Reasoning and Detecting Collaborative Attacks in Autonomous UAV Networks

Bharat Bhargava

Growing Autonomous UAV Network Market

- Autonomous UAV networks are gaining increasing prevalence.
  - The UAV market will reach 100 billion USD by 2030. [1]
- Their growing prevalence demands robust security, especially against collaborative attacks.

Autonomous UAV Network Security

- Inherit fundamental security vulnerabilities from traditional autonomous networks, while facing unique challenges
  - Wireless nature
  - Highly dynamic topology
  - Rapid node mobility
  - Communication disruptions and packet losses
- Thus, it is important to develop robust security mechanisms for autonomous UAV networks.
- These characteristics make them more vulnerable to collaborative attacks.

Collaborative Attack Definition

- Multiple adversaries coordinate and synchronize their actions to achieve disruption, deception, or usurpation of the target network.
- It is a threat that will be exacerbated by the increasing availability of commercial off-the-shelf (COTS) UAV platforms.
- Implications
  - Attackers can simultaneously compromise multiple UAVs within a network to wage attacks that exceed the capabilities of individual adversaries, such as co-ordinated jamming combined with false data injection.
  - Multiple attackers may employ different attack vectors concurrently, forcing the network to defend against diverse threats simultaneously.

Collaborative Attacks

- What are Collaborative Attacks?
  - DDoS Attacks [2]
  - Coordinated Eavesdropping [3]
- Why are they Important?
  - They are more harmful to the network
    - Amplifying Effect
    - Shortcut Effect
  - They are harder to detect and defend
    - Hiding Effect
  - The above effects are dubbed *synergy effects (SEs)*

Proposed Tasks

- Task 1: Defining Collaborative Attacks against Autonomous UAV Networks
  - Develops formal system and threat models for collaborative attacks against autonomous UAV networks, while establishing experimental validation approaches.
  - Three subtasks include:
    - characterizing system models that capture the features and resource constraints of autonomous UAV networks,
    - formulating threat models that define the capabilities and coordination patterns of collaborative attackers, and
    - developing comprehensive approaches for experimental validation of both attacks and defenses.

Proposed Tasks

- Task 2: Designing Mechanisms to Secure Autonomous UAV Networks against Collaborative Attacks
  - Resource-constrained UAVs require a lightweight, onboard filtering mechanism to handle high-volume traffic loads, for which we employ lightweight machine learning (ML) models as the initial detection layer.
  - The filtered alerts are then processed with two complementary tracks: an instant analysis track and a long-term analysis track.
    - Instant track: alerts trigger security verification against defined attack models using model checking technology to identify immediate threats.
    - Long-term track: employs sequential and attention-based ML techniques to uncover complex temporal dependencies between alerts that may indicate coordinated adversarial behavior.

Proposed Tasks

- Task 2: Designing Mechanisms to Secure Autonomous UAV Networks against Collaborative Attacks (continued)
  - The discovered long-term attack patterns are then fed back into the model checker for pattern analysis.
    - This creates a closed-loop system where model checking discovers attack patterns that guide the refinement of ML models

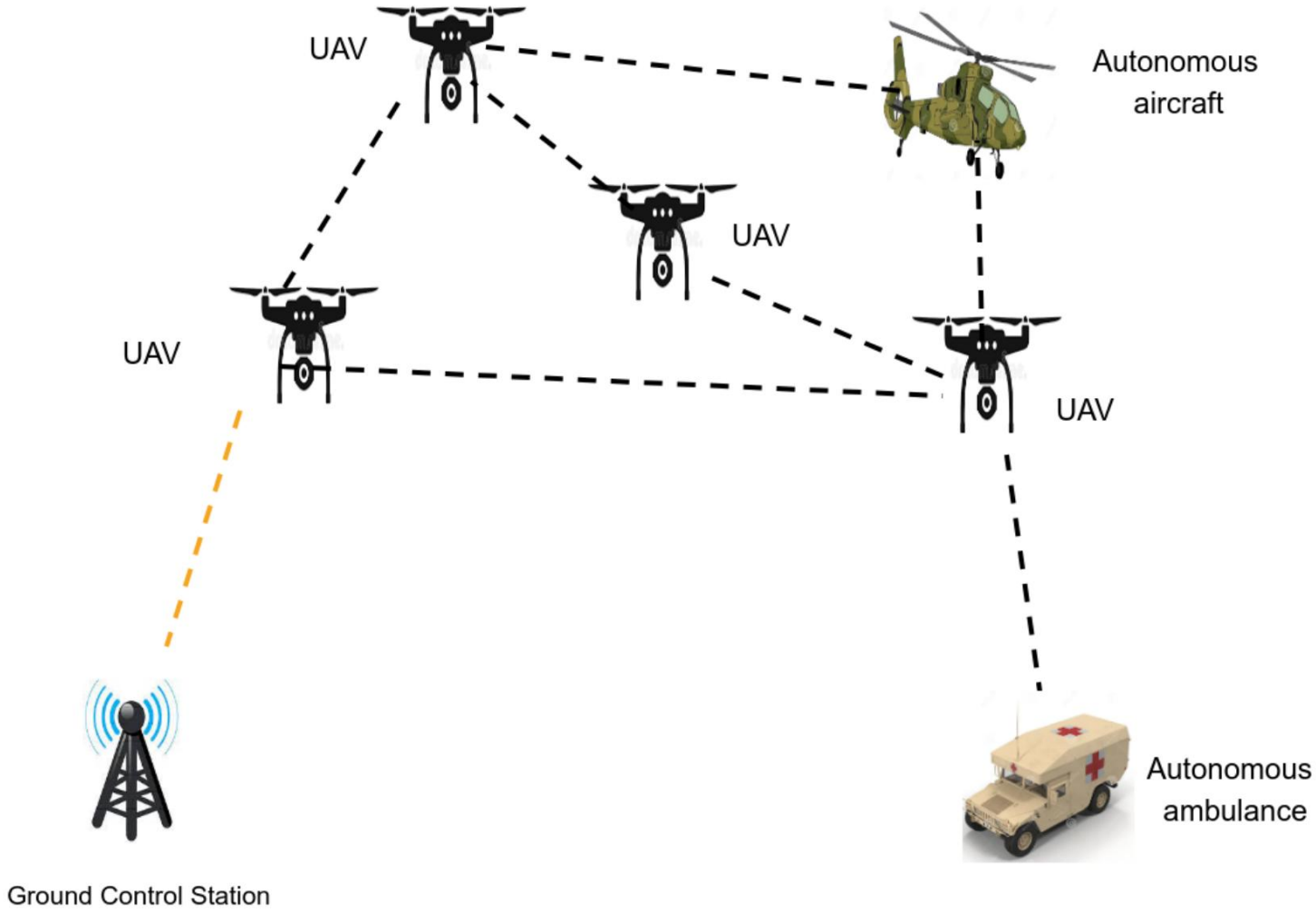
Proposed Tasks

- Task 3: Designing Defense Framework to Counter Collaborative Attacks
  - We exemplify the approach through a framework designed for a centralized autonomous UAV network topology.
  - It serves as a baseline example — through this project, the research will explore framework variations including decentralized architectures, hybrid networks with ground vehicles, and other topological configurations.
  - The modular design ensures the research not only validates the core detection mechanisms but also establishes foundations for adapting the framework to diverse autonomous UAV network deployments.

Task 1: Developing System Model of Autonomous UAV Networks

- Autonomous UAV networks also include ground vehicles and other autonomous aircraft (see figure in next slide).
- The network can operate in various configurations:
  - networks with and without ground vehicle nodes,
  - fully autonomous versus human-supervised operations, and
  - centralized versus decentralized control structures.
- Each configuration presents distinct security implications and requires tailored defensive approaches.
- Furthermore, we should consider dynamic network membership, where nodes can join or leave during task.

An Autonomous UAV Networks with Ground Vehicles and Other Autonomous Aircrafts



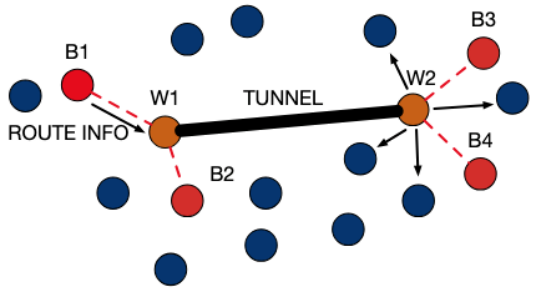
Limitations of Prior Works

- Traditional defense mechanisms are inadequate because they focus on detecting and mitigating *individual* attack patterns rather than identifying the *subtle interplay between multiple coordinated adversaries*.
  - For example, attackers deliberately distribute their malicious activities across multiple nodes to stay below detection thresholds or employ complementary attack methods that mask each other's signatures.
- Prior research on collaborative attacks
  - focused primarily on developing detection and defense strategies for traditional networks,
  - failing to address the resource limitations inherent to UAV platforms, their limited energy capacity and limited onboard computational capabilities.
- These constraints affect the feasibility and effectiveness of security solutions, as resource-intensive defense mechanisms can potentially compromise the network's operational lifetime and mission capabilities.

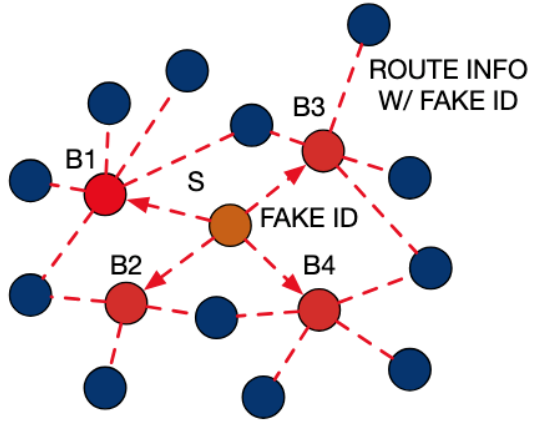
Understanding Threat Models against Autonomous UAV Networks

- Real world implications
  - In search and rescue operations, coordinated attacks can disrupt communications while injecting false data to mislead rescue efforts.
  - In precision agriculture, attackers can manipulate UAV routing while depleting ground sensor resources in critical areas.
  - In military applications, physical sabotage may be synchronized with cyber attacks to maximize fleet vulnerability.
- A collaborative attack is composed of individual, atomic attacks
  - Sybil attacks, where attackers create multiple fake identities to manipulate system-wide collaborative decisions;
  - Wormhole attacks, where adversaries record and retransmit packets between different network locations to disrupt routing;
  - Black hole attacks, where malicious nodes advertise false shortest paths to intercept network traffic.

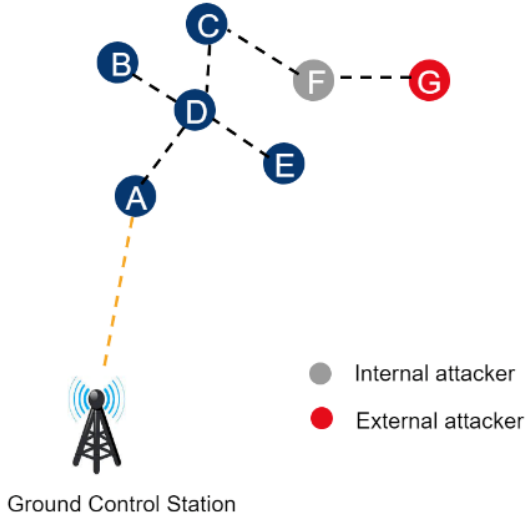
Examples of Collaborative Attacks



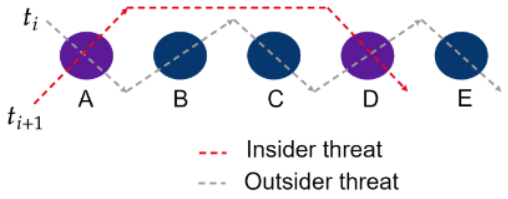
(a) Wormhole and black hole attack.



(b) Sybil attack combined with Black hole attack.



(c) Insider and outsider attack.



(d) Sequential and coordinated actions of insider and outsider attackers.

(a): Wormhole nodes (W1, W2) collaborating with blackhole nodes (B1-B4) for amplified routing disruption;
(b): Sybil node (S) providing fake IDs to blackhole nodes for detection evasion;
(c) and (d): Sequential insider (F) and outsider (G) attack coordination across time steps.

The Hiding Effect: A Blackhole-Sybil example

- Blackhole-Sybil Collaborative Attack
  - The Sybil adversary secretly transfers valid IDs to blackhole adversaries instead of abusing them.
    - SE1: It does not trigger abnormal events and, thus, is hidden from detection.
  - Blackhole adversaries broadcast false routing information with distinct IDs received from the Sybil adversary.
    - SE2: Making blacklisting-based blackhole attack defense mechanisms[4~6] invalid.

Task 2: Designing Mechanisms to Secure Autonomous UAV Networks against Collab- orative Attacks

- Defense against Multi-Stage Attacks (MSAs) vs. Defense against Collaborative Attacks:
  - Defenses against MSAs are concerned with finding patterns in sequential actions that are typically conducted by a single attacker.
  - By contrast, collaborative attacks are more general because they involve multiple attackers, who can execute their actions in an interleaved fashion which is more sophisticated than sequential events.
- Nevertheless, the research will use sequential data analysis as some building-blocks in the defenses.
- Moreover, the MSAs launched by a single attacker cannot achieve synergy effects as discussed above.

Task 2: Designing Mechanisms to Secure Autonomous UAV Networks against Collab- orative Attacks

- Current multi-stage attack defenses fail to address three challenges in UAV environments:
  - (1) coordinated actions by multiple attackers,
  - (2) rapidly changing attack surfaces due to UAV mobility, and
  - (3) resource constraints that limit computational defense mechanisms.

Using Formal Methods to Detect Collaborative Attacks

Traditional ML models struggle to detect multi-step, coordinated attack behavior. Collaborative attacks often span multiple nodes and evolve over time — difficult to capture via flow-based features alone.

Our Approach:

- Treat network events as **state transitions** in a system model.
- Use **formal verification techniques** (e.g., model checking) to:
  - Detect illegal state transitions or unexpected sequences.
  - Identify if multiple agents are collaborating across time to breach security.

Key Ideas:

- Model UAV network behavior using **finite state machines** or **temporal logic**.
- Encode known benign and malicious interaction patterns.
- Use runtime verification or offline checking to flag potential coordinated attacks.

Learning-Based Models for Sequential Detection

Hidden Markov Models (HMM):

- Probabilistic models capturing transitions between hidden system states.
- Suitable for lightweight onboard detection.
- Assumes limited memory (Markov property) and works best with relatively simple or periodic attack patterns.

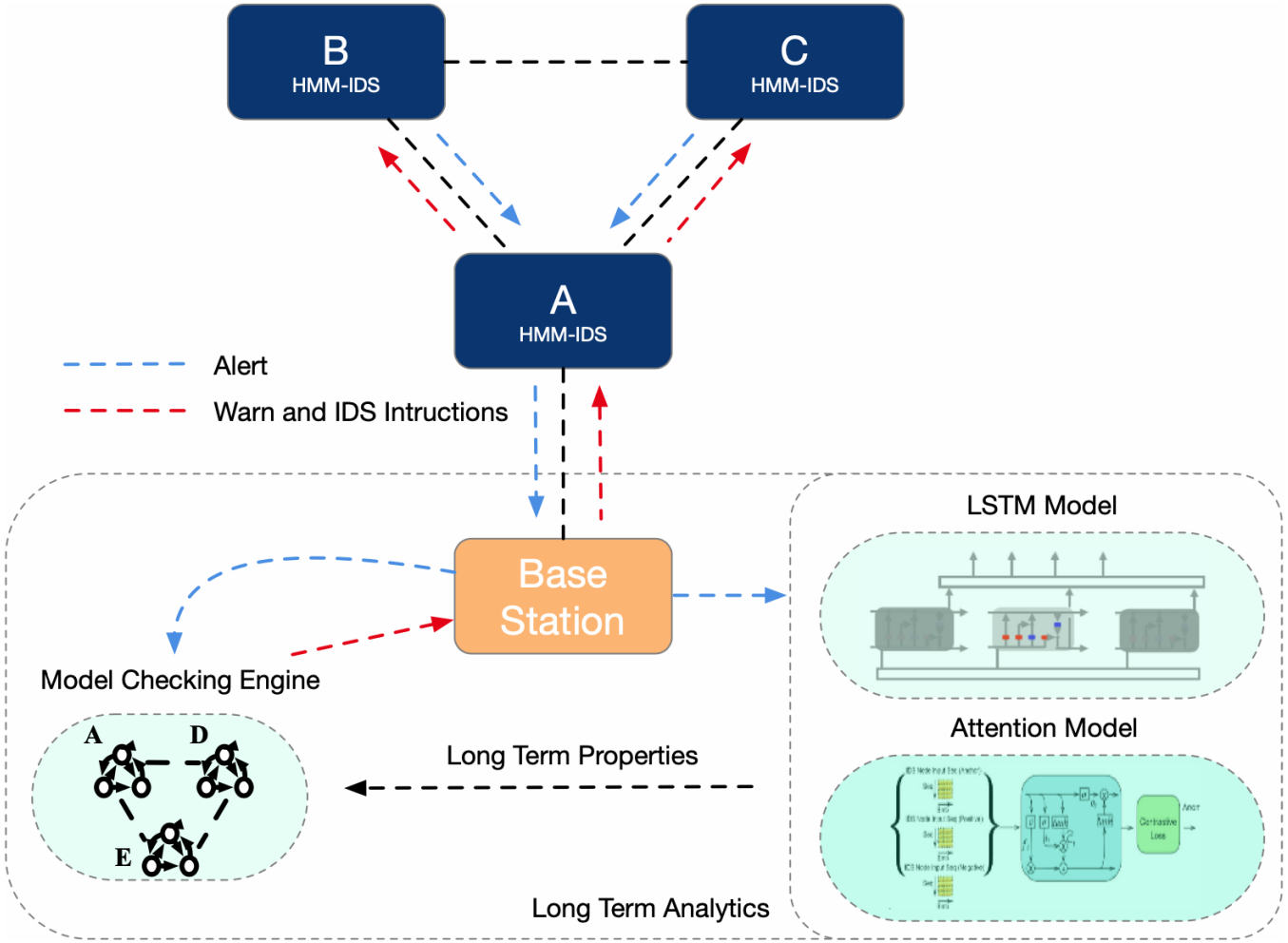
Long Short-Term Memory Networks (LSTM):

- A type of recurrent neural network designed to capture long-term dependencies.
- Effective in learning subtle, delayed, or multi-stage attack patterns.
- Requires extensive training data and higher computational resources.

Examples of events and actions associated with the four proposed mechanisms in the context of three examples of collaborative attacks.

Col. Attack	**HMM**	**LSTM**	**Attention**	**Model Checking**
Hidden Black-hole with Sybil	- Sudden routing changes - ID distribution events - Immediate routing manipulations	- Trust building period - Gradual increase in ID generation - Correlation between distributions and attacks	- ID transfer moments - Attack timing patterns - Network topology shifts	- FSM state transitions validation - ID distribution → attack sequencing - Temporal constraint verification
Insider-Outsider Coordi-nation	- Internal scanning activities - External connection attempts - Unusual data flows	- Insider's information gathering - Reconnaissance patterns - Behavioral shifts over time	- Insider-outsider communications - Behavior change points - Attack timing correlations	- Reconnaissance → attack sequencing - Temporal dependency validation - Collaboration pattern matching
Wormhole-Enhanced Black-hole	- Tunnel establishment attempts - Routing manipulations - Packet dropping events	- Position establishment patterns - Tunnel-blackhole correlations - Attack evolution across segments	- Tunnel establishment timing - Node synchronization points - Information duplication events	- Attack model conformance - Spatial-temporal constraint checks - Amplification effect verification

Task 3: Designing Defense Framework to Counter Collaborative Attacks



A Proposed Framework

- The figure highlights the preliminary framework
  - The framework implements a distributed detection architecture that balances detection capability with UAV resource constraints.
  - It leverages the Base Station (BS) for computationally intensive analysis while deploying efficient short-term detection on individual UAVs.
- Base Station Components
  - The Model Checking Engine forms the framework's foundation, providing three functions.
    1. Discovers and verifies possible attack collaboration patterns.
    2. Generates attack signatures to guide HMM deployment on UAVs.
    3. Identifies temporal dependencies and critical events for LSTM and attention analysis.

A Proposed Framework

- The architecture offers several key advantages.
  - Immediate threat detection occurs at network edges through lightweight HMMs while complex attack evolution analysis is centralized at the BS.
  - UAVs maintain minimal state information and computational overhead, yet the framework continuously adapts to new attack patterns.
  - Most importantly, all detection decisions can be verified through model checking.
- The preliminary experiments show this distributed approach provides a good tradeoff between effectiveness in detecting collaborative attacks and resource utilization.

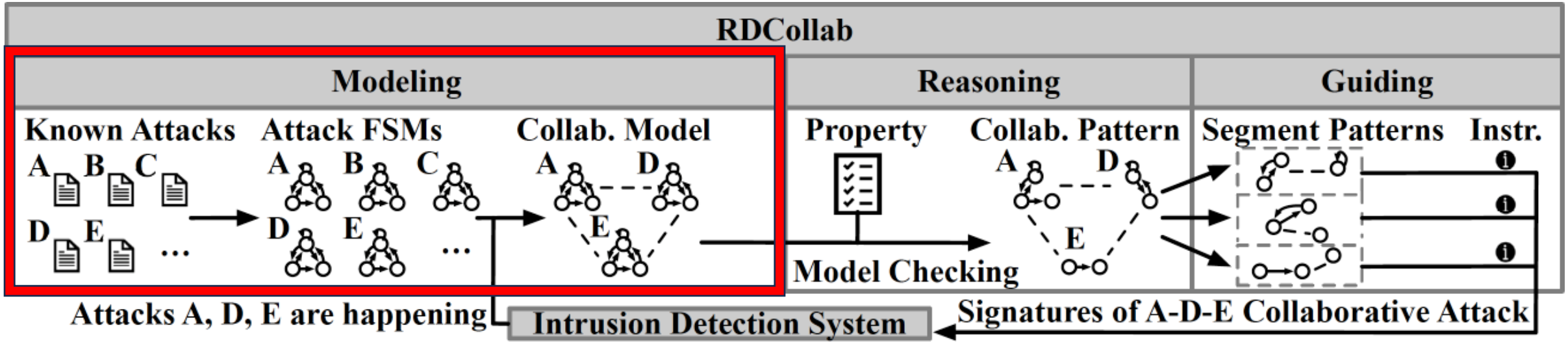
A Variant of the Proposed Framework

- RDCollab: **R**easoning and **D**etecting **Collab**orative Attacks
  - A framework to reason about and detect collaborative attacks in autonomous UAV networks that leverages model checking to explore attack patterns.
  - It models individual attacks as finite state machines (FSMs) that can be combined to represent collaboration.
  - The framework uses properties describing network safety to explore collaborative attacks and guide intrusion detection systems.
  - RDCollab improves IDSs' detection rates on non-hidden adversaries by up to 63% and can detect hidden adversaries within 6.1 seconds.

A Variant of the Proposed Framework

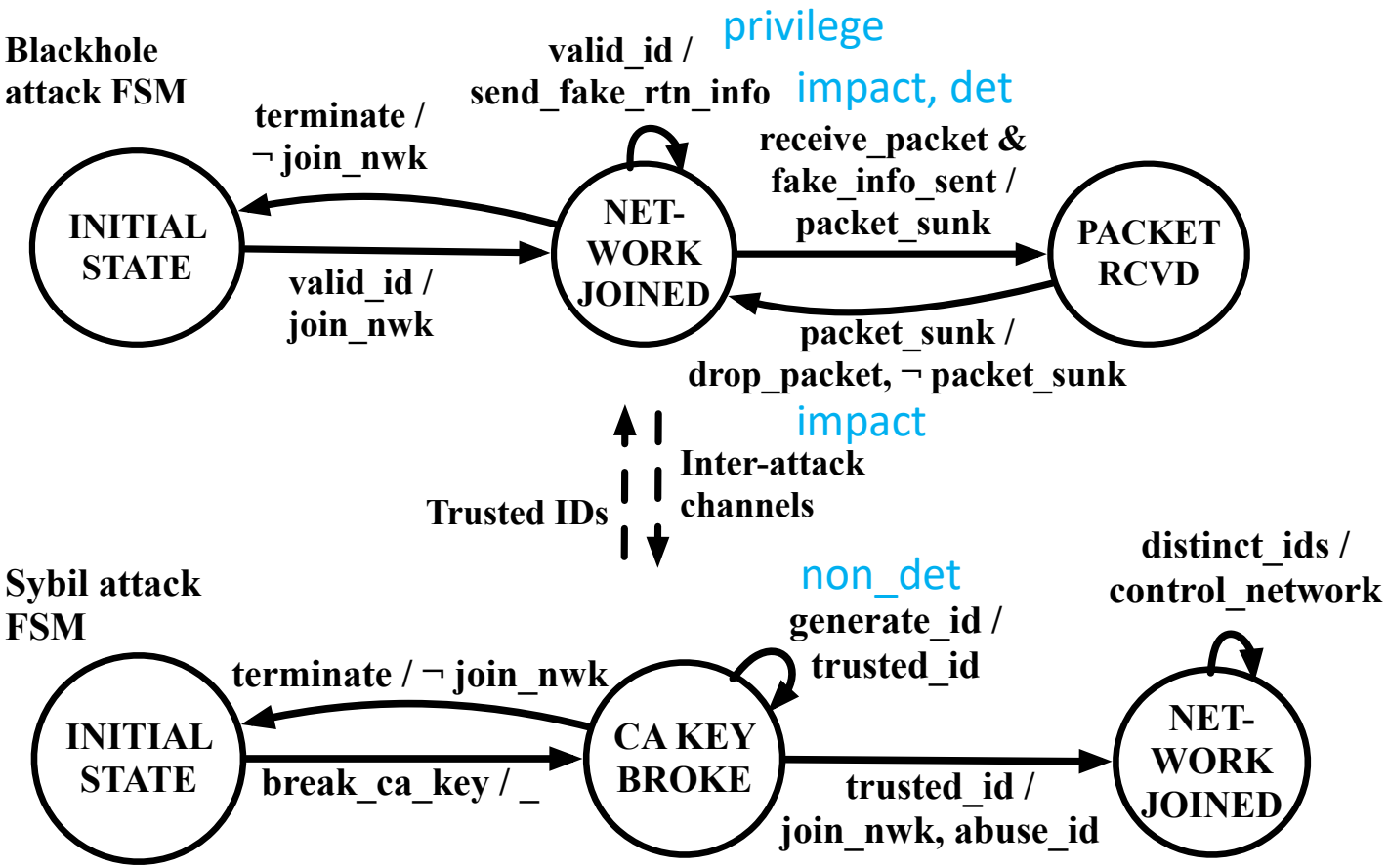
- RDCollab: **R**easoning and **D**etecting **Collab**orative Attacks
  - Three major synergy effects are identified: Hiding Effect, Amplifying Effect, and Shortcut Effect.
    - The Hiding Effect makes adversaries more difficult to detect or makes specific adversaries undetectable.
    - The Amplifying Effect occurs when coordinated attacks cause more damage than the sum of individual attacks.
    - The Shortcut Effect allows adversaries to launch attacks with fewer steps or achieve goals faster.
  - Seven novel collaborative attacks were discovered through model checking, including Hidden Blackhole Attack, Hidden Wormhole Attack, Duplicated Routing Disturbance Attack, Distributed Data Exfiltration Attack, and others.
  - The framework extracts segment patterns from collaborative attack counterexamples as signatures to guide IDSs.
  - Three baseline ML-based IDSs were used for evaluation: HMM-based, SVM-based, and RFC-based.

Our Solution: RDCollab (**R**easoning and **D**etecting **Collab**orative Attacks)



The collaborating attacks are formally modeled as finite states machines (FSMs) connected by channels for message exchange.

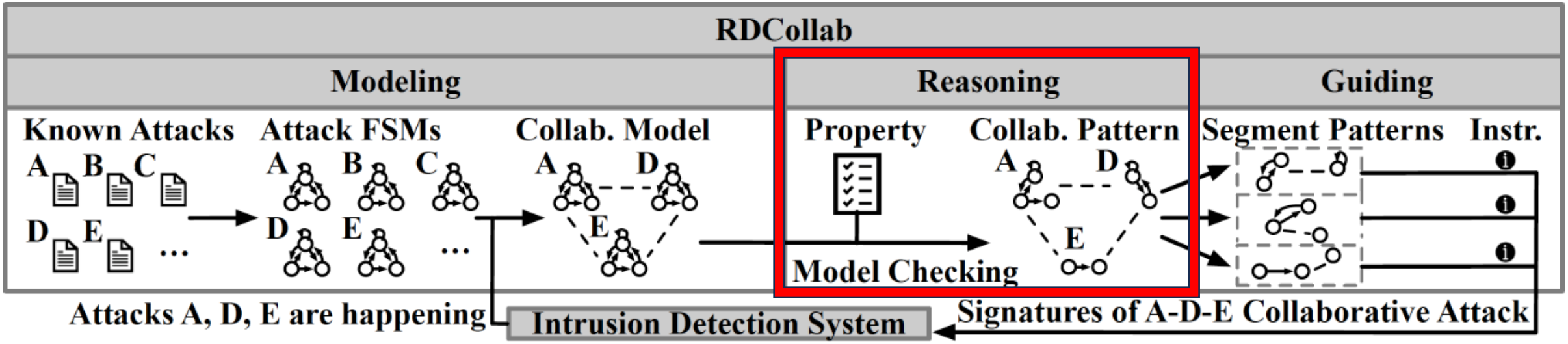
An Example of Collaborative Attack Model



More Details of the Modeling Phase

- We define impact labels that describe how an adversary's actions violate the confidentiality, integrity, or availability of the system. For example, we label "send_fake_rtn_info” (send fake routing information) with "int_viol" for integrity violation, and "drop_packet" with "avai_viol" for availability violation.
- Then we have detectability labels that indicate how observable an adversary's actions are. We use "det" for detectable actions, "part_det" for partially detectable actions, and "non_det" for actions that can't be observed externally. For instance, "send_fake_rtn_info" is labeled as "det" since it's observable, while "generate_id" in the Sybil FSM is labeled as "non_det" as it's done locally by the adversary.
- These labels help us bridge the gap between abstract security requirements and concrete system behaviors.

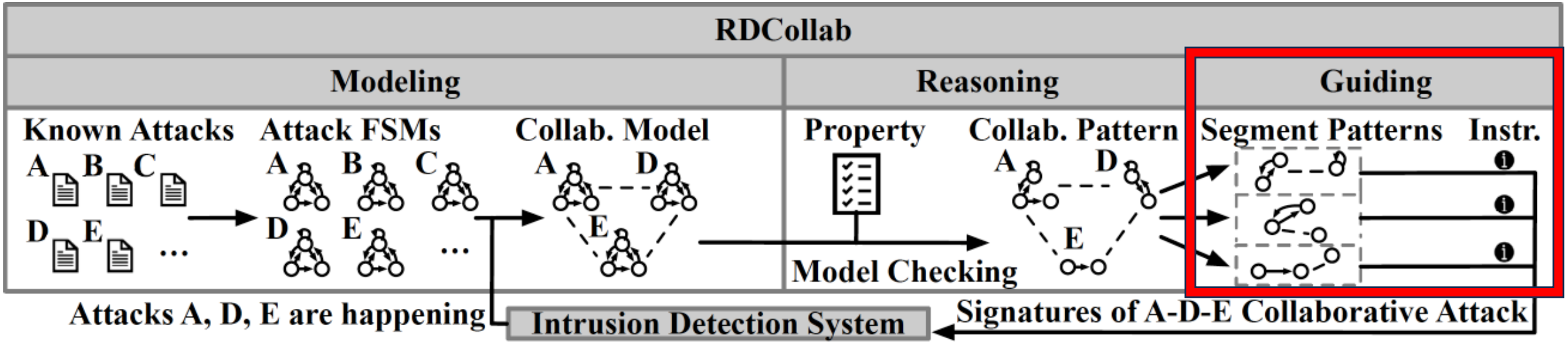
Our Solution: RDCollab (**R**easoning and **D**etecting **Collab**orative Attacks)



Synergy effects are encoded as security properties in linear temporal logic (LTL) formulas.

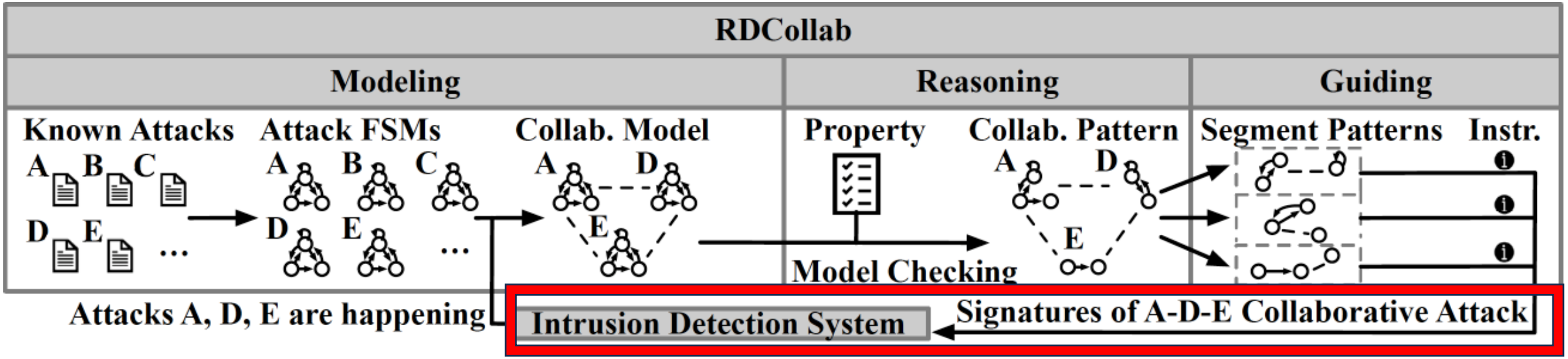
Model checking is used to check collaborating model consisting of attack FSMs against the LTL-form security properties indicating attack collaborations.

Our Solution: RDCollab (**R**easoning and **D**etecting **Collab**orative Attacks)



If the check fails, the model checker outputs a counterexample which can be referred to by IDS as oracles on how individual attacks collaborates.

Our Solution: RDCollab (**R**easoning and **D**etecting **Collab**orative Attacks)



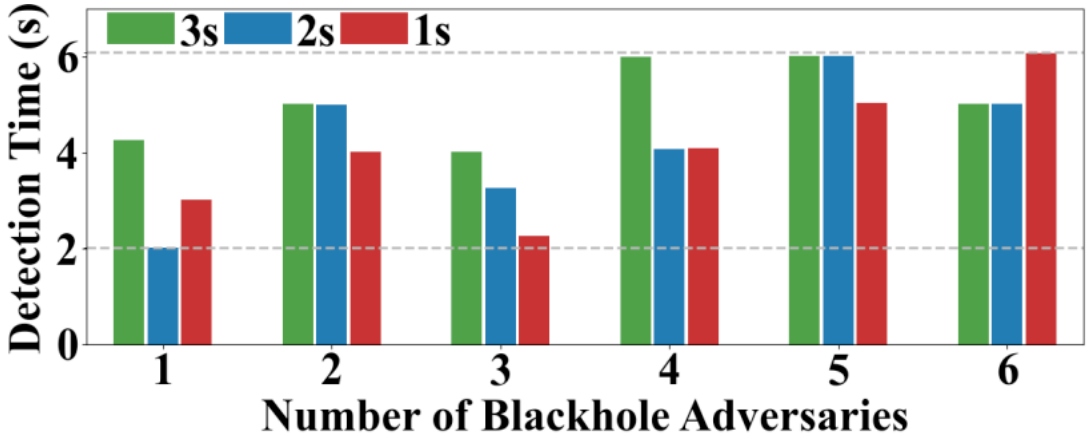
The IDS thus updates its detection and defense strategy accordingly.

More Details of the Reasoning and Guiding Phases

- RDCollab defines properties the model should satisfy in a synergy effects-free scenario (i.e., the scenario without the collaborative attacks that cause synergy effects).
- It then uses the model checking to verify the collaboration model against the properties. If the model violates a property, the model checker outputs a counterexample demonstrating a model execution that can be interpreted as a collaboration pattern with synergy effects.
- The segments of such a pattern are used in the guiding phase, as they can be used as signatures to detect collaborative attacks.
- RDCollab translates the segments into instructions to guide IDS in improving its detection effectiveness.

RDCollab Evaluation Results

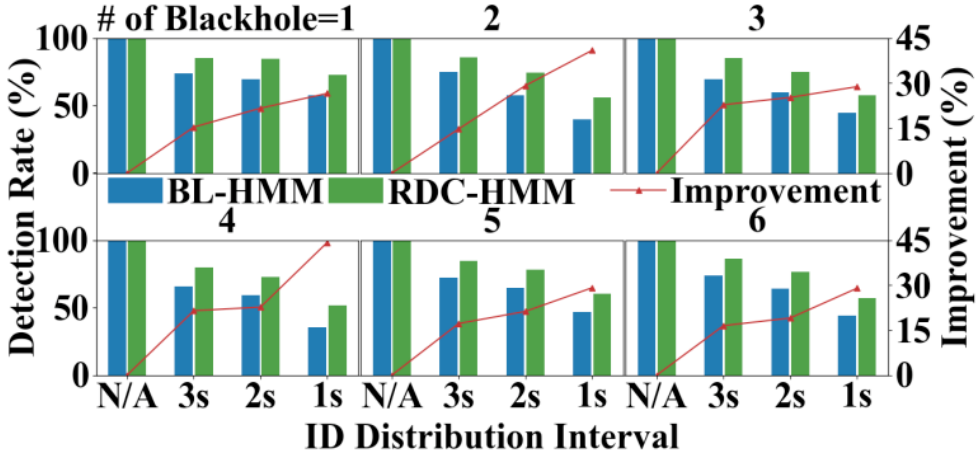
- Detecting the Hidden Sybil Adversary (SE1)



- With various numbers (1~6, x-axis) of blackhole adversaries, RDCollab can detect the Sybil adversary that transfer fake IDs to them every 3, 2 and 1 second(s) within 6 seconds (y-axis).

RDCollab Evaluation Results

- Improving the Detection of Blackhole Adversaries (SE2)



- With various numbers (1~6, subfigure titles) of blackhole adversaries, RDCollab-guided HMM-based IDS improves the detection rate of blackhole adversaries compared with baseline HMM-based IDS.
- The improvements are shown by the red lines.

Takeaways and Contributions

- Collaborative attacks pose a growing threat to UAV networks.
- RDCollab provides a comprehensive approach to tackling these challenges.
  - A solution to detect collaborative attacks against autonomous UAV networks.
  - Implementation of RDCollab instantiating the proposed solution.
  - Evaluation of RDCollab’s effectiveness of collaborative attack detection.
- The next steps are to enhance collaborative attack detection and response systems.

Another Issue: Lack of Robustness in intrusion detection systems for UAV networks

- Current IDS systems lack evaluation on diverse, dynamic UAV datasets and give high False positives caused by varying network conditions and congestion, while often demanding high computational resources.
- Existing IDS datasets lack UAV-specific attacks, aerial mobility, and real wireless traffic patterns.
- Due to High costs of collecting large network datasets, we use data augmentation using MLP function approximation method. This makes current IDS system robust against false positives caused by mobility in UAV networks.

Three-Phase Plan for Advancing Collaborative Attack Detection

Phase 1: Dataset Creation

- Develop a comprehensive dataset simulating collaborative attacks in UAV networks.
- Include realistic mobility models, packet-level logging, and diverse adversarial behaviors.
- Emphasize attack diversity (e.g., blackhole, wormhole, Sybil) and coordination mechanisms.
- Use Data Augmentation to increase diversity of dataset.

Phase 2: Transformer-Based Detection

•Leverage attention-based models (e.g., Transformers) to capture temporal dependencies and multi-agent coordination patterns.
•Fine-tune on domain-specific UAV datasets.
•Benchmark against baseline ML models and analyze detection accuracy under dynamic conditions.

Phase 3: Enhancing Detection Efficiency

•Optimize detection models for onboard deployment using techniques like knowledge distillation, pruning, and quantization.
•Explore hierarchical detection: light-weight edge models + cloud-based heavy analysis.
•Integrate detection with real-time response mechanisms.

Why Existing Datasets Fall Short for UAV Networks

Lack UAV-specific attacks

- Most datasets focus on generic IT or IoT threats, not aerial or coordinated UAV threats.

Static or limited mobility

- Many datasets assume fixed topologies, unsuitable for dynamic, mobile UAV environments.

No real UAV traffic

- Missing realistic communication patterns like video/image transmission or inter-UAV coordination.

Not designed for swarm behavior

- Fail to capture group dynamics, cooperation, or synchronized attacks.

Misaligned threat models

- Include irrelevant attack types (e.g., fuzzing, worms) not applicable to UAV use cases.

Partial relevance (e.g., IoT/WSN)

- Offer some overlap in resource constraints, but lack full UAV context.

Phase 1: Our UAV IDS Dataset: Key Features

Dynamic UAV Network & Mobility

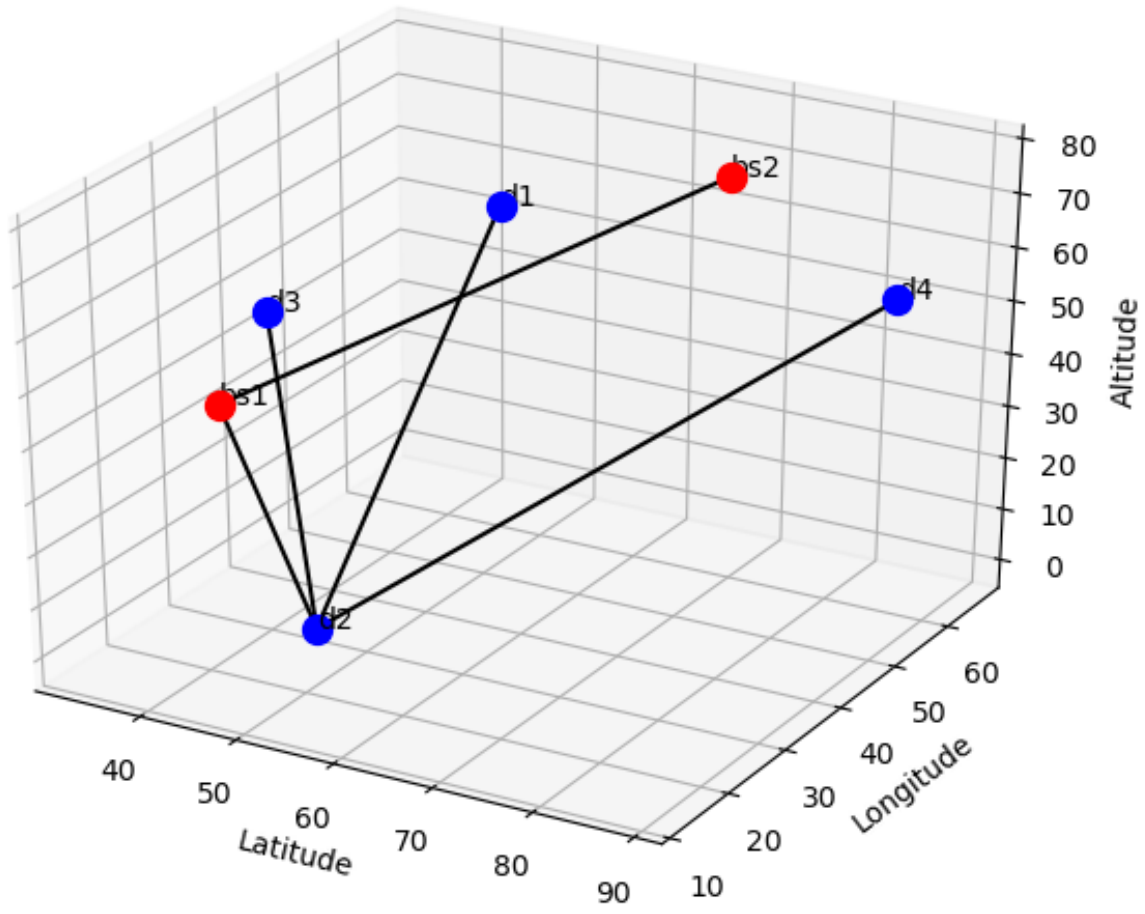
- UAVs (10-50) & Basestations (1-5) move using Gaussian-Markov model.
- Links dynamically change based on Euclidean distance, simulating WiFi characteristics (capacity, delay, loss).
- Packet loss modeled via BER, SNR, & FSPL, with retransmission-based correction.

Attacks Simulated

- DoS & DDoS – SYN floods (100 to 100K packets/sec), Black Hole, Wormhole, Replay Attack.

Realistic UAV Data Capture

- UAVs send images/videos, simulating reconnaissance transmission.
- Traffic captured at node & switch levels for IDS evaluation.
- Systematic variation: 100 runs per setup (60s each) with diverse UAV counts, attacks, & packet rates.



Phase 1: Our data augmentation: Using MLP as a function approximation

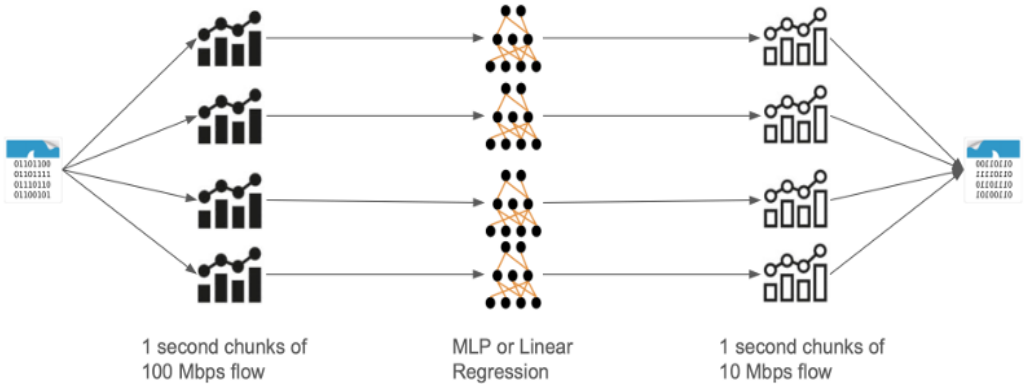
MLPs (Multi-Layer Perceptrons): are powerful function approximators that can learn nonlinear mappings between input and output feature distributions.

They capture complex relationships between flow statistics like inter-arrival times, packet sizes, and burstiness patterns across different bandwidths.

Unlike linear regression, MLPs can model diverse traffic behaviors more accurately, especially when the transformation is not linearly scalable.

Phase 1: Our data augmentation: Using MLP as a function approximation

- **Input**: Original network traffic is segmented into 1-second flow chunks from a 100 Mbps environment.
- **Feature Extraction**: We extract statistical features for each chunk (e.g., flow size, number of packets, arrival times).
- **MLP Transformation**: These features are passed through a trained MLP (or linear regression as baseline) to generate a mapped version mimicking how the same traffic would appear on a 10 Mbps link.
- **Output**: The resulting 10 Mbps-style flow chunks are recombined into an augmented dataset with realistic traffic for low-bandwidth settings.



F1- score of different ML techniques using Flow features.

- We use these vanilla models on three public datasets.
- We measure 65 different flow features based on packet size, packet time of arrival, and TCP flags.
- We measure F1-scores as these datasets are imbalanced.
- As dataset diversity and complexity increase, traditional models struggle to generalize—highlighting the need for more expressive, context-aware approaches.

Machine learning model	**CICIDS 2017[x]**	**CICIOT2023[x]**	**UNSW-NB15[x]**
1D – CNN	97.98	67.68	60.62
Long short-term memory	87.15	64.29	46.17
Random Forest	99.63	79.12	87.66
Stochastic gradient descent	95.98	50.41	39.03
Logistic Regression	93.16	48.24	37.82
Multilayer perceptron	95.87	61.55	48.45

Phase 2 – Transformer-Based Detection Ideas

Older detection methods often rely on shallow models or handcrafted features, which struggle to capture the complex temporal and multi-agent coordination patterns in collaborative attacks—necessitating more expressive architectures like Transformers.

Multi-Agent Attention Modeling

- Capture inter-node interactions by modeling UAV network as a sequence of events with node-level embeddings.

Graph-Transformer Hybrid

- Combine GNNs with Transformers to account for both network structure and temporal behavior.

Contrastive Learning

- Learn discriminative features by contrasting collaborative vs. non-collaborative attack traces.

Few-Shot Fine-Tuning

- Enable rapid adaptation to unseen coordinated attacks with minimal labeled data.

Phase 3 – Making Detection Efficient

Multi-Agent Attention Modeling

- Capture inter-node interactions by modeling UAV network as a sequence of events with node-level embeddings.

Graph-Transformer Hybrid

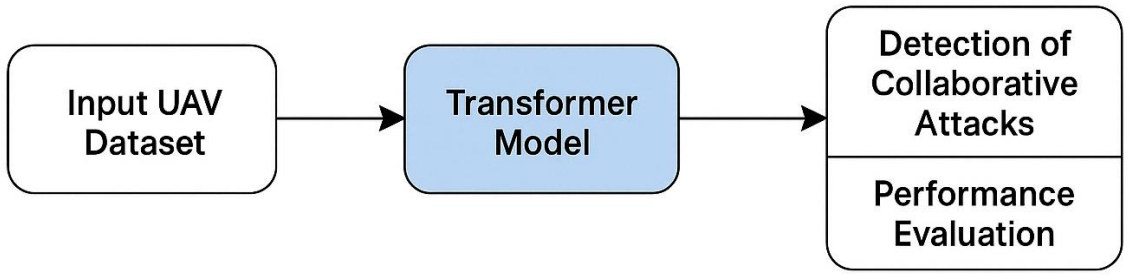
- Combine GNNs with Transformers to account for both network structure and temporal behavior.

Contrastive Learning

- Learn discriminative features by contrasting collaborative vs. non-collaborative attack traces.

Few-Shot Fine-Tuning

- Enable rapid adaptation to unseen coordinated attacks with minimal labeled data.



Phase 3 – Making Detection Efficient

Lightweight Transformer Variants

- Explore MobileBERT, TinyBERT, and Linformer for onboard inference with low overhead.

Hierarchical Deployment

- **On UAVs**: Quick screening via distilled models or rule-based alerts.
- **On Base Station**: Deep analysis using full models and historical context.

Dynamic Resource Adaptation

- Adjust model complexity based on available compute, energy, and threat severity.

On-the-Fly Model Updates

- Incorporate detected counterexamples from RDCollab to update the detection pipeline in real time.

Future Directions

- Apply RDCollab techniques to related domains like VANETs
- Apply federated learning across UAVs without raw data sharing

References

- [1] UAV Market to Worth USD 91.23 Billion by 2030 - UAV Industry Report by Fortune Business Insight : https://www.fortunebusinessinsights.com/industry-reports/unmanned-aerial-vehicle-uav-market-101603
- [2] Airlangga, Gregorius, and Alan Liu. "A Study of the Data Security Attack and Defense Pattern in a Centralized UAV–Cloud Architecture." Drones 7.5 (2023): 289.
- [3] Sarkar, Nurul I., and Sonia Gul. "Artificial intelligence-based autonomous UAV networks: A survey." Drones 7.5 (2023): 322.
- [4] T. Noguchi and T. Yamamoto, “Black hole attack prevention method using dynamic threshold in mobile ad hoc networks,” in 2017 Federated Conference on Computer Science and Information Systems (FedCSIS). IEEE, 2017, pp. 797–802.
- [5] J. Tobin, C. Thorpe, and L. Murphy, “An approach to mitigate black hole attacks on vehicular wireless networks,” in 2017 IEEE 85th vehicular technology conference (VTC Spring). IEEE, 2017, pp. 1–7.
- [6] T. N. D. Pham and C. K. Yeo, “Detecting colluding blackhole and greyhole attacks in delay tolerant networks,” IEEE Transactions on Mobile Computing, vol. 15, no. 5, pp. 1116–1129, 2015.