SoLID Tracking Simulations

Ole Hansen

Jefferson Lab

with Seamus Riordan, Rich Holmes

SoLID Collaboration Meeting
April 13, 2012

SoLID Tracking Considerations (from Seamus)

Challenges:

- High rates: $\mathcal{O}(10 \text{ kHz/mm}^2)$
- Real-time processing required for level-3 trigger
- Parity experiment: pay attention to possible helicity-dependent systematics

Approach:

- Use GEMs
- Learn from SBS
- Thoroughly simulate tracking

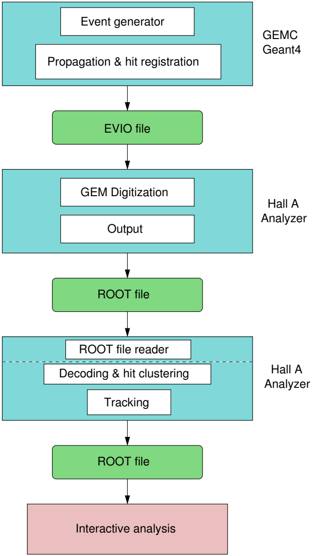
Tracking Simulation Goals (from Seamus's ideas)

- GEM occupancies at proposed luminosities
- Particle types associated with GEM hits (signal/noise)
- Noise rate from induced photons (correlated noise)
- Track rates: physics, background, ghosts
- Tracking efficiency
- Reconstruction accuracy
- Rate dependencies (background, helicity effects)
- Optimal GEM hit clustering algorithm (noise tolerance)
- Optimal readout strip configuration (x/y vs. r/ϕ)
- Optimal track finding algorithm (speed, accuracy, efficiency)

Not necessarily a complete list ...

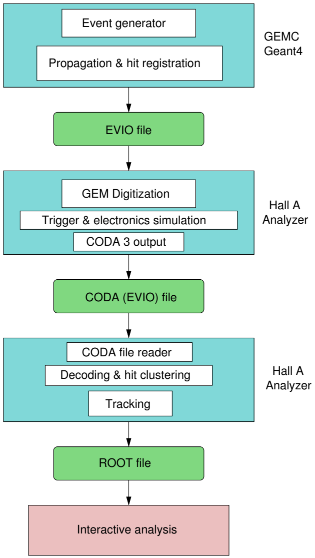
Tracking Simulation Flowchart

- solgemc EVIO files as input
- GEM digitization based on SBS work (E. Cisbani, INFN)
- APV25 pulse shape simulation done
- No other detectors digitized yet. Partial passthrough of generated data (tracks, vertices)
- ROOT file interface & decoder done
- Tracking under development



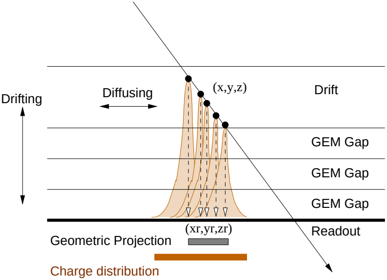
Tracking Simulation Flowchart ("data challenge" ready)

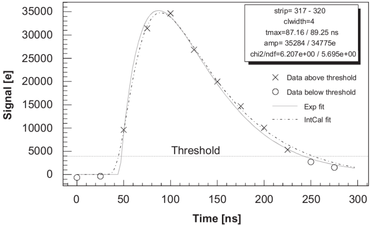
- `solgemc` EVIO files as input
- GEM digitization based on SBS work (E. Cisbani, INFN)
- APV25 pulse shape simulation done
- No other detectors digitized yet. Partial passthrough of generated data (tracks, vertices)
- ROOT file interface & decoder done
- Tracking under development
- Should eventually use actual DAQ format (CODA 3) for analyzer input



GEM & APV25 Digitization (adapted from SBS by Rich Holmes)

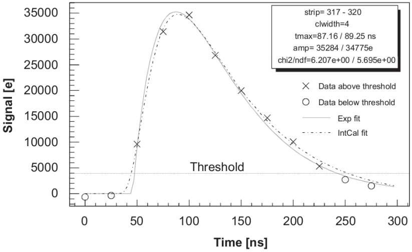
- GEMC outputs raw hits (energy deposition ΔE) in GEM layers
- GEM response tuned to match COMPASS observations
- Avalanche simulation:
  - Poisson-distributed number of ion pairs calculated from ΔE
  - Use geometric distribution for ionization probability along path
  - Assume constant-velocity diffusion and drift
  - Gaussian distribution of charge deposited on strips
- Shape output amplitude: $v = A\tau\exp(-\tau)$, record 3 samples in 25ns intervals





APV25 Pulse Shape Deconvolution

S. Gadomski *et al.*, NIM A 320, 217 (1992)



- For first-order RC circuit, signal amplitudes s_k can be deconvoluted from three measured values v_k:

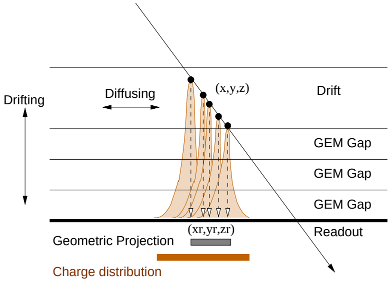
$$s_k = w_1 v_k + w_2 v_{k-1} + w_3 v_{k-2}$$

$$w_1 = e^{x-1}/x, w_2 = -2e^{-1}/x, w_3 = e^{-x-1}/x, x = \Delta t/T_p$$

$$A \approx \sum_{k=1}^{3} s_k$$

- Reject noise by cutting on ratios, $r_1 = v_3/v_1$ and $r_2 = v_2/v_1$, requiring rising slope

GEM Hit Clustering

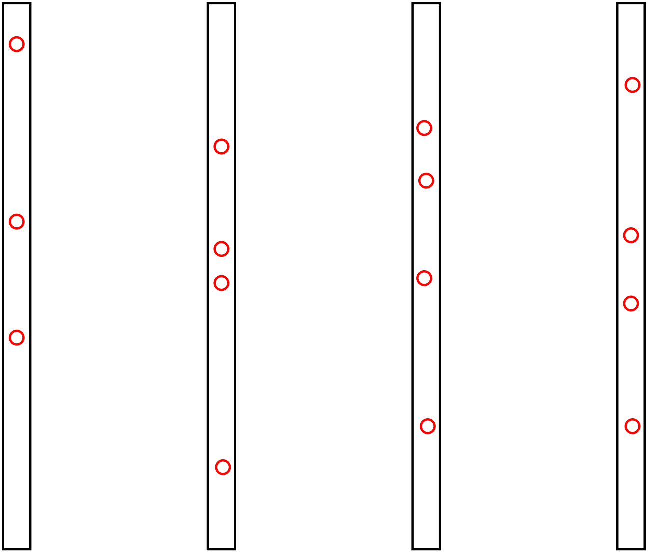


- Signals on adjacent readout strips typically belong to a single track crossing
- Sum signals to get
  - Total hit amplitude
  - Charge-weighted position centroid
- Currently use simple algorithm:
  - Look for local peak
  - When sequence "peak-valley-peak" is seen, split cluster at "valley"
  - Regardless of shape, limit clusters to a maximum size
- Improvements
  - Match hits by their pulse shape, i.e. timing centroid
  - Redo clustering after preliminary tracking
  - . . . possibly more
- Clustering does not necessarily have to be separate from tracking, could be integrated into a progressive tracking algorithm

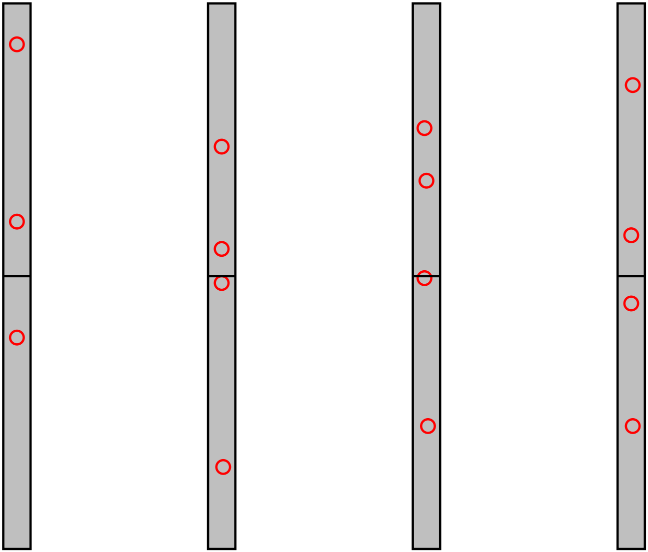
Track Reconstruction Algorithm Candidates

- Xin's Progressive Algorithm
  - Kalman filter
  - Needs seed (e.g calorimeter hit)
  - Allows arbitrary track curvature
  - Already shown to be feasible for PVDIS/SIDIS rates
  - Slow, $\approx \mathcal{O}(Nk^2)$
  - Not yet implemented in Hall A analyzer
- TreeSearch
  - Global algorithm (recursive template matching)
  - Very fast, $\mathcal{O}(\log N)$
  - Available in Hall A analyzer
  - Successfully used with BigBite and SBS simulations
  - Requires straight tracks, but may still work for small curvature

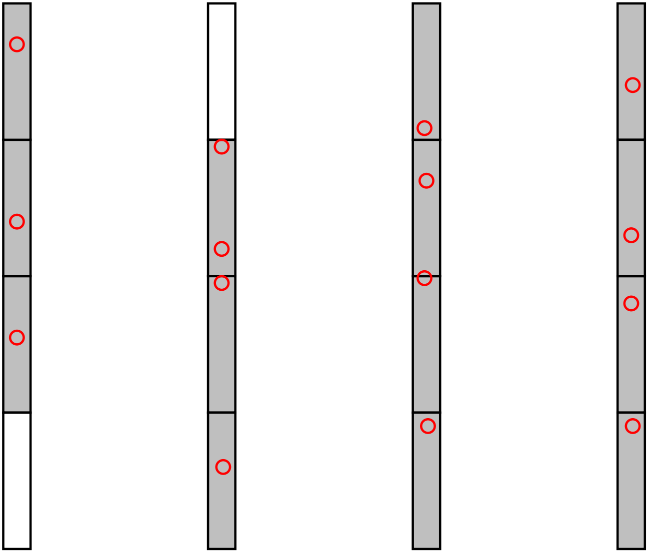
TreeSearch Illustration



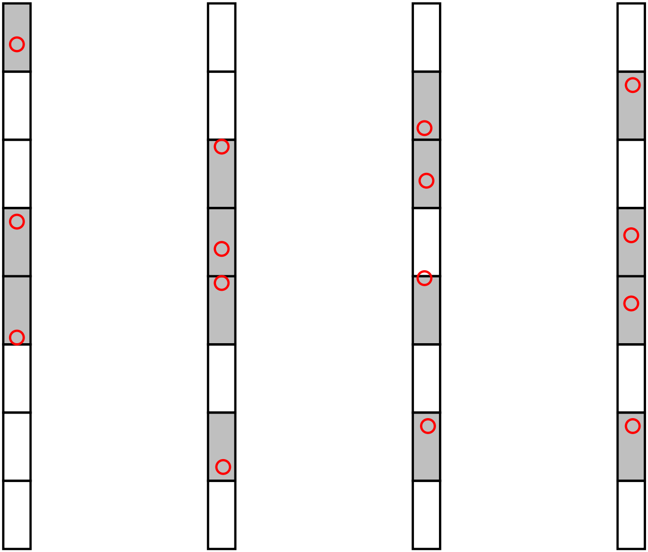
TreeSearch Illustration



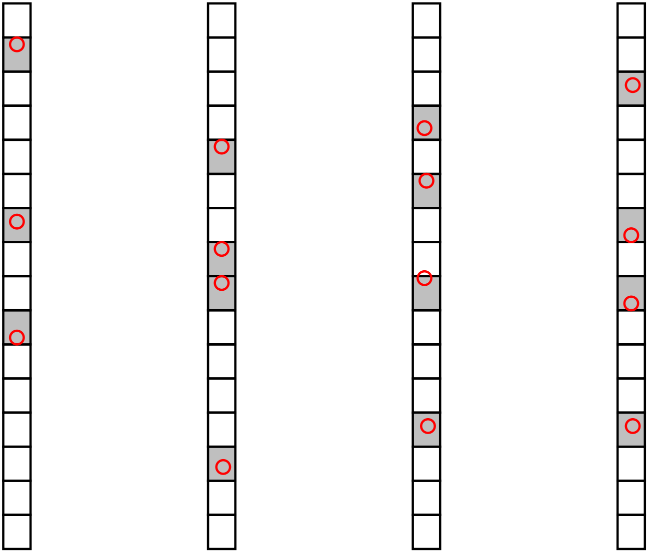
TreeSearch Illustration



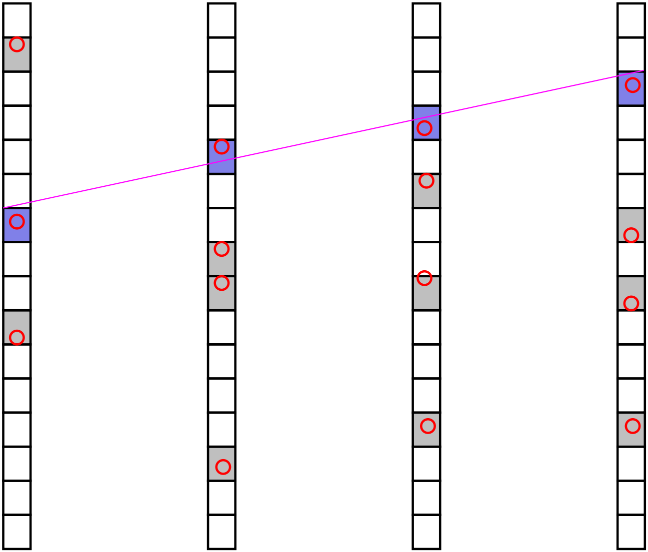
TreeSearch Illustration



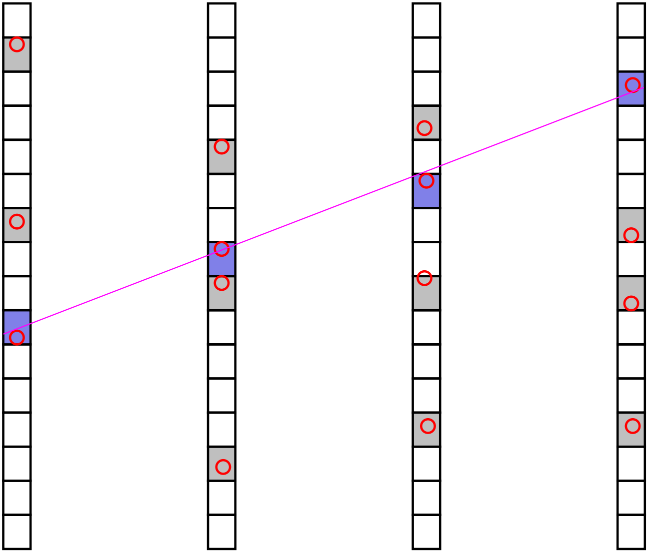
TreeSearch Illustration



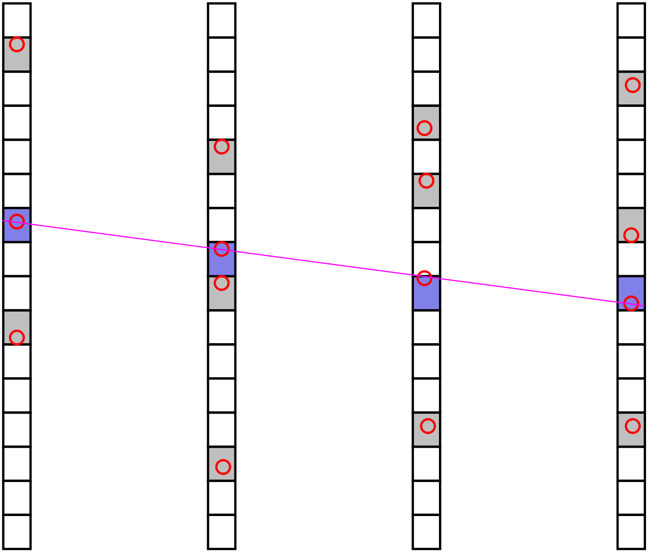
TreeSearch Illustration



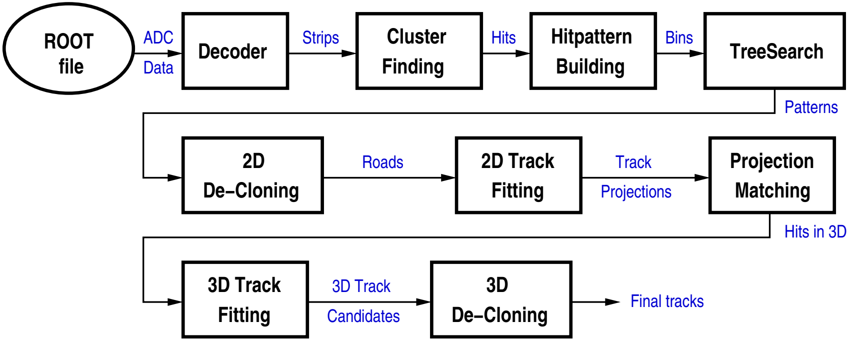
TreeSearch Illustration



TreeSearch Illustration



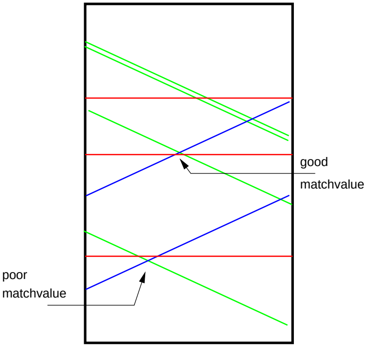
TreeSearch Reconstruction Flow



3D Matching

- Correlation of roads from different projections, either
  - geometrically (3+ projections); or
  - via hit amplitude & timing in shared readout planes (2 projections)
- Repeat for each chamber group along z

Chamber front view



TreeSearch for SoLID: Status

- Code exists from SBS simulations
- In principle, all components implemented
- To do
  - Adapt database and code to SoLID geometry
  - Generate pattern lookup tables
  - Tune parameters
  - Find tracks for field-free case
  - Investigate effect of field, track curvature
  - Handle possible curved tracks in final fitting step

TreeSearch for SoLID: Status

- Code exists from SBS simulations
- In principle, all components implemented
- To do
  - Adapt database and code to SoLID geometry
  - Generate pattern lookup tables
  - Tune parameters
  - Find tracks for field-free case
  - Investigate effect of field, track curvature
  - Handle possible curved tracks in final fitting step
- Get rough numbers for tracking efficiency & ghost track rate

TreeSearch for SoLID: Status

- Code exists from SBS simulations
- In principle, all components implemented
- To do
  - Adapt database and code to SoLID geometry
  - Generate pattern lookup tables
  - Tune parameters
  - Find tracks for field-free case
  - Investigate effect of field, track curvature
  - Handle possible curved tracks in final fitting step
- Get rough numbers for tracking efficiency & ghost track rate
- $\Rightarrow$ "basic tracking demonstration"

TreeSearch for SoLID: Status

- Code exists from SBS simulations
- In principle, all components implemented
- To do
  - Adapt database and code to SoLID geometry
  - Generate pattern lookup tables
  - Tune parameters
  - Find tracks for field-free case
  - Investigate effect of field, track curvature
  - Handle possible curved tracks in final fitting step
- Get rough numbers for tracking efficiency & ghost track rate
- ⇒ "basic tracking demonstration"
- ETA: June meeting

Outlook

- Adapt Xin's progressive tracking to Hall A analyzer & optimize performance
- Include other tracking-relevant detectors in digitization & analysis
- Run through full program of tracking studies outlined earlier

Outlook

- Adapt Xin's progressive tracking to Hall A analyzer & optimize performance
- Include other tracking-relevant detectors in digitization & analysis
- Run through full program of tracking studies outlined earlier
- $\Rightarrow$ "full tracking evaluation"