HPC with GPUs in LHC*b* analyses

Asier Pereiro Castro† **Marcos Romero Lamas**†

†Instituto Galego de Física de Altas Enerxías (IGFAE)

Motivation

LHC experiments are increasingly collecting larger amounts of data to be analyzed. Digesting those large samples is computationally expensive and we need to use High Performance Computing techniques to reduce the timescale of the analyses. CPUs usually have $\sim$ 10 cores while GPUs have $\sim$ 1000, making them very well suited to accelerate several steps of the analysis.

Core ideas

- Minimize the amount of data transferred between host and device when possible, even if that means running kernels on the GPU with no speed-up.
- Higher bandwidth is possible between the host and the device when using page-locked (or "pinned") memory.
- Eliminate per-transfer overhead by batching many small transfers into one larger transfer.
- Overlapped H2D transfers whenever is possible properly handling kernel execution.

Theoretical model

The physical PDF is given by

$$\mathcal{P}^{\text{Phys}}(t,q,\Phi_4) \propto e^{-t/\tau}\Big[(|A(\Phi_4)|^2 + |\bar{A}(\Phi_4)|^2)\cosh\left(\frac{\Delta\Gamma}{2}t\right) - 2\Re(A^*(\Phi_4)\bar{A}(\Phi_4))\sinh\left(\frac{\Delta\Gamma}{2}t\right) + q(|A(\Phi_4)|^2 - |\bar{A}(\Phi_4)|^2)\cos\Delta m t - 2q\Im(A^*(\Phi_4)\bar{A}(\Phi_4))\sin\Delta m t\Big],$$

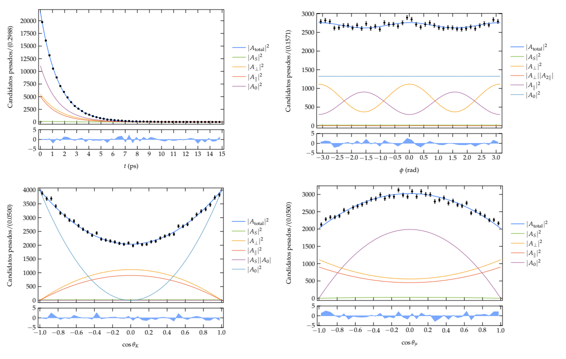
being $q=\pm1$ the flavour, t the decay time and Φ_4 the position in phase space.

The amplitude model is a superposition of quantum helicity states

$$A(\Phi_4) = \sum_k |a_k| e^{i(\delta_k + \phi_s^{q\bar{q}})} A_k(\Phi_4)$$

$B_s^0 \to J/\psi K^+K^-$ analysis

- Golden channel to measure the $\phi_s = -0.03696^{+0.00084}_{-0.00072}$ rad [2] weak oscillation phase, dominating current World Average.
- +500k events of LHC*b* data to be fitted with a 4D model: time and three helicity angles. These events are splitted by year and trigger category.
- Good decay-time resolution, to capture Δm oscillations.
- Modeling the decay-time dependence of the efficiency.
- Modeling reconstruction and selection efficiency on the three helicity angles.

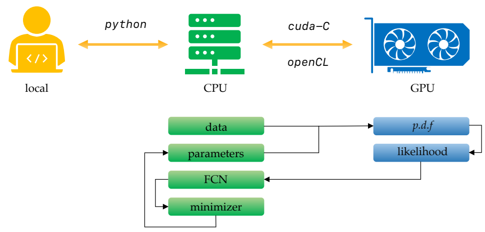


only 2016 data is shown

What is ipanema?

ipanema is a python package used as interface to control subroutines running on parallel devices [1].

- Provides a custom C99-based library expanded with commonly used HEP functions and implements complex numbers and computations.
- Has $\sim$ 15 different optimization algorithms to fit different problems.
- Classes and wrappers for parameters, confidence level estimation, fit statistics, contour plots...
- Easy to use interface to perform maximum likelihood fits.
- Compilant against p-threads in CPUs, openCL in GPUs and CUDA in nVidia GPUs.



Experimental effects

Flavour tagging introduces dilution factors. Replace

$$q \to q(1-2\omega)$$
$$1 \to (1 - q\Delta\omega)$$

We have to take into account two experimental effects

$$\mathcal{P}^{\text{Exp}}(t,q) \sim \epsilon^{\text{Ang}}(\Phi_4)\,\epsilon^{\text{Time}}(t)\int_0^\infty dt' e^{-\frac{(t-t')^2}{2\sigma^2}}\mathcal{P}^{\text{Phys}}(t',q)$$

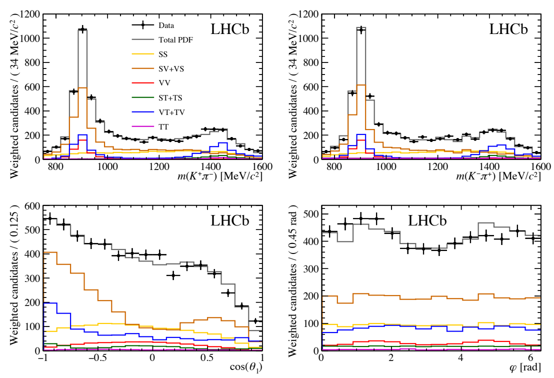
Angular acceptance: detector geometry and selection procedure induce bias on the angular variables. We need to consider this effect with an angular acceptance model.

Time acceptance: trigger selections introduce a decay time bias. A time acceptance model is needed to consider this effect.

Time resolution: the PDF has to be convoluted with a resolution model to account for the finite resolution of the detector.

$B_s^0 \to (K^+\pi^-)(K^-\pi^+)$ analysis

- About 5k events in Run 1 [3].
- Weak phase $\phi_s^{d\bar{d}}$ very sensitive to New Physics.
- Potential flavour anomaly (2.6σ) in A_0.



References

1. Ipanema-β: tools and examples for HEP analysis on GPU, D. M. Santos et al., 2017.
2. Updated results on the CKM matrix, CKMfitter group, 2019.
3. First measurement of the CP-violating phase $\phi_s^{d\bar{d}}$ in $B_s \to (K^+\pi^-)(K^-\pi^+)$ decays.