Summer student report

A Web Interface for offline Malter current monitoring for the CMS Cathode Strip Chambers

Konstantin Stoychev | August 2018

Preface

My name is Konstantin Stoychev and I just finished the fourth year of my bachelor's degree in Physics at *Sofia University "St. Kliment Ohridski"*. I was a summer student at *CERN* for 8 weeks between the 25[th] of June and the 17[th] of August. During my internship I worked in the *CMS CSC* group under the supervision of Katerina Kuznetsova. My project was to create a web interface design for offline CSC Malter current monitoring.

Project description and motivation

Cathode Strip Chambers at *CMS*

The *CMS* detector consists of trackers, calorimeters and muon detectors installed in the barrel (MB) and endcap (ME) regions of the *CMS* detector as shown in *Fig.1*. The muon detectors in the barrel are drift tubes and resistive plate chambers while the ones in the endcaps are cathode strip chambers (CSCs) and resistive plate chambers. The CSCs are multi-wire proportional chambers whose cathode plates are separated into strips which allows to obtain two-dimensional coordinates of a charged particle penetrating the chamber. Each CSC has six layers to distinguish muons from background particles. The muons produce hits in all six layers while

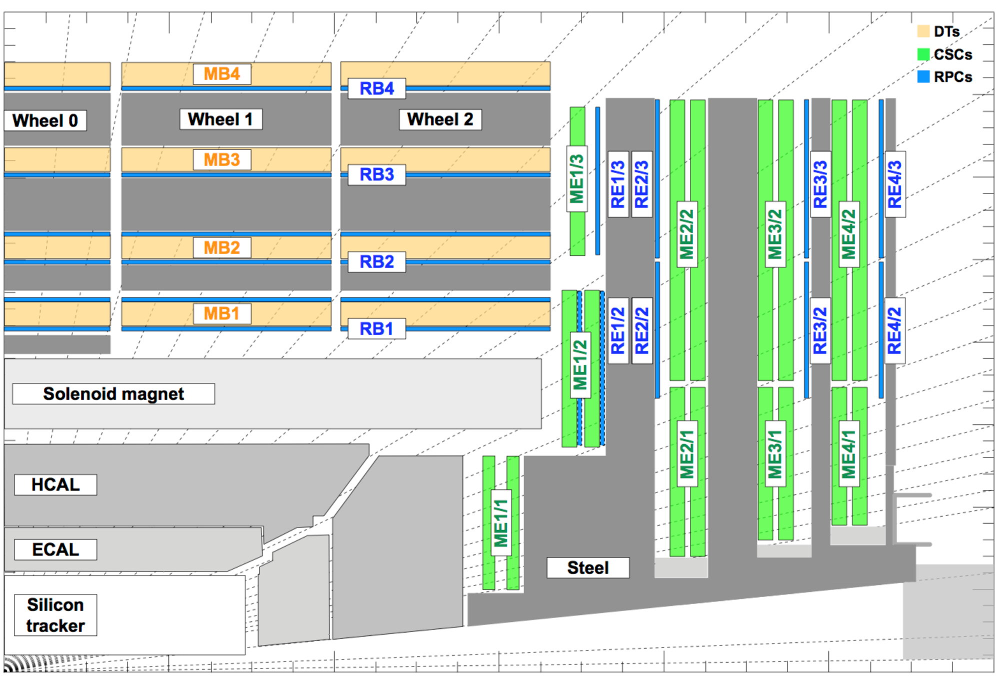

Fig.1: A quarter view of the CMS detector. The CSCs are located in the endcap region and are marked as ME#/#.

background particles – neutrons and gamma rays, produce hits only in one or two layers. Most of the CSCs are additionally divided radially into several High Voltage (HV) segments.

When a muon passes through a CSC layer, it produces electron-ion pairs. The electrons are accelerated towards the anode wires and begin avalanche development approaching them. The ions drift towards the cathode.

Malter effect

If the cathode surface is not perfect and a thin insulating layer is present on it, positive charges will accumulate on top of this layer (*Fig.2*). The accumulated positive charge produces a large electric field that results in secondary emission of electrons from the cathode. These electrons are then accelerated towards the anode wires and begin secondary avalanche development. The consequence of these processes may appear in the current readings as either a current spike or a fluctuating additional current on top of the nominal current values. This process is known as the Malter effect, and the resulting currents – Malter currents. When the insulating layer is localized it is a precondition for degrading (aging) of the detector and it is important to identify and monitor Malter currents.

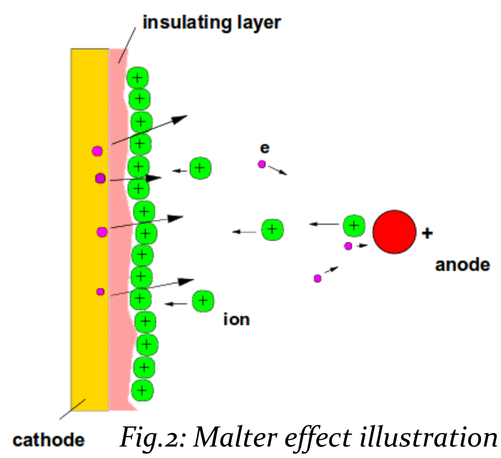

Fig.2: Malter effect illustration

CSC current monitoring

During the operation of the CSCs at *CMS*, the HV, current (I) and instantaneous luminosity (L) are recorded in the database. The initial analysis is done offline as a batch job automatically to obtain current dependence on the instantaneous collision luminosity. The results of this treatment are several plots for each HV channel including the current vs. time and current vs. luminosity correspondence. These plots can be used to identify problems such as Malter currents. An example of these plots for a channel with nominal performance is given in *Fig.3* and an example of a channel with Malter currents is given in *Fig.4*.

From the examples we see that a deviation from linear dependence and a large χ^2/NDF is the signature of a Malter current. The problem is that there are too many channels and each one has to be checked individually. And this has to be done for each new *LHC* fill.

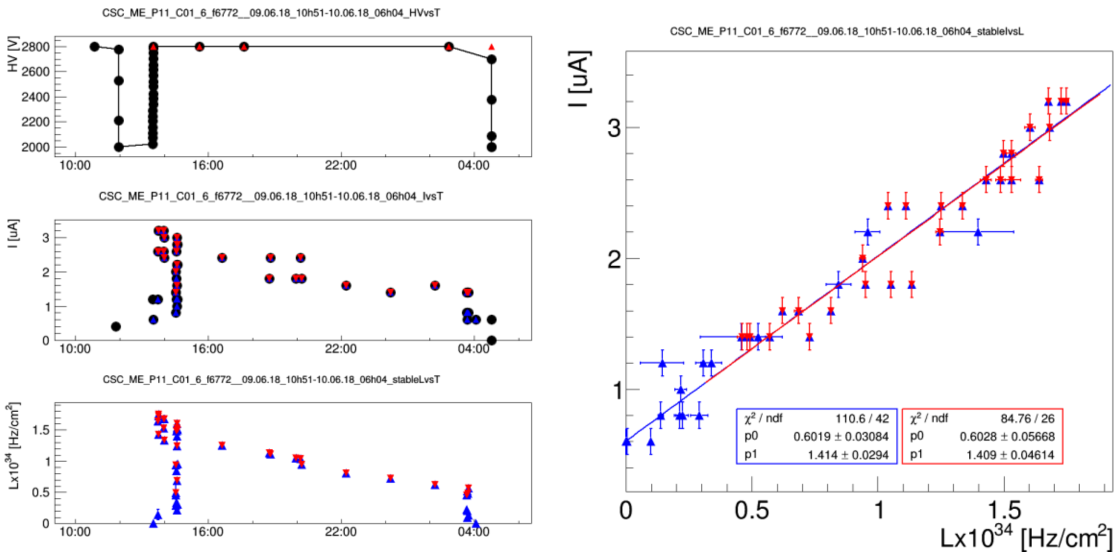

Fig.3: Example plots for a CSC HV channel with nominal performance. From top left: High Voltage vs. time, Current vs. time, Instantaneous Luminosity vs. time. On the right - Current vs. Instantaneous Luminosity. The red data points are only the points with small changes in luminosity.

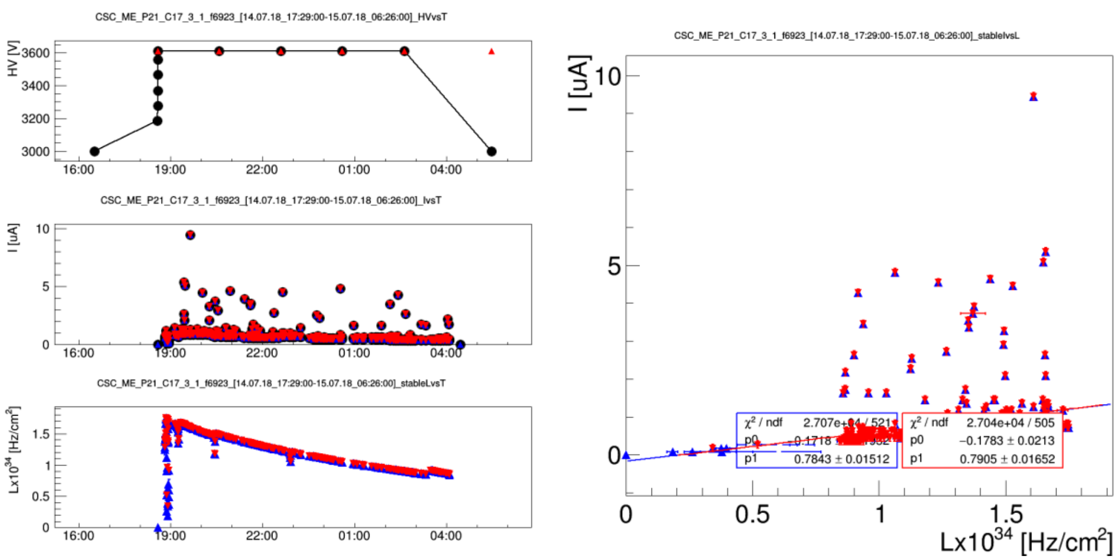

Fig.4: Example plots for a CSC HV channel with many Malter current spikes.

The goal of my project is to identify all problematic channels and to create a web interface to present the results in an accessible way. Thus the experts working on the offline CSC analysis can easily examine those problematic channels and find any tendencies in their performance over several fills.

Project development

To identify issues in the performance of each channel I learned *Python* and wrote a script using the *pandas* package. I used existing results of the current linearity analysis for station ME1/1+ for different fills. The reports contain the data for each individual channel. In total per every *LHC* fill there are 216 channels analyzed for station ME1/1+ only. My script has one argument which is a text file with a list of reports to analyze, shown on *Fig.5*. The report list also has the path to a *FillList* file, from which I get details about the fills – fill duration and instantaneous luminosity. For my work, I only used fills with a duration of more than 4 hours and peak instantaneous luminosity $L > 8 \times 10^{33}\ Hz/cm^2$.

```
## list of reports to update
/path/to/report/txt/files/
/path/to/fill/list/
/path/to/html/files/
f6759_ME11Pv10/f6759_04.06.18_20h06-05.06.18_13h33.txt
f6761_ME11Pv10/f6761_05.06.18_15h15-06.06.18_06h57.txt
f6762_ME11Pv10/f6762_06.06.18_05h01-06.06.18_20h18.txt
f6763_ME11Pv10/f6763_06.06.18_18h43-07.06.18_10h14.txt
f6770_ME11Pv10/f6770_08.06.18_17h17-09.06.18_11h48.txt
f6772_ME11Pv10/f6772_09.06.18_10h51-10.06.18_06h04.txt
f6773_ME11Pv10/f6773_10.06.18_04h08-10.06.18_21h39.txt
f6774_ME11Pv10/f6774_10.06.18_19h46-11.06.18_14h02.txt
```

Fig.5: This list of reports to be analyzed and added to the HTML pages is given as an argument to the Python script.

Problematic channel identification

The code opens the reports one by one, extracts only the columns I need and continues with the problem identification. I divided the possible problems in different groups and assigned them certain colors for the following visualization. The problem groups were defined according to the following hierarchy. The presence of Malter currents is categorized as of highest concern. Then big current deviations, which may also indicate in some cases Malter currents. After that any problems with too low HV or with a big HV deviation observed during a fill. Usually those channels are already well known to experts. Of least concern were rare cases corrupted output of the first analysis stage and the absence of HV or current data for the channel, which may happen if the channel was switched off. The exact conditions set to identify these problems and the color coding for them are shown in *Fig.6*. Each channel is checked for any problems and a color is assigned to it. For each

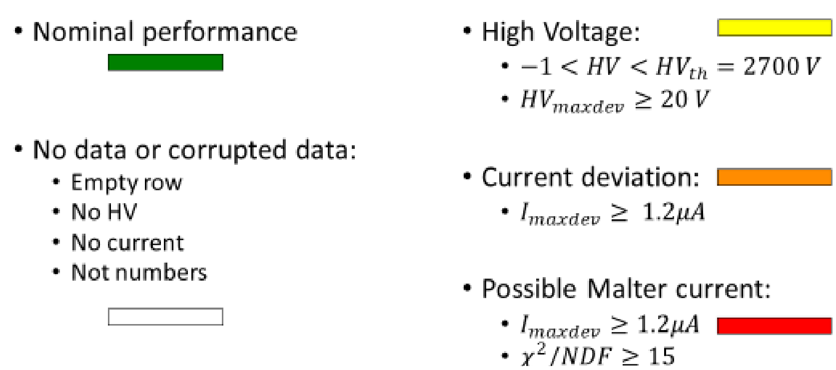

Fig.6: Problem groups and conditions for ME1/1 with their color code.

chamber a list is created with the colors of each of its channels (layers and segments). Then the highest priority color found in this list is set as the chamber color.

After the problem identification is done and colors are assigned to each channel and every chamber, the *Python* code opens the *HTML* file, corresponding to the report it has just analyzed. It adds a new row at the top of the main table with the fill information and the chamber status.

Web interface prototype

For the web interface I learned *HTML* and *CSS* and created several *HTML* pages and one *CSS* file. The *CSS* file contains the stylings for all *HTML* pages. The web pages are organized in the following way. There is a starting page where the user selects which station he wants to examine (*Fig.7*). The cells in the tables for each endcap were made to resemble the actual positions of the stations in the *CMS* detector. Each cell is a link to the corresponding page for each station. The following describes the station page for ME1/1+ as an example. It contains a small table with information on individual layers for a chamber selected by the user in the main table, a legend and the main table (*Fig.8*). The main table has the fill number, duration and peak instantaneous luminosity, and the status of each chamber for this fill. The layers table is in an *HTML iframe*,

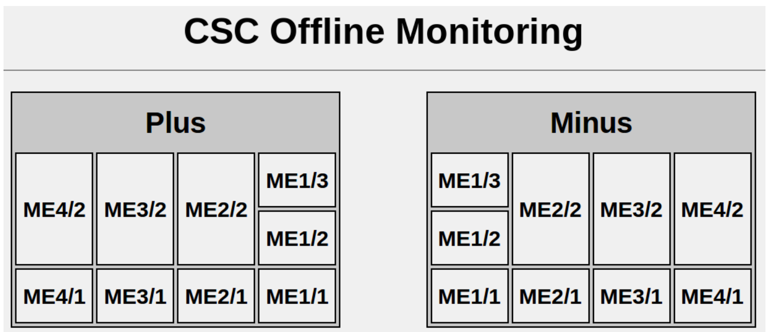

Fig.7: Starting page for the web interface

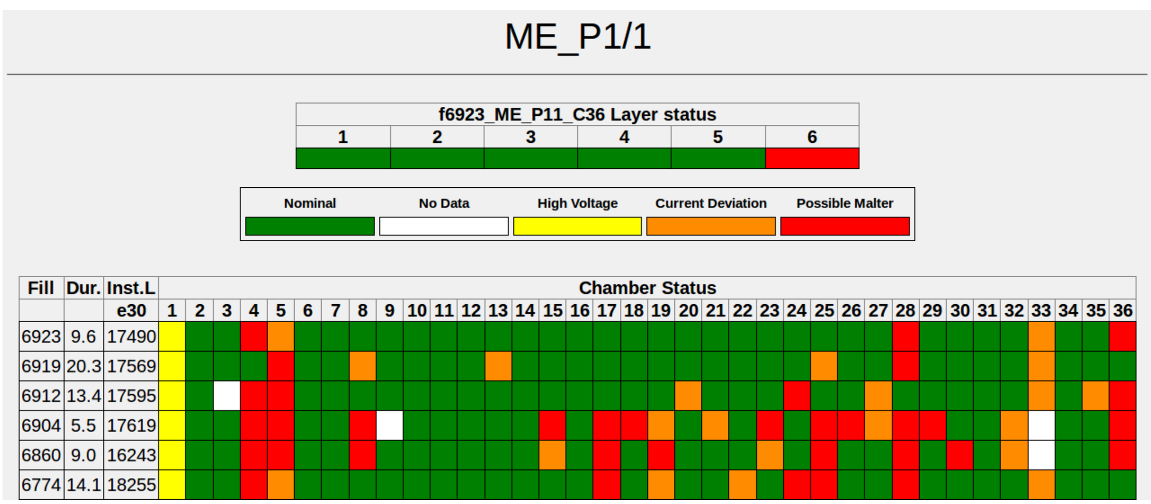

Fig.8: Final design for the main table for ME1/1+ with included table for the layer status of a selected chamber and the legend. It contains actual data for ME1/1+.

which opens the separately existing layers page on the same page as the main table. The layers table is empty at the beginning and has to be updated (using *JavaScript* or something similar) after the user selects a cell in the main table (selects a fill and a chamber). When the user has chosen a station, a fill, a chamber and a layer in the aforementioned tables, a new page opens, containing the plots for that channel. For this operation again *JavaScript* is needed.

Design for multi-segment stations

For all other stations, the layers are separated in three or five HV segments. Accordingly, the layers table on these pages have as many rows as there are segments in the station (*Fig.9*). Below is given an example with station ME2/1+ that has three segments.

When the code goes from one data file in the report list to the next it checks for which station the report is. When the script encounters a report from ME2/1, it reads the report and the next two reports. It combines them in one single file and continues with the problem identification for that file. When this is done, the script skips the next two reports, since they have already been processed. It is important to mention that the three reports for the three segments are assumed to be put in order in the report list file. This process is analogous for stations with five segments.

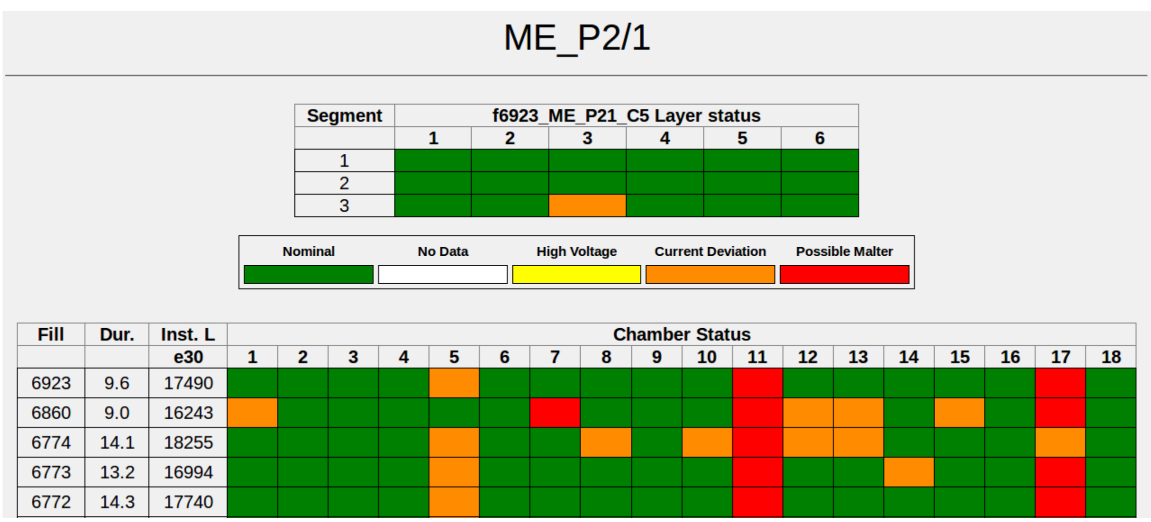

Fig.9: Final design for stations with three segments with actual data for ME2/1+

Conclusion

The web interface design is completed and all necessary *HTML* page templates are created. One *CSS* file contains the stylings for all *HTML* pages. The *Python* script properly identifies problematic channels and assigns to them the correct colors. It also updates the *HTML* pages with the new data (fills). Comments have been added to the *Python* code to make it easy to understand and use.

Two progress reports on the development were presented for discussion at weekly *CSC* meetings dedicated to longevity studies. The final project design was presented and accepted at the General *CMS CSC* meeting.

Acknowledgments

I would like to thank my supervisor, Katerina Kuznetsova, for giving me this opportunity and for her help and guidance throughout my project development. Also would like to thank the *CMS CSC* group for their valuable input and a great work atmosphere.