# ANALYSIS AND CONDITIONING OF GNSS MEASUREMENTS FROM SMARTPHONES FOR PRECISE POINT POSITIONING IN REALISTIC ENVIRONMENTS

GANGA SHINGHAL

A THESIS SUBMITTED TO FACULTY OF GRADUATE STUDIES IN

PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

GRADUATE PROGRAMME IN EARTH AND SPACE SCIENCE

YORK UNIVERSITY

TORONTO, ONTARIO

SEPTEMBER 2020

# Abstract

With the availability of raw GNSS code and carrier-phase measurements from dual-frequency chips in smartphones, Precise Point Positioning (PPP) can be used to improve the accuracy of the positioning solution offered by them, without any additional reference infrastructure.

In realistic applications in suburban and urban environments, with the smartphone in hand or on the dashboard of a car, there are numerous challenges with smartphone data such as missing measurements, poor multipath suppression and low and irregular carrier-to-noise-density ratio. The measurements are analysed and conditioned or pre-processed by implementing a prediction technique for filling in data gaps and a $C/N_0$-based stochastic model for assigning realistic *a priori* weights to the measurements in the PPP processing engine. Finally, the post-processed PPP solution accuracy and availability have been compared with the positioning solution obtained from other GNSS positioning techniques. After conditioning, the smartphone PPP positioning solution is seen to have nearly 100% solution availability and 50% more accuracy.

# Acknowledgements

I would like to express my sincere gratitude towards my supervisor Professor Sunil Bisnath, for constantly guiding, supporting, motivating and inspiring me throughout the research. Right from his classes to regular meetings, he has helped me understand the concepts and fundamentals of GNSS positioning and navigation and guided me through each step of this research, and through different conferences and presentations. I would also like to express my gratitude towards Professor Jian-Guo Wang for his valuable feedback and input during the research as a committee member and as a course professor. I also highly appreciate the inputs and help offered by my past and present laboratory colleagues: John Aggrey, Sudha Vana, Nacer Naciri, Sihan Yang, Ding Yi and Ivan Mugatu for helping with data collection and with the data analysis and processing. I would also like to acknowledge IGS, CNES, GFZ and DLR for providing antenna, clock and orbital products for processing the data and Natural Sciences and Engineering Research Council of Canada (NSERC) for providing funding.

# Table of Contents

# List of Tables

# List of Figures

# List of Acronyms

| | |
|---|---|
| **ANTEX** | Antenna Exchange Format |
| **ASCII** | American Standard Code for Information Interchange |
| **C/A code** | Coarse/Acquisition code |
| **CDMA** | Code Division Multiple Access |
| **BeiDou** | China Navigation Satellite System |
| **BPSK** | Binary Phase-Shift Keying |
| **$C/N_0$** | Carrier-to-Noise-Density Ratio |
| **dBHz** | decibel Hertz |
| **DCB** | Differential Code Bias |
| **DLL** | Delay Lock Loop |
| **DLR** | German Aerospace Centre |
| **DoD** | Department of Defense |
| **FDMA** | Frequency Division Multiple Access |
| **Galileo** | European Satellite Navigation System |
| **GFZ** | German Research Centre for Geosciences |
| **GLONASS** | Russian Navigation Satellite System |
| **GNSS** | Global Navigation Satellite System |
| **GPS** | Global Positioning System |
| **INS** | Inertial Navigation System |
| **IMU** | Inertial Measurement Unit |
| **IGS** | International GNSS Service |
| **IONEX** | IONosphere map EXchange |
| **MGEX** | Multi-GNSS Experiment |
| **NAVIC** | Indian Regional Navigation Satellite System |
| **NMEA** | National Marine Electronics Association |
| **LHCP** | Left hand circularly polarized |
| **NRCan** | Natural Resources Canada |

| | |
|---|---|
| **PPP** | Precise Point Positioning |
| **PRN** | Pseudo Random Noise |
| **RHCP** | Right hand circularly polarized |
| **RINEX** | Receiver Independent Exchange |
| **rms** | root mean square |
| **RTK** | Real-Time Kinematic |
| **SNR** | Signal-to-Noise Ratio |
| **SPP** | Single Point Positioning |
| **std** | standard deviation |
| **TEC** | Total Electron Content |
| **QZSS** | Quasi-Zenith Satellite System |
| **YorkU** | York University |
| **ZPD** | Zenith path delay |

# List of Symbols

| | |
|---|---|
| $\alpha$ | dimensionless DLL discriminator correlator factor |
| $\boldsymbol{a}$ | rms of the pseudorange multipath noise for code measurements |
| $b_{+}^{*}$ | hardware biases (m) |
| $\boldsymbol{\beta}$ | carrier phase noise and multipath error |
| $\boldsymbol{b}$ | pseudorange chipping length of C/A code or carrier-phase wavelength used in stochastic modelling |
| $\boldsymbol{B_L}$ | equivalent code loop noise bandwidth |
| $\boldsymbol{c}$ | speed of light ($ms^{-1}$) |
| $\Delta\delta_j^i(t)$ | clock offsets of the receiver and the satellite clock |
| $\boldsymbol{dt}$ | receiver clock error (s) |
| $\boldsymbol{dT}$ | satellite clock error (s) |
| $d_{orb}$ | satellite orbit error (m) |
| $\boldsymbol{d_{multi(P_{L_i})}}$ | pseudorange multipath effect on L1 or L2 or L5 (m) |
| $\boldsymbol{d_{multi(\Phi_{L_i})}}$ | carrier-phase multipath on L1 or L2 or L5 (m) |
| $\boldsymbol{D^{sat}}$ | measured Doppler shift for generic satellite (Hz) |
| $\boldsymbol{d_{ion,L_i}}$ | delay caused by the ionosphere |
| $\boldsymbol{d_{trop}}$ | delay caused by the troposphere |
| $\boldsymbol{\epsilon(P_{L_i}^{GNSS})}$ | pseudorange measurement noise (m) |
| $\boldsymbol{\epsilon(\varphi_{L_i}^{GNSS})}$ | carrier-phase measurement noise (m) |
| $\boldsymbol{el_k^m}$ | satellite elevation angle |
| $\boldsymbol{f_{1\|2\|5}}$ | carrier-frequency of L1, L2 or L5 (Hz) |
| $\boldsymbol{f_{clockkdrift}}$ | Doppler caused by the clock drift of the receiver |
| $\boldsymbol{f_{motion}}$ | Doppler caused by the relative motion of satellite and receiver |

| | |
|---|---|
| $f_{PLL}$ | Doppler measurement obtained by carrier tracking loop |
| $f_{pred}$ | predicted Doppler of the channel |
| $\eta$ | pseudorange noise and multipath error |
| $P_{L_j}$ | measured pseudorange on L1 or L2 or L5 (m) |
| $\phi_{L_j}$ | measured carrier-phase on L1 or L2 or L5 (m) |
| $\lambda_{L_j}$ $\lambda_c$ | wavelength on L1 or L2 or L5 |
| $MP_{1\|2\|5}$ | multipath linear combination for C1/P1, P2 and C5 observables |
| $N_j$ | non-integer phase ambiguity on L1 or L2 or L5 (cycle) |
| $\rho$ | true geometric range (m) |
| $\rho_j^i(t)$ | modelled range |
| $p^i$, $p$ | position of the $i^{th}$ satellite and receiver, respectively |
| $R_i^j(t)$ | measured pseudorange |
| $\phi_k$ | carrier phase at epoch k |
| $\Delta\phi$ | carrier phase estimation error |
| $\sigma$ | standard weighting parameter for the code or carrier phase measurement (m) |
| $\sigma_{P_k^m}$ | *a priori* standard deviation of the pseudorange |
| $\sigma_P$ | zenith code standard deviation of the pseudorange |
| $v^i$, $v$ | velocity of the $i^{th}$ satellite and receiver, respectively |
| $v_r$: | radial component of the difference between the satellite and the antenna velocities (m/s) |
| $X^i$, $Y^i$, $Z^i$ | components of the geocentric position vector of the satellite |
| $X_j$, $Y_j$, $Z_j$ | unknown ECEF receiver coordinates |

# Chapter 1 Introduction to GNSS and PPP

Global Navigation Satellite Systems (GNSS) is a framework of multiple satellite constellations of different countries to provide positioning ability. Various global satellite navigation constellations are in existence including GPS, GLONASS, BeiDou and Galileo and regional constellations such as QZSS and NAVIC. Moreover a number of point and relative positioning algorithms are used to generate position using a method called triangulation which relies on the distance between a minimum of four satellites and the receiver to generate the three dimensional positioning coordinates and the receiver clock error. This chapter describes the different GNSS constellations, positioning techniques and then discusses the problem statement and the research objectives and contribution.

## 1.1 GNSS Constellations and Types of Positioning

GPS was created by the U.S. Department of Defense (DoD) and was originally operated with 24 satellites. It became fully operational in 1994. Each GPS satellite continuously transmits a microwave radio signal composed of carrier, multiple code (with different precision, e.g., C1P, C1C, C1L, C1W, C1M) and a navigation message. Each GPS satellite emits a different Pseudo Random Noise (PRN) code that applies the code division multiple access (CDMA) principle (El-Rabbany 2002). There are currently three signals being transmitted at three different frequencies, L1, L2 and L5 - of which certain smartphones track the L1 and L5 signals.

GLONASS is a radio-based satellite navigation system operated for the Russian government by the Russian Aerospace Defence Force. It is independent of GPS and currently has a global coverage of 24 satellites with signals at two frequencies (L1 and L2). Smartphones currently track only the L1 signal. GALILEO has been realised by the European Commission (EC) and European Space Agency (ESA). As with GPS, GALILEO's modulation scheme follows the CDMA principle. They are currently 24 satellites in operation with signals at three frequencies : E1, E5 and E6, with E5 being sub-divided into E5a and E5b (European Space Agency 2015). Certain smartphones track the E1 and E5 signals for this constellation. The Chinese government's BeiDou navigation satellite system currently consists of five Geostationary Earth Orbit (GEO) and thirty Medium Earth Orbit (MEO) satellites transmitting on various ranges of carrier frequencies. It has 35 satellites in orbit with signals in the B1, B2 and B3 frequency (Thoelert et al. 2013). Some smartphones currently track only the B1 signal. QZSS (Quasi-Zenith Satellite System) operated by the Japan Aerospace Exploration Agency (JAXA) and Indian Regional Navigation Satellite System (IRNSS) operated by the Indian Space Research Organisation (ISRO) are two Regional Navigation Satellite Systems used to enhance the positioning solution available from the existing constellations in the Asia-Oceania region.

A receiver utilizes these time-tagged signals to determine the range to each GPS satellite in view by measuring signal travel time, which is then multiplied by the speed of light to estimate range. Since the satellite clocks and the receiver clocks are not synchronized, the term pseudorange is used to refer to the code-based range measurement, which is the uncorrected distance between the receiver and the satellite (Hofmann-Wellenhof et al., 2001). The other form of measurement is the carrier-phase measurement, which is the number of carrier cycles a signal travels from the satellite to the receiver. The carrier-phase measurement is much more precise than the code measurements as it is less affected by different errors such as multipath but is ambiguous since the exact phase at which it started transmitting is unknown.

Several GNSS positioning techniques such as Single Point Positioning (SPP), relative techniques such as Real-Time Kinematic, (RTK), and Precise Point Positioning (PPP) have been used to generate positioning solutions. The Single Point Positioning solution is the most basic and widely used positioning technique. It uses the code measurements from GPS, broadcast orbit and clock products and other error corrections to generate a solution with a metre-level accuracy. Relative positioning is a technique that uses a base station with known coordinates to estimate the location of an unknown receiver. With the growing need for accuracy, several relative techniques such as DGPS ( Differential Global Positioning System), RTK, and network RTK were developed which require a reference GNSS station and a baseline range of shorter than 20 metres to transmit error corrections and obtain precise positioning solutions. The RTK technique makes use of the ambiguous but precise carrier-phase measurements along with the noisy code measurements. PPP is a stand-alone positioning technique that is based on standard, single-receiver, single or dual-frequency point positioning using pseudorange (code) measurements and precise but ambiguous carrier-phase measurements. Metre-level satellite broadcast orbit and clock information is replaced with centimetre-level precise orbit and clock information, along with additional error modelling. Numerous PPP positioning engines have been developed over the years, demonstrating few centimetre-level accuracies in both static and kinematic mode (Kouba and Héroux 2001; Bisnath and Gao 2009; Landau et al. 2009; Banville 2014; Aggrey 2015; Laurichesse and Blot 2016).

Initially, techniques such as RTK and PPP were employed for high-quality surveying and positioning applications using geodetic-quality equipment. With the shift towards mass-market applications that focus on low-cost and low-quality receivers for different purposes such as autonomous navigation and precision agriculture, such techniques are being customised to be employed on these receivers. Moreover, with the advent of augmented reality, navigation-based games, tourism and bicycle rental applications, contact tracing, etc, there has been an unprecedented push to improve the positioning accuracy offered by smartphones.

## 1.2 Overview of Smartphone GNSS Positioning

Today GNSS-based positioning in smartphones is a necessity and is used in daily activities such as automotive and personal navigation and gaming. Smartphones have become ubiquitous devices that fulfil a large variety of purposes and hence, there is the need for smartphones to be equipped with extremely accurate positioning capability. Smartphones contain extremely low-cost GNSS chip receivers with low-cost antennas that are mostly, single-frequency GPS L1 receivers. Smartphones have been providing locations based on a combination of GPS L1 code-based SPP solution aided by measurements from an internal Inertial Measurement Unit (IMU) such as accelerometers, magnetometers and gyroscopes and cell tower information. The accuracy of such a technique is at the metre or tens-of-metres level with drifts in cases of signal loss from either satellite or cell towers. Also, the positioning accuracy offered by the smartphone is subject to its manufacturer. Benefon launched the first GPS-enabled personal mobile phone, named “Benefon Esc!” in 1999. (Ketola 1999; Massarweh et al. 2020). It was a commercial navigation device which was capable of tracking up to 12 satellites, by using a twelve-parallel-receiver continuously tracking GPS signal (Massarweh et al. 2020).

In 2018, dual-frequency multi-GNSS chips made their way into smartphones. Since the advent of Android N phones, both carrier and pseudorange measurements were made available to the public. The measurements can be logged and processed to calculate the receiver's position using different techniques such as SPP, RTK and PPP, to name a few.

There has been considerable research in analysing the quality of the solution obtained by techniques such as RTK and PPP and to identify them as potential replacements of the existing positioning technique being used in smartphones which have been summarised by Paziewski (2020) and are also described in Section 2.2.4. RTK allows users to obtain few centimetre-level positioning but increase the infrastructure cost of setting up a network of base stations to provide corrections. Being stand-alone, portable and ubiquitous devices, PPP seems a potential candidate to replace the existing metre-level SPP solution.

The challenges of implementing PPP in smartphones are numerous and hence, this research focuses on analysing and addressing some of these issues. The research on smartphone GNSS measurements from dual-frequency receiver chips is relatively new and a great deal of analysis must be performed to understand the quality of measurements and the various metrics such as carrier-to-noise-density ratio, multipath, measurement noise, missing measurements and cycle slips.

## 1.3 Problem Statement and Research Objectives

With dual-frequency multi-GNSS tracking chips making their way into smartphones in 2018, researchers from all over the world have started analysing the positioning capability provided by smartphones using different measurement processing techniques such as RTK and PPP. Current research has shown that existing PPP processing software, designed for high-cost, geodetic receivers, is not ideally suited for smartphones that have their limitations when it comes to antenna and receiver quality, which effectively impacts the quality of the received measurements. Smartphones come with their shortcomings such as linearly polarised antennas, which is a limitation on the hardware part, and bugs in the applications to log these measurements, which is a limitation on the software. The inability of the antenna to distinguish between incoming and reflected signals and suppress this multipath impacts the quality of the measurement collected (Humphreys et al. 2016; Robustelli et al. 2019; Guo et al. 2020). The operational irregularity of these various applications on different kinds of smartphones with different Android versions is a challenging task since the logged data has problems such as incorrect observation file format and wrongly computed measurements. In the GNSS research on smartphones, there is also a limited understanding of the equipment biases which introduce varying magnitudes of observable error due to each receiver-antenna combination in different smartphones.

Currently, elevation-based weighting is considered suitable to supply *a priori* weighting to code and carrier-phase measurements from existing high-quality geodetic equipment (Paziewski et al. 2019; Banville et al. 2019). Elevation-based weighting is not suitable for smartphones since signals from higher elevation angles also suffer from multipath because the smartphone antennas are unable to distinguish between incoming and reflected signals due to their linearly polarised antennas. The measurement noise and availability of measurements is closely related to the $C/N_0$ value, rather than the elevation angle. Hence, it is crucial to shift to a $C/N_0$-based *a priori* measurement weighting strategy.

Suburban environments which are the prime focus of this research which are characterised by a medium density of tall structures such a buildings and trees. Multipath affects the GNSS measurements, especially in the realistic usage of the smartphone in suburban and dense urban environments. If not corrected for, e.g., by modelling, it affects the positioning solution. It is necessary to model this error. One way to do so is to include the code multipath and $C/N_0$ as a parameter determining the *a priori* weighting factor of the code and carrier-phase measurements from different satellites. Currently, there is no standard model for generating the parameter values used for $C/N_0$-based *a priori* weighting of the code and carrier-phase measurements.

Low-cost antenna and various software applications logging data in Android phones also cause limitations. The frequent cycle slips and phase loss of lock coupled with missing raw measurements in the logged data pose problems in the PPP processing. Multiple satellite rejections in these cases lead to no solution over several epochs. Most applications require a continuous solution, where a less-than-accurate solution is always better than no solution at all. Techniques such as prediction of missing measurements over short durations is a definite solution to this problem. Prediction of measurements requires testing different techniques such as extrapolation, and carrier Doppler estimation techniques to come up with a model with the least prediction error.

Therefore, the objectives of this research is to address the following questions:

1) How do parameters such as $C/N_0$, multipath, measurement noise, cycle slips and data gaps vary with respect to each other, environment and data collection scenario?
2) How will a prediction technique impact the availability and accuracy of a kinematic positioning solution in a suburban medium multipath environment?
3) How will a $C/N_0$-based stochastic modelling technique impact accuracy of the static positioning solution and improve the availability and accuracy of a kinematic positioning solution in a low to medium multipath environment?
4) How does the PPP solution compare with the internal positioning solution and other positioning solutions such as SPP and RTK in different environments?

## 1.4 Research Contributions and Significance

Fuelled by the increasing use and demand for positioning-based apps in smartphones and the development of low-cost, dual-frequency, multi-GNSS tracking chips and antennas in smartphones, this research focuses on contributing to the overall increased need for higher positioning accuracy in smartphones. Challenged by the limitations of these low-cost receivers and antennas, this research focuses on a few key aspects related to the analysing and conditioning the quality of the measurements and customising the quality checks in the existing PPP software, to make it suitable for GNSS measurements from smartphones. Conditioning measurements is a technique that prepares raw data for further processing by ensuring consistency and continuity and ensures the rejection of noisy and missing data.

Most research about smartphone positioning has been limited to single-frequency time-based filtering techniques, Hatch-filtering, SPP, RTK, and in some cases PPP. There has been little work with regards to a deeper understanding of the quality of the measurements in different environments. Most data collection and testing have been limited to static open-sky environments with little or no signal blockages and low multipath. However, the realistic usage of smartphones lies in medium to high obstruction in outdoor environments with the phone in hand or on the dashboard of a car. Such usage brings with it numerous challenges and limitations, some of which have been analysed and addressed in this research.

The primary significance of this work lies in data collection and processing in realistic environments using mannequins and phone holders to mimic real-life usage. To deal with the high multipath, low and irregular carrier-to-noise-density ratio, a new measurement-weighting model has been devised that considers these factors along with the chipping-length or wavelength. Several researchers have mentioned the use of carrier-to-noise-density ratio-based weighting for smartphones, but no conclusive comparison with existing techniques has been elaborated on (Langley 1996; Banville et al. 2019).

Collecting data in realistic environments using different phones and applications has brought up issues such as missing measurements and incorrectly logged measurements. The problem of missing measurements has not found mention in any current research publication. The testing of several different prediction techniques, such as by estimating carrier Doppler measurements using existing logged Doppler measurements to settle on a linear extrapolator for it predicts with the least amount of error, is extremely significant. This prediction technique could potentially make real-time usage of smartphone-based PPP processing possible in difficult or medium multipath environments where frequent phase-loss-of-lock due to the cheap antenna can be a challenge. Lastly, estimating the pre-fit and post-fit residual checks, the carrier-to-noise-density ratio and elevation cut-offs are some important quality control parameters, which if not set correctly, will affect the positioning solution accuracy and availability.

PPP processing will only be considered an ideal replacement to the existing aided-GPS internal solution provided by smartphones if it gives a more continuous and accurate positioning solution. The four major contributions that make this research significant and novel are:

1. In-depth analysis of GNSS measurements from smartphones collected in different multipath environments.
2. Development and implementation of a stochastic model that uses $C/N_0$ and multipath to weigh the measurements.
3. Filling in missing code and carrier-phase measurements with a prediction technique.
4. Testing of the accuracy of the positioning solution with the SPP, RTK and internal positioning solution in realistic environments to assess the positioning solution for smartphone applications in real-life applications.

As a result of completing the above objectives, the following are the outcomes expected:

1. A deeper understanding of the quality of smartphone GNSS measurements in different multipath environments and data collection scenarios.
2. Increased positional accuracy with regards to static and kinematic positioning using smartphones than the exiting internal solution provided.

3. Near continuous and more accurate positioning solution in smartphone data collected in realistic environments with low to medium multipath and signal blockages.
4. A framework ready for future real-time implementation of PPP in smartphones

## 1.5 Thesis Outline

Chapter 2 deals with the background work of this research including a brief discussion of the types of positioning and the history of positioning in smartphones. There is context on the introduction of dual-frequency, multi-GNSS chips in smartphones and the ability to access raw measurements. Ongoing research and analysis on PPP processing of smartphone measurements including different kinds of analysis and pre-processing are discussed. An insight into different research publications based on the quality of measurements from smartphones, history of positioning in smartphones, evolution of GNSS chips and antenna types and errors affecting GNSS measurements are discussed.

Chapter 3 focuses on details regarding the hardware equipment, types of smartphones and their GNSS tracking capabilities and data logging applications. The standard PPP processing technique is discussed along with an overview of the York University PPP processing software used for the processing. There is an in-depth insight into the quality of measurements from smartphones such as carrier-to-noise-density ratio, multipath, data gaps, quality control customizations and various pre-processing, post-processing and filtering techniques applied on the data before the PPP software generates a solution.

Chapter 4 deals with the data gap analysis and forms the core of the research as it presents the research on the prediction of measurements in realistic environments. Various prediction techniques and a dataset has been tested for validating the necessity and implementation of the model.

Chapter 5 focuses on the design and implementation of a realistic stochastic model for the measurements. It compares the positioning accuracy obtained using different measurement weighting methods. Pre-and post-prediction and stochastic model implementation solutions have been qualitatively and quantitatively compared. The positioning solution and residuals have been analysed for different datasets after the conditioning and customization.

Chapter 6 compares the PPP positioning solution obtained after applying the conditioning and customization techniques to the measurements and processing software with other positioning solutions such as SPP, RTK and the smartphone’s internal positioning solution to prove its effectiveness over other techniques.

Finally, Chapter 7, concludes the thesis by summarizing all the outcomes of this work. There is a section devoted to the major challenges and limitations of positioning in smartphones and the means to address those problems. The chapter also provides suggestions and recommendations for future work.

# Chapter 2 GNSS Measurement Processing in Smartphones

This chapter presents a brief overview of the types of GNSS positioning and the history of GNSS positioning in smartphones. Context is provided on different GNSS chips in smartphones and the ability to access raw measurements. Ongoing research related to analysis and processing of smartphone measurements and the evolution of GNSS positioning methods, chips and antennas in smartphones is discussed and contemporary research on the design of stochastic models, error modelling and real-time PPP processing in smartphones is presented.

## 2.1 Overview of GNSS Positioning Techniques

GNSS signals are electromagnetic waves travelling at the speed of the light. Their frequencies fall in the spectrum between about 1.1 and 1.6 GHz (L-band). The L-band prevents significant attenuation of the signal in the atmosphere and ensures signal integrity. The different navigation satellite systems are divided into two main categories:

- Systems providing global coverage called Global Navigation Satellite Systems (GNSS). They include GPS, GLONASS, Galileo and BeiDou. They generally follow Medium Earth Orbit (MEO) configuration to provide global coverage. BeiDou also has inclined geosynchronous orbits (IGSO).
- Systems providing regional coverage referred to as Regional Navigation Satellite Systems (RNSS). They include QZSS and IRNSS. IGSO and geostationary orbits (GEO) are preferred for such systems.

The application, desired positioning accuracy desired, budget and equipment decide the GNSS positioning mode. These modes vary in terms of cost of implementation, accuracy, and convergence time. GNSS positioning can be realised by either real-time or post-processing and can be classified into two main categories - point positioning and differential positioning with the latter requiring a base station with known coordinates and a connection with the receiver to transmit corrections. The most used point positioning technique is single-frequency pseudorange-based positioning SPP, found in consumer-based receivers such as smartphones. Until recently, high-integrity methods for uses such as autonomous navigation to surveying have been limited to expensive and sophisticated receivers with chips that track multi-frequency GNSS measurements. These carrier-phase-based techniques are capable of a centimetre to the sub-centimetre level accuracy. Differential GNSS (DGNSS), RTK and Network RTK are a few kinematic relative techniques. DGNSS is a GNSS Augmentation system or an enhancement to primary GNSS constellation measurements using a network of ground-based reference stations that broadcast differential information to the user. With the recent shift towards low-cost GNSS positioning and integrated navigation, relative and stand-alone PPP techniques have been deployed on these low-cost receivers. A brief description of the different types of positioning techniques is represented in Figure 2.1 and is henceforth, discussed.

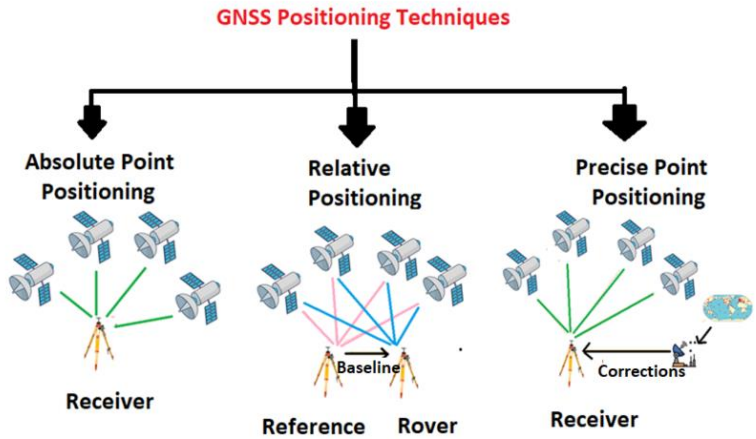



**Figure 2.1: Three types of GNSS Positioning Techniques**

### 2.1.1 Point Positioning

Single Point Positioning (SPP) is the most basic form of satellite-based positioning technique. The navigation solution is either a simple least-squares estimate, weighted least squares estimate, or an extended Kalman filter estimate of the measurement equations made at a single epoch. SPP, along with cell tower data has traditionally been used to produce smartphone positioning solutions. The GNSS observation equation for each satellite is written as:

$$R_{rec}^{i} = \rho_{rec}^{i} + c(dt^{s} - dt_{rec}) \tag{2.1}$$

$dt^{s}$ and $dt_{rec}$ represent the satellite and receiver clock offset from the GPS time (GPST), $R_{rec}^{i}$, the measured pseudorange between the satellite and the receiver and $\rho_{rec}^{i}$ is the geometric range between the satellite and the receiver, computed as follows.

$$\rho_{rec}^{i} = \sqrt{(X - x)^2 + (Y - y)^2 + (Z - z)^2} \tag{2.2}$$

X, Y and Z are the satellite coordinates in the ECEF frame and x, y, z are the receiver coordinates.

Since the observation equation is non-linear in the unknowns $(x, y, z, dt_r)$, it is linearized about the reference state before applying the least-squares solution, iteratively as follows:

$$a = a_0 + \delta \tag{2.3}$$

where:

$$\delta = (H^T P_i H)^{-1} H^t w \tag{2.4}$$

$a_0$ is the *a priori* estimate of the observable, H is the Jacobian matrix that consists of partial derivates of the observations with respect to the unknowns, and w is the misclosure vector. The post-fit residuals can be computed as:

$$r = w - H\delta \tag{2.5}$$

SPP uses noisy pseudorange measurements with limited error corrections and hence, has an accuracy at the metre-level with considerable jumps. In this research, data have been processed in the SPP mode as a reference with the results from the PPP solution.

### 2.1.2 Relative Positioning

Relative positioning is a technique that uses a base station with known coordinates to estimate the location of an unknown receiver. It involves determining the baseline vector between the two points which is generally limited to 10-20 km after which tropospheric and ionospheric variations and biases affect the solution. Assuming the position vectors of the base to be $X_B$ and the unknown receiver to be $X_R$, with the baseline vector $b_{BR}$, the relation can be determined as:

$$X_R = X_B + b_{BR} \tag{2.6}$$

Here, the baseline vector can be represented as:

$$b_{BR} = \begin{matrix} X_R - X_B \\ Y_R - Y_B \\ Z_R - Z_B \end{matrix} \tag{2.7}$$

Relative positioning can be performed with code-only ranges or both code and carrier-phase measurements. Though, for higher accuracy and precision of the solution, carrier-phase measurements are commonly employed in relative positioning. Relative positioning can be employed by single, double or triple-differencing, Double-differencing is the most commonly employed mode of relative positioning, which cancels out the errors related to receiver and satellite clocks. Centimetre-level horizontal accuracy is attainable using double differencing.

Real-Time Kinematic (RTK) is a common kinematic relative positioning technique and will be also be used to compare the positioning solution obtained from PPP. Since smartphones are designed to function independent of baseline limit from a base station, this method cannot be considered a viable replacement to the existing internal solution. Network RTK is based on the use of a large network of base stations and cover large areas. However, they are limited by large infrastructural setup cost and maintenance and are also not available everywhere.

### 2.1.3 Precise Point Positioning

Precise Point Positioning (PPP) is a technique to process raw GNSS measurements to obtain the receiver position with very high accuracy. It uses precise orbits, additional error correction and modelling and a single receiver, to obtain precise positioning solution (Bisnath and Gao 2009). The PPP technique models a considerable number of errors such as antenna offset, ocean loading, phase windup and clock errors besides the normal ionospheric and tropospheric corrections and does not need an additional base station for transmitting error corrections.

PPP processing can be done in single frequency and combined and uncombined dual-frequency modes. Dual-frequency PPP is based on forming ionosphere-free linear combinations of the code and carrier-phase measurements (Wells 1987). However, one could also utilize slant ionospheric estimates as an alternative, which is what is employed in the context of this research. The un-differenced observation equation can be expressed as (Aggrey 2015):

$$P_{L_i}^{GNSS} = \rho^i + c(\partial t^s - \partial t_r) + d_{trop} + d_{ion,L_i} + b_{P_{r_i}}^{L^i} + b_P^{L^I,S_j} + d_{multi\left(P_{L_i}\right)} + \epsilon\left(P_{L_i}^{GNSS}\right) \quad (2.8)$$

$$\varphi_{L_i}^{GNSS} = \rho^i + c(\partial t^s - \partial t_r) + d_{trop} + d_{ion,L_i} + b_{\Phi_{r_i}}^{L^i} + b_{\Phi}^{L^I,S_j} + \lambda_{L_I}, N_{L_i} + d_{multi\left(\Phi_{L_i}\right)} + \epsilon\left(P\varphi_{L_i}^{GNSS}\right) \quad (2.9)$$

The variables are:

$P_{L_i}^{GNSS}$ - Pseudorange measurement on L1, L2 or L5 (m)

$\varphi_{L_i}^{GNSS}$ - Carrier-phase measurement on L1, L2 or L5 (m)

$\rho^I$ - Geometric range (m)

c - Speed of light (m/s)

$\partial t^s$ - Satellite clock error (sec)

$\partial t_r$ - Receiver clock offset (sec)

$d_{trop}$ - Tropospheric delay (m)

$d_{ion,L_I}$ - Ionospheric delay on L1 or L2 or L5 (m)

$b_{P_{r_i}}^{L^I}, b_{\Phi_{r_i}}^{L^I}$ - Receiver equipment biases for pseudorange and carrier-phase measurements (m), respectively

$b_P^{L^I,S_j}, b_\Phi^{L^I,S_j}$ – Satellite equipment biases for pseudorange and carrier-phase measurements (m), respectively

$\lambda_{L_i}$ - Wavelength of L1 or L2 carrier waves (m)

$N_{L_I}$ - Unknown cycle ambiguity term on L1, L2 or L5 carrier-phases (cycles)

$d_{multi(P_{L_I})}, d_{multi(\Phi_{L_I})}$ - Pseudorange/Carrier-phase multipath on L1, L2 or L5 (m)

$\epsilon(P_{L_i}^{GNSS}), \epsilon(\varphi_{L_i}^{GNSS})$ - Pseudorange/Carrier-phase measurement noise (m)

The challenge with PPP is the time it takes to converge, especially with GNSS measurements from smartphones due to large convergence time for the ambiguity float solution. The combination of measurements from multiple constellations reduces convergence time due to geometry improvement. The uncombined positioning model is slowly being adopted as it allows for the integration of additional information, such as precise ionospheric errors. The elimination of ionospheric effects will help reduce convergence time in real-time processing. However, the accuracy depends on the quality of ionospheric products and the efficiency of eliminating systematic effects in the measurement model. Global Ionosphere Maps (GIM) products which provide global maps of the ionosphere's total electron content (TEC) can be used for ionospheric correction. They correct only 80% of the total errors and can affect the accuracy of the positioning results. For this research uncombined processing technique is used which estimates the slant total electron content.

Processing of GNSS data for PPP is performed in five main stages, which involve: 1) reading of GNSS observations; 2) data pre-processing and measurement conditioning; 3) application of error correction models; 4) filtering and output of estimated parameters. The GNSS observations from the smartphone together with precise orbit and clock products, along with antenna corrections and DCB's (differential clock biases) undergo data pre-processing or measurement conditioning. Error correction models are applied to account for any possible error sources. These conditioned measurements are then pre-processed in a sequential least-squares filter or extended Kalman filter. Applying functional and stochastic PPP models, the positioning coordinates and other parameters are re-estimated again. The final output includes station coordinates, receiver clock offset, tropospheric delay and GNSS system time differences.

## 2.2 GNSS processing in smartphones

This section focuses on the evolution of GNSS-based positioning in smartphones in terms of processing techniques, GNSS chips, accessing raw measurements and the introduction of PPP based processing mode to smartphones. A brief history of positioning in smartphones is also provided.

### 2.2.1 Brief Overview of the Evolution of GNSS chips in Smartphones

Historically, smartphones have supported single-frequency GPS single-point positioning which was accurate only up to a few metres, because of poor antenna quality and signal blockages. Loss of signal and rapid position variation were frequent. In 1999, the phone manufacturer Benefon launched the first commercially available GPS phone Benefon Es. Since then there has been an evolution in smartphone technology (Ketola 1999). From one phone model with GPS in 1999 to fewer than 10 phone models with GPS in the next five years, to over 200 models by 2017, the smartphone GPS chips underwent drastic improvement in quality (van Diggelen 2009). The E-911 was the big catalyst for getting GPS into phones along with the introduction of Massive Parallel Correlation (MPC) in which all code-delays are searched in parallel, speeding up the acquisition process. In 2005, the first smartphone implementation of MPC was found in the HP iPaq, which used the GL20000 GPS chip (van Diggelen 2009).

The number of Android smartphones compatible with the Google Application Programming Interface (API) is increasing continuously. In early 2017, the Samsung S8 and Huawei P10 were released as the first multi-GNSS smartphones which had chips possessing the capability to track carrier-phase measurements. However, the breakthrough came later in 2018. Until May 2018, the GPS chipsets incorporated in smartphones were single-frequency and mostly GPS-tracking. In some cases, they were already multi-constellation, but the mono-frequency limited the performance because the ionospheric error could not be eliminated but only estimated using a single frequency model with different techniques such as the Klobuchar method. Moreover, equipped with inverted-F monopole antenna single frequency chips in smartphones, positioning was limited to a few metres with poor multipath suppression and inability to extract it. With recent advancements in GNSS receiver and antenna technology, several manufacturers started working on chips suitable for low-cost mass-market applications with the hope of achieving higher positioning accuracy, like geodetic receivers. Only recently, has a breakthrough been brought about with the advent of dual-frequency, multi-constellation GNSS chips (EGSA 2018). Dual-band GNSS can mitigate multipath interference effects, which are especially present in areas with a high density of buildings and deliver significantly higher accuracy than single-frequency devices.

With Location-Based Services (LBS) applications in mind, Broadcom released the BCM47755 dual-frequency chip, in 2018. They claimed the system's architecture achieved the synergistic benefits that cannot be reached by multiple integrated circuits, thereby reducing its overall size and power consumption (GPS World 2018). It is a dual-frequency (E1/L1+E5/L5) GNSS chip. The newly added L5/E5 signals with binary phase-shift keying (BPSK (10)) and alternative binary offset carrier (AltBOC (15,10)) modulation is certainly an improvement from traditional GPS chips, especially in cities, where they are less prone to distortions from multipath reflections than L1/E1 (Guo et al. 2020). Other important manufacturers in this market have also come forward with dual-frequency chips, including Qualcomm with the Snapdragon X24 LTE modem and HiSilicon with the Kirin 980 system-on-a-chip. Both attribute their superior energy efficiency and form factor to intelligent data processing and FinFET transistor design. A list of some of the phones and the GNSS chips, along with their capabilities is presented in Table 2.1.

**Table 2.1: Smartphones along with their GNSS chips and tracking capabilities**

| Phone | Model Number | Frequency Band | Constellations Tracked | GNSS chipset |
|---|---|---|---|---|
| **Xiaomi MI 8** | MI803E1A | L1, L5 | GPS, Galileo, GLONASS, BeiDou, QZSS | Broadcom BCM47755 |
| **Huawei Mate 20** | HMA-L29 | L1, L5 | GPS, Galileo, GLONASS, BeiDou | HiSilicon Kirin 980 |
| **Samsung S9** | SM-G960F | L1 | GPS, Galileo, GLONASS, BeiDou | Exynos 9810 – Broadcom BCM 47752 |
| **Google Pixel 3** | Pixel 3 | L1 (code-only) | GPS, Galileo, GLONASS, QZSS | Qualcomm SDM845 Snapdragon |

The different frequencies tracked by the BCM47755 chip in the Xiaomi phone, used for this research, include code and carrier-phase measurements from L1 and L5 for GPS, E1 and E5a for Galileo, single-frequency L1 for GLONASS and B1 for BeiDou (EGSA 2018; Robustelli et al. 2019; Aboutaleb et al. 2019). A brief overview of the GNSS constellations and frequencies tracked by the BCM47755 chip in the Xiaomi MI 8 and the reference receiver, SwiftNav Piksi, is given in Figure 2.2.

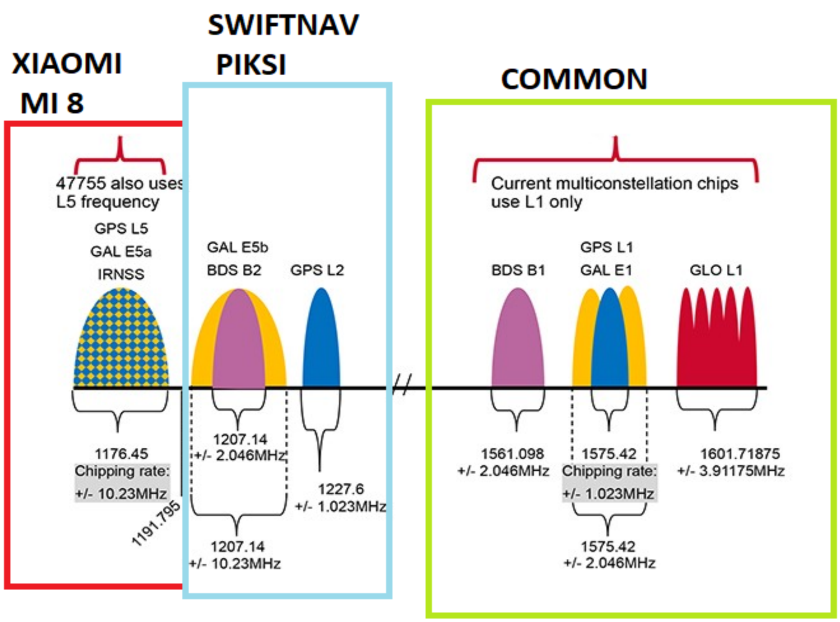



**Figure 2.2: GNSS constellations and frequencies tracked by the BCM47755 chip in the Xiaomi MI 8 and the SwiftNav Piksi (Barbeau 2018)**

The different applications of smartphone-based positioning and navigation have increased from infotainment, maps, social networking. Moreover, in recent years, a novel group of applications has emerged within smartphones that require far more accurate positioning results, such as augmented reality, image-based navigation for tourism purposes, and smartphone-based photogrammetric unmanned aerial vehicle (UAV) system. Such high accuracy services require positioning solutions obtained from multi-GNSS chips with both code and carrier-phase tracking capabilities (Guo et al. 2020).

Moreover, a second-generation dual-frequency GNSS chip, the BCM47765, has been released by Broadcom in May 2020. It is capable of tracking 30 new L5 signals, the new BeiDou-3 constellation B2a signal and signals from the NAVIC constellation which will increase the availability of dual-frequency signals on three different GNSS constellations instead of two. It is expected to have a significant improvement in positioning accuracy and will be used for innovative lane-level driving navigation instructions, allowing driving applications to know which highway lane the vehicle is in (Cozzens 2020).

### 2.2.2 Accessibility of Raw Measurements from Smartphones

Before 2016, the GNSS chipsets in smartphones only output the position-velocity-time and limited satellite information such as the elevation and azimuth (Guo et al. 2020). The pseudorange, Doppler, and carrier phase observables were not available to developers, as they were protected by chip manufacturers. The position computed by the GNSS chipsets was available to everyone. The positioning accuracy only reached 2 to 3 metre, and even degraded to 10 metre or more in the case of adverse multipath conditions.

A breakthrough in the field of smartphone-based navigation happened in May 2016 when Google, during the annual developer-focused conference, announced that raw GNSS measurements will be available to apps in the Android Nougat operating system. Before the Google announcement, measurements collected by a smartphone could not be processed by post-processing software, and therefore, under the open sky, static conditions, positioning accuracy could reach between three to five metres, while under adverse multipath conditions and in urban areas, it degraded to tens of metres (Robustelli et al. 2019). Shortly after, in July 2016, the first Galileo-ready smartphone, the BQ Aquaris X5 Plus was made available. With the release of the Android Nougat operating system and subsequent version (version 7.x), some smart devices allow direct access to raw data and PVT solutions by acquiring pseudorange and carrier-phase measurements from the chipset. (Humphreys et al. 2016; Zhang et al. 2018).

To access these raw measurements Google released its GNSS logger app that logs raw measurements in National Marine Electronics Association (NMEA) like string format. They do not output code measurements directly, but it can be computed from the received and elapsed time and the pseudorange rate of change. The GNSS logger was also supported by a publicly available weighted least squares SPP MATLAB processing engine, that helped analyse the measurements and process the GPS code-only measurements (Android Developers 2016). Alongside, they released a GNSS Analysis App, that takes the observation file as input and generates plots for the positioning solution, carrier-to-noise-density ratio, satellite elevation and azimuth and receiver velocity (Android Developers 2016). Furthermore, other organisations started releasing their loggers which log these measurements in RINEX format. These organisations followed the recent RINEX 3.03 version. A brief overview of some of them is given in Table 2.2.

**Table 2.2: Existing GNSS loggers for smartphones**

| Logger | Organization | Output Format | Logged measurements | | |
|---|---|---|---|---|---|
| | | | Pseudoranges | Carrier-phase | Ephemeris |
| **GNSS Logger** | Google | NMEA | Yes | Yes | Yes |
| **Geo++ RINEX Logger** | Geo++ Gmbh | RINEX 3.03 | Yes | Yes | No |
| **RINEX ON** | Nottingham Scientific Ltd | RINEX 3.03 | Yes | Yes | Yes |
| **Galileo Pvt** | ESA | CSV/NMEA raw Android measurements | Yes | Yes | No |
| **G-RitZ Logger** | Ritsumeikan University | RINEX 2.11/ 3.03 | Yes | Yes | No |

The shift from SPP to PPP and RTK seemed a viable option for the raw code and carrier-phase measurements were available from smartphones for different constellations. However, the loggers do suffer from bugs and multiple updated versions of these apps have been released over the years 2018 and 2019. These bugs range from the incorrect computation of measurements, time-logging issues and incorrect logging of the measurements, violating the RINEX 3.03 version. These applications also tend to interact differently with different phones and Android versions, with some phones not supporting one or many applications. Some applications also crashed or hung during data collection, due to over-heating of the phone.

Researchers from all over the world began working on the integration of measurements from multiple GNSS constellations, transitioning from GPS based SPP with aided solution from devices like accelerometers and gyroscopes to an autonomous single receiver GNSS PPP, for poor quality data available from the low-cost antenna and GNSS chips in smartphones. For this research, the data collection was done using the Geo++ RINEX Logger and the GNSS Logger, as they gave the best possible logged data with few errors regarding the format and computation of logged measurements.

### 2.2.3 Internal Positioning Solution

Internal positioning solution obtained from smartphones is generally output using SPP. Smartphones naturally operate in duty-cycling mode, in which the measurements are tracked for 200 milliseconds and then switched off for 800 milliseconds, which causes drift in solution (GNSS 2016). To correct this drift, most smartphones use corrections from cell-towers transmitted over the internet which can cause biases of a few tens of metres based on the position of the cell tower. The smartphone internal positioning result can also be aided by inertial sensors such as in-built accelerometers and gyroscopes, which are not very accurate themselves, considering their low quality.

The internal solution is computed using different techniques in different phones, and hence, can vary in their accuracy. Some phones output the internal solution in the form of NMEA strings, while others output the latitude, longitude and altitude. The internal solution has been plotted and compared with the PPP solution in this research, for it is crucial to understand how PPP is a viable replacement to this existing solution.

### 2.2.4 A Brief History of GNSS Positioning in Smartphones

GNSS positioning with smartphones is not entirely new. Previously, tablets and smartphones have been tested for GPS only positioning using code measurements which expanded into multi-GNSS constellation positioning along with using the more-precise carrier-phase measurements. Yoon et al. (2016) verified a DGNSS coordinate projection method for smartphone positioning. The method is a development version of position-domain and addressed past limitations such as the unavailability of raw GNSS observations. Using this method it is possible to improve positional accuracy by approximately 30%–60%, and eventually to achieve a 1 m level of accuracy. On the other hand, a range-domain correction model is considered as superior to a position-domain approach , with regards to the current availability of raw smartphone observations.

Early GNSS positioning experiments on smart devices used mostly the Nexus 9 tablet. The experiment conducted by Gim and Kwon-dong (2017) was a pseudorange positioning test using the Nexus 9 tablet, and the results show positioning rms errors in horizontal and three-dimensional were 3.05 and 3.82 m, respectively. E. Realini et al. (2017) carried out a single-frequency carrier phase double-difference positioning experiment using the tablet and several base stations, and a positioning accuracy better than 20 cm was achieved within 20 minutes. One of the most important conclusions drawn was regarding the quality of the signal and measurements. The carrier-to-noise-density ratio ($C/N_0$) of GNSS raw observations collected by mobile devices is low and irregular, approximately 10 dBHz lower than those obtained from a geodetic-quality antenna and receiver. The time-differenced (TD) filtering method, achieved static positioning with horizontal and vertical positioning rms errors less than 0.6 and 1.4 m, respectively. These studies have important implications for subsequent experiments using smartphones; however, the positioning of ordinary smartphones is not comparable to this tablet. The initial point positioning performance was not as promising as the smartphone positioning accuracy was several tens of metres.

In September 2017, Sikirica et al. (2017) used a Huawei P10 smartphone, and the positioning rms errors were about 10 m in the north and east directions and about 20 m in the up direction using the single code positioning performance in a relatively open sky condition. In November 2017, Dabove and Di Petra (2019) conducted an NRTK (network real-time kinematic) positioning test using a Samsung S8+ smartphone and a Huawei P10+ smartphone. They achieved an average horizontal positioning error of about 60 cm. The combination of pseudorange, carrier phase and doppler observations helped reach a positioning accuracy of 0.6 m in the horizontal direction and 1.4 m in the vertical direction. The measurements were processed using the time series differential algorithm with the $C/N_0$-dependent weighting method (Zhang et al. 2018). Maritime positioning requires positioning accuracy of a few metres and for that Specht et al (2019) conducted a test with 6 Samsung Galaxy smartphones and showed that a horizontal positioning accuracy below 10 m can be achieved, which meets most of the maritime navigation accuracy requirements.

Compared with geodetic and other survey-grade receivers, the conditioning and pre-processing of smartphone GNSS raw observations and the analysis of smartphone positioning performance is still in its preliminary stages (Chen et al. 2019). After raw GNSS measurements were made public, several research teams have been involved in assessing the quality of the measurements. The NSL's FLAMINGO (Nottingham Scientific Limited's Fulfilling Enhanced Location Accuracy in The Mass-Market Through Initial Galileo Services) Team assessed the quality of the GNSS raw measurements from Xiaomi MI 8 smartphones in July 2018 (Fortunato et al. 2019)  They considered that the carrier phase was not affected by duty cycling, and the quality of the raw observations of L5/E5 frequency is better than L1/E1 frequency. Robustelli et al. (2019) also performed a carrier phase differential positioning test on a Xiaomi MI 8 smartphone (using single frequency data) and achieved horizontal positioning rms errors of 1.02 and 1.95 m at low and high multipath sites. One of the major limitations hampering the achievement of centimetre-level accuracy using smartphone GNSS positioning is the presence of unaligned chipset initial phase biases (IPBs) (Geng et al. 2018) . To overcome this constraint, Geng and Li (2019) calibrated the IPBs in a post-processing mode. This helped achieve ambiguity resolution and, hence, provided a precise centimetre-level position in a relative mode, which is an improvement of up to 80% as compared to the float solution.

### 2.2.5 PPP Positioning by other Researchers using Smartphones

As a stand-alone positioning technique, PPP in smartphones could bring about a revolution in the use of smartphones and low-cost receivers for different positioning applications such as sports, precision agriculture, geophysics, gaming, autonomous navigation and even help in surveying. These commercially popular devices could also potentially replace expensive geodetic receivers and hardware used for applications such as cadastral surveying, autonomous navigation, and deformation surveying.

Several researchers have observed that due to the use of an inverted-F linearly polarised monopole antenna, which costs a few tens of dollars, the received signal suffers in quality and the measurement noise of the pseudorange measurements is large ( Gill et al. 2017, Zhang et al. 2018; Robustelli et al. 2019; Guo et al. 2020; Wanninger and Heßelbarth 2020). Gill et al (2017) used a Nexus 9 tablet to process GPS measurements in the single-frequency PPP mode. They observed that a positioning rms of 37 cm and 51 cm could be achieved in horizontal and vertical components respectively. The raw measurement analysis revealed that on average for all elevation angles, the $C/N_0$ of cellphone-grade hardware is ~7.5 dBHz lower compared with those from the geodetic-grade hardware. Fortunato et al. (2019) conducted dynamic positioning experiments on Xiaomi 8 smartphones. Though sub-metre level of positioning accuracy could not be achieved, it did open a wide range of discussions and research on how to condition the raw measurements. The horizontal positioning rms errors of 1.17 and 2.23 m was achieved using the RTK and PPP methods, respectively when walking with the phone in hand in a suburban environment. Wu et al. (2019) processed GNSS measurements from a Xiaomi MI 8 smartphone in the dual-frequency PPP mode and obtained rms positioning errors in the E, N, and U directions of 21.8, 4.1, and 11.0 cm, respectively. However, it took 102 min for the three-dimensional positioning error to converge to 1 m. In the kinematic mode, the data quality of the dual-frequency smartphone was poor, and it was difficult to obtain a continuous positioning solution. Compared to the geodetic receiver, the positioning accuracy of the double-frequency smartphone was 3 to 5 m in kinematic mode. Aggrey et al. (2020) reached a positioning accuracy of 4 cm and 9 cm in 2D and 3D directions, respectively, in open sky conditions, using the Xiaomi MI 8 smartphone kept flat on the rooftop and processing the raw GNSS measurements in the dual-frequency PPP mode.

## 2.3 Smartphone GNSS Hardware

The current dual-frequency multi-GNSS tracking smartphone chips are found in certain smartphones such as the Xiaomi MI 8 and 9, the Huawei Mate 20, Huawei P 10 and 30. Certain other smartphones such as the Samsung S9 track single-frequency code and carrier-phase measurements. The BCM47755 chip found in the Xiaomi MI 8 is shown in Figure 2.3. The chips are equipped with a timing clock and a phase-loop tracker.


**Figure 2.3: Broadcom BCM47755 chip found in the Xiaomi MI 8 (EGNSSA 2017)**

Different constellations and frequencies follow different chipping rates and data modulation techniques that must be countered for in the manufacturing design. In most smartphones, there is a single antenna for GNSS, as well as mobile-based applications. A detailed overview of antennas is given in Section 2.3.2.

### 2.3.1 Impact of Duty Cycle

Duty cycling is a power-saving option, which is by default, turned on in smartphones. Duty cycling is a mechanism by which the receiver chip in smartphones does not continuously track its location so that the battery power can be conserved. However, this is not ideally suited for GNSS positioning as the lack of continuous measurements equals re-initialisation of the PPP filter and carrier-phase ambiguities and the PPP solution does not converge to an accurate solution. Since continuous use of a GNSS receiver would drain the battery, the receiver tracks GNSS data for 200 ms before shutting down for 800 ms. The non-continuous tracking and logging is not good for carrier-phase measurements and leads to cycle slips on all satellites in every epoch (Paziewski et al. 2019). Paziewski et al. (2019) used the difference between the code and carrier-phase measurements to impact the effect of duty-cycling on smartphone-based PPP positioning. The phase-code differencing combination is commonly used to assess the code noise; however, in this case, these values can provide the potential presence of effects related to the duty cycling. The phase minus code combination between satellite, m, and receiver, k, in the units of metres can be expressed as:

$$\lambda\phi_k^m - P_k^m = \lambda N_k^m + c\left(B_{k,\phi} - B_\phi^m - B_{k,P} + B_P^m\right) - 2I_k^m + M_{k,P}^m - M_{k,\phi}^m + \epsilon_\phi - \epsilon_P \qquad (2.10)$$

where λ is the signal wavelength, φ is the carrier phase observable in cycles, P denotes the code pseudorange, N states for integer ambiguity, superscript m and subscript k denote satellite and station, respectively, and B corresponds to receiver/satellite bias. I is the ionospheric delay, M is the multipath effect, and ε denotes observation noise for code and carrier-phase measurements, separately.

Each time series is corrected with the value related to the initial epoch. The observables of the high-grade receiver show that the difference between the code and phase measurements is mainly due to code measurement noise and a very weak trend related to the impact of the ionosphere. For smartphones, this trend is seen only for the first five minutes and is related to the fact that duty cycling is activated by the smartphone after this time, which is required by the chipset for ephemerides acquisition (Pirazzi et al. 2017). That takes several minutes, which agrees with the results shown in Figure 2.4. During this period the smartphone phase minus code differences are subjected to two anomalies. The first is a noticeable drift, responsible for increasing divergence between phase and code data. The latter is repeated variations featured every 3 seconds and causing jumps in phase minus code data in the range of about 20 m. Hence, the anomalies in the smartphone carrier-phase measurements are driven by the duty-cycling mode in Figure 2.4. The blue represents the trend for the smartphone while the red is for the Javad (reference receiver) and the green represents the $C/N_0$.

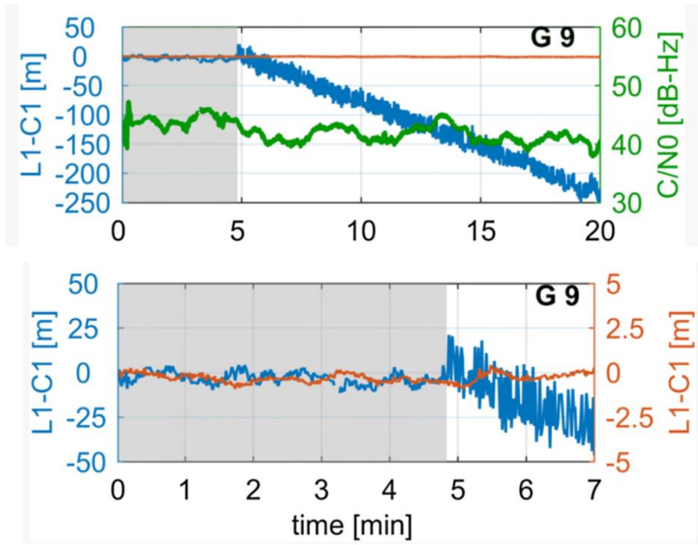



**Figure 2.4: Time series of phase minus code differences (blue) and C/$N_0$ (green) of Huawei observables versus the phase-code differences (red) for the Javad receiver (Paziewski et al. 2019)**

Phase standard deviations are the subject of strong drifts observed earlier in the smartphone phase–code combination. Its occurrence leads to the conclusion that the effect originated in the smartphone phase observables. The gradual divergence between the smartphone phase and code clocks occurs during the duty-cycling period. The drift in phase measurements is depicted in Figure 2.5, and hence duty cycling must be turned off before collecting measurements to make full use of the precise carrier-phase measurements in generating an accurate positioning solution.

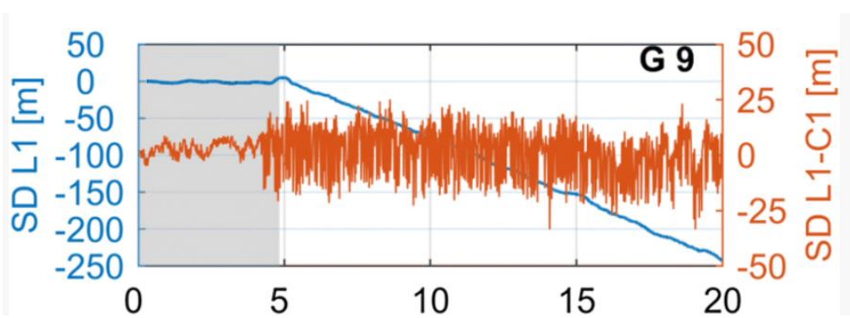
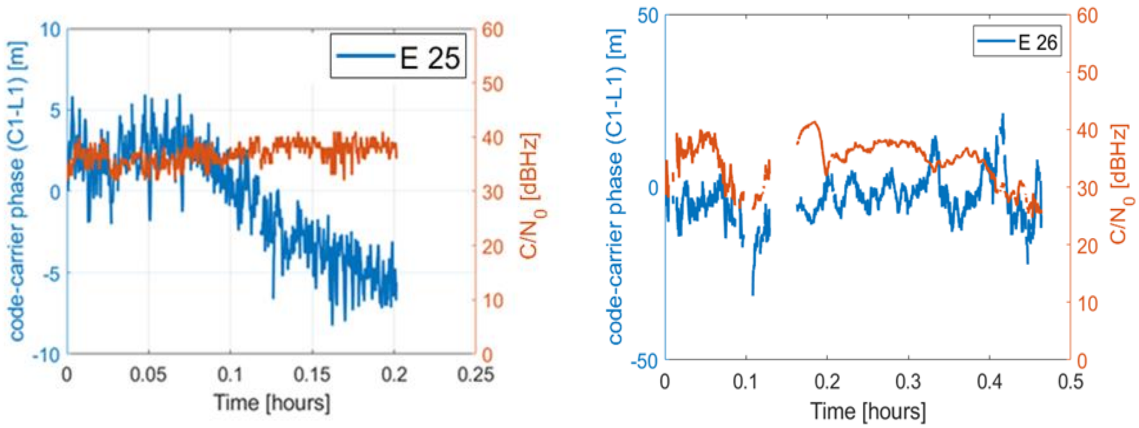



**Figure 2.5: Time series of difference in phase standard deviations (blue) and phase minus code (red) single-difference for Huawei P20 (Paziewski et al. 2019)**

To further verify this phenomenon, a dataset was collected with a Xiaomi MI 8 phone on DOY 102, 2020 with duty-cycling on. Figure 2.6 (a) shows the marked drift in the measurement noise for Galileo PRN 25, coinciding with the non-consistent tracking of the carrier-phase measurements, while no such effect can be seen on Galileo PRN 26 in Figure 2.6(b) with duty cycling turned off.




(a) Duty-cycling on (b) Duty-cycling off

**Figure 2.6: Time series of phase minus code differences (blue) and $C/N_0$ (red) for Xiaomi MI 8 when duty cycling is (a) on and (b) off**

### 2.3.2 Impact of Antenna on Signal Quality

One of the biggest impacts affecting the quality of the solution is the nature of the antenna. Table 2.3 describes a range of antenna grades of decreasing quality. The loss numbers in the rightmost column represent the average loss in gain relative to a survey-grade antenna, where the average is taken over elevation angles above 15° since signals below have high multipath and measurement noise and inevitably result in poor positioning solution quality. (Pesyna and Heath 2013).

**Table 2.3: Types of antennas and their properties (Pesyna and Heath 2013)**

| Antenna Class | Axial Ratio | Polarization | Loss (relative) |
|---|---|---|---|
| **Survey-grade** | 1 dB | Circular | 0 dB |
| **High-quality patch** | 2 dB | Circular | 0-0.5 dB |
| **Low-quality patch** | 3 dB | Circular | 0.6 dB |
| **Smartphone-grade** | 10+ dB | Linear | 11 dB |

Survey-grade antennas have a uniform quasi-hemispherical gain pattern, right-hand circular polarization, a stable phase centre, and a low axial ratio. The second and third rows list properties for high- and low-quality patch antennas. These antennas have properties like a survey-grade antenna and lose, on average, less than 0.5 dB and 1 dB, respectively, in sensitivity as compared to the survey-grade antenna. The smartphone antenna is an inverted-F, linearly polarised antenna that has a low and irregular gain pattern, high relative loss and inability to distinguish between multipath and non-reflected incoming signals. Also, the received signal strength of the L5 signals is lower than the L1 signals, although L5 signals have higher emitted power (Canal Geomatics 2007).

Several researchers have attempted to characterise the quality of the signal based on the kind of antenna used. One way to do so is to use the carrier-to-noise-density ratio to analyse the quality of the signal. According to Paziewski et al. (2019), the carrier-to-noise-density ratio ($C/N_0$) of signals collected by a Javad geodetic receiver fits in the range of 35–55 dBHz, whereas in the case of the smartphone, these values did not exceed 48 dBHz in ideal open sky conditions. They further investigated the antenna characteristics by relating the elevation angle of the satellites with the $C/N_0$ of the received signal. The relationship between the satellite elevation and $C/N_0$ of smartphone measurements is not obvious, especially at low elevations. The signals are characterized by a noticeable number of low values of $C/N_0$, reaching the level of 10 dBHz for smartphones. The occurrence of these outliers is probably related to multipath affecting the smartphone observables due to the poor multipath suppression of the antenna (Pesyna et al. 2014).

### 2.3.3 Single-Versus Dual-Frequency PPP

The tradition in the smartphone market has been single-frequency GPS. What this means is that the smartphone tracks a single radio signal from each satellite. While most current devices only use the L1/E1 frequency, dual-frequency-enabled devices make use of both. The L5/E5a signals are transmitted at higher power and are thus, less prone to multipath errors. They can be used to refine position accuracy to as low as 30 cm with low-cost receivers, with processing techniques such as PPP. Currently, fourteen GPS satellites transmit on L1 and L5.

The GPS L1 signal has a binary phase-shift keying (BPSK) modulation with a chipping rate of 1.023 MHz, often represented as BPSK (1). In the case of GPS L5, the same modulation technique is used but with a chipping rate that is 10 times higher than GPS L1, which can be written as BPSK (10). A higher chipping rate causes the power spectral density of the L5 signal to have a wider frequency occupation, directly impacting the auto-correlation function which is narrower. These characteristics allow more accurate code tracking, more consistent rejection of narrowband interference and increased multipath suppression. L5 code-based pseudorange measurements have a higher quality than their L1 counterparts (Ciuban Sebastian et al. 2020). For smartphones, however, the current signal strength of the L5 band is lower than the L1 band which will be discussed in Chapter 3, mainly due to the differential behaviour exhibited by the smartphone antenna.

For single-frequency PPP (SF-PPP) applications, the ionospheric delay is a dominant error source. Therefore, the performance of SF-PPP depends highly on the elimination of ionospheric delays. Dual-frequency PPP (DF-PPP) is based on forming ionosphere-free linear combinations of the code and carrier-phase measurements (Wells 1987). However, one could also utilize slant ionospheric delay estimation as an alternative, which is the uncombined method of processing measurements and is preferred over the traditional combined ionosphere-free techniques (Aggrey 2015).

## 2.4 Analysis of the Quality of Smartphone GNSS Measurements by Other Researchers

The quality of the raw measurements is affected by the quality of the antenna. Paziewski (2020) has summarised the work done in the field of smartphone GNSS positioning by multiple researchers. He concluded that limitations in the quality of smartphone GNSS measurements restrict the use of smartphones for highly precise position-based applications. These limitations range from the low quality of currently embedded antennas implying high susceptibility to multipath, and low gain and linear polarization, which prevents the differentiation of right-hand circularly polarized direct line-of-sight signals from reflected left-hand circularly polarized non-line-of-sight signals. Another challenge was the lack of phase centre offset and variation models for the majority of smartphone GNSS antennas (Paziewski 2020).

The quality of the code pseudoranges can be assessed from the differences between code and carrier phase measurements as applied by Lachapelle et al. (2018). They obtained an rms value of over 2.2 m for a Huawei P10 smartphone, approximately ten times greater than that of a high-grade receiver. Similarly, a comparable ratio between the code noise of a Nexus 9 smartphone and a high-grade receiver was obtained by Li and Geng(2019), who used a third-derivative approach. In the case of phase observations, the smartphone measurement noise was only 3–5 times larger compared to high grade receivers reaching up to 0.04 cycles. Another way to analyse the quality of measurements is to use the carrier-to-noise-density ratio ($C/N_0$) to analyse the quality of the signal. Paziewski et al. (2019) noticed that the $C/N_0$ of signals collected by a Javad geodetic receiver is as high as 55 dBHz whereas, in the case of the smartphone, these values did not exceed 48 dBHz. The signals are characterized by a noticeable number of low values of $C/N_0$, reaching the level of 10 dBHz for smartphones. The relationship between the satellite elevation and $C/N_0$ of smartphone measurements is not obvious, especially at low elevations.

Guo et al. (2020) used a Xiaomi MI 8 smartphone for data collection. They observed that for static data, the pseudorange outlier percentages were in the range 1.31%~4.79% for GPS L1 and L5, 1.7% and 1.3% for Galileo E1 and E5a, respectively.  They observed that in general, the pseudorange outlier rate of L5/E5 observations was significantly lower than that of L1/E1. The L5/E5 observations have the potential to outperform L1/E1 in positioning and navigation and should help increase positioning accuracy in dual-frequency processing.  Unexpectedly, they observed that the percentage of pseudorange outliers was as high as 42.1% and 19.6% for GLONASS and BeiDou, respectively. Elmezayen and El-Rabbany (2019) also conducted some pre-processing analysis on the measurements obtained from a Xiaomi MI 8 smartphone. They attributed the poor measurement quality to the combination and quality of the receiver and the antenna. Unable to distinguish LHCP signals from RHCP, they observed the $C/N_0$ to be on an average 10 dBHz lower than a geodetic receiver.

## 2.5 Error Modelling

Various errors plague the quality of the PPP solution and must be corrected. PPP is based on undifferenced observations, so most common mode errors do not cancel out. A brief overview of some of them is listed subsequently.

**Atmospheric errors**

The most common type of errors which need to be corrected for in any kind of positioning are those related to the troposphere and the ionosphere. These two media cause electromagnetic signals to be refracted, which needs to be corrected for accurate positioning.

**Tropospheric delay**

Tropospheric delay is variable and difficult to correct accurately for since it varies based on factors such as the atmospheric temperature, pressure, humidity, receiver altitude, and satellite elevation angle. Extending to approximately 50 km into the atmosphere from the surface of Earth, it comprises the dry component causing ~ 90% of the delay and wet component causing ~10% of the delay. The delay caused by the dry and the wet components of the troposphere is usually modelled at the zenith angle and then scaled by an appropriate mapping function to any satellite elevation angles. Expressing total tropospheric delay as a combination of the delay caused by the dry and the wet components is as follows (Choy 2009) (Shen 2002):

$$\mathrm{d_{trop}} = \mathrm{M_{dry}}\mathrm{d_{dry}} + \mathrm{M_{wet}}\mathrm{d_{wet}} \tag{2.11}$$

here:

$d_{dry}$: Zenith path delay (ZPD) caused by dry component

$d_{wet}$: ZPD caused by wet component

$M_{wet}$, $M_{\mathrm{dry}}$ : Wet and dry mapping functions, respectively.

There are several tropospheric models such as Hopefield and Sasstamoinen and mapping functions such as Niell Mapping Functions (NMF) and Isobaric Mapping Functions (IMF). For this research, the wet component is estimated with other parameters in PPP processing to reduce the residuals of the wet troposphere delay, based on the work by Kouba and Héroux (2001).

**Ionospheric delay**

The ionosphere, being the uppermost layer of the atmosphere and having a high density of electrons affects the propagation of the electromagnetic waves (Choy 2009). The density of the electrons primarily dependent on solar activity, the Earth's magnetic field, as well as the location of the user on the Earth (Misra and Enge 2006), cause errors ranging from a few metres to several tens of metres (Wells et al. 1999). The ionosphere delay is inversely proportional to the square of the transmission frequency. Dual-frequency GPS receivers can measure and remove the ionospheric effect by forming the dual-frequency, ionosphere-free linear combination. The combined technique is, however, being replaced by the uncombined technique. For this research slant ionospheric corrections, transmitted by IGS stations, were used to correct for the ionospheric error. Taking the dual-frequency combination induces noise due to large measurement noise in pseudorange observations. Global Ionospheric Maps (GIM's) are also used though they only correct for about 70-80% of the ionospheric errors.

### Multipath

Multipath noise is the biggest error source for code measurements from smartphones. Multipath is the error caused by the reflected signals entering the front end of the receiver and masking the real direct signal correlation peak (Parkinson and Spilker 1996). It is mainly caused by the reflected GNSS signals from the tall buildings in urban landscapes, trees or even the ground. In smartphones with linear antennas, the smartphone is unable to distinguish between the incoming right circularly polarised signal and the left circularly polarised signal. This error can be many tens of metres for C/A-code observations and up to a few centimetres for the carrier-phase measurements (Georgiadou and Kleusberg 1988). The most effective way to mitigate the multipath effect is to place the GNSS antenna away from the reflecting objects but for smartphones to work in realistic environments this problem is unavoidable. An effective way to counter this is to model this error in the measurement weighting. For low-cost receivers and smartphones, there is a strong correlation between the multipath and the $C/N_0$ values and will hence be used to estimate the weighting parameters for the code and carrier-phase measurements in the stochastic model. Multipath can also be mitigated at the receiver end by making GNSS receiver discriminator design less sensitive to multipath, that is, by a narrower early-late correlator spacing.

### Differential Code Biases

Differential Code Biases (DCBs) are systematic errors, or biases, between two GNSS code observations at the same or different frequencies. DCBs are primarily required for code-based positioning of GNSS receivers and extracting ionosphere total electron content (TEC). The application of DCBs is used for navigation applications and non-navigation applications such as ionospheric analysis and time transfer.

## 2.6 Real-Time versus Post-Processed PPP in Smartphones

On October 29, 2018, by Spaceopal Company in Germany announced GPS and Galileo orbit and clock corrections which are to be used with the broadcast ephemeris to convert it to the precise ephemeris counterpart. These products are produced based on the RETICLE algorithm developed by the German Aerospace Centre (DLR) and are called NAVCAST products. These real-time products are in addition to the broadcast navigation file and supply orbital and clock corrections to be applied to the products available in the navigation file.

Elmezayen and El-Rabbany (2019) used a Trimble R9 geodetic receiver and a Xiaomi MI-8 receiver with access to GPS L1 and L5 and Galileo E1 and E5 raw measurements and performed real-time PPP processing. The real-time precise ephemeris obtained from the NAVCAST GNSS PPP service were used in real-time PPP mode. As expected, the post-processed results were better than the real-time results in static processing. The results are shown in Table 2.4.

**Table 2.4: Mean and rms of East, North, and Up errors for the data set collected using Xiaomi MI 8 (Elmezayen and El-Rabbany 2019)**

| **PPP type** | **dE (cm)** | | **dN (cm)** | | **dU (cm)** | |
|---|---|---|---|---|---|---|
| | Mean | rms | Mean | rms | Mean | rms |
| **Post-Processed** | 36.4 | 50.7 | 31.0 | 47.3 | 30.5 | 55.9 |
| **Real-Time** | 43.0 | 55.8 | 42.9 | 54.2 | 42.4 | 62.1 |

Since the range of orbit and clock error, after applying the NAVCAST products were at the centimetre-level it is a viable option for processing smartphone data in real-time and in subsequent, PPP applications. Both post-processed and real-time PPP were implemented in the kinematic test case, but the results do not seem as promising and the smartphone positioning is still at the metre level for most cases. For one of the better datasets, collected by them, the rms of the post-processed PPP solution errors are 0.7 m, 0.6 m, 0.6 m in East, North, and Up directions, respectively, compared to 0.8 m, 0.6 m, and 0.7 m for the RT-PPP counterpart (Elmezayen and El-Rabbany 2019).

# Chapter 3 Measurement Quality Analysis and Quality Control

This chapter focuses on the GNSS receivers used for the data logging along with the software applications and the design of the YorkU PPP engine. The different types of datasets and the respective environments in which they have been collected are also described. This chapter also attempts to understand the nature of the smartphone data by analysing the different parameters (quality analysis) and to use the analysis to implement suitable checks which ensure integrity and resilience of the solution (quality control). Various aspects of the measurement quality will be analysed such as measurement noise, multipath, data gaps and carrier-to-noise-density ratio and their interdependence. The final portion of the chapter presents the various quality control measures implemented such as the pre-processing and post-processing quality control checks used in the PPP software.

## 3.1 Experimental Hardware, Data and PPP Processing Parameters

Every data collection requires some hardware and software for data logging. The measurements are subsequently processed in a PPP engine, with certain processing parameters. The first section of this chapter deals with a list of the hardware, i.e., the smartphone and the reference receiver, the software used, the datasets collected, and the data formats processed.

### 3.1.1 Smartphones and Receivers

Four models of Android phones were initially purchased to collect measurement data. The models and their GNSS hardware capabilities are listed in Table 3.1. The four phones vary in their capability of the number of GNSS constellations and frequencies they track. Dual-frequency multi-GNSS constellation tracking smartphones are new to the market and their positioning capabilities are yet to be fully researched. Though all were initially analysed for their measurement quality, all the data processing and analysis for this research were done using the Xiaomi MI 8 phone because the carrier-phase measurements from the Huawei Mate 20 suffer from bias, while the logged measurements do not follow the RINEX 3.03 format with missing spaces. Moreover, there are a high number of incorrectly logged measurements. The incorrect measurements could be both due to the application used and the smartphone antenna and receiver. Hence, it was not suitable to be used for PPP processing. The smartphone positioning results were compared against a relatively low-cost receiver, the SwiftNav Piksi. A geodetic grade receiver, the Topcon NET-G3A was used as the base station for kinematic PPP data collection. The smartphone chips are in the range of around 10 US dollars (USD) against the few hundred to few thousand USD chip prices for the other two receivers. These relatively expensive receivers were used to generate a reference solution. The SwiftNav Piksi tracks L1 and L2 signals from different GNSS constellations, the Topcon NET-G3A tracks L1 and L2 signals for GPS and GLONASS while the smartphones track L1/E1 and L5/E5a from some GPS and all Galileo Satellites.

**Table 3.1: Receivers used for data collection**

| Device | Frequency Band | Constellations Tracked | GNSS chipsets |
| --- | --- | --- | --- |
| **Xiaomi MI 8 (smartphone)** | L1, L5 | GPS(L1&L5), Galileo (E1, E5a) GLONASS (L1), BeiDou (B1), QZSS (L&L5) | Broadcom BCM47755 |
| **SwiftNav Piksi (reference)** | L1, L2 | GPS (L1&L2), Galileo (L1&E5b), GLONASS (L1&L2), BeiDou (B1&B2) | Xilinx Zynq 7020 |
| **Topcon NET-G3A** | L1, L2 & L5 carrier | GPS (L1, L2 & L5 carrier) , GLONASS (L1, L2 & L5 carrier) | Paradigm -G3 |

### 3.1.2 Data Loggers

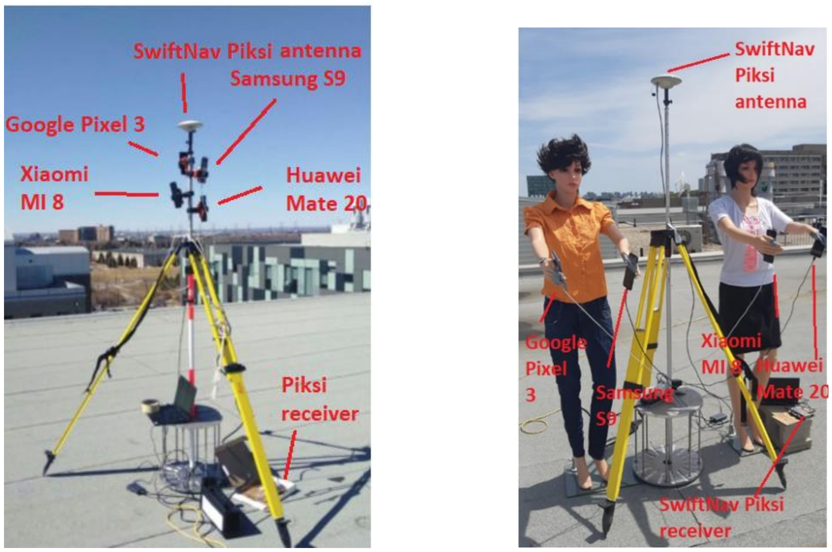
After raw code and carrier-phase measurements were made available to the public, the first app released was the GNSS Logger Application (Diggelen and Khider 2018). For this research, most of the data were collected using the Geo++ RINEX Logger (Geo++ GmbH 2018). A few datasets were collected with the RINEX ON app. For initial SPP processing, data were collected using the GNSS Logger app to analyse measurements through the GNSS analysis tools app from Google. Table 3.2 identifies the different loggers used for data collection, their developer, data output format and their logging capabilities.

**Table 3.2: Android applications used to collect GNSS measurements from smartphones**

| Logger | Organization | Output Format | Logged measurements | | |
| --- | --- | --- | --- | --- | --- |
| | | | Pseudoranges | Carrier-phase | Ephemeris |
| **GNSS Logger** | Google | NMEA | Yes | Yes | Yes |
| **Geo++ RINEX Logger** | Geo++ Gmbh | RINEX 3.03 | Yes | Yes | Yes |
| **RINEX ON** | Nottingham Scientific Ltd | RINEX 3.03 | Yes | Yes | Yes |

### 3.1.3 Datasets

For this research datasets were collected in static and kinematic mode and different environments. These datasets were collected by using mannequins and tripods on rooftops and by mounting phones on the dashboard of cars, in armbands or holding it in the hand. These scenarios are shown in Figure 3.1 A summary of the different datasets collected and processed is provided in Table 3.3.




(a) Static – Mannequin and Tripod


(b) Kinematic – driving (medium multipath) and walking (high multipath)

**Figure 3.1: Data collection in static and kinematic environments**

**Table 3.3: An overview of the different datasets collected, analysed and processed**

| Date | Mode | Duration | Environment | Description |
|---|---|---|---|---|
| **23 Feb 2019**<br><br>**DOY 54** | kinematic | 35 minutes | Urban (High multipath) | **Smartphone** held in hand while running |
| **7 Mar 2019**<br><br>**DOY 66** | kinematic | 30 minutes | Low to medium Multipath (Highway) | **Xiaomi MI 8** clamped to phone holder on the car dashboard |
| **8 Mar 2019**<br><br>**DOY 67** | kinematic | 10 minutes | Remote - Forested (High multipath) | **Xiaomi MI 8** phone kept in pocket while skiing |
| **26 Mar 2019**<br><br>**DOY 85** | kinematic | 25 minutes | Suburban (Medium multipath) | **Xiaomi MI 8** clamped to phone holder on car dashboard<br>**SwiftNav Piksi** on car roof |
| **26 May 2019**<br><br>**DOY 146** | static | 6 hours | Open sky (low multipath) | **Xiaomi MI 8** phone placed on hand of mannequin on Petrie Rooftop<br>**SwiftNav Piksi** on adjacent pole |
| **25 Oct 2019**<br><br>**DOY 298** | static | 2 hours | Open sky (low multipath) | **Xiaomi MI 8** phone clamped to pole on Petrie Rooftop<br>**SwiftNav Piksi** on top of same pole |
| **21 Nov 2019**<br><br>**DOY 325**<br><br>**(two tracks)** | kinematic | 25 and 30 minutes, respectiv ely | Suburban (Medium multipath) | **Xiaomi MI 8** lay flat on car dashboard<br>**SwiftNav Piksi** placed on car roof; connected to a low-cost Tallysman antenna and SwiftNav antenna, alternately in the two runs<br>**Topcon NET-G3A** receiver used as base station on Petrie Rooftop |
| **21 Dec 2019**<br><br>**DOY 355** | kinematic | 30 minutes | Urban (High multipath) | **Xiaomi MI 8** kept in the hand while walking |
| **11 Apr 2020**<br>**DOY 102 2020** | kinematic | 12 minutes | Urban (High multipath) | **Xiaomi MI 8** kept in the hand while walking with duty-cycling turned on |
| **23 Apr 2020**<br>**DOY 2020** | kinematic (real-time PPP) | 30 min | Medium multipath | **Xiaomi MI 8** in hand while walking |

### 3.1.4 Data Formats

Data formats define the organization of data in the file according to pre-set specifications, which are necessary for the efficient processing of data in various processing engines. Traditionally, National Marine Electronics Association (NMEA) format and different versions of RINEX file format have been used to log GNSS data. The GNSS Logger Format is a new addition since the advent of smartphone raw GNSS measurement logging and processing. It is necessary to describe the different data formats before moving onto the data collection and processing.

**RINEX Data Format**

Receiver Independent Exchange Format (RINEX) is a data interchange format for GNSS data. RINEX is an ASCII data format developed by the Astronomical Institute of the University of Berne for the easy exchange of GNSS data. RINEX is used as a post-processing file format for both static and kinematic applications (Manandhar 2017). The files in the RINEX format includes GNSS observation data, navigation messages, meteorological data, geostationary satellite data, satellite and receiver clock files. Currently, smartphone applications such as Geo++ RINEX Logger, RINEX On and G-Ritz Logger log data in version 3.03. The RINEX file contains logged pseudornage, carrier-phase, signal-to-noise-ratio (SNR) and doppler measurements for each satellite tracked at any given epoch. The RINEX-like file logged by the smartphone outputs the original SNR values in dBHz and not in the standardised scaled weighing scale between 1-9. The GNSS data collection in this format was processed in the YorkU PPP engine.

**Google GNSS Logger Data Format**

Google released its GNSS Logger application which was the first application to provide access to raw measurements. The measurements are in comma separated format and include outputs like pseudorange rate, received and transmitted time, accumulated delta range and rate, carrier-to-noise-density ratio, navigation message, the hardware clock and biases (Banville and Van Diggelen 2016). The GNSS analysis apps and the publicly released source code can be used for processing and obtaining a positioning solution. It resembles the NMEA format (Manandhar 2017). Most of the SPP solutions had initially been generated using files in this format.

### 3.1.5 YorkU PPP Measurement Processing Engine

The York GNSS PPP software is a scalable and modular GNSS PPP processor written and built-in C++ using Visual Studio in the Microsoft.NET platform. It is capable of processing triple-frequency (L1/L2/L5) measurements in post-processing and real-time from GPS, Galileo, GLONASS and BeiDou. It has been made modular and scalable to handle data formats such as RINEX, Radio Technical Commission for Maritime Services (RTCM) and GNSS Logger.

Figure 3.2 illustrates the software architecture of the YorkU PPP engine. The York-PPP engine has been developed by the GNSS laboratory research team at York University. (Seepersad 2012; Aggrey 2015). Only portions of the software relevant to smartphone processing are discussed here in detail. It consists of four main segments: data input module; error correction; data processing model (sequential least-squares, extended Kalman filter (EKF)) module; and the parameter output module. The user is required to specify processing parameters and input files in a configuration file. All supplied data are read and stored in internally defined structures before data-handling checks are performed. These data-handling checks comprise the data pre-processing module. It involves checking for missing measurements, predicting missing carrier-phase measurements as well as rejecting missing measurements. It also involves applying the necessary elevation and carrier-to-noise-density ratio measurement filters. The correction module depends on user-required data supplied in the form of an observation file, precise satellite orbits and clocks, ANTEX file, DCB file and ocean loading coefficients. The data processing model comprises the sequential least-squares filtering module, the EKF module or the EKF+IMU module. Users can specify their choice of a processing module in the configuration file. The corrected observation data goes through the filtering module where position estimates, as well as other parameters, are obtained. The output parameter segment is intended for evaluation and the analysis of results. The output files can be plotted using various MATLAB scripts. The important pre-processing and stochastic modules which are a contribution as a result of this research are highlighted in orange boxes.

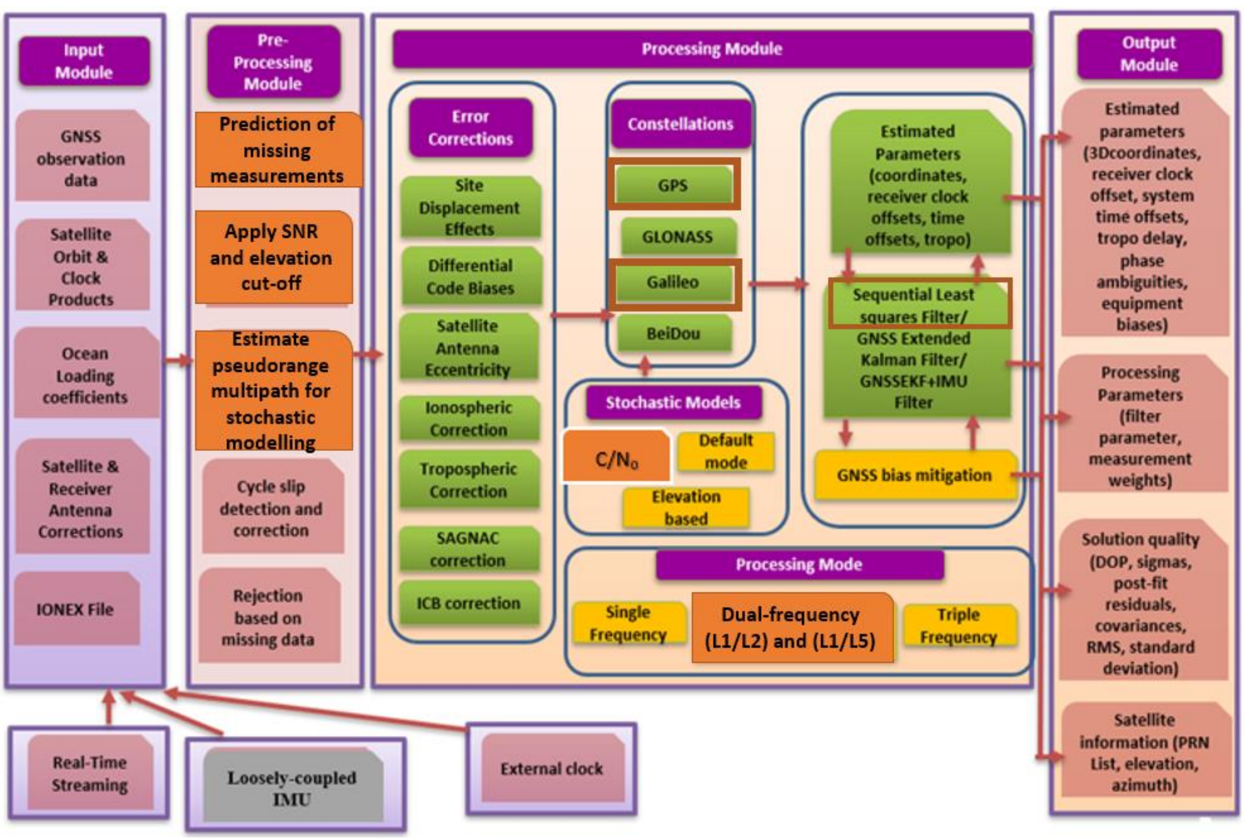



**Figure 3.2: Schematic representation of the YorkU PPP engine**

### 3.1.6 YorkU PPP Processing Parameters

Table 3.4 displays the processing parameters and settings used in the YorkU PPP processing engine for smartphones and the reference receivers. Precise orbits and clocks from Centre National d'Etudes Spatiales (CNES) and German Research Centre for Geosciences (GFZ) were used because of the availability of orbits and clocks for all GNSS constellations. The uncombined mode of processing was preferred because it makes use of the raw strength of the measurements. In the combined mode, the measurement noise for the two signals adds up, which degrades the accuracy of the positioning solution for smartphones.

**Table 3.4: YorkU-PPP processing parameters for smartphones GNSS processing**

| Processing parameters | York U GNSS PPP engine settings |
|---|---|
| PPP processing mode | Uncombined dual-frequency |
| Estimator | sequential least squares |
| Antenna corrections | IGS ANTEX |
| Satellite orbits and clocks | CNT (CNES) |
| Elevation mask | 10° |
| $C/N_0$ mask | 20 dBHz for smartphone, 15 dBHz for SwiftNav Piksi |
| GNSS system | GPS, GALILEO |
| Observations processed | L1, L5, E1, E5a |
| Measurement data format | RINEX 3.03 |
| Ionospheric mitigation | Slant ionospheric delay estimation |
| Tropospheric modelling | Hydrostatic delay: Davis (GPT)<br>Wet delay: Estimated<br>Mapping function: Global Mapping Function (GMF) |

The PPP processing was done using a sequential least squares filter since the variability in the measurement noise for different antennas in smartphones and different environments of data collection require extensive tuning of the noise parameters if processed using the Extended Kalman Filter. Moreover, such repeated tuning is not possible for Real-Time PPP Processing, which is what smartphone-based GNSS positioning is meant for. Appropriate elevation and $C/N_0$ measurement filters were applied to eliminate measurements with high noise which will be discussed in Section 3.3. Only GPS and Galileo measurements were processed since smartphones track L1 and L5 measurements for only those two constellations.

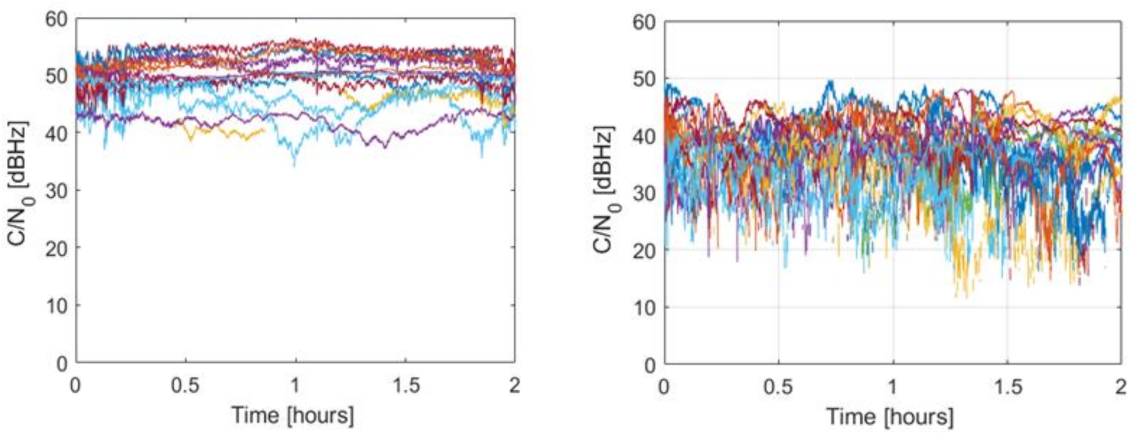
## 3.2 Analysis of Measurement Quality

The smartphone raw GNSS measurements were analysed for different quality parameters in various data collection environments. These quality parameters such as $C/N_0$, cycle slips, measurements noise and multipath and their interdependence will be discussed below with several examples. These parameters will also be compared against the values and plots obtained for a higher-quality SwiftNav Piksi to gain a better understanding and for quantitative analysis. The frequency of occurrence and length of data gaps for different observables will be looked at for different multipath environments since it is crucial to customize and condition the measurements which will be discussed in Chapters 4 and 5.

### 3.2.1 Carrier-to-Noise-Density Ratio

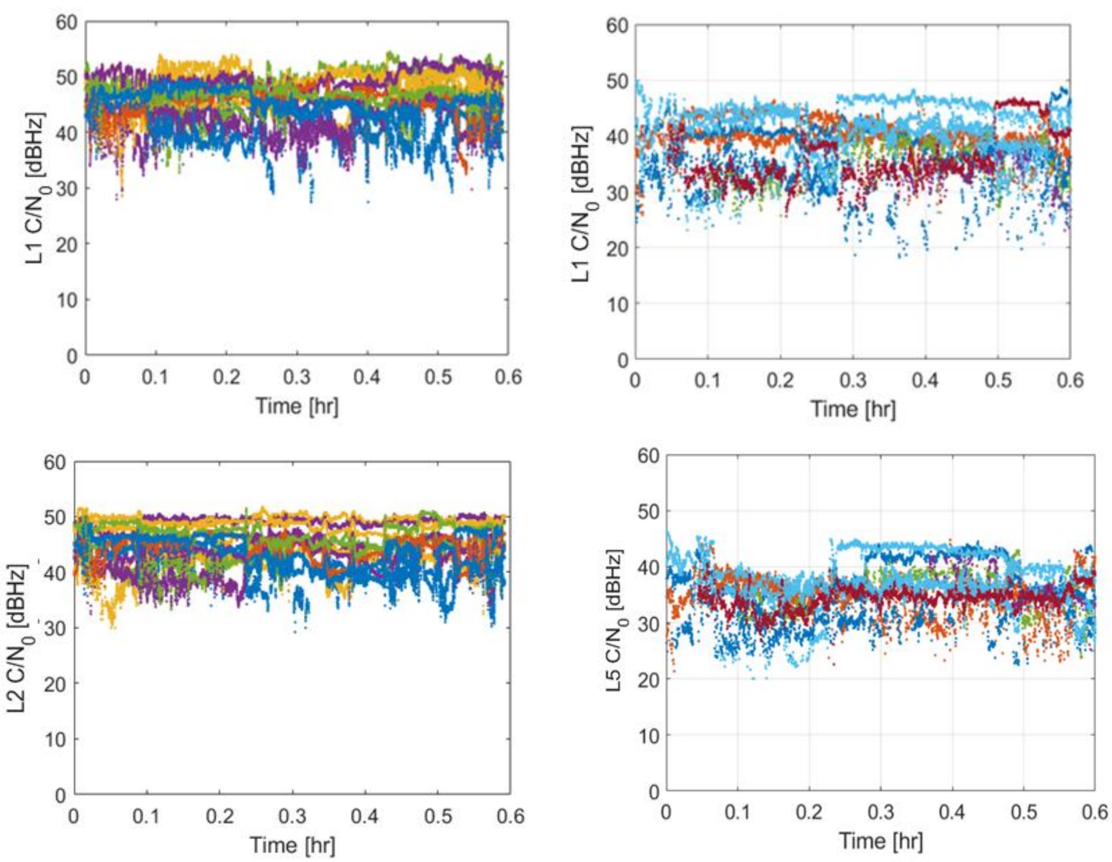
Carrier-to-noise-density ratio ($C/N_0$) is defined as the ratio of the signal carrier power to the noise power in a 1 Hz bandwidth. In other words, it is the power in the received signal compared to the power spectral density of the receiver noise. This indicator strongly depends not only on the satellite antenna, signal propagation and deterioration but also on the hardware, including antenna, receiver and cables (Montenbruck et al. 2012; Wang et al. 2012). The four major factors influencing $C/N_0$ received at the antenna phase centre are (1) power density of the incoming GNSS signal; (2) reception area of the antenna; (3) receiver antenna gain; and (4) satellite elevation (Braasch and van Dierendonck 1999; Fortunato et al. 2019). Since the $C/N_0$ of the received signal depends on hardware such as the quality of the antenna, smartphones with cheap linearly polarised antennas have lower $C/N_0$, as compared to geodetic and survey-grade receivers, or even the reference receiver used in this research, the SwiftNav Piksi. The increased variability in $C/N_0$ impacts the quality of the measurements and these signals have higher measurement noise. Another cause for this low and irregular values of $C/N_0$ is the inability of the smartphone monopole antennas to distinguish between incoming right-circularly polarised signals and reflected left-circularly polarised signals. Such signals have high multipath and using such measurements affect the positioning solution quality (Chen et al. 2019; Robustelli et al. 2019; Guo et al. 2020).

The L1 and L2/L5 $C/N_0$ for the two receivers were analysed in both the static and kinematic scenario as shown in Figure 3.3 and Figure 3.4. The $C/N_0$ for the smartphone is on an average 10-15 dBHz lower than that of the SwiftNav Piksi and has several jumps when it picks up reflected signals. The above-mentioned observations are in accordance with observations made by several researchers (Chen et al. 2019; Robustelli et al. 2019; Guo et al. 2020). In a high multipath environment, there was a high percentage of signals with $C/N_0$ as low as 10 dBHz as shown in Figure 3.5 with a mean of around 28 dBHz. When signals having suffered from multiple reflections are picked up the smartphone antenna, the $C/N_0$ values are extremely low and such measurements have high multipath noise. The plot for the Xiaomi MI 8 $C/N_0$ in Figure 3.3 and 3.4 also show gaps which highlight the periodic loss-of-signal by the low-cost antenna in kinematic scenarios.




(a) SwiftNav Piksi (b) Xiaomi MI 8

**Figure 3.3: L1 C/$N_0$ plot for the Xiaomi and SwiftNav Piksi in a static low multipath scenario (DOY 146, 2019)**




(a) SwiftNav Piksi (b) Xiaomi MI 8

**Figure 3.4: L1 and L2/L5 C/$N_0$ plot for the Xiaomi and SwiftNav Piksi in a kinematic medium multipath scenario (DOY 325, 2019)**

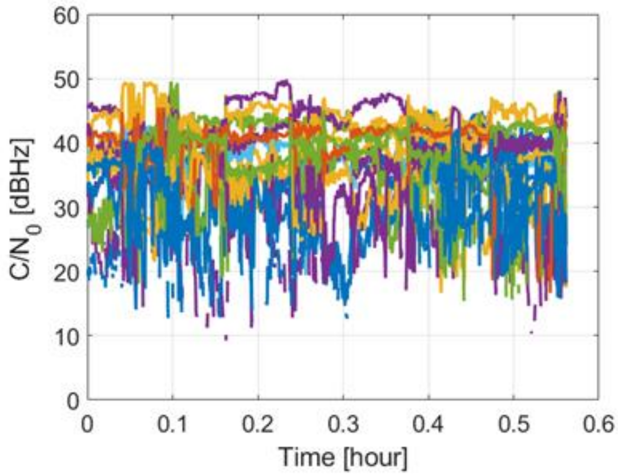



**Figure 3.5: C/N$_0$ plot for the Xiaomi MI 8 in a high multipath kinematic scenario (DOY 355, 2019)**

The mean C/N$_0$ value of the signals for GPS and Galileo was computed. The C/N$_0$ of the smartphone L1 signal was 27% lower than the reference receiver in the static scenario, 23% lower in the kinematic medium multipath scenario and 24% lower in the high multipath scenario, as shown in Table 3.5. Similar results were observed for the L5/E5 signal as well. Noticeably, the L5/E5a signal for the smartphone has a lower signal strength compared to the L1/E1 signal while the L2/E5b signal for the SwiftNav Piksi had a higher signal strength than the corresponding L1/E1 signal. The following conclusions can be drawn:

1. The signal strength is low and irregular with drastic degradations corresponding to increase in multipath.

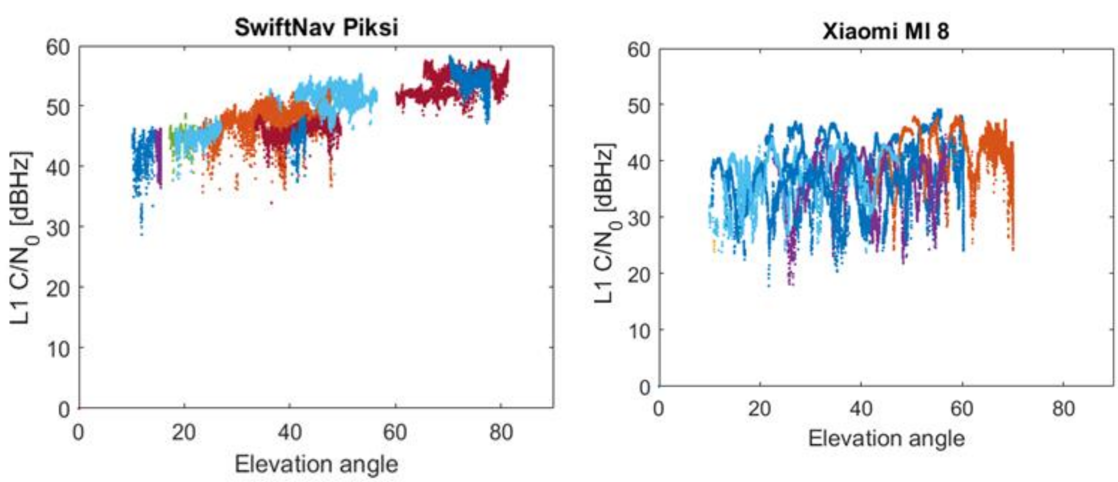
2. The C/N$_0$ values for smartphones can be as low as 10 dBHz. Using such signals, the code measurements have large multipath noise affecting the accuracy of the positioning solution.

**Table 3.5: Mean $C/N_0$ for Xiaomi and SwiftNav Piksi in different multipath scenarios**

| Receiver | Static Mean $C/N_0$ (dBHz) | | Kinematic Medium Multipath Mean $C/N_0$ (dBHz) | | Kinematic High Multipath Mean $C/N_0$ (dBHz) | |
|---|---|---|---|---|---|---|
| **Frequency** | **L1** | **L2/L5** | **L1** | **L2/L5** | **L1** | **L2/L5** |
| **Xiaomi Mi 8** | 35 | 34 | 35 | 33 | 29 | 26 |
| **SwiftNav Piksi** | 49 | 49 | 47 | 48 | 44 | 44 |
| **Difference** | 27 | 29 | 23 | 31 | 34 | 40 |

### $C/N_0$ versus the Elevation Angle

The distribution of the magnitude of $C/N_0$ versus the elevation angle also reveals certain interesting characteristics. Figure 3.6 represents the plot for $C/N_0$ as a function of elevation angle. For the SwiftNav Piksi, a clear linear relationship is observed between higher satellite elevation angle and higher $C/N_0$. As antennas in smartphones track signals from all sides, they are unable to distinguish between multipath and incoming signals and signals from higher elevation angle satellites also suffer from multipath. These results correspond to similar findings by Paziewski et al. (2019) and Wanninger and Heßelbarth (2020) with other smartphones such as the Huawei P 30.



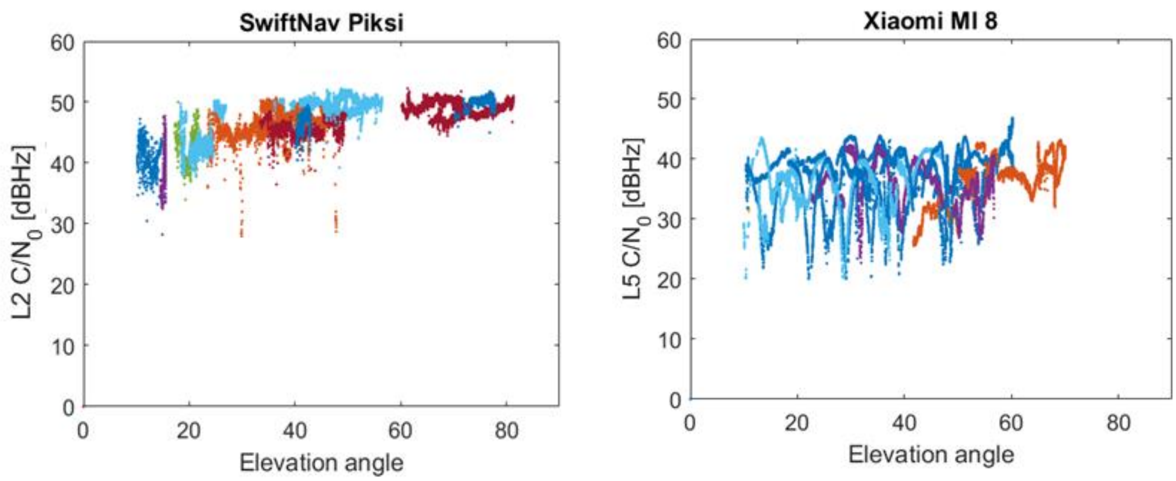
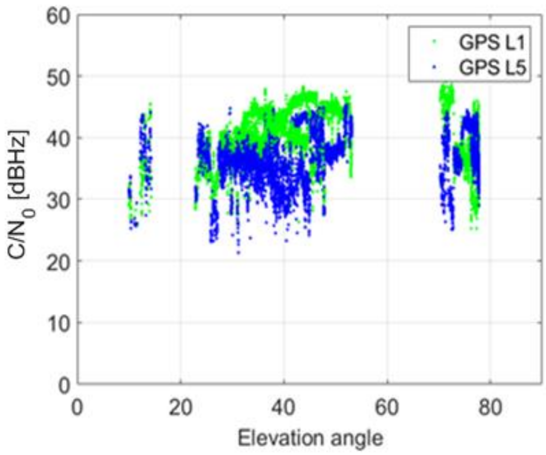


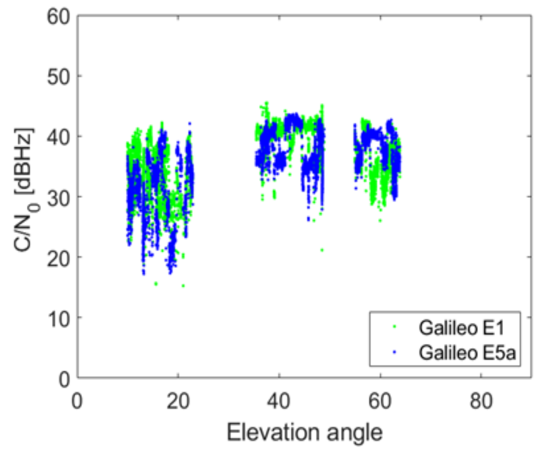

**Figure 3.6: C/N$_0$ as a function of elevation angle for satellites tracked by Xiaomi MI 8 and SwiftNav Piksi in a kinematic medium multipath scenario (DOY 325, 2019)**

**Comparison of C/N$_0$ for different Constellations and Frequencies**

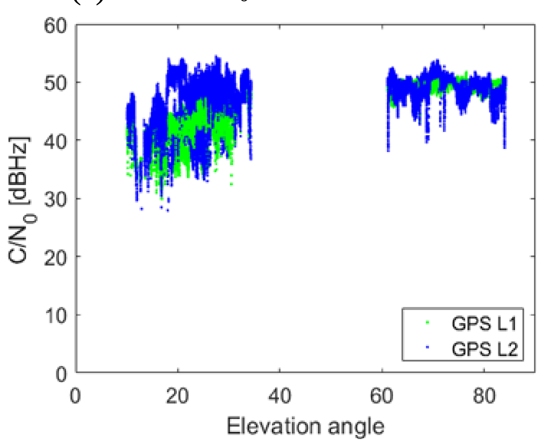
The L2/E2 and L5/E5a signals are transmitted at a higher chipping rate and lower frequency than the L1 signal. Ideally, for most receivers, they are less affected by noise and should have higher C/N$_0$ values (Leclère et al. 2018). Figure 3.7 shows the relation between the mean C/N$_0$ of L1/E1 and L5/E5 signals from a smartphone for GPS and Galileo. For the same elevation angle, the signal strength for the L1 signal is higher than for L5, which ideally should not be the case for other better-quality receivers. Table 3.6 shows the average C/N$_0$ for the two frequencies for the static and kinematic scenarios.

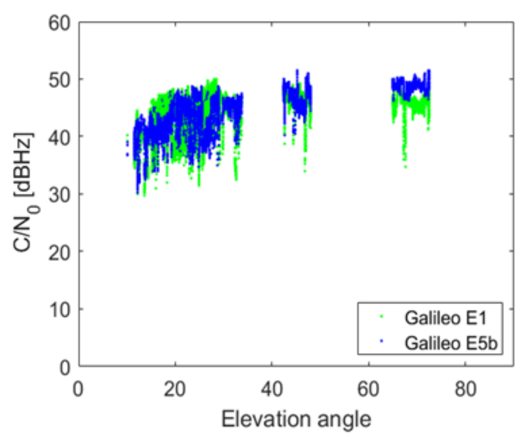
With a geodetic receiver and antenna, the C/N$_0$ for the L5/E5 or L2/E2 signal band is approximately 5 and 3 dBHz higher than the L1/E1 measurements. However, for the smartphone, the received L1 and E1 signals are considerably stronger than L5/E5a, almost certainly due to the antenna, especially in medium to high multipath environments. The C/N$_0$ for GPS L1 is an average 3.1 dBHz stronger than L5 and Galileo E1 3.4 dBHz stronger than E5a. The smartphone antenna is notably, not as sensitive for L5/E5a signals as compared to L1/E1. Also, for L5/E5a, it is observed that the $C/N_0$ decreases above an elevation angle of 50° in high multipath environments and above 60° in static open sky conditions. These findings correspond to results from Wanninger and Heßelbarth (2020) in static open sky environments.



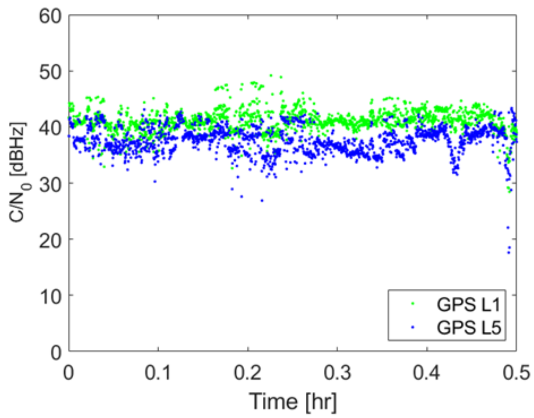




(a) GPS $C/N_0$ versus elevation– Xiaomi (b) Galileo $C/N_0$ versus elevation – Xiaomi

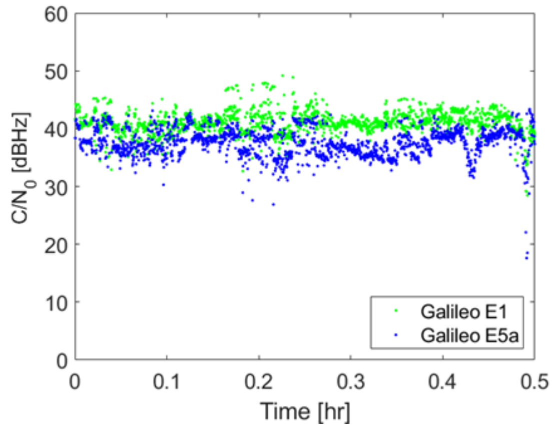



(c) GPS $C/N_0$ versus elevation - Piksi (d) Galileo $C/N_0$ versus elevation - Piksi




(e) GPS C/N$_0$ comparison-Xiaomi MI 8




(f) Galileo C/N$_0$ comparison-Xiaomi MI 8

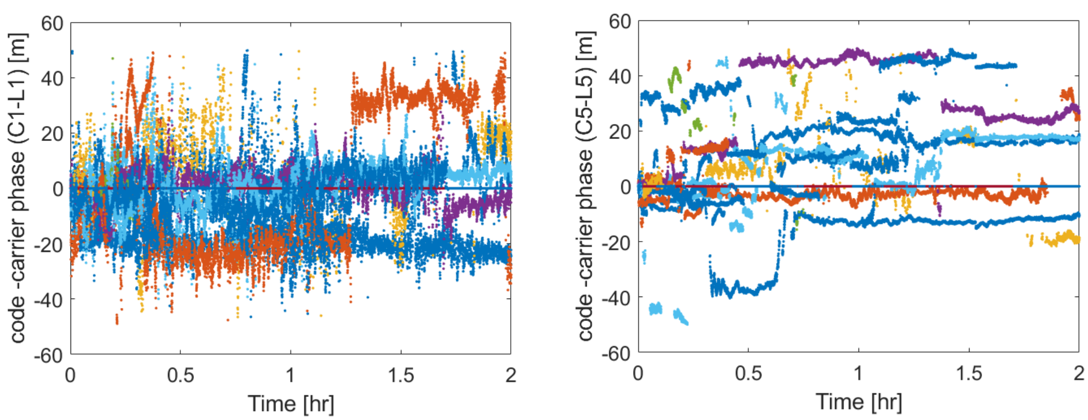
**Figure 3.7: C/N$_0$ plot comparison for L1/E1 (green) and L2/L5/E5(blue) frequencies for Xiaomi MI 8 and SwiftNav Piksi in a kinematic medium multipath scenario**

**Table 3.6: Mean C/N$_0$ values for L1 and L2/L5 frequencies for GPS and E1 and E5 frequencies for Galileo for Xiaomi MI 8 and Piksi**

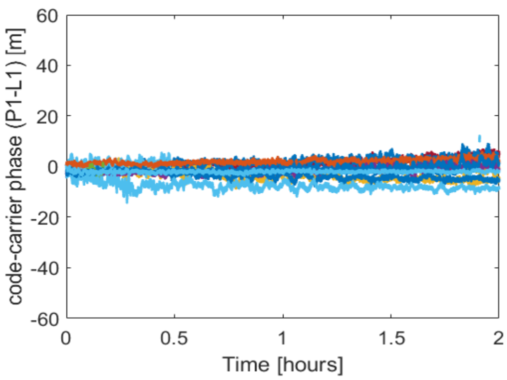
| Device | Scenario | Frequency | C/N$_0$ [dBHz] | |
|---|---|---|---|---|
| | | | GPS | Galileo |
| **Xiaomi MI 8** | Static | L1/E1 | 39 | 36 |
| | | L5/E5a | 37 | 36 |
| | Kinematic | L1/E1 | 39 | 35 |
| | | L5/E5a | 36 | 32 |
| **SwiftNav Piksi** | Static | L1/E1 | 48 | 49 |
| | | L2/E5b | 48 | 50 |
| | Kinematic | L1/E1 | 47 | 48 |
| | | L2/E5b | 48 | 49 |

### 3.2.2 Measurement Noise

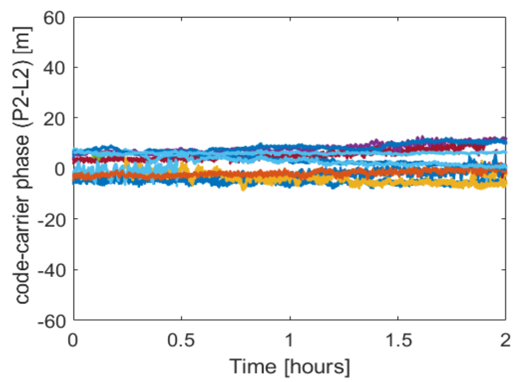
Another measure is the measurement noise computed by differencing the code and carrier-phase measurements. The difference of C/A-code and carrier-phase measurements represents the combined effect of pseudorange and carrier-phase multipath and signal noise with the assumption that the atmospheric error is negligible. The measurement noise was analysed for both the smartphone and the SwiftNav Piksi in the same static and multipath environment and is shown in Figure 3.8 and Figure 3.9. Due to the use of a linearly polarised antenna in smartphones, it was expected that the quality of the measurements will deteriorate in comparison to a better-quality receiver and antenna (Gill et al. 2017; Gill 2018). As expected, the rms of the measurement noise for the smartphone is several metres more than the SwiftNav Piksi, as shown in Table 3.7. The large biases in the measurement noise for the smartphone correspond to undetected cycle slips. Moreover, the frequent phase-loss-of-lock and missing carrier-phase measurements lead to non-continuities and jumps in the observed noise pattern.




(a) Xiaomi MI 8 C1-L1 noise level (b) Xiaomi MI 8 C5-L5 noise level

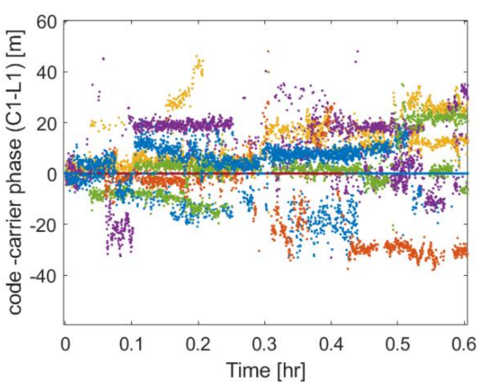



(c) SwiftNav Piksi P1-L1 noise level

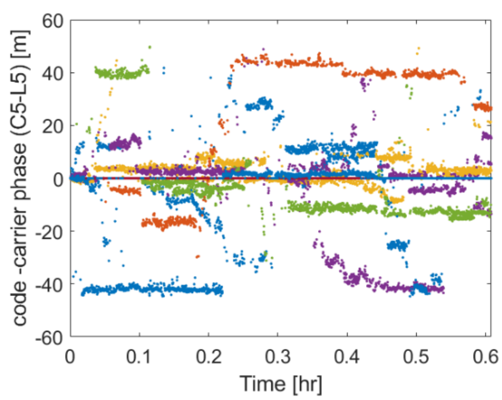



(d) SwiftNav Piksi P2-L2 noise level

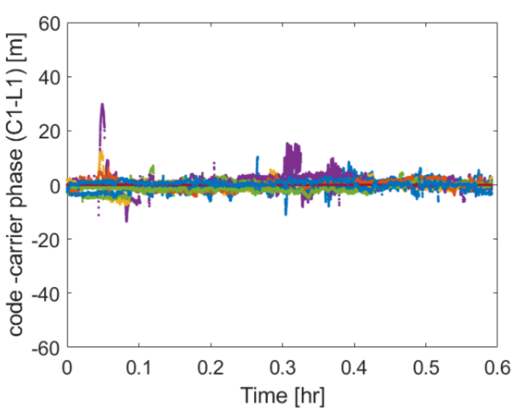
**Figure 3.8: Signal noise level estimate (in metres) for Xiaomi Mi 8 and SwiftNav Piksi receiver in static testing (DOY 146, 2019)**



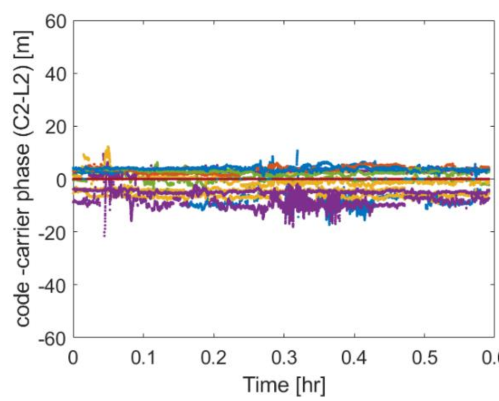

(a) Xiaomi MI 8 L1-C1 noise level




(b) Xiaomi MI 8 L5-C5 noise level




(c) SwiftNav Piksi L1-P1 noise level




(d) SwiftNav Piksi L2-P2 noise level

**Figure 3.9: Signal noise level estimate (in metres) for Xiaomi Mi 8 and SwiftNav Piksi receiver in kinematic testing (DOY 325, 2019)**

**Table 3.7: Measurement quality analysis for the Xiaomi MI 8 versus SwiftNav Piksi**

| Scenario | Device | Measurement Noise | |
|---|---|---|---|
| | | C1 -L1 [m] | P2-L2 / C5-L5 [m] |
| **Static** | Xiaomi MI 8 | 11.8 | 18.5 |
| | SwiftNav Piksi | 2.1 | 3.5 |
| **Kinematic** | Xiaomi MI 8 | 13.9 | 20.3 |
| | SwiftNav Piksi | 2.9 | 4.6 |

The measurement noise for the L5 signal is significantly higher than L1, which corresponds to the lower received signal strength for the L5 signal. The lower $C/N_0$ value for the L5 signal is an anomaly since it is transmitted at a higher signal strength and should be of better quality, but the current smartphone antenna is not designed to receive L5 signals accurately. The L5 signal has 56% more noise than L1 in the static case and 45% more in the kinematic case. Since the measurement noise depicts the combined effect of multipath and signal noise, analysing the multipath observable is crucial and will be performed next.

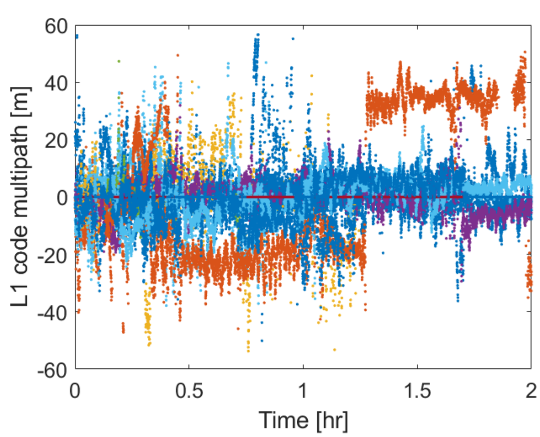
### 3.2.3 Multipath

Multipath affects observations from all satellites, irrespective of elevation angle since the antenna tracks signals from all directions and the received Carrier-to-noise-density ratio of the signals from different satellites is an indicator of the multipath that affects the code measurements. Multipath is not only affected by the quality of the hardware used, but also by the environment in which the data are collected. There is a multipath error in both pseudorange and carrier-phase observations. However, the carrier-phase multipath is limited to a few (4-5) cm and can be ignored given smartphone noise levels, while the multipath affecting the code observations can be up to a few tens of metres. The positioning solution deteriorates if noisy measurements are included in the processing. To implement this process, multipath observables are generated for both L1 and L5 frequency by making use of the standard multipath estimation formula (Seepersad 2012). The mean of the observables has to be subtracted to generate the final code multipath value.

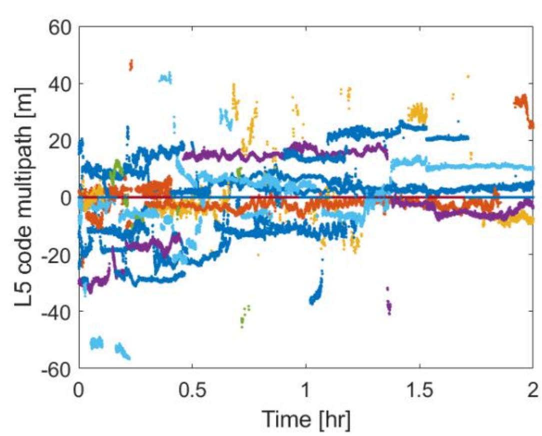
$$MP_1 = P_1 - \left(1 + \frac{2}{\alpha - 1}\right)\phi_1 + \left(\frac{2}{\alpha - 1}\right)\phi_5 \tag{3.1}$$

$$MP_5 = P_5 - \left(\frac{2*\alpha}{\alpha - 1}\right)\phi_1 + \left(\frac{2*\alpha}{\alpha - 1} - 1\right)\phi_5 \tag{3.2}$$

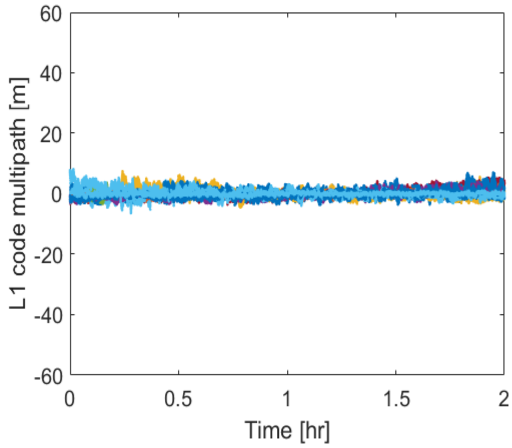
Multipath affecting the pseudorange measurements is prominent in an urban or suburban area, in forested areas and or when the car turns or crosses an intersection for kinematic data collection. The same static and kinematic dataset analysed for measurement noise in Section 3.2.2 on measurement noise were analysed for the code multipath. Figure 3.10 and 3.11 show the MP1 and MP5 observable for the smartphone and the MP1 and MP2 observable for the SwiftNav Piksi. Once again, undetected cycle slips cause the large biases in the multipath plot for the smartphone. Missing carrier-phase measurements are the cause of gaps in the multipath plot for all satellites.



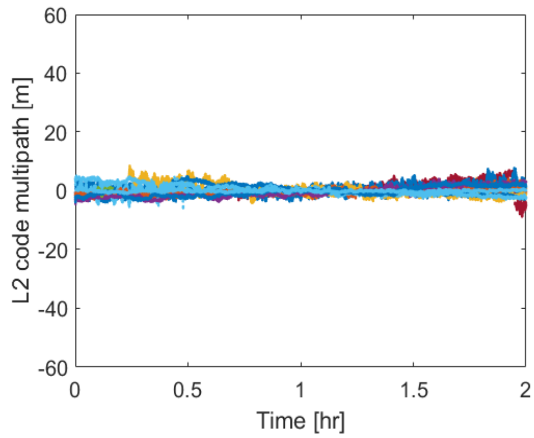

(a) Xiaomi MI 8 L1 multipath




(b) Xiaomi MI 8 L5 multipath

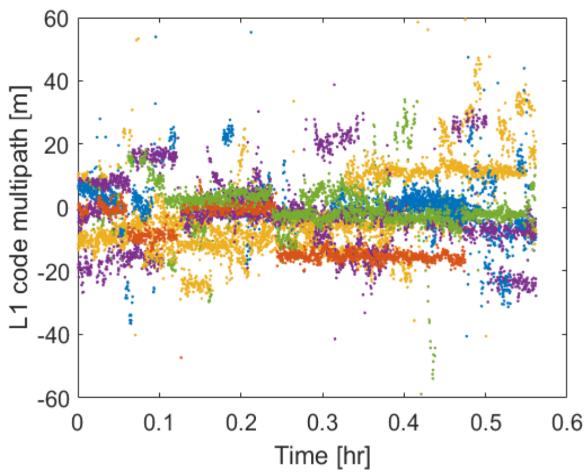



(c) SwiftNav Piksi L1 multipath

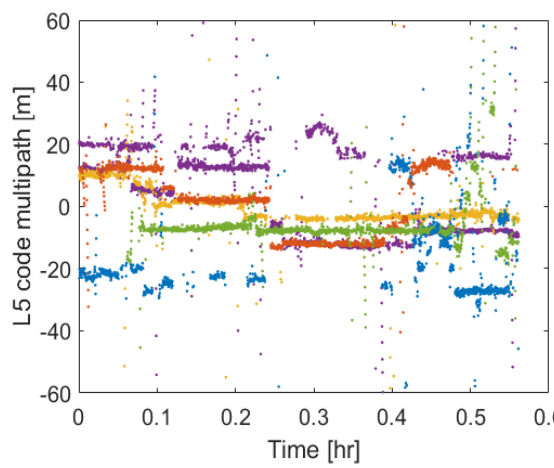



(d) SwiftNav Piksi L2 multipath

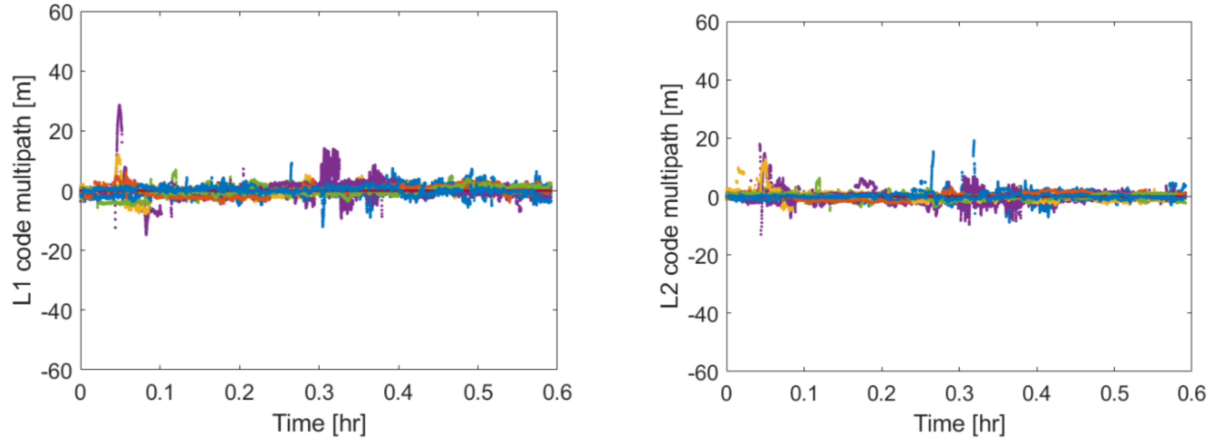
**Figure 3.10: Code multipath for Xiaomi MI 8 and SwiftNav Piksi in a static low multipath environment (DOY 146, 2019)**




(a) Xiaomi MI 8 L1 multipath




(b) Xiaomi MI 8 L5 multipath




(c) SwiftNav Piksi L1 multipath (d) SwiftNav Piksi L2 multipath

**Figure 3.11: Code multipath for Xiaomi MI 8 and SwiftNav Piksi receiver in a kinematic medium multipath environment (DOY 325, 2019)**

The multipath effect on the signals for Xiaomi MI 8 is on an average 84% higher than that of the effect on the SwiftNav Piksi in a static environment and 83% higher in the kinematic environment. The multipath rms of the smartphone and the reference receiver is shown in Table 3.8. These rms values are crucial and will be used in the stochastic modelling, discussed in Chapter 5, to weigh the pseudorange measurements. It is interesting to note that the effect of multipath is higher on L5 code measurements than the L1 for the smartphone, while it is lower for the L2 code measurements than L1 for the SwiftNav Piksi.

**Table 3.8: Multipath analysis for the Xiaomi MI 8 versus SwiftNav Piksi**

| Scenario | Device | Multipath | |
|---|---|---|---|
| | | C1/P1 [m] | C5/P2 [m] |
| **Static** | Xiaomi MI 8 | 8.4 | 13.2 |
| | SwiftNav Piksi | 1.4 | 1.1 |
| **Kinematic** | Xiaomi MI 8 | 10.2 | 16.1 |
| | SwiftNav Piksi | 1.7 | 1.3 |

**$C/N_0$ versus Multipath**

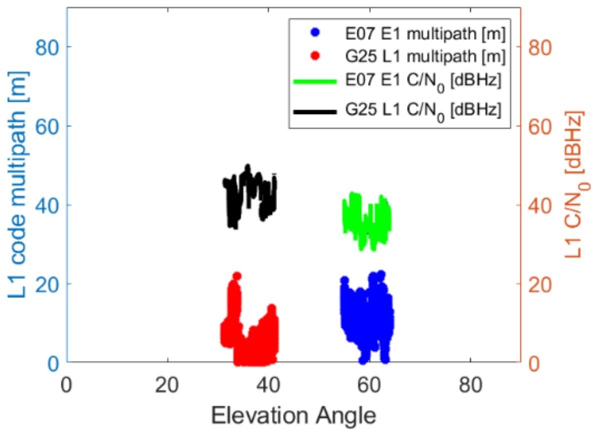
For the smartphone, a clear relationship between multipath and the $C/N_0$ can be established, especially in a medium-to-high multipath environment. Since the smartphone antenna senses reflected signals from all directions, the multipath is related to the $C/N_0$ rather than the elevation angle as observed in Figure 3.12. Two inferences that can be drawn from Figure 3.12(a), (b) and (c) are:

1). The elevation angle does not influence $C/N_0$ values and multipath for smartphones. G25 with a lower elevation angle of 35° as compared to 60° for E07 had a higher mean $C/N_0$ of 45 dHBz and lower L1 code multipath with an rms of 5.9 m. E07 had a mean $C/N_0$ and rms multipath of 26.5 dBHz and 10.4 m, respectively. (Figure 3.12(a)).

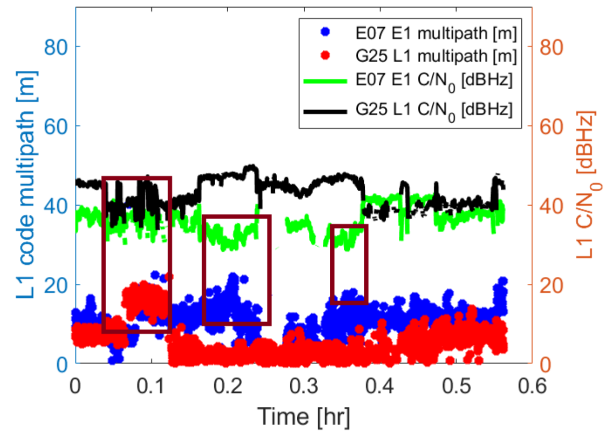
2). There was a general decrease in multipath with increase in $C/N_0$ values as highlighted by the brown boxes in Figure 3.12(b) implying that the antenna picked up reflected signals with weaker signal strength.

3) The L5/E5 signals are transmitted at higher power levels and chipping rate than L1/E1 and therefore should have higher received $C/N_0$ and better noise suppression. However, the tested smartphone shows the opposite with received $C/N_0$ for the L5 signal for, e.g., G25 in a medium multipath environment on an average is 3 dBHz lower than for L1, while the pseudorange multipath rms for L5 is ~5 m more than for L1 as shown in Figure 3.12(c). The smartphone antenna affects these signal strength and noise values as it not as sensitive for the L5/E5a signal (Wanninger and Heßelbarth 2020).

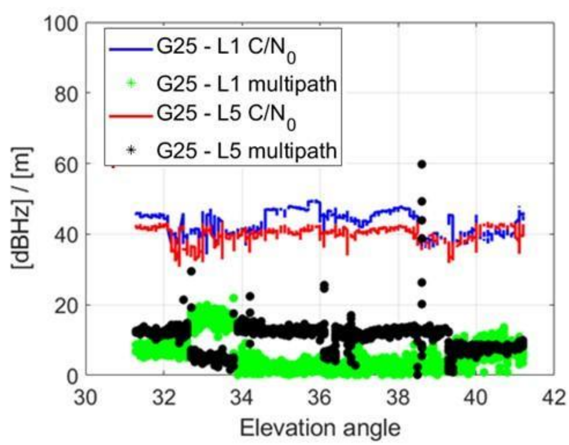
Based on this analysis, one way to model this multipath and its relationship with C/$N_0$ is to have a weighting strategy for pseudorange measurements that considers the C/$N_0$ and multipath along with the pseudorange chipping-length which is different for the L1 and L5 frequency band.




(a) C/$N_0$ and multipath as a func of elevation angle




(b) C/$N_0$ and multipath as a func of time




(c) Comparison of C/$N_0$ and multipath values for the L1 and L5 signal

**Figure 3.12: Variation in C/$N_0$, elevation angle and pseudorange multipath for the L1 and L5 frequency for different satellites in a kinematic, medium multipath scenario**

### 3.2.4 Cycle Slips

Cycle slips affect the precise carrier-phase measurements. At any epoch, the carrier phase measurement comprises the observed accumulated phase $\Delta\phi$ and the integer number of wavelengths N, also known as the ambiguity term. The receiver constantly tracks measures of this accumulated phase, whereas N remains constant if there is no loss of signal (Hofmann-Wellenhof et al. 2007). Cycle slips are caused by the obstruction in the line satellite-receiver line of sight because of trees, buildings, etc. They are also caused by low Carrier-to-noise-density ratio and multipath, which is very common in smartphones due to the low-quality antenna. The low-quality antennas in smartphones suffer from frequent phase-loss-of-lock which cause cycle-slips, in which the float ambiguities must be reset. These cycle slips, if not detected and corrected for cause positioning errors. Currently, the Melbourne-Wubenna combination is being used in the YorkU PPP code to detect cycle slips. (Bezmenov et al. 2019).

Figure 3.13 shows the cycle slips for 4 GPS satellites for a kinematic dataset collected in a forested area on DOY 67, 2019, while skiing. The satellite single-difference method was used to detect cycle slips where the carrier-phase residuals between a reference satellite and all other satellites are taken to identify a potential-jump in value, which is associated with a cycle slip. GPS PRN 31 was taken as the reference satellite as it had the highest elevation angle. It was not possible to use double or triple-differencing approach since the dataset was collected in the presence of a single receiver. Multiple and periodic cycle slips are observed which can be identified by the black crosses in the carrier-phase residuals. Most of these cycle slips are followed by gaps highlighting missing measurements as the antenna faces a phase-loss-of-lock. Further cycle slip plots are shown in Section 3.2.5. Understanding the occurrence of cycle slips is necessary since they are the cause of frequent data gaps which will be discussed in the next section.

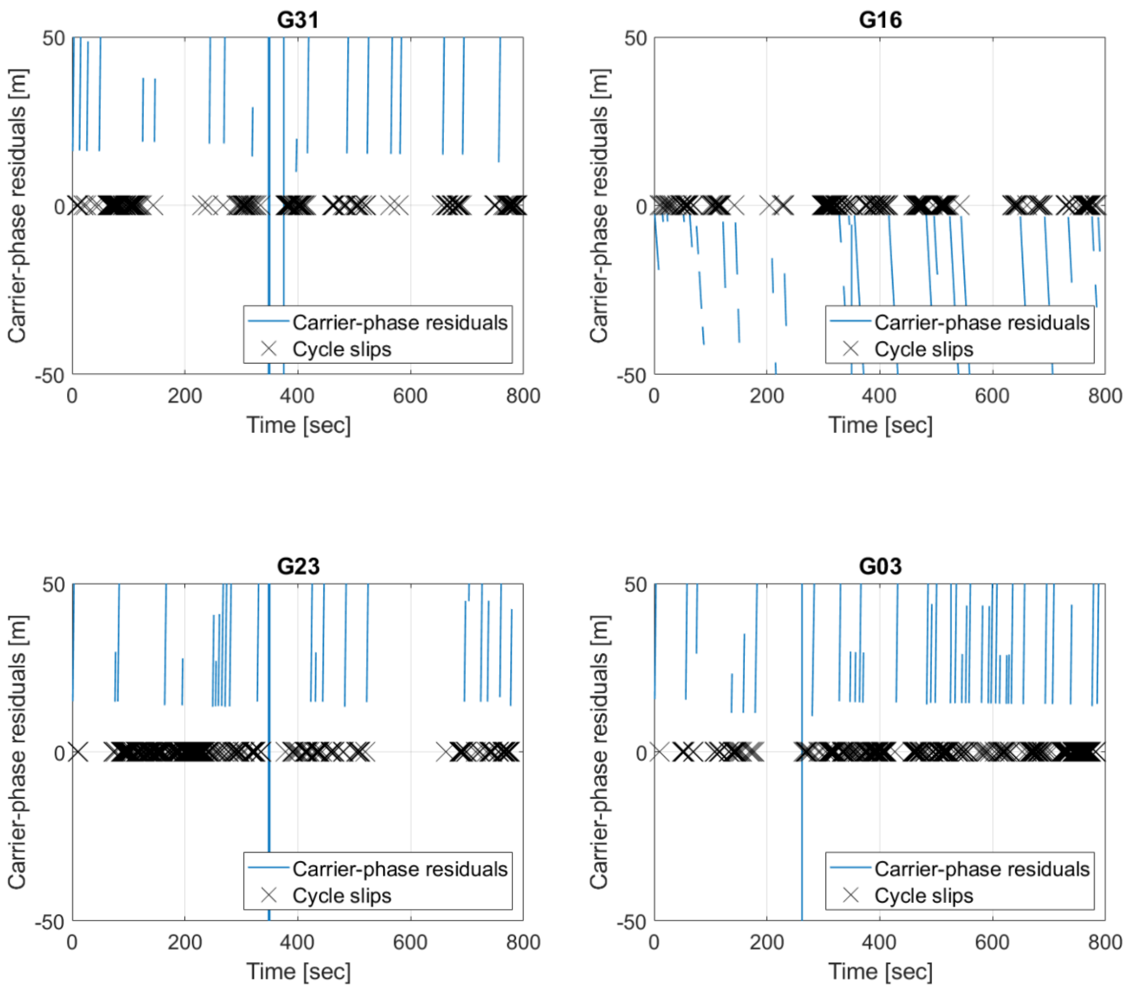



**Figure 3.13: Carrier-phase residuals showing cycle slips (black crosses) at regular intervals for 4 GPS satellites for a kinematic dataset collected in a remote forested environment (DOY 67, 2019)**

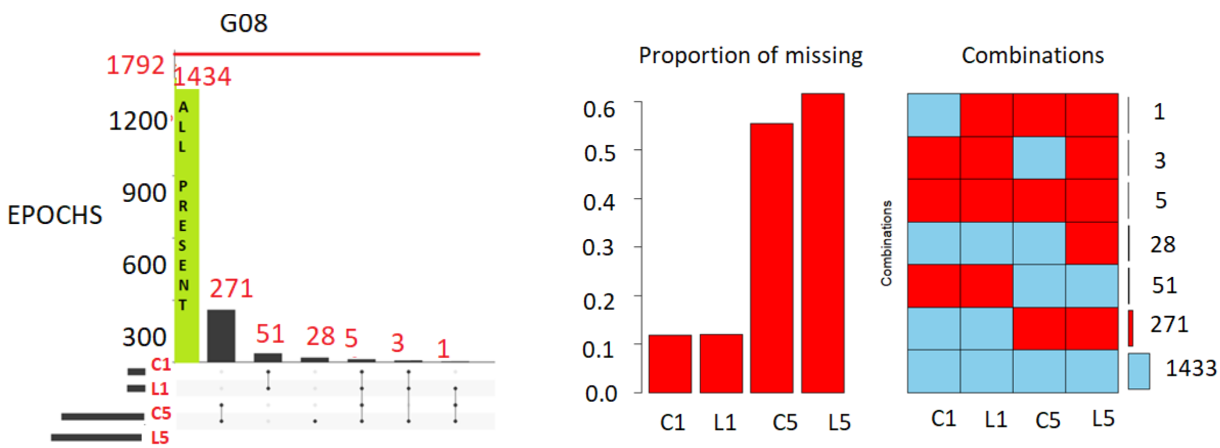
### 3.2.5 Data Gaps

Due to signal blockages, reflections, or high multipath, especially in obstructed environments such as urban and forested areas, the receiver's antenna suffers from loss of phase lock. A receiver loses track of the signal and takes an average of 3-5 seconds before re-acquiring it, causing missing measurements. The carrier phase measurements are more vulnerable to signal blockages than pseudorange and Doppler measurements (Lachapelle et al. 1992; Curran 2015; Marçal and Nunes 2016). When the GNSS signal is affected by blocking or interference the phase-locked loop (PLL) of the receiver loses lock and the carrier phase is not continuous (Sennott 1999; Ge et al. 2008).

Missing measurements in the logged data pose the biggest challenge in processing data from smartphones. Missing code or carrier-phase observations causes rejection of the satellite in PPP processing. In suburban, urban, or even open sky environments, there are signal blockages due to tall structures such as buildings and trees and the receiver antenna can track a maximum of 6-8 satellites. There are missing measurements present for some of the satellites tracked. In many cases, the total satellites available after filtering out satellites with low elevation angle, low $C/N_0$, large code and carrier-phase residuals and missing measurements are two to three. To generate a positioning solution in PPP dual-frequency processing, a minimum of 5 satellites is needed since we have two ambiguity and an ionosphere state per satellite, besides the receiver coordinates.

Various datasets in different multipath scenarios were analysed to understand the ratio of missing measurements and their frequency of occurrence. In the first scenario, which is an ideal open-sky static environment with minimal multipath, it is observed that there are data gaps, through the duration of the gaps is small and their frequency of occurrence is also small. Figure 3.14 depicts the trend and percentage of missing observations for G08. Below the bar graph are dots that represent the missing observables whose count is being depicted in the bar graph. Bars with multiple dots connected with a line show the absence of multiple observations simultaneously (Figure 3.14(a)). The numbers in red indicate the duration in seconds (epochs). Another way of representing the trend of missing observations is to also see which observable is mostly absent.

The left side of the figure highlights the proportion of missing observables in red and the different combinations with the epoch count are depicted on the right (Figure 3.14(b)).  The conclusion drawn is that in ideal static scenarios at least 80% of epochs had all the observables present for a given satellite.




(a) Duration of missing observations (b) Combination of missing observations

**Figure 3.14: Trend and percentage of missing observables for G08 for static data collected in a low multipath environment (DOY 146, 2019)**

The data gap analysis of kinematic datasets aid in understanding the behaviour of the smartphone measurements, since most applications require a moving user, either walking or driving with the smartphone. Two kinematic datasets collected in a medium multipath environment were taken and the data gaps for different satellites were analysed. For the kinematic track 2 collected on DOY 325, 2019 the observations for G01 are analysed. It is observed that only 657 measurements were present which is a meagre 30% (657/2189). Figures 3.15 and 3.16 depict the relationship between the missing observations for G01 satellite and cycle slips. The cycle slip plot for G01 was computed by taking G23 as reference. Black crosses indicate cycle-slips and subsequent phase-loss-of-lock.

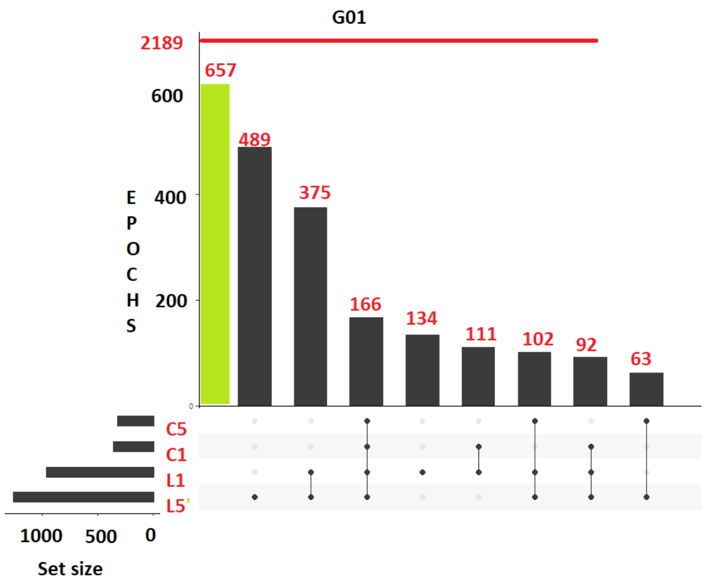



**Figure 3.15: Trend of missing observations for G01 for kinematic data in a medium multipath environment (DOY 325, 2019, track 2)**

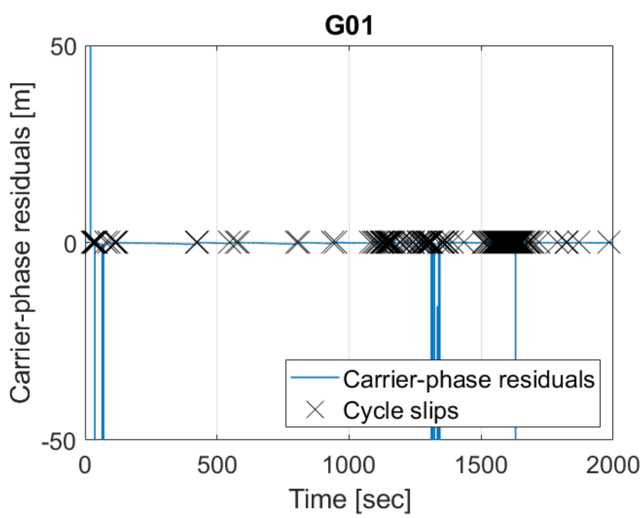



**Figure 3.16: Cycle slips (black crosses) for G01 for kinematic data in a medium multipath environment (DOY 325, 2019, track 2)**

The graph in Figure 3.15 can be interpreted as follows:

- 22% epochs with missing L5 measurements.
- 5% epochs with missing L1 measurements.
- 20% epochs with missing L1 and L5 measurements

- 3% epochs with only C1, L1 measurements logged
- 2% epochs with only C5, L5 measurements logged
- 5% epochs with no measurements.

The satellite G01 was also analysed, for data gaps, for track 1 of the kinematic data collected on DOY 325, 2019. Only 28% (572/2021) of the data had all observations present. Figure 3.17 shows the maximum missing percentage for L5 measurement at 60% followed by L1 measurement at 45%.

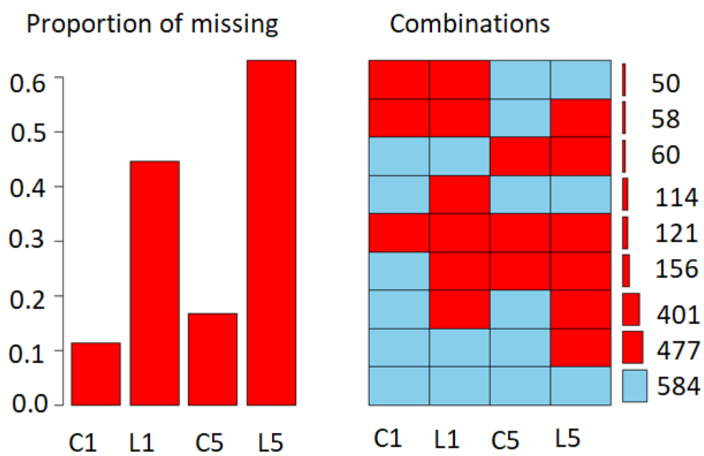



**Figure 3.17: Percentage of missing observables for G01 for kinematic data in a medium multipath environment (DOY 325, 2019, track 1)**

To understand the duration of these data gaps and the frequency of their occurrence, measurements from G01 for track 1 were analysed in Figure 3.18. For all four measurements show the largest frequency count for 1 epoch of missing measurements, indicating that predicting such measurements will be useful in having a continuous solution. Also, as expected the lower duration of data gaps had a higher frequency of occurrence as compared to the longer data gaps. The carrier-phase measurements, both L1 and L5 had an extremely large duration of data gaps as well.

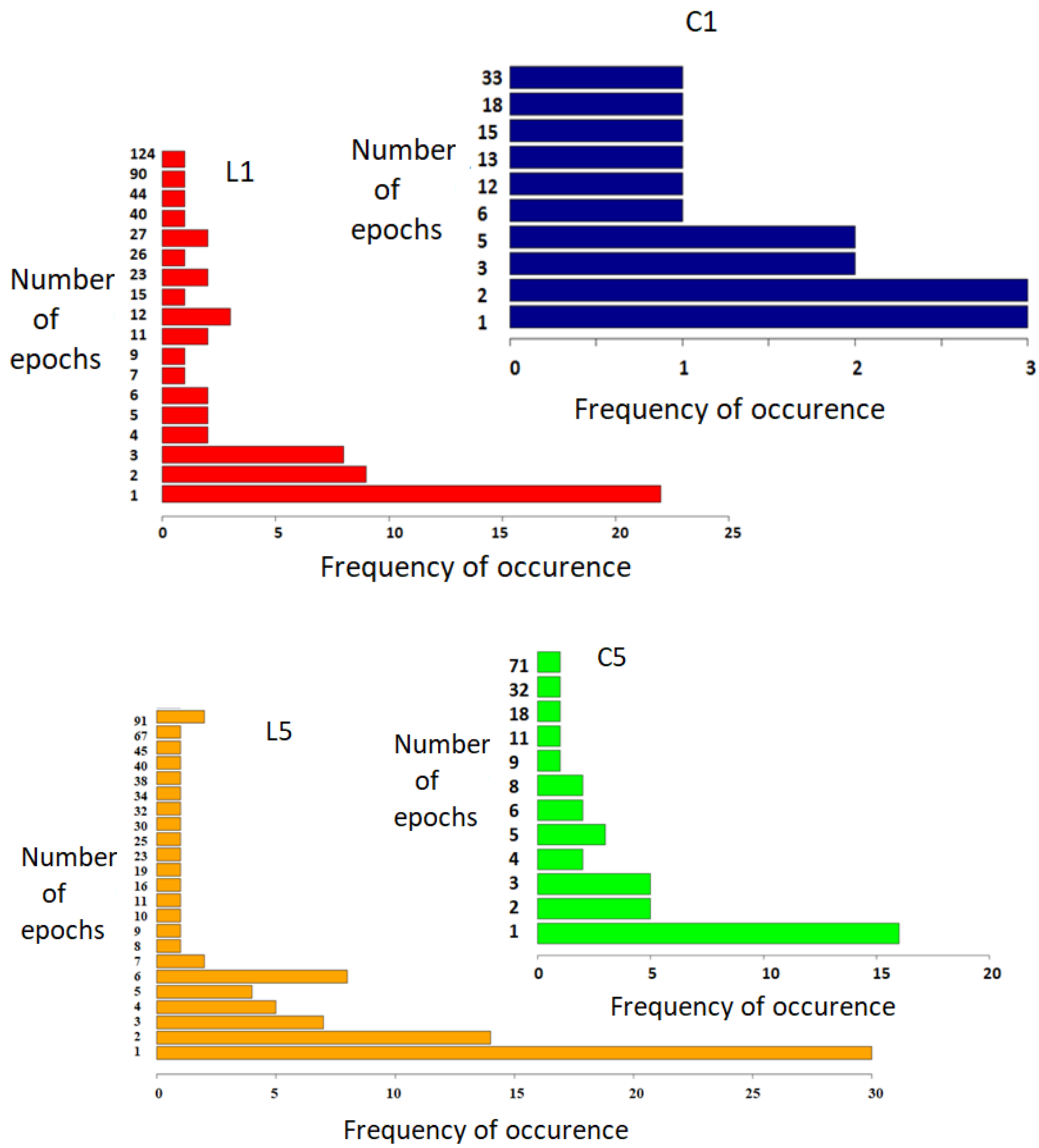



**Figure 3.18: Duration of data gap and its frequency of occurrence for G01 for kinematic data collected in the medium multipath environment (DOY 325, 2019)**

In the Galileo satellites, for E02 there is a similar trend of a higher number of missing carrier-phase measurements and this pattern is common for all the observed GPS and Galileo satellites tracked. A surprising trend for this satellite is the fact that there are roughly 60 epochs where measurements from the second frequency are available, while there are no code and carrier-phase measurements from the first frequency as shown in Figure 3.19.

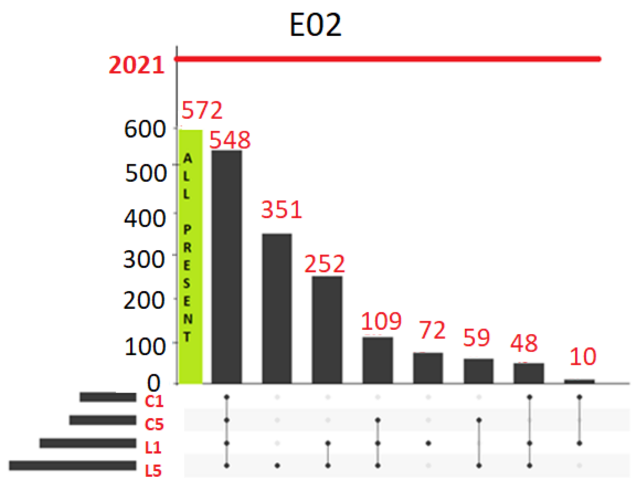



**Figure 3.19: Trend of missing measurements for E02 for kinematic data in a medium multipath environment (DOY 325, 2019, track 1)**

A second dataset was collected on DOY 343 2019 by keeping the Xiaomi phone close to the windscreen of the car. For the GPS and Galileo constellations, a mean of 56% raw measurements from all the different satellites tracked was found to be usable, while 44% of the data had to be rejected due to missing L1 or L5 observable despite the code measurements being present. Due to rejections in the PPP pre-processing, GNSS measurements from only 3 satellites could be utilised during certain periods.

Moving on to a high multipath environment, a detailed analysis of the data gaps helps understand the pattern and relation among the trend of observed missing measurements for data collected on DOY 355, 2019. Figure 3.20 shows the trend of missing code and carrier-phase measurements for

G10 for data collected using the Xiaomi MI 8 phone, which was kept in the hand while walking in downtown Toronto for 28 minutes. Corresponding to these data gaps, Figure 3.21 highlights the cycle slips using black crosses. The number of epochs with missing L1 and L5 carrier-phase observables was higher than the percentage of all measurement present. Only 28% of epochs had all four measurements present while 68% of epochs had missing carrier-phase measurements for either L1, L5 or both while the corresponding code measurements were present.

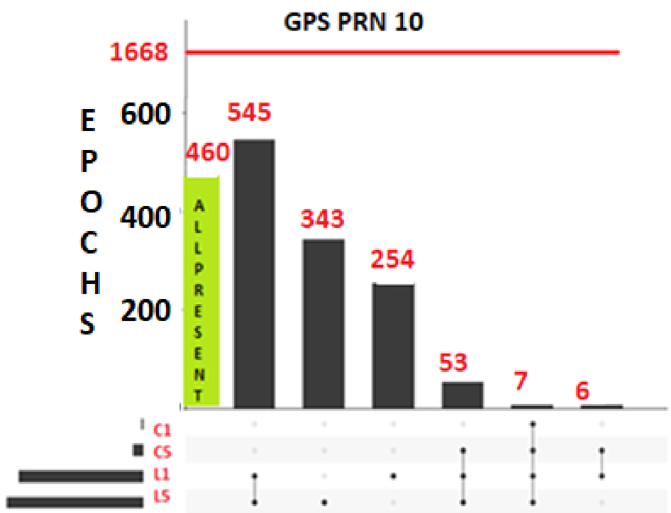



**Figure 3.20: Trend of missing observables for G10 for kinematic data collected in a high multipath environment (DOY 355, 2019)**

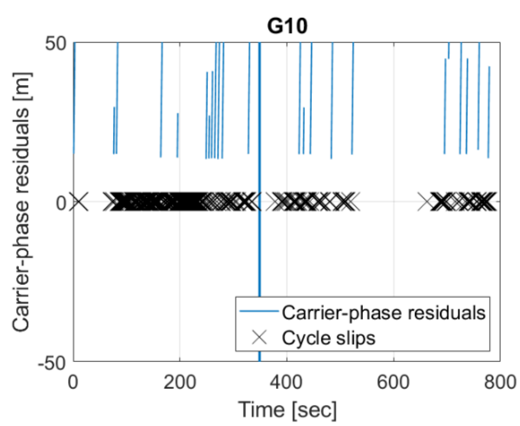



**Figure 3.21: Cycle slips (black crosses) for G10 for kinematic data collected in a high multipath environment (DOY 355, 2019)**

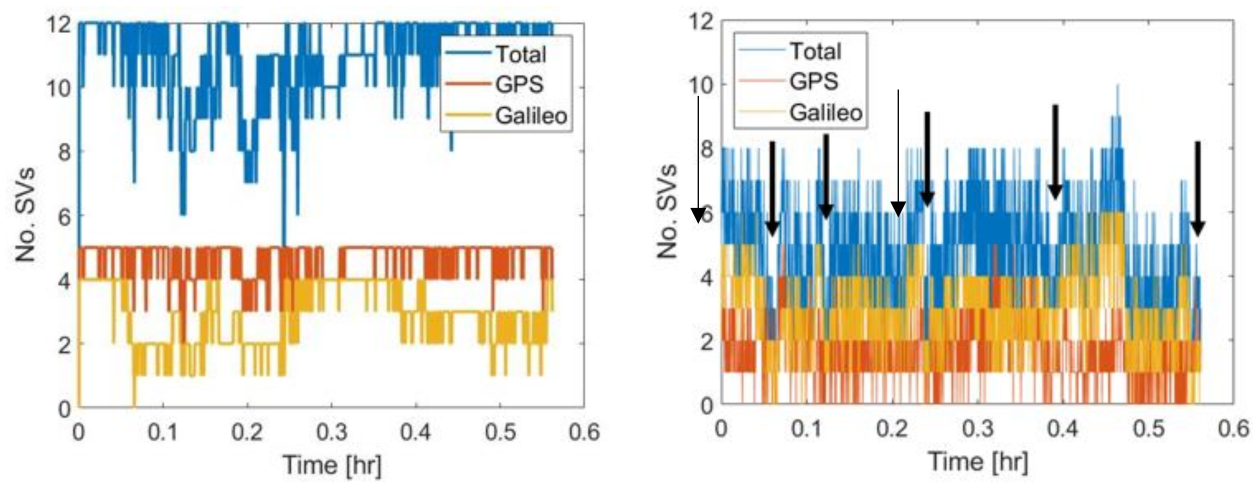
The analysis of data gaps and their frequency along with the frequent cycle slips confirms the nature of missing observables from smartphones, highlighting the frequent absence of carrier-phase measurements due to periodic cycle slips and phase-loss-of lock. The situation worsens as the multipath increases and in real-life situations with the phone in hand in urban environments. It is, therefore, essential to have a real-time measurement synthesis mechanism that fills in data gaps so there should be enough satellites to have a continuous solution.

The analysis highlights the deteriorating condition of the quality of the measurements and the increase in the number of data gaps with an increase in multipath in realistic environments. Summarising the analysis, it was seen that approximately 80% of measurements were present in low-multipath open-sky static environments which fell to as low as 25% in kinematic urban environments with considerable multipath, proving that prediction is necessary for a continuous solution in realistic environments. Table 3.9 summarises the percentage of missing measurements in different multipath scenarios. Observations from three satellites G01, G10 and E02, were analysed and averaged for all three scenarios.

**Table 3.9: Percentage of missing measurements for the GNSS measurements from smartphones data in different multipath environments**

| Scenario | All present | All missing | L1 / L5 missing | C1 / C5 missing | Only single frequency present |
|---|---|---|---|---|---|
| **Static Low multipath** | 79 | 0.2 | 20 | 18 | 8 |
| **Kinematic Medium multipath** | 29 - 55 | 6 - 27 | 55 | 22 | 11 |
| **Kinematic high multipath** | 25 | 12 | 73 | 22 | 21 |

Figure 3.22 compares the count of actual satellites available versus utilizable satellites after rejection for the driving dataset in a medium multipath environment (DOY 325, 2019, track 1), with the no solution portion highlighted with black arrows. On an average 11 satellites were tracked, but only 5 could be processed after rejections due to missing measurements from various satellites.




(a) Satellites tracked (b) Satellites processed

**Figure 3.22: Satellites tracked versus satellites processed for the kinematic dataset (DOY 325, 2019 -track 1)**

### 3.2.6 Correlating Data Gaps with C/$N_0$

Understanding the periodicity and the occurrence of the data gaps is crucial to devising a strategy to predict them. The underlying effect of the low and irregular C/$N_0$ is visible in the occurrence of these data gaps. To investigate this, different satellites were observed for 1000 epochs for missing L1 and L5 carrier-phase measurements, since they dominated the count of the highest frequency of missing observables. The kinematic data collected in a medium multipath environment were used to analyse the relationship between C/$N_0$ and data gaps. The red circles in Figure 3.23 indicate the missing measurements and they identify with a sudden drop in the $C/N_0$. The sudden drop in $C/N_0$ implies that a signal having suffered from blockage or reflection has been acquired. The reflected signal has a lower power while the receiver having suffered from a loss-of-lock leads to missing carrier-phase measurements. These occurrences of data gaps increase in urban canyons, forested areas or high multipath environments as shown by multiple cycle slips.

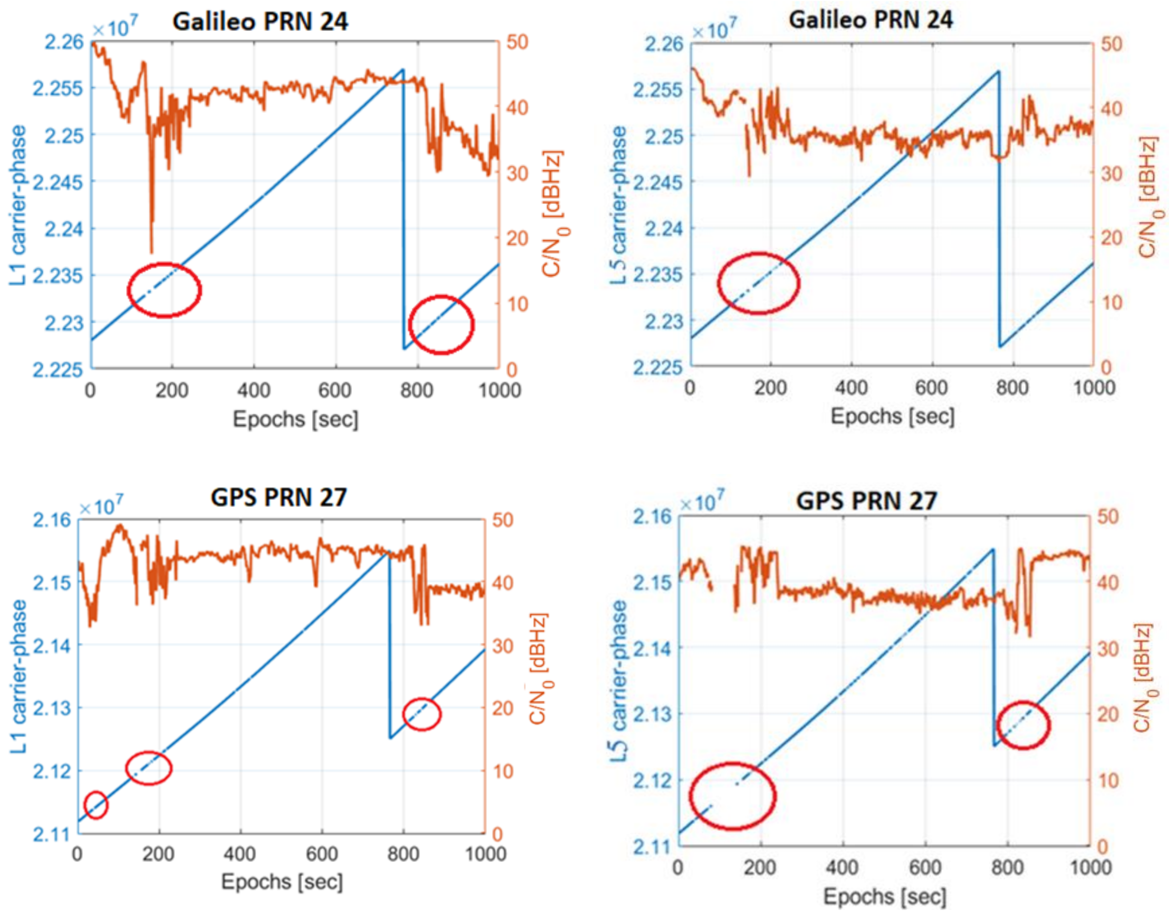



**Figure 3.23: Relationship between missing carrier-phase measurements (blue) and $C/N_0$ (red) in a medium multipath environment (DOY 325, 2019)**

Figure 3.24 illustrates the relationship in a high multipath environment for GPS PRN 27 and Galileo PRN 8. Here the number of missing measurements for two different satellites is observed.

The number of missing measurements is extremely high because of high multipath and multiple signal blockages. Once again, the red circles identify regions of very low and irregular C/N$_0$ values which correspond to the missing carrier-phase measurements. This analysis identifies a relationship between C/N$_0$ and multipath and C/N$_0$ and data gaps. Hence, a C/N$_0$-based stochastic model to weight measurements is extremely crucial. Moreover, even in the prediction of measurements, it will be taken as an error estimate since the quality of the measurements is dependent on it.

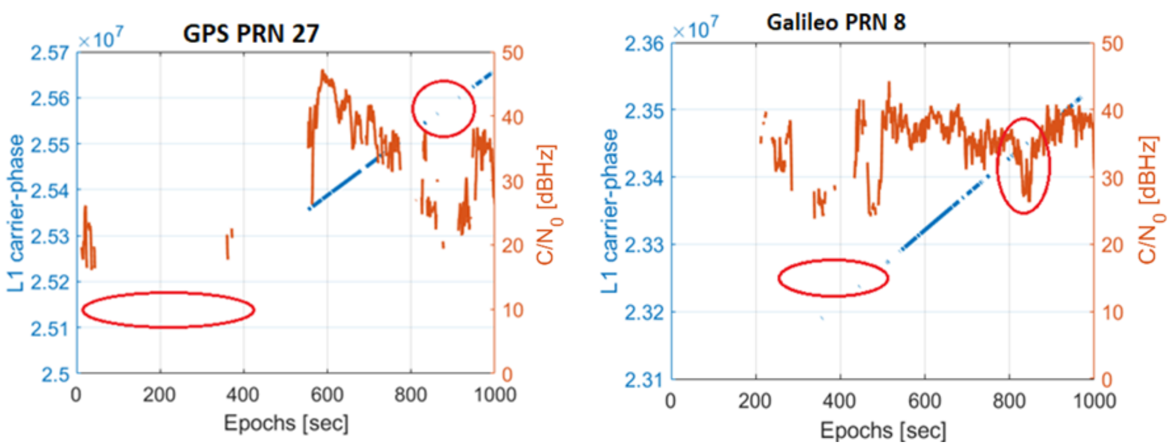



**Figure 3.24: Relationship between the missing carrier-phase measurements (blue) and C/N$_0$ (red) in a high multipath environment (DOY 355, 2019)**

## 3.3 Quality Control Customisation

The existing YorkU PPP software had initially been designed for geodetic receivers with low measurement noise and high C/N$_0$. With low-cost receivers, the focus on processing measurements from them required the software to be customized to handle measurements with higher multipath and noise and multiple cycle slips. Smartphone measurements, as described in the previous sections suffer from multiple issues such as high measurement noise and multipath, low and irregular $C/N_0$, frequent cycle slips, and multiple data gaps. Moreover, since the measurements are from the L1 and L5 frequency band, the software had to be customized to process measurements from the L5 band, which was not previously present. Also, certain pre-and post-processing checks had to be applied to ensure the quality and integrity of the positioning solution. They included elevation and $C/N_0$-based cut-offs and pre-and post-fit residual checks.

### 3.3.1 Pre-Processing Quality Control Checks

The pre-processing of measurements is conducted individually for each satellite. Firstly, after assigning the observables, they are tested for missing measurements. Though the prediction of measurements is being carried out for smartphone measurements and will be discussed in Chapter 4, there are some extremely large data gaps over which prediction is not possible. Moreover, there are certain satellites and epochs with measurements from only one frequency present. Such satellites are rejected in dual-frequency processing.

Secondly, the measurements are tested to detect any large outliers and subsequently rejected, in what is called pre-fit residual checks. The Melbourne-Wubenna is formed to detect any cycle-slips. If the $C/N_0$ value is less than the empirically set threshold value, the satellite is rejected for that epoch. The elevation-angle threshold is also checked. For measurements from smartphones, a $C/N_0$ cut-off of 20 dBHz was selected for all multipath environments. Satellites having $C/N_0$ values less than 20 dBHz had about 90-95% of carrier-phase measurements missing. Hence, such satellites could not be used. Moreover, even if they rarely had a measurement present, the measurement noise and multipath was extremely high. For the SwiftNav Piksi and other low-cost receivers, the threshold was maintained at 15 dBHz since there was no such issue of missing measurements. Figure 3.25 shows the L1 code and carrier-phase measurements for GPS PRN 22 having a $C/N_0$ of less than 20 dBHz (19 dBHz). There were no carrier-phase measurements present though 80 % of the code measurements were present.

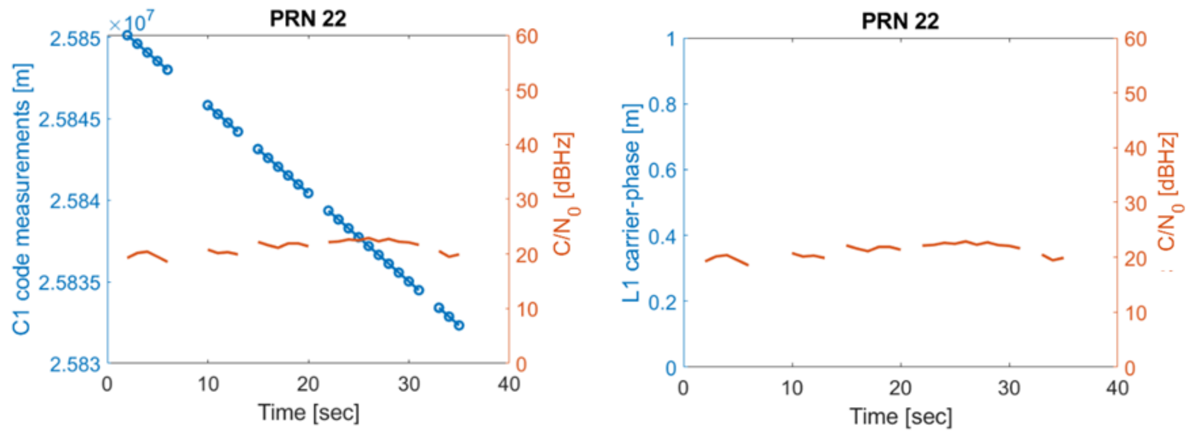



**Figure 3.25: Relationship between the missing carrier-phase measurements and $C/N_0$**

Figure 3.26 shows that there is no relation between measurement noise and missing measurements with the elevation angle for smartphones. Two satellites, one with low elevation (18°) and the other with high elevation (70°) were analysed for missing measurements and measurement noise. The gaps in the plots identify missing measurements. G32 having a higher elevation angle had 12% higher measurement noise and 15% more data gaps as compared to E26 with a lower elevation angle. The elevation angle cut-off was maintained at 10° for all receivers.

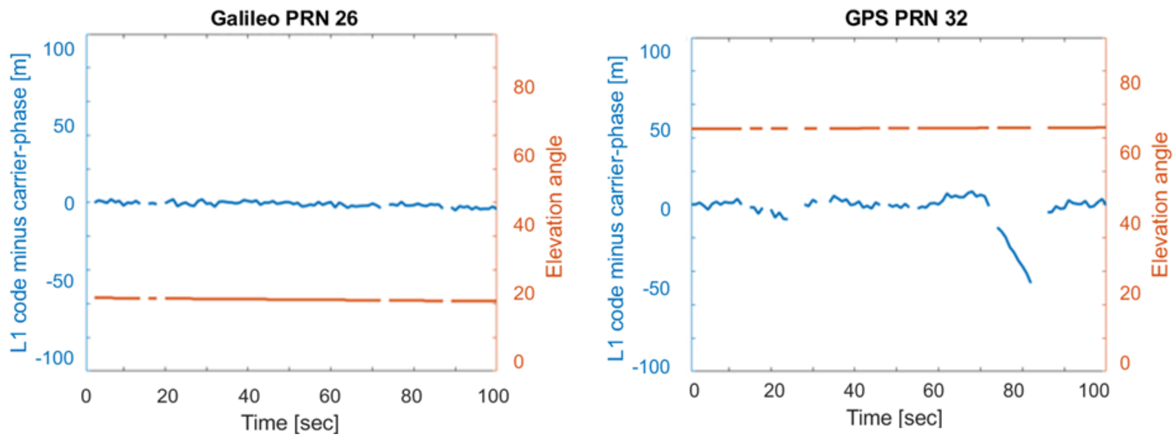



**Figure 3.26: Relationship between missing measurements and satellite elevation angle**

### 3.3.2 Post-Processing Quality Control Checks

Once all the measurements have passed the initial quality control (QC) checks, the solution is computed with the required least squares filter. Residuals are computed and statistically analyzed for outliers. If any residual does not pass the threshold limit, the corresponding observation is removed, and the solution is again computed. The re-computation of the solution happens till a threshold of the maximum number of iterations has increased and the solution still has not converged to a definite threshold (Gill 2018). These post-fit residual checks are computed based on an integral multiple of the standard deviation of the code and carrier-phase measurements taken to weigh the measurements. Since, for the smartphone processing the multipath noise plays a decisive factor in deciding the measurement weighting, it forms the basis of deciding the post-fit residual check, decided after an empirical analysis. For this research, it was empirically set at 10.47 scaled by the *a priori* measurement weighting factor. This value was empirically settled upon after testing several datasets collected in a low and medium multipath environment.

# Chapter 4 Measurement Prediction

In a medium to high multipath environment, the low-cost and correspondingly, low-quality GNSS antennas in smartphones cause the carrier-phase measurements to suffer from frequent cycle-slips due to phase loss-of lock-leading to data gaps as discussed in Chapter 3. To ensure the increased availability of the positioning solution in realistic environments, a measurement prediction strategy has been devised for smartphones. To prevent the complete rejection of the satellite in such cases, measurement prediction can be carried out for a few epochs in low to medium multipath environments. However, in urban landscapes or heavily forested areas, the data gaps can exist for several tens of epochs and occur frequently. In such cases, it is best to reject such satellites and most of them do get rejected, due to low $C/N_0$ values, as described in Chapter 3.

This chapter contains a background on the different measurement prediction techniques, provided by various researchers, which focus on carrier-phase measurements. These techniques are tested for the prediction error for several datasets. Finally, a final prediction technique is adopted, based on extrapolation. Different kinematic datasets are tested for the efficiency and quality of the predicted solution. Though prediction for both the pseudorange and carrier-phase measurements has been implemented in the code, the primary focus has been on carrier-phase measurements since they are more vulnerable to signal blockages and missing measurements (Lachapelle et al. 1992; Curran 2015; Marçal and Nunes 2016).

## 4.1 Carrier-phase measurement prediction

Smartphones with cheap antennas and phase lock loops in receivers have cycle slips affecting carrier-phase observations, even in open sky conditions. Various research studies have proposed cycle slip detection and repair methods to avoid re-fixing ambiguities with hardly any reference to missing measurements and overcoming data gaps. Common methods of cycle slip detection include polynomial fitting (Wang et al. 2016), high-order differencing (Hu and Fang 2009), pseudo-range and phase combination (Bisnath 2000), Kalman filtering (Liu et al. 2018) and ionospheric residual (Fan et al. 2015; Chen and Huang 2016). These methods are effective for a small number of cycle-slips; however, they do not account for the data gaps. After signal recovery, the receiver takes approximately 3 seconds to re-acquire and track the signal and up to 10 to 12 seconds to re-enter the frame synchronization to ensure the integer characteristics of carrier phase observations. Several ways have been suggested to correct these missing measurements. The first is the prediction of carrier-phase measurements by estimating the carrier Doppler shift between the satellite and the receiver between two epochs. The second is an L1 backup navigation model in the event of intermittent loss of the L2 signal.

### 4.1.1 Carrier-Doppler Estimation Technique

If the time of blockage is short, predicted measurements may be helpful to increase the availability of the carrier phase measurements. A few researchers have developed carrier phase prediction methods for PPP in difficult environments. The predicted carrier phase can be obtained by integrating the predicted Doppler of the blocked satellite, which is given by the equation (Li et al. 2019)

$$\phi_{k+1} = \phi_k + \int_{t_k}^{t_{k+1}} f_{predict} dt + \delta\phi \tag{4.1}$$

where

$\varphi_{k+1}$, $\varphi_k$ are the carrier phase at epoch $k + 1$ and k, respectively.

$f_{predict}$ is the predicted Doppler of the channel.

$\Delta\varphi$ is the carrier phase estimation error.

The measured Doppler of the receiver is mainly composed by four parts: (1) Doppler caused by relative motion between the satellite and the receiver; (2) clock drifts of the satellite and the receiver; (3) change rate of the propagation path delay; and (4) thermal noise (Li et al. 2019). The Doppler frequency caused by the clock drift of the satellite, the change rate of the propagation path delay and thermal noise is relatively small over a short duration. Hence, the Doppler caused by the relative motion of the receiver and satellite along with the Doppler caused by the clock drift shall be taken into consideration to predict the carrier Doppler, which will then be integrated to predict the carrier-phase measurement. The predicted Doppler can be expressed as (Li et al. 2019):

$$f_{pred} = f_{motion} + f_{clockdrift} = f_{motion} + (f_{PLL} - f_{motion}) \tag{4.2}$$

$$f_{pred} = \frac{(v^i - v)(p^i - p)}{\lambda \left\| p^i - p \right\|} + (f_{PLL} - f_{move}) \tag{4.3}$$

where

$f_{move}$ is the Doppler caused by the relative motion of the satellite and receiver.

$f_{clockkdrift}$ is the Doppler caused by the clock drift of the receiver.

$f_{PLL}$ is the Doppler measurement obtained by the carrier tracking loop.

p, v are the position and velocity of the receiver, respectively.

$p^i$, $v^i$ are the position and velocity of the $i^{th}$ satellite, respectively.

$\lambda$ is the carrier wavelength.

To predict the Doppler, it is necessary to estimate $f_{move}$ and $f_{clockdrift}$ for each channel. The positioning module of the receiver can output the position and velocity of the receiver and the satellites. Since the Doppler caused by receiver clock drifts are the same in all satellite tracking channels, the clock drifts obtained from normal tracking channels can be used to assist the open-loop tracking channels. With no IMU measurements, the estimation of the velocity of the receiver is crucial to the effectiveness of this technique (Silva 2013).

### 4.1.2 L1 Backup Navigation Model

Research on synthesizing carrier-phase measurements has also been carried out by Yang et al. (2003) for L1 backup navigation in the event of intermittent loss of the L2 signal. A three-state Kalman filter is used for each GPS satellite. When both L1 and L2 signals are available for the satellite, L1 and L2 observables are used to estimate ionospheric refraction delay, delay rate, and a combination of integer ambiguities on L1 and L2 carrier phase measurements. The technique described here could be applied to synthesize L1, L2 or L5 measurements if any one of the three frequencies is retained. The missing measurements at one of the frequencies can be synthesized using the measurements from the retained frequency that was not lost. The ionospheric refraction effects are modelled, which are corrected by the measurements taken when both frequencies are available. When the measurements of one frequency are missing, the divergence between the retained code and carrier phase measurements can be used to detect slowly changing deviations from the ionospheric refraction model.

The pseudorange and carrier phase model for both L1 and L5 can be written as (Yang et al. 2003):

$$\rho_1 = r + E + \frac{f_2}{f_1} I_a + \eta_1 \quad (4.4)$$

$$\rho_2 = r + E + \frac{f_1}{f_2} I_a + \eta_2 \quad (4.5)$$

$$(\phi_1 + N_1)\lambda_1 = r + E - \frac{f_2}{f_1} I_a + \beta_1 \quad (4.6)$$

$$(\phi_2 + N_2)\lambda_2 = r + E - \frac{f_1}{f_2} I_a + \beta_2 \quad (4.7)$$

with $\rho$ being the pseudorange, r being the true range from satellite to receiver, $E_{cm}$ being the common error that can be removed by differential technique, f being the carrier frequency, $I_a = \frac{40.3}{f1f2}$, TEC being the ionosphere delay, $\eta$ being the pseudorange noise and multipath error, and $\beta$ being the carrier phase noise and multipath error. Differentiating, the common errors are removed and estimates of the ionospheric-corrected pseudorange, carrier-phase and ionospheric models are produced. When both L1 and L2 measurements are available, the dynamic model and the measurement model will contain both the measurements when only L1 measurements are present, the measurement model will change to exclude L2. When L2 measurements return, the L2 carrier-phase measurements will have a cycle slip. The slipped cycles $\delta N_2$ and the new $N_1\lambda_1 - N_2\lambda_2$ can be re-calculated as:

$$\delta N_2 = round(\Delta\phi_1 + \Delta\phi_2) \quad (4.8)$$

$$(N_1\lambda_1 - N_2\lambda_2) = (N_1\lambda_1 - N_2\lambda_2)_{old} + \delta N_2\lambda_2 \quad (4.9)$$

The process is called the floating ambiguity re-initialization after signal outages and it avoids the long process of determining the new value of the floating ambiguity via filtering. The L1 backup navigation technique is good in scenarios where there are no missing measurements from the L1 frequency. However, it is not as effective for smartphones since measurements from even the L1 frequency are missing. Therefore, for this research, both carrier-Doppler and logged Doppler estimates and simple extrapolation will be discussed and analysed for positioning accuracy of the final solution.

## 4.2 Implementation of Prediction Algorithms

In section 4.2.1 to 4.2.3, two methods of carrier-phase measurement prediction will be analysed and tested for carrier-phase measurement prediction error for different satellites.

1) Predicting missing measurements using the estimated and logged Doppler values
2) Predicting missing measurements using the estimated and logged Doppler values
3) Predicting missing measurements using a simple extrapolation technique

### 4.2.1 Predicting Missing Measurements using Estimated Doppler Measurements

Carrier-phase prediction is done using predicted carrier Doppler estimations based on the relative position and velocity of the receiver concerning each satellite. This technique of prediction has been adequately described in Section 4.1. The predicted Doppler for each channel is given by

$$f_{pred} = f_{motion} + f_{clockdrift} \tag{4.10}$$

$$f_{pred} = \frac{(p_i - p)(v_i - v)}{\lambda||p_i - p||} + \frac{c.d}{\lambda} \tag{4.11}$$

In other words, the Doppler shift in Hertz is related to radial velocity by:

$$\mathrm{D}^{\mathrm{sat}} = -\frac{\mathrm{v_r}}{\lambda} \tag{4.12}$$

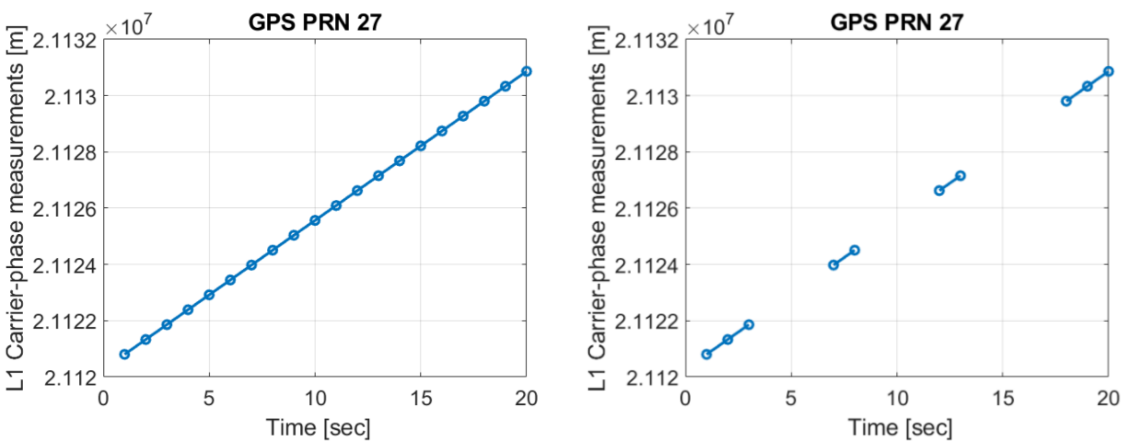
where $\mathrm{D}^{\mathrm{sat:}}$ measured Doppler shift for generic satellite (Hz)

$v_{r:}$ radial component of the difference between the satellite and the antenna velocities (m/s)

λ: carrier wavelength (metres)

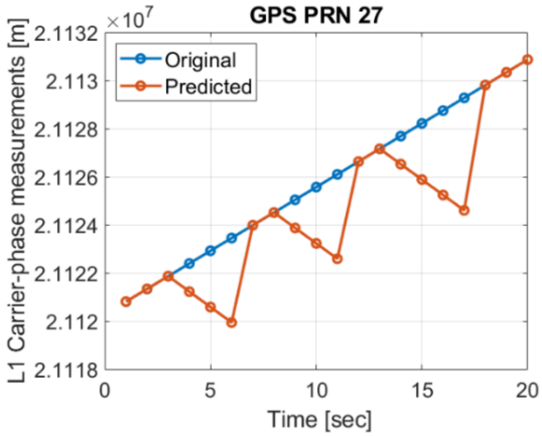
The position and velocity estimates of the receiver are used from the previous epoch to update the current epoch. A record is maintained of the position and velocity of the previous epochs and it is used to compute the carrier Doppler estimations for each epoch, irrespective of whether it is necessary to predict the carrier-phase predictions or not. By doing so, as soon as a missing measurement is detected, the carrier-phase prediction is available to be calculated from the estimated carrier Doppler. To verify the accuracy of this positioning technique, several satellites were analysed for the error in their predicted carrier-phase measurement. Different satellites, some with no outages, and others with short outages, were chosen. Artificial outages were created for durations varying from 2-5 seconds. The carrier-phase measurements were predicted over these outages and then the differences were compared with the original measurements to verify the accuracy of the technique. The most important value to be estimated in this method is the velocity of the receiver phase-centre since the other estimates can be computed. The velocity of the receiver phase-centre is variable, as the receiver may stop, turn or increase in speed based on the user's and receiver's motion.

Take the example of a 20 second period for data collected in kinematic mode on DOY 325, 2019, for G 27. The carrier-phase measurement L1 from 0 to 20 epochs was considered and gaps were synthesised which were then filled by using the estimated Doppler measurements to predict the carrier-phase measurements. Figure 4.1 (a) shows the carrier-phase measurements (in metres) for the first 20 epochs while Figure 4.1 (b) shows the same plot after certain measurements were artificially removed. Figure 4.1 (c) shows the synthesised carrier-phase measurements using the estimated Doppler measurements while Figure 4.1(d) represents the prediction error. The prediction shows an error of up to a few tens of metres far from ideal for PPP processing, though each time a measurement is available the filter is suitably corrected. Similar results were obtained for the L5 measurements and different satellites. Using estimated Doppler measurements computed from relative position and velocity are highly dependent on the accuracy with which the velocity of the car is estimated at different points. The focus of this thesis is GNSS-only processing and hence, the use of measurements from the internal smartphone IMU are not considered. Thus, the prediction estimates may be in error by a few metres. This Doppler prediction technique has produced similar results for Silva (2012) who used a Pro-Flex GNSS receiver along with a uBlox receiver and obtained errors ranging up to several tens of metres in the prediction.

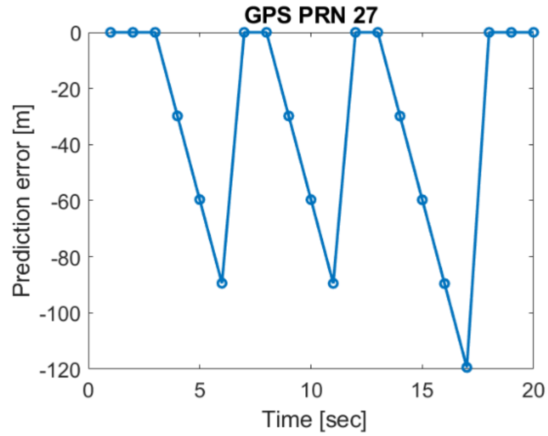



(a) Original carrier-phase measurement

(b) Synthesised gaps




(c) Predicted carrier-phase measurement




(d) Prediction error

**Figure 4.1: L1 carrier-phase prediction using estimated Doppler measurements for G27**

### 4.2.2 Predicting Missing Measurements using Logged Doppler Measurements

The magnitude of error in predicting code and carrier-phase measurements by using the logged Doppler measurements was also tested. Figure 4.2 depicts the relationship between the logged Doppler measurements and the change in L1 and L5 carrier-phase measurements (cycles) over successive epochs. It was observed that sudden spikes or jumps in the logged Doppler measurements associate with missing carrier-phase observations as seen in the plot for G01. Using these irregular doppler measurements to predict missing carrier-phase measurements will lead to a large error in the predicted measurements. Moreover, in kinematic medium to high multipath environments, the logged doppler measurements were itself not continuous as identified by the green circle on the plot for G11. Though the carrier-phase measurements were present at epoch 40, the Doppler measurements were absent. In regions of high multipath and signal blockage, the gaps for the logged Doppler measurements is larger than the gap in the E5a carrier-phase measurements as is seen for E30. For G32 the spikes in the logged Doppler measurements corresponded to missing carrier-phase measurements for 25 seconds even though the satellite had a mean elevation of above 70° and a mean $C/N_0$ of 25 dBHz. The lack of L1 carrier-phase measurements in such a scenario could be attributed to the low-cost antenna which was unable to reacquire tracking of the L1 carrier-phase measurements after a potential loss-of-lock.

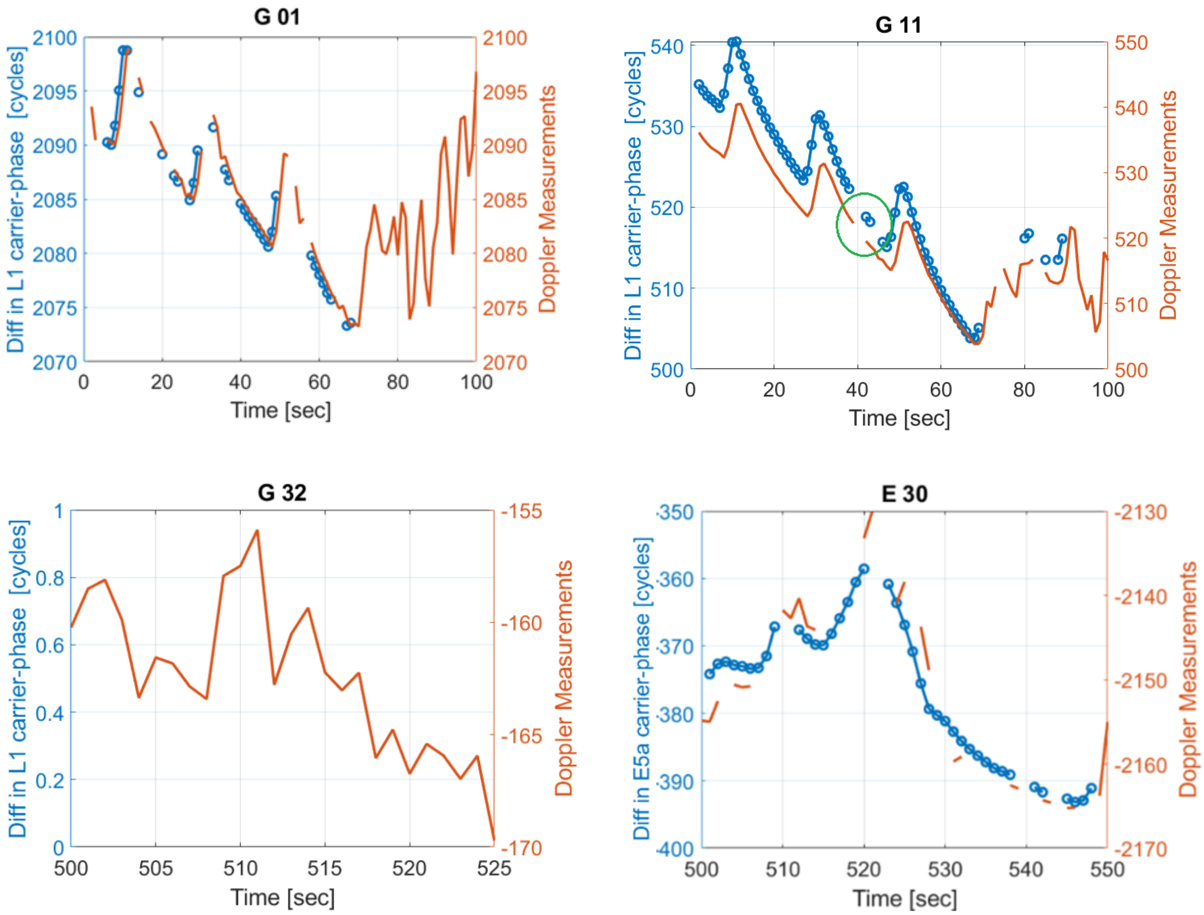



**Figure 4.2: Relation between missing carrier-phase measurements (blue) and logged Doppler measurements (red) in a kinematic scenario (DOY 325, 2019)**

### 4.2.3 Predicting Missing Measurements using Extrapolation Technique

The extrapolation technique is a simple mathematical method of predicting a value based on the trend and change in the past values. It could be linear, cubic, quadratic and so on. For this analysis, the linear and cubic extrapolation techniques were tested for prediction error. The linear regression model is a simple mathematical technique that computes the slope based on the past measurement and uses it to predict the next measurement. If the epochs are taken as X and the carrier-phase / pseudorange measurement is taken as Y, then

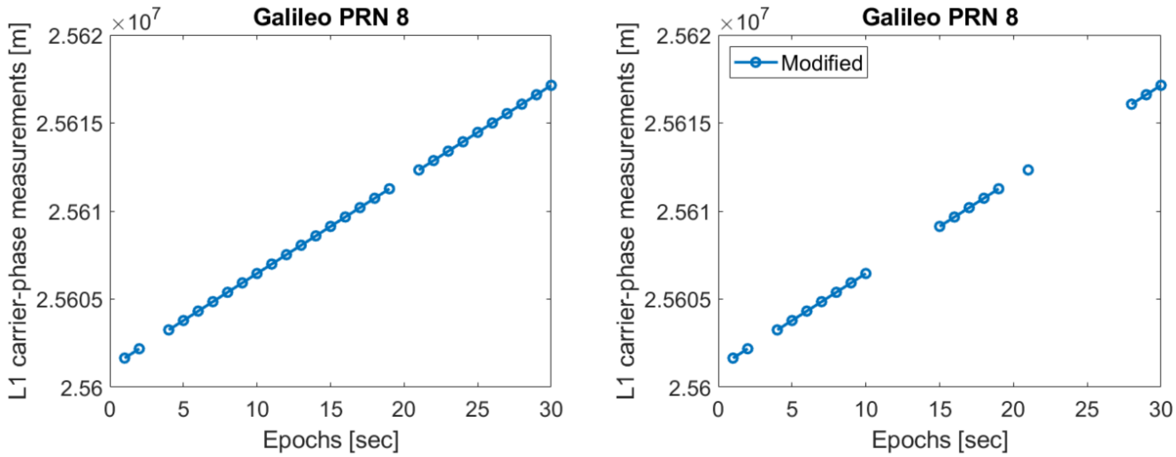
$$m = \frac{n\,\Sigma\,XY - \Sigma X\,\Sigma Y}{n\Sigma X^2 - (\Sigma\,X)^2} \tag{4.13}$$

$$c = \frac{\Sigma Y\ - m\Sigma \mathrm{X}}{n} \tag{4.14}$$

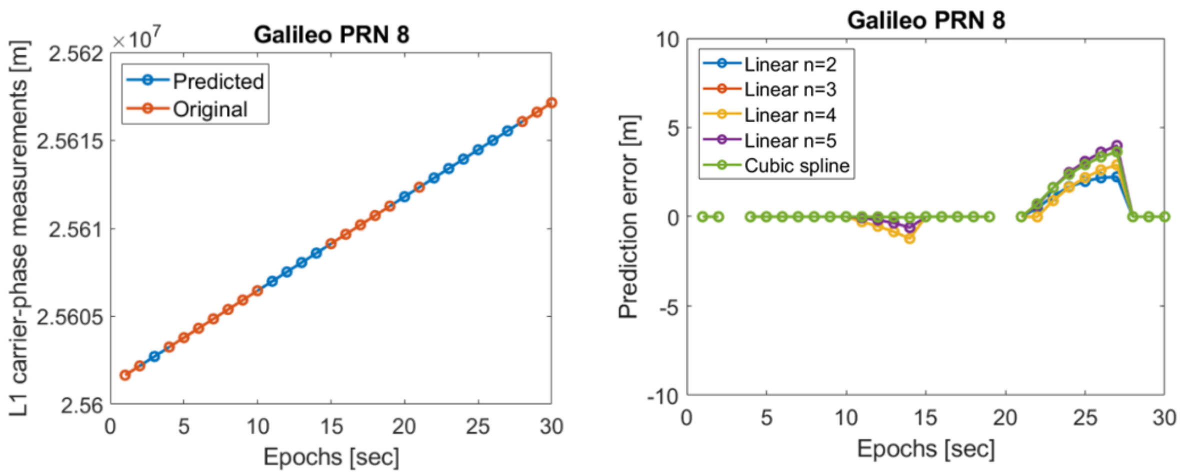
where n is the number of past measurements considered. The extrapolated measurements are taken into consideration if the gap is greater than an epoch. X is the index to count the number of epochs; Y is the value of the observable being predicted. The next measurement at x can be predicted as:

$$c\ +\ m * x \tag{4.15}$$

The carrier-phase measurements for Galileo PRN 8 in a kinematic medium multipath scenario are analysed in Figure 4.3(a). Similar outages ranging from 2-5 sec were simulated and the extrapolation technique is tested as shown in Figure 4.3(b). There were pre-existing data gaps during epoch 3 and 20. Artificial gaps are synthesised and filled with predicted values using the different techniques to test for the strategy with the least magnitude of prediction error. Different values of 'n' are tested to determine the size of the window or number of past epochs that should be considered to predict the next measurement. Since the range of the measurements lies in the range of $10^7$, it is difficult for the naked eye to observe any differences in the prediction error for different values of n. Overall, the linear extrapolation technique using the past two measurements yielded the least possible prediction error. Using a linear extrapolation technique with 2 past measurements gave a minimal rms prediction error of 1.4 m which increased to 1.6 m for 3 and 4 measurements, 2.2 m with 5 measurements and 2.0 m with cubic spline extrapolation.



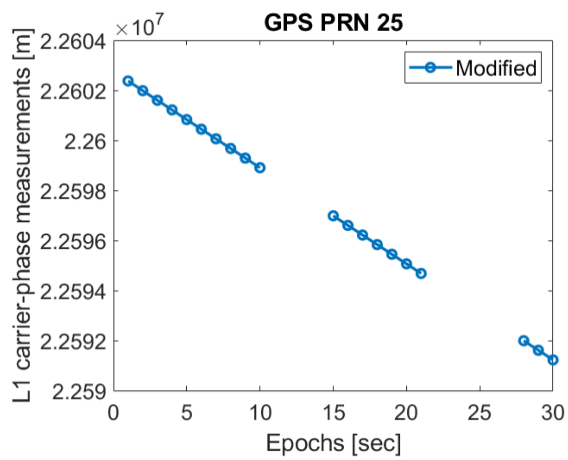

(a) Original carrier-phase measurements (b) Synthesised gaps



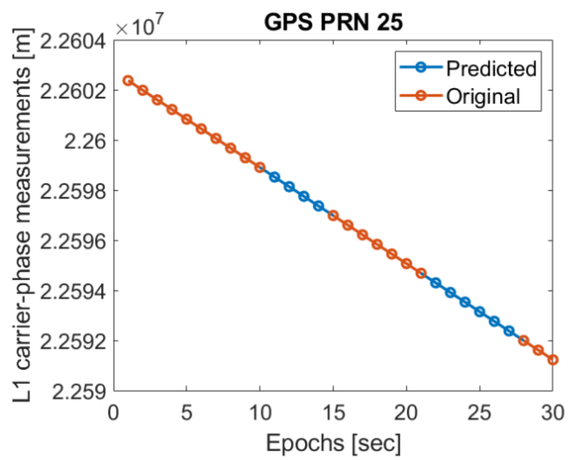

(a) Predicted L1 measurements (b) Prediction error

**Figure 4.3: Predicted L1 measurements and prediction error using different extrapolation techniques for Galileo PRN 8 (DOY 325, 2019)**

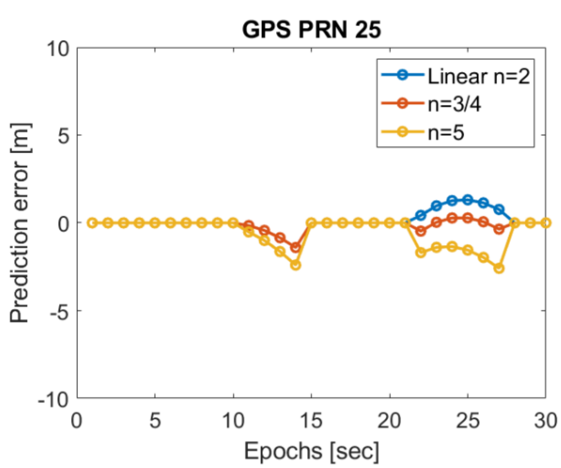
Several other satellites were tested to check for the prediction error and the best possible extrapolation technique to be used. The carrier-phase measurements for satellites GPS PRN 25 and Galileo PRN 26 are depicted in Figure 4.4 and 4.5. Using a linear extrapolator with 3 past measurements, a minimal rms error of 57 cm was seen in the prediction which grew to 94 cm with 2 measurements and to 1.7 m using 5 measurements, for GPS PRN 25. The cubic spline extrapolation had an rms error of 3.1 m. The Galileo PRN 26 L5 carrier-phase measurements already had a data gap present at epoch 17. The linear extrapolator with the past 2 measurements gave an rms prediction error of 15.9 cm over the 4 epochs (epochs 21-24, inclusive) of prediction while the prediction errors using 3 and 4 measurements were 2.8 and 2.6 m, respectively. The cubic spline extrapolation yielded an error of 3.5 m for it.



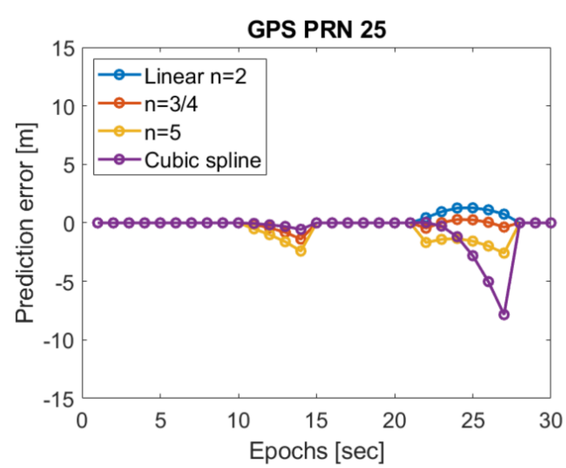

(a) aModified carrier-phase measurement




(b) Predicted Missing Measurements

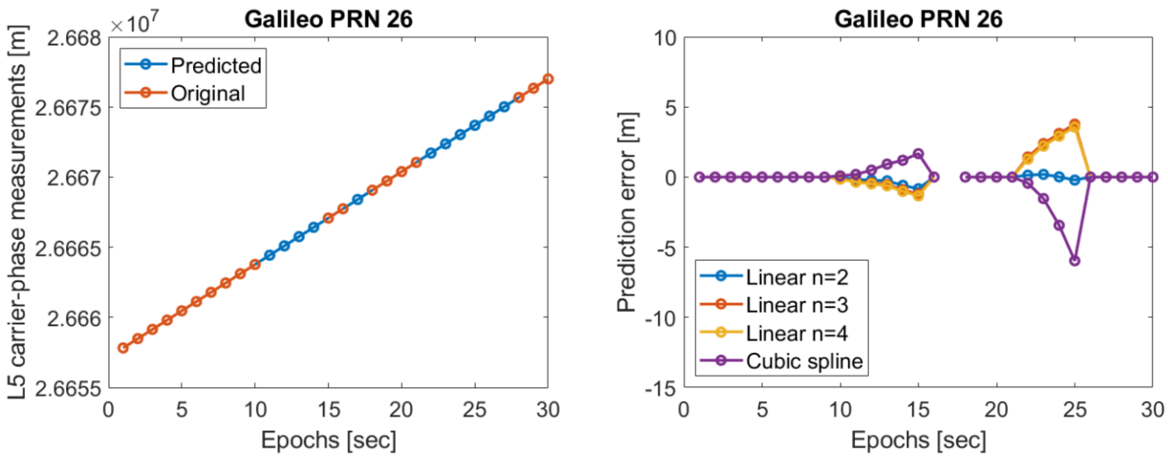



(c) Prediction error




(d) Overall Prediction error

**Figure 4.4: Predicted L1 measurements and prediction error using different extrapolation techniques for GPS PRN 25 (DOY 325, 2019)**




(a) Predicted carrier-phase measurement (b) Prediction Error

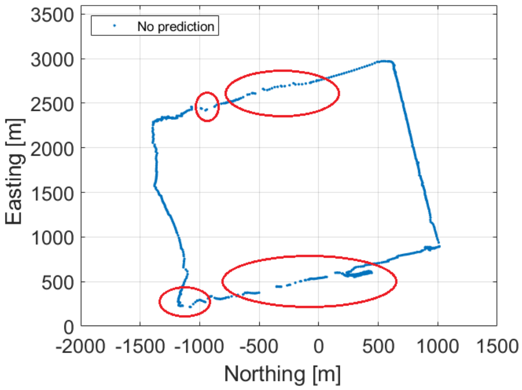
**Figure 4.5: Predicted L5 measurements and prediction error using different extrapolation techniques for Galileo PRN 26 (DOY 325, 2019)**

Based on the analysis, it was concluded that the best possible prediction occurs with linear extrapolation using measurements from the previous two epochs. The linear extrapolation technique using the past 2 measurements resulted in the least amount of prediction error as compared to the other extrapolation techniques such as the cubic spline technique since the variability in the magnitude of measurements changes drastically over successive epochs for smartphone GNSS measurements collected in kinematic scenarios. It is best to use a minimum number of measurements from the past epochs to predict the next measurement.

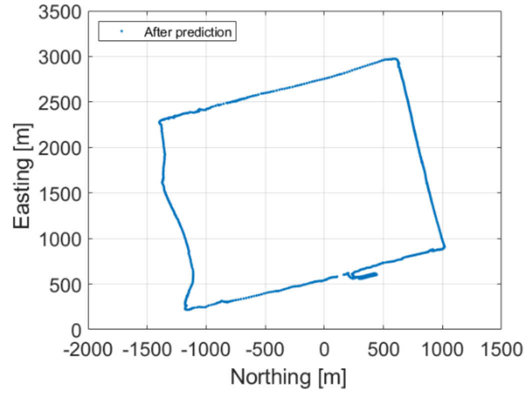
The carrier-Doppler technique was plagued with the problem of unforeseen variability in the dynamics of the receiver movement which could not be accurately modelled in a local Kalman filter. The logger Doppler measurements suffered from data gaps themselves. The implementation of the linear extrapolation technique is a preliminary step in the direction of solving the data gap problem. It gave cm-dm level prediction error for prediction of up to 5 epochs in a kinematic medium multipath environment. Future implementation will work on implementing higher-order extrapolation techniques and filling in data gaps for at least up to 10 seconds in medium to high multipath environments.

## 4.3 Assessment of the Performance of Prediction on Kinematic Data

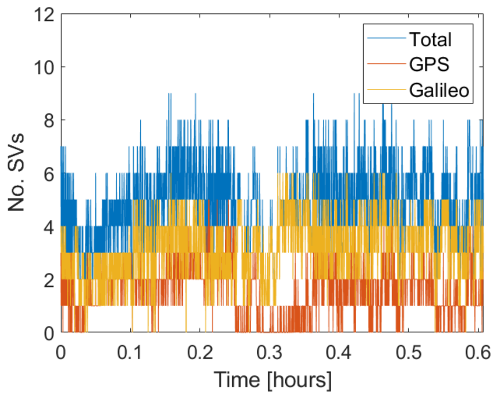
A kinematic dataset collected in a medium multipath environment was assessed for positioning accuracy and availability of the solution before and after the implementation of the prediction technique. The smartphone was placed on the dashboard of the car while the low-cost SwiftNav antenna connected to the SwiftNav Piksi receiver was on the rooftop of the car. The base station comprised a Topcon-NET G3A antenna placed on Petrie rooftop. Figure 4.6 shows the horizontal track for the kinematic driving dataset before and after prediction. This dataset was analysed to compare the availability, accuracy and convergence of the positioning solution before and after deploying the prediction technique and using the RTK processing technique. The red boxes in Figure 4.6(a) indicate the periods of no solution without any pre-processing and conditioning. Figure 4.6(b) is the same plot after the prediction. The average number of processed satellites increased by 42% to 7 per epoch as shown in Figure 4.6(d). Finally, the solution was plotted against the SwiftNav Piksi RTK reference plot (Figure 4.6(f)).



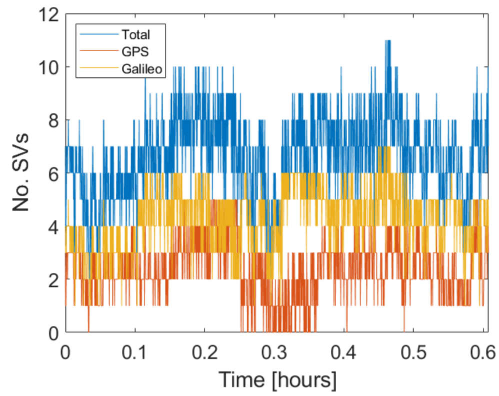

(a) Before Prediction



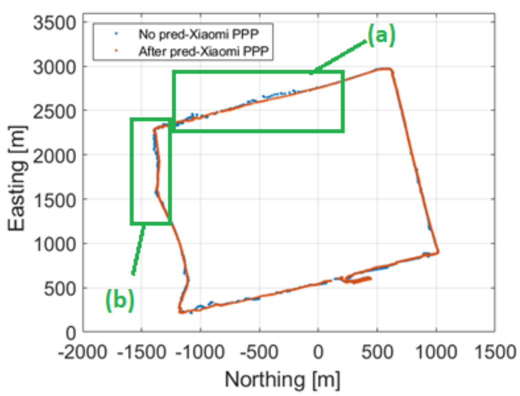

(b) After prediction

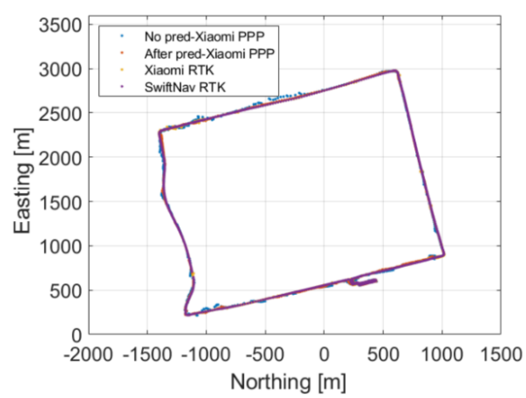



(c) Satellite count before Prediction




(d) Satellite count after Prediction

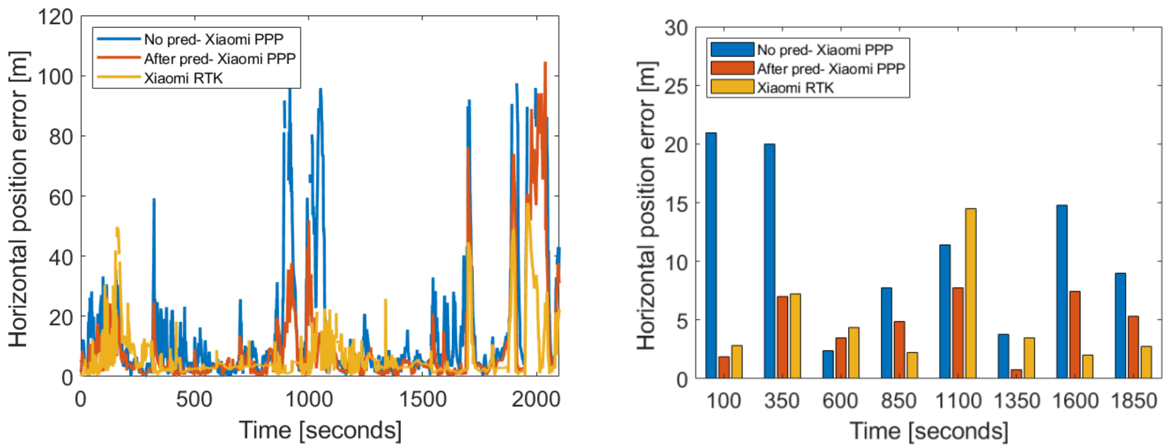



(e) Pre and post-prediction comparison (f) Solution comparison with SwiftNav RTK solution

**Figure 4.6: Horizontal track and number of satellites processed before and after the prediction (DOY 325, 2019, track 2)**

The horizontal positioning error for the two scenarios was compared and the standard deviation of the horizontal positioning error was computed with the SwiftNav Piksi solution as a reference. Figure 4.7 (a) and (b) indicate the horizontal positioning error over the entire solution and certain epochs, respectively, after prediction which shows a decrease of over 5 m in the std of positioning error. An important point to be noted is that the phase centres of the SwiftNav and the phones were not aligned. The SwiftNav antenna was on the roof and had better signal availability while the smartphone was inside, on the dashboard of the car (to mimic the real usage of a smartphone while navigation), which implies that there should be at least a metre offset in the solution and that the number of satellites tracked by the smartphone antenna is limited by its position within the car. Moreover, the data collected by the smartphone is limited to 1 Hz, while the SwiftNav Piksi can log data at 5 Hz. All these factors must be taken into consideration while comparing the positioning solution. Hence, the overall objective was to compare the positioning solution accuracy before and after prediction and against the solution offered by positioning techniques such as RTK. A detailed comparison with the RTK technique will be done in Chapter 6. The predicted solution gives a solution with a 99.8% availability, but also increases the accuracy of the positioning solution by several metres, especially during satellite blockages at turns or intersections as seen in Figure 4.6 (e). Comparing with the RTK solution, it was observed that only 2114 out of 2189 epochs could be processed (96%) and the RTK solution had a horizontal error with a standard deviation of 8.2 m. Hence, a prediction technique could be beneficial for RTK processing of smartphone GNSS measurements, as well.




(a) Horizontal positioning error (b)Epoch-specific horizontal positioning error

**Figure 4.7: Horizontal positioning error before and after prediction vs the RTK solution for the smartphone (DOY 325, 2019, track 2)**

**Table 4.1: Standard deviations of the horizontal positioning error and availability of the solution before and after the prediction for the PPP technique compared with the RTK solution for the Xiaomi MI 8 (DOY 325, 2019, track 2)**

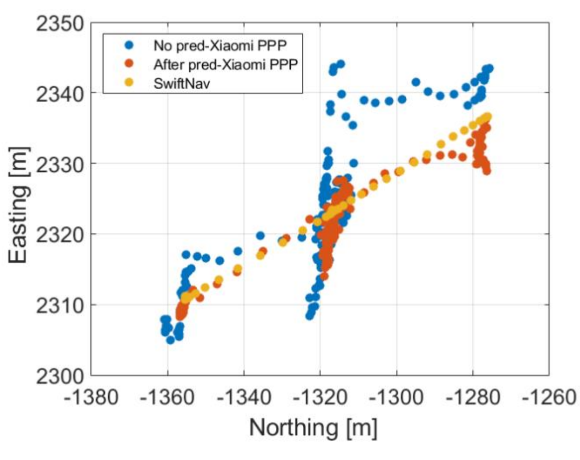
| Scenario | Horizontal standard deviation (m) | Percentage of solution |
|---|---|---|
| **Before prediction - PPP solution** | 14.7 | 2149/2189 = 98.2% |
| **After prediction - PPP solution** | 9.6 | 2184/2189 = 99.8% |
| **RTK solution** | 8.2 | 2114/2189 = 95.6% |

Since the smartphone antenna was inside the car, it was vulnerable to signal blockages by external traffic and buildings, especially when the car turned. The linearly polarised antenna suffers from phase-loss-of-lock and code multipath is enhanced by reflections due to the windshield and the car interiors. Two portions of the track between (a) epochs 1200 and 1400; and (b) 1400 and 1600 were selected to analyse the effectiveness of the prediction strategy in areas most affected by signal blockage and multipath. The selected section, highlighted in green in Figure 4.6(e), is shown in

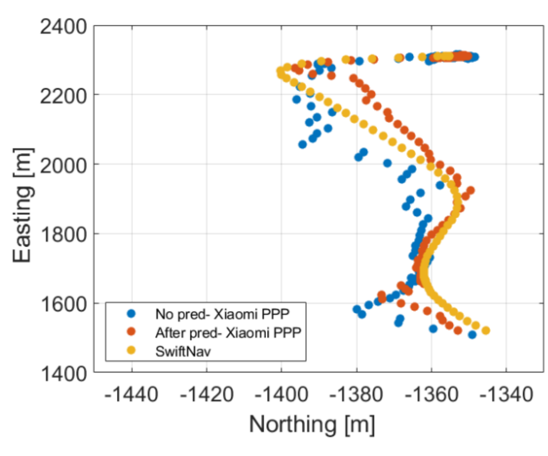
Figure 4.8(a) and (b). The chosen section was analysed for the horizontal positioning error. The major conclusions were:

1. After the prediction, there was a continuous solution with no outages, especially when the car turned.
2. There was a 55% decrease in the standard deviation of the horizontal position error for Figure 4.8(a) from 5.2 m to 2.3 m and a 54% decrease after the use of the prediction technique from 4.6 m to 2.1 m for Figure 4.8 (b). Before prediction, frequent outages increased the convergence and re-convergence time of the solution, decreasing the accuracy of the solution.

Figure 4.9 highlights the portion of the gaps in solution before prediction and the large magnitude of the horizontal error as the car turned and in areas where multipath affected the measurements considerably. Overall, the prediction strategy improved the position availability and accuracy and reduced convergence and re-convergence time.

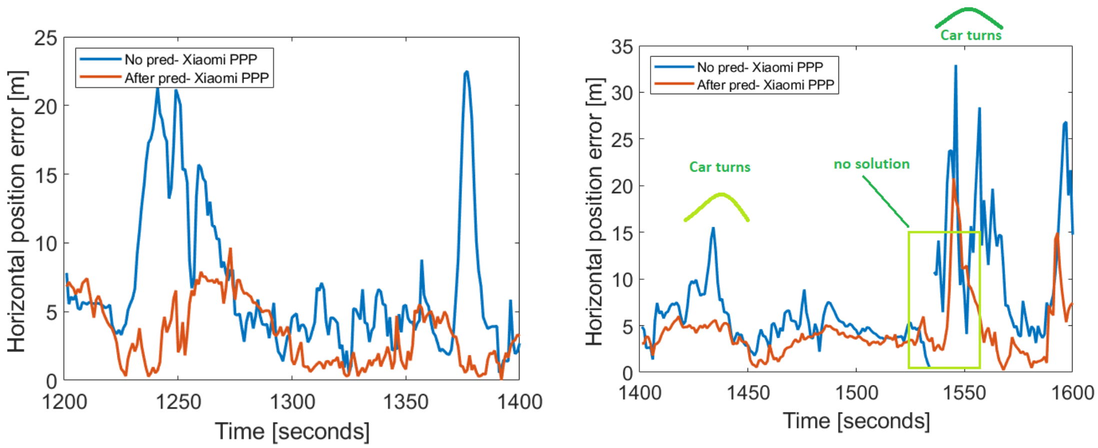



(a) Epoch 1200-1400




(b) Epoch 1400-1600

**Figure 4.8: Selected portions of the kinematic track before and after applying prediction, (DOY 325, 2019, track 2)**




(a) Epoch 1200-1400

(b) Epoch 1400-1600

**Figure 4.9: Horizontal positioning error for selected portions of the kinematic track before and after applying prediction (DOY 325, 2019, track 2)**

Overall, with the help of the prediction technique, there was a reduction of approximately 5 m in the standard deviation of the horizontal positioning error because of the improved convergence and increase in the availability of satellites. It was possible to achieve nearly 100% positioning solution availability despite the missing measurements and satellite rejections. Several other datasets were tested using this technique since it was crucial to test the datasets with the implementation of the stochastic model and prediction technique. The results for these datasets will be displayed in Chapter 5.

# Chapter 5 C/$N_0$-based Stochastic Model Enhancement

Measurement weighting parameters are an important aspect of PPP processing of raw measurements and are used to assign *a priori* weight to the code and carrier-phase measurements in the measurement weighting matrix. Typically, these weighting parameters are based on the type of measurement and quality of receivers. Carrier-phase measurements, being more precise and with less noise are given higher weighting compared to code measurements, while the weighting also depends on the quality of the receiver. In the majority of GNSS software implementations, pseudorange measurements are treated as 100 times noisier than carrier phases (Kazmierski et al. 2018).

There are a few *a priori* measurement weighting schemes such as static weight assignment, elevation-based weighting, and the C/$N_0$-based weighting. The C/$N_0$-based weighting technique is primarily preferred for the new generation of low-cost receivers. The literature contains different models to weigh measurements from low-cost receivers such as smartphones which do not have much correlation with the elevation weighting parameter, unlike geodetic receivers (Langley 1996, Banville et al. 2019).

This chapter discusses research on the different stochastic models and the implementation of a novel stochastic model that takes the code and phase multipath noise, C/$N_0$ and chipping-length/wavelength into consideration. The development of the model and its impact on the positioning solution in comparison with other stochastic models are discussed. Finally, positioning solutions obtained after implementation of the prediction and stochastic model will be presented and the overall increase in position accuracy and availability will be quantified.

## 5.1 Research on Different Stochastic Models

Typically, a function of satellite elevation has been used for observation weighting in GNSS positioning and has been traditionally used for processing measurements from geodetic or survey-grade receivers. Then, *a priori* standard deviation of the pseudorange is derived adopting zenith code standard deviation ($\sigma_P$) and satellite elevation (el) as follows (Paziewski et al. 2019):

$$\sigma_{P_k^m} = \sqrt{\frac{\sigma_P^2}{el_k^m}} \tag{5.1}$$

Another elevation-dependent weighting scheme suggested by Banville et al. (2019) is as follows:

$$\sigma = \frac{\sigma_0}{\sqrt{sin(E)}} \tag{5.2}$$

where $\sigma_0$ is the precision of the observation at zenith and E is the elevation angle.

Since the carrier-to-noise-density ratio is one of the key indicators assessing the quality of GNSS observations, this ratio can be alternatively and efficiently employed for stochastic modelling (Hartinger and Brunner 1998; Braasch and van Dierendonck 1999). For high-grade survey or geodetic receivers, both approaches (elevation and $C/N_0$-dependent) show a comparable performance due to a clear relationship between $C/N_0$ and satellite elevation angle. However, a clear benefit from $C/N_0$-dependent stochastic models is obtained in signal-degraded environments with high multipath (Medina et al. 2018), where the smartphone antenna is unable to suppress it.

In this case, *a priori* code standard deviation factor used in the weighting process can be approximated by the following equation (Langley 1996):

$$\sigma_{P_k^m} = \sqrt{\frac{\alpha B_L}{C/N_0}}\lambda_c \tag{5.3}$$

where $C/N_0$ is the carrier-to-noise-density ratio, which equals $10^{\frac{C/N_0}{10}}$ for $C/N_0$ in dBHz, λ corresponds to a wavelength of C/A or P-code (29.305 m or 293.05 m for L1 and L5, respectively), $B_L$ denotes the equivalent code loop noise bandwidth (Hz), α denotes the dimensionless DLL (delay lock loop) discriminator correlator factor. Langley (1996) defined the values for α and β, but they are for geodetic or survey-grade single-frequency GPS receivers. In research conducted by Paziewski et al (2019), it was noticed that there was a slight azimuthal asymmetry, as well as the presence of patches of low signal power or even lack of signals for a mobile device, which are absent in the case of high-grade receivers. The above observation is aligned with the investigations of Humphreys et al. (2016), who indicated a high impact of multipath affecting smartphone measurements.

There are also techniques based on the quality of satellite orbits expressed using signal-in-space ranging errors (SISRE) and $C/N_0$. SISRE is a quantity to describe the uncertainty of a modelled GNSS pseudorange due to broadcast orbit and clock errors. For smartphones and other low-cost receivers, even signals from high elevation angles can suffer from multipath due to reflection from all sides, and Carrier-to-noise-density ratio is an ideal parameter to be used in the stochastic modelling. Measurement noise, $\sigma^2_{C/A}$, $\sigma^2_{L1}/\sigma^2_{L2}/\ \sigma^2_{L5}$, of the C/A-code range and carrier-phase measurement at zenith is directly proportional to the square of the pseudorange chip length, $\lambda_{C/A}$ or carrier wavelength $\lambda_{L1}$ / $\lambda_{L2}/\ \lambda_{L5}$ and inversely proportional to the carrier-to-noise-density ratio, $C/N_0$ (Braasch and van Dierendonck 1999; Bona and Tiberius 2000; Bakker and Tiberius 2017).

Banville et al. (2019) have also suggested measurement weighting using a $C/N_0$-based stochastic model given by:

$$\sigma = a + b * 10^{-\frac{1}{2}*\frac{C/N_0}{10}} \tag{5.4}$$

where σ is the precision of the observation at zenith and coefficients a and b are estimated from filter residuals. They used a Huawei Mate 20 smartphone attached to the mast of a pole as part of a static open sky-test and concluded that the stochastic model could play a crucial role in low-cost GNSS positioning. Weighed observations using $C/N_0$ values reduced position errors by 50% over the standard elevation-based weighting that assigned same weights to observations from different constellations. They also concluded that fitting code residuals to a $C/N_0$-dependent function provided improved code-based positioning results and, therefore, resulted in smaller errors for the initial epochs of PPP solutions (Banville et al. 2019).

## 5.2 *A priori* Measurement Weighting Assignment

Low-cost receivers such as smartphones have more measurement noise and multipath affecting the raw measurement which is closely related to the irregularity in the $C/N_0$ values. $C/N_0$ values are logged in the observation files and are different for the two frequencies tracked. The $C/N_0$-based detection metrics are preferred for these low-cost receivers, along with other monitoring techniques that are more sensitive to medium and long-delay multipath. For a given signal modulation and reflection path delay, the $C/N_0$ value will be affected more by lower chipping-rate signals such as GPS L1 C/A and Galileo E1 than for high chipping-rate ones, such as GPS L5 and Galileo E5 (Groves et al. 2013; Pirsiavash et al. 2017). For geodetic receivers, this weighting is based either on widely used static values, with the standard deviation of the code measurements being 1 metre and carrier-phase being 0.01 metre or derived from the elevation value of the satellite. The standard deviation of the code and carrier-phase measurements being inversely based on the sine of the elevation value. Ultimately, the measurement weight applied varies with the elevation angle such that satellites with higher elevation receive a higher weighting of their code and carrier-phase observables since signals from satellites at high elevation angles have less noisy measurements. Humphreys et al. (2016) investigated the sky plots depicting the azimuthal-elevation dependence of $C/N_0$ values and observed that there is a slight azimuthal asymmetry as well as the presence of patches of low signal power or even lack of signals for a mobile device, which are absent in the case of a high-grade receiver. He concluded there was a high impact of multipath affecting smartphone measurements and was the primary component of the measurement noise found in the GNSS measurements from smartphones.

The post-fit residuals for the smartphone have a stronger dependence on $C/N_0$ rather than the elevation angle, while the SwiftNav residuals are directly impacted by the elevation angle. The results for three satellites observed during the entire duration of the data collection are given in Figures. 5.1 which suggests that the application of a $C/N_0$-dependent weighting scheme seems to be more appropriate than an elevation-dependent one for the smartphone. For the smartphone, the post-fit residuals for C1 remained uniformly distributed across different elevation angles for the three satellites. Hence, there is no clear evidence of elevation angle impacting the positioning solution and quality of the measurements from smartphones. However, there is a decrease in the magnitude of the residuals for satellite G01 having the highest $C/N_0$ value as shown in blue. On the other hand, for the SwiftNav Piksi, a decrease in the magnitude of the residuals with increasing elevation angle is observed with E01 having the least residual magnitude. Surprisingly, E01 had the highest residual magnitude for the smartphone, confirming the lack of satellite elevation angle playing a role in the quality of measurements from smartphones.

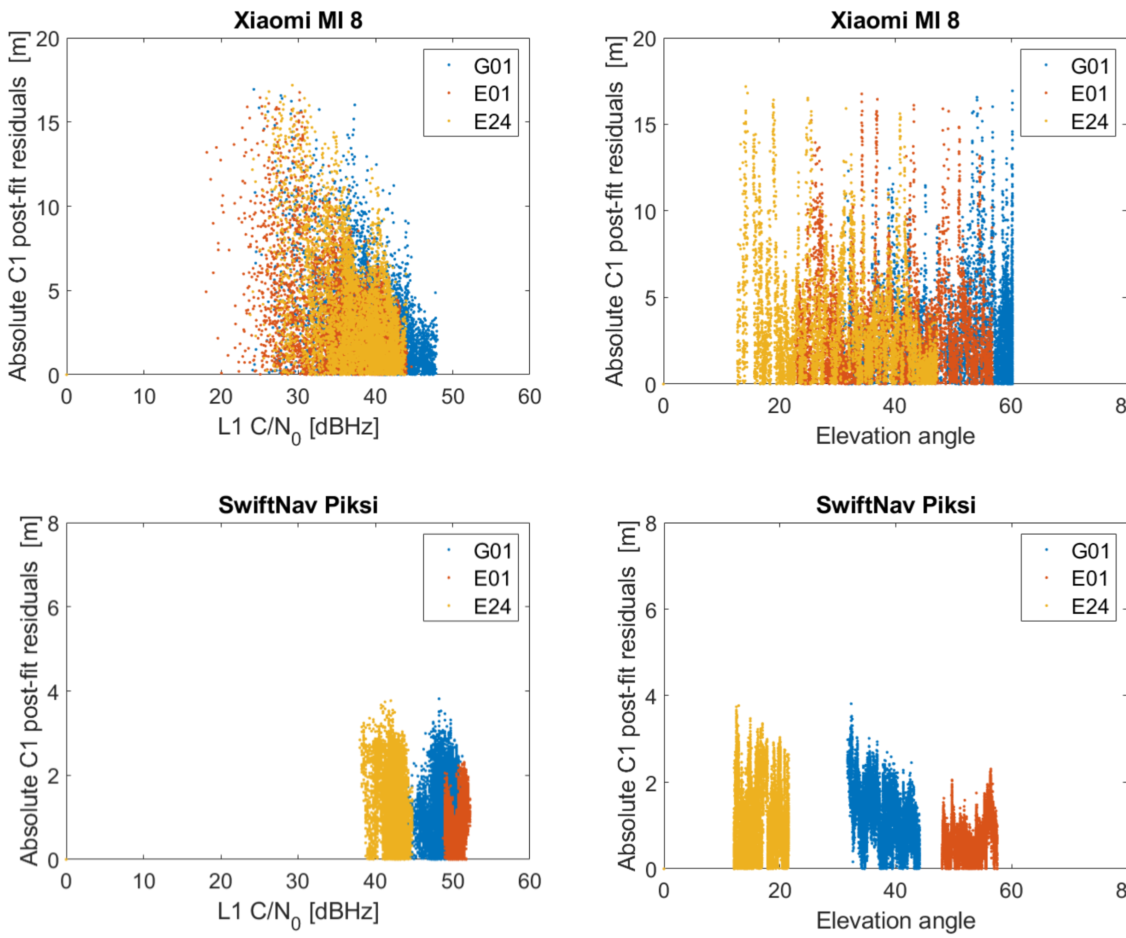



**Figure 5.1: Post-fit residuals C1/P1 as a function of elevation angle and $C/N_0$ for Xiaomi MI 8 and SwiftNav Piksi (DOY 146, 2019)**

The general equation to estimate the standard deviation of the code and carrier-phase measurements, adopted from Banville et al. (2019), is as follows:

$$\sigma = a + b * 10^{-\frac{1}{2} * \frac{C/N_0}{10}} \tag{5.5}$$

- '$\sigma$' is the standard weighting parameter for the code or carrier phase measurement in metres
- 'a' is the rms of the pseudorange multipath noise for code measurements while it is limited to the quarter of the wavelength for carrier-phase
- 'b' is the pseudorange chipping length of C/A-code or carrier-phase wavelength for code and carrier parameters, respectively.
- '$C/N_0$' is the carrier-to-noise-density ratio of the signal in dBHz

The estimated values from these computed standard deviations were used in the *a priori* weighting of the code and carrier-phase measurements. For the code measurements, 'a' was derived using the rms of the pseudorange multipath noise for all datasets. For the mannequin dataset (static testing) presented in section 5.3, 'a' was 6 m for the L1/E1 code measurements and 7.5 m for the L5/E5a code measurements. For the kinematic driving dataset presented in Section 5.4, 'a' was 8 m for the L1/E1 code measurements and 9.1 m for the L5/E5a code measurements For the carrier-phase measurements, the multipath noise is limited to a quarter of a wavelength for geodetic and other low-cost receivers. For smartphones, however, it is higher. Hence for static testing, the value of 'a' was limited to 10 cm for both L1/E1 and L5/E5a carrier-phase measurements. It was ideal taking 1 m in kinematic medium to high multipath scenarios since the uncertainty due to the prediction in the carrier-phase measurement leads to an error of around 1 m. For the code measurements, 'b' is the pseudorange chipping length of C/A code which is 293 m for L1/E1 code measurements and 29.3 m for L2 and L5/E5 code measurements. For carrier-phase measurements, it is the carrier-phase wavelength, which is 19.05 cm for L1, 24.45 cm for L2 and 25.48 cm for L5 measurements.

An elevation-based model has been used for assigning weighting for measurements from receivers other than smartphones in the YorkU PPP software and is used for processing data from the SwiftNav receiver. The standard deviation of the code and carrier-phase measurements can be defined as follows:

$$\sigma_{code} = \frac{\sigma_{sdp}}{a+b*sin\left(\frac{ele*\pi}{180}\right)} \tag{5.6}$$

$$\sigma_{phase} = \frac{\sigma_{sdcp}}{a+b*sin\left(\frac{ele*\pi}{180}\right)} \tag{5.7}$$

where

$\sigma_{code}$ is the weight assigned to the code measurement

$\sigma_{phase}$ is the weight assigned to the phase measurement

ele is the elevation angle of the satellite

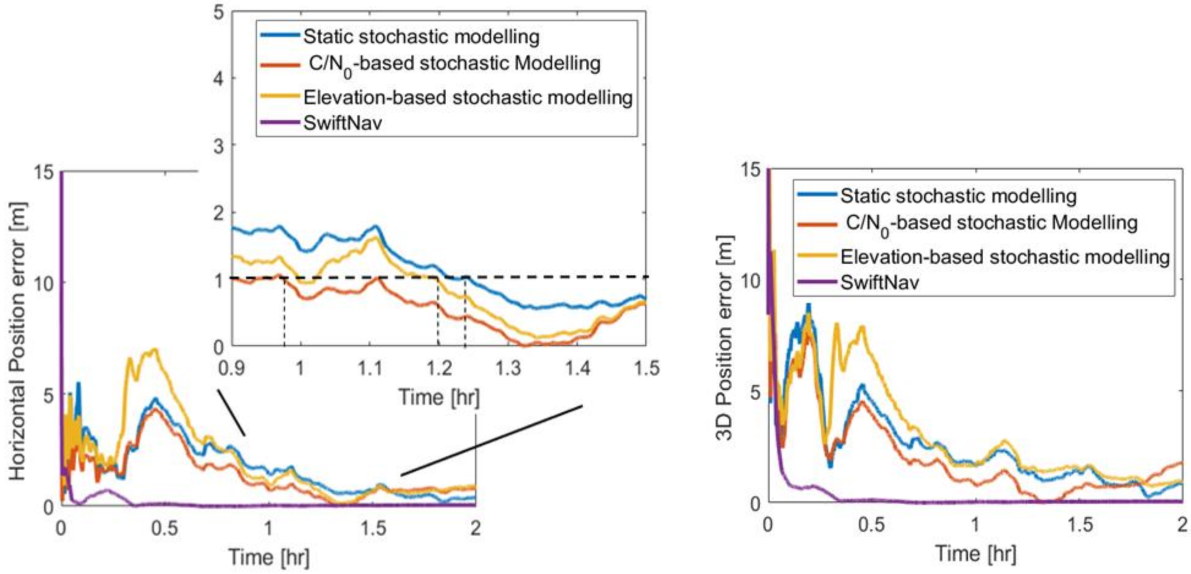
$\sigma_{sdp/sdcp}$ is the *a priori* weight assigned to the measurements based on the quality of the receiver.

For IGS stations, the typical noise level of the pseudorange measurements is around 0.3 m while, at low elevation angles of around 10°, the pseudorange noise reaches around 1-1.5m. For carrier-phase measurements, the noise-levels are around 0.003 cm. For low-cost receiver such as the SwiftNav Piksi, the pseudorange noise reaches around 2-3m at low elevation angles of around 10°. For carrier-phase measurements, the noise-levels are around 0.02 cm. Based on the values that the function takes for these specific elevation angles, a=0.15 gives a function that follows the expected noise level for elevation angle 90°.

## 5.3 Assessment of the Performance of the C/$N_0$-based Stochastic Model on Static Data

The static dataset collected with the phone clamped onto the hand of the mannequin was processed using three different stochastic models to assess positioning accuracy and residual magnitude. These models are 1) static stochastic modelling; 2) elevation-based stochastic modelling; and 3) C/$N_0$-based stochastic modelling. For the static weight assignment, 8 m and 1 m was selected as the *a priori* weighting factor for the L1/E1 and L5/E5a code and carrier-phase measurements, respectively. The elevation-based measurement weight assignment used the residuals to estimate the *a priori* weights. For the C/$N_0$-based measurement weight assignment, 6 m for the L1/E1 code measurements, 7.5 m for the L5/E5a code measurements and 0.1 m for the L1/E1 and L5/E5a carrier-phase measurements, were empirically chosen as the *a priori* weights for the code and carrier-phase measurements. The PPP solution from the SwiftNav Piksi was taken as reference. An elevation-based weighting strategy was employed to process the GNSS measurements from the Piksi receiver.

Figure 5.2 compares the horizontal and vertical positioning accuracy and convergence time for the three stochastic modelling methods. The horizontal 2D and 3D positioning accuracies after convergence are also presented in Table 5.1. Dual-frequency L1/E1 and L5/E5a code and carrier-phase measurements from GPS and Galileo were processed. The convergence threshold was chosen to be 1 m since the desire was to achieve sub-metre level positioning accuracy. Post-C/$N_0$-based stochastic modelling the dataset converged in 58 minutes as compared to 72 minutes for the elevation-based model and 75 minutes for the static stochastic model. The C/$N_0$-based stochastically modelled data converged to a horizontal rms of 0.7 m compared to 0.8 m for the statically modelled data and 0.9 m for the elevation-based stochastically modelled data. Moreover, the $C/N_0$-based stochastically modelled data had the least 2D and 3D initialization error as shown using the red line in Figure 5.2. Hence, the $C/N_0$-based stochastic model showed better initialization, convergence and the least 2DRMS and 3DRMS errors.




(a) 2D (b) 3D

**Figure 5.2: 2D and 3D position plots using the three different stochastic models compared against the SwiftNav Piksi (DOY 146, 2019)**

**Table 5.1: Positioning error using the three different stochastic models for the smartphone (DOY 146, 2019)**

| Scenario | 2DRMS (m) | 3DRMS (m) | 2D convergence time (min) |
|---|---|---|---|
| Static stochastic modelling → Xiaomi MI 8 | 0.8 | 1.4 | 75 |
| $C/N_0$-based stochastic modelling → Xiaomi MI 8 | 0.7 | 1.3 | 58 |
| Elevation-based stochastic modelling → Xiaomi MI 8 | 0.9 | 1.7 | 74 |
| Elevation-based stochastic modelling → SwiftNav Piksi | 0.1 | 0.2 | 9 |

The residuals obtained in the static weighting case clearly with higher rms values, re-enforce the need of using a stochastic model that takes both multipath noise factor, $C/N_0$ and chipping-length or wavelength into consideration. Table 5.2 represents the post-fit residuals for the four observables using the three different stochastic models. The residuals using the static model were the nosiest since they were randomly selected without analysing the noise, multipath or residuals of the dataset. Such a model can cause higher residual magnitude with other datasets if the *a priori* values chosen do not correspond with the quality of the measurements as can be seen for the L1 and L5 carrier-phase residuals. The $C/N_0$-based stochastic model had a lower residual magnitude than the elevation because it ensured measurements with higher noise had a lower *a priori* weight.

**Table 5.2: Rms of the post-fit residuals using the three different stochastic models for the smartphone (DOY 146, 2019)**

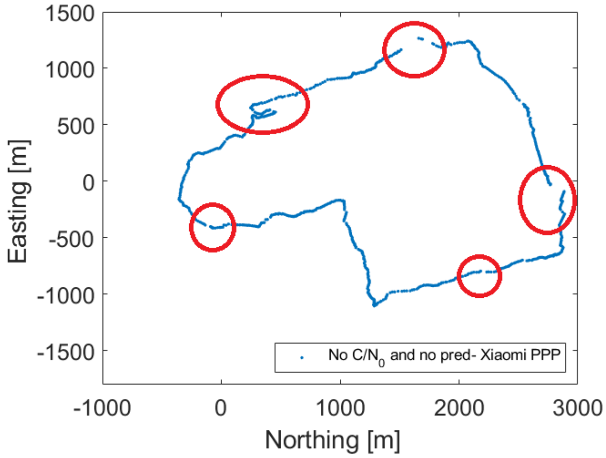
| Scenario | Post-fit C1 (m) | Post-fit L1 (cm) | Post-fit C5 (m) | Post-fit L5 (cm) |
|---|---|---|---|---|
| Static stochastic modelling → Xiaomi MI 8 | 13.8 | 32.7 | 2.3 | 120 |
| $C/N_0$-based stochastic modelling → Xiaomi MI 8 | 4.3 | 6 | 1.9 | 4.3 |
| Elevation-based stochastic modelling → Xiaomi MI 8 | 5.2 | 9 | 2.1 | 6.1 |

Overall, the $C/N_0$-based stochastic model increases positioning accuracy and decreases convergence time with less residual noise as it considers the measurement quality parameters. The $C/N_0$-based stochastic model along with the prediction technique will now be applied on a kinematic dataset and will be used to obtain the final positioning solution.

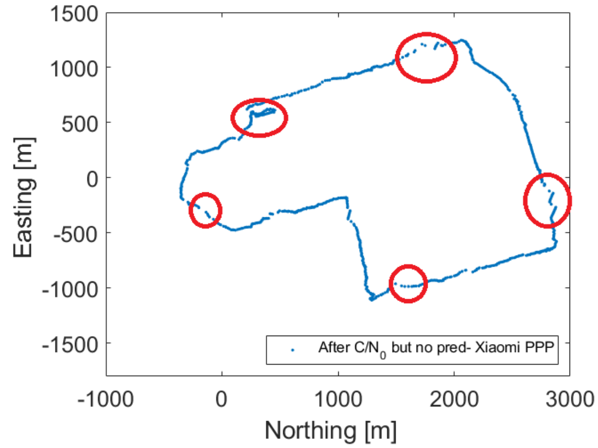
## 5.4 Assessment of the Performance of Measurement Prediction and the C/$N_0$-based Stochastic Model on Kinematic Data

The kinematic dataset collected while driving in a medium multipath environment on DOY 325, 2019 (track 1), was analysed for positioning accuracy and solution availability (a) before the application of the C/$N_0$-based stochastic model and prediction technique; (b) after the implementation of the C/$N_0$-based stochastic model but no prediction; and (c) after the application of the C/$N_0$-based stochastic model and prediction. The base station comprised a Topcon-NET G3A placed on the Petrie rooftop. As mentioned in Table 3.3, the smartphone was placed on the dashboard of the car while the low-cost SwiftNav Piksi antenna connected to the SwiftNav Piksi receiver was on the rooftop of the car. Again, it is important to note that the antenna phase centres of the SwiftNav and the phones were not aligned. The SwiftNav Piksi antenna was on the roof and had better signal availability while the smartphone was inside the car, on the dashboard to mimic the real usage of a smartphone while navigation. The multipath and signal blockage is enhanced by reflections due to the windshield and the car interiors. There should be at least a metre offset in the solutions for the two receivers and the number of satellites tracked by the smartphone antenna is limited by its position within the car. All these factors must be taken into consideration while comparing the positioning solution. Hence, the overall objective was to compare the positioning solution accuracy before and after applying the C/$N_0$-based stochastic model and prediction.

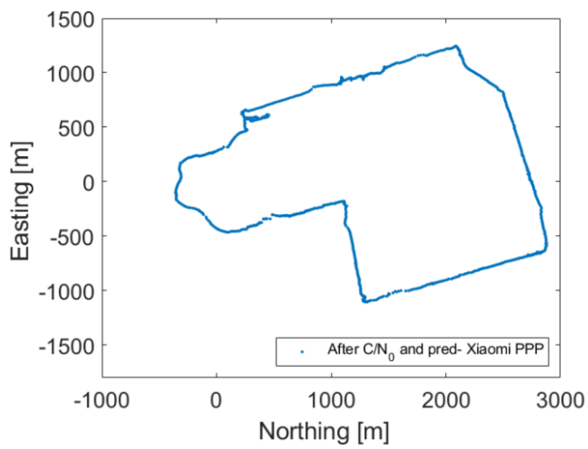
Figure 5.3 represents the horizontal position solution for the three scenarios. The red boxes indicate the periods of no solution and divergence. There were significant divergence and periods of no solution in the PPP processing done without any measurement conditioning and using the static stochastic model. With the implementation of prediction and C/$N_0$-based stochastic model, a solution with 100% availability was achieved in Figure 5.3(c). The accuracy of the positioning solution improved considerably over the region between epoch 1500 and 1900, highlighted in Figure 5.3(e).



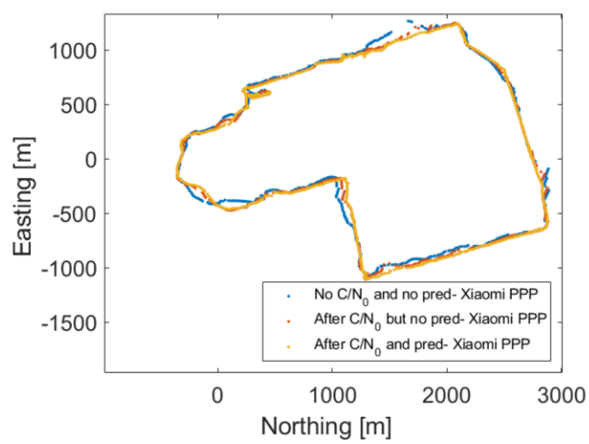

(a) No stochastic model and no pred



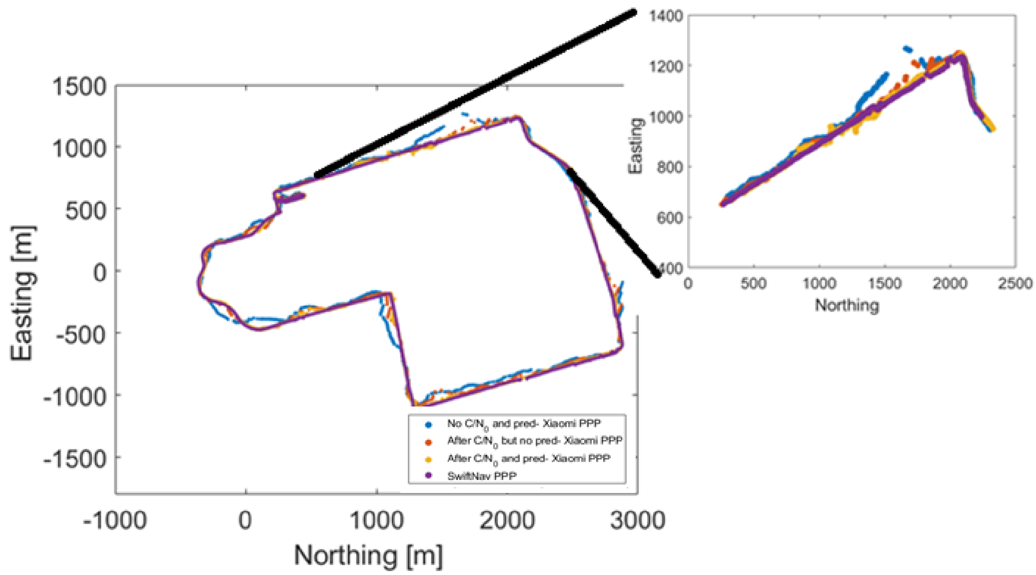

(b) After C/N$_0$ stochastic model but no pred




(c) After C/N$_0$ stochastic model and pred



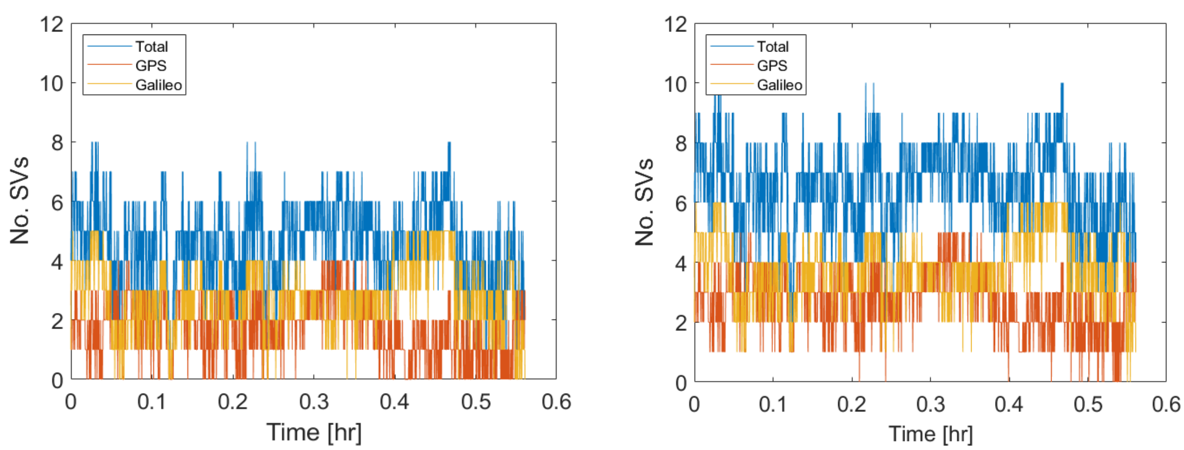

(d) Comparison of the three solutions




(e) Overall solution comparison with SwiftNav PPP reference solution

**Figure 5.3: Xiaomi MI 8 PPP kinematic track before and after the use of a C/N$_0$-based stochastic model and prediction against the SwiftNav Piksi reference (DOY 325, 2019)**

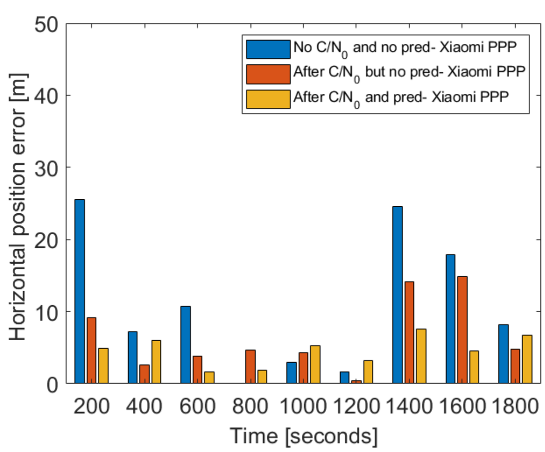
Observing the number of satellites processed before and after prediction as seen in Figure 5.4 it was seen that the average number of satellites being processed rose from an average of 5 to 8.



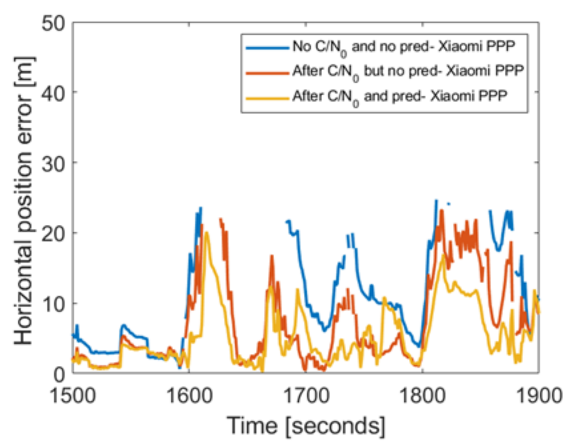

(a) Before prediction (b) After prediction

**Figure 5.4: Number of satellites processed before and after applying C/N$_0$-based stochastic model and prediction (DOY 325, 2019)**

Overall, there was a 64% decrease in rms and a 64.3 % decrease in the standard deviation of the horizontal positioning error and a 1.3% increase in solution availability after the conditioning and prediction. The prediction ensured faster convergence and re-convergence of the PPP solution. For the portion between epoch 1553-1953, a major outage happened due to signal blockage and reflection. For this portion, 22.75% (91/400) of epochs had no solution before the $C/N_0$-based stochastic model and prediction which decreased to 5.25% (21/400) after the stochastic model implementation, while after the stochastic model and prediction, a 100% continuous solution was obtained. The rms horizontal positioning error also decreased from 15.53 m to 8.38 m and finally to 5.80 m in the three steps for the portion between epochs 1553-1953. Figure 5.5 compares the horizontal positioning error for the entire track (Figure 5.5(a)) and for the portion between epochs 1500 and 1900 (Figure 5.5(b)). Table 5.3 lists the associated statistics.




(a) Entire solution




(b) Selected portion between epoch 1553-1953

**Figure 5.5: Horizontal positioning error before and after implementing the $C/N_0$-based stochastic model and prediction for the Xiaomi MI 8 (DOY 325, 2019)**

**Table 5.3: Rms and std of the horizontal positioning errors and availability of solution before and after implementing the C/N$_0$-based stochastic model and prediction for Xiaomi MI 8 (DOY 325, 2019)**

| Scenario Xiaomi MI 8 | Horizontal rms [m] | Horizontal std [m] | Percentage of solution |
|---|---|---|---|
| No prediction and no C/N$_0$-based stochastic model | 12.7 | 7.6 | 1995/2021=98.7% |
| Before prediction but after C/N$_0$-based stochastic model | 8.0 | 5.1 | 2004/2021=99.2% |
| After Prediction and C/N$_0$-based stochastic model | 4.6 | 2.7 | 2021/2021=100% |

The horizontal positioning solution does not portray a complete picture unless the residuals before and after the use of the prediction technique are scrutinized. Table 5.4 depicts the residuals for the four measurement types for the three scenarios. Overall, the C/N$_0$-based stochastic model reduced the magnitude of the residuals as compared to the statically modelled data. However, after applying prediction there was an average increase of 84% in the magnitude of the L1/E1 and L5/E5a carrier-phase residuals because of the prediction error, but it was still significantly less than the residual threshold cut-off. The code residuals were minimally impacted since they do not require any significant prediction.

**Table 5.4: Post-fit residual comparison before and after implementing the C/N$_0$-based stochastic model and prediction for the Xiaomi MI 8 (DOY 325, 2019)**

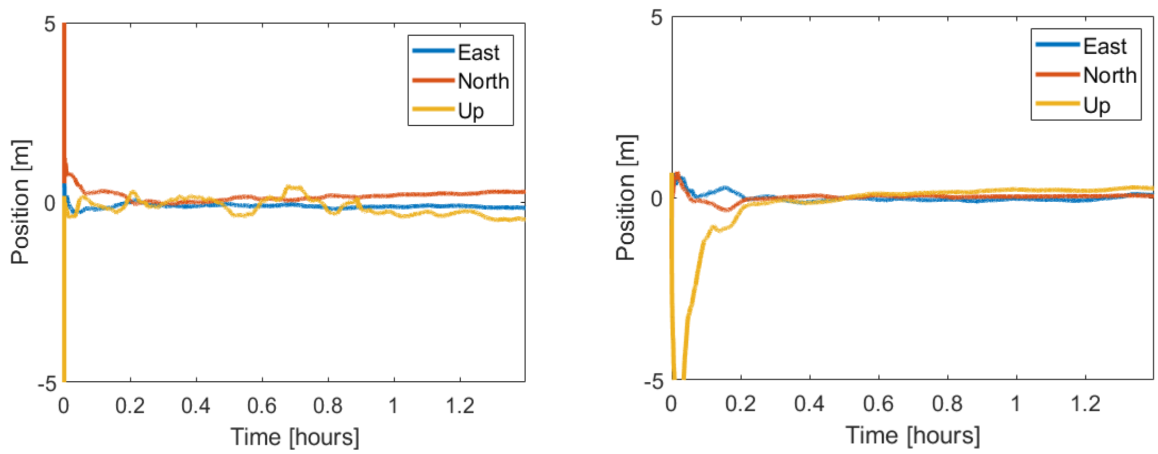
| Scenario | Post-fit C1 (m) | Post-fit L1 (cm) | Post-fit C5 (m) | Post-fit L5 (cm) |
|---|---|---|---|---|
| No prediction and no C/N$_0$-based stochastic model | 4.7 | 12.1 | 2.2 | 7.3 |
| Before prediction but after C/N$_0$-based stochastic model | 3.4 | 8.9 | 1.7 | 4.6 |
| After Prediction and C/N$_0$-based stochastic model | 3.7 | 53.9 | 3.2 | 34.8 |

## 5.5 Overall Assessment of Measurement Conditioning in Different Data Collection Scenarios

This section compares the PPP positioning performance some other static and kinematic datasets collected using the Xiaomi MI 8 with the SwiftNav Piksi reference after applying the prediction and $C/N_0$-based stochastic model.

### 5.5.1 Static Testing

Various static datasets were collected on the Petrie Rooftop. The dataset of interest was collected by attaching the phone vertically to a holder on a tripod. The ENU convergence plot for the static test is shown in Figure 5.6. The convergence time was taken as the time for the solution to converge to a positional accuracy of 1m as the smartphone solution is limited by the poor measurement quality and multipath. The positioning solution was compared against the SwiftNav Piksi.



(a) Xiaomi MI 8 (b) SwiftNav Piksi

**Figure 5.6: ENU plot for static data for the Xiaomi MI 8 and the SwiftNav Piksi for static data (DOY 298, 2019)**

With the Xiaomi MI 8, decimetre-level positioning solution was achieved, and the performance of the smartphone was comparable to the SwiftNav Piksi. The smartphone positioning solution converged to a sub-metre horizontal positioning displacement in 9 minutes and had a horizontal accuracy of 21.9 cm, as compared to 11.4 cm offered by the Piksi whose positioning solution converged in 6 minutes. The Piksi solution, expectedly, had a better initialization and convergence. The horizontal convergence plot is shown in Figure 5.7 and the associated statistics are given in Table 5.5.

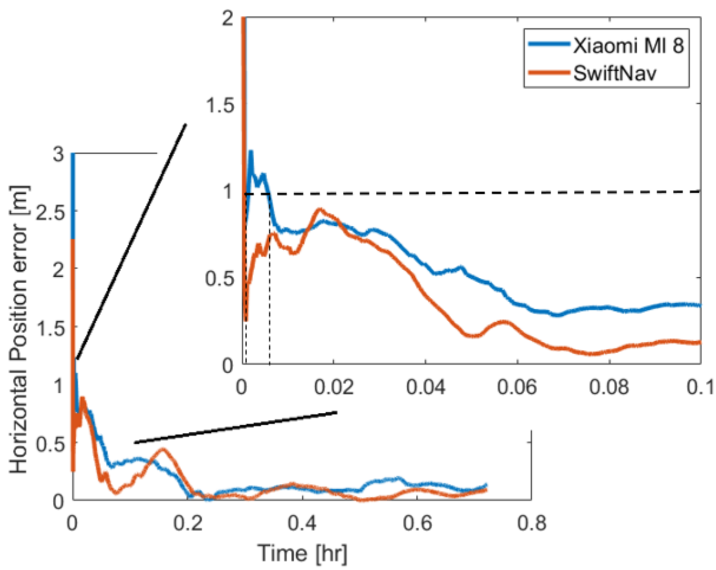



**Figure 5.7: Horizontal convergence plot for static data for Xiaomi MI 8 versus SwiftNav Piksi for data collected in an open sky environment (DOY 298, 2019)**

**Table 5.5: Positional accuracy for static data for Xiaomi MI 8 versus SwiftNav Piksi collected in an open sky environment (DOY 298, 2019)**

| Device | 2DRMS (cm) | 3DRMS (cm) | Convergence Time (min) |
|---|---|---|---|
| Xiaomi MI 8 | 21.9 | 34.8 | 9 |
| SwiftNav Piksi | 11.4 | 29.7 | 6 |

The pseudorange and carrier-phase residuals for the smartphone and the Piksi are shown for the static case in Figure 5.8 and 5.9. Though they both follow typical noise-like trends, the smartphone residuals had a few outliers and jumps, which could be attributed to uncorrected cycle slips. Overall, the smartphone residuals had comparable than their Piksi counterparts. The C1 code residuals for smartphone, however, had an rms which was 82% higher than their P1 counterparts for the Piksi. This is possibly due to the high multipath noise found in the code measurements. The rms values for the residuals have been tabulated in Table 5.6.

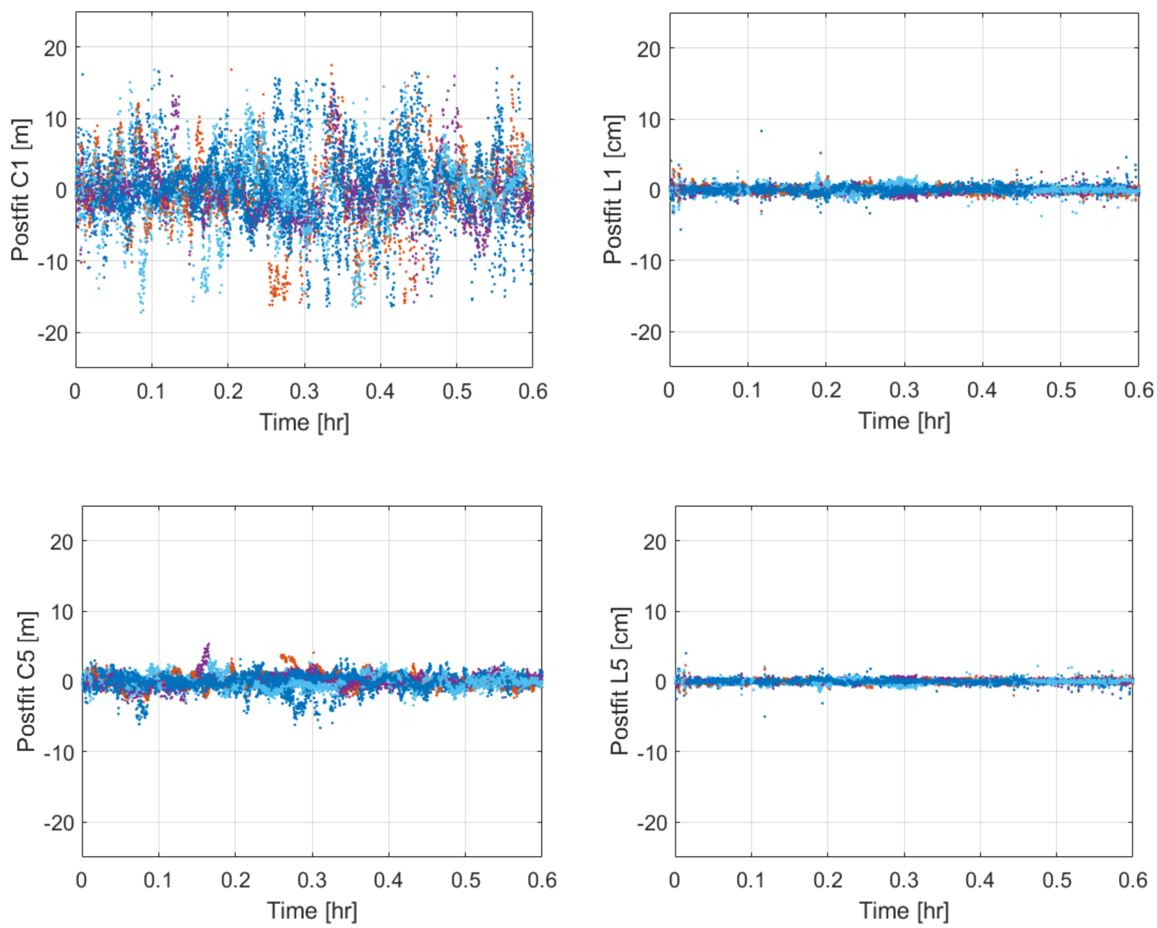



**Figure 5.8: Post-fit residuals for the Xiaomi MI 8 for static data (DOY 298, 2019)**

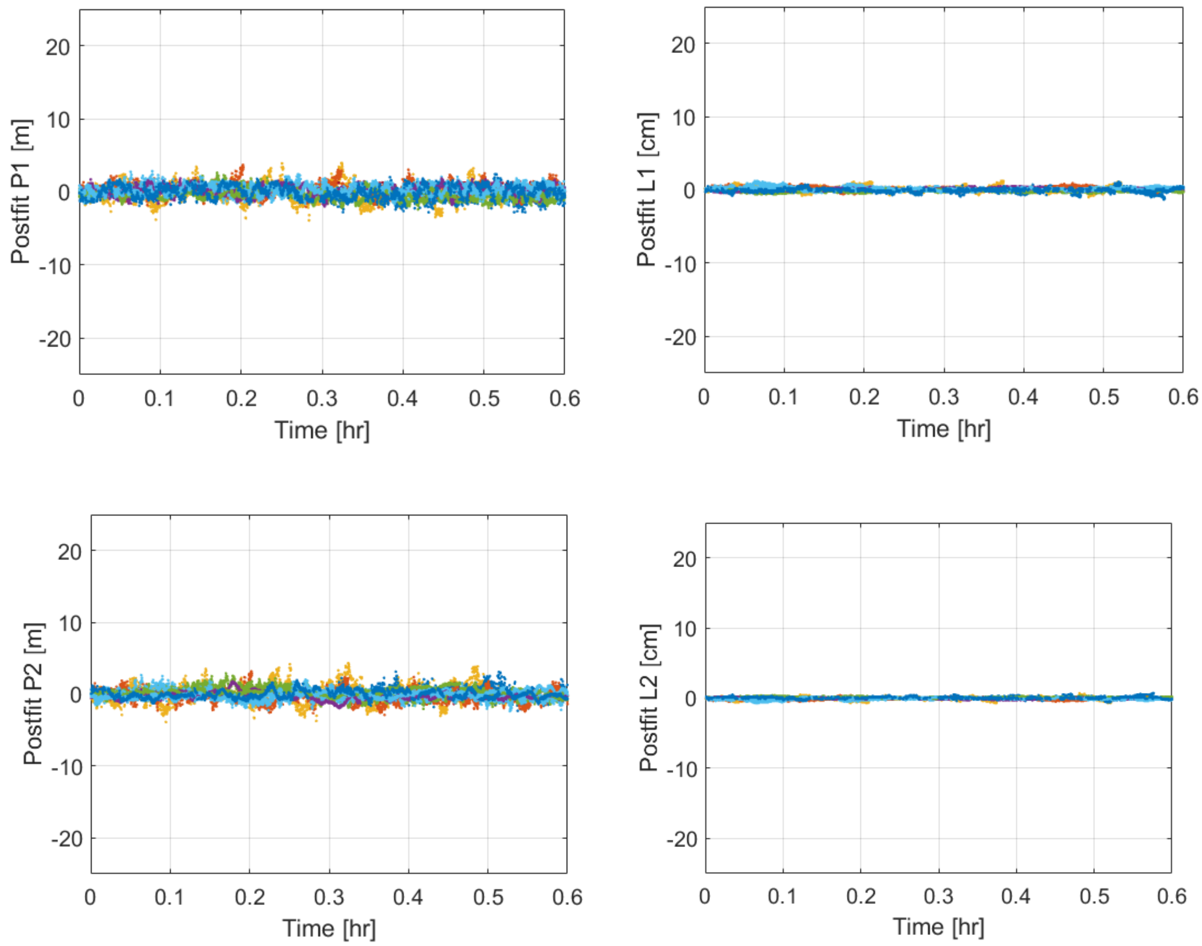


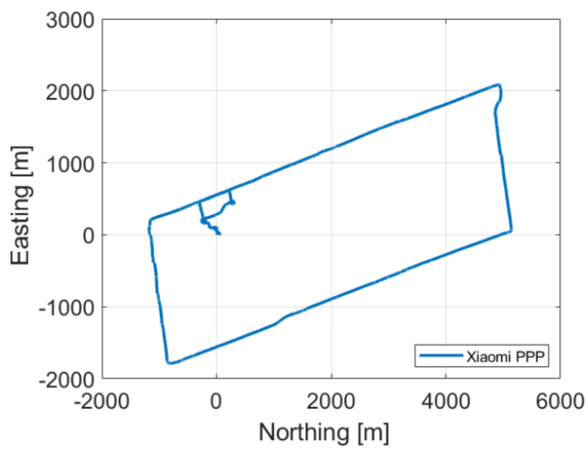

**Figure 5.9: Post-fit residuals for the SwiftNav Piksi for static data (DOY 298, 2019)**

**Table 5.6: Post-fit residuals for Xiaomi Mi 8 and SwiftNav Piksi for static data (DOY 298, 2019)**

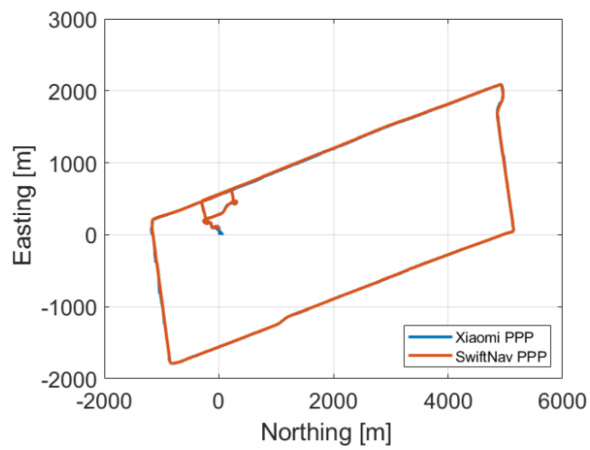
| Device | Post-fit C1/P1 (m) | Post-fit L1 (cm) | Post-fit C5/P2 (m) | Post-fit L2/L5 (cm) |
|---|---|---|---|---|
| **Xiaomi MI 8** | 4.5 | 1.2 | 0.9 | 0.9 |
| **SwiftNav** | 0.8 | 0.3 | 0.8 | 0.2 |

### 5.5.2 Kinematic Testing

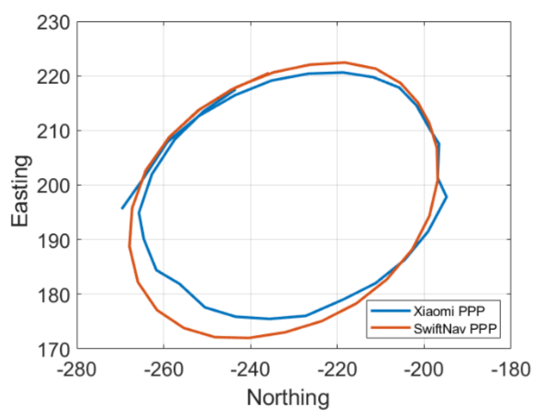
The kinematic dataset being analysed was collected on DOY 85, 2019 by keeping the phone on the dashboard of a car. The horizontal PPP positioning solution is shown in Figure 5.10(a). The positioning solution was compared against the SwiftNav Piksi PPP solution (Figure 5.10(b)). A specific portion between epoch 2290-2320 has been additionally shown to highlight the performance of the smartphone receiver during a circular track when the car turned, making the antenna vulnerable to additional signal blockages and multipath (Figure 5.10(c)).




(a) Xiaomi MI 8 PPP




(b) Xiaomi vs SwiftNav PPP



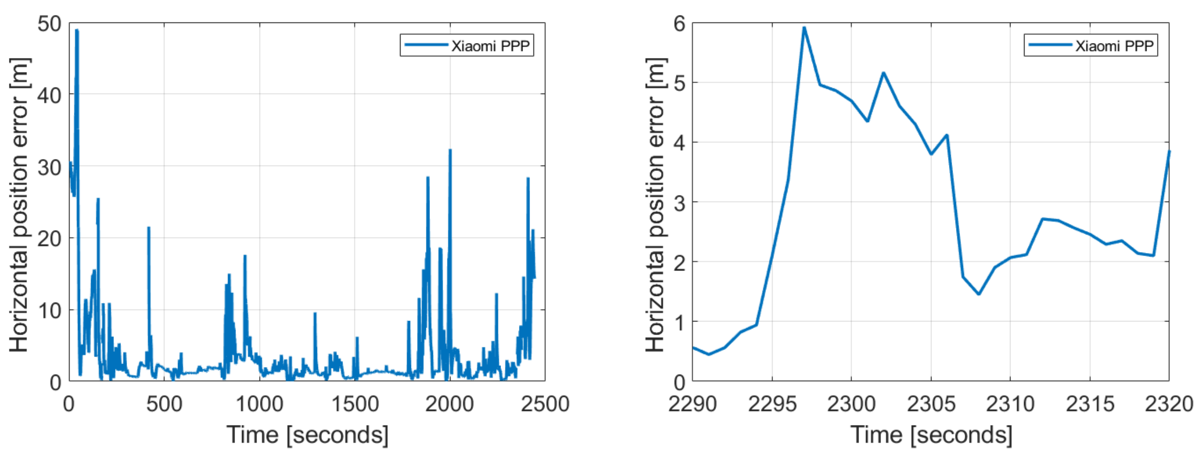

(c) The zoomed-in portion between epochs 2290-2320

**Figure 5.10: Horizontal positioning plot for kinematic data (DOY 85, 2019)**

The rms and standard deviation of the horizontal positional error for the Xiaomi MI 8 was computed taking the SwiftNav solution as a reference. The rms and std of the positioning solution were 6.84 m and 5.20 m for the whole track as shown in Table 5.7. Figure 5.11 presents the horizontal positioning errors for the entire track and the selected portion. For the portion between epochs 2290-2320, the rms error was 5.58 m as shown in Figure 5.11 (b). The PPP solution with the $C/N_0$-based stochastic model and prediction technique had 100% solution availability. In Chapter 6, these results will be compared with the positioning solution offered by other techniques such as RTK, SPP and the smartphone's internal positioning solution.

**Table 5.7: Horizontal positional accuracy for kinematic data for the Xiaomi MI 8 versus the SwiftNav Piksi**

| Device | Horizontal Positioning Error | |
|---|---|---|
| | rms (m) | Standard deviation (m) |
| **Xiaomi MI 8** | 6.84 | 5.20 |




(a) Overall solution

(b) Selected portion between epochs 2290-2320

**Figure 5.11: Horizontal Position error for the Xiaomi PPP solution (DOY 85, 2019)**

The pseudorange and carrier-phase residuals for the smartphone and the Piksi are shown in Figure 5.12 and 5.13. The Piksi pseudorange and carrier-phase residuals were at the few metre-level range and centimetre or sub-centimetre range, respectively. The relatively higher magnitude of and certain biases in the code residuals observed for the smartphone can be due to reasons such as unmodelled hardware bias and noise given the hardware limitations of the smartphone. The carrier-phase residuals for the smartphone had an average 34% lesser magnitude than their SwiftNav counterparts which could be due to higher measurement weights applied to the SwiftNav measurements during the stochastic modelling. A normal distribution comparison of the residuals has also been carried out, subsequently. Table 5.8 summarizes the magnitude residuals for the sensors.

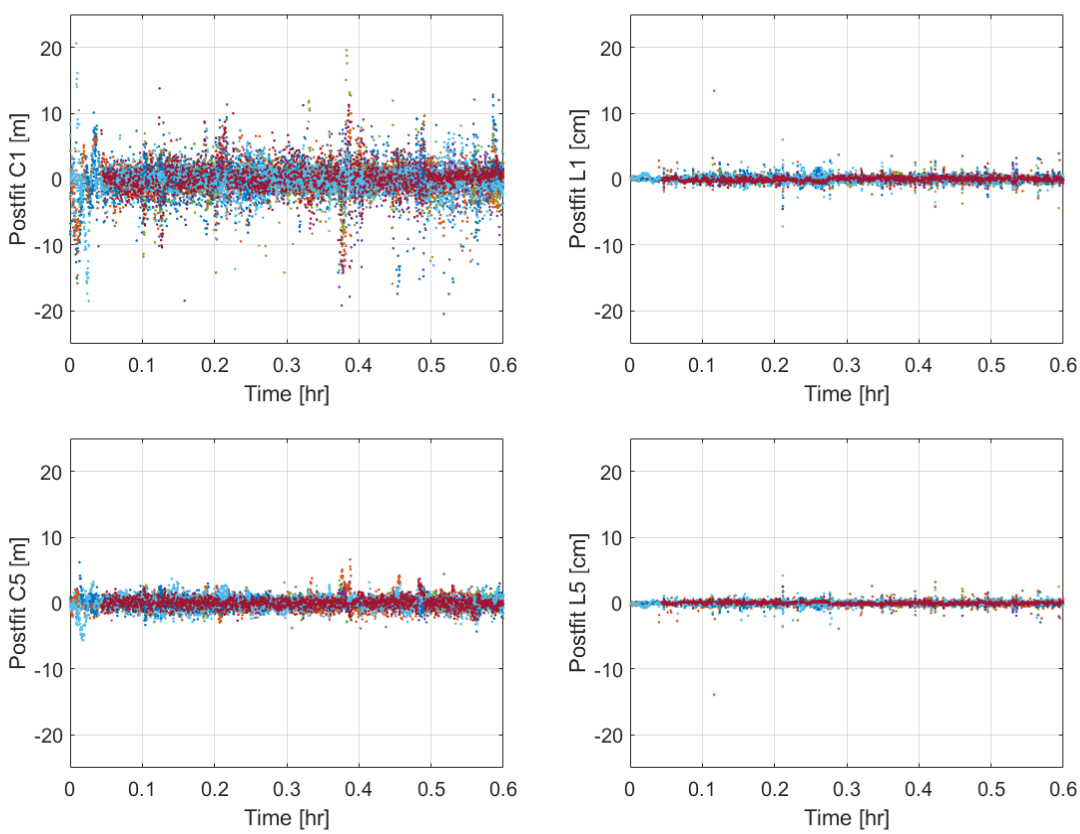



**Figure 5.12: Post-fit residuals for the Xiaomi MI 8 for kinematic data (DOY 85, 2019)**

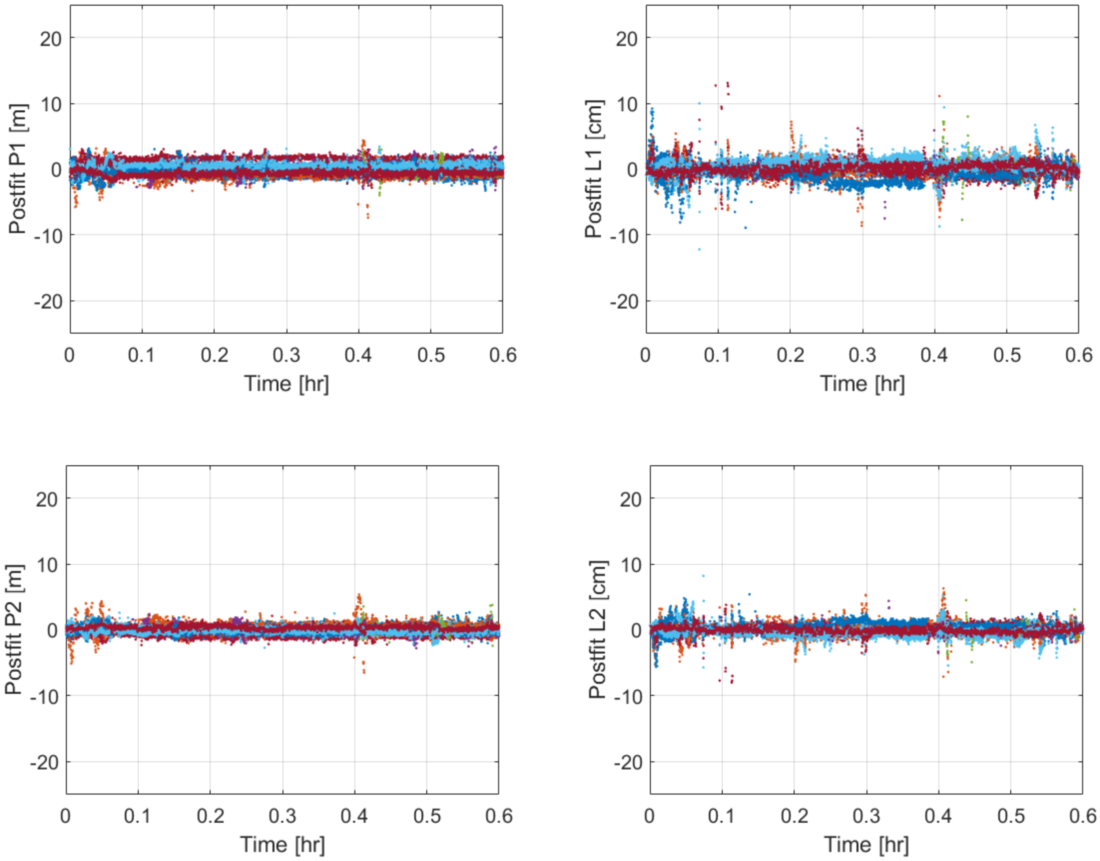



**Figure 5.13: Post-fit residuals for the SwiftNav Piksi for kinematic data (DOY 85, 2019)**

**Table 5.8: Post-fit residuals for the Xiaomi Mi 8 and SwiftNav Piksi for kinematic PPP processing (DOY 85, 2019)**

| Device | Post-fit C1/P1 (m) | Post-fit L1 (cm) | Post-fit C5/P2 (m) | Post-fit L2/L5 (cm) |
|---|---|---|---|---|
| **Xiaomi MI 8** | 2.6 | 0.9 | 1.2 | 0.7 |
| **SwiftNav** | 1.4 | 1.3 | 1.1 | 1.1 |

It is crucial to analyse the noise in the residuals as well. Ideally, the residual distribution is expected to closely follow a standard normal distribution, though practically, due to unmodelled hardware biases and noise, the residuals vary. The residuals from two satellites, GPS PRN 09 and Galileo PRN 24 are analysed. The mean and standard deviation of the normal distribution is computed and analysed for the four observables as shown in Table 5.9. Though the mean of the residuals for all the four measurements for both the receivers was close to 0, the major difference comes when the standard deviation of the observables are compared. The standard deviation of the residuals for the Xiaomi MI 8 is highlighted in orange and green for the SwiftNav Piksi. As expected, the standard deviation of the smartphone residuals have a higher magnitude and highlight the wide spectrum of values over which they are distributed. The Piksi residuals fit far better into a normal distribution curve as compared to the Xiaomi MI 8 residuals, considering the lower quality of the smartphone measurements.

**Table 5.9: Mean and standard deviation (Std) of the post-fit residuals for the Xiaomi MI 8 and SwiftNav Piksi during kinematic PPP processing (DOY 85, 2019)**

| PRN | Device | | Post-fit C1/P1 (m) | Post-fit L1 (cm) | Post-fit C5/P2 (m) | Post-fit L2/L5 (cm) |
|---|---|---|---|---|---|---|
| GPS PRN 09 | Xiaomi MI 8 | Mean | -0.3 | -0.4 | 0.1 | 0.1 |
| | | Std | 1.3 | 2.6 | 1.2 | 2.8 |
| | SwiftNav Piksi | Mean | 0.2 | 0.3 | 0.1 | -0.8 |
| | | Std | 0.9 | 0.8 | 0.9 | 0.7 |
| Galileo PRN 24 | Xiaomi MI 8 | Mean | -0.4 | -0.9 | 0.1 | 0.3 |
| | | Std | 1.9 | 2.5 | 0.9 | 2.2 |
| | SwiftNav Piksi | Mean | 0.4 | 1.4 | 0.0 | 0.8 |
| | | Std | 0.7 | 1.1 | 0.9 | 0.8 |

Overall, the Xiaomi MI 8 achieved decimetre level positioning accuracy in static testing with the help of the $C/N_0$-based stochastic model. The positioning accuracy of the smartphone in the open-sky static case was comparable to the performance of the SwiftNav. The $C/N_0$-based stochastic model and prediction technique decreased the horizontal rms positioning accuracy by 64% compared to the static stochastic model with no prediction. The prediction technique also helped improve position availability to 100%. The residuals for the smartphone followed typical noise-like behaviour and had comparable magnitude to their SwiftNav counterparts.

Overall, it is seen the three weighing techniques can be used based on the data collection scenario and the receiver used. For most geodetic and medium to low-cost equipment, the elevation-based weighing technique is preferred since the measurement quality is directly dependent on the elevation angle. For geodetic equipment and a controlled static environment with prior information on the quality of measurements, the static measurement weight assignment can also be used. However, for smartphones, since there is a direct dependence of measurement quality on multipath and $C/N_0$, the $C/N_0$-based stochastic model is preferred.

# Chapter 6 Comparison of Different Positioning Solutions

There are several other ways of generating GNSS positioning solutions such as RTK, SPP and the internal phone solution. Their principle of operation and functioning have been described in Chapter 2. This chapter focuses on processing the GNSS measurements obtained from smartphones using RTK, SPP and extracting the internal phone solution for comparison with the positioning accuracy with the obtained PPP solution. The different comparisons are useful to analyse the improvement in positioning accuracy and availability with the help of PPP augmentation.

## 6.1 SPP and PPP Solution Comparison

Most GNSS chips found in smartphones track only single-frequency GPS code measurements. The positioning has traditionally been computed using the SPP technique. Hence, it was necessary to test the positioning accuracy of the SPP technique with PPP solution obtained by the YorkU PPP engine. A weighed-least squares epoch-by-epoch SPP software is used to process the GPS L1 C/A code measurements.

Data were collected on the rooftop of Petrie Science and Engineering Building, York University on DOY 298, 2019 which was the standard open sky static positioning test where the phones were clamped on to a phone holder attached to a tripod. It was best to compare the solutions in ideal open sky conditions to achieve the best possible SPP result as in degraded and difficult environments, the SPP solution quality was expected to deteriorate, further. It is observed that the PPP solution has dm-level accuracy compared to the SPP's metre-level accurate solution. In section 5.4, the PPP solution of the dataset has been discussed while taking the SwiftNav PPP as a reference. The PPP solution converges to a horizontal positional accuracy of 1 m in 9 minutes and is smooth and continuous. The rms of the positional accuracy in the horizontal and vertical directions are 0.2 m and 0.3 m, respectively as presented in Table 6.1. On the other hand, the SPP solution is scattered and irregular with a horizontal and vertical rms of 1.9 m and 3.0 m, respectively. Figure 6.1 compares the two positioning solutions using a scatter plot.

**Table 6.1: Position accuracy for the SPP versus the PPP solution (DOY 298, 2019)**

| Xiaomi MI 8 | 2D rms (m) | 3D rms (m) |
|---|---|---|
| SPP | 1.9 | 3.0 |
| PPP | 0.2 | 0.3 |

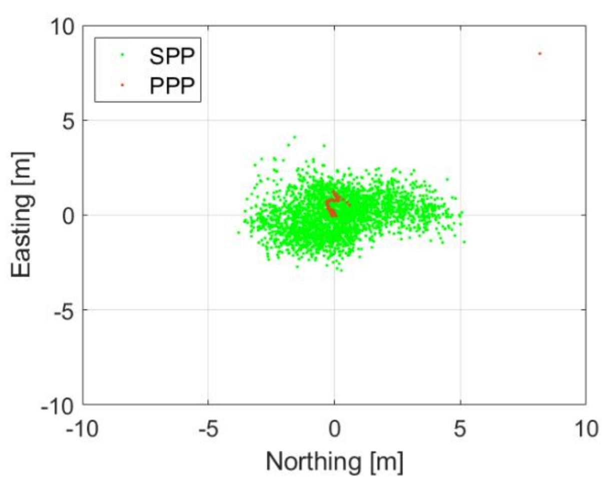
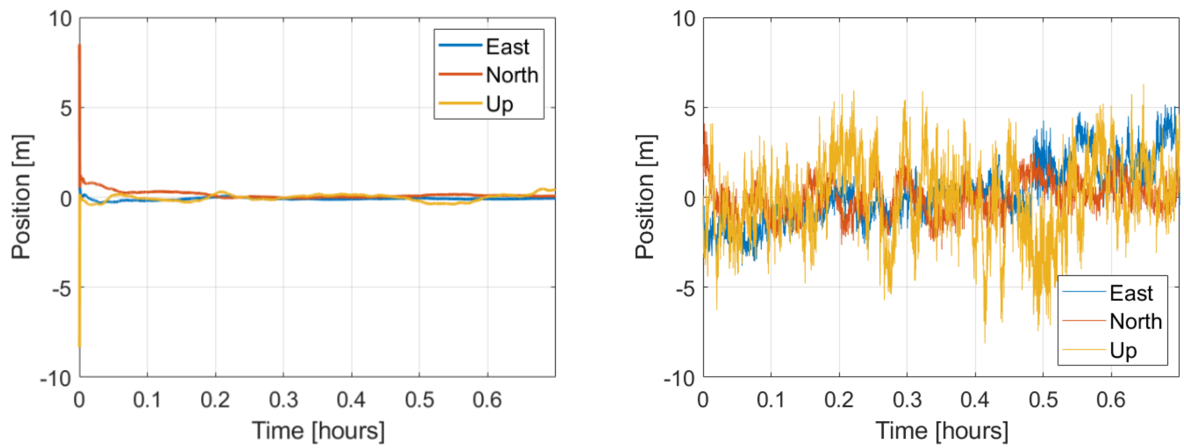



**Figure 6.1: Horizontal scatter positioning plot for the SPP and PPP solution (DOY 298, 2019)**

The horizontal and vertical positioning time series for the SPP and PPP solution is shown in Figure 6.2. The PPP solution converges smoothly and is more accurate than the SPP solution. The SPP solution is irregular with multiple jumps.




(a) PPP (b) SPP

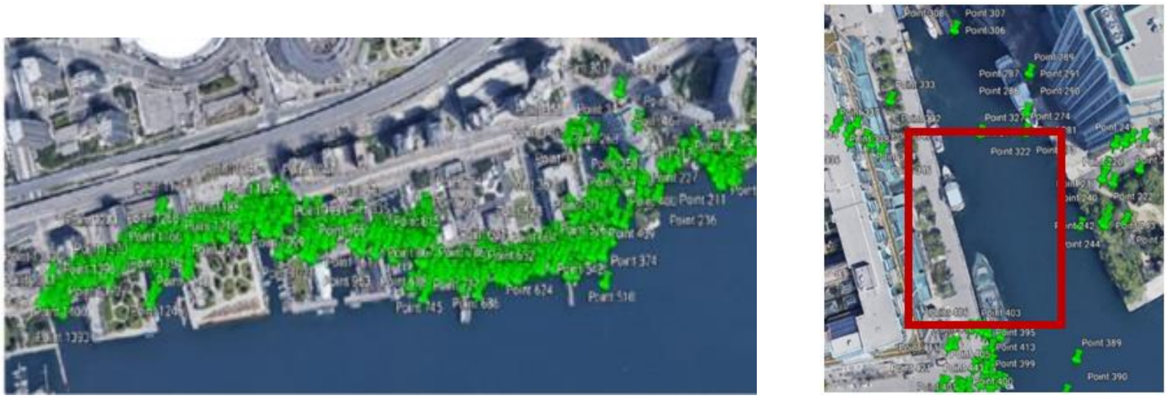
**Figure 6.2: Positioning time series plot comparing of the SPP and PPP solution (DOY 298, 2019)**

To further test the accuracy of the GPS-only L1 code SPP solution, data were collected in an urban high multipath environment with the phone in hand on DOY 54, 2019 for 40 minutes at harbourfront area of Toronto, an area characterised by tall structures and significant signal blockages. Figure 6.3 displays the SPP solution which shows biases of a few metres, irregularity and large jumps. In high multipath environments, the SPP solution has biases of tens of metres with no solution during several epochs due to signal blockages as seen by the blue line with several epochs of the positioning solution found in Lake Ontario. There are frequent reflections and large multipath noise as seen by the zig-zag nature of the position trajectory as seen in Figure 6.3(b).


(a) Complete track (b) Zoomed-in portion

**Figure 6.3: GPS L1 code-only SPP solution for a kinematic dataset collected in a high multipath urban environment (DOY 54, 2019)**

The data was processed in the PPP mode as well, though it was not possible to set-up a reference station to compare the positioning solution against a reference. Future work would deal with setting up a base station and analysing data collected in urban canyons and remote environments. Figure 6.4 presents the PPP solution with a green track. Predication was attempted but the frequent and large data gaps and the changing dynamics and environment causes metre-level biases and errors in the prediction. There were several epochs with no solution as highlighted by the gaps in the red box in the zoomed-in portion of the solution in Figure 6.4(b). Due to high multipath, there were large jumps in the PPP positioning solution with certain epochs having a solution within the lake. Though no numerical analysis of the positioning results is possible for lack of reference solution, the analysis of the data quality in a high multipath environment has been presented in Chapter 3.

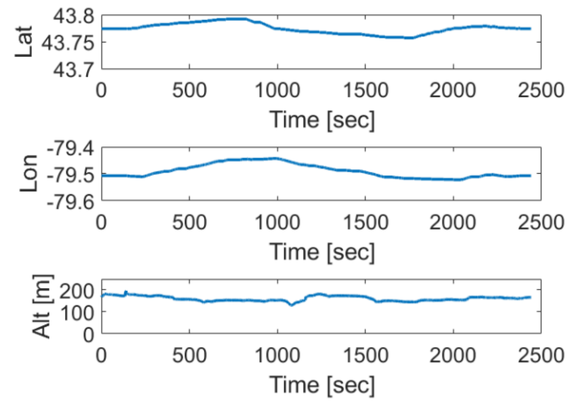


(a) Complete track (b) Zoomed-in portion

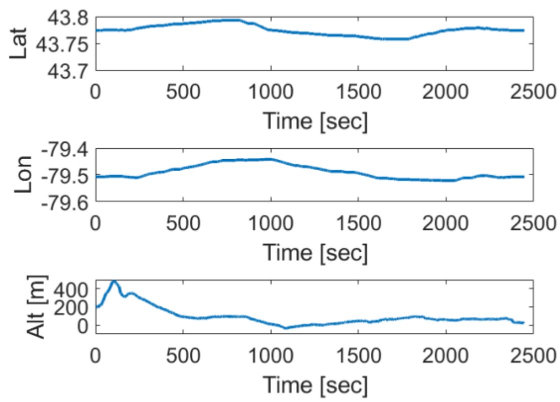
**Figure 6.4: GNSS PPP solution for a kinematic dataset collected in a high multipath urban environment (DOY 54, 2019)**

## 6.2 Internal Phone Solution

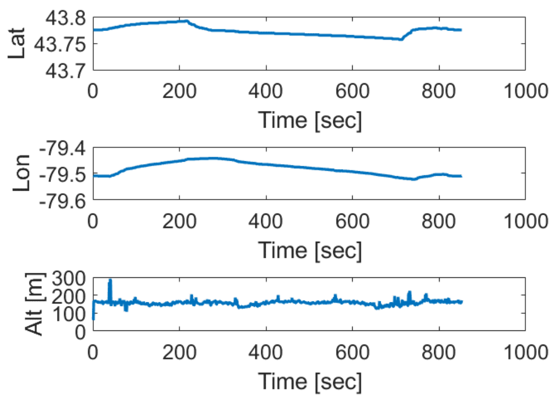
All phones output an internal solution that is a combination of a GPS L1 code-only positioning solution smoothed by possible corrections from the cell tower and ranging measurements from inertial sensors. The internal positioning solution has an accuracy of several metres in both the horizontal and vertical directions. Each phone has its proprietary method of computing the internal solution. Figure 6.5 shows the internal solution computed by different phones, Xiaomi MI 8, Huawei Mate 20, Samsung S9 and Google Pixel 3. Though the phones were kept adjacent to each other in kinematic mode on the car dashboard, the solution outputted by the phones is different. The Google Pixel 3 solution shows jumps in the altitude while Huawei Mate 20 has a large initial bias in the up direction as shown in Figure 6.5 (f). Notably, Google Pixel 3 offered a positioning solution every 3 seconds, unlike the other smartphones where there was solution present for every second. Figure 6.5(e) presents the horizontal positioning track for the internal solution offered by the four smartphones alongside with the SwiftNav Piksi internal solution. Comparison of the internal solution for the Xiaomi MI 8 with the RTK and PPP solution is discussed next.



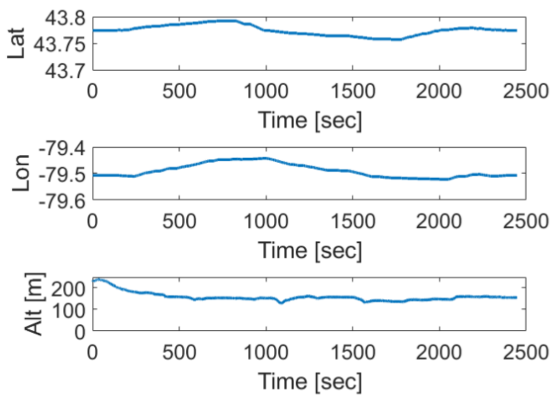

(a) Xiaomi MI 8



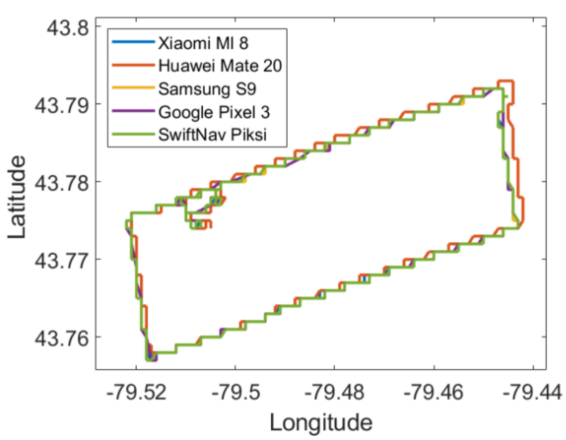

(b) Huawei Mate 20



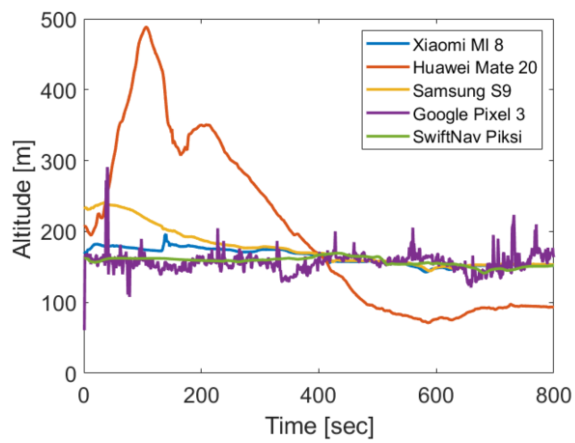

(c) Google Pixel 3




(d) Samsung S9




(e) Horizontal Track comparison




(f) Vertical comparison

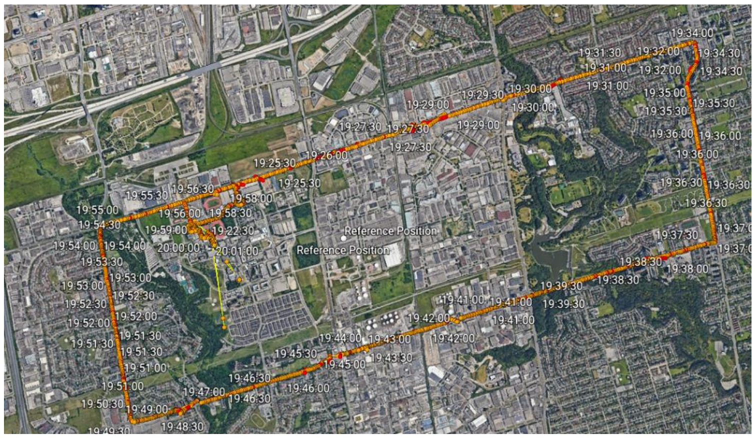
**Figure 6.5: Internal positioning solution from four different smartphones in a kinematic environment (DOY 85, 2019)**

## 6.3 RTK, Internal Phone and PPP Solution Comparison

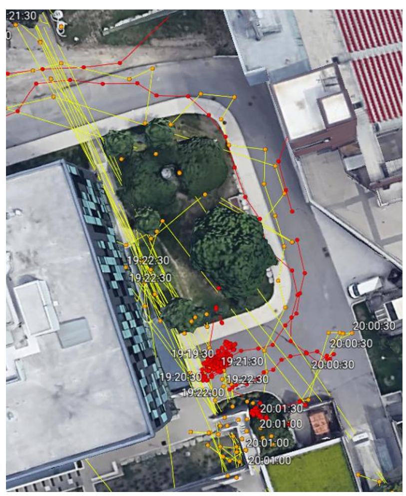
Real-Time Kinematic (RTK) is another precise positioning technique that uses code and carrier-phase measurements in addition to relying on a single reference station or interpolated virtual station to provide real-time corrections. It can provide up to centimetre-level accuracy. The RTK positioning technique is used in many high-accuracy applications such as UAV flight mapping and land surveying. RTK generally operates on a baseline of up to 20 km. The advantage of RTK is that many errors are cancelled out except for individual hardware biases. Unlike PPP, RTK processing does not suffer from drawbacks such as a long initial convergence or re-convergence time.

For analysing the positioning accuracy obtained via RTK, a base station was set up on the Petrie rooftop, which comprised the Topcon NET-G3A receiver. The receiver used to generate a reference solution was the SwiftNav Piksi. To process the Piksi measurements, the RTKLib ver.2.4.2 software was used. An elevation and a $C/N_0$ mask of 15$^{\circ}$ and 15 dBHz, respectively were used. Measurements from GPS and Galileo were processed to maintain uniformity with the PPP processing.

For comparison purposes, the kinematic dataset collected on DOY 85, 2019 was chosen where the phone was on the dashboard of the car while driving. The positioning accuracy and availability for the 1) RTK solution; 2) internal phone solution; 3) PPP solution were compared. Figure 6.6 shows the horizontal track for the RTK solution of the Xiaomi MI 8 in orange with yellow lines versus the RTK solution for the SwiftNav in the red line. The overall horizontal track does not give a clear picture of the positioning errors. Hence, a zoomed-in version of the initial few epochs and a circular portion between epochs 2290-2320 (Figure 6.6(c)) of data collection shows tens of metres of error for the RTK solution because of poor initialization and multipath errors.



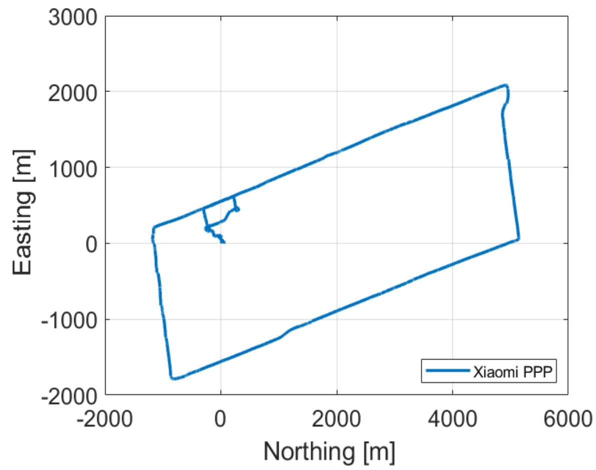

(a) Complete track



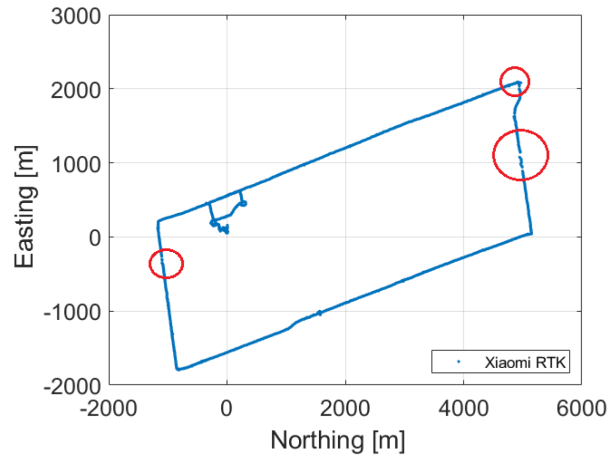

(b) A zoomed-in portion at the car park

**Figure 6.6: RTK solution for the Xiaomi MI 8 (yellow) versus SwiftNav Piksi (red) horizontal positioning solution (DOY 85, 2019)**

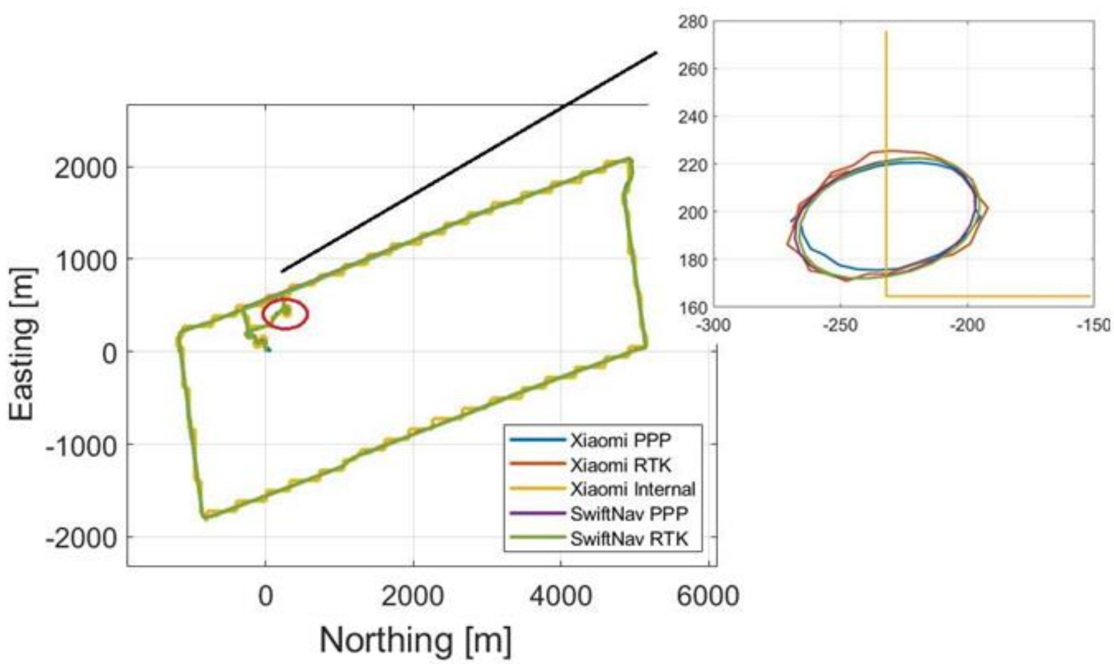
The horizontal positioning track for comparing three positioning solutions is shown in Figure 6.7. A specific portion between epoch 2290-2320 has been additionally shown to highlight the performance of the smartphone receiver during a circular track when the car turned, making the antenna vulnerable to additional signal blockages and multipath (Figure 6.7(c)). Figure 6.7(a) shows the PPP solution while the RTK solution with gaps is shown in Figure 6.7(b).




(a) Xiaomi PPP




(b) Xiaomi RTK



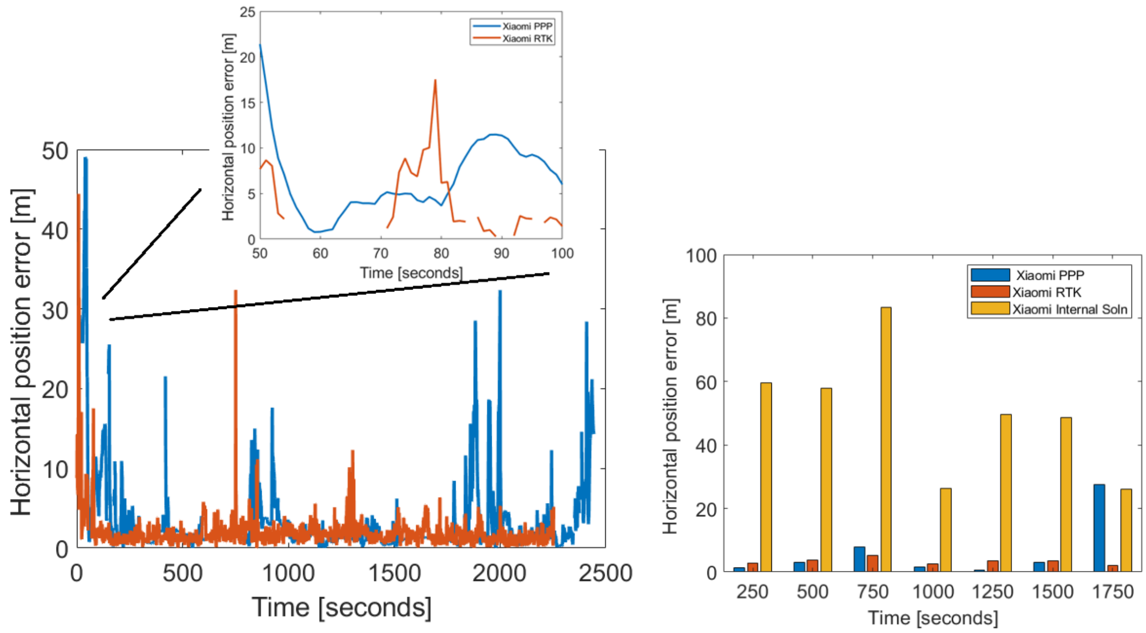

(c) Overall solution comparison

**Figure 6.7: Horizontal positioning plot for kinematic data (DOY 85, 2019)**

The internal solution had periodic deviation and higher positioning error as compared to the RTK and the PPP solutions as shown by the track in yellow. Moreover, the logged internal positioning solution is rounded off to three decimal places and an accurate comparison is not possible. The positioning solution was compared with the Piksi RTK solution. The rms and standard deviation of the horizontal positioning error are depicted in Table 6.2. The first few epochs were ignored since the car was parked in a high multipath environment (parking lot). The RTK solution had a 33% lesser rms error than the PPP solution but only had 92.2% solution availability as marked by the data gaps in Figure 6.8 (a) between epochs 50-100. The error for the three positioning solutions is also shown in Figure 6.8 (b) while 6.8 (a) compared the PPP solution versus the RTK solution. The internal solution performed the worst in the circular track, highlighted in Figure 6.7, due to frequent switching on and off the position tracking system. During this circular portion of the track, the PPP solution outperformed the RTK and the internal solution. The internal solution failed to follow the circular track while the rms of the RTK solution was 6.1 m compared to the PPP solution which had an rms of 5.6 m.

**Table 6.2: Comparison of the rms and std of the positioning error for PPP, RTK and internal positioning solution for the Xiaomi MI 8, (DOY 85, 2019)**

| Processing scenario Xiaomi MI 8 | Horizontal Positioning Error rms (m) | std(m) | Solution percentage |
|---|---|---|---|
| PPP | 6.8 | 5.2 | 2443/2443 = 100% |
| RTK | 4.6 | 2.8 | 2253/2443 = 92.2% |
| Internal solution | 42.3 | 16.2 | 2443/2443 = 100% |




(a) PPP vs RTK solution

(b) PPP vs RTK vs internal phone solution

**Figure 6.8: Comparison of the rms error for the PPP, RTK and the internal phone solution for the Xiaomi MI 8 (DOY 85, 2019)**

Overall, the PPP solution offered 100% availability and had 84% less rms error than the internal positioning solution. The PPP solution accuracy was comparable to the RTK solution and better than it in terms of availability. The PPP solution outperformed the RTK solution during circular portions of the track where there were more signal reflection and blockage. Overall, PPP with $C/N_0$-based stochastic modelling and prediction has higher positioning accuracy as compared to the internal solution.

# Chapter 7 Conclusions and Future Work

This chapter focuses on providing an overall summary of the smartphone-based PPP positioning by the YorkU PPP software. A summarization of the analysis and results is provided in the concluding section. Since there are several challenges with smartphone-based raw GNSS measurement collection and processing, a section focussing on these challenges is also discussed. The problems that have been addressed are listed. An insight into possible future work is mentioned.

## 7.1 Conclusions

Dual-frequency code and carrier-phase tracking smartphones made their entry into the market in 2018. Moreover, with the development of Google and other RINEX loggers, raw measurements from smartphones are now freely available to process. PPP in smartphones has achieved better solution quality and accuracy due to more available signals, better geometry of satellites, and the ability to extract continuous measurements.

The GNSS measurements obtained from smartphones were analysed in different multipath environments. It was observed that the $C/N_0$ of the smartphone L1 signal was 27% lower than the reference receiver in the static scenario, 23% lower in the kinematic medium multipath scenario and 24% lower in the high multipath scenario. The $C/N_0$ and elevation angle did not correlate for the smartphone measurements. As antennas in smartphones track signals from all sides, they are unable to distinguish between multipath and incoming signals and signals from higher elevation angle satellites also suffer from multipath. For the smartphone, the received L1 and E1 signals are considerably stronger than L5/E5a despite the L5/E5a signal being transmitted at higher power. The $C/N_0$ for GPS L1 is an average 3.1 dBHz stronger than L5 and the Galileo E1 is 3.4 dBHz stronger than E5a. Due to the inability of the linearly polarised smartphone antenna to distinguish between incoming and reflected signals, the smartphone pseudorange measurements suffer from high multipath noise. The multipath effect on the signals for Xiaomi MI 8 is roughly 84% higher than that of the effect on the SwiftNav Piksi in a static environment and 83% higher in the kinematic environment. The magnitude of multipath is directly correlated to the $C/N_0$ values. Due to frequent phase loss-of-lock, the carrier-phase measurements from the smartphone suffered from cycle slips. These cycle slips were followed by 3-5 seconds of missing measurements in low to medium multipath environments. It was observed that the L5 carrier-phase measurements were the most affected, followed by the L1 carrier-phase measurements and that the presence of all four measurements decreased from 90% in low multipath static environments to 40-55% in medium multipath kinematic environments and 25% in extremely high multipath scenarios. These data gaps were correlated to sudden drops in $C/N_0$ values. Data gaps were also observed in the logged Doppler measurements. It was observed for several satellites that for signals having a $C/N_0$ value less than 20 dBHz, the carrier-phase measurements were virtually absent. On the other hand, there was no such observation with a low elevation angle. Hence, the $C/N_0$ and elevation angle thresholds for PPP processing were empirically set at 20 dBHz and 10°, respectively.

The primary focus of the work was to condition and pre-process the measurements and to customise the PPP processing software to make it suitable for processing GNSS measurements from smartphones. The first step involved reading in the raw measurements available either in the RINEX format or the GNSS Logger format and assigning them to a data structure. To increase the positioning solution availability, a simple extrapolation-based prediction strategy was developed and implemented. Prediction was carried out before rejecting satellites with only single-frequency measurements present or with extremely large data gaps. The measurements were then checked to see if they satisfied the $C/N_0$ and elevation angle thresholds and subsequently rejected if they did not meet the requirements. Post all the rejections, measurement weighing was applied to the code and carrier-phase measurements. To improve the positioning solution accuracy, a realistic stochastic model was developed that assigned measurement weights based on the $C/N_0$ and multipath values. The process noise was also empirically set based on the dynamics of the moving receiver for a kinematic dataset and maintained at 0 for a static dataset. Finally, the measurements were processed in the sequential least squares filter after applying various corrections such as ionospheric, tropospheric, antenna offset, phase wind-up, orbital and clock corrections and correcting for cycle slips. The post-fit residuals for satellites are checked to see if they satisfy the post-fit residual rejection thresholds which have been empirically determined and the satellite is rejected if they not. The solution was checked for convergence based on the convergence threshold. If any residual did not pass the threshold limit, the corresponding observation is removed, and the solution was again computed until a threshold of the maximum number of iterations was reached.

The performance of the York-PPP software was examined using static and kinematic data collected from smartphones and compared with positioning results from other techniques and sources such as the internal positioning solution, RTK and SPP positioning performance. In static testing with the phones. the 2DRMS and 3DRMS positioning errors were 0.2 m and 0.3 m, respectively. The PPP solution is certainly an improvement from the Single Point Positioning solution which has an rms of 1.9 cm in the horizontal direction and 3.1 m in the vertical direction, in ideal open sky conditions. The $C/N_0$-based stochastic model offered better position accuracy than the elevation and static stochastic model when the PPP solution was computed for a static dataset where the phone was placed in the hand of a mannequin on the rooftop.

In the kinematic scenario, data were collected in suburban and urban canyon areas from within a vehicle. Using the prediction and $C/N_0$-based stochastic modelling technique, there was an average 64% decrease in the rms of the horizontal positioning error and a 1.3% increase in positioning solution availability There was an average 62% decrease in the positioning error and a 23% increase in positioning availability over regions where the signal was significantly impacted by multipath and blockage. With the help of prediction and the $C/N_0$-based stochastic modelling, the post-processed PPP solution had 100% availability. The PPP solution was 84% more accurate than the internal solution which suffered from biases due to non-constant tracking. The PPP solution had comparable accuracy with the RTK solution. These results promise significant improvement in the smartphone positioning solution using PPP augmentation.

### 7.1.1 Challenges Faced with Smartphone Data

Some of the challenges faced with smartphone GNSS data processing are the following:

1. **Availability of GPS signals in the L5 band –** Currently, there are about 14 satellites in the GPS constellation that transmit in the L5 band, out of which only 3-5 are tracked at any given time from different parts of the world.

2. **Incorrect computation of measurement values and missing data –** The interaction between the smartphone and the application logging the data is also crucial to understanding the quality of the logged measurements. There were several instances of incomplete data with missing measurements, that were not limited to just carrier-phase measurements. Certain phones showed multiple missing code measurements which cannot be explained by phase loss of lock, which is often used as a justification for missing carrier-phase measurements. There is a persistent problem with GLONASS satellites in all the data collected. The data not only follows the incorrect format, but the code and carrier-phase measurements are incorrectly computed.

3. **Heating up of phones –** Phones batteries suffer from discharge and excessive heating when data are collected for more than a few hours, sometimes even less than an hour depending on the outside temperature. As the antenna and chip heat up, the measurement logging and satellite tracking count are affected.

4. **Support for different applications affected by Android version –** With simultaneous Android and logger application updates, there were certain periods when certain applications were not supported on different phones. Other applications had to be used during the period which logged measurements incorrectly and thus, the datasets could not be processed.

5. **Antenna Quality and Description –** The problem of the poor quality linearly polarised antennas has been discussed in various sections regarding low and irregular $C/N_0$ and high measurement noise. There is no description regarding the placement of the antenna or the antenna phase centre, hence, corrections cannot be applied in this field. Moreover, the angle of the phone while collecting the data affected the measurement quality with different effects being observed on different phones.

### 7.1.2 Problems Addressed

1. **How do parameters such as $C/N_0$, multipath, measurement noise, cycle slips and data gaps vary with respect to each other and the environment?** The GNSS measurements obtained from smartphones were analysed in different multipath environments. Various realistic data collection scenarios were mimicked such as holding the phone in hand or placing it on the dashboard of the car. The measurements were analysed for measurement quality factors such as $C/N_0$, measurement noise, multipath and their interdependence. The duration and frequency of data gaps for the four measurements were also studied in different multipath environments and is novel to this research. Several anomalies were detected such as quality and availability of measurements being dependent on the $C/N_0$ value rather than the elevation angle, the L1/E1 measurements having higher received power than the L5/E5a measurements and missing code, carrier-phase and Doppler measurements.
2. **How does the prediction technique impact the availability and accuracy of a kinematic positioning solution in a suburban medium multipath environment? -**

Having a continuous solution is important to any navigation technique. Poor quality antenna coupled with signal blockages during realistic usage of smartphones brought about problems of missing measurements. With the prediction of missing measurements, nearly 100% solution availability was achieved in low to medium multipath environments with an average 50% increase in positioning accuracy.

3. **How does the $C/N_0$-based stochastic modelling technique impact accuracy of the static positioning solution and improve the availability and accuracy of a kinematic positioning solution in a low to medium multipath environment?** - *A priori* weighting for GNSS measurements from geodetic and other high-grade receivers is generally carried out either statically or based on the elevation angle of satellites since signals from higher elevation satellites face fewer blockages and minimal multipath which is generally not the case for the linearly polarized antenna of smartphones. The smartphone code measurements have poor multipath suppression. For such measurements, understanding the dependence of multipath noise and post-fit residuals on the $C/N_0$ was crucial to developing a model to assign *a priori* weights to the measurements. Based on the analysis and implementation, there was a considerable improvement in positioning solution, especially kinematic solutions where signals suffered from multipath and the antenna could not distinguish between the incoming and reflected signals.
4. **How does the PPP solution compare with the internal positioning solution and other positioning solutions such as SPP and RTK in different environments?** - The different positioning solutions comparisons are useful to analyse the improvement in positioning accuracy and availability with the help of PPP augmentation in smartphones. Data collected in static low multipath, kinematic medium multipath and kinematic high multipath

environments were processed in the PPP mode and the accuracy and availability of the positioning solution were compared against different techniques. Overall, the PPP solution offered 100% availability and had significantly less rms error than the internal positioning solution. Its accuracy was comparable to the RTK solution and better than it in terms of availability. The PPP solution outperformed the RTK solution during circular portions of the track where there were more signal reflection and blockage.

## 7.2 Future Work

PPP processing using dual-frequency code and carrier-phase measurements from smartphones is still in its preliminary stages, with the possibility of improvement in the near future. Some of the possible future work is listed below.

1. **Data collection and PPP processing using the BCM47765 chip** - The newly released second-generation, dual-frequency GNSS chip, the BCM47765, should be tested for positioning accuracy. With the ability to track the latest L5 signal from three constellations (GPS, Galileo and BeiDou), it will improve satellite availability in difficult environments.

2. **Multi-GNSS PPP processing by including GLONASS and BeiDou observations:** Future implementations of GNSS PPP in smartphones would require consistent logging of measurements, unlike the current applications that not only violate the RINEX 3.03 format for GLONASS measurements, but also log incorrect pseudorange and carrier-phase measurements. Moreover, current smartphone chips track only dual-frequency measurements from GPS, Galileo and QZSS constellations. Future implementation can

include improved versions of logging applications or the development of new applications that log measurements.

3. **Real-time PPP Processing**: Addition of the data processing options of the York-PPP engine to make it suitable to process real-time data from smartphones. Real-time precise satellite orbits and clock corrections would be required, and the smartphone measurements could be transmitted to a cloud where the processing would take place with these products. The positioning solution would subsequently be transferred to the smartphone.

4. **Addition of external antenna:** Smartphone antennas are not equipped to separate reflected and incoming signals affecting the measurement quality. Processing and testing data using a low-cost patch antenna with the smartphone can help overcome several challenges with smartphone positioning, improve positioning results in forested areas and urban canyons.

5. **Integration with other sensors:** Smartphones have self-contained navigation units with GNSS chips alongside accelerometers, gyroscopes and magnetometers. These IMU measurements can enhance the positioning accuracy of smartphones in the future, especially in obstructed environments when there are not enough satellites to generate a positioning solution. Integration with the camera to give a virtual reality-like experience is also a possibility.

6. **Resolving carrier-phase float ambiguities:** Ambiguity resolution is necessary for positioning results with better accuracy and has been implemented on geodetic receivers. However, it would be a challenge to implement it on low-cost hardware.

7. **Coupling single and dual-frequency PPP processing:** Smartphones with low-quality antennas do not track many satellites even if they are visible. For smartphones, however, there are certain cases where only L5 code and carrier-phase measurements are present. So future implementations should be dual-frequency coupled with single frequency L1 or L5 processing, based on whichever is present.

# References

Aboutaleb A, Ragab H, Nourledin A (2019) Examining the Benefits of LiDAR Odometry Integrated with GNSS and INS in Urban Areas. Miami, Florida, pp 3057–3065. https://doi.org/10.33012/2019.17010.

Aggrey J, Bisnath S, Naciri N, Shinghal G, Yang S (2020) Multi-GNSS precise point positioning with next-generation smartphone measurements. 65:79–98. https://doi.org/10.1080/14498596.2019.1664944

Aggrey JE (2015) Multi-GNSS precise point positioning software architecture and analysis of GLONASS pseudorange biases. MSc Thesis, York University

Android Developers (2016) Raw GNSS Measurements. In: Android Developers. https://developer.android.com/guide/topics/sensors/gnss. Accessed 17 Jul 2020

Bakker PF, Tiberius CC (2017) Real-time multi-GNSS single-frequency precise point positioning. GPS Solut 21:1791–1803. https://doi.org/10.1007/s10291-017-0653-2

Banville S (2014) Improved convergence for GNSS Precise Point Positioning. PhD Thesis, University of New Brunswick

Banville S, Lachapelle G, Ghoddousi-Fard R, Gratton P (2019) Automated Processing of Low-Cost GNSS Receiver Data. Miami, Florida, pp 3636–3652. https://doi.org/10.33012/2019.16972.

Banville S, Van Diggelen F (2016) Precise GNSS for Everyone: Precise Positioning Using Raw GPS Measurements from Android Smartphones. GPS World 27:43–48

Barbeau SJ (2018) Dual-frequency GNSS on Android devices. https://medium.com/@sjbarbeau/dual-frequency-gnss-on-android-devices-152b8826e1c. Accessed 17 Jul 2020

Bezmenov IV, Blinov IYu, Naumov AV, Pasynok SL (2019) An Algorithm for Cycle-Slip Detection in a Melbourne–Wübbena Combination Formed of Code and Carrier Phase GNSS Measurements. Meas Tech 62:415–421. https://doi.org/10.1007/s11018-019-01639-5

Bisnath SB, Gao Y (2009) Precise Point Positioning: A Powerful Technique with a Promising Future. GPS World 43–50

Bona P, Tiberius C (2000) An experimental comparison of noise characteristics of seven high-end dual frequency GPS receiver-sets. In: Position Location and Navigation Symposium, IEEE 2000. IEEE, pp 237–244

Braasch MS, van Dierendonck AJ (1999) GPS receiver architectures and measurements. Proceedings of the IEEE 87:48–64. https://doi.org/10.1109/5.736341

Canal Geomatics (2007) NovAtel GPS-702-GGL Dual-Frequency Pinwheel GNSS Antenna with L-Band. In: Canal Geomatics. http://www.canalgeomatics.com/product/novatel-gps-702-ggl-dual-frequency-pinwheel-gnss-antenna-with-l-band/. Accessed 17 Jul 2020

Chen B, Gao C, Liu Y, Sun P (2019) Real-time Precise Point Positioning with a Xiaomi MI 8 Android Smartphone. Sensors 19:2835. https://doi.org/10.3390/s19122835

Chen Z, Huang G (2016) Phase reduce false distance law and ionosphere residual error method detect and correct the cycle slip. In: 2016 Chinese Control and Decision Conference (CCDC). pp 1251–1255

Choy S (2009) An investigation into the accuracy of single frequency Precise Point Positioning (PPP). PhD Dissertation, RMIT University

Ciuban Sebastian, Krainski Mateusz, Perz Dominika, Burba Mareike (2020) GNSS Compare: Real-time Algorithms with Raw GNSS Measurement on Android Smartphones - Inside GNSS. https://insidegnss.com/compare/. Accessed 22 Apr 2020

Cozzens T (2020) Broadcom's second-generation dual-frequency GNSS uses new L5 signals. In: GPS World. https://www.gpsworld.com/broadcoms-second-generation-dual-frequency-gnss-uses-l5-signals/. Accessed 20 Jun 2020

Curran JT (2015) Enhancing Weak-Signal Carrier Phase Tracking in GNSS Receivers. In: International Journal of Navigation and Observation. https://www.hindawi.com/journals/ijno/2015/295029/. Accessed 6 May 2020

Dabove P, Di Pietra V (2019) Towards high accuracy GNSS real-time positioning with smartphones. Advances in Space Research 63:94–102. https://doi.org/10.1016/j.asr.2018.08.025

Diggelen F van, Khider M (2018) GPS measurement tools. In: Github. https://github.com/google/gps-measurement-tools/tree/master/GNSSLogger. Accessed 27 Apr 2019

EGNSSA (2017) Broadcom announces world's first dual frequency GNSS receiver. https://www.gsa.europa.eu/newsroom/news/broadcom-announces-world-s-first-dual-frequency-gnss-receiver-smartphones. Accessed 9 May 2020

EGSA (2018) World's First Dual-Frequency GNSS Smartphone Hits the Market. In: EUropean Global navigation Satellite Systems Agency. Accessed 22 Mar 2019

Elmezayen A, El-Rabbany A (2019) Precise Point Positioning Using World's First Dual-Frequency GPS/GALILEO Smartphone. Sensors (Basel) 19:. https://doi.org/10.3390/s19112593

Fan L, Wang L, Zhang M, Zheng Z (2015) A combination of MW and second-order time-difference phase ionospheric residual for cycle slip detection and repair. Wuhan Daxue Xuebao (Xinxi Kexue Ban)/Geomatics and Information Science of Wuhan University 40:790–794. https://doi.org/10.13203/j.whugis20130521

Fortunato M, Critchley-Marrows J, Siutkowska M, Ivanovici ML, Benedetti E, Roberts W (2019) Enabling High Accuracy Dynamic Applications in Urban Environments Using PPP and RTK on Android Multi-Frequency and Multi-GNSS Smartphones. In: 2019 European Navigation Conference (ENC). pp 1–9

Ge M, Gendt G, Rothacher M, Shi C, Liu J (2008) Resolution of GPS carrier-phase ambiguities in Precise Point Positioning (PPP) with daily observations. J Geod 82:389–399. https://doi.org/10.1007/s00190-007-0187-4

Geng, Jianghui, Li, Guangcai, Zeng, Ran, Wen, Qiang, Jiang, Enming, "A Comprehensive Assessment of Raw Multi-GNSS Measurements from Mainstream Portable Smart Devices," Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018), Miami, Florida, September 2018, pp. 392-412. https://doi.org/10.33012/2018.15965

Geo++ GmbH (2018) Logging Of GNSS Raw Data On Android. In: Geo++. http://www.geopp.de/logging-of-gnss-raw-data-on-android/. Accessed 27 Apr 2019

Georgiadou Y, Kleusberg A (1988) On carrier signal multipath effects in relative GPS positioning. Manuscripta Geodaetica 13:172–179

Gill M (2018) GNSS Precise Point Positioning using low-cost GNSS receivers. MSc Thesis, York University

Gill M, Bisnath S, Aggrey J, Seepersad G (2017) Precise Point Positioning (PPP) using Low-Cost and Ultra-Low-Cost GNSS Receivers. In: Proceeedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navgation (ION GNSS+ 2017). p 11

Gim J, Kwon-dong P (2017) Comparison of Positioning Accuracy Using the Pseudorange from Android GPS Raw Measurements. http://www.koreascience.or.kr/article/JAKO201732839399687.page. Accessed 21 Mar 2020

GNSS G (2016) PPP with Smartphones: Are We There Yet? In: BlackDot GNSS. http://www.blackdotgnss.com/2016/09/20/ppp-with-smartphones-are-we-there-yet/. Accessed 17 Jul 2020

GPS World (2018) Dual-band GNSS market moving to billions. In: GPS World. https://www.gpsworld.com/dual-band-gnss-market-moving-from-insignificant-to-billions-in-less-than-5-years/. Accessed 21 May 2020

Groves PD, Jiang Z, Rudi M, Strode P (2013) A Portfolio Approach to NLOS and Multipath Mitigation in Dense Urban Areas. https://www.semanticscholar.org/paper/A-Portfolio-Approach-to-NLOS-and-Multipath-in-Dense-Groves-Jiang/a9297d24a58a41d3759a20f8e863b05451343750. Accessed 29 Apr 2020

Guo L, Wang F, Sang J, Lin X, Gong X, Zhang W (2020) Characteristics Analysis of Raw Multi-GNSS Measurement from Xiaomi Mi 8 and Positioning Performance Improvement with L5/E5 Frequency in an Urban Environment. Remote Sensing 12:744. https://doi.org/10.3390/rs12040744

Hartinger H, Brunner FK (1998) Signal Distortion in High Precision Gps Surveys. Survey Review 34:531–541. https://doi.org/10.1179/sre.1998.34.270.531

Hofmann-Wellenhof B, Lichtenegger H, Wasle E (2007) GNSS–global navigation satellite systems: GPS, GLONASS, Galileo, and more. Springer Science & Business Media

Hui Hu, Lian Fang (2009) GPS cycle slip detection and correction based on high order difference and lagrange interpolation. In: 2009 2nd International Conference on Power Electronics and Intelligent Transportation System (PEITS). pp 384–387

Humphreys TE, Murrian M, van Diggelen F, Podshivalov S, Pesyna KM (2016) On the feasibility of cm-accurate positioning via a smartphone's antenna and GNSS chip. In: 2016 IEEE/ION Position, Location and Navigation Symposium (PLANS). pp 232–242

Kazmierski K, Hadas T, Sośnica K (2018) Weighting of Multi-GNSS Observations in Real-Time Precise Point Positioning. Remote Sensing 10:84. https://doi.org/10.3390/rs10010084

Ketola J (1999) GSM&GPS in the Same Package from Benefon - AfterDawn. https://www.afterdawn.com/news/article.cfm/1999/10/13/gsm_gps_in_the_same_package_from_benefon. Accessed 17 Jul 2020

Kouba J, Héroux P (2001) Precise point positioning using IGS orbit and clock products. GPS solutions 5:12–28

Lachapelle G, Cannon ME, Lu G (1992) High-precision GPS navigation with emphasis on carrier-phase ambiguity resolution. Marine Geodesy 15:253–269. https://doi.org/10.1080/01490419209388062

Lachapelle G, Gratton P, Horrelt J, Lemieux E, Broumandan A (2018) Evaluation of a Low Cost Hand Held Unit with GNSS Raw Data Capability and Comparison with an Android Smartphone. Sensors 18:4185. https://doi.org/10.3390/s18124185

Landau H, Chen X, Klose S, Leandro R, Vollath U (2009) Trimble's RTK and DGPS solutions in comparison with precise point positioning. In: Observing our Changing Earth. Springer, pp 709–718. https://doi.org/ 10.1007/978-3-540-85426-5_81.

Langley RB (1996) GPS receivers and the observables. In: Kleusberg A, Teunissen PJG (eds) GPS for Geodesy. Springer, Berlin, Heidelberg, pp 141–173

Laurichesse D, Blot A (2016) Fast PPP Convergence Using Multi-Constellation and Triple-Frequency Ambiguity Resolution. In: Proceedings of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2016). pp 2082–2088

Leclère J, Landry R, Botteron C (2018) Comparison of L1 and L5 Bands GNSS Signals Acquisition. Sensors (Basel) 18:. https://doi.org/10.3390/s18092779

Li G, Geng J (2019) Characteristics of raw multi-GNSS measurement error from Google Android smart devices. GPS Solut 23:1–16. https://doi.org/10.1007/s10291-019-0885-4

Li Z, Zhang T, Qi F, Tang H, Niu X (2019) Carrier phase prediction method for GNSS precise positioning in challenging environment. Advances in Space Research 63:2164–2174. https://doi.org/10.1016/j.asr.2018.12.015

Manandhar D (2017) Data Formats: NMEA, RINEX. The University of Tokyo 14

Marçal J, Nunes F (2016) Robust vector tracking for GNSS carrier phase signals. In: 2016 International Conference on Localization and GNSS (ICL-GNSS). pp 1–6

Misra P, Enge P (2006) Global Positioning System: Signals, Measurements and Performance, 2nd edn. Ganga-Jamuna Press, Massachusetts

Montenbruck O, Steigenberger P, Schönemann E, Hauschild A, Hugentobler U, Dach R, Becker M (2012) Flight Characterization of New Generation GNSS Satellite Clocks. NAVIGATION 59:291–302. https://doi.org/10.1002/navi.22

Paziewski J (2020) Recent advances and perspectives for positioning and applications with smartphone GNSS observations. Meas Sci Technol 31:091001. https://doi.org/10.1088/1361-6501/ab8a7d

Paziewski J, Sieradzki R, Baryla R (2019) Signal characterization and assessment of code GNSS positioning with low-power consumption smartphones. GPS Solut 23:98. https://doi.org/10.1007/s10291-019-0892-5

Pesyna KM, Heath Rober W. (2013) Centimeter Positioning with a Smartphone-Quality GNSS Antenna

Pesyna KM, Heath RW, Humphreys TE (2014) Centimeter positioning with a smartphone-quality GNSS antenna. In: Proceedings of the ION GNSS+ Meeting. Tampa, FL, pp 1568–1577

Pirsiavash A, Broumandan A, Lachapelle G (2017) Characterization of Signal Quality Monitoring Techniques for Multipath Detection in GNSS Applications. Sensors (Basel) 17:. https://doi.org/10.3390/s17071579

Robustelli U, Baiocchi V, Pugliano G (2019) Assessment of Dual Frequency GNSS Observations from a Xiaomi Mi 8 Android Smartphone and Positioning Performance Analysis. Electronics 8:91. https://doi.org/10.3390/electronics8010091

Seepersad G (2012) Reduction of initial convergence period in GPS PPP data processing. MSc Thesis, York University

Sennott JW (1999) Receiver Architectures for Improved Carrier Phase Tracking in Atenuation, Blockage, and Interference. GPS Solutions 3:40–47. https://doi.org/10.1007/PL00012790

Sikirica N, Malić E, Rumora I, Filjar R (2017) Exploitation of google GNSS measurement API for risk assessment of GNSS applications. In: 2017 25th Telecommunication Forum (TELFOR). pp 1–3

Silva P (2012) Cycle Slip Detection and Correction for Precise Point Positioning

Thoelert S, Furthner J, Meurer M (2013) Signal Quality of Galileo, BeiDou. In: GPS World. https://www.gpsworld.com/signal-quality-of-galileo-beidou/. Accessed 26 Jul 2020

van Diggelen F (2009) The Smartphone Revolution. In: GPS World. https://www.gpsworld.com/wirelesssmartphone-revolution-9183/. Accessed 17 Jul 2020

Wang L, Groves PD, Ziebart MK (2012) GNSS Shadow Matching: Improving Urban Positioning Accuracy Using a 3D City Model with Optimized Visibility Prediction Scoring. 15

Wanninger L, Heßelbarth A (2020) GNSS code and carrier phase observations of a Huawei P30 smartphone: quality assessment and centimeter-accurate positioning. GPS Solut 24:64. https://doi.org/10.1007/s10291-020-00978-z

Wells D (ed) (1987) Guide to GPS positioning, 2. Canadian GPS Associates, Fredericton, New Brunswick, Canada

Wu Q, Sun M, Zhou C, Zhang P (2019) Precise Point Positioning Using Dual-Frequency GNSS Observations on Smartphone. Sensors 19:2189. https://doi.org/10.3390/s19092189

Yang Y, Sharpe RT, Hatch RR (2003) L1 Backup Navigation for Dual Frequency GPS Receiver. 6. Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003), Portland, OR, September 2003, pp. 1258-1263.

Yoon D, Kee C, Seo J, Park B (2016) Position accuracy improvement by implementing the DGNSS-CP algorithm in smartphones. Sensors 16:910. https://doi.org/10.3390/s16060910.

Zhang X, Tao X, Zhu F, Shi X, Wang F(2018) Quality assessment of GNSS observations from an Android N smartphone and positioning performance analysis using time-differenced filtering approach. GPS Solutions 22:. https://doi.org/10.1007/s10291-018-0736-8