GPS Solutions (2023) 27:187
https://doi.org/10.1007/s10291-023-01528-z

GPS TOOLBOX

POSGO: an open-source software for GNSS pseudorange positioning based on graph optimization

Zhen Li[1] · Jing Guo[1] · Qile Zhao[1]

Received: 3 September 2022 / Accepted: 1 August 2023 / Published online: 17 August 2023

Abstract
Graph optimization (GO) can correlate more historical information to increase the resistance against the GNSS outliers. Therefore, GO has the potential to obtain a higher accuracy and robust position in urban canyon. Here, we develop POSGO (POSition based on Graph Optimization), an open-source software designed for single-point positioning and relative positioning with multi-GNSS pseudorange in a GO framework. It is coded in C/C++ language and recommended to run in the Linux environment. It can be easily extended to process the carrier phase and fuse the data from multiple sensors by adding corresponding graph factors. To assess the performance of the current version, data from a kinematic vehicle experiment in urban area are processed. The results indicate GO has better accuracy and robustness than classic least squares or Kalman filters, particularly in areas with severe occlusions.

Keywords GNSS · Open-source · Graph optimization · POSGO

Introduction

A traditional GNSS position is generally acquired with least squares (LSQ) or extended Kalman filter (EKF), and high performance can be obtained in open-sky areas. Unfortunately, GNSS positioning performance degrades in urban areas (Li et al. 2022), mainly due to none-line-of-sight (NLOS), multipath and obstructions. Graph optimization (GO) is an algorithm used to express optimization problems as graphs, the relationship between the states and measurements is represented with a graphic model, and then the state estimation is converted to a nonlinear least squares problem (Watson and Gross 2017). GO can correlate more historical position through various factors (such as velocity) and may further increase the resistance against outliers and decrease the position error, especially in the urban areas (Wen and Hsu 2021; Jiang et al. 2022).

EKF and LSQ can also be derived as instances of GO (Loeliger 2002). However, traditional EKF is based on Markov assumption, and current state is only related to the previous state. To solve this problem, sliding window filter and multi-state constraint Kalman filter processes all the measurements over a period of time (Mourikis and Roumeliotis 2007; Huang et al. 2011). However, they still suffer from large linearization errors, especially in the fusion of version (Huang 2019). In addition, GO has better flexibility. It can easily realize constraint between epochs and the addition of sensors by increasing the factors, while LSQ requires more algorithm-level adjustments (Indelman et al. 2013).

Due to the above characteristics, GO has been applied to GNSS positioning in urban area. Wen and Hsu (2021) conducted positioning experiments in the urban canyon of Hong Kong, and the results indicated that the accuracy of GO is better than traditional SPP based on LSQ and RTK based on EKF. Jiang et al. (2022) derive the same conclusion and show the position performance relies on the velocity accuracy. Taro Suzuki won the Google Smartphone Decimeter Challenge (GSDC) based on GO (Suzuki 2021), which shows the distinctive advantages of GO in urban areas.

The GPS Toolbox is a topical collection dedicated to highlighting algorithms and source code utilized by GNSS engineers and scientists. If you have a program or software package you would like to share with our readers, please submit a paper to the GPS Toolbox collection or email ngs.gps.toolbox@noaa.gov for more information. To download source code from this or any GPS Toolbox paper visit our website at http://geodesy.noaa.gov/gps-toolbox.

✉ Jing Guo
jingguo@whu.edu.cn

[1] GNSS Research Center, Wuhan University, No. 129 Luoyu Road, Wuhan 430079, China

At present, there are many excellent open-source GNSS software programs for high-accuracy positioning based on filter or least squares approaches, such as RTKLIB (Takasu and Yasuda 2009), GAMP (Zhou et al. 2018), and more. Here, we offer POSGO (POSition based on Graph Optimization), a software for multi-GNSS pseudorange positioning based on GO. Single-point positioning (SPP) based on single-frequency or dual-frequency ionospheric-free as well as relative positioning (RP) with multi-frequency pseudorange are supported. POSGO can serves as a fundamental platform for multi-sensor fusion due to high-precision, robustness and flexibility of GO.

First, a brief description of the graph structure for SPP and RP used in POSGO is given following the introduction. Afterward, the main features and general framework of POSGO are presented. The performance of the software is evaluated with a vehicle-based experiment in an urban area by comparison with LSQ and EKF. Finally, the study is concluded.

Mathematical model

In this section, we present a brief introduction of GO for SPP and RP using GNSS pseudorange and Doppler observations in POSGO software.

SPP using graph optimization

The graph structure of SPP is shown in Fig. 1. There are two factors in the SPP graph structure: the pseudorange factor $e^{s}_{\mathrm{P},i}$ and Doppler velocity factor e^{Dop}_{i}. GO can choose to use Doppler observations to estimate velocity and correlate historical positions to achieve better estimation. The objective function for SPP can be expressed as (Wen and Hsu 2021):

$$X^{*} = \operatorname{argmin} \sum\nolimits_{i}^{n} \left(\left\| e^{s}_{\mathrm{P},i} \right\|^{2}_{\Sigma^{s}_{\mathrm{P},i}} + \left\| e^{\mathrm{Dop}}_{i} \right\|^{2}_{\Sigma^{\mathrm{Dop}}_{i}} \right) \tag{1}$$

where n is the length of the sliding window. $\sum^{s}_{\mathrm{P},i}$ and $\sum^{\mathrm{Dop}}_{i+1}$. denotes the covariance matrix of pseudorange and velocity, respectively. $\|\|^{2}_{\Sigma}$ represents Mahalanobis norm. The optimal estimation X^{*} is obtained by minimizing the objective function.

RP using graph optimization

The graph structure of RP is shown in Fig. 2. Like SPP, there are also two factors in RP graph structure: the double-differenced pseudorange factor $e^{s}_{\mathrm{DD},i}$ and Doppler velocity factor e^{Dop}_{i}. Therefore, the objective function for RP can be obtained as follows (Wen and Hsu 2021):

$$X^{*} = \operatorname{argmin} \sum\nolimits_{i}^{n} \left(\left\| e^{s}_{\mathrm{DD},i} \right\|^{2}_{\Sigma^{s}_{\mathrm{DD},i}} + \left\| e^{\mathrm{Dop}}_{i} \right\|^{2}_{\Sigma^{\mathrm{Dop}}_{i}} \right) \tag{2}$$

$\sum^{s}_{\mathrm{DD},i}$ denotes the covariance of double-differenced pseudoranges.

POSGO software

POSGO is a multi-frequency and multi-GNSS pseudorange processing software based on GO. This software is written in the C/C+ +programming language (version 11 or higher). Two open-source math libraries are used, i.e., the matrix operation library Eigen (https://eigen.tuxfamily.org) and the Ceres for the optimization problem (http://www.ceres-solver.

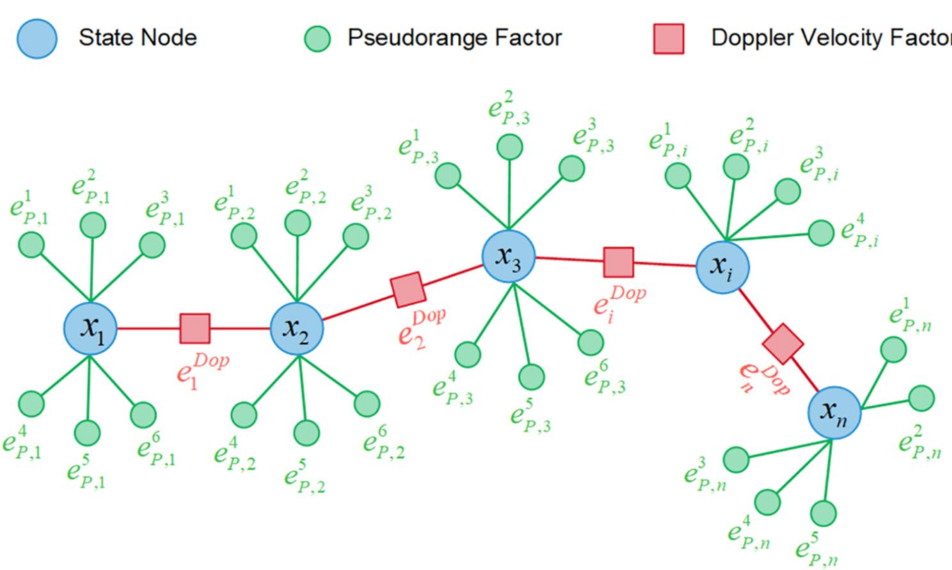

Fig. 1 Graph structure of SPP in POSGO

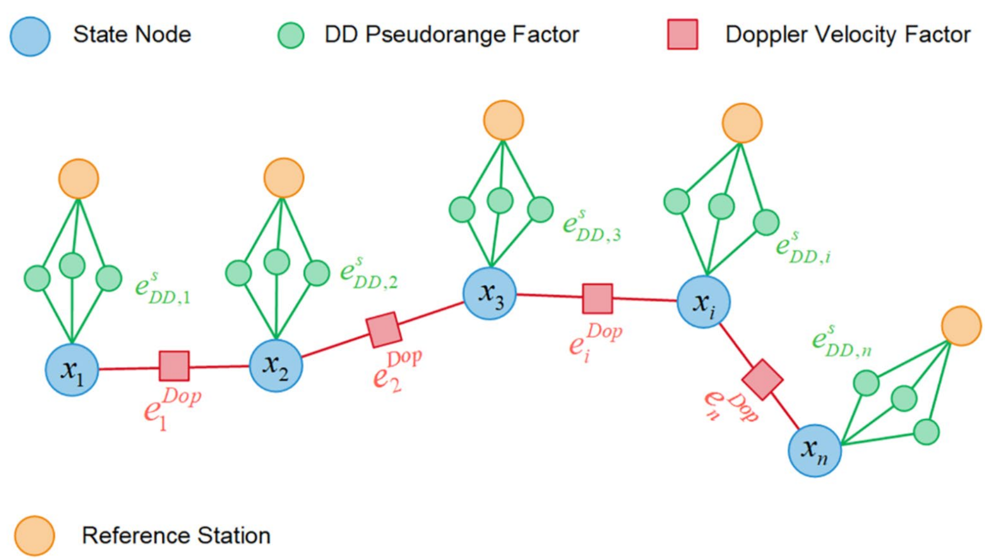

Fig. 2 Graph structure of RP in POSGO

org/). Detailed instructions of compiling, installation and instructions can be found in the User Manual. The main features of POSGO are listed in the following:

- SPP and RP based on GO with multi-frequency and multi-constellation pseudorange observations
- Support Cauchy and Huber loss functions for robust estimation in GO

In addition, to ensure the completeness of the software, POSGO also supports SPP based on LSQ and RP based on EKF. General framework of POSGO is shown in Fig. 3.

Experiment and analysis

Kinematic vehicle GNSS data in 1 Hz from the FoZuLing to the campus of Wuhan University in urban area with several heavily occluded areas were used to evaluate the performance of POSGO. Figure 4 shows the trajectory. The smoothed RTK solution from the NovAtel Waypoint 8.9 software is used as reference. Considering the effectiveness and computational efficiency of GO, 30-s sliding window was used.

SPP results and analysis

Multi-GNSS single-frequency data are processed based on SPP in the following scenarios:

- RTKLIB: Solver based on Single mode of RTKLIB version demo5 (https://github.com/rtklibexplorer/RTKLIB)
- GAMP: SPP based on GAMP
- LSQ: SPP based on LSQ with IGG3
- LSQD: like LSQ with Doppler observations under constant velocity model
- GO: SPP based on GO with a sliding window length of 1 s
- GOL: like GO, but Cauchy loss function was used
- GOLD: GO with 30 s window, Cauchy loss function, and Doppler factor

Table 1 lists the results for each scenario, where the solutions with horizontal or vertical errors less than 3.0 m or 5.0 m are considered as successful positioning. The results demonstrate that due to the addition of IGG3, all scenarios show better performance than that of RTKLIB and GAMP. By using Doppler for constant velocity model over consecutive epoch, LSQD outperforms LSQ. The accuracy of GO is worse than LSQ, mainly because the loss function is not used for quality control. By employing the loss function, the performance of GOL is close to LSQ. However, it does not demonstrate the advantage of GO, as the estimation is performed epoch-wise. GOLD shows the best performance, as it uses all the data in 30-s window to estimate the current position, which makes it more resistant to outliers.

GO has some advantages in areas with severe occlusion, even in the open-sky environment. Figure 5 shows the results of LSQ, LSQD and GOLD in a T-shaped intersection with

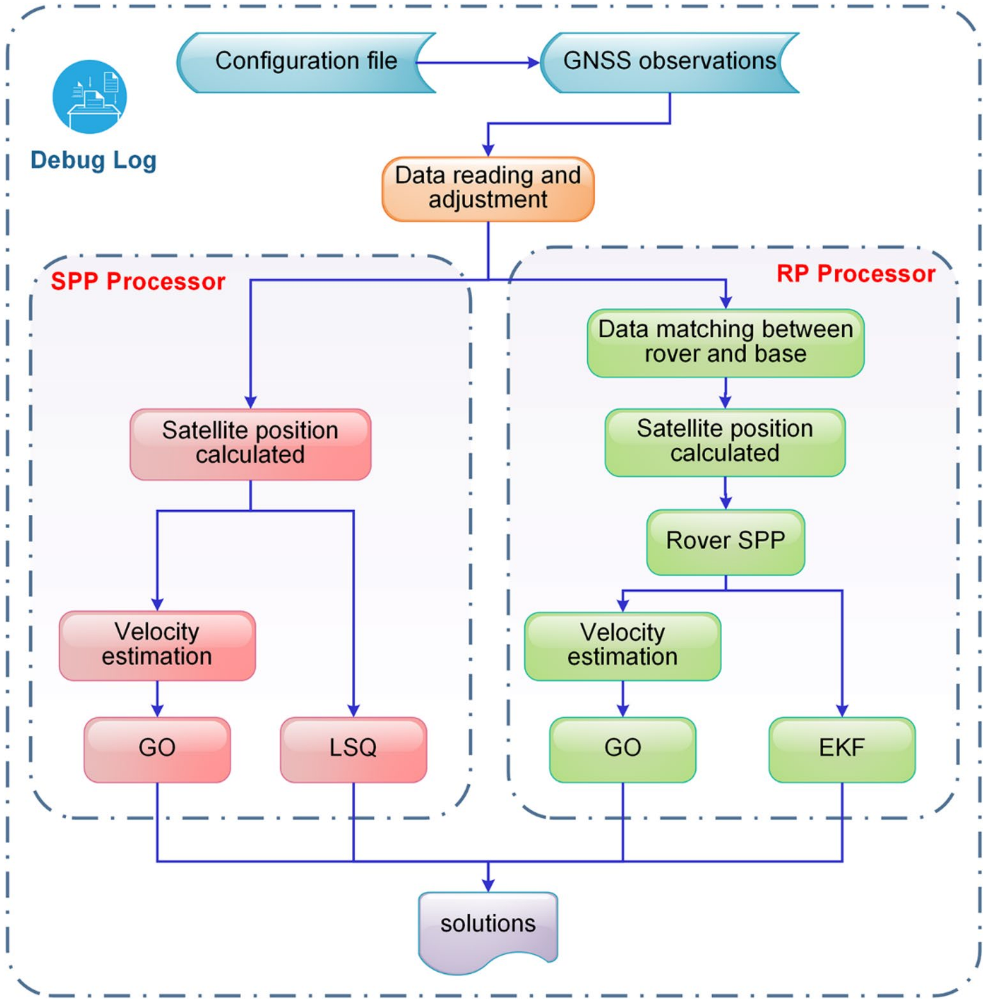

Fig. 3 General framework of POSGO. The software supports SPP with GO or LSQ and RP with GO or EKF, as well as the output of the debug information

buildings (Marker ② in Fig. 4) and an open-sky intersection where the vehicle stops for traffic lights(Marker ③ in Fig. 4). The 2D position error CDF (cumulative distribution function) around these two areas is also plotted in the left-upper corner in each subplot. It can be observed that GOLD shows a relatively better performance and stability.

RP results and analysis

The same data as SPP are processed in RP mode in the following scenarios:

- RTKLIB: Solver based on DGNSS mode of RTKLIB
- EKF: RP based on EKF with IGG3
- EKFD: like EKF with Doppler observations
- GO: RP based on GO with a 1 s window
- GOL: like GO, but Cauchy loss function is used
- GOLD: GO with a 30 s window, Cauchy loss function, and Doppler factor

The results for the different solutions are listed in Table 2. The solutions with horizontal or vertical errors less than 2 m or 3 m are considered as successful positioning. In the north direction, the solution EKF shows worse performance than that of RTKLIB. The degeneration is mainly due to the filter divergence caused by NLOS from the buildings in the last few epochs (see Fig. 6, Marker ① in Fig. 4). EKFD shows a slight improvement compared to EKF. As the solution EKF uses the prior information of the previous epoch, the result will be better than that of the epoch-wise optimization. By using the loss function, the accuracy of GOL is improved. Further improvement can be obtained by using a

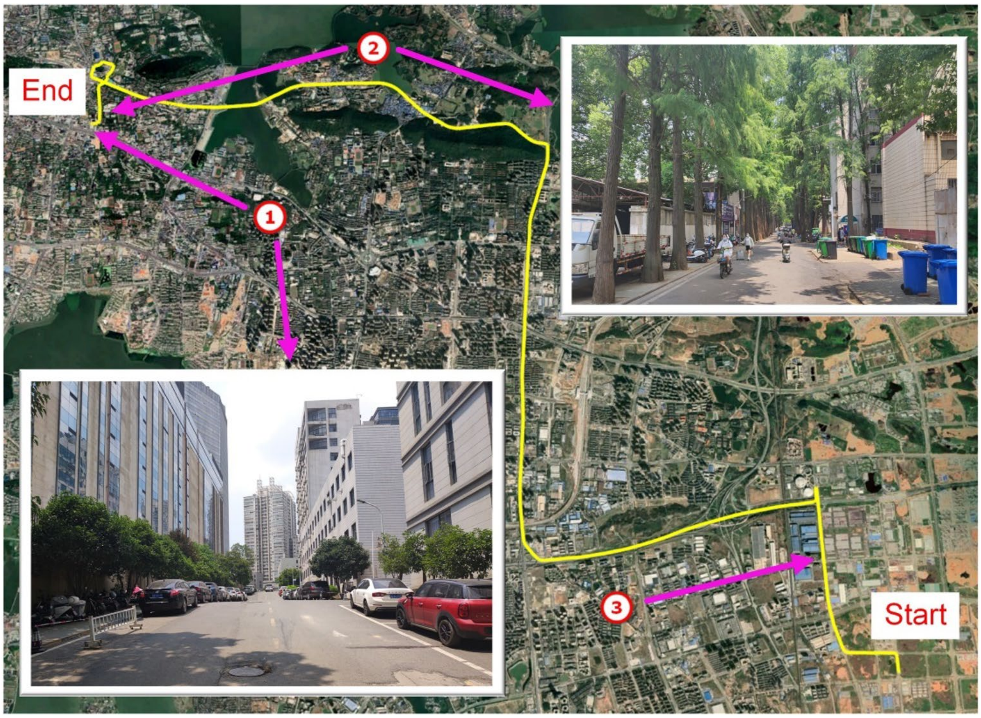

Fig. 4 Trajectory (yellow) of the vehicle. The marker ① represents the road heavily occluded by buildings, marker ② indicates a dense tree-shade road in the campus, and ③ denotes an open-sky road intersection

Table 1 Statistical of positioning accuracy for each SPP solution

Mode	Position accuracy (m)			Epoch number		
	North	East	Up	Success	Fail	Total
RTKLIB	1.97	1.73	1.64	2101	129	2230
GAMP	1.95	1.97	2.00	2078	145	2223
LSQ	1.28	1.32	1.35	2123	100	2223
LSQD	0.99	1.24	1.30	2136	92	2228
GO	1.68	1.53	1.40	2118	112	2230
GOL	1.02	1.35	1.45	2129	101	2230
GOLD	0.74	1.19	1.32	2161	69	2230

30-s window and Doppler factor to connect the states. Figure 6 shows the solutions of EKF, EKFD and GOLD near the Starlake Building of Wuhan University. This clearly demonstrates that GOLD can resist outliers better.

Summary and outlook

The open-source software POSGO is a multi-GNSS pseudorange position software based on GO. The performance of the software is evaluated with kinematic vehicle data, and results indicate good performance. In addition, GO shows better accuracy and robustness, particularly in areas with severe obstructions. GO can also be modified for real-time applications, as only the last node of the sliding window is output as the position result.

The POSGO is mainly to provide a new framework for GNSS positioning, particularly in urban areas, and promote the application of GO in GNSS and multi-sensor fusion. The software is continuing to evolve, and the carrier phase will be supported to achieve higher accuracy in the future. Furthermore, multi-sensor fusion is

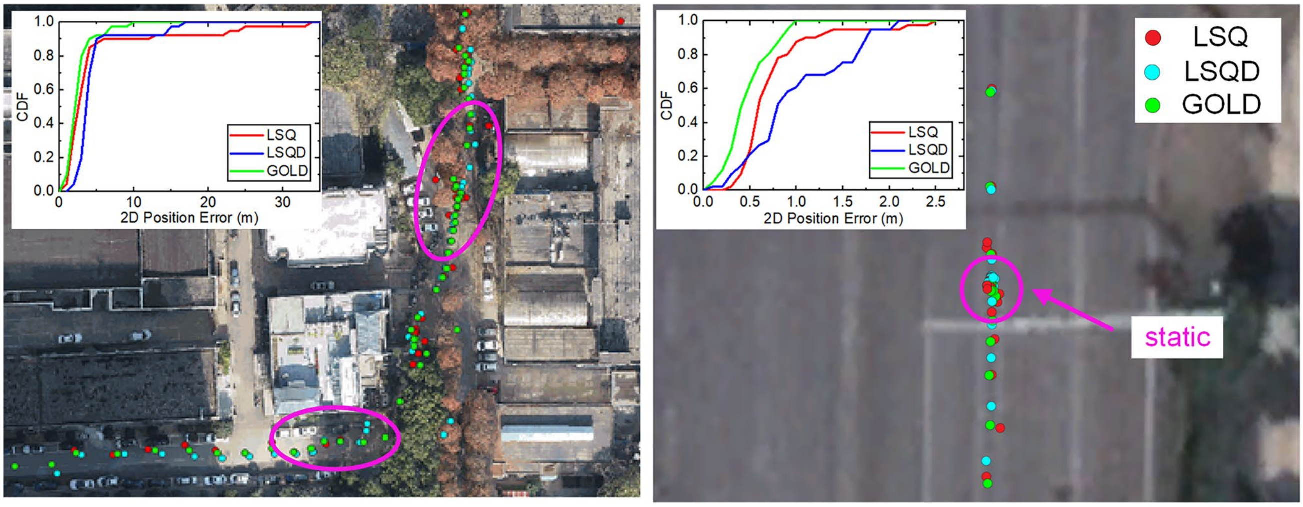

Fig. 5 SPP solutions of LSQ, LSQD and GOLD in a T-shaped intersection with buildings (left panel) and an open-sky intersection (right panel). The CDF of the positioning errors is also plotted

Table 2 Statistics of positioning accuracy for each RP solution

Mode	Position accuracy (m)			Epoch number		
	North	East	Up	Success	Fail	Total
RTKLIB	1.61	1.33	1.04	2134	96	2230
EKF	2.08	1.07	0.89	2139	91	2230
EKFD	1.85	1.02	0.81	2139	91	2230
GO	1.38	1.46	1.33	2123	105	2230
GOL	0.59	1.24	1.25	2143	93	2230
GOLD	0.59	1.13	0.79	2144	83	2230

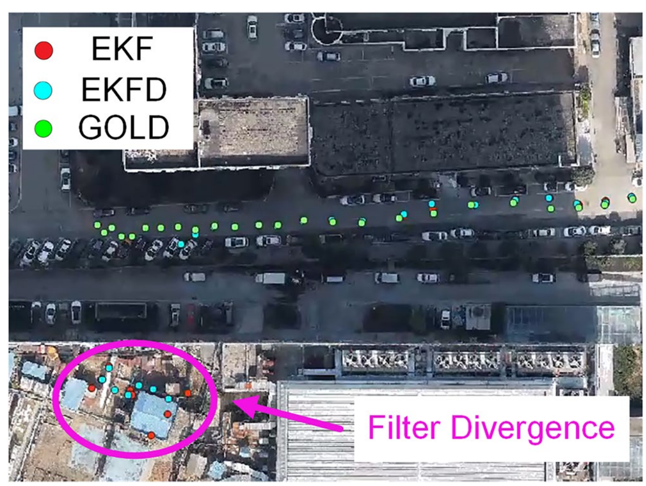

Fig. 6 RP solutions of EKF, EKFD and GOLD near the Starlake Building of Wuhan University

also planned to support. Some potential bugs may exist in POSGO, and we will continue to improve it for better performance. Comments and suggestions from users are welcome.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s10291-023-01528-z.

Acknowledgements This study is financially supported by the National Natural Science Foundation of China (41974035, 42030109), Yong Elite Scientists Sponsorship Program by China Association of Science and Technology (2018QNRC001), and Fundamental Research Funds for the Central Universities (2042021kf0065). The editor and reviewers are acknowledged.

Author contributions ZL and JG contributed to the initial idea; ZL assisted in coding, data analysis, and draft prepared; JG was involved in the data analysis; JG and QZ performed the manuscript review and revision.

Data availability The source code, user manual and sample data sets of POSGO can be found on the GPS Toolbox Web at https://geodesy.noaa.gov/gps-toolbox/ or GitHub at https://github.com/lizhengnss/POSGO.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

Huang G, Mourikis AI, Roumeliotis SI (2011) An observability-constrained sliding window filter for SLAM. 2011 IEEE/RSJ international conference on intelligent robots and systems. IEEE, San Francisco, CA, pp 65–72

Huang G (2019) Visual-inertial navigation: a concise review. In: 2019 international conference on robotics and automation (ICRA). pp 9572–9582

Indelman V, Williams S, Kaess M, Dellaert F (2013) Information fusion in navigation systems via factor graph based incremental smoothing. Robot Auton Syst 61(8):721–738. https://doi.org/10.1016/j.robot.2013.05.001

Jiang C, Chen Y, Jia J, Sun H, Wang T, Duan Z, Hyyppa J (2022) Open-source optimization method for android smartphone single point positioning. GPS Solut 26(3):90. https://doi.org/10.1007/s10291-022-01272-w

Li Z, Xu G, Guo J, Zhao Q (2022) A sequential ambiguity selection strategy for partial ambiguity resolution during RTK positioning in urban areas. GPS Solut 26(3):92. https://doi.org/10.1007/s10291-022-01279-3

Loeliger H-A (2002) Least squares and kalman filtering on forney graphs. In: Blahut RE, Koetter R (eds) Codes, graphs, and systems: a celebration of the life and career of G. David Forney, Jr. on the occasion of his sixtieth birthday. Springer US, Boston, MA, pp 113–135

Mourikis AI, Roumeliotis SI (2007) A multi-state constraint Kalman filter for vision-aided inertial navigation. In: Proceedings 2007 IEEE international conference on robotics and automation. pp 3565–3572

Suzuki T (2021) First place award winner of the smartphone decimeter challenge: global optimization of position and velocity by factor graph optimization. Proceedings ION GNSS 2021, institute of navigation, St. Louis, Missouri, USA, 20–24 Sept 2021 pp 2974–2985

Takasu T, Yasuda A (2009) Development of the low-cost RTK-GPS receiver with an open source program package RTKLIB. Proceedings international symposium on GPS/GNSS, Jeju, Korea, 4–6 November 2009

Watson RM, Gross JN (2017) Robust navigation In GNSS degraded environment using graph optimization. In: Proceedings ION GNSS 2017, institute of navigation, Portland, Oregon, USA, 25–29 Sept 2017, pp 2906–2918

Wen W, Hsu L-T (2021) Towards robust GNSS positioning and real-time kinematic using factor graph optimization. In: 2021 IEEE international conference on robotics and automation (ICRA). IEEE, Xi'an, China, pp 5884–5890

Zhou F, Dong D, Li W, Jiang X, Wickert J, Schuh H (2018) GAMP: An open-source software of multi-GNSS precise point positioning using undifferenced and uncombined observations. GPS Solut 22(2):33. https://doi.org/10.1007/s10291-018-0699-9

Zhen Li is a Ph.D. candidate at the GNSS Research Center, Wuhan University. His current research mainly focuses on precise positioning.

Jing Guo is an associate professor at GNSS Research Center of Wuhan University. He works on precise orbit determination for GNSS satellites and satellite-based argumentation systems.

Qile Zhao is a professor at the GNSS Research Center of Wuhan University. His current research interests are precise orbit determination.