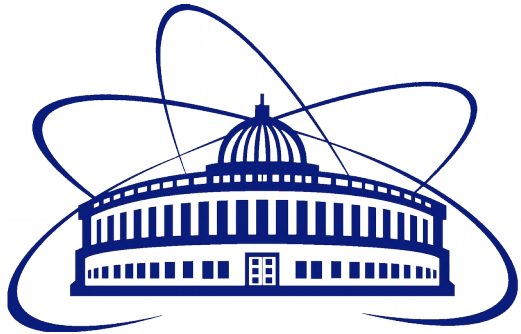


Implementation of Centrality Wagon for Global Polarization (Request 30)

Elizaveta Nazarova¹



23.03.2023

¹ Joint Institute of Nuclear Research, Dubna, Russia





- Location: `mpdroot/physics/evCentrality`
- Calculates centrality based on TPC multiplicity for each accepted event
 - Correspondence between centrality and multiplicity is taken from file (e.g. `nTr_Centr_Req30-PHSD.root`) → any plans to add the calculation inside?
- Returns centrality '-1' for rejected events (not included in the further analysis):
 - empty events
 - events with no vertex by TPC
 - events with reconstructed vertex $|z\text{-vertex-TPC}| > 130$ cm
 - events that failed to fire the FFD||FHCL trigger (assessed based on event track multiplicity using efficiency file)
- Event centrality is available for all other wagons in the train:
`event.getCentrTPC();`
- Example on how centrality variable is used in the analysis:
`mpdroot/physics/pairKK`

- **Implementing Centrality Wagon (and train framework) to the global polarization analysis (for Request 30)**

- `cd mpdroot/physics/polarization/macros`

➤ `root -b -q RunAnalyses.C`

```
void RunAnalyses (){  
    gROOT->LoadMacro("mpdloadlibs.C");  
    gROOT->ProcessLine("mpdloadlibs()");  
    MpdAnalysisManager man("ManagerAnal");  
    man.InputFileList("list.txt");  
    man.ReadBranches("*");  
    man.SetOutput("histos.root"); → needed or not?  
    MpdCentralityAll pCentr("pCentr","pCentr");  
    man.AddTask(&pCentr);  
    MpdGlobalPolarization pGlobalPol("pGlobalPol","pGlobalPol");  
    man.AddTask(&pGlobalPol);  
    man.Process();  
}
```

Input (config) file



output file





- For now to avoid recompilation:
- `root -b -q MpdAnalysisTask2.cxx MpdGlobalPolarization.cxx -e "runit(\"pGlobalPol\")`;
 - `MpdAnalysisTask2.cxx` → replaces the necessity of having separate Params files, they are defined in `MpdGlobalPolarization.h` and read in `MpdGlobalPolarization.cxx`

```
MpdGlobalPolarization::MpdGlobalPolarization(const char *name, const char *outputName) :
```

```
MpdAnalysisTask2(name, outputName)
```

```
{
```

```
  readParameters(name);
```

```
  param("mZvtxCut", mZvtxCut, 130.0);
```

```
  param("mNofHitsCut", mNofHitsCut, 10);
```

```
  param("mEtaCut", mEtaCut, 0.5);
```

```
  param("mPtminCut", mPtminCut, 0.1);
```

```
  param("mDcaCut", mDcaCut, 2.0);
```

```
  param("NITER_CENT", NITER_CENT, 4);
```

```
  param("NITER", NITER, 20);
```

```
  param("cent_cut_choice", cent_cut_choice, 0);
```

```
  param("cent_cut", cent_cut, 70.0);
```

```
  param("particle_choice", particle_choice, "Lambda");
```

```
}
```



- First test: MpdGlobalPolarization.cxx for MCTest analogously to what I did with the «old» version of centrality
- Global Polarization of Lambda (ALambda) from MC distributions, compared to the values obtained by fitting angular distributions of $\Delta\phi_p^* = \Psi_{RP}^1 - \phi_p^*$
- Compare with the results obtained with «old» centrality, which I did previously

- «Old» centrality:

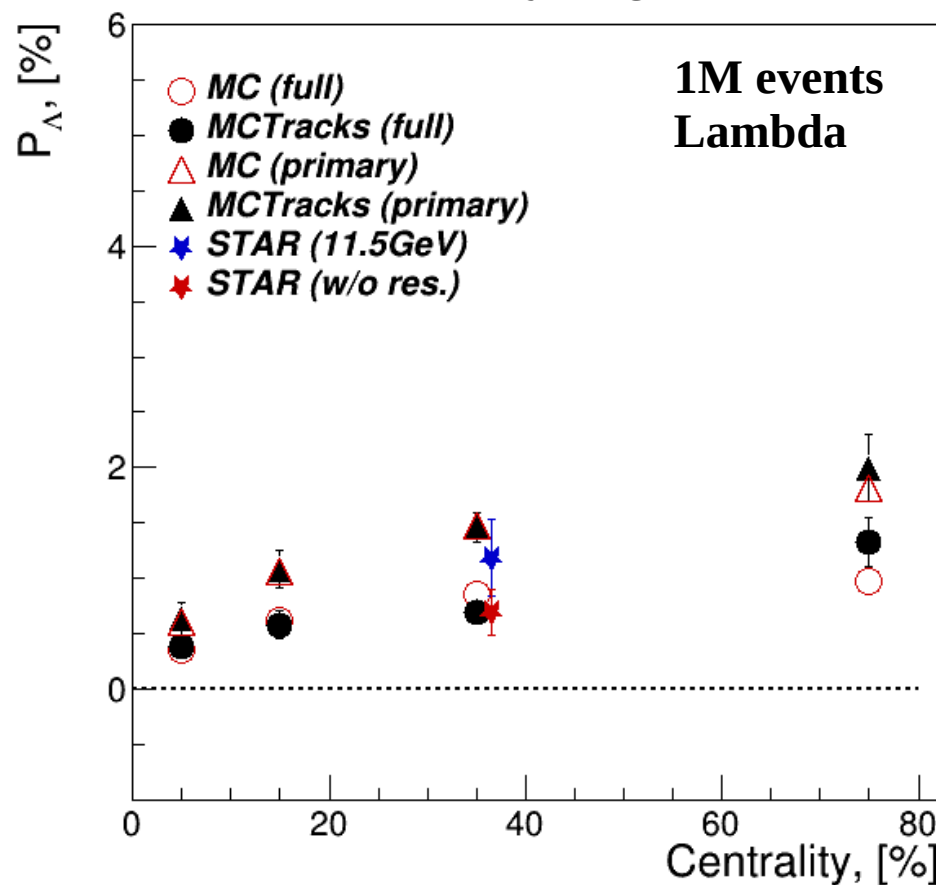
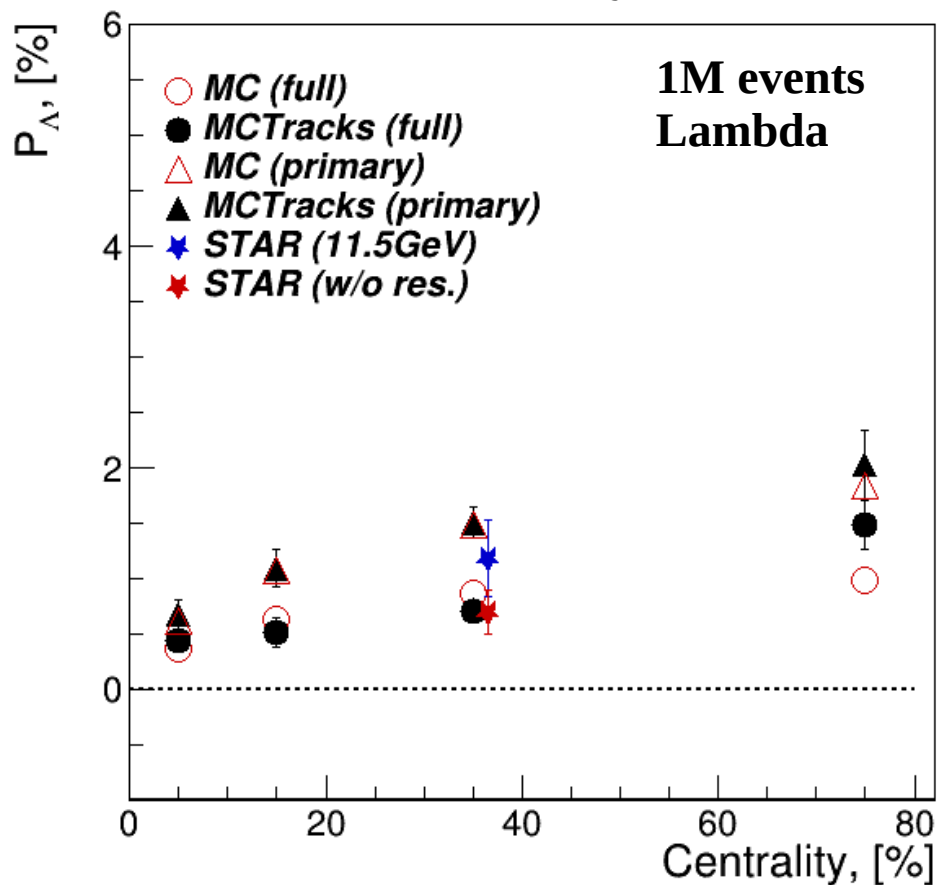
- MC-Glauber based on TPC
- $|\eta| < 1.0$
- $|p_T| > 0.15$ GeV
- $N_{\text{hits}} > 16$
- $|\text{DCA}| < 1$

- Centrality Wagon:

- $|\eta| < 0.5$
- $|p_T| > 0.1$ GeV
- $N_{\text{hits}} > 10$
- $|\text{DCA}| < 2$
- Cuts on empty events and vertex, trigger efficiency

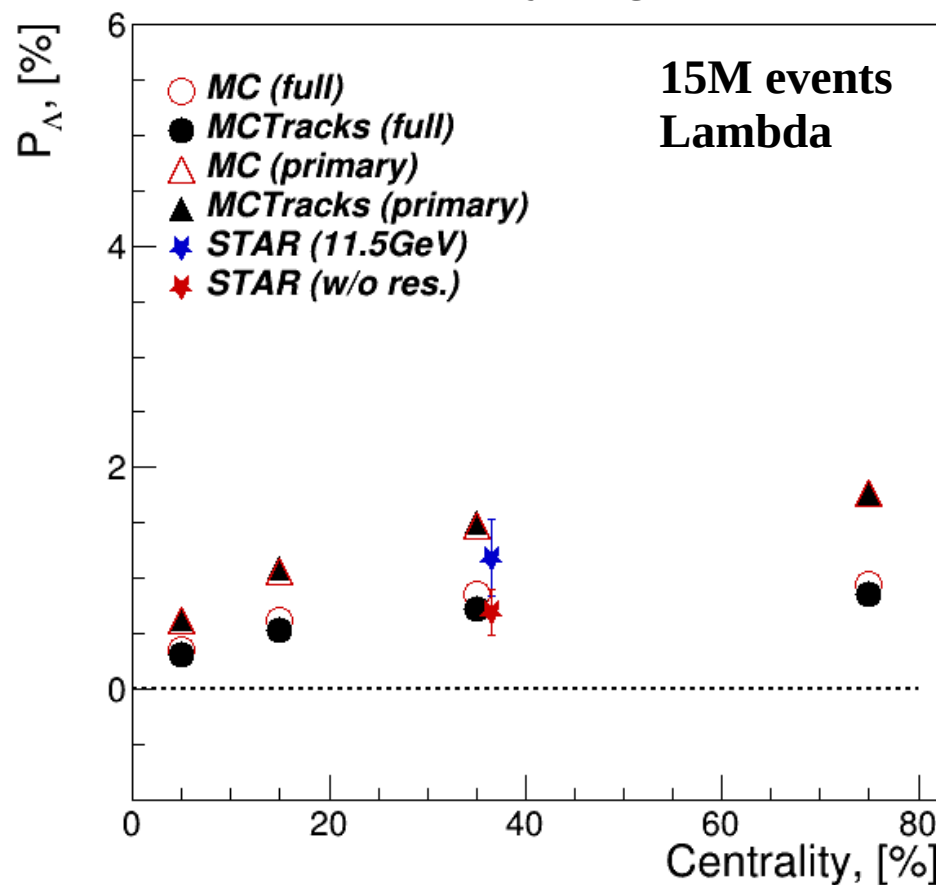
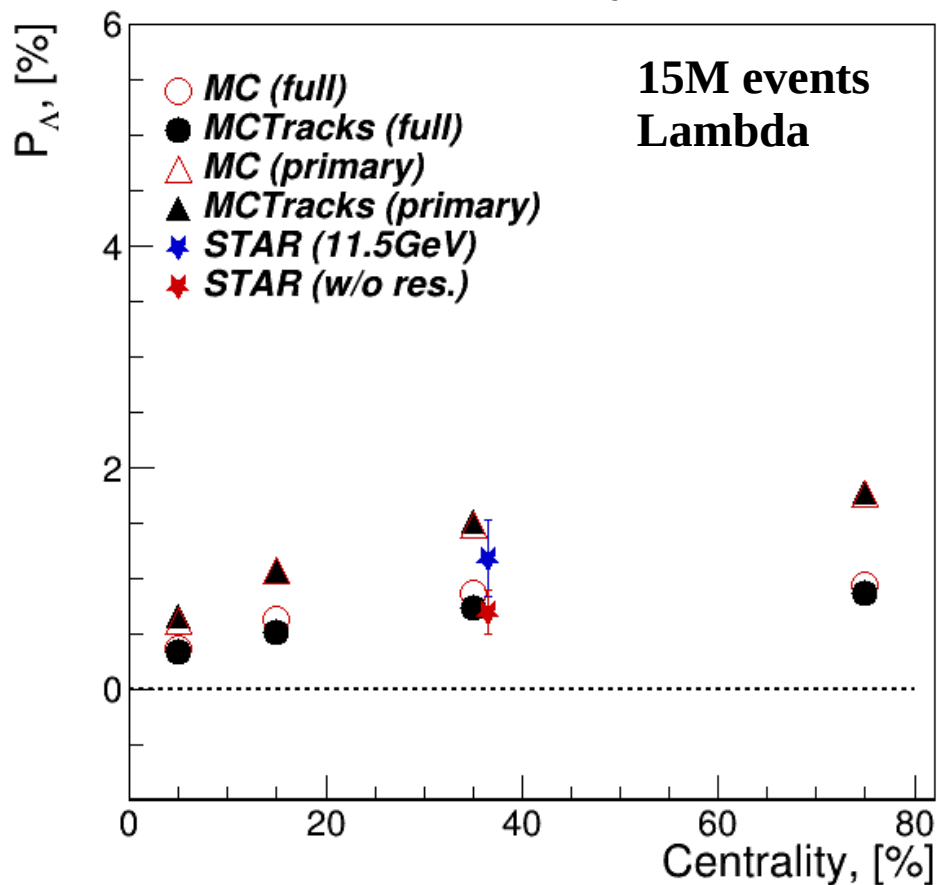
«Old» Centrality

Centrality Wagon



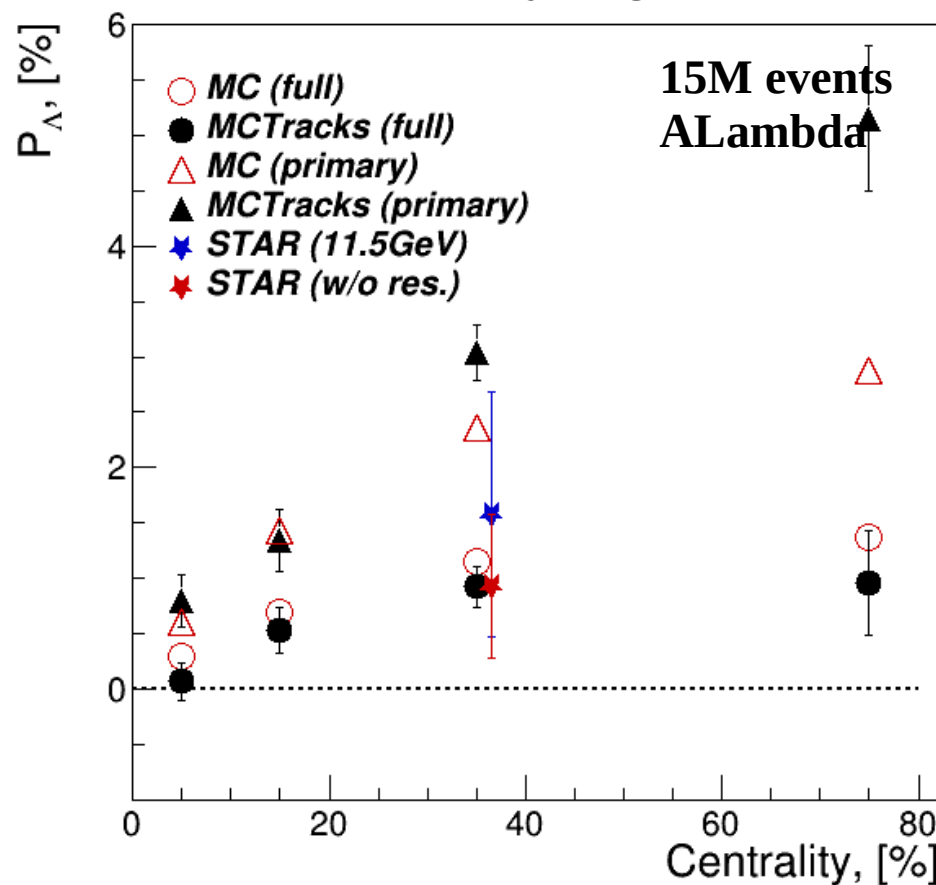
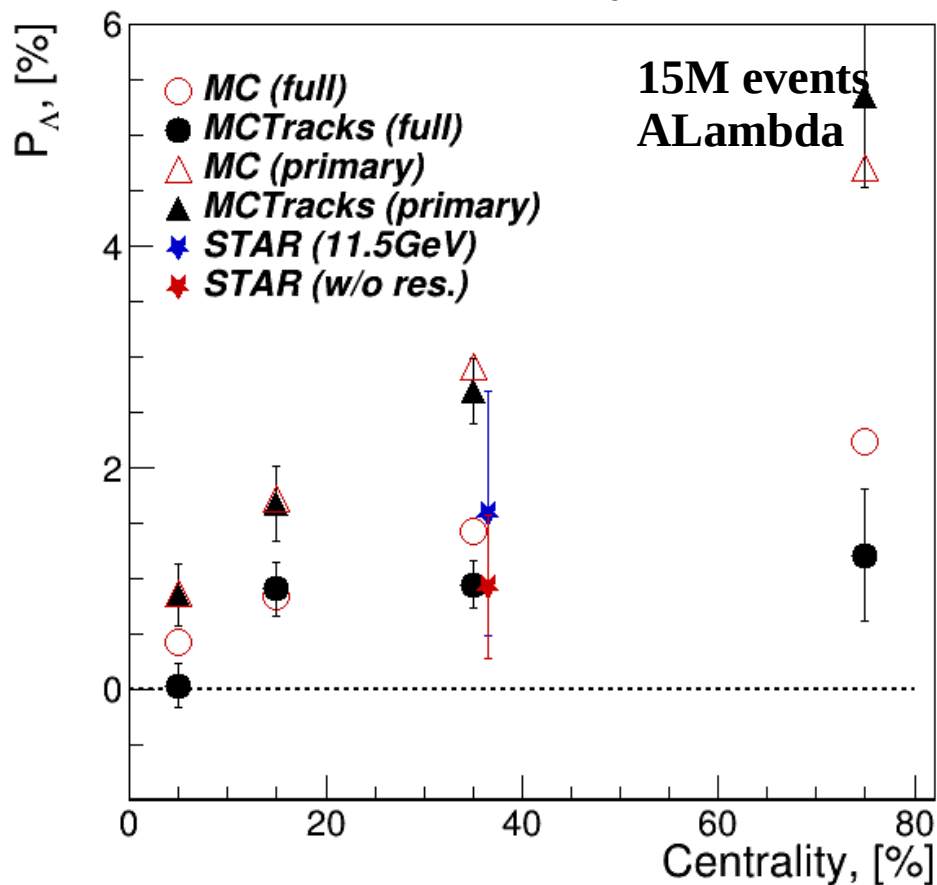
«Old» Centrality

Centrality Wagon



«Old» Centrality

Centrality Wagon





- Centrality Wagon works, so far without any problems
 - Is there a plan to calculate the centrality within it, or the plan is to use it like it is now — reading the conversion histograms from file?
- (First test) Inclusion of MCTest code for global polarization analysis is successful → now working on inclusion of full analysis
- Are there plans for internal parallelization of the train?
 - I have some global parameters in the code, which then will need to be rewritten
- Should we change to have readParameters in the AnalysisTask, to avoid having Param files for each analysis?



Thank you for your attention!