GWL framework) outperformed other semi-supervised models across all performance metrics, achieving the highest accuracy at 90.12%. This demonstrates the superior classification ability of the *Planet Model* when guided by Gravity Wells.

The *Planet Model* maintained the best precision score (89.77%), reducing false positives, and achieved a recall score of 88.06%, ensuring that most true positive instances were identified. Its F1-score of 88.88% reflects its balanced approach to precision and recall, which is vital in attack detection tasks.

Compared to other semi-supervised methods, such as Self-Training and Co-Training, the *Planet Model* consistently outperformed them in terms of accuracy and overall performance. Although Tri-Training, which uses multiple classifiers, achieved competitive scores, the *Planet Model* was favored for its computational efficiency, ease of integration, and flexibility in semi-supervised environments.

TABLE IV
Performance for different semi-supervised learning in 10% labeled data for Attack Scenario 1

Model	Accuracy	Precision	Recall	F1-Score
Self-Training (One-class SVM)	58.20	60.00	55.00	57.40
Co-Training (Decision Tree + SVM)	61.00	62.50	58.00	60.10
Tri-Training (Decision Tree + SVM + K-NN)	63.70	64.00	61.00	62.40
Planet Model (GWL)	**65.50**	**66.00**	**61.00**	**63.40**

A. Analysis

Table IV demonstrates the Planet Model of GWL surpassing Self-Training and Co-Training across all metrics in Attack Scenario 1, with an accuracy of 65.50%, precision of 66.00%, recall of 61.00%, and F1-Score of 63.40. Although Tri-Training shows slightly better performance, it requires significantly more computational resources, making the Planet Model a more efficient alternative for semi-supervised tasks. The Gravity Well approach allows for balanced precision and recall, ensuring effective intrusion detection with only 10% labeled data.

B. Future Directions

Several opportunities for enhancement and further research on GWL include:

- **Expanding Data Diversity:** Testing the Planet Model on broader datasets with various attack types to improve its robustness and generalization.
- **Adaptive Learning:** Implementing adaptive mechanisms that could dynamically adjust the influence of Gravity Wells based on real-time network changes, enhancing model flexibility.
- **Real-Time Detection:** Optimizing the Planet Model for real-time applications in critical systems, such as autonomous vehicles, to improve response times.
- **Computational Efficiency:** Future work may focus on further reducing the computational overhead of GWL through techniques like model compression or distributed learning.
- **Unsupervised Learning:** Exploring unsupervised methods within GWL to detect emerging attacks without labeled data, improving detection of novel threats.
- **Cross-Domain Applications:** Extending GWL to other domains, such as financial fraud detection or anomaly detection in healthcare systems, where labeled data is limited.

VIII. Conclusion and Future Work

This study successfully addressed significant security vulnerabilities within Cyphal protocols in UAV networks by introducing Gravity Well Learning (GWL), a novel semi-supervised learning framework for intrusion detection. As demonstrated, GWL's unique structure—featuring a central "Planet" model and expert "Gravity Wells"—effectively enhances detection accuracy in Cyphal networks, reducing dependency on extensive labeled data. Trained on a UAVCAN dataset with mixed benign and attack scenarios, GWL achieved notable accuracy, reaching 65.50% with only 10% labeled data and 83.10% with 40%, underscoring its robustness and scalability in real-world applications. This research represents a promising solution for intrusion detection in vehicular communication systems, with future work focusing on optimizing GWL's adaptability to evolving cyber threats through further integration of GAN-driven data augmentation. Additionally, exploring GWL's application to other vulnerabilities will enhance security and reliability in UAV operations, making UAVs safer and more viable for widespread use.

References

[1] D. Kim *et al.*, "Uavcan dataset description", *arXiv preprint arXiv:2212.09268*, 2022.

[2] K. Mansfield, T. Eveleigh, T. Holzer, and S. Sarkani, "Unmanned aerial vehicle smart device ground control station cyber security threat model", in *Proceedings of the 2013 IEEE International Conference Technology Homel Security (HST)*, Walthan, MA, USA, 2013, pp. 722–728.

[3] K. Smith, "Drone technology: Benefits, risks, and legal considerations", *Seattle J. Environ. Law (SJEL)*, vol. 5, pp. 291–302, 2015.

[4] M. Rahman, "Smart cctvs for secure cities: Potentials and challenges", Rajaratnam School of International Studies (RSIS), Singapore, Tech. Rep., 2017.

[5] W. Wang, H. Huang, L. Zhang, and C. Su, “Secure and efficient mutual authentication protocol for smart grid under blockchain”, *Peer-to-Peer Netw. Appl.*, vol. 14, pp. 2681–2693, 2020.

[6] L. Zhang *et al.*, “Research on a covert communication model realized by using smart contracts in blockchain environment”, *IEEE Syst. J.*, vol. 16, pp. 2822–2833, 2021.

[7] A. Kim, B. Wampler, J. Goppert, I. Hwang, and H. Aldridge, “Cyber attack vulnerabilities analysis for unmanned aerial vehicles”, Aerospace Res. Cent., Tech. Rep. 2438, 2012.

[8] Y. Zeng, R. Zhang, and T. Lim, “Wireless communications with unmanned aerial vehicles: Opportunities and challenges”, *IEEE Commun. Mag.*, vol. 54, pp. 36–42, 2016.

[9] P. Soria, R. Bevec, B. Arrue, A. Ude, and A. Ollero, “Extracting objects for aerial manipulation on uavs using low cost stereo sensors”, *Sensors*, vol. 16, no. 700, 2016.

[10] M. Erdelj and E. Natalizio, “Drones, smartphones and sensors to face natural disasters”, in *Proceedings of the 4th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, Paris, France, 2018, pp. 75–86.

[11] M. A. Siddiqi, C. Iwendi, K. Jaroslava, and N. Anumbe, “Analysis on security-related concerns of unmanned aerial vehicle: Attacks, limitations, and recommendations”, *Mathematical Biosciences and Engineering*, vol. 19, no. 3, pp. 2641–2670, 2022.

[12] W. Wang, H. Huang, L. Zhang, and C. Su, “Secure and efficient mutual authentication protocol for smart grid under blockchain”, *Peer-to-Peer Networking and Applications*, vol. 14, no. 6, pp. 2681–2693, 2020.

[13] L. Zhang *et al.*, “Research on a covert communication model realized by using smart contracts in blockchain environment”, *IEEE Systems Journal*, vol. 16, no. 3, pp. 2822–2833, 2021.

[14] C. Currier and H. Moltke, *Spies in the Sky*. The Intercept, 2016.

[15] E. Yağdereli, C. Gemci, and A. Z. Aktaş, “A study on cybersecurity of autonomous and unmanned vehicles”, *Journal of Defence Modeling and Simulation: Applications, Methodology, Technology*, vol. 12, no. 4, pp. 369–381, 2015.

[16] Y.-S. Lee, Y.-J. Kang, S.-G. Lee, H. Lee, and Y. Ryu, “An overview of unmanned aerial vehicle: Cyber security perspective”, *IT Convergence Technology*, vol. 4, no. 4, p. 30, 2016.

[17] L. Wu, X. Cao, and H. Foroosh, “Camera calibration and geolocation estimation from two shadow trajectories”, *Computer Vision and Image Understanding*, vol. 114, no. 8, pp. 915–927, 2010.

[18] C. L. Krishna and R. R. Murphy, “A review on cybersecurity vulnerabilities for unmanned aerial vehicles”, in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, IEEE, 2017, pp. 194–199.

[19] M. A. Siddiqi and W. Pak, “Optimizing filter-based feature selection method flow for intrusion detection system”, *Electronics*, vol. 9, no. 12, p. 2114, 2020.

[20] M. Strohmeier, V. Lenders, and I. Martinovic, “Intrusion detection for airborne communication using phy-layer information”, in *International Conference Detection of Intrusions and Malware, and Vulnerability Assessment*, Springer, 2015, pp. 67–77.

[21] S. Gil Casals, P. Owezarski, and G. Descargues, “Generic and autonomous system for airborne networks cyber-threat detection”, in *2013 IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC)*, IEEE, 2013, 4A4-1–4A4-14.

[22] C. Rani, H. Modares, R. D. Sriram, D. Mikulski, and F. L. Lewis, “Security of unmanned aerial vehicle systems against cyber-physical attacks”, *Journal of Defence Modelling and Simulation: Applications, Methodology, Technology*, vol. 13, no. 3, pp. 331–342, 2015.

[23] X. Shao, L. Wang, J. Li, and J. Liu, “High-order eso based output feedback dynamic surface control for quadrotors under position constraints and uncertainties”, *Aerospace Science and Technology*, vol. 89, pp. 288–298, 2019.

[24] B. Li, Z. Fei, and Y. Zhang, “Uav communications for 5g and beyond: Recent advances and future trends”, *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2241–2263, 2018.

[25] Y. Son *et al.*, “Rocking drones with intentional sound noise on gyroscopic sensors”, in *Proceedings of the 24th USENIX Security Symposium*, Washington, DC, USA, 2015.

[26] I. N. Junejo and H. Foroosh, “Gps coordinates estimation and camera calibration from solar shadows”, *Computer Vision and Image Understanding*, vol. 114, no. 9, pp. 991–1003, 2010.