

2ème année  
Télécommunications  
Semestre 7  
Novembre 2015

# **Simulation d'un émetteur / récepteur ADS-B**

Communications  
Numériques – Projet TS226

*Benjamin BONNOTTE & Abdoul GUISET*

# Table des matières

<b>Introduction .....</b>	<b>2</b>
<b>II. Partie 1 – Couche Physique ADS-B.....</b>	<b>3</b>
2.1 Travail préliminaire.....	3
2.2 Simulation d'émission/réception de signaux modulés avec une PPM.....	7
2.3 Synchronisation en temps et en fréquence .....	9
<b>III. Partie 2 – Traitement / Décodage de signaux réels .....</b>	<b>11</b>
3.1 Structure des trames ADS-B .....	11
3.2 Extraction des données .....	11
<b>IV. Partie 3 – Affichage des trajectoires.....</b>	<b>13</b>

# Introduction

Après avoir étudié de façon théorique les différentes chaînes de communication numériques en première année de formation ce projet en ait la traduction pratique. Nous allons donc à travers *Matlab* pourvoir simuler un émetteur récepteur de données ADS-B, cela appliqué au décodage de trajectoire d'avions.

Nous étudierons donc ce sujet à travers trois parties détaillées ci-après.

## II. Partie 1 – Couche Physique ADS-B

### 2.1 Travail préliminaire

*Étude théorique sur les signaux transmis ADS-B sont modulés en position d'amplitude PPM.*

#### Question 1:

Montrons que :

$$s_l(t) = 0.5 + \sum_{k \in \mathbb{Z}} A_k p(t - kT_s)$$

Le signal :  $s_l(t) = \sum_{k \in \mathbb{Z}} p b_k(t - kT_s)$  se décompose sous la forme :

$$s_l(t) = \sum_{k \in \mathbb{Z}, b_k=0} p_0(t - kT_s) + \sum_{k \in \mathbb{Z}, b_k=1} p_1(t - kT_s)$$

Ou encore :

$$s_l(t) = \sum_{k \in \mathbb{Z}, b_k=0} \frac{p_0(t) + p_1(t)}{2} + \frac{p_0(t) - p_1(t)}{2} + \sum_{k \in \mathbb{Z}, b_k=1} \frac{p_0(t) + p_1(t)}{2} + \frac{p_1(t) - p_0(t)}{2}$$

$$s_l(t) = \sum_{k \in \mathbb{Z}} \frac{p_0(t) + p_1(t)}{2} + \sum_{k \in \mathbb{Z}, b_k=0} p(t) - \sum_{k \in \mathbb{Z}, b_k=1} p(t)$$

En effet :

$$\sum_{k \in \mathbb{Z}} \frac{p_0(t) - p_1(t)}{2} = \frac{1}{2}$$

En posant :  $A_k = \begin{cases} 1 & b_k = 0 \\ -1 & b_k = 1 \end{cases}$

En conclusion :  $s_l(t) = 0.5 + \sum_{k \in \mathbb{Z}} A_k p(t - kT_s)$  (1)

La modulation obtenue avec  $s_l(t) - 0.5$  est équivalente à une modulation 2-PAM.

#### Question 2:

Les composantes  $p_0(-t)$  et  $p_1(-t)$  représentent les filtres adaptés. L'intérêt d'utiliser de tels filtres en réception est d'éviter d'avoir de l'IES (de minimiser l'impact du bruit) en respectant le critère de Nyquist.

#### Question 3:

Les deux couples de filtres vérifient le critère de Nyquist, une application triviale de ce critère le prouve.

Question 4:

Montrons que la moyenne du signal  $s_l(t)$  est indépendante du temps.  
On a:  $m_{sl}(t) = E[s_l(t)] = E[0.5 + \sum_{k \in \mathbb{Z}} A_k p(t - kT_s)]$   
Comme les  $b_k$  sont i.i.d d'après l'énoncé, les  $A_k$  le sont aussi. Par conséquent, leur moyenne est nulle. On en conclut par linéarité de l'espérance :

$$m_{sl} = 0.5 \quad (2)$$

Question 5:

Calcul de la fonction d'autocorrélation du signal  $s_l(t)$  :

$$\begin{aligned} R_{sl}(t, \tau) &= E[s_l(t)s_l^*(t + \tau)] \\ &= E\left[\frac{1}{4}\right] + E\left[\frac{1}{2} \sum_{k \in \mathbb{Z}} A_k p(t - kT_s)\right] + E\left[\sum_{k \in \mathbb{Z}} \sum_{l \in \mathbb{Z}} A_k A_l p(t - kT_s) p(t - lT_s + \tau)\right] \\ &\quad + E\left[\frac{1}{2} \sum_{k \in \mathbb{Z}} A_k p(t - kT_s + \tau)\right] \\ &= \begin{cases} 0.25 \text{ si } k \neq l \text{ car } E[A_k]E[A_l] = 0 \text{ par indépendance} \\ 0.25 + \sum_{k \in \mathbb{Z}} \sigma_A^2 p(t - kT_s) p(t - kT_s + \tau) \text{ si } k = l \end{cases} \quad (3) \end{aligned}$$

Question 6:

Pour montrer que notre signal est cyclo-stationnaire, deux points sont à vérifier :

- $m_{sl}(t) = m_{sl}(t + T_s)$  (Trivial)
- $R_{sl}(t + T_s, \tau) = R_{sl}(t, \tau)$ . On retrouve les mêmes calculs qu'effectuer précédemment, un changement de variable  $t = t + T_s$  nous permet de conclure.

La période de stationnarité est donc égal à  $T_s$ .

Question 7:

La fonction d'autocorrélation moyennée sur une période de stationnarité est donnée par la relation suivante :

$$\begin{aligned} \widetilde{R_{sl}}(t, \tau) &= \frac{1}{T_s} \left[ \int_0^{T_s} R_{sl}(t, \tau) dt \right] \\ \widetilde{R_{sl}}(t, \tau) &= \frac{1}{T_s} \left[ \int_0^{T_s} \frac{1}{4} + \sum_{k \in \mathbb{Z}} \sigma_A^2 p(t) p(t - kT_s + \tau) dt \right] \end{aligned}$$

On pose:  $t' = t - kT_s$

$$\widetilde{R_{sl}}(t', \tau) = \frac{1}{4} + \frac{1}{T_s} \sum_{k \in \mathbb{Z}} \int_{-kT_s}^{(1-k)T_s} \sigma_A^2 p(t') p(t' + \tau) dt$$

$$R_{sl}(\widetilde{t'}, \tau) = \frac{1}{4} + \frac{\sigma_A^2}{2T_s} \left[ \int_{\frac{1}{2}T_s}^{T_s} p(t + \tau) dt \right] - \frac{\sigma_A^2}{2T_s} \left[ \int_0^{0.5T_s} p(t + \tau) dt \right]$$

On pose:  $t = t' + \tau$

$$R_{sl}(\widetilde{t}, \tau) = \frac{1}{4} + \frac{\sigma_A^2}{2T_s} \left[ \int_{\frac{1}{2}T_s + \tau}^{T_s + \tau} p(t + \tau) dt \right] - \frac{\sigma_A^2}{2T_s} \left[ \int_{\tau}^{0.5T_s + \tau} p(t) dt \right]$$

Plusieurs cas sont à considérer suivant les valeurs de  $\tau$  :

$$\left\{ \begin{array}{l} R_{sl}(\widetilde{t}, \tau) = 0.25 \text{ si } \tau < -T_s \text{ ou si } \tau > T_s \\ R_{sl}(\widetilde{t}, \tau) = 0.25 + \frac{\sigma_A^2}{4T_s} (T_s - 3\tau) \text{ si } 0 < \tau < 0.5T_s \\ R_{sl}(\widetilde{t}, \tau) = 0.25 - \frac{\sigma_A^2}{4T_s} (T_s - \tau) \text{ si } 0.5T_s < \tau < T_s \\ R_{sl}(\widetilde{t}, \tau) = 0.25 - \frac{\sigma_A^2}{4T_s} (T_s + \tau) \text{ si } -T_s < \tau < -0.5T_s \\ R_{sl}(\widetilde{t}, \tau) = 0.25 + \frac{\sigma_A^2}{4T_s} (T_s + 3\tau) \text{ si } -0.5T_s < \tau < 0 \end{array} \right. \quad (4)$$

#### Question 8:

Calculons tout d'abord la transformée de Fourier du signal  $p(t)$  on a :

$$p(t) = -0.5u(t) + u(t - \frac{T_s}{2}) - 0.5u(t - T_s)$$

$$p'(t) = -0.5\delta(t) + \delta(t - \frac{T_s}{2}) - 0.5\delta(t - T_s)$$

Comme :

$$TF\{p'(t)\} = j2\pi f TF\{p(t)\}$$

$$P(f) = -\frac{1}{j4\pi f} [1 - e^{-j\pi f T_s}]^2$$

D'où en factorisant par l'angle moitié on obtient :

$$P(f) = -\frac{e^{-j\pi f T_s}}{j4\pi f} \left[ 2j \sin\left(\frac{\pi}{2} f T_s\right) \right]^2$$

On peut à présent calculer la DSP :

$$\Gamma_{s1}(f) = TF\{R_{sl}(\widetilde{t}, \tau)\} = 0.25 + \delta(f) + \frac{\sigma_A^2}{T_s} TF\left\{ \int_{-\infty}^{\infty} p(t)p(t + \tau) dt \right\}$$

On reconnait un produit de convolution pour l'intégral et grâce aux calculs précédents on en conclut :

$$\Gamma_{s_1}(f) = 0.25\delta(f) + \frac{\sigma_A^2 (\pi f)^2 T_s^3}{16} \text{sinc}^4\left(\frac{\pi}{2} f T_s\right) \quad (5)$$

*Question 9:*

Pour reprendre les notations du cours, on pose  $g_0 = \mathbf{p0} * (-t)$  et  $g_1 = \mathbf{p1} * (-t)$  :

$$R_{nl',0} = R_{g_0}(\tau) \frac{N_o}{2} \text{ \& } R_{nl',1} = R_{g_1}(\tau) \frac{N_o}{2}$$

Or  $R_{g_0}(\tau)$  revient au calcul de la fonction d'autocorrélation d'une porte après un changement de variable en  $t'=-t$

Concernant la représentation, rappelons que la fonction d'autocorrélation d'une porte est une fonction triangle.

*Question 10 :*

La formule théorique de la probabilité d'erreur binaire dans le cas d'une modulation PPM est équivalente à celle d'une modulation 2-PAM comme vu à la question 1 :  $\mathbf{P_b} = \frac{1}{2} \text{erfc}\left(\sqrt{\frac{E_b}{N_o}}\right)$

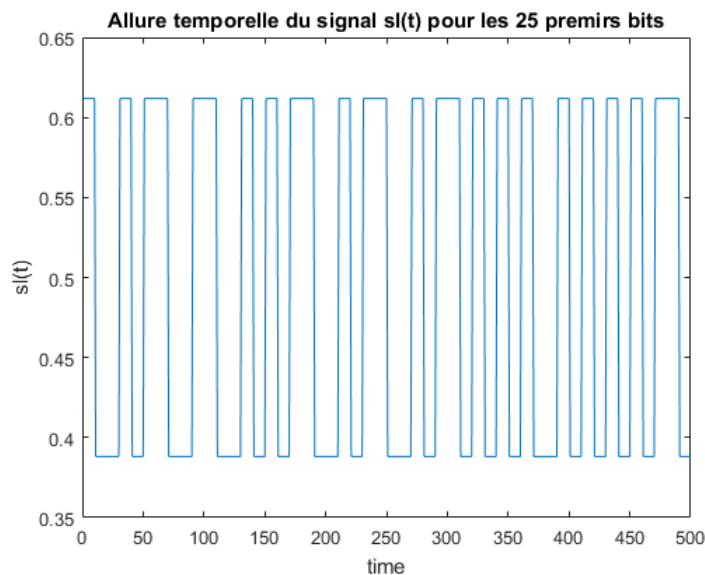
## 2.2 Simulation d'émission/réception de signaux modulés avec une PPM

*Implémentation et tests (diagramme de l'œil, DSP, TEB...) sur la chaîne de communication numérique dans le cas de signaux modulés avec une PPM.*

Récapitulatif des paramètres de simulation	
Fréquence d'échantillonnage	$f_e = \frac{1}{T_e} = 20\text{kHz}$
Débit symbole	$D_s = \frac{1}{T_s} = 1\text{MHz}$
Facteur de sur-échantillonnage	$F_{se} = \frac{T_s}{T_e} = 20$
Séquence de bits aléatoires et iid	$P(b_k=0) = P(b_k=1) = 1/2$
Nombre de points pour la FFT	$N_{fft} = 512$
Taille message à l'émission	<b>1000 bits</b>

### Question 11:

Après que la séquence de bits aléatoire soit passée par le bloc de Modulation PPM, nous obtenons le signal  $s_l(t)$ . La figure ci-après représente ce signal pour les 25 premiers bits émis.

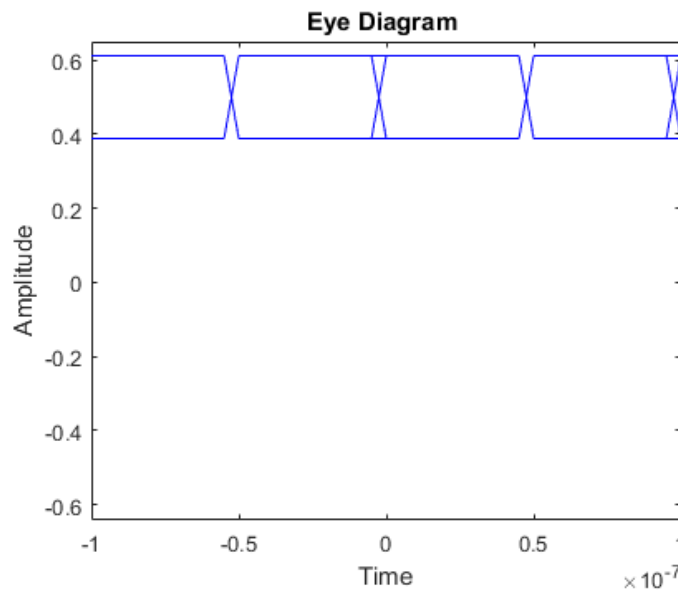


**Figure 1 :** Allure de  $s_l(t)$  pour les 25 premiers bits



### Question 12:

La figure suivante représente le **diagramme de l'œil** de notre signal  $s_i(t)$  sur les 100 premiers bits obtenus par superposition des signaux de durée finis, ici :  $2T_s$ . Le diagramme de l'œil va nous donner une information sur l'**IES** (Interférence entre symboles) au niveau temporel, autrement dit, une information sur la **pureté de notre signal**.

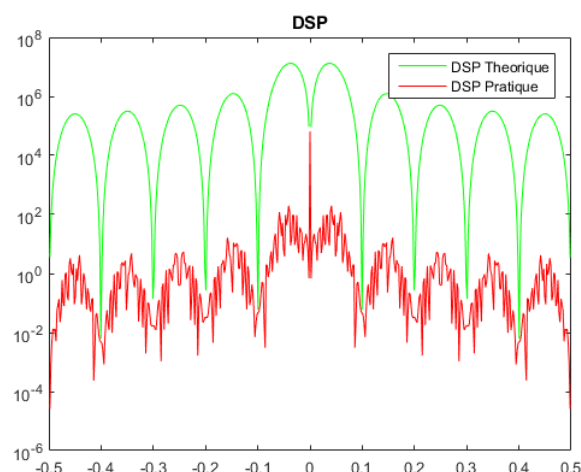


*Figure 2 : Diagramme de l'œil pour les 100 premiers bits envoyés*

Ici, nous pouvons constater au vue de la fermeture de «l'œil » que notre **IES = 0**. Il n'y a donc pas d'interférence entre symboles, notre signal est « pur ».

### Question 13:

A présent, nous allons calculer la **DSP** (Densité Spectral de Puissance) du signal :  $s_i(t)$  et tracer cette dernière sans fenêtre de pondération et sans chevauchement. Nous la comparerons avec la DSP théorique calculé en Question 8.



*Figure 3 : DSP du signal  $s_i(t)$  théorique et pratique*

La DSP théorique et la DSP pratique sont très proche malgré un certain décalage, ce qui valide notre modèle.

#### Question 14:

Désormais nous allons nous intéresser au TEB en fonction du rapport signal sur bruit  $\frac{Eb}{N_0}$  que nous allons faire varier de 0dB à 10dB par pas de 1dB. Afin de rendre cette valeur légitime nous allons la calculer pour un nombre d'erreur calculé supérieur à 100. Cette analyse est comparé à la valeur théorique du **Pb**.

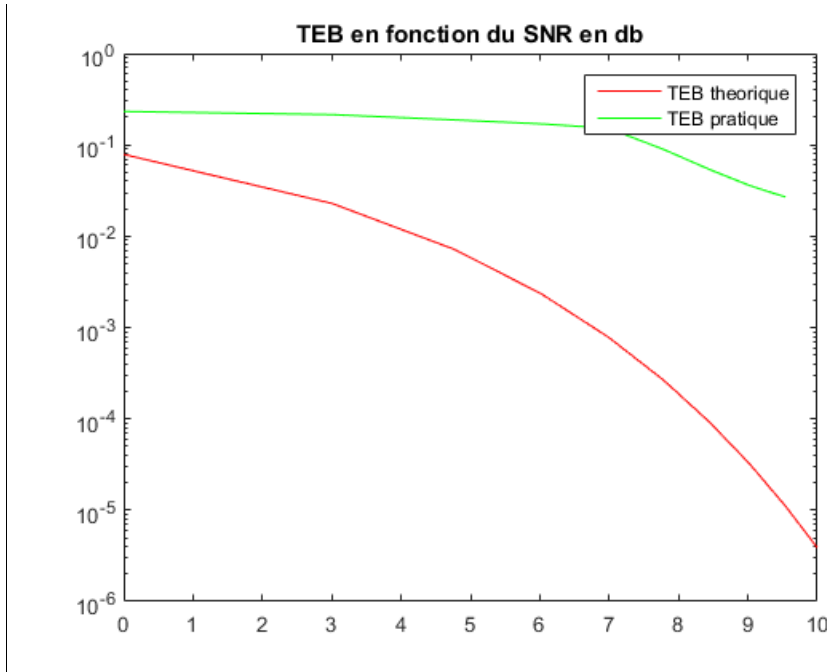


Figure 4 : Évolution du TEB théorique et pratique

Le TEB après expérimentation est donc plus élevé que le TEB théorique.

### 2.3 Synchronisation en temps et en fréquence

*Simulation du système avec prise en compte de deux nouveaux défauts : la synchronisation temporelle et la synchronisation fréquentielle.*

Rappelons que les deux effets cités ci-dessus sont pris en compte avec le modèle en bande de base suivant :

$$y_l(t) = s_l(t - \delta_t) e^{-j2\pi\delta_f t} + n_l(t)$$

#### Question 15:

Par l'inégalité de Cauchy-Schwarz, nous obtenons :

$$|\rho(\delta'_t, \delta'_f)| \leq \frac{\sqrt{\int_{\delta'_t}^{\delta'_t + T_p} |s_p(t - \delta'_t) e^{-j2\pi\delta'_f t}|^2 dt} \cdot \sqrt{\int_{\delta'_t}^{\delta'_t + T_p} |y_l(t)|^2 dt}}{\sqrt{\int_{\delta'_t}^{\delta'_t + T_p} |s_p(t)|^2 dt} \sqrt{\int_{\delta'_t}^{\delta'_t + T_p} |y_l(t)|^2 dt}}$$

Par changement de variable  $t' = t - \delta'_t$ , on obtient finalement :

$$|\rho(\delta'_t, \delta'_f)| \leq 1 \text{ Pour tout couple } (\delta'_t, \delta'_f).$$

L'égalité est obtenue lorsque les deux signaux  $y_l$  et  $s_p$  sont proportionnels.

#### Question 16:

Pour le calcul de la fréquence Doppler les données suivantes sont nécessaires :

- $V_D$  : Vitesse de déplacement de l'avion = **900 km/h**
- $f_{\text{emetteur}}$  : Fréquence porteuse des signaux ADS-B = **1090 Mhz**
- $c$  : Vitesse de la lumière = **299 792 458 m / s**

Nous appliquons donc la formule suivante :

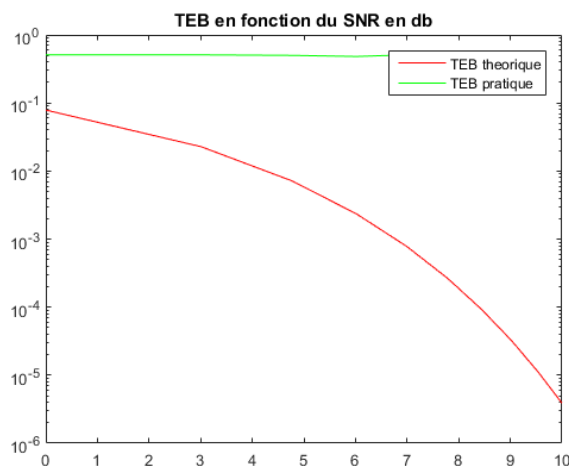
$$f_{\text{recepteur}} = f_{\text{emetteur}} \frac{c}{c - V_D}$$
$$\delta'_f = f_{\text{recepteur}} - f_{\text{emetteur}}$$

Après application numérique le résultat suivant est obtenu :

$$\delta'_f = 900 \text{ Hz}$$

#### Question 17:

Après avoir implémenté l'algorithme de synchronisation temps/fréquence nous allons nous intéresser de nouveau TEB en fonction du rapport signal sur bruit  $\frac{Eb}{N_0}$  que nous allons faire varier de 0dB à 10dB par pas de 1dB mais cette fois ci, en désynchronisant aléatoirement chaque nouvelle trame émise.



**Figure 5 :** Évolution du TEB selon des trames aléatoirement désynchronisées

Malgré une implémentation qui nous permet d'obtenir une estimation des valeurs de décalage en temps et en fréquence correct grâce au maximum de corrélation une erreur persiste au niveau du TEB pratique dû au fait que le nombre d'erreur calculé est insuffisant. Le résultat attendu est toujours que la courbe du TEB pratique surpasse celle du théorique.

## III. Partie 2 – Traitement / Décodage de signaux réels

### 3.1 Structure des trames ADS-B

*L'objectif de cette partie est de pouvoir tester le décodeur implémenté sur des signaux réels. Il s'agit ici d'extraire les informations d'une trame de type ADS-B pour pouvoir ensuite afficher les trajectoires d'un avion.*

#### Question 18:

Les valeurs FTC qui correspondent à des trames de position en vol sont situés de : **9 à 18** et **20 à 22**. Quant aux messages d'indentifications les valeurs de FTC correspondantes vont de : **1 à 4**.

### 3.2 Extraction des données

*Conversion du message binaire en registre, extraction des données sur les appareils concernées*

#### Question 19:

Dans cette partie, nous avons enregistré les informations liées à l'avion sous formes de registre, il a donc été nécessaire de déterminer pour chaque trame quels types d'informations ces bits véhiculaient. De manière générale, il y a 2 types :

- En vol.
- Identification.

#### (1) En vol

L'annexe A, nous donne les valeurs du champ FTC qui correspond à un type de trames. Le premier type donne alors des informations sur la l'adresse, la latitude, la longitude, l'altitude, le format, la trajectoire, des indicateurs horaires ainsi qu'une trajectoire. La fonction **bit2registre** implémenté s'occupe de récupérer les bits d'intérêts et de remplir le registre à condition que la trame ne contienne pas d'erreur. Le contrôle d'erreur a été effectué par bits de parité : Le polynôme générateur étant donné, nous avons implémenté ce détecteur d'erreur en utilisant les fonctions **crc.detector** et **detect**. De plus, les calculs de longitude, latitude et de l'altitude sont menés en suivant les étapes proposées dans les annexes B et C.

#### (2) Identification

Le deuxième type de message permet de savoir le nom de l'avion pour lequel on affiche la trajectoire. Il a été nécessaire d'effectuer un décodage des caractères pour retrouver le nom de l'avion correspondant. En effet, pour un message de type identification, les huit caractères sont codées sur un certain nombre de bits.

La fonction **decodeur** est celle qui effectue le décodage : les quatre premiers bits du caractère nous donne la ligne sur laquelle elle se trouve dans notre tableau fournit en annexe D et les deux derniers bits nous donne la colonne. Pour des soucis d'implémentation, nous avons préféré convertir les bits de chaque caractère en décimal pour pouvoir plus facilement les comparer. On obtient alors un registre final dont le nom est mise à jour :

```

registre =

    adresse: '726648'
    format: 'ADS-B'
    type: 'en vol'
    nom: {'AF255YO '}
    altitude: 9950
    timeFlag: 0
    cprFlag: 1
    latitude: 44.9767
    longitude: -0.7328
    trajectoire: [21x2 double]

```

Figure 6 : Registre mis à jour

#### Question 20:

Après avoir implémenté nos différentes fonctions, nous pouvons tester l'affichage de trajectoire après extraction des données. Le résultat suivant est alors obtenu :

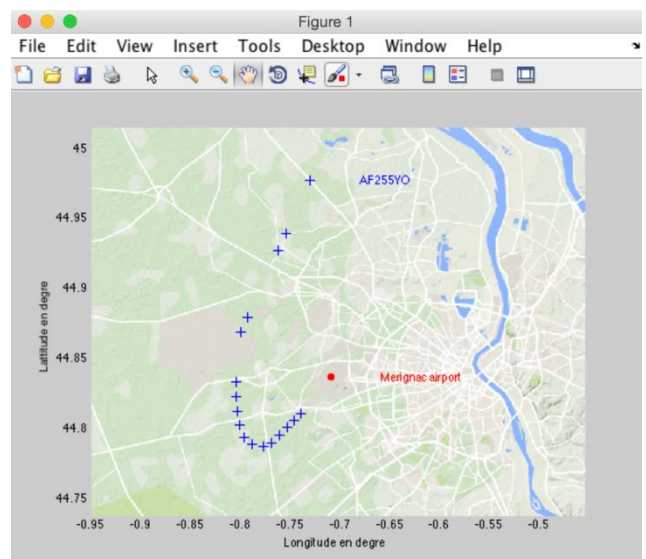
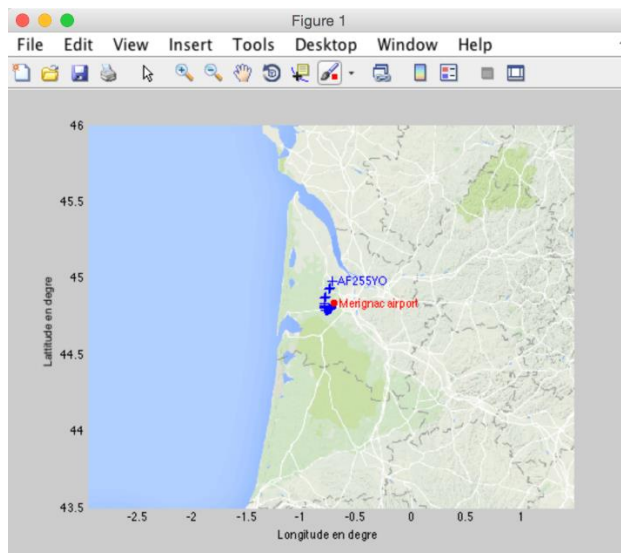


Figure 7 : Affichage de la trajectoire de l'avion

La trajectoire de l'avion est bien cohérente avec les données.

#### Question 21 (BONUS) :

A partir du nom de l'avion « AF255YO » nous pouvons facilement obtenir les informations concernant la compagnie, le modèle ou encore la route de l'avion :

---

**AIRFRANCE** / AF255YO - AIR FRANCE - AIRBUS A320 (A320) - F-HBND

---

## IV. Partie 3 – Affichage des trajectoires