2018-2019

Data Analysis Manual

Marianne Schaaphok
Nova Electric Racing
2018-2019

Index

1. Getting the data

The first step towards data analysis is to get data sets. The data sets are made during testing or during the actual races. Data from the bike can be very useful to detect problems and to improve the performance of the bike. In this small section you can read which data can currently be retrieved from the DAQ system and how to save the data correctly.

1.1 DAQ system

The data is retrieved using the Data Acquisition Unit (DAQ), which is fixed to the bike. The documentation on the DAQ can be found on Confluence. Here we shortly state which sensors the DAQ contains, which data we can get and how it is and should be saved.

1.1.1 Sensors and Data

The DAQ gets the data from the sensors that are placed in the system and from the Battery Management System and the motor controller. Note that the data from every sensor has its own time frame. At this moment the DAQ contains the following sensors:

1. GPS
2. Accelerometer
3. Gyroscope

And it returns the following data:

Data	**Unit**
GPS coordinates (gps) - Longitude - Latitude	degrees
Gyroscope (gyro) - X - Y - Z	Lbs/s
Accelerations (Acc) - X - Y - Z	m/s^2
BMS Voltage (BMS_V) - Min - Max - Average - Total	V
BMS Current & Charge (BMS_C)	
BMS Temperature (BMS_T) - Min - Max - Average	
Motor controller temperature (MC_T)	
Motor controller power source temperature (MC_PS)	

Motor controller motor temperature (MC_m)	
Motor controller speed (MC_Speed)	
Motor controller current (MC_Current) - pA - pB - pC - DC	
Motor controller Voltage (MC_Voltage) - DC - Output - VAB - VBC	
Motor controller Flux - Com - FB - Id - Iq	
Motor controller Fault - Post_lo/post_hi - Run_lo/run_hi	
Motor controller torque (MC_Torque) - Com - FB	

1.1.2 DAQ converter

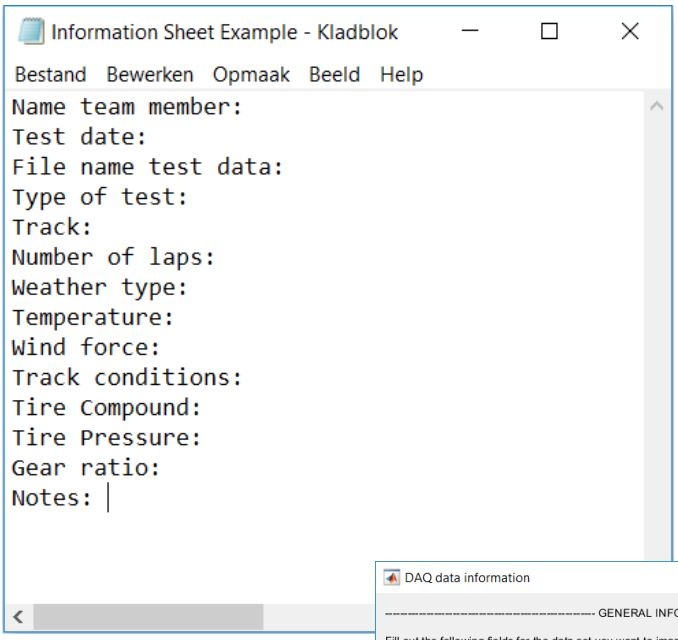
The DAQ is programmed using Labview and at this moment the information from the DAQ first needs to be loaded into Labview in order to convert it to a CSV file which can be further analyzed. The DAQ does not work in real-time, so the data can only be collected after the test/race is finished.

If the datafile is still in DAQ format and not yet in CSV, you will need to install the 'DAQconverter' app. In the *Data management* folder is a folder named '*DAQInstaller'*. Open and run the setup.exe file. Open de installed program, select de .daq file you wish to convert and the program will save the converted .csv file in the same file as the original dataset.

1.2 Saving the data

After the data has been converted to CSV it is important that the data set is saved with all the information regarding the circumstances in which the data was made. It is important to know for example which tires and gear ratio was used, at which track the data was made, whether it was raining or windy as this all influences the results obtained from the data.

In order to make sure that the circumstances of the data set are always retrievable, next to the csv file containing the data, a .txt file (with the same name as the data set file) needs to be saved containing the following information:



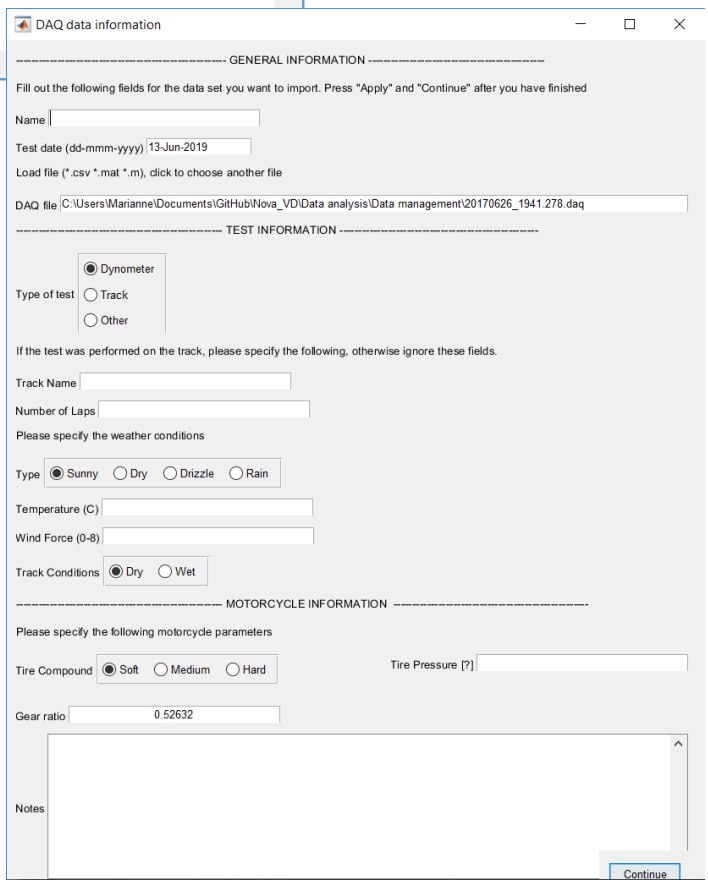
If the MATLAB program is used, the following question will be asked:

"*Does this dataset already have a corresponding information sheet (.txt)?* "

If you answer "no" here, the program will be provide you with a sheet to fill out. (see image on the right). This will automatically save a .txt file with the corresponding information in the folder with your data set.

At the notes section you can add whatever you think is relevant, e.g. the main purpose of the test run, which rider etc.

It will automatically select a file from your current folder, **don't forget** to check if it is the correct file. If not click to change the file!

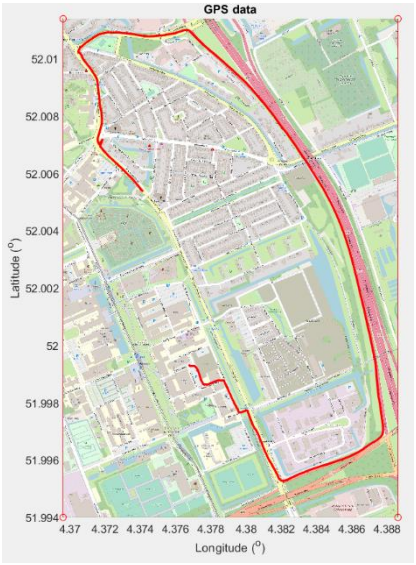


2. User information

In this chapter you can read how to use the MATLAB data analysis program for data analysis. You can also find some guidelines on how to interpret the data.

2.1 How to use the MATLAB program?

First of course you need to have MATLAB installed on the computer you are using. Secondly you need to have downloaded and unpacked the zip-file containing the MATLAB programs from Confluence. If correct these contain all the functions you will need. Now you can start with the data analysis!



Steps to be taken

1. Open the file "DataAnalysisMain.m" in MATLAB.
2. Press the green "RUN" button. In principal you can just follow the instructions on the screen, but below there is further explanation.
3. The program will ask:
   "Does this dataset already have a corresponding information sheet (.txt)? [yes,no,y,n]"
   The information sheet is the file explained in 1.2. If you have already made one before, type 'yes', or 'y'. If not type 'no' or 'n' and press ENTER.
4. Now you are asked to choose a file. You can choose (.csv or .mat). You might need to change the extension in the file viewer in order to be able to see the .mat files.
5. The program reads the loaded data set and asks the following:
   "Do you want to plot the gps data on the map? [yes,no,y,n]"
   This will plot the gps data on map of the surroundings, like in the image below. Note that constructing this image will take some time, so if you don't want to do anything with it, you can better choose 'no'.
6. In order to be able to show the data per sector the program will ask:
   "Select one of the following tracks in order to plot sectors? ["Assen"," "]"
   If the data is not from a track or the track is not in the list, type " ", in that case the sectors are not available.
7. The program will now open a figure GUI with 3 subplots and a panel with buttons and dropdown menu's. It will also open a second GUI with 4 channel subplots, which can plot the data against the distance. Note that sometimes it can require some time to update all the graphs.
   You can play around with the parameters.

Options for the GUI

If you change the 'x-label' or 'y-label' you need to click 'refresh'.
If you change the 'graph' option it updates automatically.
The 'extract figure' option provides a separate figure which allows for zooming.

X-label options:
- Track: Views the data in colors over the track
- Time: Views the data over total time
- Distance: Views the data over the total distance

Y-label options:
GPS, Acc, Gyro, Velocity, BMS_T, BMS_C, BMS_V, MC_m, MC_T, MC_Speedr, MC_PS, MC_Current, MC_Voltage, MC_Flux, MC_Fault, MC_Torque.
Meaning and Units can be found in the table in 1.1.1.

Graph options:
Gives the sub-variables of Y-label options for the figures. E.g. only plot minimum temperature.

Lap option:

Sectors option:
The GUI also has the function that it can show only certain sectors.

2.2 What conclusions to draw?

Now that we are able to view the data it is also important to know what to look for. Of course it is hard to write down exactly what we will see in the new data set and what to conclude from that, but below we give a few guidelines on what to look for. Further use your intuition and your knowledge on how it should behave.

Many information on motorcycle data analysis can be found on the webpage datamc.org.

TODO: Write guidelines

3. Editing the program

In order to edit the program, you will need to understand the MATLAB language and to have some experience with programming. Unless you only need to do small edits. Below there is an overview of the structure of the programs and the functions. More detailed documentation of the code can be found in the MATLAB Scripts.

Data information sheet

Data analysis GUI