logo.png

Valentin Pouce – Jan Barillec – Antonin Lyaët – Alrick Giry

coach : M. Vincent grimaud

2019-2020

Automatisation du placement d’une parabole et d’une antenne râteau pour la station spatiale internationale ISS

D3R3 – Groupe 303

logo.png

Valentin Pouce – Jan Barillec – Antonin Lyaët – Alrick Giry

coach : M. Vincent grimaud

2019-2020

Automatisation du placement d’une parabole et d’une antenne râteau pour la station spatiale internationale ISS

D3R3 – Groupe 303

Table des matières

[1 Introduction 7](#_Toc26426522)

[2 Diagramme de Gantt 8](#_Toc26426523)

[3 Carte mentale 9](#_Toc26426524)

[4 Budget 10](#_Toc26426525)

[5 Projet 11](#_Toc26426526)

[5.1 Matériel 11](#_Toc26426527)

[5.1.1 Le vérin 11](#_Toc26426528)

[5.1.2 Le rotor 12](#_Toc26426529)

[5.1.3 Le modem Tooway 13](#_Toc26426530)

[5.1.4 GPS 13](#_Toc26426531)

[5.1.5 Raspberry Pi 14](#_Toc26426532)

[5.2 Programme 15](#_Toc26426533)

[5.2.1 Classe Rotor 15](#_Toc26426534)

[5.2.2 Classe Vérin 16](#_Toc26426535)

[5.2.3 Classe GPS 17](#_Toc26426536)

[6 Notre travail 18](#_Toc26426537)

[6.1 Correction et Amélioration 18](#_Toc26426538)

[6.1.1 L’IHM 18](#_Toc26426539)

[6.2 Asservissement 20](#_Toc26426540)

[6.2.1 Partie théorique 20](#_Toc26426541)

[6.2.2 Partie pratique 22](#_Toc26426542)

[6.3 Documentation 24](#_Toc26426543)

[6.4 Communication avec l’ISS 24](#_Toc26426544)

[7 Conclusion 26](#_Toc26426545)

[8 Résumé 27](#_Toc26426546)

[9 Mots clés 28](#_Toc26426547)

[10 Tables des illustrations 29](#_Toc26426548)

[11 Annexes 30](#_Toc26426549)

[11.1 Protocole Ethernet (TCP/IP) : Câblage, trame, adressage et communication | Valentin Pouce 30](#_Toc26426550)

[11.1.1 Introduction 30](#_Toc26426551)

[11.1.2 Indentification 30](#_Toc26426552)

[11.1.3 Evaluation du contenu 30](#_Toc26426553)

[11.1.4 Liste des mots clés 30](#_Toc26426554)

[11.1.5 Compte rendu 31](#_Toc26426555)

[11.1.6 Avis critique du document 33](#_Toc26426556)

[11.2 Les différents écrans tactiles | Alrick Giry 34](#_Toc26426557)

[11.2.1 Identification 34](#_Toc26426558)

[11.2.2 Évaluation du contenu 34](#_Toc26426559)

[11.2.3 Liste des mots clefs pour ce document 34](#_Toc26426560)

[11.2.4 Compte rendu 34](#_Toc26426561)

[11.2.5 Pertinence des informations 35](#_Toc26426562)

[11.2.6 Aperçu du document 35](#_Toc26426563)

[11.3 Annexe 2 | Python | Jan BARILLEC 36](#_Toc26426564)

[11.3.1 Identification 36](#_Toc26426565)

[11.3.2 Liste des mots clés 36](#_Toc26426566)

[11.3.3 Compte rendu 36](#_Toc26426567)

[11.3.4 Évaluation du contenu 37](#_Toc26426568)

[11.3.5 Pertinence des informations 37](#_Toc26426569)

[11.3.6 Illustration du livre étudié 37](#_Toc26426570)

[11.1 Les technologies de géolocalisation 38](#_Toc26426571)

[11.1.1 Introduction 38](#_Toc26426572)

[11.1.2 Identification 38](#_Toc26426573)

[11.1.3 Évaluation du contenu 38](#_Toc26426574)

[11.1.4 Liste des mots clés 38](#_Toc26426575)

[11.1.5 Compte rendu 38](#_Toc26426576)

[11.1.6 Avis critique du document 40](#_Toc26426577)

# Introduction

Dans le cadre de notre projet de deuxième année, nous reprenons un projet déjà travaillé en collaboration avec Christophe Taillez Strategic Telecom Sécurité Civile.

L’objectif premier de ce projet est la création d’un système permettant le pilotage d’une antenne parabole, placée sur un camion, de façon à diriger l’antenne vers le satellite géostationnaire EUTELSAT 9A. Le système doit pour interagir par le biais d’un écran tactile codée en Python à l’aide d’un Raspberry Pi avec la parabole et doit présenter deux mode de contrôle, le mode Manuel et mode Automatique

Le but de cette année est de conclure cette partie en élaborant un algorithme permettant l’asservissement de l’orientation de la parabole, autrement dit la création du mode automatique permettant l’orientation autonome de la parabole ainsi qu’une documentation technique permettant une compréhension claire et précise du fonctionnement du projet.

Ce projet est doté d’une seconde partie, qui consiste la conception d’un système permettant la communication par satellite avec l’ISS (International Space Station) qui servira à des écoliers ou radioamateurs pour entrer en contact avec la station. Cette partie sera étudié d’avantage suite ç l’aboutissement de la première partie

Pour la réalisation de ce projet, notre groupe, le groupe 303 est composé de :

* Valentin Pouce
* Alrick Giry
* Antonin Lyaët
* Jan Barillec

Par la supervision de M. GRIMAUD Vincent.

Dans ce rapport, nous allons tout d’abord expliquer et définir l’existant du projet quand nous l’avions commencé, afin de voir ce qui a déjà été pensé et définir les points à améliorer ou bien à faire. Ensuite, dans une seconde partie, nous allons présenter ce que nous avons fait mais aussi ce que nous allons effectuer afin de clore au ce projet dans le temps imposé.

# Diagramme de Gantt

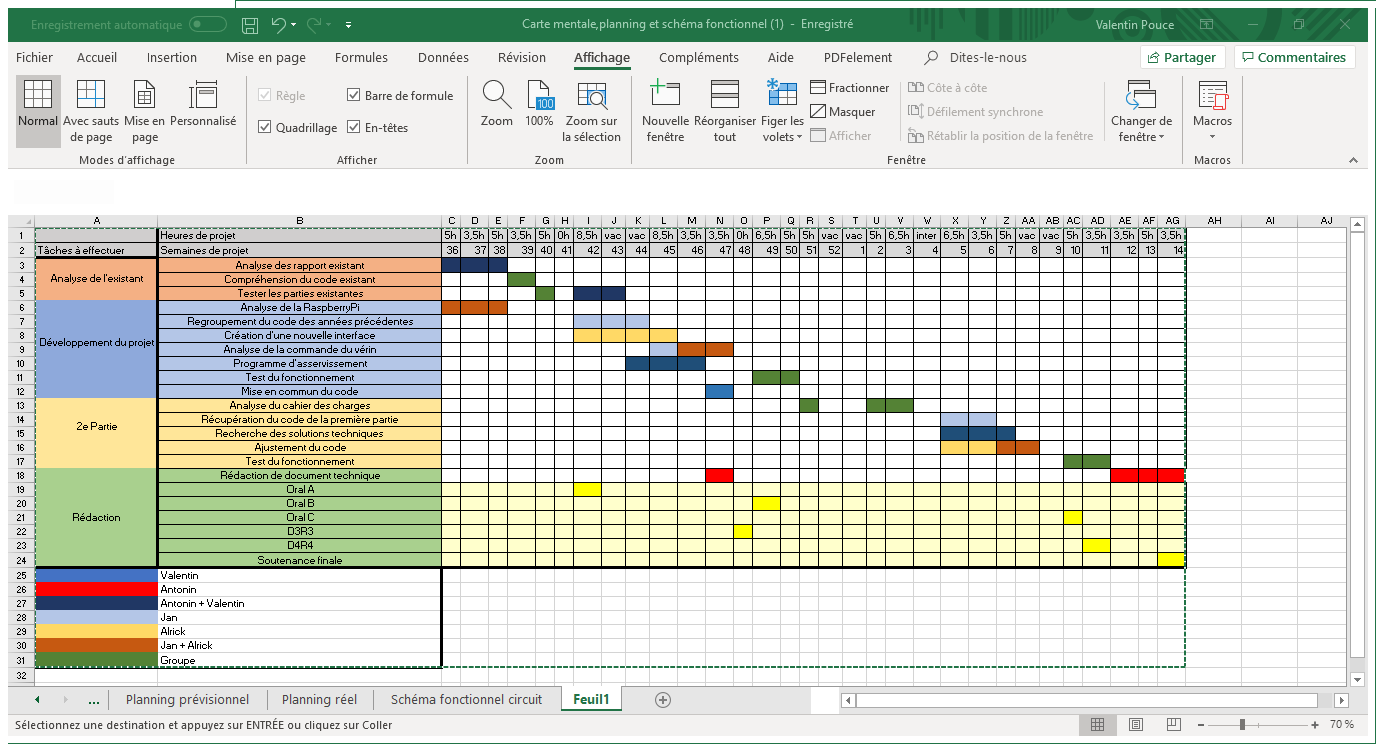
Le planning de ce projet est réparti en deux parties qui représentent les parties du projet, c’est-à-direle pilotage de la parabole jusqu’à mi-décembre puis dans un deuxième temps, l’antenne râteau avec l’ISS.

Figure Diagramme de Gantt

# Carte mentale

Notre projet se déroule en quatre grandes parties pour arriver à un but, l’automatisation du placement de la parabole. Une première partie qui consiste à l’étude de l’existant, comprendre le système, son fonctionnement, les algorithmes et le langage de programmation qui est aussi nouveau pour l’ensemble du groupe. Une deuxième et une troisième parties qui vont être la programmation, l’amélioration et l’ajout de nouvelle fonctionnalités. Puis la dernière, partie cruciale, qui est les tests avec quoi sans ça, nous ne pourrions valider notre travail.

Figure Carte Mentale du projet

# Budget

La totalité du matériel nous a été fournie par Strategic Telecom, les précédents groupes qui ont travaillé sur le projet (2 au total) ont acheté le nécessaire. Cependant du matériel qui a été acheté et nous est inutile comme la Boussole CMP12.

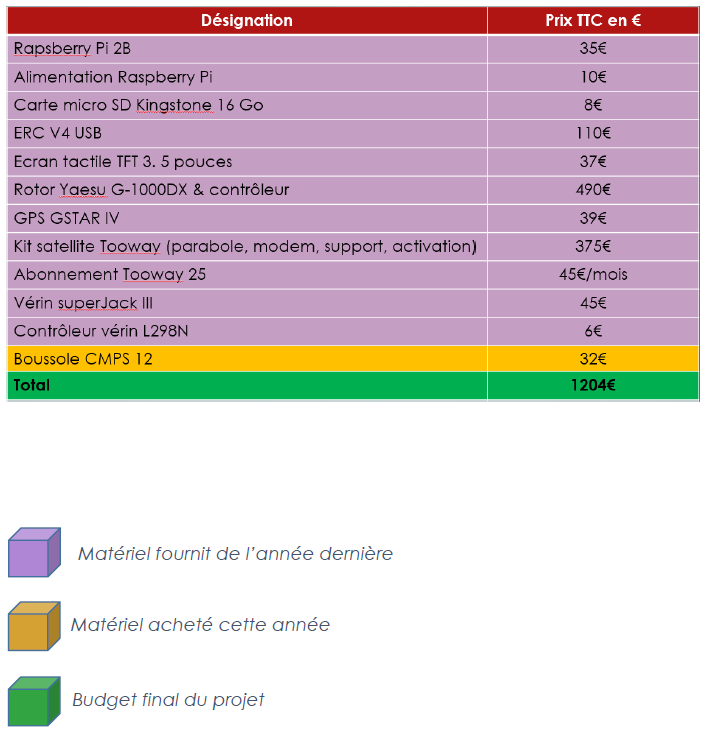


Figure Budget du projet

# Projet

Avant de nous plonger dans notre avancement sur le projet, faisons un point sur le projet et l’existant, c’est-à-dire étudier le choix des composants, du matériel et du programme provenant des années précédentes.

Une image contenant intérieur, carte, texte

Description générée automatiquement

Figure Schéma de la structure globale du système

## Matériel

### Le vérin

Le choix du vérin a été fait sur deux critères par le groupe de l’année 2016-2017, c’est-à-dire :

* Avec une Parabole d’environ 15 kilogrammes, le vérin se doit d’avoir une poussée au minimum équivalente afin de pouvoir l’orienter convenablement.
* Le vérin est installé sur le camion, alors la seule alimentation possible est le 24 V des batteries présentes dans le camion.

Le vérin qui a été retenu est le Super Jack III de chez Jaeger avec les caractéristiques techniques suivantes :

* Tailles : 12 pouces
* Charge statique : 225 kg
* Charge dynamique : 135 kg
* Alimentation : 36 V DC
* Précision du capteur : 76 impulsions par pouce
* Température de fonctionnement : -30°C à 50°C

D’après les informations qui nous ont été fournis par les rapports des années précédentes, le vérin prévu avec une alimentation de 36V DC fonctionne correctement avec une tension inférieure qui est celle du camion, c’est-à-dire 24V DC, deux batteries de 12V misent en série pour atteindre 24V.

Cependant, ce vérin nécessite une carte d’interface afin d’interagir avec pour un contrôle plus aisé.

#### Carte d’interface

Une image contenant équipement électronique, circuit

Description générée automatiquementAfin de piloter le moteur du vérin, nous avons besoin d’une carte d’interface capable de délivrer la tension d’alimentation nécessaire pour le bon fonctionnement des moteurs. La Raspberry Pi n’étant pas capable de fournir une telle tension, il en est alors plus que nécessaire. Pour accomplir cette tâche, il s’agit du module L298N qui a été sélectionnée par les groupes des années précédentes pour sa compatibilité avec la Raspberry et les 24V de l’alimentation.

Figure Carte d'interface L298N

Une image contenant capture d’écran

Description générée automatiquement

**GND**

**24V**

Figure Schéma cablâge du vérin

Cette carte permet également le pilotage du moteur dans les deux sens de rotation par l’intermédiaire des entrées IN1 et IN2.

### Le rotor

Pour la rotation de la parabole en azimut, nous avons besoin d’un rotor. Ce rotor a déjà été choisi par le premier groupe. Il s’agit du YASEU G-1000DS.

Ce rotor a pour caractéristiques :

* Charge statique : 6000kg/cm
* Charge dynamique : 600 à 1100 kg/cm
* Temps d’une rotation à 360 ° : 43 à 93 secondes
* Diamètre : 186 mm
* Hauteur : 300 mm
* Charge verticale : 200 kg

La Raspberry Pi communique avec le rotor par le biais d’un ERC Mini qui permet une liaison entre la Raspberry et le contrôleur du rotor.

### Le modem Tooway



Figure Parabole et module Tooway

Un module Tooway est installé dans le camion et permet de recevoir des données internet par satellite à très haut débit.

Le signal est capté par une parabole qui est montée sur le toit du camion. Il est aussi possible d’envoyer des données par le même module.

Ce module est développé par les sociétés Eutelsat et ViaSat.



Figure Zones de récupération du signal

Il est possible de récupérer des données sur l’entièreté de l’Europe grâce à un abonnement.

Les données sont ensuite transférées par une liaison Ethernet à la Raspberry Pi.

Afin de pouvoir positionner la parabole, le programme récupère la position du camion et grâce à celles-ci et à un site internet, il est possible de connaitre l’élévation et l’azimuthe à lui donner.

### GPS

Le module GPS GSTAR IV a été fourni par la sécurité civile, il permet de géolocaliser le camion.

Le module est connecté par USB à la Raspberry Pi 3 et permet la réception de trames qui contiennent la latitude, longitude et l’altitude, ainsi que l’heure. Les trames suivent le protocole standardisé NMEA 0183 (National Electronics Association). Les différentes trames reçues fournissent chacune des informations différentes.

Dans notre cas, nous utilisons la trame GGA qui fournit l’heure, la latitude et longitude, l’altitude et le nombre de satellites trouvés.



Figure GPS GSTAR IV

Les groupes des années précédentes ont créé une classe GPS qui permet de lire directement la valeur de la latitude et longitude de la trame GGA ainsi que de déterminer l’azimut et l’élévation que la parabole doit avoir pour recevoir correctement le signal du satellite.

### Raspberry Pi

Pour relier et faire cohabiter tous les éléments du projet il faut un dispositif qui permette d’accueillir des périphériques USB, un écran tactile, des ports GPIO (General Purpose Input Output).

Figure Raspberry Pi 3

Les groupes qui ont travaillés sur le projet précédemment ont fait le choix d’utiliser un Raspberry Pi 3, car il s’agit d’un système linux embarqué peu onéreux et compact, mais aussi puissant.

L’écran tactile permettras de réaliser l’IHM, les ports GPIO permettrons de contrôler le vérin et le port USB serviras à recevoir les trames.

## Programme

Pour utiliser ce matériel, il a fallu mettre en place différents algorithme afin d’utiliser finement ce dont on avait à disposition. La programmation est la partie clé de ce projet, c’est elle qui détermine le fonctionnement de ce système. Pour cela, les groupes des deux années précédentes élaborer quelques programmes qui permettent le fonctionnement du GPS, du rotor ou bien du vérin.

### Classe Rotor

Figure Classe Rotor

Cette classe permet de piloter le rotor. Chaque fonction à un rôle bien définie qui envoie une trame au rotor en fonction de ce que l’on veut qu’il fasse.

Liste des fonctions :

* « \_\_init\_\_(self, chemin) : créer un objet Rotor, c’est le constructeur. Si le rotor n’est pas connecté à la Raspberry, elle envoie un message d’erreur ;
* « tourner(self, angle) : permet de faire tourner le rotor jusqu’à un certains angle ;
* « angle(self) » : permet de lire l’angle à sur lequel le rotor est actuellement ;
* « tournerHoraire(self) » : fait tourner le rotor dans le sens horaire jusqu’à qu’on lui dise de se stopper ;
* « tournerAntiHoraire(self) » : fait tourner le rotor dans le sens anti-horaire jusqu’à qu’on lui dise de se stopper ;
* « stop(self) » : stop la rotation du rotor ;
* « config(self) » : calibre et configure le rotor. Cette fonction est appelée dans le constructeur.

### Classe Vérin

Figure Classe Vérin

Cette classe permet de piloter le vérin. Chaque fonction à un rôle bien définie qui envoie une trame au vérin en fonction de ce que l’on veut qu’il fasse.

Liste des fonctions :

* « \_\_int\_\_(self, pinPush, pinPull, pinInterrupt) » : Constructeur du vérin qui définit les pin pour monter ou descendre ainsi qu’un pin d’interruption ;
* « impulsion(self, channel) » : fonction de débogage. Écrit le nombre d’impulsion envoyé au vérin ;
* « monter(self) » : Changer les pins actifs et inactifs pour faire monter le vérin ;
* « descendre(self) » : Changer les pins actifs et inactifs pour faire descendre le vérin ;
* « arreter(self) » : arrête le mouvement du vérin. Mets donc les deux pins en inactif

### Classe GPS

La classe GPS permet de faciliter l’utilisation du module GPS, afin d’y récupérer les valeurs essentielles aux placements de la parabole.

Deux processus sont à l’œuvre dans la classe GPS, la lecture de la trame GGA et la conversion de la latitude et de la longitude en azimut et Elévation.

Figure Schéma fonctionnel de la Classe GPS

Trames GPS

CLASSE GPS

Conversion

Latitude

Longitude

Longitude

Azimut

Lecture

GGA

Latitude

Elévation

La lecture de trame GGA se fait à l’aide de la bibliothèque **pynmea2,** qui permet d’extraire les variables Latitude et Longitude de la trame. La conversion est réalisée à l’aide de relations trigonométrique qui ont été déduite par le groupe de l’année dernière.

# Notre travail

//Intro à faire

## Correction et Amélioration

### L’IHM

L’IHM ou l’Interface Homme Machine est un outil qui permet à l’utilisateur d’interagir avec le système qu’il souhaite utiliser. Dans notre cas, nous voulons pouvoir contrôler l’élévation et l’azimut de la parabole. Il faut aussi pouvoir activer le mode placement automatique et rentrer l’orientation du camion par rapport au nord magnétique. L’écran devra afficher les coordonnées GPS les valeurs d’inclinaison de la parabole et la puissance du signal.



Nous repartirons avec la base de la première IHM car celle-ci est plus ergonomique. Toute l’interface a été codé en python avec la bibliothèque Pygame pour un écran tactile LCD de dimension 320x480.

Figure Écran utilisé (320x480)



Figure Logo de Pygame

Figure Logo de Python

En entrée utilisateur nous avons 4 boutons de direction et un bouton « auto » et un pavé numérique pour rentrer la valeur du nord magnétique.

L’IHM devra appeler les fonctions :

* rotor.tournerAntihoraire() / rotor.tournerHoraire()
* verin.monter() / verin.descendre()
* rotor.stop()
* verin.arreter
* positionnementauto()

Nous pourrons envisager de faire des menus pour visualiser les valeurs et en rentrer d’autres afin d’avoir une interface plus claire et lisible.



Figure IHM 2017

Ceci est la première interface réalisée pour ce projet (2017)



Figure IHM 2018

Ceci est la deuxième version de l’interface (2018)

Certaines fonctions avaient du mal à fonctionner nous les avons donc corrigés.

L’interface n’étant pas très pratique, ni ergonomique, nous avons décidé de remanier les éléments et changer les couleurs.

Voici donc la dernière interface à ce jour :

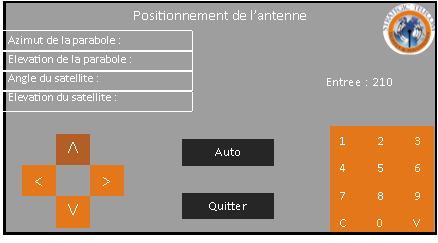


Figure Esquisse IHM 2019

Il faudra surement rajouter des informations sur l’IHM mais sa forme globale restera la même.

Pour paramétrer l’écran tactile nous avons suivi le tutoriel du constructeur.

## Asservissement

Dans notre projet, nous devons mettre en place un mode automatique de la parabole pour qu’elle se calibre sans intervention humaine sur le satellite de communication voulue à un degré près. Pour effectuer ce mode automatique, on a dû mettre en place un algorithme qui permet à la parabole si elle doit monter ou descendre et aller à gauche ou à droite.

Pour effectuer un asservissement correct, on a dû mettre en place un algorithme permettant savoir par où le satellite est en fonction de la puissance du signal qu’on reçoit.

### Partie théorique

Pour simplifier le code et l’algorithme, nous avons choisis de faire fonctionner le rotor et le vérin chacun leurs tours dans une boucle.

Grâce à la fonction « getData » créée le groupe d’il y a deux ans que l’on expliquera dans la partie pratique, on peut récupérer la puissance et le bruit que reçoit la parabole. On vient donc récupérer la puissance du signal à chaque fois que la boucle recommence et enregistrer l’ancienne puissance pour comparer.

En étudiant ce que devait faire l’asservissement, nous nous sommes retrouvés face à 4 cas lorsque le rotor fonctionne et 4 cas lorsque le vérin fonctionne.

Voici les différents cas :

#### Différents cas pour le rotor



Voici les 4 cas pour le rotor.

-Le rond rouge correspond à l’ancienne puissance ;

-Le rond bleu correspond à la nouvelle puissance ;

-La flèche correspond au sens de rotation du rotor ;

-Le rond avec un H à une plus grande puissance que le rond avec un B en dessous.

Si on prend cas par cas :

Dans le cas numéro 1, on voit que la puissance de la nouvelle mesure est plus puissante que celle de l’ancienne mesure. Le rotor part à l’opposé de l’ancienne mesure donc le rotor tourne dans le bon sens.

Dans le cas numéro 2, on voit que la puissance de la nouvelle mesure est moins puissante que celle de l’ancienne mesure. On en déduit donc que l’on a soit passé la puissance maximale soit on est parti dans le mauvais sens au début. On fait donc un changement de sens, le rotor va donc tourner vers l’ancienne mesure.

Dans les cas numéro 3 et 4, ils sont respectivement identiques au cas numéro 2 et 1 mais dans l’autre sens.

Avec ce fonctionnement, le rotor va faire un changement de sens lors du démarrage s’il est parti dans le mauvais sens puis en fera une infinité lorsqu’il s’approchera de la puissance maximale si on ne l’arrête.

#### Différent cas pour le vérin

Une image contenant capture d’écran, graphiques vectoriels

Description générée automatiquement

Voici les 4 cas pour le rotor.

-Le rond rouge correspond à l’ancienne puissance ;

-Le rond bleu correspond à la nouvelle puissance ;

-La flèche correspond au sens d’élévation du vérin ;

-Le rond avec un H à une plus grande puissance que le rond avec un B en dessous.

Le fonctionnement de l’asservissement pour le vérin reste le même que celui du rotor mais l’on gère l’élévation de la parabole.

Dans les cas numéros 1 et 3, il n’y a pas de changement de sens car la nouvelle mesure de puissance est plus élevée que la précédente.

Cependant pour les cas numéros 2 et 4, la nouvelle mesure de puissance est plus faible que la précédente donc le vérin change de sens d’élévation

### Partie pratique

Une fois les différents cas différenciés et qu’on a réfléchis au fonctionnement de l’algorithme que l’on devait mettre en place. On a pu commencer à coder l’algorithme.

#### La fonction « getData »

La fonction « getData » est une fonction créée par le groupe d’il y a deux ans. Elle permet de récupérer la puissance reçus par la parabole. Cette fonction utilise la page de configuration du modem Tooway où l’on peut retrouver la puissance et le bruit.

Les étudiants d’il y a deux ans avaient réussi à récupérer ces informations via un bibliothèque python « Request ». Cette bibliothèque vient faire une requête HTTP et reçoit une série de valeurs séparées par des « ## ». Par analyse de ces données, ils en ont déduit que la puissance était la 15 valeur (14 dans le tableau) et le bruit la 12 valeur (11 dans le tableau).

Figure Fonction getData()

Cette fonction vient nous renvoyer dans l’ordre : la puissance et le bruit.

#### Le code de l’asservissement

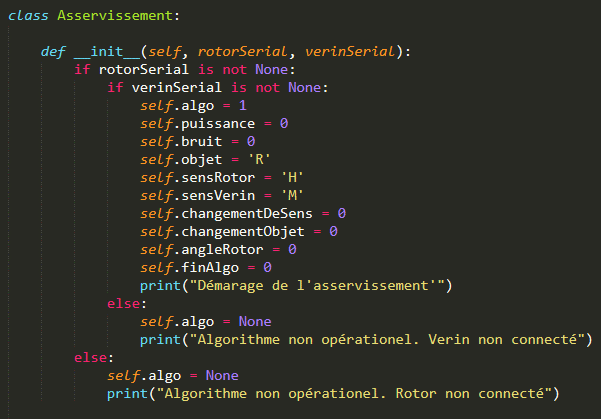
Pour l’asservissement, on choisit d’établir quelques règles pour que l’algorithme ne dure pas trop longtemps mais que la précision soit aussi présente. 3 règles pour l’asservissement ont été mise en place :

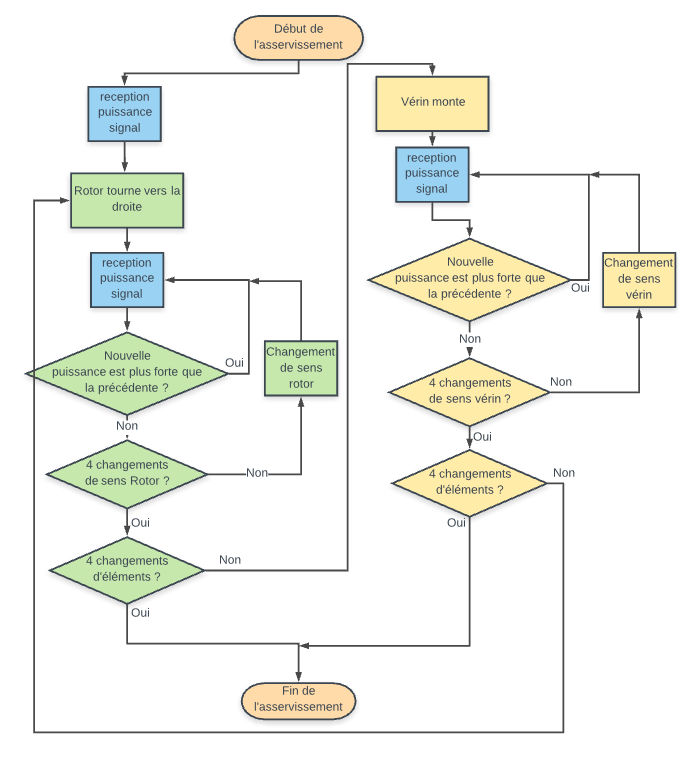
\_Le pilotage du rotor et vérin se fait chacun leur tour ;

\_On change d’élément pilotant la parabole (rotor->vérin ou inversement) lorsque l’objet piloté à fait 4 changements de sens depuis son cycle ;

\_L’asservissement se finis après avoir changé 4 fois d’objets.

Avec ces 3 règles, on s’assure que le rotor et vérin soient bien calibrés pour que la parabole pointe vers le satellite.

Voici l’asservissement :

En résumé, voici un schéma bloc du fonctionnement de l’asservissement :

## Documentation

## Communication avec l’ISS

Ce projet est composé de 2 parties. La première partie consistait à finir l’automatisation d’un système déjà réfléchi par des étudiants d’années précédentes. La seconde partie commencera seulement quand la première partie sera terminer.

Une image contenant satellite, transport, extérieur, clôture

Description générée automatiquementCette partie consiste au début du développement d’une nouvelle application pour quatre antennes râteaux. En effet, ce système devra communiquer par l’intermédiaire de ces antennes, avec la Station Spatiale International, autrement dit l’ISS. Ce projet servira par la suite à des écoles ou bien à des radios amateurs, pour entrer en contact avec la station.

Figure ISS

Pour des raisons techniques et que le cahier des charges le permet, nous pourrons utiliser des algorithmes déjà élaborer dans la première partie afin d’accomplir le projet. On peut prendre pour exemple, le pilotage du vérin qui pourra être similaire.

A l’heure actuelle, cette partie n’est pas encore démarré puisque la première partie du projet en est encore à l’état d’expérimentation.

# Conclusion

# Résumé

# Mots clés

automatisation, 18

azimut, 12, 14, 15, 16

élévation, 13, 14, 16

Ethernet, 5, 13, 23, 24, 25, 26

GGA, 13, 14, 15

GPS, 5, 13, 14, 15, 16, 22

IHM, 5, 14, 16, 17, 22, 28

latitude, 13, 14, 15

parabole, 7, 8, 9, 12, 13, 14, 15, 16

pilotage, 7, 8, 12, 18

python, 16

Raspberry, 5, 7, 12, 13, 14, 22

vérin, 5, 11, 12, 14, 15, 18, 22

# Tables des illustrations

[Figure 1 Diagramme de Gantt 7](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385904)

[Figure 2 Carte Mentale ddu projet 8](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385905)

[Figure 3 Budget du projet 9](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385906)

[Figure 4 Schéma de la structure globale du système 10](#_Toc26385907)

[Figure 5 Carte d'interface L298N 11](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385908)

[Figure 6 Schéma cablâge du vérin 11](#_Toc26385909)

[Figure 7 Parabole et module Tooway 12](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385910)

[Figure 8 Zones de récupération du signal 12](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385911)

[Figure 9 GPS GSTAR IV 13](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385912)

[Figure 10 Raspberry Pi 3 13](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385913)

[Figure 11 Schéma fonctionnel de la Classe GPS 14](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385914)

[Figure 12 Écran utilisé (320x480) 15](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385915)

[Figure 13 Logo de Pygame 15](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385916)

[Figure 14 Logo de Python 15](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385917)

[Figure 15 IHM 2017 16](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385918)

[Figure 16 IHM 2018 16](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385919)

[Figure 17 Esquisse IHM 2019 16](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385920)

[Figure 18 ISS 17](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385921)

[Figure 19 Table d'adresse MAC 23](#_Toc26385922)

[Figure 20 Aperçu document les différents types d'écrans tactiles 27](file:///C:\Users\valen\Desktop\GEII\Projets\PROJET%20S3\ISS-PROJECT\D3R3%20cette%20année%20(1).docx#_Toc26385923)

# Annexes

## Protocole Ethernet (TCP/IP) : Câblage, trame, adressage et communication | Valentin Pouce

### Introduction

Ce document va me permettre d’analyser et de comprendre le fonctionnement de la communication par Ethernet et son protocole. Il retrace l’évolution du l’Ethernet, son fonctionnement et des différentes normes en vigueur concernant cette communication et de ses avantages.

### Indentification

Ce document est un cours en ligne sur le protocole Ethernet et ses spécificités sur le site coursreseaux.com. Ce cours s’intitule Protocole Ethernet (TCP/IP) : Câblage, trame, adressage et communication. Ce cours permet de répondre aux questions Qu’est ce que le protocole Ethernet ? A quoi sert-il ? et aussi Comment fonctionne-t-il ?

Lien : https://coursreseaux.com/protocole-ethernet/

### Evaluation du contenu

|  |  |  |  |
| --- | --- | --- | --- |
| Critères | Faible | Moyen | Bien |
| Fiabilité |  |  |  |
| Illustration |  |  |  |
| Lisibilité |  |  |  |
| Pertinence |  |  |  |
| Accessibilité |  |  |  |

### Liste des mots clés

|  |  |
| --- | --- |
| Protocole | Ethernet |
| Communication | Adressage |
| Trame[[1]](#footnote-1) | TCP[[2]](#footnote-2) |
| IP[[3]](#footnote-3) | Norme |
| Mbps[[4]](#footnote-4) | Réseau |
| Câble droit | Câble croisé |
| Technologie |  |

### Compte rendu

#### Evolution vers la communication des trames (Ethernet Switching)

* Introduction du câblage 10BASE-T et de UTP 🡪 Popularisation des Hub en tant que moyen économique et facile d’utilisation
* Limitations des hubs avec les collisions de trames sur un même segment
* Ponts Ethernet pour résoudre les problèmes dans les réseaux locaux partagés 🡪 Amélioration des performances du réseau et réduction de trafic inutile
* Apparition des commutateurs 🡪 Augmentation des interfaces et augmentation du débit
  + Bande passante dédiées à chaque port
  + Environnement sans collision
  + Full Duplex

#### Logique de commutation des trames

* Commutateur Ethernet transfère les trames individuelles d’un port source vers un nœud ou un port de destination
* Connexion logique en fonction des adresses MAC sources et destinations via l’en-tête de la trame
* Le Commutateur décide :
  + Quand transférer ou filtrer une trame en fonction de l’adresse MAC de destination
  + Examine l’adresse de la source
  + Créer un environnement sans boucle avec les autres Commutateurs via le protocole Spinning Tree (STP)
* Pour le filtrage, il utilise une table d’adresse MAC inscrite dynamiquement dans la RAM et compare avec la destination

Une image contenant capture d’écran

Description générée automatiquement

Figure Table d'adresse MAC

#### Vue d’ensemble de la technologie Ethernet

* 802.3 est la norme IEEE pour l’Ethernet 🡪 Désigne une famille de normes définissant l’ensemble des couches physiques et la liaison de données de la technologie LAN en place
* L’Ethernet sépare la couche liaison du modèle OSI en deux couches distinctes :
  + Logical Link Control (LCL) établi dans la norme 802.2
  + Media Access Control (MAC) établi dans la norme 802.3
* LCL gère la connexion entre la couche MAC et celle réseau
* MAC encapsule les données :
  + Assemblage des trames avant transmission
  + Analyse de la trame lors de la réception
  + Contrôle l’accès aux médias 🡪 Par la méthode Carrier Sense Multiple Access with Collision Detection (CSMA/CD) lors du Half Duplex

#### Technologie Ethernet Traditionnelle et actuelles

* Spécifications 10BASE5 et 10BASE2 connectés sur un bus
* Arrivé UTP 🡪 Développement des HUB avec la topologie Etoile et nécessite le CSMA/CD
* Topologie étoile permet le Full Duplex ainsi désactive le CSMA/CD
* Ethernet 🡪 10Mbps sur cuivre
* Fast Ethernet 🡪 100Mbps sur cuivre
* GigaBit Ethernet 🡪 1000Mbps par fibre

#### Câblage UTP

* Utilisation courante de 2 ou 4 paires de fils 🡪 Généralement utilisé avec un connecteur RJ-45 (8 fils)
* Telecommunication Industry Association (TIA) 🡪 Norme des couleurs des fils
* Appareil utilisant paire de broche opposées pour transmettre 🡪 câble droit
* Appareil utilisant la même paire de broche pour transmettre 🡪 câble croisé

#### Adressage Ethernet

* IEEE définit un format d’attribution d’adresse LAN
* Adresse MAC unique :
  + 1ère moitié provient du fabricant de la carte 🡪 Identificateur unique de l’organisation (OUI)
  + Seconde moitié est donnée par le fabricant de la carte et n’est jamais utilisée par une autre carte avec le même OUI
* Ethernet possède des adresses de groupe :
  + Adresse de diffusion : Tous les appareils doivent traiter la trame suivante FFFF.FFFF.FFFF.FFFF
  + Adresse multidiffusion : Permet à un sous ensemble d’appareils sur un même réseau local de communiquer

#### Ethernet Framing

* « Framing » représente la définition des informations supposées se situer dans les données reçues

Une image contenant capture d’écran

Description générée automatiquement

* Préambule : Synchronisation des données
* SFD (Délimiteur du début de trame) : Marque le début du champ MAC de destination
* Longueur : Taille du champ de données de la trame
* FCS (Séquence de contrôle de trame) : Méthode pour la carte réseau de déterminer la présence d’erreur

### Avis critique du document

Je donnerai la note de 8/10 à ce cours sur le protocole Ethernet. En effet, ce cours a bien ciblé mes attentes en termes d’informations pour mieux comprendre cette technologie pour mon projet comme par exemple avec le détail de la trame Ethernet.

Ce cours a d’abord traité des moyens de communique par l’Ethernet avec le fonctionnement des Hub et des Commutateurs. Puis, le développement des fonctionnements des émissions de trames avec cette technologie. Enfin, un brief vu d’ensemble des différentes évolutions et technologie de l’Ethernet compléter avec le câblage adéquat pour la communication. Cours intéressant en termes d’informations et d’images qui facilitent la compréhension du sujet bien que par moment il y est difficile de bien cerner les notions évoquées.

## Les différents écrans tactiles | Alrick Giry

### Identification

* Auteurs : Pierre-Sylvain ROOS, Laurent ACHARD
* Titre : Le tactile, troisième géné utilisateurs
* Collection : Techno sans frontière
* Parution : Mai-Juin 2014

### Évaluation du contenu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Critères d'évaluation** | **Faible** | **Moyen** | **Bien** | **Très bien** |
| Degré de fiabilité de l'information |  |  |  |  |
| Utilité et accessibilité, pertinence de ces informations |  |  |  |  |
| Qualité, utilité des illustrations |  |  |  |  |
| Lisibilité du document… |  |  |  |  |

### Liste des mots clefs pour ce document

* Capteur
* Ergonomie
* Informatique
* Logiciel
* Multitouch

### Compte rendu

* Interfaces homme machine : lignes de commandes (CLI) puis clavier souris (GUI) puis écran tactile
* Premier écran tactile né en 1953, introduit dans notre vie quotidienne en 2007 avec l’iPhone
* Dalle tactile n’a pas d’écran contrairement aux écrans tactiles
* Écrans tactiles communiquent par USB, données interprétées par driver et librairies et affiché sur écran grâce à la carte graphique.
* Deux types de tactile : électriques et optiques
  + Résistif (électrique) : résistant, utilisable avec n’importe quel pointeur, pas de pression simultanée
  + Capacitif (électrique) : utilisable sur surface parfaitement lisse et par pointeur tactiles seulement, apte au multitouch.
  + Quadrillage de faisceaux infra-rouge et capteurs (optique) : détectent plusieurs doigts (jusqu’à 40), résistant.
  + FTIR (frustrated total internal reflection), DI (diffused illumination), PSD (planar scatter detectection), optical imaging et DSI (diffused screen illumination) : projection infra-rouges captés par caméra.
* Prix : 1500€ un écran capacitif 25 pouces, 80 000€ mur tactile 7m²
* Écrans classiques 6 à 10 points
* Écrans plus avancés 32 à 64 points
  + Permet le travail collaboratif
* Les grands écrans tactiles permettent donc de gagner en rapidité et en organisation
* Recherche pour transformer n’importe quelle surface en surface tactile
* Recherche écrans simulant objets en 3D
* Recherche impression d’écrans tactiles

### Pertinence des informations

Je donnerai la note de 8/10, à ce document. Il correspond à ma partie dans notre projet car il traite des écrans tactiles et des IHM. L’information est fiable car les deux auteurs sont soit ingénieur ou chercheurs et elle provient d’un magazine spécialisé dans la technologie. Les sources sont citées en fin de document ce qui permet de vérifier l’information. Les images sont très claires et permettent de bien comprendre le fonctionnement de chaque technologie. Il est parfois difficile de suivre le cours de sa phrase quand la ligne change de page et qu’une illustration avec sa description est positionnée entre les deux pages. Je reproche aussi à l’article de rester trop vague sur certains points et en même temps d’aller trop loin dans certains détails. Le début est aussi un peu confus car l’auteur commence à expliquer comment fonctionnent les écrans tactiles puis fait un historique des découvertes et ensuite revient au fonctionnement.

### Aperçu du document

Figure Aperçu document les différents types d'écrans tactiles

## Annexe 2 | Python | Jan BARILLEC

### Identification

|  |  |
| --- | --- |
| Auteur : | Gérard Swinnen |
| Titre : | Apprendre à programmer avec Python |
| Edition : | O’REILLY |
| Date de 1ère parution : | 28 novembre 2003 |

Etude du chapitre 11, de la page 145 a 152

### Liste des mots clés

* Classe
* Objets
* Attributs
* Variables
* Arguments
* Fonction
* Encapsulation
* Dérivation
* Polymorphisme

### Compte rendu

* Classes, objets, attributs
* Un objet est une entité qui est créé par l’instanciation d’une classe.
* Les objets peuvent être composés d’autres objets plus simples.
* Utilité des classes
  + Les objets permettent de regrouper un ensemble d’opérations pour permettre au code d’être plus clair et concis.
  + Les objets sont plus faciles a manipuler car le programmeur n’as pas besoins de savoir comment l’objet est codé mais seulement les actions qu’il permet de faire.
  + Les objets permettent à tout programmeur de réutiliser et de créer de nouveau objets à partir d’anciens.
  + Il est possible de créer une classe enfant à partir d’un autre parent, ainsi les enfants héritent des propriétés du parent, c’est le principe de dérivation.
  + Les objets peuvent également agir de façon différente selon le contexte
* Définition d’une classe élémentaire
  + En Python on déclare une nouvelle classe avec le mot clé « class »
  + Pour créer un objet on réalise la chose suivante : NomObjet = NomClasse()
* Attributs (ou variables) d’instance
  + Les attributs sont des variables contenues dans l’objet et accessibles ou non en dehors de l’objet.
* Passage d’objet comme arguments lors de l’appel d’une fonction
  + Il est possible de fournir a une fonction un objet mais il est aussi possible pour une fonction de renvoyer un objet
* Similitude et unicité
  + Deux objets issus d’une même classe et comportant les mêmes attributs, restent des objets distincts et ne sont pas identiques
* Objets composés d’objets
  + On accède aux objet à l’intérieur d’autres objet en utilisant des points de cette façon : Objet2.Objet1.AttributObjet1
* Les Objets sont modifiables
  + Sous Python les attributs sont toujours publics c’est-à-dire qu’ils sont accessibles en dehors de la classe.

### Évaluation du contenu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Critère d’évaluation | Faible | Moyen | Bien | Très bien |
| Degré de fiabilité de l’information |  |  |  | X |
| Utilité, accessibilité, pertinence des informations |  |  |  | X |
| Utilité des illustrations |  |  | X |  |
| Qualité, nombre d’illustrations |  |  | X |  |

### Pertinence des informations

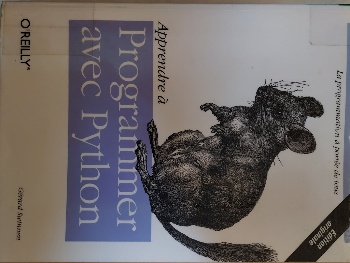
Le livre est ouvert a tous puisqu’il reprend l’ensemble des notions de bases de la programmation, de la création de variable jusqu’à des notions plus complexes comme la programmation orienté objet.

Chaque notion sont contenues dans des chapitres détaillés avec une partie de théorie et un exemple afin d’illustrer le propos. Cependant des illustrations en couleurs auraient permis une meilleure lisibilité du code.

Mon étude s’est portée sur le chapitre du livre qui présente les objets, car malgré le fait que nous avons eu des cours sur la programmation orienté objet en C++, ce livre a permis d’identifier les différences avec le Python.

**Note : 8/10**

### Illustration du livre étudié

****

## Les technologies de géolocalisation | Antonin Lyaët

### Introduction

Aujourd’hui, une des technologies les utilisés par tous est la géolocalisation. Même si elle utilisée par tous, peu de personnes ne connaissent le fonctionnement de cette technologie ainsi que les différents moyens de géolocalisation.

### Identification

Ce document est un article qui présente les principales technologies de géolocalisation, leurs avantages et leurs inconvénients. Cet article, posté sur « Organilog.com », s’intitule « Fonctionnement des systèmes de géolocalisation des smartphones » et vient énoncer les différents types de géolocalisation, les avantages et les inconvénients. Ce guide vient donc nous aider à répondre à la question suivante : « Quelles est la meilleure technologie de géolocalisation et comment fonctionnent-elles ? »

Lien : <https://fr.organilog.com/454-fonctionnement-geolocalisation-mobile/>

### Évaluation du contenu

|  |  |  |  |
| --- | --- | --- | --- |
| Critères | Faible | Moyen | Bien |
| Fiabilité |  |  |  |
| Illustration |  |  |  |
| Lisibilité |  |  |  |
| Pertinence |  |  |  |
| Accessibilité |  |  |  |

### Liste des mots clés

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| GPS | Géolocalisation | Satellite | Wifi | Technologie | GSM |
| Prix | Consommation | Précision | Fonctionnement | Intérieur | Extérieur |
| Vitesse | Matériel | Rural | Urbain | Batterie | Solution |

### Compte rendu

#### La géolocalisation

La géolocalisation est un système se situant dans beaucoup de technologie d’aujourd’hui comme les smartphones, les voitures, les drones, … Ce système est un concept très utilisé par tous mais peu de personne ne connaissent son fonctionnement. Mais connaitre les différents procédés de géolocalisation et leurs fonctionnements, cela permet une optimisation de la batterie pour les appareils qui utilisent ce système. Les 3 procédés les plus connus sont : le GPS (Global Positioning System), le GSM (Global System for Mobile) et le Wifi.

#### Géolocalisation par GPS

Ce système de géolocalisation doit disposer de deux éléments essentiels : un vue sur le ciel pour communiquer avec des satellites de géolocalisation ainsi qu’une puce GPS. Ce procédé géolocalise une puce GPS grâce à des signaux émis par un réseaux de satellites. Une fois que la puce intercepte un des signaux, elle renvoie ce signal au satellite qui l’a émis. Puis par triangulation des signaux renvoyés par la puce, on peut déterminer les coordonnées GPS de la puce.

Cette géolocalisation permet de géolocaliser une puce avec précision de 15 à 100 mètres mais des satellites européens, stellite Galileo, viennent d’être lancés et permettent une précision de 1 mètres.

Cette puce a un néanmoins deux gros défauts, elle consomme énormément de batterie. C’est pour cela que les concepteurs d’appareils équipés de ce système ont mis en place un moyen de l’activer ou non. Son deuxième gros défaut est que la réception et/ou le renvoie de signaux ne se fait pas correctement en intérieur

#### Géolocalisation par GSM

La géolocalisation par GSM fonctionne grâce au antennes GSM qui sont utilisés principalement pour la transmission de données aux téléphones ou voiture. Ce système fonctionne donc comme le GPS, avec le temps de réception en fonction des antennes qui les reçoivent ou émettent. Un algorithme de triangulation peut alors déterminer l’emplacement de l’appareil.

La précision dépend fortement de la densité des antennes présentent dans les environs. Elle est variée donc de 200 mètres à plusieurs kilomètres. Elle fonctionne donc beaucoup plus dans les milieux urbains que rural.

#### Géolocalisation par Wifi

Cette géolocalisation assez récente utilise des points d’accès Wifi connue pour déterminer là où l’appareil se trouve. C’est-à-dire à quelques dizaines de mètres. Il faut néanmoins que l’appareil en question soit muni d’une puce Wifi.

Cette puce consomme très peu et à une très bonne précision. Mais il faut être proche d’un accès Wifi connue pour que la géolocalisation par ce procédé se face. Ce concept vient à se répandre de plus en plus car Google vient référencer de nouveau point d’accès Wifi lorsque ses voitures utilisées par Google Street View passent devant de nouveau point.

#### Conclusion

D’après ce document, aucune des technologies de géolocalisation n’est mieux que les autres car elles dépendent toutes de là où on se situe et de comment on l’utilise. Pour avoir la meilleure géolocalisation possible, il faut combiner ces trois concepts de géolocalisation. Grâce à un algorithme prenant en compte ces système, on peut donc déterminer très facilement une géolocalisation très précise.

### Avis critique du document

Si je devais noter ce document, je lui mettrais la note de 7/10. Ce document est très complet, il nous apprend le fonctionnement des différents moyens de géolocalisation. Mais ce document est très court et manque d’illustration/schéma. Malgré ce gros défaut, il va tout de même droit au but et on comprend très bien comment fonctionne ces systèmes, leurs avantages et leurs inconvénients. Sa conclusion, rapide mais efficace, répond parfaitement à nos attentes même si nous n’avons pas de classement des meilleurs moyens de géolocalisation.

1. Série de bits organisés de façon à établir une communication [↑](#footnote-ref-1)
2. Transmission Control Protocol [↑](#footnote-ref-2)
3. Internet Protocol [↑](#footnote-ref-3)
4. Megabit [↑](#footnote-ref-4)