PWG3 analysis (barrel)

https://twiki.cern.ch/twiki/bin/view/ALICE/PWG3Hadron
https://twiki.cern.ch/twiki/bin/view/ALICE/PWG3Electron
https://twiki.cern.ch/twiki/bin/view/ALICE/PWG3JpsiDielectrons

Andrea Dainese

(INFN Padova)

with A.Andronic, F.Antinori,

G.E.Bruno, G.Martinez, S.Masciocchi

Contents

- Analysis strategy
- New developments for the three groups
  - Hadronic charm
  - Heavy-flavour electrons
  - J/ψ to di-electrons
- Requirements

General analysis strategy

- Candidates produced in one go:
  - decays (hadronic charm & J/psi): starting from the AOD (or ESD) and writing output to deltaAOD
  - electrons: starting from ESD and writing full PID info in AOD

  this is done in the Official AOD-creation train (which starts from ESDs)
    - for hadronic charm, can be done also in successive train that reads the already-produced AOD and re-creates only the deltaAOD file
      - **never tested (deltaAOD storing issue)**
- Candidates analysed with trains that read AODs+deltaAOD
  - output as histograms (+some ntuple-like objects, only if size is very limited, example later)
  - **are and will be run very frequently**
- Final candidates (selected by final cuts) should be promptly accessible for fast analysis
  - will try using the tags (book-keeping machinery mandatory)
    - but may end up having 1 tagged event per AOD file ...
  - mini-AODs with selected events to be staged on CAF?

Hadronic Charm (D2H) [PWG3/vertexingHF]

Heavy-flavour vertexing: production of charm candidates

- AliAnalysisVertexingHF: main class to produce the candidated charm decays: 2-prong, 3-prong, 4-prong, cascades (D*). In one go.
- It is the “core” of AliAnalysisTaskSEVertexingHF
  - wagon of the Official Analysis Train
- Output written to AOD event (friend tree in deltaAOD file AliAOD.VertexingHF.root)
- Configured by AliAnalysisVertxingHF* ConfigVertexingHF.C
- A wagon of the Official Train since the beginning
- Runs smoothly in pp
- Not in Pb-Pb (combinatorics too large)
  - need to tighten the cuts for Pb-Pb (“high-flux” configuration)

Heavy-flavour vertexing: preparation of event mixing

- Goal: possibility to use mixed events to study background
- Need to mix AODs on Grid (not yet available, in progress)
- Need to mix 2, 3 or 4 events
- Use AliAnalysisTaskME
- Dedicated class AliMixedEvent : AliVEvent implemented by Andreas
- Event selection
  - multiplicity (OK)
  - vertex position in z (OK)
  - event shape/jettiness (under way)
- Computation of mean primary vertex and translation of tracks to this common vertex (needed for vertexing)
- Integration in common code for candidates production is essentially ready

Heavy-flavour vertexing: Analysis of candidates from AOD

- Input: AliAODs.root + AliAOD.VertexingHF.root
- Prepared "D2H" analysis train:
  - Steering macro RunAnalysisAODVertexingHF.C
  - runs on the Grid with AliEn plugin and on CAF
    - very convient to use same code in both cases
    - plugin makes job submission and input data selection very easy
    - CAF very efficient to test analyses, and more... (16M AOD evts analysed)
  - 8 wagons (TaskSE) up to now
    - CompareHF (vertex resolutions)
    - D0 mass analysis
    - Dplus mass analysis
    - Like-sign bkg D0→Kpi
    - D0→Kpi CORRFW task
    - Prompt D0 fraction
    - Like-sign bkg Jpsi→ee (from JpsiDielectrons group)
    - Analysis for Jpsi from B (from JpsiDielectrons group)
  - **6 of these are ready to run in Official AOD Train**

Heavy-flavour vertexing: analysis experience on Grid

- Production of AODs from ESDs inefficient? (<60-70% compl. rate)
  - AODs for (much) less than 100M min. bias pp events (~70M)
- Untill July we could analyse 17M of LHC09a4 (min. bias) from 1st Official Train run (may)
  - this was mainly due to the "unzip error"
- At the beginning of September we could analyze, for 2 weeks, about 38M (40%) of LHC09a4 from 2nd Off Train run (aug)
  - "unzip error" was still there
- From mid-September to last week, the maximum we could analyze was about 5M events
  - AODs from new run of Off Train (Oct, AOD1) tried
- Main problems:
  - at some point had to use par files, because AliRoot tag delayed by 10 days (though for good reasons)
  - all AODs in one single SE (no replicas)
    - SE was overloaded due to lack of prod jobs and too many "hot" datasets kept
- Now trying with AODs from last Off Train (AOD2, replicated in 5 SEs)

Heavy flavour electrons (HFE) [PWG3/hfe]

Status of electron analysis

- Analysis Task reading ESDs
  - being included in the official train this week
  - very good support from Mihaela and Andreas
  - testing phase on GRID problematic at the moment
- Selection of Electron Candidates in the ESD filter
  - very loose selection (pixel OR, 3σ from TPC electron line)
  - tracks are labeled in the standard AODs, and the full AOD PID object is saved
  - already included in the LHC09a4/AOD2 production

Status of electron analysis

- Next: Analysis Task reading AODs
  - to go on a second official train, with AODs as input
  - some code in not yet in place (e.g. use the AOD PID object for individual detectors)
    - discussion with PID task force

→ The HFE analysis is developing very quickly at the moment, and needs frequent cycles

→ Completely based on electron PID, will need ESD based analysis with OCDB access and the tender, to make use of improved calibration

J/psi → ee
[not yet a dedicated module]

$J/\psi \rightarrow ee$ analysis in the barrel

- New framework for J/ψ candidate selection and analysis being developed
  - to decouple from D2H software
  - to fully exploit the HFE package for electron i.d.

  … but nothing yet in aliroot
- code developed so far (for $B \rightarrow J/\psi$ analysis) within D2H module will be maintained and further developed
  - can handle total J/ψ analysis
  - PID part to be improved
- the specific issue discussed next refers to both cases

Specific issue of this analysis

- To separate prompt and secondary (i.e. from B) J/ψ, the final analysis consists of a likelihood fit to the un-binned data
  - fits have to be run interactively (initial conditions, fixing of parameters, trying it several times, etc...) over all the candidates (≈10K) of the full statistics
- The analysis scheme is the following:
  - Selection of pair candidates:
    - at the level of [AOD + delta AOD] building task
      - a very few candidates / events
        even without pid cut, but simply with $M(e^+e^-)>M\_min$
  - Task (on AOD+deltaAOD) to select candidates
    - output: histograms and **ntuple** (→unbinned data)

Can we produce output which is not "histogram" in the official AOD train ?

Request: produce AODs + vertexingHF delta AODs for LHC09a7,a9

- Run the analysis train to produce the standard AODs (with MC info) plus vertexingHF delta AODs on those productions:
  1. LHC09a7 (pp min-bias + 1 J/ψ into e^+e^-)
  2. LHC09a9 (pp min-bias + 1 particle of the χ_c family)
  - how? As done for LHC09a8 or a4,a5
  - official request already submitted via savannah this monday

General PWG3-barrel Requirements

- Fixed schedule for ESD train (monthly?)
  - producing AODs and running analyses on ESDs and AODs
    - techinical development to make deltaAODs available as input is needed (in progress): will be very useful!
    - systematic AOD replication (e.g. LHC09a4/AOD2 replicated 5 times: OK)
- Weekly analysis tags of AliRoot (fixed day)
  - allows to plan the developments and commits
- Create an official train to run on AODs (weekly?)
- Possibility to consider an approach in which we produce mini-AOD objects of small size (for final analysis on CAF)

EXTRA SLIDES

AliAnalysisVertexingHF class

single-track cuts using AliESDtrackCuts
[2 sets of cuts: “displaced track”, “soft pion (D*)”, can be extended e.g. electrons]

build (+,-) pairs and like-sign with “displ. tracks” and compute secondary vtx

for $D^0 \to K\pi$, $J/\psi \to ee$

create AliAODRecoDecayHF2Prong
apply reco cuts

for $D^* \to D^0\pi$

special D0 cuts,
loop on “soft pions”
for D* candidates

store D^0

store D^{*+}

store J/ψ

for D^+, D_s^+, Λ_c^+

loop on all tracks (+ & -):
build triplets,
create AliAODRecoDecayHF3Prong,
apply reco cuts (common for the 3 particles?)

store D^+, D_s^+, Λ_c

for $D^0 \to K\pi\pi\pi$

loop on all tracks (+ & -) …
create AliAODRecoDecayHF
…

store D^0

AliVEvent as input

- Motivation: AOD as well as ESD
  - AODs are smaller in size, will be replicated in more SEs
- Strategy followed:
  - AOD well suited for “kinematics” analysis, not for vertexing
  - ESD well suited for vertexing (AliExternalTrackParam+AliESDVertex)
  - Use “virtual” interfaces to read ESD or AOD with the same code
    - in case of AOD input, convert to ExternalTrackParam and ESDVertex
  - Do vertexing as from ESD

Use virtual interfaces

Virtual interfaces: AliVEvent | AliVParticle | AliVTrack | AliVVertex

ESD world	AOD world
AliESDEvent	AliAODEvent
AliESDVertex	AliAODVertex
AliESDtrack	AliAODTrack

- Possible issues:
  - no common interface for single-track cuts (AliESDtrackCuts used, but a ESD track created from a AOD tracks lacks some info, e.g. nTPC clusters)
  - PID (e.g. electron ID in TRD will need momentum at the TRD)
  - cuts used to create AOD from ESD have to be known!!! stored in alien together with AOD? database?

Storing the candidates in the AOD: AliAODRecoDecay family