Photon Conversion in sPHENIX

*Background*

Photon conversion is an interaction of a photon with matter. It produces an electron positron pair. This is a well studied phenomenon. It apples to sPHENIX in a few significant ways. First it is not difficult the create a material map of the detector using reconstructed photon conversions. The number of photon conversions is in direct correspondence with the density of the material. More importantly, it is a significant source of background for prompt electron measurements and photons measurements (1). It would be ideal to remove the electrons produced by conversions and reconstruct photons. The reconstructed photons can then be used for pi0 decay reconstruction which removes much of the pi0 background from the prompt photon measurement. There already exists two established methods for reconstructing conversions in calorimeter/tracker systems (2 pg115). The outside-in and the inside-out methods. They refer to the way in which candidates are identified and then cut. In the inside-out method every track begins as a candidate track and then cuts are applied using tracker info until the conversion candidates are identified and calorimeter information is finally used to decide which candidates will be accepted. The outside-in method is a similar idea but uses calorimeter information to first cut the tracks down and then uses tracking information.

*Truth motivation*

The long term goal is to identify and reconstruct photon conversions in data. Prior to this analysis my group (Nagle-Perpelitsa) was unsure of how to do this, so we decided to investigate the existing capabilities of sPHENIX simulation using truth tools and then plan for how to make similar observations using reconstruction data. Developing a truth package enables us to create a precise baseline to eventually compare the reconstruction observations to. It also provides an easy way to make an initial proof of concept framework that outlines how photons will be reconstructed.

*Truth Infrastructure*

The first task for my analysis was to identify what the existing capabilities of sPHENIX secondary vertexing and tracking system were for tagging photon conversions in truth and reconstructing them in single photon events. I did this part of the analysis with the inside-out method.

**



*Photon conversion observations for sPHENIX simulation of single photon events uniformly distributed 5-30GeV with a fixed trivial primary vertex and acceptance rapidity of 1. The decrease in reco vertex efficiency at larger R is due to lack of silicon hits. The efficiency loss at higher pT is consistent with other studies.*

I then made a proof of concept for material mapping using photon conversions in sPHENIX.

**

*Truth location of photon conversions in sPHENIX.*

There was no existing sPHENIX framework for reconstructing conversions in events with a background. In events with a background such as pp it is not feasible to do secondary vertexing on every track combination. Therefore, it is advantageous to use the outside-in method. This complicates the problem as it is difficult to identify a photon conversion without a vertex to point to. The truth framework address this issue by checking every electron to see if its parent is a photon. Any electron whose parent is a photon is tagged as a conversion. The daughter electrons are grouped together by matching the conversion radius. As far as developing a solely truth frameworks goes, this is sufficient. The mother, daughters, and vertex of the conversion have been identified which allows almost all significant parameters of a conversion to be recorded.

To develop a reconstruction framework much more is required. In data, we will not have the privilege of checking all the particles for who their mothers are, but the truth framework is still useful for developing the reco framework. The reconstruction framework will have to go over the tracks and pair them to make secondary vertices. I believe this problem can be done in O(nlogn) time. As stated, it is not feasible to check every track pair therefore it will be necessary to cut the number of tracks that are checked. ATLAS has already developed and tuned cuts for doing this, but the tunings are dependent on the geometry of the detector (2 pg 126). Additionally, new or different cuts may perform better in sPHENIX, but the ATLAS variables are a good starting point. I developed my truth package to not only identify conversions, but record relevant parameters calculated by the existing reconstruction framework for both signal and background events. It will be straightforward to further develop and tune the cuts for reconstruction using the data recorded by the truth package. Distribution of the variables I have selected are included at the end of this report.



*Ratio of reconstructed conversion tracks to truth conversion particles as a function of pT. Ntruth=20,000*

*Reco outline*

I have already created an outline for the conversion reconstruction package in code. Briefly, the process is to go over all tracks and reject as much background as possible. This includes applying all track level cuts such as initial pT and rapidity, electric charge of the track, and any additional single track cuts that will be developed. Then matching the track to an EM cluster and applying cuts to the cluster. I have not explored any cuts on the cluster aside from the EM prob. I believe tracks with no associated EM cluster should be rejected. Note that in about 15% of the events both tracks will associate with the same cluster while in the rest of the events the clusters are distinguishable. It may be beneficial to make a separate set of cuts for these two types of events. Each track that passes these cuts should go through the remaining tracks and check if the pair passes the pairwise cuts. The most important one will likely be the delta eta, but other pairwise cuts should also be explored. It remains to be studied if one track is likely to pass the pairwise cuts with multiple tracks.

Pairs that pass the pairwise cuts are then considered vertex candidates. Currently, sPHENIX does not have an algorithm for creating a vertex from 2 tracks. It has the RaveVertex system which processes all the tracks in an event. ATLAS describes a method for estimating a vertex using a candidate, and cuts that can be applied during the vertexing process (pg 127). ATLAS also developed a separate vertexing algorithm specifically for conversion vertices because different assumptions can be made than if the mother particle was say a pion (pg 127). I’m not sure a separate algorithm will be necessary in sPHENIX. Once the vertex for the vertex candidate is calculated vertex level cuts such as radius and chi2 should be applied. If a vertex passes these cuts it is considered a conversion candidate and very likely is actually a conversion. A Photon can be reconstructed from this conversion using 4-vector addition, and cuts can be applied to the photon’s invariant mass and pT. It may be the case that sufficient purity is achieved earlier in the cut process. In that case, subsequent cuts should be turned off as the efficiency will go down.

Once the full reco package is developed its performance should be compared to the truth package.

*Variables and Background*

Cut Variables: The following plots show the distribution of signal and background for several of the variables that maybe useful to cut on in the reco package. The plots are made by running TMVA on the TTrees created by the truth package. The background in these events is sampled by the truth package. Each truth particle is marked as either a conversion participant or background. The presented variables are calculated from the associated tracks except for the two vertex (vtx) variables because the reconstruction vertexing script is not ready yet.

* Track\_pT – a track level cut
* Track\_layer – a track level cut, the first tracking layer that the track has a hit in
* Cluster\_prob – a track level cut, the EM probability of the associated cluster. Note that truth EM particles have uniform distribution. The spike at zero is partially described by events where the two tracks from a conversion associate to the same cluster
* Track\_deta – a pair level cut, the delta eta between a track pair
* Track\_dlayer – a pair level cut, the difference in track\_layer for the two tracks
* Vtx\_radius – a vertex level cut, the distance in x,y space of the secondary vertex candidate to the trivial vertex
* vtxTrack\_dist – a vertex level cut, distance from the vertex to the first hit for either track

No photon level cuts are included because I did not find variables that gave sufficient discriminating power and were independent from the existing variables. Note vtxTrack\_dist, track\_layer, and vtx\_radius are all significantly correlated.

