

Iterative L3 Trigger Reconstruction

- Objectives and timeline:
 - Timeline TSG workshop at beginning of November (MUO POG) 01 Nov 2016
 - Timeline first month of 2017 (TSG) 31 Jan 2017
 - Final deadline for having integrated in CMSSW, March 2017. 31 Mar 2017
- Task list for the next milestone: 31 Jan 2017
- Design:
- OI design and testing
 - Optimization
 - Performance results:
- IO design and testing
 - Optimization
 - Making a ROC curve:
 - Performance studies
- Combination procedure
- Other issues and tests
 - efficiency drop for $\phi \sim 2$ rads.

Objectives and timeline:

Timeline TSG workshop at beginning of November (MUO POG) 01 Nov 2016

1. Need to have the new L3 reconstruction sequence implemented, following the previous work by Benjamin. DONE
2. Once is implemented, it is needed to compare it with the current L3 sequence. Minimal set of the studies for old/new L3 comparisons: DONE
 1. Efficiency versus η , ϕ , pT, PU DONE + vs d_{xy} and d_z in case of displaced muons DONE
 2. Timing (using Neutrino gun or HLT physics data, depending of what is recommended by STEAM now) DONE
 3. Track quality variable comparisons (Chi2, d_{xy} , ...) for DY and background like muons (QCDMuEnriched, HLTPHysics or ZeroBias if enough stat...) DONE
3. Comparison of the new L3 tracking with the one used in the Tk muon reco DONE

Timeline first month of 2017 (TSG) 31 Jan 2017

1. Optimize the new L3 reconstruction: DONE
 1. IO algorithm – ROI DONE
 2. OI algorithm DONE
2. More complete set of studies on performance comparisons using MuXX OR TkMuXX as a FOM, different signal samples should be tested in order to have all possible phase spaces covered: DY, ttbar, close-by-muon , displaced muons, low pT muon (from JPsi) PENDING
 1. Efficiency versus η , pT, PU, (for the early studies using DY and displaced muons) + vs d_{xy} and d_z in case of displaced muons PENDING
 2. Timing (using Neutrino gun or HLT physics data, depending of what is recommended by STEAM now) PENDING
 3. Rate comparisons for tracks passing a given pT threshold (QCDMuEnriched, HLTPHysics or ZeroBias if enough stat...) PENDING
 4. Track quality variable comparisons (Chi2, d_{xy} , ...) for DY and background like muons (QCDMuEnriched, HLTPHysics or ZeroBias if enough stat...) PENDING
3. Revise/design a smarter combination of the out-in and in-out sequence. ON-GOING
4. Ideas/prelim studies on the combination between new L3 and Tk muon reconstruction PENDING
5. Have already in place in CMSSW the new L3 sequence using current geometry. ON-GOING
 1. Including the new module which runs the TkMu path if the New L3 does not reconstruct the muon successfully. PENDING
 2. List of unused classes that could be removed from CMSSW when switching to the new L3. ON-GOING

Final deadline for having integrated in CMSSW, March 2017. 31 Mar 2017

1. Switch to new geometry and test everything still works. PENDING
2. Clean-up CMSSW by removing unused classes. PENDING

Task list for the next milestone: 31 Jan 2017

- Properly **define efficiency**: ON-GOING
 - removing all cuts from the filter** to avoid including the filter efficiency on the calculation.
 - Check **pattern recognition**, by shared hits (probably not on every sample, just a cross-check of goodness) ON-GOING
- Performance comparisons** for the between **Iterative L3 IO/OI** and **IOH/OIH/OIS**: ON-GOING

- efficiency vs eta, pT, PU, dxy, dz, chi2... ON-GOING
- Put a **combination** in place: ON-GOING
 - Understanding current under-performance: revise sequence. ON-GOING
- Performance** comparisons for the **IterL3 vs Cascade**: PENDING
 - efficiency vs eta, pT, PU, dxy, dz, chi2... PENDING
 - timing measurements. (follow TSG recipe) PENDING
 - rate comparisons/ (follow TSG recipe) PENDING
- Prepare a complete talk containing the full package ready to deliver PENDING
- Performance** comparisons for the **IterL3 vs Mu50 OR TkMu** PENDING
 - efficiency vs eta, pT, PU, dxy, dz, chi2...
 - timing measurements. (follow TSG recipe)
 - rate comparisons/ (follow TSG recipe)
- Based on that, **decide whether a combination procedure is needed** or if we can drop TkMu50 once for all. PENDING
 - repeat performance comparisons.

- Switch to new geometry** PENDING
- Clean-up CMSSW** by removing unused classes PENDING
 - List of unused classes that could be removed from CMSSW when switching to the new L3

Design:

A two-sequence algorithm based on iterative tracking is being implemented: (write here more details...)

OI design and testing

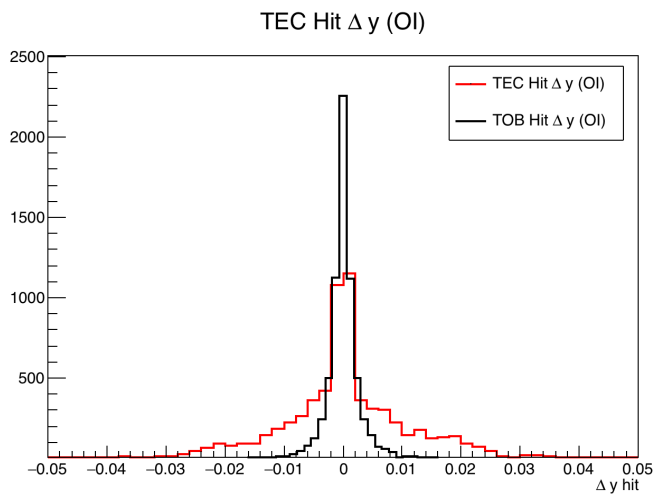
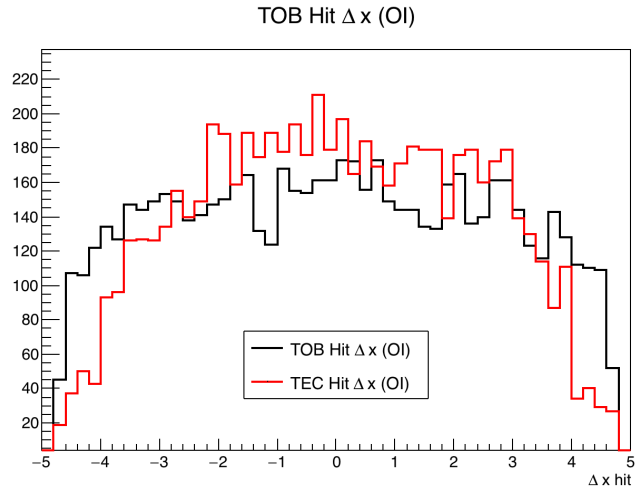
The outside-in algorithm consist of four steps:

1. Seed generation: [TSGforOI.cc](#)
2. TrajectoryCandidate: [CkfTrackCandidateMaker.cc](#)
3. TrackProducer: [TrackProducer.cc](#)
4. L3MuonProducer: [L3MuonProducer.cc](#)

The important step to optimise/study is the **seed generation**, current implementation allows for a great number of combinations varying the number of seeds, the rescaling factors and a number of parameters, this is the default configuration as of 15/11/2016:

```
UseHitlessSeeds = cms.bool(True),
UseHybridSeeds = cms.bool(False),
adjustErrorsDyanmicallyForHitless = cms.bool(False),
adjustErrorsDyanmicallyForHits = cms.bool(False),
estimator = cms.string('hltESPChi2MeasurementEstimator100'),
fixedErrorRescaleFactorForHitless = cms.double(5.0),
fixedErrorRescaleFactorForHits = cms.double(2.0),
pT1 = cms.double(13.0),
pT2 = cms.double(30.0),
pT3 = cms.double(70.0),
eta1 = cms.double(1.0),
eta2 = cms.double(1.4),
SF1 = cms.double(3.0),
SF2 = cms.double(4.0),
SF3 = cms.double(5.0),
SF4 = cms.double(7.0),
SF5 = cms.double(10.0),
hitsToTry = cms.int32(5),
layersToTry = cms.int32(3),
maxEtaForTOB = cms.double(1.2),
maxSeeds = cms.uint32(15),
minEtaForTEC = cms.double(0.8),
```

First thing checked is the distance between the generated seed compared to the reconstructed hit:



Several parameters can be tuned in the seed generation step:

- run hybrid / hit seeds
- number of seeds
- range of TOB, TEC
- hits/layers to try
- adjust errors dynamically or not
- rescaling factors

Optimization

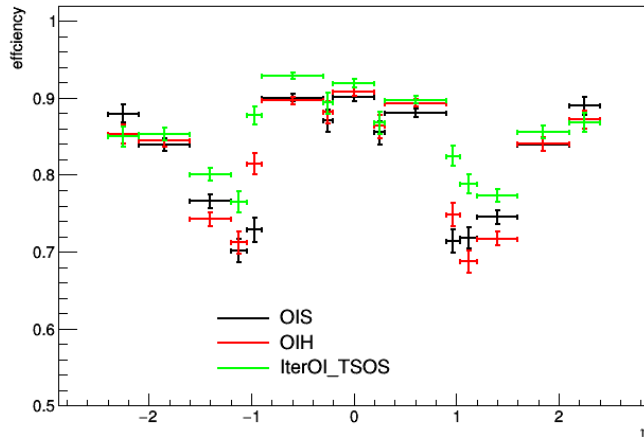
Here is a list of checks to optimise the algorithm:

- number of seeds
- range of the TOB, TEC
- once on TEC, use the stereo layers only instead of a random hit?
- hit/layers to try
- adjust errors dynamically
- rescaling factors.

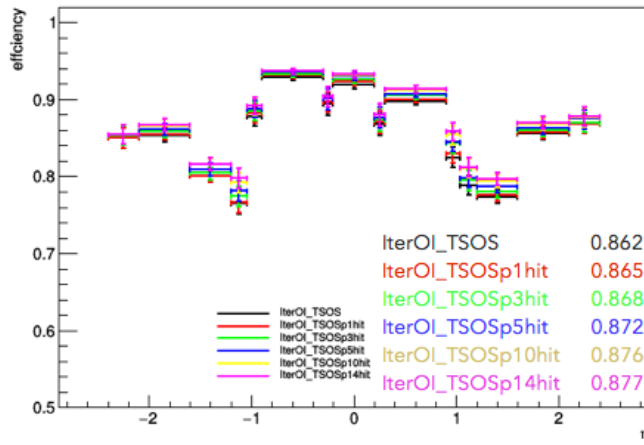
A number of checks have been **performed to optimised** the outside-in algorithm, after some basic tunes, we gain 3% w.r.t. the performance of OIS and OIH. All the plots here: https://folguera.web.cern.ch/folguera/private/MUO/161115/OI_Optimisation/

We check that increasing the number of seeds improves the efficiency, it somehow saturates at ~10 seeds (1 TSOS + 9 hit)

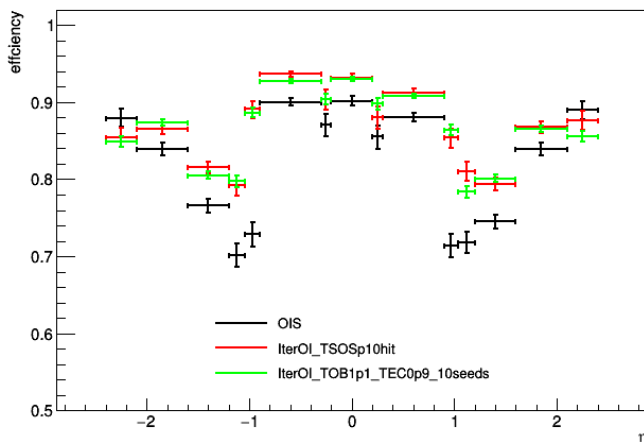
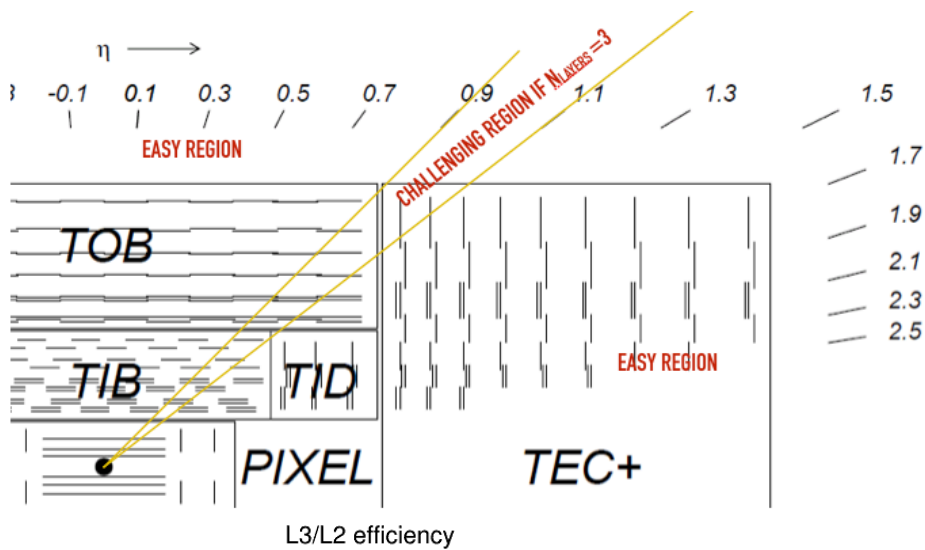
L3/L2 efficiency



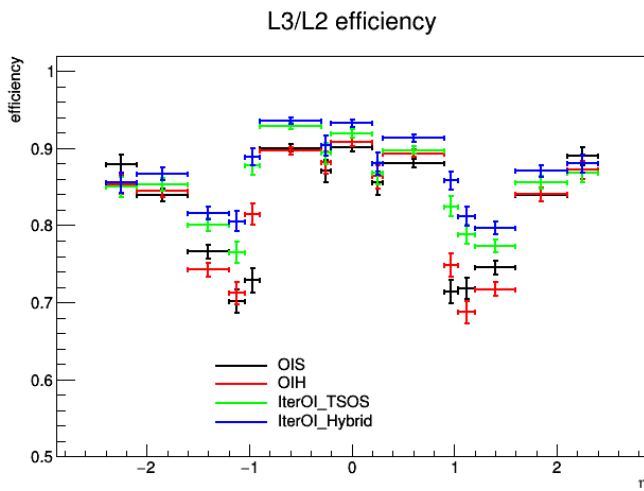
L3/L2 efficiency



Then we checked the overlap region between the TOB and the TEC. In this region, after looping over TOB the maximum number of seeds is re-initialised to 0 and the count starts in the TEC, we narrowed the region with little impact in the efficiency (0.3%), this narrowing will be faster but we still need absolute numbers for that:



We tried using what was called Hybrid seeds (now HitlessAndHitSeeds), which is deciding whether to add a hit seed depending on the parameters of the L2: 3% gain w.r.t using only the TSOS and 5% w.r.t old L3.



We'll now focus on the HitlessAndHitSeeds combination. Now we will try to find an optimal choice of (dynamical) rescaling errors.

Lastly we check if we can make use of stereo layer hits instead of mono-hits in the TEC.

Performance results:

IO design and testing

Taking all the parameters from the tracker community. Already in the current HLT menu, as it is used for the HLT_TkMu50 path:

```
fragment.HLTHighPt50TrackerMuonSequence = cms.Sequence( fragment.HLTDoLocalPixelSequence + fragment.
HLTDoLocalStripSequence + fragment.hltL1MuonsPt15 + fragment.HLTIterativeTrackingHighPtTkMu + fragment.
hltHighPtTkMu50TkFilt + fragment.hltHighPtTkMuons + fragment.hltHighPtTkMuonCands )
fragment.HLTIterativeTrackingHighPtTkMuIteration0 = cms.Sequence( fragment.hltPixelLayerTriplets +
fragment.hltIter0HighPtTkMuPixelTracks + fragment.hltIter0HighPtTkMuPixelSeedsFromPixelTracks +
fragment.hltIter0HighPtTkMuCkfTrackCandidates + fragment.hltIter0HighPtTkMuCtfWithMaterialTracks +
fragment.hltIter0HighPtTkMuTrackSelectionHighPurity )
fragment.HLTIterativeTrackingHighPtTkMuIteration2 = cms.Sequence( fragment.
hltIter2HighPtTkMuClustersRefRemoval + fragment.hltIter2HighPtTkMuMaskedMeasurementTrackerEvent +
fragment.hltIter2HighPtTkMuPixelLayerPairs + fragment.hltIter2HighPtTkMuPixelSeeds + fragment.
hltIter2HighPtTkMuCkfTrackCandidates + fragment.hltIter2HighPtTkMuCtfWithMaterialTracks + fragment.
hltIter2HighPtTkMuTrackSelectionHighPurity )
fragment.HLTIterativeTrackingHighPtTkMu = cms.Sequence( fragment.
HLTIterativeTrackingHighPtTkMuIteration0 + fragment.HLTIterativeTrackingHighPtTkMuIteration2 + fragment.
hltIter2HighPtTkMuMerged )
```

Optimization

Things to be checked:

- Test extreme cases to make sure things work Unknown User (sfolguer): large/tiny region using TTBar or WJets
- Produce a ROC curve efficiency vs timing
- Define the working point and close this topic.

Need to optimise the definition of the ROI: the current choice seems to match no other choice in the HLT, check if it is the best choice:

```
MasterMuonTrackingRegionBuilder = cms.PSet(
  Rescale_eta = cms.double( 3.0 ),
  Rescale_phi = cms.double( 3.0 ),
  Rescale_Dz = cms.double( 4.0 ), #Normally 4
  EscapePt = cms.double( 3.0 ), #Normally 1.5 but it should be at least 8 for us
  EtaR_UpperLimit_Par1 = cms.double( 0.25 ), #Normally 0.25
  EtaR_UpperLimit_Par2 = cms.double( 0.15 ), #Normally 0.15
  PhiR_UpperLimit_Par1 = cms.double( 0.6 ), #Normally 0.6
  PhiR_UpperLimit_Par2 = cms.double( 0.2 ), #Normally 0.2
  UseVertex = cms.bool( False ), #Normally False
  Pt_fixed = cms.bool( False ), #Normally True
  Z_fixed = cms.bool( False ), #True for IOH
  Phi_fixed = cms.bool( True ), #False for IOH
  Eta_fixed = cms.bool( True ), #False for IOH
  Pt_min = cms.double( 3.0 ), #Is 0.9 for Tau; normally 8 here
  Phi_min = cms.double( 0.1 ),
  Eta_min = cms.double( 0.1 ),
  DeltaZ = cms.double( 24.2 ), #default for tau: 24.2, for old IOH: 15.9
  DeltaR = cms.double( 0.025 ), #This changes for different iterations. for old IOH: ?
  DeltaEta = cms.double( 0.04 ), #default 0.15
  DeltaPhi = cms.double( 0.15 ), #default 0.2
  maxRegions = cms.int32( 2 ),
  precise = cms.bool( True ),
  OnDemand = cms.int32( -1 ),
  MeasurementTrackerName = cms.InputTag( "hltESPMeasurementTracker" ),
  beamSpot = cms.InputTag( "hltOnlineBeamSpot" ),
  vertexCollection = cms.InputTag( "pixelVertices" ), #Warning: I am not generating colleciton. Vertex
is off anyway
  input = cms.InputTag( 'hltL2Muons', 'UpdatedAtVtx' )
)
```

```
fragment.HLTSetMuonTrackingRegionBuilder8356 = cms.PSet(
```

```
  Rescale_eta = cms.double( 3.0 ),
  Rescale_phi = cms.double( 3.0 ),
  Rescale_Dz = cms.double( 3.0 ),
  EtaR_UpperLimit_Par1 = cms.double( 0.25 ),
  EtaR_UpperLimit_Par2 = cms.double( 0.15 ),
  PhiR_UpperLimit_Par1 = cms.double( 0.6 ),
```

```

PhiR_UpperLimit_Par2 = cms.double( 0.2 ),
UseVertex = cms.bool( False ),
Pt_fixed = cms.bool( False ),
Z_fixed = cms.bool( True ),
Phi_fixed = cms.bool( False ),
Eta_fixed = cms.bool( False ),
Pt_min = cms.double( 1.5 ),
Phi_min = cms.double( 0.1 ),
Eta_min = cms.double( 0.1 ),
DeltaZ = cms.double( 15.9 ),
DeltaR = cms.double( 0.2 ),
DeltaEta = cms.double( 0.2 ),
DeltaPhi = cms.double( 0.2 ),
maxRegions = cms.int32( 2 ),
precise = cms.bool( True ),
OnDemand = cms.int32( -1 ),
MeasurementTrackerName = cms.InputTag( "hltESPMeasurementTracker" ),
beamSpot = cms.InputTag( "hltOnlineBeamSpot" ),
vertexCollection = cms.InputTag( "pixelVertices" ),
input = cms.InputTag( 'hltL2Muons','UpdatedAtVtx' )
)

```

Making a ROC curve:

Efficiency (WJets) vs Timing (HLTPysics)

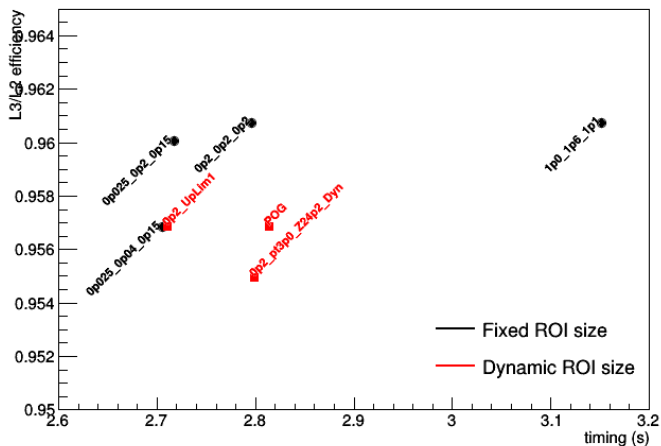
```

## Efficiency:
ssh -X lxplus
/afs/cern.ch/user/f/folguera/workdir/MUO/CMSSW_8_0_19/src/MuonHLT/MuonHLTDebugger/test/ROI
  cmsRun myHLT_Mu50_withL3_IO_0p6_0p6_0p6_Fix.py &>myHLT_Mu50_withL3_IO_0p6_0p6_0p6_Fix.log&

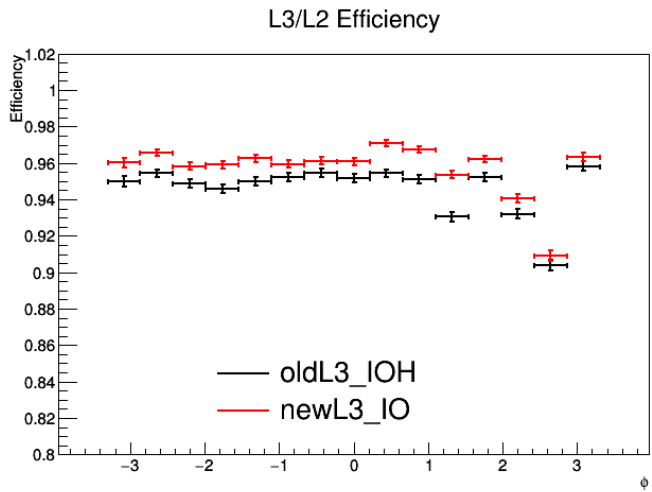
## Timing:
ssh -X vocms003
cd /data/user/folguera/CMSSW_8_0_19/src/TimingScripts/test/ROI
nohup taskset -c 1 cmsRun myHLT_Mu50_withL3_IO_0p2_0p2_0p2_Fix.py >&
myHLT_Mu50_withL3_IO_0p2_0p2_0p2_Fix.log&

```

ROI optimization



A drop in efficiency around 2 radians have been observed in both algorithms:



We