τ EDM

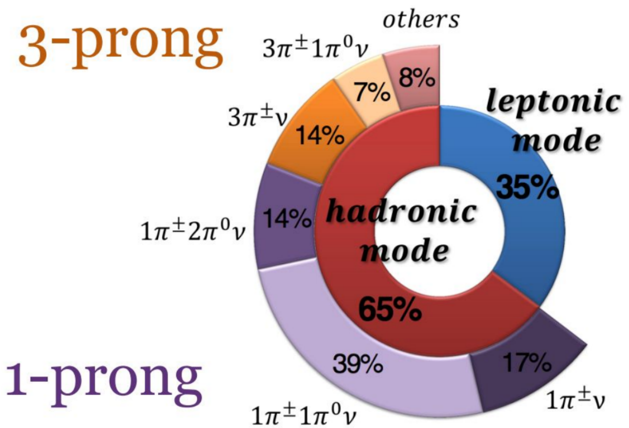
Michael Hübner, University of Bonn
23.10.2019
ATLAS Software Tutorial

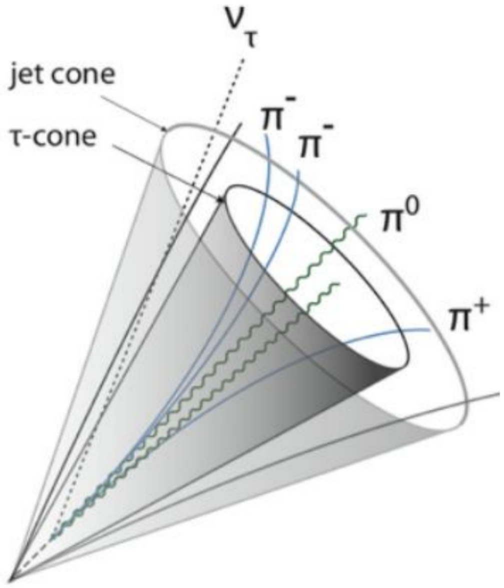
Thanks to Adam Bailey who allowed me to largely use his slides!

UNIVERSITÄT BONN

τ Leptons

- $m_\tau \approx 1.777$ GeV, $c\tau \approx 87$ µm
- 65% τ → hadrons:
  - main decay modes 1 or 3 $\pi^{+/-}$ and 0-2 π^0
- 35% τ → leptons:
  - e or µ and ν (see e and µ reco talks)





τ Reconstruction

- Seeded by AntiKt4LCTopo jet (will probably move to EM or PFlow jets)
- Determine τ vertex TJVA (Jet Vertex Finder-like)
- Associate tracks to τ-candidate
- Correct calorimeter cluster energy depending on N tracks and eta
- Calculate input variables for RNN discriminants; jet or τ, electron or τ
- Classify decay type and π^0 reconstruction
- Calibrate tau energy using calorimeter clusters and particle flow information

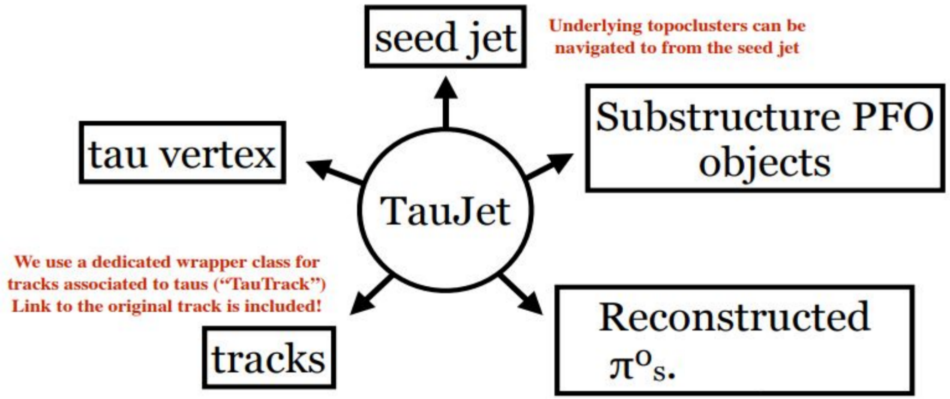
also constructs vars aimed at separating taus from background using tracks/clusters

Includes dedicated correction for pileup (nVtx)

Substructure and 5-way classification

τ EDM

- xAODTau on git:
  - v21.2: https://gitlab.cern.ch/atlas/athena/tree/21.2/Event/xAOD/xAODTau
- Main τ object is xAOD::TauJet, is an IParticle
  - Standard kinematics accessible via IParticle methods
  - τ-specific attributes via specific methods: tau ID bits, classification scores, τ-ID variables
  - All information encoded in enums in TauDefs.h
- ElementLinks to:



τ EDM

UNIVERSITÄT BONN

- τ 4-vector. Call with **xAOD::TauJet::p4**(TauCalibType):
  - Calo-based p_T, η, ϕ, m Original „robust“ calibration, will no longer be supported with recommendations!
  - MVA τ energy scale (TES) – ptFinalCalib → Multivariate calibration with better resolution, baseline for R21
- τ identification. Access with **xAOD::isTau**(isTauFlag):
  - **jetRNNSig**(VeryLoose/Loose/Medium/Tight)
  - EleBDT(Loose/Medium/Tight) is higher rejection alternative to EleOLR
    - Note: additional eBDT decorations added in AODFix
- τ details and ID variables. Access with **xAOD::detail**(Detail)
- ID scores. Access with **xAOD::discriminant**(TauID)
  - Can be useful for a deeper understanding of τs

τs in Analysis / TauAnalysisTools

UNIVERSITÄT BONN

- Main package is TauAnalysisTools
  (documentation in README.rst, displayed on git pages):
  - Dual-use – AthAnalysis and Athena
  - Select taus based on p_T, η, Ntracks, ID, electron veto, etc.
  - Apply efficiency and energy corrections + uncertainties
  - Truth matching (see xAOD::TruthParticleContainer) → Useful to ensure proper calculation of visible momentum etc.
- Example code can be found in TauAnalysisToolsExample
  (short extract on next slide)

Example – energy calibration

UNIVERSITÄT BONN

```
TauAnalysisTools::TauSmearingTool TauSmeTool( "TauSmearingTool" );
TauSmeTool.msg().setLevel( MSG::DEBUG );
CHECK(TauSmeTool.initialize());
std::vector<CP::SystematicSet> vSmearingSystematicSet;
for (auto SystematicsVariation : TauSmeTool.recommendedSystematics())
{
    vSmearingSystematicSet.push_back(CP::SystematicSet());
    vSmearingSystematicSet.back().insert(SystematicsVariation);
}
// here comes the part where you create a shallow copy of the taus
for ( auto xTau : *xTauShallowContainer.first )
{
    for (auto sSystematicSet: vSmearingSystematicSet)
    {
        CHECK( TauSmeTool.applySystematicVariation(sSystematicSet));
        CHECK( TauSmeTool.applyCorrection(*xTau) );
    }
}
```

Set up tool

Save recommended systematics in a vector

Loop over taus in shallow copy

Apply smearing

R21 Recommendations

UNIVERSITÄT BONN

Recommended way of using taus

- Taus in official ATLAS analysis
  - p_T > 20 GeV, |η| < 2.5 (excluding 1.37 < |η| < 1.52), 1 or 3 tracks (selection tool)
  - ID working points (selection tool, efficiency correction tool)
  - MVA TES (smearing tool)
- Use of decay mode information experimental (efficiency correction tool)
- Use of π^0 4-vectors soon (smearing tool)

Links

- Tau WG Twiki
- R21 Recommendations
- tauRec and tauRecTools
- τ xAOD Instructions
- ATLAS hadronic τ reconstruction paper
- τ substructure paper

Backup

Example – truth matching

UNIVERSITÄT BONN

```
TauAnalysisTools::TauTruthMatchingTool T2MT( "TauTruthMatchingTool");
T2MT.msg().setLevel( MSG::INFO );
CHECK(T2MT.setProperty("WriteTruthTaus", true));
CHECK(T2MT.initialize());
```

Set up tool

```
RETRIEVE(xAOD::TauJetContainer, xTauJetContainer, "TauJets");
std::pair< xAOD::TauJetContainer*, xAOD::ShallowAuxContainer* >
    xTauShallowContainer = xAOD::shallowCopyContainer(*xTauJetContainer);
CHECK( xEvent.copy("TauJets") );
```

Create shallow copy of tau container and copy taus

```
for ( auto xTau : *xTauShallowContainer.first )
```

Loop over taus in shallow copy

```
{
    auto xTruthTau = T2MT.getTruth(*xTau);
    if ((bool)xTau→auxdata<char>("IsTruthMatched"))
    {
        if (xTruthTau→isTau()) Info("Tau was truth matched");
    }
}
```

Do truth matching