The track finding algorithm of the Belle II vertex detectors

T. Lück
on behalf of the Belle II tracking group

Connecting the Dots / Intelligent Trackers 2017
LAL-Orsay (France)

March 8, 2017

Istituto Nazionale
Fisica Nucleare
Sezione di Pisa

outline

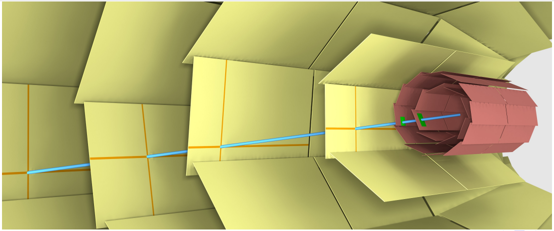
- Introduction
- The Belle II vertex detector
- Concept of the track finding at Belle II
- Implementation at Belle II
- Preliminary results
- Summary

Introduction

- Belle II multipurpose detector operated at the SuperKEKB accelerator
- SuperKEKB will deliver a peak luminosity of $8 \times 10^{35} cm^{-2} s^{-1}$
- electrons and positrons will be collided at a center of mass energy corresponding to $m(\Upsilon(4S)) = 10.58 GeV/c^2$
- aims to test the standard model with unprecedented precision
- on average 11 tracks in a $\Upsilon(4S)$ event
- tracking devices:
  - 2 layers of pixel detectors (PXD) + 4 layers of double sided silicon strip detectors (SVD)
  - central drift chamber (not part of this talk)
- reliable track finding is crucial for the success of the Belle II experiment

Track finding used for data reduction

- Belle II has two trigger stages:
  - hardware based trigger L1
  - High Level Trigger (HLT): software based trigger
- track finding algorithm will be used on the HLT to reduce the amount of data read out by the PXD
  - find tracks in the SVD
  - extrapolate found tracks to the PXD
  - define Regions Of Interest (ROI) on PXD sensors
  - read only PXD - hits found in ROI (data reduction factor $\approx 10\%$)



The Belle II inner Vertex Detectors

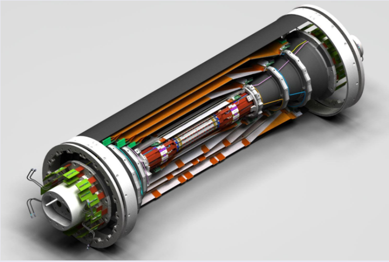
Silicon Vertex Detector (SVD)

- 4 layers of Double-sided Silicon Strip Detectors (DSSD) at radii of 38, 80, 115, and 140 mm from the interaction point
- $17^\circ < \theta < 150^\circ$ coverage in polar angle

Pixel Detector (PXD)

- two layers of DEPFET pixel detectors at radii of 14 and 22 mm

Computer rendering of the Belle II vertex detectors (SVD+PXD)



Mechanical mockup of the PXD

- on average an $\Upsilon(4S)$ event has 11 low momentum tracks
- due to machine background largely increased number of space points
- MC studies predict 318 space points for the first layer of the SVD (234 from bkg alone)
- for the whole SVD a total of 721 space points (535 from bkg alone)
- needed: fast and efficient way to reduce the number of combinations
- adopted the sector on sensor concept proposed by R. Frühwirth et. al.

Y(4S) event with and without background

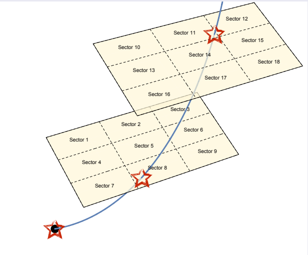
Case	$\Upsilon(4S)$-only	BG-only	$\Upsilon(4S)$ + BG	$\Upsilon(4S)$ + 2×BG
L3 strips u/v	49.2/36.7	260.0/121.7	308.1/158.0	562.2/278.8
L3 clusters u/v	11.8/11.8	39.0/37.9	50.3/49.3	87.0/86.1
L3 SPs	26.1	233.9	318.0	791.0
L4 strips u/v	39.4/29.1	120.3/61.2	159.1/90.1	277.8/150.6
L4 clusters u/v	12.7/12.6	29.9/26.7	42.5/39.2	71.8/65.3
L4 SPs	22.5	100.5	143.1	320.4
L5 strips u/v	37.3/28.5	122.7/67.2	160.1/95.8	282.7/162.9
L5 clusters u/v	12.3/12.1	35.0/30.5	47.3/42.7	82.0/72.9
L5 SPs	19.2	99.3	132.3	299.3
L6 strips u/v	38.3/28.6	134.6/76.8	172.9/105.4	307.1/182.0
L6 clusters u/v	12.4/12.2	42.1/36.3	54.4/48.5	96.2/84.5
L6 SPs	17.0	100.8	127.9	283.1
Average strips/layer u/v	164.3/122.8	159.4/81.7	200.1/112.3	1429.8/774.4
Total clusters u/v	49.2/48.7	146.0/131.3	194.4/179.6	337.1/308.9
Total SPs	84.8	534.6	721.3	1693.8
	Signal	Noise		

- SP : Space Points. "u" is the local r φ direction. "v" is the local z direction.
- The combinatorial problem is dominated by background hits

The sector on sensor concept

- divide sensor into rectangular sectors
- first step track finding: combine two space points (SP) to form a *segment*
- SP combinations searched only on two sectors which are *friends*
- two sectors are friends if during training same MC particle passes both sectors

Illustration of the sector concept



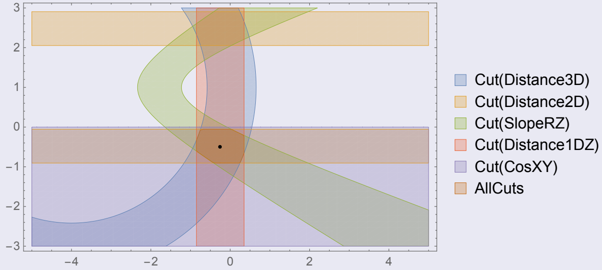
Filtering of space point combinations

- simple geometric cuts are used to accept/reject segments and tracklets (combinations of segments):
    - 2 SP: e.g. direction of the segment, distance between the two space points
    - 3 SP: angle between the two segments, transverse momentum
    - 4 SP: difference in transverse momentum between the first three SP and the last three SP, zig zag
- simulation is used to obtain cut values (upper / lower bound)
- customized set of cut values provided:
    - i.e. a separate set of cut values for each occurring sector combinations is given

Illustration of the filtering of two SP combinations

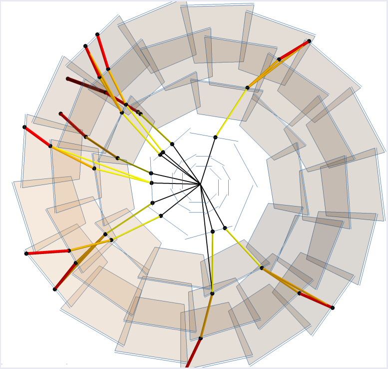
- shown is the effect of various cuts applied on 2-hit combinations
- each coloured area presents the selected region by the respective cut

Out[59]=



Connecting the Dots

- Cellular automaton collects segments and connects them to track candidates
- virtual IP is used to increase the number of space points



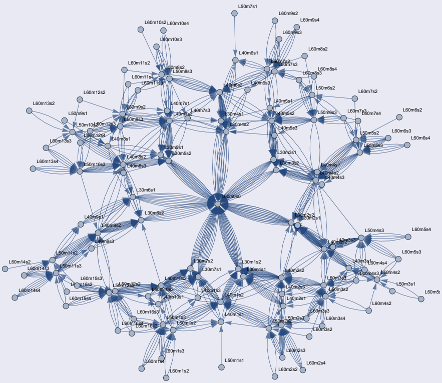
Overlap removal

- the cellular automaton provides a set of potentially overlapping track candidates
  - share at least one common space point
- these are used to construct a Hopfield neural network
- input to the network:
  - connection between track candidates (shared space points)
  - a quality indicator for each track candidate (e.g. a circle fit)
- output of the Hopfield network is a unique (non overlapping) set of track candidates

Sector Map

- store friend relations
- needs to be trained on simulated events to learn the allowed connections between sectors
- store a set of individual cuts for each sector combination
- adapts to geometry during the training
- currently uses 13MB of disc space, will be stored in the Belle II condition database

Graphical representation of a sector map for the Belle II geometry



Current status of the implementation

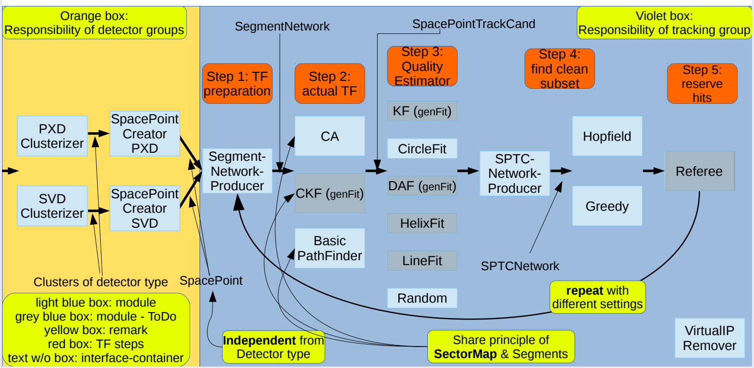
VXD Track Finder version 1 (VXDTFv1)

- implemented and working stably
- some issues (hard to maintain, main developer left)

VXD Track Finder version 2 (VXDTFv2)

- currently under development
- fix the issues with version 1:
  - increase modularity
  - facilitate maintainability
  - facilitate adding of new features
  - facilitate exchanging of features
- some features not yet implemented (e.g. multiple passes)

Data flow of the VXDTF version 2



- space point creation is provided by the different sub detector reconstruction software
- each reconstruction step is (at least) one separate module in the code: better maintainability, easier to switch out parts, easier to add new features

(Very) preliminary results in the following

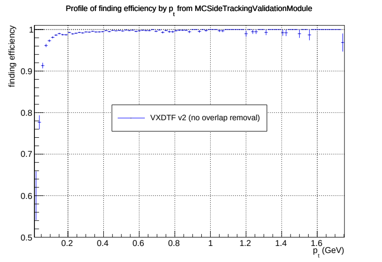
- VXDTFv2 trained on 900k $\Upsilon(4S)$ - events (no background)
- 4x4 sectors per sensor (same as in VXDTF v1)
- no PXD information used (same for VXDTF v1)
- tested: 10 k $\Upsilon(4S)$-events (no bkg., differ from trainings sample)

NOTE: when comparing VXDTFv1 with VXDTFv2

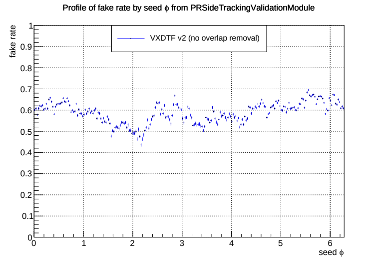
- VXDTFv1 has multiple passes:
  - three different momentum ranges ($35 < p_T <= 100 MeV$, $100 < p_T <= 400 MeV$, $400 MeV < p_T$)
  - specialization of training
  - better ability to reject false combinations
  - less combinatorial background per pass
- VXDTFv2 single pass only (multipass not implemented yet)
- VXDTFv1 uses high occupancy cut (i.e. more than 5000 overlapping TC or more than 5500 segments event will be skipped). VXDTFv2 not.

- pattern recognition alone (without overlap removal) is almost 100% efficient over a wide momentum range
- but without overlap removal high fake rate observed
- normalized to tracks found using MC information (theoretical trackable particles)

Track finding efficiency for $\Upsilon(4S)$ MC events (overlap not removed)

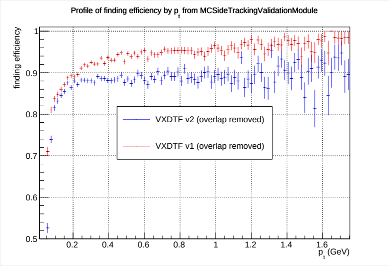


Fraction of fake tracks for $\Upsilon(4S)$ MC events (overlap not removed)

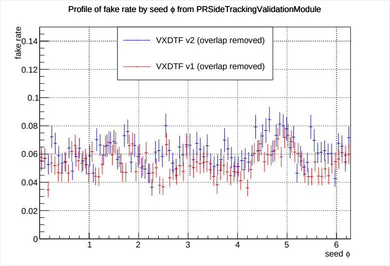


- at present performance of VXDTFv2 is slightly decreased compared to VXDTFv1
- most track candidates lost in overlap removal
- expected to improve with new track quality estimators and introduction of multiple passes

Track finding efficiency for $\Upsilon(4S)$ MC events (overlap removed)



Fraction of fake tracks for $\Upsilon(4S)$ MC events (overlap removed)

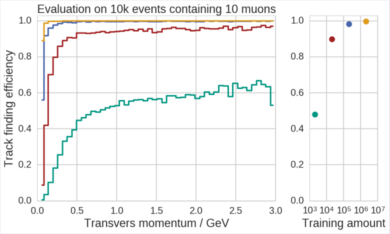


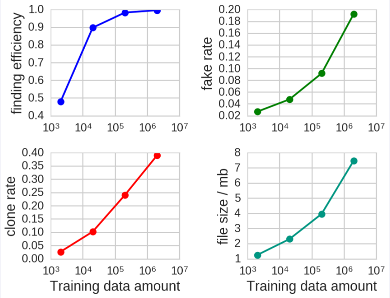
How to train a sector map?

- training the sector map is crucial for optimal performance
- many things to take into account and the optimal way yet unclear
- how many sectors per sensor, use different amount of sectors in different regions of the detector?
- selection of trainings sample:
  - $\Upsilon(4S)$ events,
  - particle gun equally distributed in phase space
  - mix of both ... ?
- use multiple passes? How many? Which momentum ranges?
- how many trainings events, too many increase the combinatorics, too few results in drop of efficiency
- use all tracks for training or only those which are "usable" for physics (i.e. give good estimate for track parameters)

- studies to optimize the training of the sector map
- performed in very clean environment (events 10 muons generated by particle gun, flat probability in momentum)

Performance of the VXD-TF2 as a function of the amount of data used in the training (overlap not removed)

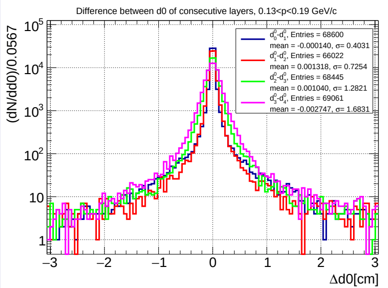




Study new Ansätze to select tracks for training of the sector map

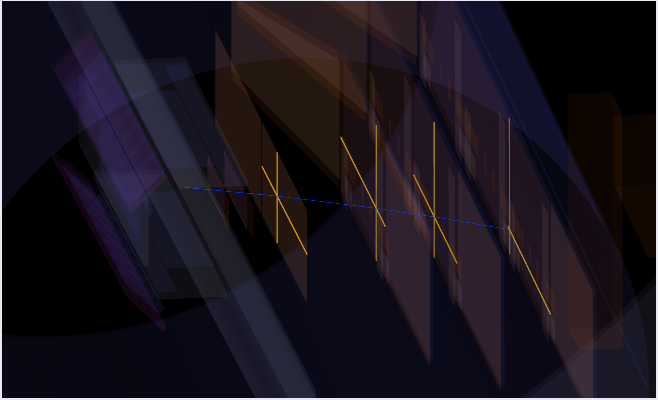
- scattering on material increases the connectivity of the sector map and thus decreases ability to rejcect bad SP combinations
- currently investigated: select tracks for training which undergone no hard scattering
- study change of track parameters between two hits on simulated events
- select tracks without drastic changes in the track parameter

As example the difference of the track parameter d_0 between two subsequent hits (d_0: impact parameter of the helix in x-y-plane at closest approach to IP)



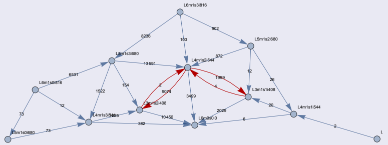
- 13th Feb - 5th Mar: test a part of the SVD+PXD in electron beam at DESY
- successful test VXDTF under data taking conditions (on HLT and offline reconstruction), both VXDTFv1 and prototpye of VXDTFv2

Event display for a reconstructed track

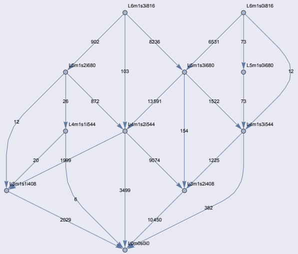


- problem: cycles in the sector map
- currently under investigation
- cycles can be removed by requiring more than 5 connections between two sectors

Graphical illustration of a sector map for the test beam geometry



Same as to the left, but require more than 5 connections

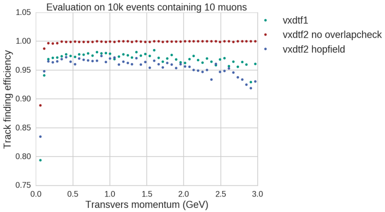


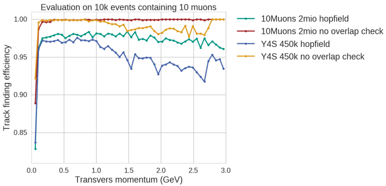
Summary and outlook

Summary

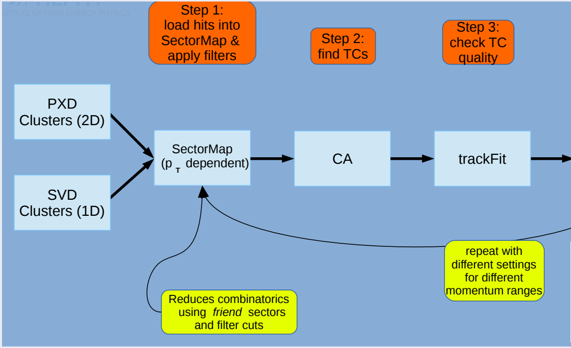
- track finding with the Belle II vertex detectors
- sector on sensor concept
- VXDTFv1: already implemented and working
- VXDTFv2: currently implemented to imporve version 1
- tested track finder (version 1 and version 2) on a test beam at DESY

BACKUP


Evaluation on 10k events containing 10 muons
Track finding efficiency
Transvers momentum (GeV)
vxdtf1
vxdtf2 no overlapcheck
vxdtf2 hopfield

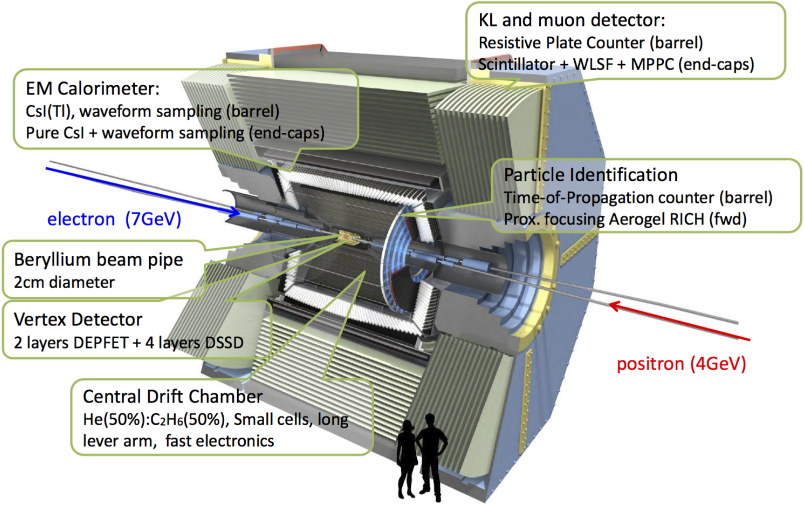

Evaluation on 10k events containing 10 muons
Track finding efficiency
Transvers momentum (GeV)
10Muons 2mio hopfield
10Muons 2mio no overlap check
Y4S 450k hopfield
Y4S 450k no overlap check

Data flow of the VXDTF version 1



- CA: Cellular Automaton
- TC: Track Candidate
- NN: Neural Network

Belle II Detector



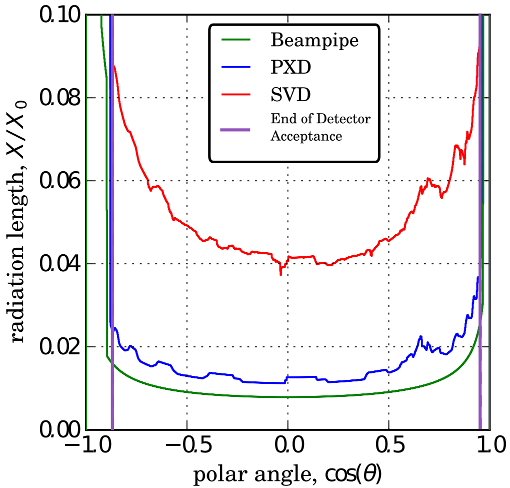


Figure 2.10: The cumulative material budget of the Beampipe, the PXD and the SVD. The violet vertical lines mark the end of the acceptance region. Plot courtesy of Martin Ritter and the Belle II Collaboration.

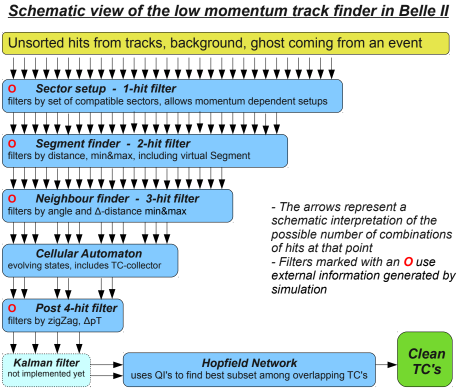


Figure 3.2.: A schematic view representing the parts of the track finder and their use in reducing combinatorics