Quality control procedures for GNSS precise point positioning in the presence of time correlated residuals

Matthew Emyr David Goode

Thesis submitted for the degree of

Doctor of Philosophy

School of Civil Engineering and Geosciences

Newcastle University

June 2014

Abstract

Precise point positioning (PPP) is a technique for processing Global Navigation Satellite Systems (GNSS) data, often using recursive estimation methods e.g. a Kalman Filter, that can achieve centimetric accuracies using a single receiver. PPP is now the dominant real-time application in offshore marine positioning industry. For high precision real-time applications it is necessary to use high rate orbit and clock corrections in addition to high rate observations. As Kalman filters require input of process and measurement noise statistics, not precisely known in practice, the filter is non-optimal.

Geodetic quality control procedures as developed by Baarda in the 1960s are well established and their extension to GNSS is mature. This methodology, largely unchanged since the 1990s, is now being applied to processing techniques that estimate more parameters and utilise many more observations at higher rates. "Detection, Identification and Adaption" (DIA), developed from an optimal filter perspective and utilising Baarda's methodology, is a widely adopted GNSS quality control procedure. DIA utilises various test statistics, which require observation residuals and their variances. Correct derivation of the local test statistic requires residuals at a given epoch to be uncorrelated with those from previous epochs. It is shown that for a non-optimal filter the autocorrelations between observations at successive epochs are non-zero which has implications for proper application of DIA. Whilst less problematic for longer data sampling periods, high rate data using real-time PPP results in significant time correlations between residuals over short periods.

It is possible to model time correlations in the residuals as an autoregressive process. Using the autoregressive parameters, the effect of time correlation in the residuals can be removed, creating so-called whitened residuals and their variances. Thus a whitened test statistic can be formed, that satisfies the preferred assumption of uncorrelated residuals over time. The effectiveness of this whitened test statistic and its impact on quality control is evaluated.

Acknowledgements

I wish to thank my supervisors in Newcastle, Dr Stuart Edwards and Professor Philip Moore, whose guidance, support and great knowledge throughout the project has been invaluable. Thanks are also due to my industrial supervisor in the Netherlands, Dr Kees de Jong, who despite the distance has always been on hand to answer questions and provide feedback.

The dataset collected at Albemarle Barracks, Northumberland, has been key to the research and would not have been possible without Captain Neale Smiles, who arranged access to the airfield, and to Martin Robertson and Ian Martin for assisting in collecting the data. Thanks must also go to Dr Simon Williams for his assistance in understanding aspects of time correlation.

I must also extend my gratitude to my colleagues and fellow students in the geodesy research group, who have always been on hand to provide invaluable advice and helpful discussion.

Finally, I wish to thank my family and friends who have supported me throughout my studies. Without their encouragement and support none of this would be possible.

Contents

B Derivation of the test statistic for the post-fit residual in a recursive estimator   **276**

List of Figures

List of Tables

List of abbreviations

AR	autoregression, autoregressive
CODE	Center for Orbit Determination in Europe
CORS	Continuously Operating Reference Station
DIA	Detection, Identification and Adaptation
GLONASS	Globalnaya Navigatsionnaya Sputnikovaya Sistema
GNSS	Global Navigation Satellite Systems
GPS	Global Positioning System
IGS	International GNSS Service
LOM	local overall model
LS	local slippage
MDB	minimal detectable bias
MDE	marginally detectable error
PPP	precise point positioning

Chapter 1

Introduction

1.1 Background

1.1.1 Development of GNSS processing

A great deal of positioning tasks in the offshore industry require precise Global Navigation Satellite System (GNSS) positioning for both navigation and surveying. Examples of its application include measuring the subsidence of oil rigs (Hofmann-Wellenhof et al. 2008), as part of bathymetric surveying (Ernstsen et al. 2006) and aiding the positioning of unmanned autonomous vehicles (Groves 2008).

Offshore survey companies have traditionally used differential GPS in order to position vessels in real time. This involves a permanently mounted reference receiver from which corrections to the pseudoranges obtained from code observations are calculated and then transmitted via a terrestrial radio signal or more commonly a satellite link to the vessels requiring positioning. These corrections are combined with the observation data and used in a least squares estimation process to calculate the position of the vessel. However the quality of correction decreases with distance from the base station at a rate of 0.22 m for every 100 km from the broadcast site (Monteiro et al. 2005). This situation can be mitigated to a certain extent by creating a dense network of reference receivers, therefore minimising the distance to the base stations. However, this is not a particularly feasible solution, as it is both costly and in some cases impossible to implement where a vessel is in the middle of

the ocean.

A development of this idea was the concept of global DGPS (GDGPS) pioneered by the Jet Propulsion Laboratory (JPL) at the National Aeronautics and Space Administration (NASA). Instead of creating a dense network of reference stations as previously mentioned, a relatively sparse but well distributed network of receivers using both code and carrier observations is used to calculate corrections to orbit and clock parameters. These corrections are independent of distance from a particular reference station and have the advantage that accuracy is homogeneous globally (Lachapelle and Petovello 2006). This approach has been shown to have a root mean square (RMS) vertical error of around 19 cm and a RMS horizontal error of less than 10 cm (Muellerschoen et al. 2001).

The principle of GDGPS has been taken one step further by using both code and carrier phase observations and estimating tropospheric and carrier phase ambiguity parameters (Zumberge et al. 1997, Kouba and Héroux 2001). This method is known as precise point positioning (PPP) and uses a recursive estimator, such as a Kalman filter, which allows parameters which vary predictably over time to be modelled by prediction and filtering. This has brought real-time kinematic positional 3D accuracy to the order of 5 to 20 cm. In order to get an accurate position the recursive estimator must first converge, a process known as initialization that can typically take more than 20 minutes (Lachapelle and Petovello 2006).

1.1.2   Quality control

In a broad sense quality control is a process whereby the performance of a system is evaluated, compared to a required goal and if necessary acted upon (Juran and Godfrey 1999). In the context of GNSS processing quality control can be defined as fitness for purpose with adequate precision and reliability (Teunissen 1998b). In order to get the best possible result errors in the measurements must be correctly detected and either removed or dealt with accordingly at the data processing stage (Kim and Langley 2001).

No measurement can be said to be free of errors and GNSS processing is no excep-

tion. Errors can either be classed random or biases. A random error is unpredictable and as such independent of any other event. It is assumed for GNSS measurements that these random errors are normally distributed (IMCA/OGP 2011). Conversely biases cannot be statistically modelled. They take the form of either a gross or systematic error. A gross error or blunder is one which occurs due to operator error, such as an incorrect antenna height. A systematic error is repeatable, hence it can be observed and then predicted from a known mathematical process. Using this information it can either be eliminated or modelled.

Current procedures for offshore GNSS positioning are based on guidelines laid down by the International Marine Contractors Association (IMCA) and International Association of Oil & Gas Producers (OGP) (IMCA/OGP 2011), although the recommended quality control procedures are largely based on a report by the UK Offshore Operators Association (UKOAA 1994). These procedures were drawn up in the context of DGPS services which use a single-frequency code only baseline approach. As previously mentioned, the DGPS method corrects the pseudorange whereas the PPP method uses corrections to the orbit and clock parameters. A greater number of parameters are estimated during PPP processing and both code and carrier phase observations are utilised. As GNSS has developed the number of systems and frequencies that are available has increased, which increases the complexity of the processing algorithms. This also applies to DGPS, but to a lesser extent.

1.1.3 Time correlation

In many GNSS processing applications it is assumed that the observations are uncorrelated, but in many cases this has been shown not to be true (Bona 2000). Making such an assumption results in an incomplete stochastic model when estimating position and other parameters. There have been many studies into determining the correlations between the observations using techniques such as variance-component estimation (VCE) (Satirapod 2001, Bischoff et al. 2006, Amiri-Simkooei 2007).

Nowadays many real-time applications require high rate GNSS data and it is

common for GNSS receivers to sample observations at 1 Hz up to rates up of 50 Hz (Genrich and Bock 2006). For the purposes of this study the term 'high rate' will be applied to datasets with sample rates of 1 Hz and higher. It has been observed that sampling GNSS observations at higher rates can lead to time correlation and this is more significant as the sampling rate is increased (Borre and Tiberius 2000). This can be caused by receiver equipment hardware delays, such as tracking loop error, and multipath (see Chapter 2). Some studies extend the estimation of covariances by also taking into account temporal correlations (Wang et al. 2002, Wang et al. 2012).

When using a recursive estimator that employs filtering and prediction, such as a Kalman filter, in addition to the correct specification of the stochastic model for the observations, which may be correlated, the stochastic model of the dynamic model must also be correct. An estimator that uses the 'true' values is said to be optimal, but since in practice this very difficult this leads to a non-optimal filter. It is possible to use the residuals from the recursive estimator to adjust the observation and process noise, of which the adaptive Kalman filter is an example (Mehra 1970, Mohamed and Schwarz 1999, Yang et al. 2001, Hu et al. 2003).

In addition to the correlations in the observations, it is also important to note that PPP is heavily reliant on the orbit and clock corrections. The manner in which these slowly time-varying errors are calculated results in them being time correlated, as shown by Olynik (2002). Since these are given as known values in the PPP measurement model they become another potential source for correlation in the estimator.

1.2 Research motivation

Much work has been done to improve the accuracy of PPP through increasing redundancy by adding GLONASS observations and recent advancements in PPP integer ambiguity resolution. However, less emphasis has been placed on quality control. Additionally, although much work has been done into the difficulties associated with high rate data, much of this work has concentrated on relative positioning (El Ra-

banny 1994, Bona 2000, Miller et al. 2011). PPP is an interesting combination of the challenge of correctly specifying the stochastic model of the observations, choosing the appropriate process noise for the dynamic model and the additional complications that come with high rate orbit and clock corrections.

1.3   Aim and objectives

The overall aim of the research is to assess the extent to which time correlation is a factor in PPP processing and establish a method of statistical testing that can be performed in the presence of time correlation. To achieve this aim the following objectives will be pursued:

1. Undertake a review of the PPP measurement model and the error sources that must be modelled or removed.

2. Give an overview of the principal estimation methods that can be used in PPP.

3. Derive and explain the GNSS quality control procedures and how they are applied in PPP.

4. Assess the level of time correlation in PPP residuals.

5. Investigate methods by which the time correlation in PPP residuals can be reduced or removed.

6. Develop a modified statistical test by which time correlated residuals can be tested.

7. Identify how any novel method can be implemented in real-time PPP software.

1.4   Thesis outline

The thesis has been ordered into eight chapters. Chapter 1 provides the background to the research, including the historical context to PPP, the importance of quality control and the importance of time correlation. The motivation for the research is outlined along with the aim and the objectives of the project.

The aim of Chapter 2 is to introduce PPP as a GNSS processing technique. The measurement model for GNSS observations is introduced along with a brief overview of the error sources in GNSS, how they are applicable to PPP and their contribution, where appropriate, to time correlation. Precise orbits and clocks are also introduced as a fundamental ingredient in PPP processing, as well as the issue of convergence time. Recent developments in PPP methodology and the user requirements for PPP in the offshore industry are also discussed.

In Chapter 3 least squares estimation is introduced, after which an overview of the Kalman filter is given. A least squares derivation of the Kalman filter is then presented, which gives context to the semi-recursive Kalman filter. This is introduced as a recursive method whereby some parameters in the state vector are predicted, whereas others are estimated at each step without any a priori values.

An overview of the quality control procedures for GNSS are given in Chapter 4. This includes hypothesis testing, the T test statistic and measures of reliability. It is shown that a fundamental assumption for a Kalman filter and recursive quality control theory is that the residuals are not temporally correlated.

The concept of filter optimality is presented in Chapter 5, which shows that when a recursive estimator is non-optimal the residuals may appear temporally correlated. This is shown to be the case using the residuals obtained from processing four datasets using the PPP software used in the study. Two methods for dealing with the time correlation are then presented.

In Chapter 6 it is first shown that the residuals from our PPP software can be modelled as a first order autoregressive (AR) process. The different AR estimation methods are reviewed and using the methods in Chapter 5, it is shown that the residual can be 'whitened' using the AR coefficients. The AR process is then extended so that the order of the AR process can be automatically determined.

Since it is possible to generate 'whitened' residuals that are not temporally correlated, a 'whitened' test statistic is derived in Chapter 7 that satisfies the assumption that there is no time correlation in the residuals. This is then implemented into our software and its ability to detect biases in the observations is assessed. Suggestions

are made as to how the statistic may be practically implemented in production software.

The thesis is concluded in Chapter 8 by summarising the key findings, identifying how the research objectives have been met and suggesting how the research may be continued.

Chapter 2

GNSS precise point positioning

2.1   Introduction

It is assumed that the reader is already familiar with Global Navigation Satellite Systems (GNSS) as a technology and as such the underlying principles of the technology will not be discussed here. A variety of text books are available that explain the fundamentals of GNSS, such as Strang and Borre (1997), Leick (2004), Kaplan and Hegarty (2006) and Hofmann-Wellenhof et al. (2008). It is also assumed that the reader is aware of the evolution of GNSS processing prior to the introduction of precise point positioning (PPP), including differential GNSS techniques. More information on the evolution of GNSS processing is available in the aforementioned texts.

Precise point positioning is a GNSS processing technique that uses code pseudorange and carrier phase observations from a single receiver to determine position. This is in contrast to other precise GNSS processing techniques which require at least two receivers in order to compute a baseline solution. The technique, first implemented by Zumberge et al. (1997), achieves a precise position through the high precision of the carrier phase observations and by using precise satellite orbit and clock values in place of those from the broadcast ephemeris.

In this chapter the positioning model for PPP will be presented and discussed. As with all GNSS processing techniques the error sources in GNSS positioning must be taken care of, however since much has been written about GNSS error sources

(Strang and Borre 1997, Leick 2004, Kaplan and Hegarty 2006, Hofmann-Wellenhof et al. 2008) in general only those significant to PPP estimation will be considered. Where appropriate examples will be given to exemplify the magnitude of the error and the effect on the solution. Since a fundamental feature of PPP is the use of precise satellite orbits and clocks these will be considered, along with their extension to real-time applications. The need to wait for the solution to converge and the impacts of this will then be discussed.

2.2   Measurement model

As mentioned in the previous section, the PPP observation model consists of code and carrier phase observations. These two measurement types have very different precision, but each has a unique contribution. The carrier phase is the more precise of the two observation types and is the key to the high precision attained in the PPP technique. However, the initial unknown ambiguity of each carrier phase observation must be estimated and hence this precision cannot be realised immediately. The code observations are used to aid solving for these parameters during initialisation and to ensure that the system of equations is regular.

The measurement model for a code observation at receiver r tracking satellite s on frequency i at any given epoch, after Teunissen and Kleusberg (1998b), is given as

$$
\begin{aligned}
p_{r,i}^s = R_r^s &+ \gamma_i I_{r,1}^s + T_r^s \\
&+ c[\delta t_r - \delta t^s] \\
&+ \eta_{r,i} - \eta_i^s \\
&+ m_{r,i,p}^s + e_r^k
\end{aligned}
\tag{2.1}
$$

where $p_{r,i}^s$ is the pseudorange observable, R_r^s the geometric distance between the receiver and the satellite, $I_{r,1}^s$ the delay due to the ionosphere on the first frequency (e.g. L1 for GPS), T_r^s the delay due to the troposphere, c the speed of light through a vacuum, δt_r and δt^s are the receiver and satellite clock biases, $\eta_{r,i}$ the delay at the

receiver between the antenna and the signal correlator, η_i^s the delay at the satellite between the signal generator and antenna, $m_{r,i,p}^s$ the delay due to multipath, and e_r^k the pseudorange measurement error resulting from additional parameters, such as tracking error (see §2.2.7). The ionospheric delay term refers to the first frequency; the effect on other frequencies is found through scaling by the constant $\gamma_i = \lambda_i^2/\lambda_1^2$, where λ_i is the carrier wavelength corresponding to frequency i.

Similarly, the measurement model for a carrier phase observation at receiver r tracking satellite s on frequency i at any given epoch, after Teunissen and Kleusberg (1998b), is given by

$$
\begin{aligned}
\phi_{r,i}^s = R_r^s &- \gamma_i I_r^s + T_r^s \\
&+ c[\delta t_r - \delta t^s] \\
&+ \mu_{r,i} - \mu_i^s \\
&+ \lambda_i[\varphi_r - \varphi^s] + \lambda_i N_{r,i}^s \\
&+ m_{r,i,\phi}^s + \varepsilon_r^k
\end{aligned}
\tag{2.2}
$$

where $\phi_{r,i}^s$ is the carrier phase observable, $\mu_{r,i}$ the delay at the receiver between the antenna and the signal correlator, μ_i^s the delay at the satellite between the signal generator and antenna, φ_r and φ^s the non-zero initial phases of the signals generated at the receiver and satellite respectively, $N_{r,i}^s$ the integer carrier phase bias, $m_{r,i,\phi}^s$ the delay due to multipath, and ε_i^k the carrier phase measurement error resulting from additional parameters, such as tracking error (see §2.2.7) and phase wind-up (see §2.2.6). Some of these additional parameters, for example phase wind-up, can be accounted for a priori, whereas others cannot.

Since the hardware delays and the non-zero initial phases of the carrier phase bias cannot be separated Eq. (2.2) is often simplified to

$$
\begin{aligned}
\phi_{r,i}^s = R_r^s &- \gamma_i I_r^s + T_r^s \\
&+ c[\delta t_r - \delta t^s] \\
&+ \mu'_{r,i} - \mu'^s_i + \lambda_i N_r^s \\
&+ m_{r,i,\phi}^s + \varepsilon_i^k
\end{aligned}
\tag{2.3}
$$

where $\mu'_{r,i}$ and μ'^{s}_{i} contain both the hardware biases and the initial phases at the receiver and satellite respectively.

The geometric distance R_r^s in Eq. (2.1) - (2.3) is given by

$$R_r^s = \sqrt{(X^s - X_r)^2 + (Y^s - Y_r)^2 + (Z^s - Z_r)^2} \qquad (2.4)$$

where (X_r, Y_r, Z_r) and (X^s, Y^s, Z^s) are the receiver and satellite coordinates respectively. The estimated coordinates may need to be adjusted to take into account the difference between the instantaneous and conventional coordinates, such as the effect of solid earth tides (see §2.2.6).

For positioning applications it is the receiver position, contained within Eq. (2.4), that is the unknown that is of primary interest. In the proceeding subsections each of the remaining parameters given in Eq. (2.1) - (2.3) will be looked at in greater detail.

2.2.1   Ionosphere

The ionosphere is a dispersive medium for GNSS signals and as such the delay varies with frequency, hence the frequency dependency expressed in Eq. (2.1) - (2.2). From these equations it can also be seen that carrier phase measurements are advanced by the ionosphere, whereas pseudorange measurements are delayed (Langley 1998).

Since the effect of the ionosphere varies with frequency it is possible to eliminate most of the delay by taking a linear combination of more than one frequency, which results in the first-order terms cancelling out. This accounts for over 99% of the ionospheric effect. It is most common to combine L1 and L2, referred to as LC. It is also possible to make a linear combination of the carrier phase and pseudorange measurements to remove the first-order ionospheric effects (Teunissen and Kleusberg 1998b). The remaining delay is made up of second- and third-order effects, which are of centimetre and millimetre magnitude respectively (Elmas et al. 2011).

When calculating a baseline solution the effect of the ionosphere is removed if the distance between the two stations is not long (Langley 1998). It is also possible to model the ionospheric effect, such as that by Klobuchar (1987). However,

ionospheric models are not sufficiently accurate enough for centimetre-level PPP applications.

An alternative to forming a linear combination is to use an ionosphere float solution, where the unknown ionospheric delays are explicitly estimated instead of being eliminated (Teunissen 1997, Odijk 2000). The ionosphere float method is typically done with L1 and L2 observations for GPS, although it can be done with any two frequencies. It has been shown in Wells et al. (1987) that the ionosphere free and ionosphere float processes are mathematically equivalent.
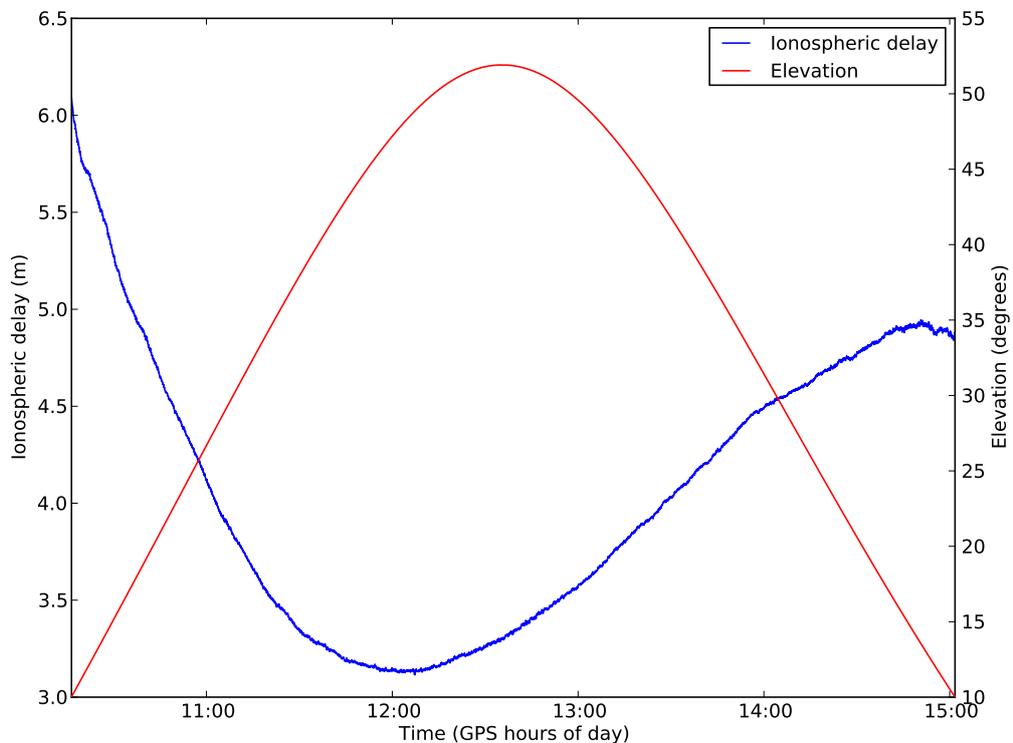
Figure 2.1: Estimated ionospheric path delay between satellite G05 and a receiver located in Perth, Australia on 5th August 2011 generated using the E-HP/PPP software.

As there is a linear relationship between the first-order ionospheric effect on differing frequencies it can be seen from Eq. (2.1) - (2.2) that only the L1 ionospheric delay term is estimated for each satellite. Using the L1 ionospheric effect as a datum, the scalar γ_i is used to transform this value to other frequencies. The variation in the ionospheric delay for an arbitrary satellite pass is shown in Figure 2.1. It can be seen that there is a clear link between the elevation of the satellite and the ionospheric

delay. This is intuitive, since the GNSS signal must pass through more atmosphere as the elevation decreases.

When estimating the ionospheric effect it is possible to treat the parameter as one that either can be estimated at each epoch or one that can be propagated from one epoch to the next in a recursive estimator such as a Kalman filter. Although there are advantages in doing the latter, in areas of the world where the ionosphere may be particularly changeable a bias could be easily introduced.

Irrespective of whether the ionosphere is estimated or eliminated, it is only the first-order effects that are considered. Although the higher order effects do not have as great an impact, when left unestimated they are lumped into the residual error and may be temporally correlated.

2.2.2 Troposphere

Tropospheric refraction is divided into hydrostatic and non-hydrostatic components, more commonly referred to as the dry and wet delay respectively. The former is much easier to model as it follows the law of ideal gases and can be calculated using pressure measurements taken at the receiver. The wet delay is very site-specific and a function of temperature and partial pressure of water vapour (Hofmann-Wellenhof et al. 2008). When calculating a baseline solution, if the distance between the two receivers is not great the effect of the troposphere is removed as the signal is assumed to have passed through the same part of the troposphere (Langley 1998). In the case of PPP the dry delay can be modelled, but the wet delay is estimated as an unknown parameter.

Models for tropospheric delay involve a model for the zenith delay, i.e. the delay if the satellite were immediately above the receiver, and a mapping function for elevation dependence, since as elevation decreases the signal must pass through more atmosphere and hence the delay is increased. The simplest mapping function is to multiply by the reciprocal of the sine of the elevation. Examples of mapping functions include that by Marini (1972), the Neill Mapping Function by Neill (1996) and the Vienna Mapping Function (Boehm et al. 2006).

The two main models for a priori calculation of the hydrostatic zenith delay are those by Hopfield (1969) and Saastamoinen (1973). The latter is a function of total ground pressure, latitude and receiver altitude, as well as a set of generalised approximations.
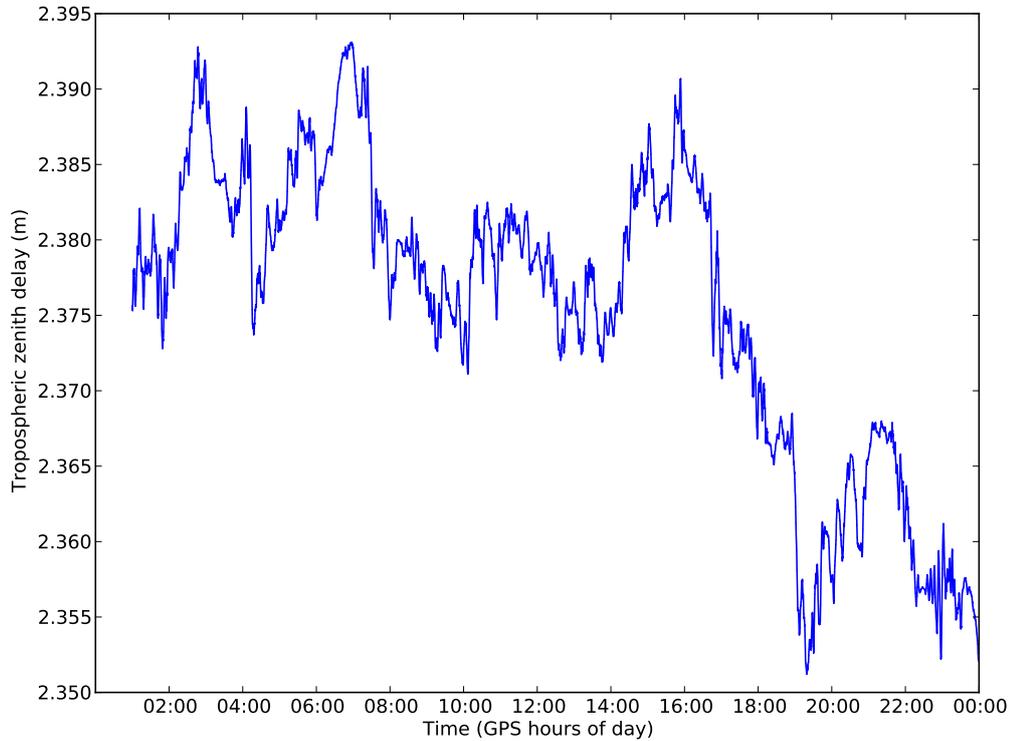
Figure 2.2: Estimated tropospheric zenith delay at a receiver located in Perth, Australia on 5th August 2011 generated using the E-HP/PPP software.

An example of the tropospheric zenith delay over 23 hours for a station located in Perth, Australia is shown in Figure 2.2. The delay has a range of 45 mm and is the combination of the modelled dry component and the estimated wet component.

It is also possible to model the estimated troposphere as a gradient parameter. This allows azimuthal variations of tropospheric refraction to be taken into account (Meindl et al. 2004, Kleijer 2004). When estimating zenith delays it is important to note that by not taking into account this azimuthal refraction leads to small residual errors in the tropospheric estimate. As the troposphere is considered to change slowly over time and is estimated as a common parameter in the recursive estimator (see Chapter 3) these residual errors may be temporally correlated.

2.2.3 Clock biases

As seen in Eq. (2.1) – (2.2) the clock biases consist of receiver and satellite components.

The receiver clock varies unpredictably from one epoch to the next, although the receiver clock drift varies very slowly (Petovello 2011). The receiver clock and its drift are usually estimated as nuisance parameters. How the unpredictable part of this variation compares with the accuracy requirements depends on the application and therefore influences whether from an estimation point of view it is considered to vary unpredictably from one epoch to the next or not. It is possible to eliminate the receiver clock bias by forming a single difference between satellites or estimate it as an unknown parameter. Leandro et al. (2011) state that the common approach adopted in PPP software is to estimate the receiver clock bias. If more than one satellite system is being used, such as GPS and GLONASS, a separate receiver clock bias must be estimated for each system. This can also be estimated as an offset by choosing a reference clock, since the difference between the two clock values is relatively stable over time it can be constrained and propagated from one epoch to the next.

A correction to the clock for each satellite is transmitted as a polynomial in the broadcast ephemeris and the post-processed clock values generated by the IGS (see §2.3) improve on these clock corrections. These corrections are treated as known values in the PPP technique and is one of the major contributors to its high accuracy. In real-time applications these clock values are predicted and supplied to the user in real-time.

2.2.4 Hardware delays

There are a number of hardware delays that must be accounted for at both the satellite and the receiver. In the case of the satellite, these include the thermal noise that is associated with the electronics of the satellite itself as well as the transmitting antenna. At the receiver there are delays associated with the receiving antenna, the connecting cables and also within the hardware of the receiver itself. It is common

for the ambiguity terms to be lumped with the hardware delays.

2.2.5 Multipath

It is generally assumed that the signal received at the GNSS antenna from a satellite has followed an uninterrupted path. However, this is not always the case. In some cases the antenna can receive additional indirect signals that are reflected off nearby surfaces. This could be an adjacent building, the ground or even the GNSS antenna itself. Reflections can also occur at the satellite at the time of transmission. This phenomenon is known as multipath. In the case where only one signal is received, but it is an indirect signal, this is known as non-line-of-sight (NLOS) reception.
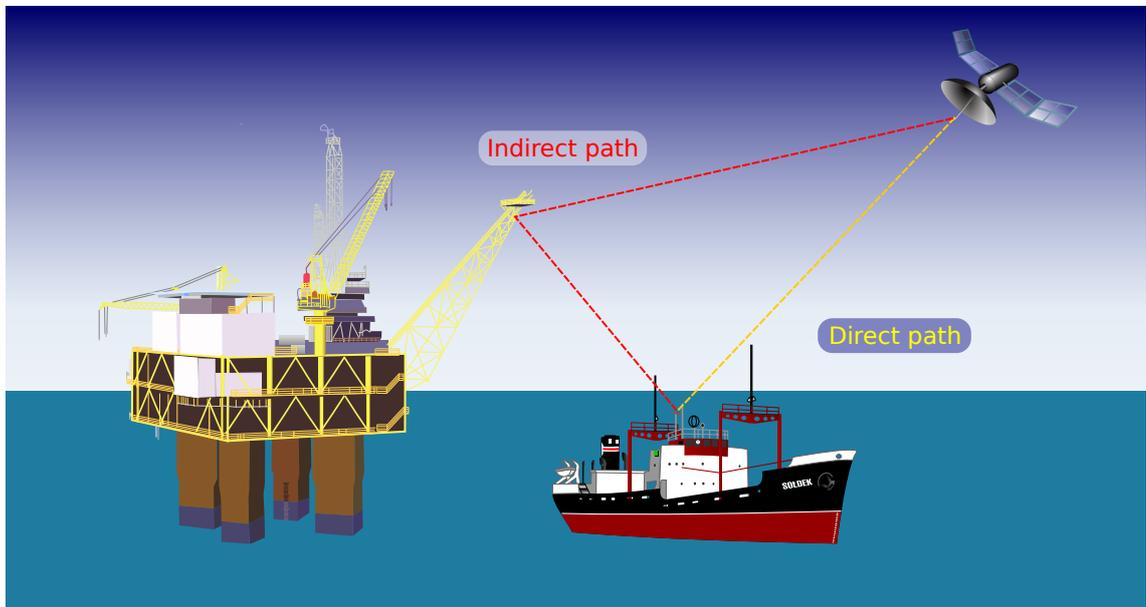
Figure 2.3: An illustration of the direct and indirect paths of a GNSS signal at the antenna. The diagram is not to scale.

Multipath signals disturb the time correlation shape that is used for time delay estimation in GNSS receivers due to the inseparability of direct and indirect signals (Hofmann-Wellenhof et al. 2008, Sahmoudi and Landry 2008). The effect of this a slowly varying error and hence can result in time correlated errors (Groves 2008).

In many offshore applications multipath can be prevalent due to the reflective nature of the sea surface. Certain applications, such as mobile offshore drilling units, are particularly affected as in addition to a number of tall structures, which may cause multipath, any moving structures, such as cranes, will result in varying

multipath conditions (IMCA/OGP 2011).

2.2.6 Additional parameters

Antenna offsets

The orbits in the GPS navigation message are relative to the satellite antenna phase centre, whereas the orbit and clock products provided by the IGS are relative to the satellite centre of mass. These offsets are expressed in a satellite-fixed coordinate system and differ between satellite types. Each satellite has an individual frequency dependent phase centre. Satellite specific z-offsets and block specific x- and y-offsets are provided in an antenna calibration file published by the IGS (Schmid et al. 2007). It is important to apply these offsets when undertaking positioning at the centimetre level. It is also important to have the most recent values, since the use of incorrect offset values may occur where the PRN of a satellite may change or be reassigned when a satellite is decommissioned. Furthermore, it is important to know the precise offsets of the phase centre of the receiving antenna. These are given for each frequency the antenna is capable of receiving. If the antenna can receive signals from more than one satellite system then these will also be given separately. These values are also contained within the antenna calibration file published by the IGS.

Antenna phase centre variation

The antenna reference point (ARP) describes the point to which all measurements are made relative to an antenna. In theory all received signals are relative to the electrical phase centre of the antenna. However, such a point does not exist in reality as the electrical phase centre will vary for each satellite based on its azimuth, elevation, signal intensity and frequency (Hofmann-Wellenhof et al. 2008). The difference between the mean electrical antenna phase centre and each individual measurement is expressed as a frequency-dependent phase centre offset (PCO) and an azimuth- and elevation-dependent phase centre variation (PCV). These values are either specified by the manufacturer of the antenna and found empirically through

calibration. It is important to apply these values as part of the measurement model.

Phase wind-up corrections

The GPS signal is a right-hand circularly polarised L-band signal and as such when a receiver changes direction relative to the satellite the phase that is recorded by the tracking loop varies. This can occur both when the receiver is moving and when it is static, since the satellite is always moving. This phenomenon is known as carrier phase wind-up (Wu et al. 1993). The effect of phase wind-up is at the few centimetre level and therefore for high precision applications such as PPP its effect is significant. As the phase wind-up only affects the carrier phase measurements, the application of corrections in PPP is particularly important as both pseudorange and carrier phase observations form part of the measurement model. More information on how to correct for phase wind-up can be found in Kouba (2009).

Solid earth tide

The gravitational attraction of the sun and moon that creates the ocean tides also has an effect on the solid earth and is referred to as the solid earth tide. Its effect is dependent on the latitude and longitude of the receiver and the frequency at which the tide occurs (Kouba 2009). In the case of differential systems with short baselines the effects of solid earth tides can largely be ignored, but for high precision long baseline applications and PPP the effect must be modelled. The solid earth tide displacement components for Perth, Australia, on 5th August 2011 are shown in Figure 2.4. It is also important to note that these tidal displacement components change with time and the pattern will be different each day.

The horizontal and vertical displacements are modelled using spherical harmonics of degree m and order n and calculated using the Love number h_{nm} and the Shida number l_{nm} (Kouba 2009). These are dimensionless numbers that represent the rigidy of the Erath and its potential to react to tidal forces. The values in Figure 2.4 were calculated using software by Milbert (2012) according to the method set out in Petit and Luzum (2010).
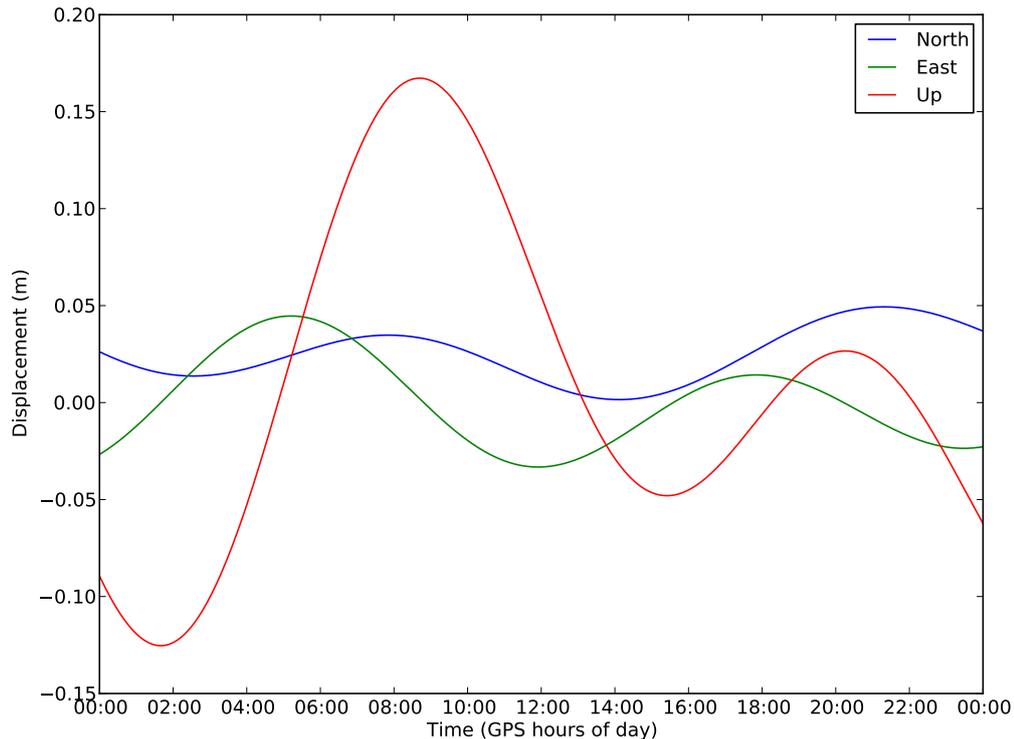
Figure 2.4: Computed solid earth tide on 5th August 2011 for Perth, Australia.

Ocean tide loading

Ocean tide loading is the displacement effect caused by the redistribution of the oceans' weight on the seafloor. Its effect is smaller than solid earth tides and as the effect is a function of distance from the ocean it is localised as a result (Kouba 2009). However, this displacement effect must be accounted for in sub-centimetre level applications and as such must be corrected for in a PPP implementation where high accuracy is required. It is is more critical to account for ocean tide loading at sites where it has a much greater effect, such as those close to the coast.

There exist a number of models that can be used to correct for the displacement caused by ocean tide loading (Ray 1999, Matsumoto et al. 2000, Lyard et al. 2006). These provide the amplitude and phase of the signal for the major tidal constituents, of which the greatest proportion of the effect comes from the 12-hour semi-diurnal and 24-hour diurnal periods (e.g. Clarke and Penna (2010)). A web service provided by Scherneck and Bos (2002) with eleven ocean loading models is a common source

of tidal coefficients.

2.2.7 Tracking loop error

The GNSS receiver generates code and carrier phase observables by correlating the signal that is received with a reference signal that is generated inside the receiver itself. The two observation types are generated using two separate methods.

There are two main types of carrier tracking loop: a phase lock loop (PLL) and a frequency lock loop (FLL). The PLL is more accurate, but more sensitive to dynamic stress than the FLL. In each case the tracking loop may employ a Costas loop. These must be used for data-modulated signals, such as the C/A and P(Y) codes. Since there is a compromise to be made between the different types of tracking loop a GNSS receiver often contains a combination of tracking loops that assist each other (Ward et al. 2006).

In order to generate the carrier phase observable the GNSS receiver must create two replicas of the reference carrier frequency that are 90 degrees phase shifted, known as the in-phase and quadrature phase signals (Ward et al. 2006). These are often referred to in the literature as the I/Q operations. The carrier loop filter analyses the beat phase caused by the Doppler shift on the signal, extracts the remaining frequency offset and the phase shift between the satellite signal and receiver in-phase signal. A wide filter bandwidth is needed for high dynamics, but the downside to this is that the noise level is increased (Hofmann-Wellenhof et al. 2008).

The predominant sources of error in a PLL carrier tracking loop are phase jitter and dynamic stress error and for an FLL it is frequency jitter, as a result of thermal noise, and dynamic stress error. The main source of jitter is a result of thermal noise and is a function of the carrier-power-to-noise density. The error due to jitter for a Costas PLL carrier tracking loop is given by Langley (1997) as

$$\sigma_{\mathrm{PLL}} \approx \sqrt{\frac{B_P}{c/n_0}\left[1 + \frac{1}{2\,T\,c/n_0}\right]} \cdot \frac{\lambda}{2\pi} \qquad (2.5)$$

where B_P is the carrier loop noise bandwidth (Hz); c/n_0 the carrier-power-to-noise

density ratio; T is the predetection integration time; and λ the carrier wavelength. Note that when the carrier-power-to-noise density ratio is expressed in decibel form the upper case equivalent, C/N_0, is used (Groves 2008) where

$$C/N_0 = 10 \log_{10}(c/n_0) \tag{2.6}$$

For signals of nominal strength this can be approximated as

$$\sigma_{\text{PLL}} \approx \sqrt{\frac{B_P}{c/n_0}} \cdot \frac{\lambda}{2\pi} \tag{2.7}$$

Taking $C/N_0 = 45$ dB-Hz and $B_P = 2$ Hz the error for GPS L1 carrier phase is 0.2 mm (Langley 1997).

The GNSS receiver uses a delay lock loop (DLL) in order to track the code that is modulated on the carrier. Its thermal noise is much higher than the carrier loop thermal noise (Ward et al. 2006). To generate the code observable the pseudo-random code modulated onto the carrier phase is correlated against three internal replica of the pseudorandom code. These are time shifted to give an early, prompt and late correlation result (Hofmann-Wellenhof et al. 2008). By calculating the transmit time from the satellite to the receiver it is possible to obtain a pseudorange measurement.

Thermal noise range error jitter and dynamic stress error are the major sources of code tracking loop error after the effects of interferences, such as multipath (see §2.2.5) (Ward et al. 2006). The DLL jitter for an early/late one-chip spacing correlator is given by Langley (1997) as

$$\sigma_{DLL} \approx \sqrt{\frac{\alpha B_L}{c/n_0}\left[1 + \frac{2}{T \ c/n_0}\right]} \cdot \lambda_c \tag{2.8}$$

where α is the dimensionless DLL discriminator correlator factor; B_L the equivalent code loop noise bandwidth (Hz); and λ_c the wavelength of the pseudorandom code.

This can be approximated for moderate to strong signals as

$$\sigma_{DLL} \approx \sqrt{\frac{\alpha B_L}{c/n_0}} \cdot \lambda_c \qquad (2.9)$$

Taking $\alpha = 0.5$, $C/N_0 = 45$ dB-Hz and $B_L = 0.8$ Hz the error for the GPS C/A code is 1.04 m (Langley 1997).

For this study it is important to consider the impact that the tracking loop error can have on time correlation in the observations (Abbott and Lillo 2003). Code-tracking errors can become correlated with the Doppler shift, although this should be at the millimetre level for high-accuracy geodetic receivers (Blewitt 1998). Carrier-tracking errors induced by dynamics are typically proportional to the line-of-site jerk.

For an introduction to the subject of receiver noise see Langley (1997) and for a more comprehensive overview of signal acquisition and processing consult Ward et al. (2006).

2.2.8   Code pseudorange smoothing

The pseudorange code measurement is inherently noisy, but by the use of code pseudorange smoothing it is possible to reduce this effect (Hofmann-Wellenhof et al. 2008). The smoothed pseudorange code measurement for frequency i is found recursively for an epoch k, where $k = 1, 2, \ldots$, as (Groves 2008, de Jong 2013)

$$p_{\mathrm{sm},i,k} = W_p \, p_{i,k} + (1 - W_p)(p_{\mathrm{sm},i,k-1} + \phi_{i,k} - \phi_{i,k-1}) \qquad (2.10)$$

where p_{sm} is the smoothed pseudorange code measurement and W_p is the code weighting factor. When using dual-frequency data this combination limits code-carrier divergence due to the ionosphere and the original structure of the code observations is preserved (de Jong 2013).

A variation of this algorithm is given by Hofmann-Wellenhof et al. (2008) that uses a time-dependent weight factor to form the smoothed code observation. Full weight is initially given to the code observation, but as time passes more weight is

given to the carrier phase observation. If there is a sudden change in the ambiguity of the carrier phase observation due to a cycle slip (see §4.2) the weight is reset.

An alternative method for code pseudorange smoothing is given by Teunissen (1991) whereby the smoothed observations are obtained using a recursive least squares estimator. This has the advantage of proper functional and stochastic modelling. Le and Teunissen (2006) show that when applied to PPP it is shown to yield a 30% improvement over traditional methods.

It is important to note that given the method used for smoothing the code observations this will inevitably introduce time correlation into the PPP estimation procedure and this should be taken into account when analysing the data.

2.3 Orbit and clock products

The International GNSS Service (IGS) provides post-processed tabulated values for the satellite orbits and satellite clock biases. These are computed as a combination of solutions from analysis centres around the world. The signal-in-space error for the broadcast orbits and satellite clocks given by Heng et al. (2011) are shown in Table 2.1. This is compared to the most precise products produced by the IGS, which have quoted accuracies of 2.5 cm for GPS orbits and 0.05 ns for satellite clocks (IERS 2011). The accuracies of the IGS products are summarised in Table 2.2.

Table 2.1: Standard deviation of signal-in-space (SIS) errors grouped by GPS satellite block type. (Heng et al. 2011)

	IIA	IIR	IIR-M
Radial (m)	0.243	0.130	0.145
Alongtrack (m)	1.258	0.921	1.000
Crosstrack (m)	0.675	0.575	0.594
Clock (m)	1.074	0.384	0.498

For post-processed PPP the IGS Final product yields the best results, although if results are required quickly then the latency of this product may mean that the Rapid or even Ultra-Rapid products must be used. Kouba and Héroux (2001) show the positional accuracy of a static solution using either the IGS Rapid or IGS Final both to be around the 2 cm level. It is noted that the IGS Rapid is of comparable

Table 2.2: [

Summary of the quality of the IGS orbit and clock products at end of 2011.]Summary of the quality of the IGS orbit and clock products at end of 2011 (IERS 2011)

Product	IGS Final	IGS Rapid	IGS Ultra Rapid	
			Adjusted	Predicted
Updates	Weekly	Daily	Every 6 h	Every 6 h
Delay	~ 13 days	17 hours	3 hours	Real-time
GPS orbits	2.0 cm	2.5 cm	3 cm	5 cm
GPS satellite clocks	0.05 ns	0.1 ns	~ 0.2 ns	~ 5 ns
Station clocks	0.05 ns	0.1 ns	-	-
GLONASS orbits	~ 5 cm	-	-	-

quality to the IGS Final product.
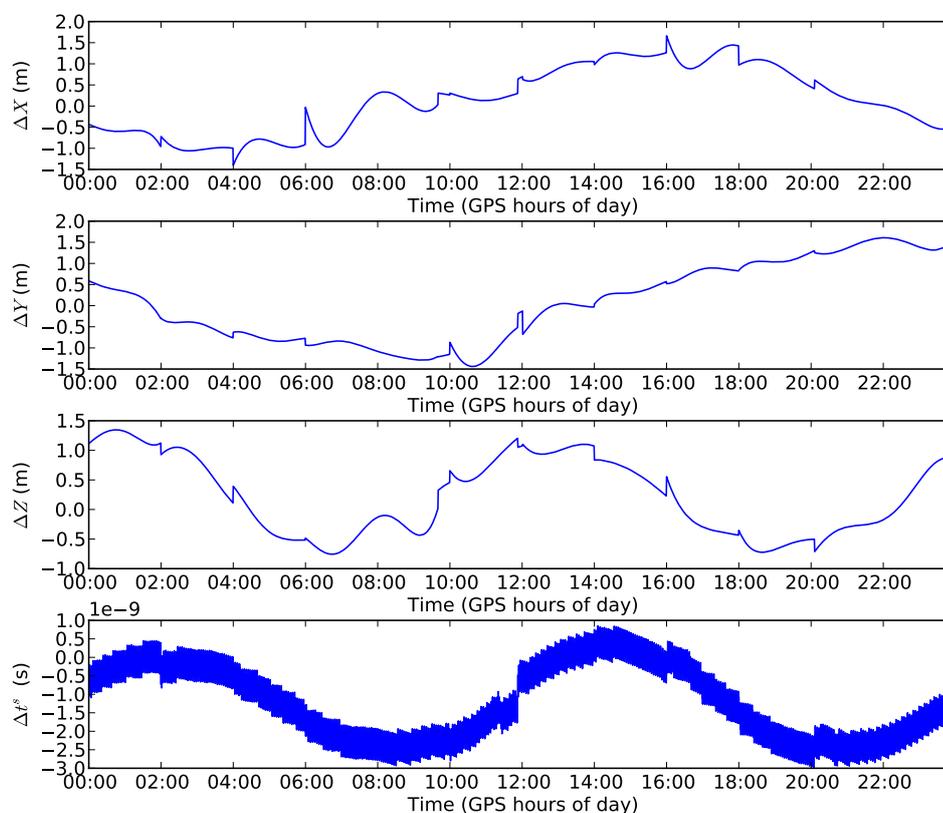
Figure 2.5: Difference between broadcast and IGS Final orbits and clocks for GPS SVN01 on 1st January 2012

Figure 2.5 shows the difference between the broadcast orbit and satellite clock correction and the IGS Final product. Since the IGS precise clocks at 30 second intervals have been used the broadcast clocks at 30 second intervals have been used. Since the Keplerian elements that model the orbit are refreshed every two hours it is possible to see jumps that occur whenever these values change. The difference

between the clock values shows a high frequency oscillation and a second much lower frequency noise, due to the unpredictability of satellite clock values.

In addition to the post-processed products, a number of institutions now produce GPS orbit and clock corrections in real-time. One of the first providers of such a service was NASA's Jet Propulsion Laboratory (JPL) (Muellerschoen et al. 2001). Gao and Chen (2004) report the positional accuracy of the JPL real-time product to be at the centimetre level for static and kinematic positioning applications. Since the launch of the JPL service a number of commercial companies undertake their own orbit and clock estimation (Melgard et al. 2009), which also incorporate corrections for GLONASS satellites. The IGS is also trialling a real-time orbit and clock service providing corrections using standard communication methods through the internet.

2.4   Convergence of PPP solutions

PPP implementations can either use batch least squares estimation or recursive estimation , such as the Kalman filter (see Chapter 3), to process the data. The batch least squares method takes all epochs of data and treats the estimation procedure as one single adjustment problem. In comparison, a recursive estimator processes each epoch separately, using all available information up to that point.

Whereas both methods can be used when data is being post-processed, when estimating in real-time only a recursive estimator is feasible as the batch solution would only be available once all data has been collected. The estimation begins using an initial value with a large a posteriori error. At the start of processing the carrier phase ambiguities are unknown and code observations are required to be able to separate them from the position and other parameters. As successive epochs are processed the ambiguity can be estimated with greater certainty and the a posteriori error decreases rapidly to a steady state. This period is known as the convergence time. It is the convergence time that is most crucial in real-time applications of PPP, since reliable positioning cannot begin before convergence has been reached.

GPS data from the IGS station ABPO, located on the island of Madagascar in the Indian Ocean, recorded on 25th December 2009 was processed using the E-HP/PPP
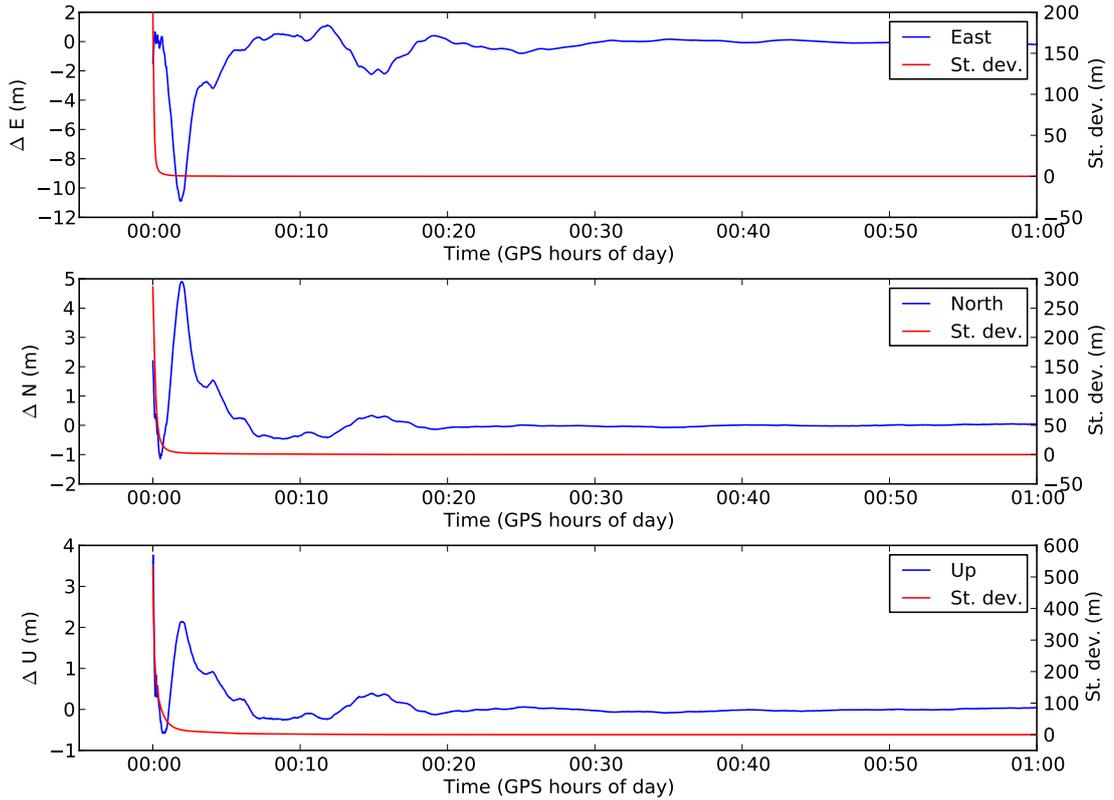
Figure 2.6: Difference between coordinates calculated using PPP in 'kinematic mode' and the known coordinates during the convergence period for ABPO (located on the island of Madagascar in the Indian Ocean) on 25th December 2009.

software (see §5.3.1) in kinematic mode, i.e. the position was allowed to change freely from one epoch to the next. Figure 2.6 shows the difference in the local east, north and up coordinates with respect to the known coordinates calculated using a static PPP solution over 24 hours. It can be seen that the east, north and up components of the coordinate settle to a steady value after approximately 30 minutes. Similarly, the a posteriori error is high at first and then falls rapidly to a steady value, although it can be seen that this cannot be used as an indicator of convergence.

2.5   Recent developments in PPP methodology

Since its inception there have been a number of developments in the PPP methodology that have brought increased accuracy and availability. Bisnath and Gao (2008) provide a comprehensive overview of the state of PPP and identify a number of areas of potential. A further update is given by Rizos et al. (2012) questioning the

continued need for differential positioning in light of advances in PPP methodology.

One such area is the reduction of the convergence time (see §2.4), which is an important aspect in commercial applications of PPP. Karabatic et al. (2010) have approached this problem by generating a regional set of satellite clocks, which takes into account spatial and temporal correlations between stations. Another approach has been to introduce additional navigation systems, although in much of the literature this does not appear to have brought significant improvements in convergence time (Cai and Gao 2007, Hesselbarth and Wanninger 2008, Anquela et al. 2013). Another approach by Elsobeiey and El-Rabbany (2012) is to compute the second-order ionospheric effects and apply these as corrections in their PPP software along with orbits and clock corrections calculated by taking into account second-order ionospheric effects. A 15% improvement in convergence time is reported.

In the case of double differenced processing the carrier phase ambiguities are integer, but in the case of standard PPP it is only possible to estimate carrier phase ambiguities as a float number. By estimating the so-called uncalibrated phase delays (UPDs) and providing these to the user it is possible to fix the ambiguities to integer values (Wuebbena et al. 2005, Ge et al. 2008, Teunissen et al. 2010).

In most PPP implementations the effect of the ionosphere can largely be removed by using a dual-frequency receiver and forming the ionosphere-free combination or by using an ionosphere float model (see §2.2.1). However, to use a single-frequency receiver, which is preferable in low-cost applications, the choice of ionosphere model is crucial (Chen and Gao 2005). This has prompted research into improving the accuracy of PPP using such receivers. Le et al. (2008) show that real-time regional ionospheric maps can be applied to the single-frequency PPP measurement model to improve the positional accuracy, although poor geographical coverage is a limitation.

With the increasing availability of GLONASS satellites much research has been done into integrating GLONASS observations with those of GPS. It is also reliant on precise orbit and clock products being available for GLONASS satellites. An early study by Cai and Gao (2007) showed that there was no significant improvement in the result achieved, mainly due to there not being sufficient additional satellites

to have a noticeable impact. Píriz et al. (2009) show that improvements in the constellation allow the computation of a GLONASS only static solution to sub-centimetre level, although the combined solution once again does not show significant improvement. It is also noted that a GLONASS only solution is unreliable at times. Melgard et al. (2009) show that the convergence time of kinematic results can be improved by an average of 39%, where this is defined as the 3D position being within 40 centimetres of the reference coordinates for more than 10 minutes. A more recent study by Cai and Gao (2013) reports a 50% improvement in accuracy for kinematic data and predicts further improvements with better GLONASS orbit and clock products. It also concludes by stating that improvements in functional and stochastic models will yield even better results.

2.6 User requirements

As the scope of this study is within the context of positioning for the offshore industry it is useful to consider the requirements of the user and how current PPP algorithms meet these. Only real-time and kinematic situations will be considered, as these requirements differ greatly from users requiring post-processed static solutions.

In the offshore industry PPP is often used as a replacement for traditional differential GNSS services. Guidelines set out in IMO (2001) state that high accuracy offshore applications are typically around 0.1 m (see Table 2.3). The accuracy of real-time PPP given by IMCA/OGP (2011) is around 15 cm in the horizontal (2σ) and 15-25 cm in the vertical (2σ). This suggests that for these type of applications the accuracy requirements are not yet met. However, the recent advances in integer ambiguity resolution for real-time PPP has shown that this can be reduced to 2.7 cm in the horizontal (Chen et al. 2011). This level of accuracy also opens up new applications to PPP and has seen it applied in the agricultural industry for precision farming (Bisnath and Gao 2008, van Bree and Tiberius 2012).

It is important to be able to measure the reliability of the test statistics used to screen observations (see §4.7). As such, the reliability indicators should give

Table 2.3: Summary of accuracy requirements in offshore positioning (IMO 2001). All values represent a 95% confidence limit.

	Accuracy	
	Horizontal (m)	Vertical (m)
Automatic docking	0.1	-
Hydrography	1 - 2	0.1
Dredging	0.1	0.1
Cable and pipeline laying	1.0	-
Offshore construction	0.1	0.1

a realistic estimate of the size of error that can be detected. As the accuracy of positioning improves the ability to detect smaller biases becomes increasingly important. Current algorithms can meet this expectation, although as the accuracy improves it is important to ensure that these measures are still relevant.

In many cases positioning systems are required at all times since much operational work is continuous. In some cases continuity may also be crucial, where loss of lock may compromise safety. This is reflected in the availability and continuity requirements set out in IMO (2001) (see Table 2.4). As such the system must always be available. For this reason two fully independent positioning systems are recommended (IMCA/OGP 2011). Current technology allows this availability criteria to be met. Since it is clear that availability is crucial in offshore applications it is important to ensure that any statistical testing of the PPP solution does not inadvertently result in a loss of positioning.

Consider the scenario where the minimum number of four satellites are being tracked. Should the quality control procedure in place cause carrier phase data on this satellite to be rejected then the system will not be solvable and a reset will be triggered. If the lack of positioning halts work then this can prove very costly. The chance of this scenario can be reduced with multi-constellation GNSS, as this brings improved satellite availability. The procedures presented in Chapter 4 are currently sufficient at doing this, although research into multiple outlier detection (Wang and Wang 2007, Knight et al. 2009, Baselga 2011) show that with increased redundancy comes the increased chance of more than one bias in the observations at a single epoch. Yang et al. (2013) show that the existing procedures based on the occurrence of a single outlier are not sufficient and that the theory should be

extended to cover multiple hypotheses.

Table 2.4: Summary of availability and continuity requirements in offshore positioning. (IMO 2001)

Application	Availability	Continuity
	% over 30 days	% over 3 hours
Automatic docking	99.8	99.97
Hydrography	99.8	-
Dredging	99.8	-
Cable and pipeline laying	99.8	-
Offshore construction	99.8	-

In real-time applications latency can have a significant impact on position and is reported by Chen et al. (2013) to be one of the principle challenges in PPP at present. It is shown in IMCA/OGP (2011) that as the age of corrections supplied to the user increases, the accuracy decreases. The latency of corrections to the user should be within one minute to ensure an accurate position can be achieved. In addition to the latency experienced in providing the corrections to the user the time taken to calculate a solution is crucial. As the sample rate of observation data increases the speed at which a PPP algorithm performs is also extremely important. Current algorithms are sufficient at providing low latency in computing position, although with higher rate observation data a more powerful CPU may be required. Any additional algorithms that complement the PPP procedure should similarly be designed with latency in mind.

2.7 Summary

In this chapter the concept of PPP as a positioning technique was introduced. It is a processing method that requires only one receiver and utilises precise orbits and clock corrections in order to achieve decimetre accuracy. The measurement model was presented, which consists of both pseudorange code and carrier phase observations. It is through the combination of these two observation types that the best results can be achieved, since the code observations are important for the initialisation period.

There are a number of error sources that appear in the model equations. These

must either be eliminated from the observation model by supplying known values or by modelling them, whereas others must be estimated as an unknown parameter. Some of the error sources, such as multipath and tracking noise, are known to cause time correlation and this must be taken into account during data analysis.

Since it is the precise orbits and clocks that are a major contributor to the accuracy of PPP these were considered, as well as the development of real-time services. The limiting factor of convergence time was highlighted as a caveat to the PPP technique. Finally, the requirements of users of PPP were considered and the extent to which current PPP algorithms meet these.

Chapter 3

The Kalman filter

3.1 Introduction

Geodetic problems have historically been solved using least squares adjustment theory developed by Carl Friedrich Gauss in the 19th century. When confined to batch solutions this is an adequate technique, but when processing data in real time, where the parameters being estimated vary predictably over time, it is necessary to use a recursive estimation technique.
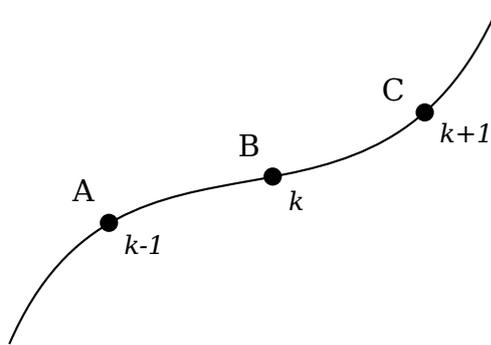
Figure 3.1: Positions of a moving vessel over three consecutive epochs.

Consider the example in Fig. 3.1 of a moving vessel at positions A, B and C at epochs $k-1$, k and $k+1$ respectively. At epoch k it is possible to estimate the position of the vessel at epoch $k+1$ using knowledge of the vessel's dynamics. This is known as prediction. Using an update in information at epoch $k+1$ it is possible to calculate an improvement on this predicted position. This is known as filtering. Using the additional information obtained at epoch $k+1$ it is then possible

to calculate corrected positions for the vessel at k, $k - 1$ and all previous epochs. This is process is known as smoothing.

One method that employs prediction and filtering is the Kalman filter (Kalman 1960) and along with its derivatives is widely applied across many fields (Leondos 1970, Pearson and Stear 1974, Harvey 1994, Haykin 2002). It is particularly suited to navigation systems as it utilises all data processed up to a particular epoch, but, as it is recursive, does not suffer from data storage issues. An historical background to the link between the Kalman filter equations and least squares adjustment theory is presented in Sorenson (1970).

In this chapter a brief summary of least squares estimation will be given in §3.2 and then in §3.3 the fundamental Kalman filter equations will be presented. Since most geodetic problems are nonlinear, §3.4 discusses the extension of the Kalman filter to nonlinear systems. In §3.5 the Kalman filter equations will then be presented as a derivation in terms of the least squares estimator. An alternative flavour of the Kalman filter is then derived in §3.6 that modifies the state vector to include parameters that cannot be predicted from one epoch to the next. Due to the nature of this modified filter, the predicted residuals cannot be used for identifying model errors. As such, the relationship between predicted and post-fit residuals is derived in §3.7. All of the above is then brought into the context of PPP in §3.8, along with a few notable implementation issues.

Since there are a number of forms of notation for Kalman filters, the style used in this chapter will be continued throughout the thesis. Refer to Appendix A for further explanation of the notation used.

3.2   Least squares estimation

Before looking at recursive estimation methods it is important to review least squares estimation as the basis of position fixing in geodesy. In this section the model using observation equations will be the focus, but the model of condition equations will also be briefly covered. The Kalman filter and least squares estimation are closely linked (Sorenson 1970) and it will be shown in §3.5 that the Kalman filter can

be derived with respect to the least squares model using the model of observation equations.

3.2.1 Model of observation equations

The standard least squares model using observation equations reads

$$E\{\underline{y}\} = A\hat{x} \quad ; \quad D\{\underline{y}\} = Q_y \tag{3.1}$$

where $\underline{y}$ is an $m \times 1$ random vector of observations, $\hat{x}$ is an $n \times 1$ vector of unknown parameters and A is the $m \times n$ design matrix, which consists of coefficients that form a linear transformation between the estimable parameters and observations. It is assumed that the $\underline{y}$ is Gaussian. Eq. (3.1) is also known as the Gauss-Markov model (Koch 1999).

The least squares estimator aims to provide the best linear unbiased estimate by minimising the weighted sum of the squares of residuals, $\underline{e}$ (Cross 1994),

$$\min \sum \hat{\underline{e}}^T Q_y^{-1} \hat{\underline{e}} \tag{3.2}$$

where the residual $\hat{\underline{e}}$ is the deviation of the observations from the estimator of their expected values (Koch 1999). This is done by forming the normal equations, which read

$$(A^T Q_y^{-1} A)\hat{\underline{x}} = A^T Q_y^{-1} \underline{y} \tag{3.3}$$

From this, the estimator for the unknown parameters is formed, which reads

$$\hat{\underline{x}} = (A^T Q_y^{-1} A)^{-1} A^T Q_y^{-1} \underline{y} \tag{3.4}$$

with its variance-covariance matrix

$$Q_{\hat{x}} = (A^T Q_y^{-1} A)^{-1} \tag{3.5}$$

From Koch (1999) the vector of least squares residuals reads

$$\underline{\hat{e}} = (I - A(A^T Q_y^{-1} A)^{-1} A^T Q_y^{-1}) \underline{y} \qquad (3.6)$$

which reduces to

$$\underline{\hat{e}} = \underline{y} - A\underline{\hat{x}} = \underline{y} - \underline{\hat{y}} \qquad (3.7)$$

and its variance-covariance matrix is given by

$$Q_{\hat{e}} = Q_y - A(A^T Q_y^{-1} A)^{-1} A^T \qquad (3.8)$$

So far only the case of a linear system has been shown. In many applications, of which GNSS positioning is one, the observation equations are not linear. In this case Eq. (3.1) becomes

$$E\{\underline{y}\} = \mathcal{A}(x) \quad ; \quad D\{\underline{y}\} = Q_y \qquad (3.9)$$

where $\mathcal{A}(x)$ is a non-linear function representing the relationship between the observations and the parameters. In order to make the system solvable linearisation is necessary. It is possible to approximate the nonlinear model using a first order Taylor series expansion (Cross 1994, de Jong et al. 2002) of the observation equations, giving

$$\underline{y} = \mathcal{A}(x_0) + \left.\frac{\partial \mathcal{A}}{\partial x}\right|_{x_0} \Delta\underline{x} \qquad (3.10)$$

with approximate values for the vector of parameters, x_0, computed such that

$$\underline{x} = x_0 + \Delta\underline{x} \qquad (3.11)$$

where $\Delta\underline{x}$ is a vector of small corrections to the provisional values. From Tiberius (1998) the least squares model now becomes

$$E\{\Delta\underline{y}\} = \delta_x \mathcal{A}(x_0) \, \Delta x \quad ; \quad D\{\Delta\underline{y}\} = Q_y \qquad (3.12)$$

where

$$\Delta \underline{y} = \underline{y} - \mathcal{A}(x_0)$$

and $\delta_x \mathcal{A}(x_0)$ is a Jacobian matrix corresponding to the design matrix in the linear model, $\mathcal{A}$, with respect to the approximate vector of parameters, x_0. The estimate of the parameters becomes

$$\hat{\underline{x}} = x_0 + \widehat{\Delta \underline{x}} \qquad (3.13)$$

This solution may be iterated, with each successive estimate of $\hat{\underline{x}}$ used as the approximate value in the next iteration, until the estimated correction is sufficiently small that the solution is considered to have converged.

Eq. (3.12) – (3.13) show a correction to the state vector ($\Delta \underline{x}$) is estimated using an a priori correction to the original observations ($\Delta \underline{y}$). Using GNSS processing as an example of a non-linear least squares problem, it is in fact corrections to an a priori estimate of the observations that are used and not the original observations themselves. For this reason $\underline{y}$ shall henceforth be referred to as the vector of observables to avoid confusion with the raw observations. More information on solving non-linear problems can be found in Teunissen (1990b).

3.2.2 Model of condition equations

An alternative representation of the least squares problem is using the model of condition equations. This is a function of only the observables and the unknown parameters have been eliminated (de Jong et al. 2002).

The model with condition equations is given as

$$B^T E\{\underline{y}\} = 0 \quad D\{\underline{y}\} = Q_y \qquad (3.14)$$

where $\underline{y}$ is an $m \times 1$ random vector of observables and B is a $m \times (m - n)$ matrix of condition coefficients (with n equal to the number of eliminated parameters). The estimators for the observables and the residuals, along with their variance-covariance

matrices, are given respectively by

$$\hat{\underline{y}} = (I - Q_yB(B^TQ_yB)^{-1}B^T)\underline{y} \qquad Q_{\hat{y}} = (I - Q_yB(B^TQ_yB)^{-1}B^T)Q_y \qquad (3.15)$$

$$\hat{\underline{e}} = Q_yB(B^TQ_yB)^{-1})B^T\underline{y} \qquad\qquad Q_{\hat{e}} = Q_yB(B^TQ_yB)^{-1})B^TQ_y \qquad (3.16)$$

The model of observation equations and the model of condition are equations are connected through the following relation (Salzmann 1993):

$$B^TA = 0 \qquad (3.17)$$

Although the model of observation equations is not useful in GNSS processing, since estimates of the unknown parameters are not made, it forms part of the derivation of the relationship between the predicted and post-fit residuals given in §3.7.

3.3 Kalman filter equations

The basic model consists of a vector of observables y and a vector of parameters x that are to be estimated, known as the state vector. The measurement model at epoch k, also known as the primary model, is given by

$$\underline{y}_k = A_kx_k + \underline{\varepsilon}_k \qquad (3.18)$$

where the random measurement error is modelled by $\underline{\varepsilon}_k$. At a given epoch k this relates the $m_k \times 1$ vector of observables, y_k, to the $n \times 1$ state vector, x_k, through the $m_k \times n$ design matrix, A_k. This can also be expressed in terms of the mathematical expectation and dispersion as

$$E\{\underline{y}_k\} = A_kx_k \quad ; \quad D\{\underline{y}_k\} = Q_{y_k} \qquad (3.19)$$

The dynamic model, also known as the secondary model, is given by

$$\underline{x}_k = \Phi_{k,k-1}\underline{x}_{k-1} + \underline{w}_k \tag{3.20}$$

where $\Phi_{k,k-1}$ is the state transition matrix and $\underline{w}_k$ is the vector of process noise. The state transition matrix models how the state vector at a given epoch is related to that at the next and the process noise vector is the random error in the dynamic model.

There are two steps in the Kalman filter, namely prediction and filtering. Although not part of the original Kalman filter, the procedure can also be extended to a third smoothing step. There are a number of ways in which the Kalman filter formulae can be derived. Detailed derviations of the Kalman filter equations from a geodetic perspective can be found in Krakiwsky (1975), Salzmann (1993) and Cross (1994).

Prediction

The first step, also referred to in the literature as the time update stage, is based on the dynamic model which specifies how the state vector changes from one epoch to the next, along with the process noise or uncertainty that is placed on this change. The estimate for the predicted state vector follows from Eq. (3.20) and is given by

$$\hat{\underline{x}}_{k|k-1} = \Phi_{k,k-1}\hat{\underline{x}}_{k-1|k-1} \tag{3.21}$$

where $\Phi_{k,k-1}$ is the transition matrix and $\hat{\underline{x}}_{k-1|k-1}$ the state vector at the previous epoch. Since $E\{\underline{w}_k\} = 0$ for all k and and $E\{\underline{w}_k, \underline{w}_l^T\} = 0$ for $k \neq l$ the $\underline{w}_k$ term vanishes (Kalman 1960, Salzmann 1993). The associated variance-covariance matrix is given by

$$Q_{\hat{x}_{k|k-1}} = \Phi_{k,k-1}Q_{\hat{x}_{k-1|k-1}}\Phi_{k,k-1}^T + Q_{w_k} \tag{3.22}$$

where $Q_{\hat{x}_{k|k-1}}$ is the variance-covariance matrix of the predicted state vector and Q_{w_k} the variance-covariance matrix representing the uncertainty in the noise in the dynamic model.

Filtering

The new observables at epoch k are now used to correct the predicted state vector generated as part of the previous step. It is for this reason that the term measurement update is also used in the literature. The updated state vector and its variance-covariance matrix are given by

$$\hat{\underline{x}}_{k|k} = \hat{\underline{x}}_{k|k-1} + K_k(\underline{y}_k - A_k\hat{\underline{x}}_{k|k-1}) \tag{3.23}$$

$$Q_{\hat{x}_{k|k}} = (I - K_kA_k)Q_{\hat{x}_{k|k-1}} \tag{3.24}$$

with

$$K_k = Q_{\hat{x}_{k|k-1}}A_k^T(Q_{y_k} + A_kQ_{\hat{x}_{k|k-1}}A_k^T)^{-1} \tag{3.25}$$

where K_k is the so-called Kalman gain matrix. The term K_k in Eq. (3.23) controls how much influence the observables have in comparison to the predicted state in the updated state vector. This can be illustrated by taking the simplification $A_k = I$. The updated state vector rearranges to

$$\hat{\underline{x}}_{k|k} = K_k\underline{y}_k + (I - K_k)\hat{\underline{x}}_{k|k-1} \tag{3.26}$$

and the Kalman gain matrix simplifies to

$$K_k = Q_{\hat{x}_{k|k-1}}(Q_{y_k} + Q_{\hat{x}_{k|k-1}})^{-1} \tag{3.27}$$

From Eq. (3.26) and (3.27) it can be seen more clearly how the Kalman gain matrix is used to regulate the influence of the observables and the predicted state.

The vector of predicted residuals, also known as the innovation in the literature, is an important tool in identfying model errors (Teunissen 1990c, Salzmann 1993, Tiberius 1998, Teunissen 1998b). It is defined as

$$\underline{v}_k = \underline{y}_k - A_k\hat{\underline{x}}_{k|k-1} \tag{3.28}$$

Its variance-covariance matrix is defined as

$$Q_{v_k} = Q_{y_k} + A_k Q_{x_{k|k-1}} A_k^T \qquad (3.29)$$

In least squares estimation the post-fit residuals are used to identify model errors. Although predicted residuals can be considered weaker than post-fit residuals, they have the advantage that they can be be tested before the filtering stage. Hence any outlying observables are removed before the state vector is updated. The relationship between the predicted residuals and the post-fit residuals can be found in §3.7. The use of the predicted residuals to identify model errors is covered in Chapter 4.

Smoothing

Although not part of the Kalman filter, as described in §3.1, it is possible to use information from later epochs to provide a better estimate of those estimated earlier on by smoothing. There are three types of smoothing problem (Brown and Hwang 1996): fixed-interval smoothing, fixed-point smoothing and fixed-lag smoothing.

A fixed interval smoother is one where the time interval between epochs is fixed. The process consists of backward sweep that works sequentially from the last point to the first. The smoothed state vector and its variance-covariance matrix, after Meditch (1973), is estimated by

$$\hat{\underline{x}}_{k-1|k} = \hat{\underline{x}}_{k-1|k-1} + J_{k-1}(\hat{\underline{x}}_{k|k} - \hat{\underline{x}}_{k|k-1}) \qquad (3.30)$$

$$Q_{\hat{x}_{k-1|k}} = Q_{\hat{x}_{k-1|k-1}} + J_{k-1}(\hat{\underline{x}}_{k|k} - \hat{\underline{x}}_{k|k-1})J_{k-1}^T \qquad (3.31)$$

with

$$J_{k-1} = Q_{\hat{x}_{k-1|k-1}} \Phi_{k,k-1}^T Q_{\hat{x}_{k|k-1}}^{-1} \qquad (3.32)$$

where J is the so-called smoothing gain.

The fixed point smoother is similar to the fixed interval problem except that only one point in the past is of interest. As each filtered solution is obtained the fixed point chosen is updated. An example would be to update the starting value at each epoch as more data is obtained. The smoothing gain is replaced by the product of

all smoothing gain matrices seen in Eq. (3.32) up to the last epoch, hence

$$J_{k-1} = \prod_{i=k}^{j=1} Q_{\hat{x}_{k-1|k-1}} \Phi_{k,k-1}^T Q_{\hat{x}_{k|k-1}}^{-1} \tag{3.33}$$

where k represents the epoch of the fixed point and j each successive epoch at which a measurement is taken.

A fixed lag smoother estimates a smoothed solution at a set interval in the past. An example could be a ship that estimates its position every second subsequently smoothing its estimated position for one minute ago. This gives the advantage that the position can be obtained without waiting too long, but equally being able to benefit from the additional information from the one minute time lag. Brown and Hwang (1996) show the simplest way in which to implement this is to apply the fixed-interval smoother recursively up to and including the lag time, although a more complex algorithm is given in Meditch (1973).

Since smoothing is not applicable to real-time estimation it is beyond the scope of this thesis and as such it will therefore not be covered in further detail. It is important to note that data collected in real-time can be smoothed for analysis after the event.

Summary of the procedure

A flowchart summarising the Kalman filter process is shown in Fig. 3.2. In order to start the prediction and filtering process an initial estimate at $k = 0$ of the state vector (x_0) and its variance-covariance matrix (Q_{x_0}) are required. This is typically a standard least squares estimate. In some cases initial estimates may be required for more than one epoch. For example, if the state vector contains position and velocity parameters and only position observations are available then it is not possible to make an initial estimate of velocity without at least two epochs of data. Eq. (3.21) and (3.22) are then used to find the predicted state $(\hat{\underline{x}}_{k|k-1})$ and its variance-covariance matrix $(Q_{\hat{x}_{k|k-1}})$. At $k = 0$ an initial estimate of $Q_{\hat{x}_{k|k-1}}$ is also required. This should be set to values that reflect the accuracy of the initial estimates. The predicted state and its variance-covariance matrix are then used in Eq. (3.25) along

Initial estimate:

$$x_0 \; ; \; Q_{x_0}$$

$$k = 1$$

Prediction:

$$\hat{\underline{x}}_{k|k-1} = \Phi_{k,k-1}\hat{\underline{x}}_{k-1|k-1}$$
$$Q_{\hat{x}_{k|k-1}} = \Phi_{k,k-1}Q_{\hat{x}_{k-1|k-1}}\Phi_{k,k-1}^T + Q_{w_k}$$

Filtering:

$$K_k = Q_{\hat{x}_{k|k-1}}A_k^T(Q_{y_k} + A_kQ_{\hat{x}_{k|k-1}}A_k^T)^{-1}$$
$$\hat{\underline{x}}_{k|k} = \hat{\underline{x}}_{k|k-1} + K_k(y_k - A_k\hat{\underline{x}}_{k|k-1})$$
$$Q_{\hat{x}_{k|k}} = (I - K_kA_k)Q_{\hat{x}_{k|k-1}}$$

$$\hat{\underline{x}}_{k|k} \; ; \; Q_{\hat{x}_{k|k}}$$

Continue?

Yes

$$k{+}{=}1$$

No

STOP
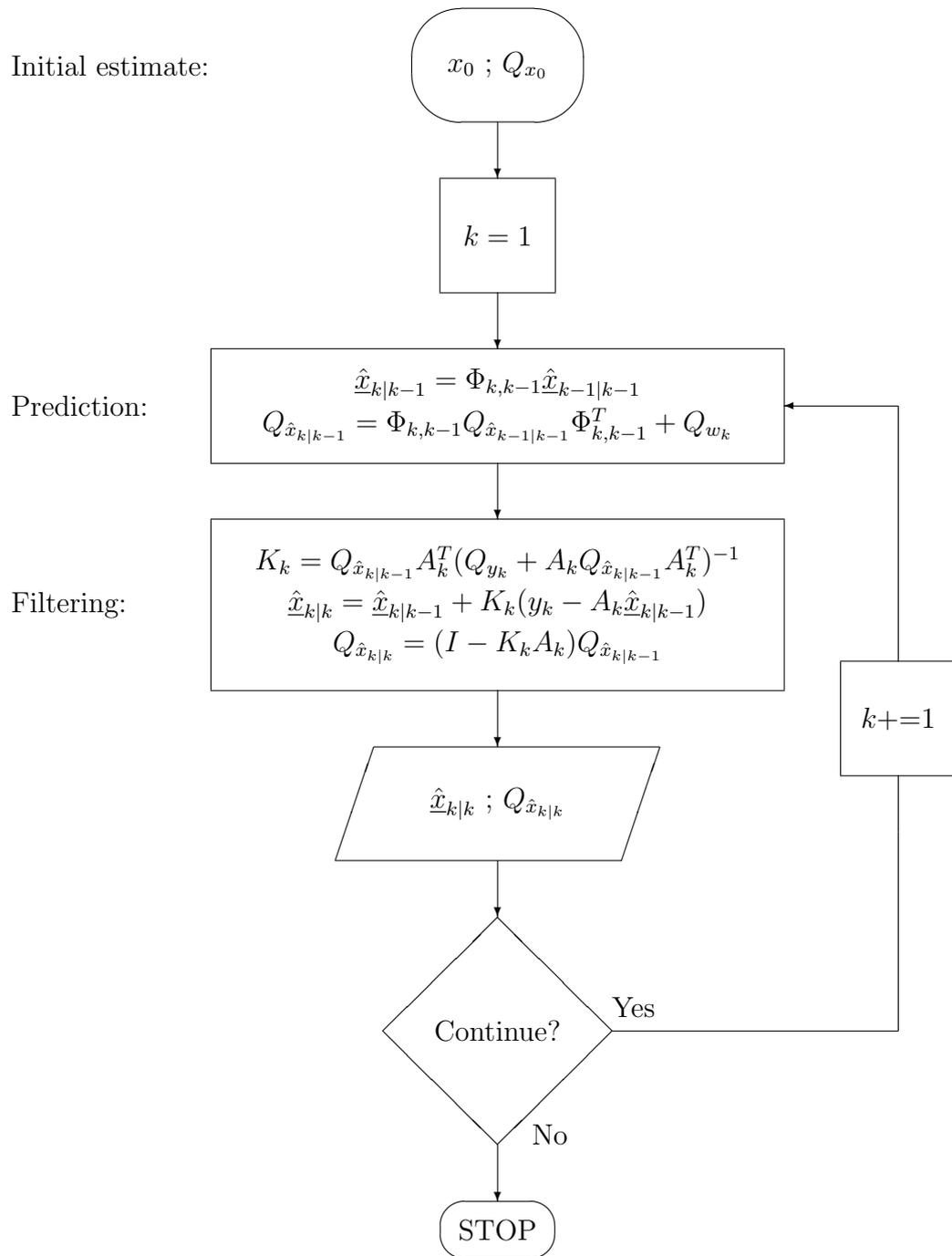
Figure 3.2: Flowchart showing the Kalman filter process

with the variance-covariance matrix of the observed measurements (Q_{y_k}) and the design matrix (A_k) to calculate the Kalman gain, which regulates the contribution of the predicted state and the observed measurements to the updated state. The updated state ($\hat{\underline{x}}_{k|k}$) and its variance-covariance matrix ($Q_{\hat{x}_{k|k}}$) are calculated using Eq. (3.23) and (3.24) respectively.

This filtered solution of the parameters and its variance-covariance matrix becomes the reported solution for that epoch. The process then continues, using the updated state vector and its variance-covariance matrix at the next epoch ($k+1$).

3.4 Nonlinear filtering

The method outlined in §3.3 only applies to linear models and it has already been mentioned in §3.2 that many estimation problems are non-linear. As with least squares, one such solution to this problem is through linearisation. Where this method is applied to the Kalman filter it is known as the extended Kalman filter.

In the case where the dynamic and measurement models are nonlinear it is shown in Gibbs (2011) that Eqs. (3.20) and (3.19) become respectively

$$\underline{x}_k = \varphi(\underline{x}_{k-1}, \underline{w}_k) \tag{3.34}$$

$$\underline{y}_k = a(x_k, \underline{e}_k) \tag{3.35}$$

where φ and a are non-linear functions of the arguments.

The linearised Kalman filter then utilises the same procedure as the standard Kalman filter with the exception that the state transition matrix and design matrix respectively are given by

$$\Phi_{k,k-1} = \left. \frac{\partial \varphi(x_{k-1|k-1}, w_k)}{\partial x} \right|_{k-1} \tag{3.36}$$

$$A_k = \left. \frac{\partial a(x_{k|k-1})}{\partial x} \right|_{k} \tag{3.37}$$

The iterated extended Kalman filter takes the above method and extends it in order to address the problem of bias and divergence that is caused by the linearisation

approximation. It is noteworthy to point that although linearisation is required for the PPP observation equations, where parameters cannot be reliably predicted from one epoch to the next the transition matrix is in fact taken to be the identity matrix, i.e. $\Phi_{k,k-1} = I_k$, and hence is always linear. More information on the extended Kalman filter can be found in Strang and Borre (1997) and Gibbs (2011). Other methods of solving the problem of estimating a nonlinear system include the unscented Kalman filter (Julier and Uhlmann 1997) and the particle filter (Carpenter et al. 1999).

3.5 Least squares derivation of the Kalman filter

The quality control procedures outlined in Chapter 4 are based on the least squares estimator. It is possible to derive the Kalman filter equations using the least squares estimator. It will also be shown in §3.6 that the Kalman filter can be adapted so that the state vector is partitioned into those that can be predicted and those that cannot. This can be done using an extension of the equations that are derived in this section.

Recalling Eq. (3.19), the standard least squares model reads

$$E\{\underline{y}_k\} = A_k x_k \quad ; \quad D\{\underline{y}_k\} = Q_{y_k}$$

In each case the $\underline{y}_k$ vector is extended with additional information from the state vector. It is possible to solve this extended model using both the observation equations and condition equations approaches to least squares estimation. Only the former will be shown. Salzmann (1993) provides an in-depth derivation of the Kalman filter equations for a model with condition equations. Following the observation equations approach the extended model is solved by forming the normal equations, yielding an estimator for $\hat{\underline{x}}_{k|k-1}$ and $\hat{\underline{x}}_{k|k}$ at the filtering and prediction stages respectively.

Prediction

In this least squares derivation the prediction step will be not be treated as a least squares problem, but the predicted state vector and its associated variance-covariance matrix, given by Eq. (3.21) and (3.22), are required in the filtering stage as part of a least squares problem and must still be computed.

Filtering

The least squares model for the prediction step consists of two sets of observables, namely the predicted state vector and the observables from the current epoch. As such the predicted state vector can be thought of as being a set of 'pseudo observables' and the new measurements as 'real observables'.

$$E\left\{\begin{pmatrix} \hat{\underline{x}}_{k|k-1} \\ \underline{y}_k \end{pmatrix}\right\} = \begin{pmatrix} I \\ A_k \end{pmatrix} x_k \quad ; \quad \begin{pmatrix} Q_{\hat{\underline{x}}_{k|k-1}} & \\ & Q_{\underline{y}_k} \end{pmatrix} \tag{3.38}$$

Forming the normal equations and solving for x in the overall model yields

$$\hat{\underline{x}}_{k|k} = (Q_{x_{k|k-1}}^{-1} + A_k^T Q_{y_k}^{-1} A_k)^{-1}(Q_{x_{k|k-1}}^{-1} \hat{x}_{k|k-1} + A_k^T Q_{y_k}^{-1} y_k) \tag{3.39}$$

which is mathematically equivalent to Eq. (3.23), as shown by Koch (1999). Similarly, the variance-covariance matrix of the updated state vector is given by

$$Q_{\hat{x}_{k|k}} = (Q_{x_{k|k-1}}^{-1} + A_k^T Q_{y_k}^{-1} A_k)^{-1} \tag{3.40}$$

which is mathematically equivalent to (3.24), as shown by Koch (1999).

The post-fit residual vector hence contains parts corresponding to $\hat{\underline{x}}_{k|k-1}$ and $\underline{y}_k$ respectively.

$$\hat{\underline{e}}_{k|k} = \begin{pmatrix} \hat{\underline{e}}_{x_{k|k-1}} \\ \hat{\underline{e}}_{y_k} \end{pmatrix} = \begin{pmatrix} -Q_{\hat{\underline{x}}_{k|k}} A_k^T Q_{y_k}^{-1} y_k + (I - Q_{\hat{\underline{x}}_{k|k}} Q_{x_{k|k-1}}^{-1}) x_{k|k-1} \\ (I - A_k Q_{\hat{\underline{x}}_{k|k}} A_k^T Q_{y_k}^{-1}) y_k - A_k Q_{\hat{\underline{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} x_{k|k-1} \end{pmatrix} \tag{3.41}$$

Its variance-covariance matrix is given by

$$Q_{\hat{e}_{k|k}} = \begin{pmatrix} Q_{x_{k|k-1}} - Q_{\hat{\underline{x}}_{k|k}} & -Q_{\hat{\underline{x}}_{k|k}} A_k^T \\ -A_k Q_{\hat{\underline{x}}_{k|k}} & Q_{y_k} - A_k Q_{\hat{\underline{x}}_{k|k}} A_k^T \end{pmatrix} \tag{3.42}$$

A derivation of Eq. (3.41) and (3.42) is given in Appendix B.

3.6   Semi-recursive Kalman filter

In §3.3 the Kalman filter was presented as a recursive method for solving a linearised system with a state vector that can vary predictably over time. In some cases as well as containing parameters that can be reliably predicted from one epoch to the next, it may contain those that are extremely difficult or in fact impossible to predict, even if they do vary over time.

The semi-recursive Kalman filter is presented in de Jong (2008b). It is equivalent to the estimation models presented in de Jonge (1998) and Kleijer (2004). Consider the following modification to the $n \times 1$ state vector

$$x_k = \begin{pmatrix} x_{k,a} \\ x_{k,b} \end{pmatrix} \tag{3.43}$$

where $x_{k,a}$ is a $n_a \times 1$ vector containing parameters which can be predicted at the next epoch and a priori information is known and $x_{k,b}$ is a $n_b \times 1$ vector containing parameters where no a priori information is known at each new epoch. These shall be referred to as the *common* and *epoch* parameters respectively. The number of common and epoch parameters should equal the total number of parameters, i.e. $n = n_a + n_b$.

3.6.1   Model equations

In this subsection the theoretical aspects of the semi-recursive Kalman filter will be outlined following the procedure given in de Jong (2008b). Only the prediction and filtering steps will be considered, although the method can be extended to

smoothing. The implications of the semi-recursive approach are covered in §3.6.2.

Prediction

The prediction step only applies to the parameters in x_a, since in the filtering step no a priori information about x_b can be assumed. Therefore, using the same methodology as in §3.2 the predicted state vector becomes

$$\hat{\underline{x}}_{k|k-1,a} = \Phi_{k,k-1}\hat{\underline{x}}_{k-1|k-1,a} \tag{3.44}$$

where $\Phi_{k,k-1}$ is the $n_a \times 1$ state transition matrix.

It therefore follows that the variance-covariance matrix of the predicted state vector is given by

$$Q_{\hat{x}_{k|k-1,a}} = \Phi_{k,k-1}Q_{\hat{x}_{k-1|k-1}}\Phi_{k,k-1}^T + Q_{w_k} \tag{3.45}$$

where Q_{w_k} is the $n_a \times n_a$ variance-covariance matrix of process noise.

Filtering

The filtering step now includes the additional information from the observables, but the vector of parameters to be estimated includes both common and epoch parameters. As such, Eq. (3.38) becomes

$$E\left\{\begin{pmatrix} \hat{\underline{x}}_{k|k-1,a} \\ \underline{y}_k \end{pmatrix}\right\} = \begin{pmatrix} I & 0 \\ A_k & B_k \end{pmatrix}\begin{pmatrix} x_{k,a} \\ x_{k,b} \end{pmatrix} \quad ; \quad \begin{pmatrix} Q_{\hat{x}_{k|k-1,a}} & \\ & Q_{y_k} \end{pmatrix} \tag{3.46}$$

where the design matrix partitioned is into $\begin{pmatrix} A_k & B_k \end{pmatrix}^T$, representing the common and epoch parameters respectively. The estimator for the filtered state vector is therefore given by

$$\hat{x}_{k|k} = \begin{pmatrix} Q_{x_{k|k-1,a}}^{-1} + A_k^T Q_{y_k}^{-1} A_k & A_k^T Q_{y_k}^{-1} B_k \\ B_k^T Q_{y_k}^{-1} A_k & B_k^T Q_{y_k}^{-1} B_k \end{pmatrix}^{-1} \begin{pmatrix} Q_{x_{k|k-1,a}}^{-1}\hat{x}_{k|k-1,a} + A_k^T Q_{y_k}^{-1}\underline{y}_k \\ B_k^T Q_{y_k}^{-1}\underline{y}_k \end{pmatrix} \tag{3.47}$$

and its associated variance-covariance matrix is given by

$$Q_{x_{k|k}} = \begin{pmatrix} Q_{x_{k|k-1,a}}^{-1} + A_k^T Q_{y_k}^{-1} A_k & A_k^T Q_{y_k}^{-1} B_k \\ B_k^T Q_{y_k}^{-1} A_k & B_k^T Q_{y_k}^{-1} B_k \end{pmatrix}^{-1} \tag{3.48}$$

Although only the common parameters have a priori information it can be seen by the fact that there are off-diagonal elements in $Q_{x_{k|k}}^{-1}$ that the additional information also has an influence on the epoch parameters. The full expansion of $\hat{\underline{x}}_{k|k}$ and $Q_{x_{k|k}}^{-1}$ has not been derived but it does not add to the understanding of the method.

3.6.2   Notes on the semi-recursive Kalman filter

The predicted residual

Recalling Eq. (3.28), the predicted residual is given as

$$\underline{v}_k = \underline{y}_k - A_k \hat{\underline{x}}_{k|k-1}$$

For the semi-recursive Kalman filter the predicted state vector is constructed in terms of $\hat{\underline{x}}_{k|k-1,a}$, i.e. the common parameters, and can only be expressed in terms of the partitioned design matrix relating to the common parameters, i.e. A_k. Since this does not include the epoch parameters $\underline{v}_k$ will not be a realistic estimate of the predicted residual. As such model errors must be detected using the post-fit residual. It is shown in §3.7 that the predicted and post-fit residuals are related.

Relationship to the standard Kalman filter

This theoretical derivation may appear complicated, but it can be significantly simplified in practice by constructing $Q_{x_{k|k-1}}^{-1}$ with common and epoch parameters, where the latter take on zero values. This means that the design matrix no longer needs to be partitioned, although parameters should be ordered such that common and epoch parameters are separated. This in practice makes the estimation procedure equivalent to the Kalman filter outlined in §3.3 where the process noise for epoch parameters is said to be infinite.

Non-linear applications

The method described in this section is of a linear filter. The same method outlined in §3.4 may be used for the semi-recursive Kalman filter. Recalling Eq. (3.46), the design matrix is partitioned into $\left(\begin{array}{cc} A_k & B_k \end{array} \right)^T$. In this case both A_k and B_k must be linearised.

3.7 Relating the predicted and post-fit residuals

The least squares residual given by Eq. (3.7) is

$$\hat{\underline{e}} = \underline{y} - A\hat{\underline{x}}$$

When this is applied to the least squares model for the filtered state given by Eq. (3.38),

$$\hat{\underline{e}}_{k|k} = \left(\begin{array}{c} \hat{\underline{x}}_{k|k-1} \\ \underline{y}_k \end{array} \right) - \left(\begin{array}{c} I \\ A_k \end{array} \right) \hat{x}_{k|k} \tag{3.49}$$

$$= \left(\begin{array}{c} \hat{\underline{x}}_{k|k-1} - \hat{\underline{x}}_{k|k} \\ \underline{y}_k - A_k\hat{\underline{x}}_{k|k} \end{array} \right) \tag{3.50}$$

Following on from the work of de Jong (2008b) it follows that Eq. (3.23) can be expanded as

$$\hat{\underline{e}}_{k|k} = \left(\begin{array}{c} -K_k(\underline{y}_k - A_k\hat{\underline{x}}_{k|k-1}) \\ \underline{y}_k - A_k\hat{\underline{x}}_{k|k} \end{array} \right) \tag{3.51}$$

This shows that the residual vector consists of two parts, namely the residuals related to the predicted state vector $(\hat{\underline{x}}_{k|k-1})$ and the residuals related to the observables $(\underline{y}_k)$.

It is possible to partition $\hat{\underline{e}}_{k|k}$ as

$$\hat{\underline{e}}_{k|k} = \left(\begin{array}{c} \hat{\underline{e}}_{\hat{x}_{k|k-1}} \\ \hat{\underline{e}}_{y_k} \end{array} \right) \tag{3.52}$$

and substitute Eq. (3.23) into $\hat{\underline{e}}_{y_k}$ giving

$$
\begin{aligned}
\hat{\underline{e}}_{y_k} &= \underline{y}_k - A_k(\hat{x}_{k|k-1} + K_k(y_k - A_k\hat{x}_{k|k-1})) \\
&= \underline{y}_k - A_k\hat{x}_{k|k-1} - A_kK_k(y_k - A_k\hat{x}_{k|k-1}) \\
&= (I - A_kK_k)(\underline{y}_k - A_k\hat{x}_{k|k-1})
\end{aligned}
\tag{3.53}
$$

With Eq. (3.28) and (3.51),

$$
\begin{aligned}
\hat{\underline{e}}_{k|k} &= \begin{pmatrix} -K_k(y_k - A_k\hat{x}_{k|k-1}) \\ (I - A_kK_k)(\underline{y}_k - A_k\hat{x}_{k|k-1}) \end{pmatrix} \\
&= \begin{pmatrix} -K_kv_k \\ (I - A_kK_k)v_k \end{pmatrix}
\end{aligned}
\tag{3.54}
$$

It is also possible to derive the relationship between the predicted and post-fit residuals by considering the least squares method in terms of condition equations (de Jong et al. 2002), see §3.2.2. The model Eq. (3.14) now reads

$$
\begin{pmatrix} -A_k & I \end{pmatrix} E\left\{ \begin{pmatrix} \hat{\underline{x}}_{k|k-1} \\ \underline{y}_k \end{pmatrix} \right\} = 0
\tag{3.55}
$$

Using Eq. (3.16) this yields

$$
\begin{pmatrix} \hat{\underline{e}}_{\hat{x}_{k|k-1}} \\ \hat{\underline{e}}_{y_k} \end{pmatrix} = \begin{pmatrix} -Q_{x_{k|k-1}}A_k^T(Q_{y_k} + A_kQ_{x_{k|k-1}}A_k^T)^{-1} \\ Q_{y_k}(Q_{y_k} + A_kQ_{x_{k|k-1}}A_k^T)^{-1} \end{pmatrix} (\underline{y}_k - A_k\hat{\underline{x}}_{k|k-1})
\tag{3.56}
$$

where $\hat{\underline{e}}_x$ and $\hat{\underline{e}}_y$ are estimators of the corrections to the model. Since,

$$
\begin{pmatrix} \hat{\underline{x}}_{k|k} \\ \hat{\underline{y}}_k \end{pmatrix} = \begin{pmatrix} \hat{\underline{x}}_{k|k-1} \\ \underline{y}_k \end{pmatrix} + \begin{pmatrix} -\hat{\underline{e}}_{\hat{x}_{k|k-1}} \\ -\hat{\underline{e}}_{y_k} \end{pmatrix}
\tag{3.57}
$$

and recalling Eq. (3.28) and (3.29),

$$
\underline{v}_k = \underline{y}_k - A_k\hat{\underline{x}}_{k|k-1} \quad ; \quad Q_{v_k} = Q_{y_k} + A_kQ_{x_{k|k-1}}A_k^T
$$

it follows that

$$
\begin{pmatrix} \underline{\hat{x}}_{k|k} \\ \underline{\hat{y}}_k \end{pmatrix} = \begin{pmatrix} \underline{\hat{x}}_{k|k-1} \\ \underline{y}_k \end{pmatrix} + \begin{pmatrix} Q_{\hat{x}_{k|k-1}} A_k^T Q_{v_k}^{-1} \underline{v}_k \\ -Q_{\hat{y}_k} Q_{v_k}^{-1} \underline{v}_k \end{pmatrix} \tag{3.58}
$$

This leads to the following relationship:

$$
\underline{\hat{e}}_{y_k} = \underline{y}_k - \underline{\hat{y}}_k = Q_{y_k} Q_{v_k}^{-1} \underline{v}_k \tag{3.59}
$$

It is therefore possible to obtain the post-fit residuals from the predicted residuals. Since Q_{y_k} is invertible it is also possible to do the reverse and obtain the predicted residuals the from post-fit residuals, i.e.

$$
\underline{v}_k = Q_{v_k} Q_{y_k}^{-1} \underline{e}_{y_k} \tag{3.60}
$$

3.8 Applying the Kalman filter to Precise Point Positioning

There exist many PPP implementations which differ in the estimation methods used and the parameters estimated. Several utilise the extended Kalman filter model as outlined in §3.3, such as Kjørsvik and Brøste (2009) and Salazar et al. (2010). The PPP processing software used in this thesis follows the non-linear form of the model given by de Jong (2008b) outlined in §3.6.

State vector

The GNSS absolute positioning models for pseudorange and carrier phase observations are given by Eq. (2.1) - (2.4), which are repeated below for clarity.

$$p_{r,i}^s = R_r^s + \gamma_i I_{r,1}^s + T_r^s \qquad \phi_{r,i}^s = R_r^s - \gamma_i I_r^s + T_r^s$$
$$+ c[\delta t_r - \delta t^s] \qquad\qquad + c[\delta t_r - \delta t^s]$$
$$+ \eta_{r,i} - \eta_i^s \qquad\qquad + \mu'_{r,i} - \mu'^s_i + \lambda_i N_r^s$$
$$+ m_{r,i,p}^s + e_r^k \qquad\qquad + m_{r,i,\phi}^s + \varepsilon_i^k$$

with

$$R_r^s = \sqrt{(X^s - X_r)^2 + (Y^s - Y_r)^2 + (Z^s - Z_r)^2}$$

In order to estimate the receiver coordinates a number of the variables can be eliminated from the equations. As discussed in §2.2.1 it is possible to remove I_r^s by eliminating the effect of the ionosphere using an ionosphere-free linear combination (see §2.2.1). This is the approach taken in many PPP implementations (Kouba and Héroux 2001). In the PPP processing software used in this thesis the ionosphere is estimated as a parameter that changes from epoch-to-epoch, which is mathematically equivalent to the ionosphere free approach (Wells et al. 1987). In the case where there are more than two frequencies available no choice needs to be made about the type of ionospheric free linear combination that is to be used.

One of the signature features of PPP is that the precise orbit and satellite clock values, as outlined in §2.3, are used to constrain the solution and hence provide known values for (X^s, Y^s, Z^s) and δt^s. It is important to remember that these values are not perfect, even in the case of IGS Final products. It is therefore necessary to consider residual orbit and clock biases in the estimation model, although the effect of these can be partly absorbed by another parameter through reparameterisation. The advantage of this is that it reduces the number of parameters that must be solved, although this could be at the expense of a small bias in the estimated parameters.

The remaining values are separated into those that can be reliably predicted from

Table 3.1: Summary of estimated parameters in the state vector

Parameter	Type	Count
Position	Epoch	3 (X, Y, Z)
Receiver clock	Epoch	1 per system
Tropospheric delay	Common	1
Ionospheric delay	Epoch	1 per satellite
Code hardware delays	Common	1 per satellite and frequency
Carrier phase hardware delays	Common	1 per satellite and frequency

one epoch to the next and those where a new value is estimated at each epoch with no a priori information, according to the model set out in §3.6. From Eq. (2.1) - (2.4) the values that can be predicted from one epoch to the next are the tropospheric delay, ionospheric delay, hardware delays and carrier ambiguities. As mentioned earlier, the ionospheric delay is estimated on an epoch-by-epoch basis. If the data is being treated as kinematic then it is not known how the receiver coordinates will change from one epoch to the next. The receiver clock error is similarly unknown. As such receiver position, receiver clock bias and ionosphere are treated as values to be estimated at each epoch with no a priori information.

As the model stands in Eq. (2.1) - (2.4) all parameters are not estimable since the model is not of full rank. This can be overcome through reparameterisation of the model. In the case of in the case of PPP navigation it is the receiver position that is of primary interest, hence parameters that are less important can be combined with other parameters in order to reduce the number of unknowns in the equation and remove any singularities. An alternative to this is the selection of a reference satellite for a given parameter. Using this approach the parameter for the reference satellite is subtracted from each observation equation. This therefore eliminates the parameter for the reference satellite, whilst the parameter for each other satellite becomes relative to the reference satellite. More on reparameterisation through selection of a reference satellite, sometimes referred to as the 'S-basis', can be found in de Jonge (1998). Using these two approaches it is possible to remove any singularities from the design matrix, whilst still ensuring that the observation equations still hold true. These reparameterisations can be unique to each implementation of the PPP technique.

Following a reparameterisation the state vector contains the parameters as laid out in Table 3.1. Since the semi-recursive Kalman filter (see §3.6) is used the state vector is partitioned into common and epoch parameters, as specified by the second column of Table 3.1. The ambiguity parameter is contained within the carrier phase hardware delays and is modelled as a float parameter, since zero-difference biases are not integer (Blewitt 1989). It is possible to fix biases in PPP using uncalibrated phase delays generated by a reference network, which form double-difference ambiguites and hence allow fixing to integers. A variety of methods of doing this can be found in Wuebbena et al. (2005), Laurichesse and Mercier (2007) and Ge et al. (2008). This method is not considered in this thesis.

Dynamic model

From §3.6 it is known that the dynamic model for the semi-recursive Kalman filter consists of a transition matrix, $\Phi_{k,k-1}$, and a variance-covariance matrix of the process noise, Q_{w_k}, for the common parameters of the state vector. The transition matrix in the software used in this study is given as

$$\Phi_{k,k-1,a} = I_k \tag{3.61}$$

and hence,

$$\hat{\underline{x}}_{k|k-1,a} = \hat{\underline{x}}_{k-1|k-1,a} \tag{3.62}$$

Similarly, $Q_{w_{k,a}}$ is given as

$$Q_{w_{k,a}} = \text{diag}\left(\sigma^2_{w_{k,a,1}}, \ldots, \sigma^2_{w_{k,a,n}}\right) \tag{3.63}$$

where $\sigma^2_{w_{k,a,n}} = q \times \Delta t$, with a spectral density of q and time difference Δt between epochs. In comparison with specifying the measurement noise, the covariance-matrix of the process noise is relatively difficult as the process is not directly measurable. It is often a case of giving enough uncertainty to the parameters to allow enough room to move. The process of adjusting the spectral densities until the desired result is achieved is commonly referred to as 'tuning'. Extensive testing to find the

optimum process noise values for offshore positioning scenarios was carried out prior to this study. In some cases self-tuning Kalman filters have been developed where knowledge about how the parameters change over time is not known (Hu et al. 2003, Mohamed and Schwarz 1999, Yang and Gao 2005).

Measurement model

As outlined in Eq. (3.19) the measurement model at epoch k consists of the vector of observables, $\underline{y}_k$, its variance-covariance matrix Q_{y_k} and the design matrix, A_k. PPP utilises both pseudorange and carrier phase observations, as discussed in §2.2. Since the ionosphere is being estimated as a parameter all available frequencies are used. In the case where L1 and L2 observations are available, the vector of observables consists of four observations for each satellite, such that

$$\underline{y}_k = \left(\begin{array}{cccccccc} p_{r,1}^1 & \phi_{r,1}^1 & p_{r,2}^1 & \phi_{r,2}^1 & \cdots & p_{r,1}^n & \phi_{r,1}^n & p_{r,2}^n & \phi_{r,2}^n \end{array} \right)^T \qquad (3.64)$$

where n is the number of satellites observed. Since the size of the vector of observables is dependent on the number of satellites observed, it can vary between epochs as satellites come in and drop out of view of the receiver's antenna.

The variance-covariance matrix contains the a priori variance of each observable. The standard deviation of each observable is calculated using

$$\sigma(E) = A_0 + A_1 \exp\left(\frac{-E}{E_0}\right) \qquad (3.65)$$

where E is the elevation in degrees, $\{A_0, A_1\}$ are coefficients measured in metres and E_0 is a coefficient measured in degrees (Xiang Jin and de Jong 1996). For code measurements $A_0 \approx 1.0$m and carrier phase measurements $A_0 \approx 0.004$m. The coefficients A_1 and E_0 are receiver and site specific, hence can vary between different processing scenarios.

It is assumed that different observation types on different frequencies are uncorrelated, i.e. the variance-covariance matrix is diagonal. The assumption is also made that the errors in the observables are uncorrelated in time. In reality neither

of these assumptions are strictly true, but these assumptions are commonly made in GNSS processing software. As correlations between observation types on different frequencies are receiver dependent (Bona 2000) it is impractical to attempt to model these covariances in the stochastic model.

As the design matrix is intrinsically linked to the vector of observables the size of the matrix is dependent on the number of observables and hence, the number of satellites in view. Similarly the number of parameters that will need to be estimated will also be dependent on the number of satellites that are in view. Consequently the overall dimensions of the design matrix are a product of the size of the observable and state vectors, which are themselves a product of the number of satellites in view, and therefore are also liable to change from one epoch to the next.

Notes on implementation

Since this thesis is not primarily concerned with the design of PPP processing software the implementation aspects of such will not be covered in detail. More information on the parameterisation used the structure of the design matrix can be found in de Jong (2008a). However, a number of implementation aspects of particular interest will be addressed below.

It has already been discussed in the previous section that the size of the state vector is dependent on the number of observed satellites. Should a satellite come into view then it is necessary to extend the state vector for the number of parameters that this entails. As such, the parameters need to be accounted for in the predicted state vector, $\hat{\underline{x}}_{k|k-1}$, and its associated variance-covariance matrix, $Q_{\hat{x}_{k|k-1}}$, during the prediction step. At the filtering step the vector of observables also contains the additional observables from the new satellite and hence its variance-covariance matrix, Q_{y_k}, must also be extended to include the additional observables. Finally, the design matrix, A_k, must also be extended to account for both the additional parameters in the state vector and the measurements in the vector of observables. Similarly, should a satellite no longer be in view or an observable removed from the model then the relevant parameters in the state vector and rows and columns in the

design matrix and variance-covariance matrices should also be removed.

As mentioned at the beginning of this section, the GNSS measurement model is non-linear. The design matrix must therefore be linearised, as described in §3.4.

3.9   Summary

The theory of the Kalman filter as a recursive estimator based on prediction and filtering was introduced at the beginning of this chapter. It is ideal for PPP given it can be used to provide the best estimate based on all information available whilst still remaining computationally efficient. The model equations for each step were presented and the least squares derivations for the steps relevant to real-time processing were given. These derivations can be used to derive the semi-recursive Kalman filter presented in de Jong (2008b), which separates those parameters that can be predicted from one epoch to the next from those that must be estimated at each epoch. The implication of such a system is that the predicted residuals can no longer be used for hypothesis testing and as such the post-fit residuals must be used. It was shown that the predicted and post-fit residuals are related and their relationship derived. Although in the case where predicted residuals are available quality control can be performed before the measurement update, post-fit residuals can also be used for additional quality control. Finally, the more specific case of a recursive filter for PPP processing was covered along with some of the implementation issues.

Chapter 4

Quality control procedures for GNSS

4.1 Introduction

In Chapter 3 the least squares estimator and Kalman filter were presented as two methods of estimating a vector of unknown parameters from a set of observables. Both of these models work providing the stochastic and observation models used hold true. Should a model misspecification occur then this will lead to an error in the estimated results. In the case of a recursive estimator, such as a Kalman filter, should a model misspecification occur at one epoch then this will be carried on to successive epochs. The residuals produced from the estimation procedure are used with statistical testing to examine whether a model misspecification has occurred. A bias in a GNSS observable results in a misspecification in the observation model, which shall be referred to as a model error. Generally for code observations this is in the form of an outlier, where the bias occurs at one epoch, and for a carrier phase observation it is in the form of a slip, where the bias persists for all subsequent epochs. The source of the model misspecification is found using the residuals and it is removed. In parallel to this, it is also important to monitor the size of bias that can be detected and the potential impact it can have on the estimated parameters. Together these processes form the quality control procedure.

In this chapter hypothesis testing is introduced to investigate the observation

model against outliers in the code observations and cycle slips in the carrier phase. The T test statistic (Baarda 1968, Teunissen 1985) is presented as the general test statistic used for testing for model errors, from which various forms can be derived. These make up the detection, adaptation and identification (DIA) procedure first presented by Baarda (1968). Since it has been shown in Chapter 3 that testing using the Kalman filter can be undertaken using the predicted residual, but testing for the semi-recursive Kalman filter presented in §3.6 must be done using the post-fit residual, both the batch and recursive quality control procedures are considered.

The minimal detectable bias (MDB) and the minimal detectable effect (MDE) are given as measures of reliability for GNSS. These convey the smallest possible error that can be detected by the test statistics in terms of observables and parameters respectively. These will also be derived and their relationship to the test statistics shown.

In this chapter a number of references to probability distributions are made. A normal distribution with mean μ and variance σ^2 is written as $N(\mu, \sigma)$, a non-central χ^2-distribution with d degrees of freedom and a non-centrality parameter of λ is written as $\chi^2(d, \lambda)$ and a non-central F-distribution with degrees of freedom d_1, d_2 and non-centrality parameter λ is written as $F(d_1, d_2, \lambda)$. In each the level of significance which is being tested against is given as a subscript, e.g. $F_{\alpha_{m-n}}(m - n, \infty, 0)$.

4.2 Hypothesis testing

4.2.1 What is hypothesis testing?

A key statistical tool, hypothesis testing is used to identify if a model error has occurred. This is done by forming a null hypothesis, commonly referred to as H_0, and one or more alternative hypotheses, referred to collectively as H_A. These hypotheses refer to the population as a whole, but in practice testing can only be done on a sample of the population and as such the process of hypothesis testing is not infallible. Should an error occur these are divided into Type I and Type II errors,

which are defined in Table 4.1.

Table 4.1: Types of error in hypothesis testing

	H_0 true	H_0 false
H_0 accepted	No error	Type II error
H_0 rejected	Type I error	No error

The probability of a Type I error occurring is defined by the false positive rate, usually referred to as the level of significance, α. The probability of a Type II error occurring is defined by the false negative rate, β, or more commonly by the power of test, defined as $\gamma = 1 - \beta$ (Teunissen 1998a).

4.2.2 Modelling errors

A model error can be defined more accurately as an error in either the measurement or stochastic model. Incorrectly modelling the relationship between the observables and the parameters through the design matrix is an error in the measurement model. For GNSS observations this can occur because of either an outlier in a code observation or a cycle slip in a carrier phase measurement. Both of these model errors are biases, but they differ in character. When a slip occurs the bias that is introduced persists for all of the proceeding epochs. An outlier may occur at one or more epochs, but the bias by contrast is not maintained. In the case of a single-epoch outlier the observables may be similar to those before the bias, whereas for a ramp fault the bias may increase with size over time. Examples of an outlier and bias are shown diagrammatically in Figure 4.1.

(a) Outlier   (b) Slip
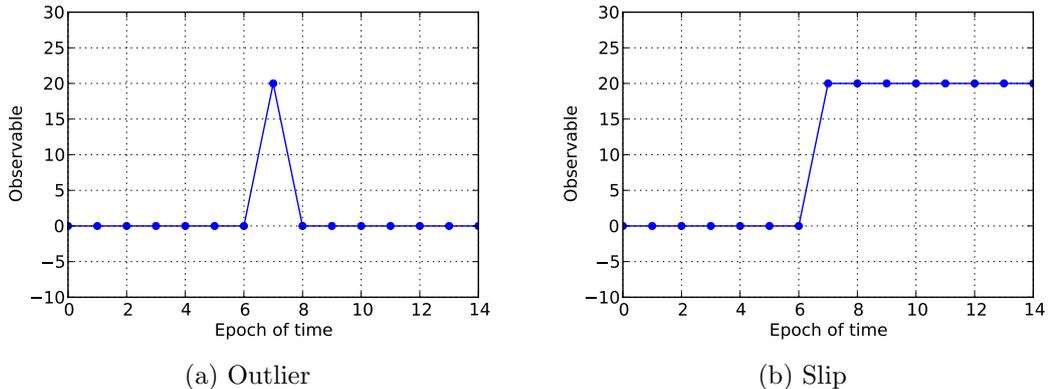
Figure 4.1: Types of observation model error.

The stochastic model relates the precision of the observables themselves. In least squares estimation it is the inverse of the variance-covariance matrix of the observables, Q_y, which is used to weight the observables. For GNSS processing the precision of the observations is fairly well known. Since PPP uses both code and carrier phase observations it is important to get the correct ratio of precision between the two observation types.

The model in Eq. (3.1) can be parameterised in the presence of an unknown model error as

$$E\{\underline{y}\} = Ax + \nabla \tag{4.1}$$

where ∇ represents the unmodelled error vector. Assuming there are m observables and n parameters, the error can be parameterised as

$$\nabla = C_y \nabla_y \tag{4.2}$$

where C_y is a known $m \times q$ matrix and ∇_y an unknown $q \times 1$ vector. Hence, Eq. (4.1) becomes

$$E\{\underline{y}\} = \begin{pmatrix} A & C_y \end{pmatrix} \begin{pmatrix} x \\ \nabla_y \end{pmatrix} \tag{4.3}$$

In order for there to be enough observables for the extra parameters, $q + n \leq m$. In ∇_y there are q explanatory parameters to account for the model error. The number q is known as the dimension of the test (Teunissen 1998b). It is also clear that the matrix C_y must be specified beforehand. This comes from knowledge of the expected errors, which for GNSS processing may be a bias in a code observation or a cycle slip in a phase observation. A number of examples for matrix C_y can be found in Teunissen (1998b).

4.2.3 Testing for model errors

When testing for model errors the null and alternative hypotheses must first be specified. In this section only the model of observation equations will be considered, since the model of condition equations cannot be used for parameter estimation.

Recalling Eq. (3.1) and Eq. (4.3), the null and alternative hypotheses become:

$$H_0 : E\{\underline{y}\} = Ax \qquad H_A : E\{\underline{y}\} = Ax + C_y \nabla_y \qquad (4.4)$$

In order to better understand the concepts in testing for modelling errors it is helpful to visualise the least squares model as a geometric problem. It is has been shown in Chapter 3 that $\hat{y}$ is an estimate of the random variable $\underline{y}$ using the design matrix A. Teunissen (2006) shows that the estimate $\hat{y}$ is an orthogonal projection of $\underline{y}$ on to the range space of A, denoted as $R(A)$. This is shown geometrically in Figure 4.2a, where $\underline{y}$ is a vector representing the true observables and $\hat{y}$ is a vector representing

(a) One estimate ($\hat{y}$)   (b) Three estimates ($\hat{y}_0, \hat{y}_1, \hat{y}_3$)
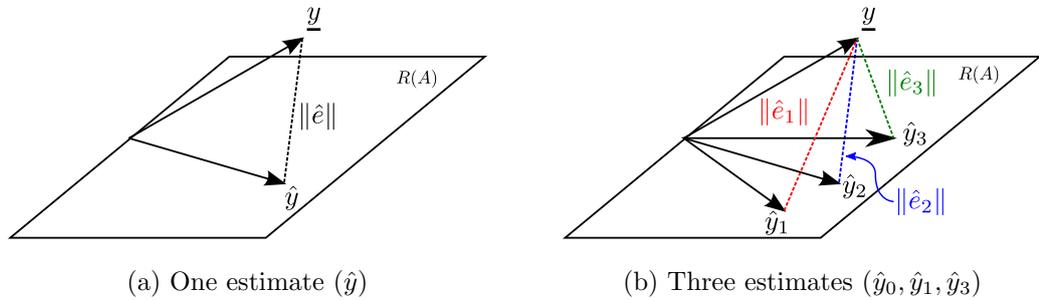
Figure 4.2: Geometric representation of least squares problem.

the estimated observables. On this diagram the distance between the true and the estimated observables is in fact the least squares residual, $\hat{e}$. It is possible to calculate any number of estimates for the observables, but there is only set of 'true' observables, as shown in Figure 4.2, where $\{\hat{y}_0, \hat{y}_1, \hat{y}_3\}$ are three different estimates of the observables. However, it can be seen that the distance between the true and estimated observables is different in each case. The least squares solution arises from the case where this distance is the smallest.

Using this form of geometrical representation the null and alternative hypotheses given by Eq. (4.4) are shown in Figure 4.3. In this case $\hat{y}_A$ is the orthogonal projection of $\underline{y}$ on to the range space of $R(A \vdots C_y)$ (see Eq. 4.3), of which $R(A)$ is a linear subspace, as shown by Teunissen (2006). From this it can be seen that the deviation of the alternative hypothesis from the null hypothesis is not only measured by $(\hat{y}_0 - \hat{y}_A)$, but also $(\|\hat{e}_0\|^2 - \|\hat{e}_A\|^2)^{\frac{1}{2}}$. This is because of the right-angled triangle that is made up by $\underline{y}$, $\hat{y}_0$ and $\hat{y}_A$.
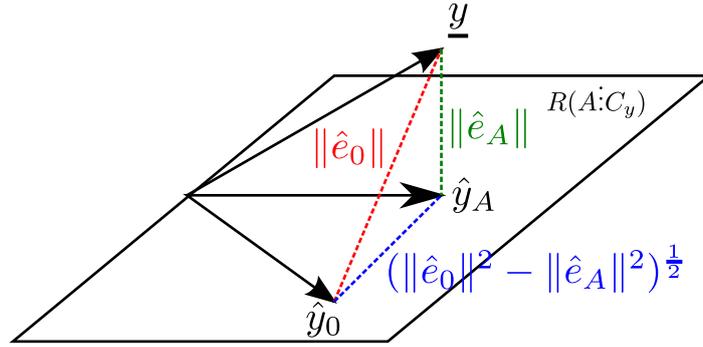
Figure 4.3: Geometric representation of least squares problem with null and alternative hypotheses.

If there is no model error present in the observables all elements of ∇_y will be zero and H_0 will be coincident with H_A. In the presence of a model error in the observables the distance between $\hat{y}_0$ and $\hat{y}_A$ will increase proportionally to the size of the error. The test statistic that arises therefore determines whether $\hat{y}_A$ differs significantly to $\hat{y}_0$, rejecting H_0 if this is the case.

The T test statistic is given by Teunissen (2006) as

$$\underline{T}_q = \hat{\underline{e}}^T Q_y^{-1} C_y (C_y^T Q_y^{-1} Q_{\hat{e}} Q_y^{-1} C_y)^{-1} C_y^T Q_y^{-1} \hat{\underline{e}} \tag{4.5}$$

with the following distribution

$$\begin{aligned} H_0 &: \underline{T}_q \sim \chi^2(q, 0) \\ H_A &: \underline{T}_q \sim \chi^2(q, \lambda) \end{aligned} \tag{4.6}$$

where q is the length of the vector ∇_y and

$$\lambda = \nabla^T C_y^T Q_y^{-1} Q_{\hat{e}} Q_y^{-1} C_y \nabla \tag{4.7}$$

is the non-centrality parameter. A comprehensive derivation of the T test statistic can be found in Teunissen (2006).

4.3   Batch processing

Although the focus of this thesis is on real-time applications, an explanation of the development of the T test statistic for batch processing naturally leads on to its application in recursive processing. Two cases will be considered; the overall model test and the slippage test.

The development of these two test statistics can be approached from opposing directions. Baarda (1968) considers p alternative hypotheses, each of which can be considered analogous to the local slippage test. This is then generalised so that all alternative hypotheses can be considered simultaneously, which leads to the overall model test. By contrast, Teunissen (1998b) starts with the general test statistic given by Eq. (4.5) and derives both the overall model and slippage test accordingly.

4.3.1   Overall model test

Following the derivation by Teunissen (1998b), the dimension q of the test given by Eq. (4.5) is set equal to the degrees of freedom in the model, which is defined by the difference between the m observables and n parameters. The null and alternative hypothesis are in terms of the vector of observables, $\underline{y}$, and hence read

$$
\begin{aligned}
H_0 &\quad : \quad E\{\underline{e}\} = 0 \\
H_A &\quad : \quad E\{\underline{e}\} \in R^{m_k}
\end{aligned}
\tag{4.8}
$$

Teunissen (1998b) shows that as $q = m - n$ the matrix $\left(\begin{array}{ccc} A & \vdots & C_y \end{array} \right)$ is square it follows that

$$
\underline{T}_q = \frac{\underline{e}^T Q_y^{-1} \underline{e}}{m - n}
\tag{4.9}
$$

Since the number of alternative hypotheses is infinite these are left unspecified and only the general case is considered. The test is passed where $T_{m-n} < F_{\alpha_{m-n}}(m - n, \infty, 0)$. Note that the F-test is used in this case as the test statistic has been divided through by $(m-n)$ (Teunissen 1990a). Should the test pass then H_0 can be accepted over all H_A. This can said to be the case as the multi-dimensional and one-dimensional tests are coupled (Baarda 1968, Salzmann 1993). It is however reliant

on the correct alternative hypothesis for the one-dimensional test being specified.

4.3.2 Slippage test

In the case where the overall model test fails then one of the alternative hypotheses should be considered over H_0. By considering only one alternative hypothesis at a time the assumption is made that the model error is one-dimensional (Baarda 1968). In this case the matrix C_y simplifies to a vector, c, and Eq. (4.5) simplifies to

$$\underline{w} = \frac{c^T Q_y^{-1} \hat{\underline{e}}}{\sqrt{c^T Q_y^{-1} Q_{\hat{e}} Q_y^{-1} c}} \tag{4.10}$$

where $\underline{T}_{q=1} = (\underline{w})^2$. This is often referred to as the *w-test*. The assumption that there is only one bias at one given time is known as 'data-snooping' (Baarda 1968).

If the model error is considered to be an outlier in only one of the observables the c vector becomes

$$c = \begin{pmatrix} 0 & \dots & 0 & 1 & 0 & \dots & 0 \end{pmatrix}^T \tag{4.11}$$

where 1 coincides with the observable that is being tested. If the error is modelled by Eq (4.11) and the observables are not correlated, i.e. Q_y is diagonal, then the test statistic for the ith observable is

$$\underline{w}_i = \frac{\hat{\underline{e}}_i}{\sigma_{\hat{e}_i}} \tag{4.12}$$

The null hypothesis that observable i is not the source of the model error is accepted when

$$|\underline{w}_i| < N_{1-\alpha_{m_k}/2}(0,1) \tag{4.13}$$

In practice, the observable where $|\underline{w}_i|$ is the largest and $|\underline{w}_i| \geq N_{1-\alpha_{m_k}/2}(0,1)$ is taken to be the observable containing the model error.

The methodology outlined above only deals with the case of a single bias in the vector of observables. In the case of more than one bias it is assumed that the largest bias will be detected. By repeating the process each outlier will be found in turn

until the overall model test, and by design, the slippage test is passed (Teunissen 1998b). One approach taken has been to extend the existing theory to support multiple outliers (Hewitson and Wang 2006, Baselga 2011). An alternative method is to use robust estimation in place of least squares estimation (Wang and Wang 2007, Knight and Wang 2009). This considers biases at the time of adjustment instead of performing the adjustment and then screening for biases. Therefore, in the case of multiple outliers all biases are taken into account during the adjustment. An in-depth discussion of robust estimation can be found in Rousseeuw and Leroy (1987) and Huber (1981).

4.4   Recursive processing

It has been shown in Chapter 3 that the least squares technique can be extended such that the state vector is propogated from one epoch to the next. The Kalman filter was given as an example of this. Teunissen (1990b) presented the extension of the methodology presented in §4.3 using the vector of predicted residuals, given by Eq. (3.28), to identify model errors. The advantage of this over the post-fit residuals is that a model error can be identified before the measurement update takes place.

The test statistic given in Eq. (4.5) can also be derived with respect to the predicted residuals. It is possible to rearrange Eq. (3.60) to give

$$Q_{v_k}^{-1}\underline{v}_k = Q_{y_k}^{-1}\hat{\underline{e}}_k \qquad\qquad \underline{v}_k^T Q_{v_k}^{-1} = \hat{\underline{e}}_k^T Q_{y_k}^{-1} \qquad (4.14)$$

and taking the expectation of $\hat{e}_k$, using Eq. (3.59) and simplifying the notation of $\hat{e}_{y_k}$ to $\hat{e}_k$,

$$E\{\hat{\underline{e}}_k, \hat{\underline{e}}_k^T\} = Q_{y_k}Q_{v_k}^{-1}E\{\underline{v}_k, \underline{v}_k^T\}Q_{v_k}^{-1}Q_{y_k}$$

$$Q_{\hat{e}_k} = Q_{y_k}Q_{v_k}^{-1}Q_{v_k}Q_{v_k}^{-1}Q_{y_k}$$

$$= Q_{y_k}Q_{v_k}^{-1}Q_{y_k} \qquad (4.15)$$

and hence

$$Q_{v_k}^{-1} = Q_{y_k}^{-1} Q_{\hat{e}_k} Q_{y_k}^{-1} \tag{4.16}$$

Substituting Eq. (4.14) and (4.16) into Eq. (4.5) gives

$$\underline{T}_q = \underline{v}^T Q_v^{-1} C (C^T Q_v^{-1} C)^{-1} C^T Q_v^{-1} \underline{v} \tag{4.17}$$

As shown in §5.2 one of the conditions of an optimal Kalman filter is that the predicted residuals are not correlated in time, such that

$$E\{v_k, v_l^T\} = 0 \tag{4.18}$$

where $k \neq l$. Given this assumption, Q_v becomes block diagonal and the test statistic can now be extended to multiple epochs.

$$\underline{T}_{q,\forall i} = \left[\sum_{i=1}^{k} C_{v_i}^T Q_v^{-1} \underline{v}_i \right]^T \left[\sum_{i=1}^{k} C_{v_i}^T Q_v^{-1} C_{v_i} \right]^{-1} \left[\sum_{i=1}^{k} C_{v_i}^T Q_v^{-1} \underline{v}_i \right] \tag{4.19}$$

where the summation is performed over k epochs and the subscript $\forall i$ shows this is for all epochs up to k.

The very nature of recursive estimation means that more information about the observables is available at each subsequent epoch. It is therefore possible to test a subset of observables up to and including the epoch in question, referred to as a global test. Considering all observables up to the current epoch can be considered the 'most global' case. Testing only one epoch in question, which is usually taken as the current epoch, is referred to as a local test. In some cases a global test is appropriate in detecting a model error that may grow slowly over time. By contrast, a local test can determine if a model error has been introduced at the current epoch and, if necessary, action can be taken immediately.

4.4.1 Local testing

It is important to note that for local testing it is assumed that no model error has occurred before the epoch that is being tested (Teunissen 1998b), which will be

referred to as epoch k. If local testing is used at each successive epoch then the assumption is made that any previous model error will have been detected by local testing at the epoch at which it occurred and subsequently dealt with so that its effect is not propagated to future epochs. This is a valid assumption providing that the statistical testing is operating correctly. If the level of significance and power of test chosen do not accurately reflect reality then this may not be the case and model errors may not be detected.

Assuming that statistical testing is operating normally, taking Eq. (4.19) for just one epoch gives

$$\underline{T}_{q,k} = \underline{v}_k^T Q_{v_k}^{-1} C_{v_k} (C_{v_k}^T Q_{v_k}^{-1} C_{v_k})^{-1} C_{v_k}^T Q_{v_k}^{-1} \underline{v}_k \tag{4.20}$$

The local overall model test utilises the predicted residuals to determine whether a model error has occurred at epoch k, much in the same way as the overall model test in §4.3. The null and alternative hypotheses (Teunissen 1998b) read

$$
\begin{aligned}
H_0 &: E\{\underline{v}_k\} = 0 \\
H_A &: E\{\underline{v}_k\} \in R^{m_k}
\end{aligned}
\tag{4.21}
$$

where m_k is the number of observables at epoch k. Since H_0 is formulated in terms of v_k the dimension of the test is now set as $q = m_k$. The known error matrix C_{v_k} is square and regular as the redundancy is equal to the number of observables. The local overall model test statistic therefore reads

$$\underline{T}_{q,k} = \frac{\underline{v}_k^T Q_{v_k}^{-1} \underline{v}_k}{m_k} \tag{4.22}$$

The null hypothesis is accepted when $\underline{T}_{q,k} < F_{\alpha_{m_k}}(m_k, \infty, 0)$. This is almost the same critical value used for batch processing, except that the number of degrees of freedom used for the F-test is set to m_k.

As in §4.3 when the local overall model test is rejected a number of suitable alternative hypotheses must be tested in order to identify the source of the model error, through the known matrix C_{v_k}. Similarly, this also reduces to a vector when

a one-dimensional test is considered (i.e. $q = 1$), represented as c_k. Since Q_{v_k} is block diagonal for all epochs the local slippage test becomes

$$\underline{t}_k = \frac{c_k^T Q_{v_k}^{-1} \underline{v}_k}{\sqrt{c_k^T Q_{v_k}^{-1} c_k}} \qquad (4.23)$$

The null hypothesis is accepted when $|\underline{t}_{q=1,k}| < N_{1-\alpha_{m_k}/2}(0,1)$. It can be seen that this closely resembles the slippage test used for batch processing given by Eq. (4.10).

4.4.2   Global testing

It is the block diagonal nature of Q_v that forms the basis of global testing. It can be seen in Eq. (4.19) that the test statistic can be shown as a summation of tests over successive epochs. Testing observables for epochs between l and k, the null and alternative hypotheses become

$$\begin{aligned} H_0 &: & E\{\underline{v}_{l,k}\} = 0 \\ H_A &: & E\{\underline{v}_{l,k}\} = C_{v_{l,k}} \nabla \end{aligned} \qquad (4.24)$$

where $\underline{v}_{l,k} = \begin{pmatrix} \underline{v}_l^T & \cdots & \underline{v}_k^T \end{pmatrix}^T$.

In comparison to the local test, the dimension of the test for the global case is a summation of all $m_{l,k}$, i.e. $q = \sum_{i=l}^{k} m_i$. Once again as the redundancy is equal to the number observables, $C_{v_{l,k}}$ is square and therefore

$$\underline{T}_{q,l,k} = \frac{\sum_{i=l}^{k} \underline{v}_i^T Q_{v_i}^{-1} \underline{v}_i}{\sum_{i=l}^{k} m_i} \qquad (4.25)$$

As seen with the local overall model test statistic, the null hypothesis is accepted when $\underline{T}_{q,l,k} < F_\alpha(\sum_{i=l}^{k} m_i, \infty, 0)$. It is clear when comparing Eq. (4.22) and (4.25) that the latter is the summation over values $\{l, \ldots, k\}$ of the numerator and denominator of the former. It is also interesting to note that if $l = k$ then the global test reduces to the local test.

In order to reach the one-dimensional global slippage test the matrix $C_{v_{l,k}}$ reduces

to the vector $c_{l,k}$. Like the global overall model, based on the assumption that Q_v is block diagonal, the global slippage test becomes

$$\underline{t}_{q=1,l,k} = \frac{\sum_{i=l}^{k} c_i^T Q_{v_i}^{-1} \underline{v}_i}{\sqrt{\sum_{i=l}^{k} c_i^T Q_{v_i}^{-1} \hat{\underline{c}}_i}} \qquad (4.26)$$

Similarly, the null hypothesis is accepted when $|\underline{t}_{q=1,k,l}| < N_{1-\frac{1}{2}\alpha}(0,1)$.

The advantage of using global testing is that any model error that may remain undetected by the local tests may become apparent further on with the use of the global tests. However, this comes at the cost of having to store historic information on each observable. It is possible to overcome this problem to a certain extent by recursively computing both the global overall model and global slippage test statistics, more information on which can be found in Teunissen (1998b).

4.5 DIA procedure

The test statistics given in §4.3 and §4.4 are presented by Teunissen (1990c) as a quality control procedure that can be readily implemented in GNSS processing software. It consists of three stages: detection, identification and adaptation. For this reason it is known as the DIA procedure and can be applied to both batch and recursive solutions using the appropriate test statistics (Teunissen 1998a).

Detection. The detection stage uses the overall model test to determine whether a model error has occurred. Eq. (4.9) is chosen if a batch solution is tested. For a recursive solution either a local or global procedure must be chosen, using Eq. (4.22) or Eq. (4.25) respectively. This is tested against the F-test using a chosen level of significance.

Identification. If a model error is detected during the identification stage an alternative hypothesis corresponding to each observable is formed that the bias is contained in that observable. This is in accordance with Baarda's data snooping principle. The slippage test statistic is then used as a dimensionless measure to see which alternative hypothesis is considered most likely. The observable that has the

highest slippage value and exceeds the critical value is considered to contain the model error. As Baarda (1968) makes the assumption that there is only one bias in the data the observable with the highest test statistic value is taken and the process is repeated until no more model errors are said to exist.

Adaptation. As an extension to the test statistics Teunissen (1998a) presents the adaptation stage to make the quality control procedure complete. One approach is to repeat the measurement that was in error, but in all types of GNSS processing this is not possible since the observables can only be measured once. The method proposed is to attempt to add an additional parameter that estimates the error in the model. In the case of an outlier in a code observation the bias at the epoch of the observable is estimated. A cycle slip in a carrier phase measurement results in a bias that applies to all successive epochs. These two situations are resolved in the software used in this study by removing the outlying observable for both code and carrier phase observations. For the latter the assumption is made that this is a cycle slip and the parameter containing the ambiguity is reset.

The general DIA procedure for batch processing is presented in Fig 4.4 and the procedure for recursive solutions in Fig 4.5.

4.6   Application in PPP processing

As PPP processing requires a recursive estimator the procedure outlined in §4.4 naturally lends itself to PPP processing. It can be seen that the recursive methodology is a natural progression of the batch procedure with the main difference being that the predicted residuals ($\underline{v}_k$) are used in place of the post-fit residuals ($\hat{\underline{e}}$).

Cycle slips have a significant influence on the PPP solution and using the procedure outlined in this chapter if such a bias is detected then the parameter must be reset. This is recognised by Banville and Langley (2009), who propose an instantaneous cycle slip correction method. Another approach taken by Jokinen et al. (2012) is to calculate a separate test statistic for the code and carrier phase observations in addition to testing all observations at once.

Since the PPP processing software used in this study uses the semi-recursive

Least squares adjustment
$$\hat{\underline{x}} = (A^T Q_y^{-1} A)^{-1} A^T Q_y^{-1} \underline{y}$$
$$\hat{\underline{e}} = \underline{y} - A\hat{\underline{x}}$$
$$Q_{\hat{e}} = Q_y - A(A^T Q_y^{-1} A)A^T$$

Calculate overall model test
$$\underline{T}_q = \frac{\hat{\underline{e}}^T Q_y^{-1} \hat{\underline{e}}}{q}$$

Does the test pass?
$$\underline{T}_q < F_{\alpha_{m-n}}(m - n, \infty, 0)$$

Yes

No

Calculate the slippage test
$$\underline{w}_i = \frac{\hat{\underline{e}}_i}{\sigma_{\hat{e}_i}}$$
for observations $i = 1 \dots m$

Remove observation i where
$$\arg\max_{0 < i < m} |\underline{w}_i| \text{ and } \underline{w}_i > N_{1 - \frac{1}{2}\alpha}$$

Accept solution
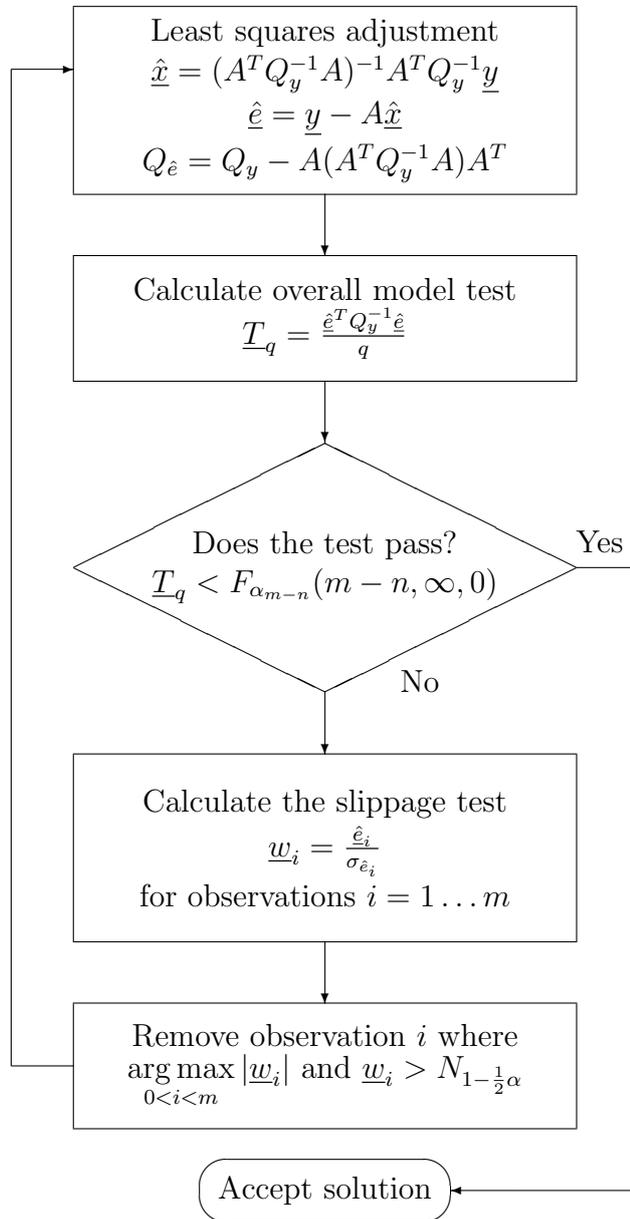
Figure 4.4: Quality control procedure for batch processing.

$$\textit{Prediction}$$
$$\underline{\hat{x}}_{k|k-1} = \Phi_{k,k-1}\underline{\hat{x}}_{k-1|k-1}$$
$$Q_{\hat{x}_{k|k-1}} = \Phi_{k,k-1}Q_{\hat{x}_{k-1|k-1}}\Phi_{k,k-1}^T + Q_{w_k}$$

Calculate predicted residual and
its variance-covariance matrix
$$\underline{\hat{v}}_k = \underline{y}_k - A_k\underline{\hat{x}}_{k|k-1}$$
$$Q_{\hat{v}_k} = Q_{y_k} + A_kQ_{x_{k|k-1}}A_k^T$$

Calculate local overall model test
$$\underline{T}_{q,k} = \frac{\underline{v}_k^T Q_v^{-1} \underline{v}_k}{m_k}$$

Does the test pass?
$$\underline{T}_{q,k} < F_{\alpha_{m_k}}(m_k, \infty, 0)$$

Yes

No

Calculate the local slippage test
$$\underline{t}_{q=1,k} = \frac{c_k^T Q_{v_k}^{-1} \underline{\hat{v}}_k}{\sqrt{c_k^T Q_{v_k}^{-1} c_k}}$$
for observations $i = 1 \ldots m$

Remove observation i where
$$\underset{0<i<m}{\arg\max} |\underline{t}_i| \text{ and } |\underline{t}_{q=1,k,l}| > N_{1-\frac{1}{2}\alpha}(0,1)$$

$$\textit{Filtering}$$
$$K_k = Q_{\hat{x}_{k|k-1}}A_k^T(Q_{y_k} + A_kQ_{\hat{x}_{k|k-1}}A_k^T)^{-1}$$
$$\underline{\hat{x}}_{k|k} = \underline{\hat{x}}_{k|k-1} + K_k(\underline{y}_k - A_k\underline{\hat{x}}_{k|k-1})$$
$$Q_{\hat{x}_{k|k}} = (I - K_kA_k)Q_{\hat{x}_{k|k-1}}$$

Figure 4.5: Quality control procedure for recursive processing.

Kalman filter it is not possible to use the predicted residuals for hypothesis testing (see §3.6). In this case the post-fit residual at epoch k must be used. It has been shown in §3.7 that there is a linear relationship between $\underline{\hat{e}}_k$ and $\underline{v}_k$, therefore it follows that the batch procedure can be used in recursive applications for local testing as shown in Appendix B. Note the subscript in the former, denoting the post-fit residual at epoch k. The main difference is that in using the recursive procedure the quality control is performed prior to the measurement update $\underline{v}_k$, whereas using the batch procedure it is performed afterwards using $\underline{\hat{e}}_k$. Since the post-fit residual contains the additional information that the updated state vector brings then the statistical test can be seen as stronger. This is the approach that is recommended by Bisnath and Gao (2008).

4.7   Reliability

Although the tests outlined in §4.3 and §4.4 aim to identify any model error that occurs, the degree to which they can do this is limited by the accuracy of the tests themselves. Just as the precision of the estimated parameters must be considered, so must the reliability of the test statistics.

When quantifying the reliability of a test statistic the statistic can be given in either observation space or parameter space. The former gives the size of the bias that can just be detected given a certain power and level of significance and is known as *internal reliability*. The effect of this can then be translated to show how the size of the error impacts on the parameters themselves, which is described as *external reliability*. This is especially important for navigation applications as it can give an indication of the potential size of a model error on the position in addition to the precision of the position itself.

4.7.1   Internal reliability

For the batch processing case the alternative hypothesis given in Eq. (4.6) is characterised by a non-central F-distribution, where this non-centrality is defined as

$$\lambda = \nabla^T C^T Q_y^{-1} Q_{\hat{e}} Q_y^{-1} C \nabla \tag{4.27}$$

It has already been shown in §4.2 that the probability of a Type II error occurring, that is the null hypothesis being incorrectly accepted, is defined by the power of the test, γ. In order to calculate a measure of reliability instead of requiring the probability of an error of a given magnitude occurring it is the inverse that is required, namely to determine the size of an error at a given probability.

Since the non-centrality of the F-distribution is a product of the level of significance α, the dimension of the test q and the power γ it can be expressed as $\lambda = \lambda(\alpha, q, \gamma)$ and hence Eq. (4.27) becomes

$$\lambda(\alpha, q, \gamma) = \nabla^T C^T Q_y^{-1} Q_{\hat{e}} Q_y^{-1} C \nabla \tag{4.28}$$

As it is ∇ that is of interest, it is shown in Teunissen (1998a) that when $q = 1$, corresponding to the one-dimensional slippage test, then Eq. (4.28) rearranges to give the minimal detectable bias (MDB),

$$|\nabla| = \sqrt{\frac{\lambda(\alpha, q, \gamma)}{c^T Q_y^{-1} Q_{\hat{e}} Q_y^{-1} c}} \tag{4.29}$$

In some texts this is referred to as the marginally detectable error (Cross 1994). The MDB is a measure of the smallest model misspecification error that the one-dimensional T test statistic can measure. In much the same way that Eq. (4.10) can be simplified when using Baarda's data snooping method, the equivalent MDB for each alternative hypothesis reduces to

$$|\nabla_i| = \sigma_{y_i}^2 \sqrt{\frac{\lambda(\alpha, q, \gamma)}{\sigma_{e_i}^2}} \tag{4.30}$$

where $\sigma_{y_i}^2$ is the a priori variance of the ith observable and $\sigma_{\hat{y}_i}^2$ is the a posteriori variance of the ith observable.

For recursive solutions the MDB is calculated with respect to $\underline{v}_k$ and hence is derived from Eq. (4.17). Taking this into account, the MDB corresponding to the local slippage test at time k is defined as

$$|\nabla_k| = \sqrt{\frac{\lambda(\alpha, q, \gamma)}{c_{v_k}^T Q_{v_k}^{-1} c_{v_k}}} \tag{4.31}$$

and the MDB corresponding to the global slippage test for the period $\{l, \ldots, k\}$ is defined as

$$|\nabla_{l,k}| = \sqrt{\frac{\lambda(\alpha, q, \gamma)}{\sum\limits_{i=l}^{k} c_{v_i}^T Q_{v_i}^{-1} c_{v_i}}} \tag{4.32}$$

4.7.2   External reliability

As the MDB is a measure of the smallest possible model error in the observables that the slippage test can detect for practical applications this can have little relevance. For example, when navigating it is more useful to know what size such an error would have on your position.

Recalling Eq. (4.2), the effect of ∇ on $\underline{y}$ is given by

$$\nabla = C_y \nabla_y$$

Tiberius (1998) describes the effect of this error on $\hat{\underline{x}}$ as

$$\nabla_{\hat{x}} = (A^T Q_y^{-1} A)^{-1} A^T Q_y^{-1} \nabla_y \tag{4.33}$$

As $\nabla_{\hat{x}}$ is a vector and there is such a vector for each alternative hypothesis this can be difficult to interpret. One alternative is to consider the largest absolute value of $\nabla_{\hat{x}}$, termed the minimal detectable effect (MDE) by de Jong et al. (2002). The minimal detectable effect, which is the effect of a bias on the estimated parameters, should not be confused with the marginally detectable error, which is the estimated size of the bias in an observable. Since both terms have the same acronym all further

uses of MDE shall refer to the minimal detectable effect. It is important to note that the largest MDB may not necessarily result in the largest MDE.

Another alternative is the bias-to-noise ratio (BNR), given by

$$\lambda_x = \sqrt{\nabla_{\hat{x}}^T Q_{\hat{x}}^{-1} \nabla_{\hat{x}}} \qquad (4.34)$$

This scalar is dimensionless and gives a measure of how the biases in the observables influence the estimated parameters (Teunissen 1998b, de Jong et al. 2002).

4.8 Summary

In this chapter the theory of hypothesis testing was introduced as a method for identifying model errors when processing GNSS observations. The T test statistic (Teunissen 1985, Teunissen 1990c, Salzmann 1993, Tiberius 1998, Teunissen 1998b) was presented as the basis of the statistical testing procedure, which itself was used to derive the overall model test for testing the set of observables as a whole and the slippage test for testing individual observables.

Approaches for both batch and recursive processing were presented, although it is important to note that the recursive procedure is dependent on the condition of the Kalman filter that predicted residuals at different epochs are not correlated since the matrix Q_v must be block diagonal in order to derive the local overall model and local slippage test statistics. This assumption is further explored in Chapter 7.

The DIA procedure (Teunissen 1990c, Teunissen 1998b) was shown as a method of forming the test statistics presented in §4.3 and §4.4 into a formal procedure that can be easily followed. The detection step is based on the overall model test and the adaptation stage on the slippage test. The identification step in its simplest form involves the removal of the observable causing the model error.

Finally, the reliability measures for the test statistic were addressed. This consists of internal reliability, quantified by the MDB, and external reliability, which transforms the MDB from observation to parameter space yielding the MDE.

Chapter 5

Time correlation in residuals

5.1 Introduction

It has been shown in Chapter 3 that the underlying assumption of the Kalman filter is that the predicted residuals from one epoch are not correlated with those at any other epoch. This extends to the semi-recursive Kalman filter presented in §3.6. In Chapter 4 it was also demonstrated that the test statistic used for recursive estimation can only be properly derived when the residuals from one epoch are not correlated with those at any other epoch. This applies to both the predicted and post-fit residuals.

This chapter builds on this concept by firstly outlining the concept of filter optimality in §5.2. This is explained using the linear Kalman filter in terms of the predicted residual, the preferred method in most literature. The theory can be extended to the post-fit residuals, and hence to the semi-recursive Kalman filter, given the linear relationship between $\underline{v}_k$ and $\underline{e}_k$ (see §3.7). It is shown that for an optimal linear filter the predicted residuals are not correlated with those at any other epoch, but for a non-optimal linear filter this is no longer the case.

Four high-rate GNSS datasets have been processed using the E-HP/PPP software and the post-fit residuals are used as a basis of the analysis in §5.3 to determine whether the filter can said to be optimal. A variety of methods are used to show that the residuals exhibit varying degrees of time correlation, dependent on the observation type and the environment in which it is collected.

A number of approaches that can be taken to eliminate or reduce the time correlation thereby achieving an optimal filter are discussed in §5.4. One alternative is to determine the optimum process noise values for the prediction step of the Kalman filter by choosing the value that results in the minimum correlation. A second alternative is to estimate and remove the time correlation in the residuals.

5.2   Filter optimality

The Kalman filter was presented in Chapter 3 as a recursive estimator that uses prediction and filtering to obtain an improved estimate over standard least squares in dynamic systems. In order to achieve an optimum result using the Kalman filter the variance-covariance matrices of the process noise, Q_{w_k}, and the measurement noise, Q_{y_k} must be known exactly (Mehra 1970). In practical terms these are usually only an approximation and often are not known at all. In most cases these matrices are adjusted or 'tuned' until the desired result is achieved. In the case of navigation the tuning of these matrices may be dependent on where and when the filter is being used.

It is assumed for a Kalman filter that $E\{\underline{v}_k, \underline{v}_l^T\} = 0$ where $k \neq l$. In this section it will be shown that for an optimal filter this is the case. This derivation will closely follow that given by Salzmann (1993). Using the same approach it will then be shown that for a non-optimal filter $E\{\underline{v}_k, \underline{v}_l^T\} \neq 0$ where $k \neq l$. In the following derivations the assumption that the observables are not time correlated will be held, as this assumption is often made in GNSS processing software, although this assumption in reality is not the case.

All of the derivations in this section done in terms of the predicted residual, $\underline{v}_k$, may also be applied to the post-fit residual because of the relationship between the predicted and post-fit residuals. Using Eq. (3.59),

$$E\{\underline{e}_k, \underline{e}_l^T\} = Q_{y_k} Q_{v_k}^{-1} E\{\underline{v}_k, \underline{v}_l^T\} Q_{v_k}^{-1} Q_{y_k} \tag{5.1}$$

From this it is clear that where $E\{\underline{v}_k, \underline{v}_l^T\} = 0$ it follows that $E\{\underline{e}_k, \underline{e}_l^T\} = 0$.

In the interests of simplifying the notation the transition matrix $\Phi_{k,k-1}$ will henceforth be abbreviated to Φ_k.

5.2.1 Optimal linear filter

In this subsection a number of underlying assumptions will be made. The first assumption is that at any given epoch the observables are not correlated with each other. The second assumption is that the errors in the observables between any given epochs are not correlated with each other. Whilst these two assumptions are not entirely true for high-rate GNSS observations, they are often made when designing GNSS processing software (see §3.8). Thirdly, it is assumed that the observables are Gaussian. Finally, all derivations in this section are with respect to the linear Kalman filter.

In the interests of clarity the observables will be represented by $\underline{Y}$ and the state vector by $\underline{X}$. For the linear Kalman filter $\underline{Y} \equiv \underline{y}$ and $\underline{X} \equiv \underline{x}$. Where linearisation has taken place $\underline{Y} \equiv \Delta \underline{y}$ and $\underline{X} \equiv \Delta \underline{x}$, as modelled by Eq. (3.12). These are small corrections to the a priori estimates of $\underline{y}$ and $\underline{x}$ which arise due to the linear approximation of the design matrix (see §3.2.1).

An optimal filter displays a number of characteristics. Firstly, the predicted residuals should have zero mean, such that

$$E\{\underline{v}\} = 0 \tag{5.2}$$

and should not be correlated in time, such that

$$E\{\underline{v}_k, \underline{v}_l^T\} = 0 \text{ where } k \neq l \tag{5.3}$$

The characteristic of zero mean shows that there is no systematic bias that has been unaccounted for during estimation and the characteristic of no correlation in time shows the predicted residual at a given epoch is not influenced by those at any other epoch.

The above conditions specify an optimum filter, as the noise Q_{Y_k} and Q_{w_k} are

both known. The predicted residual, given by Eq. (3.28), is

$$\underline{v}_k = \underline{Y}_k - A_k \hat{\underline{X}}_{k|k-1}$$

Consider the covariance of $\underline{v}_k$ and $\underline{v}_l$, where $l < k$.

$$
\begin{aligned}
E\{\underline{v}_k, \underline{v}_l^T\} &= E\{\underline{Y}_k - A_k \hat{\underline{X}}_{k|k-1}, \underline{Y}_l^T - \hat{\underline{X}}_{l|l-1}^T A_l^T\} \\
&= E\{\underline{Y}_k, \underline{Y}_l^T\} - A_k \, E\{\hat{X}_{k|k-1}, \underline{Y}_l^T\} - E\{\underline{Y}_k, \hat{\underline{X}}_{l|l-1}^T\} \, A_l^T \\
&\quad + A_k \, E\{\hat{\underline{X}}_{k|k-1}, \hat{\underline{X}}_{l|l-1}^T\} \, A_l^T
\end{aligned}
$$

(5.4)

Following Teunissen (1990a), $E\{\underline{Y}_k, \underline{Y}_l^T\} = 0$, as it is assumed that observables at epochs k and l are uncorrelated, and $E\{\underline{Y}_k, \hat{\underline{X}}_{l|l-1}^T\} = 0$, as the state vector at time l is unaffected by observables at epoch k. Eq. (5.4) therefore simplifies to

$$E\{\underline{v}_k, \underline{v}_l^T\} = -A_k E\{\hat{\underline{X}}_{k|k-1}, \underline{Y}_l^T\} + A_k E\{\hat{\underline{X}}_{k|k-1}, \hat{\underline{X}}_{l|l-1}^T\} A_l^T \tag{5.5}$$

By further defining

$$Q_{\hat{X}_{k|k-1} Y_l} = E\{\hat{\underline{X}}_{k|k-1}, \underline{Y}_l^T\} \tag{5.6}$$

$$Q_{\hat{X}_{k|k-1} \hat{X}_{l|l-1}} = E\{\hat{\underline{X}}_{k|k-1}, \hat{\underline{X}}_{l|l-1}^T\} \tag{5.7}$$

then Eq. (5.5) can be written as

$$E\{\underline{v}_k, \underline{v}_l^T\} = -A_k Q_{\hat{X}_{k|k-1} Y_l} + A_k Q_{\hat{X}_{k|k-1} \hat{X}_{l|l-1}} A_l^T \tag{5.8}$$

as shown in Appendix A of Salzmann (1993).

Substituting the equation for the predicted state vector, given by Eq. (3.21), into the updated state vector, given by Eq. (3.23), means that the predicted state

vector can now be written as

$$
\begin{aligned}
\underline{\hat{X}}_{k|k-1} &= \Phi_k \underline{\hat{X}}_{k-1|k-1} \\
&= \Phi_k[\underline{\hat{X}}_{k-1|k-2} + K_{k-1}(\underline{Y}_{k-1} - A_{k-1}\underline{\hat{X}}_{k-1|k-2})] \\
&= \Phi_k[I - K_{k-1}A_{k-1}]\underline{\hat{X}}_{k-1|k-2} + \Phi_k K_{k-1}\underline{Y}_{k-1} \quad (5.9)
\end{aligned}
$$

By then combining epochs $l+1$ to k the predicted state vector then reads

$$
\underline{\hat{X}}_{k|k-1} = \prod_{i=l}^{k-1} \Phi_{i+1}(I - K_i A_i)\underline{\hat{X}}_{l|l-1} + \sum_{j=l}^{k-1} \left\{ \prod_{i=j+1}^{k-1} [\Phi_{i+1}(I - K_i A_i)] \right\} \Phi_{j+1} K_j \underline{Y}_j \quad (5.10)
$$

From Eq. (5.10) it is possible to find $Q_{\hat{X}_{k|k-1}Y_l}$ by taking expectations with respect to $\underline{Y}_l$. Since $E\{\underline{\hat{X}}_{k|k-1}, \underline{Y}_l^T\} = 0$ and $E\{\underline{Y}_j, \underline{Y}_l^T\} = 0$ where $j \neq l$ this gives

$$
Q_{\hat{X}_{k|k-1}Y_l} = \prod_{i=l+1}^{k-1} [\Phi_{i+1}(I - K_i A_i)] \Phi_{l+1} K_l E\{\underline{Y}_l, \underline{Y}_l^T\} \quad (5.11)
$$

Similarly, from Eq. (5.10) it is possible to find $Q_{\hat{X}_{k|k-1}\hat{X}_{l|l-1}}$ by taking expectations with respect to $\underline{\hat{X}}_{l|l-1}$. Since $E\{\underline{Y}_j, \underline{\hat{X}}_{l|l-1}^T\} = 0$ where $j \neq l$, as the predicted state vector cannot be affected by observables after the event, this gives

$$
\begin{aligned}
Q_{\hat{X}_{k|k-1}\hat{X}_{l|l-1}} &= \prod_{i=l}^{k-1} \Phi_{i+1}(I - K_i A_i) \, E\{\underline{\hat{X}}_{l|l-1}, \underline{\hat{X}}_{l|l-1}^T\} \\
&= \prod_{i=l+1}^{k-1} \left[\Phi_{i+1}(I - K_i A_i) \right] \Phi_{l+1}(I - K_l A_l) \, E\{\underline{\hat{X}}_{l|l-1}, \underline{\hat{X}}_{l|l-1}^T\} \quad (5.12)
\end{aligned}
$$

By reformulating Eq. (5.8) as

$$
E\{\underline{v}_k, \underline{v}_l^T\} = -A_k[Q_{\hat{X}_{k|k-1}Y_l} - Q_{\hat{X}_{k|k-1}\hat{X}_{l|l-1}}A_l^T] \quad (5.13)
$$

it can be seen that for an optimal filter, i.e. one where $E\{\underline{v}_k, \underline{v}_l^T\} = 0$, differencing

the terms contained within the square brackets equals zero. By defining

$$Q_{Y_l} = E\{\underline{Y}_l, \underline{Y}_l^T\} \tag{5.14}$$

$$Q_{\hat{X}_{l|l-1}} = E\{\underline{X}_{l|l-1}, \underline{X}_{l|l-1}^T\} \tag{5.15}$$

and using Eq. (5.12) and Eq. (5.11) the terms inside the square brackets become

$$Q_{\hat{X}_{k|k-1}Y_l} - Q_{\hat{X}_{k|k-1}\hat{X}_{l|l-1}} A_l^T = K_l Q_{Y_l} - \Phi_{l+1}(I - K_l A_l)Q_{\hat{X}_{l|l-1}} A_l^T \tag{5.16}$$

From Eq. (3.25) it is known that

$$K_l = Q_{\hat{X}_{l|l-1}} A_l^T (Q_{Y_l} + A_l Q_{\hat{X}_{l|l-1}} A_l^T)^{-1}$$

and hence

$$K_l Q_{Y_l} - (I - K_l A_l)Q_{\hat{X}_{l|l-1}} A_l^T = 0 \tag{5.17}$$

showing that $E\{\underline{v}_k, \underline{v}_l^T\} = 0$ and therefore an optimal filter.

5.2.2 Non-optimal linear Filter

In the case where the filter is non-optimal the covariance matrices Q_{Y_k} and Q_{w_k} are not known and are approximated as $Q_{Y_k}^*$ and $Q_{w_k}^*$ respectively. In this subsection the work of Salzmann (1993) will be extended to consider the non-optimal case. In all subsequent derivations subscript $*$ shall denote a non-optimal value and $\sim$ shall denote the 'true' value as given by the expectation operator. As such the Kalman gain matrix, given by Eq. (3.25), is also non-optimal and becomes

$$K_l^* = Q_{X_{l|l-1}}^* A_l^T (Q_{Y_l}^* + A_l Q_{X_{l|l-1}}^* A_l^T)^{-1} \tag{5.18}$$

The non-optimal formulations of Eq. (5.11) and (5.12) similarly become

$$Q^*_{\hat{X}_{k|k-1}Y_l} = \prod_{i=l+1}^{k-1} \left[\Phi_{i+1}(I - K_i^* A_i)\right] \Phi_{l+1} K_l^* \ E\{Y_l, Y_l^T\} \qquad (5.19)$$

$$Q^*_{\hat{X}_{k|k-1}\hat{X}_{l|l-1}} = \prod_{i=l+1}^{k-1} \left[\Phi_{i+1}(I - K_i^* A_i)\right] \Phi_{l+1}(I - K_l^* A_l) \ E\{\hat{X}_{l|l-1}^*, \hat{X}_{l|l-1}^{*T}\} \quad (5.20)$$

Replacing the expectations in Eq. (5.19) and (5.20) with the variance-covariance matrices given by Eq. (5.6) and (5.7) gives

$$Q^*_{\hat{X}_{k|k-1}Y_l} = \prod_{i=l+1}^{k-1} \left[\Phi_{i+1}(I - K_i^* A_i)\right] \Phi_{l+1} K_l^* \ Q_{Y_l} \qquad (5.21)$$

$$Q^*_{\hat{X}_{k|k-1}\hat{X}_{l|l-1}} = \prod_{i=l+1}^{k-1} \left[\Phi_{i+1}(I - K_i^* A_i)\right] \Phi_{l+1}(I - K_l^* A_l) \ \tilde{Q}^*_{\hat{X}_{l|l-1}} \qquad (5.22)$$

Eq. (5.8) can now be written in its non-optimal form,

$$E\{\underline{v}_k^*, \underline{v}_l^{*T}\} = -A_k Q^*_{\hat{X}_{k|k-1}Y_l} + A_k Q^*_{\hat{X}_{k|k-1}\hat{X}_{l|l-1}} A_l^T$$

$$= -A_k \left[\prod_{i=l+1}^{k-1} \Phi_{i+1}(I - K_i^* A_i)\right] \Phi_{l+1}[K_l^* Q_{Y_l} - (I - K_l^* A_l)\tilde{Q}^*_{\hat{X}_{l|l-1}} A_l^T]$$

$$\qquad (5.23)$$

where $\tilde{Q}^*_{Y_{l|l-1}} \equiv E\{\underline{Y}_l, \underline{Y}_{l-1}^T\} \neq Q^*_{Y_{l|l-1}}$; that is the true covariance differs from the value derived from the filter. Since now $K_l^* Q_{Y_l}^* - (I - K_l^* A_l) Q_{\hat{X}_{l|l-1}^*} A_l^T \neq 0$ it follows that $E\{\underline{v}_k^*, \underline{v}_l^{*T}\} \neq 0$ for a non-optimal filter.

5.3 Measuring correlation

It has been shown in §5.2 that for a non-optimal filter $E\{\underline{v}_k^*, \underline{v}_l^{*T}\} \neq 0$ when $k \neq l$. The PPP software used in this study is known as E-HP/PPP, which is developed by Fugro Intersite B.V. In this section the various components of the E-HP/PPP software will be described along with the datasets that were chosen. A number of statistical methods will then be used to show that the post-fit residuals from the E-HP/PPP software exhibit time correlation. In the case of the lag plots in

§5.3.4 the entire time series of residuals is considered, whereas for high rate data the correlations over smaller time periods are of greater interest and therefore smaller samples of the time series were considered.

The methodology for determining the most suitable sample size to be used is given in §5.3.3. This is based on the mean and standard deviation of the lag 0 autocovariance. It is also important to consider that a sufficiently large sample is required to determine whether the residuals are correlated, but not so long so that the shorter term correlations are hidden within the data.

5.3.1 Software

The GNSS PPP processing software used in this thesis is developed by Fugro Intersite B.V. and known internally as E-HP/PPP. It uses the semi-recursive Kalman filter given by de Jong (2008b) (see also §3.6) and implemented as outlined in §3.8. The test statistics used in the software were implemented prior to this study, but use the equations given in §4.3.1 and §4.3.2. All processing and subsequent development is based upon a branch of the E-HP/PPP software taken on 30 November 2008. A subsequent branch from the software was taken on 9 April 2010 with all previous modifications merged into the new branch.

There are three stages to processing RINEX data using the E-HP/PPP software, which are given in the proceeding three subsections.

combrine

The *combrine* software combines a RINEX observation file, which contains the observations recorded by a GNSS receiver, with the appropriate navigation file, which contains the ephemerides used for determining the position, velocity, acceleration and clock information for observed satellites, as well as auxiliary information such as health and transmission time. This combined file is subsequently referred to as an ON file.

rinsp3

The *rinsp3* software takes an SP3 file containing precise orbit and clock values and merges these with the observations and navigation data contained within an ON file. These are in the form of corrections to the orbit and satellite clocks as specified by the navigation file. The same underlying library is utilised by the GNSS receivers used by customers of Fugro Intersite B.V. These receive corrections in real-time to the broadcast ephemeris over a satellite uplink. The reason for the *rinsp3* software generating corrections relative to the navigation message is in order to maintain equivalency with this real-time application. This merged file is subsequently referred to as an ONC file.

ppprin

The *ppprin* software is a front end to the E-HP/PPP software that can process an ONC file using the PPP technique. Since this only requires a single station it requires only one input file. The software is accompanied with an extensive configuration file that allows a number of parameters to be adjusted, e.g. the processing mode, a priori variances, spectral densities for common parameters, etc. All data is output in a human readable form, although for this project the software has been tailored to output the quality control statistics outlined in Chapter 4 that can subsequently be analysed automatically by custom scripts written using the Python programming language.

5.3.2   Datasets

Four datasets were chosen for the study and their locations are shown in Figure 5.1. Three of these were taken from IGS Continuously Operating Reference Station (CORS) sites with a data interval of one second. This was the highest rate CORS data that could be obtained and was only available from a subset of the entire network. Since the data is collected from a static receiver, i.e. one that does not move, they are referred to as the 'static' datasets. The sites that were chosen were ABPO (located on the island of Madagascar in the Indian Ocean), FALK (located

on the Falkland Islands in the South Atlantic) and GUAM (located on the island of Guam in the Pacific Ocean). Their locations were selected such that there would
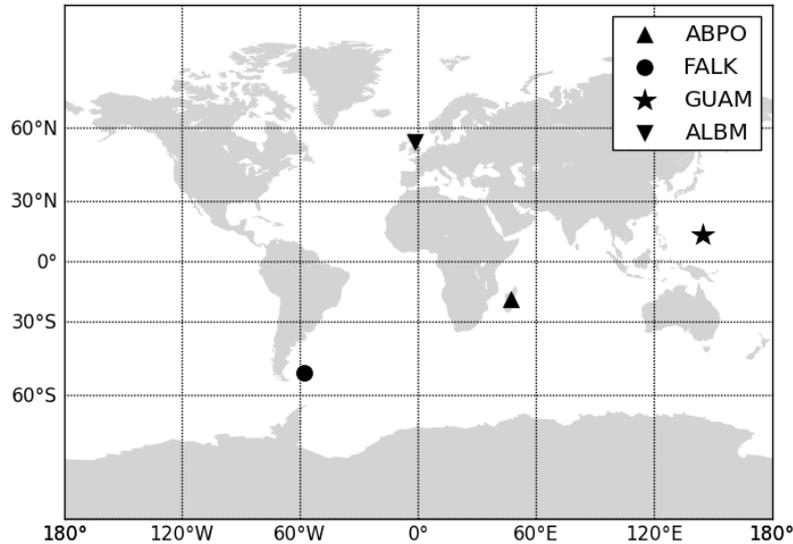
Figure 5.1: Locations of the 'static' sites chosen.

be an equal number of sites above and below the equator, at low ($0°$ – $30°$) and high ($30°$ – $60°$) latitudes and approximately equally spaced in longitude. Metadata examined on the IGS website was also used to ensure sites of high data quality.

For each site 24 hours of data from 25th December 2009 were downloaded from the IGS website. The final orbits and 5-second interval satellite clocks from the Center for Orbit Determination in Europe (CODE) for 24th, 25th and 26th December were obtained also from the IGS website. CODE orbits were chosen over the IGS in order to get the highest possible rate of clock data, as the IGS only provides clock data with an interval of 30 seconds. It is inconsistent to use orbits and clock data from different sources, hence the corresponding orbit data from CODE was also used.

The fourth dataset was collected on 22nd August 2008 at Albemarle Barracks in Northumberland, United Kingdom. This dataset will use the abbreviation ALBM throughout this study. The location of the site and the trajectory of the vehicle is shown in Figure 5.2. A GNSS antenna and receiver was mounted on top of a

Figure 5.2: Location of the 'kinematic' dataset at Albemarle Barracks in Northumberland, United Kingdom. The trajectory of the vehicle on which the GNSS antenna was mounted is shown in red.

Land Rover Defender and data collected whilst the vehicle drove continuously up and down the entire length of a disused runway at the airfield. Since the receiver is mounted on a moving platform it is referred to as the 'kinematic' dataset.

At the beginning of data collection the vehicle was driven at the slowest speed that was possible from one end of the runway to the other. The vehicle then turned around and was driven back again at an increased speed. This was repeated until the vehicle reached a maximum speed of 80 miles per hour. In addition to this the vehicle was also driven across the width of the runway, where it did a 180° turn and travelled back again in a trajectory parallel the previous one. This was done at the end of the runway and repeated to the other end of the runway.

The data period lasts for 6 h 39 m 30 s, is continuous and has a sample rate of one observation per second. A second receiver was erected alongside the runway and operated as a static reference station. This started collecting data before the vehicle

data was collected and continued for a short time afterwards to ensure that the data from both receivers would overlap. This was done so that a baseline solution between the reference and moving receiver could be computed and serve as a 'truth' solution.

It can be seen from Figure 5.2 that the site is fairly open with the exception of an area of trees on both sides of the north-east end of the runway and an area of trees at the south-west end of the runway, although this is only on the south-west facing side. All other areas of vegetation are far from the runway itself and buildings located in Albemarle Barracks are low rise.

Information about the four datasets is summarised in Table 5.1. The three CORS sites are all equipped with Ashtech UZ-12 receivers, whereas the data at Albemarle Barracks was collected using a Leica GX1230 GG. Similarly, Ashtech 701945-NN antennae with radomes are used at all three CORS sites used, although they vary in revision code. The Albemarle Barracks data was collected using a Leica AX1202 GG antenna with no radome. These differences in hardware used should be considered during the analysis.

In all cases the observation data were post-processed using E-HP/PPP in kinematic mode, hence no constraint was placed on the position parameter. Both L1 and L2 frequencies were treated as separate observations, instead of a linear combination being formed, and the ionosphere was included as a parameter in the estimation process. The post-fit residuals for each satellite and observation type were logged at each epoch. The analysis done here only includes the observations from GPS satellites, although the methods used here can be applied to other systems, such as GLONASS, Galileo and BeiDou.

Figures 5.3 - 5.6 each show an example of a residual time series from the four datasets. In each case an arbitrary satellite with a timespan of more than 3 hours has been chosen and each observation type plotted separately. Since there are four observation types for each satellite (L1 Code, L2 Code, L1 Carrier and L2 Carrier) they shall be referred to as a satellite and observation type pair.

Before performing any objective analysis of these residual time series it can be

Site name	ABPO	FALK	GUAM	ALBM
Latitude	19° 01' 5.898" S	51° 41' 37.144" S	13° 35' 21.588" N	55° 1' 19.7286" N
Longitude	47° 13' 45.163" E	57° 52' 26.641" W	144° 52' 6.099" E	1° 52' 41.5992" W
Receiver type	Ashtech UZ-12	Ashtech UZ-12	Ashtech UZ-12	Leica GX1230 GG
Antenna type	Ashtech 701945-NN REV G	Ashtech 701945-NN REV E	Ashtech 701945-NN REV B	Leica AX1202 GG
Radome	SCIT	SCIT	JPLA	None
First observation	25/12/2009 00:00:00	25/12/2009 00:00:00	25/12/2009 00:00:00	22/08/2008 07:51:13
Last observation	25/12/2009 23:59:59	25/12/2009 23:59:59	25/12/2009 23:59:59	22/08/2008 14:30:43

Table 5.1: Summary of datasets.
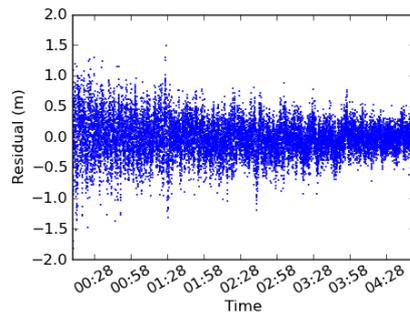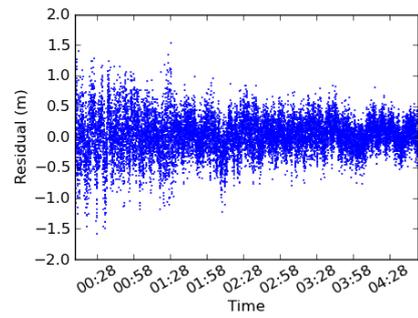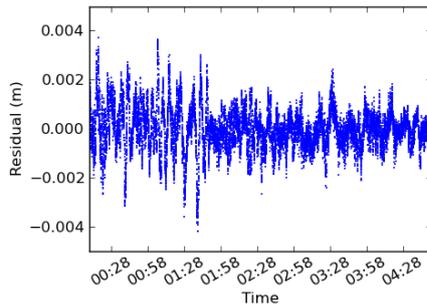
(a) L1 Code

(b) L2 Code
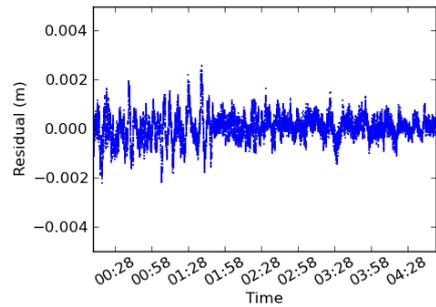
(c) L1 Carrier

(d) L2 Carrier

Figure 5.3: Time series of residuals for GPS SVN05 at ABPO.
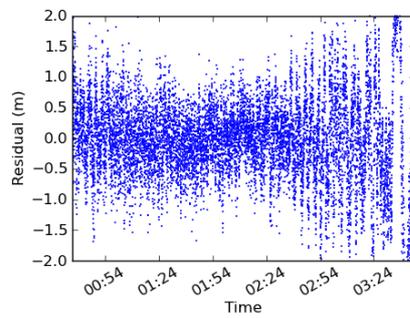
(a) L1 Code
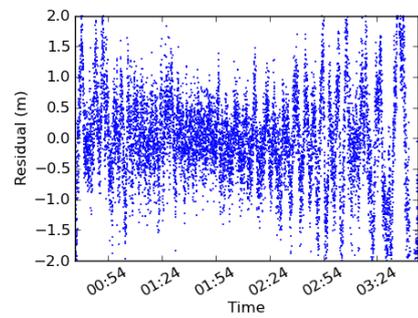
(b) L2 Code
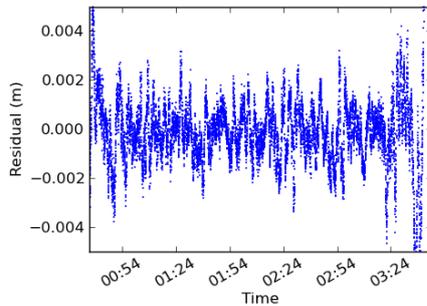
(c) L1 Carrier
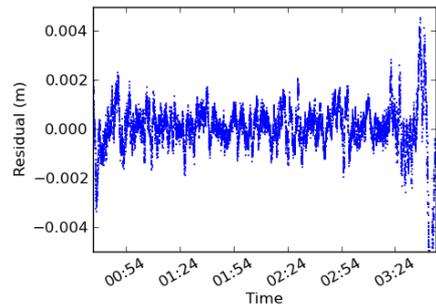
(d) L2 Carrier

Figure 5.4: Time series of residuals for GPS SVN05 at FALK.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
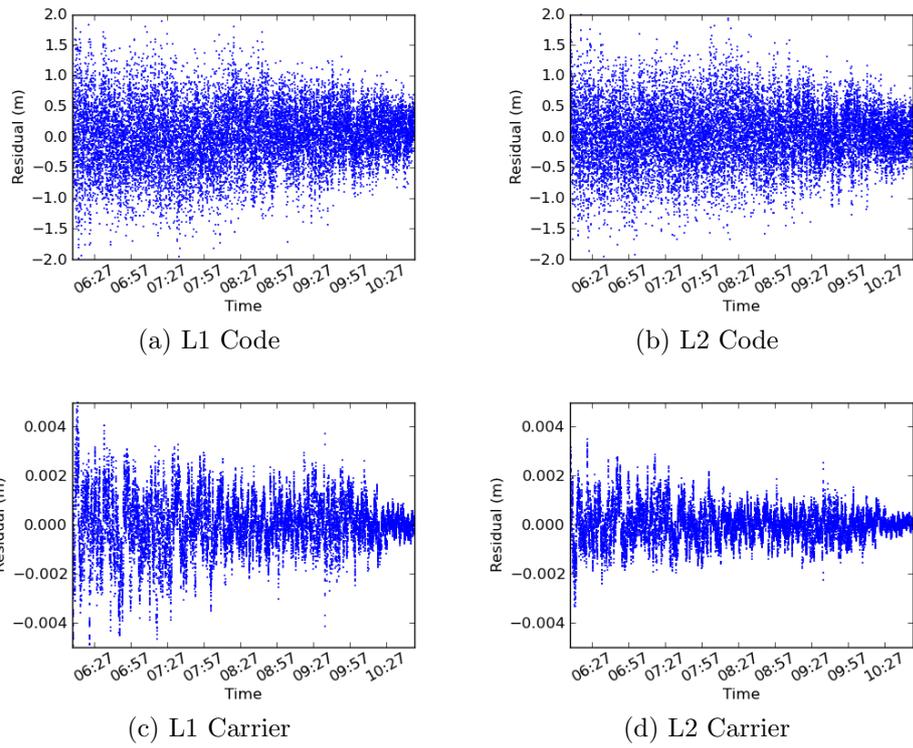
Figure 5.5: Time series of residuals for GPS SVN05 at GUAM.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
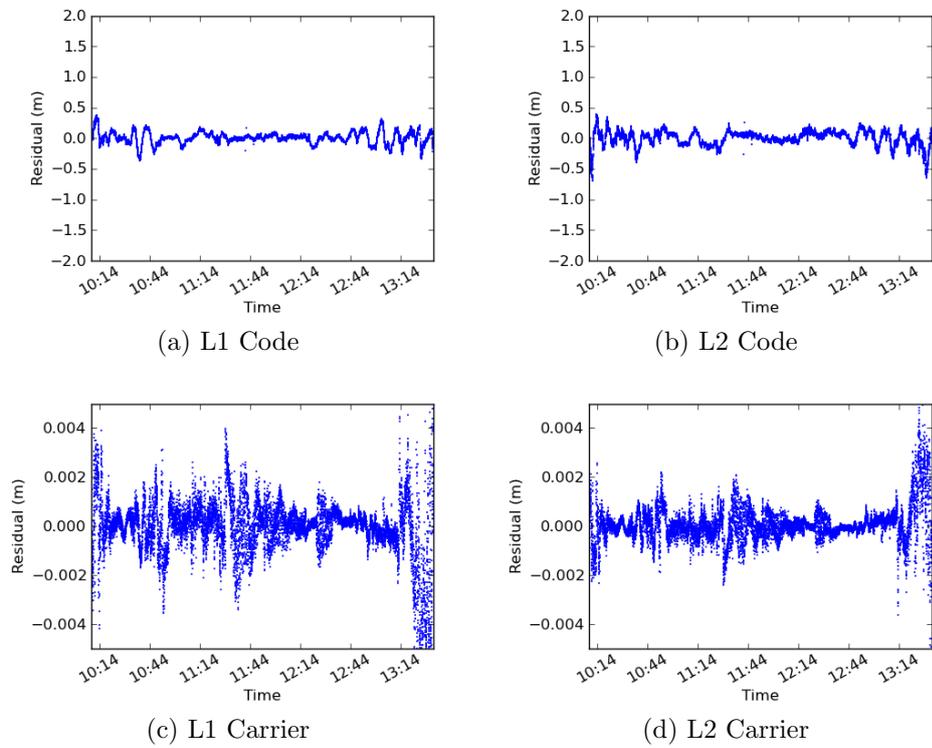
Figure 5.6: Time series of residuals for GPS SVN17 at ALBM.
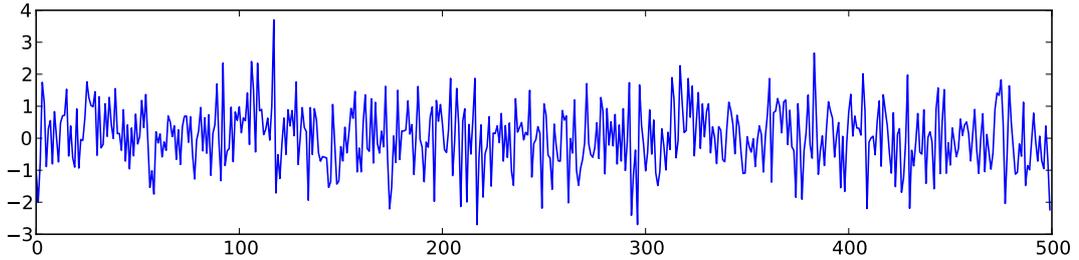
Figure 5.7: Normally distributed random noise.

seen that they do not characterise white noise. For comparison Figure 5.7 shows a synthetic sample of normally distributed random noise. It can also be seen that the residuals for L1 and L2 Carrier appear to be mirrored. A result of estimating a float ionosphere parameter at each epoch is that the carrier phase residuals on L1 and L2 are scaled such that

$$\underline{\hat{e}}_{\phi_2}^s \approx - \left(\frac{f_2}{f_1} \right)^2 \underline{\hat{e}}_{\phi_1}^s \tag{5.24}$$

where $\underline{\hat{e}}_{\phi_i}^s$ is the post-fit residual of the carrier phase observation on frequency i for satellite s and f_1 and f_2 are the frequencies of the L1 and L2 carrier phase. This scalar is -1.64, hence the mirrored appearance. As a result in many cases the analysis of carrier phase residuals is done on residuals regardless of frequency in the interests of brevity.

One further observation is that the residual time series of code observations for ALBM (see Figure 5.6) is much smoother than those seen at ABPO, FALK and GUAM (see Figures 5.3 - 5.5). It has already been noted that the ALBM dataset was collected using a Leica GNSS receiver, whereas the other datasets were collected using Ashtech receivers (see Table 5.1). The different results for ALBM are likely due to carrier-aided smoothing (see §2.2.8) of the code measurement by the Leica receiver.

5.3.3 Determining sample size

The methods that will be presented in §5.3.6 and §5.3.7 require a sufficiently large sample size of residuals in order to calculate an accurate estimate of the autocovariance and cross-covariance, and by definition autocorrelation and cross-correlation.

However, where the sample size is too large shorter term correlations may remain hidden.

The estimate of the covariance for a stochastic process $x(k)$, as shown later by Eq. 5.30 in §5.3.6, is

$$\hat{R}(s) = \frac{1}{N} \sum_{t=1}^{N-s} (x_k - \bar{x})(x_{k-s} - \bar{x})$$

where s is the lag time, N is the sample size and $\bar{x}$ is the mean of the stochastic process. It is therefore intuitive that as the sample size increases both the estimate of the mean and the autocovariance itself will improve. Similarly, the difference between each successive estimate of the autocovariance will become increasingly smaller. Therefore, by increasing the sample size sequentially it should be possible to determine the optimum value by finding the point at which the difference in size becomes negligible.

This was implemented by using a moving window to calculate the lag 0 autocovariance of all combinations of satellite, frequency and observation type over time. This process was repeated for sample sizes between 5 and 900 seconds in steps of 5 seconds. For each sample size the mean and standard deviation of the time series of lag 0 autocovariance values was recorded.

Figure 5.8 shows how the mean of the lag 0 autocovariance values for satellite G09 of the ALBM dataset varies with sample size. It can be seen that the mean does
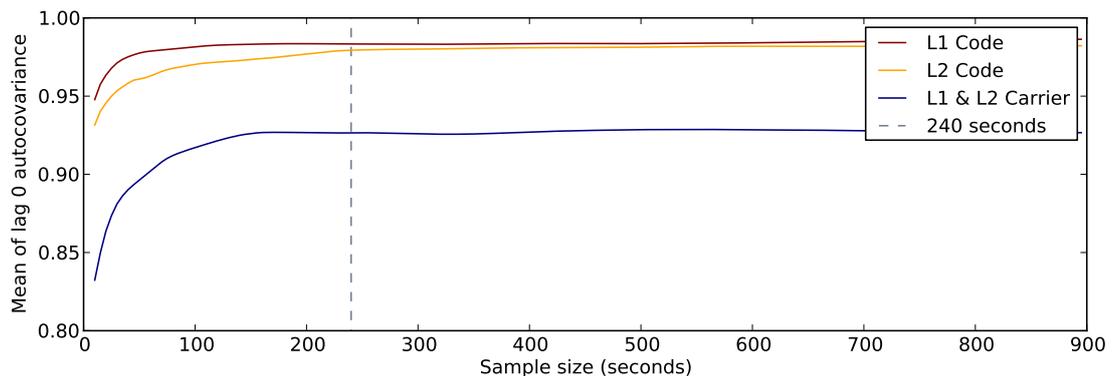
Figure 5.8: Mean of the lag 0 autocovariance for satellite G09 of the ALBM dataset with increasing sample size.

not change greatly after 240 seconds, as shown by the grey dashed line. Similarly, the change in standard deviation for the same lag 0 autocovariance values against

the change in sample size is shown in Figure 5.9b. Since the standard deviation

(a) Standard deviation   (b) First difference of standard deviation
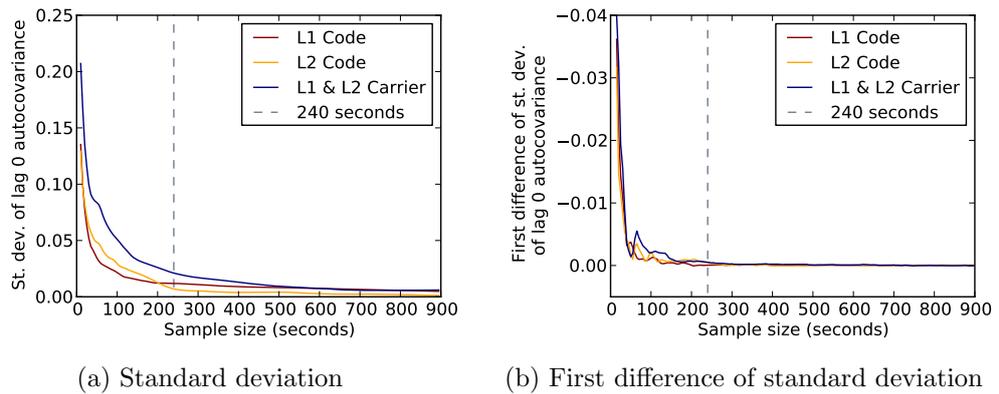
Figure 5.9: Standard deviation of the lag 0 autocovariance for satellite G09 of the ALBM dataset with increasing sample size.

will always decrease as more samples are added it is of interest to see the point at which the rate of change slows to a steady value. This can be done by taking a first difference of the standard deviation, as shown in Figure 5.9b. It can be seen that as the sample size increases the rate of change converges to a steady value at around 240 seconds, one again indicated by the grey dashed line.

It can be seen from Figures 5.8 and 5.9 that a sample size of 240 seconds is suitable for the analysis of the residuals. It is also important to consider the impact of the sample size on the user requirements outlined in §2.6. The value of 240 seconds does not significantly impact on the required availability and continuity set out in IMO (2001). Therefore the value of 240 seconds was chosen and will also be used in the estimation of autoregression parameters in Chapters 6 and 7.

5.3.4 Lag plots

A simple method of testing whether a time series is exhibiting whiteness is to produce a lag plot. This is a simple scatter plot where the values from the time series at two different lag points are plotted on opposing axes (Ryan 2007, Stockwell 2007). If the time series is random then the scatter plot will resemble a circular or elliptical cloud, as shown in Figure 5.10a. If the time series is autocorrelated, that is values at one epoch are in some way affected by another (see also Chapter 6) then the values will be diagonal. The pattern shown in Figure 5.10b is indicative of weak

(a) Random Noise

(b) Weak autocorrelation

(c) Strong autocorrelation

(d) Periodic signal
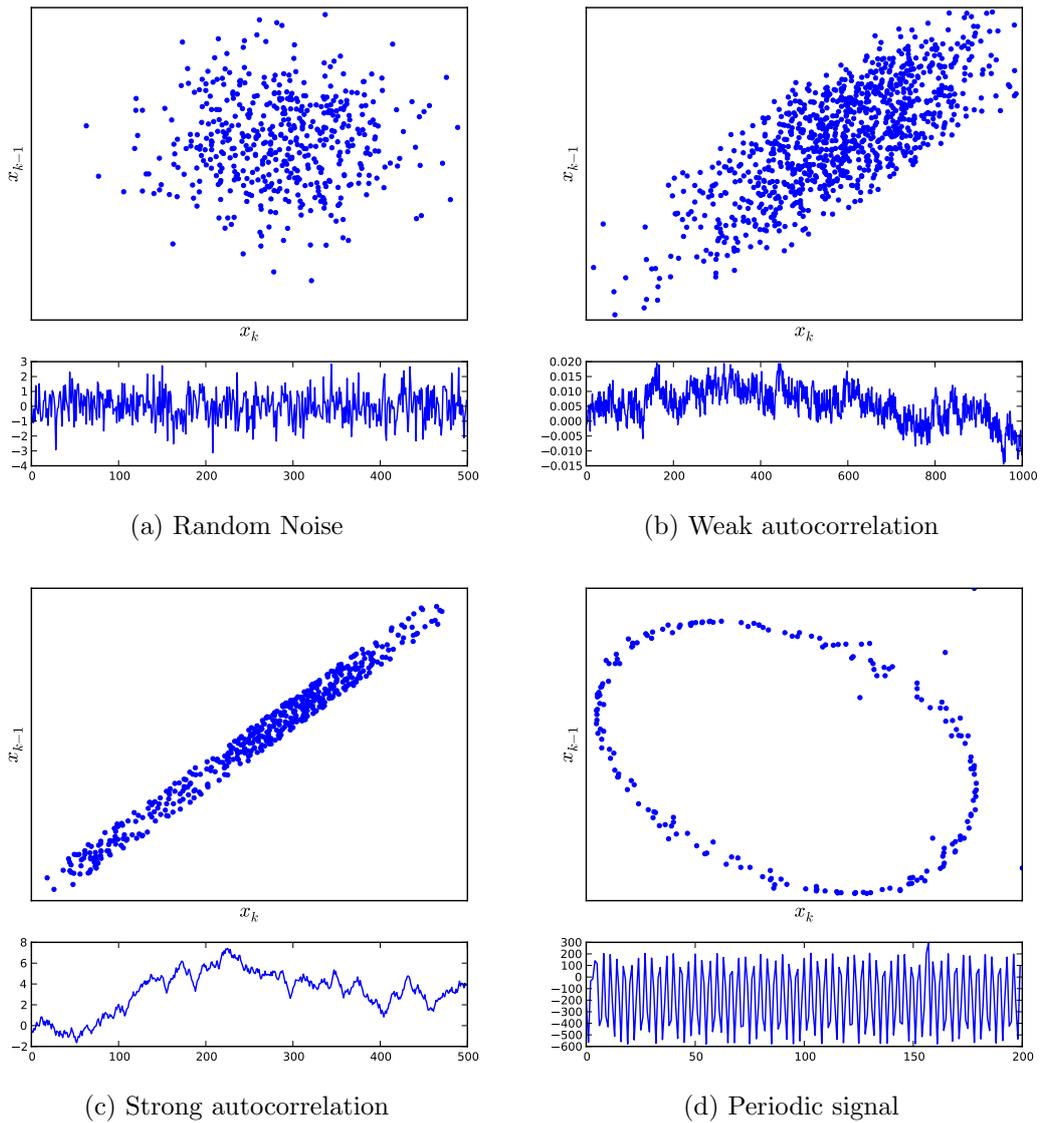
Figure 5.10: Examples of lag plots for different noise types with the corresponding time series shown below.

autocorrelation, whereas the much tighter diagonal form seen in Figure 5.10c shows strong autocorrelation. Finally, if the values are periodic then the pattern will appear circular, as shown in Figure 5.10d.

A time series of residuals was created for each satellite and observation type where data was continuous, i.e. there were no data gaps greater than the sample interval (1 second). From each time series a scatter plot was created with data at a lag of 0 seconds plotted against a lag of 1 second, represented by the values in bold

below,

$$k \qquad : e_1 \quad \boldsymbol{e_2} \quad \boldsymbol{e_3} \quad \cdots \quad \boldsymbol{e_{N-1}} \quad \boldsymbol{e_N}$$

$$k-1 \qquad : \qquad \boldsymbol{e_1} \quad \boldsymbol{e_2} \quad \cdots \quad \boldsymbol{e_{N-2}} \quad \boldsymbol{e_{N-1}} \quad e_N$$

where N is the length of the time series that is being analysed. In accordance with the literature, the term *lag 0, lag 1*, etc. will be used to refer to *a lag of zero epochs, a lag of one epoch*, etc., where an epoch is equal to the sample interval of 1 second.

Figures 5.11 - 5.14 show lag plots for the four observation types for a chosen satellite from the ABPO, FALK, GUAM and ALBM datasets respectively. The plots shown are considered representative of the lag plots for each dataset.

The lag plots of the code residuals of ABPO and GUAM (Figures 5.11a, 5.11b, 5.13a and 5.13b) resemble the random elliptical pattern of Figure 5.10a. The lag plots for the code residuals of FALK have a much larger ellipse that suggest that the dataset could be made up of a mixture of random and weakly correlated values. These contrast greatly to the lag plots for the code residuals of the ALBM dataset (Figures 5.14a and 5.14b), which have a very defined diagonal shape very similar to Figure 5.10c that suggests the residuals are strongly correlated. In all datasets the carrier phase residuals appear to show evidence of strong correlation. The elliptical diagonal seen in the carrier phase residuals of ABPO (Figures 5.11c and 5.11d) appears to be particularly elongated and hence indicative of strong correlation. The ALBM dataset appears to be the only one that shows strong correlation for all observation types. This reason for this is likely due to the probable carrier-aided smoothing of the code observations by the Leica receiver, which in turn affects the code residuals.

Although a general indicator of correlation, lag plots provide a useful method of quickly determining if any time correlation exists in a time series. Since the expectation is that the residuals should be random then any pattern other than the one seen in Figure 5.10a should be cause for further investigation. It is clear that the carrier phase residuals are strongly time correlated, as are the code observations in the kinematic dataset. A quantitative analysis of the correlation is performed in

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
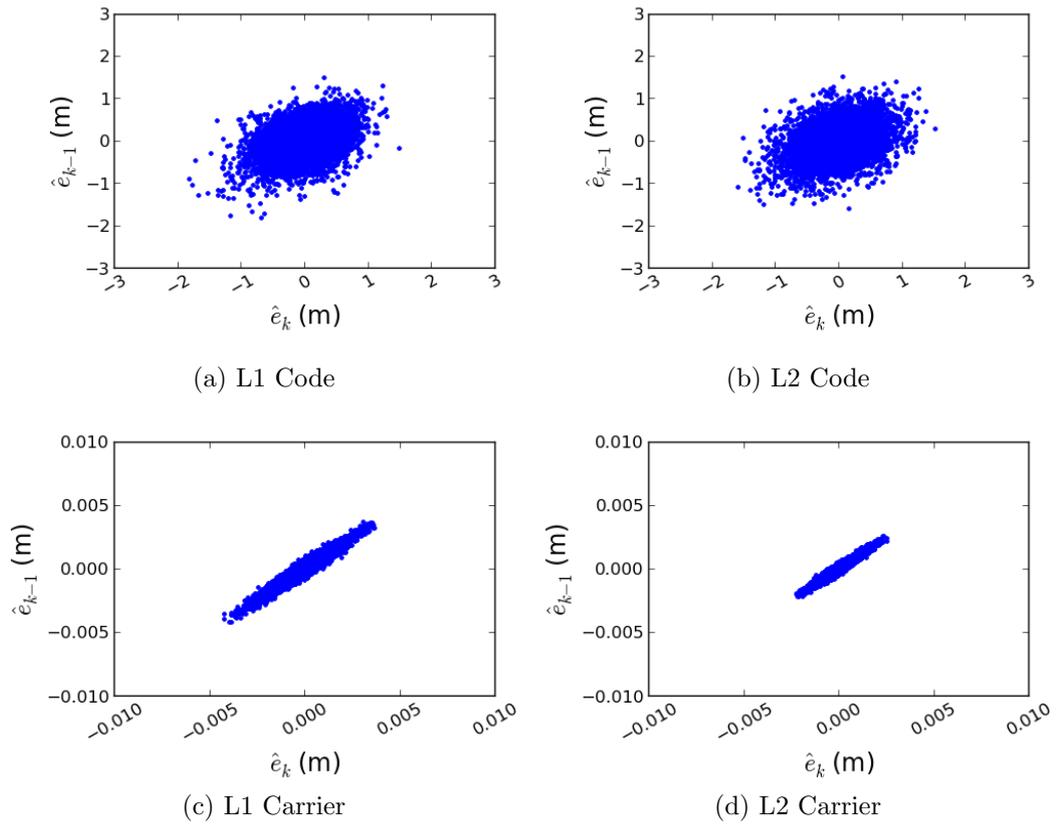
Figure 5.11: Lag plot for GPS SVN05 at ABPO.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
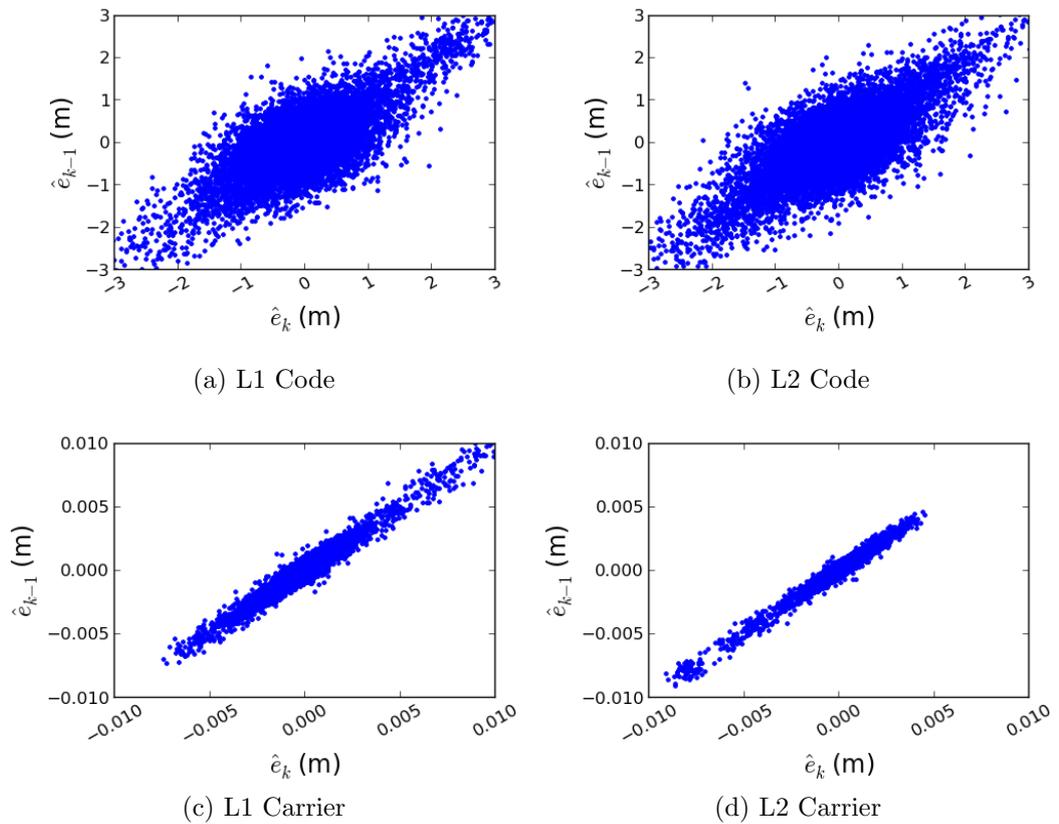
Figure 5.12: Lag plot for GPS SVN05 at FALK.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
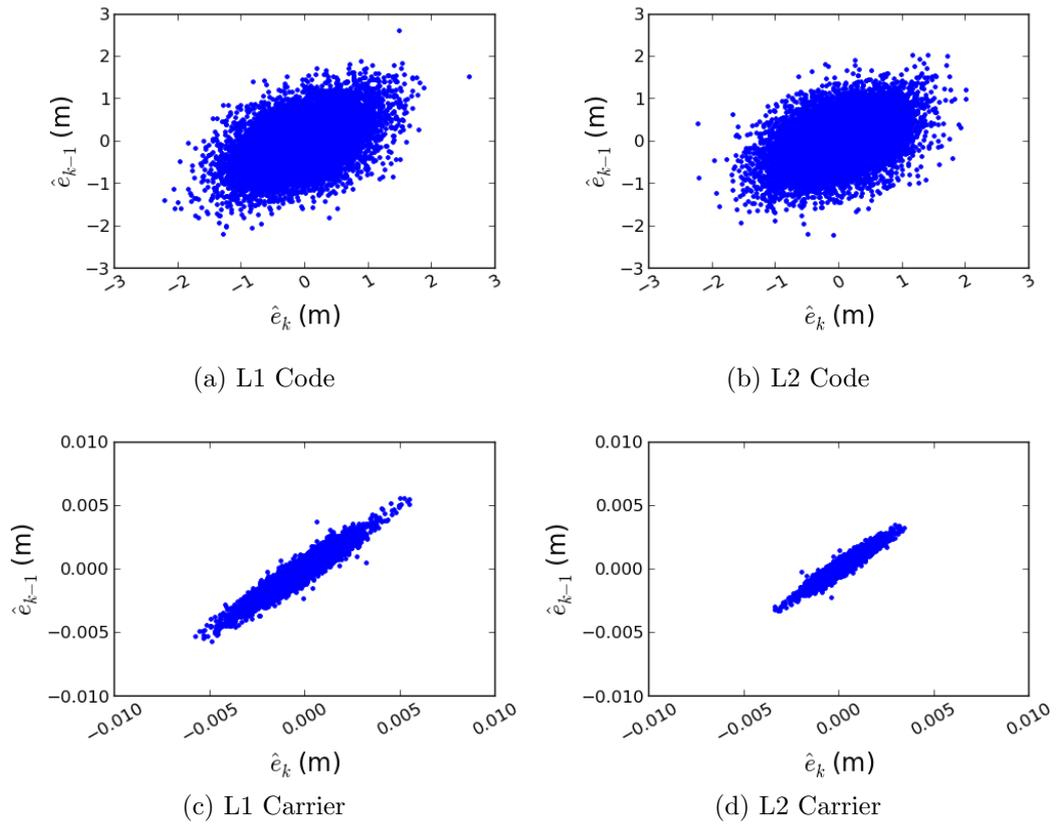
Figure 5.13: Lag plot for GPS SVN05 at GUAM.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
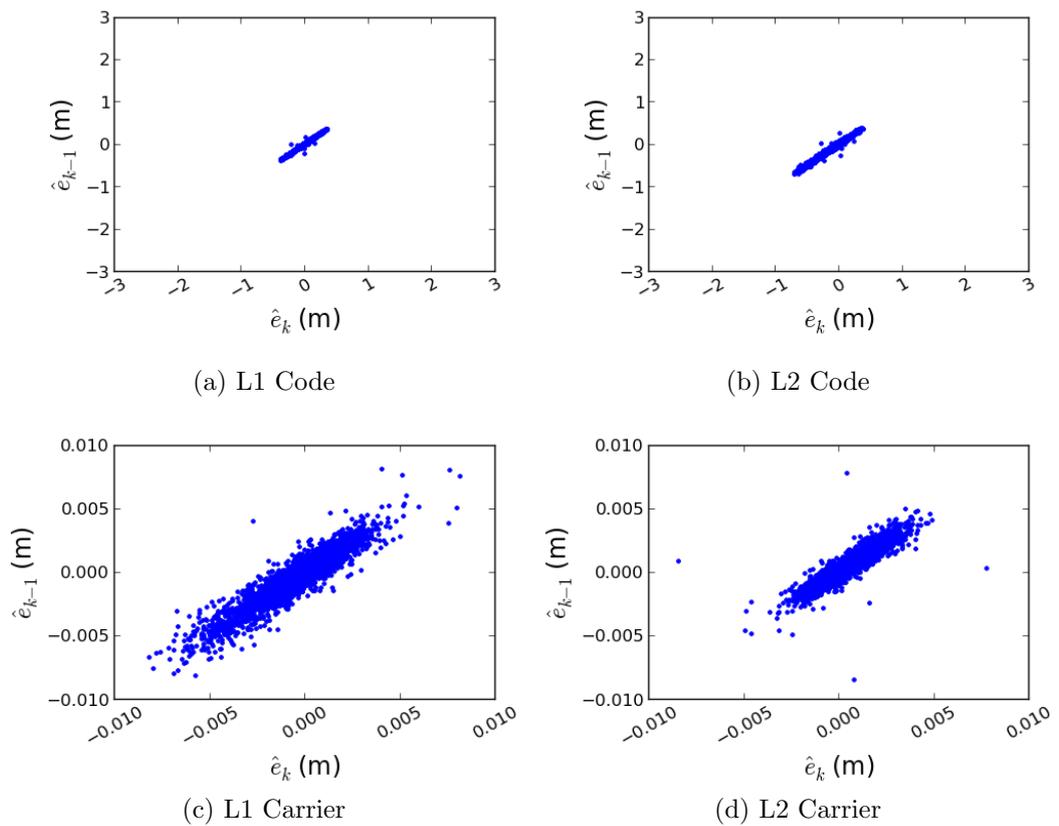
Figure 5.14: Lag plot for GPS SVN17 at ALBM.

99

the following section.

5.3.5   Durbin-Watson statistic

Durbin and Watson (1950) present a test criterion for determining whether the residuals from a least squares estimate are serially correlated. It is assumed that the error is distributed independently of the least squares parameters being estimated and that this error has zero mean and variance. The criterion is built upon the underlying assumption in least squares theory that the successive error terms in the regression model are distributed independently of each other. If this is not the case then the regression coefficients no longer have least variance and may lead to an underestimate of the variance of the estimate, as the usual formula for variance no longer applies.

Positive serial correlation is defined by Dougherty (2011) as a previous observable influencing the next one, i.e. v_{k-1} affects the value of v_k. This shows a tendency for a system to remain in the same state from one epoch to the next. The definition given for negative serial correlation is a previous observable being influenced by the current one, i.e. v_k effects the value of v_{k-1}. Since this definition comes from the field of econometrics it is difficult to transfer its definition when applied to physical processes. In the context of econometrics the first difference of observables are often used instead of the observables themselves and testing for negative serial correlation can ensure that the observables have not been overcorrected (Durbin and Watson 1951). As it will be seen in Chapter 6, this can be applied to physical processes that have been artificially altered, as is the case with whitening.

For the least squares model given by Eq. (3.1) the test statistic against positive serial correlation given in Durbin and Watson (1950) reads

$$d = \frac{\sum_{i=2}^{N} (\hat{\underline{e}}_i - \hat{\underline{e}}_{i-1})^2}{\sum_{i=1}^{N} \hat{\underline{e}}_i^2} \tag{5.25}$$

where $\hat{\underline{e}}$ is the vector of residuals, given by Eq. (3.7), and N is the size of the sample

of residuals used in the test.

It is not possible to calculate a single critical value that determines whether the null hypothesis that no positive serial correlation exists can be accepted. Instead a lower and upper bound are calculated, referred to as d_L and d_U respectively. If $d < d_L$ then residuals are said to exhibit positive serial correlation and if $d > d_U$ then they exhibit no serial correlation. For values where $d_L < d < d_U$ the test is deemed to be inconclusive. The values for d_L and d_U are tabulated for given given levels of significance in Durbin and Watson (1951) as a function of the sample size and the number of estimable parameters. As this is a one-tailed test, in order to test against negative serial correlation d is subtracted from 4 and then treated in the same manner as the test against positive serial correlation. This can be done as $0 < d < 4$, where $d = 2$ indicates no serial correlation in the residuals. Conversely, the critical values for testing against negative correlation can be calculated using

$$d_L^- = 4 - d_U^+ \qquad\qquad d_U^- = 4 - d_L^+ \qquad (5.26)$$

where d_L^+, d_U^+ and d_L^-, d_U^- are the lower and upper bounds for testing against positive and negative correlation respectively.

The tables published in Durbin and Watson (1951) are restricted to a sample of 100 observables. The number of observables that could be calculated was restricted given the computational limitations at the time. Tables with sample sizes up to 200 were later published by Savin and White (1977). In the PPP estimation procedure the number of parameters that are estimated at each epoch is dependent on the number of satellites and as a consequence the total number estimable parameters in most cases exceeds 20, the maximum number seen in Savin and White (1977). The chosen sample size of 240 also exceeds the maximum number of sample sizes tabulated. An algorithm by Farebrother (1980) allows for the computation of critical values for larger sample sizes and a greater number of regressors. Critical values calculated using this algorithm are published by Cummins (2012) and used as the source of the chosen critical values; $d_L = 1.61128$ and $d_U = 1.97055$. This is based on a sample size of 240 and 21 estimable parameters with a significance level of 5%.

The standard d test was used to determine whether to accept the null hypothesis that there is no positive or negative serial correlation in the post-fit residuals from the least squares estimate performed at the measurement update of the semi-recursive Kalman filter. A time series of residuals was created for each satellite and observation type and the Durbin-Watson test statistic calculated for a moving window of 240 seconds. Since continuous data is required, any window of data that contained a time gap greater than the sample frequency (1 second) was not used. A histogram of these values was then created to indicate the spread of values. Figures 5.15 - 5.18 each show a histogram of the Durbin-Watson values for a representative satellite in the four datasets. The critical values are not shown, since the histograms are intended to be indicative of the general trend. The general distribution of the Durbin-Watson values with respect to the critical values is shown in Table 5.2. Values where $d < d_L^+$ indicate positive serial correlation, $d_U^+ \leq d \leq d_L^-$ indicate no serial correlation and $d > d_U^-$ indicate negative serial correlation. Values where $d_L^+ \leq d \leq d_U^+$ and $d_L^- \leq d \leq d_U^-$ lie in the regions of inconclusivity.

Table 5.2: Distribution of Durbin-Watson values of residuals from all datasets with respect to the critical values.

Dataset	Observation type	$d < d_L^+$	$d_L^+ \leq d \leq d_U^+$	$d_U^+ < d < d_L^-$	$d_L^- \leq d \leq d_U^-$	$d > d_U^-$
ABPO	L1 Code	80.55%	16.47%	1.22%	1.71%	0.04%
	L2 Code	74.19%	20.56%	1.92%	3.30%	0.03%
	L1 & L2 Carrier	100.00%	0.00%	0.00%	0.00%	0.00%
FALK	L1 Code	83.89%	13.50%	1.03%	1.58%	0.00%
	L2 Code	83.22%	13.51%	1.18%	2.08%	0.01%
	L1 & L2 Carrier	99.91%	0.09%	0.00%	0.00%	0.00%
GUAM	L1 Code	77.20%	19.26%	1.38%	2.15%	0.01%
	L2 Code	71.61%	22.92%	2.00%	3.46%	0.02%
	L1 & L2 Carrier	100.00%	0.00%	0.00%	0.00%	0.00%
ALBM	L1 Code	100.00%	0.00%	0.00%	0.00%	0.00%
	L2 Code	100.00%	0.00%	0.00%	0.00%	0.00%
	L1 & L2 Carrier	100.00%	0.00%	0.00%	0.00%	0.00%

From Figures 5.15 - 5.17 it appears that the L1 and L2 code observations are distributed around $d = 1.4$. It can be seen by examining Table 5.2 that for these datasets the majority of values are below d_L^+, and hence suggest that the alternative hypothesis that the residuals show positive serial correlation must be accepted. Virtually no values for the ALBM dataset and very few in the 'static' datasets lie in the region indicating negative correlation. However, for the static datasets between
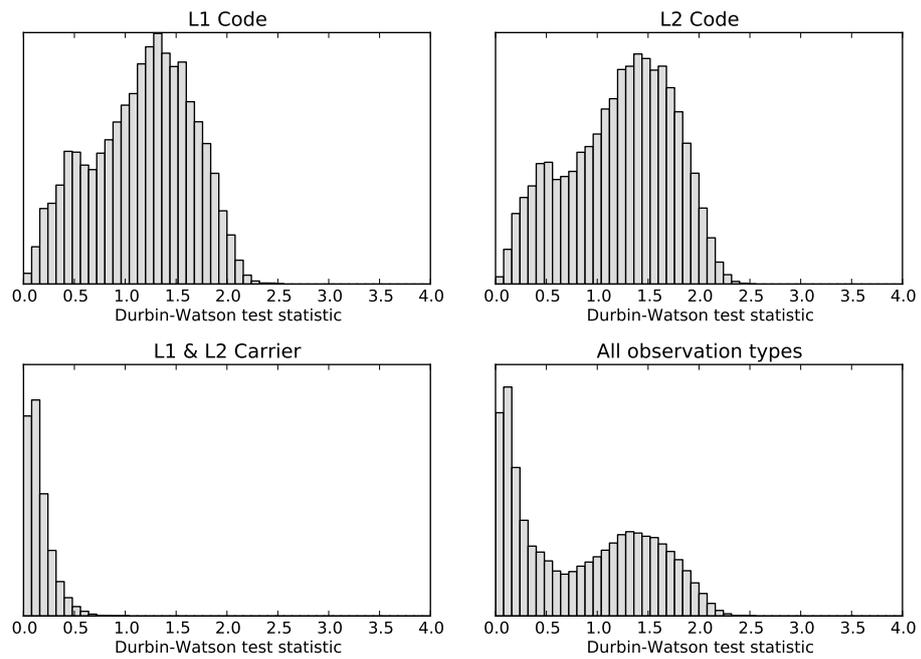
Figure 5.15: Histogram of Durbin-Watson values for ABPO.
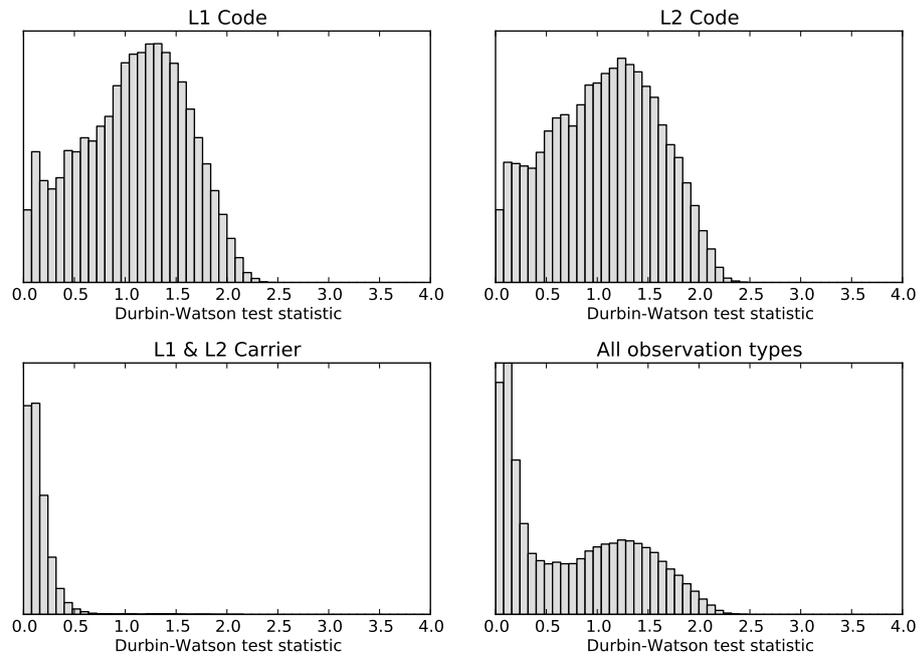
Figure 5.16: Histogram of Durbin-Watson values for FALK.
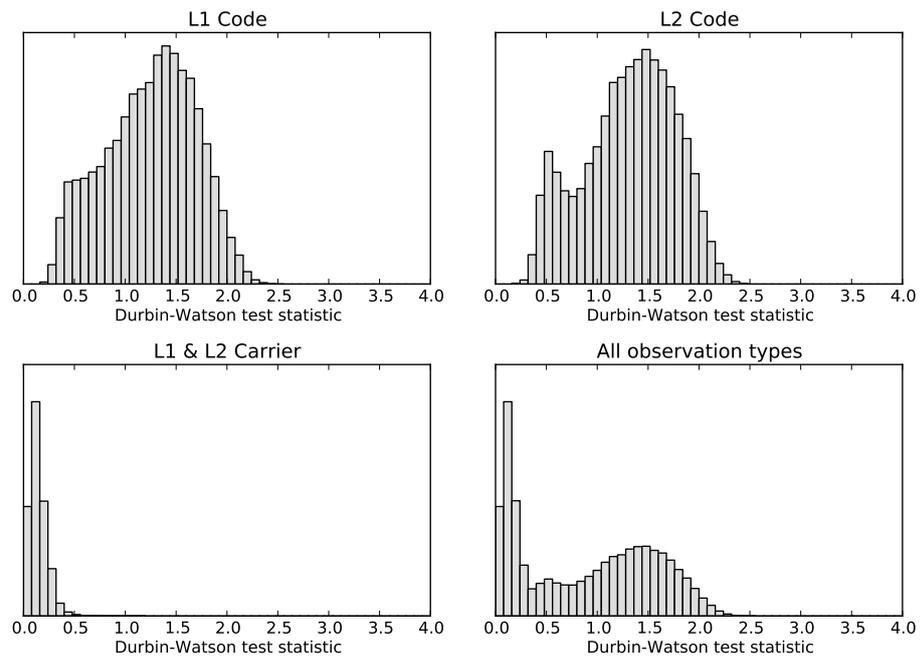
Figure 5.17: Histogram of Durbin-Watson values for GUAM.
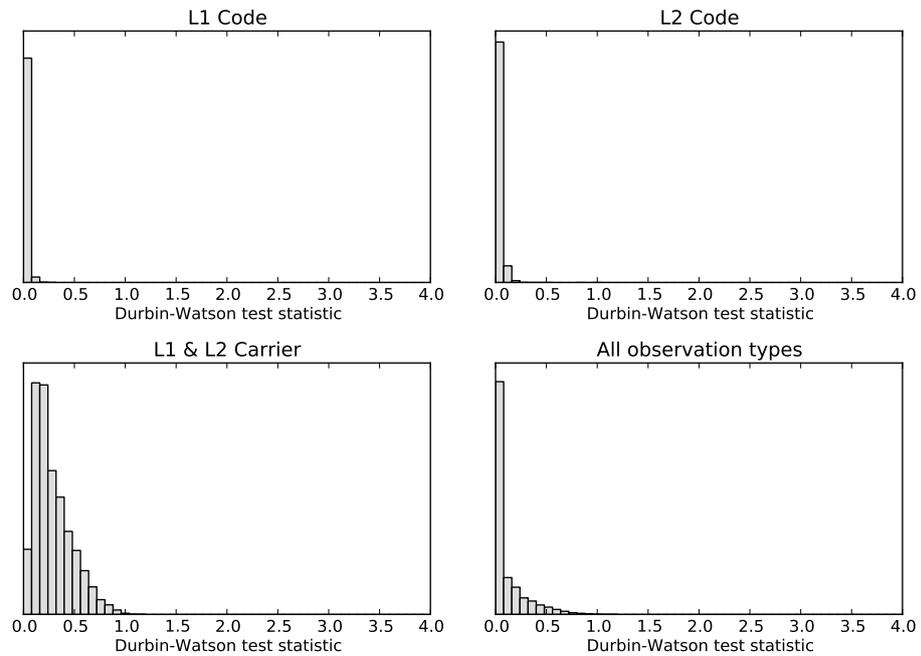
Figure 5.18: Histogram of Durbin-Watson values for ALBM.

15% and 26% of values for the code observations lie in the regions of inconclusivity. Despite this, there is a greater proportion of residuals that show positive serial correlation than those that either show no serial correlation or lie in the region of inconclusivity.

The carrier phase residuals for the static datasets show a greater proportion of values that are below d_L^+ when compared to the values for the code residuals. Not many values lie in the region of inconclusivity, although a reasonably large proportion accept the null hypothesis that the residuals are random. GUAM shows the highest proportion of values (22.92%) that can be said to be random. Despite this, the proportion of values that exhibit serial correlation (71.61%) is still far greater.

The values for the kinematic ALBM dataset in Table 5.2 along with Figure 5.18 contrast greatly with the three static datasets. Virtually none of the residuals exceed d_U^+ and very few are in the region of inconclusivity. By contrast, the code residuals appear to be more correlated than the carrier phase. This can be seen in the histogram, where the overwhelming majority of the code values are concentrated in the first column. The carrier phase values are distributed between 0.0 and 1.0, although skewed towards 0. As with the 'static' datasets, although to a lesser extent, there is a small peak at 2.0 in the histogram for the ALBM carrier phase values.

It can therefore be concluded that both the code and carrier phase residuals on L1 and L2 frequencies exhibit positive serial correlation. For the static datasets the extent of time correlation is greater for the carrier phase residuals than for the code residuals, whereas for the kinematic dataset the time correlation is very high for the code observations.

5.3.6 Autocovariance and autocorrelation functions

Although the Durbin-Watson test statistic presented in §5.3.5 gives a good indication of the level of correlation in a time series, it is only calculated for lag 1. For a process $x(k)$ the autocorrelation function is a dimensionless measure of correlation between x_k and x_{k-s} as the lag time s increases (Priestley 1981).

The autocovariance of x_k and x_{k-s} is defined as

$$R(s) = E\{x_k - E\{x\}, x_{k-s} - E\{x\}\} \qquad (5.27)$$

and the autocorrelation function is then given by

$$\rho(s) = \frac{R(s)}{R(0)} \qquad (5.28)$$

As s increases then the correlation between x_k and x_{k-s} is expected to decrease, hence $\rho(s) \to 0$ as $s \to \infty$, although this may not be the case if the process is periodic (see Figure 5.19d). For all autocovariance functions the following properties apply (Box and Jenkins 1976, Priestley 1981):

1. $R(0) = \sigma_x^2$, the variance of the process $x(k)$, remains the same throughout, i.e. it is a stationary process (see §6.2).

2. $|R(s)| \leq R(0)$ for all s.

3. $R(-s) = R(s)$ for all s when $x(k)$ is a real valued process.

Similarly, for all autocorrelation functions the following properties apply:

1. $\rho(0) = 1$, since $\frac{R(0)}{R(0)} = 1$.

2. $|\rho(s)| \leq 1$ for all s.

3. $\rho(-s) = \rho(s)$ for all s when $x(k)$ is a real valued process.

In practice, the best estimate of the sth lag of autocorrelation $\rho(s)$ is calculated by

$$\hat{\rho}(s) = \frac{\hat{R}(s)}{\hat{R}(0)} \qquad (5.29)$$

where

$$\hat{R}(s) = \frac{1}{N} \sum_{t=1}^{N-s} (x_k - \bar{x})(x_{k-s} - \bar{x}) \qquad (5.30)$$

is the estimate of autocovariance $R(s)$ and $\bar{x}$ is the mean of the time series (Box and Jenkins 1976).

(a) Random noise

(b) Weak correlation

(c) Strong correlation

(d) Periodic correlation
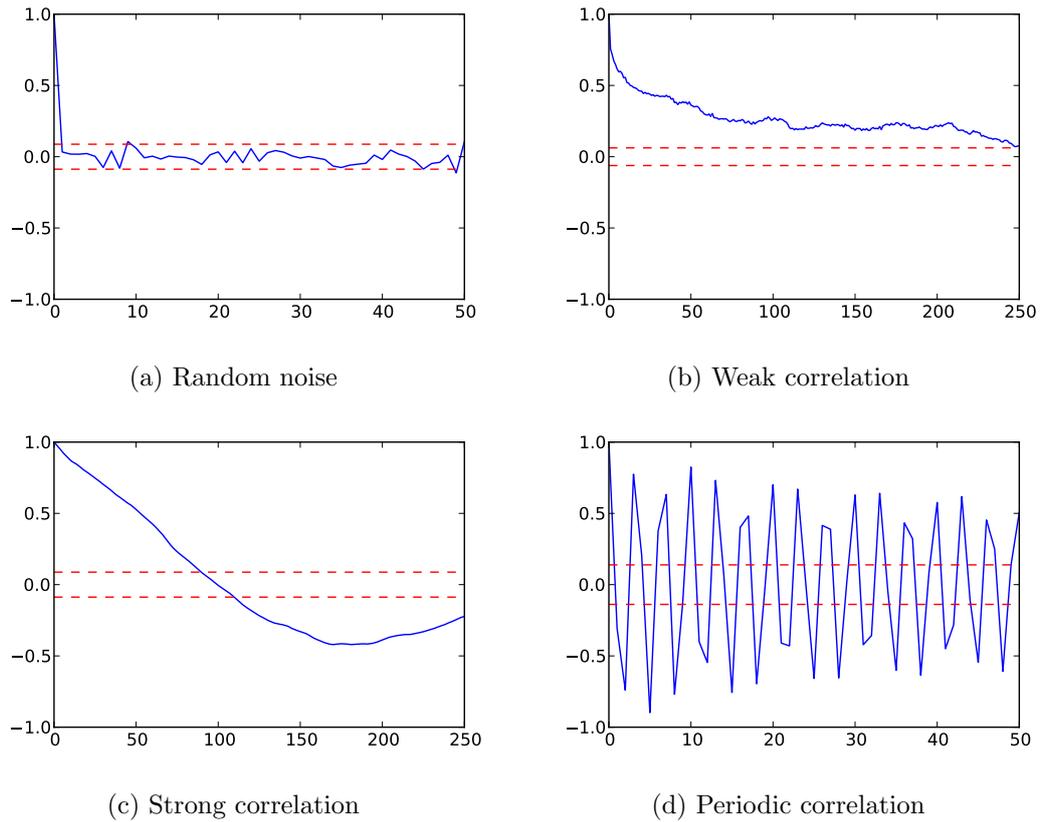
Figure 5.19: Examples of autocorrelation plots for different noise types.

Figure 5.19 shows autocorrelation plots for the four time series given in Figure 5.10. Since for all lags (Box and Jenkins 1976)

$$\text{var}\{\rho(s)\} \approx \frac{1}{N} \tag{5.31}$$

the 95% confidence levels, c, are calculated as

$$c = \pm\frac{z_{1-\alpha/2}}{\sqrt{N}} \tag{5.32}$$

where z is the percent point function of the standard normal distribution, α is the chosen level of significance and N is the sample size. In this case $\alpha = 0.05$, $N = 240$ and $z_{0.975} = 1.96$. For each plot the 95% confidence levels are shown as dotted lines. Any values contained within these bounds are said to exhibit no autocorrelation with a confidence level of 95%.

It can be seen in Figure 5.19a that there is no discernible pattern in the auto-correlation graphs. This dataset consists of normally distributed random numbers,

where $\mu = 0$ and $\sigma = 1$. Figures 5.19b and 5.19c are both examples of positive autocorrelation, the latter showing a more pronounced autocorrelation. In the case of a periodic pattern occurring in residuals a sequence such as Figure 5.19d will occur. This represents a signal that is repeating at a fixed frequency. The lag plots in Figure 5.19 use the same data that was used to create Figure 5.10.

A time series of residuals was generated from the data for each satellite and observation type where data was continuous, i.e. there were no data gaps greater than the sample interval (1 second). Each time series was divided into 240 second blocks and the autocorrelations for lags $\{1, \dots, 120\}$ were calculated. The mean autocorrelation for each lag was calculated and these values used for the autocorrelation plot. This approach was chosen to best portray the autocorrelations over a short period of time instead of the whole satellite pass.

Figures 5.20 - 5.23 show the autocorrelation graphs for the ABPO, FALK, GUAM and ALBM datasets. In each case a satellite has been chosen that is indicative of the type of correlation seen at the site in question. The autocorrelation graphs for all four observation types are shown.

In general the autocorrelation graphs for the ALBM dataset show more correlation than those for ABPO, FALK and GUAM. For the latter three sites it is common in all cases that the autocorrelations are higher and for longer lags for the carrier phase residuals than for the code residuals. It is also noticeable that for the code residuals there is a sharp drop in the autocorrelation at lag 1, whereas for the phase residuals the decrease is more gradual.

The autocorrelation plots for both the code and carrier residuals of the ALBM dataset are much different in character. In this case the code residuals show much more correlation, with a gently decreasing curve. It is particularly interesting to note that for the ALBM dataset the code observations appear to show more correlation than the carrier phase observations, but for the three static sites the opposite is seen. As noted previously this can be explained by the probable carrier-aided smoothing of the code observations by the Leica receiver. By comparison the carrier phase residuals drop more rapidly and have a more changeable pattern. This can be
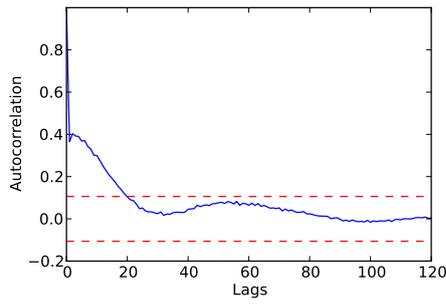
(a) L1 Code
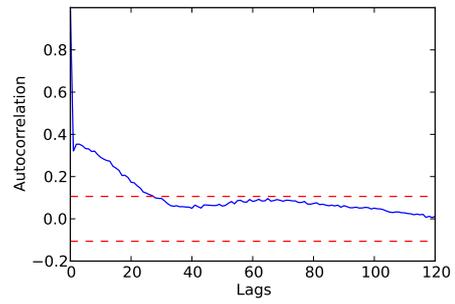
(b) L2 Code
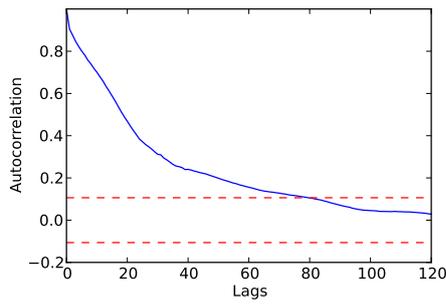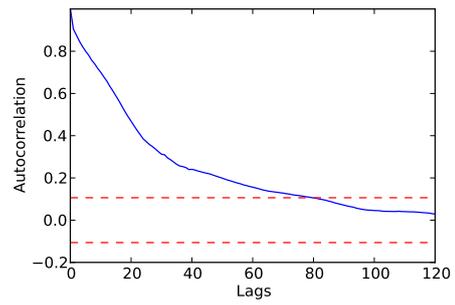
(c) L1 Carrier

(d) L2 Carrier

Figure 5.20: Autocorrelation plot for GPS SVN05 at ABPO.
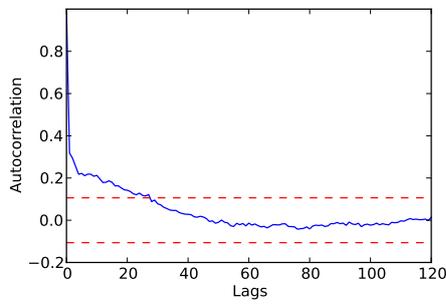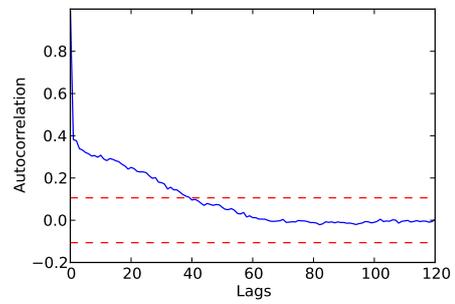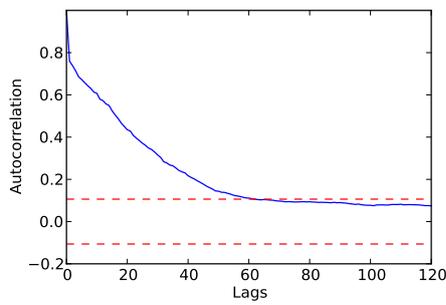
(a) L1 Code

(b) L2 Code
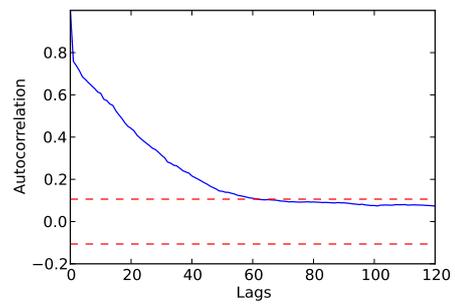
(c) L1 Carrier

(d) L2 Carrier

Figure 5.21: Autocorrelation plot for GPS SVN05 at FALK.
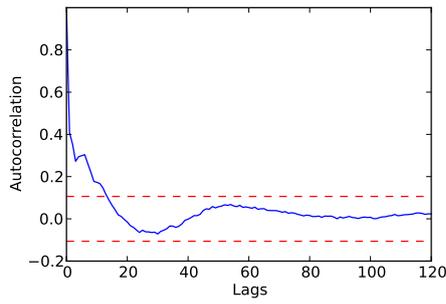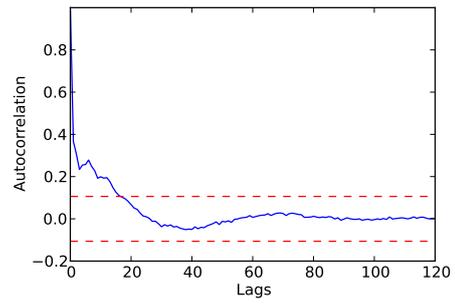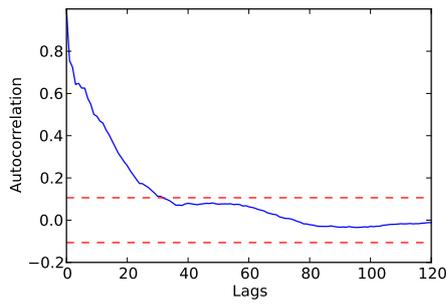
(a) L1 Code

(b) L2 Code
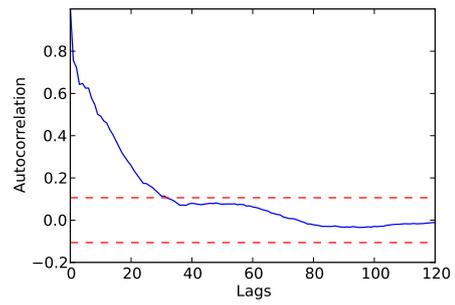
(c) L1 Carrier

(d) L2 Carrier

Figure 5.22: Autocorrelation plot for GPS SVN05 at GUAM.
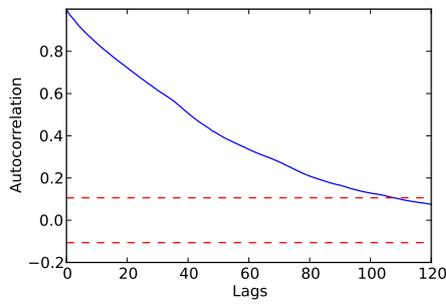
(a) L1 Code
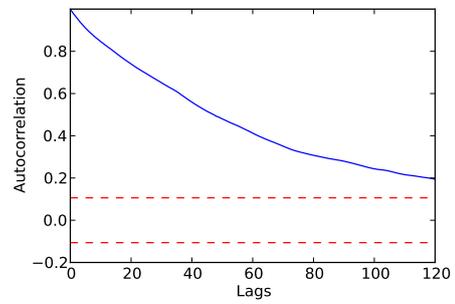
(b) L2 Code
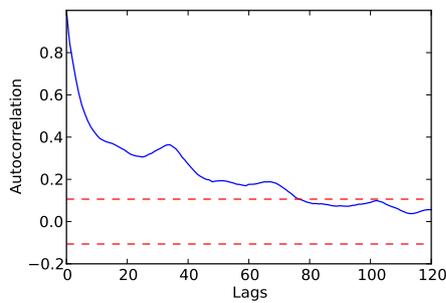
(c) L1 Carrier
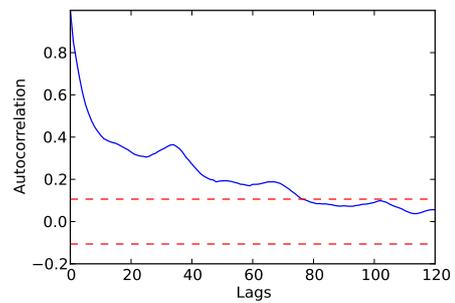
(d) L2 Carrier

Figure 5.23: Autocorrelation plot for GPS SVN17 at ALBM.

attributed to the motion of the antenna.

The autocorrelation plots point to high correlations at shorter lag times which decay as lag time increases. Confirming what was seen in the lag plots and the Durbin-Watson values, the autocorrelations of the code residuals for the static datasets tail off much quicker than those for the carrier phase. The opposite is seen for the kinematic dataset, where the code observations appear to be much more correlated than the carrier phase ones.

Autocorrelations can only measure the correlation for the residuals of one satellite and observation type pair and does not give any indication of how it is correlated with any other lagged residual. This is addressed in the following section.

5.3.7 Cross-covariance and cross-correlation functions

In §5.3.6 the autocovariance function was presented, but only considered as a function of a residual time series for one satellite and observation type. By considering the entire vector of residuals at one epoch with the same vector of residuals at a lag epoch it is possible to determine not only the relationship between the residual for one satellite and observation type pair between two epochs, but the relationship between the residuals of two different satellite and observation type pairs at two different epochs.

Given two time series $x_1(k)$ and $x_2(k)$ the cross-covariance between the two at epochs k and $k - s$, where s is the lag time, is given by Priestley (1981) as

$$R_{1,2}(s) = E\{x_{1,k} - E\{x_1\}, x_{2,k-s} - E\{x_2\}\} \tag{5.33}$$

It can be seen that this closely resembles the equation for autocovariance, given by Eq. (5.27). In practical terms the cross-covariance is calculated using

$$R_{1,2}(s) = \frac{1}{N} \sum_{k=1}^{N-s} (x_{1,k} - \bar{x}_1)(x_{2,k-s} - \bar{x}_2) \tag{5.34}$$

where N is sample size. This can be extended to the case where there are i time series, $x_1(k), \ldots, x_i(k)$. The cross-covariance matrix of lag s for i time series is

defined as

$$
R_{1,\dots,i}(s) = \begin{pmatrix}
R_{1,1}(s) & R_{1,2}(s) & \dots & R_{1,i}(s) \\
R_{2,1}(s) & R_{2,2}(s) & \dots & R_{2,i}(s) \\
\vdots & \vdots & \ddots & \vdots \\
R_{i,1}(s) & R_{i,2}(s) & \dots & R_{i,i}(s)
\end{pmatrix}
\tag{5.35}
$$

For the two time series $x_1(k)$ and $x_2(k)$ the cross-correlation function is given by

$$
\rho_{1,2}(s) = \frac{R_{1,2}(s)}{\sqrt{R_{1,1}(0)R_{2,2}(0)}}
\tag{5.36}
$$

and the cross-correlation matrix is similarly given by

$$
\rho_{1,\dots,i}(s) = \begin{pmatrix}
\rho_{1,1}(s) & \rho_{1,2}(s) & \dots & \rho_{1,i}(s) \\
\rho_{2,1}(s) & \rho_{2,2}(s) & \dots & \rho_{2,i}(s) \\
\vdots & \vdots & \ddots & \vdots \\
\rho_{i,1}(s) & \rho_{i,2}(s) & \dots & \rho_{i,i}(s)
\end{pmatrix}
\tag{5.37}
$$

When calculating the cross-correlation of the least squares residuals each satellite and observation type pair is considered to be made up of a time series of 240 values. In the case of GNSS processing the length of the vector $\hat{\underline{e}}$ is liable to change as satellites come in and out of view. If an observable has been rejected at a previous epoch then this could also alter the vector $\hat{\underline{e}}$. Therefore Eq. (5.34) is calculated using all common satellite and observation type pairs.

Since the kinematic ALBM dataset shows a greater degree of correlation than the static datasets it is of particular interest to see if this correlation extends into the cross-correlations. For each cross-covariance matrix a reference epoch must be chosen. To minimise the computational burden the end of the processing time is chosen as the reference epoch, since this is the point at which the filter will have sufficiently converged and be utilising the greatest amount of data.

For the reference epoch a vector of residuals was formed and a vector of residuals for the lagged epoch was also formed. This was done for the previous 240 epochs, referred to as the sample size. In the event that a satellite and observation type pair did not exist for all epochs then it was discarded from the calculation.
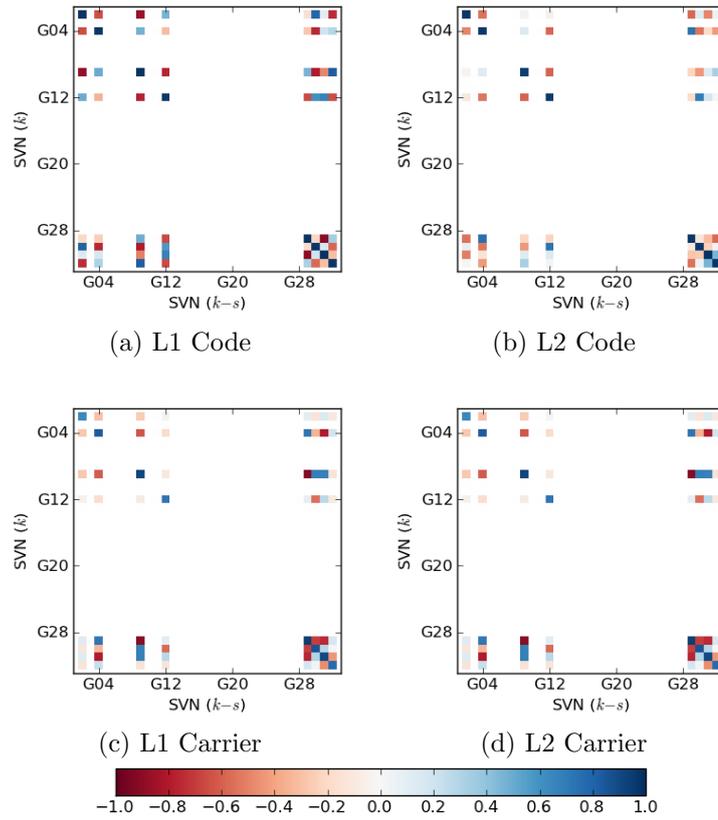
Figure 5.24: Lag 1 cross-correlation plot for ALBM dataset.

The cross-correlation plot at lag 1 is shown in Figure 5.24. It can be seen along the leading diagonal of the matrix that the autocorrelations correspond with the high autocorrelation values seen in §5.3.6. It is also of note that the off-diagonal values where there is a satellite are also very high. The difference in magnitude of the cross-correlations for the code residuals compared to the carrier phase residuals supports what has already been seen for the Durbin-Watson values and the auto-correlation plots. In some cases the cross-correlation between satellites is higher for some satellite and observation type pairs compared to others. Since the observation types have been separated it is not possible from these figures to see the cross-correlation between different observation types.

A numerical summary of values in Figures 5.24 - 5.28 is shown in Table 5.3. It is clear from the minima and maxima that there is a very high degree of correlation present in the residuals even at a lag time of 30 seconds. The mean values are non-zero, but close to zero, which is to be expected as the cross-correlation matrices are not symmetrical for lags greater than zero. The standard error is provided as

(a) L1 Code   (b) L2 Code

(c) L1 Carrier   (d) L2 Carrier
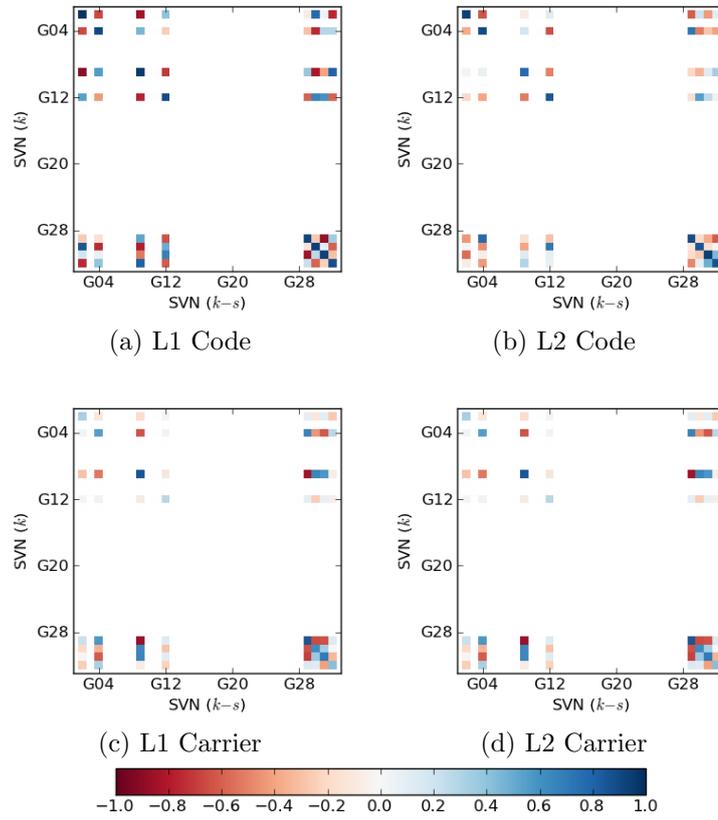
Figure 5.25: Lag 5 cross-correlation plot for ALBM dataset.

(a) L1 Code   (b) L2 Code

(c) L1 Carrier   (d) L2 Carrier
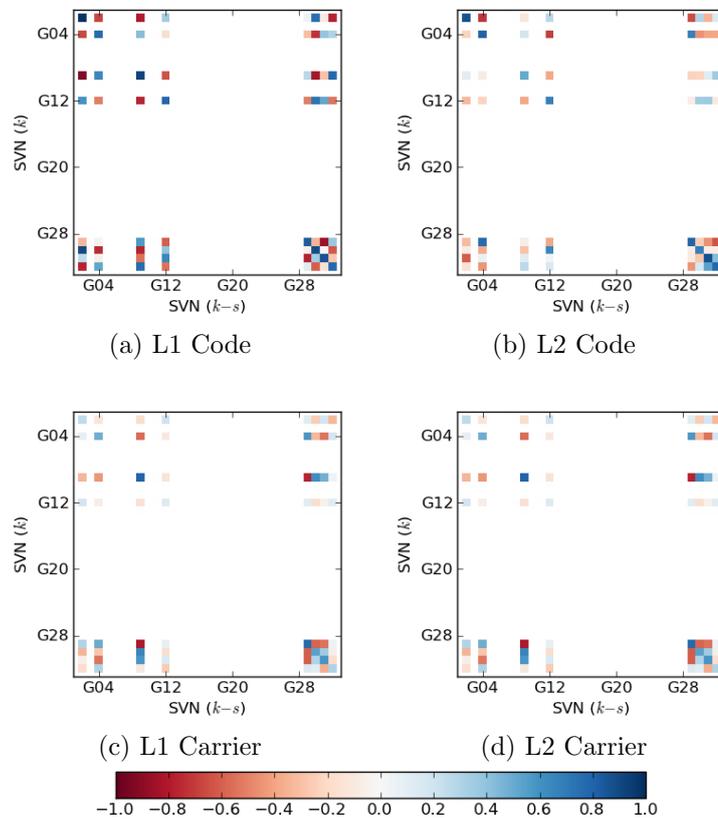
Figure 5.26: Lag 10 cross-correlation plot for ALBM dataset.
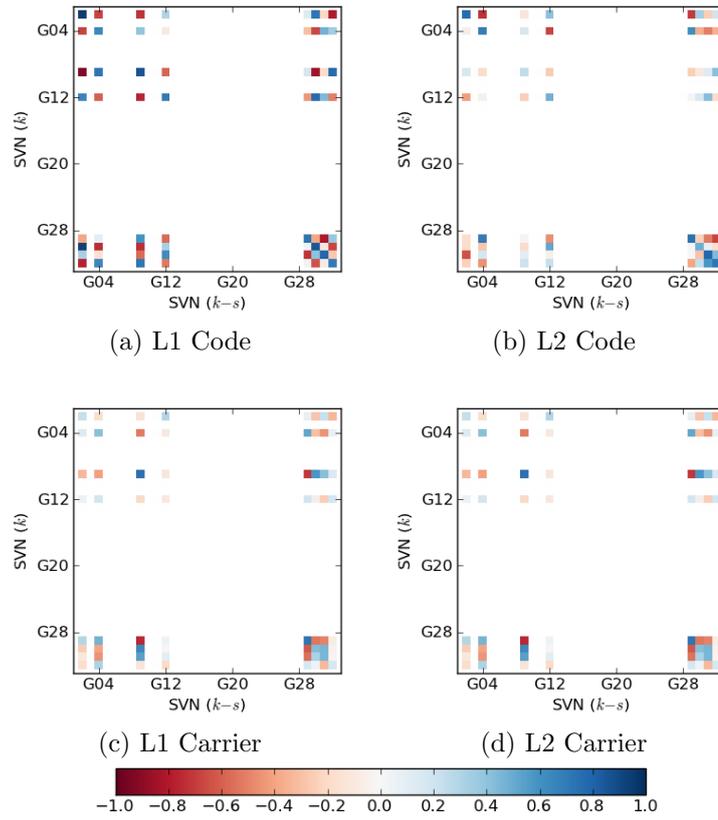
Figure 5.27: Lag 15 cross-correlation plot for ALBM dataset.
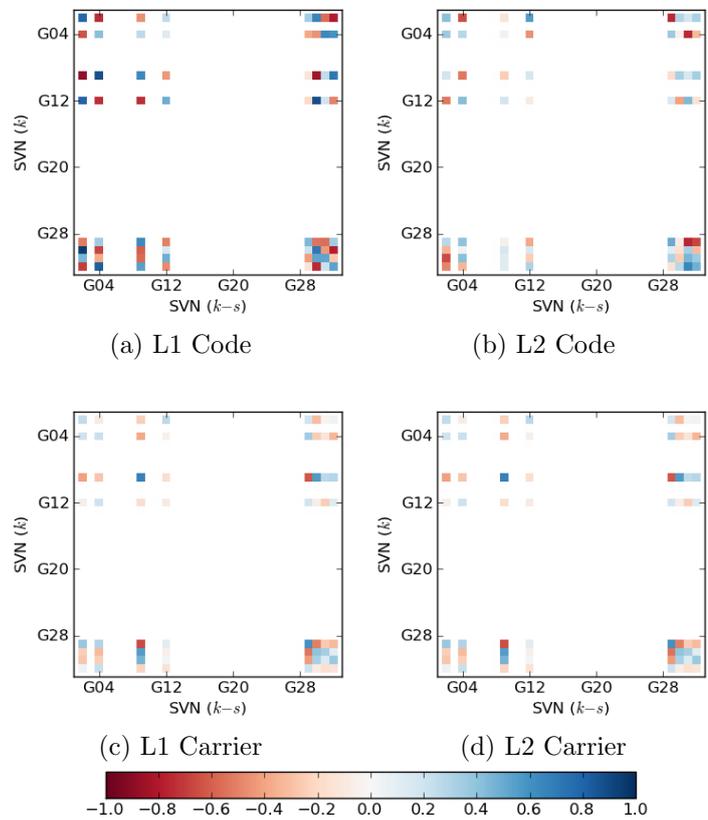
Figure 5.28: Lag 30 cross-correlation plot for ALBM dataset.

Table 5.3: Minimum, maximum, mean, standard error and standard deviation of cross-correlation values for all satellites and observation types for ALBM dataset at lags 1, 5, 10, 15 and 30.

Lag	Observation type	Min	Max	Mean	St. err.	St. dev.
1	L1 Code	-0.9024	1.0000	0.0073	0.0201	0.6416
	L2 Code	-0.5898	0.9914	0.0229	0.0157	0.5008
	L1 & L2 Carrier	-0.9194	0.9494	0.0084	0.0159	0.5098
	All observations	-0.9494	1.0000	0.0021	0.0036	0.4647
5	L1 Code	-0.9133	1.0000	0.0064	0.0196	0.6268
	L2 Code	-0.6491	0.9501	0.0200	0.0147	0.4695
	L1 & L2 Carrier	-0.8696	0.8670	0.0022	0.0134	0.4292
	All observations	-0.9133	1.0000	0.0018	0.0034	0.4340
10	L1 Code	-0.9352	0.9994	0.0055	0.0191	0.6097
	L2 Code	-0.7190	0.8824	0.0175	0.0136	0.4348
	L1 & L2 Carrier	-0.8130	0.8086	0.0022	0.0121	0.3880
	All observations	-0.9352	0.9994	0.0016	0.0032	0.4138
15	L1 Code	-0.9358	0.9613	0.0044	0.0187	0.5979
	L2 Code	-0.7406	0.7899	0.0146	0.0127	0.4065
	L1 & L2 Carrier	-0.7614	0.7526	0.0012	0.0113	0.3622
	All observations	-0.9358	0.9613	0.0012	0.0031	0.3994
30	L1 Code	-0.8945	0.9593	0.0018	0.0178	0.5700
	L2 Code	-0.7634	0.6551	0.0086	0.0117	0.3737
	L1 & L2 Carrier	-0.6568	0.6853	0.0008	0.0097	0.3100
	All observations	-0.8945	0.9593	0.0003	0.0029	0.3761

a quality measure for the mean, showing the error margin to be relatively small. The standard deviations show that the spread of values is quite large, although this appears greater for the L1 code values compared to the L2 code and L1 & L2 carrier phase values. In some cases a maximum value of 1.000 is given. This is an artefact of numerical rounding and these values are likely to be very close, but not equal, to unity.

It is expected that as the lag time increases the cross-correlation between satellite and observation type pairs should decrease. Figures 5.25 - 5.28 show the cross-correlation plots at lags 5, 10, 15 and 30 respectively. It can be seen that the cross-correlation of carrier phase residuals is at a much lower level by lag 30. However, this is not the case for the code observations. This appears to be particularly the case for the L1 Code residuals.

The cross-correlation plots show that not only are the post-fit residuals correlated with themselves, but there is a high degree of correlation between the residuals of one satellite and observation type pair and a different one at a lagged epoch. This

degree of correlation is hence contrary to the condition that $E\{\underline{e}_k, \underline{e}_l^T\} = 0$ for $k \neq l$ that is required by the Kalman filter and the test statistic described in Chapter 4.

5.3.8 Discussion

In this section the different methods of measuring correlation have been in broad agreement with each other. All datasets showed evidence of time correlation, although some sites were more correlated than others. In general, more time correlation is seen in the residuals from the 'kinematic' ALBM dataset than the three 'static' sites (ABPO, FALK and GUAM). However, whereas the carrier phase residuals are more correlated than the code residuals for the 'static' datasets, the opposite is true for the 'kinematic' dataset. There is insufficient data to determine whether in general terms the correlations are greater for kinematic data than for static data. This could be the case, since in a kinematic environment the environment is continually changing.

One main distinction between the 'static' and 'kinematic' datasets is that the former were all collected using Ashtech receivers (see Table 5.1), whereas the ALBM dataset was collected using a Leica receiver. It is very likely that the difference in results seen is related to the different equipment used. It is likely that the Leica receiver utilises carrier-aided smoothing of the code observations, which would explain the greater degree of time correlation seen for 'kinematic' dataset. Time correlation seen in the carrier phase residuals at all sites may be attributable to hardware, such as tracking loop errors. Hence the use of the same receiver model will be important. Bona (2000) has shown that correlations between observation types are receiver dependent. In addition, the 'static' sites use variations on the same antenna model, whereas the 'kinematic' dataset uses an antenna by a completely different manufacturer. These two factors combined will mean that similar behaviour should be expected from the 'static' sites.

Examining the environment around the antennas, a further contributing factor is that for the 'kinematic' dataset the antenna is mounted on the top of the metal roof of a vehicle. This has an increased chance of multipath. By comparison the three

'static' sites are IGS reference sites, where measures will be taken to minimise the effects of multipath. As mentioned in §2.2.5 multipath can be significant contributor to time correlation.

Additionally, the same precise orbits and satellite clocks were used to process the three 'static' sites. Any time correlations that may occur in the orbits and clocks will be applied to all three sites at the same time, provided the satellites are tracked simultaneously. Although the ALBM dataset uses precise orbits and satellite clocks from a different day the source is the same.

5.4   Methods for dealing with correlation

It is clear from §5.3 that the post-fit residuals from the PPP software are time correlated, although to differing extents. In the case of the static datasets the code observations appear to be much less correlated than those in the kinematic dataset. In all datasets the carrier phase observations appear to be correlated, although to a greater extent in the kinematic dataset.

It can therefore be said that the recursive estimator is non-optimal, since it is a condition of the Kalman filter that $E\{\underline{v}_k, \underline{v}_l^T\} = 0$ for $k \neq l$ and similarly for the semi-recursive Kalman filter that $E\{\hat{\underline{e}}_k, \hat{\underline{e}}_l^T\} = 0$ for $k \neq l$. It was shown in Chapter 4 that the recursive quality control procedures are also derived on the assumption that $E\{\underline{v}_k, \underline{v}_l^T\} = 0$ for $k \neq l$. It can therefore be deduced that not only is the filter non-optimal, but the quality control procedures being used can no longer be assumed to be valid.

At first glance the obvious approach would be to re-examine the observational model to ensure that the parameters being estimated are modelling the physical properties correctly. In the case where a time dependent parameter is not being estimated then this could be reflected in the residuals. However, it has been seen in §3.8 that a reparameterisation is required in GNSS PPP processing as there are already too many parameters in relation to the number of observables. It is therefore simply impractical to extend the number of estimated parameters further.

It should also be noted that the precise orbits and clocks are themselves stochastic

and, as mentioned in §7.5, small residual orbit and clock biases will result. Although the effect of orbit and clock corrections will be different for each satellite, they will be the same for all observation types of a given satellite. Any time correlations in the orbit and clock corrections will therefore manifest themselves in the residuals for observations from the same satellite.

Mehra (1970) shows that the correlation in the residuals can be caused by incorrectly specifying the a priori variance-covariance matrix of the observables and the variance-covariance matrix that controls the process noise during the prediction stage of the Kalman filter, given by Q_{y_k} and Q_{w_k} (see Chapter 3). Q_{y_k} is generally set as a diagonal matrix, which implies that the observables are not correlated. This has shown not to be the case in many GNSS receivers (Teunissen et al. 1998, Bona 2000, Schön and Brunner 2008, Amiri-Simkooei et al. 2009). It would seem that one solution would be to specify these off-diagonal elements. This is not as easy as it may first seem, since the cross-correlation of different observation types is receiver specific. Since Q_{y_k} is inverted to form a weight matrix for least squares estimation (and hence also in recursive filtering) this matrix must also be invertible.

The correct specification of Q_{w_k} is a much more difficult process and in many cases can be dependent on the application. For example, considering the common parameters in the semi-recursive Kalman filter given in Table (3.1) the tropospheric and hardware delays must both be specified in Q_{w_k} as part of the prediction step of the filter. The incorrect specification of either of these values will have the effect of under or over constraining the solution. In the case of the troposphere, this value can be site specific. By taking time correlation as a measure of optimality it is possible to find the 'best' solution by minimising the correlation through varying the process noise in the filter. This will be discussed in more detail in §5.4.1.

An alternative standpoint is to accept that the time correlation is present in the residuals and that it is not possible to remove this through tuning of the Kalman filter. This can be considered an acceptable solution, since it has already been discussed that adding parameters to the state vector is often not practical and in many cases a reparameterisation must be done in order to make the system

solvable. This alternative is briefly introduced in §5.4.2 and then covered in detail in Chapter 6.

5.4.1 Adjusting the process noise of common parameters

Taking the principle of an adaptive Kalman filter, such as the approach taken by Mehra (1970) and Oussalah and Schutter (2001), it may be possible to reduce the time correlation in the residuals by adapting the process noise in the Kalman filter based on estimates of the time correlation. A study was done whereby Q_{y_k} was held constant and the values used in Q_{w_k} adjusted with the aim of investigating the effect of time correlation on the post-fit residuals. In the E-HP/PPP software the values in Q_{w_k} correspond to the tropospheric delay and the hardware delays, which include the ambiguities (see §3.8). The values in Q_{w_k} are calculated based on the spectral densities given for each parameter (see §3.8). The spectral density for the hardware delays was varied and for the tropospheric delay held fixed. In §5.3.7 the cross-correlation matrix was presented as a method of showing how the residuals between different satellite and observation type pairs were correlated. These matrices are used as a measure of the correlation in the set of residuals as a whole and hence used to find the best spectral densities.

Since it is difficult to use a matrix in its entirety the matrix norm will be used as a measure. Four types of matrix norm will be used in order to measure the correlation (Fox 1964, Demmel 1997).

1-norm. This is defined as the largest column sum of absolute values, i.e.

$$\|A\|_1 = \max_j \sum_{i=1}^{m} |a_{ij}| \tag{5.38}$$

where m is equal to the number of rows in matrix A.

Infinity norm. This is defined as the largest row sum of absolute values, i.e.

$$\|A\|_\infty = \max_i \sum_{j=1}^{n} |a_{ij}| \tag{5.39}$$

where n is equal to the number of columns in matrix A.

2-norm. This is defined given by

$$\|A\|_2 = \sqrt{\max\{\Lambda(A^T A)\}} \tag{5.40}$$

where $\Lambda(A^T A)$ denotes the set of eigenvalues of $(A^T A)$.

Frobenius norm. This entrywise norm flattens the matrix into a vector and then uses a vector norm operation. It is calculated using

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2} \tag{5.41}$$

where m and n are equal to the number of rows and columns in matrix A respectively.

The ALBM dataset was processed using spectral densities for the hardware delay ranging from 1×10^{-11} m²s⁻¹ to 1×10^{-8} m²s⁻¹ in steps of 2×10^{-11} m²s⁻¹. The spectral densities for the tropospheric delay were held fixed. In each case a cross-correlation matrix was calculated for lags $\{0, \ldots, 15\}$. For each lag the four matrix norms listed above were calculated for each process noise value and then plotted with the minimum correlation along with the value of its corresponding process noise.

The matrix norms of the cross-correlation matrix for lag 1 are shown in Figure 5.29. The 1-norm and infinite norm place the optimal process noise to be 4.3×10^{-10} m²s⁻¹ and 3.7×10^{-10} m²s⁻¹ respectively. These values are relatively close, which is intuitive since the former is a column sum of the absolute values and the latter is a row sum. For lag 0 the cross-correlation matrix is symmetrical, i.e. $\rho(a) = \rho(-a)$. As the lag increases the matrix no longer becomes symmetrical as $E\{\hat{\underline{e}}^s_{k,i}, \hat{\underline{e}}^s_{k-s,i}\} \neq E\{\hat{\underline{e}}^s_{k-s,i}, \hat{\underline{e}}^s_{k,i}\}$. The minima of the 1-norm and infinite norm are very well defined, both showing a distinctive 'knife-edge' shape.

The spectral densities given by the 2-norm and the Frobenius norm are similarly in close agreement, giving values of 9.9×10^{-10} m²s⁻¹ and 1.17×10^{-9} m²s⁻¹ respectively. With contrast to the 1-norm and infinite norm, the 2-norm and Frobenius norm both have smoothed minima owing to the squaring operation that is present in both methods. These values are very close to the default of 1×10^{-9} m²s⁻¹ used by the E-HP/PPP software.
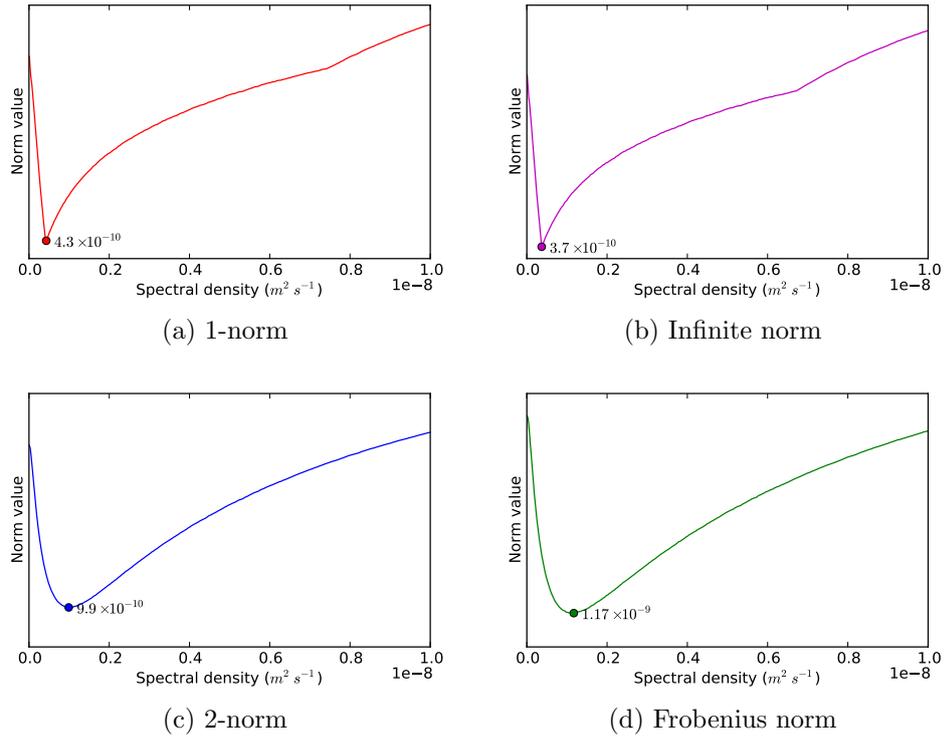
Figure 5.29: Norm of the cross-correlation matrix for lag 1 for different spectral densities. In each case the minimum norm is shown along with the equivalent process noise value.

Figures 5.30 - 5.33 show the matrix norms of the cross-correlation matrices for lags 5, 10, 15 and 30. As the lag time increases the 1-norm and infinite norm take on very different shapes and the minima vary considerably. By contrast, the curves for the 2-norm and Frobenius norm take the same shape and the minima are not as changeable. The two norms also give very close values for the best process noise and do not change by more than 1×10^{-9} m^2s^{-1}.

By taking the optimum process noise value from lag 1 and computing the matrix norm for every other lag the expectation is that the norm value should decrease as the size of the lag increases. This is supported by Figure 5.34, which shows the magnitude of cross-correlation as computed by the Frobenius norm decreases as the lag time increases. In this case the Frobenius norm was chosen as Figures 5.30 - 5.33 show it gives the most consistent results.

From the results above it can be concluded that whilst it is possible to minimise the correlation seen in the residuals by adjusting the process noise that is applied in the prediction step of the filter it is not possible to remove it. Even by minimising

(a) 1-norm   (b) Infinite norm

(c) 2-norm   (d) Frobenius norm
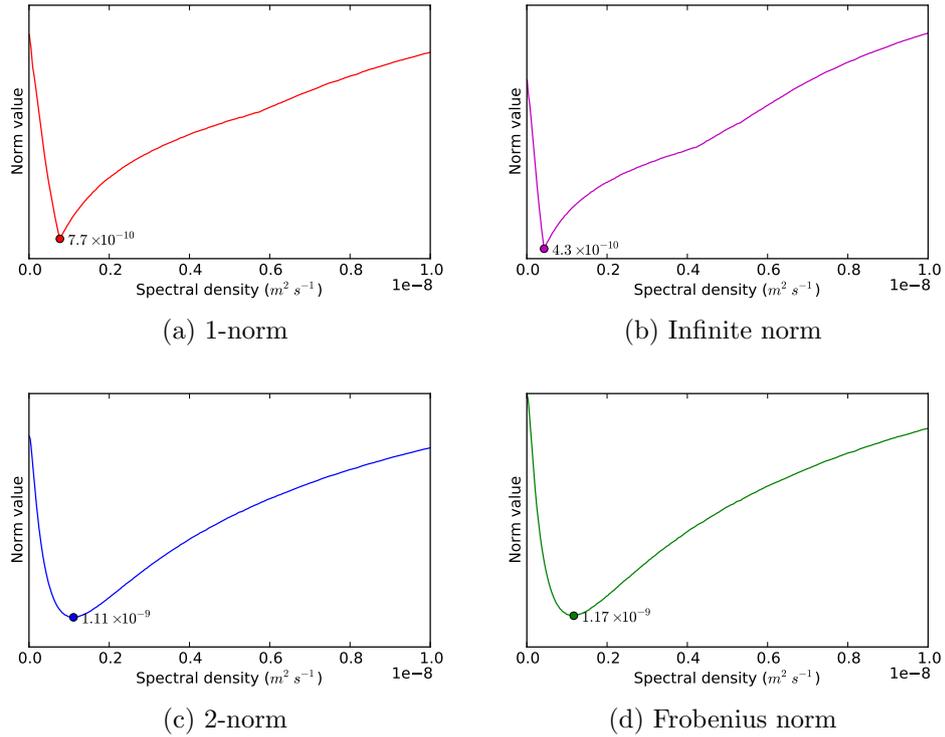
Figure 5.30: Norm of the cross-correlation matrix for lag 5 for different spectral densities. In each case the minimum norm is shown along with the equivalent process noise value.

(a) 1-norm   (b) Infinite norm

(c) 2-norm   (d) Frobenius norm
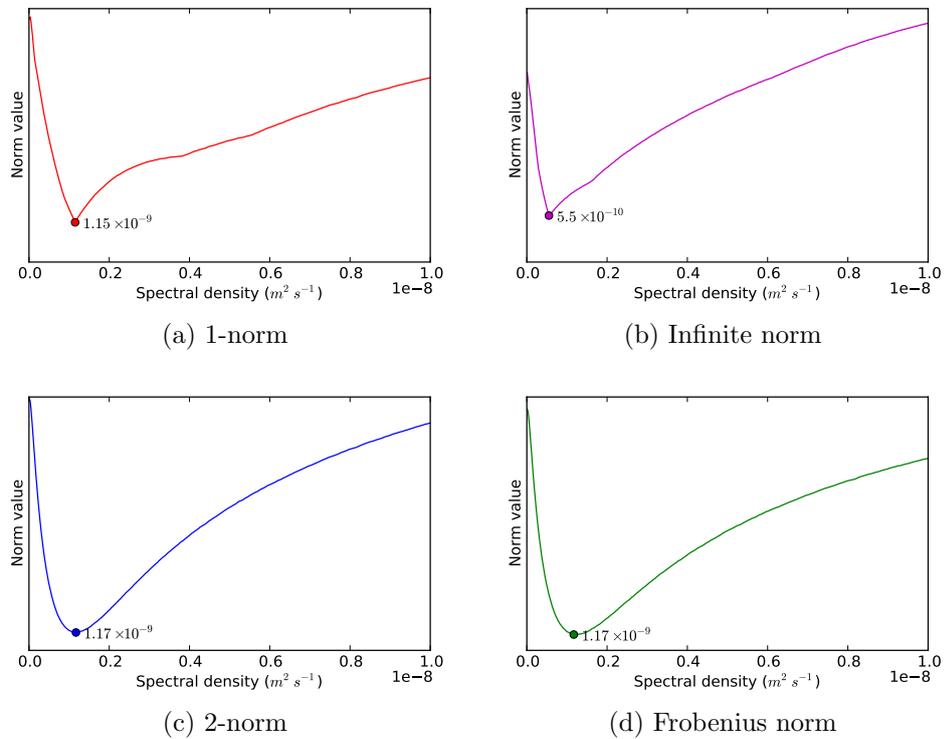
Figure 5.31: Norm of the cross-correlation matrix for lag 10 for different spectral densities. In each case the minimum norm is shown along with the equivalent process noise value.

(a) 1-norm   (b) Infinite norm

(c) 2-norm   (d) Frobenius norm
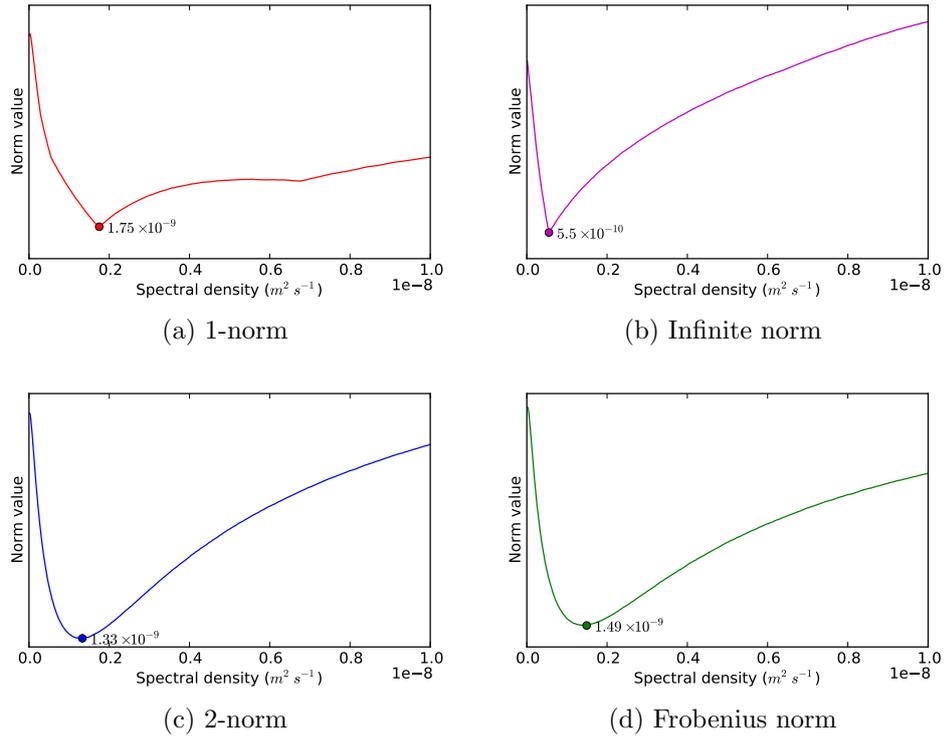
Figure 5.32: Norm of the cross-correlation matrix for lag 15 for different spectral densities. In each case the minimum norm is shown along with the equivalent process noise value.

(a) 1-norm   (b) Infinite norm

(c) 2-norm   (d) Frobenius norm
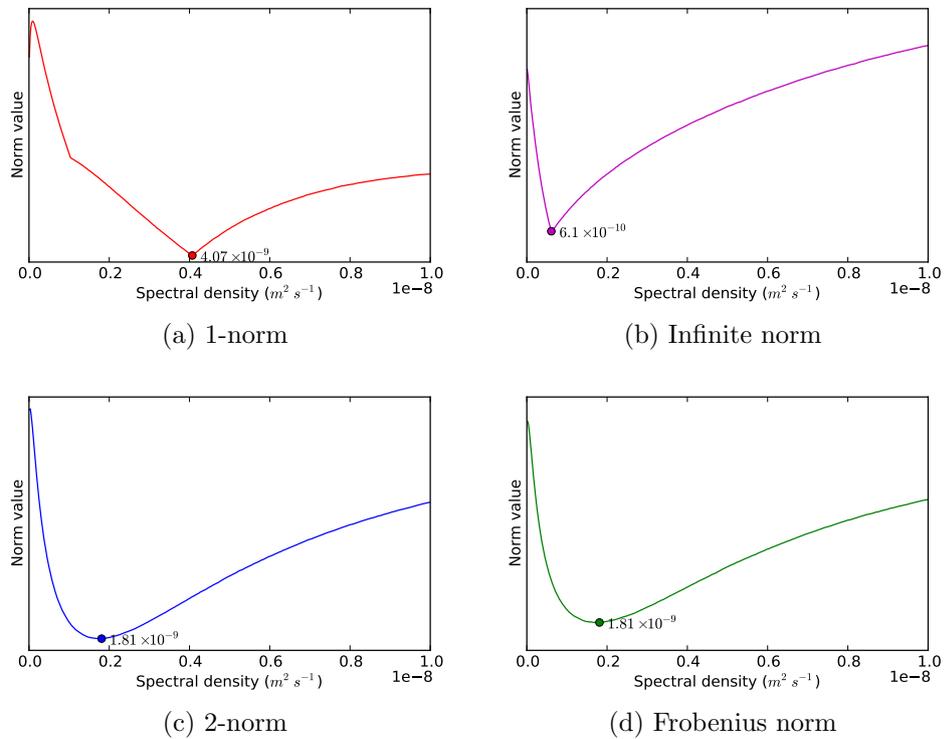
Figure 5.33: Norm of the cross-correlation matrix for lag 30 for different spectral densities. In each case the minimum norm is shown along with the equivalent process noise value.
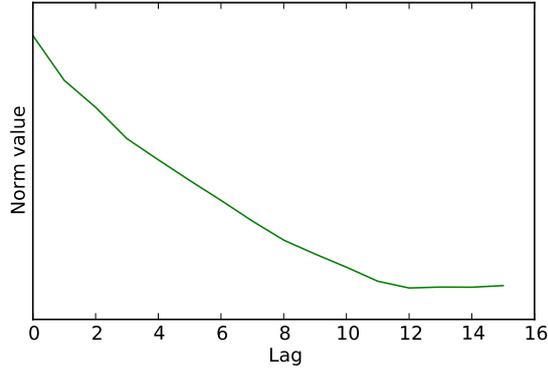
Figure 5.34: Relationship between the magnitude of cross-correlation as computed by the Frobenius norm and lag time.

the time correlation, the filter will still be non-optimal. Although this may not influence the parameters, it will still have an impact on the residuals and hence have a large impact on the validation of the data. It is therefore not feasible to implement an adaptive Kalman filter based on temporal correlations. However, the method used in this section can be used as an effective tool for determining the best spectral densities to form Q_{w_k} for use in a filter. It was also seen that the spectral densities that were inferred from the minima of the 2-norms and Frobenius norms closely matched the existing spectral densities that had been used in the software.

5.4.2 Autoregression

If it assumed that it is not possible to make any more adjustments to the filter that result in a significant reduction in the correlation of the residuals then in order to test for biases in the estimated parameters it is necessary to adapt the testing procedure to reflect this. The tests outlined in Chapter 4 are conditional on the residuals exhibiting no time correlation. It is possible to estimate the time correlation in the residuals using an autoregressive model. This is introduced in Chapter 6. By removing the correlations through a process known as whitening it is possible to generate a whitened test statistic, which is shown in Chapter 7.

5.5   Summary

In this chapter the mathematical difference between an optimal and non-optimal filter was presented, showing that where a Kalman filter is non-optimal the predicted residuals exhibit time correlation. Since it has already been shown in §3.7 that there is a linear relationship between the predicted and post-fit residuals, this can be also said to be the case for the post-fit residuals in a recursive estimator.

Using four high-rate datasets, of which three were collected from static receivers and one from one that was kinematic, through the use of the Durbin-Watson test statistic, lag plots, autocorrelation and cross-correlation plots it was shown that over short periods the residuals experience a high degree of correlation. The extent of the correlation was dependent on the observation type and varied between datasets. This could be attributable to the different receivers used, site dependent factors such as multipath or residual orbit and clock errors.

Following a discussion of the approaches that can be taken to eliminate or reduce the correlation, and hence arrive at an optimal filter, two approaches were suggested. The first involved empirical testing of the process noise values to minimise the correlation and hence find an optimal value which can be used in the prediction step of the recursive estimator. Although effective at reducing the correlation, not to an extent to which the correlation was reduced to a small enough extent to have no impact on the residuals. It would therefore not be effective to implement any kind of adaptive Kalman filter to achieve an optimal filter. The best value that was arrived at was also very close to the value that was used initially, hence showing that the filter was already using process noise values that resulted in the least temporal correlation.

Finally, the use of autoregression to remove the time correlation seen in the residuals and hence form a test statistic that satisfies the conditions set out in Chapter 4 was proposed. This is to be explored further in Chapters 6 and 7.

Chapter 6

Modelling time correlated residuals using autoregression

6.1 Introduction

From Chapter 3 it is known that one of the conditions of an optimal Kalman filter is that the predicted residuals should not be correlated in time. Since there is a linear relationship between the predicted and post-fit residual, as proved in §3.7, then it follows that the post-fit residuals should also not be correlated in time. In addition to this it is a requirement of the quality control theory presented in Chapter 4 that the residuals are temporally uncorrelated. In Chapter 5 the post-fit residuals from our PPP software were analysed and found to display signs of time correlation.

Two methods for dealing with time correlated residuals were suggested in §5.4. The second of these methods was to model the residual time series as an autoregressive (AR) process and then use the estimated coefficients to remove the correlation from the residual. Contained within this chapter is an overview of stationarity in §6.2, followed by the theory of autoregression in §6.3. An overview of the different methods for estimating the AR coefficients is given in §6.4 and criteria by which the model order can be chosen subsequently in §6.5. In §6.6 the theory relating to non-optimal filters developed in the previous chapter is developed to show that it is theoretically possible for the predicted and post-fit residuals to be modelled as an AR process.

Initially the post-fit residuals are modelled as an AR(1) process in §6.7, as this is the simplest model to implement. This initial implementation is also used to consider the different AR estimation methods that are available. The residuals are 'whitened' and then tested to see if they can be truly said to be random. This is done using the methods of §5.3.

This implementation is then extended in §6.8 so that a higher model order can be selected. A criterion from §6.5 is used to automatically determine the model order that is most suitable. Once again the residuals are then whitened and then tested to see if the whitening process has been successful. Model order selection is then discussed in §6.9.

Since the two implementations in §6.7 and §6.8 use a univariate AR model, the multivariate model is introduced in §6.10 and the implementation issues are discussed.

6.2 Stationarity

A stochastic process is said to be *wide sense stationary* when the following conditions are met (Box and Jenkins 1976, Priestley 1981, Broersen 2006, Dougherty 2011):

(i) The mean remains constant for the whole time period of the process, i.e.

$$E\{x(k)\} = \mu_x, \text{ for all } k \tag{6.1}$$

where μ_x is the mean of $x(k)$.

(ii) The variance remains constant for the whole time period of the process, i.e.

$$E\{x_k - \mu_x, x_k - \mu_x\} = \sigma_x^2, \text{ for all } k \tag{6.2}$$

where σ_x^2 is variance of $x(k)$.

(iii) The covariance between any two epochs is said to be dependent only on the

lag time and not the epochs themselves, i.e.

$$E\{x_k - \mu_x, x_{k-s}^T - \mu_x\} = R(s) = R(-s), \text{ for all } k \tag{6.3}$$

where $R(s)$ is the autocovariance matrix for lag s (see §5.3.6).

A wide sense stationary process is also referred to as *stationarity up to order two* (Broersen 2006). A process in a dynamic system generally reaches stationarity when the system is said to be stable and has achieved a steady state.

In order to explore the concept of stationarity further it is helpful to look at the linear filter model as defined by Box and Jenkins (1976). The model takes a weighted sum of the random noise from previous epochs, such that for the process $x(k)$ at epoch k

$$x_k = \mu_x + \underline{\varepsilon}_k + \psi_1 \underline{\varepsilon}_{k-1} + \psi_2 \underline{\varepsilon}_{k-2} + \cdots \tag{6.4}$$

where μ_x is the mean of the process if it is stationary (or a reference point for the level of the process if it is not), $\psi_1, \psi_2, \ldots$ are a series of weights and $\underline{\varepsilon}_k, \underline{\varepsilon}_{k-1}, \underline{\varepsilon}_{k-2}, \ldots$ represent random process noise at previous epochs. Using the backward shift operator function, B, defined by

$$B\underline{\varepsilon}_k = \underline{\varepsilon}_{k-1} \qquad B^2\underline{\varepsilon}_k = \underline{\varepsilon}_{k-2} \qquad \cdots \qquad B^p\underline{\varepsilon}_k = \underline{\varepsilon}_{k-p} \tag{6.5}$$

and Eq. (6.1), the invariance of the mean with respect to k, the process $x(k)$ can be expressed as

$$x_k = \mu_x + \underline{\varepsilon}_k + \psi_1 B\underline{\varepsilon}_k + \psi_2 B^2 \underline{\varepsilon}_k + \cdots \tag{6.6}$$

$$= \mu_x + (1 + \psi_1 B + \psi_2 B^2 + \cdots)\underline{\varepsilon}_k \tag{6.7}$$

By defining the transfer function

$$\psi(B) = 1 + \psi_1 B + \psi_2 B^2 + \cdots \tag{6.8}$$

the equation can be expressed more concisely as

$$x_k = \mu_x + \psi(B)\underline{\varepsilon}_k \tag{6.9}$$

If the series $\psi_1, \psi_2, \ldots$ is finite, or infinitely summable such that $\sum\limits_{j=0}^{\infty} |\psi_j| < \infty$, then the filter can be said to be stationary and the process $x(k)$ is stationary.

Since the mean of the stationary stochastic process is constant it is possible to express $x(k)$ as a function of deviations from the mean,

$$\bar{x}(k) = x(k) - \mu_k \tag{6.10}$$

Eq. (6.4) becomes

$$\bar{x}_k = \underline{\varepsilon}_k + \sum_{j=1}^{\infty} \psi_j \underline{\varepsilon}_{k-j} \tag{6.11}$$

with Eq. (6.9) becoming

$$x_k = \psi(B)\underline{\varepsilon}_k \tag{6.12}$$

It will now be shown that $\bar{x}_k$ can also be expressed as a weighted sum of past values and the random process noise at the current epoch. Assume

$$\bar{x}_k = \pi_1 \bar{x}_{k-1} + \pi_2 \bar{x}_{k-2} + \cdots + \underline{\varepsilon}_k \tag{6.13}$$

$$= \sum_{j=1}^{k-1} \pi_j \bar{x}_{k-j} + \underline{\varepsilon}_k \tag{6.14}$$

Once again, using the backward difference operator this can be expressed as

$$\pi(B)\bar{x}_k = \underline{\varepsilon}_k \tag{6.15}$$

where

$$\pi(B) = 1 - \pi_1 B - \pi_2 B^2 - \cdots \tag{6.16}$$

is known as the generating function of the π weights. On using Eq. (6.12) and Eq. (6.15)

$$\psi(B)\pi(B) = 1 \tag{6.17}$$

and hence

$$\pi(B) = \psi^{-1}(B) \tag{6.18}$$

This relationship is important when considering the stationarity of an autoregressive process (see §6.3). The coefficients in $\pi(B)$ of Eq. (6.18) establish Eq. (6.13).

Using the autocovariance function, as given by Box and Jenkins (1976),

$$\gamma_s = \sigma_\varepsilon^2 \sum_{j=0}^{\infty} \psi_j \psi_{j+s} \tag{6.19}$$

and setting $s = 0$ it follows that

$$\gamma_0 = \sigma_\varepsilon^2 \sum_{j=0}^{\infty} \psi_j^2 = \sigma_x^2 \tag{6.20}$$

From this the condition that the series $\psi_1, \psi_2, \ldots$ is finite, or infinitely summable such that $\sum_{j=0}^{\infty} |\psi_j| < \infty$, also implies that the summation in Eq. (6.20) converges and that the process has a finite variance.

In the derivations given by both Box and Jenkins (1976) and Priestley (1981) the generating function, B, can take on complex values. For a complex number $z = x + yi$ an absolute value is the Euclidean distance of z to 0 in the complex plane, i.e. $|z| = \sqrt{x^2 + y^2}$. All points in the complex plane where $|z| = 1$ form what is known as the unit circle, as shown in Figure 6.1. It can be seen that all points on the red line have a distance of 1 from the origin. In the case where $|B| < 1$, $|B| = 1$ and $|B| > 1$ the complex number B is said to lie inside, on or outside the unit circle respectively.

The question arises whether the residual time series seen in Chapter 5 can be said to be stationary. From Figure 5.3 - 5.6 it is clear that the residuals have a constant mean, which is intuitive since $E\{\hat{e}_k\} = 0$. Since elevation dependent weighting is used for the observables, the magnitude of the residuals is much higher at the beginning and end of a satellite pass and so strictly speaking the variance of the time series of residuals cannot be said to be constant. Broersen (2006) states that even if a whole time series may appear nonstationary, over a small interval a time series may appear stationary.
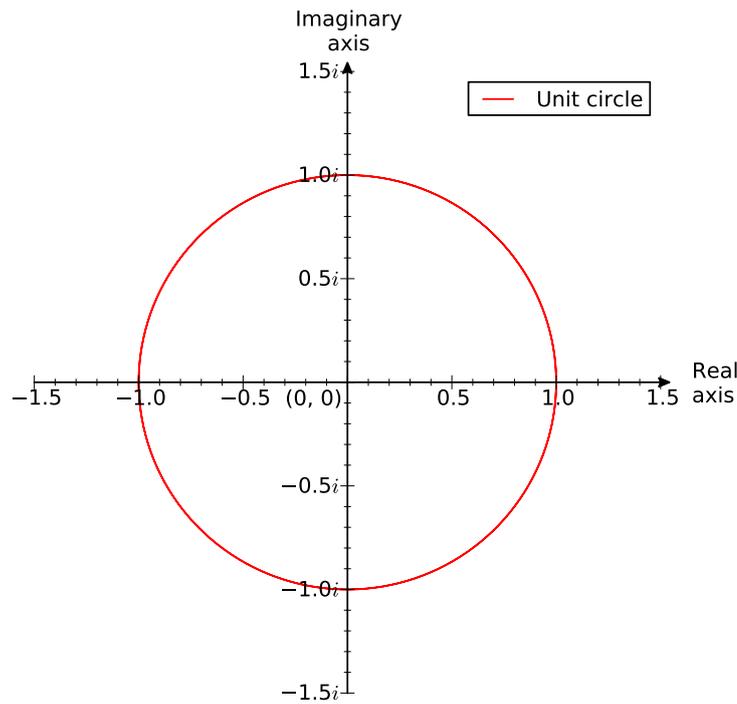
Figure 6.1: Unit circle

This theory of stationarity over short time intervals is applied to the post-fit residuals, in that the correlation seen at one epoch will be characterised by a sample of the previous N epochs. This has already been applied in §5.3.5 for calculating the Durbin-Watson test statistic, in §5.3.6 for plotting autocorrelation and in §5.3.7 for creating cross-correlation matrices. For these applications a sample size of $N = 240$ was used (see §5.3.3). This value will also be used for the autoregressive parameter estimation.

6.3   General autoregressive model

There are two equivalent models for the general autoregressive model given in the literature and the references given in this chapter do not all use the same autoregressive model. For the purposes of this study the two models have therefore been given the following names so that they can be readily distinguishable.

(i) *Box-Jenkins model.* This model is used by Box and Jenkins (1976).

(ii) *Priestley model.* This model is used by Priestley (1981) and Broersen (2006).

Box-Jenkins model

An autoregressive (AR) process of order p, commonly referred to as an AR(p) process, is given by Box and Jenkins (1976) as

$$\bar{x}_k = \phi_1 \bar{x}_{k-1} + \cdots + \phi_p \bar{x}_{k-p} + \underline{\varepsilon}_k \tag{6.21}$$

where $\bar{x}(k)$ is a stationary stochastic process with zero mean (i.e. $\mu_x = 0$), $\phi_1, \cdots, \phi_p$ are a set of autoregressive parameters and $\underline{\varepsilon}_k$ is a purely random stationary process.

Using the backwards operator function (see Eq. 6.5) it is possible to write Eq. (6.21) as

$$(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p)\bar{x}_k = \underline{\varepsilon}_k \tag{6.22}$$

Taking the example of an AR(1) process,

$$(1 - \phi_1 B)\bar{x}_k = \underline{\varepsilon}_k \tag{6.23}$$

and hence

$$\bar{x}_k = (1 - \phi_1 B)^{-1} \underline{\varepsilon}_k = \sum_{j=0}^{\infty} \phi_1^j \underline{\varepsilon}_{k-j} \tag{6.24}$$

For general $\underline{\varepsilon}_k$, Eq. (6.24) will converge only if $|\phi_1| < 1$. Recalling Eq. (6.18) this can be expressed in terms of $\psi(B)$ giving

$$\psi(B) = (1 - \phi_1 B)^{-1} = \sum_{j=0}^{\infty} \phi_1^j B^j \tag{6.25}$$

As $\psi(B)$ must converge for $|B\phi_1| < 1$, the condition $|\phi_1| < 1$ implies $|B| \leq 1$. Note that the solution of $1 - \phi_1 B = 0$, i.e. $B = \phi_1^{-1}$, thus lies outside the unit circle.

For a general AR(p) process,

$$x_k = \Phi^{-1}(B)\underline{\varepsilon}_k \tag{6.26}$$

with

$$\Phi(B) = (1 - G_1 B)(1 - G_2 B) \cdots (1 - G_p B) \tag{6.27}$$

where $G_1^{-1}, \ldots, G_p^{-1}$ are the solutions of $\Phi(B) = 0$. In order for the AR(p) to be a stationary process these solutions of $\Phi(B) = 0$ must lie outside the unit circle.

Priestley model

The AR(p) process is given by Priestley (1981) and Broersen (2006) as

$$x_k + a_1 x_{k-1} + \cdots + a_p x_{k-p} = \underline{\varepsilon}_k \tag{6.28}$$

where $a_1, \cdots, a_p$ are a set of autoregressive parameters.

The two methods presented above are equivalent as $\phi_j = -a_j$, but result in autoregressive parameters of opposite sign.

Regardless of the model that is used an AR(p) process must have the following properties (Box and Jenkins 1976, Priestley 1981, Broersen 2006):

(i) $E\{\underline{\varepsilon}_k\} = 0$, i.e. zero mean.

(ii) $\text{cov}(\underline{x}_{k-q}, \underline{\varepsilon}_k) = 0 \quad q \geq 1$

(iii) The time series must be stationary, i.e. σ_ε^2 is constant for all k.

(iv) $E\{\underline{\varepsilon}_k, \underline{\varepsilon}_{k-q}\} = 0$ where $q \geq 1$.

All new derivations in this study will be expressed using the Box-Jenkins model as this is more intuitive when considering the process of residual whitening given in §6.7 and §6.8.

6.4 Estimation methods

In this section a variety of AR estimation methods are presented. In §6.4.1 the equations presented by Yule (1927), commonly known as the Yule-Walker equations, are given. These show the link between the autocovariances and the AR parameters. Levinson-Durbin recursion is introduced in §6.4.2 as a recursive method of solving the Yule-Walker equations. It is also possible to estimate the AR parameters using the least squares method. The forwards and backwards least squares methods are

described in §6.4.3. The final method presented in §6.4.4 is Burg's method, which is an extension of Levinson-Durbin recursion that also guarantees stationarity of the AR parameters. This will be treated in more detail as it is the chosen method for the AR estimation in this study.

6.4.1 Yule-Walker equations

For an AR process defined using the Box-Jenkins model (see Eq. 6.21) the Yule-Walker equations are given as (Box and Jenkins 1976)

$$
\begin{aligned}
\rho(1) &= & \phi_1 &+ & \phi_2\rho(1) &+ \cdots + & \phi_p\rho(p-1) \\
\rho(2) &= & \phi_1\rho(1) &+ & \phi_2 &+ \cdots + & \phi_p\rho(p-2) \\
&\vdots & \vdots & & \vdots & \ddots & \vdots \\
\rho(p) &= & \phi_1\rho(p-1) &+ & \phi_2\rho(p-2) &+ \cdots + & \phi_p
\end{aligned}
\tag{6.29}
$$

Expressed in matrix form the Yule-Walker equations become

$$
\rho = P\phi
\tag{6.30}
$$

where

$$
\rho = \begin{pmatrix} \rho(1) \\ \rho(2) \\ \vdots \\ \rho(p) \end{pmatrix}
\quad
P = \begin{pmatrix} \rho(0) & \rho(1) & \dots & \rho(p-1) \\ \rho(1) & \rho(0) & & \vdots \\ \vdots & & \ddots & \rho(1) \\ \rho(p-1) & \dots & \rho(1) & \rho(0) \end{pmatrix}
\quad
\phi = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_p \end{pmatrix}
\tag{6.31}
$$

with $\rho(s)$ the autocorrelation at lag s defined by Eq. (5.28). Note that the diagonal elements of P are equal to 1, since $\rho(0) = 1$. An equivalent realisation of the Yule-Walker equations can be made using the autocovariances.

$$
R\phi = r
\tag{6.32}
$$

where

$$R = \begin{pmatrix} R(0) & R(1) & \ldots & R(p-1) \\ R(1) & R(0) & & \vdots \\ \vdots & & \ddots & R(1) \\ R(p-1) & \ldots & R(1) & R(0) \end{pmatrix} \qquad r = \begin{pmatrix} R(1) \\ R(2) \\ \vdots \\ R(p) \end{pmatrix} \tag{6.33}$$

with $R(s)$ the autocovariance at lag s as defined by Eq. (5.27). It can readily be seen that Eq. (6.30) can be derived from Eq. (6.32) by diving through by $R(0)$.

In reality since the true autocovariances are not known their estimates must be used. Broersen (2006) gives the estimator for the covariance which guarantees the property of positive-definite as

$$\hat{R}(s) = \frac{1}{N} \sum_{i=1}^{N-p} x_i x_{i+s} \tag{6.34}$$

where N is the number of elements in the stochastic process $x(k)$.

From Eq. (6.32), and letting $R = \hat{R}$, it follows that the estimate of ϕ is given by

$$\hat{\phi} = \hat{R}^{-1} \hat{r} \tag{6.35}$$

The equivalent The Yule-Walker equations using the Priestley model definition of an AR process (see Eq. 6.28) are given by Priestley (1981). Using this model Eq. (6.30) becomes

$$Pa = -\rho \tag{6.36}$$

where

$$a = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{pmatrix} \tag{6.37}$$

with the definition of P and a as given in Eq. (6.31). The equivalent realisation of

the Yule-Walker equations using the autocovariances is given by

$$Ra = -r \tag{6.38}$$

with R and r as given in Eq. (6.33). The solution for a is hence given by

$$a = -R^{-1}r \tag{6.39}$$

Once again it can be seen that all solutions result in $a_1, \ldots, a_p$ being of opposite sign to $\phi_1, \ldots, \phi_p$.

The Yule-Walker equations give a one-step solution for the AR parameters, but it is noted by Box and Jenkins (1976) that they can lead to poor coefficient estimates when the roots of the AR process are close to the unit circle (see §6.2). It is also shown by Broersen (2006) that using the Yule-Walker equations can result in an AR estimate with a very large bias. It is for these reasons that the Yule-Walker method is considered unsuitable for data with unknown characteristics and as such is not recommended for automated processes.

6.4.2 Levinson-Durbin recursion

Solving the Yule-Walker equations by inverting the autocorrelation matrix or autocovariance matrix is typically not done in practice in preference of more numerically stable algorithms (Broersen 2006). The Levinson-Durbin algorithm is a recursive method that takes advantage of the partial autocorrelation function (Box and Jenkins 1976).

Consider the stochastic process $x(k)$. $\rho(1)$ is defined as the autocorrelation between x_k and x_{k-1} and $\rho(2)$ as the autocorrelation between x_k and x_{k-2}. The partial autocorrelation is described by Box and Jenkins (1976) and Priestley (1981) as the additional contribution of an AR order to the autocorrelation. Hence the partial autocorrelation between x_{k-1} and x_{k-2}, which shall be denoted as $\rho(k-1, k-2)$, is the difference between $\rho(2)$ (the autocorrelation between x_k and x_{k-2}) and the extrapolation of $\rho(2)$ from $\rho(1)$. More generally, the partial autocorrelation is described

by Priestley (1981) as the correlation between random variables of the process $x(k)$ at two time points with the linear relationship of the intermediate random variables removed.

This property of the partial autocorrelation function allows it to be used to describe the new contribution of AR order to the autocorrelation (Broersen 2006). Levinson-Durbin recursion starts by estimating ϕ_1, the AR(1) coefficient, and then using the aforementioned property of the partial autocorrelation function to sequentially calculate AR coefficients of increasing order.

To ensure it is clear for what order of AR process the coefficient signifies, $\phi_{p,n}$ will be the nth coefficient of an AR(p) process. The method given here is based on that of Broersen (2006), although it will be given in relation an AR process using the Box-Jenkins model.

The recursion is initialised by calculating ϕ_1 assuming an AR(1) process, which can easily be calculated from the Yule-Walker equations (see §6.4.1) as

$$\phi_{1,1} = \frac{\hat{R}(1)}{s_0^2} \tag{6.40}$$

where

$$s_0^2 = \hat{R}(0) \tag{6.41}$$

is the estimated variance of the stochastic process. The next step is to calculate $\phi_{2,1}$ and $\phi_{2,2}$. Priestley (1981) calculates the partial autocorrelation as

$$\phi_{2,2} = \frac{\hat{R}(2) + \hat{R}(1)\phi_{1,1}}{\hat{R}(0)} \tag{6.42}$$

which directly yields ϕ_2 for the AR(2) process. It is now possible to update $\phi_{1,1}$ with the partial autocorrelation to give $\phi_{2,1}$.

$$\phi_{2,1} = \phi_{1,1} + \phi_{2,2}\phi_{1,1} \tag{6.43}$$

For models $p > 2$ the method can be summarised by the following recursive formulae, where $n = 1, 2, \ldots, p-1$. It can be readily seen that the example given above can be

obtained by substituting $n = 2$ into the recursive formulae that follow. To simplify the presentation of the formulae the following vectors will be defined.

$$\alpha_n = \begin{pmatrix} \phi_{n,1} & \phi_{n,2} & \cdots & \phi_{n,n} \end{pmatrix}^T \qquad \alpha'_n = \begin{pmatrix} \phi_{n,n} & \cdots & \phi_{n,2} & \phi_{n,1} \end{pmatrix}^T$$

$$\beta_n = \begin{pmatrix} \hat{R}(1) & \cdots & \hat{R}(n-1) & \hat{R}(n-1) \end{pmatrix}^T \quad \beta'_n = \begin{pmatrix} \hat{R}(n) & \hat{R}(n-1) & \cdots & \hat{R}(1) \end{pmatrix}^T$$

$$(6.44)$$

It can be seen that vector α_n contains the AR coefficients up to order n, with α'_n being the same vector in reverse. Vector β_n contains the autocovariances from lag 1 to n, with β'_n similarly being the the same vector in reverse.

The partial correlation coefficient for the next order is calculated by Broersen (2006) using

$$\phi_{n+1,n+1} = \frac{\hat{R}(n+1) + \beta'^T_n \alpha_n}{s_n^2} \qquad (6.45)$$

The new AR coefficient for model order $n + 1$ is appended to the vector of AR coefficients and the coefficients for orders $1, \ldots, n$ are updated using

$$\alpha_{n+1} = \begin{pmatrix} \alpha_n \\ \phi_{n+1,n+1} \end{pmatrix} = \begin{pmatrix} \alpha_n + \phi_{n+1,n+1} \; \alpha'_n \\ \phi_{n+1,n+1} \end{pmatrix} \qquad (6.46)$$

Expanding this equation can show more clearly how the AR coefficients are updated using the partial correlation coefficient.

$$\begin{pmatrix} \phi_{n+1,1} \\ \phi_{n+1,2} \\ \vdots \\ \phi_{n+1,n} \\ \cdots\cdots \\ \phi_{n+1,n+1} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} \phi_{n,1} \\ \phi_{n,2} \\ \vdots \\ \phi_{n,n} \end{pmatrix} + \phi_{n+1,n+1} \begin{pmatrix} \phi_{n,n} \\ \vdots \\ \phi_{n,2} \\ \phi_{n,1} \end{pmatrix} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \phi_{n+1,n+1} \end{pmatrix} \qquad (6.47)$$

The estimated variance of the AR process for this order, $n + 1$, is found using

$$s_{n+1}^2 = s_n^2 (1 - \phi_{n+1,n+1}^2) \qquad (6.48)$$

An interesting by-product of this recursive algorithm is that the estimated variance for each successive model order, s_n^2, can be used for choosing the AR model order. More information on this is given in §6.5.

A summary of the Levinson-Durbin recursion algorithm is given below:

1. Begin with $p = 1$, i.e. an AR(1) process.

2. Calculate the autocovariance at lag 0 for the stochastic process, s_0^2, using Eq. (6.41)

3. Calculate the first partial correlation coefficient, $\phi_{1,1}$, using Eq. (6.40).

4. Calculate the estimated variance for the AR(1) process, $s_1^2 = s_0^2(1 - \phi_{1,1})$.

5. Calculate the partial correlation coefficient $\phi_{p+1,p+1}$ using Eq. (6.45).

6. Update all AR parameters using Eq. (6.46).

7. Calculate the estimated variance for AR order $p + 1$ using Eq. (6.48)

8. Update the AR order, i.e. $p = p + 1$. If the desired AR has not been reached repeat from Step 5.

It is important to note that the method that is given in Broersen (2006) is based on the AR process using the Priestley model. The partial correlation coefficient is therefore of opposite sign to the final AR coefficient of an AR(p) model, i.e. $a_p = -\phi_{p,p}$. Broersen (2006) instead calculates the reflection coefficient, κ_p, which is of opposite sign to the partial correlation coefficient, i.e. $\kappa_p = -\phi_{p,p}$. Therefore, for an AR process using the Priestley model the final AR coefficient is equal to the reflection coefficient, i.e. $\kappa_p = a_p$.

Since Levinson-Durbin recursion is a method for solving the Yule-Walker equations it also is susceptible to poor coefficient estimates when the AR process is close to the unit circle and again is considered unsuitable for data with unknown characteristics. It is therefore not considered suitable for this study.

6.4.3   Least squares estimation

It is possible to estimate the AR parameters using the least squares estimator by considering the 'observables' to be the samples from the stochastic process and the 'parameters' to be the AR coefficients that are to be estimated.

Consider a stochastic process $x(k)$ from which N samples have been made. Characterising this as an AR(1) process using the Box-Jenkins model gives

$$
\begin{aligned}
x_2 &= \phi_1 x_1 + \underline{\varepsilon}_2 \\
x_3 &= \phi_1 x_2 + \underline{\varepsilon}_3 \\
&\vdots \\
x_N &= \phi_1 x_{N-1} + \underline{\varepsilon}_N
\end{aligned}
\tag{6.49}
$$

Using the model of observation equations (see §3.2.1) the least squares model can be expressed as

$$
E\left\{
\begin{array}{c}
x_2 \\
x_3 \\
\vdots \\
x_N
\end{array}
\right\}
=
\left(
\begin{array}{c}
x_1 \\
x_2 \\
\vdots \\
x_{N-1}
\end{array}
\right)
\phi_1
\tag{6.50}
$$

As $E\{\varepsilon_k\} = 0$ for all k it disappears from the equation. An unweighted solution is computed, i.e. $Q_y = I$, and the solution is readily found using Eq. (3.4).

For a general AR(p) solution the model becomes

$$
E\left\{
\begin{array}{c}
x_{p+1} \\
x_{p+2} \\
x_{p+3} \\
\vdots \\
x_N
\end{array}
\right\}
=
\left(
\begin{array}{ccccc}
x_p & x_{p-1} & x_{p-2} & \cdots & x_1 \\
x_{p+1} & x_p & x_{p-1} & \cdots & x_2 \\
x_{p+2} & x_{p+1} & x_p & \cdots & x_3 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
x_{N-1} & x_{N-2} & x_{N-3} & \cdots & x_{N-p}
\end{array}
\right)
\left(
\begin{array}{c}
\phi_1 \\
\phi_2 \\
\phi_3 \\
\vdots \\
\phi_p
\end{array}
\right)
\tag{6.51}
$$

This method, known more specifically as the forward least squares method (Broersen 2006), omits the first p measurements from the vector of observables.

It is also possible to reverse the measurements and calculate a least squares

estimate of the AR parameters, known as the backward least squares method. The general AR(p) model now becomes

$$
E\left\{\begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_{N-p} \end{array}\right\} = \left(\begin{array}{cccc} x_2 & x_3 & \cdots & x_{p+1} \\ x_3 & x_4 & \cdots & x_{p+2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N-p+1} & x_{N-p+2} & \cdots & x_N \end{array}\right)\left(\begin{array}{c} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_p \end{array}\right) \tag{6.52}
$$

The forward and backward least squares methods are equivalent, since both yield equivalent theoretical autocovariances. Broersen (2006) also shows that it is possible to minimise the sum of the forwards and backwards residuals.

Unfortunately all of the least squares methods mentioned cannot guarantee a stationary model where the roots are close to the unit circle (see §6.2). Therefore, as with the Yule-Walker equations and Levinson-Durbin recursion, this method is not considered suitable for this study.

6.4.4   Burg's method

It has already been seen that the Levinson-Durbin method estimates an AR(1) process and uses the partial correlation coefficient to sequentially estimate increasing AR orders. Burg's method (Burg 1967) is based upon this principle, but uses a different method for estimating the partial correlation coefficient.

The partial correlation coefficient at each stage is estimated using all available information at the AR order at which it is calculated. This comes from the fact that it utilises the forward and backward residuals, as seen in the least squares methods (see §6.4.3).

Broersen (2006) provides an extensive explanation of Burg's method and as such forms the basis of the following derivations. It should be noted that the derivation given in Broersen (2006) is based on the AR process using the Priestley model and therefore is in terms of the reflection coefficient and not the partial correlation coefficient. As this study prefers the AR process usung the Box-Jenkins model the derivations may differ slightly as they use the partial correlation coefficient, but are

nonetheless equivalent.

Let the forward residuals of orders $1, \ldots, p$ be defined as

$$f_0(k) = x_k$$

$$f_1(k) = x_k + \phi_{1,1} x_{k-1}$$

$$\vdots$$

$$f_p(k) = x_k + \phi_{p,1} x_{k-1} + \cdots + \phi_{p,p} x_{k-p}$$

(6.53)

and the backward residuals of orders $1, \ldots, p$ be defined as

$$b_0(k) = x_k$$

$$b_1(k) = x_{k-1} + \phi_{1,1} x_k$$

$$\vdots$$

$$b_p(k) = x_{k-p} + \phi_{p,1} x_{k-p+1} + \cdots + \phi_{p,p} x_k$$

(6.54)

The forwards and backwards residuals can also be written as

$$f_p(k) = \begin{pmatrix} x_k & x_{k-1} & \cdots & x_{k-p} \end{pmatrix} \begin{pmatrix} 1 \\ \phi_{p,1} \\ \vdots \\ \phi_{p,p} \end{pmatrix}$$

(6.55)

$$b_p(k) = \begin{pmatrix} x_k & x_{k-1} & \cdots & x_{k-p} \end{pmatrix} \begin{pmatrix} \phi_{p,p} \\ \vdots \\ \phi_{p,1} \\ 1 \end{pmatrix}$$

(6.56)

What is interesting to note here is that the order of values from $x(k)$ have been reversed for $b_p(k)$, which in turn has reversed the order of the AR parameters. This provides computational advantages in the implementation (Broersen 2006). Recalling the definitions of α_n and α'_n given by Eq. (6.44) used in the Levinson-

Durbin recursion algorithm the forward and backward residuals can be written as

$$f_p(k) = \begin{pmatrix} x_k & x_{k-1} & \cdots & x_{k-p} \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_n \end{pmatrix} \tag{6.57}$$

$$b_p(k) = \begin{pmatrix} x_k & x_{k-1} & \cdots & x_{k-p} \end{pmatrix} \begin{pmatrix} \alpha'_n \\ 1 \end{pmatrix} \tag{6.58}$$

Recalling the step in the Levinson-Durbin recursion that adjusts the AR values from the previous order to the next order, given by Eq. (6.46),

$$\alpha_{n+1} = \begin{pmatrix} \alpha_n + \phi_{n+1,n+1} \ \alpha'_n \\ \phi_{n+1,n+1} \end{pmatrix}$$

the forward and backward residuals are given by Broersen (2006) as

$$f_p(k) = f_{p-1}(k) + \phi_{p,p} \ b_{p-1}(k-1) \tag{6.59}$$

$$b_p(k) = b_{p-1}(k-1) + \phi_{p,p} \ f_{p-1}(k) \tag{6.60}$$

In the case of Burg's method $\phi_{p,p}$ is estimated by minimising the sum of squares of the forward and backward residuals, which results in (Broersen 2006)

$$\phi_{p,p} = 2 \ \frac{\sum\limits_{k=p+1}^{N} f_{p-1}(k) \ b_{p-1}(k-1)}{\sum\limits_{k=p+1}^{N} f_{p-1}^2(k) + b_{p-1}^2(k-1)} \tag{6.61}$$

where N is the sample size of the stochastic process $x(k)$, from which the forward and backward residuals are derived. In §6.4.3 it was mentioned that it is possible to estimate the AR parameters using the forward and backward least squares method. Whereas that method estimates all AR parameters using the forward and backward residuals in one single adjustment, Burg's method estimates the AR parameters using the forward and backward residuals from AR(1) to AR(p) at each successive AR order.

An important property of the partial correlation coefficient, $\phi_{p,p}$, that is esti-

mated using Burg's method is that it is always smaller that one in absolute value, as shown in Broersen (2006), and hence guarantees that the AR process estimated is stationary. This is because all the roots of the generating function are outside the unit circle (assuming an AR process using the Box-Jenkins method, see §6.2).

Unlike for Levin-Durbin recursion, calculation of the partial correlation coefficient and the subsequent updating of the AR parameters does not require an estimate of the residual variance of the AR process. However, this is still important as it can be used for determining the AR model order (see §6.5). This can be achieved using Eq. (6.48) of Levinson-Durbin recursion,

$$s_{n+1}^2 = s_n^2(1 - \phi_{n+1,n+1}^2)$$

A summary of Burg's method is given below:

1. Begin with $p = 1$, i.e. an AR(1) process.

2. Calculate the autocovariance at lag 0 for the stochastic process, s_0^2, using Eq. (6.41)

3. Calculate the first partial correlation coefficient, $\phi_{1,1}$, using Eq. (6.61).

4. Calculate the estimated variance for the AR(1) process, $s_1^2 = s_0^2(1 - \phi_{1,1})$.

5. Calculate the partial correlation coefficient $\phi_{p+1,p+1}$ using Eq. (6.61).

6. Update all AR parameters using Eq. (6.46).

7. Calculate the estimated variance for AR order $p + 1$ using Eq. (6.48)

8. Update the AR order, i.e. $p = p + 1$. If the desired AR has not been reached repeat from Step 5.

Burg's method is the chosen method for estimating the AR parameters for the post-fit residuals. The first reason for this is that the roots of the AR process are guaranteed to lie outside the unit circle (see §6.2), hence resulting in a stationary process. It is also a computationally efficient method, hence suitable for implementation in real-time PPP software.

6.5 Determining the model order

A number of methods for estimating the coefficients of an AR process up to order p were discussed in §6.4. Burg's method was chosen to be the best method for estimating the AR parameters from the residual time series. However, the question of how to determine the correct value of p has not been addressed. In this section a number of methods for determining the order of an AR process will be discussed.

6.5.1 Residual variance

It is shown in Broersen (2006) that for an AR(p) process

$$\phi_i = 0 \text{ where } i > p \tag{6.62}$$

Hence the AR parameters have no influence beyond the true order of the model. One method of determining model order uses this property by analysing the estimated residual variance, for which Priestley (1981) gives the approximately unbiased estimate as

$$\hat{\sigma}_\varepsilon^2 = \frac{N - p}{N - 2p - 1} \left[R(0) - \phi_1 R(1) - \ldots - \phi_p R(p) \right] \tag{6.63}$$

If the model order is underdetermined then $\hat{\sigma}_\varepsilon^2$ will be larger than the true residual variance, σ_ε^2, as the neglected parameters will account for the larger estimated variance. Where the model order is overdetermined, given the condition in Eq. (6.62), any increase in the model order will not significantly reduce $\hat{\sigma}_\varepsilon^2$. The model order is found by plotting $\hat{\sigma}_\varepsilon^2$ for increasing values of p and then observing the point at which $\hat{\sigma}_\varepsilon^2$ does not significantly change.

6.5.2 Partial autocorrelation

It has already been demonstrated in §6.4.2 that the partial autocorrelation coefficient can be used to describe the new contribution of an AR order to the autocorrelation. From the condition in Eq. (6.62) it follows that if a model is truly an AR(p) process then $\phi_{i,i} = 0$ where $i > p$. As such, plotting $\phi_{i,i}$ for increasing values of i should

give an indication of the true model order. For a large enough sample size N an approximate measure for testing $\phi_{i,i} = 0$ is to determine whether $\phi_{i,i}$ lies in the range $-2\sqrt{1/N} \leq \phi_{i,i} \leq 2\sqrt{1/N}$ (Priestley 1981).

6.5.3 Final Prediction Error

Developing the residual variance concept presented in §6.5.1, Akaike (1970) takes the approach of fitting models of increasing order p and for each computes the final prediction error (FPE) criterion, given by

$$\text{FPE}(p) = \hat{\sigma}_\varepsilon^2 \frac{N+p}{N-p} \tag{6.64}$$

where N is the sample size and $\hat{\sigma}_\varepsilon^2$ is the estimated residual variance. The order that corresponds to the lowest FPE is chosen as the true AR order. An in depth derivation for the FPE can be found in Priestley (1981) and Broersen (2006).

6.5.4 Akaike's Information Criterion

Akaike (1974) introduced the order selection criterion known as Akaike's Information Criterion (AIC), defined as

$$\text{AIC}(p) = \ln \hat{\sigma}_\varepsilon^2 + 2\frac{p}{N} \tag{6.65}$$

where N is the sample size and $\hat{\sigma}_\varepsilon^2$ is the estimated residual variance. Although developed in relation to the maximum likelihood of residual variance, it is commonly applied to estimates of σ_ε^2 obtained through AR parameter estimation (Broersen 2006). As with the FPE, the order that yields the minimum AIC value is taken as the most likely candidate.

It is interesting to note that when using a large sample size and taking $\ln(1+\delta) \approx \delta$,

$$\text{AIC}(p) \approx \ln[\text{FPE}(p)] \tag{6.66}$$

The result of this is that FPE(p) and AIC(p) invariably yield the same estimated

order (Priestley 1981, Broersen 2006).

Given its wide use, AIC is the chosen method for calculating model order in §6.8.

6.6 Modelling a non-optimal filter as an AR process

In this section the derivations given in §5.2 will be further developed to show that in certain cases a non-optimal filter can be modelled as an AR process.

6.6.1 Predicted residuals

It was shown in §5.2 that for an optimal filter $E\{\underline{v}_k, \underline{v}_l^T\} = 0$ where $k \neq l$ and that for a non-optimal filter $E\{\underline{v}_k, \underline{v}_l^T\} \neq 0$ where $k \neq l$. It will be shown that there exists a condition where the predicted residuals can be modelled as an AR process. The notation used in §5.2 will be continued in this section.

Recalling Eq. (5.23) the expectation of the vector of predicted residuals at epoch k with those at epoch l, where $l < k$, is given by

$$E\{\underline{v}_k^*, \underline{v}_l^{*T}\} = -A_k \left\{ \prod_{i=l+1}^{k-1} \Phi_{i+1}(I - K_i^* A_i) \right\} \Phi_{l+1}\left[K_l^* Q_{y_l} - (I - K_l^* A_l) \, Q_{x_{l|l-1}}^* A_l^T\right]$$

(6.67)

From Eq. (3.29) the non-optimal estimate for the variance-covariance matrix of the vector of predicted residuals at epoch l is given by

$$E\{\underline{v}_l^*, \underline{v}_l^{*T}\} = Q_{y_l} + A_l Q_{x_{l|l-1}}^* A_l^T$$

(6.68)

Generalising Eq. (6.67) to expectation of the vector of predicted residuals at epoch k with those at epoch $k - p$, where $p \geq 1$, gives

$$E\{\underline{v}_k^*, \underline{v}_{k-p}^{*T}\} = -A_k \left\{ \prod_{i=k-p+1}^{k-1} \Phi_{i+1}(I - K_i^* A_i) \right\}$$

$$\cdot \Phi_{k-p+1}\left[K_{k-p}^* Q_{y_{k-p}} - (I - K_{k-p}^* A_{k-p}) \, Q_{x_{k-p|k-p-1}}^* A_{k-p}^T\right]$$

(6.69)

For simplicity, denote

$$[\,\cdot\,] = \Phi_{k-p+1}\left[K^*_{k-p}Q_{y_{k-p}} - (I - K^*_{k-p}A_{k-p})\,Q^*_{x_{k-p|k-p-1}}A^T_{k-p}\right] \qquad (6.70)$$

Expanding the product in Eq. (6.69) sequentially gives

$$E\{\underline{v}^*_k, \underline{v}^{*T}_{k-p}\} = -A_k\Phi_k(I - K^*_{k-1}A_{k-1})\left\{\prod_{i=k-p+1}^{k-2}\Phi_{i+1}(I - K^*_iA_i)\right\}[\,\cdot\,]$$

$$= -A_k\Phi_k\left\{\prod_{i=k-p+1}^{k-2}\Phi_{i+1}(I - K^*_iA_i)\right\}[\,\cdot\,] \qquad (6.71)$$

$$+ A_k\Phi_kK^*_{k-1}A_{k-1}\left\{\prod_{i=k-p+1}^{k-2}\Phi_{i+1}(I - K^*_iA_i)\right\}[\,\cdot\,]$$

$$= -A_k\Phi_k\left\{\prod_{i=k-p+1}^{k-2}\Phi_{i+1}(I - K^*_iA_i)\right\}[\,\cdot\,] \qquad (6.72)$$

$$- A_k\Phi_kK^*_{k-1}E\{\underline{v}^*_{k-1}, \underline{v}^{*T}_{k-p}\}$$

$$= -A_k\Phi_k\Phi_{k-1}\left\{\prod_{i=k-p+1}^{k-3}\Phi_{i+1}(I - K^*_iA_i)\right\}[\,\cdot\,]$$

$$- A_k\Phi_k\Phi_{k-1}K^*_{k-2}E\{\underline{v}^*_{k-2}, \underline{v}^{*T}_{k-p}\} \qquad (6.73)$$

$$- A_k\Phi_kK^*_{k-1}E\{\underline{v}^*_{k-1}, \underline{v}^{*T}_{k-p}\}$$

$$= -A_k\Phi_k\Phi_{k-1}\cdots\Phi_{k-p+2}[\,\cdot\,]$$

$$- A_k\Phi_kK^*_{k-1}E\{\underline{v}^*_{k-1}, \underline{v}^{*T}_{k-p}\}$$

$$- A_k\Phi_k\Phi_{k-1}K^*_{k-2}E\{\underline{v}^*_{k-2}, \underline{v}^{*T}_{k-p}\} \qquad (6.74)$$

$$\cdots$$

$$- A_k\Phi_k\Phi_{k-1}\cdots\Phi_{k-p+1}K^*_{k-p+1}E\{\underline{v}^*_{k-p+1}, \underline{v}^{*T}_{k-p}\}$$

The term in Eq. (6.69) given by Eq. (6.70) is now expanded and Eq. (6.68) substituted giving

$$[\,\cdot\,] = \Phi_{k-p+1}\left[K^*_{k-p}Q_{y_{k-p}} - (I - K^*_{k-p}A_{k-p})\,Q^*_{x_{k-p|k-p-1}}A^T_{k-p}\right]$$

$$= \Phi_{k-p+1}\left[K^*_{k-p}Q_{y_{k-p}} - Q^*_{x_{k-p|k-p-1}}A^T_{k-p} + K^*_{k-p}A_{k-p}Q^*_{x_{k-p|k-p-1}}A^T_{k-p}\right]$$

$$= \Phi_{k-p+1}\left[K^*_{k-p}Q_{y_{k-p}} - Q^*_{x_{k-p|k-p-1}}A^T_{k-p} + K^*_{k-p}(E\{v^*_{k-p}, v^{*T}_{k-p}\} - Q_{y_{k-p}})\right]$$

149

$$= \Phi_{k-p+1} K^*_{k-p} E\{v^*_{k-p}, v^{*T}_{k-p}\} - \Phi_{k-p+1} Q^*_{x_{k-p|k-p-1}} A^T_{k-p} \tag{6.75}$$

Substituting Eq. (6.75) into Eq. (6.74) gives

$$
\begin{aligned}
E\{\underline{v}^*_k, \underline{v}^{*T}_{k-p}\} = &-A_k \Phi_k K^*_{k-1} E\{\underline{v}^*_{k-1}, \underline{v}^{*T}_{k-p}\} \\
&- A_k \Phi_k \Phi_{k-1} K^*_{k-2} E\{\underline{v}^*_{k-2}, \underline{v}^{*T}_{k-p}\} \\
&\cdots \\
&- A_k \Phi_k \Phi_{k-1} \cdots \Phi_{k-p+2} K^*_{k-p+1} E\{\underline{v}^*_{k-p+1}, \underline{v}^{*T}_{k-p}\} \\
&- A_k \Phi_k \Phi_{k-1} \cdots \Phi_{k-p+1} K^*_{k-p} E\{\underline{v}^*_{k-p}, \underline{v}^{*T}_{k-p}\} \\
&+ A_k \Phi_k \Phi_{k-1} \cdots \Phi_{k-p+1} Q_{\hat{x}_{k-p|k-p-1}} A^T_{k-p}
\end{aligned}
\tag{6.76}
$$

This can be simplified to

$$
\begin{aligned}
E\{\underline{v}^*_k, \underline{v}^{*T}_{k-p}\} = &-\left\{ \sum_{i=1}^{p} A_k \left\{ \prod_{j=k}^{k-p+1} \Phi_j \right\} K^*_{k-i} E\{\underline{v}^*_{k-i}, \underline{v}^{*T}_{k-p}\} \right\} \\
&+ A_k \left\{ \prod_{j=k}^{k-p+1} \Phi_j \right\} Q_{\hat{x}_{k-p|k-p-1}} A^T_{k-p}
\end{aligned}
\tag{6.77}
$$

It has already been shown in §3.8 that $\Phi_k = I_k$ for PPP, therefore Eq. (6.77) can be reduced to

$$E\{\underline{v}^*_k, \underline{v}^{*T}_{k-p}\} = -\left\{ \sum_{i=1}^{p} A_k K^*_{k-i} E\{\underline{v}^*_{k-i}, \underline{v}^{*T}_{k-p}\} \right\} + A_k Q_{\hat{x}_{k-p|k-p-1}} A^T_{k-p} \tag{6.78}$$

The final term of Eq. (6.78) can be written as

$$A_k Q_{\hat{x}_{k-p|k-p-1}} A^T_{k-p} = E\{A_k \hat{\underline{x}}_{k-p|k-p-1}, [A_{k-p} \hat{\underline{x}}_{k-p|k-p-1}]^T\} \tag{6.79}$$

This is the covariance between the linearised observables at epoch $k - p$ with the linearised observables at epoch k using the predicted solution at epoch $k - p$. For both static and kinematic solutions it reasonable to assume that this term will be zero for some p and hence Eq. (6.78) is a linear relationship between the expectations $E\{\underline{v}^*_{k-i}, \underline{v}^{*T}_{k-p}\}$ where $i = 1, \ldots, p$. This is also seen in an AR process, as shown by the Yule-Walker equations (see §6.4.1).

Recalling Eq. (6.21), an AR(p) process for a stochastic process $x(k)$ using the Box-Jenkins model is given by

$$x_k = \phi_1 x_{k-1} + \phi_2 x_{k-2} + \cdots + \phi_p x_{k-ps} + \underline{\varepsilon}_k$$

By considering that the vectors of predicted residuals are an AR(1) process where the temporal correlation is only between observations from the same GNSS satellite then this becomes

$$\underline{v}_k - \phi_1 \underline{v}_{k-1} = \underline{\epsilon}_k \tag{6.80}$$

where $\underline{\epsilon}_k$ is Gaussian noise and ϕ_1 a diagonal matrix. It is later shown that the post-fit residuals closely match an AR(p) process, with $p = 1$ often adequate. In anticipation of this result a theoretical consideration is sought that will corroborate this assumption. In the first instance consider the predicted residuals.

Eq. (6.80) can be rewritten as using Eq. (3.21), (3.26) and (3.28) as

$$
\begin{aligned}
\underline{v}_k - \phi_1 \underline{v}_{k-1} &= (\underline{y}_k - A_k \hat{\underline{x}}_{k|k-1}) - \phi_1 \underline{v}_{k-1} \\
&= \underline{y}_k - A_k \Phi_k \hat{\underline{x}}_{k-1|k-1} - \phi_1 \underline{v}_{k-1} \\
&= \underline{y}_k - A_k \Phi_k [\hat{\underline{x}}_{k-1|k-2} + K_{k-1} \underline{v}_{k-1}] - \phi_1 \underline{v}_{k-1} \\
&= (\underline{y}_k - A_k \Phi_k \hat{\underline{x}}_{k-1|k-2}) - (A_k \Phi_k K_{k-1} + \phi_1) \underline{v}_{k-1}
\end{aligned}
\tag{6.81}
$$

Now consider

$$\underline{y}_k = A_k \tilde{\underline{x}}_k + \tilde{\underline{\epsilon}}_k \tag{6.82}$$

where $\tilde{\underline{x}}_k$ is the 'true' solution and $\tilde{\underline{\epsilon}}_k$ is Gaussian noise. Inserting this into Eq. (6.81) gives

$$\underline{v}_k - \phi_1 \underline{v}_{k-1} = (A_k \tilde{\underline{x}}_k + \tilde{\underline{\epsilon}}_k - A_k \Phi_k \hat{\underline{x}}_{k-1|k-2}) - (A_k \Phi_k K_{k-1} + \phi_1) \underline{v}_{k-1} \tag{6.83}$$

From Eq. (3.28) it follows that the predicted residual at $k - 1$ is

$$\underline{v}_{k-1} = \underline{y}_{k-1} - A_{k-1} \hat{\underline{x}}_{k-1|k-2} \tag{6.84}$$

This is used to remove $\underline{v}_{k-1}$ on the right hand side of Eq. (6.83) to give

$$
\begin{aligned}
\underline{v}_k - \phi_1 \underline{v}_{k-1} = {} & (A_k \underline{\tilde{x}}_k + \underline{\tilde{\epsilon}}_k - A_k \Phi_k \hat{\underline{x}}_{k-1|k-2}) \\
& - (A_k \Phi_k K_{k-1} + \phi_1)(\underline{y}_{k-1} - A_{k-1} \hat{\underline{x}}_{k-1|k-2})
\end{aligned}
\tag{6.85}
$$

By substituting

$$
\underline{y}_{k-1} = A_{k-1} \underline{\tilde{x}}_{k-1} + \underline{\tilde{\epsilon}}_{k-1}
\tag{6.86}
$$

into Eq. (6.85) gives

$$
\begin{aligned}
\underline{v}_k - \phi_1 \underline{v}_{k-1} = {} & (A_k \underline{\tilde{x}}_k + \underline{\tilde{\epsilon}}_k - A_k \Phi_k \hat{\underline{x}}_{k-1|k-2}) \\
& - (A_k \Phi_k K_{k-1} + \phi_1)(A_{k-1} \underline{\tilde{x}}_{k-1} + \underline{\tilde{\epsilon}}_{k-1} - A_{k-1} \hat{\underline{x}}_{k-1|k-2})
\end{aligned}
\tag{6.87}
$$

By assuming that the dynamic model is correct, namely

$$
\underline{\tilde{x}}_k = \Phi_k \underline{\tilde{x}}_{k-1}
\tag{6.88}
$$

it is possible to remove $\underline{\tilde{x}}_k$ such that Eq. (6.87) becomes

$$
\begin{aligned}
\underline{v}_k - \phi_1 \underline{v}_{k-1} = {} & (A_k \Phi_k \underline{\tilde{x}}_{k-1} + \underline{\tilde{\epsilon}}_k - A_k \Phi_k \hat{\underline{x}}_{k-1|k-2}) \\
& - (A_k \Phi_k K_{k-1} + \phi_1)(A_{k-1} \underline{\tilde{x}}_{k-1} + \underline{\tilde{\epsilon}}_{k-1} - A_{k-1} \hat{\underline{x}}_{k-1|k-2})
\end{aligned}
\tag{6.89}
$$

Through rearrangement and simplification Eq. (6.89) becomes

$$
\begin{aligned}
\underline{v}_k - \phi_1 \underline{v}_{k-1} = {} & (A_k \Phi_k (I - K_{k-1} A_{k-1}) - \phi_1 A_{k-1})(\underline{\tilde{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-2}) \\
& - (A_k \Phi_k K_{k-1} + \phi_1)\underline{\tilde{\epsilon}}_{k-1} + \underline{\tilde{\epsilon}}_k
\end{aligned}
\tag{6.90}
$$

In a non-optimal filter it cannot be assumed that $(\underline{\tilde{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-2})$ is Gaussian noise. In order to simplify the notation this is defined as

$$
\Delta \underline{x}_{k-1} = \underline{\tilde{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-2}
\tag{6.91}
$$

Furthermore, the Gaussian noise from epochs k and $k-1$ is defined as

$$\tilde{\underline{\epsilon}}_{k,k-1} = \tilde{\underline{\epsilon}}_k - (A_k \Phi_k K_{k-1} + \phi_1)\tilde{\underline{\epsilon}}_{k-1} \qquad (6.92)$$

since the sum of two normal distributions is also a normal distribution. It is now possible to rewrite Eq. (6.90) as

$$\underline{v}_k - \phi_1 \underline{v}_{k-1} = (A_k \Phi_k(I - K_{k-1}A_{k-1}) - \phi_1 A_{k-1})\Delta\underline{x}_{k-1} + \tilde{\underline{\epsilon}}_{k,k-1} \qquad (6.93)$$

In order to show that $\underline{v}_k - \phi_1 \underline{v}_{k-1}$ is random noise all terms that precede $\tilde{\underline{\epsilon}}_{k,k-1}$ must be shown to be white noise. In the following examples the operator $\overset{?}{=}$ is used, where $x \overset{?}{=} y$ asks the question 'Is x equal to y?'

For the right hand side of this equation to be random noise the following test case is required:

$$(A_k \Phi_k(I - K_{k-1}A_{k-1}) - \phi_1 A_{k-1})\Delta\underline{x}_{k-1} \overset{?}{=} \hat{\underline{\epsilon}}_{k,k-1} \qquad (6.94)$$

where $\hat{\underline{\epsilon}}_{k,k-1}$ is Gaussian noise. Consider the left hand side of Eq. (6.94). In the case of GNSS it can be assumed $\Phi_k \equiv I$. For high rate data it can also be assumed that $A_k \approx A_{k-1}$, hence A_k can be replaced by A_{k-1}. This simplifies Eq. (6.94) to

$$A_{k-1}(I - K_{k-1}A_{k-1})\Delta\underline{x}_{k-1} - \phi_1 A_{k-1}\Delta\underline{x}_{k-1} \overset{?}{=} \hat{\underline{\epsilon}}_{k,k-1} \qquad (6.95)$$

A further simplification can be made by defining

$$\Delta\bar{\underline{x}}_{k-1} = A_{k-1}\Delta\underline{x}_{k-1} \qquad (6.96)$$

which results in Eq. (6.95) becoming

$$[(I - A_{k-1}K_{k-1}) - \phi_1]\Delta\bar{\underline{x}}_{k-1} \overset{?}{=} \hat{\underline{\epsilon}}_{k,k-1} \qquad (6.97)$$

The left hand side of Eq. (6.97) is equal to the right hand side if $\Delta\bar{\underline{x}}_{k-1}$ and ϕ_1 are the eigenvectors and eigenvalues of $(I - A_{k-1}K_{k-1})$.

It is important to note that this is not intended to be a rigorous proof, rather to illustrate that there exists conditions under which the predicted residuals can be time correlated as in an AR(1) process.

6.6.2 Post-fit residuals

Now consider the vectors of post-fit residuals are an AR(1) process and we investigate under what conditions

$$\hat{\underline{e}}_k - \phi_1 \hat{\underline{e}}_{k-1} = \underline{\varepsilon}_k \tag{6.98}$$

where $\underline{\varepsilon}_k$ is the true random noise and ϕ_1 a diagonal matrix. Following a similar procedure as to that done in the previous section this equation expands as

$$
\begin{aligned}
\hat{\underline{e}}_k - \phi_1 \hat{\underline{e}}_{k-1} &= \underline{y}_k - A_k \hat{\underline{x}}_{k|k} - \phi_1 \hat{\underline{e}}_{k-1} \\
&= \underline{y}_k - A_k [\hat{\underline{x}}_{k|k-1} + K_k (y_k - A_k \hat{\underline{x}}_{k|k-1})] - \phi_1 \hat{\underline{e}}_{k-1} \\
&= (I - A_k K_k)(\underline{y}_k - A_k \Phi_k \hat{x}_{k-1|k-1}) - \phi_1 \hat{\underline{e}}_{k-1}
\end{aligned}
\tag{6.99}
$$

Substitute

$$\underline{y}_k = A_k \tilde{\underline{x}}_k + \tilde{\underline{\varepsilon}}_k \tag{6.100}$$

into Eq. (6.99) to give

$$\hat{\underline{e}}_k - \phi_1 \hat{\underline{e}}_{k-1} = (I - A_k K_k)(A_k \tilde{\underline{x}}_k + \tilde{\underline{\varepsilon}}_k - A_k \Phi_k \hat{\underline{x}}_{k-1|k-1}) - \phi_1 \hat{\underline{e}}_{k-1} \tag{6.101}$$

and substitute

$$\tilde{\underline{x}}_k = \Phi_k \tilde{\underline{x}}_{k-1} \tag{6.102}$$

to give

$$\hat{\underline{e}}_k - \phi_1 \hat{\underline{e}}_{k-1} = (I - A_k K_k)(A_k \Phi_k \tilde{\underline{x}}_{k-1} + \tilde{\underline{\varepsilon}}_k - A_k \Phi_k \hat{\underline{x}}_{k-1|k-1}) - \phi_1 \hat{\underline{e}}_{k-1} \tag{6.103}$$

Reorder

$$\hat{\underline{e}}_k - \phi_1 \hat{\underline{e}}_{k-1} = (I - A_k K_k)(A_k \Phi_k \tilde{\underline{x}}_{k-1} - A_k \Phi_k \hat{\underline{x}}_{k-1|k-1} + \tilde{\underline{\varepsilon}}_k) - \phi_1 \hat{\underline{e}}_{k-1} \tag{6.104}$$

From Eq. (3.49) it follows that

$$\hat{\underline{e}}_{k-1} = \underline{y}_{k-1} - A_{k-1}\hat{\underline{x}}_{k-1|k-1} \tag{6.105}$$

and substituting Eq. (6.100) gives

$$\hat{\underline{e}}_{k-1} = A_{k-1}\tilde{\underline{x}}_{k-1} + \tilde{\underline{\varepsilon}}_{k-1} - A_{k-1}\hat{\underline{x}}_{k-1|k-1} \tag{6.106}$$

Substituting this into Eq. (6.104) gives

$$
\begin{aligned}
\hat{\underline{e}}_k - \phi_1\hat{\underline{e}}_{k-1} &= (I - A_k K_k)(A_k\Phi_k\tilde{\underline{x}}_{k-1} - A_k\Phi_k\hat{\underline{x}}_{k-1|k-1} + \tilde{\underline{\varepsilon}}_k) \\
&\quad - \phi_1(A_{k-1}\tilde{\underline{x}}_{k-1} + \tilde{\underline{\varepsilon}}_{k-1} - A_{k-1}\hat{\underline{x}}_{k-1|k-1})
\end{aligned}
\tag{6.107}
$$

This can be simplified somewhat with the grouped terms $[\tilde{\underline{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-1}]$ to give

$$
\begin{aligned}
\hat{\underline{e}}_k - \phi_1\hat{\underline{e}}_{k-1} &= (I - A_k K_k)(A_k\Phi_k[\tilde{\underline{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-1}] + \tilde{\underline{\varepsilon}}_k) \\
&\quad - \phi_1(A_k[\tilde{\underline{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-1}] + \tilde{\underline{\varepsilon}}_{k-1})
\end{aligned}
\tag{6.108}
$$

or

$$
\begin{aligned}
\hat{\underline{e}}_k - \phi_1\hat{\underline{e}}_{k-1} &= (I - A_k K_k)A_k\Phi_k[\tilde{\underline{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-1}] + (I - A_k K_k)\tilde{\underline{\varepsilon}}_k \\
&\quad - \phi_1 A_k[\tilde{\underline{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-1}] - \phi_1\tilde{\underline{\varepsilon}}_{k-1}
\end{aligned}
\tag{6.109}
$$

Factorising $[\tilde{\underline{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-1}]$ once again gives

$$
\begin{aligned}
\hat{\underline{e}}_k - \phi_1\hat{\underline{e}}_{k-1} &= [(I - A_k K_k)A_k\Phi_k - \phi_1 A_{k-1}][\tilde{\underline{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-1}] \\
&\quad + (I - A_k K_k)\tilde{\underline{\varepsilon}}_k - \phi_1\tilde{\underline{\varepsilon}}_{k-1}
\end{aligned}
\tag{6.110}
$$

Again assuming $\Phi_k \equiv I$ and $A_k \approx A_{k-1}$ with

$$\Delta\tilde{\underline{x}}_k = A_k[\tilde{\underline{x}}_{k-1} - \hat{\underline{x}}_{k-1|k-1}] \tag{6.111}$$

$$\tilde{\underline{\varepsilon}}_{k,k-1} = (I - A_k K_k)\tilde{\underline{\varepsilon}}_k - \phi_1\tilde{\underline{\varepsilon}}_{k-1} \tag{6.112}$$

the following test case results:

$$[(I - A_k K_k) - \phi_1] \Delta \underline{\tilde{x}}_k = \underline{\tilde{\varepsilon}}_{k,k-1} \tag{6.113}$$

As for the predicted residuals the left hand side of Eq. (6.113) is equal to the right hand side if $\Delta \underline{\tilde{x}}_k$ and ϕ_1 are the eigenvectors and eigenvalues of $(I - A_k K_k)$.

Thus as for the predicted residuals, it is feasible that the post-fit residuals can be modelled as an AR(1) process. Once again this is not intended to be a rigorous proof, but rather to demonstrate that there exists conditions under which the post-fit residuals can be time correlated as in an AR(1) process.

6.7 Post-fit residuals as an AR(1) process

Analysis of the data in §5.3 and the subsequent theoretical derivations in §6.6 have indicated that a time series of residuals for each satellite and observation type pair can probably be characterised as an AR(1) process. Using the Box-Jenkins model for an autoregressive process, given by Eq. (6.21), the post-fit residual at epoch k expressed as an AR(1) process becomes (c.f. Eq. 6.98)

$$\underline{\hat{e}}_k - \phi_1 \underline{\hat{e}}_{k-1} = \underline{\varepsilon}_k \tag{6.114}$$

where $\underline{\varepsilon}_k$ is in fact the 'true' residual at epoch k, i.e. the residual with the effect of time correlation removed. This will be called the whitened residual. Since Burg's method guarantees that $\phi_1 > 0$ then it can be seen that the whitened residual is found by subtracting the weighted influence of the previous post-fit residual from the post-fit residual at epoch k.

6.7.1 Estimation of the AR(1) coefficient

As part of an initial trial the post-fit residuals from the ALBM dataset were used to estimate the AR coefficients using the forward least squares method (see §6.4.3). The AR(1) coefficient, ϕ_1, was firstly estimated as a constant value over the whole

time series and subsequently as a moving window of 240 seconds. This was done in order to see if the estimated values were stable enough to be considered constant over a satellite pass. The sample size of 240 seconds was chosen to be consistent with the data analysis performed in Chapter 5.
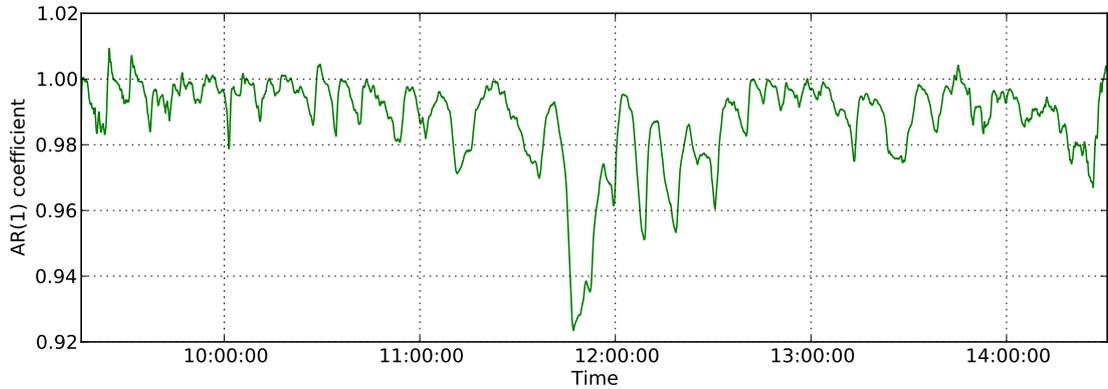
Figure 6.2: Variation of the AR(1) coefficient estimated using the forwards least squares method over a satellite pass (G9 L1 Code).

The results of this initial implementation are shown in Figure 6.2. They show that the estimated coefficient is very close to unity but at some epochs greater than 1. These two facts point to an unstable estimator. It has already been discussed in §6.4 that the least squares methods can be unstable where the AR process is close to the unit circle. For this reason least squares estimation of the AR coefficient was dropped in favour of Burg's method, since this would guarantee a stationary model.
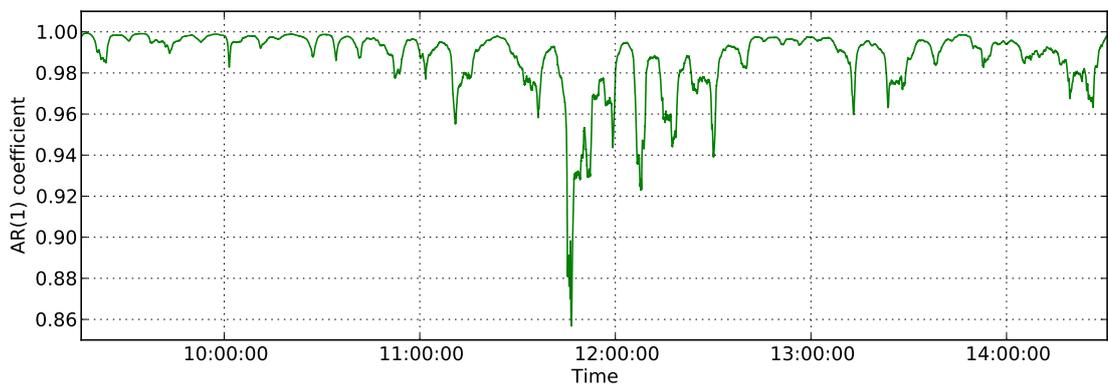
Figure 6.3: Variation of the AR(1) coefficient estimated using Burg's method over a satellite pass (G9 L1 Code).

Following the implementation of Burg's method as the chosen estimator of the parameter, the results yielded coefficients that were all outside the unit circle (see

§6.2), as anticipated. Figure 6.3 shows the variation in the AR(1) coefficient using Burg's method. By making a comparison between Figures 6.2 and 6.3 it can be seen that whilst the general variation in the coefficient is the same, Burg's method satisfies the requirement for stationarity.

With a successful method for the estimating the AR(1) coefficients, the algorithm was used to estimate the AR(1) coefficients for the three static sites described in §5.3.2; ABPO, FALK and GUAM.

6.7.2   Whitening the residual

By taking a 240 second window for a specified satellite as an example, it is possible to use the estimated coefficient to whiten the time series using

$$\underline{\varepsilon}_k = \underline{\hat{e}}_k - \phi_1 \underline{\hat{e}}_{k-1} \qquad (6.115)$$

An example time series is shown in Figure 6.4, with a comparison between (a) the original time series and (b) the whitened time series where an AR(1) process has been assumed and the AR coefficient was estimated ($\phi_1 = 0.995$) using Burg's method.

The first noticeable feature is that the whitened time series appears to resemble what would be expected for white noise. The amplitude of the signal is also much smaller. This is not surprising as it was shown in §5.3.5 that the post-fit residuals are displaying positive correlation and hence subsequent residuals are likely to be of a similar value. Given the high values for ϕ_1, which as seen in Figures 6.2 and 6.3 are in general close to 1, this will result in a residual that is much smaller in magnitude.

Whereas the whitened residuals in Figure 6.4b are calculated by whitening each value using a common value for all 240 second windows, the method is modified so that each residual is whitened based on an AR(1) estimate of the previous 240 epochs (see §5.3.3). This moving window method allows it to be used both automatically and in real time. This also reflects the fact that the time correlation in the residuals will vary over a satellite pass.
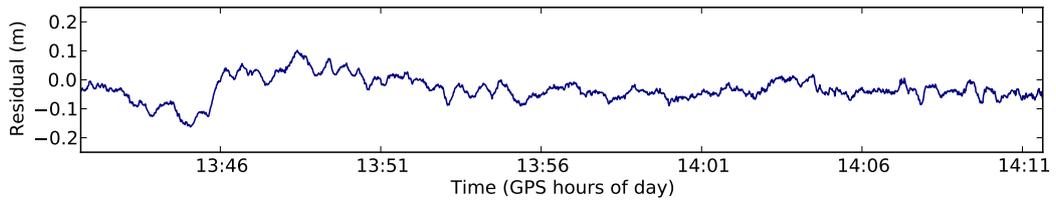
(a) Original residuals
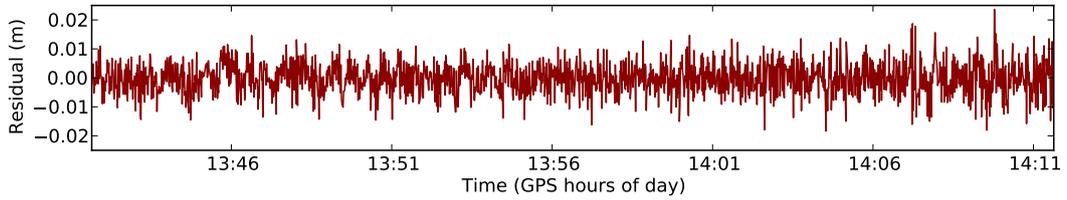
(b) AR(1) whitened residuals (fixed, $\phi_1 = 0.995$)
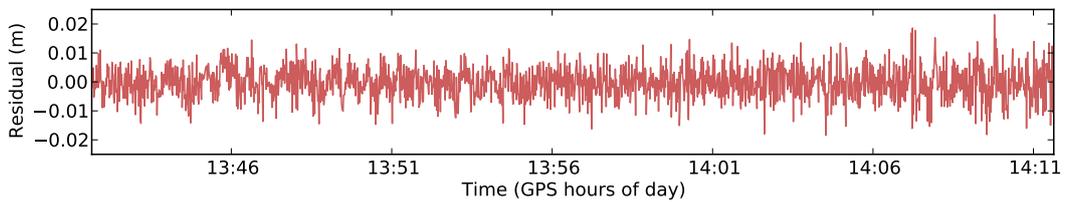
(c) AR(1) whitened residuals (variable coefficient determined by a 240 second moving window)
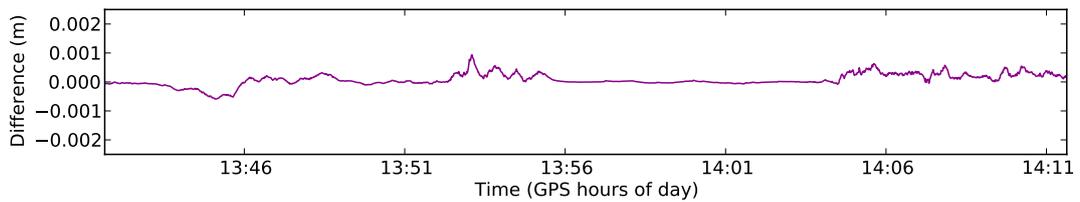
(d) Difference in AR(1) residual determined using a fixed coefficient and variable coefficient determined by a 240 second moving window

Figure 6.4: Time series of residuals for G9 L1 Code at ALBM over a 30 minute period.

The method requires 240 seconds of data to calculate a reliable estimate of the AR parameters, as determined in §5.3.3. If this requirement is to be held then the first 240 epochs of a satellite pass cannot be whitened. It is also a requirement that the data is continuous. If it is assumed that the AR parameter may change unpredictably from one epoch to the next then any gap in the data breaks this discontinuity. If this condition is strictly held a further 240 seconds will have to elapse before the AR coefficient can be estimated once again. By relaxing this assumption it can be assumed that over small periods of time the AR parameter will not significantly change. This is discussed further in §7.5.

The time series in Figure 6.4a was whitened with an AR(1) process, where ϕ_1 is

determined using a moving window of 240 seconds, and the resulting time series is shown in Figure 6.4c. It can be seen that residuals determined using these methods are almost identical, but as is clear in Figure 6.4d the two time series do differ slightly. It is important to note that even though the absolute differences may be small, the amplitude of the signal is much smaller and so smaller differences can have a much greater effect on the whitened test statistic that will be presented later in the study. It will be shown in Chapter 7 that the variance of the whitened residual is also much smaller than the variance of the unwhitened residual, therefore these small differences have the possibility to be magnified when forming the whitened test statistic.

6.7.3   Testing for whiteness

In order to determine whether the whitened residuals can actually said to be characteristic of random noise the methods for measuring correlation used in §5.3 are repeated in this section using the whitened residuals. Wherever possible the same data will be used so that direct comparisons can be made. In all cases there will be fewer whitened residuals than unwhitened residuals, as an initial 240 epochs are required for estimating the AR(1) coefficients.

Lag plots

The method for calculating the autocorrelation function is given in §5.3.4. Figures 6.5 - 6.8 show a lag plot for the four observation types for a chosen satellite from the ABPO, FALK, GUAM and ALBM datasets respectively.

In all cases the residuals now closely resemble the circular pattern seen in Figure 5.10a. Inspection of the axes in these plots shows that the amplitude of the residuals has decreased, which is to be expected when the AR(1) coefficient is close to unity. It is interesting to note the sparsely scattered values in the lower left and upper right of Figures 6.7c and 6.7d which suggest that some of the correlation has not been removed.

It can therefore be concluded from visual interpretation of the lag plots that the

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
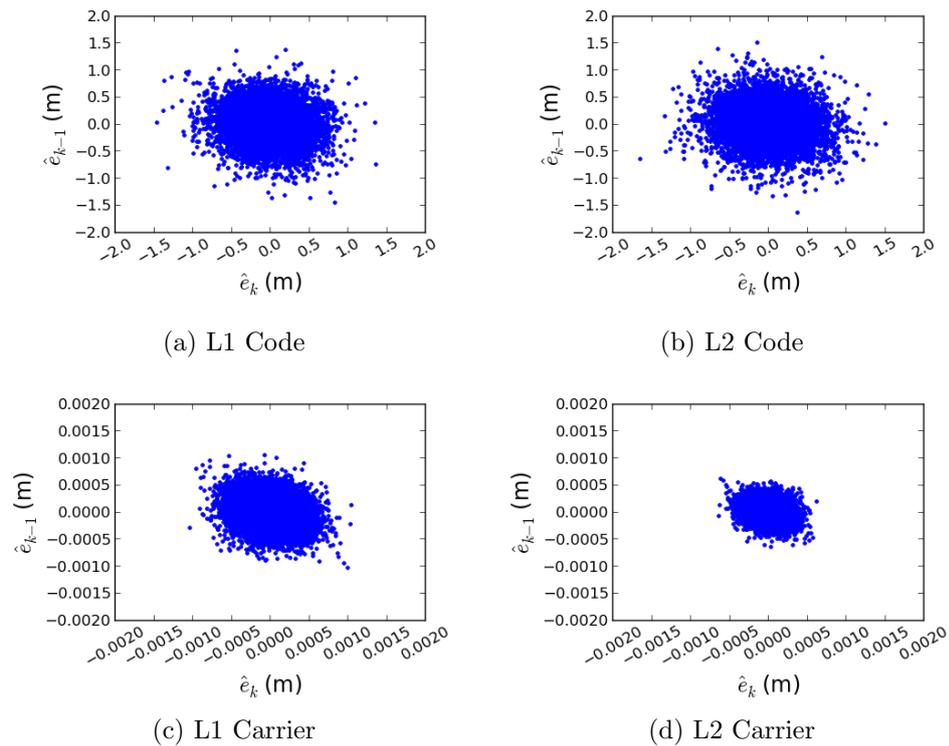
Figure 6.5: Lag plot for the whitened residuals of GPS SVN05 at ABPO.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
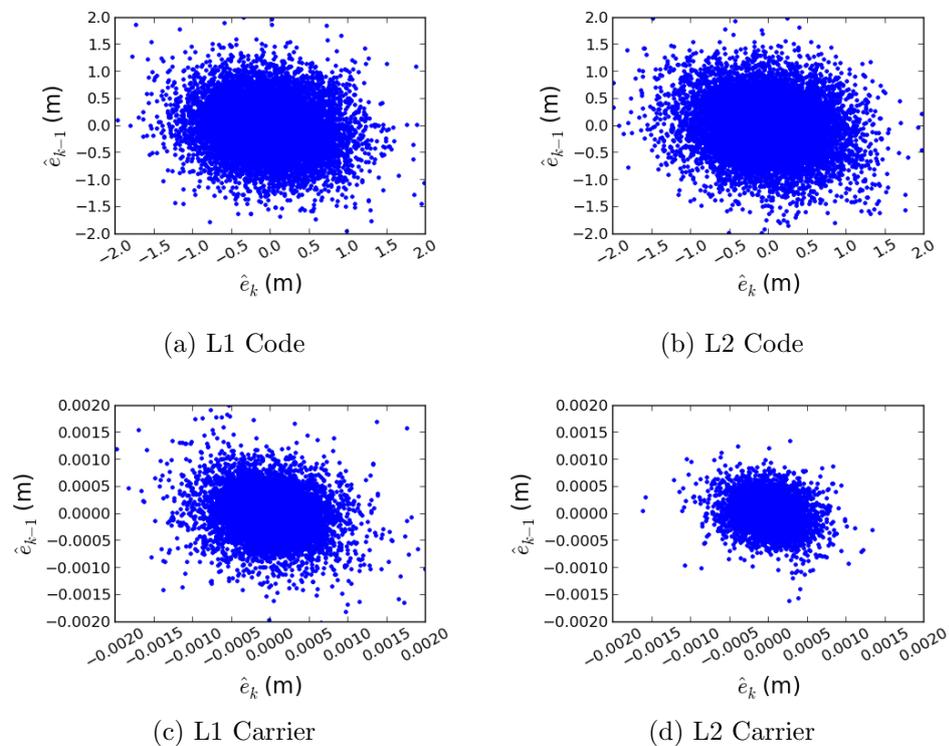
Figure 6.6: Lag plot for the whitened residuals of GPS SVN05 at FALK.

(a) L1 Code   (b) L2 Code

(c) L1 Carrier   (d) L2 Carrier
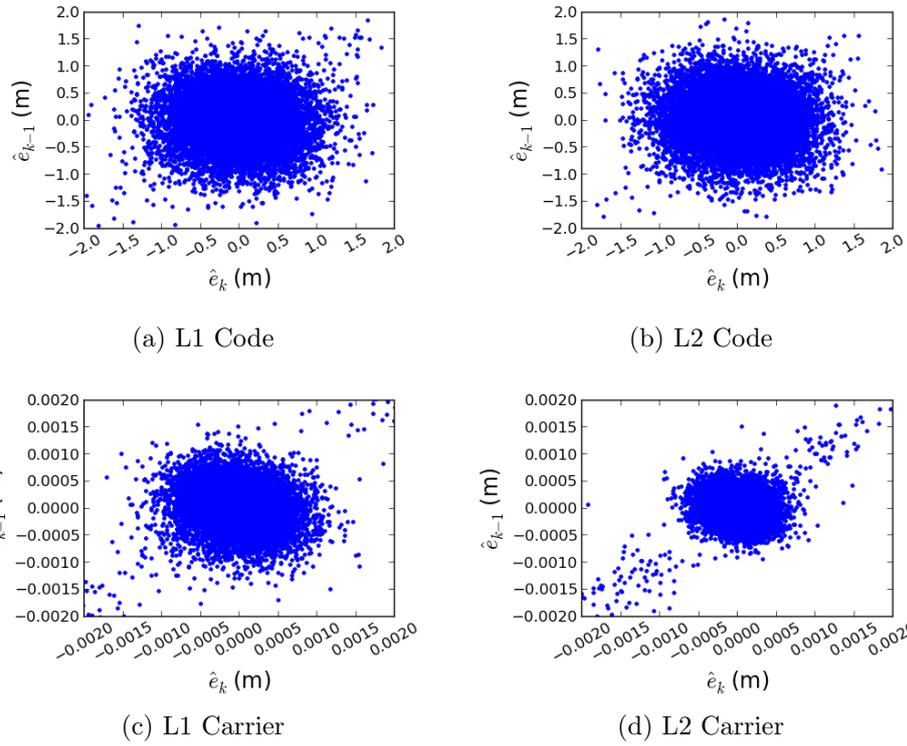
Figure 6.7: Lag plot for the whitened residuals of GPS SVN05 at GUAM.

(a) L1 Code   (b) L2 Code

(c) L1 Carrier   (d) L2 Carrier
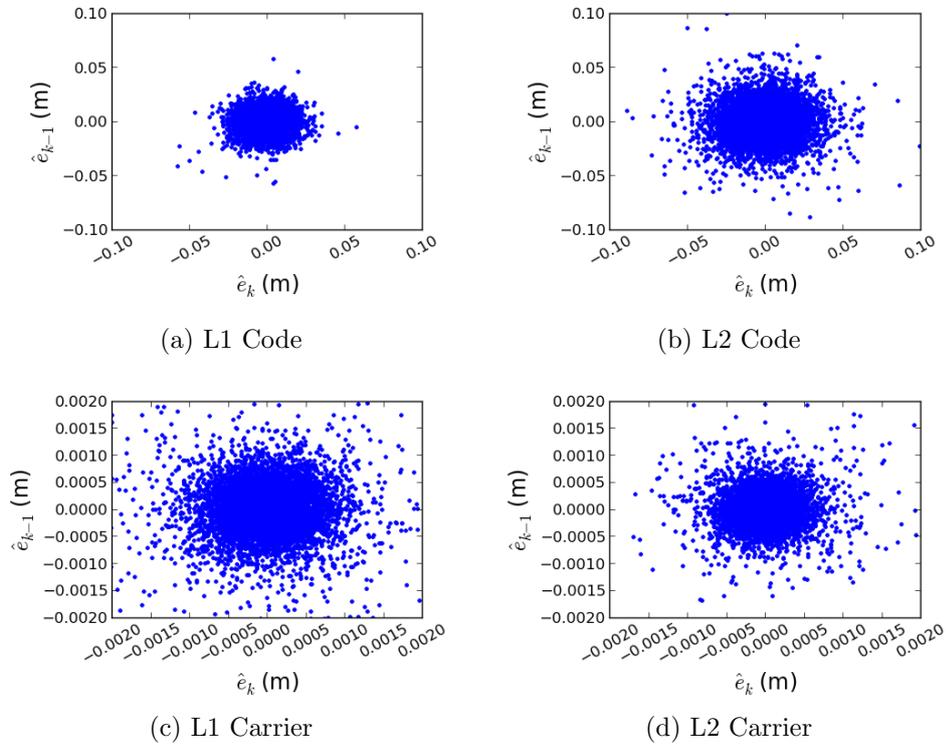
Figure 6.8: Lag plot for the whitened residuals of GPS SVN17 at ALBM.

residuals have been successfully 'whitened', although their amplitude has changed.

Durbin-Watson test statistic

The method for calculating the Durbin-Watson test statistic is given in §5.3.5. A time series of whitened residuals was created for each satellite and observation type. Once again the Durbin-Watson test statistic was calculated for a moving window of 240 seconds, where any time gap greater than the sample frequency (1 second) was not used. Figures 6.9 - 6.12 show histograms of the Durbin-Watson values for the whitened residuals for L1 code observations, L2 code observations, L1 and L2 carrier phase observations combined and all observations. The general distribution of the Durbin-Watson values with respect to the critical values are shown in Table 6.1.

Table 6.1: Distribution of Durbin-Watson values of AR(1) whitened residuals from all datasets with respect to the critical values.

Dataset	Observation type	$d < d_L^+$	$d_L^+ \leq d \leq d_U^+$	$d_U^+ < d < d_L^-$	$d_L^- \leq d \leq d_U^-$	$d > d_U^-$
ABPO	L1 Code	1.59%	25.53%	10.60%	48.04%	14.23%
	L2 Code	1.79%	28.86%	11.40%	45.81%	12.13%
	L1 & L2 Carrier	0.42%	3.59%	7.71%	34.25%	53.07%
FALK	L1 Code	2.61%	23.39%	10.54%	46.45%	17.00%
	L2 Code	3.14%	23.38%	9.36%	44.86%	19.26%
	L1 & L2 Carrier	0.51%	4.33%	14.78%	34.91%	44.33%
GUAM	L1 Code	3.69%	29.50%	12.27%	45.48%	9.06%
	L2 Code	4.46%	31.80%	11.69%	43.46%	8.58%
	L1 & L2 Carrier	4.04%	10.48%	23.22%	40.78%	18.21%
ALBM	L1 Code	3.42%	30.34%	9.96%	50.26%	6.01%
	L2 Code	2.26%	31.69%	11.24%	50.73%	4.08%
	L1 & L2 Carrier	4.91%	23.13%	10.40%	49.71%	11.85%

The effect of whitening the residuals has brought all histograms closer to 2.0. In the case of the static datasets, the histograms are skewed from the central point (2.0), with greater skewness seen for the carrier phase than code. This may be due to the effect of 'overwhitening' as a result of the process being incorrectly modelled as an AR(1) process when it should be a higher order. Underestimating the AR order of a stochastic process, referred to as underfit, can result in a bias (Broersen 2006). In the case where an AR(p) process, where $p > 1$, is estimated as an AR(1) process, the estimate of ϕ_1 may be overestimated due to the unmodelled effect of higher order lags. When applied to the whitening of residuals, this can result in
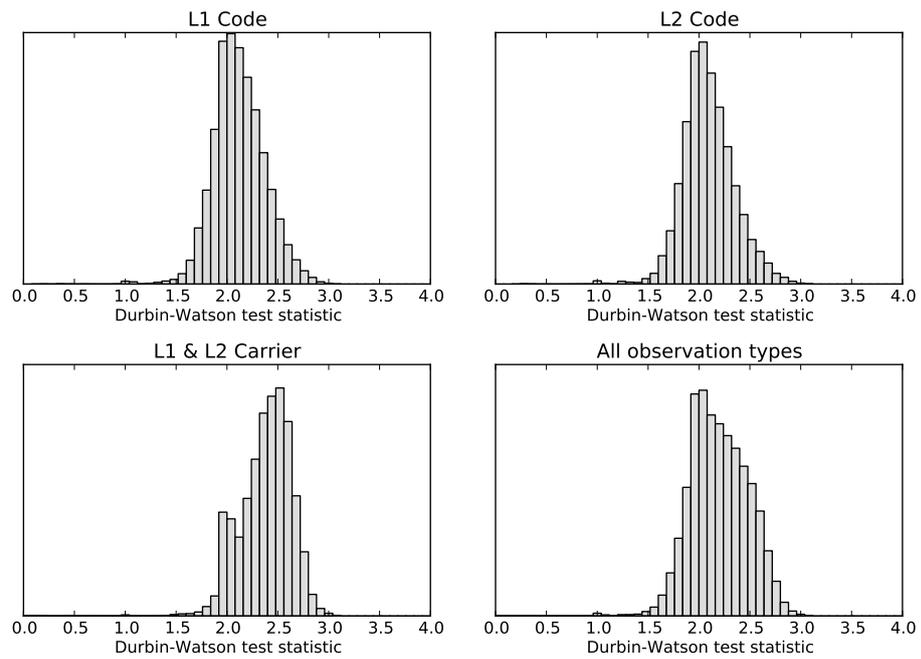
Figure 6.9: Histogram of Durbin-Watson values for the whitened residuals of ABPO.
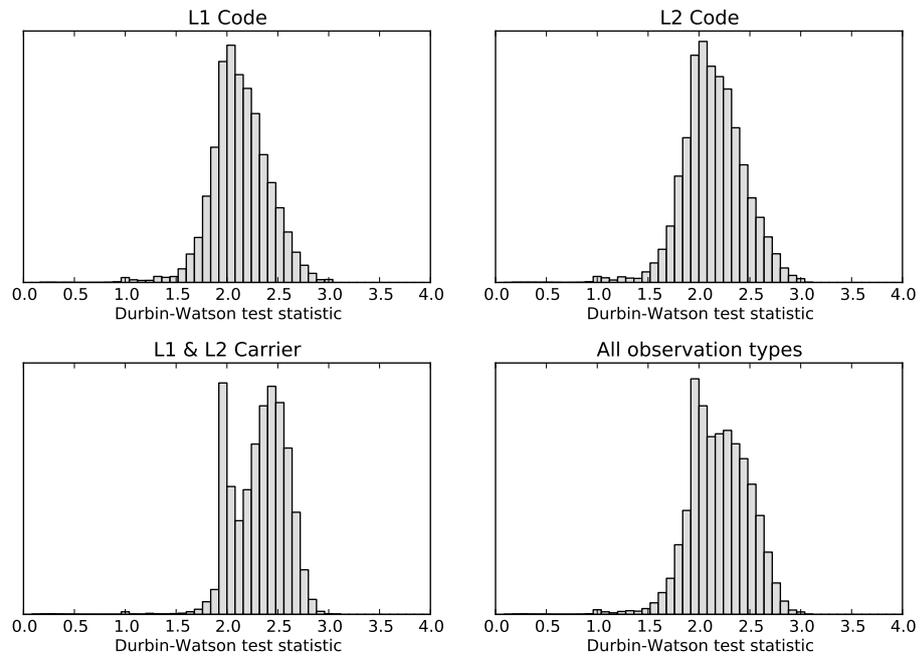
Figure 6.10: Histogram of Durbin-Watson values for the whitened residuals of FALK.
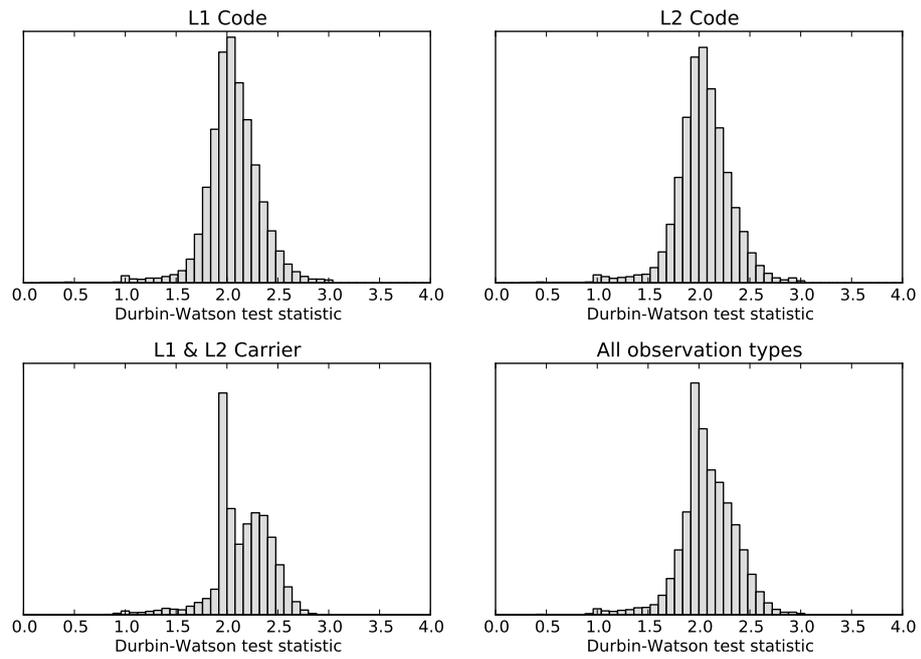
Figure 6.11: Histogram of Durbin-Watson values for the whitened residuals of GUAM.
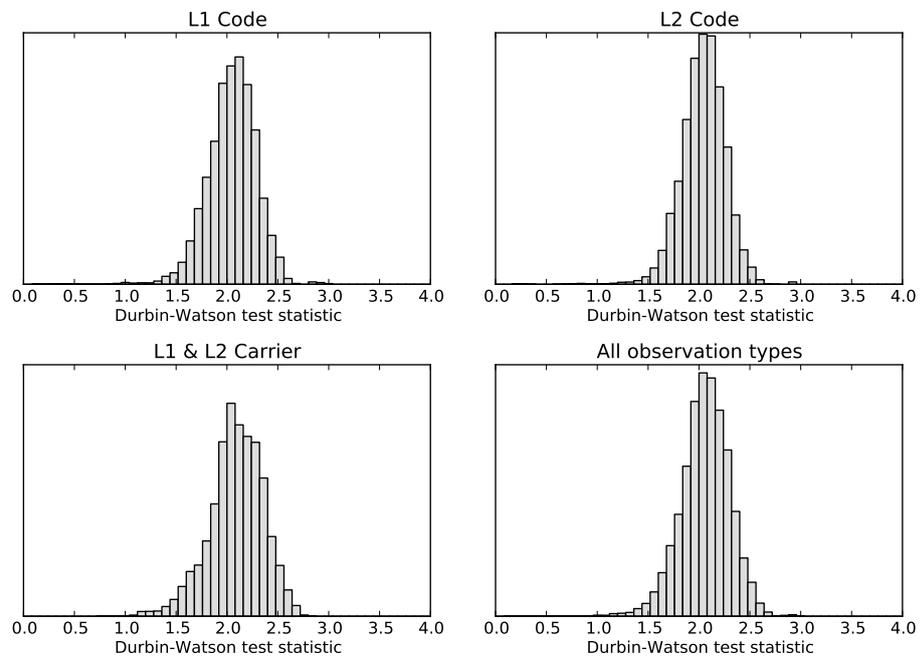
Figure 6.12: Histogram of Durbin-Watson values for the whitened residuals of ALBM.

an overestimation of the weight assigned to $\hat{\underline{e}}_{k-1}$ and therefore a greater proportion of $\hat{\underline{e}}_{k-1}$ is subtracted from $\hat{\underline{e}}_k$. This can cause the whitened residual to become negatively correlated at lag 1.

From Table 6.1 it can be seen that 53.07% of the carrier phase residuals in the ABPO dataset lie above d_U^-, hence showing that these values are significantly negatively correlated. This effect can also be seen in the carrier phase residuals for the FALK dataset. In many of the histograms there appear to be two peaks, namely one around 2.0 and a second at a higher value. This supports the idea that some residuals have been 'overwhitened', whereas others have been sufficiently whitened. The carrier phase residuals for the ALBM dataset appear to be less skewed, although the peak of its histogram appears to be greater than 2.0.

By contrast, the code residuals for all datasets appear more centred around 2.0, although the data in Table 6.1 shows a higher proportion of values in the region of inconclusivity close to negative correlation ($d_U^- < d < d_L^-$) and once again suggests that 'overwhitening' is occurring. A slightly lower proportion of values were in this region for the original residuals (see Table 5.2), but in the case of the AR(1) whitened code residuals these regions can be considered the tails of the overall distribution of Durbin-Watson values.

What is significant is that a very small proportion of all Durbin-Watson values are in the region that would indicate that the residuals are positively correlated. It can therefore be concluded that the time correlation has been successfully removed, although in some cases an AR(1) process does not appear to have been sufficient.

Autocorrelation function

The method for calculating the autocorrelation function is given in §5.3.6. Eq. (5.29) was used to calculate the autocorrelation function for the whitened residuals of each satellite and observation type pair. As before each time series was divided into 240 second blocks and the autocorrelations for lags $\{1, \ldots, 120\}$ were calculated. The 95% confidence region was calculated using Eq. (5.32).

Figures 6.13 - 6.16 show the autocorrelation graphs for the whitened residuals

of the ABPO, FALK, GUAM and ALBM datasets. The same satellites chosen for the lag plots in §5.3.6 have once again been used here so that comparisons can be made.

In all cases the autocorrelation plots are much more in line with what is expected for random noise (see Figure 5.19a for reference), but in the case of the static datasets the autocorrelation values up to approximately lag 10 are outside the 95% confidence region. Considering that the Durbin-Watson values for the same datasets and observation types were shown to be slightly above 2.0 in §6.7.3 this could support the theory that the AR(1) estimation may be due to over-whitening of the residuals and that a higher order of AR model may be required (see §6.8). In the case of the ALBM dataset the whitening appears to have the greatest effect, although it can be seen from Figure 6.16 that the whitening does not appear to be incomplete as seen with the static datasets.

With the exception of FALK, the carrier phase residuals all have autocorrelations that are within the 95% confidence region for all lags. The L1 and L2 carrier phase autocorrelations at lag 1 for FALK (see Figures 6.14c and 6.14d) are negative and lie outside the 95% confidence region. In Figure 6.10 it can be seen that the highest proportion of Durbin-Watson values lie above 2.0, matching what is seen in the autocorrelation plot. Once again this could be due to incomplete whitening of the residual as a result of an incorrect AR order.

Cross-correlation matrices

In §5.3.7 it was particularly noteworthy that, not only were residuals autocorrelated, but there were also high cross-correlations. This meant that the residual of one satellite and observation type pair was correlated with the residual for a different satellite and observation type pair at a lagged epoch. In this section the AR coefficients have been determined by a univariate approach, i.e. using the autocorrelations and not the cross-correlation. It is important to validate that not only have the correlations within a satellite and observation type pair been removed, or at the very least reduced, but also that this has reduced the cross-correlations. Conversely,

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
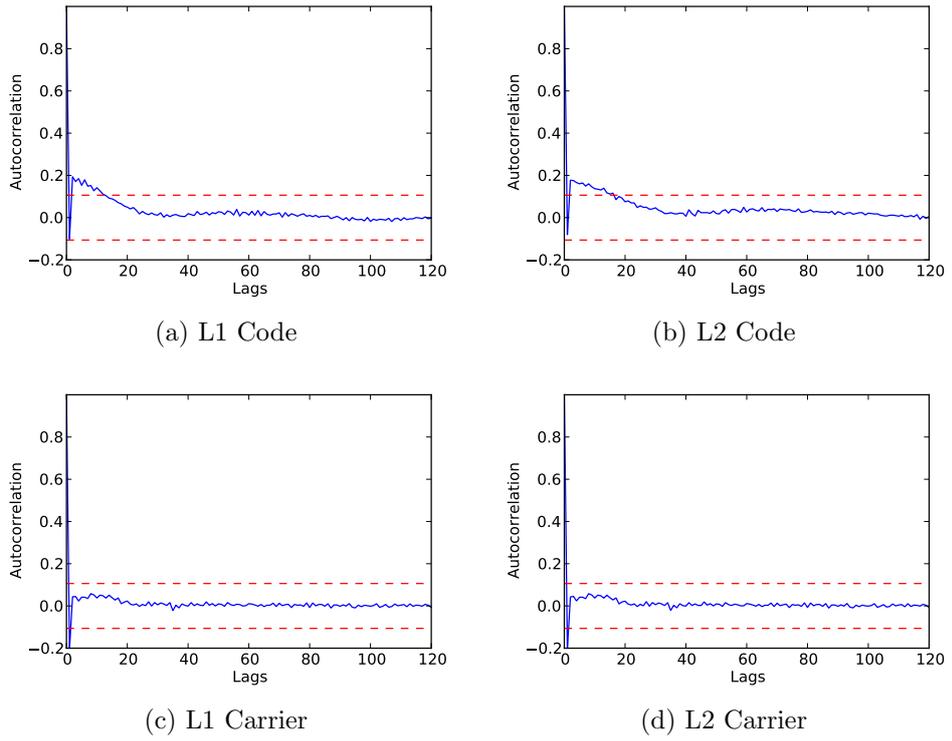
Figure 6.13: Autocorrelation plot for the whitened residuals of GPS SVN05 at ABPO.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
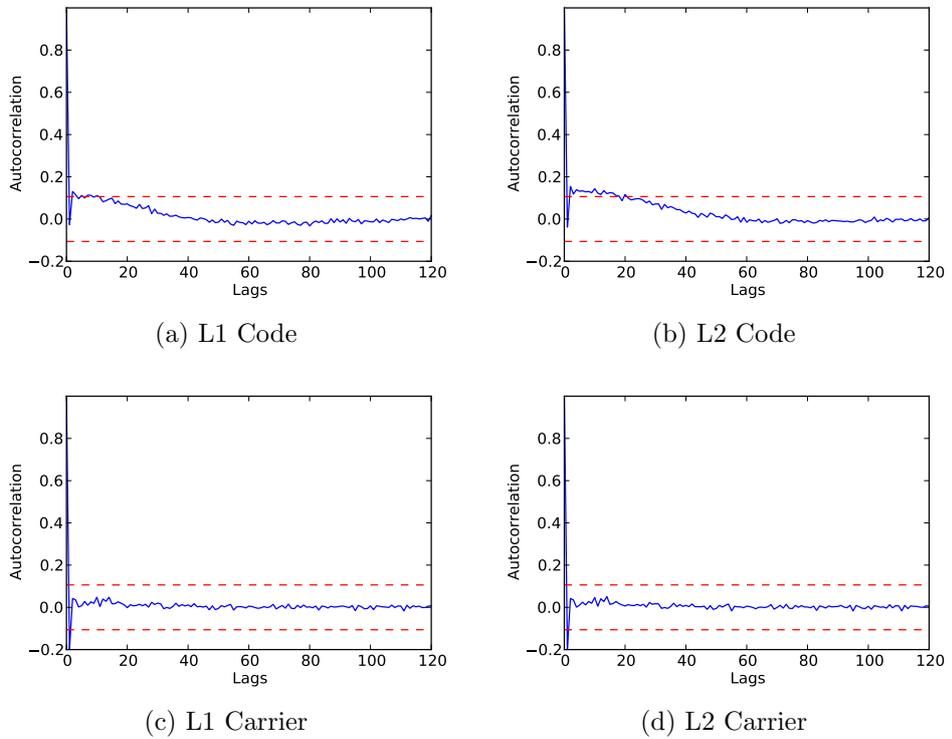
Figure 6.14: Autocorrelation plot for the whitened residuals of GPS SVN05 at FALK.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
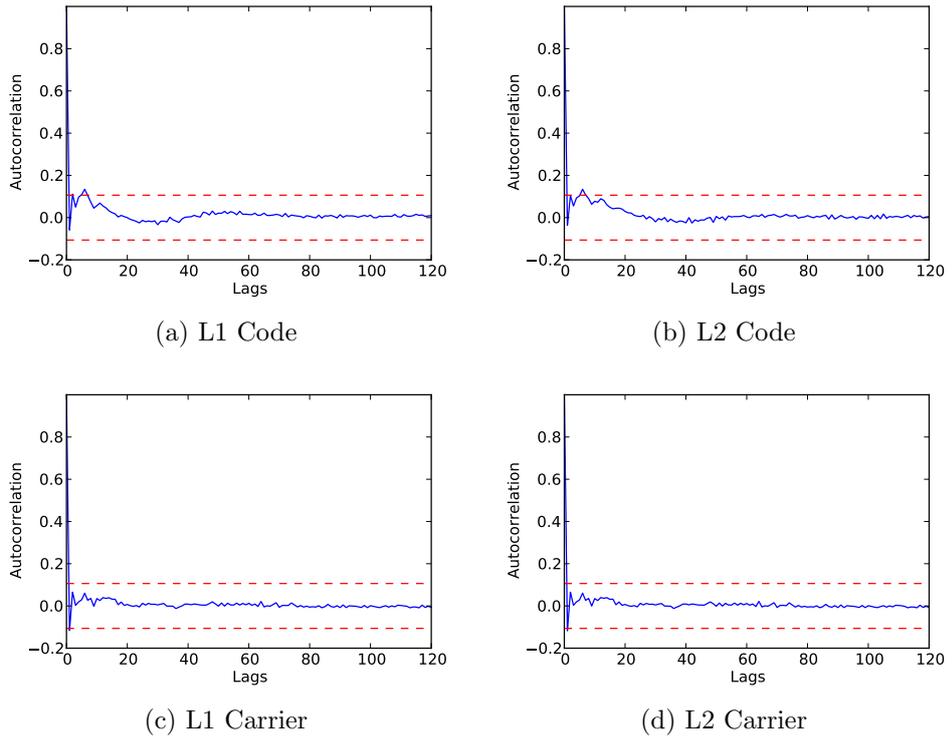
Figure 6.15: Autocorrelation plot for the whitened residuals of GPS SVN05 at GUAM.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
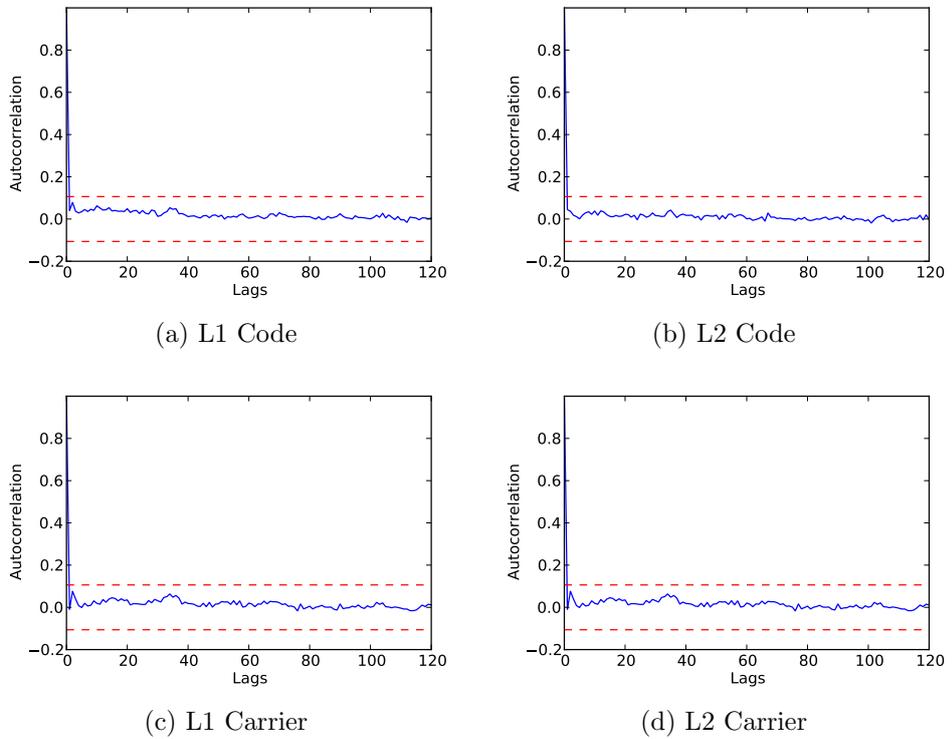
Figure 6.16: Autocorrelation plot for the whitened residuals of GPS SVN17 at ALBM.

it is also important to ensure that no cross-correlations have been introduced.

The cross-covariance and cross-correlation matrices are calculated as per the method set out in §5.3.7 using Eq. (5.36) and (5.37). The same sample size of 240 seconds is used and only common satellite and observation type pairs are included.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
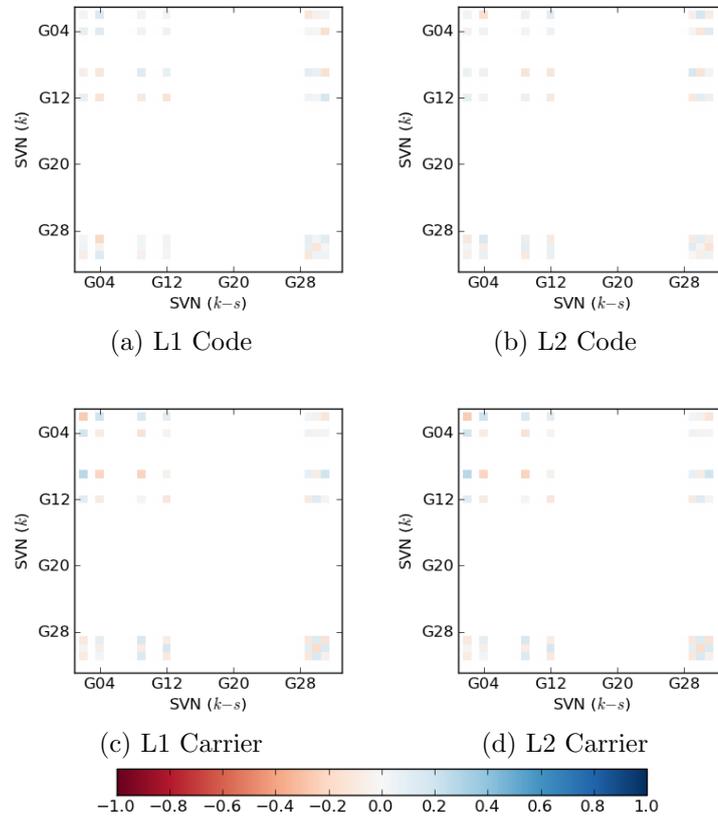
Figure 6.17: Lag 1 cross-correlation plot for ALBM dataset.

The cross-correlations for lags 1, 5, 10, 15 and 30 are shown in Figures 6.17 - 6.21. It is immediately clear that the correlations are much smaller than seen previously. The values in Table 6.2 support this, with mean values that are very close to zero, along with a small standard error, and very little deviation from the mean. As the lag time increases the range of values, as shown by the minimum and maximum values in Table 6.2 decreases as expected. It is interesting to note that the mean and standard deviations appear to be around the same value. However, using the level of confidence given by Eq. (5.32), values where $\rho_{ij}(s) < 0.1061$ are considered insignificant.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
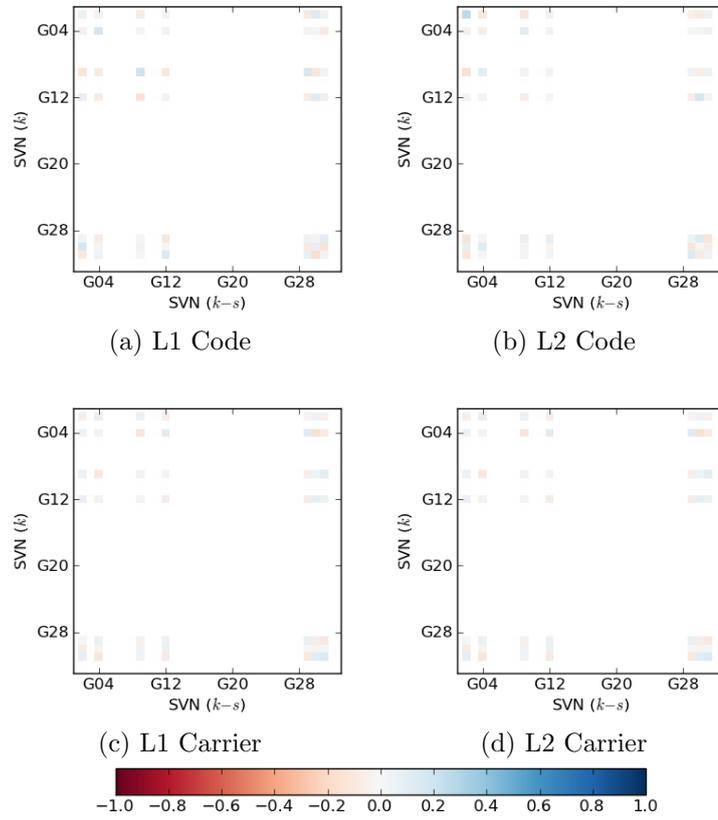
Figure 6.18: Lag 5 cross-correlation plot for ALBM dataset.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
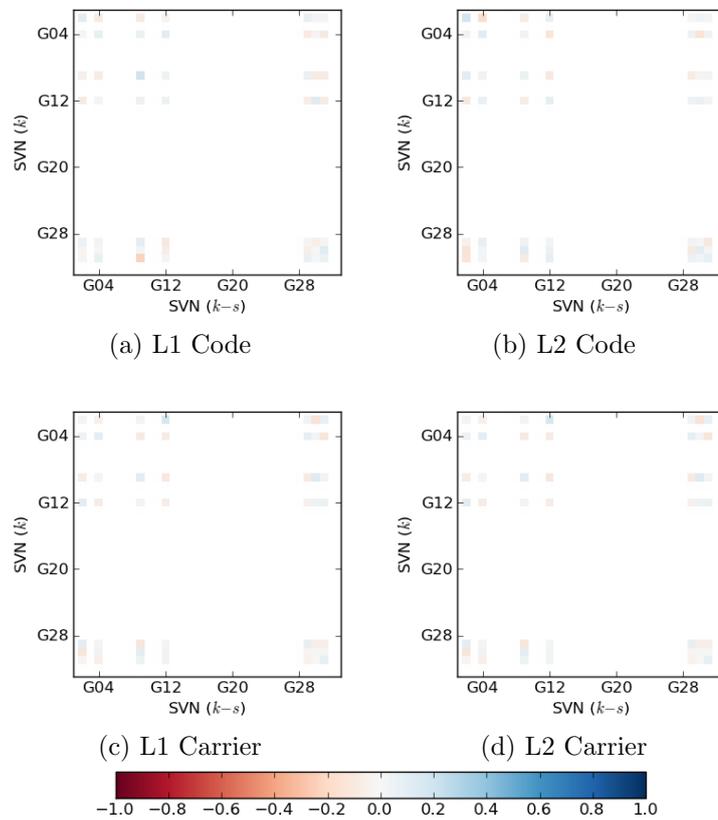
Figure 6.19: Lag 10 cross-correlation plot for ALBM dataset.

(a) L1 Code (b) L2 Code

(c) L1 Carrier (d) L2 Carrier
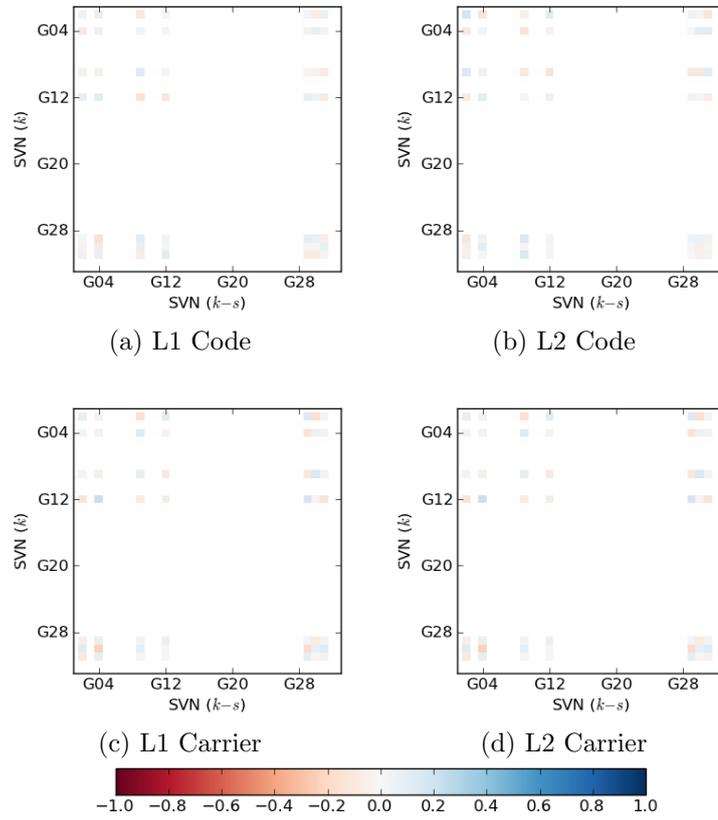
Figure 6.20: Lag 15 cross-correlation plot for ALBM dataset.

(a) L1 Code (b) L2 Code

(c) L1 Carrier (d) L2 Carrier
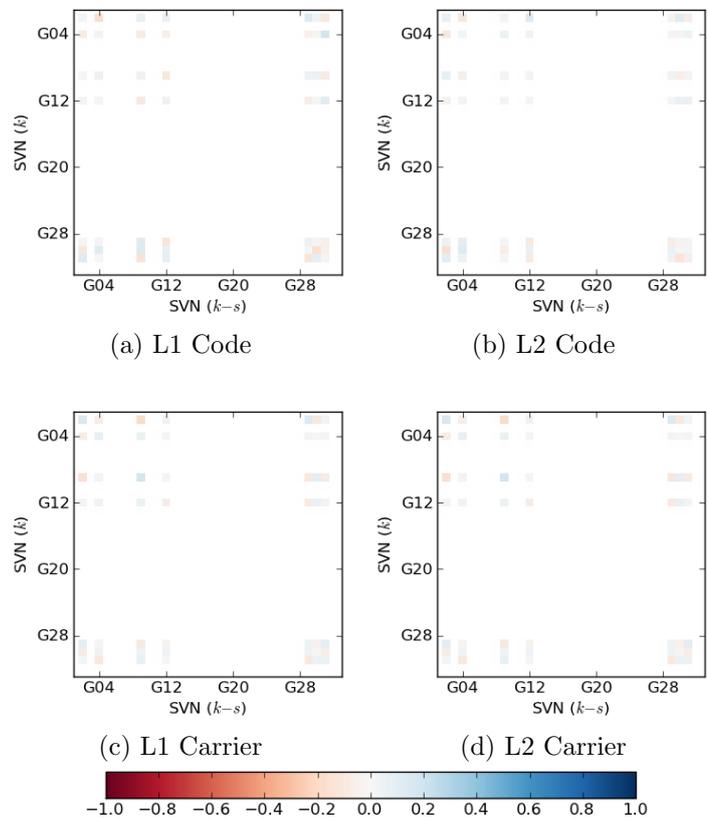
Figure 6.21: Lag 30 cross-correlation plot for ALBM dataset.

Table 6.2: Minimum, maximum, mean, standard error and standard deviation of cross-correlation values of AR(1) whitened residuals for all satellites and observation types for ALBM dataset at lags 1, 5, 10, 15 and 30.

Lag	Observation type	Min	Max	Mean	St. err.	St. dev.
1	L1 Code	-0.1922	0.1605	-2.438×10^{-4}	2.560×10^{-3}	0.0794
	L2 Code	-0.1435	0.1647	8.554×10^{-4}	2.604×10^{-3}	0.0807
	L1 & L2 Carrier	-0.2482	0.2704	-1.770×10^{-4}	3.769×10^{-3}	0.1168
	All observations	-0.2704	0.2704	5.829×10^{-5}	7.142×10^{-4}	0.0886
5	L1 Code	-0.1792	0.1972	-7.454×10^{-5}	2.781×10^{-3}	0.0862
	L2 Code	-0.1362	0.2520	9.120×10^{-4}	2.534×10^{-3}	0.0786
	L1 & L2 Carrier	-0.1533	0.1256	4.621×10^{-4}	2.127×10^{-3}	0.0659
	All observations	-0.2471	0.2520	2.726×10^{-5}	5.785×10^{-4}	0.0717
10	L1 Code	-0.2259	0.1808	2.721×10^{-4}	2.263×10^{-3}	0.0702
	L2 Code	-0.1617	0.1651	1.806×10^{-3}	2.339×10^{-3}	0.0725
	L1 & L2 Carrier	-0.1386	0.1512	6.505×10^{-4}	2.149×10^{-3}	0.0666
	All observations	-0.2259	0.2253	2.152×10^{-4}	5.790×10^{-4}	0.0718
15	L1 Code	-0.1375	0.1079	7.122×10^{-5}	2.030×10^{-3}	0.0629
	L2 Code	-0.1157	0.1674	9.295×10^{-5}	2.344×10^{-3}	0.0727
	L1 & L2 Carrier	-0.2301	0.2150	4.040×10^{-4}	2.990×10^{-3}	0.0927
	All observations	-0.2301	0.2302	-2.057×10^{-5}	6.272×10^{-4}	0.0778
30	L1 Code	-0.1587	0.1760	5.897×10^{-6}	2.520×10^{-3}	0.0781
	L2 Code	-0.1629	0.1182	2.011×10^{-4}	1.975×10^{-3}	0.0612
	L1 & L2 Carrier	-0.1776	0.1672	1.509×10^{-4}	2.304×10^{-3}	0.0714
	All observations	-0.2110	0.1778	2.042×10^{-5}	5.629×10^{-4}	0.0698

6.8 Post-fit residuals as an AR(p) model

Since it is possible to partially whiten the post-fit residual using an AR(1) model the question arises as to whether a higher order model can improve the result. Using the model in Eq. (6.21) the post-fit residual at epoch k expressed as an AR(p) process becomes

$$\hat{\underline{e}}_k - \phi_1 \hat{\underline{e}}_{k-1} - \cdots - \phi_p \hat{\underline{e}}_{k-p} = \underline{\varepsilon}_k \qquad (6.116)$$

where $\underline{\varepsilon}_k$ is the 'true' or whitened residual at epoch k.

Instead of fixing the order of autocorrelation to one particular value it is going to be assumed that the order can change with time. It is now important to correctly identify the model order that characterises the residuals at each epoch. A number of criteria for determining model order were discussed in §6.5. Given its wide use, Akaike's Information Criterion (AIC) has been chosen as the preferred method for calculating model order (see §6.5.4).

6.8.1 Estimation of the AR(p) coefficients

The implementation of the estimation routine described in §6.7 was extended to estimate the AR coefficients order $p > 1$. Since Burg's method is built upon Levinson-Durbin recursion (see §6.4.2) an AR(1) model is first estimated and with each iteration the next order of model is estimated. In addition to the AR coefficients, the residual variance for each model order is also calculated and this is used to determine the best AR order (see §6.3). The AIC value for each model order was calculated and the minimum value chosen to determine the model order. A sample size of 240 seconds was chosen to maintain consistency with the data analysis performed in Chapter 5 and the AR(1) estimation in §6.7. Model orders up to $p = 120$ were chosen.

Figures 6.22 - 6.25 show the frequency at which each AR model order has been chosen using AIC for the ABPO, FALK, GUAM and ALBM datasets. For each dataset values for all satellites are included, but separated by observation type. Due to the scaling given by Eq. (5.24) the L1 and L2 carrier phase residuals are shown together as one plot. In all cases for the carrier phase residuals an AR(1) process is determined most frequently. This is the case for both both 'static' and 'kinematic' datasets, although for the latter the frequency with order 1 is much greater than higher orders.

The code residuals for the 'static' datasets are mostly estimated as higher order processes than the carrier phase residuals, with $4 \leq p \leq 6$. This could be caused by longer period multipath signals. The exception to this are the L2 code residuals for GUAM. It is also interesting to note that for all 'static' datasets there are two peaks for the code residuals, with one at $p = 1$ and a subsequent peak at $4 \leq p \leq 6$. In the case of the 'kinematic' ALBM dataset all code residuals are primarily estimated as an AR(1) process.
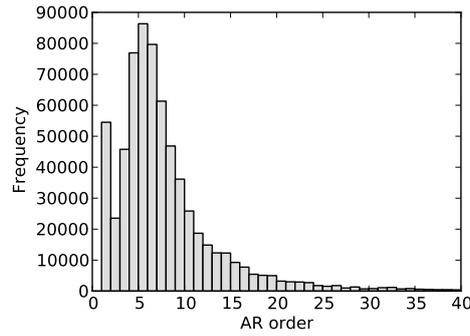
(a) L1 Code
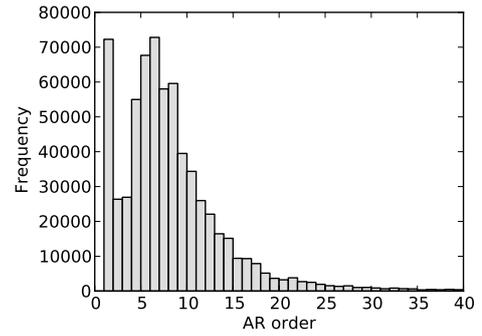
(b) L2 Code
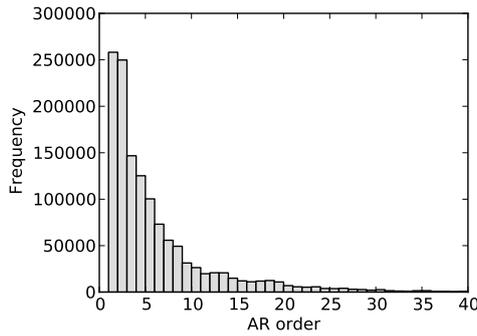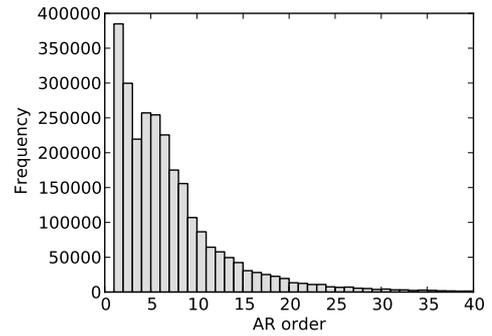
(c) L1 & L2 Carrier

(d) All observation types

Figure 6.22: Histogram showing the frequency at which each AR order has been chosen using AIC for ABPO dataset.
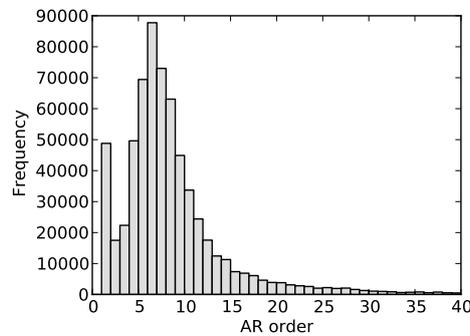
(a) L1 Code
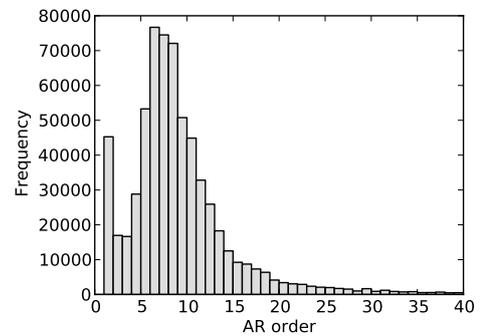
(b) L2 Code
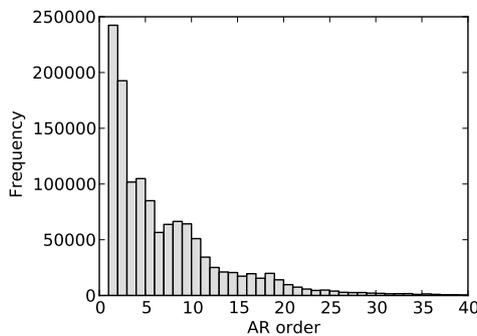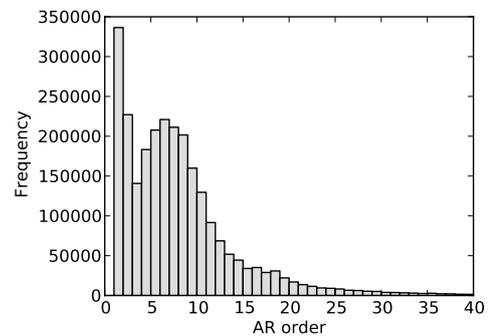
(c) L1 & L2 Carrier

(d) All observation types

Figure 6.23: Histogram showing the frequency at which each AR order has been chosen using AIC for FALK dataset.
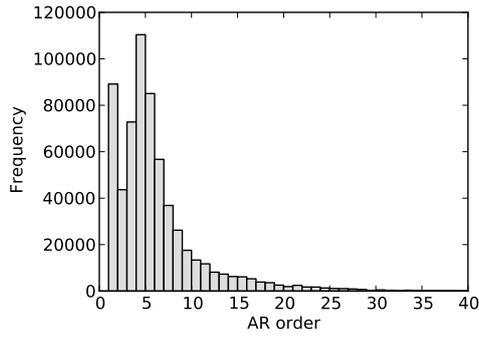
(a) L1 Code (b) L2 Code
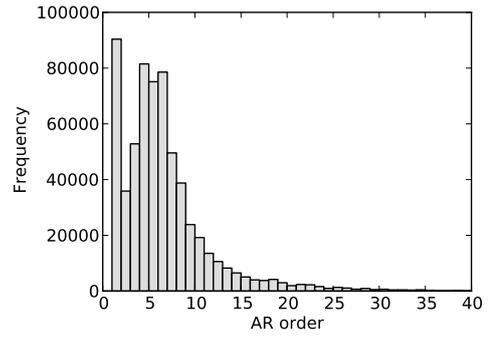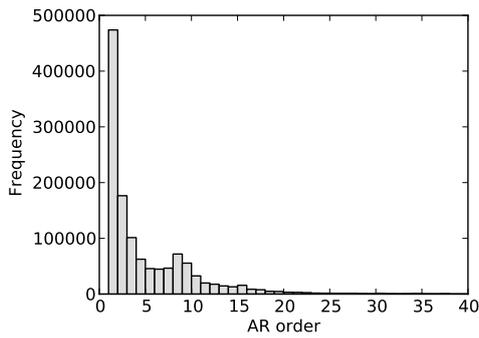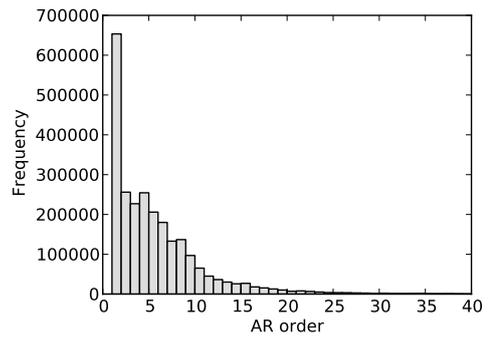
(c) L1 & L2 Carrier (d) All observation types

Figure 6.24: Histogram showing the frequency at which each AR order has been chosen using AIC for GUAM dataset.
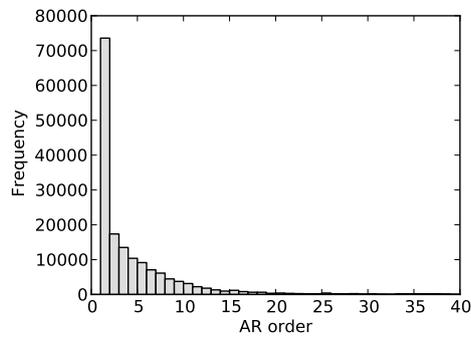
(a) L1 Code (b) L2 Code
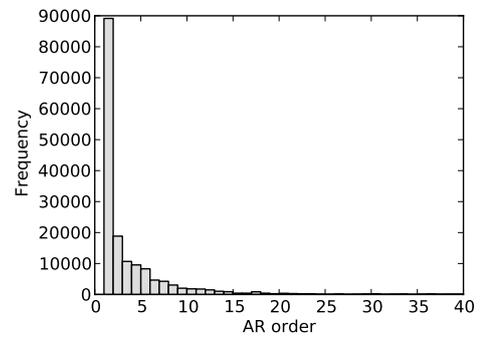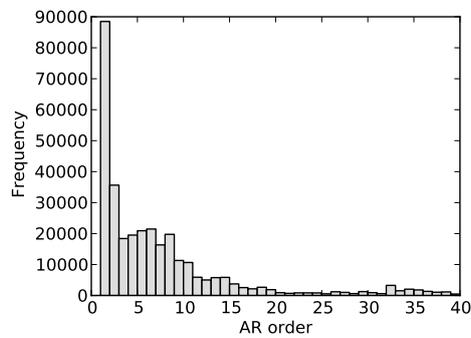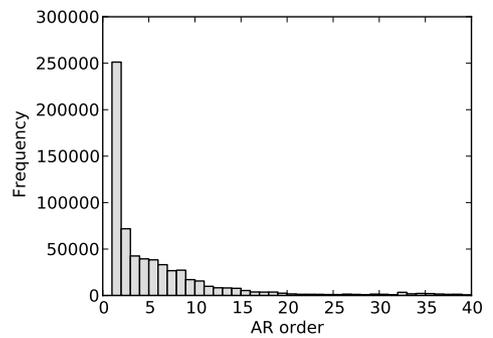
(c) L1 & L2 Carrier (d) All observation types

Figure 6.25: Histogram showing the frequency at which each AR order has been chosen using AIC for ALBM dataset.

6.8.2 Whitening the residual

As in §6.7.2 it is possible to use the coefficients estimated from an exemplar 240 second moving window to whiten the time series using

$$\underline{\varepsilon}_k = \underline{\hat{e}}_k - \phi_1 \underline{\hat{e}}_{k-1} - \cdots - \phi_p \underline{\hat{e}}_{k-p} \tag{6.117}$$

Each residual is whitened based on an $AR(p)$ estimate using the previous 240 epochs. Using the same methodology used in §6.7.2, if there is any gap in the data then there will be a latency of 240 seconds before the AR coefficient can be estimated again in order to ensure that continuous data is used.

6.8.3 Testing for whiteness

Consistent with §6.7.3, in order to determine whether the whitened residuals can actually said to be characteristic of random noise the methods for measuring correlation used in §5.3 are repeated in this section using the whitened residuals. Wherever possible the same data will be used so that direct comparisons can be made.

AR model order

Figures 6.26 – 6.29 show the AR model orders, i.e. the values of p, chosen for the results presented in this section. It can be seen that the AR models chosen are generally of order 30 or below. By listing each successive AR order along with the number of times that particular AR order was chosen it is possible to assess the frequency at which each model order is chosen. By further calculating the cumulative frequency it is possible to find the AR model which corresponds to the 68% and 95% percentile. This data is shown in Table 6.3. From this it can be seen

Table 6.3: Maximum AR model chosen for 68% (and 95%) of all epochs. The AR model corresponding to 95% is shown in parentheses.

Site	L1 Code	L1 Carrier	L2 Code	L2 Carrier
ABPO	9 (26)	8 (27)	11 (40)	8 (27)
FALK	8 (33)	7 (30)	10 (26)	7 (28)
GUAM	8 (22)	6 (15)	8 (22)	6 (15)
ALBM	4 (25)	11 (38)	5 (26)	11 (38)

that 68% of epochs have an AR model that is 11 or less. Figures 6.26 – 6.29 also show small clusters of epochs where the values are close to the maximum model order of $p = 120$. This could suggest that the residuals at this point could be better characterised by this model order or possibly higher. However, it may also be the case that the automatic procedure for selecting the model order has incorrectly identified such a high model order.

Lag plots

The method for generating the lag plots is given in §5.3.4. Figures 6.30 - 6.33 show a lag plot for the four observation types for a chosen satellite from the ABPO, FALK, GUAM and ALBM datasets respectively. The AR coefficients have been estimated using an AR(p) model using the method described in §6.8.1 and the residuals whitened as outlined in §6.8.2.

Figures 6.30 - 6.33 show a lag plot for the four observation types for the ABPO, FALK, GUAM and ALBM datasets generated using the method outlined in §5.3.4. The same satellites chosen for the lag plots in §5.3.4 have once again been used here so that comparisons can be made.

The lag plots for the whitened residuals using an AR(1), shown in Figures 6.5 - 6.8, and an AR(p) model do not appear to differ greatly, since they both have the elliptical shape expected for random noise. However, the slight negative trend seen in the AR(1) plots, which makes the ellipse appear to be at a slight angle, is corrected for in the AR(p) plots. This is especially the case for the ABPO and FALK datasets.

The sparsely scattered values in the lower left and upper right seen in the carrier phase lag plots of the residuals for the GUAM dataset (see Figures 6.32c and 6.32d) still appear to be present in the AR(p) residuals, as shown in Figures 6.7c and 6.7d. This would suggest that either an AR process with a much higher order is needed or that the time correlation cannot be removed by autoregression. This could be possible if the time correlation could be characterised by a different model.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
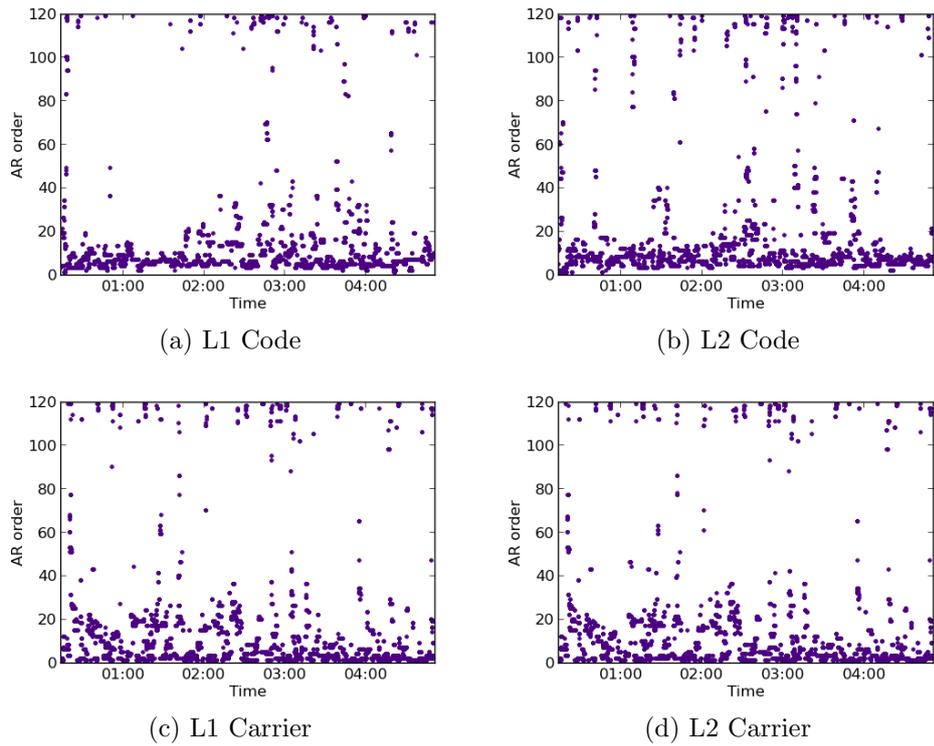
Figure 6.26: AR order chosen using AIC for GPS SVN05 at ABPO.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
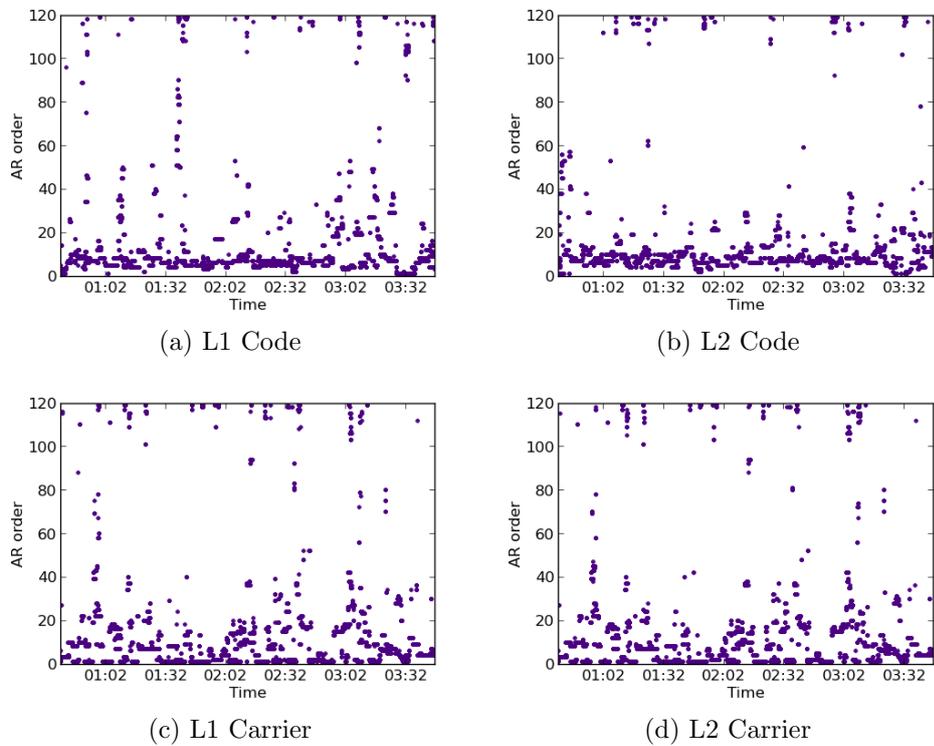
Figure 6.27: AR order chosen using AIC for GPS SVN05 at FALK.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
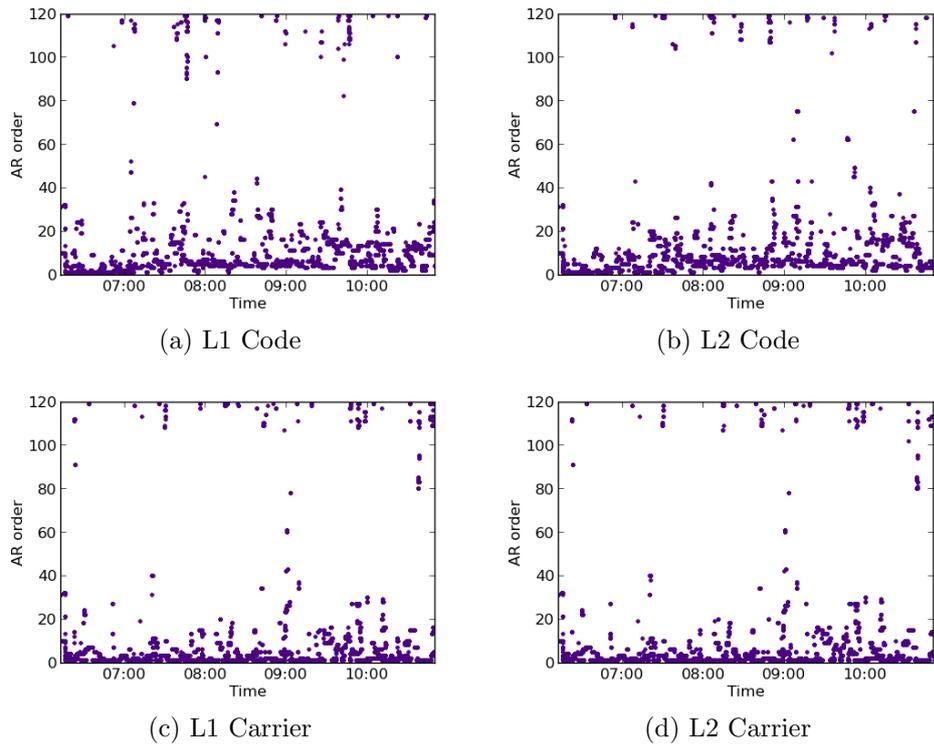
Figure 6.28: AR order chosen using AIC for GPS SVN05 at GUAM.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
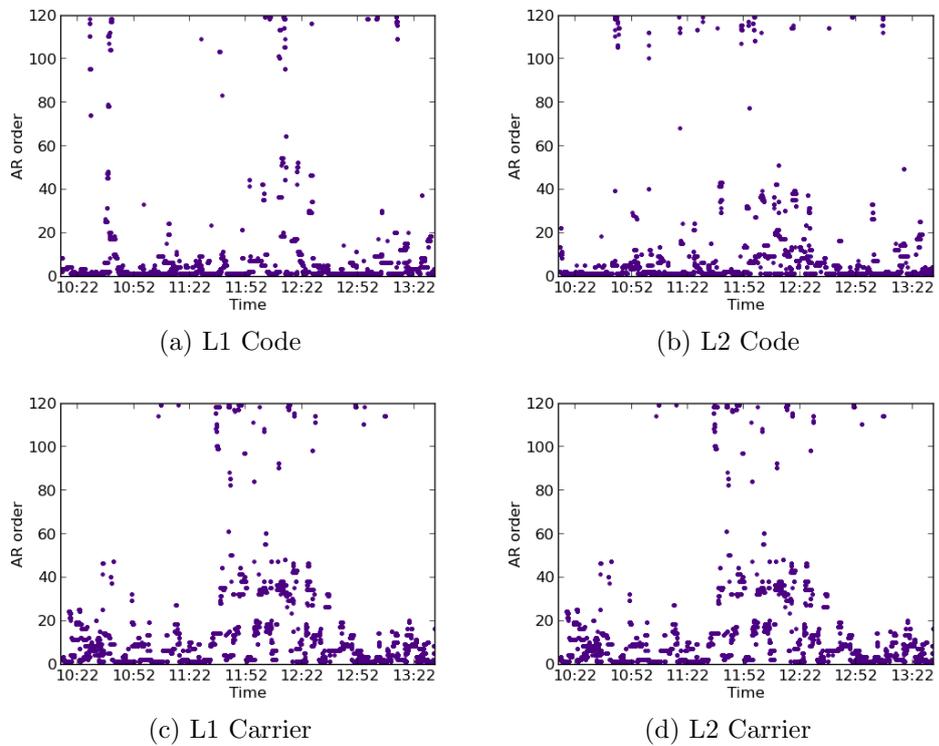
Figure 6.29: AR order chosen using AIC for GPS SVN17 at ALBM.

(a) L1 Code (b) L2 Code

(c) L1 Carrier (d) L2 Carrier
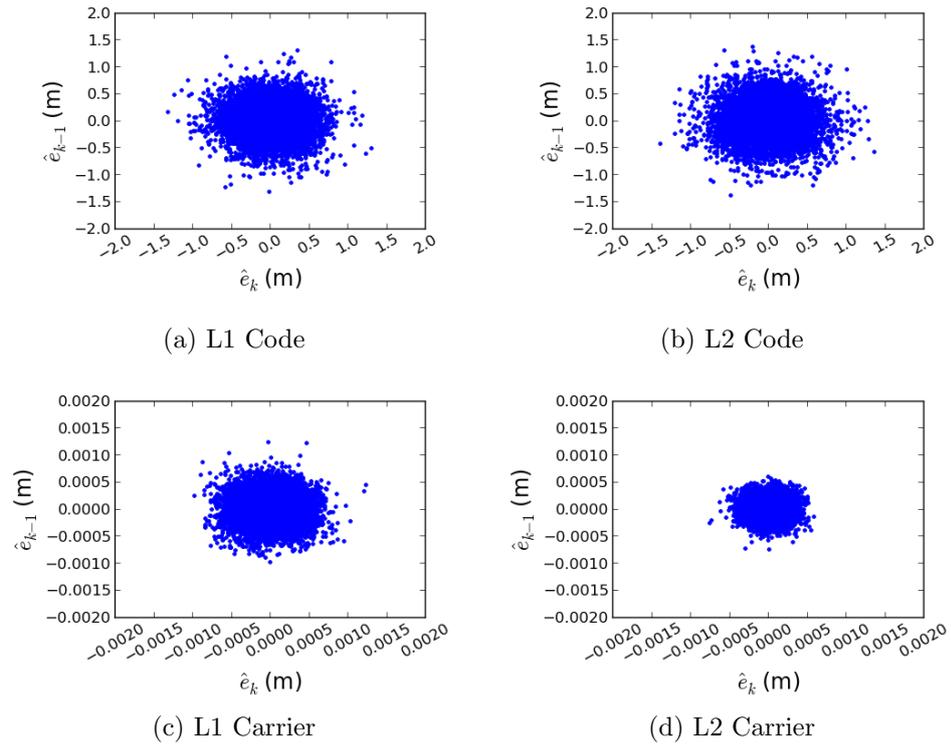
Figure 6.30: Lag plot for the AR(p) whitened residuals of GPS SVN05 at ABPO, where p is determined automatically using AIC.

(a) L1 Code (b) L2 Code

(c) L1 Carrier (d) L2 Carrier
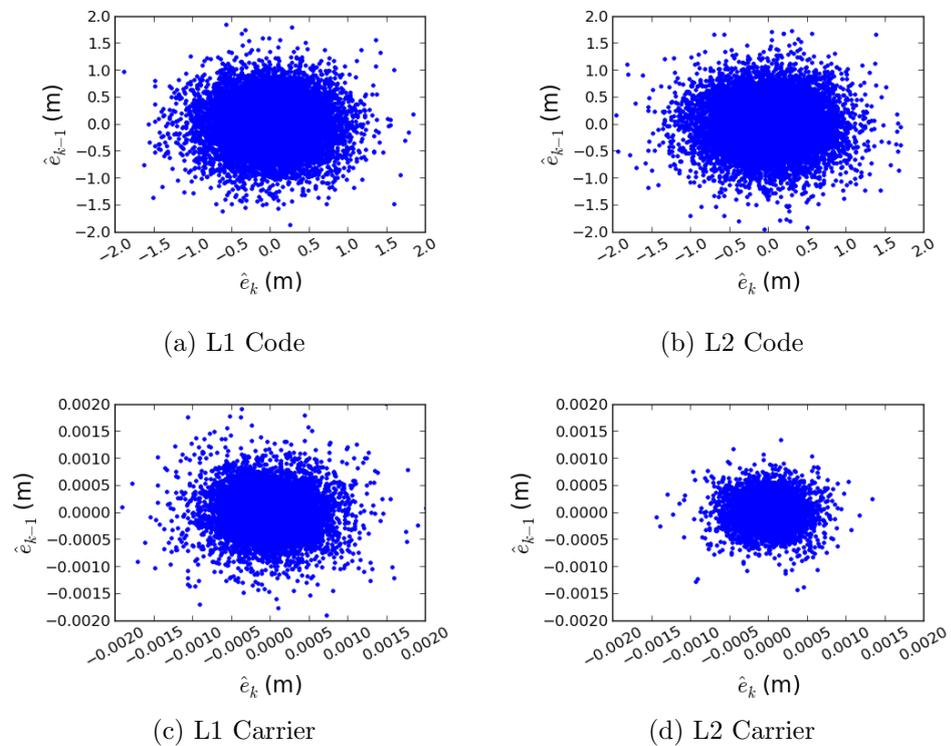
Figure 6.31: Lag plot for the AR(p) whitened residuals of GPS SVN05 at FALK, where p is determined automatically using AIC..

181

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
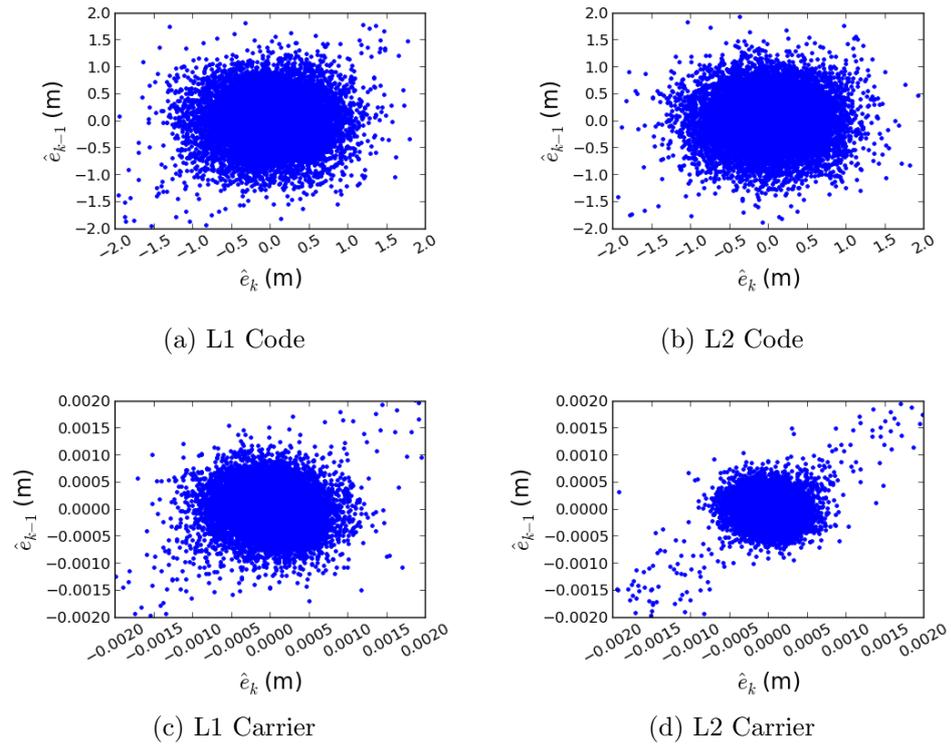
Figure 6.32: Lag plot for the AR(p) whitened residuals of GPS SVN05 at GUAM, where p is determined automatically using AIC..

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
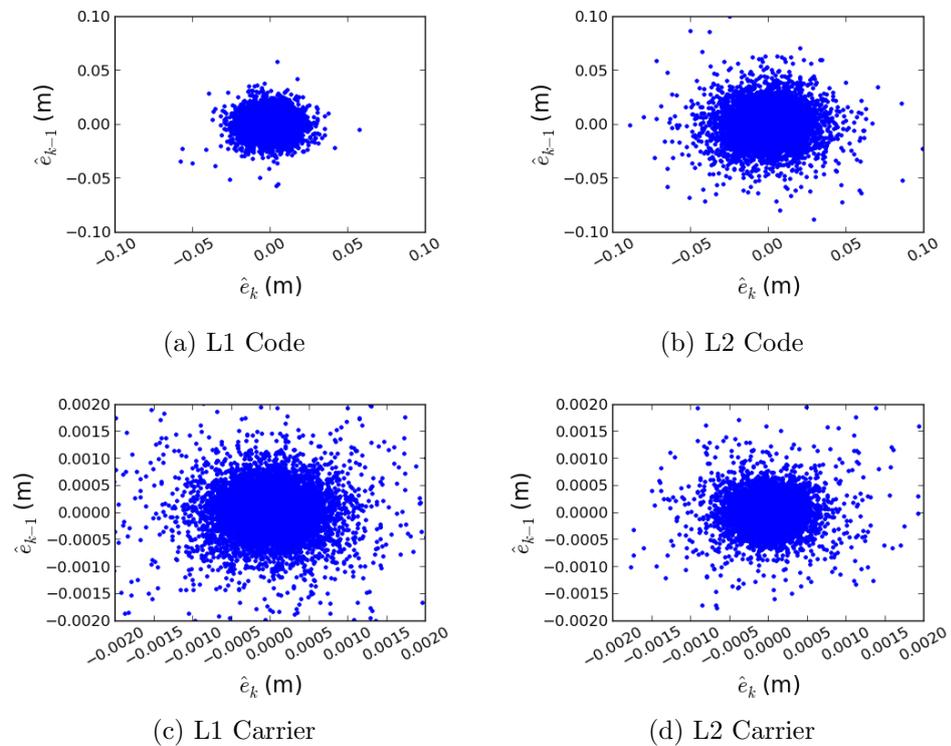
Figure 6.33: Lag plot for the AR(p) whitened residuals of GPS SVN17 at ALBM, where p is determined automatically using AIC..

Durbin-Watson test statistic

The method for calculating the Durbin-Watson test statistic is given in §5.3.5. A time series of whitened residuals was created for each satellite and observation type. Once again the Durbin-Watson test statistic was calculated for a moving window of 240 seconds, where any time gap greater than the sample frequency (1 second) was not used. Figures 6.34 - 6.37 show histograms of the Durbin-Watson values for the $AR(p)$ whitened residuals. The general distribution of the Durbin-Watson values with respect to the critical values are shown in Table 6.4.

Table 6.4: Distribution of Durbin-Watson values of $AR(p)$ whitened residuals from all datasets with respect to the critical values.

Dataset	Observation type	$d < d_L^+$	$d_L^+ \leq d \leq d_U^+$	$d_U^+ < d < d_L^-$	$d_L^- \leq d \leq d_U^-$	$d > d_U^-$
ABPO	L1 Code	1.26%	39.40%	19.57%	39.29%	0.47%
	L2 Code	1.28%	40.91%	19.09%	38.28%	0.43%
	L1 & L2 Carrier	0.37%	24.83%	19.02%	49.19%	5.61%
FALK	L1 Code	2.21%	38.83%	19.01%	38.65%	1.29%
	L2 Code	2.71%	38.65%	18.72%	38.54%	1.38%
	L1 & L2 Carrier	0.63%	24.12%	25.44%	44.45%	4.19%
GUAM	L1 Code	3.54%	38.84%	18.31%	37.47%	1.83%
	L2 Code	4.26%	39.36%	17.67%	36.58%	2.13%
	L1 & L2 Carrier	0.62%	20.40%	29.12%	42.61%	3.94%
ALBM	L1 Code	0.29%	33.10%	16.02%	48.58%	2.02%
	L2 Code	0.64%	35.27%	15.52%	46.98%	1.60%
	L1 & L2 Carrier	1.96%	40.04%	16.41%	39.90%	1.69%

Compared to the histograms for an $AR(1)$ process (see Figures 6.9 - 6.12) those for an $AR(p)$ process are much more centred around 2.0 and have a much tighter distribution. This is reflected in the data in Table 6.4, where far fewer values are in the regions indicating significant positive correlation ($d < d_L^+$). It can also be seen that there are fewer values in the region indicating significant negative correlation ($d > d_U^-$), which justifies the use of a higher order model as the residuals are no longer 'overwhitened'. Many values can still be found in the two regions of inconclusivity ($d_L^+ \leq d \leq d_U^+$ and $d_L^- \leq d \leq d_U^-$).

The $AR(p)$ whitened carrier phase residual histograms appear slightly skewed towards negative correlation. This is supported by the data in Table 6.4, where there are almost twice as many values in the region $d_L^+ \leq d \leq d_U^+$ as in $d_L^- \leq d \leq d_U^-$. This could suggest that even higher order models may be required to completely whiten the carrier phase residuals. As model orders up to 120 seconds were considered,
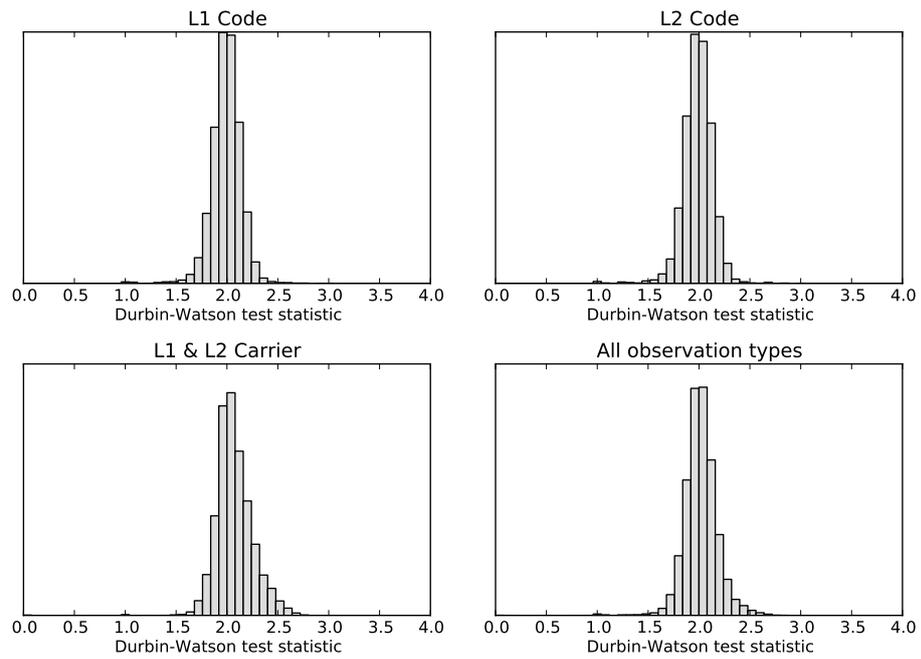
Figure 6.34: Histogram of Durbin-Watson values for the whitened residuals of ABPO.
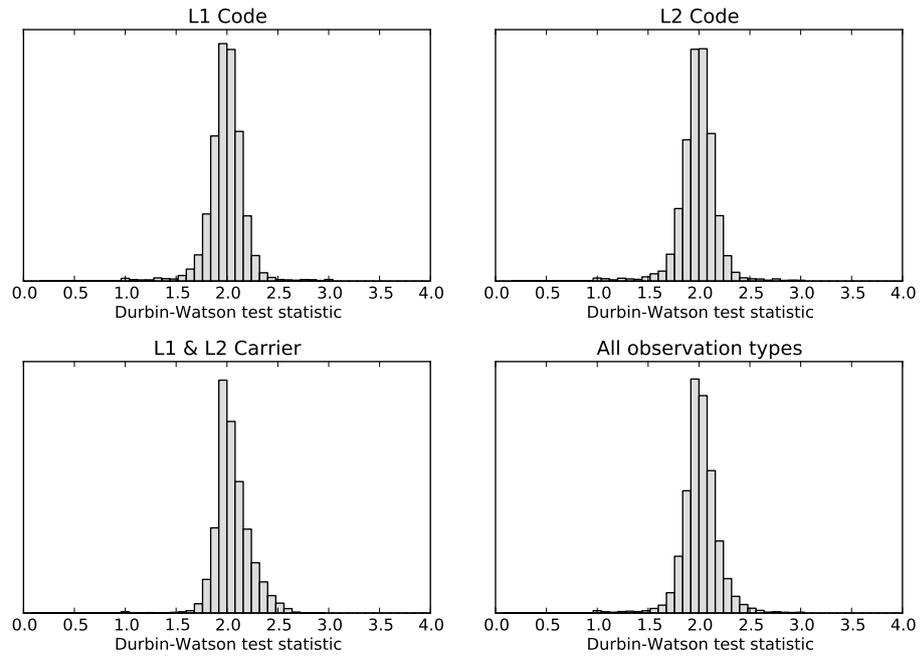
Figure 6.35: Histogram of Durbin-Watson values for the whitened residuals of FALK.
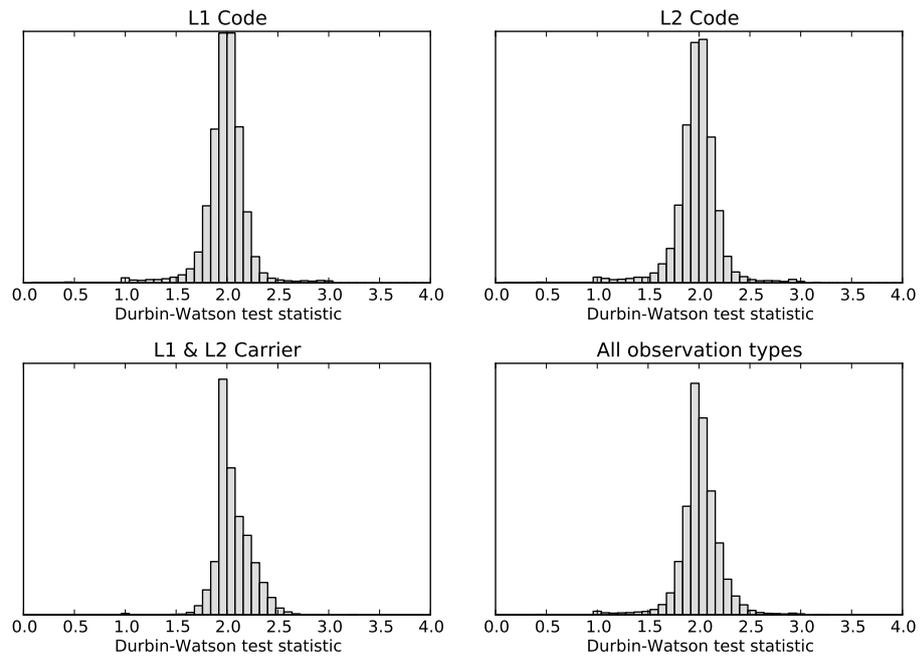
Figure 6.36: Histogram of Durbin-Watson values for the whitened residuals of GUAM.
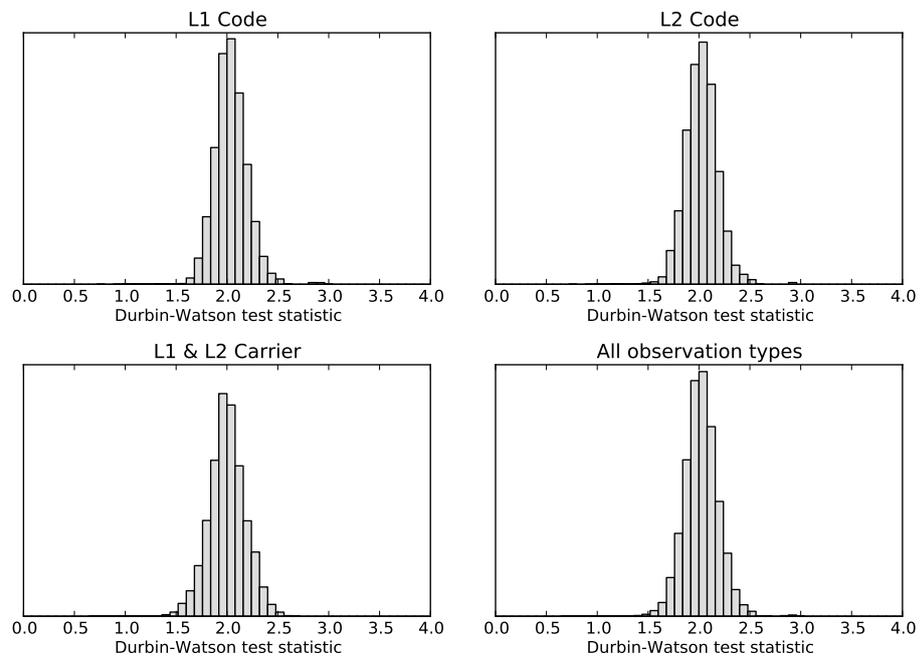
Figure 6.37: Histogram of Durbin-Watson values for the whitened residuals of ALBM.

this suggests very high order processes. If higher orders were to be estimated then greater sample sizes would be required to ensure a stable estimate.

To conclude, the histograms and data in Table 6.4 show that the by estimating the residuals as an $AR(p)$ process results in less 'overwhitening' than for the $AR(1)$ whitened residuals. This appears particularly successful for the code residuals, although the carrier phase residuals for the 'static' datasets may require even higher order AR models.

Autocorrelation function

The method for calculating the autocorrelation function is given in §5.3.6. Eq. (5.29) was used to calculate the autocorrelation function for the whitened residuals of each satellite and observation type pair. As before each time series was divided into 240 second blocks and the autocorrelations for lags $\{1, \ldots, 120\}$ were calculated. The 95% confidence region was calculated using Eq. (5.32).

Figures 6.38 - 6.41 show the autocorrelation graphs for the $AR(p)$ whitened residuals of the ABPO, FALK, GUAM and ALBM datasets. In each case the same satellite as used in §5.3.6 and §6.7.3 are used for comparison.

The autocorrelation plots show that in all cases for lags greater than zero the autocorrelation lies with in the 95% confidence area, therefore showing that the whitened residuals are not correlated with any other epoch within the same satellite and observation type pair.

Cross-correlation matrices

Although in §6.7.3 the cross-correlations of the $AR(1)$ whitened residuals were very low, it is interesting to see if by estimating higher model orders these values can be further minimised. The cross-covariance and cross-correlation matrices are calculated as per the method set out in §5.3.7 using Eq. (5.36) and (5.37). The same sample size of 240 seconds is used and only common satellite and observation type pairs are included.

The cross-correlations for lags 1, 5, 10, 15 and 30 are shown in Figures 6.42 - 6.46 along with a numerical summary in Table 6.5. Once again the values are much
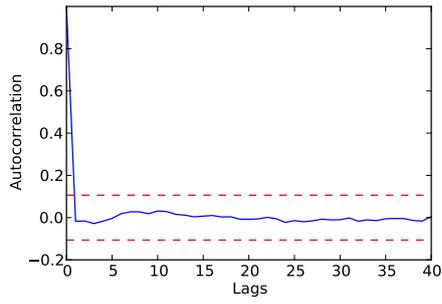
(a) L1 Code
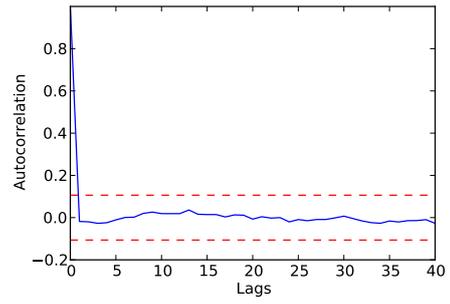
(b) L2 Code
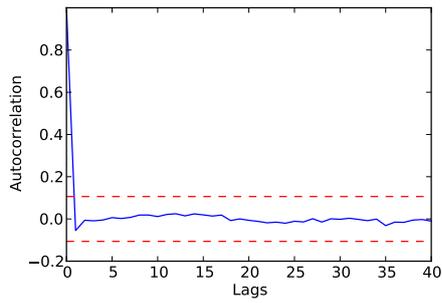
(c) L1 Carrier
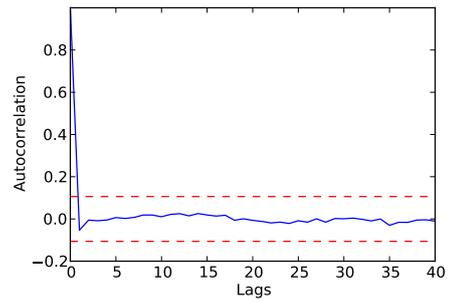
(d) L2 Carrier

Figure 6.38: Autocorrelation plot for the AR(p) whitened residuals of GPS SVN05 at ABPO.
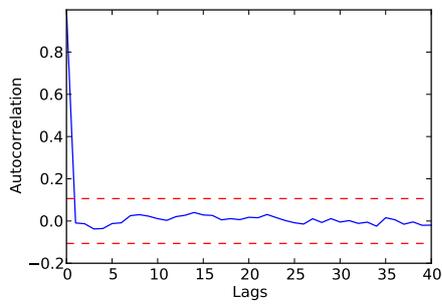
(a) L1 Code
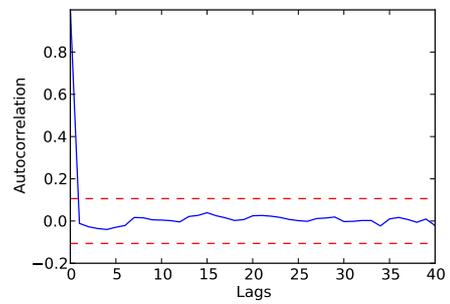
(b) L2 Code
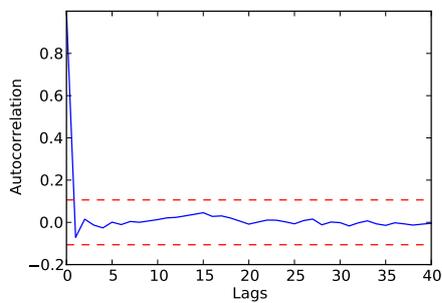
(c) L1 Carrier
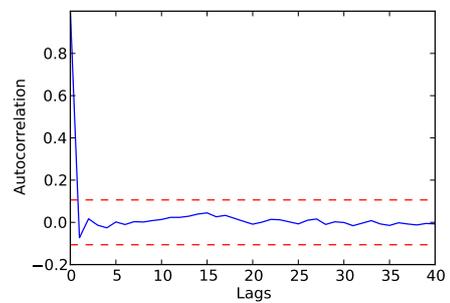
(d) L2 Carrier

Figure 6.39: Autocorrelation plot for the AR(p) whitened residuals of GPS SVN05 at FALK.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
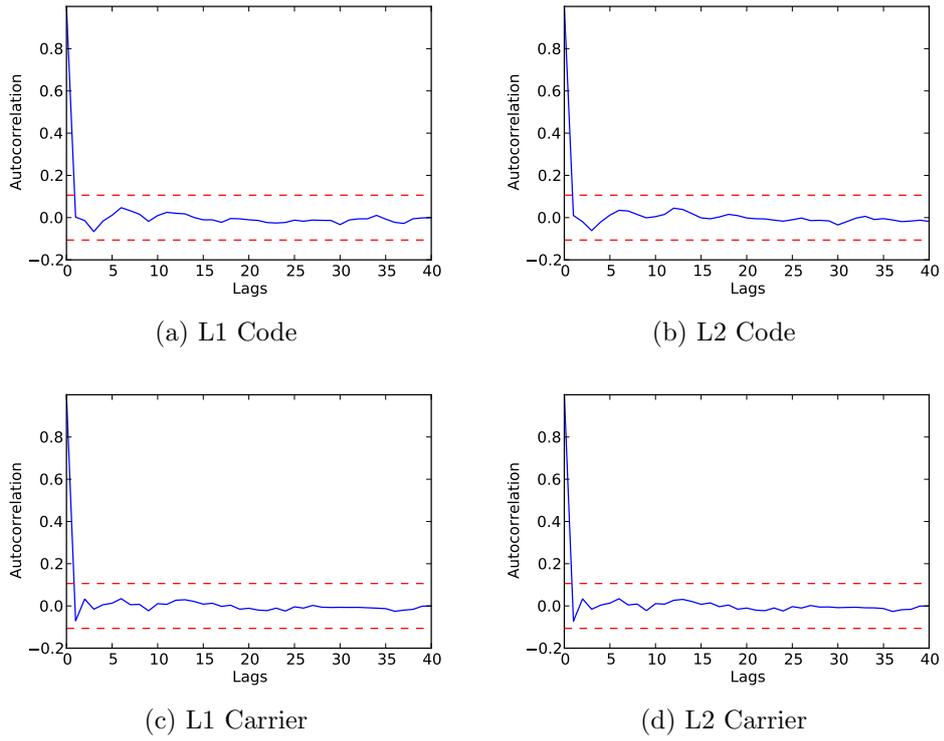
Figure 6.40: Autocorrelation plot for the AR(p) whitened residuals of GPS SVN05 at GUAM.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
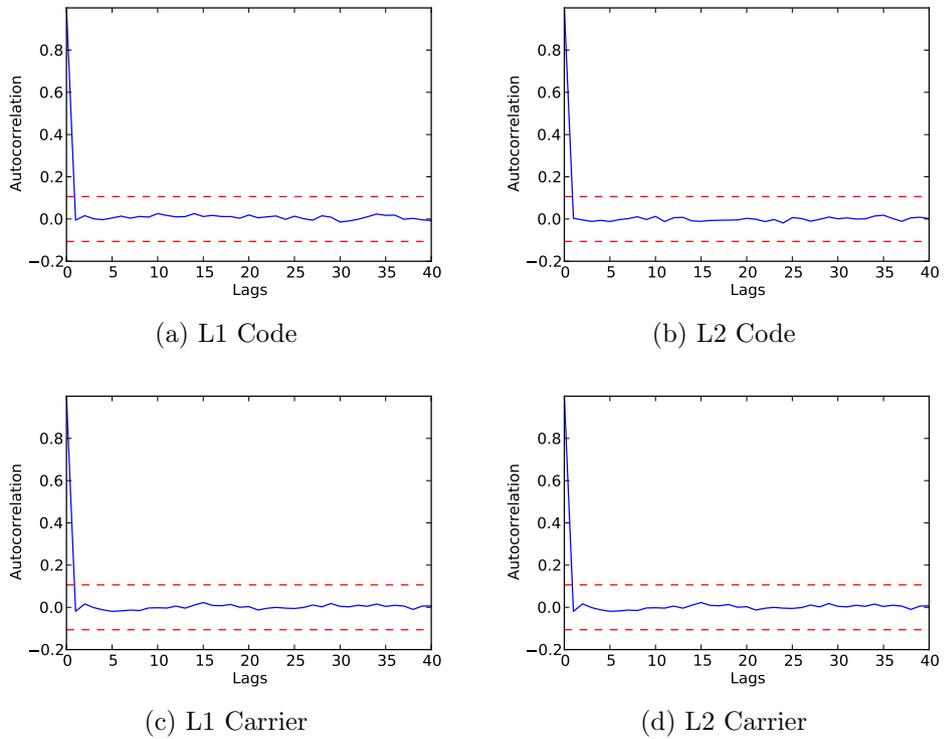
Figure 6.41: Autocorrelation plot for the AR(p) whitened residuals of GPS SVN17 at ALBM.
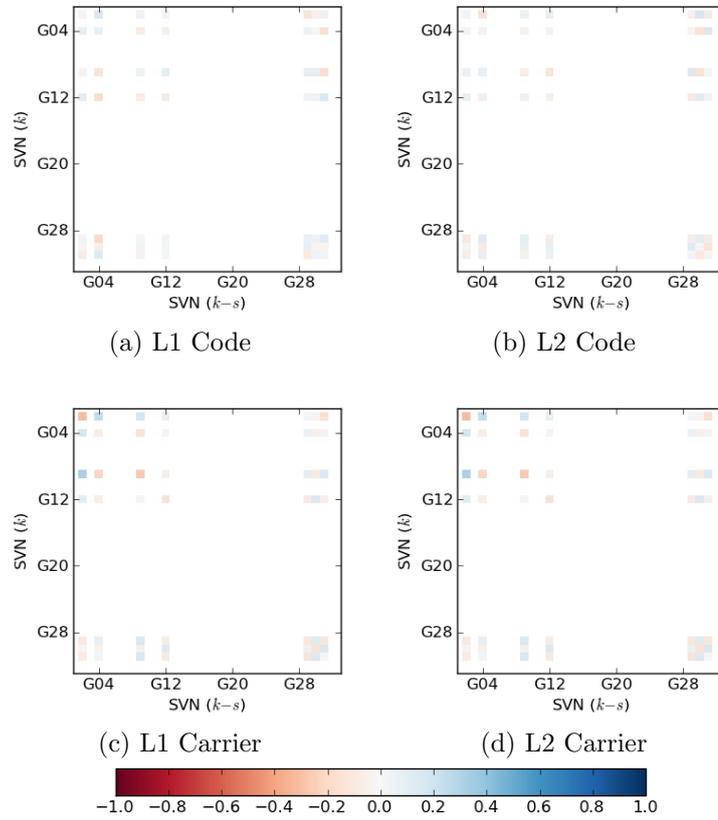
Figure 6.42: Lag 1 cross-correlation plot of AR(p) whitened residuals for ALBM dataset.

smaller than those of the unwhitened residuals. In comparison with the AR(1) cross-correlation matrices the values are of a similar magnitude, but in some cases the range and standard deviation for the AR(p) cross-correlations are smaller and in other cases it is the values in the AR(1) cross-correlations that are smaller. It must not be forgotten that values where $\rho_{ij}(s) < 0.1061$ are not statistically significant (cf. Eq. 5.32) and therefore many of the values can be considered to be indicative of random noise.

6.9 Discussion of AR(p) model order selection

From the results presented in this chapter it is clear that by not restricting the autoregression to a first-order process the resulting adjusted residuals are closer to white noise. The method chosen for dealing with higher orders, referred to as the AR(p) model, handles this by automatically choosing the model order at each epoch. An alternative method would be to fix the AR model order for each observation

(a) L1 Code   (b) L2 Code

(c) L1 Carrier   (d) L2 Carrier
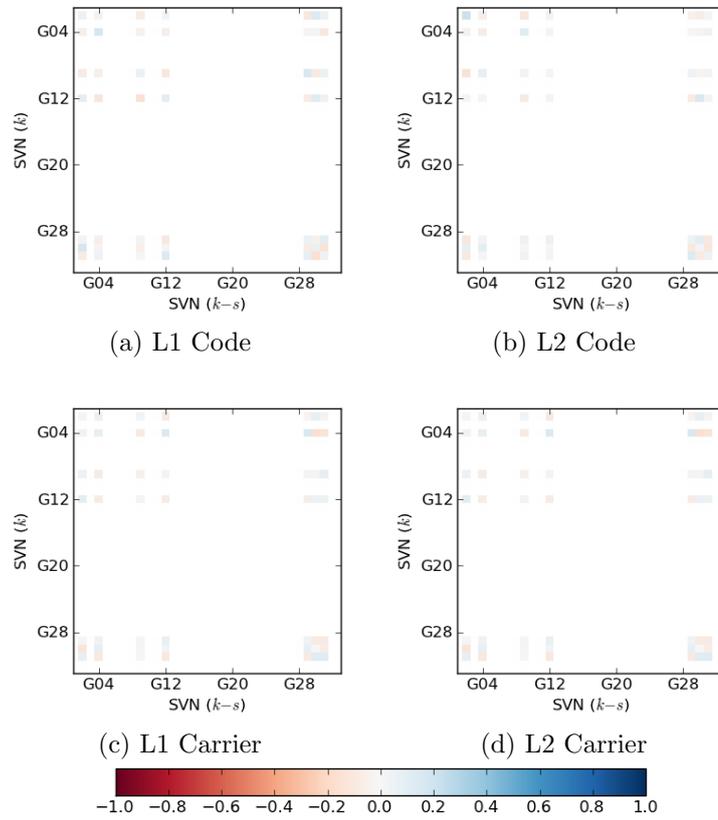
Figure 6.43: Lag 5 cross-correlation plot of AR(p) whitened residuals for ALBM dataset.

(a) L1 Code   (b) L2 Code

(c) L1 Carrier   (d) L2 Carrier
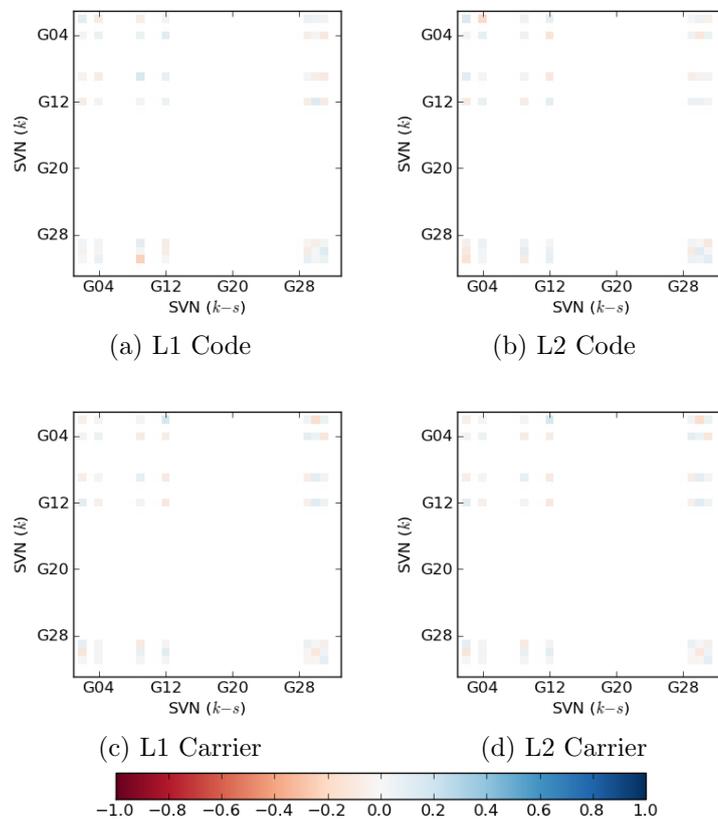
Figure 6.44: Lag 10 cross-correlation plot of AR(p) whitened residuals for ALBM dataset.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
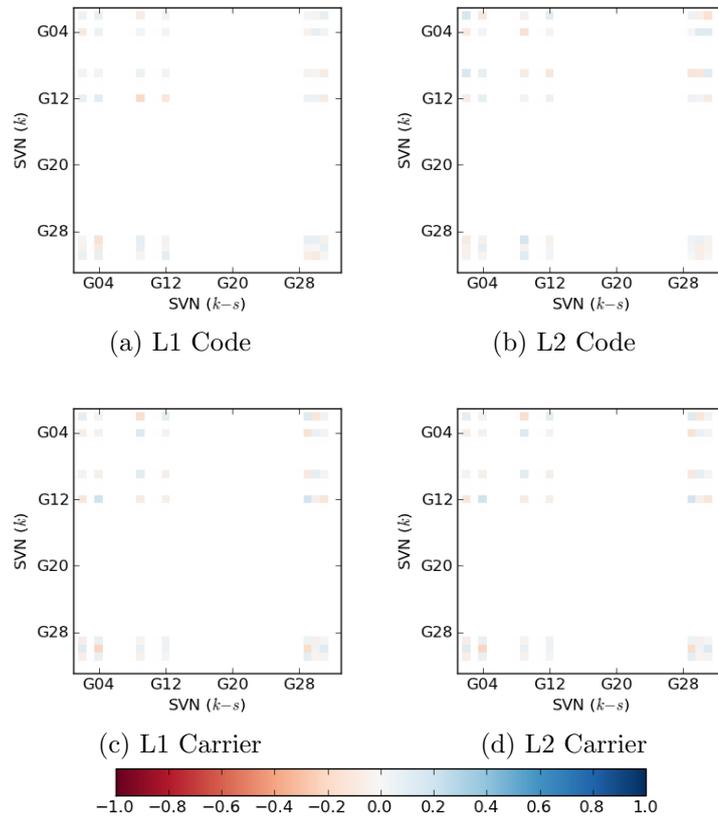
Figure 6.45: Lag 15 cross-correlation plot of AR(p) whitened residuals for ALBM dataset.

(a) L1 Code

(b) L2 Code

(c) L1 Carrier

(d) L2 Carrier
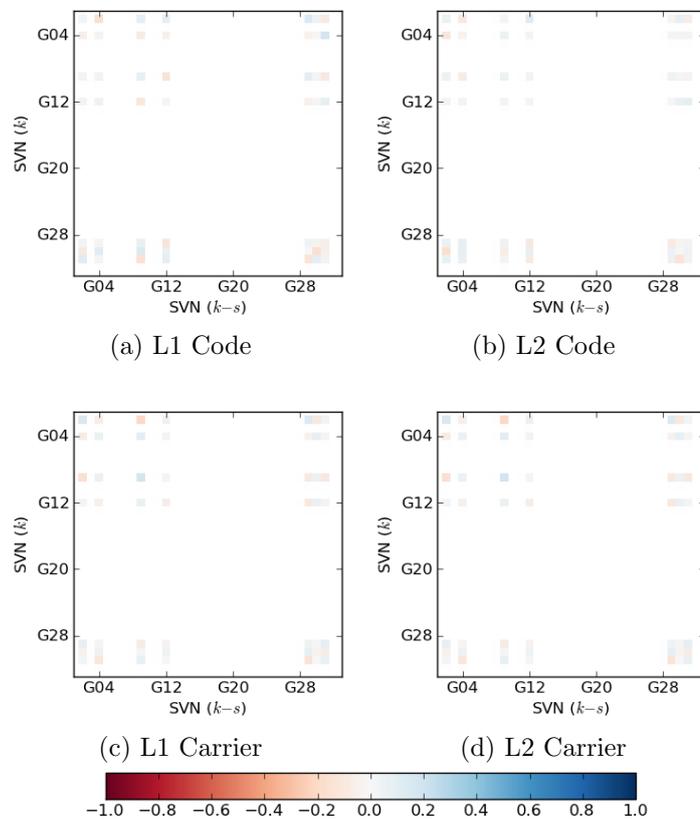
Figure 6.46: Lag 30 cross-correlation plot of AR(p) whitened residuals for ALBM dataset.

Table 6.5: Minimum, maximum, mean, standard error and standard deviation of cross-correlation values of AR(p) whitened residuals for all satellites and observation types for ALBM dataset at lags 1, 5, 10, 15 and 30.

Lag	Observation type	Min	Max	Mean	St. err.	St. dev.
1	L1 Code	-0.1835	0.1377	6.741×10^{-4}	2.464×10^{-3}	0.0764
	L2 Code	-0.1401	0.1664	2.016×10^{-3}	2.526×10^{-3}	0.0783
	L1 & L2 Carrier	-0.3068	0.3068	-3.431×10^{-4}	4.012×10^{-3}	0.1244
	All observations	-0.3068	0.3068	1.937×10^{-4}	7.291×10^{-4}	0.0904
5	L1 Code	-0.1589	0.1662	7.243×10^{-4}	2.615×10^{-3}	0.0811
	L2 Code	-0.1235	0.2046	1.081×10^{-3}	2.252×10^{-3}	0.0698
	L1 & L2 Carrier	-0.1575	0.1540	-8.438×10^{-4}	2.293×10^{-3}	0.0711
	All observations	-0.2384	0.2394	1.035×10^{-4}	5.843×10^{-4}	0.0725
10	L1 Code	-0.2283	0.1423	9.123×10^{-5}	2.171×10^{-3}	0.0673
	L2 Code	-0.1810	0.1399	2.796×10^{-3}	2.233×10^{-3}	0.0692
	L1 & L2 Carrier	-0.1711	0.1547	1.094×10^{-3}	2.111×10^{-3}	0.0654
	All observations	-0.2283	0.1908	2.488×10^{-4}	5.495×10^{-4}	0.0681
15	L1 Code	-0.1815	0.0939	-9.734×10^{-5}	1.949×10^{-3}	0.0604
	L2 Code	-0.1366	0.1645	-3.531×10^{-5}	2.395×10^{-3}	0.0742
	L1 & L2 Carrier	-0.2155	0.2015	4.181×10^{-4}	2.859×10^{-3}	0.0886
	All observations	-0.2179	0.2179	7.633×10^{-5}	6.084×10^{-4}	0.0754
30	L1 Code	-0.1594	0.1747	-5.385×10^{-4}	2.553×10^{-3}	0.0791
	L2 Code	-0.1567	0.1030	7.910×10^{-4}	1.874×10^{-3}	0.0581
	L1 & L2 Carrier	-0.1969	0.1700	-8.133×10^{-6}	2.392×10^{-3}	0.0742
	All observations	-0.1970	0.1970	1.938×10^{-5}	5.667×10^{-4}	0.0703

type, for example one value for the code and another for the carrier phase. If it assumed that the source of the time correlation is a result of hardware (e.g. tracking loop errors), site environment (e.g. multipath) and the application of corrections (e.g. orbit and clock corrections), the AR model that would be specified a priori would not only need to be determined on a site-by-site basis, but also may change in time as the satellite constellation changes. For kinematic positioning the site environment may also be continually changing.

It is also evident from 6.26 – 6.29 that the AR model order itself can vary within very wide limits from epoch to epoch. It is reasonable to assume that the AR model order would not change by such varying degrees. One possible method of overcoming this is to adapt the model order selection by placing constraints based on the previously calculated model orders. Since this is a complex procedure it has not been employed in this study, but could be considered in future work.

Another aspect of the implementation described in §6.8 is that the AR parameters are allowed to change unconstrained from one epoch to the next. The reason for this is purely from an implementation standpoint, since this was more straightfor-

ward to implement in the GNSS processing software. Even though it has previously been stated that it is assumed that the AR parameters change over time, given the probable sources of the error specified above it is safe to assume that these values will vary slowly over time. It is therefore possible to use a Kalman-like filter to estimate the AR parameters. If the model order is allowed to change from one epoch to the next then this poses obvious implementation issues, since it is not possible to propagate an estimated parameter if the AR model order is changed. One solution to this would be to implement parallel filters, although this will impact on computational efficiency.

6.10 Estimating AR parameters as a multivariate process

The estimation of the AR parameters has thus far been done separately for each satellite and observation type pair. As seen in §5.3.7 the post-fit residuals are not only autocorrelated, but cross-correlations between satellites and observation types also exist. Therefore these should theoretically be included in the AR model.

The n-variate $\mathrm{AR}(p)$ model can be expressed as

$$x_k = \Phi_1 x_{k-1} + \cdots + \Phi_p x_{k-p} = \varepsilon_k \tag{6.118}$$

where

$$x_k = \begin{pmatrix} x_{1,k} \\ \vdots \\ x_{n,k} \end{pmatrix}, \qquad \Phi_p = \begin{pmatrix} \phi_{11,p} & \cdots & \phi_{1n,p} \\ \vdots & \ddots & \vdots \\ \phi_{n1,p} & \cdots & \phi_{nn,p} \end{pmatrix}, \qquad \varepsilon_k = \begin{pmatrix} \varepsilon_{1,k} \\ \vdots \\ \varepsilon_{n,k} \end{pmatrix}$$

From Eq. (6.118) it can be seen that the AR coefficients are now replaced by p matrices with dimensions $n \times n$, hence including the cross-correlations between different time series.

If the multivariate process is assumed to be stationary it is possible to derive an

analogous set of Yule-Walker equations (see §6.4.1). This is achieved by replacing $\{R(0), \ldots, R(p)\}$ in Eq. (6.32) with the cross-covariances of the multivariate processes, given by Eq. (5.35). Taking the AR(1) process as an example, its solution is given by

$$R(0) \ \Phi_1 = R(1) \tag{6.119}$$

However for higher order processes the model takes on a more complex structure, as shown by Priestley (1981). If the process has values close to the unit circle (see §6.2) then this will also cause the same problems for stability as outlined in §6.4.

Since the cross-covariance matrices have been created for analysis in §5.3.7 it was possible to extend what had already been done to include multivariate estimation. It has already been noted in §6.7 that the AR coefficients are very close to the unit circle and the least squares methods, of which solving the Yule-Walker equations is analogous, can lead to unstable estimates of the AR coefficients. In order to solve Eq. (6.119) it is necessary to invert R_0. This is the cross-covariance matrix at lag 0, but also $E\{\underline{\hat{e}}_k, \underline{\hat{e}}_k^T\}$ and hence equivalent to $Q_{\hat{e}_k}$. Since the latter cannot be inverted when using a least squares derived method, since it has a rank equal to the degrees of freedom (Krakiwsky et al. 1999), $R(0)$ for the post-fit residuals is also non-invertible. Experimentally this was confirmed with the estimate of $R(0)$ from the ALBM dataset. For this reason the estimation of AR coefficients using post-fit residuals cannot be treated as a multivariate problem, as the semi-recursive Kalman filter used in the E-HP/PPP processing software cannot calculate predicted residuals.

6.11   Summary

In this chapter stationarity was introduced in §6.2 along with the concept of the unit circle. Following discussion it was noted that even if a whole series is not itself stationary over small time periods it can be.

Following this two equivalent models for an AR process were presented in §6.3, namely the *Box-Jenkins model* and the *Priestley model*. The latter was chosen for

use in this study as it lends itself to the concepts developed in later chapters. The properties of an AR process were also described.

Four methods for estimating an AR process were then described in §6.4. The Yule-Walker method was presented as a group of equations that describe the AR process in terms of autocovariances or autocorrelations. Levinson-Durbin recursion was then introduced as a method for obtaining a more numerically stable solution, but not considered suitable for automated processes. The least squares methods were presented as an alternative method of estimating the AR process, althogh this was also unsuitable for automated processes. Finally, Burg's method presented the best option for application in this study as it leverages the numerical efficiency of Levinson-Durbin recursion with a comboination of forward and backward residuals used in the least squares methods to ensure stationarity.

Since AR model orders greater than 1 were to be considered in the study an overview of methods for determining model order were considered in §6.5. Aikake's Information Criterion was chosen as the desired method. This has the advantage that it can easily be integrated into Burg's method.

The theory of optimal and non-optimal filters presented in §5.2 was extended in §6.6 to show that when a filter is non-optimal the predicted residuals can be shown to be characterised by an AR(1) process. Through the connection between the predicted and post-fit residuals derived in §3.7 this was shown also to be the case for the post-fit residuals.

In §6.7 only an AR(1) process was considered as this is relatively straightforward to implement and minimises the numerical expense of the process, which is an important consideration for real-time applications. It was found that such a basic model was sufficient for removing a large proportion of the correlation, although in some cases, where the best model order would ideally be higher, the residual was not sufficiently whitened. This was particularly noticeable in the Durbin-Watson values. The cross-correlation plots showed that although modelled as a univariate AR process, the cross-correlations had also been reduced and more importantly no cross-correlations had been introduced.

The whitening procedure was extended in §6.8 so that the residuals were modelled as an AR(p) process, where AIC was the chosen criterion for choosing model order. Whilst this yielded similar results to those seen for the AR(1) process, it did reduce the number of instances where the number of residuals were 'overwhitened'. This was once again reflected in the Durbin-Watson values.

Issues regarding the AR(p) model order selection were discussed in §6.9. Although it is possible to predetermine the AR order for given observation types since this is receiver and site dependent the conclusion is that this is not always practical. Additionally if the AR parameters were to propogated from one epoch to the next using a recursive estimator changing model order between epochs would be complex.

As the residuals were modelled as a univariate process, i.e. the coefficients for each satellite and observation type pair were modelled separately, it is important to consider a multivariate AR process and this was discussed in §6.10. Given that the autocovariance matrix at lag 0 is not invertible for the post-fit residuals from a least squares method this was not an option with the software that was available. It is also important to note that estimating the AR parameters as a multivariate problem is much more complex, especially when different satellite and observation type pairs may simultaneously be characterised by different model orders.

In conclusion, a reliable method was established for whitening the post-fit residuals from the PPP software with the AR model being automatically determined during processing. These were whitened post-fit residuals shown to be typical of random noise. An AR(p) process was best suited to whitening the residuals, although in many cases an approximation as an AR(1) process was sufficient. The use of these residuals in a modified test statistic is presented in Chapter 7.

Chapter 7

The whitened test statistic

7.1   Introduction

It has been shown in Chapter 6 that it is possible to model the time correlation in the residual time series as an autoregressive process. A number of autoregressive methods were presented and Burg's method (Burg 1967) was chosen as the most suitable. It was then shown in §6.7 and §6.8 that by estimating the AR coefficients the post-fit residuals from our PPP software could be whitened resulting in residuals that closely resemble white noise.

Since the quality control procedures outlined in Chapter 4 are based upon the assumption that the residuals are not correlated in time, the T test statistic given in §4.2 will be modified for use with the whitened residuals. The quality control procedures are briefly revisited in §7.2, which identify the variance of the whitened residual as a requirement for forming a whitened test statistic. The method for calculating this is derived in §7.3 and estimated for the four datasets used previously. The formulae for the whitened residual and its variance are brought together in §7.4 to give the whitened test statistic for the predicted and post-fit residuals. Since there are a number of constraints on using the whitened test statistic, a number of implementation issues are discussed in §7.5.

The whitened test statistic is firstly analysed as a time series in §7.6, where it is compared alongside the unwhitened test statistic. For specific satellite and observation types the behaviour of the whitened test statistic is then analysed in §7.7

on an epoch by epoch basis, comparing values for different satellite and observation types. A comparison of different scenarios is made.

The subject of reliability covered in §4.7 is revisited in §7.8 and the implications of using the variance of the whitened residual to calculate the MDB are discussed. Sample time series of MDB values for the four datasets are also presented. Finally, the improved power of the whitened test statistic and whitened MDB are discussed in §7.9, including the implications of the 240 second time period required for estimating the AR parameters.

7.2 Quality control procedures revisited

In Chapter 4 the quality control procedures for a recursive estimator were derived using the predicted residual, $\underline{v}_k$. It was subsequently shown that it is also possible to use the post-fit residual, $\underline{e}_k$, given the linear relationship described in §3.7. Since it has been shown in Chapter 6 that the time correlated residual can be whitened using autoregression, it should be possible to form a test statistic using this value. In this section the quality control procedures are revisited in order to determine what values are required for a whitened test statistic.

7.2.1 Predicted residual

For the predicted residuals the test statistic is given by Eq. (4.17), repeated again here for clarity,

$$\underline{T} = \underline{v}^T Q_v^{-1} C (C^T Q_v^{-1} C)^{-1} C^T Q_v^{-1} \underline{v}$$

From this it can be seen that $\underline{v}$ and Q_v are the two calculable quantities that are required in order to determine the test statistic. It has already been established that $\underline{v}$ would be replaced by the whitened residual, $\underline{\tilde{v}}$, using the method outlined in Chapter 6. The variance-covariance of the predicted residual, Q_v, must now be replaced by $Q_{\tilde{v}}$, since the former is calculated on the assumption of an optimum filter using Q_{y_k} and Q_{w_k}, which are now replaced with $Q_{y_k}^*$ and $Q_{w_k}^*$.

7.2.2 Post-fit residual

In the case of the post-fit residuals the test statistic is given by Eq. (4.5), again repeated here for clarity.

$$\underline{T} = \hat{\underline{e}}_k^T Q_{y_k}^{-1} C_{y_k} (C_{y_k}^T Q_{y_k}^{-1} Q_{\hat{e}_k} Q_{y_k}^{-1} C_{y_k})^{-1} C_{y_k}^T Q_{y_k}^{-1} \hat{\underline{e}}_k$$

There are three main constituents of this test statistic that are calculable, namely

(i) $\hat{\underline{e}}_k$, the post-fit residuals,

(ii) Q_{y_k}, the variance-covariance matrix of the observables, and

(iii) $Q_{\hat{e}_k}$, the variance-covariance matrix of the post-fit residuals.

As for the predicted residuals, $\hat{\underline{e}}$ and $Q_{\hat{e}}$ can be replaced with their whitened equivalents. The inclusion of Q_{y_k} in this model poses a problem, as it is coupled with $Q_{\hat{e}_k}$. Recalling Eq. (3.42),

$$Q_{\hat{e}_k} = Q_{y_k} + A_k Q_{\hat{x}_{k|k}} A_k^T$$

it can be seen that Q_{y_k} appears implicitly in the test statistic as part of $Q_{\hat{e}_k}$ as well as explicitly in the test statistic itself. It will be shown in §7.3 that the variance of the whitened residual is found by scaling $Q_{\hat{e}_k}$. In the standard case a change in $Q_{\hat{e}_k}$ will be coupled with a change in Q_{y_k}, but in the whitened case the implicit instance of Q_{y_k} will be scaled whereas the explicit one will not. The solution to this discrepancy is to make an adjustment to the explicit instance of Q_{y_k}, which shall be referred to as $\widetilde{Q}_{y_k}$, which will be defined in §7.4. This is not a problem for the case using predicted residuals, as Q_{y_k} only appears implicitly in Q_{v_k}.

7.3 Variance of the whitened residual

There are two approaches that can be taken in order to determine the variance of the whitened residual. In Chapter 6 the whitened residual was derived as ε_k, the white noise process that results from an AR estimation. It can be shown using

autoregression theory that the variance of this white noise process, $\sigma_{\varepsilon_k}^2$, is calculable as a function of the original estimate of the AR process.

A second approach is to take the theory already developed that shows the predicted and post-fit residuals can be modelled as an AR process and then extend this to derive the expression for the variance-covariance matrix of the whitened residual.

Whilst all the analysis done has been based on a univariate process, it is theoretically possible to model the entire problem as a multivariate AR process. Although this is not done in practice, due to the problems described in §6.10, it is important to address the theoretical consequences of this.

7.3.1 Autoregression theory

Using the approach taken by Box and Jenkins (1976) the AR(p) process

$$x_k = \phi_1 x_{k-1} + \cdots + \phi_p x_{k-p} + \varepsilon_k$$

is multiplied by x_{k-s}, for $s \geq 0$, giving

$$x_{k-s} x_k = \phi_1 x_{k-s} x_{k-1} + \cdots + \phi_p x_{k-s} x_{k-p} + x_{k-s} \underline{\varepsilon}_k \tag{7.1}$$

Taking the expectations of Eq. (7.1) gives

$$R(s) = \phi_1 R(1) + \cdots + \phi_p R(p) \tag{7.2}$$

where $s > 0$. It can be seen that the term that includes $\underline{\varepsilon}_k$ vanishes from the equation, as $E\{x_{k-s}\varepsilon_k\} = 0$ where $k > 0$. However, taking $k = 0$ the equation now becomes

$$R(0) = \phi_1 R(1) + \phi_2 R(2) + \cdots + \phi_p R(p) + \sigma_\varepsilon^2 \tag{7.3}$$

This is because for $s = 0$ the term $E\{x_{k-s}, \varepsilon_k\} = E\{\varepsilon_k, \varepsilon_k\} = \sigma_\varepsilon^2$. Since $R(s) = R(0)\rho(s)$ and $R(0) = \sigma_x^2$,

$$\sigma_x^2 = \frac{\sigma_\varepsilon^2}{1 - \phi_1 \rho(1) - \phi_2 \rho(2) - \cdots - \phi_p \rho(p)} \tag{7.4}$$

which can be rearranged and simplified to give

$$\sigma_\varepsilon^2 = \sigma_x^2 \left(1 - \sum_{i=1}^{p} \phi_i \rho(i)\right) \tag{7.5}$$

The approach taken by Broersen (2006) is in terms of the reflection coefficient, defined in §6.4.4, and forms part of Burg's method. The variance of the AR process is given as

$$\sigma_x^2 = \frac{\sigma_\varepsilon^2}{\prod\limits_{i=1}^{p}(1 - \kappa_i^2)} \tag{7.6}$$

where κ is the reflection coefficient. Recall from §6.4.4 that this of opposite sign to the partial autocorrelation. It can be seen that for an AR(1) process this resembles the formula given by Box and Jenkins (1976).

Priestley (1981) gives an estimator for σ_ε^2, which is said to be an approximately unbiased estimate.

$$\hat{\sigma}_\varepsilon^2 = \frac{N - p}{N - 2p - 1} \left[R(0) - \phi_1 R(1) - \ldots - \phi_p R(p)\right] \tag{7.7}$$

The first notable difference between the Box and Jenkins (1976) method, given by Eq. (7.5), and that by Priestley (1981), given by Eq. (7.7), is the multiplying factor that is used to inflate the estimate depending on the sample size N and AR order p. However, it can be seen that for low-order AR processes that use a large sample size N the factor is almost unity. The second difference is that this equation uses the autocovariance $(R(s))$ and not the autocorrelation $(\rho(s))$. However, by recalling that dividing $R(s)$ by $R(0)$ gives $\rho(s)$ and that $R(0) \equiv \sigma_x^2$

$$R(0) - \phi_1 R(0) - \cdots - \phi_p R(p) = \sigma_x^2 (1 - \phi_1 \rho(1) - \phi_2 \rho(2) - \cdots - \phi_p \rho(p)) \tag{7.8}$$

In all of the above cases the variance of the residual, σ_ε^2, is expressed in terms of the variance of the time series, σ_x^2. When whitening the predicted or post-fit residual from the filter the stochastic process $x(k)$ is replaced by $\underline{v}_k$ or $\hat{e}_k$ respectively. It was shown in §6.7 that ε_k of Eq. (6.116) is the whitened residual and it therefore follows that σ_ε^2 is equivalent to the variance of the whitened predicted residual, $\sigma_{\tilde{v}_k}^2$,

or the whitened post-fit residual, $\sigma_{\tilde{e}_k}^2$, depending on which residual is being used.

7.3.2 Non-optimal Kalman filter theory

By taking the expectation of $\tilde{e}_k$ it is possible to find an expression for $E\{\tilde{e}_k, \tilde{e}_k^T\}$. Similarly, taking the expectation of $\tilde{v}_k$ yields $E\{\tilde{v}_k, \tilde{v}_k^T\}$. These are equivalent to $Q_{\tilde{e}_k}$ and $Q_{\tilde{v}_k}$ respectively. The derivation below is for the post-fit residual, but the derivation is identical for the predicted residual and it is possible to directly replace references to $\tilde{e}_k$ with $\tilde{v}_k$.

The vector of whitened post-fit residuals for an AR(1) process reads

$$\underline{\tilde{e}}_k = \underline{\hat{e}}_k - \Phi_1 \underline{\hat{e}}_{k-1} \tag{7.9}$$

where Φ_1 is an $m \times m$ matrix containing the AR(1) coefficients. If the AR coefficients are estimated using a univariate model then this matrix will be diagonal.

Taking expectations,

$$E\{\underline{\tilde{e}}_k, \underline{\tilde{e}}_k^T\} = E\{\underline{\hat{e}}_k - \Phi_1 \underline{\hat{e}}_{k-1}, \underline{\hat{e}}_k^T - \underline{\hat{e}}_{k-1}^T \Phi_1^T\}$$
$$= E\{\underline{\hat{e}}_k, \underline{\hat{e}}_k^T\} - \Phi_1 E\{\underline{\hat{e}}_{k-1}, \underline{\hat{e}}_k^T\} - E\{\underline{\hat{e}}_k^T, \underline{\hat{e}}_{k-1}^T\} \Phi_1^T + \Phi_1 E\{\underline{\hat{e}}_{k-1}, \underline{\hat{e}}_{k-1}^T\} \Phi_1^T$$
$$\tag{7.10}$$

The expectations can be replaced by the covariance matrices

$$R(0) = E\{\underline{\hat{e}}_k, \underline{\hat{e}}_k^T\} \qquad\qquad R(1) = E\{\underline{\hat{e}}_k, \underline{\hat{e}}_{k-1}^T\} \tag{7.11}$$

Using this Eq. (7.10) can be expressed as

$$E\{\underline{\tilde{e}}_k, \underline{\tilde{e}}_k^T\} = R(0) - \Phi_1 R(1)^T - R(1)\Phi_1^T + \Phi_1 R(0)\Phi_1^T \tag{7.12}$$

Recalling the Yule-Walker equations, given by Eq. (6.30), the relationship between the covariances matrices, $R(0)$ and $R(1)$, and Φ_1 can be given by

$$R(0)\ \Phi_1 = R(1) \tag{7.13}$$

and substituting into Eq. (7.12) gives

$$E\{\tilde{e}_k, \tilde{e}_k^T\} = R(0) - \Phi_1 \Phi_1^T R(0) - R(0)\Phi_1 \Phi_1^T + \Phi_1 R(0)\Phi_1^T \qquad (7.14)$$

So far the solution is for a multivariate approach. By considering the ith post-fit residual Eq. (7.14) reduces to

$$\begin{aligned} E\{\tilde{e}_{k,i}, \tilde{e}_{k,i}\} &= R_{ii}(0) - \phi_1^2 R_{ii}(0) - R_{ii}(0)\phi_1^2 + \phi_1 R_{ii}(0)\phi_1 \\ &= R_{ii}(0)\left[1 - \phi_1^2\right] \end{aligned} \qquad (7.15)$$

It can be seen that Eq. (7.15) is in agreement with the approaches by Eq. (7.5), (7.6) and (7.7), when an AR(1) process is considered.

7.3.3 Discussion of the non-optimality of the unwhitened residual variance

In §7.3.1 and §7.3.2 the whitened variance of the predicted residual, $Q_{\tilde{v}_k}$, and the whitened variance of the post-fit residual, $Q_{\tilde{e}_k}$, are calculated by scaling $R(0)$. Box and Jenkins (1976) and Priestley (1981) express this scaling factor as a function of the AR coefficients and the autocorrelation functions, whereas the method by Broersen (2006) uses the reflection coefficients. Since the latter are calculated as part of Burg's method, this is the method chosen for implementation.

It is known that $R(0) = E\{\underline{v}_k, \underline{v}_k^T\}$ (predicted residuals) and $R(0) = E\{\underline{\hat{e}}_k, \underline{\hat{e}}_k^T\}$ (post-fit residuals), with both estimated using a sample of the residuals from the previous 240 seconds when estimating the AR coefficients. Since this is the average covariance over 240 seconds it is impractical to use this in a test statistic, since it is not indicative of the variance at the epoch in question.

From §4.4 it is known that $Q_{v_k} = E\{\underline{v}_k, \underline{v}_k^T\}$ and from §4.3 that $Q_{\hat{e}_k} = E\{\underline{\hat{e}}_k, \underline{\hat{e}}_k^T\}$, hence $Q_{v_k} \equiv R(0)$ (predicted residuals) and $Q_{\hat{e}_k} \equiv R(0)$ (post-fit residuals). The variance of the predicted residuals are computed using Eq. (3.29),

$$Q_{v_k} = Q_{y_k} + A_k Q_{x_{k|k-1}} A_k^T$$

and the variance of the post-fit residuals using Eq. (3.42),

$$Q_{\hat{e}_k} = Q_{y_k} + A_k Q_{x_{k|k}} A_k^T$$

The main disadvantage to using this estimate of $E\{v_k, v_k^T\}$ and $E\{\hat{e}_k, \hat{e}_k^T\}$ is that in an optimal filter they are computed using Q_{y_k} and Q_{w_k}, but in a non-optimal filter using $Q_{y_k}^*$ and $Q_{w_k}^*$. The result is that $Q_{v_k} \neq E\{v_k^*, v_k^{*T}\}$ (using Eq. 3.29) and $Q_{e_k} \neq E\{e_k^*, e_k^{*T}\}$ (using Eq. 3.42) and are likely to be underestimates. However, for the purposes of this study this difference will be assumed to be negligible enough so that Q_{v_k} and $Q_{\hat{e}_k}$ computed using the non-optimal filter are sufficient.

7.3.4 Estimates of whitened post-fit residual variance

The variances of the AR(1) and AR(p) whitened post-fit residuals were calculated using the method of Broersen (2006), using Eq. (7.6), for ABPO, FALK, GUAM and ALBM and are shown in Figures 7.1 - 7.4. The AR model order was determined using the model order that corresponded to the minimum AIC value (see §6.5.4 and §6.8.1). Since $|\kappa| < 1$, and hence the partial autocorrelation also less than 1, the scaling factor $\prod_{i=1}^{p}(1 - \kappa_i^2)$ will always be less than one. The variance of the whitened residual will therefore always be smaller than that of the unwhitened residual. This is evident in Figures 7.1 - 7.4. Where the magnitude of time correlation is small the variance of the unwhitened and whitened residual is expected to be very close, since $\prod_{i=1}^{p}(1 - \kappa_i^2) \to 1$ as $|\kappa_i| \to 0$. Given the ratio between the L1 and L2 carrier phase residuals (see §5.3.2), only the whitened L1 carrier phase residuals are shown in Figures 7.2 - 7.4.

Since the variance of the unwhitened residual does not include any of the observables, but is calculated using only a priori values, its variation is a function of the a priori variance of the observables, the process noise added to the parameters in the Kalman filter state vector (or the common parameters in the case of the semi-recursive Kalman filter, see §3.6), and the design matrix. This therefore leads to a smooth variation in the MDB, with jumps in the case where a satellite enters or leaves view. This latter effect is due to a change in the redundancy and geometry.

(a) L1 Code - unwhitened

(b) L1 Code - whitened

(c) L2 Code - unwhitened

(d) L2 Code - whitened

(e) L1 Carrier - unwhitened

(f) L1 Carrier - whitened
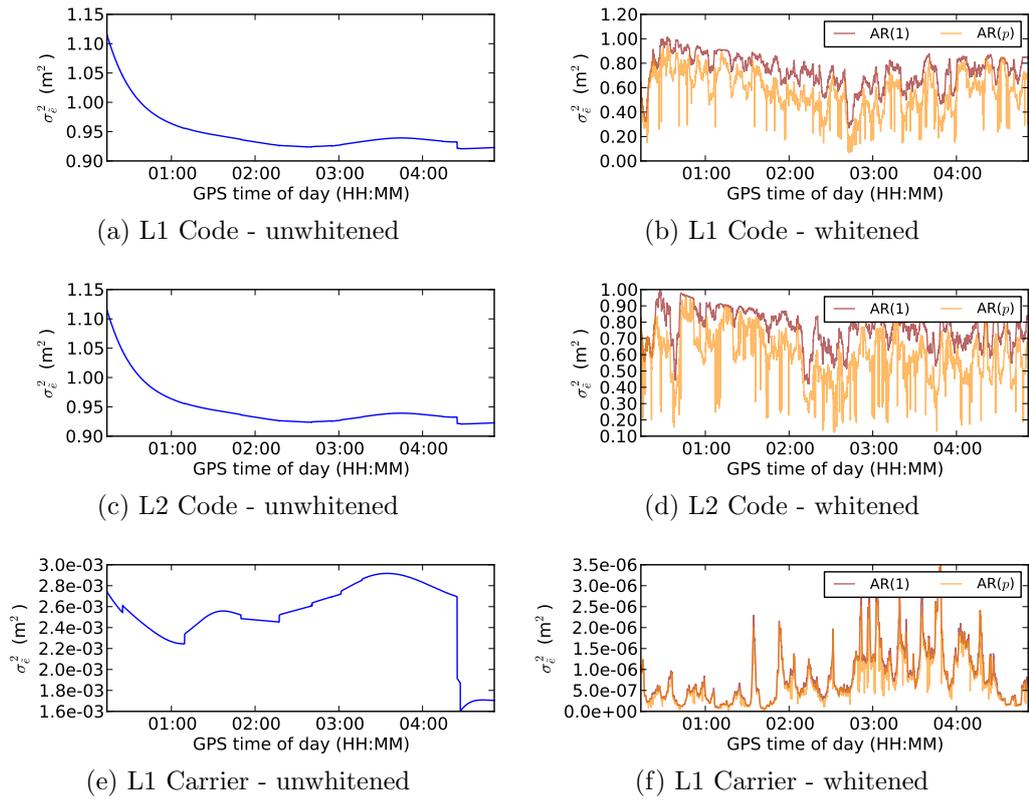
Figure 7.1: Variances of the whitened post-fit residuals for GPS SVN05 ABPO dataset.

(a) L1 Code - unwhitened

(b) L1 Code - whitened

(c) L2 Code - unwhitened

(d) L2 Code - whitened

(e) L1 Carrier - unwhitened

(f) L1 Carrier - whitened
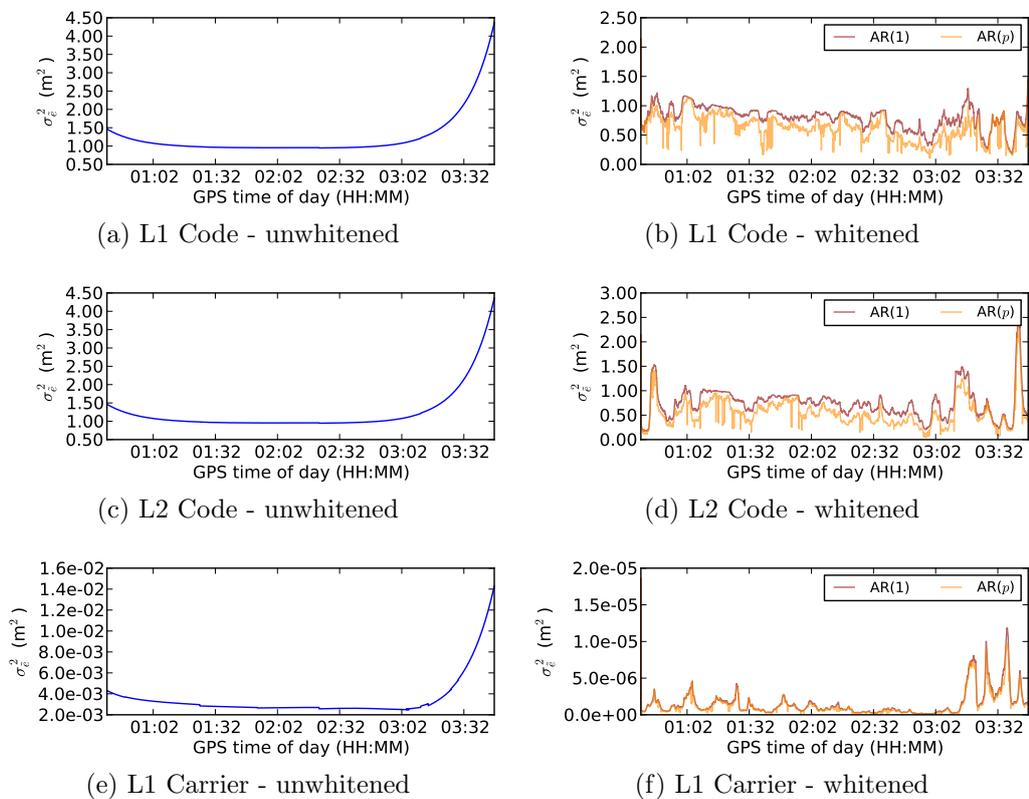
Figure 7.2: Variances of the whitened post-fit residuals for GPS SVN05 FALK dataset.

(a) L1 Code - unwhitened

(b) L1 Code - whitened

(c) L2 Code - unwhitened

(d) L2 Code - whitened

(e) L1 Carrier - unwhitened

(f) L1 Carrier - whitened
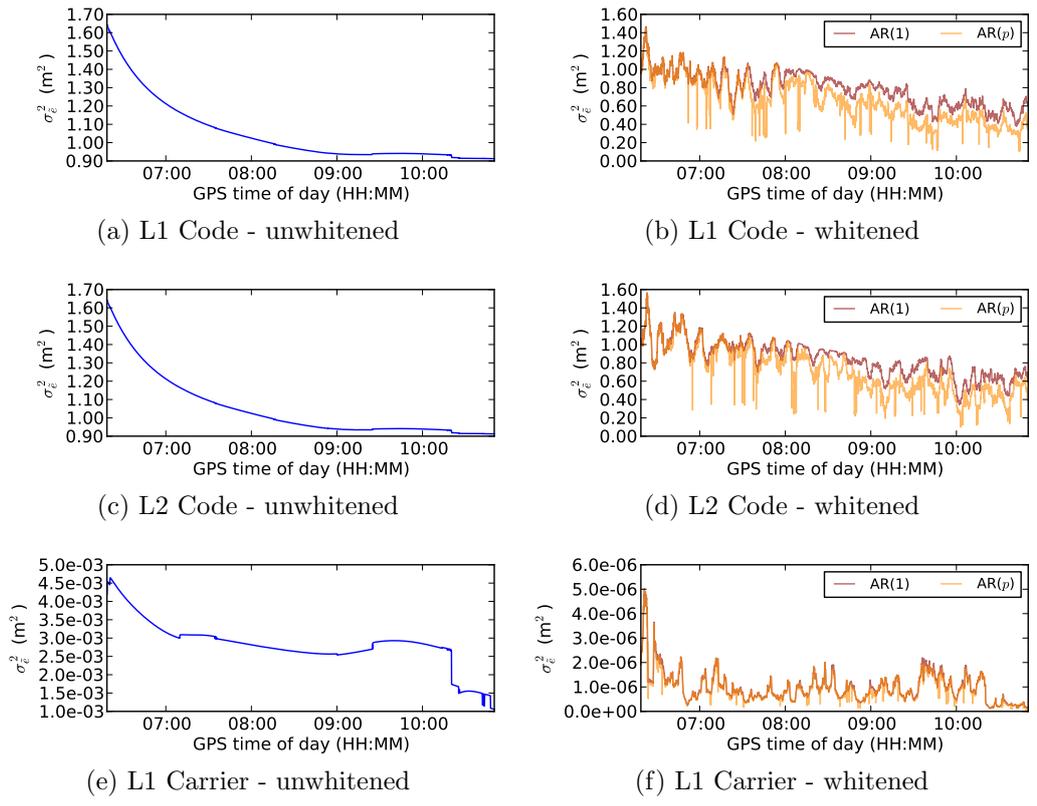
Figure 7.3: Variances of the whitened post-fit residuals for GPS SVN05 GUAM dataset.

(a) L1 Code - unwhitened

(b) L1 Code - whitened

(c) L2 Code - unwhitened

(d) L2 Code - whitened

(e) L1 Carrier - unwhitened

(f) L1 Carrier - whitened
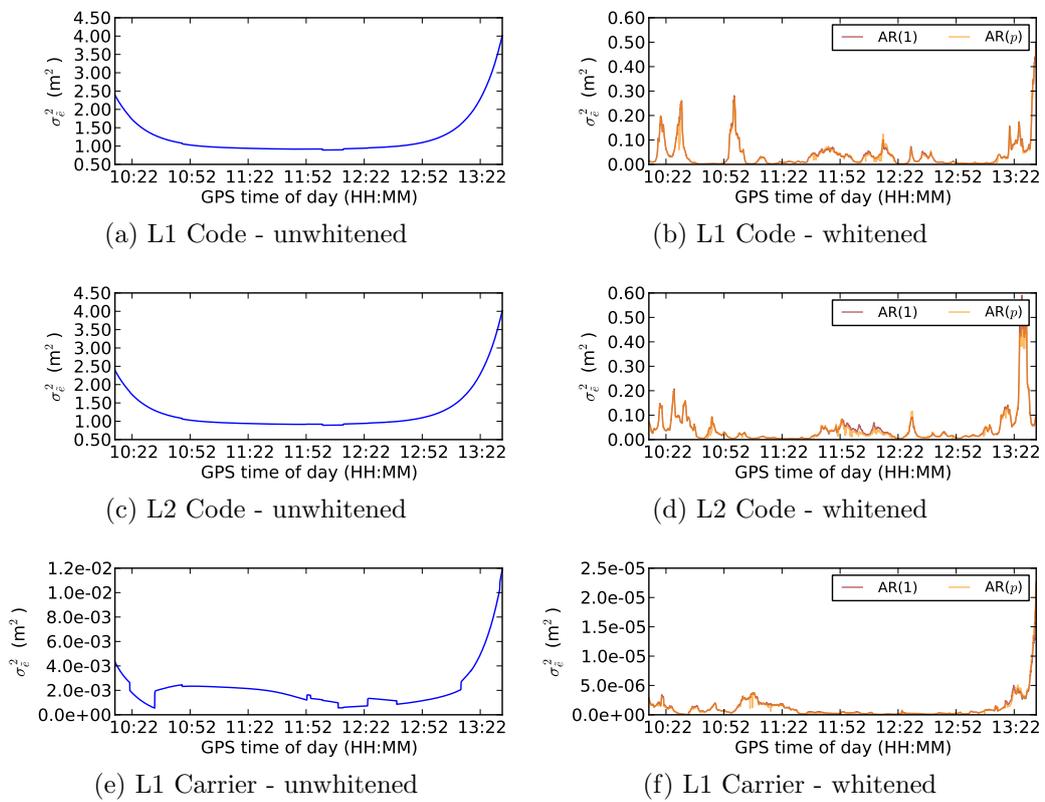
Figure 7.4: Variances of the whitened post-fit residuals for GPS SVN17 ALBM dataset.

For all datasets it is clear that the variance of the whitened residual is very closely influenced by the estimated reflection coefficients, which themselves are a function of the AR coefficients. When this is shown as a time series the variances are much less smooth than those for the unwhitened residual.

The variances for the code residuals when estimated using an AR(1) process are, in general, higher than those estimated using an AR(p) process for the ABPO, FALK and GUAM datasets. For the GUAM dataset the variances of the whitened code residuals are close until around 07:30 and then begin to diverge, indicating that the higher model orders for the AR(p) whitened residuals are being chosen after this point. In the case of the ALBM dataset the variances of the whitened code residuals estimated as an AR(1) and AR(p) process do not differ greatly. It is also noteworthy that the variances of the whitened code residuals are much smaller in magnitude than those for ABPO, FALK and GUAM. This is a result of estimates of the AR coefficients that are closer to 1, therefore causing the estimated variance of the unwhitened residual to be scaled to a smaller value.

By contrast, the variances of the unwhitened carrier phase residuals are not as smooth as those seen for the unwhitened code and carrier phase residuals, which is attributable to the higher weight placed on them in the a priori variance-covariance matrix of observables. As seen with the variances of the whitened code phase residuals, the estimated AR coefficients have a great influence on the estimated variances of the whitened carrier phase residuals. The highly variable pattern is once again present and the distinct changes in variance seen in the variance of the unwhitened residual, in Figures 7.4e and 7.4f for example, are no longer visible.

Making a comparison between the AR(1) and AR(p) estimates, with reference also to Figures 6.26 – 6.29 showing the AR orders that were selected, a number of spikes are visible in the latter that are directly attributable to the AR order selection. The variance of the AR(p) whitened post-fit residual for GPS SVN05 from the FALK dataset is shown in Figure 7.5 along with the corresponding AR model order chosen. It is clear that the spikes are a result of a much higher AR order being selected. This has caused a slight jump in the scaling factor in Eq. (7.6),
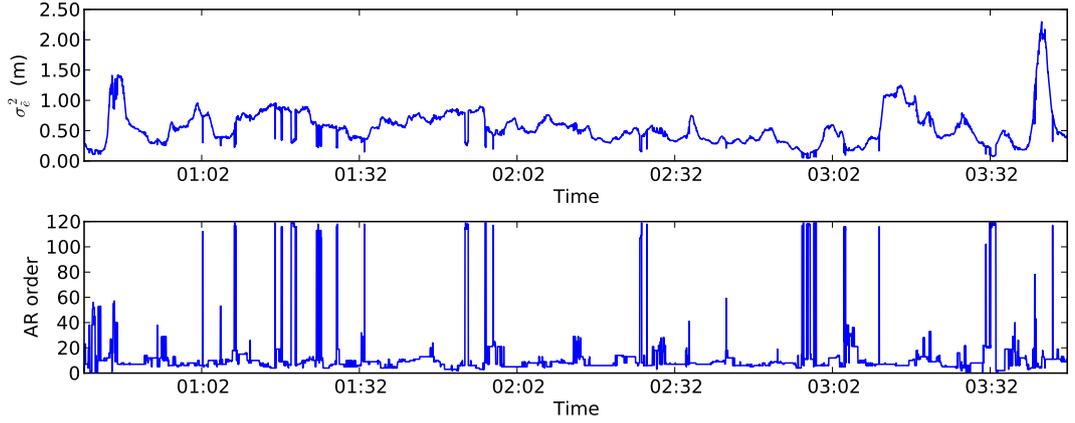
Figure 7.5: Variance of the AR(p) whitened post-fit residuals for GPS SVN05 FALK dataset with the chosen AR order for the same period.

which in turn has caused a jump in the estimated variance of the AR(p) whitened residual. This effect is particularly noticeable in Figures 7.1b and 7.1d.

7.4 Forming the whitened test statistic

The formula for the whitened predicted residuals were given in §6.8.2 and in §7.2. It was found that the variance-covariance matrices of the predicted and post-fit residuals were required in order to form a whitened test statistic. The formulae for these were derived in §7.3 using both autoregression theory and by deriving the expectation of the post-fit residual. In this section these are brought together to form the whitened test statistic using predicted residuals, for use in a Kalman filter, and the whitened test statistic using post-fit residuals, which is required for the semi-recursive Kalman filter.

7.4.1 Predicted residual

The formula for the whitened predicted residuals, adapted from Eq. (6.114), is given by

$$\tilde{\underline{v}}_k = \underline{v}_k - \sum_{i=1}^{p} \phi_i \underline{v}_{k-i} \tag{7.16}$$

where p is the order of the AR process. The variance of the predicted residual for an AR(1) process, derived in §7.3, is given by

$$Q_{\tilde{v}_k} = Q_{v_k} - \Phi_1 \Phi_1^T Q_{v_k} - Q_{v_k} \Phi_1 \Phi_1^T + \Phi_1 Q_{v_k} \Phi_1^T \qquad (7.17)$$

where Φ_1 is a diagonal matrix of AR(1) coefficients. The solution for higher order processes is more complex and can be derived by taking the expectations of $\underline{\tilde{v}}_k$ as given by Eq. (7.16) (see Priestley (1981) for more information). Using $\underline{\tilde{v}}_k$ and $Q_{\tilde{v}_k}$, Eq. (4.17) can now be modified to give the whitened test statistic,

$$\underline{\widetilde{T}}_q = \tilde{v}^T Q_{\tilde{v}}^{-1} C (C^T Q_{\tilde{v}}^{-1} C)^{-1} C^T Q_{\tilde{v}}^{-1} \underline{\tilde{v}} \qquad (7.18)$$

In §6.6 it was shown that by whitening the predicted residual $E\{\underline{\tilde{v}}_k, \tilde{v}_l^T\} = 0$ where $k \neq l$. It is now possible to express the above equation as (cf. Eq. 4.19)

$$\underline{\widetilde{T}} = \left[\sum_{i=1}^{k} C_{\tilde{v}_i}^T Q_{\tilde{v}}^{-1} \tilde{v}_i \right]^T \left[\sum_{i=1}^{k} C_{\tilde{v}_i}^T Q_{\tilde{v}}^{-1} C_{\tilde{v}_i} \right]^{-1} \left[\sum_{i=1}^{k} C_{\tilde{v}_i}^T Q_{\tilde{v}}^{-1} \tilde{v}_i \right] \qquad (7.19)$$

As shown in §4.4 this leads to the whitened local overall model test statistic,

$$\underline{\widetilde{T}}_{q,k} = \frac{\tilde{v}_k^T Q_{\tilde{v}}^{-1} \underline{\tilde{v}}_k}{m_k} \qquad (7.20)$$

where m_k is the number of observables at epoch k, and the whitened local slippage test statistic (cf. §4.4),

$$\underline{\tilde{t}}_{q=1,k} = \frac{c_k^T Q_{\tilde{v}_k}^{-1} \underline{\tilde{v}}_k}{\sqrt{c_k^T Q_{\tilde{v}_k}^{-1} c_k}} \qquad (7.21)$$

7.4.2   Post-fit residual

The formula for the whitened post-fit residuals is given by (cf. Eq. 6.114)

$$\underline{\tilde{e}}_k = \hat{e}_k - \sum_{i=1}^{p} \phi_i \hat{e}_{k-i} \qquad (7.22)$$

The coupling of $Q_{\hat{e}_k}$ and Q_{y_k} was discussed in §7.2. Since Q_{y_k} is present inside the test statistic as well as $Q_{\hat{e}_k}$ itself an adjustment to Q_{y_k} is required, which is

denoted as $\widetilde{Q}_{y_k}$. The variance-covariance matrix of the whitened AR(1) residuals can be defined as

$$Q_{\tilde{e}_k} = Q_{\hat{e}_k} - \Phi_1 \Phi_1^T Q_{\hat{e}_k} - Q_{\hat{e}_k} \Phi_1 \Phi_1^T + \Phi_1 Q_{\hat{e}_k} \Phi_1^T \qquad (7.23)$$

where Φ_1 is a diagonal matrix of AR(1) coefficients. Once again, the solution for higher order processes is more complex and can be derived by taking the expectations of $\underline{\tilde{e}}_k$ as given by Eq. (7.22) (see Priestley (1981) for more information). Where only the diagonal elements of $Q_{\tilde{e}_k}$ are required then the variance of the ith residual can be found using

$$\sigma_{\tilde{e}_{k,i}}^2 = \sigma_{\hat{e}_{k,i}}^2 \prod_{i=1}^{p} (1 - \kappa_i^2) \qquad (7.24)$$

where p is the order of the AR process and κ_i is the reflection coefficient of order i. This has the advantage that it allows higher order models to be easily estimated. Recalling Eq. (3.42),

$$Q_{\hat{e}_k} = Q_{y_k} + A_k Q_{\hat{x}_{k|k}} A_k^T$$

it can be seen that it is possible to separate Q_{y_k} from part that corresponds to the updated state vector. By considering the transformation in Eq. (7.23), it is possible to define

$$\widetilde{Q}_{y_k} = Q_{y_k} - \Phi_1 \Phi_1^T Q_{y_k} - Q_{y_k} \Phi_1 \Phi_1^T + \Phi_1 Q_{y_k} \Phi_1^T \qquad (7.25)$$

and since Q_{y_k} is assumed diagonal here, as stated in §5.2,

$$\tilde{\sigma}_{y_{k,i}}^2 = \sigma_{y_{k,i}}^2 \prod_{i=1}^{p} (1 - \kappa_i^2) \qquad (7.26)$$

Therefore, it is possible to modify Eq. (4.5) with Eq. (7.25) to give the whitened test statistic,

$$\underline{\widetilde{T}}_q = \underline{\tilde{e}}^T \widetilde{Q}_y^{-1} C_y (C_y^T \widetilde{Q}_y^{-1} \widetilde{Q}_{\tilde{e}} \widetilde{Q}_y^{-1} C_y)^{-1} C_y^T \widetilde{Q}_y^{-1} \underline{\tilde{e}} \qquad (7.27)$$

Using the same methodology applied in §4.3, this leads to the whitened overall model test statistic, given by

$$\tilde{T}_{q,k} = \frac{\underline{\tilde{e}}_k^T \widetilde{Q}_y^{-1} \underline{\tilde{e}}_k}{m_k - n_k} \qquad (7.28)$$

where m_k and n_k are the number of observables and parameters at epoch k respectively.

The whitened slippage test statistic given by

$$\underline{\tilde{w}}_k = \frac{c_k^T \widetilde{Q}_{y_k}^{-1} \underline{\tilde{e}}_k}{\sqrt{c_k^T \widetilde{Q}_{y_k}^{-1} Q_{\tilde{e}_k} \widetilde{Q}_{y_k}^{-1} c_k}} \tag{7.29}$$

and since $\widetilde{Q}_{y_k}$ is diagonal this simplifies to

$$\underline{\tilde{w}}_{k,i} = \frac{\tilde{e}_{k,i}}{\sigma_{\tilde{e}_{k,i}}} \tag{7.30}$$

for the ith observable, where $\sigma_{\tilde{e}_{k,i}}$ is derived from Eq. (7.24).

7.5 Implementation

The E-HP/PPP processing software (see §5.3.1) was modified so that the whitened local overall model (LOM) and local slippage (LS) test statistics could be calculated alongside the existing test statistics outlined in Chapter 4. In the rest of this section the issues that must be considered when implementing the whitened test statistic in GNSS PPP software will be considered in turn. Strategies for overcoming these issue and compromises, where necessary, will be given.

7.5.1 Sample size

From Chapter 6 it is known that in order to estimate the AR coefficients an estimate of the autocovariances at the lags corresponding to the AR order being estimated is required. In order to do this a sufficiently large sample is required. A method for determining sample size was presented in §5.3.3 and an interval of 240 seconds was chosen. The implication of having to estimate the AR coefficients based on a set sample size is that in GNSS processing the residuals of the observables in the first 240 seconds of a dataset cannot be whitened. Additionally, where a new satellite comes into view then the residuals for this satellite cannot be whitened until 240 seconds have elapsed.

For real-time processing this situation is unavoidable, since there is no additional data to aid in the estimation of the AR coefficients. One possible approach would be to use a smaller sample size until the point at which 240 seconds data are available. The disadvantage of this, as previously discussed, would be poorer estimates of the AR coefficients early on. Referring to the user requirements, as detailed in §2.6, this could have a significant impact on the reliability of the solution. However, the existing unwhitened test statistic could be used whilst waiting for sufficient data to estimate the autocorrelations. Additionally, this would ensure that the availability requirements of the user can be met.

In post-processing the situation is more flexible, as it is possible to use future residuals to estimate the AR coefficients for epochs where there is not a long enough history of residuals. Consider Figures 6.4b and 6.4c, where in the former the data was whitened using one value calculated for the 240 second window and in the latter a moving window of 240 seconds was used. In this case the whitened residuals do not differ greatly and therefore over such a short period using one value is not unreasonable. This would be advantageous over using a shorter sample period to estimate the AR coefficients, as suggested for real-time operation, for the same reason as given earlier.

As the E-HP/PPP processing software is written to process data to simulate real-time behaviour, the requirement to wait 240 seconds at the start of processing still holds in this implementation.

7.5.2 Gaps in the data

As mentioned earlier, if a satellite drops out of view then the moving window of 240 seconds is reset. In some cases data from a satellite is missing for just a short period of time, e.g. 5 seconds. In a kinematic environment this is entirely plausible, since there may be obstructions between the receiver and satellite as the receiver is moving. Similarly, in a static environment this may occur where an object such as a crane periodically obstructs the receiver's view of the sky.

With reference to the user requirements, as stated in §2.6, the reliability of a

solution is very important. As such, if the whitened test statistic were to be used as the sole quality control method this would lead to a degradation in the availability of the solution. In the case of small data gaps, which shall here be defined as 5 seconds or less, it can be assumed that the behaviour of the residual will not change greatly and as such the residuals that have already been stored as part of the moving window can continue to be used to estimate the AR coefficients. Thus, in the implementation, gaps of 5 seconds or less do not result in a reset of the moving window.

7.5.3 Mixing unwhitened and whitened test statistics

From the above it has been established that there will be cases where there is sufficient information in order to whiten the residual for some satellites and not others. In the situation where there is a mixture of unwhitened and whitened residuals the two different test statistics should not be compared against each other, since they are not equivalent.

In a post-processed scenario, if the assumption made above that the residuals with insufficient data can be whitened by the first available estimate, then this problem can be overcome. However, this will firstly require the AR coefficients to be estimated for all the data before the quality control can then take place. A secondary problem with this is that the AR estimates could potentially be estimated on a model error. If the assumption is made that over a 240 second period the number of model errors will be low, then the effect of one model error should not significantly affect AR estimation.

In a real-time scenario this approach is not possible. To consider only those observables that have a whitened test statistic available is also not possible as the model error could exist in one of the observables without a whitened test statistic. Hence in this implementation where all observables cannot be whitened the unwhitened test statistic is used.

7.6   Time series of the whitened test statistic

In this section the local overall model (LOM) test statistic values and the local slippage (LS) test statistic values were calculated using the whitened post-fit residuals, $\tilde{e}_k$, the variance of the whitened residuals, $Q_{\tilde{e}_k}$, and the modified variance of the observables, $\widetilde{Q}_{y_k}$ given in §7.4.2.

Local overall model test statistic

The whitened LOM test statistic was calculated using Eq. (7.28) for the ABPO, FALK, GUAM and ALBM datasets. Since the whitened test statistic cannot be calculated at every epoch and the LOM can only be calculated where all residuals are whitened example time series of the test statistic values are shown in Figures 7.6 – 7.9. The critical value F is variable as it is calculated at each epoch as a function of redundancy (see §4.3.1), therefore the minimum critical values are specified in the figure. A summary of the whitened LOM test statistic values is shown in Table 7.1.

(a) Original
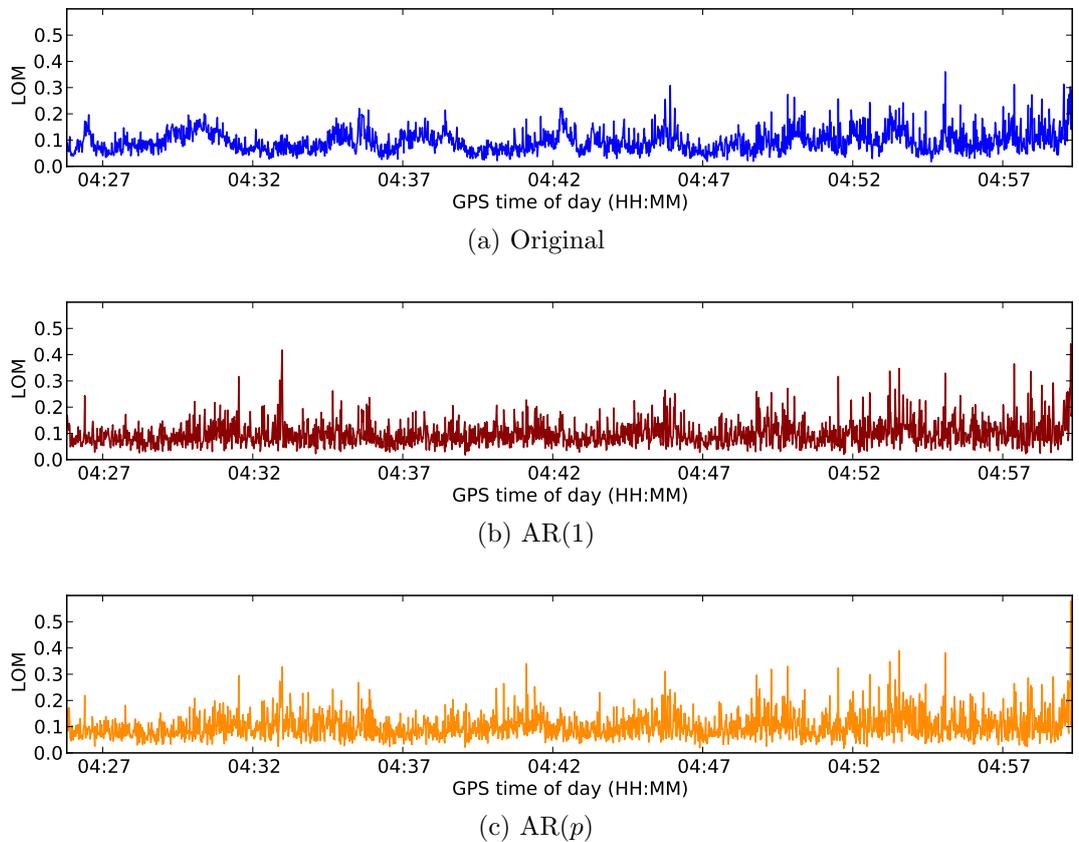
(b) AR(1)

(c) AR(p)

Figure 7.6: Example time series of local overall model test statistic values calculated using post-fit residuals for ABPO, where minimum $F_{\alpha_{m-k}} = 1.171$
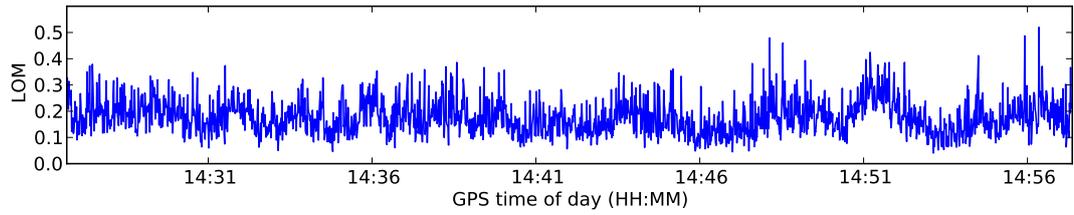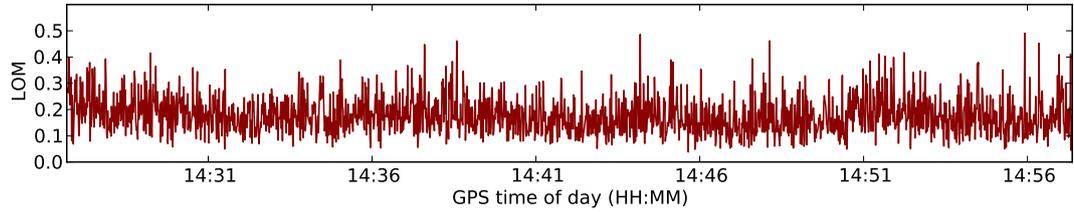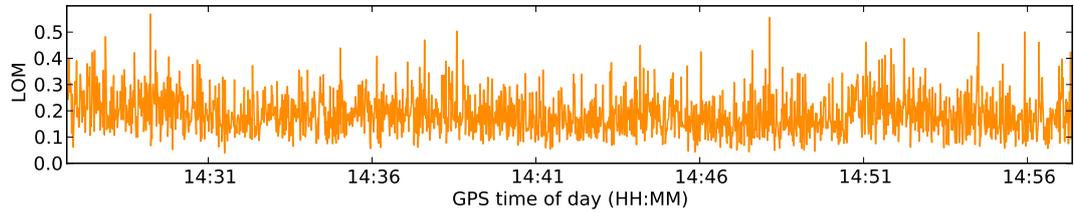
Figure 7.7: Example time series of local overall model test statistic values calculated using post-fit residuals for FALK, where minimum $F_{\alpha_{m-k}} = 1.171$
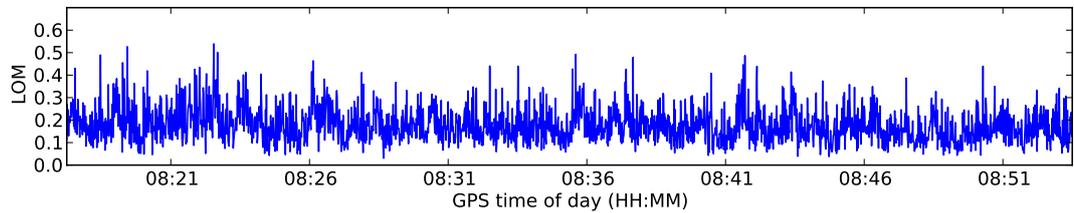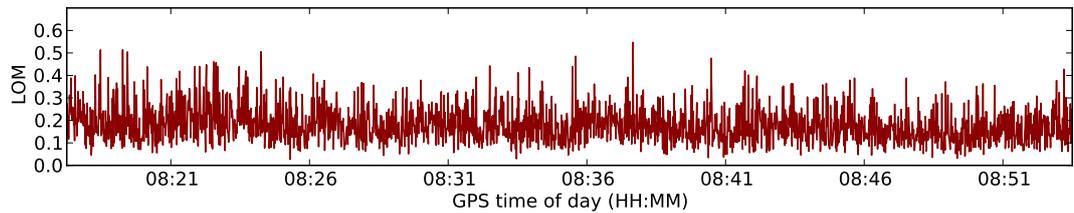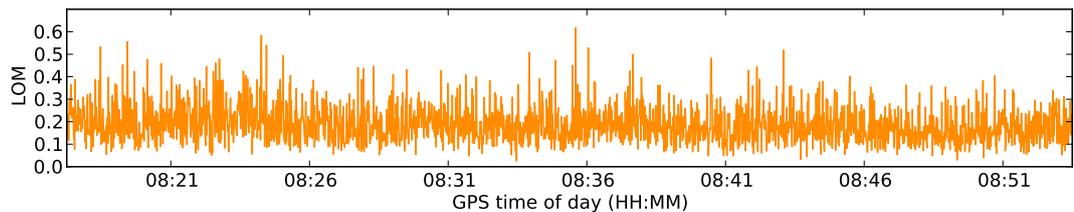
Figure 7.8: Example time series of local overall model test statistic values calculated using post-fit residuals for GUAM, where minimum $F_{\alpha_{m-k}} = 1.277$

(a) Original

(b) AR(1)

(c) AR(p)
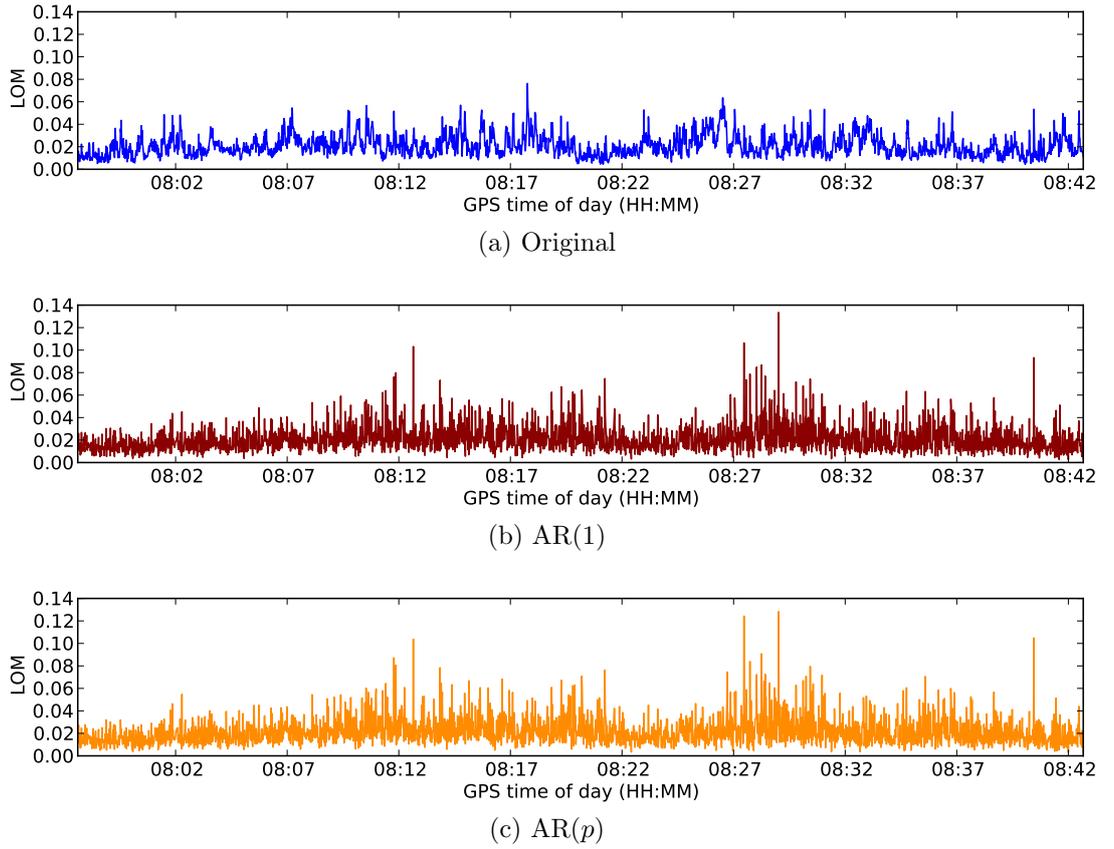
Figure 7.9: Example time series of local overall model test statistic values calculated using post-fit residuals for ALBM, where minimum $F_{\alpha_{m-k}} = 1.326$

Table 7.1: Minimum, maximum, mean, standard error and standard deviation of example unwhitened, AR(1) and AR(p) whitened local overall model test statistic time series for ABPO, FALK, GUAM and ALBM datasets.

Dataset	Solution	Min	Max	Mean	St. err.	St. dev.
ABPO	Unwhitened	0.0175	0.3603	0.0956	2.138×10^{-5}	0.0430
	AR(1)	0.0177	0.4411	0.0978	2.341×10^{-5}	0.0471
	AR(p)	0.0169	0.5782	0.1024	2.471×10^{-5}	0.0497
FALK	Unwhitened	0.0404	0.5197	0.1781	3.624×10^{-5}	0.0668
	AR(1)	0.0383	0.4908	0.1772	3.796×10^{-5}	0.0699
	AR(p)	0.0397	0.5690	0.1874	4.162×10^{-5}	0.0767
GUAM	Unwhitened	0.0304	0.5392	0.1788	3.512×10^{-5}	0.0764
	AR(1)	0.0265	0.5471	0.1803	3.588×10^{-5}	0.0781
	AR(p)	0.0260	0.6172	0.1873	3.749×10^{-5}	0.0816
ALBM	Unwhitened	0.0044	0.0763	0.0209	3.497×10^{-6}	0.0095
	AR(1)	0.0027	0.1335	0.0209	4.280×10^{-6}	0.0116
	AR(p)	0.0032	0.1283	0.0216	4.453×10^{-6}	0.0120

In each case the AR(1) and AR(p) whitened LOM test statistic values are of a similar magnitude to the unwhitened LOM test statistic values, which can be seen both in the plots and in Table 7.1. In the cases of ABPO, GUAM and ALBM the mean LOM test statistic values are lowest for the unwhitened LOM test statistic values and highest for the AR(p) whitened LOM test statistic values. In the case of standard deviations, all datasets show the unwhitened and AR(p) whitened LOM test statistic values as the lowest and highest respectively. It can also be seen in the plots that the unwhitened LOM test statistic values show time correlation characteristics, which are less noticeable for the AR(1) and AR(p) whitened LOM test statistic values. Another example time series from the ALBM dataset, shown in Figure 7.9, is a good example of this. It can also be seen that there is very close agreement between the AR(1) and AR(p) whitened LOM test statistic values.

Local slippage test statistic

The whitened LS test statistic was calculated using Eq. (7.30), which is based on the whitened post-fit residuals, for the ABPO, FALK, GUAM and ALBM datasets and time series of the test statistic values are shown in Figures 7.10 - 7.13. A summary of the mean, standard error and standard deviations of the LS test statistic values is shown in Table 7.2. The critical value for the LS test statistic is calculated using Eq. (4.13) with $\alpha_{m_k} = 0.99$, the value recommended in IMCA/OGP (2011), which gives a critical value of 2.576.

In §5.3.4 the code residuals for ABPO, FALK and GUAM did not display a high degree of time correlation (see Figures 5.15, 5.16 and 5.17, for example). As such the time series for the whitened LS test statistic does not look too dissimilar to that of the unwhitened test statistic. The last 15 minutes of the FALK dataset is one exception to this, where the test statistic appears highly correlated. However, despite this it has been successfully whitened. During this period there is an epoch where the whitened test statistic exceeds the critical value and would therefore be identified as an outlier, whereas this is not the case for the unwhitened test statistic. From Table 7.2 it can be seen that the mean of the AR(1) whitened

(a) L1 Code - unwhitened (b) L2 Code - unwhitened

(c) L1 Code - whitened, AR(1) (d) L2 Code - whitened, AR(1)

(e) L1 Code - whitened, AR(p) (f) L2 Code - whitened, AR(p)

(g) L1 Carrier - unwhitened (h) L2 Carrier - unwhitened

(i) L1 Carrier - whitened, AR(1) (j) L2 Carrier - whitened, AR(1)

(k) L1 Carrier - whitened, AR(p) (l) L2 Carrier - whitened, AR(p)
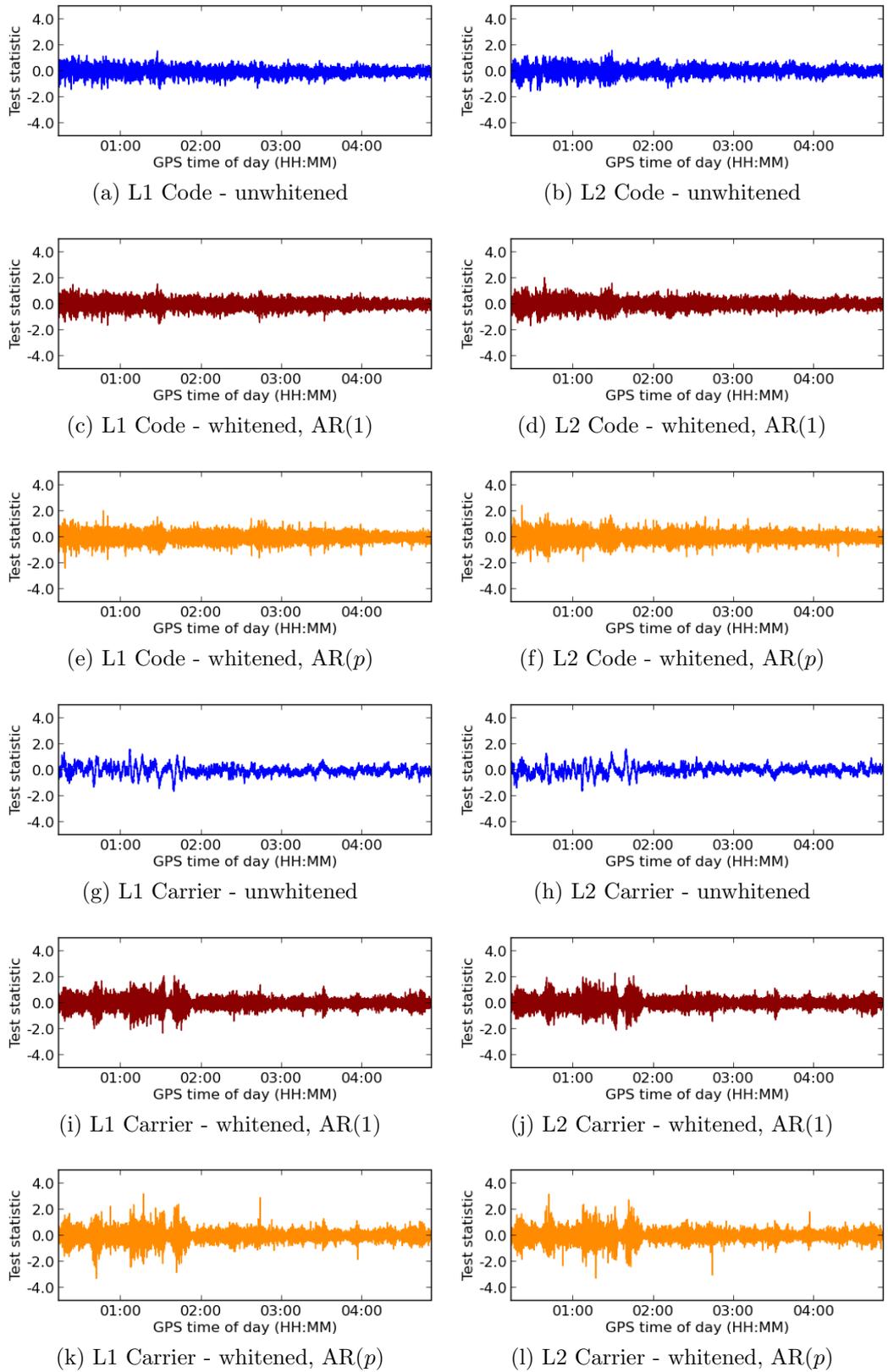
Figure 7.10: Whitened and unwhitened local slippage test statistic computed using post-fit residuals for GPS SVN05 at ABPO.

(a) L1 Code - unwhitened

(b) L2 Code - unwhitened

(c) L1 Code - whitened, AR(1)

(d) L2 Code - whitened, AR(1)

(e) L1 Code - whitened, AR(p)

(f) L2 Code - whitened, AR(p)

(g) L1 Carrier - unwhitened

(h) L2 Carrier - unwhitened

(i) L1 Carrier - whitened, AR(1)

(j) L2 Carrier - whitened, AR(1)

(k) L1 Carrier - whitened, AR(p)

(l) L2 Carrier - whitened, AR(p)
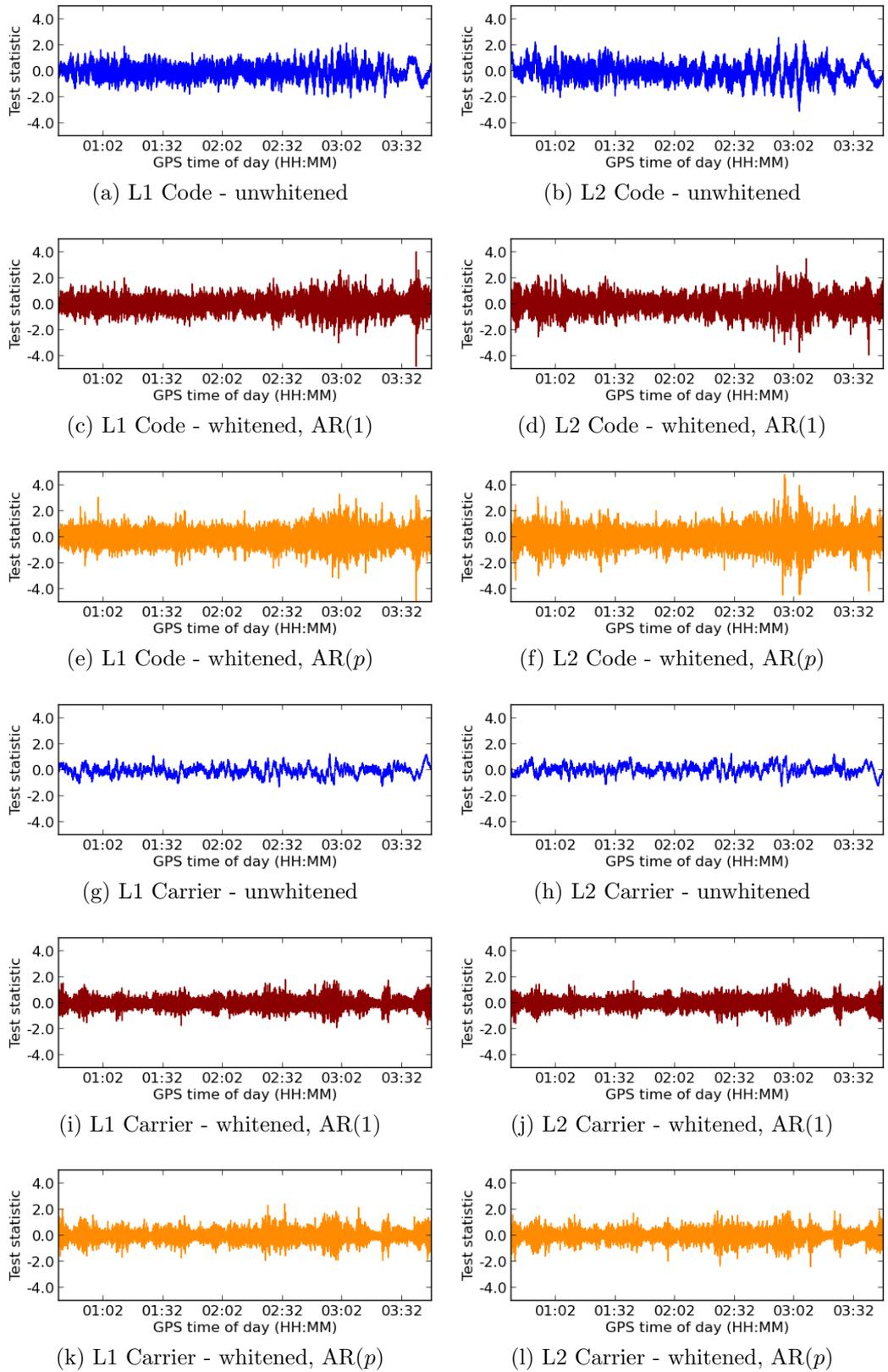
Figure 7.11: Whitened and unwhitened local slippage test statistic computed using post-fit residuals for GPS SVN05 at FALK.

(a) L1 Code - unwhitened

(b) L2 Code - unwhitened

(c) L1 Code - whitened, AR(1)

(d) L2 Code - whitened, AR(1)

(e) L1 Code - whitened, AR(p)

(f) L2 Code - whitened, AR(p)

(g) L1 Carrier - unwhitened

(h) L2 Carrier - unwhitened

(i) L1 Carrier - whitened, AR(1)

(j) L2 Carrier - whitened, AR(1)

(k) L1 Carrier - whitened, AR(p)

(l) L2 Carrier - whitened, AR(p)
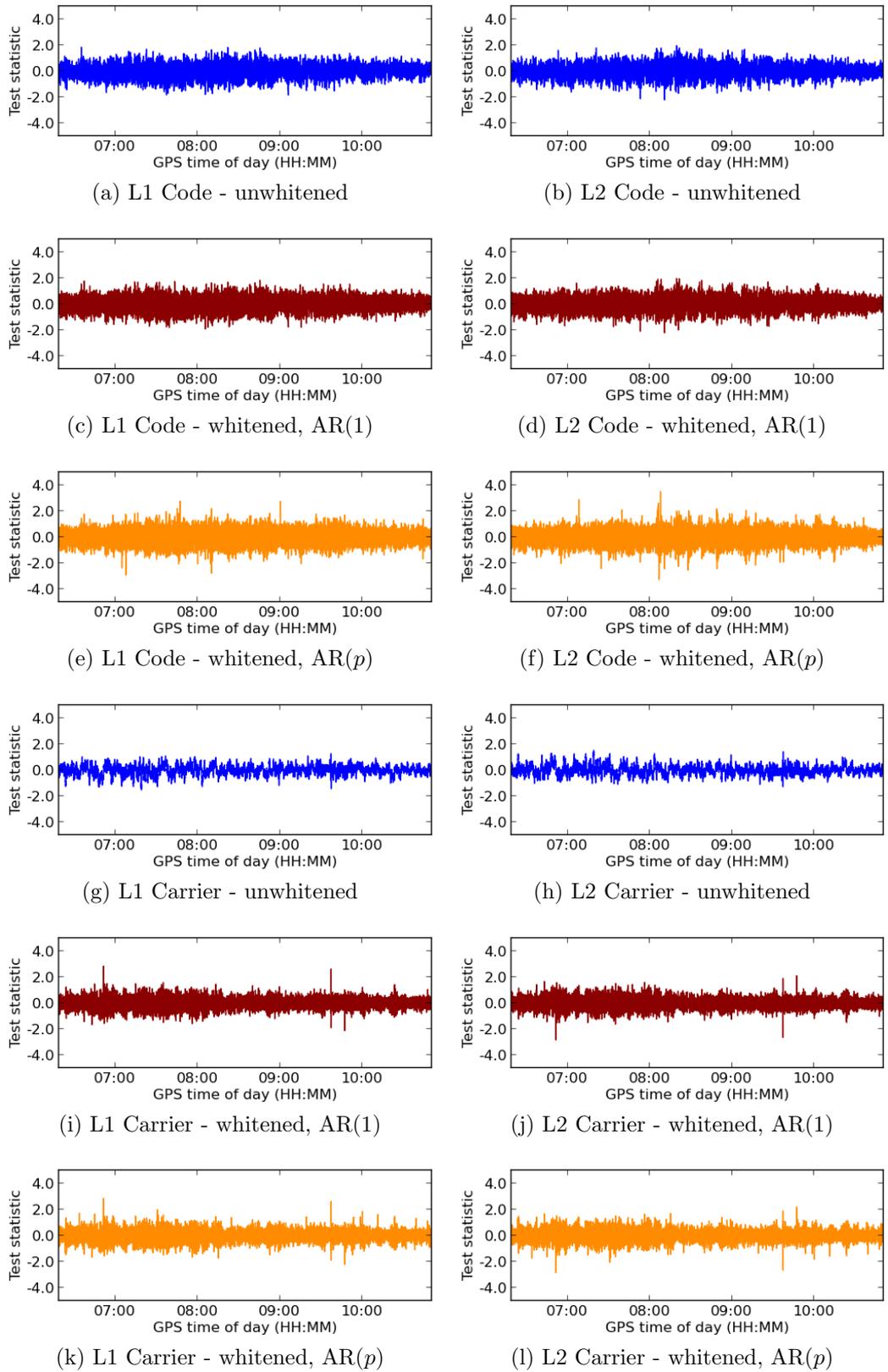
Figure 7.12: Whitened and unwhitened local slippage test statistic computed using post-fit residuals for GPS SVN05 at GUAM.

(a) L1 Code - unwhitened

(b) L2 Code - unwhitened

(c) L1 Code - whitened, AR(1)

(d) L2 Code - whitened, AR(1)

(e) L1 Code - whitened, AR(p)

(f) L2 Code - whitened, AR(p)

(g) L1 Carrier - unwhitened

(h) L2 Carrier - unwhitened

(i) L1 Carrier - whitened, AR(1)

(j) L2 Carrier - whitened, AR(1)

(k) L1 Carrier - whitened, AR(p)

(l) L2 Carrier - whitened, AR(p)
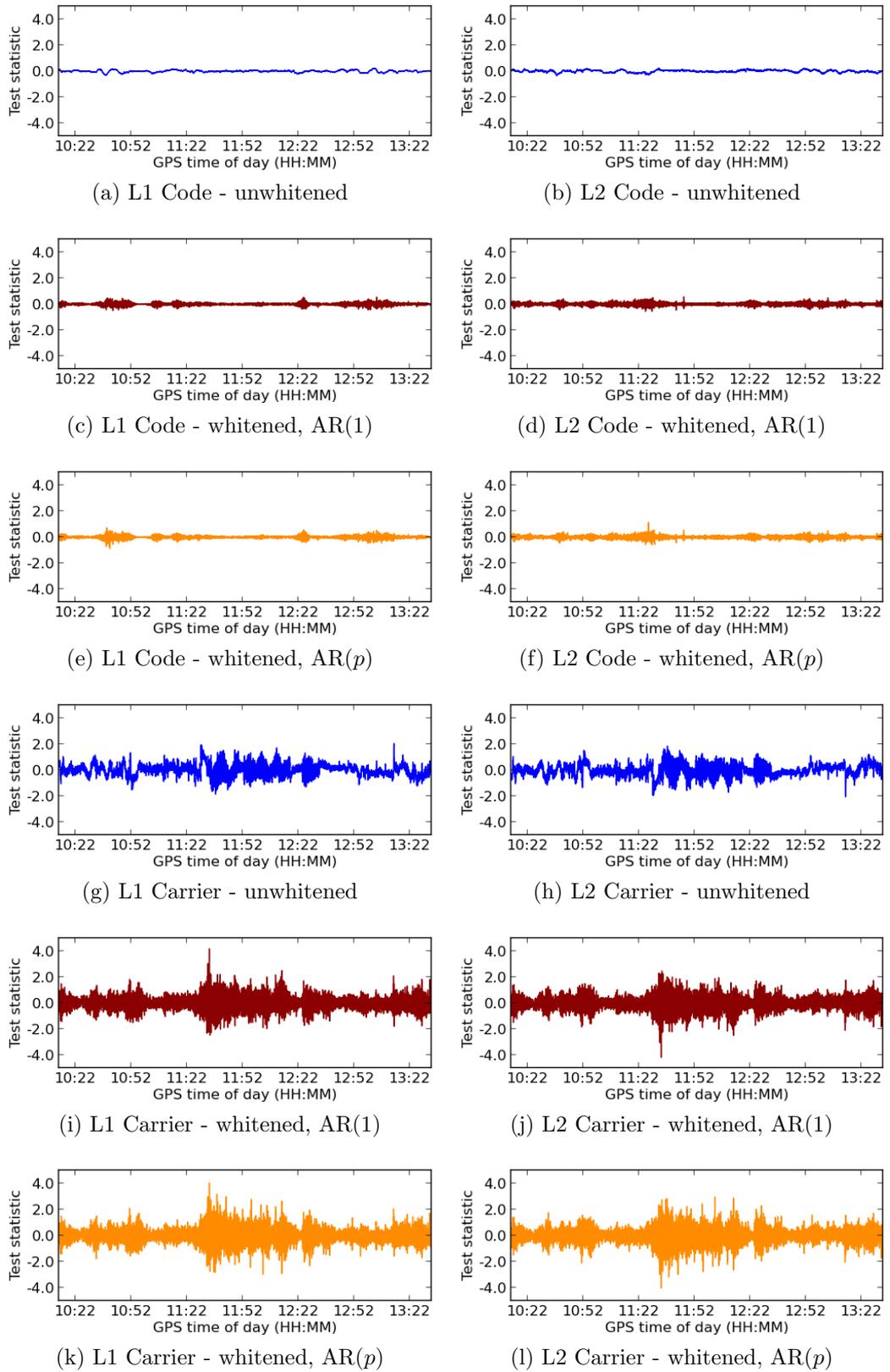
Figure 7.13: Whitened and unwhitened local slippage test statistic computed using post-fit residuals for GPS SVN17 at ALBM.

Table 7.2: Mean, standard error and standard deviation of the unwhitened, AR(1) and AR(p) whitened local slippage test statistic for ABPO, FALK, GUAM and ALBM datasets.

Dataset	Observation type	Solution	Mean	St. err.	St. dev.
ABPO	G05 L1 Code	Unwhitened	-0.0291	0.0023	0.3399
		AR(1)	-0.0157	0.0023	0.3393
		AR(p)	-0.0067	0.0025	0.3618
	G05 L1 Carrier	Unwhitened	-0.0003	0.0024	0.3481
		AR(1)	-0.0010	0.0024	0.3533
		AR(p)	-0.0018	0.0026	0.3822
	G05 L2 Code	Unwhitened	0.0321	0.0024	0.3518
		AR(1)	0.0234	0.0025	0.3563
		AR(p)	0.0081	0.0027	0.3900
	G05 L2 Carrier	Unwhitened	0.0005	0.0024	0.3479
		AR(1)	0.0011	0.0024	0.3531
		AR(p)	0.0014	0.0026	0.3816
FALK	G05 L1 Code	Unwhitened	-0.0196	0.0031	0.4673
		AR(1)	-0.0148	0.0032	0.4717
		AR(p)	-0.0079	0.0035	0.5078
	G05 L1 Carrier	Unwhitened	-0.0297	0.0026	0.3887
		AR(1)	-0.0050	0.0027	0.3883
		AR(p)	-0.0035	0.0028	0.4079
	G05 L2 Code	Unwhitened	-0.0119	0.0036	0.5307
		AR(1)	-0.0054	0.0037	0.5337
		AR(p)	-0.0032	0.0039	0.5703
	G05 L2 Carrier	Unwhitened	0.0294	0.0026	0.3890
		AR(1)	0.0050	0.0027	0.3885
		AR(p)	0.0032	0.0028	0.4071
GUAM	G05 L1 Code	Unwhitened	0.0017	0.0030	0.4506
		AR(1)	-0.0013	0.0031	0.4567
		AR(p)	-0.0019	0.0033	0.4862
	G05 L1 Carrier	Unwhitened	-0.0142	0.0025	0.3703
		AR(1)	0.0004	0.0025	0.3722
		AR(p)	0.0007	0.0026	0.3887
	G05 L2 Code	Unwhitened	0.0108	0.0030	0.4510
		AR(1)	0.0055	0.0031	0.4580
		AR(p)	0.0010	0.0033	0.4886
	G05 L2 Carrier	Unwhitened	0.0144	0.0025	0.3701
		AR(1)	-0.0004	0.0025	0.3721
		AR(p)	-0.0008	0.0026	0.3878
ALBM	G17 L1 Code	Unwhitened	-0.0030	0.0007	0.0794
		AR(1)	0.0003	0.0007	0.0797
		AR(p)	-0.0004	0.0008	0.0906
	G17 L1 Carrier	Unwhitened	0.0258	0.0040	0.4431
		AR(1)	0.0074	0.0041	0.4452
		AR(p)	0.0015	0.0045	0.4917
	G17 L2 Code	Unwhitened	-0.0014	0.0008	0.0940
		AR(1)	0.0005	0.0008	0.0891
		AR(p)	0.0008	0.0010	0.1058
	G17 L2 Carrier	Unwhitened	-0.0256	0.0040	0.4421
		AR(1)	-0.0073	0.0041	0.4443
		AR(p)	-0.0016	0.0045	0.4907

test statistic is closer to 0 than the mean of the unwhitened test statistic. The AR(p) whitened test statistic is even closer to 0. Since $E\{\underline{w}_k\} = 0$ (see §4.3.2), this shows that whitening the residuals results in a test statistic that is closer to its expected value. The unwhitened code residuals for the ALBM dataset by contrast are highly correlated (see Figure 5.18, for example) and this is reflected in the time series for the unwhitened test statistic. The magnitude of the unwhitened test statistic is also much smaller, resulting in a smaller amplitude of the time series. The magnitude of the whitened code test statistic is also small, when compared to the carrier phase test statistic. Once again Table 7.2 shows the means of the AR(1) and AR(p) whitened test statistics are closer to the theoretical mean value of zero than the unwhitened test statistic. Looking at the standard deviations of the local slippage test values for the code observations in most cases the standard deviations of the whitened test statistics are greater than those of the unwhitened ones. The exceptions to this are the standard deviations of the AR(1) whitened test statistic using L1 code observations at ABPO and the L2 code observations at ALBM. The standard deviations of the AR(p) whitened test statistic values for code observations are greater than for the unwhitened and AR(1) whitened test statistics in all cases.

As previously noted, both the unwhitened and whitened carrier phase test statistic values for the ALBM dataset are much larger in magnitude than those for the code observations. From Table 7.2 it can be seen that the standard deviations of the carrier phase test statistic values for the three solutions are very similar. The whitened carrier phase test statistic values for ABPO, FALK and GUAM show similar behaviour to those at the ALBM dataset, in that a greater number of peaks are seen in comparison to the unwhitened dataset. The whitened values also show a similar magnitude to the unwhitened ones, although once again the characteristic pattern seen with time correlation is no longer present. The mean values of the AR(1) and AR(p) whitened test statistics for carrier phase observations shown in Table 7.2 for FALK and GUAM follow what has been seen previously, being closer the theoretical mean of zero than for the unwhitened test statistic. ABPO is an exception to this, with mean values increasing. In the cases of ABPO, GUAM and

ALBM the standard deviations of the whitened test statistic values of carrier phase observations are greater than that of the unwhitened test statistic. FALK is an exception to this, where the standard deviation of AR(1) whitened test statistic values for both L1 and L2 carrier phase observations are smaller than those for the unwhitened observations. However, since the differences in both cases are less than 1×10^{-3} they can be considered negligible. In the case of the $AR(p)$ whitened test statistic the standard deviations are larger for all carrier phase observations.

In conclusion, the whitened test statistic values are of a similar magnitude to the unwhitened values. As seen with the post-fit residuals themselves, when plotted as a time series the residuals have lost the characteristic pattern of time correlation and closer resemble what would be expected from white noise. It is important to note that despite the fact that the whitened residuals have a much smaller amplitude, due to the correlation being so high, the effect of the smaller variance yields test statistics that are of similar magnitude, but without the time correlation present. In some cases the whitened test statistic contains small spikes that exceed the critical value that were not seen before whitening. However, it is important to keep in mind with the chosen level of significance ($\alpha_{m_k} = 0.99$), it is expected that 1% will be false rejections.

7.7 Detection of observation biases

The general behaviour of the whitened test statistic was shown in §7.6 to have a more consistent mean around zero and have a shape more like white noise. When detecting biases in the observables it is not a time series for one satellite and observation type that is used, rather a range of test statistic values for the different observation types at a single epoch that are compared against each other in order to determine the source of any model error (see §4.2).

It has already been discussed in §7.5 that unwhitened and whitened test statistics cannot be directly compared. However, it can be seen that where an outlier occurs at a given epoch both the unwhitened and whitened test statistic values have a similar relative magnitude.

Table 7.3: Local slippage test statistic values computed using post-fit residuals for all satellite and observation types at 09:10:56 for ALBM dataset.

(a) Unwhitened

Observation	TS
G09 L2 Carrier	-126.352
G09 L1 Carrier	126.351
G10 L1 Carrier	-61.530
G10 L2 Carrier	61.438
G18 L1 Carrier	-21.040
G18 L2 Carrier	21.029
G21 L1 Carrier	-20.746
G21 L2 Carrier	20.737
G08 L1 Carrier	12.833
G08 L2 Carrier	-12.820
G24 L1 Carrier	-12.351
G24 L2 Carrier	12.347
G26 L1 Carrier	8.837
G26 L2 Carrier	-8.830
G28 L1 Carrier	6.577
G28 L2 Carrier	-6.572
G27 L2 Carrier	1.595
G27 L1 Carrier	-1.594
G15 L2 Carrier	1.580
G15 L1 Carrier	-1.578
G09 L2 Code	-0.637
G10 L2 Code	-0.520
G21 L2 Code	-0.451
G28 L2 Code	0.428
G26 L2 Code	0.424
G10 L1 Code	-0.381
G08 L1 Code	0.380
G21 L1 Code	-0.349
G09 L1 Code	-0.310
G18 L2 Code	-0.296
G24 L2 Code	-0.255
G26 L1 Code	0.248
G24 L1 Code	-0.235
G08 L2 Code	0.214
G18 L1 Code	-0.164
G15 L2 Code	0.140
G15 L1 Code	0.136
G27 L1 Code	0.055
G28 L1 Code	0.052
G27 L2 Code	0.006

(b) Whitened, AR(1)

Observation	TS
G10 L1 Carrier	-160.807
G10 L2 Carrier	160.623
G18 L1 Carrier	-74.497
G18 L2 Carrier	74.492
G21 L1 Carrier	-68.212
G21 L2 Carrier	68.095
G24 L2 Carrier	36.102
G24 L1 Carrier	-36.079
G08 L1 Carrier	32.564
G08 L2 Carrier	-32.520
G26 L1 Carrier	13.039
G26 L2 Carrier	-13.009
G28 L1 Carrier	10.343
G28 L2 Carrier	-10.335
G08 L1 Code	4.900
G21 L1 Code	-4.477
G27 L1 Carrier	-4.281
G27 L2 Carrier	4.278
G08 L2 Code	4.263
G28 L1 Code	3.681
G10 L1 Code	-3.474
G21 L2 Code	-3.431
G26 L2 Code	3.183
G28 L2 Code	2.863
G15 L2 Carrier	2.492
G15 L1 Carrier	-2.488
G24 L1 Code	-1.988
G26 L1 Code	1.721
G18 L1 Code	-1.645
G24 L2 Code	-1.634
G10 L2 Code	-1.474
G15 L1 Code	1.014
G15 L2 Code	0.947
G18 L2 Code	-0.924
G27 L2 Code	0.037
G27 L1 Code	-0.024

(c) Whitened, AR(p)

Observation	TS
G10 L1 Carrier	-163.077
G10 L2 Carrier	162.952
G18 L1 Carrier	-78.124
G18 L2 Carrier	78.117
G21 L1 Carrier	-70.692
G21 L2 Carrier	70.563
G24 L2 Carrier	36.729
G24 L1 Carrier	-36.704
G08 L1 Carrier	34.628
G08 L2 Carrier	-34.570
G26 L1 Carrier	12.709
G26 L2 Carrier	-12.677
G28 L1 Carrier	10.807
G28 L2 Carrier	-10.799
G08 L2 Carrier	4.765
G08 L1 Code	4.680
G21 L1 Code	-4.563
G27 L1 Carrier	-4.391
G27 L2 Carrier	4.388
G10 L1 Code	-3.855
G26 L2 Code	3.838
G28 L1 Code	3.789
G21 L2 Code	-3.590
G28 L2 Code	2.862
G15 L2 Carrier	2.384
G15 L1 Carrier	-2.379
G26 L1 Code	2.005
G24 L1 Code	-2.002
G18 L1 Code	-1.945
G24 L2 Code	-1.656
G10 L2 Code	-1.314
G15 L2 Code	1.118
G15 L1 Code	0.981
G18 L2 Code	-0.958
G27 L2 Code	0.038
G27 L1 Code	-0.009

Table 7.4: Local slippage test statistic values computed using post-fit residuals for all satellite and observation types at 23:48:03 for FALK dataset.

(a) Unwhitened

Observation	TS
G09 L1 Carrier	-1296375.917
G09 L2 Carrier	1296152.614
G24 L1 Carrier	686542.044
G24 L2 Carrier	-685447.866
G30 L1 Carrier	-151748.680
G30 L2 Carrier	151660.770
G04 L1 Carrier	78156.561
G04 L2 Carrier	-78105.687
G29 L1 Carrier	55198.347
G29 L2 Carrier	-55179.419
G12 L1 Carrier	22125.374
G12 L2 Carrier	-22123.702
G09 L2 Code	-12307.127
G31 L1 Carrier	-11854.613
G31 L2 Carrier	11831.911
G24 L2 Code	7435.762
G09 L1 Code	-5945.767
G24 L1 Code	5784.018
G30 L2 Code	-3588.185
G30 L1 Code	-2922.287
G02 L1 Carrier	-1251.347
G02 L2 Carrier	1242.939
G04 L2 Code	850.140
G31 L1 Code	-634.965
G04 L1 Code	593.522
G29 L2 Code	591.715
G31 L2 Code	-583.248
G02 L2 Code	-572.268
G02 L1 Code	-546.390
G29 L1 Code	370.853
G12 L1 Code	-139.640
G12 L2 Code	-55.274

(b) Whitened, AR(1)

Observation	TS
G09 L1 Carrier	-9244490.297
G09 L2 Carrier	9183090.967
G24 L1 Carrier	3654804.161
G24 L2 Carrier	-3582041.316
G30 L1 Carrier	-693848.045
G30 L2 Carrier	693652.263
G04 L1 Carrier	-232559.361
G04 L2 Carrier	230287.629
G29 L1 Carrier	107408.001
G29 L2 Carrier	-107137.669
G12 L1 Carrier	67722.536
G12 L2 Carrier	-67294.665
G09 L1 Code	-29133.756
G31 L1 Carrier	-28528.439
G31 L2 Carrier	28070.062
G09 L2 Code	-19002.323
G24 L1 Code	9904.810
G24 L2 Code	8379.138
G02 L1 Carrier	-5567.862
G02 L2 Carrier	5522.728
G30 L2 Code	-4767.013
G30 L1 Code	-3184.374
G04 L2 Code	1732.923
G04 L1 Code	1111.546
G31 L1 Code	-734.587
G29 L2 Code	729.367
G31 L2 Code	-680.409
G02 L1 Code	-634.117
G02 L2 Code	-621.328
G29 L1 Code	379.922
G12 L1 Code	-152.675
G12 L2 Code	-68.704

(c) Whitened, AR(p)

Observation	TS
G09 L1 Carrier	-9244490.297
G09 L2 Carrier	9183090.967
G24 L1 Carrier	3654804.161
G24 L2 Carrier	-3582041.316
G30 L1 Carrier	-763633.465
G30 L2 Carrier	763458.171
G04 L1 Carrier	-262960.816
G04 L2 Carrier	244085.963
G29 L1 Carrier	147498.412
G29 L2 Carrier	-146694.250
G12 L1 Carrier	72063.775
G12 L2 Carrier	-71622.512
G09 L1 Code	-32646.463
G31 L1 Carrier	-29348.662
G31 L2 Carrier	28869.193
G09 L2 Code	-20584.368
G24 L1 Code	13310.716
G24 L2 Code	10767.704
G02 L1 Carrier	-5957.168
G02 L2 Carrier	5915.927
G30 L2 Code	-5640.479
G30 L1 Code	-3233.466
G04 L2 Code	2043.747
G04 L1 Code	1218.913
G29 L2 Code	831.533
G31 L1 Code	-820.579
G02 L1 Code	-749.722
G02 L2 Code	-747.098
G31 L2 Code	-721.697
G29 L1 Code	407.462
G12 L1 Code	-165.846
G12 L2 Code	-80.006

(a) Unwhitened, L1 Code

(b) Unwhitened, L2 Code

(c) Whitened, L1 Code

(d) Whitened, L2 Code

(e) Unwhitened, L1 Carrier

(f) Unwhitened, L2 Carrier

(g) Whitened, L1 Carrier

(h) Whitened, L2 Carrier

Figure 7.14: Local slippage test statistic values computed using post-fit residuals at 09:10:56 for ALBM dataset. In each case the absolute values are taken, i.e. negative values are now positive.

Figure 7.14 shows the test statistic at an epoch where a cycle slip in the L1 carrier phase for satellite G09 has occurred. At this particular epoch there is not enough data to estimate the AR coefficients for satellite G09 and as such does not appear in the subfigures for the whitened local slippage test statistic in Figure 7.14. However, despite this it is evident that all other values take on similar relative values. If the observables are placed in order of their test statistic values, from highest to lowest, then with the exception of satellite G09 the sequence of observation types

is the same, as verified by Table 7.3. By comparing the y-axes of the plots showing the unwhitened and whitened local slippage test statistics in Figure 7.14 it is also evident that the types of test statistic are of a different scale. Since the two test statistics are calculated using different residuals, i.e. unwhitened and whitened, it does not make sense to combine the unwhitened and whitened statistics when there is insufficient data to calculate a whitened test statistic for each observable.

Figure 7.15: Local slippage test statistic values computed using post-fit residuals at 23:48:03 for FALK dataset. In each case the absolute values are taken, i.e. negative values are now positive.

Another example of an epoch containing a cycle slip is shown in Figure 7.15.

In this case the observable containing a cycle slip, G09 L1 Carrier, is detected by the whitened test statistic as there is sufficient data to estimate the AR coefficients. There is also sufficient data to compute the whitened test statistic for all satellite and observation types, showing that in this case the whitened test statistic could be used over the original test statistic. From Table 7.4 it can also be seen that in addition to correctly identifying the cycle slip the order of the first 15 test statistic values is identical.

(a) Unwhitened, L1 Code

(b) Unwhitened, L2 Code

(c) Whitened, L1 Code

(d) Whitened, L2 Code

(e) Unwhitened, L1 Carrier

(f) Unwhitened, L2 Carrier

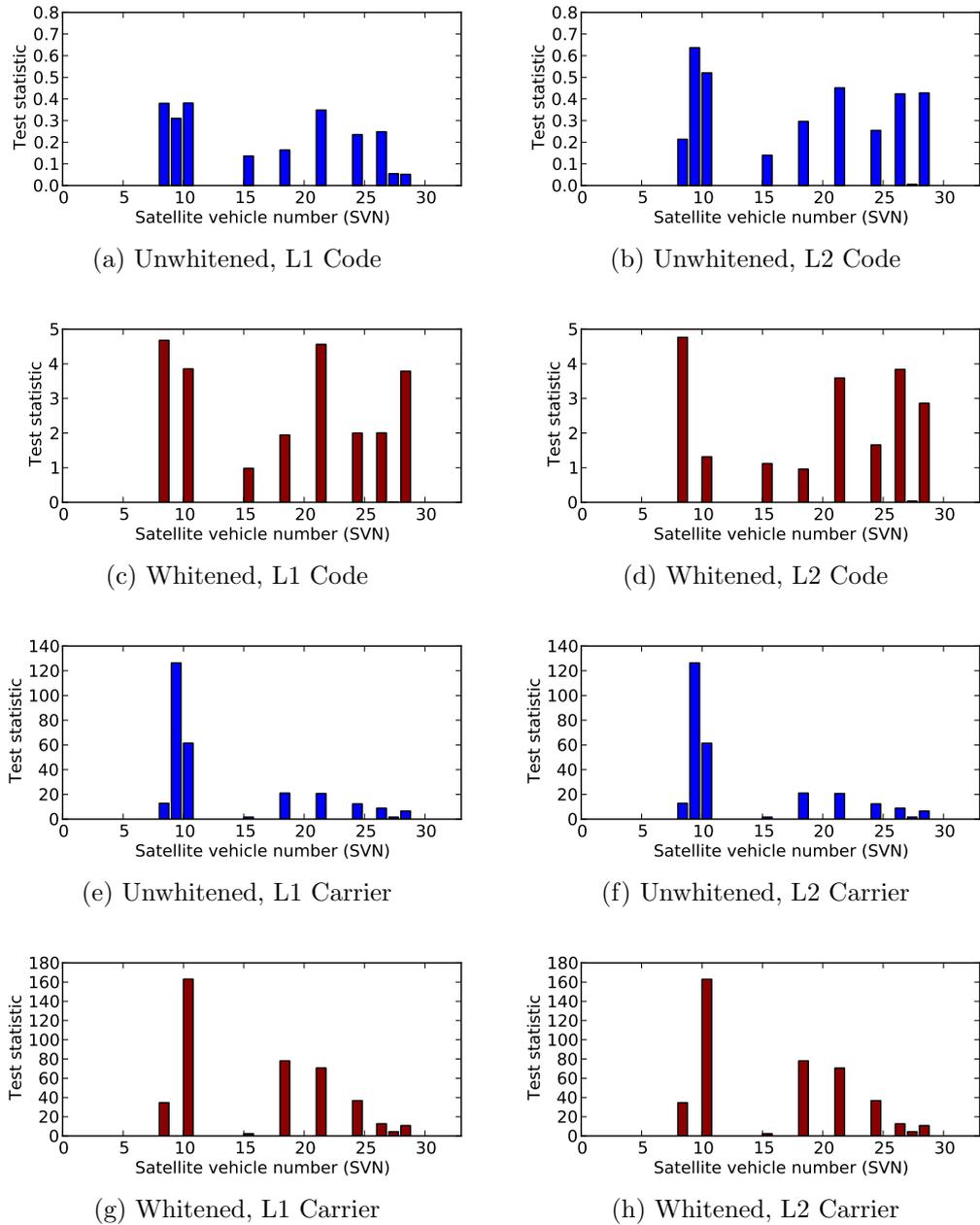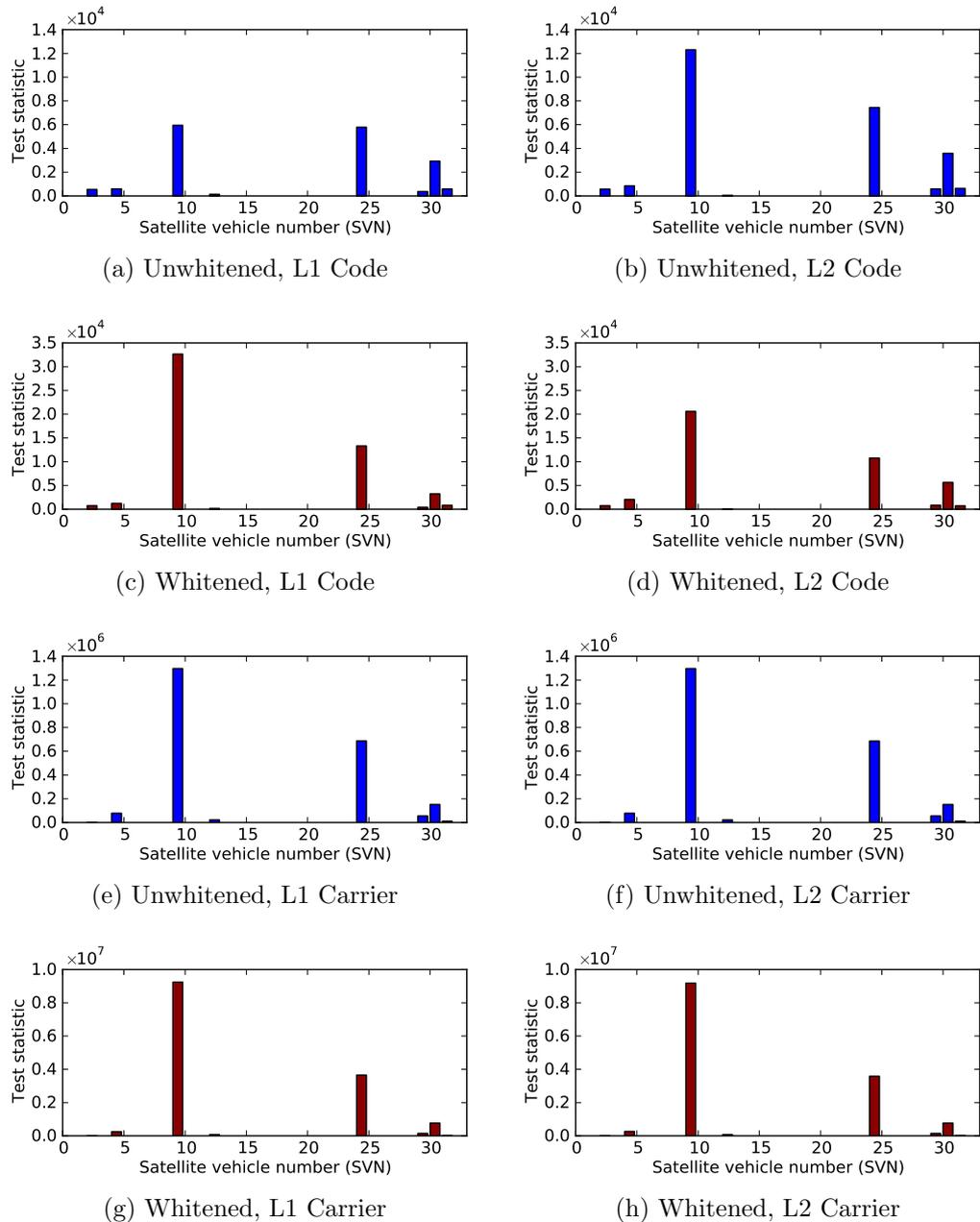(g) Whitened, L1 Carrier

(h) Whitened, L2 Carrier
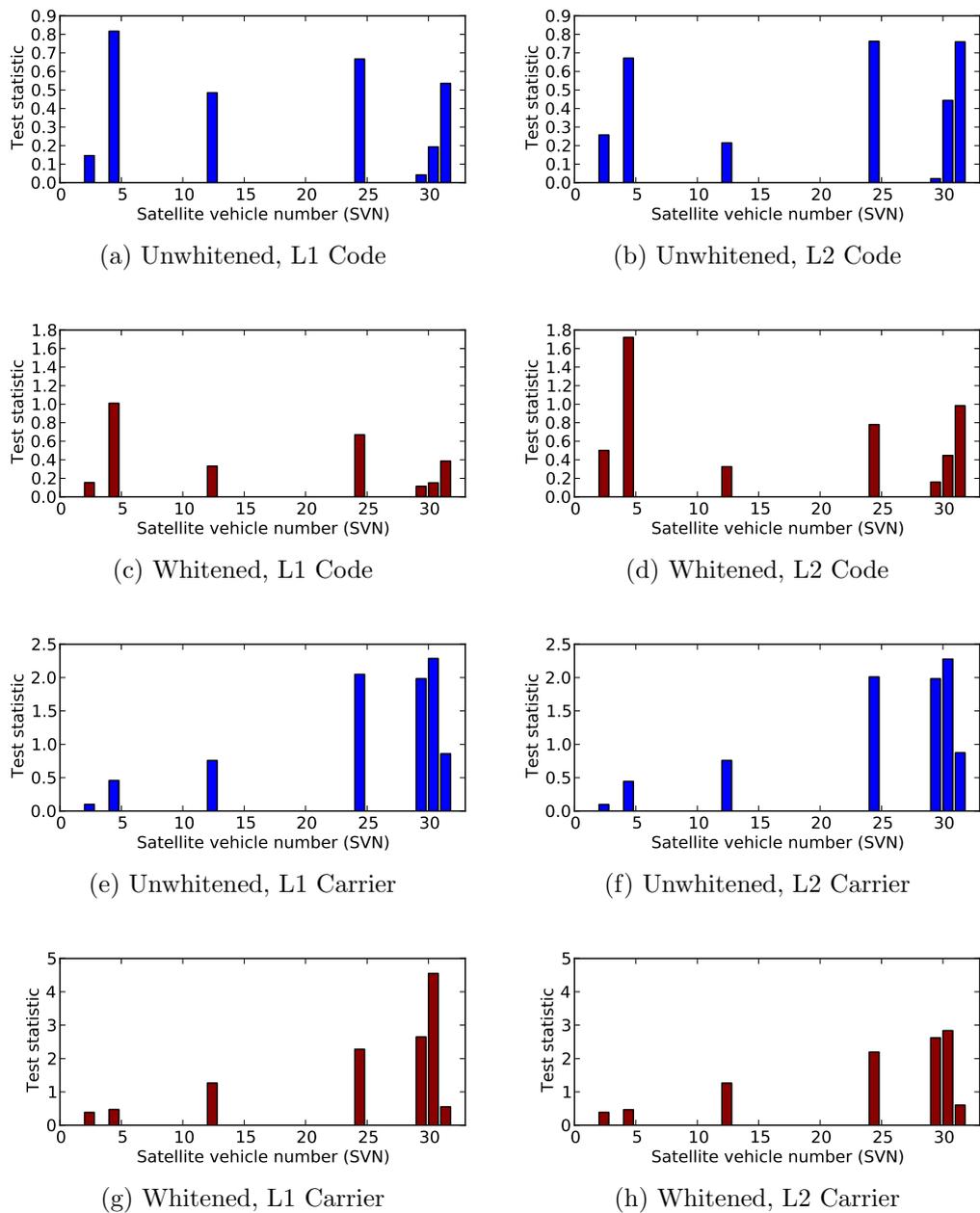
Figure 7.16: Local slippage test statistic values computed using post-fit residuals at 23:59:59 for FALK dataset. In each case the absolute values are taken, i.e. negative values are now positive.

By comparison, the values shown in Figure 7.16 are for an epoch that contains no observational biases. It can be seen that in this case there is no common pattern between the original and whitened test statistic values, although they are of a similar magnitude.

It is clear that although the values are different for epochs where there is no observational bias, when a cycle slip occurs the whitened test statistic follows a similar pattern to that seen by the unwhitened test statistic. In the case where all observables have not been whitened, such as Figure 7.14, the order is still preserved, although in this particular example it is the unwhitened value that contains the bias.

Changes in the geometry of the solution

Until now it has been assumed that following convergence the recursive filter is in a relatively steady state and as a result it is possible to model the residuals as an AR process. However, when an outlying observable is rejected this causes a sudden change in structure of the observables that make up the solution. Consider the case where there is a bias in a carrier phase observation at epoch k and that this has been successfully identified using the whitened test statistic. The carrier phase observation is removed from the solution. Given that carrier phase observations have a much higher weight, and therefore a greater impact on the solution, this will cause a change to the geometry of the solution at this epoch. In some cases this could result in a 'jump' in the time series of residuals for a given satellite and observation type. Such a 'jump' is shown in Figure 7.17a. The effect of this jump on the AR(1) whitened residual can be seen in 7.17b. The estimate of the whitened residual at the next epoch is influenced by the relatively large gap between the two values and therefore results in the spike that is marked by the vertical red line.

The local slippage test statistics that correspond to the residuals shown in Figure 7.17 are shown in Figure 7.18. Whereas the impact of the jump in the unwhitened residuals on its corresponding test statistic is minimal, the jump seen in the un-whitened AR(1) residual also results in a prominent jump in the whitened local slippage test statistic. Depending on the level of significance chosen it could result
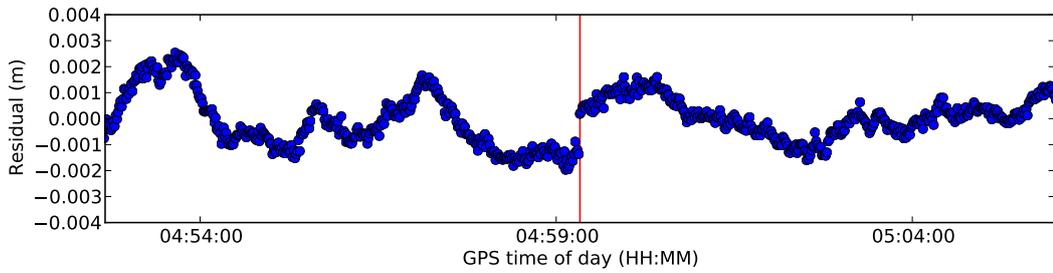
(a) Unwhitened post-fit residual
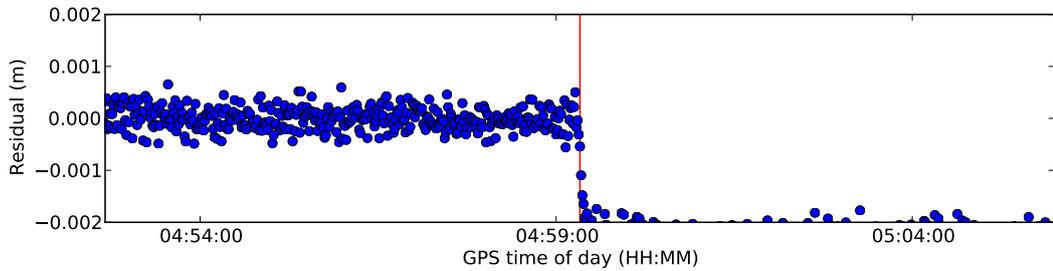
(b) Whitened post-fit residual, AR(1)

Figure 7.17: 'Jump' in time series of post-fit residuals statistic caused by a change in the geometry of the least squares solution.
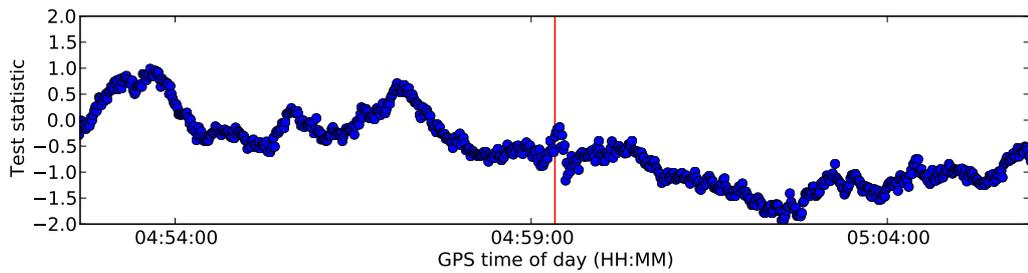
(a) Unwhitened test statistic
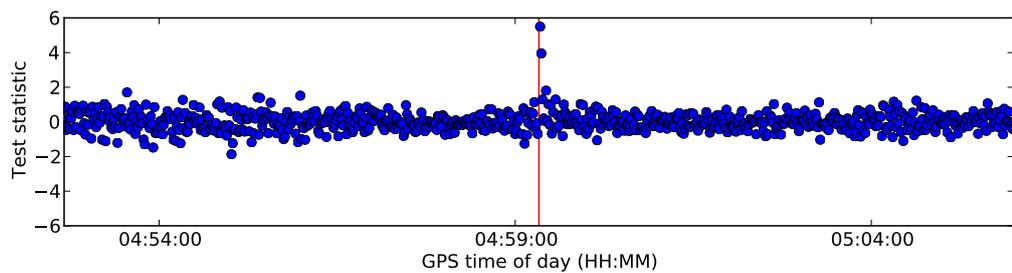
(b) Whitened test statistic, AR(1)

Figure 7.18: Effect on the local slippage test statistic of a 'jump' in time series of post-fit residuals statistic caused by a change in the geometry of the least squares solution.

in a Type I error (see 4.2).

As mentioned in §4.6, removing a carrier phase parameter means that the ambiguity for the satellite and frequency in question must be reset, as a model error detected in a carrier phase measurement could be the result of a cycle slip and hence

the previous ambiguity will no longer be valid. In contrast to the sudden change in geometry caused by the parameter reset, the geometry of the solution will be very similar at the epoch immediately after the cycle slip and will change gradually as the a posteriori error of the ambiguity decreases. Therefore, this slowly changing geometry does not cause the kind of jump in the test statistic seen here.

It was seen in Chapter 6 that for an AR(1) process the estimate for ϕ_1 is typically close to 1. As $\phi_1 \to 1$ the process can be said to approximate the first difference of the residuals. Should a 'jump' occur in the residual time series then this would be seen as a spike in the first difference solution. Considering this, when ϕ_1 is close to 1, it is likely that such a jump would be seen in the AR(1) whitened residuals. Figure 7.17b shows the AR1(1) whitened residual for Figure 7.17a. The red line indicates the point at which a cycle slip has been identified, but for a different observable. Not only is the effect of this jump increased as the amplitude of the whitened residual is much smaller, but as the variance of the whitened residual is not likely to change much over such a short time period this observable will be scaled much higher when the whitened test statistic is calculated. It will therefore be identified as containing a bias.

In order to overcome this problem it is ideal to have an estimate of the residual with the new geometry, i.e. using the same set of observables, at the epochs that are to be used to whiten the residual. For an AR(1) process this would only be $\hat{\underline{e}}_{k-1}$, but in the case of an AR(p) process this would include residuals $\{\hat{\underline{e}}_{k-1}, \ldots, \hat{\underline{e}}_{k-p}\}$. In order to do this the design matrix and state vector at previous states must be stored. For high order models this can add considerable complexity.

For the purposes of this study, adapting the software to do this would require a radical rewrite and given this complexity could not be achieved in a timely manner. As such, a compromise was made whereby the unwhitened test statistic was used once one bias had been identified and at the subsequent epoch. This was done so that any jump in the unwhitened residual would not cause observables to be erroneously identified as containing a bias.

The ABPO, FALK, GUAM and ALBM datasets were processed using the mod-
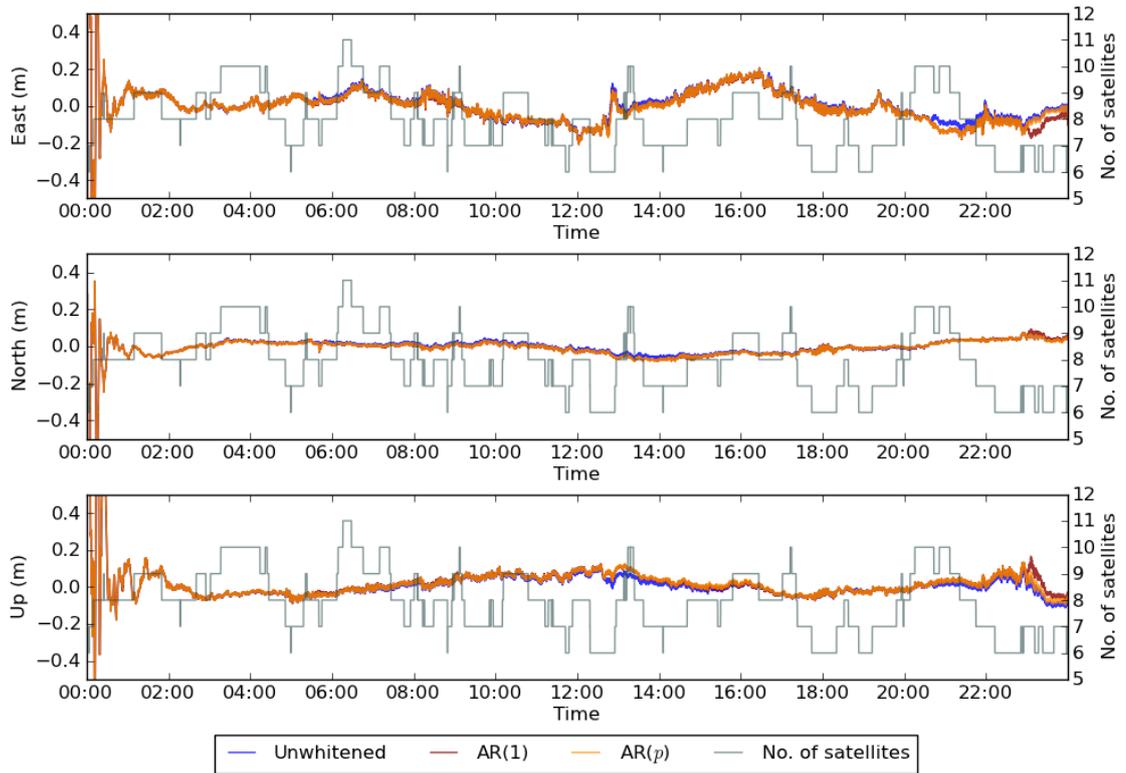
		Unwhitened	AR(1)	AR(p)
East	Mean (m)	0.0146	0.0045	0.0081
	St. dev. (m)	0.0712	0.0746	0.0738
North	Mean (m)	-0.0074	-0.0116	-0.0116
	St. dev. (m)	0.0294	0.0324	0.0312
Up	Mean (m)	0.0210	0.0313	0.0288
	St. dev. (m)	0.0409	0.0455	0.0452

Figure 7.19: Local ENU positions for ABPO using the unwhitened local slippage test statistic, AR(1) whitened local slippage test statistic and AR(p) local slippage test statistic with the level of significance set at 99%. In all cases the post-fit residuals were used to calculate the test statistics. The mean and standard deviation for each plot are shown in the table below.

ified software with both an AR(1) and AR(p) whitened test statistic used for detecting biases in the observables. The data was also processed using the unwhitened test statistic for detecting biases. In all cases the critical value for the slippage test was calculated with a level of significance of 99%. Figure 7.19 shows the estimated local east, north and up (ENU) positions for the entire 24 hour period of the ABPO dataset when processed using the unwhitened test statistic, the AR(1) whitened test statistic and AR(p) whitened test statistic. The values are all relative to a common mean east, north and up coordinate. In addition to the positions the satellite

Figure 7.20: Local ENU positions for FALK using the unwhitened local slippage test statistic, AR(1) whitened local slippage test statistic and AR(p) local slippage test statistic with the level of significance set at 99%. In all cases the post-fit residuals were used to calculate the test statistics. The mean and standard deviation for each plot are shown in the table below.
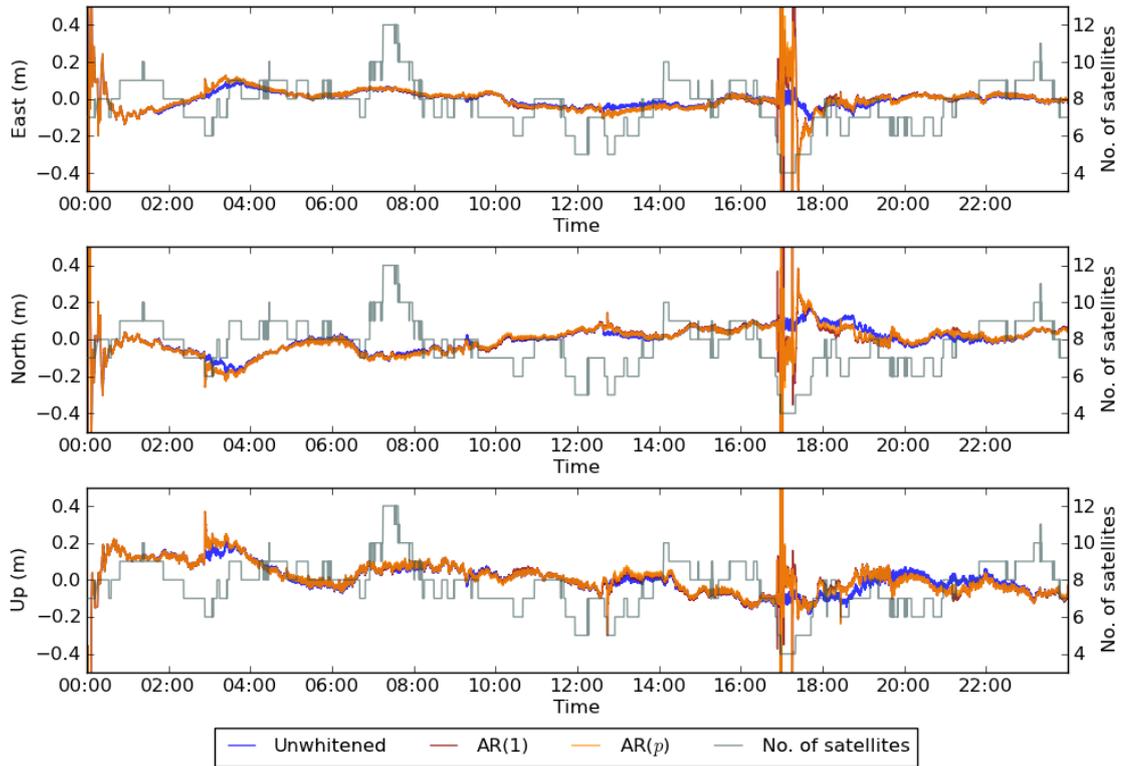
		Unwhitened	AR(1)	AR(p)
East	Mean (m)	-0.0038	-0.0034	-0.0031
	St. dev. (m)	0.0343	0.1051	0.1059
North	Mean (m)	0.0017	-0.0025	-0.0021
	St. dev. (m)	0.0631	0.0946	0.0950
Up	Mean (m)	0.0032	0.0081	0.0083
	St. dev. (m)	0.0717	0.0892	0.0895

availability at each epoch is shown in grey.

The convergence period can be seen at the beginning of the dataset. The three solutions are generally coincident with the exception of two periods: between 12:00 and 14:00 and at the end of the dataset. For the entirety of the first of these periods the positions calculated with bias detection using the AR(1) and AR(p) whitened test statistic give the same solution. However, for the latter period all three methods give different solutions. The mean and standard deviation of the time series were calculated using data after 02:00 to allow for convergence of the solution and are

Figure 7.21: Local ENU positions for GUAM using the unwhitened local slippage test statistic, AR(1) whitened local slippage test statistic and AR(p) local slippage test statistic with the level of significance set at 99%. In all cases the post-fit residuals were used to calculate the test statistics. The mean and standard deviation for each plot are shown in the table below.
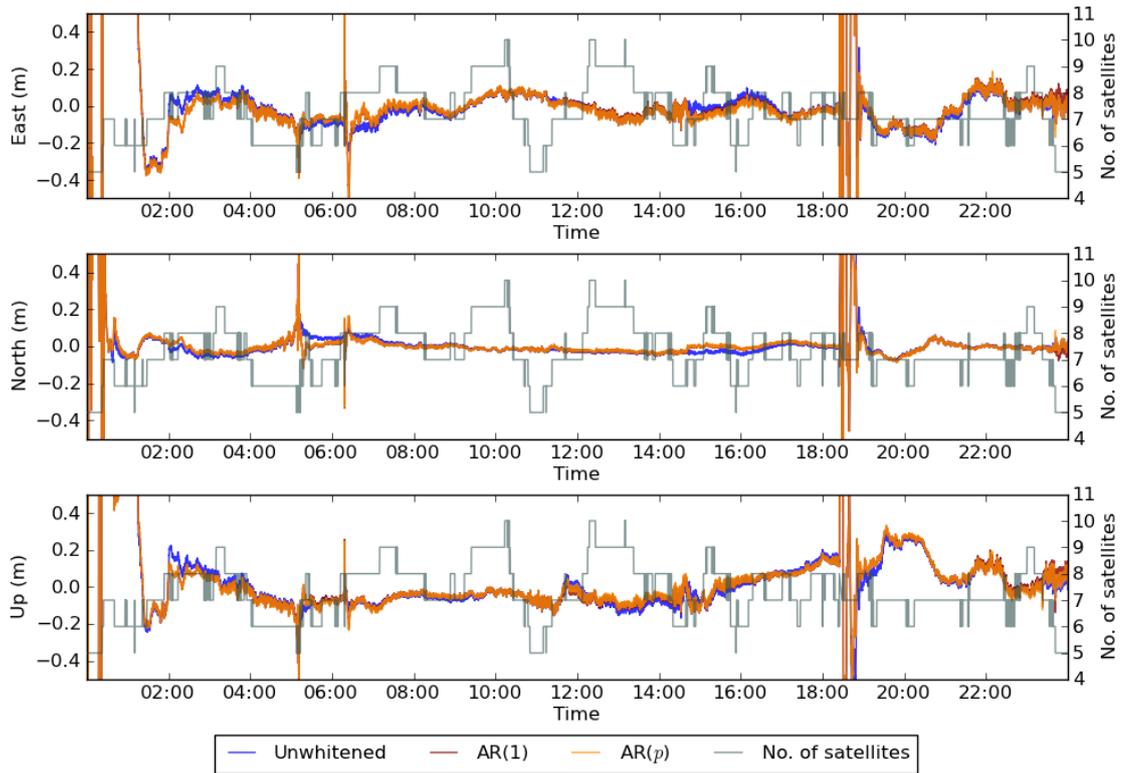
		Unwhitened	AR(1)	AR(p)
East	Mean (m)	0.0191	0.0161	0.0123
	St. dev. (m)	0.2523	0.2293	0.2294
North	Mean (m)	0.0116	0.0114	0.0125
	St. dev. (m)	0.1056	0.0934	0.0933
Up	Mean (m)	-0.0186	-0.0118	-0.0129
	St. dev. (m)	0.1544	0.1494	0.1492

shown in the table in Figure 7.19. It can be seen that whilst the AR(1) solution has a mean closest to zero for the east component, the unwhitened solution is closest to zero for the north and up components. In each case the standard deviation is smallest for the unwhitened solution and greatest for the AR(p) solution, although the difference is at the millimetre level.

The local ENU positions for the FALK dataset are shown in Figure 7.20. It is immediately clear from this plot that the whitened test statistic does not perform as well as the unwhitened test statistic. Although the AR(1) and AR(p) solutions are

Figure 7.22: Difference in local ENU positions relative to the double-difference 'truth' solution for ALBM using the unwhitened local slippage test statistic, AR(1) whitened local slippage test statistic and AR(p) local slippage test statistic with the level of significance set at 99%. In all cases the post-fit residuals were used to calculate the test statistics. The mean and standard deviation for each plot are shown in the table below.

		Unwhitened	AR(1)	AR(p)
East	Mean	-0.0013	-0.0012	-0.0005
	St. dev.	0.0130	0.0135	0.0135
North	Mean	-0.0096	-0.0096	-0.0057
	St. dev.	0.0323	0.0324	0.0313
Up	Mean	0.0272	0.0259	0.0266
	St. dev.	0.0374	0.0377	0.0412

both coincident for the entire dataset there are many points at which the solution deviates from the unwhitened solution. The most drastic of these can be seen at around 17:00, where the two solutions using the whitened test statistic break down completely. Examining the satellite availability, also shown on Figure 7.20, it is clear that this period coincides with the point at which the number of satellites drops to 4. By inspecting the behaviour of the whitened test statistic it is apparent that a number of observables have been identified as containing biases at this particular moment where the number of satellites required is at its minimum. This explains

the cause for the severe degradation of the solution and the comparatively high standard deviations of the AR(1) and AR(p) solutions.

Another noticeable deviation in the solutions occurs at around 03:00. Once again referring to the satellite availability reveals that this deviation occurs at the point at which the number of available satellites changes. Recalling the problem seen in Figure 7.17 whereby the change in the geometry of the least squares solution results in a 'jump' in the residual time series, it would appear that a similar phenomenon is occuring at these points where the number of satellites in the solution changes. When a satellite rises or sets its elevation dependent weighting results in a gradual change to the geometry. However, where a satellite comes in and out of view for other reasons, perhaps because it is obscured, then this will cause a sudden change in geometry. As such the problem seen in Figure 7.17 can be used to explain these deviations in the solution and the solution given earlier can also be used to rectify this problem.

The local ENU positions for the GUAM dataset are shown in Figure 7.21. In this dataset the whitened test statistic also contains a number of spikes that are the result of the 'jump' in the residual time series caused by a change in the least squares geometry. These can be seen around 05:00, 06:00 and just before 15:00. The solution also breaks down significantly between 18:00 and 19:00, although this occurs for the solutions using the original test statistic as well as the AR(1) and AR(p) test statistics. This suggests that there is a problem in the dataset and this is not a result of using a whitened test statistic. However, it is noticeable that for all coordinate components the whitened test statistic has a smaller standard deviation compared to the unwhitened test statistic. The AR(1) and AR(p) solutions also appear to stabilise at 02:00 quicker than the unwhitened solution. Excepting the very end of the dataset, the AR(1) and AR(p) solutions appear to be coincident.

In order to make positional comparisons for the ALBM dataset a difference was made between the east, north and up components for the three PPP solutions and those from a baseline solution, which was taken as 'truth'. The local ENU differences for the ALBM dataset are shown in Figure 7.22. It can be seen that this dataset

is the most stable of the four datasets for the AR(1) and AR(p) unwhitened test statistics. For all coordinate components the mean values for the solutions computed using the whitened test statistics are closer to zero than those using the unwhitened test statistic. There appears to be disagreement between the AR(1) and AR(p) solutions at around 09:30 and a small jump in the AR(1) solution at 11:30. Despite the mean values being closer to zero, the standard deviation of the result is higher for the AR(1) and AR(p) solutions in most cases.

7.8 Effect on reliability measures

In §4.7 the MDB was given as a measure of internal reliability. Recalling Eq. (4.31), the MDB using predicted residuals is given by

$$|\nabla_k| = \sqrt{\frac{\lambda(\alpha, q, \gamma)}{c_{v_k}^T Q_{v_k}^{-1} c_{v_k}}}$$

Since the MDB is a function of the variance-covariance matrix of predicted residuals, when using whitened residuals it follows that

$$|\widetilde{\nabla}_k| = \sqrt{\frac{\lambda(\alpha, q, \gamma)}{c_{\tilde{v}_k}^T Q_{\tilde{v}_k}^{-1} c_{\tilde{v}_k}}} \tag{7.31}$$

where the variance-covariance matrix of whitened predicted residuals $Q_{\tilde{v}_k}$ is used.

The variance of the AR(1) whitened predicted residual was shown in §7.3 to be

$$Q_{\tilde{v}_k} = Q_{v_k} - \Phi_1 \Phi_1^T Q_{v_k} - Q_{v_k} \Phi_1 \Phi_1^T + \Phi_1 Q_{v_k} \Phi_1^T \tag{7.32}$$

where Φ_1 is a diagonal matrix of AR(1) coefficients. By taking c_{v_k} as a vector of null values with a 1 in the place corresponding to the ith observable,

$$\sigma_{\tilde{v}_{k,i}}^2 = \sigma_{\hat{v}_{k,i}}^2 (1 - \phi_{1,i}^2) \tag{7.33}$$

and the MDB for the ith observable when assuming an AR(1) process is given by

$$|\widetilde{\nabla}_{k,i}| = \sqrt{\frac{\lambda(\alpha, q, \gamma)}{[Q_{\tilde{v}_k}^{-1}]_{ii}}} \tag{7.34}$$

Therefore as $\phi_{1,i} \to 1$ then $|\widetilde{\nabla}_i| \to 0$, hence the MDB will decrease as the AR(1) coefficient gets closer to unity.

Similarly, the MDB using post-fit residuals, given by Eq. (4.29), reads

$$|\nabla_k| = \sqrt{\frac{\lambda(\alpha, q, \gamma)}{c^T Q_{y_k}^{-1} Q_{\hat{e}_k} Q_{y_k}^{-1} c}}$$

In this case $Q_{\tilde{e}_k}$ and $\widetilde{Q}_{y_k}$ are inserted to give

$$|\widetilde{\nabla}_k| = \sqrt{\frac{\lambda(\alpha, q, \gamma)}{c^T \widetilde{Q}_{y_k} Q_{\tilde{e}_k} \widetilde{Q}_{y_k} c}} \tag{7.35}$$

with

$$Q_{\tilde{e}_k} = Q_{\hat{e}_k} - \Phi_1 \Phi_1^T Q_{\hat{e}_k} - Q_{\hat{e}_k} \Phi_1 \Phi_1^T + \Phi_1 Q_{\hat{e}_k} \Phi_1^T \tag{7.36}$$

$$\widetilde{Q}_{y_k} = Q_{y_k} - \Phi_1 \Phi_1^T Q_{y_k} - Q_{y_k} \Phi_1 \Phi_1^T + \Phi_1 Q_{y_k} \Phi_1^T \tag{7.37}$$

for an AR(1) process.

By taking c as a vector of null values with a 1 in the place corresponding to the ith observable, the MDB for the ith observable when assuming an AR(1) process is given by (cf. Eq. 4.30)

$$|\widetilde{\nabla}_{k,i}| = \widetilde{\sigma}_{y_{k,i}}^2 \sqrt{\frac{\lambda(\alpha, q, \gamma)}{\sigma_{\tilde{e}_{k,i}}^2}} \tag{7.38}$$

$$= (1 - \phi_{1,i}^2) \sigma_{y_{k,i}}^2 \sqrt{\frac{\lambda(\alpha, q, \gamma)}{(1 - \phi_{1,i}^2) \sigma_{\hat{e}_{k,i}}^2}} \tag{7.39}$$

$$= \sqrt{(1 - \phi_{1,i}^2)} \sigma_{y_{k,i}}^2 \sqrt{\frac{\lambda(\alpha, q, \gamma)}{\sigma_{\hat{e}_{k,i}}^2}} \tag{7.40}$$

with

$$\sigma^2_{\tilde{e}_{k,i}} = \sigma^2_{\hat{e}_{k,i}}(1 - \phi^2_{1,i}) \qquad (7.41)$$

$$\widetilde{\sigma}^2_{y_{k,i}} = \sigma^2_{y_{k,i}}(1 - \phi^2_{1,i}) \qquad (7.42)$$

Once again it is evident that as $\phi_{1,i} \to 1$ then $|\widetilde{\nabla}_{k,i}| \to 0$, hence the MDB will decrease as the AR(1) coefficient gets closer to unity.

The AR(1) and AR(p) whitened MDBs were calculated for the ABPO, FALK, GUAM and ALBM datasets and are shown in Figures 7.23 - 7.27. In each case the unwhitened MDB is shown alongside for comparison. Given the relationship between the L1 and L2 carrier phase residuals (see §5.3.2) only the L1 carrier phase MDBs are shown.

It can be seen from the difference plots in Figure 7.23 that the whitened MDB is smaller than the unwhitened MDB, for the reasons outlined earlier, and its values also no longer change smoothly over time as seen with the unwhitened MDB. The changeability of the AR coefficients have a high influence on the whitened variance of the post-fit residual.

There is a marked difference in the shape of the whitened MDBs compared with their unwhitened equivalents, see for example Figures 7.23h and 7.23g. From this it can be seen that where the residuals are more correlated, not only is the size of the whitened MDB smaller, but the effect of the variability in the AR coefficients is also greater.

Comparing the AR(1) and AR(p) whitened MDB values similar observations to those made in §7.3.4 can be made. A greater number of spikes can be seen in the AR(p) time series in Figure 7.1b compared to the AR(1) time series. This can be explained by the variation in the AR order selected, as shown by Figure 7.24. The vertical axis for AR order has been reversed to highlight the jumps seen in both the AR order and the AR(p) MDB values. Since this variation has caused the variance of the whitened residual to be smaller, the size of the whitened MDB is likewise smaller.

In conclusion, by calculating a whitened MDB based on the variance of the

Figure 7.23: Minimal detectable biases for GPS SVN05 at ABPO calculated using unwhitened, AR(1) and AR(p) residuals. The difference plots show the whitened MDB subtracted from the original MDB.
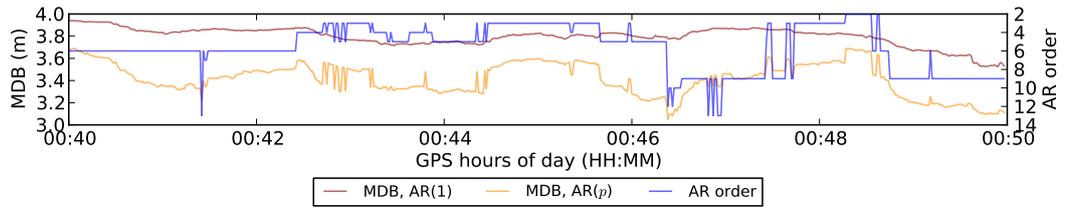
Figure 7.24: Comparison of AR(1) and AR(p) whitened MDB values for GPS SVN05 at ABPO with respect to the AR order chosen.

whitened post-fit residual results in smaller MDBs. This suggests that in a non-optimal filter where the residuals can be modelled as an AR(p) process, whether $p = 1$ or determined using the AIC criterion, the resulting non-optimal MDB that is calculated may be an overestimate of the true value. It is important to note, as discussed in §4.7, that the MDB is an indication of the internal reliability and therefore does not give any measure relative to the estimated parameters. This can be achieved by calculating the minimal detectable effect (see §4.7).

7.9 Discussion

In this section the implementation of the whitened test statistic and the whitened MDB will be discussed with respect to the user requirements outlined in §2.6 and the extent to which the improved power of the whitened test statistic has been demonstrated.

When presenting the whitened test statistic as an improved method of statistically testing GNSS observations it was not the intention that the accuracy would be improved. However, effective quality control of GNSS observations can ensure that accuracy is not degraded by model errors. It was mentioned in §2.6 that in the offshore industry developments in the positioning technology leads to the opening up of new applications. As such, any improvements in accuracy could potentially lead to new markets being exploited.

The reliability of a solution is very important in offshore survey, as remarked upon in §2.6. It has been seen that using the whitened residual variance to calculate the MDB results in smaller values, therefore suggesting that the whitening procedure allows smaller biases masked by time correlation to be detected. For off-
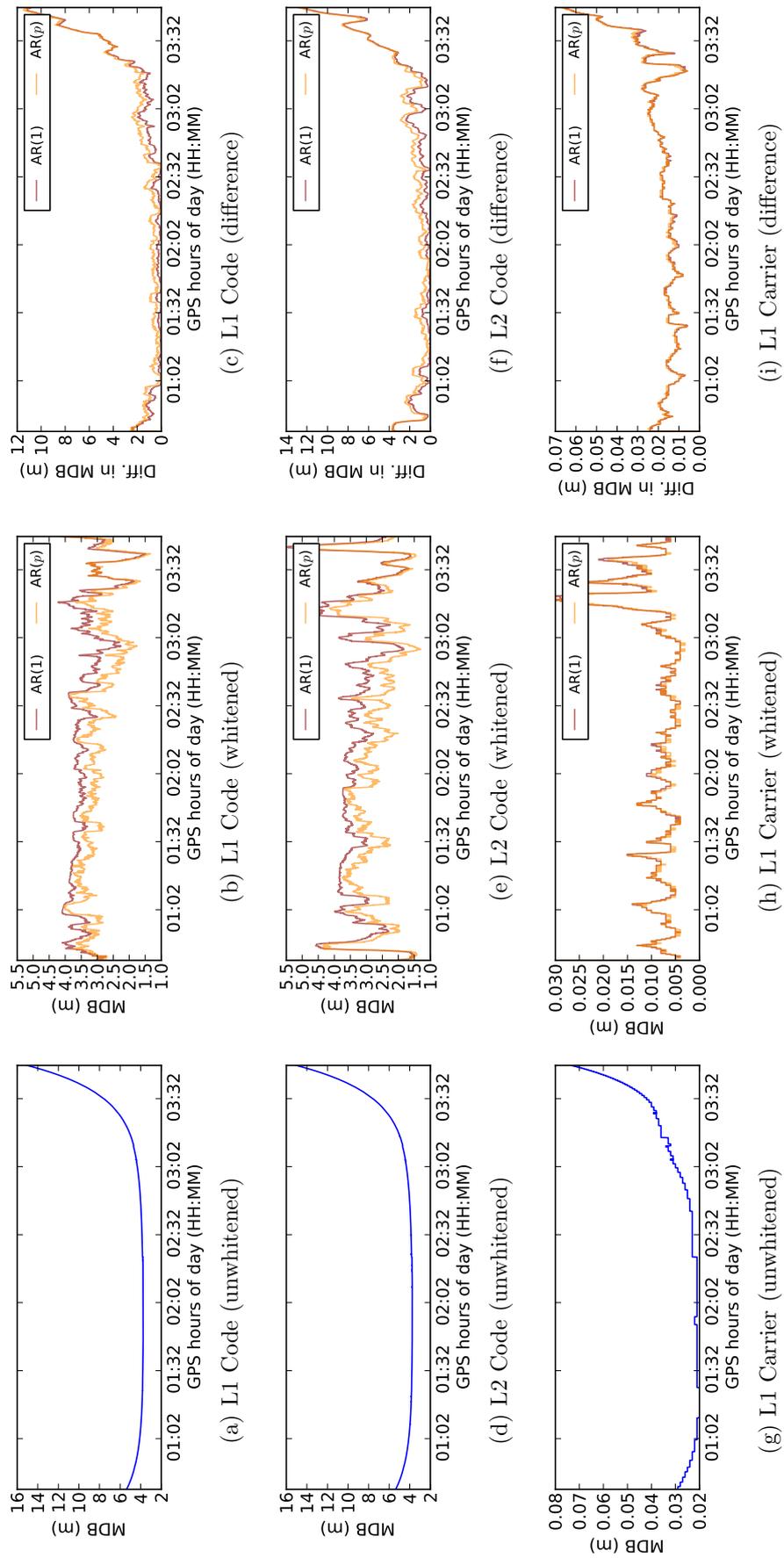
Figure 7.25: Minimal detectable biases for GPS SVN05 at FALK calculated using unwhitened, AR(1) and AR(p) residuals. The difference plots show the whitened MDB subtracted from the original MDB.
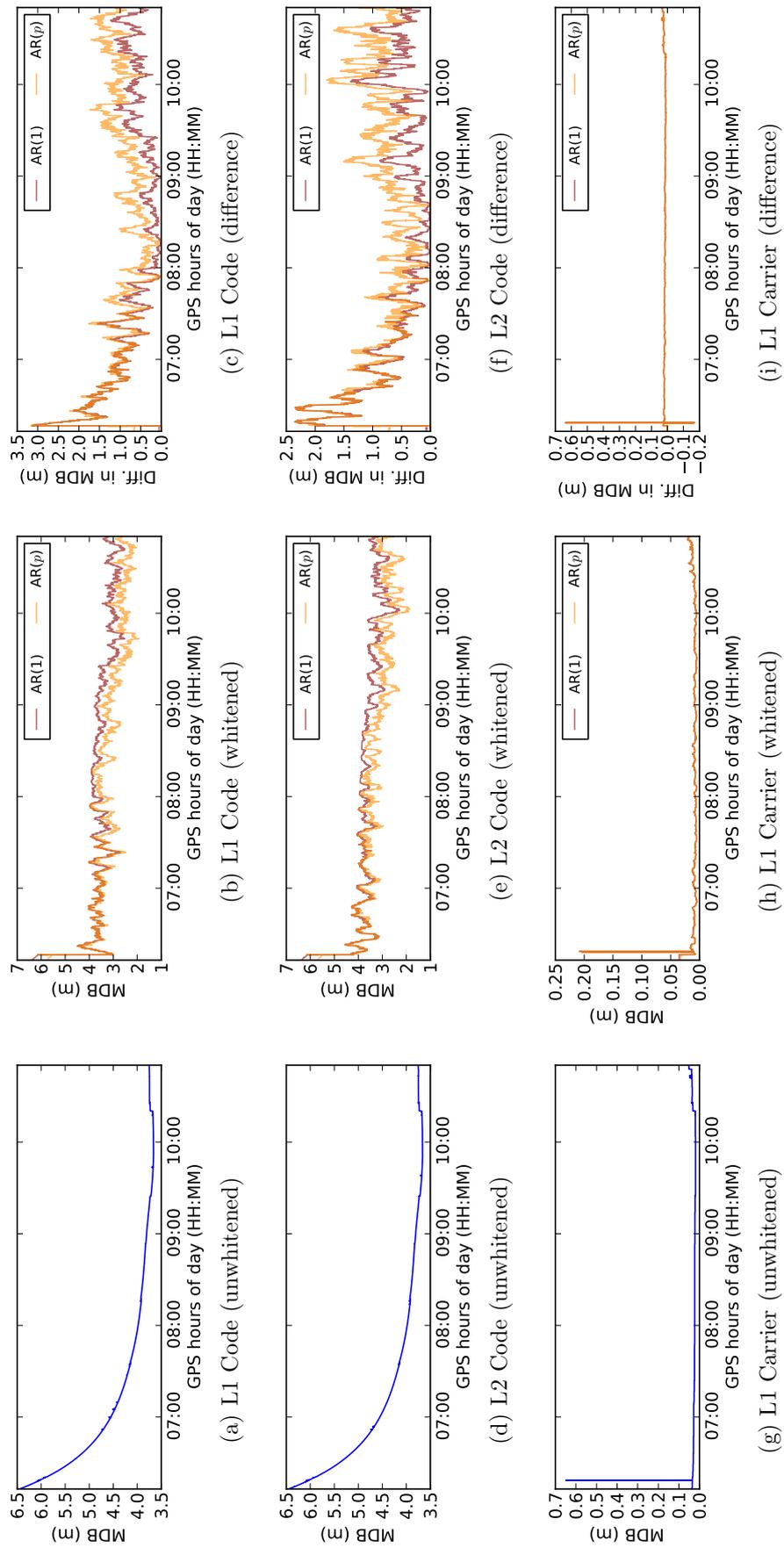
Figure 7.26: Minimal detectable biases for GPS SVN05 at GUAM calculated using unwhitened, AR(1) and AR(p) residuals. The difference plots show the whitened MDB subtracted from the original MDB.
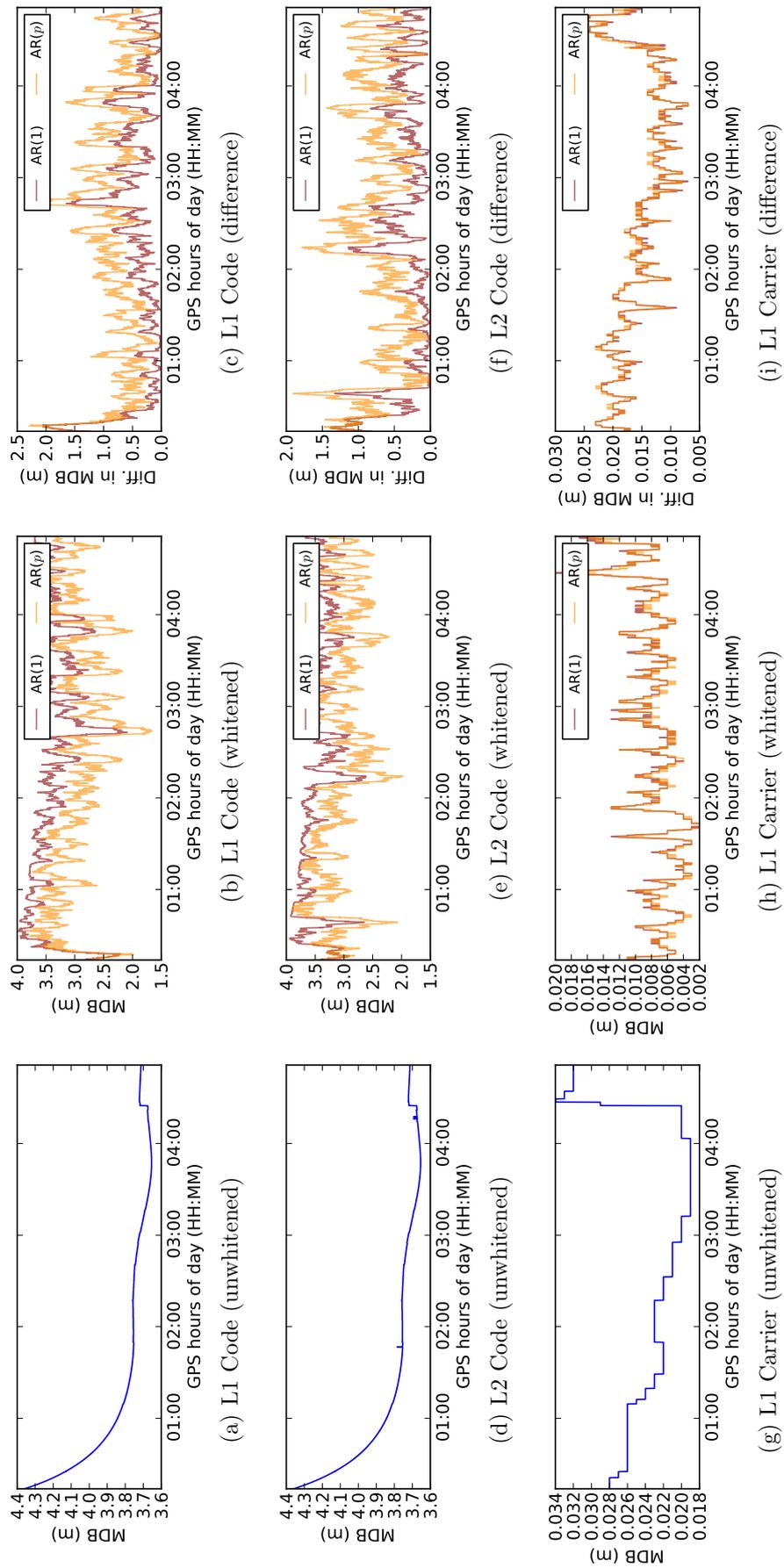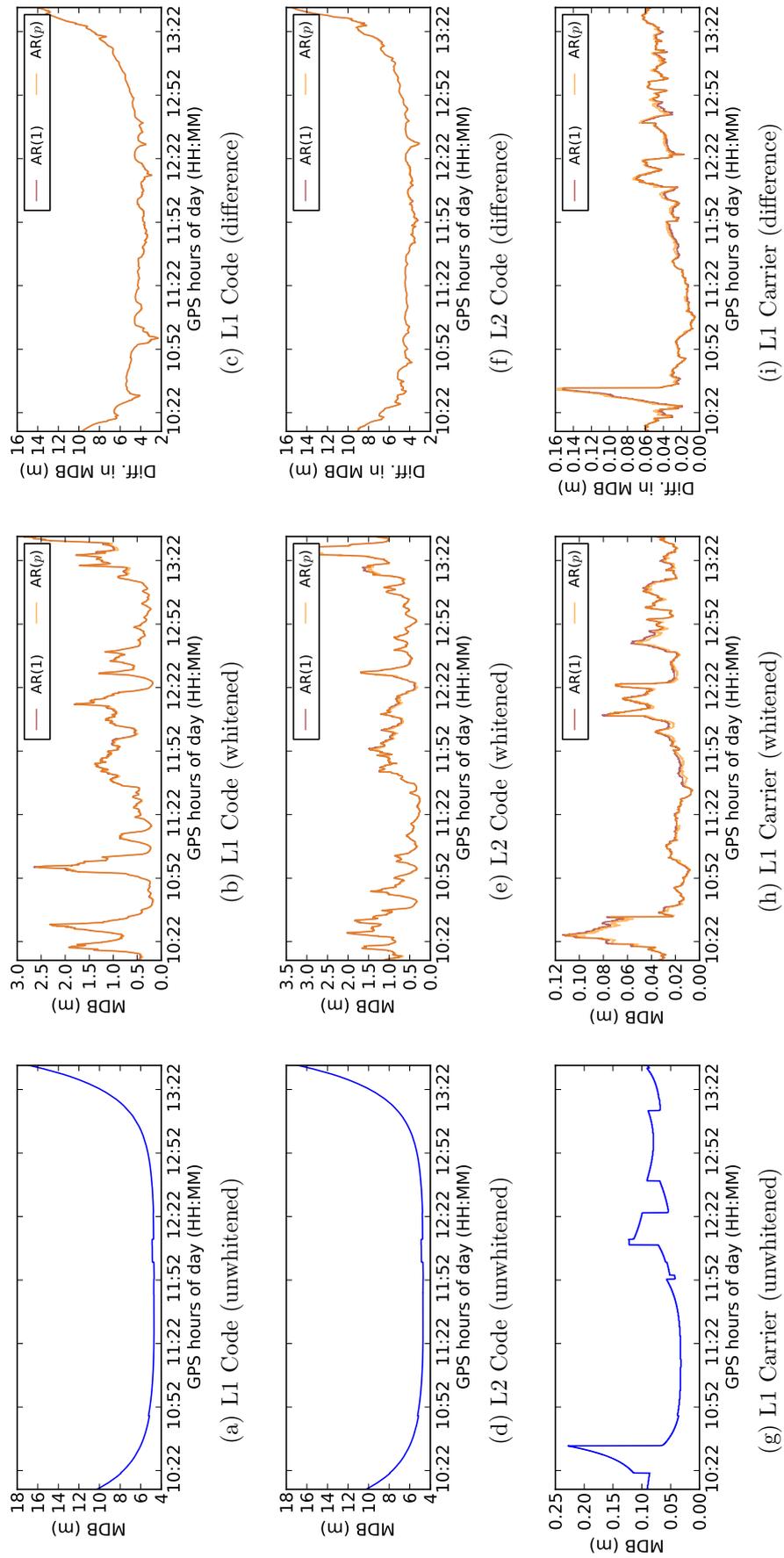
Figure 7.27: Minimal detectable biases for GPS SVN17 at ALBM calculated using unwhitened, AR(1) and AR(p) residuals. The difference plots show the whitened MDB subtracted from the original MDB.

shore positioning this allows extra assurance to be given to the client that a reliable position can be achieved.

One problem that has been addressed when calculating the whitened test statistic is that a sufficiently large sample size of historical residuals is required in order to calculate a reliable estimate of the time correlation. The time period used in this study was 240 seconds, as determined in §5.3.3. This can have a potentially negative impact on the availability of positioning if there are any missing observations and may result in the 99.8% availability over 30 days not being achieved. The problem can be partially overcome using a bridging method, as discussed in §7.5.2. If this method is applied then any availability problem is confined to when a satellite is first tracked.

Additionally, it may be possible to use a shorter period of time to estimate the AR coefficients when there is insufficient data. While the advantage of this is that the period of time to wait before the whitened test statistic can be used is reduced, it may result in a poor estimate of the autocorrelation. Further work would be required to find the shortest possible time whereby the whitened test statistic still performs adequately.

If the whitened test statistic was to be implemented in a real-time system the time correlation would be estimated at the receiver. In this sense the latency issues that affect the orbit and clock corrections supplied to the user would not apply. Any time correlation caused by the latency of the received corrections would manifest themselves in the residuals, and hence still be corrected for.

The results in this chapter show that by using the whitened residual and the variance of the whitened residual, both calculated by modelling the time correlated residuals as an AR process, it is possible to form the whitened test statistic. The magnitude of the whitened test statistic values was shown to be of the same order as the unwhitened test statistic, but displayed less evidence of time correlation. A number of implementation issues were encountered, such as problems caused by gaps in the data and changes in the geometry of the least squares solution (see §7.5.2 and §7.7). Despite this it was shown that the whitened test statistic was effective at

detecting outliers in the dataset.

Some of the implementation issues meant that the effectiveness of the whitened test statistic at detecting outliers could not be fully realised. If these could be overcome it would useful to be able to test the algorithm using a variety of datasets, including challenging datasets obtained in the field as well as data with synthetic outliers.

It was also evident from §7.8 that removal of the time correlation in the residuals resulted in a smaller MDB, suggesting that smaller biases in the observables could be detected. This seems an intuitive assumption to make, since time correlation can lead to over-estimates of the variance and can potentially mask any outlying observables.

Any future work should evaluate the size of bias that can estimated when taking into account time correlation in the residuals. An investigation using synthetic GNSS data could be performed, whereby outliers of varying size in both the code and carrier observations can be inserted. It is also possible to artificially introduce time correlation into the data that is used during processing. Not only could this applied to the data itself, for example by modelling multipath, but also in the orbit and clock corrections.

7.10   Summary

The whitened test statistic, based on the T test statistic given in Chapter 4, was derived in §7.4 using the whitened residuals introduced in Chapter 6 and the variance of the whitened residuals, derivations for which were given in §7.3. The test statistic for the predicted and post-fit residuals were given, although only analysis of the test statistic calculated using post-fit residuals was provided in §7.6 and §7.7 since the use of the semi-recursive Kalman filter in the E-HP/PPP processing software means that only post-fit residuals can be calculated (see §3.6).

From the time series analysis in §7.6 it could be seen that the whitened test statistic was more consistently centred around zero and was more characteristic of white noise than the unwhitened residual, although the magnitude in some case was

greater and there were more frequent spikes. In some cases the estimated whitened test statistic values exceeded the critical value threshold where the unwhitened test statistic did not. In §7.7 individual epochs were considered, in contrast to viewing the test statistic as a time series. This showed that where a large observation bias occurred the two test statistics yielded the same candidates as outliers, although this could only be reliably done where every observable was whitened.

A result of the AR estimation method is that a sample of 240 seconds of continuous residuals is required in order to estimate the AR coefficients. This was highlighted in §7.5, along with other implementation issues. To overcome the problem of gaps in the data, a relaxation of the requirement for continuous data was made whereby gaps of 5 seconds or less could be bridged by assuming continuity. The most important implementation issue was the case where a whitened test statistic cannot be calculated for all observables. It is not possible to use a mixture of unwhitened and whitened test statistic values as this may result in the wrong observable being identified as containing a bias where a model error has occurred.

Using the aforementioned assumptions the whitened test statistic was implemented in the software to see how it would perform in production software. This highlighted a number of issues that would require complex changes to the way in which the software was programmed. It was found that the removal of a biased observable could have a significant influence on the geometry of the least squares solution and hence result in a jump in the residual time series. This gap resulted in the whitened residual to be incorrectly calculated at epoch k, since the residual at epoch $k-1$ now included a bias. This problem was also seen where there was an unexpected change in the number of available satellites, such as when a satellite was rising or setting as its influence was limited by elevation dependent weighting. This problem can be overcome by re-estimating the residuals at the epochs required for whitening and using these values to generate the whitened residual.

The effect of the whitened test statistic on the measurement of reliability was also considered. Since the whitened test statistic is calculated using the variance of the whitened residual, it is important to see the effect of this variance on the

MDB. The whitened MDB of the whitened test statistic using both predicted and post-fit residuals was derived using their respective variances. The effect of the AR coefficients in both was shown to be that the size of the whitened MDB was influenced by the magnitude of correlation. It could also be seen that if the AR coefficients were zero then this gave the unwhitened MDB. The effect of the AR coefficients on the whitened MDB was to make the values much less smooth than that of the unwhitened MDB, depending on the variability of the AR coefficients themselves.

Chapter 8

Conclusions and future directions

8.1 Introduction

This study set out to determine the extent of time correlation in the residuals from PPP processing and establish a statistical test for identifying biases in the observables that can be performed when the residuals are time correlated. This was done using the E-HP/PPP processing software, which uses the semi-recursive Kalman filter described in §3.6. Pre-existing software was chosen over developing PPP processing software from scratch as this can be very time consuming and would limit the extent of research that could be done in the time frame of the project. Since using the semi-recursive Kalman filter results in a state vector that contains common and epoch parameters, for which the latter no a priori information is assumed, all analysis was done using the post-fit residual. In Chapter 4 the quality control procedures for both the predicted and post-fit residual in relation to the so-called DIA method were given. Since the E-HP/PPP processing software uses the post-fit residual all statistical testing of observables is done using the post-fit residual and hence use the formulae given in §4.3. Despite this, the linear relationship between the predicted and post-fit residual given in §3.7 means that it is possible to replicate all the analyses given in Chapters 5, 6 and 7 using the predicted residual where an extended Kalman filter is employed.

8.2   Time correlation in the post-fit residuals

An objective of the research was to establish to what extent time correlation was present in the PPP technique. The theoretical derivations in Chapter 5 showed that, when it is assumed the observables are not correlated in time, whereas in an optimal Kalman filter the predicted and post-fit residuals are not temporally correlated, for a non-optimal Kalman filter this may not be the case. By processing data from four datasets using the E-HP/PPP processing software it was shown that there was strong time correlation in both the code and carrier phase residuals. In addition to this the cross-correlation between satellites and observation types was also high. The level of correlation appeared to be dependent on the satellite, observation type and dataset. In the case of the latter this could be dependent on the site location or receiver type.

An optimal filter assumes that the measurement noise and the process noise are correctly specified and not correlated in time. The stochastic model of the observables is well-known, although it is often assumed there is no correlation between the different observation types and frequencies. As this is commonly implemented in GNSS software packages this assumption was held for this study. It is less trivial to determine the process noise that should be applied to the parameters as part of the time update stage of the filter. Another potential source of correlation is unmodelled parameters. As discussed in §3.8 the measurement model for PPP processing is reparameterised in order to make the system solvable, therefore it is not always possible to model all parameters. Attempts were made in §5.4 to see if the correlation could be reduced by adjusting the process noise. Although the correlation could not be significantly reduced, this method was successful in finding the best value for the process noise that minimised the time correlation. This was found to be in agreement with the default value used in the E-HP/PPP software and hence validated the process noise that was being applied.

Since the AR process is an estimation it was important to establish the best sample size to use in order to achieve this. From an implementation point of view it is necessary to have a large enough sample size to get a true representation of

the AR process, whilst minimising the time that must elapse before an estimate can be made. As implementation of any developed procedure was one of the research objectives, the storage of data to make such an estimate was also a consideration. The sample size of 240 seconds was found to be the point at which there was no extra gain in increasing sample size.

8.3   Modelling post-fit residuals as an AR process

As a step towards achieving the objective of finding a method for removing the time correlation present, the post-fit residuals were modelled as an autoregressive (AR) process in Chapter 6. Following a review of AR estimation methods, Burg's method was chosen to estimate the AR coefficients. This was chosen so that the process could be guaranteed to be stationary, since it had been found that the AR process was close to the unit circle (see §6.2). To not choose such a method could have resulted in an AR process with an exponentially growing error.

From a theoretical standpoint it was shown in §6.6 that an AR(1) process was possible. The time correlation was therefore initially estimated as an AR(1) process and the coefficients were used to obtain 'whitened' estimates. It was shown that an AR(1) process was effective at removing a large proportion of the time correlation from the residuals, resulting in a residual that was random. However, in some cases 'overwhitening' was found to have taken place. By using a higher order AR process, referred to in the thesis as an AR(p) process, it was possible to obtain an improved whitened residual that was closer to random noise.

8.4   Development of the whitened test statistic

In working towards a test statistic that could be used for time correlated residuals, the quality control procedures for GNSS processing that were presented earlier in the thesis were revisited in Chapter 7. Since the T test statistic is formed on the assumption that the residuals are not correlated in time, the whitened residual was considered a good basis by which observation biases could be tested. It was

251

found that by estimating the variance of the whitened residual it was possible to form a whitened T test statistic that satisfied the requirement of the residual to be uncorrelated in time. Viewing this whitened test statistic as a time series resulted in values that were of a comparable magnitude to the unwhitened test statistic, but more characteristic of white noise.

8.5 Implementation of the whitened test statistic

A further objective of the research was to address the implementation of such a test statistic in the E-HP/PPP processing software. From this arose the problem that using time lagged estimates of the post-fit residual to form the whitened post-fit residual resulted in incorrect estimates when the geometry of the estimation changed. This occurred when an observable was suddenly added to or removed from the vector of observables. In the case where a satellite rises or sets, the elevation dependent stochastic model limited the impact of the additional observables. However, in the case where an observable with a high weight is introduced or removed, for example due to being obscured or removed due to an observation bias, the geometry of estimation changes and the residual time series is no longer continuous. Using pre-existing software meant that adapting the PPP processing software to recalculate previous residual estimates, as suggested in §7.7, proved too complex to be able to implement in the given time frame.

8.6 Effect on reliability measures

In addition to detecting any biases in the observable it is also important to consider the reliability of the test statistic being used. The minimal detectable bias (MDB) is a function of the variance of the predicted residual. As it was shown in §7.3 the variance of the whitened residual is found by scaling the estimated variance of the unwhitened residual. As such, when using the whitened test statistic it follows that the whitened MDB should be used as a result. It was shown, both theoretically and in practice, that the whitened MDB decreased as the temporal correlation in the

residuals increased. By considering that time correlation in the residuals has the potential to mask outliers from detection then this is not unexpected. As the temporal correlation in the residuals decreases then the whitened MDB tends towards the unwhitened MDB.

8.7   Future directions

One of the objectives of this research was to investigate the implementation of any developed test statistic in production software. This was partially achieved, although it was clear that for the test statistic to be robustly implemented would require a significant re-write of the existing software. If a PPP processing software package was to be written from scratch then the experiences of implementation in this study could be used in the design of any future software so that the whitened test statistic could be much easily integrated.

In this study an AR process was estimated for each satellite and observation type. As such, the cross-correlations between different satellites and observation types were not taken into account in the estimation of the AR coefficients. The reason for this was that the use of the post-fit residuals resulted in a covariance matrix at lag 0 that could not be inverted. Should the procedures outlined in this thesis be applied to PPP processing software that uses a conventional Kalman filter then it is theoretically possible to use a multivariate AR process to estimate the AR coefficients. This would bring added complications, since the AR order would have to be fixed for all satellites and observation types. Since the AR processes seen in this thesis are very close to the unit circle, an AR estimation method that guarantees stationarity would also be required. Multivariate AR processes become increasingly complex as the order increases, therefore this could be a limiting factor.

As seen in Chapter 6 the estimate of the AR coefficient maintains a fairly constant mean over longer periods of time, but over shorter periods of time can be quite noisy. This has a direct impact on the estimates for the whitened residual and its variance. Whilst this does not have a significant impact on the whitened test statistic, since it is a ratio and is itself characterised by random noise, it results in

a noisy estimate of the variance of the whitened residual. This has a direct impact on the estimates of the whitened MDB, which in §7.8 is seen to be much noisier than the unwhitened MDB. By making the assumption that the AR coefficients do not change significantly from one epoch to the next, the AR coefficients could be determined using a recursive estimator, such as a Kalman filter. This would have the additional benefit that where there is a significant gap in data from a satellite the previous estimates of AR values could be used to forecast any time correlation until there is enough data to generate a reliable estimate. However, this would be challenging to implement if the AR order was allowed to change between epochs.

The effects of the precise orbits and clocks on time correlation in PPP processing requires further study. Since they are treated as known in the PPP measurement model it is important to model their residual effects. One area of further study would be to treat the precise orbits and clocks stochastically. It is also important to note that corrections for one satellite are applied equally over all observation types. This will inevitably result in time correlation and additional cross-correlation between observation types and is in need of further study. With the addition of other satellite systems, such as GLONASS, Galileo and BeiDou, the possibility of inter-system correlation is something that should also be investigated.

For the purposes of this study the assumption that GNSS observables are uncorrelated was held, since this is a practical decision commonly made in GNSS processing software. Many studies have shown this not to be the case. Any future work could integrate the estimates of correlations in observables into the a priori variance-covariance matrix of the observables. This can be done using variance-component estimation aided with estimates of time correlation (Satirapod 2001). Alternatively it has been shown by Wang et al. (2012) that it is possible to estimate the correlation in the observables as part of the state vector. Due to the importance of orbit and clock corrections in PPP and the issues associated with non-optimal filters demonstrated in this thesis it is still likely that time correlations would be present, but reduced.

Finally, there have been many advances in the estimation of integer ambiguities

in PPP with the use of uncalibrated phase delays (UPDs) (Laurichesse and Mercier 2007, Ge et al. 2008, Geng et al. 2010). As seen with orbit and clock corrections, the stochastic nature of the UPDs should be considered. Depending on the method by which the UPDs are applied, this may result in another source of time correlation. With such high rate of data being used this may exacerbate the effect of any temporal correlation.

References

Abbott, A. S. and W. E. Lillo (2003). Global positioning systems and inertial measuring unit ultratight coupling method. US Patent US6516021 B1. Filed 14 Sep 1999 and published 4 Feb 2003.

Akaike, H. (1970). Statistical predictor information. *Annals of the Institute of Statistical Mathematics 22*, 203–217.

Akaike, H. (1974). A new look at the statistical model identification. *Automatic Control, IEEE Transactions on 19*, 716–723.

Amiri-Simkooei, A. (2007). *Least-Squares Variance Component Estimation Theory and GPS Applications*. Netherlands Geodetic Commission.

Amiri-Simkooei, A. R., P. J. G. Teunissen, and C. C. J. M. Tiberius (2009). Application of Least-Squares Variance Component Estimation to GPS Observables. *Journal of Engineering Surveying 135*(4), 149–160.

Anquela, A., A. Martín, J. Berné, and J. Padín (2013). Gps and glonass static and kinematic ppp results. *Journal of Surveying Engineering 139*(1), 47–58.

Baarda, W. (1968). A testing procedure for use in geodetic networks. Technical Report 5, Netherlands Geodetic Commission.

Banville, S. and R. B. Langley (2009). Improving Real-Time Kinematic PPP with Instantaneous Cycle-Slip Correction. In *Proceedings of ION GNSS 2009*, Savannah, GA.

Baselga, S. (2011). Non Existence of Rigorous Tests for Multiple Outlier Detection in Least Squares Adjustment. *Journal of Surveying Engineering 137*(3), 109–112.

Bischoff, W., B. Heck, J. Howind, and A. Teusch (2006). A procedure for estimating the variance function of linear models and for checking the appropriateness of estimated variances: a case study of GPS carrier-phase observations. *Journal of Geodesy 79*, 694–704.

Bisnath, S. and Y. Gao (2008). Current State of Precise Point Positioning and Future Prospects and Limitations. In *Proceedings of IUGG 24th General Assembly*.

Blewitt, G. (1989). Carrier phase ambiguity resolution for the Global Positioning System applied to geodetic baselines up to 2000 km. *Journal of Geophysical Research 94* (B8), 10–187.

Blewitt, G. (1998). *GPS Data Processing Methodology: From Theory to Applications* (2nd ed.)., Chapter 6, pp. 231–270. In Teunissen and Kleusberg Teunissen and Kleusberg (1998a).

Boehm, J., B. Werl, and H. Schuh (2006). Troposphere mapping functions for GPS and very long baseline interferometry from European Centre for Medium-Range Weather Forecasts operational analysis data. *Journal of Geophysical Research 111*, B02406.

Bona, P. (2000). Precision, Cross Correlation, and Time Correlation of GPS Phase and Code Observations. *GPS Solutions 4* (2), 3–13.

Borre, K. and C. Tiberius (2000). Time Series Analysis of GPS Observables. In *Proceedings of the 13th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2000)*, Salt Lake City, UT, pp. 1885–1894.

Box, G. E. and G. M. Jenkins (1976). *Time series analysis : forecasting and control*. San Francisco: Holden-Day.

Broersen, P. M. T. (2006). *Automatic Autocorrelation and Spectral Analysis*. Springer.

Brown, R. G. and P. Y. C. Hwang (1996). *Introduction to Random Signals and Applied Kalman Filtering* (3rd ed.). Wiley.

Burg, J. P. (1967). Maximum entropy spectral analysis. In *37th Annual International Meeting*. Society of Exploration Geophysicists.

Cai, C. and Y. Gao (2007). Precise Point Positioning Using Combined GPS and GLONASS Observations. *Journal of Global Positioning Systems 6*(1), 13–22.

Cai, C. and Y. Gao (2013). Modeling and assessment of combined GPS/GLONASS precise point positioning. *GPS Solutions 17*, 223–236.

Carpenter, J., P. Clifford, and P. Fearnhead (1999). Improved particle filter for nonlinear problems. *Radar, Sonar and Navigation, IEE Proceedings - 146*(1), 2–7.

Chen, J., H. Li, B. Wu, Y. Zhang, J. Wang, and C. Hu (2013). Performance of Real-Time Precise Point Positioning. *Marine Geodesy 36*, 98–108.

Chen, K. and Y. Gao (2005, September). Real-Time Precise Point Positioning Using Single Frequency Data. In *Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation*, Long Beach, CA, pp. 1514–1523.

Chen, X., T. Allison, W. Cao, K. Ferguson, S. Gr unig, V. Gomez, A. Kipka, J. K ohler, H. Landau, R. Leandro, G. Lu, R. Stolz, and N. Talbot (2011). Trimble RTX, an Innovative New Approach for Network RTK. In *Proceedings of the Institute Of Navigation (ION) GNSS 2011 Meeting.*

Clarke, P. J. and N. T. Penna (2010). Ocean tide loading and relative GNSS in the British Isles. *Survey Review 42*, 212–228.

Cross, P. A. (1994). *Advanced least squares applied to position fixing* (2nd ed.). University of East London. Working Paper No. 6.

Cummins, C. (2012). Critical Values for the Durbin-Watson Test. Available at http://www.stanford.edu/~clint/bench/dwcrit.htm. Accessed 23/08/2012.

de Jong, C. D. (2008a). E-HP – Models. Internal report, Fugro Intersite B.V., Leidschendam, The Netherlands.

de Jong, C. D. (2008b). Least squares recursive estimator. Internal report, Fugro Intersite B.V., Leidschendam, The Netherlands.

de Jong, C. D. (2013). Smoothing code. Internal report, Fugro Intersite B.V., Leidschendam, The Netherlands.

de Jong, C. D., G. Lachapelle, S. Skone, and I. Elema (2002). *Hydrography*, Chapter 3. Delft University Press.

de Jonge, P. J. (1998). *A processing strategy for the application of the GPS in networks.* Ph. D. thesis, Faculty of Civil Engineering and Geosciences, Delft University.

Demmel, J. W. (1997). *Applied Numerical Linear Algebra.* Society for Industrial and Applied Mathematics.

Dougherty, C. (2011). *Introduction to Econometrics.* Oxford University Press.

Durbin, J. and G. M. Watson (1950). Testing for serial correlation in least squares regression. I. *Biometrika 37*(3/4), 409–428.

Durbin, J. and G. M. Watson (1951). Testing for serial correlation in least squares regression. II. *Biometrika 38*, 159–178.

El Rabanny, A. E.-S. (1994). *The effect of physical correlations on the ambiguity resolution and accuracy estimation in GPS differential positioning.* Ph. D. thesis, Geodesy and Geomatics Engineering, University of New Brunswick.

Elmas, Z. G., M. Aquino, H. A. Marques, and J. F. G. Monico (2011). Higher order ionospheric effects in GNSS positioning in the European region. *Annales Geophysicae 29*, 1383–1399.

Elsobeiey, M. and A. El-Rabbany (2012). On Modelling of Second-Order Ionospheric Delay for GPS Precise Point Positioning. *Journal of Navigation 65*(01), 59–72.

Ernstsen, V., R. Noormets, D. Hebbeln, A. Bartholomä, and B. Flemming (2006). Precision of high-resolution multibeam echo sounding coupled with high-accuracy positioning in a shallow water coastal environment. *Geo-Marine Letters 26*, 141–149.

Farebrother, R. W. (1980). Algorithm AS 153: Pan's Procedure for the Tail Prob-

abilities of the Durbin-Watson Statistic. *Journal of the Royal Statistical Society 29*(2), 224–227.

Fox, L. (1964). *An Introduction to Numerical Linear Algebra*. Oxford University Press.

Gao, Y. and K. Chen (2004). Performance Analysis of Precise Point Positioning Using Real-Time Orbit and Clock Products. *Journal of Global Positioning Systems 3*, 95–100.

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*, 389–399.

Geng, J., X. Meng, A. Dodson, M. Ge, and F. Teferle (2010). Rapid re-convergences to ambiguity-fixed solutions in precise point positioning. *Journal of Geodesy 84*, 705–714. 10.1007/s00190-010-0404-4.

Genrich, J. F. and Y. Bock (2006). Instantaneous geodetic positioning with $10 - 50$ Hz GPS measurements: Noise characteristics and implications for monitoring networks. *Journal of Geophysical Research 111*(B3), B03403.

Gibbs, B. P. (2011). *Advanced Kalman Filtering, Least-Squares and Modeling: A Practical Handbook*. John Wiley & Sons.

Groves, P. (2008). *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Boston: Artech House.

Harvey, A. C. (1994). *Applications of the Kalman filter in econometrics*, Chapter 8, pp. 285–321. Cambridge University Press.

Haykin, S. (2002). *Kalman Filtering and Neural Networks*. John Wiley & Sons.

Heng, L., G. Gao, T. Walter, and P. Enge (2011). Statistical characterization of gps signal-in-space errors. In *Proceedings of the 2011 International Technical Meeting of The Institute of Navigation*, San Diego, CA, pp. 312–319.

Hesselbarth, A. and L. Wanninger (2008). Short-term Stability of GNSS Satellite Clocks and its Effects on Precise Point Positioning. In *Proceedings of the ION 2008 Meeting*, Savannah, Georgia.

Hewitson, S. and J. Wang (2006). GNSS Receiver Autonomous Integrity Monitoring (RAIM) for Multiple Outliers. *The European Journal of Navigation 4*(4), 47–57.

Hofmann-Wellenhof, B., H. Lichtenegger, and E. Wasle (2008). *GNSS–Global Navigation Satellite Systems: GPS, GLONASS, Galileo, and more.* Springer-Verlag.

Hopfield, H. (1969). Two-quartic tropospheric refractivity profile for correcting satellite data. *Journal of Geophysical Research 74*(18), 4487–4499.

Hu, C., W. Chen, Y. Chen, and D. Liu (2003). Adaptive Kalman filtering for vehicle navigation. *Journal of Global Positioning Systems 2*(1), 42–47.

Huber, P. J. (1981). *Robust Statistics.* John Wiley & Sons.

IERS (2011). IERS Annual Report 2011. Technical report, International Earth Rotation and Reference Systems Service.

IMCA/OGP (2011). *Guidelines for GNSS positioning in the oil & gas industry.* London, United Kingdom: International Marine Contractors Association (IMCA) and International Association of Oil & Gas Producers (OGP).

IMO (2001). Revised maritime policy and requirements for a future global navigation satellite system (GNSS). IMO Resolution A.915(22). Technical report, International Maritime Association.

Jokinen, A., S. Feng, W. Ochieng, C. Hide, T. Moore, and C. Hill (2012). Fixed ambiguity Precise Point Positioning (PPP) with FDE RAIM. In *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, pp. 643–658. IEEE.

Julier, S. J. and J. K. Uhlmann (1997). A new extension of the Kalman filter to nonlinear systems. In *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, Volume 3, pp. 26.

Juran, J. M. and A. B. Godfrey (1999). *Juran's Quality Handbook.* New York: McGraw Hill.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering 82*(1), 35–45.

Kaplan, E. and C. Hegarty (2006). *Understanding GPS: Principles And Applications.* Artech House Mobile Communications Series. Artech House.

Karabatic, A., R. Weber, K. Huber, C. Abart, F. Heuberger, P. Berglez, J. Seybold, and C. Klug (2010). Concepts to limit convergence time of gps-based precise point positioning (ppp). In *Proceedings of the 23rd International Technical Meeting of The Satellite Division of the Institute of Navigation.*

Kim, D. and R. B. Langley (2001). Quality Control Techniques and Issues in GPS Applications: Stochastic Modeling and Reliability Test. In *International Symposium on GPS/GNSS, Jeju Island.*

Kjørsvik, N. and E. Brøste (2009). Using TerraPOS for efficient and accurate marine positioning. Technical report, TerraTec AS.

Kleijer, F. (2004). *Troposphere Modeling and Filtering for Precise GPS Leveling.* Number 56. Delft: Nederlandse Commissie voor Geodesie.

Klobuchar, J. (1987). Ionospheric time-delay algorithm for single-frequency GPS users. *Aerospace and Electronic Systems, IEEE Transactions on* (3), 325–331.

Knight, N. L. and J. Wang (2009). A Comparison of Outlier Detection Procedures and Robust Estimation Methods in GPS Positioning. *Journal of Navigation 62*, 699–709.

Knight, N. L., J. Wang, and C. Rizos (2009). Generalised Measures of Reliability for Multiple Outliers. *JGR.*

Koch, K. R. (1999). *Parameter Estimation and Hypothesis Testing in Linear Models* (2nd ed.). Springer-Verlag.

Kouba, J. (2009). A guide to using International GPS Service (IGS) products. Available at http://igscb.jpl.nasa.gov/components/usage.html. Accessed 06/06/2012.

Kouba, J. and P. Héroux (2001). GPS Precise Point Positioning Using IGS Orbit Products. *GPS Solutions 5*(2), 12–28.

Krakiwsky, E., D. Szabo, P. Vanícek, and M. Craymer (1999). Development and testing of in-context confidence regions for geodetic survey networks. Technical report, Geodetic Survey Division, Geomatics Canada.

Krakiwsky, E. J. (1975). A Synthesis of Recent Advances in the Method of Least Squares. In *Geodesy & Geomatics Lecture Notes*, Number 42. University of New Brunswick.

Lachapelle, G. and M. Petovello (2006). What is precise point positioning (PPP), and what are its requirements, advantages and challenges? *Inside GNSS*, 16–21.

Langley, R. B. (1997). GPS Receiver System Noise. *GPS World 8*(6), 40–45.

Langley, R. B. (1998). *Propogation of the GPS signals* (2nd ed.)., Chapter 3, pp. 111–150. In Teunissen and Kleusberg Teunissen and Kleusberg (1998a).

Laurichesse, D. and F. Mercier (2007). Integer ambiguity resolution on undifferenced GPS phase measurements and its application to PPP. In *Proceedings of ION GNSS 2007*.

Le, A. and P. Teunissen (2006, May). Recursive Least-Squares Filtering of Pseudorange Measurements. In *Proceedings of the European Navigation Conference (ENC GNSS 2006)*, Manchester, UK. ECN.

Le, A., C. Tiberius, H. Marel, and N. Jakowski (2008). Use of global and regional ionosphere maps for single-frequency precise point positioning. In M. Sideris (Ed.), *Observing our Changing Earth*, Volume 133 of *International Association of Geodesy Symposia*, pp. 759–769. Springer Berlin Heidelberg.

Leandro, R., M. Santos, and R. Langley (2011). Analyzing GNSS data in precise point positioning software. *GPS Solutions 15*, 1–13. 10.1007/s10291-010-0173-9.

Leick, A. (2004). *GPS Satellite Surveying* (3rd ed.). John Wiley & Sons.

Leondos, C. T. (Ed.) (1970). *Theory and Applications of Kalman Filtering*. Neuilly sur Seine, France: NATO Advisory Group for Aerospace Research & Development.

Lyard, F., F. Lefevre, T. Letellier, and O. Francis (2006). Modelling the global ocean tides: modern insights from FES2004. *Ocean Dynamics 56*, 394–415.

Marini, J. W. (1972). Correction of Satellite Tracking Data for an Arbitrary Tropospheric Profile. *Radio Sci. 7*(2), 223–231.

Matsumoto, K., T. Takanezawa, and M. Ooe (2000). Ocean Tide Models Developed by Assimilating TOPEX/POSEIDON Altimeter Data into Hydrodynamical Model: A Global Model and a Regional Model around Japan. *Journal of Oceanography 56*, 567–581.

Meditch, J. (1973). A survey of data smoothing for linear and nonlinear dynamic systems. *Automatica 9*(2), 151–162.

Mehra, R. K. (1970). On the Identification of Variances and Adaptive Kalman Filtering. *IEEE Transactions on Automatic Control 15*(2), 175–184.

Meindl, M., S. Schaer, U. Hugentobler, and G. Beutler (2004). Tropospheric gradient estimation at code: Results from global solutions. *Journal of the Meteorological Society of Japan 82/1B*, 331–338.

Melgard, T., E. Vigen, K. de Jong, L. D., V. H., and O. O. (2009). G2 – The First Real-Time GPS and Glonass Precise Orbit and Clock Service. In *Proceedings of The Institute of Navigation GNSS 2009 International Technical Meeting*, Savannah, USA.

Milbert, D. (2012). Solid earth tides. Available at http://home.comcast.net/~dmilbert/softs/solid.htm. Accessed on 16/06/2012.

Miller, C., K. O'Keefe, and Y. Gao (2011). Time Correlation in GNSS Positioning over Short Baselines. *Journal of Surveying Engineering*.

Mohamed, A. H. and K. P. Schwarz (1999). Adaptive Kalman Filtering for INS/GPS. *Journal of Geodesy 73*, 193–203.

Monteiro, L. S., T. Moore, and C. Hill (2005). What is the accuracy of DGPS? *Journal of Navigation 58*, 207–225.

Muellerschoen, R., A. Reichert, D. Kuang, M. Heflin, W. Bertiger, and Y. Bar-Sever (2001). Orbit determination with NASA's high accuracy real-time Global Differential GPS System. In *Proceedings of ION GPS 2001*, pp. 2294–2303.

Neill, A. (1996). Global mapping functions for the atmosphere delay at radio wavelengths. *Journal of Geophysical Research 101*(B2), 3227–3246.

Odijk, D. (2000). Weighting Ionospheric Corrections to Improve Fast GPS Positioning Over Medium Distances. In *Proceedings of ION GPS 2000*, Salt Lake City, UT.

Olynik, M. C. (2002). *Temporal Characteristics of GPS Error Sources and Their Impact on Relative Positioning*. Masters thesis, Department of Geomatics Engineering, University of Calgary.

Oussalah, M. and J. D. Schutter (2001). Adaptive Kalman filter for noise identification. In *Proceedings of the International Seminar on Modal Analysis*, Volume 3, pp. 1225–1232.

Pearson, J. and E. Stear (1974). Kalman Filter Applications in Airborne Radar Tracking. *IEEE Transactions on Aerospace and Electronic Systems AES-10*(3), 319–329.

Petit, G. and B. Luzum (Eds.) (2010). *IERS Conventions*. Frankfurt am Main: Verlag des Bundesamts für Kartographie und Geodäsie.

Petovello, M. (2011). Does the Magnitude of the GNSS Receiver Clock Offset Matter? *Inside GNSS 6*(2), 23–25.

Priestley, M. B. (1981). *Spectral Analysis and Time Series*. Academic Press Limited.

Píriz, R., D. Calle, A. Mozo, P. Navarro, D. Rodríguez, and G. Tobías (2009). Orbits and Clocks for GLONASS Precise Point Positioning. In *Proceedings of ION GNSS 2009*, Savannah, GA.

Ray, R. D. (1999). A global ocean tide model from TOPEX/POSEIDON altimetry: GOT99.2. Technical memorandum TM-209478, NASA.

Rizos, C., V. Janssen, C. Roberts, and T. Grinter (2012, May). Precise Point Positioning: Is the Era of Differential GNSS Positioning Drawing to an End? In *Proceedings of FIG Working Week 2012*, Rome, Italy.

Rousseeuw, P. J. and A. M. Leroy (1987). *Robust regression and outlier detection*. New York: John Wiley & Sons.

Ryan, T. P. (2007). *Modern Engineering Statistics*. John Wiley & Sons.

Saastamoinen, J. (1973). Contributions to the theory of atmospheric refraction. *Bulletin Géodésique 107*, 13–34.

Sahmoudi, M. and R. J. Landry (2008, November/December). Multipath mitigation techniques using maximum-likelihood principle. *Inside GNSS*, 24–29.

Salazar, D., M. Hernandez-Pajares, J. Juan, and J. Sanz (2010). GNSS data management and processing with the GPSTk. *GPS Solutions 14*, 293–299.

Salzmann, M. A. (1993). *Least squares filtering and testing for geodetic navigation applications*. Ph. D. thesis, Department of Geodetic Engineering, Delft University of Technology.

Satirapod, C. (2001). Improving the accuracy of static GPS positioning with a new stochastic modelling procedure. In *14th International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GPS 2001*, Salt Lake City, UT, pp. 11–14.

Savin, N. E. and K. J. White (1977). The Durbin-Watson Test for Serial Correlation with Extreme Sample Sizes or Many Regressors. *Econometrica 45*(8), 1986–1996.

Scherneck, H.-G. and M. S. Bos (2002). Ocean Tide Loading and Atmospheric Loading. In *Proceedings of IVS General Meeting*, Tsukuba, Japan, pp. 205–214.

Schmid, R., P. Steigenberger, G. Gendt, M. Ge, and M. Rothacher (2007). Generation of a consistent absolute phase center correction model for GPS receiver and satellite antennas. *Journal of Geodesy 81*, 781–798.

Schön, S. and F. K. Brunner (2008). A proposal for modelling physical correlations of GPS phase observations. *Journal of Geodesy 82*, 601–612.

Sorenson, H. W. (1970). Least-squares estimation: from Gauss to Kalman. *Spectrum, IEEE 7*(7), 63 –68.

Stockwell, D. (2007). *Niche Modeling: Predictions from Statistical Distributions.* Taylor & Francis Group.

Strang, G. and K. Borre (1997). *Linear Algebra, Geodesy, and GPS.* Wellesley-Cambridge Press.

Teunissen, P. J., D. Odijk, and B. Zhang (2010). PPP-RTK: Results of CORS Network-Based PPP with Integer Ambiguity Resolution. *Journal of Aeronautics, Astronautics and Aviation A.42*(4), 223–230.

Teunissen, P. J. G. (1985). *Quality Control in Geodetic Networks*, pp. 526–547. Berlin: Springer-Verlag.

Teunissen, P. J. G. (1990a, September). An Integrity and Quality Control Procedure for Use in Multi Sensor Integration. In *Proceedings of the 3rd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1990)*, Colorado Spring, CO, pp. 513–522.

Teunissen, P. J. G. (1990b). Nonlinear least squares. *Manuscripta Geodaetica 15*(3), 137–150.

Teunissen, P. J. G. (1990c). Quality control in integrated navigation systems. In *Position Location and Navigation Symposium, 1990. Record. The 1990's - A Decade of Excellence in the Navigation Sciences. IEEE PLANS '90., IEEE*, pp. 158–165.

Teunissen, P. J. G. (1991). The GPS phase-adjusted pseudorange. In *Proceedings of the 2nd International Workshop on High Precision Navigation*, Stuttgart/Freudenstadt, pp. 115–125.

Teunissen, P. J. G. (1997). The geometry-free GPS ambiguity search space with a weighted ionosphere. *Journal of Geodesy 71*, 370–383.

Teunissen, P. J. G. (1998a). Minimal detectable biases of GPS data. *Journal of Geodesy 72*, 236–244.

Teunissen, P. J. G. (1998b). *Quality Control for GPS* (2nd ed.)., Chapter 7, pp. 271–318. In Teunissen and Kleusberg Teunissen and Kleusberg (1998a).

Teunissen, P. J. G. (2006). *Adjustment theory: an introduction.* Delft, The Netherlands: Vereniging voor Studie- en Studentenbelangen te Delft.

Teunissen, P. J. G., N. F. Jonkman, and C. C. J. M. Tiberius (1998). Weighting GPS Dual Frequency Observations: Bearing the Cross of Cross-Correlation. *GPS Solutions 2*(2), 28–37.

Teunissen, P. J. G. and A. Kleusberg (Eds.) (1998a). *GPS for Geodesy* (2nd ed.). Springer-Verlag.

Teunissen, P. J. G. and A. Kleusberg (1998b). *GPS Observation Equations and Positioning Concepts* (2nd ed.)., Chapter 5, pp. 187–229. In Teunissen and Kleusberg Teunissen and Kleusberg (1998a).

Tiberius, C. C. J. M. (1998). *Recursive data processing for kinematic GPS surveying.* Ph. D. thesis, Delft University of Technology.

UKOAA (1994). *Guidelines for the use of differential GPS in offshore surveying.* UK Offshore Operators Association Surveying and Positioning Committee.

van Bree, R. J. P. and C. C. J. M. Tiberius (2012). Real-time single-frequency precise point positioning: accuracy assessment. *GPS Solutions 16*, 259–266.

Wang, J., C. Satirapod, and C. Rizos (2002). Stochastic assessment of GPS carrier phase measurements for precise static relative positioning. *Journal of Geodesy 76*, 95–104.

Wang, J. and J. Wang (2007). Mitigating the Effect of Multiple Outliers on GNSS Navigation with M-Estimation Schemes. In *Proceedings of the IGNSS Symposium 2007.* International Global Navigation Satellite Systems Society.

Wang, K., Y. Li, and C. Rizos (2012). Practical Approaches to Kalman Filtering with Time-Correlated Measurement Errors. *IEEE Transactions on Aerospace and Electronic Systems 48*(2), 1669–1681.

Ward, P. W., J. W. Betz, and C. J. Hegarty (2006). *Satellite Signal Acquisition, Tracking, and Data Demodulation*, Chapter 5, pp. 153–241. Volume 6 of *Artech House Mobile Communications Series* Kaplan and Hegarty (2006).

Wells, D. E., W. Lindlohr, B. Schafferin, and E. Graferend (1987). GPS design: undifferenced carrier beat phase observations and the fundamental differencing theorem. Technical report, Department of Surveying, University of New Brunswick.

Wu, J. T., S. C. Wu, G. A. Hajj, W. I. Bertiger, and S. M. Lichten (1993). Effects of antenna orientation on GPS carrier phase. *Manuscripta Geodaetica 18*, 91–98.

Wuebbena, G., M. Schmitz, and A. Bagge (2005). PPP-RTK: Precise Point Positioning using state-space representation in RTK networks. In *Proceedings of ION GNSS 2005*.

Xiang Jin, X. and K. D. de Jong (1996). Relationship Between Satellite Elevation and Precision of GPS Code Observations. *The Journal of Navigation 49*(02), 253–265.

Yang, L., J. Wang, N. Knight, and Y. Shen (2013). Outlier separability analysis with a multiple alternative hypotheses test. *Journal of Geodesy*, 1–14.

Yang, Y. and W. Gao (2005). Comparison of adaptive factors in Kalman filters on navigation results. *Journal of Navigation 58*(03), 471–478.

Yang, Y., H. He, and G. Xu (2001). Adaptively robust filtering for kinematic geodetic positioning. *Journal of Geodesy 75*(2–3), 109–116.

Yule, G. U. (1927). On a Method of Investigating Periodicities in Disturbed Series, with Special Reference to Wolfer's Sunspot Numbers. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character 226*, 267–298.

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 102*(B3), 5005–5017.

Appendix A

Summary of mathematical notation

Introduction

The mathematical notation in this thesis is derived from both estimation theory, closely following the style of Salzmann (1993) and Teunissen (1998b), and from time series analysis, mixing notation by Box and Jenkins (1976), Priestley (1981) and Broersen (2006). As such, this appendix gives a reference to the notation used grouped by subject area.

Time

In some cases various vectors and matrices are time-dependent. A particular epoch will be denoted in subscript by the letters k and l. For example, $\underline{y}_k$ refers to the vector of observables at time k. The previous epoch would similarly be denoted as $k-1$, i.e. $\underline{y}_{k-1}$.

Vectors and matrices

In general, vectors are represented in lower-case and matrices are represented in upper-case. In the case of a diagonal matrix for the sake of clarity off-diagonal zero

elements will not be shown, e.g.

$$\begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix} \Rightarrow \begin{pmatrix} a & & \\ & b & \\ & & c \end{pmatrix}$$

In some cases this may be further generalised to

$$\mathrm{diag}\,(a, b, c)$$

GNSS observation models

p	pseudorange code observation.
ϕ	carrier phase observation.
R	geometric distance between the receiver and satellite.
δt	clock biases.
T	tropospheric delay.
I	ionospheric delay.
η	code hardware delay.
μ	carrier phase hardware delay.
λ	carrier wavelength.
N	carrier phase ambiguity, integer part.
φ	carrier phase ambiguity, decimal part.

Statistical theory

$D\{\cdot\}$	mathematical dispersion operator, which represents the theoretical variance of the variable contained within the parentheses.
$E\{\cdot\}$	mathematical expectation operator, which represents the theoretical mean of the variable contained within the parentheses.

Estimation theory

A random variable, whose definition can be found in Koch (1999), is denoted by an underline, e.g. $\underline{y}$ and a hat is used to denote an estimated value, e.g. $\underline{\hat{x}}_k$ is the estimate of the random variable $\underline{x}$ at epoch k. It is often the case that a value at one epoch is dependent on a value from a different epoch. This is denoted by a vertical line, e.g. $x_{k|k-1}$ indicates vector x at epoch k based on the solution from epoch $k-1$.

$\underline{y}$	random vector of m observables.
Q_y	an $m \times m$ a priori variance-covariance matrix of the observables.
$\underline{x}$	vector of n unknown parameters.
$\underline{\hat{x}}$	estimate of n unknown parameters.
Q_x	an $n \times n$ a posterior variance-covariance matrix of the estimated parameters.
A	an $m \times n$ design matrix.
$\hat{e}$	a vector of m least squares residuals.
B	an $m \times (m-1)$ matrix of condition coefficients.

Recursive estimation theory

$\Phi_{k,k-1}$	the state transition matrix from epoch k to $k-1$.
$\underline{w}$	a vector of process noise.
$\underline{v}$	a vector of predicted residuals.
Q_v	an a posterior variance-covariance matrix of predicted residuals.
K	the Kalman gain matrix.

Hypothesis testing

α	the level of significance

β the power of test

H_0 the null hypothesis

H_A the alternative hypothesis

C_q a matrix of model errors with q dimensions.

$\underline{\nabla}$ a vector of unknown model errors or explanatory parameters.

λ the non-centrality parameter.

$\underline{T}$ the general form of the test statistic

$\underline{t}$ the one-dimensional form of the test statistic

$\underline{w}$ the scalar one-dimensional slippage test statistic, often called the w-test.

$|\nabla|$ the minimal detectable bias (MDB).

Time series analysis

d the Durbin-Watson test statistic.

d_U^+, d_L^+ the upper and lower bound for the Durbin-Watson test against positive correlation.

d_U^-, d_L^- the upper and lower bound for the Durbin-Watson test against negative correlation.

$R(s)$ the autocovariance of a stochastic process at lag s.

$\rho(s)$ the autocorrelation of a stochastic process at lag s.

$R_{ij}(s)$ the cross-covariance of a stochastic process at lag s.

$\rho_{ij}(s)$ the cross-correlation of a stochastic process at lag s.

Autoregression

$a_1, \ldots, a_p$ the autoregressive (AR) coefficients up to order p for the AR process as defined by Priestley (1981) and Broersen (2006).

$\phi_1, \ldots, \phi_p$ the AR coefficients up to order p for the AR process as defined by Box and Jenkins (1976). This is the notation used in this study.

$\underline{\varepsilon}$ the random noise in an AR process.

$\pi(s)$ the partial autocorrelation function at lag s.

κ the reflection coefficient.

$\tilde{v}_k$ the whitened predicted residual at epoch k.

$\tilde{e}_k$ the whitened post-fit residual at epoch k.

$Q_{\tilde{v}_k}$ the variance-covariance matrix of the whitened predicted residual at epoch k.

$Q_{\tilde{e}_k}$ the variance-covariance matrix of the whitened post-fit residual at epoch k.

$\underline{\tilde{T}}$ the general form of the whitened test statistic.

$\underline{\tilde{t}}$ the one-dimensional form of the whitened test statistic

$\underline{\tilde{w}}$ the scalar one-dimensional whitened slippage test statistic.

$|\tilde{\nabla}|$ the whitened MDB.

Appendix B

Derivation of the test statistic for the post-fit residual in a recursive estimator

B.1 Introduction

By following the least squares derivation of the Kalman filter given in §3.5 it is possible to show that the post-fit residuals may be used for hypothesis testing of the observation using the method presented in §4.3. The least squares model for the filtering step is given by

$$
E\left\{\left(\begin{array}{c} \underline{\hat{x}}_{k|k-1} \\ \underline{y}_k \end{array}\right)\right\} = \left(\begin{array}{c} I \\ A_k \end{array}\right) x_k \quad ; \quad \left(\begin{array}{cc} Q_{\underline{\hat{x}}_{k|k-1}} & \\ & Q_{\underline{y}_k} \end{array}\right) \tag{B.1}
$$

where $\underline{\hat{x}}_{k|k-1}$ and $Q_{\underline{\hat{x}}_{k|k-1}}$ have been computed in the prediction step (see §3.5).

It will be shown here that the local test statistic at epoch k is equivalent to the test statistic used in batch processing.

B.2 Residuals and corresponding variances

The general forms of $\underline{\hat{e}}$ and $Q_{\hat{e}}$ are

$$\underline{\hat{e}} = P_A^\perp y \qquad Q_{\hat{e}} = P_A^\perp Q_y \tag{B.2}$$

where

$$P_A^\perp = I - A(A^T Q_y^{-1} A)^{-1} A^T Q_y^{-1} \tag{B.3}$$

is the orthogonal projector of the matrix A (Teunissen 2006). As $P_A^\perp$ is present in both equations it is best to first substitute into this values from the extended model.

$$
\begin{aligned}
P_A^\perp &= I - \begin{pmatrix} I \\ A_k \end{pmatrix} Q_{\underline{\hat{x}}_{k|k}} \begin{pmatrix} I & A_k^T \end{pmatrix} \begin{pmatrix} Q_{x_{k|k-1}}^{-1} & \\ & Q_{y_k}^{-1} \end{pmatrix} \\
&= \begin{pmatrix} I - Q_{\underline{\hat{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} & -Q_{\underline{\hat{x}}_{k|k}} A_k^T Q_{y_k}^{-1} \\ -A_k Q_{\underline{\hat{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} & I - A_k Q_{\underline{\hat{x}}_{k|k}} A_k^T Q_{y_k}^{-1} \end{pmatrix}
\end{aligned}
\tag{B.4}
$$

It is possible to expand $\underline{\hat{e}}_{k|k}$ using Eq. (B.4) giving

$$
\begin{aligned}
\underline{\hat{e}}_{k|k} &= \begin{pmatrix} I - Q_{\underline{\hat{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} & -Q_{\underline{\hat{x}}_{k|k}} A_k^T Q_{y_k}^{-1} \\ -A_k Q_{\underline{\hat{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} & I - A_k Q_{\underline{\hat{x}}_{k|k}} A_k^T Q_{y_k}^{-1} \end{pmatrix} \begin{pmatrix} x_{k|k-1} \\ y_k \end{pmatrix} \\
&= \begin{pmatrix} -Q_{\underline{\hat{x}}_{k|k}} A_k^T Q_{y_k}^{-1} y_k + (I - Q_{\underline{\hat{x}}_{k|k}} Q_{x_{k|k-1}}^{-1}) x_{k|k-1} \\ (I - A_k Q_{\underline{\hat{x}}_{k|k}} A_k^T Q_{y_k}^{-1}) y_k - A_k Q_{\underline{\hat{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} x_{k|k-1} \end{pmatrix}
\end{aligned}
\tag{B.5}
$$

Based on the assumption that

$$\underline{\hat{e}}_{k|k} = \begin{pmatrix} \underline{\hat{e}}_{x_{k|k-1}} \\ \underline{\hat{e}}_{y_k} \end{pmatrix} \tag{B.6}$$

then

$$\underline{\hat{e}}_{x_{k|k-1}} = -Q_{\underline{\hat{x}}_{k|k}} A_k^T Q_{y_k}^{-1} y_k + (I - Q_{\underline{\hat{x}}_{k|k}} Q_{x_{k|k-1}}^{-1}) x_{k|k-1} \tag{B.7}$$

$$\underline{\hat{e}}_{y_k} = (I - A_k Q_{\underline{\hat{x}}_{k|k}} A_k^T Q_{y_k}^{-1}) y_k - A_k Q_{\underline{\hat{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} x_{k|k-1} \tag{B.8}$$

Similarly it is possible to expand $Q_{\hat{e}_{k|k}}$ using Eq. (B.4) giving

$$
\begin{aligned}
Q_{\hat{e}_{k|k}} &= \begin{pmatrix} I - Q_{\hat{\underline{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} & -Q_{\hat{\underline{x}}_{k|k}} A_k^T Q_{y_k}^{-1} \\ -A_k Q_{\hat{\underline{x}}_{k|k}} Q_{x_{k|k-1}}^{-1} & I - A_k Q_{\hat{\underline{x}}_{k|k}} A_k^T Q_{y_k}^{-1} \end{pmatrix} \begin{pmatrix} Q_{x_{k|k-1}} & \\ & Q_{y_k} \end{pmatrix} \\
&= \begin{pmatrix} Q_{x_{k|k-1}} - Q_{\hat{\underline{x}}_{k|k}} & -Q_{\hat{\underline{x}}_{k|k}} A_k^T \\ -A_k Q_{\hat{\underline{x}}_{k|k}} & Q_{y_k} - A_k Q_{\hat{\underline{x}}_{k|k}} A_k^T \end{pmatrix}
\end{aligned}
\tag{B.9}
$$

B.3   Local overall model test

The overall model test statistic is given in §4.3.1 as

$$
\underline{T}_{m-n} = \frac{\hat{\underline{e}}^T Q_y^{-1} \hat{\underline{e}}}{m - n}
\tag{B.10}
$$

By using the partition of $\hat{e}_{k|k}$ in Eq. (B.6) the local overall model test using the post-fit residuals becomes

$$
\begin{aligned}
\underline{T}_{m-n} &= \frac{1}{m-n} \begin{pmatrix} \hat{\underline{e}}_{x_{k|k-1}}^T & \hat{\underline{e}}_{y_k}^T \end{pmatrix} \begin{pmatrix} Q_{x_{k|k-1}} & \\ & Q_{y_k} \end{pmatrix} \begin{pmatrix} \hat{\underline{e}}_{x_{k|k-1}} \\ \hat{\underline{e}}_{y_k} \end{pmatrix} \\
&= \frac{\hat{\underline{e}}_{x_{k|k-1}}^T Q_{x_{k|k-1}} \hat{\underline{e}}_{x_{k|k-1}}}{m-n} + \frac{\hat{\underline{e}}_{y_k}^T Q_{y_k} \hat{\underline{e}}_{y_k}}{m-n}
\end{aligned}
\tag{B.11}
$$

Since the assumption is that a model misspecification is caused by one of the observations, i.e. an element in $\underline{y}_k$, and that any model errors at the previous epoch have already been identified and removed, Eq. (B.11) can be simplified to

$$
\underline{T}_{m-n} = \frac{\hat{\underline{e}}_{y_k}^T Q_{y_k} \hat{\underline{e}}_{y_k}}{m - n}
\tag{B.12}
$$

B.4   Local slippage test

The slippage test statistic is given by Eq. (4.10) as

$$
\underline{w} = \frac{c^T Q_y^{-1} \hat{\underline{e}}}{\sqrt{c^T Q_y^{-1} Q_{\hat{e}} Q_y^{-1} c}}
\tag{B.13}
$$

By partitioning c into those parts relating to $\underline{y}_k$ and $\underline{\hat{x}}_{k|k-1}$ as per Eq. (B.6) as

$$c = \begin{pmatrix} c_{x_{k|k-1}} \\ c_{y_k} \end{pmatrix} \tag{B.14}$$

and using the partition of $\hat{e}_{k|k}$ in Eq. (B.6) the local slippage test statistic using the post-fit residuals becomes

$$\underline{w}_k = \frac{\begin{pmatrix} c_{x_{k|k-1}}^T & c_{y_k}^T \end{pmatrix} \begin{pmatrix} Q_{x_{k|k-1}}^{-1} & \\ & Q_{y_k}^{-1} \end{pmatrix} \begin{pmatrix} \hat{\underline{e}}_{x_{k|k-1}} \\ \hat{\underline{e}}_{y_k} \end{pmatrix}}{\sqrt{\begin{pmatrix} c_{x_{k|k-1}}^T & c_{y_k}^T \end{pmatrix} \begin{pmatrix} Q_{\hat{x}_{k|k-1}}^{-1} & \\ & Q_{y_k}^{-1} \end{pmatrix} \begin{pmatrix} Q_{x_{k|k-1}} - Q_{\hat{\underline{x}}_{k|k}} & -Q_{\hat{\underline{x}}_{k|k}} A_k^T \\ -A_k Q_{\hat{\underline{x}}_{k|k}} & Q_{y_k} - A_k Q_{\hat{\underline{x}}_{k|k}} A_k^T \end{pmatrix} \begin{pmatrix} Q_{\hat{x}_{k|k-1}}^{-1} & \\ & Q_{y_k}^{-1} \end{pmatrix} \begin{pmatrix} c_{x_{k|k-1}} \\ c_{y_k} \end{pmatrix}}}$$

which reduces to

$$\underline{w}_k = \frac{c_{x_{k|k-1}}^T Q_{x_{k|k-1}}^{-1} \hat{\underline{e}}_{x_{k|k-1}} + c_{y_k}^T Q_{y_k}^{-1} \hat{\underline{e}}_{y_k}}{\sqrt{a+b}} \tag{B.15}$$

where

$$a = c_{x_{k|k-1}}^T Q_{\hat{x}_{k|k-1}}^{-1} (Q_{x_{k|k-1}} - Q_{\hat{\underline{x}}_{k|k}}) Q_{\hat{x}_{k|k-1}}^{-1} c_{x_{k|k-1}} + c_{x_{k|k-1}}^T Q_{\hat{x}_{k|k-1}}^{-1} Q_{\hat{x}_{k|k}} A_k^T Q_{y_k}^{-1} c_{y_k} \tag{B.16}$$

$$b = c_{y_k}^T Q_{y_k}^{-1} (Q_{y_k} - A_k Q_{\hat{\underline{x}}_{k|k}} A_k^T) Q_{y_k}^{-1} c_{y_k} + c_{y_k}^T Q_{y_k}^{-1} A_k Q_{\hat{x}_{k|k}} Q_{\hat{x}_{k|k-1}}^{-1} c_{x_{k|k-1}} \tag{B.17}$$

Since the assumption is that a model misspecification is caused by one of the observations, i.e. an element in $\underline{y}_k$, and that any model errors at the previous epoch have already been identified and removed, $c_{x_{k|k-1}} = 0$ and hence

$$\underline{w}_k = \frac{c_{y_k}^T Q_{y_k}^{-1} \hat{\underline{e}}_{y_k}}{\sqrt{c_{y_k}^T (Q_{y_k}^{-1} - Q_{y_k}^{-1} A_k Q_{\hat{\underline{x}}_{k|k}} A_k^T Q_{y_k}^{-1}) c_{y_k}}} \tag{B.18}$$

B.5 Summary

In the derivation above it was shown that using the post-fit residuals that relate to the observations ($\hat{\underline{e}}_{y_k}$) it can be seen that Eq. (4.9) and Eq. (B.12) are equivalent. It has further been shown that Eq. (4.12) and Eq. (B.18) are also equivalent.