



SORBONNE UNIVERSITÉ

RAPPORT DE STAGE M1

Positionnement Relatif

Ozgen Tunc Turker



supervisé par
Pacôme DELVA

October 31, 2020

Contents

	Page
1 Introduction	4
2 Satellites avec des récepteurs GNSS	4
3 Les Données de satellite Sentinel-3A	5
3.1 Pseudorange	5
3.2 La Position de Sentinel-3A	7
4 Les Données de satellite GPS	8
5 Lire Les Fichiers	10
5.1 RINEX et Éphémérides de satellite GPS	10
5.2 Éphémérides Orbitales Précises Fichier	10
5.3 La Trace des Données	10
6 Interpolation	11
6.1 Erreur	13
7 Les commentaires et Les Travaux Ultérieurs	14
A Appendix	18
A.1 Python Code	18

List of Figures

Figure 1	Le fichier RINEX, la tête et les observations de la première époque.	6
Figure 2	Precise Orbit Ephemerides (POE) Fichier Orbite	7
Figure 3	Éphémérides de satellite GPS (orbites)	9
Figure 4	Pseudorange de Sentinel-3A avec le satellite G01.	11
Figure 5	Pseudorange de Sentinel-3A avec le satellite G02.	11
Figure 6	Pseudorange de Sentinel-3A avec le satellite G02 et la différence entre la position de Sentinel-3A et du satellite G02.	11
Figure 7	Les interpolations	13
Figure 8	Pseudorange et la distance après l'interpolation.	13
Figure 9	Pseudorange et la distance après l'interpolation sans les points extérieurs.	13
Figure 10	L'erreur entre les pseudoranges et les distances.	14

Abstract

In the project, satellites equipped with GNSS receivers are searched. The Sentinel-3A satellite is chosen to study. The data transmitted by Sentinel-3A are read successfully and a very simplified comparison is tried to be done between the distance measured by the onboard instruments of Sentinel-3A and the distance calculated by the data received by ground-based devices. Data reading is done by using georinex package (for RINEX files) and ElementTree library. Missing points in the data are filled by interpolations.

Résumé

Dans le projet, les satellites équipés de récepteurs GNSS sont recherchés et trouvés. Le satellite Sentinel-3A est choisi pour étudier. Les données transmis par Sentinel-3A sont lues avec succès et une comparaison très simplifiée est tentée de faire entre la distance mesurée par les instruments à bord de Sentinel-3A et la distance calculée par les données reçues par des appareils basés au sol. La lecture des données est effectuée à l'aide du paquet georinex (pour les fichiers RINEX) et de la bibliothèque ElementTree. Les points manquants dans les données sont remplis par les interpolations.

1 Introduction

La plupart des satellites qui sont utilisés dans les systèmes GNSS (Géolocalisation et Navigation par un Système de Satellites) ont de transmetteurs de GNSS. Parallèlement les plupart des récepteurs de GNSS sont situés au sol. Donc c'est historiquement plus commun que les positions des objets spatiaux sont connus par les appareils basé au sol.

La motivation fondamentale du projet est d'avoir un système de positionnement sans avoir besoin des appareils au sol. C'est pourquoi il faut travailler avec les satellites qui ont des récepteur GNSS à bord.

Ici le but est d'abord de chercher et trouver les satellites qui ont des récepteur GNSS, puis de réussir à lire les fichiers transmis par des satellites et de comparer les données de satellites avec les données obtenues par les appareils basé au sol.

2 Satellites avec des récepteurs GNSS

Tous les résultats de la recherche de trouver des satellites équipés avec des récepteurs GNSS sont donnés dans le tableau 1

	SATELLITE NAME	ORBIT	GNSS RECEIVER	CONSTELLATION	DATA
1	TechDemoSat-1[1][2]	LEO	GPS (SGR-ReSI)	No	-
2	CYGNSS[3] [4]	LEO 500km	GPS	8 Satellites	✓
3	CryoSat-2 [5]	LEO	GPS (Doris)	No	✓
4	Jason-3 [6]	LEO 1,336km	GPS (DORIS+POD)	No	✓
5	MetOp [7][8]	SSO 817km	GPS(GRAS)	3 Satellites	✓
6	GRACE-FO [9] [10]	LEO 500km	GPS	Twin	✓
7	SMOS[11][12]	SSO 760km	GPS	No	-
8	ORBCOMM[13]	LEO	GPS	Yes	-
9	Astrocast CubeSat [14] [15]	LEO	Galileo	No	-
10	MMS [16][17]	Highly elliptical orbit	GPS	No	✓
11	GOES-16 & 17 [18]	GEO	GPS (GPSR)	Yes	-
12	DRAGONSat	LEO	GPS	2 Satellites	-
13	PicoSAT	LEO	GPS	No	-

14	LandSAT 7 & 8 [19]	SSO 700km	GPS	No	✓
15	TET-1 [20]	LEO	GPS (NOX)	No	-
16	Sentinel-3 [21] [22]	SSO 800km	GPS + Galileo	2 Satellites	✓
17	SWARM [23] [24]	Polar orbit	GPS(AOCS)	No	✓
18	ICESAT-2 [25]	LEO 500km	GPS	No	✓
19	EarthCare	LEO*	GPS?	No	?
20	Pléiades [26]	SSO 700km	GPS(DORIS)	2 Satellites	-
21	RADARSAT [27][28]	SSO 600km	GPS	3 Satellites	✓
22	SAC-D[29][30]	SSO 670km	GPS (ROSA+TDP)	No	✓
23	Megha-Tropiques [31]	circular 860km	GPS(ROSA)	No	✓
24	Oceansat-2[32][33]	SSO 720km	GPS(ROSA)	No	✓
25	COSMO-SkyMed [34] [35]	SSO 620km	GPS(SSTI)	4 Satellites	✓

Table 1: Les Satellites avec des récepteurs GNSS

3 Les Données de satellite Sentinel-3A

Le satellite Sentinel-3A est choisi pour étudier. Comme il est possible de voir dans le table 1, les satellites de Sentinel ont des recepteur de GNSS, GPS.

Les données sont accessibles par le site <https://scihub.copernicus.eu/gnss/#/home>. Pour y accéder, le nom d'utilisateur et le mot de passe sont *gnssguest*.

3.1 Pseudorange

Pour les données pseudorange, le fichier “AUX_GNSSRD” à la date 31-08-2018 est utilisé. Ce fichier qui est téléchargé avec une extension .DBL et qui peuvent être ouvrir avec un éditeur de texte, est le fichier de RINEX dont le tête et les observation de 3 premières époques se sont trouvés dans l'image 1. Ce qui est important de remarquer que les époques ont une intervalle d'une seconde.

```

 3.03      0      G   RINEX VERSION / TYPE
          GMV   20180907 024155 UTC PGM / RUN BY / DATE
SENTINEL-3A                               MARKER NAME
SPACEBORNE                                MARKER TYPE
ESA-EUMETSAT    ESA-EUMETSAT             OBSERVER / AGENCY
GPSR-N        SENTINEL-3 GPSR            TBD REC # / TYPE / VERS
0           SEN-3A-GPSA              ANT # / TYPE
1503586.0740 -1479013.7000  6855886.8900 APPROX POSITION XYZ
 0.0000     0.0000     0.0000  ANTENNA: DELTA H/E/N
 0.0000     0.0000     0.0000  ANTENNA: DELTA X/Y/Z
 0.0000     0.0000     0.0000  ANTENNA: B.SIGHT XYZ
 0.0000     0.0000     0.0000  CENTER OF MASS: XYZ
G 7 C1C L1C S1C C1W C2W L2W S2W   SYS / # / OBS TYPES
2018 8 31 0 0 0.0000000   GPS TIME OF FIRST OBS
1                                         RCV CLOCK OFFS APPL
+Observations timestamp based on polynomial GPS time COMMENT
Receiver clock offset to GPS time fit using 4th order pol COMMENT
Clk_offset(i) = sum(k=0..4) { a_k * ( mid(i) - mid(0) )^k } COMMENT
mid(0) = 6816.958333 days COMMENT
 a_0 = -9.355600e-19 ns COMMENT
 a_1 = 1.324295e-12 ns/s COMMENT
 a_2 = 5.688971e-16 ns/s^2 COMMENT
 a_3 = 9.172425e-18 ns/s^3 COMMENT
 a_4 = 3.584054e-18 ns/s^4 COMMENT
G L1C                               SYS / PHASE SHIFT
G L2W                               SYS / PHASE SHIFT
0                                     GLONASS SLOT / FRQ #
+Constant calibrated HW bias removed from code observations COMMENT
 Bias: 23.5 m                         COMMENT
                                         END OF HEADER
> 2018 08 31 00 00 0.0000000 0 8  0.000055179015
G02 19274009.15009 -12747861.38909 54.40009 19274008.94009 19274004.52507 -27944875.29007 46.78007
G06 21961788.11607 -6885899.08207 45.99007 21961789.19807 21961789.45905 -23947330.17705 32.38005
G09 22481472.56207 -1358401.72907 44.08007 22481473.88807 22481473.48604 -19378473.92204 29.21004
G12 23181131.83507 -1728088.97107 44.14007 23181132.58507 23181129.25204 -19769530.16804 24.37004
G23 22803337.36107 3343170.94507 44.56007 22803337.11607 22803331.98604 -15805529.67004 27.06004
G25 21495987.18508 -13517676.28808 49.15008 21495987.83108 21495988.23206 -28750424.21606 37.18006
G29 21456137.24008 -3095773.39908 52.13008 21456137.67608 21456135.58206 -21038070.05706 38.46006
G31 23064774.18407 -21919436.43207 42.85007 23064774.48107 23064771.58403 -20842995.95903 23.53003
> 2018 08 31 00 00 1.0000000 0 8  0.000055194343
G02 19272473.26709 -12755931.98909 54.37009 19272473.14509 19272468.73007 -27951164.06707 46.80007
G06 21965531.81407 -6866224.88107 46.01007 21965533.03607 21965533.43705 -23931999.62905 32.33005
G09 22486670.16007 -1331087.26707 44.18007 22486671.69507 22486671.29484 -19357189.92604 29.10004
G12 23174878.47807 -1760951.27907 44.20007 23174879.29807 23174876.01804 -19795137.15904 24.29004
G23 22810517.68707 3380905.30007 44.42007 22810517.84507 22810512.55704 -15776126.27404 26.91004
G25 21491798.25908 -135339690.81908 49.15008 21491798.55608 21491799.04406 -28767578.39506 37.21006
G29 21453937.39408 -3107333.30208 52.16008 21453937.97008 21453935.77106 -21047077.77206 38.38006
G31 23066645.57907 -2199604.17507 42.98007 23066645.85807 23066642.56003 -20835334.46203 23.93003
> 2018 08 31 00 00 2.0000000 0 8  0.000055209671
G02 19270949.94009 -12763939.08609 54.40009 19270949.48709 19270945.07207 -27957403.36507 46.81007
G06 21969283.82807 -6846510.70307 45.94007 21969284.63007 21969284.97905 -23916637.93305 32.29005
G09 22491875.07807 -1303740.40107 44.02007 22491875.55007 22491875.30504 -19335880.68104 29.04004
G12 23168627.95707 -1793800.51907 44.15007 23168628.25407 23168625.21704 -19820733.97404 24.55004
G23 22817702.12407 3418659.68107 44.48007 22817702.10607 22817697.16804 -15746707.27804 26.80004
G25 21487614.35008 -13561679.56808 49.24008 21487614.24508 21487614.69906 -28784712.48806 37.22006
G29 21451744.31008 -3118858.89908 52.16008 21451744.72908 21451742.51306 -21056058.75706 38.45006
G31 23068520.19407 -2099757.97407 42.99007 23068519.82707 23068516.25003 -20827662.10103 23.39003

```

Figure 1: Le fichier RINEX, le tête et les observations de 3 premières époques.

La tête du fichier finit par le comment “END OF HEADER” et les lignes suivant sont des observations dérivées par les 8 satellites différents. Ces satellites qui peuvent changer entre les époques sont indiqués par 3 caractères; la lettre G qui signifie la transmetteur de GPS et en plus deux chiffres qui signifient leur nombre de PRN. Les lignes qui commencent par le symbol “>” mentionnent l’époque des observations avec l’ordre: an, mois, jour, heure, minute, second, drapeau d’époque, nombre de satellites observés et décalage de l’horloge du récepteur [36].

Pour identifier les colonnes d’observations, il faut voir la ligne “SYS / / OBS TYPES” qui a des informations suivant; code du système de satellite (GPS dans ce cas), nombre de satellites, type des observations. Dans l’image 1, il y a 7 observations qui sont C1C, L1C, S1C, C1W, C2W, L2W, S2W. Dans les 3 caractères code: la première signifie type d’observation (le tableau 2) la deuxième est la bande de fréquence (de 1 à 8) et la dernière est la mode de suivi [36].

C	Pseudorange
L	Phase Porteuse
S	Puissance du Signal

Table 2: Les codes du type d’observation.

Donc la première colonne, ce que le projet intéresse, C1C, signifie le pseudorange dans la bande de fréquence L1 et *C code-based*.

3.2 La Position de Sentinel-3A

Pour avoir les données de la position de Sentinel-3A, il faut accéder le même site mentionné dans la section précédente. Il faut chercher les données de même jour *31-08-2018* mais cette fois le nom de fichier est “AUX_POEORB”. Ce fichier est un fichier *xml* qui a une extention *.EOF* qui peut être ouvert par un éditeur de texte. Il ressemble à la figure 2.

```
<?xml version="1.0" ?>
<Earth_Explorer_File>
  <Earth_Explorer_Header>
    <Fixed_Header>
      <File_Name>S3A_OPER_AUX_POEORB POD __20180925T071034_V20180830T215942_20180831T235942_DGNS</File_Name>
      <File_Description>Precise Orbit Ephemerides (POE) Orbit File</File_Description>
      <Notes></Notes>
      <Mission>Sentinel-3A</Mission>
      <File_Class>OPER</File_Class>
      <File_Type>AUX_POEORB</File_Type>
      <Validity_Period>
        <Validity_Start>UTC=2018-08-30T21:59:42</Validity_Start>
        <Validity_Stop>UTC=2018-08-31T23:59:42</Validity_Stop>
      </Validity_Period>
      <File_Version>0001</File_Version>
    <Source>
      <System>POD_-</System>
      <Creator>POD_-</Creator>
      <Creator_Version>1.5.0</Creator_Version>
      <Creation_Date>UTC=2018-09-25T07:10:34</Creation_Date>
    </Source>
  </Fixed_Header>
  <Variable_Header>
    <Source_Data>DGNS</Source_Data>
    <Ref_Frame>EARTH_FIXED</Ref_Frame>
    <Time_Reference>UTC</Time_Reference>
  </Variable_Header>
</Earth_Explorer_Header>
<Data_Block type="xml">
  <List_of_OSVs count="9361">
    <OSV>
      <TAI>TAI=2018-08-30T22:00:19.000000</TAI>
      <UTC>UTC=2018-08-30T21:59:42.000000</UTC>
      <UT1>UT1=2018-08-30T21:59:42.063756</UT1>
      <Absolute_Orbit>+13206</Absolute_Orbit>
      <X unit="m">-5389594.094949</X>
      <Y unit="m">707885.429329</Y>
      <Z unit="m">-4702577.386401</Z>
      <VX unit="m/s">4973.147378</VX>
      <VY unit="m/s">1236.663864</VY>
      <VZ unit="m/s">-5518.685352</VZ>
      <Quality>NOMINAL</Quality>
    </OSV>
    <OSV>
      <TAI>TAI=2018-08-30T22:00:29.000000</TAI>
      <UTC>UTC=2018-08-30T21:59:52.000000</UTC>
      <UT1>UT1=2018-08-30T21:59:52.063756</UT1>
      <Absolute_Orbit>+13206</Absolute_Orbit>
      <X unit="m">-5339567.128191</X>
      <Y unit="m">720177.692160</Y>
      <Z unit="m">-4757510.585792</Z>
      <VX unit="m/s">5032.152681</VX>
      <VY unit="m/s">1221.752632</VY>
      <VZ unit="m/s">-5467.856829</VZ>
      <Quality>NOMINAL</Quality>
    </OSV>
```

Figure 2: Precise Orbit Ephemerides(Éphémérides orbitales précises) (POE) Fichier Orbite

Dans l'image 2, il y a encore une tête du fichier et les 2 époques d'observation. Dans les époques il y a 3 différents types de temps qui sont TAI: "Temps Atomique International", UTC: "Universel Temps Coordonné" et UT1: "Universal Time (temps universel)". L'intervalle entre les époques est 10 seconds plutôt qu'une seconde comme le fichier RINEX.

Une autre point importante est la cadre de référence qui est indiqué dans la tête, la ligne 25, "EARTH_FIXED".

4 Les Données de satellite GPS

Le dernier ensemble de données nécessaire est la position de l'un des satellites GPS qui est indiqué dans le fichier de RINEX de Sentinel-3A. Pour obtenir ces données, il faut accéder au site web <https://kb.igs.org/hc/en-us/articles/115003935351> et accéder au lien <ftp://gssc.esa.int/gnss/products/>. Dans ce lien, pour trouver les données de la date 31-08-2018, il faut aller au fichier 2016 ce qui correspond la semaine de 26-08-2018 à 01-09-208 en terme de semaine GPS et puis il faut télécharger le fichier *igs20165.sp3*. Le dernier nombre 5 signifie le jour 31-08-2018. (Pour voir la calendrier GPS <https://www.ngs.noaa.gov/CORS/Gpscal.shtml>)

Donc la tête et les 2 époque d'observation du fichier *igs20165.sp3* est dans l'image 3

```

#cP2018 8 31 0 0 0.000000000 96 ORBIT IGS14 HLM IGS
## 2016 432000.0000000 900.0000000 58361 0.0000000000000
+ 32 G01G02G03G04G05G06G07G08G09G10G11G12G13G14G15G16G17
+ G18G19G20G21G22G23G24G25G26G27G28G29G30G31G32 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
++ 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
++ 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0
++ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
++ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
++ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
/* FINAL ORBIT COMBINATION FROM WEIGHTED AVERAGE OF:
/* cod emr esa gzf grq jpl mit ngs sio
/* REFERENCED TO IGS TIME (IGST) AND TO WEIGHTED MEAN POLE:
/* PCV:IGS14_2013 OL/AL:FES2004 NONE Y ORB:CMB CLK:CMB
* 2018 8 31 0 0 0.000000000
PG01 -13446.981410 19058.767932 12249.825989 -82.594319 6 7 8 83
PG02 15089.527570 1672.575664 -21223.053078 12.286017 10 7 9 40
PG03 -20214.097401 13521.868455 -10627.676726 129.253977 10 5 9 96
PG04 -14769.997822 -10087.145837 -19979.968361 999999.999999
PG05 25299.602039 -1601.898989 -8396.302437 -2.708183 10 8 8 110
PG06 8041.072932 18043.901047 -17715.991929 369.669524 8 10 7 114
PG07 -5168.294478 25709.955646 2291.534073 132.428940 8 5 5 118
PG08 -18283.298414 1035.983223 19337.089952 -113.638063 7 7 6 98
PG09 -549.805185 18279.081284 -19288.096030 508.710501 9 6 7 79
PG10 -8564.064431 -13124.064985 21505.436434 184.399663 7 9 8 114
PG11 -11369.983826 15438.416985 17896.000302 -721.559375 8 3 9 83
PG12 23878.980097 -9583.788414 -6397.132871 310.206193 9 8 7 101
PG13 20099.060878 7927.499852 15349.546778 -90.243370 9 7 5 119
PG14 -18792.089633 -18728.426833 -2274.250980 -97.655639 10 8 8 115
PG15 16196.421324 -6149.967771 19997.862399 -347.355447 8 7 7 112
PG16 -24898.339193 -1661.350875 -9844.080553 11.071146 10 7 8 120
PG17 13740.417114 21734.022288 7292.325718 -61.537230 9 9 9 100
PG18 -15399.168940 11007.709590 18089.883066 999999.999999
PG19 18777.419599 18406.902355 -3464.579202 -407.267099 9 9 7 100
PG20 1766.247978 -17031.167385 20161.359450 514.859601 8 9 8 126
PG21 430.155706 -26012.796596 6249.476962 -344.716116 7 8 9 92
PG22 -24201.205302 10578.164563 -2424.834749 -505.175428 8 4 9 95
PG23 -10347.381336 11465.663202 -21304.129709 -211.989779 10 7 4 112
PG24 14613.747581 -15648.205733 15405.006655 -54.877120 7 7 6 121
PG25 12666.221203 -15798.686641 -17244.051885 -639.844437 9 7 4 91
PG26 -18730.709782 -7151.691925 -17495.783033 -35.415767 8 6 8 103
PG27 -21767.595380 -9658.167053 11946.688611 349.308108 2 8 7 94
PG28 4663.371961 14299.083030 22512.483602 741.372835 6 5 6 122
PG29 2405.864136 -17164.374166 -20151.584857 385.726646 9 7 8 90
PG30 2001.702288 23540.438202 11981.252554 30.459797 8 6 5 112
PG31 -9246.695322 -15940.394077 -18934.831457 96.797144 6 8 5 104
PG32 -15427.261711 -20751.629178 6320.580515 -437.645056 7 5 7 88
* 2018 8 31 0 15 0.000000000
PG01 -13307.215812 17486.373191 14533.487366 -82.598285 6 7 8 79
PG02 14705.101716 4193.237043 -21166.439115 12.276156 10 7 8 63
PG03 -21300.036054 13628.922639 -8043.240174 129.258814 9 5 9 85
PG04 -12592.046641 -10787.242847 -21064.088230 999999.999999
PG05 24353.717515 -1015.387837 -10918.311273 -2.708311 9 7 7 112
PG06 7503.348635 19897.894527 -15871.732115 369.665154 7 9 7 110
PG07 -5478.337690 25714.086951 -582.838375 132.423402 8 5 5 113
PG08 -19795.001604 -441.404061 17821.650267 -113.639102 6 7 6 98
PG09 -2327.415186 16865.799311 -20412.942715 508.708885 9 6 6 89
PG10 -6225.167601 -13909.637878 21799.435950 184.397946 7 9 7 110
PG11 -12030.817119 13294.820168 19089.468280 -721.557183 7 3 8 85
PG12 24457.510894 -9535.637339 -3556.376369 310.203470 10 8 7 101
PG13 21047.070908 9286.297976 13180.054026 -90.242463 9 7 5 117
PG14 -18705.855685 -18989.547318 569.368608 -97.655526 10 8 8 108
PG15 17936.261278 -4683.385415 18928.274646 -347.354460 7 7 7 113
PG16 -23728.241578 -2065.284803 -12340.936333 11.069638 9 7 8 124
PG17 12966.972778 21188.836499 9957.516644 -61.533505 8 8 8 112
PG18 -15178.790660 8767.640535 19431.284900 999999.999999
PG19 18758.756416 18693.088247 -573.264027 -407.264061 8 8 7 111
PG20 3902.839460 -17851.278794 19122.144239 514.860124 8 8 8 126
PG21 782.805965 -26432.347782 3500.630852 -344.710893 7 7 9 95
PG22 -24465.768372 10299.589183 358.138118 -505.185341 9 4 8 82
PG23 -12229.215048 9822.277098 -21167.389704 -211.989268 9 7 3 115
PG24 14517.834743 -13734.326206 17227.246120 -54.877170 6 7 5 122
PG25 14463.349327 -16211.174028 -15298.193851 -639.849582 9 6 4 92
PG26 -16806.196968 -7966.224537 -19019.760391 -35.405312 7 6 7 106
PG27 -22570.536557 -10597.352978 9434.921018 349.305027 3 7 7 90
PG28 2290.886896 14847.550928 22528.761503 741.374810 6 4 6 121
PG29 4101.028682 -15530.578586 -21178.365618 385.720493 9 7 7 86
PG30 1417.397308 24665.386345 9534.066506 30.455200 7 6 5 115
PG31 -8592.935295 -17938.738290 -17356.276339 96.794865 6 7 5 101
PG32 -14617.139783 -20355.924870 8980.997765 -437.635709 7 5 7 100

```

Figure 3: Éphémérides de satellite GPS (orbites)

Cela se voit que (l'image 3), contrairement aux 2 ensembles des données précédents, ici l'intervalle entre les époques est beaucoup plus long, 15 minutes.

A fin de trouver le satellite intéressé, il est suffit de voir la première colonne des observation, après le symbole P, le reste est le code de PRN. Les 3 colonnes suivants sont des observations de la position X, Y et Z en kilomètres [37].

5 Lire Les Fichiers

Les 3 différents fichier peuvent être lu par un paquet et une bibliothèque intégrée.

5.1 RINEX et Éphémérides de satellite GPS

Les fichiers RINEX et Éphémérides de satellite GPS sont lus avec l'aide du paquet **georinex**. L'installation du paquet est documentée dans le site github du paquet, <https://github.com/geospace-code/georinex>. Après l'installation, la commande *load* est utilisé pour lire tout le fichier. Cette commande fait retourner un xarray.Dataset. Xarray est un paquet qui fait possible de travailler avec les multi-dimensionnel array avec les étiquettes (<http://xarray.pydata.org/en/stable/index.html>).

La commande *load* prends un argument optionnel, *meas* qui peut choisir les types d'observation et qui peut réduire le temps d'opération. Puis dans le Xarray qui est retourné par la commande *load*, le satellite intéressée peuvent être choisi par la méthode *.sel(sv="")*. Ces étapes peuvent être vu dans le code (Appendix A.1).

La même procédure est effectuée pour la position du satellite GPS, le fichier *igs20165.sp3*.

5.2 Éphémérides Orbitales Précises Fichier

Le fichier "AUX_POEORB" est lu par la bibliothèque ElementTree, *xml.etree.ElementTree*. Avec les commandes *parse* qui retourne un objet ElementTree et la méthode *.getroot* qui retourne un élément racine. La méthode *.iter()* aide de trouver les sub-éléments de cette racine qui correspond les lignes commencent par le mot écrit dans la méthode. Ces opérations peuvent être vu dans le code (Appendix A.1)

5.3 La Trace des Données

La graphe de pseudorange de Sentinel-3A avec les satellites G01 et G02 est donnés respectivement dans les images 4 et 5.

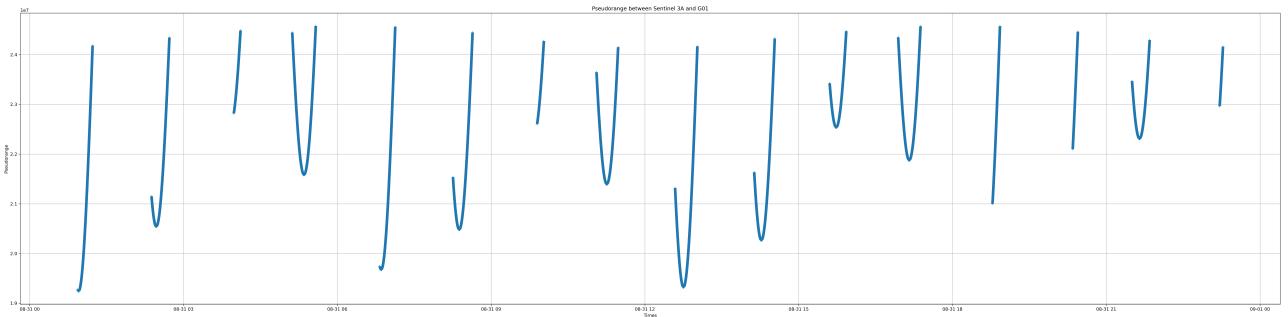


Figure 4: Pseudorange de Sentinel-3A avec le satellite G01.

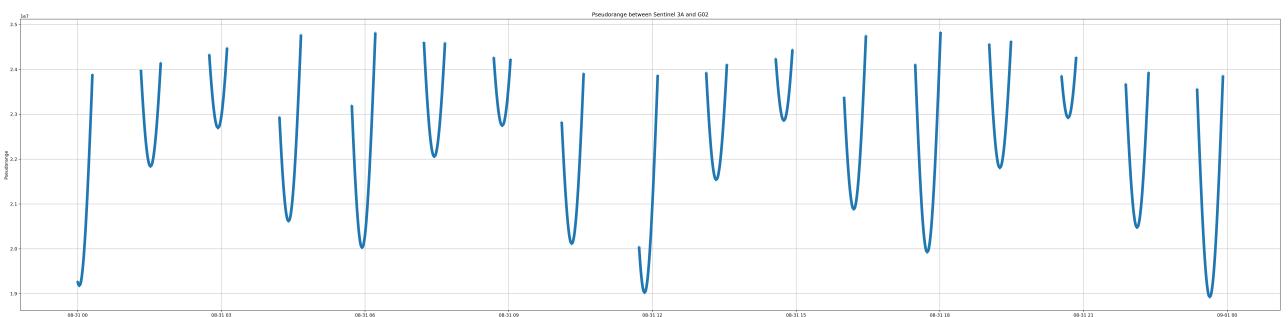


Figure 5: Pseudorange de Sentinel-3A avec le satellite G02.

Après obtenir les données de la position de Sentinel-3A et le satellite GPS, il faut soustraire les deux vecteur de position pour avoir une distance. Il est important d'avoir tous les données en même temps. Il y a des données de la position de Sentinel-3A dans chaque 10 seconds et du satellite GPS dans chaque 15 minutes. Donc les données de la position de Sentinel-3A sont sélectionnés avec un pas de 90 éléments. Après l'opération, quand cette distance est tracée dans la même graphe avec les données pseudorange, l'image 6 est obtenu.

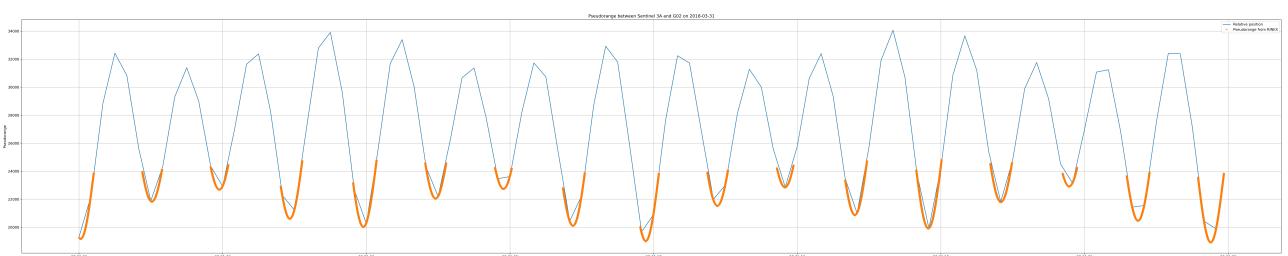
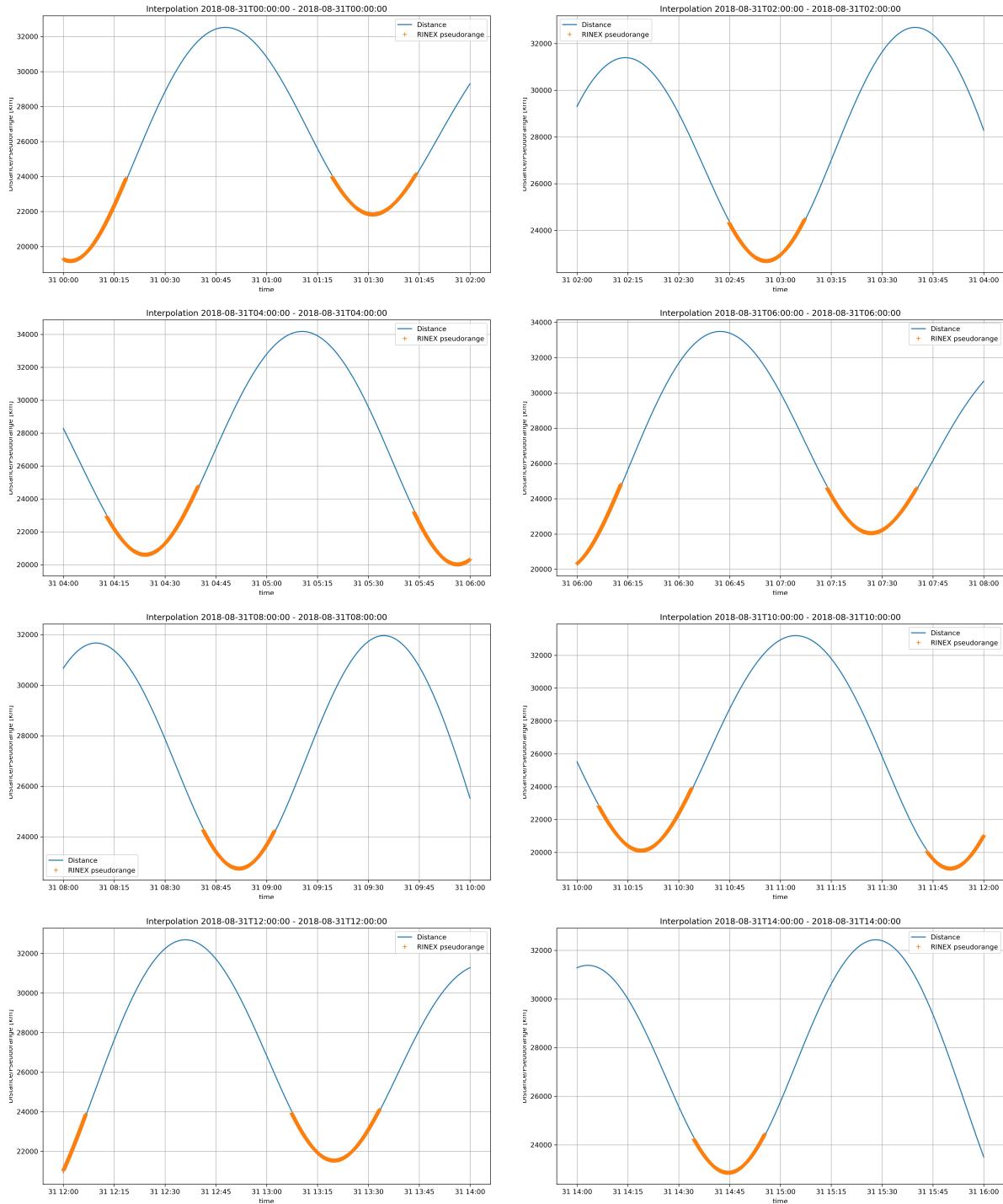


Figure 6: Pseudorange de Sentinel-3A avec le satellite G02 et la différence entre la position de Sentinel-3A et du satellite G02.

6 Interpolation

Dans la deuxième étape, l'interpolation des données de la position de Sentinel-3A et du satellite G02 sont faite pour avoir les données dans chaque seconds. l'interpolation lagrangienne est réalisée par la fonction `lagrange` de la bibliothèque `scipy.interpolate`. Polynômes de Lagrange de degré 8 est utilisés avec 9 données. Les interpolations sont effectuées aux données de distance

calculée en 11 étapes qui peuvent être vu dans les images de 7. Pour chaque ensemble de données de 9 éléments, une seul interpolation a été fait. Cette technique peut être changer pour avoir des résultats plus précises.



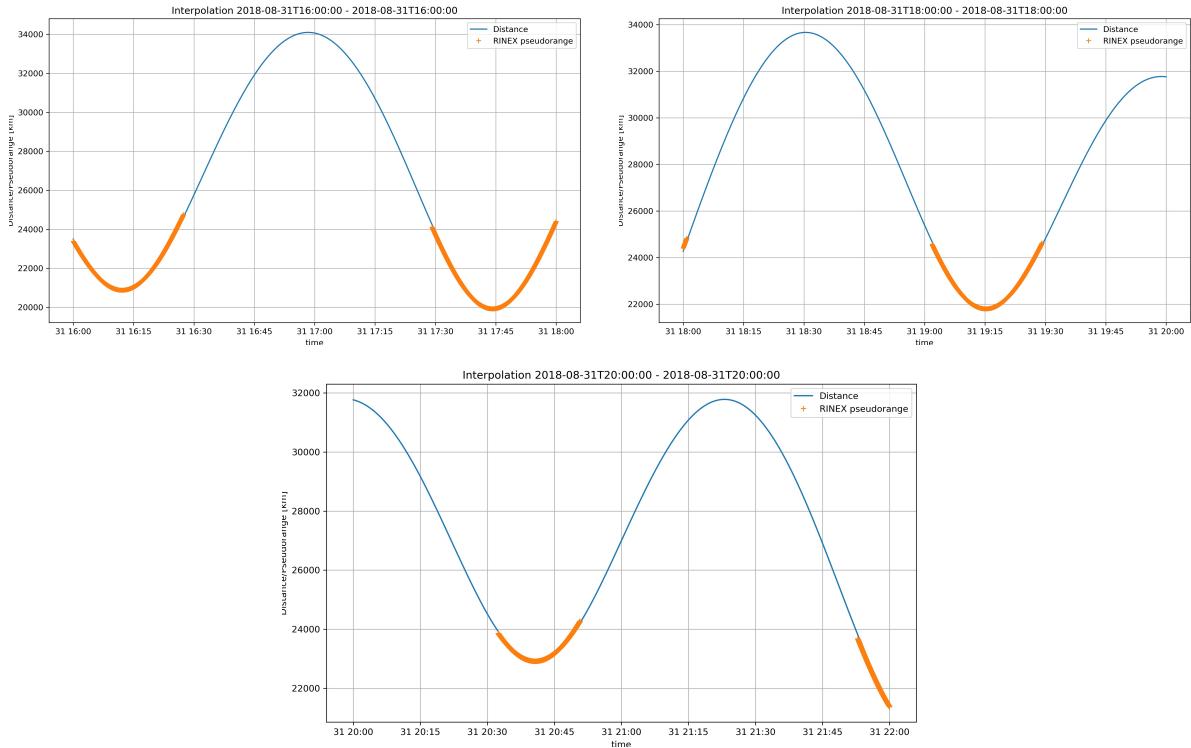


Figure 7: Les interpolations

La graphe finale qui est une graph amélioré de l'image 6 est donnée dans l'image 8 et 10.

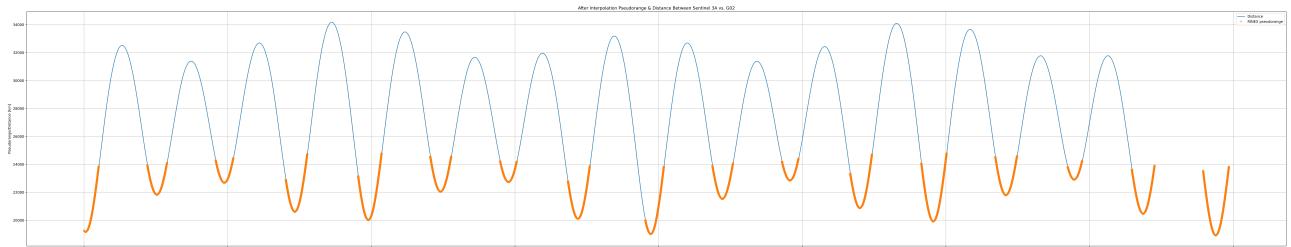


Figure 8: Pseudorange et la distance après l'interpolation.

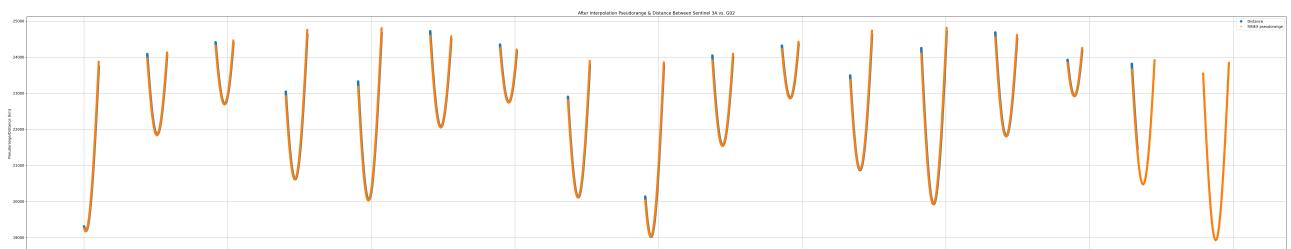


Figure 9: Pseudorange et la distance après l'interpolation sans les points extérieurs.

6.1 Erreur

À partir de résultats des interpolations, l'erreur est calculée en trouvant la valeur absolue de la différence entre les distances et les pseudoranges et elle est tracée, l'image ??

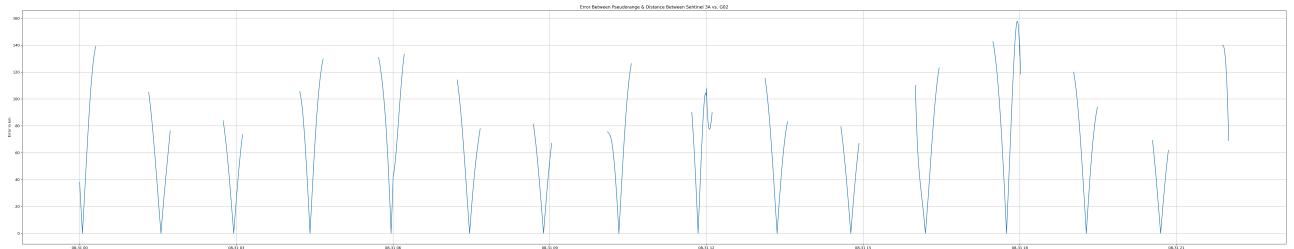


Figure 10: L’erreur entre les pseudoranges et les distances.

7 Les commentaires et Les Travaux Ultérieurs

De l’image 6 à 8, l’amélioration est clairement visible mais absolument insuffisant. Au lieu d’interpoler le vecteur de distance, d’interpoler des données séparément et puis de trouver le vecteur de distance peut donner des mieux résultats. La technique d’interpolation peut être modifiée pour plus de précision. Après avoir utilisé les 9 premiers termes, au lieu d’utiliser les 9 suivants, les 9 termes peuvent être décalés d’un élément vers la droite, et les interpoler à nouveau peut donner des résultats plus précis.

Pour aller un peu plus loin, il faut premièrement faire une correction de l’horloge. D’abord il faut trouver les fichiers des horloges (l’horloge de Sentinel-3A et le satellite GPS) et comprendre de les lire et une correction par rapport à la différence entre les horloges peut être effectué.

References

- [1] "New gnss weather data sets from techdemosat-1 | sstl." <https://www.sstl.co.uk/media-hub/latest-news/2016/new-gnss-weather-data-sets-from-techdemosat-1>. Accessed: 2020-08-30 04:40:48.
- [2] C. Chew, R. Shah, C. Zuffada, G. Hajj, D. Masters, and A. J. Mannucci, "Demonstrating soil moisture remote sensing with observations from the uk techdemosat-1 satellite mission," *Geophysical Research Letters*, vol. 43, no. 7, pp. 3317–3324, 2016.
- [3] "Home | cyclone global navigation satellite system (cygnss)." <https://clasp-research.engin.umich.edu/missions/cygnss/>. Accessed: 2020-08-30 05:14:54.
- [4] Wikipedia contributors, "Cyclone global navigation satellite system — Wikipedia, the free encyclopedia." https://en.wikipedia.org/w/index.php?title=Cyclone_Global_Navigation_Satellite_System&oldid=970571074, 2020. [Online; accessed 31-August-2020].
- [5] Wikipedia contributors, "Cryosat-2 — Wikipedia, the free encyclopedia." <https://en.wikipedia.org/w/index.php?title=CryoSat-2&oldid=970432135>, 2020. [Online; accessed 31-August-2020].
- [6] "Jason-3 - satellite missions - eoportal directory." <https://directory.eoportal.org/web/eoportal/satellite-missions/j/jason-3>. Accessed: 2020-08-31 02:14:50.
- [7] Wikipedia contributors, "Metop — Wikipedia, the free encyclopedia." <https://en.wikipedia.org/w/index.php?title=MetOp&oldid=963029493>, 2020. [Online; accessed 31-August-2020].
- [8] "Metop — eumetsat." <https://www.eumetsat.int/website/home/Satellites/CurrentSatellites/Metop/index.html>. Accessed: 2020-08-31 02:11:35.
- [9] Wikipedia contributors, "Grace and grace-fo — Wikipedia, the free encyclopedia." https://en.wikipedia.org/w/index.php?title=GRACE_and_GRACE-FO&oldid=971318054, 2020. [Online; accessed 31-August-2020].
- [10] "Grace-fo mission overview | nasa." www.nasa.gov/missions/grace-fo/overview. Accessed: 2020-08-31 02:35:41.
- [11] "Smos - eoportal directory - satellite missions." <https://directory.eoportal.org/web/eoportal/satellite-missions/s/smos>. Accessed: 2020-08-31 02:38:17.
- [12] Wikipedia contributors, "Soil moisture and ocean salinity — Wikipedia, the free encyclopedia." https://en.wikipedia.org/w/index.php?title=Soil_Moisture_and_Ocean_Salinity&oldid=956661552, 2020. [Online; accessed 31-August-2020].

- [13] Wikipedia contributors, “Orbcomm — Wikipedia, the free encyclopedia.” <https://en.wikipedia.org/w/index.php?title=Orbcomm&oldid=975633769>, 2020. [Online; accessed 31-August-2020].
- [14] “Esa - cubesat finds its way in space with galileo receiver.” https://www.esa.int/Applications/Navigation/CubeSat_finds_its_way_in_space_with_Galileo_receiver. Accessed: 2020-08-31 02:43:52.
- [15] E. Kulu, “Astrocast @ nanosats database.” <https://www.nanosats.eu/sat/astrocast>. Accessed: 2020-08-31 02:44:06.
- [16] “Magnetospheric multiscale spacecraft.” <https://mms.gsfc.nasa.gov/spacecraft.html>. Accessed: 2020-08-31 02:45:31.
- [17] Wikipedia contributors, “Magnetospheric multiscale mission — Wikipedia, the free encyclopedia.” https://en.wikipedia.org/w/index.php?title=Magnetospheric_Multiscale_Mission&oldid=955366448, 2020. [Online; accessed 31-August-2020].
- [18] S. Winkler, G. Ramsey, C. Frey, J. Chapel, D. Chu, D. Freesland, A. Krimchansky, and M. Concha, “Gps receiver on-orbit performance for the goes-r spacecraft,” 2017.
- [19] J. J. Parker, “New frontiers in space use of gnss: Moon and beyond,” 2019.
- [20] A. Hauschild, M. Markgraf, and O. Montenbruck, “The navigation and occultation experiment-gps receiver performance on board a leo satellite,” *Inside Gnss*, pp. 48–57, 2014.
- [21] “Gnss instrument – sentinel-3 altimetry technical guide – sentinel online.” <https://sentinel.esa.int/web/sentinel/technical-guides/sentinel-3-altimetry/instrument/gnss>. Accessed: 2020-08-31 02:53:37.
- [22] Wikipedia contributors, “Sentinel-3a — Wikipedia, the free encyclopedia.” <https://en.wikipedia.org/w/index.php?title=Sentinel-3A&oldid=958592505>, 2020. [Online; accessed 31-August-2020].
- [23] E. Friis-Christensen, H. Lühr, and G. Hulot, “Swarm: A constellation to study the earth’s magnetic field,” *Earth, planets and space*, vol. 58, no. 4, pp. 351–358, 2006.
- [24] “Swarm - eoportal direcory - satellite missions.” <https://directory.eoportal.org/web/eoportal/satellite-missions/s/swarm>. Accessed: 2020-08-31 02:55:06.
- [25] “Icesat-2 - eoportal direcory - satellite missions.” <https://directory.eoportal.org/web/eoportal/satellite-missions/i/icesat-2#foot13%29>. Accessed: 2020-08-31 03:21:10.
- [26] Wikipédia, “Pléiades (satellite) — wikipédia, l’encyclopédie libre.” [http://fr.wikipedia.org/w/index.php?title=Pl%C3%A9iades_\(satellite\)&oldid=168819175](http://fr.wikipedia.org/w/index.php?title=Pl%C3%A9iades_(satellite)&oldid=168819175), 2020. [En ligne; Page disponible le 26-mars-2020].

- [27] Wikipédia, “Radarsat constellation — wikipédia, l’encyclopédie libre.” http://fr.wikipedia.org/w/index.php?title=RADARSAT_Constellation&oldid=172368531, 2020. [En ligne; Page disponible le 26-juin-2020].
- [28] “Radarsat constellation - eoportal directory - satellite missions.” <https://directory.eoportal.org/web/eoportal/satellite-missions/r/rsm>. Accessed: 2020-08-31 03:06:52.
- [29] “Sac-d - eoportal directory - satellite missions.” <https://eoportal.org/web/eoportal/satellite-missions/s/sac-d>. Accessed: 2020-08-31 03:07:46.
- [30] Wikipédia, “Sac-d — wikipédia, l’encyclopédie libre.” <http://fr.wikipedia.org/w/index.php?title=SAC-D&oldid=169696271>, 2020. [En ligne; Page disponible le 18-avril-2020].
- [31] “Megha-tropiques - eoportal directory - satellite missions.” <https://directory.eoportal.org/web/eoportal/satellite-missions/m/megha-tropiques#foot20%29>. Accessed: 2020-08-31 03:13:48.
- [32] “Oceansat-2 - eoportal directory - satellite missions.” <https://earth.esa.int/web/eoportal/satellite-missions/o/oceansat-2>. Accessed: 2020-08-31 03:14:47.
- [33] Wikipedia contributors, “Oceansat-2 — Wikipedia, the free encyclopedia.” <https://en.wikipedia.org/w/index.php?title=Oceansat-2&oldid=970504112>, 2020. [Online; accessed 31-August-2020].
- [34] “Cosmo-skymed - eoportal directory - satellite missions.” <https://directory.eoportal.org/web/eoportal/satellite-missions/c-missions/cosmo-skymed>. Accessed: 2020-08-31 03:15:41.
- [35] Wikipedia contributors, “Cosmo-skymed — Wikipedia, the free encyclopedia.” <https://en.wikipedia.org/w/index.php?title=COSMO-SkyMed&oldid=975626187>, 2020. [Online; accessed 31-August-2020].
- [36] R.-S. IGS, “Rinex-the receiver independent exchange format (version 3.03),” URL: <ftp://igs.org/pub/data/format/rinex303.pdf>, 2015.
- [37] S. Hilla, “The extended standard product 3 orbit format (sp3-c)(17 august 2010),”

A Appendix

A.1 Python Code

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""

Created on Mon Aug 17 11:30:41 2020

@author: tunceturk
"""

import numpy as np
import matplotlib.pyplot as plt
import georinex as gr
import xml.etree.ElementTree as ET
from scipy.interpolate import lagrange

#####
# SENTINEL - 3A (PSEUDORANGE FROM RINEX FILE) #####
# DATA IS BROADCASTED IN EVERY SECOND #####
dat = gr.load('
    S3A_OPER_AUX_GNSSRD POD __20180907T024226_V20180830T235942_20180831T235941.DBL
    ', meas=['C1C'])
pseudotime = dat["C1C"].sel(sv='G02').time
pseudorange = dat["C1C"].sel(sv='G02')*0.001

#####
# GPS #####
# DATA IS BROADCASTED IN EVERY 15 MINS #####
G02x=np.array([])
G02y=np.array([])
G02z=np.array([])
G02t=np.array([], dtype=np.datetime64)

dat2 = gr.load('igs20165.sp3')
G02x = dat2.sel(sv='G02', ECEF='x').dropna(dim='time', how='all').position.variable
#dropna drops all values when time variable is nan.
G02y = dat2.sel(sv='G02', ECEF='y').dropna(dim='time', how='all').position.variable
G02z = dat2.sel(sv='G02', ECEF='z').dropna(dim='time', how='all').position.variable
G02t = dat2.sel(sv='G02').dropna(dim='time', how='all').time.variable

G02r = (G02x**2+G02y**2+G02z**2)**0.5
```

```

#####
    SENTINEL - 3A (POSITION) #####
    DATA IS BROADCASTED IN EVERY 10 SECONDS #####
SxTest=np.array([])
SyTest=np.array([])
SzTest=np.array([])
StTest=np.array([] , dtype=np.datetime64)

tree1 = ET.parse('
    S3A_OPER_AUX_POEORB POD __20180925T071034_V20180830T215942_20180831T235942_DGNS
    .EOF')
root1 = tree1.getroot()

for i in root1.iter('TAI'):
    #time variable
    StTest = np.append(StTest , np.datetime64(i.text.split("=".split(1)[1])))

for i in root1.iter('X'):
    SxTest = np.append(SxTest , float(i.text))

for i in root1.iter('Y'):
    SyTest = np.append(SyTest , float(i.text))

for i in root1.iter('Z'):
    SzTest = np.append(SzTest , float(i.text))

mark = np.where(StTest == np.datetime64('2018-08-30T23:59:59.000000'))[0][0]

#IN ORDER TO GET THE EXACTLY SAME TIME WITH GPS DATA WHICH IS BROADCASTED IN
EVERY 15 MINS,
#TERMS ARE TAKEN WITH A STEP OF 90 ELEMENTS WHICH MAKES 15 MINS INTERVAL
Sx=SxTest[mark:-3:90]*0.001 #*0.001 To convert it to the km
Sy=SyTest[mark:-3:90]*0.001
Sz=SzTest[mark:-3:90]*0.001
St=StTest[mark:-3:90]

Sr = (Sx**2+Sy**2+Sz**2)**0.5

#####
    DIFFERENCE BTWN GPS POSITION VECTOR & SENTINEL POSITION VECTOR #####
relative = ((Sx-G02x)**2+(Sy-G02y)**2+(Sz-G02z)**2)**0.5

#####
    PLOTING #####
plt.figure(figsize=(50,10))
plt.plot(St,relative)
plt.plot(pseudotime,pseudorange,'+')
plt.grid(True)
plt.title("Pseudorange between Sentinel 3A and G02 on 2018-03-31")

```

```

plt.xlabel("Time")
plt.ylabel("Pseudorange")
plt.tight_layout()
plt.legend(["Relative_position", "Pseudorange_from_RINEX"])
plt.savefig('yepis.jpg', dpi=300)
# plt.figure(figsize=(50,10))
# plt.plot(St[:9], relative[:9], 'o')
# plt.plot(pseudotime[:2600], pseudorange[:2600], '+')

#####
INTERPOLATION #####
def interpolate(st, S, t):
    # 8th Lagrange Interpolation
    poly = lagrange(st, S)
    c = poly.coef
    s = c[0]*t**8+c[1]*t**7+c[2]*t**6+c[3]*t**5+c[4]*t**4+c[5]*t**3+c[6]*t**2+c[7]*t+c[8]
    return s

#9 terms 8 intervals for 15mins it makes 2hours interval which is 7200 seconds
term = 9
second = 7200
st = np.arange(0, term)
t = np.linspace(0, term-1, second)
ref = np.datetime64('2018-08-31T00:00:00')

#Because the GPS values stops at 2018-08-31T23:45:00 , interpolation of last 2
# hours interval is not done.
#Thus there are 79200 terms in final array instead of 86400secs
R = np.zeros(79200)
ERR = np.zeros(79200)

date_start = np.datetime64('2018-08-31T00:00:00')
while date_start < np.datetime64('2018-08-31T22:00:00'):
    # Using lagrange polynomial 8th order that's why 9 terms are needed.
    date_final = date_start + np.timedelta64(2, 'h')
    date = np.arange(date_start, date_final, dtype='datetime64[s]')

#this part is for finding the starting index of the arrays.
inter = date_start - ref
inter = inter / np.timedelta64(1, 's')
inter = int(inter)
first = int(inter / 60 / 15)

#Interpolation of sentinel data
sx = interpolate(st, Sx[first:first+term], t)
sy = interpolate(st, Sy[first:first+term], t)

```

```

sz = interpolate(st, Sz[first:first+term], t)

sr = (sx**2+sy**2+sz**2)**0.5

#Interpolation of GPS data
g02x = interpolate(st, G02x[first:first+term], t)
g02y = interpolate(st, G02y[first:first+term], t)
g02z = interpolate(st, G02z[first:first+term], t)

g02r = (g02x**2+g02y**2+g02z**2)**0.5

#Interpolated vector
r = ((sx-g02x)**2+(sy-g02y)**2+(sz-g02z)**2)**0.5

#####
##### PLOTING #####
plt.figure(figsize=(10,6))
plt.title("Interpolation " + str(date_start) + " - " + str(date_start))
plt.plot(date, r)
plt.plot(pseudotime[inter:inter+second], pseudorange[inter:inter+second], '+')
plt.legend(['Distance', 'RINEX_pseudorange'])
plt.tight_layout()
plt.grid(True)
plt.xlabel("time")
plt.ylabel("Distance/Pseudorange [km]")
plt.savefig(str(date_start)+'.jpg', dpi=300)

date_start = date_start + np.timedelta64(2, 'h')
R[inter:inter+second] = r
# err = abs(r-pseudorange[inter:inter+7200])
# plt.figure()
# plt.plot(date, err)
# ERR[inter:inter+7200] = err

#####
##### PLOTING THE FINAL ARRAY #####
plt.figure(figsize=(50,10))
plt.title("After_Interpolation_Pseudorange & Distance_Between_Sentinel_3A_vs_G02")
plt.plot(pseudotime[:79200], R)
plt.plot(pseudotime, pseudorange, '+')
plt.grid(True)
plt.xlabel("Time")
plt.ylabel("Pseudorange/Distance [km]")
plt.tight_layout()
plt.legend(['Distance', 'RINEX_pseudorange'])
plt.savefig('After_Interpolation1.jpg', dpi=300)

```

```

m = pseudorange[:79200].copy() #just for not changing pseudorange array.
m[m.notnull()]=1
m[m.isnull()]=0
RR = R*m
RR[RR==0] = np.NaN
#R is interpolated distance btwn sentinel & GPS. The difference btwn R & RR
##RR is the same as R but the corresponding values of pseudorange array which are
#nan are also set nan in RR
Err = abs(RR - pseudorange[:79200]) #Calculating the error

```

```

#####
# PLOTING THE FINAL ARRAY #####
plt.figure(figsize=(50,10))
plt.title("After_Interpolation_Pseudorange_&_Distance_Between_Sentinel_3A_vs._G02")
plt.plot(pseudotime[:79200],RR, 'o')
plt.plot(pseudotime,pseudorange, '+')
plt.grid(True)
plt.xlabel("Time")
plt.ylabel("Pseudorange/Distance_[km]")
plt.tight_layout()
plt.legend(['Distance', 'RINEX_pseudorange'])
plt.savefig('After_Interpolation2.jpg', dpi=300)

```

```

#####
# PLOTING THE ERROR #####
plt.figure(figsize=(50,10))
plt.plot(pseudotime[:79200], Err)
plt.title("Error_Between_Pseudorange_&_Distance_Between_Sentinel_3A_vs._G02")
plt.grid(True)
plt.xlabel("Time")
plt.ylabel("Error_in_km")
plt.tight_layout()
plt.savefig('Error.jpg', dpi=300)

```