SSD: A State-based Stealthy Backdoor Attack For IMU/GNSS Navigation System in UAV Route Planning

Zhaoxuan Wang, Yang Li*, *Member, IEEE,* Jie Zhang, Xingshuo Han, Kangbo Liu, Yang Lyu, Yuan Zhou, Tianwei Zhang, *Member, IEEE,* and Quan Pan, *Member, IEEE.*

***Abstract*—Unmanned aerial vehicles (UAVs) are increasingly employed to perform high-risk tasks that require minimal human intervention. However, they face escalating cybersecurity threats, particularly from GNSS spoofing attacks. While previous studies have extensively investigated the impacts of GNSS spoofing on UAVs, few have focused on its effects on specific tasks. Moreover, the influence of UAV motion states on the assessment of cybersecurity risks is often overlooked. To address these gaps, we first provide a detailed evaluation of how motion states affect the effectiveness of network attacks. We demonstrate that nonlinear motion states not only enhance the effectiveness of position spoofing in GNSS spoofing attacks but also reduce the probability of detecting speed-related attacks. Building upon this, we propose a state-triggered backdoor attack method (SSD) to deceive GNSS systems and assess its risk to trajectory planning tasks. Extensive validation of SSD's effectiveness and stealthiness is conducted. Experimental results show that, with appropriately tuned hyperparameters, SSD significantly increases positioning errors and the risk of task failure, while maintaining high stealthy rates across three state-of-the-art detectors.**

***Index Terms*—Unmanned aerial vehicles, Cyber security, Backdoor attacks, GNSS spoofing.**

I. Introduction

Unmanned Aerial Vehicles (UAVs) are revolutionizing our understanding of low-altitude flight patterns. Currently, UAVs are widely used in various military and civilian fields, such as courier delivery, agricultural plant protection, power patrol, firefighting, rescue, and battlefield reconnaissance. These increasingly complex application scenarios have necessitated stringent requirements for the autonomous flight capabilities of UAVs. As UAVs operate autonomously to execute their missions, the system must precisely determine its global position at a centimeter level. As illustrated in Fig. 1, the localization capability is critical to route planning. It ensures the safety of flight and the ability to fulfill its mission, as positioning errors can directly cause the flight to deviate from course or fail to perform its mission.

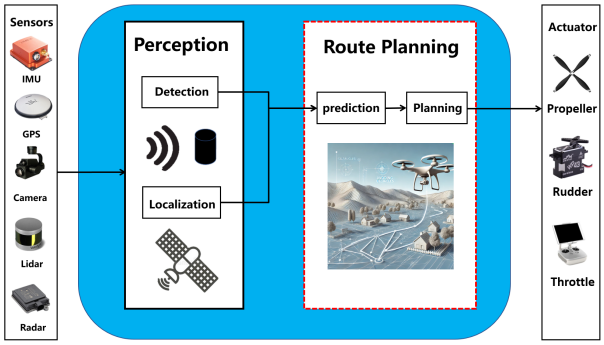


Fig. 1: The Role of Localization and Route Planing in UAV Autonomous Flight

The Integrated Navigation System (INS) serves as the cornerstone of UAVs, enabling precise positioning. It accomplishes accurate position estimates by integrating data from various sensors. Integrating Inertial Measurement Units (IMUs) and Global Navigation Satellite Systems (GNSS) forms the fundamental and core navigation system in INS. Building on this foundation, researchers usually incorporated additional sensors, such as cameras, lidar, and radar to further improve positioning accuracy in different scenarios or platforms. However, direct reliance on sensor data and communication channels' noise makes INS vulnerable to cyber-attacks [1], [2]. Research has revealed that adversaries can attack INS by using adversarial examples [3] or wireless signal injection [4] to spoof sensors. Notably, GNSS is a particularly prevalent threat since it forms the foundation of INS. The attacker can leverage low-cost devices to manipulate the position and velocity measurement captured by GNSS.

Previous attacks can be classified into the following categories:(1) **Direct Attacks** [5] : The adversary directly injects a false signal into the GNSS sensor. (2) **Stealthy Attacks** [6], [7]: The adversary takes the detector and corresponding threshold as the constraint and computes an optimization-based payload to bypass the detectors.

However, these attacks have the following limitations.**(1) Easily detectable** Sensor data fluctuations during UAV attacks are typically significant [8], prompting residual-based detection methods. However, our experiments show these

Zhaoxuan Wang is with School of Cybersecurity, Northwestern Polytechnical University, Xi'an 710129, China (e-mail: zxwang@mail.nwpu.edu.cn).

Yang Li, Yang Lyu, Kangbo Liu and Quan Pan are with School of Automation, Northwestern Polytechnical University, Xi'an 710129, China (e-mail:liyangnpu@nwpu.edu.cn; liukangbo@mail.nwpu.edu.cn; lyu.yang@nwpu.edu.cn quanpan@nwpu.edu.cn).

Jie Zhang is with CFAR and IHPC, Agency for Science, Technology and Research, Singapore. e-mail: (zhang_jie@cfar.a-star.edu.sg).

Xingshuo Han and Tianwei Zhang are with College of Computing and Data Science, Nanyang Technological University, Singapore 639798 (e-mail: xingshuo001@e.ntu.edu.sg; tianwei.zhang@ntu.edu.sg).

Yuan Zhou is with School of Computer Science and Technology, Zhejiang Sci-Tech University, Zhejiang 310018, China (email: yuanzhou@zstu.edu.cn).

Corresponding author: Yang Li.

detectors can almost always detect direct attacks. Moreover, during UAV maneuvers, such as waypoint course changes or formation shifts, the increased fluctuations make attacks even more easily identifiable. **(2) Computation efficiency** While stealthy attacks can bypass detectors through constrained optimization models, they typically require extensive matrix operations to solve high-dimensional optimization problems and derive the optimal attack payload in time. These methods introduce significant computational delays in environments with limited resources, especially affecting the attack's real-time performance. **(3) Inadequate analysis of dynamic vulnerabilities** Research [9] has shown that navigation algorithms in autonomous vehicles are vulnerable to uncertainty during specific periods. UAVs, operating with six degrees of freedom, experience frequent motion changes that affect system stability, especially during GNSS spoofing. Despite this, few studies evaluate how these dynamic motion states influence attack effectiveness. **(4) Incomplete assessment methodology** Prior studies [10], [11] often focus on immediate attack outcomes, such as crashes or path deviations, but fail to assess how attacks affect UAV mission performance and overall efficacy. This leaves a gap in understanding the broader impacts of such attacks on UAV operations.

To overcome these limitations, and gain a deep understanding of the INS vulnerability posed by GNSS attacks, we first provided a detailed interpretable analysis of the relationship between motion states and GNSS spoofing attacks. Specifically, we assess the effects of linear and nonlinear motion states on attack effectiveness. Our findings reveal that changes in motion states amplify the effectiveness of positional attacks and increase the stealthiness of velocity attacks. We then proposed SSD, a novel state-based stealthy backdoor attack for GNSS. Backdoor attacks [12], [13] are a common threat in deep neural networks, where an adversary implants a latent backdoor that remains inactive under normal conditions but is triggered by specific inputs or scenarios, leading to incorrect model predictions. Inspired by this concept, we design backdoors for GNSS by using motion state changes as a trigger to initiate staged velocity and positional attacks. In contrast to the stealthy attacks [6], [7], this attack mode doesn't need prior knowledge for the detectors and eliminates the need for complex computations. Instead, it is a direct attack that leverages carefully configured parameters and straightforward function calculations to achieve an optimal balance between effectiveness and stealthiness. This makes SSD highly valuable for engineering applications. Lastly, we selected three representative mission trajectories to assess the effectiveness of SSD. Its performance was compared against existing attack methods. The experimental results demonstrate that SSD maintains detection variables consistently within the threshold range in three classical detectors. Furthermore, we introduced evaluation metrics to measure the attack's impact on mission success rates and effectiveness. The experiments also reveal that SSD significantly increases the localization error, thereby effectively disrupting mission completion.

In summary, our contributions are summarized as follows:

- We present an interpretable mathematical security study of how motion states influence attack outcomes and demonstrate that UAVs are more vulnerable during maneuvers than in uniform linear flights. We further experimentally prove it.
- We design SSD, a novel state-based backdoor attack that utilizes motion state as a trigger to spoof GNSS data. It can execute velocity and positional attacks in stages to simultaneously and covertly attack both states.
- We conduct experiments in classic specific mission trajectories and find that SSD can significantly reduce mission completion rates and maintain constant stable stealthiness under attack detection.

II. Related Works and Background

A. UAV Route Planning and Integrated Navigation System

UAV route planning involves designing a feasible route from the start point to the destination while meeting all constraints and performance requirements [14]. Effective route planning is crucial for UAVs to complete their missions, and it depends on accurate position estimation. Since UAVs often operate in dynamic and complex environments, they rely on multi-sensor fusion to enhance their ability to perceive the environment. This approach integrates data from various sensors with different modalities and attributes, increasing redundancy and improving reliability in challenging conditions. The integrated navigation system of GNSS and IMU is a typical representative example. Its fusion strategy uses the GNSS data for quantitative updating, IMU data for state prediction, and an optimal estimation framework to achieve accurate positioning in the global coordinate system [15], [16]. In GNSS-denied environments, simultaneous localization and mapping (SLAM) that rely on camera [17], [18] and lidar [19], [20] are considered more reliable solutions. However, a single sensor alone cannot fully meet the demands for positioning accuracy and response speed. Therefore, combining these sensors with an inertial measurement unit to create IMU/Camera [21], [22] and IMU/Lidar [23] fusion form enables more precise position estimation and improved performance in dynamic environments. This paper focuses on the security analysis of IMU/GNSS INS since it plays a central role in UAVs. It is of generic meaning to study its security.

B. GNSS Spoofing attack

Since IMUs are more difficult to manipulate in real-world scenarios, we only discuss GNSS spoofing attacks for IMU/GNSS INS on UAVs. In such attacks, the adversary transmits false location coordinates to the GNSS receiver, thereby concealing the UAV's true location. As a result, unknowingly accepting these false inputs, the navigation system calculates incorrect position information. Specifically, GNSS spoofing includes direct attacks and stealthy attacks. The direct attacks [5] can be classified into these categories. (1) **Biased Signal Attack** : These attacks involve adding a bias to the GNSS sensor signals, typically following a uniform distribution. (2) **Multiplicative Attacks**: In these attacks, the GNSS signals are multiplied by a constant factor, effectively scaling the original signal values. (3) **Replacement Attacks**: These attacks involve directly replacing the GNSS signals with

false or manipulated data. Direct attacks are easy to implement and may lead to disastrous consequences, such as the UAV crashing into obstacles [10], [11]. However, most detectors can detect and respond in time. The stealthy attacks [6], [7] are diverse, with attackers often aiming to maximize navigation residuals to determine the optimal attack sequence. The design of these attacks may cause the UAV to fall into the malicious attackers' control [24]. However, it is largely influenced by the detection mechanism and often needs complex computing. SSD integrates the strengths of both approaches. It achieves the same effectiveness as stealthy attacks while requiring low computational resources.

To mitigate this threat, robust countermeasures, such as software analysis [25], cryptography-based authentication [26] and machine learning-based detections [27], [28], have been implemented to safeguard against GNSS spoofing and ensure the integrity of UAV INS. One important method is multi-sensor fusion [29]. As described in section II-A, it not only provides more accurate estimates for perception and localization but also enhances the data's trustworthiness, providing greater redundancy for detection and defense in the event of spoofing attacks. For example, Shen et al. [9] demonstrated that the effectiveness of constant offset GNSS spoofing attacks is greatly reduced in GNSS/INS/LiDAR fused navigation systems in autonomous driving. However, such multi-sensor fusion strategies also face a period of vulnerability and can not defend against constructed GNSS spoofing against the uncertainty that exists in the fusion algorithm itself. In this paper, we demonstrate a similar phenomenon in UAVs, where changes in motion states significantly amplify the uncertainty of the INS, making it more vulnerable to GNSS attacks. Therefore, we provide an in-depth analysis of the uncertainty and vulnerability and exploit it to design SSD.

C. Threat Model

Attack Goal As shown in Figure 2, the adversary aims to make the drone deviate significantly from its pre-planned route without triggering the stealth detection threshold. This objective can be formalized as the following optimization problem:

$$\begin{aligned} &\text{argmax}_{\delta} \quad \sum_{i=0}^{T_a} \mathbf{D_i^t} - \mathbf{D_i^a} \\ &s.t. \quad \chi_k \leq \tau \quad \forall k \in \{1, ..., T_a - 1\} \end{aligned} \tag{1}$$

where $\mathbf{D_i^t}$ and $\mathbf{D_i^a}$ denote the normal trajectory and the attacked trajectory, respectively. δ is the attack payload. χ_k is the in detection statistics and τ is the threshold of detectors.

Attack Scenario As shown in Fig. 2, an attacker can launch an attack in two ways: ① The attacker can use a UAV to fly alongside the victim's UAV. He can transmit legitimate GNSS signals completely using wireless attack devices such as software-defined radios (SDR). ② The attacker could also inject specific backdoors [30] or viruses [31] into the UAV by supply chain attacks, which could be used to monitor its dynamics and induce a false GNSS position [32].

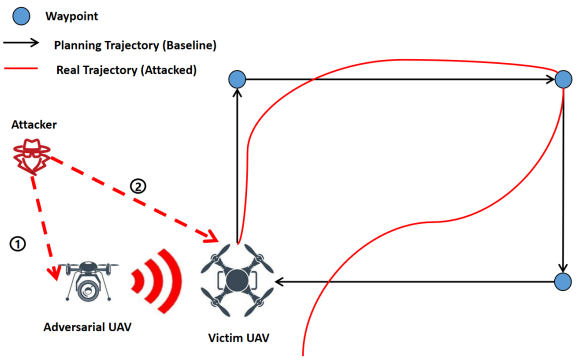


Fig. 2: Threat Model

Attacker's Capability 1) The attackers need white-box access to obtain the victim's navigation algorithms and corresponding parameters. They can get this knowledge through open-source channels since most UAVs use standardized open-source navigation algorithms [33], [34]. Also, the adversarial can use reverse engineering to access the victim's knowledge. 2) The attacker can obtain the victim's motion state, such as position and velocity. This can be achieved by monitoring UAVs using an additional GPS module or auxiliary object detection and tracking devices. 3) As IMU data is less likely to be accessed and used, an attacker can only modify the position and velocity since GNSS measurements only provide position and velocity to the UAV.

III. Preliminary

A. IMU/GNSS Integrated Navigation System

In UAVs, IMU and GNSS are often combined for highly accurate and robust navigation and positioning. IMU provides data from accelerometers and gyroscopes, which measure the acceleration and angular velocity of the UAVs. At the same time, GNSS determines the UAV's position, velocity, and time information by receiving satellite signals. This type of navigation system is known as a combined IMU/GNSS navigation system.

IMU/GNSS INS uses a 22-axis Extended Kalman Filter (EKF) structure to estimate pose in the NED reference frame. The state is defined as $\hat{X}_k = \{\hat{x}_1, \ldots, \hat{x}_n | n \in (1, 22)\}$, where the definition of each axis $\hat{x}_i$ is illustrated in Table I.

Firstly, The system model $f(\cdot)$ uses the estimated previous state $\hat{X}_{k-1}$ and a control input u_{k-1}, to predict the current state $\hat{X}_k^-$.

$$\hat{X}_k^- = f(\hat{X}_{k-1}, u_k) \tag{2}$$

where u_k are control inputs, typically angular velocity and acceleration data from IMU.

After prediction, INS predicts the measurement z_k at time k for updating the current state. we define as:

$$z_k = h(\hat{X}_k) + v_k \tag{3}$$

where z_k comprises magnetic field data from magnetometers, gravitational acceleration data from accelerometers, and

TABLE I: Element and Meaning of EKF Vector

Element	Label	Meaning
$\hat{x}_1$	$\mathbf{q_0}$	Orientation quaternion.
$\hat{x}_2$	$\mathbf{q_1}$	
$\hat{x}_3$	$\mathbf{q_2}$	
$\hat{x}_4$	$\mathbf{q_3}$	
$\hat{x}_5$	$\mathbf{P_N}$	UAV Position in local NED coordinate system.
$\hat{x}_6$	$\mathbf{P_E}$	
$\hat{x}_7$	$\mathbf{P_D}$	
$\hat{x}_8$	$\mathbf{V_N}$	UAV Velocity in local NED coordinate system.
$\hat{x}_9$	$\mathbf{V_E}$	
$\hat{x}_{10}$	$\mathbf{V_D}$	
$\hat{x}_{11}$	$\mathbf{\Delta\theta bias_X}$	Bias in integrated gyroscope reading.
$\hat{x}_{12}$	$\mathbf{\Delta\theta bias_Y}$	
$\hat{x}_{13}$	$\mathbf{\Delta\theta bias_Z}$	
$\hat{x}_{14}$	$\mathbf{\Delta vbias_X}$	Bias in integrated accelerometer reading.
$\hat{x}_{15}$	$\Delta vbias_Y$	
$\hat{x}_{16}$	$\mathbf{\Delta vbias_Z}$	
$\hat{x}_{17}$	$\mathbf{geomagneticField_N}$	Estimate of geomagnetic field vector at the reference location.
$\hat{x}_{18}$	$\mathbf{geomagneticField_E}$	
$\hat{x}_{19}$	$\mathbf{geomagneticField_D}$	
$\hat{x}_{20}$	$\mathbf{magbias_X}$	Bias in the magnetometer readings.
$\hat{x}_{21}$	$\mathbf{magbias_Y}$	
$\hat{x}_{22}$	$\mathbf{magbias_Z}$	

position and velocity from GNSS. $h(\cdot)$ is the measurement prediction model and v_k is the observational noise.

Due to hardware arithmetic limitations, INS often handles nonlinear functions by truncating their Taylor expansions with first-order linearization and neglecting the higher-order terms. This approach transforms the nonlinear problem into a linear one, as exemplified by Eq. 4 and 5:

$$f(X_{k-1}, u_k) \approx f(\hat{X}_{k-1}, u_k) + \left.\frac{\partial f}{\partial x}\right|_{\hat{X}_{k-1}, u_k} (X_{k-1} - \hat{X}_{k-1}) \tag{4}$$

$$h(X_k) \approx h(\hat{X}_k^-) + \left.\frac{\partial h}{\partial x}\right|_{\hat{X}_k^-} (X_k - \hat{X}_k^-) \tag{5}$$

INS defines the state transfer matrix F_K and the Jacobi matrix of the measurement H_k respectively:

$$F_k = \left.\frac{\partial f}{\partial x}\right|_{\hat{X}_{k-1}, u_k} \tag{6}$$

$$H_k = \left.\frac{\partial h}{\partial x}\right|_{\hat{X}_k^-} \tag{7}$$

Based on Eq. 2 and 3, we can derive the priori estimated covariance matrix P_k^- at time k.

$$e_k = \hat{X}_k - \hat{X}_k^- \tag{8}$$

$$\begin{aligned} P_k^- &= E(e_k e_k^T) \\ &= F_k P_{k-1} F_k^T + Q_k \end{aligned} \tag{9}$$

where Q_k refers to the process noise covariance matrix at time step k.

When obtaining the measurement z_k, the system will calculate the Kalman gain K_k and update the estimated state to obtain an accurate estimation of the state information in the following way:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \tag{10}$$

$$\hat{X}_k = \hat{X}_k^- + K_k(z_k - h(\hat{X}_k^-)) \tag{11}$$

$$P_k = (I - K_k H_k) P_k^- \tag{12}$$

where R_k refers to the measurement noise covariance matrix at time step k.

B. Detector

In dynamic system state estimation, the EKF optimizes the estimation of the system state successively through prediction and update steps. It computes the residual $r(k) = z_k - h(\hat{x}_k^-)$ in each step to reflect the difference between the actual measured value and the predicted value. With no attacks or anomalies, $r(k)$ will be presented as a zero-mean Gaussian distribution with a covariance matrix $Pr := H_k P_k H_k^T + R_k$.

However, the system may generate outliers due to attacks, noise, faults, and other factors, all of which contribute to the measurements deviating from the true values. To prevent the EKF state from these disruptive outliers, implementing an outlier detection mechanism becomes crucial. The chi-square statistical test serves as an efficient tool for determining outliers [35], [36]. It evaluates the current measured value by calculating the chi-square statistic χ_k^2, comparing it to a predefined statistical significance threshold. When χ_k^2 surpasses this threshold τ, the measurement is considered an outlier, and suitable measures are undertaken, including discarding the measurement or executing a partial update. The chi-square statistic χ_k^2 is defined as:

$$\begin{aligned} \chi_k^2 &= r(k)^T S_k r(k) \\ S_k &= (H_k P_k^- H_k^T + R_k) \end{aligned} \tag{13}$$

IV. Security Analysis

A. Attack Formulation

Viewed from the perspective of navigation equations, the process of a spoofing attack on GNSS signals by an attacker can be described as follows: the attacker injects n spoofing signal $\{\delta_k | k = 1, \ldots, n\}$ into the measurement data, resulting in a modification of the measurement h as follows.

$$z_k = h(\hat{X}_k) + \delta_k + v_k \tag{14}$$

Due to the higher occurrence of data errors in the GNSS *z-axis* and the availability of alternative altitude data sources, only the *NE* (North-East) directional updates are applied to the position vector. As for the attacker, they can only modify the position and speed provided by GNSS, i.e., dimensions 5-6 and 8-10 in Table I.

B. Study of Attack

The GNSS observation matrix H_{GNSS} are as follows.

$$H_{GNSS} = \begin{bmatrix} 0_{1\times4} & 1 & 0 & 0 & 0_{1\times3} & 0_{1\times14} \\ 0_{1\times4} & 0 & 1 & 0 & 0_{1\times3} & 0_{1\times14} \\ 0_{1\times4} & 0 & 0 & 0 & 0_{1\times3} & 0_{1\times14} \\ 0_{3\times4} & 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & I_{3\times3} & 0_{3\times14} \end{bmatrix} \tag{15}$$

Since H_{GNSS} is a sparse matrix, when updating position and velocity, K_k can be simplified to the following form:

$$\begin{aligned} K_k &= P_k^- H_{GPS}^T (H_{GPS} P_k^- H_{GPS}^T + R_k)^{-1} \\ &= (P_{k-1} + Q_k)(P_{k-1} + Q_k + R_k)^{-1} \\ &= I - R_k (P_{k-1} + Q_k + R_k)^{-1} \end{aligned} \tag{16}$$

From Eq.16, we can see that the state transfer error and the measurement error affect the magnitude of the gain K_k simultaneously. Q_k **and** R_k **reflect the ability to cover systematic uncertainty and measurement uncertainty, respectively**. Therefore, inappropriate Q_k and R_k can lead to filter divergence or biased estimation. However, in INS, Q_k and R_k are generally determined based on a priori knowledge by pre-running the filter calculations offline and remain constant during the filtering process online. As a result, both the process estimation error covariance R_k and the Kalman gain K_k converge quickly and remain constant during the process, demonstrating that the value of K_k is determined by the ratio of Q_k and R_k.

We assume the adversarial adds an δ_i at time i. The prediction equation for the EKF becomes

$$\begin{aligned} \hat{x}_i^a &= \hat{x}_i^- + K_i(z_i + \delta_i - h(\hat{x}_i^-, 0)) \\ &= \hat{x}_i + K_i \delta_i \end{aligned} \tag{17}$$

$$P_k = (I - K_k H_k) P_k^- \tag{18}$$

Therefore, when the UAV performs maneuvers, the impact of spoofing on localization results can be described in the following two ways:

- **Q uncertainty:** EKF uses Euler integrals to update the positional status, i.e:

$$\begin{bmatrix} P_N \\ P_E \\ P_D \end{bmatrix}_{i+1} = \begin{bmatrix} P_N \\ P_E \\ P_D \end{bmatrix}_i + \begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix}_i \Delta t \tag{19}$$

  Firstly, when performing maneuvers, the system is highly nonlinear. i.e.,

$$\exists \epsilon > 0, \| \frac{d\mathbf{v_i}}{dt} \| \geq \epsilon \tag{20}$$

  The acceleration $\mathbf{a_i} \neq \mathbf{0}$ and the update equation for position essentially becomes:

$$\begin{bmatrix} P_N \\ P_E \\ P_D \end{bmatrix}_{i+1} = \begin{bmatrix} P_N \\ P_E \\ P_D \end{bmatrix}_k + \begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix}_i \Delta t + \begin{bmatrix} \frac{1}{2}\Delta t^2, 0, 0 \\ 0, \frac{1}{2}\Delta t^2, 0 \\ 0, 0, \frac{1}{2}\Delta t^2 \end{bmatrix} \begin{bmatrix} a_N \\ a_E \\ a_D \end{bmatrix} \tag{21}$$

  According to Eq. 8 and 9, the accumulation of linearisation errors will increase e_i and thus increase the process noise P_i. leading to inaccurate mathematical modeling and huge nonlinear errors. The fixed Q_k makes it difficult to suppress the nonlinear error increased due to the change of motion state. According to Eq. 16, the value of K_i will indirectly increase, making the INS more inclined to trust the GNSS data. In addition, this complex nonlinear characteristic will be further expanded due to physical factors such as the lag of IMU data and the presence of friction in the gyroscope. Therefore, in this scenario, the prediction of the system model cannot effectively reflect the actual physical process.
- **R uncertainty:** The modification of GNSS results in a shift in the measurement data distribution, rendering a fixed R_k inadequate for accurately describing the measurement noise distribution. Consequently, δ_i experiences a significant increase. Additionally, the rising K_k value leads the system to place greater trust in GNSS, further amplifying the impact of the attack associated with δ_i.

We employ a biased signal attack and a multiplicative attack to evaluate the phenomenon above. The UAV maintains a constant velocity of $5m/s$ and performs two typical motion modes, uniform linear motion (linear motion) and uniform circular motion (non-linear motion), respectively. For each flight state, we apply an attack window of two attack inputs for the GNSS respectively and observe the changes in the localization Error $LocErr$ before and after the attack. As a result, the attack time is 2 seconds since the GNSS input is 1 Hz. The experimental results are shown in Fig. 3a and 3b. Thus, we can get Finding 1.

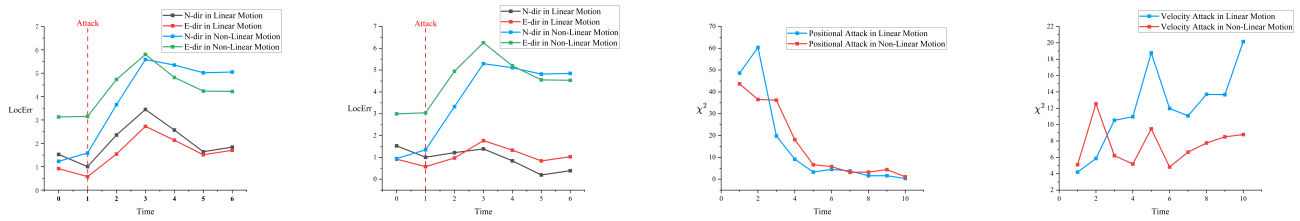
Finding 1: *For a GNSS spoofing of the same magnitude, applying it during the UAV's non-linear motion results in more pronounced fluctuations in positioning accuracy compared to linear motion. These changes in motion dynamics create greater vulnerabilities for INS.*

Position and velocity are tightly coupled in INS, making velocity attacks easier to modify the position result. However, there has been limited research on attacks targeting velocity. We would like to explore one question: Are there obvious correlations between the velocity attack stealthiness and the UAV motion states? To validate this question, we select two flight trajectories with linear and nonlinear motion states respectively. We use a biased signal attack to perturb velocity and position measurement and evaluate stealthiness by observing the changes in the chi-square detector. Fig. 3c and 3d indicate that the detector shows no significant fluctuations between the two motion states for the positional attack. However, velocity attacks demonstrate a greater sensitivity to motion states, with significantly lower cardinality detector results observed in nonlinear motion states. This reveals key insights, summarized as Finding 2.

Finding 2: *The nonlinear motion state does not significantly affect the stealthiness of positional attacks, but it enhances the stealthiness of velocity attacks.*

V. Attack Design

From the analysis of Section IV-B, we observe that UAV exhibits greater vulnerability in a non-linear motion state compared to a linear one due to the combined effects of model and measurement uncertainty. During non-linear motion, an attack of the same magnitude can produce a more significant change in the navigation output than in the linear motion state. However, this vulnerability arises only when the uncertainty is heightened due to changes in the motion state. Moreover,



(a) Bias Signal Attack (b) Multiplicative Attack (c) Positional Attack (d) Velocity Attack

Fig. 3: Study of GNSS Attack under Different Motion states, where N-dir and E-dir denote the north and east in the NE direction.

Finding2 indicates that a positional attack in non-linear motion will increase the risk of being detected. This introduces a key challenge to the attacker:

C1: How to opportunistically exploit these vulnerable periods to achieve maximum localization error while maintaining stealth.

To address **C1**, we design a backdoor-like attack to exploit these vulnerable periods directly. Inspired by the backdoor attacks in deep networks (DNNs) [12], [37], [38], we proposed a novel state-based stealthy backdoor (SSD) attack against INS in a route planning scenario. SSD utilizes the motion state as a trigger, enabling the UAV to trigger an attack in a nonlinear motion state while maintaining normal operation in a linear motion state, the attack procedures can be depicted in Fig. 4. Based on this design, not only can the detection rates of the attack be greatly reduced, but also the mission completion rate of the UAV can be effectively reduced. (UAV mission completion is usually accompanied by large maneuvers.)

$$\exists c \in \mathbb{R}, \| \frac{d\mathbf{v_i}}{dt} \| \leq c \quad \forall t \in [t_0, t_1] \tag{22}$$

Specifically, when the victim UAV is in a stable linear flight state (Equation 22), SSD adds a bias $F(t_i; \theta, \alpha)$ to attack the position stealthily.

$$F(t_i; \theta, \alpha) = \theta e^{t_i/\alpha} \tag{23}$$

When it is maneuvering, each dimension of the UAV's movement can experience both positive and negative acceleration, indicating acceleration and deceleration in that particular direction. We define the acceleration direction $\vec{a}$ as:

$$\vec{a}_i^d = \frac{\partial \vec{v}_i}{\partial t_i} \tag{24}$$

When the UAV undergoes acceleration in this dimension (i.e. $\vec{a}_i^d > 0$), SSD makes smooth changes to the velocity by multiplying a stealthy velocity bias $G(t_i, a_i^d; \phi)$ to perturb the localization result.

$$G(t_i, a_i^d; \phi) = \log_2 (2 + \phi \vec{a}_i^d t_i) \tag{25}$$

Overall, SSD can be formalized as follows:

$$\begin{cases} X_i = X_i + F(t_i; \theta, \alpha) \\ V_i = V_i * G(t_i, a_i^d; \phi) \end{cases} \tag{26}$$

where t_i is the attack time for attackers. θ, α, and ϕ are hyperparameters that can be dynamically adjusted to maintain an equilibrium between stealthiness and effectiveness. We will demonstrate how to configure these parameters in Section VI-B. SSD chooses acceleration as a trigger. It uses organic coupling between velocity and position to perform a combined attack. Its pseudocode is presented in Algorithm 1.

Algorithm 1: SSD

Input: Victim UAV position $P_i = (P_{Ni}, P_{Ei}, P_{Di})$, Victim UAV velocity $V_i = (V_{Ni}, V_{Ei}, V_{Di})$, IMU sampling frequency F_{imu}, Iteration number M

Set initialize hyperparameters θ,ϕ and α ;
for *i=1 to M* **do**
  for *j=1 to* F_{imu} **do**
    Receive IMU data ;
    Predict X_{i+1} using IMU data;
  Receive victim UAV velocity V_i and position P_i from GNSS ;
  Compute acceleration $a_i = (\frac{\partial V_{Ni}}{\partial t_i}, \frac{\partial V_{Ei}}{\partial t_i}, \frac{\partial V_{Di}}{\partial t_i})$;
  if $||a_i||_2$*=0* **then**
    // Apply position perturbation in linear motion state ;
    $P_i = P_i + \theta e^{ti/\alpha}$;
  else
    // Apply velocity perturbation in nonlinear motion state ;
    $V_{Ni} = V_{Ni} * \log_2 (2 + \frac{\partial v_{Ni}}{\partial t} \phi t)$;
    $V_{Ei} = V_{Ei} * \log_2 (2 + \frac{\partial v_{Ei}}{\partial t} \phi t)$;
  Fuse P_i and V_i to update X_{i+1}

VI. Experiment

A. Experimental Setup

1) Trajectory Dataset: We used three representative task trajectories. Trajectories I and II represent typical linear and non-linear motion states, respectively, while trajectory III combines both motion states.

- **Trajectory I: Straight-line Path** As illustrated in Fig. 5a, this trajectory is designed for executing simple tasks, such as flying to a specific location and performing actions along a predefined path (e.g., patrols, cargo transport, etc.).

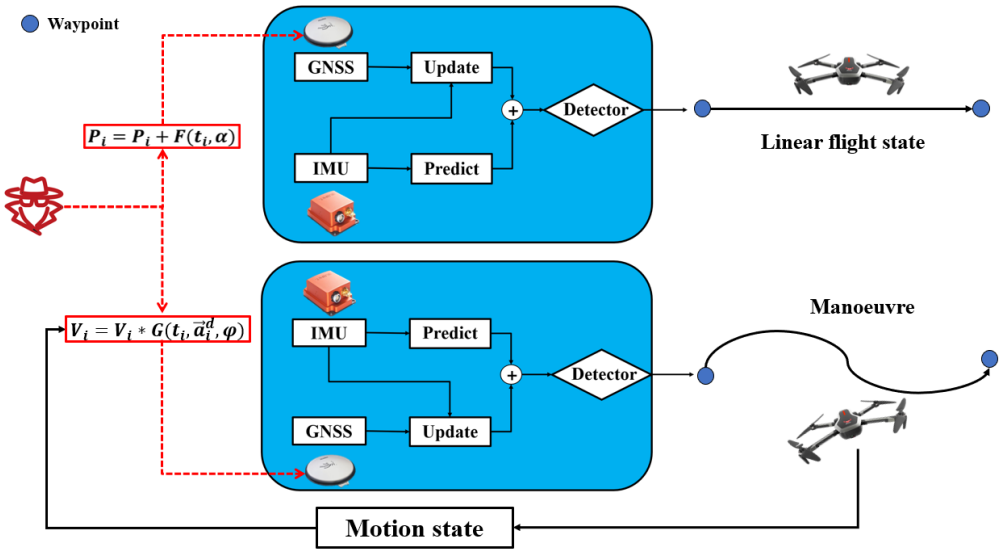
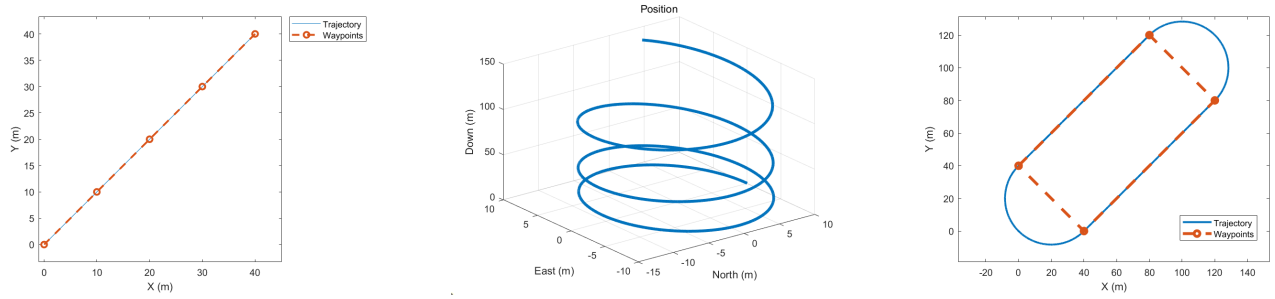


Fig. 4: Overview of SSD



(a) Straight-line Path (b) Spiral Path (c) U-shape Path

Fig. 5: Trajectory Visualization

- **Trajectory II: Spiral Path:** As illustrated in Fig. 5b, this trajectory is designed for tasks that involve changes in flight altitude, such as agricultural spraying, area scanning, and 3D mapping.
- **Trajectory III: U-shape Path:** As illustrated in Fig. 5c, the trajectory is designed for tasks that demand high precision, such as monitoring, surveying, or round-trip transportation.

2) Navigation Algorithm: We choose the estimation and control library EKF (ECL EKF2) of the PX4 drone autopilot [39] project as the target navigation algorithms, assuming white-box access, where the attacker has full knowledge of the algorithms and their parameters.

- **ECL EKF2** implements EKF to estimate pose in the NED reference frame by fusing MARG (magnetic, angular rate, gravity) and GNSS data. MARG data is derived from magnetometer, gyroscope, and accelerometer sensors. It uses a 22-element state vector to track the orientation quaternion, velocity, position, MARG sensor biases, and geomagnetic.
- **CD-EKF** is a variant of ECL EKF2. It implements a continuous-discrete EKF to estimate pose in the NED reference frame by fusing MARG and GNSS data. It uses a 28-element state vector to track the orientation quaternion, velocity, position, MARG sensor biases, and geomagnetic vector.

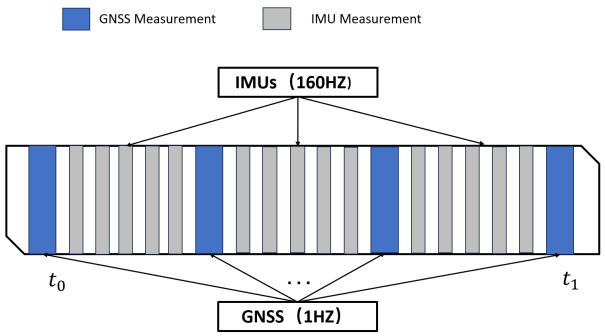


Fig. 6: Modeling of IMU and GNSS Fusion

3) Implementation details.: Accelerometers and gyroscopes operate at relatively high sample rates and necessitate high-rate processing. In contrast, GNSS and magnetometers function at relatively low sampling rates and require lower data processing rates. To replicate this configuration in our experiment, the IMUs (accelerometers, gyroscopes, and magnetometers) were

sampled at 160 Hz, while the GNSS was sampled at 1 Hz. As demonstrated in Fig 6, only one out of every 160 samples from the IMUs was provided to the fusion algorithm.

4) Evaluation Metrics: We utilize three metrics to evaluate the effectiveness of SSD comprehensively. Initially, we include two metrics that are widely used in relevant studies [40].

(1) Average Displacement Error (**ADE**): This metric measures the average deviation between the predicted and ground-truth trajectories by calculating the root mean squared error (RMSE) across all time frames. It captures the overall accuracy of the predicted trajectory compared to the actual path.

(2) Final Displacement Error (**FDE**): FDE focuses specifically on the prediction accuracy at the final time frame, quantified as the RMSE between the predicted and ground-truth positions. This metric highlights the importance of precise final positioning, which is crucial in applications such as path planning.

However, the two indicators above alone are insufficient to capture the impact of targeted attacks on UAVs. This introduces another challenge for evaluating SSD:

C2: How to evaluate SSD's impact on UAV mission.

In mission-critical scenarios, UAVs must execute precise manoeuvres at specific waypoints to ensure the successful completion of the mission. The deviation from these waypoints during flight significantly increases the likelihood of mission failure. Therefore, to address **C2**, we design the Average Per-Waypoint Displacement Error (**APDE**). The APDE is formally defined as the average of the displacement errors computed at each waypoint along the trajectory. We define APDE as follows:

$$APDE = \frac{\sum_{i=1}^{N_w} ||P_i - P_i^a||_2}{N_w} \tag{27}$$

where P_i^a and P_i refer to the predicted positions before and after the attack, respectively. N_w refers to the number of waypoints. APDE provides a more nuanced understanding of how targeted attacks impact the UAV's ability to adhere to its prescribed flight path, particularly at critical waypoints, thus enabling a more accurate assessment of the potential risks to mission success.

B. Parametric Analysis

The effectiveness and stealthiness of the attack are highly dependent on the choice of parameters. We analyze the sensitivity of SSD to various parameters by combining different values for θ, α, and ϕ across multiple scenarios (the same as Section IV-B). To measure the attack effectiveness and stealthiness, we use ADE and the maximum chi-square statistic (denoted as χ^2_{max}), respectively. The experimental results are shown in Fig 7. In positional attacks, as θ increases, χ^2_{max} exhibits a corresponding upward trend. We further observe that the rate of increase in χ^2_{max} slows as α increases. This suggests that higher values of α help mitigate the growth of χ^2_{max}. Notably, when α exceeds a critical threshold value 13, χ^2_{max} starts to stabilize, converging within a relatively narrow threshold range. Within this range, χ^2_{max} fluctuates minimally and remains largely unaffected by changes in θ, demonstrating high stability and consistency. For the velocity attack, as shown in Fig. 7b, χ^2_{max} gradually converges to about 6 when ϕ is less than 0.08. In the following experiments, we set the values of θ, α, and ϕ to 20, 11, and 0.08, respectively.

C. Ablation study

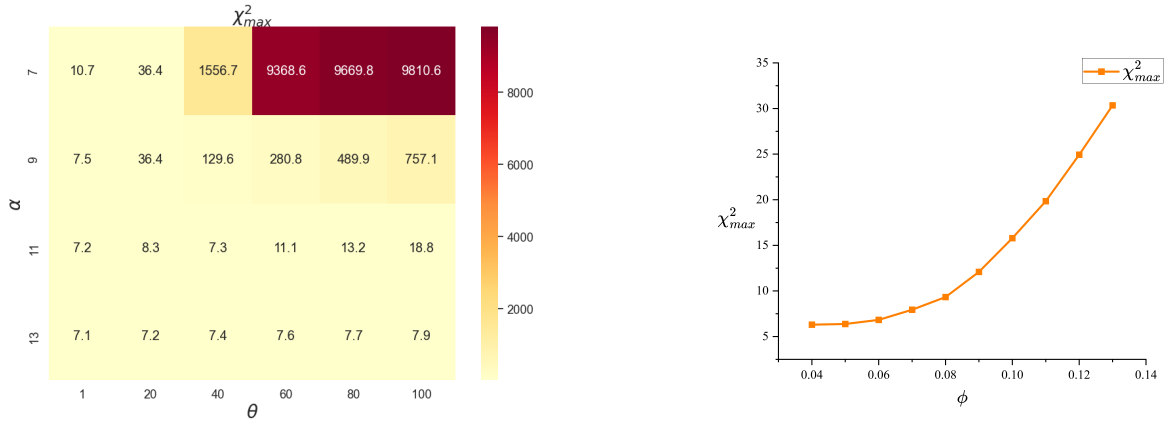
To validate the effectiveness of velocity-based and position-based attacks, we conducted an experiment where the UAV performed a 35 second flight incorporating linear and nonlinear motion states. The first 20 seconds involved uniform linear motion at a velocity of $2m/s$. Afterward, the UAV transitioned into a uniform circular motion mode, maintaining a linear velocity of $2m/s$ for the remaining 15 seconds. This flight trajectory was used for the ablation experiments.

1) Contributions of different attacks: To explore the contribution of velocity and positional attacks to overall attack effectiveness, we designed comparison experiments with three attack strategies: single position perturbation attack (SPA), single velocity perturbation attack (SVA), and combined concerted attack (CCA). The quantitative analysis of ADEs, presented in Table II, shows that both individual attacks are effective. Specifically, the ADEs for SPA and SVA are improved by 184% and 212%, respectively, compared to the baseline. When comparing the combined attack (CCA) to the single attacks, the ADE improves by 129% over SPA and 112% over SVA. This indicates that CCA not only significantly enhances attack effectiveness by leveraging the coupling effect between position and velocity but also perturbs both north-east directions simultaneously. Furthermore, the stealth assessment results in Fig. 8 show that while the CCA approach induces brief oscillations in the detection statistics, these fluctuations remain well within the acceptable threshold limits, ensuring that the attack remains stealthy.

TABLE II: Ablation Study of Different Attacks' Contribution, where N-dir denotes the north direction and E-dir denotes the east direction in the North-East direction.

	N-dir (meters)	E-dir (meters)
Baseline	1.41	0.66
SPA	2.72	0.89
SVA	2.45	2.2
CCA	2.79	2.41

2) Attack Combination: We apply the velocity attack during the linear motion state and the position attack during the nonlinear motion state and examine the impact of this combination on the attack's stealthiness. The experimental results, shown in Fig. 9, reveal that $9s$ after the attack begins, the detector quickly identifies the existence of the attack. Furthermore, even when the UAV transitions to a nonlinear motion state, the detector continues to successfully detect the attack for $5s$, despite the change in attack mode. This phenomenon occurs because the drastic velocity changes in the linear motion state cause significant fluctuations in the residuals. In contrast, the nonlinear motion state allows SSD to effectively smooth out the impact of the velocity attack, ensuring the attack remains stealthy throughout the process. This result, along with **Finding2**, further validates the rationale of the SSD framework.



(a) Positional attack

(b) Velocity attack

Fig. 7: Parameter Analysis

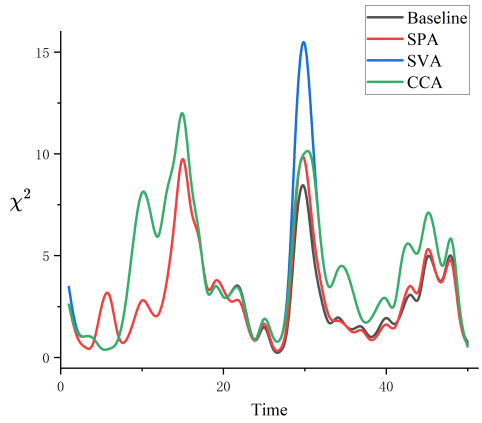


Fig. 8: Stealthiness Comparison for Attack Contribution

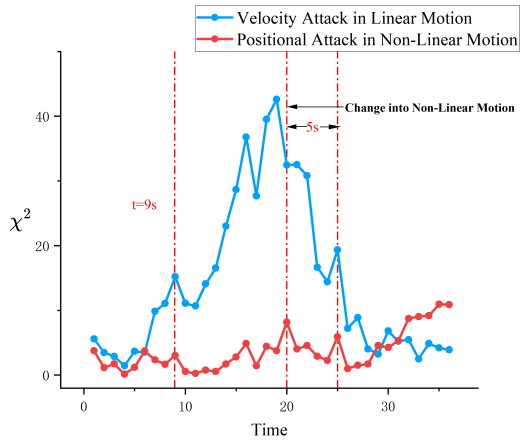
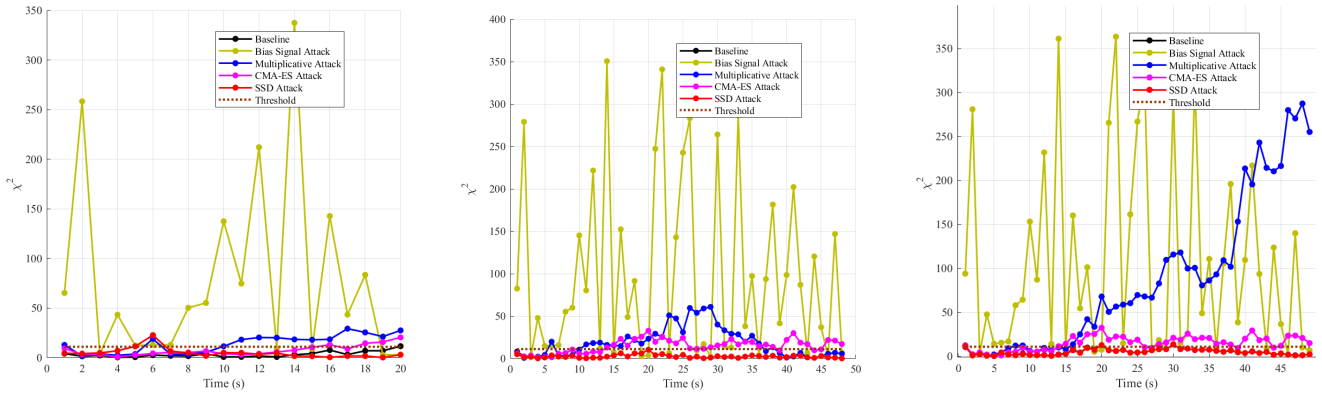


Fig. 9: Detection Statistic Comparison for Attack Combination

D. Attack Stealthiness

To validate the stealthiness, we selected a chi-square detector for attack detection. For the threshold, we selected a value corresponding to a 95% confidence level, which is 11.1. We also chose biased signal, multiplicative attacks and CMA-ES [6] for comparison and evaluated them across three mission trajectories. Based on prior research, we designed the following attack payloads: (1) GNSS positions are added with a uniform distribution U(0, 0.0005); (2) GNSS positions are scaled by a factor of 1.5; (3) As to the CMA-ES, we fix the stealthiness budget at 0.05 and constrain the injected signal norm to the range 0.01 to 0.05 so that all variants operate under the equal perturbation budget. The experimental results in Fig. 10 demonstrate that SSD successfully limits detection statistics to the threshold range and bypasses both detection methods with a carefully chosen set of attack parameters. Table III reports detection and stealth metrics under a 95% χ^2 threshold (11.1) for three trajectories. As expected, constant-bias and multiplicative spoofing are with high detection (e.g., 75.0–77.6% on Trajectory I/II)/III and large residual energy (Mean χ^2, NLC, LTW). CMA-ES [6] substantially improves stealth (e.g., 80.0% bypass on Traj2), yet our SSD method is consistently more evasive: it achieves the highest bypass (95.0/100.0/95.9% on Trajectory I/II/III) while also minimizing residuals (e.g., Mean χ^2 of 3.52/2.55/5.12 and Mean NLC of 0.790/0.620/0.106), indicating stronger detector evasion under the same perturbation budget. Notably, compared to these works [5], [41], we significantly reduced the loadings applied for the biased signal Attack. However, both detection methods were able to detect the attack effectively. It is clear that SSD exhibits strong stealthiness properties under chi-square detection methods and several motion states.

We also use two UAV-specific detectors, NLC [42] and LTW [43], as baselines for comparison. These methods are highly effective at detecting GNSS attacks, and they maintain both strong accuracy and response time, even when identifying targeted stealthy attacks. We follow the threshold in previous work [42]. Fig. 11 demonstrates that in the LTW test group, the detection statistic consistently remains below the threshold



(a) χ^2 detection of different attacks under Trajectory I (b) $\chi^2 detection$ of different attack under Trajectory II (c) $\chi^2 detection$ of different attack under Trajectory III

Fig. 10: Attack Stealthiness Evaluation

TABLE III: Attack detection and stealth metrics across three trajectories (threshold for detection: $\chi^2 = 11.1$ at 95% confidence). "Bypass" is $100\% -$ Detect. Lower is better for Detect, Mean/Max χ^2, NLC, and LTW; higher is better for Bypass.

Trajectory	Attack	Detect (%) ↓	Bypass (%) ↑	Mean χ^2	Mean NLC	Mean LTW
Trajectory I	Bias Signal	75.0	25.0	78.50	118.89	15.08
	Multiplicative	65.0	35.0	14.29	21.28	5.68
	CMA-ES	20.0	80.0	7.49	5.82	4.56
	SSD (ours)	5.0	95.0	3.52	0.79	3.48
Trajectory II	Bias Signal	77.1	22.9	98.38	343.88	17.65
	Multiplicative	58.3	41.7	19.19	109.76	7.72
	CMA-ES	66.7	33.3	14.73	76.44	6.62
	SSD (ours)	0.0	100.0	2.55	0.62	3.33
Trajectory III	Bias Signal	77.6	22.4	104.40	362.12	18.08
	Multiplicative	73.5	26.5	89.10	191.98	9.59
	CMA-ES	67.3	32.7	15.18	62.00	6.03
	SSD (ours)	4.1	95.9	5.12	0.11	3.08

boundary, indicating the superior stealth characteristics of SSD attacks. Notably, NLC statistics demonstrate the following three characters (shown in Table IV):

- In Trajectory I (pure linear motion), the detection statistic approaches the threshold at $6s$ but never exceeds the threshold.
- In Trajectory II (fully nonlinear motion), the detection statistic surpasses the threshold after $31s$ cumulative duration. The detection latency is largely increased compared to the previous work (about $0.3s$) [42].
- In Trajectory III (hybrid motion mode), no significant statistical fluctuations occur during $0 - 20s$ linear phase, with limited oscillations (peak statistic is 9.27) emerging post nonlinear component introduction at 20s.

TABLE IV: Detection Performance Comparison for NLC

	Detection Latency(s)	Peak Statistic	Detection Rate
Trajectory I	N/A	19.69	0%
Trajectory II	31s	>20	35.4%
Trajectory III	N/A	9.27	0%

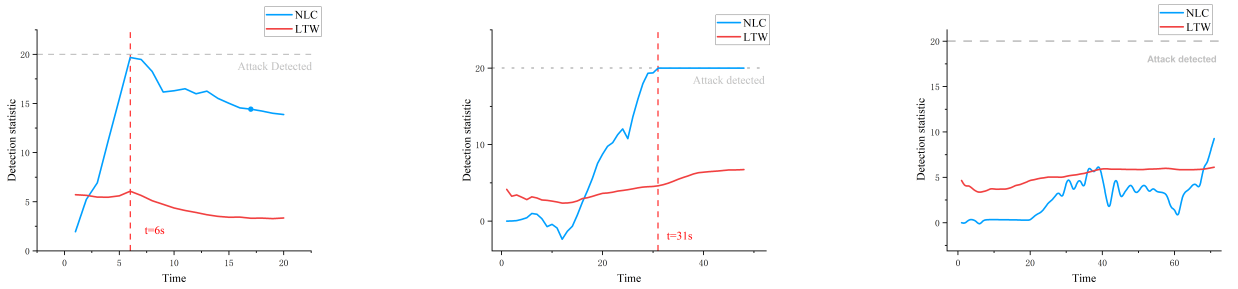
These findings demonstrate significant motion-dynamic sensitivity disparities in NLC detectors. However, when attackers combine motion-state transition strategies, positional attacks can realize residual normalization during linear phases and reduce detection sensitivity velocity attacks to counteract statistical fluctuations from nonlinear components. These two methods make SSD show greater stealthiness in hybrid motion missions.

E. Attack Effectiveness

We start by determining the optimal combination of Q_k and R_k for each scenario through offline learning. With these optimal values, we proceed to validate the effectiveness of the SSD. For each combination of INS and trajectory, the UAV is tasked with following the designated path to complete a full mission, while the SSD is deployed to attack the flight. Table V presents the changes in each metric before and after the attack. On average, ADE/FDE is increased by 425%/591%. The lateral (N)/longitude (E) deviation reaches 3.54/3.46 meters. We will analyze the factor based on the experiment on three scenarios.

1) Different Scenarios: In terms of scenarios, the SSD shows a greater increase in positioning error in the purely linear motion state (Fig. 12a) compared to the purely nonlinear state (Fig. 12b). This difference arises from the time-varying nature of the velocity vector in the nonlinear state, which leads to attenuation of the indirect positional interference caused by the velocity perturbation $G(t_i, a_i^d; \phi)$. However, experiments with Section VI-D demonstrate that a velocity attack can

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2025.3646036

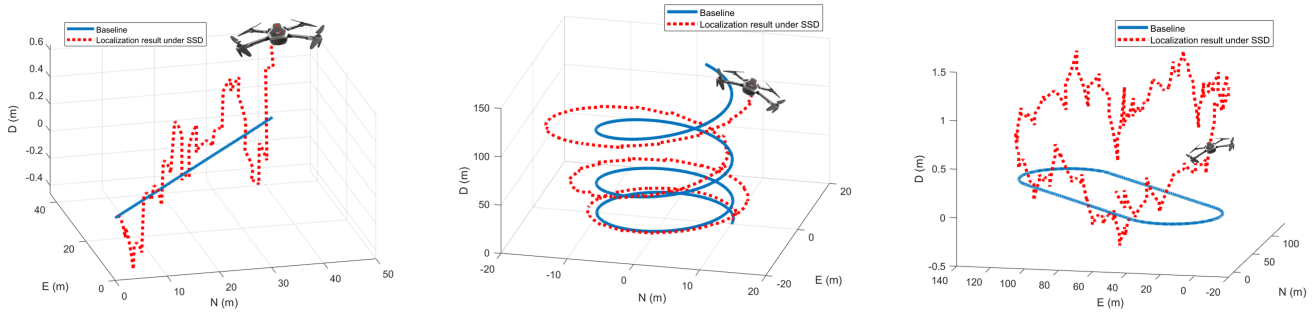


(a) Detection statistic in Trajectory I (b) Detection statistic in Trajectory II (c) Detection statistic in Trajectory III

Fig. 11: Detection Statistic Comparison for NLC and LTW under SSD.

TABLE V: Attack Effectiveness

Model	Scenario	Duration	ADE Normal/Attack (meters)	FDE Normal/Attack (meters)	APDE Normal/Attack (meters)
ECL EKF2	Trajectory I	20s	0.95/(5.68±1.17)	1.80/(5.89±1.05)	0.89/(5.66±1.15)
CD-EKF	Trajectory I		1.62/(4.64±0.80)	1.49/(7.54±1.59)	1.81/(4.92±0.87)
ECL EKF2	Trajectory II	48s	1.31/(5.30±0.47)	1.78/(4.36±0.81)	1.48/(5.45±0.51)
CD-EKF	Trajectory II		1.39/(3.31±0.35)	1.99/(5.05±1.07)	2.35/(4.05±0.60)
ECL EKF2	Trajectory III	71s	1.46/(2.48±1.04)	0.95/(10.47±2.17)	1.86/(6.49±1.30)
CD-EKF	Trajectory III		1.86/(5.53±0.38)	1.34/(8.13±0.85)	1.66/(6.51±0.27)



(a) Attack visualization in Trajectory I (b) Attack visualization in Trajectory II (c) Attack visualization in Trajectory III

Fig. 12: Attack Visualization

still ensure the fulfillment of the stealthy precondition. When the mission involves mixed kinematic modes, the combination of velocity and positional attacks results in an additive interaction, as shown in Table V. The baseline increase is 366.44% for ADE and 1102.11% for FDE, confirming that SSD enhances attack effectiveness synergistically. Next, we will quantitatively analyze the impact of this state on mission completion rates.

2) Impact on Mission: When the UAV's trajectory deviation exceeds the tolerance range, the mission completion rate shows a clear decline. Because acceptable trajectory error depends on the application, we evaluate practical impact using representative error budgets: (i) inspection/close-proximity tasks (e.g., power-line, substation, bridge) require lateral deviation $\tau_{\text{inspect}} \sim 1$ m; (ii) area-survey/coverage tasks (e.g., crop survey) tolerate $\tau_{\text{survey}} \sim 3$ m. Mission failure for path-keeping is declared if APDE $> \tau_{\text{task}}$; terminal failure (return/landing/last waypoint) if FDE $> \tau_{\text{task}}$. Given the reported (μ, σ) for each metric under attack (Table V), we estimate

$$P_{\text{fail}} = \mathbb{P}(\text{APDE} > \tau_{\text{task}}) \approx 1 - \Phi\left(\frac{\tau_{\text{task}} - \mu_{\text{APDE}}}{\sigma_{\text{APDE}}}\right) \quad (28)$$

$$P_{\text{fail@terminal}} = \mathbb{P}(\text{FDE} > \tau_{\text{task}}) \approx 1 - \Phi\left(\frac{\tau_{\text{task}} - \mu_{\text{FDE}}}{\sigma_{\text{FDE}}}\right) \quad (29)$$

where $\Phi(\cdot)$ is the standard normal CDF.

TABLE VI: Mission-failure probability (%) under task-specific budgets.

Model	Scenario	Inspection Proximity Task	Area-Survey Task	Terminal Failure
ECL EKF2	Trajetory I	100.0	98.96	99.70
	Trajetory II	100.0	100.0	95.34
	Trajetory III	100.0	99.64	100.0
CD-EKF	Trajetory I	100.0	98.63	99.79
	Trajetory II	100.0	95.99	97.23
	Trajetory III	100.0	100.0	100.0

Using task-specific budgets (1 m for inspection and 3 m for survey), SSD increases mission-failure probability to

99.0–100% at 1 m and 95.99–100% at 3 m across all scenarios (Table VI); Scenario III also yields the largest terminal error (FDE 10.47 ± 2.17 m), implying $\geq 99.9\%$ failure at a 3 m terminal budget (Fig. 12c).

3) Attack Transferability: Different INS architectures manage noise and system uncertainty in distinct ways, potentially impacting the effectiveness of SSD. To assess the cross-system adaptability of the SSD, we conducted validation experiments using CD-EKF and ECL EKF2. CD-EKF relies on continuous-time prediction with discrete-time updates, enhancing its adaptability to errors in nonlinear motion states and making its short-term error estimation more robust. However, SSD effectively exploits CD-EKF's sensitivity to state uncertainty by inducing motion instability, thereby continuously disrupting navigation accuracy. Experimental results show that under the CD-EKF system, SSD achieves an average improvement of 304.48% in ADE, 473.57% in FDE, and 305.87% in APDE (see Table V). These findings confirm SSD's generalizability across EKF-based navigation frameworks.

F. Sensitivity & Rationale for the exponential G and log-2 F choice

We conducted a controlled sensitivity study in which we held the SSD parameters and runtime constant and varied only the shape families used for the position bias $G(\cdot)$ and velocity scale $F(\cdot)$. Impact was summarised by the chi-square test statistic (higher is better for effect on the state estimate) and stealth by the detection rate under a fixed chi-square gate of 11.1 (lower is better). As shown in Table VII, exponential position bias paired with log-2 velocity scaling ("exp–log") lies on the Pareto frontier, achieving strong impact ($\chi^2 \approx 5.45$) at a low detection rate (4.1%), while alternatives that slow down too much (e.g., exp–sqrt, exp–exp) under-deliver impact, and those that amplify too aggressively in the velocity channel (e.g., log–log, sqrt–log, log10–log) are trivially caught 100% of the time. Changing the log base confirms the sensitivity: exp–log10 raises detectability (26.5%) without compensating impact. A linear–log variant produces an outlier χ^2 magnitude but relies on large, non-smooth transients confined to short windows; such profiles are operationally fragile (they violate smoothness/feasibility and would be flagged by standard change–point logic even when rate metrics remain low). In contrast, exponential bias naturally matches the stable mode of the estimator, yielding lasting displacement while its per-step innovation rapidly subsides, and log-2 produces a slow, uniform multiplicative drift in velocity—small enough per update to slip past the gate but persistent enough to integrate into position error. Together, the empirical sensitivity and the filter-dynamics argument justify our choice of exponential G and log-2 F as maximising the stealth–impact trade-off.

G. Robustness to Modeling and Timing errors

We quantified SSD detectability under joint variations of filter noise settings, timing offsets, and latency. Keeping the SSD pattern identical to the main study, we swept ±20% multiplicative changes on both the process and measurement noise parameters used by the EKF, injected 5–20 ms mis-timestamping between the GNSS measurement and the reference state used to form innovations, and emulated 0–200 ms system latency by buffering GNSS before fusion. Detection was assessed by the EKF's GNSS innovation χ^2 test at 99% confidence with degrees of freedom set by the residual dimension. Results show: (i) As shown in Figure 13a, ±20% Q/R perturbations produced a monotonic change in innovation energy (mean χ^2 from 5.11 to 3.41; max from 14.52 to 9.66) but detection remained 0.0%; (ii) As shown in Figure 13b, 5–20 ms timing offsets had negligible impact (mean χ^2 4.09–4.13); and (iii) As shown in Figure 13c, latency slightly increased innovation magnitude and yielded a modest 5.0% detection for $\geq$10 ms while RMS position error was 1.81–2.03 m over 0–200 ms. Under the criterion that detection must remain $\geq$ 95%, the maximum tolerable latency in this configuration is 0 ms, indicating that with the current SSD amplitude and threshold the attack remains largely stealthy; raising sensitivity (e.g., lower confidence level or augmented residual set) would increase detection but is beyond the scope of this robustness check.

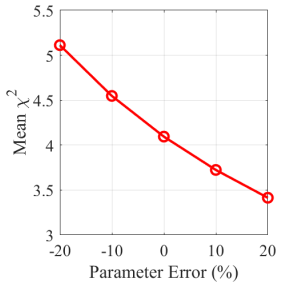
TABLE VII: Sensitivity across shape combinations (χ^2 gate = 11.1; lower detection is better). "log" denotes base-2 unless otherwise specified; all runs share identical SSD parameters and gating. The "linear–log" outlier attains extreme χ^2 by injecting large, brief transients; despite a modest average rate, such spikes are not compatible with sustained stealth and violate standard feasibility/smoothness constraints, hence are not considered practical for an attacker seeking persistent, undetected bias.

Combination	χ^2 (impact)	Detection rate (%)
exp–log	**5.45**	**4.1**
exp–log10	3.38	26.5
exp–sqrt	1.44	10.2
exp–exp	1.75	12.2
log–log	832.01	100.0
sqrt–log	1224.91	100.0
log10–log	362.00	100.0
linear–log	885,637.18	8.2

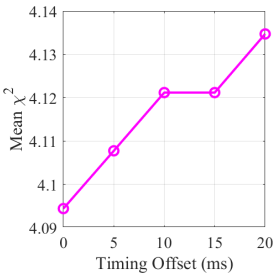
H. Validation of SITL Against u-blox Flight Data

To address the concern regarding the fidelity of PX4-SITL sensor simulation, we incorporated actual flight data recorded from a u-blox GNSS receiver and conducted a detailed comparison with SITL outputs. This approach enabled a direct, quantitative comparison of statistical properties, noise spectra, and temporal stability between the hardware logs and the SITL simulation. Figure 14a presents the Allan variance comparison between the u-blox measurements and SITL outputs. The two curves are in close agreement across integration times spanning 10^{-2} to 10 seconds, confirming that SITL reproduces both short-term stability and random-walk characteristics of the actual GNSS. This demonstrates that SITL captures the essential temporal noise dynamics relevant for navigation and sensor fusion testing.

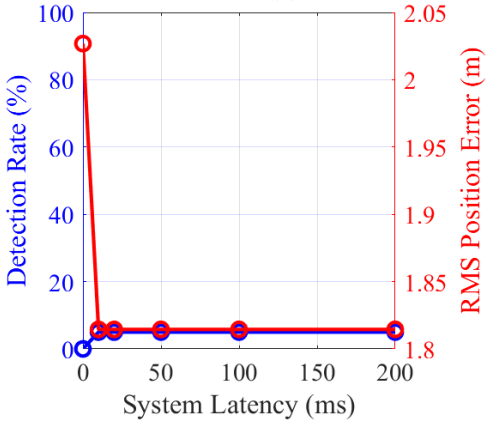
Figure 14b shows the power spectral density (PSD) of acceleration for both u-blox and SITL data. The results highlight



(a) Filter Noise Validation

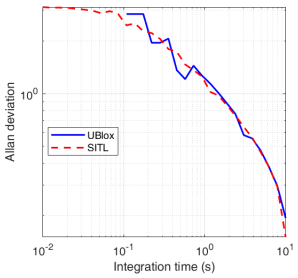


(b) Time Offset Validation

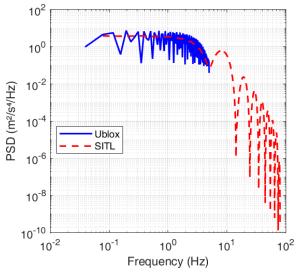


(c) Attack Latency Validation

Fig. 13: Attack Visualization



(a) Allan Variance



(b) PSD of Acceleration

Fig. 14: Validation of SITL Against u-blox Flight Data

that SITL preserves the expected flat white-noise characteristics within the operational bandwidth and closely matches the spectral level of the hardware up to the GNSS update rate. Minor discrepancies appear at higher frequencies, where SITL exhibits sharper roll-off due to deterministic sampling and the absence of receiver tracking dynamics; however, these deviations occur outside the frequency band of interest for navigation algorithms.

Together, these results provide strong empirical evidence that PX4-SITL, when configured with 160 Hz IMU and 1 Hz GNSS, is a reliable proxy for hardware measurements. The Allan variance and PSD analyses confirm that SITL replicates both the magnitude and distribution of noise consistent with u-blox specifications, while the absence of secondary effects such as RF multipath or clock drift has a negligible impact on overall accuracy ($\leq$5% of the error budget).

VII. Discussion

This study reveals the coupling mechanism between UAV motion dynamics and cyber attacks' effectiveness through systematic empirical analysis. Experimental data indicate that changes in motion state can significantly enhance the success rate of attacks. While the SSD approach demonstrates clear advantages, its engineering implementation faces two major challenges: **Dependency on A Priori Knowledge.** The effectiveness is highly contingent upon the real-time accuracy of the object detection and tracking system. However, the existing YOLOv5 architecture, for instance, exhibits exponential decay in the Intersection over Union (IoU) metric over time in dynamic target tracking scenarios. This results in a tracking failure probability exceeding 73% after 60 seconds of continuous locking. **Energy-Concealment Trade-off Paradox.** While the sustained attack mode can maintain a stealthiness threshold, it leads to a non-linear increase in energy consumption on the attacking end. This escalation doesn't align with the requirements in real-world mission scenarios.

In addition, a key assumption of the SSD is that the Q_k and R_k are preset offline. This is a common practice in current engineering applications. From a theoretical standpoint, adaptive Q_k and R_k values can modify the system's sensitivity to attacks, potentially enabling the mitigation of such attacks. This insight provides a constructive direction for defending SSD. We observed that increasing Q_k and decreasing R_k could reduce the system's sensitivity to attacks. However, this adjustment comes at the cost of a decrease in localization accuracy. While it is possible to improve positioning accuracy by increasing R_k and decreasing Q_k, this also makes the system more vulnerable to attacks. Consequently, a dynamic strategy for adjusting Q_k and R_k is crucial. This can be achieved through optimization methods or reinforcement learning, which can fine-tune these parameters in real time, balancing between accuracy and security.

VIII. Conclusion

In this paper, we investigate cybersecurity threats of UAV route planning under different motion states. We assess the effectiveness of GNSS attacks under various motion states through theoretical and experimental analyses. Our findings reveal that INS is more vulnerable during maneuvering than in linear flight. Based on this insight, we introduce SSD, a novel state-based stealthy backdoor attack, which strategically combines GNSS velocity and position attacks to exploit this vulnerability. We conducted extensive experiments, and the results show that SSD demonstrates superior effectiveness and stealthiness compared with previous methods. We hope that this work will inspire INS designers and developers to prioritize code security and implement robust dynamic defenses.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (No.62233014, No.62103330), and the Innovation Foundation for Doctor Dissertation of Northwestern Polytechnical University (CX2023023).

REFERENCES

[1] Zhaoxuan Wang, Yang Li, Shihao Wu, Yuan Zhou, Libin Yang, Yuan Xu, Tianwei Zhang, and Quan Pan. A survey on cybersecurity attacks and defenses for unmanned aerial systems. *Journal of Systems Architecture*, 138:102870, 2023.

[2] Xiaomin Wei, Jianfeng Ma, and Cong Sun. A survey on security of unmanned aerial vehicle systems: Attacks and countermeasures. *IEEE Internet of Things Journal*, 2024.

[3] Taifeng Liu, Chao Yang, Xinjing Liu, Ruidong Han, and Jianfeng Ma. Rpau: Fooling the eyes of uavs via physical adversarial patches. *IEEE Transactions on Intelligent Transportation Systems*, 25(3):2586–2598, 2023.

[4] Woohyun Kim and Jiwon Seo. Low-cost software-defined gps simulator with the capability of time synchronization. In *2018 18th International Conference on Control, Automation and Systems (ICCAS)*, pages 1087–1090. IEEE, 2018.

[5] Wenbing Tang, Yuan Zhou, Haiying Sun, Yuhong Zhang, Yang Liu, Zuohua Ding, Jing Liu, and Jifeng He. Gan-based robust motion planning for mobile robots against localization attacks. *IEEE Robotics and Automation Letters*, 8(3):1603–1610, 2023.

[6] Amir Khazraei, Haocheng Meng, and Miroslav Pajic. Black-box stealthy gps attacks on unmanned aerial vehicles. *arXiv preprint arXiv:2409.11405*, 2024.

[7] Xiaomeng Ma, Meiguo Gao, Yangguang Zhao, and Mohan Yu. A novel navigation spoofing algorithm for uav based on gps/ins-integrated navigation. *IEEE Transactions on Vehicular Technology*, 2024.

[8] Shihao Wu, Yang Li, Zhaoxuan Wang, Zheng Tan, and Quan Pan. A highly interpretable framework for generic low-cost uav attack detection. *IEEE Sensors Journal*, 23(7):7288–7300, 2023.

[9] Junjie Shen, Jun Yeon Won, Zeyuan Chen, and Qi Alfred Chen. Drift with devil: Security of multi-sensor fusion based localization in high-level autonomous driving under gps spoofing. In *Proceedings of the 29th USENIX Conference on Security Symposium*, pages 931–948, 2020.

[10] Andrew J Kerns, Daniel P Shepard, Jahshan A Bhatti, and Todd E Humphreys. Unmanned aircraft capture and control via gps spoofing. *Journal of Field Robotics*, 31(4):617–636, 2014.

[11] Alexandre Vervisch-Picois, Nel Samama, and Thierry Taillandier-Loize. Influence of gnss spoofing on drone in automatic flight mode. In *ITSNT 2017: 4th International Symposium of Navigation and Timing*, pages 1–9. Ecole nationale de l'aviation civile, 2017.

[12] Kangjie Chen, Xiaoxuan Lou, Guowen Xu, Jiwei Li, and Tianwei Zhang. Clean-image backdoor: Attacking multi-label models with poisoned labels only. In *The Eleventh International Conference on Learning Representations*, 2022.

[13] Kangjie Chen, Yuxian Meng, Xiaofei Sun, Shangwei Guo, Tianwei Zhang, Jiwei Li, and Chun Fan. Badpre: Task-agnostic backdoor attacks to pre-trained nlp foundation models. In *International Conference on Learning Representations*, 2022s.

[14] Mohamed Abdel-Basset, Reda Mohamed, Karam M Sallam, Ibrahim M Hezam, Kumudu Munasinghe, and Abbas Jamalipour. A multiobjective optimization algorithm for safety and optimality of 3-d route planning in uav. *IEEE Transactions on Aerospace and Electronic Systems*, 2024.

[15] Honghui Qi and John B Moore. Direct kalman filtering approach for gps/ins integration. *IEEE Transactions on Aerospace and Electronic Systems*, 38(2):687–693, 2002.

[16] Himilcon Carvalho, Pierre Del Moral, André Monin, and Gérard Salut. Optimal nonlinear filtering in gps/ins integration. *IEEE Transactions on Aerospace and Electronic Systems*, 33(3):835–850, 1997.

[17] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1052–1067, 2007.

[18] Christian Forster, Zichao Zhang, Michael Gassner, Manuel Werlberger, and Davide Scaramuzza. Svo: Semidirect visual odometry for monocular and multicamera systems. *IEEE Transactions on Robotics*, 33(2):249–265, 2016.

[19] Ji Zhang, Sanjiv Singh, et al. Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and systems*, volume 2, pages 1–9. Berkeley, CA, 2014.

[20] Thien-Minh Nguyen, Shenghai Yuan, Muqing Cao, Lyu Yang, Thien Hoang Nguyen, and Lihua Xie. Miliom: Tightly coupled multi-input lidar-inertia odometry and mapping. *IEEE Robotics and Automation Letters*, 6(3):5573–5580, 2021.

[21] Tong Qin, Peiliang Li, and Shaojie Shen. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE transactions on robotics*, 34(4):1004–1020, 2018.

[22] Kevin Eckenhoff, Patrick Geneva, and Guoquan Huang. Mimc-vins: A versatile and resilient multi-imu multi-camera visual-inertial navigation system. *IEEE Transactions on Robotics*, 37(5):1360–1380, 2021.

[23] Wei Xu, Yixi Cai, Dongjiao He, Jiarong Lin, and Fu Zhang. Fast-lio2: Fast direct lidar-inertial odometry. *IEEE Transactions on Robotics*, 38(4):2053–2073, 2022.

[24] Daojing He, Yinrong Qiao, Shiqing Chen, Xiao Du, Wenjie Chen, Sencun Zhu, and Mohsen Guizani. A friendly and low-cost technique for capturing non-cooperative civilian unmanned aerial vehicles. *IEEE Network*, 33(2):146–151, 2018.

[25] Marco Ceccato, Francesco Formaggio, Nicola Laurenti, and Stefano Tomasin. Generalized likelihood ratio test for gnss spoofing detection in devices with imu. *IEEE Transactions on Information Forensics and Security*, 16:3496–3509, 2021.

[26] Jason Bonior, Philip Evans, Greg Sheets, John Paul Jones, Toby Flynn, Lori Ross O'Neil, William Hutton, Richard Pratt, and Thomas Carroll. Implementation of a wireless time distribution testbed protected with quantum key distribution. In *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–6. IEEE, 2017.

[27] Chen Liang, Meixia Miao, Jianfeng Ma, Hongyan Yan, Qun Zhang, Xinghua Li, and Teng Li. Detection of gps spoofing attack on unmanned aerial vehicle system. In *Machine Learning for Cyber Security: Second International Conference, ML4CS 2019, Xi'an, China, September 19-21, 2019, Proceedings 2*, pages 123–139. Springer, 2019.

[28] Xingshuo Han, Yuan Zhou, Kangjie Chen, Han Qiu, Meikang Qiu, Yang Liu, and Tianwei Zhang. Ads-lead: Lifelong anomaly detection in autonomous driving systems. *IEEE Transactions on Intelligent Transportation Systems*, 24(1):1039–1051, 2022.

[29] Tianci Yang and Chen Lv. A secure sensor fusion framework for connected and automated vehicles under sensor attacks. *IEEE Internet of Things Journal*, 9(22):22357–22365, 2021.

[30] Sasi Rahul. Drone attacks: How i hijacked a drone, 2015.

[31] Ionut Ilascu. DJI Drone Flight Logs, Photos and Videos Exposed to Unauthorized Access. https://www.bleepingcomputer.com/news/security/dji-drone-flight-logs-photos-and-videos-exposed-to-unauthorized-access/.

[32] Yuan Xu, Gelei Deng, Tianwei Zhang, Han Qiu, and Yungang Bao. Novel denial-of-service attacks against cloud-based multi-robot systems. *Information Sciences*, 576:329–344, 2021.

[33] Ardupilo. https://ardupilot.org/ardupilot/.

[34] PX4. https://github.com/PX4/PX4-Autopilot/.

[35] Markus Schreiber, Hendrik Königshof, André-Marcel Hellmund, and Christoph Stiller. Vehicle localization with tightly coupled gnss and visual odometry. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 858–863. IEEE, 2016.

[36] Robert Piché. Online tests of kalman filter consistency. *International Journal of Adaptive Control and Signal Processing*, 30(1):115–124, 2016.

[37] Xingshuo Han, Guowen Xu, Yuan Zhou, Xuehuan Yang, Jiwei Li, and Tianwei Zhang. Physical backdoor attacks to lane detection systems in autonomous driving. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 2957–2968, 2022.

[38] Xingshuo Han, Yutong Wu, Qingjie Zhang, Yuan Zhou, Yuan Xu, Han Qiu, Guowen Xu, and Tianwei Zhang. Backdooring multimodal learning. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 3385–3403. IEEE, 2024.

[39] ECL EKF2. https://github.com/PX4/PX4-ECL/.

[40] Qingzhao Zhang, Shengtuo Hu, Jiachen Sun, Qi Alfred Chen, and Z Morley Mao. On adversarial robustness of trajectory prediction for autonomous vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15159–15168, 2022.

[41] Fan Fei, Zhan Tu, Dongyan Xu, and Xinyan Deng. Learn-to-recover: Retrofitting uavs with reinforcement learning-assisted flight control under cyber-physical attacks. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7358–7364. IEEE, 2020.

[42] Raul Quinonez, Jairo Giraldo, Luis Salazar, Erick Bauman, Alvaro Cardenas, and Zhiqiang Lin. {SAVIOR}: Securing autonomous vehicles with robust physical invariants. In *29th USENIX security symposium (USENIX Security 20)*, pages 895–912, 2020.

[43] Hongjun Choi, Wen-Chuan Lee, Yousra Aafer, Fan Fei, Zhan Tu, Xiangyu Zhang, Dongyan Xu, and Xinyan Deng. Detecting attacks against robotic vehicles: A control invariant approach. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 801–816, 2018.