

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE

ELE300 PROJET PERSONNEL EN GÉNIE ÉLECTRIQUE

Système de localisation à l'aide de la force du signal WIFI

Réalisé par :

Tristan
BROUSSEAU-RIGAUDIE

Dirigé par :

Pr. Jérôme LE NY

7 décembre, 2017

Table des matières

1	Introduction	3
1.1	Définition du problème	3
1.2	Revue de la documentation	3
2	Spécifications fonctionnelles	5
2.1	Description des besoins	5
2.2	Définitions entrées/sorties du système	5
2.3	Description quantitative des performances requises	6
3	Design préliminaire	6
3.1	Technologie de localisation	6
3.1.1	Exploration des approches de résolutions	6
3.1.2	Étude de praticabilité et justification de la solution retenue	7
3.2	Méthode de localisation	7
3.2.1	Exploration des approches de résolutions	7
3.2.2	Étude de praticabilité et justification de la solution retenue	7
3.3	Structure générale du système	7
4	Design Détaillé	8
4.1	Conceptualisation et modélisation des éléments	8
4.1.1	Extraction de la force du signal reçu	8
4.1.2	Localisation par empreinte	9
4.1.3	Filtre $\alpha - \beta$	10
4.2	Réalisation du prototype	11
4.2.1	Capter les paquets et extraire la force du signal	11
4.2.2	Localiser par empreinte	14
4.2.2.1	Phase de mesures	14
4.2.2.2	Phase de localisation	14
4.2.3	Filtrer et suivre la trajectoire	15
4.2.4	Communiquer entre le client et le serveur et interagir avec l'utilisateur	15
4.3	Résultats préliminaires et itération du design	16
4.3.1	Localiser par empreinte	16
4.3.2	Filtre $\alpha - \beta$	17
5	Validation	18
5.1	Procédure de test	18
5.2	Analyse et validation des résultats	18
5.3	Analyse des résultats en fonction des hypothèses initiales	19
6	Conclusion	20
6.1	Évaluation des limites et des incertitudes	20
6.2	Proposition des travaux subséquents	20

7	Apprentissage continu	20
7.1	Résumé des nouvelles connaissances et habiletés acquises durant le projet	20
A	Documents supplémentaires	22

1 Introduction

1.1 Définition du problème

Nous sommes présentement dans une ère connectée où pratiquement chaque personne en Amérique du Nord possède un appareil électronique dans ses poches lui permettant de se connecter à l'internet. Ces appareils fonctionnent bien souvent avec le standard sans fil Wi-Fi. Celui-ci se base sur un médium de transmission non guidé, soit l'air. Ceci implique que l'information envoyée par un appareil peut être captée par un autre qui est à l'écoute même si le message ne lui est pas destiné. Une fonctionnalité intéressante de cela est la localisation d'appareils fonctionnant avec le protocole Wi-Fi.

Plusieurs applications peuvent découler d'une plateforme de localisation comme celle-ci. Par exemple, le système pourrait être utilisé pour la détection de gens attendant à un feu de circulation ou pour guider des pompiers lorsqu'il y a un feu dans un immeuble. D'autres applications, pourraient être fait à l'intérieur d'une maison où à l'aide de la localisation des gens il serait possible de détecter la présence d'intrus ou de contrôler les lumières selon les pièces dans lesquels les habitants se trouvent. Donc, le but du projet est de développer le prototype d'une plateforme de localisation et de suivi de trajectoire à l'aide de la force du signal Wi-Fi reçu (RSS) et d'un filtre α - β .

Ce présent document est présenté comme suit, la section 1.2 introduit les travaux effectués et quelques documentations disponibles sur la localisation par Wi-Fi. La section 2 explicite les spécifications fonctionnelles du projet. La section 3 introduit une conception préliminaire du système. La section 4 enchaine sur la conception détaillée et certains résultats préliminaires sont présentés permettant de raffiner la conception. La section 5 valide les résultats obtenus. La section 6 et 7 conclut ce rapport en fermant sur les apprentissages du projet.

1.2 Revue de la documentation

Plusieurs travaux ont été faits sur la localisation par signal Wi-Fi. Beaucoup de ceux-ci portent sur la localisation basée sur la force du signal (RSS), par exemple Mohamed Taboubi a fait une mémoire en 2014 [13], dans laquelle il étudie plusieurs méthodes de localisation par RSS en monitorant la force du signal reçu par les différents points d'accès disponibles à l'école Polytechnique. Son étude comporte deux approches soit une méthode d'emprunte et une méthode de multilatéralisation. La première méthode consiste à recueillir plusieurs forces du signal pour plusieurs positions données (x, y) avec lesquelles une base de donnée d'emprunte est créée. Ces empruntes sont alors comparées avec les forces du signal reçues lors de la localisation. La position de l'emprunte ayant la plus grande ressemblance avec les RSS recueillies est alors choisie. Dans la se-

conde méthode, le modèle de propagation de l'onde est utilisé afin de déterminer la distance entre chaque émetteur et le récepteur. Ayant plusieurs émetteurs, il est possible de localiser le récepteur avec la multilatéralisation, soit dans le cas où l'on a seulement trois émetteurs on se retrouve avec un problème de triangulation. À l'aide de la première approche, il obtient une précision de moins de 2 mètres pour 56% des points de références. Pour la seconde approche, il obtient une précision de moins de 2 mètres pour 88% des points de références, mais seulement en simulation et considérant l'environnement comme un espace libre. Pour la méthode de multilatéralisation, une approche commune est de trouver les paramètres d'atténuation et de propagation de l'onde empiriquement. Un modèle de propagation bien utilisé est le suivant :

$$P_r = \frac{P_t \times \alpha}{d_{t \rightarrow r}^\beta} \quad (1)$$

Où :

- P_r et P_t sont respectivement la puissance reçue et la puissance émise.
- α est un facteur d'atténuation du signal.
- β est le facteur de perte de chemin. Dans un espace libre, il est de 2.
- $d_{t \rightarrow r}$ est la distance entre l'émetteur et le récepteur.

Soit pour ce modèle simplifié et considérant la méthode de multilatéralisation, il faudrait pour certaines régions où l'on souhaite se localiser à partir des forces du signal reçues caractériser $P_t \times \alpha$ et β pour chacun des émetteurs [4]. Il est à noter que force du signal ne varie pas si simplement que cela dans un environnement non libre. Le plancher agit comme un réflecteur pour les signaux Wi-Fi. Il peut y avoir de l'interférence entre plusieurs émetteurs. Les ondes peuvent utiliser des chemins multiples et peuvent être atténuées par des obstacles imprévus [6]. Pour ces diverses raisons la force du signal reçu tend à osciller même si le récepteur ne bouge pas.

Dans [7] les auteurs ont ajouté un filtre de Kalman à la suite de la localisation par empreinte. Avant le filtre, ils ont une erreur moyenne de 3.24 mètres et à la sortie du filtre cette erreur est diminuée à 2.7 mètres. Dans le présent projet, la localisation par empreinte va être utilisée avec une variante du filtre de Kalman, afin de diminuer le bruit, soit le filtre α - β qui a le comportement d'un filtre de Kalman en régime permanent.

2 Spécifications fonctionnelles

2.1 Description des besoins

Il y a des logiciels de localisation bien connus comme ceux offerts par Google ou Apple mais ceux-ci offrent en général une précision moyenne supérieur à 5 mètres à l'extérieur [14] et ,lorsqu'on se trouve à l'intérieur, il vont nous dire dans quel bâtiment nous nous trouvons sachant l'adresse IP de notre connexion à l'internet. Nous souhaitons alors établir un système de localisation peu coûteux et permettant la possibilité de recueillir la position de plusieurs personnes, tant à l'intérieur d'un bâtiment qu'à l'extérieur. Pour un premier prototype, nous désirons développer un logiciel déployable sur un système d'exploitation de type linux. Un appareil muni de ce logiciel doit être en mesure de se localiser dans l'atrium de la Polytechnique. Le programme doit être facilement malléable afin d'être utilisé dans d'autres environnements. Le logiciel doit être accompagné d'une interface graphique permettant de voir en temps réel la position de l'appareil l'hébergeant.

2.2 Définitions entrées/sorties du système

Le système comporte comme entrées les messages se trouvant dans l'air. Plus spécialement, on désire filtrer ces messages pour conserver seulement ceux envoyés par les différents points d'accès. Pour le cas du prototype, nous en sélectionnant quatre. La position des points d'accès est montrée à la figure 1.

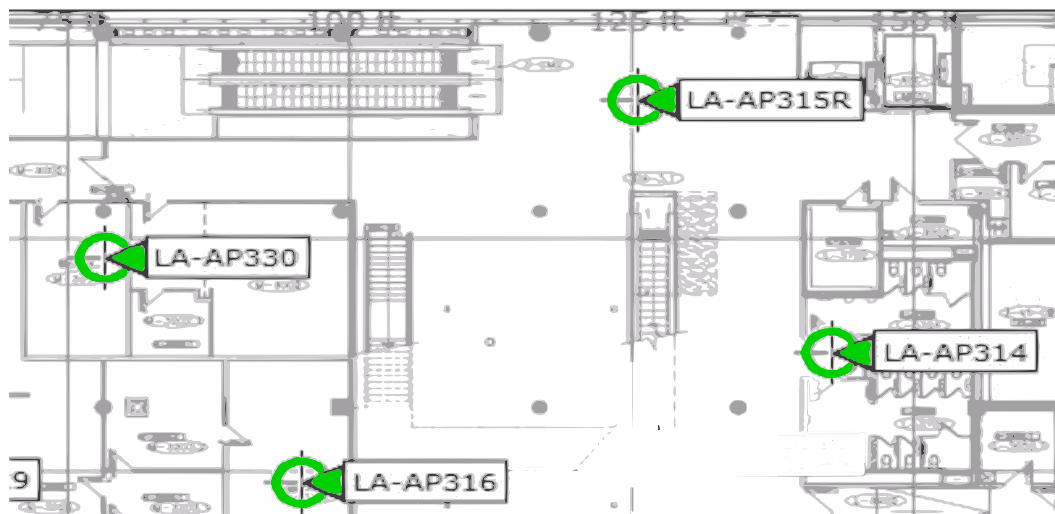


FIGURE 1 – Points d'accès de l'atrium

Soit :

- LA-AP316 (PA0) et LA-AP330 (PA1) envoient et reçoivent des trames sur le canal 1.
- LA- AP314 (PA2) envoient et reçoivent des trames sur le canal 6.
- LA-AP315R (qu'on note PA3) envoient et reçoivent des trames sur le canal 11.

Ces points d'accès sont statiques et envoient à un intervalle régulier des trames balises (beacon frames) définies par la norme 802.11 de manière à faire la gestion du réseau. Nous sommes alors assurés d'obtenir les forces du signal à un intervalle régulier.

2.3 Description quantitative des performances requises

Nous désirons concevoir un système de localisation ayant une précision moyenne inférieure à 5 mètres. Ce système devra être muni d'une interface graphique permettant de visualiser la position en temps réel de manière non critique, donc ayant un temps rafraîchissement des données inférieur à 3 secondes.

3 Design préliminaire

Dans cette section, différentes approches de localisation sont discutées. En premier lieu, différentes technologies sont présentées et en deuxième lieu différentes méthodes de localisation sont étudiées. Une solution pour chacune de ces parties est alors retenue. Pour compléter cette section, la structure générale du système à concevoir sera présentée.

3.1 Technologie de localisation

3.1.1 Exploration des approches de résolutions

L'émergence des systèmes de communication sans-fil ont apporté une panoplie de choix quant à la localisation. On peut penser à la technologie UWB qui envoie des impulsions à un intervalle régulier en utilisant peu d'énergie et fonctionnant sur de courte distance et qui permet en général une précision entre 10 et 30 centimètres [9]. Cependant, nous désirons utiliser un protocole qui est présentement intégré sur la majorité des téléphones cellulaires. De ce fait, il y a deux choix de protocole : Bluetooth et Wi-Fi. Pour le premier protocole, des balises (beacons) ont été introduits en 2013 par Apple [11]. Ces balises ressemblent à de petites roches que l'on installe sur les murs. Ceux-ci permettent l'échange de messages Bluetooth avec les téléphones cellulaires situés à proximité et propose une précision de 1 à 3 mètres [8]. Pour ce qui est du Wi-Fi, les points d'accès présents dans les bâtiments peuvent servir à la localisation comme mentionné précédemment. Avec ceux-ci, on est généralement en mesure d'obtenir une précision moyenne inférieure à 5 mètres [13].

3.1.2 Étude de praticabilité et justification de la solution retenue

Les beacons, couvrant un territoire allant jusqu'à 30 mètres [8], semblent attrayant vue la précision proposée. Cependant, cette solution nécessite l'achat de matériel supplémentaire, soit environ l'achat de 3 à 5 beacons. L'avantage d'utiliser le Wi-Fi est la disponibilité de points d'accès dans les établissements. C'est principalement pour cette raison que nous choisissons ce protocole.

3.2 Méthode de localisation

3.2.1 Exploration des approches de résolutions

Puisque nous utilisons les points d'accès pour se localiser dans un environnement et que nous n'avons pas accès à ceux-ci, on se retrouve dans une situation de localisation par soi-même (self-positioning). Il y principalement deux métriques qui peuvent être utilisées pour la localisation. La première est le temps de voyage de la trame. Connaissant la vitesse de propagation de l'onde et son temps de propagation, il est possible de déterminer la distance. Cependant, le problème est la synchronisation de l'horloge entre les émetteurs et le récepteur. La seconde métrique qui peut être utilisée est la force du signal reçu (RSS). Cet indicateur varie selon la distance entre l'émetteur et le récepteur comme le montre l'équation 1. La méthode par empreintes et la méthode par multilatéralisation présentées à la section 1.2 sont les deux principales possibilités d'obtenir la position à l'aide des RSS provenant de plusieurs points d'accès.

3.2.2 Étude de praticabilité et justification de la solution retenue

La méthode par multilatéralisation demande une connaissance du comportement de la propagation de l'onde entre les émetteurs et le récepteur que l'on peut déterminer empiriquement. Elle peut être simple à implémenter lorsque l'environnement est uniforme. Cependant, il devient plus complexe lorsqu'il y a des obstacles, comme par exemple à la figure 1 où on peut voir l'escalator qui divise l'environnement en plusieurs sections. Dans ce cas, il faudrait déterminer les paramètres de l'équation 1 pour ces différentes sections. Ainsi, il serait possible de se localiser sachant la section initiale dans laquelle on se trouve. La méthode par empreinte est moins complexe dans le sens que l'on n'a pas besoin de se soucier des irrégularités de l'environnement. Pour cette raison, c'est la méthode qui sera utilisée dans le projet.

3.3 Structure générale du système

Le système conçu est séparé par bloc. Ceux-ci sont présentés à la figure suivante.

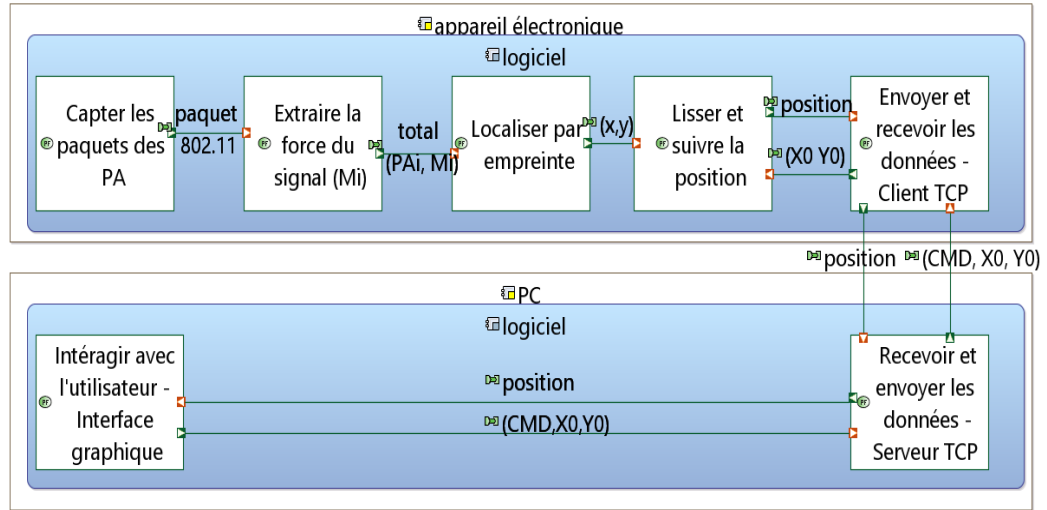


FIGURE 2 – Architecture globale du système

Sur l'appareil électronique, on commence par capter les paquets envoyés par les points d'accès. Seulement la force du signal reçu Mi et l'identifiant $i \in \{0, 1, 2, 3\}$ (rf. section 2.2) du point d'accès sont transmis au module de localisation. Ce dernier module détermine la position par empreinte. La position est par la suite filtrée à l'aide d'un filtre $\alpha - \beta$ et envoyée à un serveur qui permet l'affichage des données. Il est à noter que l'interface graphique pourrait aussi se trouver sur l'appareil électronique. L'interface graphique doit permettre à l'utilisateur de choisir s'il veut effectuer des mesures sur la force du signal reçu selon une position statique ou se localiser dans l'environnement.

4 Design Détaillé

4.1 Conceptualisation et modélisation des éléments

4.1.1 Extraction de la force du signal reçu

Au Canada, le Wi-Fi est composé de 11 canaux dans la bande du $2,4GHz$. Soit les 11 premiers de la figure 3.

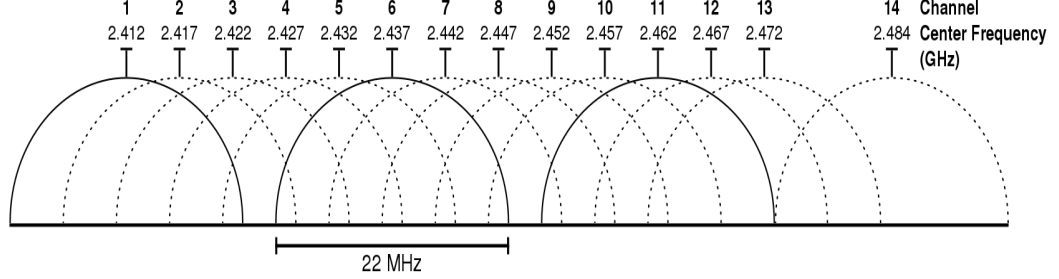


FIGURE 3 – Les quatorze canaux de la bande du $2.4GHz$ utilisés dans le monde [3].

Comme mentionné à la section 2.2, les points d'accès utilisés pour la localisation utilisent les canaux 1, 6 et 11. Pour capter les trames de ces points d'accès, il suffit de mettre son interface réseau en mode écoute (monitoring) et changer la fréquence d'écoute à un intervalle régulier. Les paquets reçus consistent à une suite de caractères. En utilisant Wireshark il est possible de déterminer la position de la force du signal reçu dans un paquet donné. On trouve que la RSS se trouve au quatorzième octet et est de type *int8_t*.

4.1.2 Localisation par empreinte

La localisation par empreinte se fait en deux phases. La première phase consiste à une phase de mesures. Dans celle-ci, nous devons mesurer et enregistrer la force du signal pour plusieurs positions dans la région où l'on souhaite se localiser. Ainsi, dans cette phase on établit une base de données de référence. Soit dans un plan à 2 dimensions avec les axes (x, y) , on obtient de cette phase la force du signal reçu pour chacun des points d'accès (PA) et chaque couple (x, y) qu'on note $S^{PA_i}(x, y)$. La seconde phase consiste à se localiser à l'aide de la base de données de référence. Du module précédent, on obtient les RSS pour chaque PA en temps réel qu'on note M^{PA_i} . On peut alors déterminer la position actuelle (x, y) à l'aide d'une mesure de ressemblance entre M^{PA_i} et $S^{PA_i}(x, y)$. La position qui est la plus encline à être la position réelle est celle qui va minimiser la distance euclidienne entre la référence et la mesure. En d'autres termes, ça revient à trouver le couple (x, y) qui minimise l'équation 2.

$$d(x, y) = \sqrt{\sum_{i=0}^3 (S^{PA_i}(x, y) - M^{PA_i})^2} \quad (2)$$

Puisque la force du signal a tendance à fluctuer même si la position de la personne reste constante et que l'environnement a tendance à varier il est recommandé d'utiliser l'algorithme des K plus proches voisins (K -NN). On

prend alors les K couples (x, y) qui minimisent $d(x, y)$ et on fait une moyenne de ces couples, comme le montre l'équation 3.

$$[\hat{x} \ \hat{y}] = \frac{\sum_{j=1}^K [x_j \ y_j]}{K} \quad (3)$$

4.1.3 Filtre $\alpha - \beta$

Pour diminuer l'impact de la fluctuation de la force du signal reçu un filtre est nécessaire. L'implémentation d'un filtre discret $\alpha - \beta$ est choisie. Celui-ci nous permet aussi d'obtenir un estimé de la vitesse de déplacement. Ce filtre contient deux phases et fonctionne avec un historique d'une période. La première phase établit une prédiction et prend pour acquis que la vitesse reste fixe entre deux échantillons séparés par la période ΔT (rf. équation 4). La seconde phase permet d'incorporer les nouvelles mesures avec un certain degré de confiance défini par les facteurs α et β (rf. équation 5).

$$\hat{\mathbf{p}}_{k|k-1} = \hat{\mathbf{p}}_{k-1} + \Delta T \hat{\mathbf{v}}_{k-1} \quad (4)$$

$$\begin{cases} \hat{\mathbf{p}}_k = \hat{\mathbf{p}}_{k|k-1} + \alpha(\tilde{\mathbf{p}}_k - \hat{\mathbf{p}}_{k|k-1}) \\ \hat{\mathbf{v}}_k = \hat{\mathbf{v}}_{k-1} + \frac{\beta}{\Delta T}(\tilde{\mathbf{p}}_k - \hat{\mathbf{p}}_{k|k-1}) \end{cases} \quad (5)$$

Avec $\tilde{\mathbf{p}}_k = [\hat{x} \ \hat{y}]^T$, soit la position provenant de la localisation par empreinte. Il existe plusieurs relations entre α et β qui découlent des gains d'un filtre de Kalman en régime permanent. Parmi ceux-là, il y a la relation développée par Benedict et Bordner [1] qui optimise les performances du filtre. Cette relation présentée à l'équation 6 permet d'obtenir un système sous-amorti.

$$\beta = \frac{\alpha^2}{2 - \alpha} \quad (6)$$

Dans [12], on propose un réarrangement de la relation présentée par Benedict et Bordner afin d'obtenir un système amorti critiquement. Ce réarrangement présenté à l'équation 7 est en fait une résolution selon β avec l'ajout d'un facteur de 0.8.

$$\beta = 0.8 \left[\frac{2 - \alpha^2 - 2 \times \sqrt{1 - \alpha^2}}{\alpha^2} \right] \quad (7)$$

La dernière relation présentée provient du critère de la variance minimum [10]. Celle-ci est présentée à l'équation 8

$$\alpha = \beta^{\frac{1}{2}} - \frac{\beta}{2} \quad (8)$$

Les trois relations décrites ci-haut sont explicitées à la figure 4.

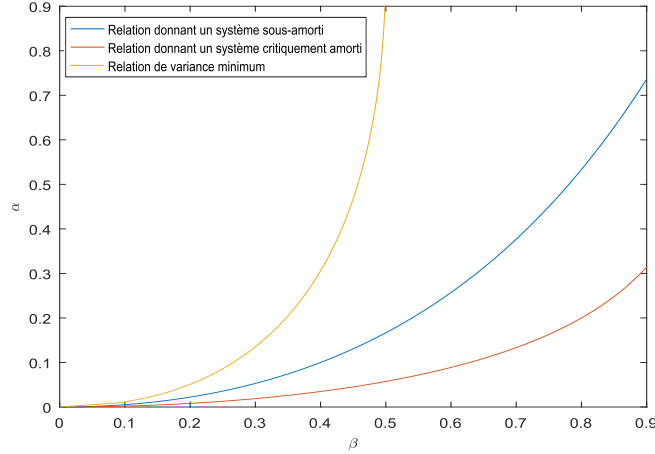


FIGURE 4 – Différentes relations d' $\alpha - \beta$

4.2 Réalisation du prototype

Le prototype s'agit d'un logiciel de localisation sur une *Raspberry Pi 3 (RPI 3)* communiquant sa position à un serveur établi sur *Windows*. Ce serveur affiche ensuite la position à l'écran et permet une interaction avec l'utilisateur. Cette sous-section vise à détailler la conception de ce système, elle est présentée sous forme de quatre parties séparant la conception par module :

- Capturer les paquets et extraire la force du signal.
- Localiser par empreinte.
- Filtrer et suivre la trajectoire.
- Communiquer entre le client et le serveur et interagir avec l'utilisateur.

4.2.1 Capturer les paquets et extraire la force du signal

De manière à avoir le maximum d'information sur les paquets perceptibles par son appareil il suffit de mettre son adaptateur Wi-Fi en mode écoute. Cependant, pas tous les adaptateurs et les pilotes matériels supportent ce mode, comme l'adaptateur *Broadcom* intégré à la *RPI 3*. Dans ce cas, il faut donc acheter une clé Wi-Fi supportant ce mode. Sur Linux la commande *iw* permet de changer les paramètres de son interface réseau [5]. Une fois qu'on a une interface en mode écoute, la librairie *C/C++ libpcap* permet de capturer les trames telles que reçues par la couche liaison du modèle OSI. Un court tutoriel sur l'utilisation de la librairie est donné en [2]. L'algorithme de capture de paquet est le suivant.

```

1  /**
2  \fn void changeFreq(int freq)
3  \brief Permet de changer la frequence de l'interface en ecoute
4      mon7
5
6  */
7  void changeFreq(int freq){
8      char command[32];
9      sprintf(command, "sudo iw dev mon7 set freq %d", freq);
10     system(command);
11 }
12
13
14 /** Nombre de point d'accès */
15
16 int NbAP = 4;
17
18 /** Nombre de paquet qu'on desire capter par PA a la fois */
19
20 int NbPacketByAP = 2;
21
22 /** Les differents filtres selon les quatre points d'accès (les MAC
23     address de chaque point d'accès) */
24 char filter[120] = "ether host f4:0f:1b:ab:28:f9\0ether host f8:c2
25 :88:14:b5:39\0ether host f8:c2:88:2a:e0:c9\0ether host f8:c2
26 :88:22:09:79\0";
27
28 /** La frequence des canaux 1, 6 et 11 */
29 int freqArray[3]= {2412,2437,2462};
30
31 int main(int argc, char **argv)
32 {
33     /** Matrice contenant les RSS */
34     int8_t arrayRSS[(NbPacketByAP+2)*NbAP+1];
35
36     char *dev;
37
38     /**... Initialisation
39
40     on demare la capture avec la structure pcap_t* descr;
41     sur l'interface en mode ecoute
42
43     ...
44     */
45
46
47     bpf_u_int32 netp;
48     struct pcap_pkthdr header;
49     const u_char *packet;
50     while(1){
51         index = 1;
52         /** Nombre de paquets qu'on capte reellement */
53         int nbPackets = 0;

```

```

54  /**Index ou le nombre de paquets va se trouver dans arrayRSS*/
55  int index_NbPacket = 0;
56  /** NB de Mac id; */
57  arrayRSS[index++] = NbAP;
58  /** Changement de frequence pour l'AP0*/
59  changeFreq(freqArray[0]);
60
61  for(int i = 0; i < NbAP; i++){
62
63      /** On sauvegarde l'identifiant de PAi
64      $i \in \{0,1,2...,NbAP\}$ */
65
66      arrayRSS[index++] = (int8_t)i; // MAC id;
67
68      index_NbPacket = index++;
69
70      /** On compile le filtre relie au PAi */
71      if(pcap_compile(descr,&fp, filter+(i*29),0,netp) == -1)
72      {fprintf(stderr, "Error calling pcap_compile\n"); exit(1)}
73  };
74
75      if(pcap_setfilter(descr,&fp) == -1)
76      {fprintf(stderr, "Error setting filter\n"); exit(1);}
77
78      for(int j = 0; j < NbPacketByAP; j++){
79          packet = pcap_next(descr, &header);
80          if(packet > 0 ){
81              /**On a capturer un paquet, on extrait la RSS
82              (avec le offset de 14 char)*/
83              nbPackets++;
84              arrayRSS[index++] = *((int8_t *)packet)+14);
85          }
86          arrayRSS[index_NbPacket] = nbPackets;
87          /** On change la frequence selon le PAi*/
88          if(i > 1){
89              changeFreq(freqArray[i-1]);
90          }
91      }
92      /**On inscrit la nombre d'element au premier indice*/
93      arrayRSS[0] = int8_t(index -1);
94
95      /** Ici on peut exploiter arrayRSS ... */
96  }
97
98 }

```

Le tableau *arrayRSS* contenant les forces des signaux reçus pour tous les points d'accès est structuré comme le montre la figure 5.



FIGURE 5 – Structure du tableau contenant les forces des signaux reçus

Selon la fréquence à laquelle les trames balises sont envoyées par les points

d'accès, on arrive à obtenir une force du signal par point d'accès par seconde. On choisit d'en capter 2 à la fois ($NbPacketByAP = 2;$), ce qui prend environ 2 secondes. Une moyenne de ces deux mesures est ensuite faite avant de passer au module suivant.

4.2.2 Localiser par empreinte

4.2.2.1 Phase de mesures

On sélectionne un endroit où l'on souhaite se localiser et on mesure à un intervalle régulier de position la force du signal reçu pour chaque point d'accès. L'endroit de localisation ainsi que les positions de mesures choisies pour le prototype sont montrées à la figure 6.

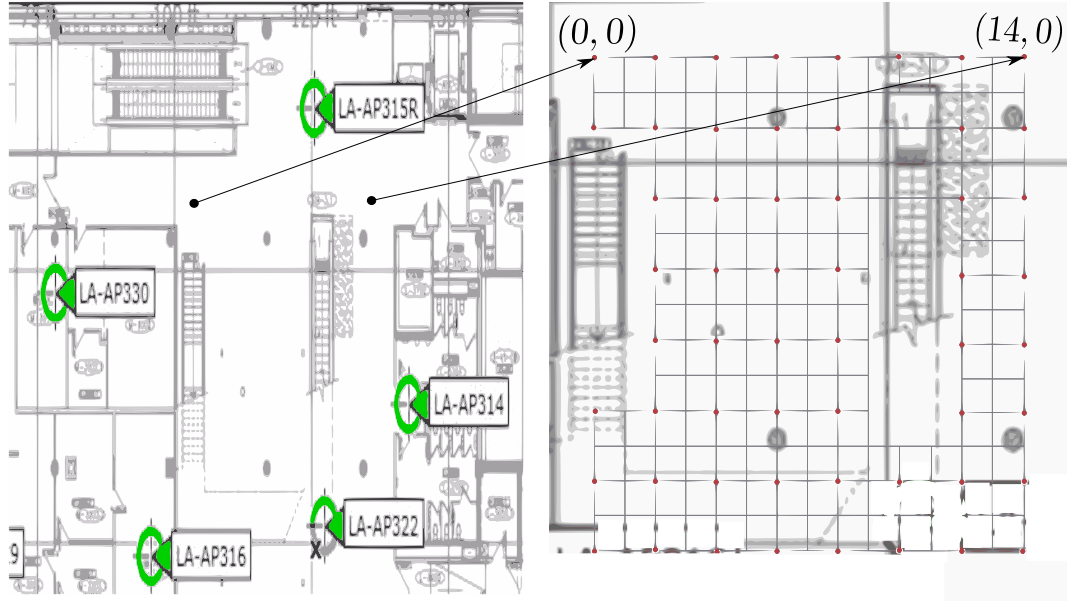


FIGURE 6 – Cartographie de l'atrium et points de mesure

La figure de droite montre les positions de référence en point rouge, elles se trouvent à deux mètres d'écart les unes aux autres et ce sur un carré de 14×14 mètres. Pour la phase mesure, on sauvegarde en moyenne une centaine de force du signal reçu par point d'accès pour chaque position.

4.2.2.2 Phase de localisation

Environ à chaque deux secondes on obtient les mesures M^{PA_i} du module précédent. On les compare alors aux références comme expliqué à la section 4.1.2. Afin de déterminer le nombre de voisins idéal pour l'algorithme K -NN,

on enregistre les résultats pour $K \in \{1, 3, 4\}$. De manière à trouver les K plus petites distances, on applique sur chacun des couples (x, y) mesurés à la phase précédente l'équation 2 au carré, soit $d^2(x, y)$ (le carré permet de diminuer les calculs).

4.2.3 Filtrer et suivre la trajectoire

Le filtre $\alpha - \beta$ va permettre d'établir un compromis entre le temps de convergence vers la moyenne des mesures et l'amplitude du bruit restant dans le système. Il faut alors établir les critères acceptables que l'on veut obtenir à la sortie du filtre. On s'entend pour dire qu'un temps de convergence d'une dizaine de secondes est acceptable pour le cas d'un piéton. On désire alors trouver le couple idéal $\alpha - \beta$ ayant un temps de réponse d'environ 10 secondes et minimisant le bruit. Pour cela, on laisse l'appareil à un endroit statique et on initialise le filtre avec une erreur acceptable. On juge l'erreur comme étant acceptable si elle est inférieure ou égale à la distance maximale que peut parcourir un piéton en 2 secondes, soit la période d'échantillonnage. Pour question de symétrie, une erreur de 2 mètres en x et en y est choisie. On applique alors le filtre pour différentes valeurs d' α et de β suivant les relations présentées à la section 4.1.3.

4.2.4 Communiquer entre le client et le serveur et interagir avec l'utilisateur

Le serveur fait office d'interface graphique. Celui-ci est programmé en *C++* à l'aide de *QT Creator*. Il permet à l'utilisateur de sélectionner le mode et de voir en temps réel la position de l'appareil. Trois modes sont proposés.

- *Mode localiser* : permet d'afficher la position de l'appareil.
- *Mode suivre* : permet d'afficher la position de l'appareil tout en traçant une ligne de suivie.
- *Mode mesure* : permet de prendre des mesures de références pour la localisation par empreinte.

L'appareil effectue les mêmes fonctions peu importe le mode à l'exception du mode mesure où les forces des signaux reçus sont enregistrées. De plus, lorsque l'utilisateur démarre un nouveau mode le filtre $\alpha - \beta$ est initialisé avec une vitesse nulle et une position déterminée par la saisie de l'utilisateur. La figure 7 représente l'interface graphique.

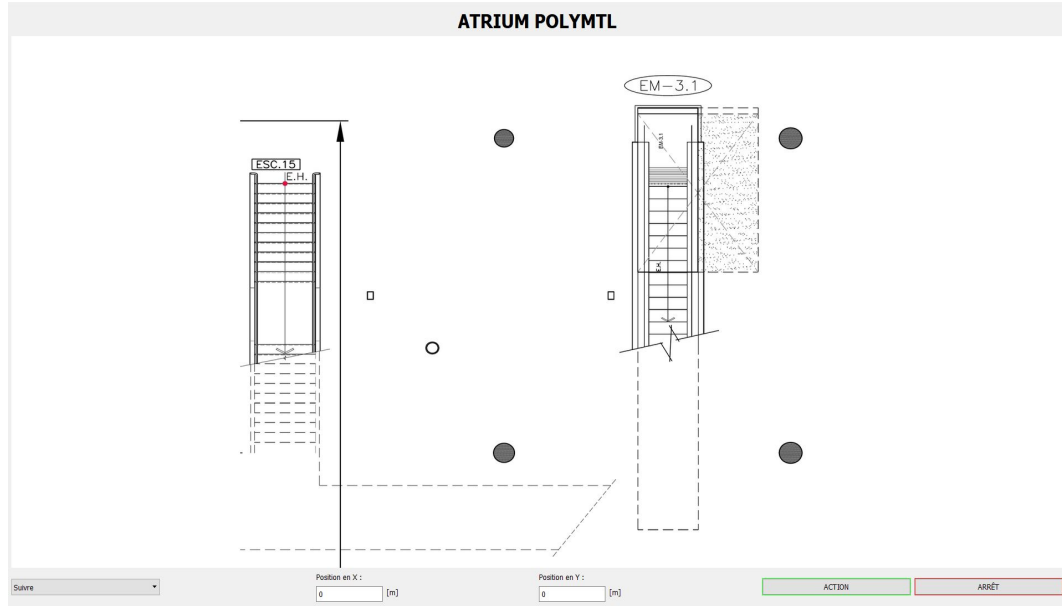


FIGURE 7 – Interface graphique.

La sélection des modes se fait en bas à gauche. Au bas centre il est possible de saisir une position en X et en Y seulement lorsque le système est en pause. Lorsqu'il est en action la position actuelle y est affichée. En mode suivie, afin d'éviter de tracer des lignes droites, une courbe de Bézier est tracée à l'instant K à l'aide des points de contrôle \hat{p}_{K-1} , $\hat{p}_{K|K-1}$ et \hat{p}_K .

La communication entre la *RPI* et l'ordinateur est établie à l'aide de sockets TCP. L'ordinateur envoie en continue un accusé de réception (« C \n ») ou le nouveau mode sélectionné, soit « M \n » pour le mode mesure ou « I \n » pour initialiser le filtre, avec la position initiale. Puisque des *floats* sont échangés entre les deux architectures ceux-ci sont convertis en *char* avant d'être envoyés.

4.3 Résultats préliminaires et itération du design

4.3.1 Localiser par empreinte

L'algorithme des K -*NN* a été effectué pour $K \in \{1, 3, 4\}$ et ce pour 11 positions différentes avec 50 mesures pour chacune d'elles. On obtient pour un vecteur d'erreur moyenne pour chacun de ses algorithmes.

- Pour $K = 1$: la norme de l'erreur moyenne est de 3.28 mètres.
- Pour $K = 3$: la norme de l'erreur moyenne est de 3.15 mètres.
- Pour $K = 4$: la norme de l'erreur moyenne est de 3.39 mètres.

On remarque qu'en moyenne on obtient la plus petite erreur avec $K = 3$, c'est alors l'algorithme qu'on choisit.

4.3.2 Filtre $\alpha - \beta$

En appliquant le filtre pour différentes valeurs d' $\alpha - \beta$ et pour différentes positions quelques valeurs ressortent selon les critères mentionnés ci-haut. Une sélection pour chacune des relations a été établie. La figure 8 présente cette sélection en montrant le comportement de l'erreur à la sortie du filtre. Celui-ci a été initialisé avec une erreur de 2 mètres en X et en Y .

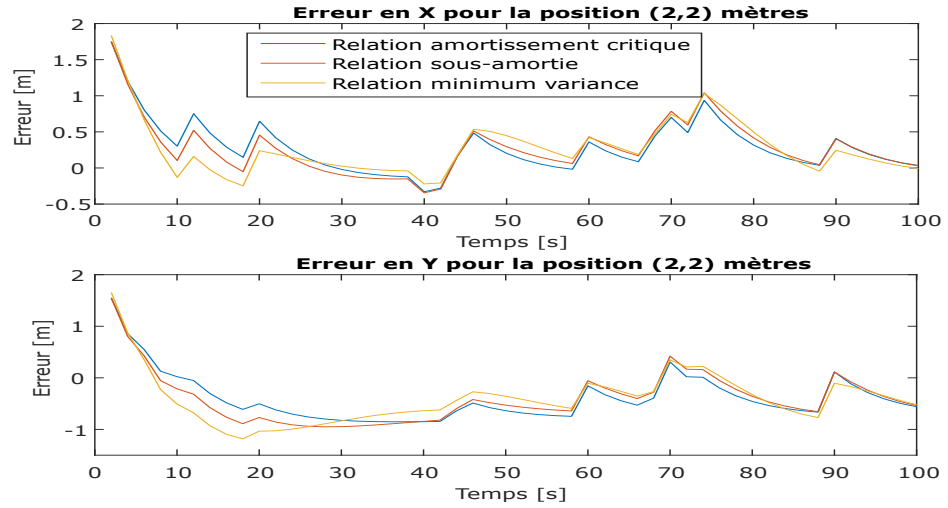


FIGURE 8 – Comportement du filtre $\alpha - \beta$ pour différentes relations

On peut remarquer que la relation de variance minimum offre un temps de montée plus rapide, mais a de plus grand dépassement. En X l'erreur devient inférieure à 30 cm après 8 secondes par contre à la 18^{em} seconde, on remarque qu'en Y l'erreur est de 1.2 mètres due au dépassement. La relation amortissement critique est la plus lente et permet alors d'éviter le dépassement. La courbe dirigée par la relation sous-amortie se trouve à être l'intermédiaire des deux autres, elle offre de légers dépassements et une vitesse de suivi moyenne. Puisqu'on ne conçoit pas un système critique on peut affirmer que l'allure générale des trois courbes est assez semblable. Cependant, afin d'éviter le dépassement au maximum on choisit la relation amortissement critique où à la 10^{em} seconde on se retrouve avec une erreur de 30 cm en X et de 2 cm en Y . Le filtre se trouve alors avec $\alpha = 0.3$ et $\beta = 0.0189$.

5 Validation

5.1 Procédure de test

La principale procédure de test consiste à prendre des mesures de la position obtenue à l'entrée et à la sortie du filtre pour des positions statiques. Cette procédure permet en premier lieu de s'assurer que la communication entre la *RPI* et le serveur est fonctionnelle. En deuxième lieu, ça permet de connaître la précision de la localisation par empreinte et les performances du filtre $\alpha - \beta$.

5.2 Analyse et validation des résultats

Les résultats de la localisation par empreinte sont présentés à la figure 9 où l'erreur moyenne pour onze différentes positions mesurées est montrée.

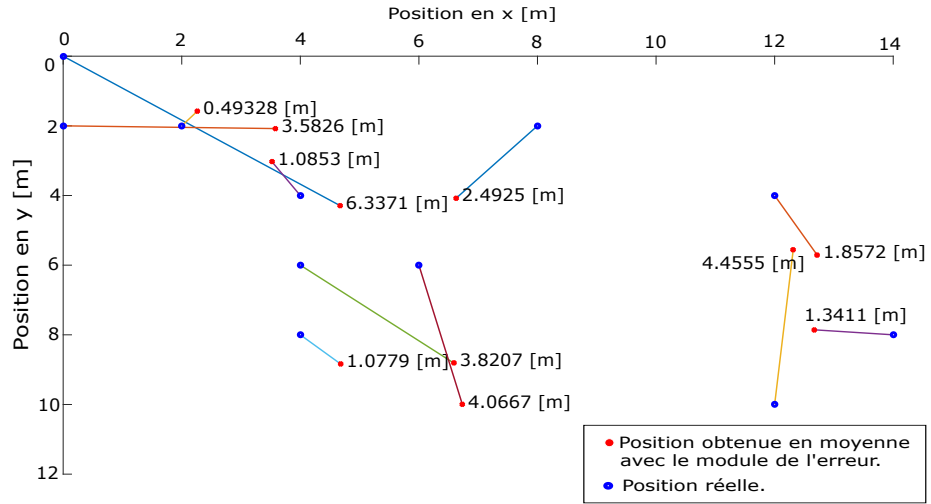


FIGURE 9 – Erreurs de la localisation par empreinte.

Pour chacune de ses onze positions cinquante mesures ont été prises. On remarque que la plus grande erreur moyenne obtenue est de 6.34 mètres, alors que la plus petite est de 50 centimètres. En somme, l'erreur moyenne est de 3.15 mètres. On se retrouve avec une erreur inférieure à 2 mètres pour 45% des résultats. La figure 10 montre l'impacte de l'ajout du filtre $\alpha - \beta$.

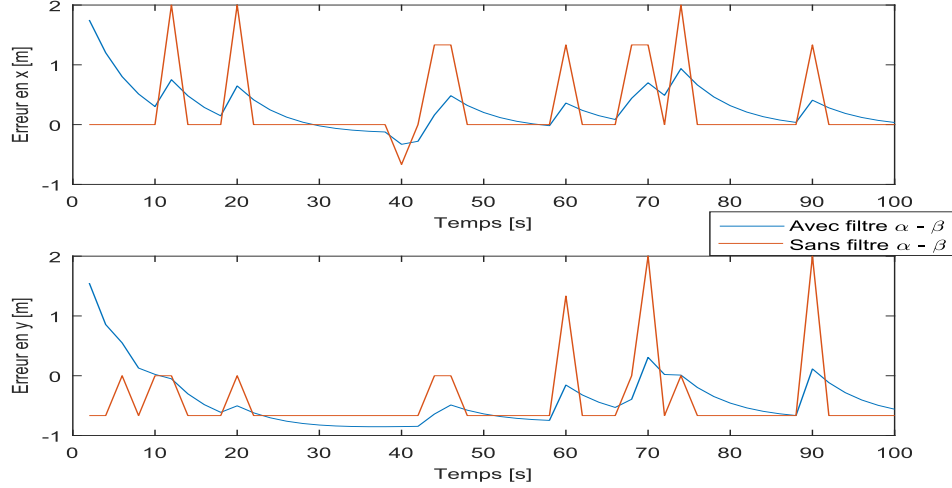


FIGURE 10 – Comparaison entre l’entrée et la sortie du filtre $\alpha - \beta$ pour la position (2, 2) mètres.

De cette figure, on remarque que le filtre diminue l’amplitude du bruit. À l’entrée du filtre la variation de l’erreur en X et en Y est de -0.67 mètres à 2 mètres. À la sortie du filtre, l’amplitude de l’erreur en X est de -0.33 mètre à 0.94 mètre et en Y elle est de -0.86 à 0.31 mètre. En somme, avec les positions mesurés ci-haut à l’aide de la localisation statique on obtient une erreur moyenne de 2.98 mètres à la sortie du filtre. Ce qui correspond à une amélioration de 17 centimètres.

5.3 Analyse des résultats en fonction des hypothèses initiales

Initialement, nous avons émis qu’une précision inférieure à 5 mètres était possible. À l’aide de la localisation par empreinte, constituée de positions de références avec un intervalle de 2 mètres, nous avons obtenue pour dix positions sur onze la précision attendue. À l’aide d’un filtre $\alpha - \beta$ l’erreur moyenne se trouve sous la bar des trois mètres où 45% des positions mesurées ont une précision inférieure à deux mètres. Les hypothèses de précisions sont donc confirmées.

6 Conclusion

6.1 Évaluation des limites et des incertitudes

Le prototype développé permet de se localiser dans un environnement à 2 dimensions. Cependant, le logiciel demande une prise de mesure qui se trouve assez fastidieuse. Un changement d'environnement impliquerait un changement de l'interface graphique ainsi que quelques modifications au niveau du code de localisation (les informations relatives aux points d'accès et à la superficie de la surface) sans oublier une nouvelle prise de mesure (!). De plus, les obstacles fixes (murs, poteaux, ...) n'ont pas été pris en compte lors du développement du logiciel, donc la position obtenue peut être quelque fois insensée.

6.2 Porposition des travaux subséquents

Dans le même thème, les travaux subséquents pourront permettre une amélioration du logiciel permettant d'y intégrer plusieurs lieux. On pourrait aussi trouver un moyen d'automatiser la prise de mesure. De manière à porter l'application sur d'autres appareils, il faudrait ajouter un mode de calibration permettant ainsi d'utiliser les données de la phase mesures pour plusieurs appareils distincts. De manière à augmenter la précision plusieurs mesures peuvent être explorées.

- Diminuer la distance entre les positions de références.
- Utiliser l'algorithme $K - NN$ avec une pondération.
- Intégrer des capteurs inertiels, au moins pour savoir si la personne est statique ou non.
- Ajouter de l'intelligence au logiciel en commençant par prendre en compte les obstacles fixes.
- Utiliser le CSI (Chanel State information) en plus du RSS [15].

7 Apprentissage continu

7.1 Résumé des nouvelles connaissances et habiletés acquises durant le projet

Ce projet m'a principalement permis d'en apprendre davantage sur le fonctionnement des systèmes de localisations. Au niveau théorique, pour comprendre la provenance du filtre $\alpha - \beta$ j'ai été introduit aux filtres de Kalman. Au niveau pratique, ce projet m'a permis d'améliorer des acquis en programmation. De plus, en utilisant la localisation par empreinte j'ai été amené à analyser beaucoup de données. En somme, le projet m'a permis d'acquérir une certaine autonomie, d'apprendre à effectuer des recherches et sélectionner des articles sur diverses

sujets et m'a permis d'en apprendre davantage sur le cycle de développement d'un produit.

Références

- [1] T. Benedict and G. Bordner. Synthesis of an optimal set of radar track-while-scan smoothing equations. *IRE Transactions on Automatic Control*, 7(4) :27–32, 1962.
- [2] Martin Casador. The sniffer’s guide to raw traffic, 2001. [Online ; accessed 7-December-2017].
- [3] Michael Gauthier. Wireless networking in the developing world, 2009. [Online ; accessed 7-December-2017].
- [4] Andrea Goldsmith. Wireless communications, 2005.
- [5] Dr Steven Gordon. Capturing wireless lan packets in monitor mode with iw, 2015. [Online ; accessed 7-December-2017].
- [6] P.D. Groves. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition* :. GNSS/GPS. Artech House, 2013.
- [7] I. Guvenc. Enhancements to rss based indoor tracking systems using kalman filters. In *In GSPx & International Signal Processing Conference*, 2003.
- [8] Insoft. Bluetooth low energy beacons, 2017. [Online ; accessed 7-December-2017].
- [9] Insoft. Indoor positioning with ultra-wideband, 2017. [Online ; accessed 7-December-2017].
- [10] Yoshio Kosuge and Masayoshi Ito. Evaluating an $\alpha\beta$ filter in terms of increasing a track update-sampling rate and improving measurement accuracy. *Electronics and Communications in Japan (Part I : Communications)*, 86(10) :10–20, 2003.
- [11] Forbes H.O. Maycotte. Beacon technology : The where, what, who, how and why, 2015. [Online ; accessed 7-December-2017].
- [12] Robert Penoyer. The alpha-beta filter, 1993. [Online ; accessed 7-December-2017].
- [13] M. Taboubi. Indoor localization using wi-fi signals. Master’s thesis, École Polytechnique de Montréal, 2014.
- [14] Paul A Zandbergen. Accuracy of iphone locations : A comparison of assisted gps, wifi and cellular positioning. *Transactions in GIS*, 13 :5–25, 2009.
- [15] L. Zhao, H. Wang, P. Li, and J. Liu. An improved wifi indoor localization method combining channel state information and received signal strength. In *2017 36th Chinese Control Conference (CCC)*, pages 8964–8969, 2017.

A Documents supplémentaires

Le code source des logiciels se trouve sur mon profil *GitHub*. Un vidéo montrant le fonctionnement du prototype est sur *Youtube*.