Karlsruhe Institute of Technology

KSETA/GK Workshop

# INFORMAL INTRODUCTION TO TRACKING IN HEP

April 2019

Michael Eliachevitchg | ETP – KIT

# WHAT IS TRACKING AND WHY DO WE NEED IT?

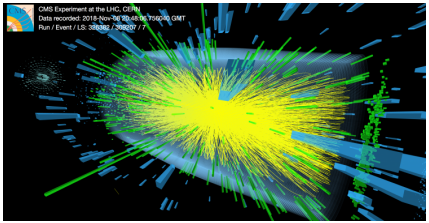
- HEP accelerator: produce particles in collisions
- goal: reconstruct physically meaningful particle properties and event topology
  - energy, momentum-vector, primary and secondary vertices, dE/dXn ($\rightarrow$ type;)
- tracking detectors: measure ionization from charged particles (stable: e, $\mu$, K, $\pi$, p)
- track reconstruction (tracking):
  - find tracks (pattern recognition)
  - extract parameters (fitting)

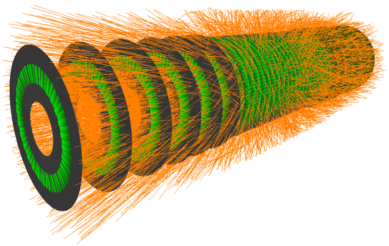
# WHAT WE MEASURE

- energy depositions: "hits"
- 2D measurements: pixel detectors
- 1D measurements: strip detectors, wire chamber (measurements are times)
- binary or non-binary information (useful for dE/dX, weighting)

# WHY IS DIFFICULT? EXAMPLE: (HL)-LHC

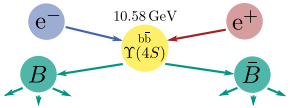
- pile-up and underlying event $\rightarrow$ large number of tracks from IP
  - combinatorics + ambiguities
- lumi + large cross section $\rightarrow$ high event rate $\rightarrow$ little time per event
- HLLHC: current CPU-time surpasses computing budget
  - faster algorithms required

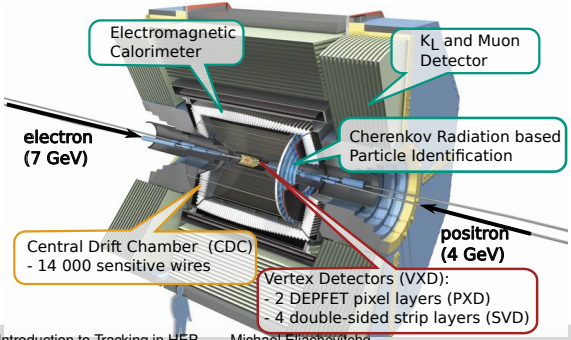


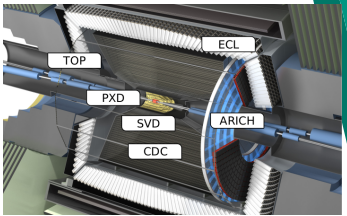



# INTERLUDE: BELLE II

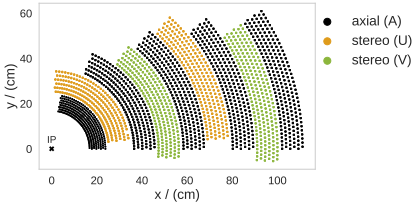
- $e^+e^-$ B-factory operating on the Y(4S) resonance at 11.58 GeV
- asymmetric beam energies to resolve B decay vertices

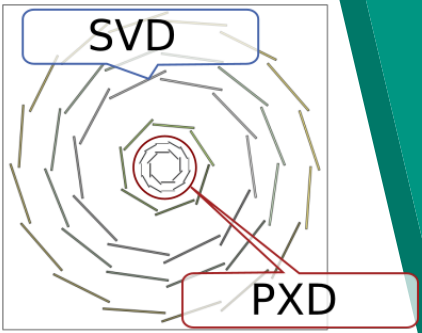


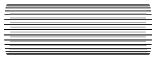



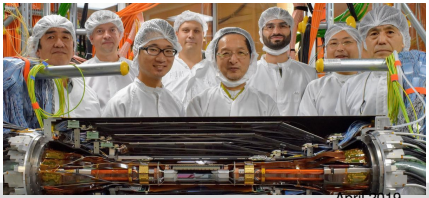



# TRACKING SUBDETECTORS





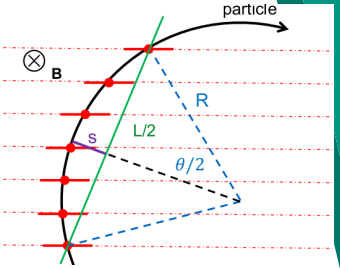
# WHY IS DIFFICULT? EXAMPLE: BELLE II

- Y(4S) decays: only **11 tracks** per event
- high luminosity $\rightarrow$ large beam-induced background
  - PXD: **$10^4$ backkground hits** expected
- $\sqrt{s} = 10.58 GeV$: **low-momentum** tracks
- curl in tracking system
- scattering significant
- physics analyzes: require reconstruction of whole event

# HOW I LEARNED IT

- curvature in B-field allows momentum estimation
- $p_T = \frac{R}{qB}$, $p_T = p \cos theta$
- estimate curvature from sagitta s

$$\frac{\sigma_{p_\perp}}{\mathrm{p}_\perp} = \underbrace{a' \frac{1}{BL^2} p_\perp}_{\text{track measure-ment}} \oplus \underbrace{b' \frac{1}{\sqrt{LX_0}B}}_{\text{multiple scattering}} = a p_\perp \oplus \mathbf{b}$$



# HOW IT'S DONE: LOCAL AND GLOBAL ALGORITHMS

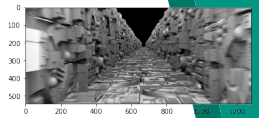
- global
  - take all hits into account at the same
  - assume helical tracks from IP
  - stable with backgrounds and gaps in hit pattern
- local
  - build track via local relations between hits
  - can handle large scattering and tracks not from IP

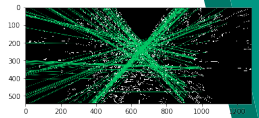
# LEGENDRE ALGORITHM: HOUGH TRANSFORM

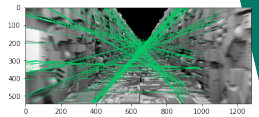
- Legendre algorithm is extension of **Hough Transform**
  - transform xy-space into $\rho\theta$-space, where each line is represented as a point
  - $\rho = x \cos\theta + y \sin\theta$
  - for each point in the original space, draw all points in the $\rho\theta$-space consistent with a line through through that point

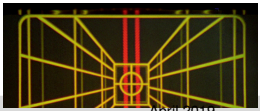
;**** Algorithmus von Wikipedia :B$_{\text{ignoreheading}}$:
:BEAMER$_{\text{env}}$: ignoreheading

```
foreach pixel != 0:
   for theta := 0 to pi:
      rho := pixel_x * cos(theta)
             + pixel_y * sin(theta)
```



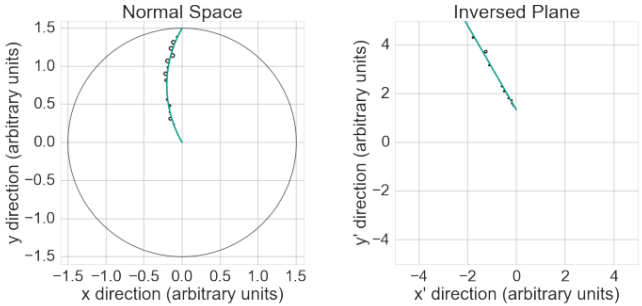



# LEGENDRE: LINEARIZE CIRCULAR TRACKS

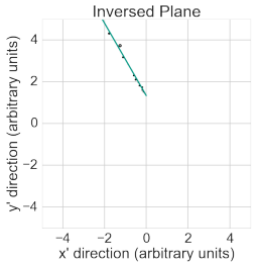
- circles through origin again described by only two parameters (in 2D)
- do the same thing again

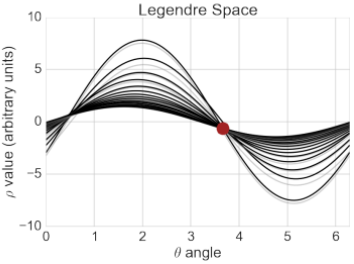
$$x' = \frac{2x}{x^2 + y^2 - d^2} \qquad y' = \frac{2y}{x^2 + y^2 - d^2} \qquad d' = \frac{2d}{x^2 + y^2 - d^2}$$



# LEGENDRE

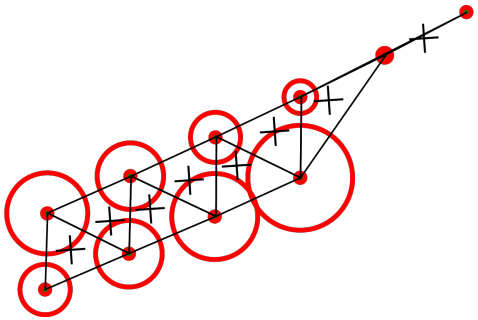
$$\rho(\theta) = x' \cos(\theta) + y' \sin(\theta) \pm d'$$



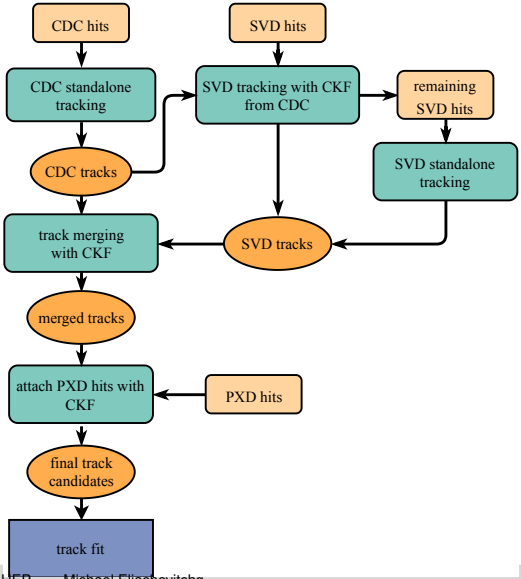
# LOCAL: CELLULAR AUTOMATON

- search for longest path in graph (DAG) with nodes and edges
- length = sum of weightes edges and nodes
- e.g. CDC: nodes are hit triplets, edges from BDT

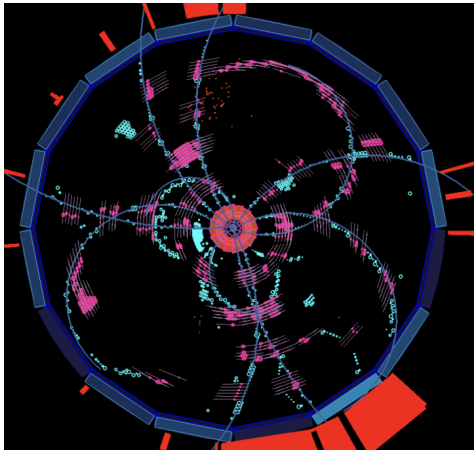


# LOCAL: COMBINATORIAL KALMAN FILTER

# HOW IT ALL COMES TOGETHER



# WHEN IT WORKS



# EXTRA: DEEP LEARNING APPROACHES TO TRACK FINDING