INNOWACYJNA GOSPODARKA
NARODOWA STRATEGIA SPÓJNOŚCI

Dotacje na innowacje
Inwestujemy w waszą przyszłość

UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ
ROZWOJU REGIONALNEGO

FNP
Fundacja na rzecz Nauki Polskiej

Confronting theoretical predictions with experimental data

Fitting strategy for multi-dimensional distributions

I. M. Nugent, T. Przedziński (speaker), P. Roig,
O. Shekhovtsova, Z. Wąs, and J. Zaremba

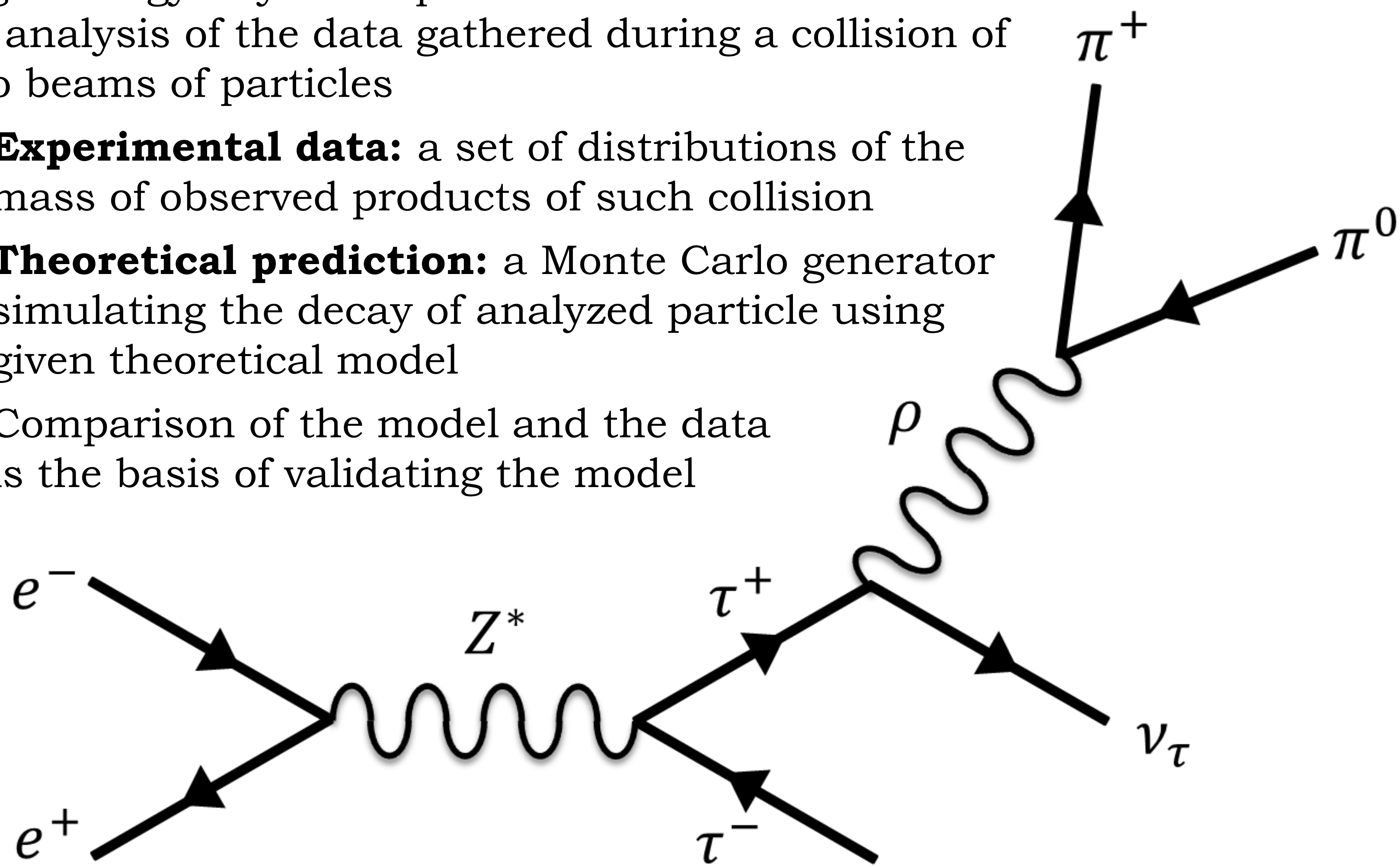
Plan:

- Introduction
  - Description of a problem
  - Difficulties
- First attempt
  - Simplified, linearized model
  - Drawbacks
  - Results
- Change of approach
  - Improving setup and theoretical model
  - Optimizing CPU-intensive algorithms
  - Results
  - Verifying results
- Outlook

Description of a problem

High Energy Physics experiment under discussion: an analysis of the data gathered during a collision of two beams of particles

- **Experimental data:** a set of distributions of the mass of observed products of such collision
- **Theoretical prediction:** a Monte Carlo generator simulating the decay of analyzed particle using given theoretical model
- Comparison of the model and the data is the basis of validating the model



Description of a problem

Goal of the analysis: validation of the new theoretical model

- first step is to adjust the parameters of the model. This is done by fitting a set of parameters, within their restricted range given by the model, to the data
- results of these fits will show the potential of the model to represent the data

Our focus is on τ lepton decay. We start from the most significant non-trivial case, which is $\boldsymbol{\tau^- \to \pi^-\pi^-\pi^+}$ decay.

- **Data:** three histograms representing distributions of mass couplings: $\boldsymbol{\pi^-\pi^-}$, $\boldsymbol{\pi^-\pi^+}$ and $\boldsymbol{\pi^-\pi^-\pi^+}$
- **Theoretical model:** Resonance Chiral Lagrangian Theory (abbreviated here as RChL). So far not used to describe τ decays

Problems

First and most obvious problem: we have three one-dimensional histograms as data while the theory gives 7-15 parameters (depending on effects taken into account) to describe the model

- multiple minima
- potentially large correlations between parameters

Limited a-priori knowledge of the fitted distributions

- at any point in parameter space of the model there might be no 1st derivative; we cannot expect the methods we use will converge
- there might be infinite number of minima; finding a global minimum might become a problem
- the model might be prone to rapid changes and fluctuations; stability of the result must be verified

Lengthy calculation of a single data point

- requires long MC simulation; about 3.5 hours on 1-core machine for sample of 20 million events (large sample size needed to reduce statistical error)

First Attempt

First Attempt
Starting point

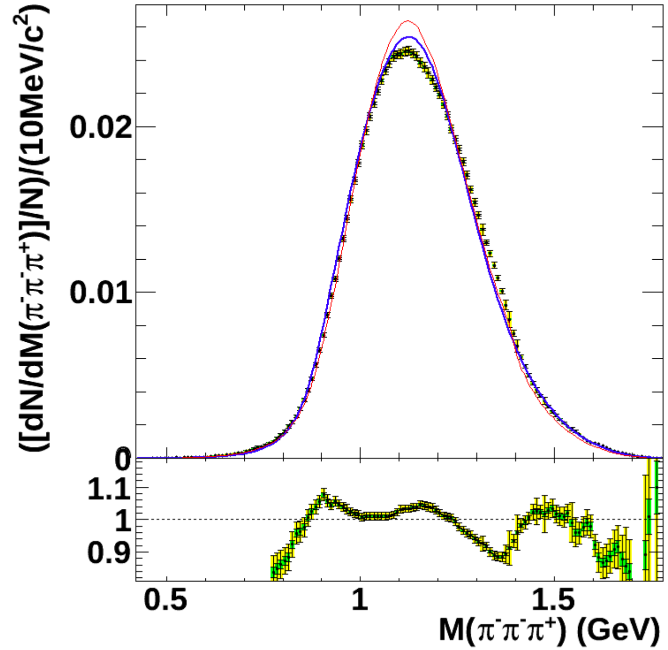
Comparison of the starting point (chosen based on educated guess) to the data.

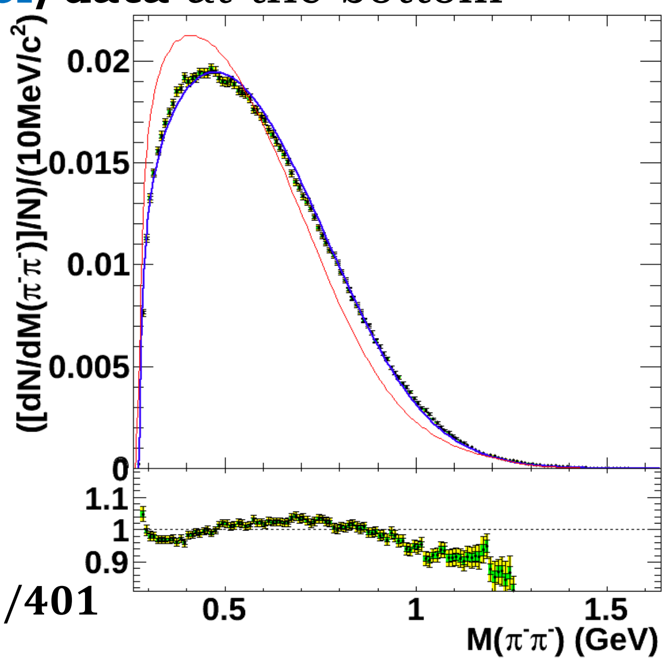
Data points: data from BaBar experiment

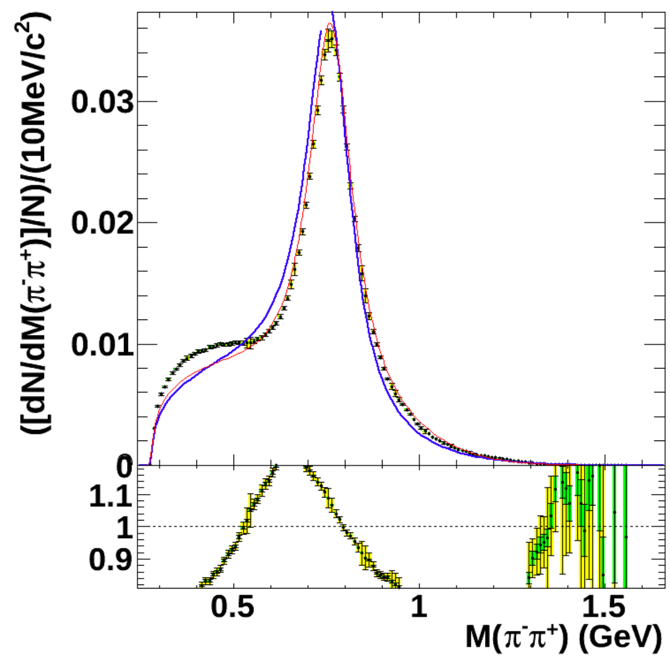
Red line: CLEO model used since 1995 (updated in 2005)

Blue line: new RChL model

Ratio **new model/data** at the bottom







$\chi^2/ndf = 183\,532/401$

First Attempt
Linearized model

We based our method on `MINUIT` algorithm (available through `ROOT` data analysis framework). Fit function is defined as:

$$F^{fit}(x) = H_0(x) + \frac{P_1 H_1(x)}{\Delta x_1} + \frac{P_2 H_2(x)}{\Delta x_2} + \dots + \frac{P_{15} H_{15}(x)}{\Delta x_{15}}$$

Where:

$P_{1\dots15}$	fit parameters,
$H_0(x)$	data sample for central value of parameters,
$H_{1\dots15}(x)$	data sample for each parameter, where value of this parameter is changed by $\Delta x_{1\dots15}$.

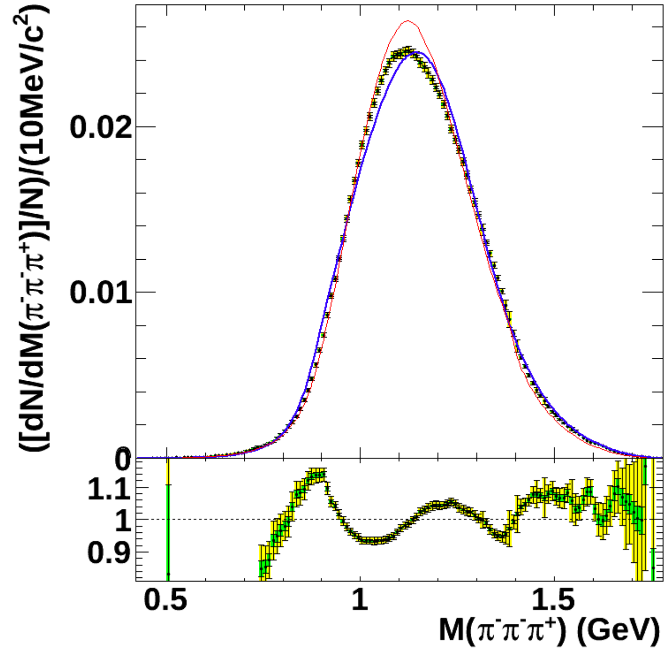
- Fitting this linear approximation of our model to the data produces a new set of values for our parameters. These values are used in the next step of the fitting procedure.
- This method yielded relatively good preliminary results in just few steps.

First Attempt
Drawbacks

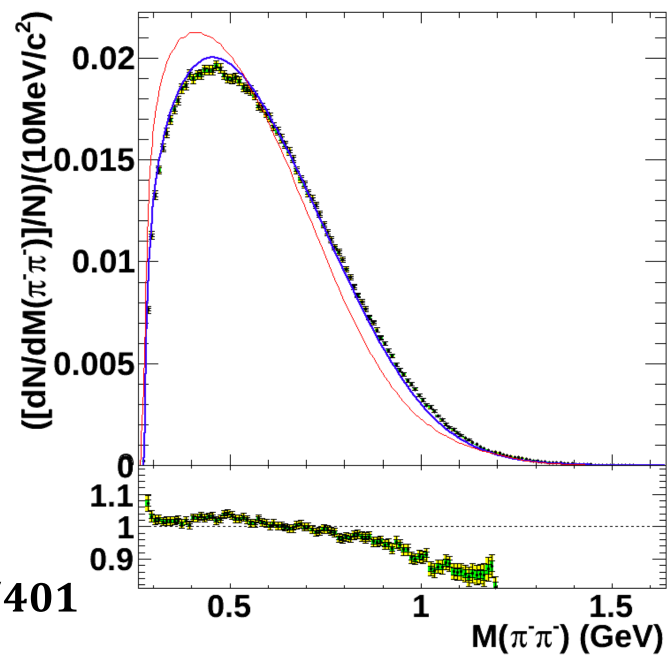
- Problems with linear approximation
  - needs additional 15 samples per step to estimate range in which linear approximation is valid
  - unstable when close to minimum. Slow gradient descent algorithm used to improve the result
  - preliminary results can be obtained after about 10-20 steps but improving the result takes another 40-80 steps
- Very slow setup
  - full Monte Carlo simulation needed for each data point with minimum 10 million events
  - we used 20 million events to reduce statistical fluctuations. This took about 3.5 hours for each core
  - ultimately, this setup takes about two weeks to produce one result

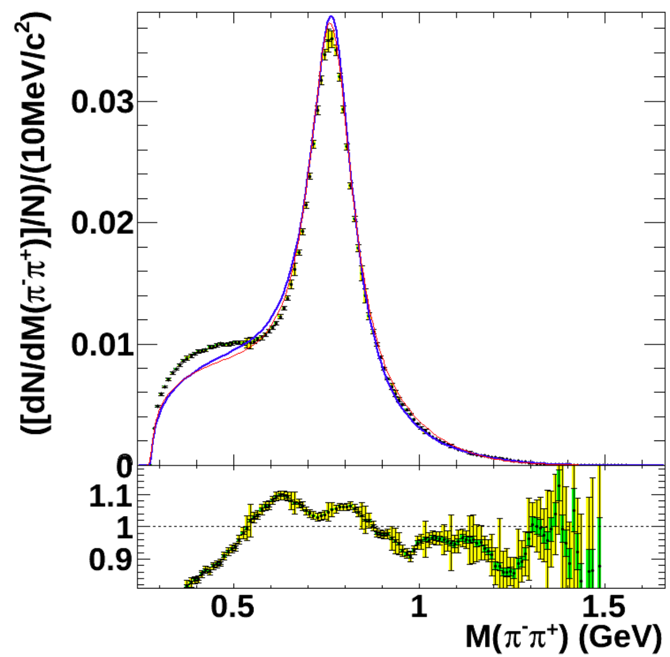
Note: *since it was our first approach, our setup was not well parallelized. We were using at most only 31 cores*

First method
Results



Best fit results using first method





$$\chi^2/ndf = 55\,264/401$$

Change of approach

Change of approach
Improving setup and theoretical model

- First results showed that theoretical model needs improvements
  - in search for further improvements, influence of coulomb interactions have been measured
  - RChL model has been extended by sigma resonance to better describe low-mass region of the distributions
- Improving the fitting algorithm required new approach
  - the first method gave as valuable information and experience
  - it still has many uses where many other methods cannot be used, e.g. when applying experimental cuts to generated sample
  - however, to optimize fitting performance, we had to drop this method
- Through the work of our theorists, a set of three semi-analytic distributions have been prepared
  - no need to generate 20 million events for each data point!
  - next performance bottleneck was integration over 3-dimensional function

Change of approach

Optimizing CPU-intensive algorithms

Optimizing integration

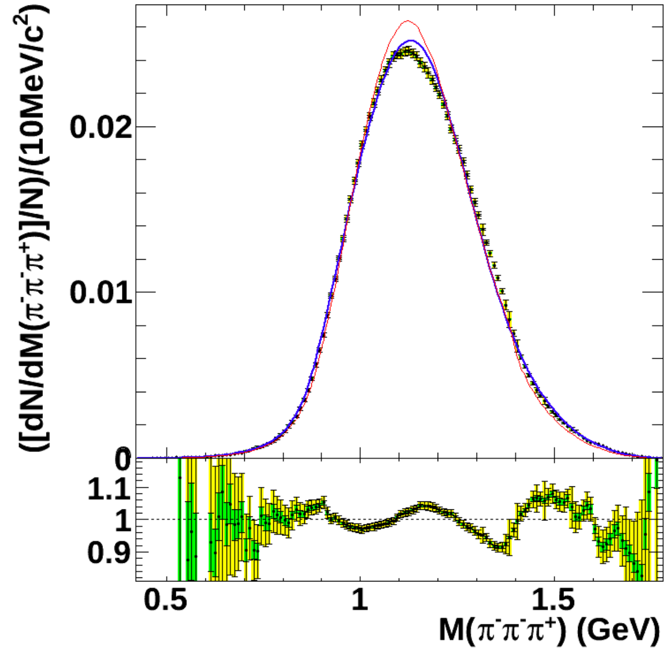
- fast 16-point Gaussian quadrature integral was used with two options: adaptable number of divisions of the integration domain (fulfilling precision requirement) and constant number of divisions
- our tests showed that though second options is simpler it is better suited for our purposes
- standard method of changing the integration variable was used to smoothen the integrand and improve convergence
- parallelized integration over whole domain

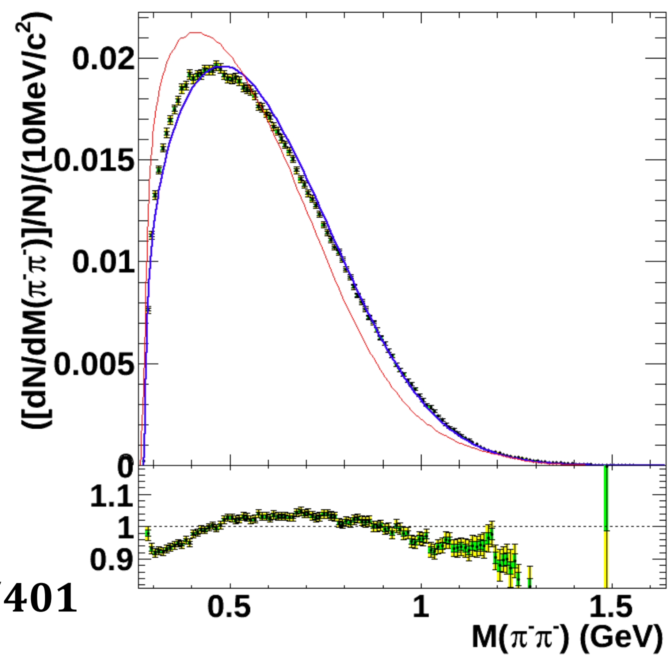
Optimizing CPU-intensive computation

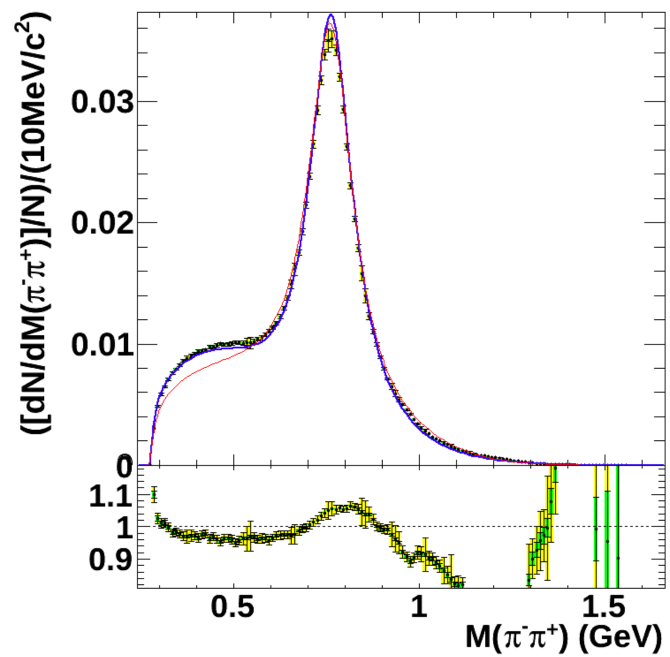
- old method from year 1992 was used to approximate the a_1 width: it's value is calculated only at several values of q_2 and predefined polynomial extrapolation function is used
- both this approximation and full a_1 width calculation was parallelized
- an option to calculate function only at the central value of the data histogram bin, as an approximation for the value for the whole bin width, was added

Change of approach
Starting point

Starting point for our improved method (including improved theoretical model)



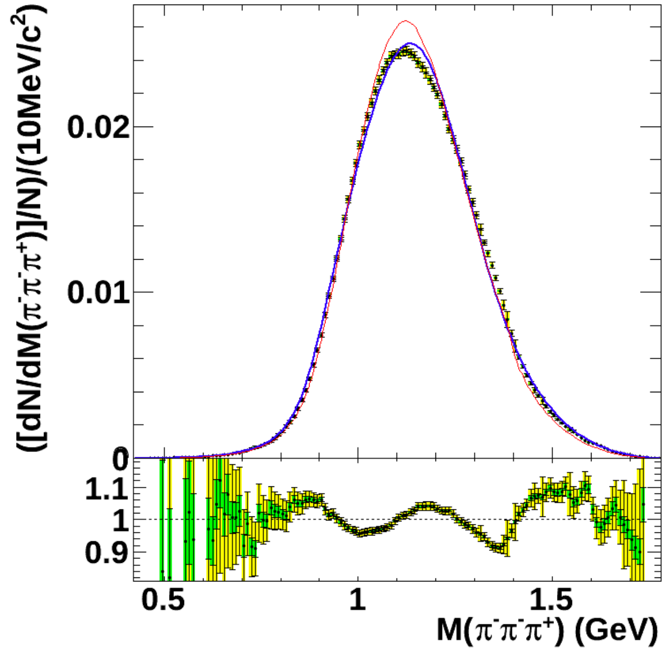


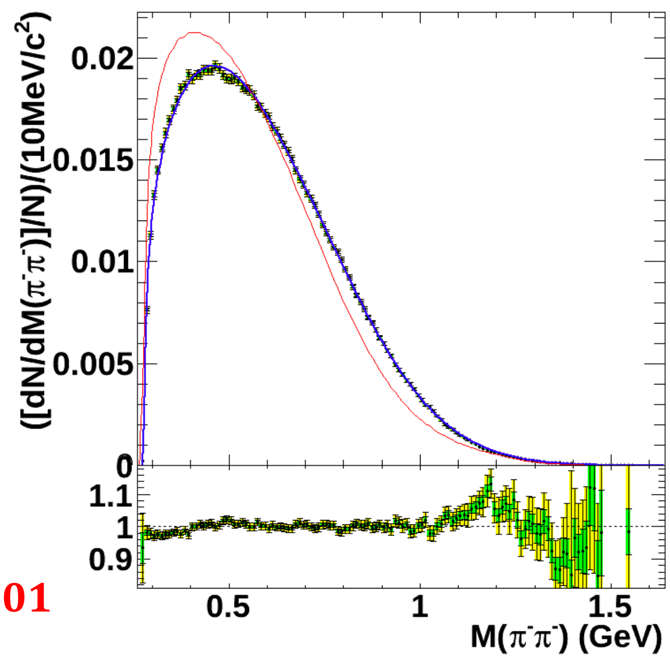


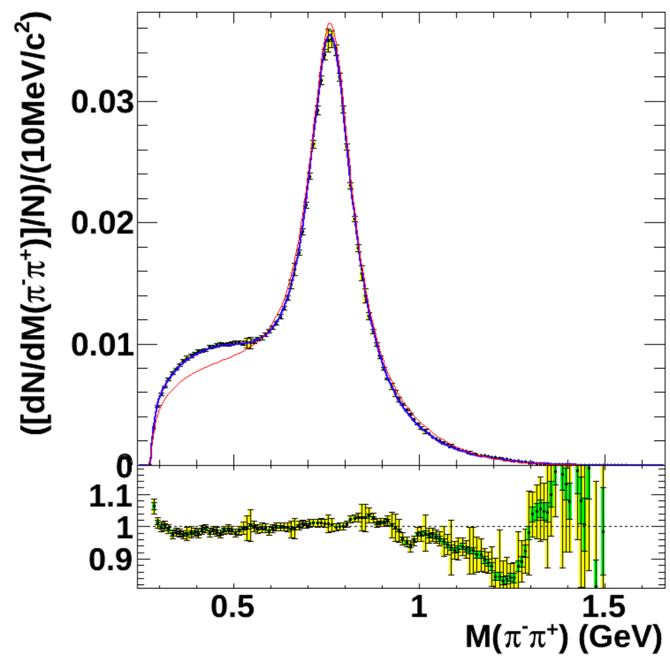
$$\chi^2/ndf = 32\,413/401$$

Change of approach
Results

Final results, after improvements and with highest available precision of calculation







$\chi^2/ndf = 6\,658/401$

Change of approach
Verifying results

- Global minimum validation
  - random scan of parameter space - first step in validation if current minimum is a global minimum and in search for potential other minima
  - approximations turn out to be very useful yielding 12 000 data points per hour using 240 cores
  - minimization from selected starting points to check if they converge to a single minimum, multiple minima or find better minimum
- Calculating statistical uncertainties
  - using `HESSE` algorithm from `MINUIT` package to calculate precise errors and parameter correlation matrix; 1 day using 64 cores
- Studies of systematic uncertainties
  - using 100 toy Monte Carlo samples generated from data (under Gaussian assumption of uncertainties)
  - analyzing how fits to these samples affects the minimum
  - up to 320 cores used (we have found it to be an optimal number in terms of how long our tasks stay in queue on Cyfronet ☺)

Outlook

- Improvements to the fitting framework reduced computation time drastically
  - from two weeks to 2-3 days for full fitting procedure
  - from 3.5h down to a minute for a single data point on 8-core machine
  - linear cores/time relation up to 24 cores
- Approximations as an effective way to reduce computing time
  - preliminary results available in less than a day
  - with all approximations in place, single data point calculated in 8 seconds on 8-core machine
  - the χ^2 for the final result ~10% smaller than for preliminary result
- Additional analysis and verification done in reasonable time
  - flexible assignment of cores - easier to optimize resources use vs time
  - in future, this fitting framework will be used for fits to 2-dimensional data as well as fits to other τ decay channels

Note: *details about physics results can be found in ref. [1]*

Thank you for listening!