Fitting with iMinuit and zFit

2024 US Belle II Summer Workshop

Thibaud Humair, DESY, 20 June 2024

Purpose of this talk

- Give you some resources to decide which tool to use when starting to build a fit
- This is all about unbinned fit! (binned fit with HistFactory is a separate story)
- Usually, the usual go-to tool is RooFit (maybe that changed in the last 5 years)
  $\Rightarrow$ would like to share my experience with python alternatives
- I am not a super pro or a software wizard, but can share my personal impressions

Examples of software overlords:
Hans Dembinski (iMinuit)
Jonas Eschle (zFit)

RooFit pros and cons

- Really made for HEP physicists, contain all you need for most of your cases:
  - Fitting with many PDF shapes
  - Plotting tools
  - Limit setting tools
  - Convenience functions to work with complicated fits:
    - Easy to work with multi-dimensional fit
    - RooFactory, RooWorkspace makes it easy to build multiple simultaneous fits
- It runs on multiple CPUs, fast

- Sometimes a bit of a black box, hard to debug
- “monolithic library”, difficult to build upon it if what you need is not readily available
- Community of experts is not so large
- A bit out-fashioned, restricts you to ROOT, incompatible with newer cool data science tools in python

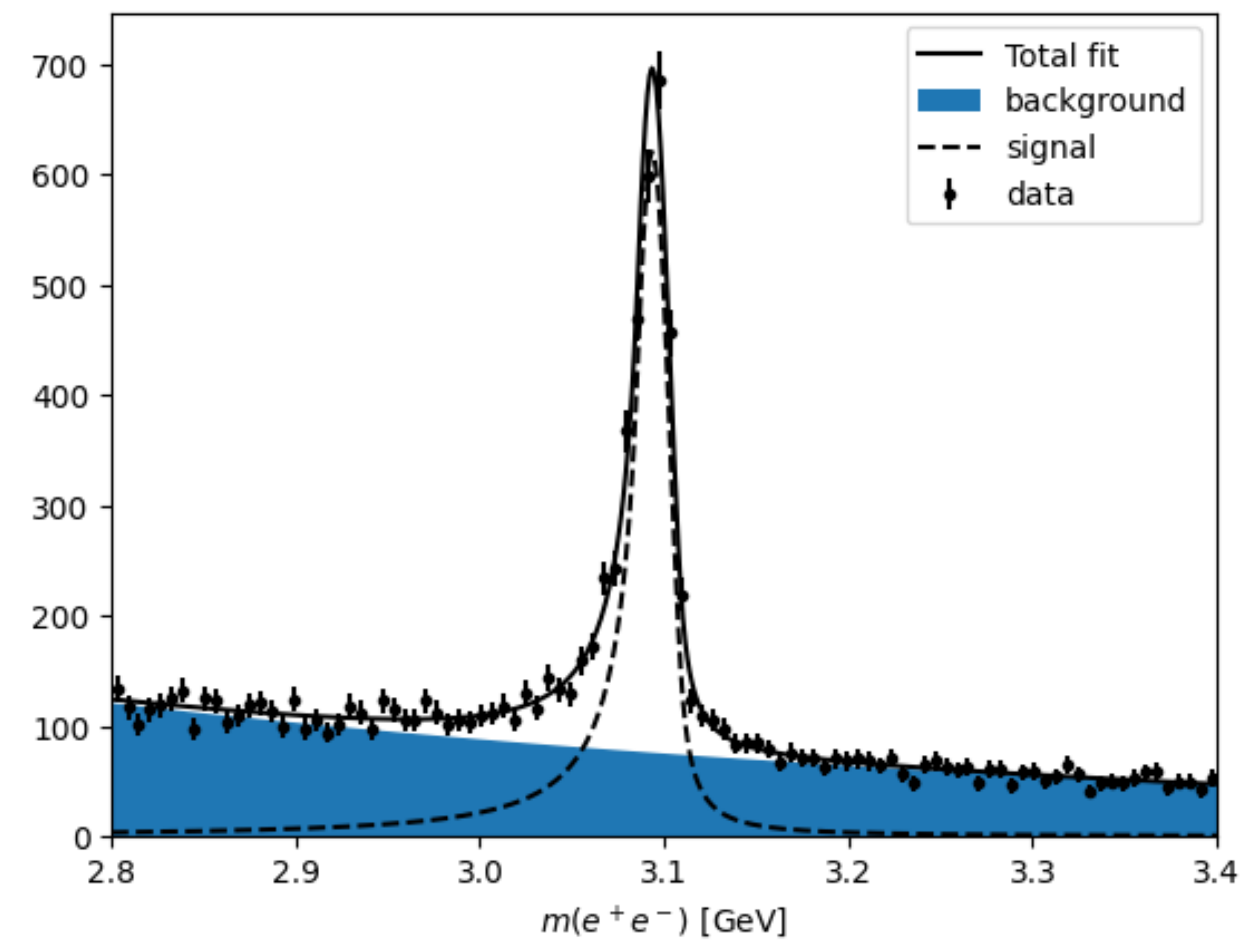
Using python to fit

Why I prefer to use python:

- Many useful python tools: Jupyter, panda dataframes, numpy… Wants the fit to be fully compatible with that
- Nice plotting library: matplotlib!
- python data structures are made to organise things easily (e.g. using dictionaries) and can be used in place of RooFactory, RooWorkspace
- Libraries in python (scipy, numpy) you can rely on to build any PDF shape easily
- Large python community, 99.9% of your questions are answered on stackoverflow.com
- But performance bottlenecks: need to be fast and memory efficient.

Resources for today

- Tutorial available on: https://gitlab.desy.de/thumair/py_fit_tuto
- Perform a J/ψ mass fit using two python libraries:
  - zFit
  - Pure numpy fit, then boosted with numba
- I have experience making fits very fast with both with multiple CPUs. I am less experienced with GPUs.



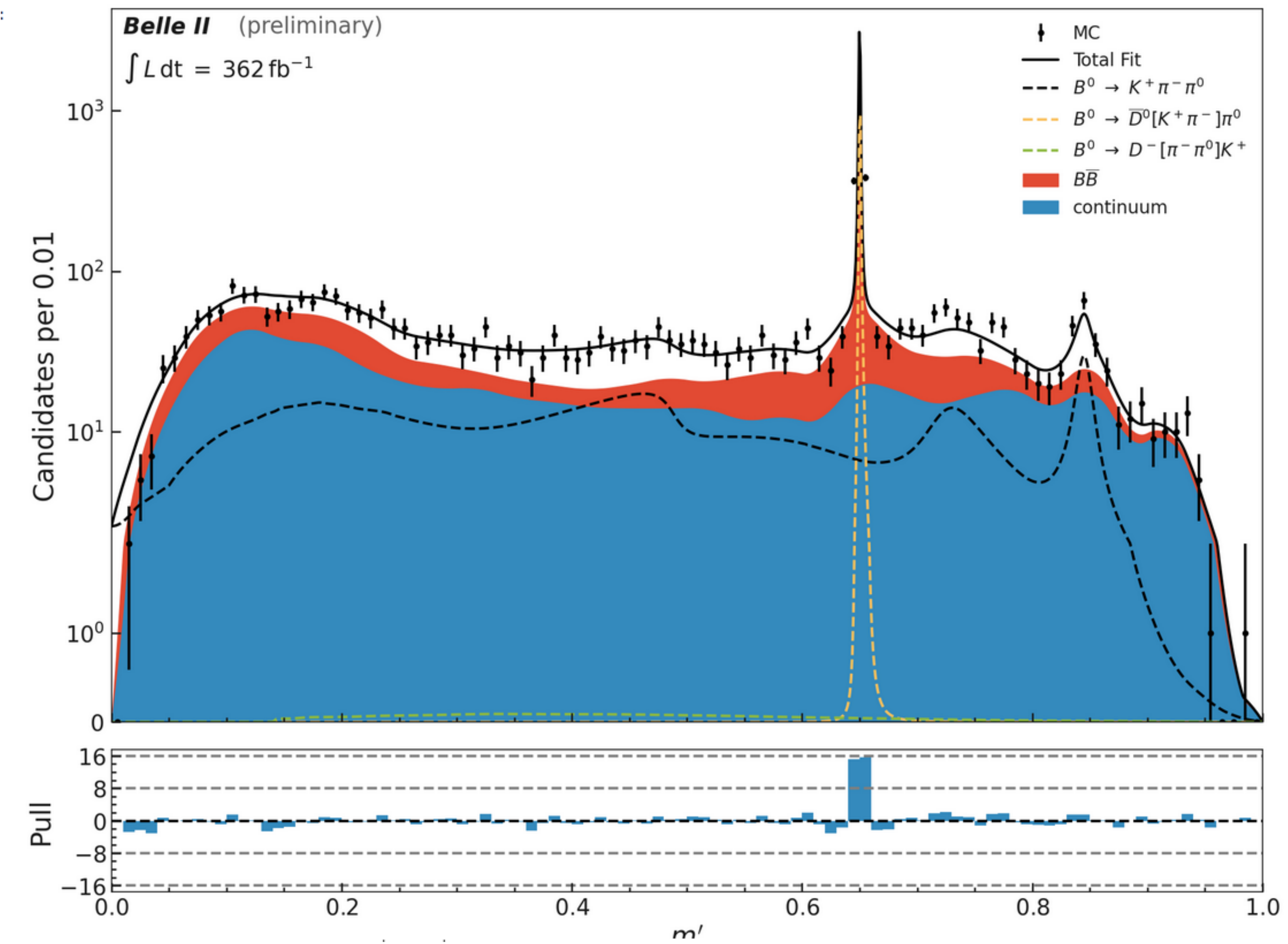
zFit

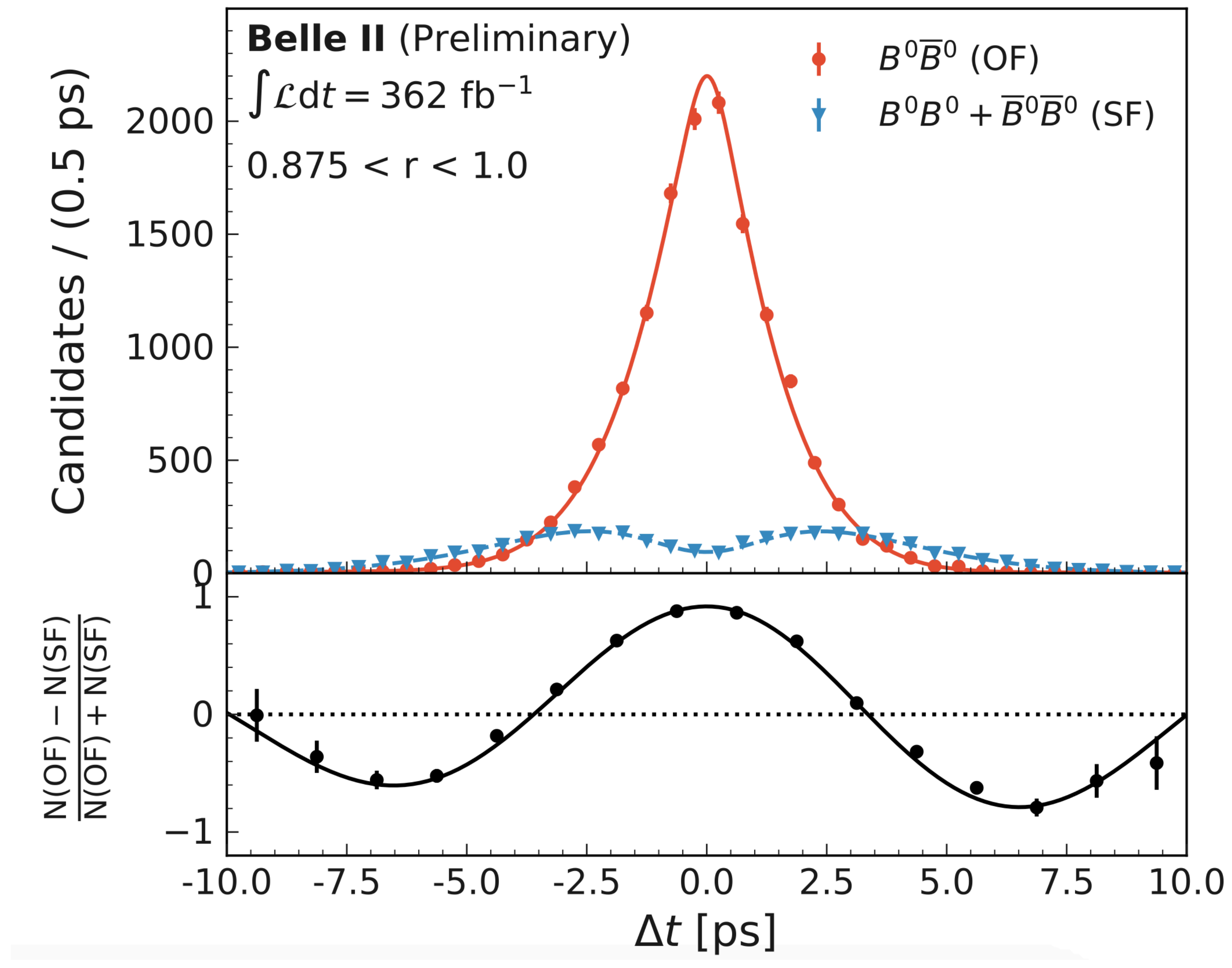
- zFit relies on TensorFlow, which uses its own optimized dataframes and mathematical functions

- TensorFlow compiles so-called graphs that can then run very fast on multiple CPUs and GPUs

- zFit is developed as a pythonic alternative to RooFit, and is a pure fitting library (no plotting, no RooWorkspace)

- Some complex Belle II analyses fully written in zFit:
  - https://gitlab.desy.de/otittel/btokspipi0
  - https://gitlab.desy.de/belle2/physics/tdcpv/software/zfit_tdbelleii





zFit disadvantages and bottlenecks

- **Ultra memory inefficient:** might become a bottleneck if you run very complicated fits!
- Still young and buggy, with a small community
- The monolithic problem is not fully resolved: building your own PDF using tfp-stats, tfp-math, or wrapping scipy functions might be tricky
- tensorflow is made for machine learning and use its own type of data structures, you might want something more "pure" numpy

Pure numpy boosted with numba

You can build your fit purely from numpy! You will be fully free, fully python, but there is some overhead as you build things from scratch!

- The one necessary tool is **iminuit**, to perform minimization
- numba can help you drastically boost your fit, based on just-in-time compilation.
- You need to build your likelihood PDF from scratch, but using:
  - numpy built-in functions, scipy.stats, scipy.special and numba equivalent: numba-scipy and numba-stats
- Fully compatible with numpy data structures. No memory issues observed.
- Example of my experimentations at Belle II: tatanumba
  (in particular, see tatanumba/functionpieces to see how to make things ultra fast with numba)

Thoughts

- I have been happy developing my fit with numba:
    - Full control, no black box
    - Pleasant to work with python and modern tools
- In practice for you, this might not be the best solution:
    - For a simple enough fit (e.g., you measure a branching fraction and just need a 1D or 2D fit): zFit or RooFit are good!
    - You might take over a framework from previous PhD student/Postdoc. Most of the time, it is just best to take over something that has proven itself to work.