Airborne GNSS PPP Based Pseudolite System

Author:
Huang, Panpan

Publication Date:
2019

DOI:
https://doi.org/10.26190/unsworks/21435

Airborne GNSS PPP Based Pseudolite System

Panpan Huang

A thesis in fulfillment of the requirements for the degree of

Doctor of Philosophy

School of Civil and Environmental Engineering
Faculty of Engineering
UNSW
Sydney NSW 2052, Australia

June 2019

Thesis/Dissertation Sheet

Surname/Family Name	:	**Huang**
Given Name/s	:	**Panpan**
Abbreviation for degree as give in the University calendar	:	**PhD**
Faculty	:	**Faculty of Engineering**
School	:	**School of Civil and Environmental Engineering**
Thesis Title	:	**Airborne GNSS PPP Based Pseudolite System**

Abstract 350 words maximum: (PLEASE TYPE)

Airborne-Pseudolite (A-PL) systems have been proposed to augment Global Navigation Satellite Systems (GNSSs) in difficult service areas. One of the challenges in realising such a system is to determine the precise positions of the A-PLs. This research focuses on improving the A-PL positioning performance based on GNSS Precise Point Positioning (PPP). The main contributions are:

1. A-PL distributed positioning based on real-time GNSS PPP combined with inter-PL range measurements was studied. The short-term predictions of precise orbit and satellite clock corrections were analysed. Simulation tests have demonstrated that the A-PL using GNSS PPP combined with inter-PL range measurements is able to achieve better positioning performance than using the GNSS PPP-only approach. The prediction models for short-term orbit and satellite clock correction predictions can effectively reduce the impact of a disruption of communications on GNSS PPP positioning.

2. To deal with the unmodelled measurement errors for GNSS PPP two model-learning based Kalman filter (KF) algorithms were studied: least-squares support vector machine (LS-SVM) and Gaussian process regression (GPR). These two algorithms were evaluated using both static and kinematic experiments. The results confirm that both algorithms can effectively reduce the effect of unmodelled measurement errors on the positioning performance of GNSS PPP.

3. To realise the optimal integration and stable positioning performance for real-time multi-GNSS PPP, two types of stochastic models were assessed by a static experiment: the a priori stochastic models and the real-time estimated variance methods. The experimental results indicate that the a priori stochastic models based on real-time signal-in-space ranging error (SISRE) and real-time estimated stochastic models could all achieve better performance than the stochastic model based on satellite elevation angle, as used in conventional multi-GNSS PPP.

4. To select the optimal subset of satellites (and therefore measurements) for multi-constellation GNSS, an end-to-end deep learning network for satellite selection was proposed. An experiment was conducted with training and validation data from 220 International GNSS Service (IGS) stations. It was shown that the trained models are capable of selecting most of the contributing satellites with less computational time compared with the brute force approach of satellite selection.

ORIGINALITY STATEMENT

'I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.'

Signed ………………………………………………..............

Date ………………………………………………..............

INCLUSION OF PUBLICATIONS STATEMENT

UNSW is supportive of candidates publishing their research results during their candidature as detailed in the UNSW Thesis Examination Procedure.

Publications can be used in their thesis in lieu of a Chapter if:

- The student contributed greater than 50% of the content in the publication and is the "primary author", i.e. the student was responsible primarily for the planning, execution and preparation of the work for publication
- The student has approval to include the publication in their thesis in lieu of a Chapter from their supervisor and Postgraduate Coordinator.
- The publication is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in the thesis

Please indicate whether this thesis contains published material or not.

☐ *This thesis contains no publications, either published or submitted for publication (if this box is checked, you may delete all the material on page 2)*

☐ *Some of the work described in this thesis has been published and it has been documented in the relevant Chapters with acknowledgement (if this box is checked, you may delete all the material on page 2)*

☒ *This thesis has publications (either published or submitted for publication) incorporated into it in lieu of a chapter and the details are presented below*

CANDIDATE'S DECLARATION

I declare that:

- I have complied with the Thesis Examination Procedure
- where I have used a publication in lieu of a Chapter, the listed publication(s) below meet(s) the requirements to be included in the thesis.

Name	Signature	Date (dd/mm/yy)
Panpan Huang		

Postgraduate Coordinator's Declaration (to be filled in where publications are used in lieu of Chapters)

I declare that:

- the information below is accurate
- where listed publication(s) have been used in lieu of Chapter(s), their use complies with the Thesis Examination Procedure
- the minimum requirements for the format of the thesis have been met.

PGC's Name	PGC's Signature	Date (dd/mm/yy)

For each publication incorporated into the thesis in lieu of a Chapter, provide all of the requested details and signatures required

Details of publication #1: *Full title: Airborne pseudolite positioning based on GNSS PPP* *Authors: Panpan Huang, Chris Rizos, and Craig Roberts* *Journal or book name: Journal of Navigation* *Volume/page numbers:* *Date accepted/ published: 22 April 2019*					
Status	*Published*	×	*Accepted and in press*		*In progress (submitted)*
The Candidate's Contribution to the Work Conceived the study, performed the computations, verified the algorithm with experiments and wrote the paper					
Location of the work in the thesis and/or how the work is incorporated in the thesis: Chapter 3					
Primary Supervisor's Declaration I declare that: • the information above is accurate • this has been discussed with the PGC and it is agreed that this publication can be included in this thesis in lieu of a Chapter • All of the co-authors of the publication have reviewed the above information and have agreed to its veracity by signing a 'Co-Author Authorisation' form.					
Supervisor's name *Chris Rizos*	*Supervisor's signature*		*Date (dd/mm/yy)*		

Details of publication #2: *Full title: Online GPR-KF for GNSS navigation with unmodelled measurement error* *Authors: Panpan Huang, Chris Rizos, and Craig Roberts* *Journal or book name: Lecture Notes in Electrical Engineering* *Volume/page numbers: 497* *Date accepted/ published: 04 May 2018*					
Status	*Published*	×	*Accepted and in press*		*In progress (submitted)*
The Candidate's Contribution to the Work Designed the study, performed the simulations and wrote the paper					
Location of the work in the thesis and/or how the work is incorporated in the thesis: Chapter 4					
Primary Supervisor's Declaration I declare that: • the information above is accurate • this has been discussed with the PGC and it is agreed that this publication can be included in this thesis in lieu of a Chapter • All of the co-authors of the publication have reviewed the above information and have agreed to its veracity by signing a 'Co-Author Authorisation' form.					
Supervisor's name *Chris Rizos*	*Supervisor's signature*		*Date (dd/mm/yy)*		

Details of publication #3: *Full title: Satellite selection with an end-to-end deep learning network* *Authors: Panpan Huang, Chris Rizos, and Craig Roberts* *Journal or book name: GPS Solutions* *Volume/page numbers: 22* *Date accepted/ published: 16 August 2018*

<table>
<tr><td>Status</td><td>Published</td><td>×</td><td>Accepted and in press</td><td></td><td>In progress (submitted)</td><td></td></tr>
<tr><td colspan="7">The Candidate’s Contribution to the Work
Devised and performed the experiments, analysed the data and wrote the paper</td></tr>
<tr><td colspan="7">Location of the work in the thesis and/or how the work is incorporated in the thesis:
Chapter 6</td></tr>
<tr><td colspan="7">Primary Supervisor’s Declaration
I declare that:
• the information above is accurate
• this has been discussed with the PGC and it is agreed that this publication can be included in this thesis in lieu of a Chapter
• All of the co-authors of the publication have reviewed the above information and have agreed to its veracity by signing a ‘Co-Author Authorisation’ form.</td></tr>
<tr><td colspan="2">Supervisor’s name
Chris Rizos</td><td colspan="3">Supervisor’s signature</td><td colspan="2">Date (dd/mm/yy)</td></tr>
</table>

COPYRIGHT STATEMENT

AUTHENTICITY STATEMENT

Abstract

Airborne-Pseudolite (A-PL) systems have been proposed to augment Global Navigation Satellite Systems (GNSSs) in difficult service areas. One of the challenges in realising such a system is to determine the precise positions of the A-PLs. Commonly used methods based on the inverted GNSS principle or the differential GNSS (DGNSS) technique suffers from delay in monitoring the A-PLs, or the requirement for ground reference stations. To address such problems the A-PLs can be positioned using the GNSS Precise Point Positioning (PPP) technique. This thesis focuses on improving the A-PL positioning performance based on GNSS PPP. The main contributions are:

1. A-PL distributed positioning based on real-time GNSS PPP combined with inter-PL range measurements was studied for A-PLs in GNSS challenged areas. The short-term predictions of precise orbit and satellite clock corrections with different prediction models were analysed. Simulation tests have demonstrated that the A-PL using GNSS PPP combined with the processing of inter-PL range measurements is able to achieve better positioning performance than using the GNSS PPP-only approach. The prediction models for short-term orbit and satellite clock correction predictions can effectively reduce the impact of a disruption of communications on GNSS PPP positioning.
2. To deal with the unmodelled measurement errors for GNSS PPP two model-learning based Kalman filter (KF) algorithms were studied: least-squares support vector machine (LS-SVM) and Gaussian process regression (GPR). These two algorithms were evaluated using both static and kinematic experiments. The results confirm that both algorithms can effectively reduce the effect of unmodelled measurement errors on the positioning performance of GNSS PPP.
3. To realise the optimal integration and stable positioning performance for multi-GNSS PPP, two types of stochastic models for real-time multi-GNSS PPP were assessed by a static experiment: the a priori stochastic models based on real-time precise multi-GNSS signal-in-space ranging error (SISRE) and satellite elevation angle, and the real-time estimated variance methods. The experimental results indicate that the a priori stochastic models based on real-time SISRE and real-time estimated stochastic models could all achieve better performance than the stochastic model based on satellite elevation angle, as used in conventional multi-GNSS PPP.

4. To select the optimal subset of satellites (and therefore measurements) for multi-constellation GNSS, an end-to-end deep learning network for satellite selection was proposed. An experiment was conducted with training and validation data from 220 International GNSS Service stations. It was shown that the trained models are capable of selecting most of the contributing satellites with less computational time compared with the brute force approach of satellite selection.

Acknowledgements

First and foremost I would like to thank my supervisor Professor Chris Rizos for the continuous support of my PhD study, and for his patience, motivation, and immense knowledge. He makes me understand how to do research. I appreciate his insightful advices and comments to increase my confidence about the research and always keep me on the right track. I would also like to thank my co-supervisor Dr. Craig Roberts for his assistance and guidance about every experiment I have done on campus and valuable advice for my work.

I would like to acknowledge the China Scholarship Council (CSC) who sponsored my study at the University of New South Wale (UNSW). Thanks also to the China Satellite Navigation Conference (CSNC) council for awarding me the excellent paper award and fully supporting my attendance at the 9th CSNC Conference in Harbin, China, 2018.

I would like to thank all the friends I have met at UNSW, especially Ying Zhu, Jingyi Sun and Qishuo Gao for their support and company to get me through the toughest times of my PhD. Thanks are also given to my friends back in China, Yafan Xu, Fengling Guo and Wen Jiang.

Lastly, I would like to thank my family, my father Shaoping Huang, my mother Jinsheng Li, my brother Yujian Huang, and my sisters Li Huang and Yukun Huang for all their love and support throughout my PhD study. A special thanks to my sister Yukun Huang, who talks with me whenever I am in a bad mood and makes me more optimistic and positive about my study.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Definition
AC	Analysis Centre
A-PL	Airborne Pseudolite
AR	Ambiguity Resolution
BSSD	Between-Satellite Single-Difference
C/N0	Carrier-To-Noise Ratio
CDMA	Code-Division Multiple Access
CIF	Covariance Intersection Filter
DFMC	Dual-Frequency Multi-Constellation
FC	Fusion Centre
FDMA	Frequency Division Multiple Access
FE	Feature Encoding
FFT	Fast Fourier Transform
FOC	Full Operational Capability
GDOP	Geometric Dilution of Precision
GEO	Geostationary Orbits
GFZ	GeoForschungsZentrum Potsdam
GNSS	Global Navigation Satellite System
G-PL	Ground-Based Pseudolite
GPR	Gaussian Processes Regression
HALE	High Altitude Long Endurance
HAPS	High Altitude Platform System
HVCE	Helmert Variance Component Estimation
IF	Ionosphere-free
IFB	Inter-Frequency Biases
IGC	IGS Combined Corrections
IGNSS	Inverted GNSS
IGS	International GNSS Service
IGSO	Inclined Geosynchronous Orbits
IGU	IGS Ultra-Rapid
INS	Inertial Navigation System

IODE	Issue of Data Ephemeris
IOV	In-Orbit Validation
ISB	Inter-System Biases
ISM	Industrial, Scientific, and Medical
ITRF	International Terrestrial Reference Frame
KF	Kalman Filter
LS	Least-Squares
LS-SVM	Least-Squares Support Vector Machine
MEO	Medium Earth Orbit
MGEX	Multi-GNSS Experiment
MLP	Multi-Layer Perceptron
NTRIP	Network Transport of RTCM By Internet Protocol
NWM	Numerical Weather Model
PDOP	Positional Dilution of Precision
PL	Pseudolite
PNT	Positioning, Navigation and Timing
PPP	Precise Point Positioning
RBF	Radial Basis Function
ReLU	Rectified Linear Units
RMSE	Root Mean Squared Error
RTCM	Radio Technical Commission for Maritime Services
RTK	Real-Time Kinematic
RTS	Real-Time Service
SBAS	Space-Based Augmentation System
SCIF	Split Covariance Intersection Filter
SD	Single-Difference
SISRE	Signal-In-Space Ranging Error
SPF	Stratospheric Platforms
STD	Standard Deviation
SVM	Support Vector Machine
TTL	Time Lock Loop

UAV	Unmanned Aerial Vehicle
UPD	Uncalibrated Phase Delay
UT	Unscented Transformation
VCE	Variance Component Estimation
VFE	Voxel Feature Encoding
WGDOP	Weighted GDOP
ZD	Zero-Difference
ZTD	Zenith Tropospheric Delay

Chapter 1 Introduction

1.1 Background and Motivation

A pseudolite (PL) based navigation system can be designed to provide suitably equipped users their position, velocity, or timing, based on PLs transmitting GPS-like signals. Such systems have been proposed as a method of Global Navigation Satellite System (GNSS) *augmentation* in signal-shaded areas, or as an independent backup system operated in areas where satellite signals cannot be observed at all (Kim et al., 2008). One can distinguish between ground-based and airborne pseudolite (G-PL and A-PL) systems. With the PLs mounted on aircraft, airships, or unmanned aerial vehicles (UAVs), the A-PL system has advantages compared to the G-PL system, principally because of reduced “near-far” signal issues, lessened multipath disturbance, larger coverage area and better vertical component observability (Lee et al., 2015). However, to realise such an A-PL system, one of the challenges is determining the precise positions of the A-PLs in a real-time continuous mode (Chandu et al., 2007; Crespillo et al., 2015; Kang et al., 2013; Tsujii et al., 2001). Commonly used methods are based on the “inverted GNSS” principle, with ground stations monitoring the PLs, or the Real-Time Kinematic (RTK) technique with one or more ground reference stations (Lee et al., 2016; Tsujii et al., 2001). However, the inverted GNSS method introduces delay for user positioning, while RTK has stringent requirements that include simultaneous measurements made by both the PL(s) and the reference station(s), and a limitation on the distance between PL(s) and reference station(s).

Precise Point Positioning (PPP) has attracted considerable attention in the GNSS industry and academia, and has been demonstrated to be an efficient tool for applications in geodesy and geodynamics. By using precise GNSS orbits and satellite clock correction products, the PPP technique can, under appropriate conditions, deliver centimetre-level positioning accuracy with high computational efficiency, good long-term repeatability and consistency, and without the requirement of nearby reference station(s). Therefore it is worthwhile investigating a GNSS PPP based A-PL positioning system. However, there are some problems in using real-time GNSS PPP for A-PL positioning. For example, in general GNSS PPP requires a long convergence time to achieve a stable and comparable accuracy to differential GNSS RTK solutions (Li and Zhang, 2014). To improve PPP positioning accuracy and reduce the solution convergence time, a variety of methods based on GNSS PPP augmentation have been proposed; such as adding more observations using multiple frequency and/or multiple GNSS constellations (Tegedor et al., 2014), integrating with a different navigation sensor technology such as an Inertial Navigation System (INS) (Gao et al., 2017; Liu et al., 2016), introducing atmospheric constraints (de Oliveira et al., 2017; Geng et al., 2010; Teunissen and Khodabandeh, 2015), and others.

To enhance PPP performance for A-PL positioning, the combination of inter-PL ranges with GNSS measurements has been investigated. In addition, considering that real-time GNSS PPP relies on the use of precise satellite orbit and clock information, these real-time error corrections may not always be available, especially for A-PLs moving in challenging environments where signal outages may occur. To ensure continuous A-PL positioning based on GNSS PPP it is desirable to predict these error corrections during outages.

A comprehensive measurement model is essential to achieve unbiased estimation for GNSS positioning. However, the GNSS observations suffer from unmodelled errors resulting from multipath, signal interference, etc., which are difficult (even impossible) to model using parametric models. Two commonly used non-parametric algorithms, including least-squares support vector machine (LS-SVM) and Gaussian Processes regression (GPR), were investigated in this research.

Multi-constellation GNSS PPP has been widely studied because of its better performance in terms of positioning accuracy, stability and convergence time compared to GNSS PPP with a single GNSS. To achieve the optimal integration for multi-GNSS PPP, an appropriate stochastic model to appropriately weight observations from different GNSS is required for the positioning algorithm. However, due to limited tracking receiver channels and power consumption, and other issues, it may be not possible, or desirable, to use all satellites in view for multi-GNSS PPP positioning. The optimal subset is generally selected from all possible satellite combinations to minimise either Geometric Dilution of Precision (GDOP) or weighted GDOP (WGDOP), which is difficult to implement in real-time applications due to the time- and power-consuming calculation of the DOP values. An end-to-end deep learning network for satellite selection based on the PointNet and VoxelNet networks is proposed.

1.2 **Thesis Objectives**

The major aim of this research is to study A-PL positioning based on an implementation of the real-time GNSS PPP technique. To improve the positioning performance of GNSS PPP for A-PL positioning this research has focussed on the following topics:

- Investigating the contribution that can be made to positioning solutions of including inter-PL range measurements with real-time GNSS PPP. To process the inter-PL ranges, a distributed algorithm based on a split covariance intersection filter (SCIF) was proposed. Three commonly used means of implementing the SCIF algorithm were analysed. To maintain A-PL positioning accuracy using GNSS PPP when the correction message communication links are disrupted, different short-term prediction models for orbit and clock error corrections were investigated.
- Evaluating two non-parametric model-learning based Kalman filter (KF) algorithms, including LS-SVM and GPR, to deal with the unmodelled errors in GNSS observations. To enable the real-time modelling necessary for these two algorithms, more than one forward step sliding window for input training points was proposed. These two non-parametric model learning algorithms for GNSS PPP were verified with both static and kinematic experiments.
- Assessing two commonly used types of stochastic models for multi-GNSS PPP in real-time, including the a priori stochastic models based on satellite elevation angle and real-time signal-in-space ranging error (SISRE), and real-time estimated stochastic model based on Helmert variance component estimation (HVCE) and real-time variance estimation for pseudorange noise and multipath. Real-time multi-GNSS PPP performance in terms of positioning accuracy, repeatability and estimated zenith tropospheric delay (ZTD) accuracy with the different stochastic models was compared.
- Developing an end-to-end deep learning network for satellite selection using GDOP/WGDOP criteria. The satellite selection problem is converted to a satellite segmentation problem, with specified input channel for each satellite and two class labels, one for selected satellites and the other for those not selected. The proposed satellite segmentation network was validated with training and validation data from 220 International GNSS Service (IGS) stations.

1.3 Contributions of the Research

The following are the contributions of this research:

- A-PL distributed positioning based on real-time GNSS PPP combined with inter-PL range measurements has been investigated. In addition, different prediction models for the short-term predictions of precise orbit and satellite clock corrections have been analysed. A test was conducted to evaluate the A-PL positioning based on GNSS PPP and inter-PL ranges, as well as the performance of error prediction modelling. The results demonstrate the advantage of A-PL positioning using GNSS PPP combined with inter-PL range measurements in terms of accuracy and convergence times compared with the GNSS PPP-only approach.
- Two model-learning based Kalman filter algorithms have been developed to deal with the unmodelled measurement errors in GNSS observations, and evaluated using both static and kinematic experiments.
- Two types of stochastic models for real-time multi-GNSS PPP have been developed and assessed by a static experiment. A comparison of all the stochastic models has confirmed that there is better performance using the a priori stochastic model based on real-time SISRE and real-time variance estimation.
- An end-to-end deep learning network for satellite selection has been developed to select the optimal subset of satellites for multi-constellation GNSS using GDOP and WGDOP criteria.

1.4 Thesis Outline

This thesis consists of seven chapters.

Chapter 1 provides a background and introduction to the hypothetical PL system, and to the GNSS PPP technique. The research motivation and objectives are described, as well as the contributions and outline of the thesis structure.

Chapter 2 reviews generic G-PL and A-PL systems, the GNSS PPP positioning technology and algorithm, two commonly used estimation algorithms for GNSS positioning, strategies to enhance GNSS PPP performance, as well as cooperative positioning algorithms.

Chapter 3 describes the hypothetical A-PL system configuration and proposes an A-PL positioning concept based on real-time GNSS PPP. The inter-PL ranges are used to enhance A-PL positioning performance. These relative measurements are processed using SCIF algorithms to account for cross-correlations of all A-PL estimated states. Three types of SCIF implementation are described and investigated. In addition, the short-term prediction of precise orbit and satellite clock corrections with different prediction models are analysed.

Chapter 4 describes two non-parametric model-learning based KF algorithms that are able to deal with the unmodelled errors in GNSS observations. Independent LS-SVM/GPR models are trained in real-time for all observed satellites using the corresponding measurement residuals calculated in the KF. The GNSS PPP with LS-SVM/GPR for GNSS navigation was evaluated using both static and kinematic experiments.

Chapter 5 investigates two types of stochastic models for real-time multi-GNSS PPP, including the a priori stochastic model and real-time variance estimation methods. A static experiment with data from 14 IGS stations was conducted to assess performance in terms of positioning accuracy, repeatability and estimated ZTD accuracy for all the stochastic models.

Chapter 6 presents an end-to-end deep learning network for multi-GNSS satellite selection using GDOP and WGDOP criteria. The proposed algorithm was evaluated using an experiment with training and validation data from 220 IGS stations.

Chapter 7 summarises the findings of the thesis research, and presents concluding remarks and recommendations for future work.

1.5 List of Publications

1.5.1 Peer-Reviewed Journal Publications

1) **Huang, Panpan**, Chris Rizos, and Craig Roberts, 2018. Satellite selection with an end-to-end deep learning network. GPS Solutions, 22(4), 108

2) **Huang, Panpan**, Chris Rizos, and Craig Roberts. Airborne pseudolite positioning based on GNSS PPP. Journal of Navigation (Published)

1.5.2 **Full paper Peer-Reviewed Conference Publications**

3) **Huang, Panpan**, Chris Rizos, and Craig Roberts, 2018, Online GPR-KF for GNSS navigation with unmodelled measurement error. In: Proceedings of China Satellite Navigation Conference (CSNC) 2018, Harbin, China, 23-25 May, Lecture Notes in Electrical Engineering 497, Springer, Singapore (Excellent Paper Award)

4) **Huang, Panpan,** Chris Rizos, and Craig Roberts. 2016, Inter-Pseudolite Range Augmented GNSS PPP Navigation for Airborne Pseudolite System. Proceedings of IGNSS Symposium, UNSW, Sydney, Australia, presented at IGNSS Symposium, UNSW, Sydney, Australia, 6-8 Dec

Chapter 2 Review of Pseudolite System and Navigation Technologies

2.1 Overview of PL System

PL-based systems transmit GPS/GNSS-like signals, that when range measurements are made and processed, provide users, with suitably equipped receivers, position, velocity, or timing information. By either augmenting GNSS, or operating independently, such systems have been proposed for a number of application areas, such as indoor positioning, deformation monitoring, and where GNSS navigation is precluded due to poor satellite visibility and bad reception conditions, such as in deep open-cut mines, container ports, urban areas, etc (Kim et al., 2008). There are two basic types of PL system identified in the literature: G-PL systems and A-PL systems.

2.1.1 G-PL System

The first G-PL system can be traced back to the late 1970s, established before the launch of the GPS satellites (Harrington and Dolloff, 1976). There were four ground transmitters at fixed locations used to test GPS receivers by transmitting GPS L1 signals with only PL locations carried in the navigation message. In the mid-1980s, the signal specifications for a PL were developed by the Radio Technical Commission for Maritime Services (RTCM) (Stansell Jr, 1986). The PL signals using the GPS frequency band for correction transmission should not interfere with the reception of GPS signals. Message Type 8 was designated for the PL almanac, including PL locations, signal generating code, and health of a number of PLs (Kalafus et al., 1986). With the development of the PL techniques, G-PL systems have been proposed to improve availability, reliability, integrity and positioning accuracy for a number of GNSS-based applications.

In the 1990s, Stanford University proposed to solve the real-time unknown integer cycle ambiguities for aircraft precision landing using GNSS carrier phase by combining differential carrier phase measurements with some G-PLs placed under the aircraft approach path (Cohen et al., 1994; Pervan and Parkinson, 1997). The G-PL was also proposed to improve performance of a GPS-based deformation monitoring system in terms of ambiguity resolution and positioning accuracy by providing additional PL range observations (Dai et al., 2000). Unlike the GPS satellites, when using the fixed G-PLs, some additional factors have to be considered, such as the “near-far” problem, multipath effects, tropospheric delay modelling, G-PLs clock synchronisation, and others. The influence of these factors on the performance improvement with G-PL system has been analysed. In general a modified GNSS receiver is able to track and process the PL signals due to the high-power PLs transmitting signals being on the same L-band frequency as the GNSS signals. However, in principle, PLs can also transmit their signals on different frequencies. "Locata" has been under development as a terrestrial ranging system by the Locata Corporation since 2002 (Rizos et al., 2010). It can be considered an example of a G-PL system except that it has some unique characteristics. For example, the Locata system transmits the ranging signals at frequencies in the 2.4GHz Industrial, Scientific, and Medical (ISM) band. Such a system contains a number of transmitters known as "LocataLites" located within or around a defined service area, which are all time-synchronised to each other. With the synchronised transmission, the "near-far" problem can be solved using tightly controlled pulsed signals. It has been shown that the Locata technology can be used for a wide range of positioning applications with centimetre-level positioning accuracy, such as slow structural displacement monitoring (Choudhury and Rizos, 2010), kinematic maritime navigation by integrating with GNSS PPP (Jiang et al., 2015a), and even precise indoor positioning (Jiang et al., 2015c; Rizos et al. 2010). However, there are some limitations of the G-PL systems, such as low vertical component determination accuracy, susceptibility to multipath effects, and limited service area.

2.1.2 **A-PL System**

To address some of the limitations of G-PL systems, an A-PL system could be used with the PLs mounted on aircraft, airships, or UAVs (Lee et al., 2015; Pallavicini et al., 2001; Raquet et al., 1996; Tsujii et al., 2001). The first application of an A-PL was suggested by Raquet et al. (1996). However, the focus of that research was on A-PL positioning in a "reverse" mode, with some receivers deployed at known points. In the early 2000s, high altitude platforms systems (HAPS) have been suggested for providing differential correction data or telecommunications and surveillance services (Pallavicini et al., 2001). Tsujii et al. (2001) introduced a concept of a GPS augmentation system with PLs mounted on stratospheric platforms (SPF) at an altitude of about 20km to improve the accuracy, availability, and integrity of GPS-based positioning systems across all of Japan (Figure 2.1). The PLs in such a GPS augmentation system are stationary. The PL positions have to be precisely determined due to their significant effect on user navigation. Three schemes based on inverted-GPS to estimate the PLs positions were described: RTK and attitude information, inverted-GPS, and GPS-transceiver.

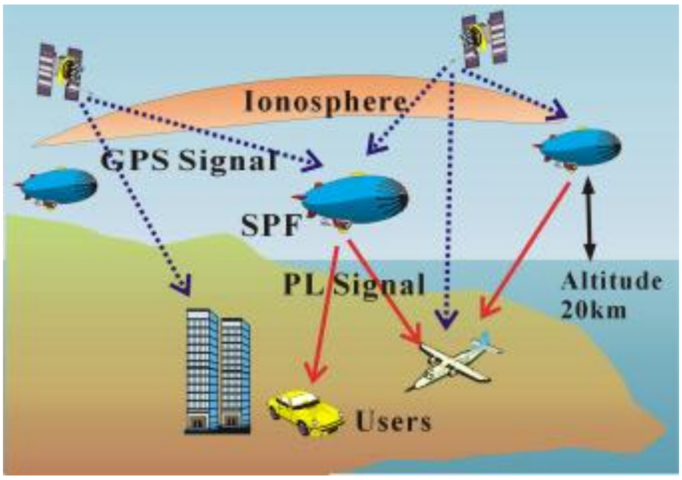


Figure 2.1 Navigation service using PLs on SPF

The European project “HeliNet - Network of High Altitude Long Endurance (HALE) unmanned solar SPF for traffic monitoring, environmental surveillance and broadband services” has also been proposed as an augmentation to GNSS navigation (Pallavicini et al., 2001; Ozimek et al., 2004). The HeliNet not only broadcasts differential corrections generated by a terrestrial reference station, it is also able to transmit GNSS-like ranging signals. The HALE platform is also sometimes referred to as a stratospheric pseudolite (stratolite), which flies at a height of about 17km. The architecture of the proposed stratolite augmentation system is shown in Figure 2.2. To realise such an augmentation system, there must be a capability for stratolite localisation. Two approaches were proposed to determine the stratolite positions by Pallavicini et al. (2001): code-phase DGPS method with a metre-level positioning accuracy, and kinematic carrier phase based techniques with higher accuracy.

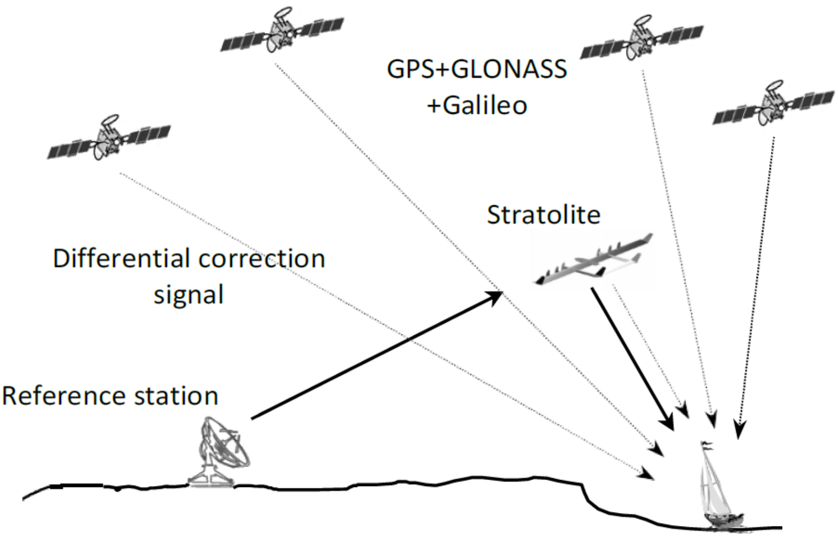


Figure 2.2 Stratolite augmentation system

Park et al. (2008) proposed an airborne transceiver system independent of GPS system. Mobile PLs at high altitude were introduced due to their high visibility and flexibility in deployment, which could move to any location. To position the mobile PLs in real-time, a trilateration method using bidirectional range measurements was proposed, which could be applied to any A-PL platform, including highly manoeuvrable UAVs or stationary SPF.

Hence, one of the challenges to realise an A-PL system for providing navigation services is precise A-PL positioning in real-time. The commonly used "inverted-GPS" methods need monitoring from ground stations, which can cause performance degradation. Chandu et al. (2007) have analysed the effect of movements of the pseudolites and their monitoring time on the accuracy of user positioning. Crespillo et al. (2015) has also identified the factors that could cause user position error, including the positioning and timing precision of the A-PL, their motion and ephemerides transmitting rate. To solve the challenge of A-PL monitoring, Lee et al. (2015) proposed an airborne relay-based positioning system as shown in Figure 2.3.

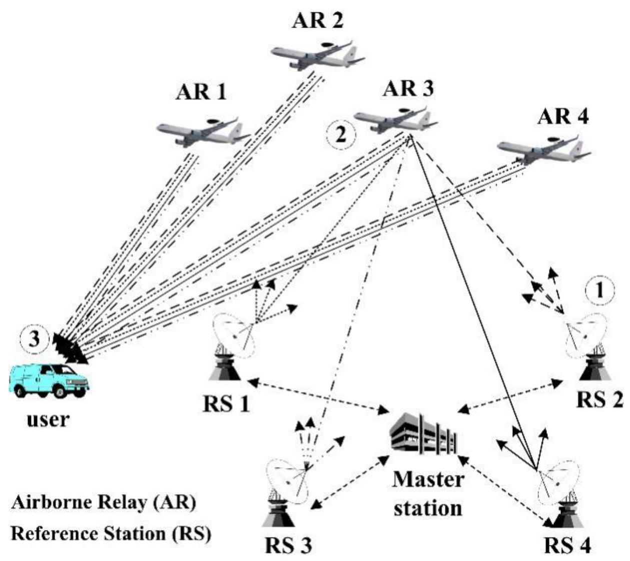


Figure 2.3 Configuration of an airborne relay-based positioning system

The positioning system consists of a master station, several ground reference stations, airborne relays, and the user. The master station is used for providing the time reference for the other ground stations. The reference stations periodically broadcast navigation signals to airborne relays, and the airborne relays then transmit the navigation signals to the user. Once the user tracks and decodes the navigation signals, it calculates both the airborne relays and its own position. However, this system certainly has increased user position estimation complexity and reduced performance robustness, for example, when one or more reference stations cease operations.

2.2 **Overview of GNSS Positioning Technology**

2.2.1 **GNSS Introduction**

GNSS technology has undergone enormous developments over the last three decades. Besides the first operational (and currently being modernised) GNSSs of GPS and GLONASS, two additional new GNSSs are being deployed: Galileo and BeiDou.

The USA's GPS is the first globally operational GNSS. As of December 2018, 31 GPS satellites were operational, of differing "generations", including 12 Block IIR, 7 Block IIR-M and 12 Block IIF. The modernised GPS satellites transmit in three frequency bands: L1 ~ centre frequency 1575.42MHz, L2 ~ centre frequency 1227.6MHz and L5 ~ centre frequency 1176.45MHz. L1 C/A, the legacy ranging signal, is broadcast by all satellites.

The Block IIR-M satellites transmit a new military ranging signal and a more robust civilian signal, known as L2C. L2C is easier for the user to track and is able to deliver improved navigation accuracy. The third civil GPS frequency (L5) at 1176.45MHz was introduced in the most recently launched Block IIF satellites, which could provide signal redundancy, improved signal accuracy and interference rejection. It is planned that there will be 24 Block IIF satellites in orbit by 2021. The third generation GPS satellites – Block III, including Block IIIA and Block IIIF series – incorporating new signals and broadcasting at higher power levels will be launched in the coming years. The first launch of GPS Block IIIA satellites was on December 23, 2018. Block IIIF launches are expected to begin no earlier than 2025 and continue through to year 2034.

The Russian Federation's GLONASS constellation reached Full Operational Capability (FOC) in October 2011 with 24 satellites in orbit, enabling full global coverage. GLONASS satellites transmit navigational signals on two frequency sub-bands (L1 ~ 1602MHz and L2 ~ 1246MHz). Different from Code Division Multiple Access (CDMA) modulating signal technique used by the other GNSSs, Frequency Division Multiple Access (FDMA) technique is employed by GLONASS. To provide better positioning accuracy, multipath resistance and interoperability with GPS and other GNSSs, the GLONASS system is undergoing modernisation with the second generation of GLONASS satellites. L3 CDMA signal centred at 1207.14MHz has already been transmitted after the launch of the first GLONASS K1 satellite. New GLONASS-K satellites deployed in the future will transmit four additional CDMA signals on L1, L2 and L3 bands along the traditional FDMA signals. Two of them are for military uses and the other two are for civil applications.

The EU's Galileo, the third GNSS interoperable with GPS, aims to provide a continuous and precise positioning service under civilian control. There are four frequency bands centred at E5a ~ 1176.45MHz, E5b ~ 1207.14MHz, E6 ~ 1278.75MHz and E1 ~ 1575.42MHz bands transmitted by the Galileo navigation signals. The Galileo has offered initial services since December 15, 2016. Until now there have been four successfully launched In-Orbit Validation (IOV) satellites. Currently Galileo's IOV phase is concluded and its FOC phase is underway and expected to be completed by 2020 (Odijk and Teunissen, 2013). 30 satellites in Medium Earth Orbit (MEO) will be placed for the full operational Galileo constellation.

China’s BeiDou Navigation Satellite System, also known as COMPASS or BeiDou-2, has been providing continuous positioning, navigation and timing (PNT) services for the Asia-Pacific region since December 2012. In 2015, the third generation BeiDou system known as BeiDou-3, began to be deployed so as to ultimately provide global coverage. BeiDou satellites transmit signals in three frequency bands (B1 ~ 1561.1MHz, B2 ~ 1207.14MHz, B3 ~ 1268.52MHz). To date 33 satellites – 6 in geostationary orbits (GEO), 6 in inclined geosynchronous orbits (IGSO) and 21 in MEO – are operational, and 6 other satellites – including 3 in MEO, 1 in GEO and 2 in IGSO – are undergoing testing or commissioning. By the end of 2020, FOC of BeiDou will be achieved with 3 GEO, 3 IGSO and 24 MEO satellites (Odijk et al., 2015).

2.2.2 **GNSS PPP**

GNSS PPP technology has attracted much attention within the GNSS scientific and applications communities. PPP has demonstrated to be an efficient tool for a variety geodetic and geodynamic applications. By using precise satellite orbit and clock correction products, for example from the IGS, the PPP technique can deliver decimetre-level positioning accuracy with lower computational burden, better long-term repeatability, and without the requirements for nearby ground reference stations, in comparison with the well-known RTK technique. To realise PPP in a single receiver, rigorous measurement bias models must be developed. For a satellite s observed by the receiver r for signal C_i on frequency i, the undifferenced pseudorange and carrier phase observations are commonly modelled as:

$$\begin{gathered} P^s_{r,C_i} = \rho^s_r + c(dt_r - dt^s) + T^s_r + \gamma_i I^s_r + B^s_{r,C_i} + e^s_{r,C_i} \\ L^s_{r,C_i} = \rho^s_r + c(dt_r - dt^s) + T^s_r - \gamma_i I^s_r + \lambda_i \left(N^s_{r,C_i} + b^s_{r,C_i}\right) + \varepsilon^s_{r,C_i} \end{gathered} \tag{2.1}$$

where

ρ^s_r denotes the receiver-satellite geometric range;

c is the vacuum speed of light;

dt_r and dt^s are the clock offsets of receiver and satellite (in seconds), respectively;

T^s_r is the slant troposphere delay;

I_r^s is the ionospheric delay for a reference frequency, e.g. L1 for GPS;

$\gamma_i = \frac{f_{C_1}^2}{f_{C_i}^2}$ the frequency-dependent coefficient of ionosphere;

N_{r,C_i}^s is the integer ambiguity;

$B_{r,C_i}^s = B_{r,C_i} - B_{C_i}^s$ is the receiver-satellite hardware bias on frequency i;

$b_{r,C_i}^s = b_{r,C_i} - b_{C_i}^s$ is the receiver-satellite uncalibrated phase delay (UPD);

λ_i is the wavelength for frequency i; and

e_{r,C_i}^s and ε_{r,C_i}^s are measurement noise and multipath of the code and phase measurements, respectively.

The ionospheric delay error can be eliminated, to first order, using an ionosphere-free (IF) measurement combination derived from any dual frequency measurements (Ge et al., 2008):

$$\Phi_{r,IF(C_1,C_2)}^s = \frac{f_{C_1}^2}{f_{C_1}^2 - f_{C_2}^2}\Phi_{r,C_1}^s - \frac{f_{C_2}^2}{f_{C_1}^2 - f_{C_2}^2}\Phi_{r,C_2}^s \tag{2.2}$$

The tropospheric delay is usually modelled as a function of the tropospheric zenith hydrostatic and wet delay (Gao and Shen, 2002):

$$T_r^s = m_H(El_r^s)Z_H + m(El_r^s)(Z_T - Z_H) \tag{2.3}$$

where

Z_T and Z_H are the tropospheric zenith total delay and hydrostatic delay, respectively; and

$m_H(El_r^s)$ and $m(El_r^s)$ are the mapping functions associated with the elevation angle El_r^s of the receiver to the satellite.

Precise orbit and clock corrections for satellites, and antenna phase centre offset and variation information, can be obtained from the IGS. Even with rigorous modelling of different measurement errors and the precise input products, there are still some residual measurement errors resulting from multipath, measurement noises, and unmodelled atmospheric errors, which could deteriorate PPP positioning accuracy (Bisnath and Collins, 2012). In addition, due to the presence of UPD originating in the receiver and satellites, the GNSS PPP requires a long convergence time to achieve the desired positioning accuracy (Ge et al., 2008).

2.2.3 **Estimation Algorithms for GNSS Positioning**

To process the GNSS measurements, there are two commonly used estimation algorithms: least-squares-(LS) based and KF-based estimation methods.

2.2.3.1 LS-based Estimation Algorithm

LS estimation is a common method for solving a system of linear equations where the number of measurements is greater than the number of unknowns. As the GNSS measurement model is nonlinear, it has to be linearised to use LS estimation. Suppose that the linearised relationship between unknown parameter vector $\boldsymbol{x} \in \boldsymbol{R}^n$ and GNSS measurement vector $\boldsymbol{z}$ can be expressed as:

$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{\varepsilon} \tag{2.4}$$

where

$\boldsymbol{H}$ is the design matrix; and

$\boldsymbol{\varepsilon}$ denotes the vector of measurement error with zero mean and covariance $\boldsymbol{R}$.

Based on LS the most probable value of $\boldsymbol{x}$ is obtained by minimising the sum of squared measurement residuals (Marquardt, 1963):

$$\begin{gathered} min: (\boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}})^T(\boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}}) \\ \hat{\boldsymbol{x}} = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{z} \end{gathered} \tag{2.5}$$

where $\boldsymbol{H}^T\boldsymbol{H}$ is a positive definite matrix with a rank of n. LS is an unbiased estimator, i.e.

$$E(\tilde{\boldsymbol{x}}) = \boldsymbol{0}, \quad \tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}} \tag{2.6}$$

where $\tilde{\boldsymbol{x}}$ is the estimated error with variance-covariance of estimated parameters:

$$E[\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^T] = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{R}\boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{H})^{-1} \tag{2.7}$$

It is not optimal to equally treat all the GNSS measurements in the estimation due to different levels of measurement errors. Hence, the weighted LS is generally used, with the "weight" equal to the inverse of covariance $\boldsymbol{R}$ (Strutz, 2016):

$$\begin{gathered}\hat{\boldsymbol{x}} = (\boldsymbol{H}^T\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{R}^{-1}\boldsymbol{z} \\ E[\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^T] = (\boldsymbol{H}^T\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\end{gathered} \tag{2.8}$$

As can be seen from the LS estimation, only the measurement model is involved in the optimal estimated solutions. One limitation of this estimation algorithm is that when the number of available measurements is less than that of the unknown parameters, it is not possible to obtain the solution.

2.2.3.2 KF-based Estimation Algorithm

As opposed to LS estimation, KF-based estimation combines the information from both dynamic and measurement models to generate the optimal estimations. In this study, the unknown parameters considered for the A-PL include position, velocity, acceleration, and other unknowns that need to be estimated for GNSS PPP, which are defined as:

$$\boldsymbol{x} = [\boldsymbol{r} \quad \boldsymbol{v} \quad \boldsymbol{a} \quad dt \quad Tr \quad \boldsymbol{N}] \tag{2.9}$$

where

$\boldsymbol{r} = [x \quad y \quad z]$ is the A-PL position vector;

$\boldsymbol{v} = [v_x \quad v_y \quad v_z]$ is the corresponding velocity vector;

$\boldsymbol{a} = [a_x \quad a_y \quad a_z]$ is the corresponding acceleration vector;

dt is the receiver clock errors;

Tr is the tropospheric zenith total delay; and

$\boldsymbol{N} = [N^{s_1} \quad N^{s_2} \quad \cdots \quad N^{s_m}]$ denotes the PL-to-satellite carrier phase ambiguities, which are preserved as "float" values in the estimation process.

The velocity and acceleration parameters are used to establish the A-PL dynamic model. The A-PL acceleration parameters are modelled as a first-order Gauss-Markov process, while the other unknown parameters are modelled as random walk processes (Jiang et al., 2015b). The discrete A-PL dynamic model can be expressed as:

$$\begin{cases} \boldsymbol{r}_k = \boldsymbol{r}_{k-1} + T \cdot \boldsymbol{v}_{k-1} \\ \boldsymbol{v}_k = \boldsymbol{v}_{k-1} + T \cdot \boldsymbol{a}_{k-1} \\ \boldsymbol{a}_k = \left(1 - \dfrac{T}{T_c}\right) \boldsymbol{a}_{k-1} + \boldsymbol{w}_{\boldsymbol{a}_{k-1}} \\ dt_k = dt_{k-1} + w_{dt_{k-1}} \\ Tr_k = Tr_{k-1} + w_{Tr_{k-1}} \\ \boldsymbol{N}_k = \boldsymbol{N}_{k-1} + \boldsymbol{w}_{\boldsymbol{N}_{k-1}} \end{cases} \tag{2.10}$$

where

k denotes the epoch number; T and T_c are the sampling time and correlation time constant, respectively; and

$w_{\cdot k}$ is the corresponding parameter white noise model.

Equation (2.10) can also be written as:

$$\boldsymbol{x}_k = \boldsymbol{\Phi}_{k|k-1} \boldsymbol{x}_{k-1} + \boldsymbol{w}_k \tag{2.11}$$

where $\boldsymbol{\Phi}_{k|k-1}$ represents the state transition matrix from the state at $(k-1)$th epoch to kth epoch; and $\boldsymbol{\omega}_k$ denotes the process noise with known covariances $\boldsymbol{Q}_k$. The initial estimate of the system state can be obtained with the LS method using pseudorange measurements, which has metre-level accuracy. Similar to LS estimation algorithm, the measurement model has to be linearised as shown in equation (2.4). Then the KF-based algorithm can be recursively implemented in the following two steps (Ristic et al., 2004):

1. Propagation step:

$$\begin{gathered} \hat{\boldsymbol{x}}_{\boldsymbol{k}|\boldsymbol{k}-\boldsymbol{1}} = \boldsymbol{\Phi}_{\boldsymbol{k}|\boldsymbol{k}-\boldsymbol{1}} \hat{\boldsymbol{x}}_{\boldsymbol{k}|\boldsymbol{k}-\boldsymbol{1}} \\ \boldsymbol{P}_{\boldsymbol{k}|\boldsymbol{k}-\boldsymbol{1}} = \boldsymbol{\Phi}_{\boldsymbol{k}|\boldsymbol{k}-\boldsymbol{1}} \boldsymbol{P}_{\boldsymbol{k}-\boldsymbol{1}|\boldsymbol{k}-\boldsymbol{1}} \boldsymbol{\Phi}_{\boldsymbol{k}|\boldsymbol{k}-\boldsymbol{1}}^{\boldsymbol{T}} + \boldsymbol{Q}_{\boldsymbol{k}} \end{gathered} \tag{2.12}$$

2. Update step:

$$\boldsymbol{K_k} = \boldsymbol{P_{k|k-1}} \boldsymbol{H_k^T} \left(\boldsymbol{H_k} \boldsymbol{P_{k|k-1}} \boldsymbol{H_k^T} + \boldsymbol{R_k} \right)^{-1}$$
$$\boldsymbol{\hat{x}_{k|k}} = \boldsymbol{\hat{x}_{k|k-1}} + \boldsymbol{K_k} \left(\boldsymbol{z_k} - \boldsymbol{H_k} \boldsymbol{\hat{x}_{k|k-1}} \right)$$
$$\boldsymbol{P_{k|k}} = (\boldsymbol{I} - \boldsymbol{K_k} \boldsymbol{H_k}) \boldsymbol{P_{k|k-1}} \tag{2.13}$$

where

$\boldsymbol{\hat{x}}_{k|k-1}$ and $\boldsymbol{P}_{k|k-1}$ denote the predicted state vector and its associated covariance matrix, respectively;

$\boldsymbol{I}$ is the identity matrix;

$\boldsymbol{K}_k$ is the Kalman gain matrix; and

$\boldsymbol{\hat{x}}_{k|k}$ and $\boldsymbol{P}_{k|k}$ denote the KF estimated state and its associated covariance matrix, respectively.

2.2.4 **Strategies for Real-time GNSS PPP Performance Improvement**

For real-time positioning applications based on GNSS PPP it is necessary to provide satisfactory performance in terms of positioning accuracy and convergence time. To improve GNSS PPP performance, there have been many strategies proposed, such as PPP with ambiguity resolution (AR) (Ge et al., 2008; Li et al., 2013; Li et al., 2016; Li and Zhang, 2015; Liu et al., 2017; Shi, 2012), PPP with atmospheric constraints (Banville et al., 2014; Elsobeiey and El-Rabbany, 2011; Juan et al., 2012; Li et al., 2014; Lou et al., 2016; Lu et al., 2016; Shi et al., 2014; Zhang et al., 2013), PPP with multi-frequency and multi-GNSSs (Elsobeiey, 2015; Deo and El-Mowafy, 2018; Li et al., 2013; Li et al., 2015; Pan et al., 2017; Wang et al., 2017; Zhou et al., 2018), and PPP integrated with other positioning systems, such as INS and Locata (Du and Gao, 2012; Gao et al., 2016; Gao et al., 2017; Han et al, 2016; Jiang et al., 2015a, 2015b; Li et al., 2017; Zhang et al., 2018; Zhao, 2017).

2.2.4.1 GNSS PPP with AR

Due to the existence of UPD, the zero-difference (ZD) satellite-receiver ambiguity or the between-satellite single-difference (BSSD) ambiguity is not an integer value. The key to successful AR for PPP is to cancel out the UPD component and then to recover the integer value of the estimated ambiguity term.

Ge et al. (2008) found that the UPD can be estimated with high accuracy and reliability due to its comparative stability in time and space. PPP AR can be realised by firstly estimating the fractional parts of the BSSD UPDs using the wide-lane and narrow-lane observables from a global reference network, and correcting SD ambiguities with the obtained BSSD UPDs. It has been demonstrated with static station positioning that PPP AR could achieve improved positioning performance in terms of repeatability and accuracy compared with PPP with float-valued ambiguities. Laurichesse el al. (2009) estimated wide-lane and narrow-lane UPDs of the ZD ambiguities in order to resolve the integer values of the ambiguities. However, the narrow-lane UPD was not directly estimated but merged into clock estimates, which was similar to the decoupled clock model with different pseudorange and carrier phase clocks proposed by Collins et al. (2010). With the estimated carrier phase clocks, it is possible to determine the integer values of the narrow-lane ambiguities. It has also been validated that this AR strategy is able to provide centimetre-level PPP positioning accuracy. In addition, Li et al. (2016) proposed a cascaded orbit error separation method for PPP to separate the effect of the orbit's line-of-sight errors on narrow-lane UPD estimations. The narrow-lane UPDs were modelled with one direction-independent component and three directional-dependent components per satellite. With the new UPD estimation method, more narrow-lane ambiguities were able to be reliably fixed, resulting in more accurate ambiguity-fixed PPP solutions.

2.2.4.2 GNSS PPP with Atmospheric Constraints

Although PPP AR is able to achieve the same positioning accuracy as standard PPP, but with less observation time, both PPP techniques suffer from inaccurately modelled atmospheric delays in the observations, slowing down the convergence of the ambiguities before they can be reliably fixed to integer values. To reduce the convergence time of PPP one popular strategy is to introduce atmospheric constraints.

For example, to eliminate the influence of the high-order ionospheric delay on the convergence of PPP solutions, Elsobeiey and El-Rabbany (2011) suggested rigorous modelling of the second-order ionospheric delay. By using the estimated GPS satellite clock and orbit corrections generated with the second-order ionosphere-corrected raw GPS measurements, it was found that the convergence time of PPP was reduced by 15%. As an alternative to the above strategy, Li et al. (2013) proposed a different PPP processing scheme by treating slant ionospheric delays as unknown parameters. Constraints for the ionospheric delays with a real-time model based on global ionospheric maps and empirical spatial and temporal ionospheric variations were introduced as pseudo-observations. The effect of this proposed approach was to reduce the convergence and observation time for reliable AR, and has been validated in both kinematic and static positioning modes. In addition, the tropospheric delay constrained GNSS PPP method has been proposed for shortening the convergence time and improving positioning accuracy. Lu et al. (2016) have developed a numerical weather model (NWM) constrained PPP using the tropospheric delay parameters derived from the European Centre for Medium-Range Weather Forecasts.

2.2.4.3 GNSS PPP with Multi-Frequency and Multi-GNSS Systems

With the two new GNSS constellations of Galileo and BeiDou, as well as the ongoing modernisation of GPS and GLONASS, multi-frequency and multi-constellation PPP is a possibility, and is another popular way to improve the positioning performance of PPP (Afifi and El-Rabbany, 2015a, 2015b; Rabbou and El-Rabbany, 2015). Such an integration of measurements from multiple GNSSs could introduce additional biases, such as time offsets and inter-system biases (ISB) due to the different frequencies and signal structures used for each GNSS system (Li et al., 2015). In addition, because FDMA is used for GLONASS satellites, inter-frequency biases (IFB) are also introduced into the models of the measurements. Due to the fact that these biases are usually stable during a typical observation period of a few hours or so, they are often lumped with the ambiguities. As a result, these biases, coupled with code observation biases and noise, could result in a lengthening of the convergence time for the solution to reach centimetre-level positioning accuracy. To reduce the influence of these biases on multi-GNSS PPP, they have to be treated carefully, either by correcting the observations or by estimating them in the measurement processing.

Geng and Bock (2013) have proposed triple-frequency GPS PPP with rapid AR, and simulation results have indicated that the AR for the triple-frequency case was more efficient than that for dual-frequency PPP. Cai and Gao (2013) integrated GPS and GLONASS measurements into a PPP algorithm and reported an improvement of about 24% in positioning accuracy. Based on the method proposed by Cai and Gao (2013), Li and Zhang (2014) proposed a BSSD PPP model to combine raw dual-frequency carrier phase measurements of GPS and GLONASS and showed that much less convergence time was required than for the case of GPS alone. Li et al. (2015) performed real-time multi-GNSS PPP combining GPS, GLONASS, BeiDou and Galileo measurements. The authors found that multi-GNSS PPP had faster convergence and better positioning accuracy solutions than GPS-only PPP due to the increased number of observed satellites and the improved positional dilution of precision (PDOP) values.

2.2.4.4 GNSS PPP Integrated with Other Positioning Systems

To improve the reliability and robustness of GNSS PPP positioning, researchers have proposed an integrated multi-sensor navigation system. The integration of GNSS PPP and INS has been widely studied. Two forms of integration are described in the literature: loosely-coupled and tightly-coupled integration. For loosely-coupled integration, the positions estimated by the GNSS PPP are combined with the INS-generated solutions. On the other hand, in the case of tightly-coupled integration, raw GNSS and inertial measurements are jointly processed in a common parameter adjustment. Tightly-coupled integration is preferred due to its possibility to update the integrated solution for scenarios where there are limited GNSS observations (Falco et al., 2017). There have been many studies that have demonstrated that an integrated GNSS PPP and INS system is able to provide better positioning performance than standalone GNSS PPP.

Gao et al. (2017) studied the challenge of tightly-coupled integration of multi-GNSS PPP and INS, and their solution has been validated with a land vehicle experiment. Liu et al. (2016) proposed a tightly-coupled ambiguity-fixed PPP/INS integration. Such an integration was able to achieve stable centimetre-level positioning after the first-fixed solution. In addition, INS could assist rapid re-convergence and re-fixing following GNSS signal outages. Zhang et al. (2018) analysed the different levels of performance improvement in INS-aided ambiguity re-fixing for kinematic GNSS PPP with different periods of GNSS signal outages. It was also found that INS information could bridge the data gaps and achieve fast ambiguity re-fixing. To further exploit the integration model, Rabbou and El-Rabbany (2015) proposed a new processing scheme for GPS PPP and INS with both undifferenced and BSSD ionosphere-free (IF) linear combinations of pseudorange and carrier phase measurements. It was shown that decimetre-level positioning accuracy could be achieved with both undifferenced and BSSD integrated systems. To further improve the robustness of the integration of GNSS and INS with respect to the error accumulation problem of an INS and the susceptibility of GNSS signals to interference, Jiang et al. (2015a) proposed a triple-integration of GNSS, INS and Locata. It has been shown that the triple-integrated system was able to continuously provide centimetre-level accuracy even when one of the subsystems became unavailable.

2.3 **Cooperative Positioning Algorithm**

For positioning of vehicular networks in GNSS-challenged environments (where GNSS signals suffer from degradation or blockage), GNSS positioning performance in terms of accuracy and availability will deteriorate. To improve GNSS positioning performance, inter-vehicle range-based cooperative positioning methods have been proposed (Alam et al., 2011; Caceres et al., 2011; Rui and Chitre, 2010; Yao et al., 2011). By sharing the positioning information among the vehicular network, multiple vehicles are able to perform localisation cooperatively. In this study, the inter-PL ranges are proposed to be measured by each A-PL to enhance A-PL positioning based on GNSS PPP. Each A-PL is able to transmit its positions and receive positions of the observed A-PLs. To process the inter-PL ranges, cross-correlations among the estimated states introduced during the measurement processing when using KF-based estimation methods have to be accounted for. There are two commonly used cooperative algorithms to address this problem: centralised and decentralised approaches.

2.3.1 **Centralised Algorithm**

A straightforward method to keep track of the cross-correlation terms among all the estimated states is to employ a centralised processing scheme (Bailey et al., 2011; Goel et al., 2017; Howard et al., 2002; Mensing and Nielsen, 2010). The centralised algorithm is often implemented with a master vehicle or a fusion centre (FC) or central processor, gathering and processing information from all vehicles in the network at every time instant. Then the FC broadcasts back the estimated positions to each vehicle. Assuming that there are N vehicles in the network, the maximum number of inter-vehicle measurements per epoch can reach $O(N^2)$. Therefore, when the nonlinear KF-based estimation methods are used to process each measurement, such as the EKF, the computational cost is $O(N^2)$ leading to an overall cost of $O(N^4)$ per time step (Carrillo-Arce et al., 2013). With an increase in the number of vehicles in the network, the high computational complexity makes the centralised algorithm difficult to implement for real-time applications. In addition, all-to-all communications at each time step is required. Each vehicle has to transmit its measurement information to the FC, which results in a communication cost of $O(N)$ for each time epoch. However, it may not be possible for a single FC to communicate with all the vehicles when the communication bandwidth is constrained. Moreover the centralised algorithm is susceptible to a single point of failure.

2.3.2 Decentralised Algorithm

To address the shortcomings of centralised algorithms, decentralised algorithms can be used instead (Leung et al., 2010; Li and Nashashibi, 2013; Kia et al., 2014; Nerurkar et al., 2009; Vásárhelyi et al., 2014; Wanasinghe et al., 2014). These algorithms can be divided into two categories: one is the class of tightly-coupled methods, i.e. centralised-equivalent algorithms; and the other is the class of loosely-coupled decentralised methods (Kia et al., 2016). For the tightly-coupled methods, the computational load of the centralised algorithm is distributed among the entire network and the cross-correlations can be accurately tracked. However, such algorithms still suffer from relatively high computational, communications and data storage costs (Kia et al., 2016; Nerurkar et al., 2009; Roumeliotis and Bekey, 2002). In loosely-coupled decentralised algorithms, the exact cross-correlations are not maintained. To obtain estimation consistency, the commonly used covariance intersection filter (CIF) and SCIF algorithms are reviewed below (Carrillo-Arce et al., 2013; Goel et al., 2017; Li and Nashashibi, 2013; Wanasinghe et al., 2014).

2.3.2.1 CIF Algorithm

Consider a pair of state estimates $\{\hat{\boldsymbol{x}}_i, \boldsymbol{P}_i\}$, where $i = \{1,2\}$, and $\hat{\boldsymbol{x}}_i$ and $\boldsymbol{P}_i$ represent the estimated state vector and the corresponding error covariance matrix, respectively. If the two estimates are independent from one another, the fused state and its corresponding error covariance can be derived from the general Kalman filter relations (Julier and Uhlmann, 1997):

$$\boldsymbol{P} = \left(\boldsymbol{P}_1^{-1} + \boldsymbol{P}_2^{-1}\right)^{-1}$$
$$\hat{\boldsymbol{x}} = \boldsymbol{P}\left(\boldsymbol{P}_1^{-1}\hat{\boldsymbol{x}}_1 + \boldsymbol{P}_2^{-1}\hat{\boldsymbol{x}}_2\right) \tag{2.14}$$

Otherwise, when there is correlation between these two estimates, the fused results in equation (2.14) would be inconsistent and there would be optimistic measures of their quality (Carrillo-Arce et al., 2013). To deal with the correlation of two estimates, the CIF uses a convex combination of the means and covariances of the two estimates, which theoretically yields a consistent estimate for any degree of correlation between the two input estimates:

$$\boldsymbol{P}^{-1} = \omega\boldsymbol{P}_1^{-1} + (1-\omega)\boldsymbol{P}_2^{-1}$$
$$\hat{\boldsymbol{x}} = \boldsymbol{P}[\omega\boldsymbol{P}_1^{-1}\hat{\boldsymbol{x}}_1 + (1-\omega)\boldsymbol{P}_2^{-1}\hat{\boldsymbol{x}}_2] \tag{2.15}$$

where $\omega \in [0,1]$. The update can be optimised with respect to different criteria, such as minimising the trace or the determinant of $\boldsymbol{P}$. One problem when using the CIF algorithm is that it treats all estimates as being correlated.

2.3.2.2 SCIF Algorithm

The SCIF algorithm, on the other hand, is able to maintain the known independent information in the estimates. Consider two vehicles in a network of N moving vehicles with states at time instant k denoted as $\boldsymbol{x}_k^i$ and $\boldsymbol{x}_k^j$, and their corresponding covariance matrices $\boldsymbol{P}_k^i$ and $\boldsymbol{P}_k^j$. The range measurement between these two vehicles can be expressed as (Carrillo-Arce et al., 2013):

$$\boldsymbol{z}_k^{i,j} = \boldsymbol{H}_k^{i,j}\left(\boldsymbol{x}_k^i - \boldsymbol{x}_k^j\right) + \boldsymbol{n}_k^{i,j} \tag{2.16}$$

where

$\boldsymbol{z}_k^{i,j}$ denotes the measurement vector;

$\boldsymbol{H}_k^{i,j}$ is an orthogonal matrix which enables the state of either vehicle to be expressed in terms of the other vehicle’s state; and

$\boldsymbol{n}_k^{i,j}$ is the zero-mean white Gaussian measurement noise vector with covariance matrix $\boldsymbol{R}_k^{i,j}$.

The ith vehicle is assumed to be the one of interest and has the prediction state vector $\hat{\boldsymbol{x}}_{k|k-1}^{i}$ and corresponding covariance matrix $\boldsymbol{P}_{k|k-1}^{i}$ derived using standard KF-based methods. When the measurement from the jth vehicle is received by the ith vehicle the information on state $\hat{\boldsymbol{x}}_{k|k-1}^{j}$ and covariance $\boldsymbol{P}_{k|k-1}^{j}$ of the jth vehicle are also shared with the ith vehicle. With the received information the ith vehicle can directly generate an estimate (and corresponding covariance) from the inter-vehicle measurement:

$$\begin{cases} \hat{\boldsymbol{x}}_k^{i*} = \hat{\boldsymbol{x}}_{k|k-1}^{j} + \left(\boldsymbol{H}_k^{i,j}\right)^T \boldsymbol{z}_k^{i,j} \\ \boldsymbol{P}_k^{i*} = \boldsymbol{P}_{k|k-1}^{j} + \boldsymbol{H}_k^{i,j} \boldsymbol{R}_k^{i,j} \left(\boldsymbol{H}_k^{i,j}\right)^T \end{cases} \tag{2.17}$$

At this instant two pairs of state and corresponding covariance of the ith vehicle, one from the time update and one from the measurement update, are derived. The fusion of these two pairs of estimates can then be performed using equation (2.15). However, the measurement cannot always be expressed as in equation (2.16) with an orthogonal matrix $\boldsymbol{H}_k^{i,j}$, which is often in a nonlinear form. When $\boldsymbol{H}_k^{i,j}$ is in any other form it is not possible to directly calculate $\hat{\boldsymbol{x}}_k^{i*}$ in equation (2.17). To use the inter-vehicle measurement to update the propagated state and covariance, a transformation of the measurement needs to be carried out. $\boldsymbol{x}_{k,m}^{i} = \boldsymbol{H}_k^{i,j}\boldsymbol{x}_k^{i} + \boldsymbol{n}_k^{i,j}$ can be derived from equation (2.16) using the transformation (Li and Nashashibi, 2013):

$$\boldsymbol{x}_{k,m}^{i} = \boldsymbol{H}_k^{i,j}\hat{\boldsymbol{x}}_{k|k-1}^{j} + \boldsymbol{z}_k^{i,j} \tag{2.18}$$

The covariance corresponding to $\boldsymbol{x}_{k,m}^{i}$ is given by $\boldsymbol{P}_{k,m}^{i} = \boldsymbol{H}_k^{i,j}\boldsymbol{P}_{k|k-1}^{j}\left(\boldsymbol{H}_k^{i,j}\right)^T + \boldsymbol{R}_k^{i,j}$. The fusion can be performed after the measurement update in KF methods as follows:

$$\begin{gathered} \boldsymbol{K}_k^{*} = \boldsymbol{P}_{k|k-1}^{i}\left(\boldsymbol{H}_k^{i,j}\right)^T \left[\boldsymbol{H}_k^{i,j}\boldsymbol{P}_{k|k-1}^{i}\left(\boldsymbol{H}_k^{i,j}\right)^T + \boldsymbol{P}_{k,m}^{i}\right]^{-1} \\ \hat{\boldsymbol{x}}_{k|k}^{i,*} = \hat{\boldsymbol{x}}_{k|k-1}^{i} + \boldsymbol{K}_k^{*}\left(\boldsymbol{x}_{k,m}^{i} - \boldsymbol{H}_k^{i,j}\hat{\boldsymbol{x}}_{k|k-1}^{i}\right) \\ \boldsymbol{P}_{k|k}^{i,*} = \left(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k^{i,j}\right)\boldsymbol{P}_{k|k-1}^{i} \end{gathered} \tag{2.19}$$

where $\hat{\boldsymbol{x}}_{k|k}^{i,*}$ and $\boldsymbol{P}_{k|k}^{i,*}$ are the updated state vector of the ith vehicle and its corresponding covariance matrix without considering the correlation introduced by the jth vehicle. Then, as in equation (2.17), the propagated and updated states can be fused using equation (2.15). This fusion can also be performed during the measurement update in a standard KF-based algorithm (Goel et al., 2017):

$$
\begin{gathered}
\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}^{i,sca}\left(\boldsymbol{H}_k^{i,j}\right)^T\left[\boldsymbol{H}_k^{i,j}\boldsymbol{P}_{k|k-1}^{i,sca}\left(\boldsymbol{H}_k^{i,j}\right)^T + \boldsymbol{P}_{k,m}^{i,sca}\right]^{-1} \\
\widehat{\boldsymbol{x}}_{k|k}^i = \widehat{\boldsymbol{x}}_{k|k-1}^i + \boldsymbol{K}_k\left(\boldsymbol{x}_{k,m}^i - \boldsymbol{H}_k^{i,j}\widehat{\boldsymbol{x}}_{k|k-1}^i\right) \\
\boldsymbol{P}_{k|k}^i = \left(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k^{i,j}\right)\boldsymbol{P}_{k|k-1}^{i,sca}
\end{gathered} \tag{2.20}
$$

where $\boldsymbol{P}_{k|k-1}^{i,sca} = \boldsymbol{P}_{k|k-1}^i/\omega$ and $\boldsymbol{P}_{k,m}^{i,sca} = \boldsymbol{P}_{k,m}^i/(1-\omega)$ are two scaled covariance matrices. Another way of performing the fusion during the measurement update is given by (Mokhtarzadeh and Gebre-Egziabher, 2014):

$$
\begin{gathered}
\boldsymbol{P}_{k|k}^i = \left[\omega\left(\boldsymbol{P}_{k|k-1}^i\right)^{-1} + (1-\omega)\left(\boldsymbol{H}_k^{i,j}\right)^T\left(\boldsymbol{P}_{k,m}^i\right)^{-1}\boldsymbol{H}_k^{i,j}\right]^{-1} \\
\boldsymbol{K}_k = (1-\omega)\boldsymbol{P}_{k|k}^i\left(\boldsymbol{H}_k^{i,j}\right)^T\left(\boldsymbol{P}_{k,m}^i\right)^{-1} \\
\widehat{\boldsymbol{x}}_{k|k}^i = \widehat{\boldsymbol{x}}_{k|k-1}^i + \boldsymbol{K}_k\left(\boldsymbol{x}_{k,m}^i - \boldsymbol{H}_k^{i,j}\widehat{\boldsymbol{x}}_{k|k-1}^i\right)
\end{gathered} \tag{2.21}
$$

These two ways of accounting for the correlations have less computational cost than the one with fusion performed after the measurement update. When the measurement used is in the form of inter-vehicle ranges, the correlated covariance components are limited to the states involved in the inter-vehicle ranges. To incorporate known independent information in the estimates based on the above algorithms, they need to be modified using the following relations (Li and Nashashibi 2013):

$$
\begin{gathered}
\boldsymbol{P}_1 = \boldsymbol{P}_{1d}/\omega + \boldsymbol{P}_{1i} \\
\boldsymbol{P}_1 = \boldsymbol{P}_{2d}/(1-\omega) + \boldsymbol{P}_{2i} \\
\boldsymbol{P}^{-1} = \boldsymbol{P}_1^{-1} + \boldsymbol{P}_2^{-1} \\
\widehat{\boldsymbol{x}} = \boldsymbol{P}\left[\boldsymbol{P}_1^{-1}\widehat{\boldsymbol{x}}_1 + \boldsymbol{P}_2^{-1}\widehat{\boldsymbol{x}}_2\right]
\end{gathered} \tag{2.22}
$$

where $\boldsymbol{P}_{1i}$ and $\boldsymbol{P}_{2i}$ are covariance matrix components with known absolute independence and $\boldsymbol{P}_{1d}$ and $\boldsymbol{P}_{2d}$ are two correlated covariance matrix components. Therefore to use the SCIF algorithm, the scaled covariances in equation (2.20) and (2.21) have to be added to the independent component before the measurement update. For example, equation (2.20) can be changed to:

$$
\begin{aligned}
\boldsymbol{K}_k &= \boldsymbol{P}^{i,all}_{k|k-1}\left(\boldsymbol{H}^{i,j}_k\right)^T\left[\boldsymbol{H}^{i,j}_k\boldsymbol{P}^{i,all}_{k|k-1}\left(\boldsymbol{H}^{i,j}_k\right)^T + \boldsymbol{P}^{i,all}_{k,m}\right]^{-1} \\
\hat{\boldsymbol{x}}^i_{k|k} &= \hat{\boldsymbol{x}}^i_{k|k-1} + \boldsymbol{K}_k\left(\boldsymbol{x}^i_{k,m} - \boldsymbol{H}^{i,j}_k\hat{\boldsymbol{x}}^i_{k|k-1}\right) \\
\boldsymbol{P}^i_{k|k} &= \left(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}^{i,j}_k\right)\boldsymbol{P}^{i,all}_{k|k-1}
\end{aligned}
\tag{2.23}
$$

where $\boldsymbol{P}^{i,all}_{k|k-1} = \boldsymbol{P}^{i,sca}_{k|k-1} + \boldsymbol{P}^{i,id}_{k|k-1}$ and $\boldsymbol{P}^{i,all}_{k,m} = \boldsymbol{P}^{i,sca}_{k,m} + \boldsymbol{P}^{i,id}_{k,m}$ consist of two covariance matrix components $\boldsymbol{P}^{i,sca}$ and $\boldsymbol{P}^{i,id}$ representing the correlated and independent covariance matrix components, respectively.

Chapter 3 A-PL Distributed Positioning Based on Real-time GNSS PPP

3.1 Introduction

A-PL systems have been proposed as a means of augmenting GNSS in difficult areas where GNSS-only navigation cannot be guaranteed, due to signal blockages, signal jamming, etc. The A-PLs are generally configured to be either station-keeping (that is, hovering or keeping a near-stationary position in the sky) or flying around the service area (for example, following some pre-defined trajectory) (Crespillo et al., 2015; He et al., 2016; Lee et al., 2018). To realise such an A-PL system, one of the challenges to be addressed is to precisely determine the positions of the A-PLs on a continuous basis. A number of positioning methods based on GNSS have been proposed. For example, based on the "Inverted GNSS" (IGNSS) principle, the A-PLs could be monitored by a network of ground stations (Tsujii et al., 2001). To accurately position the A-PL in a real-time continuous mode, the RTK technique with one or more reference stations would be typically used (Lee et al., 2016). As an alternative approach, real-time PPP has also been proposed for continuous positioning (Gross et al., 2016). This method does not have stringent requirements for simultaneous measurements made by the A-PLs and ground-based reference stations, or limitations on maintaining a comparatively short baseline to ground reference stations (typically of the order of several tens of kilometres). Furthermore, the PPP method is able to deliver comparable positioning accuracy, with lower computational burden and better long-term repeatability than RTK-based methods (Bisnath and Gao, 2009). However, there are some problems in using real-time GNSS PPP when there is GNSS signal degradation or blockage, such as longer convergence time, loss of precise orbit and clock correction data, etc. To reduce the convergence time there have been a number of methods proposed by augmenting GNSS PPP with additional information. For example, one of the most commonly used GNSS PPP augmentation is by adding more observations, with multiple frequency and/or GNSS constellations, including BDS, Galileo, modernised GPS and GLONASS (Tegedor et al., 2014), or tightly integrating with an inertial navigation system (INS) (Gao et al., 2017; Liu et al., 2016). Another strategy is to enable integer ambiguity resolution with precise (externally provided) atmospheric corrections to realise rapid convergence (de Oliveira et al., 2017; Geng et al., 2010; Teunissen and Khodabandeh, 2015).

In this chapter, an A-PL system consisting of A-PLs and G-PLs is proposed, where the A-PLs are positioned using the real-time GNSS PPP technique. For the proposed A-PL system all the A-PLs regularly broadcast their positions to the user in real-time without the need for monitoring by the ground stations. To enhance PPP performance for the A-PL in GNSS challenged areas, inter-PL range measurements could be combined with GNSS measurements. To process such "relative" measurements, cross-correlations among the A-PL estimated states introduced during the measurement updates have to be accounted for in order to obtain consistent parameter estimates. There are two commonly used strategies to address this problem: the centralised approach and the decentralised approach. The centralised algorithm is implemented with a master A-PL, or at a FC or central processor, gathering and processing information from all A-PLs in the network at every time instant (Howard et al., 2002). Then the FC broadcasts back the estimated positions to each A-PL. This approach suffers from high computational and communication costs. Moreover, it is susceptible to a single point of failure. To avoid communicating with a master A-PL or FC, and reducing the communication bandwidth, decentralised algorithms have been developed. The decentralised algorithms can also be divided into two categories (Kia et al., 2016). One is the tightly-coupled class of algorithms, often referred to as the centralised-equivalent approach, which has the computational load of the centralised algorithm distributed among the entire network and accurately tracks the cross-correlations. However, such algorithms still suffer from relatively high computational, communication and data storage costs, as a synchronous communication network for information exchange is required (Kia et al., 2016). The other category is the loosely-coupled class of algorithms. Although the exact cross-correlations are not maintained for this type of decentralised algorithm, the drawbacks in the tightly-coupled decentralised algorithms can be addressed. To obtain estimation consistency, CIF and SCIF algorithms can be used (Li and Nashashibi, 2013; Wanasinghe et al., 2014; Wu et al., 2017). The SCIF algorithm is able to maintain the known independent information in the estimates, which is treated as correlated with all estimates among the network by the CIF. Therefore, the decentralised algorithm based on the SCIF is more suitable to use for the A-PL distributed positioning, as only the states involved in the inter-PL ranges are known to be correlated with each other.

Free-to-air, real-time GNSS PPP depends on receiving IGS Real-Time Service (RTS) products – such as precise orbit and satellite clock corrections – transmitted continuously using, for example, the Network Transport of RTCM by Internet Protocol (NTRIP). Unfortunately, the availability of these real-time corrections is often not 100% (Hadas and Bosy, 2015). It can be worse for moving A-PLs in challenging environments where an outage of a caster connection is often likely to occur. To maintain the A-PL positioning accuracy based on GNSS PPP during periods of interruption, the real-time orbit and clock corrections can be predicted using appropriate models. The orbit prediction models are generally based on polynomial models with different orders (El-Mowafy et al., 2017; Hadas and Bosy, 2015). It has been demonstrated that this type of fitting model is able to predict short-term IGS RTS orbit corrections with better than 10cm accuracy. For long-term orbit predictions it has been proposed to use IGS Ultra-rapid (IGU) orbit corrections as a substitute for RTS orbit corrections, as these two IGS products are numerically compatible with each other (El-Mowafy, 2017). Unlike the orbit corrections, the satellite clock corrections have more complex characteristics, including both periodic and stochastic variations (Heo et al., 2010; Huang et al., 2014). The clock prediction models generally consist of two parts. One part models the linear or nonlinear coupling characteristics of the clock corrections using a polynomial model. The other part is often in sinusoidal form, and describes the periodic variation behaviour of the clock. These models can be used for long-term clock predictions over a few hours and even for a day. However, these models need long-term fitting data. In this chapter only short-term predictions of orbit and satellite clock corrections are studied with fitting data for periods less than 15 min in length, as this can be implemented online with A-PL positioning without storing the long-term fitting data.

In this chapter, the configuration of the proposed system is first described in section 3.2. Then the A-PL positioning method based on GNSS PPP combined with inter-PL ranges using the SCIF algorithm is introduced in section 3.3. In section 3.4, the short-term predictions of orbit and satellite clock corrections using different models are described. To evaluate the A-PL positioning performance, the results of a “semi-simulation” are analysed in section 3.5. The chapter summary is presented in section 3.6.

3.2 A-PL System Configuration

For the proposed A-PL system, the power consumption issues related to the GNSS receiver, a PL receiver for receiving signals from other PLs, a PL transmitter for broadcasting its own signals, a computer controller component, and other sensors to facilitate A-PL navigation and control, must be considered. A propulsion system could use state-of-the-art solar electric propulsion technology for long flights (more than 10 hour mission durations) to minimise A-PL operations and maintenance complexity. The A-PL flying height also affects its endurance, as well as introducing other operational constraints. The flying height could be above the weather and commercial air traffic (i.e. generally above 10-20km) or below (around 6km above mean sea level) (Tsujii, et al., 2001). If the flying height is around 20km, the A-PL can be assumed to stay (more or less) stationary and be mounted on a balloon or airship. For an A-PL at a lower altitude the platform typically would be a UAV. The communication range between the A-PL and ground control stations can be realised over more than 100km of range by using radios with frequency at the licence-free 2.4GHz band, transmitting at suitably high power (Amt and Raquet, 2007; Li et al., 2016). The cruise airspeed of A-PLs at different flying heights can be assumed to be from 50km/h to 100km/h, according to the capabilities of off-the-shelf UAVs. The A-PL flight paths need to be designed to ensure good overall geometry of the combined GNSS and A-PL system, with the A-PL signals assumed to be transmitting at lower elevation and higher power with respect to users having restricted satellite availability. In this study it is assumed the so-called "near-far" problem has already been solved using some form of signal pulsing scheme (Amt and Raquet, 2007).

To ensure good GDOP the proposed A-PL system could consist of both A-PLs and G-PLs. The conceptual A-PL system design is shown in Figure 3.1.

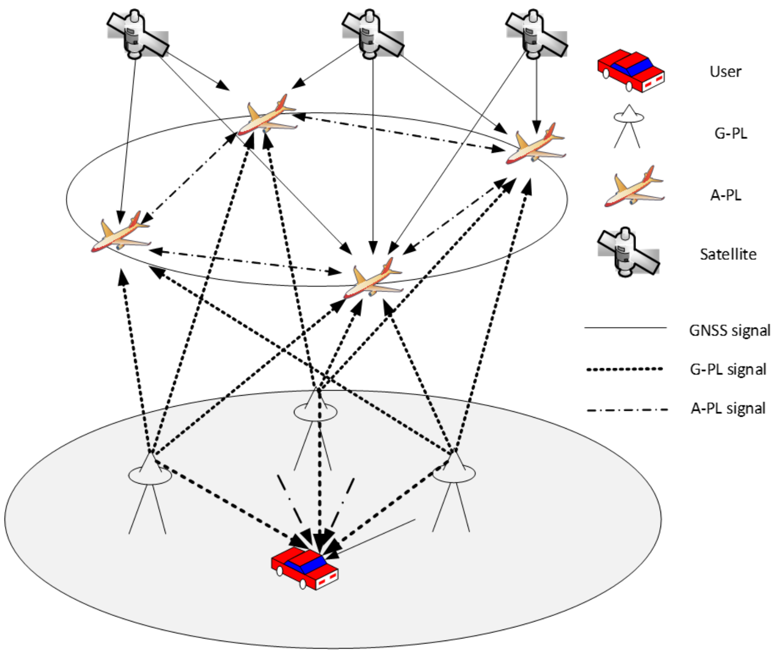


Figure 3.1 Proposed A-PL system configuration

To configure such an A-PL system, the application environment has to be taken into account. For example, considering the scenario of an urban canyon where users cannot receive satellite signals with low elevation angles, the A-PLs can therefore be designed to fly in a circular trajectory with elevation angles above 10°, depending on the specific environment scenario as well as the size of the proposed service area. The other factors to be considered include A-PL flying height(s), A-PL flight trajectory(ies), G-PL distribution, and the number of A-PLs and G-PLs in use. The A-PLs will fly at lower altitude to increase the application flexibility, but they may fly different trajectories at different heights to further improve GDOP. The G-PLs are fixed on the ground, such as on high buildings or towers, and could be distributed around the service area so as they could be viewed by some A-PLs in order to relay the time base for transmission synchronisation as well as to facilitate user positioning.

3.3 **A-PL Distributed Positioning Based on SCIF Algorithm**

All the A-PLs are assumed to be time-synchronised. The time synchronisation can be realised by chronologically synchronising the positioning signals of all the A-PLs to the time base of a designated reference transmitter with a kinematic Time Lock Loop (TTL) (Small, 2017). The reference transmitter can be synchronised to any A-PL or to a GPS satellite. The kinematic TTL is implemented by repeatedly adjusting frequency and time differences between the A-PL of interest and the reference transmitter with the self-monitored trajectory data, including location, velocity and acceleration. The time synchronisation among all the A-PLs can also be realised using the two-way time and frequency transfer method, by calculating the clock difference of ranging differential delay and error compensating between the A-PL of interest and one master A-PL as the reference without the need for A-PL accurate positions (He et al., 2016). In this chapter, the A-PL system is assumed to be able to maintain time synchronisation during the entire mission. Each A-PL can be positioned using a GNSS receiver for receiving and processing signals from satellites, and a PL receiver for receiving and processing signals from other PLs.

GNSS PPP using the dual-frequency IF measurement combination is used for the proposed A-PL absolute positioning algorithm. Since these GNSS measurements are independent of the A-PL's previous state estimates they are processed using standard KF methods.

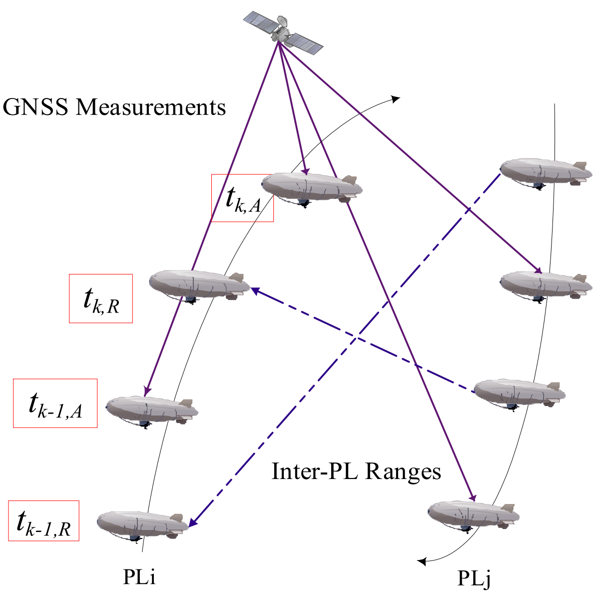


Figure 3.2 A-PL positioning scheme

Figure 3.2 is a schematic diagram for the proposed A-PL positioning scheme. PL_i makes GNSS measurements at time $t_{k-1,A}$ and $t_{k,A}$, and obtains relative measurements from PL_j at time $t_{k-1,R}$ and $t_{k,R}$. During $t_{k-1,R}$, $t_{k-1,A}$, $t_{k,R}$ and $t_{k,A}$, the state of PL_i is propagated with its corresponding dynamic model, i.e. via the time update. In addition, other sensors such as an INS, barometer or magnetometer, can be integrated to enhance the propagated state solution. If the A-PL in the network with no GNSS signal access only propagates its positions using the motion equation (2.10), the state estimate error drifts due to the noise $\boldsymbol{w}_{\cdot k}$ which grows with time without bound. To reduce the error growth, relative measurements between the A-PLs can be used. When the relative measurements are obtained the propagated state of the A-PL can be updated. The relative measurements for the A-PL positioning are the inter-PL ranges:

$$z_{k,R}^{i,j} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 + \left(z_i - z_j\right)^2} + n_k^{i,j} \tag{3.1}$$

where $[x_j \quad y_j \quad z_j]$ is the PL_j position. To perform the relative measurement update for PL_i, PL_j has to share its propagated position vector and the corresponding covariance matrix ($\hat{\boldsymbol{x}}_{k|k-1}^{j}$ and $\boldsymbol{P}_{k|k-1}^{j}$) with PL_i. PL_i has to predict the position of PL_j at the transmitting time of relative range signal according to the dynamic model in equation (2.10). Assuming that the shared trajectory data of PL_j at time $t_{k,j}$ is received by PL_i at $t_{k,R}$ along with the relative measurement $z_{k,R}^{i,j}$, the predicted position for PL_j is:

$$\boldsymbol{r}_{t_{k,R}}^{j} = \boldsymbol{r}_{t_{k,j}} + t \cdot \boldsymbol{v}_{t_{k,j}} \tag{3.2}$$

with $t = t_{k,R} - t_{k,j}$. The corresponding covariance matrix $\boldsymbol{P}_{k|k-1}^{j}$ is updated according to equation (3.2). If PL_i and PL_j are separated by a large distance it is desirable to also take into account the signal travel time, and then t can be calculated as:

$$t = t_{k,R} - t_{k,j} - \frac{z_{k,R}^{i,j}}{c} \tag{3.3}$$

where c is the speed of electromagnetic radiation in a vacuum.

With all the necessary information PL_i then updates its propagated state using the SCIF algorithm as described in chapter 2. On the other hand, the GNSS measurements are processed using standard KF methods.

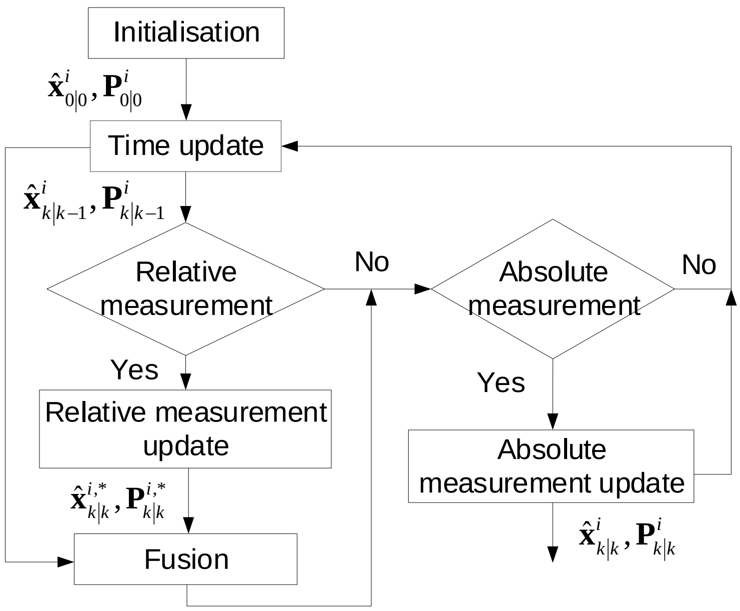


Figure 3.3 A-PL positioning based on SCIF approach

Since each A-PL determines its position in a distributed way, there is no requirement for continuous all-to-all communications, as would be necessary using the centralised approach, to keep track of the cross-correlations between different A-PLs state estimates. The A-PL can update its state using the SCIF whenever the inter-PL relative measurement is made. The detailed steps to realise the distributed localisation algorithm for A-PL positioning are illustrated in Figure 3.3. In this algorithm two independent updates are involved, one from the relative measurement update and the other from the absolute positioning measurement update. The fusion is performed after the relative measurement update.

3.4 Predictions of Orbit and Satellite Clock Corrections

Real-time GNSS PPP is dependent on precise and available orbit and satellite clock corrections. An A-PL moving in a GNSS-challenged environment may suffer from disruptions of communication links carrying the required messages. To maintain GNSS PPP positioning performance, the orbit and clock corrections can be predicted using appropriate models when there is a disrupted connection.

3.4.1 **Real-time Orbit and Clock Corrections**

IGS RTS correction streams are transmitted to users via NTRIP. They are formatted according to the RTCM Services-State Space Representation message format. The RTS orbit corrections $\delta\boldsymbol{O}$ are expressed as radial (δO_r), along-track (δO_a) and cross-track (δO_c) components. Each component has a correction term δO along with its rate-of-change $\delta\dot{O}$. The orbit correction at time t can be calculated as (El-Mowafy et al., 2017):

$$\delta\boldsymbol{O} = [\delta O_r \quad \delta O_a \quad \delta O_c]^T + [\delta\dot{O}_r \quad \delta\dot{O}_a \quad \delta\dot{O}_c]^T(t - t_0) \tag{3.4}$$

where t_0 is the reference time included in the RTS message. To apply the corrections to the broadcast orbit $\boldsymbol{X}_b$, the raw RTS corrections have to be transformed to geocentric corrections by using the radial, along-track and cross-track unit vectors (e_r, e_a, and e_c):

$$\boldsymbol{X}_p = \boldsymbol{X}_b + [e_r \quad e_a \quad e_c]\delta\boldsymbol{O} \tag{3.5}$$

where $\boldsymbol{X}_p$ is the precise orbit vector. The RTS clock correction δC is provided as a correction to the broadcast satellite clock offset. This consists of the correction quantity and its rate-of-change:

$$\delta C = C_0 + C_1(t - t_0) + C_2(t - t_0)^2 \tag{3.6}$$

where C_0, C_1 and C_2 are polynomial coefficients. Then, the corrected satellite clock offset t_{sat} is computed as:

$$t_{sat} = t_b^{sat} + \frac{\delta C}{c} \tag{3.7}$$

where t_b^{sat} is the broadcast satellite clock offset.

3.4.2 **Prediction of Orbit and Clock Corrections**

The availability of the RTS corrections has a significant influence on GNSS PPP positioning. When a correction communication link interruption occurs, the orbit and satellite clock corrections need to be predicted using appropriate models. Since the time series of orbit corrections between each Issue Of Data Ephemeris (IODE) change often exhibit a polynomial pattern, it is possible to represent (and predict) the orbit corrections by using polynomial models (Hadas and Bosy, 2015). For a short period of less than ten minutes, the polynomial models of order two to four with a few minutes of fitting data are able to achieve orbit prediction accuracy of the order of 10cm. For periods longer than one hour, it is practical to use the most recent IGU orbit corrections, which are compatible with RTS orbit corrections (El-Mowafy, 2017). However, the IGU clock correction is not good enough to be used as an alternative for the RTS clock correction during RTS outages (Nie et al., 2018). The RTS clock corrections are often predicted as a time series with both polynomial and periodic terms (Heo et al., 2010). The commonly used models for clock prediction are in the following form (Huang et al., 2014):

$$\delta t = a_0 + a_1 t + a_2 t^2 + \sum_{i=1}^{k} A_i \, sin(\omega_i t + \phi_i) \tag{3.8}$$

where

t is the time since start of modelling;

a_0, a_1 and a_2 represent the bias, drift and drift-rate of the clock corrections, respectively;

k is the number of periodic terms; and

A_i, ω_i and ϕ_i denote the amplitude, frequency and phase of the corresponding periodic term, respectively.

a_0, a_1, a_2, A_i and ϕ_i are parameters that can be estimated. The quadratic polynomial term can be neglected for current GPS satellites (Nie et al., 2018). Four main sinusoidal periods, including 15 min, 30 min, 3 h and 12 h, are found with fast Fourier transform (FFT) analysis (El-Mowafy et al., 2017). Therefore, equation (3.8) can be changed to:

$$\delta t = a_0 + a_1 t + \sum_{i=1}^{4} A_i \, sin\left(\frac{2\pi}{T_i} t + \phi_i\right) \tag{3.9}$$

where T_i is the period. In this chapter the scenario of short-term prediction of orbit and clock corrections with a fitting data period of less than 15 min is considered, and hence only one sinusoidal term is used for the clock prediction model. To account for the phase

ϕ_i within the sinusoidal term, a transform can be implemented:

$$\delta t = a_0 + a_1 t + A_s \, sin\left(\frac{2\pi}{T_1} t\right) + A_c \, cos\left(\frac{2\pi}{T_1} t\right) \tag{3.10}$$

where $A_s = A_1 \, cos(\phi_1)$ and $A_c = A_1 \, sin(\phi_1)$. To predict the orbit and clock corrections, the fitting data used for building the prediction models have to be free of outliers. To detect outliers in the orbit fitting data, one simple strategy is to check the differences between the orbit corrections and corresponding values calculated with the polynomial fitting model (El-Mowafy et al., 2017). The outlier can be iteratively detected and removed if the corresponding difference satisfies the following condition:

$$|\Delta\delta O - \mu| > f \cdot \sigma \tag{3.11}$$

where

μ and σ represent the average and standard deviation (STD) of the orbit difference $\Delta\delta O$, respectively; and

f is a scalar threshold, which is recommended to be set to the value of 3 with a 99.7% confidence level.

However, this strategy is not suitable for outlier detection of clock correction because the RTS clock correction suffers from abrupt jumps resulting from changes in reference time used by different analysis centres (Chen et al., 2017; El-Mowafy, 2017). To detect the jumps in the clock fitting data, epoch-differenced RTS clock corrections can be used. As with the exclusion condition for orbit corrections, the average and STD of the epoch-differenced RTS clock corrections are calculated. The epoch-differenced RTS clock correction $\Delta\delta t$ is flagged as an outlier if the absolute deviation around the average is larger than three times the STD. An outlier is detected if the $\Delta\delta t$ of two consecutive epochs t_{n-1} and t_n both satisfy the condition and have opposite signs. Then the clock correction at t_{n-1} is removed from the clock fitting data. However, if the clock correction at t_n does not meet the exclusion condition, a jump may exist at t_{n-1}. To identify the jump, the $\Delta\delta t$ with an extended period of time from t_{n+1} to t_{n+T} needs to be examined. T is recommended to be set as 120s (Chen et al., 2017). If all $\Delta\delta t$ does not satisfy the exclusion condition, a jump at t_{n-1} is identified. Then the fitting data for clock prediction model have to be reinitialised from t_{n-1}.

3.5 Semi-simulation Results and Analysis

To simulate an A-PL system with the A-PLs flying in a circular trajectory, a "semi-simulation" test was performed by walking on the University of New South Wales (UNSW) campus as shown in Figure 3.4 with a handheld user terminal MagicUT. This device was designed to be used on the Australia-New Zealand Space-Based Augmentation System (SBAS) Testbed. This is a second generation SBAS with Dual-Frequency Multi-Constellation (DFMC) capability. It is able to perform positioning using a PPP service in real-time. More information can be found at http://www.ga.gov.au/scientific-topics/positioning-navigation/positioning-for-the-future/satellite-based-augmentation-system. Raw L1 and L2 dual-frequency GPS measurements were collected for post-processing.

Figure 3.4 A-PL trajectory for the UNSW test

The MagicUT system has several positioning modes: SBAS L1-only, SBAS DFMC, and carrier phase-based PPP. The PPP mode was chosen for the analysis reported here. There are two modes of PPP positioning: ColdStart and QuickStart. The ColdStart needs at least 10 min for the PPP solution to converge to within 40cm, while the QuickStart uses a precisely surveyed point to ensure almost instantaneous convergence. The QuickStart mode was used for the PPP initialisation at a pre-surveyed point. To provide the "ground truth" for the A-PL trajectory, the Piksi Multi, a multi-band multi-constellation RTK

GNSS receiver capable of centimetre-level accuracy, was set up alongside the MagicUT so as to use the same antenna, as shown in Figure 3.5. Approximately 30 min of 1Hz GPS measurements were collected. The data were post-processed in a float-ambiguity PPP solution with real-time IGS combined corrections (IGC), IGC01, which is one of the RTS solutions generated by decoding the real-time product streams with latency of 25s.

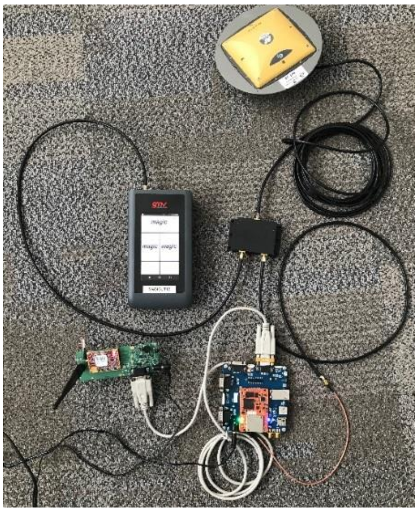


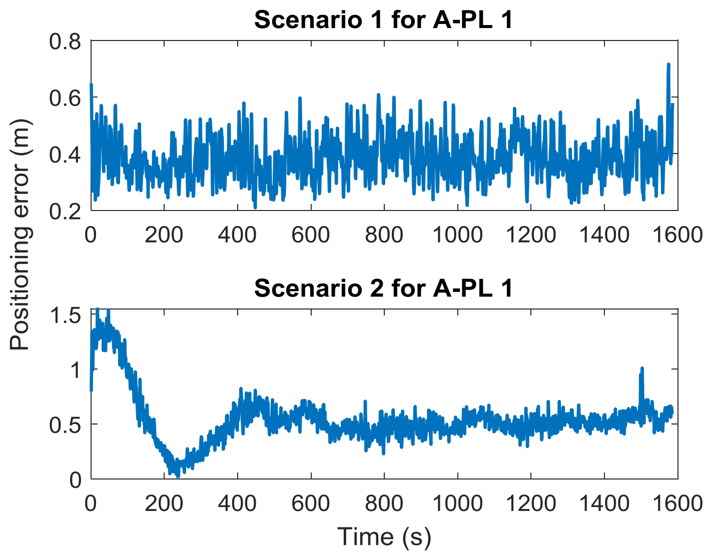
Figure 3.5 MagicUT and Piksi Multi hardware setup

3.5.1 **Simulation Assumptions**

It is assumed that there were four A-PLs moving along predefined paths. The initial positions of the A-PLs were evenly distributed on a circular trajectory. Each A-PL can measure inter-PL ranges from other A-PLs, as well as make GNSS measurements. For this simulation the data rate of inter-PL ranges was assumed to be 10Hz. The inter-PL ranges were generated with the "ground truth" provided by the Piksi Multi. The accuracy of the inter-PL range was assumed to be 5cm and simulated by adding the corresponding magnitude of white noise. T_c used for the A-PL dynamic model was set to 50. The covariance values of the carrier phase and pseudorange noise used in KF were set to 0.03^2 and 3^2, respectively; and the covariance values of acceleration and tropospheric wet delay noise were set to empirical values 10^{-8} and 10^{-14}, respectively.

3.5.2 **A-PL Positioning Performance**

To evaluate the contribution of inter-PL ranges to A-PL GNSS PPP positioning, two scenarios for A-PL positioning were simulated. The first scenario was that the A-PL of interest was able to retain the converged GNSS PPP positioning accuracy after the initial setup through the entire mission, referred to as Scenario 1. Since the experiment was performed on the university campus with some trees impacting the simulated A-PL trajectory, some GNSS signals were intermittently disrupted. To obtain converged GNSS PPP positioning accuracy for this scenario, the height component solutions provided by the Piksi Multi (with 10cm accuracy) were used to constrain GNSS PPP positioning accuracy. The second scenario was that A-PL GNSS PPP positioning has to converge to the desired accuracy during movement, referred to as Scenario 2. To simulate this scenario, the PPP initialisation with the pre-surveyed point was not used. The A-PL of interest was randomly chosen from the four simulated A-PLs, is referred to as A-PL 1, in the following simulation. The other three A-PLs, i.e. observed A-PLs, were designated A-PL 2, 3 and 4. The influence of the observed A-PL trajectory data on A-PL GNSS PPP positioning with inter-PL ranges was also investigated. As was the case for A-PL 1, the other three A-PLs were simulated with two same GNSS PPP positioning scenarios. For the simulation, the transmitted trajectory data of the observed A-PLs were assumed to be estimated positions at the previous instant of the received relative measurements. The positions of observed A-PLs were predicted with the state dynamic model before the relative measurement update. The three-dimensional (3D) positioning error achieved by A-PL 1 and the other three A-PLs' GNSS PPP for the two scenarios are shown in Figure 3.6. The converged accuracies calculated from 600s in terms of root mean squared error (RMSE) are listed in Table 3.1.



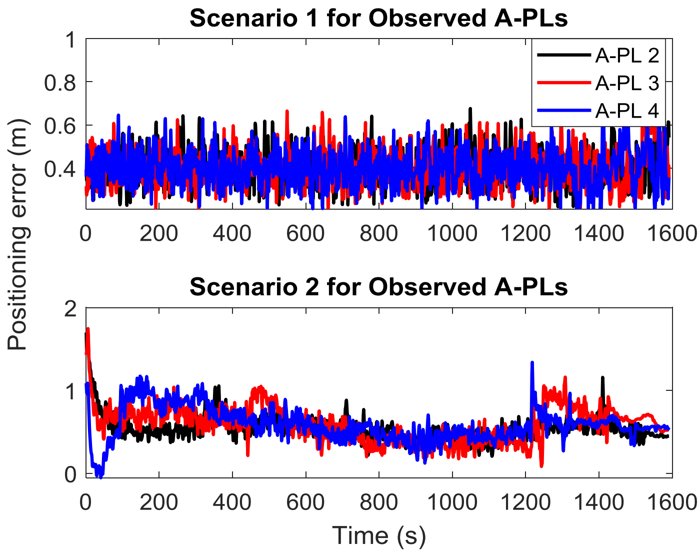
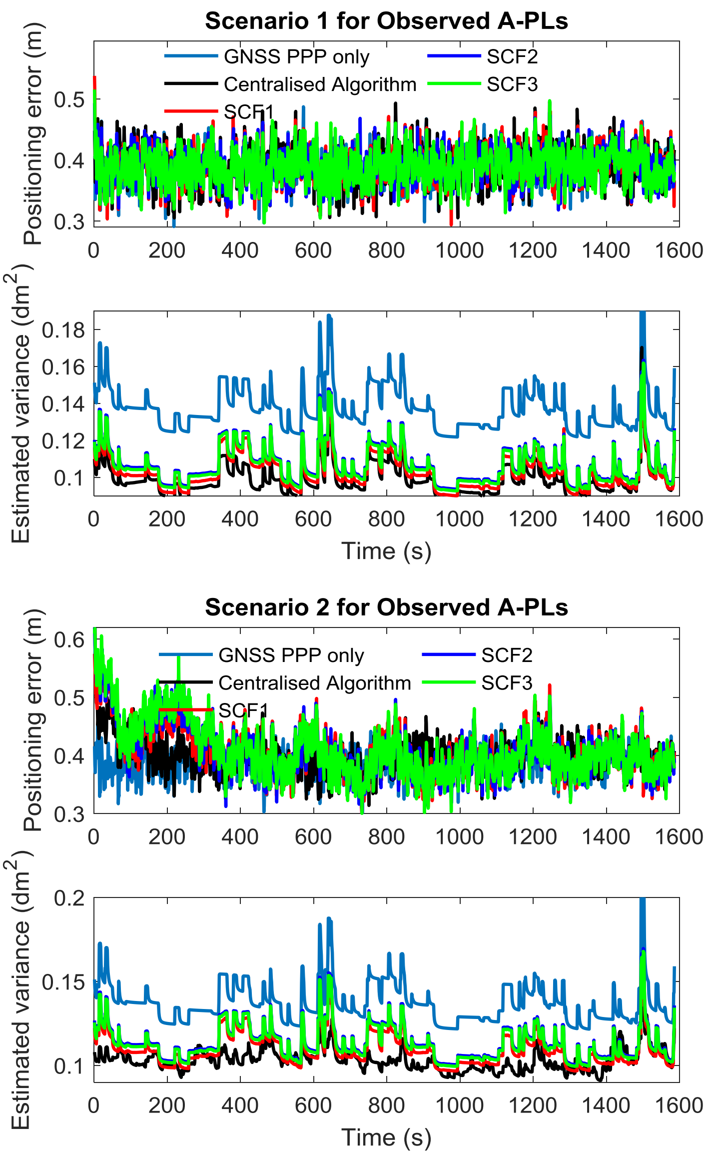


Figure 3.6 A-PL GNSS PPP positioning results

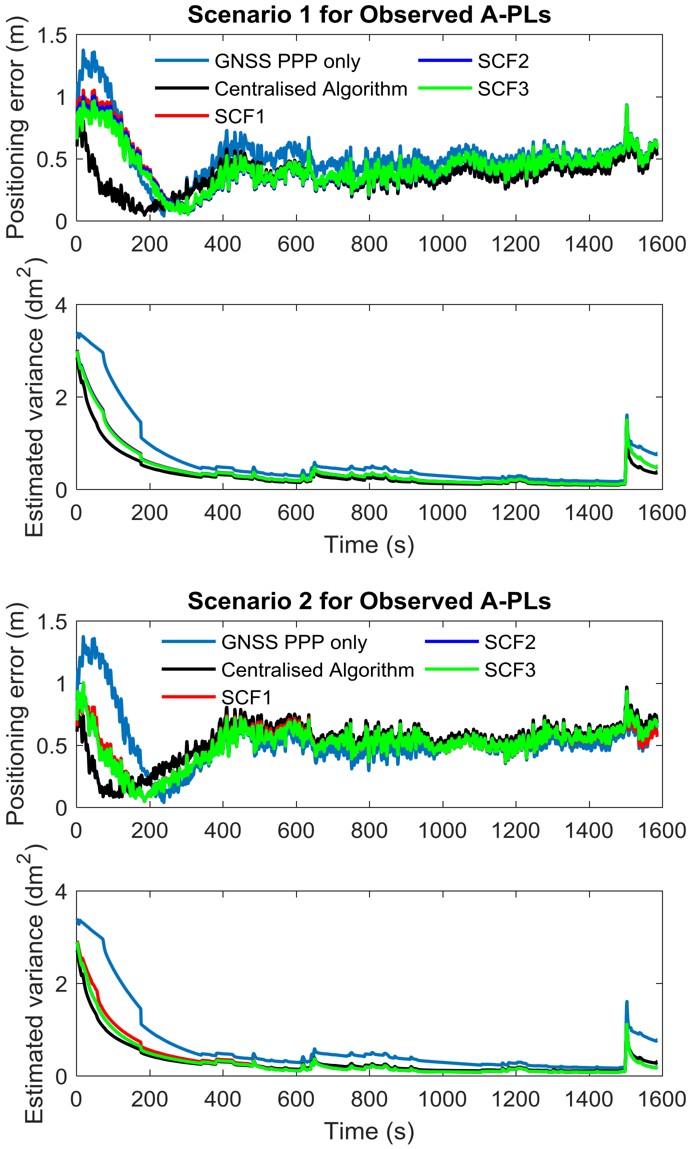
Table 3.1 A-PL GNSS PPP positioning accuracy

Scenarios	Positioning Errors (m)	
	A-PL 1	Observed A-PLs (A-PL 2, 3 and 4)
1	0.39	0.40
2	0.51	0.55

To process the inter-PL range measurements, the SCIF-based distributed positioning algorithms were evaluated and compared with the centralised algorithm. All three forms of the SCIF algorithm with the fusion implemented during and after relative measurement update based on equations (2.19), (2.20) and (2.21) are referred to in the following simulations as SCIF1, SCIF2 and SCIF3, respectively. An EKF was implemented locally on each A-PL to estimate the A-PL positions for the SCIF-based distributed algorithms. The optimum values of ω used in the distributed algorithms were determined by minimising the trace of the fused covariance. In addition, the centralised algorithm was operated with the same simulation conditions as the distributed algorithms. This estimates a joint state composed of states of all the A-PLs and tracks the cross-correlations among all the states. The estimated positioning error and corresponding variances, represented by the diagonal components of the $\boldsymbol{P}$ matrix calculated during the EKF, for the two positioning scenarios for A-PL 1 with two different observed A-PLs scenarios are shown in Figure 3.7. The estimated positioning variances in theory reflect the real positioning error. However, they could be affected by the inaccurate predefined covariances of process and measurement noises. Table 3.2 lists the corresponding converged accuracies and STDs of the estimated positioning errors for all simulated scenarios.



(Scenario 1 for A-PL 1)



(Scenario 2 for A-PL 1)

Figure 3.7 A-PL positioning accuracy of UNSW trial using different algorithms

Table 3.2 A-PL positioning accuracy using different algorithms

Algorithms	Scenario 1 for A-PL 1				Scenario 2 for A-PL 1			
	Observed A-PLs Scenario 1		Observed A-PLs Scenario 2		Observed A-PLs Scenario 1		Observed A-PLs Scenario 2	
	RMSE (m)	STD (dm)	RMSE (m)	STD (dm)	RMSE (m)	STD (dm)	RMSE (m)	STD (dm)
GNSS PPP-only	0.39	0.27	0.39	0.25	0.51	0.71	0.51	0.71

Centralised algorithm	0.39	0.28	0.40	0.26	0.40	0.82	0.59	0.72
SCIF1	0.39	0.27	0.40	0.31	0.45	0.92	0.55	0.70
SCIF2	0.39	0.27	0.40	0.30	0.45	0.92	0.55	0.71
SCIF3	0.39	0.27	0.40	0.31	0.45	0.91	0.55	0.71

From the positioning performance comparison of Scenario 1 for A-PL 1, it can be seen that the A-PL with both GNSS and inter-PL range measurements has almost the same performance in terms of positioning accuracy and smoothness as that for GNSS PPP when the A-PLs with 0.4m GNSS PPP accuracies are observed. However, when the observed A-PLs have to initialise during the movement, utilising the inter-PL ranges could degrade the A-PL GNSS PPP positioning performance, as can be seen in the bottom figure of Scenario 1 for A-PL 1. Both the centralised and distributed algorithms have to re-converge at the beginning and give slightly worse converged positioning accuracy than in the GNSS PPP-only case. The contribution of inter-PL ranges to A-PL GNSS PPP positioning is further demonstrated by the comparison in Scenario 2 for A-PL 1. It can be seen that by observing A-PLs with 0.4m GNSS PPP positioning accuracies the algorithms combining GNSS PPP with inter-PL ranges achieved better converged accuracies than for the GNSS PPP-only case. There is also a tendency for a reduction in GNSS PPP convergence time, as indicated by the positioning results and estimated variances. When the trajectory data of the observed A-PLs degrades, the algorithms with both GNSS and inter-PL range measurements reduce the convergence time of GNSS PPP, except that they converge to worse positioning accuracies. Therefore, to ensure the enhancement of inter-PL ranges in practical applications, it is necessary to first check the integrity of the transmitted trajectory data. Interested readers are referred to Goel et al. (2017) for methods of monitoring transmitted information integrity. Furthermore, when more observed A-PLs with good GDOP could be included, there will be a reduction in convergence time for GNSS PPP when adding inter-PL range measurements.

By comparing the positioning and estimated variance results of the centralised and three SCIF-based distributed algorithms as shown in Figure 3.7, it can be seen that the centralised algorithm generates a smoother positioning result and has faster convergence due to the precisely tracked cross-correlations among the A-PL states. The top figure of Scenario 2 for A-PL 1 illustrates this performance difference. As listed in Table 3.2, compared with the GNSS PPP-only case, around 20% and 10% improvement in converged positioning accuracy is achieved by the centralised and SCIF-based distributed algorithms, respectively. However, when the observed A-PLs could not provide the trajectory information with satisfactory accuracy, as shown in the bottom figure of scenario 2 for A-PL 1, the converged accuracy of the centralised algorithm is even worse than that of the distributed algorithms. In this case, the distributed algorithms tend to be more robust in dealing with the deteriorated trajectory data of the observed A-PLs. The difference in positioning performance between the centralised and SCIF-based algorithms with Scenario 1 for A-PL 1 is much smaller than that with Scenario 2 for the A-PL 1. The SCIF-based algorithms can achieve almost the same positioning performance as the centralised algorithm when the GNSS PPP-only is generated by a converged solution. In the case of the three different SCIF-based distributed algorithms, all achieve almost the same positioning performance as shown by all the simulated scenarios. In principle, any one of the three SCIF-based algorithms can be used for A-PL positioning, if the slightly higher computational cost of SCIF1 is not an issue. The positioning performance of GNSS PPP combined with the inter-PL ranges achieved in the simulations could be further improved when the inter-PL ranges and GNSS measurements are processed simultaneously in a tightly combined mode as there is no need to perform one more propagation step, which indirectly degrades accuracy of the observed A-PLs' transmitted trajectory data.

3.5.3 Analysis of Predictions of Orbit and Satellite Clock Corrections

To evaluate the fitting models for short-term RTS correction predictions, the accuracy of IGC predictions with a sliding time window was investigated. The prediction errors were derived from the difference between the predicted values and their known IGC correction values during the prediction period. The periods of the fitting and prediction data were set as 10 min and 30 min, respectively. Three polynomial models with different orders – including first, second and third order – were investigated. The clock corrections were predicted using the model with linear and sinusoidal terms or only a linear term. For the clock prediction model, the period of the sinusoidal term was first estimated using the FFT. Other parameters involved in the prediction models for the orbit and clock corrections were estimated with a built-in "fittype" function in Matlab.

Table 3.3 Mean prediction error of RTS corrections for G17

Models		Prediction error (cm)	
		Mean	STD
3D orbit (a_0, a_1, a_2, a_3)	Poly with order 1 $(1.02, 4.01 \times 10^{-4})$	10.3	5.9
	Poly with order 2 $(0.99, 5.03 \times 10^{-4}, -5.75 \times 10^{-8})$	5.8	4.1
	Poly with order 3 $(1.01, 3.36 \times 10^{-4}, 1.79 \times 10^{-7}, -8.92 \times 10^{-11})$	22.8	15.9
Clock (a_0, a_1, A_s, A_c)	Linear and sinusoidal terms $(7.28, 2.95 \times 10^{-4}, 0.03, 0.14)$	12.2	7.7
	Only a linear term $(7.42, -1.38 \times 10^{-4})$	13.2	8.8

Table 3.3 shows one example of the prediction performance for GPS satellite PRN17. The estimated coefficients of all the models are also listed in Table 3.3. The mean and STD prediction errors represent the mean and STD value of all the prediction errors calculated with a sliding window. Since the IGC corrections are not always available, the amount of fitting data in the sliding window may vary, which has to be at least larger than five. The orbit and clock prediction errors only include those calculated before each IODE change and clock jump, respectively. Figure 3.8 shows the variation of all prediction errors. It can be seen that the worst model for orbit prediction is the one with order three. The second-order polynomial model achieves the best orbit prediction performance with the smallest mean and STD prediction errors. For satellite clock corrections, the model with linear and sinusoidal terms could obtain slightly better performance than that with only a linear term. The mean prediction error of clock corrections for different types of GPS satellites are also given in Table 3.4. It can be seen that better performance can be achieved by the prediction model with linear and sinusoidal terms compared with that using only a linear term for all different types of GPS satellites. The performance difference of these two prediction models is less in block IIF than that of older blocks IIR-M and IIR.

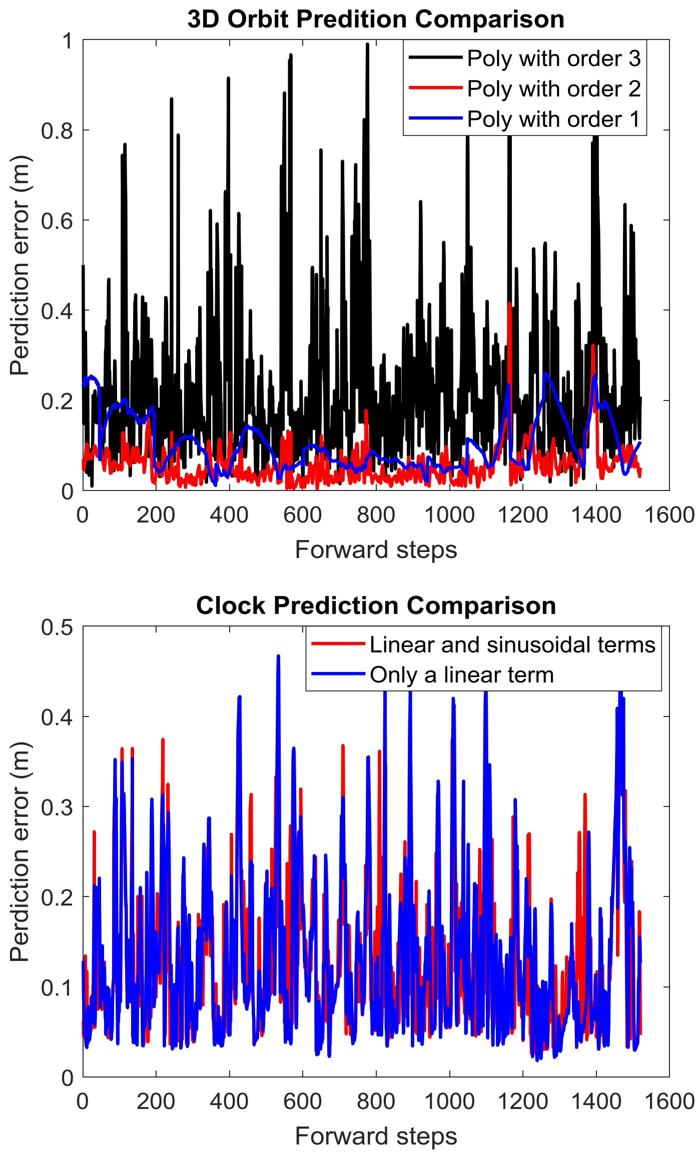


Figure 3.8 Correction prediction comparison

Table 3.4 Mean prediction error of clock corrections for all GPS satellites

Satellites		Models	
		Linear and sinusoidal terms	Only a linear term
Block IIR	Mean (cm)	11.4	12.8
	STD (cm)	7.9	8.4
Block IIR-M	Mean (cm)	11.6	14.5
	STD (cm)	8.5	9.9

Block IIF	Mean (cm)	6.0	6.1
	STD (cm)	4.7	4.6

To investigate the effect of correction prediction on A-PL positioning, it was assumed that there was one break in receiving the correction messages added to the A-PL measurements, which lasted around 10 min. The second-order polynomial and linear and sinusoidal models were used for orbit and satellite clock predictions, respectively. The A-PL positioning performance with the correction prediction was compared with the results of PPP without predictions and without breaks, as shown in Figure 3.9. Table 3.5 summarises these results. The PPP results obtained with correction predictions are almost the same as those without breaks, which demonstrates the effectiveness of the correction prediction models.

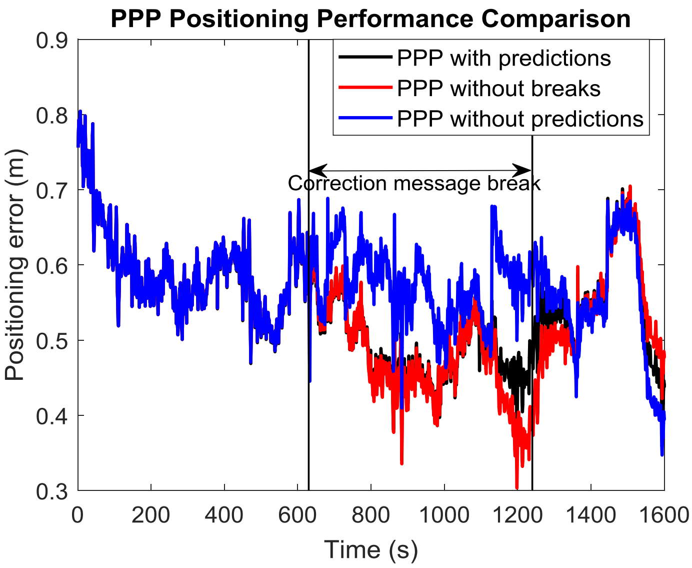


Figure 3.9 A-PL positioning performance

Table 3.5 A-PL positioning accuracy

Modes	Positioning Errors (m)
PPP with predictions	0.54
PPP without breaks	0.53
PPP without predictions	0.58

3.6 **Summary**

In this chapter an A-PL positioning concept based on real-time GNSS PPP has been proposed. The inter-PL ranges are used to enhance A-PL positioning. These “relative” measurements are processed using SCIF algorithms to account for cross-correlations of all A-PL estimated states. SCIF algorithms implemented in three forms were described and investigated. In addition, the short-term prediction of precise orbit and satellite clock corrections with different prediction models was analysed and compared when the correction message communication links were assumed to have been disrupted. Simulations have been performed to investigate the A-PL positioning performance. The simulation results demonstrate that when the A-PL has to initialise during movement GNSS PPP combining with inter-PL range measurements is able to achieve faster convergence and 20% improvement in converged positioning accuracy compared with GNSS PPP-only approach. However, the degree of improvement due to the use of inter-PL ranges is limited by the initialisation or re-initialisation of the A-PL and its observed A-PLs. To ensure the enhancement of inter-PL ranges, the transmitted trajectory data of observed A-PLs have to be provided with well-converged accuracy. Although the SCIF-based distributed algorithms indicate limited improvement compared with the centralised algorithm, they are more robust in dealing with degraded transmitted trajectory data of the observed A-PLs. In addition, the second-order polynomial model is preferable for short-term orbit correction predictions compared with the first- or third-order models. The satellite clock corrections can be predicted using either the linear model or one with linear and sinusoidal terms. The prediction models have been shown to be able to effectively reduce the impact of disruption of communication links, and hence to maintain PPP accuracy.

Chapter 4 Model-learning Based KF Algorithms for GNSS PPP with Unmodelled Measurement Error

4.1 Introduction

A precise measurement model is essential for unbiased estimation of GNSS positioning with KF based filters. However, there are some errors in the GNSS observations, such as multipath and signal interference, that are difficult to model, or do not cancel out by differencing, or cannot be accounted for using bias service products. To account for these errors one commonly used strategy is to augment the filter state with the unmodelled errors and estimate them along with user position (and other navigational parameters) (Chang, 2014; Hu et al., 2010; Zhou et al., 2017). However, a priori knowledge about the unmodelled errors in terms of their dynamic model or statistical characteristics is assumed known. Furthermore, if there is no error present in the observations, such a model augmentation approach could result in an unstable system. Another way to deal with such errors is by directly modelling them using parametric or non-parametric models (Lee and Johnson, 2017; Lv et al., 2015; Ko and Fox, 2009; Ko et al., 2007; Zhou et al., 2016). Parametric models include autoregressive (AR), moving average (MA), autoregressive-moving average (ARMA), etc. Non-parametric models include artificial neural networks (ANNs), support vector machine (SVM), GPR, etc. The disadvantage of parametric models is that substantial domain expertise is required to build these models, that are nevertheless often simplified representations (Lee and Johnson, 2017). Non-parametric models are able to flexibly approximate unknown nonlinearities with less restrictiveness than the parametric models. Two commonly used non-parametric model-learning methods – including SVM and GPR-based modelling approaches – were studied in this research. LS-SVM is one popular variant of the standard SVM, which is able to estimate highly nonlinear functions and solve noisy “black-box” modelling problems (Suykens and Vandewalle, 1999). This method has been used to estimate the unmodelled bias of dynamic models in the KF-based methods. By training the relationship of the unmodelled biases among current and previous epochs, it is able to predict, and compensate for, the bias in the KF. The LS-SVM algorithm has demonstrated its effectiveness to adapt to a time-variant dynamic model. However this algorithm cannot provide the prediction precision directly, as it has to calculate the variance of the prediction separately to account for its uncertainty in the KF. In addition, to determine the optimal parameters to be used in the LS-SVM, there have been many approaches proposed, such as a priori knowledge, statistical theory of SVM, the grid search with cross-validation, genetic algorithm, scale space theory, analytical method, and others (Cherkassky and Ma, 2004). The cross-validation based techniques are often used to select the parameters. However, this approach often has high computational cost with requirements for re-sampling data sets. To be able to set the parameters directly from the training data without resorting to re-sampling, the analytical approach based on the well-known theory of SVM regression was used in this research. Unlike the LS-SVM algorithm, the GPR-based algorithm can not only directly provide the mean function value prediction of interest, but also uncertainty estimates for the prediction. By taking both noise and regression uncertainty into account, the GPR can automatically increase its uncertainty when there is not enough training data. Furthermore, models such as AR, MA, and ARMA can be seen as special cases of GPR (Zhou et al., 2017). However, both the LS-SVM and GPR modelling approaches usually perform model-learning offline using large volumes of training data. To be able to adapt the LS-SVM/GPR modelling to changes in the environment and in the system configuration, a LS-SVM/GPR-based KF algorithm is proposed. By using training data derived from previous historical measurement residuals calculated with predicted measurements with the known states and real observations in the KF, the LS-SVM/GPR can be trained in real-time. With the trained model the unmodelled measurement error can then be predicted and compensated for in the KF. In this research, a nonlinear autoregressive model was used to train the LS-SVM/GPR model. One major limitation to training the LS-SVM/GPR model online is its high computational cost resulting from inverse matrix calculation for query point predictions. In addition, the computational burden will be increased by $O(m^3)$ when applied to the GNSS positioning with m-dimensional satellite observations. To reduce the computational cost, each output dimension is trained using a different LS-SVM/GPR model, assuming the satellite observations are independent of each other. To improve reliability of the training points and further reduce the computational burden, a sliding window with more than one forward step for the input training points and model training is proposed. To avoid over-prediction of query data and to improve the robustness of the online LS-SVM/GPR model training due to its limited effectiveness, a constraint on the query data is also introduced.

This chapter is structured as follows. First, the problem of KF with unmodelled measurement error is discussed in section 4.2. In section 4.3, the standard GPR and the proposed online GPR-KF algorithm are described. To evaluate the proposed online GPR-KF algorithm for GNSS navigation, both static and kinematic experiment results and analyses are presented in section 4.4. Finally, the chapter summary is given in section 4.5.

4.2 **KF with Unmodelled Measurement Error**

Consider a linear measurement function model with unmodelled measurement error $\Delta \boldsymbol{z}_k$:

$$\boldsymbol{z}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \Delta \boldsymbol{z}_k + \boldsymbol{\varepsilon}_k \tag{4.1}$$

The conventional KF without considering term $\Delta \boldsymbol{z}_k$ is described in chapter 2 with two recursively implemented propagation and measurement update steps. There are several ways to take into account $\Delta \boldsymbol{z}_k$ in KF. For example, one commonly used approach is to augment the state with $\Delta \boldsymbol{z}_k$ and allow its estimation along with $\boldsymbol{x}_k$. However this approach assumes the process model of $\Delta \boldsymbol{z}_k$ or a priori knowledge of its statistical characteristics is known. In this chapter, the approach via directly modelling of $\Delta \boldsymbol{z}_k$ is studied. Therefore the update step in KF can be modified accordingly as follows (Hamilton, 1994):

$$\begin{gathered} \boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T \left(\boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T + \boldsymbol{P}_{\Delta \hat{\boldsymbol{z}}_{k|k-1}} \right)^{-1} \\ \hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k \left(\boldsymbol{z}_k - \boldsymbol{H}_k \hat{\boldsymbol{x}}_{k|k-1} - \Delta \hat{\boldsymbol{z}}_{k|k-1} \right) \end{gathered} \tag{4.2}$$

where

$\hat{\boldsymbol{x}}_{k|k-1}$ and $\boldsymbol{P}_{k|k-1}$ denote the predicted state vector and its associated covariance matrix, respectively;

$\boldsymbol{K}_k$ is the Kalman gain matrix;

$\hat{\boldsymbol{x}}_{k|k}$ denotes the KF estimated state vector; and

$\Delta \hat{\boldsymbol{z}}_{k|k-1}$ and $\boldsymbol{P}_{\Delta \hat{\boldsymbol{z}}_{k|k-1}}$ denote the estimated unmodelled error vector and its associated covariance matrix, respectively.

Since the expectation of the measurement residual is:

$$E(\boldsymbol{z}_k - \boldsymbol{H}_k\widehat{\boldsymbol{x}}_{k|k}) = \boldsymbol{H}_k E(\boldsymbol{x}_k - \widehat{\boldsymbol{x}}_{k|k}) + E(\boldsymbol{\varepsilon}_k) + E(\Delta\boldsymbol{z}_k) \approx E(\Delta\boldsymbol{z}_k) \tag{4.3}$$

To estimate $\Delta\widehat{\boldsymbol{z}}_{k|k-1}$, modelling methods such as AR, MA, ARMA, SVM and GPR-based techniques can be used. These methods can be divided into two categories: parametric models and non-parametric models. Non-parametric models have less restrictiveness than the parametric models, hence they have a more flexible capability for approximating unknown nonlinearities. In this research two commonly used non-parametric model-learning methods for $\Delta\widehat{\boldsymbol{z}}_{k|k-1}$ estimation were studied: LS-SVM and GPR. LS-SVM is an invariant of the standard SVM with a quadratic cost function, the optimisation of which is simple, requiring the solution of a set of linear equations. It has been shown good generalisation performance for highly nonlinear function regression. One drawback of this algorithm is that the prediction precision with the trained LS-SVM model has to be calculated separately. This could impede its real-time application due to the increased computational cost. On the other hand the GPR-based model-learning method provides not only the mean function value prediction of interest but also uncertainty estimates for the prediction. In addition, some parametric models, such as AR, MA, and ARMA, can be seen as special cases of GPR. In order to take full advantage of such non-parametric model-learning methods the LS-SVM/GPR-based KF algorithm with model training online is demanding as it must adapt to changes in the environment and in the system configuration.

4.3 **Non-parametric Model-learning Based KF Algorithms**

4.3.1 **LS-SVM Based KF Algorithm**

Assuming a set of n training data $\{\boldsymbol{x}_i, y_i\}_{i=1}^{n}$ drawn from the noisy process, the model based on SVM can be written as (Suykens and Vandewalle, 1999):

$$y = \boldsymbol{\omega}^{\mathrm{T}}\varphi(\boldsymbol{x}) + b \tag{4.4}$$

where

$\boldsymbol{x}$ and y denote the input and output of the SVM based model, respectively;

$\boldsymbol{\omega}$ is the weight vector;

$\varphi(\boldsymbol{x})$ is the nonlinear function that maps the input to a high-dimensional feature space; and

b is the bias term.

The LS-SVM for the regression is an optimisation of a quadratic problem with equality constraints, which can be described by the following cost function:

$$\begin{gathered}\min_{\boldsymbol{\omega},b,\boldsymbol{e}} \boldsymbol{J}(\boldsymbol{\omega},\boldsymbol{e}) = \frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega} + \frac{1}{2}\gamma\sum_{i=1}^{n} e_i^2 \\ s.t.\ y_i = \boldsymbol{\omega}^{\mathrm{T}}\varphi(\boldsymbol{x}_i) + b + e_i\end{gathered} \tag{4.5}$$

where

e is the model error, also referred as the slack variable; and

γ denotes the regularisation parameter that adjusts the relative importance of the terms in equation (4.5), and is a positive real constant.

Large γ values increase the importance of the empirical risk, hence it improves the learned model but also increases its complexity. On the other hand, small γ values could avoid overfitting and result in decreased model complexity with deteriorated model accuracy. To solve the optimisation of equation (4.5), the Lagrangian function is generally used:

$$L(\boldsymbol{\omega},b,e,\alpha) = \frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega} + \frac{1}{2}\gamma\sum_{i=1}^{n} e_i^2 - \sum_{i=1}^{n}\alpha_i[\ \boldsymbol{\omega}^{\mathrm{T}}\varphi(\boldsymbol{x}_i) + b + e_i - y_i] \tag{4.6}$$

where α_i is the Lagrange multiplier. With the optimality conditions for equation (4.6), one can obtain $\boldsymbol{\alpha} = [\alpha_1\ \ \alpha_2\ \ \cdots\ \ \alpha_n]^{\mathrm{T}}$ and b solution after eliminating $\boldsymbol{\omega}$ and e:

$$\begin{bmatrix} \mathbf{0} & \mathbf{1}_n^T \\ \mathbf{1}_n & \boldsymbol{\psi}_n + \dfrac{\boldsymbol{I}_n}{\gamma} \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{y} \end{bmatrix} \tag{4.7}$$

where

$\mathbf{1}_n$ is a n-dimensional vector with elements 1;

$\boldsymbol{\psi}_n$ is a $n \times n$ matrix with the element $\boldsymbol{\psi}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ at the ith row and the jth column; and

$\boldsymbol{y} = [y_1 \ y_2 \ \cdots \ y_n]^{\mathrm{T}}$.

According to Mercer's condition, one can choose a kernel function such that:

$$\boldsymbol{\psi}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \varphi(\boldsymbol{x}_i)^{\mathrm{T}}\varphi(\boldsymbol{x}_j), i, j = 1, \cdots, n \tag{4.8}$$

Then the solution of $\boldsymbol{\alpha}$ and b can be written as:

$$\begin{gathered} b = \frac{\mathbf{1}_n^T \left[\boldsymbol{\psi}_n + \frac{\boldsymbol{I}_n}{\gamma}\right]^{-1} \boldsymbol{y}}{\mathbf{1}_n^T \left[\boldsymbol{\psi}_n + \frac{\boldsymbol{I}_n}{\gamma}\right]^{-1} \mathbf{1}_n} \\ \boldsymbol{\alpha} = \left[\boldsymbol{\psi}_n + \frac{\boldsymbol{I}_n}{\gamma}\right]^{-1} \boldsymbol{y} - \left[\boldsymbol{\psi}_n + \frac{\boldsymbol{I}_n}{\gamma}\right]^{-1} \mathbf{1}_n b \end{gathered} \tag{4.9}$$

As a result, the LS-SVM model for function estimation can then be expressed as:

$$y = \sum_{i=1}^{n} \alpha_i \ \boldsymbol{\psi}(\boldsymbol{x}, \boldsymbol{x}_i) + b \tag{4.10}$$

In this chapter, the radial basis function (RBF) kernel function is utilised, which can be expressed as:

$$\boldsymbol{\psi}(\boldsymbol{x}_i, \boldsymbol{x}_j) = exp\left(-\frac{\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|_2^2}{2\sigma^2}\right) \tag{4.11}$$

where σ denotes the kernel width, which has to be carefully chosen to avoid overfitting or underfitting. Iterative methods are often implemented to calculate $\boldsymbol{\alpha}$ and b. Its convergence speed depends on the values of γ and σ. To determine the values of these two parameters without recourse to re-sampling data sets, an analytical selection can be used. Since the kernel function is based on RBF, the width parameter σ should reflect the distribution/range of $\boldsymbol{x}$ values of the training data. It can be set to:

$$\begin{cases} \sigma \sim (0.1-0.5)\cdot \mathrm{range}(\boldsymbol{x}) & \boldsymbol{x}\in \boldsymbol{R} \\ \sigma^d \sim (0.1-0.5) & \boldsymbol{x}\in \boldsymbol{R}^d \end{cases} \tag{4.12}$$

where d denotes the dimension of the input training data. The input training data need to be pre-scaled to $[0,1]$ range. γ can be determined by directly relating it to the output values of the training data, which can be calculated as:

$$\gamma = max\left(\left|\bar{y}+3\sigma_y\right| \quad \left|\bar{y}-3\sigma_y\right|\right) \tag{4.13}$$

where $\bar{y}$ and σ_y are the mean and STD of the output values of the training data, respectively. As can be seen from equation (4.9), there is $O(n^3)$ computational complexity by calculating the inverse of matrix $\left[\boldsymbol{\Psi}_n + \frac{\boldsymbol{I}_n}{\gamma}\right]$. For *m*-dimensional outputs, the computational cost would increase to $O(m^3n^3)$. The computational complexity for the LS-SVM model training will be further increased with the number of iterations in its implementation. In order to reduce the computational cost for the GNSS PPP application, it is assumed that the output dimensions are independent of each other. In this way the computational cost can be reduced to $O(mn^3)$. To predict the unmodelled measurement error $\Delta\boldsymbol{z}_k$ based on LS-SVM in the KF, the model of $\Delta\boldsymbol{z}_k$ used is assumed to be a nonlinear autoregressive model:

$$\Delta\boldsymbol{z}_k = f\left(\Delta\boldsymbol{z}_{k-1}, \cdots, \Delta\boldsymbol{z}_{k-p}\right) + \eta_k \tag{4.14}$$

where

p is the number of lags used to represent the unmodelled measurement error; and

η_k is white Gaussian noise.

To train this model the training points of inputs $\boldsymbol{X}$ and outputs $\boldsymbol{y}$ can be obtained using the estimated $\Delta\hat{\boldsymbol{z}}_k$ of previous epochs. To improve the reliability of the training points, the input training points can use a sliding window with more than one forward step:

$$\boldsymbol{X} = \begin{bmatrix} \Delta\hat{\boldsymbol{z}}_{k-p-(n-1)\cdot s|k-p-(n-1)\cdot s} & \cdots & \Delta\hat{\boldsymbol{z}}_{k-(n-1)\cdot s-1|k-(n-1)\cdot s-1} \\ \vdots & \ddots & \vdots \\ \Delta\hat{\boldsymbol{z}}_{k-p|k-p} & \cdots & \Delta\hat{\boldsymbol{z}}_{k-1|k-1} \end{bmatrix}$$

$$\boldsymbol{y} = \begin{bmatrix} \Delta\hat{\boldsymbol{z}}_{k-(n-1)\cdot s|k-(n-1)s} \\ \Delta\hat{\boldsymbol{z}}_{k-n\cdot s|k-p-n\cdot s} \\ \cdots \\ \Delta\hat{\boldsymbol{z}}_{k|k} \end{bmatrix} \tag{4.15}$$

where s is the forward step size. With the trained model of $\Delta\boldsymbol{z}_k$, the prediction of $\Delta\hat{\boldsymbol{z}}_{k+1|k}$ can be obtained by treating $\boldsymbol{x}_{*,k} = [\Delta\hat{\boldsymbol{z}}_{k-p+1|k-p+1} \quad \Delta\hat{\boldsymbol{z}}_{k-p+2|k-p+2} \quad \cdots \quad \Delta\hat{\boldsymbol{z}}_{k|k}]^T$ as the query point. Here the training points $\boldsymbol{X}$ and the query point $\boldsymbol{x}_{*,k}$ only represent a one-dimensional output of $\Delta\boldsymbol{z}_k$. This strategy will increase the demand for memory. To further reduce the computational cost of $\Delta\hat{\boldsymbol{z}}_{k+1|k}$ prediction, the trained LS-SVM model also uses a sliding window with q steps forward after the first modelling. During the next q epochs, $\Delta\boldsymbol{z}_k$ is only predicted in the KF using the last trained model for the first l epochs and for the last $q-l$ epochs KF without $\Delta\boldsymbol{z}_k$ correction is performed. Therefore the query points for every q epochs contains:

$$\boldsymbol{x}_* = \begin{bmatrix} \boldsymbol{x}_{*,k} & \boldsymbol{x}_{*,k+1} & \cdots & \boldsymbol{x}_{*,k+l} \end{bmatrix} \tag{4.16}$$

To avoid over-prediction of l epochs query data, and to improve the robustness of the online trained LS-SVM model due to its limited effectiveness, a constraint is introduced:

$$\begin{cases} \Delta\hat{\boldsymbol{z}}_{k+1|k}, \boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k+1|k}} & norm(\boldsymbol{X}_i - \boldsymbol{x}_{*,j}) < d_{th} \\ \boldsymbol{0} & otherwise \end{cases} \quad i = 1, \cdots, n \; j = k, \cdots, k+l \tag{4.17}$$

where

$\boldsymbol{X}_i$ is the ith training data;

$\boldsymbol{x}_{*,j}$ is the jth query data; and

d_{th} is the threshold for Euclidean distance between the training data and the query data.

When the Euclidean distance between the training data and the query data is smaller than the threshold, prediction is implemented, otherwise conventional KF is performed. When there is no unmodelled error $\Delta\boldsymbol{z}_k$ the LS-SVM based KF algorithm is reduced to the conventional KF as $\Delta\boldsymbol{z}_k$ derived from the measurement residual will have the same distribution as the noise $\boldsymbol{\varepsilon}_k$ in the measurement with zero-mean and covariance $\boldsymbol{R}_k$. Since the LS-SVM cannot directly output the variance $\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k+1|k}}$ for the predicted $\Delta\hat{\boldsymbol{z}}_{k+1|k}$, it has to be calculated separately. Due to the trained nonlinear LS-SVM model, the variance estimation method based on unscented transformation (UT) technique is used in this research. Assume that the variance of the query point $\boldsymbol{x}_{*,k}$ is represented by $\boldsymbol{P}_{\boldsymbol{x}_{*,k}}$. Then the predicted vector $\Delta\hat{\boldsymbol{z}}_{k+1|k}$ and its corresponding covariance matrix $\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k+1|k}}$ with the trained LS-SVM model $f(\boldsymbol{x})$ can be approximated by $2n_{\boldsymbol{x}}+1$ weighted sigma points:

$$
\begin{aligned}
\boldsymbol{\chi}_{0,k} &= \boldsymbol{x}_{*,k}, w_0 = \tau/(n+\tau) \\
\boldsymbol{\chi}_{\boldsymbol{r},\boldsymbol{k}} &= \boldsymbol{x}_{*,k} + \sqrt{n_{\boldsymbol{x}} + \boldsymbol{\tau}}\left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}_{*,k}}}\right)_{\boldsymbol{r}}, \boldsymbol{w}_{\boldsymbol{r}} = \mathbf{1}/[\mathbf{2}(n_{\boldsymbol{x}} + \boldsymbol{\tau})] \\
\chi_{\boldsymbol{r}+\boldsymbol{n},\boldsymbol{k}} &= \boldsymbol{x}_{*,k} - \sqrt{n_{\boldsymbol{x}} + \boldsymbol{\tau}}\left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}_{*,k}}}\right)_{\boldsymbol{r}}, \boldsymbol{w}_{\boldsymbol{r}+\boldsymbol{n}} = \mathbf{1}/[\mathbf{2}(n_{\boldsymbol{x}} + \boldsymbol{\tau})]
\end{aligned}
\tag{4.18}
$$

where

τ is a scaling parameter;

$n_{\boldsymbol{x}}$ is the query point dimension;

$\left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}_{*,k}}}\right)_r$ is the rth row of the column of the matrix square root of $\boldsymbol{P}_{\boldsymbol{x}_{*,k}}$; and

w_r is the weight associated with the rth sigma point.

The transformed covariance matrix $\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k+1|k}}$ is then given by the weighted outer product of the transformed points:

$$
\begin{aligned}
\boldsymbol{\chi}_{r,k+1|k} &= f\left(\boldsymbol{\chi}_{\boldsymbol{r},\boldsymbol{k}}\right) \\
\Delta\hat{\boldsymbol{z}}_{k+1|k} &= \sum_{i=0}^{2n_x} w_i \boldsymbol{\chi}_{i,k+1|k} \\
\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k+1|k}} &= \sum_{i=0}^{2n_x} w_i \left[\boldsymbol{\chi}_{i,k+1|k} - \Delta\hat{\boldsymbol{z}}_{k+1|k}\right]\left[\boldsymbol{\chi}_{i,k+1|k} - \Delta\hat{\boldsymbol{z}}_{k+1|k}\right]^T
\end{aligned}
\tag{4.19}
$$

However, directly using $\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k|k-1}}$ in the LS-SVM based KF could cause fluctuations in the estimation, and even lead to divergence due to rapid changes of the training data. Therefore in order to further improve the robustness of the LS-SVM based KF algorithm the Kalman gain $\boldsymbol{K}_k$ can be adjusted by a scale factor f:

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^T\left[\boldsymbol{H}_k\boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^T + \frac{\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k|k-1}} + \boldsymbol{R}_k}{f}\right]^{-1} \tag{4.20}$$

If the $\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k|k-1}}$ is predicted precisely, then $\frac{\left(\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k|k-1}}+\boldsymbol{R}_k\right)}{f}$ with f set to the value 2 is equivalent to $\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k|k-1}}$ which contains both the propagated covariance of the unmodelled error and the measurement noise.

4.3.2 **GPR Based KF Algorithm**

A GPR can be thought of as a "Gaussian over functions" (Nguyen-Tuong et al., 2009; Rasmussen and Williams, 2004). A GP is fully specified by its mean and covariance functions. The observed targets can be described by a zero-mean multivariate Gaussian distribution:

$$\boldsymbol{y} \sim N(\boldsymbol{0}, K(\boldsymbol{X},\boldsymbol{X}) + \sigma_n^2\boldsymbol{I}) \tag{4.21}$$

where

$\boldsymbol{y}$ is the aggregated output vector $\boldsymbol{y} = [y_1 \quad y_2 \quad \cdots \quad y_n]^T$;

$\boldsymbol{X}$ is the aggregated input matrix $\boldsymbol{X} = [\boldsymbol{x}_1^T \quad \boldsymbol{x}_2^T \quad \cdots \quad \boldsymbol{x}_n^T]^T$;

$K(\boldsymbol{X},\boldsymbol{X})$ is the covariance matrix; and

σ_n is the variance of output noise.

To calculate the $K_{i,j}$ elements of $K(\boldsymbol{X},\boldsymbol{X})$, a kernel function in the form of a squared exponential (SE) is commonly used:

$$k(\boldsymbol{x},\boldsymbol{x}') = \sigma_s^2\, exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}')\boldsymbol{W}(\boldsymbol{x}-\boldsymbol{x}')^T\right) \tag{4.22}$$

where σ_s^2 is the signal variance and $\boldsymbol{W}$ is a diagonal matrix of the length scales of each input dimension. The optimal value of hyperparameters of a GP with SE can be derived by maximising the log marginal likelihood $\underset{\boldsymbol{\theta}}{argmax}\{log(p(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta}))\}$ (Rasmussen and Williams, 2006). The joint distribution of the observed target values and predicted value for a query point $\boldsymbol{x}_*$ is given by:

$$\begin{bmatrix} \boldsymbol{y} \\ y_* \end{bmatrix} \sim N\left(0, \begin{bmatrix} K(\boldsymbol{X},\boldsymbol{X}) + \sigma_n^2\boldsymbol{I} & k(\boldsymbol{X},\boldsymbol{x}_*) \\ k(\boldsymbol{x}_*,\boldsymbol{X}) & k(\boldsymbol{x}_*,\boldsymbol{x}_*) \end{bmatrix}\right) \tag{4.23}$$

Thus, the predictive distribution over the output y_* becomes:

$$p(y_*|\boldsymbol{x}_*,\boldsymbol{X},\boldsymbol{y}) \sim N(k_*^T(K+\sigma_n^2\boldsymbol{I})^{-1}\boldsymbol{y}, k_{**} - k_*^T(K+\sigma_n^2\boldsymbol{I})^{-1}k_*) \tag{4.24}$$

with $k_* = k(\boldsymbol{x}_*,\boldsymbol{X})$, $k_{**} = k(\boldsymbol{x}_*,\boldsymbol{x}_*)$, and $K = K(\boldsymbol{X},\boldsymbol{X})$. The mean prediction of y_* and its uncertainty are $k_*^T(K+\sigma_n^2\boldsymbol{I})^{-1}\boldsymbol{y}$ and $k_{**} - k_*^T(K+\sigma_n^2\boldsymbol{I})^{-1}k_*$, respectively.

As with the LS-SVM based KF algorithm, the major limitation of training the GPR model online is its expensive computation of the inverse matrix $(K+\sigma_n^2\boldsymbol{I})^{-1}$ which yields a cost of $O(n^3)$. Thus, the same strategies used in LS-SVM based KF algorithm to reduce the computational cost are also utilised to train GPR models. The flowchart and detailed procedure for the model-training based KF algorithms are presented in Figure 4.1 and Table 4.1, respectively. This method can be readily extended to nonlinear KF methods, such as EKF, UKF and CKF, since the measurement residual sequences for the error model training are calculated after the update step of all the nonlinear KF methods and the unmodelled error correction can be seen as new measurement noise with $\Delta\hat{\boldsymbol{z}}_{k|k-1}$ mean and covariance $\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{k|k-1}}$. Since both the LS-SVM and GPR algorithms use the historical measurement residuals to predict the unmodelled measurement error, they can only be effective for the A-PL with smooth changes in the environment and in the system configuration.

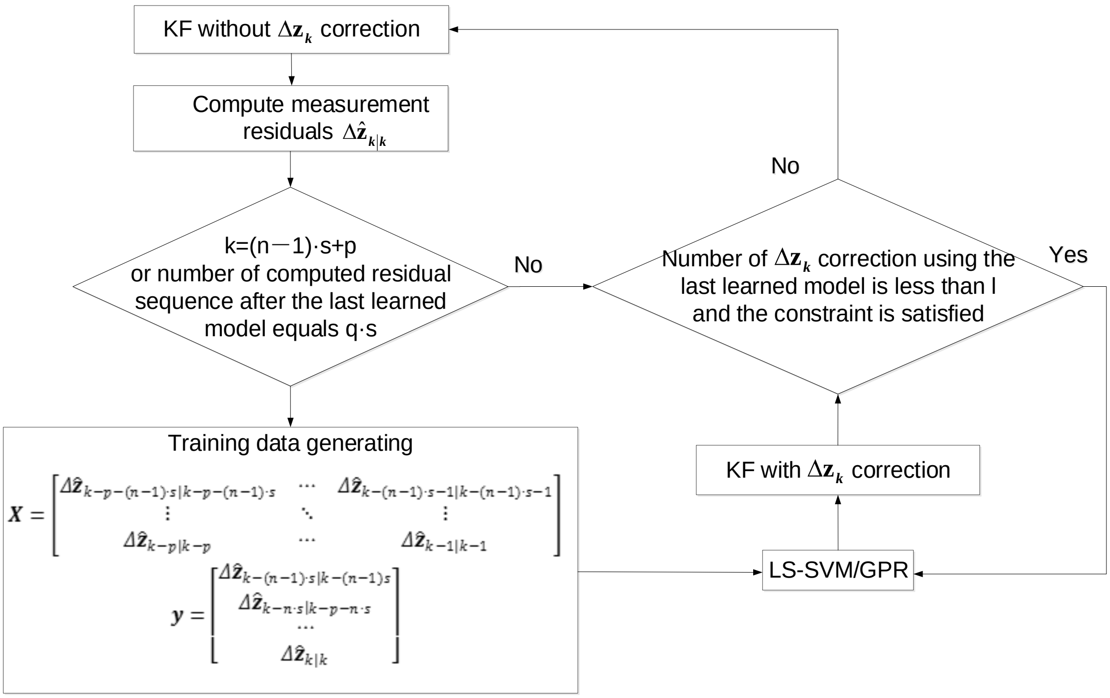


Figure 4.1 Model-learning based KF algorithm flowchart

Table 4.1 Model-learning based KF algorithm

1: Sequentially perform KF without $\Delta \boldsymbol{z}_k$ correction
$i < (n-1)\cdot s + p \quad or \quad k + l\cdot s < i < k + q\cdot s \quad when \quad k > (n-1)\cdot s + p$
2: Compute and save measurement residuals $\Delta\hat{\boldsymbol{z}}_{i\|i}$
3: LS-SVM/GPR model training using $(n-1)\cdot s + p$ training data
4: Predict $\Delta\hat{\boldsymbol{z}}_{i+1\|i}$ and $\boldsymbol{P}_{\Delta\hat{\boldsymbol{z}}_{i+1\|i}}$ based on the last trained model ($k < i \le k + l\cdot s$)
5: Perform KF with $\Delta \boldsymbol{z}_k$ correction
6: If there is no observation lost and the $q\cdot s$ measurement residuals have been calculated since the last trained model, update training data and jump to step 3; otherwise, jump to step 1

4.4 **Experiment Results and Analysis**

Both static and kinematic scenarios were investigated to validate the efficiency of the LS-SVM/GPR-based EKF algorithm for the A-PL with environment and system configuration changes. GPS observations from one of the IGS stations, with added coloured noise, were tested for the static scenario. In the case of the kinematic scenario GPS observations from a UAV experiment were used.

4.4.1 **Static Experiment**

GPS measurements at 30s intervals from the ALIC IGS station during the period day-of-year 277 and 278 in 2016 were selected for the static experiment. The data were processed using the GNSS PPP technique with float-ambiguity in the quasi-real-time mode. The parameters used for the LS-SVM/GPR-based KF were p=3, $n = 15$, $q = 10$, $l = 6$, $f = 3$, and $s = 2$. The width parameter σ in the LS-SVM was set to a value of 0.65. LS-SVM/GPR modelling for each observed satellite was assumed to be independent. To validate the effectiveness of the LS-SVM/GPR-based KF algorithm, all carrier phase measurements contaminated with coloured noise of different magnitudes were processed, as indicated in Table 4.2. The coloured noise was generated with a built-in “dsp.ColoredNoise” function in Matlab and directly added to the GNSS measurements. The noise magnitude added to pseudorange measurements was 3 orders of magnitude larger than that in the carrier phase measurements. The added coloured noise should not cause the measurement innovations to diverge. The measurement innovations are the difference between the measured and estimated ranges calculated with the propagated states. Thresholds d_{th} for pseudorange and carrier phase were set at 20 and 0.5, respectively. All the parameters involved in the model training were empirical values. The station positioning errors obtained by conventional PPP, PPP with LS-SVM and PPP with GPR, were compared with the IGS weekly solutions as shown in Figure 4.2. As can be seen from Figure 4.2, PPP with LS-SVM/GPR reduces the positioning fluctuations resulting from the rapid changing coloured noise, especially when the positioning accuracy has a tendency to decrease compared with the conventional PPP. The RMSE and STD representing positioning repeatability of the station position estimation calculated from 10h for all the algorithms are listed in Table 4.2. It can be seen that the performance in terms of positioning accuracy and repeatability derived from the PPP with LS-SVM/GPR is better than that from conventional PPP. However, the performance improvement for both the LS-SVM and GPR algorithms decreases with the increase of added noise magnitude. This is because that it becomes more difficult for the measurement residuals calculated in real-time to reflect the unmodelled errors when the coloured noises change more rapidly. In addition, PPP with LS-SVM has almost the same positioning performance as PPP with GPR. Considering the computational complexity resulting from the prediction uncertainty calculation, PPP with GPR is more promising to use for real-time applications than PPP with LS-SVM.

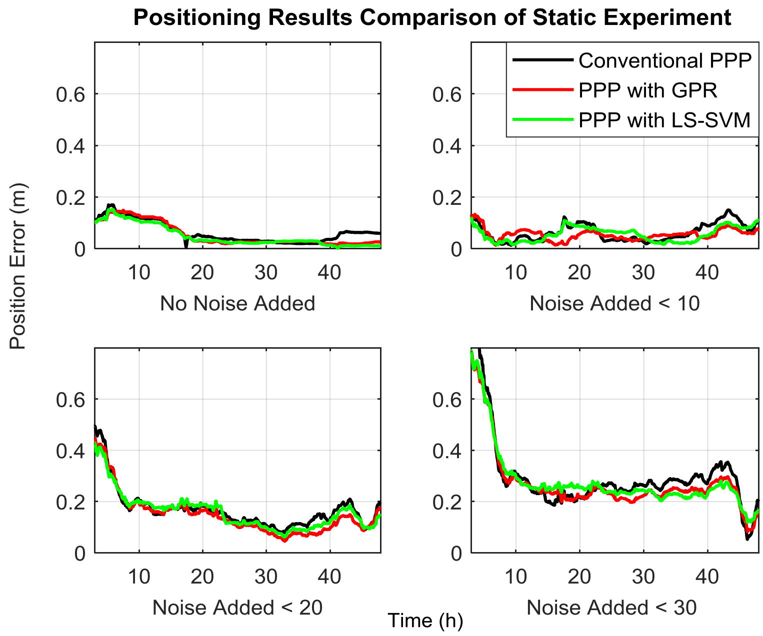


Figure 4.2 Positioning errors with different noise magnitudes

Table 4.2 Positioning accuracy with different noise magnitudes

Noise Magnitudes (cm)	Algorithms					
	Conventional PPP		PPP with LS-SVM		PPP with GPR	
	RMSE (cm)	STD (cm)	RMSE (cm)	STD (cm)	RMSE (cm)	STD (cm)
0	4	2	2	1	2	1
< 10	7	4	6	3	6	1
< 20	14	3	12	3	11	3
< 30	27	6	23	3	23	4

In theory, if there is no unmodelled error in the measurement model, the measurement innovations should have a zero-mean Gaussian distribution. This property of the innovations is a measure of filter performance. Therefore, to further validate the effectiveness of PPP with LS-SVM/GPR, the averaged mean and STD of GNSS measurement innovations over all observed satellites obtained by all the algorithms with no added noise are compared, as shown in Figure 4.3. The top plot in Figure 4.3 is the pseudorange innovations of all observed satellites, and the bottom plot is for carrier phase innovations. It can be seen that the pseudorange innovations obtained by PPP with LS-SVM/GPR have smaller mean than in the case of conventional PPP, which are more like zero-mean normally distributed quantities. The STD of pseudorange innovations achieved by PPP with GPR has the smallest value. PPP with LS-SVM barely changes the STD of pseudorange innovations compared with conventional PPP. As for the statistics of innovations of carrier phase, all three algorithms have almost the same mean and STD values. Thus, PPP with LS-SVM and GPR mainly improves the pseudorange innovations' whiteness property. This can be attributed to the lower accuracy of pseudorange measurements and their vulnerability to errors resulting from the changing environment and unstable dynamics compared with carrier phase measurements. To illustrate the comparison of pseudorange and carrier phase innovation zero-mean Gaussian distributions achieved by the three algorithms, one example for satellite PRN 20 is shown in Figure 4.4.

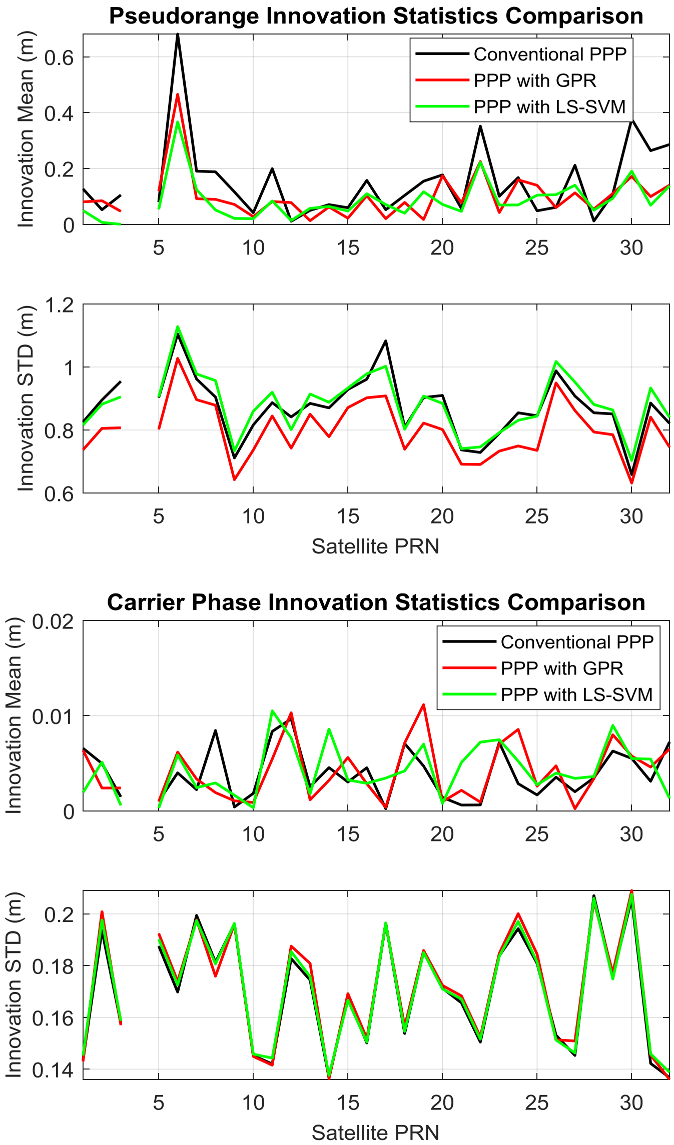


Figure 4.3 Measurement innovation statistics comparison

Table 4.3 Averaged innovation comparison of three algorithms

Algorithm	Pseudorange Innovation		Carrier Phase Innovation	
	Mean (m)	STD (m)	Mean (cm)	STD (m)
Conventional PPP	0.15	0.87	0.39	0.17
PPP with LS-SVM	0.09	0.88	0.40	0.17
PPP with GPR	0.10	0.80	0.41	0.17

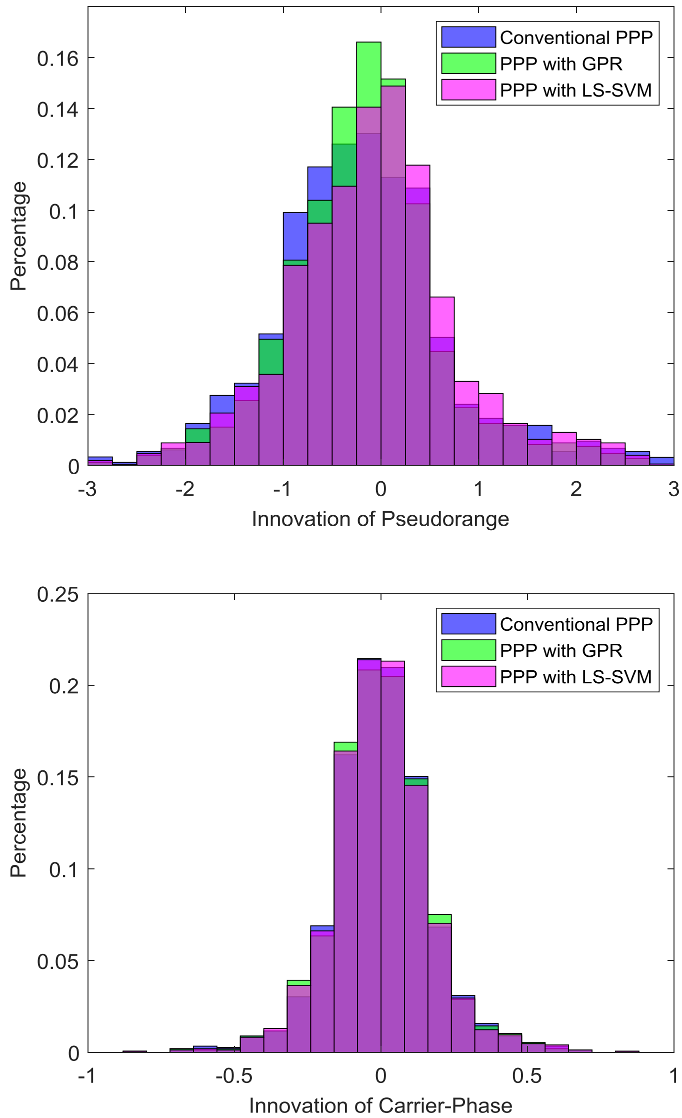


Figure 4.4 Distribution of measurement innovations

4.4.2 **Kinematic Experiment**

To investigate the efficiency of the LS-SVM/GPR-based KF algorithm for kinematic GNSS PPP scenarios a UAV experiment was carried out. Dual-frequency GPS data was collected by an OEM617 GPS receiver with sampling rate 10Hz. Figure 4.5 shows the UAV's horizontal trajectory. All three PPP algorithms were implemented epoch-by-epoch in post-processing mode. The parameters used in the kinematic experiment are the same as in the static experiment. Thresholds d_{th} for pseudorange and carrier phase are set as 10 and 0.1, respectively.

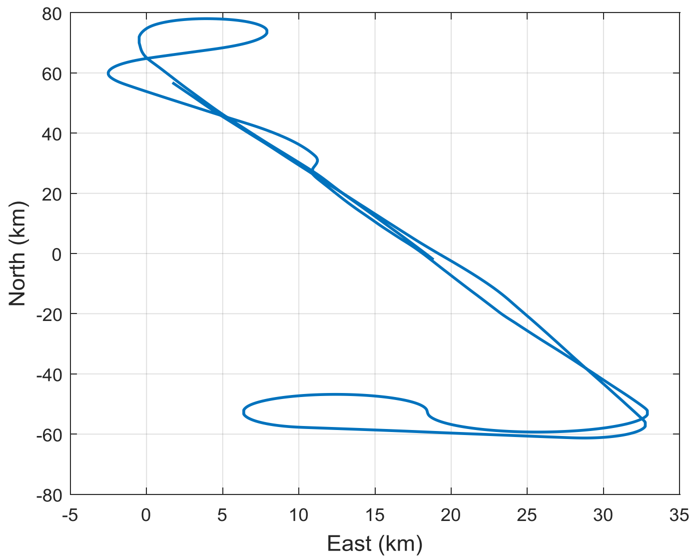


Figure 4.5 Horizontal motion trajectory

To compare the performance of the three PPP algorithms, the post-processed GPS-RTK positioning results were used as reference. Figure 4.6 is the comparison of the positioning error results of the three PPP algorithms. Table 4.4 gives the detailed positioning accuracy comparison calculated after 20 min into the flight. It can be seen that the positioning accuracy achieved by PPP with the LS-SVM and GPR algorithms are higher than for the conventional PPP, due to its ability to effectively reduce the deteriorated positioning accuracy as shown in Figure 4.6 at around the 8 min time tag.

Table 4.4 Positioning accuracy comparison

Algorithm	Positioning Errors (cm)
Conventional PPP	28

PPP with LS-SVM	22
PPP with GPR	20

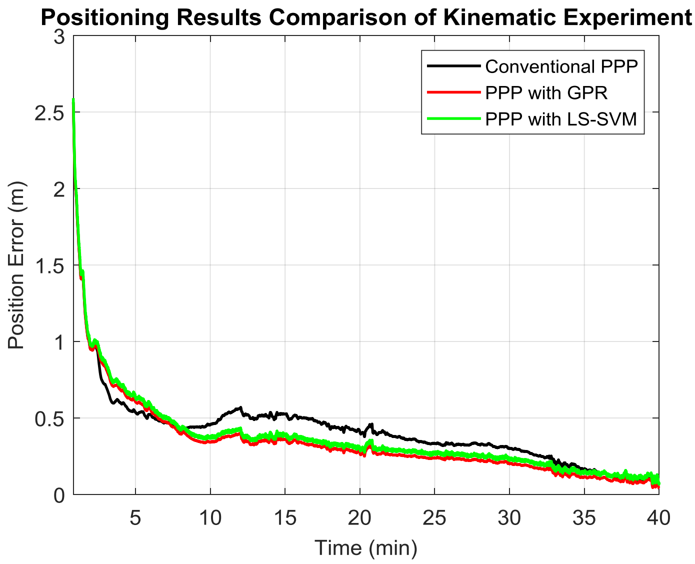


Figure 4.6 Positioning error comparison

The pseudorange and carrier phase innovation statistics of all observed GPS satellites from all three PPP algorithms are compared in Figure 4.7. The comparison of the measurement innovation distribution for satellite PRN 12 is shown as a typical example in Figure 4.8. It can be seen that the whiteness properties of both the pseudorange and carrier phase innovations are improved by PPP with LS-SVM and GPR. The improvement achieved for pseudorange measurements is higher than for carrier phase. In addition, it can be seen that PPP with GPR has a slightly higher improvement compared with PPP with LS-SVM, which further suggests that it is preferable to use GPR-based KF algorithm for real-time GNSS PPP applications.

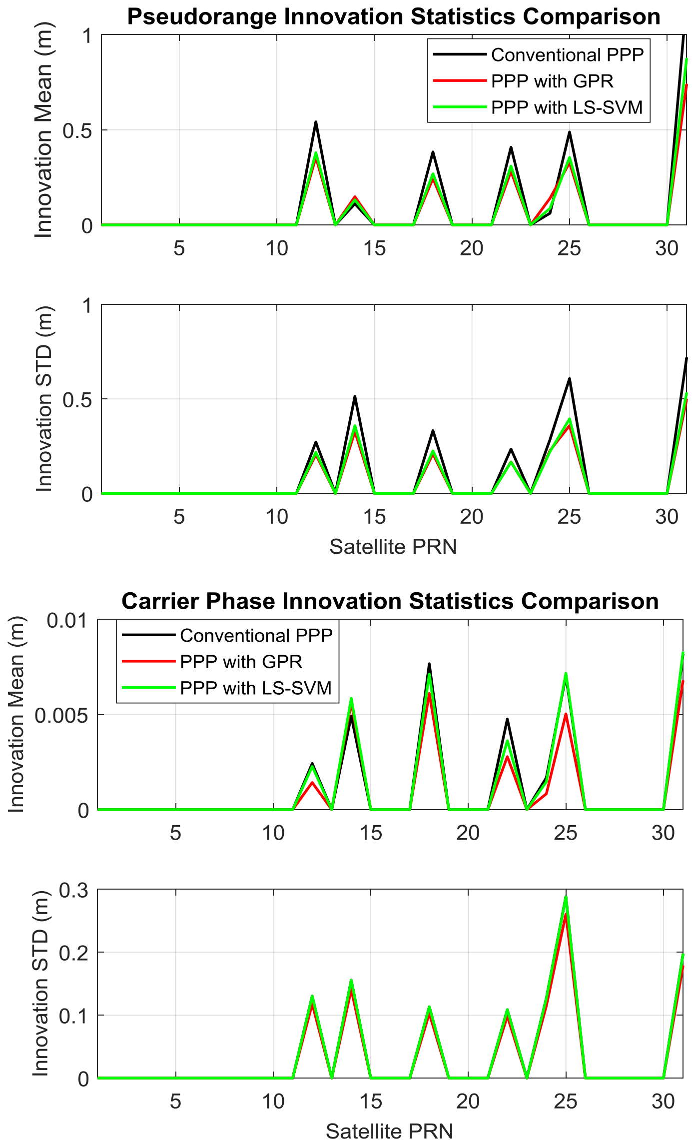


Figure 4.7 Measurement innovation statistics comparison

Table 4.5 Averaged innovation comparison of three algorithms

Algorithm	Pseudorange Innovation		Carrier-Phase Innovation	
	Mean (m)	STD (m)	Mean (m)	STD (m)
Conventional PPP	0.45	0.42	0.10	0.09
PPP with LS-SVM	0.34	0.30	0.08	0.07
PPP with GPR	0.32	0.28	0.07	0.07

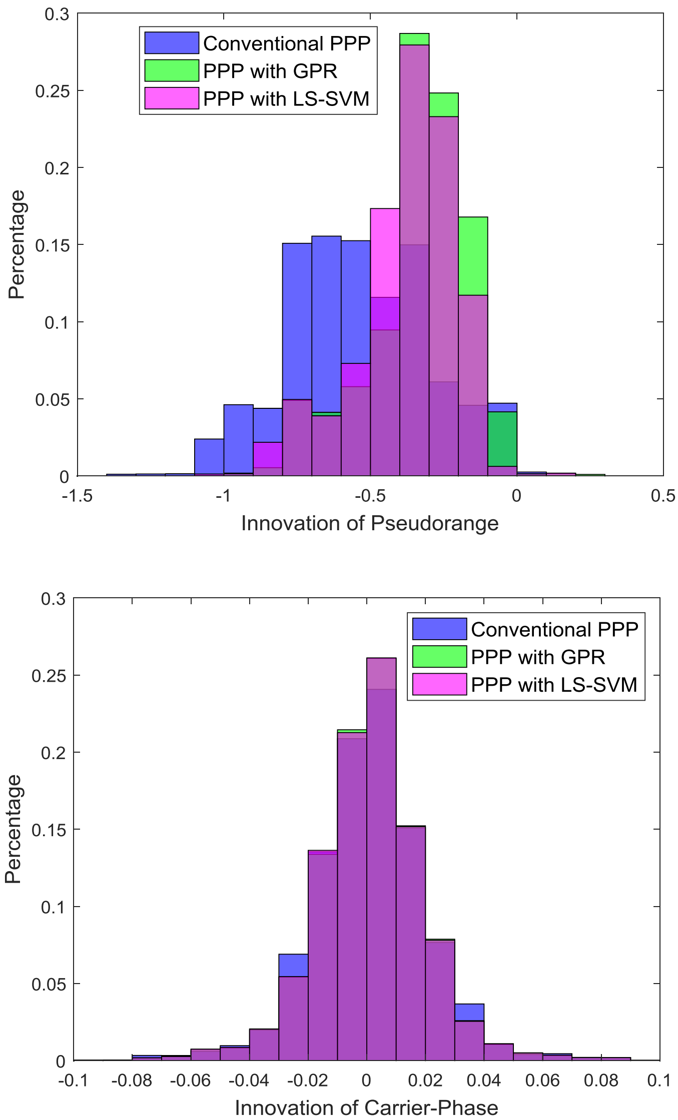


Figure 4.8 Distribution of measurement innovations

4.5 **Summary**

This chapter described two non-parametric model-learning based KF algorithms to deal with the unmodelled errors in GNSS observations. Independent LS-SVM/GPR models were trained in real-time for all observed satellites using the corresponding measurement residuals calculated in the KF. Due to the high computational cost of LS-SVM/GPR modelling resulting from inverse matrix calculation and limited training data, a more than one step sliding window was used. To avoid over-prediction using the trained LS-SVM/GPR model, a constraint on the query point was introduced. The PPP with LS-SVM/GPR for GNSS navigation was evaluated with both static and kinematic experiments. The results reveal that PPP with LS-SVM and GPR algorithms both effectively reduce the effect of unmodelled errors and do achieve better positioning performance compared with the conventional GNSS navigation algorithms. In addition, the GPR-based KF algorithm is more promising for real-time GNSS PPP applications due to its slightly better positioning performance and the fact that there is no requirement for a separate prediction uncertainty calculation.

Chapter 5 Assessment of Stochastic Models for Real-time Multi-GNSS PPP

5.1 Introduction

The GNSS PPP technique has been widely studied in recent years due to its capability for providing centimetre-to-decimetre level positioning accuracy with a single receiver (Ge et al., 2008; Geng et al., 2010; Knoop et al., 2017; Zumberge et al., 1997). However, this technique needs a long solution convergence time to achieve the desired accuracy, which makes it difficult for real-time positioning applications. Currently, the modernised GPS and GLONASS constellations, as well as the two new emerging constellations BeiDou and Galileo, make it possible to use a multi-constellation GNSS PPP technique, which has the potential to reduce the convergence time and improve positioning accuracy compared with PPP using a single GNSS. To facilitate the incorporation of new and modernised systems, the IGS initiated the Multi-GNSS Experiment (MGEX) in mid 2011 (Montenbruck et al. 2017). Various IGS Analysis Centres (ACs) and agencies have been routinely providing precise satellite orbit and clock products for BeiDou, Galileo, QZSS, in addition to GPS and GLONASS. Currently, the Centre National d'Études Spatiales (CNES) freely provides corrections for the GPS, GLONASS, Galileo, and BeiDou constellations, which facilitates real-time multi-GNSS PPP (Kazmierski et al., 2018). There have been a number of published studies focusing on further improving the performance of multi-GNSS PPP. For example, instead of using IF observations, multi-GNSS PPP with raw observations has demonstrated better performance in terms of convergence time and positioning accuracy (Liu et al., 2017; Lou et al., 2016). In addition, multi-GNSS PPP with AR has also been widely studied, a technique which is also able to shorten the convergence time and improve positioning accuracy compared to PPP AR using a single GNSS (Li et al., 2017; Odijk et al., 2015). The stochastic models used in these multi-GNSS PPP studies are all based on satellite elevation angle with fixed a priori variance that treat observations from different GNSSs equally, or pre-fixing the weight ratios for all the GNSSs to some empirical values (Gao et al., 2016; Li et al., 2015; Satirapod and Luansang, 2008; Zheng and Guo, 2016). However, these weighting strategies may not be adequate for real-time multi-GNSS PPP since the precision of observations from each GNSS are different. This is due to the heterogeneous real-time satellite orbit and clock products, the different signal structures, different levels of multipath and measurement noises, and atmospheric errors that are not elevation-angle dependent, and others (Zhang et al., 2018). To mitigate the influence of these unmodelled observation errors and achieve the best unbiased estimation of the unknown solution parameters, an appropriate stochastic model is essential, especially for real-time GNSS positioning applications. The stochastic model is often represented by a variance-covariance matrix (Aquino et al., 2009; Li et al., 2017; Shu et al., 2017; Wang, 1999).

To determine the optimal stochastic model for GNSS positioning adjustment, the a posteriori variance component estimation (VCE) based methods; such as the minimum norm quadratic unbiased estimator (MINQUE), best invariant quadratic unbiased estimates (BIQUE), restricted maximum likelihood (REML), least-squares VCE (LS-VCE) and Helmert VCE (HVCE), have been proposed (Amiri-Simkooei 2007; Amiri-Simkooei et al. 2013; Teunissen and Amiri-Simkooei 2008; Tiberius and Kenselaar 2000). BIQUE, MINQUE, REML, LS-VCE and HVCE have been shown to be identical with each other under the assumption of Gaussian distribution (Amiri-Simkooei 2007). In the VCE based methods, the stochastic model is generally assumed to be a linear combination of some known cofactor matrices with coefficients as the unknown variance components. The redundant measurement residuals that contain all the unmodelled errors calculated during positioning adjustment are used to estimate the variances of different types of GNSS observations. The estimated variances can then be used to scale the weight matrices with the corresponding weight ratios. However, the VCE based methods are often used for post-processed stochastic model estimation. To determine the stochastic model for real-time multi-GNSS PPP, the HVCE based method is investigated in this chapter.

As an alternative to VCE based methods that estimate the variances using measurement residuals, the stochastic model based on SISRE has been proposed for real-time GNSS PPP (Kazmierski et al., 2018). It takes into account the real-time satellite orbit and clock products by using a real-time SISRE as an observation quality indicator for the different GNSSs. However, the real-time SISRE based stochastic model was only assessed at the GNSS constellation level. The real-time SISRE based stochastic model at the level of individual satellites is also assessed in this chapter. In addition, to account for the varying measurement noise and multipath errors, the variance estimation of measurement noise and multipath has been widely studied. However this type of variance estimation method is often used for a single GNSS (Bisnath and Langley, 2001; Seepersad and Bisnath, 2015;

Spangenberg et al., 2010). In this chapter its application to real-time multi-constellation GNSS PPP is investigated.

5.2 **Multi-GNSS PPP Function Model**

In this chapter dual-frequency IF combinations of pseudorange P_{IF} and carrier phase L_{IF} are used for real-time multi-GNSS PPP:

$$\begin{gathered} P_{r,IF}^{s} = \rho_{r}^{s} + c(t_r - t^s) + m_r^s ZTD_r + b_{r,IF}^{s} + e_{r,IF}^{s} \\ L_{r,IF}^{s} = \rho_{r}^{s} + c(t_r - t^s) + m_r^s ZTD_r + \lambda_{IF}^{s}\left(N_{r,IF}^{s} + B_{r,IF}^{s}\right) + \varepsilon_{r,IF}^{s} \end{gathered} \tag{5.1}$$

where

ρ_r^s denotes the geometric range from receiver r to satellite s;

c is the vacuum speed of light;

t_r and t^s are the clock errors of receiver and satellite (in seconds), respectively;

ZTD_r is the ZTD and m_r^s is the corresponding mapping function;

$b_{r,IF}^{s} = b_{r,IF} - b_{IF}^{s}$ is the code hardware delay with receiver and satellite code hardware delays represented by $b_{r,IF}$ and b_{IF}^{s}, respectively;

$B_{r,IF}^{s} = B_{r,IF} - B_{IF}^{s}$ is the UPD with receiver and satellite UPD denoted as $B_{r,IF}$ and B_{IF}^{s}, respectively;

N_r^s is the integer ambiguity;

λ_{IF}^{s} is the wavelength of the IF combination (in metres); and

e_r^s and ε_r^s denote measurement noise and multipath for the pseudorange and carrier phase measurements, respectively.

Due to different frequencies and signal structures for each GNSS, the receiver biases for different GNSSs are different in a multi-GNSS receiver (Li et al., 2015). By treating GPS as the reference constellation, ISBs are introduced for other GNSSs and estimated as constants along with other parameters. For GLONASS satellites with different frequency factors, IFBs have also to be considered in the PPP algorithm. They are generally modelled as frequency-specific parameters. To get the full rank function model, GLONASS receiver code biases can be modelled as a linear function of the frequency numbers (Liu et al. 2017):

$$b_r^{SR_k} = b_r^{SR_0} + k \cdot \Omega_r^{SR} \tag{5.2}$$

where Ω_r^{SR} is the new code IFB on the corresponding frequency band and $b_r^{SR_0}$ is the code hardware delay for GLONASS satellite with frequency number 0.

The above IF observation function model can be linearised as follows:

$$\begin{gathered} \Delta P_{r,IF}^s = e_r^s \cdot \Delta\boldsymbol{x} + c(t_r - t^s) + m_r^s ZTD_r + b_{r,IF}^s + e_{r,IF}^s \\ L_{r,IF}^s = e_r^s \cdot \Delta\boldsymbol{x} + c(t_r - t^s) + m_r^s ZTD_r + \lambda_{IF}^s\left(N_{r,IF}^s + B_{r,IF}^s\right) + \varepsilon_{r,IF}^s \end{gathered} \tag{5.3}$$

where

$\Delta P_{r,IF}^s$ and $\Delta L_{r,IF}^s$ are the observed-minus-predicted code and phase observations, respectively; e_r^s is the unit vector from satellite *s*. to receiver r; and

$\Delta\boldsymbol{x}$ is the coordinate increment with respect to an approximate value.

The linearised observation model can also be written in the following matrix form:

$$\boldsymbol{Z}_k = \boldsymbol{H}_k \boldsymbol{X}_k + \boldsymbol{\eta}_k \tag{5.4}$$

where

$\boldsymbol{Z}_k$ denotes the vector of observed-minus-predicted GNSS measurements;

$\boldsymbol{H}_k$ denotes the design matrix of unknown parameters $\boldsymbol{X}_k$ at epoch k; and

$\boldsymbol{\eta}_k$ represents the observation noise with the variance-covariance matrix $\boldsymbol{R}_{\boldsymbol{Z}_k}$.

$\boldsymbol{R}_{\boldsymbol{Z}_k}$ reflects the precision and correlation of different types of observations, i.e. the stochastic model for GNSS observables. To achieve the best unbiased estimation of the unknown parameters for real-time multi-GNSS PPP based on the LS- or KF-based methods, it is necessary to use an appropriate variance-covariance matrix.

5.3 Stochastic Models for Real-Time Multi-GNSS PPP

Two types of stochastic models for real-time multi-GNSS PPP were investigated: the a priori model and the real-time estimated stochastic model. To limit the increase in computational complexity for real-time stochastic model estimation and to facilitate the performance comparison of these two types of stochastic models, time correlation and cross-correlation between different frequencies and types of observations are ignored. Only variances among different GNSSs and observations are taken into account. Therefore the stochastic model can be expressed as in the following form:

$$\boldsymbol{R}_{\boldsymbol{Z}_i} = \sigma_i^2 \boldsymbol{P}_{\boldsymbol{Z}_i}^{-1} = \begin{bmatrix} \sigma_G^2 \boldsymbol{P}_{\boldsymbol{Z}_G}^{-1} & 0 & 0 & 0 \\ 0 & \sigma_R^2 \boldsymbol{P}_{\boldsymbol{Z}_R}^{-1} & 0 & 0 \\ 0 & 0 & \sigma_E^2 \boldsymbol{P}_{\boldsymbol{Z}_E}^{-1} & 0 \\ 0 & 0 & 0 & \sigma_C^2 \boldsymbol{P}_{\boldsymbol{Z}_C}^{-1} \end{bmatrix}, i = G, R, E, C \qquad (5.5)$$

where σ_i^2 and $\boldsymbol{P}_{\boldsymbol{Z}_i}$ are the a priori variance of unit weight and the weight matrix, respectively. They are also referred as the variance factor σ_i^2 and cofactor matrix $\boldsymbol{P}_{\boldsymbol{Z}_i}$ in the literature. In the case of LS-based estimation methods the estimated unknown parameters are invariant with respect to changes in σ_i^2, but affected by changes in $\boldsymbol{P}_{\boldsymbol{Z}_i}$ (Teunissen and Amiri-Simkooei, 2008).

5.3.1 A Priori Stochastic Model

To mitigate the unmodelled elevation-related errors, such as ionospheric, tropospheric and multipath errors, the a priori stochastic model based on the satellite elevation angle is often used for GNSS-based positioning algorithms due to its simplicity and low computational complexity (Howind et al., 1999). When using this a priori elevation-dependent stochastic model for multi-GNSS positioning, different a priori variances of unit weight for different GNSSs also need to be considered due to different signal structures, different levels of measurement noise and multipath, etc. However, there are still some measurement residuals that are not elevation-dependent, resulting from input GNSS corrections such as the precise satellite orbit and clock products. To take into account these errors, the a priori stochastic model with real-time precise multi-GNSS SISRE can be used to determine the weight ratios among different GNSSs.

5.3.1.1 A Priori Stochastic Model Based on Satellite Elevation Angle

There are two commonly used elevation angle based weight matrices, based on trigonometric and exponential forms (Gao et al., 2011; Yu and Gao, 2017):

$$\begin{aligned} p_{\boldsymbol{Z}_i^{n_i}} &= a^2 + b^2/\left(\sin\theta_i^{n_i}\right)^2 \\ p_{\boldsymbol{Z}_i^{n_i}} &= \left[a_0 + a_1 \cdot \exp\left(-\theta_i^{n_i}/\theta_0\right)\right]^2 \end{aligned} \tag{5.6}$$

where

a, b, a_0, a_1 and θ_0 are constant empirical values;

$\theta_i^{n_i}$ is the n_ith satellite elevation angle of the *i*th GNSS; and

$p_{\boldsymbol{Z}_i^{n_i}}$ is the *i*th diagonal component of the weight matrix $\boldsymbol{P}_{\boldsymbol{Z}_i}^{-1}$.

The reference values for the constant empirical values can be found in, for example, Han (1997). Since there are some measurement errors that are not elevation-dependent, resulting from signal diffraction and receiver characteristics, a carrier-to-noise ratio (C/N_0) based stochastic model has been proposed to describe the carrier phase variance $\sigma^2_{\boldsymbol{Z}_i^{n_i}}$ (Brunner et al., 1999), which is generally expressed as:

$$\sigma^2_{\boldsymbol{Z}_i^{n_i}} = C_i \cdot 10^{-\left(\frac{C/N_0}{10}\right)} \tag{5.7}$$

where C_i is a constant value (in $m^2 Hz$). The C/N_0 based stochastic model can also be combined with the elevation-based stochastic model (Luo et al., 2009). One example of such a combined stochastic model $p^{com}_{\boldsymbol{Z}_i^{n_i}}$ can be written as (Gao et al., 2011):

$$p^{com}_{\boldsymbol{Z}_i^{n_i}} = (9/s)^2 p_{\boldsymbol{Z}_i^{n_i}} \tag{5.8}$$

where s is the scaling factor, which can be expressed as:

$$s = \begin{cases} 9 & int\left(\frac{C/N_0}{5}\right) > 9 \\ int\left(\frac{C/N_0}{5}\right) & else \end{cases} \qquad (5.9)$$

The above weight matrix can be used to determine the weight ratios for satellites with the same type of observations. For multi-GNSS PPP positioning using both pseudorange and carrier phase measurements, the variance factor of pseudorange measurements is often assumed to be 10^4 times larger than that of carrier phase measurements (Giorgi and Teunissen, 2012). For observations from different GNSSs with multiple frequencies, one commonly used approach is to directly weight GNSS observations with empirical variance factor ratios by analysing their measurement noises and multipath effects (Cai et al., 2016; Wang et al., 2018).

5.3.1.2 A Priori Stochastic Model Based on Real-time SISRE

To mitigate the influence of errors in the input satellite orbit and clock products on the GNSS PPP positioning, the a priori stochastic model based on real-time precise GNSS SISRE was developed (Kazmierski et al., 2018). The real-time precise GNSS SISRE can be defined in a similar way to the traditional SISRE quantity for assessing constellation-specific positioning performance using broadcast ephemerides. The real-time precise GNSS SISRE can be calculated based on the global-average instantaneous SISREs (Montenbruck et al., 2018):

$$\sigma_{SISRE,i}^{s_i} = \sqrt{[w_R(\Delta r_R - \Delta\bar{r}_R) - (\Delta\tau - \Delta\bar{\tau})]^2 + w_{A,C}(\Delta r_A^2 + \Delta r_C^2)} \qquad (5.10)$$

where

Δr_R, Δr_A and Δr_C represent the real-time orbit error in the radial-, along- and cross- directions, respectively;

$\Delta\tau$ is the real-time clock error; and

w_R, w_A and w_C are weight factors specific to the satellite orbit radius.

The values of these weight factors for different satellites from different GNSSs are listed in Table 5.1, which are the same as those used for the conventional SISRE calculation with the broadcast ephemerides. $\Delta\bar{r}_R$ and $\Delta\bar{\tau}$ are the constellation mean values of orbit radius and clock errors, respectively. These two mean values are used to remove errors, due to time system offset, group delay offset and pseudorange error, common to all satellites within a single GNSS. As the real-time precise GNSS SISRE is a satellite- and epoch-specific value, it can also be used to compute real-time precise root mean squared (RMS) GNSS SISRE $\sigma_{SISRE,i}$ over all satellites for each GNSS:

$$\sigma_{SISRE,i} = \mathrm{RMS}\left(\sigma_{SISRE,i}^{s_i}\right) \tag{5.11}$$

Table 5.1 Weight factors used in SISRE computation (Montenbruck et al. 2015)

GNSS	w_R	$w_{A,C}$
GPS	0.98	1/49
GLONASS	0.98	1/45
Galileo	0.98	1/61
BeiDou (IGSO/GEO)	0.99	1/126
BeiDou (MEO)	0.98	1/54

Using the real-time precise GNSS SISRE it is possible to weight all observations at the satellite level by comparing all the individual satellite SISRE values. In addition, it is also possible to obtain the weight ratio at the GNSS level by comparing the real-time precise RMS GNSS SISRE values. In this chapter the real-time precise RMS SISRE of GPS is used as a reference value. Therefore the weight matrix on the satellite level can be calculated as:

$$\bar{\boldsymbol{P}}_{\boldsymbol{Z}_i} = \frac{\left(\sigma_{SISRE,i}^{s_i}\right)^2}{\sigma_{SISRE,G}^2} \boldsymbol{P}_{\boldsymbol{Z}_i} \tag{5.12}$$

Similarly, the weight matrix at the GNSS level can be calculated as:

$$\bar{\boldsymbol{P}}_{\boldsymbol{Z}_i} = \frac{\sigma_{SISRE,i}^2}{\sigma_{SISRE,G}^2} \boldsymbol{P}_{\boldsymbol{Z}_i} \tag{5.13}$$

The weight matrix can be calculated on a daily basis for real-time multi-GNSS PPP or even in real-time if it is possible to provide the users with real-time monitored precise SISRE values.

5.3.2 Real-time Estimated Stochastic Model

The a priori stochastic models described above only take into account measurement errors based on empirical experiment data to determine the weight ratios of different types observations or observations from different GNSSs. To obtain the optimal weight ratios of different types of observations, the VCE based methods can be used by analysing redundant measurement residuals that contain *all* the unmodelled measurement errors. One widely used VCE based method, the HVCE, for estimating the stochastic model for real-time multi-GNSS PPP is described in the following section. In addition, the real-time variance estimation for pseudorange noise and multipath errors is also introduced.

5.3.2.1 Real-time Variance Estimation Based on HVCE

According to the HVCE theory, the variance of observations from different GNSSs at one epoch can be estimated posteriorly using the measurement residuals (Zhou et al., 2008). The measurement residuals $\boldsymbol{V}_k$ can be obtained as follows:

$$\boldsymbol{V}_k = \begin{bmatrix} \boldsymbol{V}_{G,k} \\ \boldsymbol{V}_{R,k} \\ \boldsymbol{V}_{E,k} \\ \boldsymbol{V}_{C,k} \end{bmatrix} = \begin{bmatrix} \boldsymbol{H}_{G,k} \\ \boldsymbol{H}_{R,k} \\ \boldsymbol{H}_{E,k} \\ \boldsymbol{H}_{C,k} \end{bmatrix} \boldsymbol{X}_k - \begin{bmatrix} \boldsymbol{Z}_{G,k} \\ \boldsymbol{Z}_{R,k} \\ \boldsymbol{Z}_{E,k} \\ \boldsymbol{Z}_{C,k} \end{bmatrix} \tag{5.14}$$

With the known a priori weight matrices of the four GNSSs, the unit weight variances for the four GNSSs can be estimated as:

$$\begin{bmatrix} \hat{\sigma}^2_{G,k} \\ \hat{\sigma}^2_{R,k} \\ \hat{\sigma}^2_{E,k} \\ \hat{\sigma}^2_{C,k} \end{bmatrix} = \begin{bmatrix} s_{GG} & s_{GR} & s_{GE} & s_{GC} \\ s_{RG} & s_{RR} & s_{RE} & s_{RC} \\ s_{EG} & s_{ER} & s_{EE} & s_{EC} \\ s_{CG} & s_{CR} & s_{CE} & s_{CC} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{V}^T_{G,k} \boldsymbol{P}_{\boldsymbol{Z}_{G,k}} \boldsymbol{V}_{G,k} \\ \boldsymbol{V}^T_{R,k} \boldsymbol{P}_{\boldsymbol{Z}_{R,k}} \boldsymbol{V}_{R,k} \\ \boldsymbol{V}^T_{E,k} \boldsymbol{P}_{\boldsymbol{Z}_{E,k}} \boldsymbol{V}_{E,k} \\ \boldsymbol{V}^T_{C,k} \boldsymbol{P}_{\boldsymbol{Z}_{C,k}} \boldsymbol{V}_{C,k} \end{bmatrix} \tag{5.15}$$

$$\begin{gathered} s_{ii} = n_i - 2tr(\boldsymbol{N}^{-1}\boldsymbol{N}_i) + tr(\boldsymbol{N}^{-1}\boldsymbol{N}_i)^2 \\ s_{ij} = s_{ji} = tr\left(\boldsymbol{N}^{-1}\boldsymbol{N}_i\boldsymbol{N}^{-1}\boldsymbol{N}_j\right) \quad j \neq i \quad i,j = G,R,E,C \\ \boldsymbol{N} = \boldsymbol{H}^T\boldsymbol{P}_{\boldsymbol{Z}}^{-1}\boldsymbol{H} \quad \boldsymbol{N}_i = \boldsymbol{H}_i^T\boldsymbol{P}_{\boldsymbol{Z}_i}^{-1}\boldsymbol{H}_i \end{gathered} \tag{5.16}$$

where

n_i is the number of measurements from the *i*th GNSS; and

$tr(\cdot)$ denotes the trace operation on the matrix.

If the a priori weight matrices for the four GNSSs are inaccurate, then the estimated unit weight variances are not equal. In this case, the weight matrix can be scaled by a factor $\hat{\lambda}_{i,k}$:

$$\widehat{\boldsymbol{P}}_{\boldsymbol{Z}_{i,k}} = \hat{\lambda}_{i,k}\boldsymbol{P}_{\boldsymbol{Z}_{i,k}} \tag{5.17}$$

where $\widehat{\boldsymbol{P}}_{\boldsymbol{Z}_{i,k}}$ is the new weight matrix. $\hat{\lambda}_{i,k}$ can be iteratively updated based on the estimated unit weight variances $\hat{\sigma}_{i,k}^2$:

$$\hat{\lambda}_{i,k} = \frac{a\lambda_{i,k}}{\hat{\sigma}_{i,k}^2} \tag{5.18}$$

where a is a constant, which is usually set equal to $\hat{\sigma}_{G,k}^2$. The initial value of the scale factor $\lambda_{i,k}$ is set to 1. With the estimated unit weight variances $\hat{\sigma}_{i,k}^2$ and scaled weight matrix $\widehat{\boldsymbol{P}}_{\boldsymbol{Z}_{i,k}}$, the state $\boldsymbol{X}_k$ can be re-estimated. Equations (5.14) to (5.18) are iteratively recalculated until the differences of all the estimated unit weight variances are less than a given threshold. The HVCE based variance estimation method is also able to calculate weight ratios between pseudorange and carrier phase observations in a single GNSS with the fixed ambiguities. However this method is difficult to implement for real-time applications due to the inverse operation for estimating the unit weight variances. To reduce the computational cost, simplified formulas for VCE can be used. One widely used simplified estimator is the Förstner method (Bähr et al., 2007). This method is derived with an assumption that the iteration in HVCE will be finally converged. Since the stochastic model considered in this chapter has block diagonal structure, the simplification of equation (5.15) is given as (Wang et al., 2009):

$$\hat{\sigma}_{i,k}^2 = \frac{\boldsymbol{V}_{i,k}^T\boldsymbol{P}_{\boldsymbol{Z}_{i,k}}\boldsymbol{V}_{i,k}}{r_{i,k}} \tag{5.19}$$

where $r_{i,k} = n_i - tr(\boldsymbol{N}^{-1}\boldsymbol{N}_i)$ denotes the redundancy contribution of the ith GNSS, which can also be seen as the number of degrees of freedom. Although Förstner's VCE estimates are identical to those of HVCE on the premise of convergence, independence of observations from each GNSS and good approximate values of the a priori variance of unit weight are required for (fast) convergence. In addition, the HVCE method may generate negative variance estimates, which can also be avoided by using appropriate the a priori variance of unit weight. To further reduce the computational time of the matrix product $\boldsymbol{N}^{-1}\boldsymbol{N}_i$ in equation (5.19), the unit weight variance estimation can be simplified as (Bähr et al., 2007):

$$\hat{\sigma}_{i,k}^2 = \frac{\boldsymbol{V}_{i,k}^T \boldsymbol{P}_{\boldsymbol{Z}_{i,k}} \boldsymbol{V}_{i,k}}{n_i - u_{i,k}} = \frac{\boldsymbol{V}_{i,k}^T \boldsymbol{P}_{\boldsymbol{Z}_{i,k}} \boldsymbol{V}_{i,k}}{n_i - \frac{n_i}{n} u} \tag{5.20}$$

where

$u_{i,k}$ is the decomposed number of unknowns of the ith GNSS;

$n = \sum_i n_i$ is the overall number of observations; and

u is the overall number of unknown parameters.

When there are outliers in the observations, the scaled weight matrix derived from the above VCE methods could be dramatically affected, which is not robust enough to be used directly for parameter estimation (Yang et al., 2002). To reduce the effect of outliers, an equivalent weight matrix $\bar{\boldsymbol{P}}_{\boldsymbol{Z}_k}$ based on the IGG-III can be used instead. Its diagonal elements are calculated as follows:

$$\bar{p}_{\boldsymbol{Z}_k}^{ii} = \begin{cases} p_{\boldsymbol{Z}_k}^{ii} & \left|\bar{\boldsymbol{V}}_{i,k}\right| \leq k_0 \\ p_{\boldsymbol{Z}_k}^{ii} \dfrac{k_0}{\left|\bar{\boldsymbol{V}}_{i,k}\right|} \left(\dfrac{k_1 - \left|\bar{\boldsymbol{V}}_{i,k}\right|}{k_1 - k_0}\right)^2 & k_0 < \left|\bar{\boldsymbol{V}}_{i,k}\right| \leq k_1 \\ 0 & k_1 < \left|\bar{\boldsymbol{V}}_{i,k}\right| \end{cases} \tag{5.21}$$

where

k_0 and k_1 are two constants, which are usually chosen as being in the ranges $2.0 - 3.0$ and $4.5 - 8.5$, respectively; and

$p_{\boldsymbol{Z}_k}^{ii}$ is the ith diagonal element of the weight matrix $\boldsymbol{P}_{\boldsymbol{Z}_k}$.

When sufficient observation redundancy for a single epoch cannot be satisfied there is insufficient robustness to estimate the unit weight variances on an epoch-by-epoch basis. Therefore in order to ensure reliable stochastic model estimation, one commonly used strategy is to estimate the unit weight variances using a multi-epoch model (Tiberius and Kenselaar, 2003). It is assumed that the stochastic model is the same over m epochs and the time correlation is absent. Then the estimated unit weight variance can be obtained by averaging those estimated per epoch:

$$\hat{\sigma}_{i,k}^2 = \frac{\sum_{t=k-m+1}^{k} \hat{\sigma}_{i,t}^2}{m} \tag{5.22}$$

The above HVCE based methods could all be implemented for real-time multi-GNSS PPP. However, one problem is that the different ranges of the calculated multi-GNSS residuals resulting from the different systematic errors need to be taken into account. This can be solved by unifying all the measurement residuals $\boldsymbol{V}_k$ with the corresponding unit weight variances $\sigma_{i,k}^2$ and weight matrices $\boldsymbol{Q}_i$ (Zhang et al., 2018):

$$\bar{\boldsymbol{V}}_{i,k} = \frac{\boldsymbol{V}_{i,k}}{\sigma_{i,k}^2 \sqrt{\boldsymbol{Q}_i}} \tag{5.23}$$

with

$$\boldsymbol{Q}_i = \boldsymbol{P}_{\boldsymbol{Z}_{i,k}}^{-1} - \boldsymbol{H}_{i,k} \boldsymbol{N}_i \boldsymbol{H}_{i,k}^T \tag{5.24}$$

The standardised measurement residuals can then be used in HVCE based methods for real-time multi-GNSS PPP.

5.3.2.2 Real-time Variance Estimation for Pseudorange Noise and Multipath

There are two commonly used approaches to mitigate the influence of pseudorange multipath errors on positioning: modelling pseudorange multipath errors and removing them from the observations, or pseudorange multipath error-based stochastic modelling (Seepersad and Bisnath, 2015). In this chapter the latter approach is considered. In order to effectively analyse the pseudorange multipath errors, all geometric contributions, and ionospheric and tropospheric delays in pseudorange measurements have to be cancelled out. The well-known multipath linear combination, i.e. geometric-free IF pseudorange combination, is used to estimate the variance of pseudorange multipath errors (Lei et al., 2017). The geometry-free IF measurements can be generalised as follows:

$$\begin{gathered} P_{IF} = g + mp_{P,IF} + b_{IF} + e_{IF} \\ L_{IF} = g + \lambda_{IF} N_{IF} + mp_{L,IF} + B_{IF} + \varepsilon_{IF} \end{gathered} \tag{5.25}$$

where

g is a geometric term including the geometric range between the receiver and satellite, tropospheric delay, receiver clock error and satellite clock error; and

$mp_{P,IF}$ and $mp_{L,IF}$ are the pseudorange and carrier phase multipath, respectively.

To adaptively calculate the stochastic model of pseudorange multipath errors, a linear combination of pseudorange and carrier phase can be used:

$$M_P = P_{IF} - L_{IF} \tag{5.26}$$

Compared to pseudorange multipath errors, the carrier phase multipath error is approximately two orders of magnitude smaller, and can therefore be neglected. Then the expectation and dispersion of M_P are:

$$\begin{gathered} E(M_P) \approx mp_{P,IF} + b_{IF} - \lambda_{IF} N_{IF} - B_{IF} \\ D(M_P) \approx \hat{\sigma}_P^2 \end{gathered} \tag{5.27}$$

Assuming that there are no cycle slips in the carrier phase, the phase ambiguities and the receiver hardware delays can be considered to be constants. Then $\hat{\sigma}_P^2$ can be estimated in real-time with a sliding window by subtracting the mean value $\bar{M}_P$ from the sequences of M_P:

$$\hat{\sigma}_{P,k}^2 = \frac{\sum_{t=k-m+1}^{k} \left(M_{P,k-t} - \bar{M}_P \right)^2}{m-1} \tag{5.28}$$

where m is the number of epochs in the sliding window. m has to be carefully chosen by trading off the periodic influence of the multipath effect and tracking sensitivity for time-varying pseudorange errors. Instead of directly calculating $\hat{\sigma}_P^2$ based on equation (5.28), a fading factor β $(0 < \beta \leq 1)$ can be introduced to reduce the contribution of earlier information (Zheng and Guo, 2016):

$$\hat{\sigma}_{P,k}^2 = \frac{\sum_{t=k-m+1}^{k} \beta_{k-t} \left(M_{P,k-t} - \bar{M}_P\right)^2}{\sum_{t=k-m+1}^{k} \beta_{k-t}} \tag{5.29}$$

where β_{k-t} can be set to $\frac{9}{9+k-t}$. The weight matrix used for multi-GNSS PPP can then be directly scaled with the estimated multipath error variances for each observed satellite.

5.4 **Experiment and Result Analysis**

To evaluate the stochastic models for the A-PL positioning with real-time multi-GNSS PPP, observations collected from 14 stations in the MGEX network were tested. These 14 stations are globally distributed as shown in Figure 5.1.

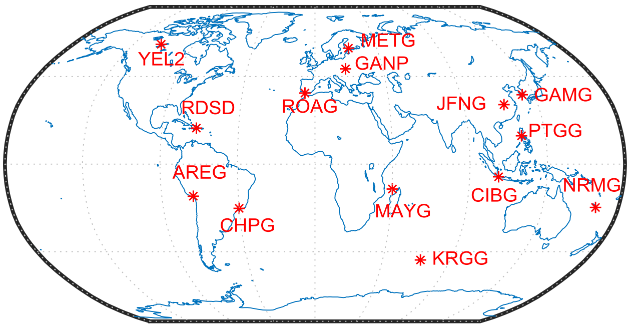


Figure 5.1 Distribution of selected MGEX stations

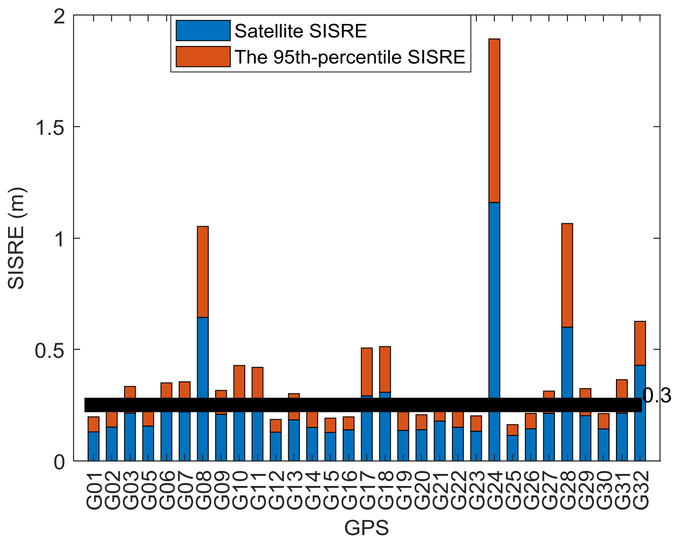
One-week GNSS data for only three GNSSs GPS, GLONASS and Galileo, observed from DOY 60 to 66 in 2018 were processed on an epoch-by-epoch basis as there was no BeiDou observation for these stations during this period. Some details of the multi-GNSS PPP processing strategy are summarised in Table 5.2.

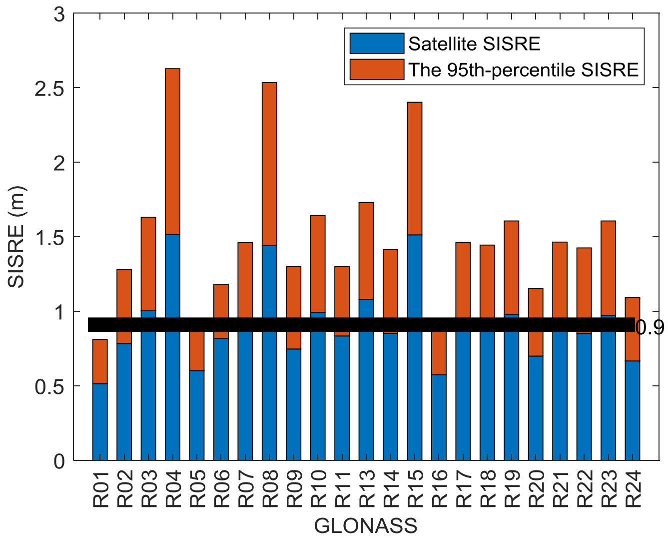
Table 5.2 Multi-GNSS PPP data processing strategy

Item	Models
Observations	IF pseudorange and carrier phase
Frequencies	GPS: L1/L2; GLONASS: L1/L2; Galileo: E1/E5a

Sampling rate	30s
Elevation cut-off angle	7°
A priori stochastic model for a single GNSS	A priori standard deviation of 0.003m and 0.3m for carrier phase and pseudorange observations; trigonometric elevation dependent weight matrix
Satellite antenna correction	Correct PCOs and PCVs for GPS, GLONASS and Galileo with IGS14.atx
Receiver antenna correction	Correct PCOs and PCVs for GPS and GLONASS with IGS14.atx; Galileo corrections are assumed to be the same as GPS
Phase-windup effect	Model corrected
Station displacement	Corrected based on IERS Convention 2010, including Solid Earth tide, pole tide and ocean tide loading
Differential code bias	Corrected with DCB products provided by CODE
Satellite orbits and clocks	Using the ultra-rapid products from GeoForschungsZentrum Potsdam (GFZ)
Station coordinates	Estimated in epoch-wise kinematic mode
Terrestrial frame	International Terrestrial Reference Frame 2014 (ITRF2014)
Receiver clock offsets	Estimated as epoch-wise white noise
Receiver ISBs	Estimated as constants
GLONASS code IFBs	Estimated as 1-day constants
Tropospheric delays	Saastamoinen model as a priori value and wet delay estimated every 1 hour as random-walk noises (10^{-7} m^2/s) with global mapping function
Phase ambiguities	Estimated as float constants
Estimator	EKF

To assess the a priori stochastic model based on real-time SISRE for multi-GNSS PPP, ultra-rapid and final multi-GNSS orbit and clock products provided by the GFZ were used for the SISRE calculation, which are denoted in the following experiment as GBU and GBM, respectively. The predicted GBU orbit and satellite clock products between 2 and 5 hours into the second half of each GBU file were used for the real-time multi-GNSS PPP. The real-time multi-GNSS SISRE values from January to July 2018 were calculated using equation (5.10). The mean RMS SISRE and the 95th-percentile SISRE of each satellite in different GNSSs for the six months were also calculated, as shown in Figure 5.2. Table 5.3 lists the mean RMS SISRE values over all satellites for each GNSS, which are also denoted in Figure 5.2 with the horizontal bar. As can be seen from Figure 5.2, the satellite-specific SISRE value in the same GNSS varies from each other. The satellite SISRE values of most satellites are nearly at the same level as that of the corresponding GNSS. However, it can also be observed that some satellites' SISRE values are significantly larger than others, such as PRN#G08, PRN#G24 and PRN#E11, which can be excluded from user positioning or applied with lower weight when they are not flagged as outliers. In addition, the mean RMS SISRE values for different GNSSs vary from each other. In the following experiment, two a priori stochastic models with the real-time SISRE for multi-GNSS PPP were evaluated. One was based on the GNSS system-specific SISRE and the other one was based on the satellite-specific SISRE.


Satellite SISRE
The 95th-percentile SISRE
SISRE (m)
GPS
0.3
G01 G02 G03 G05 G06 G07 G08 G09 G10 G11 G12 G13 G14 G15 G16 G17 G18 G19 G20 G21 G22 G23 G24 G25 G26 G27 G28 G29 G30 G31 G32
0 0.5 1 1.5 2


Satellite SISRE
The 95th-percentile SISRE
SISRE (m)
GLONASS
0.9
R01 R02 R03 R04 R05 R06 R07 R08 R09 R10 R11 R13 R14 R15 R16 R17 R18 R19 R20 R21 R22 R23 R24
0 0.5 1 1.5 2 2.5 3

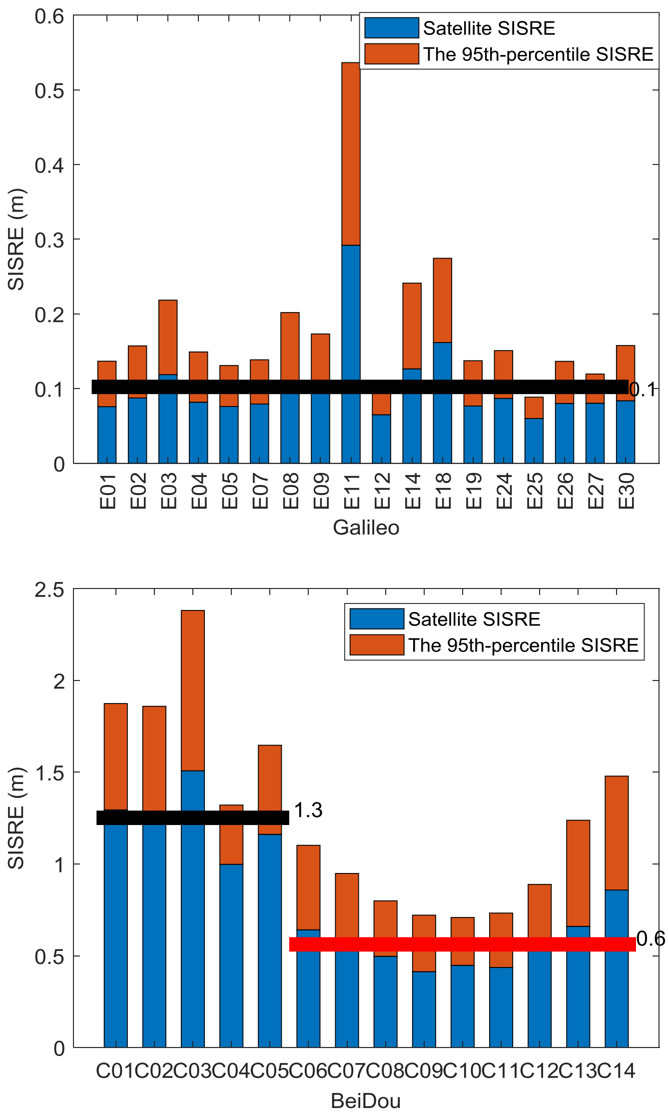


Figure 5.2 Multi-GNSS satellite SISRE comparison

Table 5.3 Multi-GNSS system SISRE comparison

System	RMS of SISRE (m)
GPS	0.3
GLONASS	0.9
Galileo	0.1
BeiDou (GEO)	1.3
BeiDou (IGSO/MEO)	0.6

The SISRE ratios among all satellites and GNSSs were used to scale the elevation-based weight matrices for the two a priori stochastic models. To calculate the SISRE ratios, all satellite- and GNSS-specific SISRE values were compared with the mean RMS SISRE value of GPS. The daily satellite and GNSS SISRE ratios were used for the two a priori stochastic models, which can be updated with the same rate as the precise orbit and satellite clock products. Since multi-GNSS observations during 1-7 March 2018 were selected for multi-GNSS PPP, the GBU orbit and clock products starting from 28 February 2018 were utilised for the satellite and GNSS SISRE calculations. In addition to the two a priori stochastic models based on the SISRE ratios, the other two real-time estimated stochastic models were also considered. The simplified multi-epoch estimator for HVCE method based on equation (5.20) with the standard measurement residuals was used in the experiment. The IGG-III was used to remove the influence of outliers. The number of epochs was set to 20, i.e. 10 mins. To achieve stable variance estimation the least number of observed satellites for each GNSS was set to 5. Only pseudorange measurement residuals were used to estimate the variance in real-time as the carrier phase ambiguities were not fixed in the experiment. The other real-time variance estimation for pseudorange noise and multipath is referred to as the PML method in the following experiment. The number of epochs to initialise and estimate the variance of pseudorange noise and multipath were set to 60 and 20, i.e. 30 mins and 10 mins, respectively. To avoid using a wrong estimate of variance of pseudorange noise and multipath for each observed satellite, a constraint was introduced. If the estimated variance is α times larger/smaller than the one estimated before, then the weight matrix used in the stochastic model is calculated with the one estimated before. α was set to 5 in the experiment. In addition, when cycle slips were detected during multi-GNSS PPP processing, the two real-time estimated stochastic models were re-initialised. To assess the four different stochastic models for real-time multi-GNSS PPP, the performance of the 14 MGEX stations with respect to positioning accuracy, repeatability and estimated ZTD accuracy were compared. The positioning accuracy and repeatability were indicated by the RMS and STD of positioning error, respectively. RMS of ZTD estimation error was used to represent the ZTD accuracy. The estimated station coordinates and ZTD were all compared against the daily SINEX and ZTD solutions provided by the IGS, which can be accessed at ftp://cddis.nasa.gov/gnss/products/.

Figure 5.3 shows the station performance improvement of real-time multi-GNSS PPP using the four different stochastic models, compared with the results of conventional multi-GNSS PPP with a priori stochastic model based on satellite elevation angle.

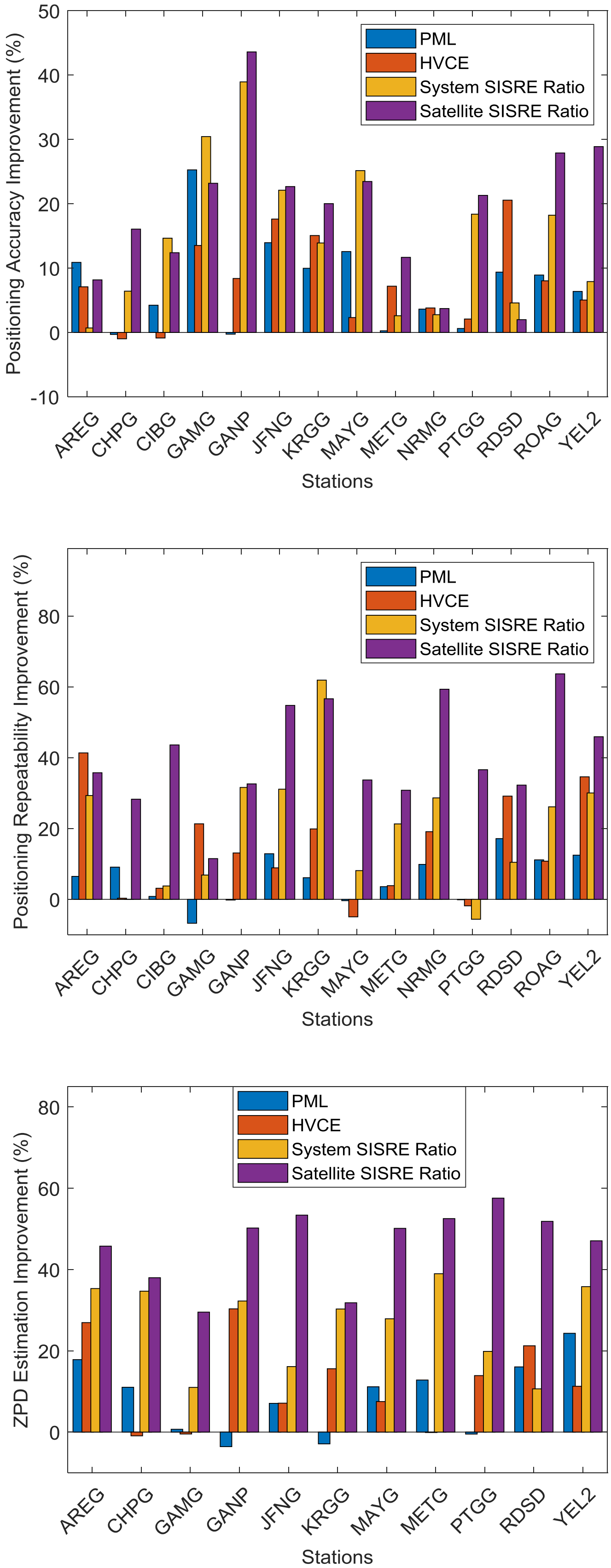


Figure 5.3 Real-time multi-GNSS PPP performance improvement with four different

stochastic models

Table 5.4, Table 5.5 and Table 5.6 list the averaged positioning accuracy, repeatability and estimated ZTD accuracy over all 14 MGEX stations as well as the corresponding percentage of improvement, respectively. It can be seen that the positioning accuracy, repeatability and estimated ZTD accuracy achieved by multi-GNSS PPP with the four stochastic models are all better than using conventional multi-GNSS PPP even though a few stations, such as CHPG, GAMG and MAYG, have slightly worse performance, as shown in Figure 5.3.

Table 5.4 Averaged positioning accuracy comparison among four different stochastic models

Stochastic Models	RMSE (cm)				Accuracy Improvement
	East	North	Up	3D	
Conventional PPP	9.1	2.8	6.6	11.5	-
PML	8.2	2.7	6.3	10.7	7.4%
HVCE	7.9	2.7	5.9	10.2	10.7%
System SISRE Ratio	7.3	2.7	5.3	9.4	18.2%
Satellite SISRE Ratio	6.7	2.9	5.1	8.9	22.7%

Table 5.5 Averaged positioning repeatability comparison among four different stochastic models

Stochastic Models	STD (cm)				Repeatability Improvement
	East	North	Up	3D	
Conventional PPP	2.6	1.4	2.5	3.9	-
PML	2.5	1.2	2.3	3.6	6.9%
HVCE	2.1	1.2	2.1	3.3	14.7%
System SISRE Ratio	2.1	1.1	1.9	3.1	20.7%
Satellite SISRE Ratio	1.3	0.9	1.7	2.3	41.5%

Table 5.6 Averaged ZTD estimation comparison among four different stochastic models

Stochastic Models	RMSE (cm)	ZTD
Conventional PPP	11.2	-
PML	10.3	7.9%
HVCE	9.6	13.4%
System SISRE Ratio	8.3	26.1%
Satellite SISRE Ratio	6.0	46.3%

As can be seen from the performance comparison, the real-time multi-GNSS PPP with stochastic model based on satellite SISRE ratio has the best performance with 22.7%, 41.5% and 46.3% improvement in positioning accuracy, repeatability and ZTD estimation accuracy, respectively. This stochastic model takes into account the errors in precise orbit and satellite clock values of each individual satellite, which can also be used for GNSS PPP with a single GNSS. Although the stochastic model based on PML also scales the weight matrix for each observed satellite, it has the least performance improvement. Since the magnitude of multipath also depends on the elevation angle, it reflects the pseudorange precision similar to the a priori elevation-dependent stochastic model. To further illustrate the difference between stochastic model based on PML compared with the one based on satellite SISRE ratio, the average RMS of pseudorange noise and multipath for each GNSS was calculated, as shown in Figure 5.4 and Table 5.7. It can be seen that observations from GPS and GLONASS have nearly the same level of pseudorange noise and multipath of about 0.6m. The lowest average RMS of pseudorange noise and multipath 0.4m was observed for Galileo observations. The weight ratios calculated with the average RMS of pseudorange noise and multipath among the different GNSSs are listed in Table 5.7. It can be seen as that only observations from Galileo were being scaled compared with those of GPS observations.

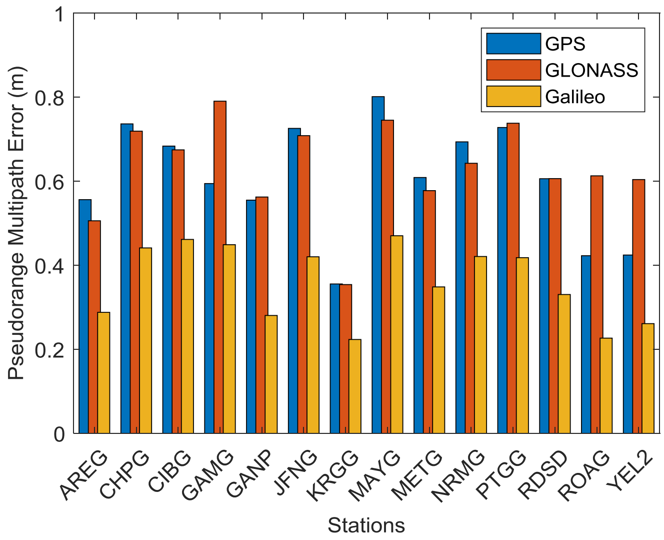


Figure 5.4 Multi-GNSS pseudorange multipath error of different stations

Table 5.7 Multi-GNSS pseudorange multipath error comparison

GNSS Systems	Average Pseudorange Multipath Error (m)	Multipath Ratio
GPS	0.6	1.0
GLONASS	0.6	1.1
Galileo	0.4	0.6

The other two stochastic models only scale the weight matrices at the GNSS level by assuming the observed satellites in the same GNSS have the same weight. The performance of these two stochastic models is worse than that with satellite SISRE ratio, and better than that using the PML method. In addition, it can be seen that the one based on system SISRE ratio has better performance than the one using the HVCE method. Unlike the stochastic model based on system SISRE ratio that indicates the quality of precise satellite products, HCVE uses the measurement residuals containing all unmodelled errors. Therefore, in theory the stochastic model based on HCVE should have better performance. However the measurement residuals calculated in real-time during the positioning adjustment may not be able to reflect all the unmodelled errors due to the unconverged estimated unknown parameters. In addition, the requirement of redundant measurements could not be satisfied all the time, affecting variance estimation for the HVCE method as it only scales the weight matrix when the variances of all the epochs in the sliding window can be estimated. To demonstrate the effectiveness of the HVCE method, the RMS of measurement residuals of GPS and Galileo measurements derived from conventional multi-GNSS PPP and HVCE for the 14 stations were calculated and compared in Figure 5.5. If HVCE could correctly estimate variance at all the epochs, the RMS residuals calculated for different GNSSs should be on the same level, i.e. the ratios among different GNSSs are around 1. The bottom plot in Figure 5.5 gives the residual ratios between GPS and Galileo. It can be seen that 8 stations achieve ratios around 1 with the HVCE method and the other 6 stations have ratios almost at the same level as those using conventional PPP.

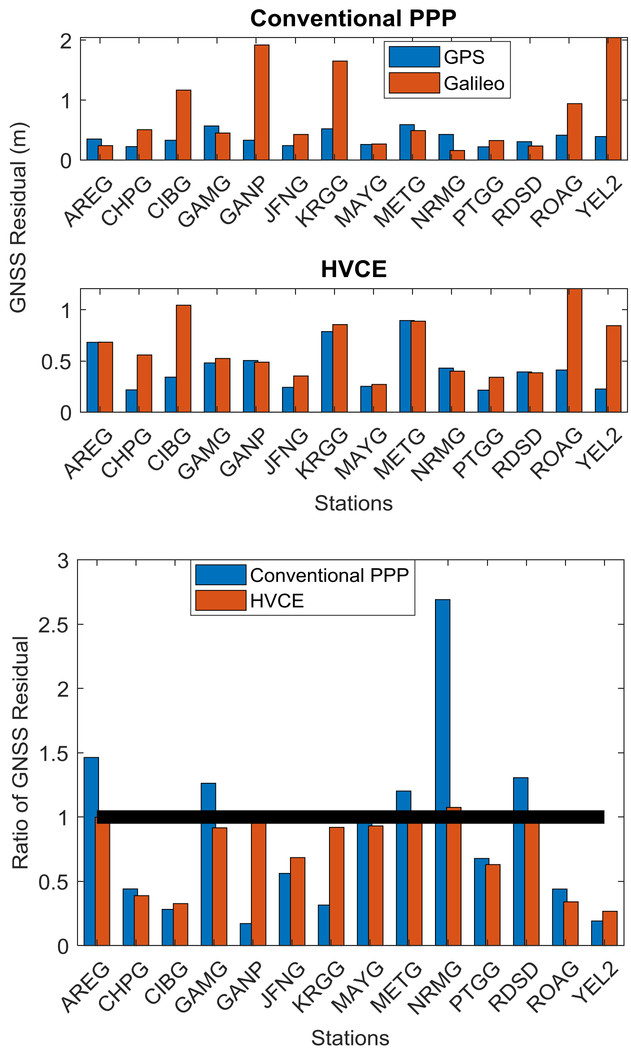


Figure 5.5 Observation residuals between conventional PPP and HVCE comparison

To further compare the performance of the HCVE method and the a priori stochastic model based on satellite SISRE ratio, another stochastic model based on HCVE combined with satellite SISRE ratio was also evaluated. Table 5.8, Table 5.9 and Table 5.10 list the detailed performance comparison over the 14 MGEX stations between these two stochastic models, as well as the percentage of improvement in terms of positioning accuracy, repeatability and estimated ZTD accuracy. It can be seen that the HCVE with satellite SISRE ratio has a slightly better performance than the stochastic model based on satellite SISRE ratio. Therefore, in practical applications it would be promising to provide the satellite SISRE values to the multi-GNSS PPP users in real-time as the HVCE method may not always be effective for real-time use. Furthermore, the a priori stochastic model with satellite SISRE ratio is simpler to implement compared to the real-time estimated stochastic models.

Table 5.8 Averaged positioning accuracy comparison between two different stochastic models

Stochastic Models	RMSE (cm)				Accuracy Improvement
	East	North	Up	3D	
Satellite SISRE Ratio	6.7	2.9	5.1	8.9	22.7%
HVCE with Satellite SISRE Ratio	6.4	2.9	5.1	8.7	24.3%

Table 5.9 Averaged positioning repeatability comparison between two different stochastic models

Stochastic Models	STD (cm)				Repeatability Improvement
	East	North	Up	3D	
Satellite SISRE Ratio	1.3	0.9	1.7	2.3	41.5%
HVCE with Satellite SISRE Ratio	1.2	0.8	1.6	2.2	43.3%

Table 5.10 Averaged ZTD estimation comparison between two different stochastic models

Stochastic Models	RMSE (cm)	ZTD Improvement
Satellite SISRE Ratio	6.0	46.3%
HVCE with Satellite SISRE Ratio	5.8	48.4%

5.5 Summary

Two types of stochastic models for real-time multi-GNSS PPP were assessed by a static experiment, performed in a kinematic mode, with one-week data from 14 MGEX stations. The performance in terms of positioning accuracy, repeatability and estimated ZTD accuracy for all the stochastic models were compared. It was found that the stochastic models based on real-time SISRE, HVCE and PML could all achieve better performance than the one based on satellite elevation angle. The best performance was obtained by the stochastic model with real-time satellite SISRE ratio. Although PML can also obtain the weight ratios at the satellite level, the estimated stochastic model only reflects the observation precision resulting from the measurement noise and multipath. In theory, the HVCE based method should give the best performance if all the unmodelled errors are reflected in the measurement residuals. However, by calculating the RMS residuals of different GNSSs for the 14 MGEX stations, only 8 stations had the same level of residuals among the observed GNSSs. This method estimates the variance with the measurement residuals calculated in real-time during the positioning adjustment but may not be able to reflect all the unmodelled errors. In addition, it requires sufficient observation redundancy for each observed GNSS and in order to improve its estimation robustness and remove the influence of outliers, averaged multi-epoch stochastic model and IGG-III were used, which could lead to limited positioning performance improvement. From the evaluation of HVCE compared to real-time satellite SISRE ratio, it was found that the positioning performance was only slightly improved in comparison with that with that of the stochastic model based on satellite SISRE ratio.

Chapter 6 Satellite Selection with an End-to-end Deep Learning Network

6.1 Introduction

Benefiting from multi-constellation GNSS, the number of satellites in view will be increased to well over 40 most of the time, at many places around the world. This significant increase in the number of available GNSS satellites will greatly improve navigation performance in terms of positioning accuracy, reliability and availability. However, it may not always be possible to track signals and process measurements of all visible satellites in real-time for a standalone GNSS receiver, especially in the case of low-cost receivers with limited tracking channels, or insufficient bandwidth for augmentation message channels, or critical power consumption (Walter et al., 2016). Even if all visible satellites are used for positioning, the positioning accuracy may not necessarily be improved, and real-time performance can deteriorate due to the high computational burden (Blanco-Delgado et al., 2017). Therefore it is necessary to select the best subset of visible satellites specially for the moving A-PLs at high altitude using real-time multi-GNSS PPP.

The most straightforward algorithm for satellite selection is based on a “brute force” approach that aims to minimise either the GDOP or the WGDOP (Zhang and Zhang, 2009). The optimal satellite subset is determined by computing GDOP/WGDOP values with all possible satellite subset combinations and selecting the subset with the minimum value. The GDOP/WGDOP calculation cycle can reach billions of computations for one epoch, with the calculation of a single GDOP or WGDOP value requiring matrix multiplication and inversion operations. This makes the brute force approach difficult to implement for real-time applications. To reduce the number of GDOP/WGDOP calculation cycles, a number of sub-optimal satellite selection methods have been proposed. For example, a quasi-optimal subset of satellites is selected by recursively removing the satellite that has the smallest increase in GDOP from all satellites in view (Liu et al., 2009), or by sequentially adding the least redundant satellite with respect to previously selected satellites (Peng et al., 2014; Roongpiboonsopit and Karimi, 2009), or just by removing the satellites that have GDOP contribution values smaller than a predefined threshold (Li et al., 2012). However, these algorithms are very likely to select a globally sub-optimal set of satellites. To track the optimal subset of satellites over time, a temporal algorithm for satellite subset selection has been proposed by evolving the best subset over time by swapping one or two satellites (Swaszek et al., 2017). This algorithm shows great promise in finding the optimal subset. However, this algorithm still suffers from the computational burden of hundreds, even thousands of calculation cycles. To minimise (or even avoid) GDOP/WGDOP calculations alternative satellite selection approaches, such as maximisation of the volume of the polytope formed by the satellites (Blanco-Delgado and Nunes, 2010; Kong et al., 2014) and satellites' comparability and distribution characteristics (Li et al., 2016; Wei et al., 2012), have been proposed. These methods cannot guarantee optimal satellite selection and need to take geometric satellite distribution into account. Furthermore, instead of directly computing GDOP using matrix multiplication and inversion operations, there are alternative methods. These include closed-form formulas (Doong, 2009; Teng and Wang, 2016) and machine learning (ML) based methods, such as the Genetic Algorithm (GA) (Mosavi, 2011; Zhu, 2018), SVM (Wu et al., 2011), and Neural Network (NN) approaches (Azami and Sanei, 2014; Jwo and Lai, 2007; Mosavi and Sorkhi, 2009; Simon and El-Sherief, 1995; Zarei, 2014). These ML-based methods treat satellite selection as a regression problem of GDOP calculation and focus on improving the performance of GDOP approximation and/or classification. The performance of some of these methods also depends on the training time and size of training data. To reduce the training time of back propagation (BP) NN, there have been many methods proposed in using either different NNs, such as the probabilistic NN and the general regression NN (Jwo and Lai, 2007), or improved BP algorithms, including resilient BP and conjugate gradient algorithms (Azami et al., 2013). Even with the GDOP approximation/classification methods, satellite selection still requires a brute force procedure to identify the subset with the smallest GDOP value from all possible subsets. Moreover, the GDOP classification can only approximately classify the GDOP value of each satellite subset into a predefined range (Azami et al., 2013; Jwo and Lai, 2007).

In this chapter an end-to-end deep learning network to select the optimal subset from the set of all visible satellites is proposed. Instead of treating the satellite selection as a regression problem for GDOP calculation, it can be considered as a segmentation problem of a small point cloud (Grilli et al., 2017), by partitioning all visible satellites into two classes indicating whether the satellite is selected or not. There is no need to design a specific training pattern to describe the input-output relationships for the GDOP or to implement a brute force selection procedure. To design such an end-to-end deep learning network for satellite selection with GDOP/WGDOP criteria, the interactions among all the input satellites need to be captured by the network.

Since different satellite feeding orders would not change the satellite segmentation results the network can be designed to be invariant to all permutations of the input, as inspired by PointNet via using a simple symmetric function, max-pooling (Qi et al., 2016). Furthermore, the max-pooling enables the network to learn the global features of the satellite input. To output the per-satellite segmentation, the local features of each satellite also have to be learned. The PointNet has shown to be effective in per-point segmentation by combining local point and global input features learned from multi-layer perceptron (MLP) on each point and max-pooling across points, respectively. Therefore, the module with local and global information combination in the PointNet is also adopted by the end-to-end deep learning network for satellite selection. To learn the features of the selected subset from all satellites in view, the input channels used for each satellite should be able to characterise its specified features to help satellite segmentation with respect to GDOP/WGDOP criteria. They can be represented by the receiver-to-satellite vector or elevation and azimuth angles. Each visible satellite is processed identically and independently at the beginning by using a few fully connected (FC) layers on each satellite to achieve the local satellite features. Then, new per-satellite features are obtained by concatenating the local satellite feature and global satellite input feature with max-pooling, which are used as input for the subsequent segmentation layers. To learn more complex features for the selected satellites with GDOP/WGDOP criteria, the architecture of stacked voxel feature encoding (VFE) layers used in VoxelNet is also employed (Zhou and Tuzel, 2017). However, to obtain better-trained models for satellite segmentation, the stacked VFE architecture is modified with reduced output sizes to achieve compact satellite internal representations. One problem with this network is that the number in the predicted subset may not always be equal to the required one. Since the output scores of the network represent probabilities of input being the predicted labels, this problem can be solved by selecting the required number of satellites according to their output scores.

This chapter is organised as follows. Following an introduction of the architectures of PointNet and VoxelNet, the end-to-end network for satellite segmentation is described in detail, including the training and validation data generation, architecture design, and training details. Models for GDOP and WGDOP with different numbers of satellites are then tested with GNSS observations from 220 IGS stations. Finally, the summary is presented.

6.2 **PointNet and VoxelNet Networks**

PointNet is a unified architecture for tasks with irregular input format such as point cloud-based 3D classification and segmentation (Qi et al., 2016). It is able to directly take point sets as input and outputs either class labels for the entire input or per-point labels for each point of the input. To realise satellite selection, each satellite has to be identified to be selected or not. Since the segmentation network of PointNet is capable of partitioning input point sets into multiple homogeneous segments, i.e. per-point segment labels for each point of the input, this network is of more interest. Each point in the point cloud can be represented by its $(x, \quad y, \quad z)$ coordinates as well as other feature channels. Assuming that there are n unordered input points $\{x_1, \quad x_2, \quad \cdots, \quad x_n\}$ with $x_i \in \mathbb{R}^d$ and m sub-categories, the model trained from the segmentation network is able to output $n \times m$ scores for each point and sub-category. The basic idea of the PointNet is to approximate a continuous function $f: \boldsymbol{\chi} \rightarrow \mathbb{R}$ that maps the unordered input points to a vector by applying a symmetric function, max-pooling, on transformed elements in the set:

$$f(\{x_1, \quad x_2, \quad \cdots, \quad x_n\}) \approx \gamma\left(\underset{i=1,2,\cdots,n}{\mathrm{MAX}}\{p(x_i)\}\right) \tag{6.1}$$

where γ and p are usually MLP networks. The output vector $\boldsymbol{f}$ represents the global signature of the input set. Each input point is identically and independently processed at the initial stages using the MLP with different hidden layers to encode statistical properties of the points. The MLP can also be seen as several FC layers operating on each point, i.e., 1×1 convolution layers. To achieve per-point segmentation, both local and global features of the input point are extracted to learn the interactions among the points. The local features of each point $\boldsymbol{p}_i \in \mathbb{R}^l$ can be obtained from the initial stages with the MLP, and the global features $\boldsymbol{f} \in \mathbb{R}^g$ can be achieved with the max-pooling algorithm across points. Then the new per-point features $\boldsymbol{p}_i^{new} \in \mathbb{R}^{l+g}$ are extracted by concatenating the local point features with the global features:

$$\boldsymbol{p}_i^{new} = [\boldsymbol{p}_i^T \quad \boldsymbol{f}^T]^T \tag{6.2}$$

Finally, the combined new per-point features are used for point label prediction with the MLP. Figure 6.1 illustrates the basic segmentation network architecture.

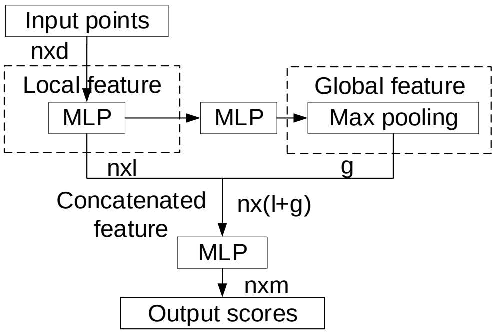


Figure 6.1 PointNet segmentation network architecture

VoxelNet is an end-to-end network for point cloud-based 3D object detection (Zhou and Tuzel, 2017). The voxel used in the VoxelNet refers to a 3D grid, which contains a number of points divided from the point cloud input. One key innovation of the VoxelNet is its architecture of stacked VFE layers to learn voxel-wise feature, as shown in Figure 6.2. As with the feature aggregation network in PointNet, each VFE layer is able to achieve the new point features by combining the point-wise features with one FC layer and the locally aggregated features with the max-pooling, i.e., the local point features and global voxel features. The main difference between the VoxelNet feature learning architecture and PointNet is that there are several global and local feature concatenations performed with the stacked VFE layers. To illustrate the detailed processing procedure of one voxel all the feature sizes involved in the different layers are shown in Figure 6.2. It is assumed that the network also takes n points with input channels d as input and outputs voxel feature $f_v \in \mathbb{R}^v$. Since there is one FC layer and one max-pooling in each VFE layer, the output feature size from each VFE is two times the local feature size. The stacked VFE layers also directly consume the points within the voxel as input and outputs new per-point concatenated features. It encodes point interactions within a voxel and enables learning complex voxel-wise features for characterising 3D shape information of each voxel.

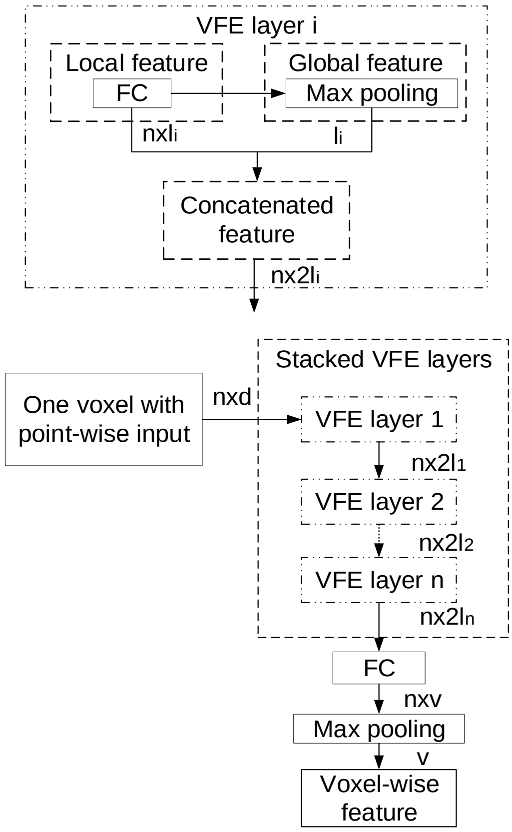


Figure 6.2 Architecture of stacked VFE layers

6.3 An End-to-end Satellite Segmentation Network

In contrast to the GDOP approximation/classification methods mentioned earlier, that treat satellite selection as a regression problem and classifying the GDOP value of satellite input into a predefined range, the proposed end-to-end satellite segmentation network is able to directly segment all the input satellites that are selected or not with no need for the brute force procedure required by GDOP approximation/classification methods. One key problem for satellite segmentation using GDOP/WGDOP criteria is that the segmentation results should be invariant to all the permutations of the input satellites. Inspired by the PointNet segmentation network that uses max-pooling to deal with unordered input, and predicts the per-point class label by combining the global feature with local point features, this architecture can be used for satellite segmentation network by treating the satellites in view at one instant as point cloud input, with two class labels – for selecting the satellite or not selecting the satellite – as output. In addition, the architecture of stacked VFE layers in the VoxelNet is employed to improve the performance of satellite segmentation using GDOP/WGDOP criteria.

6.3.1 **Train and Validation Data Generation**

Assume the selection of *m* satellites from *n* satellites in view from multiple constellations where inter-system time offsets have been accounted for. The criterion used for satellite selection is to minimise either the GDOP or WGDOP value. These values can be calculated by:

$$GDOP = \sqrt{tr(\boldsymbol{H}^T\boldsymbol{H})^{-1}} \quad WGDOP = \sqrt{tr(\boldsymbol{H}^T\boldsymbol{W}\boldsymbol{H})^{-1}} \tag{6.3}$$

where

$tr(\cdot)$ denotes trace of the matrix;

$\boldsymbol{H}$ is the design matrix; and

$\boldsymbol{W}$ is the weight matrix.

$\boldsymbol{H}$ and $\boldsymbol{W}$ can be represented by:

$$\boldsymbol{H} = \begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & 1 \end{bmatrix} \quad \boldsymbol{W} = \begin{bmatrix} w_1 & \cdots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & w_n \end{bmatrix} \tag{6.4}$$

where

$h_i = [x_i \quad y_i \quad z_i]$ is the receiver to *i*th satellite unit vector, the column of ones in the $\boldsymbol{H}$ matrix represents parameters for the receiver clock bias in units of metres; and

w_i is the weight of the *i*th satellite.

Using the brute force approach, the subset with the minimum GDOP or WGDOP value can be selected from $n!/[m!\,(n-m)!]$ possible combinations. Therefore there are two classes in the output data. One class is for the *m* selected satellites and the other is for the

$n - m$ not-selected satellites. To classify the n satellites in an end-to-end fashion, the input data of each satellite should be able to characterise its local feature. The receiver-to-satellite unit vector $[x_i \quad y_i \quad z_i]$ can be used as the input channel, which contains direct information for the optimal subset selection using GDOP/WGDOP criteria. Since it can also be converted to elevation and azimuth $[el_i \quad az_i]$, the input channel represented by $[el_i \quad az_i]$ is another option. In the following experiment, the satellite segmentation performance using both forms of input channels is compared.

6.3.2 Satellite Segmentation Network Architecture

To segment all the satellites in view, both local and global features of each satellite are needed. The local feature of each satellite can be obtained by applying inputs and feature transformation through a few FC layers on each satellite. Since the satellite order does not change the final segmentation result, max-pooling is used for the satellite segmentation network. Furthermore, max-pooling also learns the global features of the satellite input. Then new satellite-wise features can be extracted by combining the global features with the local satellite features. To learn multi-level features for characterising the pattern of the selected satellites, stacked feature encoding (FE) layers based on the architecture of stacked VFE layers are used. However, instead of concatenating local and global features every time the local feature is obtained with the FC layer in each VFE layer, the MLP with a few FC layers are used. As with the MLP in the PointNet, the FC layers operate on each satellite. The global feature obtained from the last layer in the MLP is concatenated with the local feature in the same layer or preceding one as shown in Figure 6.3. All the feature sizes derived from different layer are also illustrated in Figure 6.3. This architecture is able to reduce the inference time due to the two times increase in feature size resulting from feature concatenation of each FC layer as in the stacked VFE layers.

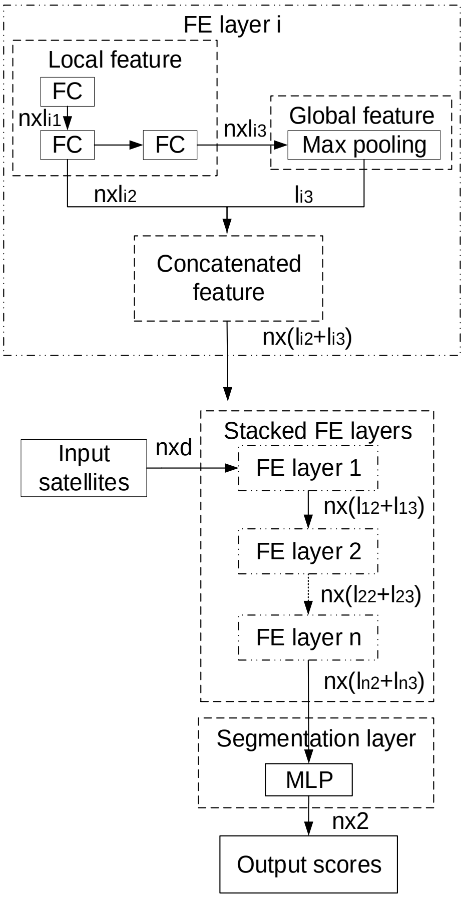


Figure 6.3 Satellite segmentation network architecture

6.3.3 Training Details

Since the number of satellites in view will change with time, zero-padding is needed. The size of the zero-padding can be determined by fixing the number of observed satellites to the maximum possible number of visible satellites, e.g. n. Therefore, the size of each input training data is $n \times d$ where d is the dimension of the satellite input channel. All training data are randomised before processing. The network is designed by trading off the computational complexity and performance of satellite segmentation. It is composed of seven stacked FE layers and two FC layers for the final segmentation. In each FE layer, there are three FC layers and a max-pooling procedure. The max-pooling procedure, performed across the local features learned from the last FC layer, is used to obtain the global feature. The global feature is then concatenated with the local feature obtained from the second FC layer, as shown in Figure 6.3. Overall, there are 23 layers in the proposed network architecture. The detailed layer output sizes are listed in Table 6.1. The number of layers and their sizes are empirical values.

Table 6.1 Satellite segmentation architecture with stacked FE layers

Layers		Output size $(l_{i1}, \ l_{i2}, \ l_{i3})$	Concatenated feature size $(l_{i2} + l_{i3})$	Number of FC layers
Stacked FE layers	FE layer 1	(32, 32, 48)	80	3
	FE layer 2	(64, 64, 96)	160	3
	FE layer 3	(128, 128, 160)	298	3
	FE layer 4	(256, 256, 320)	576	3
	FE layer 5	(512, 512, 640)	1152	3
	FE layer 6	(864, 864, 960)	1824	3
	FE layer 7	(1024, 1024, 1152)	2176	3
Segmentation layer		(1024,2)	—	2

The gradual increase in the channel size makes it possible for the network to learn more high-level features. Due to the redundancy in the concatenated feature in each FE layer, the channel size of the FC layer in the next FE layer is reduced in order to keep the compact satellite features. All FC layers use rectified linear units (ReLU) as activation function and batch normalisation. Since the satellite segmentation is satellite-wise classification, all of the input data including both satellite-input and zero-padding input have to be segmented. The same label as satellite not selected is assigned to zero-padding input. The loss function for the satellite segmentation network is based on the binary cross entropy between the target and predicted classes (De Boer et al., 2005):

$$L = -\alpha \frac{1}{N_1} \sum_{i=1}^{N_1} log\left(\hat{l}^i\right) - \beta \frac{1}{N_0} \sum_{i=1}^{N_0} log\left(1 - \hat{l}^i\right) - \gamma \frac{1}{N_e} \sum_{i=1}^{N_e} log\left(1 - \hat{l}^i\right) \quad (6.5)$$

where

N_1, N_0, and N_e denote the number of satellite-input training data with label 1, 0 and zero-padding input data with label 0, respectively;

$\hat{l}^i$ denotes the predicted value of the softmax output; and

α, β, and γ are loss weight constants to adjust the relative importance balance among different input data.

The minimisation of this loss function also indirectly minimises the difference in the GDOP or WGDOP values between the predicted satellites to be selected and the targeted ones. This loss function is optimised based on one commonly used stochastic gradient descent algorithm for deep learning networks, the Adam optimisation algorithm (Kingma and Ba, 2014). This algorithm uses first-order gradients to optimise stochastic objective functions with adaptive estimates of first and second moments. In addition, the algorithm has very little memory requirement and is very suitable for neural networks with large training datasets and parameters. All the computations in the proposed network can be implemented on a CPU/GPU in parallel. One problem when using this network is that the number of output predictions for selected satellites may not be equal to the predefined one. As the output scores of the network can be interpreted as probabilities of input being the predicted labels, the number of output satellites for selection can be fixed by selecting the satellites with high output scores for label 1.

6.4 **Experiment and Analysis**

To train a satellite selection model with GDOP and WGDOP criteria for the A-PL system, training data were generated based on the brute force approach using one day's GNSS observations (at 1-minute intervals) from 200 IGS stations, resulting in around 288,000 training data samples in total including around 10% data set for test. Both GPS and GLONASS observations were used and it was assumed that the inter-frequency bias for GLONASS satellites had been accounted for. Elevation cutoff angle for each receiver was set to 15 degrees. Validation data were generated in a similar way for another 20 IGS stations. The maximum number of satellites in view was 20, i.e. $n = 20$. The labels for selected satellites were set as class 1 and the not-selected satellites were designated class 0. Due to the slightly unbalanced training data, different loss weight values $\alpha = 1, \beta = 1.2$, and $\gamma = 0.05$ were assigned. Adam optimiser with initial learning rate 0.1, momentum 0.9 and batch size 128 values were used. The decay rate for batch normalisation was 0.7, starting from 0.5 and gradually increasing to 0.99. The learning rate was reduced at the same rate as the decay rate for batch normalisation. The models were converged in around 50 minutes using TensorFlow (Abadi et al., 2016), an open-source ML library running on a GTX1080TI GPU. The code and trained models used in the following experiment can be found at https://github.com/PanUnsw/satellite_selection.git.

6.4.1 Satellite Segmentation with Different Input Channels and Architectures

To compare the segmentation performance with input channels represented by $c_i = [x_i \quad y_i \quad z_i]$ and $c_i = [el_i \quad az_i]$, one example for selecting 9 satellites based on the GDOP criterion is shown in Figure 6.4. Table 6.2 is a comparison of training and testing accuracies. The accuracy represents the percentage of satellite input segmented correctly, i.e., the predicted subset is the same as the optimal subset. Although $[x_i \quad y_i \quad z_i]$ and $[el_i \quad az_i]$ are mathematically equivalent, the input channel with $[x_i \quad y_i \quad z_i]$ has slower training convergence, and worse converged training and testing accuracies than that based on $[el_i \quad az_i]$. This fact can be intuitively explained by the saddle points when optimising the loss function, which can considerably slow down training (Dauphin et al., 2014). It may take more time to escape the saddle points for the input channel with $[x_i \quad y_i \quad z_i]$ compared with $[el_i \quad az_i]$ at the beginning of training. The input channels represented by $[el_i \quad az_i]$ were used in the following experiment.

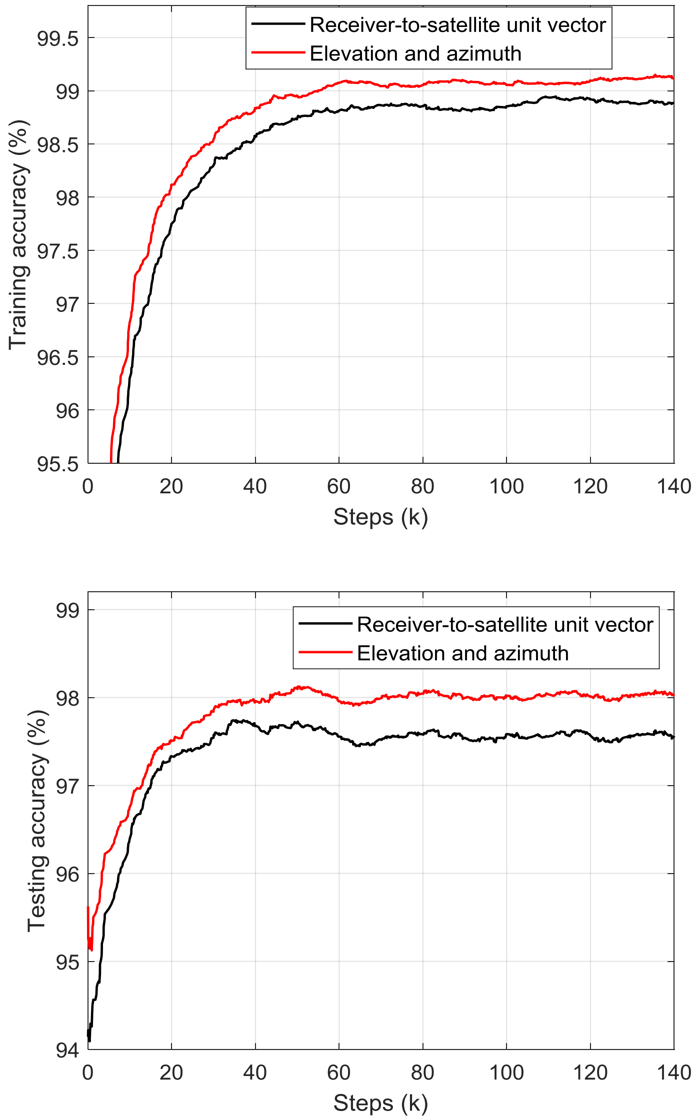


Figure 6.4 Performance comparison with different input channels

Table 6.2 Accuracy comparison with different input channels

Input channels	Training accuracy	Testing accuracy
Receiver-to-satellite unit vector	98.9%	97.6%
Elevation and azimuth	99.1%	98.0%

To illustrate the performance of the satellite segmentation architecture with stacked FE layers compared with that without stacked FE layers, another model with an architecture without stacked FE layers was also trained with elevation and azimuth input channels for selecting 9 satellites based on the GDOP criterion. The architecture, consisting of six FC layers, one max-pooling layer and one segmentation layer was used by trading off the computational complexity and performance of satellite segmentation. The output sizes of these layers are shown in Figure 6.3. Local features obtained from the third FC layer were concatenated with the global features from the max-pooling layer. The number of layers and their sizes used for the architecture without stacked FE layers are also empirical values.

Table 6.3 Satellite segmentation architecture without stacked FE layers

Layers	Output size
FC layers	(32, 64, 128, 256, 512, 1024)
Max pooling	1024
Segmentation layer	(512,2)

The initial parameters used for the architecture without stacked FE layers were the same as with stacked FE layers except that the decay rates for batch normalisation and learning rate were set to 0.3 to reduce the tendency for overfitting. Figure 6.5 and Table 6.4 show the performance comparison between the two trained models, model 1 and 2, with and without stacked FE architecture, respectively. It can be seen that both the training and test accuracies of model 2 are worse than that of model 1. Therefore, it is preferable to use the architecture with stacked FE layers for satellite segmentation.

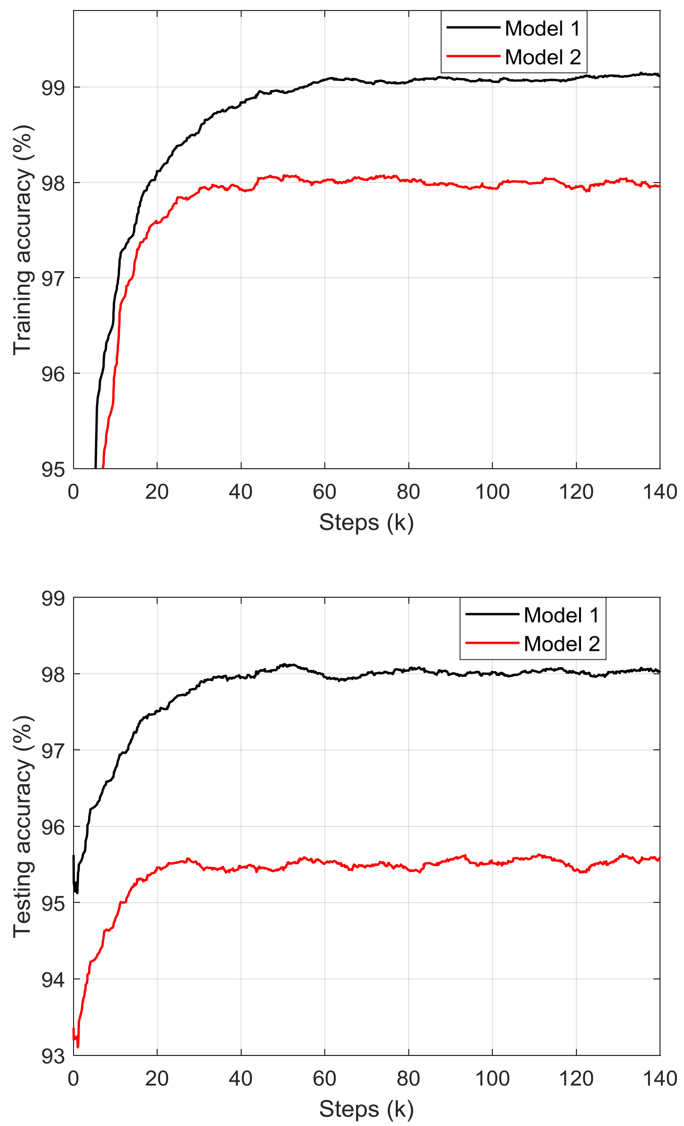


Figure 6.5 Performance comparison with different architectures

Table 6.4 Accuracy comparison with different architectures

Network architecture	Training accuracy	Testing accuracy
Model 1	99.1%	98.0%
Model 2	97.9%	95.6%

6.4.2 **Satellite Segmentation with GDOP**

Models for selecting 9 and 12 satellites from the same set of observed satellites using the GDOP criterion were trained and evaluated. With the same network architecture with stacked FE layers, the models trained were used when selecting, in turn, 9 or 12 satellites.

Table 6.5 shows the segmentation performance comparison. It can be seen that the testing accuracy is slightly worse than the training accuracy. The model for selecting 12 satellites has better performance than that for 9 satellites.

Table 6.5 Satellite segmentation performance using the GDOP criterion

Number of selected satellites	Training accuracy	Testing accuracy
9	99.1%	98.0%
12	99.2%	98.3%

To validate the trained models, Table 6.6 lists the validation results. Compared with means of the best GDOP values, the means of the validation GDOP values were only increased by 0.08% and 0.04% when selecting 9 or 12 satellites, respectively. For more than 80% of the epochs the optimal subset of the satellite input was correctly predicted, and around 20% when one satellite in the optimal subset was not selected. The one satellite in the optimal subset that was not selected was often replaced by one that produces a slightly worse GDOP value than the minimum GDOP value.

Table 6.6 Validation evaluation performance with GDOP-trained model

Number of selected satellites	Mean of best GDOP	Mean of validation GDOP	Percentage of wrong segmentation	
			0	1
9	1.981	1.983	81%	18%
12	1.809	1.810	83%	16%

One example of the predicted subset with one wrongly selected satellite is shown in Figure 6.6. The detailed input and output labels for the targets and predictions are listed in Table 6.7. The satellite PRN G9 in the optimal subset was predicted with label 0, which was replaced by satellite PRN R18 resulting in the optimal GDOP value increasing from 1.617 to 1.619.

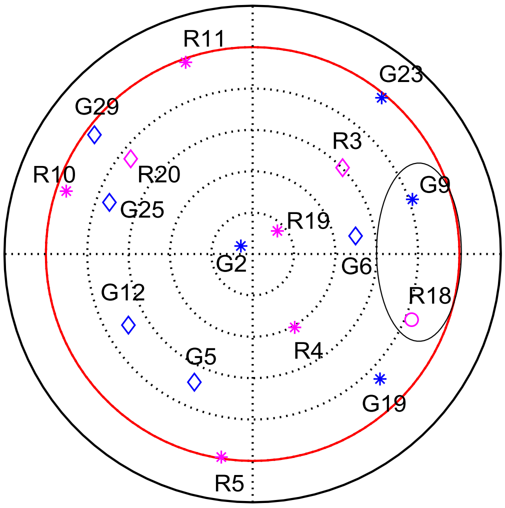


Figure 6.6 One example of a wrongly predicted subset with GDOP-trained model

Asterisk, diamond and circle markers identify the satellites optimally selected, not selected and wrongly selected, respectively.

Table 6.7 One example of satellite segmentation with GDOP-trained model

Satellite	Input		Output labels	
	Elevation (rad)	Azimuth (rad)	Targets	Predictions
G2	1.480	5.310	1	1
G5	0.680	3.569	0	0
G6	0.909	1.397	0	0
G9	0.502	1.241	1	0
G12	0.664	4.193	0	0
G19	0.442	2.346	1	1
G23	0.288	0.690	1	1
G25	0.607	5.058	0	0
G29	0.317	5.358	0	0
R3	0.782	0.806	0	0
R4	1.035	2.621	1	1
R5	0.270	3.295	1	1
R10	0.325	5.037	1	1
R11	0.285	5.946	1	1
R18	0.482	1.962	0	1
R19	1.356	0.820	1	1
R20	0.591	5.375	0	0

Figure 6.7 is a plot of the percentage of GDOP increase between the predicted and the optimal subsets. It can be seen that the GDOP increase is limited to 8% and 3% when selecting 9 or 12 satellites, respectively, as shown in the top and bottom panels.

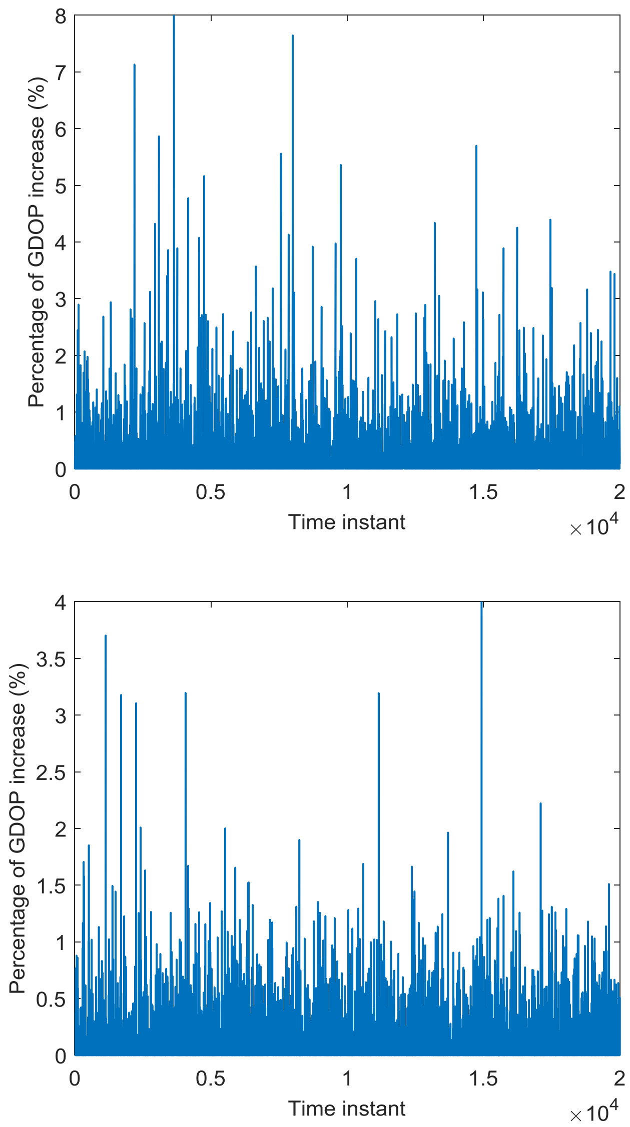


Figure 6.7 Percentage of GDOP increase

The histogram of GDOP value differences between the validation and the optimal GDOP values is plotted in Figure 6.8. For about 99% and 100% of the time, the GDOP difference is smaller than 0.03 when selecting 9 or 12 satellites, respectively.

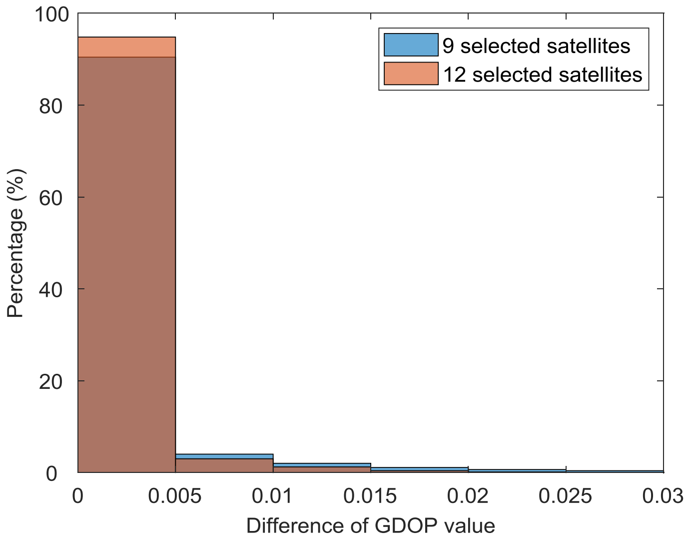


Figure 6.8 GDOP value difference comparison

6.4.3 **Satellite Segmentation with WGDOP**

In this section, satellite segmentation results are presented for two trained models using the WGDOP criterion. The weight calculated for each satellite is based on its elevation angle as $w = [sin(el)]^2$. With the same network architecture used above, the best performance results of two trained models when selecting 9 or 12 satellites are listed in Table 6.8. Similar to the earlier trained models using the GDOP criterion, the model for selecting 12 satellites has better performance than that for 9 satellite selection. However, both models trained with WGDOP have lower accuracies than the ones with GDOP. This can be attributed to more complex features introduced by the different target satellites.

Table 6.8 Satellite segmentation performance with the WGDOP criterion

Number of selected satellites	Training accuracy	Testing accuracy
9	99.1%	95.8%
12	99.2%	96.8%

Even with the larger bias between the testing and training accuracies, around 95% of the time the predicted subsets are correctly selected, or with just one wrong segmentation, compared with the optimal ones, as listed in Table 6.9. The means of the validation WGDOP values are increased by 0.31% and 0.15% when selecting 9 or 12 satellites, respectively. Similar to the earlier case, the subset with one wrongly selected satellite usually has a slightly larger WGDOP value than the optimal subset. One typical example of a wrongly predicted subset is shown in Figure 6.9. Table 6.10 lists the elevation and azimuth input and the output labels for the targets and predictions. The difference in WGDOP value between the optimal and predicted subsets is approximate 0.012, with WGDOP increased from 3.418 to 3.430.

Table 6.9 Validation evaluation performance with WGDOP-trained model

Number of selected satellites	Mean of best WGDOP	Mean of validation WGDOP	Percentage of wrong segmentation	
			0	1
9	3.742	3.753	65%	29%
12	3.355	3.359	71%	27%

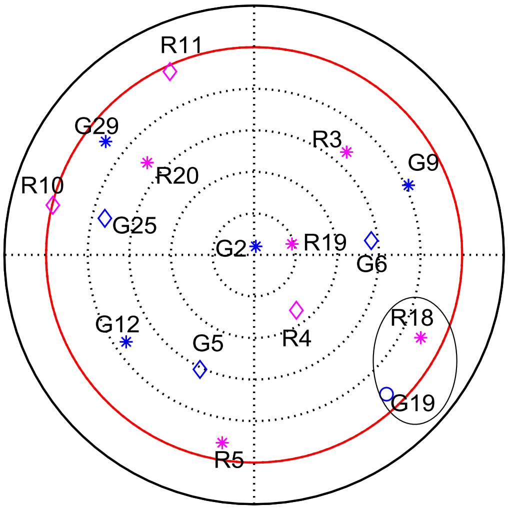


Figure 6.9 One example of a wrongly predicted subset with WGDOP-trained model

Asterisk, diamond and circle markers identify the satellites optimally selected, not selected and wrongly selected, respectively.

Table 6.10 One example of satellite segmentation with WGDOP-trained model

Satellite	Input		Output labels	
	Elevation (rad)	Azimuth (rad)	Targets	Predictions
G2	1.516	0.197	1	1
G5	0.773	3.582	0	0
G6	0.828	1.449	0	0
G9	0.505	1.146	1	1

G12	0.597	4.114	1	1
G19	0.361	2.382	0	1
G25	0.603	4.953	0	0
G29	0.395	5.365	1	1
R3	0.699	0.731	1	1
R4	1.132	2.491	0	0
R5	0.370	3.309	1	1
R10	0.266	4.955	0	0
R11	0.300	5.853	0	0
R18	0.400	2.032	1	0
R19	1.322	1.289	1	1
R20	0.682	5.425	1	1

The percentage of WGDOP increase between the predicted and the optimal subsets is mostly limited to 8% and 4% when selecting 9 or 12 satellites, respectively, as shown in the top and bottom panels of Figure 6.10.

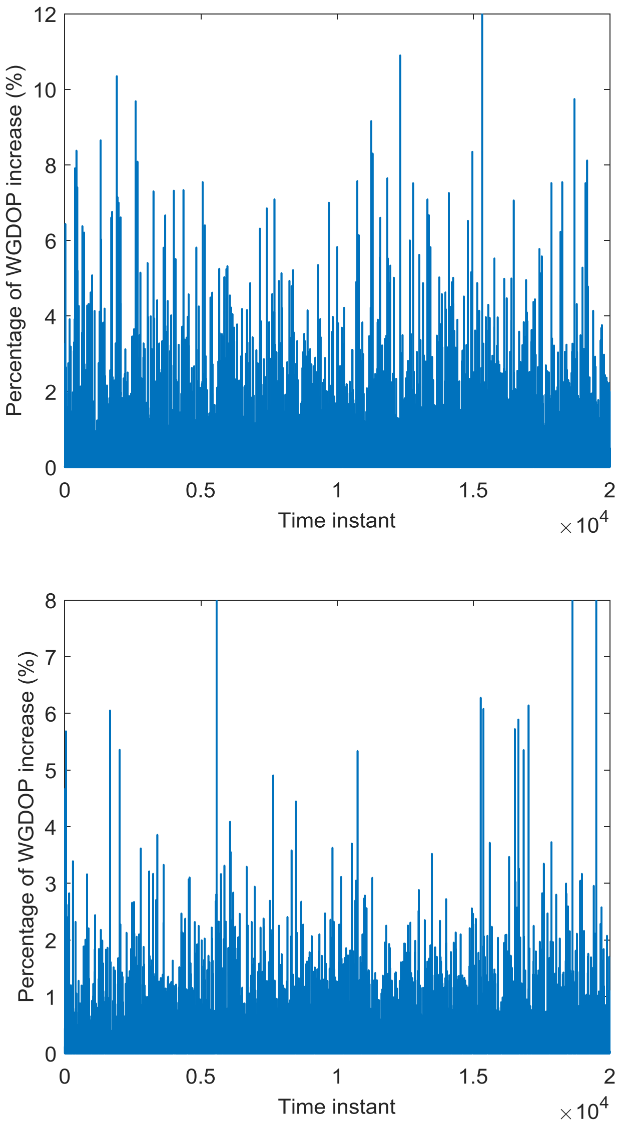


Figure 6.10 Percentage of WGDOP increase

Figure 6.11 shows the histogram of WGDOP value differences between the validation and the optimal WGDOP values. More than 99% of the differences in WGDOP value are less than 0.2, for both cases of selecting 9 or 12 satellites. This is adequate for most applications.

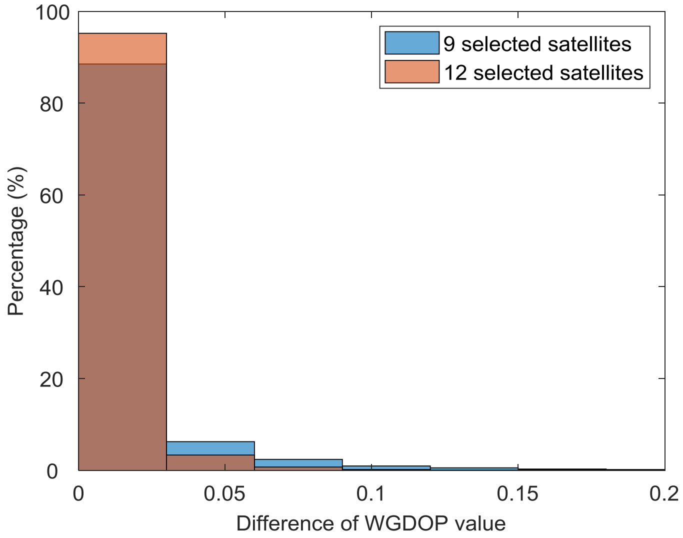


Figure 6.11 WGDOP value comparison

The comparison of average computational time with GDOP and WGDOP criteria between the brute force approach to select one optimal subset and the proposed satellite segmentation network to predict one selected subset with the trained models is summarised in Table 6.11. It can be seen that the satellite segmentation network method is about 90 times faster than the brute force approach. Even though the maximum number of input satellites for the above trained models was set to 20, the same satellite segmentation architecture can be used to train those with different numbers of input satellites from any GNSSs.

Table 6.11 Performance comparison of computation time

Methods	Average computation time (milliseconds)	
	GDOP	WGDOP
Brute force approach	110	115
Satellite segmentation network	1.2	1.3

6.5 Summary

In this chapter, an end-to-end deep learning network for satellite selection invariant to satellite input permutation based on the PointNet and VoxelNet networks was presented. The satellite selection procedure was converted to a satellite segmentation procedure, with specified input channel for each satellite and two class labels representing the selected and not-selected satellites. The proposed satellite segmentation network was composed of several simple stacked FE layers and one segmentation layer. An experiment was conducted to evaluate the proposed approach with training and validation data from 220 IGS stations. The satellite segmentation performance was compared with respect to different input channels, including receiver-to-satellite unit vector and elevation and azimuth, as well as different architectures, i.e. with and without stacked FE layers. The experiment showed that it was preferable to use an architecture with stacked FE layers and input channel represented by elevation and azimuth due to the faster training convergence and better converged accuracy. Cases for selecting 9 or 12 satellites, with GDOP and WGDOP criteria, were investigated. It was demonstrated that using the satellite segmentation network approach was around 90 times faster than the brute force satellite selection approach. From the validation results, it can be concluded that the trained models were capable of selecting the satellites that had the most contribution to the GDOP or WGDOP value and had no limitation of application locality as shown in GDOP approximation/classification methods. In addition, the trained models based on GDOP had better performance than the ones based on WGDOP. Furthermore, the models for selecting 12 satellites were more accurate than those for 9 satellites.

Chapter 7 Conclusions and Recommendations

7.1 Concluding Remarks

This research has investigated an A-PL system with A-PL positioned based on the real-time GNSS PPP technique. Kinematic GNSS PPP may suffer from reduced positioning accuracy due to the mobile A-PLs' susceptibility to loss of GNSS signal or disruptions of correction message communication links. To enhance positioning performance of GNSS PPP in terms of convergence time, positioning accuracy and stability, there have been many methods proposed. One popular approach is based on augmentation with additional information, such as adding more observations of multiple frequency from multiple GNSS constellations, integrating with a new navigation system and introducing atmospheric constraints. In our proposed A-PL system it is assumed that it is possible to obtain inter-PL range measurements that can be combined with GNSS measurements. The contribution of these additional inter-PL ranges for A-PL positioning was studied. Different short-term prediction models for orbit and clock error corrections were also discussed so as to achieve continuous precise A-PL positioning. To further improve the robustness of GNSS PPP positioning, the unmodelled errors in the GNSS measurement model, resulting from multipath, interference and atmospheric residuals, are accounted for with non-parametric model-learning algorithms. Two commonly used non-parametric algorithms, LS-SVM and GPR, were investigated. In addition, to realise the optimal integration and stable positioning performance for multi-GNSS PPP, a stochastic model to appropriately weight observations from different GNSS systems is required in the positioning adjustment. Two types of stochastic models, including the a priori stochastic models based on satellite elevation angle and real-time SISRE values, and real-time estimated variance methods based on HVCE and PML, were investigated with a comprehensive performance comparison with respect to positioning accuracy, repeatability and estimated ZTD accuracy. Finally, an end-to-end deep learning network for satellite selection is proposed to trade off positioning performance and time consumption when using real-time multi-GNSS positioning.

The main findings of this research are:

(1) To implement real-time GNSS PPP combined with inter-PL ranges for A-PL positioning, three forms of SCIF algorithms were described and investigated. It was found

that the A-PL using GNSS PPP combined with inter-PL range measurements is able to achieve better positioning performance in terms of speed of convergence and positioning accuracy than that using the GNSS PPP-only approach. However, the performance improvement is limited by the transmitted trajectory data of the observed A-PLs, which have to be provided with well-converged accuracy. The SCIF-based distributed algorithms appear to have higher robustness in dealing with degraded transmitted trajectory data of the observed A-PLs compared with the centralised algorithm. In addition, to maintain real-time GNSS PPP accuracy when the correction message communication links are disrupted, a second-order polynomial model is preferable for short-term orbit correction predictions compared with the first- or third-order models. The satellite clock corrections can be predicted using either the linear model or one with linear and sinusoidal terms.

(2) GNSS PPP with real-time non-parametric model-learning for unmodelled measurement errors based on LS-SVM and GPR was evaluated using both static and kinematic experiments. The results reveal that both the LS-SVM and GPR algorithms can effectively reduce the influence of the unmodelled error and achieve better positioning performance compared with conventional GNSS PPP. In addition, the GPR algorithm is more promising for real-time GNSS PPP applications than LS-SVM due to its slightly better positioning performance and the non-requirement of separate prediction uncertainty calculation.

(3) By comparing the positioning performance for multi-GNSS PPP with the a priori stochastic models and real-time estimated variance methods, it was found that the stochastic models based on real-time SISRE, HVCE and PML could all achieve better performance than the one based on satellite elevation angle, in terms of positioning accuracy, repeatability and estimated ZTD accuracy. The best performance was obtained by the stochastic model with real-time satellite SISRE ratio. Although PML can also obtain the weight ratios on the level of the satellite, the estimated stochastic model reflects the observation precision resulting from the measurement noise and multipath, which are mainly elevation dependent. It was found that the HVCE method could not always estimate an appropriate stochastic model for multi-GNSS PPP with measurement residuals calculated in real-time. The requirements of sufficient observation redundancy and application of an outlier removing technique also limited its positioning performance

improvement. The a priori stochastic model with real-time satellite SISRE ratio is the most promising method for multi-GNSS PPP, as further indicated by the performance comparison with HVCE with a priori stochastic model based on real-time satellite SISRE ratio, as well as considering its simplicity in implementation.

(4) The real-time satellite selection procedure was converted to a satellite segmentation procedure, with specified input channel for each satellite and two class labels representing the selected and not-selected satellites. The performance comparison with respect to different input channels, including receiver-to-satellite unit vector and elevation and azimuth, as well as different architectures was performed. The results indicate that it is preferable to use an architecture with stacked FE layers and an input channel represented by elevation and azimuth due to the faster training convergence and better converged accuracy. Cases for selecting 9 or 12 satellites, with GDOP and WGDOP criteria, were investigated. It was demonstrated that the satellite segmentation network approach is around 90 times faster than the brute force satellite selection approach. The trained models were capable of selecting the satellites that make the most contribution to the GDOP or WGDOP value and have no limitation of application locality as shown in GDOP approximation/classification methods. Furthermore, it was found that the trained models based on GDOP outperform the ones based on WGDOP. In addition, the models for selecting 12 satellites are more accurate than those for selecting 9 satellites.

7.2 **Recommendations for Future Work**

(1) In this research, when performing the A-PL positioning, all the A-PLs are assumed to be perfectly time-synchronised during the entire mission. It would be of interest to also analyse the impact of observed A-PL time synchronisation accuracy on the “to-be-positioned” A-PL positioning performance. In addition, to maintain A-PL positioning accuracy using GNSS PPP when disruption of correction message communication links occurs, different short-term prediction models for satellite clock and orbit correction have been investigated. Further research could be undertaken into the effective periods of these prediction models.

(2) To account for the unmodelled GNSS measurement errors the nonlinear autoregressive model is used to train the LS-SVM/GPR model. The number of lags for the autoregressive model is set to an empirical value. In addition, the parameters involved

in both the LS-SVM and GPR algorithms are also selected empirically. It would be desirable to analyse the influence of different values on GNSS positioning performance in order to determine the most suitable values when undertaking LS-SVM/GPR model-learning in practical applications.

(3) To assess the a priori stochastic model based on real-time SISRE in the research the predicted GBU orbit and satellite clock provided by the GFZ were used to calculate the real-time SISRE. In addition, all the stochastic model estimation methods were assessed for multi-GNSS PPP using only a static experiment. Assessment of these stochastic models for multi-GNSS PPP using real-time precise satellite orbit and clock products in kinematic mode should be undertaken.

(4) The trained models based on the end-to-end deep learning network with GDOP and WGDOP criteria are only applicable for selecting a single fixed number of satellites. In addition, the training and validation data are all from static receivers. In the future, a model capable of selecting a multiple number of satellites can be trained with the selected number of satellites added to the input channel. Kinematic data could be analysed, and the robustness of the trained model further investigated.

References

Abadi, M., et al. (2016) Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.

Afifi, A., and A. El-Rabbany (2015a) An improved between-satellite single-difference Precise Point Positioning model for combined GPS/Galileo observations. Journal of Applied Geodesy, 9(2), 101-111.

Afifi, A., and A. El-Rabbany (2015b) An improved model for single-frequency GPS/GALILEO Precise Point Positioning. Positioning, 6(2), 7-21.

Alam, N., A. T. Balaei and A. G. Dempster (2011) A DSRC doppler-based cooperative positioning enhancement for vehicular networks with GPS availability. IEEE Transactions on Vehicular Technology, 60(9), 4462-4470.

Amiri-Simkooei, A. R. (2007) Least-Squares Variance Component Estimation: Theory and GPS Applications. Ph.D thesis, Delft University of Technology, Delft, Netherlands.

Amiri-Simkooei, A. R., F. Zangeneh-Nejad and J. Asgari (2013) Least-squares variance component estimation applied to GPS geometry-based observation model. Journal of Surveying Engineering, 139(4), 176-187.

Amt, J. H. R., and J. F. Raquet (2007) Flight testing of a pseudolite navigation system on a UAV. In Proceedings of the 2007 National Technical Meeting of The Institute of Navigation, 22-24 January, San Diego, California, USA, 1147-1154.

Aquino, M., J. F. G. Monico, A. H. Dodson, H. Marques, G. De Franceschi, L. Alfonsi, V. Romano and M. Andreotti (2009) Improving the GNSS positioning stochastic model in the presence of ionospheric scintillation. Journal of Geodesy, 83(10), 953-966.

Azami, H., M. R. Mosavi and S. Sanei (2013) Classification of GPS satellites using improved back propagation training algorithms. Wireless Personal Communications, 71(2), 789-803.

Azami, H., and S. Sanei (2014) GPS GDOP classification via improved neural network trainings and principal component analysis. International Journal of Electronics, 101(9), 1300-1313.

Bähr, H., Z. Altamimi and B. Heck (2007) Variance Component Estimation for Combination of Terrestrial Reference Frames. Ph.D thesis, Karlsruhe Institute of Technology, Karlsruhe, Baden-Württemberg, Germany.

Bailey, T., M. Bryson, H. Mu, J. Vial, L. McCalman and H. Durrant-Whyte (2011) Decentralised cooperative localisation for heterogeneous teams of mobile robots. In 2011 IEEE International Conference on Robotics and Automation, 9-13 May, Shanghai, China, 2859-2865.

Banville, S., P. Collins, W. Zhang and R. B. Langley (2014) Global and regional ionospheric corrections for faster PPP convergence. Navigation: Journal of the Institute of Navigation, 61(2), 115-124.

Bisnath, S., and Y. Gao (2009) Current state of Precise Point Positioning and future prospects and limitations. In Observing Our Changing Earth, International Association of Geodesy Symposia, Springer, Berlin, Heidelberg, 133, 615-623.

Bisnath, S., and P. Collins (2012) Recent developments in Precise Point Positioning. Geomatica, 66(2), 103-111.

Bisnath, S., and R. B. Langley (2001) Pseudorange multipath mitigation by means of multipath monitoring and de-weighting. In Proceedings of the International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation, 5-8 June, Banff, Alberta, Canada.

Blanco-Delgado, N., F. D. Nunes and G. Seco-Granados (2017) On the relation between GDOP and the volume described by the user-to-satellite unit vectors for GNSS positioning. GPS Solutions, 21(3), 1139-1147.

Blanco-Delgado, N., and F. D. Nunes (2010) Satellite selection method for multi-constellation GNSS using convex geometry. IEEE Transactions on Vehicular Technology, 59(9), 4289-4297.

Brunner, F. K., H. Hartinger and L. Troyer (1999) GPS signal diffraction modelling: The stochastic SIGMA-Δ model. Journal of Geodesy, 73(5), 259-267.

Caceres, M. A., F. Penna, H. Wymeersch and R. Garello (2011) Hybrid cooperative positioning based on distributed belief propagation. IEEE Journal on Selected Areas in Communications, 29(10), 1948-1958.

Cai, C., C. He, R. Santerre, L. Pan, X. Cui and J. Zhu (2016) A comparative analysis of measurement noise and multipath for four constellations: GPS, BeiDou, GLONASS and Galileo. Survey Review, 48(349), 287-295.

Cai, C., and Y. Gao (2013) Modeling and assessment of combined GPS/GLONASS Precise Point Positioning. GPS Solutions, 17(2), 223-236.

Carrillo-Arce, L. C., E. D. Nerurkar, J. L. Gordillo and S. I. Roumeliotis (2013) Decentralized multi-robot cooperative localization using covariance intersection. In 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, 3-7 November, Tokyo, Japan, 1412-1417.

Chandu, B., R. Pant and K. Moudgalya (2007) Modeling and simulation of a precision navigation system using pseudolites mounted on airships. In Proceedings of the 7th AIAA Aviation, Technology, Integration, and Operations Conference, 18-20 September, Belfast, Northern Ireland.

Chang, G. (2014) Alternative formulation of the Kalman filter for correlated process and observation noise. IET Science, Measurement & Technology 8(5), 310-318.

Chen, L., W. Song, W. Yi, C. Shi, Y. Lou and H. Guo (2017) Research on a method of real-time combination of precise GPS clock corrections. GPS Solutions, 21(1), 187-195.

Choudhury, M., and C. Rizos (2010) Slow structural deformation monitoring using Locata–a trial at Tumut Pond Dam. Journal of Applied Geodesy, 4(4), 177-187.

Cohen, C. E., B. S. Pervan, D. G. Lawrence, H. S. Cobb, J. D. Powell and B. W. Parkinson (1994) Real−time flight testing using integrity beacons for GPS category III precision landing. Navigation, 41(2), 145-157.

Collins, P., S. Bisnath, F. Lahaye and P. Héroux (2010) Undifferenced GPS ambiguity resolution using the decoupled clock model and ambiguity datum fixing. Navigation, 57(2), 123-135.

Crespillo, O. G., E. Nossek, A. Winterstein, B. Belabbas and M. Meurer (2015) Use of High Altitude Platform Systems to augment ground based APNT systems. In 2015 IEEE/AIAA 34th Digital Avionics Systems Conference, 13-18 September, Prague, Czech Republic.

Dai, L., J. Zhang, C. Rizos, S. Han and J. Wang (2000) GPS and pseudolite integration for deformation monitoring applications. In Proceedings of the 13th International Technical Meeting of the Satellite Division of The Institute of Navigation, 19-22 September, Salt Lake City, Utah, USA, 1-8.

Dauphin, Y., R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli and Y. Bengio (2014) Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In Advances in Neural Information Processing Systems, 08-13 December, Montreal, Canada, 2, 2933-2941.

de Oliveira, P. S., L. Morel, F. Fund, R. Legros, J. F. G. Monico, S. Durand and F. Durand (2017) Modeling tropospheric wet delays with dense and sparse network configurations for PPP-RTK. GPS Solutions, 21(1), 237-250.

Deo, M., and A. El-Mowafy (2018) Triple-frequency GNSS models for PPP with float ambiguity estimation: Performance comparison using GPS. Survey Review, 50(360), 249-261.

Doong, S. H. (2009) A closed-form formula for GPS GDOP computation. GPS Solutions, 13(3), 183-190.

Du, S., and Y. Gao (2012) Inertial aided cycle slip detection and identification for integrated PPP GPS and INS. Sensors, 12(11), 14344-14362.

El-Mowafy, A. (2017) Impact of predicting real-time clock corrections during their outages on Precise Point Positioning. Survey Review, 2017, 1-10.

El-Mowafy, A., M. Deo and N. Kubo (2017) Maintaining real-time Precise Point Positioning during outages of orbit and clock corrections. GPS Solutions, 21(3), 937-947.

Elsobeiey, M. (2015) Precise Point Positioning using triple-frequency GPS measurements. Journal of Navigation, 68(3), 480-492.

Elsobeiey, M., and A. El-Rabbany (2011) On modelling of second-order ionospheric delay for GPS Precise Point Positioning. Journal of Navigation, 65(1), 59-72.

Falco, G., M. Pini and G. Marucco (2017) Loose and tight GNSS/INS integrations: Comparison of performance assessed in real urban scenarios. Sensors, 17(2), 255.

Gao, C., F. Wu, W. Chen and W. Wang (2011) An improved weight stochastic model in GPS Precise Point Positioning. In Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering, 16-18 December, Changchun, China, 629-632.

Gao, Y., and X. Shen (2002) A new method for carrier-phase-based Precise Point Positioning. Navigation, 49(2), 109-116.

Gao, Z., W. Shen, H. Zhang, M. Ge and X. Niu (2016) Application of Helmert variance component based adaptive Kalman filter in multi-GNSS PPP/INS tightly coupled integration. Remote Sensing, 8(7), 553.

Gao, Z., W. Shen, H. Zhang, X. Niu and M. Ge (2016) Real-time kinematic positioning of INS tightly aided multi-GNSS ionospheric constrained PPP. Scientific Reports, 6, 30488.

Gao, Z., H. Zhang, M. Ge, X. Niu, W. Shen, J. Wickert and H. Schuh (2017) Tightly coupled integration of multi-GNSS PPP and MEMS inertial measurement unit data. GPS Solutions, 21(2), 377-391.

Ge, M., G. Gendt, M. Rothacher, C. Shi and J. Liu (2008) Resolution of GPS carrier-phase ambiguities in Precise Point Positioning (PPP) with daily observations. Journal of Geodesy, 82(7), 389-399.

Geng, J., and Y. Bock (2013) Triple-frequency GPS Precise Point Positioning with rapid ambiguity resolution. Journal of Geodesy, 87(5), 449-460.

Geng, J., X. Meng, A. H. Dodson, M. Ge and F. N. Teferle (2010) Rapid re-convergences to ambiguity-fixed solutions in Precise Point Positioning. Journal of Geodesy, 84(12), 705-714.

Geng, J., X. Meng, A. H. Dodson and F. N. Teferle (2010) Integer ambiguity resolution in Precise Point Positioning: method comparison. Journal of Geodesy, 84(9), 569-581.

Giorgi, G., and P. Teunissen (2012) GNSS carrier phase-based attitude determination. In Recent Advances in Aircraft Technology (IntechOpen), 193-220.

Goel, S., A. Kealy, V. Gikas, G. Retscher, C. Toth, D. G. Brzezinska and B. Lohani (2017) Cooperative localization of unmanned aerial vehicles using GNSS, MEMS inertial, and UWB sensors. Journal of Surveying Engineering, 143(4), 04017007.

Grilli, E., F. Menna and F. Remondino (2017) A review of point clouds segmentation and classification algorithms. In International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 1-3 March, Nafplio, Greece, 339-344.

Gross, J. N., R. M. Watson, S. D'Urso and Y. Gu (2016) Flight-test evaluation of kinematic Precise Point Positioning of small UAVs. International Journal of Aerospace Engineering, 2016.

Hadas, T., and J. Bosy (2015). IGS RTS precise orbits and clocks verification and quality degradation over time. GPS Solutions, 19(1), 93-105.

Hamilton J. D. (1994) Time Series Analysis. Princeton University Press, Princeton, New Jersey, USA.

Han, H., T. Xu and J. Wang (2016) Tightly coupled integration of GPS ambiguity fixed Precise Point Positioning and MEMS-INS through a troposphere-constrained adaptive Kalman filter. Sensors, 16(7), 1057.

Han, S. (1997) Quality-control issues relating to instantaneous ambiguity resolution for real-time GPS kinematic positioning. Journal of Geodesy, 71(6), 351-361.

Harrington, R. L., and J. T. Dolloff (1976) The inverted range: GPS user test facility. In Proceedings of IEEE Position Location and Navigation Symposium, November 1-3, San Diego, California, USA, 204-211.

He, C., B.Yu and Z. Deng (2016) Wireless time synchronization for multiple UAV-borne pseudolites navigation system. In China Satellite Navigation Conference 2016 Proceedings, Volume II, Lecture Notes in Electrical Engineering, 389, Springer, Singapore.

Heo, Y. J., J. Cho and M. B. Heo (2010) Improving prediction accuracy of GPS satellite clocks with periodic variation behaviour. Measurement Science and Technology, 21(7), 073001.

Howard, A., M. J. Mataric and G. S. Sukhatme (2002) Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. Distributed Autonomous Robotic Systems 5, 299-308.

Howind, J., H. Kutterer and B. Heck (1999) Impact of temporal correlations on GPS-derived relative point positions. Journal of Geodesy, 73(5), 246-258.

Hu, Y., Z. Duan and D. Zhou (2010) Estimation fusion with general asynchronous multi-rate sensors. IEEE Transactions on Aerospace and Electronic Systems, 46(4), 2090-2102.

Huang, G. W., Q. Zhang and G. C. Xu (2014) Real-time clock offset prediction with an improved model. GPS Solutions, 18(1), 95-104.

Jiang, W., Y. Li and C. Rizos (2015a) Locata-based Precise Point Positioning for kinematic maritime applications. GPS Solutions, 19(1), 117-128.

Jiang, W., Y. Li and C. Rizos (2015b) Optimal data fusion algorithm for navigation using triple integration of PPP-GNSS, INS, and terrestrial ranging system. IEEE Sensors Journal, 15(10), 5634-5644.

Jiang, W., Y. Li and C. Rizos (2015c) Precise indoor positioning and attitude determination using terrestrial ranging signals. Journal of Navigation, 68(2), 274-290.

Juan, J. M., et al. (2012) Enhanced Precise Point Positioning for GNSS users. IEEE Transactions on Geoscience and Remote Sensing, 50(10), 4213-4222.

Julier, S. J., and J. K. Uhlmann (1997) A non-divergent estimation algorithm in the presence of unknown correlations. In Proceedings of the 1997 American Control Conference, 4-6 June, Albuquerque, New Mexico, USA, 4, 2369-2373.

Jwo, D. J., and C. Lai (2007) Neural network-based GPS GDOP approximation and classification. GPS Solutions, 11(1), 51-60.

Kalafus, R. M., A. L. J. Van Dierendonck and N. A. Pealer (1986) Special Committee 104 recommendations for differential GPS service. Navigation, 33(1), 26-41.

Kang, G., L. Tan, B. Hua and F. Zheng (2013) Study on pseudolite system for BeiDou based on dynamic and independent aircrafts configuration. In China Satellite Navigation Conference 2013 Proceedings, 15-17 May, Wuhan, China, 159-172.

Kazmierski, K., T. Hadas and K. Sośnica (2018) Weighting of multi-GNSS observations in real-time Precise Point Positioning. Remote Sensing, 10(1), 84.

Kia, S. S., S. F. Rounds and S. Martínez (2014) A centralized-equivalent decentralized implementation of extended Kalman filters for cooperative localization. In 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 14-18 September, Chicago, Illinois, USA, 3761-3766.

Kia, S. S., S. Rounds and S. Martínez (2016) Cooperative localization for mobile agents: A recursive decentralized algorithm based on Kalman-filter decoupling. IEEE Control Systems Magazine, 36(2), 86-101.

Kim, D., B. Park, S. Lee, A. Cho, J. Kim and C. Kee (2008) Design of efficient navigation message format for UAV pseudolite navigation system. IEEE Transactions on Aerospace and Electronic Systems, 44(4), 1342-1355.

Kingma, D. P., and J. Ba (2014) Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Knoop, V. L., P. F. de Bakker, C. Tiberius and B. van Arem (2017) Lane determination with GPS Precise Point Positioning. IEEE Transactions on Intelligent Transportation Systems, 18(9), 2503-2513.

Ko, J., and D. Fox (2009) GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models. Autonomous Robots, 27(1), 75-90.

Ko, J., D. J. Kleint, D. Fox and D. Haehnel (2007) GP-UKF: Unscented Kalman filters with Gaussian process prediction and observation models. In Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, 29 October-2 November, San Diego, California, USA, 1901-1907.

Koch, K. R. (1986) Maximum likelihood estimate of variance components. Bulletin Gæodésique, 60(4), 329-338.

Kong, J., X. Mao and S. Li (2014) BDS/GPS satellite selection algorithm based on polyhedron volumetric method. In 2014 IEEE/SICE International Symposium on System Integration, 13-15 December, Tokyo, Japan, 340-345.

Laurichesse, D., F. Mercier, J. P. Berthias, P. Broca and L. Cerri (2009) Integer ambiguity resolution on undifferenced GPS phase measurements and its application to PPP and satellite precise orbit determination. Navigation, 56(2), 135-149.

Lee, K., H. Baek and J. Lim (2016) Enhanced positioning algorithm of ARPS for improving accuracy and expanding service coverage. Sensors, 16(8), 1284.

Lee, K., H. Baek and J. Lim (2018) Relay-based positioning in TDMA networks. IEEE Systems Journal, 12(4), 3849-3852.

Lee, K., and E. N. Johnson (2017) State estimation using Gaussian process regression for coloured noise systems. In 2017 IEEE Aerospace Conference, 4-11 March, Big Sky, Montana, USA, 1-8.

Lee, K., H. Noh and J. Lim (2015) Airborne relay-based regional positioning system. Sensors, 15(6), 12682-12699.

Lei, W., G. Wu, X. Tao, L. Bian and X. Wang (2017) BDS satellite-induced code multipath: Mitigation and assessment in new-generation IOV satellites. Advances in Space Research, 60(12), 2672-2679.

Leung, K. Y. K., T. D. Barfoot and H. H. T. Liu (2010) Decentralized localization of sparsely-communicating robot networks: A centralized-equivalent approach. IEEE Transactions on Robotics, 26(1), 62-77.

Li, B., L. Lou and Y. Shen (2015) GNSS elevation-dependent stochastic modeling and its impacts on the statistic testing. Journal of Surveying Engineering, 142(2), 04015012.

Li, B., L. Zhang and S. Verhagen (2017) Impacts of BeiDou stochastic model on reliability: Overall test, w-test and minimal detectable bias. GPS Solutions, 21(3), 1095-1112.

Li, G., J. Wu, W. Liu and C. Zhao (2016) A new approach of satellite selection for multi-constellation integrated navigation system. In China Satellite Navigation Conference 2016 Proceedings, Volume III, Lecture Notes in Electrical Engineering, Springer, Singapore, 390, 359-371.

Li, G., C. Xu, P. Zhang and C. Hu (2012) A modified satellite selection algorithm based on satellite contribution for GDOP in GNSS. In Advances in Mechanical and Electronic Engineering, Lecture Notes in Electrical Engineering, Springer, Berlin, Heidelberg, 176, 415-421.

Li, H., and F. Nashashibi (2013) Cooperative multi-vehicle localization using split covariance intersection filter. IEEE Intelligent Transportation Systems Magazine, 5(2), 33-44.

Li, P., and X. Zhang (2014) Integrating GPS and GLONASS to accelerate convergence and initialization times of Precise Point Positioning. GPS Solutions, 18(3), 461-471.

Li, P., and X. Zhang (2015) Precise Point Positioning with partial ambiguity fixing. Sensors, 15(6), 13627-13643.

Li, W., P. Teunissen, B. Zhang and S. Verhagen (2013) Precise Point Positioning using GPS and Compass observations. In China Satellite Navigation Conference 2013 Proceedings, 15-17 May, Wuhan, China, 367-378.

Li, X., M. Ge, X. Dai, X. Ren, M. Fritsche, J. Wickert and H. Schuh (2015) Accuracy and reliability of multi-GNSS real-time precise positioning: GPS, GLONASS, BeiDou, and Galileo. Journal of Geodesy, 89(6), 607-635.

Li, X., M. Ge, J. Douša and J. Wickert (2014) Real-time Precise Point Positioning regional augmentation for large GPS reference networks. GPS Solutions, 18(1), 61-71.

Li, X., M. Ge, H. Zhang and J. Wickert (2013) A method for improving uncalibrated phase delay estimation and ambiguity-fixing in real-time Precise Point Positioning. Journal of Geodesy, 87(5), 405-416.

Li, X., X. Li, Y. Yuan, K. Zhang, X. Zhang and J. Wickert (2018) Multi-GNSS phase delay estimation and PPP ambiguity resolution: GPS, BDS, GLONASS, Galileo. Journal of Geodesy, 92(6), 579-608.

Li, X., X. Zhang, X. Ren, M. Fritsche, J. Wickert and H. Schuh (2015) Precise positioning with current multi-constellation global navigation satellite systems: GPS, GLONASS, Galileo and BeiDou. Scientific Reports 5, 8328.

Li, Y., Y. Gao and J. Shi (2016) Improved PPP ambiguity resolution by COES FCB estimation. Journal of Geodesy, 90(5), 437-450.

Li, Z., Y. Yao, J. Wang and J. Gao (2017) Application of improved robust Kalman filter in data fusion for PPP/INS tightly coupled positioning system. Metrology and Measurement Systems, 24(2), 289-301.

Liu, M., M. A. Fortin and R. Landry (2009) A recursive quasi-optimal fast satellite selection method for GNSS receivers. In Proceedings of ION GNSS 2009, Institute of Navigation, 22-25 September, Savannah, Georgia, USA, 2061-2071.

Liu, S., F. Sun, L. Zhang, W. Li and X. Zhu (2016) Tight integration of ambiguity-fixed PPP and INS: Model description and initial result. GPS Solutions, 20(1), 39-49.

Liu, T., Y. Yuan, B. Zhang, N. Wang, B. Tan and Y. Chen (2017) Multi-GNSS Precise Point Positioning (MGPPP) using raw observations. Journal of Geodesy, 91(3), 253-268.

Liu, Y., S. Ye, W. Song, Y. Lou and S. Gu (2017) Rapid PPP ambiguity resolution using GPS + GLONASS observations. Journal of Geodesy, 91(4), 441-455.

Lou, Y., F. Zheng, S. Gu, C. Wang, H. Guo and Y. Feng (2016) Multi-GNSS Precise Point Positioning with raw single-frequency and dual-frequency measurement models. GPS Solutions, 20(4), 849-862.

Lu, C., F. Zus, M. Ge, R. Heinkelmann, G. Dick, J. Wickert and H. Schuh (2016) Tropospheric delay parameters from numerical weather models for multi-GNSS precise positioning. Atmospheric Measurement Techniques, 9(12), 5965-5973.

Luo, X., M. Mayer and B. Heck (2009) Improving the stochastic model of GNSS observations by means of SNR-based weighting. In Observing Our Changing Earth, International Association of Geodesy Symposia, Springer, Berlin, Heidelberg, 133, 725-734.

Lv, Y., Y. Duan, W. Kang, Z. Li and F. Wang (2015) Traffic flow prediction with big data: A deep learning approach. IEEE Transactions on Intelligent Transportation Systems, 16(2), 865-873.

Marquardt, D. W. (1963) An algorithm for least-squares estimation of nonlinear parameters. Journal of the Society for Industrial and Applied Mathematics, 11(2), 431-441.

Mensing, C., and J. J. Nielsen (2010) Centralized cooperative positioning and tracking with realistic communications constraints. In 2010 7th Workshop on Positioning, Navigation and Communication, 11-12 March, Dresden, Germany, 215-223.

Mokhtarzadeh, H., and D. Gebre-Egziabher (2014) Cooperative inertial navigation. Navigation, 61(2), 77-94.

Montenbruck, O., P. Steigenberger and A. Hauschild (2015) Broadcast versus precise ephemerides: A multi-GNSS perspective. GPS Solutions, 19(2), 321-333.

Montenbruck, O., P. Steigenberger and A. Hauschild (2018) Multi-GNSS signal-in-space range error assessment – Methodology and results. Advances in Space Research, 61(12), 3020-3038.

Montenbruck, O., et al. (2017) The Multi-GNSS Experiment (MGEX) of the International GNSS Service (IGS)–achievements, prospects and challenges. Advances in Space Research, 59(7), 1671-1697.

Mosavi, M. (2011) Applying genetic algorithm to fast and precise selection of GPS satellites. Asian Journal of Applied Sciences, 4(3), 229-237.

Mosavi, M., and M. Sorkhi (2009) An efficient method for optimum selection of GPS satellites set using recurrent neural network. In 2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 14-17 July, Singapore, 245-249.

Nerurkar, E. D., S. I. Roumeliotis and A. Martinelli (2009) Distributed maximum a posteriori estimation for multi-robot cooperative localization. In 2009 IEEE International Conference on Robotics and Automation, 12-17 May, Kobe, Japan, 1402-1409.

Nguyen-Tuong, D., J. R. Peters and M. Seeger (2009) Local Gaussian process regression for real time online model learning. In Advances in Neural Information Processing Systems, 7-10 December, Vancouver, British Columbia, Canada, 1193-1200.

Nie, Z., Y. Gao, Z. Wang, S. Ji and H. Yang (2018) An approach to GPS clock prediction for real-time PPP during outages of RTS stream. GPS Solutions, 22(1), 14.

Odijk, D., and P. J. G. Teunissen (2013) Characterization of between-receiver GPS-Galileo inter-system biases and their effect on mixed ambiguity resolution. GPS Solutions, 17(4), 521-533.

Odijk, D., B. Zhang and P. J. G. Teunissen (2015) Multi-GNSS PPP and PPP-RTK: Some GPS+BDS results in Australia. In China Satellite Navigation Conference 2015 Proceedings Volume II, Lecture Notes in Electrical Engineering, Springer, Berlin, Heidelberg, 341, 613-623.

Ozimek, I., T. Javornik and F. Dovis (2004) Navigation-related services over stratospheric platforms. Electrotechnical Review, 71(3), 96-102.

Pallavicini, M. B., F. Dovis, E. Magli and P. Mulassano (2001) HeliNet project: The current status and the road ahead. In Proceedings of the Data Systems in Aerospace, 28 May-1 June, Nice, France.

Pan, Z., H. Chai and Y. Kong. (2017) Integrating multi-GNSS to improve the performance of Precise Point Positioning. Advances in Space Research, 60(12), 2596-2606.

Park, B., D. Kim, T. Lee, C. Kee, B. Paik and K. Lee (2008) A feasibility study on a regional navigation transceiver system. Journal of Navigation, 61(2), 177-194.

Peng, A., G. Ou and G. Li (2014) Fast satellite selection method for multi-constellation Global Navigation Satellite System under obstacle environments. IET Radar, Sonar & Navigation, 8(9), 1051-1058.

Pervan, B. S., and B. W. Parkinson (1997) Cycle ambiguity estimation for aircraft precision landing using the Global Positioning System. Journal of Guidance, Control, and Dynamics, 20(4), 681-689.

Qi, C., H. Su, K. Mo and L. J. Guibas (2016) PointNet: Deep learning on point sets for 3D classification and segmentation. arXiv preprint arXiv:161200593.

Rabbou, M. A., and A. El-Rabbany (2015) Precise Point Positioning using multi-constellation GNSS observations for kinematic applications. Journal of Applied Geodesy, 9(1), 15-26.

Rao, C. R. (1971) Minimum variance quadratic unbiased estimation of variance components. Journal of Multivariate Analysis, 1(4), 445-456.

Raquet, J., G. Lachapelle, W. Qiu, C. Pelletier, T. Nash, F. B. Snodgrass, P. Fenton and T. Holden (1996) Development and testing of a mobile pseudolite concept for precise positioning. Navigation, 43(2), 149-165.

Rasmussen, C. E., and C. K. Williams (2004) Gaussian processes in machine learning. Advanced Lectures on Machine Learning, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 3176, 63-71.

Rasmussen, C. E., and C. K. Williams (2006) Gaussian Processes for Machine Learning. MIT Press Cambridge.

Ristic, B., S. Arulampalam and N. Gordon (2004) Beyond the Kalman filter. IEEE Aerospace and Electronic Systems Magazine, 19(7), 37-38.

Rizos, C., G. Roberts, J. Barnes and N. Gambale (2010) Experimental results of Locata: A high accuracy indoor positioning system. In 2010 International Conference on Indoor Positioning and Indoor Navigation, 15-17 September, Zurich, Switzerland, 1-7.

Roongpiboonsopit, D., and H. A. Karimi (2009) A multi-constellations satellite selection algorithm for integrated global navigation satellite systems. Journal of Intelligent Transportation Systems, 13(3), 127-141.

Roumeliotis, S. I., and G. A. Bekey (2002) Distributed multirobot localization. IEEE Transactions on Robotics and Automation, 18(5), 781-795.

Rui, G., and M. Chitre (2010) Cooperative positioning using range-only measurements between two AUVs. In OCEANS'10 IEEE SYDNEY, 24-27 May, Sydney, Australia, 1-6.

Satirapod, C., and M. Luansang (2008) Comparing stochastic models used in GPS Precise Point Positioning technique. Survey Review, 40(308), 188-194.

Seepersad, G., and S. Bisnath (2015) Reduction of PPP convergence period through pseudorange multipath and noise mitigation. GPS Solutions, 19(3), 369-379.

Shi, J. (2012) Precise Point Positioning Integer Ambiguity Resolution With Decoupled Clocks. Ph.D thesis, University of Calgary, Calgary, Alberta, Canada.

Shi, J., C. Xu, J. Guo and Y. Gao (2014) Local troposphere augmentation for real-time Precise Point Positioning. Earth, Planets and Space, 66(1), 30.

Shu, Y., R. Fang and J. Liu (2017) Stochastic models of very high-rate (50 Hz) GPS/BeiDou code and phase observations. Remote Sensing, 9(11), 1188.

Simon, D., and H. El-Sherief (1995) Navigation satellite selection using neural networks. Neurocomputing, 7(3), 247-258.

Small, D. (2017) Method and Device for Chronologically Synchronizing a Kinematic Location Network. U.S. Patent Application 15/327, 333.

Spangenberg, M., V. Calmettes, O. Julien, J. Tourneret and G. Duchateau (2010) Detection of variance changes and mean value jumps in measurement noise for multipath mitigation in urban navigation. Navigation, 57(1), 35-52.

Stansell Jr, T. A. (1986) RTCM SC-104 recommended pseudolite signal specification. Navigation, 33(1), 42-59.

Strutz, T. (2016) Data Fitting and Uncertainty: A practical introduction to weighted least squares and beyond. Springer Vieweg.

Suykens, J. A., and J. Vandewalle (1999) Least squares support vector machine classifiers. Neural Processing Letters, 9(3), 293-300.

Swaszek, P. F., R. J. Hartnett, K. C. Seals and R. Swaszek (2017) A temporal algorithm for satellite subset selection in multi-constellation GNSS. In Proceedings of ION ITM 2017, Institute of Navigation, 30 January - 2 February, Monterey, California, USA, 1147-1159.

Tegedor, J., O. Øvstedal and E. Vigen (2014) Precise orbit determination and point positioning using GPS, GLONASS, Galileo and BeiDou. Journal of Geodetic Science, 4(1), 65-73.

Teng, Y., and J. Wang (2016) A closed-form formula to calculate geometric dilution of precision (GDOP) for multi-GNSS constellations. GPS Solutions, 20(3), 331-339.

Teunissen, P. J. G., and A. R. Amiri-Simkooei (2008) Least-squares variance component estimation. Journal of Geodesy, 82(2), 65-82.

Teunissen, P. J. G., and A. Khodabandeh (2015) Review and principles of PPP-RTK methods. Journal of Geodesy, 89(3), 217-240.

Tiberius, C., and F. Kenselaar (2000) Estimation of the stochastic model for GPS code and phase observables. Survey Review, 35(277), 441-454.

Tiberius, C., and F. Kenselaar (2003) Variance component estimation and precise GPS positioning: Case study. Journal of Surveying Engineering, 129(1), 11-18.

Tsujii, T., C. Rizos, J. Wang, L. Dai, C. Roberts and M. Harigae (2001) A navigation/positioning service based on pseudolites installed on stratospheric airships. In 5th International Symposium on Satellite Navigation Technology & Applications, Canberra, Australia, 24-27.

Vásárhelyi, G., C. Virágh, G. Somorjai, N. Tarcai, T. Szörényi, T. Nepusz and T. Vicsek (2014) Outdoor flocking and formation flight with autonomous aerial robots. In 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 14-18 September, Chicago, Illinois, USA, 3866-3873.

Walter, T., J. Blanch and V. Kropp (2016) Satellite selection for multi-constellation SBAS. In Proceedings of the 29th International Technical Meeting of the Satellite Division of The Institute of Navigation, 12-16 September, Portland, Oregon, USA, 1350-1359.

Wanasinghe, T. R., G. K. I. Mann and R. G. Gosine (2014) Decentralized cooperative localization for heterogeneous multi-robot system using split covariance intersection filter. In 2014 Canadian Conference on Computer and Robot Vision, 6-9 May, Montreal, Quebec, Canada, 167-174.

Wang, J., N. Gopaul and B. Scherzinger (2009) Simplified algorithms of variance component estimation for static and kinematic GPS single point positioning. Journal of Global Positioning Systems, 8(1), 43-52.

Wang, J. (1999) Stochastic modeling for real-time kinematic GPS/GLONASS positioning. Navigation, 46(4), 297-305.

Wang, L., Z. Li, M. Ge, F. Neitzel, Z. Wang and H. Yuan (2018) Validation and assessment of multi-GNSS real-time Precise Point Positioning in simulated kinematic mode using IGS Real-Time Service. Remote Sensing, 10(2), 337.

Wang, M., H. Chai and Y. Li (2017) Performance analysis of BDS/GPS Precise Point Positioning with undifferenced ambiguity resolution. Advances in Space Research, 60(12), 2581-2595.

Wei, M., J. Wang and J. Li (2012) A new satellite selection algorithm for real-time application. In 2012 International Conference on Systems and Informatics, 19-20 May, Yantai, China, 2567-2570.

Wu, C., W. Su and Y. Ho (2011) A study on GPS GDOP approximation using support-vector machines. IEEE Transactions on Instrumentation and Measurement, 60(1), 137-145.

Wu, Z., Q. Cai and M. Fu (2017) Covariance intersection for partially correlated random vectors. IEEE Transactions on Automatic Control, 63(3), 619-629.

Yang, Y., L. Song and T. Xu (2002) Robust estimator for correlated observations based on bifactor equivalent weights. Journal of Geodesy, 76(6-7), 353-358.

Yao, J., A. T. Balaei, M. Hassan, N. Alam and A. G. Dempster (2011) Improving cooperative positioning for vehicular networks. IEEE Transactions on Vehicular Technology, 60(6), 2810-2823.

Yu, X., and J. Gao (2017) Kinematic Precise Point Positioning using multi-constellation Global Navigation Satellite System (GNSS) observations. ISPRS International Journal of Geo-Information, 6(1), 6.

Zarei, N. (2014) Artificial intelligence approaches for GPS GDOP classification. International Journal of Computer Applications, 96(16), 16-21.

Zhang, H., Z. Gao, M. Ge, X. Niu, L. Huang, R. Tu and X. Li (2013) On the convergence of ionospheric constrained Precise Point Positioning (IC-PPP) based on undifferential uncombined raw GNSS observations. Sensors, 13(11), 15708-15725.

Zhang, M., and J. Zhang (2009) A fast satellite selection algorithm: Beyond four satellites. IEEE Journal Selected Topics in Signal Processing, 3(5), 740-747.

Zhang, P., R. Tu, Y. Gao, R. Zhang and N. Liu (2018) Improving the performance of multi-GNSS time and frequency transfer using robust Helmert variance component estimation. Sensors, 18(9), 2878.

Zhang, X., F. Zhu, Y. Zhang, F. Mohamed and W. Zhou (2018) The improvement in integer ambiguity resolution with INS aiding for kinematic Precise Point Positioning. Journal of Geodesy, 1-18

Zhao, Y. (2017) Applying time-differenced carrier phase in nondifferential GPS/IMU tightly coupled navigation systems to improve the positioning performance. IEEE Transactions on Vehicular Technology, 66(2), 992-1003.

Zheng, J., and F. Guo (2016) An adaptive stochastic model for GPS observations and its performance in Precise Point Positioning. Survey Review, 48(349), 296-302.

Zhou, F., D. Dong, W. Li, X. Jiang, J. Wickert and H. Schuh (2018) GAMP: An open-source software of multi-GNSS Precise Point Positioning using undifferenced and uncombined observations. GPS Solutions, 22(2), 33.

Zhou, X., W. Dai, J. Zhu, Z. Li and Z. Zou (2008) Helmert variance component estimation-based Vondrak filter and its application in GPS multipath error mitigation. In VI Hotine-Marussi Symposium on Theoretical and Computational Geodesy, International Association of Geodesy Symposia, Springer, Berlin, Heidelberg, 132, 287-292.

Zhou, Y., and O. Tuzel (2017) VoxelNet: End-to-end learning for point cloud based 3D object detection. arXiv preprint arXiv:171106396.

Zhou, Z., Y. Li, C. Fu and C. Rizos (2016) Least-squares support vector machine-based Kalman filtering for GNSS navigation with dynamic model real-time correction. IET Radar, Sonar & Navigation, 11(3), 528-538

Zhou Z., J. Wu, Y. Li, C. Fu and H. Fourati (2017) Critical issues on Kalman filter with coloured and correlated system noises. Asian Journal of Control, 19(6), 1905-1919.

Zhu, S. (2018) An optimal satellite selection model of Global Navigation Satellite System based on genetic algorithm. In China Satellite Navigation Conference 2018 Proceedings, Lecture Notes in Electrical Engineering, Springer, Singapore, 498, 585-595.

Zumberge, J. F., M. B. Heflin, D. C. Jefferson, M. M. Watkins and F. H. Webb (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.