TPAC test beam analysis tasks

Paul Dauncey

Run quality

- Run quality: make list (spreadsheet, ascii, whatever) with, for each run
  - Sensor id for each layer
  - Number of bunch trains and PMT coincidences
  - Threshold and average and rms temperature for each layer
  - Beam spill structure and in/out spill timing info
  - Sensible good/bad flag for configuration (per layer?)
  - Ditto for any other problems

- Some can be automated, others need to be done by hand
- Spill structure; fit each run to know when spill on and off; useful?
- Good to have simple program to select runs based on threshold, bad flags, etc, which makes a run list suitable to pass to mpsAnalysis
- I would favour ascii; a txt file can be easily read into xls but I don’t know how to read xls in C++

Bad pixels

- Bad pixel handling
  - Bad sensor configuration (masking, wrong value, config errors)
  - Memory full (per bunch train or per bunch crossing?)
  - Bad threshold setting

- Only know sensor configuration was bad at end-of-run; hence need to process each run to identify bad configuration separately from analysis run
- Some bad runs need to be identified from log by hand
- Bad thresholds (may not be an issue?) could be seen during run but easiest to handle in the same way
- Suggest storing bad pixel list (one bit/pixel = 3.5kBytes/sensor/run) for each run in files, read in during mpsAnalysis for the run
- Full memory flags would need to be added to this for each bunch train, or potentially need to be modified depending on the bunch crossing

Efficiency analysis

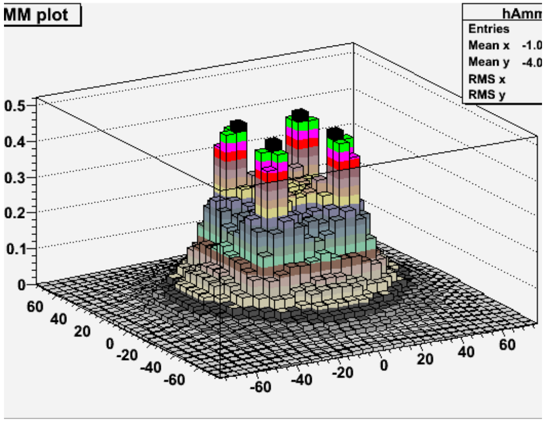
- Efficiency analysis
  - Clustering; position and errors, shape/charge spread

- Algorithm for grouping all nearest neighbours is doable but tricky
  - Need to be careful at boundaries, e.g. y=83 and 84 are not neighbours
- First use: need clustering to form hits to fit for tracks
  - For position, I would guess geometric mean position is reasonable cluster centre estimate , i.e. $\mathbf{r}_c = \sum_i \mathbf{r}_i$
  - Error is trickier, e.g. two adjacent pixels; is error 100μm/√12 or will two pixels only fire if particle very close to boundary, so error ~few μm?
  - What if cluster borders (or encloses) a bad pixel? E.g. A single pixel with a bad pixel as nearest neighbour; fired or not shifts the mean by 25μm
  - Can we afford not to use any cluster which touches a bad pixel?
- Second use: cluster size/shape studies to compare with MC
  - Number of pixels/cluster, pattern of pixels in cluster, etc
  - Similar issues apply

Efficiency analysis

- Efficiency analysis
  - Extraction of efficiency given track projection

- Discussed in previous meeting but need some ideas
- One possible approach; consider as if only one good pixel in a layer
  - Make two 2D histograms of ~±125μm in xy, centred on this good pixel
  - For every track hitting the layer, put entry in first histogram at xy of track
  - Also put entry in second histogram at xy of track if good pixel fired
  - Would expect first to be ~uniform, second to peak for tracks near pixel
  - Dividing one by the other gives efficiency vs position relative to pixel



Efficiency analysis

- In reality have many good pixels; assume all give the same response
  - Average over all good pixels in same two histograms
  - For each track projection, find all good pixels in 5×5 array around impact
  - For each, pretend this good pixel is “the” good pixel as before and so enter xy point in histograms relative to this pixel
  - Every track gives up to 25 entries in each histogram
- Would need to duplicate the plots for out-of-time hit associations
  - Check of background levels for rate of hits away from centre of plot
  - E.g. use $t_b = (t_s+4000)\%8000$ as discussed in Wed meeting
- Plot is really efficiency vs position convoluted with track position resolution
  - Need to check on uniformity of track projection errors
  - Cannot realistically deconvolute?
- Probably low on statistics; should consider using all layers, not just inner two layers, for efficiency studies
  - Need all combinations of five of the six layers projected to the other layer

Efficiency analysis

- Efficiency analysis
  - Effective threshold corrections (temperature, other pedestal shifts)
  - Inefficiency due to monostable pulse length

- May need to take new data on pedestals vs temperature for several sensors
  - Sensible to do with sensors actually used in beam test
  - Need to see general trend; pedestal ΔTU per degree C
  - Average temperature per run then gives correction to threshold value
- Monostable fires for fixed time period asynchronous to 400ns BX timing
  - If monostable length L > 400ns, then no inefficiency and rate of single hits to double hits (in sequential BX) allows monostable length estimation
  - If monostable length L < 400ns, then inefficiency due to times when pulse does not overlap BX clock edge; $\varepsilon \sim L/400$ns. Never see double hits so no way to estimate L and hence size of effect from beam test data
  - Will need to run sensors (at RAL?) and measure length (?)

Shower studies

- Tungsten shower studies
  - Pion and electron response

- Not so clear what is needed here yet as no analysis started
  - Main idea would be to compare data with MC, correcting for efficiency measured from tracking analysis
  - General study of cluster numbers, sizes and shapes in pion and electron showers
  - Correlations layer to layer may indicate tracks; random positioned hits may indicate photons; need MC to see how well this works
- We have zero knowledge of electron beam position and size
  - Tungsten was wider than sensors so particles outside sensor area can shower back into sensor region
  - MC modelling will have uncertainty due to this; how significant?

Simulation

- Simulation
  - Production; how many events per threshold? Which thresholds?
  - Multiple particles per bunch train
  - More realism in materials; upstream and within stack
  - More realism in sensors; hi-res charge spread
  - Tungsten runs

- First two are technical issues mainly
- Upstream materials; unclear how good a record was kept. Really only an issue for electron showers so no Fortis/SiLC, but only EUDET telescope within beam area. Also, air and other components (vacuum windows) for ~50m upstream.
- Material in stack probably small effect but should try to model PCBs and scintillators reasonably (not done yet)
- Not sure how to model hi-res charge spread; Gary?