Solution Separation Based Integrity Monitoring for Integer Ambiguity Resolution Enabled GNSS Positioning

Shizhuang Wang, Xingqun Zhan, Yawei Zhai, and Zhen Gao
School of Aeronautics and Astronautics, Shanghai Jiao Tong University, China

Biographies

Shizhuang Wang is a PhD candidate at Shanghai Jiao Tong University. His research interests focus on navigation integrity and multi-sensor integrated navigation.
Xingqun Zhan is a Professor at Shanghai Jiao Tong University. His research areas focus on navigation integrity, cooperative navigation, and multi-sensor fusion. He is an AIAA associate fellow and an associate editor of Aerospace Science and Technology.
Yawei Zhai is a postdoctoral researcher at Shanghai Jiao Tong University. His research interests focus on navigation integrity and multi-sensor integrated navigation.
Zhen Gao is a PhD candidate at Shanghai Jiao Tong University. His research interests focus on GBAS and navigation integrity.

Abstract

Carrier-phase ambiguity resolution (AR) facilitates rapid and precise GNSS positioning. This work describes an integrity monitoring algorithm to ensure high integrity of AR-enabled GNSS positioning systems, such as PPP-RTK and RTK. Prior studies either (a) only focused on protecting ambiguity-float GNSS positioning against faulted observations (i.e., raw measurements or correction products) or (b) evaluated the integrity risk from incorrect ambiguity resolution while assuming the observations as fault-free. These approaches or their simple combinations cannot address the cases where observation faults threaten ambiguity-fixed GNSS positioning, because the occurrence of observation faults affects the risk from incorrect-fix events. The proposed approach can accommodate observation faults and incorrect-fix events simultaneously, through a modified multiple hypothesis solution separation (MHSS) architecture. Specifically, multiple incorrect-fix events are considered under each of the monitored hypotheses that assume the health status of the observations. Fault detection is performed by comparing each subset solution with the all-in-view solution. And when evaluating the protection levels, the integrity risk from the incorrect-fix events is quantified under each of the monitored hypotheses. Simulation results prove the effectiveness of the proposed algorithm and show that enabling ambiguity resolution greatly benefits navigation performance.

Keywords: integrity monitoring, ambiguity resolution, PPP-RTK, fault detection, protection level

1. INTRODUCTION

Precise and trustworthy localization plays a crucial role in emerging autonomous systems, especially in safety- or payment-critical applications such as highly automated vehicles (HAVs) and urban air mobility (UAM). In these applications, misleading localization information may lead to serious accidents. Therefore, it becomes a solid demand to ensure high navigation integrity aside from improving navigation accuracy.

Compared to civil aircraft, emerging autonomous systems usually require their onboard navigation engines to meet more stringent navigation requirements [1], which cannot be satisfied by global navigation satellite systems (GNSS) single point positioning (SPP). To meet the strict requirement on accuracy, GNSS carrier-phase observations need to be used aside from pseudoranges. As typical GNSS positioning techniques using carrier phases, real-time kinematic (RTK) and precise point positioning (PPP) with fast ambiguity resolution (known as PPP-RTK) can offer accurate and continuous positioning services with a short convergence period. These advantages partly come from the fact that they employ observation-space-representation (OSR) or state-space-representation (SSR) correction products. And they also benefit from carrier-phase integer ambiguity resolution (AR), because correctly fixing the ambiguities will greatly facilitate rapid and precise GNSS positioning [2].

On the other hand, integrity monitoring must be implemented to ensure high integrity for PPP-RTK and RTK. PPP-RTK and RTK are vulnerable to the faults in GNSS raw measurements (i.e., pseudoranges and carriers) and those in correction products. They may also experience large errors when incorrect-fix (IF) events occur in the AR step. An IF event is said to occur when at least one of the fixed ambiguities differs from its truth. **Fig. 1** summarizes the threats that a PPP-RTK or RTK navigation engine may encounter and shows their effects on the position estimates. This figure discusses three cases, among which case (c) is the most general one and is closest to the real situation of the navigation systems for autonomous platforms. Accordingly, this paper aims at developing an integrity monitoring algorithm for ambiguity resolution enabled GNSS positioning, e.g., PPP-RTK and RTK, which can protect the navigation engine against measurement faults, product faults, and IF events, simultaneously.

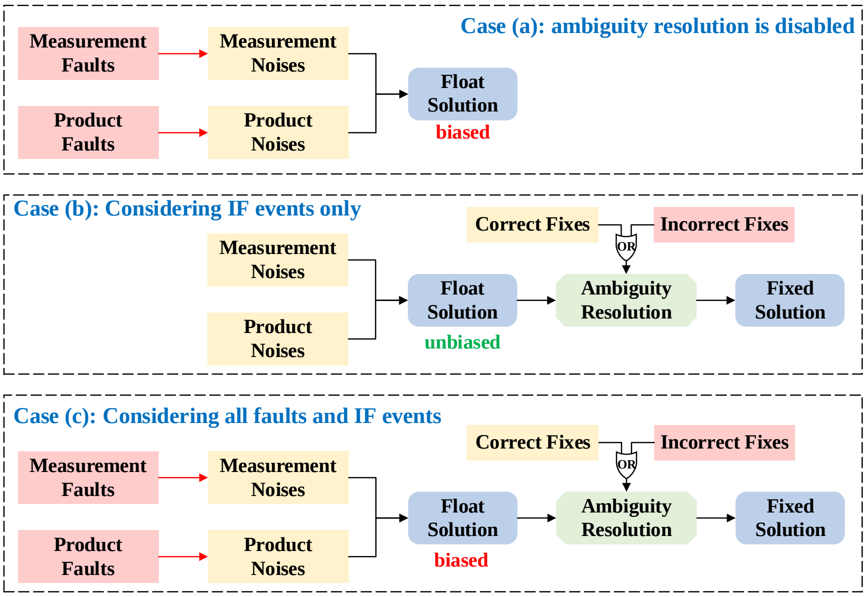

Fig. 1 *Threats for a PPP-RTK or RTK navigation engine*

Recently, there have been a few studies focusing on integrity monitoring of RTK, precise point positioning (PPP), and PPP-RTK. However, most of the prior studies focused on the first two cases shown in **Fig. 1**. To be more specific, these studies can be divided into two categories as follows:

- Only considering ambiguity-float solutions, i.e., case (a) in **Fig. 1.**

Gunning et al. designed an integrity monitoring algorithm for kinematic PPP based on solution separation [3], and then this work was extended to support tight integration of PPP and inertial measurement unit (IMU) [4]. Blanch et al. proposed several effective approaches to reduce the computational load of the PPP integrity monitoring algorithm [5].

- Only accounting for incorrect ambiguity resolution, i.e., case (b) in **Fig. 1.**

To quantify the risk from incorrect ambiguity fixes, Khanafseh et al. developed an approach to evaluate the position-domain integrity risk for cycle ambiguity resolution problems under a fault-free condition [6]. In addition, Green and Humphreys proposed a data-driven approach to evaluate the position-domain integrity risk of generalized integer aperture bootstrapping [7].

To conclude, existing studies either (a) only considered ambiguity-float solutions or (b) evaluated the risk from incorrect ambiguity fixes while assuming the measurements and correction products to be fault-free. Therefore, they are not applicable to the cases where measurement faults and/or product faults occur to ambiguity resolution enabled GNSS positioning systems. And these cases can neither be handled by a simple combination of existing methods (e.g., combining [3] and [6]), because the integrity risk coming from incorrect-fix events is affected by the occurrence of measurement faults and/or product faults.

In response, this paper describes an integrity monitoring algorithm to protect AR-enabled GNSS high-precision positioning (e.g., PPP-RTK and RTK) against measurement faults, product faults, and incorrect ambiguity fixes simultaneously. We use a simplified PPP-RTK model as an example to develop the integrity monitoring algorithm, but it is worth noting that this algorithm is also applicable to RTK with minor modifications. The simplified PPP-RTK model is established by ignoring all the spatial and temporal error correlations and considering the error models as accurately known.

The proposed algorithm is developed based on multiple hypothesis solution separation (MHSS), because it offers promising performance and provides a straightforward evaluation of protection levels (PLs). First, a list of fault modes that need to be monitored is determined. Each fault mode corresponds to a hypothesis about the health status of the GNSS measurements and the correction products. Then, this algorithm computes the position estimates for each of the monitored hypotheses. Fault detection is performed by comparing each fault-tolerant solution with the all-in-view one. And finally, the corresponding protection levels will be computed if there is no fault alarm. In this step, multiple incorrect-fix events are considered under each of the monitored hypotheses, and their contributions to the integrity risk are rigorously quantified. Please note that in this paper, the integer ambiguity resolution function is implemented using the integer bootstrapping (IB) method [8], because it offers a closed-form formula to calculate the prior probabilities of each incorrect-fix event.

The rest of this paper is organized as follows. Section 2 introduces the principles of ambiguity resolution enabled GNSS positioning techniques, and Section 3 presents the proposed integrity monitoring algorithm for AR-enabled GNSS positioning. Simulations are carried out in Section 4 to validate the proposed algorithm. And finally, Section 5 draws the conclusions.

2. BASIC PRINCIPLES OF AMBIGUITY RESOLUTION ENABLED GNSS POSITIONING

This section uses a simplified undifferenced, uncombined PPP-RTK model as an example to describe the principles of ambiguity resolution enabled GNSS high-precision positioning techniques.

2.1 Measurement Models, Correction Products, and State Dynamics

(1) Measurement Models

PPP-RTK uses the code and carrier-phase observations from a receiver to estimate the user's position and other related parameters. The code and carrier-phase observations are respectively formulated as follows:

$$\rho_{r,j}^{s,S} = \|\boldsymbol{X}^{s,S} - \boldsymbol{X}_r\| + dt_r - dt^{s,S} + isb_r^S + Mw_r^s \cdot T_r + \gamma_j^S \cdot I_{r,1}^{s,S} + (d_{r,j} - d_j^{s,S}) + \varepsilon_{r,j}^{s,S} + \xi_{r,j}^{s,S} \tag{1}$$

$$l_{r,j}^{s,S} \triangleq \lambda_j^{s,S}\Phi_{r,j}^{s,S} = \|\boldsymbol{X}^{s,S} - \boldsymbol{X}_r\| + dt_r - dt^{s,S} + isb_r^S + Mw_r^s \cdot T_r - \gamma_j^S \cdot I_{r,1}^{s,S} + \lambda_j^{s,S} \cdot (N_{r,j}^{s,S} + b_{r,j} - b_j^{s,S}) + \epsilon_{r,j}^{s,S} + \xi_{r,j}^{s,S} \tag{2}$$

where:

ρ, Φ, l	are pseudoranges, carrier phases, and the derived distance from carrier phases, respectively;
s, r, j, S	indicate the indexes of satellite, receiver, signal frequency, and constellation, respectively;
$\boldsymbol{X}$	represents the position of a receiver or a satellite;
dt	is the clock offset of receiver or a satellite in meters;
isb	is the inter-system bias in meters;
T	is the zenith troposphere wet delay;
Mw	is the mapping coefficient for troposphere wet delay;
I	denotes the slant ionosphere delay;
λ	is the signal wavelength;
γ	is a frequency-dependent factor, $\gamma_j = \lambda_j^2/\lambda_1^2$.
d	represents the code bias of a receiver or a satellite;
N	is the carrier-phase integer ambiguity of a signal;
ξ	represents the errors that can be eliminated using models, such as phase wind-up, satellite antenna offsets, etc.;
$\varepsilon,\ \epsilon$	are unmodeled code and carrier-phase white noises.

(2) Correction Products

In addition to the raw measurements above, PPP-RTK users also need to subscribe the SSR correction services, which include the following products: (a) precise satellite orbits, (b) precise satellite clocks, (c) satellite code biases, (d) satellite phase biases, (e) regional troposphere corrections, and (f) regional ionosphere corrections. These products can be modeled as:

$$\textbf{Orbit}:\ \widetilde{\boldsymbol{X}}^{s,S} = \boldsymbol{X}^{s,S} + \varepsilon_{orb}^{s,S};\quad \textbf{Clock}:\ \widetilde{dt}^{s,S} = dt^{s,S} + \varepsilon_{clk}^{s,S} \tag{3}$$

$$\textbf{Code bias:}\ \widetilde{d}_j^{s,S} = d_j^{s,S} + \varDelta_{d,j}^{S} + \varepsilon_{d,j}^{s,S};\quad \textbf{Phase bias}:\ \widetilde{b}_j^{s,S} = b_j^{s,S} + \varDelta_{b,j}^{S} + \varepsilon_{b,j}^{s,S} \tag{4}$$

$$\textbf{Troposphere}:\ \widetilde{T}_r = T_r + \varepsilon_{trp};\quad \textbf{Ionosphere}:\ \widetilde{I}_{r,1}^{s,S} = I_{r,1}^{s,S} + \varepsilon_{ion}^{s,S} \tag{5}$$

where ε_{orb} and ε_{clk} are the errors of orbit products and clock products, respectively, and their effects on the ranging measurements are bounded by the signal-in-space range error (SISRE), ε_{sis}. ε_d and ε_b represent the errors of satellite code bias and phase bias products, respectively. $\varDelta_d$ and $\varDelta_b$ are the PPP-RTK server-end systematic offsets of the satellite bias products. Finally, ε_{trp} and ε_{ion} are the errors of the troposphere and ionosphere products, respectively.

(3) State Vector

In the simplified PPP-RTK user positioning model, the state vector to be estimated is given as (using a dual-frequency dual-constellation configuration as an example):

$$\begin{aligned} \boldsymbol{x} &\triangleq [\boldsymbol{x}_{pos}^{\mathrm{T}}\ |\ \boldsymbol{x}_{clk}^{\mathrm{T}}\ |\ \boldsymbol{x}_{isb}^{\mathrm{T}}\ |\ \boldsymbol{x}_{bia}^{\mathrm{T}}\ |\ \boldsymbol{x}_{trp}^{\mathrm{T}}\ |\ \boldsymbol{x}_{ion}^{\mathrm{T}}\ |\ \boldsymbol{x}_{amb}^{\mathrm{T}}]^{\mathrm{T}} \\ &= [\boldsymbol{X}_r\ |\ dt^G\ |\ isb^C\ |\ dcb^G\ b_{r,1}^G\ b_{r,2}^G\ dcb^C\ b_{r,1}^C\ b_{r,2}^C\ |\ T_r\ |\ \boldsymbol{I}_{r,1}^G\ \boldsymbol{I}_{r,1}^C\ |\ \boldsymbol{N}_{r,1}^G\ \boldsymbol{N}_{r,2}^G\ \boldsymbol{N}_{r,1}^C\ \boldsymbol{N}_{r,2}^C]^{\mathrm{T}} \end{aligned} \tag{6}$$

where the superscript "G" denotes GPS and "C" represents Beidou. The second term dt^G is the receiver clock offset in meters. dcb represents the receiver differential code bias (DCB) between the second-frequency code measurements and the first-frequency ones. b_r denotes the receiver phase biases. $\boldsymbol{I}$ is the slant ionosphere delay vector, and $\boldsymbol{N}$ is the ambiguity vector.

(4) Stochastic Models

The stochastic error models for the GNSS raw measurements and the correction products are summarized in **Table 1**. Then, **Table 2** presents the error models for the initial state vector. Finally, **Table 3** shows the stochastic models for the state process noises. The values in these tables should be viewed as preliminary, and they are used for demonstration purposes only.

Table 1. Stochastic error models for the raw measurements and the SSR correction products

Error Sources	Variables	Error Models
Carrier phases	ϵ	$\sigma_{\epsilon}^2 = a_1^2 + a_2^2/\sin^2(\theta)$ $[m^2]$, where θ is the elevation angle; $a_1 = 0.005$, $a_2 = 0.005$
Pseudoranges	ε	$\sigma_{\varepsilon}^2 = a_3^2 \cdot \sigma_{\epsilon}^2$ $[m^2]$, $a_3 = 100$
SISRE	ε_{sis}	$\sigma_{sis} = 0.05$ $[m]$
Satellite code biases	ε_d	$\sigma_d = 0.02$ $[m]$
Satellite phase biases	ε_b	$\sigma_b = 0.02$ $[cycle]$
Troposphere corrections	ε_{trp}	$\sigma_{trp} = 0.10$ $[m]$
Ionosphere corrections	ε_{ion}	$\sigma_{ion} = 0.05$ $[m]$

Table 2. State initialization and initial state error covariances

States	Variables	Initial value	Initial covariance
Receiver position	$\boldsymbol{X}_r$	From SPP	$\mathbf{P}_{0,pos} = \mathrm{diag}(100, 100, 500)$ $[m^2]$
Receiver clock offset	dt_r	From SPP	$\mathbf{P}_{0,clk} = 10^4$ $[m^2]$
Receiver ISBs	isb	**0**	$\mathbf{P}_{0,isb} = 100$ $[m^2]$
Receiver DCBs	dcb	**0**	$\mathbf{P}_{0,dcb} = 100$ $[m^2]$
Receiver phase biases	b_r	**0**	$\mathbf{P}_{0,br} = 100$ $[cycle^2]$
Zenith wet troposphere delay	T_r	**0**	$\mathbf{P}_{0,trp} = 0.5$ $[m^2]$
Slant Ionosphere delays	$\boldsymbol{I}$	**0**	$\mathbf{P}_{0,ion} = 100$ $[m^2]$
Ambiguities	$\boldsymbol{N}$	**0**	$\mathbf{P}_{0,amb} = 10^6$ $[m^2]$

Table 3. Stochastic models for state process noises

States	Variables	Process Noise Models
Receiver position	$\boldsymbol{X}_r$	At each epoch, reinitialize it using SPP and set its covariance to $\mathbf{P}_{0,pos}$
Receiver clock offset	dt_r	At each epoch, reinitialize it using SPP and set its variance to $\mathbf{P}_{0,clk}$
Receiver ISBs	isb	Random walk: $\sigma_{q,isb} = 10^{-3}$ $[m]$
Receiver DCBs	dcb	Random walk: $\sigma_{q,dcb} = 10^{-3}$ $[m]$
Receiver phase biases	b_r	Random walk: $\sigma_{q,br} = 10^{-4}$ $[cycle]$
Zenith troposphere wet delay	T_r	Random walk: $\sigma_{q,trp} = 10^{-4}$ $[m]$
Slant Ionosphere delays	$\boldsymbol{I}$	Random walk: $\sigma_{q,ion} = 0.05$ $[m]$
Ambiguities	$\boldsymbol{N}$	Constant value: $\sigma_{q,amb} = 0$ $[cycle]$

(5) Assumptions

The simplified PPP-RTK model is established based on the models above and the following four assumptions: (a) all the error models are accurately known; (b) all the error terms follow zero-mean Gaussian distributions; (c) there is no correlation between any two error terms; (d) there is no temporal correlation for any error term. Although these assumptions may be too idealistic, they can be used to set up a preliminary integrity monitoring algorithm for AR-enabled GNSS positioning. In future work, we will improve the proposed algorithm to take temporal and spatial error correlations into account [9][10].

2.2 Ambiguity-Float State Estimation

Based on the functional models and error models above, the state vector in Equation (6) can be estimated using a Kalman filter (KF) or a sliding-window least-squares (SWLS) estimator. **Fig. 2** shows a comparison between these two estimators, and they will be briefly introduced below.

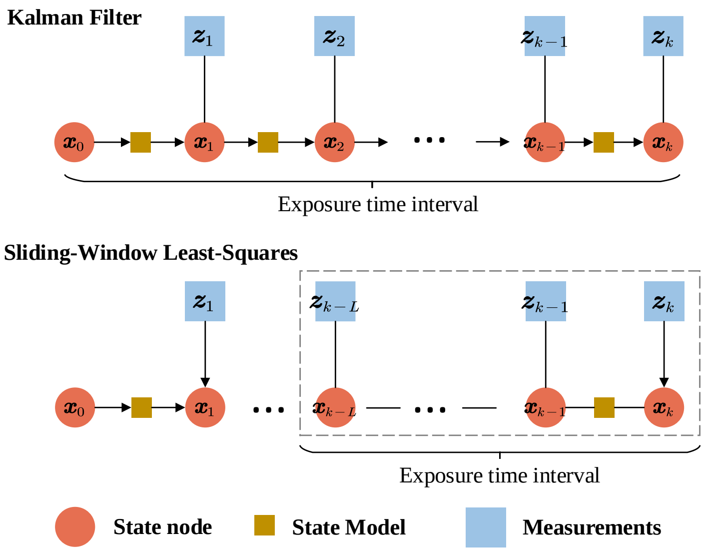

Fig. 2 *Comparison between the Kalman filter and the sliding-window least-squares estimator*

(1) Kalman Filter

The Kalman filter is a recursive estimator that includes two distinct phases in one iteration: predict and update. The predict phase uses the state estimate from the previous timestep and the state model to produce an estimate of the current state as follows:

$$\bar{\boldsymbol{x}}_k = \mathbf{F}_k \hat{\boldsymbol{x}}_{k-1} \tag{7}$$

$$\bar{\mathbf{P}}_k = \mathbf{F}_k \hat{\mathbf{P}}_{k-1} \mathbf{F}_k^{\mathrm{T}} + \mathbf{Q}_k \tag{8}$$

where $\bar{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}$ are respectively the predicted and updated state estimates; accordingly, $\bar{\mathbf{P}}$ and $\hat{\mathbf{P}}$ are respectively the predicted and updated state error covariance matrices. $\mathbf{F}_k$ denotes the state transition matrix from epoch (k-1) to epoch k, and $\mathbf{Q}_k$ is the process noise covariance matrix. $\mathbf{F}_k$ and $\mathbf{Q}_k$ can be determined based on **Table 3**.

In the update phase, the filter generates the updated state estimate by incorporating the new measurements, as shown below:

$$\mathbf{K}_k = \bar{\mathbf{P}}_k \mathbf{H}_k^{\mathrm{T}} \left(\mathbf{H}_k \bar{\mathbf{P}}_k \mathbf{H}_k^{\mathrm{T}} + \mathbf{R}_k \right)^{-1} \tag{9}$$

$$\hat{\boldsymbol{x}}_k = \bar{\boldsymbol{x}}_k + \mathbf{K}_k (\boldsymbol{z}_k - \mathbf{H}_k \bar{\boldsymbol{x}}_k) \tag{10}$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \bar{\mathbf{P}}_k (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^{\mathrm{T}} + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^{\mathrm{T}} \tag{11}$$

where $\boldsymbol{z}$ denotes the measurement vector, $\mathbf{H}$ represents the measurement matrix, $\mathbf{R}$ is the measurement noise covariance matrix, and $\mathbf{K}$ is the Kalman filter gain.

The measurement vector, the measurement matrix, and the measurement error covariance matrix are respectively given by:

$$\boldsymbol{z} \triangleq [\boldsymbol{z}_{obs}^{\mathrm{T}} \mid \boldsymbol{z}_{aug}^{\mathrm{T}}]^{\mathrm{T}} = \left[\tilde{\boldsymbol{\rho}}^{\mathrm{T}} \;\; \tilde{\boldsymbol{l}}^{\mathrm{T}} \mid \tilde{T}_r^{\mathrm{T}} \;\; \tilde{\boldsymbol{I}}_{r,1}^{\mathrm{T}} \right]^{\mathrm{T}} \tag{12}$$

$$\mathbf{H} \triangleq [\mathbf{H}_{obs}^{\mathrm{T}} \mid \mathbf{H}_{aug}^{\mathrm{T}}] = [\mathbf{H}_{\rho}^{\mathrm{T}} \;\; \mathbf{H}_{l}^{\mathrm{T}} \mid \mathbf{H}_{trp}^{\mathrm{T}} \;\; \mathbf{H}_{ion}^{\mathrm{T}}]^{\mathrm{T}} \tag{13}$$

$$\mathbf{R} \triangleq \begin{bmatrix} \mathbf{R}_{obs} & \\ & \mathbf{R}_{aug} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{obs} & & \\ & \mathbf{R}_{trp} & \\ & & \mathbf{R}_{ion} \end{bmatrix} \tag{14}$$

where $\tilde{\boldsymbol{\rho}}$ denotes the corrected pseudorange vector, $\tilde{\boldsymbol{l}}$ is the corrected carrier phase vector. To be more specific, when calculating the innovation vector for pseudoranges and carrier phases, the raw code and carrier-phase measurements are corrected by (a) eliminating the error terms that can be modeled, i.e., ξ, (b) employing the precise orbit and clock products, and (c) applying the satellite code and phase bias products. Therefore, the errors existing in (b) and (c) should be considered in the determination of $\mathbf{R}_{obs}$. For the sake of brevity, the detailed formulas of $\mathbf{H}$ and $\mathbf{R}$ are not shown here, and they can be easily obtained based on Equations (1) ~ (2) and **Table 1**.

Finally, special attention should be paid to the satellite code and phase bias products. As shown in Equation (4), there are systematic offsets in the satellite signal bias products, which come from the receiver network of the PPP-RTK server end. Fortunately, it can be verified that these offsets will not influence the position states and the ambiguity states, because they will be absorbed by the receiver clock parameters ($\boldsymbol{x}_{clk}$ and $\boldsymbol{x}_{isb}$) and the receiver signal bias parameters ($\boldsymbol{x}_{bia}$). Therefore, the estimated ambiguity vector maintains its integer nature, and the position error is unbiased.

(2) Sliding-Window Least-Squares (SWLS) Estimator

The KF employs all the current and past information to estimate the current state. In contrast, the SWLS estimator generates the optimal estimation on the current state by only using the information in a time window. From an integrity perspective, the SWLS has an advantage over the KF in that it limits the exposure time to the measurements occurring farther in the past than a certain time. If the exposure time grows over time, as is the case with a KF, the prior probability of measurement or product faults occurring will keep becoming larger, thereby potentially increasing the integrity risk [4].

The implementation of a SWLS estimator for PPP-RTK is described as follows. The batch measurement model is given by:

$$\mathbb{Z} = \mathbb{H}\mathbb{X} + \mathbb{V} \Leftrightarrow \begin{bmatrix} \boldsymbol{z}_{k-L+1} \\ \vdots \\ \boldsymbol{z}_{k-1} \\ \boldsymbol{z}_k \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{k-L+1} & & & \\ & \ddots & & \\ & & \mathbf{H}_{k-1} & \\ & & & \mathbf{H}_k \\ -\mathbf{F}_{k-L+2} & & & \\ & \ddots & & \\ & & -\mathbf{F}_k & \mathbf{I} \end{bmatrix} \times \begin{bmatrix} \boldsymbol{x}_{k-L+1} \\ \vdots \\ \boldsymbol{x}_{k-1} \\ \boldsymbol{x}_k \end{bmatrix} + \begin{bmatrix} \boldsymbol{v}_{k-L+1} \\ \vdots \\ \boldsymbol{v}_{k-1} \\ \boldsymbol{v}_k \\ \boldsymbol{\omega}_{k-L+2} \\ \vdots \\ \boldsymbol{\omega}_k \end{bmatrix} \tag{15}$$

$$\mathbb{V} \sim \mathcal{N}(0,\ \mathbb{R}) \Leftrightarrow \begin{bmatrix} \boldsymbol{v}_{k-L+1} \\ \vdots \\ \boldsymbol{v}_{k-1} \\ \boldsymbol{v}_k \\ \boldsymbol{\omega}_{k-L+2} \\ \vdots \\ \boldsymbol{\omega}_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} \mathbf{R}_{k-L+1} & & & & & \\ & \ddots & & & & \\ & & \mathbf{R}_{k-1} & & & \\ & & & \mathbf{R}_k & & \\ & & & & \mathbf{Q}_{k-L+2} & \\ & & & & & \ddots & \\ & & & & & & \mathbf{Q}_k \end{bmatrix} \right) \tag{16}$$

where L is the length of the sliding window; $\boldsymbol{v}_k$ and $\boldsymbol{\omega}_k$ denote the measurement noise vector and the process noise vector at epoch k, respectively; $\mathcal{N}(\blacktriangle, \blacktriangledown)$ represents the normal distribution with a mean $\blacktriangle$ and a covariance $\blacktriangledown$.

The measurement model above can be solved using a standard weighted least-squares estimator, as shown below:

$$\hat{\mathbb{X}} = (\mathbb{H}^{\mathrm{T}}\mathbb{R}^{-1}\mathbb{H})^{-1}\mathbb{H}^{\mathrm{T}}\mathbb{R}^{-1}\mathbb{Z} \tag{17}$$

$$\hat{\mathbb{P}} = (\mathbb{H}^{\mathrm{T}}\mathbb{R}^{-1}\mathbb{H})^{-1} \tag{18}$$

where $\hat{\mathbb{P}}$ denotes the error covariance matrix for the estimated state vector $\hat{\mathbb{X}}$. The state estimate $\hat{\boldsymbol{x}}_k$ and its error covariance $\hat{\mathbf{P}}_k$ for the current epoch can be simply extracted from $\hat{\mathbb{X}}$ and $\hat{\mathbb{P}}$.

2.3 Ambiguity Resolution

After obtaining the ambiguity-float solution using the approaches above, the positioning accuracy of PPP-RTK can be further improved by taking advantage of the integer nature of the carrier-phase cycle ambiguities. This can be achieved by implementing ambiguity resolution, which aims at fixing the float ambiguity vector to an integer vector. In this paper, the ambiguities are resolved using the integer bootstrapping (IB) approach [8]. In addition, the ambiguity resolution is performed in a "continuous" mode in the Kalman filter, i.e., only the float solution is passed forward to the next epoch.

The integer bootstrapping-based ambiguity resolution includes the following steps. First, extract the float ambiguity vector $\hat{\boldsymbol{a}} = \hat{\boldsymbol{x}}_{amb}$ and its covariance $\hat{\mathbf{P}}_{\boldsymbol{aa}} = \hat{\mathbf{P}}_{amb}$ from $\hat{\boldsymbol{x}}_k$ and $\hat{\mathbf{P}}_k$. Then, perform the decorrelation (or called "reduction") process to the original ambiguity vector based on the following Z-transformations [11]:

$$\hat{\mathbf{P}}'_{\boldsymbol{aa}} = \mathbf{Z}^{\mathrm{T}} \hat{\mathbf{P}}_{\boldsymbol{aa}} \mathbf{Z}, \ \hat{\boldsymbol{a}}' = \mathbf{Z}^{\mathrm{T}} \hat{\boldsymbol{a}} \tag{19}$$

where $\mathbf{Z}$ is the transformation matrix; $\hat{\boldsymbol{a}}'$ denotes the ambiguity vector after decorrelation, and $\hat{\mathbf{P}}'_{\boldsymbol{aa}}$ is its error covariance.

Next, implement the integer bootstrapping method to fix the decorrelated ambiguity vector $\hat{\boldsymbol{a}}'$ to the integer vector $\breve{\boldsymbol{a}}'$. This process can be explained as follows. The IB method fixes ambiguities sequentially. The order is determined according to the ambiguity conditional variances, and the ambiguity having the lowest conditional variance is fixed first. The $\mathbb{i}$th conditional variance ($\sigma^2_{\mathbb{i}|\mathbb{I}}$), defined as the variance of the $\mathbb{i}$th ambiguity conditioned on the previous ambiguities in the set $\mathbb{I} = \{1, 2, ..., \mathbb{i}-1\}$ being fixed, is given by the $\mathbb{i}$th diagonal element of a diagonal matrix $\mathbf{D}$, and $\mathbf{D}$ is obtained by:

$$\hat{\mathbf{P}}'_{\boldsymbol{aa}} = \mathbf{L}^{\mathrm{T}} \mathbf{D} \mathbf{L} \tag{20}$$

where $\mathbf{L}$ is a triangular matrix.

In the fourth step, evaluate the success rate (i.e., the probability of correct fix) of this ambiguity resolution process by [8]:

$$P_S \triangleq P(\breve{\boldsymbol{a}}' = \boldsymbol{a}'_0) = \prod_{\mathbb{i}=1}^{m} \left(2\Phi\left(\frac{1}{2\sigma_{\mathbb{i}|\mathbb{I}}}\right) - 1 \right) \tag{21}$$

where m is the number of ambiguities in the ambiguity vector $\breve{\boldsymbol{a}}'$; $\boldsymbol{a}'_0$ denotes the true value of $\boldsymbol{a}'$; Φ represents the cumulative distribution function (CDF) of a zero-mean unit normal distribution, i.e., $\Phi(\blacksquare) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\blacksquare} \exp\left(-\frac{t^2}{2}\right) dt$.

And in the fifth step, obtain the fixed value of the original ambiguity vector by the following:

$$\breve{\boldsymbol{a}} = \mathbf{Z}^{-\mathrm{T}} \breve{\boldsymbol{a}}' \tag{22}$$

Finally, use the fixed ambiguities to correct the "float" position estimate, $\hat{\boldsymbol{b}} = \boldsymbol{x}_{pos}$. As a result, one can obtain the fixed position estimate and its error covariance matrix by [11]:

$$\breve{\boldsymbol{b}} = \hat{\boldsymbol{b}} - \hat{\mathbf{P}}_{\boldsymbol{ba}} \hat{\mathbf{P}}^{-1}_{\boldsymbol{aa}} (\hat{\boldsymbol{a}} - \breve{\boldsymbol{a}}) \tag{23}$$

$$\breve{\mathbf{P}}_{\boldsymbol{bb}} = \hat{\mathbf{P}}_{\boldsymbol{bb}} - \hat{\mathbf{P}}_{\boldsymbol{ba}} \hat{\mathbf{P}}^{-1}_{\boldsymbol{aa}} \hat{\mathbf{P}}^{\mathrm{T}}_{\boldsymbol{ba}} \tag{24}$$

where $\hat{\mathbf{P}}_{\boldsymbol{ba}}$ denotes the error covariance matrix between $\boldsymbol{x}_{pos}$ and $\boldsymbol{x}_{amb}$, and $\hat{\mathbf{P}}_{\boldsymbol{bb}}$ is the error covariance of $\boldsymbol{x}_{pos}$, i.e., $\hat{\mathbf{P}}_{\boldsymbol{bb}} = \hat{\mathbf{P}}_{pos}$. Both $\hat{\mathbf{P}}_{\boldsymbol{ba}}$ and $\hat{\mathbf{P}}_{\boldsymbol{bb}}$ can be extracted from $\hat{\mathbf{P}}_k$. Accordingly, $\breve{\mathbf{P}}_{\boldsymbol{bb}}$ is the error covariance of the fixed position estimate, i.e., $\breve{\mathbf{P}}_{pos} = \breve{\mathbf{P}}_{\boldsymbol{bb}}$. This completes the whole process for the IB-based ambiguity resolution.

3. MHSS-BASED INTEGRITY MONITORING FOR AR-ENABLED GNSS POSITIONING

The PPP-RTK user positioning systems serving autonomous systems may encounter occasional faults from the GNSS raw measurements and/or the correction products. In addition, they might experience large errors when there are incorrectly fixed ambiguities. To ensure high integrity of PPP-RTK, this section describes an integrity monitoring algorithm to address these threats by employing multiple hypothesis solution separation. The overall flowchart of this algorithm is shown in **Fig. 3**, and the descriptions of each step will be provided in the following subsections.

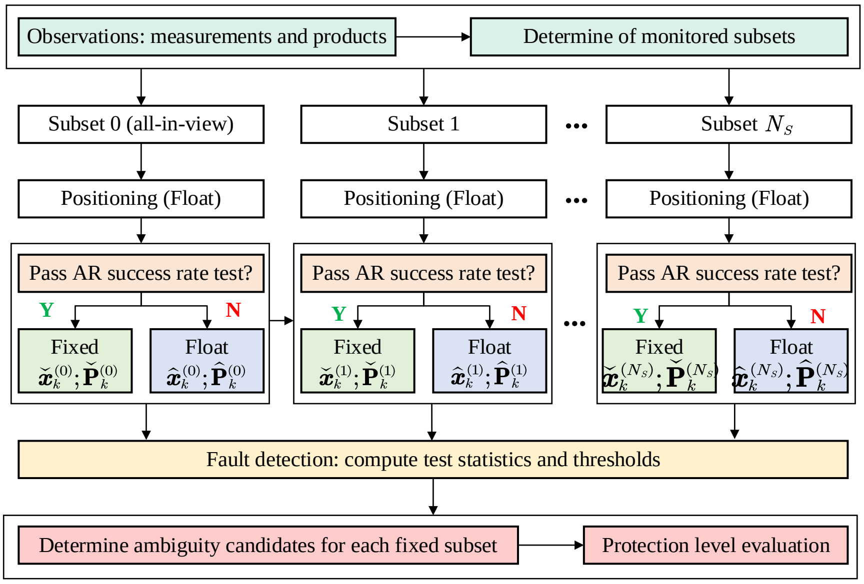

Fig. 3. *The flowchart of the proposed integrity monitoring algorithm for PPP-RTK positioning*

3.1 Threats for PPP-RTK User Positioning

Table 4. Fault Sources for PPP-RTK Positioning Systems

Sources	**Failures**
Receiver Measurements	Heavy multipath interference or non-line-of-sight (NLOS) reception
	Undetected cycle slips
	Ionosphere scintillation
Correction Products	Incorrect orbit or clock products
	Incorrect satellite bias products
	Incorrect troposphere or ionosphere products

Table 4 summarizes the possible sources of faults that may threaten PPP-RTK positioning systems. In this paper, all the fault events that affect a satellite are grouped to be the event that a satellite is faulted due to any causes. Let $\bar{S}_i$ denote the event that satellite i is faulted and $P_{sat,i}$ be its prior probability. Please note that a satellite is called to be faulted if it is faulted at any epoch over the exposure time interval. In addition, it is assumed that $\bar{S}_i$ is independent of $\bar{S}_j$ for any combination of i and j $(i \neq j)$. In future work, the integrity monitoring algorithm will be modified to be free of this assumption.

3.2 Determination of the Fault Modes that Need to be Monitored

To implement MHSS, an essential step is to determine the fault modes that are worth monitoring [12]. In this work, a fault mode is a hypothesis about the health status of each satellite, and it is accompanied with a subset that includes all the measurements and products associated with the healthy satellites. This work adopts the method given in [12] to determine the fault modes that need to be monitored and to compute the probability from unmonitored faults, $P_{H,NM}$.

For a clearer illustration, the following is an example of a fault mode, $\mathcal{H}^{(1)}$:

$$\mathcal{H}^{(1)} = \bar{\mathcal{S}}_1 \mathcal{S}_2 \mathcal{S}_3 \cdots \mathcal{S}_n \tag{25}$$

where $\mathcal{S}_i$ denotes the event that satellite i is healthy. The prior probability of $\mathcal{H}^{(1)}$ is given by:

$$P_{\mathcal{H}}^{(1)} = P_{sat,1}\left(1 - P_{sat,2}\right)\left(1 - P_{sat,3}\right)\cdots\left(1 - P_{sat,n}\right) \tag{26}$$

And the estimator associated with $\mathcal{H}^{(1)}$ will discard the measurements and products of satellite 1 over its exposure time interval.

3.3 Ambiguity Resolution Success Rate Test and Fault Detection

Let i index the monitored hypotheses, i=0, 1, …, N_S, where i=0 represents the fault-free hypothesis. For each monitored hypothesis i, the corresponding ambiguity-float navigation solution $\hat{\boldsymbol{x}}_k^{(i)}$ and its error covariance $\hat{\mathbf{P}}_k^{(i)}$ can be produced using the approaches shown in Section 2.2. Hereafter, the subscript k is omitted to lighten the notations.

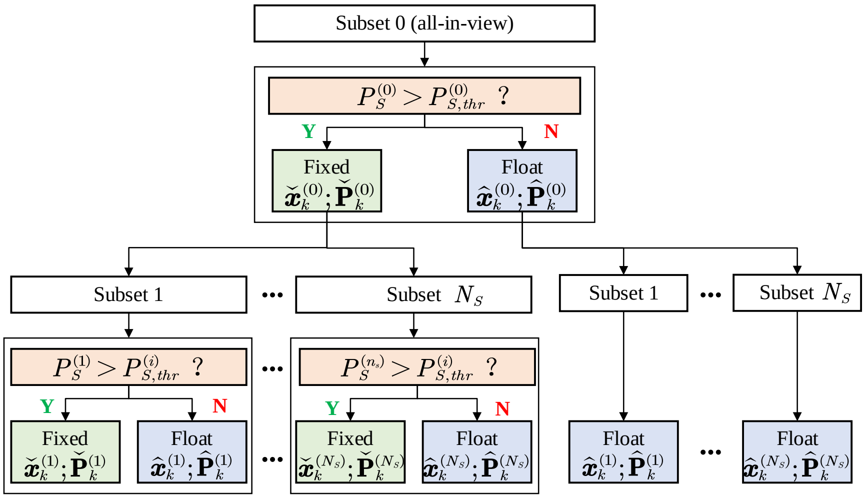

Fig. 4. *Flowchart of the proposed ambiguity resolution success rate tests*

Then, for each hypothesis i from 0 to N_S, the proposed algorithm will determine whether the float ambiguities should be fixed according to the AR success rate given in Equation (21). Although an ambiguity vector can always be fixed by the IB method, it is not recommended to do so if the success rate is low. This is because in this case, the fixed solution might be not reliable due to the occurrence of incorrect-fix events. In addition, as will be shown in the next subsection, a large number of ambiguity candidates (including one correct-fix event and various incorrect-fix events) need to be monitored in this case, which will make the computational load unacceptably large.

The proposed AR success rate tests are illustrated as follows and are summarized in **Fig. 4**. Let $P_{S,thr}^{(0)}$ be the success rate threshold for the all-in-view subset and $P_{S,thr}^{(i)}$ be the one for other subsets. For example, $P_{S,thr}^{(0)}$=0.9999 and $P_{S,thr}^{(i)}$=0.99. As shown in **Fig. 4**, if the all-in-view solution is not fixed, then other solutions should also remain float. This strategy is designed to ensure that the information used by each fault-tolerant estimator is a subset of the information used in the all-in-view estimator.

Next, fault detection is performed based on MHSS. For each fault mode, there are three solution separation tests, one for each position component. The construction of the test statistics and test thresholds varies with the fix status of the all-in-view solution and the fault-tolerant solutions. Therefore, in the following, we will discuss all the possible cases one by one.

(1) The all-in-view solution is not fixed

In this case, all the solutions are float, and thus the test statistics are given by:

$$\Delta x_q^{(i)} = \hat{x}_q^{(i)} - \hat{x}_q^{(0)},\ i = 1, 2, \ldots N_S \tag{27}$$

where q=1, 2, 3 representing east, north, and up position coordinates, respectively. The solution separation variance is given by:

$$\sigma_{ss,q}^{(i)2} = \hat{\mathbf{P}}_{q,q}^{(i)} - \hat{\mathbf{P}}_{q,q}^{(0)} \tag{28}$$

where $\mathbf{P}_{q,q}$ is the variance of the $q\text{th}$ position coordinate. Then the test threshold is determined by:

$$T_q^{(i)} = Q^{-1}\left(\frac{P_{FA,q}}{2 \cdot N_S}\right) \cdot \sigma_{ss,q}^{(i)} \tag{29}$$

where $Q^{-1}(\blacksquare)$ is the $(1 - \blacksquare)$ quantile of a zero-mean unit normal distribution; $P_{FA,q}$ denotes the false alert budget allocated to the $q\text{th}$ position component.

(2) The all-in-view solution is fixed

For each fault mode i $(i = 1, 2, \ldots, N_S)$, the test statistics and thresholds are computed by the following:

$$\Delta x_q^{(i)} = \begin{cases} \check{x}_q^{(i)} - \check{x}_q^{(0)}, & \text{if solution } i \text{ is fixed} \\ \hat{x}_q^{(i)} - \check{x}_q^{(0)}, & \text{otherwise} \end{cases} \tag{30}$$

And accordingly, the solution separation variance is computed by:

$$\sigma_{ss,q}^{(i)2} = \begin{cases} \check{\mathbf{P}}_{q,q}^{(i)} - \check{\mathbf{P}}_{q,q}^{(0)}, & \text{if solution } i \text{ is fixed} \\ \hat{\mathbf{P}}_{q,q}^{(i)} - \check{\mathbf{P}}_{q,q}^{(0)}, & \text{otherwise} \end{cases} \tag{31}$$

Finally, the test threshold can be determined using (29). However, it is noteworthy that with this method, the continuity risk from false alert events is approximate to (but not exactly equal to) the target false alert budget, because the incorrect-fix events under the fault-free condition are not considered. This issue will be addressed in our future work. Please also note that this will not influence the evaluation of the integrity risk or the protection levels.

The fault detection test is said to pass only if for all i and q we have the following:

$$\tau_q^{(i)} = \frac{|\Delta x_q^{(i)}|}{T_q^{(i)}} \leqslant 1 \tag{32}$$

If any test fails, the integrity monitoring algorithm will issue a fault alarm to the users.

3.4 Determination of the Ambiguity Candidates that Need to be Monitored

To realize the integrity monitoring for AR-enabled GNSS positioning, the effect of incorrect-fix events on the integrity risk must be taken into account. Therefore, for each monitored hypothesis, we need to establish a list of the ambiguity candidates that need to be considered. The monitored ambiguity candidates include one correct-fix candidate and multiple incorrect-fix candidates. This subsection provides a preliminary method of ambiguity candidate determination as follows.

For hypothesis i, the float ambiguity vector $\hat{\boldsymbol{a}}^{(i)}$ follows a zero-mean normal distribution, i.e., $\hat{\boldsymbol{a}}^{(i)} \sim \mathcal{N}\left(0, \hat{\mathbf{P}}_{\boldsymbol{aa}}^{(i)}\right)$. Then, $\hat{\boldsymbol{a}}^{(i)}$ is decorrelated using Equation (19), with $\mathbf{Z}^{(i)}$ being the transformation matrix. The decorrelated ambiguity vector $\hat{\boldsymbol{a}}^{(i)'}$ follows $\mathcal{N}\left(0, \hat{\mathbf{P}}_{\boldsymbol{aa}}^{(i)'}\right)$. To evaluate the probability of the incorrect-fix events, we first perform a $\mathrm{L^TDL}$ decomposition as $\hat{\mathbf{P}}_{\boldsymbol{aa}}^{(i)'} = \mathbf{L}^{(i)\mathrm{T}} \mathbf{D}^{(i)} \mathbf{L}^{(i)}$. Then the probability that the vector $\hat{\boldsymbol{a}}^{(i)'}$ is fixed to an arbitrary ambiguity candidate $\boldsymbol{a}_j^{(i)'}$ is given by [8]:

$$P_{A,j}^{(i)} \triangleq P\left(\breve{\boldsymbol{a}}^{(i)\prime} = \boldsymbol{a}_j^{(i)\prime}\right) = \prod_{\mathbb{i}=1}^{m_i}\left(\Phi\left(\frac{1-2\boldsymbol{l}_{\mathbb{i}}^{(i)\mathrm{T}}\left(\boldsymbol{a}_0^{(i)\prime}-\boldsymbol{a}_j^{(i)\prime}\right)}{2\sigma_{\mathbb{i}|\mathbb{I}}^{(i)}}\right)+\Phi\left(\frac{1+2\boldsymbol{l}_{\mathbb{i}}^{(i)\mathrm{T}}\left(\boldsymbol{a}_0^{(i)\prime}-\boldsymbol{a}_j^{(i)\prime}\right)}{2\sigma_{\mathbb{i}|\mathbb{I}}^{(i)}}\right)-1\right) \tag{33}$$

where m_i is the length of $\hat{\boldsymbol{a}}^{(i)\prime}$; $\boldsymbol{a}_0^{(i)\prime}$ denotes the truth for $\hat{\boldsymbol{a}}^{(i)\prime}$. The vector $\boldsymbol{l}_{\mathbb{i}}^{(i)\mathrm{T}}$ is the $\mathbb{i}$th row vector of $\left(\mathbf{L}^{(i)}\right)^{-\mathrm{T}}$; $\sigma_{\mathbb{i}|\mathbb{I}}^{(i)}$ is equal to square root of the $\mathbb{i}$th diagonal element of $\mathbf{D}^{(i)}$.

Algorithm 1: Establish a List of the Ambiguity Candidates that Need Considering

Inputs: $\mathbf{L}^{(i)}$, $\mathbf{D}^{(i)}$, m_i;

$P_{A,NM,thr}^{(i)}$ — the threshold for the probability from unmonitored candidates;

$P_{A,thr}^{(i)}$ — the threshold for the probability of one candidate, e.g., 10^{-13};

Step 1: Set a proper bound on the offset of an ambiguity, d, e.g., d=3.

Step 2:

```
C^(i) = [0, 0, ..., 0]^T;  P_A^(i) = P_S^(i);  P_A,NM^(i) = 1 − P_S^(i);  C_old = C^(i); % Initialize
for 𝕚 in [1, 2, ..., m_i]    % 𝕚 is the number of incorrect-fix ambiguities
  C𝕚 = [∅]; P_𝕚 = [∅]; % stores the ambiguity offsets and their probabilities
  for 𝕛 in [−d, −d+1, ..., −1, 1, 2, ..., d−1, d]   % offset value
    for c in each column of C_old
      for 𝕜 in [1, 2, ..., m_i]
        c′ = c;
        if c′(𝕜) equals to 0
          c′(𝕜) = 𝕛;
          if f(c′) > 0    % a positive offset vector
            calculate P_A^(i) = the probability of c′ occurring using (33);
            if P_A^(i) > P_A,thr^(i)   % do not monitor low-probability candidates
              append c′ to C𝕚; append P_A^(i) to P_𝕚; % a new candidate
            end
          end
        end
      end
    end
  end
append −C𝕚 to C𝕚; append P_𝕚 to P_𝕚;   % append the negative offsets
P_A,NM^(i) = P_A,NM^(i) − sum(P_𝕚); % update the probability from unmonitored offsets
if P_A,NM^(i) < P_A,NM,thr^(i)        % a threshold for the probability from unmonitored offsets
  append C𝕚 to C^(i); append P_𝕚 to P_A^(i);
  C_old = C𝕚; % for next 𝕚
  break;
end
end
```

Outputs: $\mathbf{C}^{(i)}$ — each column of it is an ambiguity offset vector;

$\boldsymbol{P}_A^{(i)}$ — each entry of it is the probability of the associated offset vector;

$P_{A,NM}^{(i)}$ — the sum of the probabilities from unmonitored offsets.

For the convenience of the following derivations, let us replace $\boldsymbol{a}_0^{(i)\prime}-\boldsymbol{a}_j^{(i)\prime}$ with $\Delta_j^{(i)\prime}$, which represents the offset between the estimated integer ambiguity vector and the truth. For example, $\Delta_0^{(i)\prime}=[0,0,...,0]^{\mathrm{T}}$ (i.e., the correct fix) and $\Delta_1^{(i)\prime}=[1,0,...,0]^{\mathrm{T}}$. According to Equation (33), we can find that for $\Delta_j^{(i)\prime}$ and $-\Delta_j^{(i)\prime}$, their probabilities of occurrence are exactly the same. Therefore, let us introduce another index, j', such that $\Delta_{j'}^{(i)\prime}=-\Delta_j^{(i)\prime}$. And we call $\Delta_j^{(i)\prime}$ and $\Delta_{j'}^{(i)\prime}$ as a positive-negative pair of ambiguity offsets, where $\Delta_j^{(i)\prime}$ is called a positive offset. To distinguish between $\Delta_j^{(i)\prime}$ and $\Delta_{j'}^{(i)\prime}$, we define a function that satisfy the following:

$$f(\Delta_j^{(i)\prime})>0;\ f(\Delta_{j'}^{(i)\prime})=-f(\Delta_j^{(i)\prime});\ f(\Delta_0^{(i)\prime})=0 \tag{34}$$

With the fundamentals above, we provide a preliminary method to establish a list of the ambiguity candidates (or offsets) that are worth considering for fault mode i. The algorithm is summarized in **Algorithm 1**. The basic idea of this algorithm is a step-by-step search with three loops: *the outer loop* is about the number of incorrect-fix ambiguities, *the middle loop* is about the values of the offsets for the incorrect-fix ambiguities, and *the inner loop* is about which ambiguities are incorrectly fixed. The search will quit once the sum of the probabilities of unmonitored candidates, $P_{A,NM}^{(i)}$, is smaller than the preset threshold $P_{A,NM,thr}^{(i)}$. It is noteworthy that with this algorithm, the ambiguity offsets that need to be considered include 1 zero offset (i.e., $\Delta_0^{(i)\prime}$) and $n_a^{(i)}$ positive-negative pairs (i.e., $\Delta_j^{(i)\prime}$ and $\Delta_{j'}^{(i)\prime}$, $j=1,2,...,n_a^{(i)}$). Therefore, the total number of the monitored incorrect-fix events for fault mode i is given by:

$$N_A^{(i)}=2\times n_a^{(i)} \tag{35}$$

Because $\Delta_j^{(i)\prime}$ and $\Delta_{j'}^{(i)\prime}$ are the offsets for the decorrelated ambiguity vector, we need to determine the corresponding offsets for the original ambiguity vector, i.e., $\Delta_j^{(i)}=\check{\boldsymbol{a}}_j^{(i)}-\boldsymbol{a}_0^{(i)}$ and $\Delta_{j'}^{(i)}=-\Delta_j^{(i)}$. This can be achieved by:

$$\Delta_j^{(i)}=(\mathbf{Z}^{(i)})^{-\mathrm{T}}\Delta_j^{(i)\prime},\quad \Delta_{j'}^{(i)}=(\mathbf{Z}^{(i)})^{-\mathrm{T}}\Delta_{j'}^{(i)\prime} \tag{36}$$

$\Delta_j^{(i)\prime}$ and $\Delta_{j'}^{(i)\prime}$ will be used in the next subsection to evaluate the integrity risk from incorrect-fix events.

3.5 Evaluation of the Protection Levels

If there is no fault alarm, the corresponding protection levels (PLs) of PPP-RTK positioning can be evaluated. The PL represents a safety-assured error bound, of which the mathematical definition is given by [12]:

$$P_{HMI,q}\times\left(1-\frac{P_{NM}}{P_{HMI}}\right)=P\left(|\mathring{x}_q^{(0)}-x_q|>PL_q,\bar{\mathcal{D}}\right) \tag{37}$$

where:

P_{HMI} denotes the total target integrity risk requirement;

$P_{HMI,q}$ is the integrity risk requirement allocated to the $q\,\mathrm{th}$ position component;

P_{NM} is the sum of the probabilities of all the unmonitored events;

$\mathring{x}_q^{(0)}$ is the all-in-view solution; if it is fixed, then $\mathring{x}_q^{(0)}=\check{x}_q^{(0)}$; otherwise, $\mathring{x}_q^{(0)}=\hat{x}_q^{(0)}$.

x_q is the truth for the $q\,\mathrm{th}$ position component;

$\bar{\mathcal{D}}$ is the event that there is no fault alarm;

PL_q represents the protection level for $q\,\mathrm{th}$ position component.

The evaluation of the protection levels varies with the fix status of the all-in-view solution and the fault-tolerant solutions. Therefore, in the following, we will discuss all the possible cases one by one.

(1) The all-in-view solution is not fixed

In this case, all the solutions are float, and thus the formula to evaluate the PLs is the same as that for ambiguity-float PPP-RTK positioning. To be more specific, the PL can be computed using the baseline MHSS algorithm, as shown below [12]:

$$P_{HMI,q} \times \left(1 - \frac{P_{NM}}{P_{HMI}}\right) = 2Q\left(\frac{PL_q}{\widehat{\sigma}_q^{(0)}}\right) + \sum_{i=1}^{n_s} Q\left(\frac{PL_q - T_q^{(i)}}{\widehat{\sigma}_q^{(i)}}\right) \cdot P_{\mathcal{H}}^{(i)} \tag{38}$$

with $P_{NM} = P_{H,NM}$ and $\widehat{\sigma}_q^{(i)2} = \widehat{\mathbf{P}}_{q,q}^{(i)}$. In this equation, Q is the tail probability of a zero-mean unit normal distribution.

(2) The all-in-view solution is fixed

If the all-in-view solution is fixed, the effects of the incorrect-fix events must be considered in the evaluation of the protection levels. Let us first consider the case where all the solutions are fixed. In this case, multiple incorrect-fix events are taken into account under each of the monitored fault modes. This is achieved by a two-layer total probability formula as follows:

$$\begin{aligned} P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}}\right) &= \sum_{i=0}^{N_S} P_{\mathcal{H}}^{(i)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}} \,\middle|\, \mathcal{H}_i\right) \\ &= \sum_{i=0}^{N_S} P_{\mathcal{H}}^{(i)} \cdot \sum_{j=0}^{N_A^{(i)}} P_{A,j}^{(i)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}} \,\middle|\, (\Delta^{(i)} = \Delta_j^{(i)} \mid \mathcal{H}_i)\right) \end{aligned} \tag{39}$$

In this equation, the first-layer hypotheses are the monitored fault modes, and the second-layer events are the monitored ambiguity candidates for each monitored fault mode.

Then Equation (39) is modified as follows to address a more general case where the subset solutions can be fixed or float:

$$\begin{aligned} &P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}}\right) = \sum_{i=0}^{N_S} P_{\mathcal{H}}^{(i)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}} \,\middle|\, \mathcal{H}_i\right) \\ &= \sum_{i \in I_F} P_{\mathcal{H}}^{(i)} \cdot \sum_{j=0}^{N_A^{(i)}} P_{A,j}^{(i)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}} \,\middle|\, (\Delta^{(i)} = \Delta_j^{(i)} \mid \mathcal{H}_i)\right) + \sum_{i \in I_{NF}} P_{\mathcal{H}}^{(i)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}} \,\middle|\, \mathcal{H}_i\right) \end{aligned} \tag{40}$$

where I_F includes the indexes of the subsets with a fixed solution, and I_{NF} consists of the indexes of the subsets that output a float solution.

Finally, we provide a formula to evaluate the PLs of AR-enabled PPP-RTK positioning systems as follows:

$$P_{HMI,q} \times \left(1 - \frac{P_{NM}}{P_{HMI}}\right) = IR^{(0)}(PL_q) + \sum_{i>0, i \in I_F} IR^{(i)}(PL_q) + \sum_{i \in I_{NF}} IR^{(i)}(PL_q) \tag{41}$$

where the first term of the right-hand side denotes the contribution of the all-in-view fault mode to the integrity risk, the second term is the contribution from the fault-tolerant fault modes that generate a fixed solution, and the last term is the contribution from the fault modes that output a float solution. These terms are respectively computed by:

$$IR^{(0)}(PL_q) = 2 \cdot P_{\mathcal{H},0} \cdot \sum_{j=0}^{N_A^{(0)}} P_{A,j}^{(0)} \cdot Q\left(\frac{PL_q - \mathbf{A}_q^{(0)} \Delta_j^{(0)}}{\breve{\sigma}_q^{(0)}}\right) \tag{42}$$

$$IR^{(i)}(PL_q) = P_{\mathcal{H}}^{(i)} \cdot \left[Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right) + \sum_{j=0}^{N_A^{(i)}} P_{A,j}^{(i)} \cdot Q\left(\frac{PL_q - T_q^{(i)} - \mathbf{A}_q^{(i)} \Delta_j^{(i)}}{\breve{\sigma}_q^{(i)}}\right)\right], \quad i > 0 \text{ and } i \in I_F \tag{43}$$

$$IR^{(i)}(PL_q) = P_{\mathcal{H}}^{(i)} \cdot \left[Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right) + Q\left(\frac{PL_q - T_q^{(i)}}{\widehat{\sigma}_q^{(i)}}\right)\right], \quad i \in I_{NF} \tag{44}$$

where $\mathbf{A}_q^{(i)}$ is a row vector that reflects the effect of the ambiguity offset on the position error. And according to Equation (21), it can be computed by:

$$\mathbf{A}_q^{(i)} = \mathbf{e}_q \cdot \widehat{\mathbf{P}}_{\boldsymbol{ba}}^{(i)} \cdot \left(\widehat{\mathbf{P}}_{\boldsymbol{aa}}^{(i)}\right)^{-1} \tag{45}$$

where $\mathbf{e}_q$ is a 1×3 row vector whose qth entry is 1 and the others are 0. As a reminder, in the subscripts, $\boldsymbol{a}$ denotes the ambiguity vector, and $\boldsymbol{b}$ represents the position vector.

Finally, in the left-hand side of Equation (41), P_{NM} should account for not only the integrity risk from unmonitored fault modes but also that from unmonitored ambiguity candidates. Therefore, P_{NM} can be determined by:

$$P_{NM} = P_{H,NM} + \sum_{i\in I_F} P_{A,NM}^{(i)} \tag{46}$$

The proof of safety associated with the protection level in Equation (41) can be found in Appendix.

4. SIMULATION RESULTS

Using the simplified PPP-RTK model as an example, we carry out multiple sets of simulations in this section to validate the proposed integrity monitoring algorithm for ambiguity resolution enabled GNSS positioning.

4.1 Simulation Set-up

The simulations are conducted around an example scenario where a PPP-RTK user end provides precise positions to enable autonomous navigation for a UAM vehicle flying above the buildings. In the simulation, the air vehicle travels northward at a constant speed of 20 meters per second, and the starting point of the trajectory is at (30.53°N, 114.36°E, 100 meters). The starting epoch is at 08:00:00 on January 1, 2022 (Beijing Time), and the trajectory lasts for 15 minutes with a 1-second sampling interval.

Table 5. List of Constants

Category	Parameters	Symbols	Values
Probabilities	Satellite fault probabilities	P_{sat}	$P_{sat}=10^{-5}$
Performance Requirements	Integrity risk allocations for east, north, and up positions	$P_{HMI,q}$	$P_{HMI,1}=P_{HMI,2}=2\times10^{-9}$; $P_{HMI,3}=9.8\times10^{-8}$
	False alert budgets for east, north, and up positions	$P_{FA,q}$	$P_{FA,1}=P_{FA,2}=9\times10^{-8}$; $P_{FA,3}=3.9\times10^{-6}$
Algorithm Settings	Success rate threshold for the all-in-view solution and fault-tolerant solutions	$P_{S,thr}^{(0)}$; $P_{S,thr}^{(i)}$	$P_{S,thr}^{(0)}=0.9999$; $P_{S,thr}^{(i)}=0.99$
	Threshold for the integrity risk from unmonitored fault modes	P_{THRES}	$P_{THRES}=6\times10^{-8}$
	Threshold for the integrity risk from unmonitored ambiguity candidates	$P_{A,NM,thr}^{(0)}$; $P_{A,NM,thr}^{(i)}$	$P_{A,NM,thr}^{(0)}=1\times10^{-9}$; $P_{A,NM,thr}^{(i)}=1\times10^{-5}$

The positions of the GNSS satellites, including GPS and BDS-3, are derived from the precise orbit file. The receiver raw measurements and the SSR products are simulated based on the error models in Section 2.1. One can refer to Equations (1) ~ (5), **Tables 1~3** for the details. The initial values for the KF are simulated according to **Table 3**, and the error models used in the Kalman filter and the sliding-window least-squares estimator are obtained based on **Tables 1~3**. Finally, the prior fault probabilities for each satellite, the navigation performance requirements, and the algorithm settings are provided in **Table 5**.

4.2 Results and Analyses

Based on the settings above, this subsection will show the simulation results and analyze the performance of the proposed integrity monitoring algorithm. **Fig. 5** compares the position errors between the fixed solution and the float solution, which are output from a Kalman filter. For the fixed solution, the ambiguities are always fixed without checking the ambiguity resolution success rate threshold. As shown in this figure, ambiguity resolution obviously benefits the navigation results.

Fig.6 compares the PLs with and without enabling ambiguity resolution for a KF. For the first 20 epochs in this figure, the all-in-view solution is not fixed because its ambiguity resolution success rate is below the corresponding threshold. Therefore, the two protection level curves are coincident. Then, it can be observed that over the following period, enabling the ambiguity resolution function can significantly improve the navigation integrity performance, i.e., reduce the protection levels. It is worth mentioning that the two protection level curves are coincident again in the segments highlighted by the red ellipses. To reveal the underlying reason behind this phenomenon, **Fig. 7** shows the AR success rates for the all-in-view solution and the subset solutions. As shown in this figure, this phenomenon is caused by the fact that the AR success rates suddenly become too high for the all-in-view solution to be fixed. And it is the addition of a new satellite that leads to the increase in the AR success rates.

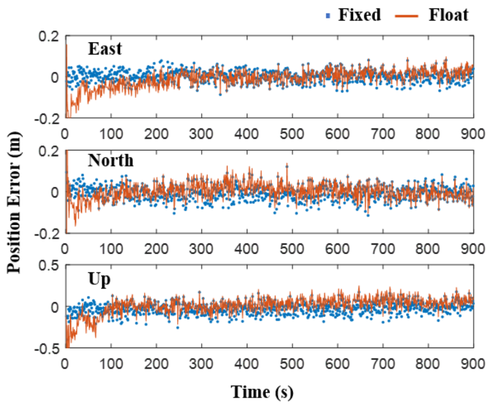

Fig. 5. *Comparison of the position errors between the fixed and float solutions.*

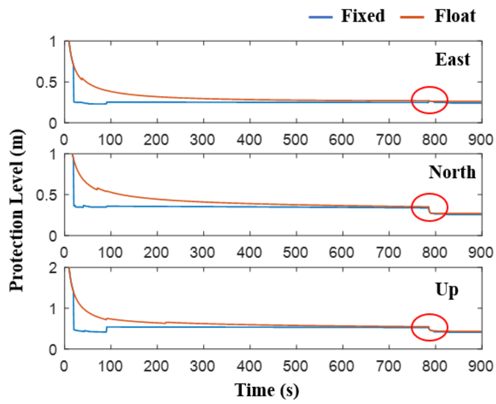

Fig. 6. *Comparison of the protection levels with and without enabling ambiguity resolution*

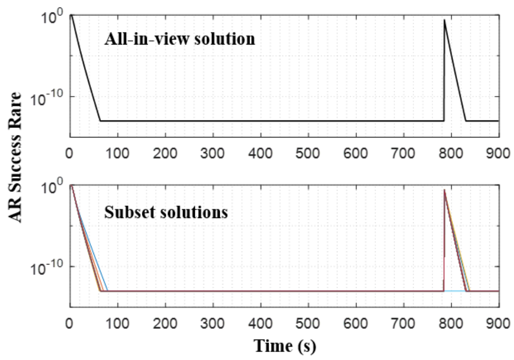

Fig. 7. *The ambiguity resolution success rates for the all-in-view and subset solutions*

To demonstrate the proposed integrity monitoring algorithm, **Fig. 8** shows the position errors and the corresponding protection levels in one figure. The results prove that the protection levels can safety bound the position errors. **Fig. 9** shows the test statistics and the thresholds for subset 1, and it suggests that there is no false alert in a fault-free condition. Besides, **Fig. 10** and **Fig. 11** present the results in a faulted condition, where an undetected cycle slip of 2 cycles is added to satellite G05 from 70s to 90s. It can be seen from the figures that this fault leads to large position errors, and the fault detector can detect the fault in time. Besides, the protection levels can always safely bound the position errors when there is no fault alarm. Please note that in **Fig. 10** and **Fig. 11**, the red squares indicate the fault detection point, and the curves in the shadow area are meaningless because the system will quit once a fault is detected.

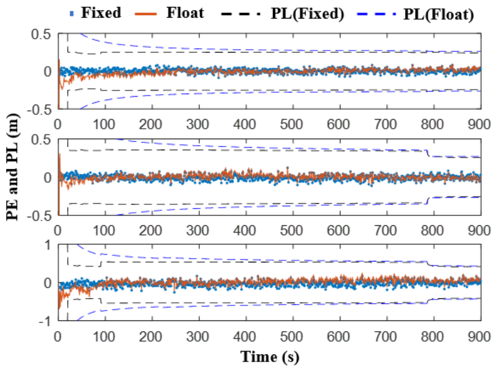

Fig. 8. *Protection levels and position errors in a fault-free condition*

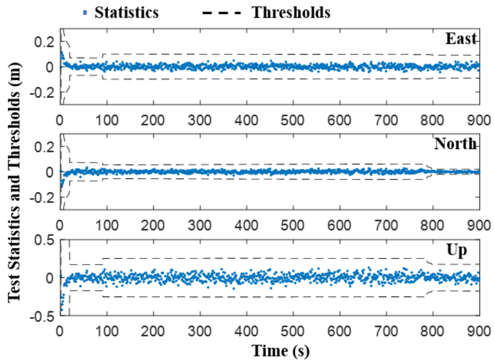

Fig. 9. *Test statistics and thresholds for subset 1 in a fault-free condition*

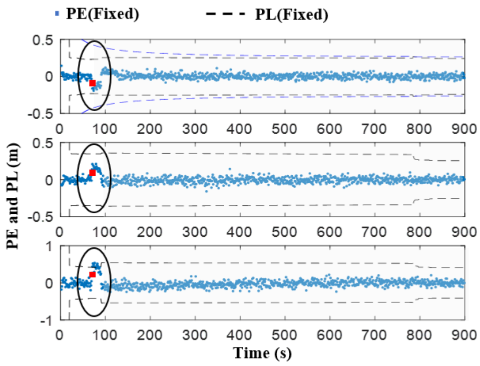

Fig. 10. *Protection levels and position errors in the presence of an undetected cycle slip*

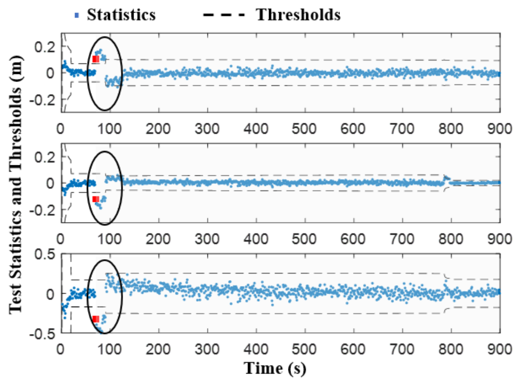

Fig. 11. *Test statistics and thresholds for subset 1 in the presence of an undetected cycle slip*

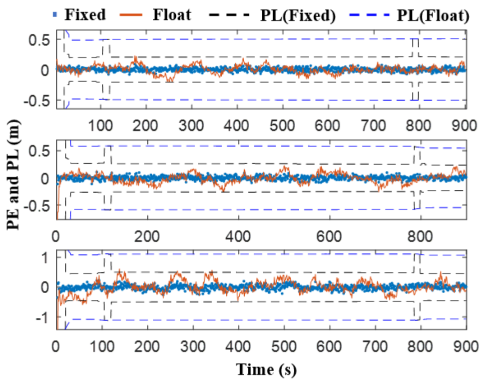

Fig. 12. *Protection levels and position errors of a sliding-window least-squares estimator*

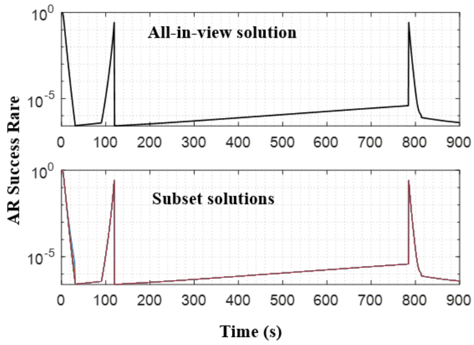

Fig. 13. *Ambiguity resolution success rates of a sliding-window least-squares estimator*

Finally, the position errors and PLs associated with the sliding-window least-squares estimator are shown in **Fig. 12**, and the AR success rates are presented in **Fig.13**. The length of the sliding window is set to 30 epochs. The results suggest that the SWLS estimator produces similar navigation performance to that of the KF. It is noteworthy that in the simulations above, we do not consider the effect of exposure time on the prior fault probability. In our future work, we will make a fairer comparison between the integrity performance of the KF and the SWLS estimator by taking the effect of exposure time into account.

5. CONCLUSIONS

We develop an integrity monitoring algorithm for ambiguity resolution enabled GNSS high-precision positioning. This algorithm is established with a simplified PPP-RTK model and supports both Kalman filters and sliding-window least-squares estimators. The proposed algorithm can accommodate faulted observations (i.e., raw measurements and correction products) and incorrect ambiguity resolution simultaneously, through a modified multiple hypothesis solution separation (MHSS) architecture. The MHSS-based fault detection mechanism is described, and the formula to evaluate the protection levels is derived accordingly, which rigorously accounts for the risk from incorrect-fix events under each of the monitored hypotheses. Simulation results prove that the proposed integrity monitoring algorithm can provide a tight and safe error bound, and it can detect the faults in an effective manner. Also, the results suggest that enabling ambiguity resolution can significantly benefit the navigation performance in terms of both accuracy and integrity, even though incorrect-fix events are considered.

The limitations of the proposed algorithm mainly include: (a) it only supports the integer bootstrapping ambiguity resolution method; (b) the computational load is heavy; (c) fault exclusion is not implemented; (d) spatial and temporal error correlations are not considered. In our future work, we will address these limitations and extend the applicability of this algorithm.

APPENDIX. PROTECTION LEVEL PROOF OF SAFETY

This appendix proves the safety of the PLs given by Equations (41) ~ (46) in the case where the all-in-view solution is fixed.

(1) The integrity risk from the fault-free fault mode

For the fault-free fault mode (i=0), its contribution to the integrity risk is given by:

$$\begin{aligned} IR^{(0)}(PL_q) &= P_{\mathcal{H}}^{(0)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}} \,\middle|\, \mathcal{H}_0\right) < P_{\mathcal{H}}^{(0)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q \mid \mathcal{H}_0\right) \\ &= P_{\mathcal{H}}^{(0)} \cdot \sum_{j=0}^{N_A^{(0)}} P_{A,j}^{(0)} \cdot P\left(|\breve{\varepsilon}_q^{(0)} + \mathbf{A}_q^{(0)} \Delta_j^{(0)}| > PL_q\right) \\ &= P_{\mathcal{H}}^{(0)} \cdot \sum_{j=0}^{N_A^{(0)}} P_{A,j}^{(0)} \cdot \left[P\left(\breve{\varepsilon}_q^{(0)} > PL_q - \mathbf{A}_q^{(0)} \Delta_j^{(0)}\right) + P\left(\breve{\varepsilon}_q^{(0)} > PL_q + \mathbf{A}_q^{(0)} \Delta_j^{(0)}\right)\right] \end{aligned} \tag{47}$$

where $\breve{\varepsilon}_q^{(0)}$ is the noise component of $(\breve{x}_q^{(0)} - x_q)$, which follows $\mathcal{N}(0, \breve{\sigma}_q^{(0)2})$. Given that there are $(N_A^{(0)} + 1)$ ambiguity offsets being monitored, including 1 zero offset (i.e., $\Delta_0^{(0)}$) and $n_a^{(0)}$ positive-negative pairs (i.e., $\Delta_j^{(0)}$ and $\Delta_{j'}^{(0)}$, where $\Delta_{j'}^{(0)} = -\Delta_j^{(0)}$), the equation above can further become:

$$\begin{aligned} IR^{(0)}(PL_q) &< P_{\mathcal{H}}^{(0)} \cdot \left[\sum_{j=0}^{N_A^{(0)}} P_{A,j}^{(0)} \cdot P\left(\breve{\varepsilon}_q^{(0)} > PL_q - \mathbf{A}_q^{(0)} \Delta_j^{(0)}\right) + \sum_{j'=0}^{N_A^{(0)}} P_{A,j'}^{(0)} \cdot P\left(\breve{\varepsilon}_q^{(0)} > PL_q - \mathbf{A}_q^{(0)} \Delta_{j'}^{(0)}\right)\right] \\ &= 2P_{\mathcal{H}}^{(0)} \cdot \sum_{j=0}^{N_A^{(0)}} P_{A,j}^{(0)} \cdot P\left(\breve{\varepsilon}_q^{(0)} > PL_q - \mathbf{A}_q^{(0)} \Delta_j^{(0)}\right) = 2P_{\mathcal{H}}^{(0)} \cdot \sum_{j=0}^{N_A^{(0)}} P_{A,j}^{(0)} \cdot Q\left(\frac{PL_q - \mathbf{A}_q^{(0)} \Delta_j^{(0)}}{\breve{\sigma}_q^{(0)}}\right) \end{aligned} \tag{48}$$

This completes the proof for Equation (42).

(2) The integrity risk from a fault mode i with a fixed fault-tolerant solution ($i > 0$ and $i \in I_F$)

Then, for a fault mode i that outputs a fixed solution, i.e., $i > 0$ and $i \in I_F$, its contribution to the integrity risk can be computed by:

$$IR^{(i)}(PL_q) = P_{\mathcal{H}}^{(i)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, \bar{\mathcal{D}} \,\middle|\, \mathcal{H}_i\right) < P_{\mathcal{H}}^{(i)} \cdot P\left(|\breve{x}_q^{(0)} - x_q| > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathcal{H}_i\right) \tag{49}$$

Under fault mode i, we have the following:

$$\breve{x}_q^{(0)} - x_q = \breve{\varepsilon}_q^{(0)} + \mathbf{A}_q^{(0)} \varDelta^{(0,i)} + b_{fault}^{(i)} \tag{50}$$

where $\varDelta^{(0,i)}$ denotes the ambiguity offset for the all-in-view solution under $\mathcal{H}_i$, and $b_{fault}^{(i)}$ is the effect of the faults on position error. For the convenience of the following derivations, we define:

$$\zeta^{(i)} \triangleq \mathbf{A}_q^{(0)} \varDelta^{(0,i)} + b_{fault}^{(i)} \tag{51}$$

Because $b_{fault}^{(i)}$ follows an arbitrary distribution, the probability density function (PDF) for $\zeta^{(i)}$ is also arbitrary. Substituting (50) and (51) into (49) yields:

$$\begin{aligned} IR^{(i)}(PL_q) &< P_{\mathcal{H}}^{(i)} \cdot P\left(|\breve{\varepsilon}_q^{(0)} + \zeta^{(i)}| > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathcal{H}_i\right) \\ &= P_{\mathcal{H}}^{(i)} \cdot \left[P\left(|\breve{\varepsilon}_q^{(0)} + \zeta^{(i)}| > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid (\zeta^{(i)} \geqslant 0 \mid \mathcal{H}_i)\right) \cdot P(\zeta^{(i)} \geqslant 0 \mid \mathcal{H}_i)\right] \\ &+ P_{\mathcal{H}}^{(i)} \cdot \left[P\left(|\breve{\varepsilon}_q^{(0)} + \zeta^{(i)}| > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid (\zeta^{(i)} < 0 \mid \mathcal{H}_i)\right) \cdot P(\zeta^{(i)} < 0 \mid \mathcal{H}_i)\right] \end{aligned} \tag{52}$$

with $P(\zeta^{(i)} \geqslant 0 \mid \mathcal{H}_i) + P(\zeta^{(i)} < 0 \mid \mathcal{H}_i) = 1$.

Let us start from the case $\zeta^{(i)} \geqslant 0$:

$$\begin{aligned} P\left(|\breve{\varepsilon}_q^{(0)} + \zeta^{(i)}| > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) &= P\left(\breve{\varepsilon}_q^{(0)} + \zeta^{(i)} > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) \\ &+ P\left(\breve{\varepsilon}_q^{(0)} + \zeta^{(i)} < -PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) \end{aligned} \tag{53}$$

where $\mathbb{E}_+^{(i)}$ represents the event $(\zeta^{(i)} \geqslant 0 \mid \mathcal{H}_i)$. For first term of the right-hand side, we have:

$$\begin{aligned} & P\left(\breve{\varepsilon}_q^{(0)} + \zeta^{(i)} > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) = P\left(\breve{x}_q^{(0)} - x_q > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) \\ & < P\left(\breve{x}_q^{(0)} - \breve{x}_q^{(i)} + \breve{x}_q^{(i)} - x_q > PL_q, \breve{x}_q^{(0)} - \breve{x}_q^{(i)} < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) \\ & < P\left(\breve{x}_q^{(i)} - x_q > PL_q - T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) \\ & = P\left(\breve{\varepsilon}_q^{(i)} + \mathbf{A}_q^{(i)} \varDelta^{(i)} > PL_q - T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) = \sum_{j=0}^{N_A^{(i)}} P_{A,j}^{(i)} \cdot Q\left(\frac{PL_q - T_q^{(i)} - \mathbf{A}_q^{(i)} \varDelta_j^{(i)}}{\breve{\sigma}_q^{(i)}}\right) \end{aligned} \tag{54}$$

And for the second term, we have:

$$\begin{aligned} & P\left(\breve{\varepsilon}_q^{(0)} + \zeta^{(i)} < -PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) = P\left(\breve{x}_q^{(0)} - x_q < -PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) \\ & < P\left(\breve{x}_q^{(0)} - x_q < -PL_q \mid \mathbb{E}_+^{(i)}\right) = P\left(\breve{\varepsilon}_q^{(0)} + \zeta^{(i)} < -PL_q \mid \mathbb{E}_+^{(i)}\right) \\ & < P\left(\breve{\varepsilon}_q^{(0)} < -PL_q \mid \mathbb{E}_+^{(i)}\right) = Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right) \end{aligned} \tag{55}$$

Substituting (55) and (54) into (53) yields:

$$P\left(|\breve{\varepsilon}_q^{(0)} + \zeta^{(i)}| > PL_q, |\breve{x}_q^{(0)} - \breve{x}_q^{(i)}| < T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right) < \sum_{j=0}^{N_A^{(i)}} P_{A,j}^{(i)} \cdot Q\left(\frac{PL_q - T_q^{(i)} - \mathbf{A}_q^{(i)} \varDelta_j^{(i)}}{\breve{\sigma}_q^{(i)}}\right) + Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right) \tag{56}$$

Similarly, for the case $\zeta^{(i)} < 0$, we have:

$$P\left(|\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}|>PL_q,|\breve{x}_q^{(0)}-\breve{x}_q^{(i)}|<T_q^{(i)}\mid \mathbb{E}_-^{(i)}\right)<\sum_{j=0}^{N_A^{(i)}}P_{A,j}^{(i)}\cdot Q\left(\frac{PL_q-T_q^{(i)}+\mathbf{A}_q^{(i)}\Delta_j^{(i)}}{\breve{\sigma}_q^{(i)}}\right)+Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right) \tag{57}$$

Given that the monitored ($N_A^{(i)}+1$) ambiguity offsets are composed of 1 zero offset (i.e., $\Delta_0^{(i)}$) and $n_a^{(i)}$ positive-negative pairs (i.e., $\Delta_j^{(i)}$ and $\Delta_{j'}^{(i)}$, where $\Delta_{j'}^{(i)}=-\Delta_j^{(i)}$), we can obtain the following:

$$\sum_{j=0}^{N_A^{(i)}}P_{A,j}^{(i)}\cdot Q\left(\frac{PL_q-T_q^{(i)}-\mathbf{A}_q^{(i)}\Delta_j^{(i)}}{\breve{\sigma}_q^{(i)}}\right)=\sum_{j=0}^{N_A^{(i)}}P_{A,j}^{(i)}\cdot Q\left(\frac{PL_q-T_q^{(i)}+\mathbf{A}_q^{(i)}\Delta_j^{(i)}}{\breve{\sigma}_q^{(i)}}\right) \tag{58}$$

And thus, Equation (52) becomes:

$$IR^{(i)}(PL_q)<P_{\mathcal{H}}^{(i)}\cdot\left[\sum_{j=0}^{N_A^{(i)}}P_{A,j}^{(i)}\cdot Q\left(\frac{PL_q-T_q^{(i)}-\mathbf{A}_q^{(i)}\Delta_j^{(i)}}{\breve{\sigma}_q^{(i)}}\right)+Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right)\right] \tag{59}$$

This completes the proof for Equation (43).

(3) The integrity risk from a fault mode i with a float fault-tolerant solution ($i>0$ and $i\in I_{NF}$)

Finally, for a fault mode i that outputs a float solution, i.e., $i>0$ and $i\in I_{NF}$, its contribution to the integrity risk can be computed by:

$$\begin{aligned}IR^{(i)}(PL_q)&=P_{\mathcal{H}}^{(i)}\cdot P\left(|\breve{x}_q^{(0)}-x_q|>PL_q,\bar{\mathcal{D}}\,\middle|\,\mathcal{H}_i\right)\\&<P_{\mathcal{H}}^{(i)}\cdot P\left(|\breve{x}_q^{(0)}-x_q|>PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathcal{H}_i\right)\end{aligned} \tag{60}$$

Similar to Equation (52), we have:

$$\begin{aligned}IR^{(i)}(PL_q)&<P_{\mathcal{H}}^{(i)}\cdot P\left(|\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}|>PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathcal{H}_i\right)\\&=P_{\mathcal{H}}^{(i)}\cdot\left[P\left(|\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}|>PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid(\zeta^{(i)}\geqslant 0\mid\mathcal{H}_i)\right)\cdot P(\zeta^{(i)}\geqslant 0\mid\mathcal{H}_i)\right]\\&+P_{\mathcal{H}}^{(i)}\cdot\left[P\left(|\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}|>PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid(\zeta^{(i)}<0\mid\mathcal{H}_i)\right)\cdot P(\zeta^{(i)}<0\mid\mathcal{H}_i)\right]\end{aligned} \tag{61}$$

with $P(\zeta^{(i)}\geqslant 0\mid\mathcal{H}_i)+P(\zeta^{(i)}<0\mid\mathcal{H}_i)=1$.

For the case $\zeta^{(i)}\geqslant 0$, we have:

$$\begin{aligned}P\left(|\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}|>PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)&=P\left(\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}>PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)\\&+P\left(\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}<-PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)\end{aligned} \tag{62}$$

where $\mathbb{E}_+^{(i)}$ represents the event $(\zeta^{(i)}\geqslant 0\mid\mathcal{H}_i)$. For first term of the right-hand side, we can obtain the following:

$$\begin{aligned}P\left(\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}>PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)&=P\left(\breve{x}_q^{(0)}-x_q>PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)\\&<P\left(\breve{x}_q^{(0)}-\hat{x}_q^{(i)}+\hat{x}_q^{(i)}-x_q>PL_q,\breve{x}_q^{(0)}-\hat{x}_q^{(i)}<T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)\\&<P\left(\hat{x}_q^{(i)}-x_q>PL_q-T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)\\&=P\left(\hat{\varepsilon}_q^{(i)}>PL_q-T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)=Q\left(\frac{PL_q-T_q^{(i)}}{\hat{\sigma}_q^{(i)}}\right)\end{aligned} \tag{63}$$

For the second term, we can get:

$$\begin{aligned}P\left(\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}<-PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)&=P\left(\breve{x}_q^{(0)}-x_q<-PL_q,|\breve{x}_q^{(0)}-\hat{x}_q^{(i)}|<T_q^{(i)}\mid\mathbb{E}_+^{(i)}\right)\\&<P\left(\breve{x}_q^{(0)}-x_q<-PL_q\mid\mathbb{E}_+^{(i)}\right)=P\left(\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}<-PL_q\mid\mathbb{E}_+^{(i)}\right)\\&<P\left(\breve{\varepsilon}_q^{(0)}<-PL_q\mid\mathbb{E}_+^{(i)}\right)=Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right)\end{aligned} \tag{64}$$

Substituting (63) and (64) into (62) yields:

$$P\left(|\breve{\varepsilon}_q^{(0)}+\zeta^{(i)}|>PL_q,|\breve{x}_q^{(0)}-\breve{x}_q^{(i)}|<T_q^{(i)} \mid \mathbb{E}_+^{(i)}\right)<Q\left(\frac{PL_q-T_q^{(i)}}{\hat{\sigma}_q^{(i)}}\right)+Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right) \tag{65}$$

Similarly, for the case $\zeta^{(i)}<0$, we can obtain the same result as above. Therefore, Equation (60) finally becomes

$$IR^{(i)}(PL_q)=P_{\mathcal{H}}^{(i)}\cdot\left[Q\left(\frac{PL_q-T_q^{(i)}}{\hat{\sigma}_q^{(i)}}\right)+Q\left(\frac{PL_q}{\breve{\sigma}_q^{(0)}}\right)\right] \tag{66}$$

This completes the proof for Equation (44).

REFERENCES:

[1] T. Reid et al. Localization Requirements for Autonomous Vehicles. *International Journal of Connected and Automated Vehicles* 2 (3), 173-190, 2019.

[2] P.J.G. Teunissen and S. Verhagen. GNSS Ambiguity Resolution: When and How to Fix or not to Fix? *Part of the International Association of Geodesy Symposia book series* (volume 132), 2008.

[3] K. Gunning, et al. Design and Evaluation of Integrity Algorithms for PPP in Kinematic Applications. *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation* (ION GNSS+ 2018), Miami, Florida, September 2018.

[4] K. Gunning, et al. Integrity for Tightly Coupled PPP and IMU. *Proceedings of the 32nd International Technical* Meeting *of The Satellite Division of the Institute of Navigation* (ION GNSS+ 2019), Miami, Florida, September 2019.

[5] J. Blanch et al. Reducing Computational Load in Solution Separation for Kalman Filters and an Application to PPP Integrity. *Proceedings of the 2019 International Technical Meeting of The Institute of Navigation*, Reston, Virginia, January 2019.

[6] S. Khanafseh and B. Pervan. New Approach for Calculating Position Domain Integrity Risk for Cycle Resolution in Carrier Phase Navigation Systems. *IEEE Transactions on Aerospace and Electronic Systems*, 2010, 46(1): 296-307.

[7] G. Green and T. Humphreys. Position-Domain Integrity Analysis for Generalized Integer Aperture Bootstrapping. *IEEE Transactions on Aerospace and Electronic Systems*, 2019, 55(2): 734-746.

[8] P.J.G. Teunissen. Success Probability of Integer GPS Ambiguity Rounding and Bootstrapping. *Journal of Geodesy*, 1998, 72(10): 606-612.

[9] J. Blanch, E. Phelts, K. Gunning, T. Walter, L. Groot, L. Norman. Bounding Temporally Correlated Measurement Noise with an Application to GNSS Integrity. *Proceedings of the 33rd International Technical Meeting of the Satellite Division of The Institute of Navigation* (ION GNSS+ 2020), September 2020, pp. 27-38.

[10] E. Gallon, M. Joerger, B. Pervan. Frequency-Domain Modeling of Orbit and Clock Errors for Sequential Positioning. *Proceedings of the 33rd International Technical Meeting of the Satellite Division of The Institute of Navigation* (ION GNSS+ 2020), September 2020, pp. 1041-1053.

[11] X.W. Chang, X. Yang, T. Zhou. MLAMBDA: A Modified LAMBDA Method for Integer Least-Squares Estimation. *Journal of Geodesy* 2005, 79, 552–565.

[12] J. Blanch et al., Baseline Advanced RAIM User Algorithm and Possible Improvements. *IEEE Transactions on Aerospace and Electronic Systems*, 51(1): 713-732.