

Vertexing to-do list

Sho Uemura

Wrapping up and getting out the door

- **thesis is defended:**

http://www.slac.stanford.edu/~meeg/presentation/2016-09-15_defense/defense.pdf

- **thesis is done:** <https://github.com/meeg/thesis>

- **code is out, with README docs:**

- ▶ **tuple maker:** <https://github.com/JeffersonLab/HPS-CODE/tree/master/ANALYSIS/tuple>

- ▶ **vertexing analysis:** <https://github.com/JeffersonLab/HPS-CODE/tree/master/ANALYSIS/vertexing>

- **to-do list:** <https://confluence.slac.stanford.edu/display/hpsg/Things-to-do%3A++Vertexing+@+1.05+GeV%2C+0.5mm>

This talk

- Focus: general to-do items, taken from <https://confluence.slac.stanford.edu/display/hpsg/Things-to-do%3A++Vertexing+@+1.05+GeV%2C+0.5mm>
 - ▶ All items are (I think) required for any high-quality analysis (2015 0.5 mm, 2015 0.5+1.5 mm, 2016, etc.); no “nice to have” except as noted
 - ▶ Who’s going to do all of this? Not me, can’t all be Holly
- Other stuff:
 - ▶ Some proposal reach plots
 - ▶ Processing the full 1.5 mm data set

Add non-L1

- You have four kinds of events based on whether the electron and positron tracks have L1 hits; I require L1 hits on both tracks.
- I think the zcuts for events with and without L1 hits are always going to be significantly different. If so, it's hard to avoid splitting the data into multiple data sets: you might be able to rescale the Z-axis for the different event types, so the zcuts all line up, but that seems weird.
- I think the way to go is to split the data into four sets; tune cuts separately for each set, and find zcuts separately for each set. Then combine limits using the established procedure (and code) for optimum interval with multiple detectors.
- N.B. we need to know our L1 efficiencies very well - if L1 efficiencies are worse in data than MC, we're adding lots of events to the non-L1 data sets that do not appear in MC. May be possible to reject such events by checking whether the track should have hit L1.

Fix vertex fit

- The resolution of the reconstructed mass should be independent of Z but is worse for displaced vertices.
- This ad-hoc correction works pretty well: $\text{corrM} = \text{uncM} - 0.15e-3 * (\text{elePX}/\text{eleP} - \text{posPX}/\text{posP}) * \text{uncVZ}/\text{uncM}$
- Hunch: the vertex mass is being calculated using the track directions at $z = 0$, or something like that.

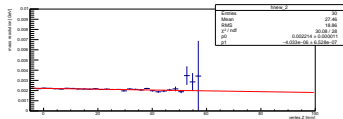
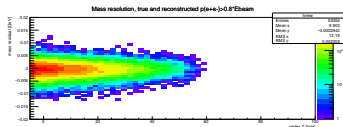
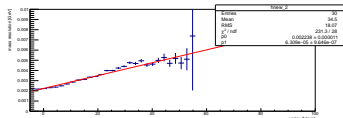
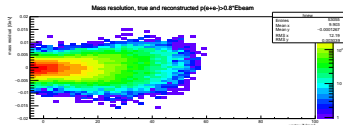
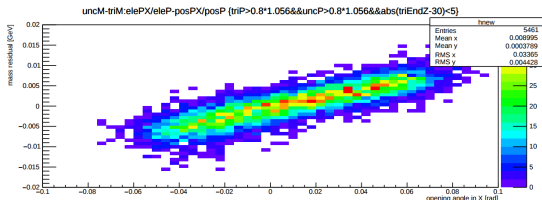
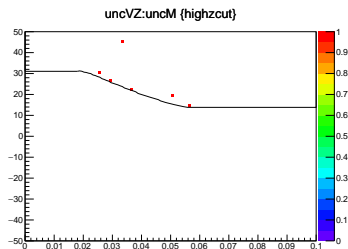
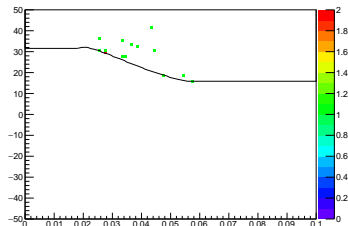


Figure out excess background

- Need to find out where the excess background events come from. Then, need to devise a cut to kill them. (If we can't, the vertexing analysis is sunk.)
- MC is the right way to go: there's one excess background event in postTriSummitFixes tritrig. Maybe there's something obviously special about it. If that doesn't work, make more MC: may require taking shortcuts in MC.



Tune cuts

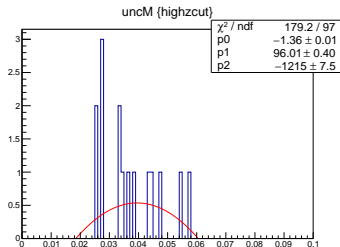
- The vertex cuts aren't really optimized: I just eyeballed the ROC curves. It should be possible to tune cuts to maximize sensitivity:
 - ▶ Pick a target mass and ϵ^2 . Generate a corresponding A' sample.
 - ▶ Make a data sample by taking a mass slice of the unblinded data.
 - ▶ To test a set of cuts, run the cuts on the data sample and the A' sample; fit the vertex distribution in the data sample to see what zcut you can get, then see how many A' you get past zcut. Tune cuts to maximize that number.

Prepare to unblind

- Run the analysis on negative Z, see if the background matches the model. Need to retune cuts, since some (isolation) are specifically tuned to reject things in +Z.
- Idea from Natalia: if you smear the mass enough, you can safely look at 100% of the positive Z events before unblinding. Since the Z distribution changes with mass, you may need to scale Z at the same time (meaning, if you send $m \rightarrow m + \delta m$, send $z \rightarrow z(1 - a\delta m)$, or something like that, so the z vs. mass distribution looks approximately the same).

Do look-elsewhere correction correctly

- The MC model I use for the look-elsewhere correction is really bad - I take the distribution of events that appear in at least one mass slice, and make a toy 1-D mass distribution based on that. This assumes that any such event shows up in every mass slice that it overlaps. This is not a good assumption.
- The right way to do the look-elsewhere correction would be to make a toy 2-D distribution (z vs. mass), then take mass slices from that. Since we have a fit to the mass distribution and have fits to the vertex tails vs. mass, we have everything we need to make the toy 2-D distribution.



Profile likelihood?

- The way I do significance in the presence of excess background (count what's in my slice, then look at the counts in neighboring slices to estimate background in my slice) is very ad-hoc and probably not good. I think the right way to do it may be to do a 2-D profile likelihood ratio, fitting to signal + (known background) + (excess background), where the excess background is something that's a simple function (linear?) in mass and something intelligent (acceptance?) in Z .

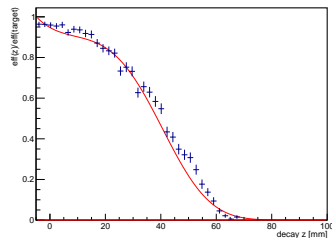
Event flags for 1.5 mm

- Holly's 1.5 mm data set starts at run 5403; older runs (the majority of the 1.5 mm data) do not have SVT bias in MYA or EPICS events
- `org.hps.monitoring.drivers.svt.SampleZeroHVBiasChecker` uses SVT noise to identify bias-off periods (used to cross-check the MYA information)
 - ▶ This was done by Pelle and me
 - ▶ Should be extended to add bias-off periods to the conditions DB
 - ▶ I suggest this be done by the SVT group with support from me

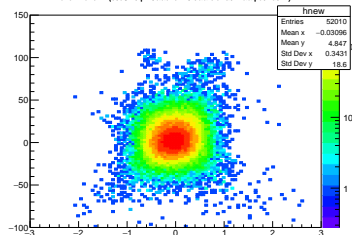
Small stuff

- Add WABs to background tails fit
 - ▶ Can't be done with wab-beam-tri, which is $O(1000)$ times less efficient than tritrig-beam-tri
- Improve acceptance fits
- Cut out tridents produced in L1

Efficiency for 50 MeV heavy photon

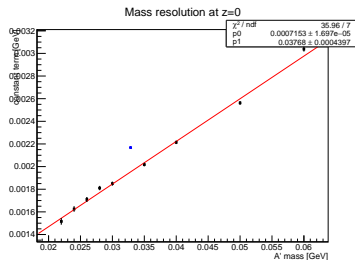
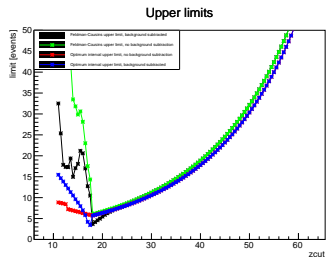


uncVZuncVY (bsc:Chisq<10&&uncP>0.5&&eleHasL1&&posHasL1)



More small stuff

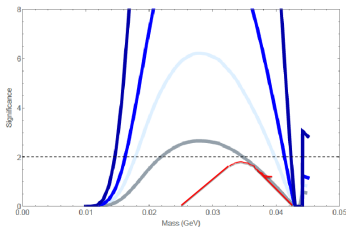
- Why doesn't optimum interval background subtraction work? (not crucial)
- Figure out mass resolution discrepancy: what's the mass dependence?
 - ▶ Is it from MS simulation effect on angle resolution? alignment effect on momentum resolution? something else?
 - ▶ Use Moller data and MC, measure contributions of momentum resolution and angle resolutions
- Systematics and stuff



Proposal reach

- Proposal estimates vs. state of the art, same assumptions (1 week of MC, uniform efficiency vs. Z)

Much Worse at $\epsilon = 10^{-9}$



Significance vs Mass (1.1 GeV; $\epsilon=3 \times 10^{-9}$)
Matt

