

SUTTER Guillaume

THAALBI Sinda

NOURRY Justine

KREMP Mathilde

Projet

Sujet : prétraitement des observations gravimétriques

encadrant : Jacques Beilin

du 11 au 29 juillet 2016



Sommaire :

Objectif

Principe de la gravimétrie

Organisation type d'un cheminement

Contraintes liée à l'appareil

Mesures effectuées

Améliorations apportées

Difficultés rencontrées

Notice de pycgxtool

Objectif :

Nous allons organiser des observations gravimétriques. Ces observations vont permettre de nous aider à comprendre les logiciels de prétraitement CGxtool et pycgxtool et ainsi nous aider à les améliorer et de les rendre utilisables pour l'IGN et la communauté gravimétrique. Nous prenons comme référence le logiciel Cgxtool pour apporter des modifications à pycgxtool.

Principes de la gravimétrie :

La gravimétrie est une branche de la géodésie qui mesure l'accélération de la pesanteur g . Il permet d'obtenir une précision de l'ordre du μGal . Cette technique est très utilisée pour la mesure du géoïde mais également pour la prospection pétrolière et de ressources énergétiques en général.

La valeur de g est liée d'une part à la densité de la roche constituant le sous sol, et d'autre part à des phénomènes de marée terrestre, de surcharge océanique, et bien sûr à la position sur l'ellipsoïde.

Nous utilisons pendant ces trois semaines un gravimètre relatif Scintrex CG5 prêté par l'Institut des Sciences de l'Univers. Ce type d'appareil mesure des variations locales de g , grâce à un système de ressort et de masse calibrée. Cette masse est soumise à un champ électrique induit par deux plaques sous tension. L'appareil mesure la tension nécessaire pour stabiliser la masse. L'ensemble du dispositif se trouve dans une enceinte thermostatée. La mesure s'effectue sous forme de cheminement (aller-retour), entre deux points, avec deux points intermédiaires. Chaque point est stationné deux fois, sauf le dernier point de l'aller qui n'est occupé qu'une seule fois. Sur chaque point, 3 séries d'acquisition de 85 secondes sont effectuées. La fréquence d'enregistrement est de 6Hz, ce qui permet de filtrer un maximum de bruit lié à l'activité aux alentours. L'altitude de l'appareil est déterminée grâce aux repères de nivellement situés à proximité du lieu de mesure.

Il existe un autre type de gravimètre, plus coûteux et plus contraignant : le gravimètre absolu. Ce type d'appareil mesure la vitesse de chute d'une masse calibrée grâce à des impulsions laser et des miroirs.

Organisation type d'un cheminement :

La veille, une reconnaissance est effectuée, afin de situer exactement les repères de nivellement utilisés. Ceci permet de gagner du temps lors du cheminement. Le jour des mesures, nous commençons le matin, pour être sûrs de finir le cheminement. Arrivés sur le point de mesure, nous bullons et nous laissons reposer l'appareil une quinzaine de minutes avant de l'utiliser. Une fois l'appareil stabilisé nous lançons les trois mesures de 85 secondes, en vérifiant entre chaque mesure le bullage du gravimètre. Pendant la mesure il est important de ne pas bouger de façon brusque et de ne pas marcher autour de l'appareil, cela pouvant perturber la mesure, l'appareil étant très sensible. Le même protocole est répété pour chaque point. Pour chaque point, la durée moyenne d'occupation est d'environ 3/4 d'heure.

Une fois le cheminement terminé, nous déchargeons le gravimètre et nous traitons les données enregistrées à l'aide des deux logiciels.

Contraintes liées à l'appareil :

Le gravimètre étant très sensible aux variations de température et aux mouvements, il faut le protéger au maximum pendant le transport et la mesure. L'appareil est transporté dans une caisse sanglée dans le coffre de la voiture. Pendant la mesure, nous protégeons l'appareil et surtout son écran au moyen d'un parasol. Nous considérons que plus l'appareil a subi de mouvements pendant son transport, plus il faudra le laisser reposer avant de l'utiliser. Cela augmente la durée du cheminement, qui dans le cas idéal est le plus court possible.

Dans l'enceinte thermostatée, un vide est maintenu continuellement pour mesurer la variation de g dans le vide et s'affranchir des gradients de g dans l'air. Il faut donc que l'appareil soit continuellement alimenté, soit sous batteries sur le terrain, soit sur secteur quand il est au repos.

Mesures effectuées :

Pour traiter les mesures, nous utilisons deux logiciels : Cgxtool, développé en C++ et pycgxtool, développé en python par Manon Girard (PPMD, ing13). Ce dernier n'est pas encore

utilisé par les géodésiens. Nous allons donc vérifier en premier lieu le bon fonctionnement de ce logiciel en effectuant nos propres mesures.

Un premier cheminement a été réalisé au centre de Forcalquier, pour se familiariser avec l'appareil et avec les deux logiciels de prétraitement. Ce cheminement nous a appris comment réaliser les mesures dans de bonnes conditions.

Un deuxième cheminement entre Forcalquier et la montagne de Lure a été effectué. Le parcours est le suivant :

Forcalquier(centre ENSG) ↔ Liman(cimetière) ↔ Saint-Etienne les Orgues (chapelle) ↔ Lure (ancienne colonie)

Nous sommes allés faire une mesure en altitude pour mettre en évidence le gradient de g .

Le traitement de ces données avec cgxtool a donné de très bons résultats, avec des erreurs faibles. Nous n'avons pas pu traiter les données avec pycgxtool aussitôt, car il y avait un problème de reconnaissance de l'appareil (problème résolu dans le fichier init). Une fois ce problème résolu, l'allure des courbes en sortie de pycgxtool est très similaire à celles de cgxtool. A ce stade nous améliorons la lisibilité et l'aspect esthétique des fichiers pdf de sortie pour se rapprocher de l'aspect de cgxtool, et les deux logiciels renvoient des pdf très similaires pour le même jeu de données. Des modifications d'échelles de temps ont été faites, entre une durée en absolu et un Δt .

Ce dernier cheminement n'étant que du relatif, nous avons trouvé intéressant d'avoir des points connus en absolu. Nous avons donc prévu un cheminement entre deux points connus en absolu, Chateau-Arnoux et Banon. Nous avons effectué le trajet suivant :

Chateaux-Arnoux ↔ Mallefougasse ↔ Saint-Etienne les Orgues ↔ Banon (réalisé en 5 heures environ). Les deux logiciels renvoient des erreurs faibles et des droites similaires. Lors de l'aller, nous avons eu un problème de mesure à Mallefougasse. En effet l'appareil, après une mesure correcte, a commencé à débuller de façon importante et le TILTX et le TILTY partaient vers des très grandes valeurs (-100). Pour expliquer cette erreur, nous avons émis l'hypothèse d'un séisme qui aurait pu perturber l'appareil. Après des vérifications sur le site USGS earthquake et les enregistrements des stations sismiques de Clermont-Ferrand, il est apparu impossible que la cause soit un séisme. Cela doit donc être dû à un problème du compensateur, d'où le débullage important.

Tous les cheminements précédents ont été fait en voiture. Nous avons voulu effectuer un cheminement en micro-gravimétrie, sur une zone pseudo plate (très faible dénivelée), afin d'avoir des très faibles variations de g , ce qui permet de vérifier une fois de plus les traitements des

logiciels. Nous avons choisi le chemin des Biabaux sur la commune de Dauphin. A pied le temps de repos de l'appareil est plutôt de l'ordre de 5 minutes. Ce cheminement à pied a mis en évidence une différence notable d'allure de courbe entre les deux logiciels. Sur la première page pdf de sortie, ce sont les données brutes qui sont représentées. Nous avons constaté que dans le pycgxtools, les données brutes sont lues dans le fichier et immédiatement corrigées et ce sont les données corrigées qui sont représentées. Nous avons rajouté une liste de stockage des valeurs brutes et le problème a été résolu. Nous n'avions pas vu cette différence plus tôt car sur nos précédents cheminements les variations de g sont assez importantes, et la différence passait inaperçue.

Améliorations de pycgxtool

Les premières améliorations apportées sont visuelles. Nous avons complété les axes et les échelles, ajouté des étiquettes de nom de points, relié les points etc.

Nous avons ajouté le diagramme de répétabilité en fonction de la variation du temps Δt (en minutes). Nous avons constaté que les histogrammes des erreurs en μGal étaient différents, les barres sont décalées d'une unité à chaque fois. Celui implémenté dans le pycgxtool semble correct, il doit s'agir d'un problème d'arrondi dans le cgxtool.

En cherchant à améliorer le deuxième pdf de sortie, nous constatons que les colonnes des fichiers de site, r et c ne correspondent pas toutes. Nous cherchions à obtenir une droite de régression linéaire similaire dans les deux logiciels, et ne trouvant pas le problème nous sommes allés chercher dans le fichier de sortie des données traitées, et les colonnes censées être dans le même ordre de grandeur ou égales ne le sont pas du tout. Dans le fichier c, le jour de l'année a repris sa bonne valeur, il était avant initialisé à 0. On le recalcule grâce à `doy` (day of year) du module `gpstime`. Il faudra donc inclure ce module dans le package mis à disposition pour les utilisateurs.

Toujours pour cette deuxième page, en regardant de plus près la régression linéaire et en affichant au moyen d'un print les valeurs au fur et à mesure, nous avons constaté que les valeurs étaient triées par point et non pas par ordre chronologique. De plus, nous nous sommes rendu compte que les valeurs brutes dans pycgxtool n'étaient en réalité pas brutes car déjà corrigées de la marée. Les valeurs étaient finalement corrigées deux fois. Nous avons enlevé la marée dans les

fichiers de coordonnées et nous avons obtenu d'une part des valeurs correctement corrigées et d'autre part des données réellement brutes.

La surcharge océanique a été rajoutée au code python. Le code n'ayant jamais été implémenté en C++, nous avons repris un code existant de calcul de surcharge et nous l'avons adapté au code de pycgxtool, avec l'aide de Jacques Beilin. Le code existant prenait en entrée deux listes contenant respectivement l'amplitude et la phase pour chaque onde. Nous avons modifié cette structure afin que le programme lise le fichier BLQ et récupère les données intéressantes (phases et amplitudes). Pour utiliser cette correction il faut au préalable télécharger pour chaque station un fichier BLQ qui contient les harmoniques des ondes de marée en fonction de l'endroit (par exemple la correction à Brest va être plus importante que la correction à Forcalquier). Sur le site <http://holt.oso.chalmers.se/loading/>, choisir la gravité en mGal, pas de plot et le format BLQ. Il faut ensuite entrer les coordonnées des stations ainsi qu'une adresse mail valide. Le résultat sera envoyé sous quelques minutes, au format BLQ. Le code de correction de la surcharge va prendre en entrée un fichier BLQ qui contient des données sur les ondes de marée pour chaque point, chaque point sera corrigé en particulier avec sa surcharge.

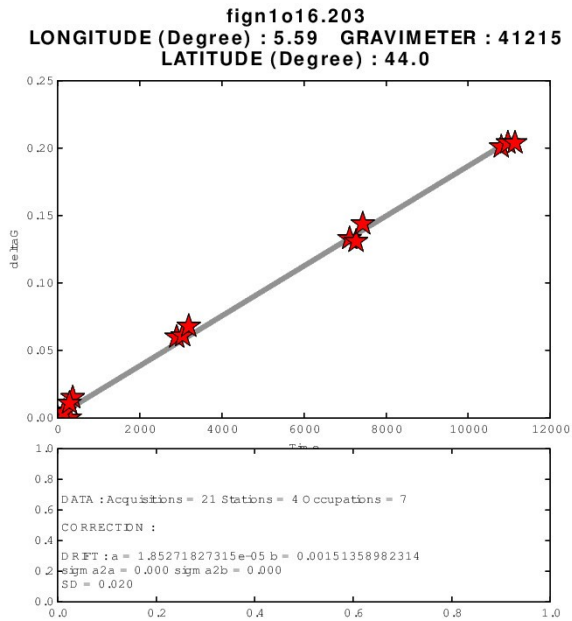
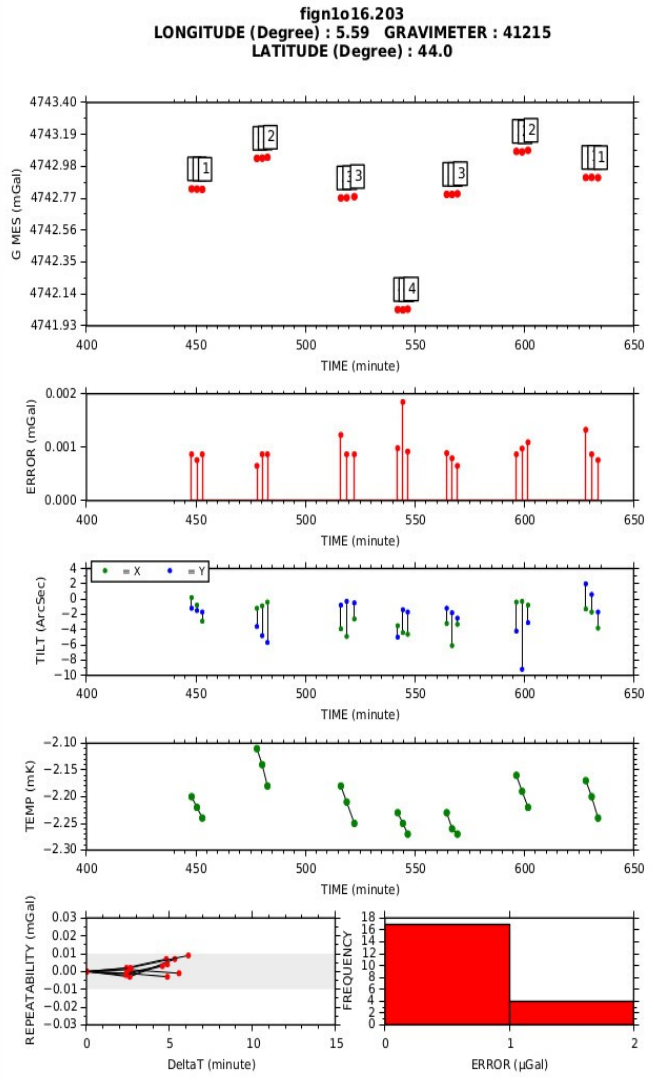
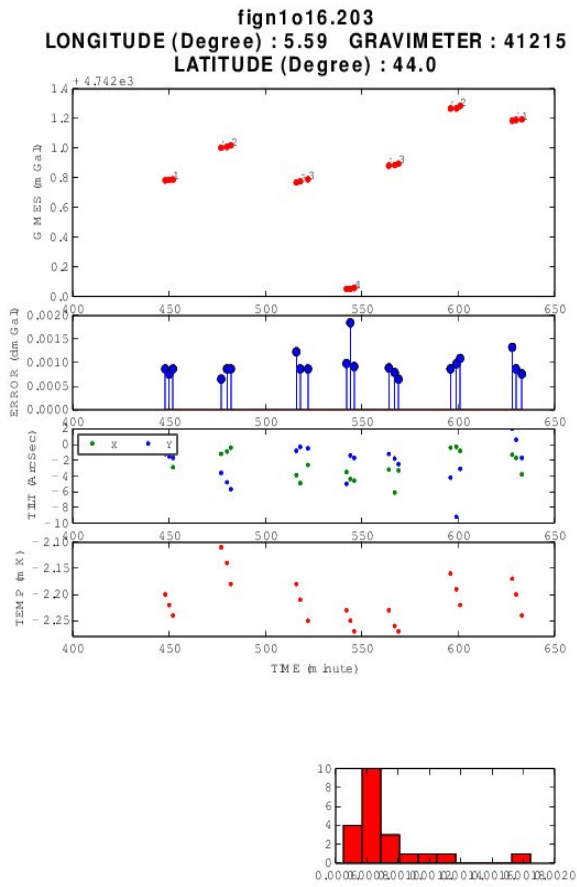
L'interface a été agrémentée de boutons à cocher pour choisir les corrections que l'on souhaite apporter aux mesures. Un bouton de chargement du BLQ a également été ajouté.

Pour vérifier la bonne implémentation de pycgxtool, il a fallu comparer les fichiers de sortie r et c des deux logiciels, en prenant comme référence cgxtool. Il est apparu que les fichiers r de pycgxtool et de cgxtool ne renvoyaient pas les mêmes valeurs d'erreur ($2\mu\text{Gal}$ contre $6\mu\text{Gal}$), il a donc fallu trouver dans les deux codes la façon dont est calculée et écrite dans le fichier r. Il en est de même pour le fichier c.

Les améliorations apportées ont été sauvegardées grâce à git et à son interface giggle. A chaque modification importante une version a été sauvegardée. Nous conservons ainsi tous les changements apportés.

Nous avons constaté que les erreurs sur les coefficients a et b de la régression linéaire étaient différentes sur pycgxtool et sur CGXtool ce qui nous a amené à remplacer la régression linéaire par un ajustement par moindres carrés. Cela permettra ultérieurement à l'utilisateur de tester un modèle non linéaire.

Améliorations visuelles :



Difficultés rencontrées :

Tout d'abord, nous avons été confrontés à du matériel coûteux et fragile, totalement nouveau pour nous. Il a également fallu s'approprier un code qui n'est pas le notre, ainsi qu'un langage (C++) que nous ne maîtrisons pas. Dans un second temps, le module matplotlib utilisé pour le rendu graphique n'est peut être pas le plus approprié compte tenu de toutes ses subtilités, mais nous avons fait le choix de conserver matplotlib et nous avons complété ce rendu en utilisant ce module.

Pour vérifier la bonne implémentation de pycgxtool, il a fallu comparer les fichiers de sortie r et c des deux logiciels, en prenant comme référence cgxtool. Il est apparu que les fichiers r de pycgxtool et de cgxtool ne renvoyaient pas les mêmes valeurs d'erreur, il a donc fallu trouver dans les deux codes la façon dont est calculée et écrite dans le fichier r.

Nous avons pensé à livrer en plus du code python une version executable sur windows ou d'autres OS. Pour cela, nous avons installé le module pyinstaller, qui rassemble le `__main__.py` et tous les `.py` dont le code a besoin pour fonctionner. Mais le code a également besoin du fichier de paramètres du gravimètre `CG3TOOL.init`, qui n'est pas pris en compte par pyinstaller. Nous obtenons un exécutable dont l'interface s'affiche mais les corrections de gravimétrie ne fonctionnent pas. L'idée d'un exécutable a donc été abandonnée.

Notice de pycgxtool :

Cette notice s'adresse à tout utilisateur de pycgxtool. Nous essaierons d'être le plus précis et le plus exhaustif possible afin de faciliter l'utilisation de ce logiciel aux géodésiens. Les calculs ne seront pas détaillés, l'interface sera précisée ainsi que les différents répertoires. Cette notice se veut rédigée en anglais, afin de faciliter l'accessibilité et de rester dans l'optique de pycgxtool, rédigé en anglais.

1.introduction

Pycgxtool was developed by Manon Girard (ENSG) and was completed by students (Guillaume Sutter, Justine Nourry, Sinda Thaalbi and Mathilde Kremp) and their teacher (Jacques Beilin) from this school. The software is based on cgxtool, which was not user-friendly because the set up of GMT was difficult.

The software was developed in Python 3, please make sure you work on this version. In the package you will find the module `gpstime`. Please install this module, or the software won't work and will report an error. To install an interpreter, you can download anaconda for your OS with python 3.5 or higher, launch it and choose spyder.

While using `pycgxtool` you will need several files (observations, BLQ, sites, etc). All these files need to remain in the same directory, or you will have to search in the code where to change the path of the files.

2.First steps on `pycgxtool`

When you download the data from the gravimeter to the computer, you will have to name the file : `f**n#oyy.jjj` with :

- f for Field Mode
- ** : two typefaces you're free to choose, for example ig for IGN (France)
- n : number of the progress, for example 1, 2 or 3
- # : number of the gravimeter, you can find it in the init file of `pycgxtool` (typically 0 to 9)
- o : for observation
- yy : two last numbers of the year (16 for 2016)
- jjj : number of the day in the year (example 058 if it's the 58th day of year).

It's important to name the file following this method, otherwise the software won't recognize the file.

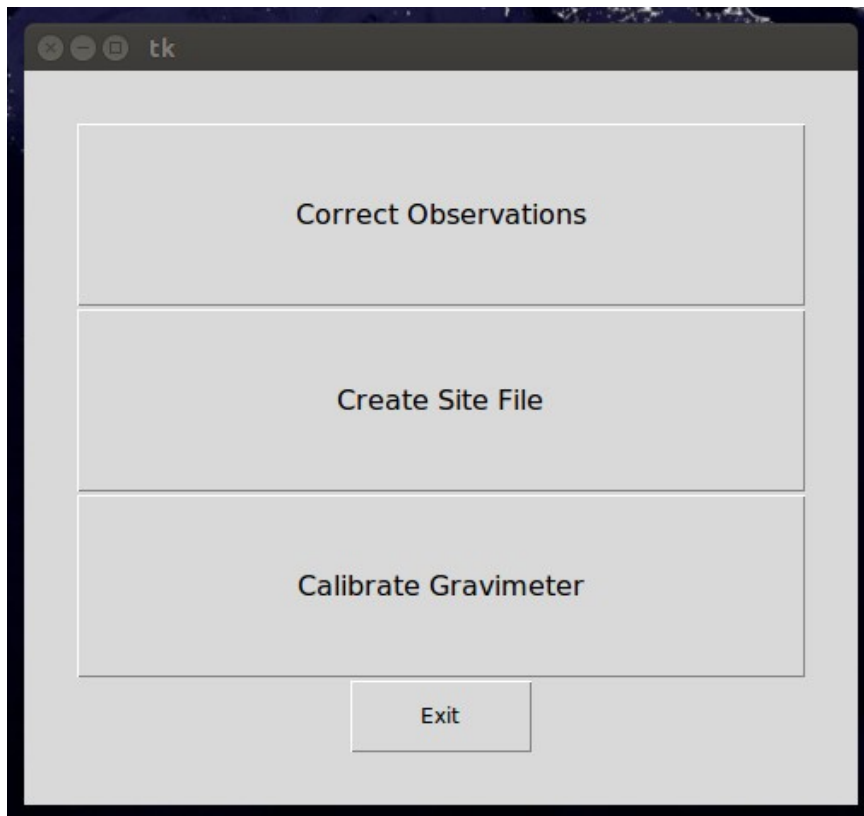
Copy this file in a sub directory (in the calculation directory) named from the day of year. You can now open the file with a text editor as Notepad++ (windows) or `gedit` (linux). You have to clean the file :

- erase the first and the last lines if they're empty
- erase the line beginning by / CG-5 SOFTWARE VER .
- erase the lines beginning by / between the measures
- erase all the blank lines between the measures
- erase the last line if it's not a measure line.

If some measures are wrong and you don't want these in the calculation, you can comment with # in the beginning of the line. Save the file.

You have to make sure that the setting file of the gravimeter contains the right number of the device. Open the file CG3TOOL.init and check or correct the number. If it is not correct, the software won't be able to run.

To open the software, go in the Pycgxtool directory. You will discover plenty of files. Don't panic !! You just need the file `__main__.py`. Open it in an interpreter as Spyder and run it. A window will appear. If you're working on Linux or Mac OS, you can launch the program with a terminal. In the directory, open a terminal and write : `__main__.py install` and press return. The same window will be opened.



3. Correct Observations

On the opening window, choose `Correct Observation` to calculate the difference of gravity with all the corrections.

In `INPUT FILES`, choose the file `f**n#oyy.jjj` in observations. You can now choose to input a coordinates file, a sites files for the site correction and a BLQ file to correct ocean loading. If you don't want to correct earth tide, tick the box `Do Not Correct Earth Tide`.

In `OUTPUT FILES`, you have to choose where you want to save the results files and their name :

- Calculated = c file
- Result = r file
- Graphics = two pdf with sum up of results and graphics

By default the name and the location of the output files is in the sub directory of your observation file. If you click on one of the box, a window appears and you can change the location of the file. If the INPUT FILE is not an observation file f**n#oyy.jjjj, the software won't load the file until you put an O file.

When all the input files are ready, click on the button process to launch the calculation and the creation of the output files. If one of the files (or more) is not recognized by the software, you will be warned by a message that pycgxtool is not able to run the calculation.

Example of a observation file cleaned and ready to feed pycgxtool :

```

/      CG-5 SURVEY
/      Survey name:      210716PLAT
/      Instrument S/N:   41215
/      Client:          1
/      Operator:        GUI
/      Date:            2016/ 7/21
/      Time:            07:18:07
/      LONG:            5.5900000 E
/      LAT:             44.0000000 N
/      ZONE:            0
/      GMT DIFF.:      0.0

/      CG-5 SETUP PARAMETERS
/      Gref:            0.000
/      Gcall:           8157.819
/      TiltxS:          681.205
/      TiltyS:          635.902
/      TiltxO:          10.007
/      TiltyO:          21.560
/      Tempco:          -0.135
/      Drift:           0.000
/      DriftTime Start: 07:18:21
/      DriftDate Start: 2016/07/21

/      CG-5 OPTIONS
/      Tide Correction: YES
/      Cont. Tilt:      YES
/      Auto Rejection: YES
/      Terrain Corr.:   NO
/      Seismic Filter: YES
/      Raw Data:        NO
/-----LINE-----STATION-----ALT.-----GRAV.----SD.---TILTX--TILTY--TEMP---TIDE---DUR-
REJ-----TIME----DEC.TIME+DATE--TERRAIN--DATE
0.0000000  1.0000000  -0.38000  4742.833  0.008    0.2   -1.2  -2.20  -0.051  85   0
07:28:03   42541.31065   0.0000  2016/07/21
0.0000000  1.0000000  -0.38000  4742.832  0.007    -0.8   -1.5  -2.22  -0.050  85   0
07:30:29   42541.31234   0.0000  2016/07/21
0.0000000  1.0000000  -0.38000  4742.830  0.008    -2.9   -1.7  -2.24  -0.048  85   0
07:32:56   42541.31403   0.0000  2016/07/21
0.0000000  2.0000000  -0.19000  4743.034  0.006    -1.2   -3.6  -2.11  -0.033  85   0
07:57:53   42541.33133   0.0000  2016/07/21

```

0.0000000	2.0000000	-0.19000	4743.036	0.008	-0.9	-4.8	-2.14	-0.031	85	0
08:00:18	42541.33301	0.0000	2016/07/21							
0.0000000	2.0000000	-0.19000	4743.041	0.008	-0.4	-5.7	-2.18	-0.029	85	0
08:02:42	42541.33467	0.0000	2016/07/21							
0.0000000	3.0000000	-0.27200	4742.773	0.011	-3.9	-0.8	-2.18	-0.006	85	5
08:36:15	42541.35793	0.0000	2016/07/21							
0.0000000	3.0000000	-0.27200	4742.775	0.008	-4.9	-0.3	-2.21	-0.003	85	0
08:38:54	42541.35977	0.0000	2016/07/21							
0.0000000	3.0000000	-0.27200	4742.782	0.008	-2.6	-0.5	-2.25	0.000	85	0
08:42:24	42541.36220	0.0000	2016/07/21							
0.0000000	4.0000000	-0.20000	4742.038	0.009	-3.5	-5.0	-2.23	0.013	85	1
09:02:12	42541.37593	0.0000	2016/07/21							
0.0000000	4.0000000	-0.20000	4742.036	0.015	-4.4	-1.4	-2.25	0.015	85	19
09:04:35	42541.37758	0.0000	2016/07/21							
0.0000000	4.0000000	-0.20000	4742.041	0.008	-4.6	-1.7	-2.27	0.017	85	9
09:06:47	42541.37910	0.0000	2016/07/21							
0.0000000	3.0000000	-0.27200	4742.797	0.008	-3.2	-1.2	-2.23	0.030	85	4
09:24:36	42541.39146	0.0000	2016/07/21							
0.0000000	3.0000000	-0.27200	4742.796	0.007	-6.1	-1.8	-2.26	0.032	85	7
09:27:02	42541.39314	0.0000	2016/07/21							
0.0000000	3.0000000	-0.27200	4742.801	0.006	-3.3	-2.5	-2.27	0.034	85	0
09:29:29	42541.39484	0.0000	2016/07/21							
0.0000000	2.0000000	-0.19000	4743.080	0.008	-0.4	-4.2	-2.16	0.054	85	0
09:56:25	42541.41352	0.0000	2016/07/21							
0.0000000	2.0000000	-0.19000	4743.077	0.009	-0.3	-9.2	-2.19	0.055	85	0
09:59:02	42541.41533	0.0000	2016/07/21							

4.Create Site file

You can create a site file which can be used in `Correct Observations` to calculate the earth tide. In `INPUT FILES` choose your observations. In `OUTPUT FILES` the name and the location of the site file will be by default the location of your sub directory. If you want to change it, click on it and choose your directory.

Note that the earth tide can be corrected without site file, but it will be less precise.

5.Calibration

To calibrate a relative gravimeter compared to an other, you will choose in `INPUT FILES` the results files of the two devices for the same process. The `OUTPUT FILE` will give you the differences between the two devices. One `INPUT FILE` corresponds to the gravimeter to calibrate, the other one is the reference file.

6.Errors

An error message will appear in case of :

- the user don't input the observation file
- the format of one or more lines of a file does not permit the calculation

- the number of gravimeter in the observation file does not match with gravimeters of the setting file
- the setting file for gravimeters is not in the right location in the tree view.

7.Results

The first pdf page presents the raw values. You will find the G (mGal) depending on time, the error (mGal) depending on time, the TILT (Arcsec) for x and y depending on time, the temperature (mK) depending on time, the repeatability (mGal) depending on Δt (minute) (the grey area gives the confidence interval) and a histogram of errors (μ Gal). The second page shows ΔG (mGal) depending on time (minute) and a box containing DATA, CORRECTION and DRIFT. In DRIFT, the correlation coefficient of the least squares method is given, it shows reliability of the least squares.

