

CKF & VXD Hit Recovery for Belle 2

Ian J. Watson

University of Tokyo

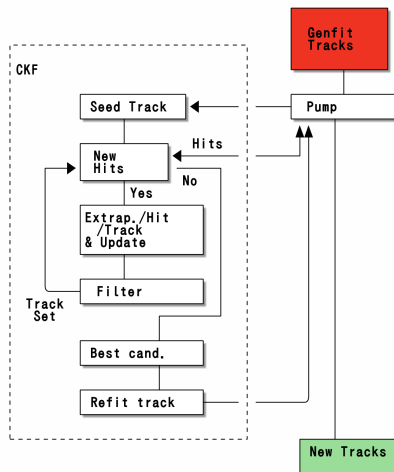
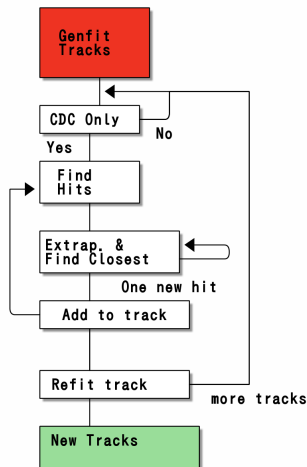
B2 Tracking Meeting F2F
September 2, 2015

Reminder - Combinatorial Kalman filter

- Searching for VXD hits to add to tracks found in CDC only

VXD Extrapolator

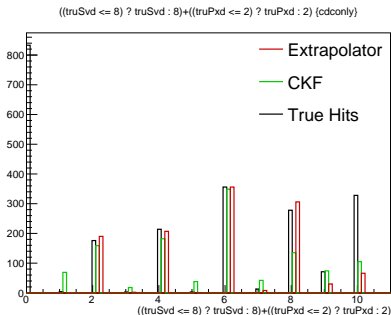
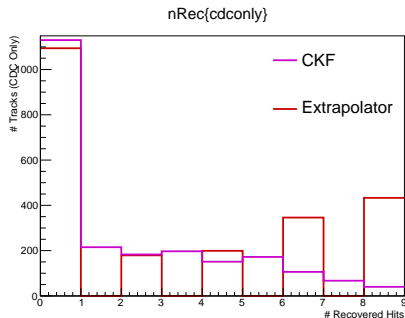
CKF



Track Metric Updates

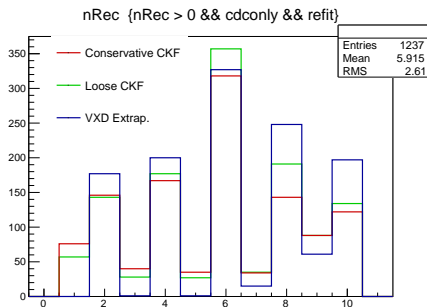
- Was previously using $\sum_{hits} \Delta\chi^2 / \sum_{hits} NDOF$ to find best track
 - With χ_{max}^2 cutoff for individual hits
- Based on papers in Rudi's talk, updated quality function
- Now, following ([Mankel, 1997](#)), maximize
$$Q = N_{hits} - N_{holes} - (1/w_{hole}) \sum (\Delta\chi_i^2 / NDOF_i)$$
- So, bonus for true hits, penalize holes
 - Hit = a pixel hit or one-sided SVD hit
 - N_{holes} is the # of *true holes*, holes which are followed by hits
 - For my tracks, track might have only hit last few layers, don't want to penalize subsequent layer search
 - Probably need to make an option to turn this on/off for other users
- $w_{hole} = 10$, weights relative impact of bad hits versus holes
- Also, restrict number of candidates under consideration to N_{max}
- Nb. plots on following pages all run using same 1000 BB event sample where one $B \rightarrow [K_S\pi\pi]_D K$ and has background files added

Recovered hits



- Left shows the # hits found by old quality CKF (using just χ^2 on hits) vs the VXD extrapolator
- Right shows the # hits found by new Q CKF, VXD Extrapolator and the # of true additional VXD hits for each of the tracks (restricted to 8 max SVD and 2 max PXD since I'm not looking for more)

Found hits



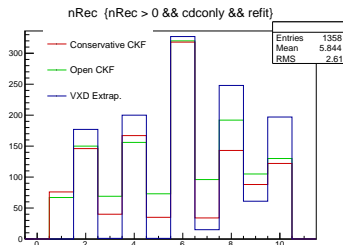
- "Conservative",
 $Q = N_{\text{hits}} - N_{\text{holes}} - 1/25 * \sum \chi^2$,
 $\chi^2_{\text{max}} = 10$, $\text{holes}_{\text{max}} = 1$, $N_{\text{tracks}} = 10$
- "Loose",
 $Q = 1.5 * N_{\text{hits}} - N_{\text{holes}} - 1/15 * \sum \chi^2$,
 $\chi^2_{\text{max}} = 40$, $\text{holes}_{\text{max}} = 5$,
 $N_{\text{tracks}} = 20$

	"Conservative"	"Loose"	VXD
Hits Recovered	6633.0	7317.0	7835.0
True Hits Recovered	6509.0	7116.0	7586.0
Purity	0.9813	0.9725	0.9682

- In principle, both CKF and VXD extrapolator are both searching within a sensor width for good hits, so why is VXD finding more?

- Tried running an "open" CKF, with the idea to accept as many hits as possible

- $Q = 10 * N_{\text{hits}} - N_{\text{holes}} - 1/15$
 $* \sum \chi^2, \chi^2_{\text{max}} = 150,$
 $\text{holes}_{\text{max}} = 7, N_{\text{tracks}} = 20$



	"Conservative"	"Loose"	"Open"	VXD
Hits Recovered	6633.0	7317.0	7936	7835.0
True Hits Recovered	6509.0	7116.0	7156	7586.0
Purity	0.9813	0.9725	0.9017	0.9682
N Tracks with Added Hits	1169	1237	1358	1227

- So, we're finding more hits, but not more truth-matched hits
 - So, where are these extra hits in the VXD Extrapolator coming from?
 - I am getting a lot of "max iteration exceeded" errors in CKF only
 - But, I realized I have neglected talking about an Extrapolator update!

Possible Solution & Other Misc. CKF2 Updates

- Realized as preparing the talk, I added a Kalman update step in the VXD extrapolator
 - So, after adding a compatible hit, it does an update step, and uses this to keep propagating our track
- For the CKF, I'm just extrapolating the seed track and looking for
 - always one sensor width cutoff, at inner layers, more likely to extrapolate to the wrong sensor if one sticks with the seed track
- For hit/hole assignment, needed to build CKFPartialTrack container inheriting from `genfit::Track` to hold hits and true/all holes
- Changed `findHits(..)` to also pass back to the user-defined hit finding function the current set of tracks under consideration

```
bool (*_findHits)(genfit::Track*, std::vector<CKFPartialTrack*>&,
                 unsigned, std::vector<genfit::AbsMeasurement*>&,
                 void*)
```

- Can retrieve the track set (not just best) from the CKF
- All changes in Belle 2 svn for anyone wanting to try running the CKF
 - `tracking/modules/cdcToVXDExtrapolator`

