PROCEEDINGS OF SCIENCE

A ROOT Package for Resonance Studies in ALICE

A. Badalà

INFN Catania

R. Barbera

University and INFN Catania

F. Blanco

University and INFN Catania

P. La Rocca

University and INFN Catania

G.S. Pappalardo

INFN Catania

C. Petta

University and INFN Catania

A. Pulvirenti[*]
University and INFN Catania
E-mail: `alberto.pulvirenti@ct.infn.it`

F. Riggi

University and INFN Catania

R. Vernet

INFN Catania

ALICE is the LHC experiment specifically aimed at studying the hot and dense nuclear matter produced in Pb-Pb collisions at 5.5 ATeV, in order to investigate the properties of the Quark-Gluon Plasma. Among the physics topics of interest in this experiment, investigation of resonances plays a fundamental role, since it allows to probe chiral symmetry restoration and to estimate the lifetime of the fireball. In the ALICE official analysis and simulation framework, a complete package devoted to this topic has been developed and optimized for an efficient management of a huge amount of data. The package has also been designed in order to be able to deal with all the distributed analysis environments available for the ALICE collaboration and has been integrated in the general correction framework, under development within the collaboration itself. Details and results of this package will be illustrated.

XII International Workshop on Advanced Computing and Analysis Techniques in Physics Research
November 3-7 2008
Erice (Sicily), Italy

1. Introduction

Heavy-ion collisions at ultrarelativistic energies investigate the properties of strong interacting matter at high density and temperature. In such extreme conditions, lattice QCD calculations allow one to expect the formation of a Quark-Gluon Plasma (QGP), which can be probed experimentally through "signatures" which should affect the spectra of produced particles. In such scenario, short-lived resonances allow one to investigate the chiral symmetry restoration expected in QGP, since it should cause modifications in their masses and widths [1].

Even in absence of a phase transition, resonances are an important probe to investigate the collective behaviour of the nuclear matter in the fireball [2, 3]. As the partonic system created in the initial stage of the collision expands and cools down, its hadronization process should pass through two fundamental steps: first, a *chemical freeze-out*, where all particle abundancies are fixed and inelastic interactions cease; then, a *kinetic* (or thermal) *freeze-out*, where all hadronic interactions stop: after this point, particles free-stream towards the detectors. Since a resonance's lifetime is comparable to that of the fireball, its decay products are likely to appear inside the fireball itself, and this may cause them to be *rescattered* by the surrounding medium: this effect prevents the mother from being reconstructed through its invariant mass, thus acting as a suppression to the measured yield. On the other hand, semi-elastic interactions can take place between particles of the same species of that resonance's daughters, and this can *regenerate* the resonance, resulting in an increase of the measured yield. Thus, the study of resonances can probe the time evolution of the source from chemical to kinetic freeze-out and test different hadronization scenarios [4]. Results on resonances detected by their hadronic decay have been reported from the NA49 experiment at SPS energies and from STAR experiment at RHIC energies. For a recent review, see [5].

Resonance studies play an important role in the ALICE physics analysis [6, 7]. In fact, ALICE is the LHC experiment precisely aimed at studying heavy ion collisions (Pb-Pb at 5.5 TeV) and investigating the QGP. Like other analysis topics related to the collective phenomena of the fireball (flow, HBT, etc.), this is one of the issues of the "soft physics" analysis programme. In order to approach it, a full software package has been developed and included in `AliRoot` [6, 8], the ALICE official simulation, reconstruction and analysis framework.

In this document, the design of this package and its performances will be illustrated. Section 2 will give a brief overview of the available computing environments for ALICE together with the general structure of the framework. In Section 3, the resonance analysis will be presented, and few examples of its output will be shown. Finally, in Section 4 some conclusions will be drawn.

2. ALICE computing scenario

The ALICE collaboration has developed a computing framework, named `AliRoot` [6, 8], based on the architecture of ROOT [9], the new generation C++ multi-purpose physics analysis framework developed at CERN. It contains a detailed simulation of all detectors and their response, the whole reconstruction chain and a complete analysis framework. This allows to make very realistic simulations, which are crucial for an optimal tuning of reconstruction and analysis strategies, and to reconstruct the real data.

---

[*] Speaker.

An important issue for LHC experiments is the amount of collected data. Both from the point of view of the required CPU and the one of the storage space, this problem cannot be faced efficiently by a unique computing center. The solution comes thanks to the GRID infrastructure, which allows to distribute both the CPU and storage space among several computing centers which participate to this project, under the coordination of CERN. They are organized into a hierarchical structure whose uppermost level (Tier 0) is at CERN, and a tree of lower level points (Tier 1, 2 and 3) are distributed worldwide: each Tier accomplishes some specific tasks concerning data storage and some steps in the reconstruction and data processing chains. The ALICE Collaboration has developed a transparent interface to the GRID named `AliEn` [10], which is able to connect to the different GRID infrastructures available to the collaboration members, providing a unique interface to all of them: this eases to the greatest degree the approach of each user to this very complex environment.

Moreover, for a preliminary tuning and monitoring of data coming from experiments, and for a fast pre-processing of analysis tasks for parameter optimizing, a small PROOF [11] cluster is available at the CERN Analysis Facility (CAF) [12]. It will be accessed by few users which will run there analysis tests to tune them before running on the full data samples available in the `AliEn` catalogue.

A fundamental issue involving the whole ALICE collaboration is the data analysis. A huge effort has been done to develop a common analysis framework [13], which provides some base C++ classes to access data in a way that is almost completely transparent to the environment where the analysis runs (local, CAF, `AliEn`), in order to allow people to develop a unique analysis task which can be run in every available infrastructure.

The resonance analysis package is based on this model, and provides all the specific features which are required for an optimal approach to this specific issue.

3. The resonance analysis package

3.1 Package structure

Due to their extremely short lifetime (few fm/c), resonances cannot be observed directly. The unique way to reconstruct one of them from data is through the correlations of its daughters, in order to build an invariant mass distribution where its signal appears as a Breit-Wigner peak on top of a combinatorial background due to all tracks which are of the same species of that resonance daughters but don't come from a resonance decay. A good estimation of this background is in order if one wants to cleanly observe the peaks. In addition, a good particle identification (PID) is crucial for a good selection of tracks for this study.

A sketch of the package classes is shown in figure 1. The main operation consists in building this peak using the 4-momenta of the reconstructed tracks. Since the track reconstruction only returns a vector momentum and a charge sign, this information must be completed by assigning a mass to each track, which allows to compute its energy. This can be done using the different particle identification (PID) informations coming from the ALICE detectors. On the other hand, in a "first physics" analysis scenario a complete PID information could be unavailable: in this case, a mass hypothesis is done depending on the studied resonance, and this determines the mass assigned to each track. Of course, in the latter scenario the mis-identified tracks will increase the background.

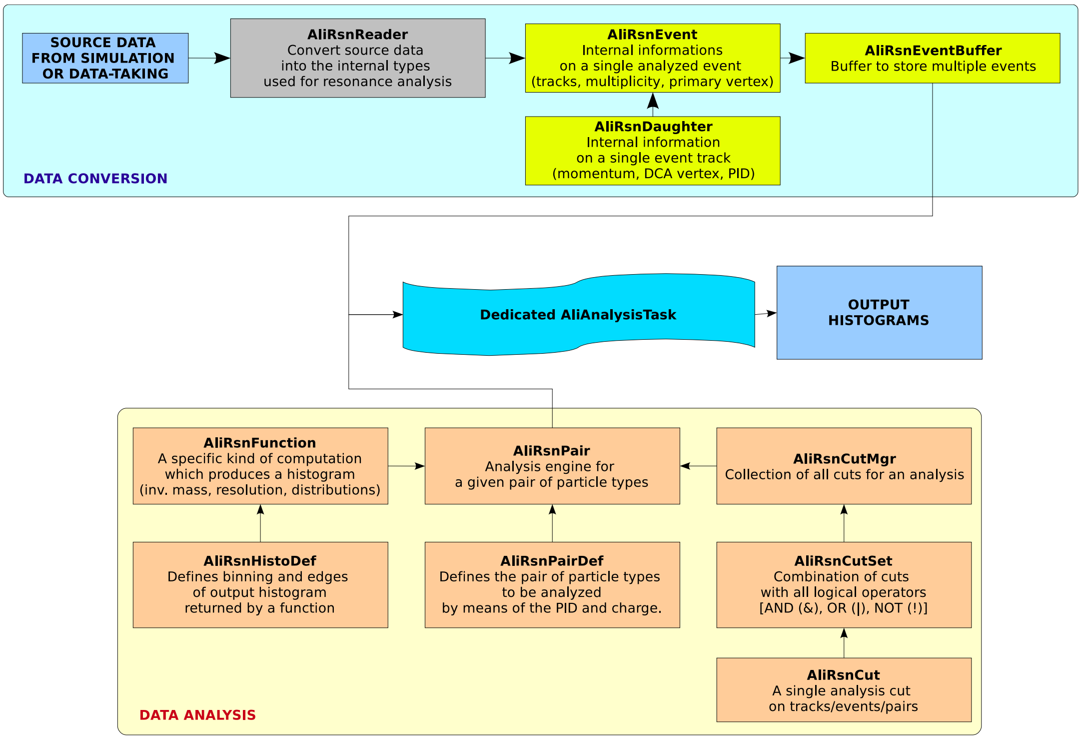

Figure 1: A sketch of the resonance analysis package structure.

A crucial step during the peak extraction is the estimation of the combinatorial background due to all track pairs which are not produced in a resonance decay. This can be done in different ways:

- "wrong sign" pairs in the same event: for example, when reconstructing a neutral resonance, its peak is computed through the invariant mass of all unlike-sign pairs of tracks, and the background can be estimated by computing a similar invariant mass distribution with all *like*-sign pairs;
- event mixing: in this case, one builds an invariant mass distribution using pairs of tracks taken from different events, which can be considered "similar" enough (e.g.: with almost the same multiplicity and position of primary vertex).

Both of these methods have been implemented and tuned, in order to minimize the discrepancy with the "true" background (which can be estimated when looking at simulated data).

A fundamental preliminary step for the analysis is the access to the data. The final output of global ALICE reconstruction is the Event Summary Data (ESD), which contains al of the information coming from the whole ALICE assembly: tracks with their momenta and PID probabilities determined by all devices, vertices, clusters from calorimeters, multiplicity estimations, and so on.

The ESD is conceived as the unique source of data which any analysis operation must access in order to extract informations of physical interest from collisions, and this causes it to be an extremely large object. On the other hand, each analysis task uses only a subset of the whole ESD information, so it is convenient to define a "derived" data structure which focuses on the relevant info only, in order to produce smaller objects which are easier to be transported through the network and can be kept in large quantities in the memory of a typical PC. A first step in this direction is the production of the Analysis Object Datasets (AOD), which converts the ESD information in a more compact and user-friendly way, and standardizes the access to ESD data for most analysis aspects. The AODs can be used as a starting point when the standard data interpretation they permit is suitable for the analysis (this will not be the case, for example, when one has special requirements for PID or track selection). Furthermore, when dealing with simulated data, it must be possible to read directly the Monte-Carlo events in order to check any effect which can be introduced by the reconstruction itself, allowing to carry out an analysis even on simulated data.

From the point of view of the resonance analysis, the AODs themselves contain much unneeded information, while some of the skipped information from ESD could be required (e.g.: a customized PID configuration). Moreover, resonance analysis takes much advantage from the possibility to study directly the Monte-Carlo data, since this could help in isolating any kind of effects caused by the reconstruction (momentum resolution, tracking efficiency, and so on). Since each of these available sources of data must be treated differently, the resonance analysis package has defined an internal data structure which helps in simplifying the package design. Then, it provides a converter from each kind of source data (ESD, AOD or MC) to this common internal format, while the package deals with the latter only, and the differences between the various sources of data can be forgotten after this preliminary conversion step. When dealing with the ESD directly, such a preliminary step allows to customize the way to use PID information. In fact, there is a “standard” way for combining the weights using a Bayesian method which returns the probability for each track to be identified as a given particle species (electron, pion, kaon, ...), but one may want to skip the information from some devices and use only a subset of them. The possibility to do this has been implemented in the package.

Another important issue during analysis is the data quality: a huge effort must be done in finding all useful criteria which allow us to select a “clean” sample of tracks in order to minimize the background and, in particular, to reduce spurious effects which can be due to tracks which are not coming from the primary vertex, or tracks not reconstructed well enough. In order to face this problem, a set of classes was implemented to cut on several types of track information (momentum, distance of closest approach to primary vertex, χ^2, number of clusters, etc.). Cuts are implemented in a way that allows one to combine them in Boolean expressions: this permits a highly customizable configuration of track selection criteria, which goes beyond the simple logical AND of several cuts.

The core of the analysis consists in a “function”: it takes each pair of tracks which pass the cuts, and from each one it computes a specified value which is used to fill an output histogram. This operation is implemented in an object which can return several types of output histograms with very few simple settings. This object can compute the invariant mass distributions and some other histograms which can be of interest in this analysis: for example, the invariant mass resolution or the efficiency (at least as long as the analysis is run on simulated data and one can check how well a resonance was reconstructed), or the momentum distributions of reconstructed or generated resonances.

The whole package structure provides an object which acts as an “analysis engine”: it must be initialized specifying the resonance decay channel to be analyzed, the selection cuts and the

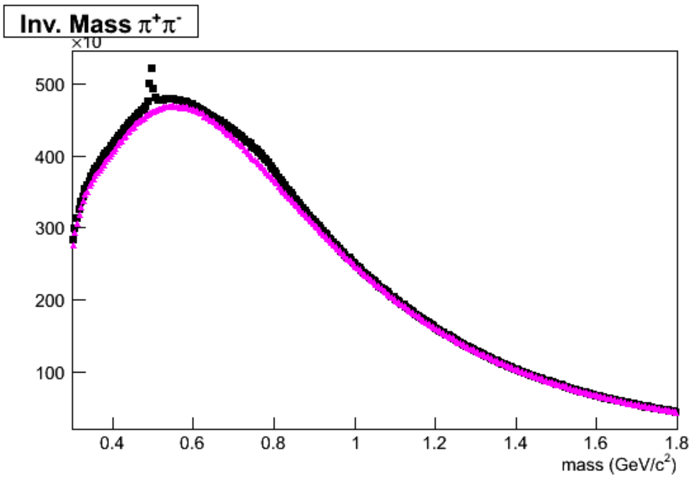

Figure 2: Package output example 1. Two invariant mass distributions computed with different track types: in black there is the distribution of "signal" where the Breit-Wigner peak can be seen above the background; in pink there is a background estimation through like-sign pion pairs.

desired functions to be computed. A user can even initialize many instances of this object for different kinds of analyses in a single job. Once each "engine" has been properly initialized, it is inserted into a suitably implemented "task" object, designed according to the standard guidelines of the `AliRoot` Analysis Framework, and then it can run on any sample of data, into any kind of available environment (GRID, CAF or locally). The final result is a set of histograms which contain the outputs of the declared functions.

3.2 Examples

Here we show two examples of the package output in the case of a $\rho^0(770)$ resonance analysis through its decay channel into a pair of unlike-sign pions. Computations are done on simulated p-p collisions at the center of mass energy of 10 TeV.

Figure 2 shows two invariant mass distributions. Black squares are the $\pi^+\pi^-$ "signal" pairs (i.e. pairs of tracks of the same type as the expected resonance daughters), where the Breit-Wigner peak can be seen on top of the combinatorial background. The pink triangles are the background estimation computed using $\pi^+\pi^+$ and $\pi^-\pi^-$ like-sign pairs, which reproduce quite well the combinatorial background: in fact, in the regions outside the peak the two distributions are quite similar, but in the latter no peaks are present. After subtraction of the estimated background from the signal, one obtains the result shown in figure 3, where the ρ^0 peak is clearly visible. In this case, one can also observe the peak coming from K_s^0 decay around $\sim 500\,\mathrm{MeV}$. The ρ^0 peak can then be fitted with a Breit-Wigner function, in order to compute its mass and width.

In order to test the performances of the package in a typical personal computer, a test analysis has been executed on a sample of simulated events generated with a simple particle generator which produced 50 pions and 20 kaons of each charge sign with a flat momentum distribution between 0 and 10 GeV$/c$. Table 1 shows an estimation of RAM occupancy and CPU time required for an analysis of these events with a task configured to execute 4 different analyses, each one with 2 functions divided into 7 transverse momentum bins. In total, each analysis computes 42

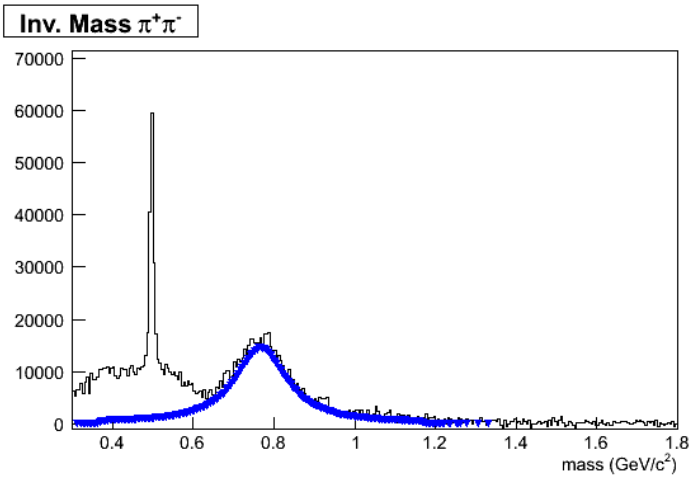

Figure 3: Package output example 2. A peak obtained from the subtraction of the background from the signal distribution shown in previous figure. The blue line is a fit of a Breit-Wigner function.

Analyzed events	5300
CPU time total	72 min
CPU time per event	0.81 s
CPU time per track pair	7.4×10^{-5} s
RAM occupancy	26.6% of 2 GB

Table 1: Results of a performance test on 5000 events generated with a 140 primaries with a flat momentum distribution between 0 an 10 GeV$/c$, with a Pentium II @ 2.60 MHz with 2 GB of RAM.

histograms and uses all tracks which pass the cuts, without any selection in terms of PID: each track is assigned the kaon mass by default and its energy is computed accordingly from reconstructed momentum. As the table shows, the execution CPU time in this case is of 72 minutes for the whole sample, which means almost 0.81 s per event. In each event almost 150 tracks are used, which determines that almost 11000 pairs are processed: then, the total CPU time required per each pair is in the order of 7.4×10^{-5} seconds. For what concerns memory occupancy, a buffering system was implemented to control the amount of RAM used to store events, and in this test it was initialized in order to contain the whole sample of 5300 events. In this case, this required 26.6% of the total available RAM. The possibility to keep in memory a huge sample of events helps in speeding up a lot the analysis, since this avoids multiple accesses to files, which could be quite time consuming especially in the case that files come from a world-wide GRID environment.

4. Conclusions

The resonance analysis is a very important issue in the soft physics ALICE analysis programme, since it is a well-suited probe of the hot nuclear matter produced in LHC heavy-ion collisions.

A whole analysis package has been developed in order to cope with this physics analysis topic. It implements a set of useful functionalities which optimize computing time and memory occupancy for the execution of this analysis on a huge amount of collision events. Moreover, the package is compliant with the standard analysis structure provided by the ALICE Analysis Framework, and is able to run on any of the available computing infrastructures provided for the ALICE collaboration, both through GRID and small local computing clusters.

The package is ready to deal with realistic data, it is completely integrated in the `AliRoot` framework and it is stored in its official Subversion repository.

References

[1] R. Rapp and J. Wambach, Adv. Nucl. Phys. **25** (2000), 1
R. Rapp, Phys. Rev. **C63** (2001), 054907.

[2] E.V. Shuryak and G.E. Brown, Nucl. Phys. **A717** (2003), 322

[3] R. Rapp, Nucl. Phys. **A725** (2003), 254

[4] J. Letessier et al., J. Phys. **G27** (2001), 427
G. Torrieri and J. Rafelski, Phys. Lett. **B509** (2001), 239

[5] P. Fachini, J. Phys. **G35** (2008), 044032
C. Markert, J. Phys. **G35** (2008), 044029

[6] ALICE Collaboration, J. Phys. **G30** (2004), 1517

[7] ALICE Collaboration, J. Phys. **G32** (2006), 1295

[8] `aliceinfo.cern.ch/Offline`

[9] `root.cern.ch`

[10] S. Bagnasco et al. J. Phys. **G119** (2008), 062012
`http://alien.cern.ch`

[11] G. Ganis, *contribution in this book*

[12] J.F. Grosse-Oetringhaus, J. Phys. **G119** (2008), 072017
`http://aliceinfo.cern.ch/Offline/Activities/Analysis/CAF/index.html`

[13] A. Gheata *contribution in this book*