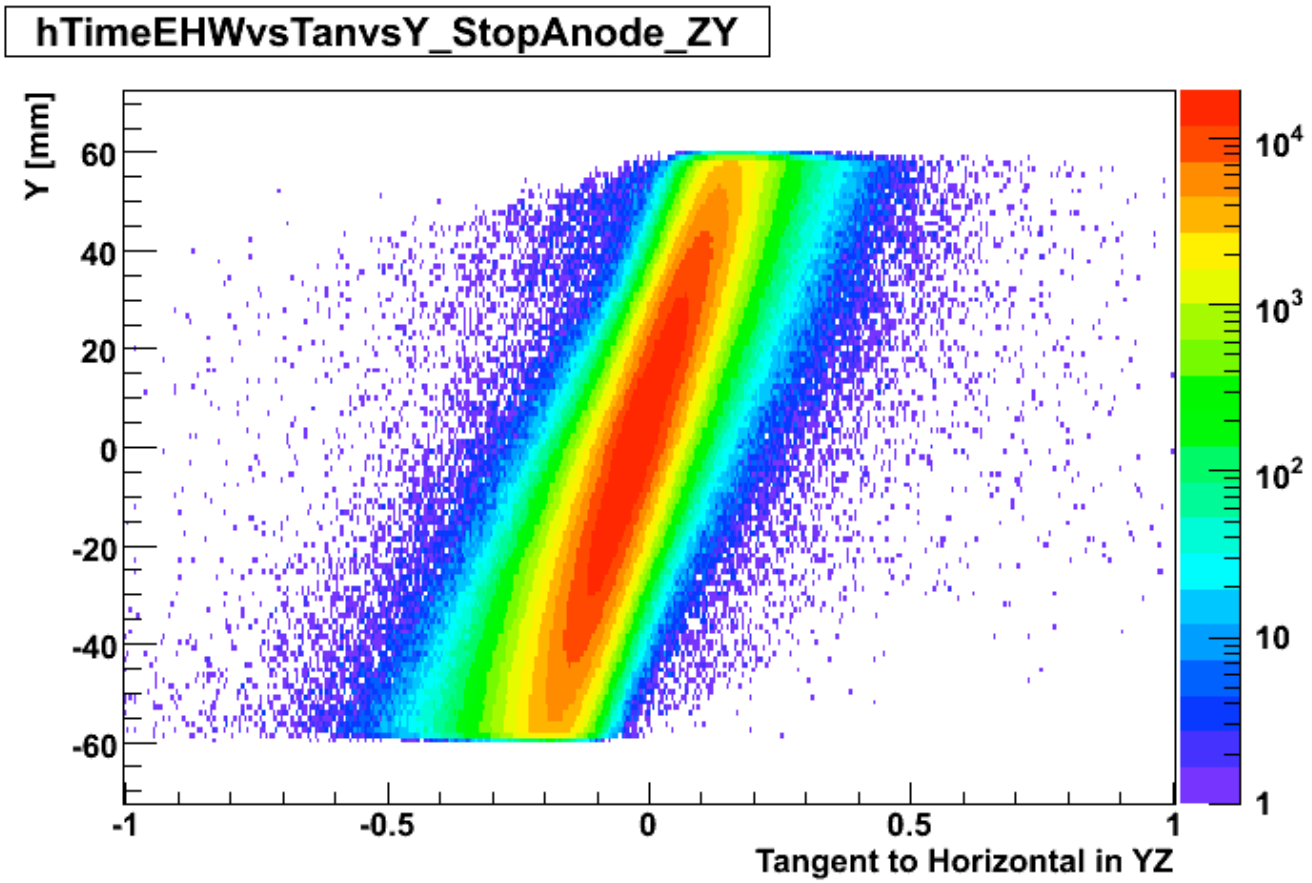
MuCap 10/22/2008: An update on the muon stop EH pulse width studies for muon stops

Sara Knaack

For events which were tracks, good stops, and accepted as best entrances I obtained…

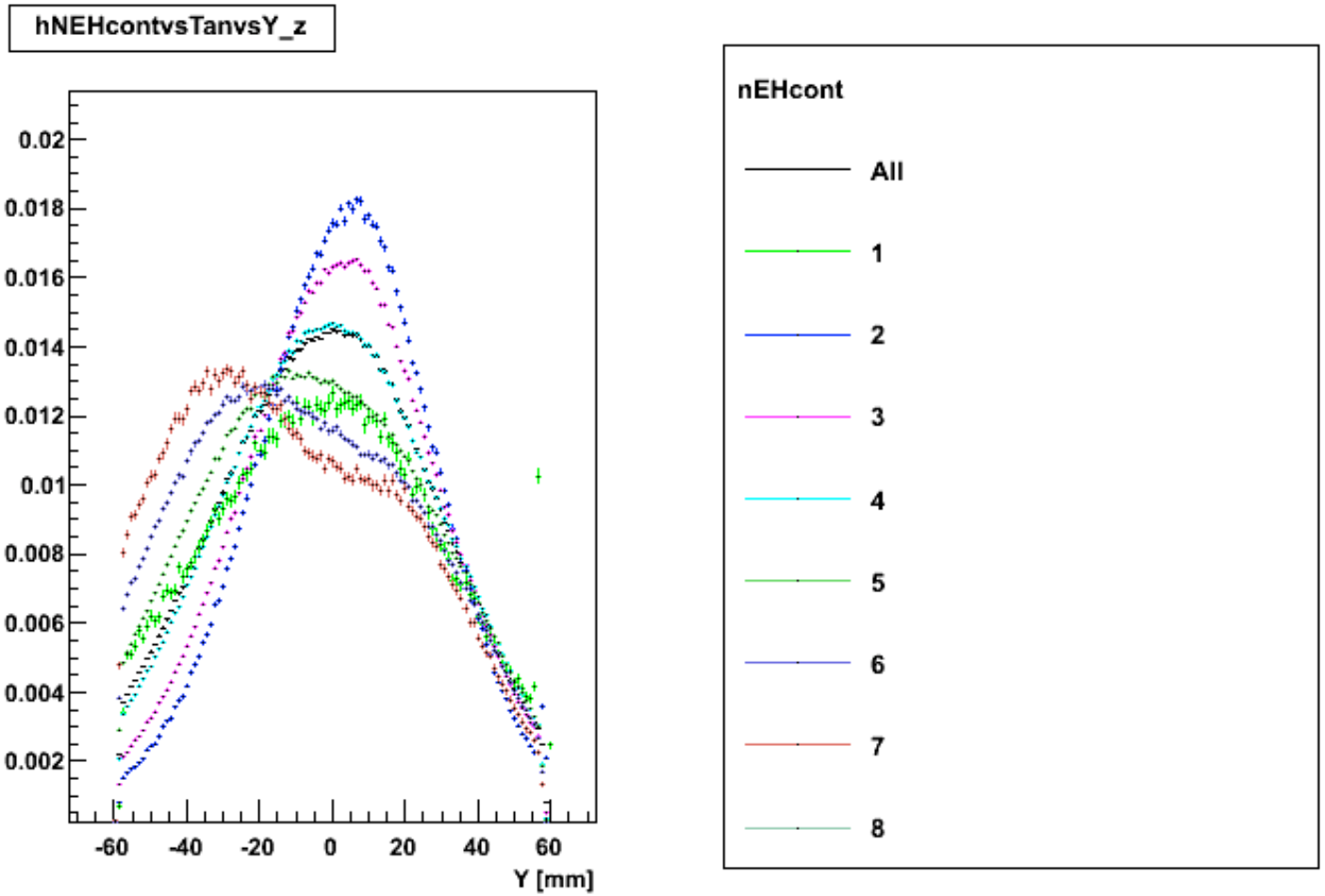
- **the number of continuous EH threshold pixels on the stop anode in the time dimension,**
- **and the number of continuous EH pixels in the anode dimension (the nEHcont).**
- **(16 files of 20 ppm Ar data; 2.5×10^7)**



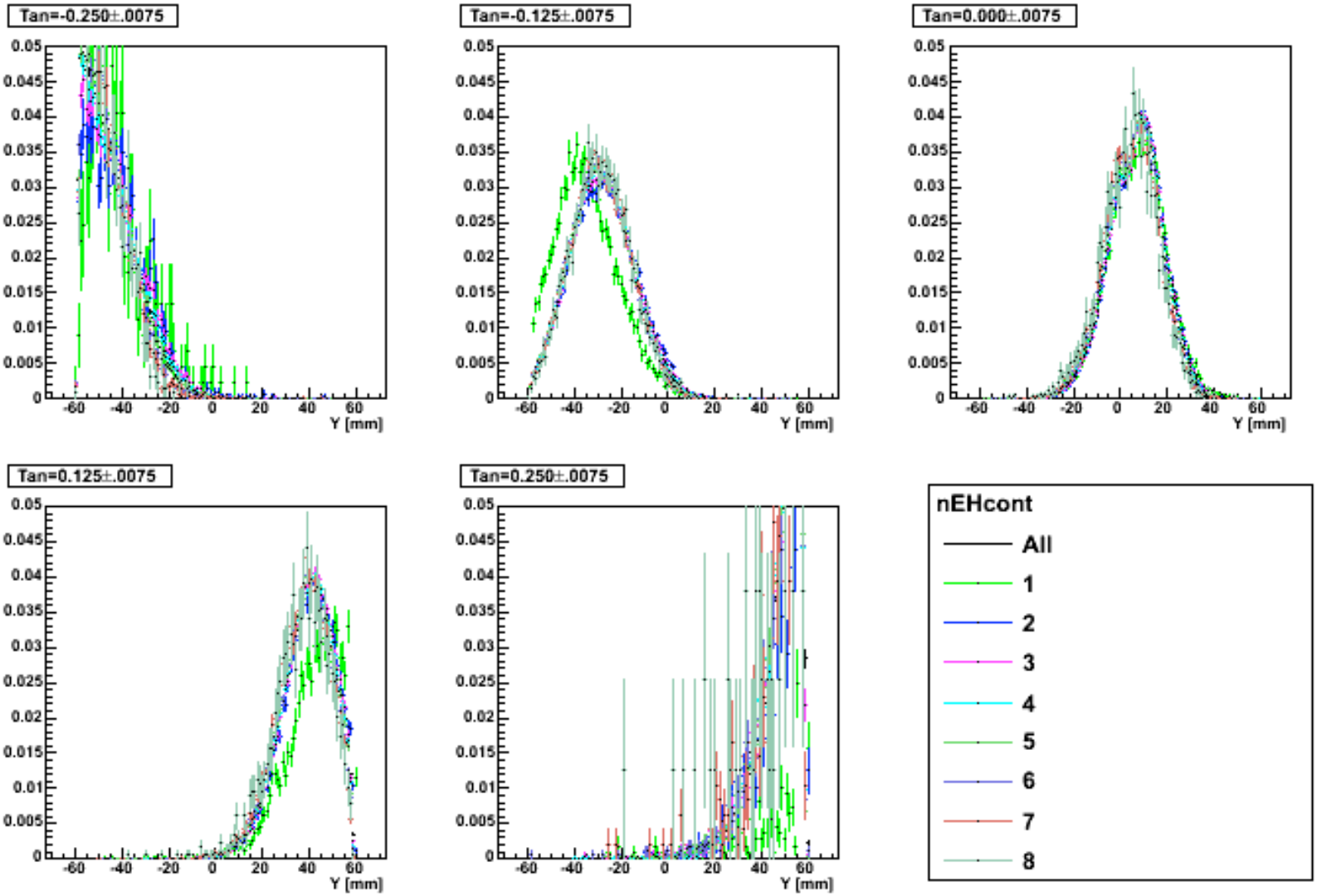
The slope of the muon stop track is obtained from the TOneLine line fit. This yields the Y/Z slope in TPC pixel space, and hence the tan of the track angle relative to the horizontal; ie. in the YZ plane of the TPC.

```
//filling the tangent issues
 Double_t tan = 1/(4/(200*mutrack_parameters.drift_speed)) * muStop->GetOneLine()->Getb();
 Double_t ang = TMath::ATan(tan);
  Float_t stopanode = muStop->GetStopAnode();
  Double_t EH_Width=0;
  UChar_t *T_EH = pixel_list->GetTAnode_EH();
  UChar_t *W_EH = pixel_list->GetWAnode_EH();
  Int_t nEH = pixel_list->GetNAnode_EH();
  Int_t nEHcont = muStop->GetNPixelsEHcont();
  //printf("Studying new muStop with %d nEHcont\n",nEHcont);
  for(Int_t a=stopanode;a>=stopanode-nEHcont+1;a--){
    Double_t t_min;
    Double_t t_max;
    Bool_t first = 1;
    Double_t last;
    //printf("Starting on anode %d\n",a);
    for(Int_t p=0;p<nEH;p++){
      if(W_EH[p]==a){
                //printf("T_EH[%d]=%f on anode %d.\n",p,(double)T_EH[p],(int)W_EH[p]);
                //start the tally for making sure the pixels are continuous in Y by noting the first pixel on this anode of the
```

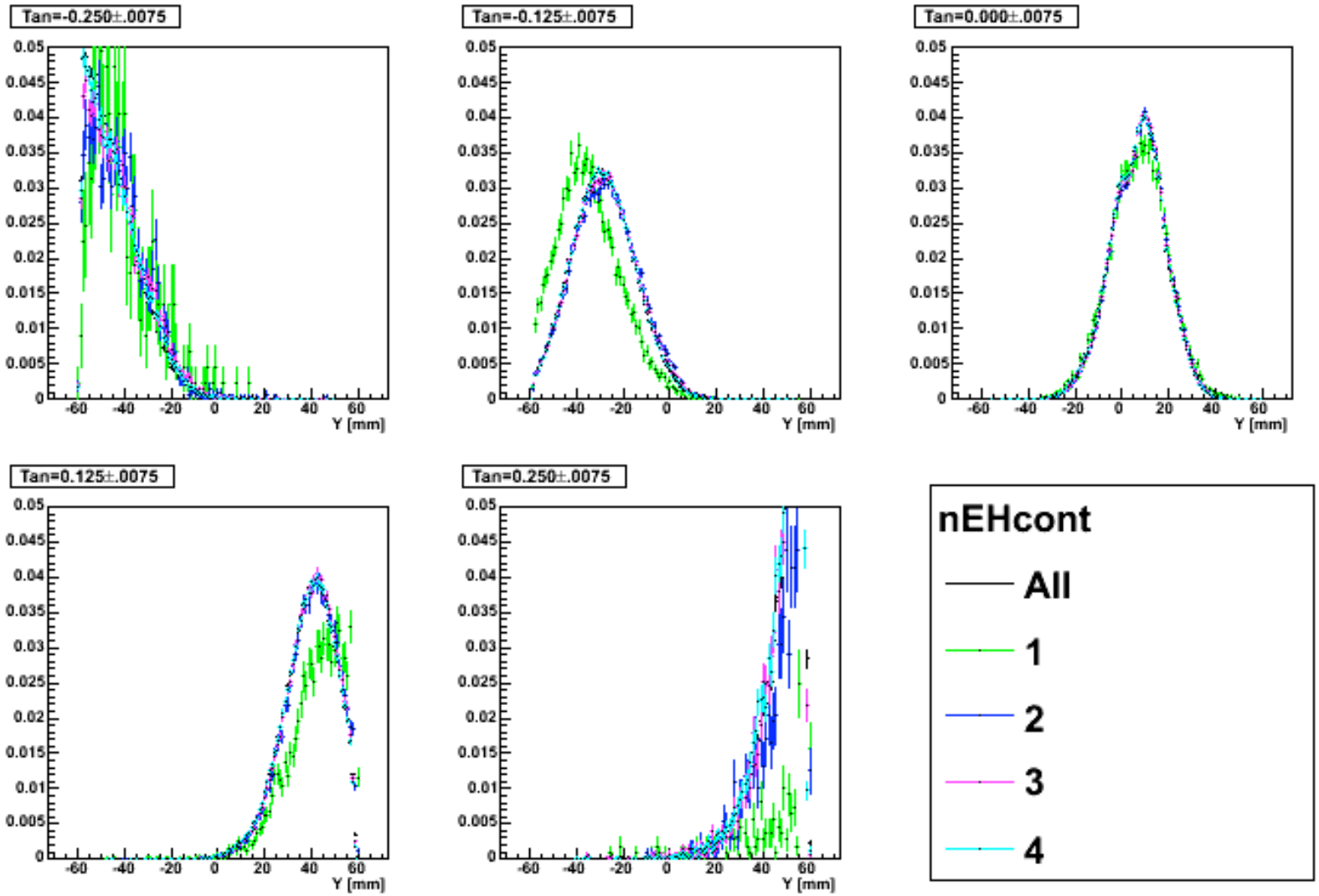
Just to recap… we observe an overall variation in the Y distribution of the muon stops vs. the continuous number of anodes at the end of the track which have the EH threshold.



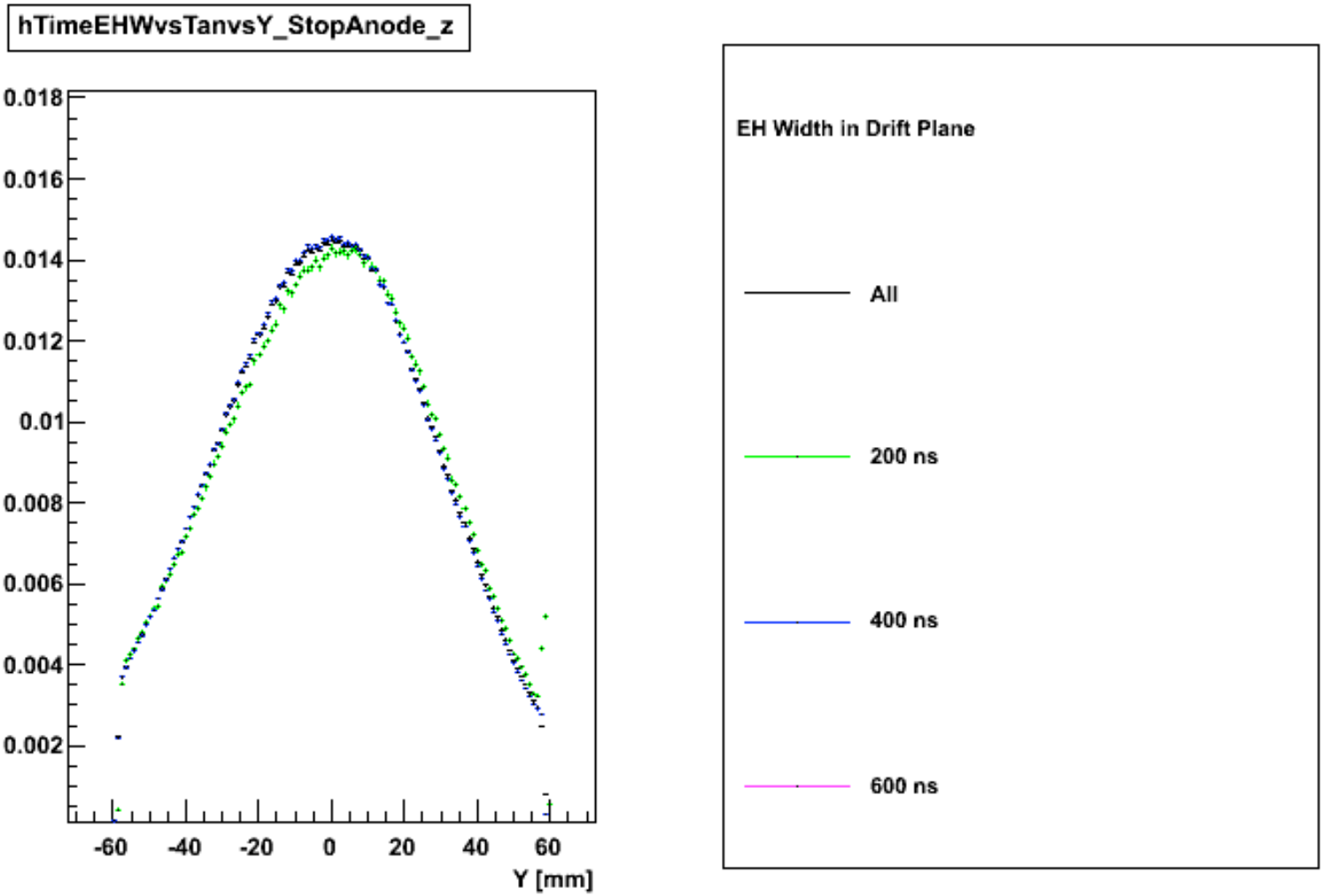
This nEHcont effect diminishes with a selection of the entrance angle (relative to horizontal) in the YZ plane.



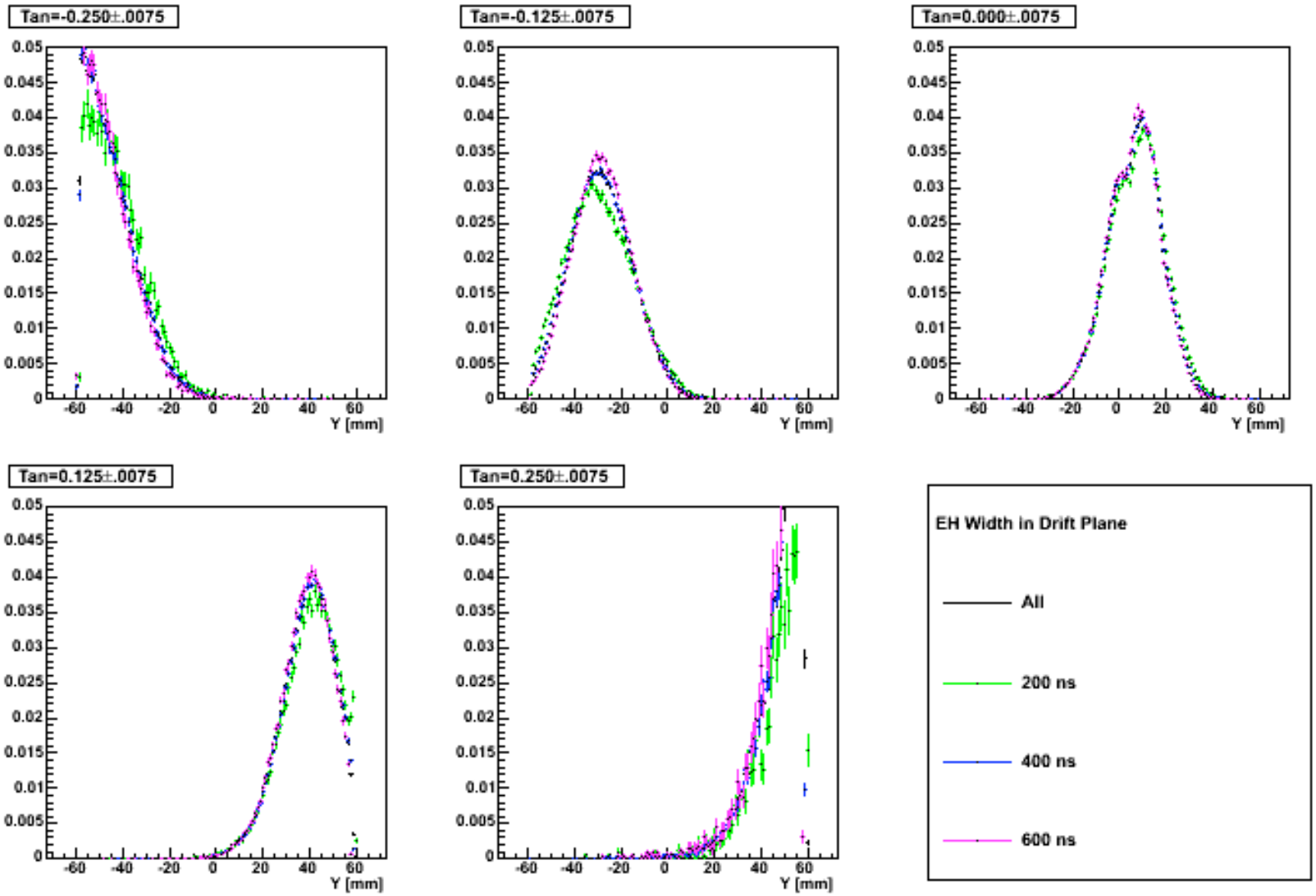
The remaining discrepancy is mostly between nEHcont=1 and nEHcont>1; nEHcont=1 stops are the most susceptible to scattering.



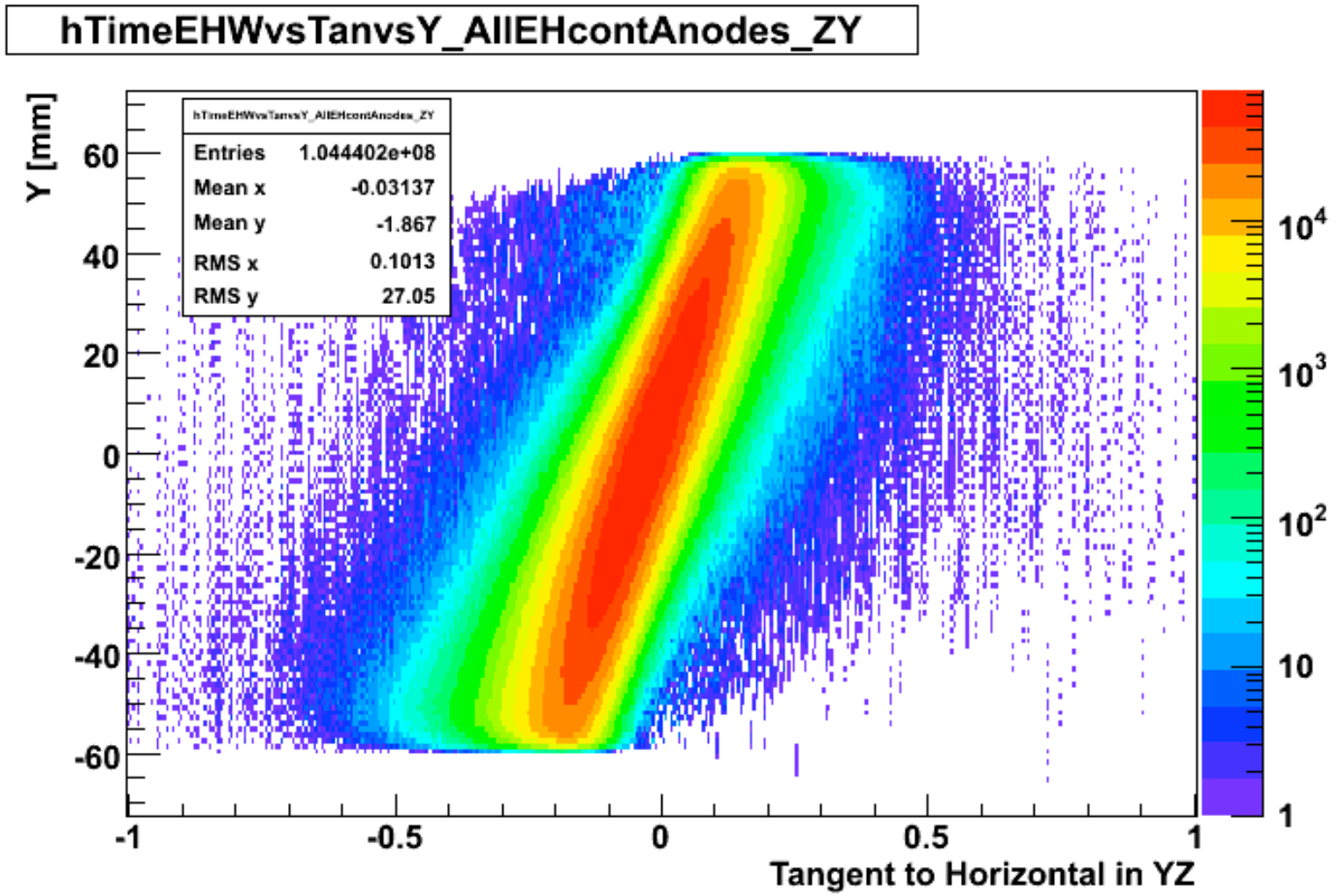
The Y distribution shows little variation, and most of that is between the 200 ns and larger pulses, in the drift plane



With a ± .0075 cut on the tangent some discrepancy remains in the Y distributions vs. EH pulse width in time, but at a minimal level. (Stop-Anode only)



In a second group of plots I looked for the EH pulse-width in time for <u>each continuous anode</u> (from the end) with an EH threshold pixel.



I obtain the number of continuous EH pixels in the drift dimension on the stop anode and then repeat that for the nEHcont-1 anodes preceding the stop.

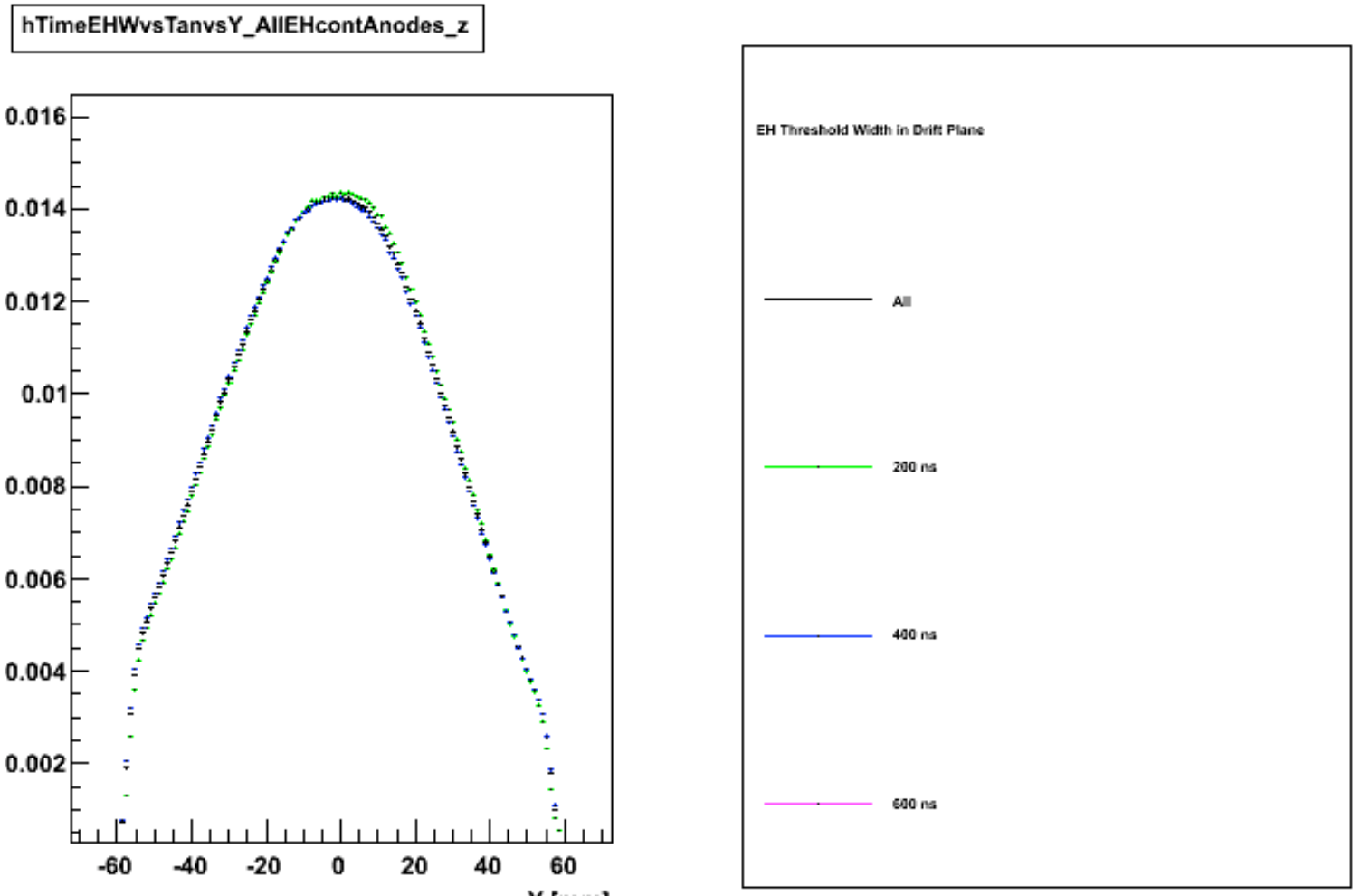
```
for(Int_t a=stopanode;a>=stopanode-nEHcont+1;a--){
  Double_t t_min;
  Double_t t_max;
  Bool_t first = 1;
  Double_t last;
  //printf("Starting on anode %d\n",a);
  for(Int_t p=0;p<nEH;p++){
    if(W_EH[p]==a){
                //printf("T_EH[%d]=%f on anode %d.\n",p,(double)T_EH[p],(int)W_EH[p]);
                //start the tally for making sure the pixels are continuous in Y
                if(first)last=T_EH[p];
                //stop search for EH pixels if there is a discontinuity of 1 pixel
                if(!first && TMath::Abs(T_EH[p]-last)>=2){
                  //printf("break!; t_max=%f\n",t_max);
                  break;
                  //t_max = T_EH[p-1];
                }
                //if we are still in the loop update the time of the last pixel which is a continuous pixel
                else if(!first)last=T_EH[p];
                //set the minimum time value to the first pixel time if this is the first pixel
                if(first)t_min=(double)T_EH[p];
                //otherwise compare the pixels and up date the minimum value
                else if((double)T_EH[p]<t_min)t_min=(double)T_EH[p];
                //do the same to set and update the maximumm value
                if(first)t_max=(double)T_EH[p];
                else if((double)T_EH[p]>t_max)t_max=(double)T_EH[p];
                //if we are still on the first pixel on the anode, update the Bool_t to know we aren't any longer.
                if(first)first=0;
    }
  }
```

```
                if(!first && TMath::Abs(T_EH[p]-last)>=2){
                  //printf("break!: t_max=%f\n",t_max);
```

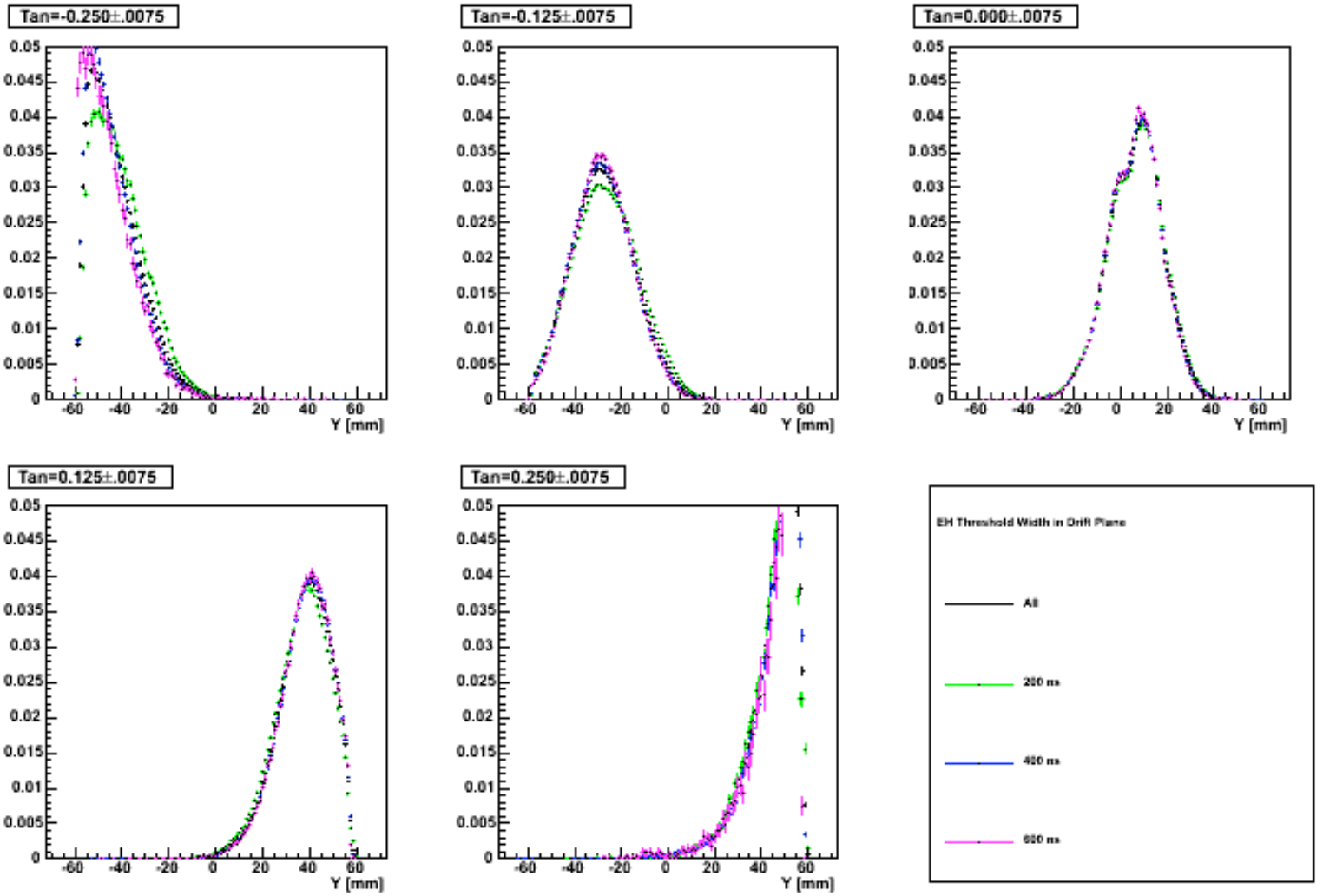
The definition of Y for the muon stop pulse is the usual GetY() definition of the muon stop Y coordinate. For the preceding continuous anodes I used the slope in YZ to get the Y of the pulse on the anode. (X 4 in statistics)

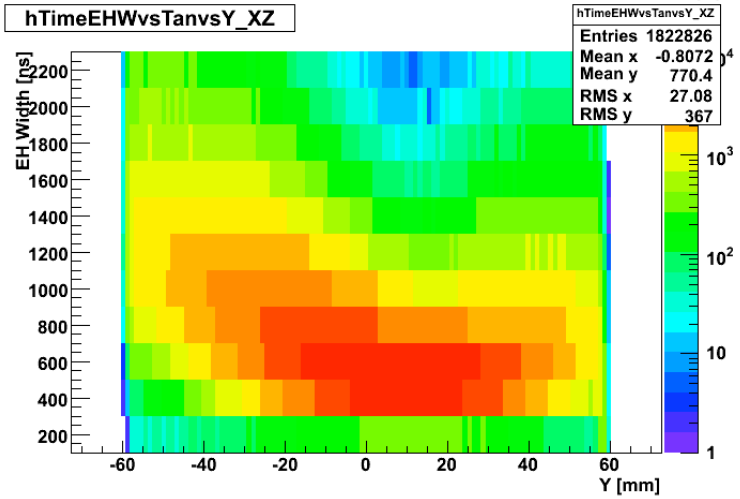
```
                  }
                }
                //printf("N Pixels wide on anode %d is %f\n",(int)W_EH[a],t_max-t_min+1);
                double TimeEHW = (t_max-t_min+1)*200;
                if(a==stopanode){
                  hNEHcontvsZvsY->Fill(nEHcont,muStop->GetZ(),muStop->GetY());
                  hTimeEHWvsZvsY_StopAnode->Fill(TimeEHW,muStop->GetZ(),muStop->GetY());
                  hNEHcontvsTanvsY->Fill(nEHcont,tan,muStop->GetY());
                  hTimeEHWvsTanvsY_StopAnode->Fill(TimeEHW,tan,muStop->GetY());
                  hNEHcontvsAngvsY->Fill(nEHcont,ang,muStop->GetY());
                  hTimeEHWvsAngvsY_StopAnode->Fill(TimeEHW,ang,muStop->GetY());
                }
                double Y_c = muStop->GetY()+tan*4*(a-stopanode);
                hTimeEHWvsZvsY_AllEHcontAnodes->Fill(TimeEHW,muStop->GetZ(),Y_c);
                hTimeEHWvsTanvsY_AllEHcontAnodes->Fill(TimeEHW,tan,Y_c);
                hTimeEHWvsAngvsY_AllEHcontAnodes->Fill(TimeEHW,ang,Y_c);
              }
            }//end if HasBestEntrance()
          } //end if IsTrack()
```

The Y distribution, again, shows little variation with the drift plane length



The total sum of 1e8 entries shows less variation than in the stop-anode case, again quite minimal.





The strong angular correlation between the EH drift-plane pulse-width and Y has disappeared with the single anode pulse width definition. Before I was limited to using the Y of the first and last EH of the muon stop.

This is true for both the single-stop-anode and all-continuous-EH pixels plots.