

Function description of the SGG Pre-Processing tab

GravLab Team

Contents

single_vs_multi_processing	2
single_process_up_2.....	3
multi_process_up_3	4
checking_NOM_PSO	5
browse_data	6
browse_data_2	7
stats_GRF	8
plot_EGG_NOM_2	9
GOCE_orbit_track	10
altitude_GOCE_GRF	11
input_graflab.....	12
graflab_gui_geograv	13
*gps2utc.....	14
*utc2gps.....	14
References	14

single_vs_multi_processing

Description:

single_vs_multi_processing enables the single processing or multiprocessing, according to the input data,. Multiprocessing is enabled when the user loads more than three products (3 EGG_NOM & 3 SST_PSO)

Syntax:

```
[app.GG_coords_GRF,stop_process] =  
single_vs_multi_processing(NOM_data_0,counter_NOM_0,PSO_data_0,counter_PSO_0,path_  
NOM,path_PSO);
```

Input variables:

Variable name	Size	Description
NOM_data_0	1xn	Contains the EGG_NOM_2 data.
counter_NOM_0	1x1	Counter, indicates if the EGG_NOM_2 products have been uploaded successfully.
PSO_data_0	1xn	Contains the SST_PSO_2 data.
counter_PSO_0	1x1	Counter, indicates if the SST_PSO_2 products have been uploaded successfully.
path_NOM	1x1	Contains information about the full path of EGG_NOM_2.
path_PSO	1x1	Contains information about the full path of SST_PSO_2.

Output variables:

Variable name	Size	Description
app.GG_coords_GRF	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
stop_process	1x1	Counter/ checks when the whole process must be stopped.
SGG_GRF.mat	15x1	The main output of the tab. Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
SGG_GRF_Report.txt	-	Report regarding to the file format .

single_process_up_2

Description:

single_process_up_2 is used in single processing. It creates the "SGG Pre-processing" folder, calls the checking_NOM_PSO and the browse_data_2 functions, classifies, and saves the computed data.

Syntax:

```
[GG_coords_GRF,stop_process] =  
single_process_up_2(NOM_data_0,PSO_data_0,counter_NOM_0,counter_PSO_0,path_NOM,  
path_PSO);
```

Input variables:

Variable name	Size	Description
NOM_data_0	1xn	Contains the EGG_NOM_2 data.
PSO_data_0	1xn	Contains the SST_PSO_2 data.
counter_NOM_0	1x1	Counter, indicates if the EGG_NOM_2 products have been uploaded successfully.
counter_PSO_0	1x1	Counter, indicates if the SST_PSO_2 products have been uploaded successfully.
path_NOM	1x1	Contains information about the full path of EGG_NOM_2.
path_PSO	1x1	Contains information about the full path of SST_PSO_2.

Output variables:

Variable name	Size	Description
GG_coords_GRF	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
stop_process	1x1	Counter/ checks when the whole process must be stopped.

multi_process_up_3

Description:

multi_process_up_3 is used in multiprocessing. It starts the parallel pool of workers (one worker per physical CPU core) in Matlab, creates the "SGG Pre-processing" folder, calls the checking_NOM_PSO and the browse_data_2 functions, and finally classifies and saves the computed data, corresponding to the circumstances of the parallel computation.

Syntax:

```
[GG_coords_GRF,stop_process] =  
multi_process_up_3(NOM_data_0,PSO_data_0,counter_NOM_0,counter_PSO_0,path_NOM,path_PSO);
```

Input variables:

Variable name	Size	Description
NOM_data_0	1xn	Contains the EGG_NOM_2 data.
PSO_data_0	1xn	Contains the SST_PSO_2 data.
counter_NOM_0	1x1	Counter, indicates if the EGG_NOM_2 products have been uploaded successfully.
counter_PSO_0	1x1	Counter, indicates if the SST_PSO_2 products have been uploaded successfully.
path_NOM	1x1	Contains information about the full path of EGG_NOM_2.
path_PSO	1x1	Contains information about the full path of SST_PSO_2.

Output variables:

Variable name	Size	Description
GG_coords_GRF	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
stop_process	1x1	Counter/ checks when the whole process must be stopped.

checking_NOM_PSO

Description:

checking_NOM_PSO checks if the EGG_NOM_2 name corresponds precisely to the SST_PSO_2 name before the parallel computation.

Syntax:

```
[result] = checking_NOM_PSO(NOM_data_0,PSO_data_0);
```

Input variables:

Variable name	Size	Description
NOM_data_0	1xn	Contains the EGG_NOM_2 data.
PSO_data_0	1xn	Contains the SST_PSO_2 data.

Output variables:

Variable name	Size	Description
result	1x1	Counter/ if the result=0, then EGG_NOM_2 name corresponds precisely to the SST_PSO_2.

browse_data

Description:

browse_data is used in the single processing. The function classifies the EGG_NOM_2 and SST_PSO_2 data and compares their lengths. It adjusts the NOM data according to the length of PSO data. It interpolates the (X, Y, Z) coordinates and then calculates the geodetic coordinates (lon, lat, altitude) in GRS80.

Syntax:

```
[GG_coords_GRF,stop_process] =  
browse_data(NOM_data_0,counter_NOM_0,PSO_data_0,counter_PSO_0,path_NOM,path_PSO);
```

Input variables:

Variable name	Size	Description
NOM_data_0	1xn	Contains the EGG_NOM_2 data.
counter_NOM_0	1x1	Counter, indicates if the EGG_NOM_2 products have been uploaded successfully.
PSO_data_0	1xn	Contains the SST_PSO_2 data.
counter_PSO_0	1x1	Counter, indicates if the SST_PSO_2 products have been uploaded successfully.
path_NOM	1x1	Contains information about the full path of EGG_NOM_2.
path_PSO	1x1	Contains information about the full path of SST_PSO_2.

Output variables:

Variable name	Size	Description
GG_coords_GRF	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
stop_process	1x1	Counter/ checks when the whole process must be stopped.

browse_data_2

Description:

browse_data_2 is used in the multiprocessing. The function classifies the EGG_NOM_2 and SST_PSO_2 data and compares their lengths. It adjusts the NOM data according to the length of PSO data. It interpolates the (X, Y, Z) coordinates and then calculates the geodetic coordinates (lon, lat, altitude) in GRS80.

Syntax:

```
[GG_coords_GRF1] =  
browse_data_2(NOM_data_0{i},counter_NOM_0,PSO_data_0{i},counter_PSO_0,path_NOM,pa  
th_PSO);
```

Input variables:

Variable name	Size	Description
NOM_data_0{i}	1x1	Contains the EGG_NOM_2 data.
counter_NOM_0	1x1	Counter, indicates if the EGG_NOM_2 products have been uploaded successfully.
PSO_data_0{i}	1x1	Contains the SST_PSO_2 data.
counter_PSO_0	1x1	Counter, indicates if the SST_PSO_2 products have been uploaded successfully.
path_NOM	1x1	Contains information about the full path of EGG_NOM_2.
path_PSO	1x1	Contains information about the full path of SST_PSO_2.

Output variables:

Variable name	Size	Description
GG_coords_GRF1	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.

stats_GRF

Description:

stats_GRF computes the statistics (min, max, mean, std, rms) of the original GOCE gravity gradients ($V_{xx}, V_{yy}, V_{zz}, V_{xy}, V_{xz}, V_{yz}$) referred to the GRF.

Syntax:

[~] = stats_GRF(app.GG_coords_GRF,currentFolder)

Input variables:

Variable name	Size	Description
app.GG_coords_GRF	17x1	The output of the Calculate SGG button. Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions.
currentFolder	-	The SGG Pre-processing working folder.

Output variables:

Variable name	Size	Description
stats_Vij_GRF.mat	nx6	Statistics of SGGs in GRF.
stats_Vij_GRF_Report.txt	-	Report regarding to the file format .

plot_EGG_NOM_2

Description:

plot_EGG_NOM_2 creates figures of the GOCE GGs ($V_{xx}, V_{yy}, V_{zz}, V_{xy}, V_{xz}, V_{yz}$) in GRF, which show how the GOCE signal changes with time.

Syntax:

```
[~] = plot_EGG_NOM_2( app.GG_coords_GRF,currentFolder);
```

Input variables:

Variable name	Size	Description
app.GG_coords_GRF	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
currentFolder	-	the SGG Pre-processing working folder.

Output variables:

Figure name	Size	Description
Vij_GRF__date.jpeg	-	A figure in .jpeg format is saved in the SGG Pre-Processing/ Gravity Gradients GRF folder.
Vij_GRF__date.fig	-	A figure in .fig format is saved in the SGG Pre-Processing/ Gravity Gradients GRF folder.

GOCE_orbit_track

Description:

GOCE_orbit_track creates global maps with the GOCE orbit track, via the m_map mapping toolbox (Pawlowicz 2020), with the condition that the m_map M-file exists in the user's search path.

Syntax:

```
[ k ] = GOCE_orbit_track( app.GG_coords_GRF,currentFolder);
```

Input variables:

Variable name	Size	Description
k	1x1	Counter/ is needed for further actions in GUI.
app.GG_coords_GRF	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
currentFolder	-	the SGG Pre-processing working folder.

Output variables:

Figure name	Size	Description
GOCE Orbit Track_date .jpeg	-	A figure in .jpeg format is saved in the SGG Pre-Processing/ GOCE Orbit Track folder.
GOCE Orbit Track_date .fig	-	A figure in .fig format is saved in the SGG Pre-Processing/ GOCE Orbit Track folder.

altitude_GOCE_GRF

Description:

altitude_GOCE_GRF creates figures with the GOCE altitude in kilometres, which show how the GOCE satellite altitude changes with time.

Syntax:

```
[ k ] = altitude_GOCE_GRF( app.GG_coords_GRF,currentFolder);
```

Input variables:

Variable name	Size	Description
k	1x1	Counter/ is needed for further actions in GUI.
app.GG_coords_GRF	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
currentFolder	-	the SGG Pre-processing working folder.

Output variables:

Figure name	Size	Description
Altitude_GOCE_GRF_date .jpeg	-	A figure in .jpeg format is saved in the SGG Pre-Processing/ GOCE Altitude folder.
Altitude_GOCE_GRF_date .fig	-	A figure in .fig format is saved in the SGG Pre-Processing/ GOCE Altitude folder.

input_graflab

Description:

input_graflab creates the appropriate input format (latitude, longitude, height) for the "GrafLab" software (Bucha and Janák 2013).

Syntax:

```
[ ~ ] = input_graflab( GG_coords_GRF1,currentFolder);
```

Input variables:

Variable name	Size	Description
GG_coords_GRF1	17x1	Contains info about the latitude, longitude, altitude, GPS time, gravity gradients in GRF and the quaternions in GRF.
currentFolder	-	the SGG Pre-processing working folder.

Output variables:

Variable name	Size	Description
data-date.txt	nx3	Contains the gravity gradient's geodetic coordinates in an appropriate format (<i>latitude, longitude, heigth</i>). The latitude and the longitude in degrees, and the ellipsoid height in meters.

graflab_gui_geograv

Description:

graflab_gui_geograv computes the Gravitational tensor components ($V_{xx}, V_{yy}, V_{zz}, V_{xy}, V_{xz}, V_{yz}$) in the LNOF via "Graflab" software (Bucha and Janák 2013) for multiple input files .

Syntax:

```
[k,stop_counter] = graflab_gui_geograv();
```

Output variables:

Variable name	Size	Description
k	1x1	Counter/ is needed for further actions in GUI.
stop_counter	1x1	Counter/ is needed for further actions in GUI.
ai_GGM_name nmax_n__ user's_input_file.txt	nx9	Contains the Ellipsoidal Latitude (deg), the Longitude (deg), the Height above the reference ellipsoid (m) and the computed gravitational tensor components (Eotvos).
ai_GGM_name nmax_n__ user's_input_file_Report.t xt	-	The corresponding Graflab's report.

(*) The output name of the saved data is creating as the following:

ai_GGM_name nmax_n__ user's_input_file.txt

Where: "i" is a counter, "GGM_name" presents the name of the used global gravity model, "n" is the maximum degree of the spherical harmonic expansion, and the "user's_input_file" depicts the user's input file name with the geodetic coordinates.

For instance, a Graflab output file via the GeoGravGOCE GUI:

a1_GO_CONS_GCF_2_TIM_R6__nmax_240__data-01-Jan-2010.txt

a1_GO_CONS_GCF_2_TIM_R6__nmax_240__data-01-Jan-2010_Report.txt

*gps2utc

Description:

gps2utc converts GPS time tags to UTC (GMT) time accounting for leap seconds (Howat 2021).

<https://www.mathworks.com/matlabcentral/fileexchange/21194-gps-to-utc-gmt-time-conversion>

*utc2gps

Description:

utc2gps corrects an array of GPS dates (in any matlab format) for leap seconds and returns an array of UTC datenums (Howat 2021).

<https://www.mathworks.com/matlabcentral/fileexchange/21193-utc-to-gps-time-coverter>

References

Howat, I. (2020a) *GPS to UTC(GMT) time conversion*, *MATLAB Central File Exchange*. Available at: <https://www.mathworks.com/matlabcentral/fileexchange/21194-gps-to-utc-gmt-time-conversion> (Accessed: 17 June 2020).

Howat, I. (2020b) *UTC to GPS time coverter*, *MATLAB Central File Exchange*. Available at: <https://www.mathworks.com/matlabcentral/fileexchange/21193-utc-to-gps-time-coverter> (Accessed: 17 June 2020).