Tightly Coupled GNSS/INS Integration Spoofing Detection Algorithm based on Innovation Rate Optimization and Robust Estimation

YE KE1,2, ZHIWEI LV1, CHAO ZHANG1, XU DENG1, WENLONG ZHOU1, AND DEBIAO SONG1

1PLA Strategic Support Force Information Engineering University, Zhengzhou 450001, China
2Unit 31618, People’s Liberation Army, Fuzhou 350000, China

Corresponding author: Zhiwei Lv (lvzhiwei@sina.com).

This work was supported in part by the State Key Laboratory of Geo-Information Engineering under NO.SKLGIE2020-Z-2-1, in part by the National Natural Science Foundation of China under Grant 41804035 and Grant 42174036.

ABSTRACT The spoofing detection algorithm for a global navigation satellite system/inertial navigation system (GNSS/INS) integrated navigation system based on the innovation rate and robust estimation has extensive or invalid detection times, high missed detection rates, and false alarm rates. This study addresses these limitations by proposing a tightly coupled GNSS/INS integration spoofing detection algorithm based on innovation rate optimization and robust estimation. The proposed algorithm improved the normalized innovation of a small step or slow-growing ramp, thereby optimizing its innovation rate test statistics. The proposed approach also reduces the spoofing effect on the innovation rate by adaptively adjusting a gain matrix using robust estimation, thus improving the detection ability further. Simulation results show that the detection time of the proposed algorithm is reduced by 51.9% on average when dealing with small step or slow-growing ramp spoofing. Moreover, the missed detection rate decreased by 58% on average, and the false alarm rate remained at approximately zero. The proposed algorithm has the advantages of fast detection and good performance and is suitable for spoofing detection in unmanned aerial vehicle applications of GNSS/INS integrated navigation systems.

INDEX TERMS Tightly coupled GNSS/INS integration, Spoofing detection, Innovation rate optimization, Robust estimation

I. INTRODUCTION

A global navigation satellite system (GNSS) and an inertial navigation system (INS) have complementary error characteristics [1]. A GNSS can provide a global all-weather continuous position, velocity, and time service [2]. In contrast, an INS affords advantages such as independence, continuous operation, and short-term anti-spoofing ability. Therefore, a GNSS/INS integrated navigation system manages increased redundancy and reliability. However, due to the low power of the GNSS signal and the open structure, the GNSS service is easily affected by spoofing interference [3]. Spoofing interference implies that a spoofer generates spoofed signals similar to the authentic signals (or forwards authentic signals) to deceive the target receiver, thereby forcing it to generate an error and potentially dangerous information [4]. Some typical spoofing cases show that hackers deceive and capture GNSS signals to control sensors such as intelligent driving cars [5], yachts, and unmanned aerial vehicles (UAVs) [6], which affects trajectory planning schemes [7], causing potential dangers. In an integrated GNSS/INS system, the GNSS module locks the spoofed signal and outputs incorrect information. This affects the Kalman filter used to measure the estimated value of the state error in the update phase, outputting incorrect navigation results. Additionally, the estimated value of the incorrect state error is fed back to the INS through information fusion, thereby further affecting the integrated GNSS/INS system [8]. Therefore, performing real-time and accurate spoofing detection is required to ensure the reliability and integrity of the integrated navigation system.

The spoofing detection algorithm of the GNSS/INS integrated navigation system is primarily based on an innovation vector as a test statistic and adopts a binary hypothesis test method. Typical methods include the chi-square test based on innovation or residuals [9], [10],
autonomous integrity monitored extrapolation (AIME) [11], extended receiver autonomous integrity monitoring (ERAIM) [12], multiple solution separation (MSS) [13], and innovation rate [14]. The chi-square detection method based on innovation uses Kalman filter innovation as detection statistics, possessing advantages of low cost, high efficiency, and less computation. This approach is broadly used as a detection method. These methods can be divided into the “snapshot method” and “sequential method” [15]. The “snapshot method” is a test statistic composed of an innovation vector and its covariance matrix at the current moment, which is suitable for detecting step spoofing interference. In contrast, the “sequential method” implies that all innovation vectors and their covariance matrices from a certain time in the past to the current time constitute the test statistics, which is suitable for detecting ramp spoofing interference [16]. However, owing to the effect of spoofing interference, the GNSS input observation has an error effects on the innovation test statistics of the Kalman filter output. This leads to a decrease in the sensitivity of this method regarding spoofing interference, resulting in long detection time problems, high false alarm rate, and missed alarm rate [17]. The challenges of spoofing interference detection in GNSS/INS integrated navigation systems are related to the small step (or slow growth ramp) spoofing detection delay and closed-loop correction feedback mechanism [18].

Bhatti et al. [14] proposed that the innovation rate should be used to judge whether the GNSS measured value was abnormal. Subsequently, the Kalman filter should be used to estimate the normalized innovation rate in real time. The detection time was 110 s when a single channel was affected by a slow-growing interference of 0.1 m/s. However, this method should be combined with AIME. Wang et al. [18] improved the test statistics of the “snapshot method” and “sequential method.” In particular, the detection time was 28 s for spoofing interference with a small step of 5 m. Moreover, the detection time was 65 s for a spoofing interference with the slow growth of 0.1 m/s. Xu et al. [19] proposed the Multipath Estimation Delay Lock Loop (MEDLL) spoofing signal detection mechanism, which successfully detected and identified 2 m/s ramp spoofing. Nevertheless, its ramp slope was 2 m/s, which was challenging to apply to 0.1 m/s slow-growing ramp spoofing interference. The three methods mentioned above only compared the detection time but did not compare the missed detection and false alarm rates. Thus, explaining the advantages and disadvantages of the detection performance of these algorithms was difficult.

Another method used to reduce the effect of spoofing interference and improve the reliability of the integrated navigation system was a robust estimation. Thus, this approach was used to solve the problem of the closed-loop correction feedback mechanism of integrated navigation. In particular, an improved detection algorithm based on robust estimation and the “detection window” was proposed in [20]. Its core idea was to select two suitable thresholds to calculate the weight factor, and adaptively adjust the measurement noise covariance matrix to reduce the weight of the deceived interference measurement value, adaptively adjusting the gain matrix. When a single channel was subjected to a 0.5 m/s ramp interference, the improved algorithm reduced the detection time by 10 s and the missed detection rate by 9% compared with those of the traditional algorithm. In addition, Zhang et al. [8] proposed a robust estimation detection algorithm for innovation rate, which effectively suppressed the effect of spoofing interference on the state vectors and improved the data utilization rate and algorithm reliability. Moreover, this algorithm maintained the missed detection and false alarm rates within 4% in a 0.1 m/s slow-growing ramp spoofing interference in a single channel. However, the detection time of these two algorithms was extensive (or even ineffective) for slow-growing ramp spoofing interference, especially for spoofing interferences with a slope less than 0.1 m/s. In the recent five years, some scholars studied spoofing detection algorithms such as neural networks [21] and support vector machines [17]. However, the calculation was complex, the compatibility was weak, and cost was high.

To address the above limitations of spoofing detection, this study first analyzed the spoofing interference model at a satellite navigation signal level. Subsequently, the spoofing interference model for a GNSS/INS tightly coupled system measurement level was developed, focusing on analyzing the added value of measurement pseudorange of the satellite channel and establishing the calculation models of step spoofing and ramp spoofing. Meanwhile, the effect of spoofing interference on the innovation vector of the Kalman filter was analyzed. This reduced the spoofing detection performance. The contribution of this study was to overcome the limitations of the traditional spoofing detection algorithm based on innovation rate robust estimation regarding extensive or invalid detection time, high missed detection rate, and high false alarm rate. In particular, a GNSS/INS tightly coupled system spoofing detection algorithm based on innovation rate optimization and robust estimation was proposed. Finally, the effectiveness, rationality, and feasibility of the proposed algorithm were verified by simulations.

II. SPOOFING INTERFERENCE MODEL OF GNSS/INS INTEGRATED NAVIGATION SYSTEM AND ITS INFLUENCE ANALYSIS

A. SPOOFING INTERFERENCE MODEL

First, the spoofing model at a satellite navigation signal level was analyzed [22] to develop the spoofing interference simulation environment of the GNSS/INS integrated navigation system and simulate the spoofing interference model at aGNSS measurement level. The spoofing interference of the target receiver is shown in Fig. 1.

A raw pseudorange of the $i$-th satellite $R^{(i)}$ at time $t$ is expressed as follows:

$$R^{(i)} = c \tau^{(i)} + c((t + \delta t_i) - (t + \delta \tau^{(i)})).$$  \hspace{1cm} (1)
By subtracting the authentic and the spoofed pseudorange measurements, the pseudorange added value, $s(t)$, after successful spoofing can be obtained as follows:

$$L_s^{(i)}(t) - L_a^{(i)}(t) = \begin{cases} s(t), & t \geq t_{lock} \\ 0, & t < t_{lock} \end{cases},$$

where $t_{lock}$ is the time when the spoofing signal locks the tracking loop of the target receiver so that $s(t)$ is the pseudorange added value of the spoofing interference, $a$ is the slope, and $a(t-t_{lock})+b$ is the pseudorange deviation between the spoofed and authentic pseudorange. Thus, two methods can be used to develop the spoofing interference model at the measurement level: 1) $a \neq 0$ and $b = 0$ represent step spoofing; 2) $a = 0$ and $b \neq 0$ represent ramp spoofing.

**B. SPOOFING INTERFERENCE IMPACT ANALYSIS**

A tightly integrated navigation system uses the GNSS pseudoranges and pseudorange rates as inputs. In a closed-loop correction, each filter iteration feeds back the estimated position, velocity, and attitude errors to the INS processor to correct the INS solution. The 17-dimensional state vector of the error state extended Kalman filter (EKF) is $X$ expressed as in [23] as follows:

$$X = \begin{bmatrix} \hat{\phi} ; \hat{\delta v} ; \hat{\delta r} ; \hat{b}_a ; \hat{b}_g ; \hat{b}^{\alpha \alpha} ; \hat{b}^{\gamma \gamma} \end{bmatrix}^T.$$  \hspace{1cm} (10)

In (10), $\hat{\phi}$, $\hat{\delta v}$, and $\hat{\delta r}$ are the attitude, velocity and position vectors of the INS estimation error, respectively. $b_a$ and $b_g$ are the accelerometer and gyro biases of the inertial sensor, respectively, and $b^{\alpha \alpha}$ and $b^{\gamma \gamma}$ are the GNSS clock error and clock drift, respectively. The symbol “$\wedge$” denotes the estimated value, the superscript “$-$” denotes the prior estimate, and the symbol “+$” denotes the posterior estimate.

Let $Z_k$ be the observation vector differing from the GNSS observation value and the INS prediction value, $H_k$ be the observation matrix, $\hat{X}_k$ be the prior estimation state vector, $P_k$ be the prior estimation state covariance matrix, and $R_k$ be the observation noise covariance matrix. Then, the observation $Z_k$, innovation vector $r_k$ and their covariance matrix $V_k$ respectively can be expressed as in [24]:

$$Z_k = \begin{bmatrix} Z_{\phi,k} \\ Z_{v,k} \\ Z_{r,k} \end{bmatrix} = \begin{bmatrix} \rho_{a} - \rho_{\phi} \\ \vdots \\ \rho_{a} - \rho_{\phi} \\ \rho_{\hat{\delta v}} - \rho_{\hat{\delta v}} \\ \vdots \\ \rho_{\hat{\delta v}} - \rho_{\hat{\delta v}} \end{bmatrix},$$  \hspace{1cm} (11)

$$r_k = Z_k - H_k \hat{X}_k.$$  \hspace{1cm} (12)

$$V_k = H_k P_k H_k^T + R_k.$$  \hspace{1cm} (13)
In (11), $\rho_{Gw}'$, $\hat{\rho}_{Gw}$, $\rho_{G}$, and $\rho_{I}$ are the GNSS observation pseudorange and pseudorange rate, INS prediction pseudorange, and prediction pseudorange rate, respectively, $n$ represents the number of visible satellites. The normalized innovation is defined as follows:

$$\omega_i = \frac{r_i'}{\sqrt{V_i''}}.$$  \hspace{1cm} (14)

In (14), $r_i'$ is the $i$-th value of the innovation vector at time $k$ ($i = 1, \ldots, n$, $n$ is the number of visible satellites), and $V_i''$ is the variance of $r_i'$. Moreover, $\omega_i$ represents the $i$-th innovation value after normalization, which can reflect the $i$-th GNSS measurement error. The innovation vector reflects the pseudorange added value disturbed by spoofing, which is called spoofing innovation $r_{k,s}$.

However, because the filter has the effect of spoofing interference and closed-loop correction in the prediction and update stages, that is, the authentic innovation is not equal to the spoofing innovation. When spoofing interference is applied at time $k$, in the state filtering loop, the change of the GNSS measurement value $Z_i$ produces spoofing interference effects, affecting the innovation vector $r_i$ and the state filtering loop at time $k+1$. The analysis and explanation of the spoofing interference effect are shown in Fig. 2, and the specific derivation is described below.

Assuming that deception interference occurs at time $k$, the observation is expressed as follows [17]:

$$Z_{k,s} = Z_k + \Delta Z_k.$$  \hspace{1cm} (15)

In (15), $Z_{k,s}$, $Z_k$, and $\Delta Z$ are the spoofing observation, the expected observation, and the amplitude of spoofing, respectively. When spoofing occurs, the spoofing innovation $r_{k,s}$ is as follows:

$$r_{k,s} = Z_{k,s} - H_k \hat{X}_k = Z_k + \Delta Z_k - H_k \hat{X}_k.$$  \hspace{1cm} (16)

According to the Kalman filter theory, the state estimation value is as follows:

$$\hat{X}_{k,s} = \hat{X}_k + K_k r_{k,s}$$  
$$\hat{X}_{k,s} = \hat{X}_k + K_k (r_k + \Delta Z_k),$$  \hspace{1cm} (17)

where $\hat{X}_k$ and $\hat{X}_{k+1,s}$ represent the posterior state estimation at time $k$ and the posterior state estimation with spoofing interference, respectively. For time update at time $k+1$:

$$\hat{X}_{k+1,s} = \Phi \hat{X}_{k,s},$$  \hspace{1cm} (18)

$$\hat{X}_{k+1,s} = \Phi (\hat{X}_k + K_k \Delta Z_k),$$  \hspace{1cm} (19)

$$r_{k+1,s} = Z_{k+1} - H_{k+1} \hat{X}_{k+1,s},$$  \hspace{1cm} (20)

where $\Phi_k$ and $K_k$ are the transfer matrix and the gain matrix, respectively, the spoofing innovation $r_{k+1,s}$ is derived as follows:

$$r_{k+1,s} = Z_{k+1} - H_{k+1} \hat{X}_{k+1,s} = Z_{k+1} + \Delta Z_{k+1} - H_{k+1} \hat{X}_{k+1,s} = Z_{k+1} + \Delta Z_{k+1} - H_{k+1} \hat{X}_{k+1,s} - K \Delta Z_k.$$  \hspace{1cm} (21)

which can be expressed as

$$I_{k,s} = r_{k,s} - \Delta r_{f(k+1)},$$  \hspace{1cm} (22)

in addition,

$$r_{f(k+1)} = r_{k+1,s} + \Delta Z_{k+1},$$  \hspace{1cm} (23)

where $r_{f(k+1)}$ and $\Delta r_{f(k+1)}$ represent the real value of the innovation when spoofing interference occurs and the component of the innovation caused by spoofing interference, respectively. It can be concluded that the increment of

![FIGURE 2. Analysis of spoofing effects.](image-url)
innovation decreases $H_{k+1} \Phi K_k \Delta Z_k$, which will accumulate over time through the recursive calculation process, resulting in a greater decrease in the increment of innovation, thus reducing the detection performance of spoofing interference.

III. TIGHTLY COUPLED GNSS/INS INTEGRATION SPOOFING DETECTION ALGORITHM BASED ON INNOVATION RATE AND ROBUST ESTIMATION

A. INNOVATION RATE SPOOFING DETECTION ALGORITHM

The innovation rate spoofing detection algorithm is developed to judge whether the GNSS measured value is affected by spoofing interference by normalizing the change rate of innovation $\omega_i$. Considering the effect of the measurement noise, the Kalman filter is usually used to update the change rate of the normalized innovation $\omega_i$ in real time. The detection quantity $v_i$ of innovation rate was derived in [14].

Assuming that spoofing does not exist, the null hypothesis is $H_0: v_i \sim N(0,1)$ and the alternative hypothesis is $H_1: v_i \sim N(\delta,1)$, where $v_i$ follows a normal distribution, and $\delta$ is a noncentral parameter. According to the integrity requirements of the navigation system [25], if the false alarm probability is set to $P_{fa}$, the corresponding false alarm probability of the $i$-th measurement value $a_0$ is as follows [26]:

$$a_0 = 1 - \Phi[1 - P_{fa}].$$

(24)

Thus, the detection threshold $v_D$ corresponding to the innovation rate $v_i$ is as follows [16]:

$$v_D = \sqrt{P_v Q^{-1}(a_0)}.$$ 

(25)

In (24), $Q^{-1}$ is the inverse of the Gaussian distribution, and $P_v$ is the variance of the covariance matrix of $v_i$. Subsequently, the spoofing interference detection criteria is

$$\begin{align*}
    v_i \geq v_D, & \quad \text{With spoofing} \\
    v_i < v_D, & \quad \text{Without spoofing}.
\end{align*}$$

(26)

Because the accumulation of errors will lead to an increase or decrease in innovation, the innovation rate spoofing detection algorithm can determine whether spoofing interference exists by detecting an increase or decrease in innovation without waiting for the accumulation of errors to a certain extent before being detected. Therefore, the detection time of the innovation rate spoofing detection algorithm is shorter than that of the innovation spoofing detection algorithm. However, the filter for calculating the innovation rate cannot be detected before convergence.

B. INNOVATION RATE ROBUST ESTIMATION SPOOFING DETECTION ALGORITHM

As discussed in Section II-A, spoofing interference affects the performance of the spoofing detection algorithm. Based on the innovation rate spoofing detection algorithm, the effect of the spoofing interference can be well reduced by introducing robust estimation, selecting the IGG-3 equivalent weight function [27], and using the innovation rate $v_i$ to calculate the equivalent weight as follows:

$$w_i = \begin{cases}
    1, & v_i \leq k_0 \\
    k_0 \left( k_i - |v_i| \right)^2, & k_0 < v_i \leq k_i \\
    0, & v_i > k_i
\end{cases}$$

(27)

where $v_i$ is the innovation rate of the $i$-th GNSS measurement, and $w_i$ is the corresponding equal weight. In general, $k_i = v_D$ and $k_0 = 0.5 k_i$. When $v_i \leq k_0$, it means that there is no spoofing interference in the $i$-th measurement, and the weight of the dimensional measurement is equal to 1. When $v_i > k_i$, it means that there is spoofing interference in the $i$-th measurement, and the weight of the dimension measurement is equal to zero; therefore, it does not enter the Kalman filter update. When $k_0 < v_i \leq k_i$ indicates that the $i$-th measurement may be affected by spoofing interference, and the weight of the dimension measurement is less than 1. Weight reduction processing is performed to reduce the influence on innovation, thereby improving the spoofing detection performance.

According to the adaptive adjustment of the equivalent weight function, the equivalent weight matrix, $W$, is defined as follows:

$$W = \text{diag}(w_1 \ldots w_i \ldots w_n).$$

(28)

Adjusting the gain matrix $K_k$ yields the following [18]:

$$K_k = K_k \cdot W.$$ 

(29)

According to the previous analysis, a $\Delta r_{(k+1)}$ between the spoofed innovation and the authentic innovation exists, directly causing the decrease in innovation. Therefore, reducing $\Delta r_{(k+1)}$ is an effective way to improve the ability of spoofing detection. Thus, replacing $K_k$ in (22) by $K_k$, the following equation is obtained:

$$\Delta r_{(k+1)} = H_{k+1} \Phi K_k \Delta Z_k = H_{k+1} \Phi (K_k \cdot W) \Delta Z_k.$$ 

(30)

Therefore, when spoofing is present, the element $\Delta Z_k$ in $\Delta Z_k$ increases or decreases, causing the normalized innovation $\omega_i$ at time $k$ to increase or decrease, causing the innovation rate $v_i$ to increase, while (27) adaptively adjusts the equivalent weight matrix; thus, adjusting the gain matrix to reduce the weight of spoofing, effectively weakening the abnormal effect, and improving the detection ability of integrated navigation system. However, the algorithm requires extensive time to detect small steps and slow growth spoofing and even fails to detect it.

IV. SPOOFING DETECTION ALGORITHM BASED ON INNOVATION RATE OPTIMIZATION AND ROBUST ESTIMATION

The limitations described above are solved by improving the innovation rate robust estimation spoofing detection algorithm.
Thus, a GNSS/INS tightly coupled system spoofing detection algorithm with innovation rate optimization and robust estimation was proposed. The proposed algorithm improved the normalized innovation of a small step or slow growth spoofing interference. Thus, it optimized the statistical test amount of the innovation rate, solved the challenge of extensive or even invalid detection time for small steps or slow growth ramp spoofing, and reduced the detection time, improving detection performance. Next, the improved methods and ideas of the two algorithms will be given.

A. IMPROVED SMALL STEP SPOOFING DETECTION ALGORITHM

Assuming that a step spoofing a the small pseudorange added value \( B \) is applied, the normalized innovation \( \omega_i^{ST} \) of a small step spoofing is improved when the \( i \)-th measurement is performed as follows:

\[
\omega_i^{ST} = \frac{B + r_i'}{\sqrt{V_i'}} = b + \omega_i.
\]  

In (31), \( r_i' \) and \( V_i' \) are obtained from (12) and (13), respectively, \( \omega_i \) is the normalized innovation without spoofing, equivalent to (14), the variable \( b \) is equal to the actual small step \( B \) divided by the normalized variance \( \sqrt{V_i'} \). The innovation rate \( \nu_i^{ST} \) of small step is obtained by updating \( \omega_i^{ST} \) in real time using the Kalman filter.

Defining the state vector as \( x \) as follows:

\[
x = (\hat{\omega}_i^{ST} ; \hat{\nu}_i^{ST} ; \hat{a}_i^{ST} ; p_i^{ST})^T.
\]

In (31), \( \hat{\omega}_i^{ST} \) is the estimated value of \( \omega_i^{ST} \), \( \hat{\nu}_i^{ST} \) is the innovation rate of \( \omega_i^{ST} \), \( \hat{a}_i^{ST} \) is the innovation acceleration of \( \omega_i^{ST} \), and \( p_i^{ST} \) is the constant deviation of \( \omega_i^{ST} \).

The system model is defined as in [14] as follows:

\[
\begin{bmatrix}
\hat{\omega}_i^{ST} \\
\hat{\nu}_i^{ST} \\
\hat{a}_i^{ST} \\
p_i^{ST}
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & -\alpha & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\hat{\omega}_i^{ST} \\
\hat{\nu}_i^{ST} \\
\hat{a}_i^{ST} \\
p_i^{ST}
\end{bmatrix}
+ 
\begin{bmatrix}
\nu_i^{ST}
\end{bmatrix}.
\]

In (33), the improved innovation rate is defined as a time-dependent stochastic process, and \( \alpha \) is the correlation coefficient, generally 0.5–0.9, \( \nu \) is the noise, generally \( 10^{-7} \)–\( 10^{-9} \).[28]

The observation model is defined as follows:

\[
\omega_i^{ST} = [1 \ 0 \ 0 \ -1] \begin{bmatrix}
\hat{\omega}_i^{ST} \\
\hat{\nu}_i^{ST} \\
\hat{a}_i^{ST} \\
p_i^{ST}
\end{bmatrix} + \nu_i^{ST}.
\]

In (34), \( \omega_i^{ST} \) is the measured value input, and \( \nu_i^{ST} \) is the observed noise. The improved innovation rate test statistic is calculated by using a double-layer Kalman filter. The flow chart of its implementation is shown in Fig. 3. The specific steps and detailed derivation are as follows:
FIGURE 3. Flow chart of improved innovation rate test statistics.

1) SPECIFIC STEPS

Specific steps involve a double-layer Kalman filter, the main navigation Kalman filter was used to calculate the innovation value, and the innovation rate Kalman filter was used to calculate the innovation rate.

1) Calculate the innovation vector. In the main navigation Kalman filter, the innovation vector \( r_i \) and the variance \( V_i^u \) are calculated from (12) and (13).

2) Calculate the improved normalized innovation. An improved normalized innovation \( \omega_{ST}^{i} \) was obtained by substituting \( r_i \), \( V_i^u \), and the pseudorange added value \( B \) of small step into (31).

3) Initialize the innovation rate Kalman filter. State variables were initialized, covariance values, noise matrices, and dynamic matrices were estimated.

4) Calculate the innovation rate. \( \omega_{ST}^{i} \) was fed back to the innovation rate Kalman filter for real-time updating, and the innovation rate \( v_{ST}^{i} \) was obtained.

5) Compare thresholds. \( v_{ST}^{i} \) was compared with the detection threshold \( v_D \).

2) DETAILED DERIVATION

Time update:

Step 1: prior state estimation

\[
\hat{x}_{k-1}^i = \phi_{k-1}\hat{x}_{k-1}^i ,
\]  

(35)

Step 2: prior covariance estimation

\[
p_{k-1}^i = \phi_{k-1}p_{k-1}^i \phi_{k-1}^T + q_{k-1} ,
\]  

(36)

where the symbol “\( \wedge \)” denotes the estimated value, the superscript “\( \wedge \)" denotes the prior estimate and the symbol “\( + \)” denotes the posterior estimate. \( x_k^i \), \( x_{k-1}^i \), and \( \phi_{k-1} \) represent the prior estimate at time \( k \), the posterior estimate and the state transition matrix at time \( k-1 \), respectively. \( p_k^i \), \( p_{k-1}^i \), and \( q_{k-1} \) represent the prior error covariance matrix at time \( k \), the posterior error covariance matrix and the system noise covariance matrix at time \( k-1 \), respectively.

Measurement update:

In the update stage of the traditional Kalman filter, the observation vectors \( Z_i \) and \( H_i\hat{X}_i \) were subtracted to obtain innovation vectors \( r_i \). In contrast, in the proposed algorithm, the improved normalized innovation \( \omega_{ST}^{i} \) and \( H_i\hat{X}_i \) were subtracted to obtain a small step spoofing innovation vector \( r_{ST}^{i} \) and subsequently updated in real time to obtain \( \hat{x}_{k}^i \).

Step 1: improved spoofing innovation vector

\[
r_{ST}^{i} = \omega_{ST}^{i} - H_i\hat{X}_i ,
\]  

(37)

corresponds to the observation value \( Z_i \), and \( \omega_{ST}^{i} \) was used as an input to the innovation rate observation value.

Step 2: gain matrix

\[
K_i = p_i^i H_i^T (H_i p_i^i H_i^T + R_i)^{-1} .
\]  

(38)

Step 3: posteriori state estimation

\[
\hat{x}_k^i = \hat{x}_{k-1}^i + K_i r_{ST}^{i} ,
\]  

(39)

Step 4: posteriori covariance estimation

\[
p_{k}^i = (I - K_i H_i) p_{k-1}^i ,
\]  

(40)

where \( H_i \) and \( R_i \) represent the observation matrix and the observation noise covariance matrix. In the measurement update, the improved test statistic, \( \omega_{ST}^{i} \), was inputted into the Kalman filter as the observation value for cyclic updating. Then, the optimized small step innovation rate \( v_{ST}^{i} \) was calculated through innovation rate output matrices \( C \) and \( \hat{x}_k^i \) as follows:

\[
v_{ST}^{i} = C \cdot \hat{x}_k^i .
\]  

(41)

In (40), \( C = [0 \ 1 \ 0 \ 0] \).

The criteria for judging whether a small step spoofing was detected are as follows:

\[
\begin{aligned}
&v_{ST}^{i} \geq v_D , &\text{With spoofing} \\
&v_{ST}^{i} < v_D , &\text{Without spoofing} .
\end{aligned}
\]  

(42)

B. IMPROVED SLOW-GROWING RAMP SPOOFING DETECTION ALGORITHM

Suppose that ramp spoofing with a slow-growing pseudorange added value \( A \) is applied, and the change rate \( a \) of the normalized innovation amplitude remains unchanged over time, which occurs in the \( i \)-th measurement. In this case, the normalized innovation \( \omega_{SG}^{i} \) of the improved slow-growing ramp spoofing is as follows:

\[
\omega_{SG}^{i} = \frac{A + \omega_i}{\sqrt{V_i^u}} = \frac{a(t-t_{Lock})}{\sqrt{V_i^u}} + \omega_i = a' + \omega_i .
\]  

(43)

In (43), the variable \( a' \) is equal to the actual slow growth \( a(t-t_{Lock}) \) divided by the normalized variance \( \sqrt{V_i^u} \). The innovation rate \( v_{SG}^{i} \) of a small step is obtained by updating \( \omega_{SG}^{i} \) in real time using the Kalman filter, similar to the small step spoofing detection algorithm. Thus, this is not described in detail here.

As described above, the improved spoofing algorithm for a small step or slow growth is shown in Fig. 4, where "\( \# \)" represents the innovation rate test statistic of the small step \( (v_{ST}^{i}) \) or slow growth \( (v_{SG}^{i}) \). The specific steps are as follows:
Spoofing detection based on innovation rate optimization and robust estimation

Improved normalized innovation of main navigation Kalman filter

Optimized innovation rate of innovation rate Kalman filter

Calculate the gain matrix

Filter measurement update at time $k$

End of the filtering process?

FIGURE 4. Flow chart of the proposed algorithm.

1) The main navigation Kalman filter for the innovation detection method: perform time and measurement updates to prepare for the next calculation of the normalized innovation.

2) Initialize the innovation rate Kalman filter: including state variables, covariance values, measurement noise, and others.

3) Improve normalized innovation: from Section IV-A, “improved small step spoofing detection algorithm”, as an example, the improved normalized innovation $w_i^{ST}$ is calculated from (31) at time $k$.

4) Optimize the innovation rate: the improved normalized innovation is inputted into the Kalman filter using (37) to obtain a posterior estimated state $\hat{x}_i^+$, and the innovation rate $v_i^{ST}$ is calculated by (41).

5) Robust estimation: the equivalent weight matrix $W$ is calculated from (27).

6) Gain matrix: the gain matrix $K_R$ is calculated from (29) and the measurement is updated.

7) Filter cycle update: the spoofing detection process is completed at time $k$ and step 1) is returned at time $k + 1$.

V. RESULTS AND DISCUSSION

Various detection algorithms were implemented to verify the effectiveness of the proposed algorithm. In particular, the innovation rate spoofing (M1), innovation rate robust estimation spoofing (M2), improved small step spoofing detection algorithm (M3), and improved slow growth ramp spoofing (M4) detection algorithms were implemented.

According to simulation experiments of the GNSS/INS tightly coupled system spoofing detection, four scenarios were designed. 1) The detection ability of the M1 and M2 were compared for the case when three channels were spoofed by step or ramp spoofing with the same pseudorange added value.

2) The detection ability of the M2 was verified for the case when one channel was spoofed by step or ramp spoofing with different pseudorange added value.

3) The detection abilities of the M2 and M3 were compared for the case when one channel was spoofed by a small step.

4) The detection abilities of the M2 and M3 were compared for the case when one channel was spoofed by a small step.

5) The detection abilities of the M2 and M3 were compared for the case when one channel was spoofed by a small step.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
of the M2 and M4 were compared for the case when one channel was spoofed by a slow growth ramp.

A. SIMULATION CONDITIONS

Referring to the simulation software guide [29], the GNSS constellation model was a single constellation, dual-frequency, and circular orbit. Moreover, the satellites were all distributed on six orbital planes, without GNSS signal occlusion, attenuation, interference, or reflection. In particular, the following simulation conditions were considered: the false alarm rate was $P_{fa} = 4 \times 10^{-6}$ [30], and the thresholds of the M1, M2, M3, and M4 were all 0.0029517 m/s. The parameters of the GNSS and IMU modules are listed in Table 1. The airborne motion was simulated in Matlab, considering a speed of 200 m/s, two 45 turns, and climbs that last 746 s. The flight path is shown in Fig. 5.

TABLE 1. Simulation parameters.

| Sensors | Parameter                  | Value       |
|---------|----------------------------|-------------|
| GNSS    | Number of visible satellites | 8           |
|         | Output rate                | 1 Hz        |
|         | Pseudorange noise (1 σ)    | 2.5 m       |
| IMU     | Accelerometer noise root PSD* | 20 mg/√Hz |
|         | Accelerometer biases        | (30, -45, 26) mg |
|         | Gyro Noise Root PSD         | 0.002°/h    |
|         | Gyro biases                 | (-0.0009, 0.0013, -0.0008)°/h |
|         | Output rate                 | 100 Hz      |

* PSD: Power Spectral Density.

B. SIMULATION RESULTS AND ANALYSIS

1) SCENARIO 1

Scenario 1 was set as shown in Table 2. Two sets of experiments were set up. The detection capabilities of M1 and M2 were compared when channels 1, 2, and 3 were spoofed by the same pseudorange added value.

![Flight trajectory](image)

FIGURE 5. Flight trajectory.

![Comparison of the simulation results for step spoofing of M1 and M2](image)

FIGURE 6. Comparison of the simulation results for step spoofing of M1 and M2.

The first set of step spoofing simulation results is shown in Fig. 6. In particular, a step spoofing with a pseudorange deviation of 30 m was applied to channels 1, 2, and 3 in 350–550 s. In the legend, “C” and “T” represent channel and threshold, that is, “C1”, “C2”, and “C3” represent channels 1, 2, and 3, respectively, and the rest are analogous, which will not be repeated in the following legends. Fig. 6(a) shows the M1 simulation results. Note that the detection times of channels 1, 2, and 3 were 10, 9, and 10 s, respectively. However, spoofing interference affected the other five channels to varying degrees, resulting in the corresponding innovation rate deviating from the normal value and false alarm. In addition, Fig. 6(b) shows a diagram of the M2 simulation results. Note that the detection times of channels 1, 2, and 3 were 9, 8, and 9 s, respectively, which were 1, 2, and 1 s shorter than that of the corresponding channels in Fig. 6(a). Moreover, the innovation rate of other channels was normal.

TABLE 2. Spoofing Scenario 1.

| Spoofing type | Pseudorange added value | Channel | Time     |
|---------------|-------------------------|---------|----------|
| Step          | 30 m                    | 1, 2, 3 | 350–550 s|
| Ramp          | 0.3 m/s                 |         |          |
In addition, as seen in the figure, when the step spoofing with a relatively large pseudorange deviation was applied, the detection efficiency of M2 did not improve compared with that of M1. However, M2 can restrain the innovation rate from deviating from the normal value, increasing the fault tolerance and robustness of the system.

The second set of ramp spoofing simulation results is shown in Fig. 7. The ramp spoofing with a slope of 0.3 m/s was applied to channels 1, 2, and 3 in 350–550 s. In particular, Fig. 7(a) shows the M1 simulation results. Note that the detection time of channels 1, 2, and 3 were 59, 50, and 64 s, respectively. Moreover, the spoofing interference affected the other channels to varying degrees, resulting in a false alarm. In addition, Fig. 7(b) depicts the M2 simulation results, showing that the detection times of channels 1, 2, and 3 were 47, 43, and 49 s, respectively, shortening by 12, 7, and 15 s, respectively, as compared with that of the corresponding channels in Fig. 7(a). The innovation rate of other channels was normal.

Monte Carlo simulation was performed according to Scenario 1 for 100 cycles to illustrate the effect of the M2 robust estimation. The missed detection and false alarm rates of the two algorithms are listed in Table 3.

**TABLE 3.** Scenario 1 Monte Carlo simulation results.

| Method  | Missing detection rate (%) | False alarm rate (%) |
|---------|----------------------------|----------------------|
| M1      | Chann- | Chann- | Chann- | Chann- | Chann- | Chann- |
| el 1*   | 0     | 0     | 0     | 100    | 100    | 100    |
| M2      | 0     | 0     | 0     | 3      | 1      | 1      |

From the result shown in Figs. 6 and 7 and Table 3, the following conclusions can be drawn: 1) When M2 was spoofed by the step, the detection time of M2 was reduced by 10%, 22.2%, and 10%, respectively, as compared with that of M1. Moreover, the average time was reduced by 35.5%. Compared with the detection time of M1, that of M2 was reduced by 20.3%, 14%, and 64.2%, respectively. The average time was shortened by 32.8%. Thus, the average detection efficiency of M2 was approximately one-third higher than that of M1. 2) For the missed detection rate, channels 1, 2, 3, and 6 of M1 and M2 were all zero. The false alarm rate of channels 4, 5, and 6 of M1 were 100%, while that of channels 4, 5, and 6 of M2 is 3%, 1%, and 1%, respectively, which were reduced by 97%, 99%, and 99%, respectively. The average reduction was 98.3%. Finally, the robust estimation of M2 suppressed the innovation rate of the normal channels from the normal value and reduced the false alarm rate. The detection time was reduced, and the detection performance improved.

2) SCENARIO 2

Scenario 2 was set as shown in Table 4. Sun et al. [31] set the scenario of channel 3 with different pseudorange added values for different spoofing types and verified that M2 had limitations on small step spoofing of 5 m or slow growth ramp spoofing of 0.05 m/s. The detection time was extensive or even invalid.

**TABLE 4.** Spoofing Scenario 2.

| Spoofing type | Pseudorange added value | Channel | Time |
|---------------|-------------------------|---------|------|
| Step          | 40, 30, 20, 10, and 5 m | 3       | 350-550 s |
| Ramp          | 0.4, 0.3, 0.2, 0.1, and 0.05 m/s | 3 | 350-550 s |

The simulation results are shown in Fig. 8. In particular, Fig. 8(a) shows the simulation result of M2 step spoofing. Note that in the time from 350–550 s, step spoofing with pseudorange deviations of 40, 30, 20, 10, and 5 m is applied to channel 3, and the detection times were 4, 6, 10, and 25 s, respectively, in which small step spoofing detection of 5 m is invalid. In addition, Fig. 8(b) depicts the simulation result of M2 ramp spoofing. A ramp spoofing with pseudorange deviations of 0.4, 0.3, 0.2, 0.1, and 0.05 m/s was applied to channel 3; The detection time were 36, 44, 57, 92, and 157 s, respectively. Therefore, this scenario verified that with a decrease in pseudorange deviation of step application (from 40–5 m) and slope application (from 0.4–0.05 m/s), the detection time increased, (even the detection was ineffective). In particular,
the maximum detection time for a slow growth ramp spoofing of 0.05 m/s was 157 s and the detection of small a step spoofing of 5 m was invalid.

FIGURE 8. Comparison of the simulation results for different pseudorange added values of different spoofing types in M2.

3) SCENARIO 3
Based on the limitations of M2 in detecting small step spoofing in Scenario 2, a small step spoofing in Scenario 3 was set as shown in Table 5. For this scenario, the detection capabilities of M2 and M3 were compared.

TABLE 5. Spoofing Scenario 3.

| Spoofing type | Pseudorange added value | Channel | Time       |
|---------------|------------------------|---------|------------|
| Small step    | 10 and 5 m             | 3       | 350–550 s  |

The simulation results are shown in Fig. 9. A step spoofing with pseudorange deviations of 10 and 5 m was applied to channel 3 from 350–550 s. In particular, Fig. 9(a) shows the simulation results for the small step spoofing for M2 and M3. Note that for the pseudorange deviation of 10 m, the detection times of M2 and M3 were 25 and 9 s, respectively. For 5 m pseudorange deviation, M2 detection was ineffective and M3 detection time was 22 s. Therefore, M3 was more sensitive than M2 in spoofing detection of small steps. In addition, Fig. 9(b) shows the effect of the small step spoofing on position error. When 10 and 5 m spoofing were applied, the maximum errors in the north direction are 1.04091 and 1.04169 m, respectively, and the maximum errors in the east direction were 1.30492 and 1.39802 m, respectively. The altitude error had slight changes, showing that the small step spoofing was hidden and the error accuracy requirements were satisfied. This was generally difficult to detect, even though it was paramount in high-precision positioning applications, such as missile precision guidance, intelligent driving, and unmanned aerial vehicle.

FIGURE 9. Comparison of simulation results for small step spoofing between M2 and M3.

Furthermore, Monte Carlo simulations were performed for Scenario 3 for 100 cycles to show that M3 was superior to M2. The missed detection and false alarm rates of the two algorithms are listed in Table 6.

TABLE 6. Scenario 3 Monte Carlo simulation results.

| Method | Missing detection rate (%) | False alarm rate (%) |
|--------|----------------------------|----------------------|
|        | Channel 3* | Channel 4 | Channel 5 | Channel 6 |
| M2     | 74         | 0         | 0         | 1         |
| M3     | 0          | 0         | 0         | 1         |
The results in Fig. 9 and Table 6 show that: 1) when channel 3 was spoofed by 10 m, the detection time of M3 was shortened by 64% as compared with that of M2. M3 detection was effective, and M2 detection was invalid when it was interfered with by 5 m small step spoofing. 2) For the missed detection rate, M3 was 74% lower than M2. For the false alarm rate, the 3, 4, and 5 channels of M2 and M3 were approximately zero. Therefore, M3 was more sensitive to small step spoofing detection and inherited the robust estimation effect of M2.

4) SCENARIO 4

Based on the limitations of M2 in Scenario 2 to slow-growing ramp spoofing detection, Scenario 4 for slow-growing ramp spoofing was set as shown in Table 7. The detection capabilities of M2 and M4 were compared.

TABLE 7. Spoofing Scenario 4.

| Spoofing type          | Pseudorange added value | Channel | Time   |
|------------------------|-------------------------|---------|--------|
| slow-growing ramp      | 0.1 and 0.05 m/s        | 3       | 350-550 s |

The results in Fig. 10 and Table 8 show that: 1) when channel 3 was spoofed by the slow growth of 0.1 and 0.05 m/s, the detection time of M4 were reduced by 38% and 41.5%, respectively, as compared with that of M2, with an average reduction of 39.8%. 2) For the missed detection rate, M3 was 42% lower than that of M2, and for the false alarm rate, channels 3, 4, and 5 of M2 and M4 were approximately zero. Therefore, M4 was more sensitive to slow-growing ramp spoofing detection.

VI. CONCLUSION

A GNSS/INS spoofing detection algorithm based on robust estimation of innovation rate is effective for large-scale spoofing. However, this algorithm requires substantial time to detect a small step and slow-growing ramp spoofing interference. In this study, a tightly coupled GNSS/INS integration spoofing detection algorithm based on innovation rate optimization and robust estimation was proposed. The proposed algorithm established a two-layer Kalman filter. First, the normalized innovation was improved using an innovation detection method in the main navigation Kalman filter. Subsequently, the improved normalized innovation was inputted into the innovation rate Kalman filter for measurement update, thus optimizing the innovation rate test statistics. Simultaneously, the robust estimation was introduced to adaptively adjust the gain matrix, which reduced the effect of spoofing interference on the innovation rate and further improved the detection and processing ability of small step or slow growth ramp spoofing interference. Simulation results showed that the detection time of the proposed algorithm was reduced by 64% and 39.8%, respectively, with an average reduction of 51.9% when detecting step mutation.
or slow growth spoofing interference. Moreover, the missed detection rate decreased by 74% and 42%, respectively, with an average decrease of 58%. The false alarm rate was maintained at approximately zero. Compared with existing algorithms, the proposed algorithm exhibited fast detection and low missed detection and false alarm rates when detecting the spoofing of a small step and slow growth ramp. Our algorithm was suitable for the spoofing detection of tightly coupled GNSS/INS integration user high-precision unmanned aerial vehicle applications.

To improve the applicability of the new algorithm, further work can be performed in the following aspects: 1) In the integrity detection level and protection level of navigation systems, the change of detection probability should be studied by changing the false alarm rate. 2) Adding real data to verify the improved algorithm, and 3) Research on the multi-channel spoofing detection algorithm.

REFERENCES
[1] T.-S. Lou et al., “Robust partially strong tracking extended Kalman filtering for INS/GNSS integrated navigation,” IEEE Access, vol. 7, pp. 151230-151238, 2019, doi: 10.1109/ACCESS.2019.2948229.
[2] B.S.F et al., “Research status and prospect of GNSS anti-spoofing technology (in Chinese),” Sci. Sin. Inform., vol. 47, no. 003, pp. 275-287, 2017, doi: 10.1360/NNJ16-00073.
[3] R. T. Ioannides et al., “Known vulnerabilities of global navigation satellite systems, status, and potential mitigation techniques,” Proc. IEEE, vol. 104, no. 6 pp. 1174-1194, 2016, doi: 10.1109/JPROC.2016.2535898.
[4] T. E. Humphreys et al., “Assessing the spoofing threat: Development of a portable GPS civilian spoofer” in Proc. 21st International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2008), 2008, pp. 2314-2325.
[5] L. Zhang et al., “Fault-tolerant control for intelligent electrified vehicles against front wheel steering angle sensor faults during trajectory tracking,” IEEE Access, vol. 9, pp. 65174-65186, 2021, doi: 10.1109/ACCESS.2021.3075325.
[6] A. J. Kerns et al., “Unmanned aircraft control and capture via GPS spoofing,” J. Field Robot., vol. 31, no. 4, pp. 617-636, 2014, doi: 10.1002/rob.21513.
[7] Z. Zhang et al., “An enabling trajectory planning scheme for lane change collision avoidance on highways,” IEEE Trans. Intell. Veh., vol. 1–1, 2021, doi: 10.1109/TIV.2021.3117840.
[8] Z. Chao et al., “A spoofing detection algorithm for INS/GNSS integrated navigation system based on innovation rate and robust estimation (in Chinese),” J. Chin. Inertial Technol., vol. 29, no. 3, Jun., 2021.
[9] T. S. Abuhishim et al., “Building a robust integrity monitoring algorithm for a low cost GPS-aided INS system,” Int. J. Control Auton. Syst., vol. 8, no. 5, pp. 1108-1122, 2010, doi: 10.1109/IJCAST.2010.05020-1.
[10] M. Joeger and B. Pervan, “Kalman filter-based integrity monitoring against sensor faults,” J. Guid. Control Dyn., vol. 36, no. 2, pp. 349-361, 2013, doi: 10.2514/1.J09480.
[11] J. Diesel and S. Luu, “GPS/IRS AIME: Calculation of thresholds and protection radius using chi-square methods” in Proc. 8th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS) 1995, 1995, pp. 1599-1604.
[12] S. Hewison and J. Wang, “Extended receiver autonomous integrity monitoring (e raim) for gns/s integration,” J. Surv. Eng., vol. 136, no. 1, pp. 13-22, 2010, doi: 10.1061/(ASCE)0733-9453(2010)136:1(13).
[13] M. Orejas et al., “Multi-constellation GNSS/INS to support LPV200 approaches and autolanding” in Proc. 25th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2012), 2012, pp. 790-803.
[14] U. I. Bhatti et al., “Performance of rate detector algorithms for an integrated GPS/INS system in the presence of slowly growing error,” GPS Solut., vol. 16, no. 3, pp. 293-301, 2012, doi: 10.1007/s10291-011-0231-y.
[15] Y. Liu et al., “Analysis of Kalman filter innovation-based GNSS spoofing detection method for INS/GNSS integrated navigation system,” IEEE Sens. J., vol. 19, no. 13, pp. 5167-5178, 2019, doi: 10.1109/JSEN.2019.2902178.
[16] U. I. Bhatti et al., “Integrity of an integrated GPS/INS system in the presence of slowly growing errors. Part I: A critical review,” GPS Solut., vol. 11, no. 3, pp. 173-181, 2007, doi: 10.1007/s10291-006-0048-2.
[17] L. Zhong et al., “Approach for detecting soft faults in GPS/INS integrated navigation based on LS-SVM and AIME,” J. Navig., vol. 70, no. 3, pp. 561-579, 2017, doi: 10.1017/S037346331600076X.
[18] S. Wang et al., “GNSS/INS tightly coupling system integrity monitoring by robust estimation,” J. Aeronaut. Astronaut. Aviat. I., vol. 50, pp. 61-80, 2018, doi: 10.6152/JAAA.201803.051106.
[19] X. U. Rui et al., “Spoofing interference identification technique of MEDLL aided GNSS/INS system (in Chinese),” J. Chin. Inertial Technol., 2018.
[20] C. Zhang et al., “Improved fault detection method based on robust estimation and sliding window test for INS/GNSS integration,” J. Navig., vol. 73, no. 4, pp. 776-796, 2020, doi: 10.1017/S0373463319000778.
[21] R. Sun et al., “A novel online data-driven algorithm for detecting UAV navigation sensor faults,” Sensors (Basel), vol. 17, no. 10, pp. 2243, 2017, doi: 10.3390/s17102243.
[22] Y. Liu et al., “Impact assessment of GNSS spoofing attacks on INS/GNSS integrated navigation system,” Sensors (Basel), vol. 18, no. 5, pp. 1433, 2018, doi: 10.3390/s18051433.
[23] A. Angrisano, “GNSS/INS integration methods.,” Dottorato di Ricerca (PhD) in Science Geodetiche [e Topografiche thesis], Universita degli studi di napoli Parthenope, Naples, vol. 21, 2010.
[24] M. S. Grewal and A. P. Andrews, Kalman Filtering: Theory and Practice with MATLAB, 4th ed., 2014.
[25] J. Blanch et al., “Baseline advanced RAIM user algorithm and possible improvements,” IEEE Trans. Aeros. Electron. Syst., vol. 51, no. 1, pp. 713-732, 2015, doi: 10.1109/TAES.2014.130739.
[26] R. J. Kelly, “The linear model, RNP, and the near-optimum fault detection and exclusion algorithm,” J. Glob. Positioning Syst., vol. 5, pp. 227-259, 1998.
[27] Y. Yang et al., “Robust estimation for a dynamic model of the sea surface,” Surv. Rev., vol. 35, no. 271, pp. 2-10, 1999, doi: 10.1179/sre.1999.35.271.2.
[28] R. Brown and P. Hwang, Introduction to Random Signals and Applied Kalman Filtering(Book), New York: John Wiley & Sons, Inc., 1992, p. 1992. 512.
[29] P. D. Groves, “Principles of GNSS, inertial, and multisensor integrated navigation systems, [book review],” IEEE Aerosop. Electron. Syst. Mag., vol. 30, no. 2, pp. 26-27, 2015, doi: 10.1109/MAES.2014.141101.
[30] J. Blanch et al., “Advanced RAIM user algorithm description: Integrity support message processing, fault detection, exclusion, and protection level calculation” in Ion Institute of Navigation Global Navigation Satellite Systems Conference, 2012.
[31] R. Sun et al., “GNSS/INS integration with integrity monitoring for UAV no-fly zone management,” Remote Sens., vol. 12, no. 3, p. 524, 2020, doi: 10.3390/rs12030524.
YE KE received the B.S. degree in software engineering from East China Jiaotong University, Nanchang, Jiangxi, China in 2015, and is currently studying for a M.S. degree at Zhengzhou University of Information Engineering, Henan, China.

He has published several articles on satellite navigation data processing and anti-spoofing techniques. He is currently mainly researching GNSS/INS integrated navigation anti-spoofing technology.

ZHIWEI LV received the Ph.D. degree in space geodetic survey and navigation from Information Engineering University, Zhengzhou, Henan, China, in 2010. He is currently a Professor with Information Engineering University.

He has published several articles on satellite navigation data processing and space geodetic survey. His current research focuses mainly on satellite navigation data processing, space geodetic survey, and anti-spoofing technology.

CHAO ZHANG received the M.S. degree in software engineering from Information Engineering University, Zhengzhou, Henan, China, in 2021.

He has published several articles on satellite navigation data processing and satellite navigation anti-spoofing technology. His current research focuses mainly on satellite navigation anti-spoofing technology.

XU DENG received the B.S. degree in navigation engineering from Information Engineering University, Zhengzhou, Henan, China, in 2019, where he is currently pursuing the M.S. degree.

His current research focuses mainly on satellite navigation spoofing countermeasures. He has published several papers on satellite navigation data processing and satellite navigation anti-spoofing technology.

WENLONG ZHOU received the B.S. degree in navigation engineering from Information Engineering University, Zhengzhou, Henan, China, in 2020, where he is currently pursuing the M.S. degree.

He has published several articles on satellite navigation data processing and satellite navigation anti-spoof technology. His current research focuses mainly on satellite navigation anti-spoofing countermeasures.

DEBIAO SONG received the B.S. degree in navigation engineering from Information Engineering University, Zhengzhou, Henan, China, in 2020.

His current research mainly focuses on the tight combination of satellite inertial navigation, and he is in the stage of verifying the experimental results.