Université Fédérale

Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut Supérieur de l'Aéronautique et de l'Espace

Présentée et soutenue par :
Fabio FABOZZI

le jeudi 3 février 2022

Titre :

Robust GNSS phase tracking using variational bayesian inference

Méthodes de poursuite robuste de phase pour signaux GNSS basées sur l'inférence bayésienne variationnelle

---

École doctorale et discipline ou spécialité :

ED MITT : Signal, image, acoustique et optimisation

Unité de recherche :

Équipe d'accueil ISAE-ONERA SCANR

Directeur(s) de Thèse :

Mme Stéphanie BIDON (directrice de thèse)

Jury :

Mme Audrey GIREMUS Maître de conférences Université de Bordeaux - Présidente
Mme Stéphanie BIDON Professeure ISAE-SUPAERO - Directrice de thèse
M. Antoine BOUCHAIN Chargé d'expertise en radionavigation DGA/MI - Examinateur
M. Serge REBOUL Professeur Université du Littoral Côte d'Opale - Rapporteur
M. Sébastien ROCHE Ingénieur Airbus Defence and Space - Examinateur
M. Laurent ROS Professeur Grenoble-INP - Rapporteur

Thesis funding

The Ph.D. thesis is supported by the DGA (under grant 2018 60 0014) and Airbus Defence and Space (Toulouse, France).

Acknowledgments

First of all, I would like to thank DGA and Airbus D&S for their financial support. Besides, I would thank ISAE-SUPAERO, and in particular the DEOS department, for having warmly "hosted" for these three years.

I would thank the jury for their time and congratulations. In particular, Laurent Ros and Serge Reboul for their valuable work as reviewers; Audrey Giremus for being the jury's president; and, the DGA and Airbus D&S representatives, respectively, Antoine Bouchain and Sébastien Roche, especially the latter which has been present for any technical advice over the years.

Moreover, I express my gratitude to my thesis director, Stéphanie Bidon, for her guidance during the entire journey. Her technical and motivational support has been indispensable to reaching the finish line.

I then thank the adventure companions of office 07.183 and colleagues of the DEOS department for letting me feel a welcomed Rital.

Finally, I would like to express my recognition to my family.

Contents

List of Figures

List of Tables

Acronyms

ADC	*Analog-Digital Converter*
ATAN2	*Four-Quadrant Arctangent*
AWGN	*Additive White Gaussian Noise*
BOC	*Binary Offset Carrier*
BPF	*Radio Frequency Bandpass Filter*
C/A	*Coarse Acquisition*
CDMA	*Code Division Multiple Access*
C/N_0	*Carrier-to-Noise Ratio*
COTS	*Commercial Off-The-Shelf*
CRB	*Cramér–Rao Bound*
DLL	*Delay Locked Loop*
DPLL	*Digital Phase Locked Loop*
DS-SS	*Direct-Sequence Spread-Spectrum*
EKF	*Extended Kalman Filter*
E,P,L	*Early, Prompt, Late*
FLL	*Frequency Lock Loop*
FPGA	*Field Programmable Gate Array*
GNSS	*Global Navigation Satellite System*
GPS	*Global Positioning System*
IF	*Intermediate Frequency*
I,Q	*In-phase, Quadrature*
ITU	*International Telecommunications Union*
KF	*Kalman Filter*
KL	*Kullback-Leibler*
LAMBDA	*Least-Squares AMBiguity Decorrelation Adjustment*

LNA *Low Noise Amplifier*

LOSS *Line-Of-Sight Signal*

MAP *Maximum A Posteriori*

MCAR *Multiple Carrier Ambiguity Resolution*

MCMC *Monte Carlo Markov Chain*

MEDL *Multipath Estimating Delay Lock Loop*

MEO *Medium Earth orbit*

MMSE *Minimum Mean-Square Error*

MRF *Markov Random Field*

MTFS *Mean Time First Slip*

NCO *Numerical Control Oscillator*

P *Precise*

PD *Phase Detector*

PDF *Probability Density Function*

PDI *Pre-Detection Integration*

PF *Particle Filter*

PLI *Phase Lock Indicator*

PLL *Phase Locked Loop*

PPP *Precise Point Positioning*

PRN *Pseudo-Random Noise*

PSD *Power Spectral Density*

PVT *Position, Velocity and Timing*

QAM *Quadrature Amplitude Modulation*

RAIM *Receiver Autonomous Integrity Monitoring*

RF-FE *Radio Frequency Front End*

RMSE *Root Mean Square Error*

RTK *Real-Time Kinematic*

RVB *Restricted Variational Bayesian*

SDR *Software Defined Radio*

SNR *Signal-to-Noise Ratio*

SNV *Signal-to-Noise Variance*

SOC *System-On-Chip*

SoI *Signal-of-Interest*

SSM *State-Space Model*

STOA *STate-Of-the-Art*

TCAR *Three Carrier Ambiguity Resolution*

TTFF *Time To First Fix*

VB *Variational Bayesian*

VSM *Variance Summing Method*

Notations

Signal parameters

s	transmitted signal;
d	navigation message;
c	pseudo-random noise code;
f_c	carrier frequency;
f_d	Doppler frequency;
τ	time delay;
ϕ	carrier phase;
$\overset{n}{\dot{\phi}}$	n-th derivative of the phase;
$\hat{\phi}$	estimated carrier phase;
ϕ_0	initial step phase;
$\dot{\phi}_0$	initial ramp phase;
$\ddot{\phi}_0$	initial parabolic phase;
α	carrier amplitude;
σ_n	noise standard deviation;
T_s	sampling time;
T	integration time;

Processing parameters

M_c	Monte Carlo runs;
SNR	Signal-to-Noise Ratio;

C/N_0 Carrier-to-Noise Ratio;

State-space model

$\boldsymbol{A}$ state matrix;

$\boldsymbol{H}$ measurement matrix (KF-based DPLL);

$\boldsymbol{x}$ state vector;

DPLL

B_L loop bandwidth;

KF-based DPLL

$\boldsymbol{P}$ error covariance matrix;

σ_ϕ standard deviation of the phase;

$\sigma_{\dot{\phi}}$ standard deviation of the first-order phase derivative;

$\sigma_{\ddot{\phi}}$ standard deviation of the second-order phase derivative;

$\boldsymbol{Q}$ noise covariance matrix;

$\boldsymbol{R}$ measurement covariance matrix;

RVB

$q_{\max}$ highest order of Bessel functions;

σ_ϕ standard deviation of the phase;

$\sigma_{\dot{\phi}}$ standard deviation of the first-order phase derivative;

$\sigma_{\ddot{\phi}}$ standard deviation of the second-order phase derivative;

$\boldsymbol{Q}$ noise covariance matrix;

Introduction

Context and motivation

Satellite navigation systems have gone through an impressive development since their emergence. Considerable events or developments are always subject to a couple of differentiators: technological developments and political decisions. The navigation systems are generally referred to as GNSS, short for *Global Navigation Satellite Systems*. First, the *Global Positioning System* (GPS) by the USA opened the application to the mass market, and in recent years, the forthcoming Galileo by Europe renewed the research interest on this topic.

GNSS is used to determine the position of a receiver on land, at sea, or in space by means of the constellation of multiple satellites. Determining receiver position (*i.e.*, latitude, longitude, and height) relies on the estimated distance to several satellites. Each satellite continuously broadcasts a navigation message. The receiver uses the received signal, from satellites in view, to determine the transit time of each message and compute the distance to each satellite. Finally, the receiver position is calculated by using the trilateration method, which is a mathematical calculation to find out the position of something by knowing its distance from a number of known points.

The distance between the satellite and receiver is estimated by two types of measurements: pseudorange (or code) and carrier-phase measurements. The former is a measure of the difference between the receiver clock at signal reception and the satellite clock at signal transmission (scaled by the speed of light). The latter is a measure of the instantaneous beat phase and the accumulated number of zero-crossings obtained after mixing with a reference signal with the nominal frequency. For applications requiring high-positioning accuracy, carrier-phase measurements are typically used due to their high-precision (which is in the cm-range, ≈ 2 orders of magnitude more accurate than the code measurements).

The advantage of carrier-phase measurements comes at the cost of a possible presence of ambiguous phase jumps. In fact, phase measurements obtained by conventional phase tracking algorithms can be severely impaired by these jumps, well-known as cycle slips, which particularly arise in degraded environments. These local losses of lock can even lead the tracking loop to a complete drop-lock from which it never recovers. Since even more realistic applications require precise user positioning and receivers are daily used in challenging environments, robust carrier-phase tracking is needed.

To address this problem, this Ph.D. thesis aims at developing a robust carrier-phase tracking algorithm that may decrease cycle slip occurrence and improve precision in estimation with respect to the existing carrier-phase tracking techniques. Note that, in this thesis, a degraded environment is simulated by using a low signal-to-noise ratio.

Thesis outline

The dissertation consists of five chapters, where the first two are part of the *STate-Of-the-Art* (STOA) analysis and the last three are part of the contribution.

The first chapter describes the GNSS context. In particular, the principle of GNSS is introduced and the signal structure and model (transmitted and received signals) is presented. Afterward, the GNSS receiver architecture is described, going into details of the tracking loops. Finally, we dwell on carrier-phase measurement and the positioning techniques who use it.

The second chapter focuses on carrier-phase tracking. In particular, we introduce the "father" of all the existing techniques, which is the *Digital Phase Lock Loop* (DPLL). We also investigate the cycle slip phenomenon when the nonlinear analysis is presented. Afterward, we make an overview of the robust carrier-phase tracking methods. In particular, we focus our attention on the KF (*Kalman Filter*)-based DPLL, which together with DPLL is used for comparisons with our proposed algorithms in the next chapters.

In the third chapter, we propose a new robust carrier-phase algorithm that aims to decrease cycle slip occurrence. In particular, we start from slow phase dynamics (*i.e.*, first-order loop). The methodology is based on *Variational Bayesian* (VB) inference which, via an approximate solution of the phase tracking problem, has low computational cost and can be applied to large-scale problems (*i.e.*, long-time tracking) compared to stochastic techniques such as Markov chain Monte Carlo. In particular, applying the VB inference and using the *Restricted VB* (RVB) approximation, we are able to obtain a closed-form recursive expression of the estimator [1]. The latter is then compared using a synthetic signal in terms of estimation's accuracy and cycle slipping with previously presented techniques such as DPLL and KF-based DPLL.

For realistic GNSS applications, it is necessary to consider higher order dynamics. Therefore, the fourth chapter is about the RVB algorithm at any order [2], where not only the phase but even its derivatives are estimated. The methodology follows the same path as described in the third chapter, however the complexity of the state-space model is now augmented. Though we found that the recursive RVB problem still has a closed-formed expression with a similar functional form as that encountered at first order, and characterized by an intrinsic nonlinear behavior. The latter is compared using a synthetic signal in terms of estimation's accuracy and cycle slipping with DPLL and KF-based DPLL at third order.

In the last chapter, we implement the proposed RVB algorithm into a GNSS *Software Defined Radio* (SDR) receiver. The latter has been previously developed by our university lab, and its architecture is recalled. From an initial phase tracking loop based on a third-order DPLL, the GNSS SDR is then modified by adding the RVB algorithm in one configuration and substituting the DPLL with KF-based DPLL in another. Afterwards, the proposed RVB algorithm is compared with third-order DPLL and KF-based DPLL by using a synthetic signal with a realistic GNSS phase and real GNSS signal. With the former signal, the estimators are compared to have a first idea of their behavior when using a realistic phase. Finally, with

the latter signal, the estimators are performed by using the GNSS SDR receiver.

Finally, the manuscript ends with a conclusion paragraph that summarizes the work carried out and suggests future prospects that would be relevant to explore.

(Note that part of the fourth chapter on the RVB algorithm at any order and results of the fifth chapter based on real GNSS data are not published yet. A journal is in preparation [3].)

List of publications

Conference papers

- Fabozzi Fabio, Bidon Stéphanie, Roche Sébastien and Priot Benoît. "Robust GNSS Phase Tracking in Case of Slow Dynamics Using Variational Bayes Inference". In: *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. Bibliography reference: [1]
- Fabozzi Fabio, Bidon Stéphanie and Roche Sébastien. "Robust estimation of high-order phase dynamics using Variational Bayes inference". In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Bibliography reference: [2]

Journal paper (in preparation)

- Fabozzi Fabio and Bidon Stéphanie. "Robust Phase Tracking using Restricted Variational Bayes Inference". In: *IEEE Transactions on Aerospace and Electronic Systems*. (In preparation). Bibliography reference: [3]

Résumé de la thèse

Introduction

Contexte et motivation

Depuis les années 1990, les méthodes GNSS (*Global Navigation Satellite Systems*) de positionnement précis ont connu une expansion qui s'est accélérée dans les années 2000 avec l'avènement de multiple GNSS. Dans ce secteur en pleine évolution, de nouvelles constellations de satellites sont mises en place pour compléter la constellation GPS (*Global Positioning System*) et rendre ces systèmes encore plus disponible et plus fiable comme GLONASS pour la Russie et Galileo pour l'Europe. Le GNSS est utilisé pour déterminer la position d'un récepteur sur terre, en mer ou dans l'espace au moyen de la constellation de plusieurs satellites. La détermination de la position du récepteur, c'est-à-dire la latitude-longitude-hauteur, repose sur la distance estimée par plusieurs satellites. Cette distance entre le satellite et le récepteur est calculée par deux types de mesures : des mesures de pseudo-distance (ou code) et des mesures de phase de porteuse. Le premier est une mesure de la différence entre l'horloge du récepteur à la réception du signal et l'horloge du satellite à la transmission du signal. Le deuxième type de mesure concerne le calcul de la phase instantanée de battement et du nombre cumulé de passages par zéro obtenus après mélange avec un signal de référence à la fréquence nominale. L'avantage des mesures de phase porteuse est qu'elles donnent une précision de positionnement plus élevée que la pseudo-distance ; cependant, les mesures de phase obtenues par les algorithmes de poursuite de phase conventionnels peuvent être gravement altérées par ces sauts qui surviennent particulièrement dans les environnements dégradés. Ces sauts de cycle peuvent même conduire la boucle de poursuite à un décrochage complet dont elle ne se remet jamais. Pour résoudre ce problème, cette thèse de doctorat vise à développer des algorithmes de poursuite de phase de porteuse robuste qui permettant de réduire l'occurrence des sauts de cycle et améliorer la précision de l'estimation par rapport aux techniques de poursuite de phase de porteuse existantes.

Plan du manuscrit

Ce manuscrit de thèse se compose de 5 chapitres. Les chapitres 3, 4 et 5 regroupent nos contributions.

Le premier chapitre décrit le contexte GNSS et introduit le principe du GNSS et le modèle des signaux. Ensuite, l'architecture du récepteur GNSS est décrite, en détaillant les boucles de poursuite.

Le deuxième chapitre se concentre sur la poursuite de la phase porteuse et l'introduction de la *Digital Phase Lock Loop* (DPLL). Nous étudions cette dernière sous une approche par

analyse linéaire et analyse non linéaire. Ensuite, nous étudierons un aperçu des méthodes robustes de suivi de la phase porteuse. En particulier, nous concentrons notre attention sur le KF-DPLL *(DPLL based on Kalman Filter)*.

Dans le troisième chapitre, nous proposons un nouvel algorithme robuste de poursuite de phase de porteuse qui vise à diminuer l'occurrence des sauts de cycle. Nous considérons ici une faible dynamique (c'est-à-dire, boucle du premier ordre). La méthodologie est basée sur l'approximation *Restricted Variational Bayesian* (RVB) qui nous permet alors d'obtenir une expression récursive sous forme explicite de l'estimateur [1]. Les performances de ce dernier sont étudiées ensuite à l'aide d'un signal synthétique en termes de précision d'estimation et de sauts de cycle et comparées à celles des techniques précédemment présentées telles que DPLL et KF-DPLL.

Pour des applications GNSS réalistes, il est nécessaire de considérer une dynamique d'ordre supérieur. Par conséquent, le quatrième chapitre concerne l'algorithme RVB avec une dynamique d'ordre élevé [2], où non seulement la phase mais même ses dérivées sont estimées. La méthodologie suit le même chemin que celui décrit dans le troisième chapitre, mais avec une complexité plus élevée du modèle d'état. Nous obtenons également une expression explicite de l'estimateur récursif RVB. De manière remarquable, celui-ci a une forme similaire à celle obtenue pour une dynamique d'ordre 1 caractérisée par un comportement non linéaire intrinsèque. Ce dernier est comparé à l'aide d'un signal synthétique en termes de précision d'estimation et de sauts de cycle avec une DPLL et une KF-DPLL, et ce, au troisième ordre.

Dans le dernier chapitre, nous implémentons l'algorithme RVB proposé dans un récepteur GNSS *Software Defined Radio* (SDR). À partir d'une boucle de poursuite de phase initiale basée sur une DPLL de troisième ordre, le GNSS SDR a ensuite été modifié en considérant également l'algorithme RVB et la KF-DPLL. Ensuite, une comparaison entre les trois estimateurs est effectuée en utilisant un signal synthétique avec une phase réaliste et des données GNSS réelles.

Enfin, nous concluons le manuscrit par un résumé des travaux réalisés tout au long de la thèse et les perspectives à venir qu'il serait pertinent d'explorer.

(Notons qu'une partie du quatrième chapitre sur l'algorithme RVB d'ordre supérieur n'est pas encore publiée. Nous prévoyons de finaliser un article journal une fois le manuscrit finalisé [3].)

Chapitre 1: Introduction au GNSS

Introduction

Ce chapitre vise à préciser le contexte et le périmètre de la thèse. Les principes fondamentaux du système de navigation par satellite sont présentés. Nous concentrons alors notre attention sur les boucles de poursuite du récepteur. En particulier, les mesures qu'elles fournissent habituellement sont décrites, en mettant l'accent sur les avantages que procurent les mesures de phase de porteuse.

Synthèse des travaux

Principe du GNSS Le *Global Navigation Satellite System* (GNSS) est le concept général désignant les systèmes qui permettent la localisation de l'utilisateur sur la base d'une constellation de satellites. Ces systèmes reposent sur le même principe [4, 5, 6] : l'utilisateur calcule sa position grâce aux distances mesurées entre l'ensemble des satellites visibles et le récepteur. Ces distances sont calculées en estimant le temps de propagation depuis les signaux émis par un satellite jusqu'au récepteur utilisateur. Le temps de propagation est ensuite multiplié par la vitesse du signal pour obtenir la distance satellite-récepteur. En mesurant le temps de propagation du signal diffusé par plusieurs satellites à des emplacements connus, le récepteur peut déterminer sa position au sol par trilatération. Cette méthode permet théoriquement de calculer position, vitesse et temps en utilisant le signal de trois satellites : la distance à laquelle se situe un satellite positionne l'utilisateur à la surface d'une sphère dont le centre est le satellite. L'intersection de 3 sphères permet d'identifier un point unique valide dans l'espace. Un quatrième satellite au minimum est requis pour permettre de déterminer le décalage des horloges et réduire les incertitudes liées aux autres sources de perturbation du signal, on parle alors de multilatération. Le GNSS fonctionne à travers une infrastructure appropriée composée de : 1) le segment spatial (satellites); le segment terrestre (stations au sol); 3) le segment utilisateur (récepteurs).

Structure du signal Pour la plupart des systèmes de navigation, un modèle de signal général consiste en un signal *Direct-Sequence Spread-Spectrum* (DS-SS), qui est transmis de manière synchrone par tous les satellites de la constellation [7, 8, Chap.5]. La technique DS-SS permet alors la mise en oeuvre de la méthode *Code Division Multiple Access* (CDMA, à chaque signal satellite sont associés des codes différents). Le signal GNSS est, souvent, composé de trois composants principaux : un code *Pseudo-Random Noise* (PRN), un message de navigation et un signal porteur.

- *PRN* : le concept de base des schémas CDMA est que chaque satellite se voit attribuer un code PRN qui module le signal transmis. La forme d'onde d'étalement est une série contiguë d'impulsions rectangulaires générées à l'aide d'un PRN déterministe. Dans

le récepteur, une copie locale de la séquence PRN est générée, et est continuellement comparée et alignée avec le signal reçu des satellites.

- *Message de navigation* : il fournit toutes les informations nécessaires pour permettre à l'utilisateur d'effectuer le service de positionnement tels que les paramètres d'éphémérides (pour calculer les coordonnées du satellite avec suffisamment de précision), les paramètres de temps et les corrections d'horloge (pour calculer les décalages d'horloge satellite), et les almanachs (qui permettent le calcul de la position de tous les satellites de la constellation).
- *Signal porteur* : il a un impact sur divers aspects des signaux de navigation par satellite. Son influence va des comportements de propagation incluant l'effet Doppler et les interférences atmosphériques. De nos jours, chaque système de navigation utilise une bande de fréquence comprise entre 1-2 GHz, appelée bande L.

Une fois transmis, le signal se propage à travers un canal qui modifie son amplitude, sa phase/fréquence et son retard. Ce signal est traité par le récepteur GNSS.

Récepteur GNSS Nous analysons une version simplifiée d'un récepteur GNSS générique [9, Chap.6]. En particulier, nous présentons les détails de ses trois parties principales : la frontend, le bloc de traitement du signal (acquisition et poursuite) et enfin la solution de navigation.

- *Frontend* : L'antenne reçoit le signal RF (*Radio Frequency*) des satellites en vue. Le niveau du signal RF capté est très faible. La présence de filtres passe-bande est nécessaire pour rejeter les interférences RF en dehors de la bande passante. Le signal RF amplifié et filtré est ramené en *Fréquence Intermédiaire* (IF) grâce au produit par une sinusoïde pure provenant de l'oscillateur local, générée par un synthétiseur de fréquence qui est piloté par l'horloge de référence. Le signal IF analogique est numérisé par un échantillonnage dont le rythme est déterminé selon la bande de fréquence de ce signal.
- *Bloc de traitement du signal* : Les deux étapes principales du traitement du signal en bande de base sont l'acquisition et le tracking. Dans un premier temps, le récepteur doit détecter les satellites en vue et obtenir une estimation grossière de leurs paramètres de synchronisation tels que le retard et le décalage Doppler. L'acquisition est réalisée par une corrélation entre le signal entrant et une réplique générée localement. Le signal satellite est alors dégagé du niveau de bruit ambiant. Une fois le signal d'un satellite détecté, le récepteur passe en mode poursuite. En plus de la synchronisation avec le message de navigation, son objectif est de mettre à jour les paramètres de synchronisation des satellites détectés (par exemple, les retards, les décalages Doppler et la phase porteuse). Le délai est généralement suivi au moyen d'une *Delay Lock Loop* (DLL). De même, *Digital Phase Lock Loop* (DPLL) et *Frequency Lock Loop* (FLL) sont respectivement utilisés pour le suivi de phase et de fréquence.
- *Solution de navigation* : Le bloc de navigation est chargé de calculer les observables, de démoduler le message de navigation et enfin de calculer la localisation de l'utilisateur.

Mesures GNSS et sauts de cycle Les principales mesures utilisées pour estimer la position de l'utilisateur sont [4, 5, 6] :

- Les *mesures de pseudodistance* entre le satellite et le récepteur : la pseudo-distance est une mesure de la distance entre le satellite et l'antenne du récepteur obtenue en estimant le temps de propagation entre les deux. Ce dernier est évalué à partir d'une analyse de corrélation maximale entre le code du récepteur et le signal GPS.

- Les *mesures de phase* de la porteuse du signal reçu : Le récepteur aligne la porteuse entrante du satellite avec la réplique et mesure le déphasage fractionnaire des deux signaux. En raison de la courte longueur d'onde de la phase porteuse (environ 19-25 cm), la mesure de la phase porteuse est beaucoup plus précise que la mesure du code ($\approx$ 2 ordres de grandeur). Cet avantage se fait au prix d'une mesure ambiguë contrairement à la mesure de code. En effet, si l'on considère un scénario sans erreur, sans mouvement entre le satellite et le récepteur et une synchronisation parfaite de l'horloge, la mesure de la phase de la porteuse resterait fixe à chaque fraction de cycle. Dans ce cas, la distance entre le satellite et le récepteur sera mesurée par un nombre inconnu de cycles entiers, plus le cycle fractionnaire mesuré. Le nombre entier ambigu de cycles de la mesure de la phase porteuse est appelé *ambiguïté*.

  De plus, le récepteur GNSS peut temporairement perdre son verrouillage en raison d'effets tels qu'une dynamique du signal élevée inattendue, une perte courte de signal, un fort trajet multiple ou un bruit élevé. Lorsqu'un tel événement se produit, une discontinuité du nombre entier de cycles dans la phase de porteuse mesurée se produit, connue sous le nom de *sauts de cycle*. Même lorsque cet événement ne se produit que pendant une fraction de seconde et que le verrouillage de phase est rétabli, le nombre d'ambiguïté peut avoir changé, car la boucle de poursuite ne réajuste que la partie fractionnaire mais pas la partie entière. De plus, les pertes de verrouillage locales peuvent conduire la boucle de poursuite à un décrochage complet dont elle ne se remet jamais. Dans ce cas, une ré-acquisition est nécessaire, affectant gravement la qualité du positionnement. (L'étude plus approfondie du phénomène de sauts de cycle se fait au cours du chapitre suivant).

Conclusion

Ce chapitre donne un aperçu du système de navigation. La première partie rappelle notamment le principe de fonctionnement d'un GNSS et la structure typique d'un signal GNSS (transmis et reçu). Dans une deuxième partie, le récepteur GNSS est présenté. En particulier, la boucle de poursuite est examinée avec plus d'attention. Enfin, les principales mesures GNSS fournies par le canal de poursuite sont présentées en expliquant les avantages et les inconvénients des mesures de phase porteuse.

Chapitre 2: DPLL et techniques de poursuite de phase robustes

Introduction

Ce chapitre présente la technique de poursuite de phase porteuse la plus courante, appelée *Digital Phase Lock Loop* (DPLL). En particulier, nous étudions son comportement en régimes linéaire et non linéaire. Ce dernier se produit en raison de la présence de sauts de cycle. Pour y remédier, plusieurs techniques ont été proposées dans la littérature. Un aperçu de ces techniques de suivi de phase robustes est ensuite présenté. En particulier, nous concentrons notre attention sur la DPLL basée sur le *Kalman Filter* (KF-DPLL), qui avec la DPLL seront utilisés comme algorithmes de référence pour la comparaison avec celui que nous proposons dans les prochains chapitres.

Synthèse des travaux

DPLL La DPLL est un élément clef de la chaîne de tracking et est en charge de la poursuite de la phase porteuse. Une DPLL est une boucle synchronisant en phase le signal de sortie (généré localement par un oscillateur) avec un signal de référence ou entrant [10]. L'architecture DPLL est formée par :

- un *corrélateur* : il est implémenté comme un mélangeur et sa tâche est de comparer le signal entrant avec la réplique locale ;

- un *discriminateur* : son but est d'extraire l'erreur de paramètre de signal des sorties du corrélateur I (en-phase) et Q (quadrature). Suivant le type de discriminateur implémenté, on peut distinguer deux types de boucles de poursuite de phase [5, Chap.5] : une DPLL pure (ou cohérente) et une DPLL dite de Costas (ou non cohérente). Dans ce manuscrit, nous utiliserons une DPLL basée sur un discriminateur ATAN2, puisque nous considérerons plus loin un signal sans message de navigation ;

- un *filtre de boucle* : il est placé à la sortie du discriminateur pour filtrer le bruit sur l'erreur d'estimation. La plupart des récepteurs GNSS utilisent des DPLL de deuxième ou troisième ordre. Comme souligné précédemment, plus l'ordre de la DPLL est élevé, plus la dynamique de phase qu'il peut suivre est élevée. Cependant, le prix à payer est une diminution de la robustesse au bruit. Nous utilisons une boucle du premier ordre au chapitre 3 et une boucle du troisième ordre aux chapitres 4-5 pour la comparaison avec nos algorithmes proposés ;

- a *Numerical Control Oscillator* (NCO) et *générateur de réplique* : le NCO est un générateur de signaux numériques. En fait, il peut être vu comme un simple accumulateur du rythme de phase pour produire la phase. Cette dernière est ensuite transmise à un générateur de porteuse pour produire une réplique déphasée à l'itération suivante.

La DPLL est un algorithme de poursuite non linéaire. La non-linéarité est liée au discriminateur qui est périodique et nécessairement non linéaire. Un modèle linéaire de la DPLL est généralement applicable lorsque l'erreur de phase est faible, une condition normalement atteinte lorsque la boucle est verrouillée [11]. Sous ce régime, les outils conventionnels des systèmes linéaires comme l'analyse de la fonction de transfert équivalente peuvent être utilisés pour la détermination des réponses de la DPLL. En fait, grâce aux outils de l'analyse linéaire, l'erreur en régime permanent est facilement quantifiable. En particulier, nous l'avons calculé pour différentes dynamiques d'éntrée et différents ordres de boucle. De plus, nous avons noté que l'erreur en régime permanent et le temps de convergence de la DPLL sont fortement influencés par la valeur de la bande passante du bruit de boucle. Nous avons observé que lorsque la bande passante de la boucle est élevée, nous obtenons de meilleures performances en termes de réponse temporelle de la boucle et d'erreur en régime permanent. Cependant, la bande passante de la boucle influence également directement la puissance du bruit de phase en sortie. Par conséquent, le choix de la bande passante résulte d'un compromis entre la dynamique de phase à suivre et le niveau de bruit en sortie. En outre, l'analyse linéaire échoue lorsque l'erreur réelle en présence de bruit n'est pas nécessairement petite. On se tourne alors vers d'autres outils d'analyse, à savoir : la densité de probabilité (PDF, *Probability Density Function*) de l'erreur de phase (qui est non Gaussienne) ; le calcul de la variance de la phase à partir de cette dernière, l'étude de l'occurence des sauts de cycle. Notez que l'analyse non linéaire est, de manière générale, difficile car elle exige un haut niveau de sophistication mathématique. A titre d'exemple, nous nous référons à l'étude de Viterbi d'une boucle analogique du premier ordre qui a apporté de nombreuses informations pour comprendre le comportement non linéaire de la DPLL [12]. Dans ce résumé, nous nous concentrons plutôt sur le phénomène de sauts de cycle. Pour avoir une compréhension visuelle de ce dernier, nous présentons des exemples de sauts de cycle et de décrochage dus à un faible SNR (*Rapport SigNal-sur-bruit*) dans la Fig. 1 et Fig. 2, respectivement. Une DPLL de second ordre avec un discriminateur ATAN2 est utilisée. Comme on peut le voir en cas de SNR faibles, un décrochage se produit. Notez que, dans la suite du manuscrit, le phénomène de sauts de cycle est évalué à l'aide des métriques *Mean Time To First Slip* (MTFS) et du taux de saut de cycle. Ces dernières sont utilisées pour la comparaison entre les techniques conventionnelles et les nouvelles que nous proposons aux chapitres 3, 4 et 5.

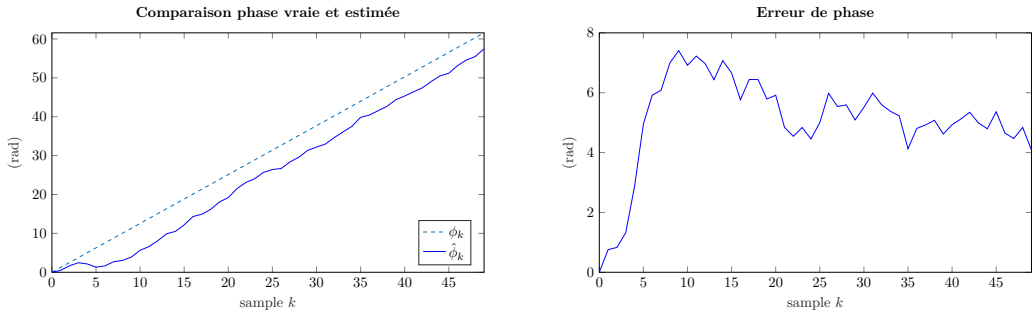


Figure 1: Sauts de cycle en raison d'un faible SNR. ($T = 0.02$ s; $\sigma_n = 1$; $B_L = 10$ Hz; $\dot{\phi}_0 = \pi/25$ rad/sample; $\hat{\phi}_0 = 0$ rad/sample; $C/N_0 = 22$ dB-Hz).

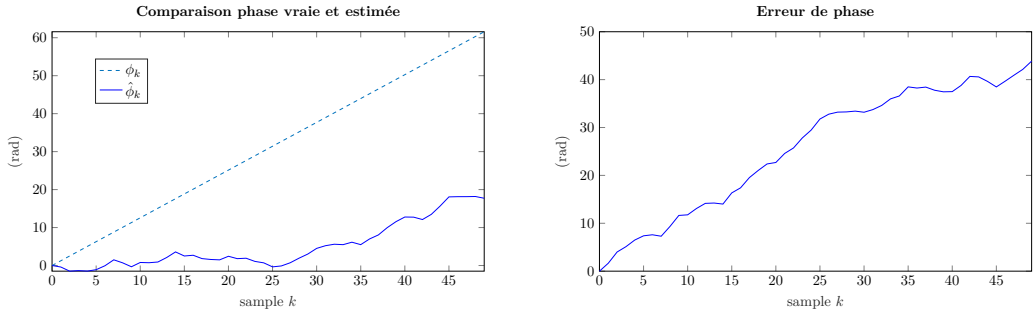


Figure 2: Décrochage en raison d'un faible SNR. ($T = 0.02$ s; $\sigma_n = 1$; $B_L = 10$ Hz; $\dot{\phi}_0 = \pi/25$ rad/sample; $\hat{\dot{\phi}}_0 = 0$ rad/sample; $C/N_0 = 17$ dB-Hz).

Techniques de suivi de phase robustes et zoom sur la KF-DPLL Les exigences strictes, telles que la robustesse à faible SNR, données par les applications de poursuite de phase sont fondamentales pour obtenir un positionnement précis. Malheureusement, la DPLL peut être incapable de les remplir dans des environnements hostiles. La raison principale est due au fait que la DPLL est conçue en considérant un comportement linéaire et gaussien de l'erreur de phase. Afin de surmonter ces limitations, des techniques de poursuite de phase robustes ont été proposées dans la littérature [13]. Nous concentrons notre attention sur la KF-DPLL qui a été proposée et étudiées dans la communauté du tracking [14, 15]. La KF-DPLL correspond au filtre optimal Bayésien dans l'hyothèse d'un modèle d'état linéaire et Gaussien. Contrairement à la DPLL, la KF-DPLL ajuste alors la bande passante de la boucle au cours du temps.

Conclusion

Ce chapitre a introduit le principe de la DPLL. En particulier, sa structure est examinée en analysant chaque bloc faisant partie de la boucle. De plus, une analyse linéaire et non linéaire de la boucle de poursuite a été exposée avec également l'étude du phénomène de sauts de cycle. Nous avons évoqué les limites de cette architecture traditionnelle, notamment liées à sa capacité à travailler dans des environnements sévères. En fait, à faible SNR, la DPLL manque de robustesse puisqu'elle est conçue en considérant un comportement linéaire et gaussien de l'erreur de phase. Par conséquent des méthodes alternatives ont été proposées dans la litérature. Dans la suite, nous avons concentré notre attention sur la célèbre KF-DPLL, qui est une approche optimale pour estimer les paramètres variant dans le temps (c'est-à-dire la bande passante de la boucle). Cette dernière et la DPLL traditionnelle seront utilisées pour comparer leurs performances à celles obtenues pour les nouveaux algorithmes que nous proposons dans les chapitres suivants.

Chapitre 3: Algorithme RVB en cas de dynamique lente

Introduction

Un nouvel algorithme de suivi de phase porteuse robuste aux sauts de cycle est présenté ici. Ce dernier est basé sur l'approximation bayésienne variationnelle utilisée localement dans le problème de filtrage bayésien optimal. Dans un premier temps, le modèle d'état considéré est introduit. Ensuite, l'inférence VB locale et sa version restreinte sont appliquées. Dans notre problème, elles permettent d'obtenir une expression explicite de l'estimateur récursif de phase. Enfin, des simulations numériques sont mises en oeuvre pour mesurer les performances de l'algorithme RVB proposé. Elles sont comparées aux deux techniques DPLL et KF-DPLL, montrant ainsi des améliorations en termes de précision d'estimation et de sauts de cycle.

Synthèse des travaux

Algorithme RVB à faible dynamique de phase Avant d'introduire l'algorithme proposé, un *State-Space Model* (SSM) du problème de poursuite de phase est nécessaire. Cette dernière est formée par les équations de mesure et de dynamique dans la cas d'une évolution de phase présumée lente. Contrairement à ce qui est pratiqué pour développer la KF-DPLL, le SSM ici n'est pas linéarisé mais est étudié tel quel. L'équation de mesure est représentée par le signal d'entrée de la boucle, qui dans notre thèse est une exponentielle complexe avec un *Additive White Gaussian Noise* (AWGN). Certaines hypothèses sont faites sur le signal et l'architecture du récepteur. En particulier : 1) architecture du récepteur GNSS : le signal d'entrée est celui du corrélateur Prompt (une réplique alignée avec le signal entrant), et une synchronisation parfaite du retard et de la fréquence est supposée ; 2) signal d'entrée : un signal sans message de navigation est considéré (signal pilot ou mise en oeuvre d'un data-wipeoff), le bruit est modélisé par un AWGN. L'évolution de la phase est modélisée par un *Markov Random Field* (MRF) gaussien du premier ordre qui assure une certaine régularité dans la séquence de phase estimée [16]. Le problème de filtrage bayésien optimal de notre SSM est simplifié en utilisant l'inférence *Variational Bayesian* (VB). La première étape de cette approche est le choix de la forme de factorisation des a posteriori approchées. Ensuite, l'expression analytique de la distribution approchée est obtenue en minimisant la divergence de Kullback-Leibler. Au sein de notre algorithme de filtrage, une seconde approximation est alors appliquée. Cela se fait par l'approximation *Restricted VB* (RVB) qui laisse invariante la forme fonctionnelle de la distribution postérieure sur le processus de récursivité. L'approche RVB dans notre problème de poursuite de phase s'avère nous donner une solution itérative tractable du moment que la moyenne a posteriori (au sens de la distribution approchée) peut-être évaluée. Notez que la méthodologie est la même que celle utilisée dans [17], où la poursuite de phase VB est appliquée à des mesures bifréquence corrélées avec une dynamique

de phase lente. La formule de récurrence de l'estimateur RVB est [1]

$$\hat{\phi}_k^{\text{rvb}} = \hat{\phi}_{k-1}^{\text{rvb}} + 2\sigma_\phi^2 \frac{\sum_{q=1}^{+\infty} q I_q(\beta_k) \sin[q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})] e^{-\frac{q^2\sigma_\phi^2}{2}}}{I_0(\beta_k) + 2\sum_{q=1}^{+\infty} I_q(\beta_k) \cos[q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})] e^{-\frac{q^2\sigma_\phi^2}{2}}}, \tag{1}$$

où $\hat{\phi}_k^{\text{rvb}}$ est la phase estimée, les $I_q(\cdot)$ sont les fonctions de Bessel modifiées de première espèce au q-ième ordre et β_k est un paramètre de traitement qui dépend de l'amplitude du signal et de la variance de bruit (toutes deux supposées connues). Notez qu'une forme implémentable de (1) est obtenue en tronquant la somme impliquée pour $q = 1, ..., q_{\max}$, où $q_{\max}$ est l'ordre le plus élevé choisi. L'un des paramètres de traitement les plus importants est la puissance de bruit du processus σ_ϕ^2. Elle intervient dans le terme d'innovation porté par d'innovation $\psi_k - \hat{\phi}_{k-1}^{\text{rvb}}$ et y intervient de manière non linéaire (dans les termes exponentiels). La puissance de bruit du processus σ_ϕ^2 peut être considérée comme équivalent à la bande passante de boucle du DPLL : elle impacte le niveau de bruit passant dans la boucle et la dynamique de phase qui peut être suivie. Enfin, on peut noter une certaine similarité avec le KF [18], où l'état est estimé par la somme du terme prédit et d'un terme d'innovation.

Analyse de performance Les performances de l'estimateur RVB proposé dans ce chapitre sont évaluées avec des données synthétiques. L'estimateur RVB est comparé aux techniques traditionnelles de poursuite de phase, à savoir la DPLL et la KF-DPLL. Étant donné que l'algorithme RVB est conçu dans ce chapitre au premier ordre uniquement, la comparaison est évaluée pour les dynamiques de phase lente, telles que les entrées de phase de types échelon et rampe. Les métriques de performance considérées peuvent être regroupées en deux familles distinctes : 1) précision des métriques d'estimation (*Root Mean Square Error* (RMSE) et temps d'acquisition) où le phénomène de sauts de cycle est négligé puisque l'on considère l'erreur de phase modulo-2π ; 2) les métriques de sauts de cycle, à savoir le taux de saut de cycle et le MTFS. Dans cette synthèse, nous allons présenter les résultats pour une entrée de type échelon uniquement. Ces derniers sont présentés dans les Figs. 3-4. La simulation a été mise en œuvre à faible SNR pour réprésenter un environnment dégradé. Comme on peut le voir pour les deux familles de métriques considérées, l'algorithme RVB surpasse les deux autres estimateurs traditionnels. Un résultat similaire est obtenu dans le cas d'un phase d'entrée de type rampe [1].

Conclusion

Dans ce chapitre, nous proposons un algorithme robuste pour la poursuite de phase. Ce dernier est basé sur l'inférence bayésienne variationnelle, et plus précisément, sur une approximation dite RVB que nous donne une expression explicite de l'estimateur et facile à mettre en œuvre. L'algorithme RVB préserve le caractère non linéaire de l'équation de mesure ce qui est bénéfique à un faible rapport signal sur bruit. En fait, par simulations nous avons montré l'avantage de l'estimateur RVB par rapport à des techniques traditionnelles pour poursuivre des dynamiques de phase lentes (dans notre cas de type échelon et rampe) en termes de

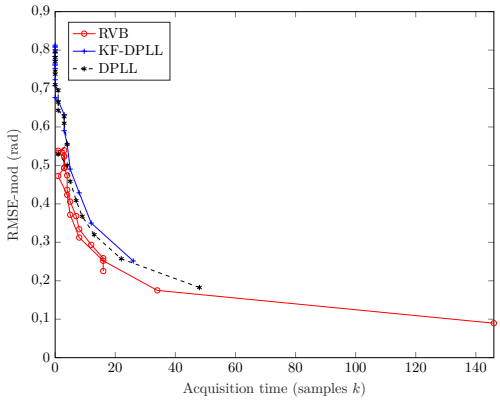


Figure 3: Temps d'acquisition et RMSE-mod utilisant une entrée échelon de phase pour la DPLL, la KF-DPLL et le RVB. ($\phi_0 = \pi/4$ rad; $T = 0.02$ s; $C/N_0 = 17$ dB-Hz; $M_c = 10000$ runs.)

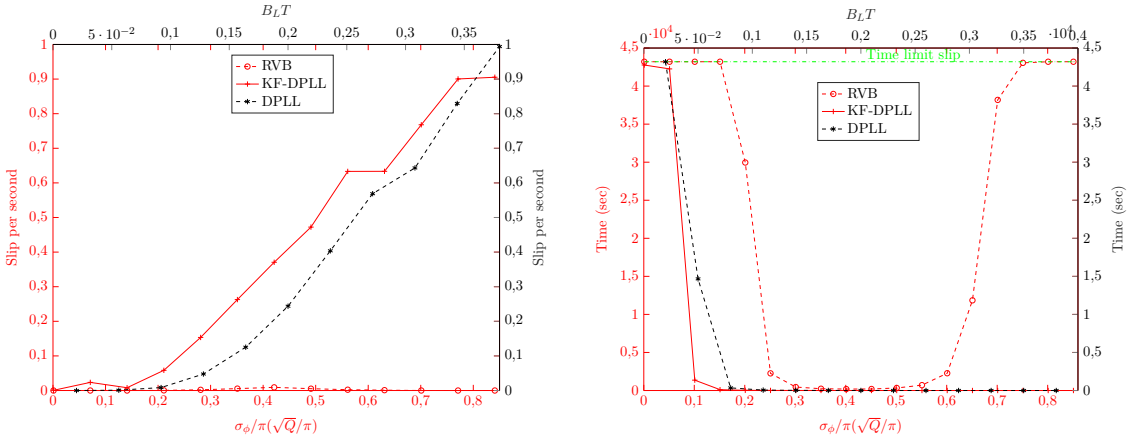


Figure 4: Taux de saut de cycle et MTFS utilisant une entrée échelon de phase pour la DPLL, la KF-DPLL et le RVB. ($\phi_0 = \pi/4$ rad; $T = 0.02$ s; $C/N_0 = 15$ dB-Hz; $M_c = 500$ runs; $T_{\text{Limit}} = 3200$ s.)

précision d'estimation et sauts de cycle lorsque le rapport signal sur bruit est faible. Dans le chapitre suivant, pour aborder des scénarios plus pratiques, l'algorithme RVB proposé est étendu à une dynamique de phase d'ordre supérieur.

Chapitre 4: Algorithme RVB pour des dynamiques d'ordre supérieur

Introduction

Dans ce chapitre, l'algorithme RVB en cas de dynamique d'ordre élevé est présenté. Le vecteur d'état à estimer est formé par la phase porteuse et ses dérivées. L'approche commence par l'introduction du modèle d'état, puis décrit la mise en oeuvre de l'inférence VB, et enfin, se termine par le calcul de la solution sous forme explicite de l'estimateur RVB. Ensuite, ce dernièr est comparé aux techniques conventionnelles telles que la DPLL et la KF-DPLL dans des scénarios de dynamique de phase d'ordre élevé. Comme dans le cas de l'algorithme RVB du premier ordre, la comparaison est faite en tenant compte des métriques portant à la fois sur la précision d'estimation (module 2π) et de l'occurence des sauts de cycle.

Synthèse des travaux

Algorithme RVB à dynamique d'ordre élevé Dans le chapitre précédent, nous avons considéré un SSM où l'équation de la dynamique était exprimée au premier ordre. Dans ce cas, seule la phase de la porteuse a été directement estimée par l'algorithme RVB proposé. Cependant, pour des applications GNSS réalistes, il est nécessaire de considérer une dynamique d'ordre supérieur. À cet effet, le vecteur d'état considéré comportera non seulement l'information de phase de porteuse mais aussi ses dérivées jusqu'à un ordre prédéfini. Il est intéressant de noter que bien que le SSM ait une complexité supérieure à celle du premier ordre comme dans le chapitre 3, l'expression récursive sous forme explicite de l'algorithme RVB reste similaire (cf. (1)) et est caractérisée par un comportement non linéaire intrinsèque. Il est intéressant de noter qu'en faisant fi de l'aspect non linéaire du terme d'innovation, l'estimateur RVB préserve une forme quelque peu similaire à celui d'un estimateur de Kalman constitué d'un terme de prédiction plus un terme d'innovation. L'algorithme d'obtient comme suit : dans un premier temps, le problème de filtrage bayésien optimal de notre SSM est formulé ; ensuite, l'inférence VB et l'approximation RVB sont appliquées, nous donnant une expression sous forme fermée de l'estimateur pour la dynamique d'ordre élevé. La mise à jour de l'estimateur RVB s'écrit alors [2].

$$\hat{\boldsymbol{x}}_k^{\mathrm{rvb}} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\mathrm{rvb}} + 2\frac{\sum_{q=1}^{+\infty} qI_q(\beta_k)\sin[q(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\mathrm{rvb}}]_1)]e^{-\frac{q^2Q_{1,1}}{2}}}{I_0(\beta_k) + 2\sum_{q=1}^{+\infty} I_q(\beta_k)\cos[q(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\mathrm{rvb}}]_1)]e^{-\frac{q^2Q_{1,1}}{2}}}\boldsymbol{Q}_1, \tag{2}$$

où $\hat{\boldsymbol{x}}_k^{\mathrm{rvb}}$ est le vecteur d'état estimé formé par la phase porteuse et ses dérivées, les $I_q(\cdot)$ sont les fonctions de Bessel modifiées de première espèce au q-ième ordre et β_k est un paramètre de traitement qui dépend de l'amplitude du signal et de la variance de bruit de mesure (toutes deux supposées connues). Notez que comme dans le cas du premier ordre, à savoir (1), l'équation (2) doit être tronquée à un certain $q_{\max}$ afin d'être implémentée. $\boldsymbol{Q}_1$ représente la première colonne de la matrice de covariance du bruit du vecteur d'état. Notons que

l'algorithme maintient le caractère non linéaire de l'équation de mesure. De plus, on remarque immédiatement que l'estimation de l'état $\hat{\boldsymbol{x}}_k^{\text{rvb}}$ dépend en partie de la matrice de covariance $\boldsymbol{Q}$; nous avons observé que cela a un impact sur le choix numérique de $\boldsymbol{Q}$. Enfin, comme dans le cas de l'algorithme RVB au premier ordre, une certaine similitude avec la solution KF est évidente. L'analyse des performances de cet algorithme est présentée tout de suite.

Analyse de performance Dans cette section, l'estimateur RVB est comparé à des techniques traditionnelles de suivi de phase, à savoir la DPLL et la KF-DPLL. Notez que nous présentons les résultats de notre principal paramètre d'intérêt, c'est-à-dire l'estimation de la phase $[\boldsymbol{x}_k]_1$. Dans les simulations, la phase est générée avec une évolution parabolique. Les métriques comparées sont : la RMSE modulo 2π (RMSE-mod), le taux de sauts de cycle et le MTFS. Les paramètres de génération du signal et des trois algorithmes sont présentés dans le Tab. 1. Les métriques sont évaluées via un nombre M_c de simulations de Monte Carlo ; $M_c = 10000$ pour l'évaluation du RMSE-mod et $M_c = 500$ pour les métriques de sauts de cycle (le nombre de simulations est diminué ici pour éviter des temps de simulation excessifs : en pratique nous nous sommes restreints à des simulations ne dépassant jamais la demi-journée). Les paramètres de traitement (tels que le type de discriminateur des deux estimateurs conventionnels, $q_{\max}$ pour l'estimateur RVB, etc.) des trois estimateurs sont définis comme dans la légende des Figs. 3-4 du chapitre 3. Comme on peut le voir dans les Figs. 5-6, pour toutes les métriques considérées, l'algorithme RVB a de meilleures performances que les deux autres techniques [2]. En particulier, une meilleure précision d'estimation est obtenue pour la quasi-totalité des SNR considérés et une meilleure robustesse en termes de sauts de cycle est obtenue aux faibles SNR. À cet égard, notez que pour un SNR supérieur à zéro décibel, aucun saut ne se produit pour l'algorithme RVB jusqu'à la limite de temps imposée sur le nombre de récursions de l'estimateur (soit une demi-journée de poursuite pour un récepteur).

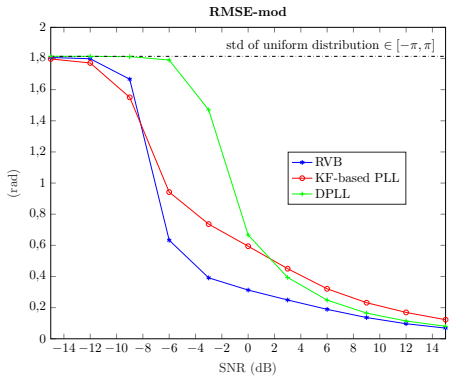


Figure 5: RMSE-mod pour une boucle du troisième ordre utilisant une entrée parabolique

Table 1: Paramètres d'entrée

Paramètre	**Variable**	**Valeur**
Phase	ϕ_0	0 rad
Taux de phase	$\dot{\phi}_0$	0 rad/s
Accélération de phase	$\ddot{\phi}_0$	$\pi/0.16$ rad/s^2
Temps d'intégration	T	0.02 s
RVB PSDs (Power spectral Density)	$\sigma_{\phi}^{\text{rvb}} \triangleq \sqrt{\Sigma_{\text{p}} T}$	0.8π rad
	$\sigma_{\dot{\phi}}^{\text{rvb}} \triangleq \sqrt{\Sigma_{\text{pv}} T}$	20π rad/s
	$\sigma_{\ddot{\phi}}^{\text{rvb}} \triangleq \sqrt{\Sigma_{\text{pva}} T}$	100π rad/s^2
KF-DPLL PSDs	$\sigma_{\phi}^{\text{kf-dpll}} \triangleq \sqrt{\Sigma_{\text{p}} T}$	0.2π rad
	$\sigma_{\dot{\phi}}^{\text{kf-dpll}} \triangleq \sqrt{\Sigma_{\text{pv}} T}$	0.8π rad/s
	$\sigma_{\ddot{\phi}}^{\text{kf-dpll}} \triangleq \sqrt{\Sigma_{\text{pva}} T}$	0.2π rad/s^2
Bande passante de boucle DPLL	B_L	10 Hz

Conclusion

Dans ce chapitre, nous avons présenté une technique de poursuite de phase robuste vers les sauts de cycle à n'importe quel ordre de dynamique. Il peut être vu comme une extension de l'algorithme présenté au chapitre 3, qui estime une dynamique de phase du premier ordre qui peut ne pas suffire pour certaines applications GNSS. Cette fois, le vecteur d'état est formé par la phase et ses dérivées. Comme pour l'algorithme RVB à dynamique lente, la méthode est également basée sur l'approximation RVB, qui conduit à un filtre non linéaire avec une expression explicite. Les performances de l'estimateur MMSE associé sont évaluées en termes de RMSE-mod, de taux de sauts de cycle et du MTFS, et comparées à une DPLL conventionnelle et à une KF-DPLL. Le nouvel algorithme RVB non linéaire montre une amélioration significative des performances à un faible rapport signal sur bruit.

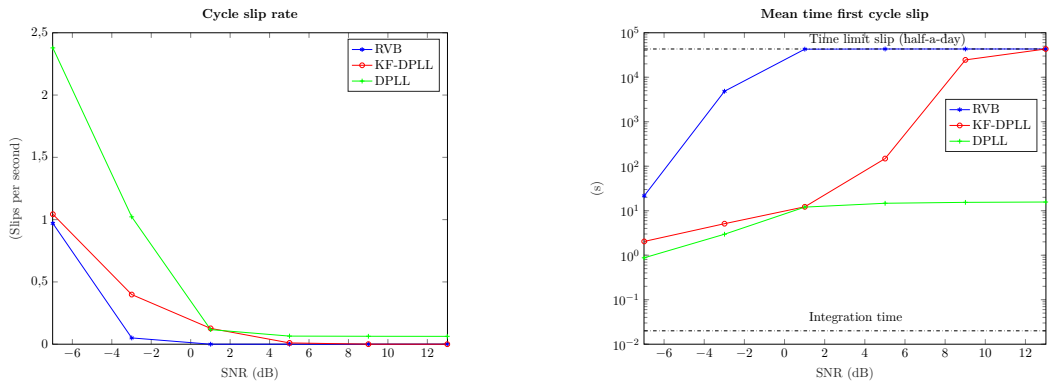


Figure 6: Taux de sauts de cycle et MTFS utilisant une entrée de phase parabolique

Chapitre 5: Algorithme RVB utilisant des données GNSS réelles

Introduction

Dans ce dernier chapitre, l'algorithme RVB de poursuite de phase est étudié en utilisant des données GNSS réelles. Nous avons pour ceci utilisé un récepteur GNSS SDR, développé par l'ISAE-SUPAERO antérieurement à cette étude. Nous décrivons dans un premier temps à haut niveau ce dernier. Ensuite, les caractéristiques réelles des données GNSS utilisées sont introduites. Dans ce chapitre, la comparaison entre l'algorithme proposé et les techniques traditionnelles se fait sur deux niveaux : 1) signal synthétique avec une phase réaliste (phase estimée par une DPLL de troisième ordre dans le GNSS SDR) ; 2) données GNSS réelles. Pour les deux types de données, la comparaison se fait pour un seul satellite. Les résultats montrent des améliorations significatives en termes de robustesse aux sauts de cycle en utilisant l'algorithme RVB par rapport aux techniques traditionnelles.

Synthèse des travaux

Récepteur radio logicielle GNSS et données réelles Les récepteurs GNSS SDR font, de nos jours, l'objet d'une attention considérable. L'objectif majeur d'un récepteur SDR est de mettre en œuvre efficacement des calculs complexe tout en conservant la flexibilité souhaitée inhérente à une approche logicielle, ce qui n'est pas le cas des récepteurs GNSS classiques. Notre GNSS SDR a été implémenté au sein de notre laboratoire. Son architecture est présentée du frontend jusqu'à la solution de navigation. En particulier, nous portons notre attention sur le bloc de poursuite, où la DPLL de troisième ordre originellement implémentée peut-être remplacée par un estimateur RVB ou bien une KF-DPLL de même ordre. À noter qu'en raison des hypothèses faites sur le modèle de signal supposé pour la DPLL et l'architecture du récepteur, nous avons dû modifier légèrement le signal Prompt entrant avant la poursuite. En particulier, les données de navigation sont supprimées, et le retard et la fréquence sont compensés via le retard estimé à partir d'une DLL (*Delay Lock Loop*) de premier ordre et la fréquence estimée par l'étage d'acquisition, respectivement. Les données GNSS réelles sont collectées auprès d'un utilisateur statique via le récepteur Ettus Research USRP X310. Les échantillons sont enregistrés en bande de base avec une fréquence d'échantillonnage de 4 MHz (2 x 4 bits par échantillon complexe). Nous analysons un seul satellite avec le C/N_0 le plus élevé, puis le dégradons en ajoutant un AWGN complexe. La comparaison entre les trois algorithmes se fait sur deux niveaux : l'un considérant un signal synthétique avec une phase réaliste (estimée par un GNSS SDR avec un DPLL du troisième ordre) et un autre considérant des données GNSS réelles (intactes et dégradées). L'analyse des performances sur ce signal GNSS synthétique nous aide à avoir une première idée du comportement des trois algorithmes que nous allons comparer sur signaux réels. Enfin, il est important de noter que l'algorithme RVB suppose connaître l'amplitude du signal et la variance du bruit de mesure. Nous proposons alors de mettre en œuvre des estimateurs ad-hoc de ces deux derniers paramètres. En particulier, ces dernières sont estimées en utilisant la *Variance Summing Method* (VSM) [19, 20].

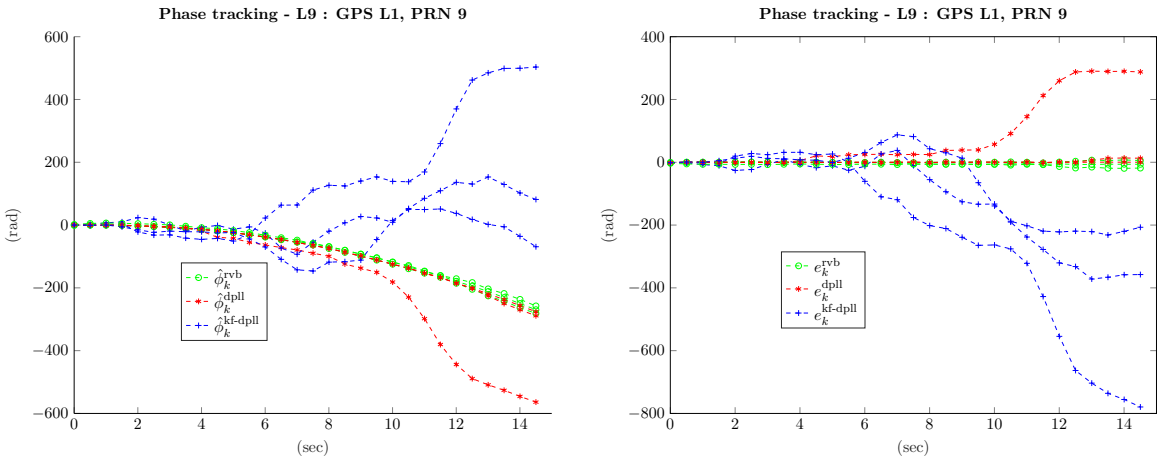
Analyse de performance Dans ce résumé, nous montrons les résultats des seules données GNSS réelles dégradées par un AWGN. Il faut aussi dire que, bien que tous les résultats ne soient pas montrés ici, les trois estimateurs étudiés sont capables de poursuivre la phase lorsque le vrai signal GNSS n'est pas dégradé ($C/N_0 = 52$ dB-Hz). Les paramètres de traitement des trois estimateurs sont définis dans le Tab. 1. Cependant, notons que maintenant le temps d'intégration T est fixé à 1 ms. Les résultats sont présentés dans la Fig. 7. L'estimateur RVB ne présente aucun décrochage sur l'ensemble des trajectoires et n'expérience que très peu de sauts de cycle. Au contraire, la DPLL est affectée par plusieurs sauts qui arrivent parfois par salve. La KF-DPLL fait preuve d'un comportement différent puisque de très nombreux sauts de cycle ont lieu qui mènent à décrochage. Pour mieux comprendre le comportement de la KF-DPLL, nous avons utilisé des résultats de la litérature [21, 22] montrant qu'en régime permanent le traitement est équivalent à une DPLL. En particulier, les gains de filtre équivalents sont exprimés en fonction des gains de Kalman. A partir de ces premiers, la bande de boucle équivalente peut être calculée [23, Tab.IV]. Dans le scénario décrit dans le Tab. 1, la bande de boucle équivalente est ainsi égale à ≈ 17 Hz. Par conséquent, pour avoir une comparison plus équitable, nous avons changer les paramètres de la KF-DPLL pour obtenir une bande de boucle $B_L \approx 10$ Hz comme pour le DPLL. Les nouvelles PSDs considérées sont présentées dans le Tab. 2. Les nouvelles simulations sont affichées pour le KF-DPLL dans la Fig. 8. Comme on peut le constater, l'occurrence des sauts est diminuée comme prévu, cependant la boucle paraît toujours moins robuste que l'algorithme RVB.

Table 2: Paramètres de traitement KF-DPLL

KF-DPLL PSDs	$\sqrt{\Sigma_{\mathrm{p}} T}$	0.08π rad
	$\sqrt{\Sigma_{\mathrm{pv}} T}$	0.8π rad/s
	$\sqrt{\Sigma_{\mathrm{pva}} T}$	0.2π rad/s^2

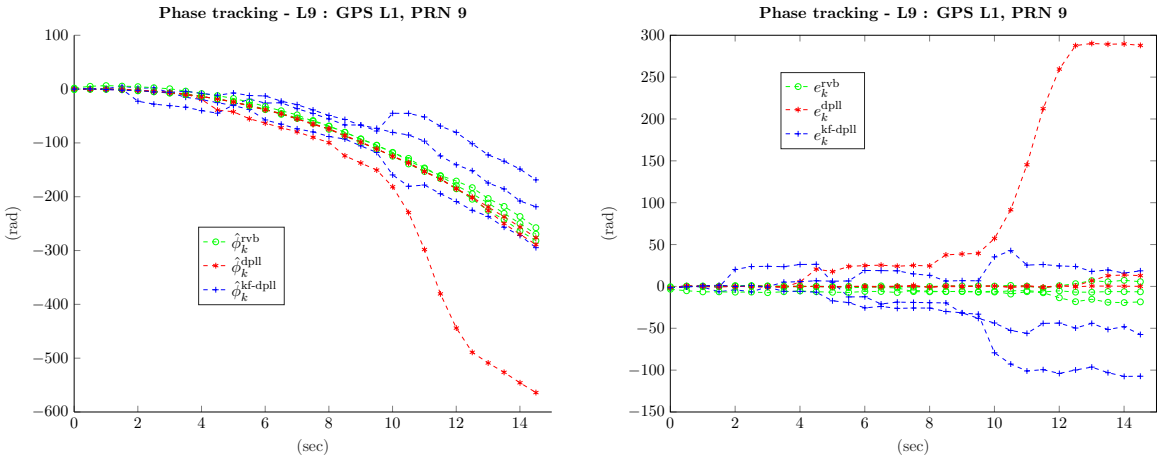
Conclusion

Dans ce chapitre, nous avons analysé l'algorithme RVB pour une dynamique de phase d'ordre élevé en utilisant des données GNSS réelles. Un récepteur radio logicielle GNSS développé précédemment par l'ISAE-SUPAERO a été utilisé. En particulier, une comparaison avec les techniques traditionnelles telles que la DPLL et la KF-DPLL est effectuée dans le cas d'un signal synthétique utilisant une dynamique de phase réaliste et un signal GNSS réel. Concernant le premier signal, l'analyse a pour but d'avoir une première idée du comportement des trois estimateurs considérés en utilisant une dynamique de phase réaliste. Ensuite, ils sont comparés via un ensemble de données GNSS brutes qui est dégradé en ajoutant du bruit blanc gaussien. Les résultats montrent que les trois estimateurs sont capables de poursuivre la dynamique de phase en cas de faible niveau de bruit. A l'inverse, lorsque le signal est dégradé, seul l'algorithme RVB montre une faible occurence de sauts de cycle.



(a) Phase tracking (b) Phase error

Figure 7: Poursuite de phase utilisant des données GNSS réelles à C/N_0 environ 15 dB-Hz



(a) Phase tracking (b) Phase error

Figure 8: Poursuite de phase utilisant des données GNSS réelles à C/N_0 environ 15 dB-Hz (paramètres de traitement KF-DPLL changés, Tab.2)

Conclusion et travaux futurs

Conclusion Cette thèse de doctorat porte sur le développement d'un algorithme de poursuite de phase robuste basé sur l'inférence bayésienne variationnelle. L'estimation de la phase porteuse est devenue une tâche fondamentale dans de nombreuses applications d'ingénierie telles que la navigation. À propos de ce dernier, les mesures de phase fournissent une grande précision de la position d'un utilisateur/système. Cependant, les mesures de phase obtenues par les algorithmes de suivi de phase conventionnels peuvent être gravement altérées par la présence de sauts de phase ambigus, appelés sauts de cycle. Les sources des sauts peuvent être diverses comme un faible rapport signal sur bruit. Il peut arriver que ces pertes de verrouillage locales puissent même conduire la boucle de poursuite à un décrochage complet dont elle ne récupère jamais. Par conséquent, le problème principal de cette thèse est de proposer un algorithme de poursuite de phase permettant de décroître l'occurrence des sauts de cycle par rapport aux méthodes traditionnelles.

Dans le premier chapitre, le contexte GNSS est introduit. En particulier, nous concentrons notre attention sur les principes fondamentaux du GNSS, la structure du signal et le récepteur GNSS. Nous introduisons les mesures de phase porteuse, en expliquant les avantages de ces dernières par rapport aux mesures de code. Cependant, les mesures de phase comportent des inconvénients bien connus comme le phénomène de sauts de cycle.

Dans le deuxième chapitre, nous présentons la célèbre DPLL. En particulier, nous présentons une analyse linéaire et non linéaire de cette dernière, en se concentrant sur l'étude du phénomène de sauts de cycle. Nous examinons les limites de cette boucle de poursuite, en particulier la faible robustesse dans les environnements dégradés. Un aperçu des techniques de poursuite de phase robuste existantes est ensuite présenté. En particulier, nous concentrons notre attention sur la KF-DPLL. Ces deux techniques, DPLL et KF-DPLL, sont utilisées pour comparer leurs performances avec celles des nouveaux algorithmes proposés dans ce travail de thèse.

Dans le troisième chapitre, nous proposons un algorithme de poursuite de phase robuste considérant une dynamique de phase lente. Celui-ci découle d'une approximation variationnelle Bayésienne dite restreinte et nous donne un estimateur sous forme explicite. L'algorithme dit RVB préserve le caractère non linéaire de l'équation de mesure ce qui s'avère bénéfique à un faible rapport signal sur bruit. Pour analyser les performances de l'algorithme RVB, une comparaison à l'aide de données synthétiques est faite entre ce dernier et des techniques traditionnelles telles que la DPLL et la KF-DPLL. En particulier, nous les avons comparés d'une part à travers des métriques évaluant la précision de l'estimation avec erreur de phase-modulo, d'autre part à travers des métriques évaluant les sauts de cycle telles que le taux de sauts de cycle et le temps moyen avant premier saut. Pour les deux types de métriques considérés, nous avons montré les avantages de l'estimateur RVB pour les deux dynamiques de phase lente considérées (c'est-à-dire les entrées de phase de type échelon et rampe). (Notez que dans ce résumé français, seulement quelques résultats significatifs ont été présentés).

Dans le quatrième chapitre, étant donné que l'algorithme RVB au premier ordre peut ne

pas être suffisant dans des applications réalistes, nous présentons une technique de poursuite de phase robuste à une dynamique d'ordre supérieur. Cette fois, non seulement la phase mais aussi ses dérivées sont estimées. Nous appliquons ici encore l'approximation bayésienne variationnelle restreinte qui nous donne une solution explicite au problème de filtrage. Bien que le modèle d'état soit plus complexe que celui du troisième chapitre, l'expression RVB récursive a remarquablement une forme similaire à celle de l'algorithme au premier ordre. Le filtre RVB est, entre autres, non linéaire et tractable. Une comparaison utilisant des données synthétiques est faite entre l'algorithme RVB et les techniques traditionnelles telles que la DPLL et la KF-DPLL au troisième ordre. L'algorithme RVB à des ordres supérieurs montre ainsi une amélioration significative des performances en termes de précision d'estimation et d'occurrence de sauts de cycle à un faible rapport signal sur bruit.

Jusqu'à présent, les algorithmes proposés ont été évalués en utilisant uniquement des données synthétiques. Dans le cinquième chapitre, nous implémentons l'algorithme RVB à une dynamique d'ordre trois dans un récepteur radio logicielle GNSS. Ce dernier a été préalablement développé par notre laboratoire universitaire, et son architecture est décrite. En particulier, nous concentrons notre attention sur le bloc de tracking où l'algorithme RVB est implémenté et comparé aux techniques traditionnelles telles que la DPLL et la KF-DPLL. La comparaison est faite à l'aide de données GNSS réelles. Nous montrons que l'algorithme RVB exhibite très peu de sauts de cycle par rapport aux deux autres techniques traditionnelles.

Travaux futurs À la fin de cette thèse, nous pouvons établir quelques lignes directrices pour les travaux futurs sur le sujet. En particulier, nous proposons d'explorer les trois axes de recherche suivants pour l'analyse à court et à long terme.

1. *Modes de la distribution postérieure pour l'algorithme RVB à des ordres supérieurs* : Nous avons observé (bien que non montré dans le manuscrit) que l'algorithme RVB proposé peut diverger de la vraie phase en raison d'instabilités numériques dans le cas de très faibles variances a priori sur la dynamique. Ce phénomène n'est pas nouveau : un comportement similaire a été rencontré pour un algorithme de la litératreu basé également sur une approche RVB dans la cas d'une faible dynamique. Le contournement proposé consistait à remplacer localement l'estimateur MMSE par l'estimateur MAP de la phase [17]. Nous proposons donc d'étendre en premier lieu cette méthode au cas d'ordre élevé ou de trouver des méthodes alternatives pour résoudre ce problème numérique.

2. *Influence de la matrice de covariance du bruit $\boldsymbol{Q}$ sur le processus de récursivité de l'algorithme RVB* : Comme nous l'avons noté au chapitre 4, l'expression récursive de l'algorithme RVB ne dépend que de la première colonne de la matrice de covariance du bruit $\boldsymbol{Q}$, à savoir $\boldsymbol{Q}_1$. Par conséquent, la signification physique de cette dernière est alors quelque peu perdue, et un réglage empirique de ses PSD a été nécessaire. Dans notre cas, nous les avons choisies pour donner suffisamment de poids au terme d'innovation, à savoir $\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\mathrm{rvb}}]_1$. Cependant, il pourrait être intéressant d'étudier plus précisément comment chaque terme de $\boldsymbol{Q}_1$ influence le processus de mise à jour de l'estimateur.

3. *Bornes Bayésiennes d'estimation* : Dans le cadre de l'estimation de paramètres fixes et déterministes, la borne de Cramer Rao permet de donner une limite inférieure à la variance de tout estimateur non biaisé [24]. Dans notre cas d'estimation Bayésienne récurvise, il pourrait être intéressant de calculer des bornes similaires mais adaptées à notre problème (temps discret, filtre non linéaire) [25, 26].

Bibliographie

[1] Fabio Fabozzi, Stéphanie Bidon, Sébastien Roche, and Benoît Priot. "Robust GNSS Phase Tracking in Case of Slow Dynamics Using Variational Bayes Inference". In: *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. 2020, pp. 1189–1195. DOI: 10.1109/PLANS46316.2020.9110176.

[2] Fabio Fabozzi, Stéphanie Bidon, and Sébastien Roche. "Robust estimation of high-order phase dynamics using Variational Bayes inference". In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 4980–4984. DOI: 10.1109/ICASSP39728.2021.9415028.

[3] Fabio Fabozzi and Stéphanie Bidon. "Robust Phase Tracking using Restricted Variational Bayes Inference". In: *IEEE Transactions on Aerospace and Electronic Systems*.

[4] Bradford W Parkinson, Per Enge, Penina Axelrad, and James J Spilker Jr. *Global positioning system: Theory and applications, Volume II*. American Institute of Aeronautics and Astronautics, 1996.

[5] Elliott Kaplan and Christopher Hegarty. *Understanding GPS: principles and applications*. Artech house, 2005.

[6] Peter Teunissen and Oliver Montenbruck. *Springer handbook of global navigation satellite systems*. Springer, 2017.

[7] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and Elmar Wasle. *GNSS–global navigation satellite systems: GPS, GLONASS, Galileo, and more*. Springer Science & Business Media, 2007.

[8] Rodger E Ziemer and Roger L Peterson. *Digital communications and spread spectrum systems*. Macmillan New York, 1985.

[9] Mohinder S Grewal, Angus P Andrews, and Chris G Bartone. *Global navigation satellite systems, inertial navigation, and integration*. John Wiley & Sons, 2020.

[10] Roland E Best. *Phase-locked loops: design, simulation, and applications*. McGraw-Hill Education, 2007.

[11] W. C. Lindsey and Chak Ming Chie. "A survey of digital phase-locked loops". In: *Proceedings of the IEEE* 69.4 (1981), pp. 410–431. ISSN: 0018-9219. DOI: 10.1109/PROC.1981.11986.

[12] Andrew J Viterbi. "Phase-locked loop dynamics in the presence of noise by Fokker-Planck techniques". In: *The Foundations Of The Digital Wireless World: Selected Works of AJ Viterbi*. World Scientific, 2010, pp. 13–29.

[13] J. A. Lopez-Salcedo, J. A. D. Peral-Rosado, and G. Seco-Granados. "Survey on Robust Carrier Tracking Techniques". In: *IEEE Communications Surveys Tutorials* 16.2 (2014), pp. 670–688. ISSN: 1553-877X. DOI: 10.1109/SURV.2013.082713.00228.

[14] J. A. Del Peral-Rosado, J. A. López-Salcedo, G. Seco-Granados, J. M. López-Almansa, and J. Cosmen. "Kalman filter-based architecture for robust and high-sensitivity tracking in GNSS receivers". In: *2010 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)*. 2010, pp. 1–8. DOI: 10.1109/NAVITEC.2010.5708005.

[15] JONG-HOON WON, Dominik Dötterböck, and Bernd Eissfeller. "Performance Comparison of Different Forms of Kalman Filter Approaches for a Vector-Based GNSS Signal Tracking Loop". In: *Navigation* 57.3 (2010), pp. 185–199.

[16] Ross Kindermann. "Markov random fields and their applications". In: *American mathematical society* (1980).

[17] Stéphanie Bidon and Sébastien Roche. "Variational Bayes phase tracking for correlated dual-frequency measurements with slow dynamics". In: *Signal Processing* 113 (2015), pp. 182–194.

[18] Rudolph Emil Kalman. "A new approach to linear filtering and prediction problems". In: *Journal of basic Engineering* 82.1 (1960), pp. 35–45.

[19] Mark L Psiaki, Dennis M Akos, and Jonas Thor. "A comparison of direct RF sampling and downconvert & sampling GNSS receiver architectures". In: *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)*. 2003, pp. 1941–1952.

[20] Mark L Psiaki. "Block acquisition of weak GPS signals in a software receiver". In: *Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2001)*. 2001, pp. 2838–2850.

[21] Huaqiang Shu, Eric Pierre Simon, and Laurent Ros. "Third-order kalman filter: Tuning and steady-state performance". In: *IEEE Signal Processing Letters* 20.11 (2013), pp. 1082–1085.

[22] Sébastien Roche and Stéphanie Bidon. "On the comparison between KF-PLL and DPLL". unpublished. 2021.

[23] S. A. Stephens and J. B. Thomas. "Controlled-root formulation for digital phase-locked loops". In: *IEEE Transactions on Aerospace and Electronic Systems* 31.1 (1995), pp. 78–95. ISSN: 0018-9251. DOI: 10.1109/7.366295.

[24] Steven M Kay. *Fundamentals of statistical signal processing*. Prentice Hall PTR, 1993.

[25] P. Tichavsky, C.H. Muravchik, and A. Nehorai. "Posterior Cramer-Rao bounds for discrete-time nonlinear filtering". In: *IEEE Transactions on Signal Processing* 46.5 (1998), pp. 1386–1396. DOI: 10.1109/78.668800.

[26] Carsten Fritsche, Emre Özkan, Lennart Svensson, and Fredrik Gustafsson. "A fresh look at Bayesian Cramér-Rao bounds for discrete-time nonlinear filtering". In: *17th International Conference on Information Fusion (FUSION)*. IEEE. 2014, pp. 1–8.

Chapter 1

Introduction to GNSS

This chapter introduces the thesis context. Fundamentals of navigation satellite system including classical satellite signal structure are presented. We concentrate then our attention on the receiver tracking loops. In particular, the measurements they usually provide are described, while focusing on the advantages that the carrier-phase measurements provide. Finally, we conclude by highlighting issues related to the latter, opening the doors to the second chapter.

Contents

1.1 Principle of GNSS and satellite signal

Satellite navigation has become an essential part of our modern society and is used by people over the entire globe. The term *Navigation* is defined as the science of getting a craft or person from one place to another [4]. It is present, not only in daily routine chores, but also multiple domains of application from surveying and geodesy to machine guidance and precision agriculture and many others.

Before 2000, practically only one system was fully operational and available, the *Global Positioning System* (GPS) built up by the United States. However, during the times of Cold

War in the 1970s and 1980s, the Russian developed their own system called GLONASS. Only at the beginning of the new millennium, Europe decided to design its own global satellite system called Galileo. Today, we count four global satellite navigation systems available (GPS — USA, GLONASS — Russia, BeiDou — China, Galileo — Europe), as well as two regional systems (IRNSS/NavIC — India, QZSS — Japan). Differences exist between these systems, however, fundamentals of navigation remain the same. The latter are explained in the next section.

1.1.1 Principle of GNSS

Global Navigation Satellite System (GNSS) is the general concept used to identify systems that allow user localization based on a constellation of satellites. These systems rely on the same principle: the user computes its position through measured distances between the set of visible satellites and the receiver. These distances are calculated while estimating the propagation time from transmitted signals by a satellite to the user receiver. The propagation time is then multiplied by the speed of the signal (*e.g.*, speed of light) to obtain the satellite-to-receiver distance. By measuring the propagation time of the signal broadcasted from multiple satellites at known locations, the receiver can determine its position. The procedure is referred to as trilateration which is explained in the next paragraph. Useful GNSS textbook references can be found in the literature, for instance [5, 4, 6, 7].

1.1.1.1 Fundamentals of satellite navigation

As mentioned before, the receiver estimates the time $\tau_{\mathrm{r}}^{\mathrm{s}}$ necessary for the GNSS signal to propagate from the satellite antenna (s) to the receiver antenna (r). Since the signal travels at the speed of light, c, the propagation time can be converted in distance through a simple multiplication

$$\rho_{\mathrm{r}}^{\mathrm{s}}(t) = c\tau_{\mathrm{r}}^{\mathrm{s}}. \tag{1.1}$$

For the sake of simplicity, we assume for the time being that the satellite and receiver clocks are synchronized, and that the atmosphere, which delays the GNSS signal propagation, is not present. Under these circumstances, the observation equation is

$$\rho_{\mathrm{r}}^{\mathrm{s}}(t) = ||\boldsymbol{r}_{\mathrm{r}}(t) - \boldsymbol{r}^{\mathrm{s}}(t-\tau)||, \tag{1.2}$$

where $||\cdot||$ is the Euclidian norm. $\boldsymbol{r}_{\mathrm{r}} = (x_{\mathrm{r}}, y_{\mathrm{r}}, z_{\mathrm{r}})^{\intercal}$ is the unknown position vector of the receiver antenna and $\boldsymbol{r}^{\mathrm{s}} = (x^{\mathrm{s}}, y^{\mathrm{s}}, z^{\mathrm{s}})^{\intercal}$ the satellite antenna position which is known from the navigation message (see section 1.1.2.1). With a single range measurement (1.2), the position of the receiver lies on a sphere, centered on the satellite and of radius equal to the measured range, ρ_{r}^{1}. With a double range measurements (1.2), the receiver position lies on the so-called line of position which is formed by loci of intersections points between two spheres. Finally, a third measurement (1.2) gives us a third sphere which intersects the previous two in two points. One of these two points can be immediately dismissed since it is far out of

receiver location. Therefore, simultaneous measurements from three satellite are sufficient to determine the position of a fixed receiver in three dimensions. This procedure is also well-known as trilateration technique [8] and is represented in Fig. 1.1. The trilateration is

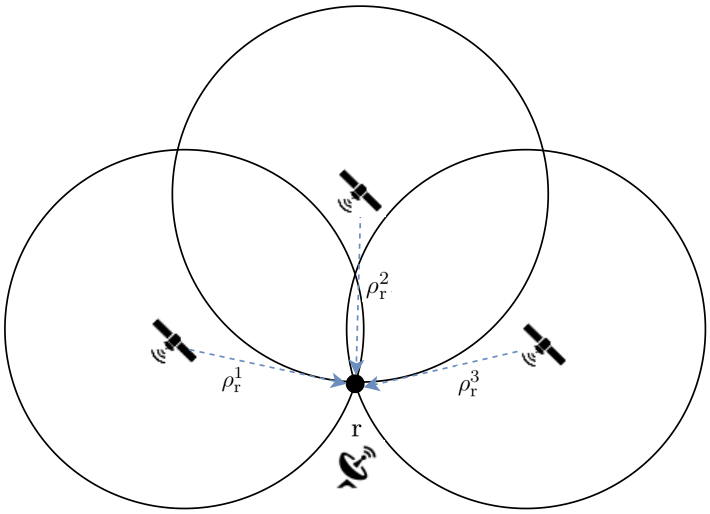


Figure 1.1: Positioning through intersecting spheres - Trilateration

a problem of simple geometry and a solution algorithm is given by [9]. Mathematically, the solution of receiver position $(x_r, y_r, z_r)^\intercal$ using the simultaneous range equations ρ_r^1, ρ_r^2, and ρ_r^3 is usually obtained through a linearization procedure [7, 4, 10, Chap.1]. In fact, the system formed by the three nonlinear equations (1.2) can be approximated at a first order as follows

$$\boldsymbol{p} = \boldsymbol{p}_0 + \boldsymbol{A}\Delta\boldsymbol{x}, \tag{1.3}$$

where, for simplicity and compactness, a vector formalism is used and the indices are dropped. $\boldsymbol{p} = [\rho_r^1, \rho_r^2, \rho_r^3]^\intercal$ and $\boldsymbol{p}_0$ is the vector of the computed range values given an initial estimate $\boldsymbol{x}_0 = (x_{r,0}, y_{r,0}, z_{r,0})^\intercal$ and satellite coordinates (x^i, y^i, z^i) with $i = 1, 2, 3$. Furthermore, the design matrix $\boldsymbol{A}$ is

$$\boldsymbol{A} = \begin{pmatrix} \frac{\partial\rho_r^1}{\partial x_r} & \frac{\partial\rho_r^1}{\partial y_r} & \frac{\partial\rho_r^1}{\partial z_r} \\ \frac{\partial\rho_r^2}{\partial x_r} & \frac{\partial\rho_r^2}{\partial y_r} & \frac{\partial\rho_r^2}{\partial z_r} \\ \frac{\partial\rho_r^3}{\partial x_r} & \frac{\partial\rho_r^3}{\partial y_r} & \frac{\partial\rho_r^3}{\partial z_r} \end{pmatrix}, \tag{1.4}$$

with $\frac{\partial\rho_r^i}{\partial x_r} = \frac{(x_{r,0} - x^i)}{\rho_{r,0}^i}$ $(i = 1, 2, 3)$, and $\Delta\boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_0$ is the increment to the initial vector of receiver coordinates that is to be determined. Solving for $\Delta\boldsymbol{x}$

$$\Delta\boldsymbol{x} = \boldsymbol{A}^{-1}(\boldsymbol{p} - \boldsymbol{p}_0), \tag{1.5}$$

and then

$$\boldsymbol{x} = \boldsymbol{x}_0 + \Delta\boldsymbol{x}. \tag{1.6}$$

However, in reality, the clock in the GNSS receiver is not synchronized with the clocks in the satellite — as assumed at first instance. The range measurements estimated by the receiver are biased by the receiver and satellite clock errors, dt_r and dt^s, and are therefore referred to as *pseudoranges*

$$p_\mathrm{r}^\mathrm{s} = \rho_\mathrm{r}^\mathrm{s} + c(dt_\mathrm{r} - dt^\mathrm{s}). \tag{1.7}$$

A timing error of a millisecond could bring to an error in position of about 300 km [7, Chap.1]. Thus now four unknown parameters are present in the pseudorange observation equations. Therefore, we need at least four simultaneous pseudoranges to estimate the three receiver coordinates and the receiver clock offset (measured in units of distance). With $\boldsymbol{x} = (x_\mathrm{r}, y_\mathrm{r}, z_\mathrm{r}, dt_\mathrm{r})^\intercal$ and the four-by-four design matrix

$$\boldsymbol{A} = \begin{pmatrix} \frac{\partial \rho_\mathrm{r}^1}{\partial x_r} & \frac{\partial \rho_\mathrm{r}^1}{\partial y_r} & \frac{\partial \rho_\mathrm{r}^1}{\partial z_r} & 1 \\ \frac{\partial \rho_\mathrm{r}^2}{\partial x_r} & \frac{\partial \rho_\mathrm{r}^2}{\partial y_r} & \frac{\partial \rho_\mathrm{r}^2}{\partial z_r} & 1 \\ \frac{\partial \rho_\mathrm{r}^3}{\partial x_r} & \frac{\partial \rho_\mathrm{r}^3}{\partial y_r} & \frac{\partial \rho_\mathrm{r}^3}{\partial z_r} & 1 \end{pmatrix}, \tag{1.8}$$

the same iterative procedure as described before can then be applied.

Note that not all the error sources (*e.g.*, ionosphere, troposphere, relativity effects, etc.) are considered. A more detailed pseudorange equation is shown in section 1.3. Therefore, due to these unmodeled errors, it is helpful to use simultaneous pseudoranges to higher-than-four available satellites for estimating the receiver coordinates and clock offset. This may require the use of a *least-squares estimation* procedure. For further details on this topic, the following textbook reference is recommended [7, Chap.22]

1.1.1.2 GNSS system segments

To make a GNSS work, a proper infrastructure has to be built. The latter is shown in Fig. 1.2 and is composed by the following three parts.

- The *space segment* consists of a nominal constellation of operating satellites that transmit signals that give the current GNSS satellite coordinates, time and other useful information (*e.g.*, ephemeris, almanac; more details are given in section 1.1.2.1). The orbit altitude is about 20,000 km above the Earth, which is the so-called *Medium Earth orbit* (MEO).

- The *ground segment* is formed by a ground-based network of master control stations, data uploading stations and monitor stations. This segment is responsible for the monitoring and operation of the space segment and includes civilian tracking networks that provide the user segment with reference control and update of parameters in the navigation message. Finally, it can also provide user's position corrections to the user segment.

- The *user segment* consists of the receiver equipment, which receives the signals from the satellites and calculates the user's three-dimensional position. The equipment ranges from smartphones to specialized receivers used for high-end survey and mapping applications.

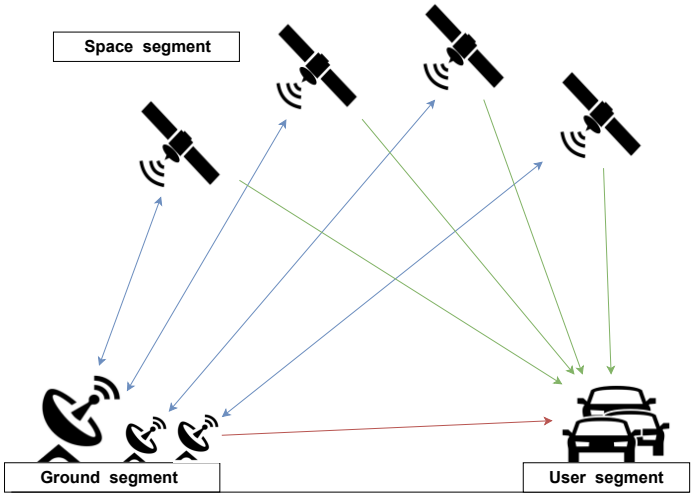


Figure 1.2: GNSS infrastructure

1.1.2 Signal structure

For most of navigation systems, a general signal model consists of a *Direct-Sequence Spread-Spectrum* (DS-SS) signal [11, 12, Chap.5], which is synchronously transmitted by all the satellites in the constellation. *Spread spectrum* systems are commonly used for communications, since they are more resistant to jamming. The basic concept is that the information waveform with small bandwidth is converted (or spread) by modulating it with a large-bandwidth waveform [11, Chap.5]. Additionally, DS-SS indicates a spread-spectrum signal generated by the direct mixing of the data with a spreading waveform before the final carrier modulation [13]. In GNSS, DS-SS enables the *Code Division Multiple Access* (CDMA) transmission: at each satellite signals is associated different codes (more details are described as follows). The section is divided in two parts: transmitted and received signal.

1.1.2.1 Transmitted signal

At a glance, the GNSS signal is composed by three main components: a *Pseudo-Random Noise* (PRN) code, a navigation message and carrier signal. The transmitted signal by the i-th satellite is expressed as

$$s_{\mathrm{T}}^{i}(t) = \sqrt{P^{i}} d^{i}(t) c^{i}(t) e^{j2\pi f_{c}^{i} t}, \tag{1.9}$$

where

- P^i is the transmitted power, considered equal for all satellite and elevation-dependent [14, Chap.5];
- $c^i(t) \in \{-1, 1\}$ is the PRN code which identifies the transmitter satellite;
- $d^i(t) \in \{-1, 1\}$ is the navigation message which is a low rate data-link and contains useful parameters to calculate the receiver position;
- f_c^i is the carrier frequency.

A time-plot of the GPS signal is shown in Fig. 1.3. For the sake of simplicity, the carrier, PRN code and navigation message are simulated at lower frequencies with respect to (wrt) the real ones. In case of Galileo signals, an additional modulation is applied to the signal, namely the so-called BOC (*Binary Offset Carrier*). Its purpose is to minimize the signal interference between GNSS systems and consists in the multiplication of the PRN sequence with a square wave sub-carrier [15, 16, 17].

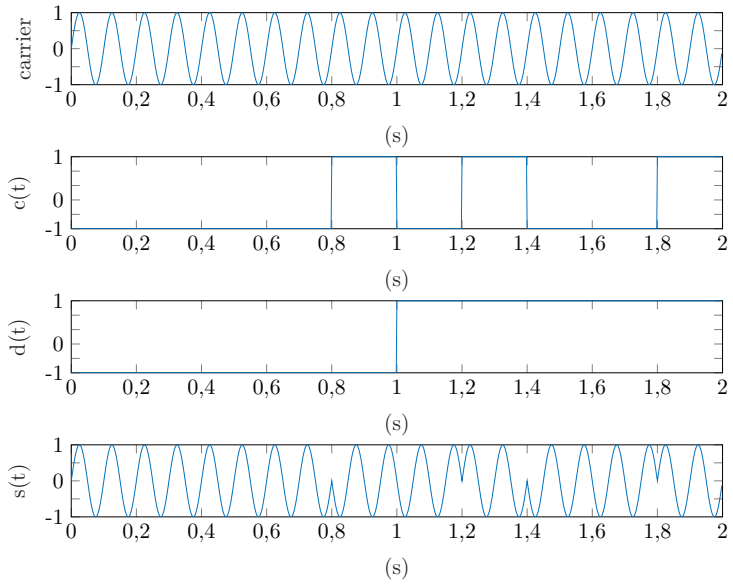


Figure 1.3: GPS signal structure

Navigation message

The navigation message $d(t)$ in (1.9) provides all the necessary information to allow the user to perform the positioning service [4]. It includes the ephemeris parameters, needed to compute the satellite coordinates with enough accuracy, the time parameters and clock corrections, to compute satellite clock offsets and time conversions, the service parameters with satellite health information, and the almanacs, which allow the position computation of all satellites in the constellation. In case of GPS L1 C/A (*Coarse Acquisition*) signal, which is widely used, the current navigation message is modulated on both carriers at 50 bit-per-second (each bit

is transmitted every 20 milliseconds). The whole message contains 25 pages (or frames) of 30 seconds each, forming the master frame that takes 12.5 minutes to be transmitted [18, 4]. Each sub-frame contains specific information. Finally, in case of absence of navigation messages, the signal is dubbed *pilot signal*. The latter is the type of signal we use for our carrier-phase tracking problem. However, note that the use of combined data/pilot channels for the carrier-phase tracking is applicable and studied in the literature [19, 20].

Pseudorandom noise codes

As mentioned before, most of GNSS signals use the CDMA technique to multiplex several satellite signals onto the same frequency [21]. The basic concept behind the CDMA schemes is that each satellite is assigned with a PRN (*Pseudo-Random Noise*) code that modulates the transmitted signal. The spreading waveform is a contiguous series of rectangular pulses generated using a deterministic, digital PRN. Within the receiver, a local copy of the PRN sequence is generated, which is continuously compared and aligned with the signal received from the satellites. PRN codes are described into the "Interface Control Document", such as the *Open Service Signal-In-Space Interface Control Document* (SIS ICD) for Galileo [22].

For instance, two types of PRN code are used for the GPS: 1) code C/A — its frequency is at 1.023 MHz and its period $T_{C/A}$ is equal to 1 millisecond. 2) code P (*Precise*)— its frequency is at 10.23 MHz and its period T_P lasts one week. The Galileo PRN codes have the same generating method as the GPS, however, Galileo uses also a *secondary code*. Adding the latter helps data bit synchronization, since longer spreading sequences improve the correlation properties of the spreading code [23, 24]. Moreover, the secondary code improves the signal acquisition performance in Galileo receivers, achieving longer integration for pilot channels than one primary code length [25].

Carrier signal

The carrier frequency impacts various aspects of the satellite navigation signals. Its influence goes from the propagation behaviors including Doppler effect and atmospheric interference to hardware components. Nowadays, every navigation system uses a frequency band between 1-2 GHz, which is termed L-band. In this band, different sub-frequency bands have been allocated and made available by the *International Telecommunications Union* (ITU). The advantages of L-band especially concerns the propagation conditions which include significant attenuation of atmospheric effects [7, Chap.4]. Furthermore, the antenna size for signals in L-band is quite limited which is beneficial for miniaturized satellite navigation receivers, an essential aspect for mobile applications.

1.1.2.2 Received signal

Once transmitted, the signal propagates through a channel which modifies its amplitude, phase/frequency and delay. Various are the propagation errors from the atmosphere and relativistic effects to multi-path [4, Chap.6]. In this section, we focus our attention on the so-called *Doppler effect*, which is the most interesting error source related to our thesis context. The name takes place by his inventor C. Doppler, who discovered it in mid 19^{th} century as reported in his manuscript [26]. The Doppler effect is due to the relative motion between transmitter and receiver, which causes a frequency shift in the received signal wrt the transmitted one. Assuming the arrival of only direct signal (or *Line-Of-Sight Signal* — LOSS) to the receiver antenna and a single satellites in view, the received signal consists of an attenuated, time-delayed, phase shifted replica of the transmitted one in (1.9). The latter can be simplified with

$$s_{\mathrm{T}}(t) = c(t)e^{j2\pi f_c t}, \tag{1.10}$$

where the amplitude $\sqrt{P}$ and the navigation data $d(t)$ are disappeared, therefore the received signal is

$$s_{\mathrm{R}}(t) = c(t - \tau(t))e^{j2\pi f_c(t-\tau(t))}, \tag{1.11}$$

where $\tau(t)$ is the excess delay at the time t. Approximating the delay τ as a first-order Taylor expansion with constant radial velocity v_{rad}, namely $\tau(t) \approx \tau_0 + {}^{v_{\mathrm{rad}}t}/_{c}$, we obtain

$$s_{\mathrm{R}}(t) \approx c((1 - {}^{v_{\mathrm{rad}}}/_{c})t - \tau_0)e^{j2\pi f_c(1-{}^{v_{\mathrm{rad}}}/_{c})t+j\phi_0}, \tag{1.12}$$

where the Doppler frequency is

$$f_d = -f_c\frac{v_{\mathrm{rad}}}{c}, \tag{1.13}$$

and $\phi_0 = -j2\pi f_c\tau_0$. Note that the Doppler frequency increases if the source is closing and decreases if the source recedes [27]. The general expression of (1.13) can be readily found in the literature, therefore we omit its formulation [4, Par.2.5]. To compensate for the Doppler effect, the frequency must be adjusted to keep both phases of receiver and satellite synchronized. This frequency adjustment is done by the tracking loop and is reported by the receiver as the Doppler measurement [28, p.467].

Finally, it needs to be said that the Doppler information has broad applications in signal processing and communications. In case of GNSS applications, it is used to discriminate signals from various GNSS satellites, to evaluate the accuracy of GNSS speed measurement [29], to monitor signal integrity and to detect spoofing [27], among other various applications. However, its most important use is for the determination of user's velocity vector through the received carrier frequency [30, Chap.6].

1.2 GNSS receiver architecture

One of the most important factor to design a GNSS receiver is knowing the specific application the receiver is going to be used for. Therefore, many receivers design characteristics exist and

differ from each other, however the main core remains the same [30, Chap.6].

GNSS receivers are now under development and production for over four decades [7, Chap.13]. In general, the development guideline of GNSS receiver technology is mostly correlated with the history of the specific satellite navigation program and with technology progress. Over the years, digital receivers have taken over the single-analog hardware channel. Most receivers have a front end based on analog technology, while the signal is digitized after with ADCs (*Analog-Digital Converter*). Therefore, the post-correlated signals, pseudo-range, carrier-phase measurements, and demodulated navigation data are processed in the digital domain with a navigation processor. For the tracking process, a parallel multichannel structure is used, which makes it possible to simultaneously track multiple satellites for re-acquisition as well as for searching next a satellite in view. Finally, in recent years, new trends following the *Software Defined Radio* (SDR) receiver philosophy emerged [31, 32]. In this case, the down-conversion and other processes, typically performed in analog domain, are now implemented in a digital platforms such as *Field Programmable Gate Array* (FPGA) and *System-On-Chip* (SOC). The aim of GNSS SDR receiver is to simplify the front end chain by digitizing closer and closer to the antenna. The interest of the GNSS community in using SDR receivers is increasing in recent times [14, 33, 34].

In this section, we analyze a simplified version of a generic GNSS receiver. The latter is shown in Fig. 1.4. In particular, we present details of its three main parts: the front end chain, the consecutive signal processing block, and finally the navigation solution.

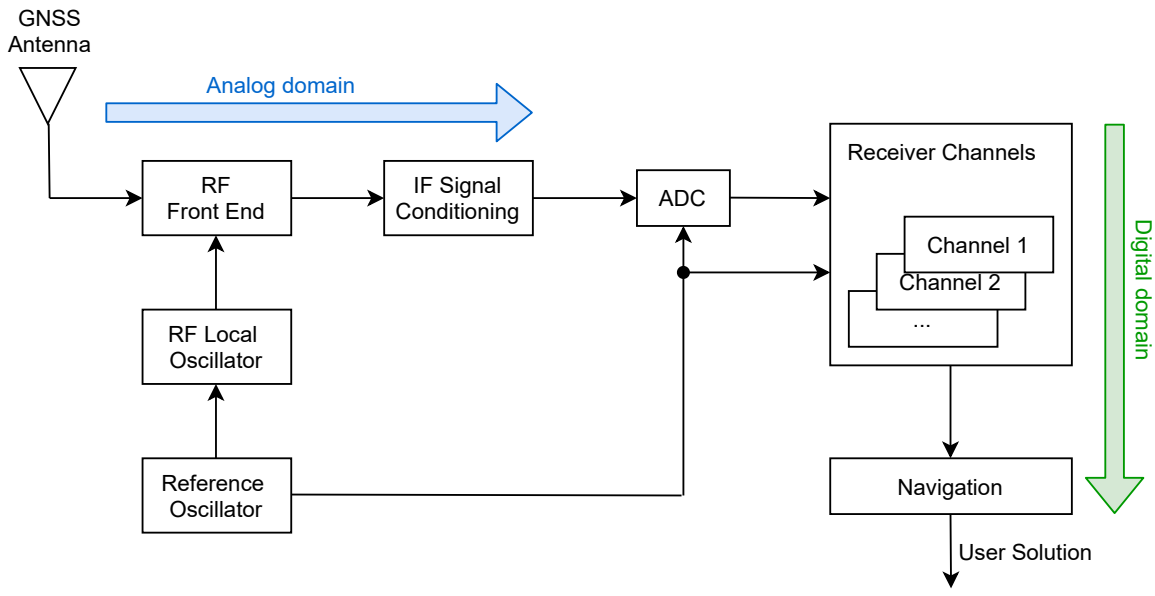


Figure 1.4: Simplified version of a generic block diagram GNSS receiver [30, Chap.6]

1.2.1 From the antenna to the discrete-time baseband signal

The role of the antenna is to receive an electromagnetic wave and convert it to an electronic signal [7, Chap.13]. The goal in satellite navigation is to work with an all-in-view constellation, and in this context, a single antenna is sufficient for a standard GNSS receiver operation. However, the use of multiple antenna has shown various benefits such as multipath mitigation [35] and spoofing mitigation [36].

Immediately after the antenna terminals, the RF-FE (*Radio Frequency Front End*) unit takes place. Its aim is to filter and down-convert the incoming signal. In a modern GNSS receiver, the RF unit is a critical subsystem, because it has a strong impact on the quality of the signal reception. A simplified version of RF-FE architecture is shown in Fig. 1.5.

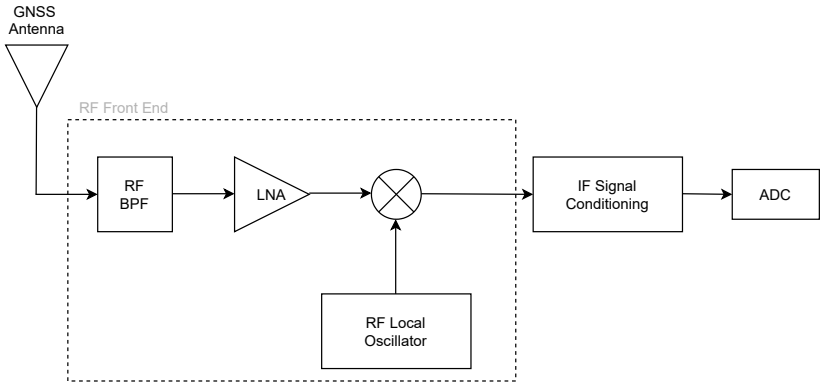


Figure 1.5: Front End and IF signal conditioning [30, Chap.6]

For high-quality RF front end circuits, an *RF Bandpass Filter* (BPF) is often placed directly after the antenna. This RF passive BPF can be used to reduce out-of-band interference without degradation of the GPS signal waveform. Moreover, it helps prevent the *Low Noise Amplifier* (LNA) from saturation, which is placed soon after. The role of the latter is to boost the received signal so that it can be used for additional processing. Usually, the amplification done by the LNA is of around 35-55 dB of gain [30, Chap.6]. Afterwards, a down-conversion is accomplished by multiplying the incoming signal with a sinusoid (local oscillator signal) using a mixer.

The local oscillator frequency is either larger or smaller than the GNSS carrier frequency by an amount equal to the IF (*Intermediate Frequency*) stage. Typically, the IF signal is selected to be the difference between the signal and local oscillator frequencies [30, Chap.6]. Finally, the IF signal is digitized by an ADC which discretizes the signal stream into a set of samples. Afterwards, samples are quantized by a specific number of bits. Many low-costs receivers use the 1-bit quantization of the digitized samples [30, Chap.6]. However, higher quantization levels provides lower noise level at the price of increasing the computational complexity of receiver's design [37].

1.2.2 Baseband signal processing

In conventional receivers, baseband signal processing refers to any algorithms which acquire and track the incoming GNSS signal, provide code and phase measurements, and finally extract the navigation data. As mentioned before, these outputs provide information concerning the relative distance between the receiver and the corresponding satellite, which then is used in the computation of the *Position, Velocity and Timing* (PVT) solution. As underlined before, the receiver has usually dedicated units to process the signal from the individual satellite and frequency bands — these units are called channels, as shown in Fig. 1.4. In this section, we describe the two main stages of the baseband signal processing: acquisition and tracking.

1.2.2.1 Acquisition

In a first instance, the receiver has to detect the satellites in view and obtain a rough estimation of their synchronization parameters such as the time-delay τ and Doppler shift f_d. This stage is the so-called *acquisition*. It is performed through a correlation between the incoming signal and a local generated replica. The aim of the correlation is to bring the signal of the considered satellite above the noise floor, separates it from the other satellite signals and provides an estimate of the signal parameters. Since the local replica is shifted in time and frequency, a two-dimensional search is performed where the maximum provides an estimate of the synchronization parameters. Computationally, using (1.9) in (1.12), the locally generated replica signal for a single satellite at the IF stage (after the ADC) can be written as

$$\hat{r}_{\mathrm{IF}}[k] = c(T_s k - \hat{\tau}) e^{j(2\pi(f_{\mathrm{IF}} + \hat{f}_d)T_s k)}, \tag{1.14}$$

where T_s is the sampling time interval such that $t = kT_s$, which is usually shorter or equal to the navigation data bit/symbol period. $k = 1, \ldots, K$, where K is the number of samples used in the correlation. Referring to Fig. 1.5, the subscript IF means that the incoming signal passed the RF-FE, therefore it has been filtered, down-converted to the IF signal conditioning block and then sampled through the ADC. Note that the notation $\hat{\cdot}$ is intentionally used in (1.14) because the locally generated replica signal depends on the estimated synchronization parameters $\hat{\tau}$ and $\hat{f}_d$. The correlator compares the received signal with the locally generated replica given a coherent integration time of K samples, *i.e.,*

$$g(\hat{\tau}, \hat{f}_d) = \sum_{k=1}^{K} s_{\mathrm{R_{IF}}}[k] \hat{r}^*_{\mathrm{IF}}[k], \tag{1.15}$$

where $s_{\mathrm{R_{IF}}}$ is the received signal at IF stage. The notation * indicates the complex conjugate operator. Therefore, (1.15) is tantamount to a two-dimensional search of both the code phase and Doppler. If the peak magnitude of this function exceeds a certain threshold, then the satellite's signal is declared to be present and the position of the peak is given by the coarse estimates $\hat{\tau}$ and $\hat{f}_d$. The value chosen for the threshold is a compromise between the conflicting goals of maximizing the probability of detection and minimizing the probability of false alarm. Several alternatives exist to perform the two-dimensional search such as *Sequential Search*

acquisition and *Parallel Search parameters* [30, Chap.6]. To gain some sensitivity and increase the detection robustness, a search and decision logic is often used [38].

1.2.2.2 Tracking

Once detected the satellite's signal, the receiver switches to the tracking mode. In addition to the synchronization with the navigation message, its objective is to keep track of synchronization parameters of the detected satellites (*e.g.*, time-delays, Doppler-shifts and carrier-phase) [30, Chap.6]. Time-delay is typically tracked by means of a *Delay Locked Loop* (DLL), which basically performs the same correlation operation as done in acquisition (see section 1.2.2.1) but with lower computational complexity due to the absence of a searching space of the parameters. Likewise, *Digital Phase Locked Loop* (DPLL) and *Frequency Lock Loop* (FLL) are used for phase and frequency tracking, respectively.

A block diagram of a GNSS tracking architecture of a single-channel is shown in Fig. 1.6. In this example, two tracking loops are used, one refers to the code and another to the carrier

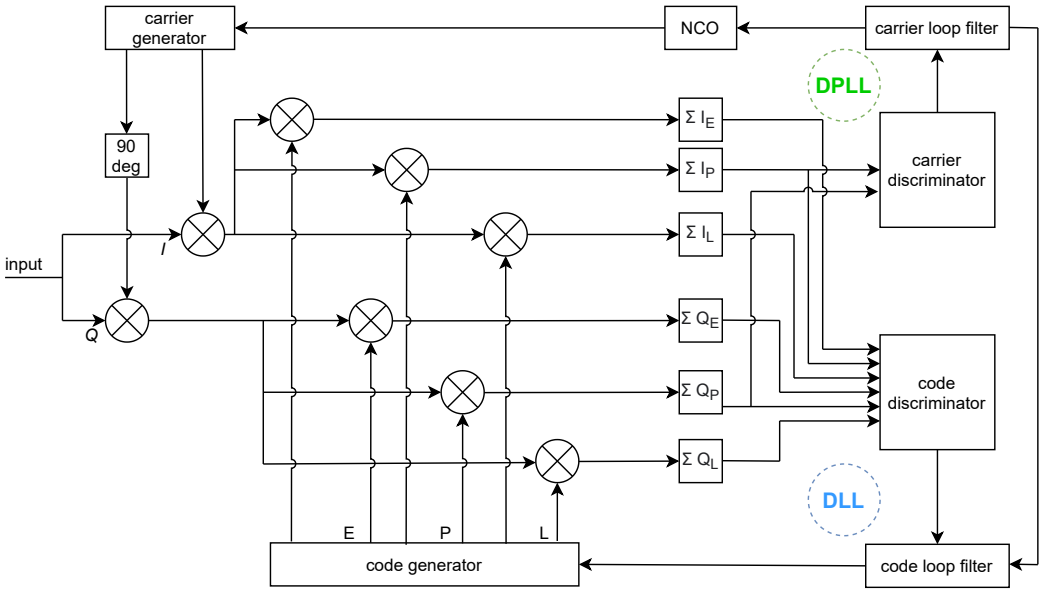


Figure 1.6: A generic block diagram of code and carrier tracking loops [11]

phase tracking. Each loop is composed of correlator, discriminator, loop filter and *Numerical Control Oscillator* (NCO) (each of these blocks will be discussed in section 2.1.1 for a DPLL). The carriers are wiped off by the replica carrier signals to produce the I (*In-phase*) and Q (*Quadrature*) signal components. The latter are then correlated with the replica codes at *Early* (E), *Prompt* (P), and *Late* (L) branches. Usually, the correlator output at P branches is used in the carrier-phase tracking whereas the correlator output at E and L branches are used in the code tracking. In particular, the I and Q components are used in the carrier discriminator to determine the carrier angle residual (*i.e.*, phase $\delta\phi$) of the incoming signal wrt the I replica signal of the P branch. For code tracking, the combination of outputs at

E and L branches (*e.g.*, early-minus-late) is used in the code discriminator to estimate the time-delay residual $\delta\tau$ of the incoming codes wrt the replica codes. An example of E, P, L correlation is shown in Fig. 1.7. When the replica signal is successfully aligned with the

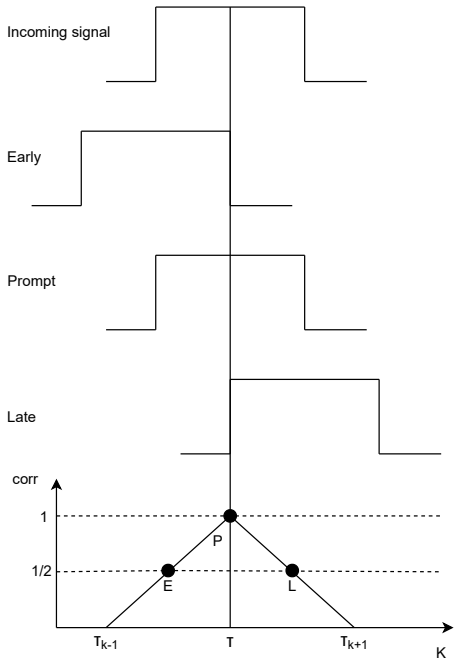


Figure 1.7: Locked code — Maximum correlation

incoming signal, the vector sum of I and Q at E and L branches is balanced for the DLL, and the absolute value of I at the P branch is maximum for the DPLL. In this case, the sign of I represents exactly the navigation data bit/symbol. Finally, to ensure that the tracking loops are correctly following the incoming signal, code and phase lock detectors (or indicators) are often used [4, 5]. Although basic receivers use three correlators to track the incoming signal, one can thought of an increased number of correlators to extract further information as with the *Multipath Estimating Delay Lock Loop* (MEDL) [39] and BOC modulations (Galileo signals) [40]. However, this is at the cost of more power and resource consumption.

Since the DPLL will be widely discussed in chapter 2, a few words are here spent for the DLL. It needs to be said that several DLL alternatives can be found in literature [41], depending on: the space between E/L and P branches (so-called δ-spacing), how the EPL are combined to form the discriminator function, and so on. Moreover, different DLL-based architectures have been designed to mitigate the multipath effect [42, 43, 44].

1.2.3 Navigation

The navigation block presented in Fig. 1.4 is in charge of calculating the observables (or measurements), demodulating the navigation message and finally computing the user location. The observables involve the time-delay or phase-differences estimates which are the outputs

of the tracking algorithms. As already mentioned in section 1.1.1.1, since the incoming signals are modified by channel propagation errors (*e.g.*, atmosphere effects, multi-path, clock biases, etc), the range is typically referred to as pseudorange. The latter and other parameters from the navigation message (such as satellites' orbits and clocks bias) are then processed to calculate the user's position. Finally, the computed PVT solution is forwarded to a receiver user's interface. In some cases, redundant pseudorange measurements are used to enhance the integrity of the solution by *Receiver Autonomous Integrity Monitoring* (RAIM) [45].

1.3 GNSS measurements

The primary measurements used to estimate user's position are:

- The pseudorange (or code) measurements between satellite and receiver.
- The carrier-phase (or carrier pseudorange) measurements of the received signal.

These two values are computed for each tracked GNSS signal by the receiver and reflect its true position and clock behavior. Both measurements are described in this section. Note that we mostly focus our attention on carrier-phase measurements, since it is of our primary interest in our thesis context.

1.3.1 Pseudorange measurements

As mentioned before (see section 1.1.1.1), the pseudorange is a measure of the distance between the satellite and the receiver's antenna obtained by estimating the propagation time between both. The latter is assessed from a maximum correlation analysis between the receiver code and the GPS signal; it is done in practice by a DLL (see section 1.2.2.2). However, the measured pseudorange is different from the geometric distance between the satellite and the receiver's antenna because of clock errors and the other related inaccuracies.

Starting from (1.7) and taking the ionospheric, tropospheric effects, multipath and relativistic effects as well as remaining errors into account, the pseudorange model for the i-th satellite can be completed by

$$p_{\mathrm{r}}^i = \rho_{\mathrm{r}}^i + c(dt_{\mathrm{r}} - dt^i + dt_{\mathrm{rel}}^i) + I_c^i + T_c^i + m_c^i + \epsilon_c^i, \tag{1.16}$$

where I_c^i and T_c^i denote the ionospheric and tropospheric effects on the code, m_c^i denotes the multipath effects, and finally, dt_{rel}^i summarizes the relativistic clock correction and the relativistic signal delay due to space-time curvature. The term ϵ_c^i considers all the remaining errors (*e.g.*, measurement noise, Earth tide and ocean loading tide effects). Note that another important effect to model the geometric term ρ_{r}^i is the rotation of the Earth reference frame during signal propagation. It is referred to as Earth rotation correction (or Sagnac correction).

Considering the altitude of the MEO satellites and the Earth rotation (about 15 arcsec sec^{-1}), the related Earth rotation correction is about 1 arcsec [46]. The impacts of such a correction depend on the latitude of the station (at the equator, 1 arcsec rotation is equivalent to about 31 meters position displacement) [47, Chap.4].

1.3.2 Carrier-phase measurements

A receiver not only provides pseudorange measurements but carrier-phase measurements via the DPLL. The latter aligns the incoming carrier from the satellite with the replica, and measures the fractional phase shift of both signals. Due to the short wavelength of the carrier-phase (approximately 19-25 cm), the carrier-phase measurement is much more precise than the code measurement ($\approx$ 2 order of magnitude [7, Chap.1]). This advantage comes at the cost of an ambiguous range measurement unlike with code measurement. In fact, if we consider an error-free scenario without motion between satellite and receiver and a perfect clock synchronization, the carrier-phase measurement would remain fixed at each fraction of a cycle. In this case, the distance between the satellite and the receiver would be an unknown number of whole cycles plus the measured fractional cycle [28, 48, Chap.5]. The ambiguous integer number of cycles in the carrier-phase measurement is called *ambiguity*.

In the case of a vacuum medium and an error-free situation, the measured phase can be presented by

$$\phi_{\text{r}}^{i} = \phi_{\text{r}} - \phi^{i} + N_{\text{r}}^{i} \triangleq \Delta\phi_{\text{r}}^{i} + N_{\text{r}}^{i}, \tag{1.17}$$

where ϕ_{r}^{i} is the received phase from the i-th satellite, ϕ_{r} is the phase of receiver's oscillator and ϕ^{i} is the phase of i-th satellite. N_{r}^{i} denotes the ambiguity number related to receiver and satellite i. (1.17) can also be written as [47, Chap.4]

$$\phi_{\text{r}}^{i} = \frac{\rho_{\text{r}}^{i} f_c}{c} + N_{\text{r}}^{i} = \frac{\rho_{\text{r}}^{i}}{\lambda} + N_{\text{r}}^{i}, \tag{1.18}$$

where λ is the carrier wavelength. Taking now the ionospheric, tropospheric, multipath and relativistic effects as well as remaining errors into account, the carrier-phase model can be completed by

$$\phi_{\text{r}}^{i} = \Delta\phi_{\text{r}}^{i} + 2\pi f_c^{i}(\delta t_{\text{r}} - \delta t^{i} + dt_{\text{rel}}^{i}) + I_{\phi}^{i} + T_{\phi}^{i} + m_{\phi}^{i} + \lambda N_{\text{r}}^{i} + \epsilon_{\phi}^{i}, \tag{1.19}$$

where I_{ϕ}^{i} and T_{ϕ}^{i} denote the ionospheric and tropospheric effects on the carrier-phase, m_{ϕ}^{i} denotes the multipath effects, and finally, dt_{rel}^{i} summarizes the relativistic clock correction and the relativistic signal delay due to space-time curvature. The term ϵ_{ϕ}^{i} considers all the remaining errors (*e.g.*, measurement noise, Earth tide and ocean loading tide effects). As can be seen, (1.19) appears to be similar to that for pseudorange measurements based on code tracking (1.16), however there is a significant difference. Code tracking provides essentially unambiguous pseudorange; carrier-phase measurements, instead, ensure high-precision estimation of the satellite-receiver distance, but they are ambiguous by nature. Moreover, the carrier-phase integer remain constant as long as the carrier tracking is on lock. As soon as

the latter breaks out, the integer value N_{r}^{i} may change [28, 11, Chap.5].

1.3.2.1 Ambiguity resolution and cycle slips

To take advantage of the carrier-phase measurements and obtain then an accurate user's position, the ambiguity problem has to be solved. The estimation of the integer number N_{r}^{i} is well-known as *integer ambiguity resolution*. The latter is the key to have very precise positioning and navigation processing, and it applies to various GNSS models with applications in surveying, navigation and geodesy. The theory that carries ultra-precise GNSS carrier-phase ambiguity resolution is the theory of integer inference [49], which started with the *Least-Squares AMBiguity Decorrelation Adjustment* (LAMBDA) method [50, 51]. Afterwards, more sophisticated approaches have been developed based on the observation of multiple bands, such as *Three Carrier Ambiguity Resolution* (TCAR) or its generalization, *Multiple Carrier Ambiguity Resolution* (MCAR). A summary of the ambiguity problem and its existing search methods can be found here [47, 7, Chap.8, Chap.23].

As mentioned before, the GNSS receiver can temporarily lose lock while tracking due to effects like unexpected high user dynamics, short signal blocking, high multipath or high noise. When such event happens, a discontinuity of an integer number of cycles in the measured carrier-phase occurs, known as *cycle slip*. Even when this event occurs only for a fraction of a second and phase lock is established again, the ambiguity number N_{r}^{i} may have changed, as the tracking loop only readjusts the fractional part but not the integer part. Consequently, N_{r}^{i} is unknown again. Various cycle slips detection and correction techniques have been accordingly designed [47, 52, 53, 54, Chap.8]. However, to be attractive in practice, they need also to have an acceptable complexity and not to be too much time consuming (in some cases, a few minutes are needed [55]). Finally, note that local losses of lock may even lead the tracking loop to a complete *drop-lock* from which it never re-covers. In this case, reacquisition is necessary, severely afflicting the positioning accuracy. More details about the cycle slip phenomenon are given in the next chapter (section 2.1.4).

1.3.2.2 Positioning techniques using carrier-phase measurements

There is a variety of GNSS positioning techniques of different degrees of complexity and precision. These range from the standard single-frequency pseudorange-based approach used by most consumer receivers (such as smartphone) to sophisticated multi-frequency carrier-phase-based techniques capable of centimeter position accuracy for demanding applications. In this section, we briefly look at the two positioning techniques based on carrier-phase measurements.

1. *Precise Point Positioning* (PPP) uses carrier-phase measurements as the primary observable with pseudorange measurements playing a secondary role. PPP can be used for processing data from either static sites or moving platforms, providing a few centimeter level accuracies in each coordinate (north, east, and up) [7, Chap.25]. However,

PPP requires the availability of precise reference satellite orbit and clock biases. Its application varies from atmosphere remote sensing to military uses.

2. *Real-Time Kinematic* (RTK) positioning is a classic technique dating to the early 1980s. It combines data from one (or more) reference stations with user data [7, Chap.26]. Positioning accuracies at the decimeter level and better can be obtained. This can be accomplished through a combination of differential GNSS correction data, transmitting GNSS signal carrier-phase measurements and integer ambiguity number via statistical tests, possibly with processing in real-time [6, Chap.7].

In some cases, the two techniques are merged, giving the name PPP-RTK. This combination could bring significant advantages in ambiguity resolution, convergence time, and position accuracy [56].

1.4 Summary

In this chapter, an overview of the navigation system is given. In particular, in the first section, the principle of GNSS is explained through a basic user's position estimation and trilateration method. A typical GNSS signal structure is presented focusing the attention on the difference between the transmitted signal and the received one, which is affected by numerous perturbations through the propagation channel (such as multipath and Doppler effect). In a second section, the GNSS receiver is introduced, and its three main blocks (front end, acquisition and tracking channels, and navigation solution) are detailed. In particular, the tracking part, and specifically, the DLL-DPLL architecture is examined with more attention. In the final section, the main GNSS measurements provided by the tracking channel are presented (code and carrier-phase measurements). Advantages of carrier-phase measurements in terms of precise positioning are explained. However, as underlined, the latter are subject to tracking robustness issues. This last topic is deeply discussed in the next chapter.

Chapter 2

DPLL and robust phase tracking techniques

We have previously introduced the common architecture of GNSS tracking loops. In this chapter, we detail first the most common carrier-phase tracking technique, the so-called *Digital Phase Locked Loop* (DPLL). In particular, we study its behavior in linear and non-linear regimes. The latter occurs due to the detrimental presence of cycle slips. To deal with that, several techniques have been proposed in the literature. A survey of these robust phase tracking techniques is then presented. In particular, we focus our attention on the well-known *Kalman Filter* (KF)-based DPLL. These two techniques will be used later as reference algorithms for comparison with our proposed one in the next chapters.

Contents

2.1 Digital Phase Lock Loop

As anticipated in chapter 1, one of the main components of the tracking channel is the *Digital Phase Locked Loop* (DPLL), which is responsible for the carrier-phase tracking. A DPLL is a loop synchronizing the output signal (locally generated by an oscillator) with a reference or incoming signal in phase [57].

An early description of the analog *Phase Lock Loop* (PLL) was given by de Bellescize [58], a French engineer who dealt with synchronous reception of radio signals. The first of these circuits were linear devices, built by semiconductor technologies. A few years later — around 1960 — the first partially digital (or sampled) PLLs become available on the market [59, Chap.1]. Perhaps the first full digital PLL was reported by Drogin in the early 70's [60]. Although at that time the PLL was cutting-edge technology, the latter was not used extensively for various reasons such as its cost. The first widespread use of analog PLL was in the synchronization of horizontal and vertical scans in television receivers. After few years, it was also used in colored television to keep the color bust synchronized. With the evolution of the technology, the analog PLL landed on the digital terrain with a full DPLL architecture. Aside from the obvious advantages associated with digital systems (mostly, flexibility and versatility), the DPLL alleviated some of the problems associated with its analog counterpart such as difficulties encountered in building higher-order loops [61]. Today, DPLL is widely used for synchronization purposes and has different architectures according to its applications which could go from space communications to bit and symbol synchronization in computers or mobile. In the next section, we analyze the principle of the DPLL and its basic architecture.

2.1.1 DPLL architecture

The DPLL architecture is formed by:

- a correlator,
- a discriminator,
- a loop filter,
- a *Numerical Control Oscillator* (NCO) and replica generator.

Note that in the literature the union of the correlator and discriminator blocks is often called *Phase Detector* (PD). A basic model for a single-channel signal tracking loop in a GNSS receiver is shown in Fig. 2.1.

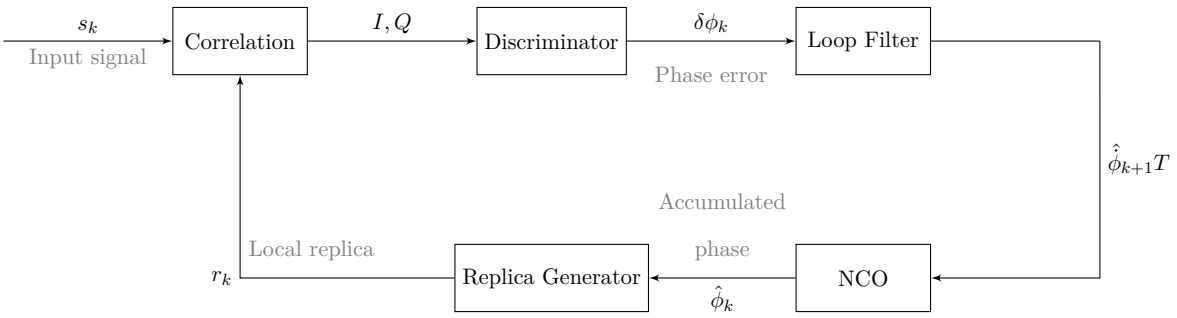


Figure 2.1: DPLL architecture

The input of the tracking loop is the signal in the discrete-time domain, s_k, which is nonlinear in the signal parameter of interest, the phase ϕ_k , and its output is the corresponding estimated phase $\hat{\phi}_k$ used then in the local replica r_k. The discriminator output is the signal parameter error, $\delta\phi_k$, obtained by comparing the incoming signal and the local replica. In fact, this process is accomplished by the combination of the correlator outputs, that is the baseband I and Q signal components. Since this discriminator output contains noise, the latter has to be efficiently filtered out by the loop filter. Note that the output of the loop filter is the rate of change information of the signal parameter of interest, $\hat{\dot{\phi}}_{k+1}T$, which is then integrated in the NCO to predict the signal parameter estimate, $\hat{\phi}_k$, for the next step. This signal parameter estimate is used in the local signal generator to produce the local replica signals for the correlation. In the synchronized — so-called *locked* — state, the phase error between the local replica and incoming signals is zero or remains constant.

2.1.1.1 Input signal and scenario

Before detailing each DPLL's block, a description of the incoming signal s_k is useful. The latter is proposed assuming that:

1. there is no navigation message (*e.g.*, pilot or wiped-off signal);
2. the noise can be reduced to an *Additive White Gaussian Noise* (AWGN).

Furthermore, regarding the GNSS receiver architecture, we assume that

1. the input signal is taken after the integration and at the output of the Prompt correlator, therefore no code sequence is present;
2. we assume that either delay or both delay and frequency are perfectly synchronized after the acquisition stage. To obtain this, two possible architectures could be designed: 1) the tracking block entails only a DLL and the frequency is compensated with that estimated at the acquisition stage (in that case, the frequency is assumed to remain almost constant during a certain amount of time); 2) the tracking block entails both a DLL-FLL. So far, in this thesis project, the former architecture is used and presented when using real GNSS data in chapter 5.

Note that all these hypotheses related to the input signal and the GNSS receiver scenario will be held during the entire manuscript for both exemplifying (when appropriate) the state-of-art as well as building our contribution.

As a consequence of these assumptions, considering a single channel propagation, the input signal can be defined as [1]

$$s_k = \alpha e^{j\phi_k} + n_k, \tag{2.1}$$

[1]For sake of notation clarity, note that $x_k \overset{\text{def}}{=} x[k]$, where x is a generic parameter.

where α is the amplitude of the signal, assumed to be constant over the sample indices k, and ϕ_k is the phase of the *Signal-of-Interest* (SoI), to be estimated. The noise component n_k is described as

$$n_k \sim \mathcal{CN}(0, \sigma_n^2), \tag{2.2}$$

where σ_n^2 is the noise variance.

Noise level

In signal processing, a fundamental parameter used to analyze the signal is the *Signal-to-Noise-Ratio* (SNR). The latter compares the level of the desired signal to the level of background noise. Considering our signal model (2.1) which corresponds to a post-correlated signal for assumptions made, the SNR is defined as

$$\text{SNR} = \frac{P_{\text{SoI}}}{P_{n_k}} = \frac{\mathcal{E}\{|\alpha e^{j\phi_k}|^2\}}{\mathcal{E}\{|n_k|^2\}} = \frac{|\alpha|^2}{\sigma_n^2}, \tag{2.3}$$

where P denotes the power of a signal component. In GNSS, a commonly used metrics to express the noise level is the *Carrier-to-Noise-Density Ratio* (C/N_0), which is the ratio of the power level of the carrier signal to the noise power N_0 in a 1 Hz bandwidth. It is defined as follows

$$C/N_0 = \frac{P_{\text{SoI}}}{N_0}. \tag{2.4}$$

The relation between these two parameters is given by

$$\text{SNR} = (C/N_0)T, \tag{2.5}$$

where T is the integration time of the loop. Fig. 2.2 shows the SNR value as a function of integration time T at different C/N_0. A maximum value of T equal to 20 milliseconds is chosen, equivalent to the navigation data bit/symbol period of the GPS L1C signal.

2.1.1.2 Correlator

The first operation of the DPLL is the correlation process. Its task is to compare the incoming signal with the local replica. The correlator is implemented as a mixer (or multiplier) as shown in Fig. 2.3. The output g_k is function of the phase difference between the incoming signal and the replica. Through the real and imaginary parts of the mixer output, we can calculate the In-phase (I) and Quadrature (Q) components as

$$I = \Re(g_k) \qquad \text{and} \qquad Q = \Im(g_k). \tag{2.6}$$

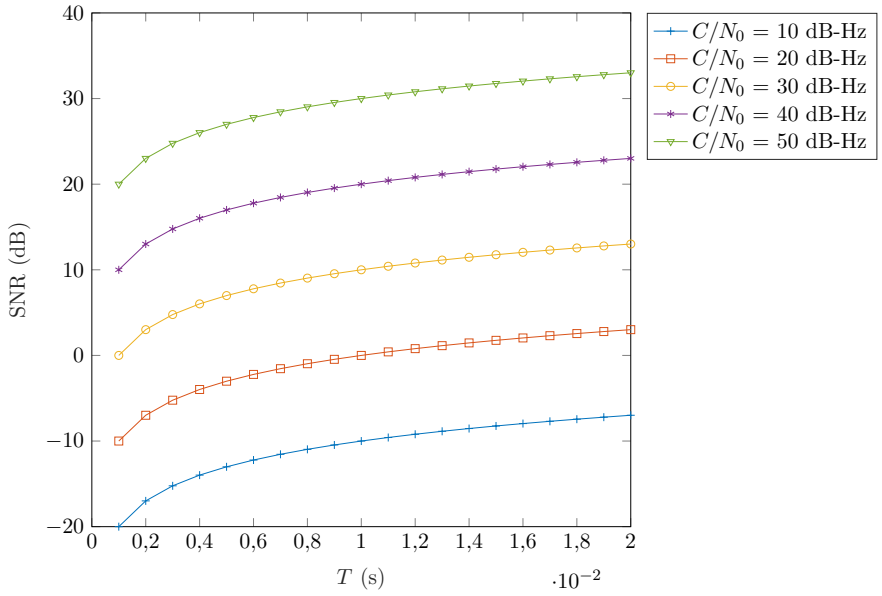


Figure 2.2: Relation between SNR and C/N_0

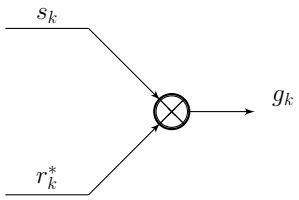


Figure 2.3: DPLL correlator block

2.1.1.3 Discriminator

The purpose of a discriminator is to extract the signal parameter error from the correlator outputs I and Q. In relation to the discriminator type, we can distinguish two types of carrier-phase tracking loops: a pure (or coherent) DPLL and a Costas (or non-coherent) DPLL. Their main difference is that the pure DPLL is sensitive to the presence of bit/symbol modulation on the signal. The expressions of common phase discriminators are presented in Tab. 2.1 [4, Chap.5]. Assuming a noise-free signal, the phase error at the discriminator output is shown in Fig. 2.4. As can be seen, all outputs have a periodic behavior and a linear trend around zero. Finally, an interesting alternative to the classical discriminator types is the one based on the only quadrature component Q as proposed in [62, 63, 64].

Note that in our framework the input signal is assumed to be without a navigation message (see section 2.1.1.1), therefore both discriminator types may be used. However, since we will work at low SNR, from now on we only use the pure DPLL with ATAN2 discriminator due to its significant performance in noise rejection and robustness to dynamics compared to Costas discriminators.

Table 2.1: Discriminators expressions for the phase estimation

Discriminator type	*Expression*
Four-Quadrant Arctangent (*ATAN2*)	arctan2(Q,I)
Cross-Product (*CP*, Costas)	Q × I
Decision Directed Cross-Product (*DDCP*, Costas)	Q × sign(I)
Arctangent (*ATAN*, Costas)	arctan(Q/I)

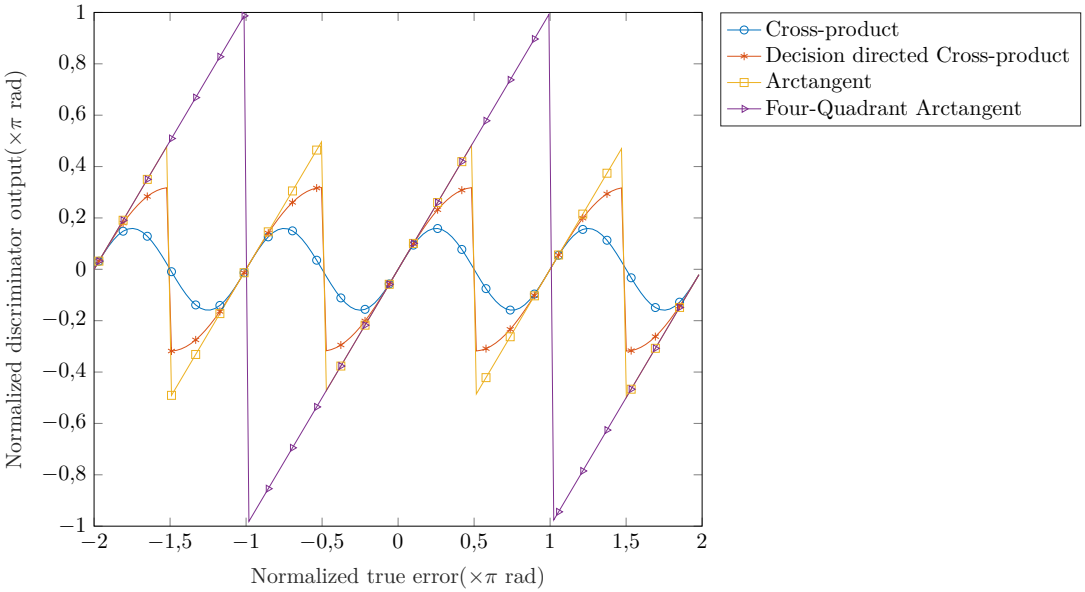


Figure 2.4: Comparison between the phase error output of different discriminator types

2.1.1.4 Loop filter

Since the discriminator output may still contain an important noise level, a filter is placed soon after. There are many different approaches to designing a digital loop filter [65]. One of the most common is to design the filter in the continuous-time domain, and then to convert it in the discrete-time domain. This can be done through different methods such as the forward/backward rectangular method (also known as the forward/backward Euler method), trapezoidal method (also known as bi-linear method), and Simpson method [66, 67, 68]. Considering the rectangular method, an Nth-order loop filter can be represented as follows

$$F(z) = \frac{z\hat{\dot{\phi}}(z)T}{\delta\phi(z)} = \bar{K}_1 + \frac{\bar{K}_2}{1-z^{-1}} + \frac{\bar{K}_3}{(1-z^{-1})^2} + \cdots + \frac{\bar{K}_N}{(1-z^{-1})^{N-1}}, \tag{2.7}$$

where $F(z)$ is the filter transfer function[2], z^{-1} is the delay in the loop filter. The latter is retrieved by [69]. The $\bar{K}_n$'s represent the Nth-order filter coefficients where $n = 1, \ldots, N$. Most of GNSS receivers use second- or third-order DPLLs [30, Chap.6]. As previously underlined, the higher the DPLL order, the higher the phase dynamics it can track. However, the price to pay is then a decreased robustness to noise [30, Chap.6].

2.1.1.5 Numerical Control Oscillator and replica generator

The loop filter produces the signal phase rate. The latter is then fed to the NCO, which is a digital signal generator. Computationally, the NCO can be seen as a simple integrator (or accumulator) of the phase rate to produce the phase, namely $\hat{\phi}_k$. Considering the rectangular method, we have

$$\hat{\phi}_k = \hat{\phi}_{k-1} + \hat{\dot{\phi}}_k T, \tag{2.8}$$

where k and $k-1$ indicate the current and previous discrete-time steps, respectively. The estimated carrier-phase is then fed to a carrier generator to produce the phase shifted replica at the next iteration, namely

$$r_k = e^{j\hat{\phi}_k}. \tag{2.9}$$

2.1.2 Linearized DPLL model

Every DPLL is nonlinear. The nonlinearity is related to the discriminator function which is periodic and necessarily nonlinear as shown in Fig. 2.4, but also to the presence of noise and dynamics. A linear model of the DPLL typically is applicable if phase error is small, a condition normally attained when the loop is locked [61]. Under this regime, conventional tools of linear systems as transfer-function analysis can be used for the determination of DPLL responses. Since we are working in the discrete-time domain, the z-transform technique is used [70, Chap.3].

2.1.2.1 Linearization

Under the assumption that the tracking phase error is small enough, we can state that DPLL works around its equilibrium point, obtaining then a linear expression of the discriminator. In particular, we assume that in absence of noise, the discriminator output is

$$\delta\phi_k = \phi_k - \hat{\phi}_k. \tag{2.10}$$

In this way, the linear model of the DPLL can be designed as in Fig. 2.5. The input phase is

[2]To reach this expression, we used the relation $\mathcal{Z}\{x_{n+1}\} = z\mathcal{Z}\{x_n\}$, where x is a generic parameter and $\mathcal{Z}\{\}$ represents a z-transform [69].

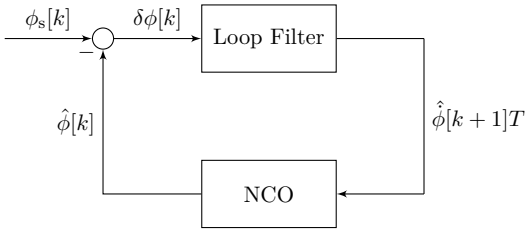


Figure 2.5: Linear model of the DPLL

defined as[3]

$$\phi_s[k] = \phi[k] + \phi_n[k], \tag{2.11}$$

where $\phi_s[k]$ is the phase of the incoming signal (2.1), $\phi[k]$ and $\phi_n[k]$ are the phase of the SoI and phase noise, respectively. Note that, in a linear regime (at high SNR) whenever the original noise is Gaussian and white, its phase can be considered as a white Gaussian process. The former statement has been proofed in the case of analog PLL [71], however, we assume it to be valid for DPLL too. Furthermore, following [71], we assume that the *Power Spectral Density* (PSD) of $\phi_n[k]$ in case of DPLL is

$$S_{\phi_n}(f) = \frac{N_0}{2\alpha^2}, \tag{2.12}$$

where N_0 is the noise spectral density and α is the amplitude of the incoming signal.

Let us denote $N(z)$ and $F(z)$ the transfer functions of the NCO and the loop filter in Fig 2.5, respectively, we have that

$$H(z) = \frac{\hat{\phi}(z)}{\phi_s(z)} = \frac{z^{-1}F(z)N(z)}{1 + z^{-1}F(z)N(z)}, \tag{2.13}$$

where $H(z)$ is the transfer function of the loop in z-domain. Therefore, the estimated phase can be simply calculated as follows

$$\hat{\phi}(z) = \phi_s(z)H(z). \tag{2.14}$$

2.1.2.2 Steady-state error

The steady-state error is used to study the asymptotic behavior of a system [72, Chap.8]. Considering the final-value theorem in discrete-time domain [73, Chap.2], the steady-state is expressed as

$$e_{\text{ss}} = \lim_{k \to +\infty} e[k] = \lim_{z \to 1}[(1 - z^{-1})E(z)], \tag{2.15}$$

where, in our case,

$$e[k] = \phi_{\text{s}}[k] - \hat{\phi}[k]. \tag{2.16}$$

[3]To avoid having a double subscript, the notation $x[k]$ is used (instead of x_k), where x indicates a generic parameter.

Its z-transform is

$$\begin{aligned} E(z) &= \phi_{\mathrm{s}}(z) - H(z)\phi_{\mathrm{s}}(z) = (1 - H(z))\phi_{\mathrm{s}}(z) \\ &= \Big(1 - \frac{z^{-1}F(z)N(z)}{1 + z^{-1}F(z)N(z)}\Big)\phi_{\mathrm{s}}(z) = \Big(\frac{1}{1 + z^{-1}F(z)N(z)}\Big)\phi_{\mathrm{s}}(z). \end{aligned} \tag{2.17}$$

Therefore,

$$e_{\mathrm{ss}} = \lim_{z\to 1}\Big[(1 - z^{-1})\Big(\frac{1}{1 + z^{-1}F(z)N(z)}\Big)\phi_{\mathrm{s}}(z)\Big]. \tag{2.18}$$

Note that the above steady-state depends on the loop filter order and the input dynamics.

Steady-state using different input dynamics

Herein, the steady-state error e_{ss} is analyzed considering three different input dynamics. The phase input of the incoming signal is supposed to have the following expression

$$\phi_s[k] = \phi_0 + \dot{\phi}_0 Tk + \frac{\ddot{\phi}_0 T^2 k^2}{2}, \tag{2.19}$$

where ϕ_0, $\dot{\phi}_0$, $\ddot{\phi}_0$ are respectively the initial phase (step), phase rate (ramp), acceleration (parabola).

1. *Step input*:
   The phase input $\phi_s[k]$ is a zero order polynomial input, equivalent to a step input [4].
   The steady-state is equal to

   $$e_{\mathrm{ss}} = \lim_{z\to 1}\Big(\frac{\phi_0}{1 + z^{-1}F(z)N(z)}\Big) = \frac{\phi_0}{1 + K_p}, \tag{2.21}$$

   where $K_p = \lim_{z\to 1} z^{-1}F(z)N(z)$ is the error coefficient for a step input.

2. *Ramp input*:
   The phase input $\phi_s[k]$ is a first-order polynomial input, equivalent to a ramp input[5].
   The steady-state is equal to

   $$e_{\mathrm{ss}} = \lim_{z\to 1} \frac{\dot{\phi}_0 z}{(z-1)F(z)N(z)} = \frac{\dot{\phi}_0}{K_v}, \tag{2.23}$$

[4] Its z-transform is calculated using [70, Chap.3]

$$u[k] \longrightarrow \frac{1}{1 - z^{-1}}. \tag{2.20}$$

[5] Its z-transform is calculated using [70, Chap.3]

$$ka^k u[k] \longrightarrow \frac{az^{-1}}{(1 - az^{-1})^2}. \tag{2.22}$$

where $K_v = \lim_{z\to 1} \frac{z}{(z-1)} F(z)N(z)$ is the error coefficient for a ramp input.

3. *Parabolic input*:
   The input $\phi[k]$ is a second-order polynomial input, equivalent to a parabolic input (the z-transform is calculated using (2.22)). The steady-state error is

$$e_{\text{ss}} = \ddot{\phi}_0 \lim_{z\to 1} \frac{(z+1)z}{(z-1)^2[1+F(z)N(z)]} = \frac{\ddot{\phi}_0 z}{\lim_{z\to 1}(z-1)^2 F(z)N(z)} = \frac{\ddot{\phi}_0}{K_a}, \tag{2.24}$$

   where $K_a = \lim_{z\to 1} \frac{z}{(z-1)^2} F(z)N(z)$ is the error coefficient for a parabolic input.

Steady-state using different filter orders

The steady-state error for first-, second- and third-order loop is summarized in Tab. 2.2.

Table 2.2: Steady state errors for different DPLL filter's orders and dynamics inputs.

Filter order	Step input	Ramp input	Parabolic input
1st	$\frac{\phi_0}{1+\bar{K}_p}$	∞	∞
2nd	0	$\frac{\dot{\phi}_0}{K_v}$	∞
3rd	0	0	$\frac{\ddot{\phi}_0}{K_a}$

Steady-state example using a first order DPLL

Considering

$$F(z) = \bar{K}_1 \qquad \text{and} \qquad N(z) = \frac{1}{1-z^{-1}}, \tag{2.25}$$

the steady-state errors are shown in Tab 2.3. The latter are calculated using (2.21),(2.23), and (2.24) for step, ramp, and parabola input type, respectively. Fig. 2.6 shows the comparison between theoretical and empirical steady-state errors. The latter have been obtained by using a first-order linear model of the DPLL. As expected, the empirical steady-state errors converge to the theoretical ones.

2.1.2.3 Loop Bandwidth

The steady-state error and the convergence time of the DPLL are strongly influenced by the value of the loop noise bandwidth B_L. The single-sided form of the latter for a digital loop

Table 2.3: Error constants and steady-state errors values

Dynamics	**Error coefficient**	**Steady-state error**
Step	$K_p = \infty$	$e_{ss} = 0$
Ramp	$K_v = \bar{K}_1$	$e_{ss} = \dot{\phi}_0 / K_v$
Parabolic	$K_a = 0$	$e_{ss} = \infty$

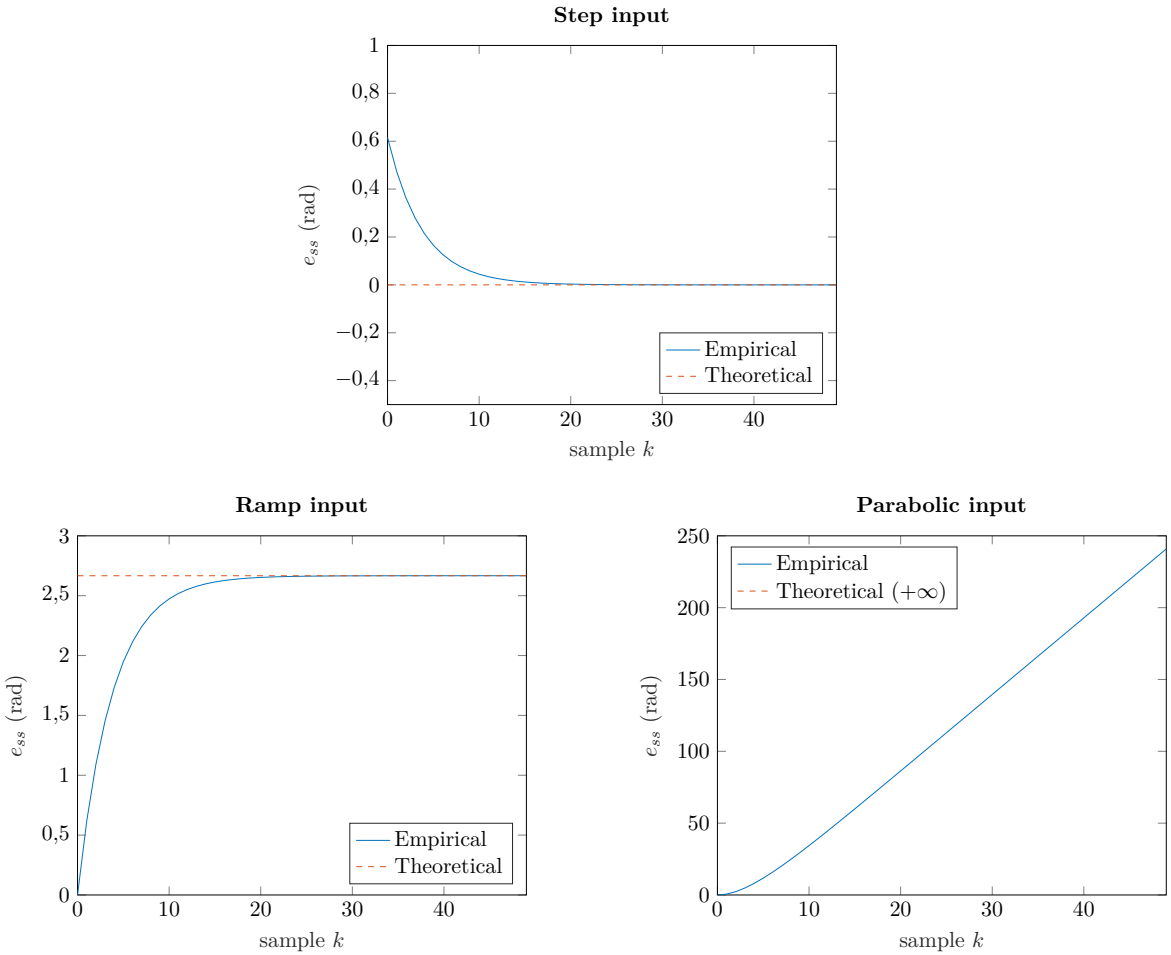


Figure 2.6: Steady-state error for a linearized first-order DPLL in absence of noise. ($\bar{K}_1 = 0.3$ rad; step: $\phi_0 = 0.8$ rad; ramp: $\dot{\phi}_0 = 0.2$ rad/sample; parabola: $\ddot{\phi}_0 = 0.8$ rad/sample^2)

is defined by [69, 74]

$$2B_L = \int_{-\frac{1}{2T}}^{\frac{1}{2T}} |H(e^{i2\pi fT})|^2 df. \tag{2.26}$$

By using the transformation $z = e^{i\omega}$, we can rewrite this integral as a contour integral as [69, 61]

$$2B_L T = \frac{1}{2\pi i} \oint H(z)H(z^{-1})z^{-1}dz, \tag{2.27}$$

where the close path is along the unit circle. While solving the contour integral, it turns out that the resulting B_L can be expressed as a function of only the loop constants $\bar{K}_i$, where $i = 1, \ldots, N$. On that regard, B_L expressions from-one-to-four order loop are presented in [69]. Another way to derive (2.27) is by solving the two-sided complex integral as in [75, 76, pp.297-299], knowing the expression of the transfer function $H(z)$.

Loop bandwidth influence on steady-state using a first-order DPLL

In case of first-order loop, the product between loop bandwidth B_L and integration time T is expressed as [69]

$$B_L T = \frac{\bar{K}_1}{2(2 - \bar{K}_1)}. \tag{2.28}$$

Using a linearized first-order DPLL in absence of noise, the steady-state is calculated for different B_L values. Results are shown in Fig. 2.7. The higher the loop bandwidth B_L, the lower the time response (step case) and steady-state error (ramp and parabola cases). On the other hand, as we will underline later, increasing the loop bandwidth leads to a lower robustness to noise. In typical GNSS applications, the DPLL must have a relatively small bandwidth (1-10 Hz) to obtain both noise rejection and agile carrier tracking [30, Chap.6].

2.1.2.4 Phase noise variance

The variance of the phase output is [77, Chap.3]

$$\sigma^2_{\hat{\phi}} = \int_{-1/2T}^{1/2T} |H(f)|^2 S_{\phi_n}(f)df. \tag{2.29}$$

If the noise input applied to the loop is white, using (2.26) and (2.12), the phase variance is given by the simple expression

$$\sigma^2_{\hat{\phi}} = \frac{N_0 B_L}{\alpha^2}. \tag{2.30}$$

We can immediately notice that the noise variance is proportional to the loop bandwidth: the greater the loop bandwidth B_L, the greater the noise variance $\sigma^2_{\hat{\phi}}$. Therefore, as mentioned in the previous section, a compromise when choosing B_L is needed to obtain both agile tracking and robustness to noise. A more detailed expression of the variance of the phase output has been calculated in the case of a linear first-order loop using a Costas discriminator, namely [5, 4, Chap.8, Chap.5]

$$\sigma^2_{\hat{\phi}} = \frac{B_L}{C/N_0}\left(1 + \frac{1}{2C/N_0 T}\right). \tag{2.31}$$

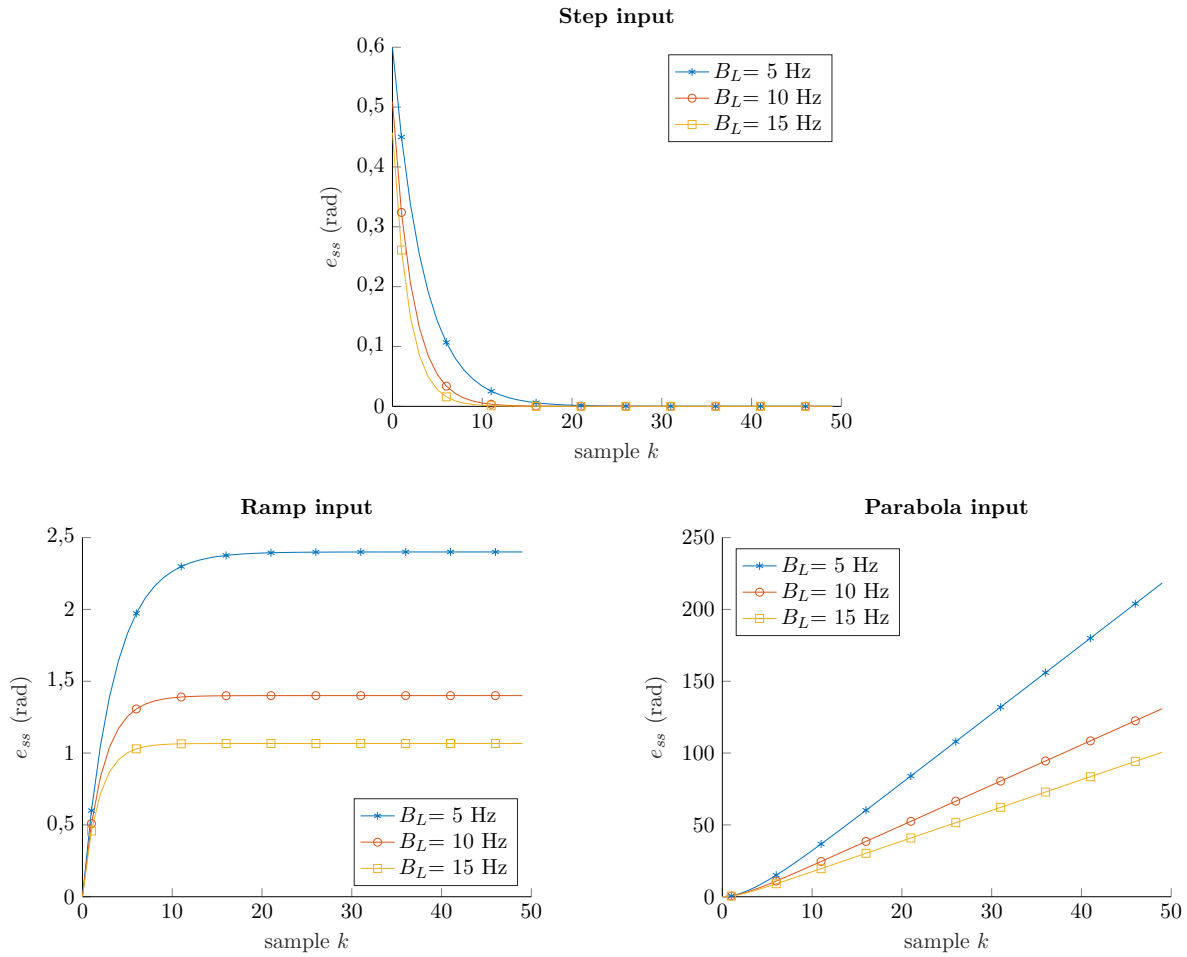


Figure 2.7: Steady-state error for a linearized first-order DPLL in absence of noise — Loop bandwidth impact. ($T = 0.02$ s; step: $\phi_0 = 0.8$ rad; ramp: $\dot{\phi}_0 = 0.8$ rad/sample; parabola: $\ddot{\phi}_0 = 0.8$ rad/sample^2)

The comparison between (2.30) and (2.31) is shown in Fig. 2.8. The two phase variances result to have a similar trend and achieve the same values at high C/N_0, proofing a good approximation of (2.30) for a preliminary study of the DPLL.

2.1.3 Noise in DPLL

The linear analysis presented in the former section was based on an assumption of a small phase error in the loop, but the actual error in presence of noise is not necessary small. As a result, the linear analysis fails when its assumption is violated. As said before, in a linear system, a Gaussian input gives rise to a Gaussian output. In these circumstances, a centered Gaussian process is completely defined by its autocorrelation, or equivalently, by its spectral density as derived in (2.14). However, the response of a nonlinear system to a Gaussian input is generally non-Gaussian [77, Chap.3]. Therefore, in case of DPLL, nonlinear analysis has

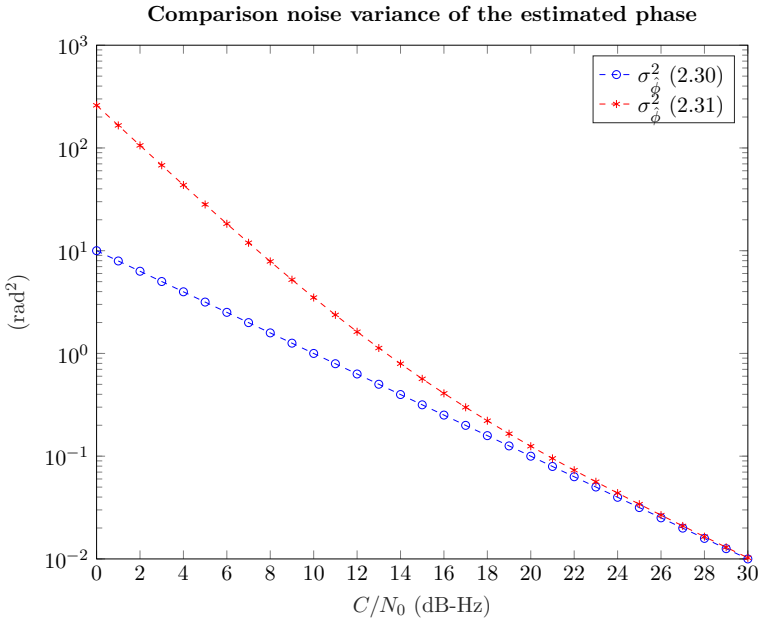


Figure 2.8: Variance of the output phase. ($T = 0.02$ s; $B_L = 10$ Hz)

been concerned with deriving the non-Gaussian *Probability Density Function* (PDF) of the *phase error*, computing the phase variance from the latter, and finally the investigation of cycle slip (see section 2.1.4). Note that the nonlinear analysis is extremely challenging since it demands a high level of mathematical sophistication. We refer to Viterbi's exact study of a first-order analog loop which has brought many insights to understand nonlinear analysis [71].

2.1.3.1 Phase error and its statistic

The phase error is defined by

$$e_k = \phi_k - \hat{\phi}_k. \tag{2.32}$$

It is not necessarily small especially at low SNRs. To obtain a bounded error, a new variable is then used in the further analysis, namely

$$\tilde{e}_k = e_k \bmod(2\pi), \tag{2.33}$$

which means that $e_k = \tilde{e}_k + 2n\pi$, where n is chosen so that $\tilde{e}_k$ lies between $[-\pi, \pi)$.

Theoretical PDF of the phase error using analog PLL

The PDF of the phase error $\tilde{e}_k$ is denoted as $f(\tilde{e}_k)$. The latter is found by Viterbi for a first-order analog PLL with a constant input as the steady-state solution of a nonlinear, stochastic

partial differential equation known as the *Fokker–Planck equation* [6] [71]

$$f(\tilde{e}_k) = \frac{\exp\Big(\mathrm{SNR}_L \cos(\tilde{e}_k)\Big)}{2\pi I_0(\mathrm{SNR}_L)}, \tag{2.34}$$

which is the Von Mises distribution (also known as the circular normal distribution or Tikhonov distribution) [81, Chap.3]. SNR_L is the SNR in the bandwidth of the analog loop, defined as

$$\mathrm{SNR}_L = \frac{\alpha^2}{N_0 B_L}. \tag{2.35}$$

The latter is a fictitious quantity without firm physical meaning; its conceptual meaning for an analog PLL is explained in [77, Chap.3]. $I_0(\cdot)$ is the modified Bessel function of the first kind at zero-order [82, Chap.9], which can be replaced by the asymptotic expansion [81, Chap.3], namely

$$I_0(\mathrm{SNR}_L) = \frac{\exp(\mathrm{SNR}_L)}{(2\pi\mathrm{SNR}_L)^{1/2}}\left\{1 + \frac{1}{8\mathrm{SNR}_L} + \frac{9}{2!(8\mathrm{SNR}_L)^2} + \dots\right\}. \tag{2.36}$$

The PDF $f(\tilde{e}_k)$ in (2.34) is shown in Fig 2.9. Note that, for high SNR, knowing (2.36) and expanding $\cos(\tilde{e}_k)$ in a Taylor series, we have

$$f(\tilde{e}_k) \underset{\mathrm{SNR}_L \to +\infty}{=} \frac{\exp\Big[-\frac{\mathrm{SNR}_L \tilde{e}_k^2}{2}\Big(1 - \frac{2\tilde{e}_k^4}{4!} + \frac{2\tilde{e}_k^6}{6!} \dots\Big)\Big]}{(2\pi/\mathrm{SNR}_L)^{1/2}}. \tag{2.37}$$

When SNR_L is large, $f(\tilde{e}_k)$ decays rapidly, going from a Von Mises distribution to nearly Gaussian one. In this case, linear analysis is quite accurate [71].

Empirical PDF of the phase error using DPLL

The empirical PDF of the phase error using a first-order DPLL is here showed and compared with the theoretical PDF, namely (2.34), using a first-order analog PLL. A *Four-Quadrant Arctangent* (ATAN2) discriminator is considered (see Tab. 2.1). The simulation has been implemented through M_c Monte Carlo runs [83, 84]. The DPLL is considered at steady-state with $\hat{\phi}_0 = \phi_0 = 0$. Results are shown in Fig. 2.10. As can be seen, theoretical PDFs match the empirical ones in most of the C/N_0s simulated, showing that at high C/N_0 the theoretical expression of the PDF using a first-order analog PLL in (2.34) can be used for a DPLL too up to C/N_0 around 10 dB-Hz.

[6] The Fokker-Planck equation was firstly used by Fokker and Plank to describe the Brownian motion of particles [78, 79]. For further details on this topic, a recommended book is [80].

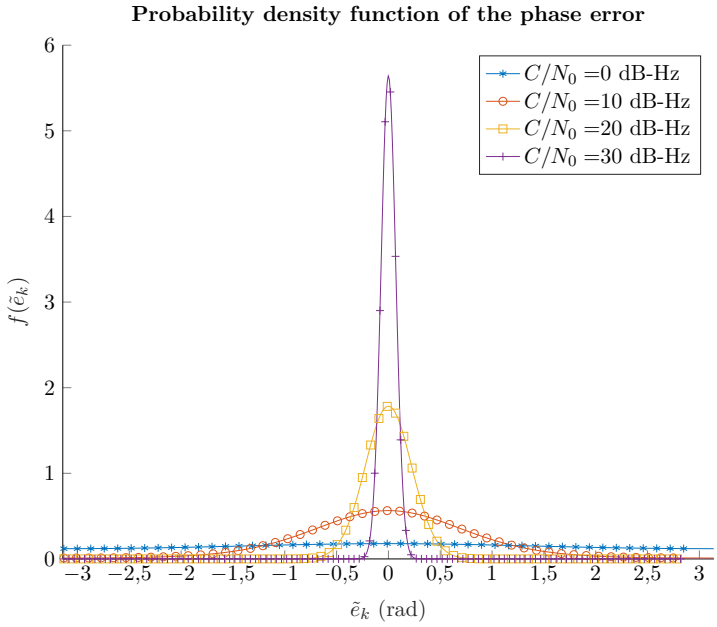


Figure 2.9: Theoretical PDF of the phase error using (2.34) for an analog PLL. ($T = 0.02$ s; $B_L = 10$ Hz)

2.1.3.2 Phase noise variance

The variance of the phase error can be found by numerical evaluation of [77, Chap.3]

$$\sigma^2_{\tilde{e}_k} = \int_{-\pi}^{\pi} \tilde{e}_k^2 f(\tilde{e}_k) d\tilde{e}_k. \tag{2.38}$$

The latter is compared with the variance of the phase output in (2.30), knowing that in case of an assumed unbiased estimator and linear regime

$$\sigma^2_{\hat{\phi}} = \mathcal{E}\{(\hat{\phi}_k - \mathcal{E}(\hat{\phi}_k))^2\} = \mathcal{E}\{(\hat{\phi}_k - \phi_k)^2\} = \mathcal{E}\{e_k^2\}. \tag{2.39}$$

A comparison is shown in Fig. 2.11. The variance of the phase error in nonlinear regime has been calculated considering a first-order DPLL at steady-state with an ATAN2 discriminator and constant phase input. As can be seen, the exact variance agrees with the linear analysis for large C/N_0 and approaches $\pi^2/3$ rad^2 for very small C/N_0 (the variance of a random variable, uniformly distributed over $(-\pi, \pi)$, is $\pi^2/3$), as found in literature [77, Chap.3].

The variance of the phase error is informative but not enough. Statistics of cycle slips are an essential attribute of DPLL analysis, especially at low C/N_0. The former phenomenon is therefore explained in the next section.

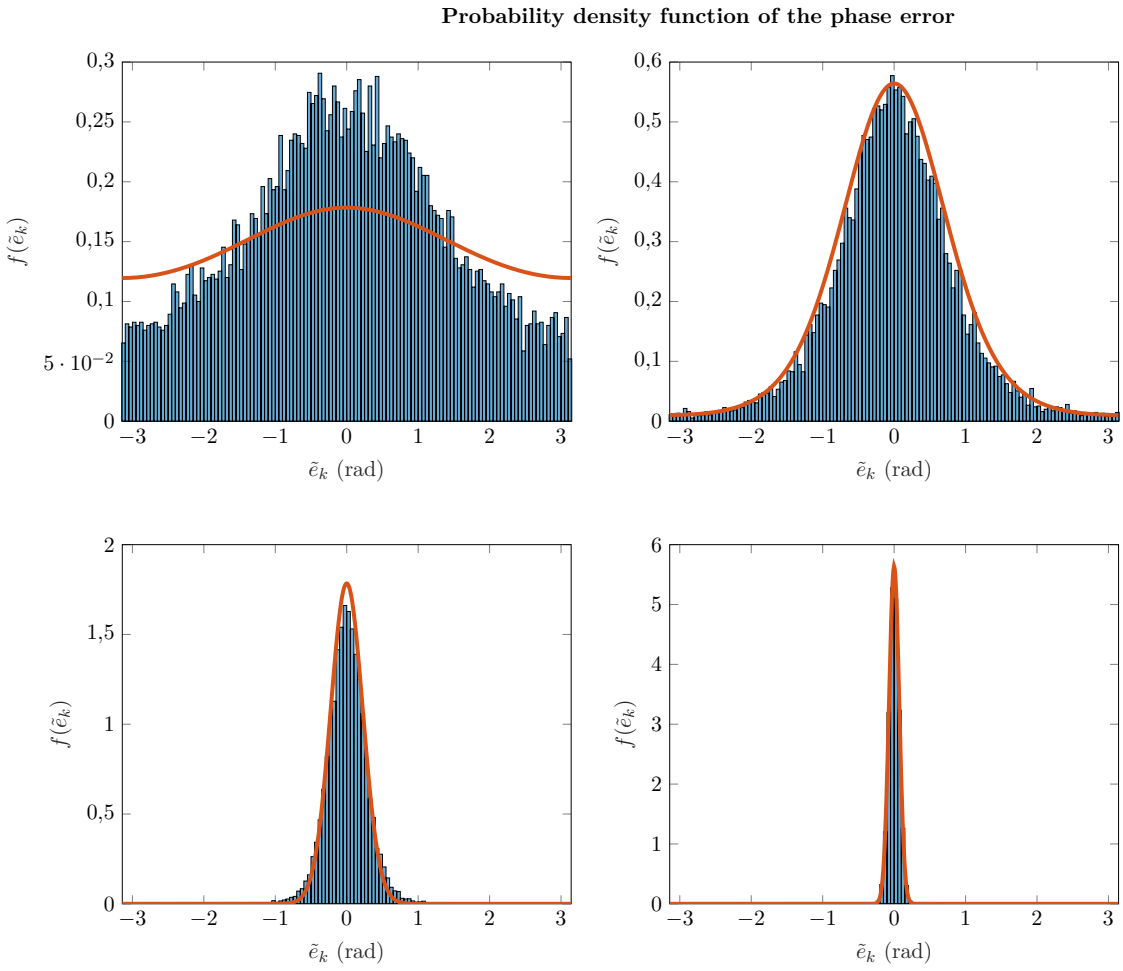


Figure 2.10: Empirical PDF of the phase error (at time 2 s). (Up-left: $C/N_0 = 0$ dB-Hz; up-right: $C/N_0 = 10$ dB-Hz; bottom-left: $C/N_0 = 20$ dB-Hz; bottom-right: $C/N_0 = 30$ dB-Hz. $T = 0.02$ s; $B_L = 10$ Hz. Parameter setting for empirical PDF: filter coefficients are calculated using (2.28) in [69]; $M_c = 1500$ realizations)

2.1.4 Cycle slip study

An important phenomenon to be taken into account while analyzing the DPLL behavior is *cycle slipping*. As shown in section 2.1.1.3, the phase discriminator is periodic, meaning that it cannot distinguish between the phase errors e_k (defined in (2.32)) and $e_k + 2n\pi$ (or e_k and $e_k + n\pi$ in case of Costas loop), where n is an integer. As a result, an infinite set of stable attractors exists for the nonlinear equations that describe the DPLL error dynamics. Especially at low SNRs, the phase error may slip from one stable attractor to another, leading to infinite variance of the phase error, namely $\sigma^2_{e_k}$. A so-called *drop-lock* (or permanent phase unlock) occurs where the equilibrium is never recovered.

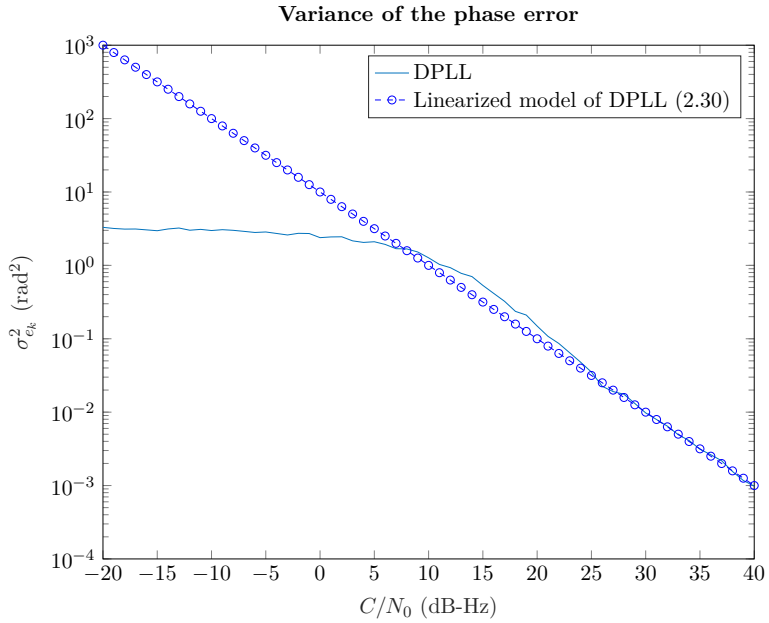


Figure 2.11: Variance of the phase error. ($B_L = 10$ Hz; $T = 0.02$ s; $\sigma_n = 1$. Parameter setting for nonlinear analysis (at time 2 s): $M_c = 1000$ realizations; $\hat{\phi}_0 = \phi_0 = 0$ rad; filter coefficients are calculated using (2.28) in [69])

2.1.4.1 Cycle slip in literature

Cycle slipping has been widely studied in the literature. Seminal papers on the cycle slip phenomenon have been published between 1960 and 1980 with the widespread use of analog PLL. Everything started with Viterbi's analysis that was able to derive the Fokker-Planck (or diffusion) equation for a general loop, and then to manipulate it to analyze a first-order loop [71]. As shown in section 2.1.3.1, Viterbi gave an exact expression for the steady-state phase error distribution for a first-order loop. Furthermore, he derived an expression for the average time between cycle slips (presented in the next section 2.1.4.3), explaining it through parallelism between a first-order loop and the pendulum behavior. Viterbi also laid the foundations for a second-order analysis, giving an approximated and asymptotic expression of the phase error distribution [71]. However, in this case, Fokker-Planck equations can be written, but exact closed-form solutions have been unattainable [77]. Note that the second-order loop is practically a relevant system, therefore there is a strong motivation to determine its statistics and cycle slipping at low SNRs. Experimental results were firstly found by Charles and Lindsey in [85], which realized the existence of a sharp dichotomy where cycle slipping occurs frequently and not. In particular, they found that, in the case of a second-order loop, the average time to first slip is shorter than in a first-order, especially for low damping factor. After some years, Ascheid and Mayer performed an extensive empirical analysis of cycle slips in the second-order loop, avoiding advanced mathematical tools [86]. As predicted in [85], they found that for damping factors higher than a given threshold cycle slips occur in bursts. Furthermore, to calculate the average time between cycle slips in the case of second-order loops, several indications have been found, mainly by Lindsey and Tausworthe in [87,

88, 89]. However, since the formulas are cumbersome and since their derivation necessarily involved approximations, the first-order analysis by Viterbi remains the main reference in the tracking community. Finally, in the case of third- and higher-order loops, there is a lack of knowledge in nonlinear analysis, and specifically, in cycle slipping. To overcome this void, a way to proceed is assuming that a third-order loop behaves much the same as a second-order loop of the same noise bandwidth [77].

2.1.4.2 Cycle slip examples and PDF of the phase error

As already anticipated in chapter 1 (section 1.3.2.1), the sources causing the slip occurrence can be various such as signal obstruction in the propagation channel, low SNR or incorrect signal processing (such as failure in the receiver software [90]) [11, Chap.7]. To have a visual understanding of slipping during carrier tracking, we present examples of cycle slip and drop-lock due to low SNR in Fig. 2.12 and Fig. 2.13, respectively. A second-order DPLL with an ATAN2 discriminator is used. As can be seen in case of low SNRs, a drop-lock occurs.

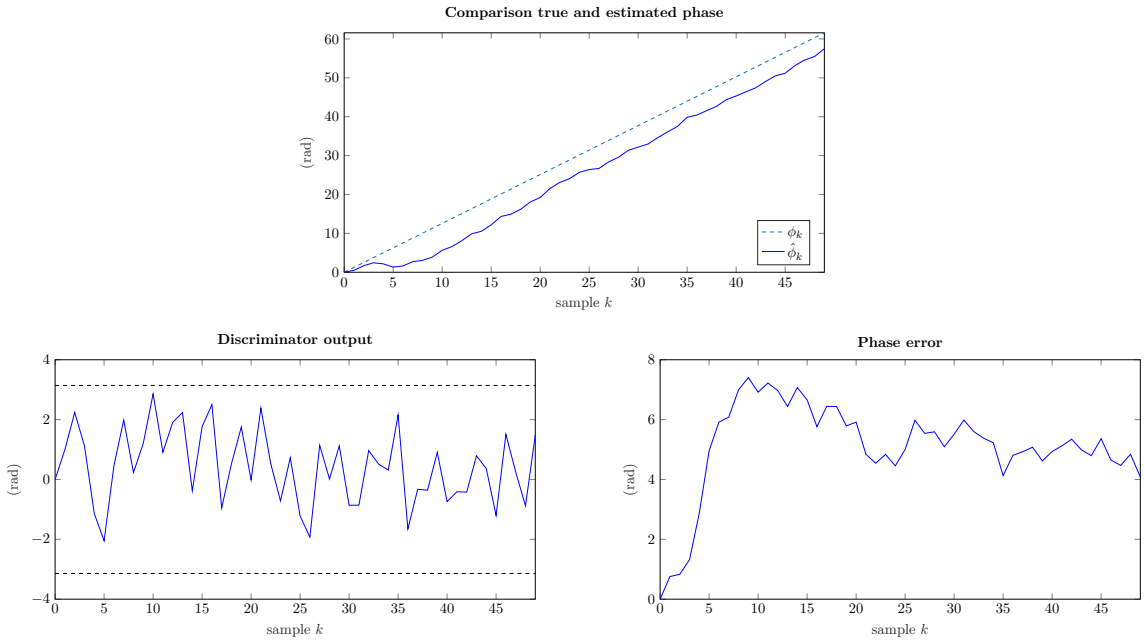


Figure 2.12: Cycle slips due to low SNR. ($T = 0.02$ s; $\sigma_n = 1$; $B_L = 10$ Hz; $\dot{\phi}_0 = \pi/25$ rad/sample; $\hat{\dot{\phi}}_0 = 0$ rad/sample; $C/N_0 = 22$ dB-Hz. Filter coefficients are calculated using [69, Tab. V])

Furthermore, for the sake of coherency with the linear analysis, we present the PDF of the phase error e_k when cycle slipping is not neglected (previously, the phase error modulus 2π, namely $\tilde{e}_k$, was considered; see section 2.1.3.1). This analysis is beneficial to analyze the occurrence of cycle slips and may be a useful tool to assess the robustness of the loop. Therefore, considering always a second-order DPLL with an ATAN2 discriminator and ramp phase input, the PDF of the phase error is shown in Fig. 2.14. As can be seen, the distribu-

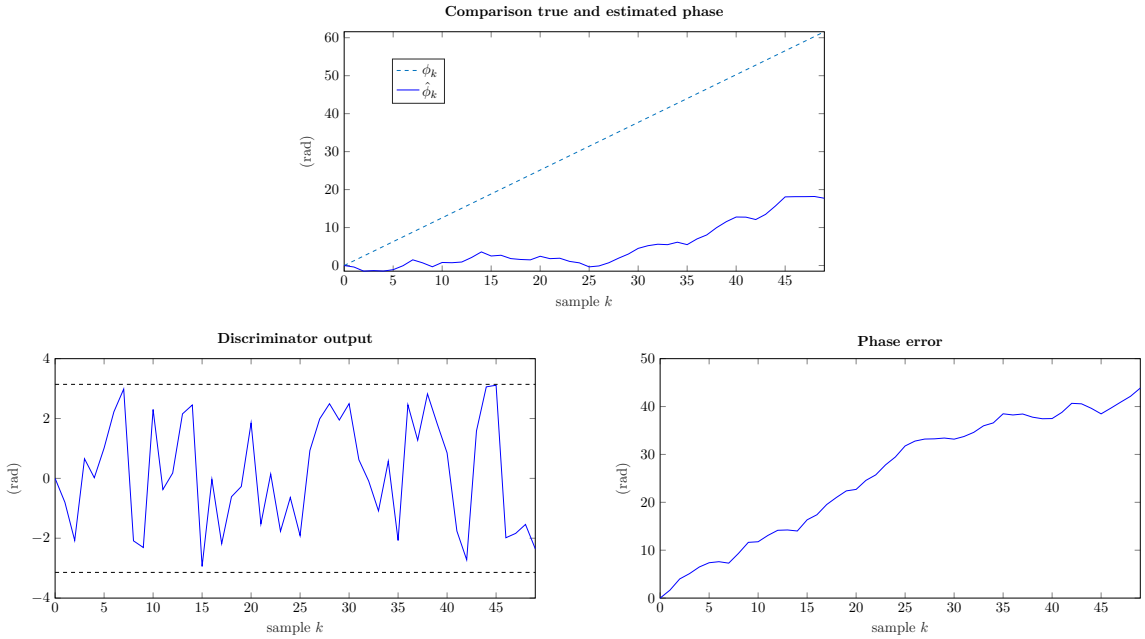


Figure 2.13: Drop-lock due to low SNR. ($T = 0.02$ s; $\sigma_n = 1$; $B_L = 10$ Hz; $\dot{\phi}_0 = \pi/25$ rad/sample; $\hat{\dot{\phi}}_0 = 0$ rad/sample; $C/N_0 = 17$ dB-Hz. Filter coefficients are calculated using [69, Tab. V])

tion is a multi-modal function, where each mode is centered around the equilibrium points spaced $k(\times 2\pi)$ radians. This modal periodicity is expected by using an ATAN2 discriminator. Finally, note that most of the slips are positive, which may be explained by a positive slope of the true phase input. An expression of number of positive/negative slips occurring can be found in [86, Eq.(47)], where a first-order loop is considered; however, it remains a good reference for higher-order too.

2.1.4.3 Slip metrics

In this section, we introduce some metrics that can be found in the literature and that we will mainly use to analyze the cycle slip phenomenon.

Mean time to first slip

A useful tool to understand the slipping behavior of the loop is the average time between cycle slips, known as *Mean Time to First Slip* (MTFS), or also mean time of loss of lock. The latter is defined as the time required for the loop phase error to reach the equilibrium line $\pm 2\pi$ ($\pm\pi$ for the Costas loop) for the first time from steady-state [71]. When cycle slips are independent events, the MTFS results to be the same as the average time between cycle slips. Otherwise, both do not have such relation [86]. Following [71], the MTFS for a first-order

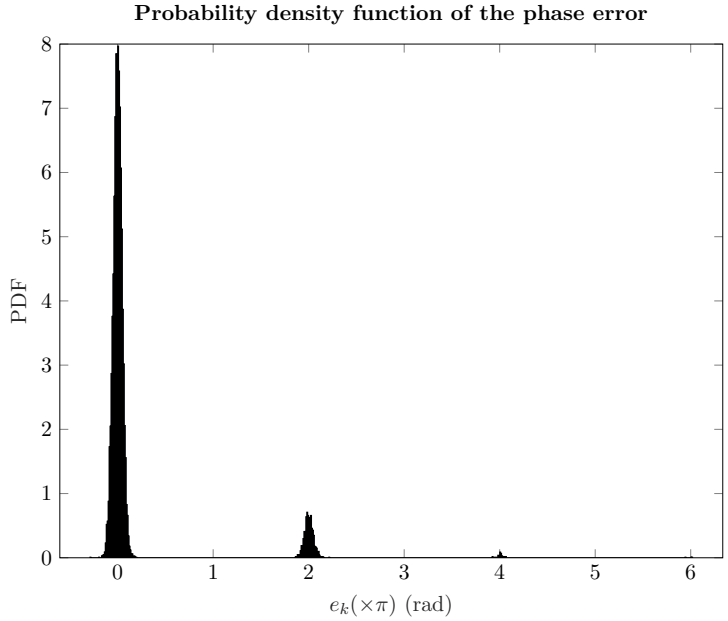


Figure 2.14: PDF of the phase error in presence of noise (at time 5 s). ($T = 0.02$ s; $\sigma_n = 1$; $B_L = 10$ Hz; $\dot{\phi}_0 = \pi/2$ rad/sample; $\hat{\dot{\phi}}_0 = 0$ rad/sample; $C/N_0 = 27$ dB-Hz; $M_c = 10000$ realizations. Filter coefficients are calculated using [69, Tab. V])

analog PLL with zero static phase error is

$$T(2\pi) = \frac{\pi^2 \mathrm{SNR}_L I_0^2(\mathrm{SNR}_L)}{2B_L}, \tag{2.40}$$

which is approximated for large SNR_L by

$$T(2\pi) \approx \frac{\pi}{4B_L} \exp(2\mathrm{SNR}_L). \tag{2.41}$$

It is named $T(2\pi)$ since it is calculated considering a pure PLL (ATAN2). Finally, the frequency of slipping cycles is equal to $1/T(2\pi)$ [71].

The frequency of slipping cycles, namely $1/T(2\pi)$ using (2.40), and the inverse of MTFS found empirically using a DPLL are compared and shown in Fig. 2.15. To be faithful with the Viterbi's analysis, a pure first-order DPLL with ATAN2 discriminator and constant input is used. Note that SNR_L is calculated as defined in (2.35), which is equivalent to $C/N_0/B_L$. As can be seen, even if the empirical and theoretical trends result to be the same, a discrepancy between the two curves is evident. The main reason for this difference may be due to the fact that Viterbi's formulation is based on analog PLL, contrarily to the empirical analysis.

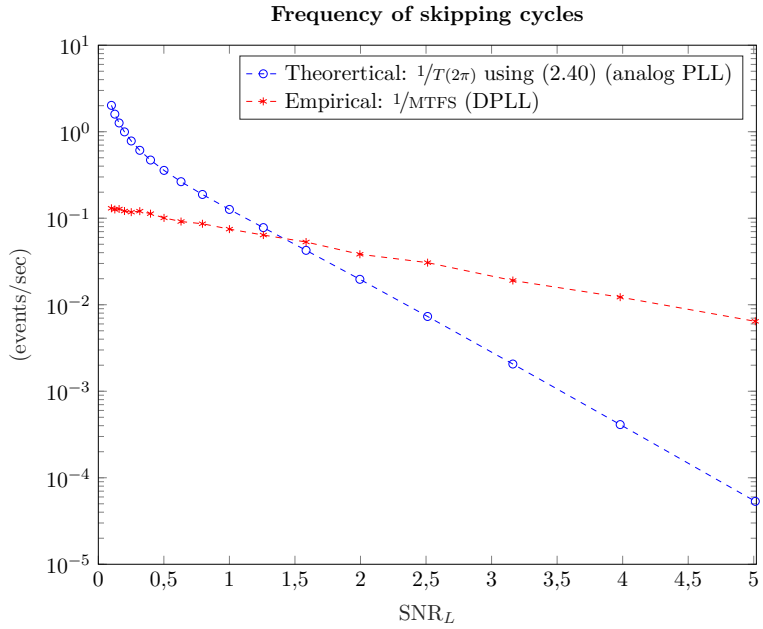


Figure 2.15: Empirical and theoretical comparison of "frequency of slipping cycles". (B_L = 10 Hz; T = 0.02 s; σ_n = 1; Empirical setting parameters: M_c = 2000; $\hat{\phi}_0$ = 0 rad; filter coefficients are calculated using (2.28) from [69])

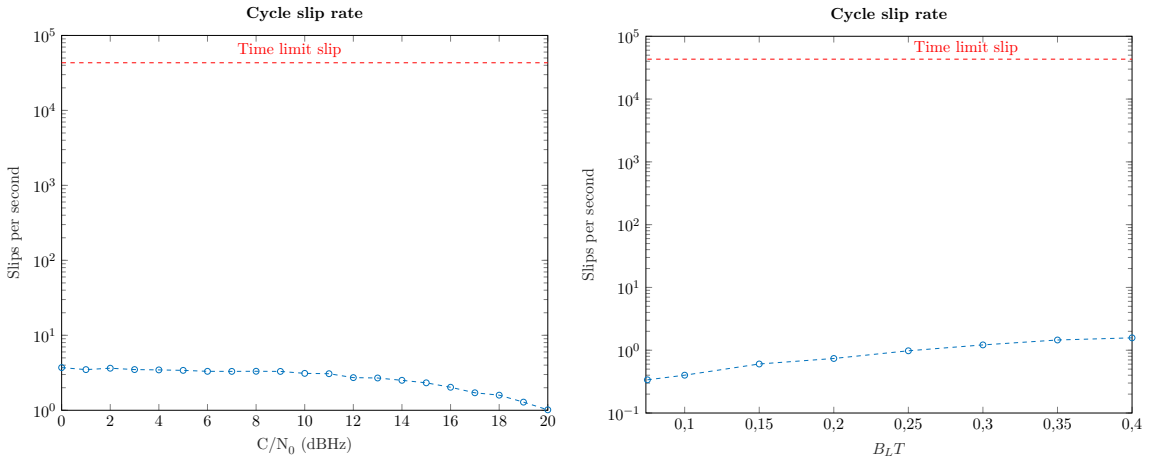
Cycle slip rate

Empirically, the cycle slip rate is estimated while fixing a number of cycle slips N_{cs} to be observed. As for the MTFS, according to Viterbi's criterion [71], a cycle slip is detected as soon as the phase error crosses a new equilibrium line. The slip rate, namely R_s, is defined as

$$R_s \overset{\text{def}}{=} \frac{1}{M_c} \sum_{n=1}^{M_c} \frac{N_{cs}}{T_{N_{\text{cs}}}(n)} \tag{2.42}$$

where M_c is the number of Monte-Carlo runs and $T_{N_{\text{cs}}}$ is the time necessary to observe these N_{cs} slips. This metrics is also evaluated while starting the track at steady-state [69]. In practice, a time limit of the simulation, namely T_{Limit}, is set: in case N_{cs} is not attained before T_{Limit}, the last occurred slip is considered in (2.42). Though the estimation can be degraded, this allows the simulation to be contained to a reasonable time duration. Finally, note that, differently to the MTFS, there is not a theoretical expression of the cycle slip rate in literature. Therefore, a comparison between the empirical and theoretical results cannot be done as for MTFS in Fig. 2.15.

The cycle slip rate is exemplified as a function of C/N_0 and B_L in Fig. 2.16 for a first-order DPLL with ATAN2 and ramp phase input. As expected, in Fig. 2.16b the cycle slip rate increases at higher loop bandwidths due to a major noise coming in the loop, and contrarily in Fig. 2.16a the rate decreases at higher the C/N_0 due to a clearer signal from noise.



(a) (b)

Figure 2.16: Cycle slip rate for a first-order DPLL with ATAN2. ($T = 0.02$ s; $\sigma_n = 1$; $M_c = 500$; $N_{cs} = 10$ slips; $T_{\text{Limit}} = 43200$ s (half-a-day); $\dot{\phi}_0 = \pi/30$ rad/sample; $\hat{\phi}_0 = 0$ rad; filter coefficients are calculated using (2.28) from [69]. Left: $B_L = 10$ Hz; Right: $C/N_0 = 10$ dB-Hz)

2.2 Robust phase tracking techniques

2.2.1 Overview

The stringent requirements such as robustness at low SNR given by phase tracking applications (for example the PPP [91], see section 1.3.2.2) are fundamental to obtain an accurate positioning. Unfortunately, the DPLL may be unable to fulfill them in hostile environments. The main reason is due to the fact that DPLL is designed considering a linear and Gaussian behavior of the phase error. However, in case of low SNR, this brings robustness issues due to nonlinearity of the discriminator, as analyzed in previous sections, namely 2.1.2, 2.1.3 and 2.1.4.

In order to overcome these limitations, four main approaches have been proposed in literature summarized in [92].

1. *Optimized DPLL (constant bandwidth)*: by the introduction of modifications to the DPLL, an optimized scheme of the latter is reached. Some solutions regard the optimization of the correlator with a *Pre-Detection Integration* (PDI) before feeding the discriminator (as done in GNSS receiver, see section 5.2.2.1) to reduce noise [61, 93], the optimization of the phase discriminator using a cooperative strategy between the FLL and DPLL (so-called FLL-assisted DPLL) [94], and finally the optimization of the filter by improving the continuous-discrete gains conversion [95]. Interesting are also the studies on new DPLL architectures based on phase unwrapping algorithms to correct cycle slips due to phase noise in low C/N_0 such as in here [96, 97].

2. *Variable bandwidth DPLL*: as often said, a trade-off on loop bandwidth choice is necessary to both accurately track high phase dynamics and reject as much noise as possible. Therefore, adaptive tracking techniques have been developed such as *Fast adaptive bandwidth DPLL* (FAB-DPLL) [98] and *Projected loop bandwidth* (PLB-DPLL) [99], where the phase errors at the discriminator output are automatically adjusted based on C/N_0 estimates and dynamics stress prediction, or *Fuzzy Logic* (FL-DPLL) [100, 101], which uses fuzzy logic automatic control based on phase dynamics.

3. *Kalman Filter (KF)-based DPLL*: following the same philosophy of point 2., the KF-based DPLL adjusts the loop bandwidth when linear-Gaussian assumptions of the state-space model are verified in practice. More insights on this technique is presented in section 2.2.3.

4. *Innovative tracking loop*: when the KF philosophy cannot be applied due to the nonlinear or non-Gaussianity nature of the propagation effects, innovative tracking techniques are proposed. A notable dissertation of nonlinear filters beyond the KF is presented in [102]. Promising techniques for carrier-phase tracking are: 1) *Particle Filter* (PF)-based tracking [103, 104, 64, 54], where a set of particles is used to represent the posterior distribution of a stochastic process (no linearization as in the *Extended Kalman Filter* (EKF)-based DPLL [105, 106, 107]). 2) *Interactive Multiple Model* (IMM) phase estimator [108, 109, 110], which is based on a combination of different KF-based techniques, each of them configured to various dynamics model. 3) *Open Loop* (OL) estimator [111, 112, 54], which is well-suited for both batch and on-line processing implementation.

Note that most of these tracking algorithms (*e.g.*, KF techniques and PF) are based on the Bayesian philosophy. Therefore, more insights on this topic will be discussed in the next section.

2.2.2 Bayesian philosophy

The *Bayesian philosophy* is considered as a consistent framework to deal with uncertainties, which are expressed in terms of probability [113, 114, 115]. The method firstly formulates a *prior* (or predicted) distribution over the unknown parameters of the model, which is meant to capture beliefs about the situation before seeing the data. After observing some data, the Bayes' rule (name took by its inventor Reverend Thomas Bayes and presented in his original essay [116]) is applied to obtain a *posterior* (or updated) distribution for these unknowns, which takes into account both the prior and the data. From this posterior distribution, the predictive distributions for future observations can be computed.

Mathematically, the nonlinear filtering problem can be defined through a *State-Space Model* (SSM) formed by the dynamics and measurement equations [117, Chap.5]. In our phase tracking framework, the former in the discrete-time domain is expressed as

$$\boldsymbol{x}_k = \boldsymbol{f}_{k-1}(\boldsymbol{x}_{k-1}, \boldsymbol{w}_{k-1}) \tag{2.43}$$

and the latter equation is

$$\boldsymbol{y}_k = \boldsymbol{h}_k(\boldsymbol{x}_k, \boldsymbol{v}_k), \tag{2.44}$$

which is a generalization of the input signal (2.1). $\boldsymbol{f}_{k-1}(\cdot)$ and $\boldsymbol{h}_k(\cdot)$ are two known possibly nonlinear functions; $\boldsymbol{w}_{k-1}$ and $\boldsymbol{v}_k$ are referred to as process and measurement noises (assumed mutually independent stochastic processes). Let denote by $\boldsymbol{Y}_k = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_k]$ the set of observations till the instant k. Assuming that the initial PDF, namely $f(\boldsymbol{y}_0)$, is known, as said before the PDF $f(\boldsymbol{y}_k|\boldsymbol{Y}_k)$ is obtained recursively in two stages: prediction and update. The former is obtained via Chapman-Kolmogorov equation [118, 119, Chap.1],

$$f(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}) = \int f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) f(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1}) d\boldsymbol{x}_{k-1}, \tag{2.45}$$

while the latter is calculated using Bayes' rule, namely

$$f(\boldsymbol{x}_k|\boldsymbol{Y}_k) = \frac{f(\boldsymbol{y}_k|\boldsymbol{x}_k) f(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1})}{f(\boldsymbol{y}_k|\boldsymbol{Y}_{k-1})}. \tag{2.46}$$

Note that $f(\boldsymbol{y}_k|\boldsymbol{x}_k)$ represents the likehood function. Neglecting the normalization term, namely $f(\boldsymbol{y}_k|\boldsymbol{Y}_{k-1})$, we can write

$$f(\boldsymbol{x}_k|\boldsymbol{Y}_k) \propto f(\boldsymbol{y}_k|\boldsymbol{x}_k) f(\boldsymbol{x}_k|\boldsymbol{Y}_{k-1}). \tag{2.47}$$

The knowledge of the posterior density $f(\boldsymbol{x}_k|\boldsymbol{Y}_k)$ enables one to compute an optimal state estimate with respect to any criterion (*e.g.*, *Minimum Mean-Square Error* (MMSE)) [118, Chap.1]. However, the Bayesian filtering is analytically tractable only in cases that: 1) the marginalization over $\boldsymbol{x}_{k-1}$ in (2.45) is analytically tractable; 2) the resulting posterior distribution in (2.47), namely $f(\boldsymbol{x}_k|\boldsymbol{Y}_k)$, has the same functional form as the previous step, namely $f(\boldsymbol{x}_{k-1}|\boldsymbol{Y}_{k-1})$, allowing the procedure to be iterative [120]. Unfortunately, these conditions are satisfied for only a very limited class of models [102]. In such situations, we usually resort to approximation schemes, and these fall broadly into two classes, according to whether they rely on stochastic or deterministic approximations. The former such as *Markov Chain Monte Carlo* (MCMC) have the property that given infinite computational resource, they can generate exact results, and the approximation arises from the use of a finite amount of processor time [121, Chap.11]. The latter, instead, are based on analytical approximations to the posterior distribution, giving an accurate (but not exact) solution, and they are usually used in large-scale problems due to their low computational costs [121, Chap.10]. In our thesis, we will focus our attention on Variational inference which is a leading approach for approximate inference. More details about the latter are described in chapter 3.

In a linear-Gaussian case, the optimal algorithm for recursive Bayesian state estimation becomes the so-called Kalman filter. A proper dissertation of this technique is presented in the next section.

2.2.3 Kalman Filter-based DPLL

Rudolf E. Kálmán published his filtering algorithm in 1960 in his seminal work [122]. As said, the *Kalman Filter* (KF) is based on Bayesian philosophy and gives a systematic and optimal solution to the estimation of the time-varying unknown state of a system. Considering our state-space model, namely (2.43) and (2.44), and the linear-Gaussian assumptions, the principle of the KF is presented in the following section.

2.2.3.1 Fundamentals of KF

Under the Kalman assumptions of linearity, namely the two functions $\boldsymbol{f}_{k-1}(\cdot)$ and $\boldsymbol{h}_k(\cdot)$ and Gaussianity of the noise terms in (2.43) and (2.44), we have that both prior and posterior densities in (2.45) and (2.47) are Gaussian distributed [118, 123, Chap.1]. Computationally, we have

$$\boldsymbol{x}_k = \begin{bmatrix} \phi_k \\ \dot{\phi}_k \\ \vdots \\ \overset{n}{\dot{\phi}}_k \end{bmatrix} = \boldsymbol{A}_{k-1}\boldsymbol{x}_{k-1} + \boldsymbol{w}_{k-1}, \tag{2.48}$$

where $[\boldsymbol{x}_k]_1$ (or ϕ_k) is the phase of the input signal (2.1) and $\overset{n}{\dot{\phi}}_k$ is the nth derivative of the phase discretized at the time instant kT, and

$$\boldsymbol{y}_k = \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{v}_k. \tag{2.49}$$

$\boldsymbol{A}_{k-1}$ and $\boldsymbol{H}_k$ are known matrices defining linear functions, the so-called transition and measurements matrix, respectively. $\boldsymbol{w}_{k-1}$ and $\boldsymbol{v}_k$ are mutually independent zero-mean white Gaussian with covariance matrices $\boldsymbol{Q}_{k-1}$ and $\boldsymbol{R}_k$, respectively. The Gaussian mean and covariance in prior and posterior steps are computed, respectively, as follows [118, Chap.1]

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{A}_{k-1}\hat{\boldsymbol{x}}_{k-1|k-1}, \tag{2.50a}$$
$$\boldsymbol{P}_{k|k-1} = \boldsymbol{Q}_{k-1} + \boldsymbol{A}_{k-1}\boldsymbol{P}_{k-1|k-1}\boldsymbol{A}_{k-1}^{\intercal}, \tag{2.50b}$$
$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_{k|k-1}), \tag{2.50c}$$
$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{G}_k\boldsymbol{K}_k^{\intercal}, \tag{2.50d}$$

where

$$\boldsymbol{G}_k = \boldsymbol{H}_k\boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^{\intercal} + \boldsymbol{R}_k, \tag{2.51}$$

is the covariance of the innovation term $(\boldsymbol{y}_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_{k|k-1})$, and

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^{\intercal}\boldsymbol{G}_k^{-1} \tag{2.52}$$

is the Kalman gain. Note that with (2.51) and (2.52), covariance update (2.50d) can be written as

$$\boldsymbol{P}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k)\boldsymbol{P}_{k|k-1}, \tag{2.53}$$

where $\boldsymbol{I}$ is the identity matrix. As can be seen, the KF recursively estimates the mean and covariance of the Gaussian posterior. Eventually, in the case of low arithmetic precision caused by numerical stability, the posterior error covariance is derived through the Joseph formula [124, 125, Chap.7]

$$\boldsymbol{P}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k)\boldsymbol{P}_{k|k-1}(\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k)^{\intercal} + \boldsymbol{K}_k \boldsymbol{R}_k \boldsymbol{K}_k^{\intercal}. \tag{2.54}$$

2.2.3.2 KF-based DPLL

Due to the nonlinearity of the observations (2.1), the KF is placed after the discriminator where its output, the state error, has a linear dependence with the state (to be estimated). In this way, the loop configuration perfectly follows the architecture of traditional DPLL in which the received signal is correlated with the local replica and the error metrics is obtained at the discriminator output. Considering our state (2.48) and innovation term e_k (discriminator output), the KF-based DPLL architecture is shown in Fig. 2.17. This tracking loop has received significant attention in literature, especially for its robustness to phase tracking problem [126, 127]. Furthermore, it is interesting to note the parallelism between the KF and the traditional carrier tracking loop, namely the DPLL. In fact, the latter is shown to be a particular case of KF, in which the filtering coefficients are all set to be constant [128, 129]. Contrarily, the KF provides a more general framework, where the processing parameters are automatically adjusted so as to minimize the mean square error between the measurement and the prediction [124, 130, Chap.5, Chap.5].

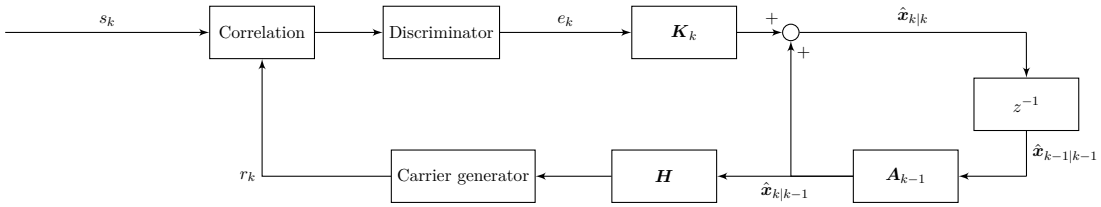


Figure 2.17: KF-based DPLL architecture

Note that the KF approach requires the initialization of state vector and error covariance, respectively $\boldsymbol{x}_{0|0}$ and $\boldsymbol{P}_{0|0}$, and the knowledge of the noise statistics, namely $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$. We focus our attention on presenting the only latter since we mainly work at steady-state in the next chapters. Therefore, considering a third-order loop as example, we have that the measurement matrix $\boldsymbol{H}$ is

$$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}. \tag{2.55}$$

With that, the measurement covariance matrix $\boldsymbol{R}$ reduces to a scalar. In such case, the latter

can be computed as [131, 5, Eq.(A-13)]

$$\boldsymbol{R} = \sigma_{n_\phi}^2 = \frac{1}{2C/N_0T}\left(1 + \frac{1}{2C/N_0T}\right), \tag{2.56}$$

which is found using a pure DPLL. In case of a Costas loop, the approximated variance of the phase noise $\sigma_{n_\phi}^2$ is found as in [132, Eq.(19)]. Finally, as it is not relevant at the moment, a detailed description of the choice of noise covariance matrix $\boldsymbol{Q}$ is later presented in section 4.3.1.1.

Example of a first-order KF-based DPLL

An example of carrier-phase tracking results is shown in Fig. 2.18. A first-order KF-based DPLL with ATAN2 discriminator is considered. Furthermore, a noise-free observation and a step phase dynamics is used. Note that the loop bandwidth is adjusted through the time-varying Kalman gain in (2.28).

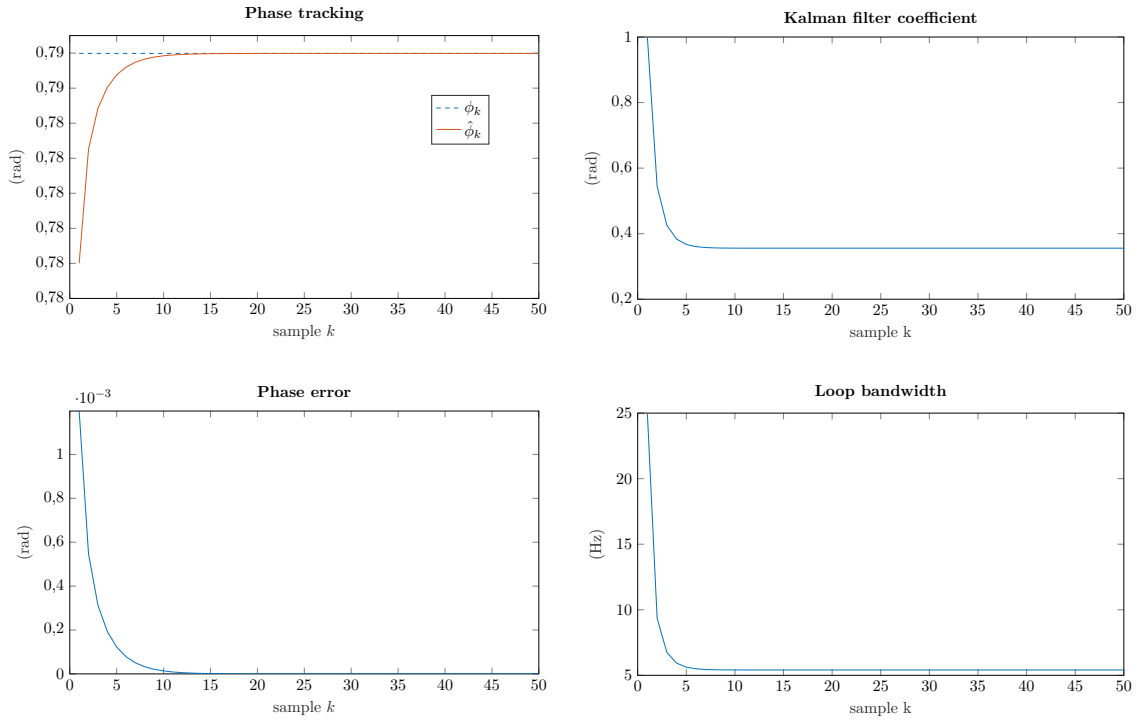


Figure 2.18: Carrier-phase tracking using a KF-based DPLL. ($T = 0.02$ s; $\phi_0 = \pi/4$ rad; $\sigma_n = 1$; SNR = 20 dB; $\phi_{0|0} = 0$ rad; $P_{0|0} = \pi^2/3$ rad^2 (as suggested in [133]); A = 1; H = 1; Q = $(0.01\pi)^2$ rad^2; R as defined in (2.56))

2.3 Summary

In this chapter, the "father" of all carrier tracking techniques is presented, namely the DPLL. In particular, its structure is examined by analyzing each block that is part of the loop. Moreover, a linear and nonlinear analysis of the tracking loop has been shown while deriving the phase variance, the phase error PDF, and finally, the investigation of the cycle slip phenomenon. We showed the limitations of this traditional architecture, in particular related to its capability to work in harsh environments. In fact, at low SNR, we have seen that the DPLL lacks robustness since it is designed considering a linear and Gaussian behavior of the phase error. Therefore, to cope with the technical challenges to obtaining robust carrier tracking, an overview of innovative techniques has been presented. Since most of them are based on Bayesian philosophy, fundamentals of the latter are introduced. Specifically, in our study, we focused our attention on the well-known KF-based DPLL, which is an optimal approach for estimating time-varying parameters (*i.e.*, loop bandwidth). The latter and the traditional DPLL will be compared with our proposed algorithms, which are presented in the next chapters.

Chapter 3

RVB algorithm in case of slow dynamics

A new robust carrier-phase tracking algorithm is here presented. The latter is based on Variational Bayesian approximation in the optimal Bayesian filtering problem due to low computational costs and suitable applicability to large-scale problems of this methodology compared to the sampling techniques. Firstly, the state-space model of our case study is introduced. Afterwards, the local VB inference and its restricted version are used to obtain a closed-form solution of the estimator. Finally, numerical simulations of the proposed algorithm compared with conventional techniques such as DPLL and KF-based DPLL are implemented, showing improvements in terms of precision of estimation and cycle slipping.

Contents

3.1 State-space model

Before introducing the proposed algorithm, a *state-space model* of the carrier-phase tracking problem is needed. As a matter of fact, we already presented it by using a general high-order formulation into the Bayesian filtering problem (see section 2.2.2), and then linearized it with the KF assumptions (see section 2.2.3.1). Herein, instead, we present the SSM as it is through

both measurement and dynamics (or state) equations, considering in a first place slow phase dynamics.

3.1.1 Measurement equation

In this study, the measurement equation is represented by the signal as described in section 2.1.1.1, as well as the scenario. To summarize, the assumptions are: 1) about the input signal: navigation message is neglected and an AWGN is considered; 2) about the GNSS receiver scenario: the input signal is that of the Prompt correlator, and a perfect synchronization of either delay or both delay and frequency is supposed (with DLL-FLL or DLL only with frequency compensation from acquisition assumed to slowly vary). That said, information from (2.1) and (2.2) are here reformulated as a likelihood function, which is helpful for our upcoming Bayesian analysis. The latter is the probability density of the data, viewed as a function of the parameters, and it is central to the process of estimating the unknown parameters [134]. Using (2.1) and (2.2), the likelihood is expressed as

$$\begin{aligned} f(s_k|\phi_k,\alpha,\sigma_n^2) &= \frac{1}{\pi\sigma_n^2}\exp\Big\{-\frac{|s_k-\alpha e^{j\phi_k}|^2}{\sigma_n^2}\Big\} \\ &= \frac{1}{\pi\sigma_n^2}\exp\Big\{-\frac{1}{\sigma_n^2}\Big[|s_k|^2+\alpha^2-2\alpha|s_k|\cos(\phi_k-\psi_k)\Big]\Big\}, \end{aligned} \tag{3.1}$$

with $\psi_k = \text{atan2}(\Im\{s_k\},\Re\{s_k\})$ the angle that lies between $[-\pi,\pi]$. In this work, we assume to know the value of the amplitude $\alpha > 0$ and noise power σ_n^2, we will thus omit them in the conditional terms. The latter can be obtained for instance from a lock estimator. Accordingly, the likelihood function (3.1) can be simply written $f(s_k|\phi_k)$. Assuming a fixed sequence of data s_k, the sensor factor associated to the likelihood (3.1) is

$$S(\phi_k) \overset{\text{def}}{=} f(s_k|\phi_k) \propto \exp\Big\{\beta_k\cos(\phi_k-\psi_k)\Big\}, \tag{3.2}$$

where

$$\beta_k = \frac{2\alpha|s_k|}{\sigma_n^2}. \tag{3.3}$$

In (3.2), we recognize a Von Mises (or Tikhonov) distribution with concentration parameter β_k and mean direction ψ_k. Considering (2.1), β_k is distributed according to a Rice distribution with PDF [70]

$$f\Big(\beta_k|\alpha,\frac{\sigma_n^2}{2}\Big) = \frac{2\beta_k}{\sigma_n^2}\exp\left(\frac{-(\beta_k^2+\alpha^2)}{\sigma_n^2}\right)I_0\left(\frac{2\beta_k\alpha}{\sigma_n^2}\right), \tag{3.4}$$

where I_0 is the modified Bessel function of first kind at zeroth order. The theoretical and empirical PDFs of β_k are compared when varying the SNR as shown in Fig.3.1. As can be seen, there is a perfect match between the theoretical and empirical PDFs, proving the veracity of expression in (3.4).

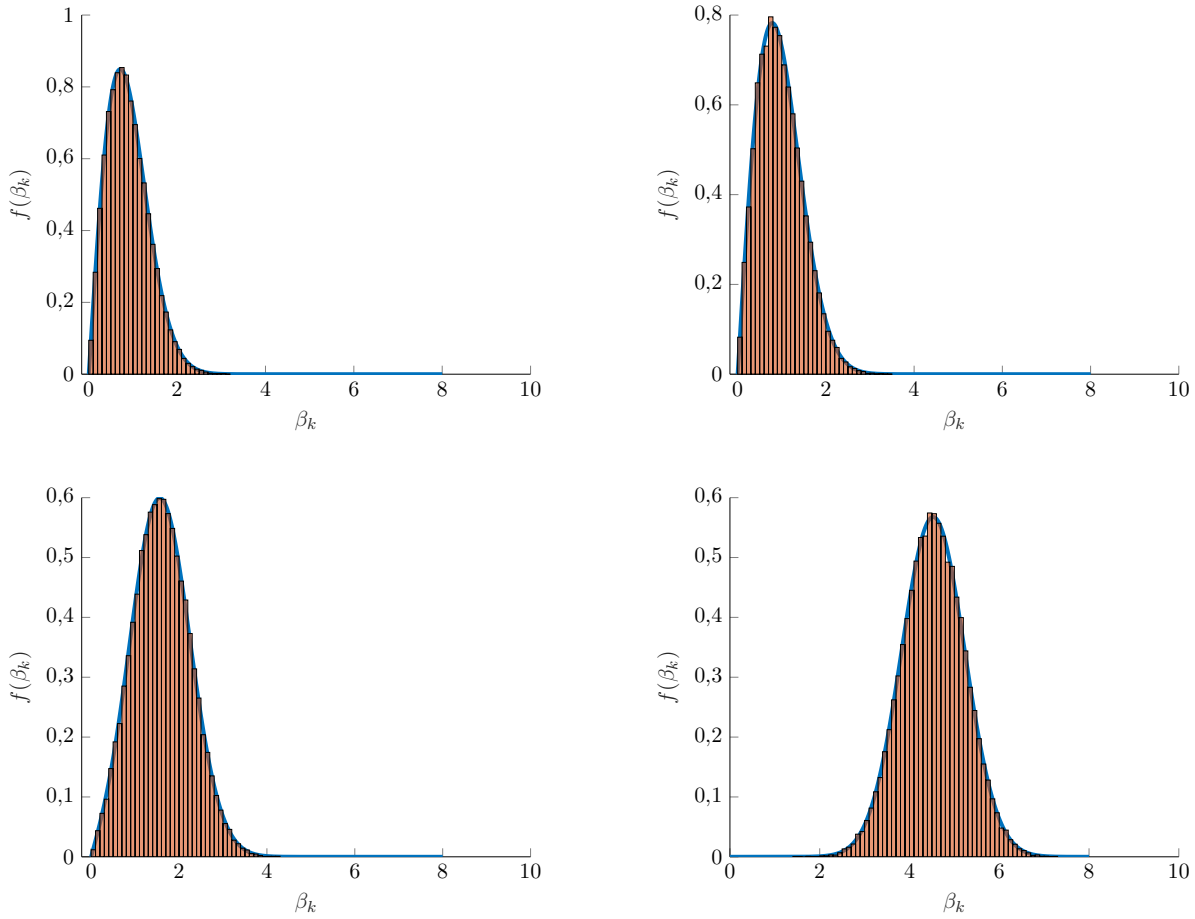


Figure 3.1: PDF (3.4) of the concentration parameter β_k. (Up-left: $C/N_0 = 0$ dB-Hz; up-right: $C/N_0 = 0$ dB-Hz; bottom-left: $C/N_0 = 20$ dB-Hz; bottom-right: $C/N_0 = 30$ dB-Hz. $T = 0.02$ s; $\phi_0 = 0.5$ rad; $\sigma_n^2 = 1$)

3.1.2 Dynamics equation

Since we are interested in estimating the *absolute* phase ϕ_k recursively in time, it is necessary to assume a phase dynamic model that, hopefully, prevent from any ambiguity to arise during estimation (*i.e.*, no cycle slip). To that end, the phase evolution is modeled by a *Markov Random Field* (MRF) [135]. The latter has been widely used as Bayesian prior in many phase processing problems; [136, 137, 138] are just a few of them. Here, we assume a first-order Gaussian MRF in order to ensure some smoothness in the estimated phase sequence [137], *i.e.*,

$$\phi_k = \phi_{k-1} + w_k, \tag{3.5}$$

where $w_k \sim \mathcal{N}(0, \sigma_\phi^2)$ is a white Gaussian noise. More specifically, the initial state ϕ_1 is supposed to be uniformly distributed over a symmetric set $\mathcal{I}$ (its choice is discussed later in section 3.2.2), while the phase at time $k > 1$ is thought to be Gaussian distributed with mean

ϕ_{k-1} and variance σ_ϕ^2, namely,

$$\phi_1 \sim \mathcal{U}_{\mathcal{I}}, \tag{3.6a}$$

$$\phi_k|\phi_{k-1} \sim \mathcal{N}(\phi_{k-1}, \sigma_\phi^2), \tag{3.6b}$$

which leads to the following PDFs

$$f(\phi_1) \propto \mathbb{I}_{\mathcal{I}}(\phi_1), \tag{3.7a}$$

$$f(\phi_k|\phi_{k-1}, \sigma_\phi^2) = \frac{1}{\sqrt{2\pi\sigma_\phi^2}} \exp\left\{-\frac{[\phi_k - \phi_{k-1}]^2}{2\sigma_\phi^2}\right\}. \tag{3.7b}$$

3.2 RVB tracking algorithm

Once our SSM introduced in case of carrier-phase tracking, the proposed algorithm is here explained. The section is described as follows: initially, the optimal Bayesian filtering problem of our SSM is presented; then, the *Variational Bayesian* (VB) inference is used to obtain a tractable solution; finally, the *Restricted VB* (RVB) approximation is applied, giving us a closed-form expression of the estimator [1]. Note that the methodology is the same as used in [139], where the VB phase tracking is applied to correlated dual-frequency measurements with slow dynamics.

3.2.1 Distributional approximations in Bayes filtering problem

As already explained in section 2.2.2, the problem of inferring the state variable is described by (3.2) and (3.7b), namely

$$f(s_k|\phi_k) \quad \text{and} \quad f(\phi_k|\phi_{k-1}).$$

Denoting by $\boldsymbol{S}_k = [s_1, \ldots, s_k]$ the set of observations till the instant k, the optimal Bayes filtering that iteratively evaluates the filtering distribution $f(\phi_k|\boldsymbol{S}_k)$ is obtained by alternating between the equations (2.45) and (2.47). As mentioned in section 2.2.2, the Bayesian problem is analytically tractable only in stringent conditions, therefore approximation schemes are usually used as alternate. A typical way to proceed is to use distribution approximation [118]

$$f(\phi_k|\boldsymbol{S}_k) \sim \tilde{f}(\phi_k|\boldsymbol{S}_k). \tag{3.8}$$

Various approximation strategies have been developed, such as certainty equivalence approximation, Laplace's approximation, fixed-form minimum distance approximation, and VB approximation [140, Chap.2]. In particular, we focus our attention on the latter, applying a local VB inference where the functional form of $\tilde{f}(\phi_k|\boldsymbol{S}_k)$ is restored at each time k.

3.2.2 Local VB filtering

The VB approximation is applied in various applications, also known as Ensemble Learning in control engineering [141, 142] or naïve Mean Field Theory in statistical physics [143, 144]. Well-known papers on this topic are [145, 146, 147, 148, 149]; however, for our dissertation we will mostly refer to the work of [120]. The local VB approach consists in locally imposing conditional independence between ϕ_k and ϕ_{k-1} [120]

$$\tilde{f}(\phi_k, \phi_{k-1}|\boldsymbol{S}_k) = \tilde{f}(\phi_k|\boldsymbol{S}_k)\tilde{f}(\phi_{k-1}|\boldsymbol{S}_k), \tag{3.9}$$

where $\tilde{f}(\cdot)$ refers to the approximated posterior distribution. The latter are then chosen to minimize the *Kullback-Leibler* (KL) divergence [121, 120, Chap.10]. A geometric interpretation of the KL divergence into the VB inference is presented in [150]. The prediction distribution can be expressed as

$$\tilde{f}(\phi_k|\boldsymbol{S}_{k-1}) \propto \exp\Big(\mathrm{E}_{\tilde{f}(\phi_{k-1}|\boldsymbol{S}_k)}\Big[\ln(f(\phi_k|\phi_{k-1}))\Big]\Big), \tag{3.10}$$

with

$$\tilde{f}(\phi_{k-1}|\boldsymbol{S}_k) \propto \exp\Big(\mathrm{E}_{\tilde{f}(\phi_k|\boldsymbol{S}_k)}\Big[\ln(f(\phi_k|\phi_{k-1}))\Big]\Big) \times \tilde{f}(\phi_{k-1}|\boldsymbol{S}_{k-1}), \tag{3.11}$$

where $\mathrm{E}_{\tilde{f}(\boldsymbol{x}_{/i}|\boldsymbol{S})}\Big[g(\boldsymbol{x})\Big]$ [1] denotes the expected value of the function $g(\boldsymbol{x})$ with respect to the function $f(\boldsymbol{x})$. Accordingly, the filtering distribution is now expressed as

$$\tilde{f}(\phi_k|\boldsymbol{S}_k) \propto f(\boldsymbol{s}_k|\phi_k)\tilde{f}(\phi_k|\boldsymbol{S}_{k-1}). \tag{3.12}$$

At this stage, the data update equation (3.12) describes an implicit equation in $\tilde{f}(\phi_k|\boldsymbol{S}_k)$ since the prediction distribution depends itself via (3.11) of the latter. To have a closed-form solution, a second approximation is applied [120]. This is done through the *RVB approximation* which finally replaces the distribution $\tilde{f}(\phi_{k-1}|\boldsymbol{S}_k)$ in (3.10) by the fixed posterior distribution $\tilde{f}(\phi_{k-1}|\boldsymbol{S}_{k-1})$. This way, the functional forms of both the prediction and filtering distributions are now preserved from one iteration to another. Furthermore, the expectations in (3.10) are now taken wrt $\tilde{f}(\phi_{k-1}|\boldsymbol{S}_{k-1})$ instead of $\tilde{f}(\phi_{k-1}|\boldsymbol{S}_k)$, namely

$$\tilde{f}(\phi_k|\boldsymbol{S}_{k-1}) \propto \exp\Big(\mathrm{E}_{\tilde{f}(\phi_{k-1}|\boldsymbol{S}_{k-1})}\Big[\ln(f(\phi_k|\phi_{k-1}))\Big]\Big). \tag{3.13}$$

Using the sensor factor (3.2) and the phase dynamics distributions (3.7), it can be shown that, following the same path as in [139], the RVB filtering has the following closed-form expression [1]

[1] $\boldsymbol{x}_{/i}$ denotes the complement of x_i in $\boldsymbol{x}$ (e.g., $\boldsymbol{x}_{/1} = x_2$).

- Prediction and data update for $k = 1$

$$\tilde{f}(\phi_1|S_0) \overset{\text{def}}{=} f(\phi_1), \tag{3.14a}$$

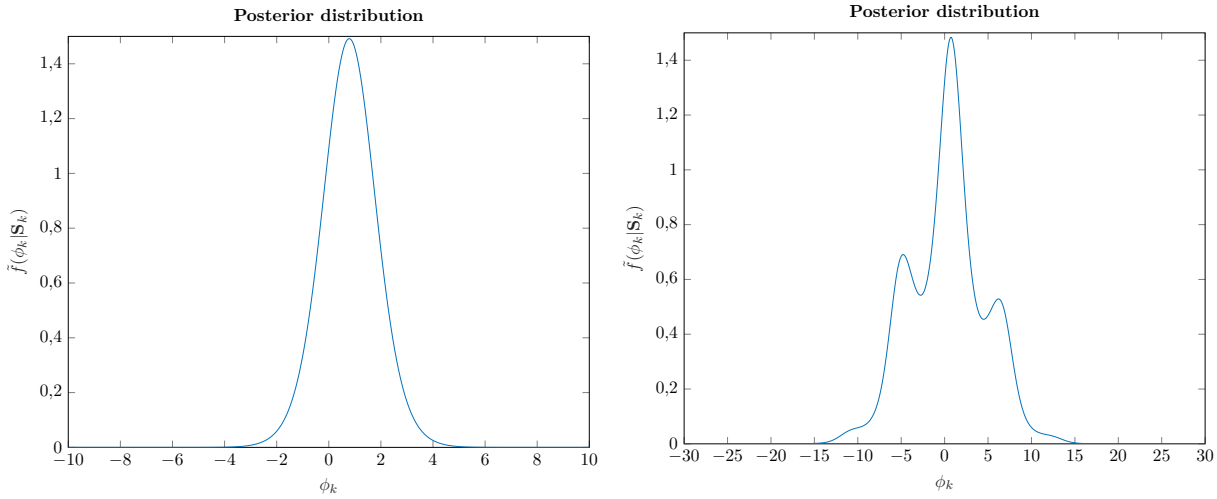
$$\tilde{f}(\phi_1|S_1) \propto f(\phi_1)\exp\Big\{\beta_1 \cos(\phi_1 - \psi_1)\Big\}. \tag{3.14b}$$

- Prediction and data update for $k > 1$

$$\phi_k|\boldsymbol{S}_{k-1} \underset{\tilde{f}}{\sim} \mathcal{N}\Big(\mathrm{E}_{\tilde{f}(\phi_{k-1}|\boldsymbol{S}_{k-1})}[\phi_{k-1}], \sigma_\phi^2\Big), \tag{3.15a}$$

$$\tilde{f}(\phi_k|\boldsymbol{S}_k) \propto \exp\left\{\beta_k \cos(\phi_k - \psi_k) - \frac{\Big[\phi_k - \mathrm{E}_{\tilde{f}(\phi_{k-1}|\boldsymbol{S}_{k-1})}[\phi_{k-1}]\Big]^2}{2\sigma_\phi^2}\right\}. \tag{3.15b}$$

Considering the expression (3.14b), we choose the interval $\mathcal{I} = [-\pi, \pi]$ which ensures that the VB filtering distribution at $k = 1$ results a Von Mises (or Tikhonov) distribution [81, Chap.3], thus simplifying the estimation of $\hat{\phi}_k^{\text{rvb}}$. Furthermore, it is worth noticing that for $k > 1$ the functional form of the VB filtering distribution (3.15b) is not new in carrier-phase estimation, as reported in [151, 152, 153]. It is a periodic function formed by the product of a Von Mises distribution (with mean direction ψ_k and concentration parameter β_k) and a Gaussian distribution (with mean $\mathrm{E}_{\tilde{f}(\phi_{k-1}|S_{k-1})}[\phi_{k-1}]$ and variance σ_ϕ^2) [154, 139]. Due to the periodic component multiple modes can arise when $\beta_k\sigma_\phi^2 > 1$ and thus lead to cycle slips. The multimodality of the VB filtering distribution is recalled in Appendix of [139]. An example of unimodal and multimodal distribtuion of (3.15b) is shown in Fig. 3.2. Typically,



(a) $\beta_k\sigma_\phi^2 < 1$ (b) $\beta_k\sigma_\phi^2 > 1$

Figure 3.2: Example of the posterior distribution (3.15b). ($T = 0.02$ s; $K = 500$ samples; $C/N_0 = 10$ dB-Hz; $\phi_0 = \pi/4$ rad)

to deal with multimodality problems, techniques based on Gaussian sum approximations are developed [155, 156]. However, the latter are precise enough for high concentration parameters

β_k, and much less accurate for lower values [81, pag.38]. As a consequence of this, the design of the estimation scheme is pursued using the filtering distribution as it is.

3.2.3 RVB algorithm in case of slow dynamics

According to (3.15b), the recursive propagation of the posterior density requires exclusively the evaluation of the mean of ϕ_k wrt the approximated posterior distribution $\tilde{f}(\phi_k|S_k)$. Therefore, it is necessary and sufficient to derive the *Minimum Mean Square Error* (MMSE) estimator of ϕ_k wrt the VB posterior [157]. The latter is referred to, in the following, as the RVB estimator

$$\hat{\phi}_k^{\text{rvb}} \stackrel{\text{def}}{=} \mathrm{E}_{\tilde{f}(\phi_k|\boldsymbol{S}_k)}[\phi_k] = \int \phi_k \tilde{f}(\phi_k|\boldsymbol{S}_k)(\phi_k) d\phi_k. \tag{3.16}$$

Note that once (3.16) is derived, the entire VB filtering distribution is known and one could define other estimators (*e.g.*, the Maximum A Posteriori (MAP) estimator). It is shown in Appendix A that the RVB estimator can be expressed as [1]

- $\hat{\phi}_k^{\text{rvb}}$ for $k = 1$

$$\hat{\phi}_1^{\text{rvb}} = -2\frac{\sum_{q=1}^{\infty}(-1)^q I_q(\beta_1)\frac{\sin(q\psi_1)}{q}}{I_0(\beta_1)}, \tag{3.17}$$

- $\hat{\phi}_k^{\text{rvb}}$ for $k > 1$

$$\hat{\phi}_k^{\text{rvb}} = \hat{\phi}_{k-1}^{\text{rvb}} + 2\sigma_\phi^2 \frac{\sum_{q=1}^{+\infty} q I_q(\beta_k)\sin[q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})]e^{-\frac{q^2\sigma_\phi^2}{2}}}{I_0(\beta_k) + 2\sum_{q=1}^{+\infty} I_q(\beta_k)\cos[q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})]e^{-\frac{q^2\sigma_\phi^2}{2}}}. \tag{3.18}$$

where the $I_q(\cdot)$'s are the modified Bessel functions of the first kind at q-th order. Note that since $I_q(\beta_k)$ decreases rapidly with respect to q, an implementable form of (3.17) and (3.18) is obtained by truncating the sum involved for $q = 1, ..., q_{\max}$, where $q_{\max}$ is the highest chosen order. To have a visual understanding of the recursive procedure, the block diagram of the RVB estimator is shown in Fig. 3.3. With the term $f_{\text{n.l.}}(\cdot)$ we indicate the nonlinear function expressed in the second term of (3.18). Note that the RVB algorithm uses not only the information brought by the phase of s_k but also from its modulus. Interestingly, we can see a similarity between (3.18) and the traditional KF update state expression in (2.50c). As a matter of fact, (3.18) is the sum of the predicted state estimate $\hat{\phi}_{k-1}^{\text{rvb}}$ plus a function of the innovation term $\psi_k - \hat{\phi}_{k-1}^{\text{rvb}}$ that is nonlinear (unlike the Kalman gain in (2.52)). This similarity can be immediately seen when analyzing the RVB estimator (3.18) at high SNR. In fact, in this case, we have

$$\psi_k \approxeq \hat{\phi}_{k-1}^{\text{rvb}}, \tag{3.19}$$

which brings to

$$\sin[q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})] = q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}}) \qquad \text{and} \qquad \cos[q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})] = 1. \tag{3.20}$$

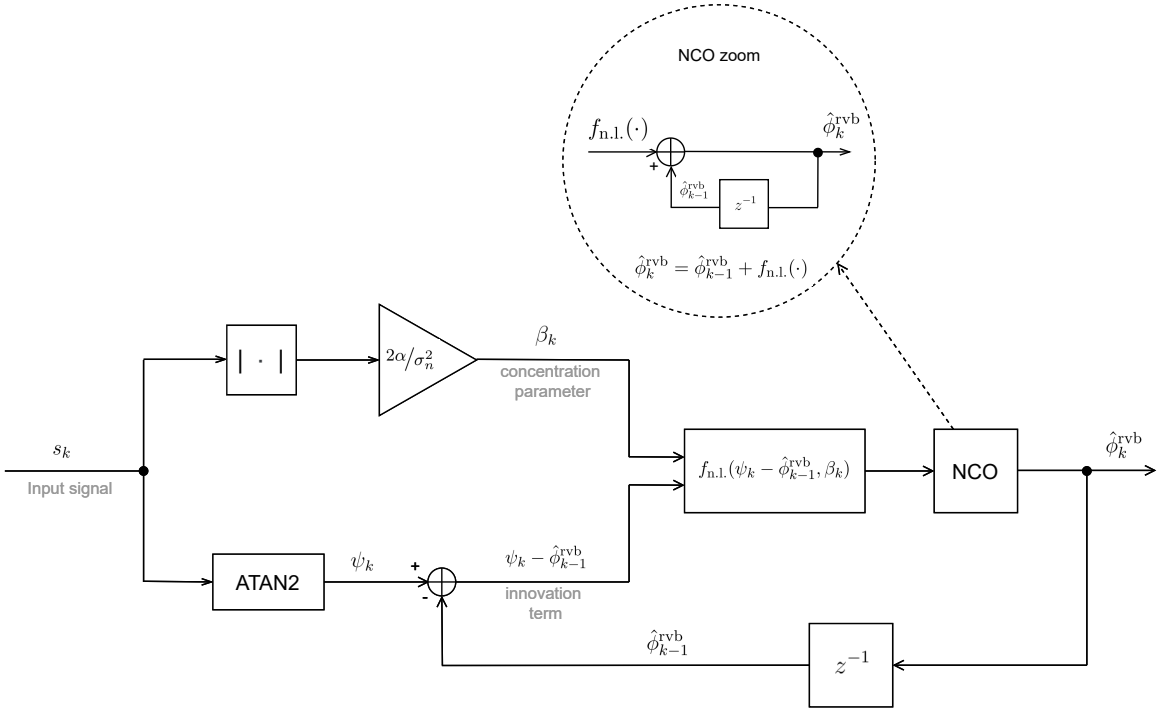


Figure 3.3: Block diagram of the RVB estimator

Therefore, we can simplify the RVB estimator (3.18) such that

$$\hat{\phi}_k^{\text{rvb}} = \hat{\phi}_{k-1}^{\text{rvb}} + K_k^{\text{rvb}}(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}}), \tag{3.21}$$

where K_k^{rvb} is the nonlinear RVB gain, defined as

$$K_k^{\text{rvb}} = 2\sigma_\phi^2 \frac{\sum_{q=1}^{+\infty} q^2 I_q(\beta_k) e^{-\frac{q^2\sigma_\phi^2}{2}}}{I_0(\beta_k) + 2\sum_{q=1}^{+\infty} I_q(\beta_k) e^{-\frac{q^2\sigma_\phi^2}{2}}}. \tag{3.22}$$

3.3 RVB performance

In this section, the performance of the RVB estimator (3.17)-(3.18) is assessed via synthetic data. Before comparing it with traditional carrier-phase tracking techniques such as the DPLL and KF-based DPLL, the RVB estimator is firstly performed independently to gain some insight into its behavior.

Remark
To avoid numerical issues that may arise at high C/N_0 (where β_k has large values), a scaled version of the Bessel function in (3.17)-(3.18) *is implemented (i.e., $I_q(\beta_k)exp(-\beta_k)$) [82].*

3.3.1 Single phase track using a noise-free observation

To better understand the role played by the process noise power σ_ϕ, the RVB estimator is tested with noise-free signal (*i.e.*, $s_k \overset{\text{def}}{=} \alpha e^{j\phi_k}$). The phase dynamics considered are of polynomial form up to the second-order, as defined in (2.19). Furthermore, a truncation value $q_{\max} = 50$ is chosen for the reasons of: 1) convenient balance between computational cost/precision of estimation; 2) the Bessel functions do not strongly impact the recursion process anymore when q is increased. RVB responses are presented in Fig. 3.4. As could be reasonably expected, the proposed first-order RVB estimator (3.18) tracks with no bias a step input, with a finite bias a ramp input (with slow slope) and with an infinite bias an acceleration. Additionally, the parameter σ_ϕ strongly influences the response of the estimator in terms of acquisition time and bias. It thus plays a similar role as that of the loop bandwidth in DPLL though performance depends nonlinearly on σ_ϕ. On this point, it is worth to analyze

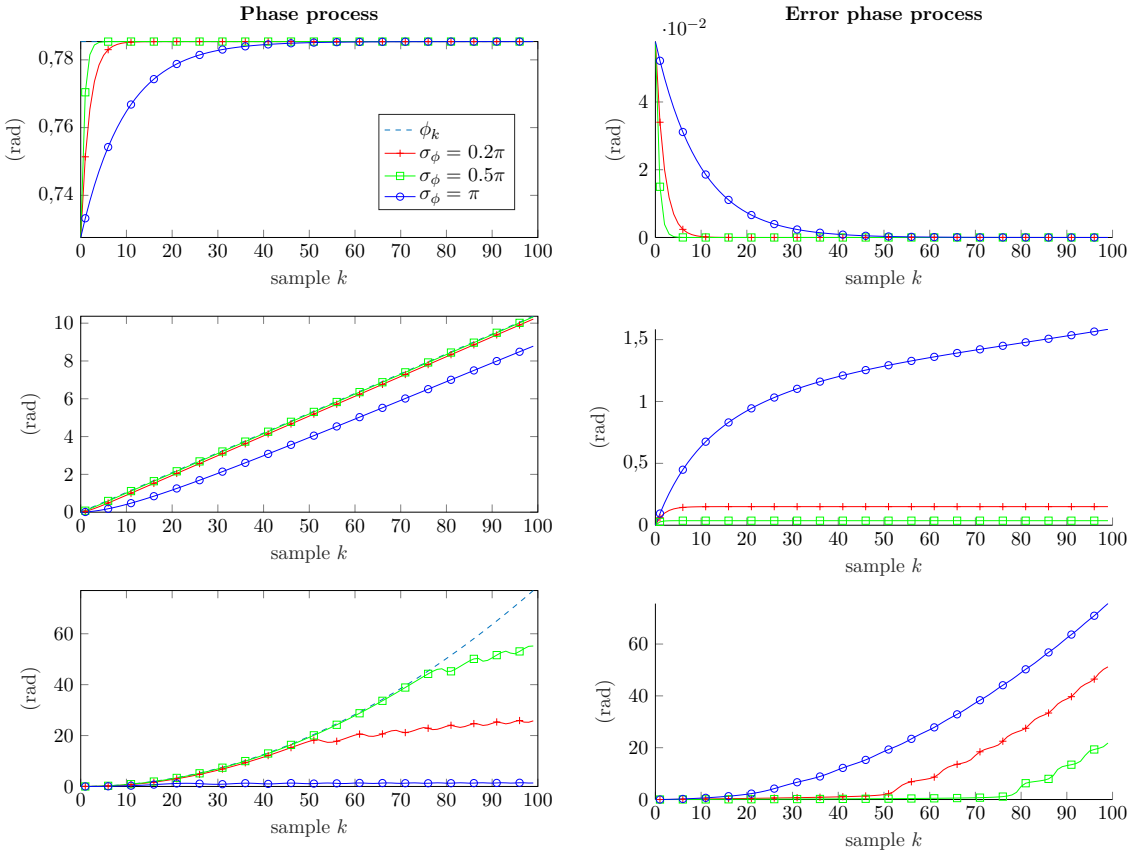


Figure 3.4: Phase responses of the RVB estimator to a step- (up), ramp- (middle), and acceleration- (bottom) input. ($T = 0.02$ s; $\sigma_n^2 = 1$; $\phi_0 = \pi/4$ rad; $\dot{\phi}_0 = \pi/30$ rad/sample; $\ddot{\phi}_0 = \pi/(20^2)$ rad/sample^2)

the impact of noise process σ_ϕ on the innovation term $\psi_k - \hat{\phi}_{k-1}^{\text{rvb}}$. In particular, assuming to work under a linear regime (high SNR), the RVB gain K_k^{rvb} in (3.22) at steady-state is varied wrt σ_ϕ using a noise-free signal as depicted in Fig. 3.5. We see that the impact of the noise

process σ_ϕ on the recursive RVB expression (3.21) decreases when it increases.

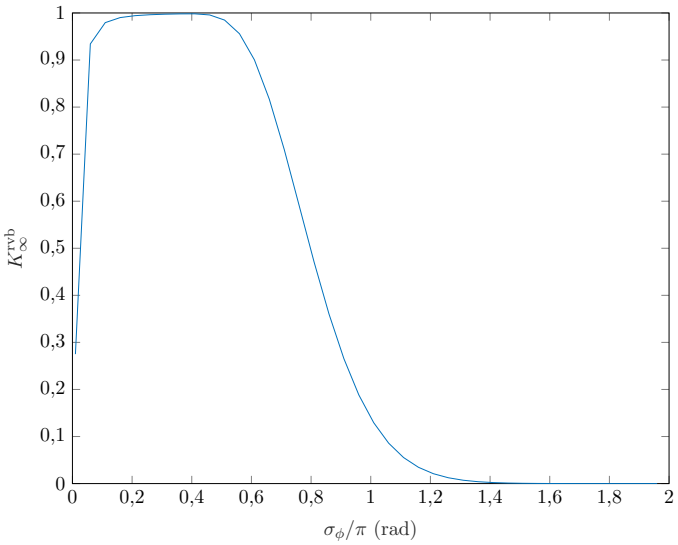


Figure 3.5: RVB gain using a noise-free signal. ($\phi_0 = \pi/4$ rad; $T = 0.02$ s; $C/N_0 = 40$ dB-Hz)

3.3.2 Comparison with DPLL and KF-based DPLL

In this section, the RVB estimator is compared with traditional carrier-phase tracking techniques, namely the DPLL and KF-based DPLL. Since the RVB algorithm is designed in this chapter at first-order only, the comparison is evaluated for slow phase dynamics, such as step- and ramp-phase inputs. The performance metrics considered can be regrouped in two distinct families: 1) precision of estimation metrics where the cycle slip phenomenon is neglected since we consider the phase error modulo-2π, namely $\tilde{e}_k$, defined in (2.33); 2) cycle slipping metrics, described in section 2.1.4.3.

3.3.2.1 Precision of estimation metrics

To assess the precision of estimation, the algorithm is analyzed when cycle slips are discarded. More specifically, the following performance metrics are considered.

1. The *Root Mean Square Error* (RMSE) modulo-2π (denoted as RMSE-mod) is evaluated. The latter is expressed as follows

$$\text{RMSE-mod} = \sqrt{\mathcal{E}\{\tilde{e}_k^2\}} \overset{\text{def}}{=} \sqrt{\frac{1}{M_c}\sum_{n=1}^{M_c}\tilde{e}_k^2(n)}, \tag{3.23}$$

where M_c is the number of Monte Carlo realizations. An example of RMSE-mod in case of a phase-step input is shown in Fig. 3.6. The metrics is evaluated for the DPLL, KF-

based DPLL and RVB for different loop bandwidths and noise processes, respectively. As can be noted, a convergent value of the RMSE-mod is achieved after a certain time. Furthermore, for the two former algorithms, we note that the RMSE-mod increases while increasing the loop bandwidth and noise process, respectively. Contrarily, for the latter, we do not see the same since the RVB algorithm depends nonlinearly on the noise process σ_ϕ. More details about that are explained in the next section.

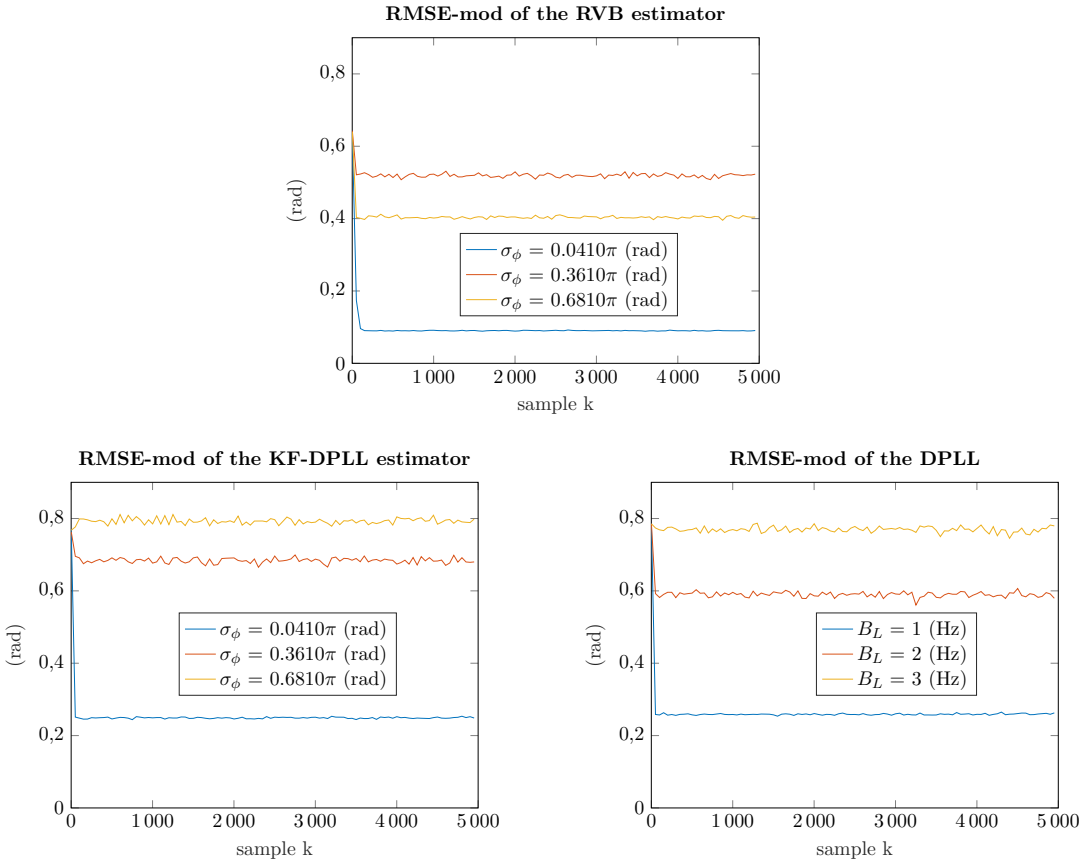


Figure 3.6: RMSE-mod for the DPLL, KF-based DPLL and RVB. ($\phi_0 = \pi/4$ rad; $T = 0.02$ s; $C/N_0 = 17$ dB-Hz; $M_c = 10000$ runs; RVB: $q_{\max} = 50$; DPLL: $\phi_0 = 0$ rad; Filter coefficient calculated using (2.28); discriminator type: ATAN2; KF-based DPLL: $\phi_{0|0} = 0$ rad; $P_{0|0} = \pi^2/3$ rad^2; discriminator type: ATAN2)

2. As soon as the RMSE-mod attains a constant value wrt time, the *acquisition time* is recorded.

Note that the noise process standard deviations, namely σ_ϕ and $\sqrt{Q}$, and the product loop bandwidth-integration time, namely $B_L T$, have been set $[0.001, 0.89] \times \pi$ and $[0.001, 0.5]$, respectively. About the former set, in case of RVB algorithm setting, on one hand, the higher the noise process σ_ϕ the higher is the occurrence of the slip (more noise is letting enter in the recursion procedure); on the other hand, the lower the noise process σ_ϕ the lower is its impact on the innovation term and so on the updated state parameter (the influence of

σ_ϕ on the innovation term is also explained in Fig. 3.5 in linear regime). Accordingly, the noise process standard deviation $\sqrt{Q}$ of the KF-based DPLL method is chosen in the same range of values. About the latter set, note that the product $B_L T$ of the DPLL is chosen in the range where a stable-loop at first-order is obtained [69]. The equivalence between the process noise covariance matrix Q of the KF-DPLL and the loop bandwidth B_L is presented in Appendix B. From Fig. B.1, we note that the chosen range of noise process standard deviation $\sqrt{Q}$ corresponds to a typical loop bandwidths values, where B_L goes from 5 Hz to 20 Hz.

Phase-step input

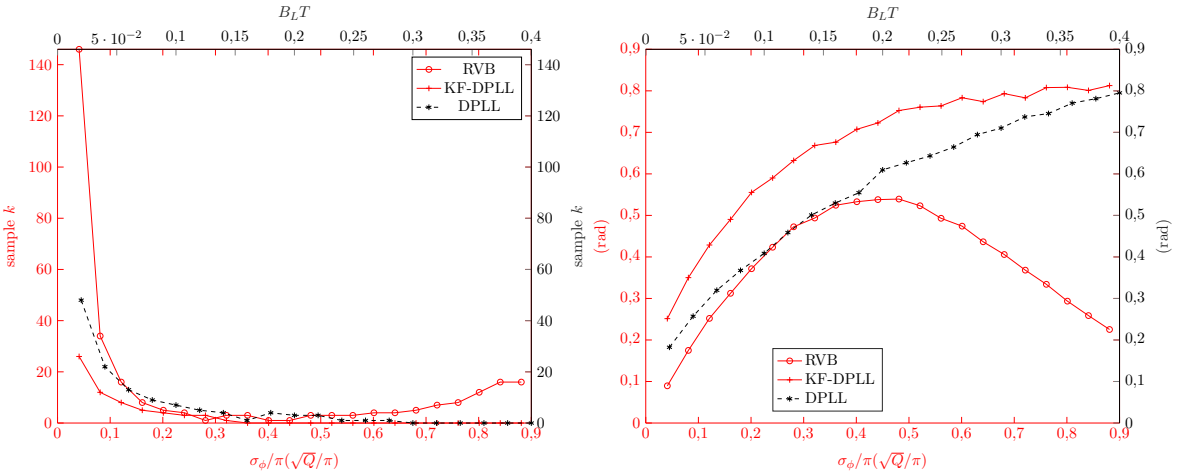
RMSE-mod and acquisition time of the RVB, DPLL and KF-based DPLL are depicted in Fig. 3.7 for different standard deviations σ_ϕ, loop bandwidths B_L and noise covariance matrices Q, respectively. As is well-known, increasing the DPLL loop bandwidth leads to a faster acquisition (in $\approx 1/(B_L T)$ samples) at the expense of a lower estimation precision. A similar behavior can be seen for the KF-based DPLL. For the RVB, the standard deviation σ_ϕ has a similar influence as that of the loop bandwidth and noise covariance matrix Q, but beyond a given value $\sigma_\phi \approx 0.5\pi$ rad the influence is reversed so that the acquisition time increases while reaching a higher precision. This is due to the nonlinear dependence of the innovation term on σ_ϕ^2 in (3.18). As a consequence, the same RMSE-mod can be obtained for two different values of σ_ϕ. Additionally, for any fixed acquisition time, the RMSE-mod of the RVB estimator is always lower than that of DPLL and KF-based DPLL.

Phase-ramp input

As in the previous section, RMSE-mod and acquisition time of the RVB, DPLL and KF-based DPLL are depicted in Fig. 3.8 for different standard deviations σ_ϕ, loop bandwidths B_L and noise covariance matrices Q, respectively. As in the case of phase-step input, higher precision of estimation can be reached with the RVB estimator wrt the DPLL and KF-based DPLL for a given acquisition time.

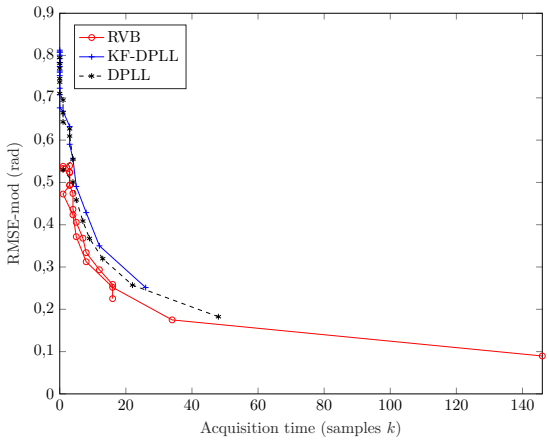
3.3.2.2 Cycle slip metrics

Some cycle slip metrics have been already defined in section 2.1.4.3. In particular, in this section, the cycle slip rate and MTFS are presented using a phase -step and -ramp inputs. Note that due to the nonlinear nature of the RVB algorithm (3.18), defining an analytically steady-state is not straightforward. In the simulations, we say that steady-state is reached when the phase error remains constant while assuming a noise-free signal (*i.e.*, $s_k \stackrel{\text{def}}{=} \alpha e^{j\phi_k}$) and a fixed input phase dynamics. Steady-state is checked and assessed numerically in what follows for phase-step and phase-ramp inputs, respectively. Finally, as announced in section 2.1.4.3, a time limit of simulation, namely T_{Limit}, is set for both metrics to avoid limitless simulations.



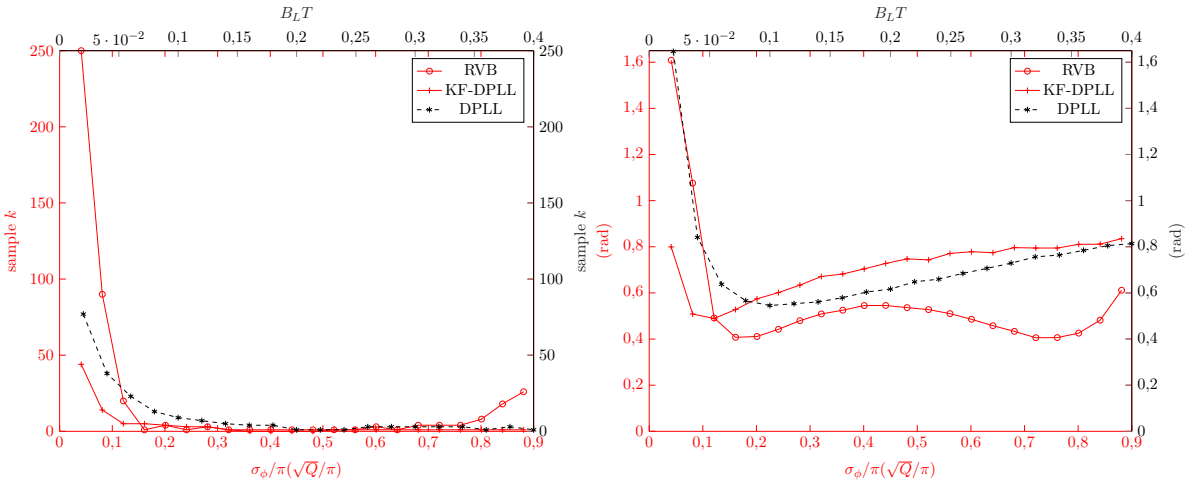
(a) Acquisition time

(b) RMSE-mod



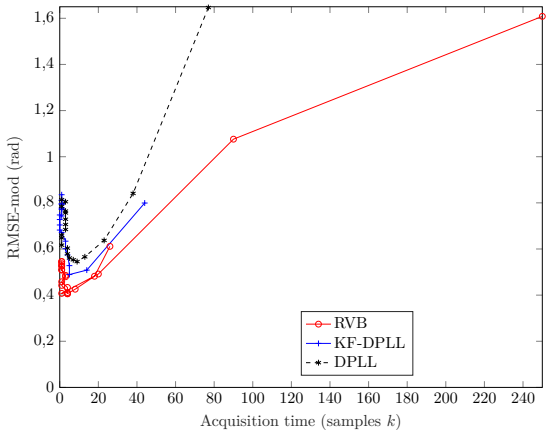
(c) Acquisition time and RMSE-mod

Figure 3.7: Acquisition time and RMSE-mod using a phase-step input for the DPLL, KF-based DPLL and RVB. ($\phi_0 = \pi/4$ rad; $T = 0.02$ s; $C/N_0 = 17$ dB-Hz; $M_c = 10000$ runs; RVB: $q_{\max} = 50$; DPLL: $\phi_0 = 0$ rad; Filter coefficient calculated using (2.28); discriminator type: ATAN2; KF-based DPLL: $\phi_{0|0} = 0$ rad; $P_{0|0} = \pi^2/3$ rad^2; discriminator type: ATAN2; R is found as in (2.56))



(a) Acquisition time

(b) RMSE-mod



(c) Acquisition time and RMSE-mod

Figure 3.8: Acquisition time and RMSE-mod using a phase-ramp input for the DPLL, KF-based DPLL and RVB. ($\dot{\phi}_0 = \pi/30$ rad/sample; $T = 0.02$ s; $C/N_0 = 17$ dB-Hz; $M_c = 10000$ runs; RVB: $q_{\max} = 50$; DPLL: $\phi_0 = 0$ rad; Filter coefficient calculated using (2.28); discriminator type: ATAN2; KF-based DPLL: $\phi_{0|0} = 0$ rad; $P_{0|0} = \pi^2/3$ rad^2; discriminator type: ATAN2; R is found as in (2.56))

For the same reason, we choose the numerical values of T_{Limit} lower for the cycle slip rate metrics than for MTFS. In particular, note that in case of cycle slip rate, a number of slips, namely N_{cs} in (2.42), has to be reached. (In case N_{cs} is not attained before T_{Limit}, the last occurred slip is considered in (2.42)). Contrarily, a single occurred slip is enough for computing the MTFS.

Note that, here too, the range of values for the noise process standard deviations and the product loop bandwidth-integration time are set as described in the previous section, namely 3.3.2.1.

Phase-step input

The cycle slip rate and MTFS using a phase-step input are presented in Figs. 3.9-3.10, respectively. The metrics are depicted as a function of standard deviation σ_ϕ, loop bandwidth B_L and noise covariance matrix Q for RVB, DPLL and KF-based DPLL, respectively. In the scenarios simulated, a few cycle slips occurred for the RVB algorithm during the simulations contrarily to the DPLL and KF-based DPLL. As expected, for the latter two algorithms, the cycle slip rate increases with the loop bandwidth B_L and noise variance Q, respectively. In addition, for the linear filters the lower the process noise, the longer the average time to observe a first slip. On that regard, a different behavior of the MTFS for the RVB occurs due to its nonlinear dependence of the innovation term on σ_ϕ^2 in (3.18). In conclusion, the RVB algorithm outperforms the traditional carrier-phase tracking techniques considered with both metrics.

Phase-ramp input

The cycle slip rate and MTFS using a phase-ramp input are presented in Figs. 3.11-3.12, respectively. Note that for the RVB algorithm we have depicted a vertical line that indicates the lowest σ_ϕ where steady-state can be reached. Actually, for very small σ_ϕ the innovation term in (3.18) is not significant enough to compensate for the phase rotation taking place during the estimator update, *i.e.*, for $\sigma_\phi \approx \dot{\phi}_0 \times T$ as also observed in [139]. As in the case of phase-step input, the cycle slip rate increases with high loop bandwidth B_L and noise covariance matrix Q for DPLL and KF-based DPLL, respectively. Contrarily, the RVB algorithm can reach extremely low cycle slip occurrence (around zero slip per second for most standard deviations σ_ϕ considered). Furthermore, about the MTFS in Fig. 3.12, it clearly appears that using the RVB algorithm instead of a DPLL or KF-based DPLL can lead to a drastic decrease of slip occurrence particularly when σ_ϕ is chosen adequately, *i.e.*, $\sigma_\phi \approx 0.15\pi$ rad and 0.8π rad. Note that the nonlinearity of σ_ϕ wrt the innovation term in (3.18) brings to decrease the MTFS after $\sigma_\phi \approx 0.8\pi$ rad since the innovation term becomes too small and is not able to compensate for the phase rotation during the estimator update anymore. In any event, note that the RVB technique is much less sensitive to the tuning of σ_ϕ than the DPLL is to that of B_L or Q. Finally, we can state that significant improvements in terms of cycle slipping are achieved by using the RVB algorithm.

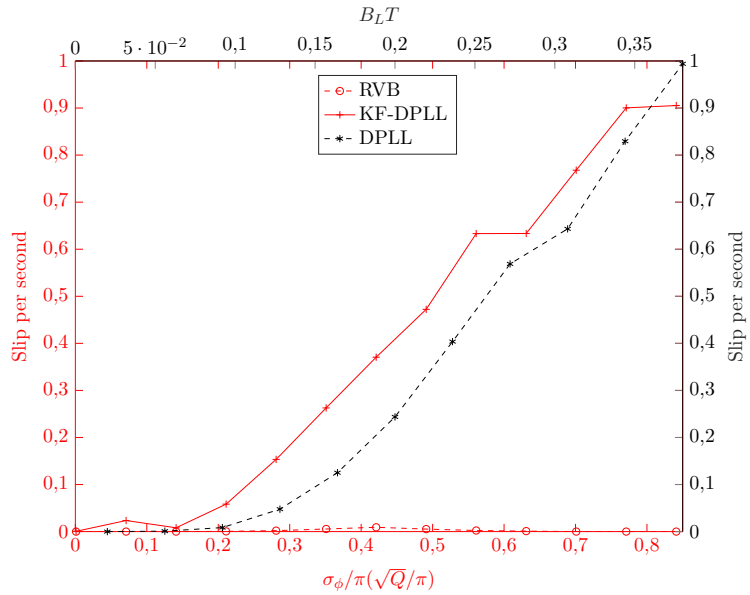


Figure 3.9: Cycle slip rate using a phase-step input for the DPLL, KF-based DPLL and RVB. ($\phi_0 = \pi/4$ rad; $T = 0.02$ s; $C/N_0 = 15$ dB-Hz; $M_c = 500$ runs; $T_{\text{Limit}} = 3200$ s; $N_{cs} = 5$ slips; RVB: $q_{\max} = 50$; DPLL: $\phi_0 = 0$ rad; Filter coefficient calculated using (2.28); discriminator type: ATAN2; KF-based DPLL: $\phi_{0|0} = 0$ rad; $P_{0|0} = \pi^2/3$ rad^2; discriminator type: ATAN2; R is found as in (2.56))

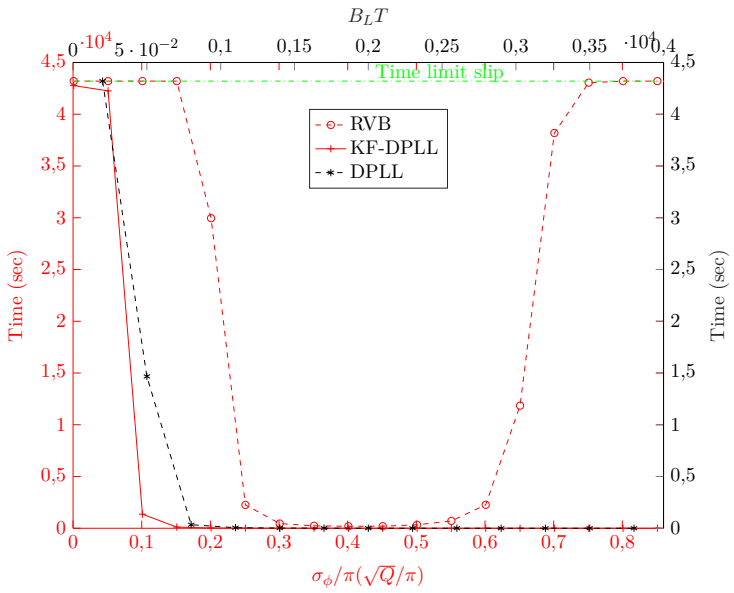


Figure 3.10: MTFS using a phase-step input for the DPLL, KF-based DPLL and RVB. ($\phi_0 = \pi/4$ rad; $T = 0.02$ s; $C/N_0 = 15$ dB-Hz; $M_c = 500$ runs; $T_{\text{Limit}} = 43200$ s; RVB: $q_{\max} = 50$; DPLL: $\phi_0 = 0$ rad; Filter coefficient calculated using (2.28); discriminator type: ATAN2; KF-based DPLL: $\phi_{0|0} = 0$ rad; $P_{0|0} = \pi^2/3$ rad^2; discriminator type: ATAN2; R is found as in (2.56))

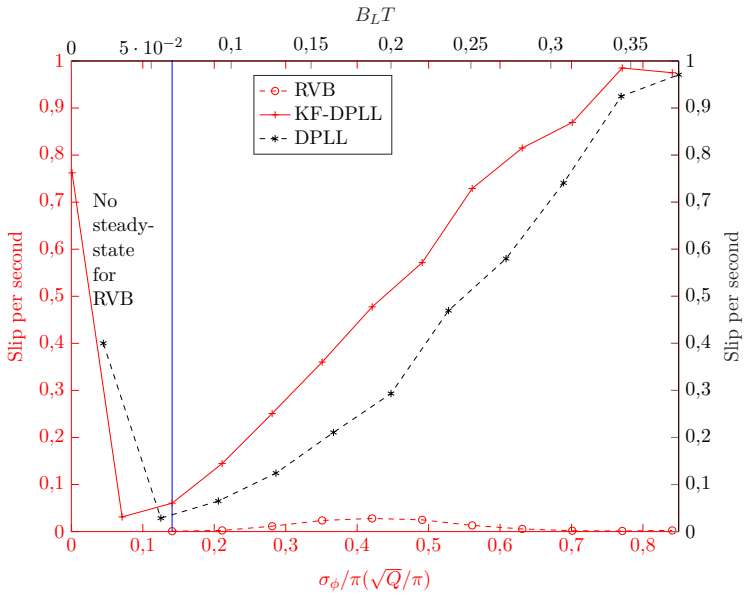


Figure 3.11: Cycle slip rate using a phase-ramp input for the DPLL, KF-based DPLL and RVB. ($\dot{\phi}_0 = \pi/30$ rad/sample; $T = 0.02$ s; $C/N_0 = 15$ dB-Hz; $M_c = 500$ runs; $T_{\text{Limit}} = 3200$ s; $N_{cs} = 5$ slips; RVB: $q_{\max} = 50$; DPLL: $\phi_0 = 0$ rad; Filter coefficient calculated using (2.28); discriminator type: ATAN2; KF-based DPLL: $\phi_{0|0} = 0$ rad; $P_{0|0} = \pi^2/3$ rad^2; discriminator type: ATAN2; R is found as in (2.56))

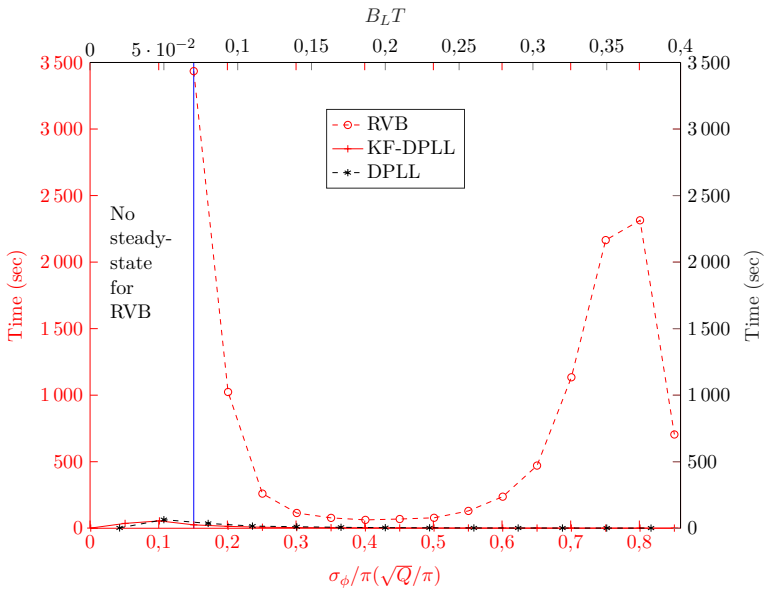


Figure 3.12: MTFS using a phase-ramp input for the DPLL, KF-based DPLL and RVB. ($\dot{\phi}_0 = \pi/30$ rad/sample; $T = 0.02$ s; $C/N_0 = 15$ dB-Hz; $M_c = 500$ runs; $T_{\text{Limit}} = 43200$ s; RVB: $q_{\max} = 50$; DPLL: $\phi_0 = 0$ rad; Filter coefficient calculated using (2.28); discriminator type: ATAN2; KF-based DPLL: $\phi_{0|0} = 0$ rad; $P_{0|0} = \pi^2/3$ rad^2; discriminator type: ATAN2; R is found as in (2.56))

3.4 Summary

In this chapter, we propose a robust algorithm for carrier-phase tracking. The latter is based on Variational Bayesian inference, which allows obtaining a tractable yet suboptimal solution to the Bayesian filtering problem. More specifically, our estimator is based on a so-called Restricted Variational Bayes approximation and gives us a closed-form and easy-to-implement expression of the estimator. The RVB algorithm preserves the nonlinear nature of the measurement equation which is beneficial at a low signal-to-noise ratio. In fact, by numerical simulations with respect to traditional techniques such as DPLL and KF-based DPLL, we have shown the benefits of RVB estimator for both slow phase dynamics considered (*viz.* phase -step and -ramp inputs) in terms of precision of estimation and cycle slipping when signal-to-noise ratio is low. In the next chapter, to address more practical scenarios, the proposed RVB algorithm is extended to higher order phase dynamics.

Chapter 4

RVB algorithm in case of high-order dynamics

In this chapter, the RVB algorithm in case of high-order dynamics is presented. The state vector to be estimated is formed by the carrier-phase and its derivatives. The approach starts with the introduction of the state-space model, then follows the use of the VB inference, and finally, ends with the calculation of the closed-form solution of RVB estimator. Afterwards, the latter is compared with conventional techniques such as DPLL and KF-based DPLL using high-order phase dynamics. As in case of first-order RVB algorithm, the comparison is made considering the precision of estimation and cycle slipping metrics. Results show that the proposed method outperforms the conventional linear filters with respect to all the considered metrics, when signal-to-noise ratio is low.

Contents

4.1 State-space model

In the previous chapter, we have considered as SSM where the dynamics equation was expressed at the first-order. In that case, only the carrier-phase was directly estimated by the proposed RVB algorithm (3.18). However, for realistic GNSS applications, it is necessary to

consider higher order dynamics. To that end, the state vector under consideration will entail not only the carrier phase information but also its derivatives up to a predefined order. The corresponding SSM is formulated as follows.

4.1.1 Measurement equation

By now we know that the measurement equation is described by the input signal which has been already defined in section 2.1.1.1. All the hypotheses about the input signal and the receiver scenario are kept unchanged which can be summarized in: 1) absence of navigation message and noise assumed to be AWGN; 2) about the scenario: input signal at the output of the Prompt correlator (therefore, code sequence is absent), and finally, perfect synchronization of either delay or both delay and frequency. To account for a high-order dynamics, we rewrite the signal by using the updated notation of the state, namely

$$s_k = \alpha e^{j[\boldsymbol{x}_k]_1} + n_k, \tag{4.1}$$

where $\boldsymbol{x}_k$ is the state vector. Note that the measurement (4.1) only involves the first element of the n_x-length state vector $\boldsymbol{x}_k$ (4.4), namely $[\boldsymbol{x}_k]_1$ [1] (or ϕ_k), which is the phase to be tracked by our estimator. As in (2.1), the noise component n_k is supposed to be complex white and Gaussian with known power σ_n^2, namely $n_k \sim \mathcal{CN}(0, \sigma_n^2)$ at instant k. As proceeded in section 3.1.1, the likelihood function remains unchanged up to the notation involving $\boldsymbol{x}_k$, namely

$$f(s_k|\boldsymbol{x}_k) = f(s_k|[\boldsymbol{x}_k]_1) = \frac{1}{\pi\sigma_n^2}\exp\Big\{ -\frac{1}{\sigma_n^2}\Big[|s_k|^2 + \alpha^2 - 2\alpha|s_k|\cos([\boldsymbol{x}_k]_1 - \psi_k)\Big]\Big\}, \tag{4.2}$$

with $\angle s_k \overset{\text{def}}{=} \psi_k = \text{atan2}(\Im\{s_k\}, \Re\{s_k\})$ the angle that lies between $[-\pi, \pi]$. As proceeded for the first-order RVB algorithm, we assume known the amplitude α and the noise power σ_n^2. For conciseness reason, they are omitted in the conditional terms. Accordingly, the sensor factor associated with (4.2) is

$$s([\boldsymbol{x}_k]_1) \propto f(s_k|\boldsymbol{x}_k) \propto \exp\Big\{\beta_k \cos([\boldsymbol{x}_k]_1 - \psi_k)\Big\}, \tag{4.3}$$

where $\beta_k = {}^{2\alpha|s_k|}/_{\sigma_n^2}$. As in (3.3), we recognize in (4.3) a Tikhonov distribution.

[1] Note that $[\boldsymbol{x}_k]_1 \overset{\text{def}}{=} x_{1,k}$.

4.1.2 Dynamics equation

The time-varying evolution of the assumed high-order phase dynamics is described through the so-called transition equation

$$\boldsymbol{x}_k = \begin{bmatrix} \phi_k \\ \dot{\phi}_k \\ \vdots \\ \overset{n}{\dot{\phi}}_k \end{bmatrix} = \boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{w}_{k-1}, \tag{4.4}$$

where $\overset{n}{\dot{\phi}}_k$ is the n-th derivative of the phase discretized at the time instant kT and $\boldsymbol{w_k}$ represents a centered Gaussian noise with a covariance matrix $\boldsymbol{Q}$, *viz.* $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q})$. The so-called state-transition matrix $\boldsymbol{A}$ is with appropriate dimension. The transitional density is thus

$$f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = (2\pi)^{(-n_x/2)}|\boldsymbol{C}|^{1/2}\exp(-1/2((\boldsymbol{x}_k - \boldsymbol{A}\boldsymbol{x}_{k-1})^\intercal \boldsymbol{C}(\boldsymbol{x}_k - \boldsymbol{A}\boldsymbol{x}_{k-1})), \tag{4.5}$$

where $\boldsymbol{C}$ is the precision matrix of $\boldsymbol{x}_k|\boldsymbol{x}_{k-1}$ given by

$$\boldsymbol{C} = \boldsymbol{Q}^{-1}. \tag{4.6}$$

4.2 RVB tracking algorithm

Herein, we recall the principle of the RVB approximation and apply it to high-order phase dynamics. As already mentioned, the increase of the tracking algorithm's order is necessary for realistic GNSS applications. Furthermore, interesting to note is that although the SSM presented in section 4.1 has a higher complexity wrt to that at first-order as described in chapter 3, the recursive closed-form expression of RVB algorithm remains similar, characterized by an intrinsic nonlinear behavior and a sort of similarity with the KF solution in (2.50c). The methodology was originally described in section 3.2 for first-order phase dynamics. Coherently, the section is described as follows: initially, the optimal Bayesian filtering problem of our SSM is presented; then, the VB inference is used to obtain a tractable solution; finally, the RVB approximation is applied, giving us a closed-form expression of the estimator for high-order dynamics [2].

4.2.1 Optimal filtering problem

As already explained in section 2.2.2, the optimal Bayes filtering that iteratively evaluates the filtering distribution $f(\boldsymbol{x}_k|\boldsymbol{S}_k)$ is obtained by alternating between (4.7) and (4.8) as fol-

lows [115]

$$f(\boldsymbol{x}_k|\boldsymbol{S}_{k-1}) = f(\boldsymbol{x}_1), \qquad k = 1 \tag{4.7a}$$

$$f(\boldsymbol{x}_k|\boldsymbol{S}_{k-1}) = \int f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) f(\boldsymbol{x}_{k-1}|\boldsymbol{S}_{k-1}) d\boldsymbol{x}_{k-1}, \qquad k > 1 \tag{4.7b}$$

and

$$f(\boldsymbol{x}_k|\boldsymbol{S}_k) \propto f(\boldsymbol{s}_k|\boldsymbol{x}_k) f(\boldsymbol{x}_k|\boldsymbol{S}_{k-1}), \qquad k > 1 \tag{4.8}$$

where $f(\boldsymbol{x}_1)$ is the prior distribution at $k = 1$. However, given the likehood function (4.2) and the transitional density (4.5), the recursive propagation (4.8) cannot be analytically determined. Though, using the VB approximation [120], we can consider a fixed functional form for the filtering distribution, *viz* $f(\boldsymbol{x}_k|\boldsymbol{S}_k) \sim \tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_k)$. In our case, we can show as presented after that it allows us to obtain a tractable Bayes filter.

4.2.2 Local VB filtering

The RVB-based method relies on a twofold approximation of the filtering problem. Following [120], in the first stage a local Variational Bayes approximation is made, in particular a conditional independence between $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k-1}$ is assumed

$$\tilde{f}(\boldsymbol{x}_k, \boldsymbol{x}_{k-1}|\boldsymbol{S}_k) = \tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_k) \tilde{f}(\boldsymbol{x}_{k-1}|\boldsymbol{S}_k), \tag{4.9}$$

where $\tilde{f}(\cdot)$ refers to the approximated posterior distribution. The latter is then chosen to minimize the KL divergence [120]. This approximation leads to recursively evaluating the posterior distribution $f(\boldsymbol{x}_k|\boldsymbol{S}_k)$ as follows

- Prediction and data update for $k = 1$

$$\tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_{k-1}) \propto \exp\Big(\mathrm{E}_{\tilde{f}(\boldsymbol{x}_{k-1}|\boldsymbol{S}_k)}\Big[\ln(f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}))\Big]\Big), \tag{4.10}$$

  with

$$\tilde{f}(\boldsymbol{x}_{k-1}|\boldsymbol{S}_k) \propto \exp\Big(\mathrm{E}_{\tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_k)}\Big[\ln(f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}))\Big]\Big) \times \tilde{f}(\boldsymbol{x}_{k-1}|\boldsymbol{S}_{k-1}).$$

- Prediction and data update for $k > 1$

$$\tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_k) \propto f(\boldsymbol{s}_k|\boldsymbol{x}_k) \tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_{k-1}). \tag{4.11}$$

We apply a second approximation that, as shown in what follows, will lead to a closed-form filter. In particular, through *RVB approximation* the distribution $\tilde{f}(\boldsymbol{x}_{k-1}|\boldsymbol{S}_k)$ in (4.10) is replaced by the fixed posterior distribution $\tilde{f}(\boldsymbol{x}_{k-1}|\boldsymbol{S}_{k-1})$. The prediction distribution becomes then

$$\tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_{k-1}) \propto \exp\Big(\mathrm{E}_{\tilde{f}(\boldsymbol{x}_{k-1}|\boldsymbol{S}_{k-1})}\Big[\ln(f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}))\Big]\Big). \tag{4.12}$$

Using the sensor factor (4.3) and the transitional density (4.5), we can show that RVB filtering has the following closed-form expressions

- Prediction and data update for $k = 1$

$$\tilde{f}(\boldsymbol{x}_1|\boldsymbol{S}_0) \overset{\text{def}}{=} f(\boldsymbol{x}_1), \tag{4.13a}$$
$$\tilde{f}(\boldsymbol{x}_1|\boldsymbol{S}_1) \propto \mathrm{T}([\boldsymbol{x}_1]_1|\psi_1, \beta_1) \times f(\boldsymbol{x}_1). \tag{4.13b}$$

- Prediction and data update for $k > 1$

$$\tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_{k-1}) = \mathrm{N}\Big(\boldsymbol{x}_k|\boldsymbol{A}\mathrm{E}_{\tilde{f}(\boldsymbol{x}_{k-1}|\boldsymbol{S}_{k-1})}[\boldsymbol{x}_{k-1}], \boldsymbol{C}^{-1}\Big), \tag{4.14a}$$
$$\tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_k) \propto \mathrm{T}([\boldsymbol{x}_1]_1|\psi_k, \beta_k) \times \tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_{k-1}), \tag{4.14b}$$

where the notation $\mathrm{T}(x|\psi, \beta)$ indicates the Tikhonov PDF and $\mathrm{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Omega})$ represents the multivariate Gaussian PDF [81].

4.2.3 RVB algorithm in case of high-order dynamics

The functional form of the filtering distribution is stable through the iteration (4.13)-(4.14) and is tractable if one can derive the MMSE estimator wrt the approximated RVB distribution (4.14b) that we denote as

$$\hat{\boldsymbol{x}}_k^{\text{rvb}} \overset{\text{def}}{=} \mathrm{E}_{\tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_k)}[\boldsymbol{x}_k] = \int \boldsymbol{x}_k \tilde{f}(\boldsymbol{x}_k|\boldsymbol{S}_k) d\boldsymbol{x}_k. \tag{4.15}$$

Using the former notation, the recursion formula is given by

$$\hat{\boldsymbol{x}}_1^{\text{rvb}} = \frac{\int \boldsymbol{x}_1 \mathrm{T}([\boldsymbol{x}_1]_1|\psi_1, \beta_1) f(\boldsymbol{x}_1) d\boldsymbol{x}_1}{\int \mathrm{T}([\boldsymbol{x}_1]_1|\psi_1, \beta_1) f(\boldsymbol{x}_1) d\boldsymbol{x}_1}, \tag{4.16a}$$
$$\hat{\boldsymbol{x}}_k^{\text{rvb}} = \frac{\int \boldsymbol{x}_k \mathrm{T}([\boldsymbol{x}_k]_1|\psi_k, \beta_k) \mathrm{N}(\boldsymbol{x}_k|\boldsymbol{A}\boldsymbol{x}_{k-1}, \boldsymbol{C}^{-1}) d\boldsymbol{x}_k}{\int \mathrm{T}([\boldsymbol{x}_k]_1|\psi_k, \beta_k) \mathrm{N}(\boldsymbol{x}_k|\boldsymbol{A}\boldsymbol{x}_{k-1}, \boldsymbol{C}^{-1}) d\boldsymbol{x}_k}. \tag{4.16b}$$

Hereafter, the initialization and recursion steps of the RVB at high-order dynamics are detailed [2]. Proof of the results is described in Appendix C.

1. *Initialization*: We propose a factorized form to describe the initial condition: $f(\boldsymbol{x}_1) = f(x_{1,1}) f(\boldsymbol{x}_{1_{-1}})$ where the notation $\boldsymbol{x}_{-1}$ designates the (n_x-1)-length vector $\boldsymbol{x}$ to which the first element has been removed. In that case, using (4.15) with (4.13b) the RVB estimator at $k = 1$ is

$$\hat{\boldsymbol{x}}_1^{\text{rvb}} \overset{\text{def}}{=} \begin{bmatrix} \hat{x}_{1,1}^{\text{rvb}} \\ \\ \hat{\boldsymbol{x}}_{1_{-1}}^{\text{rvb}} \end{bmatrix} \tag{4.17}$$

where

$$\hat{x}_{1,1}^{\text{rvb}} = \frac{\int x_{1,1}\mathrm{T}(x_{1,1}|\psi_1,\beta_1)f(x_{1,1})dx_{1,1}}{\int \mathrm{T}(x_{1,1}|\psi_1,\beta_1)f(x_{1,1})dx_{1,1}} \tag{4.18a}$$

$$\hat{\boldsymbol{x}}_{1_{-1}}^{\text{rvb}} = \int \boldsymbol{x}_{1_{-1}} f(\boldsymbol{x}_{1_{-1}})d\boldsymbol{x}_{1_{-1}}. \tag{4.18b}$$

Without any knowledge of the initial phase and its derivatives, we can choose as an example a uniform distribution for the prior of $\boldsymbol{x}_1$, *i.e.*

$$f(x_{1,1}) \overset{\text{def}}{=} f(\phi_1) \propto \mathbb{I}_{[-\pi,\pi]}(\phi_1) \tag{4.19a}$$

$$f(\boldsymbol{x}_{-1}) \propto \mathbb{I}_{\mathcal{I}}(\boldsymbol{x}_{-1}) \tag{4.19b}$$

where $\mathcal{I}$ denotes a set of symmetrical intervals. Using (4.19a) and (4.18a), the expression of $\hat{x}_{1,1}^{\text{rvb}}$ (or ϕ_1) turns out to be the same as in (3.17). Using (4.19b) and (4.18b), we obtain then $\hat{\boldsymbol{x}}_{1_{-1}}^{\text{rvb}} = \boldsymbol{0}$.

2. *Recursion*: The recursive equation of the estimator for $k > 1$ is derived substituting the posterior distribution (4.14b) into (4.15). Its expression is [2]

$$\hat{\boldsymbol{x}}_k^{\text{rvb}} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}} + 2P_1 \frac{\sum_{q=1}^{+\infty} qI_q(\beta_k)\sin[q(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}]_1)]e^{-\frac{q^2P_1}{2}}}{I_0(\beta_k) + 2\sum_{q=1}^{+\infty} I_q(\beta_k)\cos[q(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}]_1)]e^{-\frac{q^2P_1}{2}}} \begin{bmatrix} 1 \\ -\boldsymbol{C}_{-1}^{-1}\boldsymbol{c}_{-1} \end{bmatrix} \tag{4.20}$$

with

$$P_1 \overset{\text{def}}{=} [C_{1,1} - \boldsymbol{c}_{-1}^{\intercal}\boldsymbol{C}_{-1}^{-1}\boldsymbol{c}_{-1}]^{-1} \tag{4.21}$$

where

- $[\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}]_1$ is the first element of $\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}$;
- $I_q(\cdot)$'s are still the modified Bessel functions of the first kind at qth order;
- $C_{1,1}$ is the first diagonal element of $\boldsymbol{C}$;
- $\boldsymbol{C}_{-1}$ is the matrix $\boldsymbol{C}$ to which the first column and row have been removed;
- $\boldsymbol{c}_{-1}$ is the first column of $\boldsymbol{C}$ where the first element has been removed.

Using the blockwise inversion [158], we can simplify the RVB algorithm (4.20), knowing that

$$P_1 = Q_{1,1} \qquad \text{and} \qquad -P_1\boldsymbol{C}_{-1}^{-1}\boldsymbol{c}_{-1} = \boldsymbol{q}_{-1}, \tag{4.22}$$

where $\boldsymbol{q}_{-1}$ is the first column of $\boldsymbol{Q}$ where the first element has been removed. Therefore, the simplified RVB algorithm is

$$\hat{\boldsymbol{x}}_k^{\text{rvb}} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}} + 2\frac{\sum_{q=1}^{+\infty} qI_q(\beta_k)\sin[q(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}]_1)]e^{-\frac{q^2Q_{1,1}}{2}}}{I_0(\beta_k) + 2\sum_{q=1}^{+\infty} I_q(\beta_k)\cos[q(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}]_1)]e^{-\frac{q^2Q_{1,1}}{2}}}\boldsymbol{Q}_1, \tag{4.23}$$

where $\boldsymbol{Q}_1$ is the first column of $\boldsymbol{Q}$.

As can be seen, the expression (4.23) maintains the nonlinear nature of the measurement equation (4.1). Furthermore, we can immediately note that the estimation of the state $\hat{\boldsymbol{x}}_k^{\text{rvb}}$ partially depends on the covariance matrix $\boldsymbol{Q}$; we will see that this has an impact on the numerical choice of $\boldsymbol{Q}$. As already done for first-order RVB algorithm, since the Bessel function decreases rapidly wrt the summation index q, a truncation of the infinite sum in (4.23) to $q_{\max}$ is enforced. It is worth noticing that at first-order, namely $n_x = 1$, then the filtering of (3.18) is recovered, as expected.

Finally, we can see a similarity between (4.23) and the traditional KF update state expression (2.50c), as in the case of first-order RVB expression (3.18). As a matter of fact, (4.23) is the sum of the predicted state estimate $\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}$ plus a function of the innovation term $\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}]_1$ that is nonlinear (unlike the Kalman gain function). This similitude can be immediately seen when analyzing the RVB in a linear regime (at high SNR). As done in section 3.2.3, at high SNR, we can write

$$\hat{\boldsymbol{x}}_k^{\text{rvb}} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}} + \boldsymbol{K}_k^{\text{rvb}}(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}]_1), \tag{4.24}$$

where

$$\boldsymbol{K}_k^{\text{rvb}} = 2\frac{\sum_{q=1}^{+\infty} q^2 I_q(\beta_k) e^{-\frac{q^2 Q_{1,1}}{2}}}{I_0(\beta_k) + 2\sum_{q=1}^{+\infty} I_q(\beta_k) e^{-\frac{q^2 Q_{1,1}}{2}}}\boldsymbol{Q}_1, \tag{4.25}$$

which represents the nonlinear RVB gain.

4.3 RVB performance

In this section, the performance of the RVB estimator (4.23) is assessed via synthetic data. Before comparing it with traditional carrier-phase tracking techniques such as the DPLL and KF-based DPLL, the RVB estimator is firstly performed independently to gain some insight into its behavior.

Remark
As in the first-order RVB algorithm (3.18)*, to avoid numerical issues that may arise at high C/N_0 (where β_k has large values), a scaled version of the Bessel function in* (4.23) *is implemented (i.e., $I_q(\beta_k)exp(-\beta_k)$) [82].*

4.3.1 Single phase track using a noise-free observation

4.3.1.1 Scenario

To show the performance of the RVB estimator, a dynamics equation which models the carrier-phase of the received signal (4.1) is needed. We choose the latter to follow a third-

order polynomial evolution as

$$\phi_k = \phi_0 + \dot{\phi}_0 T k + \frac{\ddot{\phi}_0 T^2 k^2}{2} \tag{4.26}$$

where ϕ_0, $\dot{\phi}_0$, $\ddot{\phi}_0$ are respectively the initial phase (step), phase rate (ramp), acceleration (parabola) and T is the tracking update time. Accordingly, due to the motion between the satellite and the receiver, a third-order dynamics model may be relevant and is often considered [106, Chap.9]. In particular, a *Position-Velocity-Acceleration* (PVA) motion model is chosen in discrete-time domain [159, 160, 106]. Therefore, the state matrix $\boldsymbol{A}$ in (4.4) is thus equal to

$$\boldsymbol{A} = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \tag{4.27}$$

A complete form of the process noise covariance matrix $\boldsymbol{Q}$ in (4.4) is chosen, where all the dynamics until the third-order are included. Namely,

$$\boldsymbol{Q} = \Sigma_{\text{pva}} \begin{bmatrix} T^5/20 & T^4/8 & T^3/6 \\ T^4/8 & T^3/3 & T^2/2 \\ T^3/6 & T^2/2 & T \end{bmatrix} + \Sigma_{\text{pv}} \begin{bmatrix} T^3/3 & T^2/2 & 0 \\ T^2/2 & T & 0 \\ 0 & 0 & 0 \end{bmatrix} + \Sigma_{\text{p}} \begin{bmatrix} T & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{4.28}$$

where the notation Σ represents the PSD of the continuous-time white noise associated with the PVA model [160]. In case of small time update T, the expression of $\boldsymbol{Q}$ can be simplified, ignoring all the terms expect for the element in position (3,3) of the matrix [106, Chap.4]. The matrix $\boldsymbol{Q}$ depends then on three terms that we define as follows:

- $\sigma_\phi \triangleq \sqrt{\Sigma_{\text{p}} T}$ (rad): which represents the incertitude assumed a priori on the fluctuation of the phase;
- $\sigma_{\dot{\phi}} \triangleq \sqrt{\Sigma_{\text{pv}} T}$ (rad/s): which represents the incertitude assumed a priori on the fluctuation of first-order phase derivative;
- $\sigma_{\ddot{\phi}} \triangleq \sqrt{\Sigma_{\text{pva}} T}$ (rad/s^2): which represents the incertitude assumed a priori on the fluctuation of second-order phase derivative.

Note that the expression of the noise covariance matrix $\boldsymbol{Q}$ in (4.28) is often used in GNSS framework, where clock errors and Doppler effect are considered while tracking the signal [131, Chap.5]. In the reminder, we will consider the complete form of $\boldsymbol{Q}$ (4.28) in our simulations, and give its numerical values in terms of σ_ϕ, $\sigma_{\dot{\phi}}$ and $\sigma_{\ddot{\phi}}$.

4.3.1.2 Performance using a noise-free observation

To better understand the behavior of the RVB algorithm, a state vector formed by the phase and its derivatives is here estimated by the third-order RVB technique in (4.23). The phase dynamics of the considered noise-free signal is generated as described in (4.26). Results are shown in Fig. 4.1. As can be seen, the estimator is able to track a parabolic carrier-phase input and its derivatives, obtaining an error that goes to zero with time. Note that the standard deviations in (4.28), namely σ_ϕ, $\sigma_{\dot{\phi}}$ and $\sigma_{\ddot{\phi}}$, have been chosen to obtain a sufficiently significant impact on the innovation term $(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}]_1)$ in the recursion process of (4.23) About that, their values have been empirically set acknowledging that the RVB does not depends on the PVA model anymore but only on $\boldsymbol{Q}_1$, and a physical (at least direct) meaning of the standard deviations is therefore lost. On that regard, a future work analysis is presented in the conclusion section of the manuscript. Finally, it is worthy noting that when analyzing the RVB algorithm in a linear regime (4.24)-(4.25) (at high SNR), for certain value of $\boldsymbol{Q}_1$ the innovation term tends towards zero and thus has no impact on the recursive process (4.23). In particular, as happened in case of first-order RVB algorithm (see section 3.3.1), the RVB gain (4.25) differs from zero only for a certain range of PSDs. An example of RVB gain in case of a PVA model is shown in Fig. 4.2. Note that the standard deviation $\sigma_{\ddot{\phi}}$ has been set to display the RVB gain. In particular, two very different values (low and high) of $\sigma_{\ddot{\phi}}$ have been chosen to observe any effects of the latter on the RVB gain. As can be seen, for both $\sigma_{\ddot{\phi}}$ chosen, the gain behaves the same way. More specifically, only a certain combination of the noise processes σ_ϕ and $\sigma_{\dot{\phi}}$ gives a gain $\boldsymbol{K}_\infty^{\text{rvb}}$ value which has effects on the innovation term.

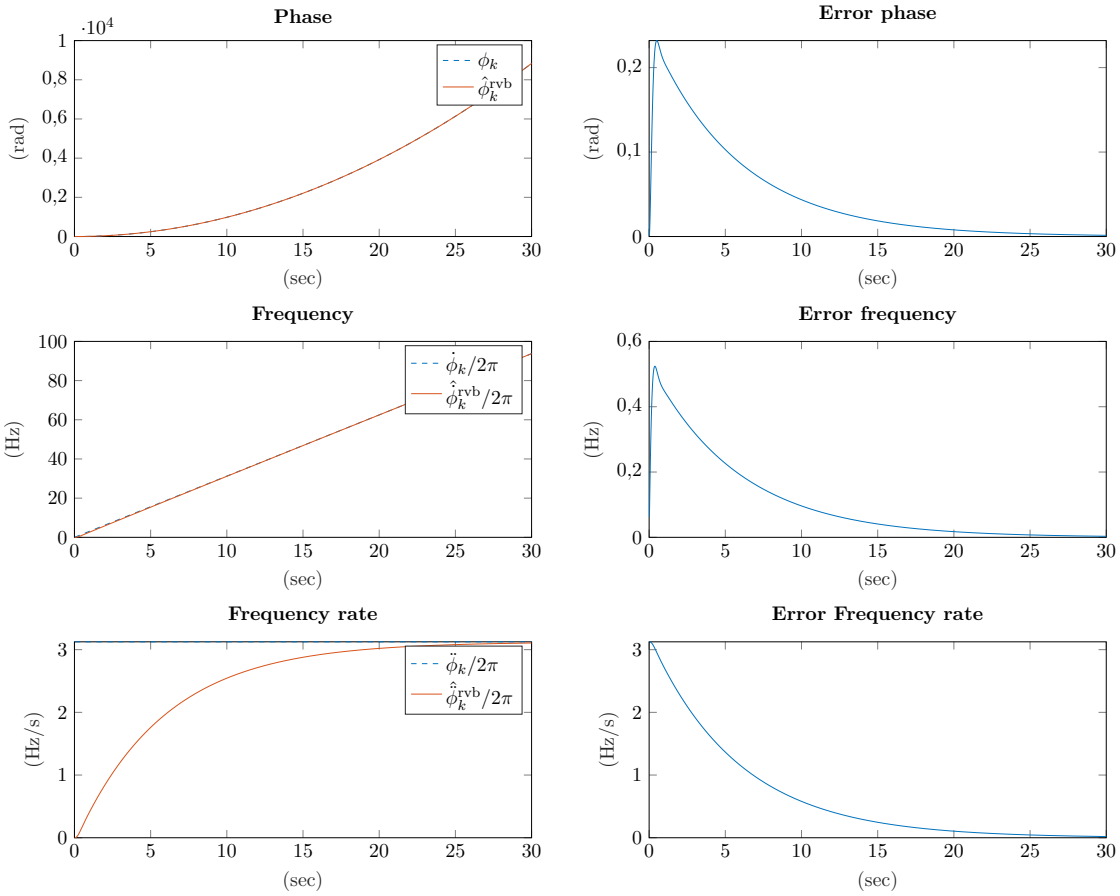
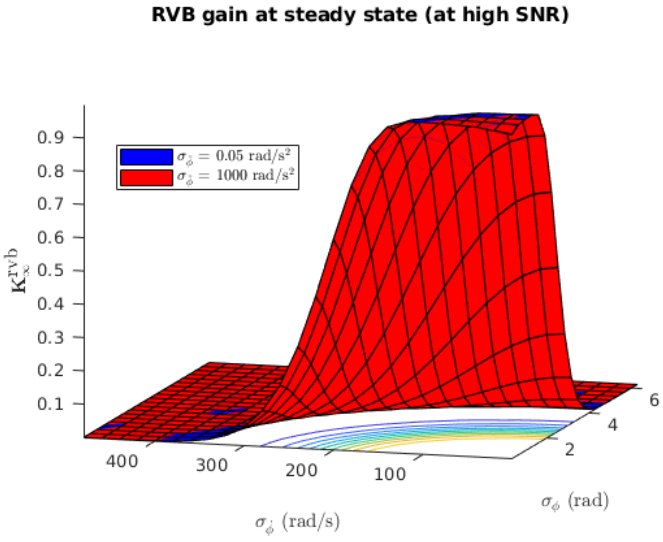
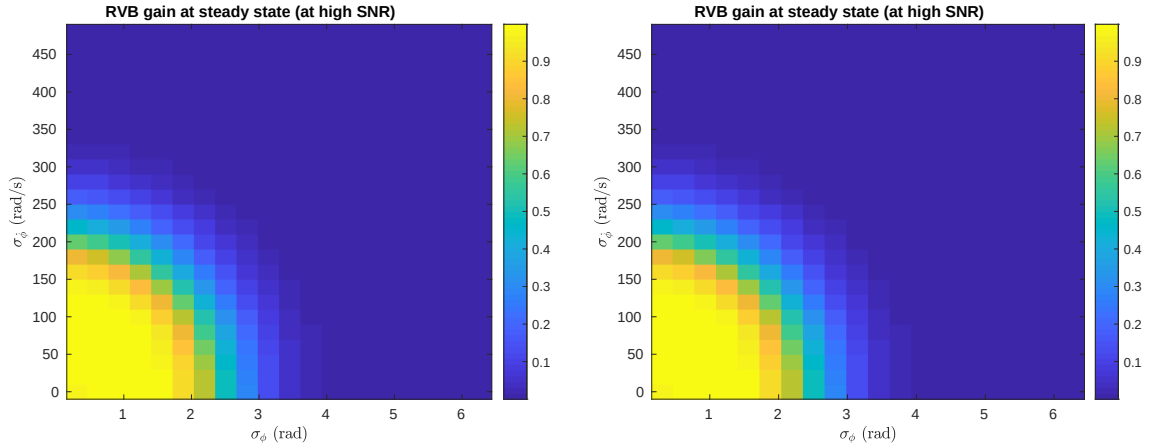


Figure 4.1: State estimation of the RVB algorithm using a noise-free signal and a PVA model to an acceleration input. ($T = 0.02$ s; $\sigma_n^2 = 1$; $C/N_0 = 15$ dB-Hz; $\phi_0 = 0$ rad; $\dot{\phi}_0 = 0$ rad/sample; $\ddot{\phi}_0 = \pi/(20^2)$ rad/sample^2; $q_{\max} = 50$; $\sigma_\phi = 0.8\pi$ rad; $\sigma_{\dot{\phi}} = 20\pi$ rad/s; $\sigma_{\ddot{\phi}} = 100\pi$ rad/s^2)





(a) ($\sigma_{\ddot{\phi}} = 0.05$ rad/s^2) (b) ($\sigma_{\ddot{\phi}} = 1000$ rad/s^2)

Figure 4.2: RVB gain (4.25) in case of PVA model using a noise-free signal. ($T = 0.02$ s; $\sigma_n^2 = 1$; $C/N_0 = 40$ dB-Hz; $q_{\max} = 50$)

4.3.2 Comparison with DPLL and KF-based DPLL

In this section, the RVB estimator is compared with traditional carrier-phase tracking techniques, namely the DPLL and KF-based DPLL. Note that we present the results of our main parameter of interest, that is the estimation of the true phase $[\boldsymbol{x}_k]_1$ that is chosen to be a parabolic-phase input. As done in case of first-order analysis (see section 3.3.2), given the nonlinear nature of RVB algorithm at high-order too (4.23), two metrics are performed to characterize on the one hand the precision of estimation letting alone the slip phenomenon and on the other hand the occurrence of slips. More precisely, for the former the RMSE-mod is evaluated, for the latter the cycle slip rate and MTFS are considered. All the metrics are evaluated wrt the signal-to-noise ratio in (2.3).

4.3.2.1 Processing parameters

The input parameters of the signal and of the three algorithms are presented in Tab. 4.1. Note that tuning the PSDs of the process noise covariance matrices $\boldsymbol{Q}$ (4.28) for the RVB is not straightforward. In particular, the RVB update equation (4.23) is not only nonlinear but also depends only partially on the covariance matrix $\boldsymbol{Q}$. Therefore, as already discussed in section 4.3.1.2, we increase the PSD values in order to obtain a sufficiently significant impact on the innovation term, namely $(\psi_k - [\boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\mathrm{rvb}}]_1)$. Regarding the KF-DPLL, its PSD values are experimentally set so that both KF-DPLL and RVB filters result in a non-informative estimator below approximately the same SNR value (see later Fig. 4.3 when so-called RMSE-mod is approximately equal to $\pi/3$). Finally, all the metrics are evaluated via M_c Monte Carlo simulations; $M_c = 10000$ runs for the evaluation of the RMSE-mod and $M_c = 500$ runs for the cycle slip metrics (number of runs are decreased here to avoid too long simulation times).

1. *RVB initialization*: a value of $q_{\max} = 50$, as done in case of first-order RVB algorithm;
2. *DPLL setting*: an ATAN2 discriminator is used for the DPLL. The loop coefficients of the filter have been calculated by using Tab.V (phase/phase-rate feedback and supercritically damped response) in [69];
3. *KF-based DPLL setting*: to have a fair comparison, an ATAN2 discriminator is used for the KF-based DPLL. Finally, note that since the measurement matrix $\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$, the phase noise power reduces to a scalar, as in (2.56) [131, A-13].

4.3.2.2 Precision of estimation metrics

As can be seen in Fig. 4.3, the RVB estimator outperforms both DPLL and KF-based DPLL. In particular, the RMSE-mod of RVB appears to be lower for most of the SNR values. Below a SNR of ≈ -10 dB, all the phase estimators can be considered as noninformative since their

Table 4.1: Input parameters

Parameter	**Variable**	**Value**
Phase	ϕ_0	0 rad
Phase rate	$\dot{\phi}_0$	0 rad/s
Phase acceleration	$\ddot{\phi}_0$	$\pi/0.16$ rad/s^2
Tracking update time	T	0.02 s
RVB PSDs	$\sqrt{\Sigma_{\mathrm{p}}T}$	0.8π rad
	$\sqrt{\Sigma_{\mathrm{pv}}T}$	20π rad/s
	$\sqrt{\Sigma_{\mathrm{pva}}T}$	100π rad/s^2
KF-DPLL PSDs	$\sqrt{\Sigma_{\mathrm{p}}T}$	0.2π rad
	$\sqrt{\Sigma_{\mathrm{pv}}T}$	0.8π rad/s
	$\sqrt{\Sigma_{\mathrm{pva}}T}$	0.2π rad/s^2
DPLL loop bandwidth	B_L	10 Hz

RMSE-mod is equal to that of a uniform distribution on [-π,π]; more precisely, the DPLL has poor performance already below an SNR of ≈ -4 dB. Above, the precision of estimation of the algorithms increases wrt the SNR. Though, the RVB outperforms here clearly the DPLL and KF-based DPLL particularly at low to medium SNR (*i.e.*, approximately from -6 to 10 dB). Finally, for higher SNRs than 10 dB, the RMSE-mod performance converges to the same low value, as expected.

4.3.2.3 Cycle slip metrics

The cycle slip rate and MTFS are shown in Figs. 4.4-4.5, respectively. We assess it as already described in sections 2.1.4.3 and 3.3.2.2. Both metrics are evaluated via Monte Carlo simulations and presented wrt the SNR. As in case of slow dynamics, a time limit, namely T_{Limit}, is set while assessing both metrics to avoid limitless simulations. The latter has been differently set for both metrics: in particular, T_{Limit} is set to one-hour for the cycle slip rate and to half-a-day for the MTFS. As can be seen, the RVB algorithm outperforms the other techniques in both cycle slip metrics. More precisely, using the RVB estimator no slip occurs until the time limit for almost all the depicted SNRs, contrarily to the DPLL and KF-based DPLL. A similar behavior can be seen while analyzing the MTFS metrics, where the RVB algorithm has no slip occurrence until half-a-day time for most of SNRs evaluated.

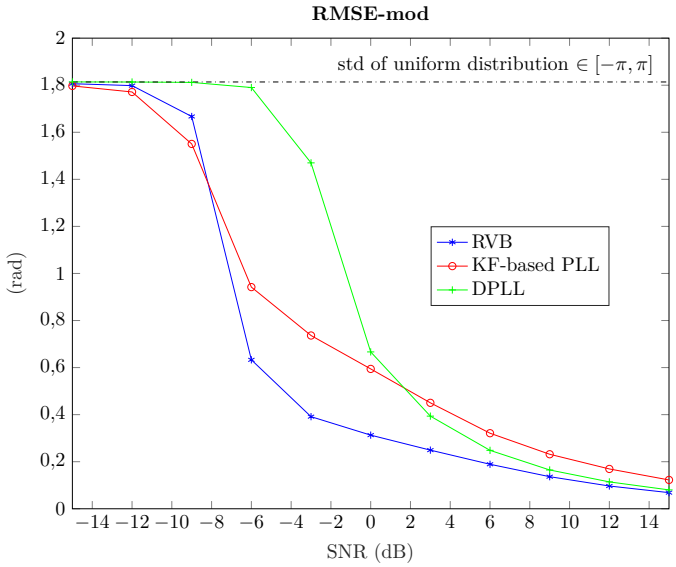


Figure 4.3: RMSE-mod comparison for a third-order loop using a parabolic-phase input

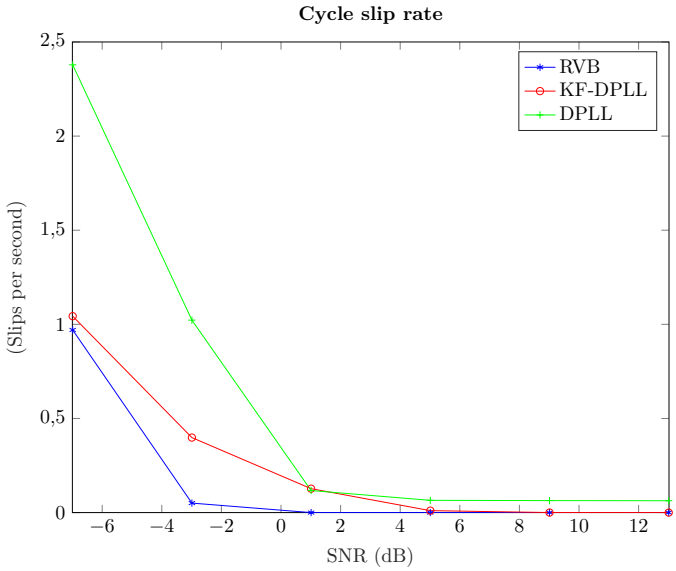


Figure 4.4: Cycle slip comparison for a third-order loop using a parabolic-phase input

4.4 Summary

In this chapter, we presented a robust phase tracking technique towards cycle slipping at any order dynamics. It can be seen as an extension of the algorithm presented in chapter 3, which estimates a first-order phase dynamics. In this case, instead, the state vector is formed by the phase and its derivatives. As proceeded for RVB algorithm at slow dynamics, the method is also based on the RVB approximation, which leads to a nonlinear and tractable filter with closed-form expression. Performance of the associated MMSE estimator is evaluated in terms

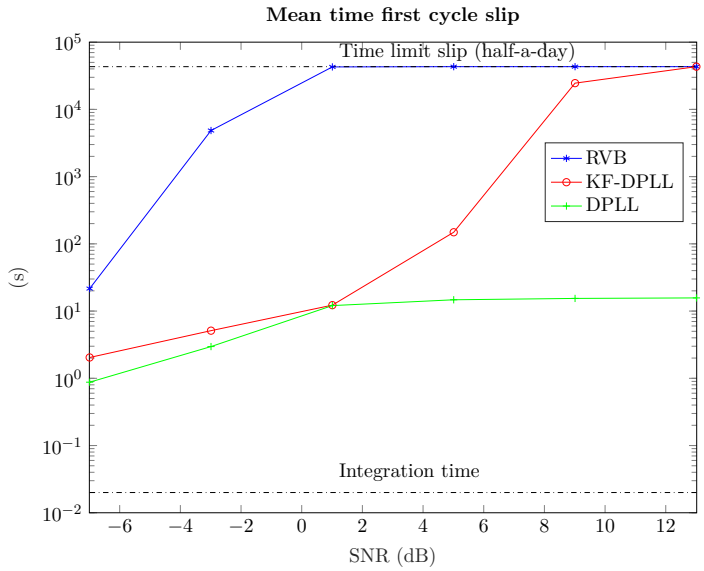


Figure 4.5: MTFS comparison for a third-order loop using a parabolic-phase input

of RMSE-mod, cycle slip rate and MTFS, and compared with a conventional DPLL and KF-based DPLL. The new nonlinear RVB algorithm shows significant performance improvement at a low signal-to-noise ratio for all the metrics considered. In the next chapter, the proposed algorithm is implemented in a more practical scenario, namely in a software-based receiver using real GNSS data.

Chapter 5

RVB algorithm using real GNSS data

In this last chapter, the RVB algorithm with high-order phase dynamics is studied by using real GNSS data. Firstly, the GNSS software-defined receiver previously developed by ISAE-SUPAERO is presented. Then, the real GNSS data characteristics are introduced. From the latter, synthetic (or so-called realistic) carrier-phase trajectory is retrieved from a single-satellite in-view and used for comparison between the RVB algorithm and both DPLL and KF-based DPLL. This step gives us a first idea of the algorithms' behavior when using realistic GNSS carrier-phase. Finally, we focus on the fully experimental GNSS data using the GNSS SDR receiver. Results show significant improvements in terms of robustness to cycle slips when using the RVB algorithm wrt traditional techniques.

Contents

5.1 Software-defined GNSS receiver

In the last decades, a new trend in designing GNSS receivers has emerged that implements digitization closer to the receiver antenna front-end to create a system that works at increasingly higher frequencies and wider bandwidth. This development relies on the *Software Defined Radio* (SDR) approach, originating from signal processing technologies used in military applications. More precisely, in the early 1990s, the U.S. (*United States*) military services

were facing several communications-related challenges such as interception of radio messages by hostile elements. Therefore, taking advantage of the rapid technology changes, a U.S. *Department of Defense* (DoD) multi-phase joint service project named Speakeasy was undertaken with the objective of proving the concept of a programmable waveform, multiband, multimode radio [161]. The Speakeasy project was based on an approach that underlies most software receivers: the ADC is placed as near as possible to the antenna front-end and all baseband functions are processed in a programmable microprocessor using software techniques rather than hardware elements. Today, GNSS SDR receiver has been receiving considerable attention [32, 162]. The major objective of SDR receiver is to efficiently implement the high rate computations while maintaining the desired flexibility inherent in a software-based approach, which is not the case of hardware-based GNSS receivers [163]. Other advantages can be synthesized in the following key points [164]: 1) configurability: within a specific communication technology, this means that a unique transceiver for different applications can be factory-configured without any hardware change; 2) updatability: since the core is software-based, the receiver can be updated and upgraded in case of new and improved algorithms or amendments to a standard, while maintaining the same hardware components.

The GNSS SDR here used has been developed by ISAE-SUPAERO laboratory. The latter is described in the next section.

5.1.1 SDR receiver architecture overview

A simplified block diagram of the GNSS SDR receiver architecture considered is shown in Fig. 5.1, with the following blocks:

1. the RF-FE block which provides samples coming from the antenna to the system;
2. the Data IQ Handler block which distributes the incoming samples to the appropriate processing blocks;
3. the Acquisition block which searches for satellites in-view from the received signal;
4. the Tracking block which follows the acquired satellite signals;
5. the Manager block deals with both acquisition and tracking processes. It also initializes and monitors the tracking loops;
6. the Navigator block that calculates the PVT solution.

The software was developed in Matlab language by using an *Object-Oriented Programming* (OPP) method, ensuring then enough flexibility to allow for a wide range of configurations involving different processors, receiver architecture, acquisition and tracking strategies.

We will focus our attention on the tracking block, implementing our proposed RVB algorithm (4.23) and comparing it with traditional techniques such as DPLL and KF-based DPLL. The detailed architecture of the tracking block is presented in section 5.2.2.1.

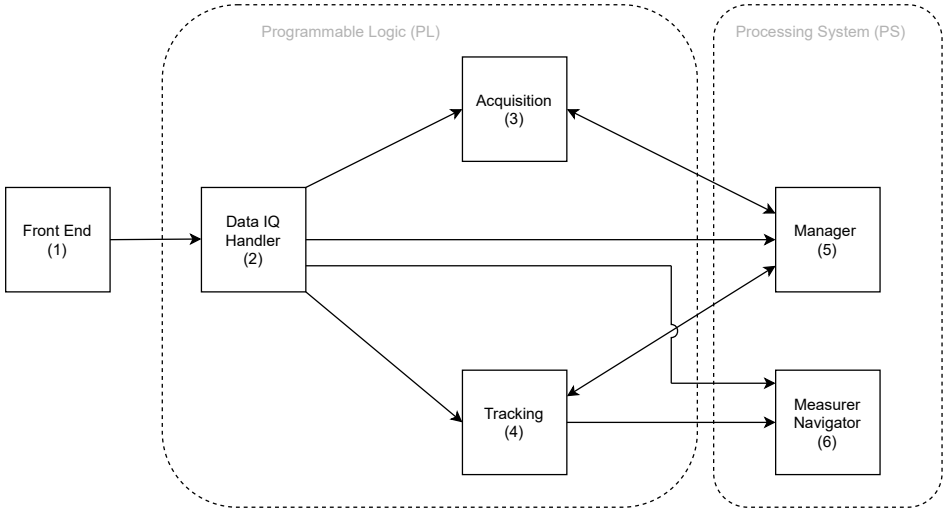


Figure 5.1: Block diagram of the GNSS SDR receiver previously developed at ISAE-SUPAERO [165]

5.1.2 Real data characteristics

The real GNSS data are collected from a static user through the Ettus Research USRP X310 receiver. The samples are recorded at baseband with a sampling frequency of 4 MHz (2 x 4 bits per complex sample). The satellites in-view are summarized in Tab. 5.1. We analyze a single satellite with the highest C/N_0 in Tab. 5.1, therefore satellite number *9* is taken into account. Note that the single satellite is tracked on its L1 C/A channel. In what follows, the real GNSS signal may be degraded by adding a complex AWGN to simulate an harsh environment. More details of that are explained in the next section.

Table 5.1: Real data estimated by the GNSS SDR receiver [165]

PRN	**elev** (deg)	**azim** (deg)	$\boldsymbol{C/N_0}$ (dB-Hz)	**Doppler** (Hz)	**Doppler rate** (Hz/s)
2	32.973	-51.964	45.9	2677	-0.3913
3	19.034	101.61	43.6	-2808	-0.1651
6	65.163	-98.701	50.6	149	-0.6715
7	35.6	163.51	45.6	3105	-0.3201
9	84.594	-9.4387	52.7	-96	-0.4739
23	47.735	54.313	50.2	-1997	-0.2808
30	10.306	-175.83	42.8	3826	0.0001

5.1.2.1 Real data degradation

The degradation is applied to the real GNSS signal by using a complex AWGN. To analyze the performance of the only tracking block, the noise component is added after the acquisition stage. Computationally, the equivalent (eq) incoming signal is

$$s_{\text{eq}}[k] = s_{\text{real}}[k] + w[k], \tag{5.1}$$

where $s_{\text{real}}[k]$ is the baseband real GNSS data (not degraded) from the signal file. $w[k]$ is a complex AWGN with a variance

$$\sigma_w^2 = \sigma_{\text{eq}}^2 - \sigma_n^2, \tag{5.2}$$

where σ_n^2 is the noise variance of the real GNSS signal $s_{\text{real}}[k]$ and

$$\sigma_{\text{eq}}^2 = \frac{|\alpha|^2}{(C/N_{0_{\text{eq}}})T}, \tag{5.3}$$

where α is the estimated amplitude of the real GNSS signal and $C/N_{0_{\text{eq}}}$ is the desired carrier-to-noise ratio.

5.1.3 Synthetic GNSS data based on realistic carrier-phase

To have a first idea of the algorithms' behavior in terms of cycle slip, we firstly compare them by using realistic GNSS carrier-phase trajectories. To obtain the latter, we process real GNSS data with the GNSS SDR receiver and retrieve the estimated phase. As already mentioned, a single-satellite in-view is considered from Tab. 5.1. In particular, we picked the PRN9 for its highest C/N_0 value compared to the others. Note that this satellite is then used for the entire comparison analysis in both synthetic and real GNSS data. The input data of the GNSS SDR receiver are summarized in Tab. 5.2. Furthermore, a third-order DPLL with ATAN2 discriminator is used to estimate the carrier-phase of the PRN9 signal. Results of the phase tracking are shown in Fig. 5.2. Note that the latter is considered as ground truth to generate synthetic GNSS data as in (5.4).

Table 5.2: Input parameters of GNSS SDR receiver

Sampling frequency (Hz)	4e6
Signal duration (s)	0-15
Measurement and navigator period (s)	0.001
Number of tracking channels	1
Satellite mask list	9

To show the quality of the tracking by the third-order DPLL, various metrics are evaluated

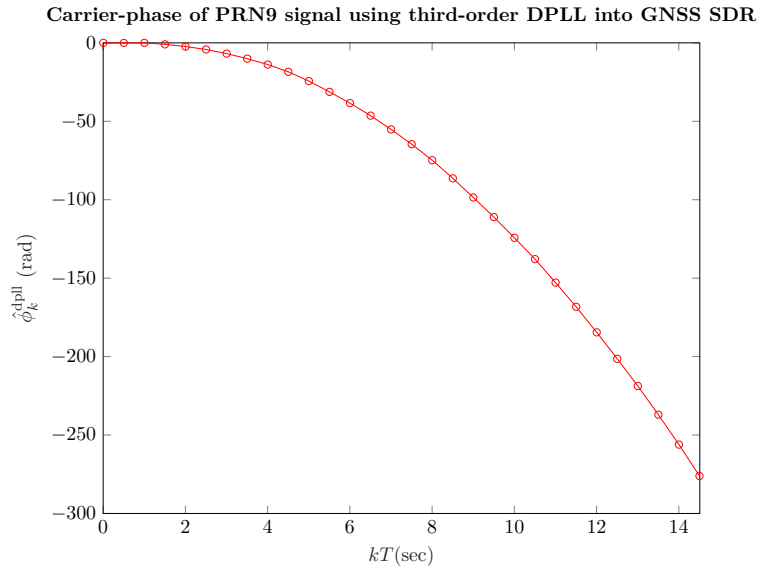


Figure 5.2: Carrier-phase tracking of PRN9 satellite by using a third-order DPLL with ATAN2 into GNSS SDR receiver

in Fig. 5.3. In particular,

1. up-left: the I and Q components of the Prompt correlator are shown. As can be seen, there is no correlation power on Q while we see on I the entire correlation power with ± 1 variation due to the navigation bit.

2. up-right: the Early (E), Prompt (P) and Late (L) components are shown. The track does seem to be achieved given by the higher value P component than E and L components which are symmetric, as expected from theory (more details are explained in section 1.2.2.2)

3. bottom-left: the scatter plot of I and Q components. We recognize a classical *Binary Phase Shift Keying* (BPSK) mapping [166], supporting a successful track.

4. bottom-right: the *Phase Lock Indicator* (PLI) is shown. This is a qualitative metrics of estimated track in DPLL [5]. Theoretically, when the PLI is around the value 1, the carrier-phase is on lock, which is exactly our case because we consider a high C/N_0 signal.

Finally, the obtained phase in Fig. 5.2 is used as phase-input into the synthetic signal which is generated as in (4.1) (or (2.1)), *viz.*

$$s_k^{\text{realistic}} = \alpha e^{j\phi_k^{\text{realistic}}} + n_k, \tag{5.4}$$

where $\phi_k^{\text{realistic}} \triangleq \hat{\phi}_k^{\text{dpll}}$ in Fig. 5.2, α is the signal amplitude and n_k is complex white and Gaussian with known power σ_n^2, namely $n_k \sim \mathcal{CN}(0, \sigma_n^2)$ at instant k.

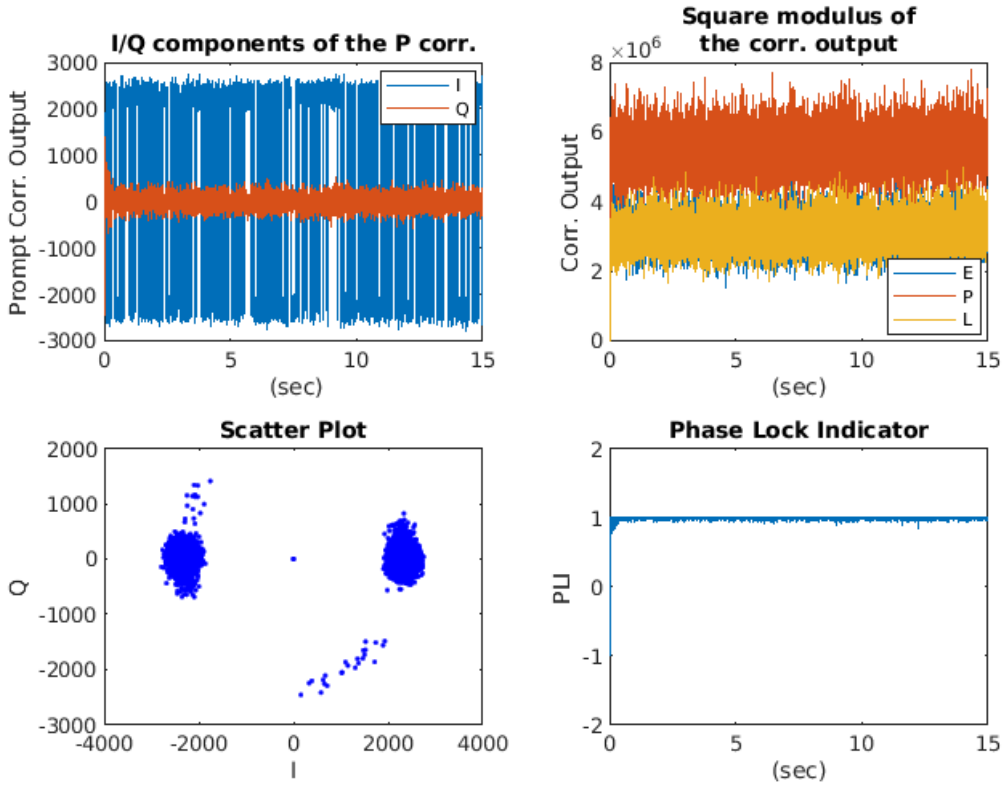


Figure 5.3: Metrics used to validate the estimated carrier-phase by the DPLL into the GNSS SDR receiver

5.1.4 Carrier-to-noise-ratio, noise variance and amplitude estimation

Before showing the performance analyses, in particular the one using real GNSS data, a brief description of the estimation of parameters such as carrier-to-noise-ratio, noise variance and amplitude used in the GNSS SDR is needed. As remarked in the design of the RVB algorithm in the previous chapters, we assume that the signal amplitude α and the noise variance σ_n^2 are known. Note that, on that regard, the knowledge of the carrier-to-noise ratio C/N_0 is useful. The latter is also considered an important parameter describing the GNSS receiver performance and its value determines the precision of pseudo-range and carrier-phase calculations [167]. Therefore, the estimation of C/N_0 was already implemented by default in our GNSS SDR receiver [165]. The method used is the so-called *Variance Summing Method* (VSM) [168, 169]. The latter turns out to be of particular interest for medium-to-low SNRs while taking into account both estimation error performance and implementation complexity [170]. It is based on computing the mean and variance of the tracked in-phase (I) and quadrature (Q) signal components, from which noise and signal power levels are estimated, and then the $\widehat{C/N_0}$ is computed. Note that the noise variance $\hat{\sigma}_n^2$ at the denominator of $\widehat{C/N_0}$ expression [168, Eq.(12)] is the one used in our RVB processing. Finally, the amplitude $\hat{\alpha}$ is then calculated using (2.3) where SNR is given in (2.5).

5.2 Performance analysis

In this section, we compare the RVB algorithm (4.23) with traditional techniques such as the DPLL and KF-based DPLL using synthetic signal with a realistic GNSS phase input and real GNSS signal.

5.2.1 Comparison using synthetic GNSS data based on realistic carrier-phase

5.2.1.1 Processing parameters

The signal in (5.4) is used as input for comparing the estimators such as RVB, DPLL and KF-based DPLL. The latter are set at third-order and initialized as follows:

1. *RVB initialization*: a value of $q_{\max} = 50$ is chosen;
2. *DPLL setting*: an ATAN2 discriminator is used for the DPLL. The loop coefficients of the filter have been calculated by using Tab.V (phase/phase-rate feedback and super-critically damped response) in [69]. Note that the chosen loop bandwidth B_L in Tab. 5.3 is a classical loop settings in *Commercial Off-The-Shelf* (COTS);
3. *KF-based DPLL setting*: to have a fair comparison, an ATAN2 discriminator is used for the KF-based DPLL. Accordingly, the phase noise power is approximated as in (2.56) [131, A-13].

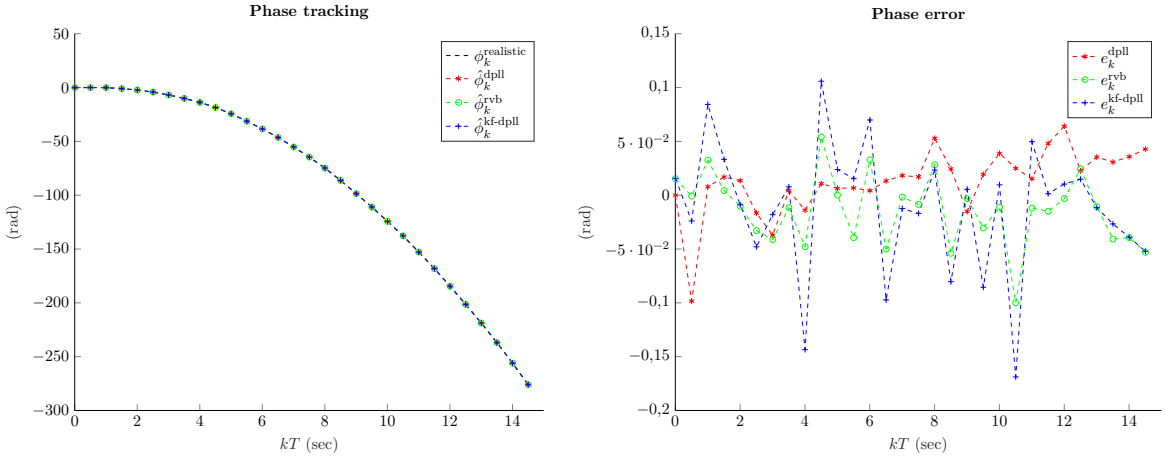
Their processing parameters are summarized in Tab. 5.3.

Table 5.3: Processing parameters

Integration time	T	0.001 s
RVB PSDs	$\sqrt{\Sigma_{\mathrm{p}} T}$	0.8π rad
	$\sqrt{\Sigma_{\mathrm{pv}} T}$	20π rad/s
	$\sqrt{\Sigma_{\mathrm{pva}} T}$	100π rad/s^2
KF-DPLL PSDs	$\sqrt{\Sigma_{\mathrm{p}} T}$	0.2π rad
	$\sqrt{\Sigma_{\mathrm{pv}} T}$	0.8π rad/s
	$\sqrt{\Sigma_{\mathrm{pva}} T}$	0.2π rad/s^2
DPLL loop bandwidth	B_L	10 Hz

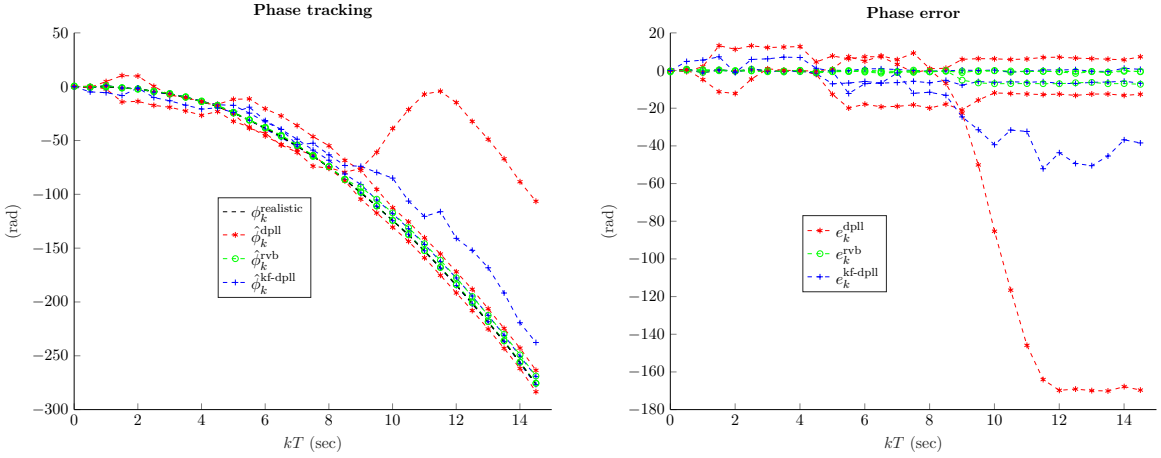
5.2.1.2 Comparison using synthetic GNSS data

In this paragraph, we compare the three estimators using the synthetic signal presented in (5.4). Before using the real GNSS data, this analysis gives us a first look to the behavior of the considered techniques by using a realistic GNSS phase input. The comparison is done on single runs, firstly using a high C/N_0 and then with a degraded one in Figs.5.4a-5.4b, respectively. As can be noted, all the estimators are able to track the phase when the noise level is low. On the other hand, things get complicated when C/N_0 decreases. In fact, the DPLL and KF-DPLL result to be less robust in terms of cycle slips. Note that the RVB is the technique that is less affected by the occurrence of the slips.



(a) Phase tracking (b) Phase error

Figure 5.4: Phase tracking using synthetic GNSS data at C/N_0 around 52 dB-Hz



(a) Phase tracking (b) Phase error

Figure 5.5: Phase tracking using synthetic GNSS data at C/N_0 around 15 dB-Hz

5.2.2 Comparison using real GNSS data

In this section, the comparison between the three estimators is performed by using real GNSS data. The same value of the processing parameters as in previous section 5.2.1.1 are used. Before showing the comparison results, the GNSS receiver tracking block diagrams are first presented.

5.2.2.1 GNSS receiver tracking block diagram

Figs. 5.6-5.7-5.8 show the GNSS receiver tracking block diagrams of DPLL, KF-based DPLL and RVB estimators, respectively. Firstly, let us concentrate on DPLL architecture, namely Fig. 5.6. The latter represents a more advanced version of Fig. 2.1, however keeping the basic functions. Following the flowchart of Fig. 5.6, we note that the tracking loop starts with a frequency compensation as assumed for our signal model (see section 2.1.1.1). In particular, the frequency is compensated with that estimated at the acquisition stage. Therefore, the acquired phase at the output of the NCO block is equal to $\phi_{\text{acq}_k} = \phi_{\text{acq}_{k-1}} + f_{\text{acq}}/F_s$, where F_s is the sampling frequency which value is given in Tab. 5.2. The generated complex exponential with the acquired phase signal is then correlated with the input signal. After that, the code sequence, estimated by first-order DLL, is removed by correlating it with the incoming signal. The signal is then brought to lower frequency rate by using an integration block. Following, the navigation data is removed too from the signal so as too match with our signal model. In practice, the original receiver is run once to estimate and store the navigation data. It is then removed within the three modified architectures of Figs. 5.6-5.7-5.8. In what follows, the loop in Fig. 5.6 has the same architecture of the generic version of DPLL presented in Fig. 2.1, with a discriminator (ATAN2), filter (third-order) and NCO/signal generator blocks. The two remaining architectures in Figs. 5.7-5.8, namely KF-based DPLL and RVB estimators, have the same structure as described for the DPLL. The only difference regards the use of KF block in Fig. 5.7 and the use of RVB structure after the navigation message removal in Fig. 5.8. Note that the RVB architecture is the same presented in Fig. 3.3, but for high-order loop. Finally, note that though all three architectures implement a third-order loop, we focus our attention on the analysis of the phase estimate in what follows.

5.2.2.2 Numerical results

Phase tracks obtained from the RVB, DPLL and KF-based DPLL algorithms are depicted for the raw signal in Fig. 5.9 and with an added synthetic white Gaussian noise (as described in section 5.1.2.1) in Fig. 5.10 [3 runs are shown]. The respective C/N_0 are ca. 52 dB-Hz and 15 dB-Hz. We consider that track from Fig. 5.9 to be ground truth for Fig. 5.10. Before analyzing the results at a low noise level, we note that all the estimators are capable to track the carrier-phase while using (not degraded) real GNSS data. In particular, zooming on the track at its first stage, we clearly see that the acquisition time of RVB is lower than the traditional techniques which start from a zero state value and adjust their estimation with

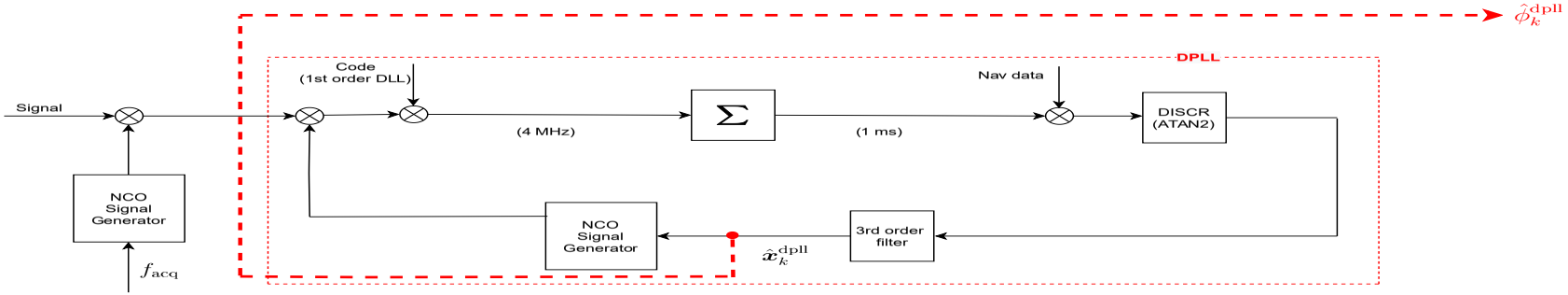


Figure 5.6: GNSS receiver tracking block diagram — DPLL estimator

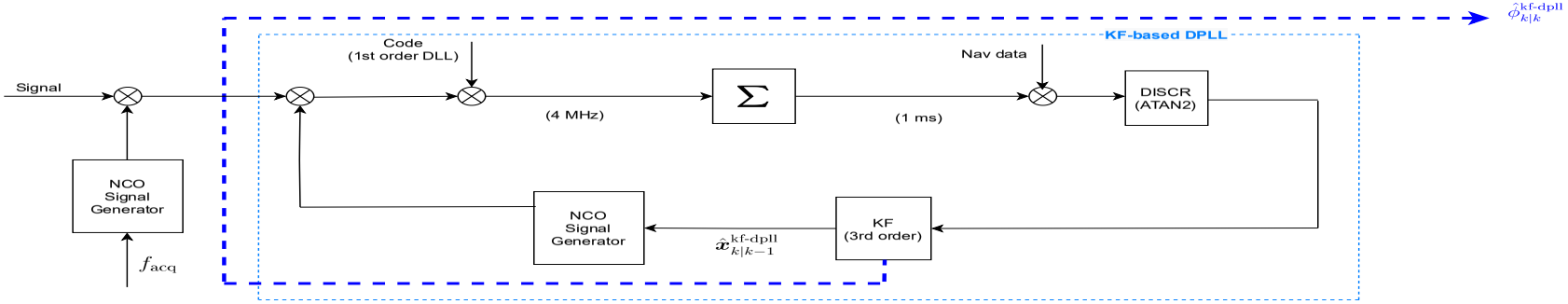


Figure 5.7: GNSS receiver tracking block diagram — KF-based DPLL estimator

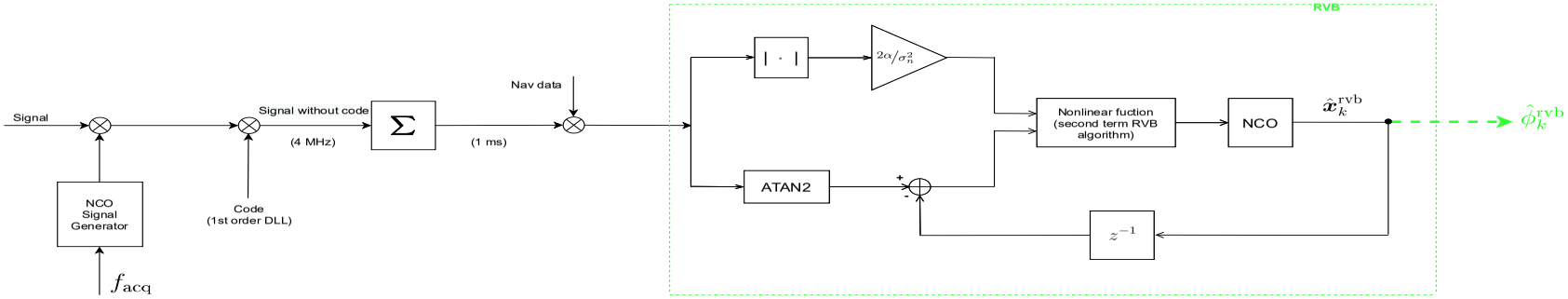


Figure 5.8: GNSS receiver tracking block diagram — RVB estimator

time. In case of low SNR instead, namely in Fig. 5.10, the RVB estimator demonstrates only a very few cycle slips and endures no drop-lock in the depicted runs. Contrarily, the DPLL is affected by a few cycle slips in two phase trajectories and a burst of slips in another. These results are in accordance with that found by using a realistic phase in section 5.2.1.2. The KF-based DPLL exhibits a behavior different from that observed with synthetic data in Fig. 5.5. It seems indeed that in case of fully experimental data it experiences much more cycle slips and drop-lock. The reason of that is still under investigation. However, to start investigating this we used results of the literature [171, 172, 173] that establish an equivalence at steady-state between the KF-based DPLL and the DPLL at third-order. In particular, these results establish a direct expression between the gain filter of the equivalent DPLL and the Kalman gain. From the former, the equivalent loop bandwidth is derived using the expressions given in [69, Tab.IV]. Therefore, we found that the PSDs setting in Tab. 5.3 at 1 ms correspond to a loop bandwidth $B_L \approx 17$ Hz. Consequently, to have fairer analysis between the three estimators, the parameter setting of the KF-based DPLL in Tab. 5.3 have been changed to obtain $B_L \approx 10$ Hz as for the DPLL. The new PSDs considered are presented in Tab. 5.4. The new simulations are run for the KF-DPLL in Fig. 5.11. As can be noted, the occurrence of slips is decreased as expected, however it still appears less robust wrt the RVB algorithm.

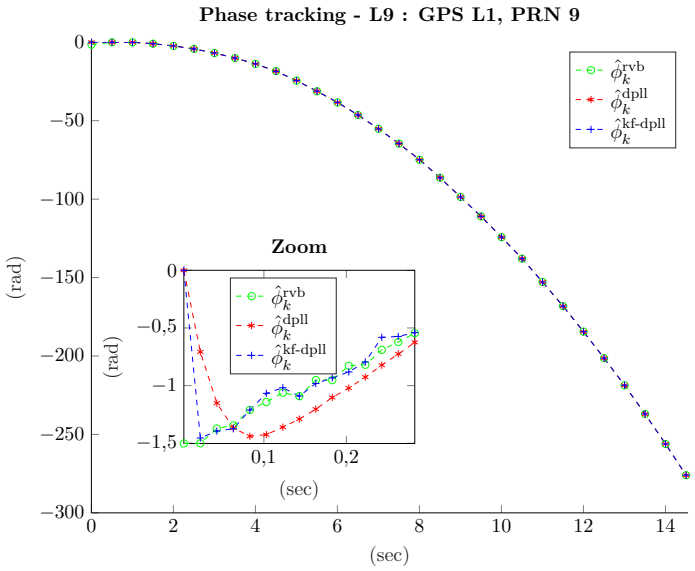
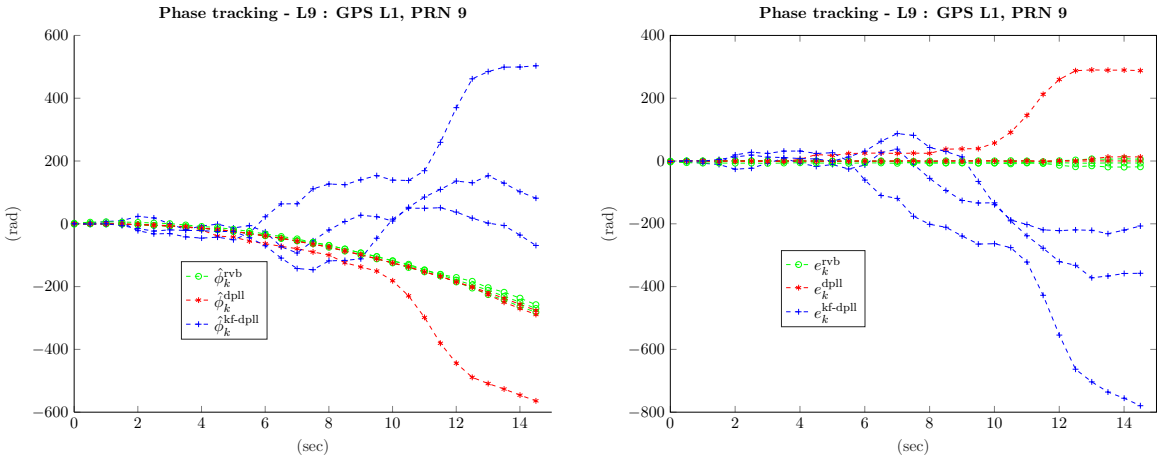


Figure 5.9: Phase tracking using real GNSS data at C/N_0 around 52 dB-Hz

Table 5.4: Processing parameters

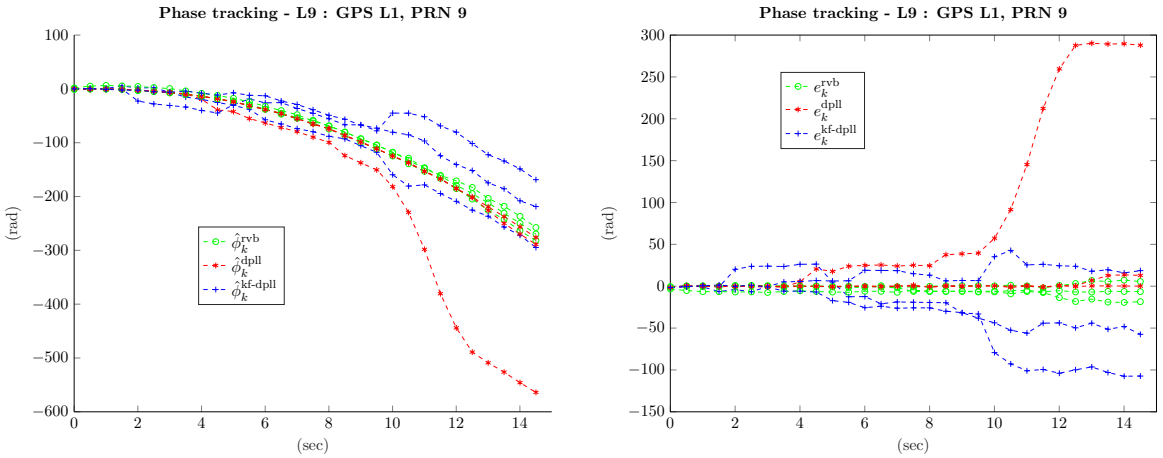
KF-DPLL PSDs	$\sqrt{\Sigma_{\text{p}}T}$	0.08π rad
	$\sqrt{\Sigma_{\text{pv}}T}$	0.8π rad/s
	$\sqrt{\Sigma_{\text{pva}}T}$	0.2π rad/s^2



(a) Phase tracking

(b) Phase error

Figure 5.10: Phase tracking using real GNSS data at C/N_0 around 15 dB-Hz



(a) Phase tracking

(b) Phase error

Figure 5.11: Phase tracking using real GNSS data at C/N_0 around 15 dB-Hz (when KF-DPLL setting is modified as in Tab.5.4)

5.3 Summary

In this chapter, we analyzed the RVB algorithm for high-order phase dynamics by using real GNSS data. A GNSS software-defined-radio receiver previously developed by ISAE-SUPAERO has been used. In particular, a comparison with traditional techniques such as DPLL and KF-based DPLL is performed in case of a synthetic signal using a realistic phase dynamics and real GNSS signal. Regarding the former signal, the analysis is carrier out to have a first idea of the behavior of the three estimators while using realistic phase dynamics. Then, they are compared with a raw GNSS data set that is degraded by adding white Gaussian noise. Results show that all estimators are capable to track the phase dynamics in case of

low noise level. Contrarily, when the signal is degraded, only the RVB algorithm keeps a lock of phase for all runs simulated with a few cycle slips.

Conclusion and future work

Conclusion

This Ph.D. thesis is concerned by the development of a robust carrier-phase tracking algorithm based on Variational Bayesian inference. Carrier-phase estimation has become a fundamental task in many various engineering applications such as navigation. About the latter, phase measurements provide high accuracy of a user/system's position (*e.g.*, in case of GNSS, carrier phase measurements offer a centimeter-level position estimation against the metric-level accuracy provided by the conventional code measurements). However, phase measurements obtained by conventional phase tracking algorithms can be severely impaired by the presence of ambiguous phase jumps, known as cycle slips. Sources causing the slip occurrence can be various from signal obstruction in the propagation channel to low signal-to-noise ratio. Therefore, it may happen that these local losses of lock can even lead the tracking loop to a complete drop-lock from which it never recovers. Reacquisition of the satellite signal is then necessary which severely afflicts the positioning efficiency. Therefore, the main problem of this thesis is the cycle slipping occurrence in carrier-phase tracking which is weakened by proposing a robust nonlinear phase algorithm.

In the first chapter, the GNSS context is introduced. In particular, we focus our attention on fundamentals of GNSS, the signal structure and the GNSS receiver. We introduce the carrier-phase measurements, explaining the advantages of the latter over the code measurements. However, the cons of the phase measurements are acknowledged, in particular the problems due to cycle slips. Finally, the positioning techniques which use carrier-phase measurements are presented.

In the second chapter, we introduce the well-known DPLL. In particular, we present a linear and nonlinear analysis of the latter with the derivation of the phase variance, the phase error PDF, and finally, the investigation of the cycle slip phenomenon. We examine the limitations of this tracking loop in terms of robustness, due to its design based on linear and Gaussian behavior of the phase error. Therefore, an overview of existing robust phase techniques from variable bandwidth algorithms to particle filter is presented. In particular, we focus our attention on KF-based DPLL, which derives from a Gaussian-linear Bayesian filtering problem and is presented as optimal approach for estimating time-varying parameters (*i.e.*, loop bandwidth). These two techniques, DPLL and KF-based DPLL, are used for the comparison wrt our proposed algorithms in the following chapters.

In the third chapter, we propose a robust nonlinear carrier-phase algorithm considering a slow phase dynamics. The former stems from a Variational Bayesian inference, which allows us in our case to obtain a tractable solution while approximating the optimal Bayesian filtering problem. In particular, the Restricted Variational Bayesian approximation is used, which gives us a closed-form and easy-to-implement expression of the estimator. The so-called RVB

algorithm preserves the nonlinear nature of the measurement equation which is beneficial at a low signal-to-noise ratio. Moreover, we underline similitudes and differences between the new RVB algorithm and the traditional KF-based DPLL. To analyze the performance of the RVB algorithm, a comparison using synthetic data is made between the latter and traditional techniques such as DPLL and KF-based DPLL. In particular, we compared them on one hand through metrics assessing the precision of estimation with phase error-modulo, on the other hand through metrics assessing cycle slipping such as the slip rate and mean time to first slip. The comparison is made while fixing a low signal-to-noise ratio and varying the noise processes of the robust estimators and loop bandwidth of the DPLL. In both metrics types considered, we have shown the benefits of RVB estimator for both slow phase dynamics considered (*viz.* phase -step and -ramp inputs). Furthermore, this analysis let us understand the role played by the noise process of our nonlinear RVB algorithm wrt the loop bandwidth of linear techniques such as DPLL and KF-based DPLL.

In the fourth chapter, since the first-order algorithm may be not sufficient in realistic applications, we design a robust phase tracking technique at any order dynamics. The latter can be seen as an extension of the first-order RVB algorithm presented in the third chapter, where now not only the phase but are also estimated its derivatives. In this case too, two approximations are applied to retrieve the estimator expression: on one hand, we use the Variational Bayesian inference; on the other hand, we apply the Restricted Variational Bayesian approximation which gives us a closed-form solution of the estimator. Although the state-space model is of higher-complexity than the one in third chapter, the RVB recursive expression results to remarkably have similitude with that obtained at first order in terms of nonlinearity. As in case of slow dynamics, a comparison using synthetic data is made between the RVB algorithm and traditional techniques such as DPLL and KF-based DPLL. To address real-life applications, a third-order phase dynamics (*viz.* phase-acceleration input) is considered. Accordingly, a third-order motion model such as PVA model is used. Similar to the results obtained for the proposed estimator in the third chapter, the nonlinear RVB algorithm at any order dynamics shows significant performance improvement in terms of precision of estimation and cycle slipping occurrence, especially at low signal-to-noise ratio.

So far, the proposed algorithms have been assessed using only synthetic data. Therefore, in the fifth chapter, we implement the RVB algorithm at high-order dynamics in a GNSS software-defined-radio receiver. The latter has been previously developed by our university lab, and its architecture is described. In particular, we focus our attention on the tracking block where the RVB algorithm is implemented and compared with traditional techniques such as DPLL and KF-based DPLL. Before presenting the real GNSS data results, the comparison is firstly showed using synthetic signal with an actual (and validated) tracked phase by a third-order DPLL using real GNSS data. This analysis is conducted to have a first idea of the algorithms' behavior when using a realistic phase input. Then, the comparison is made using real GNSS data. In particular, a single-satellite is tracked in two different scenarios: 1) considering the original real GNSS data; 2) considering a degraded version of the real GNSS data by adding additive white Gaussian noise. We show that the RVB algorithm is the only estimator to keep the lock with a few cycle slips occurring in both scenarios evaluated wrt the other two traditional techniques, where a burst of slips occurs for various runs at low

signal-to-noise ratio.

Future work

At the end of this thesis, we can establish some guidelines for future work. In particular, we suggest investigating the following three research axes for the short- and long-term analysis.

Modes of the posterior distribution for RVB algorithm at high-order dynamics We have observed (though not depicted in the manuscript) that the RVB algorithm at high-order dynamics may diverge from the true phase due to numerical instabilities. In fact, we have noted that for certain scenarios (when the process noise variance is low) the denominator of the recursive equation (4.23) becomes very small numerically, inducing an erroneous track. This phenomenon has already been encountered for the RVB-based algorithm in [139]. The bypass proposed was to replace locally the MMSE estimator by the MAP estimator of the phase. We propose thus to extend in the first place this method to the high order case or find alternative methods to address this numerical problem.

Influence of the noise covariance matrix $\boldsymbol{Q}$ on the recursion process of the RVB algorithm at high-order dynamics As we noted in (4.23), the recursive expression of the RVB algorithm at high-order dynamics only depends on the first column of the noise covariance matrix $\boldsymbol{Q}$, namely $\boldsymbol{Q}_1$. Therefore, the physical meaning of the latter is in a way lost, and an empirical set of its PSDs (or standard deviations) is needed. In our case, we tried to set them to give enough weight to the innovation term in (4.23). However, it could be interesting to computationally investigate how each term of $\boldsymbol{Q}_1$ influence the recursion process and their "physical" meaning.

Bayesian bounds Lower bounds give an indication of performance limitations, and consequently, they can also be used to determine wherever imposed performance requirements are realistic or not. In time-invariant statistical models, a commonly used lower bound is the *Cramér–Rao Bound* (CRB), given by the inverse of the Fisher information matrix [157]. Therefore, it could be interesting to calculate this bound for our nonlinear RVB estimator. In case of discrete-time nonlinear filtering, more specifically Bayesian bounds can be calculated as described in [174, 175].

Appendix A

RVB algorithm in case of slow dynamics

In this Appendix, we derive the expression (3.17) and (3.18) of the RVB estimator. The same principle described in [139] is applied here too. Note that our derivations use extensively the expression of the Fourier series [82, p. 376, 9.6.34]

$$\exp\{\beta\cos(x)\} = I_0\{\beta\} + 2\sum_{q=1}^{+\infty} I_q\{\beta\}\cos(qx), \tag{A.1}$$

where $I_q\{x\} = {}^1\!/_\pi \int_0^\pi e^{x\cos(\theta)}\cos(q\theta)d\theta$ is the modified Bessel function of the first kind and of order q.

A.1 Proof of (3.17)

Using (3.16) and (3.14b), the RVB estimator at $k = 1$ can be expressed as

$$\hat{\phi}_1^{\text{rvb}} = \frac{\int_{\phi_1} \phi_1 f(\phi_1)\exp(\beta_1\cos(\phi_1 - \psi_1))d\phi_1}{\int_{\phi_1} f(\phi_1)\exp(\beta_1\cos(\phi_1 - \psi_1))d\phi_1}. \tag{A.2}$$

Using (A.1), (A.2) can be rewritten as

$$\hat{\phi}_1^{\text{rvb}} = \frac{\int_{-\pi}^{\pi} \phi_1\{I_0(\beta_1) + 2\sum_{q=1}^{+\infty} I_q(\beta_1)\cos[q(\phi_1 - \psi_1)]\}d\phi_1}{\int_{-\pi}^{\pi}\{I_0(\beta_1) + 2\sum_{q=1}^{+\infty} I_q(\beta_1)\cos[q(\phi_1 - \psi_1)]\}d\phi_1} = \frac{n_1}{d_1}. \tag{A.3}$$

On one hand, the denominator d_1 corresponds to the normalizing constant of a Von Mises pdf [81, p. 36], so that

$$d_1 = 2\pi I_0(\beta_1), \tag{A.4}$$

since, for each $q \in \mathcal{N}^*$,

$$\int_{-\pi}^{\pi} \cos[q(\phi_1 - \psi_1)]d\phi_1 = 0. \tag{A.5}$$

On the other hand, the numerator n_1 can be solved through integral by parts. Its expression is equal to

$$\begin{aligned} n_1 &= \int_{-\pi}^{\pi} \phi_1 \{I_0(\beta_1) + 2\sum_{q=1}^{+\infty} I_q(\beta_1)\cos[q(\phi_1-\psi_1)]\} d\phi_1 \\ &= \int_{-\pi}^{\pi} \phi_1 I_0(\beta_1) d\phi_1 + \int_{-\pi}^{\pi} 2\phi_1 \sum_{q=1}^{+\infty} I_q(\beta_1)\cos[q(\phi_1-\psi_1)] d\phi_1 \\ &= I_0(\beta_1)\Big[\frac{\phi_1^2}{2}\Big]_{-\pi}^{\pi} + 2\sum_{q=1}^{+\infty} I_q(\beta_1)\int_{-\pi}^{\pi} \phi_1 \cos[q(\phi_1-\psi_1)] d\phi_1. \end{aligned} \tag{A.6}$$

As anticipated before, the integral in the second term is solved through integral by parts method. Therefore,

$$\int_{-\pi}^{\pi} \phi_1 \cos[q(\phi_1-\psi_1)] d\phi_1 = \Big[\frac{\phi_1 \sin[q(\phi_1-\psi_1)]}{q}\Big]_{-\pi}^{\pi} - \frac{1}{q}\int_{-\pi}^{\pi} \sin[q(\phi_1-\psi_1)] d\phi_1. \tag{A.7}$$

Knowing that the integral at second term is equal to zero, therefore, the numerator n_1 can be rewritten as

$$n_1 = 2\sum_{q=1}^{+\infty} I_q(\beta_1)\Big[\frac{\pi\sin[q(\pi-\psi_1)] + \pi\sin[q(-\pi-\psi_1)]}{q}\Big]. \tag{A.8}$$

However, using trigonometric formula $\sin(\alpha-\beta) = \sin\alpha\cos\beta - \cos\alpha\sin\beta$, (A.8) can be rewritten as

$$n_1 = -4\pi\sum_{q=1}^{+\infty}(-1)^q I_q(\beta_1)\frac{\sin(q\psi_1)}{q}. \tag{A.9}$$

Finally, $\hat{\phi}_k^{\text{rvb}}$ for $k=1$ is

$$\hat{\phi}_k^{\text{rvb}} = \frac{n_1}{d_1} = -2\frac{\sum_{q=1}^{\infty}(-1)^q I_q(\beta_1)\frac{\sin(q\psi_1)}{q}}{I_0(\beta_1)}. \tag{A.10}$$

A.2 Proof of (3.18)

In this section we develop the expression (3.18) of the RVB estimator for $k>1$. Using (3.16) and (3.15b), the RVB estimator can be formulated as

$$\hat{\phi}_k^{\text{rvb}} = \frac{\int_{\phi_k} \phi_k \exp\left\{\beta_k\cos(\phi_k-\psi_k) - \frac{[\phi_k-\hat{\phi}_{k-1}^{\text{rvb}}]^2}{2\sigma_\phi^2}\right\} d\phi_k}{\int_{\phi_k} \exp\left\{\beta_k\cos(\phi_k-\psi_k) - \frac{[\phi_k-\hat{\phi}_{k-1}^{\text{rvb}}]^2}{2\sigma_\phi^2}\right\} d\phi_k}. \tag{A.11}$$

The numerator and the denominator of the former expression are denoted respectively as n_k and d_k, i.e., $\hat{\phi}_k^{\text{rvb}} \overset{\text{def}}{=} {}^{n_k}/_{d_k}$. We will start the calculation by the denominator d_k. The latter is

equal to

$$d_k = \int_{\phi_k} \exp\left\{ \beta_k \cos(\phi_k - \psi_k) - \frac{[\phi_k - \hat{\phi}_{k-1}^{\text{rvb}}]^2}{2\sigma_\phi^2} \right\} d\phi_k, \tag{A.12}$$

which, as can be noted, is the product of a Von Mises and Gaussian distributions. Therefore, it can be rewritten as

$$d_k = 2\pi\sqrt{2\pi}\sigma_\phi I_0(\beta_k) \int_{\phi_k} \mathrm{N}\Big(\phi_k | \hat{\phi}_{k-1}^{\text{rvb}}, \sigma_\phi^2\Big) \mathrm{T}\Big(\phi_k | \psi_k, \beta_k\Big) d\phi_k. \tag{A.13}$$

The solution of the integral is given by [176, Eq.(41)], therefore, the denominator d_k can be expressed as

$$d_k = \sqrt{2\pi}\sigma_\phi \Big\{ I_0(\beta_k) + 2 \sum_{q=1}^{+\infty} I_q(\beta_k) e^{-\frac{q^2\sigma_\phi^2}{2}} \cos[q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})] \Big\}. \tag{A.14}$$

The numerator n_k is then equal to

$$n_k = \int_{\phi_k} \phi_k \exp\left\{ \beta_k \cos(\phi_k - \psi_k) - \frac{[\phi_k - \hat{\phi}_{k-1}^{\text{rvb}}]^2}{2\sigma_\phi^2} \right\} d\phi_k, \tag{A.15}$$

which can be rewritten as

$$n_k = \int_{\phi_k} \exp\left\{ \beta_k \cos(\phi_k - \psi_k) \right\} \phi_k \exp\left\{ \frac{[\phi_k - \hat{\phi}_{k-1}^{\text{rvb}}]^2}{2\sigma_\phi^2} \right\} d\phi_k. \tag{A.16}$$

Using Parseval's theorem,

$$\int_{t=-\infty}^{+\infty} x(t) y(t)^* dt = \int_{f=-\infty}^{+\infty} \mathcal{F}\{x\}(f) \mathcal{F}\{y\}^*(f) df, \tag{A.17}$$

where $\mathcal{F}\{\cdot\}(f)$ denotes the Fourier transform of a specific function. The numerator n_k can be rewritten as

$$n_k = \int_f \mathcal{F}\left\{ \exp\Big[\beta_k \cos(\phi_k - \psi_k)\Big] \right\}(f) \mathcal{F}^*\left\{ \phi_k \exp\Big[\frac{(\phi_k - \hat{\phi}_{k-1}^{\text{rvb}})^2}{2\sigma_\phi^2} \Big] \right\}(f) df. \tag{A.18}$$

Separately, the two Fourier transforms are developed. The latter are calculated with respect to the variable ϕ_k. Therefore, using (A.1), the first Fourier transform of (A.18) can be written

as

$$
\begin{aligned}
&\mathcal{F}\Big\{\exp\big[\beta_k\cos(\phi_k-\psi_k)\big]\Big\}(f)=\\
&=\int_{\phi_k}\Big\{I_0(\beta_k)+2\sum_{q=1}^{+\infty}I_q(\beta_k)\cos[q(\phi_k-\psi_k)]\Big\}e^{-i2\pi f\phi_k}d\phi_k\\
&=\int_{\phi_k}I_0(\beta_k)e^{-i2\pi f\phi_k}d\phi_k+2\sum_{q=1}^{+\infty}I_q(\beta_k)\int_{\phi_k}\cos[q(\phi_k-\psi_k)]e^{-i2\pi f\phi_k}d\phi_k\\
&=I_0(\beta_k)\delta(f)+2\sum_{q=1}^{+\infty}I_q(\beta_k)\frac{1}{2}\Big[\int_{\phi_k}e^{iq(\phi_k-\psi_k)}e^{-i2\pi f\phi_k}d\phi_k+\int_{\phi_k}e^{-iq(\phi_k-\psi_k)}e^{-i2\pi f\phi_k}d\phi_k\Big]\\
&=I_0(\beta_k)\delta(f)+\sum_{q=1}^{+\infty}I_q(\beta_k)\Big[e^{-iq\psi_k}\int_{\phi_k}e^{iq\phi_k}e^{-i2\pi f\phi_k}d\phi_k+e^{iq\psi_k}\int_{\phi_k}e^{-iq\phi_k}e^{-i2\pi f\phi_k}d\phi_k\Big]\\
&=I_0(\beta_k)\delta(f)+\sum_{q=1}^{+\infty}I_q(\beta_k)\Big[e^{-iq\psi_k}\delta(f-\frac{q}{2\pi})+e^{iq\psi_k}\delta(f+\frac{q}{2\pi})\Big].
\end{aligned}
\tag{A.19}
$$

The second Fourier transform of (A.18) is developed as

$$
\begin{aligned}
&\mathcal{F}\Big\{\phi_k\exp\Big[\frac{(\phi_k-\hat{\phi}_{k-1}^{\mathrm{rvb}})^2}{2\sigma_\phi^2}\Big]\Big\}(f)=\\
&=\int_{\phi_k}\phi_k\exp\Big[\frac{(\phi_k-\hat{\phi}_{k-1}^{\mathrm{rvb}})^2}{2\sigma_\phi^2}\Big]e^{-i2\pi f\phi_k}d\phi_k\\
&=\int_{\phi_k}\phi_k\exp\Big[-\frac{1}{2\sigma_\phi^2}\phi_k^2+\frac{2\hat{\phi}_{k-1}^{\mathrm{rvb}}-i4\pi f\sigma_\phi^2}{2\sigma_\phi^2}\phi_k-\frac{(\hat{\phi}_{k-1}^{\mathrm{rvb}})^2}{2\sigma_\phi^2}\Big]d\phi_k\\
&=\exp\Big[-\frac{(\hat{\phi}_{k-1}^{\mathrm{rvb}})^2}{2\sigma_\phi^2}\Big]\int_{\phi_k}\phi_k\exp\Big[-\frac{1}{2\sigma_\phi^2}\phi_k^2+\frac{2\hat{\phi}_{k-1}^{\mathrm{rvb}}-i4\pi f\sigma_\phi^2}{2\sigma_\phi^2}\phi_k\Big]d\phi_k.
\end{aligned}
\tag{A.20}
$$

Using the integral by parts technique and Gaussian integral, the solution can be found as

$$
\int_{-\infty}^{+\infty}xe^{-ax^2+bx}dx=\frac{\sqrt{\pi}b}{2a^{3/2}}e^{\frac{b^2}{4a}}. \tag{A.21}
$$

Therefore, (A.20) can be rewritten as

$$
\begin{aligned}
&\mathcal{F}\Big\{\phi_k\exp\Big[\frac{(\phi_k-\hat{\phi}_{k-1}^{\mathrm{rvb}})^2}{2\sigma_\phi^2}\Big]\Big\}(f)=\\
&=\exp\Big[-\frac{(\hat{\phi}_{k-1}^{\mathrm{rvb}})^2}{2\sigma_\phi^2}\Big]\frac{\sqrt{\pi}\frac{2\hat{\phi}_{k-1}^{\mathrm{rvb}}-i4\pi f\sigma_\phi^2}{2\sigma_\phi^2}}{2\Big(\frac{1}{2\sigma_\phi^2}\Big)^{3/2}}\exp\Big[-\frac{\Big(\frac{2\hat{\phi}_{k-1}^{\mathrm{rvb}}-i4\pi f\sigma_\phi^2}{2\sigma_\phi^2}\Big)^2}{4\frac{1}{2\sigma_\phi^2}}\Big].
\end{aligned}
\tag{A.22}
$$

Manipulating (A.22), the solution of the second Fourier transform of (A.18) is

$$\begin{aligned}&\mathcal{F}\Big\{\phi_k \exp\Big[\frac{(\phi_k-\hat{\phi}_{k-1}^{\text{rvb}})^2}{2\sigma_\phi^2}\Big]\Big\}(f) = \\ &= \sqrt{2\pi}\sigma_\phi(\hat{\phi}_{k-1}^{\text{rvb}} - i2\pi f\sigma_\phi^2)\exp\Big[-2\pi^2 f^2\sigma_\phi^2 - i2\pi f\hat{\phi}_{k-1}^{\text{rvb}}\Big].\end{aligned} \tag{A.23}$$

Plugging then (A.19) and (A.23) into (A.18) yields

$$\begin{aligned}n_k = \int_f \Big\{ I_0(\beta_k)\delta(f) + \sum_{q=1}^{+\infty} I_q(\beta_k)\Big[e^{-iq\psi_k}\delta(f-\frac{q}{2\pi}) + e^{iq\psi_k}\delta(f+\frac{q}{2\pi})\Big]\Big\} \\ \times \Big\{\sqrt{2\pi}\sigma_\phi(\hat{\phi}_{k-1}^{\text{rvb}} + i2\pi f\sigma_\phi^2)\exp\Big[-2\pi^2 f^2\sigma_\phi^2 + i2\pi f\hat{\phi}_{k-1}^{\text{rvb}}\Big]\Big\} df.\end{aligned} \tag{A.24}$$

The solution of this integral is given as follows

$$\begin{aligned}n_k = \sqrt{2\pi}\sigma_\phi\Big\{ &\int_f I_0(\beta_k)\delta(f)\hat{\phi}_{k-1}^{\text{rvb}} e^{[-2\pi^2 f^2\sigma_\phi^2 + i2\pi f\hat{\phi}_{k-1}^{\text{rvb}}]} df \\ &+ \int_f I_0(\beta_k)\delta(f)(+i2\pi f\sigma_\phi^2) e^{[-2\pi^2 f^2\sigma_\phi^2 + i2\pi f\hat{\phi}_{k-1}^{\text{rvb}}]} df \\ &+ \int_f \sum_{q=1}^{+\infty} I_q(\beta_k)\hat{\phi}_{k-1}^{\text{rvb}}\Big[e^{-iq\psi_k}\delta(f-\frac{q}{2\pi}) + e^{iq\psi_k}\delta(f+\frac{q}{2\pi})\Big] e^{[-2\pi^2 f^2\sigma_\phi^2 + i2\pi f\hat{\phi}_{k-1}^{\text{rvb}}]} df \\ &+ \int_f \sum_{q=1}^{+\infty} I_q(\beta_k)(+i2\pi f\sigma_\phi^2)\Big[e^{-iq\psi_k}\delta(f-\frac{q}{2\pi}) + e^{iq\psi_k}\delta(f+\frac{q}{2\pi})\Big] e^{[-2\pi^2 f^2\sigma_\phi^2 + i2\pi f\hat{\phi}_{k-1}^{\text{rvb}}]} df \Big\}.\end{aligned} \tag{A.25}$$

Remembering the shifting propriety of the Dirac Delta,

$$\begin{aligned}\int_{-\infty}^{+\infty} x(t)\delta(t-t_0)dt = x(t_0), \\ x(t)\delta(t-t_0) = x(t_0)\delta(t-t_0),\end{aligned} \tag{A.26}$$

the integral (A.25) can be solved as

$$\begin{aligned}n_k = \sqrt{2\pi}\sigma_\phi\Big\{ &I_0(\beta_k)\hat{\phi}_{k-1}^{\text{rvb}} + \sum_{q=1}^{+\infty} I_q(\beta_k)\hat{\phi}_{k-1}^{\text{rvb}}\Big[e^{\left(-\frac{q^2\sigma_\phi^2}{2} - iq(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})\right)} + e^{\left(-\frac{q^2\sigma_\phi^2}{2} + iq(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})\right)}\Big] \\ &+ \sum_{q=1}^{+\infty} I_q(\beta_k)\Big[(iq\sigma_\phi^2)e^{\left(-\frac{q^2\sigma_\phi^2}{2} - iq(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})\right)} + (-iq\sigma_\phi^2)e^{\left(-\frac{q^2\sigma_\phi^2}{2} + iq(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})\right)}\Big]\Big\}.\end{aligned} \tag{A.27}$$

Finally, knowing the Euler's formula about sin and cos, the numerator n_k is expressed as

$$\begin{aligned} n_k = \sqrt{2\pi}\sigma_\phi \Bigg\{ & I_0(\beta_k)\hat{\phi}_{k-1}^{\text{rvb}} + 2\sum_{q=1}^{+\infty} I_q(\beta_k) e^{-\frac{q^2\sigma_\phi^2}{2}} \\ & \times \left[\hat{\phi}_{k-1}^{\text{rvb}} \cos\left(q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})\right) + q\sigma_\phi^2 \sin\left(q(\psi_k - \hat{\phi}_{k-1}^{\text{rvb}})\right)\right] \Bigg\}. \end{aligned} \tag{A.28}$$

Finally, combining (A.28) and (A.14), the expression (3.18) can be recovered.

Appendix B

Equivalence between noise process covariance matrix and loop bandwidth in case of a first-order KF-DPLL

An equivalence between a first-order KF-DPLL and DPLL can be found in literature [173]. In particular, at steady-state the filter gains can be equalized such as

$$K_1^{\text{dpll}} = K_1^{\text{kf-dpll}} \tag{B.1}$$

From the former, the equivalent loop bandwidth is derived using the expression in Tab.IV in [69]. Namely,

$$B_L = \frac{1}{\mathrm{T}} \frac{1}{4\Big(H + \frac{R}{(AP_{k|k}A^{\intercal}+Q)H^{\intercal}}\Big) - 2}, \tag{B.2}$$

where T is the integration time, A and H are respectively the state and measurement matrix, and finally, Q and R are respectively the process and measurement noise covariance matrix. $P_{k|k}$ represents the updated covariance matrix in the KF algorithm as described in (2.50d). In Fig. B.1 the loop bandwidth B_L is shown wrt the noise process covariance matrix Q.

Note that the equivalence has been also studied in case of third-order loop [171, 172, 173], which has been used in section 5.2.2.2.

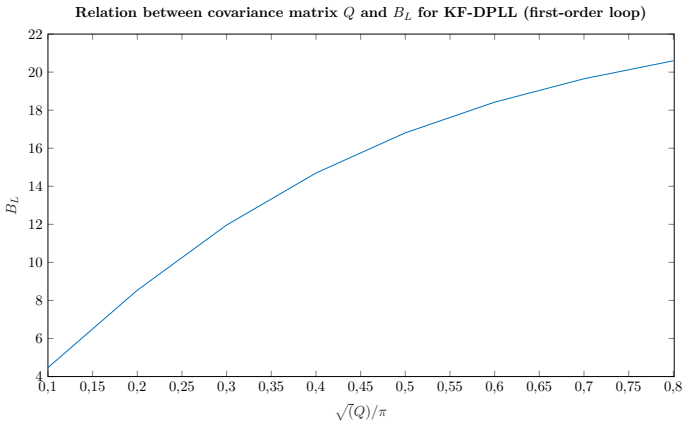


Figure B.1: Relation between noise process covariance matrix and loop bandwidth in case of first-order KF-DPLL. (C/N_0 = 17 dB-Hz; T = 0.02 s; discriminator ATAN2 for both estimators)

Appendix C

RVB algorithm in case of high-order dynamics

C.1 Preliminary results

Herein, we introduce some results that are used in the demonstration of C.2. Note that the indices k have been dropped to lighten the expressions.

1. The first useful result is given by the expression of two integrals already derived in Appendix A (section (A.2)) and reformulated as follows

$$\begin{aligned}\mathcal{I}(\psi,\beta,\mu,\sigma^2) &\stackrel{\text{def}}{=} \int_{\mathcal{R}} \mathrm{T}(x|\psi,\beta)\mathrm{N}(x|\mu,\sigma^2)dx \\ &= \frac{1}{2\pi}\left[1+2\sum_{q=1}^{+\infty}\frac{I_q(\beta)}{I_0(\beta)}e^{-\frac{\sigma^2q^2}{2}}\cos(q[\psi-\mu])\right],\end{aligned} \tag{C.1}$$

and

$$\begin{aligned}\mathcal{J}(\psi,\beta,\mu,\sigma^2) &\stackrel{\text{def}}{=} \int_{\mathcal{R}} x\mathrm{T}(x|\psi,\beta)\mathrm{N}(x|\mu,\sigma^2)dx \\ &= \mu\mathcal{I}(\psi,\beta,\mu,\sigma^2) + \frac{\sigma^2}{\pi}\big[\sum_{q=1}^{+\infty}\frac{I_q(\beta)}{I_0(\beta)}e^{-\frac{\sigma^2q^2}{2}}\sin(q[\psi-\mu])\big].\end{aligned} \tag{C.2}$$

2. The second useful result is a Gaussian product formula, *i.e.*,

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{C}^{-1}) = \mathcal{N}(x_1|\mu_1,P_1)\mathcal{N}(\boldsymbol{x}_{-1}|\boldsymbol{m}(x_1),\boldsymbol{C}_{-1}^{-1}), \tag{C.3}$$

with

$$\boldsymbol{m}(x_1) = (\boldsymbol{\mu}_{-1}+\mu_1\boldsymbol{C}_{-1}^{-1}\boldsymbol{c}_{-1}) - x_1\boldsymbol{C}_{-1}^{-1}\boldsymbol{c}_{-1}, \tag{C.4}$$

and where the notation $\boldsymbol{C}_{-1}$, $\boldsymbol{c}_{-1}$, P_1 have been defined in section 4.2.3. The proof of (C.3) can be found while partitioning a Gaussian multivariate PDF and calculating its marginal PDF; the solution is shown in [177, Par.7.2.6].

3. The third useful result is the result of integral

$$\int x\mathrm{N}(x|m,C)dx = m, \tag{C.5}$$

which is solved by using the Gaussian integral (A.21).

C.2 Proof of (4.20)

In this section, we derive the expression of the RVB estimator (4.20). According to (4.16b), the latter has the form

$$\hat{\boldsymbol{x}}^{\text{rvb}} \overset{\text{def}}{=} \frac{\boldsymbol{n}}{d}, \tag{C.6}$$

with

$$\boldsymbol{n} = \int \boldsymbol{x}\mathrm{T}(x_1|\psi,\beta)\mathrm{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{C}^{-1})d\boldsymbol{x}, \tag{C.7a}$$

$$d = \int \mathrm{T}(x_1|\psi,\beta)\mathrm{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{C}^{-1})d\boldsymbol{x}, \tag{C.7b}$$

with $\boldsymbol{\mu} \overset{\text{def}}{=} \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^{\text{rvb}}$ and where indices k have been dropped to lighten the expressions. Using the Gaussian product formula (C.3), the denominator reduces to

$$\begin{aligned} d &= \int_{x_1} \mathrm{T}(x_1|\psi,\beta)\mathrm{N}(x_1|\mu_1,P_1)\Big[\int_{\boldsymbol{x}_{-1}} \mathrm{N}(\boldsymbol{x}_{-1}|\boldsymbol{m}(x_1),\boldsymbol{C}_{-1}^{-1})d\boldsymbol{x}_{-1}\Big]dx_1 \\ &= \mathcal{I}(\psi,\beta,\mu_1,P_1), \end{aligned} \tag{C.8}$$

where the integral $\mathcal{I}$ is defined in (C.1). Similarly, using (C.3), (C.2) and (C.5), the first element and the last remaining elements of the numerator $\boldsymbol{n}$ are respectively given by

$$\begin{aligned} n_1 &= \int_{x_1} x_1\mathrm{T}(x_1|\psi,\beta)\mathrm{N}(x_1|\mu_1,P_1)\Big[\int_{\boldsymbol{x}_{-1}} \mathrm{N}(\boldsymbol{x}_{-1}|\boldsymbol{m}(x_1),\boldsymbol{C}_{-1}^{-1})d\boldsymbol{x}_{-1}\Big]dx_1 \\ &= \mathcal{J}(\psi,\beta,\mu_1,P_1), \end{aligned} \tag{C.9}$$

and

$$\begin{aligned} \boldsymbol{n}_{-1} &= \int_{x_1} \mathrm{T}(x_1|\psi,\beta)\mathrm{N}(x_1|\mu_1,P_1)\Big[\int_{\boldsymbol{x}_{-1}} \boldsymbol{x}_{-1}\mathrm{N}(\boldsymbol{x}_{-1}|\boldsymbol{m}(x_1),\boldsymbol{C}_{-1}^{-1})d\boldsymbol{x}_{-1}\Big]dx_1 \\ &= \int \boldsymbol{m}(x_1)\mathrm{T}(x_1|\psi,\beta)\mathrm{N}(x_1|\boldsymbol{\mu},\boldsymbol{C}^{-1})dx_1 \\ &= (\boldsymbol{\mu}_{-1} + \mu_1\boldsymbol{C}_{-1}^{-1}\boldsymbol{c}_{-1})\mathcal{I}(\psi,\beta,\mu_1,P_1) - \mathcal{J}(\psi,\beta,\mu_1,P_1)\boldsymbol{C}_{-1}^{-1}\boldsymbol{c}_{-1}. \end{aligned} \tag{C.10}$$

In the last line of (C.10), we have used the expression of the Gaussian mean (C.4). Using results (C.1),(C.2), (C.8) and (C.9), the expression of the RVB estimator is

$$\frac{\boldsymbol{n}}{d} = \boldsymbol{\mu} + 2P_1\frac{\sum_{q=1}^{+\infty} qI_q(\beta)\sin[q(\psi-\mu_1)]e^{-\frac{q^2P_1}{2}}}{I_0(\beta) + 2\sum_{q=1}^{+\infty} I_q(\beta)\cos[q(\psi-\mu_1)]e^{-\frac{q^2P_1}{2}}}\begin{bmatrix}1\\ \\ -\boldsymbol{C}_{-1}^{-1}\boldsymbol{c}_{-1}\end{bmatrix}, \tag{C.11}$$

where we finally recognize the recursive formula (4.20).

Bibliography

[1] Fabio Fabozzi, Stéphanie Bidon, Sébastien Roche, and Benoît Priot. "Robust GNSS Phase Tracking in Case of Slow Dynamics Using Variational Bayes Inference". In: *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. 2020, pp. 1189–1195. DOI: 10.1109/PLANS46316.2020.9110176.

[2] Fabio Fabozzi, Stéphanie Bidon, and Sébastien Roche. "Robust estimation of high-order phase dynamics using Variational Bayes inference". In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 4980–4984. DOI: 10.1109/ICASSP39728.2021.9415028.

[3] Fabio Fabozzi and Stéphanie Bidon. "Robust Phase Tracking using Restricted Variational Bayes Inference". In: *IEEE Transactions on Aerospace and Electronic Systems*.

[4] Elliott Kaplan and Christopher Hegarty. *Understanding GPS: principles and applications*. Artech house, 2005.

[5] Bradford W Parkinson, Per Enge, Penina Axelrad, and James J Spilker Jr. *Global positioning system: Theory and applications, Volume II*. American Institute of Aeronautics and Astronautics, 1996.

[6] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins. *Global positioning system: theory and practice*. Springer Science & Business Media, 2012.

[7] Peter Teunissen and Oliver Montenbruck. *Springer handbook of global navigation satellite systems*. Springer, 2017.

[8] Bertrand T Fang. "Trilateration and extension to global positioning system navigation". In: *Journal of Guidance, Control, and Dynamics* 9.6 (1986), pp. 715–717.

[9] Pedro Ramon Escobal. "Methods of orbit determination". In: *New York: Wiley, 1965* (1965).

[10] Gilbert Strang and Kai Borre. *Linear algebra, geodesy, and GPS*. Siam, 1997.

[11] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and Elmar Wasle. *GNSS–global navigation satellite systems: GPS, GLONASS, Galileo, and more*. Springer Science & Business Media, 2007.

[12] Rodger E Ziemer and Roger L Peterson. *Digital communications and spread spectrum systems*. Macmillan New York, 1985.

[13] Don Torrieri. *Principles of spread-spectrum communication systems*. Vol. 1. Springer, 2005.

[14] James Bao-Yen Tsui. *Fundamentals of global positioning system receivers: a software approach*. Vol. 173. John Wiley & Sons, 2005.

[15] John W Betz. "Binary offset carrier modulations for radionavigation". In: *Navigation* 48.4 (2001), pp. 227–246.

[16] L Ries, F Legrand, L Lestarquit, W Vigneau, and JL Issler. "Tracking and multipath performance assessments of BOC signals using a bit-level signal processing simulator". In: *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)*. 2001, pp. 1996–2010.

[17] J Godet. "Technical Annex to Galileo SRD Signal Plans". In: *STF annex SRD* 2003 (2001).

[18] JJ Spilker Jr. "GPS signal structure and performance characteristics". In: *Navigation* 25.2 (1978), pp. 121–146.

[19] Olivier Julien. "Carrier-phase tracking of future data/pilot signals". In: *ION GNSS*. 2005, pp. 13–16.

[20] Kannan Muthuraman. "Tracking techniques for GNSS data/pilot signals". In: *Ph. D. dissertation* (2010).

[21] Guowang Miao, Jens Zander, Ki Won Sung, and Slimane Ben Slimane. *Fundamentals of mobile data networks*. Cambridge University Press, 2016.

[22] ICD Galileo. *Galileo open service, signal in space interface control document (OS SIS ICD)*. 2008.

[23] Giovanni E Corazza, Claudio Palestini, Raffaella Pedone, and Marco Villanti. "Galileo primary code acquisition based on multi-hypothesis secondary code ambiguity elimination". In: *Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation, Fort Worth, TX, USA*. Vol. 2528. 2007, p. 24592465.

[24] W. De Wilde, J. . Sleewaegen, A. Simsky, J. Van Hees, C. Vandewiele, E. Peeters, J. Grauwen, and F. Boon. "Fast signal acquisition technology for new GPS/Galileo receivers". In: *2006 IEEE/ION Position, Location, And Navigation Symposium*. 2006, pp. 1074–1079. DOI: 10.1109/PLANS.2006.1650712.

[25] Nagaraj C Shivaramaiah, Andrew G Dempster, and Chris Rizos. "Exploiting the secondary codes to improve signal acquisition performance in Galileo receivers". In: *Proceedings of ION GNSS*. 2008, pp. 1497–1506.

[26] Christian Dopper. "Ueber das farbige Licht der Doppelsterne und einiger anderer Gestirre des Himmels". In: *Abhandlungen der k. B^hm* 2 (1842), pp. 465–482.

[27] L. A. van Mastrigt, A. J. van der Wal, and P. J. Oonincx. "Exploiting the Doppler effect in GPS to monitor signal integrity and to detect spoofing". In: *2015 International Association of Institutes of Navigation World Congress (IAIN)*. 2015, pp. 1–8. DOI: 10.1109/IAIN.2015.7352259.

[28] Pratap Misra and Per Enge. "Global positioning system: Signals measurements and Performance (Second Edition)". In: (2006).

[29] S Dinesh, S Shalini, MA Zainal Fitry, J Asmariah, and A Siti Zainun. "Evaluation of the accuracy of Global Positioning System (GPS) speed measurement via GPS simulation". In: *Defence S&T Tech. Bull* 8 (2015), pp. 121–128.

[30] Mohinder S Grewal, Angus P Andrews, and Chris G Bartone. *Global navigation satellite systems, inertial navigation, and integration*. John Wiley & Sons, 2020.

[31] Jeffrey Hugh Reed. *Software radio: a modern approach to radio engineering*. Prentice Hall Professional, 2002.

[32] Jeffrey Hugh Reed. *Software radio: a modern approach to radio engineering*. Prentice Hall Professional, 2002.

[33] Kent Krumvieda, Premal Madhani, Chad Cloman, Eric Olson, John Thomas, Penina Axelrad, and Wolfgang Kober. "A complete IF software GPS receiver: A tutorial about the details". In: *Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2001)*. 2001, pp. 789–829.

[34] Letizia Lo Presti, Politecnico di Torino, Emanuela Falletti, Mario Nicola, and Micaela Troglia Gamba. "Software defined radio technology for GNSS receivers". In: *2014 IEEE Metrology for Aerospace (MetroAeroSpace)*. IEEE. 2014, pp. 314–319.

[35] Jayanta K Ray, M Elizabeth Cannon, and P Fenton. "Mitigation of static carrier phase multipath effects using multiple closely-spaced antennas". In: *Proceedings of the 11th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1998)*. 1998, pp. 1025–1034.

[36] Saeed Daneshmand, Ali Broumandan, John Nielsen, and Gérard Lachapelle. "Interference and multipath mitigation utilising a two-stage beamformer for global navigation satellite systems applications". In: *IET Radar, Sonar & Navigation* 7.1 (2013), pp. 55–66.

[37] Kai Borre, Dennis M Akos, Nicolaj Bertelsen, Peter Rinder, and Søren Holdt Jensen. *A software-defined GPS and Galileo receiver: a single-frequency approach*. Springer Science & Business Media, 2007.

[38] D. Borio, L. Camoriano, and L. Lo Presti. "Impact of GPS acquisition strategy on decision probabilities". In: *IEEE Transactions on Aerospace and Electronic Systems* 44.3 (2008), pp. 996–1011. ISSN: 0018-9251. DOI: 10.1109/TAES.2008.4655359.

[39] Richard DJ Van Nee, Jaap Siereveld, Patrick C Fenton, and Bryan R Townsend. "The multipath estimating delay lock loop: approaching theoretical accuracy limits". In: *Proceedings of 1994 IEEE Position, Location and Navigation Symposium-PLANS'94*. IEEE. 1994, pp. 246–251.

[40] Paul Fine and Warren Wilson. "Tracking algorithm for GPS offset carrier signals". In: *Proceedings of the 1999 national technical meeting of The Institute of Navigation*. 1999, pp. 671–676.

[41] John G Proakis, Masoud Salehi, Ning Zhou, and Xiaofeng Li. *Communication systems engineering*. Vol. 2. Prentice Hall New Jersey, 1994.

[42] AJ Van Dierendonck, Pat Fenton, and Tom Ford. "Theory and performance of narrow correlator spacing in a GPS receiver". In: *Navigation* 39.3 (1992), pp. 265–283.

[43] Bryan Townsend and Patrick Fenton. "A practical approach to the reduction of pseudorange multipath errors in a L1 GPS receiver". In: *Proceedings of the 7th International Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, UT, USA*. Citeseer. 1994, pp. 20–23.

[44] Audrey Giremus, Jean-Yves Tourneret, and Vincent Calmettes. “A particle filtering approach for joint detection/estimation of multipath effects on GPS measurements”. In: *IEEE Transactions on Signal Processing* 55.4 (2007), pp. 1275–1285.

[45] R Grover Brown, BW Parkinson, and JJ Spilker. “Receiver autonomous integrity monitoring”. In: *Global Positioning System: Theory and applications.* 2 (1996), pp. 143–165.

[46] Clyde C Goad. “Single-site GPS models”. In: *GPS for Geodesy.* Springer, 1998, pp. 437–456.

[47] Guochang Xu and Yan Xu. *GPS: Theory, Algorithms and Applications.* Springer, 2007.

[48] Mark Petovello and Cillian O’Driscoll. *Carrier phase and its measurement for GNSS.* 2010.

[49] Peter Teunissen. “Towards a unified theory of GNSS ambiguity resolution”. In: *Journal of Global Positioning Systems* 2.1 (2003), pp. 1–12.

[50] Peter JG Teunissen. “Least-squares estimation of the integer GPS ambiguities”. In: *Invited lecture, section IV theory and methodology, IAG general meeting, Beijing, China.* 1993.

[51] Paul De Jonge, CCJM Tiberius, et al. “The LAMBDA method for integer ambiguity estimation: implementation aspects”. In: *Publications of the Delft Computing Centre, LGR-Series* 12.12 (1996), pp. 1–47.

[52] Zhizhao Liu. “A new automated cycle slip detection and repair method for a single dual-frequency GPS receiver”. In: *Journal of Geodesy* 85.3 (2011), pp. 171–183.

[53] Malek Karaim, Tashfeen B Karamat, Aboelmagd Noureldin, and Ahmed El-Shafie. “GPS cycle slip detection and correction at measurement level”. In: *Current Journal of Applied Science and Technology* (2014), pp. 4239–4251.

[54] Georges Stienne, Serge Reboul, Jean-Bernard Choquel, and Mohammed Benjelloun. “Cycle slip detection and repair with a circular on-line change-point detector”. In: *Signal Process.* 100 (2014), pp. 51–63.

[55] Tomoji Takasu and Akio Yasuda. “Cycle slip detection and fixing by MEMS-IMU/GPS integration for mobile environment RTK-GPS”. In: *Proceedings of the 21st international technical meeting of the satellite division of the Institute of Navigation (ION GNSS 2008).* 2008, pp. 64–71.

[56] PJG Teunissen and Amir Khodabandeh. “Review and principles of PPP-RTK methods”. In: *Journal of Geodesy* 89.3 (2015), pp. 217–240.

[57] Roland E Best. *Phase-locked loops: design, simulation, and applications.* McGraw-Hill Education, 2007.

[58] Henri de Bellescize. *La reception synchrone.* E. Chiron, 1932.

[59] Donald R Stephens. *Phase-locked loops for wireless communications: digital, analog and optical implementations.* Springer Science & Business Media, 2007.

[60] EM Drogin. “Steering on course to safer air travel”. In: *Electronics* (1967), pp. 95–102.

[61] W. C. Lindsey and Chak Ming Chie. "A survey of digital phase-locked loops". In: *Proceedings of the IEEE* 69.4 (1981), pp. 410–431. ISSN: 0018-9219. DOI: 10.1109/PROC.1981.11986.

[62] Jean-Marc Brossier. *Signal et communication numérique: égalisation et synchronisation*. Hermes, 1997.

[63] Umberto Mengali. *Synchronization techniques for digital receivers*. Springer Science & Business Media, 2013.

[64] Pierre-Olivier Amblard, Jean-Marc Brossier, and Eric Moisan. "Phase tracking: what do we gain from optimality? Particle filtering versus phase-locked loops". In: *Signal Processing* 83.1 (2003), pp. 151–167.

[65] Andreas Antoniou. *Digital filters*. McGraw Hill, 1993.

[66] Kendall Atkinson and Weimin Han. *Theoretical numerical analysis*. Vol. 39. Springer, 2005.

[67] John Fitz Randolph. *Calculus and analytic geometry*. Wadsworth Publishing Company, 1965.

[68] Endre Süli and David F Mayers. *An introduction to numerical analysis*. Cambridge university press, 2003.

[69] S. A. Stephens and J. B. Thomas. "Controlled-root formulation for digital phase-locked loops". In: *IEEE Transactions on Aerospace and Electronic Systems* 31.1 (1995), pp. 78–95. ISSN: 0018-9251. DOI: 10.1109/7.366295.

[70] Alan V Oppenheim and Ronald W Schafer. "Digital signal processing". In: *Research supported by the Massachusetts Institute of Technology, Bell Telephone Laboratories, and Guggenheim Foundation. Englewood Cliffs, N. J., Prentice-Hall, Inc., 1975. 598 p* (1975).

[71] Andrew J Viterbi. "Phase-locked loop dynamics in the presence of noise by Fokker-Planck techniques". In: *The Foundations Of The Digital Wireless World: Selected Works of AJ Viterbi*. World Scientific, 2010, pp. 13–29.

[72] William S Levine. *The control handbook: Control system fundamentals*. CRC press, 2010.

[73] Katsuhiko Ogata and Yanjuan Yang. *Modern control engineering*. Vol. 4. Prentice-Hall, 2002.

[74] Thomas J Karras. "Equivalent noise bandwidth analysis from transfer functions". In: (1965).

[75] Eliahu Ibraham Jury. "Theory and Application of the z-Transform Method". In: (1964).

[76] R Winkelstein. "Closed form evaluation of symmetric two-sided complex integrals". In: *TDA Progress Report* 42-65 (1981), pp. 133–141.

[77] Floyd M Gardner. *Phaselock techniques*. John Wiley & Sons, 2005.

[78] George E Uhlenbeck and Leonard S Ornstein. "On the theory of the Brownian motion". In: *Physical review* 36.5 (1930), p. 823.

[79] Ming Chen Wang and George Eugene Uhlenbeck. “On the theory of the Brownian motion II”. In: *Reviews of modern physics* 17.2-3 (1945), p. 323.

[80] Hannes Risken. “Fokker-Planck equation”. In: *The Fokker-Planck Equation*. Springer, 1996, pp. 63–95.

[81] Kanti V Mardia and Peter E Jupp. *Directional statistics*. Vol. 494. John Wiley & Sons, 2009.

[82] Milton Abramowitz and Irene A Stegun. *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. Vol. 55. Courier Corporation, 1965.

[83] George Fishman. *Monte Carlo: concepts, algorithms, and applications*. Springer Science & Business Media, 2013.

[84] A. Doucet and Xiaodong Wang. “Monte Carlo methods for signal processing: a review in the statistical signal processing context”. In: *IEEE Signal Processing Magazine* 22.6 (2005), pp. 152–170. ISSN: 1053-5888. DOI: 10.1109/MSP.2005.1550195.

[85] F. J. Charles and W. C. Lindsey. “Some analytical and experimental phase-locked loop results for low signal-to-noise ratios”. In: *Proceedings of the IEEE* 54.9 (1966), pp. 1152–1166. ISSN: 0018-9219. DOI: 10.1109/PROC.1966.5061.

[86] G. Ascheid and H. Meyr. “Cycle Slips in Phase-Locked Loops: A Tutorial Survey”. In: *IEEE Transactions on Communications* 30.10 (1982), pp. 2228–2241. ISSN: 0090-6778. DOI: 10.1109/TCOM.1982.1095423.

[87] William C Lindsey and Marvin Kenneth Simon. *Telecommunication systems engineering*. Courier Corporation, 1991.

[88] R. Tausworthe. “Cycle Slipping in Phase-Locked Loops”. In: *IEEE Transactions on Communication Technology* 15.3 (1967), pp. 417–421. ISSN: 0018-9332. DOI: 10.1109/TCOM.1967.1089589.

[89] R. Tausworthe. “Simplified Formula for Mean Cycle-Slip Time of Phase-Locked Loops With Steady-State Phase Error”. In: *IEEE Transactions on Communications* 20.3 (1972), pp. 331–337. ISSN: 0090-6778. DOI: 10.1109/TCOM.1972.1091184.

[90] Guenter W Hein, Gerald Baustert, Bernd Eissfeller, and Herbert Landau. “High-Precision Kinematic GPS Differential Positioning and Integration of GPS with a Ring Laser Strapdown Inertial System”. In: *Navigation* 36.1 (1989), pp. 77–98.

[91] Grace Xingxin Gao, Jim Spilker, Todd Walter, Per Enge, and Anthony R Pratt. “Code generation scheme and property analysis of broadcast Galileo L1 and E6 signals”. In: *ION GNSS*. 2006, pp. 1526–1534.

[92] J. A. Lopez-Salcedo, J. A. D. Peral-Rosado, and G. Seco-Granados. “Survey on Robust Carrier Tracking Techniques”. In: *IEEE Communications Surveys Tutorials* 16.2 (2014), pp. 670–688. ISSN: 1553-877X. DOI: 10.1109/SURV.2013.082713.00228.

[93] Someshwar C Gupta. “Phase-locked loops”. In: *Proceedings of the IEEE* 63.2 (1975), pp. 291–306.

[94] Phillip W Ward. "Performance comparisons between FLL, PLL and a novel FLL-assisted-PLL carrier tracking loop under RF interference conditions". In: *Proceedings of the 11th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1998)*. 1998, pp. 783–795.

[95] Pejman L Kazemi. "Optimum digital filters for GNSS tracking loops". In: *Proc. ION GNSS*. Citeseer. 2008, pp. 2304–2313.

[96] S Roche, S Bidon, O Besson, M Monnerat, L Ries, and P Thevenon. "PLL unwrapping structures using polynomial prediction algorithm for noisy carrier phase tracking". In: *Proceedings of the 25th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2012)*. 2012, pp. 47–55.

[97] Benjamin J Slocumb and John Kitchen. "A polynomial phase parameter estimation-phase unwrapping algorithm". In: *Proceedings of ICASSP'94. IEEE International Conference on Acoustics, Speech and Signal Processing*. Vol. 4. IEEE. 1994, pp. IV–129.

[98] Fabrice Legrand, Christophe Macabiau, Jean-Luc Issler, Laurent Lestarquit, and Christian Mehlen. "Improvement of pseudorange measurements accuracy by using fast adaptive bandwidth lock loops". In: *Proceedings of the 13th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2000)*. 2000, pp. 2346–2356.

[99] Faisal A Khan, Andrew G Dempster, and Chris Rizos. "Projected bandwidth loop-an alternative to adaptive bandwidth loops with reduced complexity". In: *Proceedings of IEEE/ION PLANS 2010*. 2010, pp. 1147–1153.

[100] W-L Mao, H-W Tsao, and F-R Chang. "Intelligent GPS receiver for robust carrier phase tracking in kinematic environments". In: *IEE Proceedings-Radar, Sonar and Navigation* 151.3 (2004), pp. 171–180.

[101] Dan Simon and Hossny El-Sherief. "Fuzzy logic for digital phase-locked loop filter design". In: *IEEE Transactions on Fuzzy Systems* 3.2 (1995), pp. 211–218.

[102] Fred Daum. "Nonlinear filters: beyond the Kalman filter". In: *IEEE Aerospace and Electronic Systems Magazine* 20.8 (2005), pp. 57–69.

[103] Tuomas Nyblom, Timo Roman, M Enescu, and V Koivunen. "Time-varying carrier offset tracking in OFDM systems using particle filtering". In: *Proceedings of the Fourth IEEE International Symposium on Signal Processing and Information Technology, 2004.* IEEE. 2004, pp. 217–220.

[104] William Ng, Chunlin Ji, WK Ma, and Hing Cheung So. "A study on particle filters for single-tone frequency tracking". In: *IEEE Transactions on Aerospace and Electronic Systems* 45.3 (2009), pp. 1111–1125.

[105] Mark L Psiaki. "Extended Kalman filter methods for tracking weak GPS signals". In: *Proceedings of the 2002 ION GPS Conf., Portland, Oregon*. 2002.

[106] Robert Grover Brown, Patrick YC Hwang, et al. *Introduction to random signals and applied Kalman filtering*. Vol. 3. Wiley New York, 1992.

[107] Jordi Vilà Valls. “Estimation dynamique non-linéaire de canaux de transmission pour récepteurs satellites mobiles”. PhD thesis. Institut National Polytechnique de Grenoble-INPG, 2010.

[108] X Rong Li and Vesselin P Jilkov. “Survey of maneuvering target tracking. Part V. Multiple-model methods”. In: *IEEE Transactions on Aerospace and Electronic Systems* 41.4 (2005), pp. 1255–1321.

[109] Thiagalingam Kirubarajan and Yaakov Bar-Shalom. “Kalman filter versus IMM estimator: when do we need the latter?” In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (2003), pp. 1452–1457.

[110] Cassio G Lopes, EH Satorius, P Estabrook, and Ali H Sayed. “Adaptive carrier tracking for mars to earth communications during entry, descent, and landing”. In: *IEEE Transactions on Aerospace and Electronic Systems* 46.4 (2010), pp. 1865–1879.

[111] Muhammad Tahir, Letizia Lo Presti, and Maurizio Fantino. “A Novel Quasi-Open Loop Architecture for GNSS Carrier Recovery Systems.” In: *International Journal of Navigation & Observation* (2012).

[112] Frank van Graas, Andrey Soloviev, Maarten Uijt de Haag, and Sanjeev Gunawardena. “Closed-Loop Sequential Signal Processing and Open-Loop Batch Processing Approaches for GNSS Receiver Design”. In: *IEEE Journal of Selected Topics in Signal Processing* 3.4 (2009), pp. 571–586. DOI: 10.1109/JSTSP.2009.2023350.

[113] Bruno De Finetti. *Theory of probability: A critical introductory treatment*. Vol. 6. John Wiley & Sons, 2017.

[114] Leonard J Savage. *The foundations of statistics*. Courier Corporation, 1972.

[115] Václav Peterka. “Bayesian approach to system identification”. In: *Trends and Progress in System identification*. Elsevier, 1981, pp. 239–304.

[116] FRS Bayes. “An essay towards solving a problem in the doctrine of chances”. In: *Biometrika* 45.3-4 (1958), pp. 296–315.

[117] William S Levine. *The control handbook*. CRC press, 1996.

[118] Branko Ristic, Sanjeev Arulampalam, and Neil Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech house, 2003.

[119] Jack Karush. “On the chapman-kolmogorov equation”. In: *The Annals of Mathematical Statistics* 32.4 (1961), pp. 1333–1337.

[120] Vaclav Smidl and Anthony Quinn. “Variational bayesian filtering”. In: *IEEE Transactions on Signal Processing* 56.10 (2008), pp. 5020–5030.

[121] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[122] Rudolph Emil Kalman. “A new approach to linear filtering and prediction problems”. In: *Journal of basic Engineering* 82.1 (1960), pp. 35–45.

[123] YC Ho and RCKA Lee. “A Bayesian approach to problems in stochastic estimation and control”. In: *IEEE transactions on automatic control* 9.4 (1964), pp. 333–339.

[124] Robert Grover Brown and Patrick YC Hwang. *Introduction to random signals and applied Kalman filtering: with MATLAB exercises*. J Wiley & Sons, 2012.

[125] Francesco De Vivo, Alberto Brandl, Manuela Battipede, and Piero Gili. "Joseph covariance formula adaptation to square-root sigma-point kalman filters". In: *Nonlinear Dynamics* 88.3 (2017), pp. 1969–1986.

[126] J. A. Del Peral-Rosado, J. A. López-Salcedo, G. Seco-Granados, J. M. López-Almansa, and J. Cosmen. "Kalman filter-based architecture for robust and high-sensitivity tracking in GNSS receivers". In: *2010 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)*. 2010, pp. 1–8. DOI: 10.1109/NAVITEC.2010.5708005.

[127] JONG-HOON WON, Dominik Dötterböck, and Bernd Eissfeller. "Performance Comparison of Different Forms of Kalman Filter Approaches for a Vector-Based GNSS Signal Tracking Loop". In: *Navigation* 57.3 (2010), pp. 185–199.

[128] A. Patapoutian. "On phase-locked loops and Kalman filters". In: *IEEE Transactions on Communications* 47.5 (1999), pp. 670–672. ISSN: 0090-6778. DOI: 10.1109/26.768758.

[129] D. Polk and S. Gupta. "Quasi-Optimum Digital Phase-Locked Loops". In: *IEEE Transactions on Communications* 21.1 (1973), pp. 75–82. ISSN: 0090-6778. DOI: 10.1109/TCOM.1973.1091531.

[130] Brian DO Anderson and John B Moore. *Optimal filtering*. Courier Corporation, 2012.

[131] Cyrille Gernot. *Development of combined GPS L1/L2C acquisition and tracking methods for weak signals environments*. University of Calgary, Department of Geomatics Engineering, 2009.

[132] Jordi Vila-Valls, Pau Closas, and Carles Fernandez-Prades. "Advanced KF-based methods for GNSS carrier tracking and ionospheric scintillation mitigation". In: *2015 IEEE Aerospace Conference*. IEEE. 2015, pp. 1–10.

[133] J. Vila-Valls, P. Closas, M. Navarro, and C. Fernandez-Prades. "Are PLLs dead? A tutorial on kalman filter-based techniques for digital carrier synchronization". In: *IEEE Aerospace and Electronic Systems Magazine* 32.7 (2017), pp. 28–45. ISSN: 0885-8985. DOI: 10.1109/MAES.2017.150260.

[134] Stephen E Fienberg and Joseph B Kadane. "Statistics: The Field". In: (2001).

[135] Ross Kindermann. "Markov random fields and their applications". In: *American mathematical society* (1980).

[136] J. M. N. Leitao and M. A. T. Figueiredo. "Absolute phase image reconstruction: a stochastic nonlinear filtering approach". In: *IEEE Transactions on Image Processing* 7.6 (1998), pp. 868–882. DOI: 10.1109/83.679433.

[137] J-F Giovannelli, Jérôme Idier, Rédha Boubertakh, and Alain Herment. "Unsupervised frequency tracking beyond the Nyquist frequency using Markov chains". In: *IEEE Transactions on signal processing* 50.12 (2002), pp. 2905–2914.

[138] R. F. Barrett and D. A. Holdsworth. "Frequency tracking using hidden Markov models with amplitude and phase information". In: *IEEE Transactions on Signal Processing* 41.10 (1993), pp. 2965–2976. DOI: 10.1109/78.277803.

[139] Stéphanie Bidon and Sébastien Roche. “Variational Bayes phase tracking for correlated dual-frequency measurements with slow dynamics”. In: *Signal Processing* 113 (2015), pp. 182–194.

[140] Václav Šmídl and Anthony Quinn. *The variational Bayes method in signal processing*. Springer Science & Business Media, 2006.

[141] David JC MacKay. “Probable networks and plausible predictions—a review of practical Bayesian methods for supervised neural networks”. In: *Network: computation in neural systems* 6.3 (1995), pp. 469–505.

[142] James W Miskin. “Ensemble learning for independent component analysis”. In: *in Advances in Independent Component Analysis*. Citeseer. 2000.

[143] Manfred Opper, Ole Winther, et al. “From naive mean field theory to the TAP equations”. In: *Advanced mean field methods: theory and practice* (2001), pp. 7–20.

[144] Giorgio Parisi. *Statistical field theory*. Addison-Wesley, 1988.

[145] Christopher M Bishop. “Variational principal components”. In: (1999).

[146] Hagai Attias. “A variational Bayesian framework for graphical models”. In: *Advances in neural information processing systems* 12.1-2 (2000), pp. 209–215.

[147] Christopher M Bishop, JM Winn, D Spiegelhalter, Christopher Bishop, and John Winn. “VIBES: A variational inference engine for Bayesian networks”. In: (2002).

[148] John Winn, Christopher M Bishop, and Tommi Jaakkola. “Variational message passing.” In: *Journal of Machine Learning Research* 6.4 (2005).

[149] Zoubin Ghahramani, Matthew J Beal, et al. “Variational Inference for Bayesian Mixtures of Factor Analysers.” In: *NIPS*. Vol. 12. 1999, pp. 449–455.

[150] Rudolf Kulhavỳ. *Recursive nonlinear estimation: a geometric approach*. Vol. 216. Springer, 1996.

[151] Jose MN Leitao and José MF Moura. “Acquisition in phase demodulation: application to ranging in radar/sonar systems”. In: *IEEE Transactions on aerospace and Electronic Systems* 31.2 (1995), pp. 581–599.

[152] José MB Dias and José MN Leitao. “InSAR phase unwrapping: a Bayesian approach”. In: *IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No. 01CH37217)*. Vol. 1. IEEE. 2001, pp. 396–400.

[153] Jose MF Moura and Arthur B Baggeroer. “Phase unwrapping of signals propagated under the Arctic ice crust: a statistical approach”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 36.5 (1988), pp. 617–630.

[154] Sébastien Roche. “Méthodes de poursuite de phase pour signaux GNSS multifréquence en environnement dégradé”. PhD thesis. Toulouse, ISAE, 2013.

[155] Gabriel Terejanu, Puneet Singla, Tarunraj Singh, and Peter D Scott. “Adaptive Gaussian sum filter for nonlinear Bayesian estimation”. In: *IEEE Transactions on Automatic Control* 56.9 (2011), pp. 2151–2156.

[156] Daniel Alspach and Harold Sorenson. “Nonlinear Bayesian estimation using Gaussian sum approximations”. In: *IEEE transactions on automatic control* 17.4 (1972), pp. 439–448.

[157] Steven M Kay. *Fundamentals of statistical signal processing*. Prentice Hall PTR, 1993.

[158] Dennis S Bernstein. *Matrix mathematics: theory, facts, and formulas*. Princeton university press, 2009.

[159] R. A. Singer. “Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets”. In: *IEEE Transactions on Aerospace and Electronic Systems* AES-6.4 (1970), pp. 473–483. ISSN: 2371-9877. DOI: 10.1109/TAES.1970.310128.

[160] X Rong Li and Vesselin P Jilkov. “Survey of maneuvering target tracking: dynamic models”. In: *Signal and Data Processing of Small Targets 2000*. Vol. 4048. International Society for Optics and Photonics. 2000, pp. 212–235.

[161] Jong-Hoon Won, Thomas Pany, and GüNTeR W HeiN. “GNSS software defined radio”. In: *Inside GNSS* 1.5 (2006), pp. 48–56.

[162] Walter HW Tuttlebee. *Software defined radio: enabling technologies*. John Wiley & Sons, 2003.

[163] Mark G Petovello, Cillian O’Driscoll, Gérard Lachapelle, Daniele Borio, and Hasan Murtaza. “Architecture and benefits of an advanced GNSS software receiver”. In: *Journal of Global Positioning Systems* 7.2 (2008), pp. 156–168.

[164] Fabio Principe, Giacomo Bacci, Filippo Giannetti, and Marco Luise. “Software-defined radio technologies for GNSS receivers: A tutorial approach to a simple design and implementation”. In: *International Journal of Navigation and Observation* 2011 (2011).

[165] Benoit Priot, Arnaud Dion, Guillaume Beaugendre, and Raghuveer Kasaraneni. “Accurate Events Synchronization in a System-on-Chip Navigation Receiver”. In: *International Conference on Localization and GNSS*. 2019, pp. 1–5.

[166] Edward A Lee and David G Messerschmitt. *Digital communication*. Springer Science & Business Media, 2012.

[167] M. S. Sharawi, D. M. Akos, and D. N. Aloi. “GPS C/N_0 estimation in the presence of interference and limited quantization levels”. In: *IEEE Transactions on Aerospace and Electronic Systems* 43.1 (2007), pp. 227–238. ISSN: 2371-9877. DOI: 10.1109/TAES.2007.357129.

[168] Mark L Psiaki, Dennis M Akos, and Jonas Thor. “A comparison of direct RF sampling and downconvert & sampling GNSS receiver architectures”. In: *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)*. 2003, pp. 1941–1952.

[169] Mark L Psiaki. “Block acquisition of weak GPS signals in a software receiver”. In: *Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2001)*. 2001, pp. 2838–2850.

[170] Emanuela Falletti, Marco Pini, and Letizia Lo Presti. “Low complexity carrier-to-noise ratio estimators for GNSS digital receivers”. In: *IEEE transactions on aerospace and electronic systems* 47.1 (2011), pp. 420–437.

[171] Huaqiang Shu, Eric Pierre Simon, and Laurent Ros. “Third-order kalman filter: Tuning and steady-state performance”. In: *IEEE Signal Processing Letters* 20.11 (2013), pp. 1082–1085.

[172] Huaqiang Shu, Laurent Ros, and Eric Pierre Simon. “Third-order complex amplitudes tracking loop for slow flat fading channel online estimation”. In: *IET Communications* 8.3 (2014), pp. 360–371.

[173] Sébastien Roche and Stéphanie Bidon. “On the comparison between KF-PLL and DPLL”. unpublished. 2021.

[174] P. Tichavsky, C.H. Muravchik, and A. Nehorai. “Posterior Cramer-Rao bounds for discrete-time nonlinear filtering”. In: *IEEE Transactions on Signal Processing* 46.5 (1998), pp. 1386–1396. DOI: 10.1109/78.668800.

[175] Carsten Fritsche, Emre Özkan, Lennart Svensson, and Fredrik Gustafsson. “A fresh look at Bayesian Cramér-Rao bounds for discrete-time nonlinear filtering”. In: *17th International Conference on Information Fusion (FUSION)*. IEEE. 2014, pp. 1–8.

[176] F. D. Nunes and J. M. N. Leitao. “A nonlinear filtering approach to estimation and detection in mobile communications”. In: *IEEE Journal on Selected Areas in Communications* 16.9 (1998), pp. 1649–1659. ISSN: 0733-8716. DOI: 10.1109/49.737634.

[177] KB Petersen, MS Pedersen, et al. “The Matrix Cookbook”. In: *Technical University of Denmark* 15 (2005).

Résumé — Dans cette thèse de doctorat, nous nous intéressons à l'estimation robuste de la phase de la porteuse en utilisant le filtrage bayésien variationnel. La mesure de la phase porteuse est devenue une tâche fondamentale dans de nombreuses applications d'ingénierie telles que le positionnement précis dans le domaine du GNSS. Malheureusement, les mesures de phase obtenues par les techniques traditionnelles de poursuite de phase peuvent être fortement mises à mal par la présence de sauts de phase ambigus, appelés sauts de cycle. Ces derniers peuvent durement impacter les performances des algorithmes de poursuite conduisant, dans le pire des cas, à une perte permanente de verrouillage du signal. Par conséquent, pour résoudre ce problème, nous proposons un filtre non linéaire robuste de poursuite de phase basé sur l'inférence de Bayes variationnelle. Tout d'abord, l'algorithme est développé en supposant une dynamique lente de phase (c'est-à-dire la boucle au premier ordre), puis, son ordre est augmenté en estimant un vecteur d'état formé par la phase et ses dérivées. Les performances de ce nouvel algorithme sont comparées avec celles de techniques conventionnelles en terms de précision d'estimation et d'occurence de sauts de cycle. La comparaison est évaluée à l'aide de données GNSS synthétiques et réelles. Les résultats montrent que la méthode proposée a de meilleures performances par rapport aux filtres linéaires conventionnels.

Mots clés : Poursuite de phase, sauts de cycle, filtre de Kalman, filtrage bayésien, inférence bayésienne variationnelle, récepteur radio logicielle GNSS, estimation robuste.

Abstract — In this Ph.D. thesis, we are interested in robust carrier-phase estimation by using Variational Bayesian filtering. Carrier-phase measurement has become a fundamental task in many various engineering applications such as precise point positioning in GNSS (Global Navigation Satellite System). Unfortunately, phase measurements obtained by traditional phase tracking techniques may be strongly affected by the presence of ambiguous phase jumps, known as cycle slips. The latter may strongly impact the performance of the considered phase tracking algorithm leading to, in the worst case, a permanent loss of lock of the signal. Therefore, to address this problem, we propose a robust nonlinear filter for carrier-phase tracking based on Restricted Variational Bayes inference. First, the algorithm is developed only for slow phase dynamics (*i.e.*, first-order loop), then, its order is augmented by estimating a state vector formed by the carrier-phase and its derivatives. The performance of the proposed algorithm is compared with that of conventional techniques (DPLL (Digital Phase Lock Loop) and KF (Kalman Filter)-based DPLL) in terms of precision of estimation and cycle slipping occurrence. The comparison is conducted using synthetic and real GNSS data. Results show that the proposed method outperforms the conventional linear filters.

Keywords: Phase tracking, cycle slips, Phase Lock Loop, Kalman filter, Bayesian filtering, Variational Baysian inference, GNSS software-defined receiver, robust estimation.