Data structure of IGG MSS

Introduction to Exercise 1
WS 2020/2021

Tomislav Medic

Institut für Geodäsie und Geoinformation
Universität Bonn

MSS data

- all data from the MSS is stored in a special IMAR.mat file:

Name	Date modified	Type	Size
IMAR	15/09/2017 14:04	Microsoft Access ...	22,885 KB
Pop2ImarZf_GpsInfo20002.Dat	15/09/2017 13:12	DAT File	249 KB
Pop2ImarZf_GpsInfo20002	15/09/2017 14:04	Microsoft Access ...	8 KB
Pop2ImarZf_GPSPos0002.Dat	15/09/2017 13:12	DAT File	114 KB
Pop2ImarZf_GPSPos0002	15/09/2017 14:04	Microsoft Access ...	52 KB
Pop2ImarZf_GpsTime0002.Dat	15/09/2017 13:12	DAT File	53 KB
Pop2ImarZf_IMS0002.Dat	15/09/2017 13:12	DAT File	30,551 KB
Pop2ImarZf_IMS0002	15/09/2017 14:04	Microsoft Access ...	22,826 KB
Pop2ImarZf_XIODump0002.Asc	15/09/2017 14:01	ASC File	33 KB
Pop2ImarZf_XIODump0002.DMP	15/09/2017 13:12	DMP File	47,663 KB

- when loaded in Matlab/Octave it is a structure with three substructures:
- **gps**
- ~~**gps_info**~~
- **imu**

Workspace

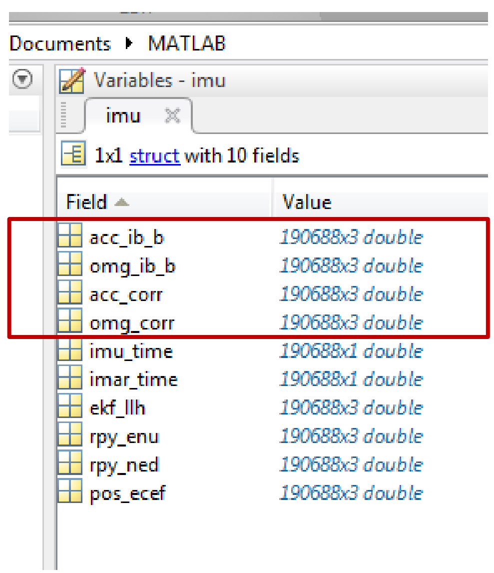
Name	Value
gps	1x1 struct
~~gps_info~~	~~1x1 struct~~
imu	1x1 struct

MSS data

IMU structure:

- **acc_ib_b** – accelerometer measurements in matrix $\boldsymbol{f}_{ib}^{b} = \begin{bmatrix} f_{ib,x}^{b} & f_{ib,y}^{b} & f_{ib,z}^{b} \end{bmatrix}_{nx3}$ [ms^{-2}]
- **omg_ib_b** – gyroscope measurements in matrix $\boldsymbol{\omega}_{ib}^{b} = \begin{bmatrix} \omega_{ib,x}^{b} & \omega_{ib,y}^{b} & \omega_{ib,z}^{b} \end{bmatrix}_{nx3}$ [rads^{-1}]
- **acc_corr** & **omg_corr** – measurements corrected for gravity influence and earths rotation rate
  - you can use this values with simplified algorithm if you omit gravity correction – (which is already applied in this data set)
  - this way you should get much better trajectory

- Comparison can be included in final presentation



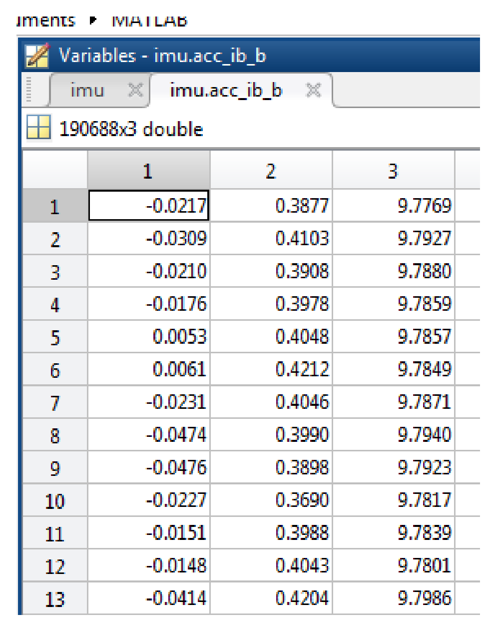
Documents ▸ MATLAB

Variables - imu

imu

1x1 struct with 10 fields

Field	Value
acc_ib_b	190688x3 double
omg_ib_b	190688x3 double
acc_corr	190688x3 double
omg_corr	190688x3 double
imu_time	190688x1 double
imar_time	190688x1 double
ekf_llh	190688x3 double
rpy_enu	190688x3 double
rpy_ned	190688x3 double
pos_ecef	190688x3 double



Variables - imu.acc_ib_b

imu | imu.acc_ib_b

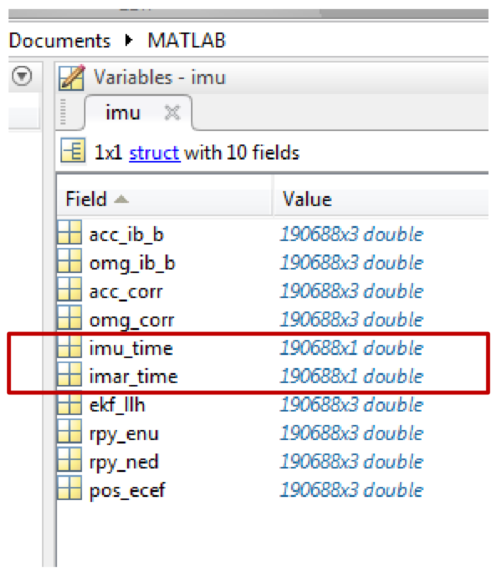
190688x3 double

	1	2	3
1	-0.0217	0.3877	9.7769
2	-0.0309	0.4103	9.7927
3	-0.0210	0.3908	9.7880
4	-0.0176	0.3978	9.7859
5	0.0053	0.4048	9.7857
6	0.0061	0.4212	9.7849
7	-0.0231	0.4046	9.7871
8	-0.0474	0.3990	9.7940
9	-0.0476	0.3898	9.7923
10	-0.0227	0.3690	9.7817
11	-0.0151	0.3988	9.7839
12	-0.0148	0.4043	9.7801
13	-0.0414	0.4204	9.7986

MSS data

IMU structure:

- **imu_time** – time stamp connected to GPS time – use it to define $t_0, t_1, t_2, \ldots$ [s]
  - 1st value in imu_time vector is the initial time t_0
- **imar_time** – another time stamp (you do not need to use it)



MSS data

IMU structure:

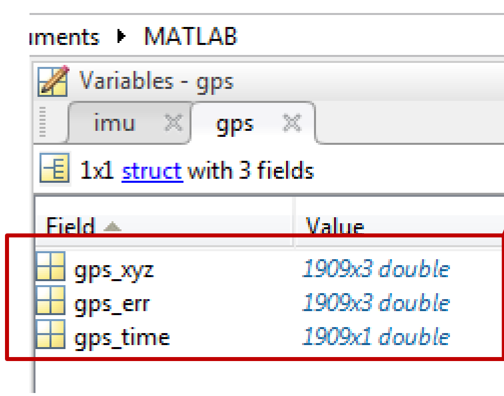
- **ekf_llh** – estimate latitude longitude and ellipsoid height of the trajectory – using kalman filter inside the MSS (probably incorrect – you do not need to use it)
- **rpy_enu** – estimated roll pitch & yaw angle for every moment in time (updated attitudes) – in EAST, NORTH, UP coordinate system (you do not need to use it)
- **rpy_ned** - roll pitch & yaw angle for every moment in time (updated attitudes) in North, East, Down reference frame, solution provided by the instrument
  - Values in the 1st row should be used for initial attitude values
  - It can be used for control of your own attitude updates
  - Comparison can be included in the final presentation
- **pos_ecef** – estimated XYZ in ECEF coordinate frame of the trajectory – recalculated from ekf_llh (probably incorrect – you do not need to use it)

acc_ib_b	190688x3 double
omg_ib_b	190688x3 double
acc_corr	190688x3 double
omg_corr	190688x3 double
imu_time	190688x1 double
imar_time	190688x1 double
ekf_llh	190688x3 double
rpy_enu	190688x3 double
rpy_ned	190688x3 double
pos_ecef	190688x3 double

MSS data

GPS structure:

- **gps_xyz** – positions estimated using only GNSS data (longitude, latitude, height)
  - Values in the 1st row should be used for initial position values
  - **Pay attention on the order of values in matrix (Long, Lat, height)**
  - Your computed navigation solution should be graphically compared with GNSS data !
- **gps_err** – estimated errors of GNSS measurements (you do not need to use it)
- **gps_time** – time stamp for all GNSS measurements – synchronised with imu_time, useful for the purpose of comparison



Long [rad] Lat [rad] h [m]

	1	2	3
1	0.1237	0.8854	107.2903
2	0.1237	0.8854	107.2908
3	0.1237	0.8854	107.2899
4	0.1237	0.8854	107.2887
5	0.1237	0.8854	107.4177
6	0.1237	0.8854	107.4337
7	0.1237	0.8854	107.6470
8	0.1237	0.8854	107.6375
9	0.1237	0.8854	107.6374
10	0.1237	0.8854	107.6519
11	0.1237	0.8854	107.6324
12	0.1237	0.8854	107.6418