OÙ SUIS-JE?

Pour l’Expo-sciences régionale 2014 - UMCE

Par

Manuel Dionne

10e année

Université de Moncton Campus d’Edmundston

vendredi, 21 février 2014

Table des matières

Introduction 3

Problématique 3

Hypothèse 3

Matériaux utilisés4

Procédure 5

**Résultats 6**

Conclusion .6

Bibliographie .7

**Introduction**

**Problématique**

Depuis des dizaines d’années les humains on utilisé des techniques de positionnement peux efficace. Ils naviguent les mers en utilisant les étoiles, la boussole et l’astrolabe, des technologies qui sont passé date. Dans les années de la guerre froide le besoin de savoir où on est devient plus pressant car les États-Unis d’Amérique veulent pouvoir se défendre en cas d’attaque nucléaire. Pour lancer les missiles balistiques intercontinentaux, il faut un système de positionnement global pour les guider à leur objectif. Pas seulement les missiles ont besoin d’être guidés, mais aussi les navires et les sous-marins qui les transportent. Ils commencent à le construire dès 1973. Ce système de positionnement devient complètement opérationnel en 1995 et est maintenant ouvert pour l’utilisation non-militaire.

**Hypothèse**

Au début je croyais que le GPS et la triangulation dépendait sur un système d’angles et de pentes mathématiques. Cela marchait en mesurant la direction des signaux des satellites et le mettre sur un graphe. Pour trouver la position de la personne, il fallait trouver ou tous les équations se croisaient. Pour faire ceci, il faillait seulement la direction du satellite. La seule façon de prouver mon hypothèse était de faire de la recherche et construire mon propre système de positionnement.

**Matériaux utilisés**

Le matériel nécessaire pour créer un système de positionnement à taille réduite est : des pseudo-satellites qui transmettent des ondes radio, un ordinateur récepteur et quelques autres morceaux divers.

Pour les pseudo-satellites, j’ai choisi d’utiliser des ordinateurs miniatures Raspberry Pi™. Ces petits ordinateurs correspondent à la tâche parfaitement, ils utilisent beaucoup moins d’électricité pour fonctionner, peuvent transmettre des ondes radio et sont faciles à programmer.

L’ordinateur réceptrice est mon ordinateur de maison où un ordinateur portable emprunté de l’école. Ils vont capturer les ondes grâce à une antenne RTL-SDR muni d’une puce R820T.

Les différentes pièces diverses pour compléter mon projet sont : des pièces LEGO™ pour déguiser nos pseudo-satellites, deux Pi Cobbler Breakout Boards pour attacher deux antennes à nos Raspberry Pis et finalement beaucoup de différents câbles d’alimentations pour les électroniques impliqués.

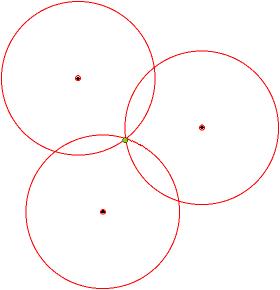
Pour transmettre sur les ondes sonores, j’ai utilisé des haut-parleurs et un microphone.

Les logiciels que je n’ai **pas** écrits sont : Rasbian, pyglet, minimodem, PiFmDma, nano, Python 2.7, GQRX, rtl\_fm et Kali Linux.

**Procédure**

Premièrement j’ai écrit un logiciel démontrant comment le système de triangulation fonctionne graphiquement en utilisant le module pyglet pour Python v2.7.

Deuxièmement, j’ai écrit divers logiciels pour faire les Raspberry Pis transmettre leur position et le temps sur 103.3mHz à toutes les deux secondes grâce à PiFmDma pour la modulation FM. J’ai ensuite écrit un autre logiciel pour encoder les signaux en AFSK avec minimodem à 110 baud.



L’ordinateur récepteur écoute sur 103.3mHz avec GQRX ou rtl\_fm. Ensuite, il décode les signaux en utilisant minimodem et extrait la position et le temps

Un exemple graphique de l’algorithme de triangulation

des signaux. Ces variables sont mis dans un algorithme pour calculer la position du récepteur puis mettre à jour le logiciel graphique du départ.

**Résultats**

Après avoir mis la procédure en action et avoir programmée tout ce que j’avais besoin, j’ai mis mon système en marche. Les Raspberry Pis ont transmis sur 103.3mHz comme prévu alors j’ai mis le récepteur en marche. Le récepteur, après quelques modifications, a put recevoir les signaux des Raspberry Pis.

J’ai écrit un logiciel pour calculer la distance entre les deux ordinateurs. Tout de suite je me suis aperçu que le temps de voyage variait entre les signaux. Alors je me suis mis à faire des modifications sur le système de transmission pour diminuer le temps de calcul de temps avec minimodem. Même avec les modifications, il semble que les ondes radios sont trop vite pour les horloges peux précis de mes ordinateurs. Alors je me suis mis à penser de d’autres moyens de transmissions sans-fil. J’ai décidé d’essayer les ondes sonores parce qu’il voyage a 300m/s et pas 300 000 000m/s comme les ondes radio.

J’ai modifié mon logiciel de transmetteur pour transmettre des ondes sonores sur 20mHz, une fréquence inaudible à l’oreille humaine. Mon récepteur, muni d’un microphone, capta les ondes et calcula la distance. La distance mesurée fut environ 120 mètres, alors que dans la réalité les deux étaient qu’a quelques centimètres de distance.

**Conclusion**

En conclusion, je raisonne que les imprécisions sont de faute de soit le temps de calcul du programme aplay ou minimodem. Une autre raison serait que les horloges sur les ordinateur soient passé précis pour calculer des distances si près.

Une chose qui pourrait peut être amélioré les résultats serait d’installer un « real-time operating system » ou un système d’exploitation qui ne partage pas le CPU entre plusieurs programmes en même temps, mais je n’ai pas exploré l’idée.

Peut importe, le système de triangulation utilisé par le GPS est complètement différent que j’avais anticipé. Il ne compte pas sur les pentes et les angles, mais sur l’intersection de cercles où le rayon est la distance entre le récepteur et le satellite. Cette méthode est vraiment plus efficace et simple que je ne pensais.

**Bibliographie**

WIKIPEDIA, Global Positioning system – Wikipedia[en ligne]. http://en.wikipedia.org/wiki/Global\_Positioning\_System[consulté le 2/21/2014]

LEVINE, John R. et Margaret YOUNG. Unix for Dummies. A Reference for the Rest of Us! Foster City, IDG Books Worldwide Inc., « For dummies », 1995, 376 pages.

GITHUB, Richard S. PiBits/PiFmDma at master · richardghirst/PiBits · GitHub[document éléctronique]. 2014, https://github.com/richardghirst/PiBits/blob/master/PiFmDma/PiFmDma.c

RASPBERRY PI FORUM, Rapberry Pi – View topic[en ligne]. http://www.raspberrypi.org/phpBB3/viewtopic.php?f=63&t=69786[2/17/2014]