



HAL
open science

Algorithmes d'hybridation pour du positionnement précis GNSS

Isambert Delphine

► **To cite this version:**

Isambert Delphine. Algorithmes d'hybridation pour du positionnement précis GNSS. Sciences de l'information et de la communication. Institut polytechnique de Paris; Telecom SudParis, 2024. Français. NNT : 2024IPPAS001 . tel-04919960

HAL Id: tel-04919960

<https://hal.science/tel-04919960v1>

Submitted on 29 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Algorithmes d'hybridation pour du positionnement précis GNSS

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Telecom SudParis



ECOLE
DOCTORALE

Ecole Doctorale de l'Institut Polytechnique de Paris
n°626

Spécialité de doctorat: Sciences de l'information et de
la communication

Thèse présentée et soutenue à Palaiseau, le 22/02/2024, par

Delphine Isambert

Composition du Jury :

Juliette MARAIS Directrice de recherche, Université Gustave Eiffel	Présidente
Laurent MOREL Professeur des universités, CNAM ESGT	Rapporteur
Bertrand MERMINOD Professeur, EPFL	Rapporteur
Nel Samama Professeur, Télécom SudParis	Directeur de thèse
Alexandre Vervisch-picois Maître de conférences, Telecom SudParis	Co-encadrant de thèse
Paul Chambon Directeur Technique Exagone Teria	Invité

Remerciements

Tout d'abord, je tiens à remercier M. Nel Samama, pour m'avoir proposé ce sujet de thèse il y a trois ans au moment où je me posais beaucoup de questions sur quoi faire après mon diplôme d'ingénieur. Merci à lui pour son accompagnement en tant que directeur de thèse et sa bienveillance.

Ensuite, je remercie M. Alexandre Vervisch-Picois et M. Paul Chambon, pour l'encadrement de qualité qu'ils m'ont offert pendant ces trois années. Ils ont su être là pour m'aider à comprendre ce nouveau monde du positionnement GNSS et répondre à mes nombreuses questions. Vos conseils et votre bienveillance ont été des éléments clé dans la bonne réussite de ces trois années de travail. Un merci particulier à Alexandre, pour ses précieuses relectures, remises en question mais aussi les nombreuses relances pour finaliser mes travaux dans les temps.

J'adresse également mes remerciements à M. Laurent Morel, M. Bertrand Merminod et Mme. Juliette Marais d'avoir accepté de faire partie du jury de ma thèse.

Je tiens à remercier par ailleurs l'ensemble de l'équipe Exagone Teria. Merci pour votre accueil chaleureux, votre soutien, vos conseils et nos moments de partage qui ont été d'une grande aide pour moi. Je remercie particulièrement Rakul et Joliane pour leurs encouragements durant les dernières semaines, qui ont toujours su me faire rire pour m'aider à avancer.

Enfin, je remercie mes proches de m'avoir toujours soutenue durant ces trois années. A mes frères et sœurs qui, malgré leur bienveillance et leur énorme soutien, n'ont, je crois, toujours pas tout compris sur quoi portait le sujet de ma thèse et de mes recherches. Merci à vous pour vos encouragements et votre bonne humeur toujours aussi motivante. Merci à ma maman qui m'a toujours accompagnée, soutenue et aidée pour poursuivre mes études jusqu'au bout. Elle ne devait probablement pas se douter au début que cela durerait aussi longtemps. Je crois pouvoir te dire qu'on se rapproche de la fin. Enfin, merci à Victor, pour son aide et son soutien qui a su m'encourager à tout moment même quand j'avais envie de baisser les bras. Ses nombreux encouragements me permettent de terminer ces 10 années d'études.

Table des matières

Introduction.....	6
1. Le positionnement précis GNSS : état de l’art des deux principales méthodes et leurs modèles de représentation.....	10
1.1. Techniques de positionnement GNSS.....	10
1.1.1 Méthode de différentiation de la phase : RTK.....	10
1.1.2 Généralisation du RTK : le NRTK.....	19
1.1.3 PPP – Precise Point Positioning.....	22
1.2 Système de représentation.....	26
1.2.1 OSR.....	27
1.2.2 SSR.....	28
2. Une extension du PPP : l’hybridation PPP-RTK, état de l’art d’un exemple de solution hybride.....	31
2.1 Paramétrage du PPP-RTK – la construction d’une solution hybride.....	31
2.1.1 Partie Réseau : première étape de l’hybridation.....	33
2.1.2 Partie Utilisateur : Indépendance des paramètres et calcul d’une position en PPP-RTK.....	40
2.2 Les critères de performances : une variabilité de solutions PPP-RTK.....	41
2.2.1 Time-to-first-fix (TTFF) en PPP-RTK.....	42
2.2.2 Précision en PPP-RTK.....	44
3. Une hybridation PPP-RTK et NRTK – Construction de ma proposition de solution hybride.....	49
3.1 Partie réseau : un réseau CORS pour un système d’équations d’observations GNSS.....	49
3.1.1 Système complet d’équations du réseau.....	49
3.1.2 Base S – Construction à partir du défaut de rang.....	54
3.1.3 Ambiguïtés du réseau.....	59
3.1.4 Corrections PPP-RTK.....	60
3.1.5 Mise en œuvre algorithmique de la fonction réseau.....	60
3.2 Construction d’une nouvelle base S avec le NRTK.....	61
3.2.1 Initialisation avec le calcul NRTK à t_0	62
3.2.2 Une nouvelle base S' – Modèle de corrections PPP-NRTK.....	63
3.2.3 Conception algorithmique dans l’hybridation PPP-RTK.....	63
3.3 Calcul de la position – partie utilisateur.....	64
3.3.1 Corrections PPP-NRTK reçues.....	64
3.3.2 Calcul de la distance satellites – récepteurs.....	65
3.3.2 Variation de la position.....	66
3.3.2 Calcul de la position et premiers résultats.....	67
4. Mise en place de la solution et tests de ma méthode PPP-NRTK.....	69

4.1 Construction et mise en place	70
4.1.1 Génération des corrections PPP-NRTK avec un réseau de stations de référence.....	70
4.1.2 Initialisation et calcul d'une première position.....	74
4.1.3 Résolution et calcul de la position – Hypothèses d'observations	75
4.2 Tests de la solution	76
4.2.1 Calcul de la position selon différents scenarios	76
4.2.2 Critères de performance PPP-NRTK – Résultats de performance.....	81
4.2.3 Impact du réseau CORS sur le calcul de la position	97
4.3 Applications de la technique – Mise en œuvre industrielle possible	100
4.4 Conférences, mise en valeur des travaux dans le monde de la recherche.....	101
Conclusion	102
Annexe 1 -Bibliographie	105
Annexe 2 -Table des figures.....	108
Annexe 3 - Glossaire.....	110
Annexe 4 - Corrections PPP-RTK Biais Code.....	111
Annexe 5 - Corrections PPP-RTK Biais Phase	112

Introduction

Les objets connectés sont aujourd'hui de plus en plus présents dans notre quotidien avec l'apparition de nouvelles technologies. Le positionnement précis vient prendre une grande place dans cette nouvelle dynamique de la smart city avec par exemple : la voiture autonome, les drones, les robots connectés utilisés dans l'agriculture ou encore dans le monde industriel. Ainsi, les techniques de positionnement précis évoluent pour répondre à cette nouvelle demande : les produits sont de plus en plus sophistiqués avec des plus grandes capacités en gestion de données comme les drones (Ekaso, et al., 2020). La principale méthode utilisée est le positionnement GNSS (Global Navigation Satellite Systems) qui utilise des données issues des constellations satellitaires. Ces dernières se multiplient avec le développement dernièrement des constellations indiennes IRNSS (Indian Regional Navigation Satellite System) et japonaises QZSS (Quasi-Zenith Satellite System) qui devraient avoir trois satellites supplémentaires d'ici la fin de l'année. Historiquement, c'est la constellation américaine GPS (Global Positioning System) mise en place entre les années 1970 et 1990 qui sera pionnière dans la géolocalisation : c'est avec ces satellites que les premiers modèles de positionnement vont être mis en place au milieu des années 1990. L'utilisation des données satellitaires a un gros avantage en termes de zone de couverture, un satellite pouvant diffuser des signaux sur une large zone. En ayant des constellations composées de dizaines de satellites, ces données sont disponibles quasi en continu depuis n'importe quelle position sur la Terre. De plus, la synchronisation entre les satellites apporte un deuxième avantage : le temps GPS devient une donnée essentielle pour synchroniser des appareils connectés et permet d'avoir une référence précise dans les calculs. La principale contrainte est d'être « visible » par le satellite : l'utilisation des données satellitaires est ainsi possible seulement depuis l'extérieur et dans une zone assez dégagée. Par exemple, un récepteur situé dans un canyon urbain, que l'on peut définir comme une zone avec des bâtiments très hauts, rencontre des problèmes de réception des signaux satellitaires liés aux multi-trajets, au masquage, à la géométrie, et d'autres phénomènes physiques comme la diffraction.

Pour élaborer des techniques de positionnement GNSS, les signaux des satellites sont utilisés comme base de résolution. Ainsi, les algorithmes de positionnement GNSS reposent sur le même principe consistant à résoudre les équations d'observations GNSS des signaux en provenance des satellites pour un récepteur. Ces équations mettent en évidence les sources d'erreurs sous leurs formes simplifiées. Ainsi, pour un récepteur r , un satellite s à une fréquence j , on peut écrire :

$$\begin{cases} R_{r,j}^s = \rho_r^s + (dt_r - dt^s) + \tau_r^s + \mu_j \Delta_r^s - \lambda_j d_{r,j}^s \\ \varphi_{r,j}^s = \rho_r^s + (dt_r - dt^s) + \tau_r^s - \mu_j \Delta_r^s - \lambda_j \delta_{r,j}^s + \lambda_j N_{r,j}^s \end{cases} \quad (1) \quad \mu_j = \frac{f_{j1}^2}{f_j^2}$$

$R_{r,j}^s$ - Mesure de la phase du code du signal du satellite s et à la fréquence j

$\varphi_{r,j}^s$ - Mesure de la phase de la porteuse du signal du satellite s et à la fréquence j

ρ_r^s - Distance entre l'antenne du récepteur et l'antenne émettrice du satellite

dt_r - Erreur d'horloge du récepteur

dt^s - Erreur d'horloge du satellite

τ_r^s - Erreur troposphérique

Δ_r^s - Erreur ionosphérique

- $d_{,j}^s$ - Biais de code du satellite
- $\delta_{,j}^s$ - Biais de phase du satellite
- λ_j - Longueur d'onde
- $N_{r,j}^s$ - Ambiguïté du signal

Ce système d'équations représente la décomposition des mesures du code et de la phase de la porteuse du signal d'un satellite reçu par un récepteur. On appelle ces mesures des « pseudodistances » car elles contiennent, en plus de la distance géométrique entre les antennes du satellite et du récepteur, principalement des termes d'horloge. L'objectif du récepteur GNSS est de calculer les coordonnées de la position à partir de ces mesures. Lorsque l'on se penche sur ces équations, on remarque que ρ_r^s est liée à la distance entre le récepteur et le satellite. C'est donc celle-ci qui nous permettra de calculer la position du récepteur. En effet, en connaissant la position des satellites grâce aux éphémérides fournies par le message de navigation du signal GNSS (ou d'une autre source), on peut relier la distance géométrique entre satellite s de coordonnées x_s, y_s, z_s et récepteur r de coordonnées inconnues x_r, y_r, z_r , par une équation simple :

$$\rho_r^s = \sqrt{(x_s - x_r)^2 + (y_s - y_r)^2 + (z_s - z_r)^2}$$

Ainsi, pour calculer la position du récepteur r , si on reprend les équations d'observation précédentes, la principale inconnue à déterminer est ρ_r^s . Pour cela, il est possible de combiner ces équations utilisant plusieurs signaux de satellites ou des sources de données extérieures. Il est aussi possible d'utiliser ces signaux à différentes fréquences : par exemple les satellites de la constellation Galileo diffusent des signaux sur les fréquences E5 (autour de 1.91795 GHz) et E1 (L1 à 1.57542 GHz élargie). Ainsi, les multi-constellations et les multifréquences deviennent deux options pour combiner de manière plus large les équations d'observations GNSS. Cependant, le positionnement précis GNSS veut s'adapter de plus en plus à certains usages du quotidien et atteindre le marché de masse. Pour cela, des techniques de positionnement monofréquence peuvent être choisies pour s'adapter plus facilement aux technologies communément en usage. Le téléphone portable par exemple, pour lequel l'utilisation de plusieurs fréquences entraînerait une modification importante de l'architecture du téléphone (antenne et tête RF) et augmenterait également sa consommation énergétique (Ruwisch, et al., 2020).

La problématique du positionnement précis qui va nous intéresser dans ce travail de thèse correspond à une façon particulière de traiter la question du positionnement GNSS. En effet, en reprenant une fois encore les équations d'observation GNSS présentées ci-dessus, on s'aperçoit que les mesures réalisées contiennent en plus de la distance géométrique, de multiples erreurs provenant de différents phénomènes physiques comme les traversées de l'ionosphère et de la troposphère, la dérive des horloges liée à la qualité des oscillateurs, etc. On explicitera plus loin les différentes influences de ces erreurs, mais il semble assez évident que pour passer d'un positionnement d'une précision de quelques mètres, qui est celui des GNSS autonome, à quelques centimètres, il faut réduire à leurs minima les influences de ces sources d'erreur. Cela conduit à développer des modes opératoires et des algorithmes de positionnement qui seront spécifiques au positionnement précis. Nous avons, au cours de ce travail de thèse, travaillé principalement sur de tels algorithmes et leur évaluation opérationnelle.

Pour cela, différents paramètres de performance peuvent être utilisés (Feng, et al., 2007). Dans notre étude, nous utiliserons principalement deux de ces critères pour évaluer et comparer les

algorithmes de positionnement GNSS : la précision de la position calculée et le temps de convergence pour une solution valable. La précision signifie l'écart d'erreur maximale autorisé autour de la position calculée. Dans le cadre du positionnement précis de notre étude de cas, la précision attendue est inférieure à 1 mètre, c'est-à-dire qu'on va s'intéresser à des écarts de quelques décimètres par rapport à la position attendue. Cette exigence peut devenir plus stricte avec certaines applications et ainsi exiger une précision de quelques centimètres. Le temps de convergence est un autre critère important à ne pas négliger. Il correspond au temps pour avoir la position avec la précision attendue. Ce temps peut varier selon l'algorithme utilisé et selon la précision attendue (quelques centimètres seront en général plus longs à obtenir que quelques décimètres). Toutefois, en pratique, ce temps correspond au temps nécessaire pour estimer les différentes erreurs présentes dans les équations d'observations (1) comme les biais, les erreurs d'horloge ou les erreurs atmosphériques. La principale difficulté dans ce processus est la détermination des ambiguïtés sur les mesures de phase de la porteuse $N_{r,j}^s$. En effet, nous verrons que calculer rapidement et précisément une position se ramène à déterminer exactement et au plus vite les ambiguïtés associées aux signaux.

Ainsi, toute approche ayant pour but de calculer précisément la position doit prendre en considération la détermination des ambiguïtés sur la phase de la porteuse ce qui est le cas des deux principales méthodes utilisées dans le cadre du positionnement GNSS, à savoir le RTK (Real Time Kinematic) et le PPP (Precise Point Positioning). Historiquement, c'est dans les années 1990, suite à l'apparition du GPS que le RTK est mis en place. Il s'agit d'une technique de positionnement différentielle qui fonctionne pour des récepteurs situés à quelques kilomètres autour d'une station de base. On a donc affaire à une technique qui s'applique à une échelle locale. En cela, elle s'oppose à la méthode PPP qui est une technique de positionnement GNSS plus globale : le PPP diffuse des données de corrections à une échelle mondiale, sans interaction avec les récepteurs qui l'utilisent. Contrairement au RTK, elle n'est pas une technique de différentielle.

Les algorithmes RTK et PPP ont des utilisations bien distinctes en pratique dû à leurs résultats différents en termes de performance sur les critères évoqués plus haut, précision et temps de convergence. Ainsi, ces deux méthodes combinent les équations d'observations GNSS pour les satellites visibles, et estiment une position. Des variantes de ces méthodes existent en multipliant le nombre de satellites, le nombre de fréquences ou encore en utilisant d'autres sources de données extérieures (capteurs, récepteurs GNSS multiples, ...). Avec des objectifs d'applications de plus en plus variés, il est important que ces algorithmes de positionnement y soient adaptés. C'est pourquoi l'hybridation des techniques de positionnement GNSS ou de variantes de celles-ci peut être un réel atout. En effet, elle permet à la fois d'espérer garder les principaux avantages des techniques déjà existantes et d'en réduire certains inconvénients. Par exemple, avoir un temps de convergence rapide comme le RTK et une application globale du PPP. La méthode hybride de positionnement GNSS la plus connue est le PPP-RTK. La principale idée de mes recherches est de réfléchir aux différentes options de positionnements hybrides qui pourraient exister et d'en proposer une nouvelle. L'hybridation de techniques peut se faire à différentes échelles : au niveau du type de connectivité, de la façon de calculer certains paramètres des équations d'observation GNSS, ou encore du type de sources de données externes utilisées. Au niveau de la connectivité, cela peut se faire en fonction du système de communication utilisé pour recevoir les données : 4G, les connectivités GNSS (NTRIP) ou encore le Bluetooth. Aussi, l'hybridation PPP-RTK par exemple, peut permettre de comprendre l'apport d'une technique RTK par rapport à l'autre qui relèverait du PPP et ainsi de permettre de changer d'échelle d'applications : d'une méthode locale à une méthode plus globale, voire mondiale. Dans tous ces cas d'hybridation d'algorithmes, notre objectif sera d'évaluer l'approche proposée à travers les deux critères précités : précision et temps de convergence. Nous

garderons en référence, une précision de quelques centimètres et un temps de convergence de quelques minutes qui correspondent, nous le verrons plus loin, à celles des techniques RTK actuelles.

A partir des considérations précédentes, mêlées aux compétences développées au sein de la société Exagone-Teria, j'ai réfléchi à une nouvelle approche en termes d'hybridation d'algorithmes de positionnement précis GNSS. L'idée est de combiner un algorithme PPP-RTK et NRTK (Network RTK) pour avoir une position calculée en continu même en cas de coupure des données fournies à l'utilisateur. L'objectif est de garder une position précise même avec une méthode de calcul globale comme le PPP ou le PPP-RTK. L'idée serait une initialisation du calcul en NRTK puis de continuer de calculer la position en PPP-RTK. On appellera cette technique hybride PPP-NRTK. L'initialisation NRTK permettrait notamment d'accélérer le processus pour la résolution des ambiguïtés et améliorer les performances du PPP-RTK sur ce point. Pour en résumer l'esprit, il s'agit de sacrifier un peu de la dimension globale du PPP via un recours au NRTK limité dans le temps. On résout ainsi plus aisément le problème du temps de convergence. Il reste ensuite à maintenir la qualité de la précision de positionnement avec le PPP de façon autonome (sans plus de dépendance au NRTK).

Je commencerai par présenter un premier état de l'art sur les deux principales techniques de positionnement GNSS : RTK et PPP. L'objectif dans cette partie est de montrer les avantages et les inconvénients des différents algorithmes et de leurs variantes en vue d'une hybridation. Une deuxième partie sera consacrée à un état de l'art de l'hybridation de ces deux techniques que l'on nomme PPP-RTK. On présentera les différentes façons de mettre en place l'hybridation, qui permettent notamment d'estimer certains paramètres des équations d'observations GNSS plus rapidement. Puis, dans une troisième partie, je construirai ma solution d'hybridation PPP-NRTK d'un point de vue mathématique à l'aide des équations d'observations GNSS et sa mise en œuvre algorithmique. Ici, on verra comment l'hybridation NRTK et PPP-RTK est mise en place et la façon dont les deux techniques sont reliées. Enfin, dans une quatrième partie, je présenterai la manière dont j'ai mis en place d'un point de vue pratique ma technique PPP-NRTK qui permet entre autre de pallier des soucis de connectivité. Ensuite, j'analyserai les performances de la méthode d'hybridation proposée sur les bases d'une étude à l'aide des critères de performances comme le temps de convergence et la précision. Toutefois, il sera intéressant de considérer un nouveau critère à ce moment-là : la continuité. Puis, dans une dernière partie, nous conclurons notre propos par une analyse comparative pour montrer l'intérêt d'une telle technique dans l'écosystème industriel GNSS actuel. Cette analyse sera aussi nuancée par les limites de l'hybridation PPP-NRTK.

1. Le positionnement précis GNSS : état de l'art des deux principales méthodes et leurs modèles de représentation

L'objectif de cette partie, est de présenter les deux techniques de positionnement GNSS les plus communément utilisées pour des applications demandant une précision centimétrique. Ainsi, je vais dresser un état de l'art de ces deux méthodes pour démontrer les principaux avantages et inconvénients de chacune.

1.1. Techniques de positionnement GNSS

Lors de l'étude de ces techniques, nous conserverons les deux principaux paramètres de validation que nous avons présentés dans l'introduction : la précision et le temps de convergence. Rappelons en les définitions : la précision correspond à l'écart, en distance ou par coordonnée, entre la position calculée et la position réelle. Dans notre cas, on s'intéressera à des précisions meilleures que cinquante centimètres. Le temps de convergence correspond au temps qu'il faut pour calculer le premier point ayant la précision requise (ou pour recalculer le point en cas d'interruption du positionnement). Un temps de convergence considéré comme raisonnable sera de l'ordre de moins de cinq minutes.

1.1.1 Méthode de différentiation de la phase : RTK

La chronologie de l'apparition des méthodes de positionnement précis me contraint quelque peu dans le choix de la première que je vais présenter. Cette absence réelle de choix a néanmoins un avantage : être apparue la première lui confère également la qualité d'avoir connu les développements les plus importants. Je vais ainsi présenter et expliciter la technique de positionnement précis GNSS la plus utilisée : Real Time Kinematic – RTK. Comme nous l'évoquons en introduction, le RTK repose sur un principe de différenciation sur la mesure de phase de la porteuse du signal. En effet, dans un signal d'un satellite la mesure de phase est plus précise que celle du code (Feng, et al., 2007). En dehors de toute autre source d'erreur, comme les trajets indirects, le bruit sur la mesure de phase introduit une erreur de l'ordre de quelques millimètres dans l'évaluation de la pseudodistance, à comparer aux quelques mètres de la mesure de code. On observe également que la mesure instantanée de la phase de la porteuse est moins sensible aux trajets indirects. Ainsi, dans le cadre du positionnement précis GNSS c'est cette valeur qui pourra nous aider à rester proche de précision de l'ordre du centimètre dans les calculs. En utilisant l'équation sur le code, qui est le calcul classique du mode de positionnement autonome des GNSS, la précision du calcul sera de l'ordre de quelques mètres (Wei, et al., 2006). C'est pourquoi parmi les équations d'observation GNSS, l'équation de la phase nous intéressera tout particulièrement dans cette section consacrée au RTK. Toutefois, l'équation de code reste essentielle pour la mise en œuvre et le calcul d'une première position approximative, qui sera alors utile en NRTK notamment mais aussi dans la résolution des ambiguïtés.

Pour construire la différenciation RTK, on utilise deux récepteurs (r et u) et deux satellites (s et p) émettant à une fréquence j. Nous obtenons l'équation d'observation GNSS sur la phase, double différenciée :

$$\varphi_{r,j}^s - \varphi_{u,j}^s + \varphi_{u,j}^p - \varphi_{r,j}^p = \rho_r^s - \rho_u^s + \rho_u^p - \rho_r^p + \lambda_j N_{r,j}^s - \lambda_j N_{u,j}^s + \lambda_j N_{u,j}^p - \lambda_j N_{r,j}^p \quad (1.1)$$

L'équation (1.1) peut être réécrite : $\Delta\varphi_{ru,j}^{sp} = \Delta\rho_{ru}^{sp} + \lambda_j N_{ru,j}^{sp}$ (1.2)

Avec $\Delta X_{ru,j}^{sp} = (X_{r,j}^s - X_{u,j}^s) - (X_{r,j}^p - X_{u,j}^p)$

$\varphi_{r,j}^s$ - Phase du signal du satellite s et à la fréquence j pour le récepteur r

ρ_r^s - Distance récepteur-satellite

λ_j - Longueur d'onde à la fréquence j

$N_{r,j}^s$ - Ambiguïté du signal

L'équation (1.2) est composée de trois parties bien distinctes : à gauche, on trouve le terme qui correspond aux mesures de phase différenciées. Dans le premier terme à droite, nous retrouvons la contribution géométrique (ρ) différenciée qui nous permet de calculer les coordonnées du point. Dans le second terme à droite, nous avons l'ambiguïté différenciée également qu'il faut déterminer. En résumé, on trouve à gauche les mesures et à droite, deux groupes d'inconnues à déterminer. La première est l'objectif de la résolution puisqu'elle est directement liée à la position géométrique. La seconde, l'ambiguïté double différenciée est une inconnue qu'on doit déterminer pour chaque satellite. Dans le cas du RTK, l'ambiguïté N est donc la principale inconnue : l'ambiguïté de la mesure de phase soit dans notre cas la double ambiguïté différenciée (qu'on appellera l'ambiguïté pour simplifier le propos). Dans le cas du RTK, $N_{ru,j}^{sp}$ est un entier ce qui est un réel avantage. En effet, dans la résolution des équations d'observations GNSS, l'ambiguïté est un nombre flottant qui n'a pas de raison particulière d'être entier. Dans le cas du RTK et de la double différenciation, les ambiguïtés sont entières. La résolution en est ainsi facilitée, nous expliquerons pourquoi dans la partie sur la résolution des ambiguïtés. Mais ce n'est pas le seul avantage du modèle de différenciation : l'élimination des erreurs atmosphériques, des horloges et de biais des matériels des récepteurs et des satellites permet de n'avoir plus aucune dépendance aux erreurs directement liées aux satellites et aux récepteurs. Ceci explique pourquoi depuis de nombreuses années le RTK est essentiellement utilisé dans les applications de positionnement précis du GNSS.

- **Paramètres et critères de performances généraux**

Pour caractériser une solution RTK, des paramètres et des critères de performance existent comme ceux présentés dans la Figure 1 (Feng , et al., 2008). On distinguera les paramètres qui vont influencer la solution RTK, liés à la station de base de référence, des critères de performances qui eux sont liés à l'utilisateur. Cette distinction est importante car, dans la généralisation du RTK par la suite, certains des paramètres seront plus faciles à estimer dans le cadre de la station de base et dans le cadre de l'utilisateur.

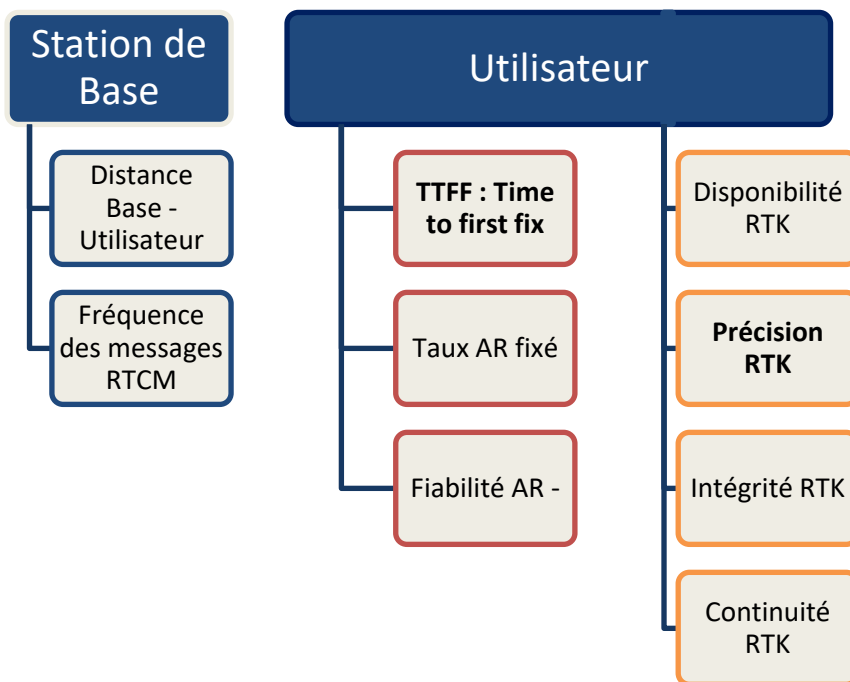


Figure 1 - Paramètres et critères de performance RTK

Explications d'abord les critères définissant les performances :

- TTFF – Time-To-First-Fix : le temps de la première solution calculée valable « fixée » après la détermination des ambiguïtés, et peut correspondre dans une certaine mesure¹ au temps de convergence évoqué dans l'introduction.
- AR (Ambiguity resolution) Fixed Rate, Disponibilité AR : le taux de résolution de l'ambiguïté fixé en pourcentage.
- Fiabilité AR : le pourcentage du total des nombres entiers de doubles différences correctement déterminés (fixés) par rapport au nombre total de nombres entiers de doubles différences fixés.
- Disponibilité RTK : le pourcentage de temps pendant lequel les solutions RTK sont disponibles avec une certaine précision en utilisant les mesures de phase à ambiguïtés fixes (entières et déterminées) et/ou à ambiguïtés flottantes (qui sont définies mais en nombres flottants non entiers).
- Précision RTK : le degré de conformité d'une position RTK estimée à un moment donné à une valeur temporelle de référence définie, obtenue à partir d'une approche indépendante.
- Intégrité RTK : le niveau de confidentialité qui peut être accordé aux informations fournies par le système RTK.
- Continuité RTK : il s'agit de la disponibilité du système RTK sur une certaine période opérationnelle et dans certaines conditions.

Ces critères de performance sont utilisés pour évaluer les algorithmes RTK. On peut donc les qualifier de critères de performance. Les encadrés rouges de la figure 1 sont les critères liés à la résolution des ambiguïtés (AR – Ambiguity Resolution). Comme explicité précédemment c'est grâce aux valeurs entières de celles-ci dans les équations différenciées qu'on pourra facilement les estimer. Nous reviendrons par la suite sur les différents procédés de l'AR. Les encadrés orange correspondent eux directement à des performances de la résolution des algorithmes RTK. Au début de ce manuscrit, nous avons introduit deux critères de performance : la précision et le temps de convergence. Ces deux paramètres correspondent dans notre Figure 1 au TTFF et à la précision RTK. Selon nous, ces

¹ Le TTFF est plus restrictif dans sa définition que le temps de convergence.

deux critères sont les plus importants. Les autres critères nous informent en quelque sorte sur les étapes intermédiaires qui vont avoir des conséquences sur ces deux-là. On pourrait et les autres paramètres vont avoir des conséquences sur ceux-là. On pourrait les qualifier de critères terminaux. C'est ce qui justifie à nos yeux que dans la comparaison de nos algorithmes de positionnement précis GNSS, nous nous bornerons à ces deux paramètres que sont le **TTFF** (assimilé au temps de convergence) et la **précision RTK**.

Les paramètres influençant les performances sont :

- Distance base – utilisateur : la distance en kilomètres entre la station de base et l'utilisateur.
- Fréquence d'envoi des messages : le temps de latence du dernier message RTCM reçu, fréquence d'envoi des corrections au format RTCM.

Nous allons étudier, dans ce qui suit, ce que la littérature nous dit de l'influence de ce paramètre essentiel pour les performances du RTK : la distance entre la base et l'utilisateur.

• **Impact de la distance base-utilisateur sur les autres paramètres de performances**

La distance utilisateur-base a une influence qui peut être considérable dans la résolution RTK. La distance maximum pour laquelle les performances sont considérées acceptables varie selon les études de cinq à une cinquantaine de kilomètres. On définit la méthode RTK comme "distance-dependent" soit dépendant de la distance entre la base et l'utilisateur. Ce critère de base aura en effet un impact direct sur la précision de la position et sur le TTFF.

Dans l'étude RTK faite par (Feng , et al., 2008), sept récepteurs GNSS ont été utilisés pour calculer la position d'un utilisateur avec différentes distances base-utilisateur allant de 2.5km à 31km. On voit par cette étude (figure 2) l'impact de la « baseline » sur nos deux critères fondamentaux : le TTFF et la précision.

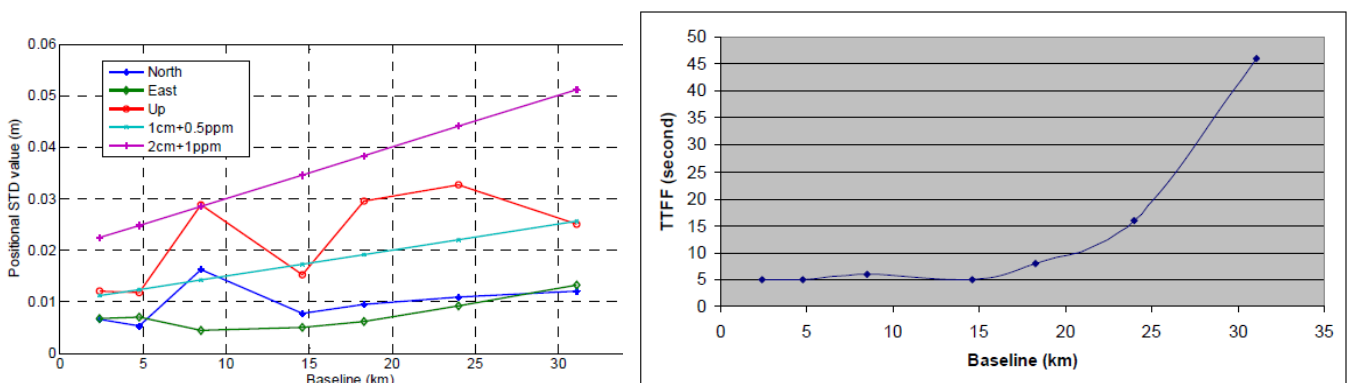


Figure 2- TTFF et la précision selon la distance utilisateur station de base (Feng , et al., 2008)

Dans l'introduction, nous avons fixé un cadre d'application pour la précision et le temps de convergence : quelques centimètres en quelques minutes. On peut voir sur la Figure 2 que, même à

plus de 30km, ces conditions sont toujours respectées, suivies d'une dégradation globale au fur et à mesure où la distance augmente. C'est pour cela que l'algorithme RTK est considéré comme très robuste. Mais ce n'est pas aussi simple : augmenter la distance entre la base et l'utilisateur entraîne de nouvelles erreurs (Wübbena, et al., 1996) .

En effet quand cette distance devient trop grande, le résultat de la différenciation de l'équation (1.2) change, un nouveau terme apparaît :

$$\Delta\varphi_{ru,j}^{sp} = \Delta\rho_{ru}^{sp} + N_{ru,j}^{sp} + \delta \quad (1.3)$$

Un biais δ est à prendre à compte car les erreurs supprimées lors de la différenciation entre deux récepteurs et deux satellites ne sont plus les mêmes, donc ne se compensent plus (Baybura, et al., 2019). Il y a plusieurs raisons à cela. Tout d'abord, la double différenciation repose sur l'hypothèse qu'entre la base et le récepteur, les chemins parcourus par les signaux depuis le même satellite sont parallèles. Plus la longueur de la « baseline » augmente, moins cette hypothèse est vraie. Autre phénomène relié au précédent : comme on peut le voir sur la Figure 3, le chemin parcouru par un signal à travers l'ionosphère peut être différent et avoir un retard plus ou moins important (Abba, et al., 2015). En augmentant la distance utilisateur-base dans le RTK, les perturbations ionosphériques pour un signal atteignant la base et l'utilisateur ne seront plus les mêmes (Wanninger, 2004). C'est une autre contribution au reste δ qui apparaît dans la différenciation. Les erreurs d'horloge peuvent elles aussi entraîner un décalage dans cette différenciation.

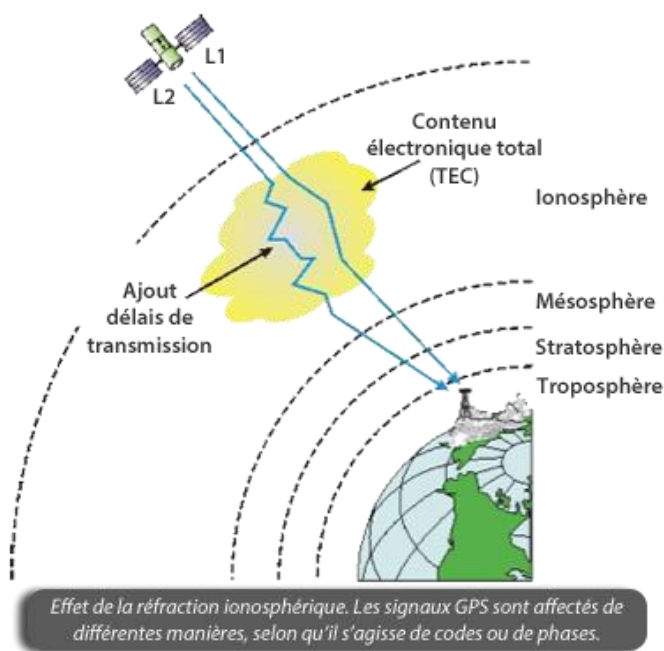


Figure 3- Erreurs ionosphériques sur deux signaux GPS

Ces erreurs ionosphériques ne sont pas prises en compte lorsque la distance est inférieure à 20 km (Wanninger, 2004) au-delà, le RTK devient plus dépendant de la distance entre l'utilisateur et la station de base. Plus la station de base sera éloignée de l'utilisateur plus l'erreur δ sera importante. Ces effets sont plus importants pour les satellites qui sont plus près de l'horizon (basse élévation).

De même que Feng, Wübbena a été un des premiers à montrer une relation entre la distance et le temps de convergence (Figure 4). Ici, le TTFF est assimilé au TTFA (Time To Fix Ambiguity), il est considéré que la résolution des ambiguïtés entraîne instantanément le calcul de la position. On peut voir qu'au-delà d'une vingtaine de kilomètres dans ce cas, le temps de convergence est au minimum de 2 minutes avec un maximum entre 5 et 10 minutes. L'état de la constellation ayant bien sûr une influence difficile à analyser sur des résultats globaux.

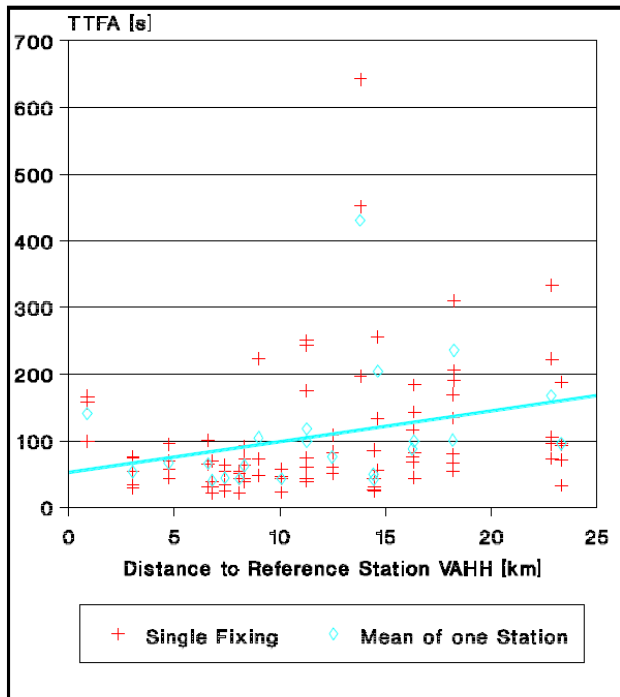


Figure 4- TTFA en lien avec la distance base-utilisateur (Wübbena, et al., 1996)

Ces deux études sur le RTK révèlent l'impact important de cette distance sur la précision et le TTFF. Toutefois on peut voir que la distance limite d'application de la méthode RTK peut être très variable. En effet, dans la première étude le temps de convergence reste rapide (moins d'une minute) même avec une distance base-utilisateur supérieure à 30 km. Alors que dans les tests faits par (Wübbena, et al., 1996), le temps moyen de convergence sera de minimum deux minutes quand on dépasse les 20km de distance base-utilisateurs. En plus de l'influence de la constellation citée plus haut, cette variation dépend des modèles utilisés dans les algorithmes RTK et des algorithmes de détermination des ambiguïtés en général.

Dans l'application standard du RTK, la distance limite utilisateur – base est fixée à 20km pour que la précision de la position reste proche du centimètre et ait un temps de convergence rapide (<1minute). Cette limite de distance explique que l'on définisse la solution RTK comme une technique de positionnement GNSS locale.

Comme expliqué précédemment, les performances (Figure 1) obtenues pour des algorithmes RTK peuvent varier en fonction des modèles utilisés dans la résolution (Wang, 1999).

- **Impact sur la résolution des ambiguïtés et sur la précision**

La principale inconnue dans l'équation (1.2) est l'ambiguïté $N_{ru,j}^{sp}$. C'est dans l'estimation de ce paramètre que des modèles sont utilisés. Le principal avantage du RTK pour le calcul des ambiguïtés est qu'une fois double différenciée, l'ambiguïté $N_{ru,j}^{sp}$ est un entier. Par cette nature, les programmes appliqués pour l'estimer seront plus simples et permettront d'avoir un temps de convergence plus rapide.

La principale technique utilisée aujourd'hui dans ce cas est la méthode LAMBDA : Least-squares AMBiguity Decorrelation Adjustment. Dans la plupart des études RTK présentes dans mon état de l'art, LAMBDA est présentée comme la seule façon de résoudre les ambiguïtés entières rapidement de manière précise (Li, et al., 2011). La manière dont les ambiguïtés sont déterminées sera détaillée quand j'aborderai la résolution mathématique de ma solution hybride dans la partie 3.

Parmi les critères de performance vus plus haut, ceux utilisés pour évaluer ce modèle LAMBDA sont le taux AR fixé, le TTFF (TTFA) et la Fiabilité AR.

Toutefois, on peut faire un lien avec la distance base utilisateur et ce modèle de résolution : il y aura un impact direct sur ces trois paramètres de performances. Un exemple d'erreur entraînée par l'augmentation de la distance sur la disponibilité de l'AR qui correspond au taux d'AR fixé en pourcentage est présenté dans la Figure 5.

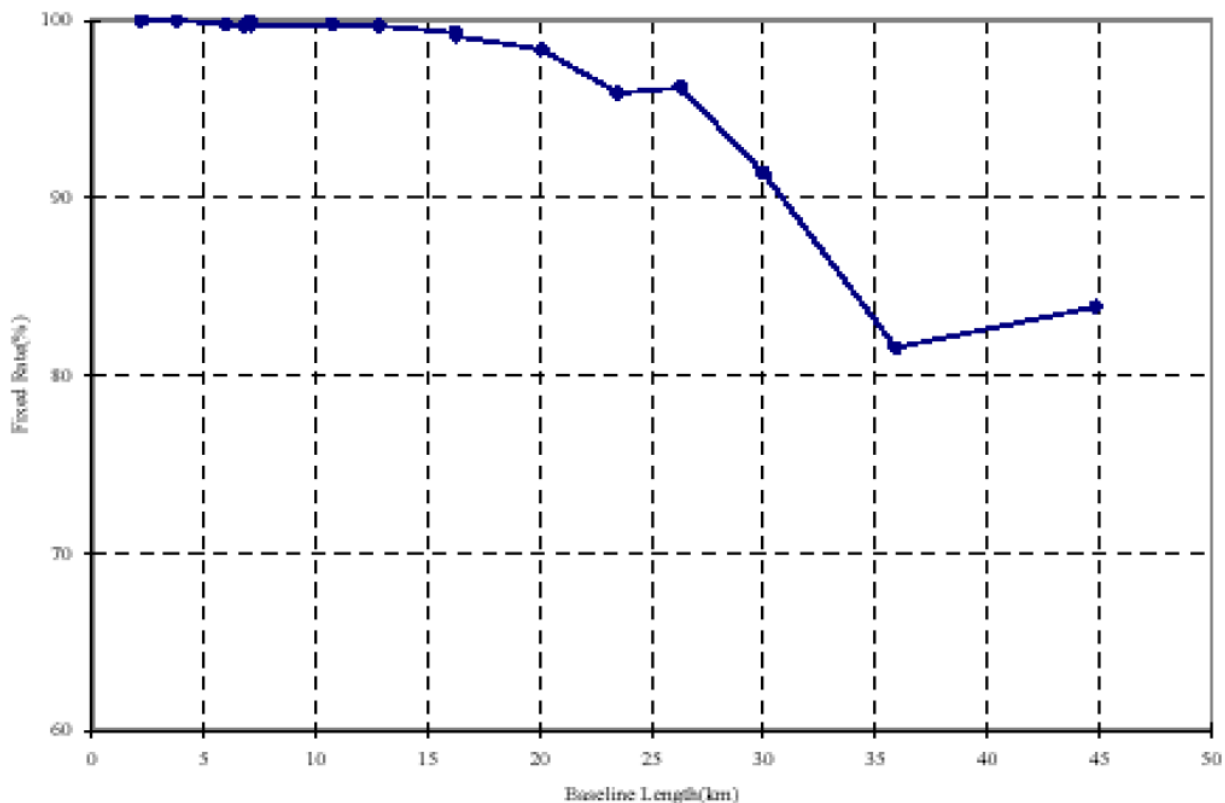


Figure 5- Taux AR fixé en fonction de la distance utilisateur-stations de base

Ainsi, plus la distance utilisateur-base est longue plus le taux de résolution des ambiguïtés sera dégradé. De même, dans le cadre de l'application du RTK dans des conditions sévères comme au

milieu d'une forêt ou dans des canyons urbains, la résolution des ambiguïtés pourra devenir plus difficile et ainsi allonger le temps de résolution. En effet, la résolution des ambiguïtés devra surement se réinitialiser plusieurs fois à cause de la perte de signal due aux différents obstacles. Mais malgré ces phénomènes, le RTK reste fiable avec une bonne précision (< 5cm) en se basant sur ces résultats on peut qualifier le RTK de solution assez robuste aux obstacles (Bakuła, et al., 2009). A noter que dans ces conditions, avoir des satellites avec une haute élévation présente un réel avantage. Ceci est à retenir lors de l'élaboration de notre solution hybride.

- **Multiplicité des signaux et apparitions de nouveaux modèles**

D'autres modèles du RTK existent pour estimer certains critères de performance quand la quantité de données GNSS disponibles augmente. On pense de façon naturelle à la multiplicité des signaux : multi-constellation et multifréquence. En effet, dans l'équation différenciée (1.2), les dépendances au satellite et à la fréquence associée sont importantes. Il est possible d'utiliser plusieurs fréquences ainsi que des satellites issus de différentes constellations.

Ainsi, si on prend plusieurs fréquences et plusieurs satellites et qu'on les applique à notre équation double différenciée (1.2), on obtient pour des m satellites $s = s_1, \dots, s_m$ et deux fréquences $j = 1, 2$:

$$\Delta\varphi_{ru,\Delta j}^{\Delta s} = \begin{matrix} \Delta\varphi_{ru,1}^{s_1s_2} & \Delta\varphi_{ru,2}^{s_1s_2} \\ \dots & \dots \\ \Delta\varphi_{ru,1}^{s_{m-1}s_m} & \Delta\varphi_{ru,2}^{s_{m-1}s_m} \end{matrix} \quad (1.4)$$

On se retrouve avec une matrice de taille (m,2) qui résume l'ensemble des équations double différenciées possibles et utilisables pour calculer la position souhaitée en RTK.

Des modèles stochastiques vont permettre de résoudre les ambiguïtés et la position grâce aux données de cette grande matrice. Utiliser des satellites de différentes constellations permet d'avoir plus de satellites visibles et disponibles pour la résolution. Ces données permettront d'estimer de manière plus fine certains paramètres grâce notamment à des filtres de Kalman (Wang, 1999). L'avantage d'utiliser plusieurs constellations et plusieurs fréquences est de réduire certaines erreurs et d'améliorer certains paramètres influençant la performance (Feng, et al., 2007). Cela a notamment un impact direct sur la fiabilité et la disponibilité de la résolution des ambiguïtés. Grâce à une constellation supplémentaire on peut augmenter les résultats sur l'AR de plus de 5% dans certains cas (Table 1).

Table 1 - Statistiques des performances RTK avec une distance utilisateur base de 38km (Feng, et al., 2007)

	GPS Measurements Only		GPS +Virtual Galileo Measurements	
	P12, L12	P12, L12, L1	P12, L12	P12, L12, L1
Total Epochs	5758	5758	5638	5638
RTK available epochs	3167	3614	4080	3705
Total DD phase measurements	36243	72489	70623	141246
Total fixed DD integers	29790	61629	63982	124843
AR Reliability	82.19%	85.02%	90.60%	88.38%
RTK Availability (AR reliability)	54.98%	62.74%	72.38%	65.71%
RTK availability (Accuracy within ± 5 cm: E, N, U)	NA	76.14% 85.78% 48.54%	NA	85.67% 92.66% 60.39%
RTK accuracy (STD in E,N U in metre)	NA	0.0190 0.0177 0.0192	NA	0.0184 0.0164 0.0213
RTK accuracy (RMS in E,N,U in metre)	NA	0.0190 0.0183 0.0192	NA	0.0183 0.0169 0.0213

Wang Gao étudie aussi cet impact de l'utilisation de plusieurs constellations avec les satellites GPS et Beidou. Dans son cas d'application, la distance base-utilisateur de 75km est très grande par rapport au standard (Gao, et al., 2017). On aura plus de satellites disponibles, même à grande distance. L'intérêt du multi-constellation prend ainsi tout son sens si on se souvient de ce que nous avons vu plus haut sur l'influence de cette distance sur la précision et le temps de convergence.

Table 2- Comparaison des erreurs de positionnement avec du multi-constellations

Number of Satellites	RMSs of Positioning Errors (cm)						Improvement (%)		
	Classical Differencing			Inter-System Differencing			N	E	U
	N	E	U	N	E	U			
7	1.93	1.33	11.25	1.25	0.96	9.90	35.2	27.6	11.9
8	1.49	1.03	8.08	1.05	0.82	6.97	29.7	20.7	13.7
9	1.04	0.78	5.02	0.95	0.74	4.88	8.5	4.7	2.7
10	0.91	0.69	2.77	0.85	0.68	2.71	6.1	2.4	2.2
11	0.77	0.65	2.63	0.72	0.64	2.60	6.2	0.6	1.1
12	0.73	0.56	2.44	0.68	0.56	2.44	6.7	1.5	0.3
All	0.68	0.51	2.27	0.65	0.51	2.27	4.4	0.2	0.0

Avec l'augmentation du nombre de satellites, les erreurs sur le positionnement semblent se réduire et les résultats du point de vue de la résolution des ambiguïtés semblent meilleures. Toutefois, dans ce cas très particulier d'une grande distance base-utilisateur, la précision du RTK a subi un grand impact : plusieurs décimètres d'erreur au lieu d'une précision centimétrique pour les distances plus courtes.

Toutefois, il existe une autre limite avec la constellation GLONASS : les paramètres d'ambiguïtés sont reliés aux paramètres d'horloge du récepteur. Ainsi, avec les satellites GLONASS, il y a un biais supplémentaire dans les ambiguïtés, généralement appelé biais GLONASS. Cela vient du fait que contrairement aux autres constellations GNSS, GLONASS utilise le FDMA (Frequency Division

Multiple Access) comme technique de contrôle d'accès. Autrement dit, chaque satellite émet à une fréquence différente. Lorsqu'on applique la différence entre les mesures provenant de deux satellites différents pour le même récepteur, les ambiguïtés ne sont pas de même nature puisqu'elles ne correspondent pas à la même longueur d'onde.

Dans cette partie, nous avons délaissé le paramètre de fréquence des messages RTCM, et les critères de continuité et d'intégrité du RTK. La raison principale vient de ce que ceux-ci sont tout d'abord liés à la façon dont les données sont transmises à l'utilisateur pour permettre le calcul de la position finale. Ils ne concernent qu'indirectement le calcul mathématique à proprement parler. Nous reviendrons néanmoins sur eux dans la partie suivante lorsque nous aborderons les différents modèles de représentations.

1.1.2 Généralisation du RTK : le NRTK

- **Mise en Réseau du RTK**

Le NRTK, pour Network RTK, est la généralisation de l'algorithme RTK pour passer la solution d'une échelle locale à une échelle régionale qui sera donc plus large. Le principe repose sur la mise en place d'un réseau de stations de base interconnectées entre elles et couvrant une zone plus grande que celle que permet la portée du RTK classique. On appelle les stations de base CORS pour Continuously Operating Reference Stations. Grâce à ce réseau de stations, il sera possible d'appliquer le principe RTK sur une zone plus large tout en conservant la possibilité de mettre en œuvre le RTK entre l'utilisateur et une des stations du réseau (souvent la plus proche).

Selon les limitations que nous avons vues à la section précédente, si on appliquait le RTK classique, il faudrait qu'au sein de ce réseau, les stations de références soient espacées d'une vingtaine de kilomètres les unes des autres. Nous allons voir que la mise en réseau permet d'augmenter cette distance. Il existe plusieurs manières de mettre en œuvre le NRTK parmi lesquelles on peut citer : FKP (Flächen-Korrektur-Parameter), MAC (Master Auxiliary Concept) et VRS (Virtual Reference Station). Le FKP et le MAC reposent sur un principe de maître-esclave : une station va contrôler les autres autour d'elle. Toutefois, la plus utilisée reste la VRS, qui est d'ailleurs la méthode déployée par Exagone. Cela passe par la création d'une station virtuelle appelée VRS (Figure 6). Ce procédé a deux objectifs : le premier est d'avoir une station de base (virtuelle) à une distance compatible RTK de l'utilisateur (moins de 10 km), le second avoir une meilleure estimation de certains paramètres grâce aux différentes données récoltées par l'ensemble des stations de base du réseau.

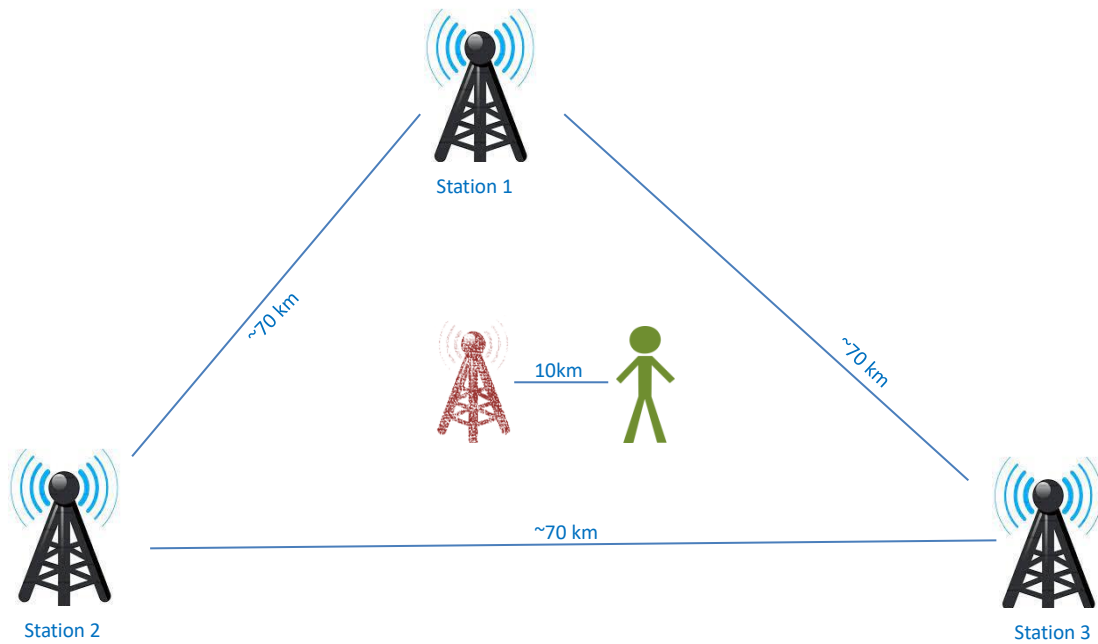


Figure 6 - Création de la station virtuelle VRS

Le NRTK qui utilise une VRS a un énorme avantage en termes de précision qui provient de l'estimation des erreurs ionosphériques de la station virtuelle. De plus, utiliser une VRS ne nécessite aucun changement algorithmique et matériel : dès lors que l'on crée la station virtuelle à partir des bons modèles, on peut utiliser les mêmes qu'en RTK (Retscher, 2002). C'est ainsi que pour notre étude, nous nous concentrerons sur ce type de NRTK.

Le séquençage se déroule de cette manière : dans un premier temps, grâce aux différents modèles implémentés, le serveur du réseau va estimer tous les paramètres des équations d'observations GNSS de la station virtuelle. Dans un second temps, le principe de double différenciation entre la station virtuelle et l'utilisateur sera mis en place. Dans la partie précédente, on a vu qu'un biais pouvait apparaître et rendre difficile l'application de la différenciation et ainsi du calcul de la position (équation (1.3)). Dans le cas de l'utilisation d'une station virtuelle, l'apparition de biais va être plus réduite. Ceci, notamment grâce aux modèles ionosphériques qui sont mis en place dans le réseau : l'estimation des erreurs liées au VEC (Vertical Electron Concentration) dans l'ionosphère est meilleure en NRTK d'après (Wanninger, 2004). Il semble qu'on limite ainsi les effets d'un potentiel biais qui viendrait perturber la précision du calcul de la position, tel qu'on l'observait en RTK avec l'augmentation de la distance station-utilisateur. Ainsi, le NRTK connaît les mêmes performances en termes de précision que le RTK dans ses conditions optimales de fonctionnement : quelques centimètres avec un TTFF rapide (<1min) (Vollath, et al., 2002).

- **Un réseau de stations de référence**

Il existe différents types de réseau de stations de bases GNSS. Par exemple le réseau Teria comportant 295 stations espacées en moyenne de 55km (Figure 7) couvre l'ensemble de la France.



Figure 7- Réseau CORS Teria

La taille et la densité des réseaux de stations sont très variables et dans la plupart des études NRTK rapportées dans la bibliographie, on se limite à une zone régionale, ce qui correspond à une dizaine de stations couvrant une zone plus petite que la France. Toutefois, il est possible d’avoir un réseau aussi grand que souhaité, les limitations que l’on rencontrera seront d’ordre plus techniques et concerneront essentiellement l’interconnexion au sein du réseau.

Le réseau de stations de références est construit selon une géométrie permettant de couvrir une large zone et répondant à des contraintes terrain particulières. Cela crée des nouveaux paramètres de performance qui viennent compléter la Figure 1 vue précédemment. Cela permet aussi de différencier les sources des erreurs que nous verrons dans la partie sur les modèles de représentations.

- **De nouveaux paramètres de performances**

Aux vues de ce que nous avons dit précédemment, on peut donc reprendre les critères de performances identifiés pour le RTK, en leur en ajoutant de nouveaux prenant en compte la multiplication des stations de référence :

- La géométrie et la densité du réseau : nombre de stations avec la taille de la zone couverte.
- Estimation des erreurs inter-stations.

En ajoutant ces critères à la Figure 1, on obtient la figure 8.

De plus, la distance Base-Utilisateur devient la distance Utilisateur-Station virtuelle. Cette distance peut être définie par l’utilisateur et peut varier en fonction du nombre de stations utilisées et de la densité du réseau.

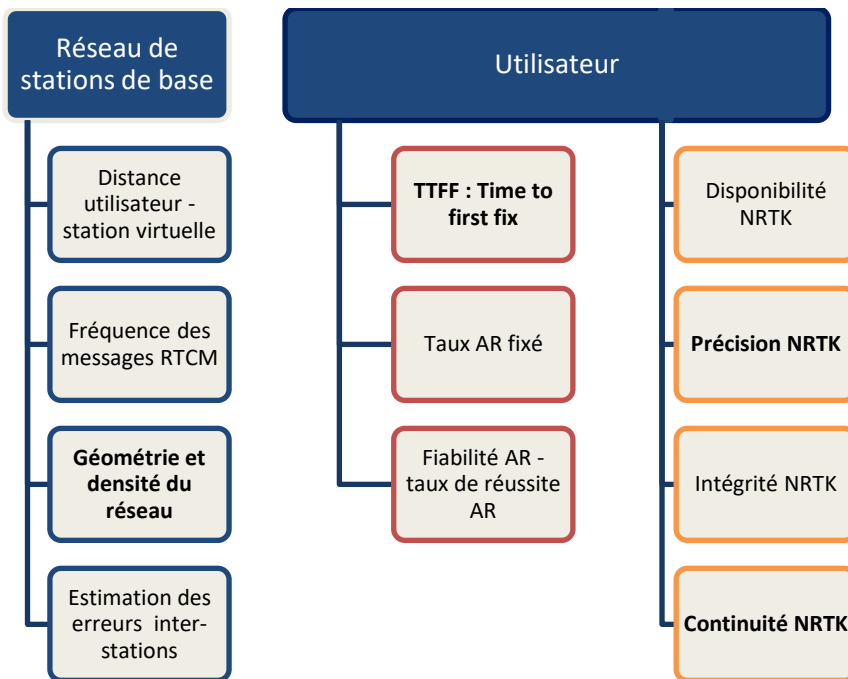


Figure 8- Paramètres et critères de performance NRTK

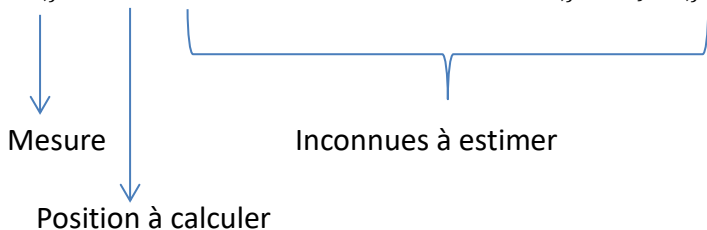
Maintenant que nous avons présenté les principes des techniques RTK et NRTK, ainsi que les critères qui permettent d'évaluer leurs performances, nous allons aborder l'approche avec laquelle nous allons réaliser l'hybridation : le PPP.

1.1.3 PPP – Precise Point Positioning

- **Le PPP une solution absolue**

La technique de résolution Precise Point Positioning est une solution dite « augmentée ». A la différence du RTK, il n'y a pas de différenciation entre les équations. En reprenant les équations d'observations GNSS, on trouve pour un utilisateur u , un satellite s , à une fréquence j :

$$\varphi_{u,j}^s - \rho_u^s = c(dt_u - dt^s) + \tau_u^s - \Delta_u^s - \delta_j^s + \lambda_j N_{u,j}^s \quad (1.5)$$



L'absence de différenciation, qui permet d'éliminer plusieurs sources d'erreur, nous contraint à estimer chaque paramètre inconnu pour calculer la position ρ_r^s . C'est pourquoi on parle de méthode de résolution dite « augmentée », car on a besoin d'information supplémentaires pour effectuer le calcul. Il est aussi commun de qualifier cette méthode comme « non différenciée et non combinée » en opposition à la méthode RTK. Toutes ces qualifications ont pour objectif de mettre en avant la qualité principale d'une telle technique : une résolution avec un seul récepteur appelé « single receiver » (Alkan, et al., 2016). En pratique, un utilisateur aura besoin d'un récepteur et des

corrections PPP pour calculer sa position. La principale limite de cette méthode est un temps long de convergence (>30min) pour une précision décimétrique. Nous verrons pourquoi par la suite.

Les corrections PPP seront composées de cinq éléments de corrections correspondant aux paramètres inconnus à estimer des équations d'observations :

- Les orbites des satellites.
- Les horloges des satellites.
- Les biais de code et de phase.
- L'ionosphère.
- La troposphère.

La mise en œuvre d'une telle solution peut se faire de manières très variées. On peut noter que quelle que soit la méthode, elle doit s'appuyer sur des sources de données externes, comme, en France, des stations de l'IGS par exemple. Le réseau de l'IGS est composé de 350 stations (Aggrey, et al., 2014). Ces stations sont utilisées pour définir les corrections PPP. Un des exemples de programme PPP est PPP Wizard (Laurichesse, et al., 2015). Dans la figure 9, nous pouvons voir l'ensemble des modules nécessaires pour programmer une technique PPP. Cela nous servira de point de départ pour la mise en place de la solution que nous présenterons dans la partie 3.

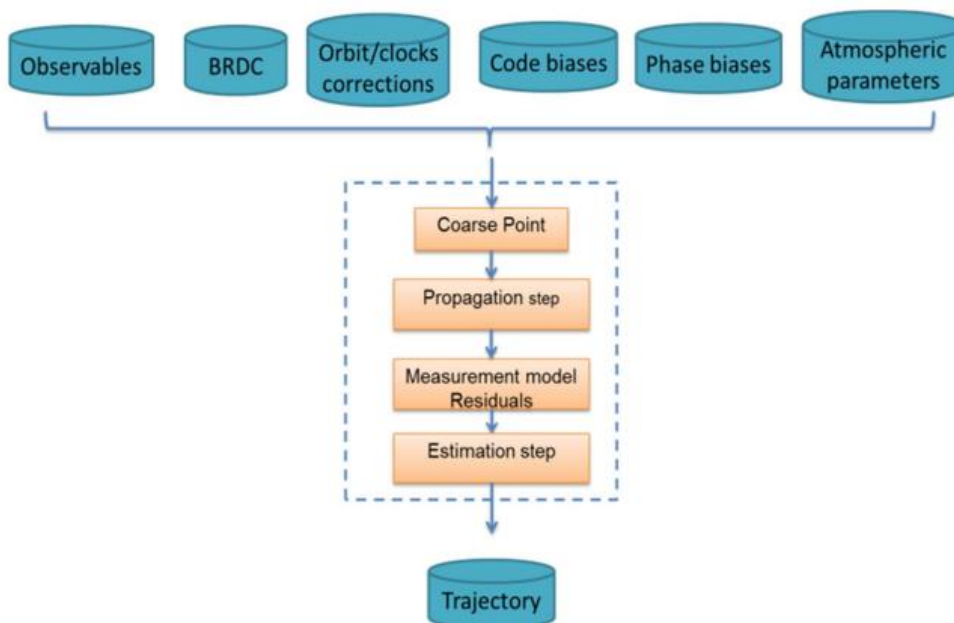


Figure 9- Module fonction PPP

Cette figure synthétise la méthode de résolution qu'un utilisateur suit pour calculer sa position.

- **Critères de performances**

Les deux principaux paramètres sur lesquels nous nous concentrons pour comparer nos techniques de positionnement précis sont la précision obtenue lorsque la position est déterminée et le temps pour obtenir une solution. Dans le cadre du PPP, ces deux critères sont vraiment indissociables. En

effet, en fonction du temps du calcul de la position, la précision peut passer d'un ordre de grandeur du mètre au décimètre voir même dans les meilleures conditions à une précision centimétrique. Il est donc important dans l'application du PPP de prendre en compte ces deux critères. Nous présentons donc une version de la Figure 1 adaptée au PPP (Figure 10).

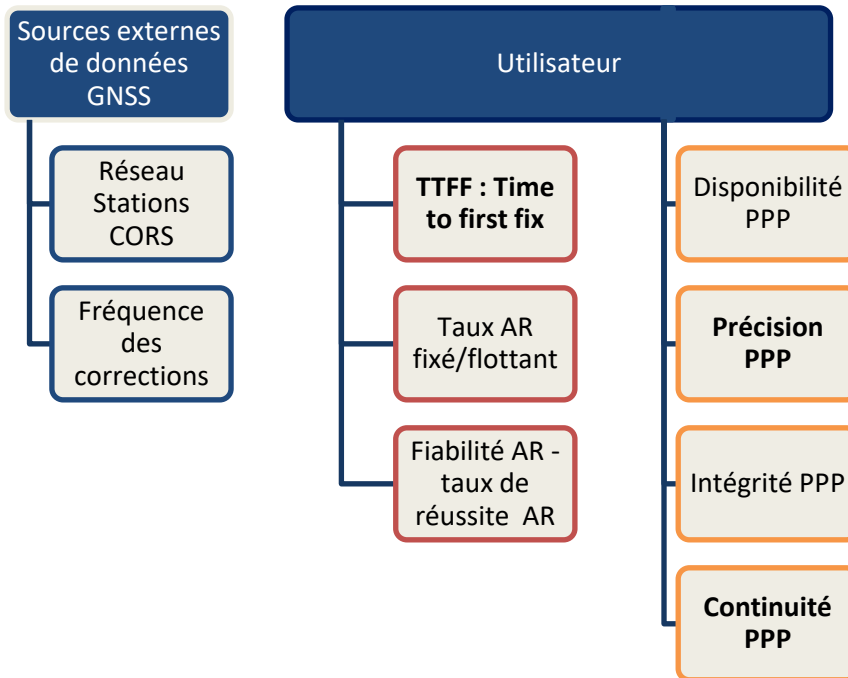


Figure 10- Paramètres et critères de performance PPP

Sur la figure 10, on retrouve un critère qu'on a déjà vu pour le RTK et que l'on doit prendre en compte. Il est à la source du long temps de convergence et de la « mauvaise » précision. Il s'agit du taux de la résolution des ambiguïtés. En effet, en PPP la résolution des ambiguïtés est plus difficile qu'en RTK. Comme on ne fait pas de différentiation, dans le cas général, l'ambiguïté $N_{u,j}^S$ n'a aucune raison d'être un nombre entier : c'est un nombre décimal (flottant). L'estimation par les moindres carrés utilisée dans LAMBDA ne peut plus être appliquée car le problème devient trop complexe. Plusieurs solutions existent consistant à « rendre l'ambiguïté entière » pour se ramener à un problème équivalent à celui du RTK. Pour cela, il est nécessaire d'estimer convenablement un biais que le satellite engendre sur la phase de la porteuse du signal à une fréquence donnée et qu'on appelle le biais de phase satellite. Une fois connu ce biais, il sera alors possible d'avoir un terme d'ambiguïté entière $\lambda_j N_{u,j}^{S'} = \lambda_j N_{u,j}^S - \delta_{j,j}^S$. Le taux de résolution AR fixé/flottant est donc un bon indicateur de performance d'un algorithme PPP, pour lequel le temps de convergence est la principale limite.

Un autre paramètre influençant les performances : les sources de données de corrections. Une autre source de données, en plus des constellations de satellites, peut être un réseau de stations au sol. Rappelons-nous que l'utilisateur n'est pas directement connecté au réseau comme en NRTK : les stations sont là pour combiner des données et générer les corrections, pas pour calculer la position de l'utilisateur. Par exemple, le réseau de l'IGS (International GNSS Service - 200 stations) est utilisé pour calculer les corrections PPP (Li, et al., 2010). En effet, nous insistons encore une fois sur ce point : ce qui différencie fondamentalement le RTK (NRTK) du PPP, c'est que la première est une

technique différentielle nécessitant une connexion bidirectionnelle entre le récepteur et la station ou le réseau, alors que le PPP est une technique qui exploite des données d'augmentation diffusées sans connexion montante au sein d'un réseau. Par exemple, le système EGNOS (European Geostationary Navigation Overlay Service) est un réseau européen qui diffuse des corrections sur la zone européenne. L'utilisateur n'a théoriquement pas besoin d'échanger avec le réseau qui les diffuse.

Un autre critère, qui peut varier d'une technique PPP à une autre, est la fréquence d'envoi des corrections au format RTCM (Radio Technical Commission for Maritime Services) à l'utilisateur. Comme chaque paramètre est calculé de manière indépendante et que le taux de variation de leur valeur dans le temps n'est pas le même, il est possible d'envoyer différents messages de corrections à différentes fréquences. Par exemple, les corrections d'horloges toutes les secondes et les corrections atmosphériques toutes les dix secondes.

- **Une solution double fréquence**

Le PPP conserve tout de même des similitudes avec le RTK, pour la résolution des erreurs ionosphériques notamment. En effet, l'utilisation de deux fréquences permet l'application du modèle sans ionosphère « ionosphere free combination » en différenciant les équations d'observations GNSS associées à chaque fréquence.

Cette combinaison permet d'éliminer le premier ordre des délais ionosphériques (Li, et al., 2021). C'est suffisant dans le cadre de la résolution PPP et revient à éliminer l'ensemble des délais ionosphériques. On pourra raisonnablement considérer qu'avec deux fréquences les erreurs ionosphériques peuvent être négligées.

- **Un exemple de variante PPP : PPP-AR**

L'essentiel de la littérature sur le PPP se focalise sur la proposition de variantes pour « accélérer » la résolution du PPP jusqu'à une précision centimétrique. Le plus fameux est PPP-AR : PPP avec résolution d'ambiguïté. Cette méthode a pour but de rendre l'ambiguïté entière, on l'a vu plus haut, et de garder en mémoire le calcul des ambiguïtés en cas de perte du signal.

Table 3-Comparatif TTFF avec solution flottante/fixe en PPP et PPP-AR

	PPP	PPP-AR
Solution flottante	1470sec	750sec
Solution Fixe (TTFF)	30min	20min

D'après la table 3, on peut voir une différence de temps de calcul entre les deux méthodes, avec à la fin une précision identique pour les deux. L'apport du calcul des ambiguïtés dans le cadre du PPP-AR, permet surtout une accélération de la résolution qui, pour intéressante qu'elle soit, 50 % plus rapide pour la solution flottante et 33 % pour la solution centimétrique, reste loin des performances du NRTK en termes de TTFF.

Ainsi, nous pouvons voir que la technique absolue² PPP permet d'avoir une approche différente dans la résolution sans présenter de dépendance directe au principe de la différenciation des équations d'observations. Toutefois, elle rencontre une importante limite en termes de temps de convergence et de précision atteinte. Cela explique pourquoi dans l'application des algorithmes GNSS c'est la solution RTK qui est la principale technique commercialisée. En effet, bien que sa mise au point finale représenterait une sorte de « Graal » qui permettrait d'alléger considérablement les infrastructures pour obtenir du positionnement précis à convergence rapide, aujourd'hui presque aucun récepteur GNSS ne permet un calcul de la position à l'aide d'un algorithme PPP. Les corrections PPP commencent néanmoins à apparaître petit à petit, ceci en lien avec un nouveau modèle de représentation SSR que nous allons présenter dans la partie suivante qui va s'intéresser au système de représentation.

1.2 Système de représentation

Il existe deux systèmes de représentation quand on parle de positionnement précis en GNSS. Ces modèles découlent des méthodes de résolution des équations d'observations GNSS. Ces systèmes ont une utilité dans la définition des erreurs entrant en jeu dans la résolution et de la source de celles-ci. De plus, définir des systèmes de représentation est intéressant et utile pour la comparaison RTK et PPP. Ainsi aujourd'hui, il existe deux systèmes de représentation :

- OSR pour « Observation Space Representation » : représentation de l'espace d'observation.
- SSR pour « State Space Representation » : représentation de l'espace d'état.

Un modèle complet d'espace d'état (modèle SSR) avec une précision millimétrique est mis en œuvre pour l'ajustement rigoureux et simultané des observables GNSS. La modélisation de l'espace d'état suit l'idée de modéliser les sources d'erreur réelles au lieu de traiter les effets des erreurs. Les effets des erreurs appartiennent à l'espace d'observation (OSR), tandis que les sources d'erreurs sont associées à l'espace d'état (SSR). Toutes les sources d'erreur constituent le modèle d'espace d'état (SSM). Le modèle d'espace d'état se compose :

- Des algorithmes de modèle.
- Des paramètres du modèle.

Ces deux représentations envoient des données, appelées corrections, à l'utilisateur sous deux formats différents détaillés dans le tableau ci-dessous :

Table 4-Comparaison des formats de corrections OSR et SSR

	OSR	SSR
Corrections envoyées	Correction de la mesure de phase	Corrections Horloges (récepteurs/satellites)
	Correction de la mesure du code	Corrections atmosphérique Corrections biais phase et code (satellites/récepteurs) Orbites

² Absolue au sens où la position est calculée directement dans le référentiel terrestre, alors que formellement le RTK calcule la position de l'utilisateur par rapport à celle de la station de base. Même si la solution est bien absolue aussi.

1.2.1 OSR

La première représentation mise en place est l'OSR. La technique généralement associée à la représentation de l'espace d'observation est le RTK ou le NRTK. En effet, le RTK est une technique qui va utiliser les messages d'observations pour sa résolution, que l'on classe dans les corrections OSR qui sont liées directement aux mesures de phase et de code.

- **Corrélation des erreurs dépendant de la distance**

Les erreurs GNSS pour les observables de la phase de la porteuse dans la figure ci-dessous, montrent qu'elles sont une somme globale d'erreurs GNSS dépendant de la distance. La correction de la phase pour la position du récepteur, ou un modèle spatial équivalent pour décrire son effet sont actuellement utilisés dans les solutions réseaux NRTK.

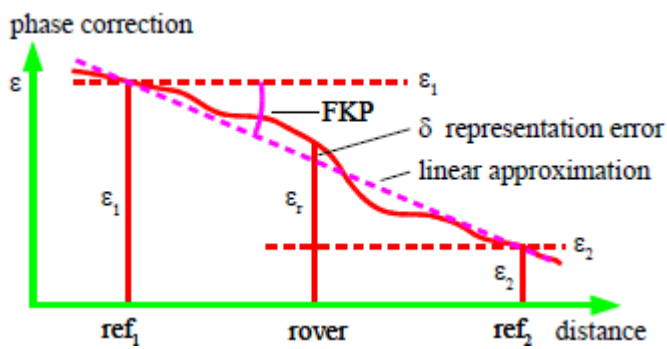


Figure 11- Représentation des erreurs sur le principe OSR avec les techniques NRTK

Ainsi, on pourra appliquer ce principe à l'ensemble des erreurs, comme on le voit dans la figure 11. Cette interpolation linéaire entre les stations sera facilement ajustable à l'ensemble du réseau.

En fonction de l'algorithme du réseau NRTK et des paramètres de conception du réseau, les corrections du réseau NRTK présentent une incertitude, généralement appelée erreur de représentation. Cette erreur est causée par les conditions physiques irrégulières entre les stations de référence réelles et ne peut être améliorée qu'en diminuant la distance entre les stations. L'OSR est la méthode actuellement utilisée dans les applications de réseau RTK pour fournir une correction ou des données corrigées à un utilisateur.

De plus, il est possible de catégoriser les sources d'erreurs dépendant de cette distance récepteur/station en 5 catégories (Baybura, et al., 2019).

- Les erreurs troposphériques.
- Les erreurs ionosphériques.
- Multi-trajets et obstruction des signaux.
- Configuration géométrique des satellites.
- Autres erreurs (biais physique des récepteurs, etc.).

Nous en avons déjà évoqué certaines dans l'introduction. Chacune de ces sources d'erreurs ne sera pas directement corrigée lors de la résolution d'un point de vue utilisateur. Ce sont les mesures de phases, des observables par nature donc, qui sont corrigées. C'est exactement la définition d'une technique OSR : on corrige les effets des erreurs en RTK, soit sur la mesure de phase ou/et sur la mesure de code. C'est un pilier fondamental des corrections OSR qui est aussi dû aux différentes différentiations effectuées dans la résolution.

Pour les sources d'erreurs du NRTK, il est aussi possible de les définir à l'aide des niveaux de configuration (Vollath, et al., 2000) : l'intégrité des données des stations, l'intégrité de la différenciation, la modélisation de la source des erreurs et la modélisation de la station virtuelle. Ces différents niveaux permettent de se rendre compte de l'effet des erreurs sur le calcul de la station virtuelle. Ces données sont nécessaires car plus la station VRS sera bien définie, plus le calcul de la position finale sera performant.

- **Des corrections dépendant des observations**

La représentation des corrections du réseau RTK dans l'espace d'observation utilise toujours les observations GNSS d'une station de référence réelle ou virtuelle, qui sont ensuite appliquées par l'algorithme conventionnel du rover RTK.

La station de référence virtuelle (VRS) dans le cadre d'une solution NRTK ne fournit pas séparément les données d'observation et les données de correction du réseau. Cela se comprend car étant « virtuelle », elle n'a pas d'existence physique en dehors du réseau qui la crée. Les observations de la station de référence et les paramètres d'état sont transformés pour estimer la position de l'utilisateur. Il est donc nécessaire que le récepteur envoie une position approximative pour déterminer celle de la station virtuelle avant que le calcul précis puisse s'effectuer. Les observations du VRS sont transmises au récepteur. Le récepteur RTK reçoit les données d'une VRS, qui est reconnue comme une station de référence à une très courte distance de la position approximative réelle de l'utilisateur.

L'impact de la représentation OSR dans les algorithmes RTK, et surtout en NRTK, se traduit par la nécessité pour l'algorithme de connaître une position approximative de l'utilisateur pour savoir quelles corrections de phase le réseau doit lui envoyer. En effet, les corrections OSR ne sont valables que pour une zone prédéfinie. Une première approximation de la position est possible grâce au calcul de la mesure du code et d'une première résolution d'équations d'observations GNSS. Cette nécessité pour la génération des corrections OSR a un impact d'application fort : une communication bidirectionnelle entre l'utilisateur et le réseau est nécessaire.

1.2.2 SSR

La deuxième principale représentation est celle dite d'espace d'état (SSR). Cette représentation est généralement associée à la technique de positionnement précis PPP. En effet, c'est le parfait exemple de ce type de modélisation. Nous allons voir pourquoi.

- **Identification des sources d'erreurs**

La représentation de l'espace d'état est une description fonctionnelle et stochastique de l'état, c'est-à-dire des composantes individuelles des erreurs GNSS. Les paramètres d'état réels sont transmis au récepteur de l'utilisateur. L'utilisateur corrige ses propres observations provenant d'un récepteur GNSS unique avec les corrections SSR préalablement calculées à partir des paramètres d'état pour sa position individuelle. Puis il effectue un positionnement grâce à des algorithmes PPP par exemple. En effet, la modélisation dans une solution d'augmentation PPP estime les composantes individuelles des erreurs GNSS. Il s'agit donc d'un concept direct d'utilisation des paramètres d'état réels pour la représentation de toutes les erreurs GNSS.

Chaque paramètre dans les équations d'observations GNSS correspond à une source d'erreurs. Sur le principe, cela constitue un énorme avantage d'un point de vue utilisateur : les corrections PPP fournies corrigent directement les erreurs présentes dans les mesures comme des biais des satellites, des corrections d'orbites et des corrections d'horloges des signaux.

- **Diffusion des corrections de type SSR**

Notre problématique de départ sur le positionnement précis GNSS consiste essentiellement en de la géolocalisation. Toutefois, il est important de le relier avec un autre problème lié cette fois-ci à l'aspect télécommunication : la façon dont les données sont diffusées pour être récupérées par l'utilisateur. En effet, en RTK il y a une communication à double sens qui n'est plus nécessaire en PPP ce qui range ainsi cette dernière dans les approches SSR. On peut considérer que le GNSS en mode de positionnement autonome le plus basique qui soit est aussi une approche SSR : les mesures de code sont corrigées grâce aux éléments sur la modélisation ionosphérique présents dans le message de navigation.

Cela met donc en lumière un autre avantage de cette représentation directement lié au mode de diffusion des corrections SSR. La bande passante utilisée pour du SSR peut être considérablement réduite par rapport à celle de l'OSR, car la dynamique des différents paramètres peut être utilisée pour optimiser son occupation. Il est possible par exemple de définir des fréquences différentes pour les paramètres SSR envoyés à l'utilisateur : par exemple les erreurs d'horloge envoyées toutes les 30 secondes et les biais de phase envoyés toutes les 10 secondes. Ainsi, la bande passante utilisée pourra être réduite et adaptée en fonction des besoins de la solution PPP ou de sa zone d'application, toujours sans liaison montante.

Le SSR utilise également des propriétés telles que l'indépendance du satellite par rapport à la modélisation de la troposphère ou l'indépendance de la station de mesure par rapport aux erreurs d'horloge du satellite. Pour l'OSR, un taux de mise à jour commun est généralement nécessaire en raison de la transformation de la somme forfaitaire des erreurs GNSS dépendant de la distance dans l'espace d'observation. On pourrait le résumer en disant que les observations changent trop vite localement. Pour le SSR, les perceptions sont différentes. Le signal ionosphérique (différence des observables de base L1-L2) et les erreurs GNSS non dispersives (ou géométriques) varient à des rythmes différents, ce qui peut être utilisé pour permettre des taux de transmission différents. Ainsi, les corrections SSR peuvent être envoyées à des fréquences différentes selon chaque paramètre et erreur corrigée (Figure 12). A contrario, la largeur de bande dédiée à l'OSR ne peut pas être

optimisée dans la même mesure que celle de la SSR, parce que les composants d'erreur ne sont pas accessibles individuellement.

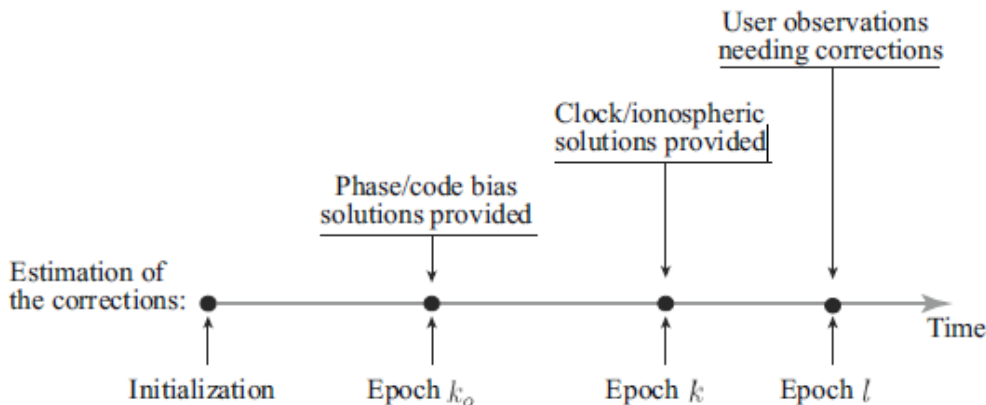


Figure 12-Diffusion de corrections de l'espace d'état (SSR) au fil du temps.

De plus, le SSR est valable pour l'ensemble d'une zone couverte et peut être rendu indépendant du nombre de stations de référence. Dans d'un réseau de stations pour une solution NRTK, on a vu plus haut que quelques restrictions sur l'espacement des stations de référence pouvaient exister en ce qui concerne le comportement irrégulier des conditions physiques entre les stations de référence. Cela permet d'identifier un autre avantage des corrections SSR : ces dernières ne sont pas liées à des stations de référence. Cela provient directement de la définition des différents modes de représentation : le SSR concerne l'environnement indépendamment des mesures qui sont le propre de l'OSR. A titre d'illustration, l'influence atmosphérique dépendante de la position peut être décrite à l'aide de paramètres fonctionnels et/ou maillés qui relèvent du SSR.

Remarquons enfin que la diffusion des SSR convient parfaitement aux services de diffusion pour une zone « illimitée ». La liaison de communication sera donc unidirectionnelle, ce qui élargit le nombre de liaisons de communication appropriées en général. Les canaux de communication généralement considérés comme ne convenant pas à l'approche RTK redeviennent praticables (par exemple, la radio et la télévision). Le SSR est également totalement compatible avec les applications cinématiques dans la zone de service (Duong, 2016).

Maintenant que nous avons décrits les deux approches RTK/NRTK et PPP séparément, voyons ce qu'il en est des techniques ayant pour but de mêler les deux. Ce qui nous amène à la partie suivante.

2. Une extension du PPP : l'hybridation PPP-RTK, état de l'art d'un exemple de solution hybride

Le PPP-RTK est le premier exemple d'hybridation d'algorithmes de positionnement précis GNSS. Elle combine le RTK et le PPP. C'est une technique étendue du PPP qui palie les inconvénients de celle-ci. En effet, nous avons vu dans la partie précédente que la principale limitation du PPP était le temps de calcul des ambiguïtés dans l'équation de la phase, dans le cas d'une résolution simple récepteur. Le PPP-RTK est elle aussi une technique fonctionnant avec un simple récepteur recevant des corrections PPP-RTK. Le principe de résolution se partage en deux parties : le réseau et l'utilisateur. Le réseau est composé de stations permanentes qui permettent notamment de fournir les biais de phases pour permettre la résolution par l'utilisateur. Ces données sont envoyées à l'utilisateur sous forme de messages RTCM appelés corrections PPP-RTK comme on le voit illustré sur la figure 13.

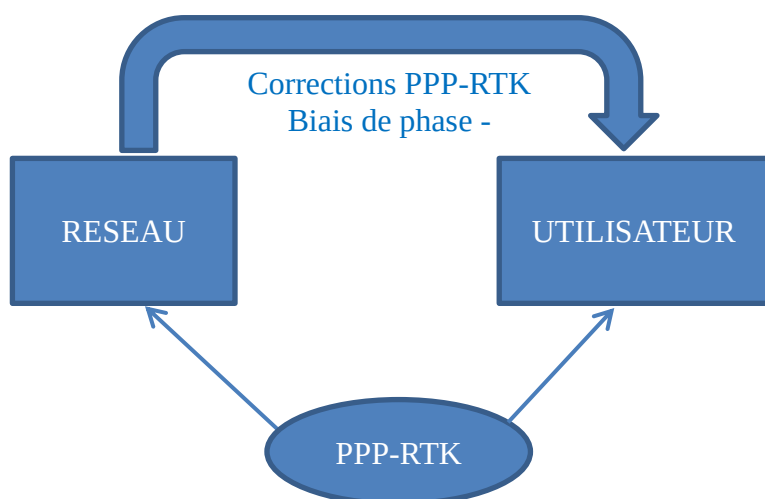


Figure 13 - Principe fonctionnement PPP-RTK

A partir de cela, nous allons voir comment est construite la solution.

2.1 Paramétrage du PPP-RTK – la construction d'une solution hybride

Dans un premier temps, il est nécessaire de définir cette technique hybride afin de comparer ses paramètres principaux à ceux du RTK et du PPP.

L'hybridation PPP-RTK a pour objectif de combiner les deux méthodes pour valoriser leurs avantages et limiter leurs inconvénients. Le principal inconvénient du NRTK est la communication bidirectionnelle nécessaire entre l'utilisateur et le réseau CORS. A l'opposé, celui du PPP est un long temps de convergence (Li, et al., 2022), comme nous l'avons vu dans la section précédente. Ce qui

orienté assez naturellement vers une solution hybride est l'orthogonalité des inconvénients de chacun. On peut espérer raisonnablement profiter des avantages de l'un pour réduire les limites de l'autre en combinant ces deux méthodes. Le RTK a un court temps de convergence alors que le SSR comme le PPP ne nécessite pas de communication bidirectionnelle et a une approche indépendante des erreurs. On peut tenter d'en tirer avantage d'un point de vue applicatif.

Concrètement, en termes de précision, le PPP-RTK est comparable au RTK avec l'avantage d'une application à une échelle plus large et moins dépendante de la distance avec les stations. Les principales fonctionnalités résumant la construction de cette hybridation sont présentées dans le tableau (Table 5) ci-dessous :

Table 5-Construction de l'hybridation PPP et RTK (Wübbena, et al., 2005)

	RTK	PPP	PPP-RTK
Représentation	OSR	SSR	SSR
Echelle	Locale	Mondiale	Régionale
Nature des ambiguïtés	Entière	Flottante	Entière
Précision	cm	dm	cm

On y voit la contribution des méthodes à l'origine de l'hybridation : le PPP apporte surtout l'approche SSR tandis que le RTK permet la résolution des ambiguïtés entières de manière double différenciée. On peut noter aussi que le « mélange » entre les échelles locales et mondiales conduit à une approche régionale. Cela semble presque trop évident.

- **De deux algorithmes de positionnement précis RTK et PPP à l'hybridation PPP-RTK**

Le PPP-RTK utilise certains éléments et principes du RTK et du PPP. On doit néanmoins garder à l'esprit l'objectif : une technique proposant une précision centimétrique en un temps de convergence rapide.

Reprenons les équations d'observations GNSS. Elles sont non différenciées et non combinées (Teunissen, et al., 2010). Pour un signal à une fréquence j , pour un satellite s et un utilisateur u , l'équation liée à la phase est :

$$\varphi_{u,j}^s - \rho_u^s = \underbrace{(dt_u - dt^s) + \tau_u^s - \Delta_u^s + \lambda_j \delta_{u,j} - \lambda_j \delta_j^s}_{\text{PPP}} + \underbrace{\lambda_j N_{u,j}^s}_{\text{RTK}} \quad (2.1)$$

$R_{r,j}^s$ - Code du signal du satellite s et à la fréquence j .

$\varphi_{u,j}^s$ - Phase du signal de la porteuse du satellite s et à la fréquence j .

ρ_u^s - Distance récepteur satellite.

dt_u - Erreur d'horloge de l'utilisateur.

dt^s - Erreur d'horloge du satellite.

τ_u^s - Erreur troposphérique.

Δ_u^s - Erreur ionosphérique.

$\delta_{u,j}$ - Biais de phase du récepteur.

δ_j^s - Biais de phase du satellite (matériel).

λ_j - Longueur d'onde.

$N_{u,j}^s$ - Ambiguïté du signal.

Dans cette équation, il est possible d'identifier les éléments qui seront résolus par tout ou une partie des techniques de l'hybridation PPP-RTK. Les éléments à résoudre liés au PPP signifient qu'ils vont être résolus de manière indépendante grâce notamment aux données des stations permanentes du réseau CORS. A l'inverse on pourra résoudre les ambiguïtés grâce à une méthode proche de la différenciation utilisée en RTK. Toutefois il faut garder en tête que cette équation devra va être d'abord résolue d'un point de vue réseau. En effet, l'objectif n'est pas de calculer la position mais de générer des corrections qui seront envoyées aux utilisateurs, en accord avec la logique SSR du PPP. Ceci se fait grâce aux différentes données des stations de références du réseau NRTK. Cependant, les inconnues ne sont pas les mêmes dans les deux cas (RTK et PPP-RTK) car les objectifs et les modèles appliqués sont différents pour chaque paramètre.

Toutefois, elle fonctionne sur le même principe général que le RTK et le PPP : un réseau CORS collecte des données et permet de construire des corrections pour le calcul de la position. Nous allons voir comment dans ce qui suit.

2.1.1 Partie Réseau : première étape de l'hybridation

La partie réseau est le premier jalon de la construction de cette technique et un premier pas sur l'hybridation PPP et RTK. Elle se compose d'un réseau de CORS qui permettent de générer les corrections PPP-RTK comme en PPP. C'est pour cela qu'on qualifie parfois le PPP-RTK de technique PPP étendue.

Ce réseau de stations CORS dans une solution hybride PPP-RTK n'a pas la même fonctionnalité qu'en NRTK. Elle est proche de celle qui peut être possible dans une méthode PPP. Elle permet de fournir des données supplémentaires à l'utilisateur sous le format de corrections. L'idée est de résoudre les équations d'observations pour créer un vecteur contenant les corrections qui seront envoyées à l'utilisateur. La partie réseau a pour objectif de générer ces corrections PPP-RTK pour l'utilisateur. Le principal paramètre qui nécessite d'être résolu grâce au réseau est le biais de phase du satellite (SPB) dont on a parlé dans la section consacrée au PPP. En effet, le réseau fournit les orbites des satellites, les biais de phase et de code des satellites et les corrections atmosphériques ou encore les erreurs d'horloges des satellites (Zhang, et al.). Les biais de phase sont les erreurs qui rendent l'ambiguïté flottante dans le cadre du PPP et du PPP-RTK et qui complexifie le calcul de la position pour l'utilisateur. De plus, dans la partie réseau la principale inconnue est le biais de phase. Le réseau CORS est interconnecté et relié à des serveurs, il fonctionne comme en NRTK. La principale différence est qu'on considère la résolution par rapport à un simple récepteur indépendamment de là où il est situé (Teunissen, et al., 2010). Aussi, le réseau peut collecter des données comme : les positions des stations, les variations atmosphériques (troposphère et ionosphère), les erreurs d'horloges des récepteurs et des satellites et les orbites des satellites.

On établit donc les équations d'observations pour l'ensemble des stations de références du réseau CORS. Dans ce cadre, les équations utilisées sont supposées comme non différenciées et non combinées (Psychas, et al., 2020).

Pour un réseau de taille n , $r=1, \dots, n$ on a donc pour un satellite s le système suivant :

$$\begin{cases} \varphi_{r_1,j}^s = \rho_{r_1}^s + dt_{r_1} - dt^s + \tau_{r_1}^s + \lambda_j(\delta_{r_1,j} - \delta_{,j}^s) - \mu_j \Delta_{r_1}^s + \lambda_j N_{r_1,j}^s \\ \dots \\ \varphi_{r_n,j}^s = \rho_{r_n}^s + dt_{r_n} - dt^s + \tau_{r_n}^s + \lambda_j(\delta_{r_n,j} - \delta_{,j}^s) - \mu_j \Delta_{r_n}^s + \lambda_j N_{r_n,j}^s \end{cases} \quad (2.2)$$

Et de même les différentes stations du réseau ont en visibilité plusieurs satellites $s = s_1, \dots, s_k$ ainsi pour une station de référence r_1 :

$$\begin{cases} \varphi_{r_1,j}^{s_1} = \rho_{r_1}^{s_1} + dt_{r_1} - dt^{s_1} + \tau_{r_1}^{s_1} + \lambda_j(\delta_{r_1,j} - \delta_{,j}^{s_1}) - \mu_j \Delta_{r_1}^{s_1} + \lambda_j N_{r_1,j}^{s_1} \\ \dots \\ \varphi_{r_1,j}^{s_k} = \rho_{r_1}^{s_k} + dt_{r_1} - dt^{s_k} + \tau_{r_1}^{s_k} + \lambda_j(\delta_{r_1,j} - \delta_{,j}^{s_k}) - \mu_j \Delta_{r_1}^{s_k} + \lambda_j N_{r_1,j}^{s_k} \end{cases} \quad (2.3)$$

Pour chaque satellite et chaque station on cherche à déterminer $\rho_r^s, dt^s, \tau_r^s, \delta_{r_1,j}, \delta_{,j}^s, \Delta_r^s, N_{r,j}^s$.

L'objectif de la résolution est de connaître les biais de code de phase. Ce sont eux qui sont nécessaires pour la résolution d'un point de vue utilisateur. En effet, $\delta_{,j}^s$ est la principale inconnue que nous souhaitons déterminer grâce au réseau. Cela correspond aux biais matériels sur le satellite s . Les biais matériels sont dus aux imperfections et aux limitations physiques matérielles du satellite s . Ces biais existent aussi pour récepteurs GNSS. Ainsi, les biais induits par le matériel (récepteur ou satellite) entraîneront une dégradation de la précision de la solution de positionnement s'ils ne sont pas traités correctement. Ceci est particulièrement important pour le positionnement de haute précision avec plusieurs GNSS.

Il s'avère que les biais de code et de phase sont difficiles à estimer sous leur forme indifférenciée, car ils sont fortement corrélés avec d'autres termes, par exemple les erreurs d'horloge. Ainsi, seules les différences entre les biais peuvent être estimées directement à partir des observations de code et de phase. C'est pour cela que dans notre solution hybride explicitée dans la partie 3 nous ferons une différenciation entre les satellites. C'est aussi le cas dans la plupart des techniques PPP-RRTK.

En construisant, la résolution d'un point de vue réseau, on constate un défaut de rang dans le système. Dit autrement, nous avons trop d'inconnues à résoudre par rapport aux équations disponibles. En effet, nous avons $n * k$ équations alors que nous avons plus de $2 * n * k$ inconnues. Pour pallier cela, le principe de la S-theory est appliqué : cela revient à fixer certains paramètres et à en construire de nouveaux qui seront dépendants de ces constantes.

- **Les estimateurs pour définir certains paramètres, construction de la base S**

Pour savoir quels paramètres fixer pour résoudre le défaut de rang de manière efficace, des variantes du PPP-RTK sont construites (Zhang, et al., 2018). Ainsi, les paramètres des équations d'observations seront rendus dépendants les uns des autres et exprimés entre eux grâce aux données fournies par le réseau. Cela permet dans un second temps de construire les corrections PPP-RTK associées à cette base S. Il existe une multitude de bases possibles, sa constitution étant au cœur de la résolution PPP-RTK. Il semble pourtant que les performances ne dépendent pas forcément de la base utilisée (Teunissen, et al., 2015).

Dans ce qui suit, je vais détailler quelques exemples de bases S pour expliciter la résolution d'un point de vue réseau. Il existe au moins trois variantes du modèle de réseau PPP-RTK indifférencié et non combiné que l'on peut utiliser pour estimer les biais de phase des satellites (SPB) (Zhang, et al., 2018). Elles permettent de définir au mieux les paramètres en posant certaines hypothèses pour la base S. Des estimateurs existent pour identifier les défauts de rang et aider dans le choix des paramètres à fixer pour éliminer le défaut de rang. Elles se caractérisent par des exigences différentes en matière d'informations a priori sur les distances récepteur-satellite et les horloges des satellites.

Les trois principaux estimateurs existants et couramment utilisés sont :

- « Geometry-free » : indépendant de la géométrie, c'est-à-dire sans connaissance de la position des stations de référence.
- « Geometry-fixed » : on connaît la position des satellites et des stations du réseau CORS utilisé.
- « Geometry-plus-satellite » : avec géométrie et horloge satellite. On connaît la position des satellites, celles des stations du réseau CORS utilisé ainsi que les horloges des satellites.

On peut remarquer que ce qui différencie les estimateurs est un niveau de connaissance plus ou moins élevés des paramètres du problème. Cela a un impact sur la définition des biais de code des récepteurs et des satellites. En effet, de nouveaux termes vont apparaître dans les équations d'observation.

$$d_j^S = d_{IF}^S + \mu_j d_{GF}^S \quad (2.4)$$

$$d_{r,j} = d_{r,IF} + \mu_j d_{r,GF} \quad (2.5)$$

Avec $d_{IF}^S = \frac{\mu_2}{\mu_2 - \mu_1} d_{j_1}^S - \frac{\mu_1}{\mu_2 - \mu_1} d_{j_2}^S$, $d_{GF}^S = \frac{1}{\mu_2 - \mu_1} (d_{j_2}^S - d_{j_1}^S)$, $d_{r,IF} = \frac{\mu_2}{\mu_2 - \mu_1} d_{r,j_1} - \frac{\mu_1}{\mu_2 - \mu_1} d_{r,j_2}$
 et $d_{r,GF} = \frac{1}{\mu_2 - \mu_1} (d_{r,j_2} - d_{r,j_1})$

Cela met en avant l'apport de la modélisation « ionosphere free » (IF), soit sans erreur ionosphérique. En effet, on utilise la combinaison de deux fréquences pour réduire drastiquement l'influence de l'ionosphère. Ce qui explique que la plupart des applications PPP-RTK utilise le bi-fréquence.

Selon les informations dont on dispose sur la géométrie (geometry free ou geometry fixed) on aura des données différentes : dans certains cas, on prendra en compte certains biais d'horloge ou de phase et dans d'autres cas on aura aussi la position des stations du réseau CORS.

Le principe revient à fixer souvent un paramètre pour permettre la résolution générale des équations non différenciées du réseau CORS. La base est construite à partir de la matrice nulle du réseau CORS (Odijk et al. 2016, Annexe détaillée). On doit en passer par là pour supprimer le défaut de rang.

Voici une illustration de ce à quoi ressemble une matrice de l'espace nul dans un cas de PPP-RTK :

$$M_{nul} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ e_k \emptyset \left\{ \begin{pmatrix} 1 & 0 & 0 \\ 0 & I_f & 0 \\ 0 & 0 & I_f \end{pmatrix} X e_n \right\} & e_k X \left\{ \begin{pmatrix} -1 \\ \Lambda^{-1} e_f \\ e_f \end{pmatrix} X C_n \right\} & 0 & e_k X \left\{ \begin{pmatrix} 0 \\ I_f \\ 0 \end{pmatrix} X - C_n \right\} & 0 \\ e_k \emptyset \left\{ \begin{pmatrix} 1 & 0 & 0 \\ 0 & I_f & 0 \\ 0 & 0 & I_f \end{pmatrix} X e_m \right\} & 0 & e_k X \left\{ \begin{pmatrix} -1 \\ \Lambda^{-1} e_f \\ e_f \end{pmatrix} X I_m \right\} & 0 & e_k X \left\{ \begin{pmatrix} 0 \\ I_f \\ 0 \end{pmatrix} X I_m \right\} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_f X (C_n X e_m) & I_f X (e_n X I_m) \end{bmatrix}$$

Avec les définitions suivantes :

Symbol	Definition	Symbol	Definition
k	= # epochs	$e_{(\cdot)}$	= $(1, 1, \dots, 1)^T$
f	= # frequencies	$I_{(\cdot)}$	= $\text{diag}(1, 1, \dots, 1)$
n	= # receivers	$D_{(\cdot)}^T$	= $[-e_{(\cdot-1)} \quad I_{(\cdot-1)}]$
m	= # satellites	μ	= $(\mu_1, \mu_2, \dots, \mu_f)^T$
v	= # geometry comp.	Λ	= $\text{diag}(\lambda_1, \lambda_2, \dots, \lambda_f)$
$c_{(\cdot)}$	= $(1, 0, \dots, 0)^T$	$C_{(\cdot)}$	= $\begin{bmatrix} 0_{1 \times (\cdot-1)} \\ I_{(\cdot-1)} \end{bmatrix}$

La matrice de l'espace nul peut être considérée comme composée de cinq "lignes" multidimensionnelles et de cinq "colonnes" également multidimensionnelles. Chacune des "lignes" représente un groupe de paramètres correspondant aux variables du problème (soit "geo" pour la géométrie, "rec" pour le récepteur, "sat" pour les satellites, "ion" pour l'ionosphère et "amb" pour les ambiguïtés). Chacune des "colonnes" représente l'un des cinq types d'insuffisance de rang. Une façon d'identifier ces déficiences de rang est d'analyser quels changements de paramètres laissent les observations invariantes, comme cela a été fait pour les observables à double fréquence dans (Teunissen et al. (2010)). Par exemple, les horloges du récepteur et du satellite peuvent toutes deux être modifiées par une perturbation identique qui n'est pas "ressentie" par les observations de la phase et du code. Cette différence de rang (de taille 1) entre les horloges du récepteur et du satellite peut être éliminée en fixant l'horloge de l'un des récepteurs (base S), de sorte qu'il ne s'agit plus d'un paramètre inconnu.

Entre les paramètres dépendant du récepteur et ceux dépendant du satellite, étant donné les matrices de conception partielle pour les paramètres dépendant du récepteur et du satellite, on peut identifier deux types similaires d'insuffisances de classement.

La matrice de l'espace nul montre également que les lignes correspondant aux positions des récepteurs/ZTD ("1ère ligne") ainsi qu'aux retards ionosphériques verticaux ("4ième ligne") sont nulles, car ces paramètres ne sont impliqués dans aucune des lacunes de rang susmentionnées. Ces paramètres sont donc estimables (pour ce modèle basé sur ses hypothèses sous-jacentes) en mode absolu et non combiné. La taille totale de l'insuffisance de rang du modèle de réseau, en présence des contraintes temporelles sur tous les paramètres, peut être donnée comme la somme des insuffisances de rang identifiées.

La matrice de l'espace nul a pour objectif d'identifier les défauts de rang et est propre à chaque base S lorsque de l'application de la S-theory. Elle permettra aussi de construire les nouveaux paramètres estimés.

Dans le tableau suivant (Table 6), on expose différents exemples de bases S utilisées, ce sont les paramètres qu'on va transmettre à l'utilisateur via les corrections PPP-RTK pour calculer sa position.

Table 6 -Exemples de bases S associées à des corrections PPP-RTK

Paramètres fixés	Horloge	Paramètres estimables PPP-RTK	Source
$\delta_j^{s-1} dt^{s1}, \rho_{r_1}^s$	Commune (CC-S)	$[\rho_{r_1}^s; dt^s - dt^p; \delta_j^s - \delta_j^p]$	(Teunissen, et al., 2015)
$d_j^{s-1} z_{r_1,j}^s, \rho_{r_1}^s$	Distincte (DC-S)	$[\rho_{r_1}^s; d^s - d^p; z_{r_1,j}^s - z_{r_1,j}^p]$	(Zhang, et al., 2018)
$\overline{\delta_j^s}, \overline{dt^s}, \overline{z_{r,j}}$	Commune (CC-S)	$[\delta_j^s - \delta_j^p; d_j^s - d_j^p; dt^s - dt^p]$	(Odiijk, et al., 2016)

Avec $\bar{x} = \frac{1}{m} \sum_{s=1}^m x^s$

Dans le tableau ci-dessus on voit l'exemple de trois bases S différentes. Elles se partagent en deux catégories : celle avec horloge commune entre le réseau et l'utilisateur et celle avec horloges distinctes entre le réseau et l'utilisateur. On a donné une valeur « arbitraire » valable pour ces paramètres et ensuite, à partir de ces données, on redéfinit les paramètres estimables. Cela revient en fait à faire des différentiations entre la valeur du réel du paramètre et celle qu'on a fixé. On aura ainsi des paramètres estimables et on transmettra certains de ces paramètres à l'utilisateur via les corrections PPP-RTK.

Maintenant que nous connaissons les paramètres qui sont envoyés à l'utilisateur, je vais détailler leur impact sur les solutions PPP-RTK.

- **Biais de code et de phase**

On en a déjà parlé plus haut, mais le biais nécessaire à la résolution des ambiguïtés pour l'utilisateur est la connaissance du biais de phase. Ainsi, dans la plupart des corrections PPP-RTK, nous aurons au moins le biais de phase et de code. En effet, le réseau a pour objectif de fournir des données à l'utilisateur. Par exemple le réseau peut fournir les orbites des satellites, les biais de phase (SPB) des satellites et les corrections atmosphériques (Zhang, et al.). Les SPB sont les corrections d'erreurs qui permettent aux ambiguïtés de redevenir entières pour l'utilisateur et permettre ainsi leurs déterminations simplement. Les biais de phase et de code sont des données propres liées au satellite. Dans certains cas on parle de biais matériel. Ces biais dépendent donc du satellite auquel il se réfère et peuvent ainsi varier d'un satellite à l'autre.

C'est aussi un paramètre essentiel en PPP. Toutefois, en PPP il est difficile à estimer et limite la résolution des ambiguïtés. Il est donc important d'y porter une attention particulière. Ainsi, c'est donc un avantage en PPP-RTK : on a une meilleure estimation (en continuité) des SPB car nous avons fixé des ambiguïtés en tant que données connues.

- **Horloge des satellites**

Le PPP-RTK est une technique qui fonctionne avec un seul récepteur avec résolution d'ambiguïtés entières (Khodabandeh, et al., 2015). Définir les horloges des satellites permet de fournir un « positionnal link » (lien positionnel) entre l'utilisateur et une station de référence. Toutefois, c'est une estimation des horloges des satellites que nous calculons dans la partie réseau. Nous pouvons avoir ci-dessous (Figure 14) un aperçu de ces erreurs d'horloges sur plusieurs minutes.

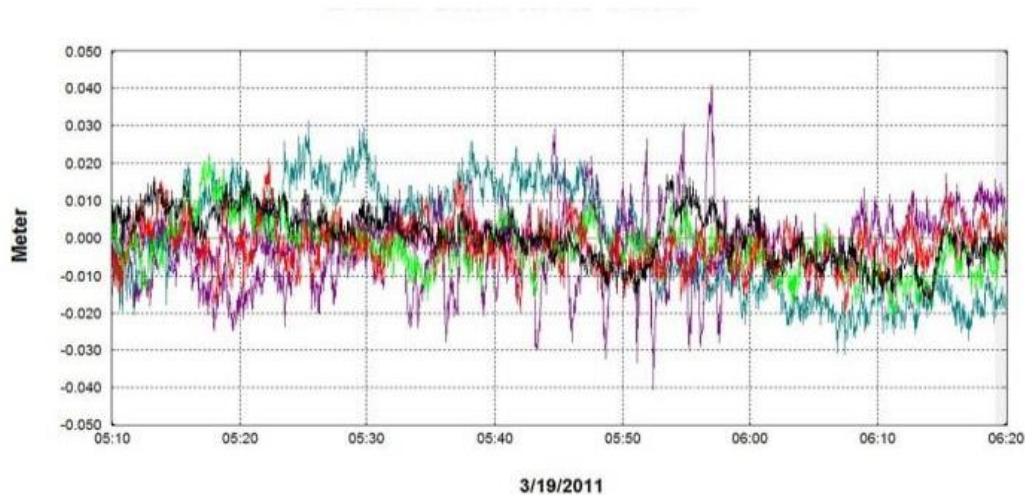


Figure 14- Corrections des horloges dans la technique RTX (PPP-RTK) avec Ionosphere free (Leandro, et al., 2011)

RTX est une méthode PPP-RTK développée par la société Trimble. On voit que les écarts des erreurs d'horloge ne sont pas très grands (quelques centimètres). Toutefois, cette erreur selon les satellites, à l'image des biais de phase, peut beaucoup varier. Toutefois, quand on fera la résolution d'un point de vue réseau, le calcul des erreurs d'horloges estimées sera plutôt facile car elle est assez stable dans le temps.

Contrairement aux orbites de satellites, les solutions donnant les horloges de satellites sont plus difficiles à comparer directement. En effet, ces différentes solutions d'horloges peuvent présenter des décalages entre elles et se comporter différemment en raison des différences dans le processus de réalisation du temps de référence GNSS et dans les approches de modélisation des observations. Cela dit, l'un des moyens de vérifier la qualité des horloges de satellite consiste à quantifier dans quelle mesure elles peuvent être utilisées pour modéliser les données d'observation réelles des récepteurs. En général, on peut y parvenir en appliquant la correction de l'orbite et de l'horloge du satellite aux données GNSS et en vérifiant les résidus restants. D'autres quantités, telles que les coordonnées du récepteur, doivent conserver leurs valeurs correctes pour que les résidus soient significatifs. Une remarque concernant ce type d'évaluation est que dans ce cas, ce n'est pas seulement la qualité de l'horloge du satellite qui est évaluée, mais l'erreur combinée de l'orbite du satellite et de l'horloge.

- Erreurs atmosphériques

Pour corriger les erreurs atmosphériques il est possible d'utiliser des données externes pour les informations ionosphériques et troposphériques (Zhang, et al., 2022). Ainsi, cela peut aussi faire partie des corrections PPP-RTK envoyées. En effet, les corrections ionosphériques peuvent être caractérisées par le réseau de stations CORS utilisé pour générer les corrections PPP-RTK. Plus le réseau sera dense plus le temps de convergence sera court et donc la résolution rapide (Psychas, et al., 2020). En effet, la résolution des ambiguïtés pourra se faire de manière plus rapide car les erreurs ionosphériques sont mieux connues. On peut voir l'impact direct du réseau sur la résolution des ambiguïtés dans la figure 15.

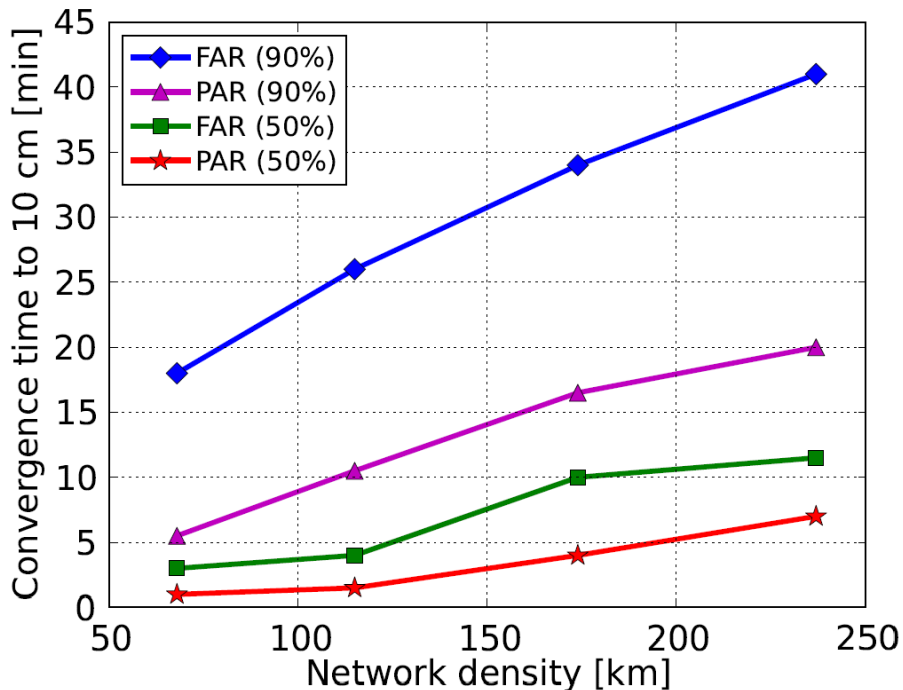


Figure 15 - Temps de convergence des erreurs de position horizontale à 10 cm en fonction de la densité du réseau pour FAR et PAR, sur la base de 50 % et 90 % des solutions de l'échantillon.

Discutons de chaque type de correction, en commençant par la troposphère.

Le retard troposphérique au zénith (ZTD – Zenith Tropospheric Delay), qui est multiplié par une fonction de cartographie dépendant de l'élévation, est l'une des inconnues typiques des équations d'observation du GNSS. La composante hydrostatique du retard troposphérique atteint environ 2,3 m dans la direction du zénith. Elle est principalement liée à la température et à la pression atmosphérique. Elle varie doucement et lentement dans le temps et peut être modélisée avec une précision de l'ordre du millimètre, voire mieux, sur la base des données météorologiques de surface. Dans la direction du zénith, la composante humide du retard troposphérique varie de quelques centimètres (ou moins) dans les régions arides à 35 cm dans les régions humides. Cette erreur a donc un impact non négligeable sur la position. Toutefois elle est facilement modélisable dans un réseau de stations. Il existe donc principalement deux options : soit corriger en amont de la résolution les ZTD ou alors fournir les corrections troposphériques dans les corrections PPP-RTK.

Pour les erreurs ionosphériques, on pourrait avoir les mêmes options : fournir les données ionosphériques à l'utilisateur ou les corriger directement dans les observations. Toutefois, il est plus difficile de modéliser l'ionosphère. En effet, les erreurs ont plus d'impact et connaissent des

variations plus importantes. Ainsi, une autre option est d'utiliser le modèle sans ionosphère. Cela est possible dans le cas où on utilise deux fréquences. Ainsi, grâce à la décomposition IF et GF présentée plus haut, on pourra facilement éliminer les erreurs liées à l'ionosphère par différenciation des équations entre deux fréquences.

2.1.2 Partie Utilisateur : Indépendance des paramètres et calcul d'une position en PPP-RTK

Comme nous l'avons vu plus haut, la technique PPP-RTK conserve un avantage considérable du PPP : la représentation en espace d'état (SSR). En effet, l'hybridation PPP-RTK a un fonctionnement en mode diffusion des corrections. Ainsi, à la différence du NRTK ici il n'y a pas de canal double de communication entre le réseau de stations et l'utilisateur. On est sur un fonctionnement de résolution mode « single receiver », c'est-à-dire récepteur en mode de positionnement autonome, comme en PPP (Teunissen, et al., 2010). De cette manière, les erreurs calculées et les observables mesurées sont indépendants.

Ainsi, nous diffusons les corrections PPP-RTK les plus communément utilisées dans la littérature ($\delta_{j,j}^s, dt^s, N_{r_1,j}^s$) à l'utilisateur et nous obtenons l'équation sur la phase suivante :

$$\varphi'_{u,j} = \varphi_{u,j}^s + dt^s + \lambda_j \delta_{j,j}^s - \lambda_j N_{r_1,j}^s \quad (2.6)$$

On peut donc réécrire cette équation ainsi :

$$\varphi'_{u,j} = \rho_u^s + (dt_u - dt^s) + \tau_u^s - \Delta_u^s + \lambda_j \delta_{u,j} - \lambda_j \delta_{j,j}^s + \lambda_j N_{u,j}^s + dt^s + \lambda_j \delta_{j,j}^s - \lambda_j N_{r_1,j}^s \quad (2.6)$$

$$\varphi'_{u,j} = \rho_u^s + dt_u + \tau_u^s - \Delta_u^s + \lambda_j \delta_{u,j} + \lambda_j N_{u,j}^s - \lambda_j N_{r_1,j}^s \quad (2.7).$$

Dans ce cas, c'est la distance satellite – récepteur qui est l'inconnue ρ_r^s . Il reste donc à déterminer pour l'utilisateur : $dt_u, \delta_{u,j}, \Delta_u^s, \tau_u^s, N_{u,j}^s$. Pour estimer ces paramètres il existe différentes solutions proposées dans la littérature :

- Différenciation entre deux satellites (Teunissen, et al., 2015) : cela permet d'éliminer les erreurs dépendantes de l'utilisateur seulement comme les erreurs d'horloge du récepteur ou le biais matériel du récepteur. De plus, suite à cette différenciation, les ambiguïtés sont ainsi double différenciées $N_{ur_1,j}^{sp} = (N_{u,j}^s - N_{r_1,j}^s) - (N_{u,j}^p - N_{r_1,j}^p)$. Ce choix est souvent fait dans la résolution PPP-RTK dès la partie réseau. En effet, nous aurons des corrections différenciées entre satellites. Cela ne change rien aux résultats mais simplifie les calculs (Odijk, et al., 2016).
- Différenciation entre deux fréquences : on aura donc les dépendances atmosphériques et les paramètres d'horloge qui seront éliminés. Cette méthode est utilisée pour appliquer le principe de « sans ionosphère » pour éliminer seulement ces erreurs atmosphériques grâce à l'utilisation de deux fréquences par satellites.
- Source de données externes : on peut utiliser des sources externes pour fournir certaines données comme les erreurs atmosphériques. Toutefois, dans le cas de l'ionosphère les modèles sont moins précis.

Voyons à présent les critères de performance.

2.2 Les critères de performances : une variabilité de solutions PPP-RTK

Des critères de performances ont été établis pour les solutions RTK et PPP. On peut maintenant établir ceux d'une hybridation PPP-RTK. Comme cette méthode est une extension du PPP, on construit ces critères par rapport au PPP mais avec en plus un apport important d'un réseau de station CORS comme en NRTK.

Les critères les plus importants dans l'hybridation PPP-RTK :

- Fréquence des corrections : il est possible d'individualiser les données envoyées à l'utilisateur, par type d'erreur en type de correction. Ainsi, les corrections atmosphériques seront envoyées à une fréquence différente de celles des corrections d'horloge (10 secondes contre 1 seconde)
- Géométrie et densité du réseau : le PPP-RTK utilise dans l'état de l'art des petits réseaux couvrant des zones plutôt locales voir régionales.
- TTFF : en moins de 15 minutes il est possible de fixer les ambiguïtés entières, plus rapidement que le PPP seul.
- Précision : qualifiée de centimétrique en PPP-RTK (<10cm)

Ce qui fait évoluer notre figure des critères de performance (figure 16).

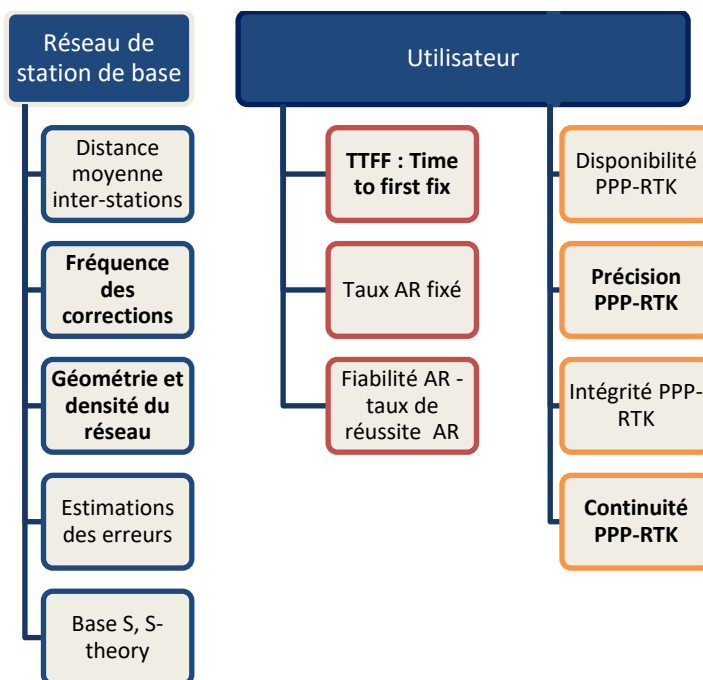


Figure 16- Paramètres et critères de performances PPP-RTK

Toutefois, l'utilisation du réseau en PPP-RTK peut facilement être définie comme une source de données externes. Pour construire les corrections associées PPP-RTK, il est nécessaire d'avoir recours à des estimateurs et ainsi les paramètres des équations non différenciées et non combinées des observations GNSS peuvent être exprimés de différentes manières (Zhang, et al., 2022).

Les deux principaux critères étudiés comme en RTK et PPP sont la précision et le TTFF. Comme le RTK, l'ajout de nouvelles constellations de satellites ou de plusieurs fréquences pourra améliorer la précision et la résolution des ambiguïtés (Ma, et al., 2020). Les résultats sont similaires à ceux explicités précédemment en PPP.

2.2.1 Time-to-first-fix (TTFF) en PPP-RTK

Nous allons commencer par regarder le temps de convergence des techniques PPP-RTK. On l'a vu plus haut, ce TTFF est lié à la résolution des ambiguïtés : plus vite les ambiguïtés seront résolues, meilleur il sera. Dans une première étude PPP-RTK menée dans (Zhang, et al., 2018) nous allons regarder les erreurs de positions en lien avec le Time-to-first-fix (TTFF). Dans ces tests, il y a deux groupes de récepteurs : 28 jouant le rôle de réseau et le reste comme récepteurs utilisateurs. La majorité des récepteurs de chaque groupe sont des récepteurs GPS TRIMBLE. Dans ce cas, seuls les satellites GPS à double fréquence sont utilisés. Le filtre de Kalman utilisé est réinitialisé toutes les 4 heures permettant d'étudier le comportement de convergence de la solution PPP-RTK.

Nous voyons sur la figure 17 un premier résultat de résolution PPP-RTK. Cette figure est composée de quatre panneaux) montrant les résultats horizontaux (a, flottant ; b. fixe) et les résultats verticaux (c, flottant ; d. fixe), respectivement. Nous définissons ici le temps de convergence comme la période minimale nécessaire pour atteindre une erreur de positionnement absolue inférieure à 10 cm. Les résultats flottants montrent que les temps de convergence de la composante horizontale (verticale) de 50 %, 75 % et 90 % des utilisateurs sont respectivement de 29 (20), 57 (29) et 67 (47) minutes. Dans le cas d'une ambiguïté fixée, nous constatons que l'erreur absolue horizontale (verticale) de 75 % des utilisateurs est inférieure à 10 cm après 25 (17) minutes, tandis que celle de 90 % des utilisateurs atteint 10 cm après 38 (30) minutes. Il est intéressant de noter que la composante verticale converge plus rapidement que la composante horizontale.

Un premier coup d'œil à cette figure donne l'impression que les résultats fixes, représentés en lignes rouges, sont disponibles environ 30 minutes après l'instant où le filtre est lancé ; en d'autres termes, le temps nécessaire pour résoudre les ambiguïtés est de l'ordre de quelques dizaines de minutes dans ce cas. En revanche, nous constatons qu'il faut environ une à deux heures pour que les résultats du flotteur (représentés par des lignes non rouges) se situent dans la fourchette de -10 à 10 cm. L'importance de la différence entre les résultats fixes et flottants est évidente au début de la série temporelle, mais diminue progressivement au fil du temps ; ce n'est pas inattendu, puisque les ambiguïtés flottantes filtrées deviennent plus précises au fil du temps.

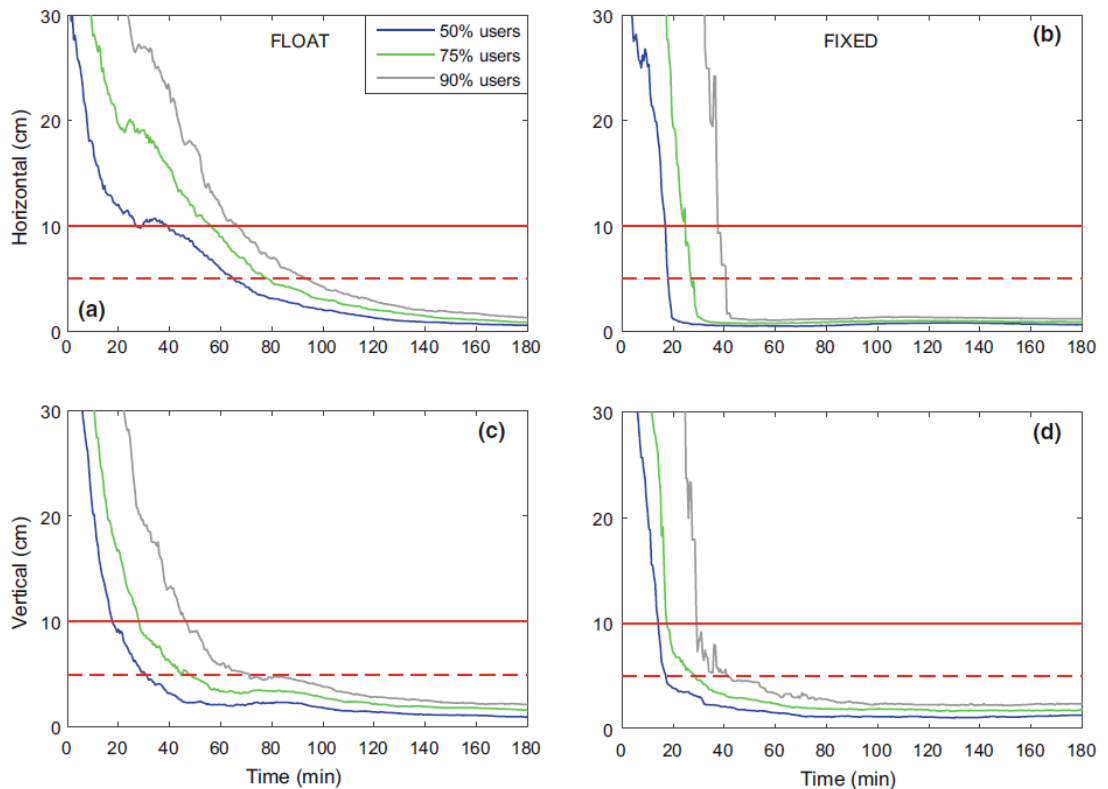


Figure 17- Erreurs de position statiques et absolues de 216 utilisateurs de PPP-RTK en fonction de l'époque d'observation (Zhang, et al., 2018)

On constate ainsi que la résolution rapide des ambiguïtés est importante, il s'agit d'un important « goulot d'étranglement » du calcul de la position. L'amélioration du temps de convergence doit donc s'intéresser de près à la résolution des ambiguïtés. En outre, une résolution réussie des ambiguïtés (b et d sur la figure), se traduit également par une diminution frappante des erreurs de positionnement au début de la convergence ou peu après. La figure 17 montre clairement que le pourcentage à 90 % chute brusquement de 10 cm à environ 2 cm, puis reste pratiquement constant au fur et à mesure que le résultat du calcul s'affine.

Ainsi, en regardant un autre exemple de solution PPP-RTK : la technique RTX développé par Trimble (Doucet, et al., 2012).

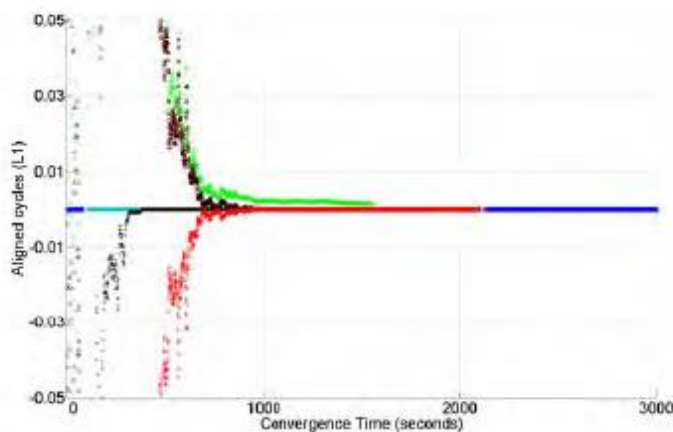


Figure 18- Convergence des ambiguïtés calculées avec RTX (Doucet, et al., 2012)

La figure 18 présente l'« aligne du cycle » sur L1, ce qui correspond pour la fréquence L1 à la résolution de l'ambiguïté. On voit qu'au bout de 1000 secondes, les ambiguïtés sont fixées. Ainsi, au bout de 16 min nous avons un premier résultat de position acceptable : en général la précision de moins de 10 cm sera atteinte à ce moment-là. Cela montre la solidité de la technique RTX qui utilisent des ambiguïtés entières.

Similairement à cette approche, en PPP-RTK nous aurons un temps de convergence entre 12 et 20 minutes (Odijk, et al., 2016). Cela reste meilleur qu'en PPP où le temps de convergence est entre 20 et 30 minutes. Toutefois, en PPP-AR le processus de résolutions des ambiguïtés est accélérée en les définissant comme entières. Cela se rapproche de ce qui est fait en PPP-RTK. Le PPP-AR a un fonctionnement proche du PPP-RTK et de RTX à la différence du réseau de stations CORS utilisé (densité, données diffusées). Ainsi, nous aurons dans ce cas de PPP un temps de convergence entre 10 et 15 minutes (Zhang, et al., 2022). Cela reste similaire à ce qui peut se faire en PPP-RTK.

2.2.2 Précision en PPP-RTK

Comme évoqué dans la partie précédente, le TTFF n'a pas beaucoup de signification si on ne le lie pas au critère de précision. Ce critère de précision subit d'autres influences dont nous allons parler.

- **Influence du nombre de stations du réseau**

Dans un premier temps regardons, sur la figure 19, les résultats de précision PPP-RTK sur un réseau CORS en Australie (Odijk, et al., 2014). Deux récepteurs sont utilisés pour tester le PPP-RTK avec ce réseau CORS : high-grade avec un récepteur TRIMBLE et low-grade avec un récepteur u-blox.

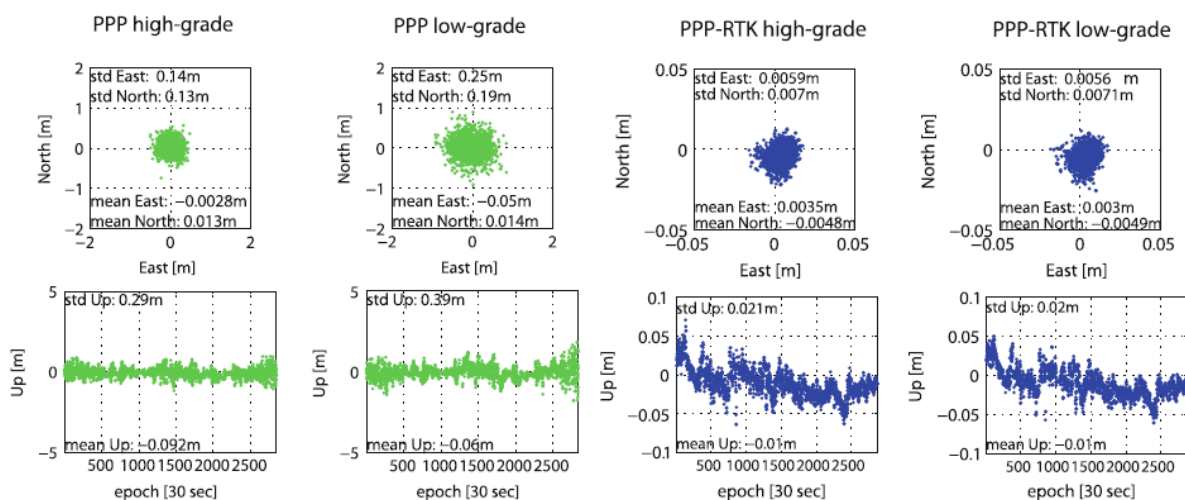


Figure 19- Résultats des positionnements PPP et PPP-RTK basés sur les données de phase et de code GPS mono-fréquence corrigées par le réseau CORS. (Odijk, et al., 2014)

Les deux graphiques de gauche (en haut : dispersion de la position nord-est ; en bas : série chronologique de la position vers le haut) correspondent au PPP basé sur le récepteur de haute qualité, tandis que les deux graphiques en deuxième position à partir de la gauche décrivent les contreparties PPP obtenues en utilisant les données du récepteur de basse qualité. Les deux colonnes de droite sont celles qui correspondent à l'approche PPP-RTK.

A titre de comparaison, les deux colonnes de gauche de la figure 19 montrent les résultats de positionnement qui auraient été obtenus sans résolution de l'ambiguïté des nombres entiers, c'est-à-dire le "PPP régional à fréquence unique", incluant donc les corrections ionosphériques interpolées, mais sans les données de phase corrigées pour les biais de phase du satellite (SPB). Afin de permettre une comparaison équitable avec le PPP-RTK, les résultats PPP présentés sont basés sur le même nombre d'époques que celui nécessaire pour la résolution d'ambiguïté du PPP-RTK. Il convient de souligner qu'en dehors des corrections de biais de phase des satellites, les hypothèses sous-jacentes aux solutions PPP et PPP-RTK sont exactement les mêmes (c'est-à-dire modèle stochastique de phase et de code, positions initiales des récepteurs à partir d'un positionnement standard à point unique, nombre de satellites, etc.)

La comparaison montre clairement l'impact de la résolution des ambiguïtés sur les nombres entiers : la précision horizontale est améliorée de quelques dm (PPP) à des valeurs inférieures au cm (PPP-RTK), tandis que la composante verticale est améliorée de 3-4 dm (PPP) à environ 2 cm (PPP-RTK). La différence de précision entre PPP high-grade et PPP low-grade est principalement due à la qualité inférieure des données de code pour le récepteur u-blox (basse qualité) par rapport au récepteur Trimble (haute qualité).

Toutefois, ce très bon résultat de précision dans ce cas PPP-RTK ne représente pas la tendance générale observée en termes de résultats PPP-RTK. En effet, la précision en PPP-RTK sera plutôt autour de 10 cm. Sur la figure 20, voici un autre exemple de technique PPP-RTK avec un réseau CORS à Hong Kong (Teunissen, et al., 2010) :

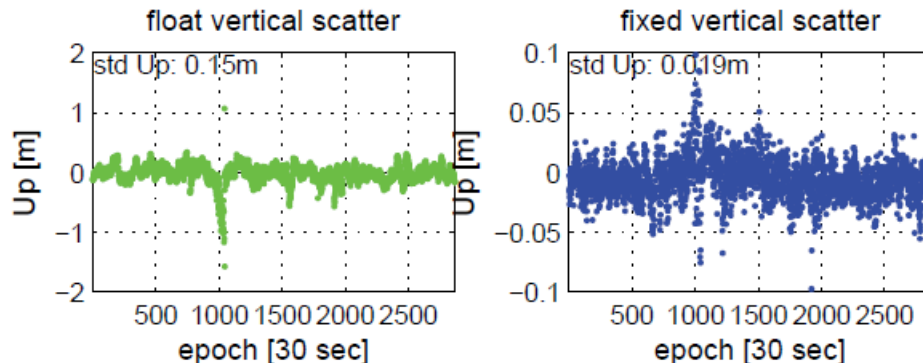


Figure 20- Erreurs sur position verticale dans une technique PPP-RTK (Teunissen, et al., 2010)

Dans la littérature sur le PPP-RTK, la précision de la résolution PPP-RTK est autour de 10 cm. Pour améliorer ce résultat il existe des variantes, comme en PPP : multiplier le nombre de satellites ou alors multiplier le nombre de stations dans le réseau CORS PPP-RTK.

D'après l'article (Khodabandeh, et al., 2015), le nombre d'époques (k) et le nombre de stations (n) fonctionnent en tandem, c'est-à-dire qu'ils contribuent de la même manière à l'amélioration de la précision des corrections.

De même, le nombre de stations a maintenant le même impact que le nombre d'époques. L'impact du nombre de stations diminue avec l'augmentation du nombre d'époques et, inversement, l'impact

du nombre d'époques diminue avec l'augmentation du nombre de stations. Cela est bien illustré dans la figure 21 :

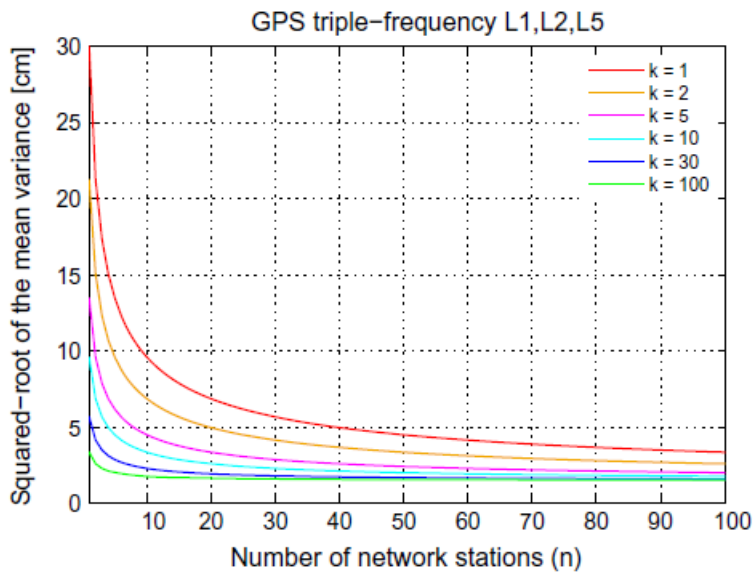


Figure 21 -Précisions de la technique PPP-RTK en fonction du nombre d'époques et de stations dans le réseau CORS

Par conséquent, avoir plus de stations dans un réseau CORS PPP-RTK permettrait de réduire le nombre d'époques et ainsi d'accélérer la résolution en ayant un temps de convergence nécessaire réduit.

- **Influence du nombre de satellites, multi-constellations**

Une analyse similaire est possible quand on multiplie le nombre de satellites. Comme en PPP, nous voyons sur la figure ci-dessous que le nombre de satellites en visibilité varie dans le temps. Ainsi, dans la résolution nous aurons un nombre de satellites qui changent en permanence (Khodabandeh, et al., 2015) :

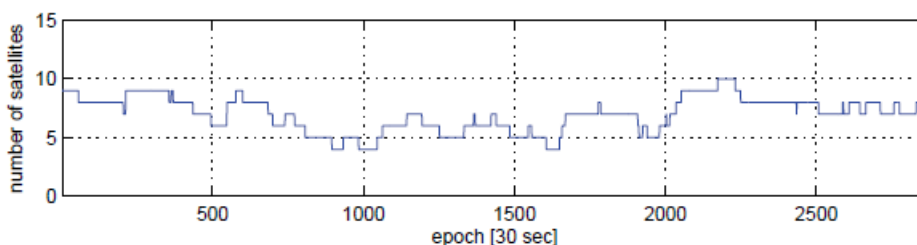


Figure 22- Variation du nombre de satellites sur une longue durée d'observations

Cette variation implique des changements dans la résolution PPP-RTK sur des longs temps d'observations. Regardons maintenant l'impact sur les performances.

La figure 23 (Ma, et al., 2020) illustre les performances globales des solutions d'ambiguïtés fixes et flottantes avec une probabilité de 90 %. Cela signifie que 90 % des erreurs de positionnement sont inférieures à celles indiquées dans la figure. La raison pour laquelle toutes les solutions de convergence ne sont pas prises en compte est d'éviter les erreurs de modèle inattendues qui pourraient entraîner un comportement de positionnement inhabituel lors du traitement des données. Dans cette figure (G correspond à la constellation GPS, E correspond à Galileo, C correspond à Beidou et R à la constellation Glonass).

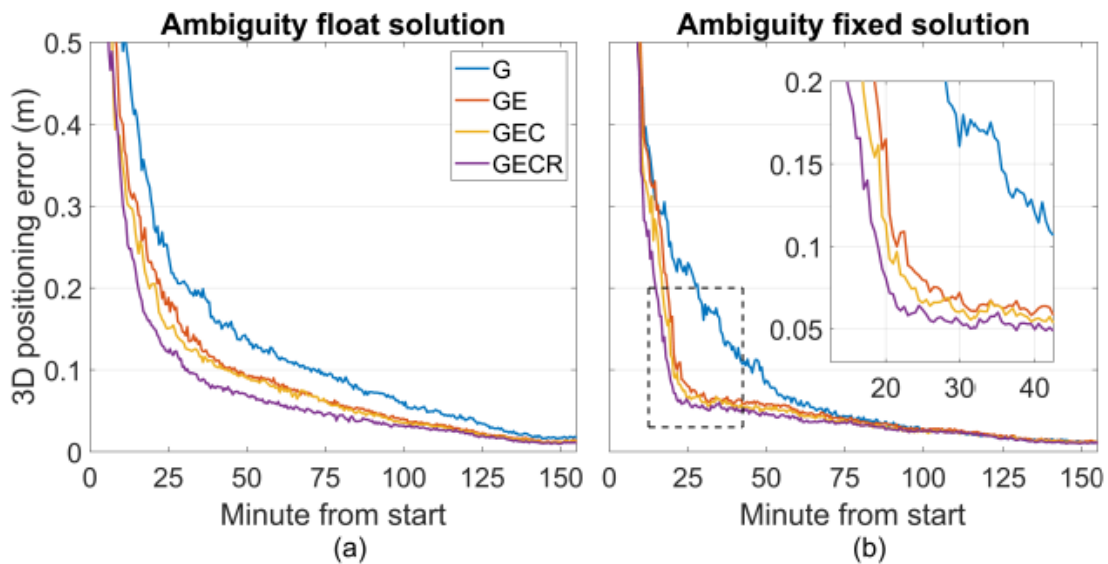


Figure 23- Erreurs de la position en fonction du nombre de satellites et du temps (Ma, et al., 2020)

On peut constater que, contrairement à la série temporelle des erreurs de positionnement liées à l'ambiguïté flottante qui prend la forme d'une courbe, les erreurs liées à l'ambiguïté fixe diminuent rapidement une fois que la plupart des ambiguïtés ont été résolues avec succès, ce qui est compatible avec ce qu'on a vu précédemment. A noter le cas du GPS seul qui, en raison du problème de mauvaise résolution des ambiguïtés, a du mal à converger plus rapidement qu'en float. Les solutions fixes d'ambiguïté du GPS seules sont principalement affectées entre la 1700e et la 2200e époque.

En ce qui concerne les combinaisons GE, GEC et GEGR, au contraire, les erreurs de positionnement fixées par ambiguïté diminuent de manière significative entre la 15e et la 20e minute. Cela démontre encore une fois que la résolution des ambiguïtés est la clé d'un positionnement GNSS rapide et de haute précision. En outre, il convient de noter qu'après un long temps de convergence, les solutions à ambiguïté flottante peuvent atteindre un niveau de précision similaire à celui des solutions à ambiguïté fixe comme on l'avait déjà observé à la figure 18. Ce qui est également valable pour le cas multi-constellation et GPS seul, bien que ce dernier souffre d'un problème de résolution des ambiguïtés entières. Nous pouvons donc conclure que l'un des objectifs de la correction des ambiguïtés est identique à celui du multi-GNSS (multi-constellations), à savoir la réduction du temps de convergence. Cela aura ainsi un effet d'amélioration sur la précision.

En conclusion de cette deuxième partie, on peut dire que la solution hybride PPP-RTK repose toujours sur les mêmes principes du RTK et du PPP. En effet, elle s'appuie sur un réseau CORS comme en NRTK et la résolution se fait de manière quasi absolue comme en PPP d'un point de vue de l'utilisateur.

En termes de performance, les précisions des techniques PPP-RTK semblent du même ordre de grandeur qu'en PPP autour du décimètre. Toutefois, selon les paramètres fixés ou l'utilisation du modèle sans ionosphère par exemple, il sera possible de les améliorer jusqu'à un temps de convergence proche de 10 minutes avec une précision inférieure à 10 centimètres, très similaire aux résultats du NRTK. On a vu qu'un troisième paramètre joue un rôle essentiel dans la vitesse de convergence : le taux de résolution des ambiguïtés. En effet, la façon dont les ambiguïtés sont résolues aura un impact direct sur le temps de convergence, moins sur la précision (à condition d'avoir le temps).

Dans tous les cas, que ça soit en RTK, NRTK, PPP ou PPP-NRTK le TTFF, la précision et le taux AR sont liés. En effet, la performance de l'un va avoir un impact sur l'autre : avoir un temps de convergence long avec une précision de l'ordre du centimètre n'est pas forcément le plus optimal. Et inversement si on fixe rapidement des ambiguïtés de mauvaise qualité nous n'aurons pas forcément une bonne précision.

A présent que nous avons balayé les problématiques des approches actuelles de l'hybridation du PPP et RTK, je vais présenter l'approche que je propose dans la partie qui suit.

3. Une hybridation PPP-RTK et NRTK – Construction de ma proposition de solution hybride

Dans cette partie, je vais construire ma technique PPP-NRTK et vous présenter le lien d'hybridation entre le PPP-RTK et le NRTK. Nous allons la séquencer en trois étapes :

1. Génération des corrections PPP-RTK grâce à un réseau CORS.
2. Initialisation de la position de l'utilisateur en NRTK.
3. Calcul de la nouvelle position de l'utilisateur grâce au NRTK et aux corrections PPP-RTK.

Pour cela je vais la présenter de deux manières différentes : une partie strictement mathématique et une partie algorithmique. L'idée est de montrer comment à partir des techniques PPP et NRTK, j'ai pu construire cette solution et aussi de démontrer l'intérêt d'hybridation telle que je la propose. Nous construirons la chaîne logicielle associée au fur et à mesure du développement mathématique. Nous commencerons par le point de vue du réseau de stations CORS.

3.1 Partie réseau : un réseau CORS pour un système d'équations d'observations GNSS

3.1.1 Système complet d'équations du réseau

Pour établir notre solution PPP-NRTK, la première étape est d'établir les équations d'observations pour le réseau de stations de bases. On suppose qu'on a un réseau composé de n stations qui ont toutes en visibilité k satellites.

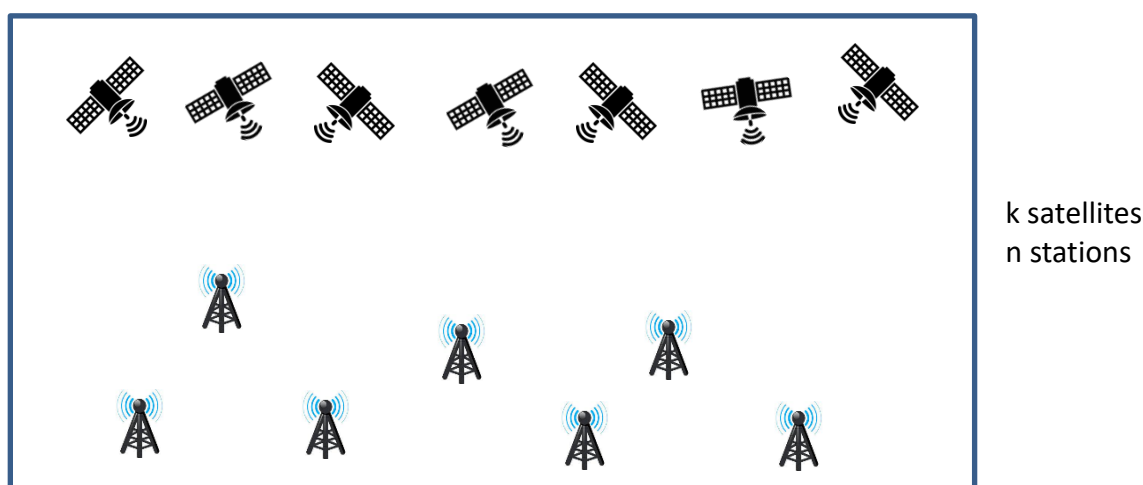


Figure 24- Réseau CORS interconnecté avec k satellites en visibilité

On peut établir pour une fréquence j , un satellite s et une station r , les équations d'observations suivantes, comme nous les avons définies dans l'introduction :

$$\begin{cases} R_{r,j}^s - \tau_r^s = \rho_r^s + dt_r + d_{r,j} - dt^s - d_{j,j}^s + \mu_j \Delta_r^s \\ \varphi_{r,j}^s - \tau_r^s = \rho_r^s + dt_r - dt^s + \lambda_j (\delta_{r,j} - \delta_{j,j}^s) - \mu_j \Delta_r^s + \lambda_j N_{r,j}^s \end{cases} \quad (3.1) \text{ avec } \mu_j = \frac{f_{j_1}^2}{f_j^2} \text{ et } j = j_1, \dots, j_p$$

- $R_{r,j}^s$ - code du signal du satellite s et à la fréquence j
- $\varphi_{r,j}^s$ - phase du signal de la porteuse du satellite s et à la fréquence j
- ρ_r^s - distance récepteur satellite
- dt_r - erreur d'horloge du récepteur
- dt^s - erreur d'horloge du satellite
- τ_r^s - erreur troposphérique
- Δ_r^s - erreur ionosphérique
- $d_{j,j}^s$ - biais de code du satellite (matériel)
- $\delta_{j,j}^s$ - biais de phase du satellite (matériel)
- λ_j - longueur d'onde
- $N_{r,j}^s$ - ambiguïté du signal

On pose une première hypothèse où l'on considère que les observations sont corrigées pour la troposphère en utilisant les données des ZTD. Ainsi le paramètre τ_r^s est supposé connu. En effet, les erreurs troposphériques dans un tel réseau sont assez faciles à gérer, nous l'avons vu à la section 2.1.1 du précédent chapitre. De plus le réseau CORS composé de N stations a une géométrie connue. On peut ainsi calculer la position de chacune des stations. Dans la partie réseau, l'équation de phase peut donc être réécrite ainsi :

$$\begin{cases} R_{r,j}^s - \rho_r^s = dt_r + d_{r,j} - dt^s - d_{j,j}^s + \mu_j \Delta_r^s \\ \varphi_{r,j}^s - \rho_r^s = dt_r - dt^s + \lambda_j (\delta_{r,j} - \delta_{j,j}^s) - \mu_j \Delta_r^s + \lambda_j N_{r,j}^s \end{cases} \quad (3.2)$$

Connaissant l'erreur troposphérique, on a :

$$\begin{cases} R_{r,j}^s = R_{r,j}^s - \tau_r^s \\ \varphi_{r,j}^s = \varphi_{r,j}^s - \tau_r^s \end{cases}$$

L'objectif du réseau est de fournir des données à l'utilisateur pour calculer sa position. Ainsi, les inconnues à déterminer dans l'équation ci-dessus sont seulement les erreurs liées aux satellites telles que les erreurs d'horloge satellites, les biais matériels de phase et de code du satellite et les erreurs ionosphériques (en vert dans 3.2).

Ainsi, nous avons un réseau GNSS multifréquence et indifférencié des équations d'observation de la phase et du code.

- **Modélisation double fréquence : décomposition « ionosphere-free » et « geometry-fixed »**

Dans le cadre de notre résolution, nous utiliserons les signaux GNSS des deux constellations satellites (GPS et Galileo) et deux fréquences différentes pour chaque satellite. On a donc $p=2$ et on pose $j = j_1, j_2$.

Table 7- Fréquences utilisées sur les deux constellations GPS et Galileo

	GPS	GPS	Galileo
j_1	L1	L1	E1
j_2	L2	L5	E5

On applique les équations d'observations sur deux fréquences j_1 et j_2 issues du même satellite, ainsi :

$$\begin{cases} R'_{r,j_1} - \rho_r^s = dt_r + d_{r,j_1} - dt^s - d_{j_1}^s + \Delta_r^s \\ R'_{r,j_2} - \rho_r^s = dt_r + d_{r,j_2} - dt^s - d_{j_2}^s + \mu_2 \Delta_r^s \\ \varphi'_{r,j_1} - \rho_r^s = dt_r - dt^s + \lambda_{j_1} (\delta_{r,j_1} - \delta_{j_1}^s) - \Delta_r^s + \lambda_{j_1} N_{r,j_1}^s \\ \varphi'_{r,j_2} - \rho_r^s = dt_r - dt^s + \lambda_{j_2} (\delta_{r,j_2} - \delta_{j_2}^s) - \mu_2 \Delta_r^s + \lambda_{j_2} N_{r,j_2}^s \end{cases} \quad (3.3)$$

Avec $\mu_1 = 1$ et $\mu_2 = \frac{f_{j_1}^2}{f_{j_2}^2}$

On peut réécrire ces équations de manière plus compacte :

$$\begin{cases} \Omega_r^s - e\rho_r^s = edt_r - edt^s + d_r - d^s + \mu\Delta_r^s \\ \Phi_r^s - e\rho_r^s = edt_r - edt^s + \Lambda(\delta_r - \delta^s + N_r^s) - \mu\Delta_r^s \end{cases} \quad (3.4)$$

Avec $\Omega_r^s = \begin{pmatrix} R'_{r,j_1} \\ R'_{r,j_2} \end{pmatrix}$, $\Phi_r^s = \begin{pmatrix} \varphi'_{r,j_1} \\ \varphi'_{r,j_2} \end{pmatrix}$, $d_r = \begin{pmatrix} d_{r,j_1} \\ d_{r,j_2} \end{pmatrix}$, $d^s = \begin{pmatrix} d_{j_1}^s \\ d_{j_2}^s \end{pmatrix}$, $\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$, $e = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, $\Lambda = \text{diag}(\lambda_{j_1}, \lambda_{j_2})$, $\delta_r = \begin{pmatrix} \delta_{r,j_1} \\ \delta_{r,j_2} \end{pmatrix}$, $\delta^s = \begin{pmatrix} \delta_{j_1}^s \\ \delta_{j_2}^s \end{pmatrix}$ et $N_r^s = \begin{pmatrix} N_{r,j_1}^s \\ N_{r,j_2}^s \end{pmatrix}$

Utiliser deux fréquences permet d'utiliser la décomposition geometry-fixed (GF) et ionosphere-free (IF). Cette décomposition a deux utilités : pouvoir identifier plus facilement les défauts de rang et définir les biais de phase matériels du satellite δ^s (Odijk, et al., 2016). En effet, les biais de phase des satellites sont les principales limites pour l'utilisateur pour converger rapidement et résoudre les ambiguïtés, comme nous l'avons exposé dans la partie 2. De plus, utiliser le variant GF signifie que les distances satellites-récepteurs sont connues. Ce qui est vrai en pratique puisque l'on sait où sont les stations de bases de notre réseau et que l'on connaît également la position des satellites grâce aux éphémérides. Les distances s'en déduisent aisément.

Ainsi on définit d_{GF} et d_{IF} :

$$d^s = \begin{pmatrix} \mu_1 & 1 \\ \mu_2 & 1 \end{pmatrix} \begin{pmatrix} d_{GF}^s \\ d_{IF}^s \end{pmatrix} \quad (3.5)$$

$$d_r = \begin{pmatrix} \mu_1 & 1 \\ \mu_2 & 1 \end{pmatrix} \begin{pmatrix} d_{r,GF} \\ d_{r,IF} \end{pmatrix} \quad (3.6)$$

Avec $d_{IF}^s = \frac{\mu_2}{\mu_2 - \mu_1} d_{j_1}^s - \frac{\mu_1}{\mu_2 - \mu_1} d_{j_2}^s$, $d_{GF}^s = \frac{1}{\mu_2 - \mu_1} (d_{j_2}^s - d_{j_1}^s)$, $d_{r,IF} = \frac{\mu_2}{\mu_2 - \mu_1} d_{r,j_1} - \frac{\mu_1}{\mu_2 - \mu_1} d_{r,j_2}$
 et $d_{r,GF} = \frac{1}{\mu_2 - \mu_1} (d_{r,j_2} - d_{r,j_1})$

Cela nous permettra dans un second temps de définir des équations différenciées entre les fréquences pour éliminer les erreurs ionosphériques. Nous établirons ces nouvelles équations au moment de générer les corrections PPP-RTK.

- **Décomposition des équations d'observations**

Ensuite, il est nécessaire d'identifier les potentiels défauts de rang qui empêche de former la matrice de base de l'espace nul, nécessaire à l'application de la théorie du système S au modèle GNSS indifférencié vu dans la section précédente (Teunissen, et al., 2015). Il est important de définir clairement les paramètres connus et inconnus dans nos équations (3.2) qui sont déjà linéaires.

Toutefois, avant de déterminer les défauts de rang, on doit dériver par rapport au temps le modèle de réseau GNSS non différencié de rang complet. Ainsi, on considère un système d'équations de rang complet. Pour ce faire, nous supposons initialement que les contraintes temporelles sur tous les paramètres sont incorporées. Les ambiguïtés sont supposées être entièrement constantes dans le temps, ce qui est vrai tant qu'il n'y a pas de saut de cycle (Odijk, et al., 2016).

Nous réécrivons donc nos équations d'observations en ajoutant l'indicateur temporel « i » : tous les paramètres dépendent du temps sauf les ambiguïtés :

$$\begin{cases} \Omega_r^s(i) - e\rho_r^s(i) = edt_r(i) - edt^s(i) + d_r(i) - d^s(i) + \mu\Delta_r^s(i) \\ \Phi_r^s(i) - e\rho_r^s(i) = edt_r(i) - edt^s(i) + \Lambda(\delta_r(i) - \delta^s(i) + N_r^s) - \mu\Delta_r^s(i) \end{cases} \quad (3.7)$$

L'idée d'imposer des contraintes temporelles aux paramètres est liée à l'utilisation d'un filtre de Kalman pour lequel les paramètres, agissant comme des vecteurs d'état, sont liés aux observations dans le temps. Pour cela, on réécrit le système sous la forme suivante :

$$\begin{pmatrix} E(\Delta y) \\ 0 \end{pmatrix} = \begin{pmatrix} A \\ B \end{pmatrix} E(\Delta x) \quad (3.8)$$

Δy correspond aux inconnues de nos équations (3.7) soit l'horloge des satellites, des biais de phase et de code des satellites.

Δx correspond aux paramètres que nous supposerons comme connus, soit les paramètres dépendant des récepteurs et la mesure de phase et de code.

$[A^T, B^T]^T$ est la matrice dite de conception des lots, dictant le défaut de rang qui dépend du filtre. Toutefois, comme explicité précédemment, les ambiguïtés sont considérées comme constantes par rapport au temps, on doit donc les isoler. Ainsi, la matrice de conception devient :

$$\begin{pmatrix} A \\ B \end{pmatrix} \rightarrow \begin{pmatrix} A & A_z \\ B & 0 \end{pmatrix} \text{ avec } A_z \text{ correspondant à la matrice de conception des ambiguïtés qui ne dépend pas du temps.}$$

Ensuite, pour établir la matrice de conception du défaut de rang du réseau nous utilisons l'ordre suivant des observables : observables de phase, observables de code, contraintes de marche aléatoire. Toutes les observables de phase et de code (double fréquence) de la première époque sont d'abord collectées, puis de la deuxième époque, etc. jusqu'à la dernière époque. Pour chaque époque, toutes les observables de la première fréquence apparaissent d'abord, puis celles de la deuxième fréquence. Pour les deux fréquences utilisées, l'ordre est le suivant : d'abord les observables du premier récepteur, puis celles du deuxième récepteur, et ainsi de suite. Enfin, pour chaque récepteur, l'ordre est d'abord les observables du premier satellite, puis du second, etc. Une

fois que toutes les données de phase et de code multi-époques sont stockées dans l'ordre ci-dessus, les contraintes de marche aléatoire pour les paramètres géométriques sont collectées, suivies par celles correspondant aux horloges des récepteurs et des biais matériels de phase et de code des récepteurs. Les contraintes de marche aléatoire pour les horloges des satellites et les biais matériels de phase et de code des satellites sont ensuite collectées, et enfin les contraintes de marche aléatoire pour les retards ionosphériques verticaux sont collectées.

- **Défaut de rang**

Nous allons donc examiner chaque paramètre du système d'équations (3.7) pour définir la matrice d'espace nulle et identifier le défaut de rang. Pour cela, on fait varier dans le temps les différents paramètres de l'équation (3.7) et on repère les cas où les observations restent invariantes.

D'un point de vue réseau, les inconnues dans ce système d'équations sont : dt_r , d_r , dt^s , d^s , δ_r , δ^s .

- Les horloges dt_r et dt^s (défaut de rang a)

Les horloges du récepteur et du satellite peuvent toutes deux être modifiées par une perturbation identique qui n'est pas "ressentie" par les observations de la phase et du code. Cela correspond à une différence de rang entre les horloges du récepteur et du satellite qui peut être éliminée en fixant l'horloge de l'un des récepteurs (base S), de sorte qu'il ne s'agisse plus d'un paramètre inconnu. Ainsi, entre les horloges du récepteur et du satellite cette insuffisance de rang est de taille 1, au lieu de 2 sinon. On décide de fixer ainsi l'horloge du récepteur r1.

- Les Biais matériels de code et de phase d'un récepteur d_r et δ_r , (défaut de rang b, c et d)

On continue d'appliquer une perturbation pour voir l'impact sur les observations de phase de code cette fois sur d'autres paramètres comme les biais matériels d'un récepteur et les biais matériels d'un satellite. Dans ce cas, l'insuffisance de rang est par défaut de taille 2. De même, entre les horloges du récepteur et les biais matériels du récepteur, un autre type d'insuffisance de rang, de taille n-1 (on rappelle que n est le nombre de stations), peut être identifié. Enfin, entre les biais matériels du récepteur pour une certaine fréquence et les ambiguïtés de la même fréquence, on identifie un défaut de rang de taille 2(n - 1).

- Biais matériel d'un satellite d^s et δ^s (défaut de rang e et f)

Comme pour le type d'insuffisance de rang précédent, il existe une insuffisance de rang (de taille k) entre les horloges du satellite et les biais de phase et de code du satellite. Enfin, une insuffisance de rang, de taille 2k, est la contrepartie satellitaire de l'insuffisance de rang identifiée précédemment entre les biais matériels du récepteur et les ambiguïtés de même fréquence. Elle s'applique aux biais et ambiguïtés de phase du satellite.

Ainsi, le défaut global de rang ν peut être calculé :

$$\nu = \underset{a}{1} + \underset{b}{2} + \underset{c}{(n-1)} + \underset{d}{2(n-1)} + \underset{e}{k} + \underset{f}{2k} = 3n + 3k \quad (3.9)$$

Dans le cas de périodes relativement courtes ou d'intervalles d'échantillonnage courts entre les observations, il peut être raisonnable de conserver certains paramètres comme constantes

temporelles dans le modèle. Dans ce cas, les déficiences de rang ci-dessus ne changent pas. D'autres déficiences de classement apparaissent cependant, non seulement si les retards ionosphériques sont modélisés comme des paramètres obliques, mais aussi pour des réseaux de taille régionale ou en l'absence de contraintes temporelles sur certains ou tous les paramètres variant dans le temps (Odijk, et al., 2016).

Pour représenter ce défaut de rang, il est intéressant de construire la matrice d'espace nulle. De plus, cette matrice est un outil efficace pour fixer certains paramètres et ainsi supprimer les défauts de rang avec pour finalité de construire la base S pour l'application de la S-theory. Nous allons maintenant appliquer directement le principe de la S-theory sans détailler cette matrice (Teunissen, et al., 2015).

3.1.2 Base S – Construction à partir du défaut de rang

Reprenons les différents défauts de rang exposés précédemment, de taille ν , afin de construire la base S associée la plus appropriée. La base S correspond à des contraintes qui sont imposées au système d'équations. Comme exposé dans la partie 2.2.3, il existe différentes bases S possibles qui permettront de résoudre le système d'équations. Pour ma solution PPP-NRTK, j'ai décidé de fixer les contraintes suivantes.

Table 8- Contraintes, données fixées pour la génération de corrections PPP-RTK

Contraintes		Défaut de rang associé
Erreur d'horloge d'une station r_1	$dt_{r_1}(1)$	a
Biais de code du récepteur r_1	$d_{r_1}(1)$	b
Biais de phase du récepteur r_1	$\delta_{r_1}(1)$	b
Biais de code du récepteur r sans ionosphère	$d_{r,IF}(1)$	c
Biais de code du satellite s sans ionosphère	$d_{IF}^s(1)$	e
Ambiguïté de la phase du récepteur r_1	$N_{r_1}^s$ $S = 1, \dots, k$	d
Ambiguïté de la phase du satellite s_1	$N_r^{s_1}$ $r = 1, \dots, n-1$	f

Les paramètres de position corrigés en troposphère (ZTD) estimable, ainsi que les paramètres ionosphériques estimables, sont identiques à leurs paramètres d'origine. Cela s'explique par le fait que ces paramètres n'ont été impliqués dans aucun des défauts de rang explicités juste avant. Tous les autres types de paramètres estimables sont des combinaisons ou des fonctions (linéaires) des paramètres originaux. Le paramètre estimable de l'horloge du satellite, c'est-à-dire $\widehat{dt}^s(i)$, est une fonction de l'horloge vraie du satellite $dt^s(i)$, combinée au biais du code du satellite sans ionosphère $d_{IF}^s(1)$, ainsi que de l'horloge du récepteur pivot $dt_{r_1}(1)$ et du biais du code sans ionosphère du récepteur pivot $d_{r_1,IF}(1)$. :

$$\widetilde{dt}_r(i) - \widetilde{dt}^s(i) = dt'_r(i) - dt'^s(i) = [dt'_r(i) + dt'_{r_1}(1)] - [dt'^s(i) - dt'_{r_1}(1)] \quad (3.10)$$

Avec $\begin{cases} dt'_r(i) = dt_r(i) + d_{r,IF}(1) \\ dt'^s(i) = dt^s(i) + d_{IF}^s(1) \end{cases}$ également connu sous le nom d'horloge satellite sans ionosphère.

On a donc :

$$\begin{cases} \widetilde{dt}_r(i) = [dt_r(i) + d_{r,IF}(1)] - [dt_{r_1}(1) + d_{r_1,IF}(1)] \\ \widetilde{dt}^s(i) = [dt^s(i) + d_{IF}^s(1)] - [dt_{r_1}(1) + d_{r_1,IF}(1)] \end{cases} \quad (3.11)$$

On réécrit (3.5) et (3.6) :

$$d^s = \begin{pmatrix} \mu_1 & 1 \\ \mu_2 & 1 \end{pmatrix} \begin{pmatrix} d_{GF}^s \\ d_{IF}^s \end{pmatrix} = \begin{pmatrix} \mu_1 d_{GF}^s + d_{IF}^s \\ \mu_2 d_{GF}^s + d_{IF}^s \end{pmatrix} = \mu d_{GF}^s + d_{IF}^s \quad (3.12)$$

De même :

$$d_r = \begin{pmatrix} \mu_1 & 1 \\ \mu_2 & 1 \end{pmatrix} \begin{pmatrix} d_{r,GF} \\ d_{r,IF} \end{pmatrix} = \mu d_{r,GF} + d_{r,IF} \quad (3.13)$$

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$$

(3.12) et (3.13) sont valables car nous sommes dans une solution double fréquence (Teunissen, et al., 2015)

Comme pour les erreurs d'horloge où on a fixé l'horloge du premier récepteur pour éliminer le défaut de rang on fixe d_{r_1} , δ_{r_1} , $N_{r_1}^s$ et $N_r^{s_1}$ pour éliminer les rangs restants. Comme on l'a fait avec l'horloge on fait une différenciation entre la valeur du paramètre pour le récepteur r et la valeur du paramètre pour le récepteur r_1 d'une part et satellite s_1 d'autre part. En ce qui concerne les biais estimables des codes des récepteurs et des satellites, une combinaison du biais du code sur une certaine fréquence et de la combinaison du biais du code sans ionosphère apparaît. Il en résulte une situation particulière pour la première époque ($i = 1$) et les deux premières fréquences ($j = 1, 2$) :

$$\begin{cases} \widetilde{d}_r(1) = d_r(1) - d_{r_1}(1) = d_{r,GF}(1) - d_{r_1,GF}(1) \\ \widetilde{d}^s(1) = d^s(1) - d_{r_1}(1) = d_{GF}^s(1) - d_{r_1,GF}(1) \end{cases} \quad (3.14)$$

Pour les temps $i > 1$, cette fois c'est la partie IF du biais de code qui sera fixée :

$$\begin{cases} \widetilde{d}_r(i) = d_r(i) - d_{r_1}(1) = [d_r(i) - d_{r,IF}(1)] - [d_{r_1}(1) - d_{r_1,IF}(1)] \\ \widetilde{d}^s(i) = d^s(i) - d_{r_1}(1) = [d^s(i) - d_{IF}^s(1)] - [d_{r_1}(1) - d_{r_1,IF}(1)] \end{cases} \quad (3.15)$$

Ainsi, les coefficients sans géométrie sont annulés lorsqu'ils sont additionnés, alors que multipliés par le vecteur de coefficients ionosphériques, on obtient 1. Il convient de noter que ces propriétés sont exactement opposées à celles du vecteur de coefficients sans ionosphère.

Il nous reste à réécrire les paramètres estimables pour le biais de phase et les ambiguïtés sachant qu'il existe un défaut de rang entre ces deux paramètres. Comme évoqué plus haut nous avons fixé δ_{r_1} , $N_{r_1}^s$ et $N_r^{s_1}$. Nous allons donc opérer encore une fois une différenciation :

$$\left\{ \begin{array}{l} \widetilde{\delta}_r(i) = [\delta_r(i) - \Lambda^{-1}d_{r,IF}(1)] - [\delta_{r_1}(1) - \Lambda^{-1}d_{r_1,IF}(1)] + [N_r^{s_1} - N_{r_1}^{s_1}] \\ \widetilde{\delta}^s(i) = [\delta^s(i) - \Lambda^{-1}d_{IF}^s(1)] - [\delta_{r_1}(1) - \Lambda^{-1}d_{r_1,IF}(1)] - N_{r_1}^s \\ \widetilde{N}_r^s = [N_r^s - N_r^{s_1}] - [N_{r_1}^s - N_{r_1}^{s_1}] \end{array} \right. \quad (3.16)$$

On peut réécrire chaque paramètre à partir des contraintes dépendant des défauts de rang et ainsi faire une différenciation entre la valeur du paramètre pour le récepteur r et la valeur du paramètre pour le récepteur r1 à l'instant 1 d'une part et satellite s et s1 d'autre part.

$$(3.17) \left\{ \begin{array}{l} \widetilde{\rho}_r^s = \rho_r^s \\ \widetilde{dt}_r(i) = [dt_r(i) + d_{r,IF}(1)] - [dt_{r_1}(1) + d_{r_1,IF}(1)] \\ \widetilde{dt}^s(i) = [dt^s(i) + d_{IF}^s(1)] - [dt_{r_1}(1) + d_{r_1,IF}(1)] \\ \widetilde{d}_r(i) = [d_r(i) - d_{r,IF}(1)] - [d_{r_1}(1) - d_{r_1,IF}(1)] \\ \widetilde{\delta}_r(i) = [\delta_r(i) - \Lambda^{-1}d_{r,IF}(1)] - [\delta_{r_1}(1) - \Lambda^{-1}d_{r_1,IF}(1)] + [N_r^{s_1} - N_{r_1}^{s_1}] \\ \widetilde{d}^s(i) = [d^s(i) - d_{IF}^s(1)] - [d_{r_1}(1) - d_{r_1,IF}(1)] \\ \widetilde{\delta}^s(i) = [\delta^s(i) - \Lambda^{-1}d_{IF}^s(1)] - [\delta_{r_1}(1) - \Lambda^{-1}d_{r_1,IF}(1)] - N_{r_1}^s \\ \widetilde{\Delta}_r^s(i) = \Delta_r^s(i) \\ \widetilde{N}_r^s = [N_r^s - N_r^{s_1}] - [N_{r_1}^s - N_{r_1}^{s_1}] \end{array} \right.$$

On réarrange (3.17) :

$$(3.18) \left\{ \begin{array}{l} \rho_r^s = \widetilde{\rho}_r^s \\ dt_r(i) = \widetilde{dt}_r(i) + [dt_{r_1}(1) + d_{r_1,IF}(1)] - d_{r,IF}(1) \\ dt^s(i) = \widetilde{dt}^s(i) + [dt_{r_1}(1) + d_{r_1,IF}(1)] - d_{IF}^s(1) \\ d_r(i) = \widetilde{d}_r(i) + d_{r,IF}(1) + [d_{r_1}(1) - d_{r_1,IF}(1)] \\ \delta_r(i) = \widetilde{\delta}_r(i) + \Lambda^{-1}d_{r,IF}(1) + [\delta_{r_1}(1) - \Lambda^{-1}d_{r_1,IF}(1)] - [N_r^{s_1} - N_{r_1}^{s_1}] \\ d^s(i) = \widetilde{d}^s(i) + d_{IF}^s(1) + [d_{r_1}(1) - d_{r_1,IF}(1)] \\ \delta^s(i) = \widetilde{\delta}^s(i) + \Lambda^{-1}d_{IF}^s(1) + [\delta_{r_1}(1) - \Lambda^{-1}d_{r_1,IF}(1)] + N_{r_1}^s \\ \Delta_r^s(i) = \widetilde{\Delta}_r^s(i) \\ N_r^s = \widetilde{N}_r^s + N_r^{s_1} + [N_{r_1}^s - N_{r_1}^{s_1}] \end{array} \right.$$

Maintenant qu'on a nos nouveaux paramètres, nous pouvons réécrire notre système d'équations à partir des nouveaux paramètres estimés, cela est possible car nos équations sont toutes linéaires :

$$(3.19) \left\{ \begin{array}{l} \Omega_r^s(i) - e\widetilde{\rho}_r^s(i) = e(\widetilde{dt}_r(i) - \widetilde{dt}^s(i)) + \widetilde{d}_r(i) - \widetilde{d}^s(i) + \mu\widetilde{\Delta}_r^s(i) \\ \Phi_r^s(i) - e\widetilde{\rho}_r^s(i) = e\widetilde{dt}_r(i) - e\widetilde{dt}^s(i) + \Lambda(\widetilde{\delta}_r(i) - \widetilde{\delta}^s(i) + \widetilde{N}_r^s) - \mu\widetilde{\Delta}_r^s(i) \end{array} \right.$$

Ces équations sont ainsi valables pour les k satellites et les n stations. On cherche à déterminer \widetilde{dt}^s , \widetilde{d}^s , $\widetilde{\delta}^s$.

Pour un satellite s on a :

$$(3.20) \left\{ \begin{array}{l} \Omega_{r_1}^s(i) - e\widetilde{\rho}_{r_1}^s(i) = -e\widetilde{dt}^s(i) - \widetilde{d}^s(i) + \mu\widetilde{\Delta}_{r_1}^s(i) \\ \Phi_{r_1}^s(i) - e\widetilde{\rho}_{r_1}^s(i) = e\widetilde{dt}_{r_1}(i) - e\widetilde{dt}^s(i) + \Lambda(\widetilde{\delta}_{r_1}(i) - \widetilde{\delta}^s(i) + N_{1r_1}^{1s}) - \mu\widetilde{\Delta}_{r_1}^s(i) \\ \Omega_{r_2}^s(i) - e\widetilde{\rho}_{r_2}^s(i) = e\widetilde{dt}_{r_2}(i) - e\widetilde{dt}^s(i) + \widetilde{d}_{r_2}(i) - \widetilde{d}^s(i) + \mu\widetilde{\Delta}_{r_2}^s(i) \\ \Phi_{r_2}^s(i) - e\widetilde{\rho}_{r_2}^s(i) = e\widetilde{dt}_{r_2}(i) - e\widetilde{dt}^s(i) + \Lambda(\widetilde{\delta}_{r_2}(i) - \widetilde{\delta}^s(i) + N_{1r_2}^{1s}) - \mu\widetilde{\Delta}_{r_2}^s(i) \\ \dots \\ \Omega_{r_n}^s(i) - e\widetilde{\rho}_{r_n}^s(i) = e\widetilde{dt}_{r_n}(i) - e\widetilde{dt}^s(i) + \widetilde{d}_{r_n}(i) - \widetilde{d}^s(i) + \mu\widetilde{\Delta}_{r_n}^s(i) \\ \Phi_{r_n}^s(i) - e\widetilde{\rho}_{r_n}^s(i) = e\widetilde{dt}_{r_n}(i) - e\widetilde{dt}^s(i) + \Lambda(\widetilde{\delta}_{r_n}(i) - \widetilde{\delta}^s(i) + N_{1r_n}^{1s}) - \mu\widetilde{\Delta}_{r_n}^s(i) \end{array} \right.$$

Que l'on peut réécrire ainsi :

$$(3.21) \left\{ \begin{array}{l} e\widetilde{dt}^s(i) + \widetilde{d}^s(i) = e\widetilde{\rho}_{r_1}^s(i) - \Omega_{r_1}^s(i) + \mu\widetilde{\Delta}_{r_1}^s(i) \\ e\widetilde{dt}^s(i) + \Lambda\widetilde{\delta}^s(i) = e\widetilde{\rho}_{r_1}^s(i) - \Phi_{r_1}^s(i) + e\widetilde{dt}_{r_1}(i) + \Lambda(\widetilde{\delta}_{r_1}(i) + N_{1r_1}^{1s}) - \mu\widetilde{\Delta}_{r_1}^s(i) \\ e\widetilde{dt}^s(i) + \widetilde{d}^s(i) = e\widetilde{\rho}_{r_2}^s(i) - \Omega_{r_2}^s(i) + e\widetilde{dt}_{r_2}(i) + \widetilde{d}_{r_2}(i) + \mu\widetilde{\Delta}_{r_2}^s(i) \\ e\widetilde{dt}^s(i) + \Lambda\widetilde{\delta}^s(i) = e\widetilde{\rho}_{r_2}^s(i) - \Phi_{r_2}^s(i) + e\widetilde{dt}_{r_2}(i) + \Lambda(\widetilde{\delta}_{r_2}(i) + N_{1r_2}^{1s}) - \mu\widetilde{\Delta}_{r_2}^s(i) \\ \dots \\ e\widetilde{dt}^s(i) + \widetilde{d}^s(i) = e\widetilde{\rho}_{r_n}^s(i) - \Omega_{r_n}^s(i) + e\widetilde{dt}_{r_n}(i) + \widetilde{d}_{r_n}(i) + \mu\widetilde{\Delta}_{r_n}^s(i) \\ e\widetilde{dt}^s(i) + \Lambda\widetilde{\delta}^s(i) = e\widetilde{\rho}_{r_n}^s(i) - \Phi_{r_n}^s(i) + e\widetilde{dt}_{r_n}(i) + \Lambda(\widetilde{\delta}_{r_n}(i) + N_{1r_n}^{1s}) - \mu\widetilde{\Delta}_{r_n}^s(i) \end{array} \right.$$

On fait une différenciation entre deux satellites s_1 et s_2 avec le système (3.21):

$$(3.22) \left\{ \begin{array}{l} e\widetilde{dt}^{s_1}(i) + \widetilde{d}^{s_1}(i) - e\widetilde{dt}^{s_2}(i) - \widetilde{d}^{s_2}(i) = e\widetilde{\rho}_{r_1}^{s_1}(i) - e\widetilde{\rho}_{r_1}^{s_2}(i) + \Omega_{r_1}^{s_2}(i) - \Omega_{r_1}^{s_1}(i) + \mu\widetilde{\Delta}_{r_1}^{s_1}(i) - \mu\widetilde{\Delta}_{r_1}^{s_2}(i) \\ e\widetilde{dt}^{s_1}(i) + \Lambda\widetilde{\delta}^{s_1}(i) - e\widetilde{dt}^{s_2}(i) - \Lambda\widetilde{\delta}^{s_2}(i) = e\widetilde{\rho}_{r_1}^{s_1}(i) - e\widetilde{\rho}_{r_1}^{s_2}(i) + \Phi_{r_1}^{s_2}(i) - \Phi_{r_1}^{s_1}(i) + \Lambda(N_{1r_1}^{1s_1} - N_{1r_1}^{1s_2}) + \mu\widetilde{\Delta}_{r_1}^{s_2}(i) - \mu\widetilde{\Delta}_{r_1}^{s_1}(i) \\ e\widetilde{dt}^{s_1}(i) + \widetilde{d}^{s_1}(i) - e\widetilde{dt}^{s_2}(i) - \widetilde{d}^{s_2}(i) = e\widetilde{\rho}_{r_2}^{s_1}(i) - e\widetilde{\rho}_{r_2}^{s_2}(i) + \Omega_{r_2}^{s_2}(i) - \Omega_{r_2}^{s_1}(i) + \mu\widetilde{\Delta}_{r_2}^{s_1}(i) - \mu\widetilde{\Delta}_{r_2}^{s_2}(i) \\ e\widetilde{dt}^{s_1}(i) + \Lambda\widetilde{\delta}^{s_1}(i) - e\widetilde{dt}^{s_2}(i) - \Lambda\widetilde{\delta}^{s_2}(i) = e\widetilde{\rho}_{r_2}^{s_1}(i) - e\widetilde{\rho}_{r_2}^{s_2}(i) + \Phi_{r_2}^{s_2}(i) - \Phi_{r_2}^{s_1}(i) + \Lambda(N_{1r_2}^{1s_1} - N_{1r_2}^{1s_2}) + \mu\widetilde{\Delta}_{r_2}^{s_2}(i) - \mu\widetilde{\Delta}_{r_2}^{s_1}(i) \\ \dots \\ e\widetilde{dt}^{s_1}(i) + \widetilde{d}^{s_1}(i) - e\widetilde{dt}^{s_2}(i) - \widetilde{d}^{s_2}(i) = e\widetilde{\rho}_{r_n}^{s_1}(i) - e\widetilde{\rho}_{r_n}^{s_2}(i) + \Omega_{r_n}^{s_2}(i) - \Omega_{r_n}^{s_1}(i) + \mu\widetilde{\Delta}_{r_n}^{s_1}(i) - \mu\widetilde{\Delta}_{r_n}^{s_2}(i) \\ e\widetilde{dt}^{s_1}(i) + \Lambda\widetilde{\delta}^{s_1}(i) - e\widetilde{dt}^{s_2}(i) - \Lambda\widetilde{\delta}^{s_2}(i) = e\widetilde{\rho}_{r_n}^{s_1}(i) - e\widetilde{\rho}_{r_n}^{s_2}(i) + \Phi_{r_n}^{s_2}(i) - \Phi_{r_n}^{s_1}(i) + \Lambda(N_{1r_n}^{1s_1} - N_{1r_n}^{1s_2}) + \mu\widetilde{\Delta}_{r_n}^{s_2}(i) - \mu\widetilde{\Delta}_{r_n}^{s_1}(i) \end{array} \right.$$

On peut simplifier (3.22) :

$$(3.23) \left\{ \begin{array}{l} \overline{edt}^{s_1 s_2}(i) + \overline{d}^{s_1 s_2}(i) = e\overline{\rho}_{r_1}^{s_1 s_2}(i) - \Omega_{r_1}^{s_1 s_2}(i) + \mu\overline{\Delta}_{r_1}^{s_1 s_2}(i) \\ \overline{edt}^{s_1 s_2}(i) + \Lambda\overline{\delta}^{s_1 s_2}(i) = e\overline{\rho}_{r_1}^{s_1 s_2}(i) - \Phi_{r_1}^{s_1 s_2}(i) + \Lambda N_{1r_1}^{s_1 s_2} - \mu\overline{\Delta}_{r_1}^{s_1 s_2}(i) \\ \overline{edt}^{s_1 s_2}(i) + \overline{d}^{s_1 s_2}(i) = e\overline{\rho}_{r_2}^{s_1 s_2}(i) - \Omega_{r_2}^{s_1 s_2}(i) + \mu\overline{\Delta}_{r_2}^{s_1 s_2}(i) \\ \overline{edt}^{s_1 s_2}(i) + \Lambda\overline{\delta}^{s_1 s_2}(i) = e\overline{\rho}_{r_2}^{s_1 s_2}(i) - \Phi_{r_2}^{s_1 s_2}(i) + \Lambda N_{1r_2}^{s_1 s_2} - \mu\overline{\Delta}_{r_2}^{s_1 s_2}(i) \\ \dots \\ \overline{edt}^{s_1 s_2}(i) + \overline{d}^{s_1 s_2}(i) = e\overline{\rho}_{r_n}^{s_1 s_2}(i) - \Omega_{r_n}^{s_1 s_2}(i) + \mu\overline{\Delta}_{r_n}^{s_1 s_2}(i) \\ \overline{edt}^{s_1 s_2}(i) + \Lambda\overline{\delta}^{s_1 s_2}(i) = e\overline{\rho}_{r_n}^{s_1 s_2}(i) - \Phi_{r_n}^{s_1 s_2}(i) + \Lambda N_{1r_n}^{s_1 s_2} - \mu\overline{\Delta}_{r_n}^{s_1 s_2}(i) \end{array} \right.$$

$$\text{avec } X_r^{s_1 s_2} = X_r^{s_1} - X_r^{s_2}$$

On pourra ainsi déterminer pour chaque paire de satellites : $\overline{dt}^{s_1 s_2}(i)$, $\overline{d}^{s_1 s_2}(i)$ et $\overline{\delta}^{s_1 s_2}(i)$. Grâce à nos hypothèses fixées dans notre base S on a pour $i=1$ pour la paire de satellite $s_1 s_2$: $\overline{d}^{s_1 s_2}(1) = d_{GF}^{s_1}(1) - d_{GF}^{s_2}(1)$ ainsi :

$$(3.24) \left\{ \begin{array}{l} \overline{edt}^{s_1 s_2}(1) = e\overline{\rho}_{r_1}^{s_1 s_2}(1) - \Omega_{r_1}^{s_1 s_2}(1) + \mu\overline{\Delta}_{r_1}^{s_1 s_2}(1) - d_{GF}^{s_1}(1) - d_{GF}^{s_2}(1) \\ \Lambda\overline{\delta}^{s_1 s_2}(1) = e\overline{\rho}_{r_1}^{s_1 s_2}(1) - \Phi_{r_1}^{s_1 s_2}(1) + \Lambda N_{1r_1}^{s_1 s_2} - \mu\overline{\Delta}_{r_1}^{s_1 s_2}(1) - \overline{edt}^{s_1 s_2}(1) \\ \overline{edt}^{s_1 s_2}(1) = e\overline{\rho}_{r_2}^{s_1 s_2}(1) - \Omega_{r_2}^{s_1 s_2}(1) + \mu\overline{\Delta}_{r_2}^{s_1 s_2}(1) - d_{GF}^{s_1}(1) - d_{GF}^{s_2}(1) \\ \Lambda\overline{\delta}^{s_1 s_2}(1) = e\overline{\rho}_{r_2}^{s_1 s_2}(1) - \Phi_{r_2}^{s_1 s_2}(1) + \Lambda N_{1r_2}^{s_1 s_2} - \mu\overline{\Delta}_{r_2}^{s_1 s_2}(1) - \overline{edt}^{s_1 s_2}(1) \\ \dots \\ \overline{edt}^{s_1 s_2}(1) = e\overline{\rho}_{r_n}^{s_1 s_2}(1) - \Omega_{r_n}^{s_1 s_2}(1) + \mu\overline{\Delta}_{r_n}^{s_1 s_2}(1) - d_{GF}^{s_1}(1) - d_{GF}^{s_2}(1) \\ \Lambda\overline{\delta}^{s_1 s_2}(1) = e\overline{\rho}_{r_n}^{s_1 s_2}(1) - \Phi_{r_n}^{s_1 s_2}(1) + \Lambda N_{1r_n}^{s_1 s_2} - \mu\overline{\Delta}_{r_n}^{s_1 s_2}(1) - \overline{edt}^{s_1 s_2}(1) \end{array} \right.$$

Ainsi, pour la première époque ($i=1$) et pour les stations du réseau r on a :

$$(3.25) \left\{ \begin{array}{l} \overline{edt}^{s_1 s_2}(1) = e\overline{\rho}_r^{s_1 s_2}(1) - \Omega_r^{s_1 s_2}(1) + \mu\overline{\Delta}_r^{s_1 s_2}(1) - d_{GF}^{s_1}(1) - d_{GF}^{s_2}(1) \\ \overline{d}^{s_1 s_2}(1) = d_{GF}^{s_1}(1) - d_{GF}^{s_2}(1) \\ \Lambda\overline{\delta}^{s_1 s_2}(1) = -\Phi_{r_2}^{s_1 s_2}(1) + \Lambda N_{1r_2}^{s_1 s_2} - \mu\overline{\Delta}_{r_2}^{s_1 s_2}(1) + \Omega_{r_2}^{s_1 s_2}(1) - \mu\overline{\Delta}_{r_2}^{s_1 s_2}(1) + d_{GF}^{s_1}(1) - d_{GF}^{s_2}(1) \end{array} \right.$$

Les erreurs d'horloges peuvent être approchées grâce aux paramètres fixés, avec $i=1$.

C'est une première initialisation dans nos paramètres à déterminer. En continuant l'itération, grâce aux différentes données des stations ainsi qu'à celles fixées par nos contraintes, on peut déterminer l'ensemble des données satellitaires de notre base S.

Enfin, pour que ces modèles soient valables, on estime que les horloges des récepteurs sont connectées dans le temps, sans saut de cycle. On parle ici d'un modèle d'horloge commune car les horloges de la phase et du code sont les mêmes.

Toutefois, il reste deux paramètres dans notre système d'équations (3.25) à déterminer :

- Les erreurs ionosphériques : nous possédons un réseau de stations CORS dont on connaît la position. Ainsi, il est possible d'estimer les erreurs ionosphériques et on les suppose donc comme connue.
- Les ambiguïtés : dans ce cas les ambiguïtés sont double différenciées et sont donc entières. On rentre dans le cadre d'une résolution d'ambiguïtés comme en RTK. On détaillera la méthode d'estimation des ambiguïtés dans la partie suivante.

3.1.3 Ambiguïtés du réseau

La résolution des ambiguïtés GNSS est le processus de résolution des ambiguïtés inconnues des données de phase de la porteuse Double Différenciées sous forme de nombres entiers. Comme explicité dans l'état de l'art, c'est un procédé fortement utilisé dans les techniques différenciées comme le RTK. Les ambiguïtés dans notre solution PPP-RTK d'un point de vue réseau, sont désormais double différenciées donc entières :

$$N_{1r,j}^{1s} = [N_{r,j}^s - N_{r,j}^{s_1}] - [N_{r_1,j}^s - N_{r_1,j}^{s_1}]$$

Pour la résolution nous utilisons donc la technique LAMBDA comme en RTK. Les modèles GNSS sur lesquels repose la résolution des ambiguïtés peuvent tous être intégrés dans le cadre conceptuel suivant d'équations d'observation linéarisées, écrites ainsi :

$$y = Aa + Bb + e$$

où a est le vecteur de paramètres entiers avec les ambiguïtés entières de Double différences. b est le vecteur de paramètres à valeurs réelles, y compris les composantes de la ligne de base et éventuellement les paramètres de retard troposphérique et ionosphérique, etc. Les matrices de coefficients sont A et B , avec $[AB]$ de rang pleine colonne. Le vecteur d'observation y contient les pseudodistances et les phases de la porteuse observées moins celles qui sont calculées. Ces valeurs sont contaminées par le vecteur de bruit aléatoire e . En général, on suppose que e est normalement distribué avec une moyenne nulle et une matrice de variance-covariance Q_{yy} .

Dans un premier temps, la propriété entière des ambiguïtés a n'est pas prise en compte et les estimations dites flottantes ainsi que leur matrice de variance-covariance sont calculées.

$$\begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} \begin{bmatrix} Q_{\hat{a}\hat{a}} & Q_{\hat{a}\hat{b}} \\ Q_{\hat{b}\hat{a}} & Q_{\hat{b}\hat{b}} \end{bmatrix}$$

Dans la deuxième étape, l'estimation de l'ambiguïté des flottants \hat{a} est utilisée pour calculer l'estimation des ambiguïtés des entiers correspondant. Dans cette étape, il existe différents choix possibles de la fonction de mappage, qui correspondent aux différentes méthodes d'estimation des

nombre entiers. Les choix les plus courants sont la méthode des moindres carrés entiers (ILS), le bootstrapping entier (IB) et l'arrondi entier (IR). La méthode ILS est optimale, car on peut montrer qu'elle a le taux de réussite le plus élevé de tous les estimateurs de nombres entiers (Li, et al., 2011). Cependant, IR et IB peuvent également donner d'excellents résultats, en particulier après l'application de la décorrélation LAMBDA. Leur avantage par rapport à l'ILS est qu'aucune recherche de nombres entiers n'est nécessaire.

La troisième étape consiste à décider d'accepter ou non la solution entière une fois que des estimations entières des ambiguïtés ont été calculées. Plusieurs tests de ce type ont été proposés dans la littérature. Les exemples actuellement utilisés dans la pratique comprennent le test du rapport, le test du rapport F, le test de la différence et le test du projecteur. Ces tests, ainsi que d'autres, peuvent être intégrés dans le cadre de l'estimation par ouverture d'un nombre entier, LAMBDA Software Package (TU Delft).

Dans la quatrième étape, les solutions flottantes des paramètres réels dans la première étape sont mises à jour à l'aide des paramètres entiers fixes.

Ainsi, à la fin de ce procédé les ambiguïtés de phase au sein du réseau sont supposées connues.

3.1.4 Corrections PPP-RTK

On établit un vecteur de corrections suite à la résolution de (3.25) Il sera au format :

$V_{Corr} = [\widetilde{dt}^{ss}, \widetilde{d}^{ss}, \widetilde{\delta}^{ss}]$, ce sont ces données que l'on va transmettre au réseau.

Ce vecteur correspond à l'ensemble des données satellitaires différenciées :

$$V_{corr} = \left[\begin{pmatrix} \widetilde{dt}^{s_1 s_2} \\ \dots \\ \widetilde{dt}^{s_{k-1} s_k} \end{pmatrix}, \begin{pmatrix} \widetilde{d}^{s_1 s_2} \\ \dots \\ \widetilde{d}^{s_{k-1} s_k} \end{pmatrix}, \begin{pmatrix} \widetilde{\delta}^{s_1 s_2} \\ \dots \\ \widetilde{\delta}^{s_{k-1} s_k} \end{pmatrix} \right] \quad (3.26)$$

L'échelle du réseau va avoir un impact sur la résolution car selon le nombre de stations ces paramètres de corrections vont être moins précis. Le réseau permet d'estimer certaines erreurs associées aux biais des récepteurs, et biais matériels des satellites. Selon la densité du réseau, c'est à dire le nombre de stations et leur éloignement, les corrections envoyées connaîtront des variations plus ou moins importantes. L'application de notre solution veut qu'elle s'applique à une zone régionale de plusieurs centaines de kilomètres. Il faudra donc porter une attention particulière à la validité de ces corrections quand on élargit notre zone d'application. C'est une conséquence d'une représentation SSR. Voyons à présent l'organisation concrète de la résolution.

3.1.5 Mise en œuvre algorithmique de la fonction réseau

La fonction du réseau en PPP-RTK est claire : permettre de générer des corrections à partir d'un réseau de stations de bases. Ces corrections seront ainsi envoyées à l'utilisateur.

Dans notre cas nous pouvons établir la chaîne logicielle suivante (Figure 25) pour la partie réseau :

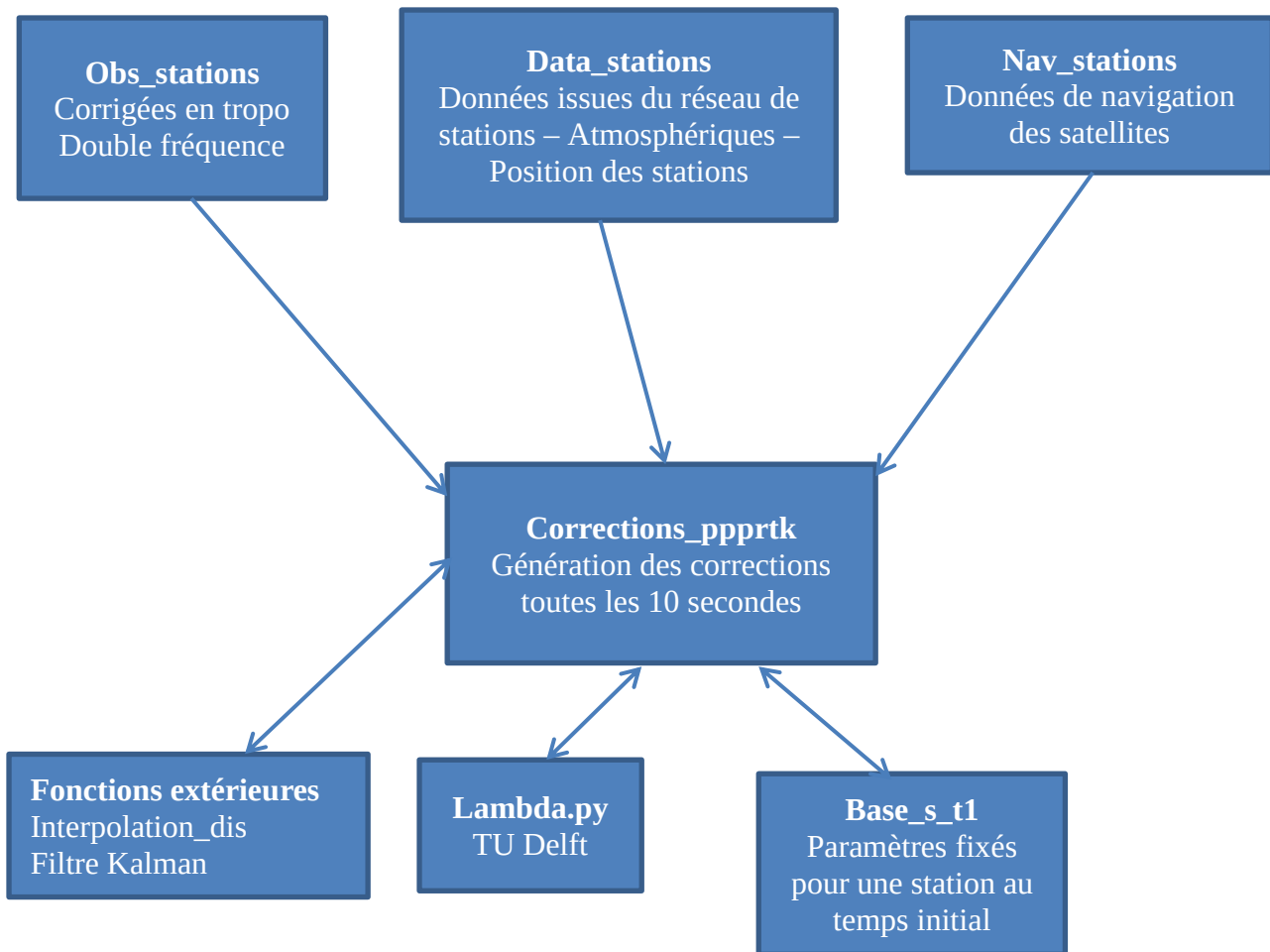


Figure 25- Chaîne Logicielle Partie Réseau, génération des corrections PPP-RTK

Les flèches simples correspondent aux données qui sont collectées, les flèches doubles montrent qu'il y a ce qu'on peut qualifier de « dialogue » au sens où les corrections PPP-RTK se nourrissent aussi des résultats que produisent les fonctions extérieures, la méthode LAMBDA et les paramètres de base. La structure présentée correspond à une programmation sous python. On y retrouve les différentes étapes présentées précédemment dans les équations : collecte des données, détermination des différentes erreurs et mise en place de la s-theory.

Nous venons de voir ce qui se passe niveau réseau, voyons à présent ce qu'il en est côté utilisateur.

3.2 Construction d'une nouvelle base S avec le NRTK

Nous changeons donc de point de vue et passons du côté utilisateur. Celui-ci est initialement connecté au réseau NRTK. Nous allons résoudre les équations afin de déterminer la position définissant le paramètre ρ_u^s , la distance récepteur satellite.

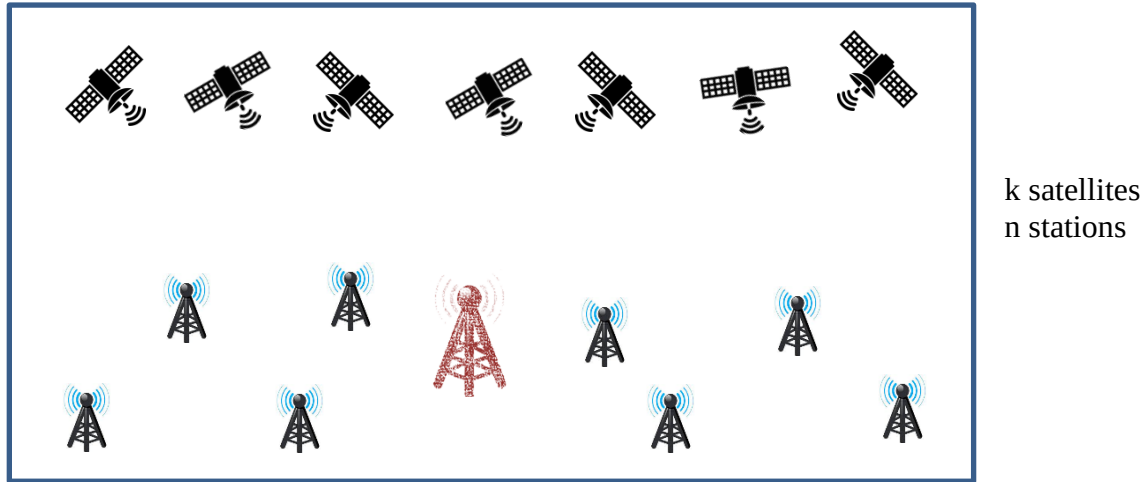


Figure 26- Utilisateur dans un réseau CORS interconnectés avec k satellites en visibilité

3.2.1 Initialisation avec le calcul NRTK à t0

C'est le temps d'initialisation dans ce cas, on se place au temps $i=0$, pour un utilisateur u , et un satellite $s=1, \dots, k$ pour une fréquence $j=1,2$:

$$\begin{cases} \Phi_u^s(0) - e\rho_u^s(0) = edt_u(0) - edt^s(0) + d_u(0) - d^s(0) + \mu\Delta_u^s(0) \\ \Phi_u^s(i) - e\rho_u^s(0) = edt_u(0) - edt^s(0) + \Lambda(\delta_u(0) - \delta^s(0) + N_u^s) - \mu\Delta_u^s(0) \end{cases} \quad (3.27)$$

Les observations de phase et de code sont corrigées sur la troposphère comme dans les équations du réseau.

Ensuite, on applique le principe de double différenciation avec les corrections VRS et deux satellites s et p :

$$\begin{cases} \Omega_u^{sp}(0) - R'_{VRS}^{sp}(0) = \rho_u^{sp}(0) - \rho_{VRS}^{sp}(0) \\ \Phi_u^{sp}(0) - \Phi_{VRS}^{sp}(0) = \rho_u^{sp}(0) - \rho_{VRS}^{sp}(0) + \Lambda(N_u^{sp} - N_{VRS}^{sp}) \end{cases} \quad (3.28)$$

Avec $X_a^{sp} = X_a^s - X_a^p$

Dans la suite des équations on s'intéressera seulement à l'équation de la phase :

$$\rho_u^{sp}(0) = \Phi_u^{sp}(0) - \Phi_{VRS}^{sp}(0) + \rho_{VRS}^{sp}(0) - \Lambda(N_u^{sp} - N_{VRS}^{sp}) \quad (3.29)$$

L'utilisation des corrections VRS correspond pour un utilisateur à se connecter au réseau Teria et ainsi calculer sa position. Pour cela, il recevra un flux de corrections VRS en fonction de l'endroit où il se trouve dans le réseau CORS.

Résolution des ambiguïtés

Dans ce cas, les ambiguïtés sont double différenciées et considérées comme entières

$$\Delta N_0 = N_{u,j}^{sp} - N_{VRS,j}^{sp}$$

La méthode LAMBDA est appliquée comme dans la partie 3.1.3.

3.2.2 Une nouvelle base S' – Modèle de corrections PPP-NRTK

L'utilisateur a calculé sa position $i=0$ avec le NRTK et reçoit des corrections PPP-RTK tel que :

$$V_{corr} = \left[\begin{pmatrix} \widetilde{dt}^{s_1 s_2} \\ \dots \\ \widetilde{dt}^{s_{k-1} s_k} \end{pmatrix}, \begin{pmatrix} \widetilde{d}^{s_1 s_2} \\ \dots \\ \widetilde{d}^{s_{k-1} s_k} \end{pmatrix}, \begin{pmatrix} \widetilde{\delta}^{s_1 s_2} \\ \dots \\ \widetilde{\delta}^{s_{k-1} s_k} \end{pmatrix} \right] \quad (3.30)$$

Reprenons les équations d'observations GNSS (2.12) :

$$\begin{cases} \Omega_u^s(i) - e\widetilde{\rho}_u^s(i) = e(\widetilde{dt}_u(i) - \widetilde{dt}^s(i)) + \widetilde{d}_u(i) - \widetilde{d}^s(i) + \mu\widetilde{\Delta}_u^s(i) \\ \Phi_u^s(i) - e\widetilde{\rho}_u^s(i) = e\widetilde{dt}_u(i) - e\widetilde{dt}^s(i) + \Lambda(\widetilde{\delta}_u(i) - \widetilde{\delta}^s(i) + N_{1u}^{1s}) - \mu\widetilde{\Delta}_u^s(i) \end{cases} \quad (3.31)$$

Cette fois on aura la base S [\widetilde{dt}^{ss} , \widetilde{d}^{ss} , $\widetilde{\delta}^{ss}$, $\rho_{uVRS}^{ss}(0)$, ΔN_0]

3.2.3 Conception algorithmique dans l'hybridation PPP-RTK

Reprenons la chaîne logicielle mise en œuvre dans la partie réseau pour la génération des corrections PPP-RTK. Désormais nous allons intégrer l'initialisation NRTK. La figure 27 montre la version mise à jour de l'algorithme :

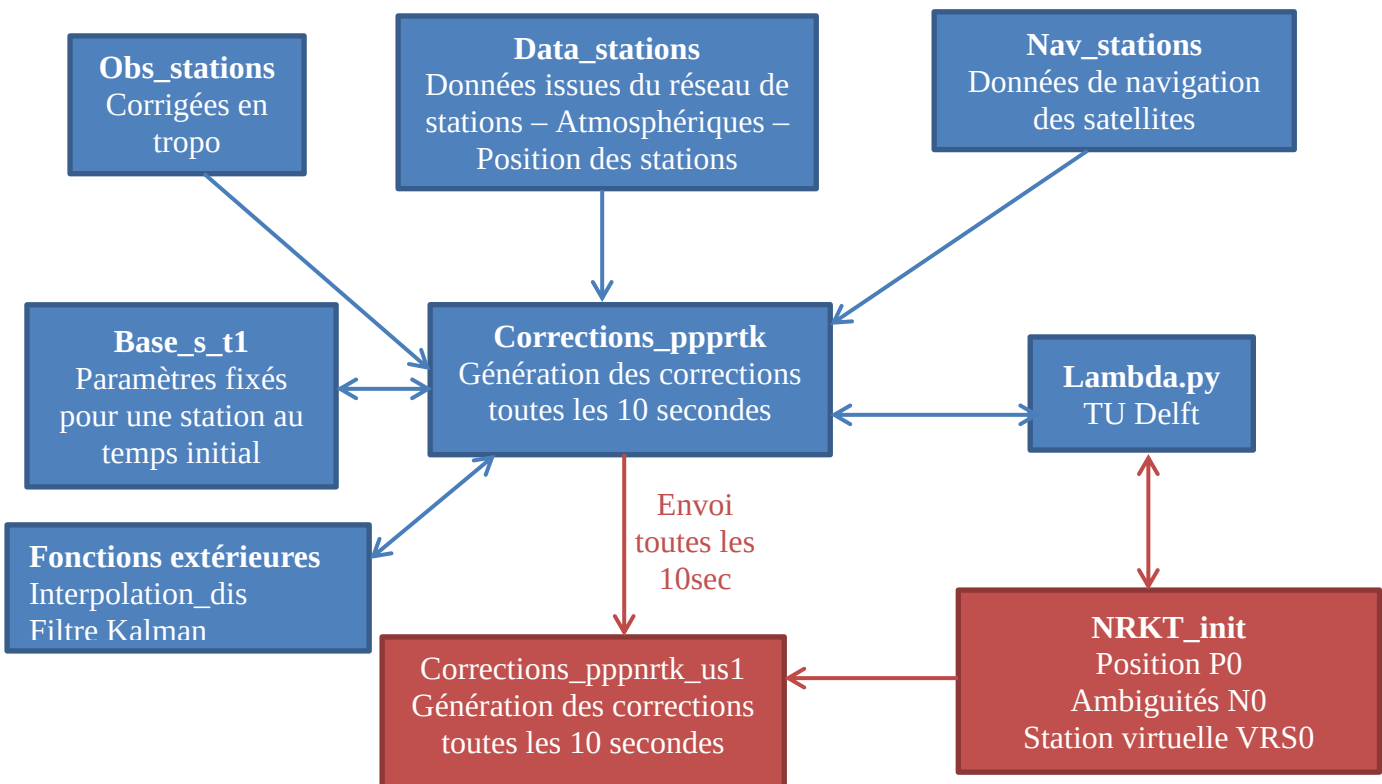


Figure 27- Chaîne logicielle étape 2 avec l'initialisation NRTK

Ainsi, à la sortie nous obtenons un nouveau format de corrections qui prend en compte l'initialisation NRTK. Ces corrections ne sont pas indépendantes de l'utilisateur. En effet, à chaque fois qu'un récepteur s reçoit les corrections PPP-RTK, il calcule sa position en NRTK puis à un nouveau format de corrections pour continuer à calculer sa position. Ainsi, à chaque arrivée de corrections PPP-RTK un nouveau vecteur de données PPP-NRTK est mis en place avec toujours la même initialisation NRTK.

3.3 Calcul de la position – partie utilisateur

3.3.1 Corrections PPP-NRTK reçues

Les données fournies par l'utilisateur sont donc :

$$V_{corr} = [\widehat{dt}^{ss}, \widehat{d}^{ss}, \widehat{\delta}^{ss}, \rho_{uVRS}^{ss}(0), \Delta N_0] \quad (3.32)$$

Il est important dans l'application de notre solution hybride de faire un lien entre l'initialisation du temps $i=0$ et le temps qui suit $i>0$. Il est utile de noter ici, que ce temps correspond à l'utilisateur et ne sera pas le même que dans le réseau : $i=0$ correspond au temps de résolution NRTK et $i>0$ au temps de résolution en PPP-RTK, mais dans les deux cas pour l'utilisateur. Il passe du NRTK au PPP-RTK.

Ce sont des données utilisant un satellite pivot, ainsi on réapplique une différenciation entre les satellites (s_1, s_2) d'un point de vue utilisateur, en injectant nos corrections PPP-NRTK (3.32) :

$$\begin{cases} \Omega_u^{s_1 s_2}(i) = e\widehat{\rho}_u^{s_1 s_2}(i) - e\widehat{dt}^{s_1 s_2}(i) - \widehat{d}^{s_1 s_2}(i) + \mu\widehat{\Delta}_u^{s_1 s_2}(i) \\ \widehat{\Phi}_u^{s_1 s_2}(i) = e\widehat{\rho}_u^{s_1 s_2}(i) - e\widehat{dt}^{s_1 s_2}(i) - \Lambda\widehat{\delta}^{s_1 s_2}(i) + \Lambda N_{1u}^{s_1 s_2} - \mu\widehat{\Delta}_u^{s_1 s_2}(i) \end{cases} \quad (3.33)$$

La principale inconnue qu'on cherche à déterminer est $\widehat{\rho}_u^{s_1 s_2}(i)$ la distance utilisateur satellite.

On insère les corrections envoyées à l'utilisateur :

$$\begin{cases} \widehat{\Omega}_r^{s_1 s_2}(i) = \Omega_u^{s_1 s_2}(i) + e\widehat{dt}^{s_1 s_2}(i) + \widehat{d}^{s_1 s_2}(i) - \rho_{uVRS}^{ss}(0) \\ \widehat{\Phi}_u^{s_1 s_2}(i) = \Phi_u^{s_1 s_2}(i) + e\widehat{dt}^{s_1 s_2}(i) + \Lambda\widehat{\delta}^{s_1 s_2}(i) - \rho_{uVRS}^{ss}(0) - \Lambda N_0 \end{cases} \quad (3.34)$$

$$\begin{cases} \widehat{\Omega}_u^{s_1 s_2}(i) = e\widehat{\rho}_u^{s_1 s_2}(i) - \rho_{uVRS}^{ss}(0) + \mu\widehat{\Delta}_u^{s_1 s_2}(i) \\ \widehat{\Phi}_u^{s_1 s_2}(i) = e\widehat{\rho}_u^{s_1 s_2}(i) - \rho_{uVRS}^{ss}(0) + \Lambda N_{1u}^{s_1 s_2} - \Lambda N_0 - \mu\widehat{\Delta}_u^{s_1 s_2}(i) \end{cases} \quad (3.35)$$

$$\begin{cases} e\widehat{\rho}_u^{s_1 s_2}(i) - \rho_{uVRS}^{ss}(0) = \widehat{\Omega}_u^{s_1 s_2}(i) - \mu\widehat{\Delta}_u^{s_1 s_2}(i) \\ e\widehat{\rho}_u^{s_1 s_2}(i) - \rho_{uVRS}^{ss}(0) = \widehat{\Phi}_u^{s_1 s_2}(i) - \Lambda N_{1u}^{s_1 s_2} + \Lambda N_0 + \mu\widehat{\Delta}_u^{s_1 s_2}(i) \end{cases} \quad (3.36)$$

Ces corrections permettent de supprimer les éléments dépendant du satellite dans le cadre de la partie PPP-RTK. On ajoute aussi une information supplémentaire due à notre initialisation qui aura deux effets. D'abord dans le calcul des ambiguïtés, on aura une initialisation de la méthode LAMBDA grâce aux ambiguïtés calculées en NRTK avec la VRS. Ensuite, le second intérêt sera pour préciser le

calcul de la position. En ayant un vecteur de base utilisateur satellite en $i=0$, on pourra avoir une donnée de vérification. En effet, on disposera d'une valeur de référence pour valider notre calcul pour une solution plus robuste.

Après l'initialisation en NRTK, la bascule avec les corrections PPP-RTK reçus, on reprend le postulat de doubles fréquences posé en début de résolution :

$$\left\{ \begin{array}{l} e\widehat{\rho}_u^{s_1s_2}(i) - \rho_{uVRS}^{ss}(0) = \widehat{R}_{u,j_1}^{rs}(i) - \mu_1 \widehat{\Delta}_u^{s_1s_2}(i) \\ e\widehat{\rho}_u^{s_1s_2}(i) - \rho_{uVRS}^{ss}(0) = \widehat{R}_{u,j_2}^{rs}(i) - \mu_1 \widehat{\Delta}_u^{s_1s_2}(i) \\ e\widehat{\rho}_u^{s_1s_2}(i) - \rho_{uVRS}^{ss}(0) = \widehat{\varphi}_{u,j_1}^{s_1s_2}(i) - \lambda_{j_1} N_{1u,j_1}^{1s_1s_2} + \lambda_{j_1} N_{0,j_1} + \mu_1 \widehat{\Delta}_u^{s_1s_2}(i) \\ e\widehat{\rho}_u^{s_1s_2}(i) - \rho_{uVRS}^{ss}(0) = \widehat{\varphi}_{u,j_2}^{s_1s_2}(i) - \lambda_{j_2} N_{1u,j_2}^{1s_1s_2} + \lambda_{j_2} N_{0,j_1} + \mu_2 \widehat{\Delta}_u^{s_1s_2}(i) \end{array} \right. (3.37)$$

Le modèle sans ionosphère revient à faire une différence entre les équations des deux fréquences sur le code et sur la phase. On différencie entre les deux fréquences en appliquant le principe de « ionosphere-free » et on obtient :

$$\left\{ \begin{array}{l} \Delta\widehat{\rho}_{u0}^{s_1s_2}(i) = \widehat{R}_{u,j_1}^{rs}(i) - \widehat{R}_{u,j_1}^{rs}(i) \\ \Delta\widehat{\rho}_{u0}^{s_1s_2}(i) = \widehat{\varphi}_{u,j_1}^{s_1s_2}(i) - \widehat{\varphi}_{u,j_2}^{s_1s_2}(i) + \lambda_{j_2} N_{1u,j_2}^{1s_1s_2} - \lambda_{j_1} N_{1u,j_1}^{1s_1s_2} + \lambda_{j_1} N_{0,j_1} - \lambda_{j_2} N_{0,j_1} \end{array} \right. (3.38)$$

Avec

$$\Delta\widehat{\rho}_{u0}^{s_1s_2}(i) = e\widehat{\rho}_u^{s_1s_2}(i) - \rho_{uVRS}^{ss}(0)$$

Maintenant, la principale erreur qui reste à résoudre correspond aux ambiguïtés qui sont double différenciées. Pour accélérer le processus il est possible d'utiliser l'ambiguïté initiale calculée en VRS. Cette connaissance des ambiguïtés initiales permet de les estimer plus rapidement pour $i>0$ et d'appliquer la méthode LAMBDA plus efficacement.

Il faut noter qu'on vient de faire la résolution avec seulement deux satellites. Il est possible de généraliser notre application à plus de deux satellites. On a alors une matrice avec l'ensemble des distances différenciées satellites-utilisateur :

$$\rho_u(i) = \begin{pmatrix} \Delta\widehat{\rho}_{u0}^{s_1s_2}(i) \\ \dots \\ \Delta\widehat{\rho}_{u0}^{s_{k-1}s_k}(i) \end{pmatrix} (3.39)$$

3.3.2 Calcul de la distance satellites - récepteurs

L'étape suivante est de déterminer les coordonnées de l'utilisateur à partir des distances satellites-utilisateur qu'on a déterminées précédemment :

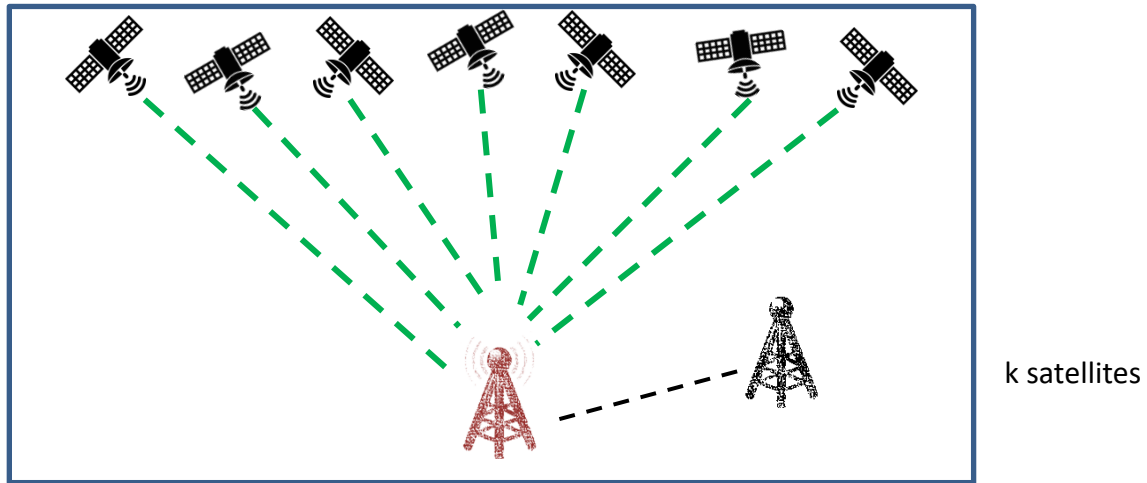


Figure 28- Utilisateur en visibilité k satellites et une VRS

Sur la Figure 28, les distances correspondent aux traits en vert pointillé qu'on connaît et à celle de la différence avec la VRS. On peut déterminer les coordonnées de l'utilisateur car on connaît la position des satellites et de la VRS à l'époque 0. D'abord en reprenant 3.38 :

$$\rho_u^s = \rho_{u0}^s + \rho_{uVRS}^s \quad (3.40)$$

$$\|\vec{v}_u\| = \rho_{u0}^s + \rho_{uVRS}^s = \sqrt{(x_s - x_u)^2 + (y_s - y_u)^2 + (z_s - z_u)^2} \quad (3.41)$$

3.3.2 Variation de la position

Ainsi on a k distances satellites-utilisateur. On peut donc augmenter nos vecteurs distances satellites/ VRS ainsi jusqu'à k éléments :

$$\begin{cases} \rho_{u0}^{s_1} + \rho_{uVRS}^{s_1} = \sqrt{(x_{s_1} - x_{u_1})^2 + (y_{s_1} - y_{u_1})^2 + (z_{s_1} - z_{u_1})^2} \\ \rho_{u0}^{s_2} + \rho_{uVRS}^{s_2} = \sqrt{(x_{s_2} - x_{u_2})^2 + (y_{s_2} - y_{u_2})^2 + (z_{s_2} - z_{u_2})^2} \\ \dots \\ \rho_{u0}^{s_k} + \rho_{uVRS}^{s_k} = \sqrt{(x_{s_k} - x_{u_k})^2 + (y_{s_k} - y_{u_k})^2 + (z_{s_k} - z_{u_k})^2} \end{cases} \quad (3.42)$$

On aura donc un nuage de points autour de notre position désirée selon trois axes :

$$x_u = \begin{pmatrix} x_{u_1} \\ \dots \\ x_{u_k} \end{pmatrix}, y_u = \begin{pmatrix} y_{u_1} \\ \dots \\ y_{u_k} \end{pmatrix} \text{ et } z_u = \begin{pmatrix} z_{u_1} \\ \dots \\ z_{u_k} \end{pmatrix} \quad (3.43)$$

On peut déterminer différents paramètres statistiques pour étudier la performance du réseau CORS utilisé et des corrections envoyées pour calculer la position. Cela permet d'avoir un aperçu des performances de la technique qu'on utilise comme la variance ou l'écart type. De plus, les équations d'observations GNSS que nous avons résolues dépendent du temps. Ainsi, ces vecteurs de positions sont eux aussi dépendant du temps :

$$x_u(i) = \begin{pmatrix} x_{u_1}(i) \\ \dots \\ x_{u_k}(i) \end{pmatrix}, y_u(i) = \begin{pmatrix} y_{u_1}(i) \\ \dots \\ y_{u_k}(i) \end{pmatrix} \text{ et } z_u(i) = \begin{pmatrix} z_{u_1}(i) \\ \dots \\ z_{u_k}(i) \end{pmatrix} \quad (3.44)$$

En effet, dans les paramètres de performance que nous avons établis au chapitre précédent, nous verrons que le temps est un facteur important. Nous portons attention à la performance de la continuité entre le NRTK et le PPP-RTK mais aussi à la continuité en PPP-RTK dans le temps avec les données d'initialisation NRTK en $i=0$ qui ne changent pas.

3.3.2 Calcul de la position et premiers résultats

D'un point vu mathématiques l'apport du NRTK permet d'avoir un nouveau vecteur de corrections qui est différencié avec les données de la station virtuelle générée en NRTK. Cela mène à une nouvelle évolution de la solution présentée à la Figure 29 :

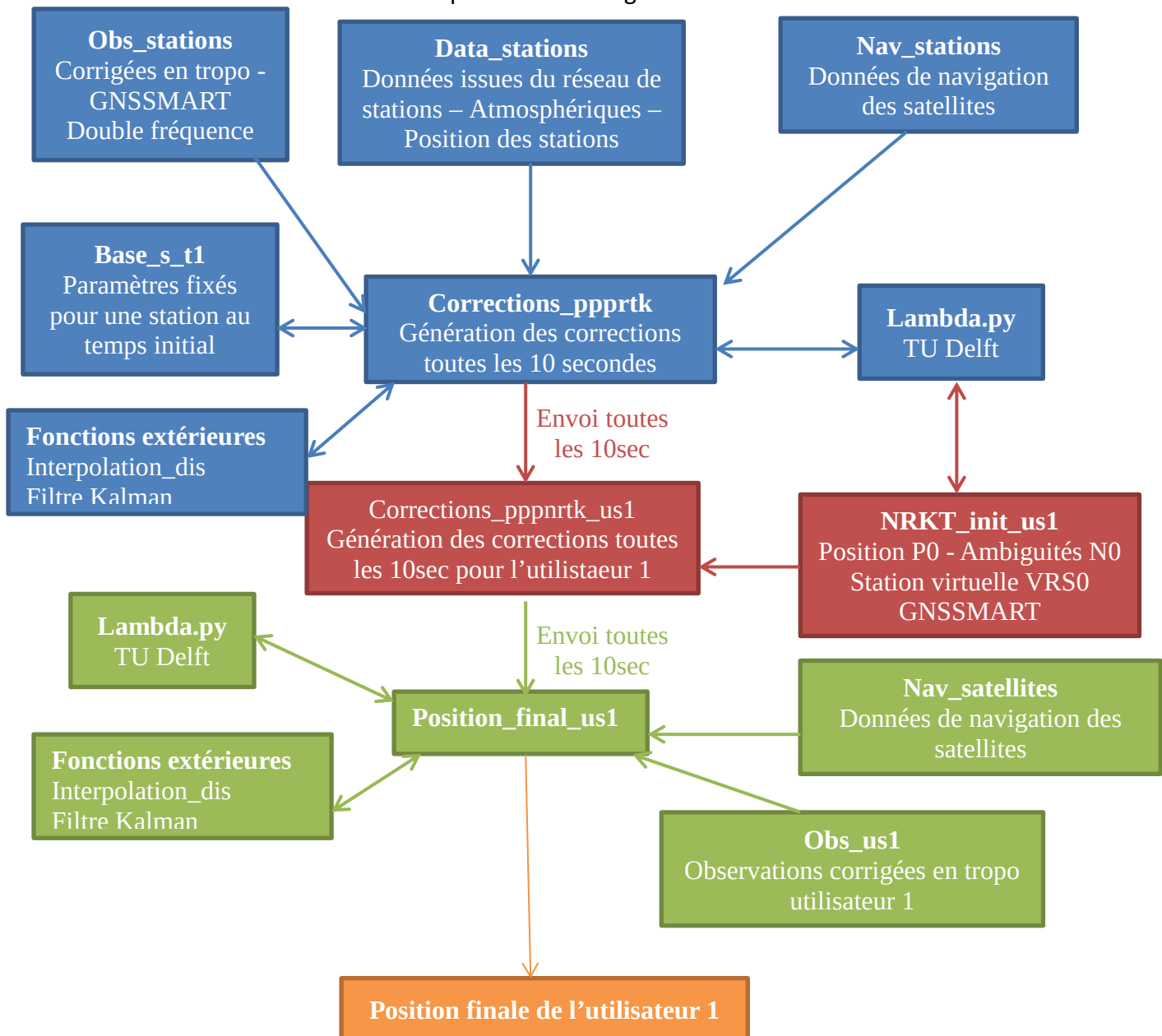


Figure 29 - Chaîne logicielle complète de la technique PPP-NRTK proposée

Ainsi, d'après la chaîne logicielle exposée, nous distinguons bien les trois différentes étapes de la technique PPP-NRTK proposée. Le processus continu de calcul de corrections se fait toutes les 10 secondes à chaque nouveau calcul avec des observations des stations et de l'utilisateur. On relèvera que l'étape la plus novatrice par rapport aux approches classiques est l'ajout du NRTK comme nouvelle étape d'initialisation pour des corrections PPP-RTK propre à un utilisateur (`corrections_pppnrtk_us1`) qui reçoit les corrections PPP-RTK générale (`corrections_pprtk`). L'avantage de cette approche est de lever la contrainte principale du PPP qui est le temps de convergence (TTFF). Il reste à montrer dans quelles conditions cette approche est « tenable ». Nous l'entendons au sens de : combien de temps peut-on rester en mode « PPP » sans que les performances se dégradent de manière trop importante ? C'est l'objet de notre quatrième chapitre.

4. Mise en place de la solution et tests de ma méthode PPP-NRTK

Après avoir abordé la partie mathématique et algorithmique de ma solution hybride GNSS, nous allons pouvoir la tester pour comprendre l'application et la mise en place de cette méthode PPP-NRTK « sur le terrain ».

Nous commençons par une réflexion plus approfondie menée sur les données des corrections SSR qui sont nécessaires et utiles à la résolution d'une solution hybride PPP-RTK, dans le contexte d'un défaut de rang. Un nouveau modèle de correction PPP-NRTK sera ainsi défini : une solution de calcul continue entre NRTK et PPP-RTK, notamment en cas d'interruption des corrections NRTK.

Comme exposé précédemment, la mise en place de la solution hybride proposée s'organise en trois étapes : d'abord la génération des corrections PPP-RTK à l'aide d'un réseau de stations de base, puis l'initialisation avec le calcul de la position en NRTK et enfin le calcul de la position de l'utilisateur en PPP-RTK. L'objectif est ainsi d'étudier cette solution hybride à l'aide des différents paramètres et critères de performances évoqués précédemment que l'on va pouvoir étudier grâce à deux tests. Sur la Figure 30, nous présentons un schéma qui résume les trois étapes de la chaîne logicielle de notre technique PPP-NRTK.

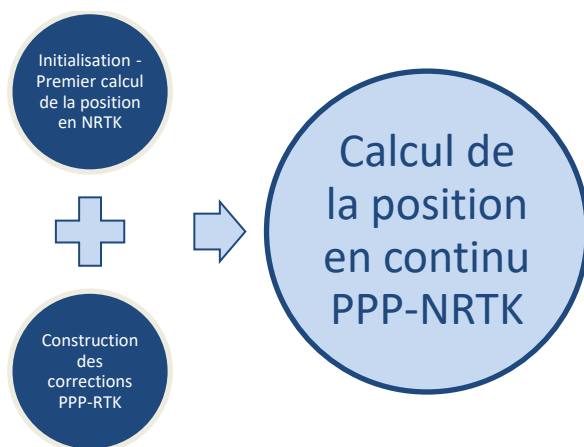


Figure 30-Technique PPP-NRTK

D'abord, je vais présenter la mise en œuvre de ces étapes avec les différentes contraintes imposées à chacune. Ensuite, j'aborderai la partie tests de la solution en m'appuyant sur les critères et les paramètres de performances que nous pourrions établir et analyser. Nous suivrons trois scénarii dont nous comparerons les résultats selon les différents critères de performance. Enfin, nous analyserons l'intérêt de l'application industrielle de notre technique PPP-NRTK pour Teria.

4.1 Construction et mise en place

Il s'agit de mettre en œuvre les trois étapes présentées dans la figure 30. Nous travaillerons essentiellement en post traitement pour la partie hybridation PPP-RTK : le calcul en NRTK sera possible en temps réel mais le reste sera compilé a posteriori. L'implémentation opérationnelle en temps réel n'étant pas l'objet des travaux de cette thèse.

Table 9-Etape du calcul de la position avec ma solution hybride

Etape 1	Etape 2	Etape 3
Côté réseau de PPP-RTK - Lien avec les corrections PPP-RTK	Initialisation de la position NRTK avec VRS (PO)	Calcul de la position avec les corrections PO et PPP-RTK
Post traitement/Python	GNSSMART	Post traitement/Python
Précision des corrections	Précision de la position	Précision de la position

Ces trois étapes reposent sur différents algorithmes et logiciels de calcul que nous avons présentés dans la partie 3. La plupart des algorithmes sur lesquels j'ai travaillé sont codés en python. Seule la partie initialisation NRTK sera faite à l'aide des corrections Geo++ générées grâce au programme GNSSMART. Selon qu'on se situe du côté du réseau ou de l'utilisateur, on s'intéressera à certains paramètres en particulier. On reviendra en détail sur cela à la fin de cette partie.

4.1.1 Génération des corrections PPP-NRTK avec un réseau de stations de référence

Comme vu précédemment, l'objectif du réseau est de fournir les corrections des paramètres dépendant uniquement des satellites. Il s'agit des biais matériels de code et de phase des satellites et les erreurs d'horloge des satellites. De plus, une attention particulière sera apportée aux données atmosphériques et au calcul des ambiguïtés doubles différenciées. En effet, cela pourra constituer des critères de comparaison au moment des tests des différents réseaux CORS utilisés et de pouvoir qualifier les corrections PPP-NRTK. Nous allons utiliser deux réseaux CORS différents pour mettre en place nos corrections PPP-NRTK. Ces deux réseaux sont situés dans deux zones distinctes en France métropolitaine. Ainsi, nous pourrons voir l'adaptabilité de notre technique et des corrections PPP-NRTK sur deux zones géographiques différentes utilisant les mêmes corrections. Cela permet entre autres d'avoir un aperçu de l'impact des corrections atmosphériques qui sont dépendantes d'une zone donnée.

- **Le réseau CORS-1**

Le réseau sert ainsi de source de données pour la construction des corrections PPP-RTK. Le nombre de stations importe peu, l'important c'est l'éloignement des stations entre elles dans la zone couverte. Cela aura un impact sur les corrections PPP-RTK calculées. Le premier réseau CORS que j'ai utilisé est composé de stations de référence situées dans le nord de la France dont les références sont : DOUL, CAUD, DOUA HRSN et COAU.

Table 10- Distances inter-stations du réseau CORS-1

Distance inter-stations	DOUL	CAUD	DOUA	HRSN	COAU
DOUL		60 km	44 km	110 km	95 km
CAUD	60 km		36 km	55 km	45 km
DOUA	44 km	35 km		95 km	80 km
HRSN	110 km	55 km	95 km		50 km
COAU	95 km	45 km	80 km	50 km	

Il a été mis en place en prenant en compte deux critères : une distance maximale de 120 km et une distance inter-stations correspondant à la distance généralement utilisée en NRTK. En effet, comme évoqué dans l'état de l'art, les stations sont éloignées en moyenne de 70 km. La distance inter-stations moyenne dans ce premier réseau de stations de base est de 67 km.

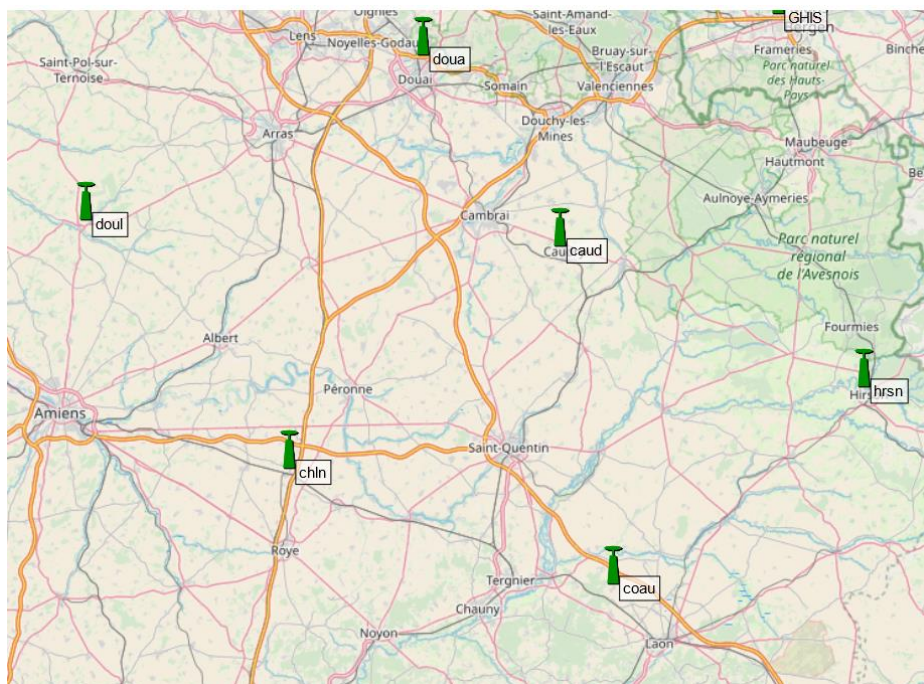


Figure 31 – Réseau CORS-1 composé de 5 stations (en rouge)

Ceci est le premier réseau CORS utilisé pour tester la technique hybride PPP-NRTK. Ce réseau est composé de cinq stations. Dans ce premier cas d'utilisation, on appellera ce réseau : CORS-1. Deux paramètres connus sont nécessaires à la génération des corrections : les données des observations et les positions des stations. Les coordonnées cartésiennes des stations utilisées sont résumées dans le tableau 11.

Table 11- Coordonnées cartésiennes ECEF des stations de référence du réseau CORS-1(en mètres)

	X	Y	Z
DOUL	4090657.653	168022.055	4874445.316
CAUD	4090103.156	243087.822	4871835.139
DOUA	4068144.055	220182.797	4891023.785
HRSN	4103970.430	292251.354	4857632.046
COAU	4130433.851	253914.196	4837313.646

Dans nos équations d’observations, c’est la distance station-satellite qui est prise en compte. Il faudra donc aussi prendre en compte la position des satellites au moment auquel on utilise les observations. Pour cela nous utiliserons les éphémérides des satellites, données issues des fichiers de navigation reçus en parallèle des observations de phase de code. En fonction du jour et de la période de tests, nous n’aurons pas toujours les mêmes satellites en visibilité. C’est pour cela que nous avons fixé le nombre de satellites utilisés, indépendamment de leurs identités, pour comparer les résultats dans des conditions les plus proches possibles. En effet, si on se réfère à l’état de l’art présenté précédemment, le nombre de satellites nécessaires en NRTK est de 6 et pour le PPP-RTK entre 6 et 10 pour la génération des corrections. Ainsi, le nombre de satellites que nous utiliserons pour nos tests sera de 10 satellites issus de deux constellations : GPS et Galileo.

- **Le réseau CORS 2**

Pour élargir l’application de notre solution PPP-NRTK, j’ai choisi d’utiliser un deuxième réseau de stations de référence : CORS-2. Là encore notre réseau est aussi composé de cinq stations dans le sud de l’Île de France : MLN, BRET, SIRT, GRIG et MLVL.

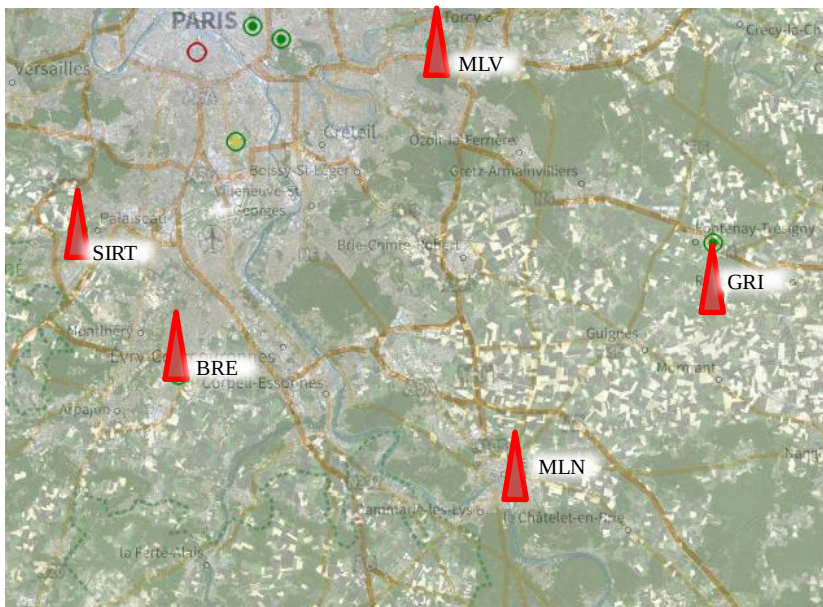


Figure 32- Réseau CORS-2 composé de 5 stations

La principale différence avec le réseau CORS-1 est la distance moyenne inter-stations. Dans ce cas, elle vaut 35 km et est donc plus petit. L’objectif ici est de pouvoir travailler avec des corrections PPP-RTK théoriquement plus précises et qui pourront éventuellement être utilisées avec des scénarii où l’utilisateur est mobile.

Table 12- Distance inter stations des stations du réseau CORS-2

Distance inter-stations	GRIG	MELN	BRET	SIRT	MLVL
GRIG		24 km	43 km	50 km	26 km
MELN	24 km		28 km	39 km	34 km
BRET	43 km	28 km		14 km	33 km
SIRT	50 km	39 km	14 km		31 km
MLVL	26 km	34 km	33 km	31 km	

Comme pour le réseau CORS-1, il est nécessaire de connaître la position des stations de référence. Les coordonnées en mètres de celles du réseau CORS-2 sont visibles dans le tableau 13.

Table 13 - Coordonnées cartésiennes ECEF des stations du réseau CORS-2 en mètres

	X	Y	Z
GRIG	4212020.615	4212020.615	4212020.615
MELN	4226432.992	4226432.992	4226432.992
BRET	4261945.930	4261945.930	4261945.930
SIRT	4213551.158	4213551.158	4213551.158
MLVL	4201577.208	4201577.208	4201577.208

- **Modèle Troposphérique – Observations corrigées**

Les observations utilisées sont déjà corrigées en termes d'erreurs troposphériques. En effet, les ZTD (Zenithal Total Delay), les paramètres d'erreur de la troposphère, présents dans les équations d'observation GNSS, sont estimés en amont et corrigés directement sur les observations. Les ZTD sont facilement estimés dans le module GNSSMART, figure 33, grâce au modèle Saastamonien (Zhang, et al., 2018). C'est un premier traitement que j'ai effectué sur les observations utilisées dans la partie réseau, mais aussi dans la partie utilisateur. Les variations de la troposphère sur une année sont faibles et ainsi les écarts sur plusieurs heures auront un impact réduit sur le calcul de la position. La densité des stations pour calculer les erreurs troposphériques a un impact sur la précision des corrections : nous prenons l'erreur ZTD comme référence et trois chiffres significatifs suffisent pour une précision centimétrique d'un point de vue utilisateur.

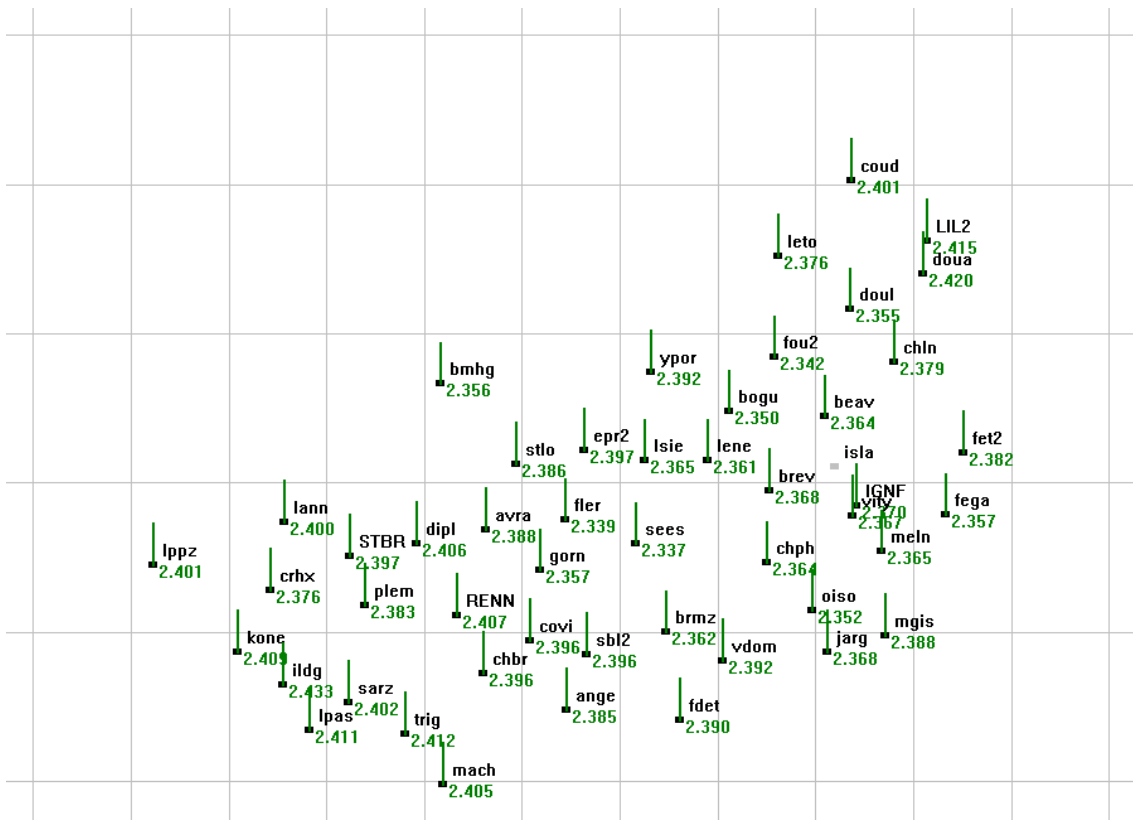


Figure 33-ZTD – Erreur troposphérique sur une partie de notre zone de test pour l’utilisation de corrections PPP-RTK (en mètres)

Nous avons donc les corrections troposphériques pour les différentes stations de références utilisées :

Table 14-ZTD pour les stations des deux réseaux CORS-1 et CORS-2

Stations	DOUL	COAU	COUD	HRSN	DOUA	MLN	MLVL	SIRT	BRET	GRIG
ZTD (en m)	2.355	2.380	2.401	2.382	2.420	2.365	2.357	2.367	2.368	2.370

Nous pouvons aussi avoir accès à des données similaires pour corriger les observables de code et de phase de l'utilisateur assez simplement. Cette étape de correction des erreurs troposphériques est faite en amont, nous n'aurons donc plus besoin de les prendre en considération lors de nos essais.

4.1.2 Initialisation et calcul d'une première position

Dans un premier temps, l'utilisateur se connecte au réseau Teria pour calculer sa position en NRTK. Il s'agit de la première étape, au temps $i=0$ dans les équations d'observations, comme démontré dans la partie précédente.

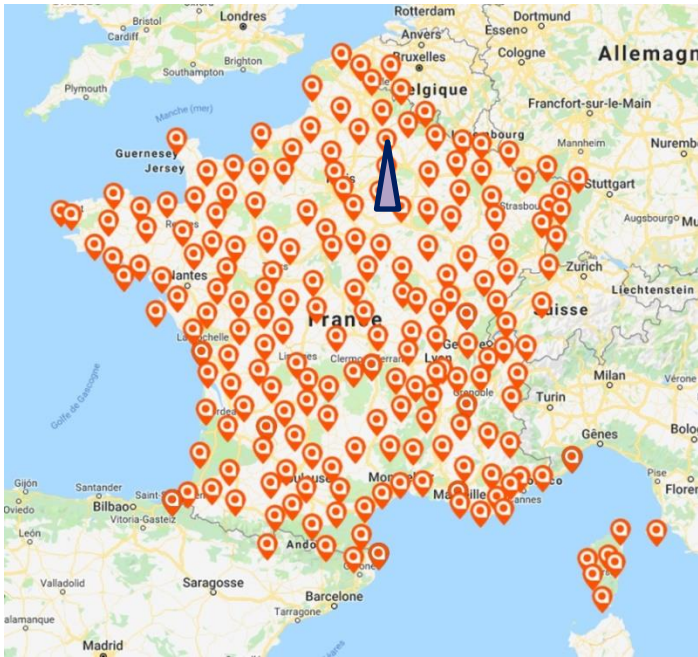


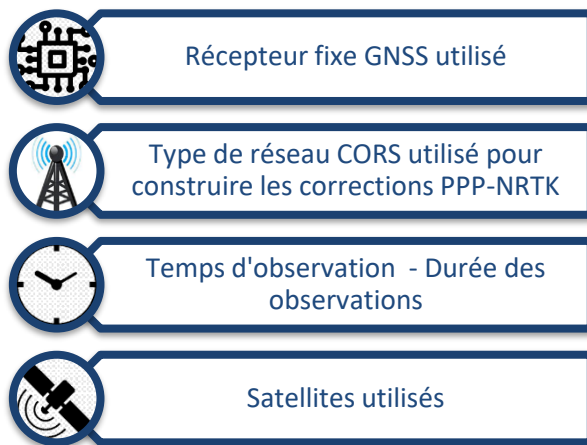
Figure 34- L'utilisateur connecté au réseau Teria

Le calcul de cette première position se fait à l'aide de la technique NRTK. Cette technique converge vers une précision centimétrique en quelques secondes. Nous choisissons une station virtuelle proche de l'utilisateur (<10km), ce choix se fait au sein du logiciel GNSSMART pour le calcul NRTK. On utilise un récepteur multi-constellations, ainsi l'utilisateur pourra calculer sa position à l'aide d'une quinzaine de satellites GPS, Galileo, GLONASS ou Beidou. Dans notre cas, nous nous limiterons à GPS et Galileo.

Ainsi, on mesure la position une première fois en NRTK grâce au principe de la VRS et du réseau Teria (Figure 34). En sortie de cette résolution nous utiliserons la position de l'utilisateur, soit les coordonnées cartésiennes de l'utilisateur $P_0 (x_0, y_0, z_0)$ et les ambiguïtés double différenciées N_0 . Elles constituent les données d'entrée pour le temps $t > 0$.

4.1.3 Résolution et calcul de la position – Hypothèses d'observations

Pour comparer les résultats obtenus entre nos différents réseaux CORS et les différents scénarii, nous établirons des hypothèses d'observations autour de certains critères. Pour les différentes phases de tests qui vont suivre j'utiliserai le modèle d'hypothèse suivant :



Nous allons, pour chaque scenario, poser un cadre d'application suivant :

- Le récepteur GNSS utilisé : selon le type de récepteur on ne pourra avoir qu'un certain nombre de paires de fréquences dépendant des constellations satellitaires disponibles. Certains récepteurs sont par exemple seulement GPS.
- Le type de réseau CORS : soit CORS-1 soit CORS-2 avec son système de corrections PPP-RTK associé.
- Le temps d'observation : durée de la session durant laquelle la position sera calculée. En effet, deux critères de performance dépendent de ce temps : le temps de convergence et la continuité de la technique utilisée. D'autant que notre solution hybride a pour but d'être une solution stable sur une longue durée.
- Les satellites utilisés : comme évoqué plus haut, nous avons fixé le besoin d'avoir 10 satellites pour la génération des corrections PPP-RTK et pour le calcul de la position d'un point de vue utilisateur.

4.2 Tests de la solution

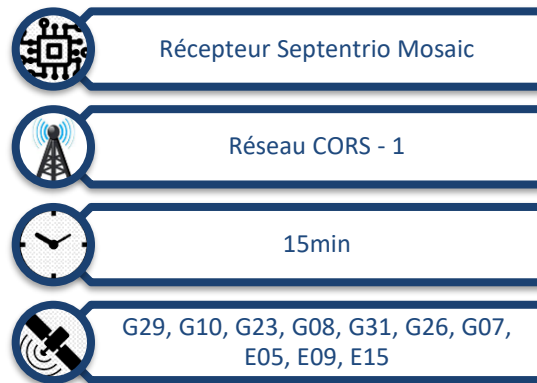
Maintenant, passons à l'étude des résultats de la technique hybride PPP-NRTK. Nous allons utiliser différents scénarii pour comparer les variations de positions selon le temps d'observation notamment.

4.2.1 Calcul de la position selon différents scenarios

Nous allons d'abord présenter les trois scénarios qui seront utilisés pour comparer les critères de performances de notre technique hybride PPP-NRTK.

- **Scenario A Récepteur fixe**

Le premier scénario est le calcul de la position d'un récepteur fixe. En effet, dans ce cas l'utilisateur ne bouge pas et l'objectif est d'initialiser sa position en NRTK.



Puis, à l'aide des corrections PPP-RTK de calculer sa position pour les temps $t > 0$. Cette session se déroulera pendant 1h. Toutefois, des phases d'observation du calcul de la position d'un point utilisateur aura lieu au bout de 15 min, puis 30 min et enfin au bout d'une heure. Les satellites utilisés seront issus des constellations GPS et Galileo et on en utilisera 10. Sur la figure 35, l'utilisateur (violet) se situe « au milieu » du réseau CORS-1. Ce sera une des stations qui fera office d'utilisateur.

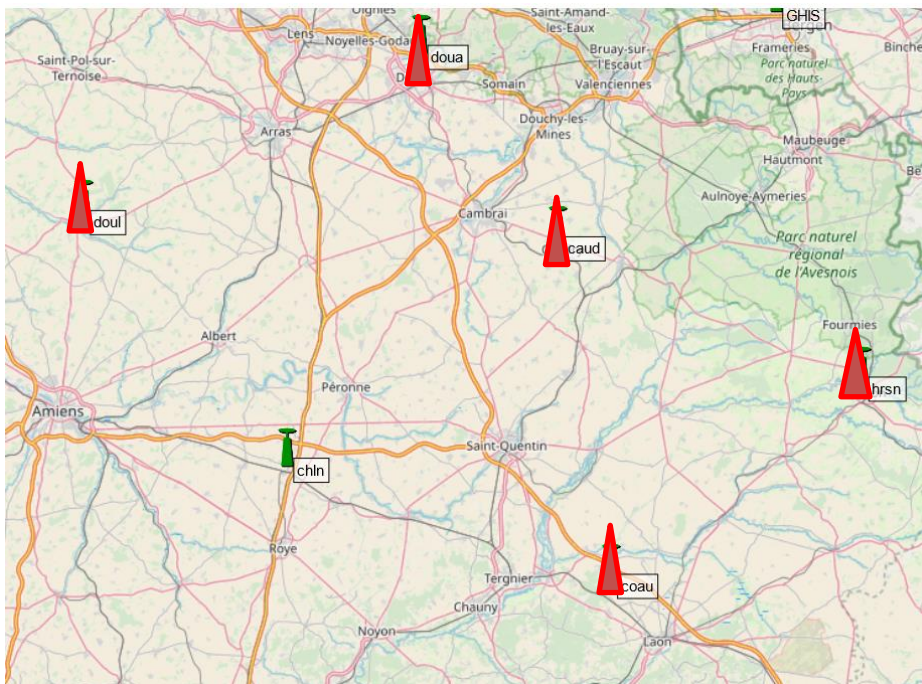


Figure 35 - Utilisateur dont la position est inconnue dans le réseau CORS-1

Les corrections seront générées grâce à nos cinq stations du réseau CORS-1 et aux satellites utilisés dans ce cas. 10 ont été sélectionnés parmi les 20 satellites visibles à ce moment (tableau 15). On a choisi ceux qui étaient tous en vue de nos stations de référence et de l'utilisateur sans prendre en considération d'autres critères comme la dilution de précision par exemple.

Table 15- Satellites visibles et utilisés dans le scénario A

	GPS	Galileo
Satellites en visibilité	G29, G10, G23, G08, G31, G26, G07, G15, G18, G27, G16	E05, E09, E15, E27, E30, E26, E34, E21, E13
Nombre satellites en visibilité	11	9
Satellites utilisés	G29, G10, G23, G08, G31, G26, G07	E05, E09, E15
Nombre satellites utilisés	7	3
Fréquences utilisés	L1/L2 + L1/L5	E1/E5

- **Scenario B Récepteur fixe**

Dans le scénario B, on travaille avec le réseau CORS-2 qui a une densité plus importante. L'utilisateur sera aussi fixe et on calculera sa position en NRTK pour initialisation et en PPP-RTK pour la durée de l'observation. Cette durée sera comme dans le scénario précédent : 1h avec des phases d'observations à 15 min, 30 min et au bout d'une heure.

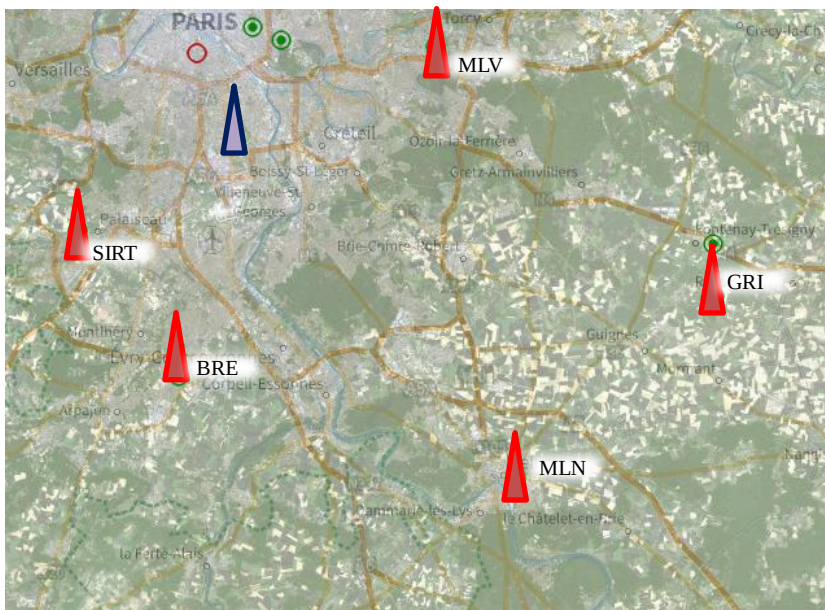
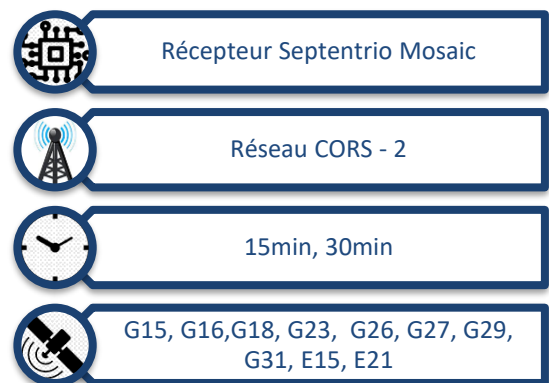


Figure 36- Utilisateur fixe dans le réseau CORS-2 - Scénario B

Dans ce cas, nous avons analysé les satellites en visibilité du réseau de stations de références CORS-2 et de l'utilisateur (Figure 36 et tableau 16).

Table 16- Satellites visibles et utilisés dans le scénario B

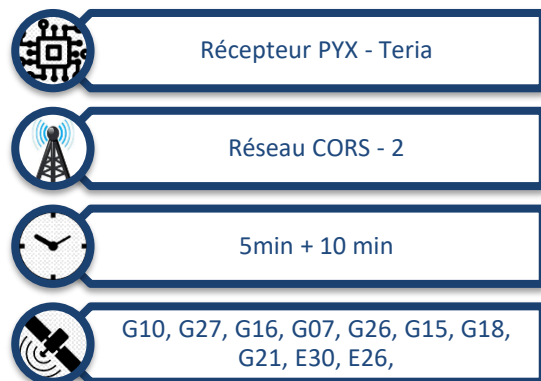
	GPS	Galileo
Satellites en visibilité	G07, G08, G10, G15, G16,G18, G23, G26, G27, G29, G31	E13, E15, E21, E27, E30, E34
Nombre satellites en visibilité	11	6
Satellites utilisés	G15, G16,G18, G23, G26, G27, G29, G31	E15, E21
Nombre satellites utilisés	8	2
Fréquences utilisés	L1/L2 + L1/L5	E1/E2

- **Scenario C Récepteur mobile**

Le dernier scénario est un scénario totalement différent des deux précédents : l'utilisateur est mobile. En effet, l'objectif est de voir comment réagit notre technique hybride PPP-NRTK quand l'utilisateur bouge et cherche à recalculer sa position avec seulement les corrections PPP-RTK. On pourra aussi voir dans quelles limites l'initialisation en NRTK reste valable pour nos applications. Nous utiliserons dans ce scénario le réseau CORS-2. En effet, c'est celui qui se situe autour des locaux de la société EXAGONE-Teria, ce qui était plus facile pour multiplier les tests. De plus, dans ce réseau, une station fixe non pris en compte dans le réseau servira de point de comparaison.

Au sein de ce scénario avec récepteur mobile, nous considérons deux types de mouvement pour voir les effets des corrections PPP-RTK :

- Mouvement rectiligne Scenario C1 écart de positions selon un axe (x)
- Mouvement en cercle Scenario C2- écart de position selon deux axe (xy)



Nous avons choisi d'opérer deux mouvements distincts pour lesquels nous aurons des mouvements de positions selon un ou deux axes. Ainsi, lors du calcul de la position nous pourrions voir le comportement de chaque coordonnée x ou y, ainsi que la hauteur qui change peu durant le mouvement.

Les deux scénarii se dérouleront ainsi : initialisation en NRTK, puis 5 minutes en mobile et enfin 10 minutes en fixe. Les corrections PPP-RTK sont recalculées toutes les 10 secondes, ainsi nous avancerons de manière à avoir une prise de point tous les mètres toutes les 10 secondes. Les choix de ce scénario sont dus aux observations faites dans l'état de l'art : en générale les techniques PPP-RTK convergent en une dizaine de minutes. Ainsi, nous aurons deux phases d'observations : d'abord l'utilisateur qui bouge pendant moins de dix minutes pour voir l'évolution du calcul de la position.

Dans le NRTK, en générale cette phase converge déjà au centimètre. La seconde phase d'observation sera un temps où l'utilisateur est fixe et pourra être comparé avec les deux premiers scénarios A et B.

De plus, les tests ont été faits au même moment. Dans les deux cas, nous serons dans le réseau CORS-2 et nous allons utiliser le modèle de corrections PPP-RTK-2 issu de ce réseau. Ainsi, nous aurons les mêmes biais de phase et de code des satellites et des erreurs d'horloges des satellites. Nous avons aussi la troposphère et l'ionosphère qui, même si les erreurs liées à leur influence sont théoriquement éliminées, seront ainsi es mêmes. Dans le tableau 17, j'ai résumé les satellites utilisés dans ces deux cas (C1 & C2).

Table 17- Satellites visibles et utilisés dans le scénario C

	GPS	Galileo
Satellites en visibilité	G10, G27, G16, G07, G26, G15, G18, G21, G23, G13, G02, G08, G30	E30, E26, E34, E21, E13
Nombre satellites en visibilité	13	5
Satellites utilisés	G10, G27, G16, G07, G26, G15, G18, G21	E30, E26,
Nombre satellites utilisés	8	2
Fréquences utilisés	L1/L2 + L1/L5	E1/E2

Le scénario C1 correspond à l'utilisateur mobile qui suit un mouvement rectiligne pendant 5 minutes comme sur la figure suivante (Figure 37).



Figure 37- Scénario C1, utilisateur mobile

Le second scénario où l'utilisateur est mobile est le scénario C2. Dans ce cas, l'utilisateur exécute un mouvement circulaire pendant 5 minutes dans la zone du réseau CORS-2 (Figure 38).



Figure 38- Scénario C2, utilisateur mobile

Dans les deux cas, nous sommes dans un environnement dégagé, c'est à dire sans arbre, sans hauts d'immeubles autour (canyon urbain) ou encore tout ce qui pourrait gêner la réception des signaux satellites ou créer des erreurs comme des multi-trajets. C'est une attention particulière à avoir dans le cadre des scénarii C1 et C2. En effet, quand l'utilisateur est fixe il est simple d'avoir une zone dégagée où installer le récepteur de manière fixe comme sur le toit d'un bâtiment. Dans le cas de l'utilisateur mobile, il faut veiller à rester dans une zone dégagée ce qui a bien été le cas dans nos deux scénarii présentés ci-dessus.

4.2.2 Critères de performance PPP-NRTK – Résultats de performance

Ensuite, nous allons pouvoir effectuer les tests de notre technique PPP-NRTK en utilisant les critères de performances comme point de comparaison entre les scénarii et ainsi entre les différentes corrections PPP-RTK des deux réseaux CORS utilisés.

D'abord, reprenons les critères de performances pour une technique PPP-RTK présentée dans la partie 2.2. Nous avons pour la partie réseau les mêmes critères de performance car notre technique utilise la même approche PPP-RTK. Toutefois, ce sont ceux d'un point de vue utilisateur qui vont changer. On aura notamment un critère en plus concernant l'initialisation en NRTK avec les calculs des ambiguïtés et de la position de l'utilisateur. De plus, dans le cas du PPP-NRTK, un critère de performance sera particulièrement important : celui de la continuité de la solution. En effet, d'abord dans le basculement NRTK et PPP-RTK puis dans le calcul PPP-RTK durant une longue période d'observation en gardant toujours la même initialisation NRTK.

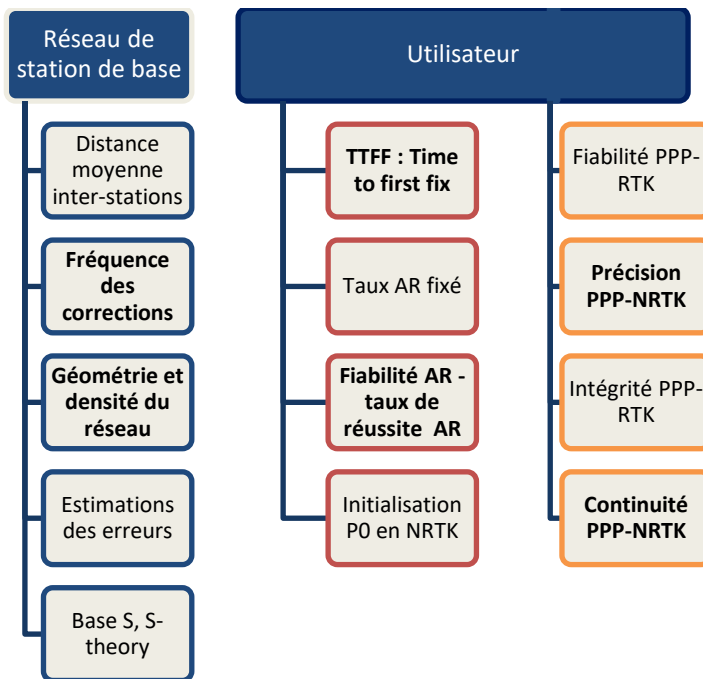


Figure 39- Paramètres et critères de performance PPP-NRTK

La figure 39 est une version de la figure 16 vu au chapitre 2 prenant en compte les spécificités de notre algorithme. Ainsi, pour analyser notre technique PPP-NRTK nous allons reprendre les deux parties réseau et utilisateur et pour chaque scénario étudier les différents critères de performance adéquats.

4.2.2.1 Réseau stations de base et génération des corrections

Nous allons maintenant étudier chaque critère de performance pour les deux réseaux CORS mis en œuvre dans les scénarii. Nous analyserons ainsi, les corrections PPP-RTK générées dans chaque réseau CORS qui seront utilisées dans les 3 scénarios.

- **Résolution des ambiguïtés**

Pour la résolution des ambiguïtés nous utilisons la technique LAMBDA et plus particulièrement le pack LAMBDA Python développé par l'université de Delft. La source du code vient essentiellement de Teunissen, Jonge, and Tiberius (1993). Notons que pour avoir des ambiguïtés double différenciées, nous avons fixé une ambiguïté dans notre réseau de stations de référence et d'un satellite. Pour le réseau CORS 1 il s'agit de la station DOUA et du satellite G29 ou E05. C'est grâce à ces données que nous avons pu construire ces ambiguïtés doubles différenciées et ainsi avoir des ambiguïtés entières dans notre système d'équations. Par conséquent, il sera possible d'appliquer la méthode LAMBDA. Cette résolution est à faire pour chaque fréquence par constellation de satellite. En effet, nous utilisons une méthode double fréquence pour les deux constellations de satellites utilisés (GPS et Galileo). Nous allons ici présenter pour chaque réseau un exemple des ambiguïtés double différenciées pour une fréquence (L1) sur la constellation pour chaque réseau CORS. Nous allons faire la résolution sur une station du réseau comme COAU.

Pour la station COAU du réseau CORS-1 on doit déterminer 10 ambiguïtés. On applique l'algorithme LAMBDA. On obtient en sortie plusieurs éléments en lien avec la résolution des ambiguïtés.

D'abord on a la matrice de variance-covariance de décorrélation des ambiguïtés. Ci-dessous un exemple de ce type de matrice :

$$Q_{var.cov.cors1}^{decorr,amb} = \begin{pmatrix} 6.24940757e^{-2} & 1.00952566e^{-2} & 4.07413914e^{-3} & \dots & -2.71840121e^{-3} & -1.22137018e^{-2} & -1.53694616e^{-2} & -5.50796608e^{-3} \\ 1.00952567e^{-2} & 6.14401290e^{-2} & -5.77390859e^{-3} & \dots & 8.43220923e^{-3} & -1.21424967e^{-2} & 1.83814574e^{-2} & -1.05663853e^{-2} \\ 4.07413972e^{-3} & -5.77390857e^{-3} & 4.90784577e^{-2} & \dots & 1.16589921e^{-2} & 3.26683343e^{-3} & -7.27954607e^{-3} & 3.13729816e^{-3} \\ -9.12702250e^{-3} & 1.93318414e^{-2} & 3.36061975e^{-3} & \dots & -2.96653525e^{-6} & -1.54928780e^{-2} & 1.33208932e^{-5} & 3.56888604e^{-3} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ -2.71840101e^{-3} & 8.43220950e^{-3} & 1.16589923e^{-2} & \dots & 5.20518399e^{-2} & 1.51025635e^{-2} & 1.22159349e^{-2} & -2.34100096e^{-3} \\ -1.22137013e^{-2} & -1.21424968e^{-2} & 3.26683313e^{-3} & \dots & 1.51025631e^{-2} & 4.67451635e^{-2} & 3.48646300e^{-3} & 1.09465438e^{-2} \\ -1.53694614e^{-2} & 1.83814574e^{-2} & -7.27954589e^{-3} & \dots & 1.22159350e^{-2} & 3.48646313e^{-3} & 3.94519127e^{-2} & -9.06334603e^{-3} \\ -4.94149359e^{-3} & -1.99176912e^{-2} & 9.61767252e^{-3} & \dots & 9.74038438e^{-4} & 6.02270338e^{-3} & -6.29054225e^{-3} & 8.10964698e^{-3} \\ -5.50796652e^{-3} & -5.50796652e^{-3} & -1.05663854e^{-2} & \dots & -3.25942454e^{-4} & -2.34100094e^{-3} & 1.09465436e^{-2} & 8.10964707e^{-3} \end{pmatrix}$$

Ce type de matrice est une étape clé dans la résolution des ambiguïtés.

Ensuite, nous obtenons les ambiguïtés fixées associées pour la station COAU :

$$N_{fix}^{COAU} = \begin{pmatrix} -28537 \\ 65473 \\ 38692 \\ 4939 \\ -29228 \\ -504 \\ -22237 \\ 51018 \\ 30150 \\ 3849 \end{pmatrix}$$

Nous avons donc fixé 10 ambiguïtés doubles différenciées correspondant à nos 10 équations de notre système résultant des observations GNSS (10 satellites sur la fréquence L1 ou E1). Rappelons qu'un des critères de performance que nous prenons en compte est le taux de réussite de la résolution des ambiguïtés entières. Dans ce cas, notre premier modèle de corrections PPP-RTK donne :

$$N_{succes,rate}^{cors1} = 0.9499$$

Pour l'ensemble des ambiguïtés entières résolues nous avons les taux de réussite suivants pour la station COAU, nous gardons seulement 4 chiffres après la virgule, précision suffisante pour les comparer :

Table 18- Taux de réussite AR pour la station COAU du réseau CORS-1

Fréquence	L1 GPS	L2 GPS	L5 GPS	E1 Galileo	E5 Galileo
$N_{succes,rate}^{cors1}$	0.9499	0.9771	0.9671	0.9566	0.9688

Ceci, est un exemple de résolution d'ambiguïtés, pour une station de notre réseau. Nous avons la résolution sur plusieurs fréquences comme nous différencions les signaux des constellations (GPS et Galileo) pour l'interopérabilité. On peut faire de même sur l'ensemble des stations des deux réseaux CORS. Sur l'ensemble des stations nous avons des taux de réussite supérieurs à 93%. J'ai détaillé l'ensemble des taux de réussites de nos différentes résolutions d'ambiguïtés dans les annexes. Notons que ces résultats sont plutôt bons et encourageant pour la suite de la résolution.

Maintenant que les ambiguïtés sont déterminées, il est possible de générer les corrections PPP-RTK et ainsi de déterminer les biais matériels des satellites et les erreurs d'horloges de ces mêmes satellites, soit les vecteurs de corrections PPP-RTK qui seront fournis à l'utilisateur pour lui permettre de calculer sa position.

- **Corrections PPP-RTK – Biais matériel satellite sur le code**

Nous allons déterminer pour chaque satellite sur les deux fréquences les erreurs liées au biais de code du satellite. Toutefois, nos corrections sont en réalité des différences de biais de deux satellites. La résolution est aussi possible grâce au fait d'avoir fixé cette valeur pour un satellite sur le biais de code IF (Ionosphere Free) c'est-à-dire différencié entre les deux fréquences (L1/L2 ou L1/L5 ou E1/E5). Je vais exposer les différentes étapes pour une paire de satellites du réseau CORS-1. Ensuite, je donnerai juste les résultats des deux fréquences avec l'ensemble des paires des satellites pour ce même réseau. Enfin, je mets en annexe l'ensemble des résultats des biais de code des paires de satellites pour les deux formats de corrections PPP-RTK. Nous ferons de même pour les biais de phases matériels des satellites.

Nous allons expliciter la méthode pour le premier réseau sur la fréquence L1. En effet, comme nous fixons un biais de code IF il est nécessaire de le faire sur deux fréquences. On arrive donc à avoir une paire de satellites (ici on sera G29/G10 de la constellation GPS) sur la fréquence L1 avec un biais IF L1/L2.

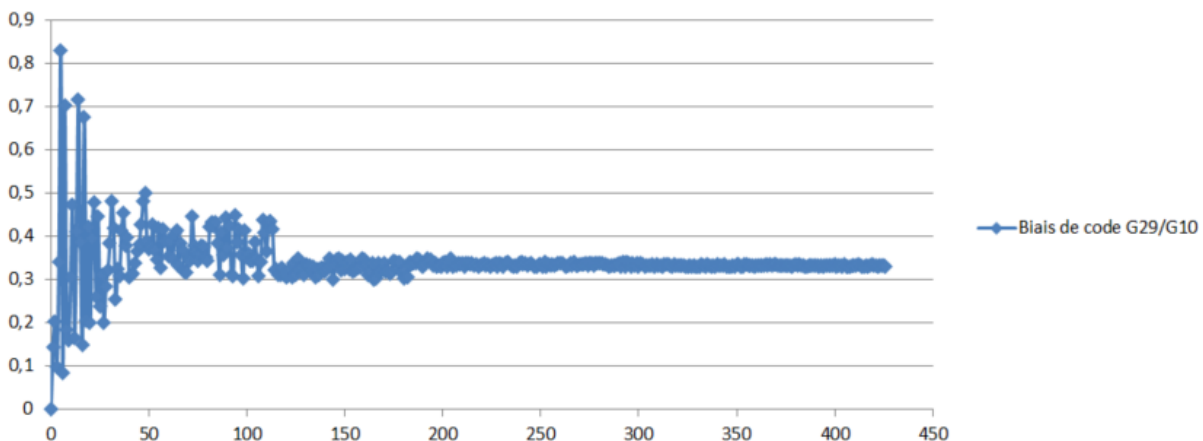


Figure 40- Biais de code différencié G29/G10 sur la fréquence L1, avec biais IF L1/L2 en mètres

Ainsi, on voit qu'au bout de trois minutes on arrive à avoir un biais de code valable. Il faut répéter ce processus sur toutes les paires de satellites sur les différentes fréquences : L1/L2 et L1/L5 pour les satellites GPS et E1/E5 pour les satellites Galileo. Pour certaines paires de satellites il arrive que le biais de code ne converge pas aussi vite et qu'il faille attendre au-delà de 3 minutes. En effet, cela est dû à la donnée d'initialisation pour le biais IF. La qualité de son estimation influe sur la rapidité de convergence du biais de code. C'est la valeur la plus facile à estimer dans notre système de corrections PPP-RTK.

Ci-dessous, j'ai résumé dans un tableau les biais de code différencié entre les satellites pour les fréquences L1/L2 au sein du réseau CORS-1.

Table 19- Biais de code pour le réseau CORS-1 sur les fréquences L1/L2 en mètres

L1/ L2	G29	G10	G23	G08	G31	G26	G07
G29	0,000	0,777	-0,623	-0,457	-0,054	0,265	-0,632
G10	-0,388	0,000	-0,101	0,2903	0,412	-0,376	-0,299
G23	0,014	0,037	0,000	-0,199	-0,091	0,509	0,284
G08	-0,829	0,507	0,599	0,000	-0,277	-0,501	0,54
G31	0,517	0,245	-0,061	0,812	0,000	0,061	-0,221
G26	-0,009	-0,482	-0,045	-0,149	0,074	0,000	0,604
G07	-0,806	0,448	-0,369	-0,104	-0,205	0,185	0,000

Nous voyons que celui-ci peut être négatif ou positif selon le sens de la différence. La différence de biais entre deux satellites est comprise entre -1 et 1. Cet écart peut être plus ou moins grand selon les satellites. Toutefois, il faut noter qu'il n'aura pas forcément d'impact direct sur le calcul de la position de l'utilisateur. En effet, le biais de code sert à estimer les erreurs d'horloge et le biais de phase. Donc il a de l'importance pour la génération des corrections mais du côté utilisateur on utilisera surtout les deux autres biais.

Les résultats sont similaires pour les autres fréquences et sont présentés en annexe.

Pour le deuxième vecteur de corrections PPP-RTK sur le réseau CORS-2, nous aurons des erreurs de code similaires pour les fréquences L1/L2 pour les satellites GPS visibles par ce réseau.

Table 20- Biais de code pour le réseau CORS-2 sur les fréquences L1/L2 en secondes

L1/ L2	G15	G16	G18	G23	G23	G27	G29	G31
G15	0,000	0,926	-0,778	-0,947	-0,299	0,117	-0,858	-0,875
G16	0,804	0,000	0,042	-0,069	0,332	-0,244	-0,764	0,425
G18	-0,55	-0,91	0,000	-0,082	-0,848	0,304	-0,959	-0,747
G23	-0,664	0,951	0,497	0,000	0,217	0,711	-0,821	0,526
G26	-0,426	-0,599	0,627	0,613	0,000	0,796	-0,572	0,831
G27	0,767	-0,041	-0,780	0,293	0,656	0,000	-0,876	0,351
G29	0,318	-0,511	-0,007	0,866	-0,911	-0,316	0,000	-0,302
G31	0,162	-0,665	0,904	-0,22	0,146	0,067	-0,668	0,000

- **Corrections PPP-RTK – Biais d'horloge satellites**

De même que pour le biais de code il est nécessaire de fixer l'horloge d'un satellite pour déterminer ainsi les différences des paires des satellites. Les horloges des satellites ne dépendent pas de la fréquence donc ici nous aurons seulement à définir les erreurs des horloges des satellites présents dans les corrections PPP-RTK1 et des corrections PPP-RTK2.

Ensuite, pour les prendre en compte dans notre résolution il faut préalablement les multiplier par la vitesse de la lumière dans le vide c (299792458 m/s) avant de les diffuser, les biais d'horloge étant exprimés en secondes. Précisons que nous avons différencié nos équations d'observations GNSS, comme expliqué dans le chapitre 3, ainsi ce sont les erreurs d'horloge différenciées que nous allons

déterminer. Nous présentons ci-dessous les erreurs d'horloges (biais d'horloges multipliés par c) en mètres.

Table 21 -Erreurs d'horloges pour le réseau CORS-1 en mètres

Horloge	G29	G10	G23	G08	G31	G26	G07
G29		-0,139	0,142	0,009	0,651	-0,457	0,879
G10			-0,602	-0,413	0,247	-0,814	-0,227
G23				0,429	-0,092	0,725	0,102
G08					0,699	0,659	0,813
G31						0,559	-0,918
G23							-0,794

De même pour le réseau CORS-2, on détermine les erreurs d'horloge des satellites.

Table 22 -Erreurs d'horloges pour le réseau CORS-2 en mètres

Horloge	G15	G16	G18	G23	G23	G27	G29	G31
G15		0,926	-0,778	-0,947	-0,299	0,117	-0,858	-0,875
G16			0,042	-0,069	0,332	-0,244	-0,764	0,4253
G18				-0,082	-0,848	0,304	-0,959	-0,747
G23					0,217	0,718	-0,821	0,526
G26						0,796	-0,572	0,831
G27							-0,876	0,350
G29								-0,302

Ce sont ces erreurs d'horloges que nous allons envoyer à l'utilisateur. Nous voyons que nous avons les mêmes ordres de grandeur que les erreurs d'horloges présentées dans la partie 2 sur les techniques PPP-RTK.

- **Corrections PPP-RTK – Biais matériel satellite sur la phase**

Le dernier paramètre de mon vecteur de correction PPP-RTK à déterminer est le biais de phase. Nous continuons à résoudre nos paramètres estimés présentés dans la partie précédente. Les calculs sont les mêmes pour les réseaux CORS-1 et CORS-2. Nous allons illustrer les biais de phase seulement pour le réseau CORS-2 et pour les fréquences L1/L2.

Grâce à nos données fixés et définis dans notre base S, nous arrivons à déterminer les biais de phase estimés.

Les corrections PPP-RTK sont recalculées en permanence. En effet, les différents biais ne seront valables que pendant un certain temps. Les corrections seront donc fournies de manière régulière à l'utilisateur. Ce calcul aura lieu tous les 10 secs. Toutefois, pour le biais de code d'un satellite, il n'est pas nécessaire de le renvoyer aussi souvent. Une mise à jour du biais de code toutes les minutes (donc six fois moins fréquente) sera amplement suffisante, car le biais de code n'a pas un impact direct sur le calcul de la position. Il permet surtout de définir le biais de phase dans les premières époques. Ainsi, sur un temps plus long nous renouvelerons les biais de code toutes les minutes mais les biais de phases toutes les 10 secondes.

Table 23- Biais de phase pour le réseau CORS-2 sur les fréquences L1/L2 en degrés

L1/ L2	G29	G10	G23	G08	G31	G26	G07
G29	0,635	0,168	0,485	0,761	0,569	-0,860	0,610
G10	0,635	0,433	0,864	-0,644	0,016	-0,348	0,320
G23	-0,758	0,433	0,557	0,718	-0,662	0,958	-0,605
G08	-0,354	-0,081	0,557	-0,954	-0,156	0,836	0,886
G31	0,687	-0,735	0,929	-0,954	0,89289	0,621	-0,440
G26	0,432	0,390	0,297	0,504	0,89289	0,1052	-0,720
G07	-0,577	-0,517	0,821	-0,312	-0,86414	0,1052	0,610

Les ordres de grandeur pour les biais de phase du réseau CORS-1 et des autres fréquences sont similaires. Ces résultats sont présentés en annexe.

Maintenant, notre vecteur de correction des biais $V_{\text{Corr}} = [\widetilde{dt}^{ss}, \widetilde{d}^{ss}, \widetilde{\delta}^{ss}]$ est connue pour toutes les paires de satellites possibles dans les réseaux CORS-1 et CORS-2.

4.2.2.2 Point de vue Utilisateur

Maintenant nous avons nos corrections PPP-RTK1 et PPP-RTK-2 qui sont opérationnelles. Nous allons passer du côté utilisateur pour calculer sa position selon les différents scénarios.

- **Initialisation en NRTK**

Dans tous les cas, l'initialisation se fera en NRTK et sera presque la même. Comme attendu, la position calculée va converger très vite : on va atteindre un écart de quelques centimètres en moins de 2 minutes. Ainsi il est possible de fixer la position P0 et ses ambiguïtés. Ce sont ces deux données qui vont nous intéresser pour la construction de la nouvelle base S' pour avoir un nouveau vecteur de corrections PPP-NRTK :

$$V_{\text{corr}} = [\widetilde{dt}^{ss}, \widetilde{d}^{ss}, \widetilde{\delta}^{ss}, \rho_{uVRS}^{ss}(0), \Delta N_0] \quad (4.32)$$

Les figures 41 à 43 dessous montre les résultats du calcul de la position NRTK pour les quatre scénarii :

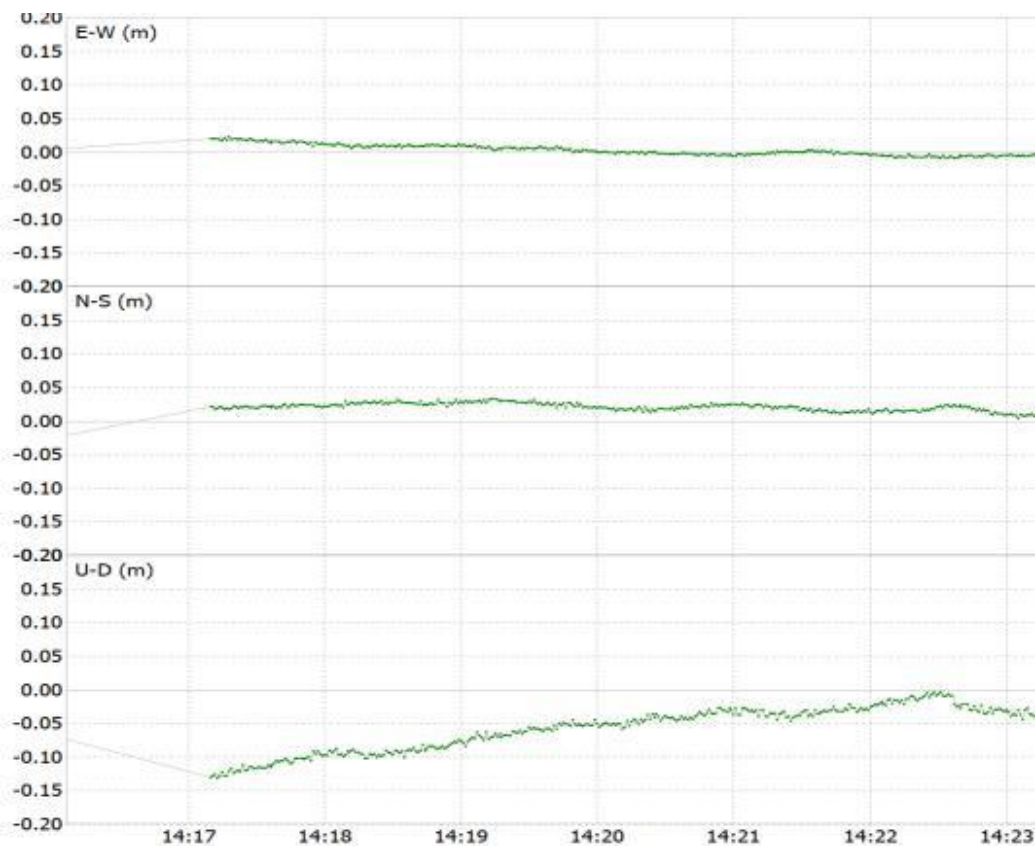


Figure 41- Calcul de la position P0 en NRTK Scénario A

Sur cette première résolution NRTK, on voit que les variations les plus importantes sont sur l'axe Z. Cela est dû à l'observation classique des techniques de positionnement GNSS qui sont moins robuste sur la verticale. Dans la suite de notre étude nous ne nous intéresserons qu'aux coordonnées X et Y. Cela limite un peu la portée de la méthode, mais le problème de la verticale reste essentiellement lié à la DOP (dilution de précision) verticale. On pourra espérer améliorer l'approche en verticale avec plus de satellites (ici on a une dizaine de satellites ce qui correspond à ce que reçoit un récepteur GPS seul standard aux latitudes moyennes).

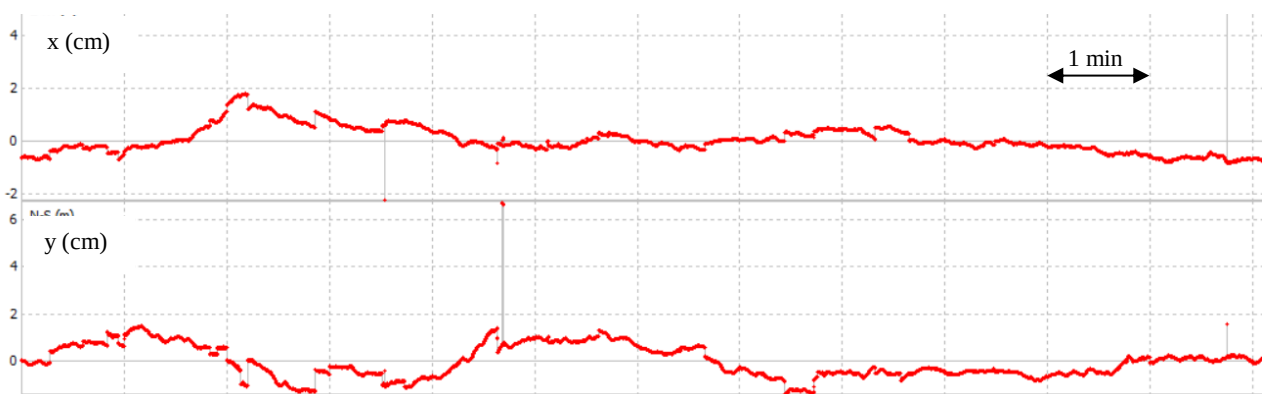


Figure 42-Calcul de la position P0 en NRTK Scénario B

Sur les figures 41 et 42 correspondants aux deux premiers scénarios, on observe une précision de moins d'2 cm quasiment en instantanée (< 1min). C'est bien ce que l'on attend en termes de performance NRTK.

Nous procédons à la même initialisation NRTK sur les scénarios C1 et C2. Le point de départ des deux scénarios où l'utilisateur est mobile partent du même endroit et utilisent la même initialisation NRTK.

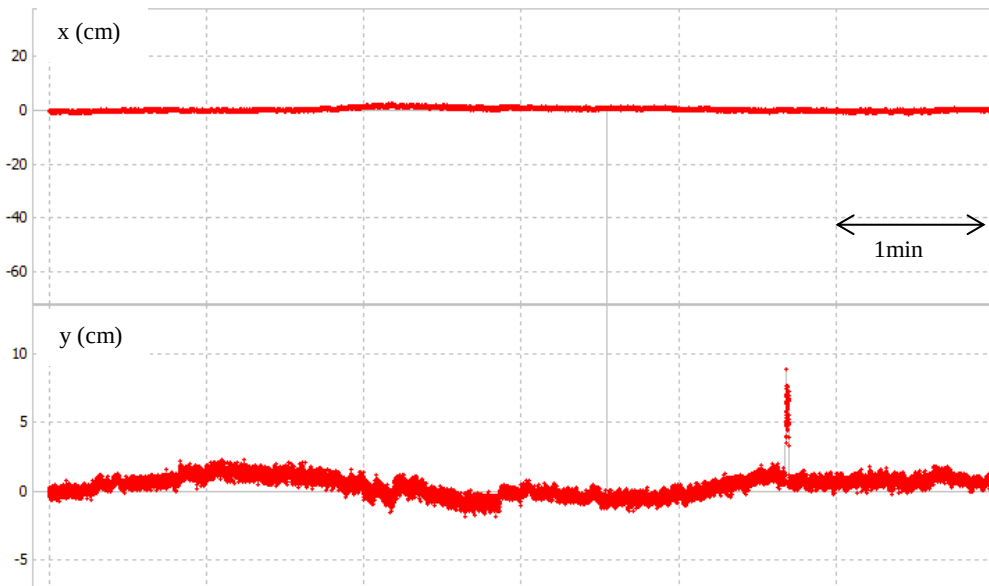


Figure 43-Calcul de la position P0 en NRTK Scénarios C1 et C2

Sur la figure 43, nous avons des résultats similaires aux deux premières initialisations NRTK : une convergence rapide ($< 1\text{min}$) et une bonne précision ($< 3\text{cm}$) sur les axes x et y . Toutefois, on observe une particularité autour de $4\text{min}30$ de résolution NRTK : un pic avec une erreur qui dépasse les 5cm . Cet écart peut s'expliquer de plusieurs manières mais comme nous commençons notre initialisation NRTK proche d'un bâtiment, la plus probable dans ce cas est l'impact des multi-trajets dû à un changement rapide de l'environnement (passage d'une voiture ou d'un piéton près de l'antenne). Comme nous prenons la position NRTK P0 à 1min ce pic n'aura pas d'incidence sur l'application de notre hybridation.

Les valeurs récupérées suites à cette initialisation pour chacun des scénarios sont donc :

- Double ambiguïtés différenciées & double différences de phase : la différence de phase et d'ambiguïtés entre la VRS et l'utilisateur et deux satellites, ces deux matrices de données seront utilisées pour la partie PPP-RTK
- Position P0 : la solution converge très vite en moins d'une minute. Ainsi, la position initiale enregistrée sera P0 au bout d'une minute, nous utiliserons donc en PPP-RTK la position P0, dans nos équations d'hybridation.

- **Temps de convergence PPP-NRTK**

Le premier critère de performance que nous allons étudier est celui du TTFF (Time to first fix). Il s'agit du temps de convergence pour fixer les ambiguïtés et ainsi déterminer la position de l'utilisateur. Une fois que nous avons reçu les corrections PPP RTK, l'utilisateur peut résoudre ses ambiguïtés grâce aux données d'initialisation en NRTK et ainsi calculer sa position. Le TTFF est relié aux taux de réussite des ambiguïtés. Si celui-ci est supérieur à 90% nous partons du principe que la solution est valide. A priori, vu qu'on détermine les ambiguïtés pour le PPP avec le NRTK, on s'attend à une convergence rapide.

Scénario A

Dans le premier scénario, Scénario A, l'utilisateur est fixe et reçoit des corrections PPP-RTK1 toutes les 10 secondes (pour les biais de phase et horloges) et 1 minute pour les biais de code.

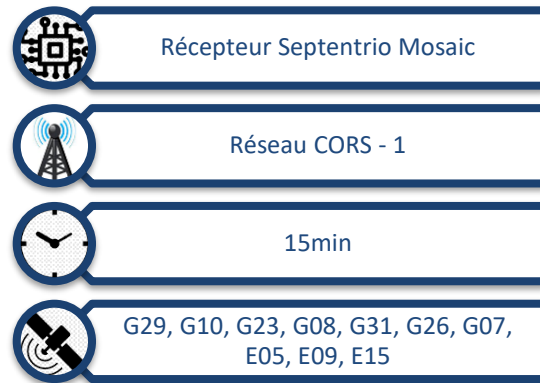


Table 24- Taux de réussite de AR , Scénario A au bout de 15min

Fréquence	L1 GPS	L2 GPS	L5 GPS	E1 Galileo	E5 Galileo
$N_{success}^{cors1}$, rate	0.9402	0.9544	0.9744	0.9887	0.9455

Les taux de réussite de la résolution des ambiguïtés sont tous supérieurs à 90 % on peut donc regarder ce que cela donne sur le temps de convergence avec le calcul en termes de précision.

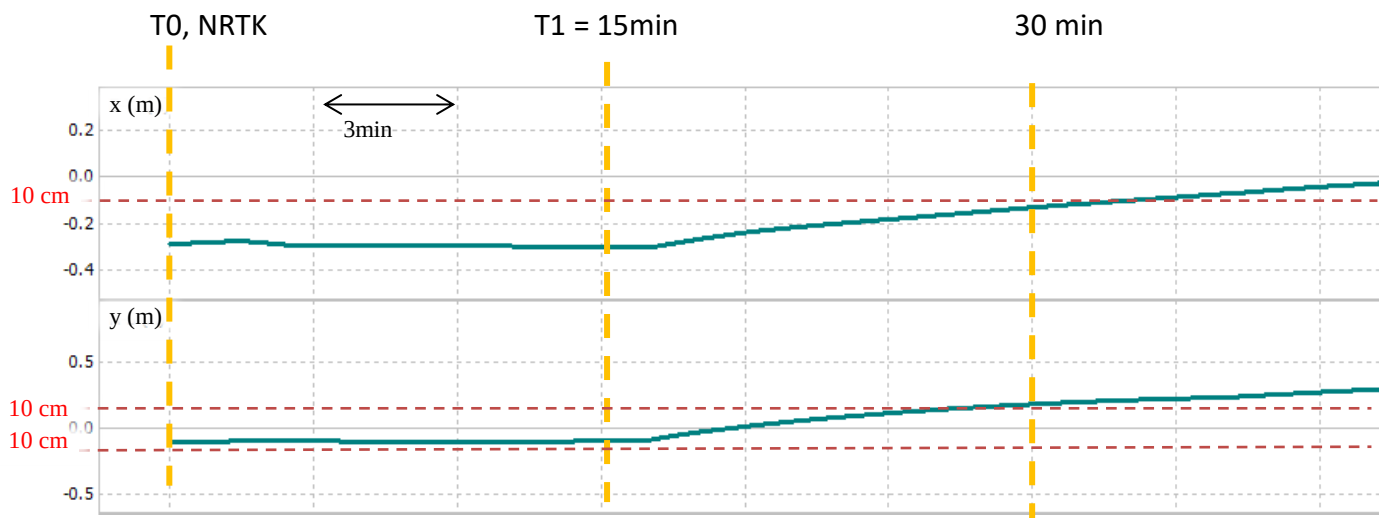


Figure 44- Convergence PPP-NRTK dans le scénario A

Dans le premier scénario, le temps d'observation est de 15min. On peut voir que la solution converge et atteint une précision de moins de 20cm en moins de 15min pour l'axe x et en moins de 3 minutes pour l'axe y. De plus, quand on atteint les 30min, on a une position qui converge avec une précision de moins de 10cm en x et en seulement 15min pour y. Cela correspond aux performances que nous visions dans l'introduction pour l'axe y. En effet, nous souhaitons une précision inférieure à 10 cm et en moins de 15 min. Il semble donc que dans ces conditions, l'initialisation NRTK de notre approche remplisse sa fonction.

Scénario B

Dans le second scénario, Scénario B, l'utilisateur est aussi fixe et reçoit des corrections PPP-RTK toutes les 10 secondes (pour les biais de phase et horloges) et 1 minute pour les biais de code. Cette fois, le temps d'observation est plus loin. Toutefois, les ambiguïtés sont déterminées dans les mêmes temps, et les taux sont aussi supérieurs à 93% en moins de 15min.

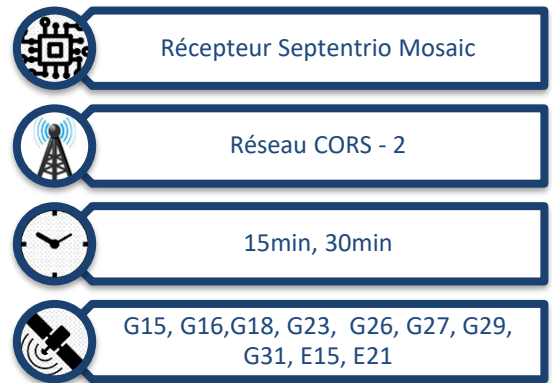


Table 25- Taux de réussite de AR , Scénario B

Fréquence	L1 GPS	L2 GPS	L5 GPS	E1 Galileo	E5 Galileo
$N_{success}^{cors2}$	0.9588	0.9655	0.9887	0.9355	0.9544

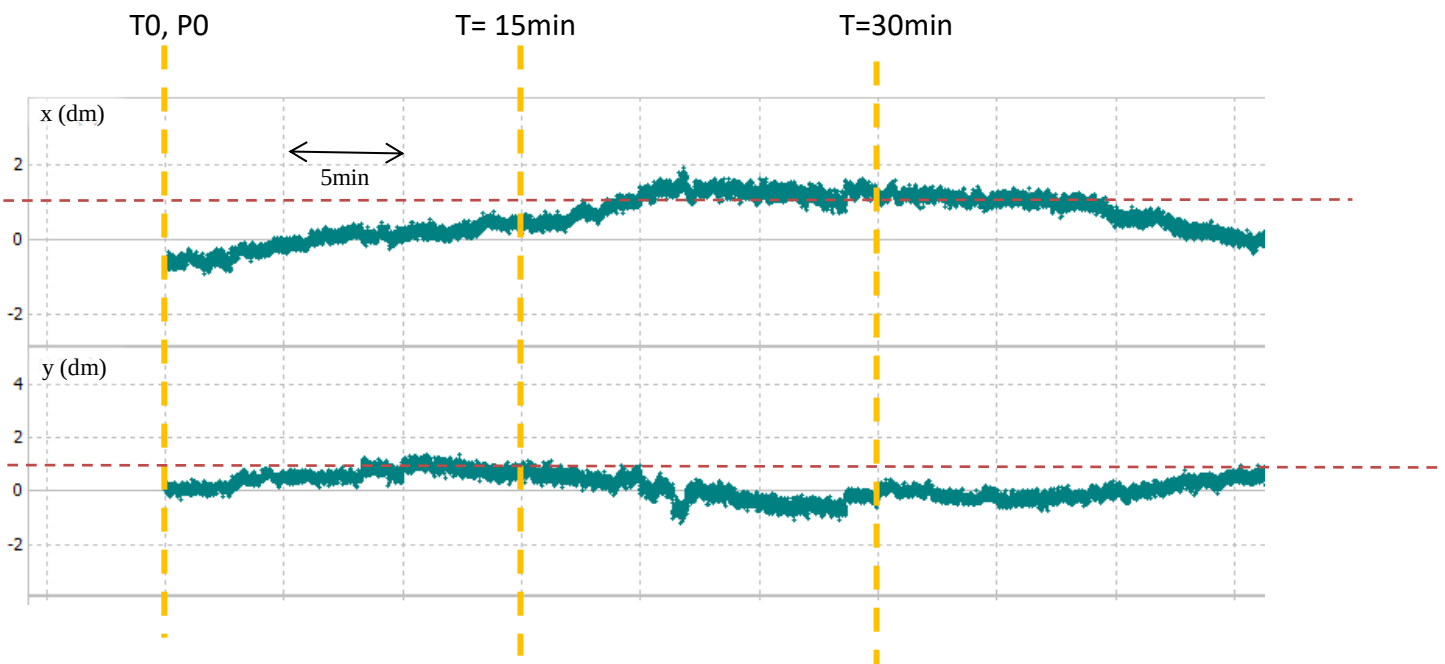


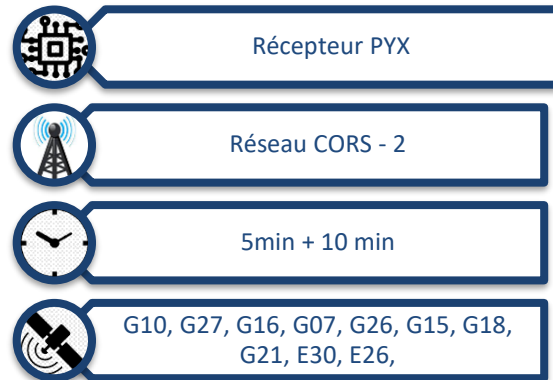
Figure 45- Convergence PPP-NRTK dans le scénario B

Dans ce scénario, on voit que le temps de convergence est inférieur aussi à 15min, on atteint les 10cm de précision sur l'axe y assez rapidement. La résolution reste plus longue sur l'axe X autour de 5min. Toutefois, sur les 30min d'observations et de calculs la solution PPP-NRTK reste stable. En comparaison, avec ce qui est fait en PPP-RTK (Teunissen, et al., 2010) seulement on a un temps de convergence similaire mais une précision meilleure. On peut aussi comparer avec le scénario

précédent, où nous voyons que nous avons une précision meilleure et un temps de convergence plus rapide.

Scénario C1

Dans le premier scénario mobile, Scénario C1, l'utilisateur est mobile et reçoit des corrections PPP-RTK toutes les 10 secondes (pour les biais de phase et horloges) et 1 minute pour les biais de code.



Le scénario se déroule en deux étapes : d'abord 5min en mobile après l'initialisation NRTK et l'utilisateur sera ensuite fixe pendant 10min mais pas au même endroit où l'initialisation NRTK a eu lieu.

La figure 46 représente le temps en fonction des écarts avec la position attendue. Le NRTK sert de référence pour l'erreur sur la position, avec toutes les limites que cela comporte. On peut ne pas être à l'abri, pendant le mouvement, d'une erreur telle qu'on l'a observée figure 43 à l'initialisation. Pour les scénarii mobiles, j'ai fait parallèlement les mêmes tests en NRTK pour avoir un aperçu des résultats attendus. Ces deux phases de tests sont synchronisées dans le temps et sont donc facilement comparables. Ainsi, cette courbe représente la différence en mètres entre la position calculée en PPP-RTK suite à l'initialisation et la position attendue calculée en NRTK. Dans ce cas, j'ai lissé la courbe pour voir plus facilement la tendance. Il faut prendre en compte une incertitude sur la position calculée en NRTK de 1 cm.

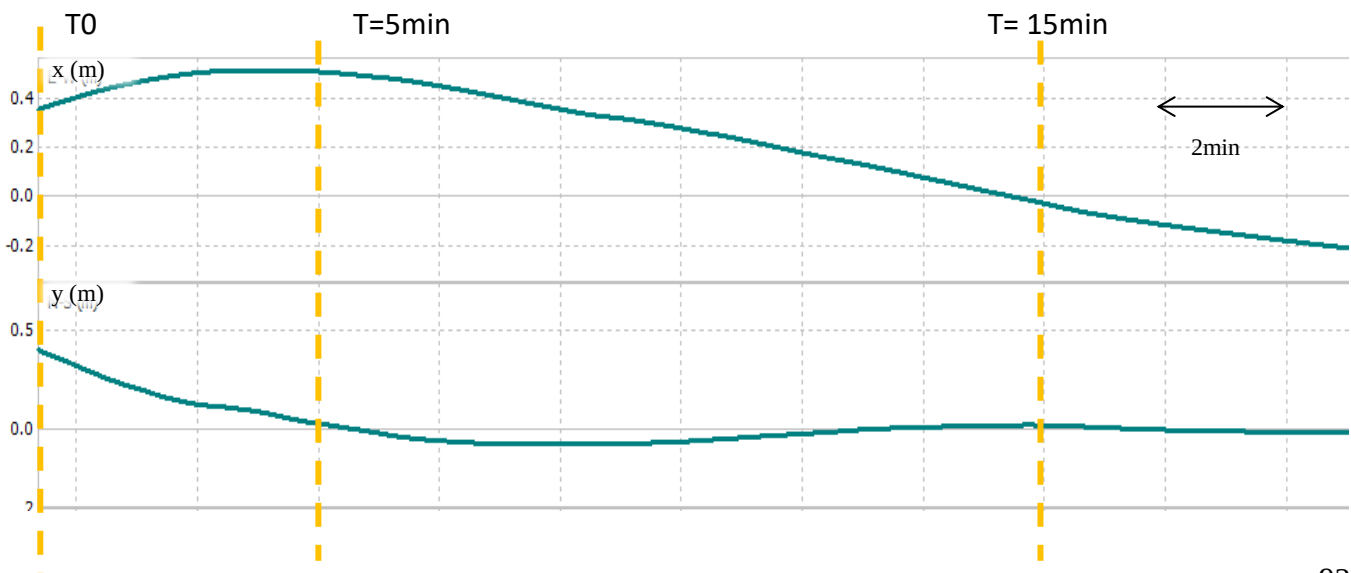


Figure 46- Convergence PPP-NRTK dans le scénario C1

Dans ce cas, on voit que la solution a du mal à converger dans les 5 premières minutes. En effet, un utilisateur mobile rend la résolution des ambiguïtés plus difficile, ce qui aura un impact sur le calcul de la position. En effet, quand l'utilisateur est mobile, on calcule sa position toutes les 10 secondes et on ré-effectue la détermination des ambiguïtés à l'aide de la première ambiguïté calculée en NRTK en P0. On a un écart entre 20 et 40cm dans la partie mobile. Dès que l'utilisateur s'arrête de bouger on arrive facilement à converger et à voir un écart de moins de 10 cm assez rapidement.

Scenario C2

Dans le Scénario C2, l'utilisateur est mobile et reçoit des corrections PPP-RTK2 toutes les 10 secondes (pour les biais de phase et horloges) et 1 minute pour les biais de code.

Le scénario se déroule en deux étapes : d'abord 10 minutes en mobile après l'initialisation NRTK et l'utilisateur sera ensuite fixe mais pas au même endroit où l'initialisation NRTK a eu lieu.

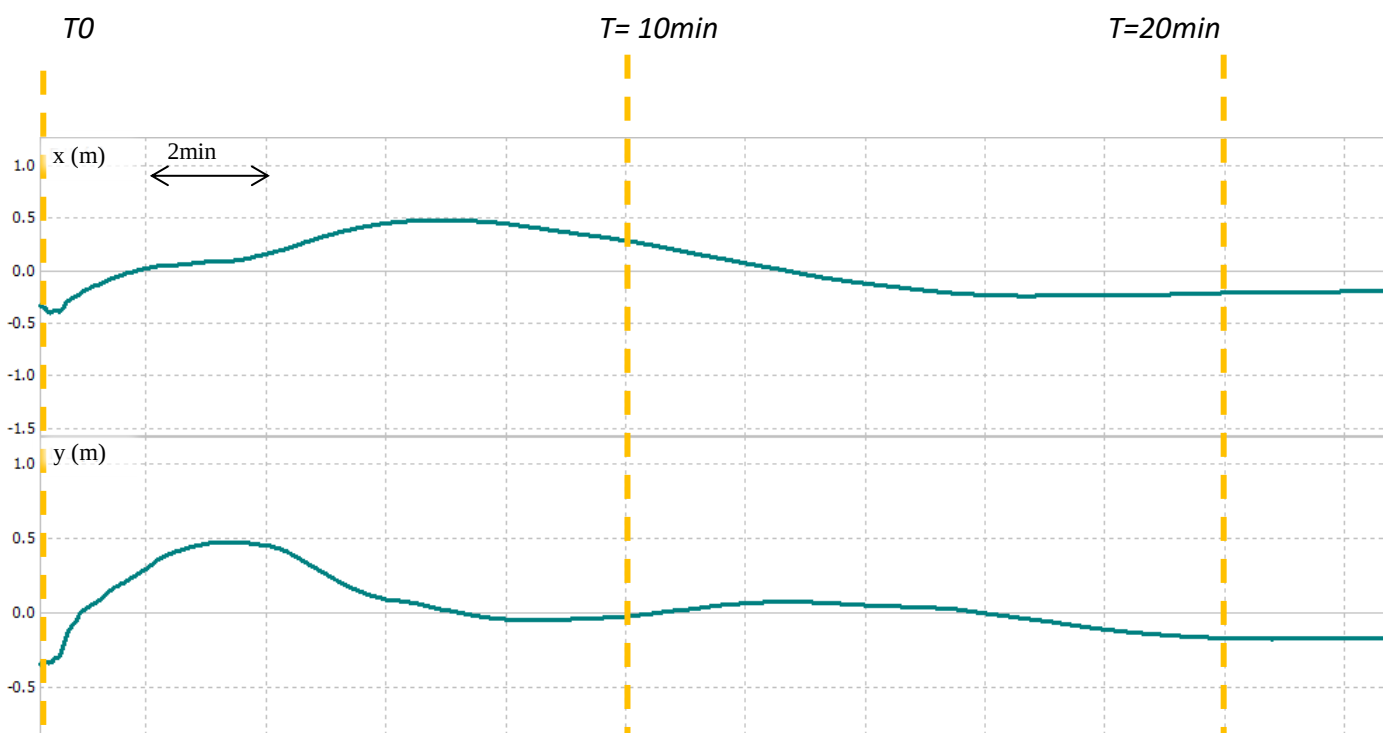


Figure 47 - Convergence PPP-NRTK dans le scénario C2

Cette fois ci l'utilisateur se déplace en cercle. Durant la première phase de 10min, l'utilisateur est mobile : on observe des gros écarts de précision, la résolution des ambiguïtés est difficile. Ainsi les erreurs de la position sont importantes atteignant les 50cm. Mais on observe un début de convergence des ambiguïtés au bout de 10min. Ainsi, même en mouvement l'utilisateur arrive à déterminer ses ambiguïtés en 10 minutes.

Toutefois, dès que l'utilisateur s'arrête de bouger, la détermination des ambiguïtés se fait plus facilement. Ainsi en moins de 5 minutes on arrive à avoir une précision inférieure à 10 cm, ce qui nous ramène logiquement au premier scénario.

On peut remarquer globalement ici que la nature du mouvement ne semble pas influencer sur le temps de convergence. En revanche, le fait d'être statique ou non, a bien une influence.

- **Précision PPP-NRTK**

Maintenant nous allons nous intéresser à la précision de calcul soit l'écart avec la position réelle de l'utilisateur. Nous avons validé les différents temps de convergence et nous prenons la précision au bout des 15 minutes de convergence nécessaires. Nous allons regarder de plus près les écarts de position pour définir la précision que l'on peut attendre de notre technique PPP-NRTK. En mode statique, les comparaisons seront aisées car on connaît les positions. En mode dynamique, on utilisera comme on l'a vu précédemment le NRTK comme référence.

Scénario A

Nous partons du principe que l'initialisation NRTK a déjà eu lieu, nous regardons donc les erreurs de position suite à cela sans prendre en compte le temps de convergence.

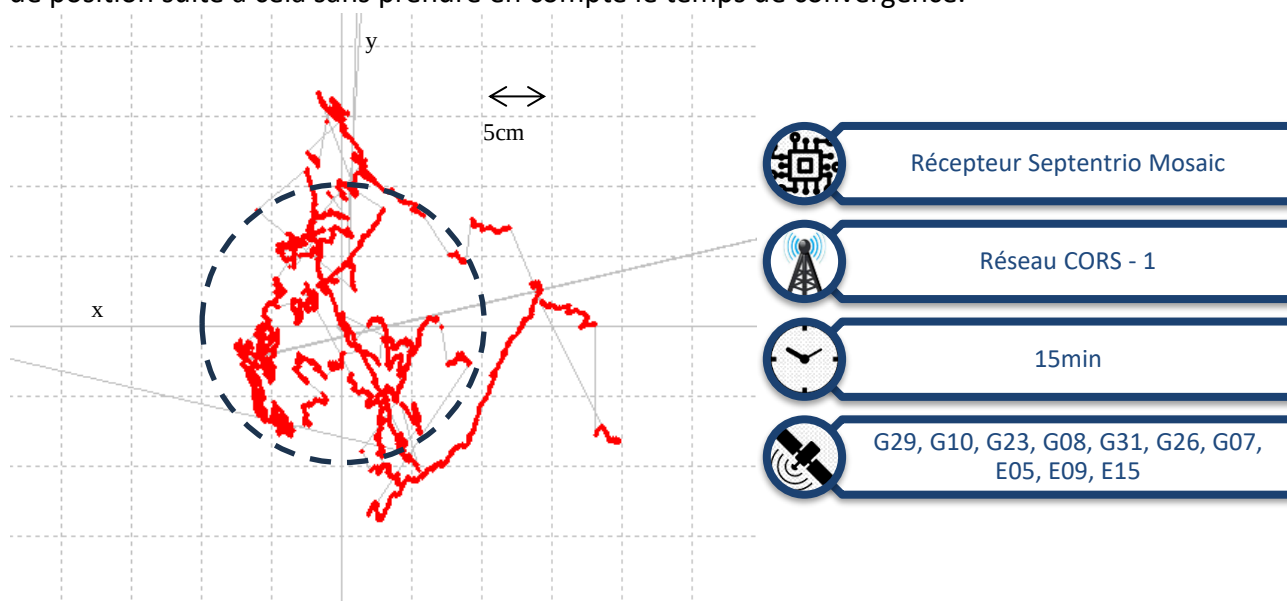


Figure 48- Ecart de la position autour de l'utilisateur dans le scénario A

Dans ce premier scénario, on observe sur la figure 46 que la plupart de nos points se trouvent dans un écart type de 20cm en x et 15cm en y. C'est une précision que l'on peut considérer comme acceptable pour une technique PPP-RTK. Toutefois, on peut noter que dans le carré de 10cm sur x et sur y, on trouve l'essentiel des points centrés autour de notre position.

Scenario B

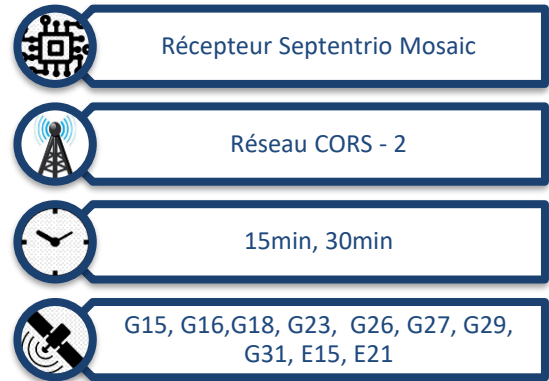
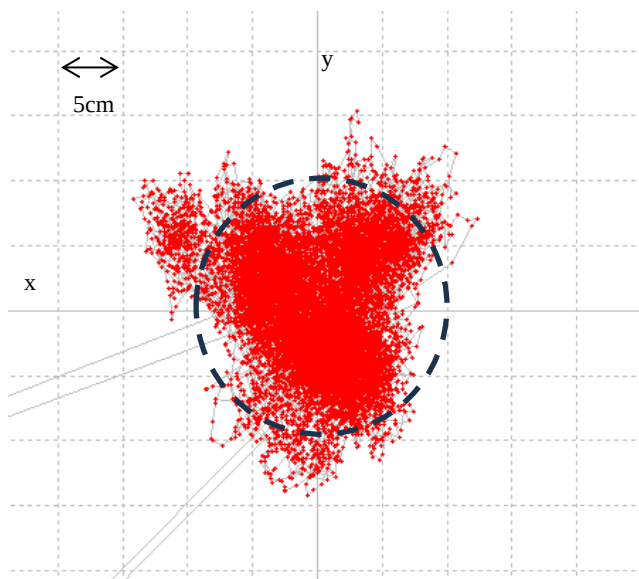


Figure 49- Ecart de la position autour de l'utilisateur dans le scenario B

Dans ce deuxième scénario, nous observons plus de points qui sont mieux centrés. D'abord, nous avons plus de points car ce deuxième test de la solution PPP-NRTK a un temps d'observation plus long par rapport au premier scénario : 30min (B) contre 15min (A). De plus, on observe que les points sont beaucoup plus concentrés dans notre zone de précision de 10cm. On peut y voir un premier élément de comparaison. Un temps d'observation plus long aiderait à une meilleure précision. Toutefois, ce n'est pas la seule explication. Nous discuterons de l'impact du choix du réseau CORS et des résultats dans la partie 4.3.

Scénario C1

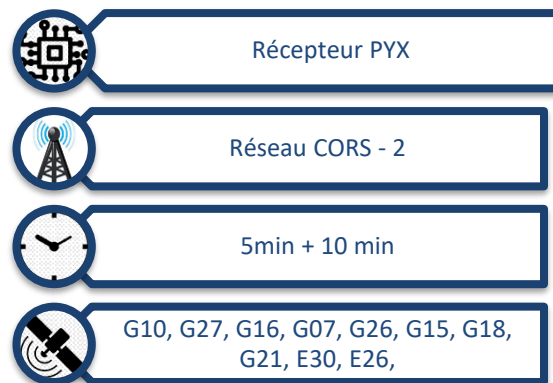
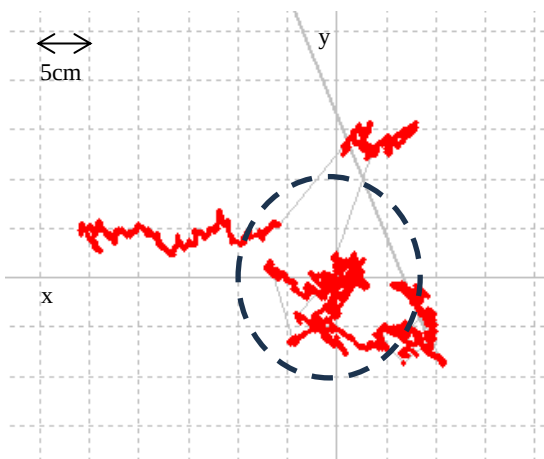


Figure 50 - Ecart de la position autour de l'utilisateur dans le scenario C1

Dans ce troisième scénario l'utilisateur se déplace en mouvement rectiligne. Le temps d'observation étant court nous n'avons pas beaucoup de points. D'abord, on voit que la plupart des points sont

centrés dans un écart type de 10cm. Par contre, un comportement singulier est visible sur la gauche. Cet écart est un décalage d'erreur de la position sur l'axe x. Cela peut s'expliquer par la difficulté à converger quand l'utilisateur bouge en lien avec la position des satellites par rapport à la coordonnée y. On aura une position calculée avec une erreur ayant un décalage de plus de 25cm.

Scénario C2

Rappelons que ce scénario se déroule en deux phases : une première où l'utilisateur va être mobile pendant 10 minutes, puis une autre où l'utilisateur s'arrêtera de bouger pendant les 10 minutes suivantes.

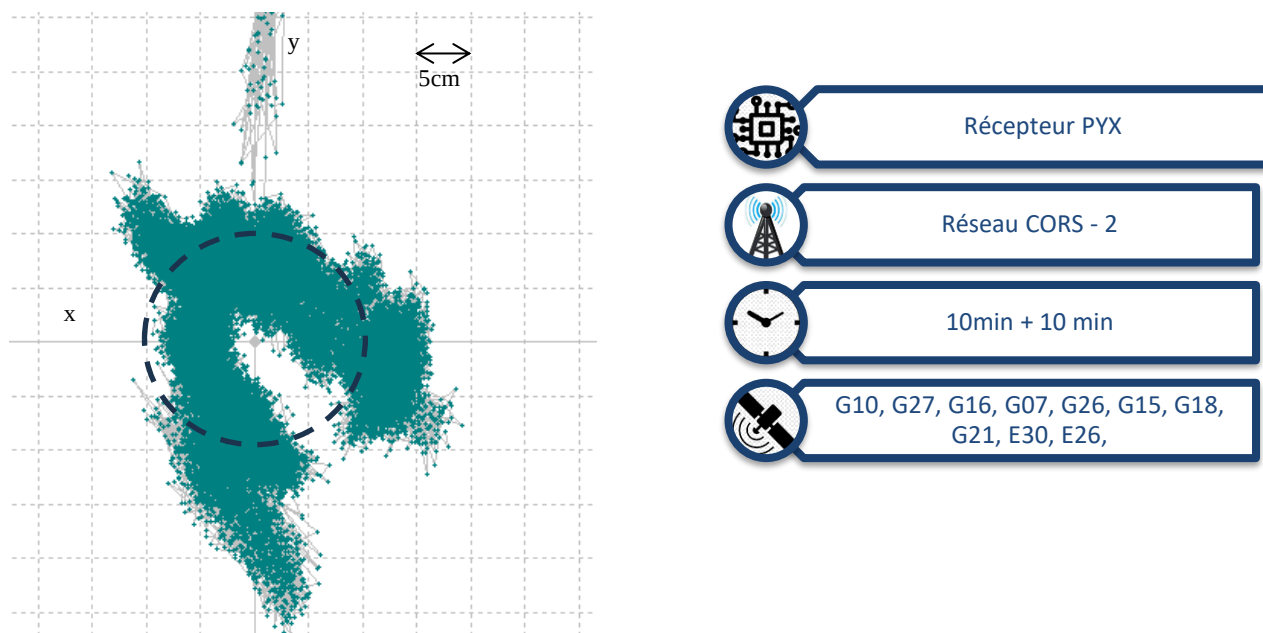


Figure 51 -Ecart de la position autour de l'utilisateur dans le scénario C2

En ce qui concerne les écarts de positions dans cette situation, deux choses importantes sont à mettre en évidence. D'une part l'écart d'erreur de la position est de 15cm selon x et 30cm selon y (on a mis le cercle de référence à 10cm). On voit que l'erreur est moins importante selon l'axe x. D'autre part, on observe sur le haut de la figure un amas de points séparés du reste. Cela peut s'expliquer par la réinitialisation et la re-détermination des ambiguïtés toutes les 10 secondes. On retrouve sur y le comportement qu'on avait observé sur x lors du scénario C1. Cette section de résultats consacrée à la précision nous permet d'affiner ce qu'on a vu sur la convergence. Il semble que le fonctionnement statique reste de meilleure qualité et qu'en dynamique, le mouvement ait tendance à réduire la précision en entraînant parfois des déviations inattendues causées par la difficulté de maintenir le calcul des ambiguïtés sans l'assistance du NRTK comme à l'instant initial.

- **Continuité**

Enfin, un dernier critère de performance à étudier pour notre solution hybride est la continuité. En effet, le principe de notre technique d'hybridation repose sur la confiance dans la continuité. La continuité entre ces deux techniques pour voir le comportement de la précision est ce que nous avons vu à la section précédente. Mais la continuité sur le long terme est aussi intéressante. Nous allons allonger le temps d'observation à 1 heure dans le scénario. Nous alternerons des passages fixes et mobiles toutes les 10 minutes. Nous allons pouvoir voir comment l'approche se comporte sur le long terme. De plus, il est important de noter que nous garderons la valeur NRTK initiale du début du scénario à T0 sans remise à jour pendant les 60 minutes qui suivent.

Scénario C1bis

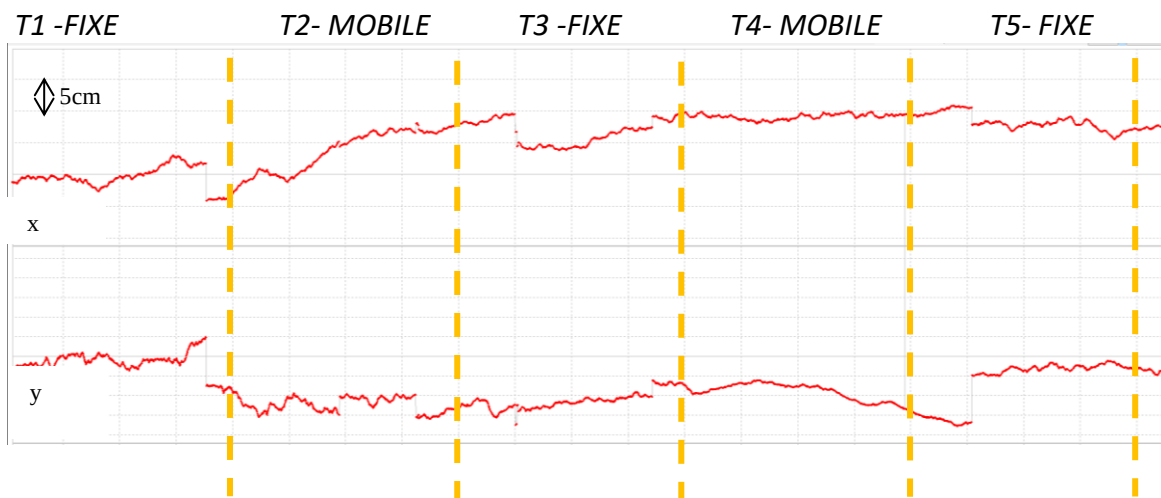


Figure 52- Continuité PPP-NRTK durant 55min

Dans ce scénario, toutes les 10 minutes l'utilisateur alterne entre mouvement et position statique. Première observation : on distingue des sauts aux transitions, parfois avec un décalage temporel mais on les voit systématiquement.

La principale difficulté rencontrée dans ce scénario est de fixer les ambiguïtés quand l'utilisateur bouge. En effet, toutes les 10 secondes l'utilisateur recalculé les ambiguïtés et sa position. Dans la figure 52, on voit des sauts de positions fréquents qui peuvent s'expliquer par ce recalcul des ambiguïtés. Toutefois, la précision reste stable sur le long terme (elle varie entre 10 et 20 cm) sans trop de déviation comme dans les scénarii C. A nos yeux ces résultats justifient l'approche que nous avons choisie. En effet, il semble que la difficulté majeure du PPP soit bien liée à la détermination des ambiguïtés entières. Le NRTK permet de lever cette difficulté initiale, on peut ensuite assurer le positionnement avec le PPP seul, même lorsque les ambiguïtés sont recalculées, sans pertes de performances et de stabilités significatives.

4.2.3 Impact du réseau CORS sur le calcul de la position

Nous allons maintenant analyser les différents critères de performance présentés ci-dessus pour les comparer en fonction du système de correction du réseau dont il est issu.

Comme présenté au début de cette partie nous avons deux réseaux CORS que nous avons utilisés dont un a une densité plus faible, ce qui revient à dire que les stations sont plus proches dans le CORS-2 que dans le CORS-1.

Table 26- Configuration des 2 réseaux CORS utilisés

	CORS-1	CORS-2
Constellations utilisées	GPS Galileo	GPS Galileo
Nombres de stations	5	5
Nombres de satellites utilisés	10	10
Distance moyenne inter-stations	65 km	34 km

Tout d'abord, analysons les temps de convergence entre les différents scénarios A et B en ayant une durée d'observation de 30min dans les deux cas. Nous prenons le TTFF quand nous avons fixé les ambiguïtés et que le taux de réussite est supérieur à 90%. De plus, nous répétons ces scénarios plusieurs fois, en respectant les mêmes conditions (mêmes nombres de satellites utilisés, même durée d'observation). Nous avons répété chaque scénario trois fois : A1, A2, A3, B1, B2 et B3. L'objectif est d'étudier les deux principaux critères de performance pour comprendre l'impact du réseau sur les performances.

Table 27- Comparaison des performances PPP-NRTK

	Scénario A1	Scénario A2	Scénario A3	Scénario B1	Scénario B2	Scénario B3
Réseau CORS	CORS-1	CORS-1	CORS-1	CORS-2	CORS-2	CORS-2
TTFF après init NRTK (P0)	16min	19min	18min	10min	12min	11min
Taux de réussite AR moyen	94%	92%	93%	98%	95%	97%
Précision moyen après TTFF, sur x et y	x<8cm y<10cm	x<7cm y<9cm	x<6cm y<10cm	x<5cm y<5cm	x<7cm y<5cm	x<5cm y<6cm
Ecart Type	x 10cm y 11 cm	X 7 cm Y 10cm	X 7 cm Y 10cm	X 5cm Y 5cm	X 7cm Y 10cm	X 6cm Y 7 cm

D'abord, nous pouvons comparer les différents temps de convergence. On voit que dans le réseau CORS-2, les temps pour déterminer les ambiguïtés sont plus rapides. Cela est effet relié aux taux de réussite de résolutions des ambiguïtés. Nous pouvons dire que c'est significativement plus rapide, car on a presque divisé par deux ce temps dans les scénarios utilisant les corrections PPP-RTK du réseau CORS-2.

De plus, en parallèle, nous voyons que plus le temps de convergence est rapide plus le taux de réussite AR sera élevé. On avait déjà observé cette corrélation dans l'état de l'art de la partie 1 et de la partie 2. En effet, dans n'importe quelle solution de positionnement GNSS, la détermination des

ambiguïtés sera le principal défi auquel les algorithmes devront répondre. Toutefois, on observe une amélioration quand le réseau CORS est plus dense. Ce n'est pas toujours le cas en PPP-RTK. On peut expliquer cela avec notre initialisation NRTK. En effet, la première étape de notre méthode hybride est un premier calcul de la position en NRTK. On détermine ainsi une première fois des ambiguïtés doubles différenciées entre l'utilisateur et la station virtuelle (VRS). Quand nous passons à la résolution PPP-RTK, nous utilisons ces ambiguïtés pour accélérer la résolution des ambiguïtés doubles différenciées. Cette fois, les ambiguïtés impliquent une station du réseau et un satellite qui sont fixées par les corrections et l'utilisateur. Ce ne sont pas les mêmes ambiguïtés toutefois, au vu de la proximité entre la VRS et une station du réseau, déterminer les ambiguïtés va pouvoir se faire plus rapidement. En effet, le réseau CORS-2 est dense avec des distances inter-stations moyennes de 33km. Par contre, si on utilise les corrections PPP-RTK1, on aura des distances inter stations quasiment deux fois plus grandes. Ainsi, les ambiguïtés doubles différenciées entre une station et la VRS précédemment calculés seront ainsi très différents vu que la VRS située à 10 km de l'utilisateur sera plus éloignée de la station du réseau CORS-1 dont nous avons fixé les ambiguïtés. Le problème sera le même d'un point de vue résolution du réseau et de la génération des corrections. Il sera plus difficile d'estimer les biais et les erreurs d'horloge dans le réseau CORS-1 que dans le réseau CORS-2.

Maintenant, nous allons regarder de plus près le meilleur scénario présent dans le tableau : le scénario B1. Dans ce dernier, le taux de résolution AR est de 98% avec un temps de convergence inférieur à 10 minutes. Si nous comparons avec les techniques PPP, présentées dans la partie 2, et PPP-RTK, présentées dans la partie 3, nous voyons qu'avec notre technique hybride, nous obtenons un temps de convergence plus rapide.

Dans la figure 53 ci-dessous, on observe la précision après l'initialisation en NRTK et le temps de convergence de 10 minutes.

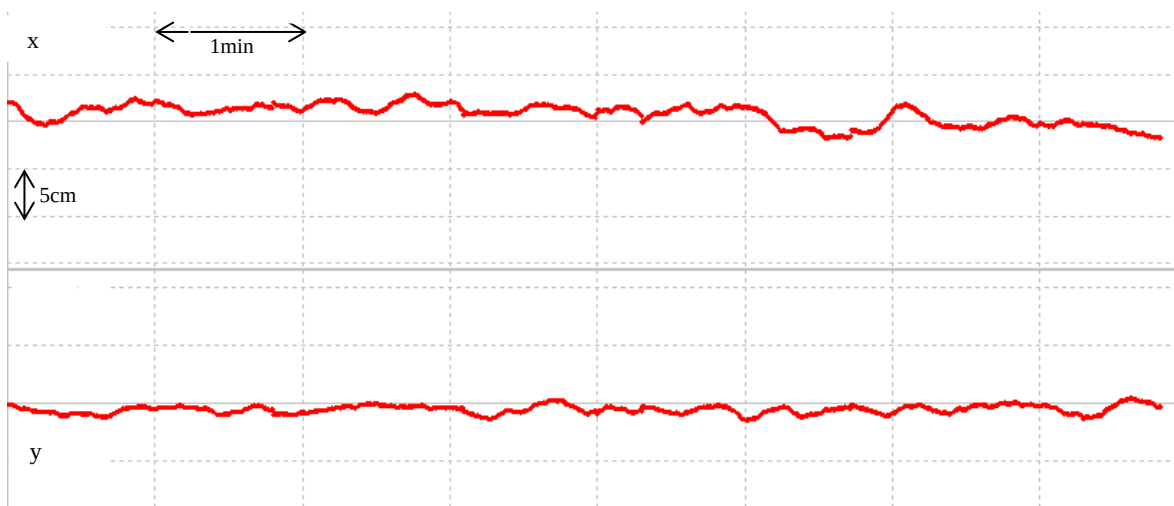


Figure 53- Précision PPP-NRTK avec initialisation NRTK et une convergence de 10min

Cette figure montre que le PPP-NRTK mis en œuvre peut atteindre des performances proches de celles du NRTK dans des conditions réseaux qu'on peut qualifier d'optimales. Toutefois, le temps de convergence reste de 10 minutes pour atteindre une précision proche du centimètre. Selon les utilisations, dont nous allons brièvement discuter ensuite, si une précision de moins de 10 centimètres est requise en moins de 5 minutes, alors l'utilisation de la technique PPP-NRTK proposée peut raisonnablement être envisagée.

4.3 Applications de la technique – Mise en œuvre industrielle possible

Cette thèse a été menée dans le cadre d'une thèse CIFRE en partenariat avec la société Exagone. La société Exagone propose un service de positionnement GNSS appelée réseau Teria. Ce réseau Teria se compose de stations couvrant l'ensemble de la France comme présentée dans la première partie. Le réseau Teria fournit un service NRTK dont les principaux utilisateurs sont les géomètres experts.

Lorsqu'un géomètre est sur le terrain, il cherche à mesurer des points et à déterminer leurs positions de manière précise (< 5cm). L'utilisateur se connecte au réseau Teria via les données mobiles (3G, 4G, 5G) et va prendre ses mesures à l'aide de son récepteur GNSS. Comme c'est un service NRTK, l'utilisateur va en se connectant au réseau Teria fournir une première approximation de sa position. En réponse le réseau va fournir les corrections NRTK appropriées comme les données de la VRS. Ensuite, le géomètre continue son relevé de points tout en étant connecté au réseau Teria. S'il perd sa connexion 4G, alors l'utilisateur devra reprendre le calcul depuis le début pour relancer le processus de réception des corrections NRTK avec la bonne VRS. Ces coupures arrivent plus souvent dans les zones où le réseau mobile est moins stable comme dans les endroits autour des zones blanches du territoire français.

La solution hybride PPP-NRTK qui a été présentée dans ce document peut facilement s'intégrer dans cet écosystème. En effet, le géomètre n'aurait besoin de se connecter qu'une seule fois au réseau Teria. Ainsi, il initialiserait sa position grâce au NRTK puis il pourrait basculer sur les corrections PPP-RTK. Ces dernières n'ont pas besoin de communication une fois l'initialisation effectuée. En effet, grâce à l'approche SSR, il suffit de diffuser les corrections PPP-RTK en L-band par exemple et le géomètre pourra ainsi mesurer en continue sa position sans avoir besoin des corrections NRTK.

On pourrait également imaginer étendre l'utilisation à d'autres applications nécessitant du positionnement décimétrique, mais sans les contraintes dans la durée du calcul NRTK.

Du point de vue du réseau, on peut également noter qu'en fiabilisant encore la solution PPP-NRTK à partir de notre proposition, on pourrait économiser de la bande passante de communication, et donc l'encombrement du réseau, puisque le calcul NRTK, le plus lourd dans ce domaine, n'est requis qu'épisodiquement.

4.4 Conférences, mise en valeur des travaux dans le monde de la recherche

Durant mes trois années de thèses, j'ai pu participer à au moins trois conférences, dont deux avec actes pour présenter les différents travaux menés et qui m'ont beaucoup aidé pour comprendre les enjeux du positionnement GNSS.

Isambert, D., Chambon, P., & Vervisch, A. (2021). PPP-RTK : the advantageous result of a hybridization of GNSS accurate positioning techniques. In A. Basiri, G. Gartner, & H. Huang (Eds.), LBS 2021: Proceedings of the 16th International Conference on Location Based Services (pp. 63–68).

Isambert, Delphine, Chambon, Paul, Vervisch-Picois, Alexandre, "A Hybrid Solution PPPRTK, a Major Asset for the CORS Network," Proceedings of the 35th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2022), Denver, Colorado, September 2022, pp. 1271-1279.

Isambert, D. "Precise Positioning for Mass-Market: Optimal Data Dissemination DAB+ Demonstrator", in Proceedings of the European Navigation Conference 2023, Noordwijk, Zuid-Holland, 31 May–2 June 2023, MDPI: Basel, Switzerland, doi:10.3390/ENC2023-15446

Je précise également qu'un papier destiné à une publication revue est en cours de finalisation.

Conclusion

Nous allons dans cette conclusion reprendre les éléments principaux de notre manuscrit. D'abord, le positionnement précis GNSS est de plus en plus présent dans l'écosystème de la géolocalisation. Dans les deux premières parties, nous avons présenté les différentes méthodes de positionnement GNSS précis qui existent dans la littérature : RTK, NRTK, PPP, PPP-RTK et les différentes variantes de ces techniques. L'ensemble de celles-ci repose sur la résolution des équations d'observations GNSS. Nous pouvons résumer la façon dont la résolution se fait de deux manières :

- Combinaisons et différenciations des équations d'observations GNSS : c'est une solution possible pour faciliter la résolution. C'est ce qui est fait essentiellement en RTK et NRTK. En effet, cela pourra se faire entre différentes stations de référence et l'utilisateur, entre différents satellites ou entre différentes fréquences.
- Sources de données externes : l'algorithme de positionnement GNSS utilise des données externes pour accélérer la résolution des équations. Cela peut être des données disponibles au niveau mondial comme l'IGS sur les orbites précises par exemple. Mais il sera aussi possible d'utiliser un réseau de stations qui pourra fournir des données supplémentaires à l'utilisateur. Cette deuxième façon correspond plus au PPP et au PPP-RTK.

Toutefois, dans l'ensemble des techniques présentées, nous trouvons en fait un peu de ces deux façons de faire pour répondre aux mieux aux critères de performances requis. Nous pouvons notamment estimer certains paramètres des équations d'observations GNSS plus rapidement. En effet, nous avons identifié trois principaux critères de performance qui nous ont servis de ligne directrice pour la comparaison des solutions :

- La précision.
- Le temps de convergence ou TTFF.
- Le taux de réussite de résolution des ambiguïtés.

Suite aux différents résultats que nous avons trouvés dans l'état de l'art, nous avons défini nos objectifs de performances pour notre future solution hybride : un temps inférieur à 10 minutes et une précision inférieure à 10 centimètres.

Ainsi, j'ai mis en œuvre une nouvelle façon d'hybrider le NRTK avec du PPP. L'idée est de combiner un algorithme PPP-RTK et NRTK pour avoir une position calculée en continue même en cas de coupure des données fournies à l'utilisateur. L'objectif est de garder une position précise même avec une méthode de calcul globale comme le PPP ou le PPP-RTK. Prenant conscience que l'essentiel de ce qui limite le temps de convergence du PPP est la détermination des ambiguïtés (notamment de les rendre entières pour pouvoir les résoudre avec des méthodes efficaces), j'en ai déduit que la meilleure manière d'associer PPP et RTK serait de s'appuyer sur la force du second pour réduire la faiblesse principale du premier. Mon idée consiste donc en une initialisation du calcul en NRTK, puis

de continuer de calculer la position en PPP, avec les données de corrections calculées par le réseau NRTK. On appellera cette technique hybride PPP-NRTK que résume la figure 54.

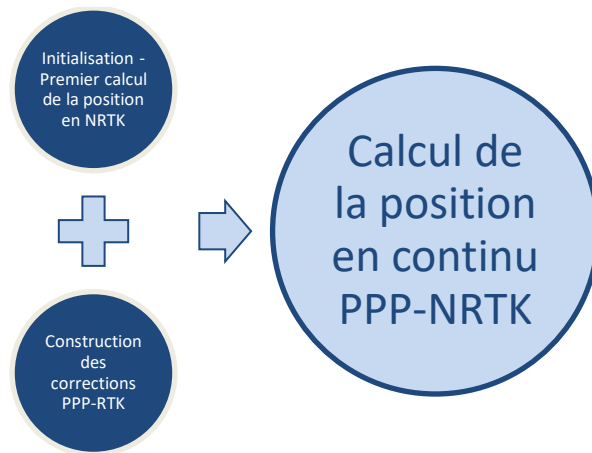


Figure 54- Technique PPP-NRTK

De plus, d'un point de vue plus mathématique, mon approche permet d'avoir des informations supplémentaires : une ambiguïté double différenciée initiale sur P0 ainsi que la position initiale calculée en NRTK. Ainsi, cet apport d'informations permet à la fois d'accélérer la résolution des ambiguïtés mais aussi d'avoir un critère de validation. En effet, nous connaissons la position initiale et nous l'utilisons comme référence pour détecter par exemple la présence de multi trajets ou de sauts de cycles ou autres contraintes extérieures quand l'écart entre la nouvelle position calculée et l'initiale est trop grand. On peut fixer ce paramètre de différentes manières dans le cas d'un scénario où l'utilisateur est mobile ou fixe.

En termes de performances de la technique PPP-NRTK, nous pouvons nous retrouver dans les standards PPP-RTK : un temps de convergence inférieur à 20 minutes menant à une précision de moins de 10 centimètres. Toutefois, on observe des variations dans les résultats qui dépendent de plusieurs facteurs. Cela est notamment dû au type de réseau CORS utilisé. En effet, dans notre phase d'expérimentation nous avons utilisé deux réseaux dont l'un est plus dense que l'autre. Nous avons observé, sans trop de surprise, qu'avec un réseau plus dense notre technique hybride aura de meilleurs résultats s'approchant des performances en NRTK seul. Cela est dû au fait que notre initialisation en NRTK aura un impact plus important mais aussi que les différentes corrections (biais de phase et d'horloge satellites) sont de meilleures qualités dans ce type de réseau plus dense.

Toutefois, cela peut devenir une limite d'application de notre méthode hybride. En effet, aujourd'hui le réseau Teria a une distance moyenne inter stations de 70 km. Pour avoir les mêmes résultats qu'en NRTK il faudrait densifier ce réseau pour arriver à une distance moyenne inter stations d'au moins 40 km. Nous avons mené plusieurs études sur l'impact de la taille du réseau durant cette thèse, cela a fait l'objet d'une publication dans une conférence en 2022, mais aucune ne permet de conclure à un réel critère pouvant trancher sur la distance optimale entre deux stations.

Il faudra s'intéresser au cadre d'application de notre méthode : est-ce que l'objectif est de remplacer le NRTK ou alors de multiplier les méthodes ? La multiplication de méthodes permettrait à de nouvelles applications d'implémenter la fonction de géolocalisation précise. La précision dans ce cas autour de 10 cm pourrait suffire, mais l'intérêt serait l'approche SSR par la diffusion de correction

sur des plus larges zones. De plus, l'intérêt dans ces conditions serait de prendre temporairement le relai pour décharger le réseau. Le NRTK consomme de la ressource réseau. On pourrait donc envisager aussi cette solution comme un moyen temporaire de soulager la surcharge. En ajoutant à cela les possibles coupures de réseau, cette technique pourrait prendre le relai dans plusieurs cas de figures, apportant de la redondance au calcul de position NRTK.

De plus, une autre limite est à soulever : la précision des corrections GNSS. En effet, l'ensemble de la résolution dans les méthodes de positionnement GNSS repose sur la réception des corrections, via des messages permettant la résolution de la position d'un point de vue utilisateur. En effet, aujourd'hui peu de techniques permettent de quantifier la précision et la validité des corrections que l'on diffuse que cela soit en NRTK ou en PPP-RTK. Ainsi une nouvelle approche de notre méthode hybride serait de garantir une précision de corrections avec par exemple la mise en place d'algorithmes de validation et de vérification sur chaque paramètre en sortie de la partie réseau. Cela rajouterait une étape supplémentaire dans la génération de nos corrections PPP-RTK. Des algorithmes d'intégrité commencent à apparaître pour certifier la validité des corrections. Nous pourrions approfondir cette approche pour l'intégrer à notre solution PPP-NRTK et ainsi avoir un nouveau critère pour basculer d'une technique NRTK à PPP-NRTK.

Annexe 1 -Bibliographie

- Abba Ibrahim [et al.]** Ionospheric Effects on GPS Signal in Low Latitude Region: A Case Study Review of South East Asia and Africa [Revue]. - [s.l.] : Nigerian Journal of Technology, 2015. - 3 : Vol. 34.
- Aggrey John et Bisnath Sunil**, Analysis and modelling of pseudorange and carrier phase biases in GNSS precise point positioning [Revue] // Proceedings of the 27th International Technical Meeting of the Satellite Division of The Institute of Navigation. - 2014. - pp. 2512-2522.
- Alkan Reha Metin, Ozulu Ibrahim Murat et İlçi Veli** Precise Point Positioning (PPP) Technique versus Network RTK GNSS [Revue] // FIG Working Week. - 2016.
- Bakula Mieczysław, Oszczak Stanislaw et Renata Pelc Mieczkowska** Performance of RTK Positioning in Forest Conditions: Case Study [Revue]. - [s.l.] : Journal of Surveying Engineering, 2009. - pp. 125-130.
- Baybura Tamer [et al.]** Examining the Accuracy of Network RTK and Long Base RTK Methods with Repetitive Measurements [Revue] // Sensors. - 2019. - pp. 1-12.
- Dabove Paolo [et al.]** Network Real Time Kinematic (NRTK) Positioning Description, Architectures and Performances [Revue] // Satellite Positioning - Methods, Models and Applications . - 2015. - pp. 23-46.
- Doucet K. & Herwig, M. [et al.]** Introducing ambiguity resolution in web hosted global multi GNSS precise positioning with trimble RTX PP [Revue] // Proceedings of the 25th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2012). - 2012. - pp. 1115-1125.
- Duong Viet & Harima, Ken & Choy, Suelynn & Rizos, Chris** Performance of Precise Point Positioning using current multi frequency and multi GNSS measurements in Australia [Revue] // Proceedings of International Global Navigation Satellite Systems. - [s.l.] : Proceedings of International Global Navigation Satellite Systems, 2016. - pp. 1-15.
- Ekaso Desta, Nex Francesco et Kerle Norman** Accuracy assessment of real time kinematics (RTK) measurements on unmanned aerial vehicles (UAV) for direct georeferencing [Revue] // Geospatial Information Science. - 2020. - pp. 165-181.
- Feng Yamming et Wang Jinling** GPS RTK Performance Characteristics and Analysis [Revue] // Journal of Global Positioning Systems. - 2008. - p. Vol 1 No 13.
- Feng Y. et Wang J.** Exploring GNSS RTK Performance Benefits with GPS and Virtual Galileo Measurements [Revue] // Proceedings of IONNTM207. - San Diego, CA : Proceedings of IONNTM2007, 2007. - Vol. p. 218-226. - pp. 218-226.
- Gao Wang [et al.]** Inter-System Differencing between GPS and BDS for Medium Baseline RTK Positioning [Revue]. - [s.l.] : Remote Sensing, 2017. - 948 : Vol. 9.
- Guo Yao [et al.]** Carrier Characteristic Bias Estimation between GNSS Signals and Its Calibration in High Precision joint Positioning [Revue] // Remote Sensing . - 2023. - 4 : Vol. 15.
- Khodabandeh Amir et Teunissen P.** An analytical study of PPP RTK corrections: precision, correlation and user impact [Revue] // Journal of Geodesy. - 2015. - Vol. 89. - pp. 1-24.
- Laurichesse D. et Privat A.,** "An Open source PPP Client Implementation for the CNES PPP WIZARD Demonstrator [Revue] // Proceedings of the 28th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2015). - Tampa : [s.n.], 2015.
- Leandro R. [et al.]** RTX positioning: The next generation of cm accurate real time GNSS positioning [Revue] // Proceedings of the 24th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2011). - 2011. - pp. 1460-1475.

Li Bofeng et Teunissen P. J. G. High Dimensional Integer Ambiguity Resolution: A First Comparison between LAMBDA and Bernese [Revue] // Journal of Navigation. - 2011. - pp. S192-S210.

Li Xingxing [et al.] Multi constellation GNSS PPP instantaneous ambiguity resolution with precise atmospheric corrections augmentation [Revue]. - [s.l.] : GPS Solutions, 2021. - 107 : Vol. 25.

Li Xingxing [et al.] Review of PPP RTK: achievements, challenges, and opportunities [Revue] // Satellite Navigation. - 2022. - 28 : Vol. 3.

Li Xingxing, Zhang Xiaohong et Ge Maorong Regional reference network augmented precise point positioning for instantaneous ambiguity resolution [Revue]. - [s.l.] : Journal of Geodesy, 2010. - Vol. 85. - pp. 151-158.

Ma Hongyang [et al.] Assessing the performance of multi GNSS PPP RTK in the local area [Revue] // Remote Sensing. - 2020. - 20 : Vol. 12.

Norin Dan [et al.] Practical Evaluation of RTCM Network RTK Messages in the SWEPOS™ Network [Revue] // Proceedings of the 2009 International Technical Meeting of The Institute of Navigation. - 2009. - pp. 353-359.

Odijk Dennis [et al.] On the estimability of parameters in undifferenced, uncombined GNSS network and PPP RTK user models by means of S-system theory [Revue] // Journal of Geodesy. - 2016. - 90. - pp. 15-44.

Odijk Dennis, Teunissen P. et Khodabandeh Amir Single frequency PPP RTK: Theory and experimental results [Revue] // Earth on the Edge: Science for a Sustainable Planet. - 2014. - pp. 571-578.

Psychas D. et Verhagen S. Real Time PPP RTK Performance analysis using ionospheric corrections from multi scale network configurations [Revue] // Sensors 2020. - 2020. - 11.

Retscher G. Accuracy Performance of Virtual Reference Station (VRS) Networks [Revue] // Journal of Global Positioning Systems. - [s.l.] : Journal of Global Positioning Systems, 2002. - 1 : Vol. 1. - pp. 40-47.

Ruwisch F., Jain A. et Schön S. Characterisation of GNSS Carrier Phase Data on a Moving Zero Baseline in Urban and Aerial Navigation [Revue] // Sensors. - 2020.

Teunissen P. J. G. et Khodabandeh Amir Review and principles of PPP RTK methods [Revue] // Journal of Geodesy. - 2015. - 89. - pp. 217-240.

Teunissen P. J. G., Odijk Dennis et Zhang Baocheng PPP RTK: Results of CORS network based PPP with integer ambiguity resolution [Revue] // Journal of Aeronautics, Astronautics and Aviation, Series A. - 2010. - pp. 223-230.

Vollath Ulrich [et al.] Multi Base RTK Positioning Using Virtual Reference Stations [Revue]. - [s.l.] : Proceedings of the 13th International Technical Meeting of the Satellite Division of The Institute of Navigation, 2000. - pp. 123-131.

Vollath Ulrich [et al.] Network RTK Versus Single Base RTK Understanding the Error Characteristics [Revue]. - [s.l.] : Proceedings of the 15th International Technical Meeting of the Satellite Division of The Institute of Navigation, 2002. - pp. 2774 - 2781.

Wang Jinling Stochastic modeling for real time kinematic GPS/GLONASS positioning [Revue]. - [s.l.] : NAVIGATION: Journal of the Institute of Navigation, 1999. - 4 : Vol. 46. - pp. 297 - 305.

Wanninger Lambert Ionospheric Disturbance Indices for RTK and Network RTK Positioning [Revue]. - [s.l.] : Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation, 2004. - pp. 2849 - 2854.

Wei Erhu [et al.] VRS Virtual Observations Generation Algorithm [Revue] // Journal of Global Positioning Systems. - 2006.

Wübbena Gerhard [et al.] Reducing distance dependent errors for real time precise DGPS applications by establishing reference station networks [Revue] // Proceedings of the 9th International Technical Meeting of the Satellite Division of The Institute of Navigation . - 1996. - pp. 1845 - 1852.

Wübbena Gerhard, Schmitz Martin et Bagge Andreas. PPP RTK: Precise Point Positioning Using State Space Representation in RTK Networks [Revue] // Proceedings of the 18th

International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2005). - 2005. - pp. 2584 - 2594.

Zhang Baocheng, Hou Pengyu et Odolinski Robert PPP RTK: from common-view to all-in-view GNSS networks [Revue] // Journal of Geodesy. - 102 : Vol. 96.

Zhang Baocheng, Yongchang Chen et Yuan Yunbin PPP RTK based on undifferenced and uncombined observations: theoretical and practical aspects [Revue] // Journal of Geodesy. - 2018. - Vol. 93. - pp. 1011–1024.

Zhang Xiaohong [et al.] Investigating GNSS PPP RTK with external ionospheric constraints [Revue] // Satellite Navigation. - 2022. - 6 : Vol. 3.

Annexe 2 -Table des figures

Figure 1 - Paramètres et critères de performance RTK.....	12
Figure 2- TTFF et la précision selon la distance utilisateur station de base (Feng , et al., 2008).....	13
Figure 3- Erreurs ionosphériques sur deux signaux GPS	14
Figure 4- TTFA en lien avec la distance base-utilisateur (Wübbena, et al., 1996)	15
Figure 5- Taux AR fixé en fonction de la distance utilisateur-stations de base	16
Figure 6 - Création de la station virtuelle VRS.....	20
Figure 7- Réseau CORS Teria.....	21
Figure 8- Paramètres et critères de performance NRTK.....	22
Figure 9- Module fonction PPP	23
Figure 10- Paramètres et critères de performance PPP.....	24
Figure 11- Représentation des erreurs sur le principe OSR avec les techniques NRTK	27
Figure 12-Diffusion de corrections de l'espace d'état (SSR) au fil du temps.....	30
Figure 13 - Principe fonctionnement PPP-RTK.....	31
Figure 14- Corrections des horloges dans la technique RTX (PPP-RTK) avec Ionosphere free (Leandro, et al., 2011).....	38
Figure 15 - Temps de convergence des erreurs de position horizontale à 10 cm en fonction de la densité du réseau pour FAR et PAR, sur la base de 50 % et 90 % des solutions de l'échantillon.....	39
Figure 16- Paramètres et critères de performances PPP-RTK	41
Figure 17- Erreurs de position statiques et absolues de 216 utilisateurs de PPP-RTK en fonction de l'époque d'observation (Zhang, et al., 2018)	43
Figure 18- Convergence des ambiguïtés calculées avec RTX (Doucet, et al., 2012).....	43
Figure 19- Résultats des positionnements PPP et PPP-RTK basés sur les données de phase et de code GPS mono-fréquence corrigées par le réseau CORS. (Odijk, et al., 2014).....	44
Figure 20- Erreurs sur position verticale dans une technique PPP-RTK (Teunissen, et al., 2010)....	45
Figure 21 -Précisions de la technique PPP-RTK en fonction du nombre d'époques et de stations dans le réseau CORS	46
Figure 22- Variation du nombre de satellites sur une longue durée d'observations	46
Figure 23- Erreurs de la position en fonction du nombre de satellites et du temps (Ma, et al., 2020)	47
Figure 24- Réseau CORS interconnecté avec k satellites en visibilité	49
Figure 25- Chaîne Logicielle Partie Réseau, génération des corrections PPP-RTK.....	61
Figure 26- Utilisateur dans un réseau CORS interconnectés avec k satellites en visibilité.....	62
Figure 27- Chaîne logicielle étape 2 avec l'initialisation NRTK	63
Figure 28- Utilisateur en visibilité k satellites et une VRS	66
Figure 29 - Chaîne logicielle complète de la technique PPP-NRTK proposée.....	67
Figure 30-Technique PPP-NRTK.....	69
Figure 31 – Réseau CORS-1 composé de 5 stations (en rouge)	71
Figure 32- Réseau CORS-2 composé de 5 stations	72
Figure 33-ZTD – Erreur troposphérique sur une partie de notre zone de test pour l'utilisation de corrections PPP-RTK (en mètres)	74
Figure 34- L'utilisateur connecté au réseau Teria	75
Figure 35 - Utilisateur dont la position est inconnue dans le réseau CORS-1.....	77
Figure 36- Utilisateur fixe dans le réseau CORS-2 - Scénario B.....	78
Figure 37- Scénario C1, utilisateur mobile	80

Figure 38- Scénario C2, utilisateur mobile	81
Figure 39- Paramètres et critères de performance PPP-NRTK.....	82
Figure 40- Biais de code différencié G29/G10 sur la fréquence L1, avec biais IF L1/L2 en mètres	84
Figure 41- Calcul de la position P0 en NRTK Scénario A.....	88
Figure 42- Calcul de la position P0 en NRTK Scénario B.....	88
Figure 43- Calcul de la position P0 en NRTK Scénarios C1 et C2	89
Figure 44- Convergence PPP-NRTK dans le scénario A	90
Figure 45- Convergence PPP-NRTK dans le scénario B	91
Figure 46- Convergence PPP-NRTK dans le scénario C1	93
Figure 47 - Convergence PPP-NRTK dans le scénario C2	93
Figure 48- Ecart de la position autour de l'utilisateur dans le scénario A.....	94
Figure 49- <i>Ecart de la position autour de l'utilisateur dans le scénario B</i>	95
Figure 50 - Ecart de la position autour de l'utilisateur dans le scénario C1	95
Figure 51 -Ecart de la position autour de l'utilisateur dans le scénario C2.....	96
Figure 52- Continuité PPP-NRTK durant 55min.....	97
Figure 53- Précision PPP-NRTK avec initialisation NRTK et une convergence de 10min.....	99
Figure 54- Technique PPP-NRTK.....	103

Annexe 3 - Glossaire

GNSS - Global Navigation Satellite Systems
IRNSS - Indian Regional Navigation Satellite System
QZSS - Quasi-Zenith Satellite System
GPS - Global Positioning System
RTK - Real Time Kinematic
PPP - Precise point positioning
NRTK - Network Real Time Kinematic
NTRIP - Networked Transport of RTCM via Internet Protocol
RTCM - Radio Technical Commission for Maritime Services
EGNOS - European Geostationary Navigation Overlay Service
TTFF - Time-To-First-Fix
AR - Ambiguity resolution
TTFA - Time To Fix Ambiguity
LAMBDA - Least-squares AMBIGUITY Decorrelation Adjustment
FDMA - Frequency Division Multiple Access
FKP - Flächen-Korrektur-Parameter
MAC- Master Auxiliary Concept
VRS – Virtual Reference Station
PPP-Wizard: Precise Point Positioning With Integer and Zero-difference Ambiguity Resolution Demonstrator
IGS - International GNSS Service
OSR - Observation Space Representation
SSR - State Space Representation
SSM - State Space model (modèle d'espace d'état)
CORS - Continuously Operating Reference Stations
SPB - Satellite phase Bias
ZTD - Zenith Tropospheric Delay
FAR – Full Ambiguity resolution
PAR – Partial Ambiguity resolution
ILS – Estimation Moindre carrés
DOP -Dilution of precision

Annexe 4 - Corrections PPP-RTK Biais Code

1. Biais Code Réseau CORS 1PPP-RTK

1.1 Fréquence L5 GPS

L5	G29	G10	G23	G08	G31	G26	G07
G29		0,432	0,16	-0,685	-0,307	-0,797	0,687
G10			-0,429	0,845	-0,897	-0,854	0,0557
G23				0,448	0,826	0,638	-0,644
G08					-0,254	-0,316	-0,087
G31						-0,892	-0,263
G26							-0,855

1.2 Fréquence E1/E5 Galileo

E1/ E5	E05	E09	E15
E05		0,181	0,979
E09	-0,357		0,486
E15	-0,382	-0,574	

2 Biais Code Réseau CORS 2 PPP-RTK

2.1 Fréquence L5 GPS

L5	G15	G16	G18	G23	G26	G27	G29	G31
G15		0,309	0,355	0,481	0,233	0,154	-0,716	0,501
G16			0,699	-0,331	-0,372	-0,534	0,0557	0,0557
G18				0,448	0,171	-0,922	0,166	-0,652
G23					0,393	0,51	0,635	-0,782
G26						0,944	0,64	-0,311
G27							-0,714	-0,73
G29								0,991

2.2 Fréquence E1/E5 Galileo

E1/ E5	E15	E21
E15	 	0,49

Annexe 5 - Corrections PPP-RTK Biais Phase

1. Biais Phase Réseau CORS 1PPP-RTK

1.1 Fréquence L1/L2 GPS

L1/ L2	G29	G10	G23	G08	G31	G26	G07
G29	 	-0,591	0,263	-0,457	0,852	0,579	-0,799
G10	0,064	 	-0,961	0,565	-0,595	-0,292	0,99
G23	-0,314	0,816	 	-0,199	0,706	0,728	0,284
G08	-0,252	-0,213	0,134	 	-0,277	-0,733	-0,227
G31	-0,475	-0,811	0,326	0,586	 	-0,617	-0,19
G26	-0,747	0,571	-0,522	-0,785	-0,761	 	0,011
G07	-0,154	0,04	0,175	0,691	-0,515	0,636	

1.2 Fréquence L5 GPS

L5	G29	G10	G23	G08	G31	G26	G07
G29	 	0,625	0,422	-0,631	-0,23	-0,528	-0,323
G10	 	 	0,741	0,732	-0,556	-0,498	0,084
G23	 	 	 	-0,483	0,459	0,262	0,098
G08	 	 	 	 	0,858	0,286	-0,063
G31	 	 	 	 	 	-0,279	-0,777
G23	 	 	 	 	 	 	0,575

1.3 Fréquence E1/E5 Galileo

E1/ E5	E05	E09	E15
E05	 	0,035	-0,47
E09	0,554	 	0,023
E15	0,255	-0,988	

2. Biais Code Réseau CORS 2 PPP-RTK

2.1 Fréquence L5 GPS

L5	G15	G16	G18	G23	G26	G27	G29	G31
G15		-0,896	-0,884	0,551	0,462	-0,978	0,867	-0,333
G16			-0,141	0,094	0,042	0,974	0,0557	0,0557
G18				0,448	-0,54	-0,235	-0,271	-0,27
G23					-0,074	0,66	0,882	-0,31
G26						-0,311	0,702	0,096
G27							0,495	0,766
G29								-0,864

2.2 Fréquence E1/E5 Galileo

E1/ E5	E15	E21
E15		-0.554

Titre : Algorithmes d'hybridation pour du positionnement précis GNSS

Mots clés : GNSS, RTK, PPP, SSR, Hybridation

Résumé : Le positionnement précis vient prendre une grande place dans la nouvelle dynamique de la smart city avec par exemple : la voiture autonome, les drones, les robots connectés utilisés dans l'agriculture ou encore dans le monde industriel. La principale méthode utilisée est le positionnement GNSS (Global Navigation Satellite Systems) qui utilise des données issues des satellites. Aujourd'hui, les trois principales techniques sont le RTK (Real Time Kinematic), le PPP (Precise Point Positioning) et le PPP-RTK une hybridation des deux précédentes. La thèse s'inscrit dans cet écosystème et a pour principal objectif l'hybridation de techniques existantes pour une meilleure adaptation au marché industriel. Dans un premier temps, nous avons dressé un état de l'art sur le

positionnement GNSS et étudié selon différents critères de performances comme la précision et le temps de convergence, les trois principales techniques existantes. Suite à cela, l'étude d'une nouvelle hybridation a été mise en œuvre PPP-NRTK. En effet, nous avons hybridé deux techniques existantes : PPP-RTK et NRTK (Network RTK). Ainsi, nous voulions garder une position précise en continu même avec une méthode de calcul globale comme le PPP ou le PPP-RTK et avec une initialisation du calcul précise et rapide en NRTK. La dernière partie de la thèse a été consacrée à sa mise en œuvre pratique et une analyse de ses différents critères de performances comme le temps de convergence et la précision, en vue de la comparer avec les techniques déjà existantes étudiées dans l'état de l'art.

Title : Hybridization algorithm for precise GNSS positioning

Keywords : GNSS, RTK, PPP, SSR, Hybridization

Abstract : Precise positioning is playing a major role in the new dynamics of the smart city, with autonomous cars, drones and connected robots used in agriculture and industry. The main method used is GNSS (Global Navigation Satellite Systems) positioning, which uses data from satellites. Today, the three main techniques are RTK (Real Time Kinematic), PPP (Precise Point Positioning) and PPP-RTK, a hybridization of the previous two. The main aim of this thesis is to hybridize existing techniques for better adaptation to the industrial market. As a first step, we reviewed the state of the art in GNSS positioning and studied the three main existing techniques in terms of

performance such as accuracy and convergence time. This was followed by a study of a new hybridization approach. Indeed, we hybridized two existing techniques: PPP-RTK and NRTK (Network RTK) In this way, we wanted to maintain a continuously accurate position even with a global calculation method such as PPP or PPP-RTK, and with a precise and fast calculation initialization in NRTK. The last part of the thesis was devoted to its practical implementation and an analysis of its various performance criteria such as convergence time and accuracy, with a view to comparing it with the existing techniques studied in the state of the art.