Raw GNSS Data Analysis for the LEDSOL Project – Preliminary Results and Way Ahead

Elena Simona Lohan[1], Tomkouani Kodom[2], Hafida Lebik[3], Antoine Grenier[1], Xiaolong Zhang[1], Oana Cramariuc[4], Irina Mocanu[4], Kathrin Bierwirth[5] and Jari Nurmi[1]

[1]*Electrical Engineering unit, Tampere University, FI*
[2]*University of Lomé, Laboratory of Applied Hydrology and Environment, Togo*
[3]*Unité de Développement des Equipements Solaires, UDES/EPST- Centre de Développement des Énergies Renouvelables, Algeria*
[4]*Centrul IT pentru Stiinta si Tehnologie SRL (CITST), Romania*
[5]*Institut für Sozialforschung und Sozialwirtschaft (iso) e.V., Germany*

Abstract

This is a work-in-progress paper describing the methodology to acquire and process raw GNSS data from Android devices, as well as preliminary results based on measurement campaigns conducted in Finland, Togo, and Algeria. The raw GNSS data acquisition is now supported by most Android smartphones (version 10 or newer) and such data can serve as very useful research data to develop low-cost positioning algorithms as well as to do statistical analysis under various scenarios, especially when working with imperfect, inaccurate, or incomplete pseudorange data. This paper presents the methodological steps to follow in the process of data collection and provides an initial analysis of the various error models in GNSS-based positioning. The challenges and potential solutions and future steps are also emphasized.

Keywords

Global Navigation Satellite Systems (GNSS), raw pseudoranges, Android smartphones, water access, energy efficiency, Matlab-based software

1. Introduction and motivation

Since 2016, the researchers have had the possibility to access raw Global Navigation satellite Systems (GNSS) data on Android smartphones. This access has already enabled a multi-faceted research work in areas related to wireless positioning in urban areas in cooperative [1] or non-cooperative manners [2], velocity estimates for e-Health, sports and fitness applications [3], risk assessment in GNSS [4], spoofing and jamming awareness and mitigation of interference

---

WIPHAL 2023: Work-in-Progress in Hardware and Software for Location Computation, June 06–08, 2023, Castellon, Spain

elena-simona.lohan@tuni.fi (E. S. Lohan); tomkouani.kodom@gmail.com (T. Kodom); h.lebik@uses.dz (H. Lebik); antoine.grenier@tuni.fi (A. Grenier); xiaolong.zhang@tuni.fi (X. Zhang); oana.cramariuc@citst.ro, (O. Cramariuc); irina.mocanu@citst.ro (I. Mocanu); bierwirth@iso-institut.de (K. Bierwirth); jari.nurmi@tuni.fi (J. Nurmi)
0000-0003-1718-6924 (E. S. Lohan); 0000-0003-3776-5990 (T. Kodom); 0000-0002-1116-3564 (H. Lebik); 0000-0002-3440-8659 (A. Grenier); 0000-0001-7257-4028 (X. Zhang); 0000-0002-7499-5704 (O. Cramariuc); 0000-0001-5176-9344 (I. Mocanu); 0009-0006-9948-9740 (K. Bierwirth); 0000-0003-2169-4606 (J. Nurmi)

CEUR Workshop Proceedings (CEUR-WS.org)

in GNSS bands [5, 6], for agriculture applications such as sensing water in soil [7], ionospheric models such as analyzing the Total Electron Content (TEC) [8], etc. The potential of using such lower-quality raw GNSS data harnessed from mobile devices instead of high-quality raw GNSS data harnessed from professional GNSS receivers is also gaining interest in the context of multiple worldwide sustainable development goals (SDG), such as SDG6 on clean water and sanitation [9, 10, 11] and SDG7 on affordable and clean energy and energy efficiency [12], which are two of the main goals on the on-going EU-funded LEDSOL project [13].

The LEDSOL project is conducted by a transnational team with units from Europe and Africa and working on producing clean water by exploiting innovative research and novel technologies focused on three axes: water purification through Ultra-Violet (UV) and Light Emission Diode (LED) solutions, harnessing solar energy through solar panels in order to enable a self-sustainable and standalone solution, and an added-value feature of low-cost wireless positioning with only GNSS-based data (no additional sensors) to enable, for example, route optimization, mobility patterns modeling and tracking, water-sources location tagging and, possibly geo-fencing for certain area protection. Basically, the disinfection units are assumed to be portable and the wireless localization part of the project, addressed in here, can serve to know where they are in order to obtain clean water; in addition, the wireless localization part can also help the workers/people to navigate from point A to point B in unknown environments (e.g., looking for a new water source). This paper focuses only on the last aspect of LEDSOL project, namely the low-cost positioning based on raw GNSS data collected from mass-market Android devices.

The work presented here is a work in progress and the novel contributions addressed in our paper are related to the wireless positioning part of the envisaged solution of the LEDSOL project and are listed below:

- Describing a methodology of data collection and analysis of raw GNSS data from Android smartphones, with the focus on error modeling of various sources of errors and achieving good positioning estimated with noisy data;
- Presenting the data collection campaigns in Finland, Togo, and Algeria and showing the initial results based on the collected data;
- Summarizing the open-challenges towards low-cost accurate positioning based on raw GNSS data collected from Android smartphones.

On a broader note, the LEDSOL envisaged solution [14, 15] is aiming to give access to affordable energy for clean water production to a wide number of stakeholders and to maximize the socio-economic impact through its deployment to remote areas. The overall main goals of LEDSOL project are to to support clean water availability to population relying on unsafe water sources, to foster long-term collaboration between African and European organizations on sustainable and affordable technologies, and to provide off the grid clean water solutions, by using a smart portable unit based on UV/LED disinfection augmented with classical decontamination and standalone, low-cost wireless positioning engine, as well as ensuring energy efficiency through renewable energy as source of battery power. While the low-cost wireless positioning engine is not an essential component of the water-disinfection/purification system, it brings an important added value to our solution that can serve multiple purposes, such as water-source location and accurate geotagging, geofencing for resource protection, navigation and tracking, etc.

The rest of the paper is structured as follows: section 2 discusses the methodology for the data analysis, including the device selection, the data collection, the analysis software, the error models, and the positioning algorithm; section 3 presents our preliminary results based on data collected in Finland, Togo, and Algeria, and section 4 addresses the open challenges and future steps.

2. Methodology for the data analysis

While a GNSS-based positioning can be performed anywhere in the world without additional terrestrial infrastructure, the smartphones often rely on additional mobile networks in order to recover satellites information and thus save battery (i.e. Assisted-GNSS [16]). Remote environments, such as that targeted by the LEDSOL project might not have wireless connections to such networks, thus full recovery of the navigation message from GNSS signals is required. Unfortunately, not every smartphone provides access to this data. Within the project scope, multiple devices were and continue to be tested under various conditions in order to select the most adequate device.

2.1. Device selection

Since the deployment of Android 10, every Android device should provide access to raw GNSS measurements. Yet, as GNSS chipset and implementation might vary between brands and device models, measurements discrepancies are present in GNSS data acquired by smartphones [17]. A non-comprehensive list of Android device, initiated by [18] and completed by the GNSS community through crowd-sourcing, is available online [19]. A vast collection of devices entered at [19] allow everyone interested in this field to review their features and to choose the most relevant ones for the desired research scope. While this list is incomplete and might present contradictory information, it has provided a first filter on interesting devices for our research work. Yet, real testing of actual capacity of the device is required and, in some of the places of our measurements, the choice was limited to available devices at that place, which may always be an additional restriction in the system design (e.g., a phone evaluated with better features, for example based on [19] may have a prohibitive price or even be unavailable on the narket in a region of interest). Therefore, it is important to analyze the data with both low-end and high-end devices in order to acquire a comprehensive understanding of the tradeoffs between high positioning accuracy, noisy or incompletely available data, and storage and processing capacity.

As precise positioning of the user is desirable, the main selection criteria were dual-frequency devices with navigation message recovery. Out of the many listed devices, only a subset answered the requirements and seven models were selected for testing; their availability at the place of the measurements played a big role in selecting them; for example the choices in Togo and Algeria were limited to single-frequency receivers, while four out of the five measurement devices used in Finland supported dual-frequency measurements. The selected devices are listed in Table 1 along with their capacities. As seen here, the devices available and used in

Location	Device name	Android (API)	Dual-frequency	Carrier phase	Nav. message
Finland	Google Pixel 7	Android 13 (API 33)	✓	✓	✓
Finland	Xiaomi 11T	Android 12 (API 31)	✓	✓	
Finland	OnePlus Nord2 5G	Android 12 (API 31)	✓	✓	✓
Finland	Samsung A52 5G	Android 13 (API 33)			
Finland	Nokia XR20	Android 12 (API 31)	✓		
Togo	Itel P36	Android 10 (API 29)			
Algeria	Oppo F19	Android 12 (API 31)			

Table 1
List of devices tested during the data collection campaigns.

Togo and Algeria have less features than those used in Finland and this points out also towards potential challenges of bringing accurate positioning feature in our LEDSOL system on the African market, where high-end devices are not supported. This also strongly emphasize the need of finding novel high-accuracy GNSS algorithms able to work with low-quality or noisy data and also in single-frequency single-system code-pseudorange-only modes.

2.2. Data collection

Several data collection campaigns have been organised to test the devices in different environments: one campaign in Europe (Tampere, Finland, development team location), and two campaigns in Africa (in Togo and Algeria, respectively, as the LEDSOL partners location). The goal is to provide more in-situ measurements, where the LEDSOL solution aims to be used. The GNSSlogger app developped by Google [20] was used to collect the data on the smartphones; the measurement campaigns took place in the January-April 2023 interval; the current GNSSlogger app version is v3.0.6.1, but also previous versions were used for data collected during Jan-Feb 2023. The raw GNSS data is stored every 1s. Several Android smartphone models were used, as explained in the next subsections, based on the available mobile phones in each team.

The raw GNSS observables can be harnessed from Android smartphones via various applications, such as GEO++ RINEX Logger [21, 22], GNSSlogger [20, 22, 21], Camaliot [23], etc. All such data is provided in Receiver Independent Exchange Format (RINEX) formats [24]. Such data include the code pseudoranges from one or several of the four GNSS systems (GPS, Galileo, Beidou, Glonass), carrier-to-noise ratio information, GPS times, Doppler information, possibly carrier-phase observables, etc. It is to be noticed that only the more expensive smartphones support both code and carrier-phase observables; the vast majority of the Android smartphones only support the code-pseudorange observables.

We have selected GNSSLogger app as the most suitable one for the LEDSOL project. It is to me mentioned that the current GNSSLogger variants are not able to extract and record the satellite broadcast ephemeris data, even if such data should be available deeper in the Android Application Programming Interface (API); however, in our opinion, the GNSSLogger is the most comprehensive app at the moment regarding raw GNSS data recordings and both the satellite broadcast ephemeris data and the precise orbit data can be accessible from various open-access sources such as IGS, NASA, etc., thus the processing and the statistical analysis can be done in

Figure 1: Static acquisition of data in Finland using a tripod(picture taken during February 2023.

an off-line manner.

For the ephemeris data, we have used open-access data: the Broadcast Ephemeris (BE), containing GNSS satellite's orbit and clock information as well as other essential parameters of the constellations for positioning was downloaded in our current work from the IGS's daily BE data through their HTTP portal [25]. The GNSS precise orbit and clock were downloaded from the IGS's ftp sever [26].

2.2.1. Data collection in Finland

An assessment of the device quality requires the definition of a reference and a survey protocol. The devices were tested in two scenario: (1) Open-sky static acquisition; (2) Dynamic pedestrian trajectory in mixed open-sky and urban canyoning. For the static scenario, the smartphones are placed on a tripod over a known landmark (Figure 1). For the dynamic scenario, a reference receiver (i.e. Novatel PwrPak7 receiver, Novatel GNSS-850 antenna) in a backpack setup is used. For high-precision, differential GNSS processing is performed with a base station located on TAU rooftop. Figure 2 provides a summary of the setup. GNSS data was logged in raw and RINEX format, on the devices listed in Table 1. The measurement campaigns have been on-going since February 2023 and were conducted by the team member as well as by volunteer students at Tampere University. A preliminary analysis of the surveys is presented in Section 3.

2.2.2. Data collection in Togo and Algeria

Measurements were conducted in a walking environment near the University of Lomé in Togo, during March 2023 with an Itel P36 phone with Android 10, supporting only GPS and Glonass

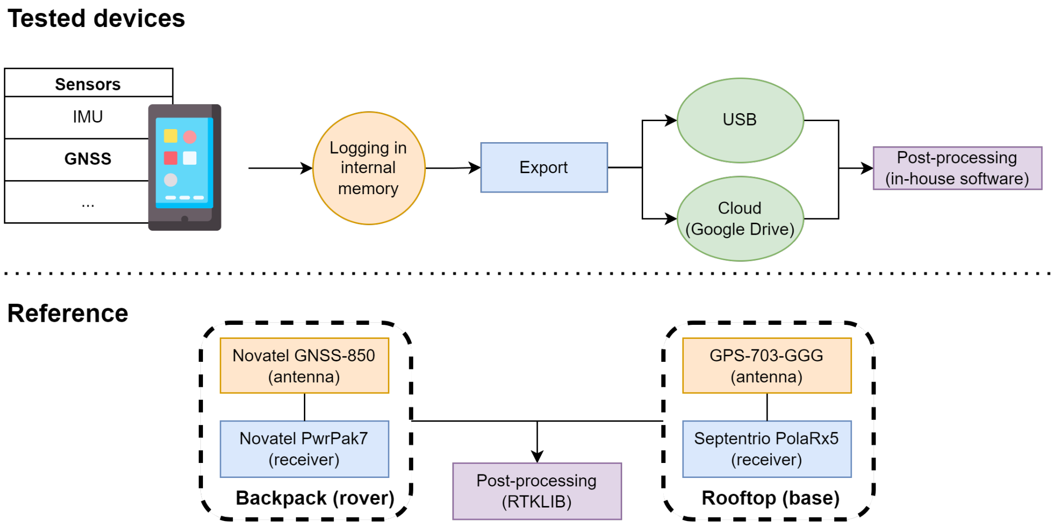

Figure 2: Diagram of the Finland data collection campaign.

Figure 3: Pictures of the Togo environments: left: Lab of Applied Hydrology and Environment of University of Lome; right: place of recording the analyzed data from Togo, near the university.

code-phase measurements. Examples of the environment where data was collected in Togo are shown in Figure 3.

The data in Algeria was collected during January and February 2023, also in a walking environment near several water sources. The duration of the individual measurements was shorter than the duration of the individual measurements in Togo. The measurements in Algeria were conducted with an Oppo F19 smartphone with Android 12 (API 31) on it. Examples of the environment where data was collected in Algeria are shown in Figure 4.

2.3. In-house data-analysis and post-processing software

As above mentioned, data pseudorange data was collected with GNSSlogger app from Android phones and this data requires post processing in order to form the final positioning solutions. For this post-processing stage we have developped an in-house simulator, as the existing post-processing tools have some limitations and were not fit for our purpose (e.g., error modeling,

Figure 4: Pictures of the environments where data was collected in Algeria near a water source/ water well.

comparison of various positioning algorithms in single and multi-system mode, etc.). The Matlab simulation environment was the platform of our choice due to its flexibility and versatility of use and its possibility of working with modular structures (presented later, in the description of Fig. 5), easy debugging of errors, manifold plotting support, and the availability of a navigation toolbox that eased some of the in-house algorithmic post processing of data collected from the Android mobile phones.

A broad review of existing software to process raw GNSS data measurements has also identified the following software tools available in Matlab environment: PPPH [27, 28], goGPS [29], and GPSTools (GT) [30]. All three of them focus on multi-GNSS data analysis; the goGPS focuses on Single Point Positioning (SPP), while PPPH and GT focus on Precise Point positioning (PPP) analysis. For low-cost implementations, possibly with incomplete or partially available data, the SPP is the positioning method of interest in this paper. As the goGPS tool was developped for Matlab version 2016, it is currently not supported by newer Matlab releases (such as Release 2022b used in our work); therefore, we have opted for developing own in-house modular simulator, following the flowchart shown in Fig. 5. The current implementation uses for positioning only the code pseudorange on the first frequency point for each constellation, namely L1 frequency for GPS, G1 frequency for Glonass, B1 frequency for BeiDou, and E1 frequency for Galileo. As most smartphones on the market support only single-frequency measurements, this first implementation works well with all the acquired data.

The accurate estimation of the positions of the visible satellites on the sky (shortly referred to as orbit determination) is the first step towards forming a position estimate and this is one part of or post-processing stage. As above-mentioned, the information about the satellite position can be found in the broadcast ephemeris or in the precise-orbit-determination data files.

The error modeling is another part in our post-processing stage. The errors in satellite-based positioning can be typically classified into three parts, satellite clock side, atmospheric delays, and receiver clock side. The receiver clock biases are typically estimated together with the position estimate, directly for each involved constellation; these estimates also absorb the hardware delay of receiver. However, the satellite clock errors as well as the atmospheric delays need to be estimated and compensated before using the measured pseudoranges to form a position estimate.

In the absence of a highly accurate reference position data, we use the National Marine

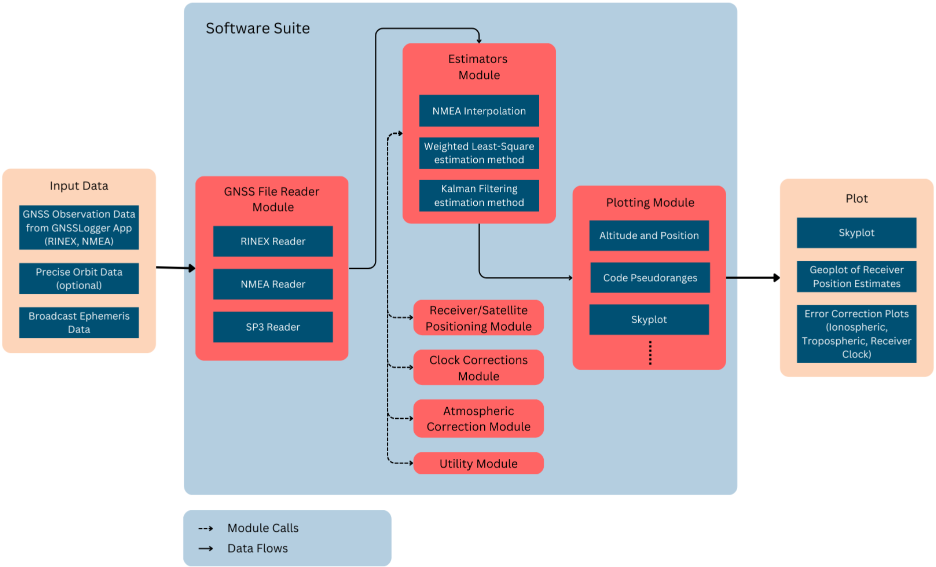

Figure 5: The adopted methodology in our in-house simulator under development.

Electronics Association (NMEA) estimates of the Android phones as the “benchmark” positions. The NMEA estimates are computed with the smartphone intrinsic positioning engine, by integrating all available sensors (e.g., GNSS, cellular, WiFi, accelerometers, gyroscopes, etc.). While such estimates are not error free, especially in urban and indoor scenarios, we believe that they can provide a good benchmark of comparison with our own positioning algorithms, developped in the post-processing software. With our post-processing software we can also characterize the error distributions of various errors in the transmitter-receiver chain, namely the transmitter and receiver clock errors, the ionospheric delay, the tropospheric delays, and the additional residual errors (multipath, interferences, etc.).

Parts of the collected data will also have a higher-accuracy reference, coming from a professional receiver (see Section 2.2.1); such data has been already collected but it has not yet been analyzed and it is a topic of future research.

2.4. Error modeling

The raw code pseudoranges ρ_s measured by an Android smartphone in position (x, y, z) (in WGS84 coordinate system) for each visible satellite s on sky can be modelled as

$$\rho_s = \sqrt{(x - x_s)^2 + (y - y_s)^2 + (z - z_s)^2} + \epsilon_{tx} + \epsilon_{iono} + \epsilon_{tropo} + \epsilon_{mp} + c\Delta t \tag{1}$$

where (x_s, y_s, z_s) are the 3D coordinates (in WGS84 coordinate system) of the s-th visible satellite on sky, ϵ_{tx} are the transmitter clock and orbital errors (in m), ϵ_{iono} are the ionospheric-delay errors (in m), ϵ_{tropo} are the tropospheric-delay errors (in m), ϵ_{mp} are the multipath errors (in m) and Δt is the receiver-transmitter clock bias (i.e., errors due to the receiver clock).

The biggest part in the transmitter clock errors is the satellite clock bias, which is the bias that the transmit time drift from the true GNSS time; it can reach more than 100 km for the pseudorange error. Our simulator uses three parameters from the broadcast ephemerides to fit it for GPS, Galileo, and Beidou system. The Glonass system is different, in the sense that it only applies one parameter for a 30-minute interval of its BE. The accuracy of GPS BE's clock is about 5 ns, corresponding about 1.5 meter error in pseudorange. The Time Group Delay (TGD) is the hardware delay at the satellite side; the BE files provide two parameters to correct it for two frequencies' delay, for GPS, Galileo, and Beidou systems. The Glonass system does not broadcast this parameter. The maximum value of TGD is 20 ns; if one ignores it, will produce a 4-5 meters error in pseudo-range correction.

The relativistic clock correction is also a correction included in our simulator and it is due to the relativistic effects caused by the high speeds of the GNSS satellites on sky. Only a periodical component of relativistic clock correction which is caused by the orbit eccentricity should be applied; ignoring this correction can produce a positioning error up to 13-meter level.

The ionospheric delay errors are typically among the largest sources of errors in GNSS-based positioning (together with multipath errors) and due to the passage of the GNSS signal through the ionospheric layer. The ionosphere is the layer above the Earth from about 45 km altitude to about 300 km altitude. Due to the random movement of the charged particles (electrons) within this layer, the signal coming from the satellite to a ground receiver suffers random delays; these are called ionospheric delays. According to [31] almost 50% of GPS positioning errors in African equatorial and low-latitude regions during the night-time were linked to ionospheric delay errors.

The most encountered ionospheric delay model is a first-order model, where the ionospheric delay (in seconds) depends on TEC - a random variable fluctuating according to the season, day of the year, and solar activity - and the carrier frequency f_c (e.g-, this is about 1.5 GHz for L1 GPS and E1 Galileo measurements.

Our simulator chose the Klobuchar ionospheric model corrections for all the GNSS systems and the Klobuchar model parameters are provided by GPS's BE. One consideration for choosing this model is that these parameters of ionospheric model have a correlation with TGD parameters and the Klobuchar model is the official model for GPS. Additionally, the other system's time biases in receiver side are estimated respect to GPS one.

The tropospheric delay errors are due to the passage of the GNSS signal through the tropospheric layer. The troposphere is the layer above the Earth from about 8 km altitude to about 14 km altitude. In our simulator, the tropospheric corrections are based only on the dry/hydrostatic Saastamoinen model, as these were found in the literature to be the most accurate for the wet-delay part of the tropospheric delay corrections [32]. In addition, it is known [33] that the dry-delay part accounts for 90% of the overall tropospheric delays, with the wet-delay part accounting for only 10% of the delay. Future steps will include also adding the wet-delay part in our tropospheric model. The input parameters for this model include the azimuth and elevation angle of each visible satellite on sky, the receiver location in latitude and longitude, and the height of receiver.

Most typical, all other residual errors are lumped together into an Additive White Gaussian Noise (AWGN) model.

2.5. Single Point Positioning

A Weighted Least Squares (WLS) SPP estimator was implemented for 3D position estimate; the WLS model is as follows [34]:

$$X = (G^T W G)^{-1)} G^T W y \tag{2}$$

where X is the vector with the estimation parameters (namely the x, y, z receiver coordinates in WGS84 reference coordinate frame and the clock bias of receiver for each included constellation). Above, W is a weights matrix of the form $diag(\sigma_1^{-2}, \sigma_1^{-2}, \ldots, \sigma_{N_{SV}}^{-2})$ with N_{SV} being the number of visible satellites on the sky and $\sigma_i^2, i = 1, \ldots, N_{SV}$ being the noise variance for the i-th visible satellite; the estimated noise variances can be extracted from the measured C/N_0 values from the collected RINEX observables. Measurements at low C/N_0 tend to have both a larger multipath error and a larger error due to atmospheric (tropospheric and ionospheric) delays. Also above, G is the satellite geometry matrix, built based on the receiver-satellite geometry (i.e., satellite positions on sky and the receiver estimated positions). The pre-fit residual, y, is the measured pseudo-range minus the distance between satellite and receiver, and minus all the corrections due to tropospheric and ionospheric delays, and clock biases of each satellite. As above-mentioned, we modeled the clock bias of satellite based on the relativity effect and the TGD. The GPS system's clock bias for the receiver is estimated directly; the other systems' clock biases (Galileo, Beidou, Glonass) are estimated as offsets respect to the GPS one.

The WLS algorithm in our simulator separate into two stages at each time step. First, the initial position and clock can be set to zero or to a rough value, the ionospheric delay and tropospheric delay are also initially set to zero. Then, we estimate the initial position and clock biases for receiver, including the ionospheric and tropospheric delays. The second stage uses the initial estimated parameters to load the full error model, with the rough position value fed into WLS iteration again, in order to get the position and clock bias of receiver. This two-stage algorithm is very useful in the context of low-cost receiver processing as targeted in LEDSOL project; the clock of the GNSS chipsets on a low-cost receiver are typically not stable; they have some drifts and unexpected clock jumps; therefore the prior of the estimated parameters suffer of high errors.

As a side note, our current simulator also includes an Extended Kalman Filter SPP estimate (see Figure 5), but since this part still needs to be tested and enhanced, results based on EKF are not included in this work-in-progress paper.

3. Preliminary results

Preliminary results are shown in the next sections in terms of skyplots, error correction models, and SPP estimates. A skyplot is showing the elevation versus azimuth angles of all the satellites visible on the sky during a measurement duration. If the measurements last more than few seconds, each satellite will be represented as a collection of points at various times, showing basically the satellite movement on sky. All analyzed data in here was collected under walking or static conditions.

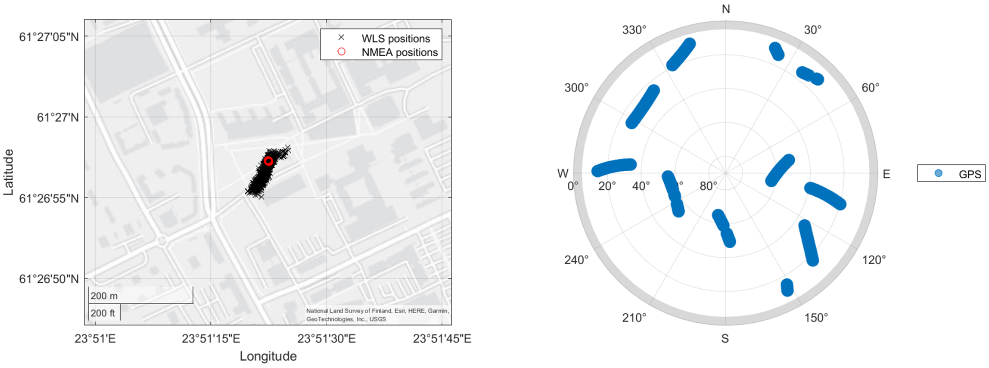

Figure 6: Left: track data; Right: skyplot. Tampere measurements with Google Pixel 7 Android 13 smartphone; data collected during 17.02.2023 in static point.

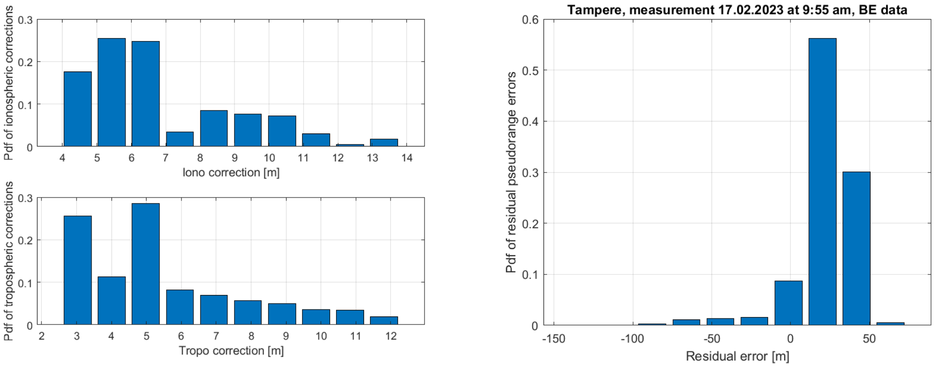

Figure 7: Left: estimates of the ionospheric and tropospheric errors [m]; Right: Residual errors (multipath, interferences, noises) [m]. Tampere measurements with Google Pixel 7 Android 13 smartphone; data collected during 17.02.2023 in static point.

3.1. Results based on Tampere data

The results based on the measurements at fixed point (with a Google Pixel 7 phone installed on the tripod shown in Fig. 1) conducted in Tampere on 17.02.2023 are shown in Figure 6 (the measured track and the sky plot based on visible GPS satellites), Figure 7 (the ionospheric and tropospheric corrections as well as the residual errors), and Figure 8 (the code-phase and carrier-phase uncorrected ranges as reported in the RINEX data collected with the GNSSlogger app). The static data was collected during about 45' (corresponding to 2704 measurement points, spaced at 1s between consecutive measurements).

3.2. Results based on Togo data

The results based on the data collected in Togo so far are illustrated in Figure 9 (the measured track and the sky plot based on visible GPS satellites), Figure 10 (the uncorrected and corrected pseudoranges as well as the difference between BE-based and SP3-based estimates), and Figure 11 (the ionospheric and tropospheric corrections as well as the residual errors).

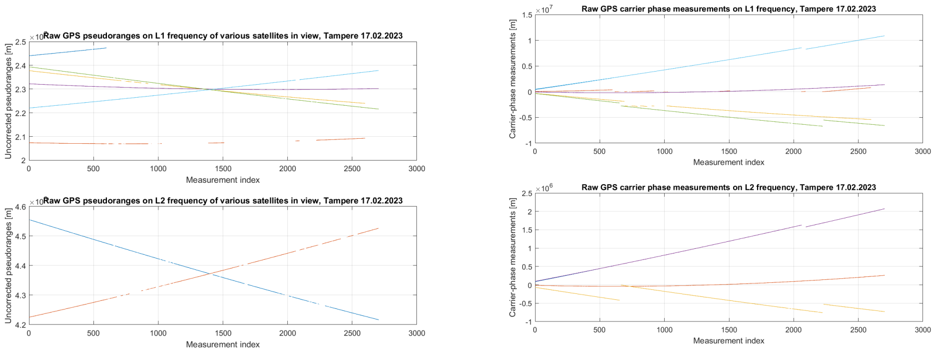

Figure 8: Left: code pseudoranges [m]; Right: carrier-phase pseudoranges [m]. Tampere measurements with Google Pixel 7 Android 13 smartphone; data collected during 17.02.2023 in static point.

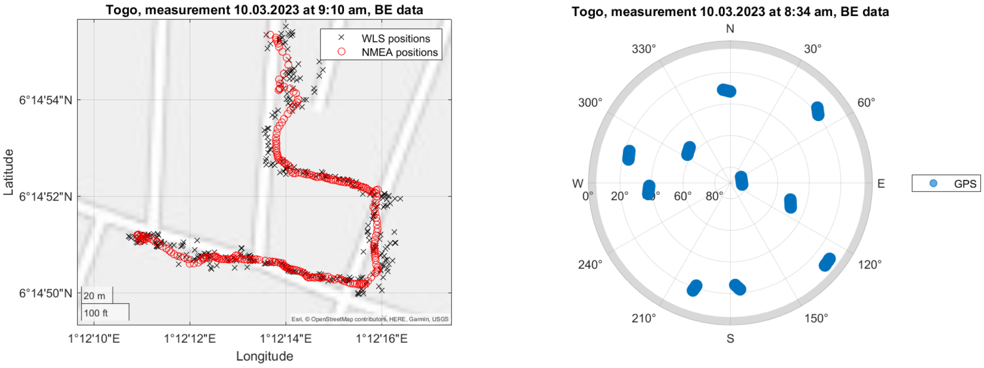

Figure 9: Left: track data; Right: skyplot. Togo measurements with Itel P36 Android 10 smartphone

3.3. Results based on Algeria data

The results based on the data collected during 14th of February in Algeria are illustrated in Figure 12 (the measured track and the sky plot based on visible GPS satellites), Figure 13 (the uncorrected and corrected pseudoranges as well as the difference between BE-based and SP3-based estimates), and Figure 14 (the ionospheric and tropospheric corrections as well as the residual errors). The data collected in Algeria, near a water source called Aïn Tagouraït, were of shorter duration than those collected in Togo.

By comparing Togo and Algeria data, one can see that the differences between BE-based estimates and precise-orbit/SP3-based estimates are of the same order of magnitude of 3-4 m. The ionospheric and tropospheric corrections in Algeria are about half of the magnitude of those in Togo, but the residual errors in Algeria are about double in magnitude compared to those in Togo, which may point to the fact that the current ionospheric and tropospheric models are not accurate enough for Togo data.

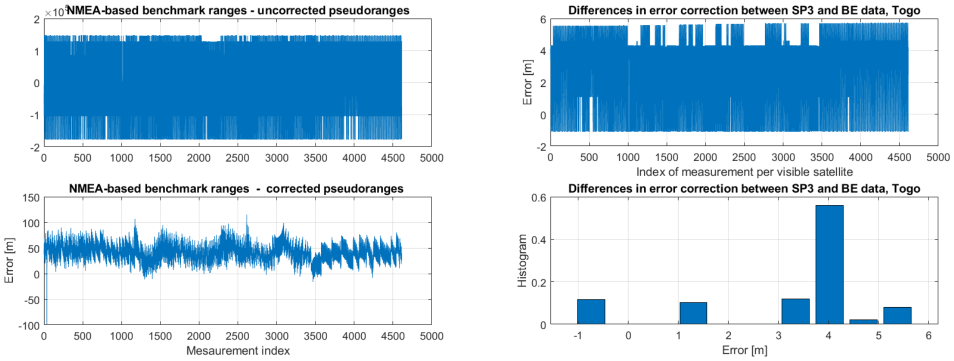

Figure 10: Left: uncorrected and corrected pseudorange errors [m] with respect to the NMEA-based benchmark; Right: differences in meters between the corrections based on the broadcast ephemeris (based on BE data) and the precise orbit data (based on SP3 data). Togo measurements with Itel P36 Android 10 smartphone

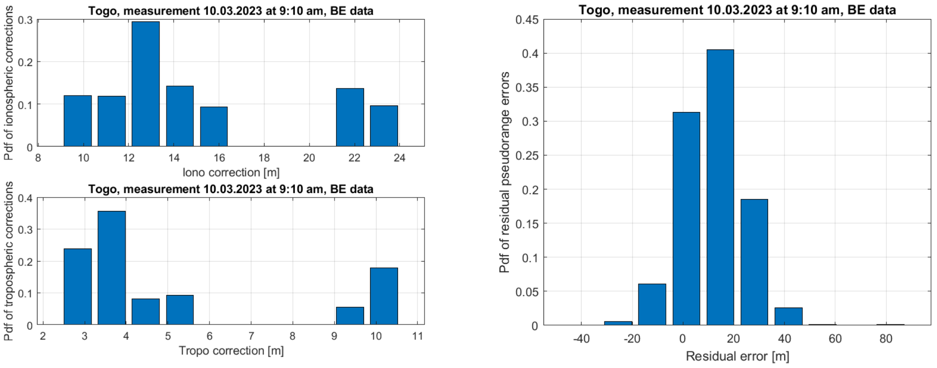

Figure 11: Left: estimates of the ionospheric and tropospheric errors [m]; Right: Residual errors (multipath, interferences, noises) [m]. Togo measurements with Itel P36 Android 10 smartphone

4. Open challenges and future steps

There is currently a wide variety of Android smartphones on the market with the ability to collect and store raw GNSS data from single or multiple GNSS systems, for single or dual frequencies, and from code and carrier-phase measurements. The noise level in collecting such raw data is typically high, as also our preliminary data analysis showed and there are multiple challenges to be overcome towards acquiring accurate positioning with such data. First, the vast majority of Android smartphones currently support only code-phase measurements; when carrier-phase measurements are also available, they are typically stored in a partially overlapping mode with the code-phase ones for the sake of keeping battery consumption to low levels; it is however expected that combined code-plus-carrier-phase measurements would reach significantly higher performance than code-phase measurements alone. Secondly, we have noticed that different smartphone devices collecting data simultaneously may see a different number of satellites on sky, based on the GNSS chipsets they are using.

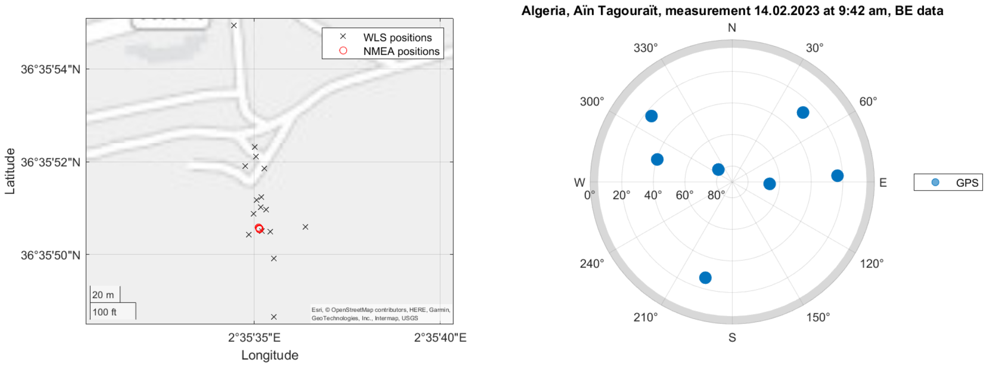

Figure 12: Left: track data; Right: skyplot. Algeria measurements with Oppo F19 Android 12 smartphone.

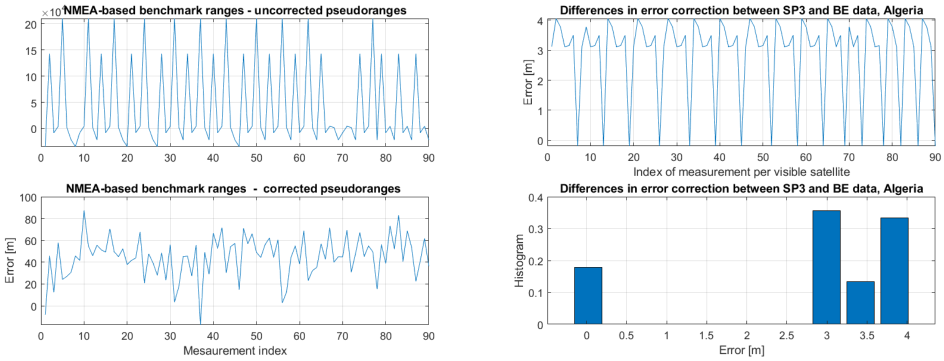

Figure 13: Left: uncorrected and corrected pseudorange errors [m] with respect to the NMEA-based benchmark; Right: differences in meters between the corrections based on the broadcast ephemeris (based on BE data) and the precise orbit data (based on SP3 data). Algeria measurements with Oppo F19 Android 12 smartphone.

No current open-access app, to the best of the Authors' knowledge can extract the navigation data (meaning the satellite broadcast ephemeris) despite the fact that few Android smartphones on the market do support such access to the data, thus novel Android app developments are necessary to support such full data extraction.

During our measurement campaigns, we have also noticed some frequently updated output format of data in GNSSlogger app (e.g., at least twice during past six months) that also require frequent updates in the data reading and processing software. In addition, we have also noticed slightly different formatting of the RINEX data stored by the GNSSlogger app on different mobile devices (e.g., Oppo smartphone stored data in a slightly different format compared to the other tested Android phones, and this also required updates in our in-house software).

To sum up, there is still high need of versatile and user friendly software both for the extraction of full raw data from Android smartphones and for processing and analyzing the raw GNSS data, as well as for improving the positioning accuracy, especially in the presence of noisy or incomplete data.

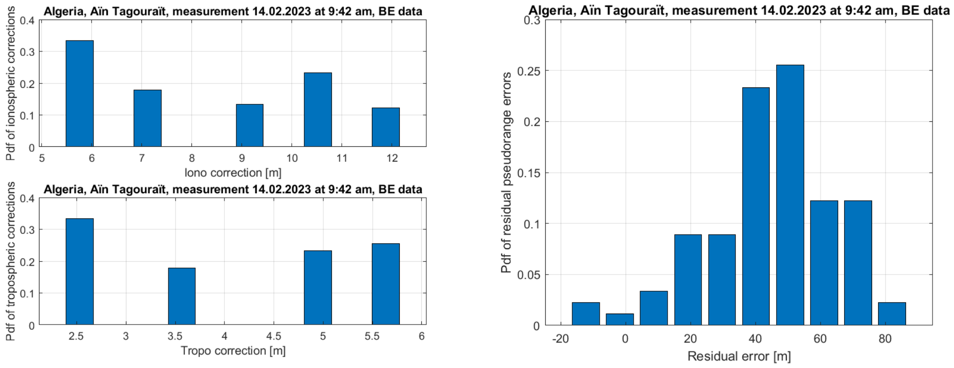

Figure 14: Left: estimates of the ionospheric and tropospheric errors [m]; Right: Residual errors (multipath, interferences, noises) [m]. Algeria measurements with Oppo F19 Android 12 smartphone.

The next two subsections details the future steps in our work pertaining to collecting and analyzing further the GNSS raw data.

4.1. Novel Android app developments

The measurement campaigns showed the limitations of the apps currently available today for sensor data surveys. While many apps exist for GNSS measurements (e.g. GeoRINEX or GNSSlogger [20]), they do not work on all Android platforms or provide an updated version of their source code, preventing modifications for our purposes. There are two solutions to this challenges: one is to extend some of the open-source applications – that are available to collect smartphone data from their sensors– with new raw GNSS capabilities; a second approach, currently on-going, is to develop own specific app to allow all onboard sensors to be logged. An initial and very preliminary version of this app is available on GitHub [35] as an open-source towards for other usage by other research teams.

4.2. Matlab-based software developments

Future steps will include the support for dual-frequency measurements, e.g., L1-L5 frequencies for GPS and E1-E5 frequencies for Galileo, implementing code-plus-carrier-phase positioning for data originating from smartphones supporting dual carrier-phase measurements, implementing lower-complexity/lower-power variants of SPP with partial information/missing data, extending our statistical analysis of the data collected under various scenarios, and collecting more data under various scenarios. In addition, referenced data with positioning coming also from a professional receiver will also be included in the Tampere data-based analysis. When our in-house Matlab-based simulator becomes more mature, it is also planned to be offered in open access to the research community.

Acknowledgments

This work has been supported by the LEDSOL project (https://www.leap-re.eu/ledsol/) funded within the LEAP-RE programme by the European Union's Horizon 2020 Research and Innovation Program under Grant Agreement 963530. This work has been done in collaboration with some of the APROPOS project team members, therefore the authors also gratefully acknowledge funding from European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska Curie grant agreement No. 956090 (APROPOS: Approximate Computing for Power and Energy Optimisation, http://www.apropos-itn.eu/). The Authors would also like to thank the Academy of Finland (project 352364), the Algerian Ministry of Higher Education and Scientific Research (MESRS) (project 31), and the Federal Ministry of Education and Research in Germany, for their support. The Authors also acknowledge a grant of the Romanian Ministry of Research, Innovation and Digitization, CNCS/CCCDI - UEFISCDI, project number COFUND-LEAP-RE-LEDSOL, within PNCDI III. We would also like to thank the following students at Tampere University who have helped with the software developments and/or data collection: Max Mecklin, Umair Raihan, Silja Nahkala, Heini Vesaranta, Henry Andersson, Petrus Jussila, Salla Rouhiainen, Severi Ruusumaa, My Nguyen, and Ha Chu.

References

[1] T. Verheyde, A. Blais, C. Macabiau, F.-X. Marmet, Analyzing android gnss raw measurements flags detection mechanisms for collaborative positioning in urban environment, in: 2020 International Conference on Localization and GNSS (ICL-GNSS), 2020, pp. 1–6. doi:10.1109/ICL-GNSS49876.2020.9115564.

[2] J. Wang, D. Weng, X. Qu, W. Ding, W. Chen, A novel deep odometry network for vehicle positioning based on smartphone, IEEE Transactions on Instrumentation and Measurement 72 (2023) 1–12. doi:10.1109/TIM.2023.3240227.

[3] M. Fortunato, A. Mazzoni, New opportunities for mass-market applications of real-time variometric velocity estimated using android gnss raw measurements, in: 2020 European Navigation Conference (ENC), 2020, pp. 1–11. doi:10.23919/ENC48637.2020.9317397.

[4] N. Sikirica, E. Malić, I. Rumora, R. Filjar, Exploitation of google gnss measurement api for risk assessment of gnss applications, in: 2017 25th Telecommunication Forum (TELFOR), 2017, pp. 1–3. doi:10.1109/TELFOR.2017.8249343.

[5] A. Rustamov, A. Minetto, F. Dovis, Improving gnss spoofing awareness in smartphones via statistical processing of raw measurements, IEEE Open Journal of the Communications Society 4 (2023) 873–891. doi:10.1109/OJCOMS.2023.3260905.

[6] D. Miralles, D. M. Akos, D.-K. Lee, A. Konovaltsev, L. Kurz, S. Lo, Robust satellite navigation in the android operating system using the android raw gnss measurements engine and location providers, in: 2020 European Navigation Conference (ENC), 2020, pp. 1–12. doi:10.23919/ENC48637.2020.9317434.

[7] M. Kurum, M. M. Farhad, A. C. Gurbuz, Integration of smartphones into small unmanned aircraft systems to sense water in soil by using reflected gps signals, IEEE Journal of

Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021) 1048–1059. doi:10.1109/JSTARS.2020.3041162.

[8] A. V. Sokolov, D. A. Kogogin, D. S. Maximov, I. A. Nasyrov, A. V. Shindin, R. V. Zagretdinov, Comparison study of technical characteristics of low-cost multi-band gnss modules in solving the problem of measuring the total electron content of the ionosphere, in: 2021 Radiation and Scattering of Electromagnetic Waves (RSEMW), 2021, pp. 300–303. doi:10.1109/RSEMW52378.2021.9494015.

[9] Sustainable development goal 6 on clean water and sanitation, United Nations; https://southafrica.un.org/en/sdgs/6, accessed 19.2.2023, 2023.

[10] R. Caceres, A. Gonzalez, G. Pereira, L. Rios, R. Rios, V. Oxilia, Systematic literature review of the water-energy- food nexus: An overview of the field, in: 2020 IEEE Congreso Bienal de Argentina (ARGENCON), 2020, pp. 1–8. doi:10.1109/ARGENCON49523.2020.9505529.

[11] M. L. de Lazaro Torres, P. Borderias Uribeondo, F. J. Morales Yago, Citizen and educational initiatives to support sustainable development goal 6: Clean water and sanitation for all, Sustainability 12 (2020) 2073. URL: http://dx.doi.org/10.3390/su12052073. doi:10.3390/su12052073.

[12] Sustainable development goal 7 on affordable and clean energy, United Nations; https://southafrica.un.org/en/sdgs/7, accessed 28.4.2023, 2023.

[13] EU-funded LEDSOL project, "Enabling clean and sustainable water through smart UV/LED disinfection and SOLar energy utilization", United Nations; https://www.leap-re.eu/ledsol/, accessed 1.6.2023, 2022.

[14] I. Mocanu, O. Cramariuc, K. Bierwirth, H. Lebik, B. Kodom, M. Mecklin, Fostering transnational research, innovation, and education trough an international Europe and Africa collaboration project, in: ICERI 2022 Proceedings, IATED, 2022. URL: https://doi.org/10.21125/iceri.2022.1264. doi:10.21125/iceri.2022.1264.

[15] O. Cramariuc, H. Lebik, T. Kodom, I. Mocanu, K. Bierwirth, E. Lohan, The importance of safe water and its perception by students in europe and africa as part of a transnational research project, in: Accepted at EDULEARN23 International Education Conference, 2023.

[16] F. Van Diggelen, "A-GPS: Assisted GPS, GNSS, and SBAS", Artech Books, 2009.

[17] G. Li, J. Geng, B. Chu, High-precision velocity determination using mass-market android gnss measurements in the case of anomalous clock variations, GPS Solutions 27 (2023) 1–16.

[18] S. Barbeau, Dual-frequency GNSS on Android - Table of devices, [Online, accessed 5.5.2023] https://barbeau.medium.com/tl-dr-dual-frequency-gnss-on-android-table-of-devices-9be4bbb83a7b, 2019.

[19] S. Barbeau, GPSTest Database, [Online] https://docs.google.com/spreadsheets/d/1jXtRCoEnnFNWj6_oFlVWflsf-b0jkfZpyhN-BXsv7uo/, 2021.

[20] Google GNSS logger App, https://github.com/google/gps-measurement-tools, accessed 28.04.2023, 2016.

[21] GNSS Raw Measurements Workshop: from research to commercial use, https://www.euspa.europa.eu/newsroom/european-space-expo/gnss-raw-measurements-workshop-research-commercial-use, accessed 28.04.2023, 2018.

[22] F. Zangenehnejad, Y. Jiang, Y. Gao, GNSS observation generation from smartphone android

location API: Performance of existing apps, issues and improvement, Sensors 23 (2023) 777. URL: https://doi.org/10.3390/s23020777. doi:`10.3390/s23020777`.

[23] ESA news - CAMALIOT Project Launches Mobile App for Raw GNSS Data Collection through Crowdsourcing, https://gssc.esa.int/news/camaliot-project-launches-mobile-app-for-raw-gnss-data-collection-through-crowdsourcing/, accessed 28.04.2023, 2022.

[24] RINEX data format description, https://igs.org/wg/rinex/, accessed 24.04.2023, 2023.

[25] Broadcast ephemeris data in open access, ftp://ftp.gfz-potsdam.de/pub/GNSS/products/mgex/, accessed 24.04.2023, 2023.

[26] Precise orbit data in open access, https://igs.bkg.bund.de/root_ftp/IGS/BRDC/, accessed 24.04.2023, 2023.

[27] B. Bahadur, M. Nohutcu, A MATLAB-based software for multi-GNSS precise point positioning analysis, https://geodesy.noaa.gov/gps-toolbox/PPPH.htm, accessed 28.4.2023, 2023.

[28] B. Bahadur, M. Nohutcu, PPPH: a MATLAB-based software for multi-GNSS precise point positioning analysis, GPS Solutions 22 (2018). URL: https://doi.org/10.1007/s10291-018-0777-z. doi:`10.1007/s10291-018-0777-z`.

[29] M. Reguzzoni, E. Realini, et al., GoGPS - an advanced GNSS observation processing software, available at https://github.com/goGPS-Project/goGPS_MATLAB and https://gogps-project.github.io/about/, 2023.

[30] GpsTools (GT): GPS/GNSS Precise Analysis Software, https://gpspp.sakura.ne.jp/gpstools/gt_release.htm, accessed 28.4.2023, 2023.

[31] O. E. Abe, B. Adeyemi, O. Ogunmodimu, I. Emmanuel, E. Oluwadare, T. Oluwadare, Characterizing the accuracy of global positioning system for single point positioning at african low-latitude region, Journal of Spatial Science 68 (2021) 139–154. URL: https://doi.org/10.1080/14498596.2021.1934577. doi:`10.1080/14498596.2021.1934577`.

[32] B. Chen, Z. Liu, A comprehensive evaluation and analysis of the performance of multiple tropospheric models in china region, IEEE Transactions on Geoscience and Remote Sensing 54 (2016) 663–678. doi:`10.1109/TGRS.2015.2456099`.

[33] A. Leick, "GPS Satellite Surveying", Wiley-Interscience Publication, USA, 1994.

[34] A. Grenier, Development of a GNSS positioning application under Android OS using GALILEO signals, Msc thesis, Ecole Nationale de Sciences Geographiques, 2019.

[35] A. Kuittinen, H. Vesaranta, H. Andersson, P. Jussila, S. Rouhiainen, S. Ruusumaa, S. Gustafsson, A. Grenier, MIMIR TAU- an open-source app designed to logged various sensors data in smart devices , Github [Online, accessed 31.5.2023] https://github.com/agrenier-gnss/MimirTAU, 2023.