Status of Analysis Tools
in PandaROOT

2. 12. 2010

K. Götzen, GSI

What are Analysis Tools?

I think of roughly three categories:

- 1st level tools
  - Access to event store
  - Particle candidate structure/container
  - List containers for particle candidates
  - Tool for combinatorics/decay tree building, ...

- 2nd level tools
  - Kinematic Selectors, PID-Selectors/Algorithms
  - Kinematic/Geometric Fitting
  - PDG database, mathematical tools, etc, ...

- 3rd level tools *(experiment/framework independent)*
  - PWA Analysis
  - Multivariate Analysis
  - RooFit-like Tools ($_s$Plot, unbinned ML Fitting, etc), ...

1st level tools – Data Access

- Access to event data: PndEventReader

```
PndEventReader evr("mymicro.root");

while (evr.GetEvent())
{
  evr.FillList(mylist, " All ");
  //... your analysis code ...
}
```

Concatenation works, but
w/o friend mechanism
(i.e. only for one output stage).

- or PndAnalysis

```
FairRunAna *fRun = new FairRunAna();
fRun->SetInputFile("mymicro.root");
fRun->AddFriend("myreco.root");

PndAnalysis evr();

while (evr.GetEvent())
{
 evr.FillList(mylist,“All“);
  //... your analysis code ...
}
```

Concatenation works for
multiple output stages now!

(Thx to Mohammad and Florian!)

1st level tools – Particles & Lists

- Particle candidates, lists of candidates, combinatorics
  TCandidate, TCandList

```
TCandList ep, em, jpsi;  // the lists for e+, e- and J/psi->e+e-
while (evr.GetEvent())
{
    evr.FillList(ep, "ElectronLoosePlus");
    evr.FillList(em, "ElectronLooseMinus");
    jpsi.Combine(ep, em);

    for (int i=0; i<jpsi.GetLength(); ++j) histo->Fill( jpsi[i].M() );
}
```

TCandidate: *M, P, E, Vtx, LV, daughters,...*

- More complicated combinatorics

```
    evr.FillList(ep, "ElectronLoosePlus");
    evr.FillList(em, "ElectronLooseMinus");
    evr.FillList(pip, "PionLoosePlus");
    evr.FillList(pim, "PionLooseMinus");

    jpsi.Combine(ep, em);

    psi2s.Combine(jpsi, pip, pim);
```

- decay tree
- combine up to 5 lists
- checks double counting
- checks overlaps

2nd level tools – Kinematic Selectors

Package rho/RhoSelector: Kinematic selectors

- TPidChargedSelector, TPidNeutralSelector, TPidPlusSelector, TPidMinusSelector – selects the corresponding charge

```
evr.FillList(myList, „All");   // fills with charged and neutrals

TPidChargedSelector *chrgSel = new TPidChargedSelector(„chrgSel“);

myList.Select(chrgSel);        // now contains only charged
```

(PndEventReader/PndAnalysis :„Charged“, „Neutral“, „Plus“, „Minus“)

- TPidMassSelector, TPidEnergySelector, TPidMomentumSelector – Selects mass, energy, momentum in window

```
jpsi.Combine(ep, em);          // combines J/psi candidates from e+ e-

TPidMassSelector *jMassSel=new TPidMassSelector(„jMassSel“, 3.096, 0.1);

jpsi.Select(jMassSel);         // selects masses m = 3.096 ± 0.05 GeV
```

2nd level tools – PID Selectors

Package rho/RhoSelector: PID selectors

- TPidSimpleElectronSelector, ..., TPidSimpleProtonSelector – PID selectors for e, μ, π, K, p

```
evr.FillList(myList, „Plus");  // fills with positiv charged particles

TPidSimpleKaonSelector *kSel = new TPidSimpleKaonSelector(„kSel“,“loose“);

myList.Select(kSel);       // now contains K candidates, loose selection
```

  Act on PID likelihood values:
  - veryLoose : P ≥ 0.0 (basically sets mass of 4-vector)
  - loose : P ≥ 0.2
  - tight : P ≥ 0.5
  - veryTight : P ≥ 0.9
- PndEventReader keywords: Combination (Electron, Muon, Pion, Kaon, Proton) + (VeryLoose, Loose, Tight, VeryTight) + (Plus, Minus), e.g. *„ElectronVeryLoosePlus“*, *„MuonTightMinus“*
- *Full PID info still missing.*

2nd level tools – Further Selectors

Package rho/RhoSelector: Further selectors (not tested yet)

- TGoodPhotonSelector – quality selector for neutrals
- TGoodTrackSelector – quality selector for tracks
- TPidNNOElectronSelector, ... , TPidNNOProtonSelector – PID Selection based on neural networks for e, μ, π, K, p; adaption needed for usage
- TPidElectronTruthSelector, ..., TPidProtonTruthSelector – Selector for MC Truth match; has to be adapted for usage
- TEventSelector – Selects whole events; has to be adapted

2nd level tools - Fitters

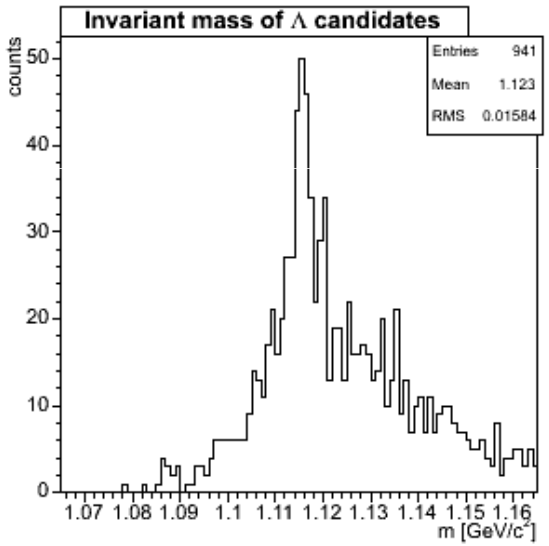
Package PndTools/AnalysisTools/Fitter

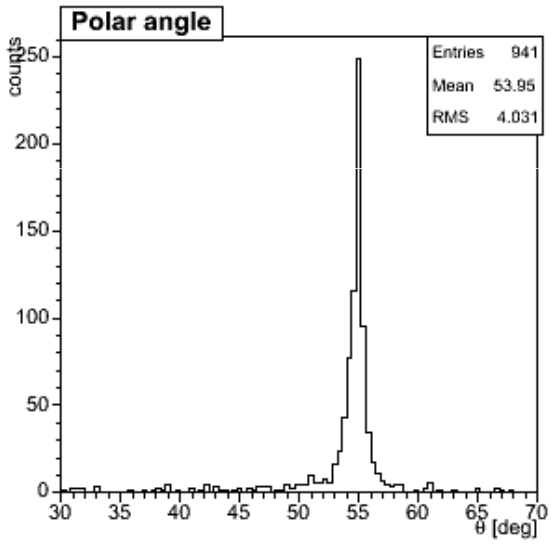
- PndVtxFitter – pure vertex fitter
- PndChiVtxFitter – pure vertex fitter
- PndKinVtxFitter – vertex fitter with additional mass and pointing constraint
- PndKinFitter – kinematic fitter, taking constraints on P4, P3, P, E, M
- PndMassFitter – pure mass constraint fitter
- Pnd4CFitter – pure 4 constraint fitter

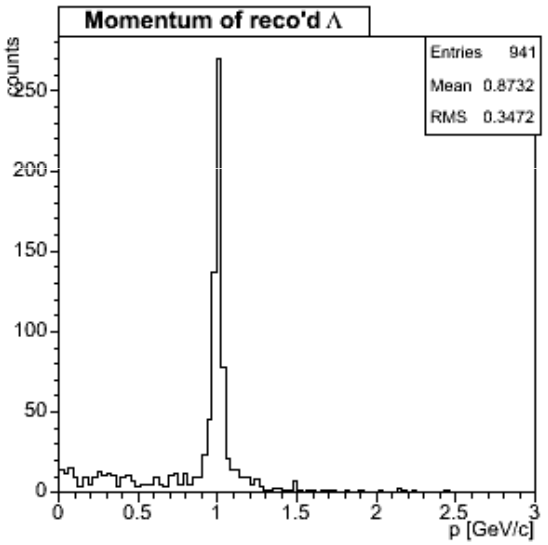
- Test vertex fitter with single Λ ($\theta = 55°$, $\varphi = 20°$, $p = 1$GeV)
- Test 4C & mass fit with $\psi(2S) \rightarrow J/\psi\ \pi^+\pi^-$

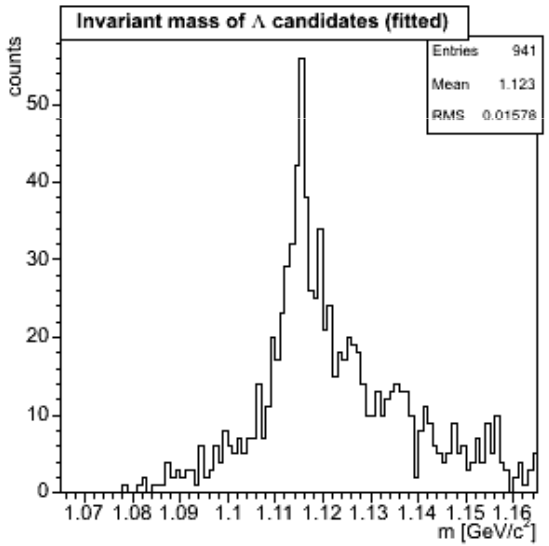
Fitting test – PndVtxFitter (Vtx-C.)

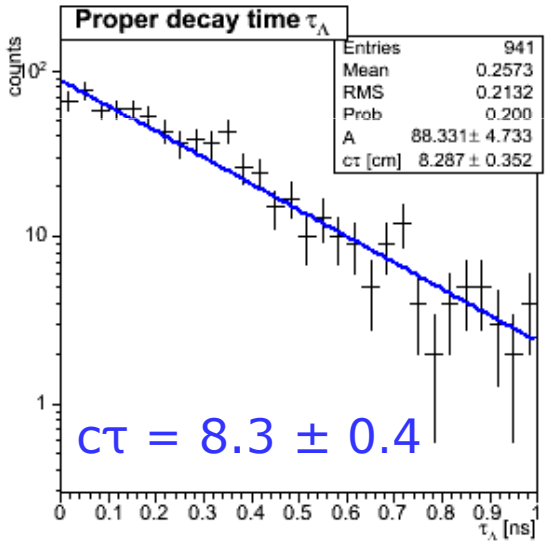
Single Λ (θ = 55°, φ = 20°, p = 1GeV), 2k events

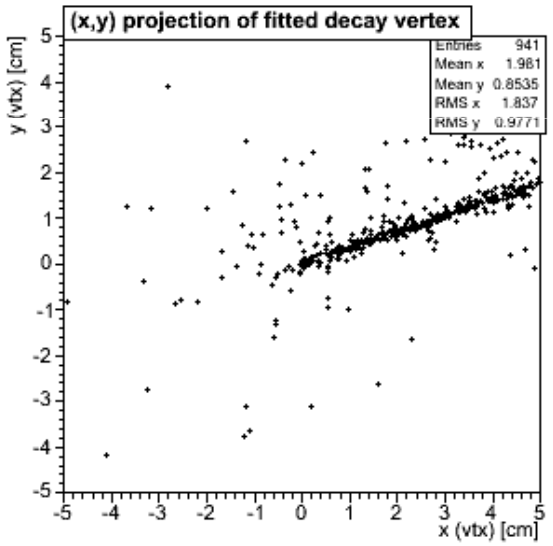






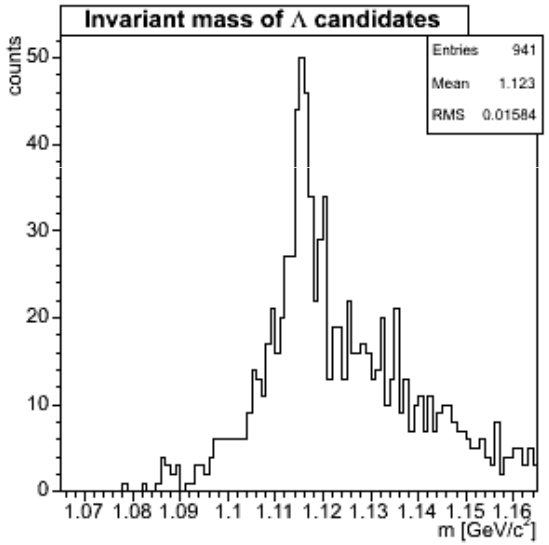


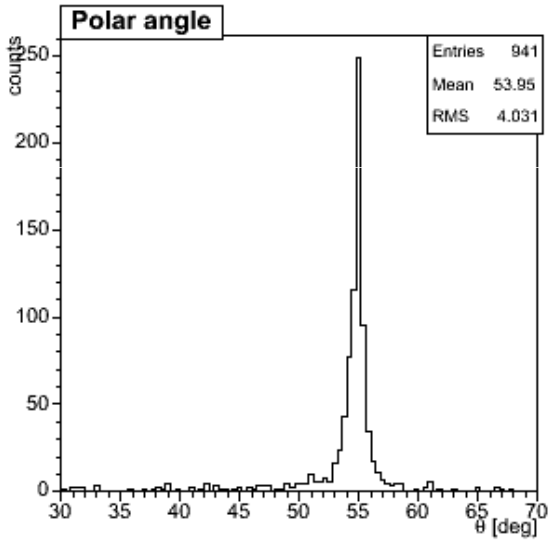


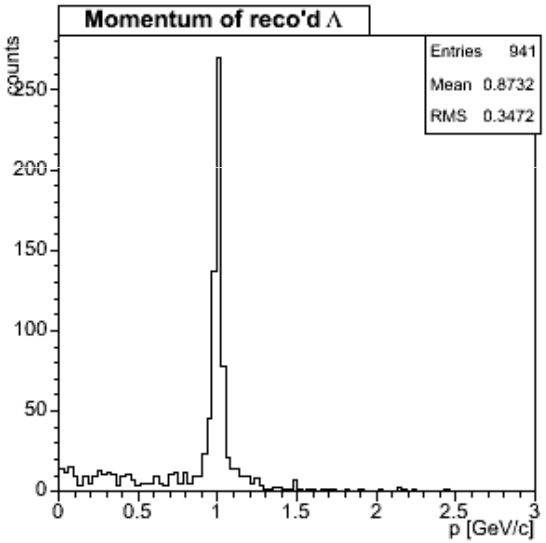


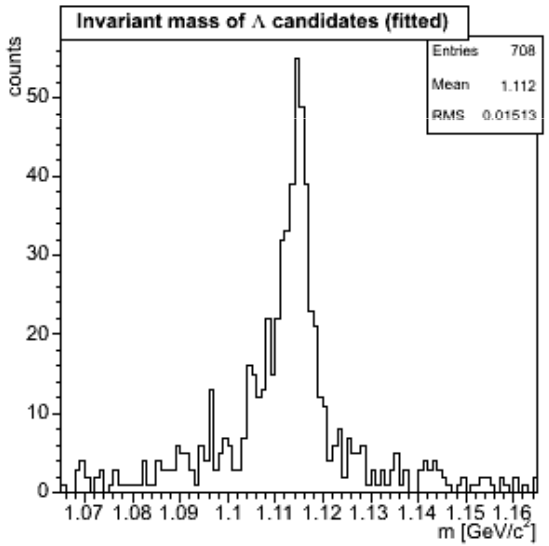
Fitting test – PndKinVtxFitter (Vtx-C.)

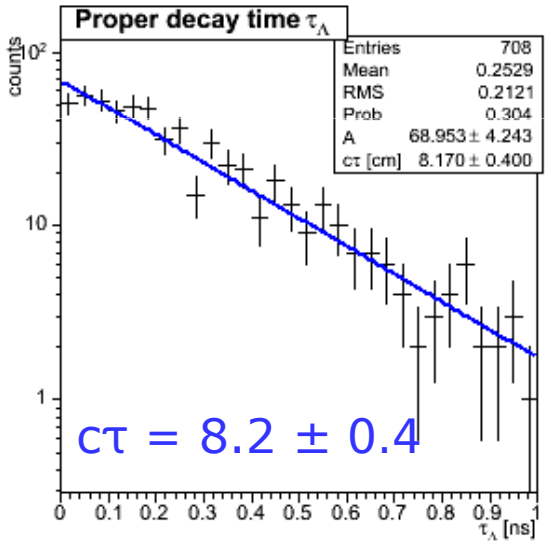
Single Λ (θ = 55°, φ = 20°, p = 1GeV), 2k events

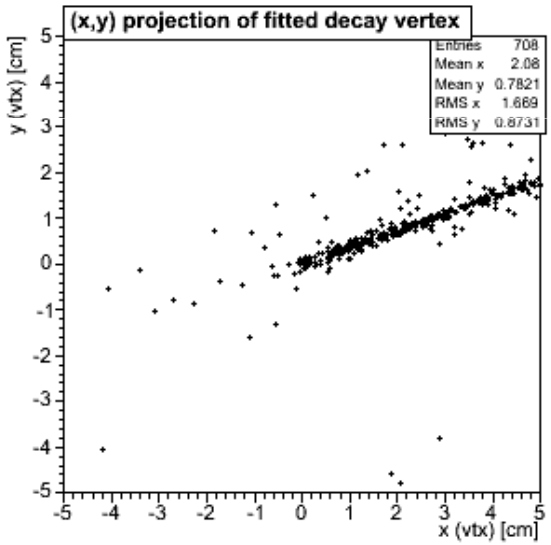






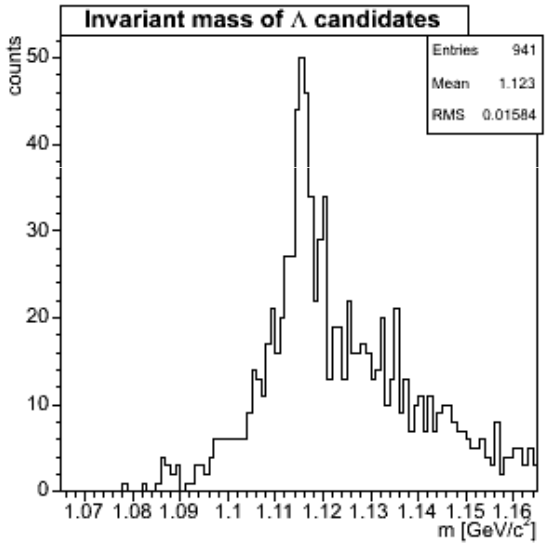


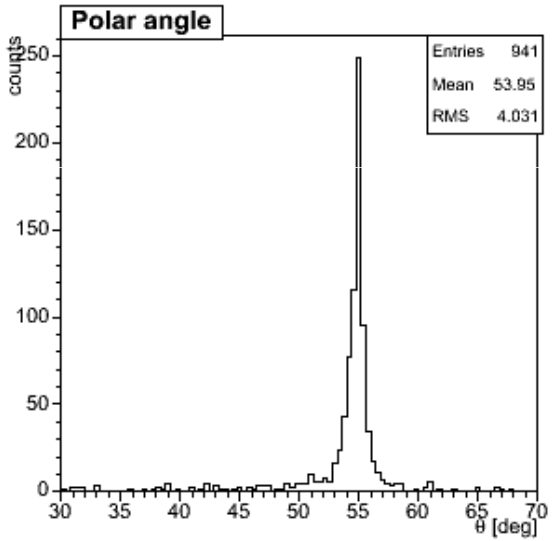


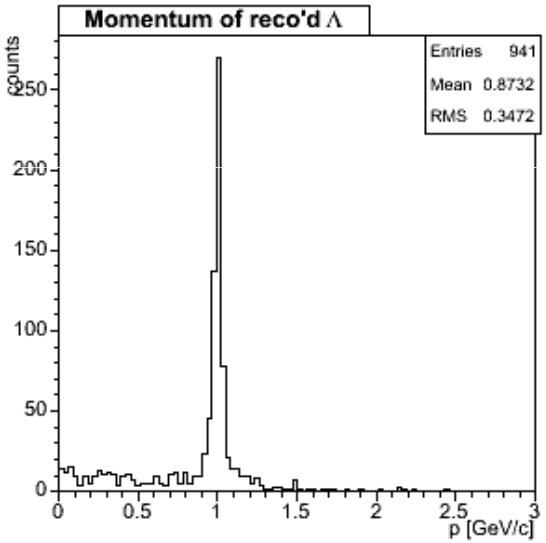


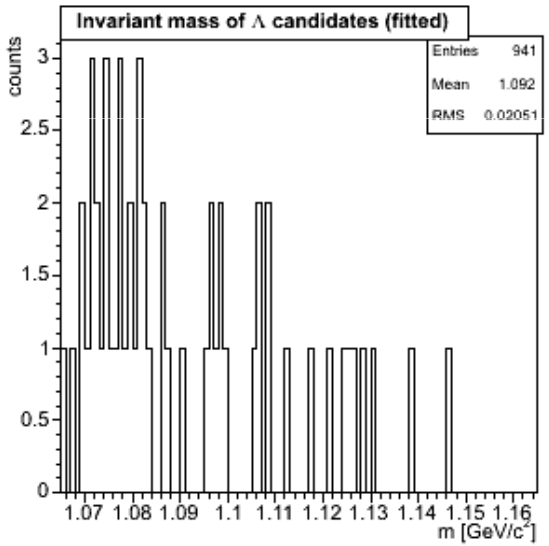
Fitting test – PndChiVtxFitter (Vtx-C.)

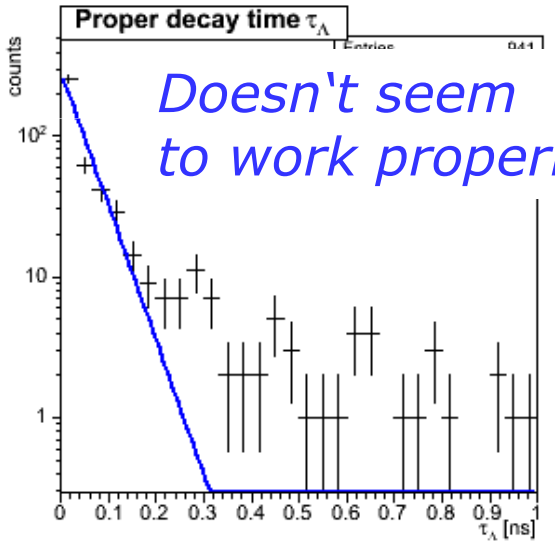
Single Λ (θ = 55°, φ = 20°, p = 1GeV), 2k events



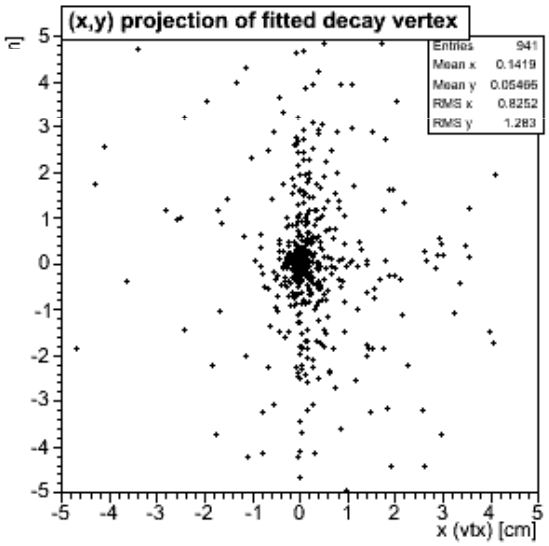






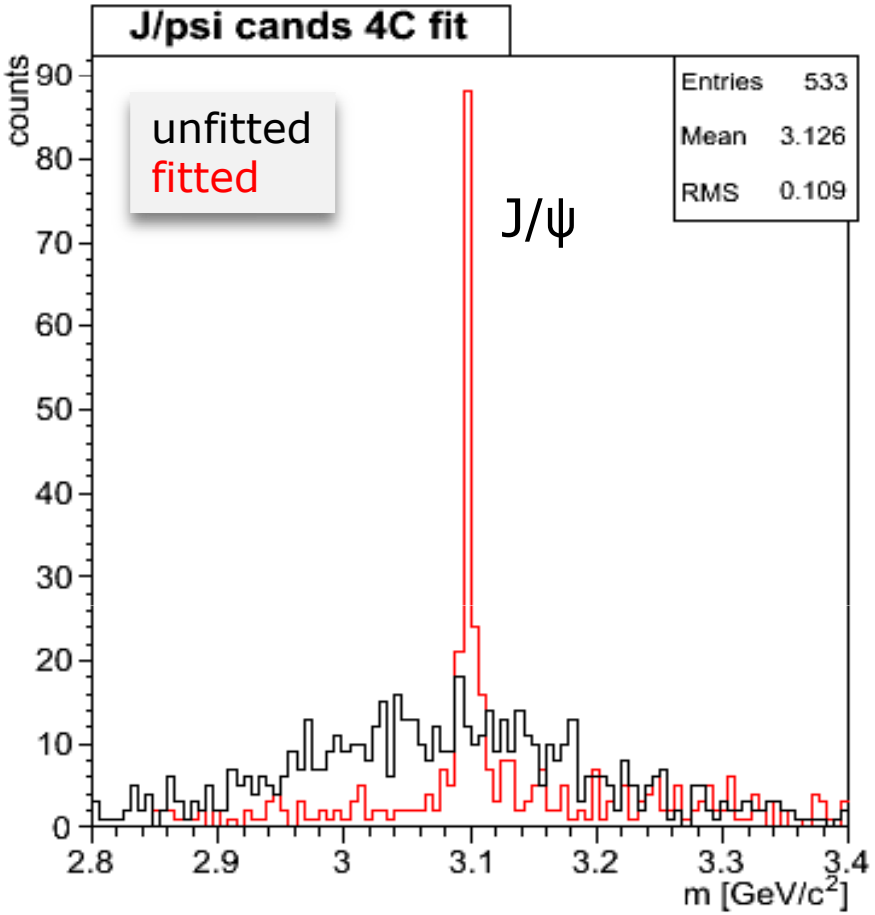


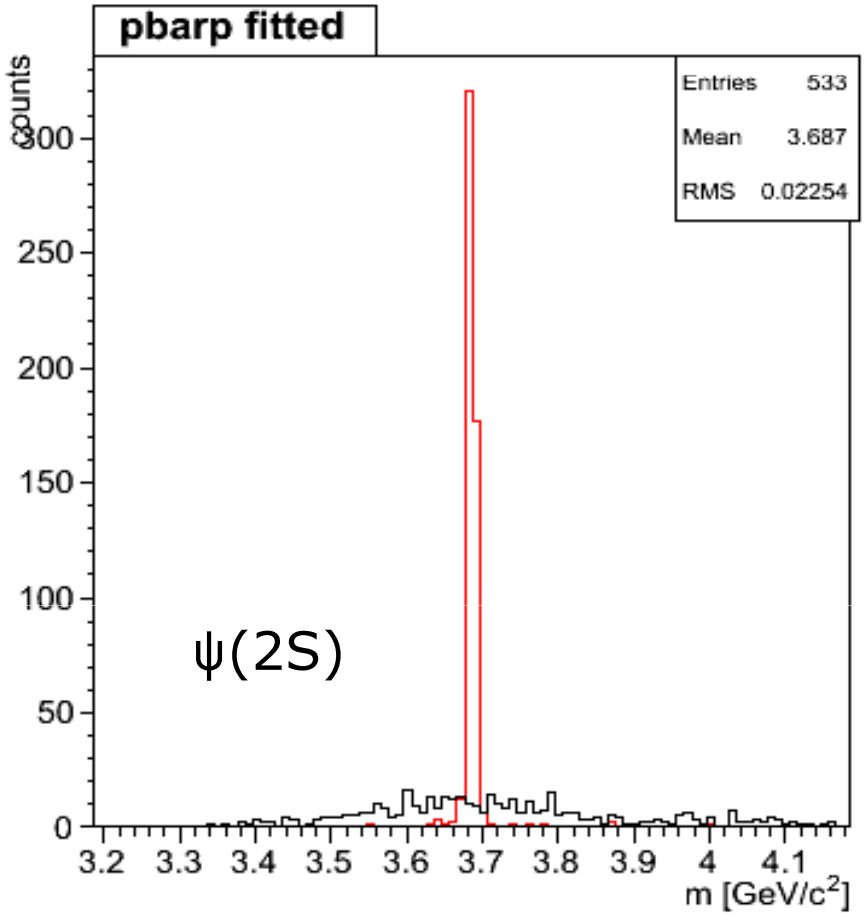
Doesn't seem to work properly!



Fitting test – PndKinFitter (4Mom-C.)

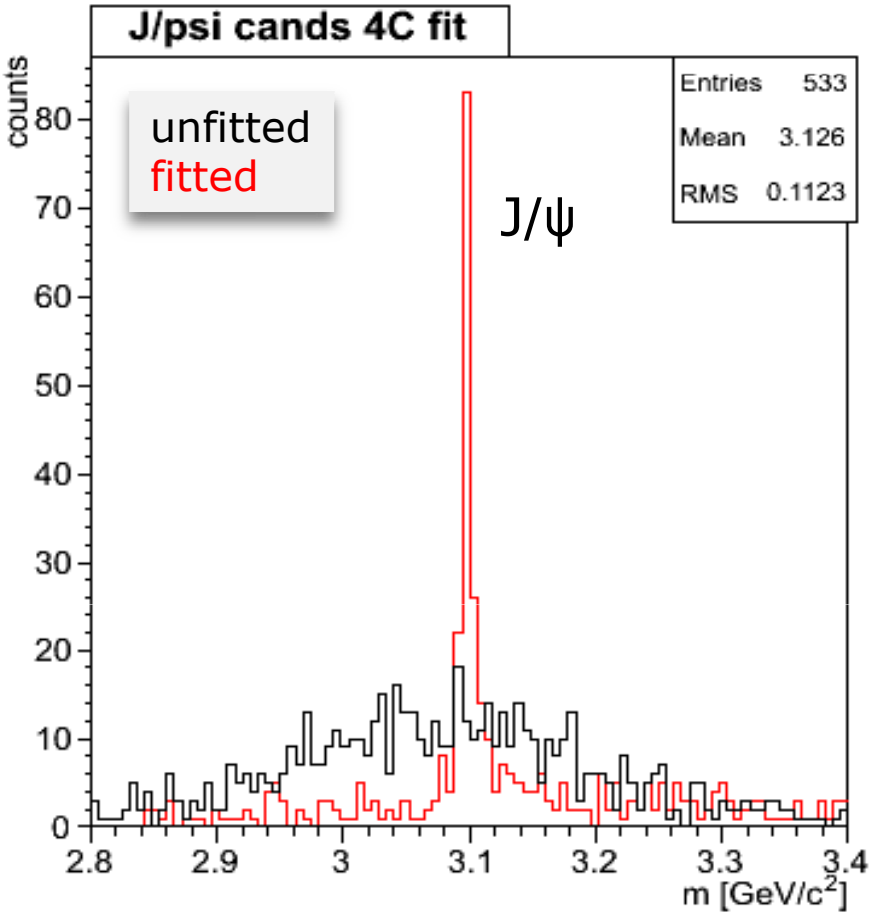
Channel $\psi(2S) \rightarrow J/\psi\ \pi^+\pi^-$, 4 constraint on $\psi(2S)$, 2k events

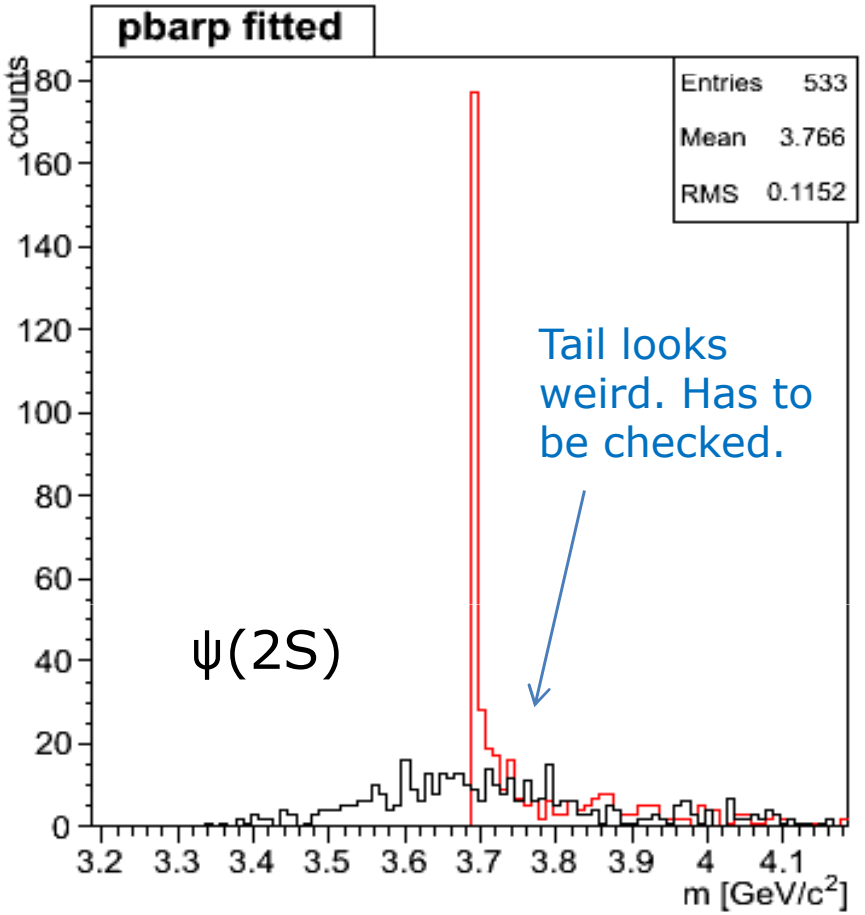




Fitting test – Pnd4CFitter (4Mom-C.)

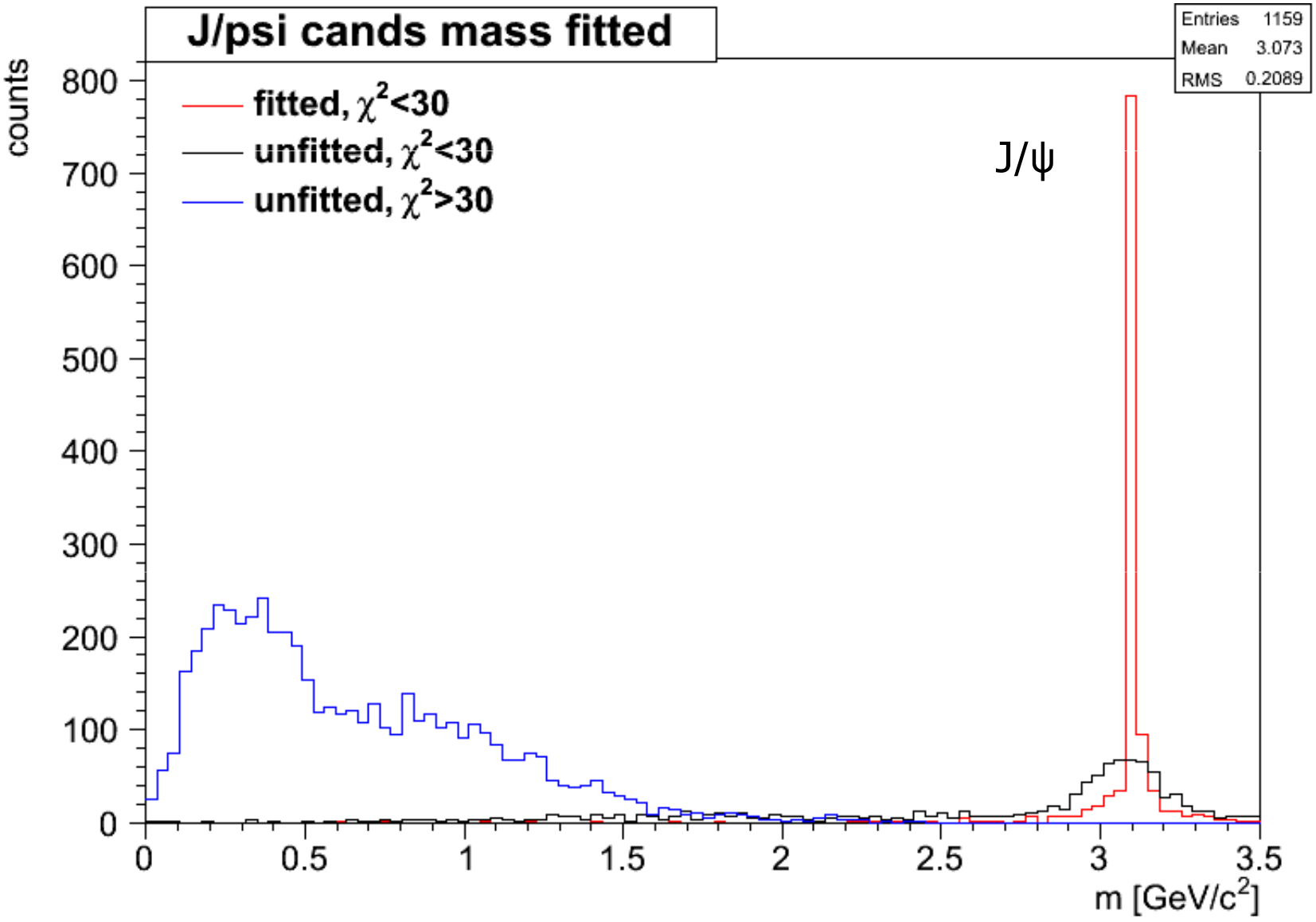
Channel $\psi(2S) \rightarrow J/\psi\ \pi^+\pi^-$, 4 constraint on $\psi(2S)$, 2k events





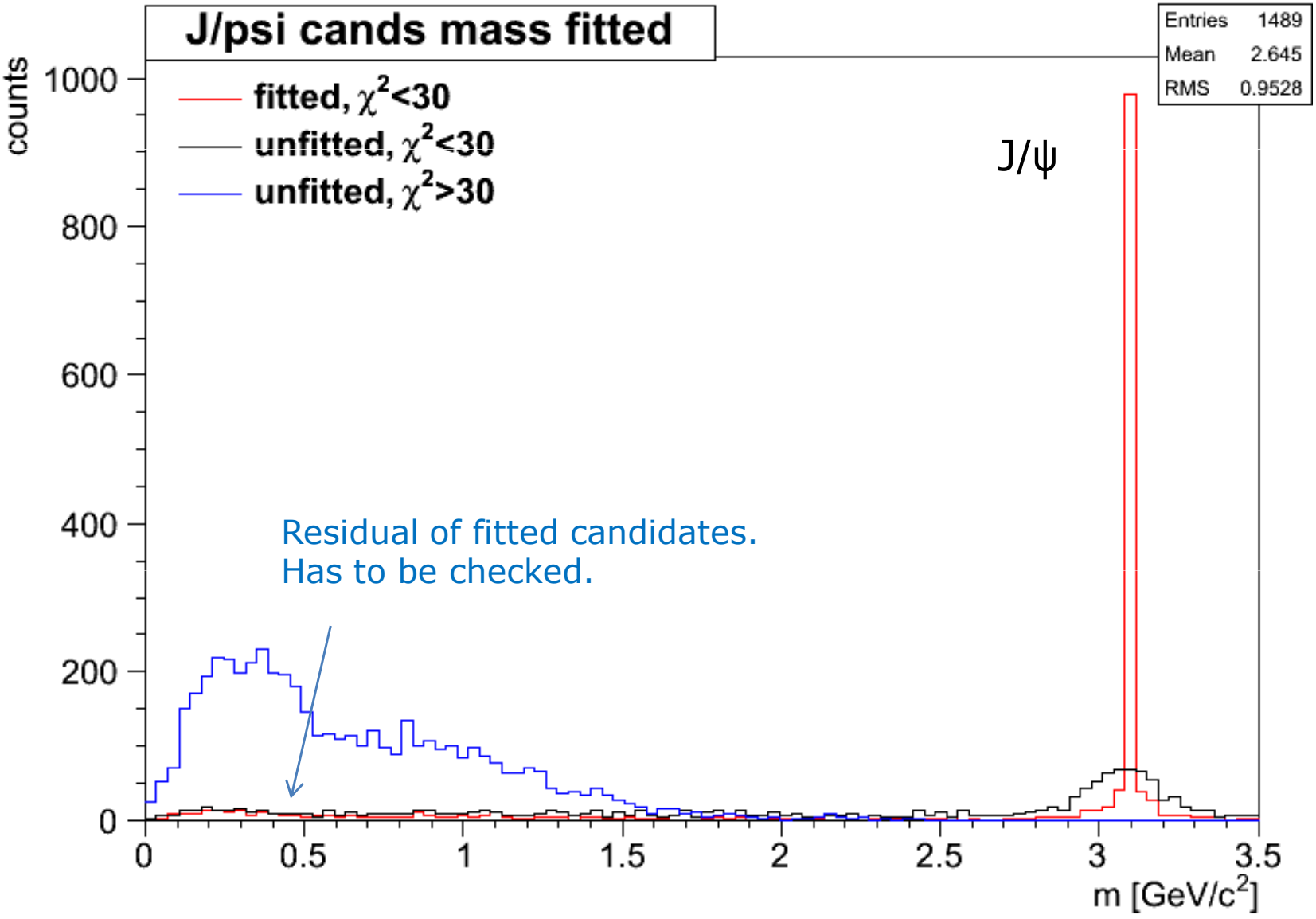
Fitting test – PndKinFitter (Mass-C.)

Channel $\psi(2S) \rightarrow J/\psi\ \pi^+\pi^-$, mass constraint J/ψ, 2k events



Fitting test – PndMassFitter (Mass-C.)

Channel $\psi(2S) \rightarrow J/\psi\ \pi^+\pi^-$, mass constraint J/ψ, 2k events



Further 2nd level tools in PandaROOT

Package rho/RhoBase

- TRho – (gRho) access to various global infos like TPdt, etc...
- TPdt – PDG particle table, *names as in EvtGen*
- TEventInfo – holds event information (beams, #neutral/chrgd, ...)
- TConstraint – holds fit constraints for a TCandidate

Package rho/RhoTools

- TTotalMomentum – total P4, P3, P, E, M, and corresponding missing quantities of a TCandList in Lab-S. and CMS
- TThrust, ThrustVector – thrust/thrustvector
- TSphericity, TMomTensor – sphericity, aplanarity, ...
- TFWMoments, TFoxWolfMom, TEventShape: Fox-Wolfram-Moments, etc.
- TLgndrMomFlow, TCones, TConeMomFlow – various energy flows
- ... and some more ...

Most not tested yet.

3rd level tools

Independent of PANDA/PandaROOT

- Partial Wave Analysis Software
  - Being developed by people from Bochum, GSI, KIT, Mainz
  - Wiki `http://panda-wiki.gsi.de/cgi-bin/view/PWA/WebHome`
  - Examples available

- Multivariate Analysis, ML Fitting Tools, sPlot

  All distributed with and available inside ROOT as
  - TMVA package (MV Analysis, signal/background separation)
  - RooFit Toolkit (Data Distribution Modelling)
  - TSPlot (ML fitting, signal/background separation)

Summary

- Data access works for multiple input files and stages
  - Splitting simulation into multiple jobs and do analysis on concatenated data works
- Basic analysis structures available, working smoothly
- Many analysis tools natively bundled with ROOT
- Various Selectors, Fitters and Kinematic Tools are available
- ToDo
  - validation still needed in particular for fitters
  - kinematic tools have to be tested
  - full propagation of PID info still missing
- But: Nothing serious keeping you from trying analysis!