

# Offline Electron Seeding Validation - Update

**Caleb Fangmeier**  
Ilya Kravchenko, Greg Snow

University of Nebraska - Lincoln

EGM Reco/Comm/HLT meeting — June 22, 2018

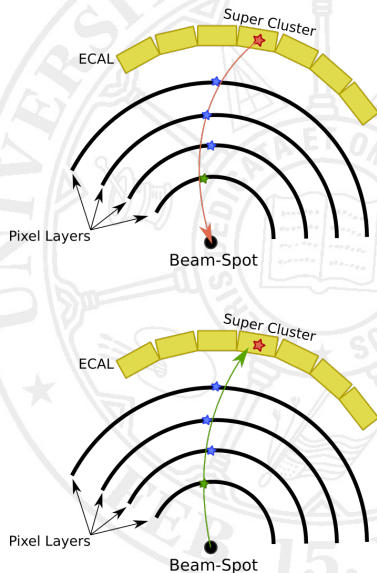


## INTRODUCTION

- ▶ Our goal is to study **seeding** for the **offline** GSF tracking with the **Phase I pixel detector**.
- ▶ Specifically, we want to optimize the new pixel-matching scheme from HLT for use in off-line reconstruction.
- ▶ This Talk:
  - ▶ Explain “Hit Skipping” and demonstrate effects on seeding performance
  - ▶ Compare performance with pileup added

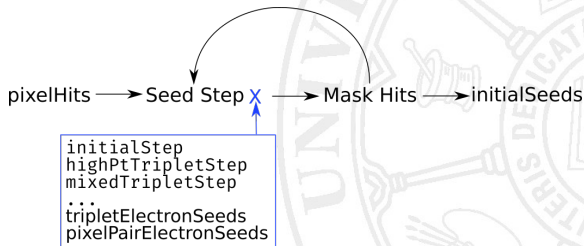
# N-HIT ELECTRON SEEDING

1. Using the beam spot, the SC position, and SC energy, propagate a path through the pixels.
2. Require the first hit to be within a  $\delta\phi$  and  $\delta z$  window. ( $\delta\phi$  and  $\delta R$  for FPIX)
3.  $\delta z$  window for first hit is huge as SC and beam spot positions give very little information about  $z$ .
4. Forget the SC position, and propagate a new track based on the vertex and first hit positions, and the SC energy.
5. Progress one-by-one through the remaining hits in the seed and require each one fit within a specified window around the track.
6. Quit when all hits are matched, or a hit falls outside the window. **No skipping is allowed.**



# HIT-SKIPPING

- ▶ Normally, general tracking seeds are made with an iterative procedure
- ▶ Each iteration masks hits from use in subsequent steps
- ▶ Reduces combinatorics for CKF tracking



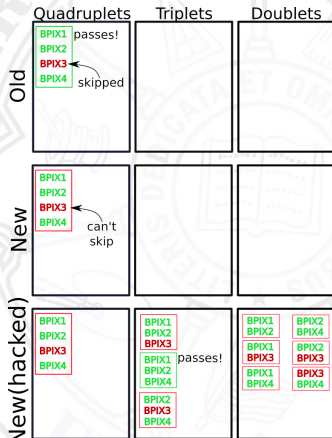
- ▶ Each iteration works on a specific set of combinations of layers. eg. `initialStep` uses:

BPix1+BPix2+BPix3+BPix4  
BPix1+BPix2+BPix3+FPix1\_pos  
BPix1+BPix2+BPix3+FPix1\_neg  
BPix1+BPix2+FPix1\_pos+FPix2\_pos

BPix1+BPix2+FPix1\_neg+FPix2\_neg  
BPix1+FPix1\_pos+FPix2\_pos+FPix3\_pos  
BPix1+FPix1\_neg+FPix2\_neg+FPix3\_neg

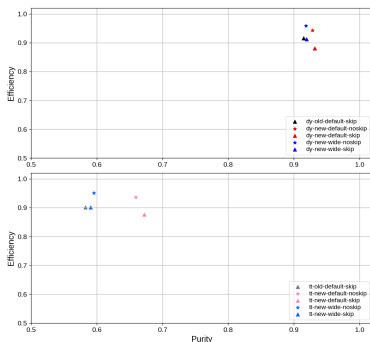
# HIT-SKIPPING

- ▶ When `NHitElectronSeedProducer` was implemented for HLT, hit skipping was not added.
- ▶ Consider an example configuration where we are generating first quadruplet, then triplet, and then finally doublet seeds, masking hits along the way.
- ▶ If we require at least 3 matched hits, the old method *with* hit skipping would create a seed of hits `BPIX1`, `BPIX2`, `BPIX3`.
- ▶ But new method *without* hit skipping wouldn't make any seed from these hits.
- ▶ The "hack" is to create seeds using only steps `tripletElectronSeeds`, and `pixelPairElectronSeeds` with **no masking**.
- ▶ Adding skipping and removing the hack would reduce cpu time from redundant seeds.



## HIT-SKIPPING - PERFORMANCE

- ▶ Enabling hit skipping and removing hack reduces number of seeds by 35% to 50%.
- ▶ 3-5x fewer seeds with respect to old seeding
- ▶ Efficiency reduced by between 4% and 6% to align more with old seeding performance.
- ▶ Purity improved by between about 1%.
- ▶ (table in backup)



Process	Seeding Method	$\langle N_{seeds} \rangle$ (no-skip)	$\langle N_{seeds} \rangle$ (with-skip)	Percent Reduction
<i>tt</i>	Old - default settings	-	12.69	-
	New - HLT settings	4.40	2.56	41%
	New - "wide" settings	7.28	4.65	36%
Drell-Yan	Old - default settings	-	11.40	-
	New - HLT settings	4.70	2.32	51%
	New - "wide" settings	5.38	2.65	51%

## ADDING PILEUP

- ▶ The simhit-rechit linkage that was previously used in efficiency/purity measurements is not saved in GEN-SIM-RAW.
- ▶ Therefore, the DIGI step was re-run, but only for the signal event.
- ▶ However, running this instead of the RAW2DIGI step discarded the previously mixed pileup in the RAW.
- ▶ So even though there is a PileupInfo collection with reasonable values, there's no actual pileup hits being used for tracking (caused quite some confusion for me).
- ▶ In the end, abandon simhit-rechit linkage and just use  $\Delta R$  matching for efficiency/purity.
- ▶ Some technical details in backup.

## CONCLUSIONS & OUTLOOK

- ▶ TODO





BACKUP



## DEFINITIONS

- ▶ **Sim-Track** - A track from a simulated electron both originating from the luminous region of CMS (beam-spot  $\pm 5\sigma$ ) and having  $|\eta| < 3.0$ .
- ▶ **ECAL-Driven Seed** - A seed created via a matching procedure between Super-Clusters and General Tracking Seeds (Either from `ElectronSeedProducer` or `ElectronNHitSeedProducer`). Must have  $HOE < 0.15$ .
- ▶ **GSF Track** - A track from GSF-Tracking resulting from an **ECAL-Driven Seed**
- ▶ **GSF Tracking Efficiency** - The fraction of **Sim-Tracks** that have a matching **GSF Track** (based on  $\Delta R$  matching)
- ▶ **GSF Tracking Purity** - The fraction of **GSF Tracks** that have a matching **Sim-Track**
- ▶ **GSF Tracking Fake Rate** - The fraction of nontruth-matched Super-Clusters which result in at least one **GSF Track**.

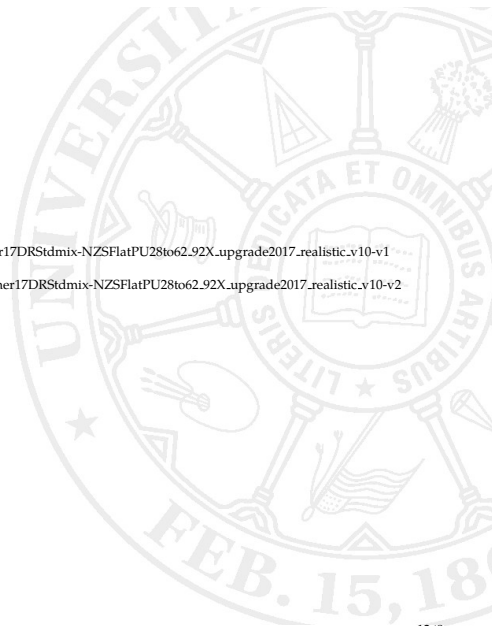
## MATCHING WINDOW PARAMETERS

		narrow	default (HLT)	wide	extra-wide
Hit 1	dPhiMaxHighEt	<b>0.025</b>	<b>0.05</b>	<b>0.1</b>	<b>0.15</b>
	dPhiMaxHighEtThres	20.0	20.0	20.0	20.0
	dPhiMaxLowEtGrad	-0.002	-0.002	-0.002	-0.002
	dRzMaxHighEt	9999.0	9999.0	9999.0	9999.0
	dRzMaxHighEtThres	0.0	0.0	0.0	0.0
	dRzMaxLowEtGrad	0.0	0.0	0.0	0.0
Hit 2	dPhiMaxHighEt	<b>0.0015</b>	<b>0.003</b>	<b>0.006</b>	<b>0.009</b>
	dPhiMaxHighEtThres	0.0	0.0	0.0	0.0
	dPhiMaxLowEtGrad	0.0	0.0	0.0	0.0
	dRzMaxHighEt	<b>0.025</b>	<b>0.05</b>	<b>0.1</b>	<b>0.15</b>
	dRzMaxHighEtThres	30.0	30.0	30.0	30.0
	dRzMaxLowEtGrad	-0.002	-0.002	-0.002	-0.002
Hit 3+	dPhiMaxHighEt	<b>0.0015</b>	<b>0.003</b>	<b>0.006</b>	<b>0.009</b>
	dPhiMaxHighEtThres	0.0	0.0	0.0	0.0
	dPhiMaxLowEtGrad	0.0	0.0	0.0	0.0
	dRzMaxHighEt	<b>0.025</b>	<b>0.05</b>	<b>0.1</b>	<b>0.15</b>
	dRzMaxHighEtThres	30.0	30.0	30.0	30.0
	dRzMaxLowEtGrad	-0.002	-0.002	-0.002	-0.002

NHit Seeding window parameters. Bold designates modified values.

## SAMPLES

- ▶ /ZToEE\_NNPDF30\_13TeV-powheg\_M.120\_200/RunII/Summer17DRStdmix-NZSFlatPU28to62.92X\_upgrade2017\_realistic\_v10-v1
- ▶ /TT\_TuneCUETP8M2T4\_13TeV-powheg-pythia8/RunII/Summer17DRStdmix-NZSFlatPU28to62.92X\_upgrade2017\_realistic\_v10-v2



## CMSDRIVER INVOCATIONS

### RAW→Step2 (old)

```
cmsDriver.py RAW2Step2 \  
  --mc \  
  --conditions 92X_upgrade2017_realistic_v10 \  
  --era Run2_2017 \  
  --eventcontent FEVTDEBUG \  
  --datatier GEN-SIM-DIGI-RAW \  
  --step DIGI:pdigi_valid,L1,DIGI2RAW \  
  --geometry DB:Extended \  
  --filein file:input.root \  
  --fileout file:step2.root \  
  --runUnscheduled
```

### Step2→TrackingNtuple (old)

```
cmsDriver.py Step2ToTrackingNtuple \  
  --mc \  
  --conditions 92X_upgrade2017_realistic_v10 \  
  --era Run2_2017 \  
  --eventcontent RECO SIM,MINIAODSIM,DQM \  
  --datatier GEN-SIM-RECO,MINIAODSIM,DQMIO \  
  --step RAW2DIGI,L1Reco,RECO,EI,PAT,VALIDATION:@standardValidation+@miniaodValidation \  
  --filein file:step2.root \  
  --fileout file:trackingNtuple.root \  
  --customise Validation/RecoTrack/customiseTrackingNtuple.customiseTrackingNtuple \  
  --runUnscheduled
```

### RAW→TrackingNtuple (new)

```
cmsDriver.py RAW2TrackingNtuple \  
  --mc \  
  --conditions 92X_upgrade2017_realistic_v7 \  
  --era Run2_2017 \  
  --eventcontent FEVTDEBUG \  
  --datatier GEN-SIM-RECO \  
  --step RAW2DIGI,RECO,EI,PAT,VALIDATION \  
  --customise Validation/RecoTrack/customiseTrackingNtuple.customiseTrackingNtuple \  
  --filein file:input.root \  
  --fileout file:trackingNtuple.root \  
  --runUnscheduled
```