

PPP-RTK에서 효율적인 미지정수 결정을 위한 재귀형 배치 측위 알고리즘

정준우¹, 박준민¹, 박재영², 고요한³, 박경민³, 박다나³, 박찬식^{4†}

Recursive Batch Positioning Algorithm for Efficient Integer Ambiguity Resolution in PPP-RTK

Junwoo Jeong¹, JunMin Park¹, Jae-Young Park², Yo Han Ko³, Gyeongmin Park³, Dana Park³, Chansik Park^{4†}

¹Department of Electronics Engineering, Chungnam National University, Daejeon 34134, South Korea

²Korea Aerospace Research Institute, Daejeon 34133, South Korea

³Satellite Systems Laboratory, LIG Nex1, Seongnam 13449, South Korea

⁴Department of Intelligent System and Robotics, Chungbuk National University, Chungcheongbuk-do 28644, South Korea

ABSTRACT

This paper proposes a recursive batch positioning algorithm to provide efficient integer ambiguity resolution and precise positioning for real-time Precise Point Positioning-Real Time Kinematic (PPP-RTK) receivers. The performance of the algorithm is validated using real measurement data. Because PPP-RTK measurements generally contain larger unmodeled errors than RTK measurements, an epoch-by-epoch approach often yields degraded integer ambiguity resolution performance. To address this issue, a batch method improves integer ambiguity resolution by using all measurements accumulated up to the current epoch. For real-time receiver implementation, the batch method is reformulated within a recursive framework. Experimental results using dual-frequency Global Positioning System (GPS) and Galileo measurements under a Single-Difference (SD) model show that ambiguities were resolved from the first epoch. Even with single-frequency data, ambiguities were successfully resolved within two minutes.

Keywords: PPP-RTK, SSR, integer ambiguity, UD/SD model, recursive batch

주요어: PPP-RTK, SSR, 미지정수, UD/SD 모델, 재귀형 배치 알고리즘

1. INTRODUCTION

최근 Global Navigation Satellite System (GNSS) 기반 측위는 측지/측량과 같은 전통적 응용을 넘어 자율주행, UAM/드론, 정밀 물류 등으로 확대되며 cm급 고정밀 항법 기술의 중요성이 커지고 있다. 이러한 요구에 대응하기 위해 전리층/대류권 지연과 위성 궤도/시계 오차 등을 State Space Representation (SSR) 형태로 제공하고, 사용자가 이를 이용해 정밀 위치를 산출하는 Precise Point Positioning-Real Time Kinematic (PPP-RTK)가 대표적인 대안으로 부상하였다. PPP-RTK는 기존 PPP의 전역성

(광역 적용 가능)과 RTK의 빠른 수렴/고정해(ambiguity fixing) 특성을 결합한 방식으로, SSR 보정치 기반 고정밀 서비스의 핵심 기술로 자리잡고 있다(Wabben et al. 2005, Teunissen & Montenbruck 2017, Lim et al. 2018, An et al. 2023).

현재 PPP-RTK 서비스로 QZSS, POINT, NGII 등에서 SSR 보정치가 제공되고 있다 (Park et al. 2021, Song et al. 2024). 이에 따라 GNSS 수신기 관점에서 SSR 보정치의 실시간 처리, 항법필터 설계, 미지정수(ambiguity) 결정 및 정수해 산출까지 포함하는 PPP-RTK 알고리즘 구현이 중요하다. 하나의 거리 오차로 표현하는 RTK과는 달리 PPP-RTK에서는 오차를 요인별로 분류(위성

Received Feb 13, 2026 Revised Mar 01, 2026 Accepted Mar 15, 2026

[†]Corresponding Author E-mail: chansp@cbnu.ac.kr



Creative Commons Attribution Non-Commercial License (<https://creativecommons.org/licenses/by-nc/4.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

시계, 궤도, 코드/위상 바이어스, 전리층, 대류권)하고 각각의 오차 보정치를 다른 주기로 전송하여 항법 성능은 유지하면서 통신의 효율성을 높인다 (Hirokawa et al. 2021). PPP-RTK에서 미지정수 결정 성공률은 측정치의 개수와 품질뿐 아니라 수신기에서 처리하는 항법 알고리즘에 영향을 받는다 (Geng et al. 2010). 또한 사용할 수 있는 위성군과 주파수에 따라 미지정수 성공률도 차이가 난다.

기존 PPP-RTK 관련 연구들은 일반적으로 SSR 보정치를 적용하여 위성 궤도·시계, 바이어스, 이온층 지연, 대류권 지연 오차를 저감하고, 안테나 위상중심(PCO/PCV), 지구 조석 등 관측모델 보정을 수행한 뒤, 공개 소프트웨어인 CLASLIB 또는 SSR2OSR과 상용 소프트웨어인 RTKLIB을 이용하여 구한 위치결정 성능을 평가한다 (Ong et al. 2021, Lim et al. 2022a, 2022b, Park et al. 2024, Song et al. 2024, Jeong et al. 2025, Park 2025). 이러한 접근은 빠른 시간에 처리가 용이하다는 장점이 있으나, 사용자단에서 수행되는 핵심 처리(보정치 처리, 상태추정 필터의 구성, 미지정수 결정 등) 알고리즘에 따른 성능 차이를 정확히 확인하기 어렵다. 실시간 수신기 구현에 필요한 모듈화 및 연산 구조 최적화가 어렵고, 수신기 구조에 이식 및 검증하는 데 제약이 발생한다.

이 논문에서는 SSR과 관측모델 보정이 된 코드와 반송파 위상 측정치를 이용하여 효과적으로 미지정수를 구하고, 구해진 미지정수를 이용하여 위치해를 구하는 recursive batch 항법 알고리즘을 제안한다. 제안한 방법은 먼저 매 epoch 측정치를 별도로 처리하는 epoch-by-epoch 기법과는 달리 미지정수는 시간 따라 변하지 않는 상수임을 고려하여 현재 epoch까지의 측정치를 모두 사용하는 batch 항법 알고리즘을 유도하였다. Batch 항법 알고리즘을 다시 recursive batch 항법 알고리즘으로 변경하여 실시간으로 구현할 수 있도록 하였다. 이 결과는 실시간 GNSS 수신기의 구현에 적용될 수 있을 것으로 기대된다.

이 논문의 구성은 다음과 같다. 2장에서는 UD와 SD 측정치 모델을 유도하고, 3장에서 이 모델을 이용한 epoch-by-epoch 항법 알고리즘을 구현하고 실제 측정치를 이용한 처리 결과로 epoch-by-epoch 기법의 한계를 확인한다. 4장에서 batch 항법 알고리즘과 이를 변형한 recursive batch 항법 알고리즘을 유도하고, 실제 측정치를 이용하여 단일주파수와 이중주파수 측정치의 성능을 보인다. 추가로 공개 소프트웨어 PPP-RTK 플랫폼인 CLASLIB을 사용한 결과와 비교한다. 마지막 5장은 결론이다.

2. PPP-RTK MEASUREMENT MODEL

2.1 Un-Difference (UD) Model

k epoch에서 위성 s 의 주파수 j 에 대한 수신기 r 에서 코드 $C_{r,j}^s(k)$ 와 반송파 위상 $L_{r,j}^s(k)$ 측정치는 Eqs. (1, 2)와 같다. 여기서 c 는 광속, λ_j^s 는 파장, $\mu_j^s = (\lambda_j^s/\lambda_1^s)^2$ 을 나타낸다. 나머지 오차의 종류와 SSR 보정치를 사용하는 PPP-RTK에서 처리 방법은 Table 1과 같다.

$$C_{r,j}^s(k) = \rho_r^s(k) + T_r^s(k) + c[B_r(k) - b^s(k)] + \mu_j^s I_r^s(k) - \Delta b_{E,j}^s(k) + \Delta A_{r,j}^s(k) + \Delta d_r^s \quad (1)$$

$$L_{r,j}^s(k) = \rho_r^s(k) + T_r^s(k) + c[B_r(k) - b^s(k)] - \mu_j^s I_r^s(k) - \Delta b_{E,j}^s(k) + \Delta A_{r,j}^s(k) + \lambda_j^s [a_{r,j}^s + \Delta p_r^s(k)] + \Delta d_r^s + \varepsilon_{r,j}^s(k) \quad (2)$$

여기서 $e_{r,j}^s(k)$ 와 $\varepsilon_{r,j}^s(k)$ 는 각각 코드와 반송파 위상 측정 잡음으로 평균이 0, 공분산이 σ_i^2 인 백색 가우시안 잡음(AWGN)으로 가정한다. IS-QZSS-L6-007에 따라 SSR 보정치에 포함된 quality indicator로 구한 signal in space, iono, tropo 공분산 $\sigma_{i, \text{sis}}^2$, $\sigma_{i, \text{iono}}^2$, $\sigma_{i, \text{tropo}}^2$ 과 사용자 수신기의 공분산 $\sigma_{i, \text{user}}^2$ 을 Eq. (3)에 대입하여 측정치 공분산을 구한다 (Cabinet Office 2025). 여기서 f , E_i 는 주파수와 앙각(elevation angle)을 나타낸다.

$$\sigma_i = \sqrt{(\sigma_{i, \text{user}})^2 + (\sigma_{i, \text{sis}}/10)^2 + \left(\frac{40.3 \times 10^{16}}{f^2} \sigma_{i, \text{iono}} \times 100\right)^2 + ((\sigma_{i, \text{tropo}}/10)/\sin E_i)^2} \quad (3)$$

Standard point positioning 등으로 구한 대략적인 사용자 위치 $r=[x_r(k) \ y_r(k) \ z_r(k)]^T$ 를 이용하여 코드와 반송파 위상 측정치는 Eqs. (4, 5)와 같이 선형화할 수 있다.

$$y_{L1}^s(k) = g_{L1}^s(k)\delta u(k) + \lambda_{L1}^s a_{L1}^s + \varepsilon_{L1}^s(k) \sim N(0, \sigma_{L1}^2) \quad (4)$$

$$y_{C1}^s(k) = g_{C1}^s(k)\delta u(k) + \varepsilon_{C1}^s(k) \sim N(0, \sigma_{C1}^2) \quad (5)$$

여기서 $y_{L1}^s(k)=L_{L1}^s(k)-\|s-r\|$, $y_{C1}^s(k)=L_{C1}^s(k)-\|s-r\|$ 는 측정치와, 위성 $s=[x^s(k) \ y^s(k) \ z^s(k)]^T$ 와 사용자 사이의 계산된 거리 $\|s-r\|$ 의 차이이다. $[\delta u(k) \ \delta x(k) \ cB(k)]$ 는 위치오차 $\delta x(k)$ 와 시계오차 $cB(k)$ 를 포함하는 사용자 오차 벡터, $g_{L1}^s(k)=g_{C1}^s(k)=[h_r^s(k) \ 1]$ 는 위성과 사용자 사이의 line of sight $h_r^s(k)$ 와 1로 구성된 측정 행렬이다.

m 개의 위성 측정치를 모두 모아 정리하면 Eq. (6)의 형태로 나타낼 수 있다. 여기서 위첨자 G 는 GPS를, 아래첨자 $L1$, $C1$ 은 각각 $L1$ 반송파와 $C1$ 코드를 나타낸다. $y_{L1}^i(k)=[y_{L1}^1(k) \ \dots \ y_{L1}^m(k)]^T$, $G_{L1}^i(k)=[g_{L1}^1(k) \ \dots \ g_{L1}^m(k)]^T$, $\Lambda_{L1}^i(k)=\text{diag}(\lambda_{L1}^i)$ 이며 $Q_{y_{L1}^i}^G(k)=\text{diag}(\sigma_{L1}^2)$, $Q_{y_{C1}^i}^G(k)=\text{diag}(\sigma_{C1}^2)$ 는 공분산 행렬이다. 여기서 diag 는 대각항에만 값을 갖는 행렬이다.

$$\begin{bmatrix} y_{L1}^G(k) \\ y_{C1}^G(k) \end{bmatrix} = \begin{bmatrix} G_{L1}^G(k) & \Lambda_{L1}^G(k) \\ G_{C1}^G(k) & 0 \end{bmatrix} \begin{bmatrix} \delta u(k) \\ a_{L1}^G(k) \end{bmatrix} + \begin{bmatrix} \varepsilon_{L1}^G(k) \\ \varepsilon_{C1}^G(k) \end{bmatrix} \sim N\left(0, \begin{bmatrix} Q_{y_{L1}^G}^G(k) & 0 \\ 0 & Q_{y_{C1}^G}^G(k) \end{bmatrix}\right) \quad (6)$$

한 개의 위성에 대한 측정치 Eq. (6)을 Eqs. (7, 8)의 Multi-Constellations/Multi-Frequencies (MCMF)로 확장한다. 여기서 $y_{(1,2,5)}^G(k)=[y_{(1)}^G(k) \ y_{(2)}^G(k) \ y_{(5)}^G(k)]^T$ 로 두면 3중 주파수로 확장할 수 있고, 다시 위성군에 대하여 확장하면 Eq. (7)과 같은 MCMF 형태를 얻을 수 있다. 여기서 위첨자 E 와 C 는 각각 Galileo와 BeiDou를 의미한다.

$$\begin{bmatrix} y_{(1,2,5)}^G(k) \\ y_{(1,2,5)}^E(k) \\ y_{(1,2,5)}^C(k) \end{bmatrix} = \begin{bmatrix} G_{(1,2,5)}^G(k) & \Lambda_{(1,2,5)}^G(k) & 0 & 0 \\ G_{(1,2,5)}^E(k) & 0 & \Lambda_{(1,2,5)}^E(k) & 0 \\ G_{(1,2,5)}^C(k) & 0 & 0 & \Lambda_{(1,2,5)}^C(k) \end{bmatrix} \begin{bmatrix} \delta u(k) \\ a_{(1,2,5)}^G(k) \\ a_{(1,2,5)}^E(k) \\ a_{(1,2,5)}^C(k) \end{bmatrix} + \begin{bmatrix} \varepsilon_{(1,2,5)}^G(k) \\ \varepsilon_{(1,2,5)}^E(k) \\ \varepsilon_{(1,2,5)}^C(k) \end{bmatrix} \quad (7)$$

$$\sim N(0, \text{diag}([Q_{(1,2,5)}^G(k), Q_{(1,2,5)}^E(k), Q_{(1,2,5)}^C(k)]))$$

GPS Galileo Time Offset를 ephemeris에서 제공하는 값으로 사용하면 $\delta u(k)=[\delta x(k) \ cB(k)]^T$, $G^s(k)=[H^s(k) \ 1]$, $1=[1 \ \dots \ 1]^T$ 이지만, 이 논문에서와 같이 ephemeris를 사용하지 않고 별도의

Table 1. Description of symbols.

Symbol	Error	Correction
$\rho_r^s(k)$	Geometric range between satellite and receiver	SSR ephemeris (30 s period)
$T_r^s(k)$	Tropospheric delay	SSR troposphere (30 s period)
$B_r(k)$	Receiver clock error	UD: estimate, SD: canceled by differencing
$b^s(k)$	Satellite clock error	SSR SV clock (5 s period)
$\mu_r I_r^s(k)$	Ionospheric delay (at frequency j)	SSR ionosphere (30 s period)
$\Delta b_{c,j}^s(k)$	Satellite code H/W bias	SSR code bias (30 s period)
$\Delta b_{L,j}^s(k)$	Satellite phase H/W bias	SSR phase bias (30 s period)
$\Delta A_{r,j}^s(k)$	Antenna PCO/PCV correction	Antex file
$\Delta p_r^s(k)$	Phase wind-up	Antex file
$\Delta d_r^s(k)$	Earth tidal	ERP, OTL files

parameter로 두고 구하는 경우 Eq. (8)과 같이 변형된다.

$$\delta u(k) = \begin{bmatrix} \delta x(k) \\ cB^G(k) \\ cB^E(k) \\ cB^C(k) \end{bmatrix}, \begin{cases} G_*^G(k) = [H_*^G(k) & \mathbf{1} & \mathbf{0} & \mathbf{0}] \\ G_*^E(k) = [H_*^E(k) & \mathbf{0} & \mathbf{1} & \mathbf{0}] \\ G_*^C(k) = [H_*^C(k) & \mathbf{0} & \mathbf{0} & \mathbf{1}] \end{cases} \quad (8)$$

위 모델은 사용하는 위성군과 주파수의 수에 따라 달라지지만, 이 논문에서는 편의를 위하여 Eq. (9)와 같은 간략한 표현을 사용한다.

$$y(k) = G(k)\delta u(k) + \Lambda a + \varepsilon(k) \sim N(\mathbf{0}, G(k)) \quad (9)$$

2.2 Single-Difference (SD) Model

선형화된 UD model에 Eq. (10)의 차분 연산자를 곱하면 SD model을 얻을 수 있다. SD 연산자를 곱하면 처음 위성을 pivot으로 두고 나머지 위성의 측정치와 차분된 측정치를 얻을 수 있으며, 차분되기 전 UD 측정치에 비하여 차수가 1 줄어든다.

$$SD = \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ & & & \ddots \\ -1 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{(m-1) \times m} \quad (10)$$

SD model의 기준 위성 선정에는 양각 또는 C/N0를 고려하는 것이 유리하다. 그러나 본 연구에서는 PRN을 오름차순으로 정렬하고 가장 먼저 나타나는 위성을 기준 위성으로 선정하는 간단한 방법을 이용하였다.

MCMF 측정치에 대한 SD model의 일반적인 형태는 Eq. (11)과 같다. 여기서 수신기 간 공통 오차인 수신기 시계 오차가 상쇄되어 측정식에 나타나지 않는다.

$$SD \cdot y(k) = SD \cdot G(k)\delta u(k) + \Lambda a_{SD} + \varepsilon_{SD}(k) \sim N(\mathbf{0}, Q_{SD}(k)) \quad (11)$$

3. EPOCH-BY-EPOCH POSITIONING

3.1 Kinematic Positioning

Kinematic positioning 기법은 매 epoch 측정치를 독립적으로 처리하는 방법으로, Eq. (8)의 선형화된 MCMF 측정치에 대하여 Weighted Least Squares (WLS)를 적용하면 Eqs. (12, 13)과 같이

실수해(float solution) $\delta u(k)$ 와 공분산 $\text{cov}(\delta u(k))$ 를 구할 수 있다.

$$\delta \hat{u}(k) = (G(k)^T Q(k)^{-1} G(k))^{-1} G(k)^T Q(k)^{-1} y(k) \quad (12)$$

$$\text{cov}(\delta \hat{u}(k)) = (G(k)^T Q(k)^{-1} G(k))^{-1} \quad (13)$$

위의 과정을 통해 계산된 반송파의 실수해와 공분산을 Least AMBIGUITY DATA ADJUSTMENT (LAMBDA) 방법에 적용하여 매 epoch마다 미지정수를 추정한다. 구한 미지정수 z^* 를 뺀 다음 측정치를 이용하여 Eq. (14)와 같이 최종 정수해(fixed solution)를 구할 수 있다.

$$y(k) - \Lambda z^* = G(k)\delta u(k) + \varepsilon(k) \quad (14)$$

같은 방법으로 SD 측정치도 Eqs. (15, 16)과 같은 실수해와 공분산을 얻고, Eq. (17)과 같은 최종 정수해를 얻을 수 있다.

$$\delta \hat{u}(k) = \left((SD \cdot G(k))^T Q_{SD}(k)^{-1} (SD \cdot G(k)) \right)^{-1} (SD \cdot G(k))^T Q_{SD}(k)^{-1} SD \cdot y(k) \quad (15)$$

$$\text{cov}(\delta \hat{x}(k)) = \left((SD \cdot G(k))^T Q_{SD}(k)^{-1} (SD \cdot G(k)) \right)^{-1} \quad (16)$$

$$SD \cdot y(k) - \Lambda z_{SD}^* = SD \cdot G(k)\delta u(k) + \varepsilon_{SD}(k) \quad (17)$$

3.2 Experimental Results

이번 절에서는 일본 Shintotsukawa에 위치한 STK200JPN 관측소의 안정적인 QZSS SSR 보정치를 이용하여 kinematic positioning을 수행한 결과를 제시한다. 2025-05-20 09:00:00 ~ 11:46:39의 RINEX 측정치와 SSR 보정치를 사용했으며 GPS L1/L2, GAL E1/E5a 측정치를 이용하였다. 해당 관측소에는 SINEX와 DCB 파일이 가용하며 이를 적용하였다.

실험에 사용한 1,000 epoch 동안 관측된 위성의 수와 PDOP과 skyplot을 Fig. 1에 나타냈다. Elevation mask angle은 15도로 두고, 위성 수의 변화가 없는 구간을 선정하였다.

Fig. 2에 UD와 SD model에 epoch-by-epoch WLS를 적용하여 얻은 수평오차(2 dRMS)를 나타냈다. Fig. 2에서 붉은색은 실수해를 노란색은 정수해를 나타낸다. 실수해는 SD (0.33613 m)와 UD (0.33634 m)로 큰 차이가 없지만 정수해는 SD (0.19396 m), UD (0.24027 m)로 차이를 보였다. 이는 SD model이 UD model에서 제거하지 못한 공통 오차를 어느 정도 제거해주기 때문으로 분석할 수 있다.

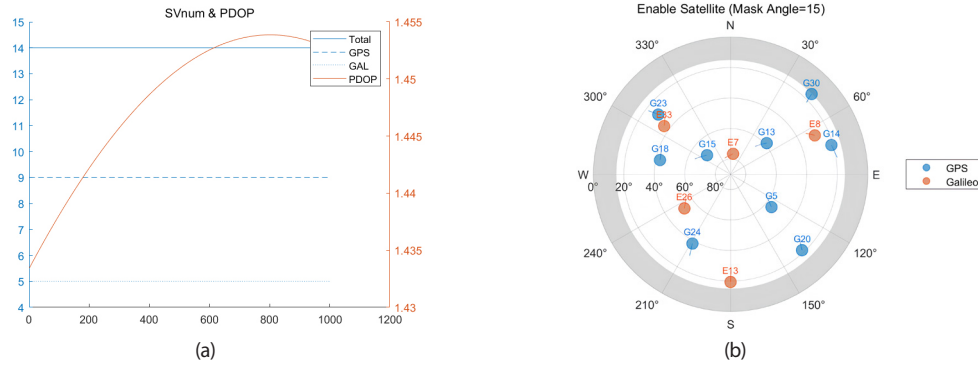


Fig. 1. (a) Number of satellites and (b) skyplot, during the experimental period.

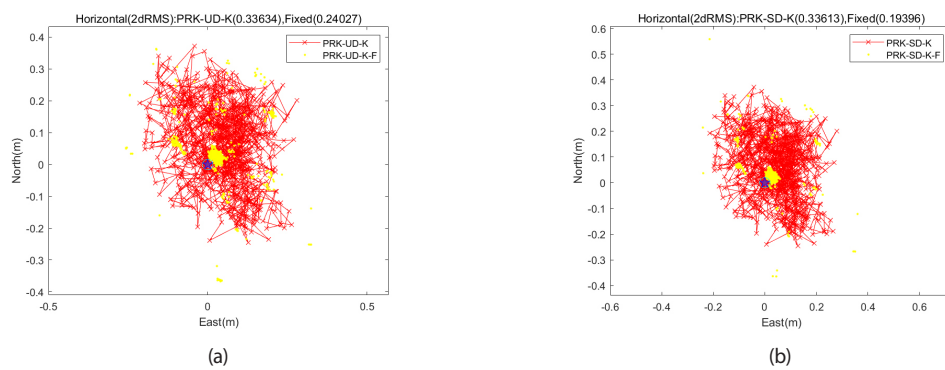


Fig. 2. Epoch-by-epoch WLS-based navigation results (dual-frequencies) (a) UD model and (b) SD model.

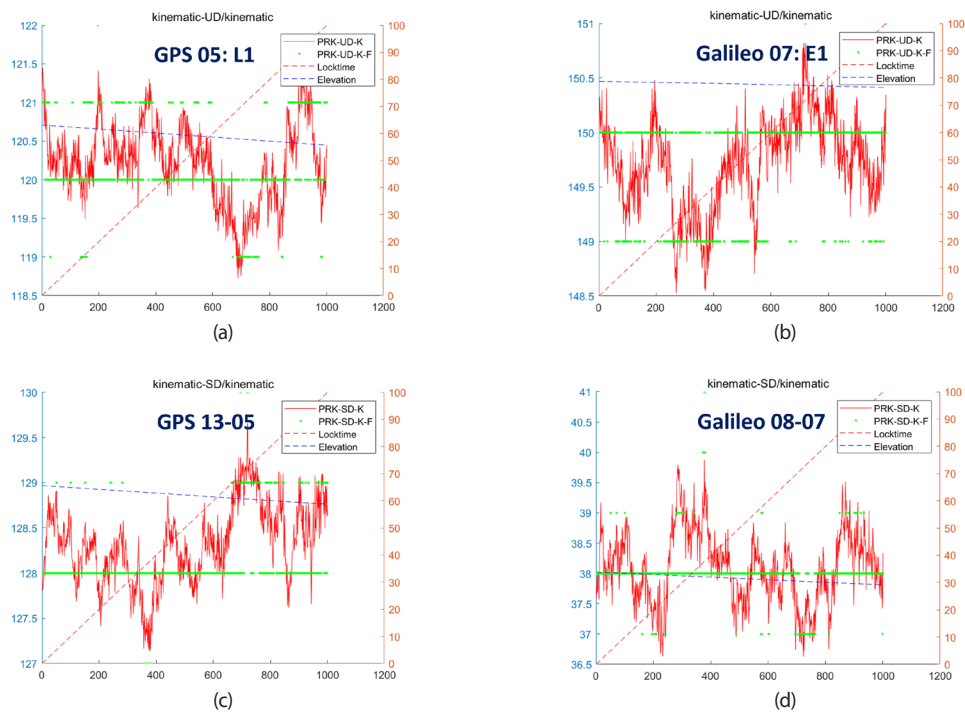


Fig. 3. Epoch-by-epoch WLS-based integer ambiguity resolution (dual-frequencies) (a) GPS PRN 05, (b) Galileo PRN 07, (c) GPS SD (13,05), (d) Galileo SD (08,07).

Fig. 3에 미지실수(float ambiguity)와 LAMBDA로 구한 미지정수(integer ambiguity)를 나타내었다. Fig. 3에서 elevation는 양각을 나타내고, locktime은 신호가 시작부터 연속적으로 수신되는 횟수를 나타내며, 신호가 없으면 0으로 초기화된다. Fig. 3에서 보는 바와 같이 미지정수 결정의 결과에 따라 위치가 바뀐다. UD model에서는 결정된 미지정수가 GPS PRN 05는 118~122 사이, Galileo PRN 07은 149~150 사이의 값을 가진다. SD model에서는 GPS PRN 13과 05 사이의 차분된 미지정수 값이 127~130, Galileo는 PRN 08과 07 사이의 차분된 값이 37~40 사이의 분포를 가진다. 미지정수 중 하나라도 바뀌면 위치가 바뀐다. 정밀한 결과를 구하기 위해서는 안정적인 미지정수의 결정이 필요함을 알 수 있다.

4. RECURSIVE BATCH POSITIONING

3장에서 정리한 epoch-by-epoch 기법은 미지정수가 모든 epoch 측정치에서 변화하지 않는 상수라는 점을 고려하지 않고 매 epoch 새로운 값으로 구한다. 이번 장의 batch processing은 미지정수가 상수라는 특성을 바탕으로 모든 epoch의 측정치를 이용하여 더 정확한 미지정수를 구하는 방법이다. Recursive batch processing은 batch processing을 recursive로 형태를 바꾼 방식으로 실시간 구현에 용이하다는 이점을 가진다.

4.1 Batch Processing

k epoch에서 선형화한 반송파와 코드 측정식 Eq. (9)를 Eq. (18)과 같이 변형하여 나타낸다.

$$\begin{aligned} y(k) &= \begin{bmatrix} G_L(k) & \Lambda \\ G_C(k) & 0 \end{bmatrix} \begin{bmatrix} \delta u(k) \\ a \end{bmatrix} + \begin{bmatrix} \varepsilon_L(k) \\ \varepsilon_C(k) \end{bmatrix} \\ &= \begin{bmatrix} G_L(k) \\ G_C(k) \end{bmatrix} \delta u(k) + \begin{bmatrix} \Lambda \\ 0 \end{bmatrix} a + \begin{bmatrix} \varepsilon_L(k) \\ \varepsilon_C(k) \end{bmatrix} \\ &= G_{LC}(k) \delta u(k) + \Lambda_0 a + \varepsilon(k) \sim N(0, Q_y(k)) \end{aligned} \quad (18)$$

여기서 미지정수가 시 불변 상수임을 고려하면 k epoch에서 해당 epoch의 측정치뿐 아니라 이전의 모든 측정치를 이용하면 더 정확한 값을 구할 수 있다. Eq. (19)와 같은 측정식을 이용하면 1부터 k epoch까지의 모든 측정치를 이용하여 k 개의 사용자 오차(3차원 위치 + 수신기 시계 오차)와 상수 미지정수를 구할 수 있다.

$$\begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(k) \end{bmatrix} = \begin{bmatrix} G_{LC}(1) & 0 & \cdots & 0 & \Lambda_0 \\ 0 & G_{LC}(2) & \cdots & 0 & \Lambda_0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & G_{LC}(k) & \Lambda_0 \end{bmatrix} \begin{bmatrix} \delta u(1) \\ \delta u(2) \\ \vdots \\ \delta u(k) \\ a \end{bmatrix} + \begin{bmatrix} \varepsilon(1) \\ \varepsilon(2) \\ \vdots \\ \varepsilon(k) \end{bmatrix} \sim N(0, \text{diag}(Q_y(k))) \quad (19)$$

위의 형태는 측정치가 증가하면 사용하기 힘든 batch 형태로 수신기에서의 구현을 고려하면 recursive 형태로 변경되어야 한다. 이번 절에서는 일반적인 recursive 형태를 구하기 전에 먼저 $k=1$ 인 경우에 대해 사용자 오차와 미지정수를 구하는 과정을 살펴보고 이를 확장하여 일반적인 형태를 구한다.

첫 epoch에서 주어진 측정치를 이용하여 해를 구하기 위해 Eq. (20)과 같은 측정식을 고려한다.

$$y(1) = G_{LC}(1) \delta u(1) + \Lambda_0 a + \varepsilon(1) = \begin{bmatrix} G_{LC}(1) & \Lambda_0 \end{bmatrix} \begin{bmatrix} \delta u(1) \\ a(1) \end{bmatrix} \sim N(0, Q_y(1)) \quad (20)$$

Eq. (20)을 Eq. (21)과 같이 정규 방정식으로 나타내고 Eq. (22)와 같이 간단한 형태로 정리할 수 있다.

$$\left[\begin{bmatrix} G_{LC}^T(1) \\ \Lambda_0^T \end{bmatrix} Q_{LC}^{-1}(1) \begin{bmatrix} G_{LC}(1) & \Lambda_0 \end{bmatrix} \right] \begin{bmatrix} \delta \hat{u}(1) \\ \hat{a}(1) \end{bmatrix} = \begin{bmatrix} G_{LC}^T(1) \\ \Lambda_0^T \end{bmatrix} Q_{LC}^{-1}(1) \quad (21)$$

$$\begin{bmatrix} G_{LC}^T(1) Q_{LC}^{-1}(1) G_{LC}(1) & G_{LC}^T(1) Q_{LC}^{-1}(1) \Lambda_0 \\ \Lambda_0^T Q_{LC}^{-1}(1) G_{LC}(1) & \Lambda_0^T Q_{LC}^{-1}(1) \Lambda_0 \end{bmatrix} \begin{bmatrix} \delta \hat{u}(1) \\ \hat{a}(1) \end{bmatrix} = \begin{bmatrix} M_{GG} & M_{GA} \\ M_{AG} & M_{AA} \end{bmatrix} \begin{bmatrix} \delta \hat{u}(1) \\ \hat{a}(1) \end{bmatrix} = \begin{bmatrix} G_{LC}^T(1) \\ \Lambda_0^T \end{bmatrix} Q_{LC}^{-1}(1) y_{LC}(1) \quad (22)$$

위의 $M_{GG}^{-1} M_{GG} \delta \hat{u}(1) + M_{GG}^{-1} M_{GA} \hat{a}(1) = M_{GG}^{-1} \{G_{LC}^T(1) Q_{LC}^{-1}(1) y_{LC}(1)\}$ 로부터 사용자 오차 벡터는 Eq. (23)과 같이 구할 수 있다.

$$\delta \hat{u}(1) = M_{GG}^{-1} \{G_{LC}^T(1) Q_{LC}^{-1}(1) y_{LC}(1) - M_{GA} \hat{a}(1)\} \quad (23)$$

그리고 Eq. (23)의 관계를 $M_{AG} \delta \hat{u}(1) + M_{AA} \hat{a}(1) = \Lambda_0^T Q_{LC}^{-1}(1) y_{LC}(1)$ 에 대입하면 Eq. (24)와 같다.

$$(M_{AA} - M_{AG} M_{GG}^{-1} M_{GA}) \hat{a}(1) = (\Lambda_0^T - M_{AG} M_{GG}^{-1} G_{LC}^T(1)) Q_{LC}^{-1}(1) y_{LC}(1) \quad (24)$$

Eq. (24)로부터 미지정수를 구할 수 있고, 이를 Eq. (23)에 대입하면 사용자 오차 벡터를 구할 수 있다. 이후 Eq. (25)와 같은 간략한 형태를 사용한다.

$$\hat{a}(1) = X^{-1} Y Q_{LC}^{-1} y_{LC}, \text{cov}(\hat{a}(1)) = X^{-1} Y Q_{LC}^{-1} Y^T X^{-T} \quad (25)$$

첫 epoch의 결과를 k epoch으로 확장하기 위하여 Eq. (19)를 Eq. (26)의 형태로 나타낸다.

$$y_{LC}(1:k) = \begin{bmatrix} G_{LC}(1:k) & \Lambda_0(1:k) \end{bmatrix} \begin{bmatrix} \delta u(1:k) \\ a(k) \end{bmatrix} \sim N(0, Q_{LC}(1:k)) \quad (26)$$

Eq. (26)에 대해 정규 방정식을 구하고 전술한 방법과 같은 과정으로 정리하면 Eq. (27)과 같다.

$$\begin{bmatrix} G_{LC}^T(1:k) Q_{LC}^{-1}(1:k) G_{LC}(1:k) & G_{LC}^T(1:k) Q_{LC}^{-1}(1:k) \Lambda_0(1:k) \\ \Lambda_0^T(1:k) Q_{LC}^{-1}(1:k) G_{LC}(1:k) & \Lambda_0^T(1:k) Q_{LC}^{-1}(1:k) \Lambda_0(1:k) \end{bmatrix} \begin{bmatrix} \delta \hat{u}(1:k) \\ \hat{a}(k) \end{bmatrix} = \begin{bmatrix} M_{GG}(1:k) & M_{GA}(1:k) \\ M_{AG}(1:k) & M_{AA}(1:k) \end{bmatrix} \begin{bmatrix} \delta \hat{u}(1:k) \\ \hat{a}(k) \end{bmatrix} = \begin{bmatrix} G_{LC}^T(1:k) \\ \Lambda_0^T(1:k) \end{bmatrix} Q_{LC}^{-1}(1:k) y_{LC}(1:k) \quad (27)$$

이를 이용하여 미지정수를 구하면 Eq. (28)과 같다.

$$\begin{aligned} \hat{a}(k) &= (M_{AA}(1:k) - M_{AG}(1:k) M_{GG}^{-1}(1:k) M_{GA}(1:k))^{-1} (\Lambda_0^T(1:k) \\ &\quad - M_{AG}(1:k) M_{GG}^{-1}(1:k) G_{LC}^T(1:k)) Q_{LC}^{-1}(1:k) y_{LC}(1:k) \end{aligned} \quad (28)$$

Eq. (28)을 간단한 형태로 나타내면 Eq. (29)와 같다.

$$\begin{aligned} \hat{a}(k) &= X^{-1}(1:k) Y(1:k) Q_{LC}^{-1}(1:k) y_{LC}(1:k) \\ \text{cov}(\hat{a}(k)) &= X^{-1}(1:k) Y(1:k) Q_{LC}^{-1}(1:k) Y^T(1:k) X^{-T}(1:k) \end{aligned} \quad (29)$$

Eq. (29)를 통해 구한 미지정수를 이용하면 Eq. (30)과 같이 1부터 k epoch까지의 모든 사용자 오차 벡터를 구할 수 있다.

$$\delta \hat{u}(1:k) = M_{GG}^{-1}(1:k) \{G_{LC}^T(1:k) Q_{LC}^{-1}(1:k) y_{LC}(1:k) - M_{GA}(1:k) \hat{a}(k)\} \quad (30)$$

측지가 아닌 항법에서는 k 번째 epoch에서의 해가 필요하며, Eq. (31)과 같이 구할 수 있다.

$$\begin{aligned}\delta\hat{u}(k) &= M_{GG}^{-1}(k)\{G_{LC}^T(k)Q_{LC}^{-1}(k)y_{LC}(k) - M_{GA}(k)\hat{a}(k)\} \\ &= [G_{LC}^T(k)Q_{LC}^{-1}(k)G_{LC}(k)]^{-1}G_{LC}^T(k)Q_{LC}^{-1}(k)\{y_{LC}(k) - \Lambda_0\hat{a}(k)\} \quad (31)\end{aligned}$$

위의 과정은 1에서 k epoch까지의 측정치를 이용하여 Eq. (29)로 미지정수를 구하고 이를 Eq. (31)에 대입하여 k epoch에서 사용자 오차벡터를 구하는 과정이다. Eq. (31)에서 $y_{LC}(k) - \Lambda_0\hat{a}(k)$ 는 구한 미지정수를 측정치에서 제거한 형태이고 이를 이용하여 해를 구할 수 있다. Epoch 수가 증가할수록 미지정수의 정확도가 향상되며 이전 epoch에서의 사용자 오차도 향상된다. 즉 $u(1), \dots, u(k-1)$ 는 k 가 증가할수록 더 정확한 결과로 개선된다. 이는 k epoch에서의 결과만 중요한 항법 응용에서는 중요하지 않지만 측지나 측량에서는 아주 중요한 특성이다.

4.2 Recursive Batch Processing

Eq. (29)를 recursive 형태로 변경하기 위해서는 각 항의 recursive한 형태가 필요하며, 이번 절에서는 $X^{-1}(1:k), Y(1:k)Q_{LC}^{-1}(1:k)y_{LC}(1:k), Y(1:k)Q_{LC}^{-1}(1:k)Y^T(1:k)$ 세 항의 recursive한 형태를 유도하고 이를 이용하여 구현한다.

먼저 각 항의 정의로부터 Eqs. (32, 33)과 같이 정리할 수 있다.

$$\begin{aligned}X(1:k) &= (M_{AA}(1:k) - M_{AG}(1:k)M_{GG}^{-1}(1:k)M_{GA}(1:k))^{-1} \\ &= \left[\sum_{i=1}^k \Lambda_0^T \left\{ Q_{LC}^{-1}(i) - Q_{LC}^{-1}(i)G_{LC}(i) \left(G_{LC}^T(i)Q_{LC}^{-1}(i)G_{LC}(i) \right)^{-1} G_{LC}^T(i)Q_{LC}^{-1}(i) \right\} \Lambda_0 \right]^{-1} \quad (32)\end{aligned}$$

$$\begin{aligned}Y(1:k)Q_{LC}^{-1}(1:k)y_{LC}(1:k) &= \left(\Lambda_0^T(1:k) - M_{AG}(1:k)M_{GG}^{-1}(1:k)G_{LC}^T(1:k) \right) Q_{LC}^{-1}(1:k)y_{LC}(1:k) \\ &= \Lambda_0^T \sum_{i=1}^k \left(I - Q_{LC}^{-1}(i)G_{LC}(i) \left(G_{LC}^T(i)Q_{LC}^{-1}(i)G_{LC}(i) \right)^{-1} G_{LC}^T(i) \right) Q_{LC}^{-1}(i)y_{LC}(i) \quad (33)\end{aligned}$$

Eq. (32)의 recursive 형태를 구하는 과정은 Eq. (34)와 같다.

$$\begin{aligned}X^{-1}(1:k) &= \left[\left(\sum_{i=1}^{k-1} x(i) \right) + x(k) \right]^{-1} \\ &= \left(\sum_{i=1}^{k-1} x(i) \right)^{-1} - \left(\sum_{i=1}^{k-1} x(i) \right)^{-1} \left\{ \left(\sum_{i=1}^{k-1} x(i) \right)^{-1} + x^{-1}(k) \right\}^{-1} \left(\sum_{i=1}^{k-1} x(i) \right)^{-1} \\ &= X^{-1}(1:k-1) - X^{-1}(1:k-1)[X^{-1}(1:k-1) + x^{-1}(k)]^{-1}X^{-1}(1:k-1) \quad (34)\end{aligned}$$

여기서 $x(k) = \Lambda_0^T \{ Q_{LC}^{-1}(k) - Q_{LC}^{-1}(k)G_{LC}(k) \left(G_{LC}^T(k)Q_{LC}^{-1}(k)G_{LC}(k) \right)^{-1} G_{LC}^T(k)Q_{LC}^{-1}(k) \} \Lambda_0$ 는 k epoch에서의 값이다. 이 식을 이용하면 $k-1$ epoch까지의 $X^{-1}(1:k-1)$ 에 새로 추가된 k epoch에서의 값을 이용하여 k epoch에서의 $X^{-1}(1:k)$ 를 구할 수 있다. 이 과정을 다음과 같이 두 단계로 정리할 수 있다: 1) Eq. (35)와 같이 epoch 1 ($k=1$)에서의 값을 계산하여 이를 누적의 초기값으로 설정하고, 2) Eqs. (36, 37)을 이용하여 k epoch에서의 누적값은 현재값과 이전 누적값을 통해 구한다.

$$X^{-1}(1:1) = x^{-1}(1) = \left[\Lambda_0^T \left\{ Q_{LC}^{-1}(1) - Q_{LC}^{-1}(1)G_{LC}(1) \left(G_{LC}^T(1)Q_{LC}^{-1}(1)G_{LC}(1) \right)^{-1} G_{LC}^T(1)Q_{LC}^{-1}(1) \right\} \Lambda_0 \right]^{-1} \quad (35)$$

$$x^{-1}(k) = \left[\Lambda_0^T \left\{ Q_{LC}^{-1}(k) - Q_{LC}^{-1}(k)G_{LC}(k) \left(G_{LC}^T(k)Q_{LC}^{-1}(k)G_{LC}(k) \right)^{-1} G_{LC}^T(k)Q_{LC}^{-1}(k) \right\} \Lambda_0 \right]^{-1} \quad (36)$$

$$X^{-1}(1:k) = X^{-1}(1:k-1) - X^{-1}(1:k-1)[X^{-1}(1:k-1) + x^{-1}(k)]^{-1}X^{-1}(1:k-1) \quad (37)$$

Eq. (33)의 recursive 형태를 구하는 과정은 Eq. (38)과 같다.

$$\begin{aligned}Y(1:k)Q_{LC}^{-1}(1:k)y_{LC}(1:k) &= \Lambda_0^T \sum_{i=1}^k \left(I - Q_{LC}^{-1}(i)G_{LC}(i) \left(G_{LC}^T(i)Q_{LC}^{-1}(i)G_{LC}(i) \right)^{-1} G_{LC}^T(i) \right) Q_{LC}^{-1}(i)y_{LC}(i) \\ &= Y(1:k-1)Q_{LC}^{-1}(1:k-1)y_{LC}(1:k-1) \\ &\quad + \Lambda_0^T \left\{ I - Q_{LC}^{-1}(k)G_{LC}(k) \left(G_{LC}^T(k)Q_{LC}^{-1}(k)G_{LC}(k) \right)^{-1} G_{LC}^T(k) \right\} Q_{LC}^{-1}(k)y_{LC}(k) \quad (38)\end{aligned}$$

Eq. (38)을 이용하면 $k-1$ epoch까지의 누적 값에 새로 추가된 k epoch에서의 값을 통해 k epoch에서의 누적 값을 구할 수 있다. 이 과정을 다음과 같이 두 단계로 정리할 수 있다: 1) Eq. (39)와 같이 epoch 1 ($k=1$)에서의 값을 계산하여 이를 누적의 초기값으로 설정하고, (2) Eq. (40)과 같이 k epoch에서의 누적값은 현재값과 이전 누적값을 이용하여 구한다.

$$Y(1:1)Q_{LC}^{-1}(1:1)y_{LC}(1:1) = \Lambda_0^T \left\{ I - Q_{LC}^{-1}(1)G_{LC}(1) \left(G_{LC}^T(1)Q_{LC}^{-1}(1)G_{LC}(1) \right)^{-1} G_{LC}^T(1) \right\} Q_{LC}^{-1}(1)y_{LC}(1) \quad (39)$$

$$\begin{aligned}Y(1:k)Q_{LC}^{-1}(1:k)y_{LC}(1:k) &= Y(1:k-1)Q_{LC}^{-1}(1:k-1)y_{LC}(1:k-1) \\ &\quad + \Lambda_0^T \left\{ I - Q_{LC}^{-1}(k)G_{LC}(k) \left(G_{LC}^T(k)Q_{LC}^{-1}(k)G_{LC}(k) \right)^{-1} G_{LC}^T(k) \right\} Q_{LC}^{-1}(k)y_{LC}(k) \quad (40)\end{aligned}$$

이상의 결과로 미지정수 $\hat{a}(k) = \{X^{-1}(1:k)\}^{-1}Y(1:k)Q_{LC}^{-1}(1:k)y_{LC}(1:k)$ 는 recursive 형태로 계산이 가능 하지만 공분산 $\text{cov}(\hat{a}(k)) = \{X^{-1}(1:k)\}^{-1}Y(1:k)Q_{LC}^{-1}(1:k)Y^T(1:k)\{X^{-1}(1:k)\}^{-1}$ 를 계산하기 위해서는 추가로 $Y(1:k)Q_{LC}^{-1}(1:k)Y^T(1:k)$ 의 recursive 형태가 필요하다. $Y(1:k)Q_{LC}^{-1}(1:k)Y^T(1:k)$ 는 Eq. (41)과 같이 정리할 수 있다.

$$\begin{aligned}Y(1:k)Q_{LC}^{-1}(1:k)Y^T(1:k) &= \Lambda_0^T \sum_{i=1}^k \left\{ I - Q_{LC}^{-1}(i)G_{LC}(i) \left(G_{LC}^T(i)Q_{LC}^{-1}(i)G_{LC}(i) \right)^{-1} G_{LC}^T(i) \right\} \\ &\quad Q_{LC}^{-1}(i) \left\{ I - Q_{LC}^{-1}(i)G_{LC}(i) \left(G_{LC}^T(i)Q_{LC}^{-1}(i)G_{LC}(i) \right)^{-1} G_{LC}^T(i) \right\}^T \Lambda_0 \quad (41)\end{aligned}$$

Eq. (41)은 k epoch에서의 값이 추가만 되는 형태이므로 다음과 같이 두 단계로 구할 수 있다: 1) Eq. (42)와 같이 epoch 1 ($k=1$)에서의 값을 계산하여 이를 누적의 초기값으로 설정하고, 2) Eq. (43)과 같이 k epoch에서의 누적값은 현재값과 이전 누적값을 이용하여 구한다.

$$\begin{aligned}Y(1:1)Q_{LC}^{-1}(1:1)Y^T(1:1) &= \Lambda_0^T \left\{ I - Q_{LC}^{-1}(1)G_{LC}(1) \left(G_{LC}^T(1)Q_{LC}^{-1}(1)G_{LC}(1) \right)^{-1} G_{LC}^T(1) \right\} Q_{LC}^{-1}(1) \\ &\quad \left\{ I - Q_{LC}^{-1}(1)G_{LC}(1) \left(G_{LC}^T(1)Q_{LC}^{-1}(1)G_{LC}(1) \right)^{-1} G_{LC}^T(1) \right\}^T \Lambda_0 \quad (42)\end{aligned}$$

$$\begin{aligned}Y(1:k)Q_{LC}^{-1}(1:k)Y^T(1:k) &= Y(1:k-1)Q_{LC}^{-1}(1:k-1)Y^T(1:k-1) \\ &\quad + \Lambda_0^T \left\{ I - Q_{LC}^{-1}(k)G_{LC}(k) \left(G_{LC}^T(k)Q_{LC}^{-1}(k)G_{LC}(k) \right)^{-1} G_{LC}^T(k) \right\} Q_{LC}^{-1}(k) \\ &\quad \left\{ I - Q_{LC}^{-1}(k)G_{LC}(k) \left(G_{LC}^T(k)Q_{LC}^{-1}(k)G_{LC}(k) \right)^{-1} G_{LC}^T(k) \right\}^T \Lambda_0 \quad (43)\end{aligned}$$

위의 과정을 통해 구한 실수해와 공분산에 LAMBDA를 적용하여 매 epoch 정수 미지정수를 구하는 미지정수 결정기법을 적용한다. 이 과정에서 구한 미지정수는 1부터 k epoch 사이의 모든 측정치를 사용하여 구한 미지정수로 epoch-by-epoch 기법에 비하여 참 미지정수를 구할 확률이 높아진다.

4.3 Experimental Results

이번 절에서는 recursive batch processing 결과를 제시한다. 실험 환경은 epoch-by-epoch processing에서 설정한 바와 동일

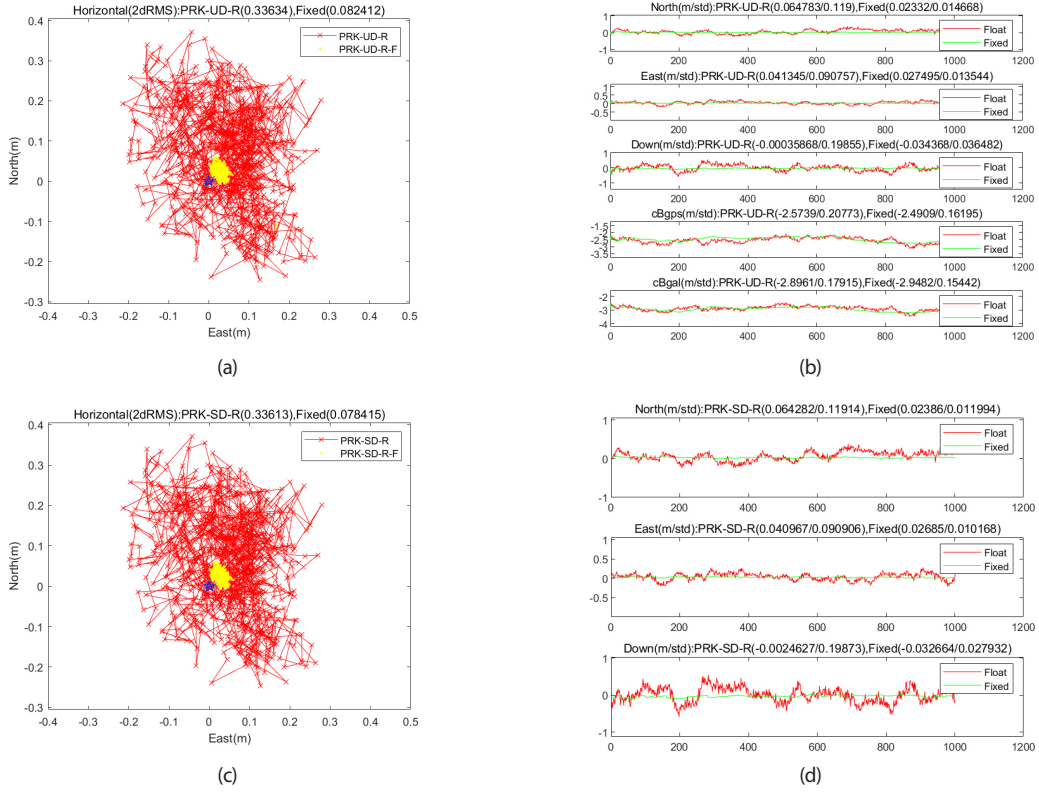


Fig. 4. Recursive batch WLS-based navigation results for UD and SD models (dual-frequencies): (a) UD model horizontal plot, (b) UD model NED plot, (c) SD model horizontal plot and (d) SD model NED plot.

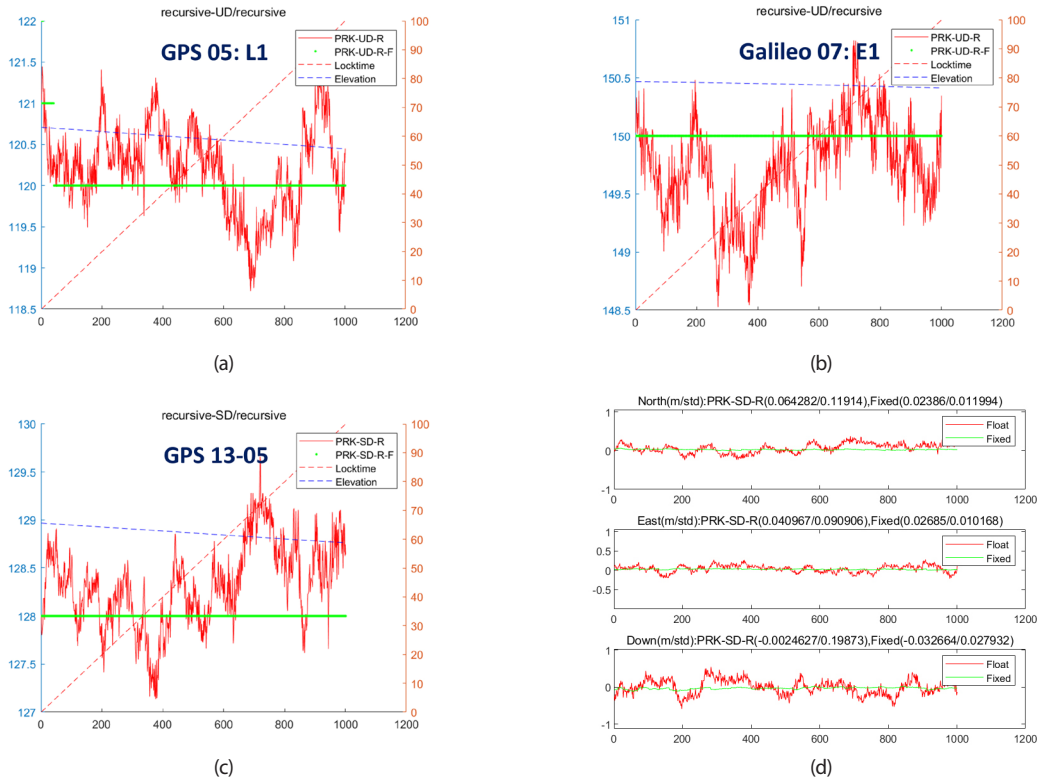


Fig. 5. Recursive batch WLS-based integer ambiguity resolution (dual-frequencies) (a) GPS PRN 05, (b) Galileo PRN 07, (c) GPS SD (13,05), (d) Galileo SD (08,07).

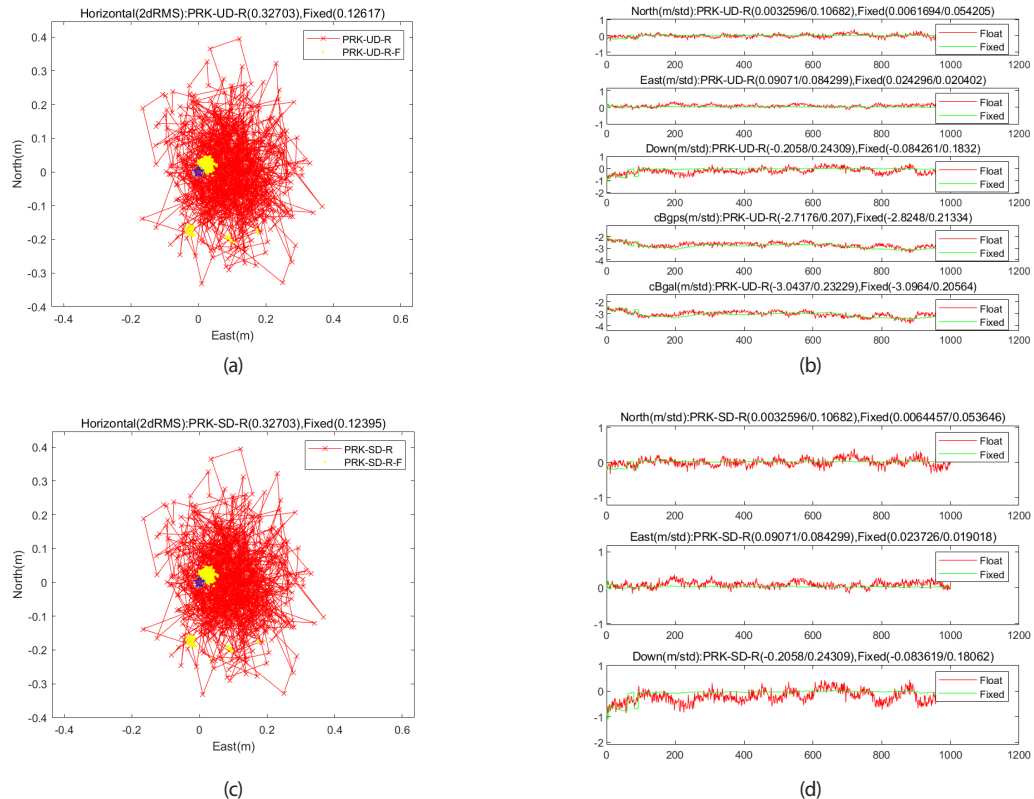


Fig. 6. Recursive batch WLS-based navigation results for UD and SD models (single-frequency): (a) UD model horizontal plot, (b) UD model NED plot, (c) SD model horizontal plot and (d) SD model NED plot.

하게 적용하였다.

4.3.1 Dual-frequencies UD/SD model

Fig. 4에 항법 결과를, Fig. 5에 미지정수를 나타냈다. Fig. 4에서 실수해 SD (0.33613 m)와 UD (0.33634 m)는 epoch-by-epoch과 같으나, 정수해는 SD (0.078415 m), UD (0.082412 m)로 급격히 줄어들을 볼 수 있다. 그리고 SD 모델에서는 수신기 시계오차가 없으며, UD 모델에서는 GPS와 Galileo 시계오차를 따로 구한 것을 볼 수 있다. Fig. 5에서 UD는 처음 20 epoch 이후에 수렴된 미지정수를 얻는 반면, SD는 처음부터 참 미지정수를 구한 것을 확인할 수 있다.

4.3.2 Single-frequency UD/SD model

이번 절에서는 single frequency만 처리 가능한 저가형 수신기에 제안하는 recursive batch processing 기법의 적용 가능성을 검토하기 위해 GPS L1, Galileo E1 측정치만을 사용했을 때의 항법 결과를 제시한다. Fig. 6에 항법 결과를, Fig. 7에 미지정수를 나타냈다. UD는 평균 94 epoch 이후에 수렴된 미지정수를 얻고, SD는 평균 63 epoch 이후에 수렴된 미지정수를 얻지만, 정수해의 수평오차는 차이가 크지 않다. 결과를 통해 단일 주파수 수신기에서도 제안하는 recursive batch 알고리즘의 적용이 가능할 것으로 기대된다.

4.3.3 Comparison with CLASLIB

이번 절에서 제안하는 recursive batch processing 기법의 항법 결과가 타당한지 확인하기 위해, 공개 소프트웨어 PPP-RTK 플랫폼인 CLASLIB과 항법 결과를 비교한다. CLASLIB을 구동하는데 필요한 파라미터는 공식 홈페이지의 “kinematic.conf” 파일을 사용했고, 위성군과 주파수는 이번 장에서 제시한 실험 환경과 동일하게 설정했다. CLASLIB은 SD model에 EKF를 사용하여 항법 결과를 제공하므로, SD model의 결과만 Fig. 8에 제시하였다. 제안한 방법은 누적된 측정치를 사용하여 미지정수를 구하고, 이 미지정수를 epoch-by-epoch 측정치에 적용하여 위치를 구하는 반면 CLASLIB는 EKF로 구하므로 약간의 차이를 보이지만 두 결과가 거의 유사함을 확인할 수 있다. CLASLIB의 결과는 위치 수렴에 6 epoch의 시간이 필요한 반면, 제안한 방법은 첫 epoch부터 수렴된 결과를 보이며 이는 미지정수 결정에서 제안한 방법의 성능이 우수함을 나타낸다. 이상의 결과로 제안한 recursive batch 알고리즘의 타당성을 볼 수 있다.

5. CONCLUSION

이 논문에서는 SSR과 관측모델 보정이 된 코드와 반송파 위상 측정치를 이용하여 효과적으로 미지정수를 구하고, 구해진 미지정수를 이용하여 위치해를 구하는 recursive batch 항법 알고리

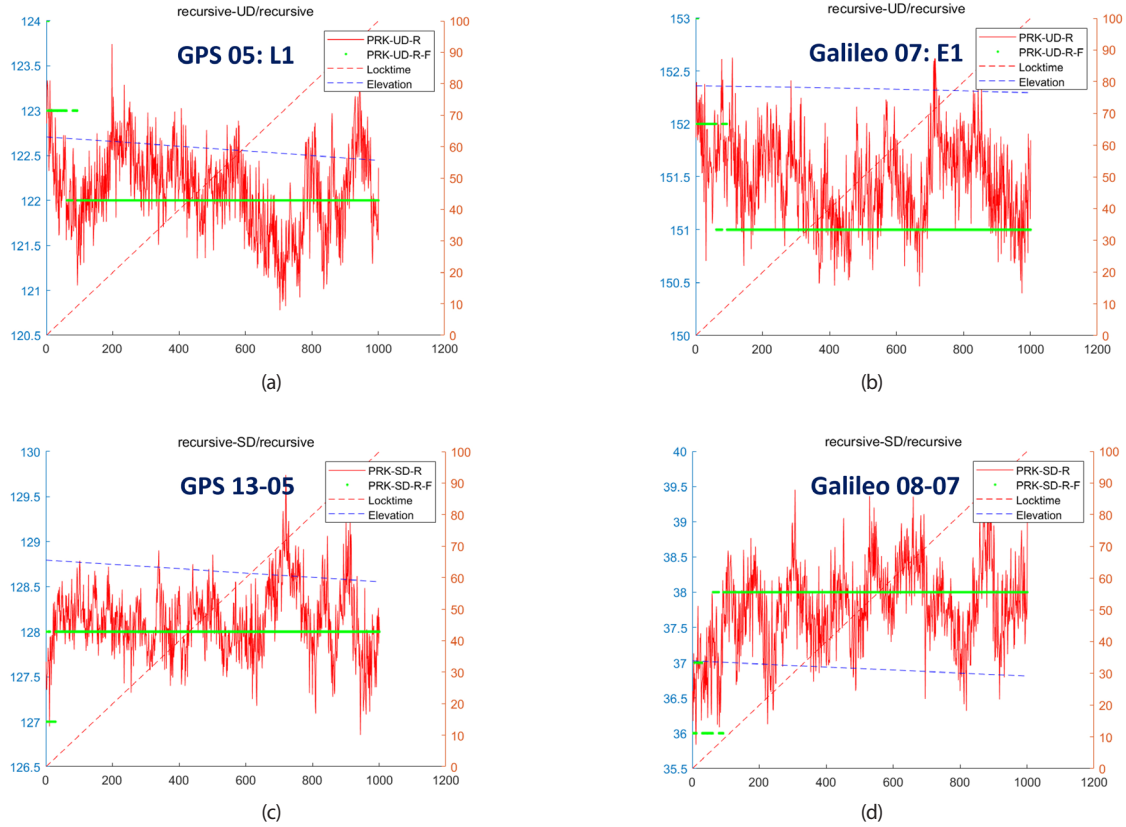


Fig. 7. Recursive batch WLS-based integer ambiguity resolution (single-frequency) (a) GPS PRN 05, (b) Galileo PRN 07, (c) GPS SD (13,05), (d) Galileo SD (08,07).

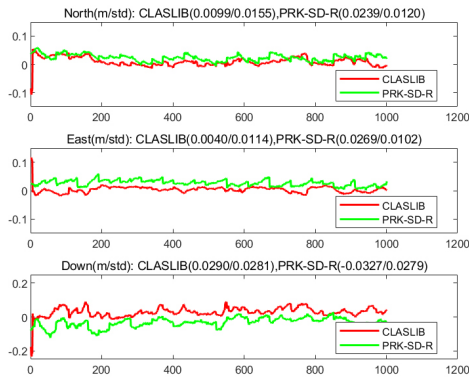


Fig. 8. Navigation results comparison between SD model/recursive batch WLS and CLASLIB (dual-frequencies).

증을 제안하였다. 이를 위하여 UD와 SD 측정치 모델을 유도하고, 모델을 이용한 epoch-by-epoch 항법 알고리즘을 구현하였다. 실제 측정치를 이용한 결과로부터 단일 측정치 사용의 한계를 확인하고 이를 극복하기 위하여 모든 측정치를 사용하는 batch 항법 알고리즘을 유도하고 실시간 구현이 가능한 recursive batch 항법 알고리즘을 제안하였다.

QZSS CLAS의 GPS, Galileo 이중주파수 측정치를 이용한 실험결과 제안한 방법은 이중주파수를 사용하는 경우 수 초 이내, 단일 주파수를 사용하는 경우 2분 이내 미지정수를 결정할 수 있음을 확인했다. 구체적으로 UD 모델을 사용하는 경우 이중주파

수 20초, 단일 주파수 94초의 수렴시간이 필요한 반면 SD 모델을 사용하면 이중주파수 1초, 단일주파수 63초의 수렴시간이 필요했다. 공개 소프트웨어 PPP-RTK 플랫폼인 CLASLIB을 사용한 결과와 대등하거나 더 나은 결과이며, 제안한 방법이 GNSS 수신기 구현에 충분히 적용될 수 있을 것으로 기대된다.

ACKNOWLEDGMENTS

This research was supported by funding from Korea government (KASA, Korea AeroSpace Administration) (grant number RS-2022-00165808).

AUTHOR CONTRIBUTIONS

Conceptualization, J. Park, and C. Park; methodology, J. Jeong, G. Park, and C. Park; software, J. Jeong, G. Park, and C. Park; formal analysis, J.-Y. Park, Y.H. Ko, and D. Park; investigation, J. Jeong, and G. Park; writing—original draft preparation, J. Jeong; writing—review and editing, J. Jeong, and C. Park; visualization, J. Jeong; supervision, C. Park; funding acquisition, J.-Y. Park, Y.H. Ko, and D. Park.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

- An, X., Ziebold, R., & Lass, C. 2023, PPP-RTK with rapid convergence based on SSR corrections and its application in transportation, *Remote Sensing*, 15, 4770. <https://doi.org/10.3390/rs15194770>
- Cabinet Office 2025, IS-QZSS-L6-007, [Internet], cited 2025 Jul 31, available from: <https://qzss.go.jp/en/technical/download/pdf/ps-is-qzss/is-qzss-l6-007.pdf?t=1770265882860>
- Geng, J., Meng, X., Dodson, A. H., & Teferle, F. N. 2010, Integer ambiguity resolution in precise point positioning: method comparison, *Journal of Geodesy*, 84, 569-581. <https://doi.org/10.1007/s00190-010-0399-x>
- Hirokawa, R., Fernández-Hernández, I., & Reynolds, S. 2021, PPP/PPP-RTK open formats: Overview, comparison, and proposal for an interoperable message, *NAVIGATION: Journal of the Institute of Navigation*, 68, 759-778. <https://doi.org/10.1002/navi.452>
- Jeong, J., Park, J., Shin, S.-Y., Ko, Y. H., Lim, J., et al. 2025, Implementation of a PPP-RTK Reception Platform and Processing of NGII and QZSS CLAS SSR Corrections, 2025 IPNT Conference, Nov 4-8 2025, Jeju, Korea, pp.809-813. <https://ipnt.or.kr/2025proc/114>
- Lim, C., Jo, Y., Lee, Y., Cha, Y., Park, B., et al. 2022a, Trends of Open PPP/PPP-RTK Correction Services, *Journal of Advanced Navigation Technology*, 26, 418-426. <https://doi.org/10.12673/jant.2022.26.6.418>
- Lim, C., Lee, Y., Cha, Y., Park, B., Park, S. G., et al. 2022b, Monitoring QZSS CLAS-based VRS-RTK Positioning Performance, *Journal of Positioning, Navigation, and Timing*, 11, 251-261. <https://doi.org/10.11003/JPNT.2022.11.4.251>
- Lim, C.-S., Park, B., Kim, D.-U., Kee, C.-D., Park, K.-D., et al. 2018, Feasibility Study on Integration of SSR Correction into Network RTK to Provide More Robust Service, *Journal of Positioning, Navigation, and Timing*, 7, 295-305. <https://doi.org/10.11003/JPNT.2018.7.4.295>
- Ong, J., Park, S. G., Park, S. H., & Park, C. 2021, Performance expectation of single station PPP-RTK using dual-frequency GPS measurement in Korea, *Journal of Positioning, Navigation, and Timing*, 10, 159-168. <https://doi.org/10.11003/JPNT.2021.10.3.159>
- Park, G. 2025, Implementation and Performance Evaluation of MCMF PPP-RTK Using SSR Corrections, Master's thesis, Chungbuk National University.
- Park, G.-M., Lee, G.-H., Ko, Y.-H., Park, D., & Park, C. 2024, The Implementation and Performance of SSR Decoder and SSR2OSR for National Geographic Information Sources Based on Matlab, 2024 IPNT Conference, Nov 6-8 2024, Jeju, Korea, pp.253-256. <https://ipnt.or.kr/2024proc/100>
- Park, K.-D., Kim, M., Kim, Y.-G., Park, J.-H., Bae, D.-H., et al. 2021, Development of Smartphone Positioning Technology Using Correction Signals, [Internet], cited 2021 Dec 8, available from: https://www.ngii.go.kr/kor/board/view.do?sq=75385&board_code=notice_ko
- Song, J., Jeon, T., Park, S. H., & Park, S. G. 2024, Design of Real-time SSR2OSR Conversion Software Using POINT Service for Maritime GNSS Applications, *Journal of Positioning, Navigation, and Timing*, 13, 475-483. <https://doi.org/10.11003/JPNT.2024.13.4.475>
- Teunissen, P. J. & Montenbruck, O. (Eds.) 2017, Springer handbook of global navigation satellite systems (Cham, Switzerland: Springer International Publishing).
- Wabben, G., Schmitz, M., & Bagge, A. 2005, PPP-RTK: precise point positioning using state-space representation in RTK networks, In *Proceedings of the 18th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2005)*, September 13-16, 2005, Long Beach, CA, pp.2584-2594. <https://www.ion.org/publications/abstract.cfm?articleID=6467>



system.

Junwoo Jeong received the B.S. degree in Electronics Engineering from Chungnam National University in 2023. He is currently a Ph.D. candidate at Electronics Engineering from Chungnam National University. His research interests include nonlinear control, adaptive control, robotic control, and navigation



physical system and inertial navigation system.

JunMin Park received the Ph.D. degree in Electrical and Electronics Engineering from PSTECH in 2019. He was a principal researcher at POSLAB from 2019 to 2021. He is now an associated professor at Chungnam National University since 2021. His research interests include control theory, cyber-



Jae-Young Park received the M.S. degree in Geoinformatic Engineering from Inha University in 2023. He is currently at Korea Aerospace Research Institute. His research interests include GNSS Receiver and PPP-RTK algorithm development.



Yo Han Ko received the Ph.D. degree in Information Communication Engineering from Chung-Ang University, Korea, in 2011. He is now a principal researcher at LIG Nex1 Satellite Lab. He worked on the development and research of multi-beamforming, MIMO-OFDM modems, navigation receiver, and navigation systems. His research interests include Korean Positioning System and LEO navigation systems.



Gyeongmin Park received M.S. degree in the Department of Intelligent Systems & Robotics from Chungbuk National University in 2025. He is now a researcher at LIG Nex1 Satellite Lab since 2025. His research interests include GNSS, SDR, Precise positioning.



Dana Park received the M.S. degree in Aerospace Engineering from Konkuk University in 2023. She is now a researcher at LIG Nex1 Satellite Lab since 2023. Her research interests include satellite navigation system, Korean Positioning System.



Chansik Park received B.S., M.S., and Ph.D. degrees in Electrical Engineering from Seoul National University in 1984, 1986 and 1997, respectively. He has been a Professor with the School of Electronics Engineering, Chungbuk National University, Cheongju, Korea, since 1997. His research interests include GNSS, PNS, SDR, integer ambiguity resolution algorithm and Error Analysis.

