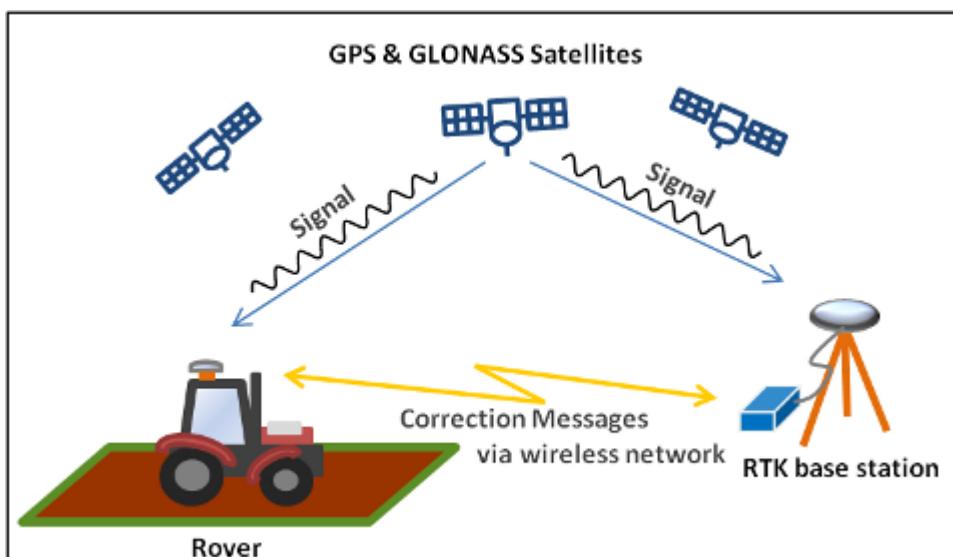


## Travail de Bachelor

# Géolocalisation centimétrique à l'aide de capteurs RTK pour les activités sportives

Confidentiel



Source : [https://www.magellan.jp/english/item/img/img151fi1239\\_1.png](https://www.magellan.jp/english/item/img/img151fi1239_1.png)

Étudiant :

Isaïa Spinelli

Travail proposé par :

Yersin Lionel

*Advanced Sport Instruments SA*

Avenue de Beaumont 5

1012 Lausanne

Enseignant responsable :

Mosqueron Romuald

Année académique

2019-2020



# Cahier des charges

## Géolocalisation centimétrique à l'aide de capteurs RTK pour les activités sportives

Confidentiel

### Contexte

Advanced Sport Instruments SA (<https://asi.swiss>) est une société Suisse basée à Lausanne qui développe, produit et commercialise des capteurs dédiés principalement à la mesure de performances dans le sport.

La grande majorité des capteurs d'ASI embarquent des récepteurs GNSS (GPS, Glonass, Galileo) de haute qualité et de bonne précision. Néanmoins, ces récepteurs sont basés sur la technologie GNSS standard qui offre des précisions limitées.

Une technologie plus avancée, le RTK (real time kinematics), qui utilise la mesure de la phase des signaux analogiques émis par les satellites, permet de déterminer avec une plus grande précision le positionnement, cela grâce à la comparaison avec les mêmes signaux reçus par un point de référence fixe nommé station de base. La technologie RTK permet d'atteindre des précisions absolues inférieures au centimètre.

Le projet proposé par ASI consiste à mettre en place un système basé sur la technologie RTK dans le but de localiser avec précision des athlètes, mais surtout de mesurer avec précision le passage des athlètes à certains points.

Par exemple, dans le cas du ski alpin, le but serait de mesurer avec précision le passage d'une ligne d'arrivée à l'aide de capteurs basés sur la technologie RTK. Pour cela, une station de base est placée à l'une des extrémités de la ligne d'arrivée. A l'autre extrémité, un capteur RTK est placé. Un autre capteur RTK est placé sur le skieur.

Les mesures effectuées par les différents capteurs permettent donc de déterminer avec une grande précision la position du skieur par rapport à la ligne d'arrivée.

## Objectifs

Le but du projet est de valider ce concept à l'aide de kits de développement RTK ou de capteurs fournis par ASI, en utilisant notamment une interface de communication basée sur la technologie LTE également disponible sur les capteurs ASI.

Concrètement, il s'agit de développer le logiciel embarqué spécifique à l'application de façon à pouvoir déterminer les différentes positions et distances. La possibilité de visualiser les résultats obtenus est nécessaire. Aucun développement matériel n'est requis pour ce projet, le matériel étant fourni par ASI.

## Déroulement global

Dans un premier temps, une partie recherche et analyse sera effectuée afin de se familiariser avec la technologie, le matériel et le moyen de communication entre une station de base et un capteur(rover).

Dans un deuxième temps, il s'agira de valider le concept à l'aide d'un kit de développement. Pour cela, deux modules seront nécessaires. Un premier utilisé comme station de base qui enverra les corrections de position via une liaison série au PC. Un deuxième module sera utilisé comme « capteur » (rover) qui corrigera sa position via les corrections reçues par la station de base via le PC. Une précision dans l'ordre du centimètre est souhaitée.

Ensuite, il faudra remplacer le rover implémenté sur un module du kit de développement par un capteur de ASI. Pour cela, il faudra implémenter la partie logiciel nécessaire au bon fonctionnement du RTK. Ultérieurement, il sera aussi nécessaire de passer par une communication sans fil et non plus sérielle pour un système final fonctionnel.

Finalement, la station de base mise en œuvre par un module du kit de développement devra être remplacée par un système de l'entreprise. Un système complet implémenté par le matériel d'ASI devra être fonctionnel.

## Critères de succès

Il sera possible de voir en temps réel la position d'un rover implémenté par un capteur de l'entreprise avec une précision inférieure à 8 centimètres. Les corrections de positions reçues par le rover devront être fournies par une station de base, elle aussi, mise en œuvre sur du matériel de l'entreprise. Ces corrections devront être communiquées sans fil via un module LTE.

Département TIC  
Filière Informatique  
Orientation Embarqué  
Étudiant Isaia, Spinelli  
Enseignant responsable Mosqueron Romuald

## Travail de Bachelor 2019-2020

### Géolocalisation centimétrique à l'aide de capteurs RTK pour les activités sportives

## Résumé

Ce travail de diplôme est mandaté par une entreprise développant des capteurs de géolocalisation dédiés principalement aux sports. Le projet consiste à améliorer la précision de leurs capteurs permettant d'atteindre une précision centimétrique.

La technologie utilisée est le RTK (*real time kinematics*). Celle-ci est basée sur la géolocalisation par satellite avec une amélioration. Elle consiste à utiliser une station au sol fixe, nommée station de base, qui mesure en permanence l'erreur et transmet un signal de correction aux capteurs mobiles environnants.

En premier lieu, une validation du RTK a été réalisée avec un kit de développement fourni par l'entreprise mandante.

Ensuite, l'implémentation de la technologie a pu être mise en œuvre sur le matériel de l'entreprise avec succès.

Finalement, des tests et analyses ont été effectués afin de garantir la précision centimétrique souhaitée des modules en mouvement. Les résultats ont montré que la précision dépend de la précision de positionnement de la station de base (fixe). Dans le meilleur des cas, une précision de 1,4 centimètres horizontale et 1 centimètre verticale a pu être obtenue.

## « Confidentialité »

### Confidentialité liée au Travail de Bachelor (TB)

Diplômant : Spinelli Isaïa  
Titre du travail de Bachelor : Géolocalisation centimétrique à l'aide de capteurs RTK pour les activités sportives  
Enseignant responsable du TB : Mosqueron Romuald  
Entreprise partenaire : Yersin Lionel (ASI)

Tous les TB sont déposés à la Bibliothèque de la HEIG-VD qui en gère l'archivage et la consultation. Quel que soit le niveau de confidentialité du TB, le nom du diplômant, le nom de l'enseignant responsable, le titre du TB et le résumé publiable figurent dans tous les documents de présentation des TB ainsi que sur la plateforme de consultation des TB (<http://tb.heig-vd.ch>). L'enseignant responsable veille à ce que le titre du TB et le résumé publiable soient rédigés conformément au niveau de confidentialité voulu.

Les TB peuvent être soumis à un logiciel anti-plagiat. Dans ce cas, leur contenu sera traité de manière confidentielle.

Le TB n'est pas confidentiel

Outre les informations mentionnées ci-dessus, les documents de présentation du TB contiennent également le nom de l'entreprise partenaire, le résumé publiable et une affiche. Le TB peut être consulté sur la plateforme des TB.

Le TB est confidentiel. Les conditions suivantes de diffusion des informations sont appliquées :

Aucune consultation ou emprunt du TB n'est permis hormis par l'enseignant responsable du TB et le diplômant qui s'engagent à ne pas faire usage des informations mises à leur disposition. Le TB porte la mention « confidentiel ».

Oui  Non Nous acceptons que le nom de l'entreprise partenaire figure dans les documents publiés (titre, résumé, affiche, etc.), ainsi que dans la plateforme de consultation des TB.

Oui  Non Nous acceptons que l'affiche du TB figure sur la plateforme de consultation des TB (l'affiche est au préalable validée par l'entreprise partenaire).

Dans tous les cas, un accord de confidentialité doit être signé par le diplômant, l'expert et toutes les personnes participant à l'évaluation du TB.

Nous déclarons accepter les conditions de diffusion du Travail de Bachelor indiquées.

Diplômant

Date Nom et signature  
13.08.2020 Spinelli Isaïa

Enseignant responsable

Date Nom et signature  
13.08.2020 Mosqueron Romuald

ASI

Date Nom et signature  
13.08.2020 Yersin Lionel

N.B.: Ce document fait partie intégrante du cahier des charges du TB.  
La forme masculine est utilisée comme genre neutre et désigne à la fois les hommes et les femmes.

# Préambule

Ce travail de Bachelor (ci-après TB) est réalisé en fin de cursus d'études, en vue de l'obtention du titre de Bachelor of Science HES-SO en Ingénierie.

En tant que travail académique, son contenu, sans préjuger de sa valeur, n'engage ni la responsabilité de l'auteur, ni celle du jury du travail de Bachelor et de l'école.

Toute utilisation, même partielle, de ce TB doit être faite dans le respect du droit d'auteur.

HEIG-VD

Peiris Vincent  
Chef du Département TIC

Yverdon-les-Bains, le 12 avril 2020

# Remerciements

Je tiens à remercier les personnes suivantes qui m'ont aidé et soutenu lors de ce travail :

- Monsieur Romuald Mosqueron, professeur responsable de ce travail de diplôme, pour son suivi régulier, ses conseils et sa disponibilité durant ce travail.
- Monsieur Lionel Yersin, mandant du travail de diplôme, pour ses renseignements, ses conseils avisés et sa disponibilité durant ce travail.
- Monsieur Romain Humair, technicien dans l'entreprise ASI, pour ses explications, sa disponibilité et toutes ses réponses.
- L'équipe de géomètre de la HEIG-VD (Jotterand Didier, Bron Maximin, Gressin Adrien, Guillaume Sébastien et Benjamin Mignot) pour leurs explications et leurs informations concernant les points de mesures.
- Madame Marion Dutu-Launay, ma compagne, pour son soutien durant toute la durée de ce travail.
- Madame Spinelli Rachel, ma tendre et chère mère adorée, et Madame Spinelli Noëlla, ma chère sœur, pour leur soutien et leur aide indispensable pour la rédaction et les corrections de ce rapport.

Finalement, je souhaite remercier tous mes proches pour leur soutien constant et leur attention à mes explications abracadabrantess.

# Table des matières

1	Introduction.....	12
1.1	Contexte.....	12
1.2	Structure du rapport.....	12
2	Pré-étude.....	13
2.1	Géolocalisation par satellite .....	13
2.1.1	Fonctionnement.....	13
2.1.2	Systèmes de navigation satellitaires existants .....	14
2.1.3	Applications.....	14
2.2	Différentes technologies d'amélioration existantes .....	14
2.2.1	Augmentation basée sur des satellites .....	14
2.2.2	Augmentation basée sur des systèmes à bord .....	15
2.2.3	Augmentation basée sur des systèmes au sol .....	15
2.3	Choix final.....	15
3	Fonctionnement du RTK .....	16
3.1	Fonctionnement global.....	16
3.1.1	Le réseau RTK (NRTK) .....	17
3.2	Les données de corrections .....	17
3.2.1	Les messages d'observations .....	17
3.2.2	Les messages de la station de référence .....	18
3.2.3	Les messages éphémérides .....	18
3.3	La communication .....	18
3.3.1	Radio .....	18
3.3.2	LTE-M .....	18
3.4	Applications.....	20
4	Validation du concept .....	21
4.1	Le kit de développement RTK .....	21
4.1.1	Le matériel .....	21
4.1.2	Le logiciel « u-center ».....	22
4.2	Prise en main et tests de fonctionnement .....	22
4.3	Configurations .....	22
4.3.1	Générales .....	22

4.3.2	Rover.....	24
4.3.3	Station de référence.....	25
4.4	Méthode d'analyse.....	28
4.4.1	Champs de comparaison .....	28
4.4.2	Les différents modes .....	28
4.5	Comparatif .....	29
4.5.1	Précision.....	29
4.5.2	Qualité .....	30
4.5.3	Déviation.....	30
5	Implémentation sur le matériel de l'entreprise .....	31
5.1	Analyse du logiciel embarqué .....	31
5.2	Implémentation sur le capteur actuel.....	33
5.2.1	Présentation du capteur de ASI .....	33
5.2.2	Implémentation sur le capteur .....	36
5.3	Implémentation sur le capteur final.....	43
5.3.1	Configuration du module GNSS .....	43
5.3.2	Fonctionnement global d'un rover .....	44
5.3.3	Fonctionnement global d'une station de référence .....	45
5.4	Implémentation des applications serveur .....	47
5.4.1	Utilité de l'application serveur.....	47
5.4.2	Fonctionnement de l'application serveur .....	49
5.4.3	Fonctionnement de la deuxième application serveur .....	54
5.4.4	Interactions du système complet.....	57
6	Tests et analyses.....	62
6.1	Tests des rovers.....	62
6.2	Analyse approfondie des données de corrections (RTCM3) .....	63
6.3	Analyse de la précision de la position absolue .....	64
6.3.1	Précision horizontale .....	65
6.3.2	Précision verticale .....	67
6.4	Analyse de la précision relative horizontale .....	68
6.4.1	ASI float.....	68
6.4.2	ASI fixe .....	69
6.4.3	HEIG fixe.....	70
6.5	Analyse de la stabilité .....	72

6.5.1	Stabilisation des temps de mise en place.....	72
6.5.2	Modes RTK .....	74
6.6	Réelle différence avec ou sans le RTK .....	75
6.6.1	Dérive.....	75
6.6.2	Exemples d'itinéraires .....	81
7	Conclusion .....	83
7.1	Conclusion du projet .....	83
7.2	Améliorations .....	83
7.3	Conclusion personnelle .....	83
8	Authentification.....	84
9	Références.....	85
10	Liste des abréviations .....	86
11	Annexes.....	87

## Table des tableaux

Tableau 2-1 : Tableau comparatif GNSS .....	15
Tableau 4-1 : Données comparatives des modes .....	29
Tableau 6-1 : Données comparatives des précisions .....	62
Tableau 6-2 : Données de l'analyse des données de corrections (RTCM3).....	63
Tableau 6-3 : Données des mesures de la précision absolue horizontale .....	65
Tableau 6-4 : Données des mesures de la précision verticale .....	67
Tableau 6-5 : Données de la précision relative (ASI float).....	68
Tableau 6-6 : Données de la précision relative (ASI fixe) .....	69
Tableau 6-7 : Données de la précision relative (HEIG fixe).....	70
Tableau 6-8 : Données de la stabilité du temps de mise en place (ASI).....	72
Tableau 6-9 : : Données de la stabilité du temps de mise en place (HEIG) .....	73
Tableau 6-10 : Données de la stabilité des modes RTK .....	74

## Table des figures

Figure 2-1 : Visualisation d'une triangulation [2] .....	13
Figure 3-1 : Schéma général RTK.....	16
Figure 3-2 : Schéma NTRIP [7] .....	19
Figure 4-1 : Matériel du kit de développement .....	21
Figure 4-2 : Configuration GNSS .....	22
Figure 4-3 : Configuration de navigation .....	23
Figure 4-4 : Configuration des ports.....	24
Figure 4-5 : Configuration du DGNSS.....	24
Figure 4-6 : Configuration de la fréquence .....	25
Figure 4-7 : Configuration du mode .....	26

Figure 4-8 : Configuration des messages.....	27
Figure 4-9 : Déviation mode Normal      Figure 4-10 : Déviation mode RTK (float) Figure 4-11 : Déviation mode RTK (fixe)	30
Figure 5-1 : Fonctionnement simplifié du logiciel embarqué.....	31
Figure 5-2 : Présentation du capteur d'ASI.....	33
Figure 5-3 : Système global souhaité.....	34
Figure 5-4 : Ajout dans le logiciel (rover).....	36
Figure 5-5 : Système de test (rover) .....	37
Figure 5-6 : Ajout dans le logiciel (base) .....	40
Figure 5-7 : Système de test (base) .....	41
Figure 5-8 : Diagramme logiciel d'un rover .....	44
Figure 5-9 : Diagramme logiciel d'une station de référence .....	45
Figure 5-10 : Fonction de l'application serveur .....	47
Figure 5-11 : Fonction de l'application serveur 2.....	48
Figure 5-12 : Diagramme du main de l'application serveur .....	49
Figure 5-13 : Diagramme du thread (station de base) .....	50
Figure 5-14 : Diagramme du thread (rover).....	51
Figure 5-15 : Exemple du fichier TXT .....	52
Figure 5-16 : Exemple du fichier KML.....	53
Figure 5-17 : Exemple d'itinéraire (KML) .....	53
Figure 5-18 : Diagramme du main de la seconde application serveur.....	54
Figure 5-19 : Analyse des paquets RTCM3 (HEIG-VD) .....	56
Figure 5-20 : Diagramme de séquence du système RTK .....	58
Figure 5-21 : Diagramme de séquence du système NRTK.....	60
Figure 6-1 : Placement d'un point de mesure.....	64
Figure 6-2 : Placement du capteur sur un point de mesure .....	65
Figure 6-3 : Dispersions des points (ASI float) .....	69
Figure 6-4 : Dispersions des points (ASI fixe) .....	70
Figure 6-5 : Dispersions des points (HEIG fixe) .....	71
Figure 6-6 : Dérive de tous les modes à 1 Hz .....	76
Figure 6-7 : Dérive des modes RTK à 1Hz .....	76
Figure 6-8 : Dérive du mode fixe à 1Hz.....	77
Figure 6-9 : Dérive des modes RTK à 5Hz .....	78
Figure 6-10 : Dérive du mode fixe à 5Hz.....	78
Figure 6-11 : Dérive de tous les modes à 10 Hz .....	79
Figure 6-12 : Dérive des modes RTK à 10 Hz.....	79
Figure 6-13 : Dérive du mode fixe à 10 Hz.....	80
Figure 6-14 : Itinéraire d'un rover en mode normal .....	81
Figure 6-15 : Itinéraire d'un rover en mode RTK.....	81
Figure 6-16 : Visualisation de la taille de la grille .....	82
Figure 6-17 : Différence de contour de la grille .....	82

# 1 Introduction

## 1.1 Contexte

De tout temps, l'être humain a toujours été dans la recherche d'une amélioration dans tous les domaines ; qualité de vie, gain de temps, d'argent et même de compétitivité.

Advanced Sport Instruments SA (<https://asi.swiss>) est une entreprise Lausannoise qui produit et commercialise des capteurs dédiés principalement à la mesure de performances sportives.

Dans le but d'améliorer la précision de son matériel, ASI propose un projet consistant à atteindre une précision au centimètre.

Effectivement, dans le domaine du sport de compétition une telle précision est vivement attendue. La localisation d'un(e) athlète par rapport à un endroit précis, comme une ligne d'arrivée, est une donnée capitale.

C'est pourquoi ce projet consiste à développer le logiciel embarqué permettant une précision centimétrique.

## 1.2 Structure du rapport

Ce rapport se divise en deux parties.

La première contient une pré-étude des différentes technologies de géolocalisation ainsi qu'une explication plus détaillée sur la technologie choisie.

La deuxième détaille le processus de conception : validation du concept, implémentation, tests et analyses.

## 2 Pré-étude

Dans ce chapitre se trouve des présentations succinctes de différentes techniques et principes de géolocalisation qui vont permettre de choisir la technologie la plus appropriée pour ce projet.

Parmi les différentes techniques de géolocalisation extérieure, les plus actives sont par satellite et par GSM (Global System for Mobile Communications). Cependant, la précision offerte par la technologie utilisant le GSM est vite limitée par des centaines de mètres. De ce fait, la méthode qui convient le mieux à ce projet est la géolocalisation par satellite. Dépendamment de plusieurs facteurs, la précision de cette technique peut varier de 10m à 100m[1].

Cette précision est intéressante. Cependant, pour atteindre la précision souhaitée de quelques centimètres, il est nécessaire d'améliorer cette technique.

### 2.1 Géolocalisation par satellite

Un système de positionnement par satellite permet de déterminer une position 3D (longitude, latitude, altitude), la date et l'heure à l'aide d'un petit récepteur portable qui reçoit des signaux horaires transmis par plusieurs satellites d'une ou plusieurs constellations de satellites artificielles.

Une constellation de satellites artificielle est composée de 18 à 30 satellites en orbite de la terre. Les satellites sont placés afin que leurs couvertures au sol respectives se chevauchent et se complètent au lieu d'interférer entre elles.

#### 2.1.1 Fonctionnement

Le récepteur reçoit simultanément plusieurs signaux envoyés par plusieurs satellites. Chaque signal comporte essentiellement deux informations importantes :

1. La position du satellite
2. L'heure et la date d'émission du signal

Le récepteur peut calculer la distance à laquelle se trouve chaque satellite grâce à la différence de temps entre l'envoi et l'arrivée du signal. Grâce à la triangulation, avec un minimum de trois distances connues, le récepteur est capable de connaître sa position.

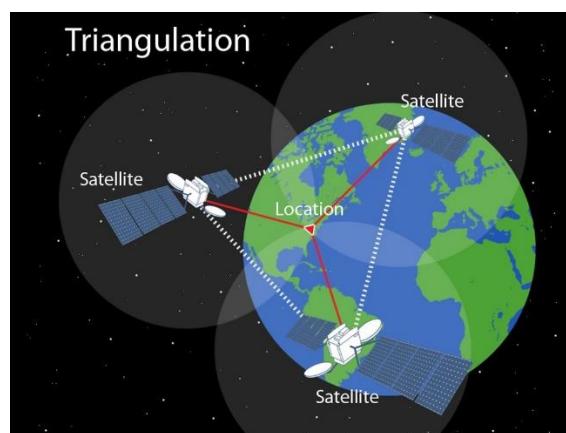


Figure 2-1 : Visualisation d'une triangulation [2]

En réalité, nous ne sommes pas sur un plan (2D) mais dans l'espace (3D). Il faudrait donc utiliser des sphères et non pas des cercles. L'intersection de trois sphères donne deux points et non plus, un seul. Cependant, il est possible d'exclure un des deux points car il ne correspond pas à une position réelle sur

terre. Toutefois, un quatrième satellite est tout de même nécessaire afin de synchroniser l'horloge du récepteur avec l'horloge atomique dont disposent les satellites.

### 2.1.2 Systèmes de navigation satellitaires existants

Pour parler d'assistant de navigation, il arrive fréquemment de parler de GPS. Pourtant, le GPS est le système de positionnement par satellite appartenant au gouvernement des États-Unis. En effet, il existe plusieurs systèmes de positionnement créés par différents états :

- **GPS (Global Positioning System)** : Appartient aux États-Unis, pleinement opérationnel depuis 1995.
- **GLONASS (globalnaïa navigatsionnaïa spoutnikovaïa sistéma)** : Appartient à la Russie, opérationnel entre 1996 et 1999, puis de nouveau opérationnel depuis 2010.
- **Galileo** : Développé par l'Union européenne. Les premiers services de Galileo sont opérationnels depuis le 15 décembre 2016. Galileo ne sera pas pleinement déployé avant 2020.
- **BeiDou** : Développé par la Chine. Les premiers services sont opérationnels depuis 2003. BeiDou ne sera pas pleinement déployé avant 2020.
- **IRNSS (L'Indian Regional Navigation Satellite System)** : Système de positionnement régional développé par l'Inde, opérationnel depuis 2014.
- **QZSS (Quasi-Zenith Satellite System)** : Système de positionnement régional développé par le Japon lancé en 2002.

De nos jours, les majorités des récepteurs GNSS utilisent simultanément plusieurs systèmes de positionnement par satellite. Ceci permet d'augmenter la précision et la stabilité du positionnement des utilisateurs. En effet, il arrive souvent que des signaux n'arrivent pas correctement au récepteur à cause de bâtiments, d'arbres, de montagnes, etc. Le fait de travailler avec différentes constellations de satellites diminue grandement ce problème de couverture.

### 2.1.3 Applications

Les utilisations des systèmes de positionnement par satellite sont variées :

- La navigation en général
- La surveillance
- La topographie
- Le suivi des animaux
- La gestion des réseaux de transport
- L'agriculture de précision
- La lutte contre le vol
- Etc.

## 2.2 Différentes technologies d'amélioration existantes

Il existe déjà plusieurs méthodes améliorant les performances de la géolocalisation par satellite. Trois catégories existantes sont définies :

### 2.2.1 Augmentation basée sur des satellites

Le SBAS (Satellite-Based Augmentation System) est une technologie qui permet d'améliorer la précision du GNSS en utilisant des satellites supplémentaires pour atteindre une précision inférieure à 2 mètre.

## 2.2.2 Augmentation basée sur des systèmes à bord

Le ABAS (Aircraft Based Augmentation System) est spécialement utilisé dans l'aviation car elle offre une haute performance dans la fiabilité et la constance des mesures.

## 2.2.3 Augmentation basée sur des systèmes au sol

Le GBAS (*Ground-Based Augmentation system*) est une augmentation utilisée entre autre sur les pistes d'atterrissements car sa précision est inférieure au mètre.

Il existe différentes technologies basées sur ce principe :

### 2.2.3.1 PPK

Le PPK (Post-Processed Kinematic) utilise une seule antenne au sol et applique les données de corrections ultérieurement.

### 2.2.3.2 NRTK

Le NRTK (Network Real Time Kinematic) consiste à mettre en commun toutes les données de plusieurs antennes de référence. Les données de corrections de l'antenne la plus proche du récepteur sont transmises en temps réel.

### 2.2.3.3 RTK

Le RTK (Real Time Kinematic) utilise une seule antenne et transmet les données de corrections en temps réel.

## 2.3 Choix final

Afin d'avoir une bonne vision comparative des différentes méthodes existantes, voici un tableau complémentaire :

	GNSS	GNSS + SBAS	GNSS + ABAS	GNSS + GBAS		
				NRTK	PPK	RTK
Compléxité	faible	faible	haute	haute	moyenne	moyenne
Précision	10m	2m	1m	5 cm	5 cm	5 cm
Fiabilité	moyen	moyen	forte	moyen	moyen	moyen
Couverture	globale	régional / continental	à bord	régional	20 km	20 km
Temps réel	oui	oui	oui	oui	non	oui

Tableau 2-1 : Tableau comparatif GNSS

Les différentes caractéristiques qui ne conviennent pas au système final recherché sont en rouge. Il est possible de constater que la technologie de géolocalisation RTK est celle qui convient le mieux pour ce projet. Ce concept permet une précision centimétrique en temps réel. La technologie NRTK est plus complexe, mais tout de même envisageable.

### 3 Fonctionnement du RTK

Le RTK est une technique qui améliore la précision du positionnement en utilisant le GNSS. Cette technique utilise une station de référence installée à une position fixe. Connaissant sa position, la station de référence est capable de générer des messages de corrections complémentaires aux signaux GNSS. Ces corrections permettent à des rovers, aux alentours de la station de référence, de calculer leur position avec une précision centimétrique.

#### 3.1 Fonctionnement global

La station de référence utilise la porteuse<sup>1</sup> modulée par le code émis, et non le code binaire contenu dans le signal, ignorant les informations qu'il contient (position du satellite, heure, date, ...). Cela permet une grande amélioration de la précision dû au fait que la fréquence de la porteuse (1575,42 MHz) est environ plus de 1000 fois plus grande que le changement de phase du code d'acquisition grossier.

Voici un schéma afin de mieux comprendre l'ensemble d'un système RTK :

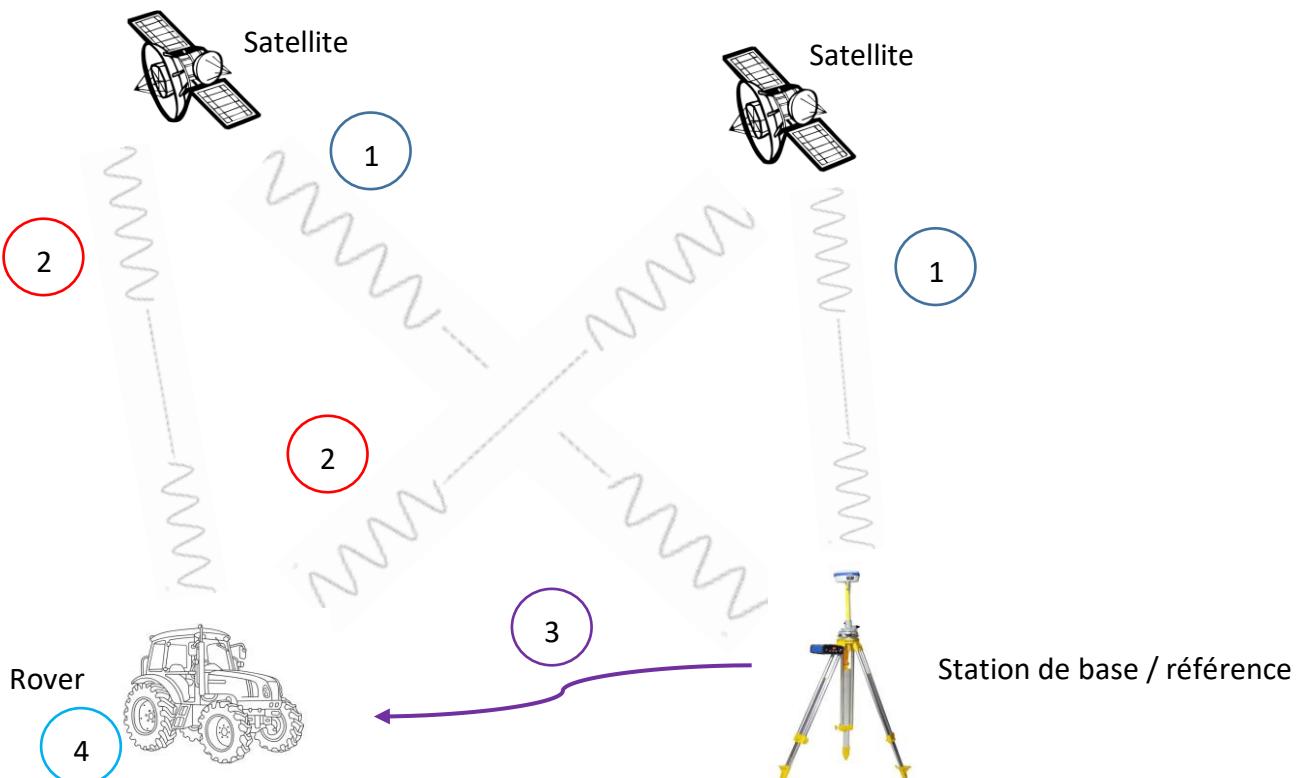


Figure 3-1 : Schéma général RTK

1. La station de base réceptionne les signaux émis par plusieurs satellites. Elle calcule les distances, sa position et génère des données de corrections<sup>2</sup> en fonction de sa véritable position.
2. Le rover réceptionne les signaux émis par plusieurs satellites. Il calcule les distances et sa position.
3. La station de référence transmet les données de corrections au rover.
4. Le rover applique les corrections afin de connaître sa position avec plus de précision.

<sup>1</sup> Une porteuse (L1) est une forme d'onde qui est modulée par un signal d'entrée dans le but de transporter des informations

<sup>2</sup> Le format des données de corrections est expliqué au chapitre « [les données de corrections](#) ».

Il est préférable que le rover ne soit pas trop loin de la base de référence ( $\sim 20$  km). En effet, les situations atmosphériques peuvent varier en fonction de la composition de l'ionosphère et de la troposphère. De plus, les dispositions géographiques aussi peuvent induire des perturbations. C'est pour ces raisons que si le rover se trouve trop loin de la station de base, les corrections pourraient ne plus être appropriées.

### 3.1.1 Le réseau RTK (NRTK)

Une station de référence a divers inconvénients. En effet, le matériel nécessaire pour une station de base coûte cher. De plus, elle demande beaucoup de temps de paramétrage et d'installation. Afin d'éviter tous ces désagréments à la communauté des utilisateurs, l'idée est de mettre en place des stations fixes permanentes. Ces stations mettent à disposition leurs données en continu via le réseau internet. De ce fait, en fonction d'où se trouve un utilisateur, il peut se connecter via une liaison internet au flux de données de corrections de l'antenne la plus proche.

Plusieurs réseaux publics et privés sont maintenant en place. En général, ces réseaux sont organisés par des états. En Suisse, swipos-GIS/GEO[3] est disponible partout. En général, ils proposent la technique VRS (stations de référence virtuelles). Cette technique permet de créer une station de référence virtuelle relativement proche d'un utilisateur.

Pour commencer, l'usager doit envoyer sa position approximative au serveur. Grâce à cette position, le serveur peut créer une station virtuelle relativement proche de l'utilisateur à l'aide d'un algorithme et des données de plusieurs stations de bases réelles autour de sa position. Ainsi, l'usager bénéficie de données de corrections précises en continu. Bien évidemment, l'inconvénient dans ce procédé est que les services qui proposent de telles prestations sont toujours payants. Pour avoir une idée, les services de swipos-GIS/GEO proposent un tarif de 0.50.- CHF par minute ou une licence à 2'000.- CHF par an.

## 3.2 Les données de corrections

Il existe plusieurs normes propriétaires applicables aux systèmes RTK. Par exemple, le CMR (Compact Measurement Record) est un protocole inventé par un fabricant de récepteur GNSS, « Trimble Navigation ».

Cependant, la norme la plus utilisée par les réseaux RTK actuels est le RTCM[4] (Commission technique radio des services maritimes) conçu par cette même organisation scientifique à but non lucratif. Il existe plusieurs versions de ce protocole, la troisième et dernière version est le RTCM3.

Plus concrètement, RTCM est une norme utilisée pour fournir aux récepteurs GNSS des données de corrections différentielles en temps réel. Plusieurs types de messages sont définis, voici ci-dessous, une courte description de ces types.

### 3.2.1 Les messages d'observations

Ce type de message permet de corriger les différents signaux reçus des satellites. Il est indispensable pour un fonctionnement RTK. Il existe deux types de messages d'observations, RTK et MSM (messages à signaux multiples).

Les messages de type RTK permettent de corriger les signaux venants des satellites des constellations de GPS et/ou GLONASS.

Les messages de type MSM permettent de corriger les signaux venants des satellites des constellations de GPS, GLONASS, Galilée, QZSS et/ou BeiDou. Il existe sept types de messages MSM numérotés de 1 à 7.

A mesure que le numéro augmente, la quantité d'informations transmises est plus élevée. En général, le type 4 est le plus utilisé : bien qu'il transmette peu de données, il s'agit du minimum suffisant pour un bon fonctionnement. Si la quantité de données n'est pas un problème, le type 7 est utilisé car il contient une meilleure résolution.

### 3.2.2 *Les messages de la station de référence*

Ce type de message permet d'informer les utilisateurs RTK des coordonnées et du matériel de la station de base. Il est indispensable qu'au moins un message de ce type parvienne aux rovers.

### 3.2.3 *Les messages éphémérides*

Ce type de message permet d'informer les utilisateurs des éphémérides des satellites. Ce sont des tables astronomiques par lesquelles on détermine, pour chaque jour, la position d'un satellite. Il est facultatif car les rovers peuvent télécharger leur propre table à partir des satellites.

Pour résumer, il existe trois grands types de messages de corrections RTCM3. Parmi ces trois, uniquement deux types sont indispensables, les informations de position de la station de référence ainsi que les informations de corrections des signaux satellites (messages d'observations).

Une liste de tous les messages de type RTCM est disponible à cette référence [\[5\]](#). Une analyse plus approfondie a été réalisée durant ce travail au chapitre [« Analyse approfondie des données de corrections »](#).

## 3.3 La communication

La communication entre une station de référence et un rover est l'élément du système le plus flexible. Il dépendra de nombreux paramètres tels que la localisation géographique, le matériel à disposition, les sens de communication, le type d'application finale, etc.

### 3.3.1 *Radio*

La communication radio (UHF, Ultra Haute Fréquence) est une des méthodes les plus utilisées pour un système RTK simple. En effet, la radio est, en général un moyen de communication simple, peu coûteux et rapide à mettre en place. Cependant, elle ne permet pas une évolutivité du système. Effectivement, si un système RTK devient relativement étendu, il peut y avoir des problèmes d'allocations de fréquences ou d'interférences. De plus, si le système souhaite exploiter le réseau RTK, avec uniquement un moyen de communication par radio, cela ne sera pas possible. C'est pourquoi de nombreux systèmes utilisent un autre moyen de communication.

### 3.3.2 *LTE-M*

Le LTE-M est un type de norme de technologie radio de réseau étendu à faible puissance développé par 3GPP<sup>3</sup>[\[6\]](#). Cette technologie semble être bien appropriée car elle permet une communication point à point, tout en ayant une couverture extrêmement large. Cette méthode de communication est adéquate pour être mobile, semblable à un dispositif rover RTK. Ce moyen de communication permet donc de profiter pleinement d'un réseau RTK. On peut donc profiter d'une grande évolutivité et flexibilité du système. Cependant, cette technologie est moins facile à mettre en place et engendre un coût.

---

<sup>3</sup> 3GPP (3rd Generation Partnership Project) est une coopération entre organismes de normalisation en télécommunications

### 3.3.2.1 MQTT

MQTT (Message Queuing Telemetry Transport) basé sur le protocole TCP/IP est un protocole déjà utilisé par l'entreprise ASI. Ce protocole est établi sur le mécanisme d'abonnement/publication, qui convient bien au principe de communication entre un rover (s'abonne) et une station de base (publie). Il a été initialement développé par Andy Stanford-Clark (IBM) et Arlen Nipper (EuroTech). MQTT a été créé afin d'être efficace en bande passante, léger et utilisant peu d'énergie de batterie.

Ce protocole correspondrait bien à ce système et au matériel de l'entreprise. On peut facilement s'imaginer que chaque rover s'abonnerait à la station de base la plus proche et recevrait donc les données de corrections en continu. De plus, MQTT convient particulièrement bien aux rovers car il consomme très peu d'énergie. Cependant, tous les réseaux publics et privés déjà mis en place n'utilisent malheureusement pas ce protocole. Si l'on souhaite être compatible avec le reste des réseaux, il est nécessaire d'utiliser le même moyen de communication qui est le protocole NTRIP.

### 3.3.2.2 NTRIP

Depuis ces dernières années, le nombre d'applications d'améliorations GPS a grandement augmenté. Le transfert de données de GPS différentielles en temps réel requiert une bande passante relativement étroite. L'Agence fédérale allemande de cartographie et de géodésie et le Département d'informatique de l'Université de Dortmund ont développé un nouveau protocole de diffusion en continu de données GPS différentielles sur internet nommé NTRIP ( Networked Transport of RTCM via Internet Protocol). Ce protocole est maintenant standard et ouvert, il a été publié en septembre 2004.

NTRIP est un protocole générique sans état, basé sur le protocole de transfert hypertexte HTTP / 1.1. Les données sont conformes aux spécifications publiées par RTCM.

Voici une image qui schématise bien le fonctionnement du protocole NTRIP :

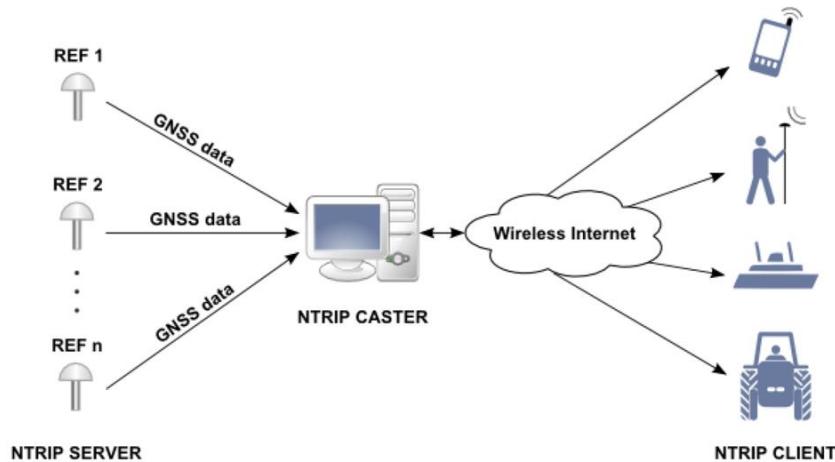


Figure 3-2 : Schéma NTRIP [7]

Les stations de références « **NTRIP serveur** » envoient les corrections de données GNSS à un serveur « **NTRIP caster** ». Ultérieurement, un utilisateur RTK fait une demande au serveur pour s'abonner à la station de base la plus proche. Ensuite, le rover reçoit le flux constant de corrections de la station de base choisie et peut se localiser avec précision.

### 3.4 Applications

Les utilisations des systèmes de positionnement précis par RTK sont variées :

- Le pilotage automatique
- L'agriculture de précision
- L'arpentage
- La topographie
- Etc.

Les systèmes RTK peuvent être utilisés pour toutes applications nécessitant une haute précision de position relative.

## 4 Validation du concept

Cette première partie pratique consiste à s'assurer que la technologie RTK fonctionne correctement et offre la précision souhaitée.

Pour faciliter la mise en œuvre de cette validation, un kit de développement RTK (C94-M8P) a été fourni par ASI.

Une fois le concept validé, une modification hardware sera effectuée sur les capteurs actuels d'ASI afin de bénéficier de la technologie RTK.

### 4.1 Le kit de développement RTK

Le kit « C94-M8P » est conçu par u-blox, une entreprise suisse qui crée des semi-conducteurs et modules sans fil pour le marché grand public.

#### 4.1.1 *Le matériel*

Ci-dessous, une image représentant les deux modules, prêts à l'emploi, inclus dans le kit de développement :

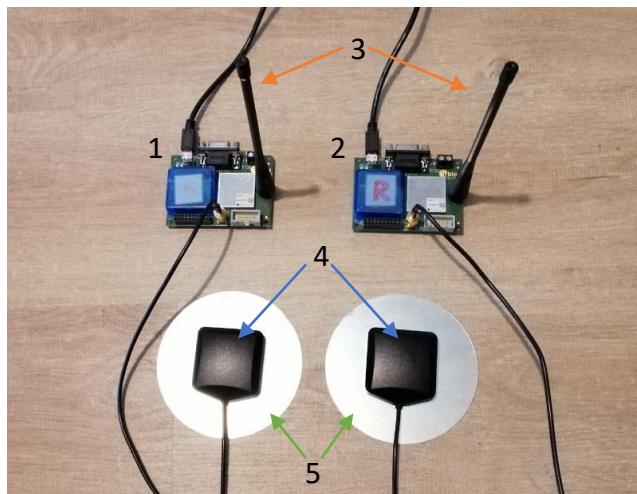


Figure 4-1 : Matériel du kit de développement

1. Une carte « NEO-M8P-2 » (en mode station de référence)
2. Une carte « NEO-M8P-2 » (en mode rover)
3. Les antennes radio (UHF)
4. Les antennes GNSS
5. Les plans de masse

Ce kit contient deux cartes d'applications avec chacune un module NEO-M8P-2 qui peut faire office de base de référence ou rover. Il est donc possible, avec ces deux cartes, de faire fonctionner un système complet avec une base et un rover. Ces deux modules peuvent aisément communiquer par UHF qui est intégré sur les cartes du kit.

#### 4.1.2 Le logiciel « u-center »

U-center est un programme interactif et facile à utiliser, il a été conçu par u-blox. Il permet de configurer et de profiter d'un affichage en temps réel des données des différents récepteurs GNSS proposé par u-blox.

### 4.2 Prise en main et tests de fonctionnement

Afin de s'assurer du bon fonctionnement du kit de développement, quelques tests ont été réalisés.

1. Vérification de la communication entre le PC et les modules.
2. Vérification la réception des signaux GNSS à l'aide de l'interface du logiciel u-center.
3. Vérification de la véracité des informations reçues à l'aide de Google maps.
4. Recherche de diverses configurations possibles.

### 4.3 Configurations

Les possibilités de configurations sont étendues. Certaines sont utiles uniquement pour un rover, d'autres pour une station de référence et d'autres sont plus générales.

#### 4.3.1 Générales

Voici une petite présentation de quelques configurations générales de bases des modules.

##### 4.3.1.1 GNSS

Le paramètre « UBX-CFG-GNSS » permet de configurer les paramètres GNSS du module. Il permet de spécifier quels signaux GNSS doivent être traités tout en limitant le nombre de canaux de suivis à allouer à chaque GNSS. Les combinaisons possibles dépendent de la capacité du récepteur à recevoir plusieurs fréquences porteuses. Voici la configuration de base du module :

UBX - CFG (Config) - GNSS (GNSS Config)							
ID	GNSS	Configure	Enable	Channels		Signals	
				min	max	<input checked="" type="checkbox"/> L1C/A	<input type="checkbox"/> L1C/A
0	GPS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	8	16	<input checked="" type="checkbox"/> L1C/A	<input type="checkbox"/> L1C/A
1	SBAS	<input type="checkbox"/>	<input type="checkbox"/>	0	0	<input type="checkbox"/> L1C/A	<input type="checkbox"/> E1
2	Galileo	<input type="checkbox"/>	<input type="checkbox"/>	0	0	<input type="checkbox"/> E1	<input type="checkbox"/> B1
3	BeiDou	<input checked="" type="checkbox"/>	<input type="checkbox"/>	8	16	<input checked="" type="checkbox"/> B1	<input type="checkbox"/> L1C/A
4	IMES	<input type="checkbox"/>	<input type="checkbox"/>	0	0	<input type="checkbox"/> L1C/A	<input type="checkbox"/> L1S
5	QZSS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0	3	<input checked="" type="checkbox"/> L1C/A	<input type="checkbox"/> L1S
6	GLONASS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	8	14	<input checked="" type="checkbox"/> L1OF	<input type="checkbox"/> L1OF

Figure 4-2 : Configuration GNSS

Le module utilise une seule fréquence porteuse pour recevoir des signaux de GPS et de QZSS car ils ont tous les deux la même fréquence. Comme eux, Galileo a la même fréquence, cependant, ce module ne prend malheureusement pas en charge ce dernier.

Pour la deuxième fréquence porteuse, le choix est possible entre BeiDou ou GLONASS. La configuration est laissée par défaut, elle utilise GLONASS car il a une bonne couverture globale.

#### 4.3.1.2 Navigation

Le paramètre « UBX-CFG-NAV5 » permet de configurer les paramètres de navigation du module. Il permet d'indiquer des informations supplémentaires au module afin d'améliorer son interprétation des mesures pour fournir une sortie de position plus précise.

Typiquement, le champ « Dynamic Model » permet d'indiquer pour quelle application le module est utilisé. L'exemple ci-dessous est la configuration d'une station de référence. Le type choisi est donc stationnaire, cela permet au module de savoir que la vitesse ainsi que l'accélération du capteur seront nulles.

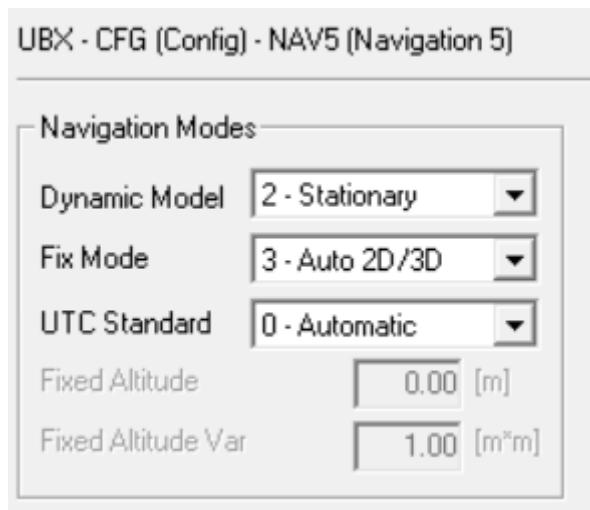


Figure 4-3 : Configuration de navigation

Plusieurs modèles sont disponibles, voici quelques modèles intéressants et leur indication :

- Portable : Indique une faible accélération.
- At sea : Indique une vitesse verticale nulle. (Niveau de la mer supposé)
- Wrist : Recommandé pour les applications portées au poignet, le récepteur filtre les mouvements de bras.
- Airborne <1g : Indique une plage dynamique plus élevée et une accélération verticale plus grande.

D'autres champs de configurations sont modifiables mais il est recommandé de les laisser par défaut.

#### 4.3.1.3 Ports

Le paramètre « UBX-CFG-PRT » permet de configurer les différents ports de communication série du module. On peut choisir parmi tous les ports (UART, SPI, I2C, ...) les différents paramètres de communication (Baudrate, Parité, slave adresse, ...). De plus, il est possible d'indiquer quel protocole utiliser en entrée, ainsi qu'en sortie pour chaque port.

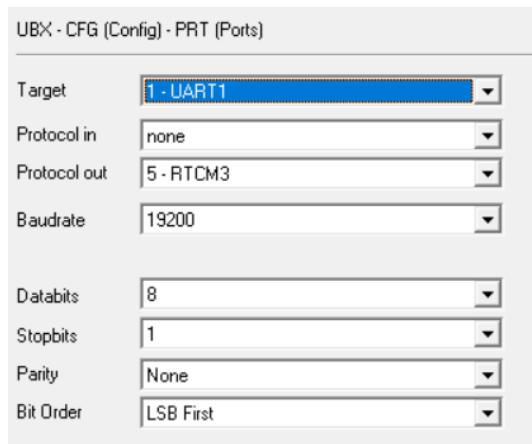


Figure 4-4 : Configuration des ports

Par exemple, ci-dessus, nous pouvons voir les paramètres pour l'uart1 du module sur kit de développement, qui correspondent à la radio UHF. On peut voir que l'on ne souhaite rien recevoir, mais que l'on souhaite envoyer les données de corrections en format RTCM3.

#### 4.3.2 Rover

Les configurations de base sur les modules du kit permettent un bon fonctionnement pour un récepteur GNSS. Cependant, il manque la réception en continu des données de corrections (RTCM).

Pour ajouter la réception des données de corrections, il faut modifier la configuration des ports présentés ci-dessus. Il est donc nécessaire d'indiquer au module par quel port et par quel protocole les corrections arriveront.

D'autres configurations sont disponibles pour un rover, en voici quelques-unes pas encore présentées.

##### 4.3.2.1 DGNSS

Le paramètre « UBX-CFG-DGNSS » permet de configurer le mode de fonctionnement du rover RTK. Il existe deux modes différents RTK :

- RTK fixe : Dans ce mode, le mobile tentera de corriger les ambiguïtés chaque fois que possible.
- RTK float : Dans ce mode, le mobile estimera les ambiguïtés comme flottantes mais ne tentera pas de les corriger.

Le temps nécessaire pour avoir une bonne précision est affecté par la distance entre le rover et la station de référence ainsi que le nombre de satellites visibles par les deux. En général, un mode fixe est toujours recherché car il propose une précision extrêmement pointue. Pour avoir un ordre de grandeur, le mode float permet une précision d'environ 20 cm alors que le mode fixe d'environ 2 cm.

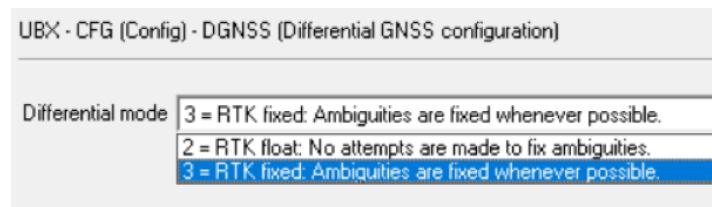


Figure 4-5 : Configuration du DGNSS

#### 4.3.2.2 Fréquence

Le paramètre « UBX-CFG-RATE » permet de configurer la fréquence à laquelle les solutions de navigation (et les mesures dont elles dépendent) sont générées par le récepteur. Ce paramètre est très important, il est intimement lié à la consommation d'énergie et au sens même des résultats. Plus il y a de correctifs requis, plus la puissance du processeur et les ressources de communication sont nécessaires. De ce fait, la fréquence de mise à jour doit être aussi petite que possible mais aussi grande que nécessaire.

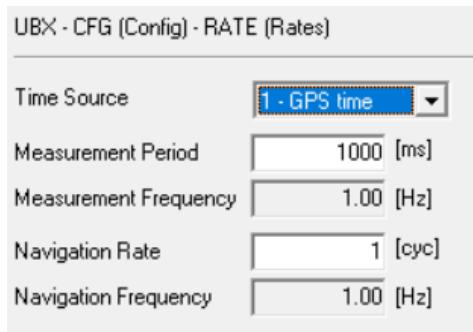


Figure 4-6 : Configuration de la fréquence

Par défaut, la fréquence est réglée à 1Hz, ce qui est suffisant pour la plupart des applications. Le champ « Navigation Rate » permet de régler le taux entre le nombre de mesures et le nombre de solutions de navigation.

D'après le datasheet du NEO-M8P (*Page 6*) [9], la fréquence maximum de navigation pour un système RTK devrait être de 5Hz pour un fonctionnement avec deux constellations (GPS et GLONASS ou GPS et BeiDou) et de 8Hz avec une seule constellation (GPS). Il est tout de même possible de configurer une fréquence de 10Hz.

#### 4.3.3 Station de référence

Une station de référence fonctionne différemment qu'un rover. La première étape d'une station de base est de connaître sa position fixe avec un maximum de précision. Ensuite, elle ne mesurera plus sa position mais générera et enverra des données de corrections.

Dans un premier temps, il s'agit donc de connaître sa position avec un maximum de précision. Il est donc recommandé de modifier la configuration de navigation, vu précédemment, pour indiquer un modèle dynamique stationnaire dans l'optique d'améliorer au maximum la précision.

Il faut aussi modifier la configuration des ports de communication série afin d'indiquer par quel port les données de corrections seront envoyées.

Deux autres paramètres sont encore très importants pour le mode de fonctionnement d'une station de base, les voici :

##### 4.3.3.1 Time Mode 3

Le paramètre « UBX-CFG-TMODE3 » est utile pour la première étape d'une station de référence. Ce paramètre permet de déterminer la position fixe de la station de base de deux différentes méthodes.

La première méthode disponible est le mode « fixed mode » qui permet d'indiquer soi-même la position de la station. L'avantage de ce mode est que le module connaîtra instantanément sa position et pourra sans attendre envoyer des données de corrections. Aucun temps de préparation n'est donc nécessaire. Cependant, il n'est pas évident de connaître une position 3D précise sur le globe.

La deuxième méthode, est le mode « Survey-in ». Ce mode permet de déterminer la position du récepteur fixe en construisant une moyenne pondérée de toutes les solutions de position 3D valides pendant un certain temps. La procédure prend fin uniquement lorsque deux exigences sont remplies. Ces deux exigences sont des paramètres à indiquer lorsque le mode « Survey-in » est choisi.

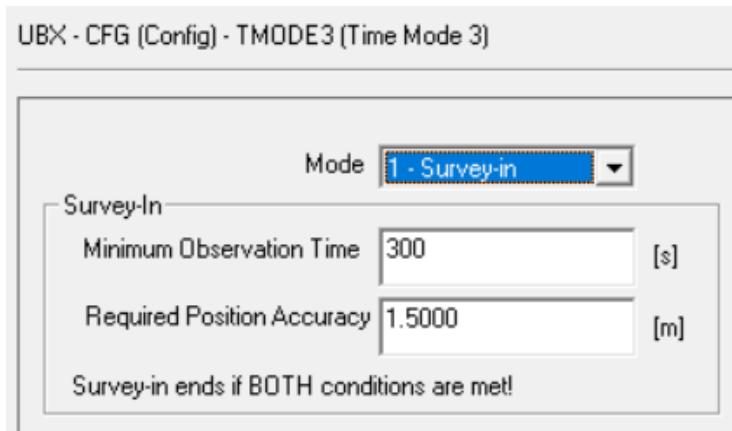


Figure 4-7 : Configuration du mode

Le premier paramètre est le temps minimum d'observation. C'est le temps minimum que le module prendra pour construire la moyenne pondérée de sa position 3D.

Le second paramètre est la précision minimale de la position souhaitée. Dans cet exemple, une précision d'au moins 1,5m est demandée. Il ne faut pas oublier qu'en général la station de référence ne dispose pas d'une correction RTK.

La procédure de recherche de position ne s'arrêtera pas tant que ces deux exigences ne sont pas remplies. Il est donc important de ne pas demander une précision trop grande. Cette méthode est souvent utilisée car elle a l'avantage de ne pas avoir besoin de connaître une position 3D précise sur le globe. Cependant, elle peut prendre plus ou moins de temps de préparation en fonction des exigences demandées. Dans la documentation officielle, il est mentionné qu'un temps de recherche raisonnable peut aller de quelques minutes à un jour pour les exigences de plus haute précision.

Il faut savoir que les erreurs dans la position absolue de la station de référence se répercuteront directement en erreurs sur la position des mobiles. La précision de position de la station de référence doit donc être au moins aussi bonne que la précision de position absolue souhaitée des mobiles. Toutefois, la position relative des rovers restera extrêmement précise.

#### 4.3.3.2 Messages

Finalement, le paramètre « UBX-CFG-MSG » permet de configurer le message que l'on souhaite recevoir ou envoyer et à quelle période. Ce paramétrage doit être en correspondance avec la configuration des ports de communication série. Par exemple, dans la configuration des ports préalablement présentés, le protocole RTCM3 a été sélectionné pour la sortie du port UART1. En voici un exemple ci-dessous.

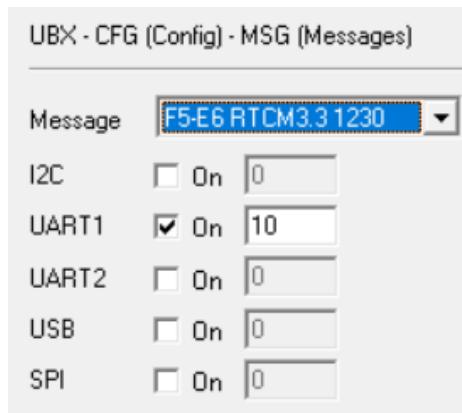


Figure 4-8 : Configuration des messages

Dans ce cas, on souhaite transmettre un message de type RTCM3 via le port UART1. On indique le message ainsi que la période. Une période de 10 signifie que ce message sera reçu une fois toutes les 10 secondes.

Dans le document du guide de configuration du kit de développement « C94-M8P »<sup>[8]</sup>, ils indiquent les fréquences et les types de messages qu'il faut configurer pour une station de référence. Voici une présentation :

- RTCM3 – 1005 - 1Hz : Coordonnée de la station de référence
- RTCM3 – 1077 - 1Hz : Observation de corrections GPS (MSM 7<sup>4</sup>)
- RTCM3 – 1087 - 1Hz : Observation de corrections de GLONASS (MSM 7)
- RTCM3 – 1230 – 0.1Hz : Erreur de la phase L1 et L2 de GLONASS

<sup>4</sup> C'est le message d'observation de type MSM numéro 7 présenté au chapitre « [messages d'observations](#) ».

## 4.4 Méthode d'analyse

Dans le but de valider au mieux la méthode choisie, les tests ont toujours été effectués dans la même zone dégagée et par beau temps de manière stationnaire.

### 4.4.1 Champs de comparaison

Afin de comparer différents modes, il a fallu choisir des paramètres intéressants sur lesquels les comparaisons pourront être faites. La fenêtre « Navigation PVT solution » présente des valeurs sur les résultats de position, vitesse et temps (PVT) y compris des chiffres de précision. Parmi ces différentes valeurs, cinq facteurs seront comparés.

1. **(TACC)** UTC Time Accuracy [ns] : Indique en nanosecondes la précision de l'heure.
2. **(PACC H)** Horizontal accuracy estimate [cm] : Indique la précision de la position horizontale.
3. **(PACC V)** Vertical accuracy estimate [cm] : Indique la précision de la position verticale.
4. SVs Used : Indique le nombre de satellites utilisés.
5. Carrier range status : Indique le mode du RTK (aucun, float, fixe)

Les trois premiers paramètres ont pour but d'indiquer la précision atteinte pendant le test alors que les deux derniers paramètres permettent d'indiquer l'état et la qualité des signaux reçus.

*Remarque : Ces informations sont générées par le module GNSS de u-blox. La véracité de ces informations en dépend.*

### 4.4.2 Les différents modes

Afin d'analyser les différences de performances d'un système avec ou sans la fonctionnalité RTK, trois différents modes ont été testés.

#### 4.4.2.1 Normal (sans RTK)

Le mode « normal » correspond à un récepteur GNSS de haute qualité sans la fonctionnalité RTK. Donc, sans réception de données de corrections de la part d'une station de référence. Ce mode est celui utilisé à présent par l'entreprise. Les paramètres ont été laissés par défaut sur un module du kit de développement.

Les constellations GPS, QZSS et GLONASS sont utilisées, un modèle dynamique « portable » est sélectionné et une fréquence de mesure de 1Hz est configurée. Ces paramètres restent les mêmes pour tous les modes.

#### 4.4.2.2 RTK UHF

Ce mode est celui proposé facilement par le kit de développement. Un module du kit est utilisé comme station de référence et l'autre module en tant que rover. Ces modules communiquent par la radio UHF embarquée sur le kit. L'avantage de cette méthode est que le rover se trouve extrêmement proche de la station de référence. Lors des tests, une distance de moins d'un mètre séparait les antennes GNSS.

#### 4.4.2.3 RTK NTRIP

Finalement, le mode NTRIP correspond à un module du kit utilisé comme rover en recevant des corrections via le réseau internet. Un smartphone a été utilisé pour avoir une connexion internet. Le logiciel u-center intègre un client NTRIP qui permet de s'abonner à une antenne de base et ainsi de recevoir des données de corrections. Ces dernières sont directement envoyées au module GNSS du kit.

L'antenne utilisée participe à un réseau ouvert en France appelé « Centipède », ce service est gratuit. Par chance, une seule antenne se situe en Suisse, dans un village appelé Pailly à environ 10 km du lieu de test.

## 4.5 Comparatif

Voici un tableau qui illustre les données mesurées :

Mode	Distance rover-base	Carrier Range Status	TACC [ns]	PACC H [cm]	PACC V [cm]	SVs Used
Normal	-	-	3	82	150	17
RTK NTRIP	10km	float	2	5.3	7.9	17
RTK NTRIP	10km	fixe	2	<b>1.4</b>	<b>1</b>	17
RTK UHF	1m	fixe	2	<b>1.4</b>	<b>1</b>	18

Tableau 4-1 : Données comparatives des modes

### 4.5.1 Précision

Il est possible de voir que la précision de l'heure (TACC) ne varie pas beaucoup. Une précision de 3 ns pour un mode normal et de 2 ns pour un mode RTK NTRIP ou UHF. Dans tous les cas, le rover est correctement synchronisé avec les satellites.

Pour ce qui est de la précision de la position horizontale (PACC H) et verticale (PACC V) une nette amélioration est visible avec le mode RTK. Un rover en mode normal (sans RTK) a pu atteindre au mieux une précision horizontale de 82 cm et une précision verticale de 150 cm. Ces valeurs prouvent déjà que ce module est d'une excellente précision.

Cependant, un rover en mode RTK présente une précision moyenne horizontale de 5.3 cm et verticale de 7.9 cm. On peut constater que la précision avec le mode RTK est clairement supérieure.

Au mieux, un rover en mode RTK a pu atteindre une précision horizontale de 1.4 cm et une précision verticale de 1 cm. Cette précision unique et extraordinaire est constante lorsque le rover est en mode fixe.

Toutefois, le mode fixe n'a pas pu être observé de façon stable en mode NTRIP contrairement au mode UHF qui était en mode fixe durant la quasi-totalité des tests. Étant donné que les tests ont été effectués au même endroit et au même moment, le mode fixe peut dépendre de la distance entre le rover et l'antenne et/ou des informations transmises<sup>5</sup>.

Pour conclure, la précision est nettement meilleure avec le mode RTK. De plus, le mode RTK peut offrir couramment une précision de position horizontale de 5.3 cm et verticale de 7.9 cm. Mais, dans des conditions parfaites, il est également possible d'atteindre une précision de position de 1.4 cm horizontalement et 1 cm verticalement.

---

<sup>5</sup> Une analyse sur les informations transmises a été réalisée au chapitre « [Analyse approfondie des données de corrections](#) ».

#### 4.5.2 Qualité

Toutes les mesures précédentes ont été réalisées avec un nombre de satellites stables et élevés dans un endroit découvert lors d'une journée avec un ciel dégagé. De plus, le matériel utilisé est celui du kit de développement, donc un matériel de haute qualité. Tous les résultats correspondent à une situation idéale.

Constatation est faite que le mode fixe est moins facilement atteint dans le mode NTRIP que UHF. La distance entre le rover et la station de référence en est peut être la cause. Malgré un test effectué à environ 100 m de la station de référence de Pailly, le mode fixe n'a toujours pas été atteint aisément.

Cela peut être dû au mode de communication ou aux différences de données communiquées. Ou bien même, un mélange de tout ça. Dans tous les cas, une fois qu'un système sera opérationnel, il serait très intéressant de se pencher sur ce problème afin d'avoir le mode fixe le plus souvent possible.

#### 4.5.3 Déviation

Finalement, afin de se représenter visuellement la précision que les différents modes peuvent atteindre, voici les déviations de la position dans les moments les plus stables :

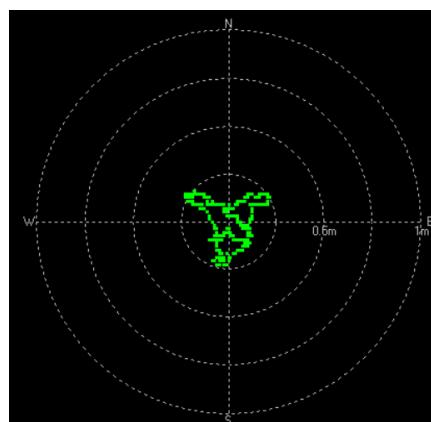


Figure 4-9 : Déviation mode Normal

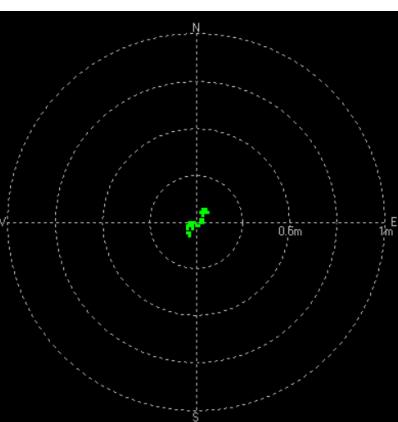


Figure 4-10 : Déviation mode RTK (float)

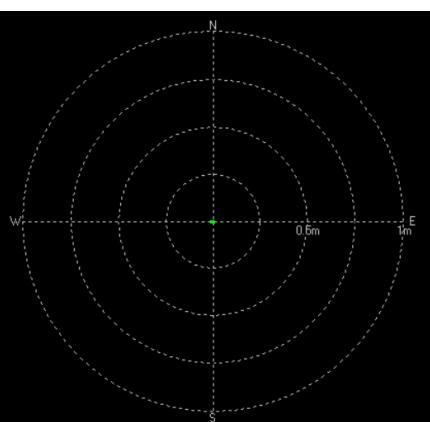


Figure 4-11 : Déviation mode RTK (fixe)

L'échelle est réglée au minimum, 25 cm de rayon pour le premier cercle, 50 cm pour le deuxième cercle, 75 cm pour le troisième cercle et 100 cm pour le dernier cercle.

La différence de précision est claire entre un récepteur sans RTK à gauche, un récepteur avec RTK en mode float au centre et avec RTK en mode fixe à droite.

Il est évident que le mode fixe est généralement recherché, il offre une précision surprenante.

## 5 Implémentation sur le matériel de l'entreprise

Le concept RTK a pu être validé avec une précision allant jusqu'à 1.4 cm horizontale et 1 cm verticale. L'objectif maintenant est d'intégrer la technologie RTK sur le capteur de l'entreprise ASI. Ce chapitre permet d'expliquer plus en détail les différentes étapes réalisées afin d'ajouter un système RTK fonctionnel au capteur de la société.

Ce chapitre s'est déroulé en trois phases :

1. Analyse du logiciel embarqué
2. Implémentation sur le capteur actuel
3. Implémentation sur le capteur final
4. Implémentation d'applications serveur

### 5.1 Analyse du logiciel embarqué

Voici un schéma expliquant de manière succincte le fonctionnement global du programme :

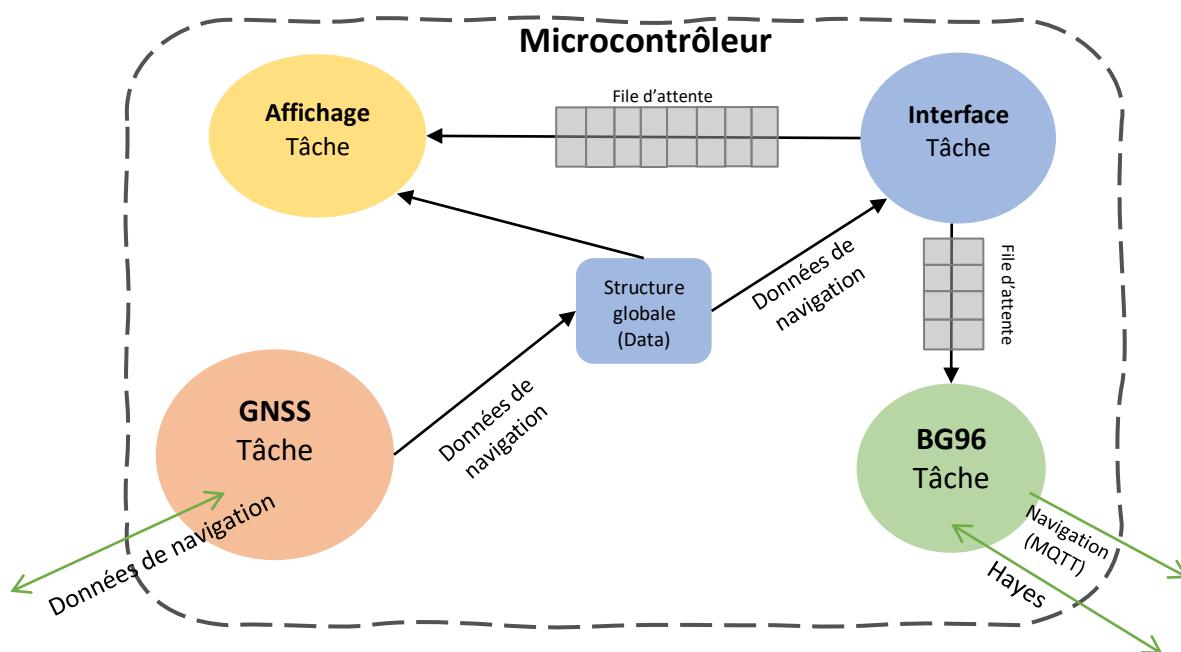


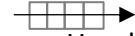
Figure 5-1 : Fonctionnement simplifié du logiciel embarqué

Une **structure globale** est utilisée afin d'enregistrer des informations utiles. Comme des informations de configuration, de version du capteur, des utilisateurs, etc.

Le logiciel est essentiellement composé de 4 tâches :

- **Affichage** : Permet l'affiche de l'interface et de quelques données concernant la qualité du signal GPS.
- **Interface** : Permet la gestion des boutons.
- **GNSS** : Permet la gestion du module GNSS (Configuration du module, réception et conversion des données de navigation).

- **BG96** : Permet la gestion du module LTE (Configuration du module et envoie des données de navigation).



La communication entre les différentes tâches se fait principalement à l'aide de « `xQueueHandle` ». Ceci est un objet proposé par l'API FreeRTOS permettant d'établir une communication entre des tâches en tant que file d'attente.

## 5.2 Implémentation sur le capteur actuel

Le capteur actuel ne dispose pas d'un module GNSS compatible avec la technologie RTK. Néanmoins, tous les autres composants, particulièrement le module LTE, sont conformes au produit final. Grâce auxquels des tests pourraient être réalisés.

### 5.2.1 Présentation du capteur de ASI

Le matériel en question est un capteur utilisé pour aider la gestion des vignobles. L'appareil est un boîtier rectangulaire qui est portable sur le poignet. Il dispose d'un écran LCD et de trois grands boutons :



Figure 5-2 : Présentation du capteur d'ASI

Il permet essentiellement la collecte de données dans les vignes et transmet les données souhaitées à un serveur. Ces données, par exemple une position d'un cep à problème, peuvent être visionnées et analysées à l'aide d'une application. Il est donc possible depuis l'application, de planifier différentes tâches à effectuer qui seront directement visibles par les différentes équipes dans les vignes. Cela permet d'optimiser la gestion quotidienne et stratégique des vignobles.

### 5.2.1.1 Vue d'ensemble

Afin d'avoir une meilleure vue d'ensemble sur le fonctionnement global du système actuel, voici un schéma :

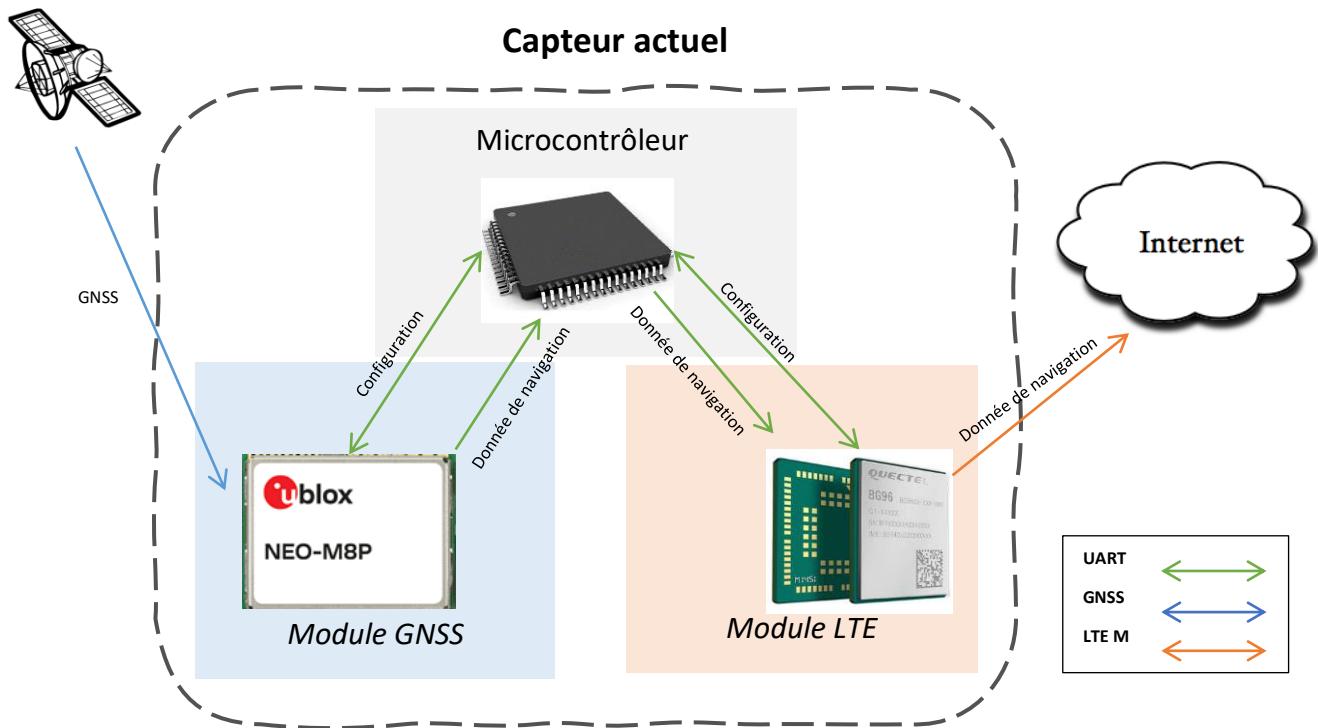


Figure 5-3 : Système global souhaité

En lien avec ce schéma, voici quelques explications :

- Configuration du module LTE *via* **UART**
- Configuration du module GNSS *via* **UART**
- Réception des signaux satellites par le module **GNSS**
- Génération de données de navigation par le module **GNSS**
- Transfert des données de navigation jusqu'au module LTE *via* **UART**
- Envoie des données de navigation à un serveur par le module **LTE**

Le produit comprend essentiellement trois éléments centraux utiles au fonctionnement du système :

- Un microcontrôleur
- Un module GNSS haute précision
- Un module LTE

### 5.2.1.2 Microcontrôleur

Le microcontrôleur utilisé sur le produit est un PIC24 de la société Microchip. Il est utilisé comme unité centrale afin de configurer les deux modules et de gérer les échanges d'informations entre eux.

Plus précisément, il permet de recevoir les données de navigation du module GNSS et de les transmettre au module LTE afin d'envoyer ces données à un serveur, ce qui permet de visualiser les différentes données.

#### 5.2.1.3 Module GNSS haute précision

Le module GNSS utilisé actuellement est le « NEO-M8Q » de u-blox. Il permet de recevoir 3 différents systèmes GNSS simultanément parmi GPS/QZSS, Galileo, GLONASS et BeiDou. Cependant, celui-ci n'est pas compatible avec la technologie RTK.

Il permet de fournir des données de navigation grâce aux signaux reçus des satellites des différents systèmes GNSS.

#### 5.2.1.4 Module LTE

Le module LTE utilisé sur le produit est le « BG96 » de l'entreprise Quectel Wireless. Il permet un débit de données maximal de 375 Kbps en liaison descendante et montante. Il offre des bandes de fréquences mondiales et une consommation d'énergie ultra-faible.

Ce module permet toutes communications externes via internet. Présentement, il est utilisé afin d'envoyer les données de navigation à un serveur.

## 5.2.2 Implémentation sur le capteur

L'objectif est donc de profiter de ce capteur afin de maîtriser le processus de développement et d'apprendre au maximum la gestion du module LTE étant donné qu'il restera inchangé.

### 5.2.2.1 Environnement de développement

Afin de développer et programmer le microcontrôleur du capteur, un environnement de développement (integrated development environment, IDE) est nécessaire. Durant ce projet, l'IDE MPLAB X a été utilisé. Il est gratuit et très répandu pour le développement de logiciel embarqué sur divers microcontrôleurs de Microchip Technology. Il est la dernière version de l'IDE MPLAB développé par cette même entreprise.

Ce logiciel permet plus exactement d'éditer, déboguer et de programmer le PIC24 du module d'ASI.

### 5.2.2.2 Réception des données (rover)

Le premier objectif d'implémentation est de recevoir des données de corrections d'une antenne de référence via le module LTE.

#### 5.2.2.2.1 Conception

Spécifiquement pour un rover, il est essentiel de pouvoir faire une requête HTTP afin de s'abonner à une antenne RTK dans un réseau NTRIP et ainsi de recevoir des données de corrections continuellement.

Une fois que des données de corrections (RTCM) sont reçues continuellement depuis la tâche BG96 (module LTE), il est indispensable de transmettre ces données au module GNSS, via la tâche GNSS. Dès lors, le module GNSS pourra exploiter la technologie RTK.

Afin de comprendre au mieux le travail à apporter au logiciel embarqué, voici un schéma explicatif :

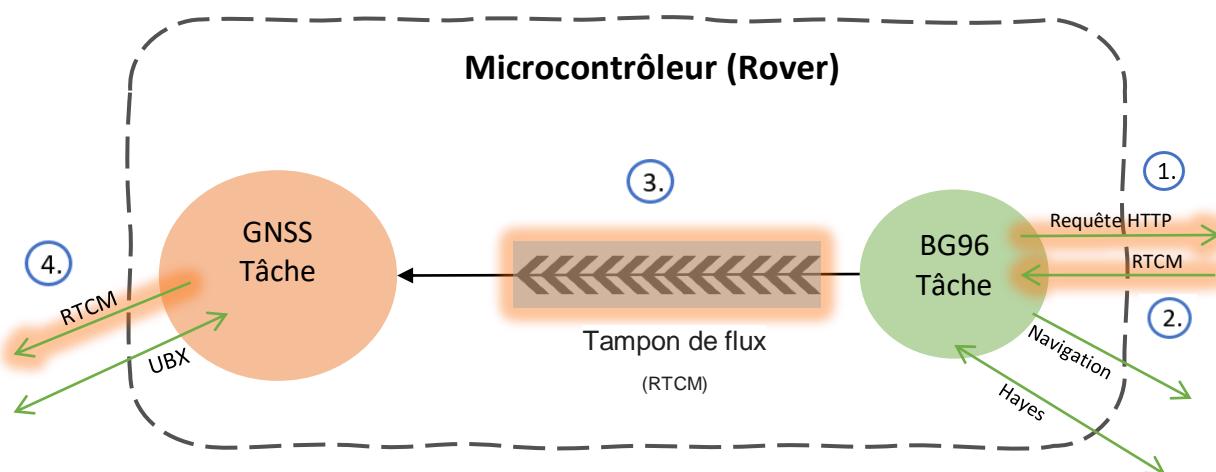


Figure 5-4 : Ajout dans le logiciel (rover)

On peut particulièrement voir en **surbrillance rouge** les parties à ajouter au programme :

- 1. Requête HTTP** : Permet d'envoyer une requêtes HTTP via le module LTE afin de s'abonner à une antenne NTRIP pour recevoir un flux constant de données de corrections RTK.

2. **RTCM** : Réception du flux de données de corrections via le module LTE.
3. **Tampon de flux** : Permet de transmettre les données de corrections de manière unidirectionnelle entre la tâche BG96 et la tâche GNSS. Cet objet, proposé par l'API de FreeRTOS, est assurément optimisé pour une communication en streaming (flux continu).
4. **RTCM** : Après avoir lu les données de corrections dans le tampon de flux, il les envoie au module GNSS.

#### 5.2.2.2 Système de test

Afin de tester ce système, un serveur a été implémenté en C sur un ordinateur Windows pour émuler un serveur NTRIP. La réalisation de ce serveur a été particulièrement utile pour deux raisons, la première est de pouvoir maîtriser un maximum le système complet. La seconde est de gérer la quantité de données transférées afin de limiter la bande passante lors du développement.

Afin de comprendre au mieux le système de test pour la réception de données, voici un schéma permettant une vue globale sur le déroulement d'un système « final » (droite) et un système de « test » (gauche) :

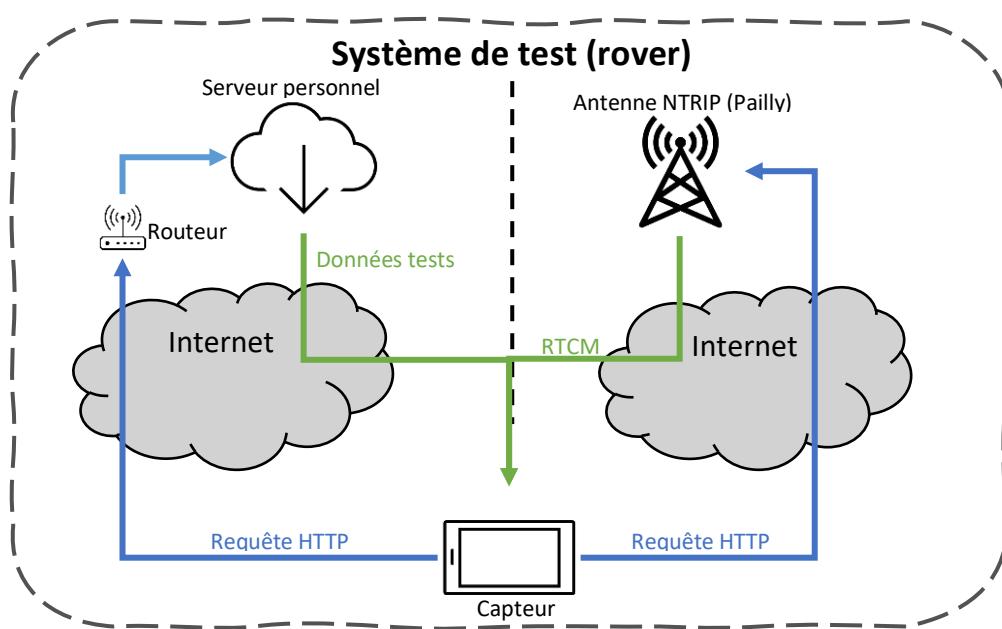


Figure 5-5 : Système de test (rover)

**A droite**, on peut voir le fonctionnement final souhaité, directement avec une antenne NTRIP faisant partie d'un réseau RTK. En effet, la première action à effectuer est une requête HTTP, via le module LTE, avec une adresse, un port et un nom d'antenne NTRIP valide. L'antenne située à Pailly faisant partie du réseau gratuit « Centipède RTK » est utilisée.

Une fois la requête HTTP validée, permettant de s'abonner à l'antenne, le flux constant de données de corrections RTK est reçu directement via le module LTE. Il faut ensuite transmettre ces données de corrections jusqu'au module GNSS.

Le problème avec cette méthode de test est que le débit de données envoyées par l'antenne de Pailly est non négligeable. En effet, un flux constant de plus de 1.5KB/s est reçu. De plus, une grande partie de ces données ne seront pas utiles car le module GNSS qui sera utilisé (NEO-M8P) ne gère pas tous les différents paquets reçus. Comme il a été possible de voir dans le chapitre [Message](#) (p.24), uniquement 4 différents paquets suffisent pour atteindre un mode fixe d'après le guide d'utilisation de u-blox et les tests effectués.

Cependant, une grande variété de paquets est envoyée, par la majorité des antennes NTRIP des réseaux RTK, dans le but d'être compatible avec un maximum de module GNSS. Le capteur final doit tout de même être capable de gérer cette quantité de données pour être compatible avec toutes les antennes NTRIP possibles.

**A gauche**, on peut voir le système mis en place avec un serveur personnel. Le but est de simuler au mieux une antenne NTRIP, tout en limitant drastiquement la quantité de données envoyées. Il a fallu paramétrier un routeur afin d'accepter et de diriger les requêtes entrantes sur le port 2101 (port NTRIP) sur l'ordinateur utilisé. Comme il est essentiel de s'assurer que le logiciel puisse gérer la réelle quantité de données reçues, une fois le système de réception fonctionnel, il fût nécessaire de le tester avec une réelle antenne NTRIP et sa quantité de données significatives.

#### 5.2.2.2.3 Problèmes et corrections

Lors des tests de fonctionnement avec le serveur personnel, tout se passait comme prévu. En effet, après la réception de la requête HTTP depuis le serveur, une petite quantité de données est envoyées constamment au capteur. Ces données étaient réceptionnées par la tâche BG96, puis insérées dans un tampon de flux, puis relues par la tâche GNSS. Des compteurs, affichés sur l'écran LCD, permettent d'avoir une vue d'ensemble du déroulement du programme ainsi qu'un compteur sur la quantité de données reçues par la tâche GNSS.

Une fois que toutes les étapes furent opérationnelles, un test sur l'antenne NTRIP de Pailly a été effectué afin de s'assurer que le débit de données reçues soit correctement géré. Malheureusement, cela ne fût pas le cas. Le logiciel tournait au ralenti et clairement, toutes les données n'étaient pas reçues.

Afin de s'assurer que le débit de données en est la cause, des tests ont été effectués avec le serveur personnel en incrémentant le débit envoyé au capteur. Après une certaine valeur, le même problème a pu être observé. Une démarche de débogage a été effectuée grâce au logiciel « MPLAB X ». Ceci a permis d'avoir une idée de la raison pour laquelle un plus grand débit n'était pas supporté.

En effet, la réception des données via l'uart connecté au module LTE se faisait caractère par caractère. De plus, entre chaque lecture de caractères, plusieurs tests sont effectués afin de faire une analyse syntaxique des différents messages reçus. En outre, plusieurs tâches doivent s'effectuer « simultanément » sur un même cœur.

Afin de faire un système le plus efficace possible, deux modifications ont été effectuées :

1. Une lecture de l'uart par paquet et non par caractère.
2. Changé le mode d'accès aux données via la connexion TCP/IP du module LTE.

Premièrement, il est bien connu qu'une lecture de paquet est plus optimisée qu'une lecture par caractère. Comme mentionné précédemment, une lecture de l'uart par paquet a été choisie.

Deuxièmement, il existe différents modes d'accès aux données via un socket<sup>6</sup> avec le module LTE « BG96 ». L'objectif était d'arriver à une solution très efficace rapidement. Un socket en mode « Transparent access mode » a été utilisé. Ce mode permet d'utiliser l'uart exclusivement pour un socket souhaité. Toutes les données reçues par le module LTE seront directement envoyées sur l'uart. Le défaut de cette méthode est qu'aucune autre opération ne peut être effectuée sur le module LTE durant ce mode.

#### 5.2.2.2.4 Résultats

Néanmoins, après une nouvelle implémentation et des nouveaux tests, le fonctionnement du système de réception s'est correctement déroulé. Il était enfin possible de s'abonner à une antenne NTRIP, de recevoir son grand débit de données de corrections RTK ainsi que de transmettre ces données jusqu'à la tâche GNSS.

---

<sup>6</sup> Un socket désigne une connexion réseau. Peut être traduit par « connecteur réseau ».

### 5.2.2.3 Envoi des données (station de référence)

Maintenant qu'une réception constante de données est possible, un envoi constant est souhaité. En effet, en tant que station de référence, le cheminement inverse est nécessaire. Des données de corrections RTK seront générées par le module GNSS en mode station de référence. Ces données devront être envoyées via une connexion TCP à un serveur.

#### 5.2.2.3.1 Conception

Pour un capteur en mode station de référence, il est essentiel de pouvoir transmettre les données de corrections à un serveur afin que celui-ci puisse transmettre ces données aux différents rovers à proximité.

Une fois que des données de corrections seront générées continuellement depuis le module GNSS, il est indispensable de transmettre ces données à un serveur.

Pour ce faire, voici le travail à apporter au logiciel embarqué :

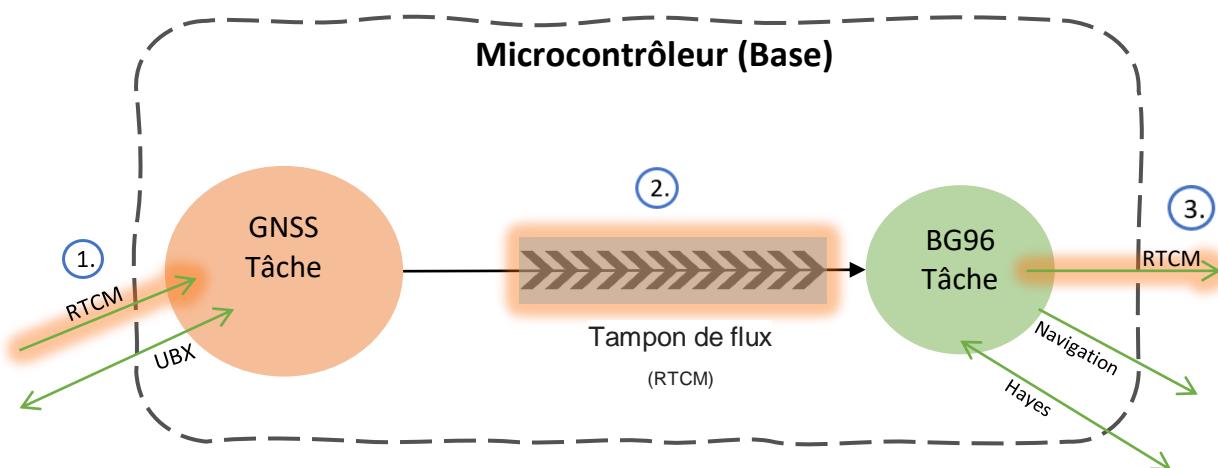


Figure 5-6 : Ajout dans le logiciel (base)

Ce schéma ressemble de près au schéma précédemment présenté. Cependant, pour un capteur en mode base (station de référence) le sens des données est inversé. On peut particulièrement voir en **surbrillance rouge** les parties à ajouter au programme :

1. Le module GNSS (NEO-M8P), en mode station de référence, génère et envoie sur l'uart des données de corrections RTK.
2. La tâche GNSS réceptionne continuellement ces données de corrections et les envoie dans le tampon de flux.
3. La tâche BG96 lit les données dans le tampon de flux et les transmet, via l'uart, au module LTE qui les envoie au serveur préconfiguré.

#### 5.2.2.3.2 Système de test

Afin de tester le système complet, le serveur personnel précédemment présenté a été modifié. De plus, une application « client TCP » a été utilisée sur un téléphone afin de simuler facilement un rover souhaitant recevoir des données du serveur. L'objectif est de pouvoir faire la chaîne complète entre le

capteur en mode station de base qui envoie continuellement des données à un serveur et un client les recevant via ce serveur.

Afin de mieux comprendre la chaîne complète de test, voici un schéma explicatif :

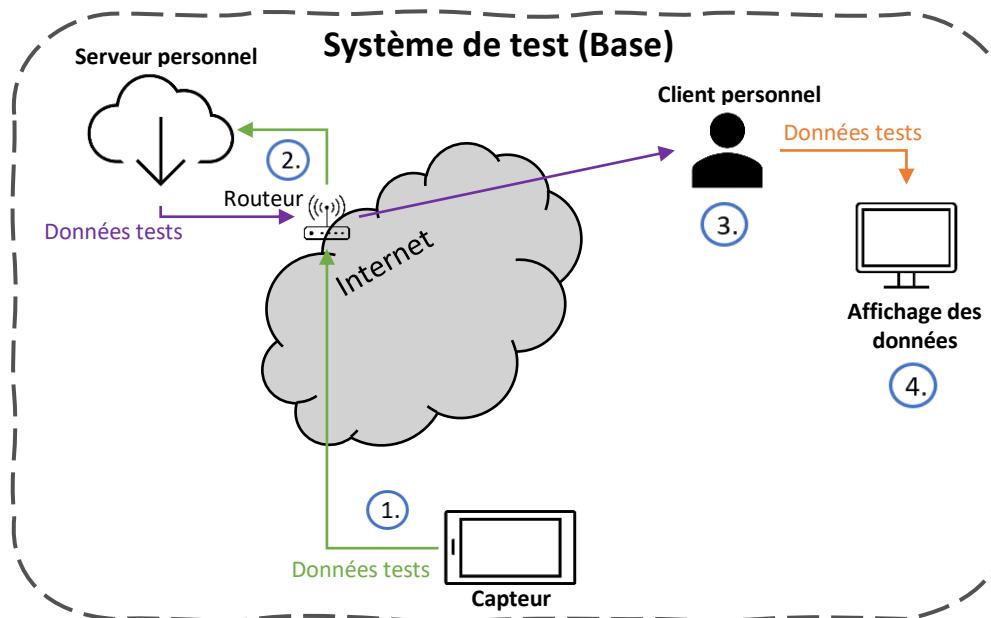


Figure 5-7 : Système de test (base)

Ci-dessous, des explications plus détaillées afin de comprendre chaque étape du système :

1. **Capteur** : C'est le **capteur en mode base** (station de référence). Le capteur **va envoyer des données de « test »** au serveur personnel. Ces données de « test » sont définies dans le programme. Elles permettent de simuler les données de corrections RTK qui seront générées par le module GNSS.
2. **Serveur personnel** : Le serveur réceptionnera toutes les données reçues sur la première connexion faite. De plus, différents affichages peuvent être faits afin d'avoir un bon aperçu du système en fonctionnement. Ce serveur pourra encore recevoir une nouvelle connexion d'un client afin de **transmettre toutes les données reçues** en continu à celui-ci.
3. **Client personnel** : Une application sur le smartphone permet de se connecter sur le serveur personnel comme client TCP. Ce dernier permet de simuler un **capteur en mode rover** souhaitant recevoir des données de corrections RTK. Une fois la connexion effectuée, le client reçoit continuellement toutes les données du serveur.
4. **Affichage** : À chaque réception de données, le client **affiche sur un écran les données reçues**.

Cette structure permet d'avoir un bon contrôle sur chaque entité nécessaire au fonctionnement d'un système RTK. De plus, il permet de vérifier la totalité des données transmises entre les différents sous-systèmes.

L'infrastructure représentant la partie serveur peut être construite de différente manière. Soit par un serveur personnel qui s'occupera d'envoyer les données aux différents rovers personnels, un peu comme la structure de test précédente. Soit par un réseau NTRIP existant, en y ajoutant notre propre station de référence, comme « Centipède RTK », « RTK2go », etc. Ou encore, d'autres infrastructures imaginables.

#### 5.2.2.3.3 Résultats

Finalement, chaque étape d'un système RTK a été testée. Le capteur de l'entreprise peut envoyer en continu des données à un serveur. Le serveur reçoit correctement toutes les données et les affiche sur un terminal afin d'en avoir la certitude. En se connectant au serveur avec un smartphone, il est possible de recevoir toutes ces données en les affichant sur un écran en continu.

#### 5.2.2.3.4 Modifications

L'objectif a bien été atteint. Cependant, afin de faire la similitude avec un fonctionnement final correct, deux principales modifications devront être apportées.

Premièrement, l'application client TCP exécutée depuis le smartphone (**client personnel**) devra être remplacée par un capteur en mode rover de l'entreprise. Celui-ci devra transmettre les données reçues du serveur au module GNSS afin de profiter de cette technologie.

Deuxièmement, le capteur envoyant des données de test (**Capteur**) devra être remplacé par un capteur en mode base (station de référence). Au lieu d'envoyer des données de test, il devra envoyer des données de corrections générées par le module GNSS.

Afin d'avancer dans ce sens du projet, il est nécessaire de posséder la nouvelle version du capteur (capteur final) comportant le module GNSS compatible avec la technologie RTK. Pour ce faire, l'entreprise ASI a travaillé sur la réalisation de cette nouvelle version du capteur.

## 5.3 Implémentation sur le capteur final

Ce chapitre va présenter les implémentations apportées au nouveau capteur d'ASI.

### 5.3.1 Configuration du module GNSS

La première chose à faire est de configurer correctement le nouveau module GNSS.

#### 5.3.1.1 Méthodologie

Afin de configurer le module GNSS, il faut envoyer au module, via l'uart, des commandes propriétaires à u-blox. Ces commandes indiquent le type de configuration et les valeurs des différents champs à modifier.

Voici un exemple de commande envoyée au module :

0X B5 62 06 00 14 00 01 00 00 00 D0 08 00 00 00 C2 01 00 23 00 01 00 00 00 00 DA 52

Dans l'ordre, de gauche à droite, la signification des blocs :

1. En-tête fixe d'un message u-blox.
2. Type de configuration. (0x0600 = Configuration des ports)
3. Longueur du message de configuration. (0x14 = 20 bytes)
4. Valeurs des différents champs.
5. Somme de contrôle. (CRC)

Toutes les configurations possibles et les champs configurables sont détaillés dans le document officiel « u-blox 8 / u-blox M8 Receiver description ». Ce document a été utilisé pour construire toutes les configurations apportées au module GNSS.

#### 5.3.1.2 Configurations apportées

Afin d'éviter les redondances, il vous est suggéré de vous rapporter au chapitre « [Configurations](#) » présenté lors de la configuration du kit de développement. Cette section présente toutes les configurations utiles apportées au nouveau module GNSS.

### 5.3.2 Fonctionnement global d'un rover

Voici un diagramme qui explique succinctement le fonctionnement global du logiciel embarqué en mode rover :

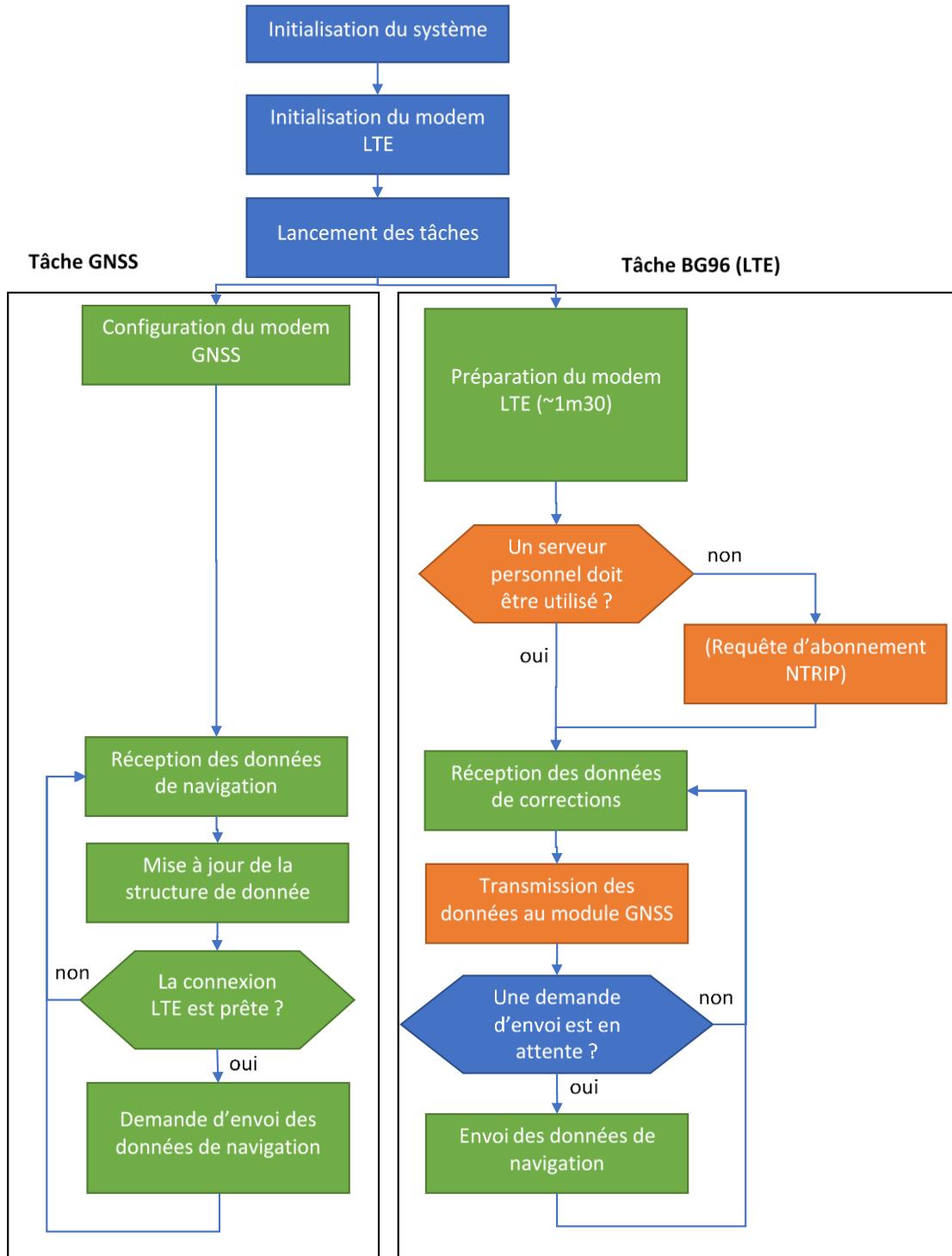


Figure 5-8 : Diagramme logiciel d'un rover

Les parties en bleu représentent des fonctionnalités du logiciel déjà existantes. Les parties en vert symbolisent des fonctionnalités existantes avec des modifications apportées. Les parties en orange modélisent les fonctionnalités implémentées durant ce projet.

Dans la tâche BG96, après la préparation du module LTE, un test est effectué afin de savoir si un serveur personnel est utilisé. Par défaut, c'est le cas. Cependant, le logiciel permet de fonctionner en mode NRTK (Réseau RTK). Si l'on souhaite que le rover s'abonne directement à une antenne appartenant à un réseau RTK, utilisant le protocole NTRIP, cela est possible.

### 5.3.3 Fonctionnement global d'une station de référence

Voici un diagramme qui explique succinctement le fonctionnement global du logiciel embarqué en mode station de référence :

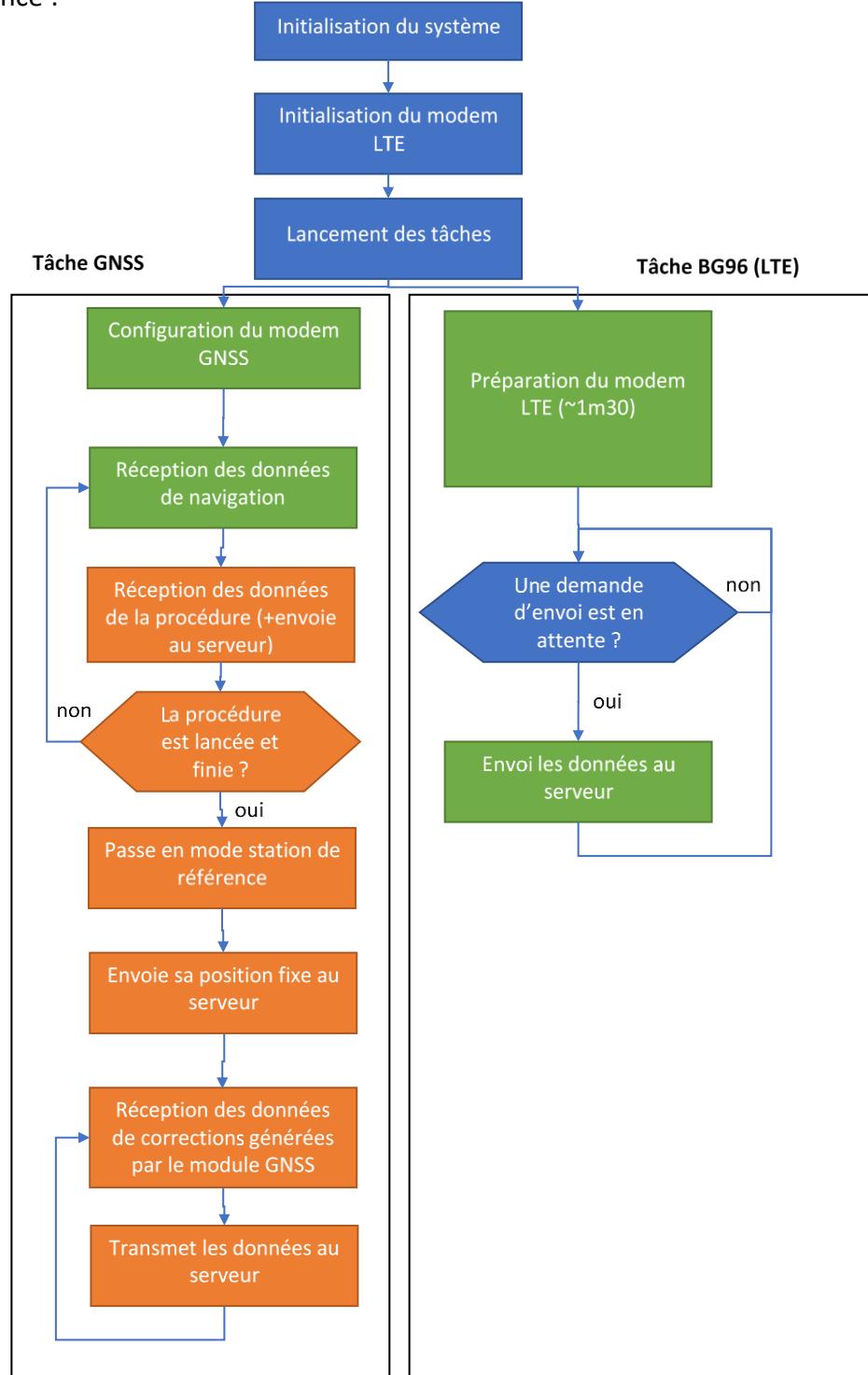


Figure 5-9 : Diagramme logiciel d'une station de référence

Comme le diagramme précédent, les parties en bleu représentent des fonctionnalités du logiciel déjà existantes. Les parties en vert symbolisent des fonctionnalités existantes avec des modifications apportées. Les parties en orange modélisent les fonctionnalités implémentées durant ce projet.

Le fonctionnement est un peu plus complexe qu'un rover car une procédure de détermination de la position fixe de la station de base est requise. Celle-ci est déclenchée en appuyant sur le bouton « SW3 » (bouton de droite).

Une fois la procédure activée, les informations la concernant sont envoyées au serveur afin d'avoir un suivi à distance. Plus précisément, quatre valeurs sont envoyées :

1. Le temps minimal de durée de la procédure définie dans la configuration. (**Condition 1**)
2. Le temps actuel de la procédure en cours.
3. La précision minimale de la position 3D définie dans la configuration. (**Condition 2**)
4. La précision actuelle de la position 3D.

Une fois les deux conditions remplies, la procédure est terminée. Le capteur peut maintenant passer en mode station de référence et générer des données de corrections.

Afin de passer en mode station de référence, quelques configurations sont envoyées au modules GNSS :

1. Les différents messages reçus sont désactivés (Navigation PVT, Procédure de positionnement, etc.)
2. Les différents messages de corrections sont activés.  
*Par défaut, les messages utilisés sont ceux conseillés dans le datasheet officiel [8] présenté au chapitre « [Messages](#) ». Cependant, il est possible avec un simple paramètre d'indiquer l'utilisation des messages de type MSM 4 afin de générer moins de données.*

Dès lors, toutes les données reçues du module GNSS (données de corrections RTCM3) sont envoyées au serveur.

## 5.4 Implémentation des applications serveur

Durant les chapitres précédents, le concept « serveur » a beaucoup été mentionné afin de faire la passerelle entre la station de référence et les rovers. Ce chapitre permet d'expliquer plus précisément l'utilité et le fonctionnement de cette entité.

### 5.4.1 Utilité de l'application serveur

Voici un schéma qui permet d'expliquer la fonction principale de l'application serveur :

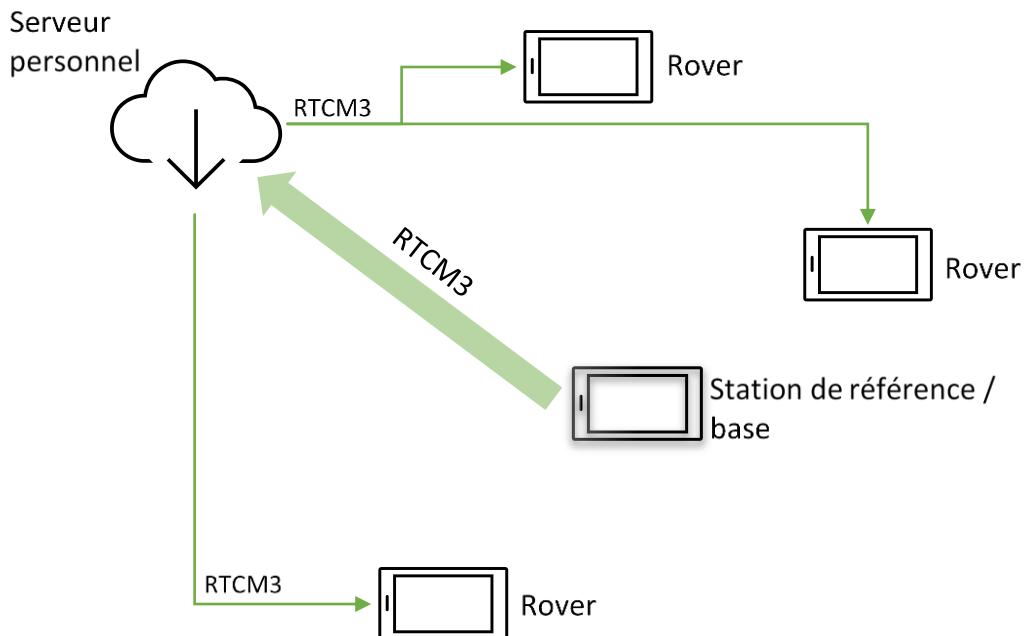


Figure 5-10 : Fonction de l'application serveur

L'application serveur permet de distribuer les données de corrections d'une station de base aux rovers situés aux alentours. Elle permet surtout d'alléger la charge du capteur en mode station de référence afin de ne pas devoir gérer toutes les connexions avec les rovers.

De plus et surtout, l'application serveur va permettre une grande flexibilité au système. Voici quelques exemples :

- Enregistrement des données reçues des rovers.
- Enregistrement des données de corrections.
- Traitement en temps réel des positions des rovers et de la station de base.
- Commande des rovers et de la station de base à distance.

Toutefois, une autre application serveur a été implémentée. Elle permet d'utiliser directement une station de référence appartenant à un réseau NTRIP. Ci-dessous, le fonctionnement de la deuxième application serveur :

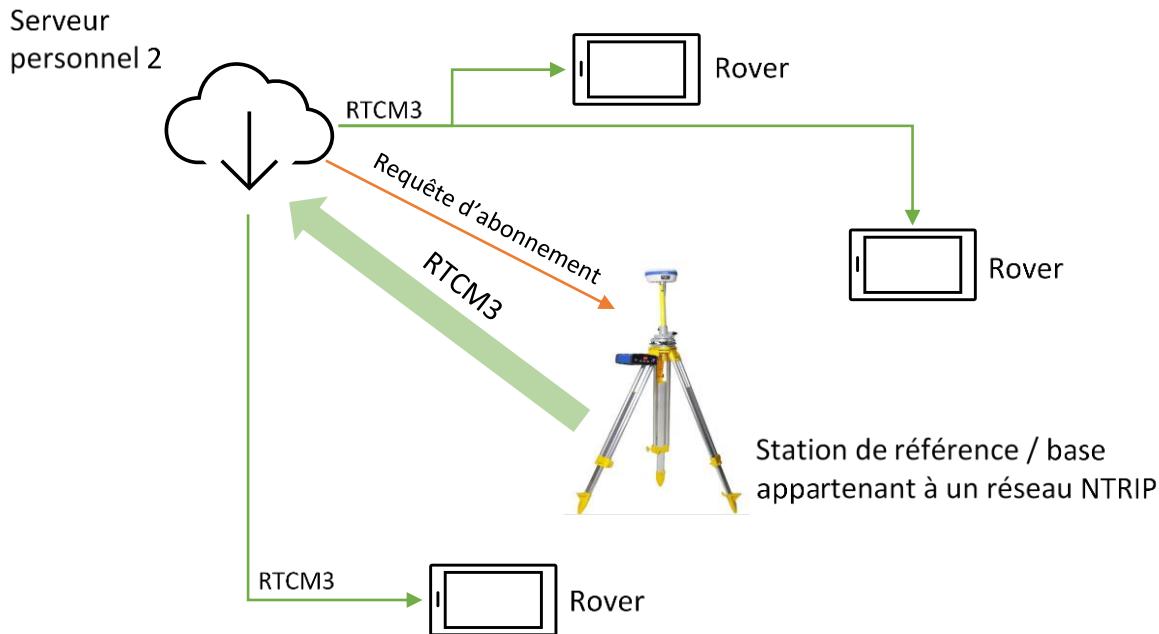


Figure 5-11 : Fonction de l'application serveur 2

Dans un premier temps, l'application serveur va faire une requête HTTP dans la norme NTRIP afin de s'abonner aux données de corrections d'une antenne d'un réseau RTK. Si les informations d'identification<sup>7</sup> de l'antenne sont correctes, le serveur recevra continuellement les données de corrections de l'antenne. Ces données pourront être transmises aux différents rovers connectés au serveur.

Cette deuxième application serveur permet de profiter pleinement des réseaux RTK ainsi d'éviter de mettre en place une station de référence. Cette deuxième application pourrait être utilisée à deux conditions :

1. Les informations d'identification de l'antenne doivent être connus.
2. Les rovers doivent se situer aux alentours de l'antenne.

Afin de profiter à tout moment de ces applications serveur, la société ASI a mis à disposition un accès à distance à une VM<sup>8</sup> Windows sur leur serveur. Il a fallu créer les exécutables des applications avec les librairies linker statiquement<sup>9</sup>. Puis, les déplacer et exécuter depuis le serveur mis à disposition.

Quelques règles de pare-feu<sup>10</sup> ont dû être ajoutées afin d'autoriser les requêtes entrantes et sortantes du serveur d'ASI.

<sup>7</sup> Afin de s'abonner à une antenne il est essentiel d'avoir quelques informations : l'adresse IP, le port, son nom et si besoin, le nom d'utilisateur et le mot de passe.

<sup>8</sup> Une machine virtuelle (VM) permet d'émuler un appareil informatique ainsi que de les isoler pour des raisons de sécurité.

<sup>9</sup> Le fichier objet de l'application et les bibliothèques sont liés dans le même fichier exécutable.

<sup>10</sup> Un pare-feu permet de faire respecter la politique de sécurité d'un réseau.

#### 5.4.2 Fonctionnement de l'application serveur

Cette section explique le déroulement de la première application serveur. Voici un diagramme concis du « *main* » :

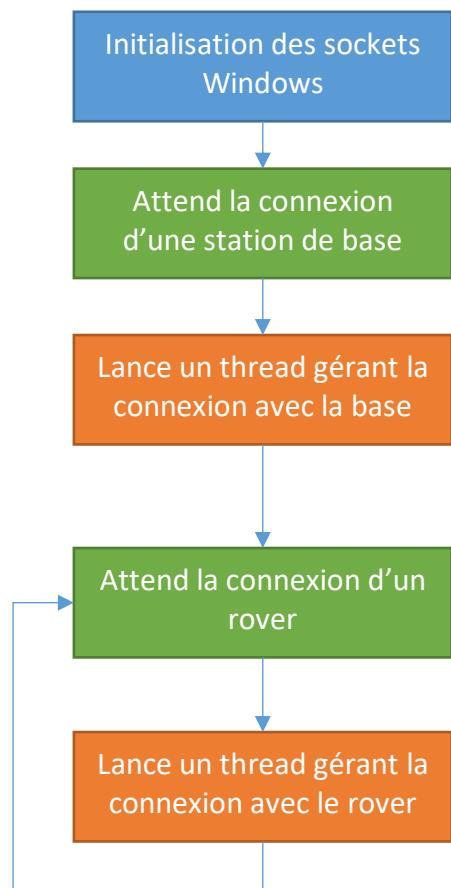


Figure 5-12 : Diagramme du main de l'application serveur

Pour des raisons de compréhension, l'application a été implémentée par plusieurs petites unités. De plus, des threads ont été utilisés afin d'exploiter au mieux les ressources matérielles du serveur.

Il y a deux types de thread :

1. Le thread qui gère la connexion avec la station de référence.
2. Le thread qui gère une connexion avec un rover.

Le fonctionnement de ces deux types de thread est expliqué ci-dessous.

#### 5.4.2.1 Thread gérant la station de référence

Le diagramme suivant modélise le déroulement du thread qui gère la connexion avec la station de base :

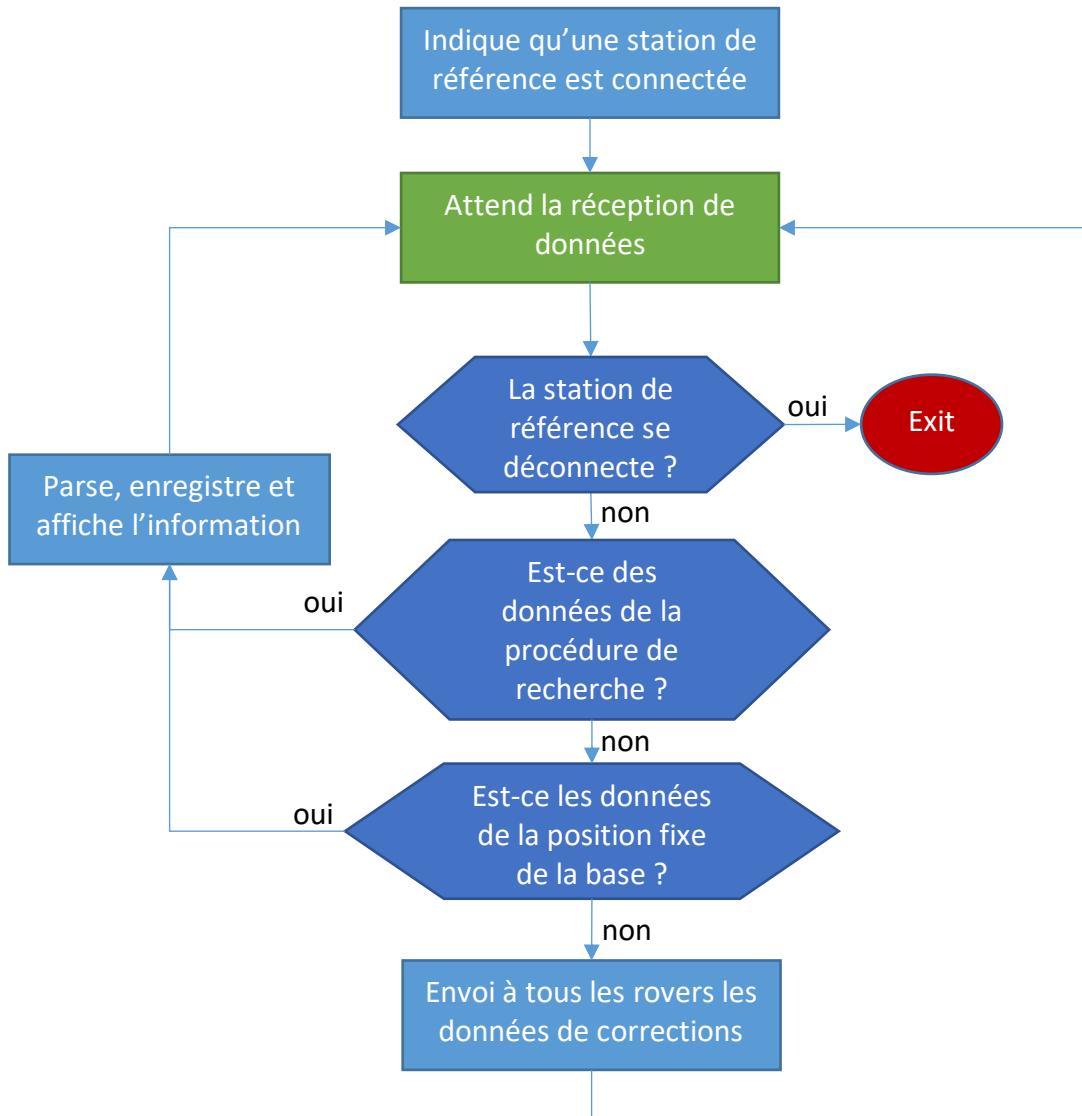


Figure 5-13 : Diagramme du thread (station de base)

Ce thread attend constamment la réception de données venant de la station de référence. Les données entrantes peuvent représenter différentes informations :

- Des informations signifiant la déconnexion de la station de base
- Des informations relatifs à la procédure de recherche de positionnement en cours
- Des informations concernant la position fixe de la station de référence (Reçus une fois que la procédure est terminée)
- Des données de corrections (RTCM3). Dans ce cas, il envoie ces données de corrections à tous les rovers actuellement connectés.

#### 5.4.2.2 Thread gérant les rovers

Le diagramme suivant modélise le déroulement d'un thread qui gère une connexion avec un rover :

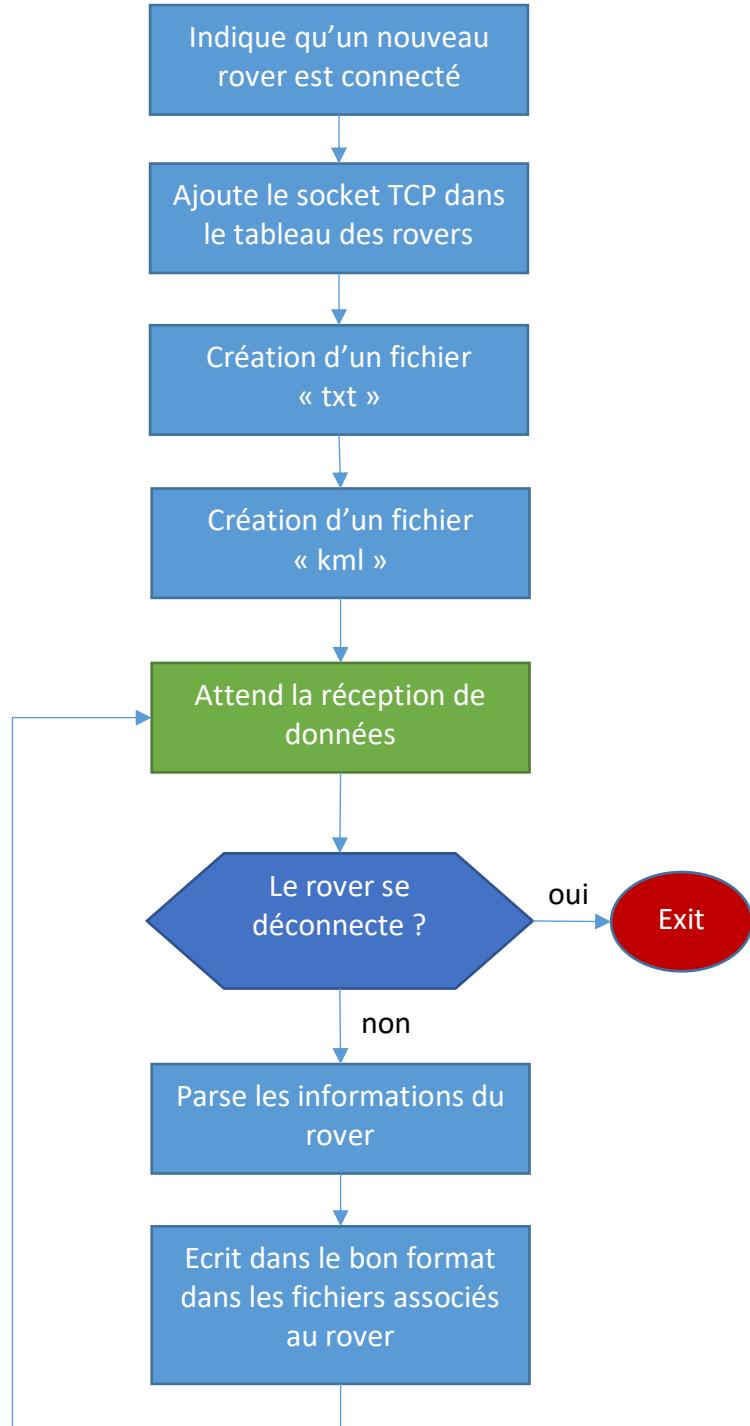


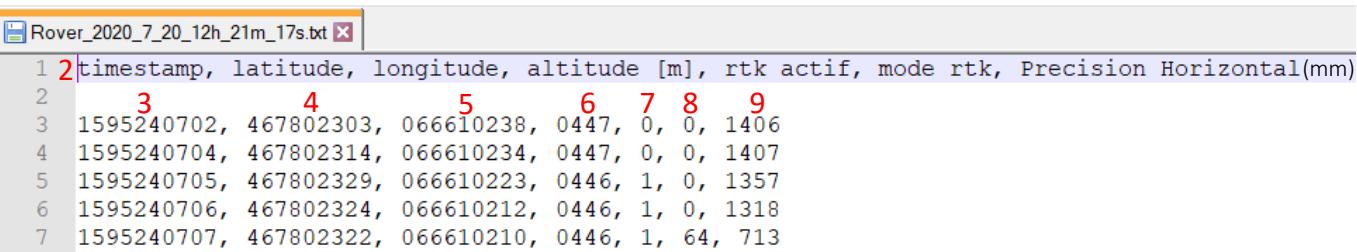
Figure 5-14 : Diagramme du thread (rover)

Lors de la connexion d'un rover, le socket correspondant est placé dans un tableau contenant tous les sockets des rovers. Ce tableau est utilisé afin d'envoyer à tous les rovers les données de corrections. Ce tableau est aussi mis à jour lorsqu'un rover se déconnecte du serveur.

Ensuite, deux types de fichiers sont créés avec comme nom, la date et l'heure de la connexion du rover :

1. Un fichier « .txt » avec les logs de toutes les données reçues du rover.
2. Un fichier « .kml » avec les coordonnées (longitude, latitude, altitude) du rover. Ce type de fichier (KML<sup>11</sup>) permet d'afficher les données de localisation en utilisant, par exemple, l'application « Google Earth ».

Pour exemple, voici le haut d'un fichier « .txt » :

1 

```
1 2|timestamp, latitude, longitude, altitude [m], rtk actif, mode rtk, Precision Horizontal(mm)
2   3      4      5      6      7      8      9
3 1595240702, 467802303, 066610238, 0447, 0, 0, 1406
4 1595240704, 467802314, 066610234, 0447, 0, 0, 1407
5 1595240705, 467802329, 066610223, 0446, 1, 0, 1357
6 1595240706, 467802324, 066610212, 0446, 1, 0, 1318
7 1595240707, 467802322, 066610210, 0446, 1, 64, 713
```

Figure 5-15 : Exemple du fichier TXT

Quelques explications en correspondance avec les numéros en rouge :

1. Le titre du fichier avec la date et l'heure de la connexion du rover.
2. L'en-tête du fichier décrivant la signification des logs.
- 3-9. Les logs du rover relatif à l'en-tête.

Voici un petit exemple, avec uniquement trois coordonnées, d'un fichier KML :

---

<sup>11</sup> Keyhole Markup Language (KML) est un langage destiné à la gestion de l'affichage de données géospatiales.



The screenshot shows a Notepad window with the file 'Rover\_2020\_7\_20\_12h\_21m\_17s.kml'. The code is annotated with curly braces and labels:

- A brace on the right side groups the first few lines as 'En-tête'.
- A brace on the right side groups the coordinate log section as 'Logs des coordonnées'.
- A brace on the right side groups the final closing tags as 'Fin du fichier'.

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
  <Document>
    <name>Rovers KML</name>
    <open>1</open>
    <description>Les itinéraires des rovers</description>
    <Style id="yellowLineGreenPoly">
      <LineStyle>
        <color>7f00ffff</color>
        <width>4</width>
      </LineStyle>
      <PolyStyle>
        <color>7f00ff00</color>
      </PolyStyle>
    </Style>
    <Folder>
      <name>Rovers</name>
      <Placemark>
        <name>Rover_2</name>
        <visibility>1</visibility>
        <styleUrl>#yellowLineGreenPoly</styleUrl>
        <LineString>
          <extrude>1</extrude>
          <tessellate>1</tessellate>
          <altitudeMode>relative</altitudeMode>
          <coordinates>
            6.6610238,46.7802303,447
            6.6610234,46.7802314,447
            6.6610223,46.7802329,446
          </coordinates>
        </LineString>
      </Placemark>
    </Folder>
  </Document>
</kml>
```

Figure 5-16 : Exemple du fichier KML

Comme pour le fichier texte, le nom du fichier contient la date et l'heure de la connexion du rover. L'en-tête du fichier est écrit dans le fichier lors de sa création. Ensuite, à chaque réception de nouvelle donnée du rover, les coordonnées sont écrites dans le bon format à la suite de l'en-tête. Finalement, lors de la déconnexion du rover, la fin du fichier est écrite.

En ouvrant ce type de fichier avec l'application « Google Earth », il est possible de voir l'itinéraire du rover. Voici un exemple :



Figure 5-17 : Exemple d'itinéraire (KML)

Une fois ces deux types de fichiers créés avec l'en-tête correspondante, le thread attendra continuellement des nouvelles données venant du rover. A chaque nouvelle réception, les données sont décodées et écrites dans les deux types de fichiers dans le format souhaité.

#### 5.4.3 Fonctionnement de la deuxième application serveur

Cette section va expliquer le déroulement de la seconde application serveur. Celle utilisant une station de référence déjà existante dans un réseau NTRIP. Voici un diagramme concis du « *main* » :

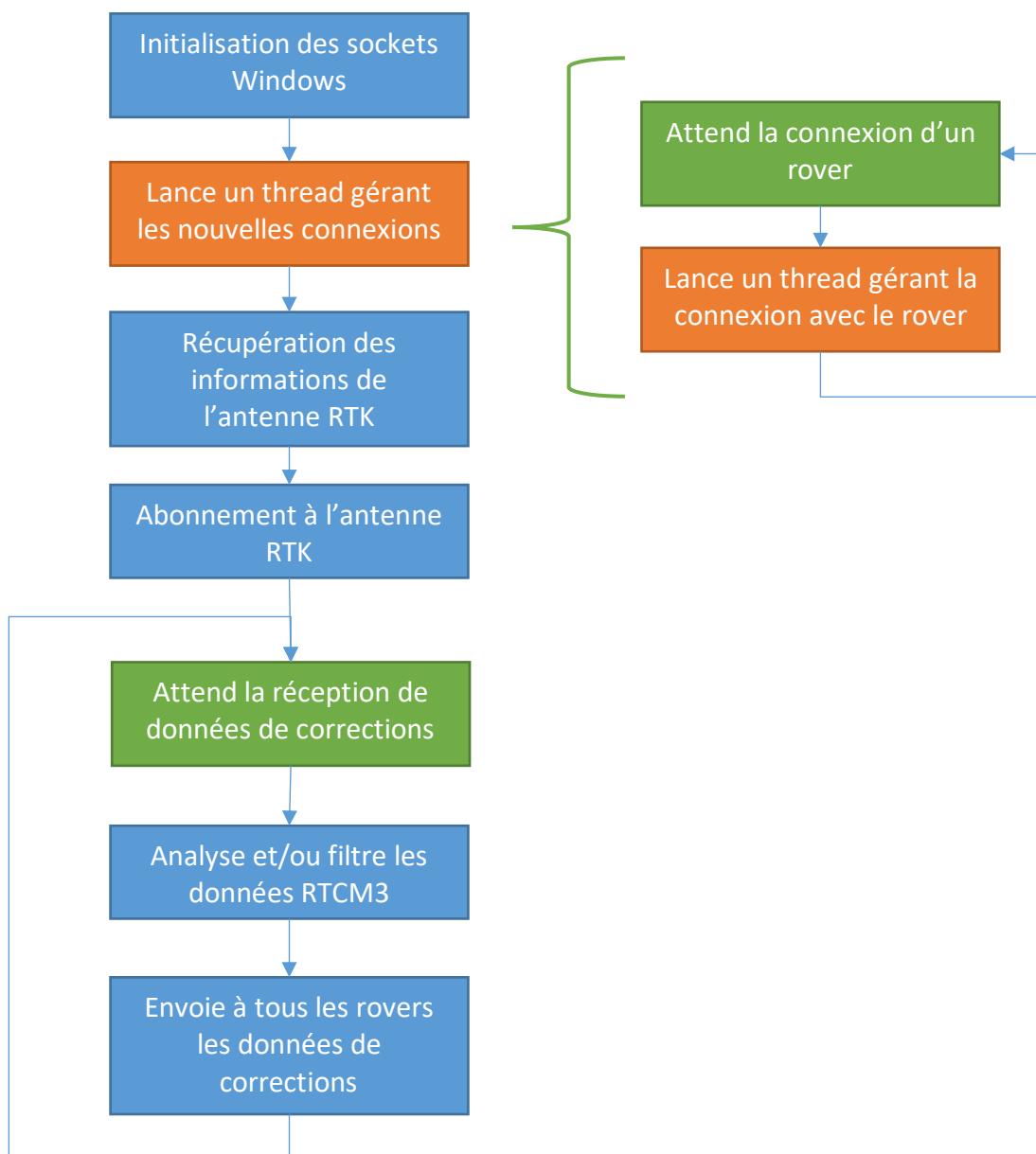


Figure 5-18 : Diagramme du main de la seconde application serveur

Après l'initialisation, cette application va lancer un thread permettant de gérer toutes les nouvelles connexions. A chaque nouvelle connexion, le même thread présenté dans le chapitre précédent « [Thread gérant les rovers](#) » sera lancé pour s'occuper du nouveau rover.

Ensuite, les informations d'identifications de l'antenne à laquelle s'abonner sont récupérées. Par défaut, l'application possède déjà les informations concernant une antenne. Cependant, il est possible via les arguments fournis en lançant l'application de redéfinir ces informations.

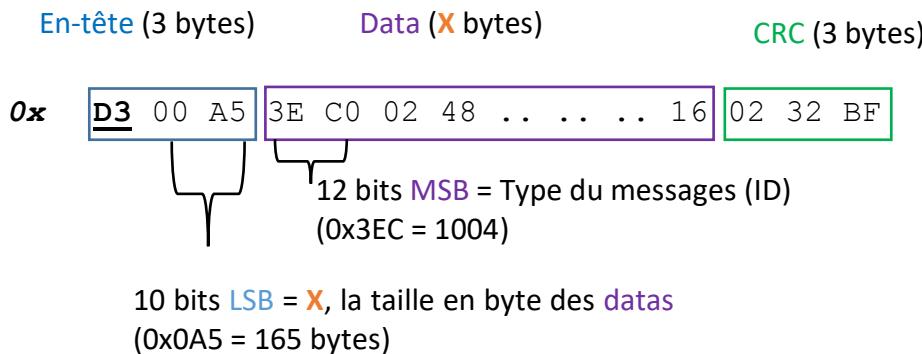
Une fois les informations prêtes, la requête HTTP est créée et envoyée pour s'abonner à la station de base définie.

A chaque réception de données de corrections, celles-ci sont envoyées à tous les rovers connectés au serveur. De plus, une fonctionnalité permet d'analyser et/ou filtrer ces données avant de les envoyer aux rovers.

#### 5.4.3.1 Analyse et filtrage des données RTCM3

Cette fonctionnalité a été implémentée afin d'alléger un maximum le nombre de byte transmis aux rovers. Un PIC24 étant limité quant aux ressources matérielles, il n'est pas aisné de recevoir une grande quantité de byte simultanément. Particulièremment, si ces données sont inutiles au bon fonctionnement du mode RTK.

Pour analyser et filtrer les messages RTCM3, quelques connaissances sur le format du protocole sont requises. Voici un exemple de la construction d'un message de type RTCM3 :



Tous les paquets commencent par un en-tête de 3 bytes dont le premier vaut constamment 0xD3. Les 10 bits LSB<sup>12</sup> de cet en-tête signifient la taille en byte des données du paquet.

Les 12 premiers bits de la donnée (MSB<sup>13</sup>) représentent le type du message RTCM3.

Finalement, tous les paquets finissent par une somme de contrôle (CRC) de 3 bytes.

Grâce à ces informations, il est possible de découper le flux de données de corrections par des paquets RTCM3. Il est aussi possible de connaitre le type et la taille de chacun de ces paquets.

La fonction d'analyse permet d'afficher régulièrement le décompte des différents types de paquets reçus ainsi que la taille du dernier paquet.

Voici un exemple de l'affichage avec l'antenne RTK de la HEIG-VD :

<sup>12</sup> LSB (least significant bit) représente les bits de poids faible.

<sup>13</sup> MSB (most significant bit) représente les bits de poids fort.

Type	Count	Size
1004	- 104	- 186
1012	- 104	- 112
1006	- 7	- 27
1008	- 7	- 32
1033	- 7	- 60
1230	- 7	- 18
1013	- 4	- 15
1045	- 1	- 68

Figure 5-19 : Analyse des paquets RTCM3 (HEIG-VD)

Cette analyse permet de savoir que la station de base de la HEIG-VD transmet les messages de 8 différents types RTCM3. Il est aussi possible d'apprendre les différentes fréquences d'envois des messages. Par exemple, les messages de types 1004<sup>14</sup> et 1012 sont envoyés nettement plus fréquemment que les autres types.

Finalement, la fonctionnalité de filtrage se comporte de la même façon, mais en plus, rempli un buffer uniquement avec les messages de types souhaités.

<sup>14</sup> Une analyse plus approfondie sur les différents types de messages RTCM3 est disponible dans le chapitre suivant « [Analyse approfondie des données de corrections \(RTCM3\)](#) ».

#### 5.4.4 Interactions du système complet

Ce chapitre permet d'assembler les trois entités (station de référence, application serveur et rover) implémentées durant ce projet. L'objectif est d'expliquer les interactions entre ces entités à travers des digrammes de séquences.

Afin de partir sur les mêmes bases, voici quelques explications sur les diagrammes de séquences qui suivront :

- Les entités sont répertoriées en haut du diagramme, avec une ligne pointillée tirée verticalement.
- Le démarrage d'une entité est symbolisé par un rectangle chevauchant sa ligne pointillée.
- L'interaction entre deux entités est indiquée par une flèche annotée.
- La dimension verticale du diagramme représente le temps.

Les diagrammes représenteront les interactions entre les entités des deux différents systèmes qui ont pu être mis en place durant ce projet :

1. Système RTK
2. Système NRTK

Le système RTK représente le fonctionnement avec uniquement le matériel de l'entreprise ASI. Il est constitué d'un capteur d'ASI en mode station de référence, d'une application serveur et d'un capteur d'ASI en mode rover.

Le système NRTK représente le fonctionnement avec une station de référence appartenant à un réseau RTK existant. Il est constitué de l'antenne RTK du réseau, d'une application serveur et d'un capteur d'ASI en mode rover.

#### 5.4.4.1 Système RTK

Ci-dessous, le diagramme de séquences correspondant au système RTK :

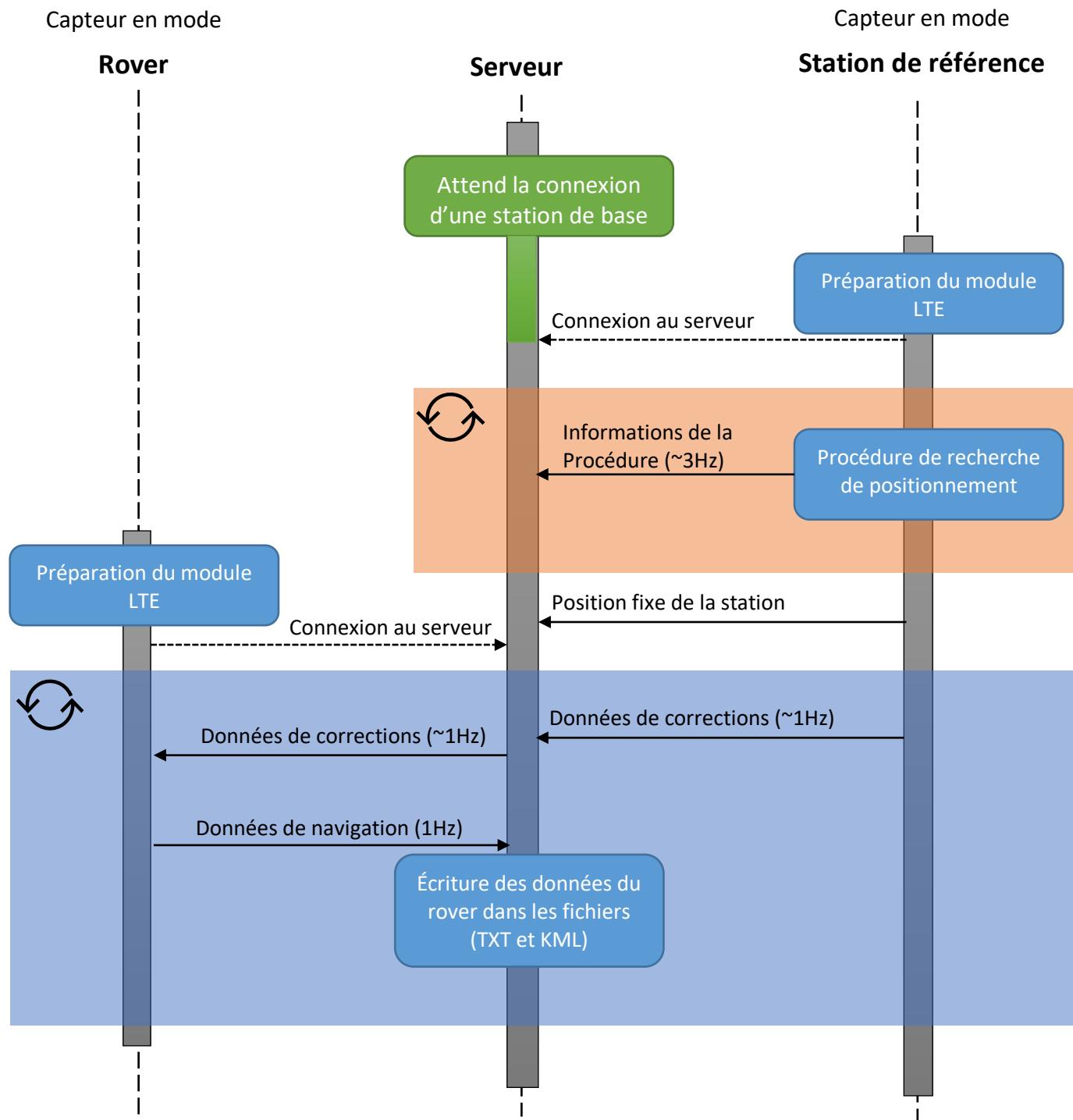


Figure 5-20 : Diagramme de séquence du système RTK

Ci-après quelques explications complémentaires :

En premier lieu, l'application serveur présentée durant le chapitre « [Fonctionnement de l'application serveur](#) » est lancée.

Ultérieurement, un capteur en mode station de référence est allumé. Une fois le module LTE prêt (~1m30), une connexion avec l'application sur le serveur est établie.

Lorsque le bouton souhaité est pressé, la procédure de recherche de positionnement de la station commence. Durant la procédure, les informations relatives à celle-ci sont envoyées au serveur.

Dès que la procédure est terminée, la position fixe de la station est envoyée au serveur.

Une fois la station de base opérationnelle, elle envoie continuellement ses données de corrections au serveur.

D'autre part, un capteur en mode rover est démarré. Une fois le module LTE prêt, une connexion avec l'application sur le serveur est établie.

Le rover va constamment envoyer ses données de navigation à l'application serveur. Dans le cas où une station de référence est en train de transmettre des données de corrections au serveur, le rover les reçoit aussitôt.

Au final, le rover peut profiter de la technologie RTK en appliquant les données de corrections de la station de référence reçues via l'application serveur. De plus, le serveur a accès aux données de navigation des rovers connectés à l'application, ainsi qu'à la position fixe de la station de référence.

#### 5.4.4.2 Système NRTK

Ci-dessous, le diagramme de séquences correspondant au système NRTK :

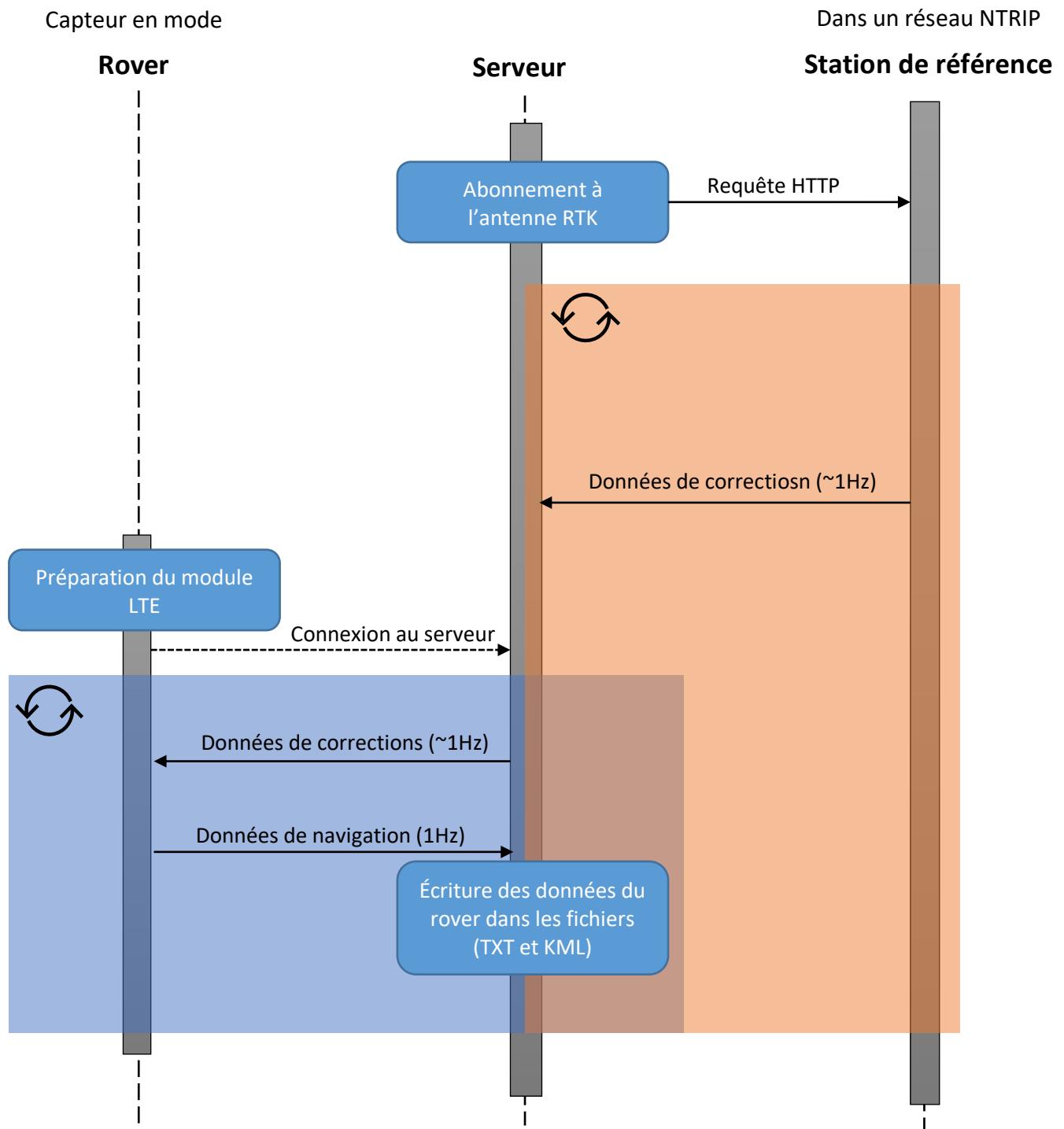


Figure 5-21 : Diagramme de séquence du système NRTK

Ci-après quelques explications complémentaires :

En premier lieu, les informations d'identification d'une antenne RTK faisant partie d'un réseau NTRIP doivent être connues.

Ensuite, la deuxième application serveur présentée durant le chapitre « [Fonctionnement de la deuxième application serveur](#) » est lancée.

L'application va commencer par s'abonner à l'antenne désignée.

Les données de corrections générées par l'antenne sont reçues continuellement sur l'application serveur.

Ultérieurement, un capteur d'ASI en mode rover est allumé. Les interactions entre un rover et l'application serveur reste identique au diagramme du chapitre précédent. L'unique différence est que les données de corrections reçues du serveur parviennent d'une antenne déjà existante.

Pour conclure ce chapitre sur l'implémentation, deux différents systèmes fonctionnels ont pu être mis en place. Il est possible de profiter de la technologie RTK via un réseau existant (NRTK) ainsi qu'avec une station de référence personnelle.

## 6 Tests et analyses

Maintenant qu'il est possible de profiter d'un système RTK fonctionnel, il est intéressant de savoir quels résultats offre ce nouveau système. Quelle différence apporte la technologie RTK comparée à un système simple. Quel paramètre peut influencer ces résultats. Ces questions et d'autres sont discutées durant ce chapitre.

### 6.1 Tests des rovers

Le but de ces tests est de comparer les résultats de précision horizontale et verticale des différents systèmes avec un capteur d'ASI en mode rover. Pour ces tests, trois systèmes sont comparés :

1. Sans RTK : Système simple sans l'utilisation de données de corrections.
2. Système NRTK : Système utilisant les données de corrections d'une antenne existante.  
(Pailly ou HEIG-VD)
3. Système RTK : Système utilisant les données de corrections d'une station de référence réalisée par un capteur d'ASI.

Deux différentes antennes existantes ont été utilisées pour effectuer les tests de ce projet :

1. HEIG-VD : Une antenne « privée » appartenant à l'école HEIG-VD située sur le toit de cette école à Yverdon-les-Bains.
2. Pailly : Une antenne « public » appartenant au réseau Centipède [9].

Voici des données récoltées représentant bien les différences de précision :

Antenne de référence RTK	Distance rover-base	Mode RTK	SVs Used	PACC H [mm]	PACC V [mm]
aucune	-	-	16	928	1586
Pailly	10 Km	float	16	68	103
HEIG-VD	200m	float	16	38	46
Capteur ASI	1 km	float	16	24	41
HEIG-VD	200m	fixe	16	14	10
Capteur ASI	1 km	fixe	16	14	10

Tableau 6-1 : Données comparatives des précisions

*Remarque : Les tests ont été effectués de manière fixe. Les valeurs rapportées sont fournies par le récepteur GNSS (NEO-M8P).*

La précision utilisant la technologie RTK est nettement supérieure à un système simple. Elle varie entre 103 mm avec l'antenne de Pailly et 10 mm en mode fixe. Alors que la précision d'un système simple est limitée à environ 900 mm.

Le mode fixe a pu être atteint avec une station de base réalisée par un capteur ainsi qu'avec l'antenne de l'HEIG-VD. Cependant, la station de base de Pailly n'a pas permis d'atteindre ce mode.

## 6.2 Analyse approfondie des données de corrections (RTCM3)

L'objectif est de comprendre plus précisément quels types de messages sont transmis par les différentes antennes et surtout de savoir quels messages sont indispensables ou superflus.

Voici un premier tableau indiquant quels types de messages sont envoyés par chaque antenne :

ID	Type	Pailly	HEIG	Capteur ASI
1004	Observation GPS (RTK)	✓	✓	
1005	Station de référence	✓		✓
1006	Station de référence	✓	✓	
1008	Station de référence		✓	
1012	Observation GLONASS (RTK)	✓	✓	
1013	(Station de référence)		✓	
1019	Éphémérides	✓		
1020	Éphémérides	✓		
1033	Station de référence		✓	
1042	Éphémérides	✓		
1045	Éphémérides	✓		
1046	Éphémérides	✓		
1074	Observation GPS (MSM4)			✓
1077	Observation GPS (MSM7)	✓		(✓)
1084	Observation GLONASS (MSM4)			✓
1087	Observation GLONASS (MSM7)	✓		(✓)
1097	Observation Galileo (MSM7)	✓		
1124	Observation BeiDou (MSM4)			
1127	Observation BeiDou (MSM7)	✓		
1230	Observation GLONASS		✓	✓
<b>Débit (byte / seconde) :</b>		~ 1500	~ 350	~ 300

Légende :

- ✗ ID pas accepté en input du module GNSS
- ✗ ID accepté en input du module GNSS
- ✓ Message transmis par l'antenne

Tableau 6-2 : Données de l'analyse des données de corrections (RTCM3)

Les IDs en rouge ne sont pas supportés pour le module GNSS contrairement aux IDs en vert. Tous les messages de types en rouge sont inutiles pour le capteur d'ASI, ils peuvent donc être filtrés.

En prenant en compte uniquement les messages supportés par le module GNSS, l'antenne de la HEIG-VD envoie que quatre messages différents (1004, 1006, 1012, 1230). Ceux-ci permettent le bon fonctionnement du mode RTK et un mode fixe. L'antenne de l'HEIG-VD envoie des messages de corrections de type RTK.

La station réalisée par le capteur permet un aussi bon fonctionnement avec le même nombre de messages mais avec deux types différents (1006, 1074, 1084, 1230). Le capteur envoie des messages de corrections de type MSM.

Cependant, l'antenne de Pailly envoie un mélange des types RTK et MSM. Ceci est déconseillé par le datasheet du récepteur GNSS. Cette incohérence est sûrement la cause d'un fonctionnement moins remarquable avec cette antenne.

### 6.3 Analyse de la précision de la position absolue

L'objectif est d'analyser la réelle précision de la position absolue du capteur sur le plan horizontal et vertical.

Pour procéder à cette analyse, le capteur a été placé à cinq reprises, sur quatre points connus précisément. Chaque mesure a été faite à environ 24h d'intervalle afin de maintenir au mieux la disposition des satellites de la constellation GPS.

Les positions des quatre points sont connues au millimètre, ils ont été fournis par l'équipe de géomètres de la HEIG-VD. Ces points se trouvent dans un champ situé à côté de cette école. Plus précisément, ils se situent chacun au centre d'une vis traversant un pilonne en béton, donc au total, il y a quatre pilonnes bien distincts.



Figure 6-1 : Placement d'un point de mesure

Pour chaque point et chaque mesure, le capteur a été placé de manière identique. Le capteur est placé en direction du nord et le centre du module GNSS sur le centre de la vis.



Figure 6-2 : Placement du capteur sur un point de mesure

Pour chaque mesure, le capteur a été déposé sans perturbation pendant plusieurs minutes afin de s'assurer d'avoir les meilleurs résultats possibles.

### 6.3.1 Précision horizontale

Les données des mesures ont été analysées et traitées afin d'obtenir la différence horizontale entre le point de référence et le point mesuré. Vous trouverez ci-dessous, un tableau contenant ces informations (Tableau 6-3). Voici quelques explications sur la signification des colonnes :

- *Nom des points* : Les noms des points de références utilisés.
- *Déférence horizontale* : La différence horizontale entre le point de référence et le point mesuré. Les N°1 à N°5 correspondent aux différents jours des mesures.
- *Moyenne des différences* : La moyenne des différences horizontales sur les 5 jours.
- *Précision attendue* : Précision horizontale indiquée par le module GNSS.

Nom des points	Différence horizontale [mm]					Moyenne des différences [mm]	Précision attendue [mm]
	N°1	N°2	N°3	N°4	N°5		
Point 1 (6170)	35	43	21	45	43	37.4	14
Point 2 (6190)	39	42	19	31	5	27.2	14
Point 3 (6160)	39	28	39	18	38	32.4	14
Point 4 (6180)	31	35	37	33	24	32	14

Tableau 6-3 : Données des mesures de la précision absolue horizontale

Les moyennes des différences varient entre 27.2 mm et 37.3 mm alors que le module nous indique une précision de 14 mm. Cela peut s'expliquer par deux raisons :

1. L'erreur de la position 3D de la station de référence lors de sa mise en place est directement transmise aux rovers.

2. Le point de référence du capteur n'est pas le centre du module GNSS mais environ le centre de l'antenne GNSS. Ces deux points sont espacés d'environ 1.42 mm.

Pour ces deux raisons, il n'est pas évident de déterminer la précision de la position absolue. Malgré cela, la précision reste étonnante.

Il est donc possible d'atteindre, comme attendu dans le cahier des charges, une précision inférieure à 8 cm. De plus, cela prouve que la mise en place de la station de référence de la HEIG-VD a été faite de manière précise.

Il est important de noter que la précision de la position absolue dépend grandement de la précision de la position fixe obtenue lors de l'installation de la station de référence.

### 6.3.2 Précision verticale

Les données des mesures ont été analysées et traitées afin d'obtenir la différence verticale entre le point de référence et le point mesuré. Le tableau contenant ces informations est ci-dessous (Tableau 6-4). Les explications sur les colonnes sont les mêmes que pour le dernier tableau. La colonne supplémentaire « *Ecart max* » représente la différence maximum en mm entre deux points de mesure.

Nom des points	Différence verticale [mm]					Moyenne des différences [mm]	Précision attendue [mm]	Ecart-type [mm]	Ecart max [mm]
	N°1	N°2	N°3	N°4	N°5				
Point 1 (6170)	2455	2460	2448	2407	2437	2441.4	10	21.08	53.00
Point 2 (6190)	2500	2484	2476	2467	2455	2476.4	10	17.04	45.00
Point 3 (6160)	2446	2476	2484	2493	2461	2472	10	18.69	47.00
Point 4 (6180)	2459	2473	2469	2477	2491	2473.8	10	11.71	32.00

Tableau 6-4 : Données des mesures de la précision verticale

Curieusement, la différence horizontale absolue est très grande, environ 2,5m. Quelques théories pouvant expliquer cette différence ont pu être apportées mais elles n'ont pas pu être confirmées :

1. La hauteur de la station de référence est erronée.
2. La hauteur du point fourni n'est pas celle de la vis.
3. Le récepteur GNSS utilise une autre référence d'altitude que les géomètres de la HIEG-VD.

Malgré cet offset surprenant, les écarts-types et les écarts max prouvent que le capteur reste cohérent avec ses mesures jour après jour.

La précision attendue de 10 mm indiquée par le récepteur GNSS est cependant inexacte. Au mieux, une différence maximum de 32 mm a été observée entre deux mesures du même point. Au pire, une différence de moins de 6 cm a été aperçue.

Comme pour la précision absolue horizontale, la précision absolue verticale n'est pas évidente à déterminer. De plus, la précision attendue indiquée est de 10 mm alors que la précision relative varie réellement entre 30 et 60 mm.

Toutefois, comme chaque mesure a été prise à des jours différents, d'autres paramètres<sup>15</sup> perturbateurs peuvent justifier cette erreur de précision.

Pour conclure, la réelle précision relative reste satisfaisante et acceptable par rapport au cahier des charges.

<sup>15</sup> La composition de l'ionosphère, la composition de la troposphère, l'emplacement et la répartitions des satellites

## 6.4 Analyse de la précision relative horizontale

Comme appris récemment durant le chapitre « [Précision horizontale](#) », l'erreur de la précision absolue peut être plus élevée qu'attendu. Cela a pu être expliqué car l'erreur de la précision sur la position fixe de la station de base, lors de l'installation de celle-ci, est transmise au rover. Cependant, la précision relative ne devrait pas être perturbée.

C'est pourquoi, l'objectif est d'analyser la précision relative. La différence avec la précision absolue est que la mesure de distance est faite entre deux points de mesures du même test et non plus entre un point de mesure et un point de référence connu.

Pour effectuer cette analyse, le capteur a été placé à différents points le long d'une règle. Plus précisément, avec un espacement de 10 cm entre chaque point de mesure. Deux paramètres ont été observés :

1. La distance entre le point relevé au 0 de la règle et les autres points mesurés.
2. La dispersion des points.

Le premier paramètre permet d'indiquer si l'espacement de 10 cm est précis et si l'erreur de précision attendue est respectée.

Le second permet de s'assurer que les points relevés sont alignés.

Cette analyse a été effectuée sur trois modes différents :

1. *ASI float* : La station de référence est réalisée avec un capteur d'ASI et le mode RTK float.
2. *ASI fixe* : La station de référence est réalisée avec un capteur d'ASI et le mode RTK fixe.
3. *HEIG fixe* : La station de référence est celle de la HEIG-VD et le mode RTK fixe.

### 6.4.1 *ASI float*

Le tableau contenant les données relatives à cette analyse est ci-dessous (Tableau 6-5). Voici quelques explications sur la signification des colonnes :

- *Distance réelle* : La distance réelle entre le capteur et le point 0 de la règle.
- *Long.* : La coordonnée longitude du point relevé.
- *Lat.* : La coordonnée latitude du point relevé.
- *Distance mesurée* : La distance mesurée entre le point relevé et le point 0 de la règle.
- *Erreur* : La différence entre la distance réelle et mesurée.
- *HA* : Précision horizontale indiquée par le module GNSS.

<b>Distance réelle [mm]</b>	<b>Long. [°]</b>	<b>Lat [°]</b>	<b>Distance mesurée [mm]</b>	<b>Erreur [mm]</b>	<b>HA [mm]</b>
0	6.6610211	46.7802354	0	-	53
100	6.6610219	46.7802358	76	24	51
200	6.6610238	46.7802359	214	-14	46
300	6.6610256	46.7802357	345	-45	47

Tableau 6-5 : Données de la précision relative (*ASI float*)

Pour tous les points mesurés, l'erreur réelle est inférieure à la précision prévue par le module GNSS.

Voici ci-dessous la représentation de la dispersion des différents points mesurés :

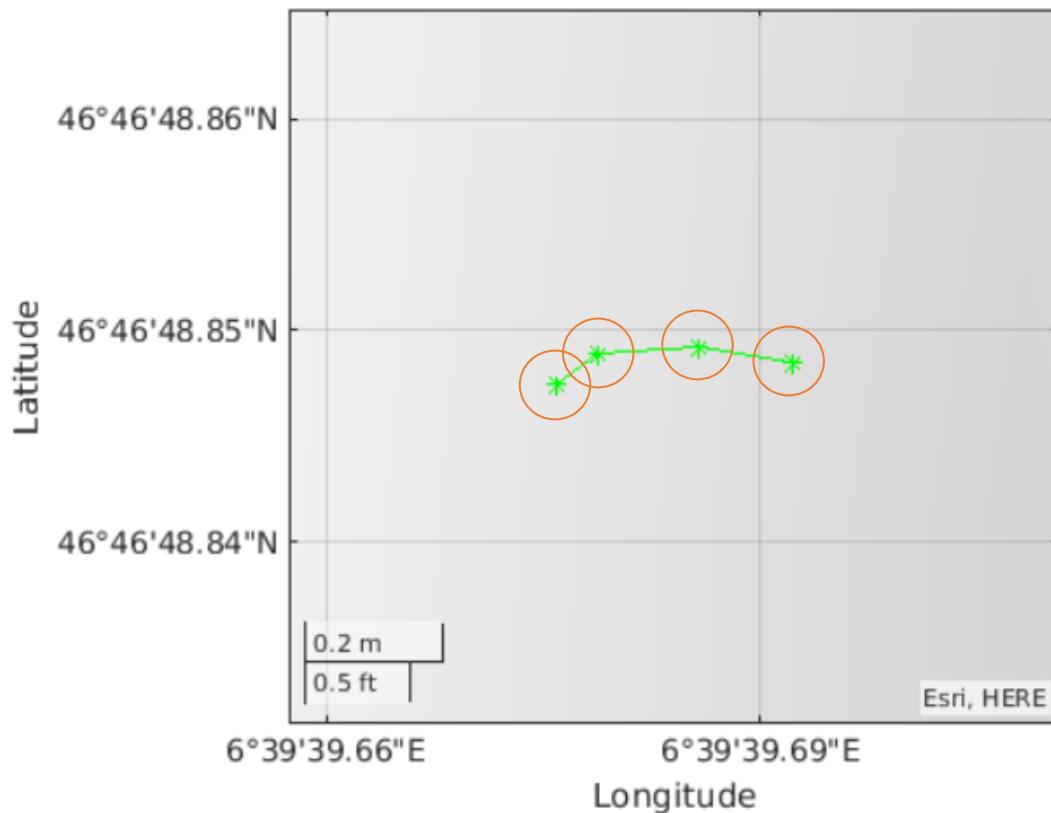


Figure 6-3 : Dispersion des points (ASI float)

Les points en vert représentent les quatre points mesurés le long de la règle. A première vue, il est possible de penser que l'alignement des points n'est pas irréprochable. C'est pourquoi, les cercles en orange ont été ajoutés afin de représenter la précision horizontale indiquée par le module GNSS (~50 mm).

En ajoutant cette possibilité d'erreur, l'alignement est aisément perceptible.

D'après les erreurs mesurées et la dispersion des points, la précision attendue par le récepteur GNSS concorde bien avec la précision relative.

#### 6.4.2 ASI fixe

Cette analyse est semblable à l'analyse précédente mais le mode RTK est fixe. Le tableau contenant les données relatives à cette analyse est ci-dessous (Tableau 6-6) :

Distance réelle [mm]	Long. [°]	Lat [°]	Distance mesurée [mm]	Erreur [mm]	HA [mm]
0	6.6610166	46.7802367	0	-	14
100	6.6610178	46.7802368	92	8	14
200	6.6610192	46.7802368	199	1	14
300	6.6610205	46.780237	300	0	14

Tableau 6-6 : Données de la précision relative (ASI fixe)

Pour tous les points mesurés, l'erreur réelle est inférieure à la précision prévue par le module GNSS.

Voici ci-dessous la représentation de la dispersion des différents points mesurés :

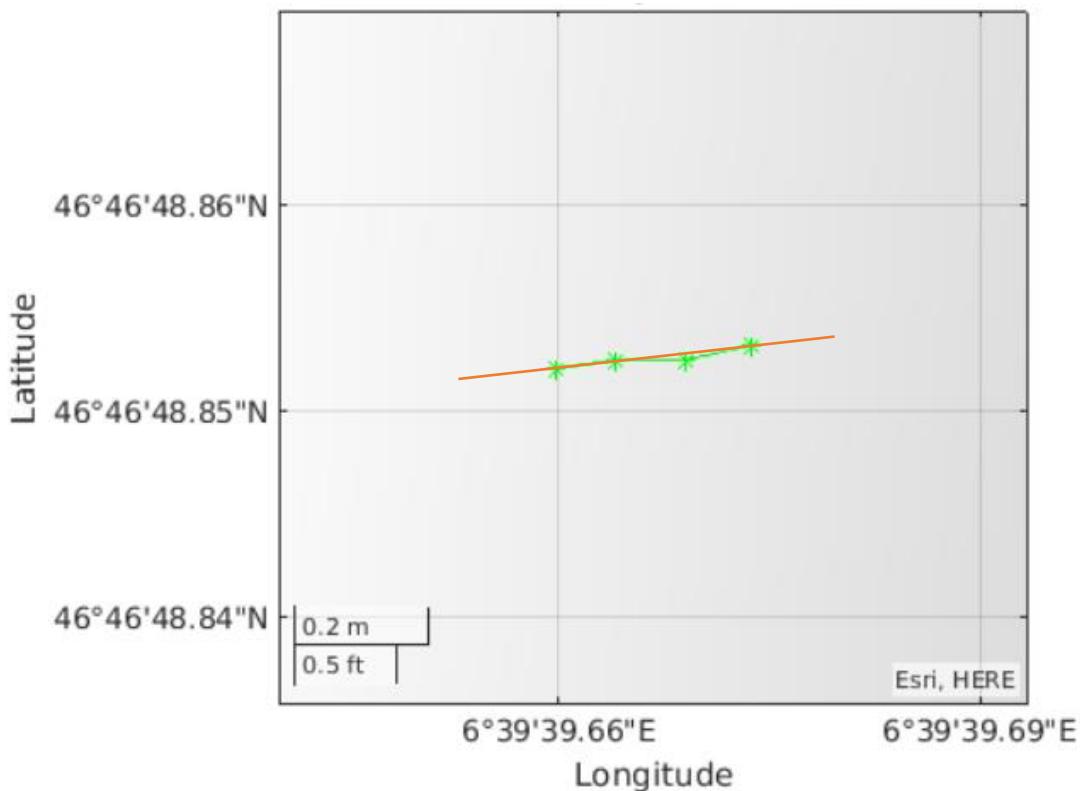


Figure 6-4 : Dispersion des points (ASI fixe)

Afin de faciliter la perception de l'alignement des points, une ligne orange a été ajoutée. Le troisième point est légèrement décalé, sans quoi, l'alignement est remarquable.

D'après les erreurs mesurées et la dispersion des points, la précision indiquée par le récepteur GNSS correspond une nouvelle fois bien avec la précision relative.

#### 6.4.3 HEIG fixe

Cette section est semblable à l'analyse précédente mais la station de référence utilisée est celle de la HEIG-VD. Le tableau contenant les données mesurées est ci-dessous (Tableau 6-7) :

Distance réelle [mm]	Long. [°]	Lat [°]	Distance mesurée [mm]	Erreur [mm]	HA [mm]
0	6.6610173	46.7802239	0	-	14
100	6.6610186	46.780224	100	0	14
200	6.66102	46.7802239	206	-6	14
300	6.6610213	46.7802239	305	-5	14

Tableau 6-7 : Données de la précision relative (HEIG fixe)

Pour tous les points mesurés, l'erreur réelle est inférieure à la précision prévue par le module GNSS.

Voici ci-dessous la représentation de la dispersion des différents points mesurés :

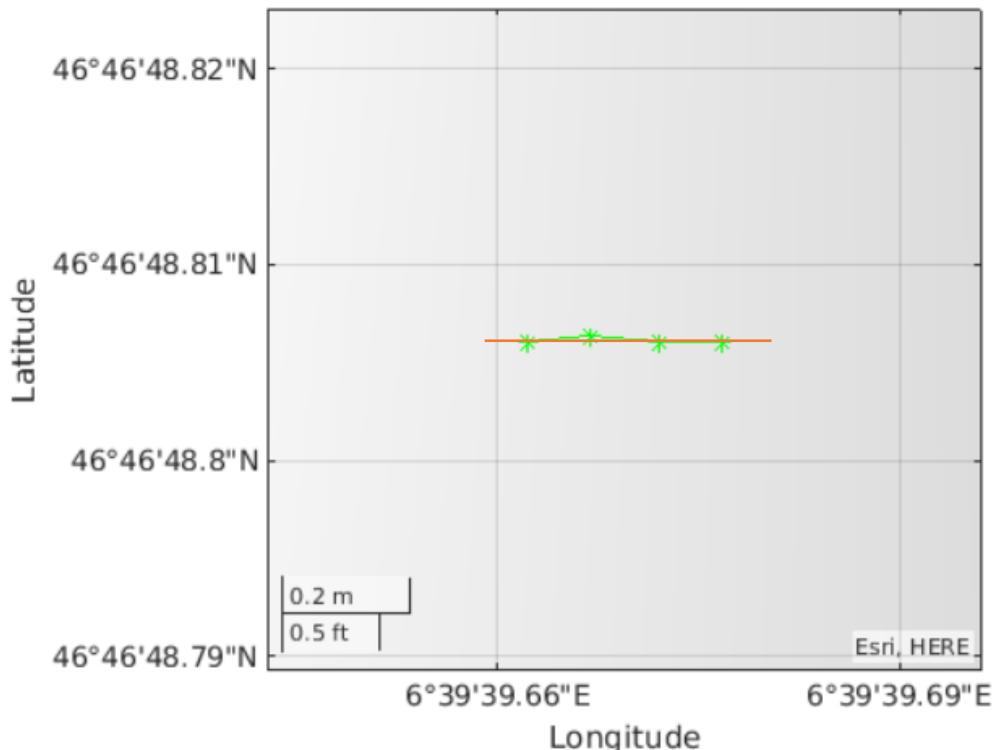


Figure 6-5 : Dispersion des points (HEIG fixe)

Afin de faciliter, la perception de l'alignement des points, une ligne orange a été ajoutée. Le deuxième point est légèrement décalé, sans quoi, l'alignement est de nouveau remarquable.

D'après les erreurs mesurées et la dispersion des points, la précision indiquée par le récepteur GNSS correspond une nouvelle fois bien avec la précision relative.

Pour conclure, la réelle précision relative observée a toujours été inférieure à la précision fournie par le module GNSS. Cette dernière correspond bien à la réelle précision relative.

## 6.5 Analyse de la stabilité

Ce chapitre va permettre d'analyser la stabilité de deux facteurs :

1. Des temps de mise en place d'un rover.
2. Des modes RTK.

### 6.5.1 Stabilisation des temps de mise en place

Plus précisément, cette analyse va permettre de déterminer les temps d'attente afin de profiter des différents états d'un rover. Voici les différents états observés :

- *LTE* : La connexion internet est prête.
- *RTK* : La technologie RTK est en cours d'utilisation.
- *FLOAT* : Le mode RTK passe en mode float.
- *FIXE* : Le mode RTK passe en mode fixe.

Les mesures ont été faites à des moments et avec des temps météo aléatoires. L'environnement choisi lors des mesures est toujours bien dégagé. Les mesures dont le mode fixe n'a pas été abouti n'ont pas été reportées. Cette analyse a été effectuée avec deux types de stations de références :

1. Antenne réalisée avec un capteur d'ASI.
2. Antenne de la HEIG-VD.

#### 6.5.1.1 Temps de mise en place (ASI)

Le tableau contenant les données relatives à cette analyse est ci-dessous (Tableau 6-8). Les temps reportés représentent le temps depuis le démarrage du rover.

N°	États			
	LTE	RTK	FLOAT	FIXE
1	00:01:32	00:01:37	00:01:40	00:02:58
2	00:01:31	00:01:34	00:01:36	00:03:26
3	00:01:27	00:01:29	00:01:32	00:04:59
4	00:01:29	00:01:30	00:01:32	00:02:26
5	00:01:35	00:01:37	00:01:39	00:03:52
6	00:01:56	00:01:58	00:02:00	00:05:41
<b>Moyenne</b>	00:01:35	00:01:38	00:01:40	00:03:54
<b>Min</b>	00:01:27	00:01:29	00:01:32	00:02:26

Tableau 6-8 : Données de la stabilité du temps de mise en place (ASI)

En moyenne, 95 secondes sont nécessaires afin d'obtenir une connexion internet. Aussitôt, il suffit en moyenne de 3 secondes pour que le rover passe en mode RTK et de 2 secondes de plus pour passer en mode float. Dès lors, la précision du rover s'accroît continuellement jusqu'à atteindre le mode fixe.

Celui-ci est atteint en moyenne 2 minutes et 14 secondes après le mode float.

Au pire, le mode fixe n'est jamais atteint, néanmoins le mode float a toujours été obtenu.

Au mieux, le mode fixe est atteint en moins de 1 minute après la connexion internet.

### 6.5.1.2 Temps de mise en place (HEIG-VD)

Le tableau contenant les données relatives à cette analyse réalisée avec l'antenne de la HEIG-VD est ci-dessous (Tableau 6-9) :

N° test	États			
	LTE	RTK	FLOAT	FIXE
1	00:01:31	00:01:46	00:01:48	00:03:41
2	00:01:30	00:01:43	00:01:45	00:05:03
3	00:01:35	00:01:44	00:01:45	00:08:57
4	00:01:34	00:01:40	00:01:42	00:01:56
5	00:01:28	00:01:36	00:01:38	00:04:32
6	00:01:44	00:01:47	00:01:49	00:06:08
<b>Moyenne</b>	00:01:34	00:01:43	00:01:45	00:05:03
<b>Mieux</b>	00:01:28	00:01:36	00:01:38	00:01:56

Tableau 6-9 : Données de la stabilité du temps de mise en place (HEIG)

En moyenne, 94 secondes sont nécessaires afin d'obtenir une connexion internet. Aussitôt, il suffit en moyenne de 9 secondes pour que le rover passe en mode RTK et de 2 secondes de plus pour passer en mode float.

Le mode fixe est atteint en moyenne 3 minutes et 18 secondes après le mode float.

Au mieux, le mode fixe est atteint en moins de 30 secondes après la connexion internet.

Le temps afin d'obtenir un rover prêt avec la technologie RTK actif et le mode float ne diffère quasiment pas en fonction de la station de référence. Ce temps est relativement stable. En effet, le mode float est couramment atteint dans un bon environnement environ 1 minute et 45 secondes après le démarrage du rover.

En revanche, le mode fixe n'est pas atteint de manière stable.

## 6.5.2 Modes RTK

Ce chapitre va permettre d'analyser la stabilité des modes RTK obtenus sur le rover d'ASI.

Pour ce faire, les plus grands fichiers de logs ont été analysés afin de déterminer le pourcentage de temps passé dans chaque mode (normal, float, fixe).

Une fois toutes les données traitées, le tableau ci-dessous (Tableau 6-10) a été créé. Chaque ligne représente un fichier différent. Pour chaque fichier, le temps total de mesure est indiqué, ainsi que, le pourcentage passé dans chaque mode.

Mode RTK			Temps total
Normal	Float	Fixe	
0.5%	14.5%	85.0%	16min
0.5%	43.5%	56.0%	15min
2.5%	59.2%	38.3%	12min
0.6%	97.0%	2.40%	16min
0.5%	25.3%	74.2%	18min
0.5%	61.6%	37.9%	14min
1.4%	97.80%	0.8%	18min

Tableau 6-10 : Données de la stabilité des modes RTK

Le pourcentage de temps passé dans le mode normal n'a jamais dépassé 2.5%. Ce qui prouve que le système complet est stable.

En revanche, la proportion de temps passé en mode float ou en mode fixe varie aisément. Pour atteindre un mode fixe, un nombre de satellites considérable doit être visible simultanément. De plus, les signaux reçus de ces satellites doivent être de bonne qualité afin que la station de base puisse créer des données de corrections et que le rover puisse les appliquer.

Malheureusement, cette analyse n'a pas été séparée par station de référence comme l'ont été la plupart des analyses. Cependant, en effectuant ces tests, une différence a été remarquée. L'antenne placée sur le toit de la HEIG-VD permet d'atteindre un mode fixe plus facilement qu'une antenne réalisée par un capteur d'ASI.

Cela peut être expliqué pour principalement deux raisons :

1. La différence de qualité d'installation des antennes.
2. La différence de placement des stations de références.

## 6.6 Réelle différence avec ou sans le RTK

Le but de ce chapitre est de visualiser la différence qu'apporte la technologie RTK par rapport à un système GNSS à haute précision.

Deux différents tests ont été effectués :

1. Le test de dérive
2. Le test d'itinéraire

Le test de dérive consiste à placer le capteur de manière immobile et d'observer la dérivation du positionnement.

L'objectif du test d'itinéraire est de visualiser des parcours de capteurs avec et sans la technologie RTK.

### 6.6.1 Dérive

Tous les tests ont été réalisés dans un champ à côté de la HEIG-VD afin de profiter pleinement de la station de base située sur le toit de cette école. Pour toutes les mesures, le temps d'observation est de 1 minute. Cependant, l'usage de différentes fréquences<sup>16</sup> ont été expérimentées :

1. 1 Hz : La fréquence recommandée par défaut.
2. 5 Hz : La fréquence maximum recommandée en utilisant deux constellations.
3. 10 Hz : La fréquence souhaitée dans le domaine du sport, pour un skieur par exemple.

Pour chaque fréquence, différents modes ont été testés :

1. NORMAL : Un capteur d'ASI sans le mode RTK.
2. RTK FLOAT : Un capteur avec RTK en mode float, avant d'atteindre le mode fixe.
3. RTK FIXE : Un capteur avec RTK en mode fixe.

*Remarque : Pour tous les graphiques qui suivront dans ce chapitre, l'unité des axes est le mètre.*

Les mesures en mode float n'ont pas pu être faites de manière stable car le mode fixe a toujours été atteint rapidement. Néanmoins, pour avoir un ordre d'idée de ces mesures, les données durant la minute avant d'atteindre le mode fixe ont été relevées pour avoir une référence pour les tests.

---

<sup>16</sup> C'est la fréquence à laquelle les données de navigation sont envoyées au serveur.

### 6.6.1.1 1 Hz

Ci-dessous, les différences de dérive pour tous les modes avec un fréquence de 1 Hz :

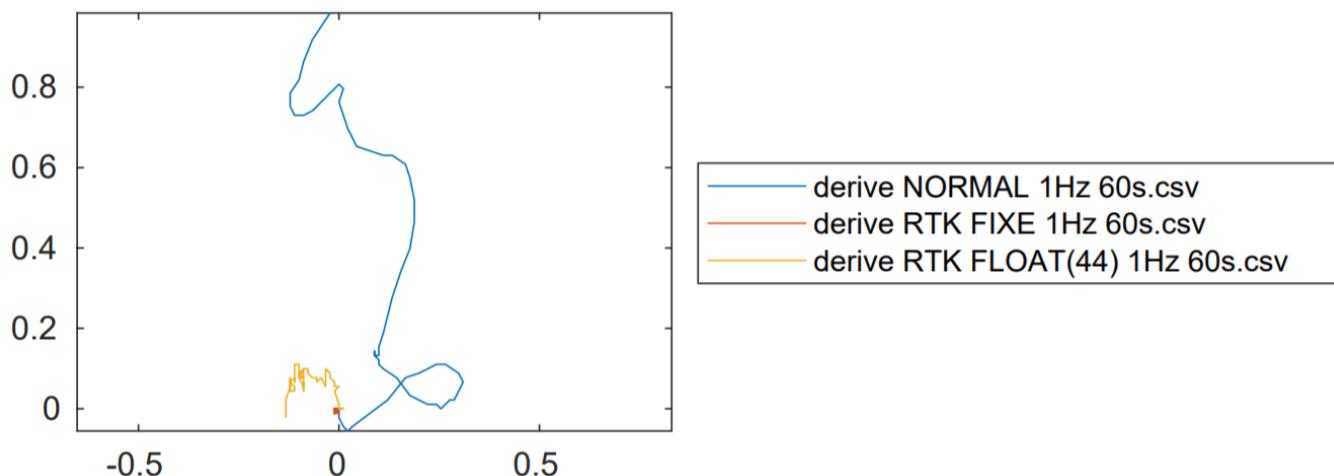


Figure 6-6 : Dérive de tous les modes à 1 Hz

La différence est frappante ; un capteur normal peut dériver jusqu'à un mètre verticalement, un capteur en mode float dérive de quelques centimètres et un capteur en mode fixe paraît comme un point.

Afin d'avoir un meilleur aperçu sur les valeurs de dérive pour les modes RTK, voici le graphique sans le mode normal :

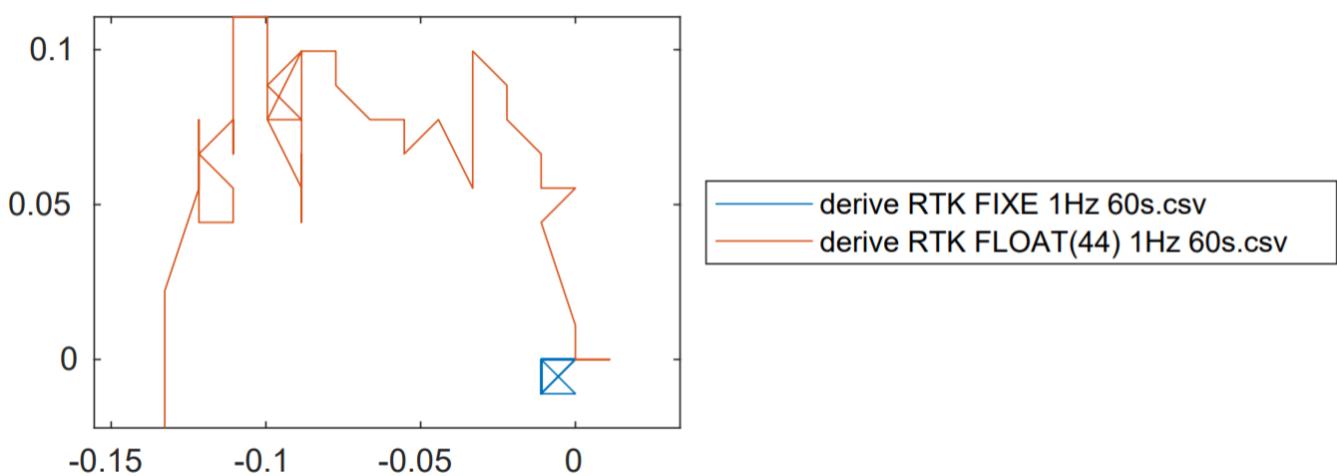


Figure 6-7 : Dérive des modes RTK à 1Hz

Les valeurs de dérive d'un capteur en mode float sont inférieures à 20 centimètres et celles du mode fixe sont dans l'ordre du centimètre.

Pour avoir une idée encore plus précise, voici le graphique uniquement en mode fixe :

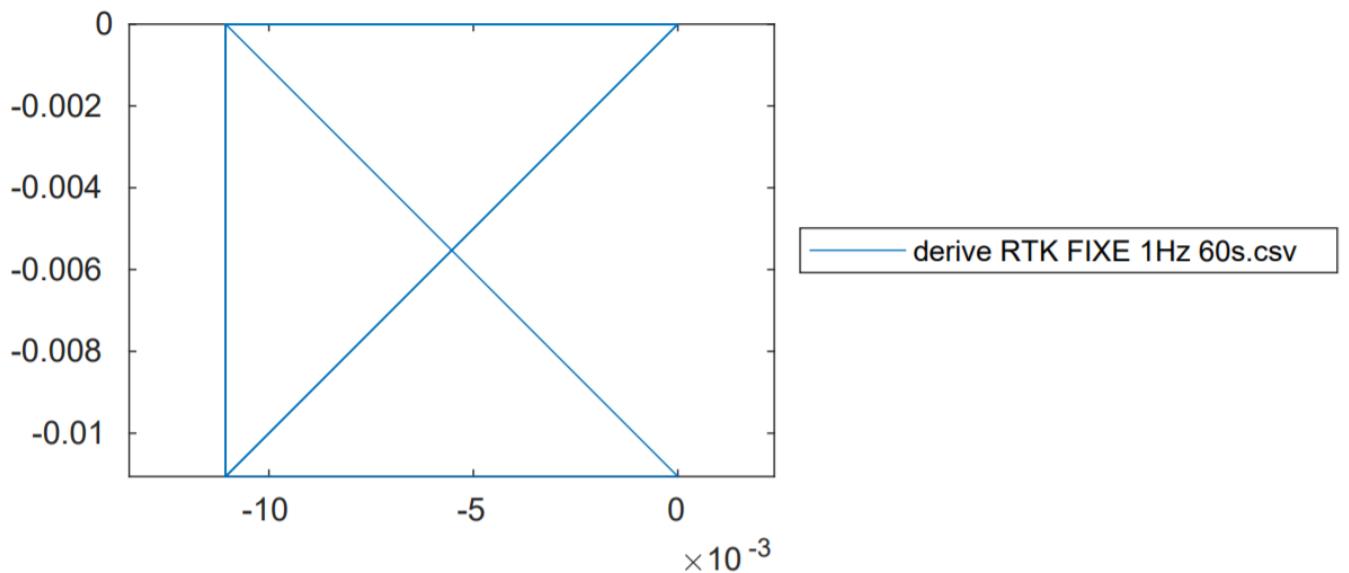


Figure 6-8 : Dérive du mode fixe à 1Hz

Comme attendu, les valeurs de dérive du mode fixe sont d'environ un centimètre.

#### 6.6.1.2 5 Hz

Ci-dessous, les différences de dérive pour une fréquence de 5 Hz :

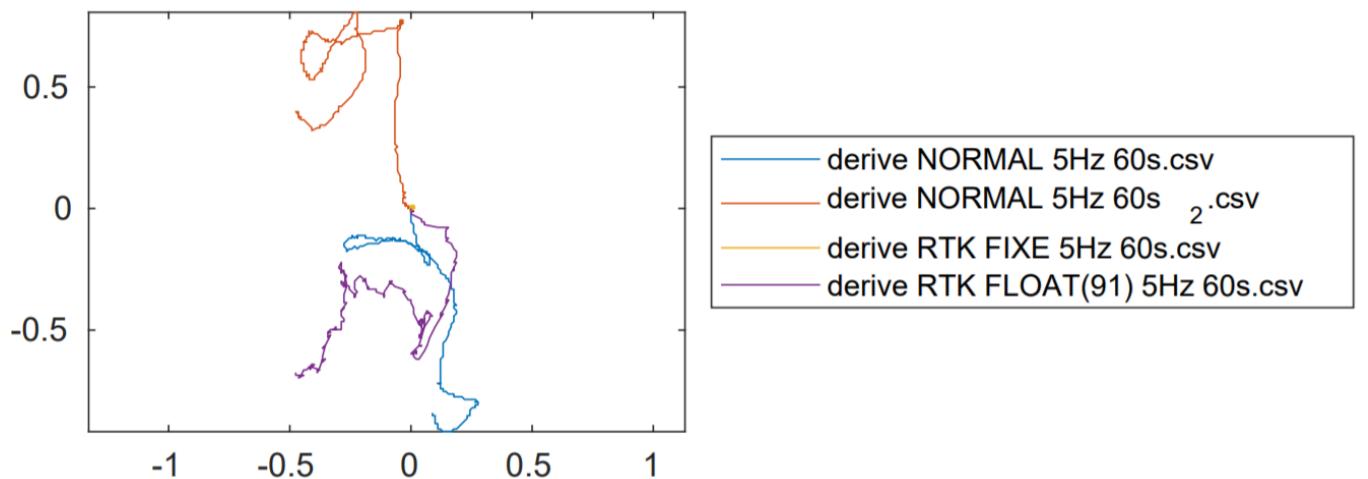


Figure 6-9 : Dérive de tous les modes à 5 Hz

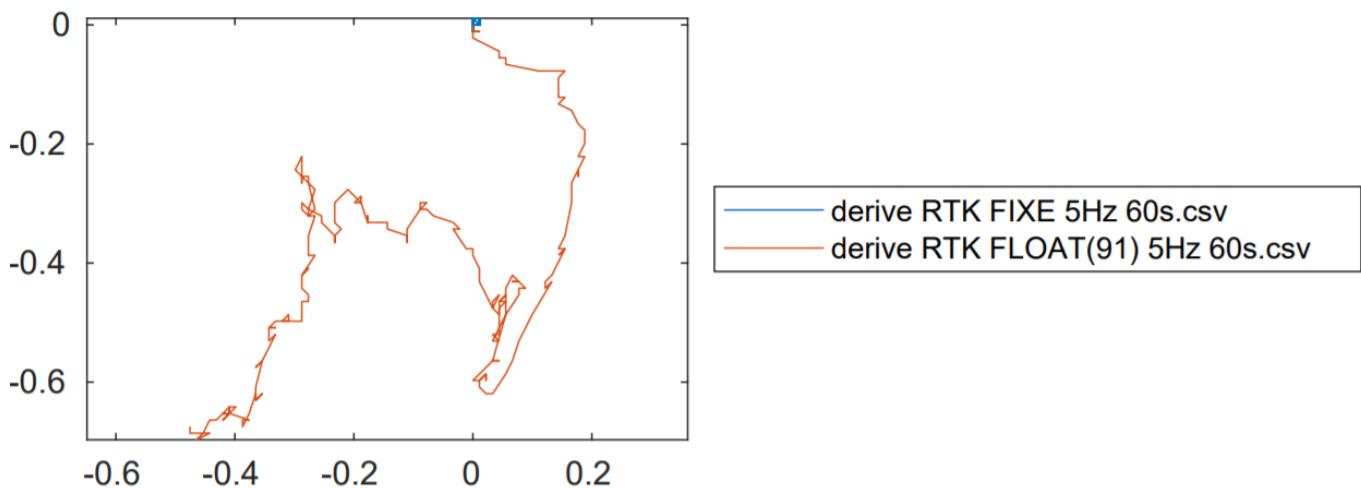


Figure 6-9 : Dérive des modes RTK à 5Hz

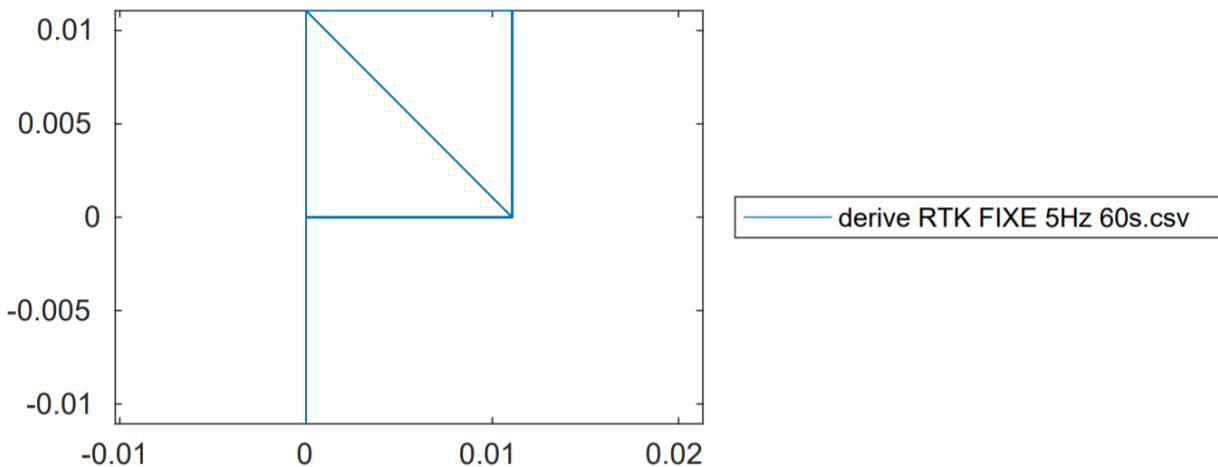


Figure 6-10 : Dérive du mode fixe à 5Hz

Les mêmes conclusions que celles pour une fréquence de 1 Hz peuvent être faites. Toutefois, la différence entre le mode normal et le mode float est moins frappante. Cela est dû au fait que le mode fixe a rapidement été atteint.

Le mode fixe garde sa précision centimétrique malgré une fréquence plus élevée.

### 6.6.1.3 10 Hz

Ci-dessous, les différences de dérive pour une fréquence de 10 Hz :

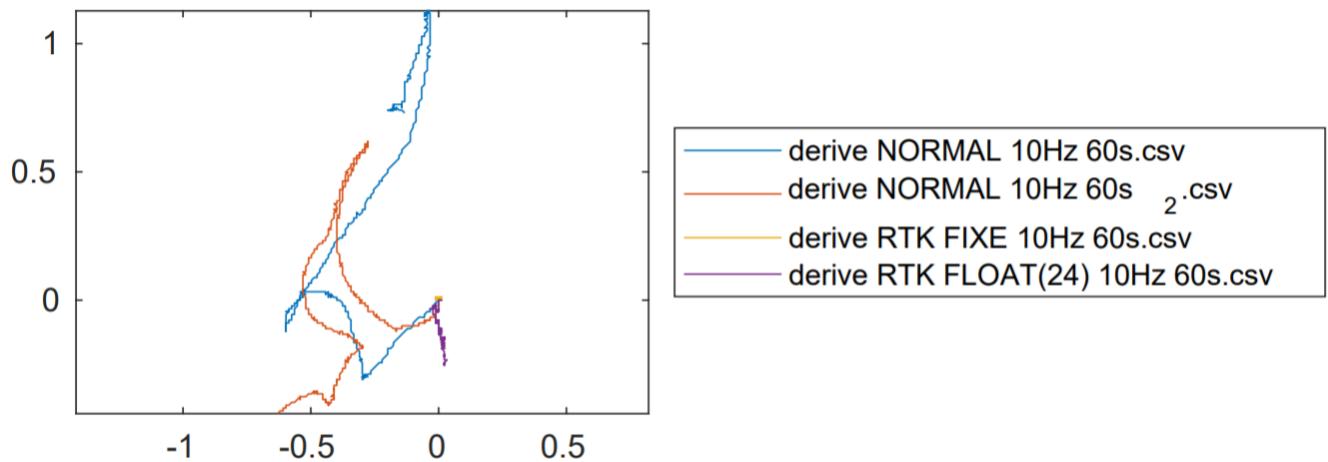


Figure 6-11 : Dérive de tous les modes à 10 Hz

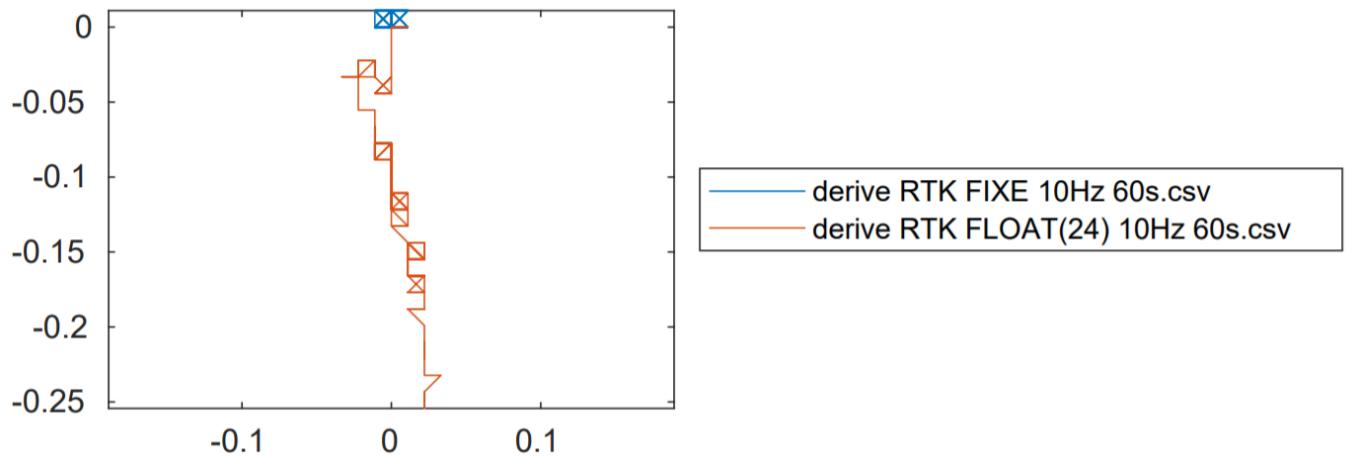


Figure 6-12 : Dérive des modes RTK à 10 Hz

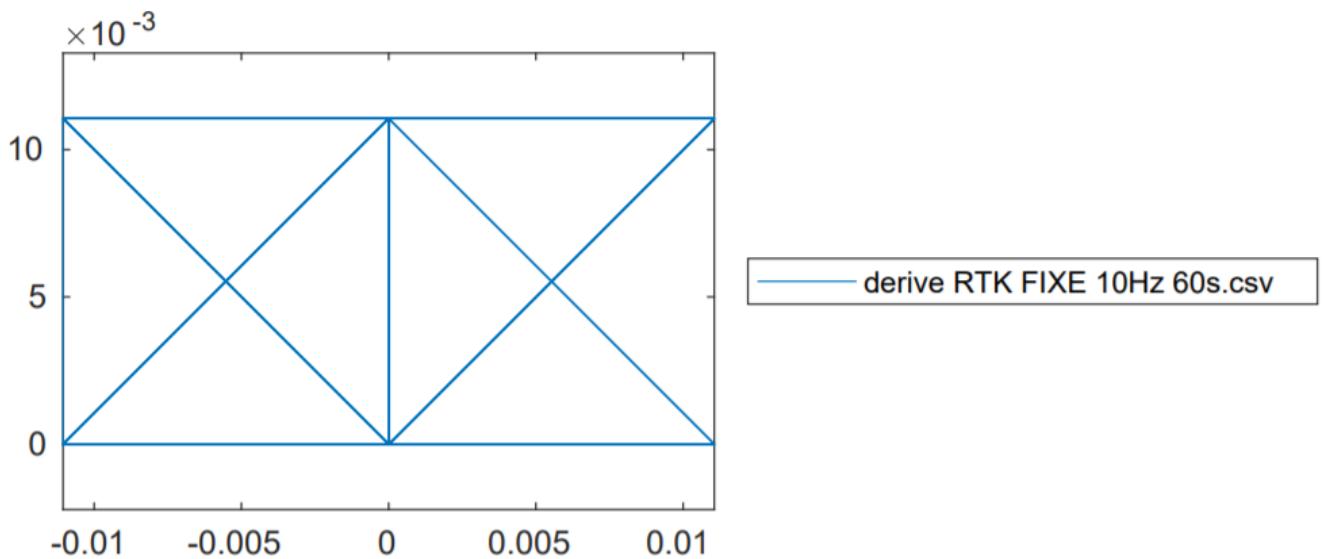


Figure 6-13 : Dérive du mode fixe à 10 Hz

Là encore, les mêmes conclusions peuvent être faites, à la différence que la valeur de dérive du mode float est particulièrement verticale, jusqu'à 25 centimètres.

Le mode fixe garde sa précision centimétrique malgré une fréquence de 10 Hz non recommandée.

Bien que les mesures des modes float n'aient pas été effectuées de manière stable, l'amélioration de la dérive est appréciable pour ce mode et impressionnante pour un mode fixe.

### 6.6.2 Exemples d'itinéraires

Afin de constater visuellement la différence de qualité qu'apporte la technologie RTK, un itinéraire a été parcouru avec deux capteurs simultanément : un capteur sans la réception de données de corrections et un capteur avec la réception.

Voici l'itinéraire des capteurs visible grâce au fichier KML ouvert avec l'application « Google Earth » :



Figure 6-14 : Itinéraire d'un rover en mode normal



Figure 6-15 : Itinéraire d'un rover en mode RTK

La différence est nette. Le traçage d'un rover en mode normal (*en haut*) est irrégulier alors que celui du rover en mode RTK (*en bas*) est impeccable.

Afin de voir une différence plus minutieuse, le contour d'une grille, à la fin du parcours (*cercle rouge*), a été suivie avec les capteurs.

Pour prendre conscience de la taille de la grille, voici une photo :



Figure 6-16 : Visualisation de la taille de la grille

Le diamètre de la grille est d'environ 25 centimètres.

Voici maintenant le rendu du traçage des capteurs mis côte à côté :



Figure 6-17 : Différence de contour de la grille

Là encore, la différence est flagrante. A gauche, le capteur en mode RTK trace un joli cercle précis. A droite, le rover en mode normal a un traçage beaucoup moins précis.

Comme attendu, la différence de précision entre un système avec ou sans la technologie RTK est significative.

## 7 Conclusion

### 7.1 Conclusion du projet

Si la technologie RTK a été choisie c'est qu'elle seule proposait une précision centimétrique en temps réel. Grâce au kit de développement fourni par la société ASI, cette précision a pu être validée.

Un système complet a pu être mis en place afin de profiter pleinement de cette technologie en utilisant uniquement le matériel de l'entreprise. De plus, un système supplémentaire a été réalisé afin de profiter de la technologie comparable NRTK. En effet, il est maintenant possible aux capteurs d'ASI d'exploiter cette technologie grâce à une station de référence se trouvant dans un réseau existant ou réalisée directement par un capteur d'ASI.

En plus de profiter de cette technologie, ces systèmes permettent d'avoir accès aux données de position de la station de base ainsi que de tous les rovers. Il est donc possible de traiter toutes ces données comme bon nous semble en profitant de la puissance de calcul du serveur. Il est donc facilement envisageable d'ajouter une fonctionnalité permettant de détecter le passage d'un skieur à une ligne d'arrivée.

Les tests et analyses effectués dans le chapitre précédent ont permis d'apprendre plusieurs choses :

1. La précision absolue dépend de la précision de positionnement de la station de référence.
2. La précision relative correspond bien à la précision fournie par le module GNSS.
3. La précision relative du système peut atteindre une précision de 1,4 cm horizontale et 1 cm verticale.
4. Le temps nécessaire au rover d'atteindre un mode float est inférieur à 2 minutes
5. Le mode fixe dépend grandement de la station de référence ainsi qu'à l'environnement.

Finalement, ce nouveau système, profitant de la technologie RTK, a apporté une amélioration de la précision considérable comparé à l'ancien système. Ceci a été prouvé dans la différence de la dérive et les exemples d'itinéraires.

### 7.2 Améliorations

Il serait possible d'améliorer divers points des systèmes, comme par exemple :

- Traitement complémentaire des données sur le serveur.
- Modification à distance de la station de référence utilisée.

### 7.3 Conclusion personnelle

Ce travail m'a vivement intéressé car il a une application réelle et utile. Les technologies LTE et RTK m'ont particulièrement plu car ce sont des domaines où les champs de possibilités sont énormes. Je suis très content d'avoir travaillé pour la société ASI et d'avoir abouti à un système fonctionnel.

## 8 Authentification

Le soussigné, Isaïa Spinelli, atteste par la présente avoir réalisé seul ce travail et n'avoir utilisé aucune autre source que celles expressément mentionnées.

Yverdon-les-Bains, le 12 avril 2020

Isaïa Spinelli

## 9 Références

- [1] Wikipédia. (2020, 06 01). *Système de positionnement par satellite*. Récupéré sur Wikipédia: [https://fr.wikipedia.org/wiki/Syst%C3%A8me\\_de\\_positionnement\\_par\\_satellites](https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_positionnement_par_satellites)
- [2] Figure : Visualisation d'une triangulation. (2020, 04 14). Récupéré sur Google: <https://metrotec.ee/wp-content/uploads/2017/01/maxresdefault.jpg>
- [3] Office fédéral de topographie. (2020, mai). *swipos-GISGEO*. Récupéré sur [https://www.swisstopo.admin.ch/fr/home/products/services/swipos/swipos-gisgeo.html#dokumente\\_und\\_publik](https://www.swisstopo.admin.ch/fr/home/products/services/swipos/swipos-gisgeo.html#dokumente_und_publik)
- [3] Trimble Navigation. (s.d.). *Page d'accueil*. Récupéré sur <https://www.trimble.com/>
- [4] RTCM. (s.d.). *Radio Technical Commission for Maritime Services (RTCM)*. Récupéré sur <https://www.rtcm.org/>
- [5] use-snip, S. . (2020, 06 02). *RTCM 3 Message List*. Récupéré sur use-snip: <https://www.use-snip.com/kb/knowledge-base/rtcm-3-message-list/>
- [6] 3GPP. (2020). *About 3GPP*. Récupéré sur <https://www.3gpp.org/about-3gpp>
- [7] Alberding. (2020). *GNSS Data Management Software (Ntrip)*. Récupéré sur Alberding: <https://www.alberding.eu/en/ntripcaster.html>
- [8] User Guide NEO-M8P. (s.d.). *NEO-M8P series*. Récupéré sur <https://www.u-blox.com/en/product/neo-m8p-series#tab-documentation-resources>
- [9] Centipède (ANCELIN Julien). (s.d.). *Centipède RTK*. Récupéré sur <https://jancelin.github.io/docs-centipedeRTK/>

## 10 Liste des abréviations

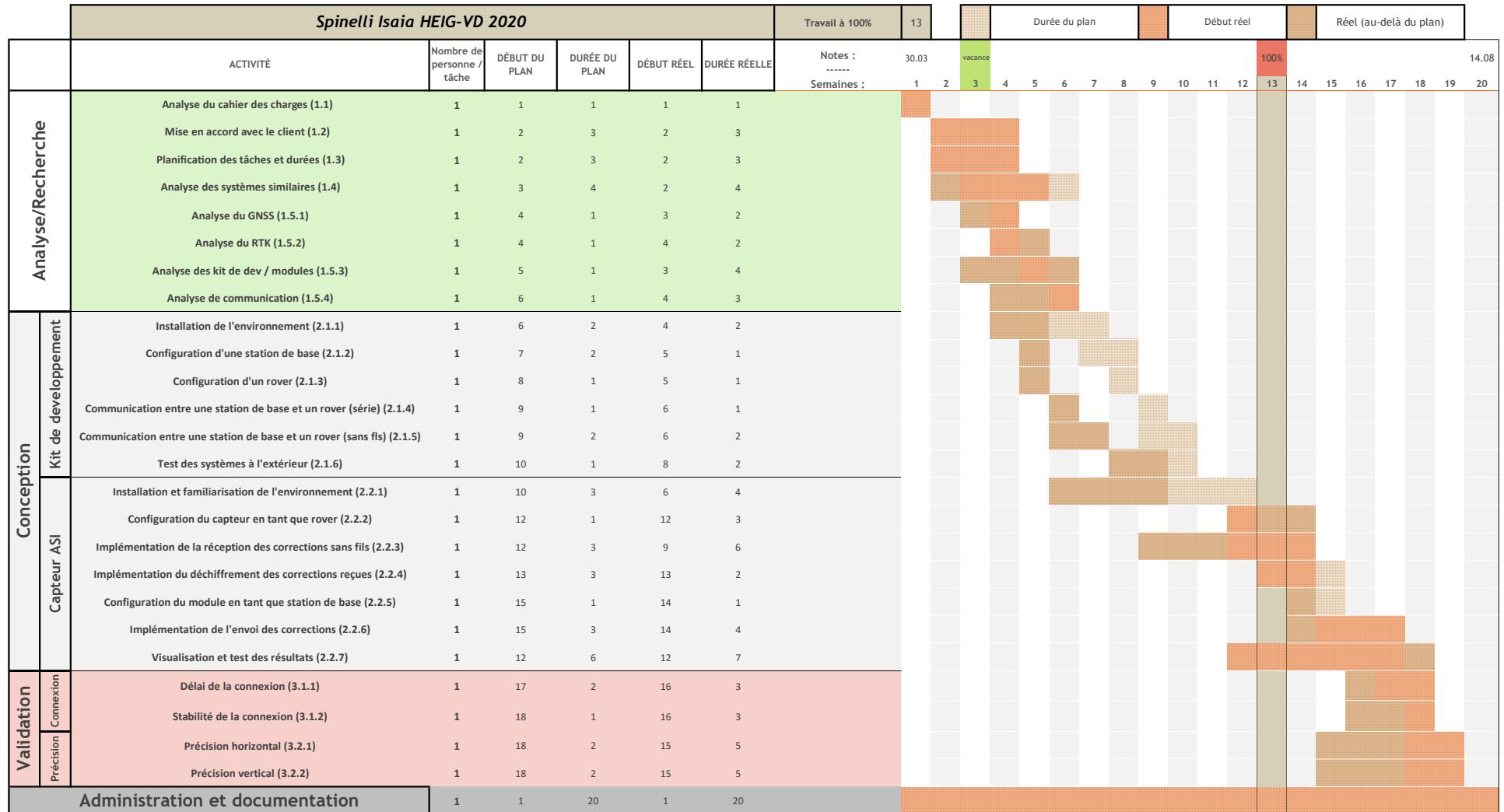
<b>ABAS</b>	<i>Aircraft Based Augmentation System</i>
<b>ASI</b>	<i>Advanced Sport Instruments SA</i>
<b>AT</b>	<i>Commandes Hayes</i>
<b>BeiDou</b>	<i>BeiDou Navigation Satellite System (système de navigation Chinois)</i>
<b>BKG</b>	<i>Agence fédérale allemande de cartographie et de géodésie</i>
<b>CRC</b>	<i>cyclic redundancy check (Somme de contrôle)</i>
<b>EGNOS</b>	<i>European Geostationary Navigation Overlay Service</i>
<b>Galileo</b>	<i>Système de positionnement Européen</i>
<b>GBAS</b>	<i>Ground-Based Augmentation system</i>
<b>GLONASS</b>	<i>Globalnaïa navigatsionnaïa spoutnikovaïa sistéma (système de navigation Russe)</i>
<b>GNSS</b>	<i>Géolocalisation et Navigation par un Système de Satellites</i>
<b>GPS</b>	<i>Global Positioning System</i>
<b>GSM</b>	<i>Global System for Mobile Communications</i>
<b>HTTP</b>	<i>Hypertext Transfer Protocol</i>
<b>IRNSS</b>	<i>Indian Regional Navigation Satellite System</i>
<b>IDE</b>	<i>Integrated development environment</i>
<b>KML</b>	<i>Keyhole Markup Language</i>
<b>LSB</b>	<i>least significant bit</i>
<b>LTE</b>	<i>Long Term Evolution</i>
<b>LTE-M</b>	<i>Long Term Evolution- Machine Type Communication</i>
<b>MSB</b>	<i>most significant bit</i>
<b>NRTK</b>	<i>Network Real Time Kinematic</i>
<b>NTRIP</b>	<i>Networked Transport of RTCM via Internet Protocol</i>
<b>PACC H</b>	<i>Position accuracy Horizontal</i>
<b>PACC V</b>	<i>Position accuracy Vertical</i>
<b>PPK</b>	<i>Post Processed Kinematic</i>
<b>PVT</b>	<i>position, velocity and time</i>
<b>QZSS</b>	<i>Quasi-Zenith Satellite System</i>
<b>RFID</b>	<i>radio-identification</i>
<b>RTCM</b>	<i>Radio Technical Commission for Maritime Services</i>

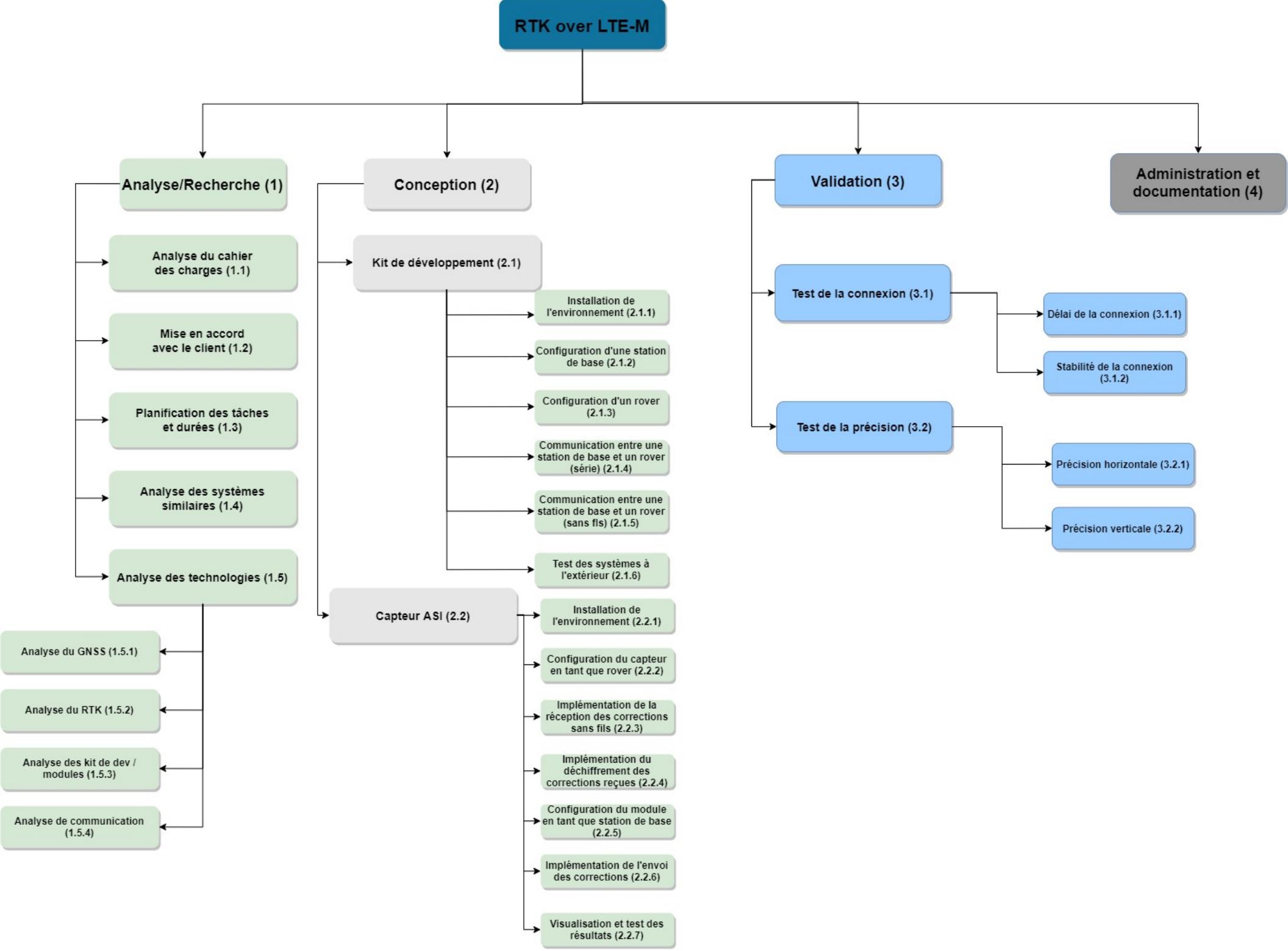
<b>RTK</b>	<i>Real Time Kinematic</i>
<b>SBAS</b>	<i>Satellite-Based Augmentation System</i>
<b>TACC</b>	<i>Position accuracy Time</i>
<b>UBX</b>	<i>Protocole propriétaire u-blox</i>
<b>UHF</b>	<i>Ultra hautes fréquences</i>
<b>VRS</b>	<i>Virtual Reference Station</i>
<b>WBS</b>	<i>Work Breakdown Structure</i>

## 11 Annexes

1. Planning
2. Organigramme des tâches du projet (WBS)
3. Journal de travail

# Planification du TB ( Géolocalisation centimétrique à l'aide de capteurs RTK pour les activités sportives)





# Journal de travail

## Géolocalisation centimétrique à l'aide de capteurs RTK pour les activités sportives

**Confidentiel**

<b>Lieu :</b>	Yverdon-les-Bains, HEIG-VD
<b>Département :</b>	TIC
<b>Filière :</b>	Informatique
<b>Orientation :</b>	Informatique embarquée
<b>Étudiant :</b>	Spinelli Isaïa
<b>Responsable :</b>	Mosqueron Romuald
<b>Entreprise partenaire :</b>	ASI (Yersin Lionel)
<b>Sujet :</b>	Géolocalisation centimétrique à l'aide de capteurs RTK pour les activités sportives

Yverdon-les-Bains, le 5 avril 2020

## Semaine N°1 : 30 mars – 05 avril

### Travaux effectués

Premier contact avec M. Mosqueron et M. Yersin le vendredi 3 avril. Commencement du document du journal de travail.

### Problèmes rencontrés

### Solution

### Remarques

M. Mosqueron a pu m'expliquer quelques détails sur les documents administratifs et m'a indiqué qu'il fallait commencer à faire des recherches sur la technologie RTK. M. Yersin m'a parlé des deux différents possibles kit de développement, de ses capteurs et il m'a montré un site pour que je commence quelques recherches. Il a aussi évoqué le fait de réfléchir sur le moyen de communication entre la station de base et les modules (rovers).

## Semaine N°2 : 06 – 12 avril

### Travaux effectués

Début des recherches sur les technologies concernées. Lecture et analyse sur les différents capteurs et kit de développement de u-blox. Commencement des documents demandés (cahier des charges, planning, WBS, journal de travail). Discussion avec M. Yersin sur la livraison du matériel. Mise au point avec M. Mosqueron.

### Problèmes rencontrés

### Solution

### Remarques

Le kit de développement C94-M8P est commandé.

## Semaine N°3 : 13 – 19 avril

### Travaux effectués

Commencement du rapport sur les aspects techniques (GNSS, RTK). Avancement sur les documents en général.

### Problèmes rencontrés

### Solution

### Remarques

## Semaine N°4 : 20 – 26 avril

### Travaux effectués

Réception du kit de développement RTK. Lecture des documents pour l'application u-center et la mise en place du kit. Meeting et envoie du journal de travail, planification, WBS et cahier des charges à M. Mosqueron. Recherche et documentation approfondie sur l'aspect techniques RTK. Premier contact avec Romain (employé chez ASI) chargé de me fournir des informations sur le matériel de l'entreprise. Meeting avec Romain pour m'informer des premières informations essentielles concernant la programmation du matériel d'ASI.

### Problèmes rencontrés

### Solution

### Remarques

Remarque de Lionel : « Le titre over NBLoT du projet n'est pas correcte ».

## Semaine N°5 : 28 avril – 3 mai

### Travaux effectués

Recherches et documentation approfondie sur les différents moyens de communication entre une base et un rover. Avancement du rapport. Recherche sur des systèmes similaires. Prise en main du kit de développement. Configuration d'une base et d'un rover avec le kit de développement.

### Problèmes rencontrés

### Solution

### Remarques

## Semaine N°6 : 4 – 10 mai

### Travaux effectués

Réception du capteur d'ASI (FELCO) et de matériel pour le programmer. Petit entretien avec Romain pour qu'il me donne des informations de base sur le boîtier et pour mettre à jour la version du software. Réception et lecture du code fourni par ASI. Test des configurations sur le kit de développement RTK. Séance hebdomadaire avec M. Mosqueron.

### Problèmes rencontrés

#### Solution

### Remarques

M. Mosqueron :

- Avoir une date pour la soutenance (31 aout au 11 septembre)
- Ajouter des sous-titres au titre introduction du rapport
- Développer pourquoi le choix du RTK

## Semaine N°7 : 11 – 17 mai

### Travaux effectués

Analyse et test du code fourni sur la capteur ASI. Avancement sur le rapport sur la partie configuration et test du kit de développement RTK. Séance hebdomadaire avec M. Mosqueron.

### Problèmes rencontrés

- Stack overflow lors de l'exécution du code
- La nouvelle tâche créée (NtripTask) ne se préempte pas.

### Solution

- J'ai pu résoudre le problème en ajoutant de la taille à la stack lors de la création de la tâche qui posait problème ou en réduisant la taille des tableaux local à la tâche.
- Pas encore de solution, je n'ai pas réussi à ajouter correctement une tâche au programme.

### Remarques

## Semaine N°8 : 18 – 24 mai

### Travails effectués

Test en extérieur avec le kit de développement RTK pour analyser les différences de performances avec ou sans le mode RTK. Avancement du rapport sur la partie analyse des différences de performances et des différents protocoles utilisés. Recherche sur comment s'abonner à une station de base via le protocole NTRIP.

### Problèmes rencontrés

### Solution

### Remarques

## Semaine N°9 : 25 – 31 mai

### Travails effectués

Test en extérieur avec le kit de développement RTK pour analyser le comportement RTK proche d'une station de base NTRIP. Avancement du rapport sur la partie analyse des différences de performances. Essaye d'ajouter du code dans le projet pour m'abonner à une antenne NTRIP via un socket grâce au bg96. Conférence avec Lionel pour répondre à mes questions. Séance hebdomadaire avec M. Mosqueron. Conférence avec Romain afin de demander une méthode de vérification de la réception des données gps.

### Problèmes rencontrés

### Solution

### Remarques

Je pensais qu'en étant proche d'une station de référence NTRIP, le mode du RTK passerai plus facilement en « fixed » mais ce ne fut pas le cas. Je pense que c'est dû à la communication qui passe par mon smartphone.

## Semaine N°10 : 1 – 7 juin

### Travaux effectués

Analyse du code du capteur ASI. Analyse de la méthode de communication entre les tâches et la gestion des tâches. Réflexion sur le système global pour un capteur en mode rover. Tentative de construction d'un système de communication en streaming entre deux tâches afin d'échanger des données rapidement entre le module LTE et le module GNSS.

### Problèmes rencontrés

1. Problème lors de l'utilisation des buffer stream de FreeRTOS.
2. Débit de données de correction trop élevé pour la réception uart implémentée dans le projet fourni.

### Solution

1. Ajouter la définition suivante dans le fichier de configuration de RTOS :  
**#define INCLUDE\_xTaskGetCurrentTaskHandle 1**
2. Utilisation du mode « transparent access » pour un meilleur débit, mais ne permet aucun autre échange avec le BG96.

### Remarques

## Semaine N°11 : 8 – 14 juin

### Travaux effectués

Implémentation d'un serveur et client socket en C sur mon Windows afin d'essayer d'échanger des informations entre un serveur/client et le capteur. Avancement du rapport général. Réunion hebdomadaire avec M. Mosqueron.

### Problèmes rencontrés

1. Impossible de se connecter à mon serveur depuis le capteur.

### Solution

1. Une application inconnue utilisait déjà le port 2101, donc mon serveur ne recevait rien du capteur.

### Remarques

- J'ai dû configurer mon routeur afin d'indiquer que les données venant du port 2101 doivent être envoyées sur mon PC.

## Semaine N°12 : 15 – 21 juin

### Travaux effectués

Avancement du rapport et schématisation du système logiciel central. Schématisation des systèmes de tests effectuées avec mon serveur et client. Comparaison avec les différences de configuration entre le nouveau module en mode rover et base de référence. Séance avec M. Mosqueron.

### Problèmes rencontrés

### Solution

### Remarques

## Semaine N°13 : 22 – 28 juin

### Travaux effectués

Implémentation des différentes configurations à ajouter au module.

Avancement du rapport sur la partie implémentation du projet.

Analyse des messages RTCM transmis pour l'antenne NTRIP de Pailly.

Tentative de fonctionnement d'un rover sur le nouveau capteur reçu avec l'antenne NTRIP de Pailly.  
Tentative de fonctionnement d'un rover avec des données de corrections envoyées depuis un module du kit GNSS en passant par un caster NTRIP de SNIP et le serveur NTRIP de u-center.

Test de la configuration du module GNSS en mode station de référence sur la nouveau capteur.

Détection de la fin de la procédure du mode Base qui permet de déterminer la position de la station en construisant une moyenne pondérée (Survey-in).

Désactivation de la réception des informations de type Survey -in, puis, activation de l'envoi des messages de correction RTCM.

Test du transfert entre le capteur en mode station de référence et un client NTRIP sur mon PC en passant par SNIP.

Ajout de la gestion de réception et parsing de différents types d'information du module GNSS afin d'avoir accès à des informations sur la qualité du mode RTK et la position relative avec la station de référence.

Réécriture du code de parsing pour la réception des données UBX du module GNSS.

Investigation sur les tailles des tampons uart pour la réception du module BG96 (LTE) et l'envoi au module GNSS pour l'amélioration de la stabilité du mode RTK.

A la place de faire transiter les données RTCM entre les deux tâches GNSS et BG96, j'ai fait un pont entre les deux uart.

### Problèmes rencontrés

1. Lors de l'activation de l'envoi des données de la tâche GNSS au module GNSS, la réception continue des données depuis l'antenne NTRIP de Pailly s'arrête après quelques secondes.
2. La tentative du fonctionnement du rover n'a pas fonctionné. Du moins, les flags indiquant l'état du mode RTK n'étaient pas actifs.
3. Le nouveau capteur s'éteint après environ 5 secondes d'utilisation même si la batterie est pleine.
4. Lorsque plus qu'un type de message d'information est demandé au module GNSS, les données se mettent à jour environ 1 fois sur 3.

### Solutions

1. L'uart entre le module GNSS et la tâche était configuré avec un Baudrate de 9600. Le fait de régler le Baudrate à 115'200 a résolu le problème. Sûrement dû au fait que le Baudrate de l'uart pour le module LTE est aussi de 115'200.
2. Il est possible que les flags RTK ne se lèvent pas car la qualité des signaux GNSS est trop mauvaise à la maison. Des tests à l'extérieur doivent être effectués. Il est aussi possible que la quantité de données soit trop élevée et que certaines données soient ignorées, de ce fait, la

- correction ne pourrait pas s'effectuée. Donc, j'ai testé avec des données de correction depuis un module du kit qui envoie beaucoup moins de donnée que l'antenne à Pailly.
3. Après une discussion avec Romain, la problème de la batterie est connue sur cette nouvelle série. Il est possible qu'en approchant trop près les fils de la batterie à l'antenne GSM cela active la protection court-circuit.
  4. Après quelques recherches et tentatives, j'ai compris que ce problème était dû à la gestion et la structure de la tâche GNSS. En effet, lorsque deux messages différents doivent être parsés, il est essentiel que la tâche GNSS s'exécute 2 fois plus souvent afin parser tous les messages attendus.

#### Remarques

2. Après avoir effectué un test à l'extérieur, j'ai pu constater que le mode RTK a pu être actif en mode float. Cependant, il a été aperçu que très rarement malgré la bonne condition météo. Il en est de même lorsque le test a été effectué avec des données venant d'un module du kit de développement. Mais, après avoir augmenté les tailles du buffers utilisé pour les réception uart du module GSM, le mode RTK a été atteint plus facilement. Afin de m'assurer que toutes les données soient correctement et efficacement transmises, j'ai fait un « pont » entre la réception de l'uart 4 (module LTE) et l'envoi de l'uart 3 (module GNSS). Ceci a permis d'être en mode RTK de manière stable. Toutefois, les modes float ou fixe ne sont toujours pas au rendez-vous.

## Semaine N°14 : 29 juin – 5 juillet

### Travaux effectués

Corrections et relecture du rapport. Crédit de la liste des abréviations. Séance avec M. Mosqueron.

Recherches et tests sur les potentielles différences entre le capteur d'ASI et le module du kit de développement qui pourrait nuire au bon fonctionnement d'un mode RTK float ou fixe :

1. Réception complète des données de correction
2. Données de correction compatible
3. Temps de transmission correcte
4. Distance avec la station de référence

*Remarque : Cependant, avec le module du kit de développement, le mode float a été facilement atteint avec la même antenne à Pailly*

5. Bonne visibilité des satellites

*Remarque : Le module du kit de développement utilisait environ 17-18 satellites. Jusqu'à présent, un maximum de 13 satellites a été utilisé simultanément.*

6. Bon environnement

7. Configuration des modules GNSS

8. Temps de stabilisation

*Remarque : Le module du kit de développement se place rapidement en mode float. Cependant, il est probable que le capteur ait besoin de plus de temps. Le problème est que la quantité de données consommées est rapidement élevée.*

9. Taille de l'antenne

*Remarque : Le module du kit de développement utilise une grande antenne GNSS fournie avec le kit. Le capteur utilise une plus petite antenne. De plus, le plan de masse de l'antenne du kit est spécialement grand.*

Pour m'assurer de plusieurs points, j'ai effectué un test avec le nouveau capteur recevant des données de correction de l'antenne NTRIP de Pailly. Ce test a été effectué au même lieu des tests pour la validation du concept RTK avec le kit de développement. Un lieu spécialement ouvert.

Ces tests furent très bénéfiques. En effet, le mode float a pu être observé de manière relativement stable. Encore mieux, le mode fixe a pu être aperçu très brièvement. Cependant, la précision récupérée du module GNSS indiquait au mieux une précision horizontale de 50 cm et verticale de 60cm. Ces précisions sont correctes mais sont loin des 8 centimètres recherchés.

Après ces tests, j'ai pu conclure plusieurs remarques :

1. J'ai effectué des tests en envoyant de gros paquets (> taille tampon uart) au capteur via LTE afin de voir comment sont gérés les grandes quantités de données. J'ai pu observer que toutes les données ne sont pas reçues. Malgré cela, le flag RTK est actif de manière stable ainsi que le mode float dans un bon environnement. Cela prouve que beaucoup de données transmises par l'antenne NTRIP de Pailly ne sont pas nécessaires dans notre cas.
2. Le fait que le flag RTK est actif, les données sont bien compatibles. Cependant, malgré la perte de données, le mode RTK est tout de même stable. Cela prouve qu'une quantité de données n'est pas requise.
3. Il est recommandé de configurer le module avec un temps maximum de 60 secondes. D'après mes tests, les données arrivent à destination beaucoup plus rapidement qu'une minute.

4. Etant donné que le test a encore été effectué avec l'antenne de Pailly, la distance est restée relativement grande. Malgré ceci, les résultats des tests ont été améliorés.
5. Malgré un environnement similaire aux tests effectués avec le kit de développement, un maximum de 15 satellites a pu être utilisé contrairement à 20 au maximum avec les modules du kit. Je pense que la taille de l'antenne GNSS joue un grand rôle pour ceci.
6. Comme attendu, j'ai pu observer qu'un bon environnement aidait grandement afin d'atteindre un mode float.
8. Dans les tests effectués précédemment, le mode float est apparu plutôt rapidement. Cependant, la fois où le mode est passé en fixe, il a tout de même dû patienter un moment.
9. Malgré la plus petite taille de l'antenne, le mode float a pu être aperçu de manière stable. Ceci est une bonne nouvelle car l'antenne ne peut pas vraiment être plus grande sur le capteur fourni.

Toutefois, une chose me préoccupe encore un peu. En effet, lors du test similaire mais avec le module du kit de développement, la précision horizontale et verticale était en dessous de 10 cm. Or, dans ce dernier test, presque tous semblent être dans le même état, cependant, la précision était toujours en dessus de 50 cm. Voici quelque facteur qui pourrait en être la cause :

1. Perte de données
2. Temps de stabilisation
3. Taille de l'antenne

Afin de réduire grandement la quantité de bytes à recevoir par le capteur ainsi que pour tester plus exactement les différents types de messages nécessaires au bon fonctionnement, j'ai pensé que ça serait pratique de pouvoir filtrer les différents messages de l'antenne de Pailly. Il se trouve que cette fonctionnalité est disponible sur le logiciel SNIP. Le nombre de byte par seconde est passé d'environ 1500 bytes/sec à 600 bytes/sec. Maintenant je suis sûr que toutes les données sont correctement réceptionnées et envoyées au module GNSS car le tampon est assez grand.

De plus, j'ai pu effectuer des recherches approfondies sur les différents types de messages RTCM (RTK / MSM). J'ai fait des tests sur l'efficacité des différentes combinaisons possibles proposées par l'antenne de Pailly.

RTK	: correction non-actif	=> accH = 6 m
MSM	: correction actif (float)	=> accH = 2m (1127/1087/1077)
RTK + MSM	: correction actif	=> 3.5 m (déconseillé dans datasheet)

Des erreurs de connexion ont été constatées même avec le logiciel u-center.

Après cette analyse, j'ai pu apprendre que les messages MSM sont plus efficaces que les autres possibilités.

Maintenant que la perte de données ainsi que la nécessité des bonnes données ont été corrigées, je vais refaire des tests avec le capteur et un module du kit de développement afin de comparer la qualité des signaux et la précision obtenue. De plus, je vais essayer non seulement avec l'antenne de Pailly mais aussi avec une station de référence réalisée par l'autre module du kit afin de réduire la distance entre la station de base et le rover.

Les tests n'ont pas pu être effectués car j'utilise internet (4G) de mon natel. Dans ce cas, il n'est pas possible de faire des redirections de PORT. Donc, il n'est pas possible d'atteindre le serveur de mon ordinateur depuis un appareil dans le réseau public.

Matinée à l'école pour avoir des retours sur le rapport, faire une demande d'accès aux données de corrections de l'antenne de la HEIG-VD et faire une demande pour faire une redirection de PORT dans le réseau de l'école.

Corrections et avancement du rapport.

Modification du serveur personnel afin de gérer la connexion et le retransmission des données avec une station de base et plusieurs rovers. Modifications afin que le serveur tourne sans arrêt. L'objectif est de laisser mon PC allumé à la maison et pouvoir effectuer des tests à l'extérieur avec les capteurs.

Modification du logiciel embarquée en mode station de référence afin de transmettre correctement les données de correction au serveur personnel.

### Problèmes rencontrés

Des tests n'ont pas pu être effectués car j'utilise internet (4G) de mon natel. Dans ce cas, il n'est pas possible de faire des redirections de PORT. De ce fait, il n'est pas possible d'atteindre le serveur de mon ordinateur depuis un appareil dans le réseau public.

### Solutions

Encore aucun solution n'a pu être trouvée. Cependant, une demande à l'école a été faite afin de faire une redirection de port dans le réseau de l'école. Une fois ceci fait, il sera peut-être possible de faire des tests dans les champs près de l'école. Sinon, peut-être qu'en passant par le VPN de l'école, il sera possible de recevoir des requêtes de l'extérieur avec le partage de connexion de mon smartphone.

### Remarques

Cette semaine j'ai profité du journal de travail afin de faire un rapide suivi des différentes remarques et conclusions de mes tests. Ce suivi ne devrait peut-être pas être dans ce document. Cependant, je le trouve très utile et pertinent afin d'avoir un suivi des réflexions sur le projet.

## Semaine N°15 : 06 juillet – 12 juillet

### Travaux effectués

Nouveau test de fonctionnement entre rover et base sur le matériel d'ASI et avec l'antenne de l'HEIG. Cette fois une précision horizontale de 5cm a été obtenue. Cependant, aucune différence n'a été effectuée comparer au dernier test. Donc la stabilité n'est pas irréprochable. Le mode fixe a pu être aperçu. Lorsque le mode float disparaît un court instant, la précision est perdue (~0.5 m) et il faut un certain temps avant qu'elle revienne aux alentours de 8 cm.

Réception d'informations complémentaires sur l'antenne RTK de la HEIG.

Séance avec M. Mosqueron pour parler des résultats des tests précédents

Tests et implémentation d'abonnement sur l'antenne de la HEIG avec les identifiants reçus.

Tests à l'extérieur avec l'antenne de la HEIG-VD. Très bon résultat mais pas de mode fixe aperçu.

Correction du rapport et du journal de travail.

Correction des unités dans les affichages.

Gestion de la déconnexion de mon serveur personnel avec les modules en mode Base et rover.

Calcul et affichage de la distance relative à la station de référence.

Recherche et comparaison des différentes types de messages envoyés par les antennes RTK.

([https://docs.novatel.com/OEM7/Content/Logs/RTCMV3\\_Standard\\_Logs.htm?TocPath=Logs%7CAII%20Logs%7CGNSS%20Logs%7C\\_155](https://docs.novatel.com/OEM7/Content/Logs/RTCMV3_Standard_Logs.htm?TocPath=Logs%7CAII%20Logs%7CGNSS%20Logs%7C_155))

Nouveau test de fonctionnement entre rover et base sur le matériel d'ASI et avec l'antenne de l'HEIG. Encore une fois, des bons résultats ont été aperçus. Cependant, il arrive que la réception de donnée s'arrête.

Test sur le temps de stabilisation.

Discussion avec M. Mosqueron et M. Messerli. La précision et la position indiquées par le module GNSS devrait être testée. Entretiens avec une équipe de géomètre de la HEIG-VD (Bron Maximin, Gressin Adrien, Guillaume Sébastien, Benjamin Mignot) afin d'avoir accès à plusieurs points et leurs localisation précise au centimètre.

Première mesure à 14h20 sur différents points. Création et remplissage d'un tableau Excel pour la récolte des données.

Recherche et mise en place d'un calcul de distance précis entre deux points en latitude et longitude.

Deuxième mesure à 14h20 sur différents points. Remplissage du tableau Excel pour la récolte des données.

Recherche sur les différentes valeurs à analyser pour les tests de précision.

Communication avec Lionel afin de mettre en place un système de démonstration pour un client.

Installation et mise en place de la VM Windows sur le serveur d'ASI. Portage de mon application serveur sur la VM fourni. Résolution des problèmes de librairie.

Création d'une application serveur pour transmettre les données de correction RTK de l'antenne de la HEIG-VD sur des modules. Plus enregistrement des données des rovers dans des fichiers sur le serveur.

Troisième mesure à 14h20 sur différents points. Remplissage du tableau Excel pour la récolte des données.

### Problèmes rencontrés

### Solution

### Remarques

Lorsque j'approche ma main du capteur, les précisions indiquées se dégradent. Une fois la précision dégradée, il faut attendre un petit moment avant qu'elle revienne à quelques centimètres.

## Semaine N°16 : 13 – 19 Juillet

### Travaux effectués

Implémentation de l'application serveur pour la démo.

Création d'un fichier KML pour les rovers connectées au serveur.

Adaptation du logiciel embarqué pour être compatible avec la nouvelle application serveur.

Connection correcte + envoie des données de position.

Tests du serveur en extérieur.

Quatrième mesure à 14h20 sur différents points. Remplissage du tableau Excel pour la récolte des données.

Implémentation d'une application serveur pour le fonctionnement avec une station de référence avec un module d'ASI.

Implémentation d'une application client afin d'utiliser un module du kit de développement comme station de référence et d'envoyer ces données de correction à l'application serveur.

Récole et mise en forme des données pour les analyses de temps de stabilisation.

Démonstration du fonctionnement du RTK avec M. Mosqueron et retour sur le rapport intermédiaire.

Test sur des vignes à Champagne.

Mise à jour du calcul de distance relative à l'antenne de référence sur le logiciel embarqué.

Recherche pour l'utilisation d'une librairie c++ pour utiliser le protocole MQTT.

### Problèmes rencontrés

L'utilisation des librairies MQTT ne fonctionne pas.

### Solution

### Remarques

## Semaine N°17 : 20 – 26 Juillet

### Travaux effectués

Récolte des données effectuées lors de tests à Genève.

Réunion avec M. Mosqueron et test des modules avec et sans RTK en voiture.

Test de dérive + Test avec règle.

Recherche pour une utilisation de MathLaB avec les données Latitude et Longitude.

Corrections et restructuration du rapport

Modification du logiciel embarqué pour gérer une fréquence de 10 Hz pour le test à Zermatt.

Modification des applications serveurs pour gérer la réception des messages des rovers à 10 Hz.

Test à l'extérieur avec une fréquence de 10 Hz.

Modification du logiciel de la station de référence afin d'ajouter la fonctionnalité de commencer le mode station de base une fois que le module GNSS est stable.

Implémentation d'un nouveau logiciel embarqué pour assembler le code d'un rover avec celui d'une station de référence et être compatible avec le logiciel d'ASI.

### Problèmes rencontrés

1. Il n'y a pas réellement 10 messages par seconde. Le nombre de message reçu par seconde varie entre 1 à 6.
2. Lors de l'ajout d'un tableau pour la configuration du module GNSS, le programme plainte quelques fois.

### Solution

1. Il a fallu descendre la fréquence de réception des messages moins intéressant (NAV\_STATUS et NAV\_RELPOSNED) afin de mettre la priorité sur les messages NAV\_PVT. De plus il a fallu régler correctement la fréquence d'appel à la tâche GNSS (Période = 1000 ms / NB de message/s).
2. C'est un problème car la stack de la tâche interface est trop petite. Il n'est pas possible de l'augmenter car le programme ne dispose pas assez de place mémoire. J'ai donc déplacé le tableau dans la partie BSS et non plus la stack.

### Remarques

J'ai perdu beaucoup de temps cette semaine car le connecteur de la batterie s'enlevait automatiquement et je pensais que le problème venait de la place mémoire.

## Semaine N°18 : 27 Juillet – 2 août

### Travaux effectués

Implémentation d'un nouveau logiciel embarqué pour assembler le code d'un rover avec celui d'une station de référence et être compatible avec le logiciel d'ASI.

Ajout de la fonctionnalité d'envoyer la position fixe de la station de référence.

Refactor des code des applications serveurs.

Ajout de la fonctionnalité d'envoyer les informations de la procédure de recherche de position d'une station de base au serveur.

Recherche approfondie sur la stabilité de la réception et de l'envoie des messages GNSS :

1. L'appel plus fréquent à la machine d'état du bg96 n'est pas concluant.
2. Les tests permettant de savoir si le modem LTE est prêt ont été améliorés => la fréquence d'envoi des messages GNSS est plus stable à 1Hz.
3. La fréquence d'appel à la tache GNSS doit être correctement définie, sinon, des messages venant du module GNSS est perdu.

Maintenant un rover réglé à 1Hz semble stable. De plus, la fermeture du socket semble se faire à chaque fois. Des tests différents à l'extérieur doivent être effectués afin de s'assurer du bon fonctionnement du rover à 1Hz.

Implémentation d'un analyseur RTCM3 côté serveur.

Implémentation d'un filtre RTCM3 côté serveur.

Test du filtre RTCM3 avec l'antenne de Pailly. Le mode RTK ne s'active pas. Correction du filtre RTCM3 et confirmation de fonctionnement avec SNIP.

Le mode RTK ne s'active toujours pas. Recherche approfondie avec le Kit RTK et le filtre RTCM3 afin de connaître toutes les combinaisons fonctionnelles des messages RTCM3 de l'antenne de l'HEIG et Pailly.

### Problèmes rencontrés

1. Rarement le logiciel plante et l'écran n'affiche plus rien de sensé.

### Solution

1. Rarement, des paquets plus grands que 512 sont reçus via la connexion TCP. Cependant, les buffers réceptionnant les données sont plus petits que 512. Des tests afin de limiter la taille des données traitées ont été ajoutés.

### Remarques

## Semaine N°19 : 3 – 9 août

### Travails effectués

Diagramme grossier du logiciel en mode rover.

Diagramme grossier du logiciel en mode base.

Schéma expliquant l'utilité de l'application serveur.

Diagramme des applications serveur.

Diagramme des threads des applications serveur.

Schémas des interactions dans un système RTK complet.

Recherche du point de référence de géolocalisation précis de l'antenne et mesure de la différence avec le point de référence utilisé lors des mesures.

Recherche pour afficher sur un mini map l'alignement de plusieurs points.

Mise en page des tableaux des données de mesure.

Avancement du rapport.

### Problèmes rencontrés

Solution

Remarques

## Semaine N°20 : 10 – 16 août

### Travaux effectués

Mise au propre des fichiers des applications serveurs et construction des fichiers annexes.

Avancement et corrections du rapport.

Tests et analyse de la dérive.

Finition du rapport.

Construction des annexes.

Résumé et affiche.

### Problèmes rencontrés

Solution

Remarques