



Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

---

Reihe C

Dissertationen

Heft Nr. 864

**Francesco Darugna**

**Improving Smartphone-Based GNSS Positioning  
Using State Space Augmentation Techniques**

**München 2021**

**Verlag der Bayerischen Akademie der Wissenschaften**

**ISSN 0065-5325**

**ISBN 978-3-7696-5276-5**

---

Diese Arbeit ist gleichzeitig veröffentlicht in:  
Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Universität Hannover  
ISSN 0174-1454, Nr. 368, Hannover 2021





Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

---

Reihe C

Dissertationen

Heft Nr. 864

## Improving Smartphone-Based GNSS Positioning Using State Space Augmentation Techniques

Von der Fakultät für Bauingenieurwesen und Geodäsie  
der Gottfried Wilhelm Leibniz Universität Hannover  
zur Erlangung des Grades  
Doktor-Ingenieur (Dr.-Ing.)  
genehmigte Dissertation

Vorgelegt von

M. Sc. Francesco Darugna

Geboren am 21.01.1992 in Feltre, Italy

München 2021

Verlag der Bayerischen Akademie der Wissenschaften

ISSN 0065-5325

ISBN 978-3-7696-5276-5

---

Diese Arbeit ist gleichzeitig veröffentlicht in:  
Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Universität Hannover  
ISSN 0174-1454, Nr. 368, Hannover 2021

## Adresse der DGK:



### Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften (DGK)

Alfons-Goppel-Straße 11 • D – 80 539 München  
Telefon +49 – 331 – 288 1685 • Telefax +49 – 331 – 288 1759  
E-Mail [post@dgk.badw.de](mailto:post@dgk.badw.de) • <http://www.dgk.badw.de>

#### Prüfungskommission:

Vorsitzender: Prof. Dr.-Ing. habil. Christian Heipke

Referent: Prof. Dr.-Ing. Steffen Schön

Korreferenten: Prof. Dr. Fabio Dosis (Politecnico di Torino)  
Prof. Dr.-Ing. habil. Monika Sester

Tag der mündlichen Prüfung: 18.11.2020

---

© 2021 Bayerische Akademie der Wissenschaften, München

Alle Rechte vorbehalten. Ohne Genehmigung der Herausgeber ist es auch nicht gestattet,  
die Veröffentlichung oder Teile daraus auf photomechanischem Wege (Photokopie, Mikrokopie) zu vervielfältigen

## ABSTRACT

---

Low-cost receivers providing Global Navigation Satellite System (GNSS) pseudorange and carrier phase raw measurements for multiple frequencies and multiple GNSS constellations have become available on the market in the last years. This significantly has increased the number of devices equipped with the necessary sensors to perform precise GNSS positioning. GNSS pseudorange and carrier phase are used to compute user positions. While both observations are affected by different error sources, e.g. the passage through the atmosphere, only the carrier-phase has an ambiguous nature. The resolution of this ambiguity is a crucial factor to reach fast and highly precise GNSS-based positioning.

Currently, several smartphones are equipped with a dual-frequency, multi-constellation receiver. The access to Android-based GNSS raw measurements has become a strong motivation to investigate the feasibility of smartphone-based high-accuracy positioning. The quality of smartphone GNSS measurements has been analyzed, suggesting that they often suffer from low signal-to-noise, inhomogeneous antenna gain and high levels of multipath. This work shows how to tackle several of the currently present obstacles and demonstrates centimeter-level positioning with a low-cost GNSS antenna and a low-cost GNSS receiver built into an off-the-shelf smartphone.

Since the beginning of the research in smartphone-based positioning, the device's GNSS antenna has been recognized as one of the main limitations. Besides Multipath (MP), the antenna radiation pattern is the main site-dependent error source of GNSS observations. An absolute antenna calibration has been performed for the dual-frequency smartphone Huawei Mate20X. Antenna Phase Center Offset (PCO), and Variations (PCV) have been estimated to correct for the antenna impact on the L1 and L5 phase observations. Accordingly, the relevance of considering the individual PCO and PCV for the two frequencies is shown. The PCV patterns indicate absolute values up to 2 cm and 4 cm for L1 and L5, respectively. The impact of antenna corrections has been assessed in different multipath environments using a high-accuracy positioning algorithm employing an uncombined observation model and applying Ambiguity Resolution (AR). Experiments both in zero-baseline and short-baseline configurations have been performed. Instantaneous AR in the zero-baseline setup has been demonstrated, showing the potential for cm-level positioning with low-cost sensors available inside smartphones. In short-baselines configurations, no reliable AR is achieved without antenna corrections. However, after correcting for PCV, successful AR is demonstrated for a smartphone placed in a low multipath environment on the ground of a soccer field. For a rooftop open-sky test case with large multipath, AR was successful in 19 out of 35 data-sets. Overall, the antenna calibration is demonstrated being an asset for smartphone-based positioning with AR, showing cm-level 2D Root Mean Square Error (RMSE).

In GNSS-based positioning, a user within a region covered by a network of reference stations can take advantage of the network-estimated augmentation parameters. Among the GNSS error sources, atmospheric delays have a strong impact on the positioning performance and the ability to resolve ambiguities. State Space Representation (SSR) atmospheric corrections, i.e. tropospheric and ionospheric delays, are commonly estimated for the approximate user position by interpolation from values calculated for the reference stations. Widely used interpolation techniques are Inverse Distance Weighted (IDW), Ordinary Kriging (OK) and Weighted Least Squares (WLS). The interpolation quality of such techniques during severe weather events and Traveling Ionospheric Disturbances (TIDs) is analyzed. To improve the interpolation performance during such events, modified WLS methods taking advantage

---

of the physical atmospheric behavior are proposed. To support this interpolation approach, external information from Numerical Weather Models (NWM) for tropospheric interpolation and from TID modeling for ionospheric interpolation is introduced to the algorithms.

The interpolation is assessed using simulated data (considering artificial and real network geometries), and real SSR parameters generated by network computation of GNSS measurements. As examples, two severe weather events in northern Europe in 2017 and one TID event over Japan in 2019 have been analyzed. The interpolation of SSR Zenith Tropospheric Delay (ZTD) and ionospheric parameters is evaluated. Considering the reference station positions as rover locations, the modified WLS approach marks a lower RMSE in up to 80% of the cases during sharp weather fluctuations. Also, the average error can be decreased in 64% of the cases during the TID event investigated. Improvements up to factors larger than two are observed. Furthermore, specific cases are isolated, showing particular ZTD variations where significant errors (e.g. larger than 1 cm) can be reduced by up to 20% of the total amount. As a final product of the analysis, tropospheric and ionospheric messages are proposed. The messages contain the information needed to implement the suggested interpolation.

Along with the need for accurate atmospheric models, the concept of consistency in the SSR corrections is crucial. A format that can transport all the SSR corrections estimated by a network is the Geo++ SSR format (SSRZ). Exploiting the features of the SSRZ format, the impact of an error in the transported ionospheric parameters is investigated. It is shown that the position estimation strongly depends on the ionospheric modeling and mismodeling can result in cm level errors, especially in the height component.

**Keywords:** *SSR, GNSS smartphone-based positioning, GNSS antenna calibration, atmospheric interpolation*

---

## KURZFASSUNG

---

In den letzten Jahren sind kostengünstige Empfänger auf dem Markt verfügbar geworden, die Pseudorange- und Trägerphasenrohmessungen für mehrere Frequenzen und mehrere Konstellationen der Global Navigation Satellite Systems (GNSS) ermöglichen. Dies erhöht die Anzahl der Geräte, die mit den erforderlichen Sensoren ausgestattet sind, um eine präzise GNSS-Positionierung durchzuführen, erheblich. GNSS-Pseudorange- und Trägerphasenbeobachtungen werden verwendet, um Benutzerpositionen zu berechnen. Während beide Beobachtungen von unterschiedlichen Fehlerquellen beeinflusst werden, z.B. beim Durchgang durch die Atmosphäre, ist nur die der Trägerphase mehrdeutig. Die Auflösung dieser Mehrdeutigkeit ist ein entscheidender Faktor für eine schnelle und hochpräzise GNSS-basierte Positionierung. Derzeit sind einige Smartphones mit einem Zwei-Frequenz-Multi-GNSS-Empfänger ausgestattet. Der Zugriff auf Android-basierte GNSS-Rohmessungen ist zu einer starken Motivation geworden, die Smartphone-basierte Positionierung zu untersuchen.

Die Ergebnisse der Qualitätsanalyse von GNSS-Messungen mit Smartphones deuten darauf hin, dass Smartphone-Beobachtungen häufig durch ein geringes Signal-Rausch-Verhältnis, einer inhomogenen Antennencharakteristik und einem hohen Mehrwegepegel (Multipath-Effekte) beeinträchtigt sind. Diese Arbeit zeigt, wie mehrere der derzeit vorhandenen Hindernisse überwunden werden können und die Positionierung mit Zentimetergenauigkeit mit einer kostengünstigen GNSS-Antenne und einem kostengünstigen GNSS-Empfänger in einem handelsüblichen Smartphone erreicht wird. Schon zu Beginn der Forschung wurde die GNSS-Antenne des Smartphone als eine der größten Fehlerquellen in der Positionierung erkannt, wobei neben Multipath (MP) die Empfangscharakteristik der Antenne die wichtigste ortsabhängige Fehlerquelle für GNSS-Beobachtungen darstellt. In dieser Arbeit werden Ergebnisse der absoluten Antennenkalibrierung des Zwei-Frequenz-Huawei Mate20X vorgestellt. Für das Zwei-Frequenz-Huawei Mate20X wurde eine absolute Antennenkalibrierung durchgeführt und der Phase-Center-Offset (PCO) und die Phase-Center-Variations (PCV) geschätzt, um den Einfluss der Antenne auf die Phasenbeobachtungen von L1 und L5 zu korrigieren. Die Relevanz dieser beiden Korrekturen wird für die beiden Frequenzen gezeigt. Die PCV-Werte variieren um bis zu 2cm für L1 und um bis zu 4cm für L5. Die Auswirkungen von Antennenkorrekturen wurden in verschiedenen Mehrwegeumgebungen mithilfe eines hochgenauen Positionierungsalgorithmus unter Verwendung der Mehrdeutigkeitslösung (Ambiguity Resolution AR) für die zwei Phasenbeobachtungen untersucht.

Es wurden Experimente sowohl in Nullbasislinien- als auch in Kurzbasislinien-Konfigurationen durchgeführt. Dabei konnte eine sofortige Mehrdeutigkeitsauflösung im Nullbasislinien-Setup demonstriert werden, die das Potenzial für die Positionierung auf Zentimetergenauigkeit mit kostengünstigen Smartphone-Sensoren zeigt. In Konfigurationen mit kurzen Basislinien wird ohne Antennenkorrekturen keine zuverlässige Mehrdeutigkeitsauflösung erreicht. Mit der Anbringung der PCV-Korrektur an die Smartphone-Antenne wird jedoch eine erfolgreiche Auflösung für ein Smartphone demonstriert, das sich in einer Umgebung mit geringem Mehrwegeeffekt auf dem Boden eines Fußballfelds befindet. Für einen Open-Sky-Testfall auf dem Dach mit starkem Mehrwegeeffekt war die Mehrdeutigkeitsauflösung in 19 von 35 Datensätzen erfolgreich. Diese Analysen zeigen, dass die Antennenkalibrierung ein Vorteil für die Smartphone-basierte Positionierung mit Mehrdeutigkeitsauflösung ist und der 2D RMS-Fehler (RMSE) im Zentimeterbereich erreicht werden kann. Die Mehrdeutigkeitsauflösung und damit die Positionierungsperformance werden stark durch atmosphärische Effekte beeinflusst. Befindet sich ein GNSS-Nutzer innerhalb einer Region, die von einem

---

Referenzstationsnetzwerk abgedeckt wird, kann dieser die mit dem Netzwerk ermittelten Korrekturdaten nutzen. Die State Space Representation (SSR) stellt u.a. atmosphärische Korrekturen bereit, die der Nutzer für seine ungefähre Position individualisieren kann. Für die Individualisierung werden verschiedene Interpolationsmethoden benutzt. Weit verbreitete Interpolationstechniken sind die Inverse Distance Weighted (IDW), Ordinary Kriging (OK) und Weighted Least Squares (WLS) Methode. Die Qualität dieser Interpolationsmethoden bei Unwetterereignissen und Traveling Ionospheric Disturbances (TIDs) wird analysiert. Um die Interpolationsleistung während solcher Ereignisse zu verbessern, werden modifizierte WLS-Verfahren vorgeschlagen, die das physikalische Atmosphärenverhalten ausnutzen. Um diesen Interpolationsansatz zu unterstützen, werden externe Informationen aus numerischen Wettermodellen (NWM) für die troposphärische Interpolation und aus der TID-Modellierung für die ionosphärische Interpolation in die Algorithmen eingeführt. Die Bewertung der Interpolationsverfahren basiert einerseits auf der Verwendung simulierter Daten in künstlichen und realen Netzwerkgeometrien und andererseits auf der Verwendung realer SSR-Parameter, die während zweier Unwetterereignisse in Nordeuropa im Jahr 2017 und einem TID-Ereignis über Japan im Jahr 2019 erzeugt wurden. Die Interpolation der SSR-Zenit-Troposphärenverzögerung (ZTD) und der ionosphärischen Parameter werden analysiert. Unter Verwendung einer Referenzstation als Rover zeigt der modifizierte WLS-Ansatz in bis zu 80 % der Fälle bei starken Wetterschwankungen einen niedrigeren RMSE. Darüber hinaus kann der durchschnittliche Fehler in 64 % der Fälle während des untersuchten TID-Ereignisses verringert werden. Verbesserungen bis zu einem Faktor größer als zwei werden beobachtet. Darüber hinaus werden solche Fälle isoliert, die bestimmte ZTD-Variationen zeigen, bei denen signifikante Fehler (z.B. größer als 1 cm) auf bis zu 20 % der Gesamtmenge reduziert werden können. Für die Anwendung der modifizierten Interpolationen in der Praxis werden Nachrichten(formate) vorgeschlagen, die die erforderlichen Informationen für die ordnungsgemäße Implementierung der vorgeschlagenen ionosphärischen und troposphärischen Interpolationen beinhalten. Neben der Notwendigkeit genauer atmosphärischer Modelle ist das Konzept der Konsistenz bei den SSR-Korrekturen von entscheidender Bedeutung. Ein Format, das alle von einem Netzwerk geschätzten SSR-Korrekturen transportieren kann, ist das Geo++ SSR-Format SSRZ. Unter Ausnutzung der Merkmale des SSRZ-Formats wird die Auswirkung eines Fehlers in den transportierten ionosphärischen Parametern untersucht. Es wird gezeigt, dass die Positionsschätzung stark von der ionosphärischen Modellierung abhängt und eine Fehlmodellierung zu Fehlern auf Zentimeterniveau insbesondere in der Höhenkomponente führen können.

**Schlagwörter:** *SSR, GNSS-Smartphone-basierte Positionierung, GNSS-Antennenkalibrierung, atmosphärische Interpolation*

---

# ACRONYMS

---

<b>A-GNSS</b>	Assisted-GNSS
<b>ADR</b>	Accumulated Delta Range
<b>AR</b>	Ambiguity Resolution
<b>BDS</b>	BeiDou Navigation Satellite System
<b>BKG</b>	Federal Agency for Cartography and Geodesy
<b>BN</b>	BiasNanos
<b>C/N0</b>	Carrier to Noise density ratio
<b>CNES</b>	National Centre for Space Studiess
<b>CORS</b>	Continuously Operation Reference Stations
<b>CT</b>	Clough Tocher
<b>DD</b>	Double Difference
<b>DLL</b>	Delay Lock Loop
<b>DLR</b>	German Aerospace Center
<b>ECEF</b>	Earth-Centered Earth-Fixed
<b>ECMWF</b>	European Center for Medium-Range Weather Forecasts
<b>ESA</b>	European Space Agency
<b>ETH</b>	Eidgenössische Technische Hochschule
<b>FBN</b>	FullBiasNanos
<b>FKP</b>	Flaechen Korrektur Parameter
<b>GDOP</b>	Geometric Dilution of Precision
<b>GDV</b>	Group Delay Variations
<b>GEONET</b>	GNSS Earth Observation Network System
<b>GFS</b>	Global Forecasting System
<b>GFZ</b>	German Research Centre for Geosciences
<b>GIM</b>	Global Ionospheric propagation Model
<b>GLONASS</b>	GLOBal'naya Navigatsionnaya Sputnikovaya Sistema
<b>GNSMART</b>	GNSS State Monitoring and Representation Technique
<b>GNSS</b>	Global Navigation Satellite System

---

<b>GPO</b>	Ground Point Origin
<b>GPS</b>	Global Positioning System
<b>GREF</b>	Integrated Geodetic Reference Network of Germany
<b>GT</b>	GPS Time
<b>GVI</b>	Global Vertical Ionosphere
<b>GRI</b>	Gridded Ionosphere
<b>GRT</b>	Gridded Troposphere
<b>GSI</b>	Global Satellite-dependent Ionosphere
<b>HDOP</b>	Horizontal Dilution of Precision
<b>IDW</b>	Inverse Distance Weighted
<b>IF</b>	Intermediate Frequency
<b>IGS</b>	International GNSS Service
<b>IGS RTS</b>	IGS Real-Time Service
<b>IPA</b>	Initial Phase Ambiguity
<b>IPB</b>	Initial Phase Bias
<b>IPP</b>	Ionospheric Pierce Point
<b>IRI</b>	International Reference Ionosphere
<b>IRNSS</b>	Indian Regional Navigation Satellite System
<b>ISB</b>	Inter-System Biases
<b>ITRF</b>	International Terrestrial Reference Frame
<b>JPL</b>	Jet Propulsion Laboratory
<b>LAMBDA</b>	Least-squares AMBiguity Decorrelation Adjustment
<b>LGLN-SAPOS</b>	Landesamt für Geoinformation und Landesvermessung Niedersachsen - Satellitenpositionierungsdienst
<b>LS</b>	Least Squares
<b>OSR</b>	Observation Space Representation
<b>MAC</b>	Master-Auxiliary-Concept
<b>Mate20X</b>	Huawei Mate20X
<b>Mi8</b>	Xiaomi Mi8
<b>MP</b>	Multipath
<b>MSTID</b>	Medium-Scale TID
<b>NCO</b>	Numerically Controlled Oscillator

---

<b>NETPOS</b>	Netherlands Positioning Service
<b>NRCan</b>	Natural Resources Canada
<b>NRP</b>	North Reference Point
<b>N-RTK</b>	Network-RTK
<b>NTRIP</b>	Networked Transport of RTCM via Internet Protocol
<b>OK</b>	Ordinary Kriging
<b>PC</b>	electrical mean Phase Center
<b>PCB</b>	Printed Circuit Board
<b>PCC</b>	Phase Center Corrections
<b>PCO</b>	Phase Center Offset
<b>PCV</b>	Phase Center Variations
<b>PDOP</b>	Position Dilution of Precision
<b>PLL</b>	Phase Lock Loop
<b>POD</b>	Precise Orbit Determination
<b>PPO</b>	Pierce Point Origin
<b>PPP</b>	Precise Point Positioning
<b>PPP-RTK</b>	Precise Point Positioning - Real-Time Kinematic
<b>QZSS</b>	Quasi Zenith Satellite System
<b>RF</b>	Radio Frequency
<b>RIU</b>	Residual Interpolation Uncertainty
<b>RMSE</b>	Root Mean Square Error
<b>RSI</b>	Regional Satellite-dependent Ionosphere
<b>RT</b>	Regional Troposphere
<b>RTCM</b>	Radio Technical Commission for Maritime Services
<b>RTK</b>	Real-Time Kinematic
<b>SC</b>	Special Committee
<b>SMA</b>	SubMiniature version A
<b>SoC</b>	System on Chip
<b>SPP</b>	Single Point Positioning
<b>SSM</b>	State Space Modeling
<b>SSR</b>	State Space Representation
<b>STD</b>	Standard deviation

---

<b>STEC</b>	Slant Total Electron Content
<b>TEC</b>	Total Electron Content
<b>TECU</b>	Total Electron Content Unit
<b>TID</b>	Traveling Ionospheric Disturbance
<b>TOD</b>	Time Of the Day
<b>TOW</b>	Time Of the Week
<b>TREASURE</b>	Training REsearch and Applications Network to Support the Ultimate Real-Time High Accuracy EGNSS Solution
<b>TTFA</b>	Time To Fix Ambiguities
<b>TISM</b>	Time To achieve Sub-Meter
<b>VDOP</b>	Vertical Dilution of Precision
<b>VGMF</b>	Vienna Global Mapping Functions
<b>VTEC</b>	Vertical Total Electron Content
<b>WLS</b>	Weighted Least Squares
<b>WGS84</b>	World Geodetic System 1984
<b>WHU</b>	Wuhan University
<b>ZHD</b>	Zenith Hydrostatic Delay
<b>ZTD</b>	Zenith Tropospheric Delay
<b>ZWD</b>	Zenith Wet Delay

# CONTENTS

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>GNSS Positioning Techniques</b>	<b>7</b>
2.1	The GNSS observation equations . . . . .	7
2.1.1	Pseudorange measurements . . . . .	8
2.1.2	Carrier phase measurements . . . . .	9
2.1.3	Atmospheric delays . . . . .	9
2.1.4	Double-Difference (DD) observations . . . . .	12
2.2	GNSS augmentation techniques . . . . .	14
2.2.1	State Space Modeling (SSM) . . . . .	14
2.2.2	State Space Representation (SSR) . . . . .	15
2.2.3	SSR corrections . . . . .	16
2.2.4	Observation Space Representation (OSR) . . . . .	21
2.3	RTK positioning . . . . .	22
2.4	Positioning with uncombined observation model . . . . .	23
<b>3</b>	<b>Smartphone-Based Positioning</b>	<b>25</b>
3.1	Motivation . . . . .	25
3.2	The construction of the observations . . . . .	26
3.2.1	Pseudorange calculation . . . . .	27
3.2.2	Phase observation computation . . . . .	29
3.2.3	Continuity of phase measurements . . . . .	30
3.3	Quality analysis of smartphone measurements . . . . .	32
3.3.1	Setup . . . . .	32
3.3.2	Phase double-difference analysis . . . . .	34
3.3.3	Code noise and multipath investigation . . . . .	44
3.3.4	Allan deviation analysis . . . . .	49
3.3.5	Expected positioning performance . . . . .	50
3.3.6	Smartphone-based ionospheric TEC measurements . . . . .	51
3.4	Positioning using smartphones . . . . .	52
3.4.1	Zero and short-baselines applications . . . . .	53
3.4.2	Network SSR-based positioning . . . . .	58
3.4.3	SPP and PPP using smartphone's observations . . . . .	60
3.5	Discussion . . . . .	60
<b>4</b>	<b>PCV Impact on Smartphone-Based Positioning</b>	<b>63</b>
4.1	Motivation . . . . .	63
4.2	Absolute robot-based field calibration of GNSS antennas . . . . .	64
4.3	The Huawei Mate20X antenna calibration . . . . .	66
4.3.1	Analysis of the Mate20X's PCV . . . . .	66
4.3.2	PCV repeatability . . . . .	71
4.3.3	Smartphone antenna location . . . . .	75
4.4	High-accuracy smartphone-based positioning . . . . .	78
4.4.1	Positioning in Scenario 4: the soccer field test . . . . .	79
4.4.2	Positioning in Scenario 3: the rooftop test . . . . .	84

---

4.5	Discussion . . . . .	89
<b>5</b>	<b>Mitigating the Impact of SSR Atmospheric Parameter Errors</b>	<b>91</b>
5.1	Motivation . . . . .	91
5.2	Expected impact on positioning . . . . .	93
5.3	Interpolation techniques . . . . .	94
5.3.1	Interpolation of scattered data . . . . .	94
5.3.2	Methods introduction . . . . .	95
5.3.3	Modified WLS methods . . . . .	97
5.4	Interpolation of simulated data . . . . .	98
5.4.1	Simulation setup . . . . .	98
5.4.2	Generation of error patterns . . . . .	100
5.4.3	Results for artificial grids . . . . .	100
5.4.4	Results for real-network geometries . . . . .	102
5.4.5	Required precision for the directional information . . . . .	104
5.4.6	Variation of the interpolation position . . . . .	105
5.4.7	Variation of the distance: WLS methods comparison . . . . .	106
5.5	ZTD interpolation during severe weather events . . . . .	108
5.5.1	Data-sets and analysis concept . . . . .	108
5.5.2	Comparison of NWM gradients and spatial distribution of GNSS-estimated ZTD . . . . .	112
5.5.3	Interpolation results using directional techniques . . . . .	113
5.5.4	Impact of test location . . . . .	115
5.6	Interpolation of residual ionospheric effects during a TID . . . . .	117
5.6.1	Data-set and extraction of TID event . . . . .	117
5.6.2	Improved interpolation strategy . . . . .	119
5.7	Error propagation: from SSR modeling to the user . . . . .	121
5.7.1	Ambiguity resolution with ionospheric biases: simulation . . . . .	121
5.7.2	Real-time impact on positioning of an ionospheric SSR error . . . . .	122
5.8	Discussion . . . . .	125
<b>6</b>	<b>Conclusions and outlook</b>	<b>129</b>
<b>A</b>	<b>RTCM-SSR Python demo</b>	<b>131</b>
<b>B</b>	<b>SSRZ Python demo</b>	<b>141</b>
<b>C</b>	<b>Smartphone-based positioning: additional information</b>	<b>149</b>
C.1	Phase DD in Scenario 1: specific satellite examples . . . . .	149
C.2	Smartphone-based positioning using GNSMART . . . . .	149
C.3	GNSS raw measurements: Android API classes . . . . .	150

# LIST OF FIGURES

---

1.1	Representation of range and SSR corrections . . . . .	2
1.2	Outline of the thesis . . . . .	4
3.1	Time reconstruction from GNSS raw measurements: GPS example . . . . .	29
3.2	Comparison of ADR computed from Android API with constant and updated FBN	31
3.3	Geo++ GmbH rooftop and zero-baseline setup between smartphone and geode- tic receiver. . . . .	33
3.4	Smartphone on soccer field experiment setup. . . . .	34
3.5	Phase DD in zero-baseline setup with smartphone in RF enclosure . . . . .	36
3.6	GPS C/N0 and elevation of smartphone measurements in zero-baseline within RF enclosure . . . . .	36
3.7	Galileo C/N0 and elevation of smartphone measurements in zero-baseline within RF enclosure . . . . .	37
3.8	Example of phase DD jump in smartphone's measurements . . . . .	37
3.9	Phase DD in zero-baseline setup with smartphone in RF enclosure with 13dB attenuation . . . . .	38
3.10	GPS C/N0 and elevation of smartphone measurements in zero-baseline within RF enclosure with 13 dB attenuation . . . . .	39
3.11	Galileo C/N0 and elevation of smartphone measurements in zero-baseline within RF enclosure with 13 dB attenuation . . . . .	39
3.12	Phase DD in open sky rooftop . . . . .	40
3.13	GPS C/N0 and elevation of smartphone measurements in an open sky rooftop .	41
3.14	Galileo C/N0 and elevation of smartphone measurements in open sky rooftop .	41
3.15	Mean of phase DD absolute values in open sky rooftop . . . . .	42
3.16	Phase DD in the soccer field . . . . .	42
3.17	GPS C/N0 and elevation of smartphone measurements in the soccer field . . . .	43
3.18	Galileo C/N0 and elevation of smartphone measurements in the soccer field . . .	43
3.19	Code DD in zero-baseline configuration with smartphone within RF enclosure .	44
3.20	Code DD in zero-baseline configuration with smartphone within RF enclosure with 13 dB attenuation . . . . .	45
3.21	Code DD in open sky rooftop environment . . . . .	45
3.22	Code DD in the soccer field . . . . .	46
3.23	Multipath in the soccer field. . . . .	47
3.24	Setup of multipath-impact experiment: smartphone at different heights. . . . .	47
3.25	Moving average of multipath combination: pillar on Geo++ rooftop. . . . .	48
3.26	Moving average of multipath combination: G30 example with C/N0 and elevation	49
3.27	Allan deviation of phase DD in Scenario 3 . . . . .	50
3.28	Example of Allan deviation versus the exponent of the power law noise process $\alpha$ of phase DD in Scenario 3 . . . . .	51
3.29	Smartphone-based positioning: zero-baseline results . . . . .	55
3.30	Smartphone-based positioning: short-baseline results . . . . .	56
3.31	Smartphone-based positioning: short-baseline results in a soccer field . . . . .	56
3.32	Smartphone-based positioning: pillar setups . . . . .	57
3.33	Smartphone-based positioning: 2D error of 30h test over a choke-ring . . . . .	57
3.34	Smartphone-based positioning: the toy train experiment . . . . .	58

3.35 Smartphone-based positioning: sub-set of the LGLN network used to generate SSR parameters . . . . .	59
3.36 Smartphone-based positioning: real-time experiment . . . . .	59
3.37 Smartphone-based positioning: network SSR-based in post-processing . . . . .	60
4.1 Smartphone antenna calibration: reference frame and estimated phase centers . . . . .	65
4.2 Smartphone antenna calibration: setup and concept. . . . .	66
4.3 Smartphone antenna calibration: Mate20X antenna PCV. . . . .	68
4.4 Smartphone antenna calibration: Mate20X antenna PCV formal STD. . . . .	69
4.5 Smartphone antenna calibration: example of PCV geodetic rover antenna . . . . .	70
4.6 Smartphone antenna calibration: Mate20X antenna PCV of L0. . . . .	71
4.7 Smartphone antenna calibration: example of Mate20X antenna PCV difference w.r.t. type-mean. . . . .	72
4.8 Smartphone antenna calibration: STD of Mate20X antenna PCV difference w.r.t. type-mean. . . . .	73
4.9 Smartphone antenna calibration: Mate20X antenna PCV repeatability w.r.t. type-mean. . . . .	74
4.10 Smartphone antenna calibration: antenna location test . . . . .	75
4.11 C/N0 color-map of the L1 reception on the front side of the Mate20X. . . . .	76
4.12 C/N0 color-map of the L5 reception on the front side of the Mate20X. . . . .	77
4.13 Smartphone antenna calibration: antenna location on the phone shell . . . . .	78
4.14 GPS phase DD in the soccer field . . . . .	80
4.15 Galileo phase DD in the soccer field . . . . .	80
4.16 GPS phase DD with C/N0 mask in the soccer field. . . . .	81
4.17 PCV impact on GPS phase DD in the soccer field . . . . .	81
4.18 PCV impact on Galileo phase DD in the soccer field . . . . .	82
4.19 Mate20X positioning performance in the soccer field . . . . .	83
4.20 Positioning performance in the soccer field: Mate20X 50 m from reference station with 5 min reset. . . . .	84
4.21 2D error locating the smartphone on poles over the pillar. . . . .	85
4.22 Geo++ GmbH rooftop: pillars considered. . . . .	86
4.23 DOP of unsuccessful AR period. . . . .	86
4.24 DOP of successful AR period. . . . .	87
4.25 Mate20X positioning performance on the Geo++ rooftop: unsuccessful AR. . . . .	88
4.26 Mate20X positioning performance on the Geo++ rooftop: successful AR. . . . .	88
4.27 Histogram of TFFA on the Geo++ rooftop. . . . .	89
5.1 Weight isolines for the WLS methods when using the distance $d$ and the modified distance $\tilde{d}$ . . . . .	98
5.2 Artificial networks built to simulate the interpolation performance . . . . .	99
5.3 Real reference stations networks . . . . .	99
5.4 Performance of the interpolation techniques in a squared grid network . . . . .	101
5.5 Performance of the interpolation techniques in a circular grid network . . . . .	102
5.6 Interpolation techniques performance using simulated data with the geometry of the NETPOS network . . . . .	103
5.7 Interpolation techniques performance using simulated data with the geometry of the LGLN network . . . . .	103
5.8 Interpolation techniques performance using simulated data with the geometry of the GEONET network . . . . .	104
5.9 Sensitivity to the error in the estimation of the direction of propagation of the parameter . . . . .	105
5.10 Map of the best WLS approach in the NETPOS network using artificial data. . . . .	106

5.11	Map of the best WLS approach in the LGLN network using artificial data . . . . .	107
5.12	Map of the best WLS approach in the GEONET network using artificial data . . . . .	107
5.13	GNSS-estimated and NWM-based ZTD comparison . . . . .	109
5.14	GNSS-estimated and NWM-based ZTD comparison . . . . .	109
5.15	DWD weather data: pressure over temperature . . . . .	110
5.16	DWD weather data: humidity over temperature squared . . . . .	111
5.17	DWD weather data: precipitations . . . . .	111
5.18	Average ZTD values computed from GNSS data and NWM gradients for the NET- POS network . . . . .	113
5.19	Average ZTD values computed from GNSS data and NWM gradients for the LGLN network . . . . .	113
5.20	Cumulative interpolation error of ZTD data during weather fluctuations . . . . .	115
5.21	ZTD distribution along gradient direction and cumulative error during a partic- ularly perturbed hour . . . . .	116
5.22	TEC values of the ionospheric grid residual SSR parameter for satellite E31 during the TID interval . . . . .	118
5.23	Comparison between average WLS2 and WLS2DL interpolation error in the Oki- nawa GEONET network . . . . .	119
5.24	Cumulative interpolation error for the E01, E13, E31 satellites . . . . .	120
5.25	Cumulative error for satellite E31 . . . . .	120
5.26	Simulation of ambiguity resolution performance in presence of ionospheric biases.	122
5.27	Setup for the evaluation of the impact of ionospheric errors in the SSR messages	123
5.28	LGLN sub-net used in the SSRZ error injection test . . . . .	123
5.29	SSR-based positioning results comparison with artificial ionospheric error in the SSRZ stream . . . . .	124
C.1	Specific satellite examples of phase DD in Scenario 1 compared to TD, C/N0 and elevation . . . . .	149



# LIST OF TABLES

---

3.1	C/N0 scenarios used for the DD test. . . . .	34
4.1	Calibration characteristic values of the Mate20X's PCV . . . . .	67
4.2	Smartphone-based positioning results. . . . .	89
5.1	Summary of the best and second best interpolation in terms of RMSE in the three real networks . . . . .	106
5.2	Summary of interpolation results of ZTD estimated using GNSS data from the NETPOS and LGLN networks. . . . .	114
5.3	Summary of interpolation results of ZTD estimated using GNSS data from the NETPOS and LGLN networks w.r.t. WLS2 method. . . . .	114
5.4	NWM gradients and TID parameters proposed messages . . . . .	127
A.1	RTCM-SSR standardized messages (RTCM Special Committee No. 104, 2016). . .	131
A.2	RTCM-SSR proposed messages. . . . .	132
B.1	Current SSRZ messages (Geo++ GmbH, 2020). . . . .	141
C.1	Android Location API - GNSSMeasurements constants. . . . .	150
C.2	Android Location API - Clock and Measurements fields. . . . .	151



---

# 1. Introduction

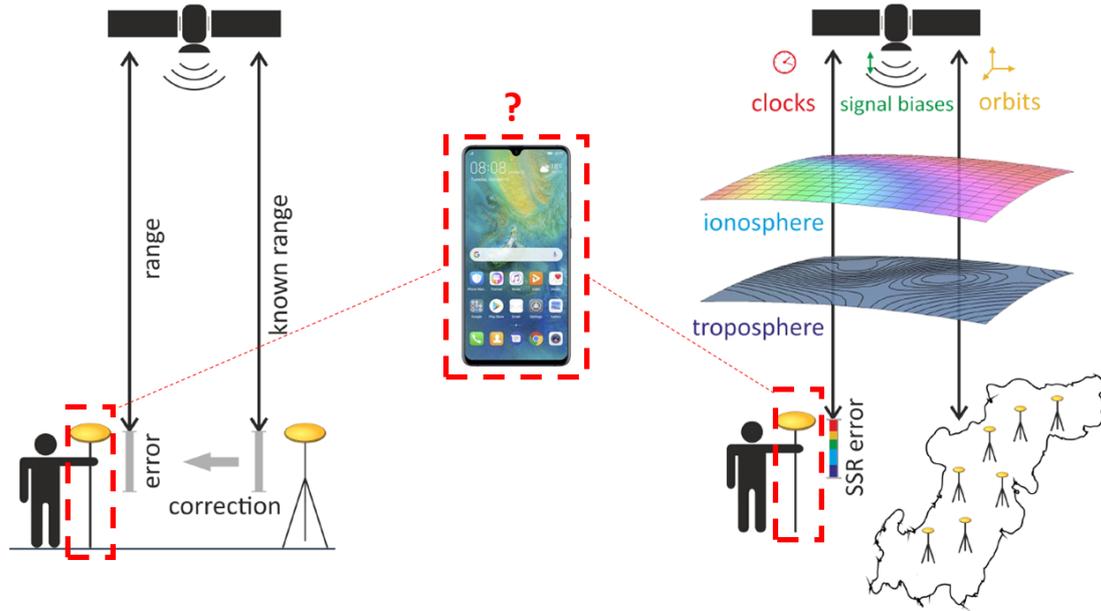
---

Currently, the individual localization on a map using portable devices is an everyday activity that almost everybody in the world can experience. The roots of the essential principle behind it can be found in the *science of mapping and measuring the Earth's surface*, i.e. the classic definition of geodesy given by Helmert (Helmert, 1880-1884). In fact, one of the basic problems that geodesy tackles is the computation of precise global, regional and local three-dimensional positions (Seeber, 2003). However, geodesy science includes many different applications, e.g. the computation of the terrestrial gravity field and ocean surface modeling. Satellite geodesy uses measurements from artificial satellites orbiting around the Earth to solve geodesy-related problems (Seeber, 2003). The use of artificial satellites implies a good knowledge of the position of the satellites in their orbital motion around the Earth. It is therefore related to Precise Orbit Determination (POD) studies. The tasks involving satellite geodesy are variegated: e.g., to measure the motion of continents precisely and to monitor global deformation of the Earth. The satellites that provide such measurements are constellations of satellites, the so-called Global Navigation Satellite Systems (GNSSs). As suggested by the name, these constellations provide global coverage of the Earth, being an asset for geodesy-related purposes.

Since the first complete GNSS constellation, i.e. the Navigation System with Time And Ranging (NAVSTAR) Global Positioning System (GPS), in 1995, in the last decades, many impressive results have been achieved and new constellations built. In addition to GPS, GNSS constellations with global coverage are the Russian GLObal'naya Navigatsionnaya Sputniko-vaya Sistema (GLONASS), the European Galileo and the Chinese BeiDou Navigation Satellite System (BDS). Furthermore, there are constellations with regional coverage, the so-called Regional Navigation Satellite Systems (RNSS): the Japanese Quasi-Zenith Satellite System (QZSS) and the Indian Regional Navigation Satellite System (IRNSS) renamed to Navigation with Indian Constellation (NavIC) in 2016. Besides the GNSS/RNSSs, there are also the Satellite-Based Augmentation Systems (SBASs). SBAS constellations employ geostationary communications satellites to provide differential correction data and integrity information to GNSS users (e.g. Teunissen and Montenbruck, 2017). For the ease of notation, hereafter, the term GNSS also includes RNSS.

Today, GNSS measurements are used for disparate purposes and not only geodesy. They include investigations in geophysics, meteorology, oceanography and space weather monitoring. Industrial applications involve agriculture (e.g. to map the field of crop or a vineyard) and automotive among many others. The technology behind GNSS-based positioning, including instrumentation, hardware, and software, is developing quickly and continuously creates new challenges.

Because of its short wavelength, the GNSS carrier-phase measurement is an essential observable for high-accuracy positioning purposes. However, this observation has an ambiguous nature. The Ambiguity Resolution (AR) is a crucial process to reach fast, highly precise GNSS-based positioning. The two main GNSS-based positioning techniques are Precise Point Positioning (PPP) and Real-Time Kinematic (RTK). RTK is a carrier-phase differential technique used to compute real-time cm-level user positions by utilizing a nearby reference station and reliably solving the phase ambiguities. The user measurement is corrected utilizing the measurements of the reference station (see left side of Fig. 1.1). However, this technique is strongly influenced by distance-dependent errors, like atmospheric effects. In the last years, many reference station networks have been set up, enabling GNSS services that can provide a state vector of corrections to the user to perform PPP. The state of each error component (e.g. orbit,



**Figure 1.1:** From left to right: range corrections for a user’s location transmitted by a reference station and visualization of the SSR corrections generated by a network of reference stations for a user’s position. One of the main questions addressed in the thesis is if the user can use a smartphone (center of the picture, within the dashed red-colored rectangle) instead of a geodetic-grade receiver (within the dashed red-colored rectangles) for RTK-level positioning. Pictures provided by Geo++ GmbH.

clock, ionosphere, troposphere) can be estimated in real-time using GNSS observations from the network. Such a process is simplified in Fig. 1.1 (right side). Receiving the complete state information allows the user to generate GNSS corrections valid for their own position. If the corrections preserve the integer nature of the observations, AR becomes possible, enabling the so-called PPP-RTK positioning. In order to provide the corrections, the error states are represented mathematically and consistently in the so-called State Space Representation (SSR).

One of the objectives of this dissertation is to develop techniques to support the improvement of high-accuracy satellite-based positioning solutions and the progress of the newest positioning applications. In the last years, an innovative GNSS-user application evolved: the use of GNSS raw data from Android devices for accurate and precise positioning. In 2016, the Android operation system enabled direct access to GNSS raw measurements in smartphones. The new availability of data *in the palm of the user’s hand*, as depicted by Kenneth et al. (2015), triggered the intense and growing interest of many research groups to understand what can be achieved with such devices. The first question to answer is if the quality of the smartphone measurements is appropriate for positioning purposes. In particular, a reliable and continuous extraction of carrier phase measurements is essential to obtain RTK-level positioning and replace a user’s geodetic-grade receiver with a smartphone (as depicted in Fig. 1.1). Due to the cellphone’s main features (e.g. receiving a signal from any direction and especially during a phone-call), this objective has been recognized as challenging since the beginning of smartphone-based positioning research. A positive outcome to this question opens dynamic and fast innovations of new low-cost applications for satellite-based positioning.

Recently, the number of devices capable of providing GNSS measurements is growing and growing. All the users equipped with cellphones with an Android version greater or equal to 7.0 are capable of retrieving GNSS raw-data. As reported by Liu (2020), in 2019, the number of Android’s users was 1.6 billion worldwide. Also, worldwide, the most popular Android version is Android 9 (Liu, 2020). This means that roughly 20% of the world population can access

---

raw GNSS measurements. The large and global availability of such low-cost GNSS receivers provides a research opportunity to develop new techniques and applications. This can unveil applications making use of GNSS measurements.

Moreover, since 2018, smartphones have been equipped with dual-frequency and multi-constellation receivers. Nowadays, there are, to the best of the author's knowledge, more than 40 smartphone models on the market that provide dual-frequency receivers. The dual-frequency capability enables the reduction of frequency-dependent errors that affect satellite measurements, e.g. the ionospheric delay. Taking advantage of such devices, promising results have been achieved already. A few decimeters accuracy positioning has been demonstrated with relatively short convergence time, i.e. a few minutes (e.g. Critchley-Marrows, 2020). However, high-accuracy performance cannot be achieved in the user's hand yet.

In this work, the main related issues are introduced and investigated, showing verified methods to achieve reliable high-accuracy positioning using smartphones. The GNSS raw-measurements retrieved from the Android Application Programming Interface (API) are presented and analyzed. Statistical considerations are introduced, especially concerning the Carrier-to-Noise-ratio ( $C/N_0$ ) over long data-sets. In fact, smartphone measurements suffer from low  $C/N_0$  that has been demonstrated to be related not only to low-elevation satellites. Also, site-dependent effects (e.g. multipath), which are a cause of error in GNSS-positioning, are investigated.

The most relevant issue related to doing positioning with smartphones is the antenna of the devices. In fact, most of the phones are equipped with an omnidirectional antenna that makes smartphones sensitive to surface reflections (i.e. multipath) of the nearby environment. Along with multipath, the antenna pattern variations are the main source of station-dependent errors. Usually, to correct this electromagnetic effect, the geodetic antenna is calibrated, providing the so-called antenna corrections. Here, the calibration of a dual-frequency smartphone is reported together with the impact on positioning performance. The antenna calibration is demonstrated to be an essential step to achieve a reliable, accurate and highly precise estimated position. The positioning performance is investigated using SSR-based techniques assessing the potential of reliable RTK-level positioning.

In SSR-based positioning, atmospheric parameters computed by a network of GNSS reference stations need to be interpolated for the user location. Atmospheric delays are GNSS error sources with a significant impact on the positioning performance and AR. As a consequence, the interpolation process has a direct impact on AR. Furthermore, in the last decades, the number of severe weather events has increased around the world. Focusing on Europe, Rädler et al. (2019) shows that the frequency of damaging convective weather events, including lightning, hail and severe wind gusts will likely increase over Europe until the end of this century. Moreover, during severe events like thunderstorms, the generated waves reach the upper stages of the atmosphere, perturbing the ionosphere. These perturbations can lead to the so-called Traveling Ionosphere Disturbances (TIDs), which can significantly affect the GNSS measurements. Thunderstorms are only one of the possible sources of TIDs, which mainly depend on solar activity. The increasing number of severe events raises interest in investigating the impact of such events on the positioning performance. In this dissertation, the interpolation quality is analyzed along with the benefit of using external information retrieved from atmospheric models. Accordingly, multiple interpolation techniques are investigated and new methods are proposed to make the AR process more robust during severe weather and TID events. These alternative methods take advantage of external atmospheric models that have been becoming more and more accurate during the last years.

The primary objective of this thesis is to investigate and show the potential for RTK-level smartphone-based positioning using state space GNSS augmentation techniques. Besides, one of the main issues in SSR-based positioning is assessed, investigating the interpolation process of atmospheric parameters. The following section introduces the structure of the thesis.

---

## Outline

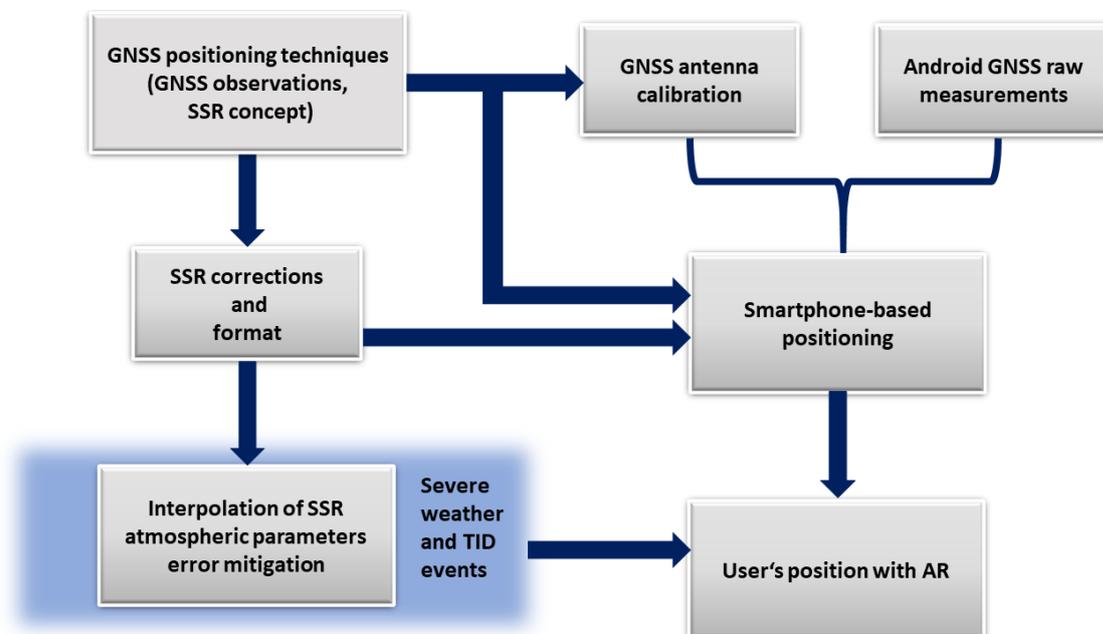
Fig. 1.2 shows a schematic representation of the connection between the topics of Chapters 2-4. The central scientific questions are:

- which level of accuracy and precision can be achieved in smartphone-based positioning? Can a user replace a geodetic-grade rover receiver with a smartphone (as depicted in Fig. 1.1) ?
- What is the impact of interpolation errors of SSR atmospheric corrections on positioning for a user within a network of GNSS reference stations? Can this error be mitigated during severe weather and TID events by including external atmospheric model information?

In order to address these questions, multiple aspects are investigated through the thesis. Apart from the introductory chapter Chapter 2, each chapter has a motivation and discussion sections. While the motivation section presents the reasons behind the analysis, the discussion section summarizes the primary outcome and examines the possible benefits of the investigation.

The positioning techniques employed in this work are introduced in Chapter 2 along with the description of the GNSS observations. In particular, the central SSR concept is presented. Chapter 2 introduces the main concepts and notation used through the thesis. In addition, the use of SSR corrections and the format utilized for their transmission are presented.

A method to reconstruct GNSS observations from the raw measurements retrieved from the Android API and to assure their use for positioning purposes is reported in Chapter 3. Also, the quality of GNSS raw measurements for positioning applications is investigated. In summary, Chapter 3 reports the potential of cm-level positioning using smartphones' measurements by employing the positioning techniques introduced in Chapter 2.



**Figure 1.2:** Schematic representation of the connection between the topics of Chapters 2-4, having the final objective to compute the user's position with AR.

---

Chapter 4 introduces the principle of the calibration of a GNSS antenna using the GNSS observations and SSR concepts presented in Chapter 2. In this work, the GNSS smartphone's antenna is calibrated by an absolute robot-based field calibration. The results of the calibration and the impact of using it in the positioning algorithm are reported achieving an estimated user's position with successful AR.

In Chapter 5, the interpolation error of SSR atmospheric parameters is investigated. Multiple interpolation techniques, e.g Inverse Distance Weighted (IDW), Ordinary Kriging (OK) and Weighted Least Squares (WLS), are analyzed. In addition, alternative approaches are proposed to mitigate the interpolation error during specific severe weather and TID events. As examples, two severe weather events in northern Europe in 2017, and one TID event over Japan in 2019 have been analyzed. The final aim of the interpolation analysis is to provide guidelines to have a robust AR for high-accuracy and precision positioning for a user's location within a network. Also, the impact of an SSR ionospheric mismodeling on the estimated position is assessed in terms of AR and positioning error.

Finally, in Chapter 6, the main conclusions are summarized with an outlook for further research steps.



## 2. GNSS Positioning Techniques

---

### 2.1 The GNSS observation equations

The main idea behind GNSS-based positioning is to localize a point on the Earth's surface by making use of the observed range between the point and the GNSS satellite. The observed range is related to a signal broadcast by GNSS satellites traveling towards the Earth's surface until being received by an appropriate device (i.e. a receiver). The navigation satellites typically make use of the L-band frequency, i.e. the band between 1 and 2 GHz, to broadcast their signals (e.g. Teunissen and Montenbruck, 2017). Taking into account the traveling time of the signal and its velocity, the range can be reconstructed as follows:

$$\rho = c(t_A - t_E), \quad (2.1)$$

where  $c$  is the speed of light,  $t_A$  the acquisition time and  $t_E$  the emission time of the signal. The localization of a point on the Earth's surface is expressed in coordinates in a specific reference frame. Therefore, the coordinates of the receiver can be written in the following way:

$$\mathbf{x}_r = \mathbf{x}_s + \rho \hat{\mathbf{u}}_{\text{rad}}, \quad (2.2)$$

where  $\mathbf{x}_r$  is the receiver vector with components the coordinates of the receiver,  $\mathbf{x}_s$  is the satellite vector and  $\hat{\mathbf{u}}_{\text{rad}}$  is the unit vector of the radial direction from the satellite towards the receiver position. The components of both  $\mathbf{x}_r$  and  $\mathbf{x}_s$  are expressed w.r.t. the same reference system, e.g. the Earth-Centered Earth-Fixed (ECEF), the International Terrestrial Reference Frame (ITRF), and the World Geodetic System 1984 (WGS84) systems.

In this dissertation, a user will be considered as someone who utilizes a GNSS receiver to gather measurements of the GNSS signals to localize a position, i.e. to express the receiver coordinates w.r.t. a defined reference system. The computation of the position vector of the receiver  $\mathbf{x}_r$  goes through an estimation process using GNSS observations as described by many authors (e.g. Teunissen and Montenbruck, 2017). Conceptually, the GNSS observation is the range  $\rho$ . In general, the estimation process can be summarized by the following equation model:

$$\mathbf{f}(\mathbf{x}) = \mathbf{y} + \mathbf{v}, \quad (2.3)$$

where  $\mathbf{x}$  is the vector of the parameters that need to be estimated (including  $\mathbf{x}_r$ ),  $\mathbf{y}$  is the observation vector,  $\mathbf{v}$  is the residual vector and  $\mathbf{f}$  is a non-linear functional relationship between observations and parameters. It is worth mentioning that all the vectors of Eq. 2.3 are functions of the time. The principles of the estimation of the parameters using GNSS observations will be introduced in Section 2.2.

In their measurements, the user observes the range  $\rho$  expressed in Eq. 2.1. However, in Eq. 2.1, many physical aspects related to a broadcast and received radio-frequency signal traveling in space and atmosphere have been neglected. First of all, synchronization delays of the satellite's and receiver's clocks need to be taken into account. Secondly, the signal travels through the atmosphere, which is not the vacuum but a medium. Hence, the signal is delayed. In their path to the receiver, the signal can be deviated by impinging other surfaces and causing a range distortion. Another aspect is the electromagnetism of the antennas used to broadcast and acquire the signal, which introduces delays that cannot be neglected. Furthermore, the measurements are affected by the relative motion between user and satellite (e.g. the Earth's rotation) and by the relativistic effects. Finally, the hardware used to transmit and receive the

signal introduces delays. Therefore, the expression of the observation that is observed by the user cannot be defined by the range expressed by Eq. 2.1, and it needs to be modified to take into account all the error sources. The observation is the so-called pseudorange, and it will be described in the following section.

Many textbooks present the equations of the GNSS observations (e.g. Teunissen and Montenbruck, 2017), and it is not the main purpose of this dissertation to provide a fully detailed description of all the related aspects. However, it is worth introducing the notation that is used in the whole work. For further details about the basic concepts of GNSS measurements, the reader is referred to Spilker Jr et al. (1996); Parkinson et al. (1996); Montenbruck and Gill (2002); Seeber (2003); Misra and Enge (2006); Teunissen and Montenbruck (2017).

### 2.1.1 Pseudorange measurements

The GNSS receiver generates a replica of the received signal as best as possible by using satellites known pseudo-random-code and its internal frequency source. The alignment of the receiver code to the received signal is done by the receiver's Delay Lock Loop (DLL). The necessary time shift combined with the code-related information provided by the satellite navigation data determines the apparent travel time of the signal. Multiplying the apparent travel time with the speed of light (in the same way done in Eq. 2.1), the apparent range, i.e. the pseudorange, is obtained. As aforementioned, this quantity differs from the actual range because of the misalignment between the receiver's and satellite's clocks. Furthermore, the signal reception is affected by other error sources, like e.g. atmospheric refraction. Taking into account all the introduced considerations, at a specific epoch  $t$  a receiver  $r$  observes the signal  $j$  of the satellite  $s$  as follows:

$$\begin{aligned}
 p_{r,j}^s(t_A) = & \rho_r^s(t_A) \\
 & + c \left( dt_r(t_A) - dt^s(t_E) + \delta t^{\text{rel}}(t_A, t_E) \right) \\
 & + c \left( d_{r,j} - d_j^s \right) \\
 & + I_{r,j}^s(t_A) + T_r^s(t_A) \\
 & + \xi_{r,j}^s(t_A, t_E) + \mu_{r,j}^s(t_A) \\
 & + e_{r,j}^s(t_A).
 \end{aligned} \tag{2.4}$$

In Eq. 2.4,  $t_A$  and  $t_E$  are the signal acquisition and emission time, respectively, and  $\rho_r^s(t_A)$  is the geometric range (i.e. Eq. 2.1 in an ideal scenario without any delays and errors). The delays and error sources that appear in the equation can be summarized as follows:

- $dt_r(t_A) - dt^s(t_E) + \delta t^{\text{rel}}(t_A, t_E)$  is the delay due to the receiver's ( $dt_r(t_A)$ ) and satellite's ( $dt^s(t_E)$ ) clock offsets, and the relativistic effects ( $\delta t^{\text{rel}}(t_A, t_E)$ ). More information about the relativistic effects can be found in, e.g., Ashby (2003) and Teunissen and Montenbruck (2017).
- $d_{r,j} - d_j^s$  is the delay caused by the hardware.
- $I_{r,j}^s(t_A) + T_r^s(t_A)$  is the range deviation due to the signal refraction in ionosphere ( $I_{r,j}^s$ ) and troposphere ( $T_r^s(t_A)$ ).
- $\xi_{r,j}^s(t_A, t_E) + \mu_{r,j}^s(t_A)$  is the station dependent error, made of two contributions: the Group Delay Variations (GDV)  $\xi_{r,j}^s(t_A, t_E)$  due to the receiver's and satellite's antennas and the multipath (MP) effect  $\mu_{r,j}^s(t_A)$ .
- $e_{r,j}^s(t_A)$  is the receiver's code noise.

### 2.1.2 Carrier phase measurements

Along with the pseudorange, the receiver also measures the signal's carrier phase from its Phase Lock Loop (PLL). The measurement is a fractional phase shift of the receiver's replica and of the received carrier phase. The carrier phase measurements are more precise (i.e. few mm) than the pseudorange measurements (i.e. dm-level) because of the short wavelength of roughly 20 cm (compared to the 293 m of the GPS C/A code). However, satellite navigation data cannot be used to obtain unambiguous carrier phase observations. The reason is that the integer number of cycles of the phase between the transmitter and receiver when the tracking starts is unknown, causing the ambiguous nature of the phase observations. Similar to the pseudorange, the carrier phase observation equation can be expressed in the following way:

$$\begin{aligned}
 \phi_{r,j}^s(t_A) = & \rho_r^s(t_A) \\
 & + c \left( dt_r(t_A) - dt^s(t_E) + \delta t^{\text{rel}}(t_A, t_E) \right) \\
 & + c \left( d_{r,j} - d_j^s \right) \\
 & - I_{r,j}^s(t_A) + T_r^s(t_A) \\
 & + \lambda_j \left( \omega_r^s(t_A, t_E) + N_{r,j}^s \right) \\
 & + \zeta_{r,j}^s(t_A, t_E) + \mu_{r,j}^s(t_A) \\
 & + \epsilon_{r,j}^s(t_A),
 \end{aligned} \tag{2.5}$$

where  $\zeta_{r,j}^s(t_A, t_E)$  is the Phase Center Variations (PCV) component due to the receiver's and satellite's antennas. The wind-up correction,  $\omega_r^s(t_A, t_E)$ , accounts for a change in the measured phase due to a change in the relative geometry between satellite and receiver antenna (e.g. Hauschild, 2017; Wu et al., 1993). Here,  $\epsilon_{r,j}^s(t_A)$  is a residual phase noise.

The comparison between Eq. 2.4 and Eq. 2.5 highlights the difference in sign of the ionospheric impact. The sign change is a consequence of the frequency-dependency of the signal propagation through a dispersive medium and different velocity of carrier and the modulation of the signal (e.g. Langley, 1998a; Petrie, 2011; Teunissen and Montenbruck, 2017).

### 2.1.3 Atmospheric delays

The GNSS signal interactions with ionosphere and troposphere are two of the major GNSS error sources. In terms of range, the tropospheric delay can be up to ~2.3-2.6 m at the zenith and ~25 m for elevations close to five degrees (at sea level, the values depend on the location). Concerning the ionosphere, the errors can vary between ~1 m and tens of meters at the zenith multiplied by roughly a factor of three at low elevation angles when using a single layer model of the ionosphere (Schaer, 1999). Atmospheric delays are obtained by integrating the refractive index  $n$  along the signal path. The refractive index depends on permittivity and permeability of the medium that the signal is passing through, which vary in space and time. Since the value of  $n$  is close to one, many publications introduce the so-called refractivity  $N$  as follows:

$$N = (n - 1) \times 10^6. \tag{2.6}$$

Although both delays are related to the signal propagation through a medium, the physical processes behind ionospheric and tropospheric delays are different. On the one hand, the ionospheric delay is mainly related to the Sun's activity and the interaction between ionized particles. On the other hand, the tropospheric delay is related to the local weather conditions. In fact, the troposphere is the lowest layer of the atmosphere, i.e. altitude < 16 km (at the equator), while the ionosphere approximately covers the altitude between 80 km and 1000 km.

A significant difference between the two atmospheric layers is that while the ionosphere is a dispersive medium, the troposphere is not. The troposphere can be seen as a layer made of gas where small distances (roughly 0.1 nm) and strong forces with fast oscillations among particles are involved. The ionosphere, instead, is made of plasma whose particles are characterized by large distances (roughly 0.1 mm), weak forces and slow oscillations. The plasma frequency, i.e. the frequency at which electrons oscillate about their equilibrium positions, is up to 22 MHz for the F2-peak (350 km in a Chapman layer description, see, e.g., Petrie (2011) for a complete review), where the electron density is  $N_e = 6 \times 10^{12}$  electrons  $\text{m}^{-3}$ . Hence, the maximum plasma frequency is 100 times lower than the L-band GNSS frequencies (e.g. L1 frequency is 1.575 GHz). In the troposphere, instead, the main transition effects are related to the atomic frequency that is about some hundreds of THz, i.e. five orders greater than the L1 GNSS frequency. Therefore, GNSS frequencies are far below atomic resonances, but far above electron (plasma) resonances that makes the ionosphere a dispersive medium. This fact causes the frequency dependency of the ionospheric delay  $I_{r,j}^s$ , compared to the tropospheric delay  $T_r^s$ . The dispersive nature implies the different phase (carrier) and group (code) velocity resulting in the opposite sign in Eq. 2.4 and Eq. 2.5.

### The tropospheric delay

Details about the modeling of the troposphere can be found in multiple textbooks (e.g. Tenissen and Montenbruck, 2017) or dissertations (e.g. Kleijer, 2004). Here, the purpose is to provide the reader useful information to analyse the results of this dissertation properly. The slant tropospheric delay,  $T_r^s$ , depends on:

- dry gases, varying little over temporal and spatial scales of hours and km;
- water vapor, with highly variable spatial and temporal distributions.

For a satellite  $s$  and receiver  $r$ , the slant tropospheric delay is computed by the following integral (e.g. Hauschild, 2017):

$$T_r^s = 10^{-6} \int_s^r N dl, \quad (2.7)$$

where  $N$  is the tropospheric refractivity, and  $l$  is the propagation path of the signal. As the troposphere can be considered as moist air, the refractivity  $N$  can be expressed as the sum of a dry (hydrostatic)  $N_d$  (depending only on pressure and temperature) and wet component  $N_w$ , depending on water vapor pressure (Essen and Froome, 1951):

$$\begin{aligned} N &= k_1 \frac{p}{T} + k_2' \frac{e}{T} + k_3 \frac{e}{T^2}, \\ &= N_d + N_w, \end{aligned} \quad (2.8)$$

where  $T$  is the temperature,  $p$  and  $e$  are the total pressure and partial pressure of the wet constituent, respectively. Furthermore, the constant  $k_2'$  is computed as follows (e.g. Hauschild, 2017):

$$k_2' = k_2 - k_1 \frac{R_d}{R_w}, \quad (2.9)$$

where  $R_d$  and  $R_w$  denote the gas constants for dry air and water vapor, respectively. The three constants are given as  $k_1 = 77.6$  K/mbar,  $k_2 = 64.8$  K/mbar and  $k_3 = 3.776 \times 10^5$  K<sup>2</sup>/mbar (Thayer, 1974). During the years, multiple models have been developed. The models relate the state of the atmosphere at an arbitrary height to the atmospheric parameters at the user height and thus allow the integration of Eq. 2.7 into zenith direction (Hauschild, 2017). As a consequence, Eq. 2.7 can be re-written in the following way:

$$T_r^s = m_h(\varepsilon) ZHD_r^s + m_w(\varepsilon) ZWD_r^s + m_g(\varepsilon) [G_{N,r}^s \cos(\alpha) + G_{E,r}^s \sin(\alpha)]. \quad (2.10)$$

In Eq. 2.10,  $m_h(\varepsilon)$  and  $m_w(\varepsilon)$  are the so-called mapping functions, which map the Zenith Hydrostatic Delay (ZHD) and Zenith Wet Delay (ZWD) into the direction of the line of sight through the elevation  $\varepsilon$ . The elevation-dependent mapping function  $m_g(\varepsilon)$  relates the North and East components of the tropospheric horizontal gradient  $\mathbf{G} = [G_N, G_E]$ .  $\mathbf{G}$  describes an azimuthal asymmetry of the delay through the dependency on the azimuth  $\alpha$ . Horizontal gradients are needed to consider a systematic component in the North/South direction towards the equator due to the atmospheric bulge (MacMillan and Ma, 1997). Typical values of the magnitude of the gradients are roughly 0.5 mm but can reach or exceed 1 mm (MacMillan and Ma, 1997; Petit and Luzum, 2010), as shown in Chapter 5.

The sum of zenith dry (hydrostatic) and wet components gives the Zenith Tropospheric Delay (ZTD). Accordingly, Eq. 2.10 can be written as:

$$T_r^s = m_t(\varepsilon)ZTD_r^s + m_g(\varepsilon)[G_{N,r}^s \cos(\alpha) + G_{E,r}^s \sin(\alpha)], \quad (2.11)$$

where  $m_t(\varepsilon)$  is the function that maps the  $ZTD$  into the line of sight direction. While the hydrostatic delay component can be accurately computed based on the Saastamoinen model (Saastamoinen, 1972; Petit and Luzum, 2010), the wet component needs to be estimated by the GNSS positioning algorithm.

Several mapping functions have been developed by many authors, e.g. Marini (Marini, 1972), Herring (Herring, 1992), Niell (Niell, 1996), Vienna Mapping Functions (VMF) (Böhm et al., 2006b), and Global Mapping Functions (GMF) (Böhm et al., 2006a). More details about the basic description of hydrostatic and wet delays, and mapping functions can be found in, e.g., Seeber (2003); Petit and Luzum (2010); Hauschild (2017); Hobiger and Jakowski (2017).

In the last years, many studies have been carried out to develop Numerical Weather Models (NWM), like, e.g., the European Center for Medium-Range Weather Forecasts (ECMWF) and the Global Forecasting System (GFS). Based on the physical quantities provided by NWM, a user can retrieve tropospheric delays. Examples of the methodology to generate slant and zenith tropospheric delays can be found in, e.g., Nafisi et al. (2011); Zus et al. (2012). In Chapter 5, the benefit of using external information like NWM for GNSS-based positioning is investigated.

### The ionospheric delay

The propagation of an electromagnetic wave through the ionosphere has been investigated deeply. For L-band signal specific assumptions can be made:

- the effects of electron collisions are neglected;
- the plasma is a cold plasma (i.e. the velocities of the electrons' thermal motions are much less than the phase velocity of the wave):
- there is a uniform magnetic field.

See e.g. Yeh et al. (1972); Davies (1990); Petrie (2011) for further information. These assumptions allow to describe the refractivity index as a function of the electron density  $N_e$  in the generally known as *Appleton-Lassen* equation (Lassen, 1927; Appleton, 1932). The Appleton-Lassen equation has been studied and re-arranged by multiple authors and a review can be found in, e.g., Petrie (2011). It is out of the scope of this work to go into the details of the electromagnetic equations. The main focus is on the effects on the pseudorange and carrier phase observations. Integrating the effect of the refractive index along the curve path and neglecting the second and third-order refractive index effects (e.g. Petrie, 2011), the ionospheric delay yields:

$$I_{r,j}^s = \frac{40.3 \times 10^{16}}{f_j^2} \int_s^r N_e dl, \quad (2.12)$$

where  $f_j$  is the frequency of the received signal and the ionospheric delay is expressed in meters. The integral of the electron density  $\int_s^r N_e dl$  is the so-called Slant Total Electron Content (STEC):

$$STEC = \int_s^r N_e dl. \quad (2.13)$$

Another quantity used in the following chapters is the Vertical Total Electron Content (VTEC) defined by:

$$VTEC = \int_s^r N_e dh, \quad (2.14)$$

where  $h$  is the height above the Earth's surface. STEC and VTEC are commonly measured in TEC Units (TECU):  $1\text{TECU} = 10^{16}\text{electrons}/m^2$ . As an example, the impact of 1 TECU on the L1 frequency is about 16 cm. Assuming a simple single layer model, the ionosphere can be represented by a thin shell surrounding the Earth at certain height above the Earth's surface (e.g. Schaer, 1999; Hauschild, 2017). In that case, the STEC can be derived from the VTEC through the zenith angle  $z'$  at the Ionospheric Pierce Point (IPP). The IPP is the intersection point where the line-of-sight vector cross the spherical ionospheric shell defined at a certain height (typical values are, e.g., 300-400 km). Using this simple model, the relationship between STEC and VTEC is the following:

$$STEC = \frac{1}{\cos(z')} VTEC. \quad (2.15)$$

The mapping function  $\frac{1}{\cos(z')}$  can vary between 1 to roughly 3 for low elevation satellites (e.g. Seeber, 2003).

As mentioned above, the dispersive nature of the ionosphere causes a different phase and group velocity. Hence the ionospheric delay has a positive sign for the pseudorange (see Eq. 2.4), and a negative sign for the carrier phase (see Eq. 2.5). It is worth mentioning that there are additional effects like high order effects and the bending effect (e.g. Leitinger and Putz, 1988). As an example, ionospheric higher order effects can reach a few cm for L1/L2 frequencies (Seeber, 2003). More information about the order of magnitude of the impact of such effects can be found in e.g. Wübbena (1991); Petit and Luzum (2010).

An efficient way of correcting the ionospheric effects is obtained by combining simultaneous measurements of two different frequencies forming the so-called ionosphere free linear combination (e.g. Seeber, 2003; Misra and Enge, 2006; Hofmann-Wellenhof et al., 2012; Leick et al., 2015). It is worth mentioning that concerning carrier phase ionospheric free signal, the resulting combination is not related to integer ambiguities. Therefore, it is not a suitable signal for very precise applications (Seeber, 2003). However, there are many external sources based on models that provide ionospheric corrections to enable precise single-frequency positioning (e.g. Øvstedal, 2002; Beran et al., 2004). Many of them are provided in terms of VTEC on a global or regional scale. Examples of those are Global VTEC maps (e.g. the Global Ionospheric propagation Model (GIM) see Béniguel, 2002), the GPS Klobuchar model (Klobuchar, 1987), the Galileo NeQuick model (Hochegger et al., 2000), empirical standard models, e.g. the International Reference Ionosphere (IRI) (Bilitza, 2001), regional VTEC models, e.g. GNSS-based corrections (see Sub-section 2.2.3).

#### 2.1.4 Double-Difference (DD) observations

Double-Difference (DD) observations are computed by a linear combination of observations of the same kind of different satellites and receivers. In particular, it is the difference between two satellite single-differences of two distinct receivers. It is worth mentioning that the DD refers to the same reception time. If that is not true, a synchronization of the individual receiver times has to be made. Indicating with  $p$  the pivoting satellite, the pseudorange, and

phase satellite  $s$  single differences for the receiver  $a$  are given by the following equations:

$$\begin{aligned}
 p_a^{sp} &= p_a^p - p_a^s \\
 &= \rho_a^{sp} \\
 &\quad + c(dt^{sp} + \delta t_a^{rel,sp}) + cd^{sp} \\
 &\quad + I_a^{sp} + T_a^{sp} \\
 &\quad + \xi_a^{sp} + \mu_a^{sp} \\
 &\quad + e_a^{sp}.
 \end{aligned} \tag{2.16}$$

$$\begin{aligned}
 \phi_a^{sp} &= \phi_a^p - \phi_a^s \\
 &= \rho_a^{sp} \\
 &\quad + c(dt^{sp} + \delta t_a^{rel,sp}) + cd^{sp} \\
 &\quad - I_a^{sp} + T_a^{sp} \\
 &\quad + \lambda^p(\omega_a^p + N_a^p) - \lambda^s(\omega_a^s + N_a^s) \\
 &\quad + \zeta_a^{sp} + \mu_a^{sp} \\
 &\quad + \epsilon_a^{sp}.
 \end{aligned} \tag{2.17}$$

It can be noticed that in the single-satellite difference, the receiver clock offset cancels out since the two observations refer to the same receivers. Furthermore, the phase satellite single difference introduces the so-called Inter-System Biases (ISB) when satellite of different constellations are considered, and, therefore, there might be a difference in the wavelength (e.g. Hauschild, 2017). The receiver single-difference is instead composed of the difference between the observation of the same satellite of two distinct receivers  $a$  and  $b$ . The pseudorange and carrier phase equations are the following:

$$\begin{aligned}
 p_{ab}^s &= p_b^s - p_a^s \\
 &= \rho_{ab}^s \\
 &\quad + c(dt_{ab} + \delta t_{ab}^{rel}) + cd_{ab} \\
 &\quad + I_{ab}^s + T_{ab}^s \\
 &\quad + \xi_{ab}^s + \mu_{ab}^s \\
 &\quad + e_{ab}^s.
 \end{aligned} \tag{2.18}$$

$$\begin{aligned}
 \phi_{ab}^s &= \phi_b^s - \phi_a^s \\
 &= \rho_{ab}^s \\
 &\quad + c(dt_{ab}^s + \delta t_{ab}^{rel,s}) + cd_{ab} \\
 &\quad - I_{ab}^s + T_{ab}^s \\
 &\quad + \lambda^s(\omega_{ab}^s + N_{ab}^s) \\
 &\quad + \zeta_{ab}^s + \mu_{ab}^s \\
 &\quad + \epsilon_{ab}^s.
 \end{aligned} \tag{2.19}$$

Hence, the pseudorange and phase double difference yields:

$$p_{ab}^{sp} = p_b^{sp} - p_a^{sp}, \tag{2.20}$$

$$\phi_{ab}^{sp} = \phi_b^{sp} - \phi_a^{sp}, \tag{2.21}$$

where the satellite and receiver clock offsets cancel out, but residual biases due to different signal travel times and non-synchronized receivers remain. In the case the receivers are co-located, or close enough to neglect the variation of the atmosphere, the DD cancels out also the atmospheric parameters thanks to the between-receiver single difference.

Although linear combinations can eliminate some parameters from the observation equations, the resulting combinations are noisier than the individual observations. In the DD case, following the error propagation law, the Standard Deviation (STD)  $\sigma_{DD}$  yields:

$$\begin{aligned}\sigma_{DD} &= \sqrt{(\sigma_a^p)^2 + (\sigma_a^s)^2 + (\sigma_b^p)^2 + (\sigma_b^s)^2} \\ &\approx 2\sigma_a^s,\end{aligned}\tag{2.22}$$

where the noise of each observation has been considered equal. Therefore, the DD approximately introduces a multiplicative factor of 2 in the noise.

## 2.2 GNSS augmentation techniques

### 2.2.1 State Space Modeling (SSM)

Equations 2.4 and 2.5 describe the observation equations of a physical system. The state variables of the system define a state space, where the system can be represented considering a state vector. In the GNSS signal reception case, the state vector can include the error sources along with the phase ambiguity term and the rover and satellite coordinates. Therefore, satellite-based positioning, i.e. the accurate and precise computation of the rover coordinates, requires an optimal state vector estimation. As stated in Gelb (1974): *an optimal estimator is a computational algorithm that processes measurements to deduce a minimum error estimate of the state of a system by utilizing: knowledge of system and measurement dynamics, assumed statistics of system noises and measurement errors, and initial condition information.* Considering Gaussian processes in Bayesian estimation, it can be assumed that the optimal filter is an optimal linear filter (e.g. Gelb, 1974). In that case, the system can be described by the following equations:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) + \mathbf{L}(t)\mathbf{u}(t),\tag{2.23}$$

$$\mathbf{y}(t) = \mathbf{H}(t)\mathbf{x}(t) + \mathbf{v}(t) ,\tag{2.24}$$

or in the discrete form:

$$\mathbf{x}_{k+1} = \mathbf{\Phi}_k\mathbf{x}_k + \mathbf{\Gamma}_k\mathbf{w}_k + \mathbf{\Lambda}_k\mathbf{u}_k,\tag{2.25}$$

$$\mathbf{y}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k .\tag{2.26}$$

Equations 2.23 and 2.25 describe the *system model*, while Eqs. 2.24 and 2.26 present the *observation model*, respectively. In the formulation mentioned above,  $\mathbf{x}(t)$  is the so called state vector,  $\mathbf{w}(t)$  is a random forcing function,  $\mathbf{F}(t)$  and  $\mathbf{G}(t)$  are matrices arising from the formulation,  $\mathbf{v}(t)$  is the measurement error and  $\mathbf{H}_k$  the measurement matrix. In the discrete case, the matrix  $\mathbf{\Phi}_k$  is the state transition matrix.

Here, a Bayesian process is considered to give a probabilistic meaning to the results. Therefore, the arguments cannot be described only by deterministic quantities, but a stochastic description has to be added. As already mentioned, we consider Gaussian processes, i.e. the process can be described by a normal distribution  $\sim N(M, \sigma^2)$  where  $M$  is the mean value and  $\sigma^2$  the variance of the distribution. Furthermore, in this case, the interest is to estimate the state vector at the actual time, i.e. the focus is on the *filtering* process. The *Kalman filter* (Kalman, 1960) is introduced as an optimal linear filter. For the discrete case, the equations of the model are:

$$\mathbf{x}_k = \mathbf{\Phi}_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \quad \mathbf{w}_k \sim N(0, \mathbf{Q}_k),\tag{2.27}$$

$$\mathbf{y}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k, \quad \mathbf{v}_k \sim N(0, \mathbf{R}_k),\tag{2.28}$$

where  $\mathbf{Q}$  is the covariance matrix of  $\mathbf{w}$  and  $\mathbf{R}$  is the covariance matrix of  $\mathbf{v}$ . In this case  $\mathbf{w}$  is the process noise and  $\mathbf{v}$  is the measurement noise.

However, in the GNSS observation equations (Eq. 2.4 and Eq. 2.5), the relation between the observation and the state is not linear. Therefore, Eq. 2.28 can be applied once the pseudorange and phase observation equations are linearized. The linearization of the observation equations is performed through the first derivative w.r.t. the state vector. Further details can be found in De Jonge's dissertation (De Jonge, 1998).

Another aspect related to the state space estimation is the availability of multiple receivers covering a defined geographic area. The setup mentioned above allows estimating the state space valid over the whole covered region. The set of receivers can be called a network of receivers. If the coordinates and the antennas' models of the receivers are accurately and precisely known, the network can be called a network of reference stations. Hence, a user within the region covered by the network can take advantage of the network-estimated parameters. Accordingly, network-based corrections can be sent to the user to correct their observations, forming a GNSS-augmentation system. A network continuously providing corrections is called Continuously Operating Reference Stations (CORS). Depending on the application, the network-based GNSS-augmentation can be employed using different approaches, which will be introduced in the following part of the chapter.

### 2.2.2 State Space Representation (SSR)

The State Space Representation (SSR) is the mathematical representation of the state that describes the system in the state space. In GNSS-based positioning, the representation provides information about the GNSS errors based on the state space estimation (Wübbena et al., 2005). The GNSS state vector of SSR involves the following parameters:

- satellite orbit errors;
- satellite clock errors;
- satellite code and phase biases;
- ionospheric propagation delays and advances;
- tropospheric delays.

The satellite orbit and clock errors are estimated to improve the broadcast information about spacecraft position and clock provided by the satellite itself. Satellite code and phase biases are delays within the satellite software and hardware.

In SSR-based positioning, the SSR corrections are generated by the processing of GNSS observations of a network of reference stations. The generated SSR parameters are broadcast to the user as corrections to apply to their observations (Wübbena et al., 2005). The use of SSR corrections is further described in the next subsection.

The estimation process used to generate the SSR corrections must correct the GNSS observations for other effects like reference station site displacements, relativity, phase wind-up, satellite and receiver PCV and GDV and residual atmospheric delays. In this way, the SSR corrections are free of such effects. Furthermore, a user applying the SSR corrections may consider correction models for:

- coordinate frame transformation to account (e.g. tectonics);
- solid Earth tides;
- ocean loading;
- atmospheric pressure loading;

- rotational deformation due to polar motion (e.g. pole tide);
- relativistic effects;
- satellite phase wind-up;
- satellite antenna code and phase center variations.

For further details about these correction models and corrections refer to Petit and Luzum (2010). As mentioned above, receiver antenna PCV and GDV should be applied in the user's positioning algorithm for accurate positioning. A particular example of the receiver antenna PCV impact on positioning (i.e. the user's receiver is a smartphone) is analyzed in Chapter 4.

### 2.2.3 SSR corrections

High accuracy and precision techniques like N-RTK (e.g. Wübbena et al., 2001a) and Precise Point Positioning - RTK (PPP-RTK) (e.g. Wübbena et al., 2005) can make use of SSR corrections generated by a network of reference stations. In the traditional PPP technique (Zumberge et al., 1997), the ionospheric effect is eliminated using an ionospheric free linear combination, and the troposphere has to be estimated by the user. Therefore AR is difficult and requires long convergence time. The availability of separate clock, code and phase biases, and atmospheric corrections enables the ambiguity resolution in the so-called PPP-RTK (Wübbena et al., 2005; Khodabandeh and Teunissen, 2015) approach. There are several methods used to perform PPP-RTK, and a mathematical analysis of the approaches is presented in Teunissen and Khodabandeh (2015).

A fundamental concept in the use of SSR corrections is consistency (Schmitz, 2012). This is due to the correlation among state parameters. Firstly, all the corrections need to be generated by the same process to be applied together. Secondly, the generating process must ensure the consistency in time of SSR parameters using defined update intervals. The meaning of the update interval is related to the validity interval of the SSR parameter. In fact, the validity interval of SSR parameters is at least the SSR update interval. The user receiving the correction must acquire the relevant SSR parameters from different update intervals to obtain consistent sets of corrections. The corrections and the required information to be applied have to be transmitted to the user. In the last years, different data formats have been developed, and some of them are discussed in this chapter.

#### State-of-the-art of the SSR format

The increasing demand of high quality and accuracy positioning in various applications makes real-time GNSS data transmissions and formats an essential topic to address. With different requirements concerning accuracy and precision, many applications (e.g. augmented reality, survey, automotive, agriculture, maritime) need real-time GNSS corrections. In particular, the reliability of consistent GNSS corrections is fundamental for real-time N-RTK, PPP and PPP-RTK (e.g. Chen et al., 2003).

The NASA Jet Propulsion Laboratory (JPL) was one of the first providers of a GNSS corrections service for high-precision users (Muellerschoen et al., 2001). Over ten years ago, the International GNSS Service (IGS) started the transmission of satellite and clock products. Currently, there are more than ten analysis centers that provide several real-time corrections. Among others, the analysis centers involved are the Natural Resources Canada (NRCan) (Cerretto et al., 2011), National Centre for Space Studies (CNES) (Laurichesse, 2011), Federal Agency for Cartography and Geodesy (BKG) (Mervart and Weber, 2011), German Aerospace Center (DLR) (Hauschild and Steigenberger, 2011), European Space Agency (ESA) (Pérez et al., 2006), German Research Center for Geosciences (GFZ) (Chen et al., 2010), and Wuhan University (WHU) (Zhang et al., 2018a).

Essential concepts concerning the standards of GNSS corrections transmission are reported in Heo et al. (2009). Crucial components are (Vana et al., 2019): transmission protocol, data format and data communication links. The transmission protocol is a system of rules to ensure secured data delivery through the network. Two standard protocols for distributing GNSS data over the internet are the Networked Transport of RTCM via Internet Protocol (NTRIP) and IGS Real-Time Service (IGS RTS). A data format is a standard specification to follow to decode bit sequences into understandable information for the user. A data communication link is the medium used to transport information.

The Radio Technical Commission for Maritime Services (RTCM) is an international standards organization established to investigate Maritime telecommunication issues. Although started in 1947 as a U.S. government advisory committee, RTCM is now an independent organization supported by its members from all over the world. Concerning GNSS communication, the Special Committee (SC) 104 Differential Global Navigation Satellite Systems (DGNSS) has the main goal to standardize GNSS data transmission (Kalafus, 1996). The latest version of standardized data format is the RTCM SC-104 Version 3.3 (RTCM Special Committee No. 104, 2016) that is partially treated in this chapter and in Appendix A.

One of the main features related to the real-time corrections is the transmission time. Multiple authors investigated the impact of the network processing on the latency and potential outages of the transmission. It has been demonstrated that there is a correlation between latency or missing data due to network positioning solution performance (e.g. Hadas and Bosy, 2015; Elsobeiey and Al-Harbi, 2016; Krzan and Przestrzelski, 2016).

Part of this thesis focuses on the use of low-cost receivers and antennas such as smartphones for satellite-based positioning. For services targeting mass applications (e.g. smartphone-based among others), the data bandwidth associated with the signal power needs to be optimized. These challenging requirements highlight the importance of bandwidth optimization of real-time corrections for navigation purposes.

Besides the bandwidth problem, the slow progress in the RTCM standardization of the SSR messages for PPP-RTK purposes challenged standardization organizations or industry groups to develop an alternative non-RTCM standard (European GNSS Agency (GSA), 2019). In addition to the RTCM format, some private companies developed their own formats. One of these formats is Compact SSR. The latter is a bandwidth efficient format of SSR for PPP-RTK services and has been implemented within QZSS CLAS (Cabinet Office Government of Japan, 2020). Compact SSR is claimed to be significantly more efficient than the proposed RTCM SSR format, requiring a data rate of approximately 71% (Hirokawa et al., 2016). Aiming to provide standards for integrity for safety of life applications, the Safe Position Augmentation for Real-Time Navigation (SPARTN) format (former SAPA format) has been developed. Also, SPARTN supports low bandwidth requirements. A more detailed description of the SPARTN format can be found in Sapcorda Services GmbH (2020); Vana et al. (2019). Another format that has bandwidth optimization as one of its features is the Geo++ SSR format (SSRZ), which is described in Section 2.2.

In this chapter, SSR messages are presented. The RTCM-SSR format is introduced along with the SSRZ format. In particular, the focus is on specific SSR corrections like atmospheric parameters and biases. Further computational details are provided in Appendix A and Appendix B.

### **RTCM-SSR messages**

The RTCM SC104 – SSR Working Group was established in 2007 and defined different phases within their work plan. The standardized SSR messages are described in RTCM Special Committee No. 104 (2016) and summarized in Table A.1 of Appendix A. The RTCM-SSR messages provide the orbit and clock corrections to apply to the orbit computed from the broadcast ephemeris. Due to the linear correlation between the estimated clock and satellite code bi-

ases, the latter need to be transmitted by the SSR corrections provided and applied by the user (e.g. Wübbena, 2012; Wübbena et al., 2014; RTCM Special Committee No. 104, 2016). It is worth observing that the satellite code biases transmitted within the RTCM-SSR format are absolute values, i.e. they must be added to the pseudorange measurements of the corresponding code signal to get corrected pseudoranges as stated in RTCM Special Committee No. 104 (2016). However, the format allows the service provider to use an alternative description of differential code biases by setting one of the biases to zero. It is important to remark that the user must use the correction as absolute. For the sake of clarity, the corrections are indicated with the pre-fix  $\delta$  hereafter. The use of RTCM-SSR corrections enables dual-frequency code positioning.

### Proposed RTCM-SSR messages

Currently, the RTCM-SSR format is still under development and SSR messages have been developed and proposed to enable specific features. Due to the frequency-dependency of the ionospheric delay (see Chapter 2) the use of single-frequency observations does not allow to get rid of the ionospheric delay in absolute positioning. Therefore, an ionospheric message has been proposed to enable code-based single-frequency absolute positioning (Wübbena, 2012; Wübbena et al., 2014; RTCM Special Committee No. 104, c). The message consists of a global multi-layer VTEC message modeled using spherical harmonics. Accordingly, for each layer  $l$ , the VTEC is computed as follows:

$$VTEC_l^{GVI}(\varphi_{IPP}, \lambda_{IPP}) = \sum_{n=0}^N \sum_{m=0}^{\min(n,M)} [A_{nm} \cos(m\lambda_S) + B_{nm} \sin(m\lambda_S)] P_{nm} \sin(\varphi_{IPP}), \quad (2.29)$$

where  $\varphi_{IPP}$  and  $\lambda_{IPP}$  are the latitude and longitude of the IPP for that specific layer. The spherical expansion has a maximum degree  $N$  and maximum order  $M$  and  $P_{n,m}$  are fully normalized associated Legendre functions. The final STEC is defined in the following way:

$$STEC^{s,GVI} = sf \sum_{l=1}^{l_{max}} VTEC_l^{GVI}, \quad (2.30)$$

where  $sf$  is a slant factor depending on the elevation of the satellite  $s$ , and the spherical Earth's central angle between user's position and the projection of the IPP to the spherical Earth's surface. The superscript GVI stays for the Global Vertical Ionosphere. The slant factor is needed to compute the impact on the line-of-sight. The satellite-dependent STEC can then be translated to a frequency-dependent range delay, as described in sub-section 2.1.3. Additional information concerning the global ionospheric correction computation is reported in Appendix A along with a Python script calculation example. As mentioned above, the addition of an ionospheric message enables the single-frequency positioning.

Another objective of the proposed RTCM-SSR messages is to extend RTCM-SSR corrections to other GNSS, i.e. Galileo, QZSS, BeiDou, and SBAS (RTCM Special Committee No. 104, a). The list of the proposed messages is reported in Table A.2. Furthermore, the proposed messages include the *satellite phase bias* message. The addition of the satellite phase bias in the SSR stream enables the single and dual-frequency code and phase bias positioning. Similarly to the satellite code biases, the phase biases must be added to the carrier-phase measurements, as stated in the document RTCM Special Committee No. 104 (b). Again, as in the code biases case, the phase biases are considered as absolute for a specific signal. However, the SSR corrections service provider can choose a differential description of the signal. In that case, the reference signal bias must be transmitted as zero. It is worth reminding that, due to the linear dependency of satellite and receiver biases, the SSR phase biases of a specific signal for all satellites may have a common offset. The latter affects the estimate of the corresponding rover receiver bias but does not harm the solution, as stated in the document RTCM

Special Committee No. 104 (b). In addition to the phase bias, the yaw angle of the satellite's attitude is transmitted within the phase bias message. The reason behind it is to account for the phase wind-up effect properly. The details about the wind-up computation can be found in (e.g. Hauschild, 2017; Wu et al., 1993). Python codes to calculate the wind-up effect from proposed RTCM-SSR messages are included in Appendix A.

It is worth mentioning that, currently, IGS is publishing a standardized format, i.e. the IGS SSR format (International GNSS Service (IGS), 2020), conceptually including the standardized and proposed RTCM-SSR messages. It is an open format, which aims both to support the IGS real-time service and the scientific research and development.

However, the RTCM-SSR and IGS formats do not allow the user to complete the state space corrections. In fact, the atmospheric corrections are missing. A tropospheric and a regional and satellite-dependent description of the ionosphere should be included to perform point positioning with RTK-level accuracy, resulting in the so-called PPP-RTK (Wübbena et al., 2005). A format transmitting a complete set of atmospheric corrections is the SSRZ format, which is introduced in the following section.

### **The Geo++ SSR format (SSRZ)**

SSRZ is the Geo++ SSR format. The content of this section is based on the Geo++ documentation (Geo++ GmbH, 2020) and it is presented to introduce some parameters and terms used in the following chapters (e.g. Chapter 5). The idea of SSRZ is to describe the variations of the SSR parameters in a bandwidth optimized format. SSRZ divides the messages into metadata and corrections. The SSRZ metadata include essential, but static parameters needed to decode SSRZ data. Furthermore, the metadata provide flexibility for service providers to decide on correction model size and correction data resolution. In fact, the resolution of the SSR parameters is dynamic and can be optimized to fit the available bandwidth through different data compression techniques. The metadata can be either offline information or online low-rate data. As the main purpose, SSR correction data is transmitted for GNSS positioning applications. The SSR parameters are grouped into low and high-rate corrections.

**Bandwidth-optimized format concepts** One of the SSRZ requirements is to transmit a data stream with a constant and low bandwidth (bits/second). Compression and bandwidth optimization concepts are:

- separation between SSR correction and metadata messages;
- scaling SSR in the time domain;
- distributing SSR to satellite groups;
- QIX biases, which describe satellite-dependent systematic differences between signals at the same frequency;
- one reference code bias per GNSS;
- rice encoded data blocks. The Rice encoder is used for compression (Rice and Plaunt, 1971).

Further details about the concepts mentioned above can be found in the SSRZ documentation (Geo++ GmbH, 2020). Concerning the bandwidth, tests have been carried out by Geo++ GmbH to assess the performance w.r.t. other open SSR formats. As an example, the SSRZ format can carry the same information (but with higher resolution) of Compact SSR with half of the bandwidth. The Compact SSR bandwidth limitation of roughly 1.6 kbits/s is compared to an SSRZ bandwidth lower than 800 bits/s. The results highlight the potential of the SSRZ format

for scenarios involving mass application with low-cost multi-receivers (e.g. smartphones). In particular, this thesis exploits the concept of multi-stage atmospheric representation.

As the atmospheric GNSS errors vary with the distance, the spatial modeling of atmospheric SSR corrections has to be considered carefully. In SSRZ, atmospheric corrections are divided into functional and residual parts with different spatial scales (Wübbena and Willgalis, 2001; Wübbena et al., 2005). The functional contribution aims to model the atmospheric behavior, while the residual component is the difference between the functional part and the observation. As a consequence, the whole ionospheric correction could be described by the residual component.

**Ionospheric corrections** The SSRZ ionospheric representation is divided into four stages: three functional and a residual. The three functional parts include GVI corrections (as in the proposed RTCM-SSR messages), STEC Satellite-dependent Global Ionosphere corrections (GSI), and Satellite-dependent Regional Ionosphere corrections (RSI). Although the satellite-dependency of the GSI and RSI corrections, they are transmitted as vertical-mapped values for the sake of bandwidth optimization. The vertical-mapped values of the GSI and RSI corrections are described by a two dimensional Chebyshev polynomial expansion as follows:

$$VTEC_l^s(\Delta N_{PP}, \Delta E_{PP}) = \sum_{i,j=0}^{i+j \leq n_{\max}} a_{ij} C_i(\Delta N_{PP}) C_j(\Delta E_{PP}). \quad (2.31)$$

In Eq. 2.31,  $\Delta N_{PP}$  ( $\Delta E_{PP}$ ) is the difference expressed in radians in the projected North (East) direction between the satellite pierce point and the Pierce Point Origin (PPO). The type of projection and the PPO, i.e. the origin of the Chebyshev expansion, are defined differently for GSI and RSI. In the equation above,  $n_{\max}$  is the maximum order of the polynomial,  $a_{ij}$  the polynomial coefficients and  $C_n(\cdot)$  a Chebyshev polynomial of order  $n$ . The Chebyshev polynomials have been chosen for multiple reasons. In terms of bandwidth, the Chebyshev polynomials allow the transmission of a single resolution for the coefficients (since Chebyshev polynomials are defined in  $[-1,1]$ ), while a resolution term for each coefficient of an algebraic polynomial expansion would be needed. Also, the use of Chebyshev polynomials for continuous functions allows us to get rid of Runge's phenomenon. The latter causes possibly large oscillations at the edges of an interval when using polynomial interpolation with polynomials of a high degree over a set of equispaced interpolation points. More information about the Chebyshev polynomial expansion can be found in, e.g., Gil et al. (2007).

The definition of the PPO characterizes the satellite dependency of the GSI. The PPO is the PP of the nadir direction of the satellite w.r.t. the spherical ionospheric layer. The regional dependency of the RSI correction is instead given by choosing the PPO as a ground point origin GPO. The ellipsoidal coordinates of the GPO are provided in the metadata. The layer  $VTEC_l$  is mapped to STEC by meaning of the  $sf$  as in Eq. 2.30. The sum of all the layers gives the total functional component of the ionosphere.

As mentioned above, the final stage of the multi-stage ionosphere model employed by SSRZ is the Gridded Ionosphere correction GRI. The GRI correction is STEC mapped to VTEC and it is defined for grid points whose ellipsoidal coordinates are provided in the corresponding metadata messages. The GRI correction needs to be interpolated for the user's location. Finally, the ionospheric correction can be defined as:

$$\delta I_j^s = \delta I_j^{s,GVI} + \delta I_j^{s,GSI} + \delta I_j^{s,RSI} + \delta I_j^{s,GRI}. \quad (2.32)$$

**Tropospheric corrections** In a similar way to the ionosphere, the SSRZ multi-stage representation is applied to the troposphere. The tropospheric delay is described either as the sum of wet and dry components (see Chapter 2) or as a total component. Each component (dry, wet, or total) is modeled as the sum of a global troposphere, a Regional Troposphere (RT), and

a residual Gridded Troposphere (GRT). Currently, the SSRZ format does not support a global troposphere message, but it will be introduced in the next version of the format. Hence, here, it is not considered. In the SSRZ format, the zenith value (e.g. the ZWD for the wet component) is transmitted and VGMF are considered for each component. Furthermore, for the sake of bandwidth optimization, only a tropospheric scale factor of the model is transmitted. The model considered is the Saastamoinen tropospheric model (Saastamoinen, 1972). For the computation of the Saastamoinen model and VGMF refer to Petit and Luzum (2010). As an example, for the RT, the scale factor is defined as:

$$t_c^{RT} = \frac{\delta T_c^{Z,RT}}{\delta T_c^{Z,model}}, \quad (2.33)$$

where  $c$  indicates the component (e.g. wet) and  $Z$  indicates that it refers to the zenith delay.

The RT is computed as horizontal Chebyshev and vertical algebraic polynomial expansion as follows:

$$t_c^{RT}(\Delta\lambda, \Delta\gamma, \Delta h) = \sum_{k=0}^{N_k-1} \sum_{l=0}^{N_l-1} \sum_{m=0}^{N_m-1} a_{klm}(\Delta h)^k C_l(\Delta N) C_m(\Delta E), \quad (2.34)$$

where  $\Delta N$  and  $\Delta E$  are the difference in radians between rover and GPO North and East components. Concerning the height component,  $\Delta h$  is expressed in meters. The GPO is defined similarly to the ionospheric case.

As in the ionosphere modeling, a final residual stage is defined over a grid that is described within the metadata messages. The interpolated value for the user's location completes the tropospheric corrections. It is worth observing that the user is free to choose the preferred interpolation technique. Finally, the SSR tropospheric correction can be written as:

$$\delta T^s = \sum_c (\delta T_c^{RT} + \delta T_c^{GRT}) \text{VGMF}_c(\varepsilon), \quad (2.35)$$

where  $\varepsilon$  is the elevation of the satellite  $s$ . In the current SSRZ version,  $\delta T_c^{GRT} = 0$  since it is not included in the messages yet.

The reader is referred to Geo++ GmbH (2020) for a complete description of the SSRZ format. For demonstration purposes, Python code to compute GSI, RSI, RT and GRT corrections are provided in Appendix B.

## 2.2.4 Observation Space Representation (OSR)

The Observation space representation (OSR) is related to the representation of the GNSS errors in the observation domain. As indicated by the name, it means that the errors are projected in the observation space, i.e. along the line-of-sight given by the geometry between satellite and receiver. Each effect of the single GNSS error sources is summed together, yielding an overall effect for each observation. Therefore, the OSR approach deals with the handling of the effects of the errors, while the SSR approach is modeling the source of the errors (Wübbena et al., 2005). An example of OSR application is RTK positioning (see Section 2.3). Also, SSR corrections can be translated into OSR corrections. This transformation is often used by RTK rover algorithms. In fact, many of such rovers still work with OSR corrections, which include the sum of all the effects.

For a user location, the SSR parameters can be projected on the line of sight and their influence on the user's observations can be computed. For a specific signal of frequency  $j$  (e.g. C1C using the RINEX conventions), receiver  $r$  and satellite  $s$ , the corrected pseudorange is computed in the following way:

$$\tilde{p}_{r,j}^s = p_{r,j}^s + \delta p_{r,j}^s, \quad (2.36)$$

where  $\delta p_{r,j}^s$  is the SSR influence on the user's location computed using SSR corrections. Considering RTCM-SSR messages, the SSR influence is computed as follows:

$$\delta p_{r,j}^s = \delta O^s + \delta C^s + \delta CB_j^s, \quad (2.37)$$

where  $\delta O^s$  is the satellite-dependent orbit correction projected on the line-of-sight,  $\delta C^s$  the satellite-dependent clock correction and  $\delta CB_j^s$  the satellite and signal-dependent code bias correction.

Furthermore, including the proposed RTCM-SSR global ionosphere message, the SSR influence described by Eq. 2.37 can be modified as:

$$\delta p_{r,j}^s = \delta O^s + \delta C^s + \delta CB_j^s + \delta I_j^{s,\text{GVI}}, \quad (2.38)$$

where  $\delta I_j^{s,\text{GVI}}$  is the correction term for the frequency-dependent ionospheric delay computed from the global STEC obtained from Eq. 2.30.

Also, adding the phase bias correction, similarly to Eq. 2.38, the carrier phase correction can be written as:

$$\delta \phi_{r,j}^s = \delta O^s + \delta C^s - \delta I_j^{s,\text{GVI}} + \delta PB_j^s + \delta W_j^s, \quad (2.39)$$

where  $PB_j^s$  indicates the phase bias for a specific signal of frequency  $j$  and  $\delta W_j^s$  the related wind-up effect.

The complete state space can be projected into the OSR domain using the SSRZ format. In that case, the pseudorange and carrier phase corrections yield:

$$\delta p_{r,j}^s = \delta O^s + \delta C^s + \delta CB_j^s + \delta I_j^s + \delta T^s, \quad (2.40)$$

$$\delta \phi_{r,j}^s = \delta O^s + \delta C^s - \delta I_j^s + \delta T^s + \delta PB_j^s + \delta W_j^s. \quad (2.41)$$

In addition to the generation of corrected observations, the SSR corrections allow to generate SSR-based observations to use as observations of a non-physical station as will be introduced in Section 2.4. The pseudorange and carrier phase observations can be described as follows:

$$p_{\text{SSR},j}^s = \rho_{\text{BR},j}^s - \delta p_{r,j}^s, \quad (2.42)$$

$$\phi_{\text{SSR},j}^s = \rho_{\text{BR},j}^s - \delta \phi_{r,j}^s, \quad (2.43)$$

where  $\rho_{\text{BR},j}^s$  is the range computed using the broadcast ephemeris.

As mentioned above, the user needs to select an interpolation technique to interpolate the atmospheric corrections for their location. In Chapter 5, an analysis of different interpolation approaches is carried out, assessing the potential impact of an interpolation error into SSR-based positioning. Furthermore, in Chapter 5, the SSRZ format is exploited to evaluate the impact in the positioning performance of an ionospheric mismodeling.

## 2.3 RTK positioning

In general, the determination of the absolute position is less accurate than the relative positioning between two stations. This is due to the high correlation among the acting errors (Seeber, 2003). To minimize the impact of errors decorrelated with distance, the coordinates are estimated w.r.t. a known reference station. RTK is the technique that utilizes code and phase observations from the reference station in real-time. RTK involves the resolution of ambiguities at the rover station *on the fly* (i.e. the receiver antenna is in motion) for reliable determination of the baseline vector in real-time. For the estimation of the baseline vector between the base and the rover station, either the concept of parameter elimination or parameter estimation (undifferenced approach, see Section 2.4) can be used (e.g. Seeber, 2003).

In the typical RTK application, the user receives the pseudorange and carrier phase corrections transmitted by a nearby physical reference station in observation space. The RTK user obtains their corrected observations by applying the reference station's corrections. RTK is a powerful technique that allows the user to achieve centimeter-level positioning accuracy in real-time over short distances. Hereafter, RTK-level positioning refers to centimeter-level accuracy. For more details about RTK see e.g. Langley (1998b); Seeber (2003).

One of the major issues related to RTK applications is that the influence of some errors, like orbit, ionosphere and troposphere, grows with the distance from the reference station. Furthermore, the impact of the atmospheric error sources depends on solar activity and weather conditions, which makes the definition of a maximum favorable distance complicated. In order to overcome this distance-dependent effect, the network-RTK (N-RTK) technique has been developed. The CORS availability allows the estimate of the state space for the area covered by the network of reference stations. As a consequence, corrections for a user within the area covered by the network can be generated and transmitted. Commonly used procedures for N-RTK are the concept of area correction parameters (FKP from the German word *Flaechen Korrektur Parameter*), Master-Auxiliary-Concept (MAC) and Virtual Reference Station (VRS).

The FKP approach represents additional corrections for the distance-dependent errors by utilizing a polynomial parametrization to describe the influence of any rover position in a certain area. These corrections are transmitted in addition to the range corrections of the reference station considered.

The MAC approach consists in the transmission of observation data of a master station and correction differences between master and auxiliary stations. The rover can re-construct the observation data of the auxiliary stations (except a common clock term) and decide how to use master and auxiliary data for its location.

The VRS approach needs a two-way communication link between rover and network. The rover communicates their approximate location to the network, which sends the corrections for the received position. Accordingly, users can correct their observations and generate a VRS. The classic VRS approach involves OSR corrections (i.e. range corrections). However, state-of-the-art applications consider SSR corrections that can be translated into OSR (see subsection 2.2.4). For more details concerning N-RTK see e.g. Weber and Tiwari (1995); Wübbena et al. (1996); Wübbena et al. (2001a,b). Again, as for the RTK technique, the positioning algorithm can work in differenced or undifferenced mode. DD observations (see subsection 2.1.4) are often used in the differenced mode for parameter elimination.

## 2.4 Positioning with uncombined observation model

The uncombined observation model is defined as a model where no combination or differentiation of observations (e.g. double difference) is considered. This is a general definition that is applied for the estimation of the absolute coordinates of the user's location. The main concept behind positioning using the uncombined observation model considered in this thesis is related to the state space approach presented in Section 2.2. Further details can be found in Wübbena (1985), De Jonge (1998) and Wübbena and Willgalis (2001). Two cases are reported:

- positioning with reference station observations;
- positioning with SSR parameters for the user's location.

In the first case, the uncombined model of the observation equation is computed in the following way:

$$\tilde{\mathbf{y}}_k = \begin{bmatrix} \mathbf{y}_k \\ \mathbf{y}_{1,k} \\ \vdots \\ \mathbf{y}_{r,k} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_k \\ \mathbf{H}_{1,k} \\ \vdots \\ \mathbf{H}_{r,k} \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \mathbf{x}_{1,k} \\ \vdots \\ \mathbf{x}_{r,k} \end{bmatrix} + \begin{bmatrix} \mathbf{v}_k \\ \mathbf{v}_{1,k} \\ \vdots \\ \mathbf{v}_{r,k} \end{bmatrix}, \quad (2.44)$$

where  $r$  is the number of reference stations available. An essential feature is that the coordinates of the reference stations are known. Furthermore, in the case of only one reference station, it can be seen as an undifferenced approach of RTK. If the reference station is close

enough (e.g. roughly within 10 km), the atmospheric parameters (i.e. troposphere and ionosphere) could be considered the same, reducing the number of parameters to estimate.

In the second case, the main idea is the same but the observations are generated using SSR corrections. Two sub-cases can be distinguished:

- physical GNSS rover observations are corrected using SSR parameters generating a VRS (Wübbena et al., 2001a);
- non-physical GNSS observations are generated using SSR parameters.

Using SSR corrections the uncombined observation model yields:

$$\tilde{\mathbf{y}}_k = \begin{bmatrix} \mathbf{y}_k \\ \mathbf{y}_{\text{SSR},k} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_k \\ \mathbf{H}_{\text{SSR},k} \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \mathbf{x}_{\text{SSR},k} \end{bmatrix} + \begin{bmatrix} \mathbf{v}_k \\ \mathbf{v}_{\text{SSR},k} \end{bmatrix}, \quad (2.45)$$

where the subscript SSR indicates the VRS or non-physical station computed by using the SSR corrections.

## 3. Smartphone-Based Positioning

---

### 3.1 Motivation

The access to smartphone GNSS raw measurements, introduced to the Android OS in 2016, started a dynamic and fast innovation of new low-cost applications for satellite-based positioning. Even before Google provided the open access to the raw data, the feasibility to perform positioning using low-cost receivers and smartphone antennas was investigated. In their work, Pesyna et al. (2014) and Humphreys et al. (2016) showed that smartphones are capable of cm-level GNSS positioning, though highlighting the challenge in the ambiguity resolution. The most critical issue is related to the employed GNSS antenna. The smartphone antenna uses linear polarization with inhomogeneous gain and high levels of local multipath, causing large and hard to predict phase errors (Humphreys et al., 2016). Supported by the release of the white paper from the European GNSS Agency (GNSS Raw Measurements Task Force – European GNSS Agency (GSA), 2017), many authors analyzed the quality of the raw measurements retrieved from smartphones. In their investigation, they faced smartphone-related issues, like the duty cycle mechanism and low C/N0 (Gogoi et al., 2019; Liu et al., 2019). Furthermore, the variation of the signal to C/N0 regardless of the elevation made researchers develop C/N0 weighting schemes, (Zhang et al., 2018b).

An example of a state-of-the-art smartphone System on Chip (SoC) chipset (Dabove and Di Pietra, 2019) was tested in Tomaščík et al. (2016), where the integrated GNSS-only position solution had an accuracy of 25 m for applications in forests (i.e. challenging environment). In the same work, the authors showed that the so-called Assisted GNSS (A-GNSS) solution, reached accuracies of 5 m. In A-GNSS, predicted ephemeris data (broadcast using mobile networks) are used to eliminate sections of the signal search space, reducing the time to compute a positioning solution. Recent works, (e.g Dabove and Di Pietra, 2019; Realini et al., 2017), performed decimeter-level accuracy using relative positioning techniques, showing the possibility to achieve cm-level accuracy with smartphone measurements. These results are supported by Pirazzi et al. (2017), who, employing a variometric approach, show decimeter accuracy in static condition and sub-meter when used in an urban vehicle scenario.

Another milestone has been set by Broadcom, announcing on September 21, 2017, the world's first mass-market, dual-frequency GNSS receiver device, the BCM47755. In May 2018, the Xiaomi Mi8 (Mi8) became the first smartphone in the world, employing a dual-frequency GNSS receiver L1/E1 – L5/E5a. This led to the next series of studies in the investigation of smartphone-based positioning. At the time of writing, to the best of the author's knowledge, there are 41 smartphone models on the market, from 10 manufacturers, offering dual-frequency capability. The number is increasing continuously. For example, one of the recent launches by a major manufacturer was the Galaxy Note10 and Galaxy Note10+ from Samsung Electronics, which hit the market in the second half of 2019. These smartphones are fitted with the BCM47755, like the Huawei Mate20X (Mate20X). Huawei also developed its chipset, embedded in, for example, the Huawei P30. Moreover, Qualcomm developed another chipset that is employed, e.g., in the Google Pixel 4 and Pixel 4XL. More information about dual-frequency smartphone capability can be found at the *UseGalileo* webpage provided by GSA (GSA, 2020).

Multiple authors tried to exploit the availability of GNSS measurements in these cheap and handy receivers. Robustelli et al. (2019) analyzed GPS and Galileo dual-frequency Xiaomi Mi8 measurements in different multipath environments, showing meter-level Single Point Posi-

tioning (SPP) as promising results for the low-cost solution of smart cities challenges. A PPP solution using dual-frequency measurements collected with the Mi8 has been investigated in Wu et al. (2019). Using the time series differential algorithm with C/N0 dependent weighting method developed in Zhang et al. (2018b), the work carried out in Wu et al. (2019) highlights the PPP sub-meter capability using a smartphone without external equipment. However, the unfixed integer ambiguity implies a long convergence time to achieve 1 m accuracy, i.e., more than one hour and a half. Other PPP performance has been reported by the research group of Calgary (Canada) in collaboration with the Canadian Geodetic Survey, Natural Resources Canada (NRCan), using the Mate20X (Lachapelle and Gratton, 2019; Banville et al., 2019). The authors showed that the PPP solution could converge to cm-level accuracies under favorable signal tracking conditions.

Several researchers investigated the potential of ambiguity resolution with a smartphone receiver, using an external GNSS antenna. For example, Geng and Li (2019) show that with a Mi8 smartphone coupled with an external antenna, it is possible to obtain a reliable ambiguity-fixed solution. A different approach to achieve ambiguity resolution with smartphones is to use the smartphone antenna directly with the phone placed in specially designed scenarios with highly reduced multipath. An example is provided in Sharma et al. (2019), where the authors show results with successful ambiguity resolution when the smartphone is undergoing a slow circular motion.

New research opportunities started in January 2020, when a new Broadcom chipset, the BCM47765 (Broadcom, 2020), has been announced in production. The new chipset is capable of using the new BeiDou-3 constellation's B2a signals (the Chinese indicator for L5). It is able to track 30 new L5 signals (60 percent more) with an expected significant impact on accuracy (Cozzens, T., 2020).

In this chapter, a method to construct the GNSS observables from Android raw measurements is presented. Furthermore, a technique to deal with the continuity of the measurements is proposed. Also, an investigation of the quality of the measurements is carried out. The quality assessment has been made through double-difference and multipath sensitivity investigations. Finally, the potential for GNSS smartphone-based positioning is analyzed.

## 3.2 The construction of the observations

In an Android smartphone, the GNSS pseudorange and carrier phase observables can be reconstructed by making use of two classes of the Android location API:

- GNSSClock, which provides the receiver time and clock biases;
- GNSSMeasurements, which provides the received satellite time and the Accumulated Delta Range (ADR) as phase measurement.

The API fields of the GNSSClock and GNSSMeasurement classes used for the observation computation are reported in Appendix C in Table C.2. Considering the two API classes, the observables can be derived and written in standardized formats, e.g. RINEX (see <ftp://ftp.igs.org/pub/data/format/rinex304.pdf>). In the last years, many companies and researchers have developed Android apps that can log the measurements in RINEX format, e.g. the Geo++@RINEX Logger (Geo++ GmbH, 2017) and rinex ON (Nottingham Scientific Ltd, 2018). In this work, a strategy to compute and write RINEX code and phase observations is proposed. A procedure to compute pseudorange and carrier phase can also be found in GNSS Raw Measurements Task Force – European GNSS Agency (GSA) (2017).

### 3.2.1 Pseudorange calculation

The pseudorange is computed following the principle described by Eq. 2.1. In the case of the smartphone, the acquisition time,  $t_A$ , and emission time,  $t_E$ , are constructed considering the GNSSClock class that provides receiver-related quantities, and the GNSSMeasurement class (see Table C.2). While there is direct access to the received satellite time, the computation of the receiver time requires more attention, and it is explained later in this section. Concerning the phase observations, the API provides the ADR in meters, which corresponds to the phase measurements when the PLL is locked. The conversion from ADR to phase observable needs to be considered carefully, and the procedure is described in this section as well.

#### Receiver acquisition time

The receiver time is generated considering the TimeNanos quantity, which is the GNSS receiver's internal hardware clock provided as an integer number of nanoseconds (see Table C.2). To get the true GPS time, the TimeNanos needs to be corrected subtracting the difference between the TimeNanos inside the GPS receiver and the true GPS time since the 6th of January 1980, i.e. the so-called FullBiasNanos. GNSS Raw Measurements Task Force – European GNSS Agency (GSA) (2017) and Google in the GNSS analysis tool available online (Google, 2020) suggest considering only the first value of the FullBiasNanos (FBN) and BiasNanos (BN). The two values are supposed to remain constant till the next discontinuity (e.g. restarting the smartphone/tracking). However, it has been observed that, depending on the device and firmware, especially the FBN can experience some significant jumps (up to 10 ms). Therefore, here, it is suggested to update it every epoch. Thus, the receiver GPS time (GT) can be computed as follow:

$$t_{Rx,GT,integer} = \text{TimeNanos} - \text{FBN}, \quad (3.1)$$

where all the quantities are expressed in nanoseconds. Moreover, the API provides the BiasNanos (BN), i.e. the clock's sub-nanosecond bias (varying between 0 and 1), which allows getting a more accurate timing. It has been observed that in many devices (e.g. Xiaomi Mi8, Huawei Mate 20X), the BN is equal to zero. However, the handling of the API quantities depends on the smartphone manufacturer. As a consequence, it could change with a firmware update. Therefore, it is worth taking it into account.

$$t_{Rx,GT,float} = t_{Rx,GT,integer} - \text{BN}, \quad (3.2)$$

#### Satellite emission time

The received satellite time (ReceivedSvTimeNanos, see Table C.2) is relative to the beginning of the system week for all constellations except for GLONASS, where it is relative to the beginning of the GLONASS system day. Therefore, depending on the GNSS involved, the condition of reliability of the received satellite time is satisfied in the following cases:

- GPS, Galileo, and BeiDou: STATE TOW KNOWN constant flag is set.
- GLONASS: STATE GLO TOD KNOWN constant flag is set.

The constant values of the state flags are report in Appendix C in Table C.1. These flags provide an insight into the state of the tracking algorithms. The status of the tracking algorithms needs to be taken into account to verify the reliability of the incoming measurement. It's worth mentioning that some devices track the Galileo E1C component (pilot component), and the tracking status is flagged as the STATE GAL E1C 2ND CODE LOCK (see Table C.1). In this case, the ambiguity of the pseudorange is 100ms, and it should be taken into account. However, to the

best of the author's knowledge, the smartphone GNSS receiver seems to resolve the ambiguity automatically. Therefore, the emission time is computed in the following way:

$$t_{E,TOW} = \text{ReceivedSvTimeNanos}, \quad (3.3)$$

in GPS Time Of the Week (TOW) for GPS, Beidou and Galileo, and

$$t_{E,TOD} = \text{ReceivedSvTimeNanos}, \quad (3.4)$$

in GPS Time Of the Day (TOD) for GLONASS.

Once the receiver and satellite time have been computed, the pseudorange observation can be reconstructed. Receiver and satellite time as computed in Eq. 3.2 and Eq. 3.3 (or Eq. 3.4) are expressed in GPS time and GPS TOW/TOD, respectively. Thus, this inconsistency needs to be solved. There are two options:

- to compute the receiver time in GPS TOW;
- to compute the satellite time in GPS time.

In the first case, the receiver acquisition time is:

$$t_{Rx} = \text{mod}(t_{Rx,GT,float}, \text{NanoSecondsWeek}), \quad (3.5)$$

for GPS and Galileo,

$$t_{Rx} = \text{mod}(t_{Rx,GT,float}, \text{NanoSecondsWeek}) + 14 \times 10^9, \quad (3.6)$$

for BeiDou and

$$t_{Rx} = \text{mod}(t_{Rx,GT,float}, \text{NanoSecondsDay}) + (7200 - ls) \times 10^9, \quad (3.7)$$

for GLONASS. In Eq. 3.7  $ls$  are the current leap seconds. In the second case, the satellite time is adjusted in the following way:

$$t_E = t_{E,TOW} + n_W \text{NanoSecondsWeek}, \quad (3.8)$$

for GPS and Galileo,

$$t_E = t_{E,TOW} + n_W \text{NanoSecondsWeek} + 14 \times 10^9, \quad (3.9)$$

for BeiDou and

$$t_E = t_{E,TOD} + \text{NanoSecondsDaySince1980} - (7200 - ls) \times 10^9, \quad (3.10)$$

for GLONASS. In the equations above,  $n_W$  is the number of weeks since 6th January 1980. Besides, in Eq. 3.10,  $\text{NanoSecondsDaySince1980}$  is the days since 6th January 1980 in nanoseconds. Furthermore, for each observation, the time offset at which the measurement was taken w.r.t. the TimeNanos has to be considered. This quantity is given by the  $\text{TimeOffsetNanos}$  (see Table C.2). Accordingly for a specific measurement, a certain epoch, a receiver  $r$ , a satellite  $s$  and frequency  $j$ , the acquisition time is:

$$t_A = \text{TimeOffsetNanos}_s^j + t_{Rx} \quad (3.11)$$

Finally, the pseudorange observation can be computed in meters in the following way:

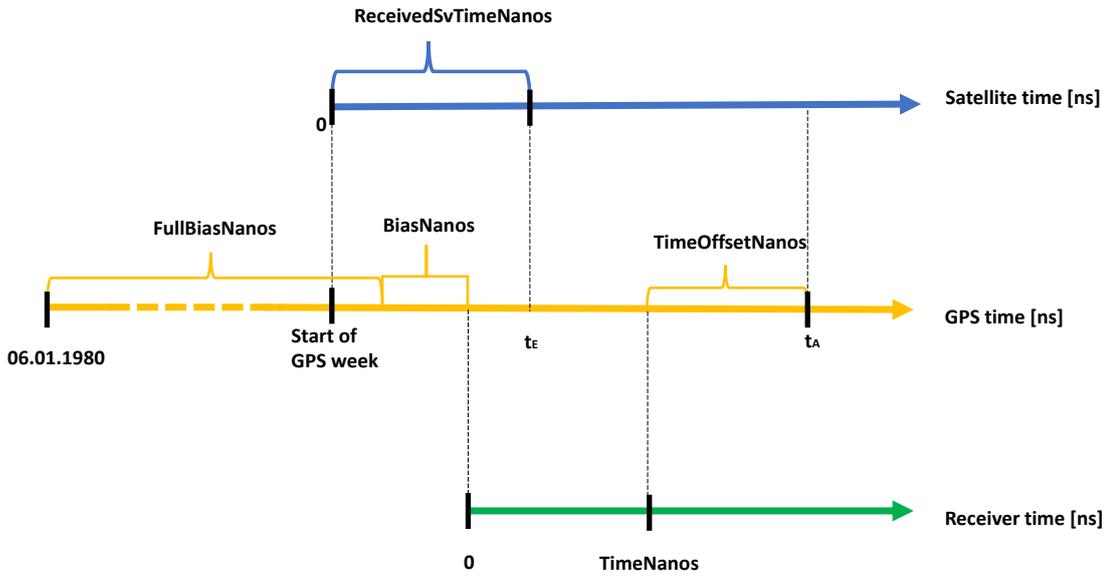
$$p_{r,j}^s(t_A) = c \Delta t_{r,s}^j \times 10^{-9}, \quad (3.12)$$

where  $\Delta t_{r,s}^j$  is the difference between  $t_A$  and  $t_E$ . Fig. 3.1 depicts the construction of the acquisition and emission time using GNSS measurements. As an example, the GPS case is reported where no leap seconds are considered and the ReceivedSvTimeNanos refers to the GPS TOW.

It might be useful for many applications to log the data in a standardized format, e.g. the RINEX format. In such a case, the resolution of the RINEX time-stamp of 100 ns needs to be taken into account. Therefore, a rounding adjustment has to be performed for the observations:

$$p_{r,j}^s(t_A) = c(\Delta t_{r,j}^s + \Delta_{\text{round}}) \times 10^{-9}, \quad (3.13)$$

where  $\Delta_{\text{round}}$  is the difference between  $t_A$  and its rounded value to 100 ns resolution.



**Figure 3.1:** Representation of the acquisition ( $t_A$ ) and emission ( $t_E$ ) time reconstruction using Android GNSS raw measurements. As an example, the GPS case is reported, where there is no leap seconds to consider. The satellite time (blue-colored axis) and receiver time (green-colored axis) are compared to the GPS time (yellow-colored axis).

### 3.2.2 Phase observation computation

The phase measurements can be retrieved from the API considering the ADR (see Table C.2). The ADR is produced by the PLL tracking loop (Shade and Madhani, 2018). Here, a method to calculate the phase observation is proposed. As already mentioned, in this work, the FBN is updated every epoch. Therefore, to get a consistent value of ADR with the pseudorange, the ADR needs to be adjusted for the difference between the current FBN (BN) and the initial FBN (BN) as shown in Fig. 3.2. Fig. 3.2 compares ADR computed from Android API with constant and updated FBN. The initial ADR value is adjusted to match the initial pseudorange value to assess the ADR evolution compared to the pseudorange. Fig. 3.2 shows an example of data gathered during hour 8 at DOY 352 of the year 2019, analyzing the L1 measurements of the GPS satellite G09. It can be observed that the pseudorange (PR) and ADR are coherent when either the first FBN value is kept constant or the FBN is updated every epoch and the ADR is adjusted accordingly. In such cases, the difference between ADR variation and pseudorange variation does not exceed a few meters, as also indicated in the example reported in Fig. 3.2.

The adjusted value of the phase, i.e. the adjusted ADR, can be written in cycles as follows:

$$\begin{aligned} \widetilde{ADR}_{r,j}^s(t_A) = & \frac{1}{\lambda_j} ADR_{r,j}^s(t_A) - \\ & \frac{c}{\lambda_j} [(FullBiasNanos(t_A) - FullBiasNanos(t_0)) - \\ & (BiasNanos(t_A) - BiasNanos(t_0))], \end{aligned} \quad (3.14)$$

where  $t_0$  is the start of the tracking period. The adjusted value of the ADR gives the carrier phase measurement that can be retrieved from smartphones. As mentioned above, it might be helpful to store the raw measurements in RINEX format. Some devices output very large ADR numbers that exceed the range defined by the RINEX format. To deal with that, the so-called *phaserange* is computed, removing the proper integer number of cycles from the original full carrier to match the corresponding pseudorange at the start of the phaserange generation (RTCM Special Committee No. 104, 2016). Accordingly, an Initial Phase Ambiguity (IPA) is computed to align the phase measurement with the code measurement:

$$IPA_{r,j}^s(t_A) = \text{int} \left[ \frac{1}{\lambda_j} \left( p_{r,j}^s(t_A) - \widetilde{ADR}_{r,j}^s(t_A) \right) \right], \quad (3.15)$$

where  $\text{int}$  is an integer operator that outputs a rounded integer value. The IPA is kept constant while the phase is continuously tracked. Eventually, the phaserange (i.e. the carrier phase observation considered in the RINEX) can be computed as:

$$\phi_{r,j}^s(t_A) = IPA_{r,j}^s(t_A) + \widetilde{ADR}_{r,j}^s(t_A), \quad (3.16)$$

where  $\phi_{r,j}^s(t_A)$  is expressed in cycles. The corresponding value in meters can be obtained by multiplying  $\phi_{r,j}^s(t_A)$  by the correct  $\lambda_j$ .

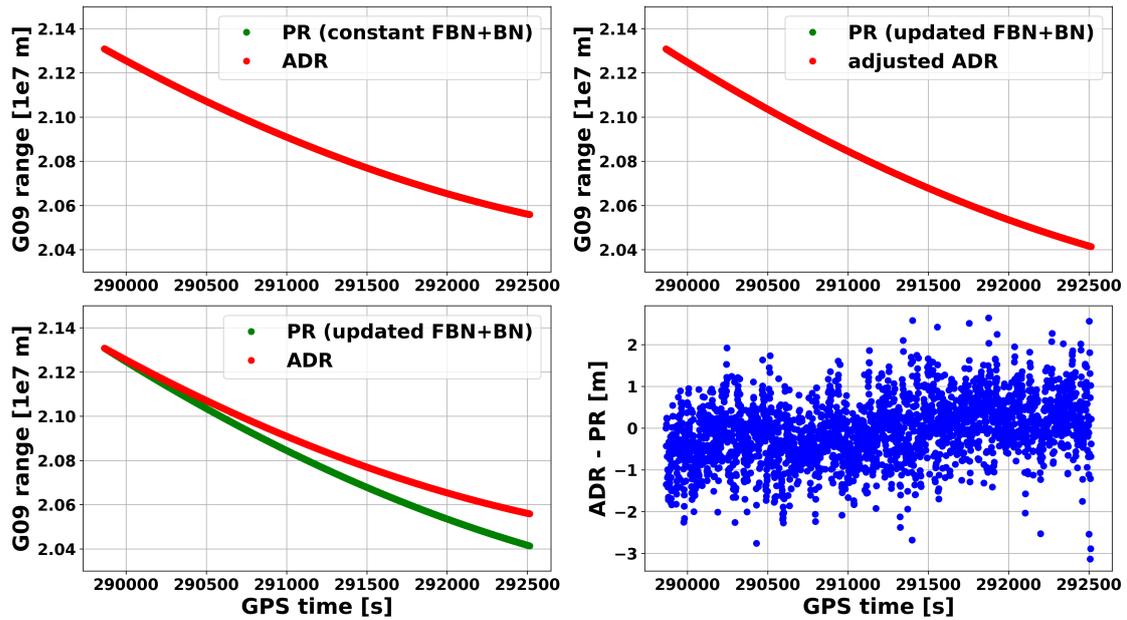
It is worth remarking that over time certain ionospheric conditions may cause a divergence between phaserange and pseudorange. In that case, a new IPA should be re-initialized. The continuity of the phase has to be taken into account carefully. The strategy employed in this work is reported in the following sub-section.

### 3.2.3 Continuity of phase measurements

One of the most important aspects to consider when using smartphone GNSS phase measurements is the continuity of the observations. To avoid heavy power consumption, smartphone manufacturers started using the duty cycle technique. The adoption of such a method implies that the carrier phase is tracked only for a short period, shutting down the tracking for the following period, yielding non-continuous GNSS carrier phase observations. Since Android 9.0, the feature “Force full GNSS measurements” has been introduced for developers to shut down duty cycling. This allows us to significantly increase data availability and the continuous nature of the phase measurements (Wu et al., 2019). Nevertheless, this feature is not available on some smartphones (e.g., the Huawei Mate20X). However, it has been tested that phase measurements can be continuously collected regardless, and the duty cycle is therefore assumed to be off during the GNSS observation period (Banville et al., 2019).

The Android API allows chipset manufacturers to provide raw GNSS measurements with various levels of synchronization (e.g. with or without millisecond ambiguity resolved, with or without half-cycle ambiguity resolved). Additionally, two flags indicate whether multipath was detected and whether cycle slips occurred. This information can be used to filter out some of the measurements before processing them with the positioning algorithms. The strategy adopted for the results in this work is introduced in this section.

Accept a measurement for further processing if:



**Figure 3.2:** Comparison of ADR computed from Android API with constant and updated FBN. The pseudorange (PR) and ADR variations in time are compared. The ADR initial value has been set equal to the PR initial value in order to evaluate the ADR evolution. Top-left panel: the PR is generated keeping the FBN constant, and the ADR is not adjusted. Top-right panel: the PR is generated updating the FBN every epoch and the ADR is adjusted for the corresponding FBN difference. Bottom-left panel: the PR is generated updating the FBN every epoch, while the ADR is not adjusted for the FBN difference. Bottom-right panel: difference between adjusted ADR and PR generated with updated FBN. The difference amounts to a few meters and is the same obtained when the FBN is kept constant. It can be observed that when the FBN is updated every epoch, the ADR needs to be adjusted for the difference between the current and initial FBN to be consistent with the PR. Here, as an example, almost one hour of data during hour 8 of DOY 352 of the year 2019 is analyzed. The observations are L1 measurements of the GPS satellite G09.

- STATE TOW KNOWN flag is set
- uncertainty defined, i.e. `getReceivedSvTimeUncertaintyNanos` is not 1 billion, which is used by some chips to mark unreliable measurements
- ADR STATE VALID is set.

Report a likely phase discontinuity if:

- ADR STATE CYCLE SLIP is set
- MULTIPATH INDICATOR DETECTED is set
- the signal was not tracked in the last epoch
- ADR STATE VALID was not set in the last epoch
- the phase range was adjusted (see below).

Adjust the phase range by an integer number of wavelengths so that it matches the code pseudorange as close as possible if:

- the satellite is measured for the first time

- the code and phase range difference is larger than 50 m (value experimentally set).

The strategy mentioned above has been successfully tested with eight different smartphone devices with different firmware: Xiaomi Mi8, Samsung S9, Samsung S8, Huawei P10, Huawei Mate 9, HTC Nexus 9, Huawei Mate20X, and Pixel 4. More details about the Android API flags can be found in Appendix C.

A statistical investigation of the correlation between the ADR-related flags and C/N0 has been carried out in Massarweh et al. (2019). In that work, the authors carried out a statistical analysis of raw GNSS measurements collected using a Mi8 equipped with the BCM47755 chipset. The data-set involves more than 200 hours of Android raw measurements gathered between the 28th of May and the 21st of June 2019. The observations have been collected in an open sky environment on the rooftop of the Geo++ building (rooftop depicted in Fig. 3.3) using a setup corresponding to Scenario 3 of Table 3.1.

One of the outcomes of the investigation is that the chipset reports a multipath flag (MULTIPATH INDICATOR DETECTED) in less than 10% of the GPS and Galileo, L1 and L5 measurements (see Fig. 8 in Massarweh et al. (2019)). Furthermore, the multipath is detected mostly for C/N0 values lower than 30 dBHz. Information about the multipath indicator was provided in Shade and Madhani (2018). The authors affirmed that the multipath indicator provided is based mainly on the shape of the correlation vector. The indicator is not using any information from the chipset navigation solution to determine this flag. If the measurement is flagged, there is a high probability that the measurement contains significant multipath. Nevertheless, if multipath is marked as unknown, there may still be multipath, but the measurement level checks did not detect it.

Also, it seems that the ADR STATE VALID is set only for a certain range of signal strength, depending on the frequency and constellation (see Fig. 9 in Massarweh et al. (2019)). For example, for GPS L1 signals, a valid ADR is reported only for C/N0 greater than 23 dBHz, while for Galileo L1 signals, valid ADR data are given only for C/N0 larger than 15 dBHz. Another aspect highlighted in Massarweh et al. (2019) is that the ADR STATE CYCLE SLIP is set mainly for C/N0 smaller than 30 dBHz, as can be observed in Fig. 10 in Massarweh et al. (2019). A similar C/N0 dependency has been reported, e.g., in Liu et al. (2019), where a safer C/N0 mask of 30 dBHz has been employed to avoid bad quality measurements.

### 3.3 Quality analysis of smartphone measurements

In this section, a quality assessment of dual-frequency Android measurements is performed. The data has been gathered following the process described in Section 3.2. Two combinations of observations have been used to evaluate the quality of phase and code measurements: DD and multipath combination. The DD provides information about the impact of station dependent errors (i.e. MP and PCV) on the smartphone's GNSS measurements, and in the measurement noise. Multipath combinations give further insights into the multipath impact. All the tests have been performed in the vicinity of Garbsen, Lower Saxony, Germany.

#### 3.3.1 Setup

Four scenarios with different levels of multipath have been considered to carry out the investigation. Smartphone's measurements have been collected in two zero-baseline and two short-baseline configurations. The zero-baseline setups are named as Scenario 1 and 2. The zero-baseline configuration is established by splitting the signal of a geodetic grade antenna, connecting both a geodetic grade receiver and a radio frequency (RF)-enclosure. The RF enclosure used for the analysis is self-built, consisting of a metal box with a feedthrough for a Sub-Miniature version A (SMA) coax cable for a transmitting antenna. The latter re-transmits the signal to the smartphone (see Figure 3.3). The dimensions of the box are 21 cm × 23 cm × 9 cm,

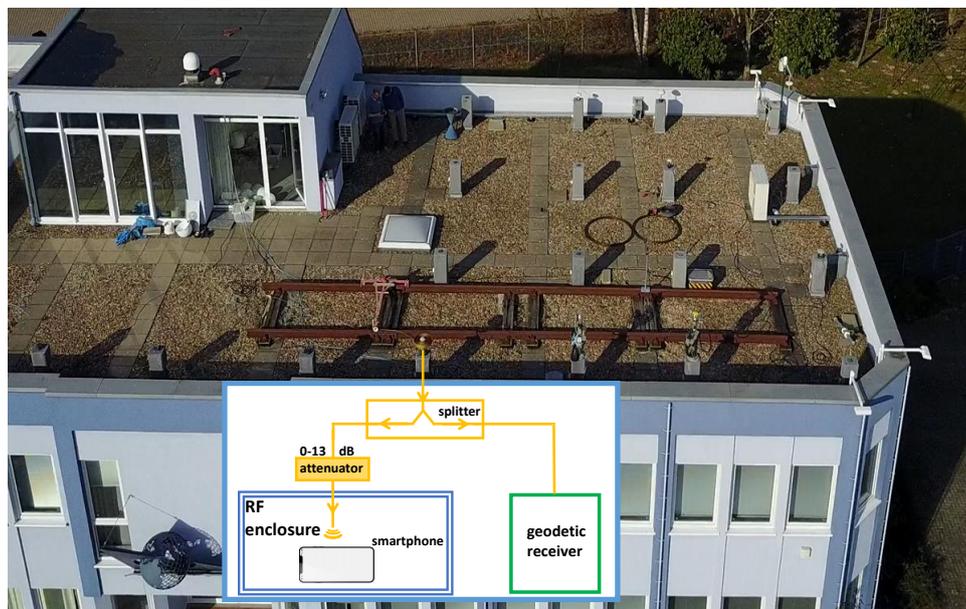
### 3.3. Quality analysis of smartphone measurements

while the space between smartphone and antenna element is roughly 2 cm. Within the enclosure, the smartphone is lying over a non-conductive support. For the sake of completeness, it has to be mentioned that there might be reflections generated within the enclosure, but they are expected to be all the same for a signal (e.g. L1). Proof of that assumption is presented in the double-difference analysis of this section.

In Scenario 2, an attenuation of 13 dB has been applied. The reason behind the attenuation is to highlight the impact of the signal's strength on the measurements' quality. In Scenario 1 and 2, as the device is within the RF enclosure, where the signal received by the choke-ring antenna is re-transmitted, the impact of use-case multipath on the smartphone antenna cannot be evaluated. Therefore, other configurations have been set up without any external antenna. As a consequence, a more challenging environment in terms of multipath is considered in Scenario 3. The smartphone is placed on one of the pillars on the Geo++ rooftop depicted in Fig. 3.3. Although the pillar removes most of the near-field ground reflections, some remaining horizontal and wall reflections can still affect the smartphone measurements.

Fig. 3.4 shows the Scenario 4 setup on a soccer field. Firstly, a JAVAD geodetic receiver has been installed over the center of the pitch. The position of the geodetic receiver has been calculated using observations of a reference station at the Geo++ building, roughly 12 km away. The resulting coordinates are then used as a reference to compute the DD. Secondly, the Mate20X has been laid down on the ground aligned to the north-south direction and with the bottom border over the reference point defined with the geodetic tripod (right panel of Fig. 3.4). Scenario 4 is an open sky scenario, where ground reflections have been removed by placing the smartphone directly on the field.

Table 3.1 summarizes the four scenarios considered. More than 150 hours of measurements have been gathered with the Mi8 and the Mate20X smartphones, which are equipped with the same dual-frequency GNSS receiver (i.e. the Broadcom BCM47755). In this section, the results obtained with the Mate20X are reported as an example. However, similar observations could be made for the Mi8. Furthermore, here, the C/N0 impact on the measurements is assessed.



**Figure 3.3:** Geo++ GmbH rooftop and zero-baseline configuration between smartphone and geodetic receiver. The signal of a choke-ring antenna on the roof of the building is split to feed a geodetic receiver and to broadcast the signal to the smartphone inside an RF enclosure.

**Table 3.1:** C/N0 scenarios used for the DD test.

Scenario	Setup	L1/E1 average C/N0	L5/E5a average C/N0
1	zero-baseline + RF enclosure	42 ±2 [dBHz]	43 ±3 [dBHz]
2	zero-baseline + RF enclosure + 13dB attenuation	32 ±3 [dBHz]	31 ±3 [dBHz]
3	short-baseline on rooftop	30 ±3 [dBHz]	29 ±3 [dBHz]
4	short-baseline in soccer field	39 [dBHz]	37 [dBHz]

The C/N0 reported in Table 3.1 are average values shown with an uncertainty that takes into account the several data-sets considered. Concerning Scenario 4, i.e. the soccer field, no uncertainty is associated with the C/N0 since the analysis is based on a single data-set.



**Figure 3.4:** Soccer field setup. Left and middle panel: a geodetic receiver has been used to compute the correct reference position of the smartphone. Right panel: the Mate20X has been aligned with the north-south direction, with the bottom border over the reference point defined with the geodetic tripod (the hole in the piece of cardboard).

### 3.3.2 Phase double-difference analysis

As shown by other authors (e.g. Fortunato et al., 2019), while the DD code provides valuable information in understanding the noise of the measurement, the DD carrier phase indicates when a solution with successful AR might be feasible with such measurements. Information on the impact of station dependent errors can be obtained. Furthermore, the DD is a useful tool to understand the presence of possible biases in the phase measurements. In fact, in some devices, e.g. the Nexus 9, some random and satellite dependent biases have been noticed. Several authors have reported this fact (e.g Riley et al., 2017; Geng and Li, 2019), and it is related to the generation of phase measurements in the smartphone receiver chipset. In general, both integer and fractional parts of the local replica signals from the Numerically Controlled Oscillator (NCO) must be stored at the same time. In particular, the integer component of the NCO phase should be counted during the continuous carrier-phase tracking (Hauschild, 2017). Moreover, the PLL aligns the difference between the incoming Intermediate Frequency

(IF) phase and the total NCO phase to zero to obtain a phase lock status (Misra and Enge, 2006). For some devices, the IF at the initial instant when the tracking begins is not the same for all the satellites. This feature results in an Initial Phase Bias (IPB) and affects the DD. More details and an IPB estimation procedure can be found in Geng and Li (2019). However, after a first initial version of the firmware, IPBs seem not to affect the BCM47755 chipset anymore, as reported in Shade and Madhani (2018); Geng and Li (2019), and shown in the following part of the analysis.

Here, the DD is computed choosing the highest elevation satellite as the reference satellite, which is named satellite  $q$ . Indicating the smartphone with  $u$  and the geodetic receiver with  $r$ , for a frequency  $f_j$  with wavelength  $\lambda$ , following Eq. 2.21, the phase DD equation can be written as:

$$\phi_{ur}^{sq} = \rho_{ur}^{sq} + c\delta_{ur}^{sq} + \zeta_{ur}^{sq} + \mu_{ur}^{sq} + T_{ur}^{sq} - I_{ur}^{sq} + \lambda(\omega_{ur}^{sq} + N_{ur}^{sq}) + \epsilon_{ur}^{sq}. \quad (3.17)$$

Where all the components have the same meaning as in Eq. 2.21. Moreover, since the distance between the two receivers considered in this analysis is no more than some tens of meters, the atmospheric ( $T_{ur}^{sq} - I_{ur}^{sq}$ ), the relativistic  $c\delta_{ur}^{sq}$  and phase wind-up effect ( $\omega_{ur}^{sq}$ ) impact can be neglected. Therefore, knowing the location coordinates, the remaining effects in the DD are the ambiguity term and the station dependent errors, i.e. PCV ( $\zeta_{ur}^{sq}$ ) and MP ( $\mu_{ur}^{sq}$ ):

$$\phi_{ur}^{sq} = \rho_{ur}^{sq} + \zeta_{ur}^{sq} + \mu_{ur}^{sq} + \lambda N_{ur}^{sq} + \epsilon_{ur}^{sq}. \quad (3.18)$$

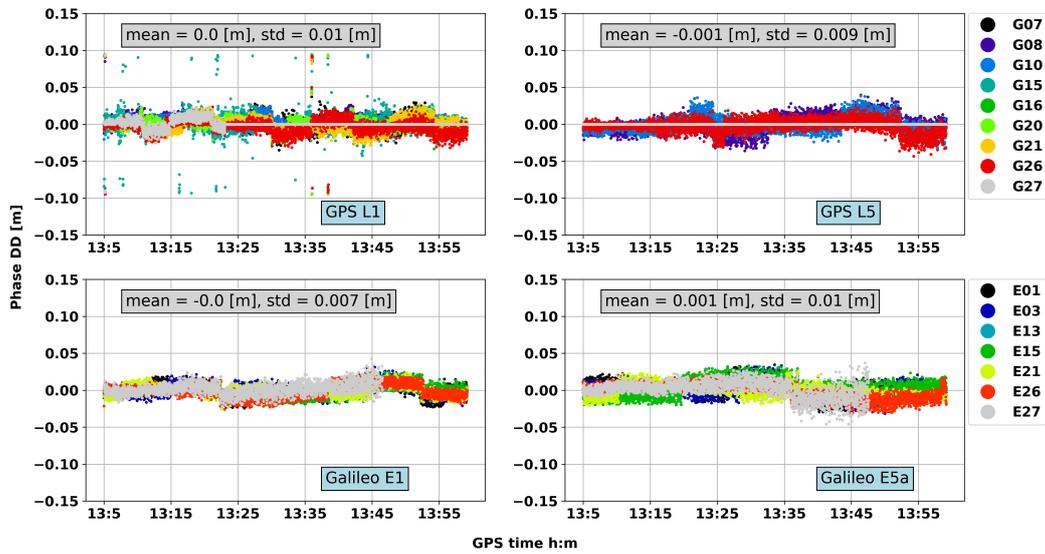
Fig. 3.5-3.18 depict examples of phase DD, C/N0 and satellite elevation that can be observed in the scenarios described in Table 3.1. The DD figures report the mean and STD as all the measurements were the same for the evaluation of the DD's noise. Besides, the different colors of the phase DD dots indicate different satellites to visualize possible phase biases. The measurements were collected during hour 13 (GPS time) at DOY 176 (year 2019) for Scenario 1, hour 9 at DOY 126 (year 2020) for Scenario 2, during hour 16 (GPS time) at DOY 338 (year 2019) for Scenario 3, and during hour 10 (GPS time) at DOY 324 for Scenario 4 (year 2019).

### Scenario 1

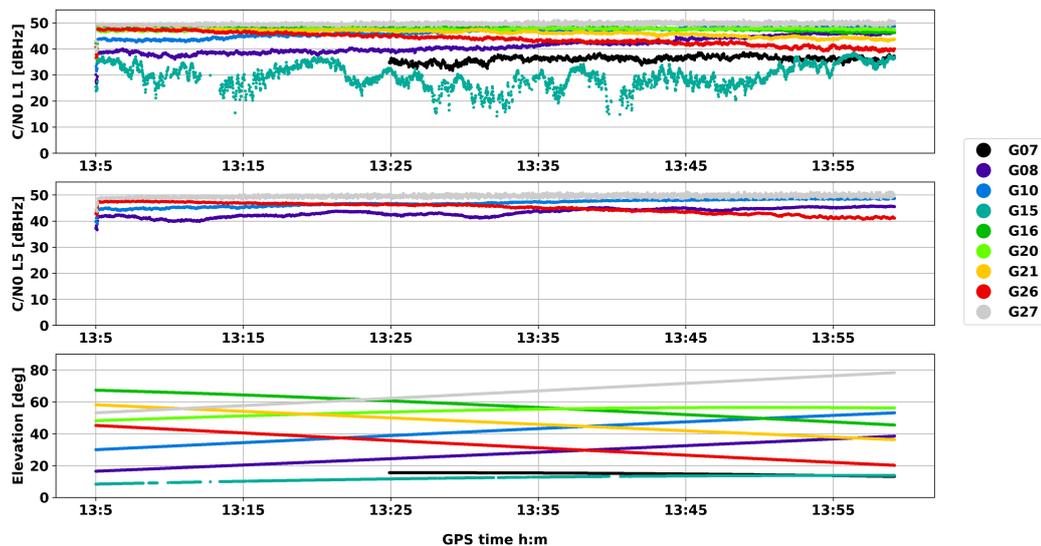
Fig. 3.5 shows the phase DD obtained in Scenario 1. The results are similar to those obtained by other authors (e.g. Riley et al., 2017; Shade and Madhani, 2018). Since ambiguity resolution is the final objective, having an unbiased phase DD is an essential requirement. Fig. 3.5 shows that there is no bias in the phase DD and indicates an STD lower than roughly 1 cm for both L1/E1 and L5/E5a. It can be observed that one satellite, i.e. G15, exhibits some cycle and half-cycle jumps in the phase DD. This is related to the low elevation (lower than  $12^\circ$ , see Fig. 3.6) and low C/N0 (lower than 30 dBHz, see Fig. 3.6) of the satellite during the period considered. For the specific example see Fig. C.1 in Appendix C where a triple-difference (TD) is shown.

Furthermore, the phase DD of each satellite manifests some irregular jumps, as shown by other authors (e.g. Shade and Madhani, 2018; Li and Geng, 2019). To the best of the author's knowledge, the reason for this anomaly is unknown, but it might be caused by the discontinuous operation of the PLL (Li and Geng, 2019). Shade and Madhani (2018) reported these jumps and mentioned that they might be related to the chipset computation. However, the jumps in phase are consistent with the variation in the pseudorange (Li and Geng, 2019). Considering the same data-set analyzed in Fig. 3.5, Fig. 3.8 shows an example for one satellite (i.e. G26) considering the L1 frequency to better investigate the jumps. In Fig. 3.8, the red arrows highlight the jumps in the phase DD. As suggested by Fig. 3.8, the jumps can be adjusted by considering a delay of a multiple of 0.04 ns. In the range, this value is roughly 1.2 cm.

However, as mentioned above, the nature of the jumps is still unknown. Nevertheless, ambiguity resolution should be feasible in Scenario 1. For a direct comparison of the variation of phase DD, TD, C/N0 and elevation of the specific satellite G26, see Fig. C.1 in Appendix C.

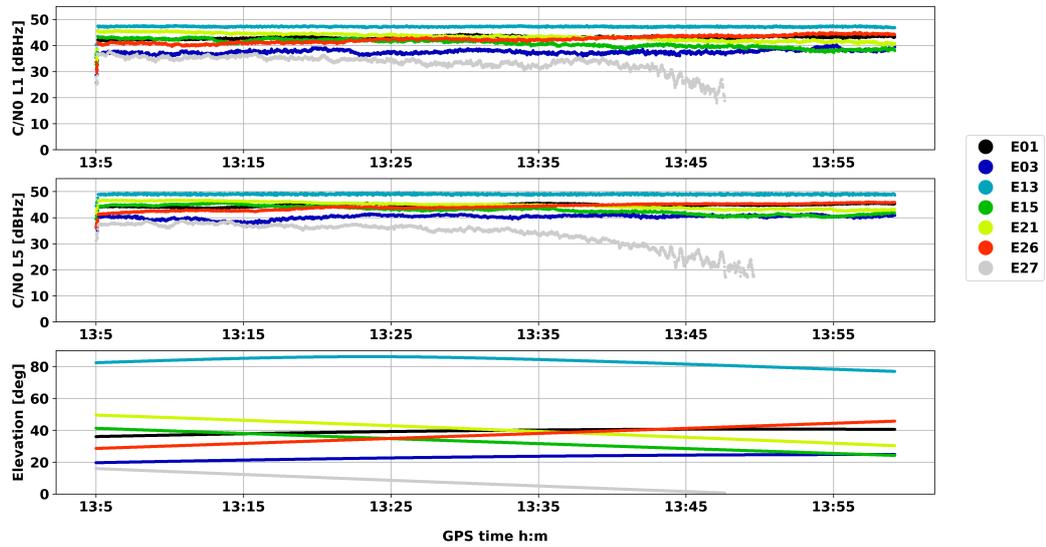


**Figure 3.5:** Example of phase DD of GPS and Galileo measurements in Scenario 1 (hour 13, DOY 176, year 2019). From top to bottom: GPS and Galileo phase DD. From left to right: L1/E1 and L5/E5a phase DD. The DD is computed between the Mate20X and a geodetic receiver in zero-baseline (Scenario 1). Here, the nearest integer number of cycles has been subtracted for each satellite.

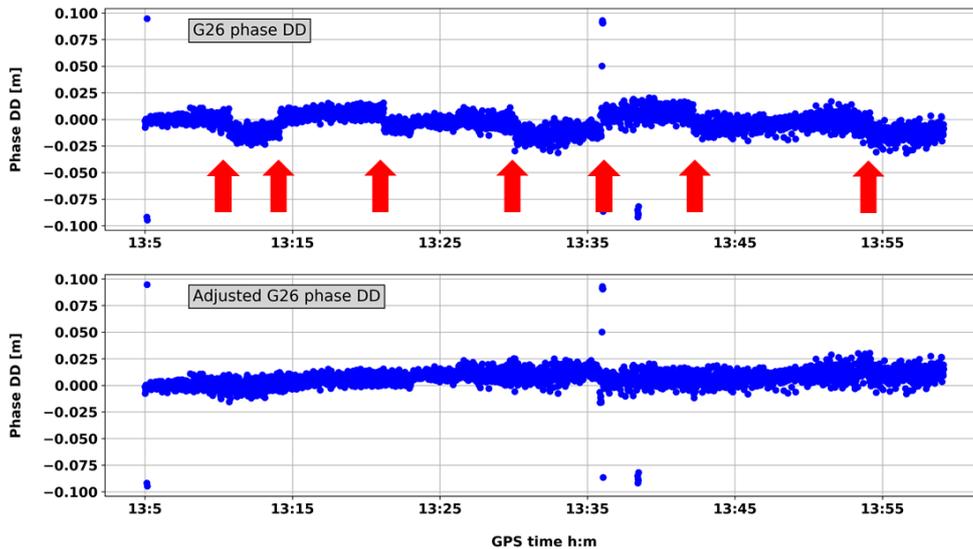


**Figure 3.6:** First row: C/N0 of the GPS L1 measurements gathered with the Mate20X. Second row: C/N0 of the GPS L5 measurements gathered with the Mate20X. Third row: elevation of the GPS satellites. The measurements have been collected in Scenario 1 (hour 13, DOY 176, year 2019).

### 3.3. Quality analysis of smartphone measurements



**Figure 3.7:** First row: C/N0 of the Galileo E1 measurements gathered with the Mate20X. Second row: C/N0 of the Galileo E5a measurements gathered with the Mate20X. Third row: elevation of the Galileo satellites. The measurements have been collected in Scenario 1 (hour 13, DOY 176, year 2019).



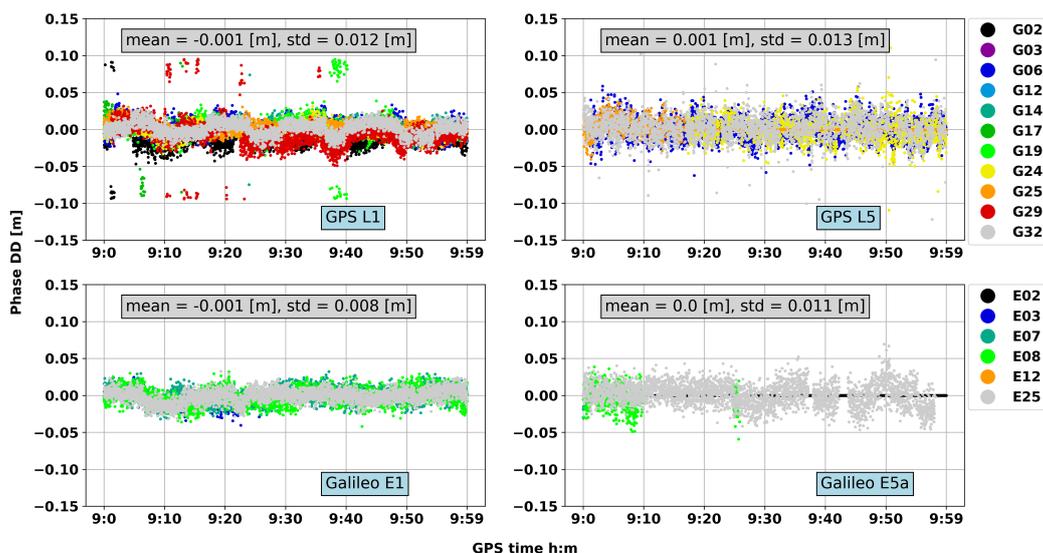
**Figure 3.8:** Example of phase DD jump in smartphone's measurements collected with devices equipped with the BCM47755 chipset. From top to bottom: L1 phase DD and adjusted phase DD of G26 observations collected in Scenario 1 with the Mate20X. The adjustment considers jumps that are integer multiples of 0.04 ns multiplied by the speed of light, i.e. roughly 12 cm. The occurrence of the jumps is indicated by a red arrow.

### Scenario 2

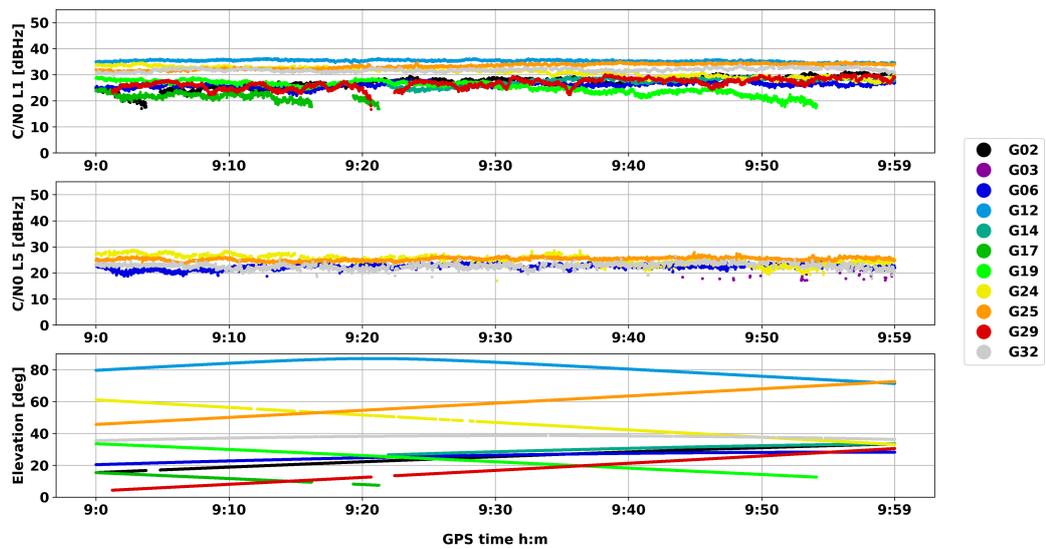
Fig. 3.9-3.11 indicates that the application of attenuation of 13 dB in Scenario 2 reduces the strength of the signal (as expected), but does not affect the quality of the measurements significantly. In fact, Fig. 3.9 suggests that the measurements are slightly noisier than Scenario 1, but no biases are introduced. Hence, with phase observations collected in Scenario 2, ambiguity resolution should be feasible as well as in Scenario 1.

Concerning the Galileo measurements, it is worth observing that, in this case, some L5 measurements are discarded even if for the same satellite L1 measurements are available. In fact, the L5 phase measurements not reported are flagged as *invalid* (see Section 3.2 and Appendix C). There is no more detailed information provided by the API concerning the receiver's internal behavior when setting such a flag.

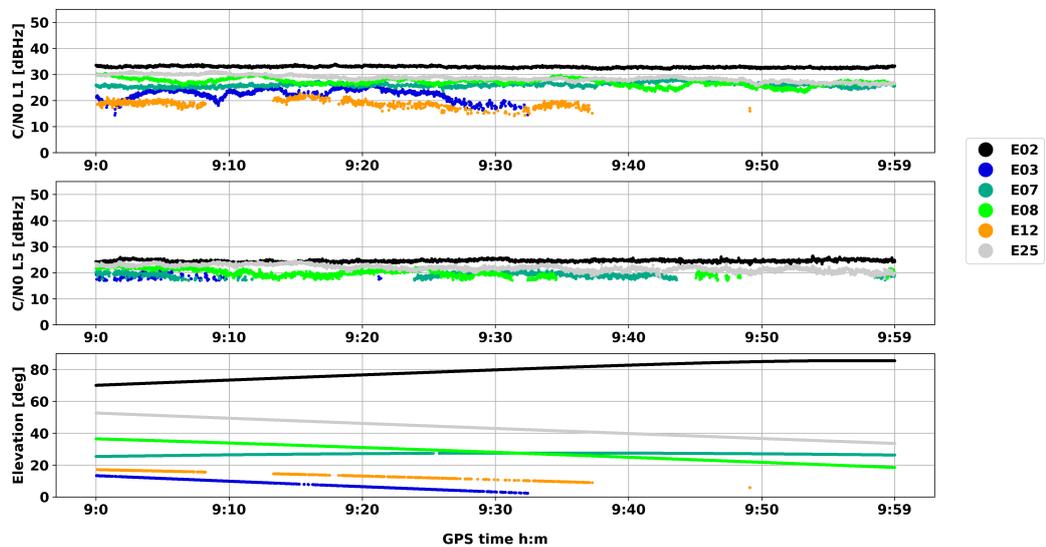
However, some reasonable considerations can be made. The PLL is highly sensitive to noise. Several authors already showed the correlation between low C/N0 and noise in the smartphone measurements. In this case, most probably, the signal's attenuation made the signal too weak and caused the failure of the PLL, resulting in a non-valid ADR. In fact, Fig. 3.11 shows that the L5 C/N0 of satellites like E03 and E08 is below 20 dBHz. Another reason behind the different behavior of L1 and L5 is the use of two different antennas. The employment of two antennas is discussed in Chapter 4 and causes a non-homogeneous gain pattern leading to different signal strength reception.



**Figure 3.9:** Phase DD of GPS and Galileo measurements in Scenario 2 (hour 9, DOY 126, year 2020). From top to bottom: GPS and Galileo phase DD. From left to right: L1/E1 and L5/E5a phase DD. The nearest integer number of cycles has been subtracted for each satellite.



**Figure 3.10:** GPS measurements collected with the Mate20X in Scenario 2 (hour 9, DOY 126, year 2020). First row: C/N0 of the L1 measurements. Second row: C/N0 of the L5 measurements. Third row: satellite elevation.

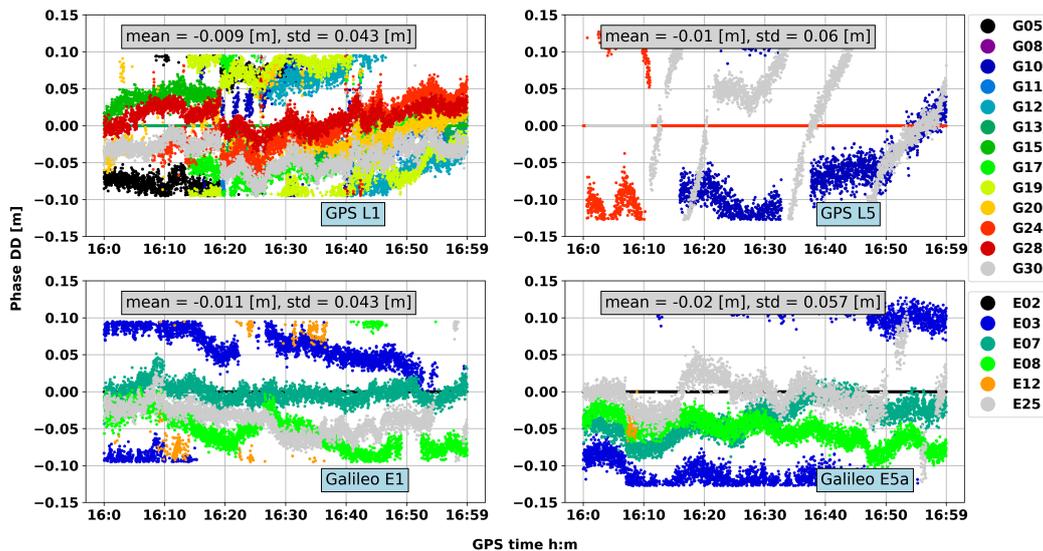


**Figure 3.11:** Galileo measurements collected with the Mate20X in Scenario 2 (hour 9, DOY 126, year 2020). First row: C/N0 of the E1 measurements. Second row: C/N0 of the E5a measurements. Third row: satellite elevation.

### Scenario 3

A different result is instead observed in Scenario 3. In that case, Fig. 3.12 suggests that there is an introduction of phase biases for both constellations and frequencies on a concrete pillar. A clear wavy perturbation is observed in Fig. 3.12. This behavior is typical in observations highly affected by multipath.

Comparing the C/N0 figures obtained in Scenario 2 and 3 (Fig. 3.10, 3.11 vs Fig. 3.13, 3.14), it can be noticed that while the signal strength is similar, the impact of station dependent errors (e.g. multipath) is completely different, resulting in biased phase DD in Scenario 3. Fig. 3.15 shows the mean of the phase DD absolute values for each satellite. The analysis of the same data-set described in Fig. 3.12 is intended to highlight the presence of biases in the analyzed phase DD. Mean values related to non-full cycles difference are reported resulting in biased phase DD, which is a limitation for successful AR.



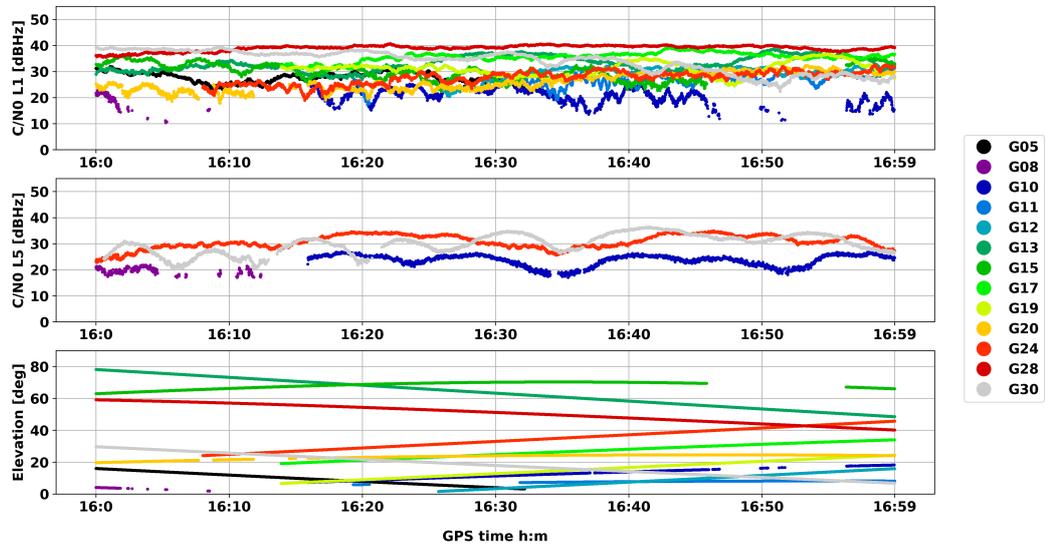
**Figure 3.12:** Example of phase DD of GPS and Galileo measurements in Scenario 3 (hour 16 of DOY 338 of the year 2019). From top to bottom: GPS and Galileo phase DD. From left to right: L1/E1 and L5/E5a phase DD. The DD is computed between the Mate20X and a geodetic receiver roughly 10 m distant in Scenario 3. Here, the nearest integer number of cycles has been subtracted for each satellite. While the geodetic receiver’s position used is the antenna location, the smartphone’s position considered is the center of the smartphone.

#### Scenario 4

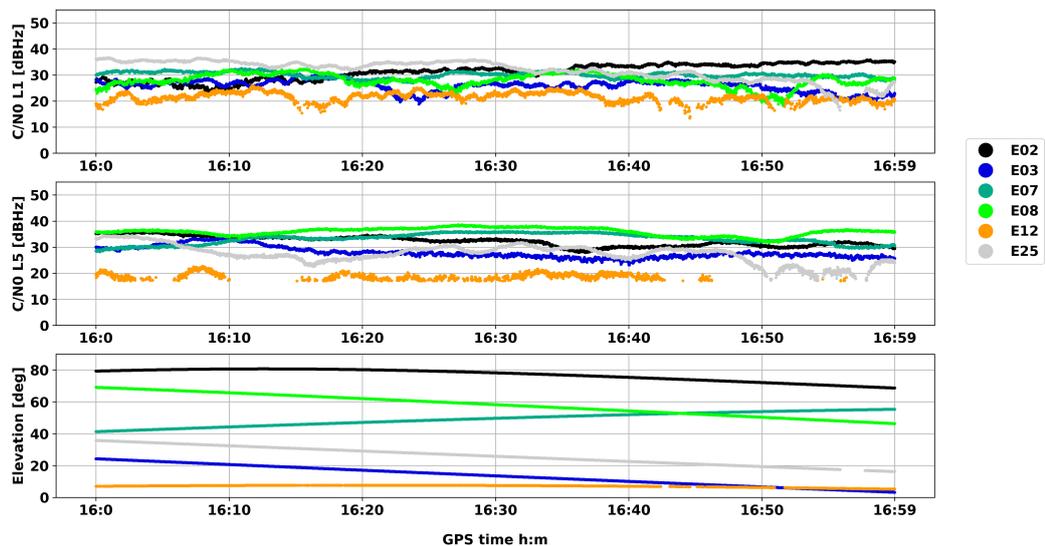
Fig. 3.16 shows the phase DD in Scenario 4, where the smartphone lays on the ground of a soccer field. In this case, while the geodetic receiver’s position considered is the antenna location, the smartphone’s position is referenced as the center of the smartphone. Although no strong wavy pattern associated with multipath is recognized in the observations, some biases can be observed. This effect might be due to the PCV of the smartphone antenna. The latter is further investigated in Chapter 4.

The comparison leads to the following conclusions. The use of the geodetic-grade antenna, together with the RF enclosure, mostly removes the impact of station dependent errors on phase DD. Furthermore, the attenuation of the signal (using a 13 dB attenuator) does not introduce any biases. Also, the wavy impact of multipath seems to be reduced by placing the smartphone on the ground of a soccer field (Scenario 4). However, significant biases are still present in the DD also in this case. Following Eq. 3.18, these biases are most likely related to the PCV of the antenna and they are further assessed in Chapter 4.

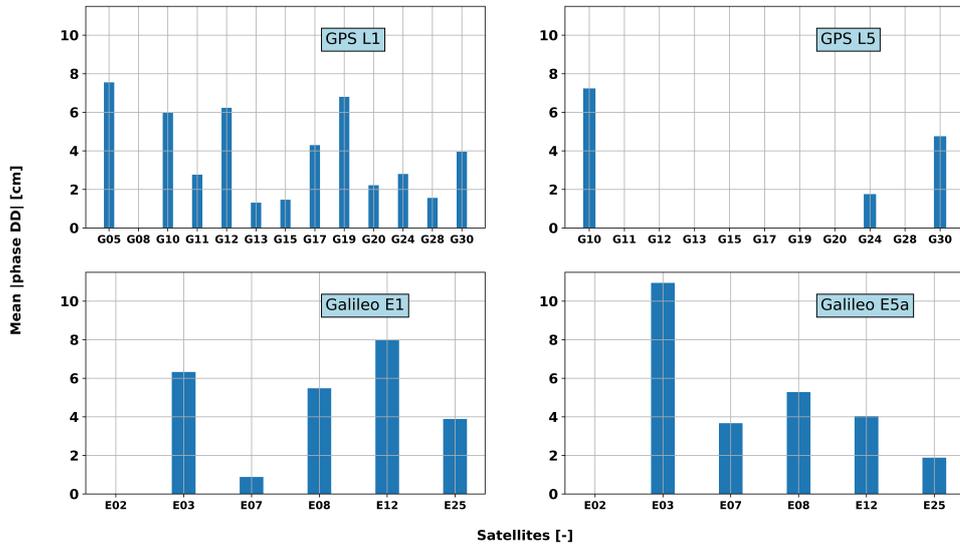
### 3.3. Quality analysis of smartphone measurements



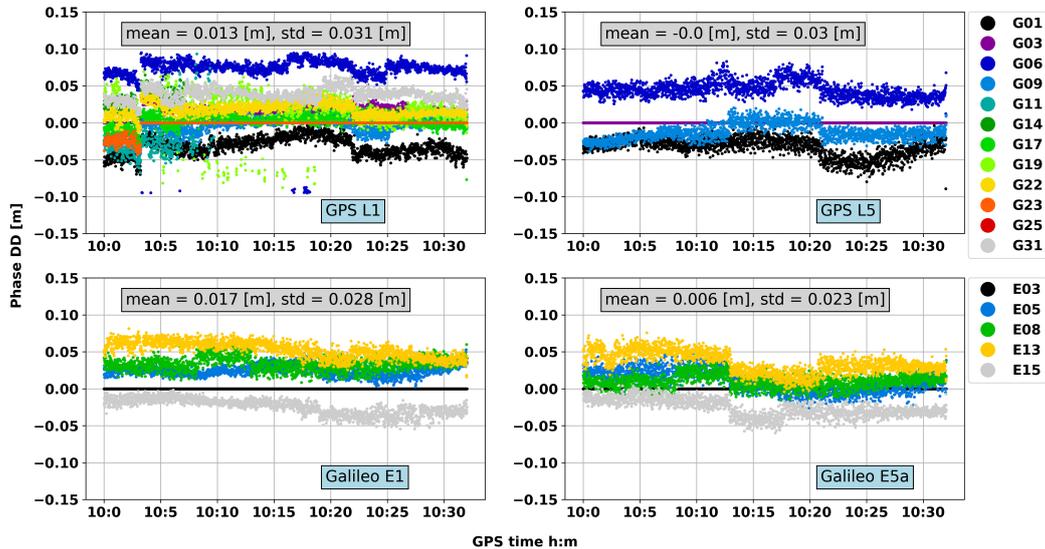
**Figure 3.13:** Example of C/N0 of GPS signals in Scenario 3. The measurements have been collected during hour 16 of DOY 338 of the year 2019. First row: C/N0 of the GPS L1 measurements gathered with the Mate20X in Scenario 3. Second row: C/N0 of the GPS L5 measurements gathered with the Mate20X. Third row: elevation of the GPS satellites.



**Figure 3.14:** Example of C/N0 of Galileo signals in Scenario 3. The measurements have been collected during hour 16 of DOY 338 of the year 2019. First row: C/N0 of the Galileo E1 measurements gathered with the Mate20X in Scenario 3. Second row: C/N0 of the Galileo E5a measurements gathered with the Mate20X. Third row: elevation of the Galileo satellites.

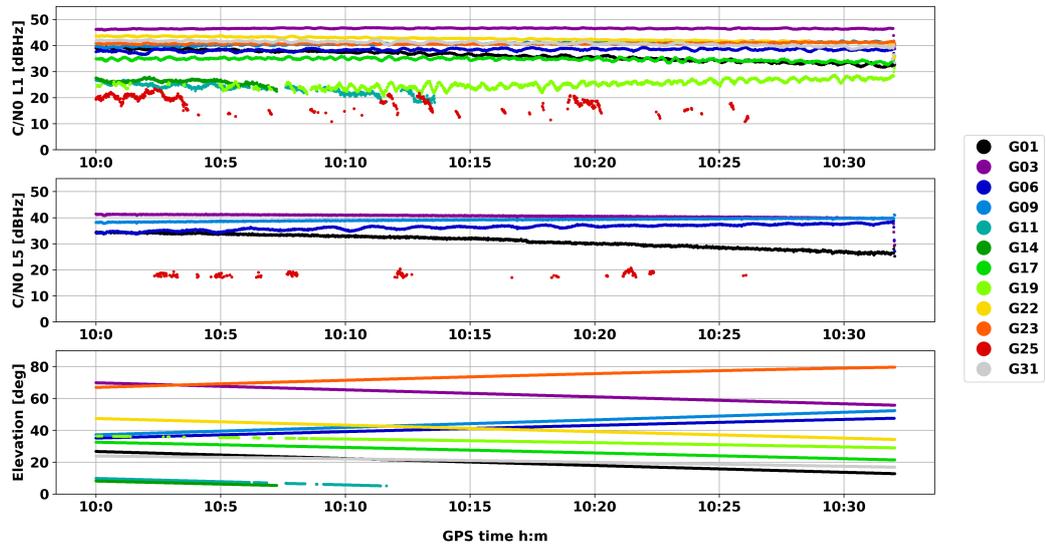


**Figure 3.15:** Example of mean phase DD absolute values of GPS and Galileo measurements in Scenario 3 (hour 16 of DOY 338 of the year 2019). From top to bottom: GPS and Galileo. From left to right: L1/E1 and L5/E5a. The DD is computed between the Mate20X and a geodetic receiver roughly 10 m distant in Scenario 3.

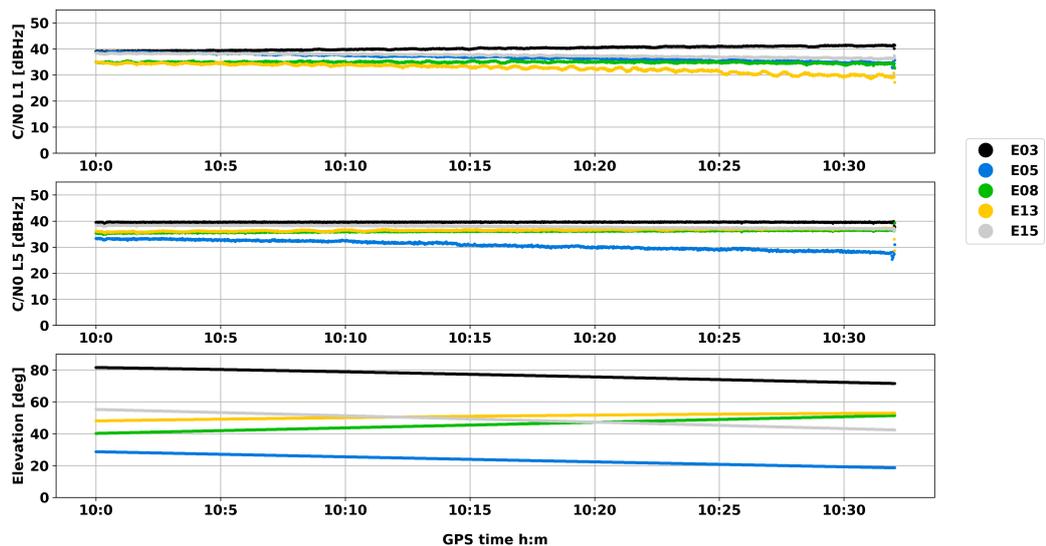


**Figure 3.16:** Phase DD of GPS and Galileo measurements in Scenario 4. From top to bottom: GPS and Galileo phase DD. From left to right: L1/E1 and L5/E5a phase DD. The DD is computed between the Mate20X and the geodetic receiver roughly 50 m distant in the soccer field. Here, the nearest integer number of cycles has been subtracted for each satellite. The measurements have been collected during hour 10 of DOY 234 of the year 2019.

### 3.3. Quality analysis of smartphone measurements



**Figure 3.17:** First row: C/N0 of the GPS L1 measurements gathered with the Mate20X in Scenario 4. Second row: C/N0 of the GPS L5 measurements gathered with the Mate20X. Third row: elevation of the GPS satellites commonly tracked by the Mate20X and JAVAD receiver used in the DD computation (e.g. satellite G25 is tracked only by the Mate20X in this case). The measurements have been collected during hour 10 of DOY 234 of the year 2019.



**Figure 3.18:** Galileo measurements collected with the Mate20X in Scenario 4 (hour 10, DOY 234, year 2019). First row: C/N0 of the E1 measurements. Second row: C/N0 of the E5a measurements. Third row: elevation of the Galileo satellites commonly tracked by the Mate20X and JAVAD receiver used in the DD computation (see Fig. 3.16).

### 3.3.3 Code noise and multipath investigation

#### Code noise analysis

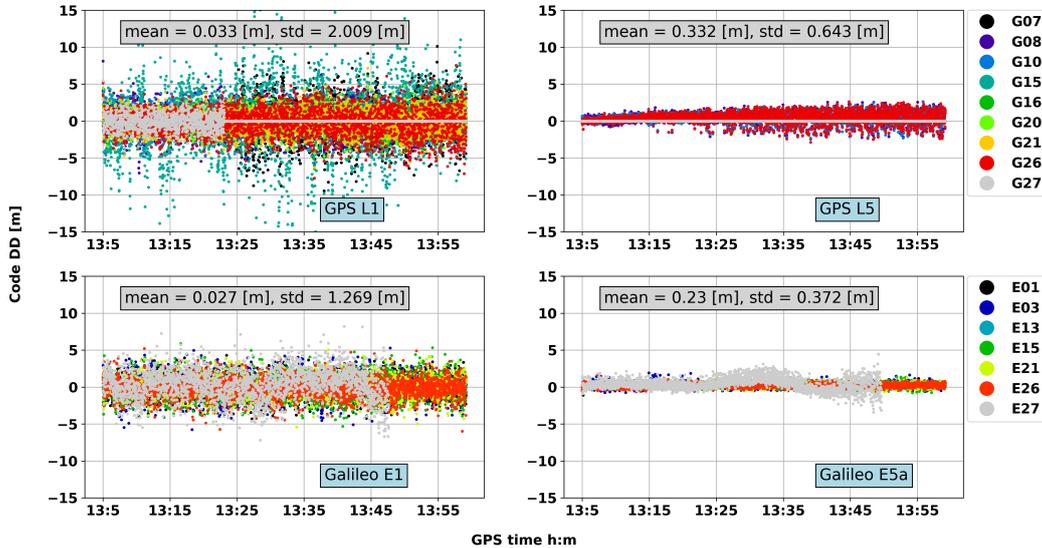
The code DD observations provide some insight into the noise of the measurement and the multipath impact. Following the same concepts introduced in subsection 3.3.2 for the phase DD equation, the non-ambiguous code equation can be retrieved as:

$$p_{ur}^{sq} = \rho_{ur}^{sq} + \zeta_{ur}^{sq} + \mu_{ur}^{sq} + \epsilon_{ur}^{sq}. \quad (3.19)$$

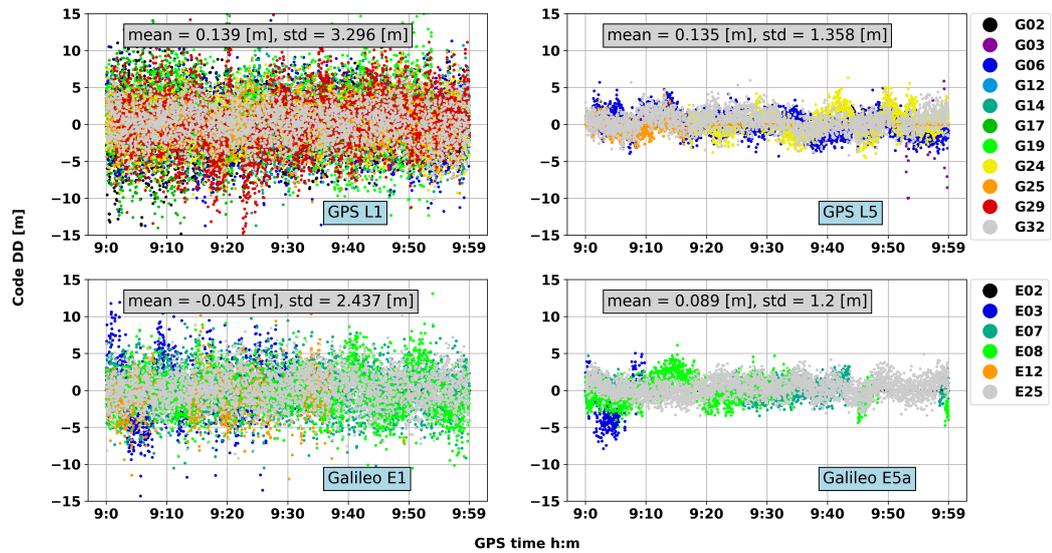
In Scenario 1, the noise of the code is lower than 2 m and 1 m, for GPS L1 and L5, respectively (see Fig. 3.19). The Galileo's are less noisy than the GPS' measurements with an STD of 1.269 m and 0.372 m for E1 and E5a, respectively.

While comparable values are observed in Scenario 1, 2 and 4 (as shown in Fig. 3.20 and Fig. 3.22), Scenario 3 describes a completely different noise and multipath environment.

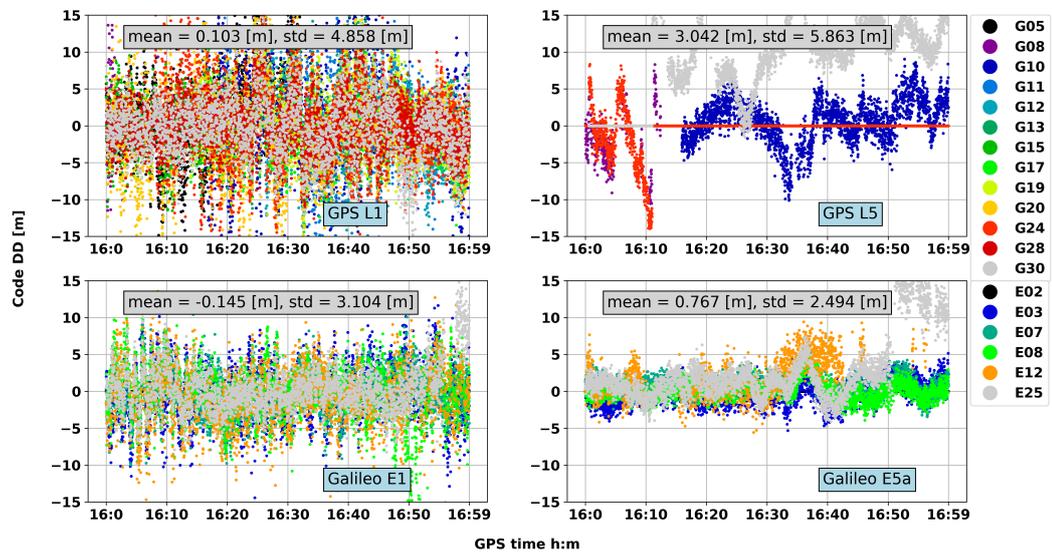
A large STD for GPS measurements is observed, i.e about 5 m and 6 m for L1 and L5, respectively. The noise also increases for the Galileo measurements with an STD larger than 2 m for both E1 and E5a. Furthermore, some wavy patterns that can be associated with multipath are observed (e.g. G30 for GPS L5 and E25 for Galileo E5a). Also, some biases can be recognized. As an example, the code DD observation of GPS G30 for L5 is biased by about 10 m. Some wavy variations can be recognized also in Scenario 4 (see Fig. 3.22). In particular, it can be observed in the L5/E5a measurements that are less noisy than L1/E1. As examples, G06 for GPS L5 and E13 for Galileo E13 exhibit some short-period variations, most likely due to multipath.



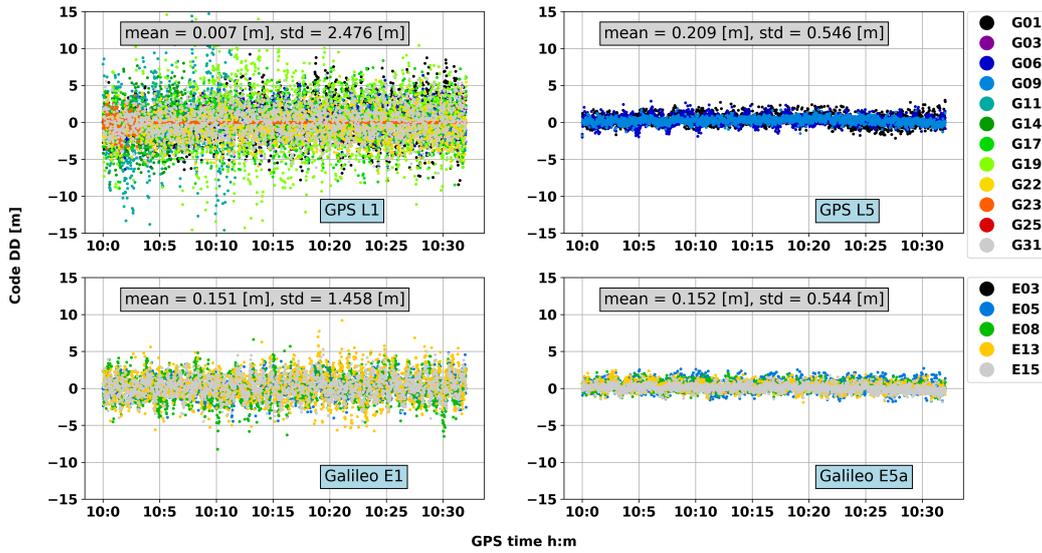
**Figure 3.19:** Code DD of GPS and Galileo measurements in Scenario 1. From top to bottom: GPS and Galileo code DD. From left to right: L1/E1 and L5/E5a code DD.



**Figure 3.20:** Example of code DD of GPS and Galileo measurements in Scenario 2. From top to bottom: GPS and Galileo code DD. From left to right: L1/E1 and L5/E5a code DD. The DD is computed between the Mate20X and the geodetic receiver in zero-baseline configuration (Scenario 2).



**Figure 3.21:** Code DD of GPS and Galileo measurements in Scenario 3. From top to bottom: GPS and Galileo code DD. From left to right: L1/E1 and L5/E5a code DD. The DD is computed between the Mate20X and the geodetic receiver roughly 10 m distant in Scenario 3.



**Figure 3.22:** Code DD of GPS and Galileo measurements in Scenario 4. From top to bottom: GPS and Galileo code DD. From left to right: L1/E1 and L5/E5a code DD. The DD is computed between the Mate20X and the geodetic receiver roughly 50 m distant in the soccer field.

### Pseudorange multipath combination analysis

It has been demonstrated that Scenario 3 and Scenario 4 are two open sky configurations with different levels of multipath affecting the smartphone measurements. Another indicator to further assess the multipath effect is the pseudorange multipath combination. For two frequencies  $f_i$  and  $f_j$ , and a satellite  $s$ , the pseudorange multipath combination is computed in the following way (e.g. Langley, 1998a):

$$O_{MP,i} = p_i^s - \phi_i^s - 2k(\phi_i^s - \phi_j^s), \quad \text{with} \quad k = \frac{f_j^2}{f_i^2 - f_j^2}. \quad (3.20)$$

In Eq. 3.20  $p_i^s$  and  $\phi_i^s$  are the pseudorange and carrier phase observations, respectively. Here, data gaps are removed, and observations corrected for cycle-slips. After filtering the data, the ambiguity term that affects Eq. 3.20 is assumed constant over the time of observation. Under this assumption, the ambiguity contribution is removed by subtracting the average over the considered period. The remaining multipath impact can be expressed in the following way:

$$\Delta O_{MP,i} = O_{MP,i} - \overline{O_{MP,i}}. \quad (3.21)$$

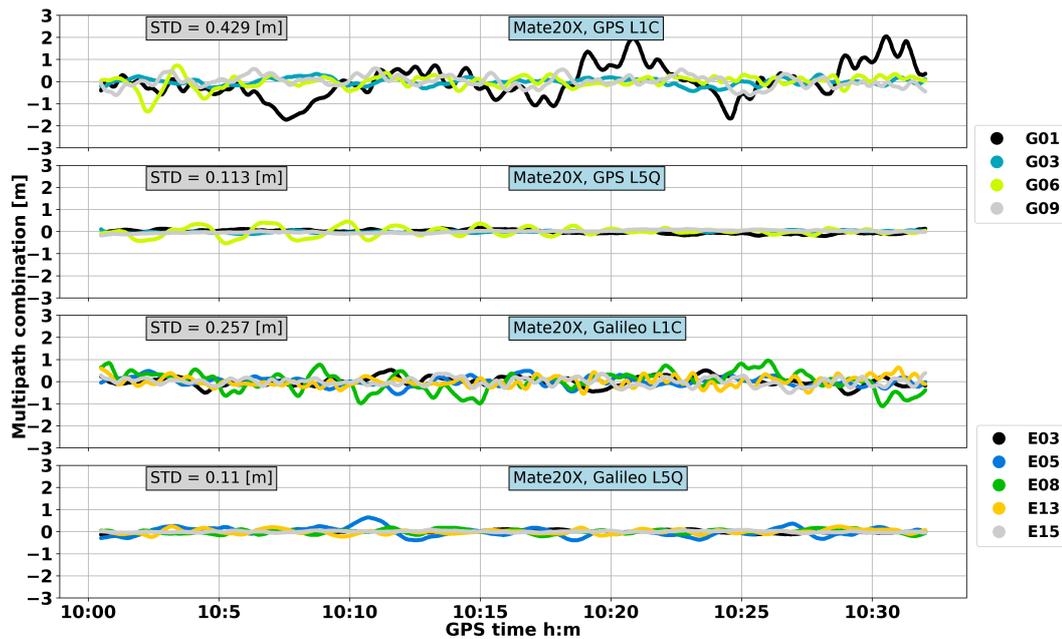
Where  $\overline{O_{MP,i}}$  is the mean value of the multipath combination throughout the observation time. Furthermore, to smooth the data and have a clearer variation per satellite, an exponential moving average (e.g. Lawrance and Lewis, 1977) is applied to  $\Delta O_{MP,i}$ .

As the analysis of the phase and code DD showed that the impact of multipath is more significant in Scenario 3 than Scenario 4, Scenario 4 is investigated first. Fig. 3.23 shows the moving average (with a 30 s window) of  $\Delta O_{MP,i}$  in Scenario 4. The results suggest that the magnitude of multipath affecting the smartphone measurements on the ground of the pitch is lower than 2 m. In general, the multipath effect can reach up to 1.5 times the wavelength of the signal on code measurements (e.g. 450 m for code L1), and up to a quarter of wavelength for phase measurements (Irsigler, 2008), e.g. roughly 5 cm for L1. However, typical values

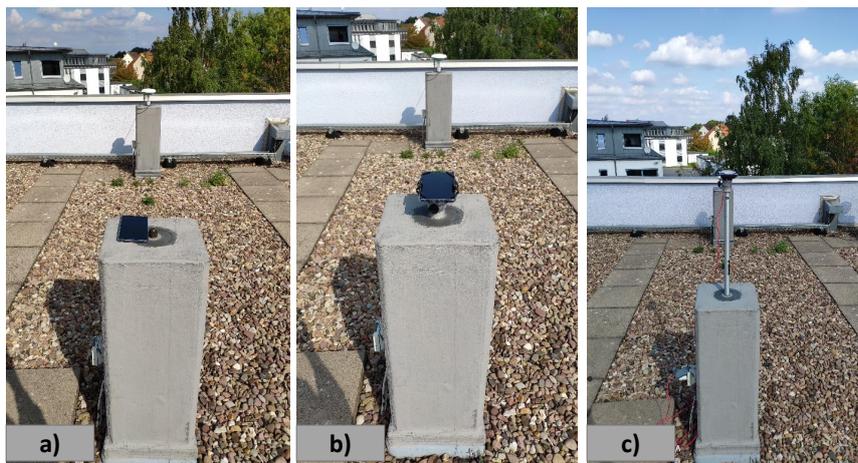
### 3.3. Quality analysis of smartphone measurements

of multipath are around a few meters. Therefore, Fig. 3.23 indicates that the soccer field is a scenario where the smartphone experiences similar multipath to standard rover receivers.

Secondly, the influence of the ground-reflections that can affect the measurements on the rooftop in Scenario 3 is assessed. To further evaluate the impact of the ground reflections, three different setups have been studied: the smartphone lying over the top of the pillar (Fig. 3.24a), the smartphone on a support 9 cm above the top of the pillar (Fig. 3.24b) and the smartphone mounted on a 60 cm pole attached to the top of the pillar (Fig. 3.24c). In all the configurations, the smartphone was placed aligning the device to the north-south direction carefully.

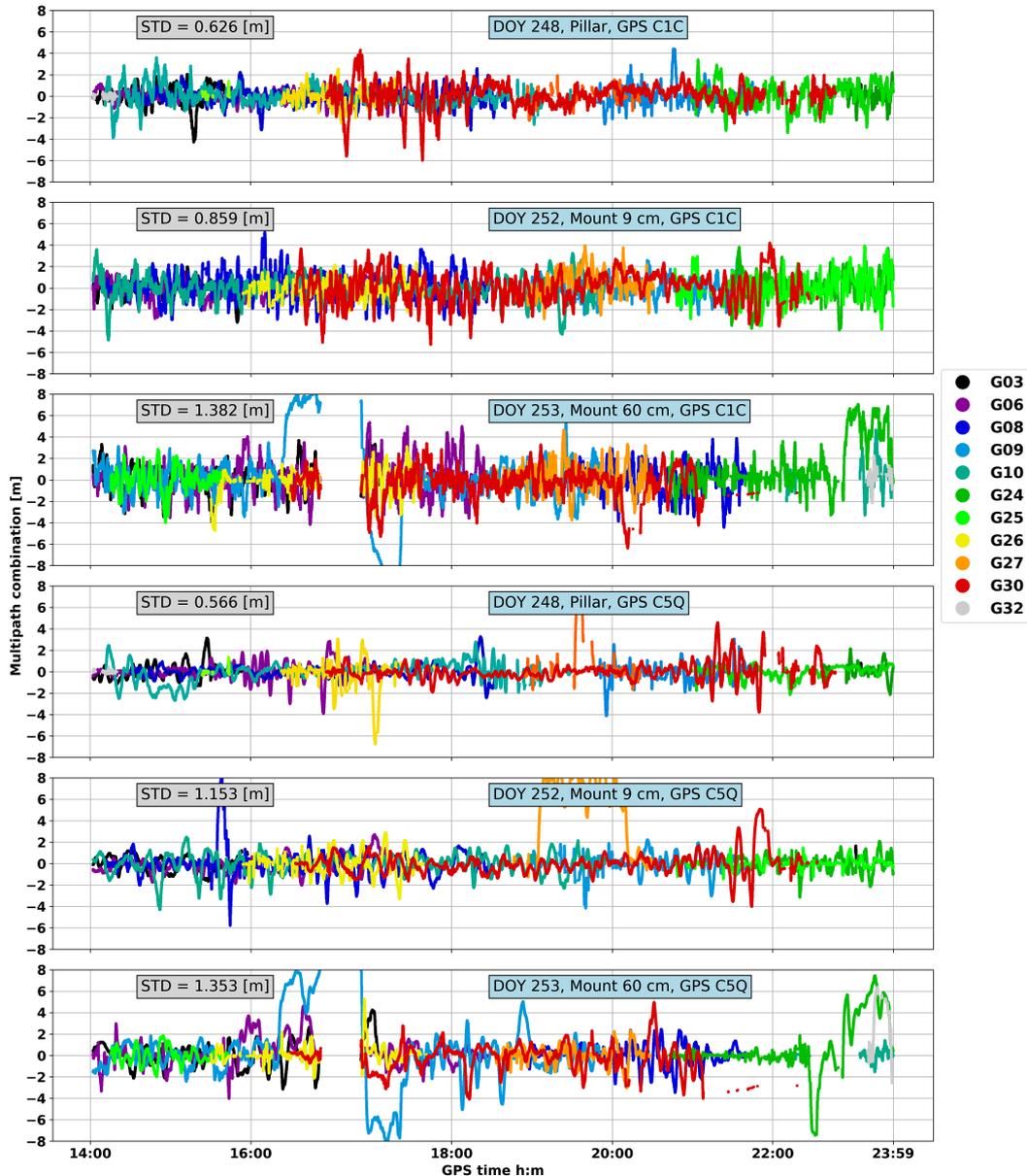


**Figure 3.23:** Exponential moving average of the pseudorange multipath combination for GPS and Galileo L1 and L5 signals (from top to bottom) collected with the Mate20X in Scenario 4. A 30 s moving window has been used.



**Figure 3.24:** Setups of the multipath-impact experiment: a) smartphone lying on the pillar, b) smartphone mounted on a 9 cm height support over the pillar, c) smartphone mounted on a 60 cm height support over the pillar. The mount is made by a metallic pole with a plastic support.

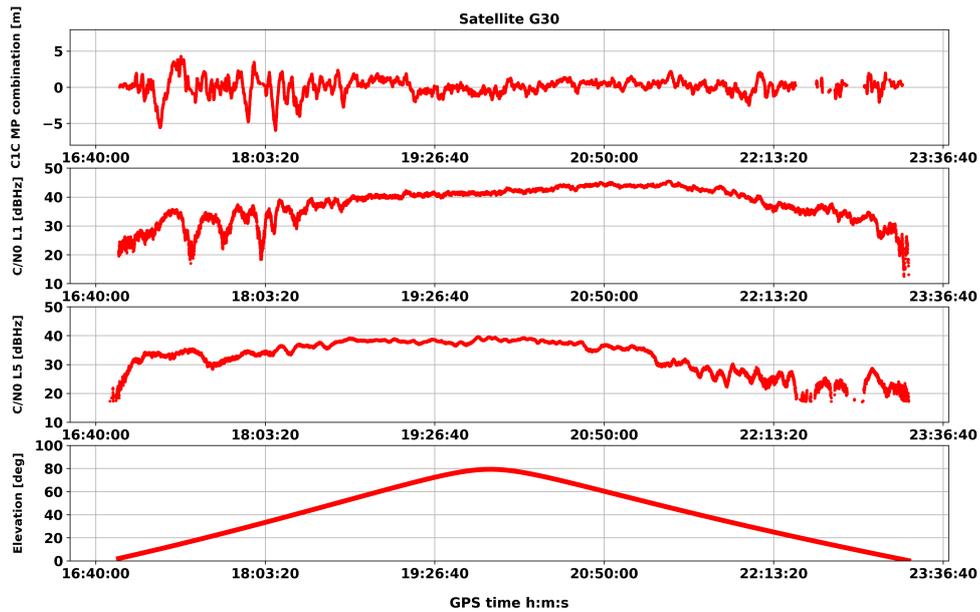
As in the soccer field scenario, the multipath combination is computed, and an exponential moving average applied for all the three setups. Fig. 3.25 shows an entirely different multipath environment w.r.t. the one depicted for the pitch in Fig. 3.23. While in the soccer field all the MP absolute values are lower than 1 m (but G01 for L1 frequency), here, the magnitudes go up to more than 8 m. Besides, the STD varies between 0.626 m and 1.382 m for L1 and between 0.566 m and 1.353 m for L5, while in the soccer field does not exceed 0.429 m and 0.257 m for L1 and L5, respectively. Furthermore, it can be observed that the higher the pole is, the bigger the multipath effect becomes. In fact, the larger distance from the pillar removes the shielding effect of the pillar itself. In this way, more reflections from the ground can reach the smartphone.



**Figure 3.25:** Exponential moving average of the pseudorange multipath error for GPS C1C and C5Q signals gathered with the Mate20X. A 120 s moving window has been used to average. The three setups shown in Fig. 3.24 have been evaluated.

As an example, Fig. 3.26 depicts the satellite G30 MP combination for L1 measurements in the configuration shown in Fig. 3.24a considered in Fig. 3.25, i.e. placed on the surface of the pillar. Fig. 3.26 highlights the correlation among MP, C/N0 and elevation. Overall, high (low) values of elevation combined to large (small) values of C/N0 correspond to low (high) noise in the multipath. In particular, large fluctuations of the C1C (RINEX convention) MP combination and C/N0 are observed for the same intervals (see Fig. 3.26).

Summarizing, Fig 3.25 depicts a challenging multipath environment for AR, in particular when the smartphone stands over one of the mounts.



**Figure 3.26:** From top to bottom: exponential moving average of C5Q MP combination, L1 C/N0, L5 C/N0 and elevation. The satellite considered is GPS G30. A 120 s moving window has been used to average. The setup is depicted in Fig. 3.24a.

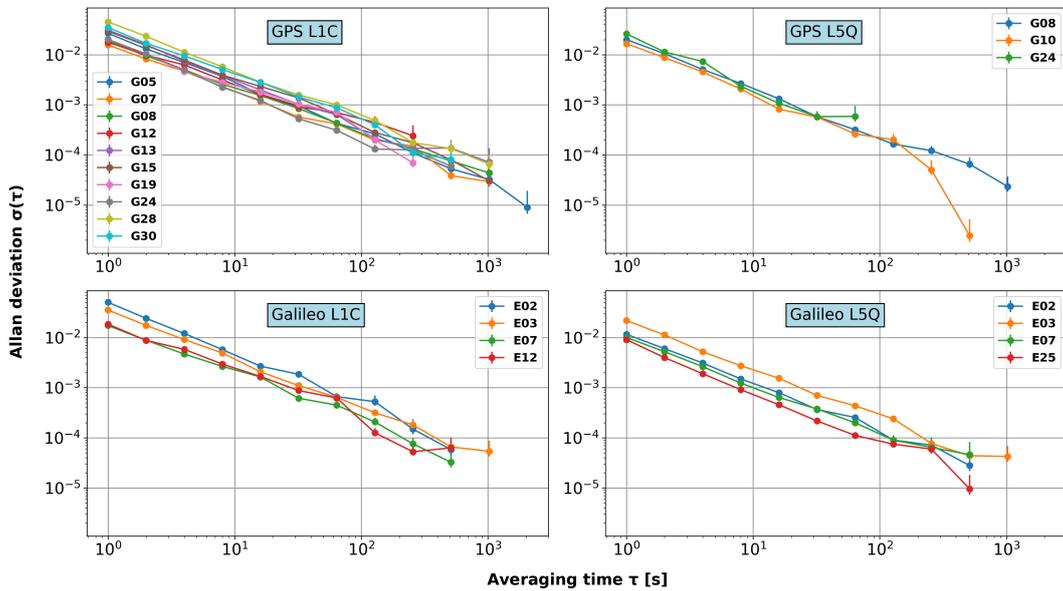
#### 3.3.4 Allan deviation analysis

An analysis using the Allan deviation is carried out to provide an insight into the process noise when using DD measurements. The Allan deviation is used to investigate the stability of a process. In fact, for some random processes, the STD is not a valuable measure of stability. This is mainly due to a non-stationary average for divergence noise types. In general, many noise types have convergence problems, e.g. the Flicker noise. From the analysis of the process stability, information about the process noise can be retrieved. In fact, the Allan deviation can be related to the exponent  $\alpha$  of the power law noise process (e.g. Riley, 2008):

$$S_y(f) = h(\alpha)f^\alpha, \quad (3.22)$$

where  $S_y(f)$  is the one-sided power spectral density of  $y$ , which is the fractional frequency fluctuations.  $f$  is the Fourier frequency,  $h(\alpha)$  the intensity coefficient and  $\alpha$  the exponent of the power law noise process. Different types of process noise can be associated with different values of  $\alpha$ . For example, for FM white noise,  $\alpha$  is 0, while for PM white noise, it is 2, and for a PM Flicker noise, it is 1. The Stable 32 software (Riley, 2014) has been used to process the data, setting a sampling time equal to the observation time ( $T/\tau = 1$ ) and introducing the data as phase data.

Fig. 3.27 shows the Allan deviation  $\sigma(\tau)$  of the phase DD in Scenario 3. Scenario 3 has been chosen for this analysis as the closest environment to a common use-case scenario. Three consecutive hours (15-18 GPS time) have been considered for the analysis. Satellites with highly discontinuous measurements have been discarded. The DD values have been scaled by the sine of the elevation to mitigate the impact of the typically higher noise of low elevation satellites. For each point, the error bar is reported. It can be recognized that increasing  $\tau$  the error bar increases because of the lower number of possible observation intervals. Fig. 3.27 indicates that below 100 s, for both L1 and L5 frequencies, increasing  $\tau$  the effect of random processes is averaged out. As specific examples, Fig. 3.28 shows the Allan deviation vs the averaging time  $\tau$  and the exponent of the power law noise process  $\alpha$  for satellites with long observation times. Fig. 3.28 highlights the variation w.r.t. the exponent of the power law noise process  $\alpha$  for specific satellites of the data-set shown in Fig. 3.27. In general, Fig. 3.27 and Fig. 3.28 suggest that the noise is similar to a PM white noise. For long observation times ( $\geq 1000$  s), the process noise is similar to a PM Flicker noise and increasing the number of observations further does not help in averaging out random processes.

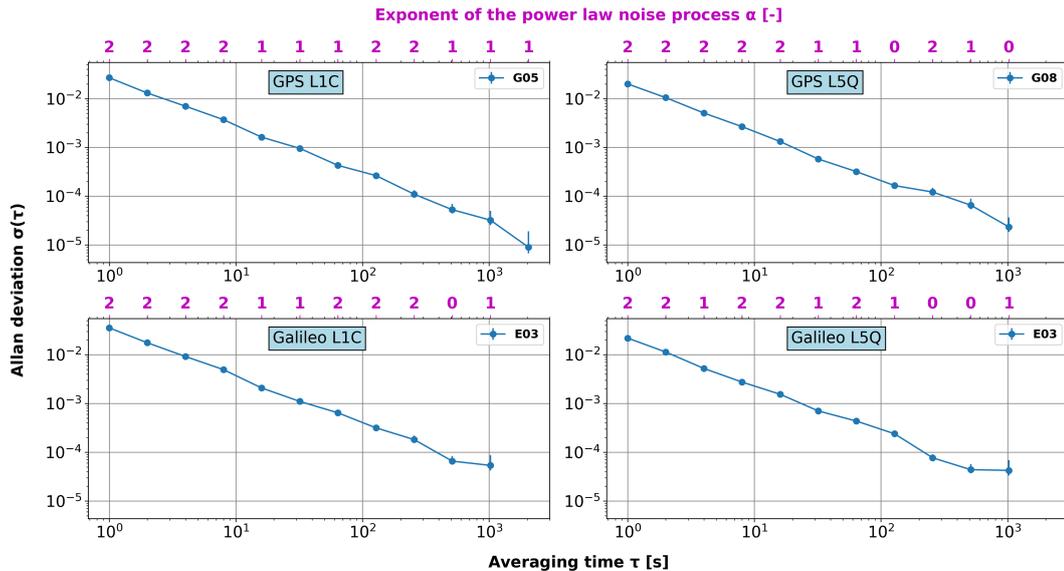


**Figure 3.27:** Allan deviation of phase DD versus the in Scenario 3. From top to bottom: GPS and Galileo. From left to right: L1/E1 and L5/E5a. The Allan deviation is shown

### 3.3.5 Expected positioning performance

The quality analysis of the smartphone measurements carried out so far allows us to draw some conclusions about the expectations for the positioning performance. The observations collected within the RF enclosure taking advantage of the geodetic-grade antenna, i.e. Scenario 1 and 2, do not exhibit phase biases in the DD. Furthermore, those observations show a phase and code DD noise that is comparable to a geodetic grade receiver. In summary, ambiguity resolution with smartphone observations is expected to be possible in Scenario 1 and 2.

The use of the smartphone antenna in an open-sky environment, e.g. Scenario 3 and 4, complicates the fixing of ambiguities to integer numbers. In fact, the impact of station dependent errors (e.g. multipath) has been demonstrated as significant in such scenarios. The phase DD computed in Scenario 3 shows large biases with short-term variations due to PCV of the smartphone antenna and multipath. In Scenario 3, although the pillar removes many



**Figure 3.28:** Example of Allan deviation of phase DD in Scenario 3. From top to bottom: GPS and Galileo. From left to right: L1/E1 and L5/E5a. The Allan deviation is shown versus the averaging time  $\tau$  and the exponent of the power law noise process  $\alpha$ . For each combination GNSS-signal, only one satellite is shown as an example.

ground reflections (see Fig. 3.25), the multipath has been demonstrated to strongly affect the measurements. On the other hand, in Scenario 4 the STD of the phase DD is smaller, and the multipath level is lower than Scenario 3. However, some residual phase biases that affect the integer nature of the ambiguity are observed.

It can be concluded that smartphone-based positioning with AR is expected to be challenging without taking care of the station-dependent errors carefully. The analysis leads us to expect AR hardly feasible in Scenario 3 and 4, especially if no PCV corrections are applied.

### 3.3.6 Smartphone-based ionospheric TEC measurements

In the framework of a collaboration with the University of Bath (UK) within the H2020 Training REsearch and Applications Network to Support the Ultimate Real-Time High Accuracy EG-NSS Solution (TREASURE) project, a quality assessment of dual-frequency smartphone-based ionospheric TEC measurements has been carried out. The quality investigation has been described in Bruno et al. (2020). The study presents a qualitative analysis for measuring the ionospheric total electron content based on more than 100 hours of smartphone phase observations. Measurements using both the Mi8 and Mate20X phones have been gathered in Scenarios 1, 2, and 3. The investigation has been performed comparing STEC differences retrieved from smartphone and geodetic receivers (with a between-receiver distance of roughly 10 m).

The STEC difference has been computed in the following way. The geometry-free linear combination was used to generate STEC measurements from both geodetic and smartphone phase observations. For a specific epoch  $i$ , a satellite  $s$  and receiver  $r$ , the geometry-free linear combination is computed as follows (Goad, 1985):

$$\phi_{r,GF}^{s,i} = (\phi_{r,1}^{s,i} - \phi_{r,5}^{s,i}) \frac{f_{GF}}{40.3 \times 10^{16}}, \quad \text{with} \quad f_{GF} = \frac{f_1^2 f_5^2}{f_1^2 - f_5^2}, \quad (3.23)$$

where  $\phi_{r,GF}^{s,i}$  is given in TECU. The geometry-free combination removes all but the frequency-

dependent parameters (e.g. station-dependent errors and phase ambiguities). After correcting the measurements for data-gaps and cycle-slips, the impact of the ambiguous nature of the phase is taken into account. As a consequence, the STEC is normalized from the entire satellite pass to the mean STEC value, computing the STEC difference in the following way:

$$\text{STEC difference} = \phi_{r,GF}^{s,i} - \frac{\sum_i^N \phi_{r,GF}^{s,i}}{N}, \quad (3.24)$$

where  $N$  is the number of epochs throughout the satellite pass. Moreover, the mean value was chosen (instead of, e.g., the initial value) to minimize the effect of the smartphone's phase noise in the normalization.

Furthermore, in that work, two data quality improvement techniques have been evaluated. The first is based on a simple C/N0 mask set to 30 dBHz as suggested in Liu et al. (2019). It can be observed that the results presented above in this chapter support this choice. The second quality check proposed in Bruno et al. (2020) is a novel technique based on two steps. Firstly, all observation data gaps are removed. Secondly, for the remaining observations, a change in TEC between two epochs higher than a given threshold is identified as cycle slip. Considering the phase noise is around 0.2 TECU for L1 for the data gathered with the smartphone antenna, and that the length of a phase cycle is around 1.1 TECU in L1, in that paper an in-between threshold value of 0.5 TECU/s is adopted. Moreover, the comparison between geodetic and smartphone STEC difference is re-initialized after every epoch identified as cycle-slip to minimize the effect. After the quality check, all valid smartphone-based STEC collected in Scenario 1-3 are compared against the STEC retrieved from the geodetic receiver on the Geo++ rooftop.

The quality of TEC relative to the geodetic receiver is compared between smartphones (Mi8 vs Mate20X), and among the different scenarios. The results obtained from the comparison between STEC differences retrieved from smartphone and geodetic receivers show excellent agreement in Scenario 1 and 2 (i.e. within the RF enclosure) when any of the filtering techniques is applied. The RMSE is smaller than 0.2 TECU in all the cases. The analysis shows that, after applying the filters, more observations are discarded in Scenario 3 rather than in the others. Moreover, the quality check proposed in the paper exhibits a similar RMSE to the C/N0-mask filter, but it rejects a smaller number of observations, i.e. 20% vs 60% of the measurements. The reader can find further details about the quality assessment and the final results in Bruno et al. (2020).

In conclusion, the investigation indicates that smartphone devices have the potential for ionospheric TEC monitoring. Moreover, in Bruno et al. (2020) a method to consider C/N0, multipath environment, and data continuity properly is proposed. The study is valuable to support the use of smartphones to enhance the global availability of low-cost TEC monitoring devices. Since the analysis suggests that Scenario 2 improves the availability of usable measurements, smartphone-based stations to sense ionosphere could be installed adopting strategies to mitigate the multipath impact. As examples, such stations could be located on areas similar to soccer fields (like in Scenario 4) as well as on the rooftop of buildings placing the smartphone away from wall reflections on supports like, e.g., large metal plates to remove possible ground reflections.

### 3.4 Positioning using smartphones

The positioning algorithm employed in this work makes use of the SSR concepts introduced in Chapter 2. In particular, for zero and short-baseline setups, the algorithm takes advantage of the observations of a geodetic-grade reference receiver with an undifferenced approach as described in Chapter 2 (see Eq. 2.44). Furthermore, in the following experiments, the two receivers involved, i.e. smartphone and geodetic receiver, are assumed to experience the same atmospheric conditions in time. Also, post-processing network-based results are investigated.

The network computation is based on the SSM concept applying the SSR corrections as described by Eq. 2.45.

In this work, epochs are considered to be fixed when at least observations from four satellites could be fixed to integer successfully. Moreover, in these cases, the ratio test has to show values higher than three, being coherent with what suggested by Euler and Schaffrin (1991). Dual-constellation, i.e. GPS + Galileo, and dual-frequency, i.e. L1 + L5, processing has been employed. Besides, for a specific epoch, a satellite is considered fixed when L1 and L5 ambiguities are fixed to an integer value. As a consequence, the Time To Fix Ambiguities (TTFA) is defined as the time needed to achieve a number of fixed satellites larger than four. The Geo++ GNSMART software was used to compute the positioning results. The GNSMART settings employed in this study are reported in Appendix C (subsection C.2) that can be used following the Geo++'s HTML-help (Geo++ GmbH, 2019).

In this section, the potential of smartphone-based positioning has been evaluated utilizing the measurements gathered in the Scenarios introduced in the previous section (see Table 3.1). Zero and short-baselines have been considered along with network-based positioning in post-processing.

#### 3.4.1 Zero and short-baselines applications

Firstly, zero-baseline tests were carried out between smartphone (Mi8 and Mate20X) and geodetic receiver, in the configuration described in Scenario 1 (see Fig. 3.3). Fig. 3.29 reports the results obtained with GNSMART and RTKLIB (Takasu, 2007). The latter was used to provide a comparison with an open-source free software. A modified demo version of RTKLIB (RTKexplorer, 2020) was used to better deal with unflagged cycle-slips. In fact, as reported in RTKLIBexplorer (2019), utilizing RTKLIB (versions 2.4.2 p13 and 2.4.3 b33), using the BCM47755 receiver, unreported cycle slips were introducing large errors into the bias states in the Kalman filter and preventing convergence. The RTKLIB demo version was used considering a *Static* positioning mode, *L1+L2+L5* (hence, L1 and L5 only in this case) frequencies, GPS and Galileo observations, with a *Forward* filter type. An elevation mask of 10 deg was applied, using broadcast ephemeris, while no atmospheric corrections were included. A *Fix and hold* ambiguity resolution was selected along with a *Min ratio to fix ambiguities* equal to three. Furthermore, a *Slip threshold to reset ambiguities* was set to 0.005 m.

In such zero-baseline configuration (see Fig. 3.3), Fig. 3.29 shows that ambiguities can be quickly fixed to integer values achieving mm-level 2D error. While using GNSMART, the ambiguities can be resolved in less than 1 min, using RTKLIB that is true for three of four analyzed cases. In fact, in one of the cases, more than 90 s is needed to fix ambiguities correctly. The reason behind it is most likely due to the Galileo constellation, which was not continuously tracked during the first minutes of the considered period. Moreover, it can be observed that some cycle-slips cause resets of the ambiguities (see the blue-colored line of Fig. 3.29b). Overall, it can be observed that in three of four analyzed cases (red, orange, and green-colored lines), the ambiguities were resolved in less than 20 s and 33 s, using GNSMART and RTKLIB, respectively. The antenna position has been considered as the reference to evaluate the positioning results. In summary, these first zero-baseline results confirm what already presented by other authors (e.g. Riley et al., 2017): cm-level positioning is feasible using observations provided by the BCM47755 chipset. Hereafter, the positioning results are computed using GNSMART.

Secondly, the measurements gathered in the setup scenarios based at the Geo++ building (i.e., Scenarios 1,2,3) have been examined in a short-baseline (roughly 10 m length) configuration with a geodetic receiver. The Mi8 and Mate20X have been tested. Fig. 3.30 compares the 2D accuracy in the different scenarios with the performance of a geodetic receiver in a short-baseline configuration (roughly 10 m). The 2D accuracy has been considered as RMSE over three one-hour measurements per scenario, recorded on three different days. The results

indicate that ambiguities were correctly resolved when there are no phase biases, i.e. when the phone is within the RF enclosure, while only a sub-meter float solution was achieved in Scenario 3. Therefore, the BCM47755 receiver has been demonstrated as capable of RTK-level positioning. In other words, this shows that cm-level accuracy can be obtained using smartphones, with a TTFA comparable to geodetic receivers. In fact, in both Scenario 1 and Scenario 2, it is possible to resolve ambiguities within a minute. In Scenario 3, the float solution converged to sub-meter accuracy within 70 seconds. It must be noted that the errors reported in Fig. 3.30, concerning Scenario 3, have an uncertainty of a few centimeters as the exact position of the antenna within the smartphone is not known to the author.

The measurements gathered in Scenario 4 have been used to evaluate the positioning performance in a short-baseline configuration. Two baselines have been analyzed: a 50 m baseline between Mate20X and a geodetic receiver on the pitch, and a 12 km baseline between Mate20X and geodetic receiver on the rooftop of the Geo++'s building. Fig. 3.31 shows the 2D error obtained in the two cases.

While, overall, the error is smaller in the 50 m baseline case, it can be observed that the two solutions seem to converge to the same value. Moreover, Fig. 3.31 indicates that a 2D error lower than 10 cm can be achieved in less than 30 minutes.

Furthermore, as already shown in Section 3.3, the comparison between Fig. 3.30 and Fig. 3.31 highlights the significant impact of the multipath on the positioning error. The investigation carried out in Section 3.3 describes the notable difference in terms of multipath and noise measurements between Scenario 3 and Scenario 4. Here, the difference is shown in the positioning domain. In fact, while in Scenario 3, more than 120 s are needed to reach a 2D error lower than 50 cm, in Scenario 4, the 2D error below 50 cm is reached within a few seconds. Again, the high multipath impact on the quality of the smartphone's measurements is demonstrated. Nevertheless, as shown in Fig. 3.16, some residual phase biases are still affecting the observations in Scenario 4. This affects the AR and no fixed epoch has been achieved. The residual phase biases might be related to the antenna of the smartphone and the resulting station-dependent effects. The impact of such errors is further investigated in Chapter 4.

The results described so far indicate that the fact of having the smartphone lying on the ground of a field removes the multipath impact significantly. Regarding multipath, Scenario 3 is a more challenging environment than Scenario 4, but closer to a real-case situation. Therefore, to further investigate the multipath impact in Scenario 3, another experiment has been set up. A ground-plane consisting of a choke-ring with removed antenna element was used as support of the device on a pillar on the rooftop, as shown in Fig. 3.32b, to reduce multipath effects.

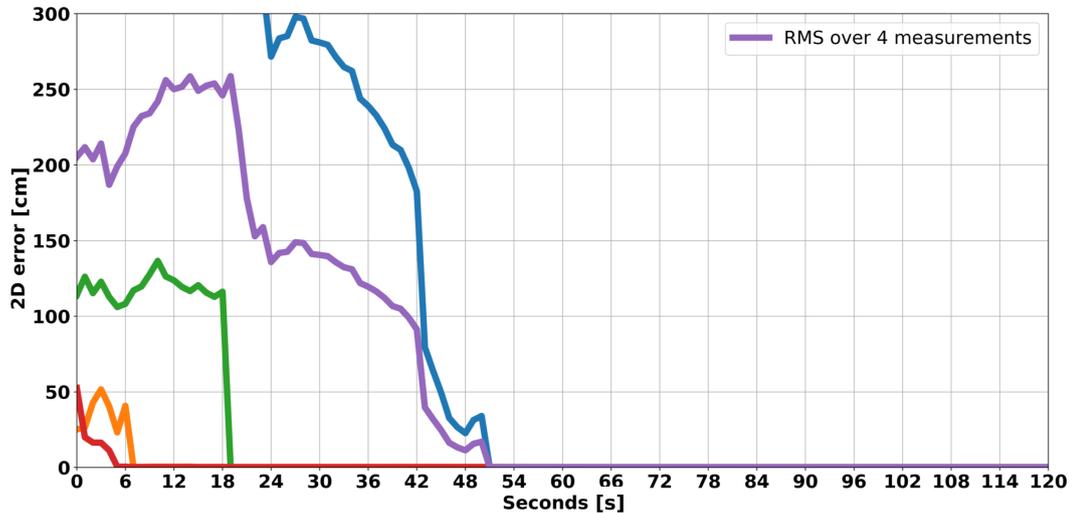
Thirty non-consecutive hours of data were evaluated, resetting the solution at the beginning of every hour. Moreover, this test is useful to study the repeatability of the convergence of the float solution using the smartphone antenna. Fig. 3.33 shows that sub-meter 2D RMSE can be achieved in less than 30 s, while sub-dm level accuracy can be reached in roughly 20 min. Therefore, it has been demonstrated that the choke-ring ground-plane helps in removing multipath effects that are strongly affecting the smartphone's measurements. Results similar to what obtained in the soccer field have been achieved. As in Scenario 4, a fixed solution is still not possible due to the remaining phase biases.

In addition, a specific experiment has been set up to further reduce the effect of multipath. The test consists in a smartphone moving on a toy train track (see Fig. 3.34a) on the Geo++ rooftop. In this way, any multipath coming from vertical reflectors (e.g. the pillars or the walls on the roof) are averaged down since the track spans a distance of several wavelengths in the horizontal directions. Additionally, azimuthal phase center variations are largely eliminated because of the two full turns taken by the train. Twenty minutes of data were collected and analyzed in post-processing, with a float-ambiguity forward-backward Kalman filter approach.

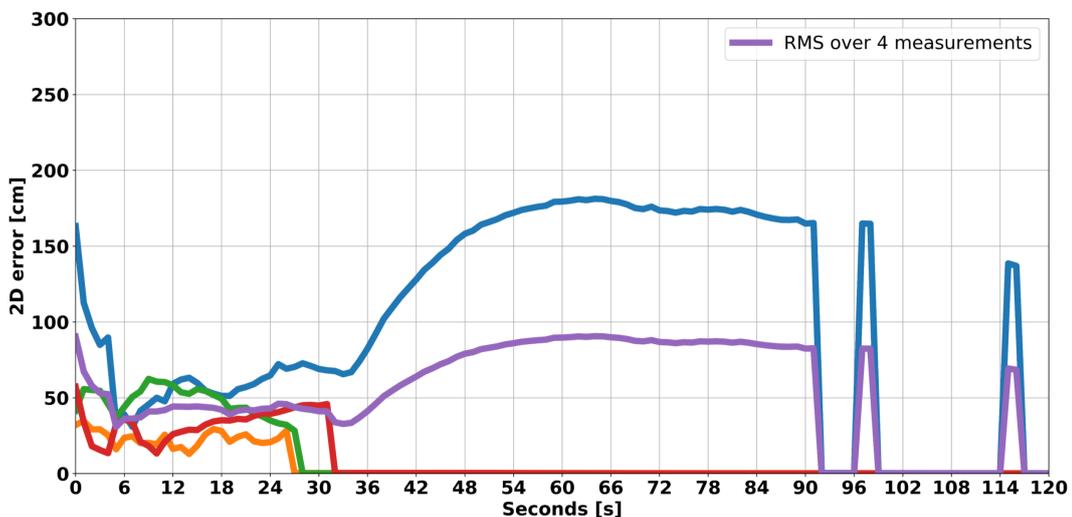
The results are shown in Fig. 3.34b together with reference track data collected with a

### 3.4. Positioning using smartphones

geodetic-grade receiver. The agreement between the two positioning solutions is always better than 10 cm. It can, therefore, be concluded that cm-level positioning accuracies can be achieved with smartphones if the level of multipath is sufficiently reduced.

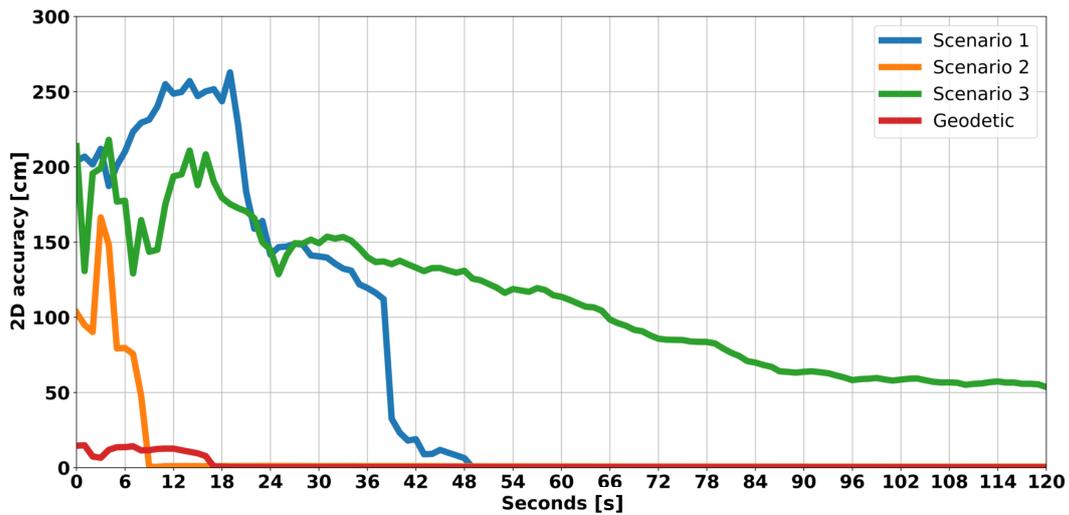


(a) *GNSMART solution.*

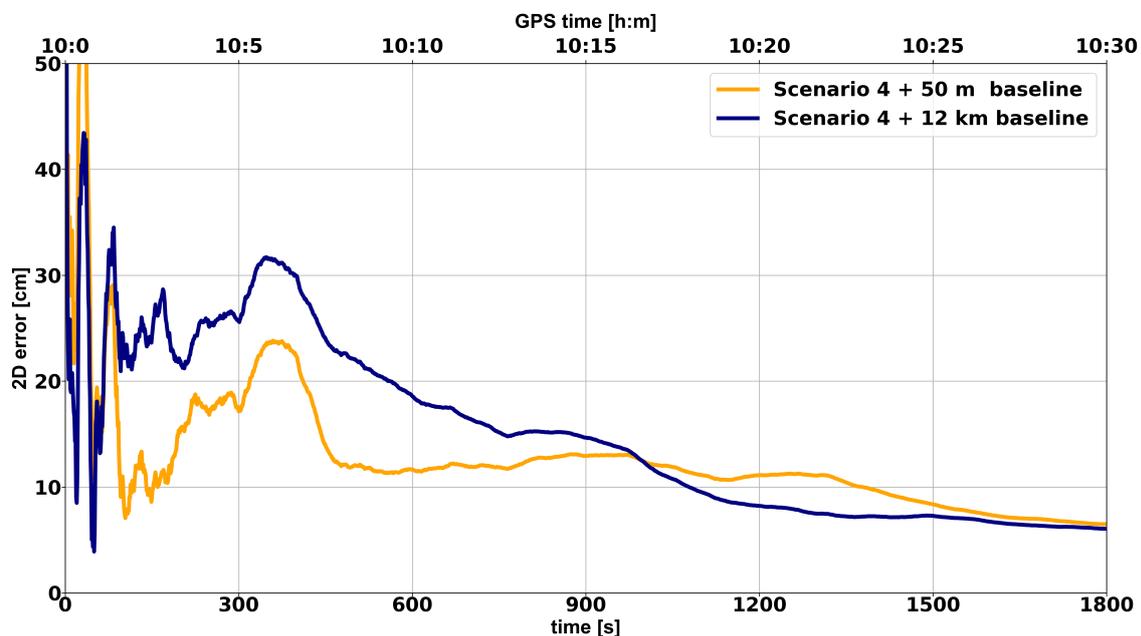


(b) *RTKLIB solution.*

**Figure 3.29:** 2D error of post-processing positioning between smartphone and geodetic receiver in zero-baseline configuration. Four different one-hour tests with Mi8 and Mate20X measurements are carried out. The RMS over the four measurements is reported in purple-colored line. Only the first 120 s are shown to highlight the AR performance. From top to bottom: results obtained using GNSMART and with RTKLIB. In the first case, overall, in roughly 50 s successful ambiguity resolution is achieved with mm-level 2D error. In the second case, one data-set needs more than 90 s to fix ambiguities and experiences resets due to cycle-slip occurrences.



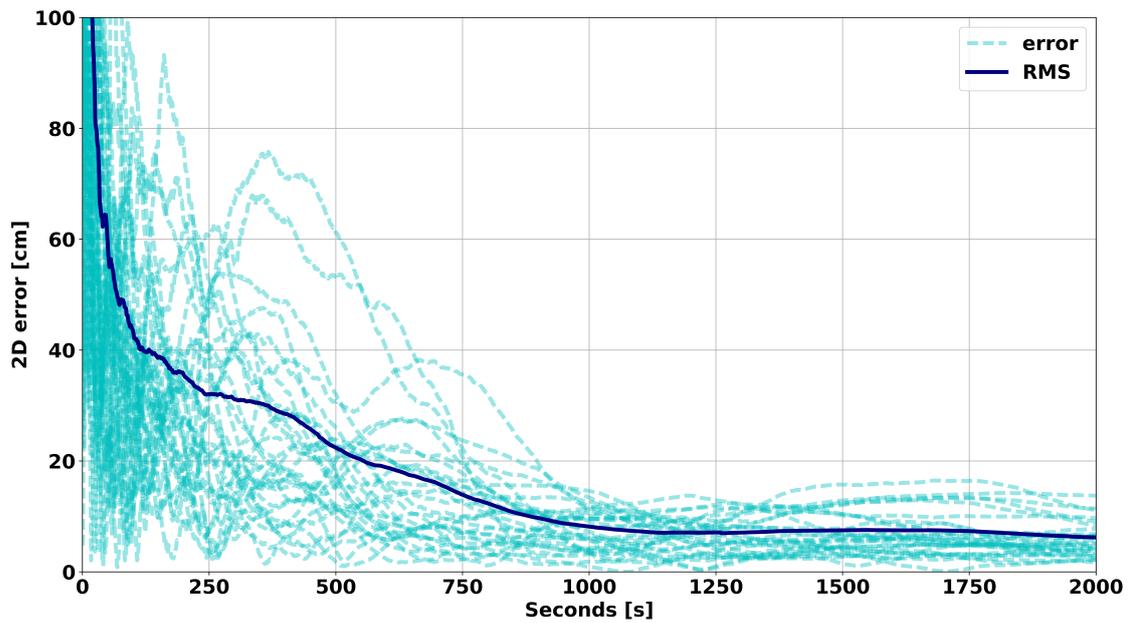
**Figure 3.30:** 2D accuracy (RMSE) of short-baseline positioning using smartphone measurements. The measurements gathered in Scenarios 1,2, and 3 have been used together with observations of a geodetic grade receiver roughly 9 m away. As a comparison the solution obtained in a similar short-baseline configuration between two geodetic receivers is reported (red-colored line). To better evaluate the AR performance only the first 120 s are shown.



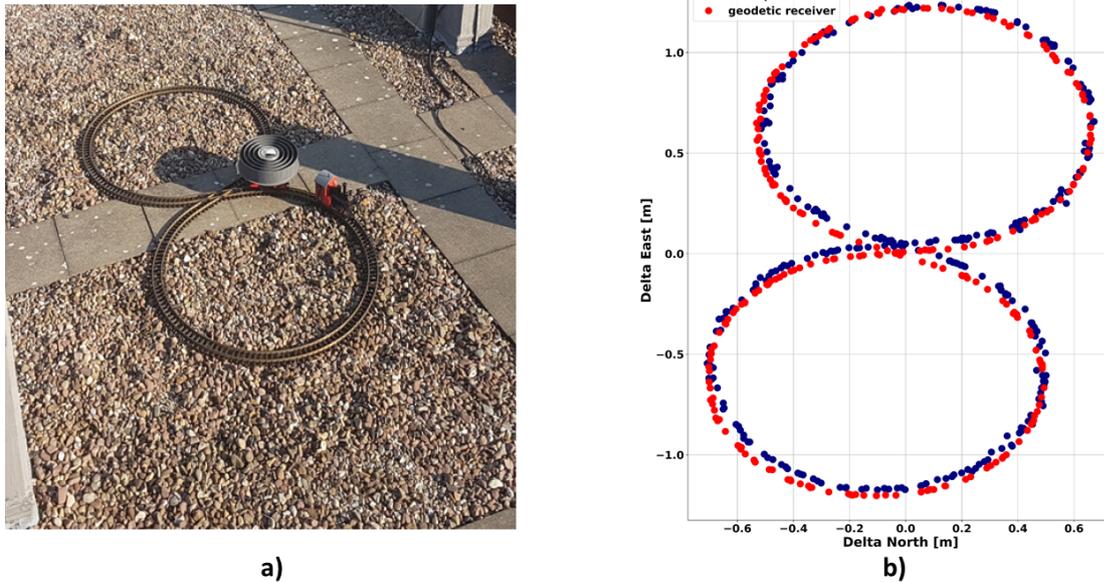
**Figure 3.31:** Positioning performance in the soccer field: the Mate20X is the rover exploiting the observations of reference stations 50 m (orange-colored line) and 12 km (blue-colored line) distant. The 2D error of the float solution is reported as no successful AR can be achieved.



**Figure 3.32:** Smartphone setups used when taking measurements on the Geo++ rooftop. a) The smartphone is lying on the surface of the pillar. b) The smartphone uses a choke ring with removed antenna element as support on the top of a pillar.



**Figure 3.33:** Short-baseline performance with scheduled reset after 1h. The blue-colored line indicates the 2D RMSE computed considering 30 hours of data with the smartphone lying on top of a geodetic choke-ring. The x axis is the time after the start of the hour, while each point of the blue-colored curve is the RMS over 30 hours of the 2D error. The light blue-colored lines depict the 2D error of each one-hour data-set.



**Figure 3.34:** Toy train experiment. From left to right: smartphone placed at the center of a geodetic choke ring carried by a toy-train over an eight shape rail-track (a), post-processing smartphone positioning (blue-colored dots) and geodetic receiver positioning (red-colored dots). The comparison between the two solutions shows differences smaller than 10 cm.

### 3.4.2 Network SSR-based positioning

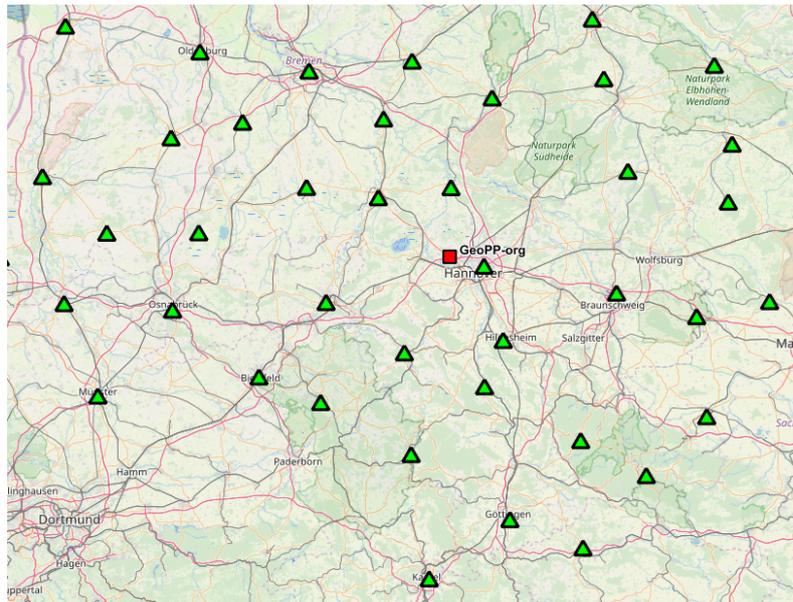
The benefit of using an expedient to mitigate the multipath for smartphone-based positioning has also been tested in real-time. The test has been performed with an Android application software (app) capable of performing SSR-based positioning. The technology used follows the SSR concepts introduced in Section 2.4. In this case, VRS observations are generated from SSR data determined from a reference station network and applied as described in Section 2.4 (see Eq. 2.45). A multi-GNSS (GPS, Galileo and GLONASS) and multi-frequency (L1, L2 and L5) network-processing was employed to compute the SSR data. The network used to generate the SSR corrections is a sub-set of the LGLN network around Garbsen, Lower Saxony, Germany, depicted in Fig. 3.35. The Geo++ building is located close to the Hannover reference station (identification number 0688). Here, only a float solution is considered because of the strong multipath impact on the phase observations discussed above. The real-time test is executed with the app that sends the smartphone measurements in RTCM format via a Wi-Fi connection to a server computing the position. The computed position is sent back to the smartphone, where the user can check and store the positioning results.

Similarly to the analysis carried out in the short-baseline configuration, two distinct setups are considered: the smartphone lying on the pillar (depicted in Fig. 3.32a) and the smartphone lying over a geodetic choke-ring on the pillar (see Fig. 3.32b). The first provides a scenario close to a real user-case (so far called Scenario 3), while the geodetic choke-ring setup shows the multipath impact on the positioning quality. While sub-meter 2D accuracy can be achieved in real-time in Scenario 3 in roughly 10 minutes, the use of the geodetic choke-ring provides benefits both in terms of accuracy and time to obtain a sub-meter solution, as shown in Fig. 3.36. The inaccurate reference position of the smartphone, due to the unknown position of the antenna, makes the evaluation of the results more challenging. Nevertheless, the obtained results suggest that sub-meter level accuracy is possible with real-time smartphone applications and highlights once more the strong impact of the type of antenna used on the achieved performance.

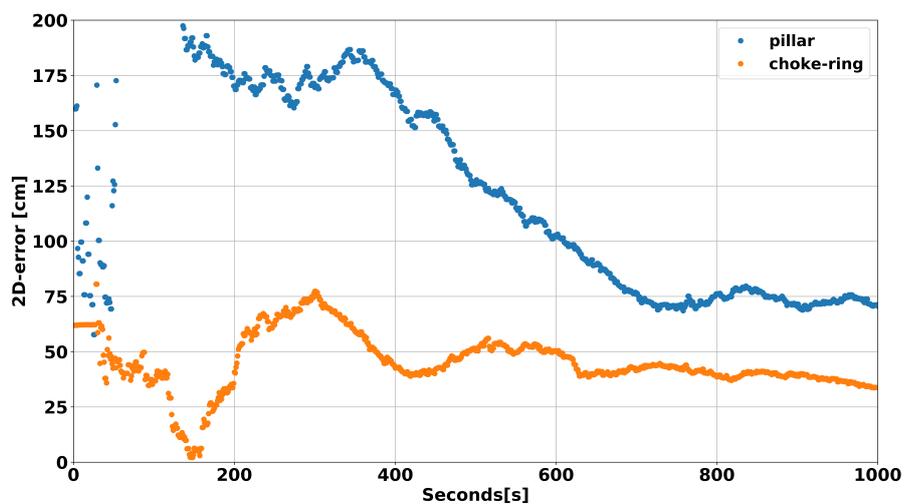
### 3.4. Positioning using smartphones

Furthermore, the potential of SSR-based positioning has been tested using smartphone's observations collected in Scenario 1. Mi8 GNSS data with the smartphone inside the RF enclosure has been analyzed. Fig. 3.37 shows the resulting 2D error in this configuration.

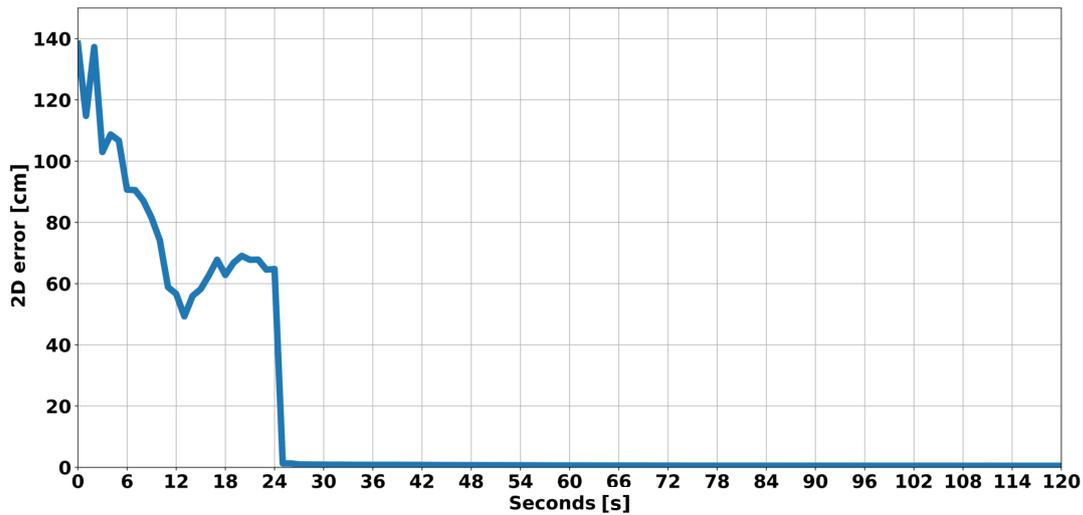
Although stronger impact of distance-dependent error sources, e.g. ionospheric effect, is present compared to the short-baseline case, ambiguity resolution is possible with a TTEFA of 25 s. After fixing ambiguities, the positioning performance quality is achieved with a 2D RMSE of 3 mm. Therefore, the potential of SSR-based positioning with successful AR using measurements from the BCM47755 receiver has been demonstrated.



**Figure 3.35:** LGLN-SAPOS sub-network used to generate the corrections used in the SSR-based positioning.



**Figure 3.36:** SSR-based smartphone positioning in real-time: 2D error with Mate20X on the pillar (blue-colored dots) and over geodetic choke-ring on the pillar (orange-colored dots).



**Figure 3.37:** Post-processing SSR-based smartphone positioning: error w.r.t. known position computed using Mi8 data collected in Scenario 1. Fixed solution is performed in 25 seconds.

### 3.4.3 SPP and PPP using smartphone's observations

In the framework of the development of the work described in this doctoral dissertation, in collaboration with other fellows of the TREASURE project and a master's student of the University of Nottingham, the GADIP3 (GNSS Android-based Dual-frequency Iono-estimated Precise Point Positioning) app has been developed (OTHiSaVRoS, 2019). This app is capable of performing GNSS-based real-time positioning in a single or dual-frequency mode, as well as considering GPS-only, Galileo-only, and GPS+Galileo system configurations.

The positioning engine employed in GADIP3 has been described in Psychas et al. (2019). Results using an SPP (e.g. Langley et al., 2017) epoch-by-epoch algorithm, which utilizes code measurements only, and a Kalman-filter-based PPP (Zumberge et al., 1997) algorithm have been presented in Psychas et al. (2019).

In that work, the authors investigated the SPP and PPP performance of a 14 hours data-set collected in Scenario 3 using the Mi8 smartphone. A single-frequency GPS-only SPP position precision of about 2.2 and 1.7 m has been achieved in the North and East components, respectively. An improvement of 40-45% in the precision of the horizontal components has been demonstrated by using the dual-frequency GPS+Galileo SPP solution. Furthermore, sub-meter positioning performance with Mi8 has been shown as possible using a static PPP technique in both real-time and post-processing. In particular, the analysis indicates that the Mi8 has the potential of horizontal sub-meter positioning with a convergence time lower than 4 minutes. For more details about the SPP and PPP analysis, the reader is referred to Psychas et al. (2019).

## 3.5 Discussion

The analysis presented a method to construct pseudorange and carrier phase observations using GNSS Android raw measurements and carefully considering the Android API flags. An investigation of the quality of the smartphone-based observations has been carried out. DD and MP combinations have been considered to assess the presence of phase-biases, code noise, and multipath level. The analysis addressed four scenarios using different configurations and

multipath environments. The use of an external antenna in a reduced multipath environment has been compared to the use of the smartphone antenna in open sky scenarios.

The results confirmed the outcome of previous works. Smartphone measurements collected in multipath-suppressed scenarios are AR-compatible. A code noise of roughly 2 m for L1, 1.2 m for E1 and less than 1 m for L5/E5a measurements and STDs of phase measurements smaller than 1 cm are observed. In short-baselines applications, solutions with successful AR were achieved in less than 50 s with a similar performance as a geodetic receiver.

In an open sky scenario, along with using a geodetic choke-ring as the basis for the smartphone to partially eliminate multipath from ground reflections, forward-backward filtering in post-processing showed cm-level agreement with a geodetic receiver in a kinematic test. Also, a real-time test was carried out, showing sub-meter-level accuracy in an open sky user-case scenario. Furthermore, the possibility to perform fixed solutions with the VRS-based technique when using the measurements collected in a repeater configuration has been shown.

It can be concluded that there is the potential for SSR-based applications using measurements retrieved by the dual-frequency devices employing the Broadcom BCM47755 chipset. However, high levels of local multipath due to the type of build-in antenna substantially limit real-time applications to accuracies of a few decimeters. Nevertheless, cm-level accuracy can be achieved in post-processing for a device undergoing a smooth motion.

The strong impact of local multipath on the smartphone's observations motivates a further investigation of the impact of station-dependent errors on smartphone-based positioning. In the next chapter, a rigorous absolute robot-based field calibration of GNSS antenna is performed for the Mate20X to estimate the PCV. Finally, the benefit of using PCV corrections is assessed for data-sets collected in Scenario 3 and 4.



## 4. PCV Impact on Smartphone-Based Positioning

---

### 4.1 Motivation

As introduced in Chapter 3, the GNSS antenna quality is a crucial factor in smartphone-based positioning. The use of an omnidirectional linearly polarized antenna in mobile devices has advantages in terms of received signal strength and the number of received signals (Pathak et al., 2003), but also makes the antenna very sensitive to multipath effects. Moreover, the smartphone antenna is affected by the other components of these portable devices, e.g. the screen of the cellphone (Xiao et al., 2019). In this chapter, the antenna pattern of the Huawei Mate20X is analyzed and the impact of antenna corrections on positioning results investigated.

Eq. 2.4 and Eq. 2.5 show that the multipath effect of GNSS measurements is not the only contributor to site-dependent errors. The other effect is related to the electromagnetism of the antenna, e.g. the PCV ( $\zeta_{r,j}^s$  in Eq. 2.5) for phase observations. However, the antenna pattern variations and multipath are almost not separable in the observations (Schmitz et al., 2002). Moreover, nowadays, high precision applications require mm-level PCV for the most extensive elevation range possible (e.g. Teunissen and Montenbruck, 2017). The full description of the antenna pattern behavior is called antenna correction, and the procedure to determine antenna center and center variations are called antenna calibration.

During the years, many groups developed antenna calibration techniques, e.g. anechoic chamber measurements, relative and absolute field calibrations. More details about anechoic chamber calibration can be found in Schupler et al. (1994). Currently, the University of Bonn is the only IGS-approved institution to provide the service of chamber calibrations, according to the IGS antenna readme available on IGS (2017). The first relative field calibrations were performed on short baselines (e.g. Rothacher et al., 1995; Mader, 1999). In that approach, the antenna is on the field together with a reference antenna. Antenna corrections are estimated relative to the reference antenna. Specific methods to perform relative antenna calibration were developed at the Geodetic Institute of the Technical University of Dresden (TU Dresden). Several institutions perform absolute antenna field calibration. Geo++<sup>®</sup> GmbH (Wübbena et al., 1997, 2000) and the University of Hannover (Menge, 2003; Kersten, 2014) are recognized as standards for robot-based absolute field calibration. The same approach is used in Australia (Riddell et al., 2015) and at the State Surveying Agency of Berlin, Germany. Another European center performing absolute field calibration is the Eidgenössische Technische Hochschule (ETH) Zürich, Switzerland (Willi, 2019; Willi et al., 2020). Comparisons between the chamber and absolute field calibration are described in Görres et al. (2006); Willi et al. (2020).

Relative antenna calibration for mass-market GNSS receivers has been performed by Biagi et al. (2018), showing particular improvements in the height component of the residuals of the positioning processing. Furthermore, Netthonglang et al. (2019) computed an approximated antenna phase center of the Mi8 smartphone by averaging the post-processing coordinates in northing and easting. In their study, after mainly removing the multipath effects, the authors show cm level relative positioning using baselines of roughly 5 and 20 km. Wanninger and Heßelbarth (2020) showed results of a relative calibration of the L1 frequency of a Huawei P30 with respect to a geodetic reference antenna. The Huawei P30 is equipped with the GNSS chip Kirin 980. In their work, the calibration has been carried out in an open sky scenario using a setup that employs a rotational device to cover the azimuthal range. Differences between individual calibrations reach up to 1–2 cm. After correcting for the PCV, they demonstrate

cm-accurate position determination, achieving 2 cm accurate positioning after 60 min of convergence. In their analysis, successful ambiguity fixing could be performed on GPS L1 only.

To the best of the author's knowledge, first results about absolute antenna calibration for smartphones has been published in Darugna et al. (2020). The work has been carried out in parallel with this dissertation. The calibration of the Mate20X phone has been performed. The calibration technique used is the robot-based absolute antenna field calibration (Wübbena et al., 1997, 2000; Schmitz et al., 2002; Rothacher, 2001; Wübbena et al., 2008). PCV corrections are estimated and analyzed for their use in smartphone-based positioning. The impact of such corrections on the positioning performance is investigated, and the outcome reported. In Section 4.2, the robot-based antenna calibration concept is presented, along with the PCV description. After the calibration, the resulting PCV patterns are discussed in Section 4.3. Successively, in Section 4.4, the positioning results after applying the antenna corrections are reported, showing solutions with ambiguities successfully fixed in open sky scenarios.

## 4.2 Absolute robot-based field calibration of GNSS antennas

For a receiving antenna, the phase pattern describes the distribution of the carrier phase delays (or advances) depending on the direction of the incoming signals (e.g. Leick et al. 2015). This pattern is related to the radiation pattern of electromagnetic waves, defined by the variation of the intensity of the electric field as a function of directions in space. Accordingly, the phase delays are not homogenous in space. They can be expressed as a function of two angles  $\varphi$  and  $\theta$ , which give the position of the source of the signal (i.e. the satellite) in a cartesian reference frame as depicted in Fig. 4.1. As presented in Rothacher et al. (1995), such a function can be expanded with spherical harmonics. Therefore, the antenna phase pattern can be described as:

$$\Psi(\varphi, \theta) = \sum_{n=0}^{n_{\max}} \sum_{m=0}^{\min(n, m_{\max})} [A_{nm} \cos(m\varphi) + B_{nm} \sin(m\varphi)] P_{nm} \cos(\theta). \quad (4.1)$$

In Eq. 4.1  $\varphi$  and  $\theta$  refer to the position of a specific satellite in the antenna coordinate system,  $P_{nm}$  are the fully normalized Legendre polynomials and  $A_{nm}$  and  $B_{nm}$  are the coefficients estimated for maximal degree  $n_{\max}$  and maximal order  $m_{\max}$ .  $\Psi(\varphi, \theta)$  is expressed in radians or degrees. Following basic rules of trigonometry  $\Psi(\varphi, \theta)$  can be described as function of the more commonly used angles azimuth  $\alpha$  and elevation  $\varepsilon$ . Where  $\alpha = 2\pi - \varphi$  and  $\varepsilon = \pi - \theta$  as shown in Fig. 4.1. The PCV are the phase pattern computed w.r.t. the electrical mean Phase Centre (PC) and the effect of a change of origin of the phase pattern can be translated into PCV (e.g. Leick et al., 2015). Therefore, assuming that the position of the PC w.r.t. the center of rotation is:

$$\mathbf{x}_{\text{pc}} = (x_{\text{pc}}, y_{\text{pc}}, z_{\text{pc}}), \quad (4.2)$$

the PCV yields:

$$\begin{aligned} \zeta_{r,j}^s(\alpha, \varepsilon) &= \frac{\lambda}{2\pi} \Psi(\varphi, \theta) - (x_{\text{pc}} \sin\theta \cos\varphi + y_{\text{pc}} \sin\theta \sin\varphi + z_{\text{pc}} \cos\theta) \\ &= \frac{\lambda}{2\pi} \Psi(\alpha, \varepsilon) - (x_{\text{pc}} \cos\varepsilon \cos\alpha - y_{\text{pc}} \cos\varepsilon \sin\alpha + z_{\text{pc}} \sin\varepsilon) \end{aligned} \quad (4.3)$$

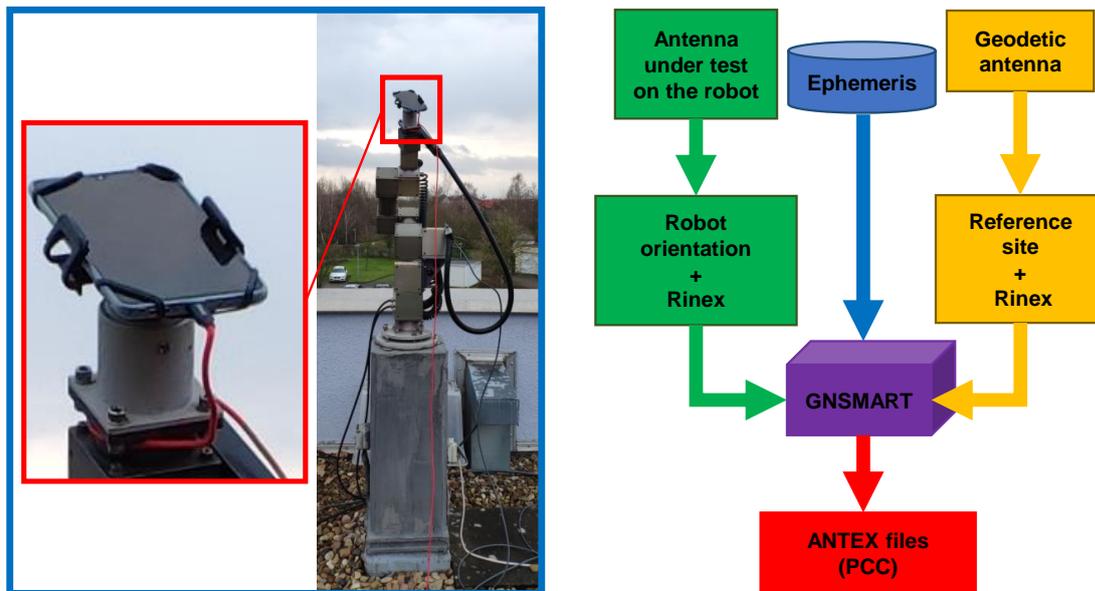
where  $\lambda$  is the wavelength of the signal and  $\zeta_{r,j}^s(\alpha, \varepsilon)$  is given in meters. Furthermore, Eq. 4.3 shows how to convert a reference center translation into PCV. In case the center of rotation is the Antenna Reference Point (ARP),  $x_{\text{pc}}$  defines the Phase Center Offset (PCO) (e.g. Leick et al., 2015). The latter is the distance vector between ARP and PC pointing towards the PC.



Spherical harmonics of degree eight and order five have been used. The values degree 8 and order 5 have been experimentally tested by Geo++ and the University of Hannover in the past. It turned out that the resolution given by degree 8 and order 5 was sufficient to model the disturbances of typical antennas while providing robust calibration results. These results are only documented in internal reports that have not been published. Moreover, the PCV is centered in order to have zero PCV values for zero values of the zenith angle. The reader can find further details about the concept of absolute robot-based calibration of GNSS antennas in Wübbena et al. (1997, 2000); Schmitz et al. (2002). At the end of the process, ANTEX files (Rothacher and Schmid, 2010) are generated containing the PCC.

### 4.3 The Huawei Mate20X antenna calibration

Figure 4.2 shows the setup for the antenna calibration of the smartphone and the data-flow to compute the corrections. The Mate20X was mounted on the robot, aligning the center of the smartphone with the rotational center of the robot. The smartphone's observations acquired during the calibration and GNSS observations from a geodetic reference station are processed using the Geo++'s GNSMART software. In the following part of this Section, the results obtained in the form of PCO and PCV are introduced.



**Figure 4.2:** Smartphone antenna calibration: setup and concept. From left to right: smartphone on the robot for the absolute field calibration and simplified processing concept.

#### 4.3.1 Analysis of the Mate20X's PCV

The magnitude of the PCV is shown in Fig. 4.3. PCV magnitudes up to  $\sim 2$  cm and  $\sim 4$  cm are observed for L1 and L5, respectively, presenting formal STDs ( $1\sigma$ ) lower than 1.6mm as shown in Fig. 4.4. Furthermore, the  $1\sigma$  formal STD is zero for elevation angles of 90 deg, as the PCV is centered on being zero at  $\theta = 0$ , i.e. elevation of 90 deg. It is worth mentioning that these STD values are related to the variance-covariance matrix of the whole state estimation process. Consequently, they are affected by both the estimation of the parameters of the spherical harmonics and the quality of the observations. Hence, the large values (e.g. for  $\alpha \in [279^\circ, 315^\circ]$  and  $\varepsilon \leq 10^\circ$  for L5, see Fig. 4.4b) might be due to either bad quality observations or a lack of availability of observations. Depending on the computation, the  $1\sigma$  STD value in Fig. 4.4 might

be too optimistic. However, it provides valuable insight into the expected quality of the calibration w.r.t. azimuth and elevation. For a complete analysis, Table 4.1 includes also the variation of the PCV magnitude. This variation is computed as the STD of the estimated PCV over the complete range of azimuth and elevation. It can be observed that the PCV have a magnitude variation of 7 and 10 mm for L1 and L5, respectively. The largest magnitudes of the PCV occur for azimuthal angles  $\alpha \in [270^\circ, 360^\circ[$  for the L1 frequency (see Fig. 2) and for  $\alpha \in [230^\circ, 360^\circ]$  for the L5 frequency (Fig. 4.3). Comparing Fig. 4.1b and Fig. 4.3 it can be observed that the largest absolute values of PCV are in directions of the major part of the smartphone's body w.r.t. the antenna phase center locations. The smartphone components (housing and active electronics) as well as near field effects in that direction might affect the signal reception resulting in larger PCV. The Mate20X PCV are larger than those of a typical rover antenna that shows typically PCV lower than 10 mm with variations lower than 2 mm.

As a comparison, the PCV of a geodetic rover antenna is shown in Fig. 4.5. Here, the geodetic rover antenna is the JAVAD TRIUMPH-1M NONE 25086, which is used in the experiment presented in the following sections. It is worth mentioning, that it is not a typical rounded-shape antenna, but squared, and it shows more variability for elevation angles close to zero than a rounded-shape antenna. Comparing Fig. 4.3 with Fig. 4.5, it can be observed that the Mate20X exhibits a much less homogeneous pattern. Furthermore, while the Mate20X manifests maximum values up to 2-4 cm, the JAVAD rover antenna has maximum PCV values lower than 1 cm with variations of roughly 2 mm for both L1 and L5. Moreover, the Mate20X shows large PCV absolute values (i.e., larger than 2 cm), even for elevation angles higher than 40 deg (e.g., see Fig. 4.3).

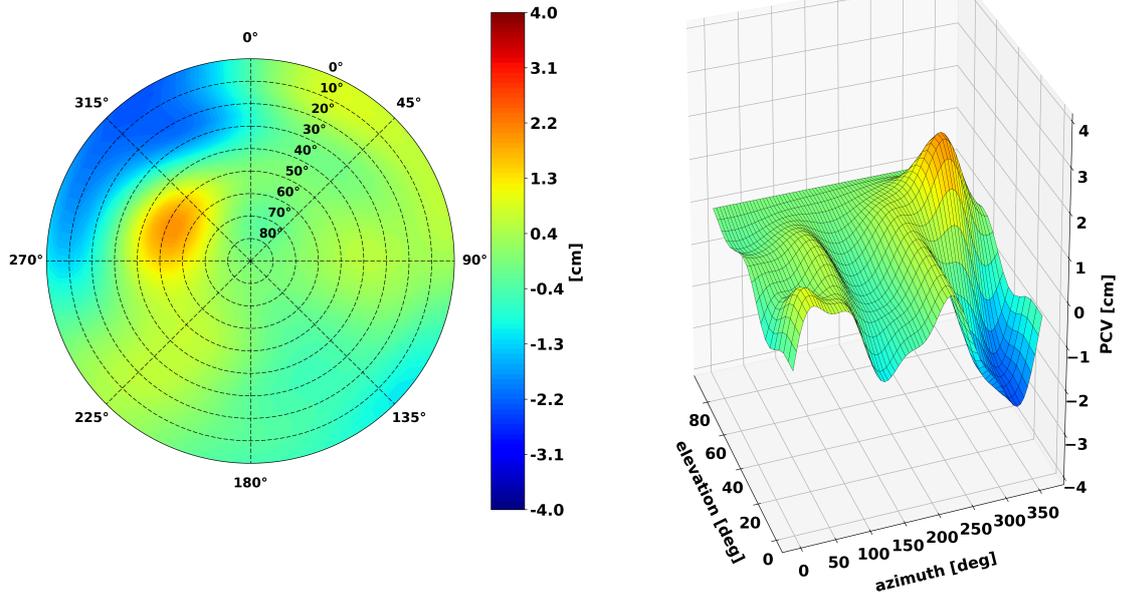
Especially during periods of strong ionosphere activity, the ionosphere-free linear combination (L0) can provide a better insight into the impact of antenna corrections on the precision of the positioning results (Schmitz et al., 2002). Fig. 4.6 shows the PCV of the L0 combination of the L1 and L5 signals. Table 4.1 indicates that the L0 maximum value for PCV corrections is 7.4 cm with a STD of 1.7cm. These values are much larger than what is presented, for example, in Schmitz et al. (2002), i.e. a STD between 1 and 4 mm for L0. The accuracy in the positioning domain can be correlated to the uncertainty in the phase through the Position Dilution Of Precision (PDOP) as described by the following expression (Schmitz et al., 2002):

$$\sigma_P = \sigma_\phi \text{PDOP}, \quad (4.4)$$

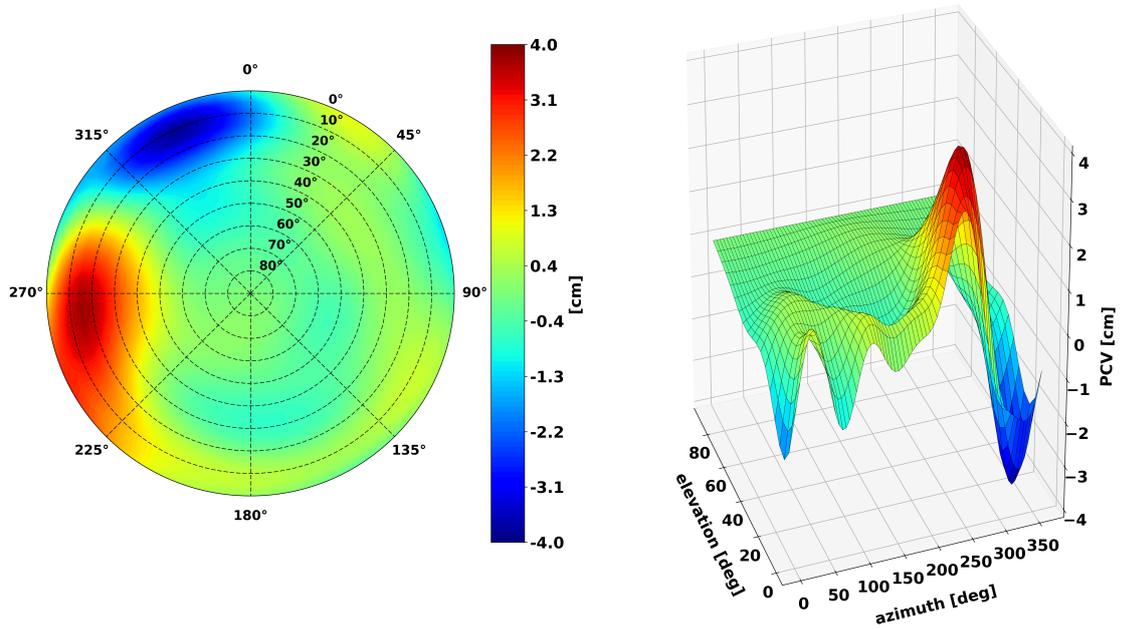
where  $\sigma_P$  is the standard deviation of position and  $\sigma_\phi$  the standard deviation of the phase observations. Therefore, the impact of PCV on the position can be computed using Eq.4.4. Considering L0 for the computation of the impact, as proposed in Schmitz et al. (2002), and a good geometry configuration exhibiting a PDOP varying between 1 and 3, the standard deviation of position can vary up to 5.1 cm. This value does not agree with high precision positioning requirements. As a consequence, PCV corrections seem to be essential for high precision applications using smartphones' measurements.

**Table 4.1:** Summary of the PCV characteristic values for the Mate20X's antenna. The results are reported in meters with mm resolution. The repeatability is reported as the maximum deviation from type mean shown in Fig. 4.9.

Frequency	Max abs. value	Magnitude variation	Max formal STD	Repeatability
L1	0.020 m	0.007 m	0.001 m	0.004 m
L5	0.038 m	0.010 m	0.002 m	0.012 m
L0	0.074 m	0.017 m	-	-

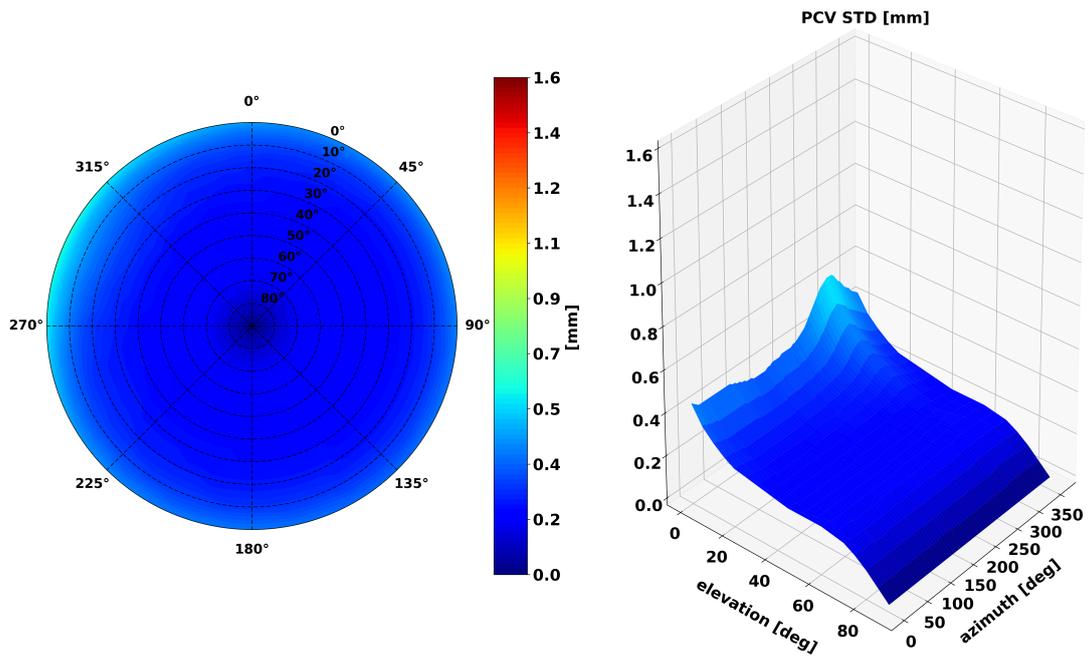


(a) Mate20X PCV for L1 signal.

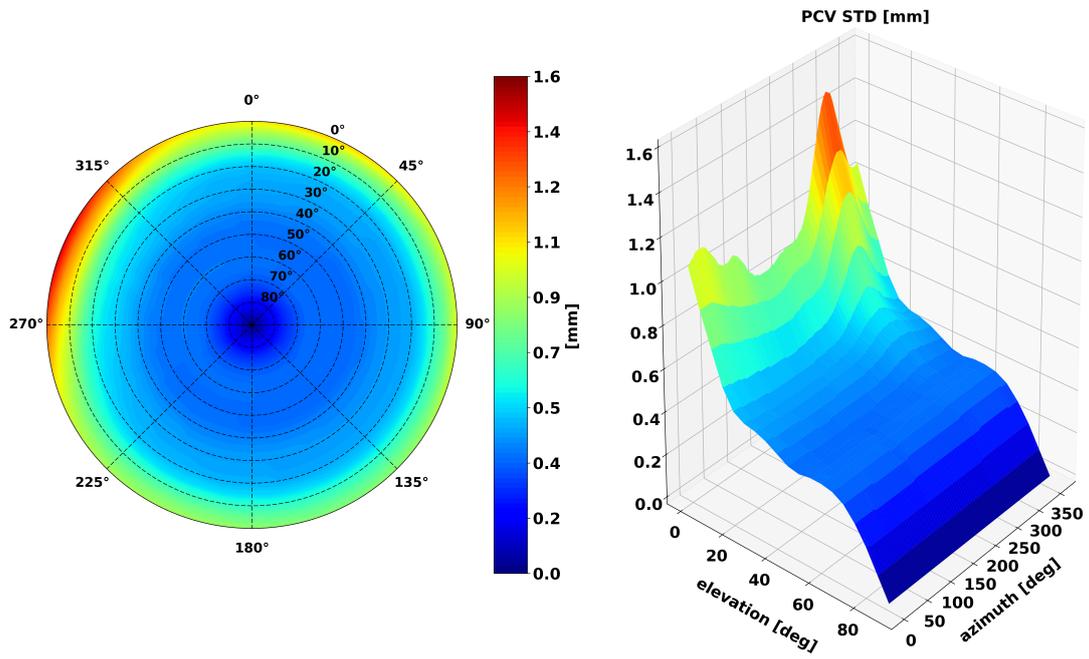


(b) Mate20X PCV for L5 signal.

**Figure 4.3:** PCV of the Mate20X antenna for L1 and L5 signals.

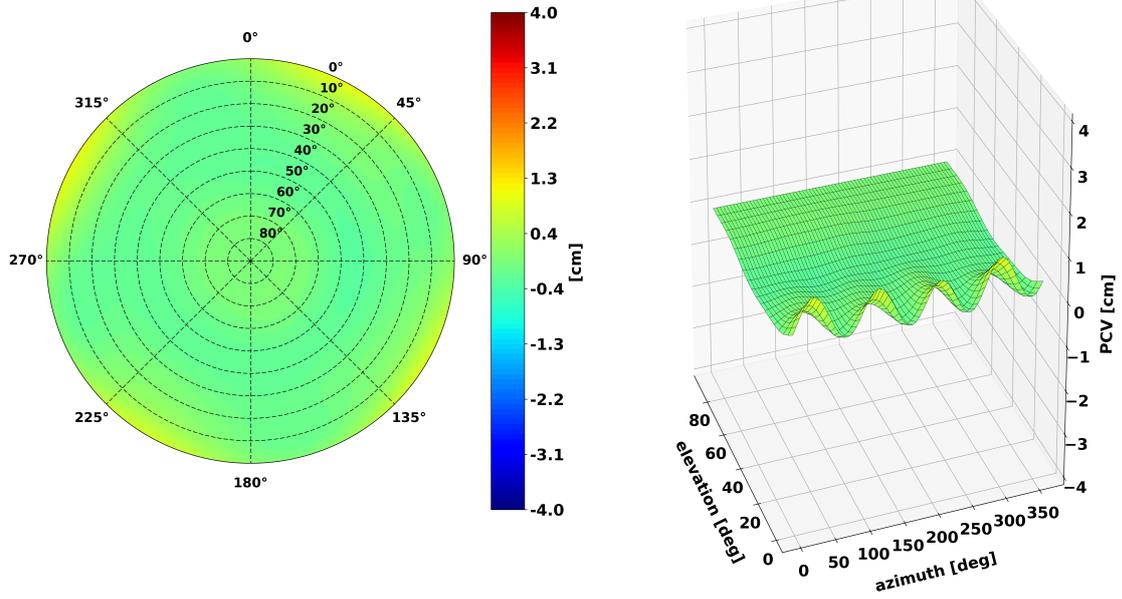


(a) *Mate20X* PCV estimation STD for L1 signal.

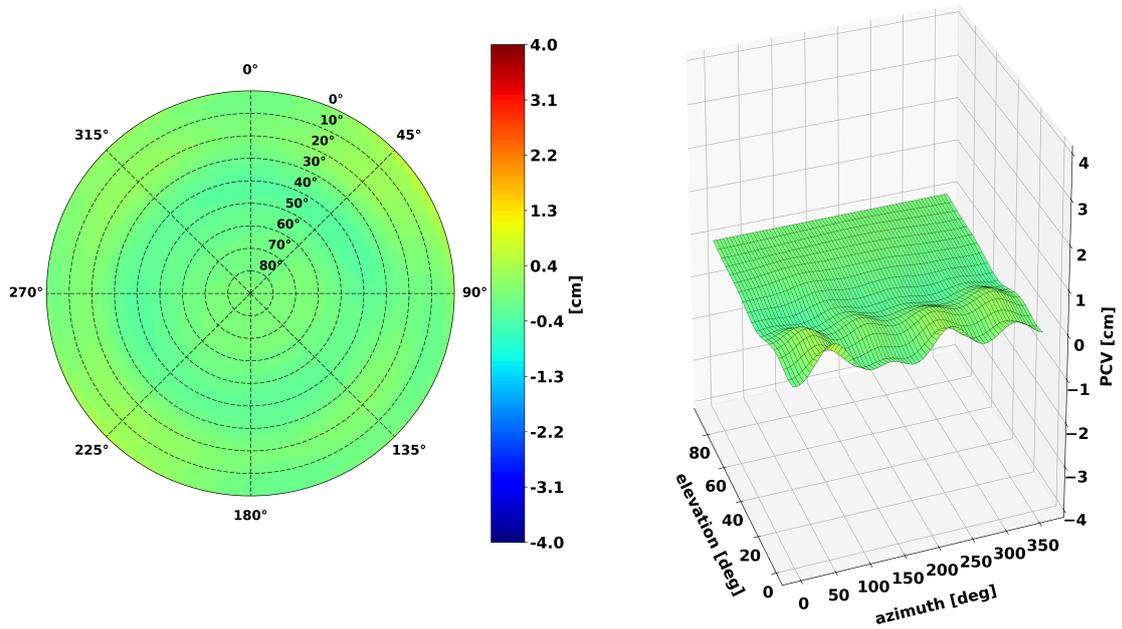


(b) *Mate20X* PCV estimation STD for L5 signal.

**Figure 4.4:** PCV formal STD of the Mate20X antenna for L1 and L5 signals. For the sake of visualization, the elevation-axis is inverted compared to Fig. 4.3.

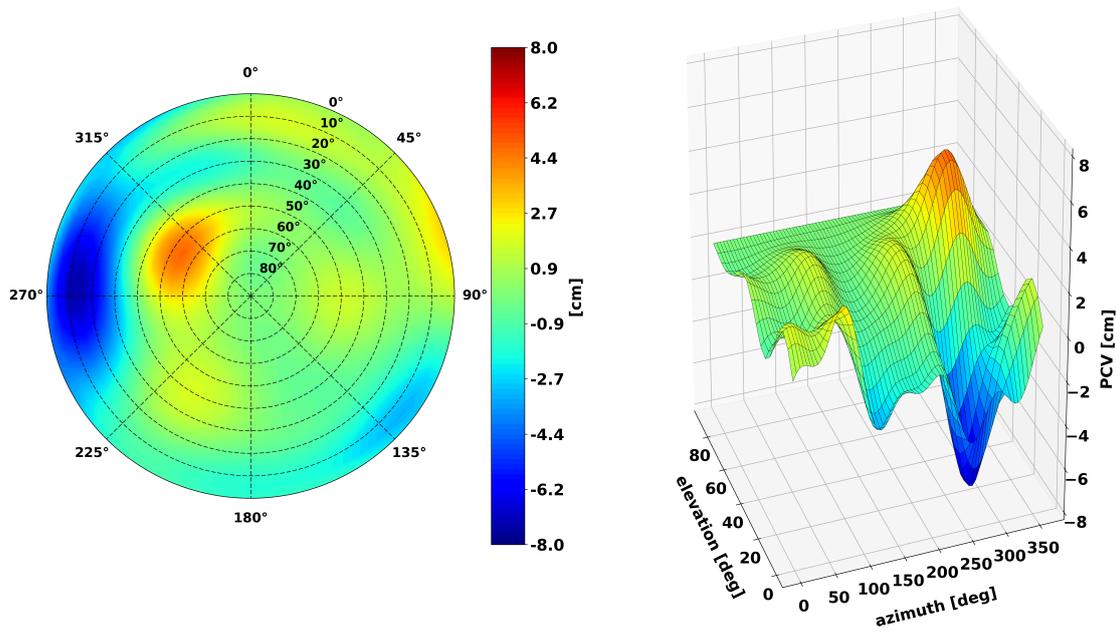


(a) JAVAD rover PCV for L1 signal.



(b) JAVAD rover PCV for L5 signal.

**Figure 4.5:** Example of PCV of a geodetic rover antenna. The PCV of the JAVAD TRIUMPH-1M NONE 25086 antenna are shown for L1 and L5 signals.



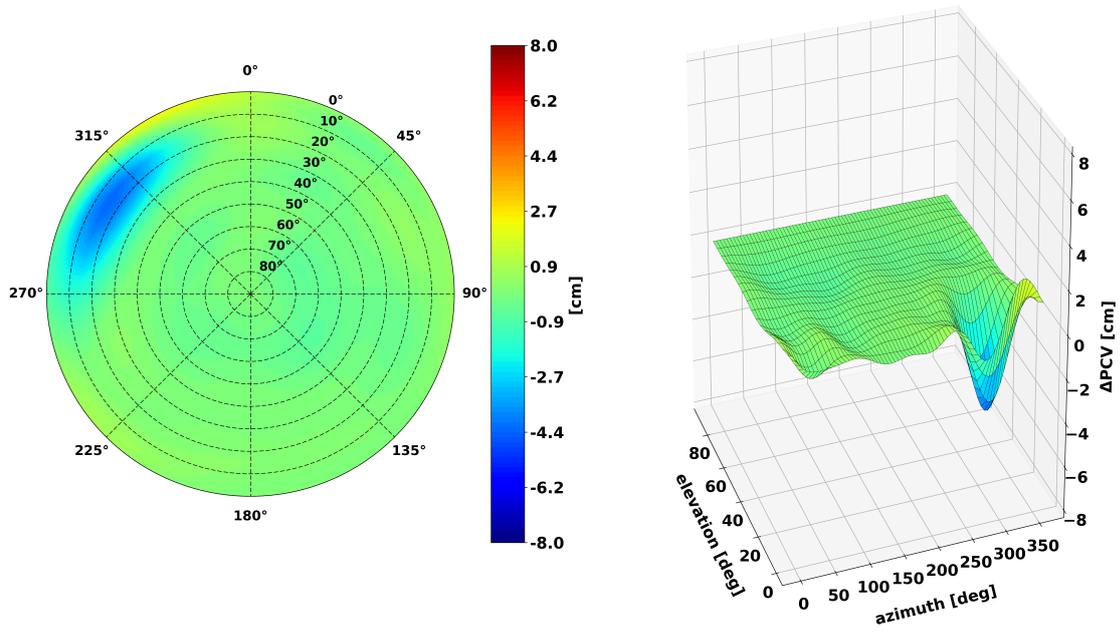
**Figure 4.6:** PCV of the Mate20X antenna for ionosphere-free linear combination.

### 4.3.2 PCV repeatability

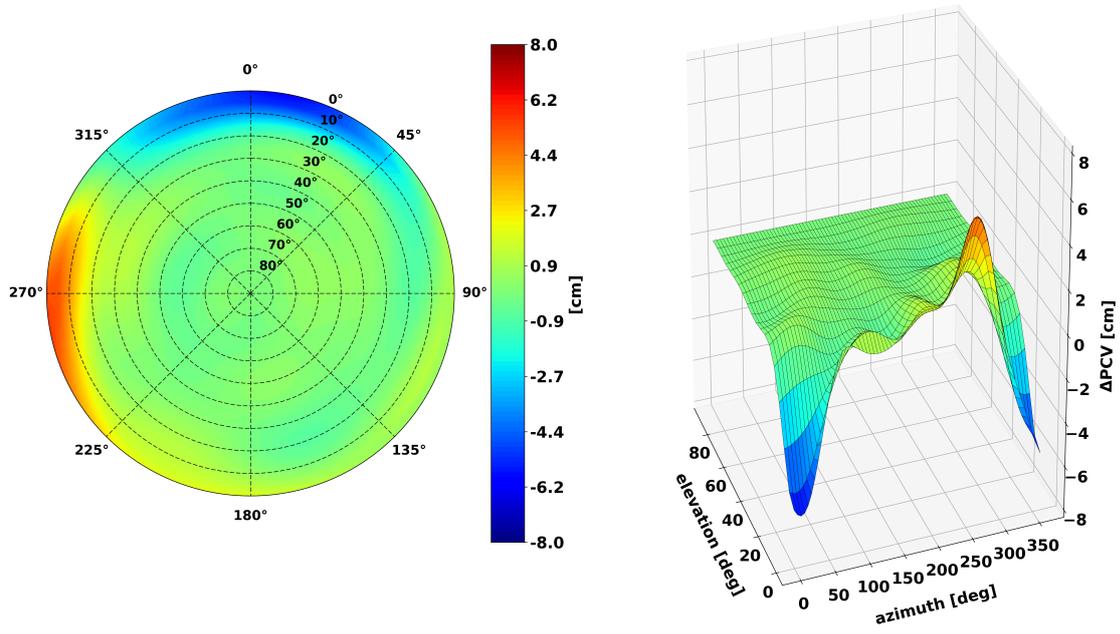
Twelve distinct antenna calibrations have been carried out to assess the repeatability of the PCV. A single antenna calibration duration goes from a minimum of six hours to a maximum of 37 hours. As an example, Fig. 4.7 shows the magnitude of the difference between a single calibration and the type-mean. In the type-mean, a rigorous adjustment of the spherical harmonics is performed using the complete variance-covariance matrix of the individual calibrations (Wübbena et al., 2006). Some elevation-dependent conclusions can be drawn by looking at Fig. 4.7. The agreement between the type-mean and the individual calibration is better than 5 mm for elevations higher than 20°. For low elevations, significant discrepancies are visible for the azimuth angle ranges mentioned above. This is uncommon for the antenna calibration and may be attributed to the capability to calibrate the smartphone antenna in those particular elevation and azimuth regions.

Fig. 4.8 shows the empirical STD of the deviation from the type-mean over the complete set of twelve separated calibrations. Again, it can be observed that the most significant differences are for elevation values lower than 20° and azimuth values larger than 260°. In particular, the values are high for elevations near to zero and azimuth  $\alpha \in [270^\circ, 315^\circ]$  for both L1 and L5 frequencies. Furthermore, it can be observed that the STD has larger values for L5 than for L1. The latter could be due to the lower availability of L5-capable satellites (because not all GPS satellites broadcast L5) during the calibration process. The STD analysis is also consistent with the formal STD shown in Fig. 4.4. It is worth mentioning again that the elevation dependency considerations are based on the centering of the PCV, i.e.  $PCV(\theta = 0) = 0$ .

For each calibration, the elevation-dependent deviation from the type-mean is computed and shown in Fig. 4.9. The PCV differences indicate a deviation up to 4 mm for L1 and 12 mm for L5. These values are more significant than what would usually be expected for a rover antenna, i.e. below ~4 mm at the horizon and, on average, roughly ~1 mm between 15-20 deg.

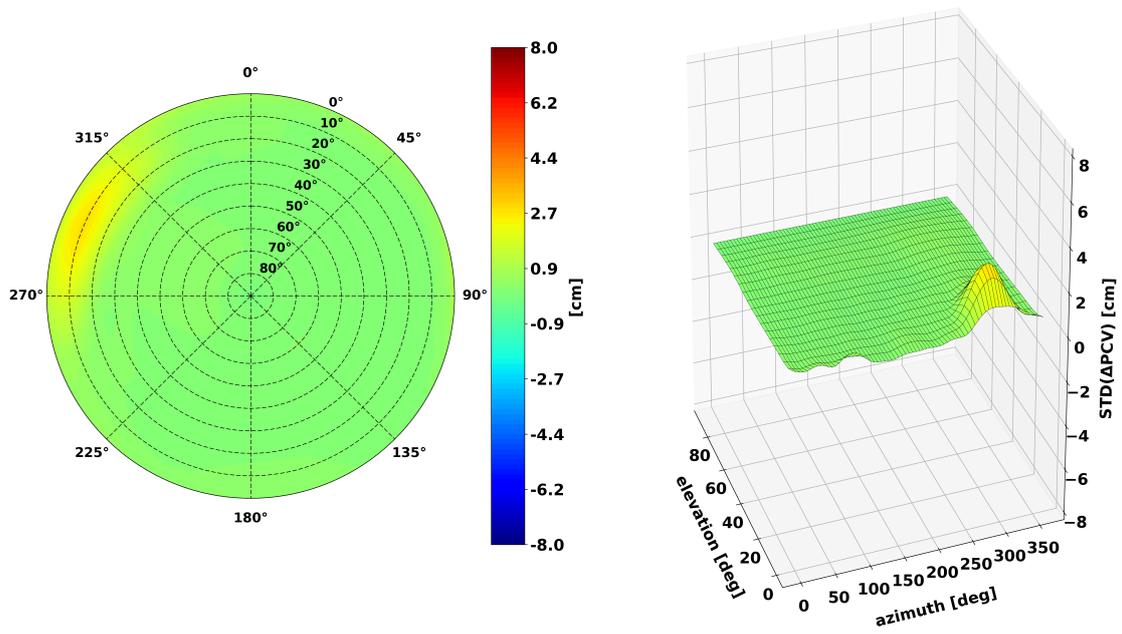


(a) *Mate20X PCV difference for L1 signal.*

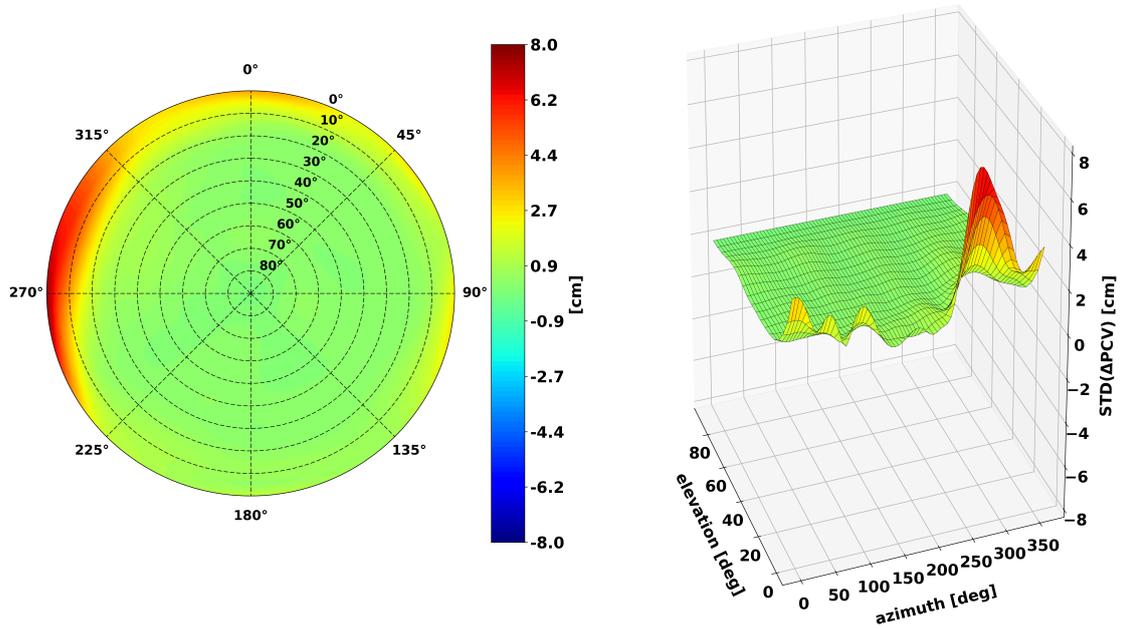


(b) *Mate20X PCV difference for L5 signal.*

**Figure 4.7:** Example of PCV difference between a single run and the type-mean of the Mate20X antenna for L1 and L5 signals.

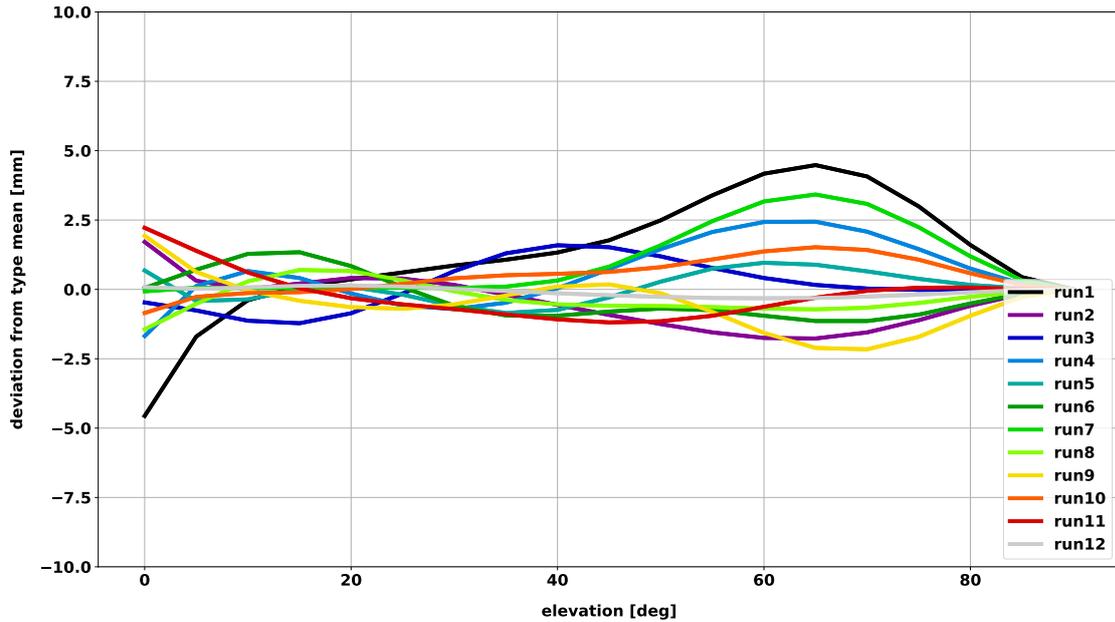
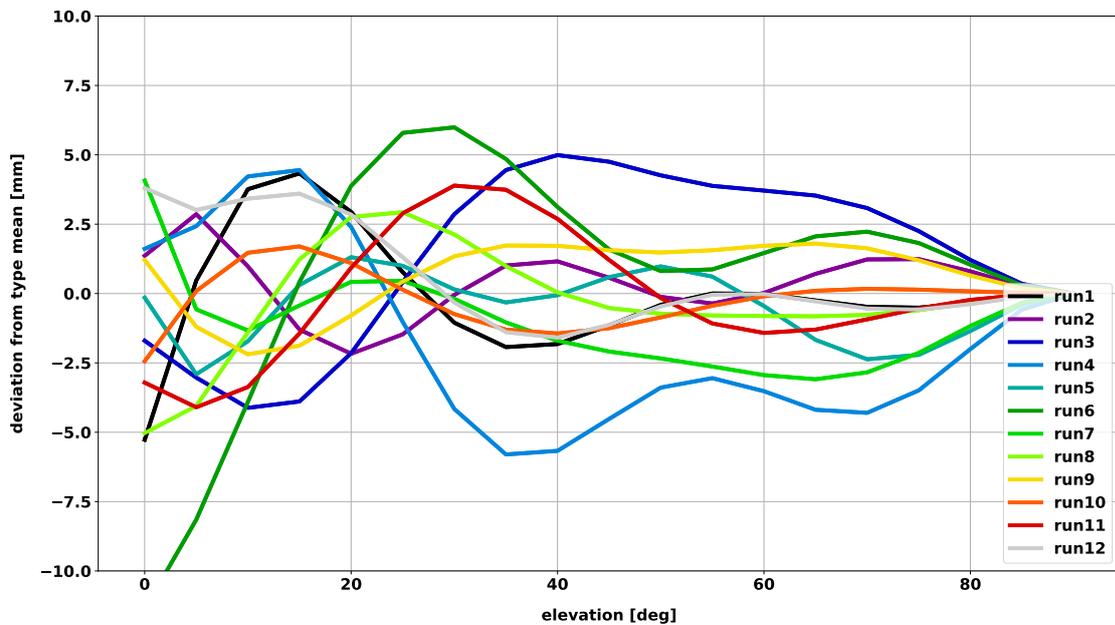


(a) STD Mate20X PCV difference for L1 signal.



(b) STD Mate20X PCV difference for L5 signal.

**Figure 4.8:** STD of PCV difference between the individual runs and the type-mean of the Mate20X antenna for L1 and L5 signals.

(a) *Mate20X PCV repeatability for L1 signal.*(b) *Mate20X PCV repeatability for L5 signal.*

**Figure 4.9:** PCV elevation dependent deviation from the type-mean of the Mate20X antenna for L1 and L5 signals of the individual runs.

To the best of the author's knowledge, no published information concerning the Mate20X antenna type and location is available. However, following what has been reported by other authors (Banville et al., 2019; Lachapelle and Gratton, 2019), the Mate20X is equipped with an omnidirectional linearly polarized antenna. Different factors might contribute to the larger PCV variation of the L5 differences. The tracking performance, in combination with the geometry of the constellation of L5-capable satellites, is not optimal (because not all the GPS satellites broadcast L5). Fig. 4.1b shows the location of the estimated antenna centers. In Fig. 4.1b, the orientation angle is the azimuth angle introduced in Eq. 4.3. In the calibration setup, the plug of the charger is defined as the North Reference Point (NRP), as shown in Fig. 4.1b. The

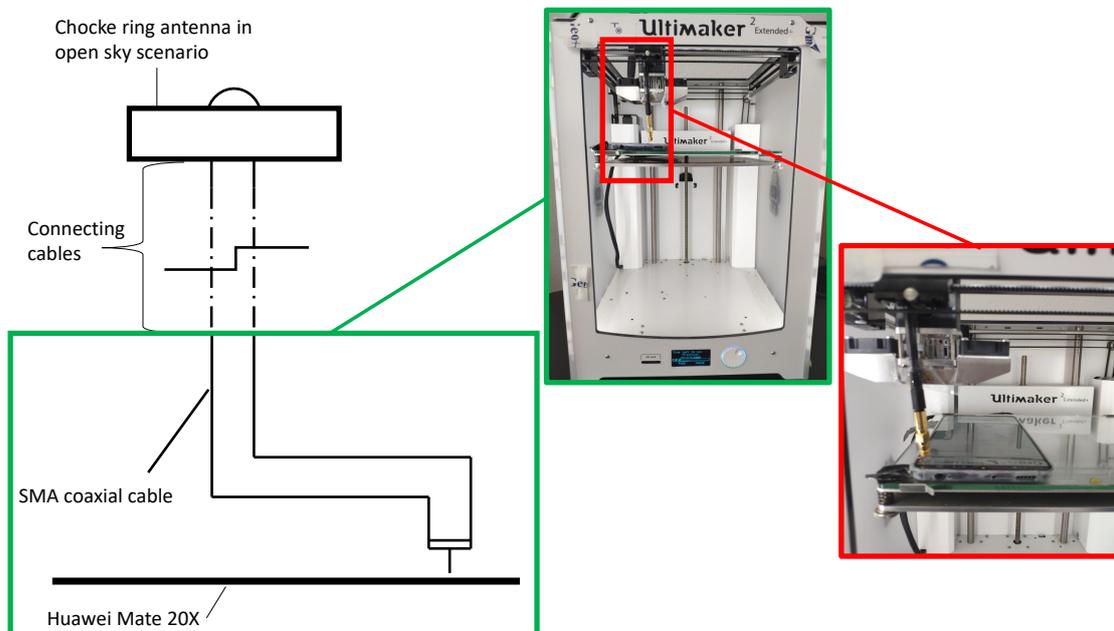
1 cm distance (in E-W direction) between the two estimated centers is in agreement with the use of two distinct antennas for L1 and L5. Also, the Up component of the PCO is 2 mm and 7 mm for L1 and L5, respectively.

Table 4.1 gathers together the main values of the PCV resulting from the antenna calibration. The maximum absolute values, the variation of the PCV magnitude, the maximum formal STD ( $1\sigma$ ), and the repeatability have been reported. In Table 4.1, the repeatability is considered as the maximum deviation from the type-mean among all the individual calibrations.

#### 4.3.3 Smartphone antenna location

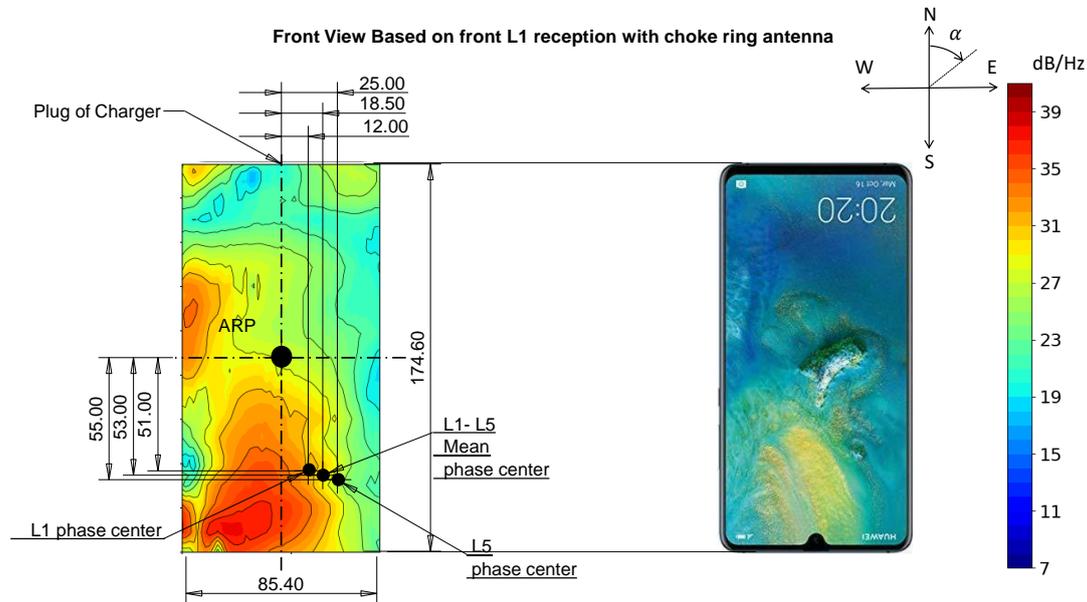
The ANTEX file output of the antenna calibration provides the north, east, and height eccentricities of the estimated antenna phase center relative to the ARP (Rothacher and Schmid, 2010). However, the antenna location can be different from the estimated phase center position, because the latter depends on the interaction among the electric components. Furthermore, often, smartphone manufacturers employ two antennas for L1 and L5, respectively. The difference between the estimated L1 and L5 phase centers shown in Fig. 4.1b depends on the different antenna response for different frequencies. To investigate the location of the GNSS antennas in the Mate20X the experiment shown in Fig. 4.10 has been set up.

The test has been carried out using a sub-miniature version A (SMA) coaxial cable connected to a choke-ring antenna in an open sky scenario. Through the SMA cable, the signal is re-transmitted and restricted to a particular position on the screen. The signal has been directed to a specific location using a small piece of wire. The piece of wire was adopted to extend the conductor element and create a small self-built antenna (see Fig. 4.10). Different locations over the smartphone screen have been tested by taking advantage of the programmable movements of the *Ultimaker 2 Extended+* 3D printer's nozzle. The SMA cable has been attached to the nozzle to move together. The nozzle has been programmed to go to specific positions in an orchestrated way.

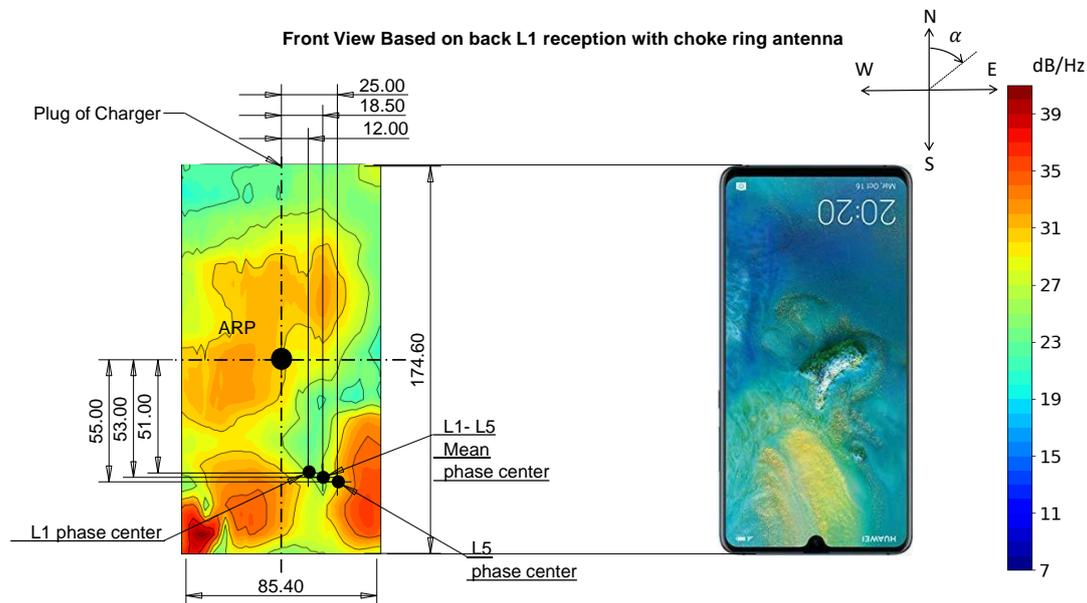


**Figure 4.10:** Sketch of the experiment carried out to test the antenna location resulted from the calibration. An SMA connector is used to direct the signal received from a geodetic antenna in an open sky scenario on specific locations on the screen of the smartphone. The L1 and L5 antenna centers output of the calibration have been tested along with the bottom part of the screen. C/N0 results are reported in Fig 4.11 and Fig. 4.12.

All the smartphone's surface has been covered by moving the nozzle with a velocity of 1 mm/s. At the same time, the GNSS measurements have been collected in the RINEX format using the Geo++ RINEX Logger (Geo++ GmbH, 2017). The experiment has been repeated twice. Once with the screen of the smartphone facing the antenna, and another time with the back facing the antenna. Fig. 4.11 and Fig. 4.12 show the obtained results in terms of C/N0 over the device surface.



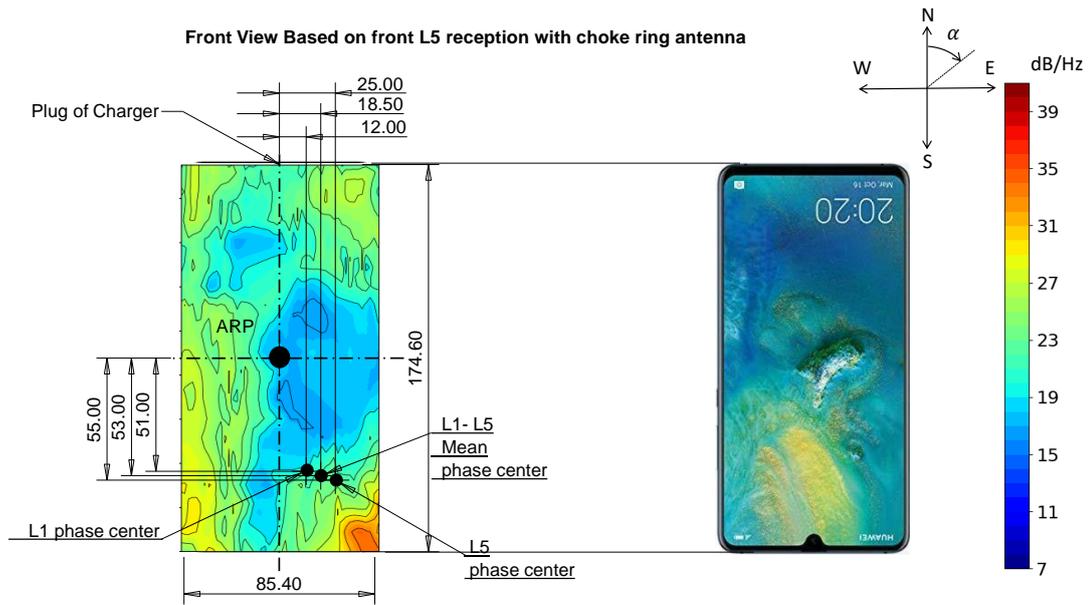
(a) Smartphone with screen up.



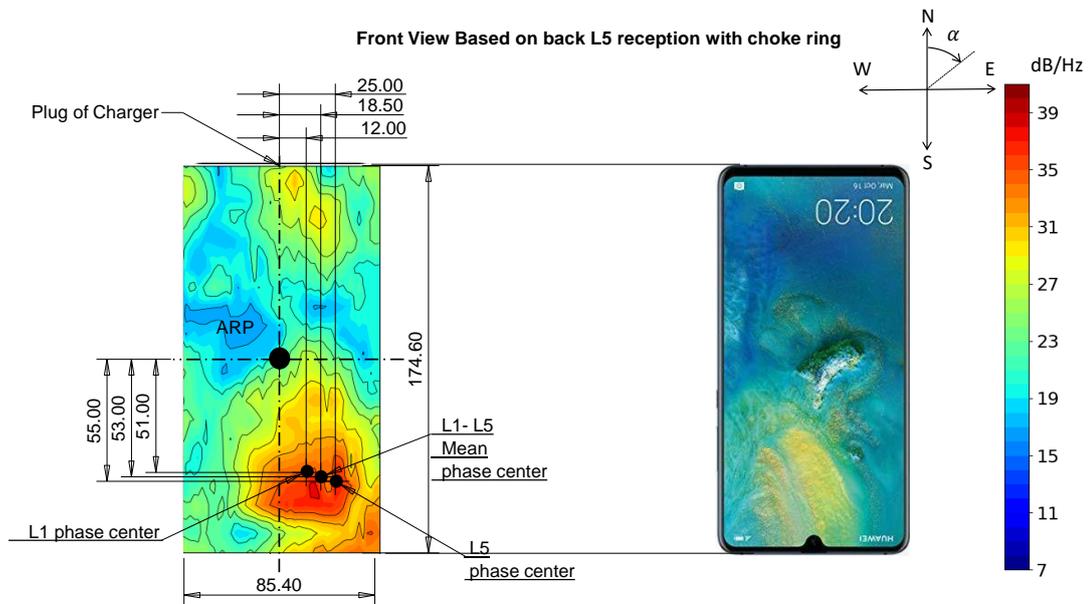
(b) Smartphone with screen down.

**Figure 4.11:** C/N0 color-map of the L1 reception on the front side of the Mate20X. From top to bottom: (a) the screen of the smartphone faces the antenna as shown in Fig. 4.10, (b) the back of the smartphone faces the antenna. The results obtained in the case (b) were mirrored to be consistent with those of case (a). The mean of the two estimated phase centers is reported as *L1-L5 Mean phase center*.

### 4.3. The Huawei Mate20X antenna calibration



(a) Smartphone with screen up.



(b) Smartphone with screen down.

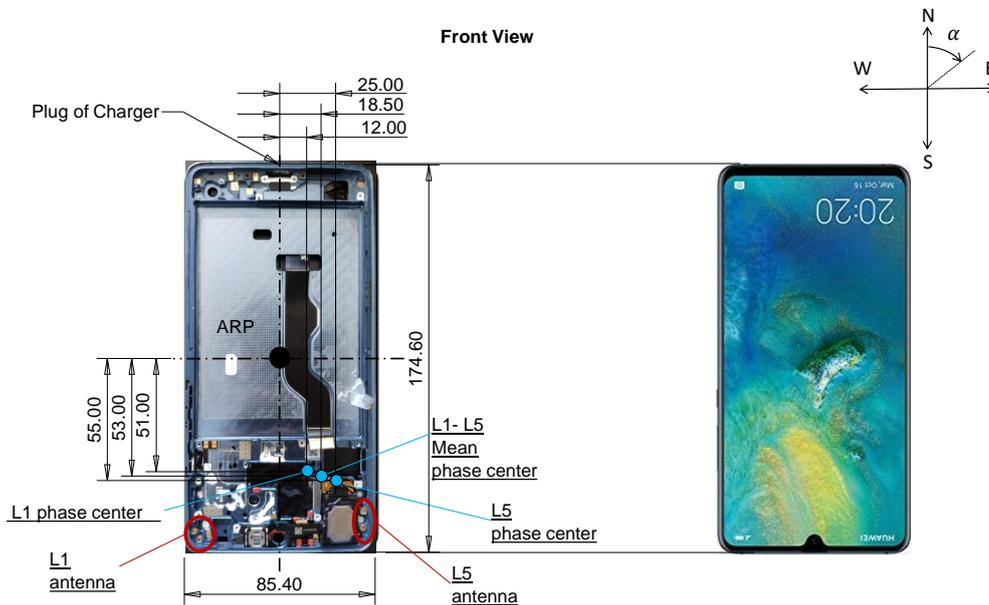
**Figure 4.12:** C/N0 color-map of the L5 reception on the front side of the Mate20X. From top to bottom: (a) the screen of the smartphone faces the antenna as shown in Fig. 4.10, (b) the back of the smartphone faces the antenna. The results obtained in the case (b) were mirrored to be consistent with those of case (a). The mean of the two estimated phase centers is reported as *L1-L5 Mean phase center*.

A color-map has been reconstructed, showing the signal strength over the front side of the smartphone. It can be recognized that there is a difference in reception between placing the smartphone with the screen up or down. This fact is most likely due to the impact of the smartphone's electronics and its display (as indicated in Xiao et al. (2019)). However, Fig. 4.11 indicates that the highest values of C/N0 for the L1 frequency can be constrained in the S-W corner of the smartphone. Concerning L5, the difference between screen up and screen down

is more evident than for L1. In fact, when the back of the smartphone is facing the antenna, the highest L5 C/N0 values are close to the estimated phase center. It is not the case when the screen is placed in front of the antenna. Nevertheless, again, a common region with large C/N0 values can be localized in the smartphone's S-E corner. Summarizing, the experiment supports the output of the antenna calibration showing two different regions with high C/N0 for L1 and L5.

The antenna locations output the experiment are in agreement with what can be reconstructed taking apart the smartphone. Fig. 4.13 shows the locations of the L1 and L5 antennas (within red-colored ellipses). The antennas are on the phone shell beneath the Printed Circuit Board (PCB) of the smartphone. A grounded point and a feeding point have been recognized and verified.

The fact that the antenna locations do not coincide with the estimated phase centers is most likely related to the electromagnetic interactions within the smartphone elements and the PCV estimation via spherical harmonics. In fact, the adjustment uses the spherical harmonics to model the PCV trying to minimize them w.r.t. the ARP.



**Figure 4.13:** L1 and L5 antennas on the Mate20X shell. The C/N0 analysis supports the antenna locations highlighted within red-colored ellipses. The location of the antennas is on the phone shell, beneath the PCB of the smartphone. A grounded point and a feeding point can be recognized within the ellipses and have been verified.

#### 4.4 High-accuracy smartphone-based positioning

The PCC obtained from the calibration have been applied in the positioning algorithm using the same approach introduced in Section 3.4. The scope of this section is to evaluate the impact of the PCC in the positioning algorithm in open sky scenarios with different levels of multipath, namely Scenario 3 and 4, introduced in Chapter 3.

Due to the complexity of the study, the aim is to have clear and controllable scenarios to be able to separate the different effects (e.g. residual phase biases) and draw verified conclusions. Scenarios 3 and 4 have been selected to remove significant sources of multipath to assess the potential of smartphone-based positioning when applying the computed PCC. Firstly, the results obtained in Scenario 4 are analyzed. Secondly, the repeatability of the results obtained in

Scenario 4 is assessed with multiple tests in Scenario 3.

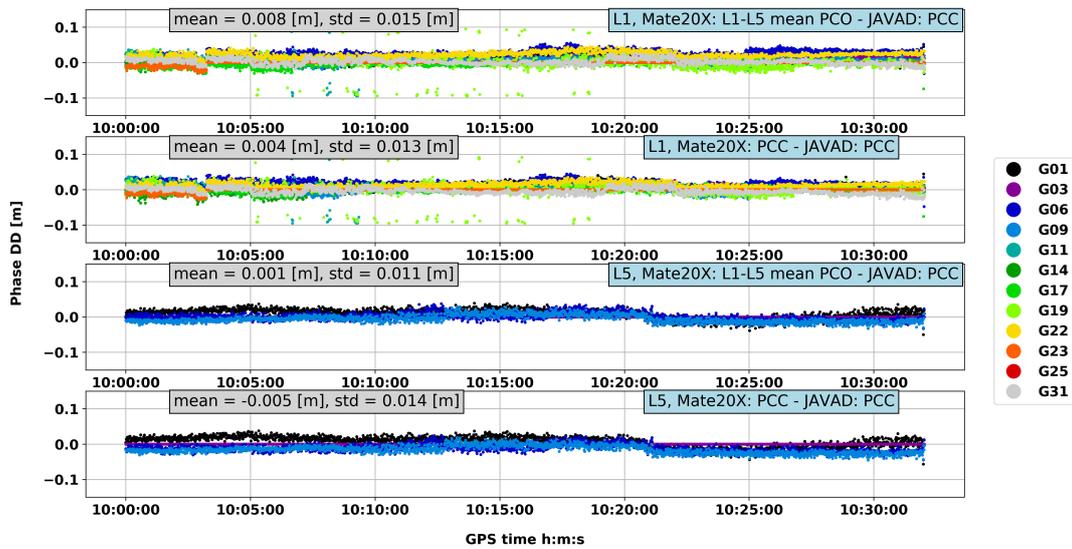
##### 4.4.1 Positioning in Scenario 4: the soccer field test

###### Phase DD analysis

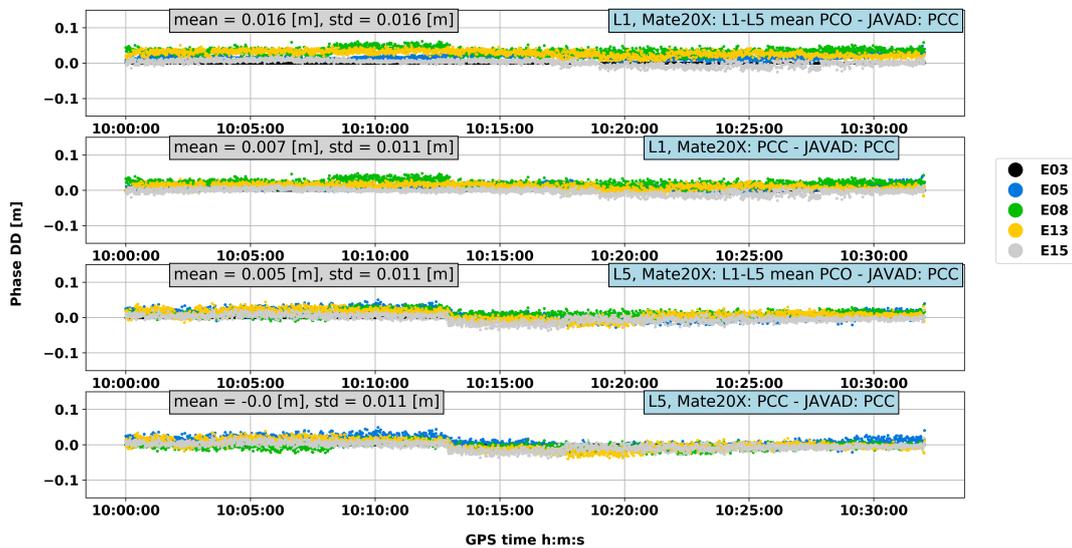
Firstly, the PCV impact on phase DD is assessed. In Chapter 3, the phase DD has been computed, showing some phase biases. Using the same approach introduced in Section 3.3, Fig. 4.14 and Fig. 4.15 show the DD between Mate20X and geodetic receiver to visualize possible remaining biases caused by station dependent errors. More precisely, Fig. 4.14 and 4.15 compare the difference in DD when making use of the complete PCC and only of the PCO of the smartphone for GPS and Galileo, respectively. As there are two distinct estimated PCs for L1 and L5 (see Fig. 4.1), a mean PCO has been considered for the smartphone, as shown in Fig. 4.1b (namely L1-L5 Mean phase center). On the one hand, the two figures suggest that there is a significant improvement for both GPS and Galileo L1/E1 measurements, improving both mean and STD values of some mm. On the other hand, while Fig. 4.15 indicates that the improvement is also there for E5a measurements, Fig. 4.14 shows an increased mean (in absolute value) and STD after applying the antenna corrections for GPS L5 measurements. The reason for this negative impact on GPS L5 measurements is related to the quality of the L5 PCV, as mentioned above. This is likely because of the lower quality of the geometry of the GPS constellation during the calibration process. Nevertheless, a positive impact of the L5 PCV is observed for the Galileo measurements showing an improvement of 5 mm in the mean. Independent of the application of the PCV, some half-cycle slips can be seen in the L1 DD depicted in Fig. 4.14. Comparing Fig. 4.14 with Fig. 3.17, it might be observed that the cycle-slips involve satellites with C/N0 below 30 dBHz. As already demonstrated by other authors (e.g. Liu et al., 2019), such measurements can be of bad quality. Fig. 4.16 indicates that after applying a CN0 mask of 30 dBHz in the DD, the half-cycle slips are removed. Moreover, the STD of the DD measurements improves by 2 mm, despite an increased mean value of 1 mm. However, it is essential to mention that observing Fig. 4.14, Fig. 3.17 and Fig. 3.18, the introduction of half-cycle slips for measurements below 30 dBHz is not a rule but can occur.

To better visualize the impact of the PCV on the phase DD, Fig. 4.17 and Fig. 4.18 directly compare the results with smartphone's PCC and PCO-only depicted in Fig. 4.14 and Fig. 4.15. After applying a CN0 mask of 30 dBHz to remove the half-cycle slips, a moving average with a 10 s window has been applied. The data has been smoothed since the noise has already been investigated in Section 3.3. Differently from Fig. 4.14 and Fig. 4.15, the satellite with the highest C/N0 has been selected as the reference. The C/N0 criterium has been applied to remove jumps due to the change of maximum elevation satellite. The reference satellites considered are G03 for GPS and E03 for Galileo (see Fig. 3.17 and Fig. 3.18 for the C/N0 visualization).

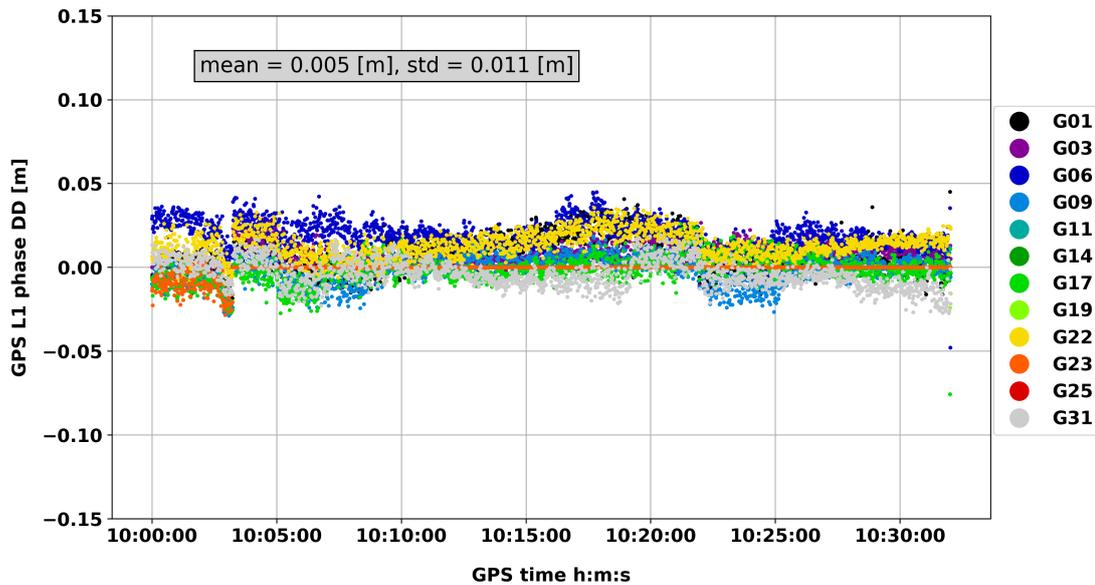
Fig. 4.17 and Fig. 4.18 compare the difference in DD when making use of the complete PCC (dot-shaped marker), and only of the PCO of the smartphone (cross-shaped marker) for GPS and Galileo, respectively. The two figures suggest that there is a significant improvement for both GPS and Galileo L1/E1 measurements, reducing some biases larger than 3 cm. For example, the mean value of the E13 DD decreases from 3.1 cm to 0.9 cm (see Fig. 4.18). Concerning GPS, a significant improvement is reported for G22 (see Fig. 4.17). As well as Fig. 4.14, Fig. 4.17 indicates the negative impact of the PCV on GPS L5 measurements, e.g. considering the G09 satellite. This result could be related to the precision of the antenna calibration for L5 signals for specific azimuthal intervals. In fact, the azimuth of the satellite G09 is around  $300^\circ$ . Hence, G09's azimuth lay in the azimuthal interval, where the L5 antenna calibration is less repeatable, as shown in Fig. 4.7. However, a positive impact of the L5 PCV is observed for the Galileo measurements showing improvements up to 1.7 cm in the mean (see E08 in Fig. 4.18).



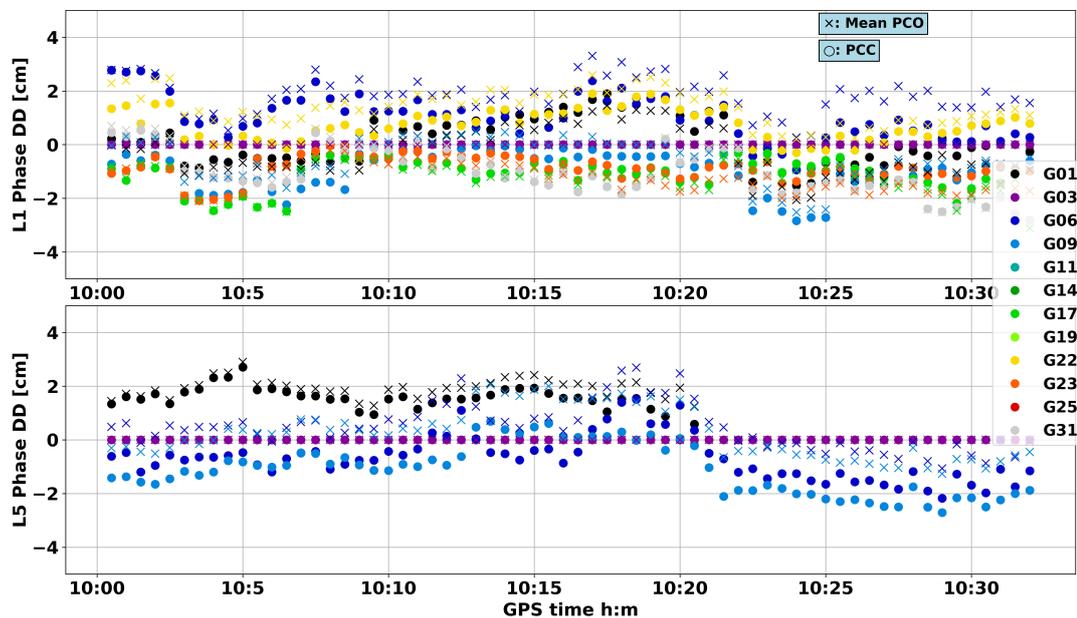
**Figure 4.14:** Phase DD of GPS L1 and L5 measurements between the Mate20X and the geodetic receiver roughly 50 m distant in the soccer field. The first and third rows show the DD applying a mean phase center offset between L1 and L5, while the second and fourth rows show the DD when applying the individual PCOs and PCV for the Mate20X. The complete AC have been applied to the geodetic receiver's observations. Here the nearest integer number of cycles has been subtracted for each satellite.



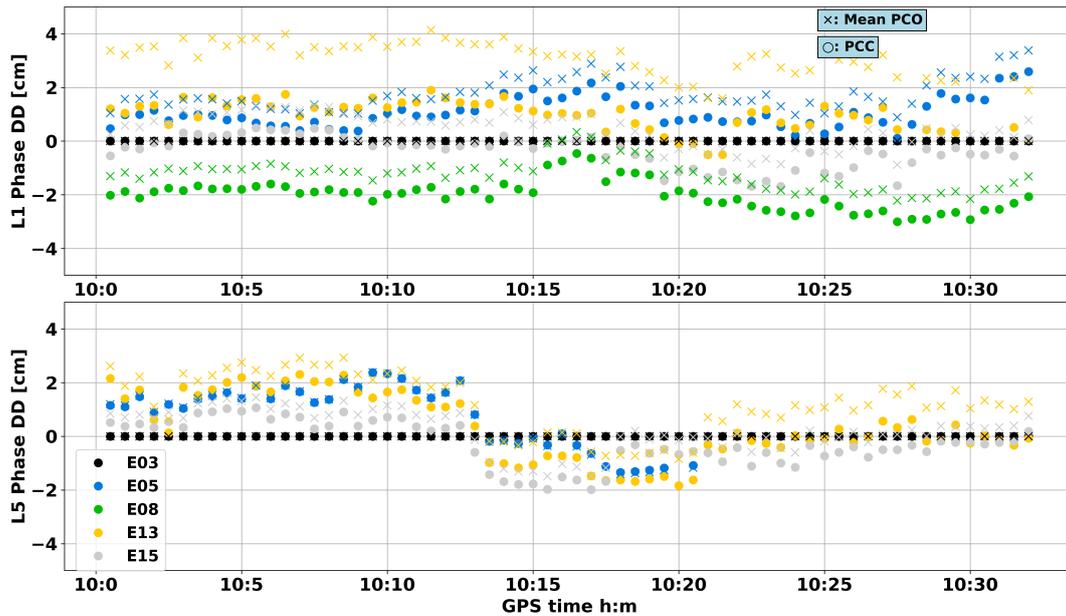
**Figure 4.15:** Phase DD of Galileo E1 and E5a measurements between the Mate20X and the geodetic receiver roughly 50 m distant in the soccer field. The first and third rows show the DD applying a mean phase center offset between L1 and L5, while the second and fourth rows show the DD when applying the individual PCOs and PCV for the Mate20X. The complete AC have been applied to the geodetic receiver's observations. Here the nearest integer number of cycles has been subtracted for each satellite.



**Figure 4.16:** Phase DD of GPS L1 measurements between the Mate20X and the geodetic receiver in the soccer field. A C/N0 mask of 30 dBHz has been applied to remove the half-cycle slips.



**Figure 4.17:** From top to bottom: phase DD of GPS L1 and L5 measurements between the Mate20X and the geodetic receiver roughly 50 m distant in Scenario 4. The DD applying a mean PCO between L1 and L5 is reported by the cross-shaped markers, while the dot-shaped markers indicate DD when applying the PCC for the Mate20X. The complete antenna corrections have been applied to the geodetic receiver's observations. Here the nearest integer number of cycles has been subtracted for each satellite.



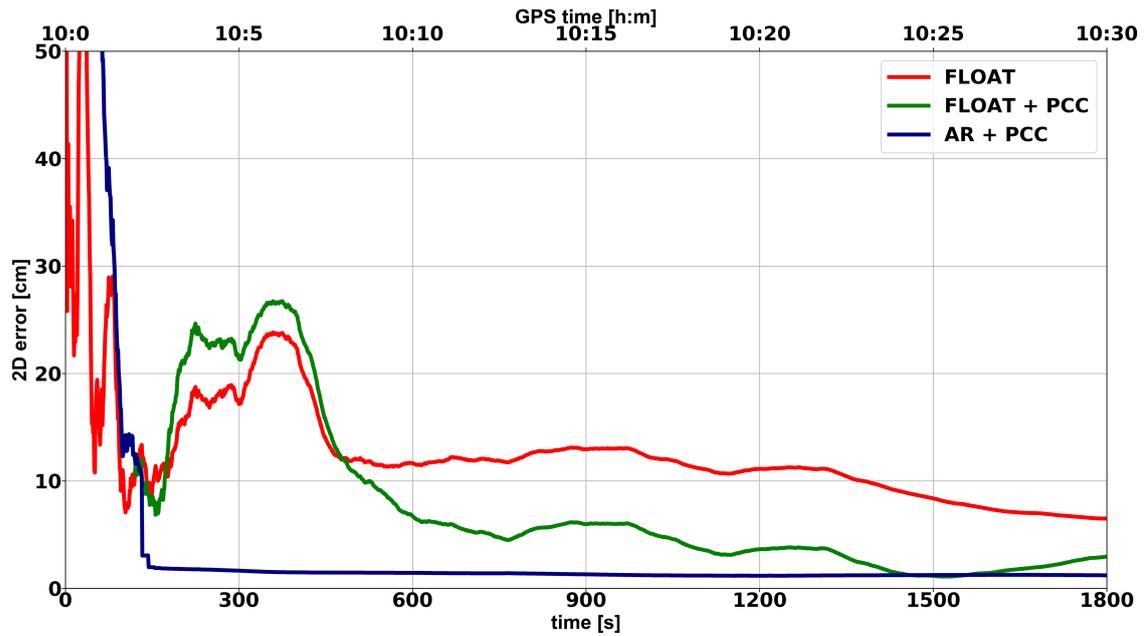
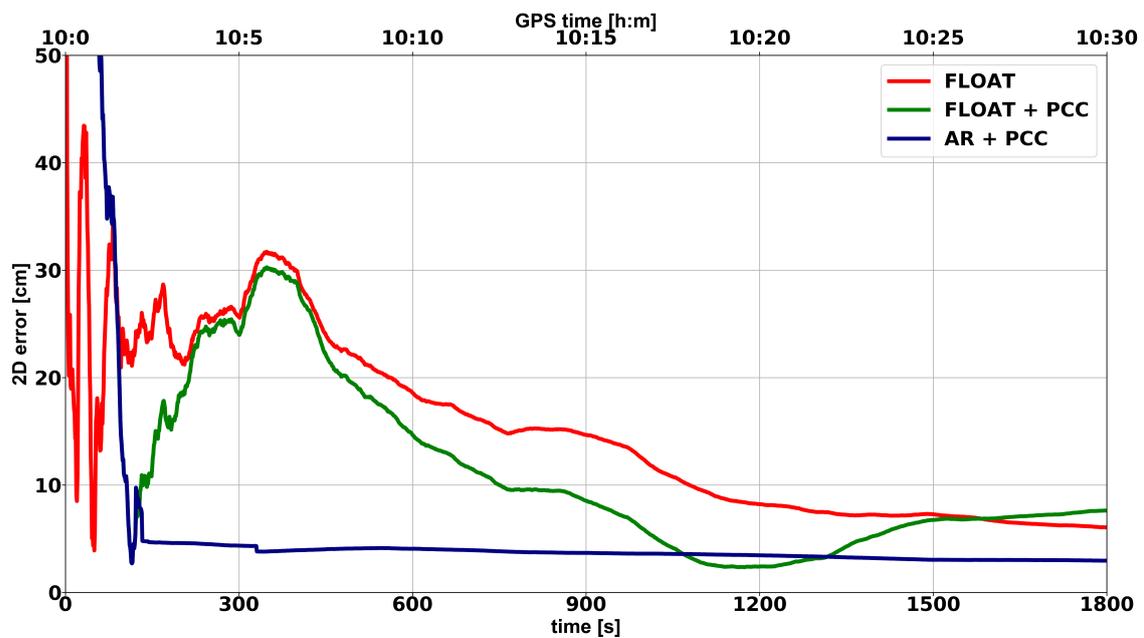
**Figure 4.18:** From top to bottom: phase DD of Galileo E1 and E5a measurements between the Mate20X and the geodetic receiver roughly 50 m distant in Scenario 4. The DD applying a mean PCO between L1 and L5 is reported by the cross-shaped markers, while the dot-shaped markers indicate DD when applying the PCC for the Mate20X. The complete antenna corrections have been applied to the geodetic receiver's observations. Here the nearest integer number of cycles has been subtracted for each satellite.

Overall, the correction for the PCV in addition to the PCO removes residual effects in the DD, making the linear combination easier to be fixed to integer values. Therefore, ambiguity resolution is expected to be possible in such a scenario after applying antenna corrections and applying a C/N<sub>0</sub> mask if required. The results in the position domain are presented in the following part of this section.

### Positioning results

Fig. 4.19 shows the improvement that antenna calibration provides in terms of 2D error. Float and ambiguity resolved results are shown in the two figures, with and without antenna corrections. Fig. 4.19a depicts the 2D positioning error obtained w.r.t. the geodetic receiver on the pitch (baseline of roughly 50 m), while Fig. 4.19b shows the results w.r.t. the reference station 12 km away. Both figures suggest that the ambiguities were fixed correctly to integers when applying the PCC in both the experiments, yielding a 1.5 cm and 3.9 cm 2D error, respectively. In both cases, the TFFA is lower than three minutes. Moreover, the STD of the fixed solution is 2 mm in the first case and 5 mm in the second case. The second solution's larger values have to be expected since the same atmospheric conditions are assumed at the two stations. Therefore, it is most probable that the difference is mainly due to the ionosphere.

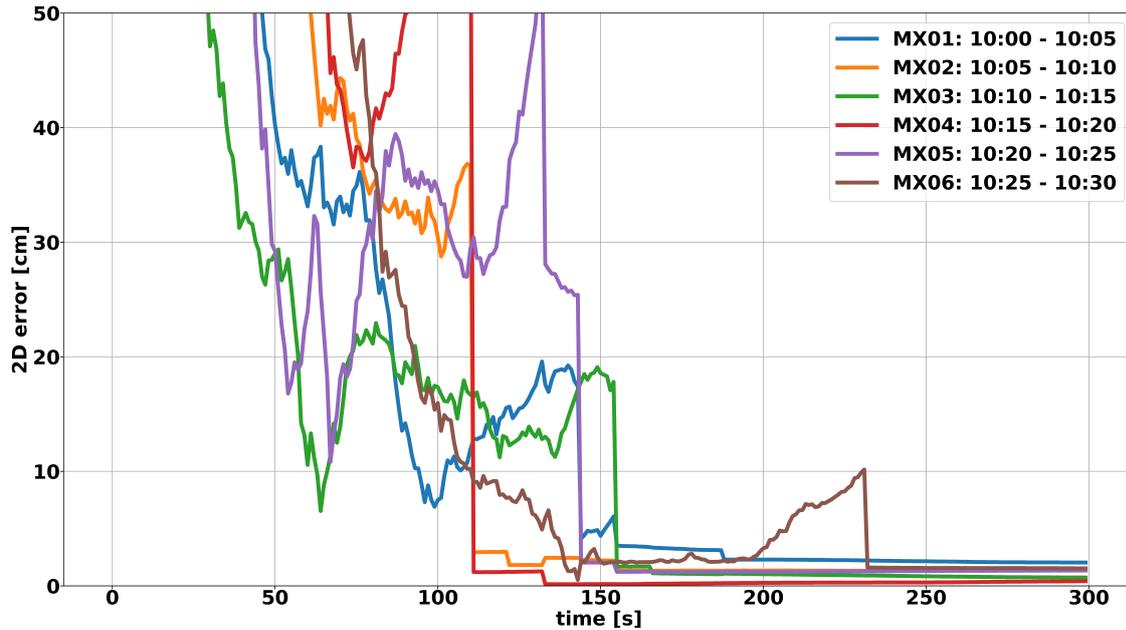
It is interesting to notice the fluctuating impact of the antenna corrections on the float solution. Although the correction for the PCV decreases the 2D error most of the time, it seems that the two float solutions (with and without antenna corrections) are converging to the same value. Overall, the experiment's primary outcome is that the use of PCC opens the possibility to fix ambiguities correctly. In this analysis, the focus is on the 2D results, since many smartphone-based applications are mainly related to 2D positioning. However, for a complete analysis of the positioning results, the RMSE in the height component is 3.5 cm and 6.1 cm, for the 50 m and 12 km baselines, respectively.

(a) *Mate20X 50 m from reference station.*(b) *Mate20X 12 km from reference station.*

**Figure 4.19:** Positioning performance in the soccer field: the Mate20X is the rover exploiting the observations of reference stations 50 m and 12 km distant. The 2D error obtained applying or not PCC is compared. Float and with ambiguity resolution (AR + PC) results are reported.

In addition, six resets every five minutes have been performed to better assess the repeatability of the results obtained using the reference station ~50 m distant. The six data-sets of five minutes are named as MX01, ..., MX06. In this way, even if the overall period is short, the geometry and the time of influence of low-quality measurements (i.e. low C/N0) change, and the ambiguity resolution might become more challenging. C/N0 masks of 30 and 35 dBHz for the MX06 period have been applied to filter out bad-quality measurements. Fig. 4.20 shows the 2D error of the six data-sets of five minutes. The step jumps observed in Fig. 4.20 occur when the TTFA is reached. For the data-sets covering the time between 10:00 and 10:25,

it can be noticed that a fixed solution can be obtained within 160 s. For the data-set named MX06, instead, more than 230 s are needed to achieve a reliable fixed solution. This effect is related to the lower number of available satellites after the application of the 35 dBHz mask. Fig. 3.17 and 3.18, introduced in Section 3.3, indicate that the employment of the mask reduce the number of observations significantly. Nevertheless, it is necessary to improve the quality of the measurements in that period (from 10:25 to 10:30, GPS time). This fact is generally true for all the small data-sets. For example, between 10:05 and 10:15, the C/N0 mask removes some very noisy measurements, e.g. G19, which has an elevation larger than  $30^\circ$  and would not be excluded by the elevation mask. In the following subsection, the repeatability of such performance in a more challenging multipath environment is further investigated.



**Figure 4.20:** Positioning performance in the soccer field: the Mate20X is the rover, while the reference station is located  $\sim 50$  m away. The 2D error obtained applying the PCC for short reset periods of 5 minutes is reported.

#### 4.4.2 Positioning in Scenario 3: the rooftop test

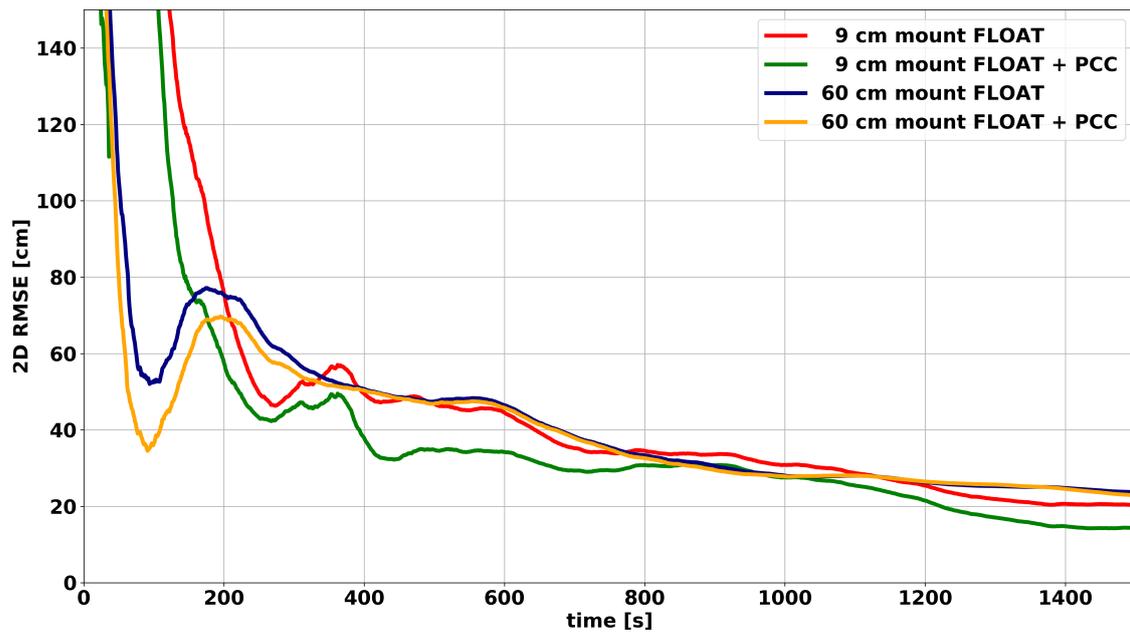
As already mentioned in Chapter 3, Scenario 3 is an open sky scenario, where several pillars with known coordinates present favorable locations for GNSS-based positioning testing. The observations of a close ( $\leq 10$  m) reference station have been exploited using the same positioning approach employed in the soccer field test. Here, the repeatability of the results obtained in the soccer field was investigated.

##### Ground-reflections influence

First, the PCC impact is analyzed w.r.t. the influence of the ground-reflections that can affect the measurements on the rooftop. As in Chapter 3, the three setups depicted in Fig. 3.24 have been studied: the smartphone lying over the top of the pillar Fig. 3.24a, the smartphone on a support 9 cm above the top of the pillar Fig. 3.24b and the smartphone mounted on a 60 cm pole attached to the top of the pillar Fig. 3.24c.

Fig. 4.21 shows the 2D RMSE computed over ten samples of consecutive one-hour data for the configurations with the smartphone mounted on a pole standing on the pillar, as shown in Fig. 3.24b and Fig. 3.24c. It can be seen that the correction for the PCV (and PCO) reduces the 2D RMSE in both the 9 cm and 60 cm mount case, but no successful AR can be achieved.

Fig. 4.21 indicates an improvement (i.e. the 2D RMSE is smaller) of roughly 6 cm when the smartphone is on the 9 cm mount, while approximately 1 cm when the smartphone is on the 60 cm mount. However, in the case of the 60 cm mount, the two solutions (with and without PCC) seem to converge to the same result. The diverse impact in the two data-sets is probably due to the multiple different reflections that the smartphone experiences because of the length of the pole. The larger multipath values, obtained with the 60 cm mount, are shown in Fig. 3.25. The outcome of this experiment is that the pillar removes a significant part of the multipath caused by the multiple surface reflections (e.g., ground and walls), which can otherwise significantly affect AR. Therefore, what can be achieved in the setup shown in Fig. 3.24a, i.e. the smartphone simply lying over the pillar, is further investigated.



**Figure 4.21:** Positioning 2D RMSE computed over ten samples of hourly data collected using the Mate20X on mounts of different heights (see Fig. 3.24b and 3.24c). The reference station is located at a distance lower than 10 m.

### The data-sets

As shown in Fig. 4.22, three pillars were chosen as support for the smartphone (the pillars are within red-colored squares). The three pillars are differently affected by the small building-block on the top left corner that can be recognized in Fig. 4.22. During the last three months of the year 2019, 35 samples of observations have been collected over the three pillars during five different DOY: 234, 235, 338, 340, 344. In 54% of the cases, a solution with successfully fixed ambiguities is achieved. It is worth mentioning that the samples with successful ambiguity resolution are related to periods with a good geometry and C/N0 levels.

### Geometric considerations

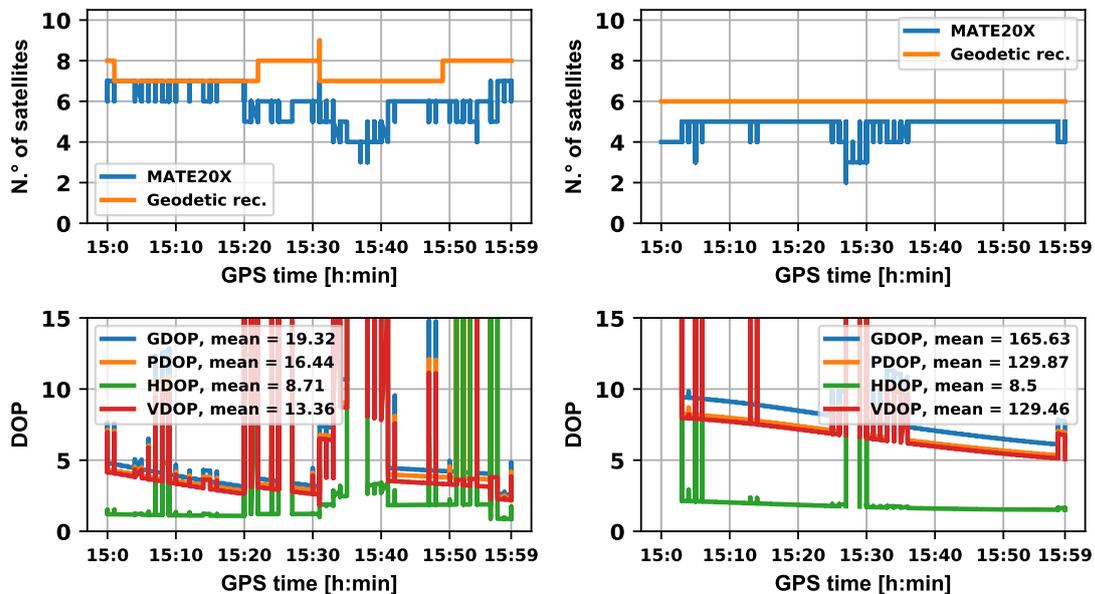
Fig. 4.23 and 4.24 show two different observation periods for the phase measurements of GPS single-frequency (L1 or L5 observation is available) and Galileo dual-frequency (both E1 and E5a observations are available). In the first case (Fig. 4.23), correct AR was not possible, while the second case was demonstrated suitable for AR. In fact, Fig. 4.23 shows poor PDOP for GPS single-frequency and high values of PDOP for Galileo dual-frequency.

Fig. 4.24 indicates that a constellation-geometry better than the one shown in Fig. 4.23 is needed for high-precision positioning purposes. This factor is a limitation of smartphone-

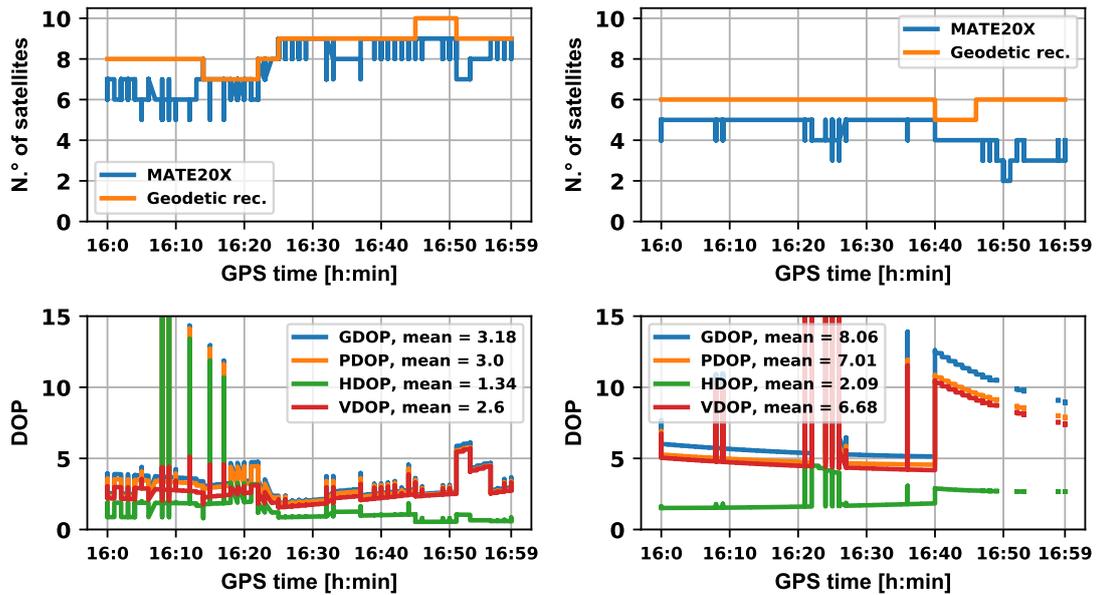
based positioning. For the sake of completeness, Geometric (G), Horizontal (H), and Vertical (V) DOP are reported.



**Figure 4.22:** Rooftop of the Geo++ building. The pillars within the red squares were chosen as locations for the tests carried out.



**Figure 4.23:** Top: number of synchronized satellites of Mate20X (blue colored line) and of the geodetic receiver (orange colored line). Bottom: GDOP, PDOP, HDOP, VDOP. GPS single-frequency constellation (left) and Galileo dual-frequency (right) during hour 15 (GPS time) day of the year (DOY) 338 of year 2019. This configuration was demonstrated non-suitable for reliable ambiguity resolution.



**Figure 4.24:** Top: number of synchronized satellites of Mate20X (blue colored line) and of the geodetic receiver (orange colored line). Bottom: GDOP, PDOP, HDOP, VDOP. GPS single-frequency constellation (left) and Galileo dual-frequency (right) during hour 16 (GPS time) day of the year (DOY) 338 of year 2019. This configuration was demonstrated suitable for reliable ambiguity resolution.

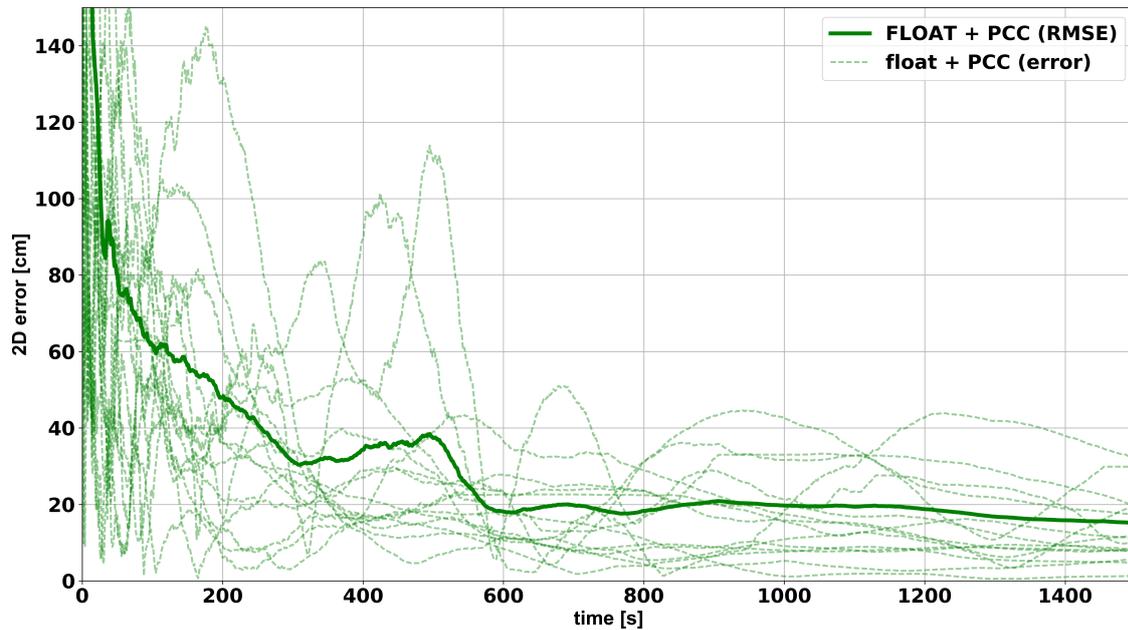
Furthermore, Fig. 4.23 and Fig. 4.24 show that the constellation is often changing in both hourly data-sets much more often than for the geodetic receiver, adding complexity to the AR process. Even if the device can track many satellites, not for all of them the phase-observations are valid, as introduced in Chapter 3. In fact, Fig. 4.23 and Fig. 4.24 highlight the difference between available satellite observations of the smartphone and geodetic receiver. Low  $C/N_0$  and high multipath values reduce the cases where ambiguity resolution is feasible. It is worth mentioning that a small PDOP does not assure a successful ambiguity resolution (Wang et al., 2020a). However, it is shown that simple considerations about geometry and signal strength provide the user with a fast apriori indication about the possibility of obtaining a precise solution with smartphone's measurements based only on the geometry.

### Positioning results

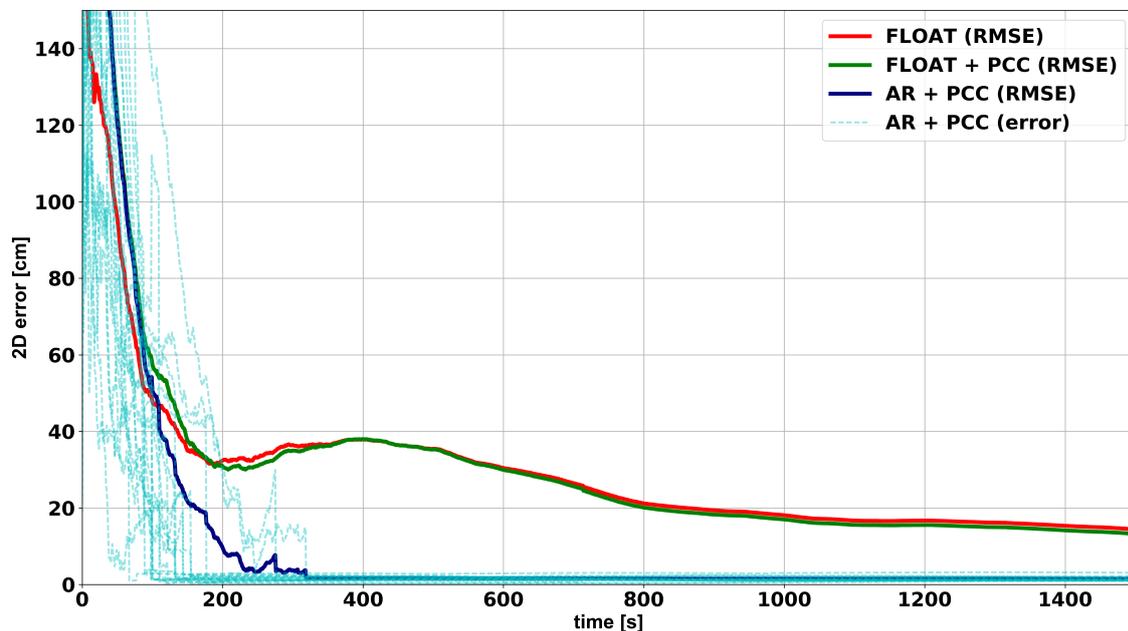
While Fig. 4.25 shows the 2D error and RMSE of the float solution during the periods where ambiguity resolution is not feasible, Fig. 4.26 exhibits the significant impact of the PCC. Moreover, while only float solutions can be performed without corrections, cm-level positioning can be achieved by applying them. Fig. 4.26 indicates that, like in the pitch case, the antenna corrections improve the 2D RMSE of the float solution by roughly 1 cm. A 2D RMSE of 1.6 cm can be achieved when the ambiguities are successfully fixed to integers applying the corrections (see Fig. 4.26 and Table 4.2). Furthermore, an RMSE of 3.8 cm is obtained for the height component. The TTFA is less than 180 s (3 min) in the 84% of the cases, while all the 19 samples are fixed in less than 350 s (roughly 6 min), as shown in Fig. 4.26 looking at the light blue colored lined and summarized in Fig. 4.27. Fig. 4.27 shows the 19 cases with successful ambiguity resolution in relation to the TTFA.

Table 4.2 indicates that in the soccer-field sub-meter 2D error was achieved within a few seconds. This result is probably due to the higher multipath in the rooftop environment. Fig. 4.25 shows the results in the 16 cases where ambiguity resolution was not possible in terms of 2D error (shaded lines) and 2D RMSE (continuous line). Although a successful ambiguity

resolution was not feasible, a 2D RMSE of 20 cm is achieved in less than 10 min, marking a significant improvement w.r.t. the performance over the mounts (see Fig. 4.21).



**Figure 4.25:** Positioning 2D error computed for 16 samples of data collected using the Mate20X lying on a pillar of the Geo++ rooftop with unsuccessful AR. The shaded dashed lines show the 2D error of all the 16 samples, while the continuous green-colored line indicates the RMSE.

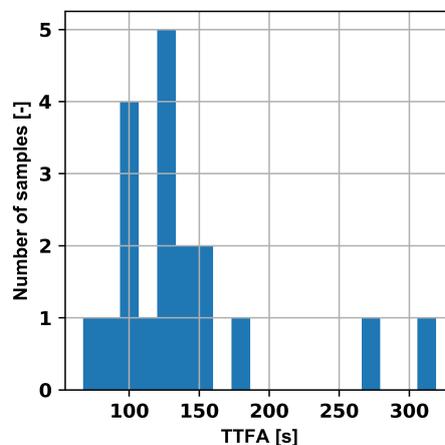


**Figure 4.26:** Positioning 2D error computed for 19 samples of data collected using the Mate20X lying on a pillar of the Geo++ rooftop. The light blue-colored lines show the 2D error of all the 19 samples with successful AR applying the PCC, while the blue-colored line depicts the RMSE. The red-colored line indicates the RMSE of the float solutions, while the green-colored line the RMSE of the float solutions applying the PCC.

Summarizing, ambiguity resolution with smartphone observations is still challenging because of the constellation geometry of the available phase measurements and the significant impact of the multipath due to the type of antenna used. Nevertheless, it has been demonstrated that, when ground-reflections are partially removed, PCC make ambiguity resolution feasible, and cm-level 2D RMSE can be achieved.

**Table 4.2:** Time To achieve Sub-Meter solution (TTSM), TTFA applying PCC and 2D RMSE using PCC for the setups analyzed in this work.

Setup	TTSM	TTFA with PCC (average)	2D RMSE with PCC
Soccer field ref. station at 50m	5 s	151 s	1.5 cm
Soccer field ref. station at 12 km	5 s	166 s	3.9 cm
Rooftop ref. station at 10m	60 s	142 s	1.6 cm



**Figure 4.27:** Histogram of the TTFA for the 19 samples with successful ambiguity resolution.

## 4.5 Discussion

The Geo++ absolute robot-based field calibration of GNSS antennas has been used to estimate PCV for the Mate20X smartphone. The results have been presented and analyzed. The PCV exhibit values up to roughly 4 cm with a formal STD that does not exceed 1.6 mm. The correction for PCV has been demonstrated as being essential for ambiguity resolution. The positioning performance was analyzed using a positioning algorithm employing an uncombined observation model, making use of the observations of a close geodetic receiver. Baselines varying from 10 m to 12 km were tested. It has been shown that, when the constellation geometry and the multipath is partly removed, a few cm-level smartphone-based positioning is feasible after correcting for PCV. The repeatability of the results was tested over 35 data-sets of one hour. In 54% of the cases, an ambiguity fixed solution was achieved successfully. The investigation showed that success is highly related to the quality of the geometry of the satellite constellation and the multipath environment. In the remaining 46% of the cases, a 2D RMSE of 20 cm is achieved in less than 10 minutes.

It can be concluded that the calculated antenna corrections are applicable for phone devices, being an asset for smartphone-based positioning with ambiguity resolution. These results open a new frontier for scientific research in high-accuracy and precision positioning using smartphones. Future studies might take advantage of several sensors that are already

inside the cellphones. A sensor fusion technique might be used. It could take into account the antenna corrections during moving operations taking care of the smartphone's attitude. Some applications in which the attitude is well defined may use low-cost smartphone-based positioning techniques. The requirements concerning the precision of the attitude are correlated with the antenna itself. In particular, it depends on the PCV variation w.r.t. the azimuth angle. Non-homogeneous antenna patterns with sudden peaks would require more precise knowledge of the attitude than antennas with homogeneous patterns. For the analyzed case, the minimum requirement for the attitude precision would be  $5^\circ$ , i.e. as large as the azimuthal resolution of the calibration. Therefore, future works could consider different orientations and analysis to lead to applications having the smartphone in hand. In addition to the discussed issues, the latter will need to deal with the interaction with the human body.

Possible fields of application are agriculture and viticulture. Machines working in a field or in a vineyard that follow pre-defined attitude movements may use smartphone-based positioning with antenna corrections. A different application is to utilize the smartphone in the vineyard as a reference to provide a GNSS-augmentation for drone-based positioning. Smartphone-based positioning algorithms could provide atmospheric corrections to the drone to improve its positioning. This concept has been proposed by the A-MICAS team composed by TREASURE fellows at the FARMING BY SATELLITE PRIZE 2018 (European Global Navigation Satellite Systems Agency (GSA), European Environment Agency and CLAAS, 2018). The idea was to improve the quality of the images obtained by the drone by improving its positioning. By enhancing the resolution of the images, the monitoring of the leaves of the vines is improved. A better knowledge of the leaves' status enhances the monitoring of the vines, having a significant economic impact. Drone's positioning augmentation is just an example, and many other applications can be realized, fulfilling the requirement of precise knowledge of the attitude and open sky.

It has been shown that the ground of a soccer field and the surface of the pillar removed entirely or partially the impact of multipath. In these scenarios, correcting for the PCV, it is possible to solve ambiguities successfully. Therefore, simple expedients, e.g., large metal plates, can be used to reduce multipath. Again, agriculture is a suitable area for applications, where wide-open sky scenarios are involved. For example, the inclusion of a metal plate in machines used in the field could mitigate the multipath due to the ground reflections. Summarizing, ambiguity resolution with smartphone measurements is possible, and the potential to substitute geodetic-grade rover receivers with smartphones for specific applications has been shown. Finally, the analysis carried out in the context of this thesis demonstrates high accuracy smartphone-based positioning, unveiling new GNSS-based applications using Android devices.

---

## 5. Mitigating the Impact of SSR Atmospheric Parameter Errors

---

### 5.1 Motivation

Atmospheric delays are distance-dependent errors with a strong impact on satellite-based positioning. The use of SSM, and consequently, SSR concepts introduced in the previous chapters, is considered state-of-the-art in relative and absolute positioning. High accuracy and precision techniques like N-RTK and PPP-RTK can make use of SSR corrections generated by a network of reference stations and transmitted in a defined format (e.g. SSRZ described in Chapter 2). The atmospheric SSR parameters need to be interpolated for the user location to be employed in the user positioning algorithm. In this chapter, the focus is on two main topics:

- the mitigation of the interpolation error during severe weather and TID events;
- the effect of the propagation of an SSR modeling error on the positioning performance.

Both topics are investigated with the main purpose of assessing the impact on the user's performance. Many authors have highlighted the strong correlation between atmospheric corrections and fast and reliable AR. The results presented in Wübbena (2007) showed the importance of having a precise estimation of the ionospheric contribution (with an STD lower than approximately 5 cm). Similar results have been obtained by Hernández-Pajares et al. (2000); Paziewski (2016) and Psychas et al. (2018). Here, the aim is to obtain a more robust AR process by mitigating the interpolation error during periods with a perturbed atmosphere. Furthermore, the SSR corrections contain modeling errors that are propagated through the interpolation to the user. The effect of such errors on the estimated user's position is addressed in this chapter.

In addition, the analysis wants to evaluate the benefit of using external information (e.g. external models) for the interpolation of SSR corrections. Multiple researchers reported on the use of atmospheric models. Hernández-Pajares et al. (2000) investigated the use of ionospheric tomography products for real-time positioning. Paziewski (2016) analyzed the impact of ionospheric corrections for N-RTK applications, while Psychas et al. (2018) considered the case of a PPP-RTK user. Li et al. (2011); Geng et al. (2011) investigated the use of regional atmospheric parameters showing their significance for fast AR.

In all of the above cases, the atmospheric corrections have to be estimated for the approximate user position. This is commonly done by interpolation from values estimated either for the reference station locations or for a grid of points within the network. Poor performance of the interpolation can introduce an additional error, and the choice of the interpolation method is, therefore, a crucial step in successfully applying PPP-RTK techniques. Recently, Wang et al. (2020b) presented a comprehensive study of several interpolation techniques for PPP-RTK applications. In their work, the authors suggested different interpolation techniques based on the geometry of the reference station network. Moreover, they showed that, in general, a selected low-order surface could reasonably fit the distance-dependent biases using a Least Squares (LS) approach. This is plausible since distance-dependent errors grow in first-order linearly with the baseline length (Beutler et al., 1988; Brunner, 1994; Schön, 2007).

An important consideration is that the output error of the interpolation process can affect AR. The interpolation error can be evaluated as a bias in a linear combination of observations

to solve ambiguities, as seen e.g. in the work of Li et al. (2014). In their work, the authors investigated the impact of the use of biased and unbiased DD models for AR, depending on the dimension of the network for both tropospheric and ionospheric parameters. Furthermore, Jongrujan and Satirapod (2020) investigate the use of a stochastic model based on Residual Interpolation Uncertainty (RIU) as the weighting schemes for N-RTK. As shown by Chen et al. (2003), the RIU can be an indicator of the network corrections quality. The analysis carried out by Jongrujan and Satirapod (2020) indicates that compared to a standard stochastic model (e.g. elevation-based), the RIU-weight model increases both the AR success rate and the positioning accuracy. Overall, many authors have highlighted the impact of network-correction interpolation error on AR for high-accuracy positioning.

The interpolation errors for an atmosphere with small gradients tend to be small. Often, however, physical processes in the atmosphere introduce additional gradients, which may be difficult to account for if the reference network is not dense enough. For example, the ionospheric group delay and phase advance, appearing as the signal passes through the dispersive ionospheric plasma, can be affected by TIDs. TIDs can be described as propagating fluctuations in the ionospheric electron density, and are thought to be the ionospheric manifestation of atmospheric gravity waves, as first theorized by Hines (1960). Unlike other ionospheric phenomena, TIDs are also common in mid-latitude regions, making them of particular interest for many GNSS users across the world. These perturbations can be observed in GNSS measurements as wave-like perturbations in TEC. Several authors have demonstrated the impact of TIDs on GNSS-based positioning (e.g. Chen et al., 2003; Wanninger, 2004; Sieradzki and Paziewski, 2016) and several research groups are working on the detection of such disturbances (e.g. Hernández-Pajares et al., 2017; Reinisch et al., 2018; Bolmgren et al., 2020).

In addition to the ionosphere, the troposphere accounts for the second component of the atmospheric delay (see Eq. 2.4, 2.5). As introduced in Section 2.1.3, the tropospheric delay is a function of meteorological parameters like pressure, temperature, and humidity. Accordingly, significant temporal and spatial variations in these parameters result in large variations of the tropospheric delays, and consequently, a more challenging interpolation. These variations frequently occur within the context of frontal developments, which are often linked to large amounts of water vapor being present in the atmosphere (Gregorius and Blewitt, 1999; Baelen et al., 2011). In addition, so-termed severe weather events such as local convective clouds, intense precipitation, and thunderstorms (Guerova et al., 2013) can lead to increased fluctuations in the tropospheric delay.

With the rapid development of high-density Numerical Weather Models (NWMs) in recent years (Bauer et al., 2015), there have also been several attempts to utilize these models to generate better tropospheric products for positioning (e.g. Douša et al., 2016, 2018; Zus et al., 2019b). Douša et al. (2016) showed that tropospheric gradients computed from NWM and GNSS measurements could indicate the variation of a weather event with good accuracy. In their work, the authors show that the horizontal gradients point in the direction of the highest values of ZWD. In another study, Zus et al. (2019a) showed that the use of horizontal tropospheric gradients could improve the interpolation of ZWD in an IDW method. In their work, the authors demonstrated a 10% and 5% improvement for post-processing and real-time applications, respectively. These results can be considered as further evidence of the valuable tropospheric information contained in the horizontal gradients.

Hobiger et al. (2010) used a fine-mesh NWM to reduce the vertical error component of GPS positioning estimates during the passage of a typhoon in Japan. Wilgan et al. (2017) used a combination of a CORS network and NWM data to generate ZTDs for test sites in Poland. They showed that NWM data addition leads to slightly improved ZTD estimates during a test period with heavy rainfall. Many different interpolation techniques have been used to estimate tropospheric delays between reference stations. The two most prominent are OK and IDW methods. Al-Shaery et al. (2010) investigate OK with different model assumptions. Pace et al. (2015) use OK to generate tropospheric grids and Zhang et al. (2017) employ an IDW method for this

purpose. An alternative approach was proposed by Shi et al. (2014), who chose an optimal polynomial model from a set of candidates and fit the model parameters to the observations at the CORS

In this chapter, the interpolation of atmospheric SSR parameters estimated by a network of reference stations during severe weather and TID events is investigated. The network processing employs an uncombined observation model (see Section 2.4). In particular, the following questions are addressed:

- Are there any significant differences between interpolation of ionosphere and troposphere parameters?
- Directional interpolations are state of the art in mathematical sciences, but they often require sophisticated models to take into account spatial and temporal variations (e.g. Andersson et al., 2017). Is it possible to modify a commonly used interpolation technique to make it *directional* by the use of external information?
- Can NWMs and TID detection methods provide useful external information during severe weather and TID events to improve the atmospheric interpolation for a GNSS user?
- How could the external information be transmitted to the user?

The analysis is carried out looking into four universally used interpolation methods: OK, IDW, Clough - Tocher (CT) and WLS. Novel techniques to interpolate SSR atmospheric parameters using a directional WLS, taking advantage of NWM and TID data is presented.

The chapter is organized in the following way. Section 5.2 presents the expected impact of interpolation errors on positioning. Section 5.3 introduces the interpolation method employed, and in Section 5.4 the interpolation performance of the proposed techniques is evaluated with simulated artificial data. Sections 5.5 and 5.6 present the assessment of the interpolation using real data during a severe weather event. ZTD interpolation during two severe weather events is investigated: the Xavier windstorm that occurred in northern Germany at the beginning of October 2017 and a period with high weather variations that occurred in the Netherlands in June 2017. The ionospheric interpolation is evaluated using data collected during May 2019 over the Okinawa area in Japan, where Medium-Scale TIDs (MSTIDs) are commonplace (e.g. Tsugawa et al., 2007). Section 5.7 analyses the impact of an error in the SSR ionospheric modeling on the positioning results. Firstly, a simulation is carried out to show the expected performance in terms of ambiguity resolution. Secondly, the effect on SSR-based positioning is investigated. Finally, in Section 5.8, possible user applications of the new interpolation methods are proposed and the main conclusions are discussed.

## 5.2 Expected impact on positioning

The use of SSR corrections enables the direct handling of the distance-dependent errors (e.g. atmospheric errors) and the possibility to solve ambiguities in absolute positioning. As introduced above, ambiguity resolution requires the separation of the ambiguity term from the satellite and receiver biases. Therefore, typically, linear combinations that reduce or eliminate some of the biases are used for ambiguity fixing. Commonly used combinations are DD, Wide Lane (WL) and Melbourne-Wübbena (MW) combinations. Hence, in both relative or absolute positioning, a linear combination of observations is needed for AR. Here, the purpose is to visualize where the interpolation error would affect the ambiguity resolution and, therefore, the positioning results. Moreover, it is worth mentioning the well-known rule of thumb introduced in Santerre (1989, 1991) according to which an error in the ZTD is propagated three times larger in the positioning domain (in the height component specifically). Concerning the ambiguity resolution, the focus is on the DD combination, which is also used in the Least-squares AMBiguity Decorrelation Adjustment (LAMBDA) method (Teunissen, 1993).

Let us consider a signal frequency  $f_j$  with wavelength  $\lambda_j$ , two satellites  $s$  and  $q$ , and two stations  $u$  and  $v$ . Also, let  $u$  be the user and  $v$  a non-physical GNSS station data generated using SSR parameters computed by a network of reference stations (as described in Chapter 2). To visualize the direct impact of the atmospheric parameters, the relativistic and station dependent (i.e. PCV and MP) errors are assumed to be corrected perfectly. Accordingly, following subsection 2.1.4, the phase DD equation yields:

$$\phi_{uv}^{sq} = \rho_{uv}^{sq} + T_{uv}^{sq} - I_{uv}^{sq} + \lambda_j N_{uv}^{sq} + \epsilon_{uv}^{sq}. \quad (5.1)$$

In a similar way, the pseudorange DD equation can be obtained:

$$p_{uv}^{sq} = \rho_{uv}^{sq} + T_{uv}^{sq} + I_{uv}^{sq} + \epsilon_{uv}^{sq}. \quad (5.2)$$

If the network SSM of the atmosphere and interpolation process worked perfectly, the quantities  $T_{uv}^{sq}$  and  $I_{uv}^{sq}$  would be null like in a zero-baseline. However, that is not true in a real-case. The error in the SSR and the interpolation need to be taken into account. Hence, the DD atmospheric parameters can be written as follows:

$$\begin{aligned} T_{uv}^{sq} &= (m_T^q - m_T^s) \mathbf{\Gamma} \nabla_{\text{ZTD}}^{\text{SSR}}, \\ I_{uv}^{sq} &= (m_I^q - m_I^s) \mathbf{\Gamma} \nabla_{\text{VTEC}}^{\text{SSR}} \frac{40.3}{f_j^2} 10^{16}. \end{aligned} \quad (5.3)$$

Where  $\mathbf{\Gamma}$  is the interpolation matrix,  $\nabla_{\text{ZTD}}^{\text{SSR}}$  the SSR ZTD error vector, and  $\nabla_{\text{VTEC}}^{\text{SSR}}$  the SSR Vertical Total Electron Content (VTEC) error vector in TECU. The error vectors  $\nabla_{\text{ZTD}}^{\text{SSR}}$  and  $\nabla_{\text{VTEC}}^{\text{SSR}}$  contain the error for each station of the network. ZTD and VTEC are valid for a satellite elevation of 90 degrees, hence they need to be mapped accordingly to the actual satellite elevation angle. The mapping function for troposphere (ionosphere) and satellite  $q(s)$  is indicated as  $m_{T(I)}^{q(s)}$ .

Concerning ambiguity resolution, following Li et al. (2014) the linearized model of the DD equations yields:

$$E(\mathbf{l}_{uv}^{sq}) = \mathbf{A}\mathbf{a} + \mathbf{B}\mathbf{b} + \mathbf{C}\nabla^{\Gamma}, \quad D(\mathbf{l}_{uv}^{sq}) = \mathbf{Q}_{ll} \quad (5.4)$$

where  $E$  and  $D$  denote the expectation and dispersion operators. The vectors  $\mathbf{a} \in \mathbb{Z}^n$ ,  $\mathbf{b} \in \mathbb{R}^p$  and  $\nabla^{\Gamma} \in \mathbb{R}^2$  are the integer ambiguity, baseline and interpolation atmospheric error vector, respectively. Their design matrices are  $\mathbf{A} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{B} \in \mathbb{R}^{m \times p}$  and  $\mathbf{C} \in \mathbb{R}^{m \times 2}$  with  $[\mathbf{A}\mathbf{B}\mathbf{C}]$  of full column rank. The observation vector  $\mathbf{l}_{uv}^{sq} \in \mathbb{R}^m$  contains the code and phase observations and is assumed to be contaminated by normally distributed random noise with zero mean and variance-covariance matrix  $\mathbf{Q}_{ll}$ . From Eq. 5.3 follows that the interpolation atmospheric error vector is the two component vector:

$$\nabla^{\Gamma} = \left[ \mathbf{\Gamma} \nabla_{\text{ZTD}}^{\text{SSR}} \quad \mathbf{\Gamma} \nabla_{\text{VTEC}}^{\text{SSR}} \right]^T \quad (5.5)$$

The interpolation atmospheric error vector can be seen as a bias vector, and its impact on ambiguity resolution was investigated in Li et al. (2014). More specifically, Li et al. (2014) evaluate the difference in terms of ambiguity resolution between the biased and unbiased model in the presence of atmospheric biases. In this chapter, the aim is to analyze the impact of the interpolation method, i.e. of the matrix  $\mathbf{\Gamma}$ , on the bias affecting the DD. In the next section, the interpolation methods used are introduced.

## 5.3 Interpolation techniques

### 5.3.1 Interpolation of scattered data

In this chapter, the interpolation is considered in two spatial dimensions. The general problem is to interpolate a bivariate function at a point of given coordinates  $\mathbf{x}^*$ , making use of the

known function values at  $n$  scattered points with known coordinates  $\mathbf{x}_{1,\dots,n}$ . The term *scattered* is related to the fact that the points  $\mathbf{x}_{1,\dots,n}$  are not assumed to satisfy any particular conditions either in space or density (e.g. Franke, 1982). Numerous authors addressed the problem of interpolation of scattered data in two (or more) independent variables (e.g. Franke, 1982; Alfeld, 1989; Amidror, 2002).

There are many families of techniques suitable for interpolation of scattered data. One big set of interpolation includes Shepard's methods (Shepard, 1968). These methods are global, i.e. they use all the scattered points. Their basic idea is that the interpolated value should be influenced more by the nearby points than by the distant ones.

Another family used often takes advantage of the radial-basis functions. These methods are global, firstly introduced by Hardy (1971). The central concept behind them is to define a radius of influence  $r$  to compute the following function:

$$g_i(\mathbf{x}) = \sqrt{d_i^2 + r^2}, \quad (5.6)$$

where  $d_i$  is the Euclidean distance between the points  $\mathbf{x}^*$  and  $\mathbf{x}_i$ . The second step is to determine the coefficients  $a_i$  so that the interpolated value is given by:

$$\hat{y}_* = \sum_{i=1}^n a_i g_i(\mathbf{x}). \quad (5.7)$$

Although the method can be quite accurate (Franke, 1982), it exhibits a strong dependency on the radius  $r$ . Hence, the technique is not further investigated.

Finite element-based methods are also suitable for the interpolation of scattered data. They are based on the concept of using  $C^1$  finite element functions on a triangulation of the convex hull of the point set (Franke, 1982). As an example, the CT element is considered (Strang and Fix, 1973). The CT interpolation is introduced later in this section.

The family of stochastic process methods is often used in spatial analyses (e.g. in geology or geostatistics). An example is the OK method, which is presented later in this section.

Other approaches are, for example, Maude's, triangle-based blending methods, and Foley's method. The latter generalizes the Newton interpolant concept making this technique appropriate for the problem. However, the technique involves an ordering processing, and sometimes it can be not as smooth as other approaches. Hence, it is not considered in this investigation. For further details about other possible interpolation methods, the reader is referred to Franke (1982); Alfeld (1989); Amidror (2002).

### 5.3.2 Methods introduction

Three commonly used global techniques to interpolate GNSS-based atmospheric parameters are chosen: IDW, OK, and WLS. In addition, a local and smooth (it involves a  $C^1$  function) technique is investigated: CT. The first technique is one of Shepard's methods, in which the following expression gives the interpolated value:

$$\hat{y}_* = \sum_{i=1}^n w_i y_i, \quad (5.8)$$

where  $w_i$  is the weight given to the value  $y_i$  at the known point  $\mathbf{x}_i$ . In the case of the IDW method, the weight is defined as follows:

$$w_i = \frac{d_i^{-p}}{\sum_{j=1}^n d_j^{-p}}, \quad (5.9)$$

where  $d_i$  is the Euclidean distance between the points  $\mathbf{x}^*$  and  $\mathbf{x}_i$ . In this work, the selected exponent is  $p = 2$ . Furthermore,  $\sum_{i=1}^n w_i = 1$  in order to have an unbiased estimation.

While IDW is a deterministic method, OK is a stochastic method. The best linear predictor of a process  $y = f(\mathbf{x})$  at a point  $\mathbf{x}^*$  in the ordinary Kriging method is described by Eq. 5.8 (e.g. Cressie, 1993). However, in OK the weight  $w_i$  is based on the concept of semivariograms expressed by the following equation:

$$\gamma(\mathbf{d}) = \frac{1}{2} \text{var}(f(\mathbf{x} + \mathbf{d}) - f(\mathbf{x})), \quad (5.10)$$

where  $\text{var}$  is the variance operator,  $\mathbf{d}$  is the distance-vector between two points  $\mathbf{x}_i$  and  $\mathbf{x}_j$  with  $i, j \in [1, n]$ . As described by several authors (e.g. Cressie, 1993; Oliver and Webster, 2014), there are many ways to model the variogram for the implementation in the Kriging method. After some experimental tests, we decided to exploit an exponential model that depends only on the Euclidean distance between the points (e.g. Press et al., 2007). The exponential-based variogram yields:

$$\gamma(d) = \alpha d^\beta, \quad (5.11)$$

where  $\alpha$  is adjusted by unweighted least squares over all pairs of data points, i.e. over all the stations considered in the input data. The exponent  $\beta$  of the distance has been experimentally tuned within the interval  $[1, 2[$  and set to 1.5 eventually.

The WLS interpolation is based on fitting a functional model to the observation points, with weights typically depending on the distance between interpolation and observation points. Here, the functional model is a plane, and the weight goes with the inverse of the distance elevated to a power  $p$ . The same approach has been used by other authors (e.g. Chen et al., 2003). In 2D, this technique requires to estimate three parameters related to the deviations in the two directions of the desired point as well as the offset of the plane, coinciding with the estimated value at the interpolation point. The concept can be expressed as:

$$y_i = a\Delta x_1 + b\Delta x_2 + y^* + \epsilon_i, \quad (5.12)$$

where  $\Delta x_1$  and  $\Delta x_2$  are the displacements in the two directions defining the 2D plane.  $a$ ,  $b$  and  $y^* = f(\mathbf{x}^*)$  are the parameters that need to be estimated and  $\epsilon_i$  the residual error. Applying the WLS approach, the estimated value  $\hat{y}^*$  can be written as:

$$\hat{y}^* = \mathbf{e}_3(\mathbf{A}^T \mathbf{W} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W} \mathbf{y}, \quad \text{with } \mathbf{W} = d^{-p} \mathbf{I}, \quad (5.13)$$

where  $\mathbf{A}$  is the design matrix,  $\mathbf{W}$  the weight matrix,  $\mathbf{I}$  is the identity matrix and  $\mathbf{y}$  the observation vector. The observation vector is the column vector of all the values at the known coordinates. The weight matrix is a diagonal matrix with values  $d^{-p}$ . The design matrix can be written as follows:

$$\mathbf{A} = \begin{bmatrix} \Delta x_{1,1} & \Delta x_{2,1} & 1 \\ \vdots & \vdots & \vdots \\ \Delta x_{1,n} & \Delta x_{2,n} & 1 \end{bmatrix}. \quad (5.14)$$

Accordingly, a unit vector  $\mathbf{e}_3 = [0 \ 0 \ 1]$  is used to retrieve the estimated value (see Eq. 5.12 and 5.14).

As mentioned above the 2D CT method is based on triangulation, and works in two steps. First, the set of points is triangulated. Second, the interpolation scheme is employed within each triangle. As a consequence, CT is a local method resulting in a fast interpolation process. It employs an interpolation scheme based on polynomials of higher order than one to assure the  $C^1$  continuity. Such polynomials are described in a *Bezier form* as:

$$p(\mathbf{x}^*) = \sum_{i+j+k+l=3} \frac{3!}{i!j!k!l!} c_{ijkl} b_1^i b_2^j b_3^k b_4^l, \quad (5.15)$$

where  $b_{1,\dots,4}$  are barycentric coordinates of the triangle containing  $\mathbf{x}^*$ .  $c_{ijkl}$  are the *Beziér ordinates* of  $p$  and contain the known data values at the points  $\mathbf{x}_{1,\dots,n}$ . The function  $p$  defined in Eq. 5.15 is a continuous piecewise cubic function. By proceeding for all the triangles a  $C^1$ -continuous piecewise interpolation surface is defined over the triangulation of the scattered point set. It is out of the scope of this thesis to add further details of the CT interpolation. More details can be found in (Strang and Fix, 1973; Alfeld, 1984; Amidror, 2002, e.g.).

At this point the interpolation matrix  $\Gamma$  applied in Eq. 5.3 can be described for each interpolation method:

$$\begin{aligned}\Gamma_{\text{IDW/OK}} &= \begin{bmatrix} w_1 & \dots & w_n \end{bmatrix}, \\ \Gamma_{\text{WLS}} &= \mathbf{e}_3(\mathbf{A}^T \mathbf{W} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W} \\ \Gamma_{\text{CT}} &= [\mathbf{P}].\end{aligned}\tag{5.16}$$

In Eq. 5.16,  $[\mathbf{P}]$  indicates the system of equations required for the application of the CT method. It is worth mentioning that in Eq. 5.16 the weights are different for the IDW and OK methods, but the matrices  $\Gamma_{\text{IDW}}$  and  $\Gamma_{\text{OK}}$  have the same form.

### 5.3.3 Modified WLS methods

In accordance with the results of previous authors (e.g. Douša et al., 2016; Zus et al., 2019a), the assumption that the atmospheric parameter varies along one defined direction in the 2D plane is made. Regarding the ionosphere, we make the additional assumption that during the passage of a TID the VTEC is perturbed in a plane wave fashion, so that the the GNSS measurement varies in the propagation direction of the wave. Although parallel wavefronts are commonly observed, TIDs caused by singular sources, like thunderstorms and earthquakes, propagate outwardly and can therefore display circular wavefront patterns (e.g. Yang et al., 2017; Azeem and Barlage, 2018). For the tropospheric delay, the effect of the height of the station can be corrected, making the ZTD referred to a station height of 0 m. Hence, the ZTD can be analyzed in a 2D plane, e.g. East-North. During severe weather events, the change of magnitude in the delay is mainly due to the wet component which has been demonstrated to vary along the gradient direction (Douša et al., 2016). As mentioned earlier, the tropospheric horizontal gradients appear to describe the direction of the weather front well.

Following these assumptions, a modified WLS approach is proposed. This approach assigns more weight to points perpendicular to the direction of variation, i.e. in the direction of propagation for a TID, and ZTD gradient for a severe weather event. The reason behind this is that we expect to find more similar values perpendicular to the variation than in the direction of change. To implement this, the distance  $d$  is replaced by the modified distance  $\tilde{d}$  defined as follows:

$$\tilde{d} = d\sqrt{1 + \cos^2\theta},\tag{5.17}$$

where  $\theta$  is the angle between  $\mathbf{d}$  and the direction of variation of the parameter. Fig. 5.1 illustrates the concept of the proposed implementation graphically. This method replaces the circular isolines of the inverse distance squared weights with ellipses with semi-major axis perpendicular to the direction of change, as shown in Fig. 5.1.

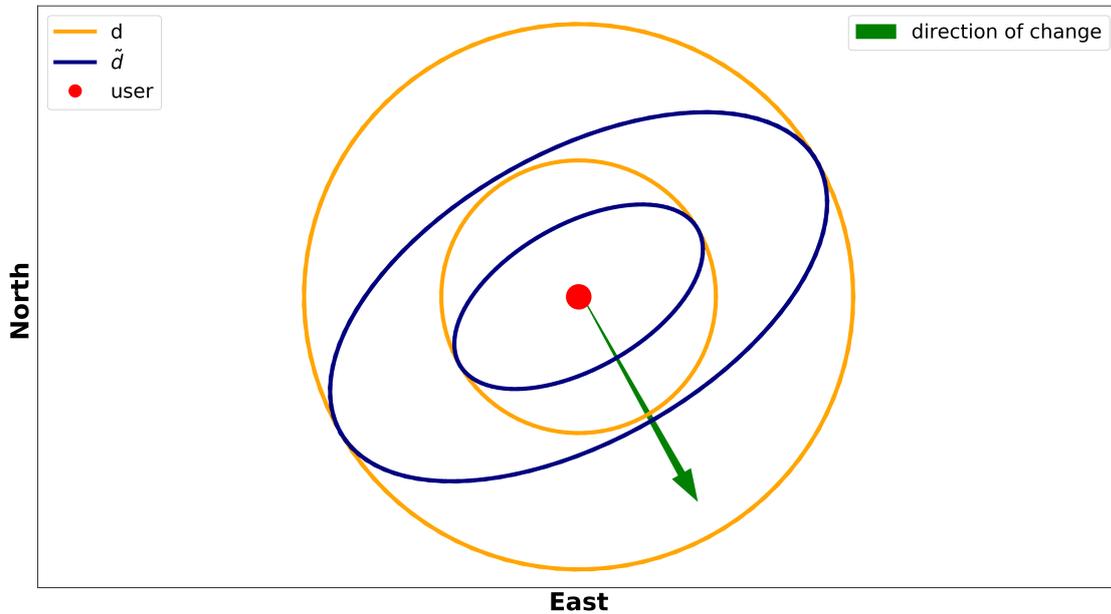
In addition to the shape of the weighting, the impact of the power  $p$  of either the distance  $d$  or  $\tilde{d}$  in the weight matrix  $\mathbf{W}$  (see Eq. 5.13) is investigated. Exponents in the interval  $[2, 8]$  have been tested, showing relevant differences between two and four. Hence, results when employing WLS with distance-weight to the power of two (WLS2) and four (WLS4) are presented. In the same way, we show the analysis using the modified distance  $\tilde{d}$  naming the techniques WLS2D and WLS4D. Concerning atmospheric (troposphere) refractivity fluctuations, a more detailed investigation about the effective separation distance among stations has been carried out in Schön and Brunner (2008). In their work, the authors derived a mathematical model based on the turbulence theory that confirms the intuitive statement that observations that

are spatially closer to each other are more strongly correlated than observations with a large separation distance.

As described in Section 5.4, the effect a TID has on the interpolation is dependent on the spatial scales of the network and the TID. If the average baseline  $b$ , the distance between reference stations in a network, is small relative to the TID horizontal wavelength  $\lambda_{\text{TID}}$  the interpolation will perform better than if  $b > \lambda_{\text{TID}}$ . Therefore, the idea is to use the ratio between the TID wavelength and the average baseline to determine the weight matrix of Eq. 5.13. In the case of a TID event, we test the use of a power defined as:

$$\tilde{p} = \sqrt{\frac{\lambda_{\text{TID}}}{b}}, \quad (5.18)$$

where the root square is used as a damping operator to avoid sudden and too large extremes. The concept is based on the idea that, for a given wavelength, larger surfaces covered by the isoline when the baseline is large are considered and to make the problem more local when the baseline is short. Hereafter, the results obtained using  $\tilde{p}$  are named as WLSL ( $p = \tilde{p}$ ), WLS2DL ( $d = \tilde{d}$  and  $p = 2\tilde{p}$ ), and WLS4DL ( $d = \tilde{d}$  and  $p = 4\tilde{p}$ ).



**Figure 5.1:** Weight isolines for the WLS methods when using the Euclidean distance  $d$  and the modified distance  $\tilde{d}$ . The semi-major axis of the isoline determined by  $\tilde{d}$  is perpendicular to the direction of change of the parameter (green-colored arrow).

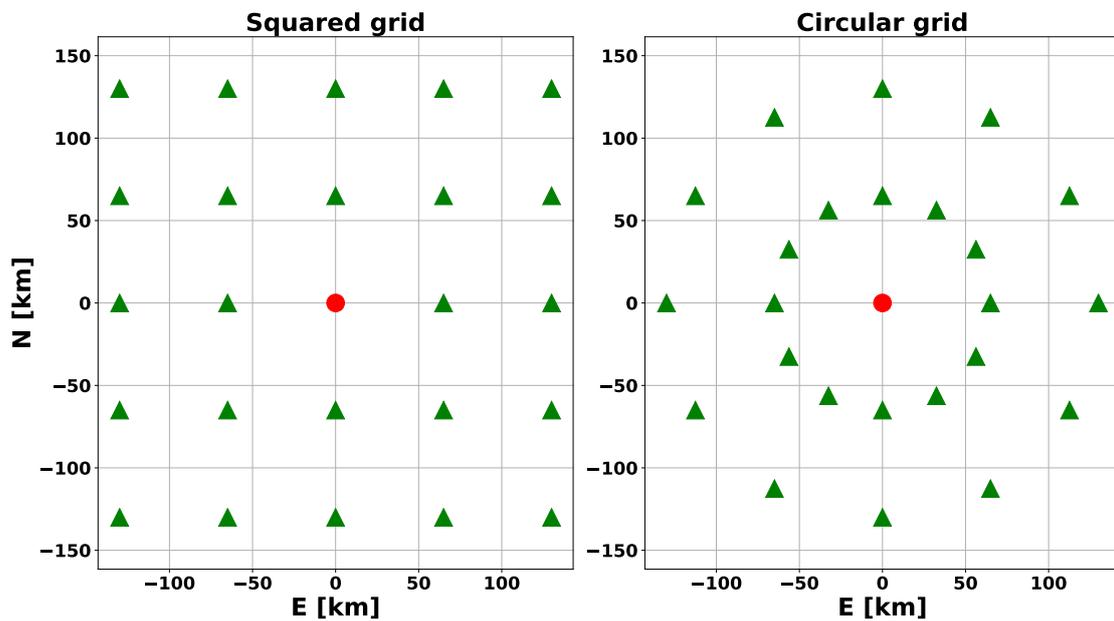
## 5.4 Interpolation of simulated data

### 5.4.1 Simulation setup

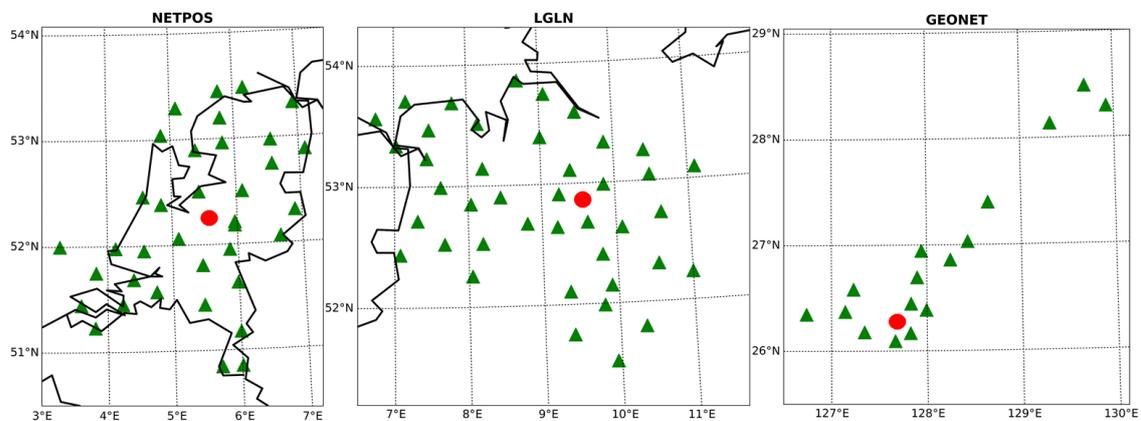
The interpolation quality is assessed in terms of interpolation error by considering different geometries of the known scattered data (i.e. the geometry of the network of GNSS reference stations in this case). Five network geometries are investigated. Two regular grids have been artificially created to evaluate the performance in a homogeneous network. Fig. 5.2 shows a squared grid and circular grid that have been generated for the simulation. Additionally, three real network geometries have been analyzed. These are subsets of the Netherlands Positioning Service (NETPOS), the Landesamt für Geoinformation und Landesvermessung Niedersachsen

- Satellitenpositionierungsdienst (LGLN-SAPOS) and GNSS Earth Observation Network System (GEONET) and are depicted in Fig. 5.3. Moreover, GNSS data from these real networks are evaluated in this chapter (see Section 5.6 and 5.5). In addition, multiple error patterns are analyzed to simulate the variation of the atmospheric parameter over the network by assuming functional models, as introduced in the next sub-section.

The analysis is organized as follows. Firstly, the interpolation error is evaluated for the different geometries and error patterns assuming a user's location in the middle of the network. The latter is an optimal case, in which the quality of the interpolation can be assessed. In this scenario, the sensitivity to the precision of the directional information for the use of the modified distance  $\tilde{d}$  is investigated. Secondly, the dependency on the user location is evaluated by varying the user's position over the area covered by the network. Finally, the benefit of using  $\tilde{d}$  and  $\tilde{p}$  is evaluated when using the WLS approach for the interpolation for the real-network geometries.



**Figure 5.2:** Artificial networks built to simulate the interpolation performance. Here, a baseline of 65 km is shown as example. The red-colored dot shows the location of the rover used in the simulation.



**Figure 5.3:** Real reference stations networks. Subsets of the NETPOS, LGLN-SAPOS and GEONET networks in the Netherlands, Germany and Japan, respectively. The red-colored dot shows the rover location used in the simulation.

### 5.4.2 Generation of error patterns

To investigate the performance of the interpolation along the defined direction, four error patterns are analyzed: step, linear with a step, quadratic and sinusoidal. These four types were selected on the base of results provided by previous authors. For example, a sudden change of 20-30 cm that can be described as a step has been seen in the ZTD analysis carried out during a typhoon in Japan by Hobiger et al. (2010). The variation before and after the step is modeled as either constant or linear. In the ionosphere, a sinusoidal perturbation is a simple approximation that still captures the essential problem faced when interpolating the ionospheric delay in the presence of a TID. It is worth mentioning that in this section there is no distinction between troposphere or ionosphere but all the cases are treated like an atmospheric parameter. For the sake of simplicity, the analysis is carried out in the East-North plane (in a local East, North, Up (ENU) Cartesian coordinate system), and the West-East direction is assumed as direction of change.

Let  $x$  and  $x^*$  be the projections in the direction of variation of  $\mathbf{x}$  and  $\mathbf{x}^*$ , respectively. The step variation is then computed as follows:

$$f(x) = \begin{cases} y_0, & \text{if } x < x^* \\ y_0 + h, & \text{if } x \geq x^* \end{cases}, \quad (5.19)$$

where  $y_0$  is a constant, and  $h$  is the entity of the step. The step is constrained to happen at  $x^*$  to consider a worst-case scenario and highlight its impact. For the remaining part of the paper the linear variation with a step is called *step-linear*. It is calculated in the following way:

$$f(x) = \begin{cases} y_0 + mx, & \text{if } x < x^* \\ y_0 + mx + h, & \text{if } x \geq x^* \end{cases}, \quad (5.20)$$

where  $m$  is the slope of the line and  $h$  the step value. A simple parabolic behavior is assumed for the *quadratic variation*:

$$f(x) = y_0 + kx^2, \quad (5.21)$$

where  $k$  is a constant value. All these first three functions have a range  $y \in [y_0, y_0 + \Delta]$ . Moreover, the step is computed as  $h = \frac{\Delta}{2}$ .

The *sinusoidal variation* is expressed as follows:

$$f(x) = A \sin\left(\frac{2\pi}{\lambda} x + \phi\right), \quad (5.22)$$

where  $A$  is the amplitude of the oscillation,  $\lambda$  the wavelength and  $\phi \in [0, 2\pi]$  the phase, and is used to simulate temporal variation of one cycle.

### 5.4.3 Results for artificial grids

Fig. 5.4 and 5.5 show the quality of the interpolation in the two artificial grids: squared and circular, respectively. The location selected as the query point of the interpolation is exactly in the middle of the networks, as shown in Fig. 5.2 (see red-colored dot) to simulate an optimal scenario for a user within a network. Concerning *step-linear* and *quadratic variations*, the results are visualized in terms of absolute error vs the ratio between the baseline length  $b$  and the maximum variation  $\Delta$ . Regarding the *sinusoidal variation*, the performance is evaluated in terms of normalised error, i.e. the ratio between absolute error and amplitude  $A$  of the oscillation, against the ratio of wavelength  $\lambda_{TID}$  over the baseline  $b$ . This error is the maximum error found by varying the phase  $\phi \in [0, 2\pi]$  (see Eq. 5.22).

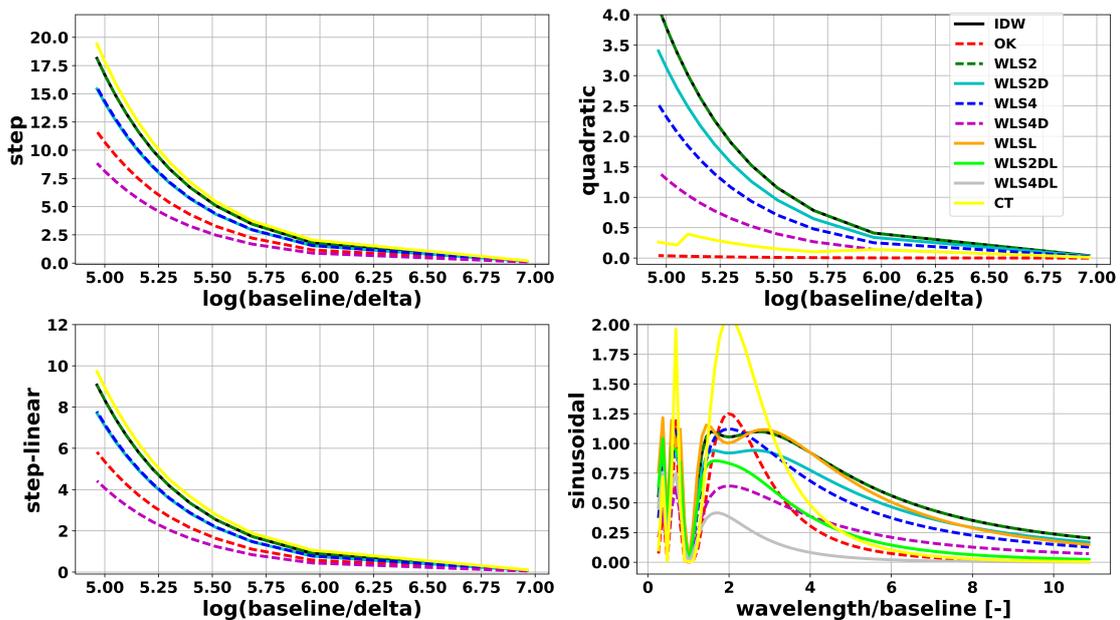
The comparison between WLS2 and WLS2D as well as WLS4 and WLS4D results suggests that the use of  $\tilde{d}$  improves the performance in both geometries for all types of variations tested.

While for large values of the ratio  $\frac{b}{\Delta}$ , i.e. small variations w.r.t. the baseline, all the interpolation techniques perform the same, significant differences are evident for small ratios. For example, a logarithm of the ratio of 5.25 means a variation of roughly 25 cm in a network with a baseline of 50 km.

In that scenario, the use of the modified distance  $\tilde{d}$  with the *step* and *step-linear variations* in the squared-grid network, reduces the error magnitude by a factor of 1.18 between WLS2 and WLS2D and by a factor of 1.6 between WLS4 and WLS4D. Furthermore, it is interesting to notice that, observing Fig. 5.4, it is almost impossible to distinguish the WLS2D from the WLS4 curve, i.e. the use of  $\tilde{d}$  has the same effect of using  $p = 4$  instead of  $p = 2$ . This is due to the squared and regular geometry of the grid. Large improvements are also found for the *quadratic variation*, but the best interpolation techniques are OK and CT for the squared and circular-grid configurations, respectively, as shown in Fig. 5.4 and Fig. 5.5. Furthermore, in a squared-grid network, the use of either the IDW or WLS2 approach leads to the same results, and in Fig. 5.4 the two methods cannot be distinguished. Similarly, in Fig. 5.5 IDW, OK, WLS2, and WLS4 cannot be distinguished, thus showing the same performance. The fact that different interpolation approaches lead to the same performance is related to the perfect regularity of the artificial network and either constant or linear variation.

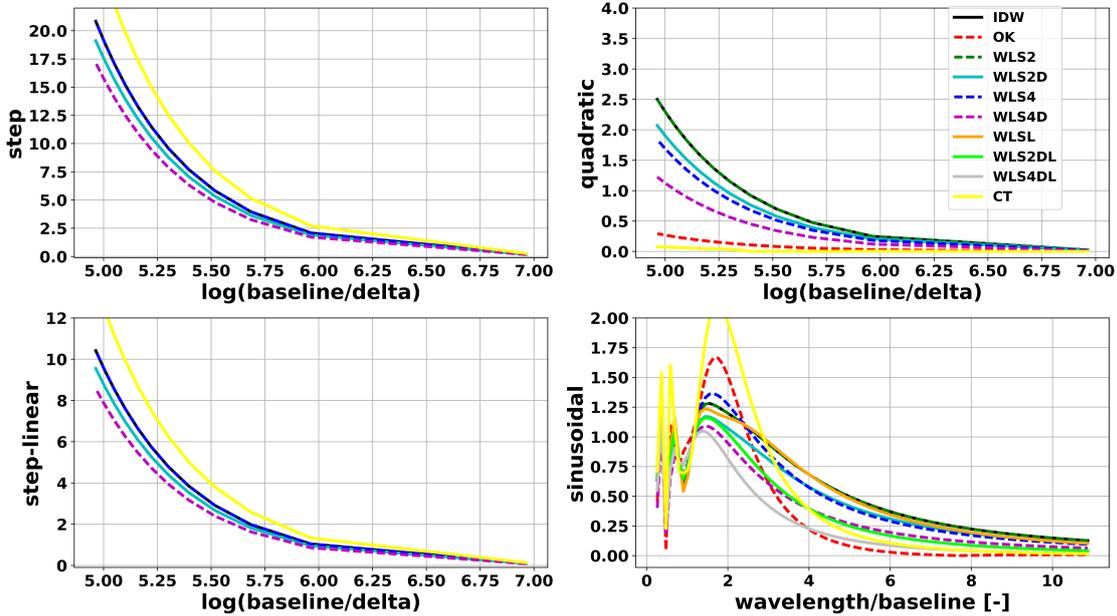
Regarding the *sinusoidal variation*, a clear improvement in using  $\tilde{d}$  and  $\tilde{p}$  can be seen in both Fig. 5.4 and 5.5. However, the two figures suggest that the behavior is different for ratios  $\frac{\lambda}{b}$  greater or lower than one. The modified power seems to improve the interpolation quality for values larger than one, while all the approaches perform similarly for values lower than one. This outcome means that, for small wavelengths relative to the baseline, the use of  $p$  and  $\tilde{d}$  is recommended, while for wavelength larger than the baseline the use of  $\tilde{p}$  and  $\tilde{d}$  is suggested. In Fig. 5.4, it should be noted that values near zero are observed for  $\frac{\lambda}{b} = 1$  for all the methods. They are due to the perfect alignment of the values in the squared grid.

It is worth commenting the CT's results. It can be observed that CT is the technique causing the largest error for *step* and *step-linear variations*, but one of the best (together with OK) for a *quadratic variation*. Concerning the sinusoidal case, CT exhibits a smaller error than IDW,



**Figure 5.4:** Performance of the interpolation techniques in a squared grid network. The four variations of error shape are reported. The rover location is shown as red-colored dot in Fig. 5.2. The different interpolation techniques are indicated with distinct colored lines. WLSL, WLS2DL, and WLS4DL are used only for the *sinusoidal variation*.

WLS2, WLS2D, WLS4, WLSL for ratios  $\frac{\lambda}{b} \geq 4$ . These results are related to the performance of the triangulation process that seems to be more suitable for *quadratic* and *sinusoidal behaviors* (with large wavelengths fixing the baseline length) rather than *linear variations*. Hence, CT appears to be strongly dependent on the type of variation considered.



**Figure 5.5:** Performance of the interpolation techniques in a circular grid network. The four variations are reported. The rover location is shown as red-colored dot in Fig. 5.2. The different interpolation techniques are indicated with distinct colored lines. WLSL, WLS2DL, and WLS4DL are used only for the *sinusoidal variation*.

#### 5.4.4 Results for real-network geometries

Figures 5.6, 5.7, 5.8 show the interpolation performance with the actual network geometries as shown in Fig. 5.3. The red-colored dot shown in Fig. 5.3 indicates the point to be interpolated, i.e.  $\mathbf{x}^*$ . The methodology employed to visualize the results is the same as with the artificial networks, and similar conclusions can be drawn. As a general comment, it can be observed that a *quadratic variation* can be handled by any interpolation technique better than other types of behavior.

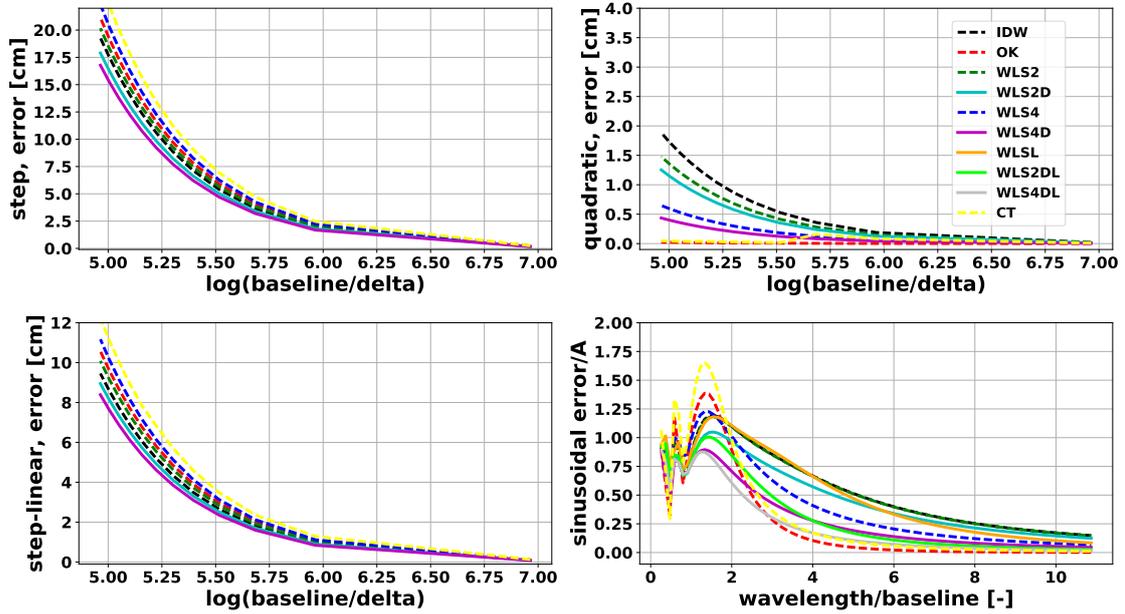
Overall, Figures 5.6- 5.12 suggest that, in networks like NETPOS, LGLN, and GEONET, for *step* and *step-linear behaviors* the use of  $\tilde{d}$  marks an improvement vs the use of  $d$  in the interpolation error. Furthermore, while increasing the distance weight  $p$  from two to four reduces the error in the NETPOS network, it does not impact the results in the LGLN network positively, increasing the interpolation error. Again, in a *quadratic variation-scenario*, the OK approach seems to be most appropriate for the problem. Also, in a *sinusoidal variation-scenario* the benefit of using  $\tilde{d}$  is observed along with a further improvement employing  $\tilde{p}$  for  $\frac{\lambda}{b} > 1$ .

The impact of network geometry itself is also highlighted. While the NETPOS and LGLN are quite dense and regular networks, the geometry of the GEONET network is more sparse, and the consequences of these differences can be observed in Fig. 5.8. Concerning either a *step* or *step-linear variation*, a significant difference among methods can be observed only for  $\log(\frac{b}{\Delta}) < 5.6$ , the use of  $\tilde{d}$  reduce the error, but IDW performs the best. However, like in the other networks, OK shows the smallest error in a quadratic variation-scenario. In a sinusoidal behavior-scenario, as in the other cases, the benefit of  $\tilde{p}$  is seen for  $\frac{\lambda}{b} > 1$ .

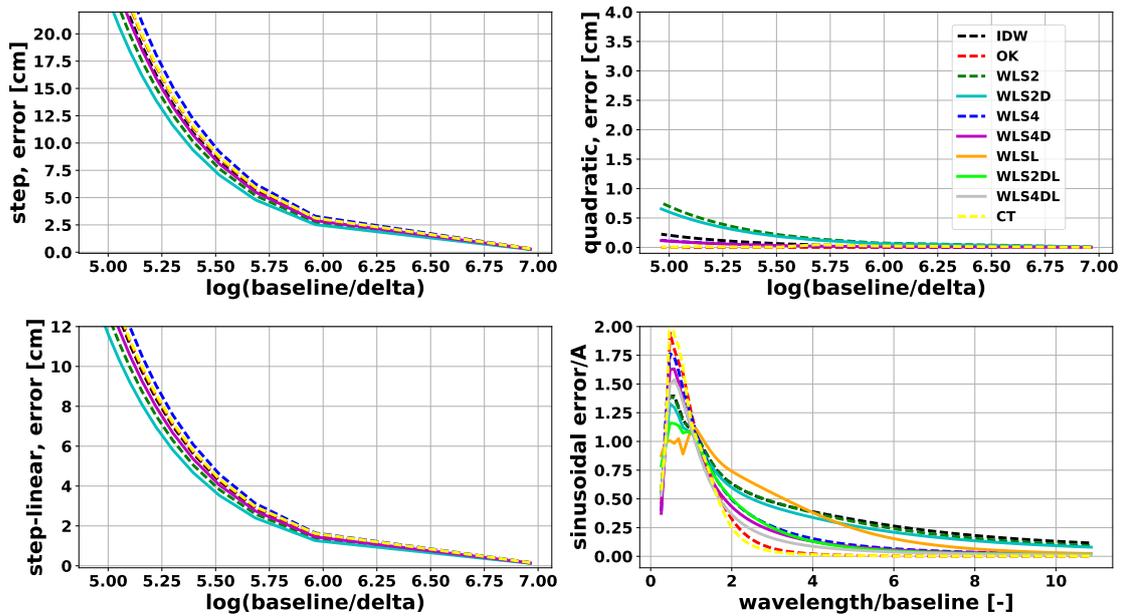
The results presented so far suggest that the CT technique would be appropriate mostly for

#### 5.4. Interpolation of simulated data

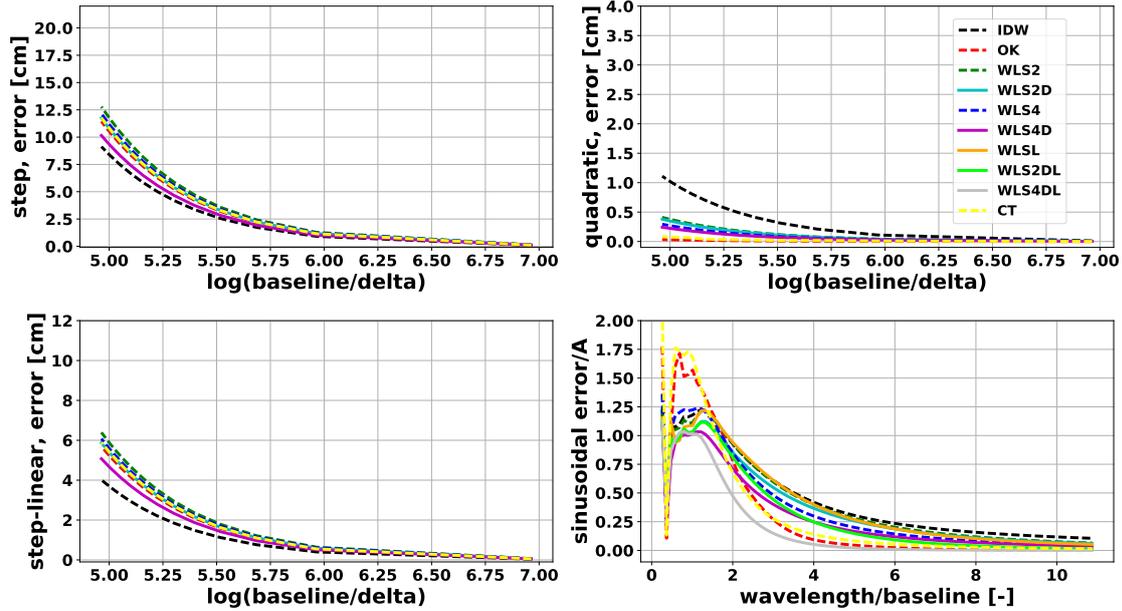
*quadratic* and *sinusoidal variations* only. As the performance in terms of interpolation error is similar or worse than OK, hereafter, the CT technique is no longer considered.



**Figure 5.6:** Performance of the interpolation techniques using artificial values but the geometry of the NETPOS network. The rover location is shown as red-colored dot in Fig. 5.3. The four variations are reported. The different interpolation techniques are indicated with distinct colored lines. WLSL, WLS2DL, and WLS4DL are used only for the *sinusoidal variation*.



**Figure 5.7:** Performance of the interpolation techniques using artificial values but the geometry of the LGLN network. The rover location is shown as red-colored dot in Fig. 5.3. The four variations are reported. The different interpolation techniques are indicated with distinct colored lines. WLSL, WLS2DL, and WLS4DL are used only for the *sinusoidal variation*.



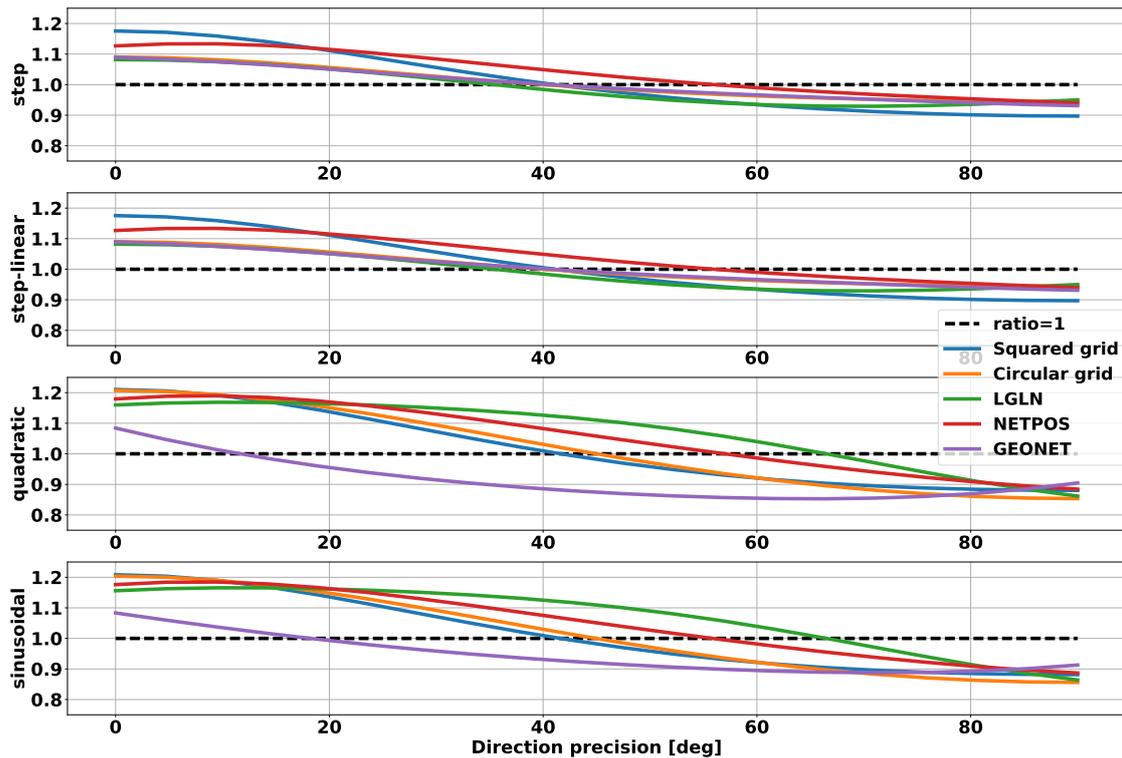
**Figure 5.8:** Performance of the interpolation techniques using artificial values but the geometry of the GEONET network. The rover location is shown as red-colored dot in Fig. 5.3. The four variations are reported. The different interpolation techniques are indicated with distinct colored lines. WLSL, WLS2DL, and WLS4DL are used only for the *sinusoidal variation*.

Summarizing, the simulations in an optimal use-case scenario (user in the middle of the network) indicate that the use of  $\tilde{d}$  instead of  $d$  is suggested for large variations  $\Delta$  for a fixed baseline  $b$ . In general, the use of  $\tilde{p}$  instead of  $p$  is recommended for ratios  $\frac{\lambda}{b} \geq 1$ .

#### 5.4.5 Required precision for the directional information

The direction of variation of the SSR parameter needs to be estimated either internally or externally to the positioning algorithm. Therefore, the interpolation sensitivity to the precision of the estimation of the direction is assessed. For this, the results obtained with WLS2 against WLS2D are compared. Fig. 5.9 depicts the ratio of the errors obtained using the two techniques, i.e.  $\frac{\text{WLS2}}{\text{WLS2D}}$ . Hence, only ratio values greater than one indicate a benefit of using  $\tilde{d}$ . The improvement factor varies from a maximum of roughly 1.21 to a minimum value of about 0.85. It can be observed that for a precision better than 10 deg the WLS2D approach has a positive effect in all cases. Concerning *step* and *step-linear variations*, the threshold value between improvement and deterioration is around 35 deg for all the analyzed networks. Regarding a *quadratic variation*-scenario, the precision constrain is more relaxed, but for the GEONET network that requires a precision better than 12 deg. The *sinusoidal* scenario shows a similar outcome. The GEONET network requires a precision lower than 20 deg, while the LGLN network shows a benefit for values of the precision up to 70 deg. In the other networks, the negative effect of using  $\tilde{d}$  starts for precision values between 40 and 60 deg.

It can be concluded that the uncertainty of the direction can be relatively considerable (i.e., in general, up to about 40 deg). However, significant improvements can be obtained only with a good knowledge of the direction angle (i.e.  $\leq 10$  deg).



**Figure 5.9:** Sensitivity to the error in the estimation of the direction of propagation of the parameter. The ratio between the WLS2 and WLS2D method is shown for the different networks and types of variation. A ratio larger than one indicates an improvement w.r.t. the WLS2 approach.

### 5.4.6 Variation of the interpolation position

The interpolation results shown so far represent the case of a GNSS user in the middle of a reference station network. However, that is not often the case in a real scenario. Therefore, it's worth investigating the interpolation quality considering  $\mathbf{x}^*$  varying throughout the network. Considering the three real network geometries NETPOS, LGLN, and GEONET, the RMSE of the interpolation is evaluated for all the different perturbation behaviors. The RMSE is computed over the variation of either  $\log \frac{b}{\Delta}$  or  $\frac{\lambda}{b}$ . The user position  $\mathbf{x}^*$  has been varied among 144 points over each network. Table 5.1 reports the best and second-best interpolation in terms of percentage of points over the 144 tests with the best RMSE.

These simulations suggest that WLS is the best approach, but OK suits the quadratic variation better. Furthermore, Table 5.1 suggests that IDW is the most accurate technique for the GEONET geometry. However, it has to be mentioned that most of the 144 points are outside the network. Therefore, in this case, it would be more appropriate to call the operation extrapolation rather than interpolation. As a consequence, Table 5.1 suggests that the IDW method works well in case of the need for extrapolation.

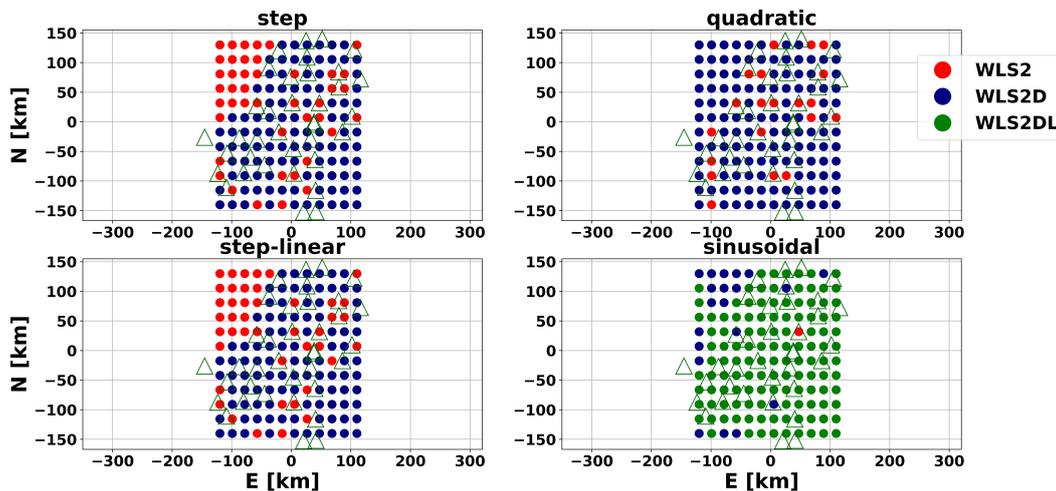
In general, the results indicate that  $\tilde{a}$  is always recommended, while  $\tilde{p}$  is suggested for all the networks except GEONET. Also, all the analyzed cases seem to benefit from the use of  $p = 4$  instead of  $p = 2$  except the LGLN network.

**Table 5.1:** Summary of the best and second best interpolation in terms of RMSE in the three real networks that have been analyzed. The four variations are evaluated and the percentage of cases over the 144 grid points reported.

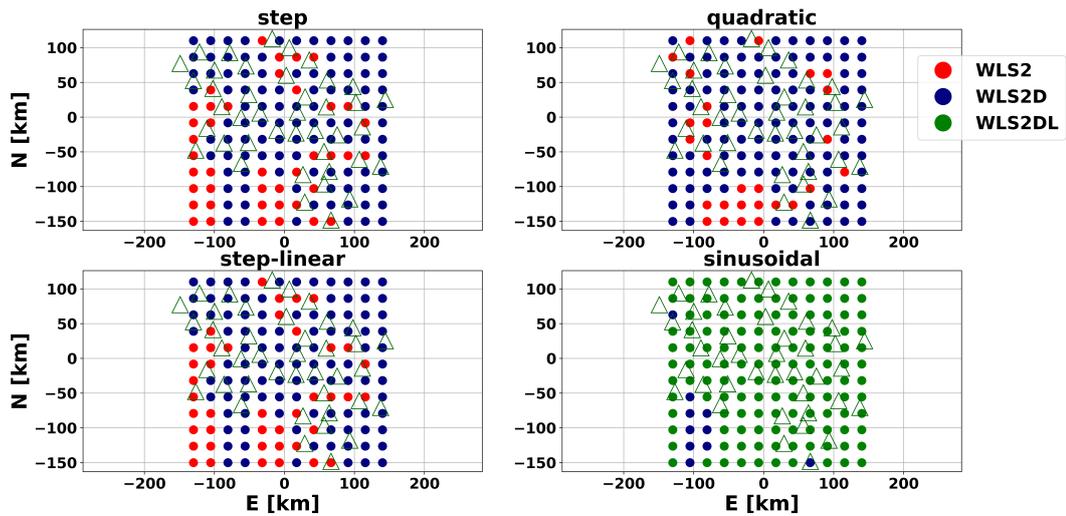
Variation	NETPOS	LGLN	GEONET
step	WLS4D: 45% WLS4 : 15%	WLS4D: 42% WLS2D: 17%	IDW: 29% WLS4D: 21%
step-linear	WLS4D: 42% WLS4 : 19%	WLS4D: 41% WLS4 : 16%	IDW: 34% WLS4D: 19%
quadratic	OK: 54% WLS4D: 37%	OK: 46% WLS4D: 32%	WLS4D: 45% OK: 31%
sinusoidal	WLS4DL: 37% WLS2DL: 31%	WLS2DL: 50% WLS4DL: 23%	IDW: 56% WLS4D: 17%

### 5.4.7 Variation of the distance: WLS methods comparison

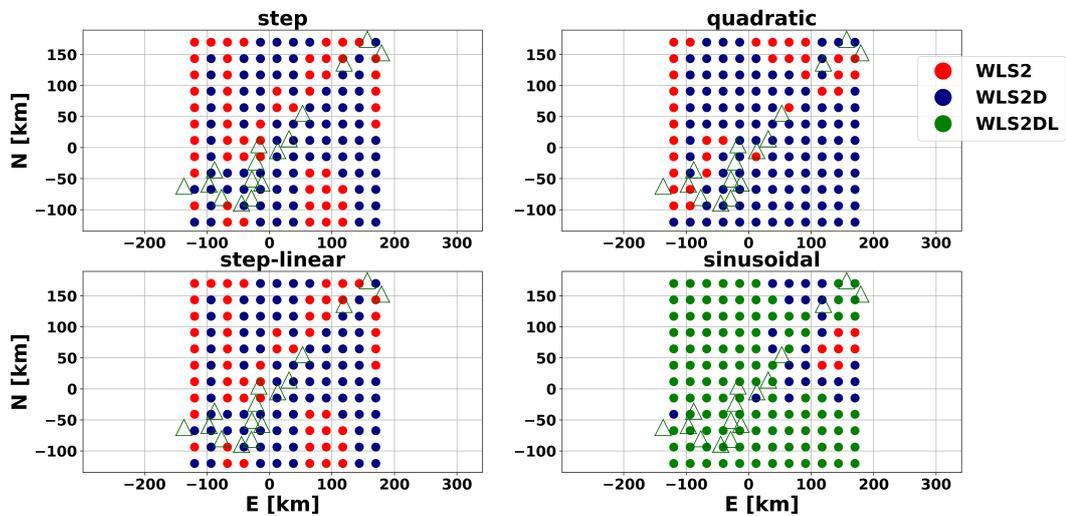
Another aspect to assess is the benefit of using  $\tilde{d}$  and  $\tilde{p}$ . To do that, the focus is only on the WLS technique. Fig. 5.10, 5.11, and 5.12 show the map of the best WLS interpolation. The color of the dots indicates which method is the best for a given location and error distribution. The figures indicate a benefit of using  $\tilde{d}$ , i.e. WLS2D (blue-colored dots), for the *step*, *step-linear* and *quadratic variation* in more than the 80% of the user points. Moreover, applying  $\tilde{p}$ , i.e. WLS2DL (green-colored dots), in case of a *sinusoidal behavior*, the RMSE is smaller in more than the 86% of the analyzed cases. It is essential to mention that the mapping of the best interpolation technique is based on RMSE. Hence, it does not reflect the actual picture during severe weather and space weather events. However, the analysis provides an insight into the potential benefit of using  $\tilde{d}$  and  $\tilde{p}$ .



**Figure 5.10:** Map of the best WLS approach in the NETPOS network using artificial data. Red-colored dots indicate that the best WLS approach is WLS2, blue-colored dots indicate WLS2D and green-colored dots indicate the WLS2DL technique. The benefit of using  $\tilde{d}$  and  $\tilde{p}$  is evaluated. The comparison among techniques is based on RMSE. All the variations tested are reported.



**Figure 5.11:** Map of the best WLS approach in the LGLN network using artificial data. Red-colored dots indicate that the best WLS approach is WLS2, blue-colored dots indicate WLS2D and green-colored dots indicate the WLS2DL technique. The benefit of using  $\tilde{d}$  and  $\tilde{p}$  is evaluated. The comparison among techniques is based on RMSE. All the variations tested are reported.



**Figure 5.12:** Map of the best WLS approach in the GEONET network using artificial data. Red-colored dots indicate that the best WLS approach is WLS2, blue-colored dots indicate WLS2D and green-colored dots indicate the WLS2DL technique. The benefit of using  $\tilde{d}$  and  $\tilde{p}$  is evaluated. The comparison among techniques is based on RMSE. All the variations tested are reported.

In summary, it can be said that the modified distance  $\tilde{d}$  is recommended for rovers within the network. In contrast, for locations outside the network, WLS2 (red-colored dots) might perform better (situation closer to extrapolation than interpolation). This is expected for the construction of the WLS2D method. Since it is a directional method, it is affected by the geometry of the network. Hence, missing stations along the perpendicular to the direction of variation might cause an increased interpolation error. However, the investigation carried out

in this research is for GNSS-users within the network of reference stations. Accordingly, the results are promising for the aforementioned scenario. Furthermore, if the variation can be associated with a sinusoidal behavior, the use of  $\tilde{p}$  can further reduce the interpolation error.

## 5.5 ZTD interpolation during severe weather events

### 5.5.1 Data-sets and analysis concept

A period with strong weather fluctuations took place in the Netherlands during DOY 170-178 of the year 2017. In particular, DOY 173 was an intensely rainy day with a significant amount of precipitations. The troposphere effect on GNSS measurements of 36 stations of the NETPOS network (see Fig. 5.3) is investigated during DOY 170, 172-174, 177. The other event taken into account is the Xavier windstorm that occurred in northern Germany on October 5th 2017, DOY 278. Five days around that date, DOY 275-279, are considered for the analysis of 41 GNSS stations of the LGLN-SAPOS network (see Fig. 5.3). The complete data-set is hence composed of GNSS measurements of two independent networks, provided as RINEX 2.11 files. Both periods involve severe weather events.

The SSR parameters, i.e., the tropospheric delay, in this case, are estimated in a network processing that employs an uncombined observation model (see Section 2.4) using the Geo++'s software GNSMART. The algorithm processed dual-frequency (L1-L2) dual-constellation (GPS-GLONASS) observations. The ZTD computation is implemented by using an a priori model based on the Saastmoinen model with weather parameters from UNB3 (Leandro et al., 2006). Station dependent correction factors to this model are then estimated based on the actual GNSS observations.

The resulting ZTDs have been compared with the IGS and the GFZ tropospheric products. The delays of five GNSS stations of the Integrated Geodetic Reference Network of Germany (GREF) (in common among all the datasets), referring to the period of the Xavier windstorm, have been computed for the comparison, indicating an overall good agreement. The largest difference of 2.5 cm has been detected during the periods of large variations of the ZTD. It might be due to the different time intervals of integration used (e.g. the GNSMART results are considered with a 10 s time-rate, while the GFZ ones with a 15 minutes time-rate).

As mention in Section 5.3, the focus is on a 2D planar variation of the parameters. As a consequence, the ZTD shown hereafter are corrected for the effect of the height of the station.

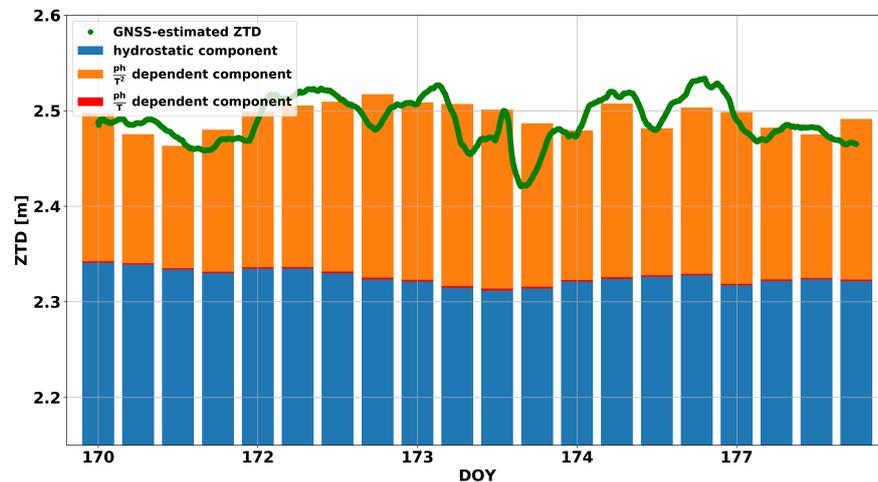
### GNSS-estimated ZTD vs NWM-based ZTD

It has been mentioned above that it is aimed to assess the potential of using external information like NWM for improving ZTD interpolation. Different types of tropospheric information can be extracted from NWMs. Among others, also physical quantities (e.g. pressure, temperature, humidity) can be retrieved from NWM databases, e.g. the ECMWF. As a comparison, the ZTD has been reconstructed from NWM data to evaluate the agreement with GNSS-estimated ZTD. Here, the ECMWF database has been used, which provides data for 37 pressure levels from 1 hPa to 1013.25 hPa. The integration through these levels generates the ZTD. The ECMWF values considered are the so-called *analyses* values. They are available daily every six hours (i.e. at 00, 06, 12, 18 UTC time). The validity is for the time they referred to. More information about ECMWF products can be found in ECMWF (2020).

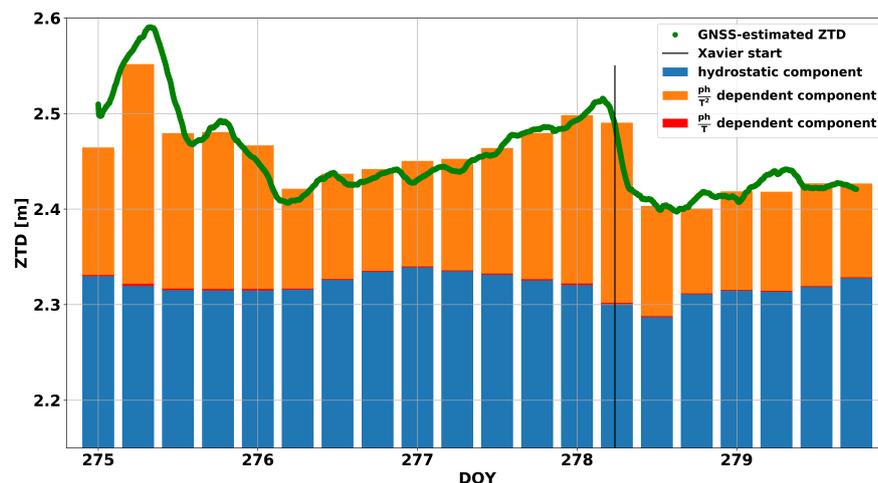
The refractivity  $N$  is computed using Eq. 2.6. The partial pressure of water vapor  $e$  can be expressed in terms of specific humidity  $h$  (which can be retrieved from NWMs) using the following relationship (e.g. Wallace and Hobbs, 2006; Zus et al., 2012):

$$e = \frac{hp}{\frac{R_d}{R_w} + (1 - \frac{R_d}{R_w})h}, \quad (5.23)$$

where  $R_d$  and  $R_w$  have the same meaning introduced in Eq. 2.9. As an example, Fig. 5.13 and Fig. 5.14 show the comparison between the GNSS-estimated (green-colored line) and NWM-based (colored bars) ZTD for a specific location in the NETPOS and LGLN network, respectively. The position considered in the NETPOS network is Utrecht (station UTR2, lat=52.4075°, lon=9.8015°), while for the LGLN network it is Hannover (station 0688, lat=52.0626°, lon=5.0931°).



**Figure 5.13:** GNSS-estimated (green-colored dots) and NWM-based ZTD (colored bars) for the Utrecht location during the period of the NETPOS data-set. Each bar is a ZTD value reconstructed from NWM. The different colors of the bar indicate the component dependency: the blue-colored part is the hydrostatic component, the orange-colored part the wet component depending on temperature squared and the red-colored part the wet component depending on the temperature.



**Figure 5.14:** GNSS-estimated (green-colored dots) and NWM-based ZTD (colored bars) for the Hannover location during the period of the LGLN data-set. The approximate start of the Xavier windstorm in Hannover is indicated by a black-colored line. Each bar is a ZTD value reconstructed from NWM. The different colors of the bar indicate the component dependency: the blue-colored part is the hydrostatic component, the orange-colored part the wet component depending on temperature squared and the red-colored part the wet component depending on the temperature.

The reported GNSS-estimated ZTD has been smoothed using an exponential moving average filter with a 1800 s window. The start of the Xavier windstorm in Hannover is indicated by a black-colored line in Fig. 5.14. In both Fig 5.13 and Fig. 5.14, the different contributions to the ZTD are highlighted (see Eq. 2.8 and Eq. 5.23 for the construction of the different components). It can be seen that the hydrostatic component (blue-colored bar) is the main contributor to the delay. However, the large fluctuations are mostly provided by the wet component depending on the  $\frac{p}{T^2}$  ratio. This result confirms the strong variability of the tropospheric delay with the local weather conditions.

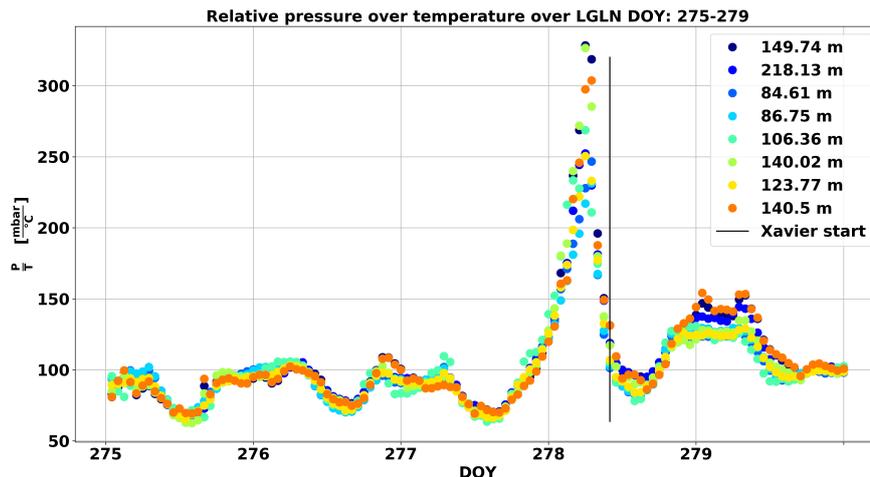
Magnitude differences lower than 4 cm are observed. Nevertheless, a different temporal resolution needs to be considered in the comparison. Overall, it can be observed that there is good agreement between the variations of the GNSS-estimated and NWM-based ZTDs.

### Ground-based weather data comparison

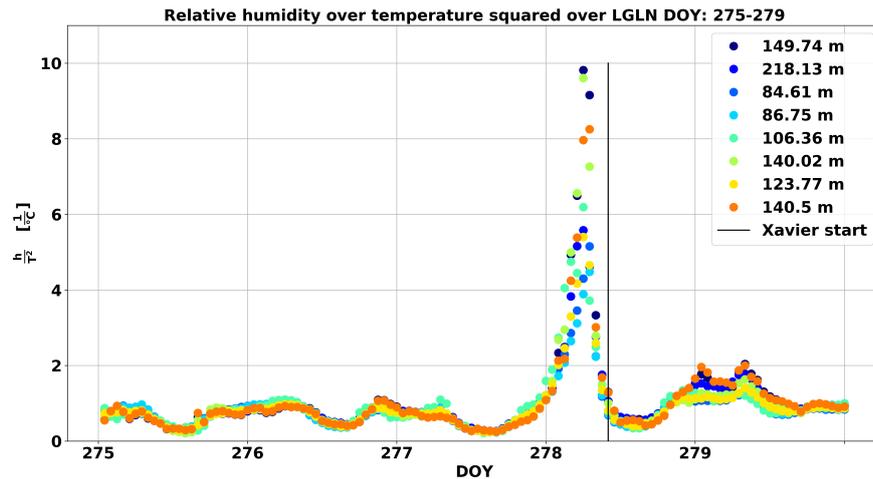
Nowadays, many ground stations are sensing the weather parameters of the troposphere. Here, the aim is to investigate possible correlations between the variation of GNSS-estimated ZTDs and climate weather parameters collected by ground stations. As an example, the LGLN data-set has been analyzed.

The ZTD fluctuation shown in Fig. 5.14 is compared with the variation of published weather data of the Deutscher Wetterdienst (DWD). The DWD values have been interpolated for the locations of Hannover and of the seven LGLN stations closest to Hannover. For the locations mentioned above, Fig. 5.15, Fig. 5.16 and Fig. 5.17 depict the total pressure over temperature, the specific humidity over temperature squared and the amount of precipitation, respectively. The ellipsoidal height of the positions considered are reported in the legend of the figures (Hannover station's ellipsoidal height is 123.77 m).

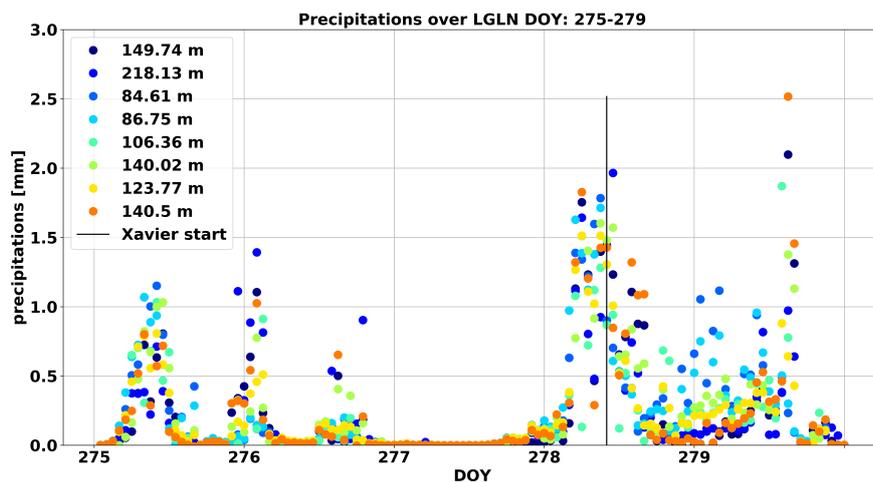
It can be observed that the largest values and variations of the ratios  $\frac{p}{T}$  and  $\frac{h}{T^2}$  are located before the beginning of the storm (indicated by a black-colored line in the figures). A large ZTD fluctuation before Xavier's start can also be observed in the GNSS-estimated ZTD (see Fig. 5.14). However, the other strong variation seen in the GNSS-estimated ZTDs during DOY 275 is not described by the ground-based ratios. Therefore, the analysis suggests that surface-based weather data fails in detecting some significant ZTD changes.



**Figure 5.15:** Pressure over temperature DWD data for the Hannover location and the seven LGLN stations closest to Hannover during the period of the LGLN data-set. The different colors indicate the stations organized by ellipsoidal height (reported in the legend in meters).



**Figure 5.16:** Humidity over temperature squared DWD data for the Hannover location and the seven LGLN stations closest to Hannover during the period of the LGLN data-set. The different colors indicate the stations organized by ellipsoidal height (reported in the legend in meters).



**Figure 5.17:** Precipitations DWD data for the Hannover location and the seven LGLN stations closest to Hannover during the period of the LGLN data-set. The different colors indicate the stations organized by ellipsoidal height (reported in the legend in meters).

A different conclusion can be drawn for the amount of precipitation. In fact, the comparison of Fig. 5.14 and Fig. 5.15-5.17 indicates that the amount of rainfall provides more valuable information about peaks in the ZTD values than the pressure, temperature, and humidity parameters at the station level. However, there is not a direct correlation between ZTD and precipitations. As an example, DOY 279 was a rainy day without large ZTD fluctuations.

It can be concluded that information on atmospheric parameters at different tropospheric heights is required for a better description of the tropospheric delay, as demonstrated by the NWM integration shown in Fig. 5.13 and Fig. 5.14.

### Horizontal tropospheric gradients

Apart from zenith delays, also tropospheric horizontal gradients are routinely computed. These are of particular interest for interpolation, since they indicate the spatial variation of the tropospheric delays. Tropospheric horizontal gradients kindly provided by GFZ are used for

this study. They are retrieved from two different numerical weather models: GFS for the LGLN test case and the ECMWF Era-Interim for the NETPOS test case. In both data-sets, the spatial resolution is 0.25 deg both in latitude and longitude. For the sake of completeness, it is worth mentioning that, concerning the GFS data covering the LGLN network, the deviation from ERA-interim is 0.25 mm (RMS) in the gradient components. GFZ provided the tropospheric products with a time-step of three hours, and they are then linearly interpolated in time in order to consider data with a resolution of one hour.

The tropospheric horizontal gradient is evaluated as in MacMillan and Ma (1997):

$$G = 10^{-6} \int_0^H z \nabla_{\mathbf{x}} N(\mathbf{x}, z) dz, \quad (5.24)$$

where the integration over the height  $z$  is from the surface of Earth to a height  $H$ , above which the atmosphere does not impact the delay significantly, while  $\mathbf{x}$  is a 2D vector in the North-East plane. More details about the derivation of the horizontal gradient can be found in Davis et al. (1993). It is important to remark that the tropospheric horizontal gradient defined by Eq. 5.24 is not a mathematical gradient, but the so-called delay gradient (Davis et al., 1993). The delay gradient, in fact, is computed by integrating along  $z$  the spatial mathematical gradient  $\nabla_{\mathbf{x}}$  of the refractivity  $N(\mathbf{x}, z)$ .

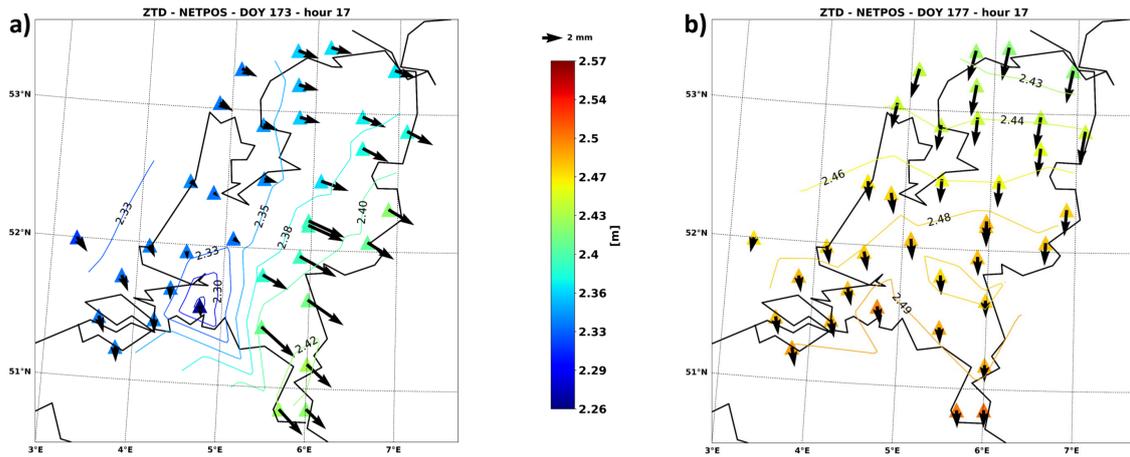
### 5.5.2 Comparison of NWM gradients and spatial distribution of GNSS-estimated ZTD

To investigate whether the tropospheric gradients reflect the tropospheric delay variations for typical inter-station distances of the test networks, the direction of variation of GNSS-derived ZTD values are compared with the NWM derived gradients. Fig. 5.18 and 5.19 show this comparison during hours with particularly high ZTD variations for the NETPOS and LGLN networks, respectively. In particular, maximum variations over the whole network larger than 10 cm are depicted in the a-panel of Fig. 5.18 and Fig. 5.19.

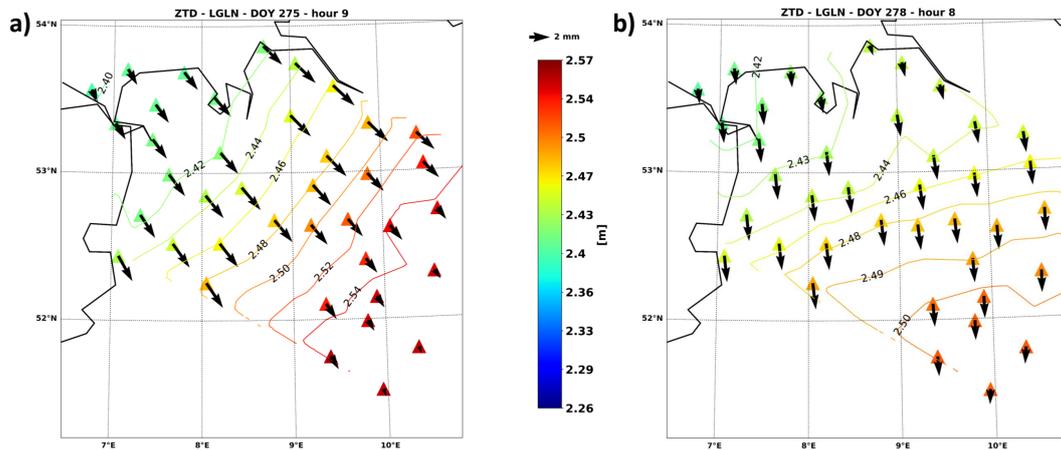
Here, based on what reported in, e.g., Douša et al. (2016); Zus et al. (2019a), tropospheric gradients are considered large when the magnitude of the gradients is larger than 2 mm for a wide area covering the network. This value is roughly four times what expected in middle latitudes (see Section 2.1). Four examples, two cases for each network, are reported in Fig. 5.18 and Fig. 5.19. The two figures depict four different hours where the ZTD varies between 2.26 and 2.57 m over the networks with horizontal gradients larger than 2 mm over many station locations. It is worth mentioning that the ZTDs displayed in the figures mentioned above are average values over one-hour data-sets.

Fig. 5.18 and 5.19 indicate a good agreement between the direction of NWM horizontal gradients and GNSS ZTD spatial variations. In fact, the magnitude of the GNSS-estimated ZTDs increases along the direction of the NWM horizontal gradients. Therefore, the intrinsically local gradient direction defined by Eq. 5.24 should give valuable information for ZTD interpolation also at a more regional scale, including multiple reference stations around an interpolation position.

Following the simulations carried out in Section 5.4, the four data-sets depicted in Fig. 5.18 and Fig 5.19 fall in the case where we have a baseline of roughly 40 km and can have a variation  $\Delta$  of roughly 0.15 m, i.e. a logarithmic value  $\log(\frac{b}{\Delta}) \approx 5.43$ . When a step in the ZTD variation or a similar-quadratic variation along the direction of the gradient occurs, we should observe a smaller error for approaches like WLS2D and WLS4D or OK in the similar-quadratic variation. For example, Fig. 5.18(a) shows a configuration similar to a linear-step scenario for the interpolation of station locations in the middle of the network where the horizontal gradients are large.



**Figure 5.18:** Average ZTD values computed from GNSS data for the NETPOS network during hour 17 of DOY 173 (a) and hour 17 of DOY 177 (b). The triangles indicate the locations of the reference stations, colored by considering the magnitude of the ZTDs (see color-bar in the legend). The arrows show the NWM horizontal gradients (the scale is reported in the legend).



**Figure 5.19:** Average ZTD values computed from GNSS data for the LGLN network during hour nine of DOY 275 (a) and hour eight of DOY 278 (b). The triangles indicate the locations of the reference stations, colored by considering the magnitude of the ZTDs (see color-bar in the legend). The arrows show the NWM horizontal gradients (the scale is reported in the legend).

### 5.5.3 Interpolation results using directional techniques

Reference ZTD values have been generated for all stations by network processing of the complete network for the whole time-span. The respective test station has been taken out of the network to interpolate at its location. As a consequence, the interpolation result is compared to the reference ZTD generated for this station in the initial network processing. The interpolation process has been repeated for all the stations of each network, i.e., 36 and 41 for the LGLN and NETPOS network, respectively. All the techniques show very similar RMS of interpolation error when considering the entire time-span and all stations for each network. One could say that all the different approaches perform the same in terms of RMSE within 1 mm, but the IDW approach that shows 2 mm discrepancies. However, in periods where considerable ZTD variations over the network (e.g., around 17 cm like in Fig. 5.18 and 5.19) and large tropospheric gradients occur, more significant differences arise among the methods used.

Table 5.2 and Table 5.3 summarizes the interpolation results in terms of RMSE during selected hours with sharp weather fluctuations. Considering all the test locations (i.e. the posi-

tions of the reference stations), the *percentage of success* reported in the table is the number of cases in which an interpolation method exhibits the smallest RMSE. Table 5.2 reports the best technique in percentage considering all the cases, while Table 5.3 compares each technique w.r.t. WLS2, which is assumed as the reference approach. In Table 5.3, the percentage of stations showing a ratio larger than one is reported for all the techniques used. A ratio larger than one means that the approach at the denominator marks a lower RMSE than the WLS2 approach.

Table 5.2 indicates that in both NETPOS data-sets, the IDW is the best interpolation technique in 28% and 22% of the cases over all the stations (i.e. 36 cases), in the two perturbed hours. However, Table 5.3 suggests that when comparing the RMSE of only the IDW and WLS2 techniques, the use of the IDW approach would be recommended just in roughly 30% of the cases. Furthermore, Table 5.3 suggests that the use of the WLS2D instead of the WLS2 technique is recommended in all the data-sets considered. In particular, during hour nine of DOY 275, the employment of the WLS2D technique would show an improvement of the RMSE in 80% of the station locations of the LGLN network.

**Table 5.2:** Summary of interpolation results of ZTD estimated using GNSS data from the NETPOS and LGLN networks. Specific hours showing strong ZTD gradients are analyzed. The best interpolation technique is reported in terms of the *percentage of success* considering all the stations of the network. The comparison among stations is made in terms of RMSE. The RMSE is computed over one-hour data-sets.

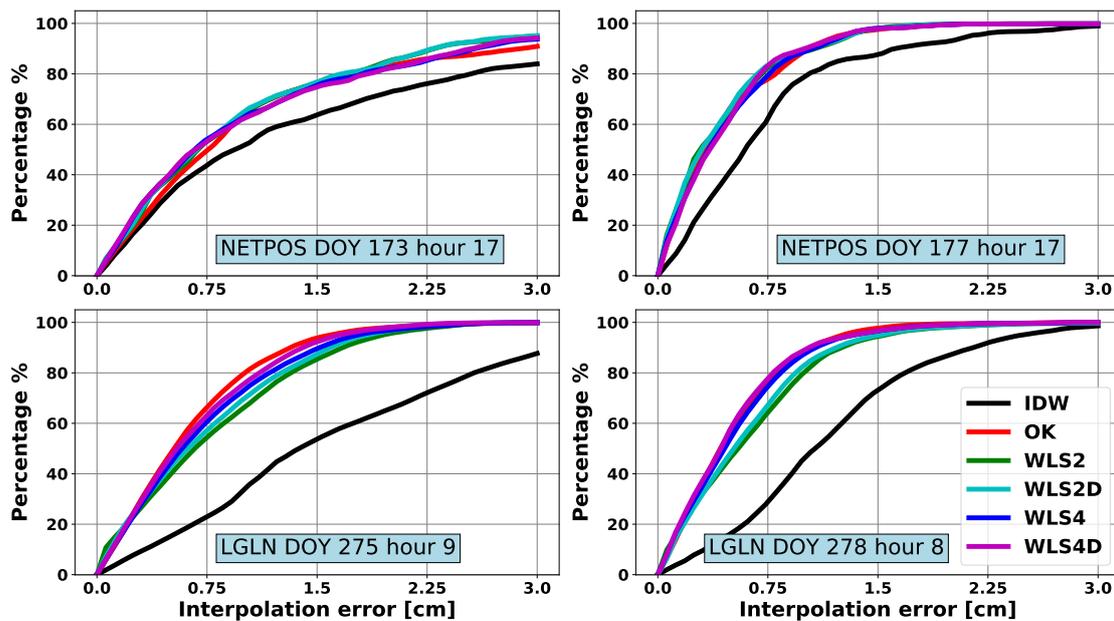
Data-set	Best technique
NETPOS,DOY 173, hour 17	OK, IDW: 28%
NETPOS, DOY 177, hour 17	WLS2, IDW: 22%
LGLN, DOY 275, hour 9	OK: 41%
LGLN, DOY 278, hour 8	OK: 37%

**Table 5.3:** Summary of interpolation results of ZTD estimated using GNSS data from the NETPOS and LGLN networks w.r.t. WLS2 method. Specific hours showing strong ZTD gradients are analyzed. The WLS2 method is considered as reference and the RMSE ratio with the other techniques investigated. The RMSE is computed over one-hour data-sets. The percentage of stations showing a ratio larger than one is reported for all the techniques used. A ratio larger than one means that the approach marks a lower RMSE than the WLS2 approach.

Data-set	WLS2/IDW $\geq 1$	WLS2/OK $\geq 1$	WLS2/WLS2D $\geq 1$	WLS2/WLS4 $\geq 1$	WLS2/WLS4D $\geq 1$
NETPOS DOY 173 hour 17	36 %	50%	58%	50%	58%
NETPOS DOY 177 hour 17	31 %	42%	53%	42%	44%
LGLN DOY 275 hour 9	10 %	70%	80%	63%	68%
LGLN DOY 278 hour 8	17 %	68%	71%	66%	71%

One of the aspects highlighted by the results of Table 5.3 is about the use of the power  $p$  in the weight. The use of  $p = 4$  instead of  $p = 2$ , giving more weight to the closest stations, would help to reduce the error in the LGLN data-sets, but not in the NETPOS data-sets. This difference is probably due to the more regular geometry of the LGLN network. Overall, Table 5.3 recommends the use of  $\tilde{d}$  instead of  $d$  to reduce the RMSE. For the same purpose, the OK technique seems to be preferred to the WLS2 method. The outcome is valid during the hours with strong perturbations that have been investigated, and the general conclusions agree with the outcome of the simulation (see Section 5.4).

Another quality indicator of the interpolation is the cumulative error. Fig. 5.20 suggests that, during the perturbed periods, all the errors are below 3 cm, except the NETPOS data-set of DOY 173, hour 17 shows larger errors. Furthermore, it might be observed that all the interpolation techniques show a similar error distribution but the IDW approach, which, in general, accumulates bigger errors. The cumulative error analysis yields similar conclusions obtained from the RMSE analysis. Fig. 5.20 suggests that the employment of  $\tilde{d}$  instead  $d$  (i.e. either WLS2D or WLS4D instead of either WLS2 or WLS4) reduce the large errors slightly. Furthermore, to give more weight to the closer stations (e.g., WLS4 instead of WLS2), changes the error distribution (smaller errors) in the LGLN data-sets but not in the NETPOS data-sets. Finally, OK seems to provide the best error distribution except for the NETPOS network during hour 17 of DOY 173, where either a WLS2 or WLS2D approach would be preferable.



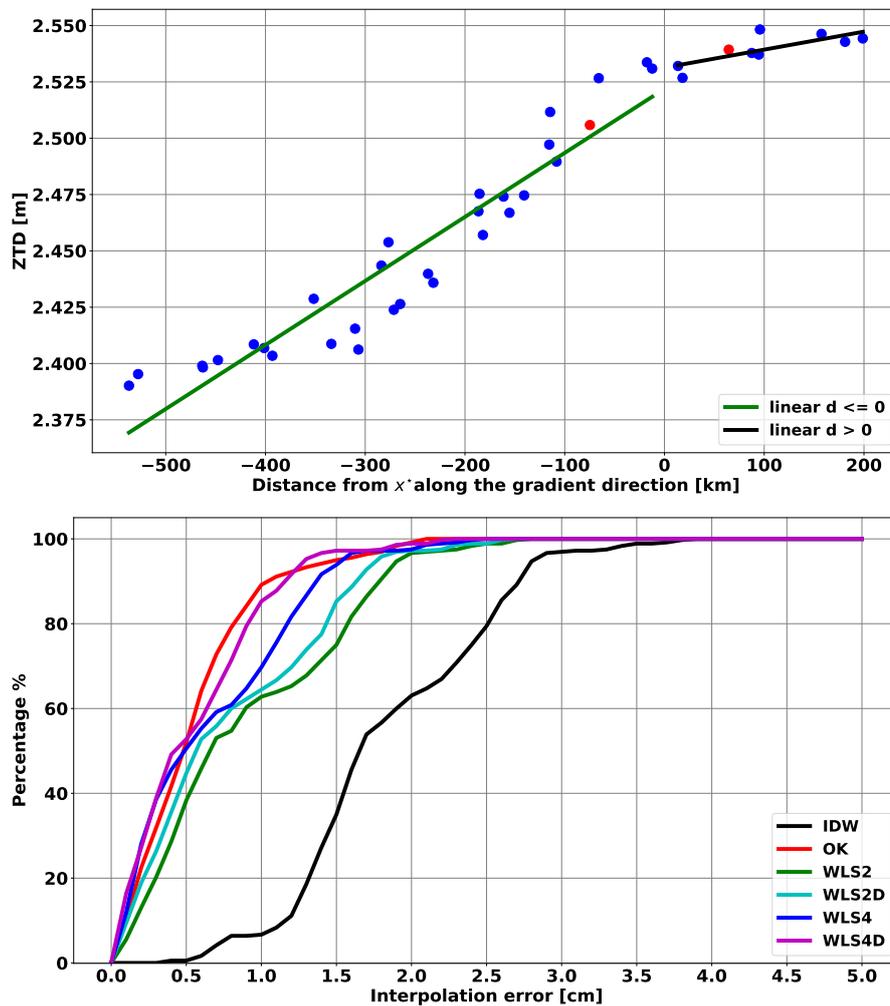
**Figure 5.20:** Cumulative interpolation error obtained using the different interpolation techniques for the ZTD data. From top to bottom: NETPOS data-set during hour 17 of DOY 173 (left) and DOY 177 (right), LGLN data-set during hour 9 of DOY 275 (left) and hour 8 of DOY 278 (right), showing larger errors during higher ZTD variations (left).

#### 5.5.4 Impact of test location

The results presented above are significant because they isolate the influence of the different interpolation techniques. However, for the sake of completeness, it has to be mentioned that the presence of the test-station observations in the network computation influences the estimation of the SSR parameters. Therefore, the relevance of this influence should be investigated. Four and three stations of the LGLN and NETPOS network respectively have been tested for the interpolation analysis. The stations chosen within the LGLN network are situ-

ated in Oldenburg (53.14° N, 8.20° E), Rotenburg (53.11° N, 9.40° E), Hannover (52.41° N, 9.8° E) and Verden (52.91° N, 9.24° E), while those of the Kadaster network are located in Utrecht (52.06° N, 5.09° E), Willemstad (51.68° N, 4.40° E) and Leeuwarden (53.20° N, 5.79° E). Again, the reference ZTD values are those generated by network processing of the complete network for the whole time-span.

Now, for the interpolation analysis, the respective test station has been taken out of the network processing, resulting in slightly changed ZTD values for the other stations. These are then used to interpolate at the location of the test station, and the interpolation result is compared to the reference ZTD generated for this station in the initial network processing. In this way, any influence of the test station observations on the interpolation performance has been excluded. However, although the estimated ZTDs are slightly different in values, the main conclusions in terms of cumulative error and RMSE remain the same as mentioned above for the complete network. Nevertheless, a more detailed analysis can be carried out.



**Figure 5.21:** Top: ZTD distribution in the gradient direction during hour 9 at DOY 275. The ZTD reported is an average value over one hour of data. The direction of the gradient is considered as the mean over the whole network. The distribution is centered at  $x^*$  (i.e. station 0688 in Hannover). A green (black)-colored shows the linear variation for negative (positive) values of the x-axis. The two red points indicate two stations at a similar distance before and after the point to interpolate. Bottom: cumulative interpolation error for the location of the Hannover station 0688.

The top panel of Fig. 5.21 shows the ZTD distribution over the LGLN network during hour 9 of DOY 275 along a mean gradient direction. The latter is allowed because the gradient direction is quite homogeneous over the network, as depicted in Fig. 5.19.

The distribution reported in Fig. 5.21 has been centered for the Hannover location. It can be seen that there is a change in the ZTD variation before and after the interpolation point. A green-colored line indicates the linear variation before the point to interpolate and a black-colored line after the query point in the direction of the horizontal gradient. The linear fit has been computed using a LS approach. The behavior could be recollected to something similar to what described by the step-linear variation of Eq. 5.20. Moreover, Fig. 5.21 (top panel) highlights two red-colored points. The two points indicate two stations close to the point to interpolate at a similar distance. In this case, a WLS2 (or WLS4) approach would assign an almost identical weight to the two stations, even though the ZTD of the stations differ by roughly 4 cm. The bottom panel of Fig. 5.21 suggests that the use of  $\tilde{d}$  instead of  $d$  could reduce the number of large errors. Furthermore, in that case, additional weight to all the closest stations (i.e.  $p = 4$  instead of  $p = 2$ ) would improve the technique in terms of the error distribution. These results are in agreement with the scenario depicted in Fig. 5.20 where OK and WLS4D were the most recommended techniques to reduce the percentage of large errors (e.g. larger than 1 cm). For example, the use of WLS4D instead of WLS would reduce the number of errors larger than 1 cm by an amount of 20% of the total cases. Here, it is worth mentioning the rule of thumb introduced above. In fact, following the latter, to mitigate ZTD interpolation errors larger than 1 cm means to reduce the number of height errors larger than 3 cm in the positioning domain.

Finally, with the limitation caused by the realistic nature of the data, it can be stated that the simulations proposed in Section 5.4 provide indications that are in agreement with the results obtained with real ZTD data-sets during specific severe weather events.

## 5.6 Interpolation of residual ionospheric effects during a TID

### 5.6.1 Data-set and extraction of TID event

In this section, the interpolation quality of SSR ionospheric parameters estimated during a period with a disturbed ionosphere is investigated. This specific analysis has been carried out in collaboration with the University of Bath in the framework of the TREASURE project. As sample day, May 14th 2019, DOY 134, was chosen. Reference SSR TEC values have been generated for all stations by network processing of the GEONET measurements for the whole day. RINEX 3 input data were considered, and a multi-constellation and multi-frequency computation was performed. Observations from GPS, Galileo, GLONASS, and QZSS satellites were processed, taking into account L1, L2, and L5 frequencies. The network processing employs an uncombined observation model.

During DOY 134, one particularly clear TID was detected between 9:30-10:30, and this period was selected for further testing. The TID characteristics were estimated from TEC using a method based on the work of Hernández-Pajares et al. (2006), where the cross-correlation of de-trended TEC series is used to set up a system of equations that can be solved for estimates of the slowness vector  $\mathbf{s} = \mathbf{v}^{-1}$  using the relation

$$(\Delta \mathbf{r}_{\text{IPP}} + \mathbf{v}_{\text{IPP}} \Delta t) \mathbf{s} = \Delta t. \quad (5.25)$$

In this equation,  $\mathbf{r}_{\text{IPP}}$  and  $\mathbf{v}_{\text{IPP}}$  are the location and velocity of the IPP (see Section 2.1.3) at an altitude of 300 km, while  $\Delta t$  is the time lag of the wave signature for the same satellite and different receivers. A system of these equations can be solved for  $\mathbf{s}$  using a LS approach. After estimating the wave period  $T$  by considering the Doppler shift of the apparent period  $T'$  using

$$T = T'(1 - \mathbf{s} \cdot \mathbf{v}_{\text{IPP}}), \quad (5.26)$$

the horizontal wavelength can be determined as  $\lambda_h = v/T$ . Only TEC series with amplitude above 0.1 TECU within the interval  $T' \in [10 \text{ min}, 60 \text{ min}]$  from satellites with elevation above  $50^\circ$  were used in this analysis to avoid correlating ionospheric features that are not related to MSTIDs.

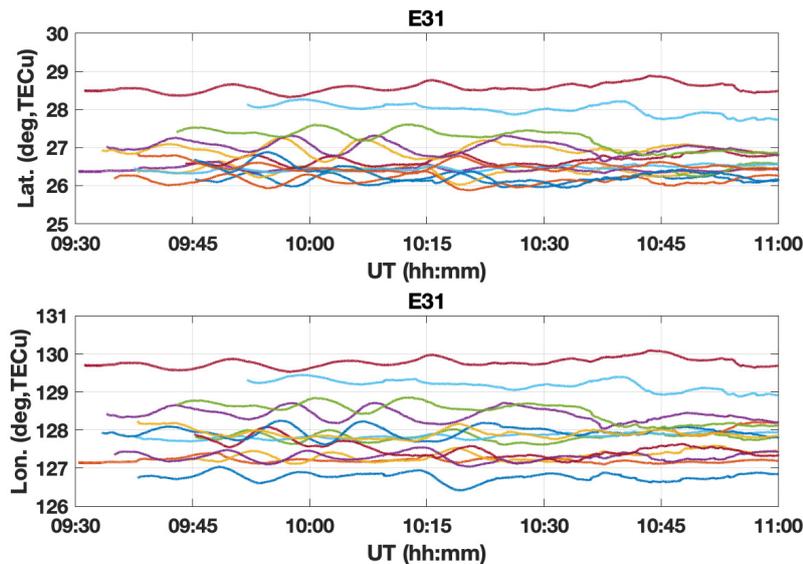
While Hernández-Pajares et al. (2006) de-trended each TEC time series individually using a type of band-pass filtering, in this work, a spatial de-trending is implemented by subtracting the ionospheric models estimated in the SSR procedure from the observed TEC using the reference network. In this case, the model consists of two parts - a global model (GVI, see Section 2.2) and a satellite-dependent regional model (RSI, see Section 2.2). The global model is a spherical harmonic expansion, and this model is further adjusted using a local, Chebyshev polynomial expansion (see Section 2.2). The resulting TEC is the GRI SSR parameter shown in Fig. 5.22.

In order to provide a direction estimate uncertainty, the procedure was performed repeatedly within a moving one-hour time window, shifted by 10 seconds for every new estimate, and the standard deviation and mean of these were selected. The estimated parameters for the TID between 09:30-10:30 UTC are:

$$\begin{aligned}\theta_{TID} &\approx 22^\circ \text{ (N-E)}, \\ \sigma_{\theta, TID} &\approx 7.5^\circ, \\ \lambda_{TID} &\approx 140 \text{ km}, \\ T_{TID} &\approx 20 \text{ min}, \\ A_{TID} &\approx 0.25 \text{ TECU}.\end{aligned}$$

Based on the outcome of Section 5.4, considering this wavelength and average baseline of the GEONET sub-network, it should be possible to reduce interpolation errors by using the modified WLS interpolation methods.

Due to the geometry of the wave and satellite movement, some satellites experience more pronounced TEC perturbations. Example de-trended TEC time series from different receivers are plotted in Figure 5.22. In this figure, each line represents the TEC perturbation around zero in TEC units, with each series centered around the location of the receiver in latitude (top) or longitude (bottom). This has been done to better visualize the movement of the wave.



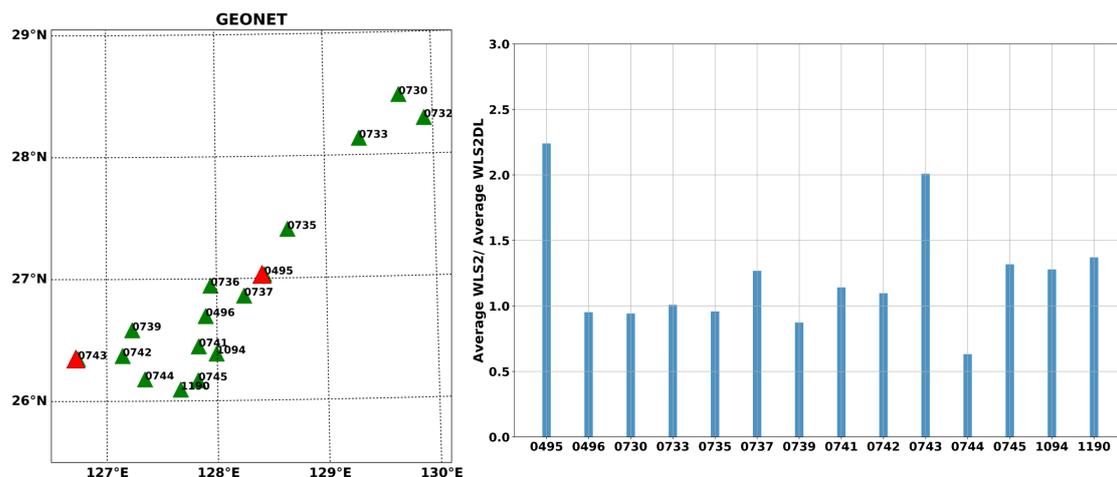
**Figure 5.22:** TEC values of the ionospheric grid residual SSR parameter (the GRI component, see Section 2.2) for satellite E31 during the TID interval. A sinusoidal variation is observed along with a propagation in the N-E direction.

### 5.6.2 Improved interpolation strategy

Each station location has been tested as interpolation query point  $\mathbf{x}^*$ . Figure 5.23 shows the improvements gained by applying both  $\tilde{d}$  and  $\tilde{p}$  (WLS2DL) over using only WLS2 for different query locations in the network. Each bar above 1 representing an improvement for the location of that station in the grid. The largest improvements, i.e. roughly a factor of 2, are found in receivers *0495* and *0743* (highlighted in red-colored triangles in Fig. 5.23). Overall, in 64% of the tested locations WLS2DL could reduce the average interpolation error.

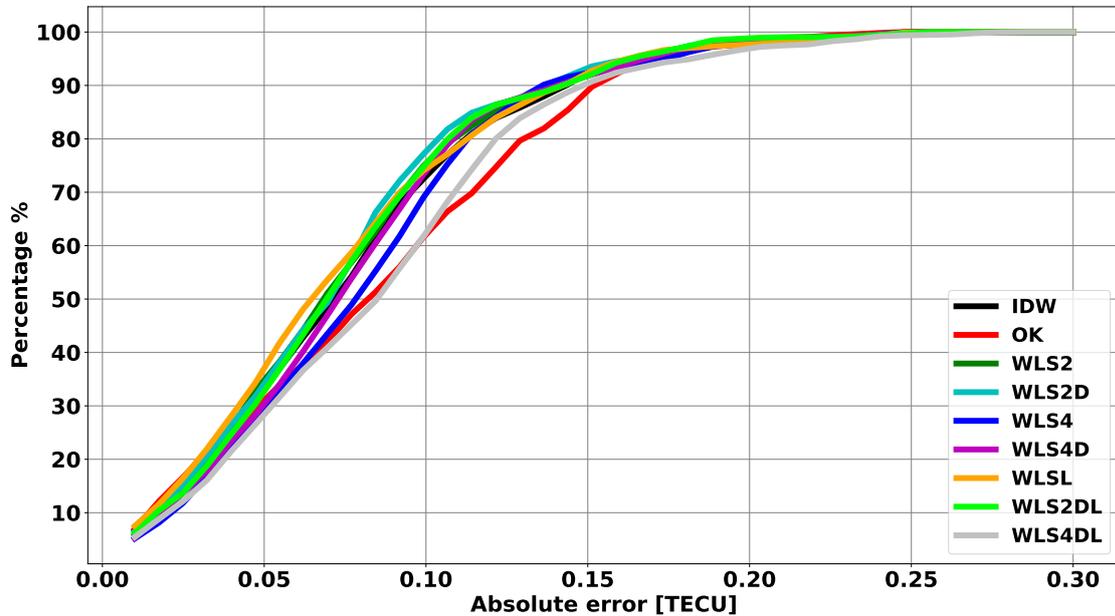
As in the tropospheric case, the presence of the test station observations in the network computation influences the estimated SSR parameters. Although for the ZTD this influence did not impact the general outcome of the investigation, it is worth analyzing it for the ionosphere as well. Therefore, the network processing has been repeated taking out one test station, resulting in slightly changed TEC values for the other stations. These are then used to interpolate at the location of the test station, and the interpolation result is compared to the reference SSR TEC generated for this station in the initial network processing. In this way, any influence of the test station observations on the interpolation performance has been excluded. The test station considered is station *0741* with approximate latitude  $26.45^\circ$  and longitude  $127.83^\circ$  (see left panel of Fig. 5.23). Additionally, *0741* is the closest reference location to the query positioning investigated in the simulations (see Section 5.4 and Fig. 5.3). Considering this test, Figure 5.24 shows the cumulative interpolation error for all interpolation methods tested for three Galileo satellites affected by the TID, namely E01, E13, and E31. It is worth observing that E01 and E31 show an elevation larger than  $50^\circ$ , while E31's elevation does not exceed  $20^\circ$ .

Some improvement can be seen from making use of the direction of the TID by using  $\tilde{d}$  (WLS2D vs. WLS2, WLS4D vs. WLS4). Adjusting the power by the estimated wavelength (WLSL) has similar errors as the power of 2 WLS interpolation (WLS2), and performs slightly better than WLS4. In this case, there appears to be little benefit from applying both techniques (WLS2DL vs. WLS2D) compared to only using the direction, and applying both  $\tilde{d}$  and  $\tilde{p}$  techniques with a power of 4 (WLS4DL) shows larger errors. OK displays the largest errors in this case. The differences become clearer when only considering the low elevation satellite E31, as shown in Fig. 5.25.

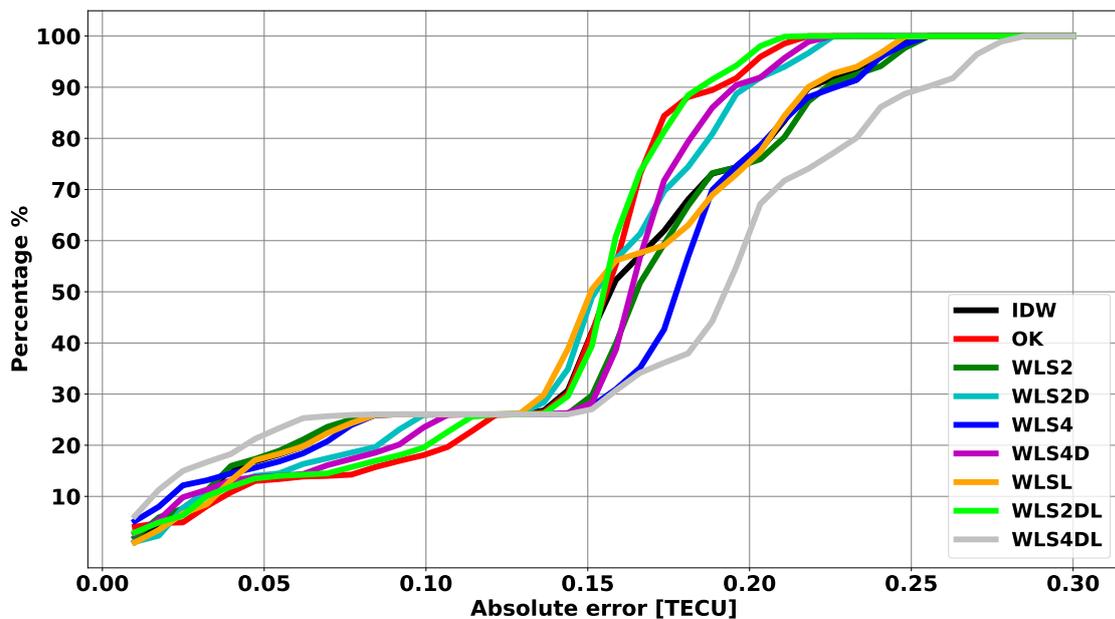


**Figure 5.23:** Comparison between average WLS2 and WLS2DL interpolation error (right) for each receiver location of the Okinawa GEONET network (left). The largest improvement factor is observed for stations *0495* and *0743*, which are highlighted in red-colored triangles.

The pattern of the results shown in Fig. 5.25 is similar to that in Fig. 5.24, but with an improved performance of OK relative to the other methods. It appears that, in this test, the use of a stronger power increases the error (e.g. WLS4 vs. WLS2), contrary to what expected from the simulation results obtained in Section 5.4. However, a clear reduction of the number of large errors can be achieved by employing  $\tilde{d}$  instead of  $d$ . For example, the number of errors lower than 0.2 TECU changes from 75% with WLS2 to 90% with WLS2D.



**Figure 5.24:** Cumulative interpolation error for the E01, E13, E31 satellites. The location considered for the interpolation is the position of station *0741* of the Okinawa GEONET network (see left panel of Fig. 5.23).



**Figure 5.25:** Cumulative error for satellite E31 (elevation lower than  $20^\circ$ ). The location considered for the interpolation is the position of station *0741* of the Okinawa GEONET network (see left panel of Fig. 5.23).

At this point, it might be useful to keep in mind that a 0.2 TECU error would correspond to roughly 3.2 cm for L1 observations. A further improvement of 6% can be achieved by replacing  $p$  with  $\tilde{p}$  (see Fig. 5.25). In addition, for this particular case, Fig. 5.25 suggests the impact of using  $\tilde{p}$  only, without adding any factor, i.e. the WLSL approach, is the same of using  $p = 2$ , i.e. WLS2.

The results in this section show that TID patterns can be visualized in ionospheric SSR residuals estimated via GNSS-based network processing, and that they affect the interpolation results. Furthermore, the interpolation error can be mitigated for particularly perturbed satellites, reducing the number of large errors by taking advantage of estimated TID parameters like wavelength and direction of propagation. It is also evident that the simulations carried out in Section 5.4 provide valuable indications about when to apply either  $\tilde{d}$  or  $\tilde{p}$ , but not the use of a power  $p = 4$  vs  $p = 2$ . The latter discrepancy between the real and simulated results might be due to the actual variation of the ionosphere, which, naturally, is more complicated than the simple models used in Section 5.4.

## 5.7 Error propagation: from SSR modeling to the user

As mentioned in the previous sections of this chapter, the ionospheric delay is strongly influencing the ambiguity resolution process. In this section, the impact of an error in the SSR ionospheric parameters on user performance is assessed. The investigation is carried out, taking advantage of the multi-stage representation employed by the SSRZ format (see Chapter 2, Section 2.2). In section 5.6, it has been demonstrated that truthful information about TID can be extracted from the SSR ionospheric grid residuals. As a consequence, in this section, the aim is to analyze the impact of a TID in the positioning domain. Firstly, some simulated data is used to analyze the expected impact of an ionospheric bias on rover performance in terms of the percentage of fixed ambiguities. Secondly, a real-time test is carried out by injecting a TID-shaped error in the SSRZ-corrections stream.

### 5.7.1 Ambiguity resolution with ionospheric biases: simulation

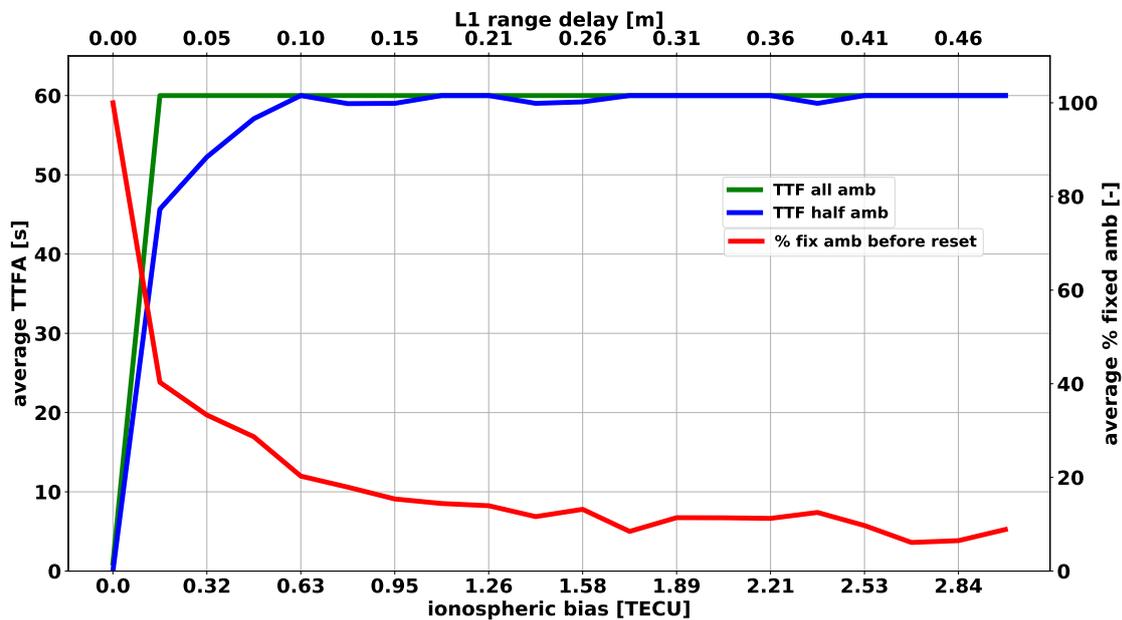
To investigate the impact of an ionospheric bias, a simulation has been performed considering reconstructed observations for a user. The latter is assumed to receive corrections computed from a network of reference stations. For the sake of simplicity, GPS only observations are considered. L1, L2 and L5 frequencies have been taken into account. GPS almanac data have been used to retrieve the satellite geometry over a specific location (i.e. Garbsen, Germany, lat=52.43074°, lon=9.6083 °) on a specific day, i.e. March, 16th 2019. The simulation follows the principles introduced in Wübbena (2007). A combination of geometry free and geometry-based algorithm with an ambiguity search method is employed to solve ambiguities (e.g. Seeber, 2003; Wübbena, 2007). No residual orbit errors have been considered, assuming precise orbits from the network. In a similar way, no tropospheric residual errors have been considered to highlight the impact of the ionosphere. A simulation of white noise for observations has been carried out considering Gauss-Markov processes for the multipath for all individual signals, with variations of 2 mm/60 s and 0.2 m/60 s for phase and code, respectively. A Kalman filter has been applied to estimate coordinates, ambiguities, ZTD, and ionospheric delay.

Here, the ionospheric bias is considered as residual from a network computation. Bias values varying between 0 and 3 TECU have been considered. As explained by the work of previous authors (e.g. Wübbena, 2007; Li et al., 2014), the magnitude of the residual ionospheric error can be associated to the inter-station distance of the network. For example, an error of 0.5 m can be associated to an inter-station distance of 500 km, while an error 0.05 can represent a distance of 100 km.

Fig. 5.26 shows the variation of the TTFA versus the residual ionospheric bias introduced in the simulation. The TTFA is reported considering the total (green-colored line) and half (blue-

colored line) number of ambiguities, respectively. A complete reset of the ambiguity resolution process is performed every 60 s to better simulate a real-time scenario. Fig. 5.26 reports average values over the 60 s interval time during the one-day simulation. It can be observed that for values larger than 0.2 TECU it is no more possible to fix all ambiguities. Furthermore, for biases larger than 0.3 TECU (i.e. roughly 5 cm of range delay for the L1 frequency), Fig. 5.26 indicates that resolving half of the ambiguities is very challenging. In addition to the TTFA, Fig. 5.26 depicts the percentage of fixed ambiguities before the reset averaged over the whole period. The figure suggests clearly that the percentage decreases while the bias increases.

Comparing Fig. 5.26 to the results reported in Section 5.6, it can be stated that the interpolation error in the presence of a TID affects the percentage of fixed ambiguities and the TTFA. This is true especially for low elevation satellites, e.g. E31 in Fig. 5.25, where 20% of the errors are larger than 0.2 TECU using, for example, the WLS2 technique. In that case, an improvement can be expected in the ambiguity resolution when one of the proposed methods (e.g. WLS2DL) is used.



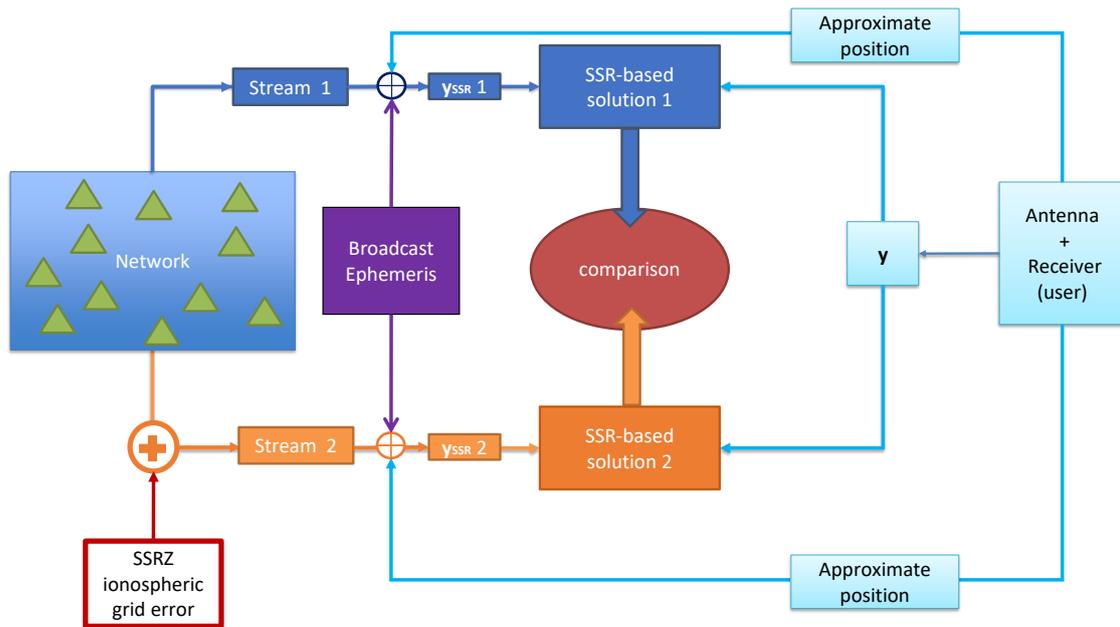
**Figure 5.26:** Simulation of ambiguity resolution performance in presence of ionospheric biases. A rover positioning simulation covering one day with a reset every 60 s has been performed. The average TTFA is reported for fixing all ambiguities (green-colored line) and half of the total amount (blue-colored line). The average percentage of fixed ambiguities before a reset is indicated by a red-colored line.

### 5.7.2 Real-time impact on positioning of an ionospheric SSR error

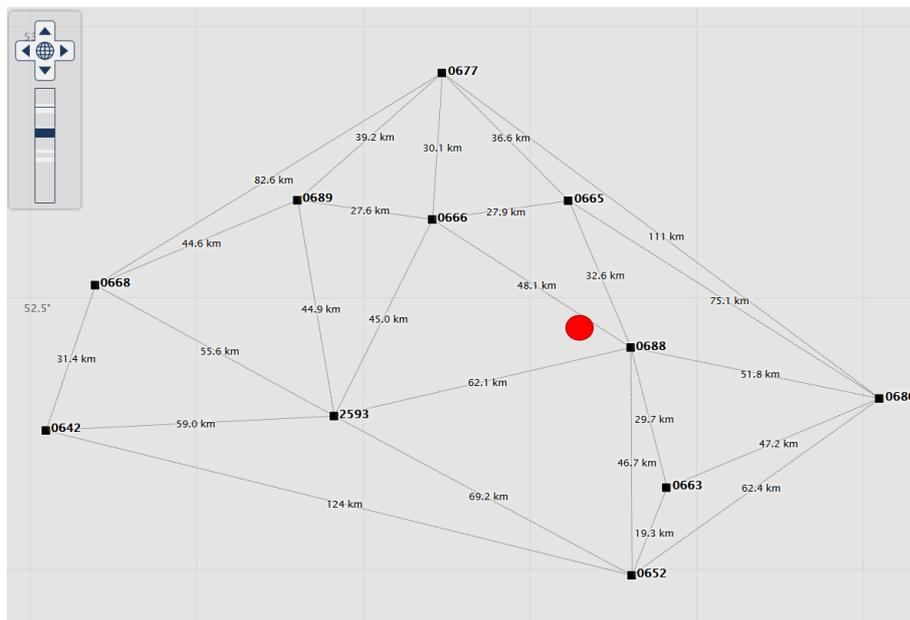
#### Setup

Fig. 5.27 shows the setup used to evaluate the impact on the positioning domain of an error in the ionospheric SSR parameter. The observations of a network of GNSS reference stations are used to generate SSR corrections that are transmitted using the SSRZ format (Geo++ GmbH, 2020). Two streams of corrections are broadcast. One of the two is perturbed by adding an error to the grid ionosphere SSRZ message (see Section 2.2) for all satellites. It is fundamental to remark that the two SSR streams are identical except the injected ionospheric error. The error is modeled as a sinusoidal variation, as described by Eq. 5.22.

The idea is to simulate the impact of a TID. In this case, the amplitude considered is 0.3 TECU with a period of 20 min and a wavelength of 140 km. Hence, very similar to the estimated



**Figure 5.27:** Setup for the evaluation of the impact of ionospheric errors in the SSR messages. A network of GNSS reference stations is used to compute SSR corrections transmitted using the SSRZ format. Two streams of SSRZ corrections are generated. One of the two (*Stream 2*) is perturbed by the injection of a modeled error affecting the SSRZ ionospheric grid message. Using the SSRZ parameters, non-physical GNSS station data generated. The SSR-based solutions with and without error injection are compared.



**Figure 5.28:** LGLN sub-net used in the SSRZ error injection test described in Fig. 5.27. All the stations are equipped with Septentrio PolaRx (4 and 5) receivers. All the antennas involved are Leica antenna (LEIAR25.R4\_\_\_LEIT), but for station 2593, which employs a Trimble antenna (TRM55971.00\_\_\_TZGD). The red-colored dot indicates the user's location considered in the experiment. The inter-station distance is reported in km.

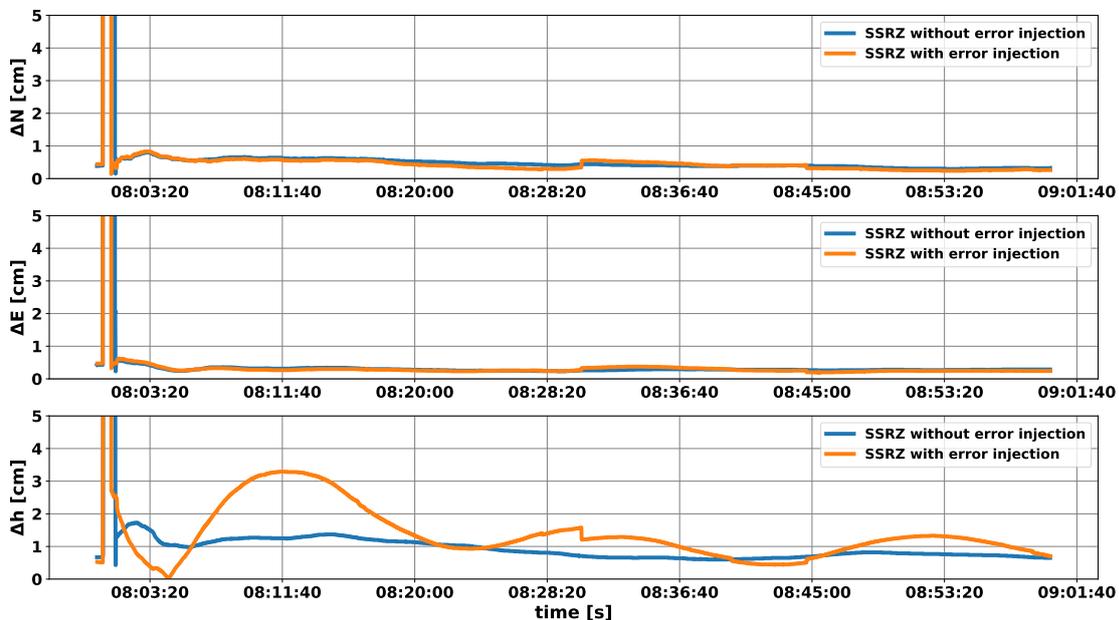
parameters of the TID occurred in the Okinawa data-set described in Section 5.6. The TID is assumed to propagate from East to West.

The network used for the generation of the SSR corrections is a sub-net of the LGLN-SAPO network, as shown in Fig. 5.28. It is composed of eleven stations around Hannover, Germany. All the stations are equipped with Septentrio PolaRx (4 and 5) receivers. All the antennas involved are Leica antenna (LEIAR25.R4\_\_\_\_LEIT), except for station 2593, which employs a Trimble antenna (TRM55971.00\_\_\_\_TZGD). The GNSS network-processing is performed utilizing multi-constellation (GPS, GLONASS, and Galileo) and multi-frequency (L1, L2, and L5) observations. The user's location is highlighted by a red-colored dot in Fig. 5.28. The location coincides with the Geo++'s building in Garbsen.

The SSR messages (i.e. *Stream 1* and *Stream 2*) along with broadcast ephemeris and an approximate position provided by the geodetic-grade receiver of the user are employed to generate the non-physical GNSS station data as described in Section 2.2, i.e.  $\mathbf{y}_{SSR\ 1}$  and  $\mathbf{y}_{SSR\ 2}$  (with the injected error). The interpolation technique used to interpolate the atmospheric SSR parameters for the non-physical observations is WLS2. Non-physical and user's observations are processed as described in Section 2.4. For the sake of simplicity, GPS-only and multi frequency (L1, L2, and L5) observations are considered for the rover processing. SSR-based non-physical GNSS observations ( $\mathbf{y}_{SSR\ 1}$  and  $\mathbf{y}_{SSR\ 2}$ ) are processed with the physical user's observations ( $\mathbf{y}$ ) using an uncombined observation model (as described in Section 2.4). The resulting SSR-based solutions, i.e. *SSR-based solution 1* (without injected error) and *SSR-based solution 2* (with injected error) are compared. Furthermore, atmospheric parameters are not estimated by the user assuming to receive accurate corrections (and that the non-physical station and user observe the same atmosphere) to evaluate the impact of the injected error properly.

## Results

Several hours have been tested, showing similar performance. As an example, the experiment carried out during hour eight of DOY 149 of the year 2020 is reported. Fig. 5.29 shows the error in the North, East, and height components of the positioning results w.r.t. known coordinates.



**Figure 5.29:** SSR-based positioning error w.r.t. known coordinates. The error obtained using the SSRZ corrections without the error injection (blue-colored line) is compared to the resulting error when using the SSRZ stream perturbed with the error injection (orange-colored line).

The error obtained using the SSRZ corrections without the error injection (blue-colored line), i.e. *Stream 1* in Fig. 5.27, is compared to the resulting error when using the SSRZ stream perturbed with the error injection (orange-colored line), i.e. *Stream 2* in Fig. 5.27.

On the one hand, results indicate that the SSR mismodeling due to the injection of artificial TID-shaped error does not impact the solution in the horizontal positioning components (i.e. North and East). On the other hand, the mismodeling has a significant impact on the height component. The outcome is expected as in agreement with the work of previous authors (e.g. Jin et al., 2005; Sieradzki and Paziewski, 2016). Larger error values are observed with a fluctuation of a few centimeters. The reliability of the ambiguity-fixed solution is reduced, and the convergence time increases.

It is worth mentioning that the injected error in the SSRZ stream resulting in the interpolated value for the non-physical observations could be read as interpolation error only. In the case of the error injection, the interpolated value affecting the positioning has maximum values lower than the amplitude of the oscillation, i.e. lower than 0.3 TECU. This is due to the propagation time considered in the oscillation construction. In Section 5.6, it has been shown that interpolation errors up to 0.25 TECU can occur in a real-case scenario. Hence, the results shown in this section underline the importance of enhancing the interpolation strategy to reduce the positioning error and improve the ambiguity resolution's reliability.

## 5.8 Discussion

### Summary

The importance of having an accurate atmosphere for SSR-based positioning makes the interpolation of network-based SSR parameters essential for techniques such as PPP-RTK and N-RTK. Accordingly, the interpolation error of atmospheric SSR parameters and its impact on the positioning have been investigated. In particular, the ZTD and the grid mapped VTEC (the SSRZ GRI parameter) have been considered as atmospheric parameters for the analysis. It has been shown that the variation of GNSS-estimated ZTDs is in agreement with the variation of NWM-based ZTDs. Furthermore, it has been proven that ground-based weather data cannot always provide reliable information about the tropospheric delay. Concerning the ionosphere, it has been demonstrated that TID parameters (e.g. wavelength and direction) can be detected by analyzing the GNSS-estimated grid mapped VTEC.

To mitigate the interpolation errors, new methods to interpolate SSR atmospheric parameters have been introduced. The approach is based on the WLS concept, where the weight takes into account the physical aspects of atmospheric parameters. Two main concepts are introduced. The first idea is to de-weight stations in the direction of atmospheric variation. The second concept takes into account the ratio between the spatial scales of network and atmospheric perturbation. Modified WLS techniques have been compared to IDW and OK methods.

Artificial and homogeneous networks together with real network geometries were used to simulate the interpolation performance in terms of interpolation error. Different functional variations were used to simulate the atmospheric behavior over the network: constant with step, linear with step, quadratic and sinusoidal. The simulations show that a directional WLS method can improve the quality of the interpolation. The benefit of using the directional WLS depends on the relative spatial scale between network geometry and propagation of the atmospheric events (e.g. the average baseline of the network w.r.t. the wavelength of the sinusoidal or the maximum variation of the atmospheric parameter). In addition, the simulation indicates that, in the case of quadratic variation, the OK technique is the method that assures the smallest errors.

As a verification, the techniques have been tested using real SSR data from network pro-

cessing of GNSS reference station observations. ZTD and residual ionospheric TEC have been considered to evaluate the interpolation methods. ZTDs during two severe weather events occurred in 2017 in the Netherlands (NETPOS network), and Germany (LGLN network) have been analyzed. During these events, clear tropospheric gradients were observed. Using NWM tropospheric horizontal gradients to define the direction of ZTD variation, the benefit of the directional WLS during particularly perturbed hours has been demonstrated. The interpolation has been evaluated considering all the stations as a single rover. Reduced RMSEs were observed between 53% and 80% of the cases. Furthermore, specific cases where a change in the ZTD variation has been observed were isolated. Improvements in the distribution of the interpolation error have been demonstrated, e.g. 20% reduction of the number of errors larger than 1 cm, i.e. roughly 3 cm positioning error in the height component.

Regarding the ionosphere, a test over a sub-network of GEONET in the Japanese prefecture of Okinawa showed that the reduced interpolation errors seen in the simulation results could be achieved in a real TID scenario. In these particular tests, the smallest errors were reached by adjusting the WLS with the estimated direction and wavelength of the TID, with improvement over unmodified WLS2 as large as a factor of 2 in some cases. A particular case scenario was analyzed by taking out a test station from the network processing and using it as a query location for the interpolation. In this test, an improvement was achieved in mitigating the number of large errors, i.e. larger than 0.2 TECU (roughly equivalent to 3 cm for the L1 frequency). The employment of adjusted WLS2 methods reduces the number of errors by 15%. This particular improvement was observed for a significantly perturbed low elevation satellite.

To conclude, the methods introduced in this chapter have the potential benefit to mitigate the interpolation error of atmospheric SSR parameters, and thus possibly allow more robust ambiguity resolution in high-accuracy and precision positioning. The analysis indicates that sharp changes in tropospheric and ionospheric parameters can occur during perturbed periods, and significant errors can be mitigated by taking advantage of information on the atmospheric behavior. Further analyses of larger data-sets are needed to more narrowly define the conditions in which to use the methods proposed in this paper. However, simulation-based guidelines have been successfully validated with real data-sets, providing first indications about when to use the interpolation methods.

In addition to the interpolation analysis, the propagation of an error in the SSR ionospheric parameter estimation to the user has been investigated. The experiment takes advantage of the multi-stage ionosphere representation employed by the SSRZ format. It has been demonstrated that the mismodeling of a TID can affect users positioning significantly. In particular, the mismodeling causes a strong impact on the height component. The SSR error can also be seen as an interpolation error only with similar error magnitudes. The outcome highlights the essential role of the interpolation of atmospheric parameters in SSR-based positioning.

### **Potential application of the proposed interpolation methods**

New approaches to interpolate atmospheric SSR parameters have been presented, and the benefit of using information related to the physical processes of the atmosphere has been shown. As a consequence, here, it is proposed how this information might be transmitted to the user. Table 5.4 summarizes the content the proposed messages could provide for tropospheric and ionospheric interpolation.

The tropospheric message would consist of data derived from NWM products. In this context, the SSR parameters provider could retrieve the NWM information from near-real-time or prediction databases (e.g. GFS, HARMONIE (Bengtsson et al., 2017)) and provide tropospheric horizontal gradients in North and East direction ( $G_N$  and  $G_E$  in Table 5.4). The data would be provided in a grid format (e.g. latitude and longitude), with the spatial resolution varying depending on the NWM chosen (e.g. 0.25 deg like in this work). The recommended update rate would be a low rate of one hour. In this way, the user could retrieve the tropospheric gradient

for their location by a simple bi-linear interpolation of the grid data. Considering the NWM gradient, along with the SSR messages that have been transmitted, the GNSS user could implement the interpolation method introduced in this work. Firstly, the magnitude of the gradient would give an indication of the benefit of using the carried information. It has been tested that magnitudes greater than 2 mm relate to a disturbed troposphere. Nevertheless, a deeper investigation should be carried out to better assess the suggested threshold. Secondly, the ratio between the maximum ZTD change along the gradient direction and the average baseline could be computed, giving a further indication on the possible benefit of using  $\tilde{d}$  instead of  $d$ , as introduced in the simulations (see Section 5.4).

Concerning the ionospheric message, estimated TID parameters would be transmitted as summarized in Table 5.4. The parameters considered are the following: the amplitude  $A$ , the wavelength  $\lambda$ , and the direction of propagation through the angle  $\theta$  measured clockwise from North. Along with the direction, the estimated precision of the estimation could be provided, i.e.  $\sigma_\theta$ . An analysis of the dependency on  $\sigma_\theta$  has been shown in Section 5.4, and the user can hence decide when to use the proposed interpolation. Furthermore, taking into account the network baseline (through SSR data) and the estimated wavelength, the user can implement the modified power  $\tilde{p}$ . Again, the Section 5.4 analyses whether the user might benefit from the proposed approach or not. Considering that the TID parameter estimation procedure in Section 5.6 assumes the TID to display a roughly coherent wave behavior within the 1 h window, we suggest a half-hour update rate for the ionospheric message. With an assumed 1 h validity of the TID parameters, the central time of the estimation time window would be at most 1 h in the past before an update is received. As an added benefit of including the amplitude  $A$  in the message is that it may be used as an estimate of the initial standard deviation if the user positioning algorithm estimates a residual ionosphere. Finally, the TID message could also report the particular satellites affected by the TID.

**Table 5.4:** Content of proposed messages to transmit to the user to improve the interpolation of atmospheric SSR parameters during strong weather fluctuations or TID events.

Name	Parameters	Type of message	Data rate
Tropospheric NWM	$G_N, G_E$	Grid	1 hour
TID	$A,$ $\lambda,$ $\theta + \sigma_\theta$	satellite dependency	30 min



## 6. Conclusions and outlook

---

The work carried out in this thesis is based on the state-of-the-art SSR concept used in GNSS-based positioning (see Chapter 2). Information about the formats available to transmit SSR corrections has been reported (see Chapter 2). In particular, the RTCM and SSRZ formats have been described together with the application of SSR corrections at the user's side. Furthermore, a demonstrator tool to decode the SSR messages has been developed and made available online.

The first main scientific question (see Chapter 1) about the potential of smartphone-based positioning has been addressed. A method to construct the observations is proposed along with an approach to deal with the continuity of phase measurements. The quality of pseudorange and phase measurements was analyzed in detail using DD and multipath combinations (see Chapter 3) and the positioning results using GNSS smartphone data were presented (see Chapter 3 and 4). The data-sets collected in four scenarios with different antennas and levels of multipath were tested. The positioning algorithm used is based on SSM and employs an uncombined observation model.

The tests demonstrated that in a zero-baseline configuration making use of an RF enclosure, AR can be achieved obtaining a 2D RMSE lower than 1 cm. A fixed solution cannot be achieved in a short-baseline (lower than roughly 50 m) configuration without applying the PCC.

The Geo++ absolute robot-based field calibration of GNSS antennas was used to determine PCC for the Mate20X smartphone. The calibration was evaluated over twelve different runs, showing repeatability with elevation-dependent PCV differences lower than 4 mm and 12 mm for L1 and L5, respectively. The absolute PCV reached up to 4 cm with a formal STD that did not exceed 1.6 mm.

Different multipath environments were tested. High-accuracy positioning using the Mate20X and a geodetic receiver was performed. Using the PCC, a 2D RMSE smaller than 2 cm was achieved without an external antenna, as described in Scenarios 3 and 4. The repeatability of the results obtained applying the PCV was tested using 35 data-sets of one hour gathered in open sky conditions on the Geo++ rooftop. In 54% of the cases (i.e. 19 samples), a successful AR was achieved. The AR feasibility in the collected data-sets is related to a good compromise between the quality of the measurements (e.g. C/N0, multipath, cycle-slips) and geometry of the satellite constellation. A 2D RMSE lower than 2 cm was obtained in less than three minutes, correcting for the PCV and PCO. Nevertheless, even without PCC, a float solution with a 2D RMSE lower than 50 cm was achieved. Furthermore, reducing the multipath impact, i.e. placing the smartphone over a choke ring, the 2D RMSE can be reduced to 10 cm.

In addition, an alternative approach can be used to separate the acting error components. To further reduce the effect of multipath, an experiment where the smartphone moved on a toy train track was set up. The smartphone was lying over a choke ring carried by the toy train over the track. Twenty minutes of data were collected and analyzed in post-processing, with a float-ambiguity forward-backward Kalman filter approach. The agreement between the smartphone-based and geodetic receiver-based solutions was always better than 10 cm.

Finally, VRS observations were generated from SSR data determined from a GNSS reference station network and applied by using GNSS-based positioning. The LGLN-SAPOS network in Lower Saxony was used for the investigation. Smartphone measurements collected within an RF enclosure fed by a geodetic-grade antenna were considered. Although a stronger impact of distance-dependent error sources, e.g. ionospheric effect, was present compared to

---

the short-baseline case, ambiguity resolution was possible with a TTFA lower than 30 s. After fixing ambiguities, a 2D RMS of 3 mm is achieved.

The inclusion of GNSS antenna information (i.e. PCO and PCV) in the Android raw measurements starting with Android 11 (Van Diggelen et al., 2020) supports the calibration of new devices to standardize the use of PCC in smartphone-based positioning. The next step for further research is developing a sensor fusion algorithm to take care of the smartphone's actual attitude when applying antenna corrections. The use of such a sensor fusion technique could unveil several new applications making use of high-accuracy smartphone-based positioning.

Exploiting the SSRZ features for atmospheric modeling, the interpolation of SSR atmospheric corrections has been investigated (see Chapter 5). Multiple interpolation techniques were analyzed to mitigate interpolation errors of SSR atmospheric corrections during periods with a perturbed atmosphere. Novel methods have been proposed, exploiting the use of external atmospheric models, showing improvements up to 3 cm during significantly perturbed periods. Simulation-based guidelines have been successfully validated with real data-sets, providing first indications about when to use the interpolation methods. The validation of these guidelines to more narrowly define the conditions in which to use the proposed methods is a task for further research. In addition, the importance of correctly modeling and interpolating the ionosphere for the user's location has been demonstrated by artificially injecting errors in an SSRZ stream of corrections. The comparison w.r.t. a stream without the injected error indicates the introduction of a few cm errors in the height component of the estimated position.

Moreover, two types of messages to transport the information to perform the proposed interpolation technique have been suggested. The messages contain NWM gradients for the tropospheric message and amplitude, wavelength and direction of the TID for the TID message. The next step would be to include these messages in a format. A suitable format is SSRZ. In fact, the NWM gradients and TID messages could be included as metadata messages either transmitted in the stream or available online.

In conclusion, in this dissertation, the use of raw GNSS measurements retrieved from Android smartphones has been exploited to assess the potential of RTK-level smartphone-based positioning. A quality assessment of the measurements has been carried out in scenarios with different levels of multipath. In open sky and controllable environments, fast cm-level positioning with successful AR has been demonstrated when using PCC computed by an absolute robot-based field calibration of GNSS antennas. The results pave the way towards more complicated configuration with different smartphone orientations and the final objective of having the smartphone in hand. State Space augmentation techniques have been used for the GNSS positioning algorithm. New approaches have been developed to improve the interpolation of atmospheric SSR corrections. The proposed methods mitigate the interpolation error during severe weather and TID events and enhance the AR process robustness during such periods.

## A. RTCM-SSR Python demo

---

An RTCM-SSR demonstrator has been developed in a Python environment. The project aims to provide an open-source tool to decode RTCM-SSR and proposed SSR messages and to translate their impact into the observation domain. The tool has been published in an openly available repository in the GitHub software development platform (Darugna and Wübbena, 2020). The SSR conversion to OSR requires to deal with general GNSS-related aspects like wind-up effect and relativistic effects. All the available GNSSs are implemented (GPS, GLONASS, Galileo, Beidou, QZSS). The following messages are involved (including proposed messages): ephemeris, satellite orbit and clock corrections, satellite code and phase bias, URA (user range accuracy measure), high rate clock, and global VTEC ionosphere. Table A.1 and Table A.2 summarized the standardized and proposed RTCM-SSR messages.

The Python code of the RTCM-SSR demo is reported below as an example for global VTEC ionosphere (see Listing A.1) and wind-up effect computations, see Listing A.1 and Listing A.2, respectively. Listing A.1 shows the computation of the STEC from GVI corrections. The GVI corrections are computed using a spherical harmonics expansion. The computation of the IPP is included in the Python code.

**Table A.1:** RTCM-SSR standardized messages (RTCM Special Committee No. 104, 2016).

Service	Group	Message Types
GPS SSR	Orbit correction	1057
	Clock correction	1058
	Combined orbit and clock corrections	1060
	High-rate clock corrections	1062
	Code bias correction	1059
	User range accuracy	1061
GLONASS SSR	Orbit correction	1063
	Clock correction	1064
	Combined orbit and clock corrections	1066
	High-rate clock corrections	1068
	Code bias correction	1065
	User range accuracy	1067

**Table A.2:** RTCM-SSR proposed messages.

Service	Group	Message Types
GPS SSR	Phase bias correction	1265
GLONASS SSR	Phase bias correction	1266
Galileo SSR	Orbit correction	1240
	Clock correction	1241
	Combined orbit and clock corrections	1243
	High-rate clock corrections	1245
	Code bias correction	1242
	User range accuracy	1244
	Phase bias correction	1267
QZSS SSR	Orbit correction	1246
	Clock correction	1247
	Combined orbit and clock corrections	1249
	High-rate clock corrections	1251
	Code bias correction	1248
	User range accuracy	1250
	Phase bias correction	1268
SBAS SSR	Orbit correction	1252
	Clock correction	1253
	Combined orbit and clock corrections	1255
	High-rate clock corrections	1257
	Code bias correction	1254
	User range accuracy	1256
	Phase bias correction	1269
BDS SSR	Phase bias correction	1258
	Clock correction	1259
	Combined orbit and clock corrections	1261
	High-rate clock corrections	1263
	Code bias correction	1260
	User range accuracy	1262
	Phase bias correction	1270

```

1 """
2 -----
3 Copyright (C) 2020 Francesco Darugna <fd@geopp.de> Geo++ GmbH,
4 Jannes B. Wuebbena <jw@geopp.de> Geo++ GmbH.
5
6 A list of all the historical RTCM-SSR Python Demonstrator contributors in
7 CREDITS.info.

```

```

8
9 The first author has received funding from the European Union's Horizon 2020
10 research and innovation programme under the Marie Skłodowska-Curie Grant
11 Agreement No 722023.
12 -----
13
14 This program is free software: you can redistribute it and/or modify
15 it under the terms of the GNU Affero General Public License as published
16 by the Free Software Foundation, either version 3 of the License, or
17 (at your option) any later version.
18
19 This program is distributed in the hope that it will be useful,
20 but WITHOUT ANY WARRANTY; without even the implied warranty of
21 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
22 GNU Affero General Public License for more details.
23
24 You should have received a copy of the GNU Affero General Public License
25 along with this program. If not, see <https://www.gnu.org/licenses/>.
26 """
27
28 import numpy as np
29 import math
30 from numpy import linalg as LA
31 import rtcm_ssr2osr
32
33 """
34 Set of classes to compute global ionospheric influence on a receiver
35 location for a particular satellite and for a specific frequency.
36
37 Input:
38 - epoch : epoch considered for the computation
39 - state : satellite state vector
40 - rec : receiver position
41 - system: GNSS system involved
42 - ID : ID of the satellite considered
43 - f1 : considered frequency
44 - iono : content of ionospheric RTCM-SSR proposed message 1264
45
46 Output:
47 - STEC value for the selected satellite and frequency for the
48 receiver location.
49 *****
50 Description:
51 the class IonoComputation compute the STEC for a particular satellite +
52 frequency for a specific receiver location. It takes in input the spherical
53 harmonics coefficients of the decoded RTCM-SSR ionospheric message.
54 The satellite and receiver positions are passed to the
55 PiercePoint class which compute pierce point parameters.
56 These and the RTCM-SSR message are the input of
57 the compute_legendre_poly method, which calculates the Legendre
58 polynomial needed to compute the VTEC through the compute_vtec method.
59 Finally, the STEC is computed for the desired frequency.
60 """
61
62 class IonoComputation:
63     def __init__(self, epoch, state, rec, system, ID, f1, iono):
64
65         self.epoch = epoch
66         self.sat = state[0:3]
67         self.rec = rec
68         self.re = rtcm_ssr2osr.Constants().re
69         self.omega_e = rtcm_ssr2osr.Constants().omega_e # [rad/s]
70         self.system = system
71
72         self.layers = iono.n_layers
73         self.stec_corr_f1 = 0
74         self.strg = ''
75         for l in range(self.layers):
76             self.height = iono.height[l]
77             self.sh_deg = iono.degree[l]
78             self.sh_ord = iono.order[l]
79             self.c = iono.c[l][:]
80             self.s = iono.s[l][:]
81
82         [lat_sph, lon_sph, height_sph,
83          el, az, psi_pp,
84          lambda_pp, phi_pp,
85          sf, sun_shift, lon_s,
86          p_nm, p_cos, p_sin, m, n,

```

```

86         vtec] = IonoComputation.compute_global_iono(self)
87         stec = vtec * sf
88         self.stec_corr_f1 += 40.3 * 1e16 / (f1 * f1) * stec
89
90         self.strg += ('### SV pos/vel for SV ' + ID + ' at ' + f'{epoch}' +
91                     ': ' + '{:16.4f}'.format(state[0]) + ' ' +
92                     '{:16.4f}'.format(state[1]) + ' ' +
93                     '{:16.4f}'.format(state[2]) + ' [m]' + ' ' +
94                     '{:9.4f}'.format(state[3]) + ' ' +
95                     '{:9.4f}'.format(state[4]) + ' ' +
96                     '{:9.4f}'.format(state[5]) + ' [m/s]' + '\n' +
97                     'Ppt at t=' +
98                     f'{epoch}' + '(sun shift= ' +
99                     '{:11.8f}'.format(sun_shift * 180 / np.pi) +
100                    ' deg)' + '\n' +
101                    'Ppt from Ref phi_R= ' +
102                    '{:11.8f}'.format(lat_sph * 180 / np.pi) +
103                    ' lam_R= ' +
104                    '{:11.8f}'.format(lon_sph * 180 / np.pi) +
105                    ' rE+hR= ' + '{:10.3f}'.format(height_sph +
106                    6370000) +
107                    '(spherical!)' + '\n' +
108                    'Ppt from Ref to SV at elev= ' +
109                    '{:11.8f}'.format(el * 180 / np.pi) + ' azim ' +
110                    '{:11.8f}'.format(az * 180 / np.pi) +
111                    '(spherical!)' + '\n' +
112                    'Ppt psi_pp= ' +
113                    '{:11.8f}'.format(psi_pp * 180 / np.pi) +
114                    ' phi_pp ' +
115                    '{:11.8f}'.format(phi_pp * 180 / np.pi) +
116                    ' lam_pp ' +
117                    '{:11.8f}'.format(lambda_pp * 180 / np.pi) +
118                    ' lon_S ' +
119                    '{:11.8f}'.format(lon_s * 180 / np.pi) +
120                    ' rE+hI: ' +
121                    '{:10.3f}'.format(self.height * 1000 + 6370000) +
122                    '\n' + 'Pnm : ')
123         # Lagrange Polynomials
124         for o in range(len(p_nm)):
125             m_ind = int(m[o])
126             n_ind = int(n[o])
127             self.strg += ('P(' + f'{n_ind}' + ', ' +
128                         f'{m_ind}' + ')=' +
129                         '{:7.4f}'.format(p_nm[o]) + '; ')
130         # Cosines
131         self.strg += '\n' + 'Pcos: '
132         for o in range(len(p_cos)):
133             m_ind = int(m[o])
134             n_ind = int(n[o])
135             self.strg += ('P(' + f'{n_ind}' + ', ' +
136                         f'{m_ind}' + ')=' +
137                         '{:7.4f}'.format(p_cos[o]) + '; ')
138         # Sines
139         self.strg += '\n' + 'Psin: '
140         for o in range(len(p_sin)):
141             m_ind = int(m[o])
142             n_ind = int(n[o])
143             self.strg += ('P(' + f'{n_ind}' + ', ' +
144                         f'{m_ind}' + ')=' +
145                         '{:7.4f}'.format(p_sin[o]) + '; ')
146         self.strg += ('\n' +
147                     'Sum VTEC=' +
148                     '{:6.3f}'.format(vtec) +
149                     '[TECU]' + ', ' + 'sf=' +
150                     '{:6.3f}'.format(sf) +
151                     ', ' + 'STEC=' +
152                     '{:6.3f}'.format(stec) +
153                     '[TECU]' + '\n' +
154                     'SSR_VTEC: SV' + ID +
155                     ' Have SSR VTEC Iono slant influence: ' +
156                     '{:6.3f}'.format(stec) + '[TECU]' +
157                     '{:6.3f}'.format(self.stec_corr_f1) +
158                     '[m-L1]' + '\n')
159
160         def __str__(self):
161             return self.strg
162
163         def compute_legendre_poly(self, max_val, lat_pp, lon_s):

```

```

164     """ Recursive Legendre polynomials computation
165
166     """
167     x = np.sin(lat_pp)
168
169     # ***** Calculate Legendre polynomials with recursion algorithm ***** #
170     nmax = int(max_val + 1)
171     p = np.zeros((nmax, nmax))
172     p[0][0] = 1.0
173
174     for m in range(1, nmax, 1):
175         p[m][m] = (2 * m - 1) * np.sqrt((1 - x * x)) * p[m - 1][m - 1]
176
177     for m in range(1, nmax - 1, 1):
178         p[m + 1][m] = (2 * x + 1) * x * p[m][m]
179
180     for m in range(0, nmax, 1):
181         for n in range(m + 1, nmax, 1):
182             p[n][m] = 1 / (n - m) * ((2 * n - 1) * x * p[n - 1][m] -
183                                     (n + m - 1) * p[n - 2][m])
184
185     # ***** Compute associated Legendre polynomials Nnm ***** #
186
187     p_nm = []
188     p_cos = []
189     p_sin = []
190
191     m_ind = []
192     n_ind = []
193
194     for n in range(0, nmax, 1):
195         for m in range(0, n + 1, 1):
196
197             s2 = (((2 * n + 1) * math.factorial(n - m)) /
198                  (math.factorial(n + m)))
199
200             if(m == 0):
201                 n_nm = np.sqrt(1 * s2) * p[n][m]
202             else:
203                 n_nm = np.sqrt(2 * s2) * p[n][m]
204
205             p_nm = np.append(p_nm, n_nm)
206
207             p_cos = np.append(p_cos, n_nm * np.cos(m * lon_s))
208             p_sin = np.append(p_sin, n_nm * np.sin(m * lon_s))
209             m_ind = np.append(m_ind, m)
210             n_ind = np.append(n_ind, n)
211
212
213
214     return (p_nm, p_cos, p_sin, m_ind, n_ind)
215
216 # =====
217 #                               VTEC computation
218 # =====
219 def compute_vtec(self, order, degree, c_nm, s_nm, p_cos, p_sin):
220     """ VTEC computation
221
222     """
223     vtec = 0
224     nmax = int(degree + 1)
225     i = 0
226
227     for n in range(0, nmax, 1):
228         mmax = int(np.min([n, order])) + 1
229         for m in range(0, mmax, 1):
230             tot_1 = 0
231             tot_2 = 0
232             if m == 0:
233                 tot_1 = c_nm[m][0][n] * p_cos[i]
234
235             else:
236                 tot_2 = (c_nm[m][0][n] * p_cos[i] +
237                         s_nm[m - 1][0][n - 1] *
238                         p_sin[i])
239
240             vtec = vtec + tot_1 + tot_2
241             i = i + 1

```

```

242     return vtec
243
244
245 # =====
246 #                               Global ionospheric corrections
247 # =====
248 def compute_global_iono(self):
249     """ Computation of IONO correction per satellite
250
251     """
252
253     c_nm = []
254     s_nm = []
255     c_print = np.array(self.c)
256     s_print = np.array(self.s)
257     order = self.sh_ord
258     deg = self.sh_deg
259     nc = len(self.c)
260     ns = len(self.s)
261     h = self.height
262     index = 0
263
264     for j in range(int(order) + 1):
265         c_nm.append([])
266         if index < nc:
267             c2append = np.concatenate((np.zeros(j),
268                                       c_print[index : index +
269                                               (int(deg) +
270                                                 1 - j)]), axis=0)
271             c_nm[j].append(c2append)
272             index = index + (int(deg) + 1 - j)
273
274     index = 0
275     for j in range(int(order)):
276         s_nm.append([])
277         if index < ns:
278             s2append = np.concatenate((np.zeros(j),
279                                       s_print[index : index +
280                                               (int(deg) - j)]), axis=0)
281             s_nm[j].append(s2append)
282             index = index + (int(deg) - j)
283
284     # ***** #
285     # #
286     #           Pierce Point computation #
287     # #
288     # ***** #
289     # height needs to be in meter, but the input h is in km
290     pp_comp = PiercePoint(self.sat, self.rec, h * 1000)
291     # apply spin correction
292     xyz_spin = pp_comp.compute_spin_corr()
293     pp_comp_spin = PiercePoint(xyz_spin, self.rec,
294                               h * 1000)
295     # compute spherical coordinates
296     [lat_sph, lon_sph, height_sph] = pp_comp_spin.compute_xyz2sph()
297     # compute az and el using spherical coordinates
298     [az, el] = pp_comp_spin.compute_az_el(lat_sph, lon_sph)
299
300     # compute pierce point
301     [psi_pp, lambda_pp,
302      phi_pp, sf,
303      sun_shift, lon_s] = pp_comp_spin.compute_pp(lat_sph, lon_sph,
304                                                  height_sph, el, az,
305                                                  self.epoch)
306
307     # computation of the iono delay
308     max_val = np.max([order, deg])
309     [p_nm, p_cos, p_sin,
310      m, n] = IonoComputation.compute_legendre_poly(self, max_val,
311                                                    phi_pp, lon_s)
312
313     vtec = IonoComputation.compute_vtec(self, order, deg, c_nm, s_nm,
314                                       p_cos, p_sin)
315
316     return (lat_sph, lon_sph, height_sph, el, az, psi_pp,
317           lambda_pp, phi_pp,
318           sf, sun_shift, lon_s, p_nm, p_cos, p_sin, m, n, vtec)
319

```

```

320 # =====
321 #               Pierce Point computation class
322 # =====
323 class PiercePoint:
324     """ Pierce Point calculator
325     including:
326         - relative elev, azi computation
327     Input:
328         - satellite and receiver coordinates
329         - ionospheric layer height
330     Output:
331         - pierce point parameters:
332           psi_pp, lambda_pp, phi_pp, slant_factor, sun_shift, lon_s
333     """
334 # =====
335 #               Data and Initialization
336 # =====
337     def __init__(self, sat, rec, layer_height):
338         # Satellite and receiver position in ECEF + layer height
339         self.sat = sat
340         self.rec = rec
341         self.layer_h = layer_height
342         # *****
343         #               Ancillary data
344         # *****
345         self.re = 6.37e6 # [m]
346
347         self.wgs84_a = 6378137.0 # [m]
348         self.wgs84_e = 0.0818191908426
349
350         self.omega_zero_dot = 7.2921151467e-5 # earth rotation rate rad/s
351         self.c = 2.99792458e+8 # gps para 4.3
352
353 # =====
354 #               Relative azimuth elevation
355 # =====
356     def compute_az_el(self, lat, lon):
357         """ Relative azimuth and elevation computation
358
359         Reference:
360         "Satellite Orbits", Montenbruck & Gill, chapter 6.2 pages 211-212
361         """
362         R = np.array([[ -np.sin(lon)           ,
363                     +np.cos(lon)           ,
364                     +0                       ],
365                    [ -np.sin(lat) * np.cos(lon),
366                     -np.sin(lat) * np.sin(lon),
367                     +np.cos(lat)           ],
368                    [ +np.cos(lat) * np.cos(lon),
369                     +np.cos(lat) * np.sin(lon),
370                     +np.sin(lat)           ]])
371
372         s = np.dot(R, self.sat - self.rec)
373
374         azimuth = np.arctan2(s[0], s[1])
375         elevation = np.arctan2(s[2], np.sqrt(s[0]**2 + s[1]**2))
376
377         if azimuth < 0:
378             azimuth = azimuth + 2 * np.pi
379
380         return np.array([azimuth, elevation])
381
382 # =====
383 #               From XYZ coord to spherical lat, lon, height
384 # =====
385     def compute_xyz2sph(self):
386         """ From coord to spherical lat, lon , height
387
388         """
389
390         height_s = LA.norm(self.rec) - self.re
391
392         p = LA.norm(self.rec[0:2])
393
394         lat_s = np.arctan2(self.rec[2], p)
395         lon_s = np.arctan2(self.rec[1], self.rec[0])
396
397

```

```

398         return np.array([lat_s, lon_s, height_s])
399
400 # =====
401 #                               From spherical lat, lon to xyz spherical
402 # =====
403     def compute_sph2sph_xyz(self, r, lat, lon):
404         """ From spherical lat, lon to xyz spherical
405             """
406         x = r * np.cos(lat) * np.cos(lon)
407         y = r * np.cos(lat) * np.sin(lon)
408         z = r * np.sin(lat)
409         return np.array([x, y, z])
410
411 # =====
412 #                               XYZ sat SPIN correction
413 # =====
414     def compute_spin_corr(self):
415         """ Compute satellite spin
416             """
417
418         self.range = np.sqrt((self.sat[0] - self.rec[0]) ** 2 +
419                               (self.sat[1] - self.rec[1]) ** 2 +
420                               (self.sat[2] - self.rec[2]) ** 2)
421
422         sat_spin = np.zeros((1,3))
423         sat_spin[0,0] = (self.sat[0] +
424                         (self.sat[1] * self.omega_zero_dot *
425                          (self.range / self.c)))
426         sat_spin[0,1] = (self.sat[1] - (self.sat[0] * self.omega_zero_dot *
427                                       (self.range / self.c)))
428         sat_spin[0,2] = self.sat[2]
429         return sat_spin[0]
430
431 # =====
432 #                               Pierce Point computation
433 # =====
434     def compute_pp(self, lat, lon, height, el, az, t):
435         """ Pierce Point computation method
436             """
437
438         # ***** Spherical Earth's central angle ***** #
439         # angle between rover position and the projection of the pierce point
440         # to the spherical Earth surface
441         tmp = ((self.re + height) /
442               (self.re + self.layer_h) * np.cos(el))
443
444         psi_pp = np.pi / 2 - el - np.arcsin(tmp)
445
446         # ***** Latitude and Longitude PP ***** #
447         tmp = np.tan(psi_pp) * np.cos(az)
448         ctg_lat = 1 / np.tan(lat)
449         # Latitude
450         phi_pp = (np.arcsin(np.sin(lat) * np.cos(psi_pp) +
451                             np.cos(lat) * np.sin(psi_pp) * np.cos(az)))
452
453         ang = np.arcsin(np.sin(psi_pp) * np.sin(az) / np.cos(phi_pp))
454
455         # Longitude
456         if(((lat >= 0) & (+tmp > ctg_lat)) |
457            ((lat < 0) & (-tmp > ctg_lat))):
458             lambda_pp = lon + np.pi - ang
459         else:
460             lambda_pp = lon + ang
461
462         sun_shift = math.fmod((t - 50400) * np.pi / 43200, 2 * np.pi)
463
464         lon_s = math.fmod(lambda_pp + sun_shift, 2 * np.pi)
465
466         slant_factor = 1.0 / np.sin(el + psi_pp)
467
468         return np.array([psi_pp, lambda_pp, phi_pp, slant_factor,
469                         sun_shift, lon_s])

```

**Listing A.1:** Python class to compute global VTEC ionosphere correction from proposed RTCM-SSR messages.

```

1 class WindUp:

```

```

2  def __init__(self, pbias, sv, dt, state, fr, rec, lat, lon):
3      """ Function to compute the wind up effect
4          Input:
5              - pbias with objects the yaw angle and the yaw rate
6              - sat state vector in ECEF
7              - rec coord in ECEF
8              - ellip lat, long of the rec
9              - the frequency of the signals considered
10         Output:
11             - phase wind up correction for the input frequency
12         """
13         # find correct satellite index
14         try:
15             idx = np.where(pbias.gnss_id == sv)
16         except AttributeError:
17             self.corr = []
18             return
19         if np.size(idx) == 0:
20             self.corr = []
21         else:
22             ii = int(idx[0])
23             sat = state[0:3]
24             vel = state[3:]
25
26             lam = Constants().c / fr
27             diff = rec - sat
28             k = diff / LA.norm(diff)
29
30             # from deg to rad for lat, lon
31             lat = np.deg2rad(lat)
32             lon = np.deg2rad(lon)
33
34             # correction for Eart rotation
35             vel[0] = vel[0] - Constants().omega_e * sat[1]
36             vel[1] = vel[1] + Constants().omega_e * sat[0]
37
38             # ee, en, eu unit vecotrs in ENU ref frame
39             ee = np.array([-np.sin(lon)
40                           ,
41                           +np.cos(lon)
42                           ,
43                           +0
44                           ])
45             en = np.array([-np.cos(lon) * np.sin(lat),
46                           -np.sin(lon) * np.sin(lat),
47                           +np.cos(lat)
48                           ])
49
50             # Computation of the ex, ey, ez unit vectors
51             ez = -sat / LA.norm(sat)
52             ey = -np.cross(sat, vel) / LA.norm(np.cross(sat, vel))
53             ex = np.cross(ey, ez)
54
55             # yaw angle rotation
56             yaw = np.deg2rad(pbias.yaw_angle[ii] + pbias.yaw_rate[ii] * dt)
57             R = np.array([[+np.cos(yaw), np.sin(yaw), 0],
58                           [-np.sin(yaw), np.cos(yaw), 0],
59                           [0, 0, 1]])
60
61             e_xyz = np.array([ex, ey, ez])
62             e_Rxyz = np.dot(R, e_xyz)
63             ex = e_Rxyz[0]
64             ey = e_Rxyz[1]
65             ez = e_Rxyz[2]
66
67             # Effective dipole for the satellite
68             flag = 'sat'
69             D_sat = WindUp.compute_eff_dipole(k, ex, ey, flag)
70
71             # Effective dipole for the receiver
72             flag = 'rec'
73             D_rec = WindUp.compute_eff_dipole(k, ee, en, flag)
74
75             # Wind up computation
76             gamma = np.dot(k, np.cross(D_sat, D_rec))
77
78             omega = np.arccos(np.dot(D_sat, D_rec) /
79                               (LA.norm(D_sat) * LA.norm(D_rec)))
80             omega = -omega / (2 * np.pi)
81
82             if gamma < 0:
83                 omega = -omega

```

```

80         # Correction for specific wavelength
81         self.corr = omega * lam
82
83     def compute_eff_dipole(k, ex, ey, flag):
84         """ Computation of the effective dipole for the phase wind up
85
86         Input:
87         - k      : dist unit vector for sat-rec in ECEF
88         - ex, ey: unit vectors in the x,y plane.
89           If the satellite is considered
90             then x = t and y = -n directions
91             (ref to radial-track-normal ref frame);
92           if the receiver is considered x = E and y = N
93             (North-East-Up ref frame)
94         - flag  : if 'sat' the formula
95                   for the satellite is considered,
96                   if 'rec' the formula
97                   for the receiver is considered
98
99         Output:
100        - D: effective dipole
101
102        Formulas:
103        --> sat: D = ex - k*dot(k,ex) - cross(k,ey)
104        --> rec: D = ex - k*dot(k,ex) + cross(k,ey)
105
106        Reference:
107        Springer Handbook for GNSS, Teunissen & Montenbruck,
108        chap.19 pag. 570
109
110        """
111
112        if flag == 'sat':
113            D = ex - k * np.dot(k, ex) - np.cross(k, ey)
114        elif flag == 'rec':
115            D = ex - k * np.dot(k, ex) + np.cross(k, ey)
116
117        return D

```

**Listing A.2:** Python class to compute the wind-up effect having as input the RTCM-SSR satellite phase bias message.

## B. SSRZ Python demo

Similarly to the RTCM-SSR demonstrator, an SSRZ demonstrator has been developed and can be shared upon request by Geo++. The decoded SSRZ messages are reported in Table B.1. As examples, Python codes to compute the RT, GRT, RSI and GSI corrections are reported in Listing B.1 and Listing B.2.

Listing B.1 shows the code to compute the tropospheric correction following Section 2.2. The ratio w.r.t. the model is computed using the Chebyshev polynomial for RT and interpolated for the GRT (using a simple IDW method). After multiplying the ratio for the value of the tropospheric model (using the UNB3 model Leandro et al. (2006)), the delay is mapped to the slant tropospheric delay using the VGME.

Listing B.2 reports the Python methods to compute the RSI and GSI corrections. The calculation follows Section 2.2. The method `compute_dne_pp` computes the difference in the projected North and East direction between satellite's pierce point and the PPO. For the calculation of the IPP see Appendix A. The North and East differences are used in the `compute_sat_vtec` to compute the VTEC for a specific satellite employing the Chebyshev polynomials (see Eq. 2.31).

**Table B.1:** Current SSRZ messages (Geo++ GmbH, 2020).

Service	Group	Message Types
SSRZ metadata	Satellite	4090.7.11
	Metadata	4090.7.12
	Grid definition	4090.7.13
SSRZ corrections	High rate	4090.7.1
	Low rate	4090.7.2
	Gridded ionosphere	4090.7.3
	Gridded troposphere	4090.7.4
	Satellite dependent regional ionosphere	4090.7.5
	Global VTEC ionosphere	4090.7.6
	Regional troposphere	4090.7.7
	QIX bias	4090.7.8
	SSRZ time tag	4090.7.9

```

1  """
2  -----
3  Copyright (C) 2020 Francesco Darugna <fd@geopp.de> Geo++ GmbH,
4                      Jannes B. Wuebbena <jw@geopp.de> Geo++ GmbH.
5
6  A list of all the historical SSRZ Python Demonstrator contributors in
7  CREDITS.info.
8
9  The first author has received funding from the European Union's Horizon 2020
10 research and innovation programme under the Marie Skłodowska-Curie Grant
11 Agreement No 722023.
12 -----
13 """

```

```

14 This program is free software: you can redistribute it and/or modify
15 it under the terms of the GNU Affero General Public License as published
16 by the Free Software Foundation, either version 3 of the License, or
17 (at your option) any later version.
18
19 This program is distributed in the hope that it will be useful,
20 but WITHOUT ANY WARRANTY; without even the implied warranty of
21 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
22 GNU Affero General Public License for more details.
23
24 You should have received a copy of the GNU Affero General Public License
25 along with this program. If not, see <https://www.gnu.org/licenses/>.
26 """
27
28 import numpy as np
29 import tropo_mdl
30 import coord_and_time_transformations as trafo
31 import interp_module as interp
32 """
33 Class of methods to compute tropospheric influences for a receiver
34 location of a specific tropospheric component (e.g. dry).
35
36 Input:
37 - epoch : epoch considered for the computation
38 - ssr : decoded ssr messages
39 - rec : receiver position ellipsoidal coordinates (lat, lon, hgt)
40 - cc : index of the component considered
41
42 Output:
43 - zenith tropo delay for a specific component
44 *****
45 Description:
46 the tropospheric ratio w.r.t. the model is computed using the coefficients
47 read from the SSRZ messages. The ratio is then multiplied by the
48 model (UNB3) and the mapping function (GVMF) to get the slant tropospheric
49 delay.
50 *****
51 Remark:
52 concerning the grid troposphere, an inverse distance weighting (IDW)
53 method is considered as simple reference. The user can choose more
54 sophisticated methods depending on the application.
55 """
56 class TropoComputation:
57     """ Class to compute tropospheric delay based on the SSR messages.
58         UNB3 model, Vienna Global Mapping Functions.
59     """
60     def __init__(self, epoch, ssr, rec_llh, el, doy, tropo_type, comp,
61                 comp_index, md=None):
62         if md is not None:
63             self.md = md
64         else:
65             self.md = None
66         self.ssr = ssr
67         self.rec_llh = rec_llh
68         self.cc = comp_index
69         self.doy = doy
70         # compute the UNB3 model with 2 components: dry[0] and wet[1]
71         model = tropo_mdl.get_model_troposphere(rec_llh, doy)
72         # apply global mapping function
73         zd = np.pi/2 - el
74         lat_rad = np.deg2rad(self.rec_llh[0])
75         lon_rad = np.deg2rad(self.rec_llh[1])
76         [gmfh, gmfw] = tropo_mdl.compute_gmf(doy, lat_rad, lon_rad,
77                                             self.rec_llh[2],
78                                             zd)
79
80         # model computation
81         model_tot_slant = model[0] * gmfh + model[1] * gmfw
82         if tropo_type == 'rt':
83             # compute regional tropo effect
84             tropo_ratio = self.compute_rt()
85             self.tropo = model_tot_slant * tropo_ratio
86         elif tropo_type == 'grt':
87             # compute grid regional tropo effect
88             tropo_ratio = self.compute_grt()
89             # compute the value in meters
90             if comp == 'd':
91                 self.tropo = gmfh * model[0] * tropo_ratio

```

```

92         elif comp == 'w':
93             self.tropo = gmfw * model[1] * tropo_ratio
94
95     def compute_rt(self):
96         """ Method to compute the regional troposphere
97         """
98         # ground point coordinates
99         gpo_llh = self.ssr.rt.gpo_llh
100        # compute lat, lon, hgt difference between receiver and ground point
101        dlat = np.deg2rad(self.rec_llh[0] - gpo_llh[0])
102        dlon = np.deg2rad(self.rec_llh[1] - gpo_llh[1])
103        dhgt = self.rec_llh[2] - gpo_llh[2]
104        # define horizontal and vertical correlation lengths
105        corr_hor = 500e3 # [m]
106        corr_ver = 2e3 # [m]
107        # scaled for the correlation length and the longitude according to the
latitude
108        dN = dlat * 6378135.0 / corr_hor
109        dE = dlon * np.cos(np.deg2rad(gpo_llh[0])) * 6378135.0 / corr_hor
110        dh = dhgt / corr_ver
111        # set max order per lat, lon, hgt
112        max_order_lat = self.ssr.rt.max_order_lat
113        max_order_lon = self.ssr.rt.max_order_lon
114        max_order_hgt = self.ssr.rt.max_order_hgt
115        # load coefficients
116        a_coeff = self.ssr.rt.coeff[self.cc]
117        # define the list of ii and jj in the coefficient of the ssrz corrections
118        # based on Table 3.6 of the SSRZ document.
119        # example: a00    a10    a01    a20    a11    a02
120        ii_list = np.array([0, 0, 1, 2, 1, 0, 3, 2, 1, 0, 4, 3, 2, 1, 0])
121        jj_list = np.array([0, 1, 0, 0, 1, 2, 0, 1, 2, 3, 0, 1, 2, 3, 4])
122        # adjust the lists to the order for latitude and longitude
123        ii_list_adj = ii_list[np.where((ii_list < max_order_lat) &
124                                     (jj_list < max_order_lon))]
125        jj_list_adj = jj_list[np.where((ii_list < max_order_lat) &
126                                     (jj_list < max_order_lon))]
127
128        rt = 0
129        nn = 0 # number of parameters
130        # initialize chebyshev polynomials
131        # Remark: the construction is assuming that max_order_hgt is 1
132        for kk in range(max_order_hgt):
133            cheb_poly_dlat = []
134            for ii in range(max_order_lat):
135                cheb_poly_dlat.append([])
136                cheb_poly_dlon = []
137                for jj in range(max_order_lon):
138                    if ((ii + jj <= max_order_lat + max_order_lon) &
139                        (nn <= max_order_hgt * max_order_lat * max_order_lon)):
140                        cheb_poly_dlon.append([])
141                        cheb_poly_dlat[ii] = self.compute_chebyshev_poly(ii, dN,
142
143        cheb_poly_dlat)
144
145        cheb_poly_dlon[jj] = self.compute_chebyshev_poly(jj, dE,
146
147        cheb_poly_dlon)
148
149        # find the correct index for the a coefficient
150        ll = np.where(((ii_list_adj==ii)&(jj_list_adj==jj)))[0][0]
151        a_ij = a_coeff[ll]
152        rt += a_ij * (dh ** kk) * cheb_poly_dlat[ii] *
153
154        cheb_poly_dlon[jj]
155
156        nn += 1
157
158        return rt
159
160     def compute_chebyshev_poly(self, ii, x, cheb_poly):
161         """ Recursive formulation for Chebyshev polynomials
162         """
163         if ii == 0:
164             t = 1
165         elif ii == 1:
166             t = x
167         elif ii == 2:
168             t = 2 * x **2 - 1
169         else:
170             t = 2 * x * cheb_poly[ii-1] - cheb_poly[ii-2]
171
172         return t
173
174     def compute_grt(self):

```

```

166     """ Method to compute the influence of the gridded troposphere
167         for the user location through interpolation based on IDW.
168         Be aware that the grid is a ratio relative to the Saastamoinen
169         model, therefore independent from the height.
170     """
171     # get gridded troposphere
172     data = self.ssr.grt.grid_values[self.cc]
173     # get grid llh from metadata
174     grid_list = self.md.grid_gr.grid_block.grid_blk_list
175     lat = []
176     lon = []
177     hgt = []
178     for ii in range(len(grid_list)): # loop over the number of grids
179         chain_blk = grid_list[ii].chain_blk
180         for jj in range(len(chain_blk)): # loop over the number of chains
181             lat = np.append(lat, chain_blk[jj].lat)
182             lon = np.append(lon, chain_blk[jj].lon)
183             hgt = np.append(hgt, chain_blk[jj].hgt)
184
185     # define rereference for the local coordinates
186     lat0 = self.rec_llh[0]
187     lon0 = self.rec_llh[1]
188     hei0 = self.rec_llh[2]
189     # query point for the interpolation
190     latq = self.rec_llh[0]
191     lonq = self.rec_llh[1]
192     heiq = self.rec_llh[2]
193     [x0, y0, z0] = trafo.ell2cart(lat0, lon0, hei0)
194     E = []
195     N = []
196     lat_deg = []
197     lon_deg = []
198     for ii in range(len(lat)):
199         lat_pt = np.rad2deg(lat[ii])
200         lon_pt = np.rad2deg(lon[ii])
201         lat_deg.append(lat_pt)
202         lon_deg.append(lon_pt)
203         [x,y,z] = trafo.ell2cart(lat_pt, lon_pt, hei0)
204         dx = x - x0
205         dy = y - y0
206         dz = z - z0
207         north = -dx * np.sin(np.deg2rad(lat_pt)) * np.cos(np.deg2rad(lon_pt))
- dy * np.sin(np.deg2rad(lat_pt))*np.sin(np.deg2rad(lon_pt)) + dz * np.cos(np.
deg2rad(lat_pt))
208         east = -dx * np.sin(np.deg2rad(lon_pt)) + dy * np.cos(np.deg2rad(
lon_pt))
209         E = np.append(E, east)
210         N = np.append(N, north)
211     # define the query point in the local system
212     [xq,yq,zq] = trafo.ell2cart(latq, lonq, heiq)
213     dxq = xq - x0
214     dyq = yq - y0
215     dzq = zq - z0
216     northq = -dxq * np.sin(np.deg2rad(latq)) * np.cos(np.deg2rad(lonq)) - dyq
* np.sin(np.deg2rad(latq))*np.sin(np.deg2rad(lonq)) + dzq * np.cos(np.deg2rad(
latq))
217     eastq = -dxq * np.sin(np.deg2rad(lonq)) + dyq * np.cos(np.deg2rad(lonq))
218     query = np.array([[eastq, northq]])
219     # grid creation
220     grid_en = np.array([[E[ii], N[ii]] for ii in range(len(E))])
221     # directions
222     directions = np.array([[E[i] - query[0][0],
223                             N[i] - query[0][1]] for i in range(len(E))])
224     distances = []
225     for ii in range(len(directions)):
226         distances = np.append(distances, np.linalg.norm(directions[ii]))
227     # compute 2D interpolation
228     values = np.array([[data[ii]] for ii in range(len(data))])
229     interpolation = interp.Interpolator2D(grid_en, values)
230     interpolation.IDW(query)
231     tropo_ratio_interp = interpolation.results
232
233     return tropo_ratio_interp

```

**Listing B.1:** Python class to compute RT and GRT corrections from SSRZ messages. An IDW method has been considered for the GRT interpolation as an example.

```

1 # =====
2 #                               Satellite-dependent ionospheric vtec
3 # =====
4 def compute_sat_vtec(self, a_coeff, max_order, nc_max, idx_gnss, idx_sat,
5                       dN, dE):
6     """
7     Computation of satellite dependent VTEC for both gsi and rsi.
8     The computation is based on Chebyshev polynomials.
9     The a_coeff in input are not considered to be already in the correct
10    order of application.
11    Recursive formulation for Chebyshev polynomials
12    if ii == 0:
13        t = 1
14    elif ii == 1:
15        t = x
16    elif ii == 2:
17        t = 2 * x **2 - 1
18    else:
19        t = 2 * x * cheb_poly[ii-1] - cheb_poly[ii-2]
20    """
21    # define the list of ii and jj in the coefficient of the ssrz corrections
22    if nc_max == 1:
23        ii_list = np.array([0])
24        jj_list = np.array([0])
25    elif nc_max == 3:
26        ii_list = np.array([0, 0, 1])
27        jj_list = np.array([0, 1, 0])
28    elif nc_max == 4:
29        ii_list = np.array([0, 0, 1, 1])
30        jj_list = np.array([0, 1, 0, 1])
31    elif nc_max == 5:
32        ii_list = np.array([0, 0, 1, 0, 2])
33        jj_list = np.array([0, 1, 0, 2, 0])
34    elif nc_max == 6:
35        ii_list = np.array([0, 0, 1, 0, 1, 2])
36        jj_list = np.array([0, 1, 0, 2, 1, 0])
37    vtec = 0
38    rho = 6.37 # scale factor
39    dN *= rho
40    dE *= rho
41    # initialize chebyshev polynomials
42    cheb_poly_dN = []
43    cheb_poly_dE = []
44    nc = 0 # initialize number of coefficients
45    for ii in range(max_order + 1):
46        cheb_poly_dN.append([])
47        cheb_poly_dE = []
48        cheb_poly.append([])
49        for jj in range(max_order + 1):
50            if ((ii + jj <= max_order) & (nc <= nc_max)):
51                cheb_poly[ii].append([])
52                cheb_poly_dE.append([])
53                cheb_poly_dN[ii] = self.compute_chebyshev_poly(ii, dN,
54                                                                cheb_poly_dN)
55                cheb_poly_dE[jj] = self.compute_chebyshev_poly(jj, dE,
56                                                                cheb_poly_dE)
57                # find the correct index for the a coefficient
58                kk = np.where(((ii_list==ii)&(jj_list==jj)))[0][0]
59                try:
60                    a_ij = a_coeff[kk][idx_gnss][idx_sat]
61                except:
62                    print(kk, idx_gnss, idx_sat)
63                vtec += a_ij * cheb_poly_dN[ii] * cheb_poly_dE[jj]
64                cheb_poly[ii][jj] = cheb_poly_dN[ii] * cheb_poly_dE[jj]
65                nc += 1
66    return vtec
67
68 def compute_chebyshev_poly(self, ii, x, cheb_poly):
69     """ Recursive formulation for Chebyshev polynomials
70     """
71     if ii == 0:
72         t = 1
73     elif ii == 1:
74         t = x
75     elif ii == 2:
76         t = 2 * x **2 - 1
77     else:
78         t = 2 * x * cheb_poly[ii-1] - cheb_poly[ii-2]

```

```

79
80     return t
81 # =====
82 #     Compute dN and dE
83 # =====
84 def compute_dne_pp(self, layer_hgt, iono_type, gpo=None):
85     """ Computation of differences in the projected North and East
86         direction (radians) between satellite's pierce point and the
87         pierce point origin
88
89     """
90     h = layer_hgt
91     # ***** #
92     # #
93     #     Receiver Pierce Point computation #
94     # #
95     # ***** #
96     # height needs to be in meter, but the input h is in km
97     pp_comp = PiercePoint(self.sat, self.rec_xyz, h * 1000)
98     # apply spin correction
99     xyz_spin = pp_comp.compute_spin_corr(self.sat, self.rec_xyz)
100    pp_comp_spin = PiercePoint(xyz_spin, self.rec_xyz,
101                               h * 1000)
102
103    # compute spherical coordinates
104    [lat_sph, lon_sph, height_sph] = pp_comp_spin.compute_xyz2sph(self.rec_xyz)
105    # compute az and el using spherical coordinates
106    [az, el] = pp_comp_spin.compute_az_el(xyz_spin, self.rec_xyz,
107                                          lat_sph, lon_sph)
108
109    # compute pierce point
110    [psi_pp, lambda_pp,
111     phi_pp, sf_pp,
112     sun_shift_pp, lon_s_pp] = pp_comp_spin.compute_pp(lat_sph, lon_sph,
113                                                       height_sph, el, az,
114                                                       self.epoch)
115
116    lon_pp = lambda_pp #lon_s_pp # rover Pierce Point coordinates
117    # ***** #
118    # #
119    #     Ground origin Pierce Point computation #
120    # #
121    # ***** #
122    if iono_type == 'gsi':
123        # the ppo is the pierce point of the nadir direction of
124        # satellite with respect to the spherical ionospheric layer
125        ppo_comp = PiercePoint(self.sat, self.sat, h * 1000)
126        [phi_ppo, lambda_ppo,
127         height_sph] = ppo_comp.compute_xyz2sph(self.sat)
128        sun_shift = math.fmod((self.epoch - 50400) * np.pi /
129                              43200, 2 * np.pi)
130
131        lon_ppo = math.fmod(lambda_pp + sun_shift, 2 * np.pi)
132    else:
133        # Compute XYZ for PPO from ellipsoidal
134        gpo_xyz = trafo.el12cart(gpo[0], gpo[1], gpo[2])
135        ppo_comp = PiercePoint(self.sat, gpo_xyz, h * 1000)
136        # apply spin correction
137        sat_spin = ppo_comp.compute_spin_corr(self.sat, gpo_xyz)
138        ppo_comp_spin = PiercePoint(sat_spin, gpo_xyz,
139                                   h * 1000)
140
141        # compute spherical coordinates
142        [lat_sph, lon_sph, height_sph] = ppo_comp_spin.compute_xyz2sph(gpo_xyz)
143        # compute az and el using spherical coordinates
144        [az, el] = ppo_comp_spin.compute_az_el(sat_spin, gpo_xyz,
145                                              lat_sph, lon_sph)
146
147        # compute pierce point
148        [psi_ppo, lambda_ppo,
149         phi_ppo, sf_ppo,
150         sun_shift_ppo, lon_s_ppo] = ppo_comp_spin.compute_pp(lat_sph, lon_sph,
151                                                             height_sph, el,
152                                                             self.epoch)
153
154        # The model to consider is a sun fixed model, always
155        lon_ppo = lambda_ppo #lon_s_ppo
156    # distance computation
157    x_pp = pp_comp.compute_sph2sph_xyz(self.re + h * 1000, phi_pp,
158                                       lon_pp)

```

```

156     x_ppo = ppo_comp.compute_sph2sph_xyz(self.re + h * 1000, phi_ppo,
157                                         lon_ppo)
158     # North Pole unit vector
159     e_np = np.array([0, 0, 1])
160
161     # Unit vector from earth center to pierce point origin
162     e_ppo = x_ppo / LA.norm(x_ppo)
163     # Unit vector from earth center to rover pierce point
164     e_pp = x_pp / LA.norm(x_pp)
165
166     n = np.cross(e_np, e_ppo) / LA.norm(np.cross(e_np, e_ppo))
167     # Distance npp of the rover pierce point PP to the meridian plane
168     # through PPO
169     n_pp = np.dot(n, e_pp)
170
171     # Angular distance of the PP from the meridian plane through PPO
172     dE_pp_t = np.arcsin(n_pp)
173
174     # Negative longitude of PP in the transversal system
175     e_f = (e_pp - n_pp * n) / LA.norm(e_pp - n_pp * n)
176     n_bar = np.cross(e_f, e_ppo) / LA.norm(np.cross(e_f, e_ppo))
177
178     # consider the sign change of the latitude at the equator
179     if np.dot(n_bar, n) > 0:
180         dN_pp_t = +np.arcsin(LA.norm(np.cross(e_f, e_ppo)))
181     else:
182         dN_pp_t = -np.arcsin(LA.norm(np.cross(e_f, e_ppo)))
183     if iono_type == 'rsi':
184         dN_pp = dN_pp_t
185         dE_pp = dE_pp_t
186     elif iono_type == 'gsi':
187         # it uses a stereographic projection of the transversal system
188         dN_pp = 2 * np.tan(dN_pp_t / 2)
189         dE_pp = 2 * np.tan(dE_pp_t / 2)
190
191     return dN_pp, dE_pp, sf_pp

```

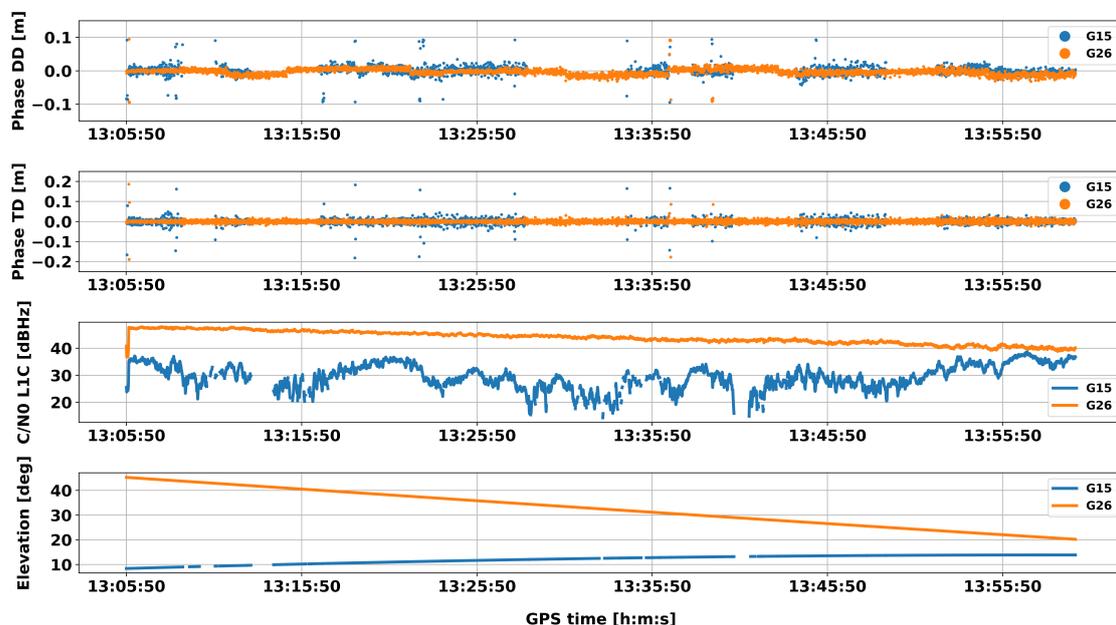
**Listing B.2:** Python methods to compute RSI and GSI corrections from SSRZ messages.



## C. Smartphone-based positioning: additional information

### C.1 Phase DD in Scenario 1: specific satellite examples

To provide additional insight to the quality analysis described in Chapter 3, specific examples of phase DD in Scenario 1 (see Chapter 3) are depicted in Fig. C.1. The figure compares the phase DD to phase TD, C/N0 and elevation variations in time. Some cycle and half cycle-slips are shown, especially for G15 (blue-colored line and dots), which has a lower elevation and C/N0 than G26. Furthermore, Fig. C.1 provides a direct comparison of the phase DD w.r.t. elevation and C/N0 that can be used in the construction of the variance-covariance matrix for the stochastic modeling of the observations.



**Figure C.1:** Analysis of GPS satellites G15 and G26 phase DD in Scenario 1. From top to bottom: phase DD, phase TD, C/N0 and elevation.

### C.2 Smartphone-based positioning using GNSMART

In Listing C.1, the processing options applied with GNSMART 2.0 version of the software (30.06.2020 updated version) are reported for GNSMART users, who want to process GNSS measurements collected with smartphones. The complete information about GNSMART's options can be found in Geo++ GmbH (2019).

A group of measurement-related options are set. While GLONASS and BeiDou observations are excluded

(*-GNSS=RC*) and satellites with single-frequency observations are included in the processing (*-SF*). A 10 deg elevation cut-off is applied using the option *-E=10*. Initial phase (*-OVPHR*) and code (*-OVPR*) observation standard deviations are set for both GPS and Galileo 1C and 5Q signals. Based on the analyses carried out in Chapter 3, an initial code STD has been set for the L1 frequency (1C signal) larger than L5 frequency (5Q signal), i.e. 2 m instead of 1 m.

Furthermore, the process of code multipath parameters has been set to a STD of 0.5 m with a correlation length of 10 s for all (defined by  $\star$  in the options) constellations and signals (-*CDMPROVE*,\*,\*,0.5,10). In the analyses performed in this work, short-baselines between rover and reference station have been considered. Accordingly, the rover and reference station are assumed to experience the same atmospheric conditions using the option -*SBL*.

The ambiguity resolution process is enabled (-*AR=ON*) using a specific Geo++ mode -*AR\_MODE=5*, which takes into account statistical parameters for the ambiguity resolution process. A mis-fixed probability is set using option -*AR\_IRRTUM=0.005*. The larger the probability is set, the faster the ambiguities is solved, but the less reliable is the ambiguity solution. Concerning the partial ambiguity resolution process, the minimum and maximum number of satellites for ambiguity search is set to five (-*AR\_SV\_MN\_MX=5,5*). This option can optimize the ambiguity search through looking for the optimum ambiguity vector for a subset of satellites by creating and scanning the subset in all meaningful combinations. The parameters min (i.e. 5 in this case) and max (i.e. 5 in this case) give the minimum and the maximum numbers of satellites. As an example, if min is set to 6 and max to 8, a loop over all combinations of six, seven and eight satellites is used for the ambiguity search. Furthermore, if single ambiguities are not solved yet, these ambiguities are searched before the search using groups of satellites and receivers (-*AR\_SINGLE*).

```

1 # Measurements options
2 -GNSS=RC
3 +SF
4 -E=10
5 -OVPHR,ROVE,*,*,0.01
6 -OVPR,ROVE,G,1C=2.0
7 -OVPR,ROVE,G,5Q=1.0
8 -OVPR,ROVE,E,1C=2.0
9 -OVPR,ROVE,E,5Q=1.0
10 -CDMP,ROVE,*,*,0.5,10
11 -SBL
12 # AR options
13 -AR=ON
14 -AR_MODE=5
15 -AR_IRRTUM=0.005
16 -AR_SV_MN_MX=5,5
17 -AR_SINGLE

```

**Listing C.1:** GNSMART options used to process GNSS measurements collected with a smartphone. In the options, ROVE stays for the four digits name of the rover (e.g. the smartphone in this case).

### C.3 GNSS raw measurements: Android API classes

The Android API fields and constants mentioned in Chapter 3 are here reported in Table C.2 and Table C.1, respectively.

**Table C.1:** Android Location API - GNSSMeasurements constants.

Constant	Integer value	Hexadecimal value
STATE TOW KNOWN	16384	0x00004000
STATE GLO TOD KNOWN	32768	0x00008000
STATE GAL E1C 2ND CODE LOCK	32768	0x00000800

**Table C.2:** Android Location API - Clock and Measurements fields.

<b>Class</b>	<b>Field</b>	<b>Description</b>
GNSSClock	TimeNanos	GNSS receiver's internal hardware clock value in nanoseconds
GNSSClock	BiasNanos	Clock's sub-nanosecond bias
GNSSClock	FullBiasNanos	Difference between TimeNanos inside the GPS receiver and the true GPS time since 0000Z, 6 January 1980
GNSSClock	DriftNanosPerSecond	clock's Drift in nanoseconds per second
GNSSClock	LeapSecond	Leap second associated with the clock's time
GNSSClock	HardwareClockDiscontinuityCount	Gets count of hardware clock discontinuities
GNSSMeasurement	ConstellationType	Constellation type
GNSSMeasurement	Svid	Satellite ID
GNSSMeasurement	TimeOffsetNanos	Time offset at which the measurement was taken in nanoseconds. The reference time is given by the TimeNanos
GNSSMeasurement	State	Per-satellite sync state. It represents the current sync state for the associated satellite
GNSSMeasurement	ReceivedSvTimeNanos	Received GNSS satellite time, at the measurement time, in nanoseconds
GNSSMeasurement	ReceivedSvTimeUncertaintyNanos	Error estimate (1-sigma) for the received GNSS time, in nanoseconds
GNSSMeasurement	AccumulatedDeltaRangeMeters	Accumulated delta range since the last channel reset, in meters
GNSSMeasurement	AccumulatedDeltaRangeUncertaintyMeters	Accumulated delta range's uncertainty (1-Sigma) in meters
GNSSMeasurement	AccumulatedDeltaRangeState	Accumulated Delta Range state
GNSSMeasurement	Cn0dBHz	CN0 in dBHz



## BIBLIOGRAPHY

---

- Al-Shaery, A., Lim, S., Rizos, C., 2010. Functional models of ordinary kriging for medium range real-time kinematic positioning based on the Virtual Reference Station technique, in: Proceedings of the 23rd International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2010), pp. 2513–2521.
- Alfeld, P., 1984. A trivariate Clough—Tocher scheme for tetrahedral data. *Computer Aided Geometric Design* 1, 169–181.
- Alfeld, P., 1989. Scattered data interpolation in three or more variables, in: Lyche, T., Schumaker, L.L. (Eds.), *Mathematical methods in computer aided geometric design*. Elsevier, pp. 1–33.
- Amidror, I., 2002. Scattered data interpolation methods for electronic imaging systems: a survey. *Journal of electronic imaging* 11, 157–76.
- Andersson, F., Ramírez, A., Wiik, T., Nikitin, V., 2017. Directional interpolation of multicomponent data. *Geophysical Prospecting* 65, 1246–1263. doi:<https://doi.org/10.1111/1365-2478.12478>.
- Appleton, E.V., 1932. *Wireless studies of the ionosphere*. Institution of Electrical Engineers—Proceedings of the Wireless Section of the Institution 7, 257–265.
- Ashby, N., 2003. Relativity in the global positioning system. *Living Reviews in relativity* 6, 1–42.
- Azeem, I., Barlage, M., 2018. Atmosphere-ionosphere coupling from convectively generated gravity waves. *Advances in Space Research* 61, 1931–1941. doi:[10.1016/j.asr.2017.09.029](https://doi.org/10.1016/j.asr.2017.09.029).
- Baelen, J.V., Reverdy, M., Tridon, F., Labbouz, L., Dick, G., Bender, M., Hagen, M., 2011. On the Relationship between Water Vapour Field Evolution and the Life Cycle of Precipitation Systems. *Quarterly Journal of the Royal Meteorological Society* 137, 204–223. doi:[10.1002/qj.785](https://doi.org/10.1002/qj.785).
- Banville, S., Lachapelle, G., Ghoddousi-Fard, R., Gratton, P., 2019. Automated Processing of Low-Cost GNSS Receiver Data, in: Proceedings of Institute of Navigation GNSS+ 2019 conference, Miami, Florida, USA, September 16-20, pp. 3636–3652.
- Bauer, P., Thorpe, A., Brunet, G., 2015. The Quiet Revolution of Numerical Weather Prediction. *Nature* 525, 47–55. doi:[10.1038/nature14956](https://doi.org/10.1038/nature14956).
- Bengtsson, L., Andrae, U., Aspelien, T., Batrak, Y., Calvo, J., de Rooy, W., Gleeson, E., Hansensass, B., Homleid, M., Hortal, M., et al., 2017. The HARMONIE–AROME model configuration in the ALADIN–HIRLAM NWP system. *Monthly Weather Review* 145, 1919–1935.
- Béniguel, Y., 2002. Global Ionospheric Propagation Model (GIM): A propagation model for scintillations of transmitted signals. *Radio Science* 37, 1–14.
- Beran, T., Bisnath, S.B., Langley, R.B., 2004. Evaluation of high-precision, single-frequency gps point positioning models, in: ION GNSS 2004, Long Beach (ION, Virginia 2004), pp. 1893–1901.

- Beutler, G., Bauersima, I., Gurtner, W., Rothacher, M., Schildknecht, T., 1988. Static positioning with the global positioning system (GPS): State of the art, in: Groten, E., Strauß, R. (Eds.), *GPS-Techniques Applied to Geodesy and Surveying*. Springer, Berlin, Heidelberg, pp. 361–380.
- Biagi, L., Grec, F.C., Fermi, A., Negretti, M., 2018. Relative antenna calibration for mass-market GNSS receivers: A case study. doi:10.13140/RG.2.2.19687.09123.
- Bilitza, D., 2001. International reference ionosphere 2000. *Radio Science* 36, 261–275.
- Böhm, J., Niell, A., Tregoning, P., Schuh, H., 2006a. Global Mapping Function (GMF): A new empirical mapping function based on numerical weather model data. *Geophysical Research Letters* 33. doi:https://doi.org/10.1029/2005GL025546.
- Böhm, J., Werl, B., Schuh, H., 2006b. Troposphere mapping functions for GPS and very long baseline interferometry from European Centre for Medium-Range Weather Forecasts operational analysis data. *Journal of Geophysical Research: Solid Earth* 111. doi:https://doi.org/10.1029/2005JB003629.
- Bolmgren, K., Mitchell, C., Bruno, J., Bust, G., 2020. Tomographic Imaging of Traveling Ionospheric Disturbances using GNSS and Geostationary Satellite Observations. *Journal of Geophysical Research: Space Physics* doi:https://doi.org/10.1029/2019JA027551.
- Broadcom, 2020. Broadcom's second-generation dual-frequency GNSS uses new L5 signals. URL: <https://www.broadcom.com/products/wireless/gnss-gps-socs/bcm47765>.
- Brunner, F., 1994. On the deformation of GPS networks, in: *Proc XX FIG Congress*, Melbourne.
- Bruno, J., Darugna, F., Bolmgren, K., Wübbena, J.B., Mitchell, C., Schmitz, M., 2020. Quality analysis of dual-frequency smartphone-based ionospheric TEC measurements: what can be achieved? *Annals of Geophysics* 63, 07.
- Cabinet Office Government of Japan, 2020. Quasi-Zenith Satellite System Interface Specification Centimetre Level Augmentation Service. URL: <https://qzss.go.jp/en/technical/ps-is-qzss/ps-is-qzss.html>.
- Cerretto, G., Tavella, P., Lahaye, F., Mireault, Y., Rovera, D., 2011. PPP using NRCan Ultra Rapid products (EMU): Near real-time comparison and monitoring of time scales generated in time and frequency laboratories, in: *2011 Joint Conference of the IEEE International Frequency Control and the European Frequency and Time Forum (FCS) Proceedings*, IEEE. pp. 1–6.
- Chen, J., Ge, M., Dousa, J., Gendt, G., Ramatschi, M., Wickert, J., Falck, C., Bartsch, M., Merx, A., 2010. Evaluating and improving GNSS real-time positioning using IGS-RTTPP products for tsunami early warning system, in: *IGS Workshop and Vertical Rates Symposium (Newcastle, England 2010)*.
- Chen, X., Landau, H., Vollath, U., et al., 2003. New tools for network RTK integrity monitoring, in: *Proceedings of IONGPS/GNSS*, pp. 1355–1361.
- Cozzens, T., 2020. Broadcom's second-generation dual-frequency GNSS uses new L5 signals. URL: <https://www.gpsworld.com/broadcoms-second-generation-dual-frequency-gnss-uses-l5-signals/>.
- Cressie, N., 1993. *Statistics For Spatial Data*. John Wiley & Sons, New York.
- Critchley-Marrows, J., 2020. The FLAMINGO Experience: Developing a 50 cm positioning service for Smartphones. 4th GNSS Raw Measurements Taskforce Workshop.

- Dabove, P., Di Pietra, V., 2019. Towards high accuracy GNSS real-time positioning with smartphones. *Advances in Space Research* 63, 94–102.
- Darugna, F., Wübbena, J.B., Wübbena, G., Schmitz, M., Schön, S., Warneke, A., 2020. Impact of robot antenna calibration on dual-frequency smartphone-based high-accuracy positioning: a case study using the Huawei Mate20X. *GPS Solutions* 25, 1–12.
- Darugna, F., Wübbena, J., B., 2020. RTCM-SSR Python Demonstrator. URL: <http://github.com/GeoppGmbH/RTCM-SSR-Python-Demonstrator>.
- Davies, K., 1990. Ionospheric radio, *iee electromagn. Waves Ser* 31, 476–483.
- Davis, J.L., Elgered, G., Niell, A.E., Kuehn, C.E., 1993. Ground-based measurement of gradients in the “wet” radio refractivity of air. *Radio Science* 28, 1003–1018.
- De Jonge, P.J., 1998. A processing strategy for the application of the GPS in networks. Netherlands Geodetic Commission. Publications in Geodesy, Delft 46.
- Douša, J., Dick, G., Kačmařík, M., Brožková, R., Zus, F., Brenot, H., Stoycheva, A., Möller, G., Kaplon, J., 2016. Benchmark Campaign and Case Study Episode in Central Europe for Development and Assessment of Advanced GNSS Tropospheric Models and Products. *Atmos Meas Tech* 9, 2989–3008. doi:10.5194/amt-9-2989-2016.
- Douša, J., Václavovic, P., Zhao, L., Kačmařík, M., 2018. New Adaptable All-in-One Strategy for Estimating Advanced Tropospheric Parameters and Using Real-Time Orbits and Clocks. *Remote Sensing* 10(2),232. doi:10.5194/amt-9-2989-2016.
- ECMWF, 2020. ECMWF products. URL: <https://www.ecmwf.int/>.
- Elsobeiey, M., Al-Harbi, S., 2016. Performance of real-time Precise Point Positioning using IGS real-time service. *GPS solutions* 20, 565–571.
- Essen, L., Froome, K., 1951. The refractive indices and dielectric constants of air and its principal constituents at 24,000 Mc/s, IOP Publishing. p. 862.
- Euler, H.J., Schaffrin, B., 1991. On a measure for the discernibility between different ambiguity solutions in the static-kinematic GPS-mode., in: Schwarz, K.P., Lachapelle, G. (Eds.), *Kinematic systems in geodesy, surveying, and remote sensing*. Springer, New York, NY.. Springer, New York, NY., pp. 285–295. International Union of Geodesy and Geophysics, International Association of Geodesy.
- European Global Navigation Satellite Systems Agency (GSA), European Environment Agency and CLAAS, 2018. FARMING BY SATELLITE PRIZE. URL: <http://www.farmingbysatellite.eu/>.
- European GNSS Agency (GSA), 2019. PPP-RTK market and technology report. Technical Report. European GNSS Agency (GSA).
- Fortunato, M., Critchley-Marrows, J., Siutkowska, M., Ivanovici, M.L., 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), IEEE. pp. 1–9.
- Franke, R., 1982. Scattered data interpolation: tests of some methods. *Mathematics of computation* 38, 181–200.
- Gelb, A., 1974. *Applied optimal estimation*. The M.I.T. Press, Massachusetts Institute of Technology, Cambridge, Massachusetts.

- Geng, J., Li, G., 2019. On the feasibility of resolving Android GNSS carrier-phase ambiguities. *Journal of Geodesy* 93, 2621–2635.
- Geng, J., Teferle, F.N., Meng, X., Dodson, A.H., 2011. Towards PPP-RTK: Ambiguity resolution in real-time precise point positioning. *Journal of Geodesy* 47(10), 1664–1673.
- Geo++ GmbH, 2017. Geo++ RINEX Logger. URL: <http://www.geopp.de/logging-of-gnss-raw-data-on-android>.
- Geo++ GmbH, 2019. GNSMART 2 HTML Help.
- Geo++ GmbH, 2020. Geo++ State Space Representation Format (SSRZ). Document Version 0.9.1.3.
- Gil, A., Segura, J., Temme, N.M., 2007. Numerical methods for special functions. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pennsylvania, USA.
- GNSS Raw Measurements Task Force – European GNSS Agency (GSA), 2017. Using GNSS Raw Measurements on Android Devices. Publications Office of the European Union .
- Goad, C., 1985. Precise relative position determination using Global Positioning System carrier phase measurements in a non difference mode, in: Proc. 1st int. symp. on precise positioning with GPS, pp. 347–356.
- Gogoi, N., Minetto, A., Linty, N., DAVIS, F., 2019. A controlled-environment quality assessment of android GNSS raw measurements. *Electronics* 8, 5. doi:10.3390/electronics8010005.
- Google, 2020. GNSS Analysis App. URL: <https://developer.android.com/guide/topics/sensors/gnss>.
- Görres, B., Campbell, J., Becker, M., Siemes, M., 2006. Absolute calibration of GPS antennas: laboratory results and comparison with field and robot techniques. *GPS solutions* 10, 136–145.
- Gregorius, T.L., Blewitt, G., 1999. Modeling weather fronts to improve GPS heights: A new tool for GPS meteorology? *Journal of Geophysical Research: Solid Earth* 104, 15261–15279.
- GSA, 2020. UseGalileo. URL: <https://www.usegalileo.eu/EN/inner.html#data=smartphone>.
- Guerova, G., Jones, J., Dousa, J., Dick, G., Haan, S.D., Pottiaux, E., ..., Vedel, H., 2013. Advanced global navigation satellite systems tropospheric products for monitoring severe weather events and climate (GNSS4SWEC), in: International Colloquium Scientific and Fundamental Aspects of the Galileo Programme, Prague, Czech Republic.
- Hadas, T., Bosy, J., 2015. IGS RTS precise orbits and clocks verification and quality degradation over time. *GPS solutions* 19, 93–105.
- Hardy, R.L., 1971. Multiquadric equations of topography and other irregular surfaces. *Journal of geophysical research* 76, 1905–1915.
- Hauschild, A., 2017. Basic observation equations, in: Teunissen, P.J.G., Montenbruck, O. (Eds.), Springer handbook of Global Navigation Satellite Systems. Springer, pp. 561–582.
- Hauschild, A., Steigenberger, P., 2011. Combined GPS and GALILEO Real-time Clock Estimation with DLR's RETICLE System, in: Proceedings of the 25th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2012), pp. 302–309.
- Helmert, F.R., 1880-1884. Die mathematischen und physikalischen Theorien der höheren Geodäsie. Minerva GmbH, Frankfurt a.M. 1961.

- Heo, Y., Yan, T., Lim, S., Rizos, C., 2009. International standard GNSS real-time data formats and protocols, in: IGNSS Symp.
- Hernández-Pajares, M., Wielgosz, P., Paziewski, J., Krypiak-Gregorczyk, A., Krukowska, M., Stepniak, K., Kaplon, J., Hadas, T., Sosnica, K., Bosy, J., Orus-Perez, R., Monte-Moreno, E., Yang, H., Garcia-Rigo, A., Olivares-Pulido, G., 2017. Direct MSTID mitigation in precise GPS processing. *Radio Science* 52, 321–337. doi:10.1002/2016RS006159.
- Hernández-Pajares, M., Juan, J., Sanz, J., 2006. Medium-scale traveling ionospheric disturbances affecting GPS measurements: Spatial and temporal analysis. *Journal of Geophysical Research: Space Physics* 111, 2009–2012. doi:<https://doi.org/10.1029/2005JA011474>.
- Hernández-Pajares, M., Juan, J.M., Sanz, J., Colombo, O.L., 2000. Application of ionospheric tomography to real-time GPS carrier-phase ambiguities resolution, at scales of 400–1000 km and with high geomagnetic activity. *Geophysical Research Letters* 27(13), 2009–2012.
- Herring, T., 1992. Modeling atmospheric delays in the analysis of space geodetic data. *Proceedings of Refraction of Transatmospheric signals in Geodesy*, eds. JC De Munck and TA Spoelstra, Netherlands Geodetic Commission Publications on Geodesy 36.
- Hines, C.O., 1960. Internal Atmospheric Gravity Waves at Ionospheric Heights. *Canadian Journal of Physics* 38, 1441–1481. doi:10.1139/p60-150.
- Hirokawa, R., Sato, Y., Fujita, S., Miya, M., 2016. Compact SSR Messages with Integrity Information for Satellite Based PPP-RTK Service, in: *Proceedings of the 29th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2016)*, Portland, Oregon, September 2016, pp. 3372–3376.
- Hobiger, T., Jakowski, N., 2017. Atmospheric signal propagation, in: Teunissen, P.J.G., Montenbruck, O. (Eds.), *Springer Handbook of Global Navigation Satellite Systems*. Springer, pp. 165–193.
- Hobiger, T., Shimada, S., Shimizu, S., Ichikawa, R., Koyama, Y., Kondo, T., 2010. Improving GPS Positioning Estimates during Extreme Weather Situations by the Help of Fine-Mesh Numerical Weather Models. *Journal of Atmospheric and Solar-Terrestrial Physics* 72, 262–270.
- Hochegger, G., Nava, B., Radicella, S., Leitinger, R., 2000. A family of ionospheric models for different uses. *Physics and Chemistry of the Earth, Part C: Solar, Terrestrial & Planetary Science* 25, 307–310.
- Hofmann-Wellenhof, B., Lichtenegger, H., Collins, J., 2012. *Global Positioning System: theory and practice*. Springer Science & Business Media.
- Humphreys, T.E., Murrian, M., van Diggelen, F., Podshivalov, S., Pesyna, K.M., 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. doi:10.1109/PLANS.2016.7479707.
- IGS, 2017. IGS antenna readme. URL: [ftp://igs.org/pub/station/general/antenna\\_README.pdf](ftp://igs.org/pub/station/general/antenna_README.pdf).
- International GNSS Service (IGS), 2020. IGS State Space Representation (SSR) Format .
- Irsigler, M., 2008. *Multipath propagation, mitigation and monitoring in the light of Galileo and the modernized GPS*. Ph. D. Thesis, Bundeswehr University, Munich, Germany .
- Jin, S., Wang, J., Park, P.H., 2005. An improvement of GPS height estimations: stochastic modeling. *Earth, Planets and Space* 57, 253–259.

- Jongrujanan, T., Satirapod, C., 2020. Stochastic modeling for VRS network-based GNSS RTK with residual interpolation uncertainty. *Journal of Applied Geodesy* 1 (ahead-of-print).
- Kalafus, R.M., 1996. New RTCM SC-104 standard for differential GNSS. *Navigation* 43, 363–374.
- Kalman, R.E., 1960. A new approach to linear filtering and prediction problems. *J. Basic Eng.* 82, 35–45.
- Kenneth, M., Pesyna, J., Robert, W., Heath, J.R.W., Humphreys, T.E., 2015. Accuracy in the Palm of Your Hand. Centimeter Positioning with a Smartphone-Quality GNSS Antenna. *GPS World* , 16–31.
- Kersten, T., 2014. Bestimmung von Codephasen-Variationen bei GNSS-Empfangsantennen und deren Einfluss auf die Positionierung, Navigation und Zeitübertragung. *Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Leibniz Universität Hannover*, Nr. 315.
- Khodabandeh, A., Teunissen, P.J.G., 2015. An analytical study of PPP-RTK corrections: precision, correlation and user-impact. *Journal of Geodesy* 89(11), 1109–1132.
- Kleijer, F., 2004. Troposphere modeling and filtering for precise GPS leveling. Ph.D. thesis. Delft University of Technology (TUDelft), Delft, Netherlands.
- Klobuchar, J.A., 1987. Ionospheric time-delay algorithm for single-frequency GPS users. *IEEE Transactions on aerospace and electronic systems* , 325–331.
- Krzan, G., Przestrzelski, P., 2016. GPS/GLONASS precise point positioning with IGS real-time service products. *Acta Geodyn. Geomater* 13, 69–81.
- Lachapelle, G., Gratton, P., 2019. GNSS precise point positioning with android smartphones and comparison with high performance receivers, in: *Proceedings of the IEEE Int. Conf. on Signal, Information and Data Processing*, Chongqing, China, December 11-13.
- Langley, R., 1998a. GPS receivers and the observables, in: Teunissen, P.J., Kleusberg, A. (Eds.), *GPS for Geodesy*. Springer, Berlin, pp. 167–171.
- Langley, R.B., 1998b. RTK GPS. *GPS World* 9, 70–76.
- Langley, R.B., Teunissen, P.J., Montenbruck, O., 2017. Introduction to GNSS, in: Teunissen, P.J.G., Montenbruck, O. (Eds.), *Springer Handbook of Global Navigation Satellite Systems*. Springer, pp. 3–23.
- Lassen, H., 1927. Über den einfluss des erdmagnetfeldes auf die fortpflanzung der elektrischen wellen der drahtlosen telegraphie in der atmosphäre. *Elektrische Nachrichten-Technik* 4, 324–334.
- Laurichesse, D., 2011. The CNES Real-time PPP with undifferenced integer ambiguity resolution demonstrator, in: *Proceedings of the ION GNSS 2011*, Portland, Oregon, 19-23 September, pp. 654–662.
- Lawrance, A.J., Lewis, P.A.W., 1977. An exponential moving-average sequence and point process (EMA1). *Journal of Applied Probability* 14(1), 98–113.
- Leandro, R., Santos, M., Langley, R.B., 2006. Unb neutral atmosphere models: development and performance, in: *Proceedings of ION NTM*, pp. 564–73.
- Leick, A., Rapoport, L., Tatarnikov, D., 2015. *GPS satellite surveying*. John Wiley and Sons., Hoboken, New Jersey.

- Leitinger, R., Putz, E., 1988. Ionospheric refraction errors and observables. Atmospheric effects on geodetic space measurements. Monograph 12, 81–102.
- Li, G., Geng, J., 2019. Characteristics of raw multi-GNSS measurement error from Google Android smart devices. *GPS Solutions* 23, 90.
- Li, X., Zhang, X., Ge, M., 2011. Regional reference network augmented precise point positioning for instantaneous ambiguity resolution. *Journal of Geodesy* 85, 151–158.
- Li, X., Zhang, X., Ge, M., 2014. Robustness of GNSS integer ambiguity resolution in the presence of atmospheric biases. *GPS solutions* 18(2), 283–296.
- Liu, S., 2020. Android - Statistics & Facts. URL: <https://www.statista.com/topics/876/android/>.
- Liu, W., Shi, X., Zhu, F., Tao, X., Wang, F., 2019. Quality analysis of multi-GNSS raw observations and a velocity-aided positioning approach based on smartphones. *Advances in Space Research* 63, 2358–2377. doi:10.1016/j.asr.2019.01.004.
- MacMillan, D., Ma, C., 1997. Atmospheric gradients and the VLBI terrestrial and celestial reference frames. *Geophysical research letters* 24, 453–456.
- Mader, G.L., 1999. GPS antenna calibration at the National Geodetic Survey. *GPS solutions* 3(1), 50–58.
- Marini, J.W., 1972. Correction of satellite tracking data for an arbitrary tropospheric profile. *Radio Science* 7, 223–231.
- Massarweh, L., Darugna, F., Psychas, D., Bruno, J., 2019. Statistical investigation of android gnss data: Case study using xiaomi mi 8 dual-frequency raw measurements, in: Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019), Miami, Florida, September 2019, pp. 3847–3861. doi:10.33012/2019.17072.
- Menge, F., 2003. Zur Kalibrierung der Phasenzentrumsvariationen von GPS-Antennen für die hochpräzise Positionsbestimmung. Ph.D. thesis. Leibniz Universität Hannover. Hannover, Germany. doi: <https://doi.org/10.15488/6229>.
- Mervart, L., Weber, G., 2011. Real-time combination of GNSS orbit and clock correction streams using a Kalman filter approach, in: Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2011), pp. 707–711.
- Misra, P., Enge, P., 2006. *Global Positioning System: signals, measurements and performance* second edition. 2 ed., Ganga-Jamuna Press, Lincoln, Massachusetts.
- Montenbruck, O., Gill, E., 2002. *Satellite orbits: models, methods, and applications*. Springer.
- Muellerschoen, R.J., Bar-Sever, Y.E., Bertiger, W.I., Stowers, D.A., 2001. Decimeter accuracy: NASA's global DGPS for high-precision users. *GPS World* 12.
- Nafisi, V., Urquhart, L., Santos, M.C., Nievinski, F.G., Bohm, J., Wijaya, D.D., Schuh, H., Ardalán, A.A., Hobiger, T., Ichikawa, R., et al., 2011. Comparison of ray-tracing packages for troposphere delays. *IEEE transactions on geoscience and remote sensing* 50, 469–481.
- Netthonglang, C., Thongtan, T., Satirapod, C., 2019. GNSS Precise Positioning Determinations Using Smartphones, in: In 2019 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), pp. 401–404.

- Niell, A., 1996. Global mapping functions for the atmosphere delay at radio wavelengths. *Journal of Geophysical Research: Solid Earth* 101, 3227–3246.
- Nottingham Scientific Ltd, 2018. rinex ON. URL: <https://play.google.com/store/apps/details?id=com.eu.nsl.rinexON>.
- Oliver, M., Webster, R., 2014. A tutorial guide to geostatistics: Computing and modelling variograms and kriging. *CATENA* 113, 56–59. doi:<https://doi.org/10.1016/j.catena.2013.09.006>.
- OTHiSaVRoS, 2019. GNSS Android-based Dual-frequency Iono-estimated Precise Point Positioning (GADIP3). URL: <http://play.google.com/store/apps/details?id=com.galileoapp.othisavros.gadip3>.
- Øvstedal, O., 2002. Absolute positioning with single-frequency GPS receivers. *GPS Solutions* 5, 33–44.
- Pace, B., Pacione, R., Sciarretta, C., Bianco, G., 2015. Computation of Zenith Total Delay Correction Fields Using Ground-Based GNSS, in: VIII Hotine-Marussi Symposium on Mathematical Geodesy. doi:[https://doi.org/10.1007/1345\\_2015\\_144](https://doi.org/10.1007/1345_2015_144).
- Parkinson, B.W., Enge, P., Axelrad, P., Spilker Jr, J.J., 1996. *Global positioning system: Theory and applications, Volume II*. American Institute of Aeronautics and Astronautics Inc., Washington DC.
- Pathak, V., Thornwall, S., Krier, M., Rowson, S., Poilasne, G., Desclos, L., 2003. Mobile handset system performance comparison of a linearly polarized GPS internal antenna with a circularly polarized antenna, in: IEEE Antennas and Propagation Society International Symposium. Digest. Held in conjunction with: USNC/CNC/URSI North American Radio Sci. Meeting (Cat. No.03CH37450), pp. 666–669 vol.3. doi:10.1109/APS.2003.1219935.
- Paziewski, J., 2016. Study on desirable ionospheric corrections accuracy for network-RTK positioning and its impact on time-to-fix and probability of successful single-epoch ambiguity resolution. *Advances in Space Research* 57(4), 1098–1111.
- Pérez, J., Agrotis, L., Fernández, J., Garcia, C., Dow, J., 2006. ESA/ESOC real time data processing, in: IGS workshop.
- Pesyna, Jr, K., Heath Jr, R., Humphreys, T.E., 2014. Centimeter Positioning with a Smartphone-Quality GNSS Antenna, in: Proceedings of ION GNSS 2014, Tampa, Florida, September 2014, pp. 1568–1577. doi:10.33012/2019.17072.
- Petit, G., Luzum, B., 2010. IERS conventions (2010). Technical Report. BUREAU INTERNATIONAL DES POIDS ET MESURES SEVRES (FRANCE).
- Petrie, E.J., 2011. A review of higher order ionospheric refraction effects on dual frequency GPS. *Surv Geophys* 32, 197–253. doi:10.1007/s10712-010-9105-z.
- Pirazzi, G., Mazzoni, A., Biagi, L., Crespi, M., 2017. Preliminary Performance Analysis with a GPS+Galileo Enabled Chipset Embedded in a Smartphone, in: Proceedings of 30th International Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2017), Portland, OR, September 2017, pp. 101–115.
- Press, W., Teukolsky, S., Vetterling, W., Flannery, B., 2007. *Numerical Recipes*. 3rd edn. Cambridge.

- Psychas, D., Bruno, J., Massarweh, L., Darugna, E., 2019. Towards Sub-meter Positioning using Android Raw GNSS Measurements, in: Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019), Miami, Florida, September 2019, pp. 3847–3861. doi:10.33012/2019.17077.
- Psychas, D., Verhagen, S., Liu, X., Memarzadeh, Y., Visser, H., 2018. Assessment of ionospheric corrections for PPP-RTK using regional ionosphere modelling. *Measurement Science and Technology* 30(1).
- Rädler, A.T., Groenemeijer, P.H., Faust, E., Sausen, R., Púčík, T., 2019. Frequency of severe thunderstorms across europe expected to increase in the 21st century due to rising instability. *npj Climate and Atmospheric Science* 2, 1–5.
- Realini, E., Caldera, S., Pertusini, L., Sampietro, D., 2017. Precise GNSS positioning using smart devices. *Sensors* 17(10), 2434.
- Reinisch, B., Galkin, I., Belehaki, A., Paznukhov, V., Huang, X., Altadill, D., Buresova, D., Mielich, J., Verhulst, T., Stankov, S., Blanch, E., Kouba, D., Hamel, R., Kozlov, A., Tsagouri, I., Mouzakis, A., Messerotti, M., Parkinson, M., Ishii, M., 2018. Pilot Ionosonde Network for Identification of Traveling Ionospheric Disturbances. *Radio Science* 53, 365–378. doi:10.1002/2017RS006263.
- Rice, R., Plaunt, J., 1971. Adaptive variable-length coding for efficient compression of spacecraft television data. *IEEE Transactions on Communication Technology* 19, 889–897.
- Riddell, A., Moore, M., Hu, G., 2015. Geoscience Australia's GNSS antenna calibration facility: Initial results, in: Proceedings of the IGNSS Symposium, Outrigger Gold Coast, Qld, Australia, pp. 14–16.
- Riley, S., Lentz, W., Clare, A., 2017. On the path to precision—Observations with android GNSS observables, in: Proceedings of ION GNSS 2017, Institute of Navigation, Portland, Oregon, USA, September 25–29, pp. 116–129.
- Riley, W., 2014. The Stable32 program for frequency stability analysis. Hamilton Technical Services, Beaufort, SC 29907.
- Riley, W.J., 2008. Handbook of frequency stability analysis. US Department of Commerce, National Institute of Standards and Technology, Washington DC.
- 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 (Switzerland)* 8. doi:10.3390/electronics8010091.
- Rothacher, M., 2001. Comparison of absolute and relative antenna phase center variations. *GPS Solutions* 4(4), 55–60.
- Rothacher, M., Schaer, S., Mervart, L., Beutler, G., 1995. Determination of antenna phase center variations using gps data., in: Proceedings of the IGS Workshop on Special Topics and New Directions, pp. 205–220.
- Rothacher, M., Schmid, R., 2010. ANTEX: the antenna exchange format, Version 1.4, 15 Sep 2010. URL: <ftp://igs.org/pub/station/general/antex14.txt>.
- RTCM Special Committee No. 104, a. Proposal of new RTCM SSR Messages SSR Stage 1: Galileo, QZSS, SBAS, BDS .
- RTCM Special Committee No. 104, b. Proposal of new RTCM SSR Messages SSR Stage 2: Satellite Phase Biases .

- RTCM Special Committee No. 104, c. Proposal of new RTCM SSR Messages SSR Stage 2: Vertical TEC (VTEC) .
- RTCM Special Committee No. 104, 2016. RTCM Standard 10403.3 differential GNSS (Global Navigation Satellite Systems) Services-Version 3. 104, Radio Technical Commission for Maritime Services, Arlington, Virginia, U.S.A.
- RTKexplorer, 2020. RTKLIB Demo5 b33c. URL: <http://rtkexplorer.com/downloads/rtklib-code/>.
- RTKLIBexplorer, 2019. A first look at the broadcom bcm47755 dual frequency receiver. URL: <https://rtklibexplorer.wordpress.com/2019/08/28/a-first-look-at-the-broadcom-bcm47755-dual-frequency-receiver/>.
- Saastamoinen, J., 1972. Atmospheric correction for the troposphere and stratosphere in radio ranging satellites. The Use of Artificial Satellites for Geodesy, American Geophysical Union 30(1), 247–251. doi:10.1029/GM015p0247.
- Santerre, R., 1989. GPS Satellite Sky Distribution: Impact on The Propagation of Some Important Errors in Precise Relative Positioning. Ph.D. thesis. The University of New Brunswick, Canada.
- Santerre, R., 1991. Impact of GPS Satellite Sky Distribution. Manuscripta Geodaetica 16, 28–53.
- Sapcorda Services GmbH, 2020. Safe Position Augmentation for Real-Time Navigation (SPARTN) Interface Control Document Version 1.8.0. Sapcorda Services GmbH, Scottsdale, Arizona, USA.
- Schaer, S., 1999. Mapping and predicting the earths ionosphere using the Global Positioning System. Ph.D. thesis. University of Bern, Bern, Switzerland.
- Schmitz, M., 2012. RTCM State Space Representation Messages, Status and Plans. PPP-RTK & Open Standards Symposium, March 12-13, 2012, Frankfurt, Germany.
- Schmitz, M., Wübbena, G., Boettcher, G., 2002. Tests of phase center variations of various GPS antennas, and some results. GPS solutions 6(1-2), 18–27. doi:10.1007/s10712-010-9105-z.
- Schön, S., 2007. Affine distortion of small GPS networks with large height differences. GPS solutions 11, 107–117.
- Schön, S., Brunner, F.K., 2008. Atmospheric turbulence theory applied to GPS carrier-phase data. Journal of Geodesy 82, 47–57.
- Schupler, B.R., Allshouse, R.L., A., C.T., 1994. Signal characteristics of GPS user antennas. Navigation 41(3), 276–296.
- Seeber, G., 2003. Satellite Geodesy. 2 ed., Walter de Gruyter, Berlin.
- Shade, S., Madhani, P., 2018. Android GNSS Measurements-Inside the BCM47755, in: Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2018), pp. 554–579.
- Sharma, H. Bochkati, M., Kestel, G., Pany, T., 2019. Researchers achieve 1–2 cm accuracy with commercial smartphone. Inside GNSS 16.
- Shepard, D., 1968. A two-dimensional interpolation function for irregularly-spaced data, in: Proceedings of the 1968 23rd ACM national conference, pp. 517–524.

- Shi, J., Xu, C., Guo, J., Gao, Y., 2014. Local Troposphere Augmentation for Real-Time Precise Point Positioning. *Earth, Planets and Space* 66(1),30. doi:<https://doi.org/10.1186/1880-5981-66-30>.
- Sieradzki, R., Paziewski, J., 2016. Study on reliable GNSS positioning with intense TEC fluctuations at high latitudes. *GPS Solutions* 20(3), 553–563. doi:<https://doi.org/10.1186/1880-5981-66-30>.
- Spilker Jr, J.J., Axelrad, P., Parkinson, B.W., Enge, P., 1996. *Global Positioning System: Theory and Applications, Volume I*. American Institute of Aeronautics and Astronautics Inc., Washington DC.
- Strang, G., Fix, G.J., 1973. *An analysis of the finite element method*. Prentice Hall series in automatic computation, Englewood Cliffs, NJ.
- Takasu, T., 2007. RTKLIB. URL: <http://www.rtklib.com/>.
- Teunissen, P.J., 1993. Least-squares estimation of the integer GPS ambiguities. Invited lecture, section IV theory and methodology, IAG general meeting, Beijing, China.
- Teunissen, P.J., Montenbruck, O., 2017. *Springer Handbook of Global Navigation Satellite Systems*. Springer.
- Teunissen, P.J.G., Khodabandeh, A., 2015. Review and principles of PPP-RTK methods. *Journal of Geodesy* 89(3), 817–240.
- Thayer, G.D., 1974. An improved equation for the radio refractive index of air. *Radio Science* 9, 803–807.
- Tomaščík, J., Saloň, Š., Piroh, R., 2016. Horizontal accuracy and applicability of smartphone GNSS positioning in forests. *Forestry: An International Journal of Forest Research* 90, 187–198.
- Tsugawa, T., Kotake, N., Otsuka, Y., Saito, A., 2007. Medium-scale traveling ionospheric disturbances observed by GPS receiver network in Japan: a short review. *GPS Solutions* 11, 139–144. doi:[10.1007/s10291-006-0045-5](https://doi.org/10.1007/s10291-006-0045-5).
- Van Diggelen, F., Khider, M., Chavan, S., Fu, M., 2020. Updated Google Tools: Logging and Analyzing GNSS Measurements. 4th GNSS Raw Measurements Taskforce Workshop.
- Vana, S., Aggrey, J., Bisnath, S., Leandro, R., Urquhart, L., Gonzalez, P., 2019. Analysis of GNSS correction data standards for the automotive market. *Navigation* 66, 577–592.
- Wallace, J.M., Hobbs, P.V., 2006. *Atmospheric science: an introductory survey*. Academic Press, Elsevier.
- Wang, K., Teunissen, P.J., El-Mowafy, A., 2020a. The ADOP and PDOP: Two Complementary Diagnostics for GNSS Positioning. *Journal of Surveying Engineering* 146, 04020008.
- Wang, S., Li, B., Gao, Y., Gao, Y., Guo, H., 2020b. A comprehensive assessment of interpolation methods for regional augmented PPP using reference networks with different scales and terrains. *Measurement* 150:107067.
- Wanninger, L., 2004. Ionospheric disturbance indices for RTK and network RTK positioning, in: *ION GPS/2004*, Inst. of Navig., Long Beach, Calif.
- Wanninger, L., Heßelbarth, A., 2020. GNSS code and carrier phase observations of a Huawei P30 smartphone: quality assessment and centimeter-accurate positioning. *GPS Solutions* 24, 64. doi:<https://doi.org/10.1007/s10291-020-00978->.

- Weber, L., Tiwari, A., 1995. DGPS architecture based on separating error components, virtual reference stations, and FM subcarrier broadcast, in: Institute of Navigation, Annual Meeting, 51 st, Colorado Springs, CO, pp. 191–200.
- Wilgan, K., Hurter, F., Geiger, A., Rohm, W., Bosy, J., 2017. A comprehensive assessment of interpolation methods for regional augmented PPP using reference networks with different scales and terrains. *Journal of Geodesy* 91(2), 117–1134. doi:<https://doi.org/10.1007/s00190-016-0942-5>.
- Willi, D., 2019. GNSS receiver synchronisation and antenna calibration. Ph.D. thesis. ETH Zurich.
- Willi, D., Lutz, S., Brockmann, E., Rothacher, M., 2020. Absolute field calibration for multi-GNSS receiver antennas at ETH Zurich. *GPS Solutions* 24, 28. doi:<https://doi.org/10.1007/s10291-019-0941-0>.
- Wu, J., Wu, S., Hajj, G., Bertiger, W., Lichten, S., 1993. Effects of antenna orientation on GPS carrier phase. *Manuscripta Geodaetica* 18, 91–98.
- Wu, Q., Sun, M., Zhou, C., Zhang, P., 2019. Precise point positioning using dual-frequency GNSS observations on smartphone. *Sensors (Switzerland)* 19. doi:10.3390/s19092189.
- Wübbena, G., 1991. Zur Modellierung von GPS-Beobachtungen für die hochgenaue Positionsbestimmung. Ph.D. thesis. Wiss. Arb. Universität Hannover Germany, Nr.168.
- Wübbena, G., Bagge, A., Seeber, G., Böder, V., Hankemeier, P., et al., 1996. Reducing distance dependent errors for real-time precise DGPS applications by establishing reference station networks, in: Proceedings of the International Technical Meeting, ION GPS-96, Kansas City, Missouri, 1845-1852, Institute of Navigation. pp. 1845–1852.
- Wübbena, G., Schmitz, M., Bagge, A., 2014. PPP with ambiguity resolution (AR) using RTCM-SSR, in: IGS Workshop (Pasadena, CA, 23–27 June 2014).
- Wübbena, G., 1985. Software developments for geodetic positioning with GPS using TI 4100 code and carrier measurements, in: Proceedings 1st international symposium on precise positioning with the global positioning system. US Department of Commerce., pp. 403–412.
- Wübbena, G., 2007. New GNSS Signals and Ambiguity Resolution, in: Proceedings of EGU General Assembly.
- Wübbena, G., 2012. Geodetic Reference via Precise Point Positioning - RTK. General assembly of CLGE Hanover/Germany, 12-13 October 2012.
- Wübbena, G., Bagge, A., Schmitz, M., 2001a. Network-based techniques for RTK applications, in: GPS JIN, pp. 14–16.
- Wübbena, G., Bagge, A., Schmitz, M., 2001b. RTK networks based on Geo++ GNS-MART-concepts, implementation, results., in: Proceedings of the National Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake, UT, USA., pp. 11–14.
- Wübbena, G., Schmitz, A., Menge, F., Seeber, G., Völkens, C., 1997. A new approach for field calibration of absolute GPS antenna phase center variations. *Navigation* 44(2), 247–255.
- Wübbena, G., Schmitz, M., Bagge, A., 2005. PPP-RTK: Precise Point Positioning Using State-Space Representation in RTK Networks., in: Proceedings of The Institute of Navigation Global Navigation Satellite Symposium 2005, September 13–16, Long Beach, CA, pp. 13–16.

- Wübbena, G., Schmitz, M., Boettcher, G., Schumann, C., 2006. Absolute GNSS antenna calibration with a robot: repeatability of phase variations, calibration of GLONASS and determination of carrier-to-noise pattern, in: Springer, T., Gendt, G., Dow, J.M. (Eds.), Proceedings of the IGS Workshop, Darmstadt, Germany.
- Wübbena, G., Schmitz, M., Menge, F., Böder, V., Seeber, G., 2000. Automated absolute field calibration of GPS antennas in real-time, in: Proceedings of the International Technical Meeting, ION GPS-00, Salt Lake City, Utah, pp. 11–14.
- Wübbena, G., Schmitz, M., Propp, M., 2008. Antenna Group Delay Calibration with the Geo++ Robot-extensions to code observable, in: IGS Analysis Workshop, Poster, June 2-6 MiamiBeach Florida, USA, 2008, pp. 2–6.
- Wübbena, G., Willgalis, S., 2001. State space approach for precise real time positioning in GPS reference networks., in: Proceedings of the Int. Symp. on Kinematic Systems in Geodesy, Geomatics and Navigation (KIS2001), pp. 5–8.
- Xiao, B., Wong, H., Wang, B., Yeung, K.L., 2019. Effect of the screen to metal-frame smartphone antennas., in: Proceedings of the International Workshop on Antenna Technology (iWAT), pp. 29–32. doi:10.1109/APS.2003.1219935.
- Yang, H., Monte-Moreno, E., Hernández-Pajares, M., 2017. Multi-TID detection and characterization in a dense Global Navigation Satellite System receiver network. *Journal of Geophysical Research: Space Physics* 122, 9554–9575. doi:10.1002/2017JA023988.
- Yeh, K.C., KC, Y., CH, L., 1972. Theory of ionospheric waves. Academic Press, London, UK.
- Zhang, H., Yuan, Y., Li, W., Zhang, B., Ou, J., 2017. A Grid-Based Tropospheric Product for China Using a GNSS Network. *Journal of Geodesy* 92, 765–777. doi:https://doi.org/10.1007/s00190-017-1093-z.
- Zhang, L., Yang, H., Gao, Y., Yao, Y., Xu, C., 2018a. Evaluation and analysis of real-time precise orbits and clocks products from different IGS analysis centers. *Advances in Space Research* 61, 2942–2954.
- Zhang, X., Tao, X., Zhu, F., Shi, X., Wang, F., 2018b. Quality assessment of GNSS observations from an android n smartphone and positioning performance analysis using time-differenced filtering approach. *GPS Solutions* 22, 70. doi:10.1007/s10291-018-0736-8.
- Zumberge, J.F., Heflin, M.B., Jefferson, D.C., Watkins, M.M., Webb, F.H., 1997. Precise point positioning for the efficient and robust analysis of GPS data from large networks. *Journal of Geophysical Research: Solid Earth* 102(B3), 5005–5017.
- Zus, F., Bender, M., Deng, Z., Dick, G., Heise, S., Shang-Guan, M., Wickert, J., 2012. A methodology to compute gps slant total delays in a numerical weather model. *Radio Science* 47, 1–15.
- Zus, F., Douša, J., Kačmařík, M., Václavovic, P., Balidakis, K., Dick, G., Wickert, J., 2019a. Improving GNSS Zenith Wet Delay Interpolation by Utilizing Tropospheric Gradients: Experiments with a Dense Station Network in Central Europe in the Warm Season. *Remote Sensing* 11(6), 674. doi:https://doi.org/10.3390/rs11060674.
- Zus, F., Douša, J., Kačmařík, M., Václavovic, P., Dick, G., Wickert, J., 2019b. Estimating the Impact of Global Navigation Satellite System Horizontal Delay Gradients in Variational Data Assimilation. *Remote Sensing* 11(1), 41. doi:https://doi.org/10.3390/rs11010041.



## ACKNOWLEDGEMENTS

---

This thesis was carried out during my employment at Geo++ GmbH in collaboration with the Leibniz Universität Hannover in the context of the TREASURE project. TREASURE has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 722023.

First of all, I would like to thank my supervisor Prof. Dr.-Ing. Steffen Schön for his excellent guidance and dedication during each stage of the thesis, always providing valuable advice. In addition, I would like to thank my supervisors at Geo++ Dr.-Ing Martin Schmitz and Dr. Jannes Wübbena. They were a constant source of knowledge and inspiration in supporting me during each step of the research. They were essential in the development of this work. I am grateful to all my supervisors for encouraging me during all the phases of my dissertation development. Furthermore, I would also like to thank my two co-reviewers, Prof. Dr. Fabio Dovis and Prof. Dr.-Ing. habil. Monika Sester for taking the time to review the work and provide constructive criticism.

I am also grateful to all the researchers involved in the TREASURE project. Special thanks to Karl Bolmgren, Jon Bruno, Lotfi Massarweh, and Dimitrios Psychas. The collaboration with them has been fruitful and a great opportunity to grow both professionally and personally. Moreover, I would like to thank the INGV team (a special mention to Dr. Vincenzo Romano and Dr. Lucilla Alfonsi), Prof. Dr. Sandra Verhagen, and Mr. Frank Boon for hosting me during my secondment at SpacEarth Technology, TUDelft, and Septentrio.

I am extremely grateful to all my colleagues at Geo++. Their continuous willingness to exchange ideas and to be able to have both scientific discussions and having fun were essential factors during the years of my doctorate. A special mention is due to Dr. Christopher Perschke, Dr. Temmo Wübbena, and Andrè Warneke. I would also like to thank Dr. Gerhard Wübbena for being a mentor, always providing valuable advice and ideas.

In addition to them, I would like to thank my dear friends Tim Fischer, Philine Schoen, Alejandro Gomez, and Daniel Battorti, who supported me during all the life phases involved during this period. They supported and encouraged me. I am incredibly grateful.

*Infine, vorrei ringraziare tutta la mia famiglia. Un ringraziamento particolare va ai miei genitori Franco e Patrizia. Li ringrazio molto per il loro costante supporto, per aver creduto sempre in me e tutti i sacrifici sostenuti durante il mio periodo all'estero. Questo lavoro è dedicato a loro.*



---

# CURRICULUM VITAE

---

## PERSONAL INFORMATION

---

Name Francesco Darugna  
Date of birth 21.01.1992 in Feltre (BL), Italy

## WORK EXPERIENCE

---

Sep. 2017 - Aug. 2020 Geo++ GmbH, Garbsen, Germany  
*PhD student*

Aug. 2016 - Mar. 2017 German Space Operations Center (GSOC) - DLR, Weßling, Germany  
*Master's thesis student*

## EDUCATION

---

Since Sep. 2017 Leibniz University of Hannover, Germany  
Institut für Erdmessung  
*PhD candidate*

Sep. 2014 - Apr. 2017 University of Padua, Italy  
Department of Industrial Engineering  
*M. Sc. in Aerospace Engineering*

Sep. 2011 - Sep. 2014 University of Padua, Italy  
Department of Industrial Engineering  
*B. Sc. in Aerospace Engineering*

Sep. 2006 - Jun. 2011 High School G. Galilei, Belluno, Italy  
*High School diploma*