

University of *Ljubljana*  
Faculty of *Mathematics and Physics*



DEPARTMENT OF PHYSICS



DESY Summer Student Programme 2012

# Production cross-sections of $t\bar{t}$ at the CMS

AUTHOR:

Jože Zobec

MENTOR:

Andreas Meyer

## Abstract

Even though the study of the currently heaviest known particle – the top quark – is overshadowed by the “Higgs euphoria” it doesn’t change the fact that unexplained phenomena are already observed in that sector. Differences in charge distributions between the  $t$  and  $\bar{t}$ , and peculiar forward-backward asymmetry are only some of the interesting things that seem to be out of tune with the standard model top quark. For my project at DESY, I was searching for new physics by testing cuts to the invariant mass of the  $t\bar{t}$  pair in the dileptonic channel on the 2011 data. After that, I modified the code so it runs on **PROOF** and wrote some additional scripts to automatize the process.

Hamburg, DESY, September 9, 2012

# 1 Introduction

Top quarks at the LHC are produced from the proton beams. Due to the extremely high energies, in 90% they are produced via the gluon fusion (fig. 1).

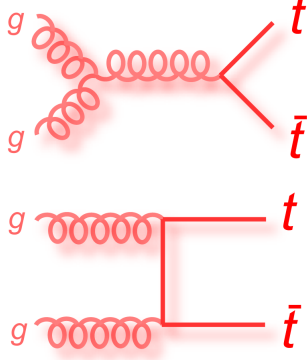


Figure 1 –  $t\bar{t}$  production via the gluon fusion.

In other 10% cases they are produced from quark the interactions (fig. 2).

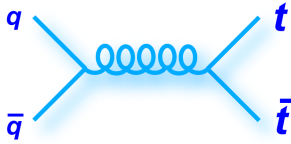


Figure 2 –  $t\bar{t}$  production via the quark fusion.

Top quarks are extremely short-lived: they don't hadronize, but decay immediately through the weak interaction:

$$\begin{aligned} t &\rightarrow W^+ + j, \\ \bar{t} &\rightarrow W^- + j, \end{aligned}$$

where  $j$  denotes a QCD-jet, coming from a quark, which is in most cases a  $b$ -jet<sup>1</sup>.

<sup>1</sup>The most probable decay. For branching ratios to other quarks please use the CKM matrix as a reference.

The  $W^+W^-$  system is still unstable, and they are short-lived and they continue decaying. Based on these decays, we have further distinctions between the channels:

- **All-hadronic** –  $WW \rightarrow jjjj$ . It has a lot of QCD background, hard to distinguish. Most probable decay channel.
- **Semileptonic** –  $W \rightarrow jj$  and the other decays  $W \rightarrow \nu + \ell$ . Less QCD background.
- **Dileptonic** – in both cases  $W \rightarrow \nu + \ell$ . Very little QCD background.

I was working on a dileptonic channel (fig. 3). This channel is further divided into three channels:  $ee$ ,  $e\mu$  and  $\mu\mu$ . The  $\tau$  leptons behave differently and their production cross-sections in their systems exhibit a different shape, so we count them as background<sup>2</sup>, which means that  $\ell$  is either an electron or a muon.

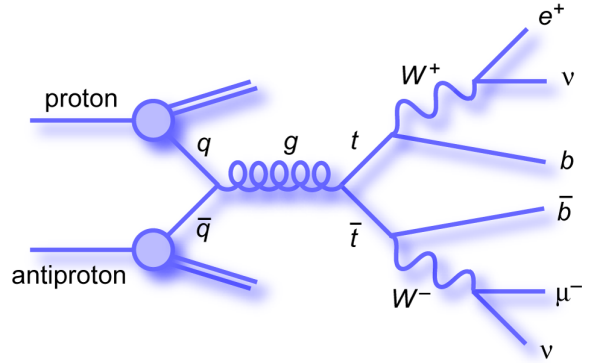


Figure 3 –  $t\bar{t}$  decay into the dileptonic (namely the ' $e\mu$ ') channel.

The dileptonic channel is clean, but has very low branching ratio (fig. 4), so we don't have much data.

Signal, characteristic for dileptonic  $t\bar{t}$  decay consists of at least two hadronic jets, two isolated leptons of opposite charge and

<sup>2</sup>The main difference is, that  $\tau$  decays fast, and can also decay hadronically.

missing  $E_T$  (MET). There must be secondary vertices involved and jets are preferably b-tagged (in 99% cases top quark decays into beauty quark).

There is an event selection process involved in accepting only the appropriate events for processing. For example, we demand that at least one of the jets is b-tagged, we apply some cuts on the  $p_T$  to get rid of the QCD background, event must be detected within the detector (obviously), so we have a cut on  $|\eta| < 2.4$ .

After this, when we are making analysis, additional cuts are applied to cut out the Drell-Yan background in the  $ee$  and the  $\mu\mu$  channels.

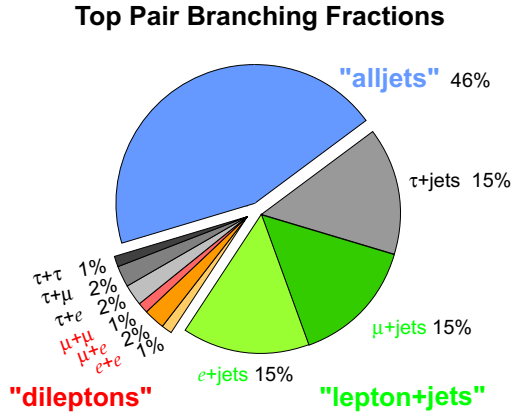


Figure 4 –  $t\bar{t}$  pair decay branching ratios.

## 2 Analysis

Since I was working on the entire 2011 data, I was processing many events. When one is searching for new physics, he must try to find it in a heap of rubbish – try to find it in the events that don't contain any new information in the academic world of physics.

This was the main motivation for applying cuts in the  $m_{t\bar{t}}$  system. By cutting the low-energetic events, we make sure that whatever we get, will belong to the currently unexplored region. After applying the cuts, I tried to compare the new plots with the control (uncut plots).

There are two types of  $m_{t\bar{t}}$  cuts that I made: at the MC generator level and at the reconstruction level (event rejection). I found out, that in order to have a self-consistent results, one must make cuts in both levels.

The algorithm also takes the detector inefficiencies into the account and does the so-called unfolding. In order for the mass cuts to be done in a correct way, the unfolding procedure must be done bin by bin.

## 3 Analysis results

In physics analysis I found some really interesting changes when I made the plots, which can (unfortunately) be explained with known physics.

As I said, I was making cuts at the invariant mass of the  $t\bar{t}$  pair ( $m_{t\bar{t}}$ ). The cuts were applied at 500 GeV, 700 GeV (fig. 6) and at 900 GeV (the last cut was really violent and took out most of the events).

In the analysis of the differential cross-section we were looking at the normalized distributions (fig. 5), so the quantity we were looking for was  $(d\sigma/dX)/\sigma$ . This is interesting for two reasons: (1) because we are looking the shape of the distribution, and (2) we get rid of the systematic errors, that are shape-dependant.

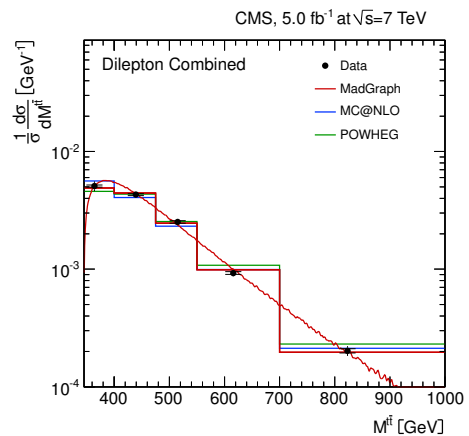


Figure 5 – Differential cross-section of the  $m_{t\bar{t}}$  without cuts.

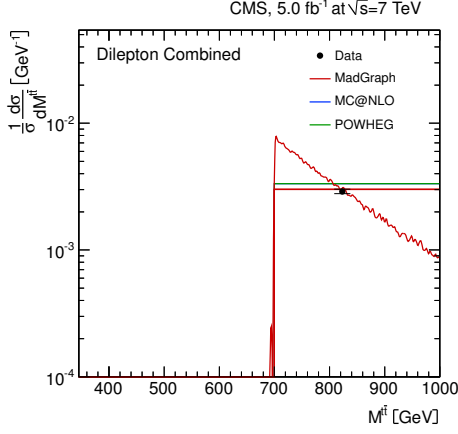


Figure 6 – Differential cross-section of the  $m_{t\bar{t}}$ , cut at 700 GeV.

Graphs that change significantly enough to be worthy of this paper are here:

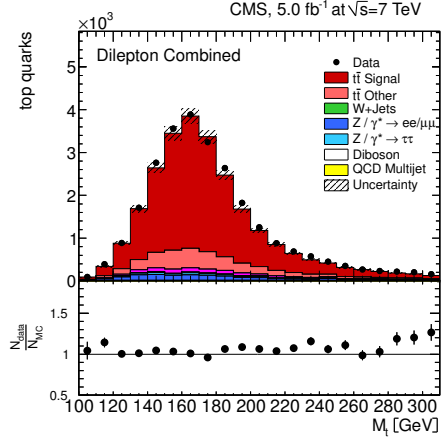


Figure 7 – Top quark mass distribution on the uncut data.

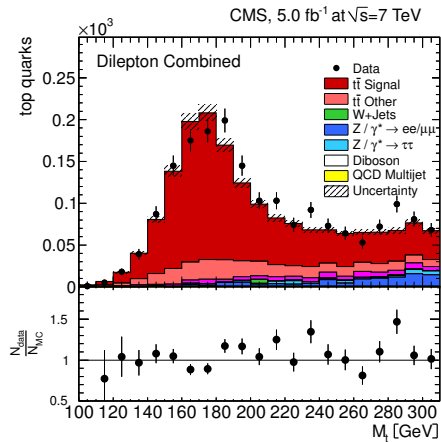


Figure 8 – Top quark mass distribution, cut at 700 GeV.

On the figures 7 and 8 with 700 GeV, we seem to have a small peak around 290 GeV. The peak is quite small and I wouldn't give it much merit. Still, the peak is higher and more pronounced in the 900 GeV cut, but we only have  $\sim 100$  events in the entire histogram, which makes it more susceptible to statistical fluctuations.

Another interesting thing is how the  $p_T$  or  $\eta$  distributions of one of the top quarks changes with rising the cuts.

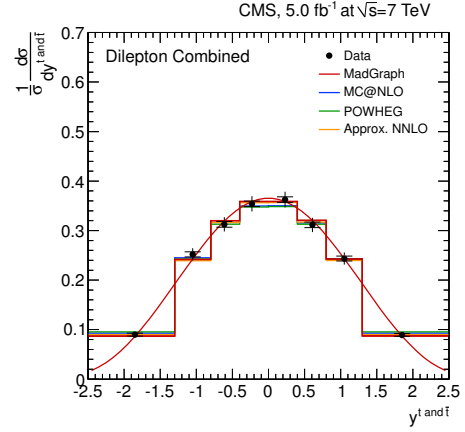


Figure 9 – Before cuts, we can see that pseudo-rapidity has a peak at 0.

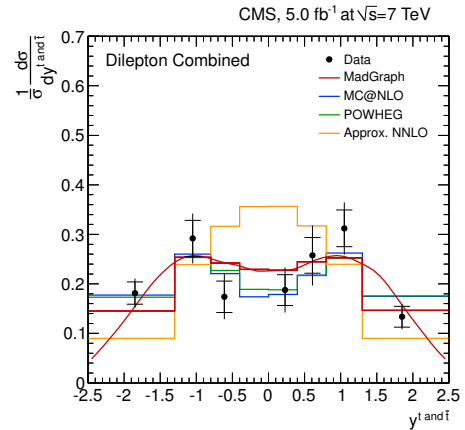
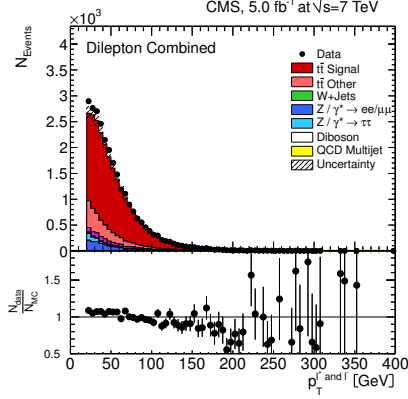
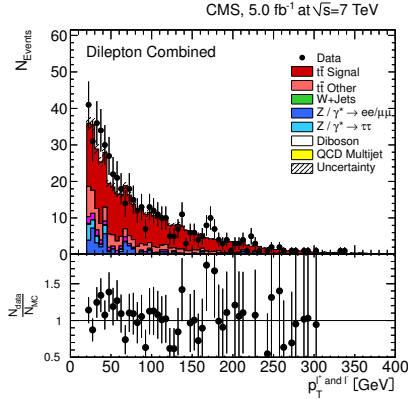


Figure 10 – Instead of one peak in the middle, we notice two bumps and a deficit, where the previous peak was. NNLO is hard-coded and does not feel the cut. The cut has been made at 900 GeV.

As we can see from figures 9 and 10, the distributions change quite a bit. But the change can be explained: we are only left with very high-energetic events and they will mostly scatter in the forward-backward direction.



(a) Uncut data –  $p_T$  of both  $\ell^\pm$ .



(b) Data with 900 GeV cut –  $p_T$  of both  $\ell^\pm$ .

Figure 11 – With my supervisor I saw an interesting peak around 170 GeV, that we couldn't put anywhere ... Probably the events come from the  $e\mu$  channel. The errors are also so high, and statistics too low to trust this plot.

On fig. 11 we can see the lepton ( $\ell$ )  $p_T$  and how it changes with cuts.

## 4 PROOF

PROOF is a part of ROOT. It is a set of classes and functions that serve for the purpose of running the processes on either single multi-core-CPU machine, or on several single or multi-core-CPU machines<sup>3</sup> – basically parallel processing. Using such division we have two respective types of PROOF: PROOF-Lite and PROOF.

Proof session is constructed like this (consult fig. 12 for further detail):

1. We (the client) open the session on the PROOF-enabled facility or on the local machine, and specify the number of slaves (workers).
2. We give commands to the master node. It gets access to the data, merges all the packages (the .root files we are going to process) together.
3. Initialize the workers. The packages are cut into separate pieces and distributed to workers. If a worker is idle, it will automatically ask for work. Fast workers get harder tasks, and slower workers get easier challenges.
4. After the work is done on each of the worker nodes, they send it to the master, which in turn merges their results together and separates the datasets like it was in the beginning.
5. Master node gives the results to the client (us).

6. We close the PROOF session.

<sup>3</sup>the so called PROOF clusters

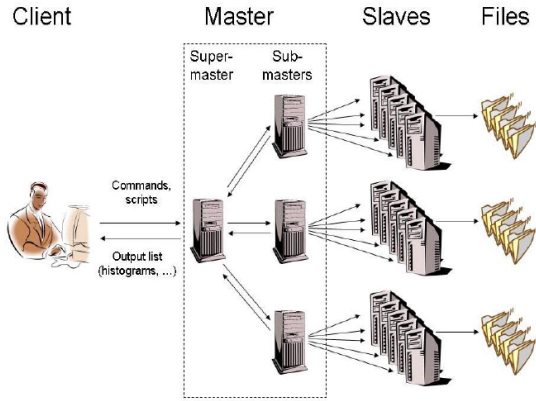


Figure 12 – A typical PROOF session.

The usage of PROOF is intimately connected to the usage of the `TSelector` class. We create a `TSelector`-inherited class and are obliged to define these specific methods in the definition:

- `virtual void Begin (TTree * tree)` – we initialize the work on the master
- `virtual void SlaveBegin (TTree * tree)` – we initialize the work on the slaves
- `virtual void Init (TTree * tree)` – this function needs to be defined *only if* we want to process `TTree *` or `TChain *` objects.
- `virtual Bool_k Process (Long64_t Entry)` – we do the processing step on the slaves
- `virtual void SlaveTerminate (void)` – close the work on the slaves
- `virtual void Terminate (void)` – close the work on the master and end the PROOF session.

For comparison I ran the analysis with and without PROOF. As I suspected, the decrease in time was linear, but still quite significant: the drop from  $\sim 3$  h to  $\sim 15$  min is

quite impressive. I used 8 CPU-cores, even though the CMS machines have 12 of them.

Table 1 – The comparison between a PROOF system and one that runs without it. I ran it on eight cores, this is where that ‘8’ comes from.

PROOF	CPU time [s]	user time [s]	real time
off	11049.21	296.23	3 : 31 : 18.92
8-on	972.52	349.27	2 : 01 : 03.89

## 5 Conclusion

The analysis results show no new physics in the CMS 2011 data. Maybe there is something and I have just overlooked it . . . . However, the analysis of the 2012 data is imminent and with the PROOF-enabled analysis code, the work will run more smoothly and will take less time. I also wrote some new scripts to help analyze the data, make the analysis versioning clearer and make sure several instances of it can be run from the same folder in parallel. I also have a suggestion, on how to further decrease the analysis user file: we merge all the `.root` files from the `mergedRoot` directory into one big `TChain *`, process them, and then use the `SlaveTerminate()` function to separate them in different files.

## 6 Acknowledgements

First of all, let me thank the entire DESY Summer Student Programme 2012 team, who really made a great deal of effort to organize everything. Thank you, Olaf, for everything.

I would like to thank my supervisor, Andreas Meyer, my colleagues, Tyler, Ivan, Carmen, Maria, Wolf and especially David, who helped me a lot with debugging my code and unlocking the secret behind the old analysis version.

Now, I would like to give thanks to my friends, people of the bunker and the “other” hostel: Dave, Jure, Omar, Mar, Khilesh, Daniel, Mikhail, Oron, Miguel, Izan, Aaron, Tom, Vlad (the Impaler), Simone... I will miss you terribly! Thank you so much for all the beer that we drank together and for all the happy moments we shared!

At last let me give thanks to Eleni Skorda! Thank you!