Performance studies of GooFit on GPUs versus RooFit on CPUs while estimating the statistical significance of a new physical signal

Adriano Di Florio [1,2] on behalf of the CMS Collaboration

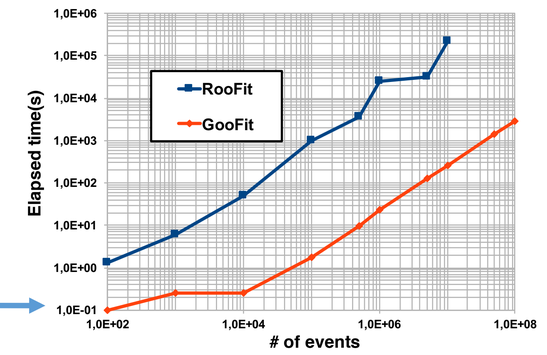
(1) INFN SEZIONE DI BARI, ITALY (2) UNIVERSITÁ DEGLI STUDI DI BARI "ALDO MORO", ITALY

Parameter estimation is a crucial part of many physics analyses & **P**robability **D**ensity **F**unction evaluation on large datasets is usually the bottleneck in the MINUIT algorithm. **GooFit**[1] is a data analysis tool for HEP that acts as an interface between the MINUIT minimization algorithm and a parallel processor (e.g. CUDA capable nVidia GPU or *multicore* CPUs via OpenMP) which allows a PDF to be evaluated in parallel. Fit parameters are estimated at each Neg-Log-Likelihood minimization step on **the host side (CPU)** while the PDF/NLL is evaluated **on the device side (GPU)** [all that until convergence] **:**

[1] **R.Andreassen et al.**, *J.Phys.:Conf.Ser.* **513 (2014) 052003 [CHEP 2013]**

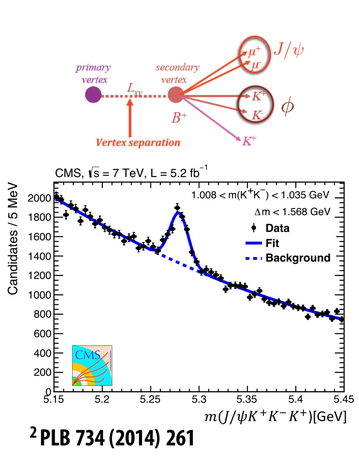
A preliminary test was done with an *Unbinned ML fit* either by using a single CPU and by using an additional GPU (an nVIDIA Tesla C2070 hosted @ Bari T2).

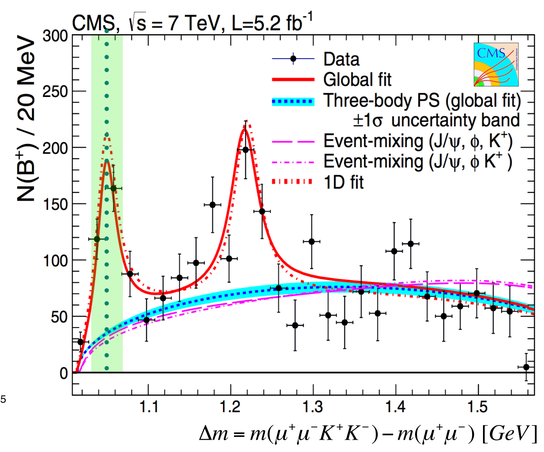
Events according to a Voigtian model (convolution is CPU-intensive) are generated & fitted. The time needed (the negligible generation time is not included) is studied as a function of the #events



For 10M events: RooFit needs 61h+23m & GooFit takes 4m+39s : speed-up ~ 750

To test the computing capabilities of GPUs with respect to CPU cores: a high-statistics toy Monte Carlo method has been implemented both in ROOT/RooFit and GooFit frameworks with the aim to estimate the (local) statistical significance of the structure observed by CMS[2] close to the kinematical boundary of the $J/\psi\phi$ invariant mass in the 3-body decay $B^+ \to J/\psi\phi K^+$ [compatible with Y(4140) by CDF]:





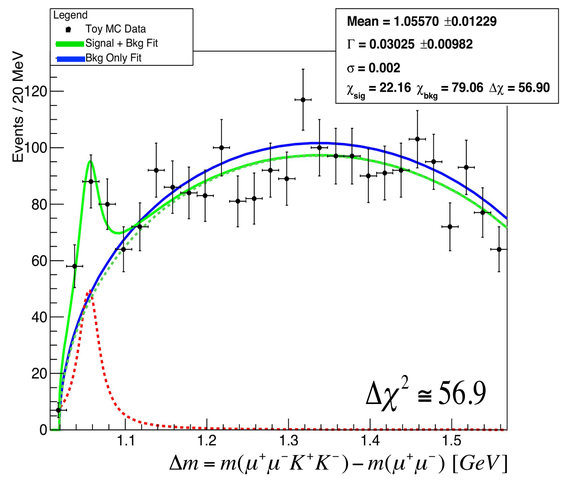
[2] **PLB 734 (2014) 261**

MC pseudo-experiments are used to estimate the probability (p-value) that background fluctuations alone would give rise to a signal as much significant as that seen in the data.

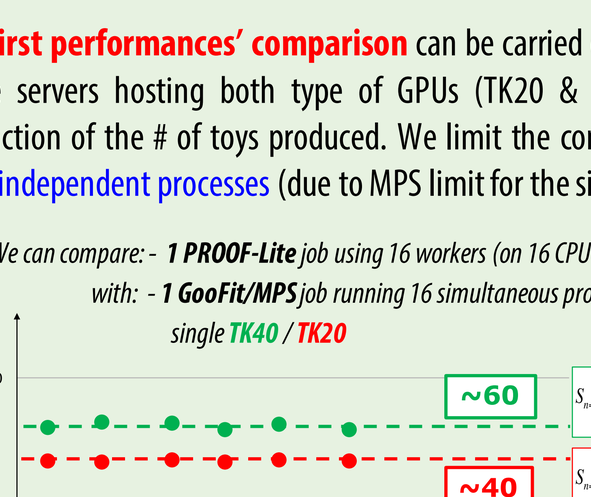
Toy MC fit cycle

1. Generation of fluctuated background binned distribution (3-body phase-space model) [total #entries fixed by data : not-extended ML fits]
2. Null Hypothesis binned ML fit performed with the PS model only
3. Alternative Hypothesis binned ML fit performed with the PS model + Voigtian PDF [truncated to correctly account for the kinematical threshold; the Gaussian resolution function has width fixed @ 2MeV]. Signal yield constrained > 0.
   - Fit performed 8 times within the known region of interest (no LEE) trying different starting values (2 masses & 4 widths).
   - For each fit calculate a $\Delta\chi^2$ w.r.t. the Null Hypothesis fit; the best $\Delta\chi^2$ fit among the 8 alternative fits is chosen
   - A $\Delta\chi^2$ (our test statistic) distribution is obtained over the sample of MC toys.

Highest $\Delta\chi^2$ generated fluctuation



GOOFIT WITH MPS SPEED-UP



$$S^{MPS}_{0<N\leq 16} = \frac{T_1}{\frac{1}{N}\sum_{n=1}^{N} T_n}$$

Hardware Setup for this study:
- 1 Server hosting 2 nVIDIA Tesla K20 +16 CPU Cores (32 w HT)
- 1 Server hosting 1 nVIDIA Tesla K40 + 20 CPU Cores (40 w HT)

http://www.recas-bari.it/index.php/en/

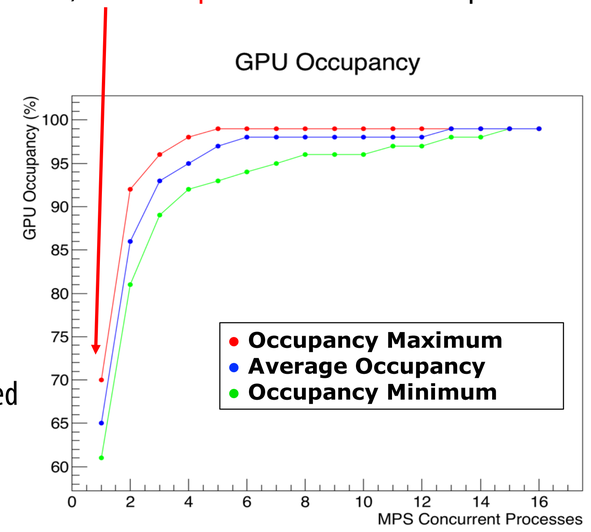
A first obtained result is simply the comparison between the MC Toys procedures running on a single GPU via GooFit and on a single CPU via RooFit. Resulting speed up:

S ~ 62 (TeslaK40) **S ~ 48 (TeslaK20)**

This kind of application (binned fit & few parameters) doesn't exploit the whole GPU computational capability.

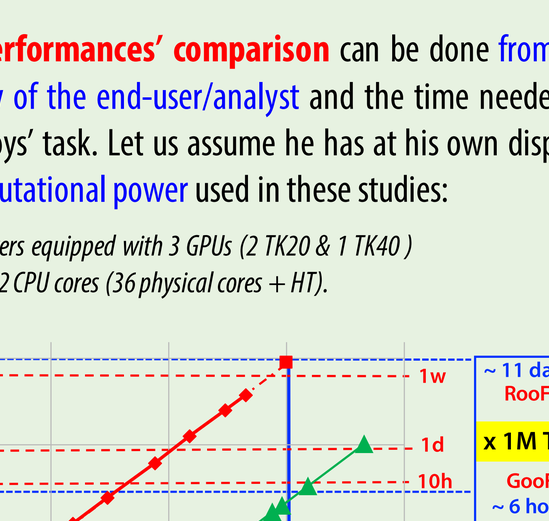
The *nVidia* **M**ulti **P**rocess **S**erver **(MPS)** is a tool developed by nVidia that allows to execute multiple processes (up to 16) on the same GPU chip. It acts as a scheduler: manages the access to memory and CUDA cores.

To efficiently run RooFit MC toys in parallel on the 72 CPUs available on the 2 servers hosting the GPUs, we use **PROOF-Lite** that is a dedicated version of PROOF[3] optimized for single multi-core machines (it has a *pull architecture*).

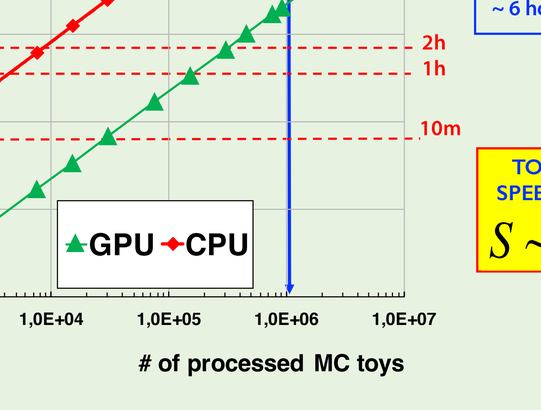


[3] G.Ganis et al., *PoS ACAT08 (2008) 007*

PROOF-Lite SPEED-UP

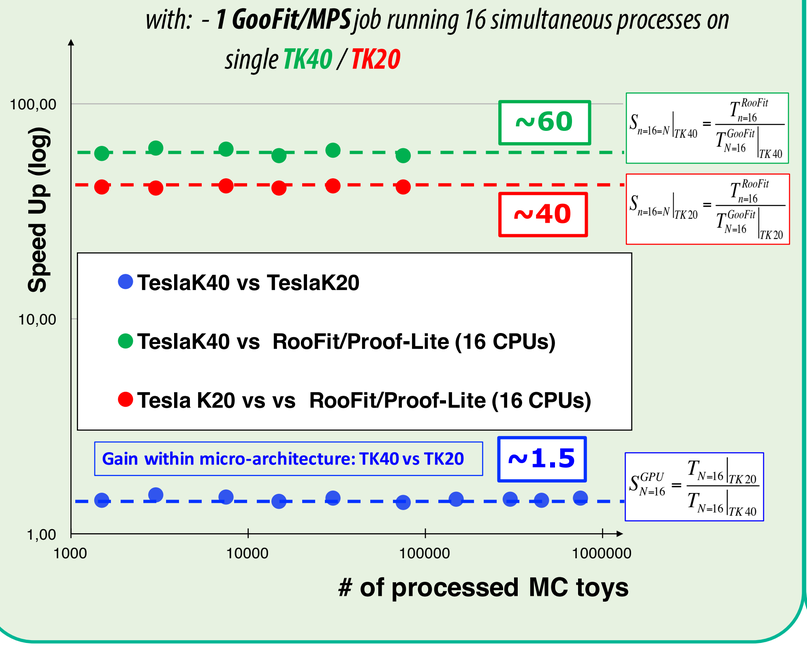


$$S^{PROOF-Lite}_{0<n\leq 32(40)} = \frac{T_1}{T_n}$$



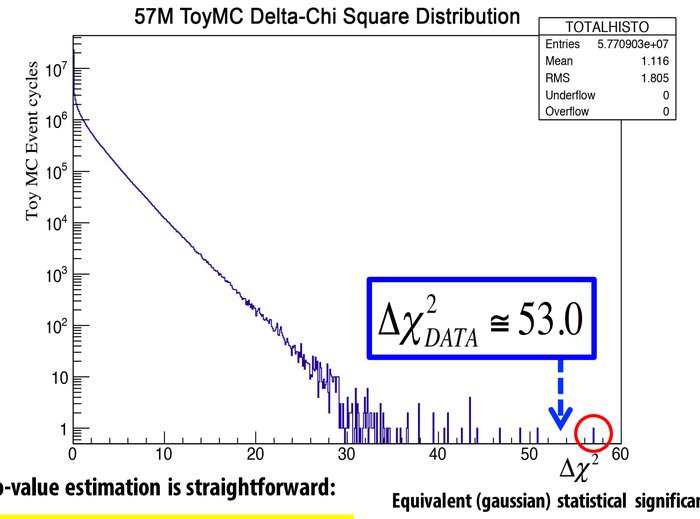
A **first performances' comparison** can be carried out on both the servers hosting both type of GPUs (TK20 & TK40) as a function of the # of toys produced. We limit the comparison to 16 independent processes (due to MPS limit for the single TK40)

*We can compare: - **1 PROOF-Lite** job using 16 workers (on 16 CPU cores)*
*with: - **1 GooFit/MPS** job running 16 simultaneous processes on single **TK40** / **TK20***



To get *a lower limit* on signal significance $>5\sigma$ a p-value $< 3\cdot 10^{-7}$ is needed, namely **at least 3.3M** toys are needed. **To estimate the actual signal significance much more toys may be needed.**

The final obtained distribution (MC toys production was stopped once a $\Delta\chi^2 > \Delta\chi^2_{DATA}$ fluctuation was found)**:**



The p-value estimation is straightforward:

$$p-value\ :\ P = \int_{\Delta\chi^2_{DATA}}^{+\infty} \Delta\chi^2 \approx \frac{1}{57.7\cdot 10^6} \cong 1.73\cdot 10^{-8}$$

Equivalent (gaussian) statistical significance:

$$Z\sigma = \Phi^{-1}(1-P)\sigma \cong 5.52\sigma$$

This result is compatible with the lower limit of 5σ for the statistical significance quoted in the CMS paper[2] on the basis of 50.5 millions of MC toys (by RooFit).

A **second performances' comparison** can be done from the point of view of the end-user/analyst and the time needed to deliver the toys' task. Let us assume he has at his own disposal the full computational power used in these studies:

2 servers equipped with 3 GPUs (2 TK20 & 1 TK40)
and 72 CPU cores (36 physical cores + HT).

The optimized GooFit applications running, by means of the MPS, on GPUs, hosted by the servers used in the presented test, provides a striking speed-up performance with respect to the RooFit application parallelized on multiple CPUs by means of PROOF-Lite.

CHEP 2016 Declination

22nd International Conference on Computing in High Energy and Nuclear Physics 2016
October 10-14, 2016 – San Francisco - CA - U.S.A.