

Proceeding Paper

Breaking the One-Meter Accuracy Level with Smartphone GNSS Data [†]

Marcus Franz Glaner *  and Robert Weber

Research Division Higher Geodesy, Department of Geodesy and Geoinformation, TU Wien, 1040 Vienna, Austria; robert.weber@geo.tuwien.ac.at

* Correspondence: marcus.glaner@geo.tuwien.ac.at

[†] Presented at the European Navigation Conference 2023, Noordwijk, The Netherlands, 31 May–2 June 2023.

Abstract: Smartphones are usually equipped with simple, cost-effective GNSS chips and antennas. They provide mainly single-frequency, low-quality, and challenging GNSS measurements. We demonstrate the difficulties of processing raw GNSS data from Android devices and introduce solutions to break the one-meter accuracy level with smartphones and Precise Point Positioning. With the logged data of a Samsung Galaxy S23+ and a Google Pixel 7 smartphone, a horizontal position accuracy of around one meter and a few decimeters was accomplished, respectively. These results were achieved after about two minutes of convergence time with our open-source software raPPPid in quasi-real-time settings. Furthermore, the corrections provided by the Galileo High-Accuracy Service proved to be sufficient to achieve sub-meter accuracy with smartphones.

Keywords: GNSS; PPP; Android; smartphone; raPPPid; Galileo HAS; low cost; uncombined; ionospheric constraint

1. Introduction

Before Android 7.0 was officially released in 2016, it was only possible to read out the position solution of a smartphone's internal algorithm [1,2]. Nowadays, everyone can access raw GNSS measurements tracked by Android smartphones and directly use the smartphone's GNSS observations to estimate the user's position [3–5]. Therefore, developing specialized algorithms and applying correction data can enhance positioning performance. However, smartphones are typically equipped with simple, cost-effective chips and antennas and, consequently, provide low-quality, single-frequency measurements with various challenges [6–8]. Altogether, experience has shown that the GNSS measurements of different smartphone devices behave diversely and unpredictably. To successfully process raw GNSS measurements from smartphones, we must study and understand their characteristics and differences from data recorded with geodetic equipment.

Precise Point Positioning (PPP) is characterized by applying precise satellite products (e.g., satellite orbits, clocks, and biases) and complex models to estimate the user's position [9]. PPP is a well-established processing technique for multi-frequency GNSS data from geodetic receivers and allows reaching accuracies at the centimeter level, or even below, after a notable convergence period [10]. Reducing or eliminating this convergence period is a major topic in scientific research. Usually, high-quality GNSS observations on two frequencies are used for PPP to build the ionosphere-free linear combination (IF LC), and researchers have found ways to integer fix the phase ambiguities [11,12]. Several adjustments of the PPP technique are necessary to process the challenging GNSS measurements from smartphones in an adapted PPP mode [13,14]. The approaches presented in the following are implemented in our open-source PPP software raPPPid [15,16], available at <https://github.com/TUW-VieVS/raPPPid> (accessed on 24 November 2023). Additionally,



Citation: Glaner, M.F.; Weber, R. Breaking the One-Meter Accuracy Level with Smartphone GNSS Data. *Eng. Proc.* **2023**, *54*, 16. <https://doi.org/10.3390/ENC2023-15465>

Academic Editors: Tom Willems and Okko Bleeker

Published: 29 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

the raPPPid wiki (<https://viewswiki.geo.tuwien.ac.at/en/raPPPid> (accessed on 24 November 2023) provides documentation and processing examples (e.g., with smartphone GNSS data).

The following section introduces the difficulties presented by using raw GNSS data from smartphones and an adapted PPP approach. Afterwards, Section 3 presents processing examples and Section 4 discusses the results. The final section provides a summary and outlook.

2. Challenges

Android devices significantly differ in their capabilities regarding raw GNSS measurements [17,18]. Considerable differences exist between smartphone devices of the same model, for example, sold in different regions. Access to the raw GNSS measurements might be limited (e.g., phase measurement) even on the most recent smartphone models [19]. Furthermore, a specific smartphone might not provide dual-frequency (DF) measurements due to, most likely, software restrictions, although it has a DF chip installed. Apart from that, DF capability means that the smartphone tracks signals on the L1 and L5 bands, and this frequency is currently not covered by all GNSS satellites (e.g., GPS modernization).

Naturally, Android repeatedly powers the GNSS hardware on and off to save battery energy, called duty-cycling [1]. This behavior causes non-continuous signal tracking and, for example, destroys the constant property of the phase ambiguity. Since Android 9, the user can force full GNSS measurements to deactivate duty-cycling and track all GNSS and frequencies in the developer settings [20]. On Android 12 or higher, this key configuration for precise positioning is available as an in-app setting. Figure 1 shows the triple-time-differenced GPS C1C code measurements of four consecutive epochs of a Samsung Galaxy S20 FE without deactivating duty-cycling. Building the triple-time difference with GNSS measurements recorded at a one-second observation interval typically nearly eliminates all effects, and the residuals represent the measurement noise. Consequently, the resulting code and phase residuals of geodetic GNSS equipment are usually the size of a few decimeters and a few millimeters, respectively. In Figure 1, the code residuals are significantly larger, and discontinuities occur due to non-deactivated duty cycling around epochs 82 and 116. These jumps also appear for other GNSS (same epochs, different magnitude) and complicate the processing of multiple GNSS in the PPP solution. However, they disappear when deactivating duty-cycling.

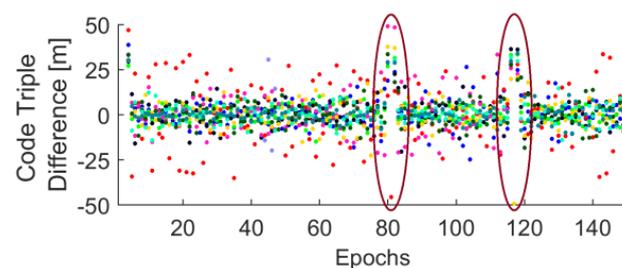


Figure 1. Triple-time-differenced code observations (GPS C1C, color-coded by satellite) for a Samsung Galaxy S20 FE when duty-cycling is not deactivated. Discontinuities due to duty-cycling are marked in dark ellipses.

The IF LC is commonly used for PPP, requiring observations on two frequencies to eliminate the ionospheric delay from the observation model. However, smartphones typically only provide single-frequency observations or a limited number of DF observations. Therefore, the uncombined PPP model with ionospheric constraint was chosen to connect the measurements and unknown parameters. This flexible PPP approach effectively handles the variable nature of smartphone GNSS observations by managing any number of frequencies, maintaining the raw observation noise, and including ionospheric pseudo-observations. For example, the uncombined model can consider different numbers

of frequencies for each satellite since missing observations are easily excluded from the parameter estimation process.

$$C_1 = \rho + c(dt_r + \delta t^g) + dTrop^{wet} + dIono_1 + \varepsilon \quad (1)$$

$$L_1 = \rho + c(dt_r + \delta t^g) + dTrop^{wet} - dIono_1 + \lambda_1 N_1 + \varepsilon \quad (2)$$

In the case of a single-frequency smartphone, Equations (1) and (2) present the observation equations of the uncombined model for the code observation C and the phase observation L , respectively. The geometric distance between the satellite and the smartphone is denoted as ρ and includes the unknown user position. The GPS receiver clock error dt_r and receiver clock offset δt^g for GNSS g are multiplied by the speed of light c . Furthermore, $dTrop^{wet}$ denotes the wet part of the tropospheric delay, and $dIono_1$ corresponds to the ionospheric delay. Equation (2) includes the float ambiguity N , which absorbs satellite and receiver phase biases and is multiplied by the corresponding wavelength λ . For readability reasons, these observation equations do not have an index to differentiate between GNSS or discriminate the term ε , containing noise and other errors. These observation Equations (1) and (2) comprise the unknown parameters, estimated with a Kalman filter in the PPP solution. Other error sources are modeled, like solid Earth tides, relativistic effects, and satellite phase center offsets.

$$C_2 = \rho + c(dt_r + \delta t^g) - DCB_{12} + dTrop^{wet} + f_1^2/f_2^2 dIono_2 + \varepsilon \quad (3)$$

$$L_2 = \rho + c(dt_r + \delta t^g) - DCB_{12} + dTrop^{wet} - f_1^2/f_2^2 dIono_2 + \lambda_2 N_2 + \varepsilon \quad (4)$$

In the case of a DF smartphone, Equations (3) and (4) are added to the parameter estimation for the corresponding satellites. DCB_{12} denotes the smartphone's differential code bias between the first and second processed frequency (e.g., GPS L1 and L5). The squared ratio of the signal frequencies f_1^2/f_2^2 converts the ionospheric delay to the first frequency.

$$dIono_{pseudo} = dIono_1 + \varepsilon \quad (5)$$

Furthermore, the uncombined model can integrate ionosphere models into the parameter estimation by adding the modeled ionospheric delays to the first frequency as pseudo-observations for each satellite ($dIono_{pseudo}$, Equation (5)). In this way, the inevitable imperfections of the ionosphere model can be adequately considered. Therefore, this so-called ionospheric constraint adds valuable information to the PPP solution, strengthens parameter estimation, and shortens the convergence time.

The phase measurements of smartphones are usually affected by ambiguity jumps and cycle slips. Detecting all these events is essential for PPP but complex, especially in single-frequency data, and not always feasible. Moreover, smartphones often do not provide access to the phase measurement. In such cases, a code-only solution has to be calculated by omitting Equations (2) and (4). Even such a code-only PPP approach allows a horizontal position accuracy around the one-meter level, as shown in the results section. However, by deactivating duty-cycling and considering the inconsistencies of some logger applications [21], the phase measurements of state-of-the-art smartphones can significantly contribute to breaking the one-meter accuracy level. Thereby, reliable cycle-slip detection is crucial, and the following approach based on Doppler observations proved to be effective.

$$L_{now} = L_{old} + dt \cdot \text{sqrt}(D_{now} \cdot D_{old}) \quad (6)$$

The phase observation of the previous epoch L_{old} is used with the Doppler observations of the previous and current epoch (D_{old} and D_{now} , respectively) to predict the phase observation of the current epoch L_{now} . For this method, the observation interval dt has to be sufficiently small, which is usually the case for smartphone data (1 s). The predicted

phase observation is compared with the actual phase measurement and a suitable threshold (e.g., 0.5 or 1 cycle).

The quality of GNSS observations from smartphones is generally worse than that of geodetic receivers. The carrier-to-noise density (C/N0) is typically around 10 dB.Hz lower and more variable (Figure 2). Furthermore, the code and phase observation noise is 10 to 100 times larger than for geodetic equipment (Figure 3). Anomalies like multipath or cycle-slips are usually much more pronounced due to the incorporated ultra-low-cost equipment. Therefore, the PPP processing should use a cutoff angle higher than usual (e.g., 10°) and exclude observations below a certain C/N0 threshold (e.g., 20 dB.Hz). Furthermore, the software should carefully check the measurements to eliminate suspicious observations. For example, the triple-time-differenced code measurements and the difference between the code measurement and observation model can be compared with reasonable thresholds (e.g., 50 and 25 m, respectively).

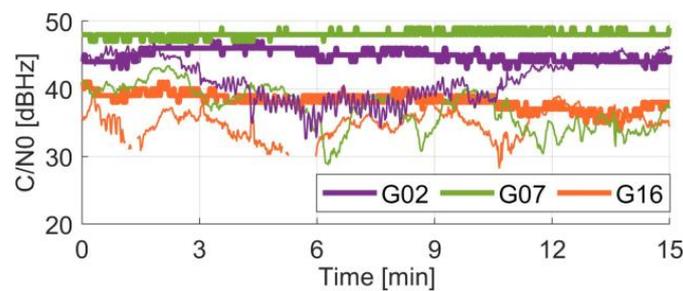


Figure 2. The carrier-to-noise density of three simultaneously observed GPS satellites tracked with a Trimble Spectra SP80 (thicker lines) and a Google Pixel 5 (thinner lines).

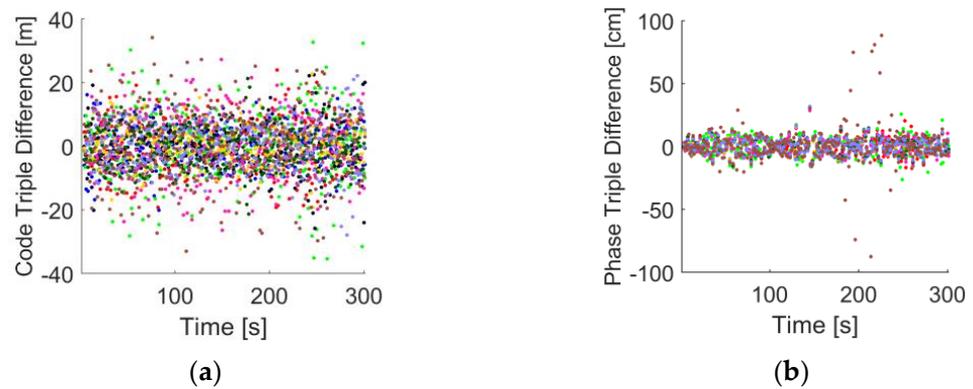


Figure 3. Triple-time differenced code (a) and phase (b) observations of a Google Pixel 7 (GPS C1C and L1C, color-coded by satellite). The corresponding standard deviations are 8.45 m and 12.9 cm.

Generally, the inaccurate weighting of the observations extends the convergence period or introduces biases to the parameters estimated with PPP. An elevation-based weighting approach (e.g., $\sin^2(e)$) is typically used in PPP since the observations’ residuals are correlated with the elevation in the case of geodetic GNSS equipment. However, no such relation usually exists for smartphones, but it can be found between the observations’ residuals and the C/N0 (Figure 4). Therefore, weighting the smartphones’ measurements based on the C/N0 usually significantly improves the performance and should be adjusted to the specific smartphone device. A simple CN/0 weighting function is used in the following (Table 1), leading to decent results in our experience.

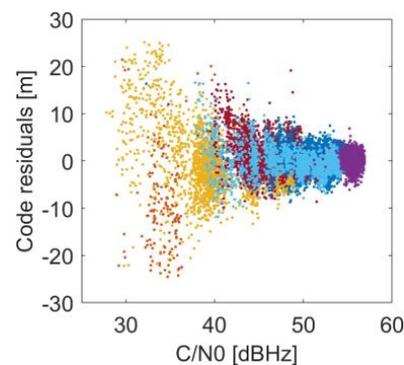


Figure 4. Code residuals (color-coded by satellite) over carrier-to-noise density for a Samsung Galaxy S20 FE.

Table 1. Quasi-real-time processing settings of raPPPid for processing raw GNSS measurements of smartphones. Options listed twice differentiate for the Galaxy S23+ and Pixel 7, respectively.

Smartphones	Samsung Galaxy S23+, Google Pixel 7
Date and period	4 and 19 April 2023; 60 min in total
Observation interval	1 s, reset of the solution every 6 min
Processed measurements	GPS (C1C + C5Q), GLONASS (C1C), Galileo (C1C + C5X), BeiDou (C2I)
Satellite orbits, clocks, and biases	Real-time corrections stream from CNES [22] or the Galileo HAS [23]
Processing mode	Static, quasi-real-time
Observation model	Uncombined model with ionospheric constraint
Raw observation noise	Code: 1 m/7 m, phase: NaN/0.01 m, ionosphere: 3 m
Observation weighting	$10^{-\frac{\max(0; 55-SNR)}{a}}$, with $a = 20$ or 10
Ionosphere model	GIOMO predicted [24] as ionospheric constraint, constant/released after 1 min
Troposphere model	GPT3 [25,26], ZWD not estimated/estimated
Correction models	Solid Earth tides, relativistic effects
Satellite exclusion criteria	$C/N_0 < 20$ dB.Hz or elevation $< 10^\circ$
Data quality checks (thresholds)	Observed minus computed (code: 25 m, phase 5 m), code triple-time difference (1 m), cycle-slip detection with Doppler (0.7 cy)

3. Results

This section presents position coordinate results obtained with two state-of-the-art smartphones (Google Pixel 7 and Samsung Galaxy S23+) and our open-source PPP software raPPPid [15,16]. The smartphones were placed on geodetic reference points on the rooftop of TU Wien, ensuring suitable reference coordinates. Considering the imperfections introduced by various logger applications to the code and phase observations [21], the raw output of all sensors was logged with the Google GnsLogger application for 30 min on two different days. The GNSS measurements of the resulting text files were converted to the RINEX format using UofC CSV2RINEX, available at <https://github.com/FarzanehZangeneh/csv2rinex> (accessed on 24 November 2023).

The PPP processing was performed in quasi-real-time (Table 1), applying real-time capable atmosphere models and satellite products (recorded real-time correction stream data from CNES [22] or the Galileo High-Accuracy Service (HAS) [23]). The PPP solution

was reset every 6 min during the PPP processing. Such a reset corresponds to a restart of the PPP calculations, and the resulting ten convergence periods can be used to assess the convergence behavior and accuracy of the coordinates.

Generally, GPS, GLONASS, Galileo, and BeiDou (GREC) observations are processed for both devices. The variance of GLONASS and BeiDou observations is increased by a factor of 1.25 because their real-time satellite products are usually less accurate. Since the Galileo HAS provides real-time corrections for GPS and Galileo, a GPS and Galileo (GE) solution was calculated with these satellite orbits, clocks, and code biases. Unfortunately, the Samsung Galaxy S23+ does not provide access to the phase measurements. Therefore, a code-only solution is calculated for this device. On the other hand, code and phase measurements are processed in the case of the Google Pixel 7. Due to the different characteristics of code-only and code + phase PPP processing, some settings must be modified (listed twice in Table 1). For example, the ZWD is estimated only for the Google Pixel 7 because the phase observations substantially improve the accuracy of the solution.

Figure 5 shows the convergence behavior of the horizontal position error for the Samsung Galaxy S23+ and the Google Pixel 7, applying real-time corrections from CNES. Table 2 presents the corresponding statistics. Convergence is defined as the period until the horizontal position difference is below 1 m for the whole remaining period. The introduced PPP approach achieves a 2D position accuracy around the one-meter level after a convergence time of 2–4 min using only code observations from the Galaxy S23+. In the case of the Google Pixel 7, the 2D position accuracy is clearly below one meter, thanks to the phase observations, and the convergence time is typically 1–2 min. After 6 min, even the 3D position error is at the one-meter level for this device (Table 2).

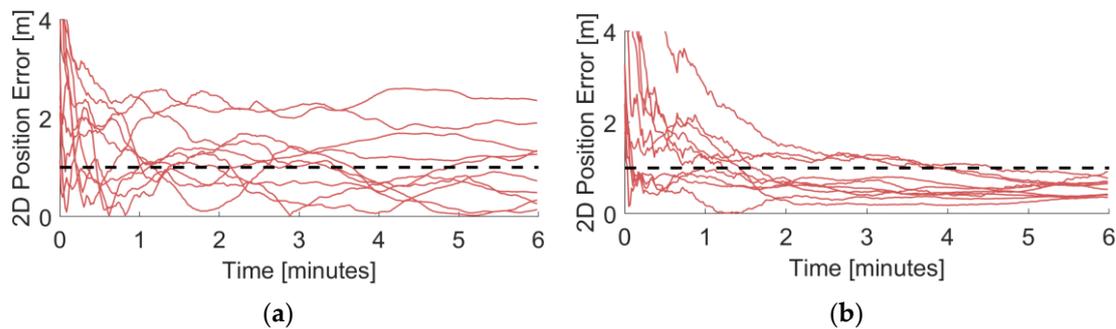


Figure 5. Horizontal position difference applying real-time corrections from CNES in a GREC processing for (a) Samsung Galaxy S23+ and (b) Google Pixel 7.

Table 2. Accuracy and convergence time applying real-time corrections from CNES in a GREC processing.

Smartphone	Median 2D Position Error [cm]	Mean 2D Position Error after 6 min [cm]	Mean 3D Position Error after 6 min [cm]	Mean Convergence [min]	Converged [%]
Samsung Galaxy S23+	112.5	101.4	258.8	2.33	50
Google Pixel 7	66.5	58.7	108.8	1.70	100

Furthermore, Figure 6 shows the convergence behavior using Galileo HAS corrections in a GPS and Galileo processing, and Table 3 presents the corresponding statistics. Naturally, these results based only on GPS and Galileo observations are worse than the GREC results (Table 2) due to significantly fewer satellites and observations. Furthermore, when comparing Table 2 with Table 4, the real-time clock corrections provided by the Galileo HAS seem slightly worse than the corresponding real-time corrections from CNES. Nevertheless, the Galileo HAS is sufficient to achieve sub-meter accuracy with smartphones.

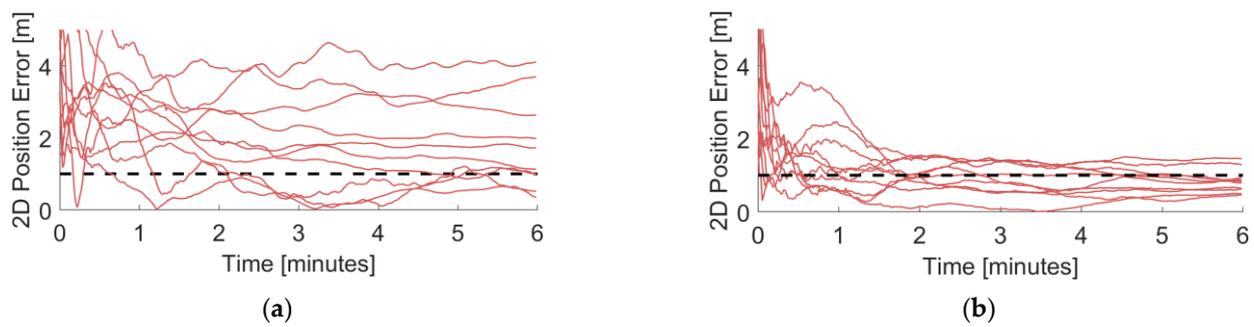


Figure 6. Horizontal position difference applying corrections from the Galileo HAS in a GE processing for (a) Samsung Galaxy S23+ and (b) Google Pixel 7.

Table 3. Accuracy and convergence time applying corrections from the Galileo HAS in a GE processing.

Smartphone	Median 2D Position Error [cm]	Mean 2D Position Error after 6 min [cm]	Mean 3D Position Error after 6 min [cm]	Mean Convergence [min]	Converged [%]
Samsung Galaxy S23+	190.3	180.9	384.3	3.96	30
Google Pixel 7	99.9	84.0	261.3	2.63	80

Table 4. Accuracy and convergence time applying real-time corrections from CNES in a GE processing.

Smartphone	Median 2D Position Error [cm]	Mean 2D Position Error after 6 min [cm]	Mean 3D Position Error after 6 min [cm]	Mean Convergence [min]	Converged [%]
Samsung Galaxy S23+	182.6	133.9	245.6	3.05	60
Google Pixel 7	93.9	79.0	208.4	2.25	70

4. Discussion

The presented results were accomplished under good conditions since the smartphones were statically placed on geodetic reference points, ensuring an open sky. Notably, they were achieved using quasi-real-time processing, and the smartphones could perform these PPP calculations directly. Therefore, sub-meter accuracy is possible with smartphones and PPP after a convergence time of 2–3 min, even when only GPS and Galileo observations are processed with corrections from the Galileo HAS. More complex modeling approaches, like a high-quality ionosphere model or more sophisticated observation weighting, might improve the results and are the subject of future investigations. Furthermore, smartphone phase center offsets are not considered due to a lack of calibrations, introducing an uncertainty of several centimeters. If possible, including smartphones' phase observations significantly improves the PPP performance. They proved the key to achieving a 2D position accuracy at the decimeter level. Unfortunately, not all Android devices currently provide access to phase measurements. Furthermore, several key issues must be considered to ensure consistency between the code and phase observations.

5. Summary

This article describes the key challenges of processing the GNSS measurements from Android smartphones and introduces solutions to achieve decimeter-level accuracy with PPP. The presented approaches are implemented in our open-source software raPPPId, available at <https://github.com/TUW-VieVS/raPPPId> (accessed on 24 November 2023). With the GNSS data from two state-of-the-art smartphones recorded under good conditions,

raPPPid achieved a 2D position accuracy around the one-meter level (code-only, Galaxy S23+) and at the few-decimeter level (code + phase, Pixel 7) in quasi-real-time processing after a convergence time of around two minutes. Also, the real-time corrections from the Galileo HAS enable sub-meter accuracy with smartphones after a similar convergence period.

Due to the variable characteristic of GNSS measurements from smartphones, more adaptations and refinements of the observation model are subject to further studies. Future work will focus on kinematic test cases and the fusion of PPP with other sensors (e.g., accelerometer). Currently, we are developing an Android application performing these PPP calculations directly on the smartphone.

Author Contributions: Conceptualization, M.F.G. and R.W.; methodology, M.F.G.; software, M.F.G.; validation, M.F.G. and R.W.; formal analysis, M.F.G.; investigation, M.F.G.; resources, M.F.G.; data curation, M.F.G.; writing—original draft preparation, M.F.G.; writing—review and editing, M.F.G. and R.W.; visualization, M.F.G.; supervision, R.W.; project administration, M.F.G. and R.W.; funding acquisition, R.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Austrian Research Promotion Agency (FFG) within the research project APPP (Advanced Precise Point Positioning for mass market GNSS receivers on Android devices).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used in this study are available on request from the corresponding author. The open-source PPP software raPPPid is available on GitHub (<https://github.com/TUW-VieVS/raPPPid>, accessed on 24 November 2023), and the raPPPid wiki provides documentation and processing examples (<https://viewswiki.geo.tuwien.ac.at/en/raPPPid>, accessed on 24 November 2023).

Acknowledgments: We want to thank all organizations which are operationally computing and disseminating satellite products enabling PPP.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zangenehjad, F.; Gao, Y. GNSS Smartphones Positioning: Advances, Challenges, Opportunities, and Future Perspectives. *Satell. Navig.* **2021**, *2*, 24. [[CrossRef](#)] [[PubMed](#)]
2. European GNSS Supervisory Authority. *Using GNSS Raw Measurements on Android Devices: White Paper*; Publications Office: Luxembourg, 2017.
3. Wang, L.; Li, Z.; Wang, N.; Wang, Z. Real-Time GNSS Precise Point Positioning for Low-Cost Smart Devices. *GPS Solut.* **2021**, *25*, 69. [[CrossRef](#)]
4. Everett, T.; Taylor, T.; Lee, D.-K.; Akos, D.M. Optimizing the Use of RTKLIB for Smartphone-Based GNSS Measurements. *Sensors* **2022**, *22*, 3825. [[CrossRef](#)] [[PubMed](#)]
5. Suzuki, T. Precise Position Estimation Using Smartphone Raw GNSS Data Based on Two-Step Optimization. *Sensors* **2023**, *23*, 1205. [[CrossRef](#)] [[PubMed](#)]
6. Zhang, X.; Tao, X.; Zhu, F.; Shi, X.; Wang, F. Quality Assessment of GNSS Observations from an Android N Smartphone and Positioning Performance Analysis Using Time-Differenced Filtering Approach. *GPS Solut.* **2018**, *22*, 70. [[CrossRef](#)]
7. Liu, W.; Shi, X.; Zhu, F.; Tao, X.; Wang, F. Quality Analysis of Multi-GNSS Raw Observations and a Velocity-Aided Positioning Approach Based on Smartphones. *Adv. Space Res.* **2019**, *63*, 2358–2377. [[CrossRef](#)]
8. Wanninger, L.; Heßelbarth, A. GNSS Code and Carrier Phase Observations of a Huawei P30 Smartphone: Quality Assessment and Centimeter-Accurate Positioning. *GPS Solut.* **2020**, *24*, 64. [[CrossRef](#)]
9. Kouba, J.; Lahaye, F.; Tétreault, P. Precise Point Positioning. In *Springer Handbook of Global Navigation Satellite Systems*; Teunissen, P.J.G., Montenbruck, O., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 723–751. ISBN 978-3-319-42928-1.
10. Choy, S.; Bisnath, S.; Rizos, C. Uncovering Common Misconceptions in GNSS Precise Point Positioning and Its Future Prospect. *GPS Solut.* **2017**, *21*, 13–22. [[CrossRef](#)]
11. Ge, M.; Gendt, G.; Rothacher, M.; Shi, C.; Liu, J. Resolution of GPS Carrier-Phase Ambiguities in Precise Point Positioning (PPP) with Daily Observations. *J. Geod.* **2008**, *82*, 389–399. [[CrossRef](#)]
12. Glaner, M.; Weber, R. PPP with Integer Ambiguity Resolution for GPS and Galileo Using Satellite Products from Different Analysis Centers. *GPS Solut.* **2021**, *25*, 102. [[CrossRef](#)]

13. Shinghal, G.; Bisnath, S. Conditioning and PPP Processing of Smartphone GNSS Measurements in Realistic Environments. *Satell. Navig.* **2021**, *2*, 10. [[CrossRef](#)]
14. Li, Z.; Wang, L.; Wang, N.; Li, R.; Liu, A. Real-Time GNSS Precise Point Positioning with Smartphones for Vehicle Navigation. *Satell. Navig.* **2022**, *3*, 19. [[CrossRef](#)]
15. Glaner, M.F.; Weber, R. An Open-Source Software Package for Precise Point Positioning: raPPPid. *GPS Solut.* **2023**, *27*, 174. [[CrossRef](#)]
16. Glaner, M.F. Towards Instantaneous PPP Convergence Using Multiple GNSS Signals. Ph.D. Thesis, TU Wien, Vienna, Austria, 2022.
17. Crowdsourcing GNSS features of Android Devices. Available online: <https://barbeau.medium.com/crowdsourcing-gnss-capabilities-of-android-devices-d4228645cf25> (accessed on 4 April 2023).
18. GPSTest Database. Available online: https://docs.google.com/spreadsheets/d/1jXtRCoEnnFNWj6_oFIVWflsf-b0jZpyhNBXsv7uo/edit#gid=0 (accessed on 4 April 2023).
19. Raw GNSS Measurements. Available online: <https://developer.android.com/guide/topics/sensors/gnss> (accessed on 4 April 2023).
20. GNSS, Interrupted: The Hidden Android Setting You Need to Know. Available online: <https://barbeau.medium.com/gnss-interrupted-the-hidden-android-setting-you-need-to-know-d812d28a3821> (accessed on 4 April 2023).
21. Zangenehnejad, F.; Jiang, Y.; Gao, Y. GNSS Observation Generation from Smartphone Android Location API: Performance of Existing Apps, Issues and Improvement. *Sensors* **2023**, *23*, 777. [[CrossRef](#)] [[PubMed](#)]
22. PPP-WIZARD: Daily Products. Available online: <http://www.ppp-wizard.net/daily.html> (accessed on 11 April 2023).
23. Galileo High Accuracy Service (HAS). Available online: <https://www.gsc-europa.eu/galileo/services/galileo-high-accuracy-service-has> (accessed on 4 April 2023).
24. Magnet, N. Giomo a Robust Modelling Approach of Ionospheric Delays for GNSS Real-Time Positioning Applications. Ph.D. Thesis, TU Wien, Vienna, Austria, 2019.
25. Landskron, D.; Böhm, J. VMF3/GPT3: Refined Discrete and Empirical Troposphere Mapping Functions. *J. Geod.* **2018**, *92*, 349–360. [[CrossRef](#)] [[PubMed](#)]
26. Landskron, D.; Böhm, J. Refined Discrete and Empirical Horizontal Gradients in VLBI Analysis. *J. Geod.* **2018**, *92*, 1387–1399. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.