
SPD meeting,
13 May 2020

# Status of track reconstruction for SPD experiment

V. Andreev

# Introduction

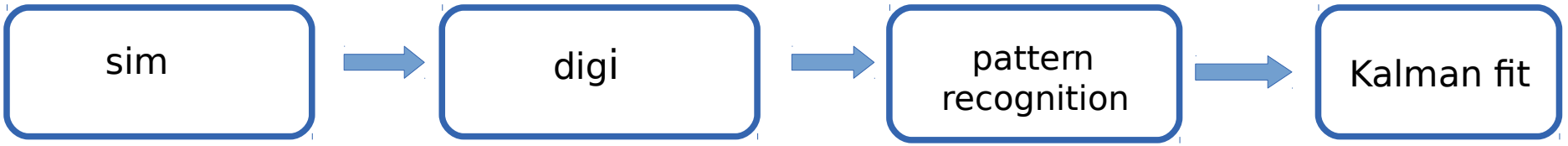
- Track reconstruction is traditionally divided into separate sub-tasks:
  - track finding
  - track fitting  (is already introduced in SPD on the base of Kalman filter)

- Track finding (pattern recognition):
  - division of set of measurements in a tracking detectors into subsets
  - each subset contains measurements believed to originate from the same particle

- Track fitting:
  - starts  with the measurements inside one subset as provided by the track finder
  - aims to estimate a  track parameters using the information from the measurements
  - evaluates the quality and final acceptance of the track candidate

# Pattern recognition (realization)

1. all change are done inside SPDroot software




2. 1-st step (simulation) => produce sim-hits using SPDroot with different generators and tracker's geometries

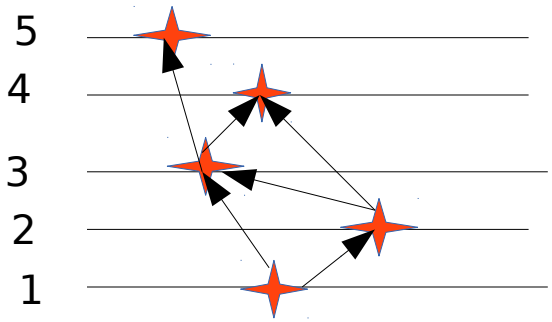
3. 2-nd step => produce digi-hits (contains x-y coordinates on detector plane with smearing, detector position and etc.)

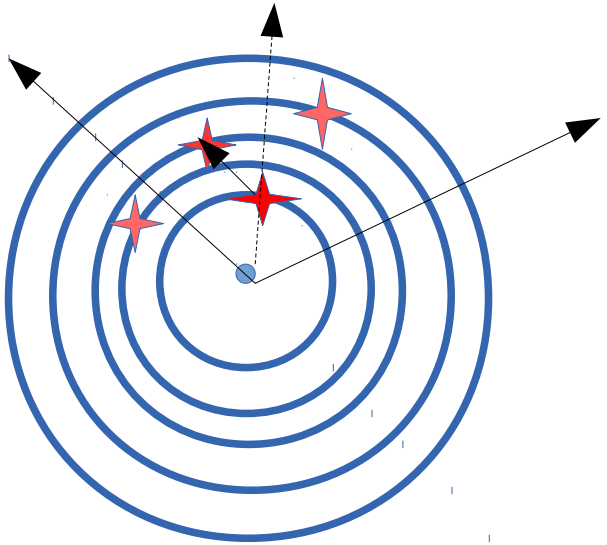
4. 3-d -step (pattern recognition and fit) => produce track candidates and do final Kalman fit

5. pattern recognition:
   - construct track seed in vertex detector;
   - add straw detector hits to vertex track seed;
   - create track seed candidates (contains as vertex and tracker hits).

# Track seed in vertex detector

1. Silicon vertex detector – 5 cylinders in barrel part and 5 disks in endcap part
   (plane pixel detector, xy- coordinate )



2. next 2-points combinations between hits are considered:
   a) 1 layer <=> 2 layer  b) 1 layer <=> 3 layer
   c) 2 layer <=> 3 layer  d) 2 layer <=> 4 layer
   e) 3 layer <=> 4 layer

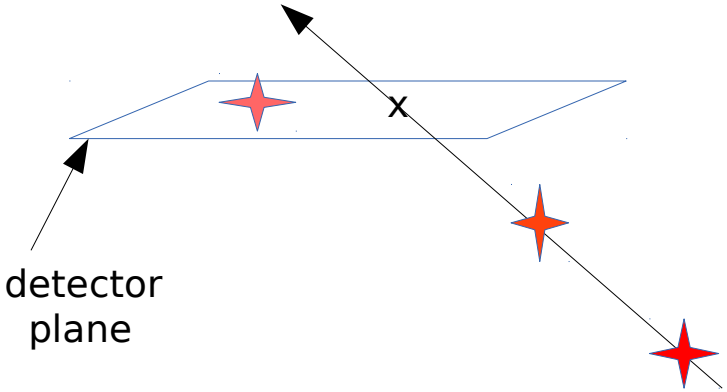
3. use points in next layer for 2-points seed only inside some phi-range

4. start 2-points seed construction from 1-st layer =>
   (1,2); (1,3); (2,3); (2,4); (3,4);

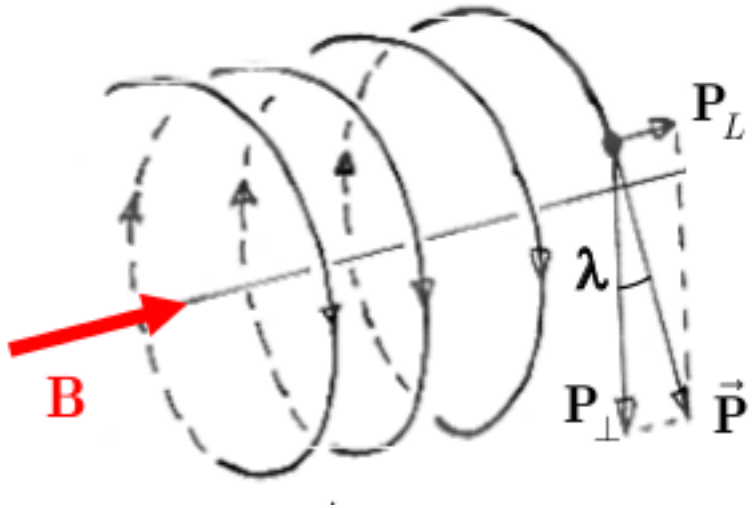
5. add 3-d point to primary 2-points seed:
   (1,2) <=> 3 layer; (1,2) <=> 4-th layer

# Track seed in vertex detector (2)

1. use 2-points seed => construct space line and check the distance between point on the next layer and point of line crossing with detector plane. Accept point if distance is inside some range => first level of selection




2. but particle trajectory in the vertex detector area is not straight line as the pseudo-solenoidal field is considered as base magnetic field configuration for SPD and track trajectory is described by helix with radius R

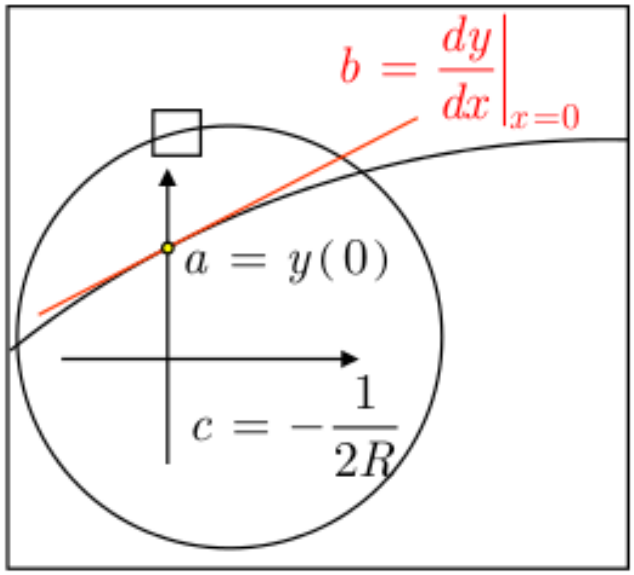



$$R(m) = \frac{p_\perp(GeV)}{0.3B(T)}$$

# Track seed in vertex detector (3)

1. trajectory projection on a plane perpendicular to the magnetic field is a circle

2. propose to use parabolic approximation of the trajectory projection
   (in the range of vertex detector)




$$y = y_o + \sqrt{R^2 - (x - x_o)^2}$$

$R^2 >> (x - x_o)^2$

$$y \approx y_o + R\left(1 - \frac{(x - x_o)^2}{2R^2}\right)$$

$$y = \left(y_o + R - \frac{x_o^2}{2R}\right) + \frac{x_o}{R}x - \frac{1}{2R}x^2$$

$$y = a + bx + cx^2$$

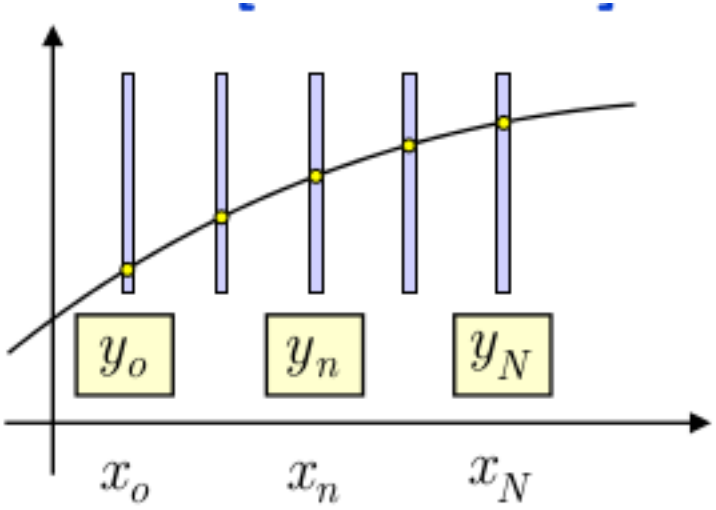
R ~4.5 m for Pt = 0.5 GeV, detector radius ~25 cm

3. The parameters a, b, c are intercept at the origin, slope at the origin and radius of curvature (momentum)

4. transform plane xy-coordinate of hits to space xyz-coordinate (using detector position) and then do projection on XY-coordinate plane where partcle trajectory should be circle

# Track seed in vertex detector (4)

1. parabola parameters and chi2 can be estimated using the next expression:




$$\begin{cases} a\sum_{i=1}^{n} x_i^2 + b\sum_{i=1}^{n} x_i + nc = \sum_{i=1}^{n} y_i, \\ a\sum_{i=1}^{n} x_i^3 + b\sum_{i=1}^{n} x_i^2 + c\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} x_i y_i, \\ a\sum_{i=1}^{n} x_i^4 + b\sum_{i=1}^{n} x_i^3 + c\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i^2 y_i; \end{cases}$$

$$\chi^2 = \sum_{n=0}^{N} \frac{\left(y_n - a - bx_n - cx_n^2\right)^2}{\sigma_n^2}$$

2. new point is added on the base of chi2 value estimation

3. this procedure starts from 2-points seed => produce 3-points seed => use 3-points seed => produces 4-points seed =>use 4-points seed => produce 5-points seed

4. then merging and cleaning procedures are applied => as the final result the set of 5,4,3 - points vertex track seeds are produced

## Pattern recognition
## (add straw tracker points )

1. use vertex detector hits for primary track seed construction for Kalman fitter

2. do Kalman fit (using track seed parameters and vertex hits) then extrapolate this track (after Kalman fit) to tracker detector (straw layer)

3. track is extrapolated to 1-st straw layer (from IP) and after straw hits are checked on consistence with this track seed (distance and chi2 criteria)

4. add “good” straw hit to track seed points, update track seed parameters (do new Kalman fit) and then do extrapolation to the next straw layer

5. if 2 or more straw hits in one straw layer are consisted with the track seed => the new track seed candidate is created (after fitting and extrapolated procedures are applied for all new created candidates)

6. as the result, finally the“big” number of track seed candidates are produced which contains vertex and straw tracker hits

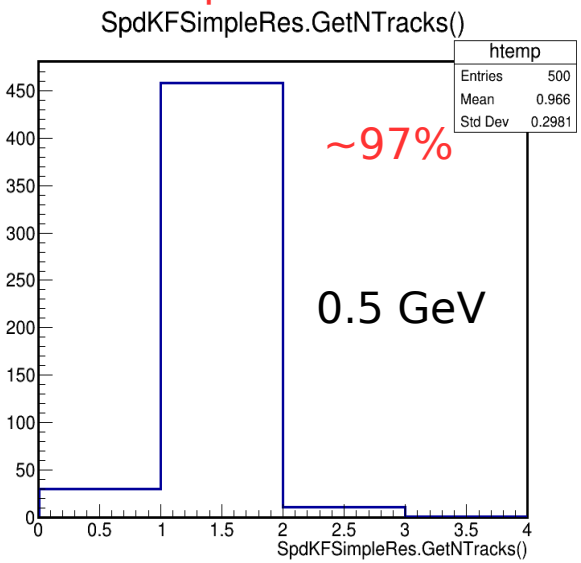
7. clean track seed candidates (on the base of chi2 and number of hits)

8. do final Kalman fit for track seed candidates (using all hits)

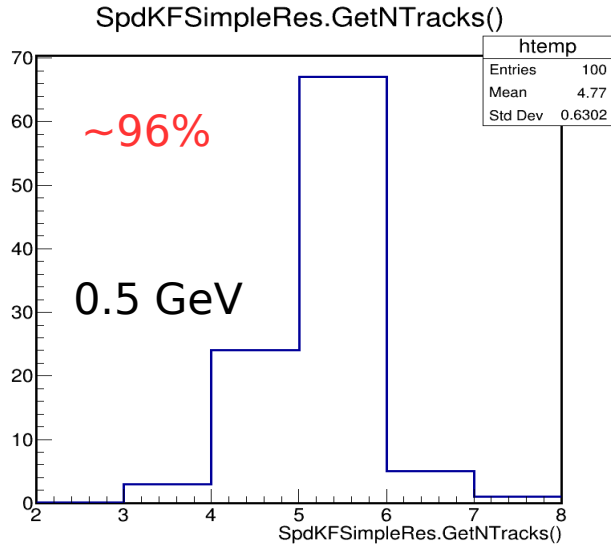
# Track reconstruction (some preliminary results )

1. the next  data sets were produced: 0.5 GeV and 1 GeV muons with 1, 5 and 10 particles for event
2. estimation of track reconstruction efficiency for 0.5 GeV and 1 GeV

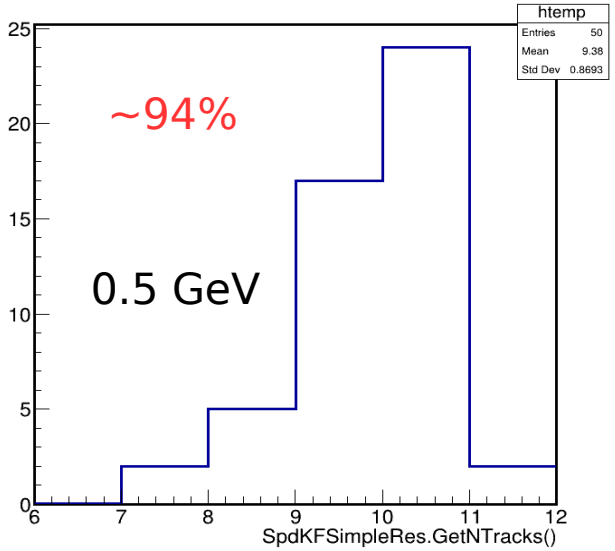
1 part./event

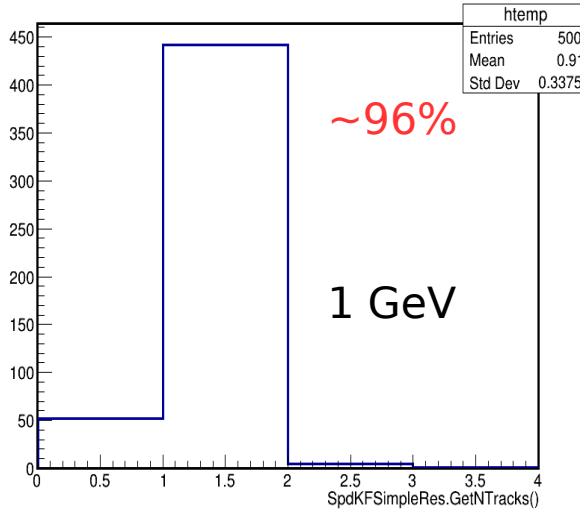


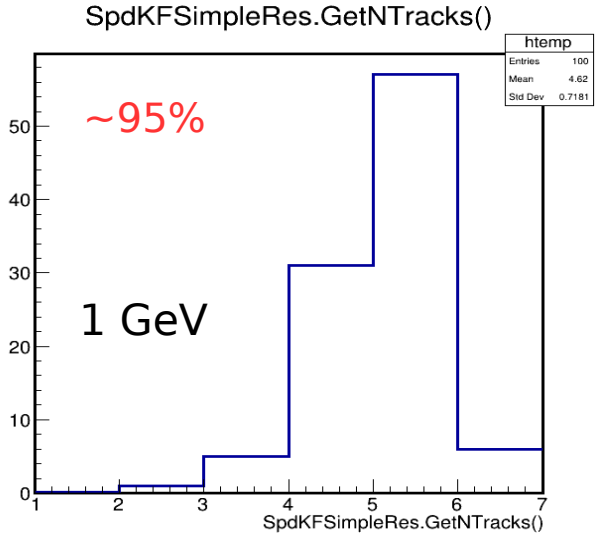

5 part./event

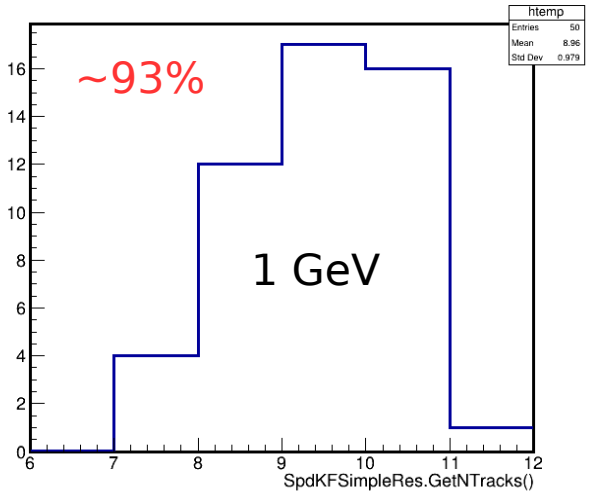



10 part./event



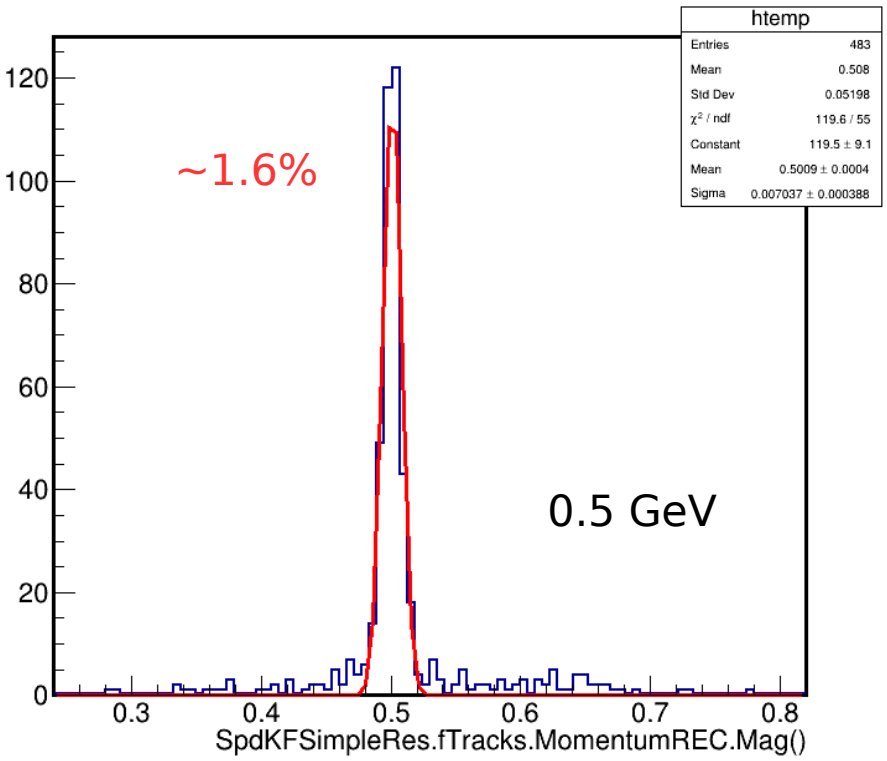



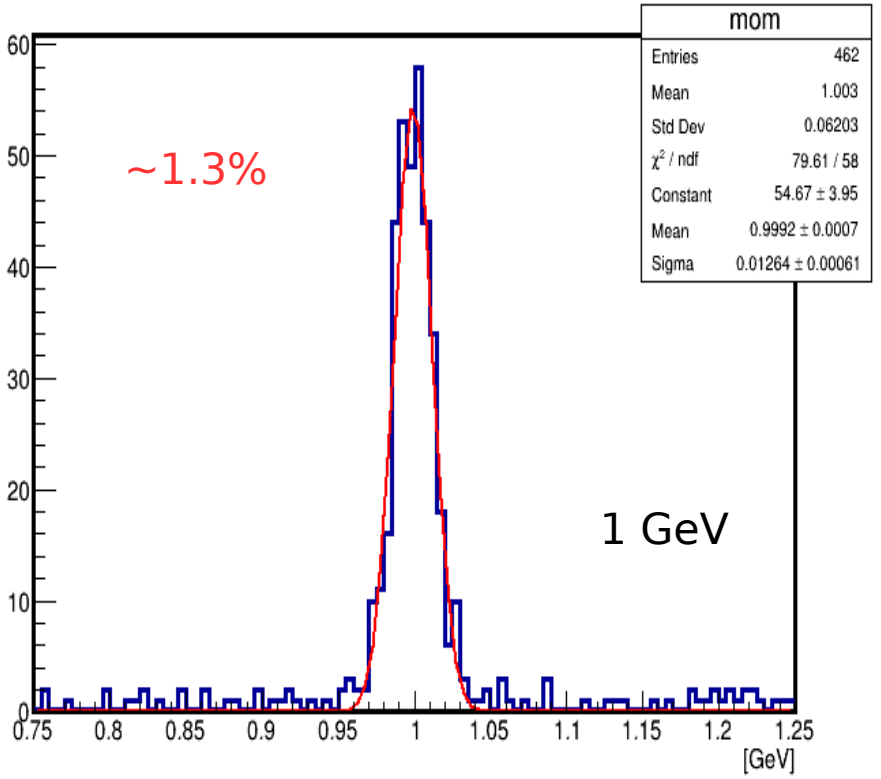



# Track reconstruction (some preliminary results )

1. track momentum reconstruction  for 0.5 GeV and 1 GeV



# Plans

1) check and optimize track seed finding in vertex detector

2) optimize finding hits procedure in straw detector

3) correct output format

4) track reconstruction release (1-st) in 1 month

# Backup slides

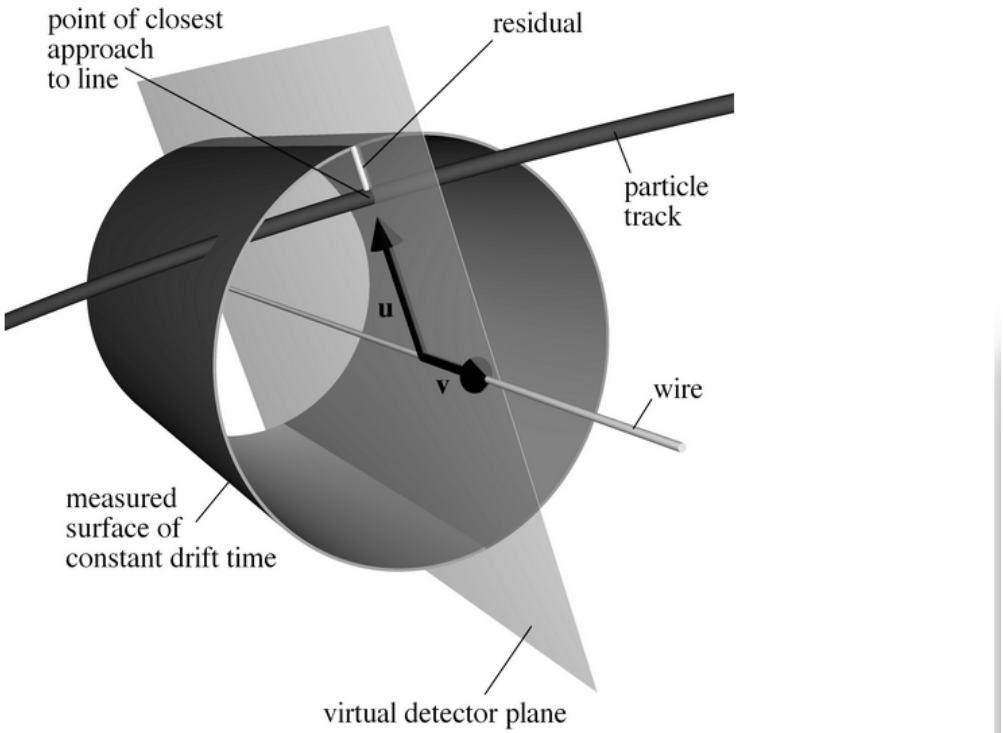



Figure 3: Virtual detector plane (spanning vectors $\vec{u}$ and $\vec{v}$) for a wire-based drift detector.