ORIGINAL ARTICLE

Multi-GNSS ultra-rapid orbit determination through epoch-parallel processing

Longjiang Tang[1,2] · Jungang Wang[2,3] · Huizhong Zhu[4] · Maorong Ge[1,2] · Aigong Xu[4] · Harald Schuh[1,2]

Abstract

High-precision Global Navigation Satellite Systems (GNSS) orbits are critical for real-time clock estimation and precise positioning service; however, the prediction error grows gradually with the increasing prediction session. In this study, we present a new efficient precise orbit determination (POD) strategy referred to as the epoch-parallel processing to reduce the orbit update latency, in which a 24-h processing job is split into several sub-sessions that are processed in parallel and then stacked to solve and recover parameters subsequently. With a delicate handling of parameters crossing different sub-sessions, such as ambiguities, the method is rigorously equivalent to the one-session batch solution, but is much more efficient, halving the time-consuming roughly. Together with paralleling other procedures such as orbit integration and using open multi-processing (openMP), the multi-GNSS POD of 120 satellites using 90 stations can be fulfilled within 30 min. The lower update latency enables users to access orbits closer to the estimation part, that is, 30–60-min prediction with a 30-min update latency, which significantly improves the orbit quality. Compared to the hourly updated orbit, the averaged 1D RMS values of predicted orbit in terms of overlap for GPS, GLONASS, Galileo, and BDS MEO are improved by 39%, 35%, 41%, and 37%, respectively, and that of BDS GEO and IGSO satellites is improved by 47%. We also demonstrate that the boundary discontinuities of half-hourly orbit are within 2 cm for the GPS, GLONASS, and Galileo satellites, and for BDS the values are 2.6, 15.5, and 9.8 cm for MEO, GEO, and IGSO satellites, respectively. This method can also be implemented for any batch-based GNSS processing to improve the efficiency.

Keywords Multi-GNSS · Real-time service · Precise orbit determination · Epoch-parallel computation · Normal equation stacking

1 Introduction

Zumberge et al. (1997) introduced the precise point positioning (PPP) concept by using precise satellite orbit and clock products to provide positioning with centimeter accuracy for a stand-alone receiver. Its real-time realization was demonstrated by Héroux et al. (2004). To speed up this development

toward a real-time precise positioning service, the International GNSS Service (IGS) (Johnston et al. 2017) launched its real-time pilot project (IGS-RTPP) (Cassey et al. 2012), aiming to acquire and distribute GNSS data and products in real-time based on the NTRIP protocol in June 2007. Serving as the essential information for real-time precise positioning, satellite orbits are usually estimated with the latest available observations and predicted for real-time applications (Elsobeiey et al. 2016; Li et al. 2022).

The orbit estimation can be carried out by the least-squares batch processing or epoch-wise filtering. The least-squares batch solution is adopted by most IGS Analysis Centers (ACs), including Center for Orbit Determination in Europe CODE (Schaer et al. 2021) and German Research Centre for Geosciences GFZ (Deng and Schuh 2018), while filtering is implemented by some of the ACs dedicated for real-time purpose, such as the square root information filter (SRIF) at JPL and Wuhan University (Bertiger et al. 2020; Lou et al. 2022)

✉ Jungang Wang
  jgwang@gfz-potsdam.de

1 Institut Für Geodäsie und Geoinformationstechnik, Technische Universität Berlin, 10553 Berlin, Germany

2 German Research Centre for Geosciences (GFZ), 14473 Potsdam, Germany

3 Shanghai Astronomical Observatory, Chinese Academy of Sciences, Shanghai, China

4 School of Geomatics, Liaoning Technical University, Fuxin 123000, China

and Kalman filter at CNES (Laurichesse et al. 2013). For the filter-based method, the stochastic constraint is introduced in the orbit state elements update, which is formed as a linear blend of the previous estimate and the current measurement information (Laurichesse et al. 2013; Parkinson et al. 1996). Currently, the available real-time orbits are mostly predicted from the batch least-squares solutions owing to its feasibility of routine processing at most IGS ACs. However, with the extension of prediction time, the orbit accuracy drops progressively (Dai et al. 2019; Duan et al. 2019). In order to improve the orbit quality, there are two aspects to consider: one is to refine the modeling, especially the non-conservative forces, and the other one is to shorten the prediction time, i.e., shorten the orbit update latency.

IGS began providing ultra-rapid (IGU) GPS orbits in November 2000 and then reduced the update latency of ultra-rapid orbits from 12 to 6 h in April 2004 (Kouba 2009; Springer and Hugentobler 2001). As a member of the Multi-GNSS Experiment (MGEX) ACs, GFZ started to provide five GNSS system ultra-rapid products with 3-h update rate since November 2015, including GPS, Globalnaya Navigazion-naya Sputnikovaya Sistema (GLONASS), Galileo, BeiDou Navigation Satellite System (BDS), and Quasi-Zenith Satel-lite System (QZSS) (Deng et al. 2016). Wuhan University (WHU) provided hourly updated multi-GNSS orbits with an accuracy of 3 to 5 cm (Zhao et al. 2018).

With more satellites and stations involved, a huge number of parameters must be handled, especially for the undifferenced (UD) measurements. In that case, the computational time is always a burden. The simplest way to reduce the computing time is parallel processing, including the parallel processing of sub-networks, individual satellite constellations, and sub-sessions (session in this study refers to the time length of a processing job). The parallel processed tasks will be combined later to generate final orbit products. In the network parallel processing, the network is divided into several sub-networks to be processed in parallel, and the normal equations (NEQs) of sub-networks are combined by utilizing a certain number of common stations to derive a final solution (Beutler et al. 1996; Bruni et al. 2018; Pintori et al. 2021; Zurutuza et al. 2019). As the satellite and receiver clocks are eliminated during parallel processing, they cannot be combined by stacking the NEQs and thus it is not equivalent to the integrated solution. In the constellation parallel processing, each constellation is processed separately in parallel and delivered to users (Chen et al. 2021). The constellation solutions can be combined via common parameters, such as earth rotation parameters (ERP) and station coordinates, which would potentially improve the consistency and precision of orbits. However, it is still not possible to combine other processing parameters such as tropospheric parameters and receiver clocks, as they are pre-eliminated once deactivated. Another method is to split an undivided

processing into several sub-sessions and each sub-session is processed separately but in parallel to generate the sub-session NEQs. All the sub-session NEQs are stacked, with all necessary parameters combined, including both global parameters such as ERP, orbits, and station coordinates, and the processing parameters such as ambiguities covering different sub-sessions. This method is much more efficient as the computation burden can be shared by several computing nodes. Jiang et al. (2021) applied the epoch-parallel into multi-GNSS POD with double-differenced observations. However, they keep all parameters in the NEQ, including the tropospheric delays and ambiguities, which can hardly be realized in the multi-GNSS POD with undifferenced observations due to the large amounts of ambiguities and clocks. In addition, keeping the ambiguities in the NEQ leads to an enormous NEQ, slowing the speed of inversion.

In addition, special algorithms are also developed for solving a GNSS network with a huge number of stations. By keeping only active parameters in the NEQ, Ge et al. (2006) reduces the requirement of computer memory and computational burden. The optimized algorithms based on newly developed processors, i.e., block-partitioned algorithms (Gong et al. 2017; Quintana-Orti et al. 2008) and open multi-processing (openMP) (Chandra et al. 2001; Chen et al. 2022) continue improving the computation efficiency. Since most active parameters in GNSS POD are ambiguities, a carrier-range method is brought up and successfully implemented in the huge network processing with the UD ambiguity resolution (Blewitt et al. 2010; Chen et al. 2014). However, precise orbits and clocks are the prerequisite for applying this method, which cannot improve ultra-rapid POD with around 100 stations. Cui et al. (2021) also designed a parallel computing of large GNSS network, which is suitable for multi-core and multi-node environments, by decomposing the GNSS modeling tasks of each epoch to different nodes and cores. However, the efficiency was demonstrated with PPP and baseline processing of large GNSS networks instead of network solution which is essential for ultra-rapid GNSS POD. Of course, with the improvement of modern computer power like central processing unit (CPU) and solid-state drive (SSD), the efficiency in GNSS POD can be further improved (Li et al. 2018).

Currently, the processing of around 120 satellites and 100 stations based on above methods still costs nearly one hour. We aim at achieving ultra-rapid multi-GNSS batch POD solution within 30 min, without losing the consistency of sequential batch solution. Therefore, in this study, an optimized multi-GNSS POD processing strategy is proposed based on the epoch-parallel processing. Different from the method introduced by Jiang et al. (2021), our proposed strategy considers parameter elimination as soon as they are deactivated, which significantly decreases memory requirement and computation burden.

The article is organized as follows. We first present an overview of the batch POD strategy and then propose the epoch-parallel processing strategy, demonstrating its equivalence to the entire sequential batch solution in Sect. 2. The analysis of the computation time and the benefits of using 30-min updated orbit against the 1-hourly updated orbits are illustrated in Sect. 3. The summary and conclusions are presented in Sect. 4.

2 Methodology

2.1 Precise orbit determination in batch solution

Suppose that the UD measurement equation of ionosphere-free (IF) phase and pseudo-range observables from satellite to receiver is described by Eq. (1)

$$
\begin{cases}
P = \left| \boldsymbol{x}^{\mathrm{sat}}(t_s) - \mathbf{R}_x \boldsymbol{x}_{\mathrm{rec}}(t_r) \right| + c\left(\mathrm{d}t_{\mathrm{rec}} - \mathrm{d}t^{\mathrm{sat}} \right) + t_{\mathrm{trop}} + \varepsilon_P \\
L = \left| \boldsymbol{x}^{\mathrm{sat}}(t_s) - \mathbf{R}_x \boldsymbol{x}_{\mathrm{rec}}(t_r) \right| + c\left(\mathrm{d}t_{\mathrm{rec}} - \mathrm{d}t^{\mathrm{sat}} \right) + t_{\mathrm{trop}} + N_{\mathrm{rec}}^{\mathrm{sat}} + \varepsilon_L
\end{cases},
\tag{1}
$$

where P and L denote pseudo-range and phase observations, respectively; $\mathbf{x}^{sat}(t_s)$ the satellite position at the signal transmitting time in Geocentric Celestial Reference System (GCRS); $\mathbf{x}_{\mathrm{rec}}(t_r)$ the station position at the signal receiving time in International Terrestrial Reference System (ITRS) and $\mathbf{R}_x$ the transformation matrix from ITRS to GCRS; c the speed of light; dt_{rec} the receiver clock offset; dt^{sat} the satellite clock offset; t_{trop} the slant tropospheric delay; N_{rec}^{sat} the phase ambiguities; and ε_p and ε_l the measurement noise and other unmodelled errors for the pseudo-range and carrier phase observations, respectively. The inter-system/frequency-dependent code biases relative to the GPS biases at the receiver end will be considered in multi-GNSS POD. As is well known, the satellite orbit is fully determined by the initial state and necessary force model parameters after the force models are specified. The satellite position at any time can be expressed by its reference orbit and the corrections of the initial state and force model parameters as:

$$
\boldsymbol{x}^{\mathrm{sat}}(t) = \tilde{\boldsymbol{x}}^{\mathrm{sat}}(t) + \left[\dfrac{\partial \boldsymbol{x}^{\mathrm{sat}}(t)}{\partial \boldsymbol{x}^{\mathrm{sat}}(t_0)} \ \ \dfrac{\partial \boldsymbol{x}^{\mathrm{sat}}(t)}{\partial \dot{\boldsymbol{x}}^{\mathrm{sat}}(t_0)} \ \ \dfrac{\partial \boldsymbol{x}^{\mathrm{sat}}(t)}{\partial \boldsymbol{q}_0^{\mathrm{sat}}} \right] \begin{bmatrix} \delta \boldsymbol{x}^{\mathrm{sat}}(t_0) \\ \delta \dot{\boldsymbol{x}}^{\mathrm{sat}}(t_0) \\ \delta \boldsymbol{q}_0^{\mathrm{sat}} \end{bmatrix}
\tag{2}
$$

where $\tilde{\boldsymbol{x}}^{\mathrm{sat}}$ is the reference orbit at certain time t; t_0 the reference time; $\boldsymbol{x}_0^{\mathrm{sat}}(t_0)$ the satellite initial conditions; $\dot{\boldsymbol{x}}^{\mathrm{sat}}(t_0)$ the satellite initial velocities; and $\boldsymbol{q}_0^{\mathrm{sat}}$ the force parameters. Note that the satellite orbits need to be expressed in CRS, and the rotation matrix is needed for transforming the receiver position vector from terrestrial reference system (TRS) into CRS

(Petit and Luzum 2010). The ERP is therefore introduced and can be estimated.

As soon as the linearized observation equations are available, they can be contributed into the NEQ epoch-by-epoch. The sequential least-squares adjustment only keeping the active parameters in the NEQ can be found in the studies by Ge et al. (2006) and Chen et al. (2022). The eliminated parameters will be recovered backwards from the last epoch to the first epoch after the final NEQ is solved and also the observation residuals for parameter update and observation quality control.

After obtaining the float solution, integer double-differenced ambiguities are resolved and then applied as a strong constraint on the corresponding four related UD ambiguities to obtain the integer ambiguity resolution (Ge et al. 2005). Hence, the pseudo-observation equation of introducing resolved double-differenced ambiguities reads as

$$
\boldsymbol{v}_b = \mathbf{D}\boldsymbol{x} - \overline{\boldsymbol{b}}, \ \mathbf{P}_b,
\tag{3}
$$

where $\mathbf{D}$ denotes the ambiguity mapping matrix, $\overline{\boldsymbol{b}}$ the fixed IF double-ambiguities, and $\mathbf{P}_b$ the weights of pseudo-observations.

2.2 Epoch-parallel processing

Currently, the ultra-rapid POD solution is usually performed using 24-h observations, which ensures a precise and reliable solution, but can hardly be shortened due to the large number of processing parameters. Instead of sequential processing from the beginning to the end, the epoch-parallel processing divides the 24-h session into a set of sub-sessions processed parallelly. Each sub-session is processed using the sequential least squares (LSQ) discussed in Sect. 2.1. Only the active parameters are introduced to NEQ, and deactivated parameters are eliminated immediately. Within a sub-session processing, parameters which are not active in the prior or next sub-session can be eliminated before connecting sub-sessions, and the eliminating equation can be used to recover the eliminated parameters to obtain the same result as for sequential processing.

To demonstrate this, we assume all the deactivated parameters are eliminated at once and the NEQs of the ith and $(i + 1)$th sub-session are expressed as

$$
\begin{bmatrix} \mathbf{N}_{11,i} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{w}_{1,i} \\ \boldsymbol{w}_2 \end{bmatrix}
\tag{4}
$$

$$
\begin{bmatrix} \mathbf{N}_{11,i+1} & \mathbf{N}_{13} \\ \mathbf{N}_{31} & \mathbf{N}_{33} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_3 \end{bmatrix} = \begin{bmatrix} \boldsymbol{w}_{1,i+1} \\ \boldsymbol{w}_3 \end{bmatrix}
\tag{5}
$$

where $\boldsymbol{x}_1$ are the active parameters which cover at least the ith and $(i + 1)^{th}$ session, for example, satellite state parameters,

x_2 and x_3 are the inactivated parameters in $(i+1)^{th}$ and i^{th} session, respectively, for example, ambiguities and epoch-wise clocks.

For sequential processing, we eliminate x_2 from Eq. (4):

$$x_2 = \mathbf{N}_{22}^{-1}(w_2 - \mathbf{N}_{21}x_1) \tag{6}$$

$$\left(\mathbf{N}_{11,i} - \mathbf{N}_{12}\mathbf{N}_{22}^{-1}\mathbf{N}_{21}\right)x_1 = w_{1,i} - \mathbf{N}_{12,i}\mathbf{N}_{22}^{-1}w_2 \tag{7}$$

contribute the next sub-session into the NEQ:

$$\begin{bmatrix} \overline{\mathbf{N}}_{11} & \mathbf{N}_{13} \\ \mathbf{N}_{31} & \mathbf{N}_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} = \begin{bmatrix} \overline{w}_{11} \\ w_3 \end{bmatrix} \tag{8}$$

with

$$\overline{\mathbf{N}}_{11} = \mathbf{N}_{11,i} + \mathbf{N}_{11,i+1} - \mathbf{N}_{12}\mathbf{N}_{22}^{-1}\mathbf{N}_{21} \tag{9}$$

$$\overline{w}_{11} = w_{1,i} + w_{1,i+1} - \mathbf{N}_{12}\mathbf{N}_{22}^{-1}w_2 \tag{10}$$

Then, eliminate x_3 from Eq. (8):

$$x_3 = \mathbf{N}_{33}^{-1}(w_3 - \mathbf{N}_{31})x_1 \tag{11}$$

$$\left(\overline{\mathbf{N}}_{11} - \mathbf{N}_{13}\mathbf{N}_{33}^{-1}\mathbf{N}_{31}\right)x_1 = \overline{w}_{11} - \mathbf{N}_{13}\mathbf{N}_{33}^{-1}w_3 \tag{12}$$

After solving Eq. (12), the eliminated parameters can be recovered with Eqs. (6) and (11).

For the parallel processing, x_2 and x_3 are eliminated in parallel. The elimination of x_2 is expressed by Eqs. (6) and (7), while that for x_3 is similar as

$$x_3 = \mathbf{N}_{33}^{-1}(w_3 - \mathbf{N}_{31})x_1 \tag{13}$$

$$\left(\mathbf{N}_{11,i+1} - \mathbf{N}_{13}\mathbf{N}_{33}^{-1}\mathbf{N}_{31}\right)x_1 = w_{1,i+1} - \mathbf{N}_{13,i}\mathbf{N}_{33}^{-1}w_3 \tag{14}$$

Combining the NEQs of the sub-sessions, i.e., Eqs. (7) and (14) results in the same NEQ of the sequential processing, i.e., Eq. (12). Therefore, both eliminating equations and the final NEQ are the same.

To guarantee the equivalence of epoch-parallel and sequential batch processing, the parameter elimination must be carried out carefully, especially when temporal constraints are involved. First of all, in each sub-session processing, only parameters which are not used in the previous and subsequent sub-sessions can be eliminated, such as ambiguities and epoch-wise clocks. However, for the parameterization of the stochastic process, such as random-walk process (RW), temporal constraints between adjacent parameters should be imposed. These parameters, for example, tropospheric delay parameters, can only be removed after both, all related

observations and the corresponding temporal constraints, are added.

It should also be pointed out that to keep the consistency of active ambiguities in the adjacent NEQs, phase windup corrections should be prepared in advance of NEQ generation, so possible integer jumps could be corrected. The phase windup correction is accumulated along with time starting from a fractional cycle at the beginning. Therefore, there could be integer cycle differences from one sub-session to another sub-session, that prevent the connection of ambiguities of continuous data arc.

From the above discussion, epoch-parallel processing strategy divides the long session into short sub-sessions and each sub-session can be processed parallelly using sequential processing software and all the sub-session results can be combined into the final solution. It is obvious that it can take full advantage of multi-cores and multiple computers with only a minor modification of GNSS sequential processing strategies. Hence, it can be easily implemented.

2.3 POD processing strategy

The flowchart of the optimized POD strategy is shown in Fig. 1, including data preparation, data preprocessing, parameter estimation and update, ambiguity resolution, and product generation. In the data preparation, hourly observation and navigation files for specified stations are downloaded from the IGS data centers and recorded from IGS real-time streams; they are merged after preliminary quality control to session-files. The main function of data preprocessing is initializing satellite orbits, generating phase windup correction files, and initial quality control using the single station editing method referred to as *TurboEdit* (Blewitt 1990), all in a parallel way. The parameter estimation part is optimized by assigning and coordinating sub-session generation, sub-session stacking and solving, orbit update and post-fit residual-based quality control. To continue shortening the computational time, the OpenMP multithreading model is adopted for parameter elimination process during sub-session generation and sub-session stacking. It must be pointed out that the estimation part is repeated several times, typically four iterations, for data cleaning and parameter update and a final iteration for generating the integer ambiguity resolution.

In the stacking of sub-session NEQs, it is also very important to eliminate deactivated parameters in a timely manner for computational efficiency. Consequently, after any two sub-session NEQs are stacked, the parameters which become deactivated in the subsequent combination should be eliminated immediately. Abide by this rule, there are usually two available approaches, adjacent stacking and sequential stacking. The adjacent stacking approach stacks two adjacent sub-session NEQs whenever available, until all sub-session
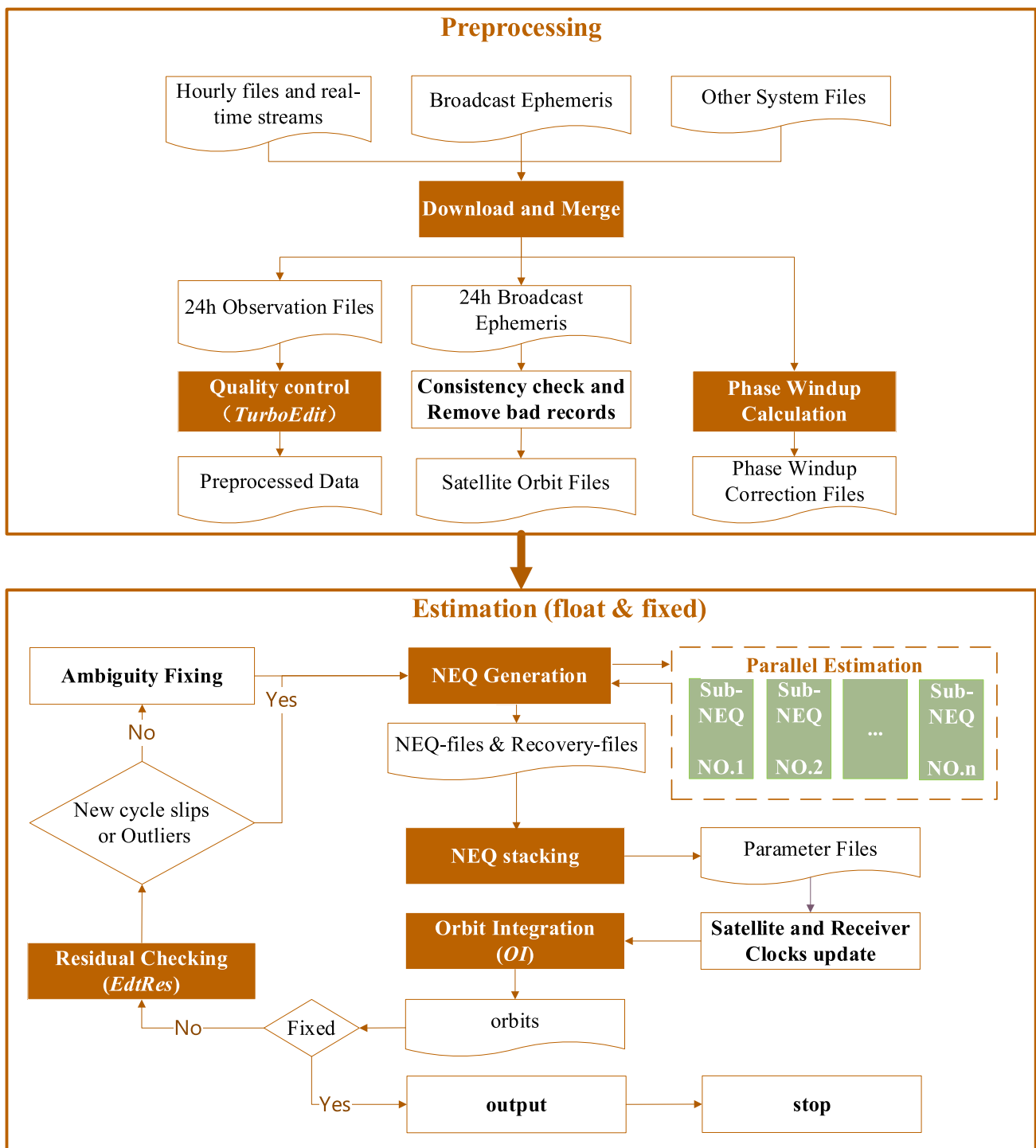
Fig. 1 The optimized processing strategy for POD. The rectangle filled with dark orange color means the processes are implemented in a parallel way. Note that the module name in the Positioning And Navigation Data Analyst (PANDA) software is in *italic* here and in the following sections, e.g., *TurboEdit, EdtRes* and *OI*
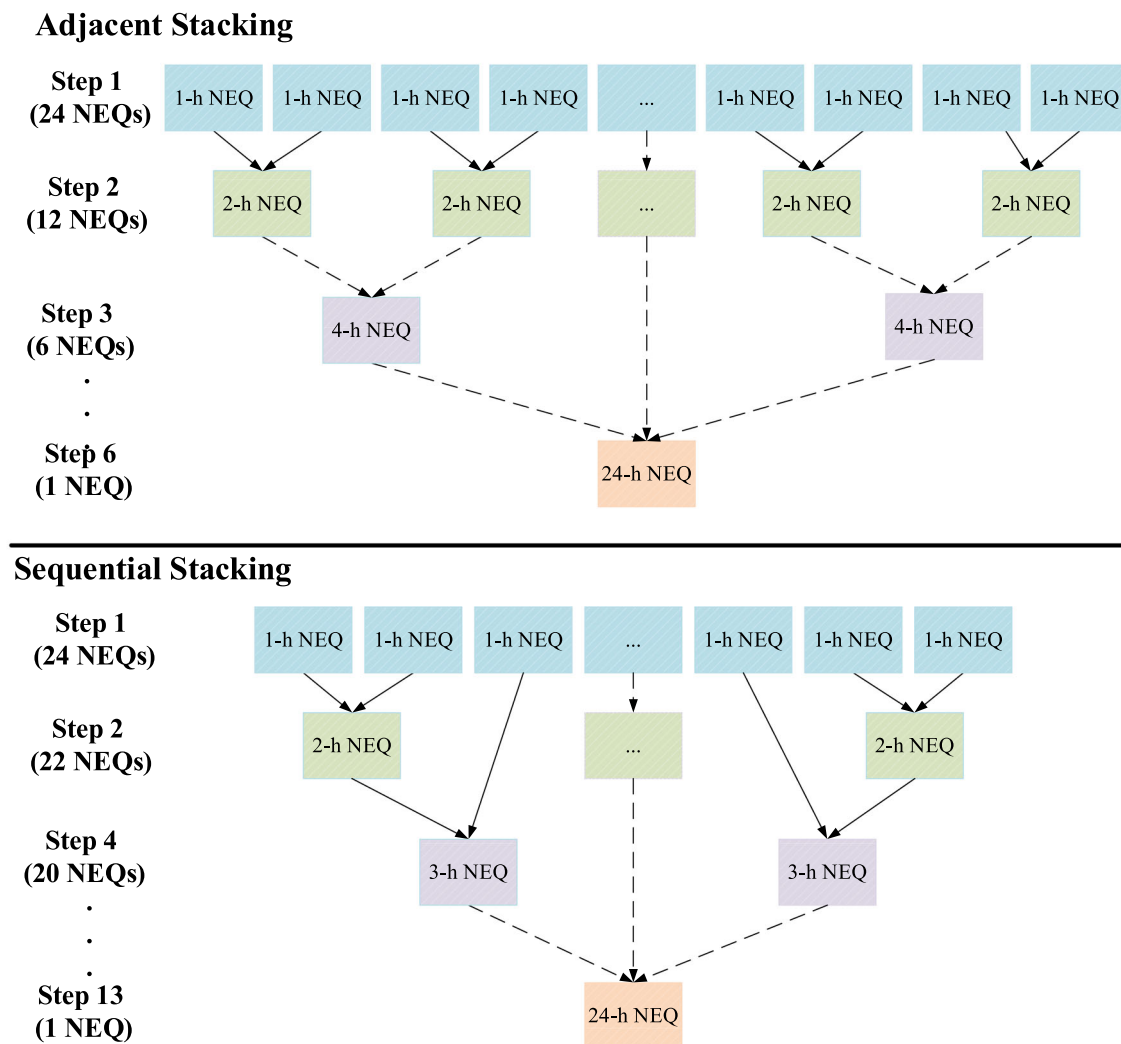
Adjacent Stacking

Step 1
(24 NEQs)
| 1-h NEQ | 1-h NEQ | 1-h NEQ | 1-h NEQ | ... | 1-h NEQ | 1-h NEQ | 1-h NEQ | 1-h NEQ |

Step 2
(12 NEQs)
2-h NEQ   2-h NEQ   ...   2-h NEQ   2-h NEQ

Step 3
(6 NEQs)
4-h NEQ   4-h NEQ

Step 6
(1 NEQ)
24-h NEQ

Sequential Stacking

Step 1
(24 NEQs)
| 1-h NEQ | 1-h NEQ | 1-h NEQ | ... | 1-h NEQ | 1-h NEQ | 1-h NEQ |

Step 2
(22 NEQs)
2-h NEQ   ...   2-h NEQ

Step 4
(20 NEQs)
3-h NEQ   3-h NEQ

Step 13
(1 NEQ)
24-h NEQ

Fig. 2 Different ways of NEQ stacking. h stands for hour. 1-h NEQ is generated by sequential POD of a sub-session

NEQs are combined. In contrast, the sequential stacking approach stacks the sub-session NEQs one by one and it can be optimized by two parallel stacking processing starting from both ends toward the middle. Taking the stacking of 24 1-h NEQs generated by each sub-session as an example, Fig. 2 illustrates the two different stacking approaches. Besides the 1-h NEQ generation step (step 1), the adjacent stacking approach requires another five steps to obtain the final 24-h NEQ, while the sequential stacking approach needs 12 steps. For the adjacent stacking approach, there are 12 2-h NEQs needed to be generated in step 2, then six 4-h NEQs in step 3, three 8-h NEQs in step 4, one 16-h NEQ and 8-h NEQ in step 5, and a 24-h NEQ in the last step. Assuming that there are sufficient threads available and the parallel combinations take the same time of a single combination, the adjacent stacking should take less time than the sequential stacking at the first glance. However, it is in fact opposite and

the computational efficiency of two approaches is evaluated in Sect. 3.2.

For this processing, multiple computer nodes should be included, among them one serves as the master or coordinator to distribute tasks, monitor processing status (start and complete, or any disrupt) and check results. The parallel tasks are distributed to the available nodes along with all necessary files for computation. The number of involved nodes is determined by how many sub-session NEQs are divided, for example, 12 nodes are required for epoch-parallel processing of 12 sub-session NEQs. On each node, the proposed strategy can also take advantage of multi-threads, for example, accelerating parameter elimination by OpenMP.

3 Results

The epoch-parallel processing strategy is evaluated in this section. We first introduce the data processing strategy for the experiment and the involved hardware configuration. Subsequently, the feasibility and timeliness of the proposed POD strategy on a cluster are verified in Sect. 3.2 and the benefits of the 30-min updated orbits are presented in Sect. 3.3.

3.1 Data processing strategies

Table 1 shows the CPU types equipped in computer nodes used in this study. The CPU frequency of each node ranges from 3.20 to 4.10 GHz. For a convenient and clear statement, we need to specify that one node here means a single server or computer. All nodes are equipped with SSDs for high-speed data input–output (IO).

A network of 120 multi-GNSS ground stations are selected for multi-GNSS POD solutions, as shown in Fig. 3. In addition, we select a subnet using 90, 100, or 110 stations within the 120 stations, to further evaluate the POD efficiency and accuracy of different networks. The time span of our experiments ranges from day of year (DOY) 335 to 365, 2021.

The proposed POD strategy in Sect. 2.2 is implemented on the Positioning And Navigation Data Analyst (PANDA) software package (Liu and Ge 2003), which is capable of multi-GNSS post and real-time processing (Zuo et al. 2021) and recently upgraded to handle multiple space geodetic techniques (Wang et al. 2022). Table 2 gives the processing details, which generally follow the IGS standard. For the solar radiation pressure, a hybrid estimation model with the extended code orbit model (ECOM), which originated from the model proposed by Beutler et al. (1994), and a prior box-wing model is applied. The 5-parameter ECOM is termed as ECOM1 (Springer et al. 1999), while the 7-parameter ECOM is termed as ECOM2 (Arnold et al. 2015). We do not apply integer ambiguity resolution to the GLONASS or BDS GEO satellites. Considering the CPU information, the number of threads for parallel parameter elimination in a node is set to four. In the following experiments, the session length is 24 h, similar to IGS ACs, even though a longer session might bring marginal improvement. The entire session is divided into at most 24 sub-sessions, as the processing of 1-h observations only takes a few seconds. Unless otherwise noted, the experiments in this section are carried out using 100 tracking stations and 120 satellites on the 4.1 GHz node.

3.2 The timeliness of multi-GNSS POD on multiple-nodes

As the length of a sub-session becomes shorter, more NEQs are generated and the NEQ stacking tends to be more time-consuming. But if a sub-session is too long, the benefit of epoch-parallel is less significant. Therefore, the balance between generating and stacking of sub-sessions needs to be further investigated. Note that the number of nodes mentioned in the following equals the number of sub-sessions as there is only one process of sub-session NEQ generation running on a node, e.g., the number of four nodes means that a 24-h session is split into four 6-h sub-sessions.

Taking into consideration all the optimized strategies mentioned in Sect. 2.3, along with parallel parameter elimination utilizing OpenMP, the computational efficiency on different numbers of sub-sessions is shown in Fig. 4a. With the assistance of parallel processing of *TurboEdit*, *OI*, and *EdtRes*, the sequential batch POD strategy with and without parallel parameter elimination costs 54 min and 73 min, respectively. Compared with sequential batch solutions, all epoch-parallel cases have the priority in computational efficiency. When there are more than six nodes (six sub-sessions) available, the computational time of the new POD processing strategy is less than 30 min, while the improvement of using 12 nodes reaches up to 49% compared to using one node. The increased time using 24 nodes is caused by the stacking of too many sub-sessions coupled with many active ambiguities. Even for the two nodes case, the computational efficiency is improved by 25%. Obviously, the most time-consuming part in a POD is the process of parameter estimation. Figure 4b illustrates the computational time of one iteration of parameter estimation, including NEQ generation and NEQ stacking, using different numbers of nodes. If four or more nodes are used, then one iteration of parameter estimation takes only around 5 min and the difference caused by more nodes is less than 1 min. The computational efficiency on two nodes is reduced by more than 2 min compared with the sequential POD method.

When the number of sub-sessions increases, the time-consumption of sub-session stacking becomes more prominent. As aforesaid, there are mainly two ways of stacking NEQs: sequential stacking and adjacent stacking. Their time-consumings are presented in Fig. 5. The step of stacking two sub-sessions only costs around 17 s. When there are more than two sub-sessions, the stacking time increases sharply. In case of six or more sub-sessions, sequential stacking is more efficient, e.g., 67 s is saved in the situation of stacking 24 sub-sessions, even though the sequential stacking requires more stacking steps. The reason is that in the sequential stacking only the active ambiguities of one side (either starting or ending part of the sub-session) are kept in NEQ, while in adjacent stacking, many parameters which are also active in

Table 1 CPU architecture information

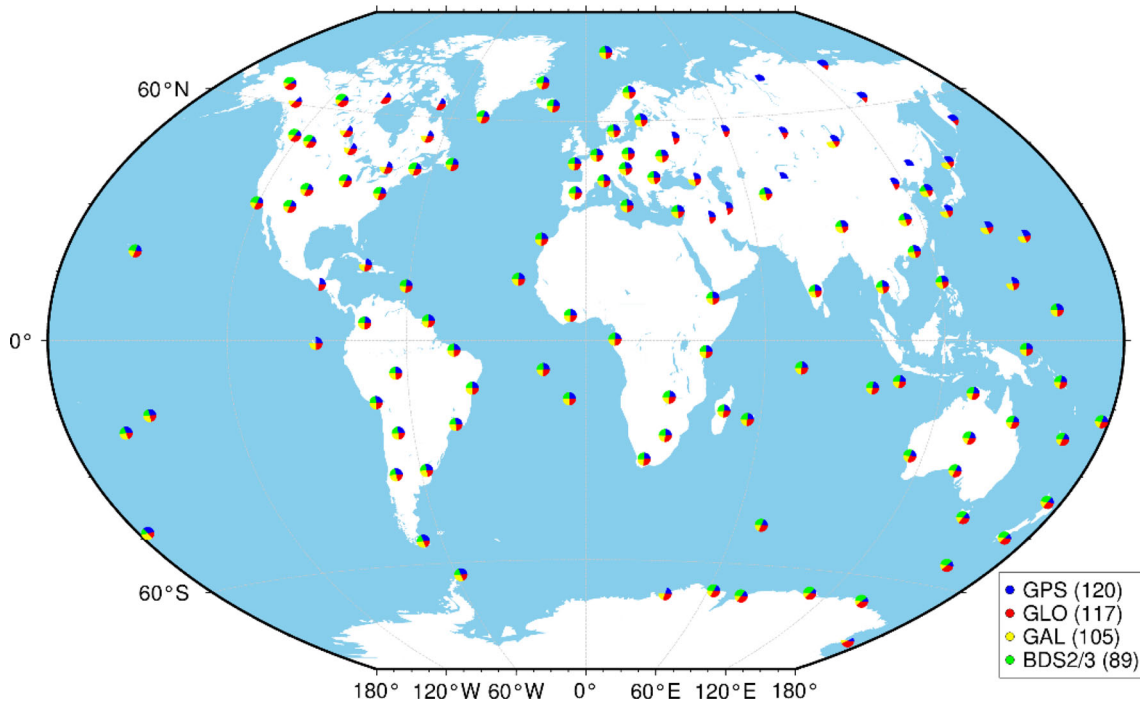| Node name	3.20 GHz	3.60 GHz	3.80 GHz	4.10 GHz
CPU type	Intel(R) Xeon(R) CPU E3-1271 v3 @ 3.20 GHz	Intel(R) Xeon(R) CPU E3-1271 v3 @ 3.60 GHz	Intel(R) Xeon(R) CPU E3-1275 v6 @ 3.80 GHz	Intel(R) Xeon(R) CPU E3-1285 v6 @ 4.10 GHz

Fig. 3 Station distribution of the experimental GNSS tracking network with 120 stations

Table 2 Strategy and parameters of POD processing

Item	Settings
Observation	Ionosphere-free combined GNSS pseudo-range and phase observations, 5-min sampling
Observation weighting	Pseudorange: 0.5 m, phase: 0.01 cycle; elevation-dependent downweighting
Cut-off elevation angle	7 degrees
Solar radiation pressure modeling	GPS: ECOM1 + box-wing (Duan and Hugentobler 2021; Tang et al. 2021) GLONASS: ECOM1 + box-wing (Duan et al. 2020) Galileo: ECOM1 + box-wing (GSA 2017) BDS: ECOM2 + Empirical model (Guo et al. 2017; Wang et al. 2018)
Receiver and satellite clock	Estimated as epoch-wise white noise
Earth radiation pressure	Applied (Rodriguez-Solano et al. 2012)
Satellite transmitter thrust	Applied (Steigenberger et al. 2017)
Station coordinates	Estimated as daily constant, no-net-rotation constraints on the datum stations to IGS14 (Rebischung and Schmid 2016)

Table 2 (continued)

Item	Settings
Surface displacement	IERS 2010 Conventions (Petit and Luzum 2010)
Tropospheric delay	GPT3 (Böhm et al. 2014) for the a priori zenith delay, residual zenith wet delay estimated as 2-h piece wise constant, north and east gradients estimated as daily constant Mapping functions: GMF (Boehm et al. 2006) for zenith delays and Chen-Herring (Chen and Herring 1997) for horizontal gradients
Ambiguity resolution	Double-differenced ambiguity resolution (Ge et al. 2005, 2006)
Earth rotation parameters	A priori value: IERS finals 2000A product Polar motion components estimated as daily offset and rate, and only daily rate (i.e., Length of Day, LoD) for UT1-UTC. The sub-daily variations of ERP are modeled according to IERS 2010 Conventions
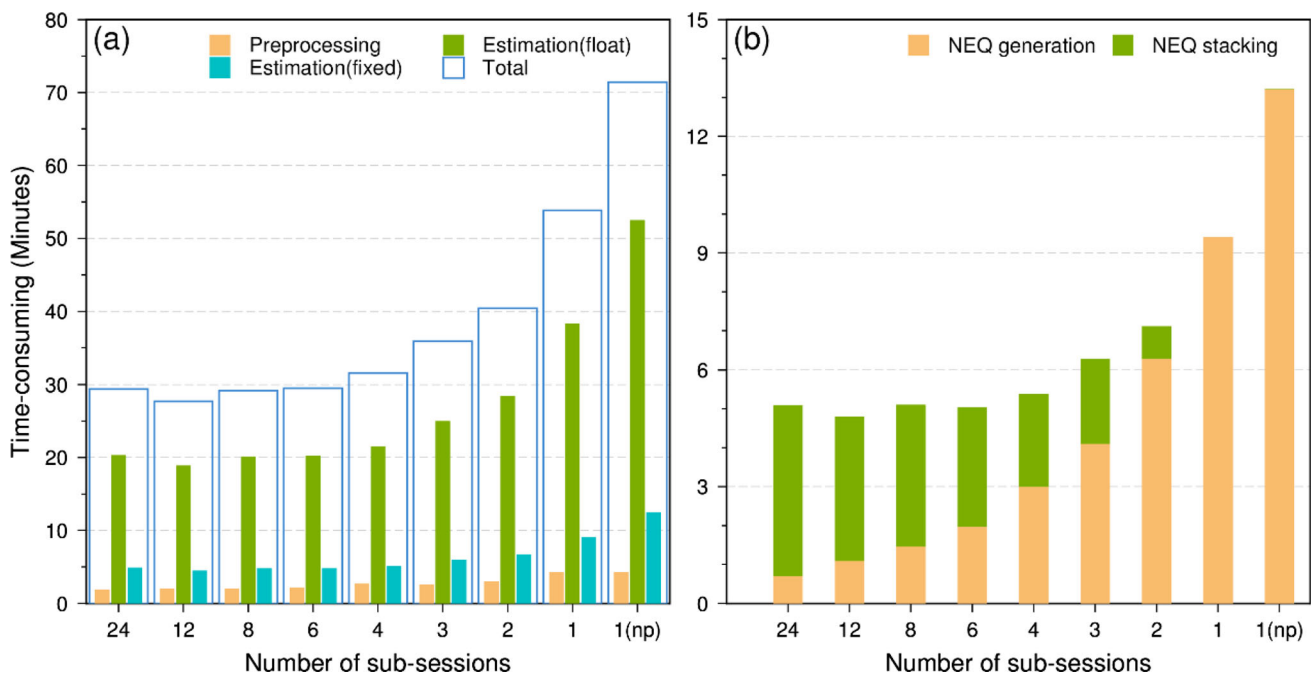
Fig. 4 Computational efficiency of POD with various numbers of nodes over the period from December 1 to 7, 2021, using 100 stations on a 4.1 GHz node. **a** The running time of an entire POD (total), where "Pre-processing" stands for processing steps before parameter estimation, "Estimation(float)" for four iterations of parameter estimation without ambiguity resolution; "Estimation(fixed)" for one time of parameter estimation with ambiguity resolution. **b** The running time of one iteration of parameter estimation including the generation of NEQs (yellow) and NEQ stacking (green). Sequential stacking method is selected here. The number of the sub-sessions is identical with the number of nodes, so that each node processes only one session for NEQ generation. Note that "1(np)" stands for using one node but non-parallel parameter elimination

the adjacent two sub-session NEQs must be kept for further stacking, leading to a larger dimension of the NEQ and more time-consuming in parameter elimination. When the number of sub-sessions is no more than four, the time-consuming of two methods are the same as they perform identically. Although both three sub-sessions and four sub-sessions need two times stacking, the latter costs much more time, as the additional third and fourth NEQ stacking spends more time than the first and second NEQ stacking. Therefore, sequential stacking is adopted in the following POD results. It is worth mentioning that the sequential stacking can be easily performed on a single node while the benefit of using two nodes is marginal.

Apart from the NEQ generation and stacking, multi-nodes are profitable for the other steps, e.g., preprocessing of GNSS observations (*TurboEdit*), orbit integration (*OI*), and posterior residual-based quality control (*EdtRes*), which can be
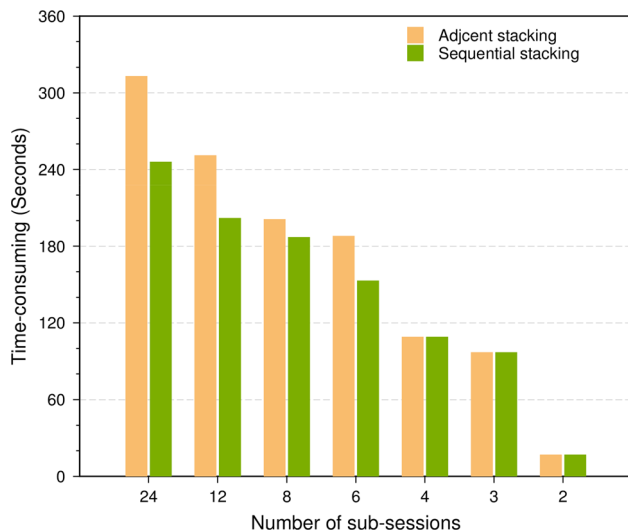
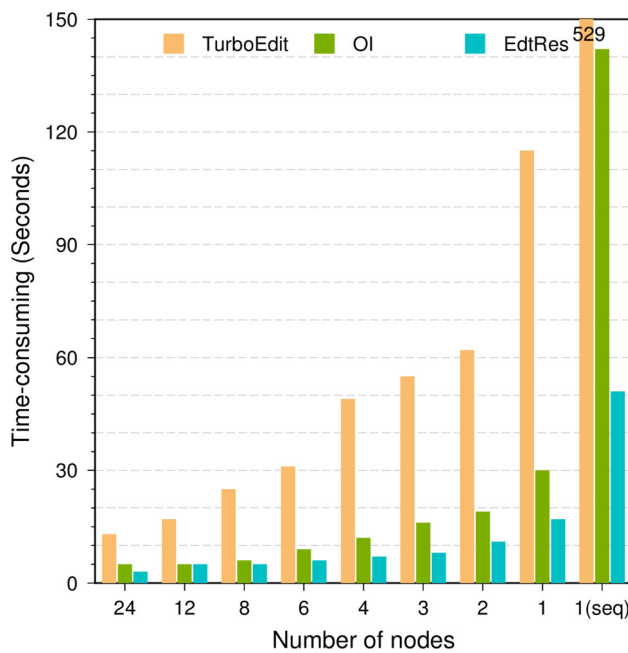**Fig. 5** Computational efficiency for different stacking methods

Fig. 6 Computational efficiency of *TurboEdit*, *OI*, and *EdtRes* with various nodes. "1(seq)" in the horizontal axis stands for sequential processing without station or satellite parallel

easily realized station-wise and/or satellite-wise. As shown in Fig. 6, the computational time of *TurboEdit*, *OI,* and *EdtRes* decreases gradually along with the increasing of the node number. When three or more nodes are involved, *TurboEdit* of 100 stations requires less than 1 min. However, with more nodes involved, the improvement of computational efficiency for *OI* and *EdtRes* decreases progressively as each program needs a little time to process single station or satellite, e.g., files reading which cannot be omitted. Provided that there are 100 stations and 120 satellites, four nodes equipped with

eight threads processing at most four times of *OI* and *EdtRes,* costs around merely 10 s. The improvement of the available nodes more than four is less than 5 s.

The above comparison and analysis are derived from processing 100 tracking stations based on the 4.1 GHZ CPU nodes. The computational efficiency of the new strategy using different types of CPU is further shown in Fig. 7. The process of POD is usually faster with higher CPU performance. Compared with the results based on nodes equipped with 3.2 GHZ CPU, the time-consuming on the 4.1 GHz CPU of three solutions, including four, two, and one sub-sessions is down by 37%, 33%, and 29%, respectively. Consistent with the above conclusion, using four nodes is always faster than using two, especially for larger number of stations. In general, the new strategy can secure a processing time within one hour except for few special cases with too many stations and/or outdated CPU such as the case of processing 120 stations on 3.2 GHz node, whereas the sequential strategy has to reduce the number of stations and on a high-performance node to satisfy the 1-h requirement, such as the case using 90 stations, and that using 100 stations on 4.1 GHz node. More important is that the half-hourly update can be achieved using the new strategy, for instance, using 90 stations on a 3.8 GHz or 4.1 GHz node.

3.3 Assessment of multi-GNSS orbits

According to the timeliness analysis in Sect. 3.2, the epoch-parallel strategy with around 90 stations and 120 satellites can realize half-hourly orbit update, while for the sequential batch processing the computation costs nearly 50 min. Currently, most IGS ACs select around 100 stations to realize the routine ultra-rapid orbit solution with a latency between one and three hours, whereas our proposed strategy can increase the number of tracking stations to 120 to achieve hourly orbits. Therefore, four solutions over a period of one month in December, 2021 are designed to evaluate the performance of epoch-parallel processing strategy, as shown in Table 3. The EP_090_030 and EP_120_060 solutions are based on the epoch-parallel strategy, updated in 30 and 60 min, respectively. The other two solutions based on the sequential batch strategy serve as a reference. Note that the sequential batch and epoch-parallel strategy are strictly equivalent, with the only difference being in processing time. Hence, the major difference between SB_090_060 and EP_120_060 is the number of stations included in processing. The SB_090_060 solution indicates that 90 stations can be processed in one hour, while 120 stations can be processed at the same time in EP_120_060 solution.

For real-time positioning purpose, only the predicted orbits are available for users, therefore it is of great importance. However, here both estimated and predicted orbits are
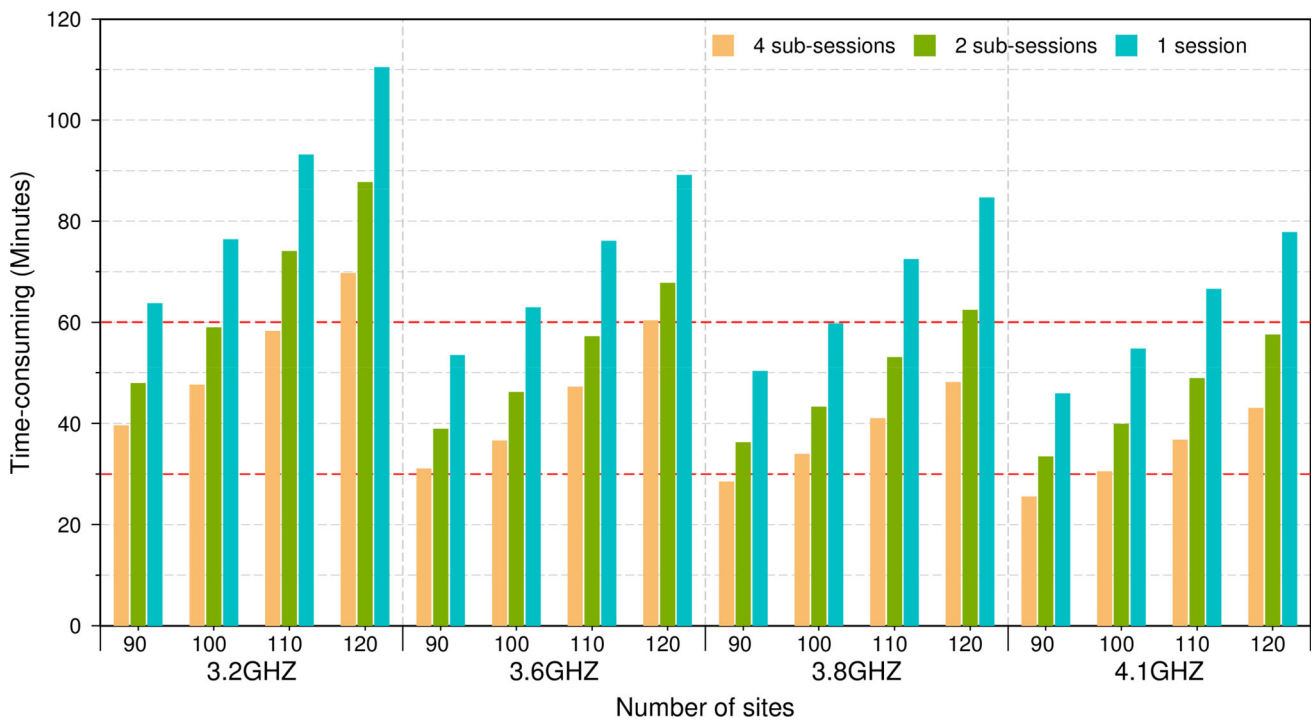
Fig. 7 Computational efficiency with different number of nodes and stations over the period from December 1 to 7, 2021. More information of the different types of nodes is provided in Table 1. One session means a sequential batch POD solution

Table 3 Four cases of ultra-rapid POD using different POD strategies

Solutions	POD strategy	Number of stations	Update time (min)
EP_090_030	Epoch-parallel	90	30
SB_090_060	Sequential batch	90	60
EP_120_060	Epoch-parallel	120	60
SB_120_120	Sequential batch	120	120

evaluated. Figure 8 shows the processing and update scenarios of ultra-rapid orbits.

Let the processing of session one start at t_1, the session time of estimated part is $(t_1 - 24^h, t_1)$, the processing period is Δt_c, then the orbits are available from $t_1 + \Delta t_u$. Δt_u could be different from Δt_c as it is determined by product provider. Afterward, the session two starts at $t_1 + \Delta t_u$, in which Δt_u is the orbit update interval, the orbits are available from $t_1 + 2\Delta t_u$. From this epoch, the predicted orbits from session one will be replaced by that of session two. The inconsistency at switch epoch of the two consecutive predicted orbits are the boundary discontinuities, that are investigated. The current used predicted orbits can also be compared with the later
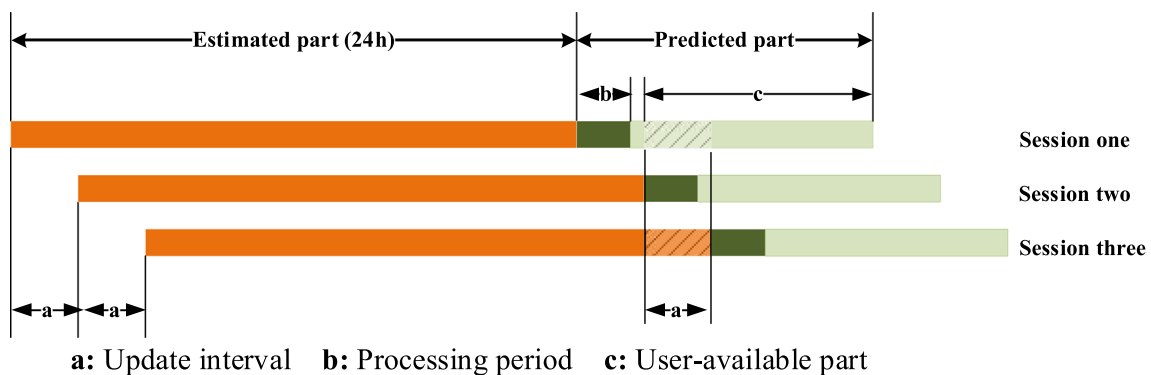

a: Update interval   **b:** Processing period   **c:** User-available part

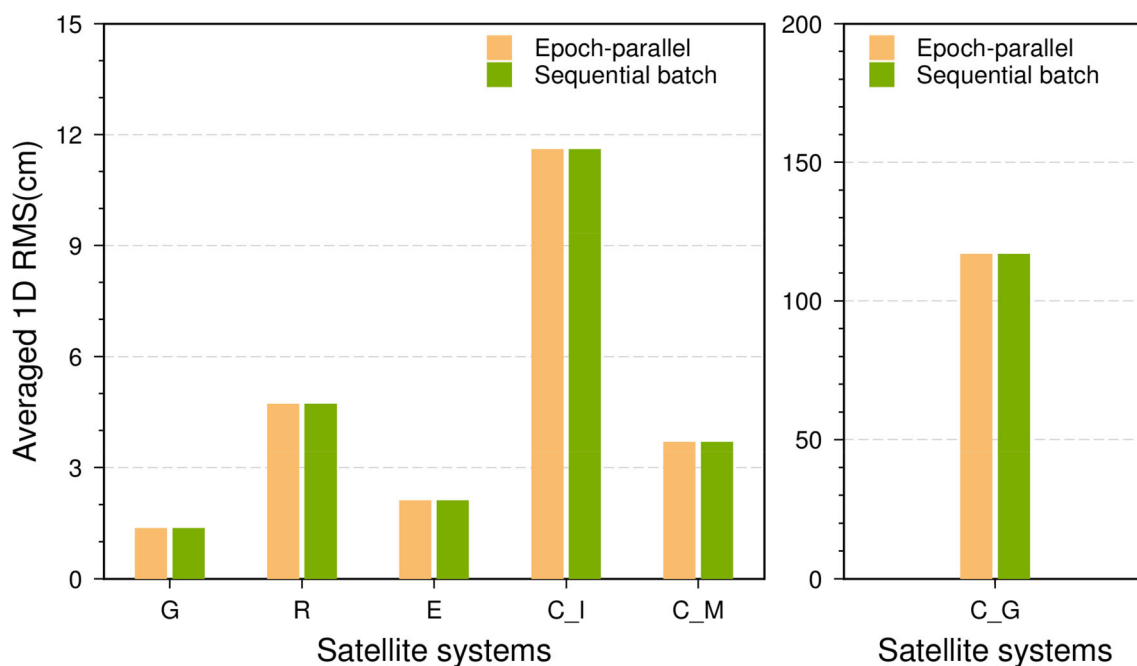**Fig. 8** Update latency for ultra-rapid orbits

Fig. 9 RMS values of estimated part for different solutions. GPS orbits are compared with the IGS Final products, while the other constellations are compared with the GBM Rapid products. Note the different y-axis scales between different panels
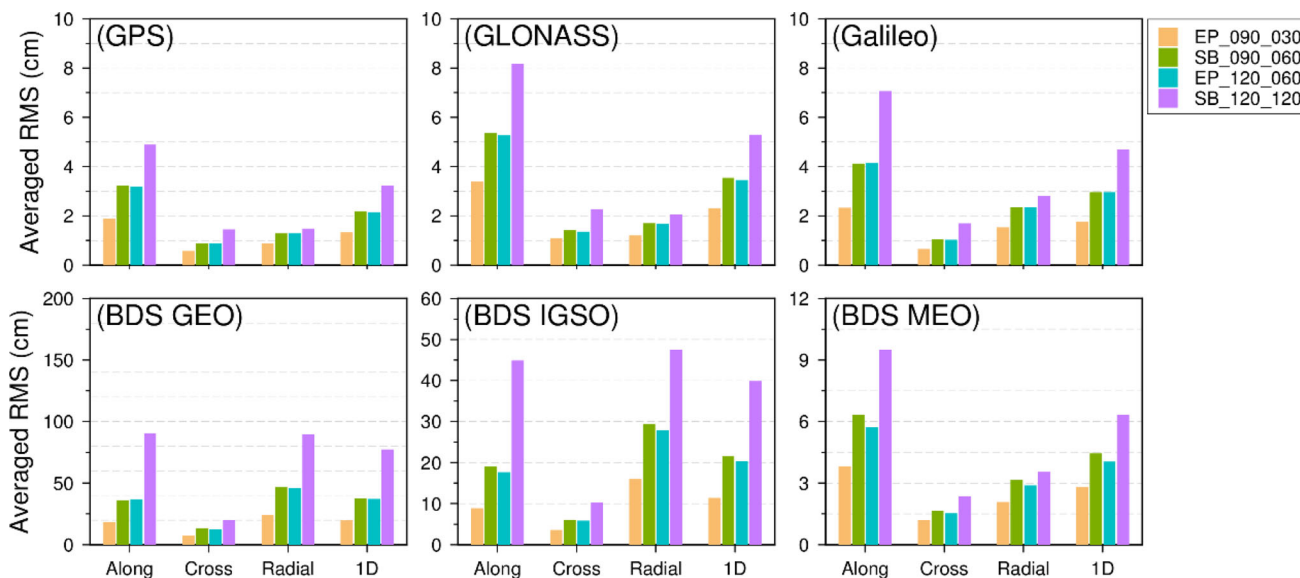

Fig. 10 RMS values of orbit overlaps for different solutions. Note the different y-axis scales between different panels

available estimated orbits for quality validation, i.e., the predicted part of session one with the estimated part of session three, termed as orbit overlap. Compared with session three, orbit overlap in session one starts from $t_1 + \Delta t_u$ to $t_1 + 2\Delta t_u$, which is represented as shade part in Fig. 8.

The estimated orbits are first evaluated and compared with the IGS Final products and Rapid products provided by GFZ, known as GBM, is shown in Fig. 9 and Table 4. The orbit accuracy for GPS, GLONASS, Galileo, and BDS MEO satellites in terms of averaged 1D RMS is 1.4, 4.7, 2.2, and

3.7 cm, respectively, while for BDS GEO and IGSO satellites the value is 117 cm and 11.6 cm, respectively. Due to the uneven tracking stations and insufficient force models, e.g., solar radiation pressure, the performance of BDS is inferior to that of GPS and Galileo, especially for BDS GEO and IGSO satellites (Zhao et al. 2022). As expected, there is almost no difference between the epoch-parallel and traditional batch solutions.

The averaged RMS values of orbit overlaps are provided in Fig. 10 and Table 5. The averaged 1D RMS values in
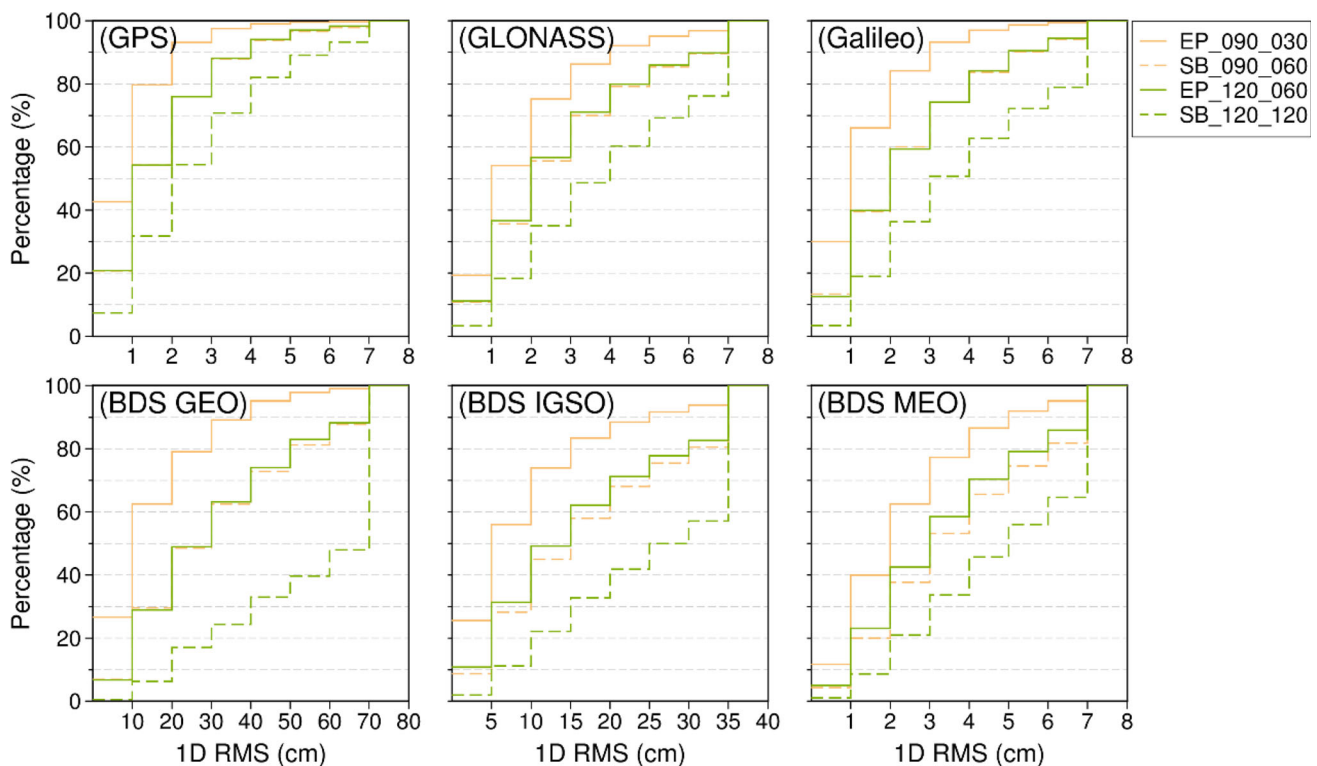
Fig. 11 Distribution of the 1D RMS values of orbit overlaps for different solutions. Note that the dashed orange lines are covered by the green line in most subplots, except for the last plot in the bottom-right corner. The x-axis scales of MEO and GEO/IGSO satellites are different in the lower panel

the EP_090_030 solution for the GPS, GLONASS, Galileo, BDS GEO, BDS IGSO, and BDS MEO satellites are 1.3, 2.3, 1.8, 19.7, 11.4, and 2.8 cm, respectively, which is better than the SB_090_060 solution by 39%, 35%, 41%, 47%, 47%, and 37%. After the number of tracking stations increases to 120, although a slight improvement is observed with respect to the SB_090_060 solution, in which BDS IGSO and MEO satellites show the largest improvement, increased by 6% and 9%, respectively, adding more stations to the network with 90 stations is not critical. When the multi-GNSS orbits are updated every two hours, a significant accuracy reduction is observed. Compared with the EP_120_060 solution, the averaged 1D RMS values in the SB_120_120 solution for GPS, GLONASS Galileo, BDS GEO, BDS IGSO, and BDS MEO satellites are decreased by 33%, 35%, 37%, 52%, 49%, and 36%, respectively. For MEO satellites, the along component exhibits the worst accuracy among the three directions and drops rapidly as the update latency becomes longer. In addition, the radial component for BDS GEO and IGSO satellites also shows a notable deterioration from the 30-min updated solution to the 60-min and 120-min updated ones, decreased by 73% and 66%, respectively.

Figure 11 shows the distribution of 1D RMS values of orbit overlaps for different constellations. For solution EP_090_030, around 80%, 54%, 67%, and 40% of the overlap RMS values are within 2 cm for GPS, GLONASS,

Galileo, and BDS MEO, respectively, which is better than the SB_090_060 solution by 26%, 19%, 27%, and 20%, respectively. Similar to the orbit results compared with GBM, the BDS MEO satellites show relatively poor performance. Despite the overall poor overlaps of BDS GEO and IGSO satellites, 79% and 89% of the overlap RMS values are within 20 cm in the EP_090_030 solution, respectively, whereas the corresponding value in the SB_090_060 solution is 49% and 68%. Comparing the SB_090_060 and EP_120_060 solutions, the station number does not contribute significantly to the orbit overlap for GPS and Galileo constellations, whereas for GLONASS and BDS satellites a slight improvement of 2% is observed. When the orbit prediction time is extended to two hours, i.e., solution SB_120_120, the overlap RMS values are much worse, and only 32%, 18%, 20%, and 9% of the values are within 2 cm for GPS, GLONASS, Galileo, and BDS MEO satellites, respectively. The four solutions demonstrate the necessity and benefits of using low-latency ultra-rapid orbits.

The positioning performance could be affected by the update of the GNSS ultra-rapid orbits, i.e., switching from one to the next orbits inevitably leads to residual fluctuations in positioning solutions due to the discontinuities of two consecutive sessions. Hence, orbit discontinuity is a key indicator for evaluating orbit quality. Figure 12 and Table 6 show the averaged 1D RMS values of orbit
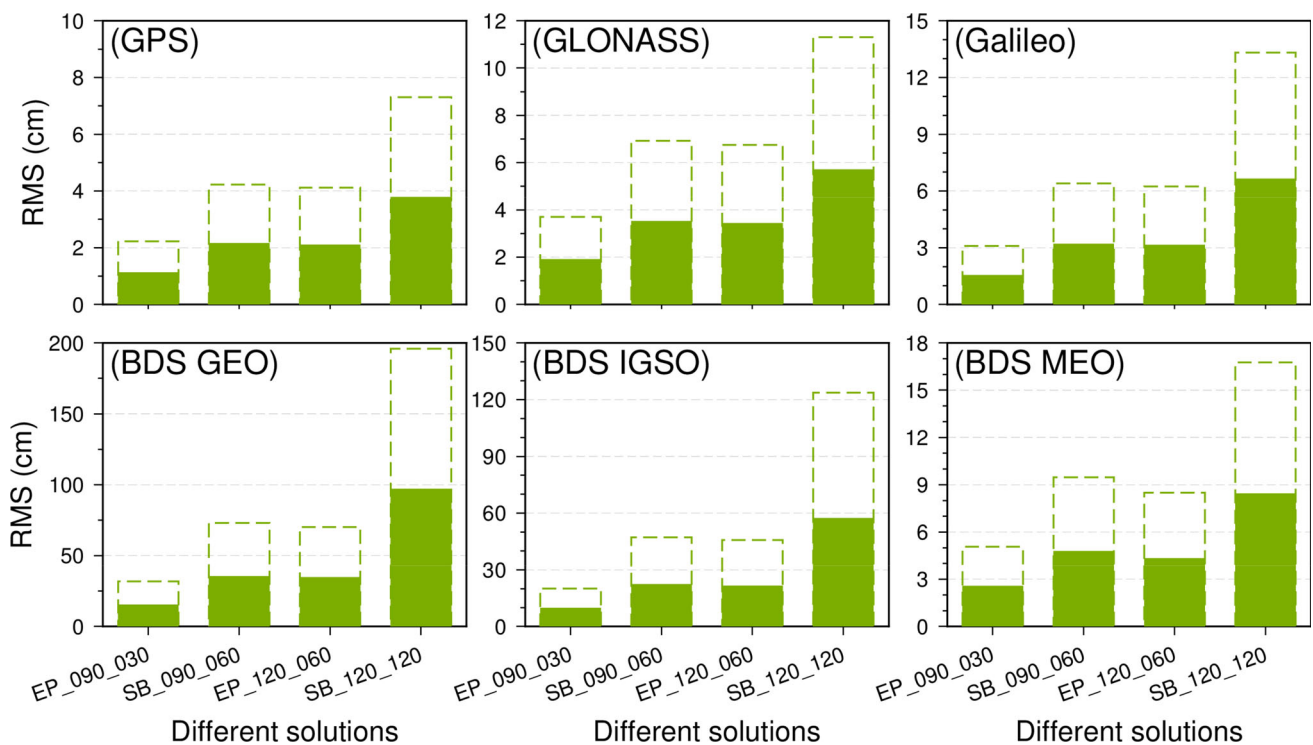
Fig. 12 Averaged 1D RMS (solid bar) values of orbit discontinuities for different solutions. Dashed box stands for the 95th percentile. Note the different y-axis scales in different panels

discontinuities. By comparing SB_090_060 and EP_120_060 solutions, increasing the number of stations from 90 to 120 only has small improvements for the discontinuities of BDS satellites. For the satellite orbits updated per 30 min, i.e., solution EP_090_030, the averaged 1D RMS values for GPS, GLONASS, Galileo, BDS GEO, BDS IGSO, and BDS MEO satellites are 1.1, 1.9, 1.6, 15.5, 9.8, and 5.6 cm, respectively. When the orbit update is extended to one hour, the averaged RMS values for all satellites decrease by a factor of 1.8 to 2.3. The accuracy of orbits updated every two hours, i.e., solution SB_120_120 is even three to six times worse than the orbits update in 30 min, especially for BDS GEO and IGSO satellites, decreased by a factor of around five. In terms of the 95th percentile, the orbit discontinuities in EP_090_030 solution for GPS, GLONASS, Galileo, BDS GEO, BDS IGSO, and BDS MEO satellites are 2.2, 3.7, 3.1, 31.9, 20.1, and 5.1 cm, respectively, which show an obvious drop if the update latency is lengthened, from 30 min to 60 min and from 60 min to 120 min. Both the orbit overlap and the discontinuity show that satellite orbits with half-hourly update perform best among the four solutions.

Taking the IGS Final products and GBM products, the statistics of user-available part of predicted orbits are shown in Fig. 13 and Table 7. The orbits in EP_090_030 solution show the best performance among four solutions. The average 1D RMS values are 2.7 cm, 6.5 cm, 3.7 cm, 129.2 cm, 26.7 cm, and 6.5 cm for GPS, GLONASS,

Galileo, BDS GEO, BDS IGSO, and BDS MEO satellites, respectively. When the orbit update interval gets longer, the accuracy of predicted orbits decreases gradually, especially in the along component. Compared to the EP_090_030 solution, the along component of Galileo and BDS IGSO in the SB_120_120 solution is decreased by a factor of two and three, respectively. The orbit difference between the SB_090_060 and EP_120_060 solution is negligible, less than 1%. Meanwhile, the accuracy decrease in the other two components is slightly smaller, except for the radial component of BDS GEO and IGSO satellites.

4 Conclusions and outlook

In this contribution, an epoch-parallel processing strategy for multi-GNSS observations is proposed, which is suitable for low-latency multi-GNSS POD. The proposed method is rigorously equivalent to the sequential batch processing strategy. Compared with the sequential batch processing, the overall computation time is improved by 25% to 49% with more than two nodes utilized. However, the benefits of using more nodes are less significant if more than four nodes are used. With optimized epoch-parallel POD strategy, the ultra-rapid orbits can be updated within 30 min for a global network consisting of 90 tracking stations and one hour for a global network consisting of 120 tracking stations.
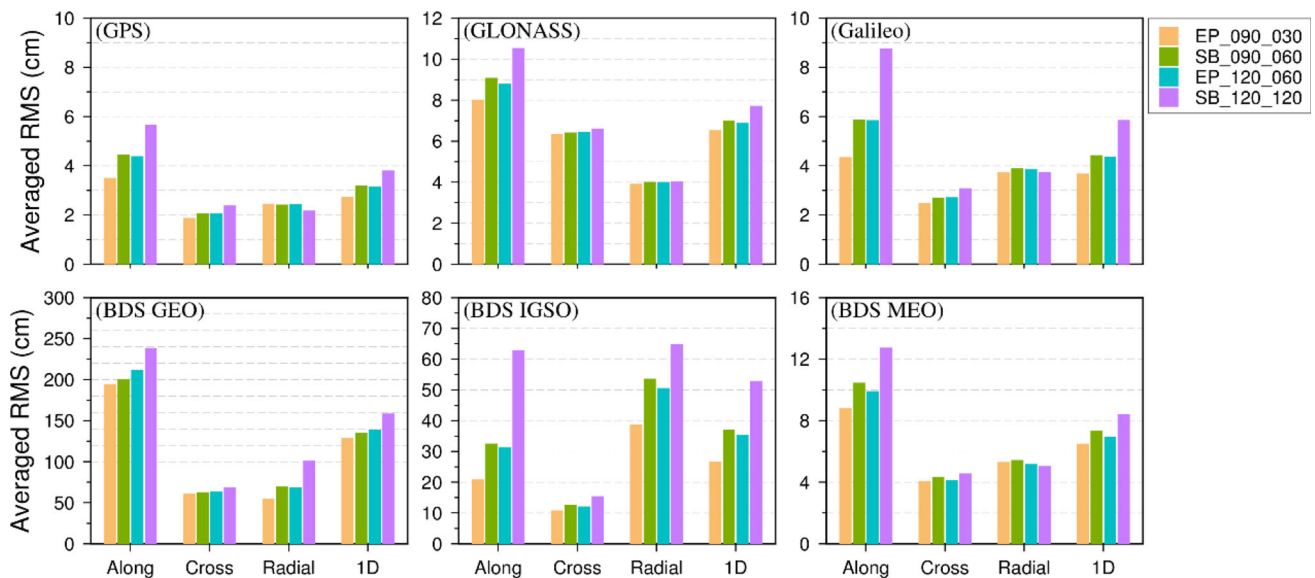
Fig. 13 RMS values of user-available part for different solutions. GPS orbits are compared with the IGS Final products, while the other constellations are compared with the GBM Rapid products. Note the different y-axis scales between different panels

In addition to the orbit computational efficiency, the orbit accuracy of user-available part is improved visibly as well. The 30-min updated orbit is better than the 60-min updated ones by half of the corresponding RMS, and the 120-min updated ones with only a quarter of the RMS value. The averaged 1D RMS of orbit overlaps of the 30-min orbits for the GPS, GLONASS, Galileo, BDS GEO, BDS IGSO, and BDS MEO satellites is 1.3, 2.3, 1.8, 19.7, 11.4, and 2.8 cm, respectively, and the corresponding values when compared with GBM products is 1.9, 6.2, 2.2, 144, 11.4, and 3.7 cm. Increasing the number of tracking stations only improves slightly to BDS satellites.

In summary, our results show that the proposed strategy can shorten orbit update time effectively without destroying the consistency of all estimated parameters. Hence, the accuracy of predicted orbit can be improved. In future, we will further optimize the POD strategies and explore the benefits of half-hourly updated orbits in real-time clock estimation, precise point positioning (Tang et al. 2023), and atmospheric sounding.

Appendix

See Tables 4, 5, 6, and 7.

Table 4 Averaged 1D RMS of the orbit differences of the epoch-parallel and batch processing strategy with respect to the GFZ Rapid products (GBM)

Type	Along	Cross	Radial	1D
G	1.4	1.1	1.5	1.4
R	3.9	5.9	3.2	4.7
E	2.0	1.6	2.4	2.1
C_I	14.8	7.8	8.5	11.6
C_M	3.7	3.0	3.8	3.7
C_G	186.6	54.4	16.0	117.0

The unit is cm

Table 5 RMS values of orbit overlaps for different solutions

	Type	EP_090_030	SB_090_060	EP_120_060	SB_120_120
G	Along	1.9	3.2	3.2	4.9
	Cross	0.6	0.9	0.9	1.4
	Radial	0.9	1.3	1.3	1.5
	1D	1.3	2.2	2.2	3.2
R	Along	3.4	5.4	5.3	8.2
	Cross	1.1	1.4	1.4	2.3
	Radial	1.2	1.7	1.7	2.1
	1D	2.3	3.5	3.5	5.3
E	Along	2.3	4.1	4.1	7.1
	Cross	0.7	1.0	1.0	1.7
	Radial	1.5	2.4	2.3	2.8
	1D	1.8	3.0	3.0	4.7
CG	Along	18.4	36.1	36.7	90.4
	Cross	7.5	13.4	12.5	19.8
	Radial	24.0	46.9	45.9	89.4
	1D	19.7	37.5	37.0	77.1
CI	Along	8.8	19	17.6	44.9
	Cross	3.6	6	5.8	10.2
	Radial	16	29.4	27.9	47.5
	1D	11.4	21.5	20.3	39.9
CM	Along	3.8	6.3	5.7	9.5
	Cross	1.2	1.7	1.5	2.4
	Radial	2.1	3.2	2.9	3.5
	1D	2.8	4.4	4.0	6.3

The unit is cm

Table 6 Averaged 1D RMS (solid bar) values of orbit discontinuities for different solutions

	Type	Mean	95th percentile		Type	Mean	95th percentile
G	EP_090_030	1.1	2.2	R	EP_090_030	1.9	3.7
	SB_090_060	2.2	4.2		SB_090_060	3.5	6.9
	EP_120_060	2.1	4.1		EP_120_060	3.4	6.8
	SB_120_120	3.8	7.3		SB_120_120	5.7	11.3
E	EP_090_030	1.6	3.1	CG	EP_090_030	15.5	31.9
	SB_090_060	3.2	6.4		SB_090_060	35.4	73.2
	EP_120_060	3.2	6.2		EP_120_060	34.7	70.3
	SB_120_120	6.7	13.3		SB_120_120	97.2	195.9
CI	EP_090_030	9.8	20.1	CM	EP_090_030	2.6	5.1
	SB_090_060	22.4	47.2		SB_090_060	4.8	9.5
	EP_120_060	21.5	45.8		EP_120_060	4.3	8.5
	SB_120_120	57.4	123.8		SB_120_120	8.5	16.8

Dashed box stands for the 95th percentile. The unit is cm

Table 7 RMS values of user-available part for different solutions. GPS orbits are compared with IGS Final products, while the other constellations are compared with GBM Rapid products

System	Type	EP_090_030	SB_090_060	EP_120_060	SB_120_120
G	Along	3.5	4.5	4.4	5.7
	Cross	1.9	2.1	2.1	2.4
	Radial	2.5	2.4	2.4	2.2
	1D	2.7	3.2	3.2	3.8
R	Along	8.0	9.1	8.8	10.5
	Cross	6.4	6.4	6.4	6.6
	Radial	3.9	4.0	4.0	4.0
	1D	6.5	7.0	6.9	7.7
E	Along	4.4	5.9	5.9	8.8
	Cross	2.5	2.7	2.7	3.1
	Radial	3.7	3.9	3.9	3.7
	1D	3.7	4.4	4.4	5.9
CG	Along	194.8	200.3	211.4	238.7
	Cross	60.9	62.6	63.6	68.2
	Radial	54.6	69.5	68.7	101.2
	1D	129.2	134.9	138.9	158.7
CI	Along	20.9	32.5	31.3	62.8
	Cross	10.8	12.6	12.1	15.4
	Radial	38.7	53.5	50.5	64.8
	1D	26.6	37.1	35.3	52.9
CM	Along	8.8	10.5	9.9	12.7
	Cross	4.1	4.3	4.1	4.6
	Radial	5.3	5.4	5.2	5.0
	1D	6.5	7.3	7.0	8.4

The unit is cm

References

Arnold D, Meindl M, Beutler G, Dach R, Schaer S, Lutz S, Prange L, Sosnica K, Mervart L, Jaggi A (2015) CODE's new solar radiation pressure model for GNSS orbit determination. J Geodesy 89:775–791. https://doi.org/10.1007/s00190-015-0814-4

Bertiger W, Bar-Sever Y, Dorsey A, Haines B, Harvey N, Hemberger D, Heflin M, Lu W, Miller M, Moore AW (2020) GipsyX/RTGx, a new tool set for space geodetic operations and research. Adv Space Res 66:469–489

Beutler G, Brockmann E, Gurtner W, Hugentobler U, Mervart L, Rothacher M, Verdun A (1994) Extended orbit modeling techniques at the CODE processing center of the international GPS service for geodynamics (IGS): theory and initial results. Manuscr Geodaet 19:367–386

Beutler G, Brockmann E, Hugentobler U, Mervart L, Rothacher M, Weber R (1996) Combining consecutive short arcs into long arcs for precise and efficient GPS orbit determination. J Geodesy 70:287–299

Blewitt G (1990) An automatic editing algorithm for GPS data. Geophys Res Lett 17:199–202

Blewitt; G, Bertiger; W, Weiss J-P. (2010). Ambizap3 and GPS carrier-range: a new data type with IGS applications. In: Proceedings of IGS workshop and vertical rates, Newcastle

Boehm J, Niell A, Tregoning P, Schuh H (2006) Global Mapping Function (GMF): a new empirical mapping function based on numerical weather model data. Geophys Res Lett. https://doi.org/10.1029/2005gl025546

Böhm J, Möller G, Schindelegger M, Pain G, Weber R (2014) Development of an improved empirical model for slant delays in the troposphere (GPT2w). GPS Solut 19:433–441. https://doi.org/10.1007/s10291-014-0403-7

Bruni S, Rebischung P, Zerbini S, Altamimi Z, Errico M, Santi E (2018) Assessment of the possible contribution of space ties on-board GNSS satellites to the terrestrial reference frame. J Geodesy 92:383–399

Cassey et al. (2012) The International GNSS Real-Time Service GPS World

Chandra R, Dagum L, Kohr D, Menon R, Maydan D, McDonald J (2001) Parallel programming in OpenMP. Morgan kaufmann, London

Chen G, Herring TA (1997) Effects of atmospheric azimuthal asymmetry on the analysis of space geodetic data. J Geophys Res Solid Earth 102:20489–20502

Chen H, Jiang W, Ge M, Wickert J, Schuh H (2014) Efficient high-rate satellite clock estimation for PPP ambiguity resolution using carrier-ranges. Sensors (basel) 14:22300–22312. https://doi.org/10.3390/s141222300

Chen Q, Song S, Zhou W (2021) Accuracy analysis of GNSS hourly ultra-rapid orbit and clock products from SHAO AC of iGMAS. Remote Sens. https://doi.org/10.3390/rs13051022

Chen X, Ge M, Hugentobler U, Schuh H (2022) A new parallel algorithm for improving the computational efficiency of multi-GNSS

precise orbit determination. GPS Solut. https://doi.org/10.1007/s10291-022-01266-8

Cui Y, Chen Z, Li L, Zhang Q, Luo S, Lu Z (2021) An efficient parallel computing strategy for the processing of large GNSS network datasets. GPS Solut. https://doi.org/10.1007/s10291-020-01069-9

Dai X, Lou Y, Dai Z, Qing Y, Li M, Shi C (2019) Real-time precise orbit determination for BDS satellites using the square root information filter. GPS Solut. https://doi.org/10.1007/s10291-019-0827-1

Deng Z, Fritsche M, Nischan T, Bradke M (2016) Multi-GNSS ultra rapid orbit-, clock- and EOP-product series. GFZ Data Serv. https://doi.org/10.5880/GFZ.1.1.2016.003

Deng Z, Schuh H (2018) Improvement of multi-GNSS orbit and clock prediction at GFZ. In: EGU General Assembly Conference Abstracts, pp 2017

Duan B, Hugentobler U (2021) Enhanced solar radiation pressure model for GPS satellites considering various physical effects. GPS Solut. https://doi.org/10.1007/s10291-020-01073-z

Duan B, Hugentobler U, Chen J, Selmke I, Wang J (2019) Prediction versus real-time orbit determination for GNSS satellites. GPS Solut. https://doi.org/10.1007/s10291-019-0834-2

Duan B, Hugentobler U, Hofacker M, Selmke I (2020) Improving solar radiation pressure modeling for GLONASS satellites. J Geod. https://doi.org/10.1007/s00190-020-01400-9

Elsobeiey M, Al-Harbi S (2016) Performance of real-time precise point positioning using IGS real-time service. GPS Solut 20(3):565–571

Ge M, Gendt G, Dick G, Zhang FP (2005) Improving carrier-phase ambiguity resolution in global GPS network solutions. J Geodesy 79:103–110. https://doi.org/10.1007/s00190-005-0447-0

Ge M, Gendt G, Dick G, Zhang FP, Rothacher M (2006) A new data processing strategy for huge GNSS global networks. J Geodesy 80:199–203. https://doi.org/10.1007/s00190-006-0044-x

Gong X, Gu S, Lou Y, Zheng F, Ge M, Liu J (2017) An efficient solution of real-time data processing for multi-GNSS network. J Geodesy 92:797–809. https://doi.org/10.1007/s00190-017-1095-x

GSA (2017) Galileo Satellite Metadata. https://www.gsc-europa.eu/support-to-developers/galileo-satellite-metadata. Accessed 31 Oct 2019

Guo J, Chen G, Zhao Q, Liu J, Liu X (2017) Comparison of solar radiation pressure models for BDS IGSO and MEO satellites with emphasis on improving orbit quality. GPS Solut 21:511–522. https://doi.org/10.1007/s10291-016-0540-2

Héroux P, Gao Y, Kouba J et al. (2004) Products and applications for Precise Point Positioning-Moving towards real-time. In: Proceedings of the 17th international technical meeting of the satellite division of The Institute of Navigation (ION GNSS 2004), pp 1832–1843

Jiang C, Xu T, Nie W, Fang Z, Wang S, Xu A (2021) A parallel approach for multi-GNSS ultra-rapid orbit determination. Remote Sens. https://doi.org/10.3390/rs13173464

Johnston G, Riddell A, Hausler G (2017) The international GNSS service. In: Teunissen PJ, Montenbruck O (eds) Springer handbook of global navigation satellite systems. Springer, Cham

Kouba J (2009) A guide to using International GNSS Service (IGS) products. https://files.igs.org/pub/resource/pubs/UsingIGSProductsVer21_cor.pdf

Laurichesse D, Cerri L, Berthias J, Mercier F (2013) Real time precise GPS constellation and clocks estimation by means of a Kalman filter. In: Proceedings of the 26th international technical meeting of the satellite division of the institute of navigation (ION GNSS+ 2013), pp 1155–1163

Li X, Chen X, Ge M, Schuh H (2018) Improving multi-GNSS ultra-rapid orbit determination for real-time precise point positioning. J Geodesy 93:45–64. https://doi.org/10.1007/s00190-018-1138-y

Li B, Ge H, Bu Y, Zheng Y, Yuan L (2022) Comprehensive assessment of real-time precise products from IGS analysis centers. Satell Navig. https://doi.org/10.1186/s43020-022-00074-2

Liu J, Ge M (2003) PANDA software and its preliminary result of positioning and orbit determination. Wuhan Univ J Nat Sci 8:603

Lou Y, Dai X, Gong X, Li C, Qing Y, Liu Y, Peng Y, Gu S (2022) A review of real-time multi-GNSS precise orbit determination based on the filter method. Satell Navig. https://doi.org/10.1186/s43020-022-00075-1

Parkinson B, Spilker J, Axelrad P, Enge P (1996) Progress in astronautics and aeronautics: global positioning system: theory and applications. AIAA, Reston

Petit G, Luzum B (2010) IERS conventions (2010). Bureau International des Poids et Mesures Sevres (France)

Pintori F, Serpelloni E, Gualandi A (2021) Common mode signals and vertical velocities in the great Alpine area from GNSS data. Solid Earth Discuss 5:1–37

Quintana-Orti G, Quintana-Orti ES, Chan E, Van de Geijn RA, Van Zee FG (2008) Scheduling of QR factorization algorithms on SMP and multi-core architectures. In: 16th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP 2008). IEEE, pp 301–310

Rebischung P, Schmid R (2016) IGS14/igs14. atx: a new framework for the IGS products. In: AGU Fall Meeting

Rodriguez-Solano C, Hugentobler U, Steigenberger P (2012) Adjustable box-wing model for solar radiation pressure impacting GPS satellites. Adv Space Res 49:1113–1128. https://doi.org/10.1016/j.asr.2012.01.016

Schaer S, Villiger A, Arnold D, Dach R, Prange L, Jäggi A (2021) The CODE ambiguity-fixed clock and phase bias analysis products: generation, properties, and performance. J Geodesy. https://doi.org/10.1007/s00190-021-01521-9

Springer T, Hugentobler U (2001) IGS ultra rapid products for (near-)real-time applications. Phys Chem Earth Part A 26:623–628

Springer TA, Beutler G, Rothacher M (1999) A new solar radiation pressure model for GPS satellites. GPS Solut 2:50–62. https://doi.org/10.1007/Pl00012757

Steigenberger P, Thoelert S, Montenbruck O (2017) GNSS satellite transmit power and its impact on orbit determination. J Geodesy 92:609–624. https://doi.org/10.1007/s00190-017-1082-2

Tang L, Wang J, Zhu H, Ge M, Xu A, Schuh H (2021) A comparative study on the solar radiation pressure modeling in GPS precise orbit determination. Remote Sens. https://doi.org/10.3390/rs13173388

Tang L, Wang J, Cui B, Zhu H, Ge M, Schuh H (2023) Multi-GNSS precise point positioning with predicted orbits and clocks. GPS Solut 27:162. https://doi.org/10.1007/s10291-023-01499-1

Villiger A, Dach R (2022) International GNSS service: technical report 2021 (IGS Annual Report). IGS Central Bureau Univ Bern. https://doi.org/10.48350/169536

Wang C, Guo J, Zhao Q, Liu J (2018) Empirically derived model of solar radiation pressure for BeiDou GEO satellites. J Geodesy 93:791–807. https://doi.org/10.1007/s00190-018-1199-y

Wang J, Ge M, Glaser S, Balidakis K, Heinkelmann R, Schuh H (2022) Improving VLBI analysis by tropospheric ties in GNSS and VLBI integrated processing. J Geodesy. https://doi.org/10.1007/s00190-022-01615-y

Zhao Q, Xu X, Ma H, Jingnan L (2018) Real-time precise orbit determination of BDS/GNSS: method and service. Geom Inform Sci Wuhan Univ 43:2157–2166

Zhao Q, Guo J, Wang C, Lyu Y, Xu X, Yang C, Li J (2022) Precise orbit determination for BDS satellites. Satell Navig. https://doi.org/10.1186/s43020-021-00062-y

Zumberge JF, Heflin MB, Jefferson DC, Watkins MM, Webb FH (1997) Precise point positioning for the efficient and robust analysis of GPS data from large networks. J Geophys Res Solid Earth 102:5005–5017

Zuo X, Jiang X, Li P, Wang J, Ge M, Schuh H (2021) A square root information filter for multi-GNSS real-time precise clock estimation. Satell Navig. https://doi.org/10.1186/s43020-021-00060-0

Zurutuza J, Caporali A, Bertocco M, Ishchenko M, Khoda O, Steffen H, Figurski M, Parseliunas E, Berk S, Nykiel G (2019) The Central European GNSS research network (CEGRN) dataset. Data Brief 27:104762