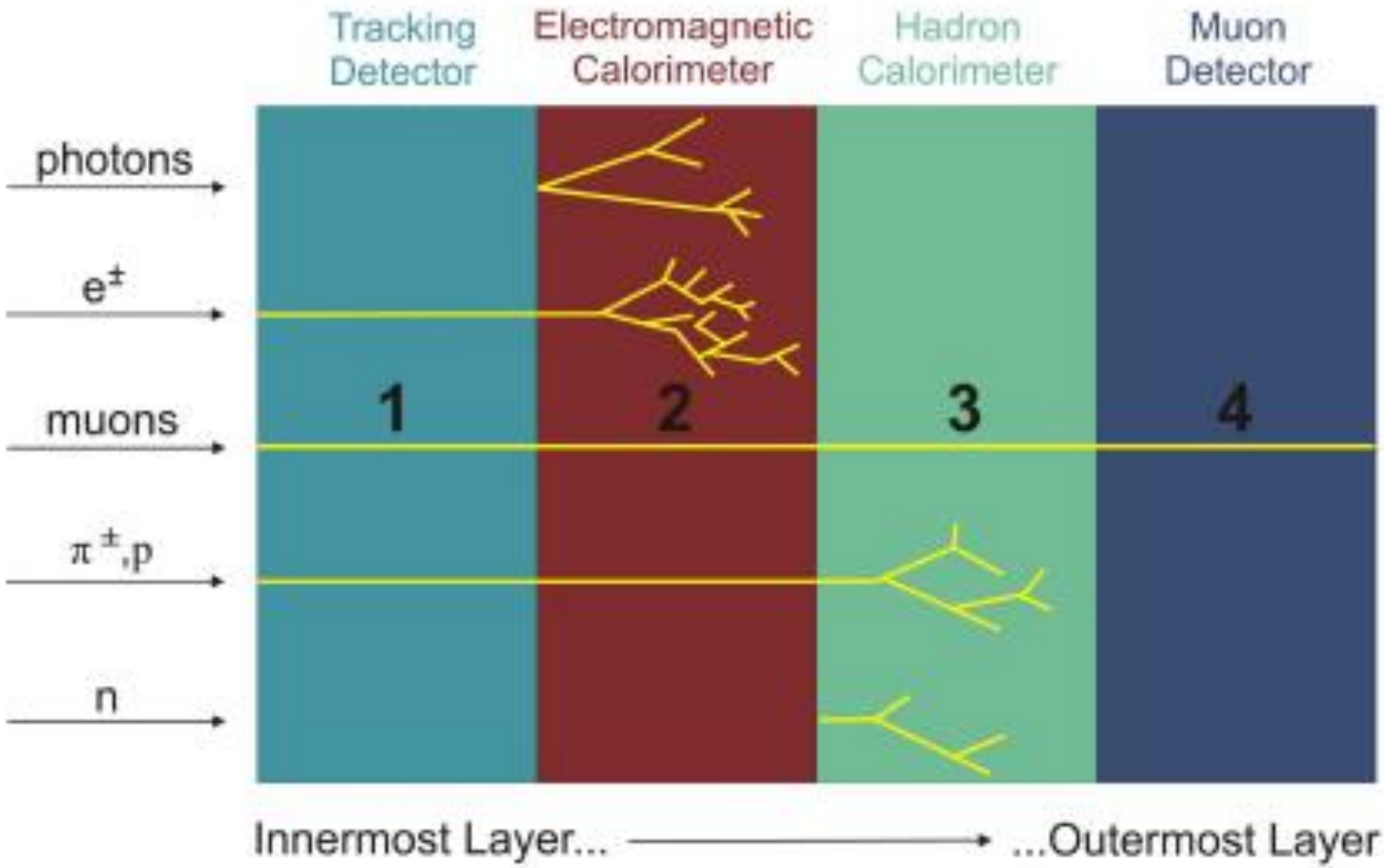
BESIII漂移室径迹重建

伍灵慧
wulh@ihep.ac.cn
BESIII MDC 软件组
2018-9-28

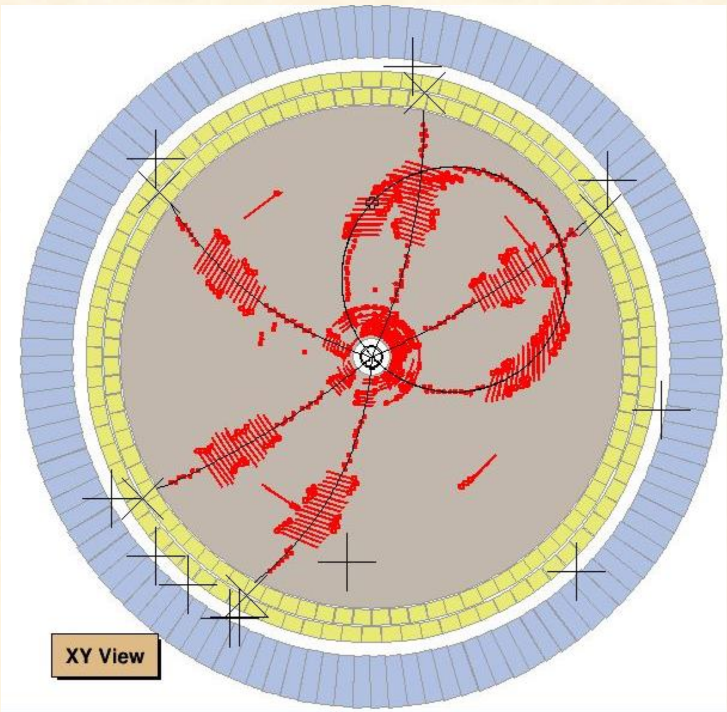
内容

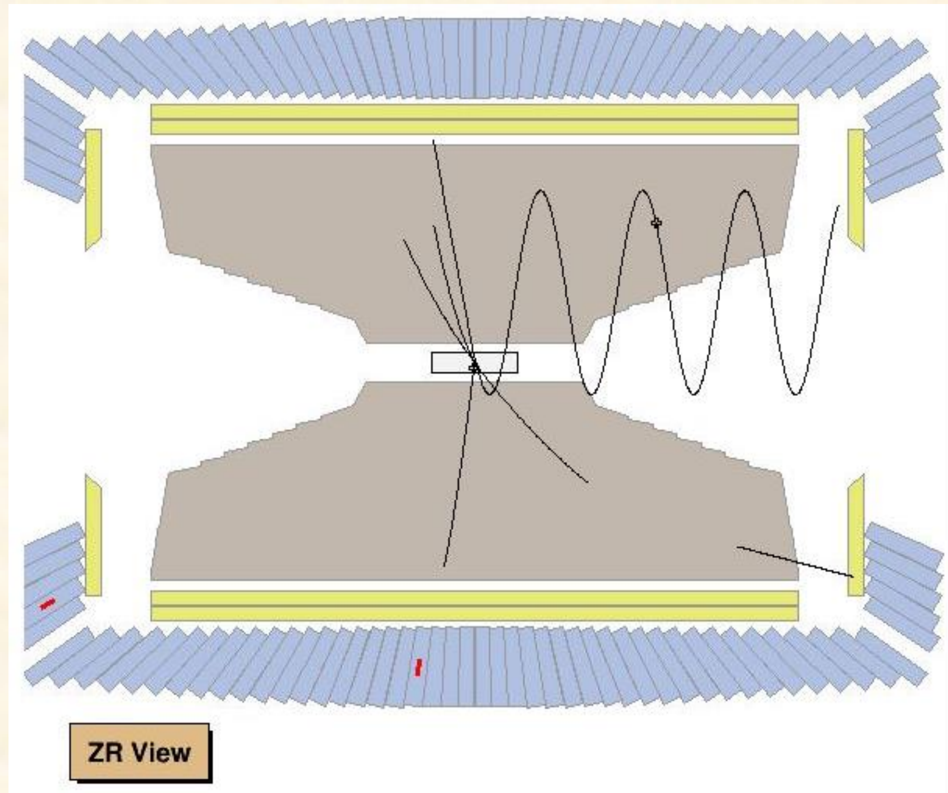
- 径迹重建目的
- 常用寻迹方法
- BESIII实验中的径迹重建
  - 径迹寻找
  - 径迹拟合
  - 寻迹效率
- 总结

Interaction of particles in diff. detector components:



带电粒子径迹探测

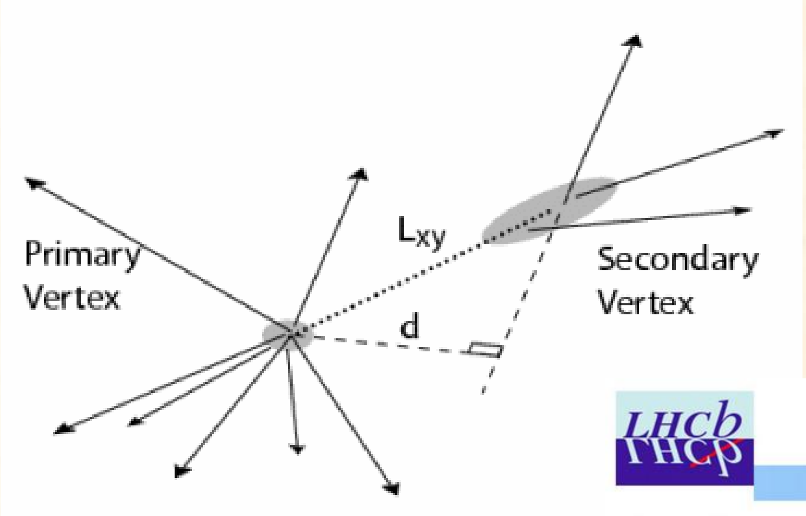


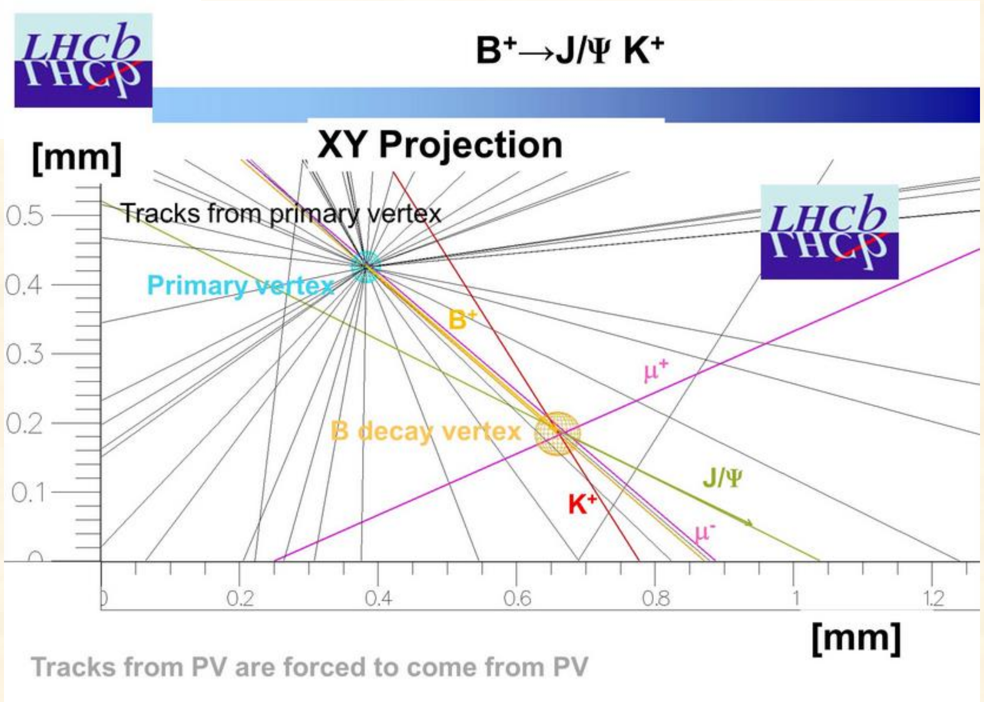


径迹重建目的

- 顶点测量
- 动量测量
- 径迹外推与匹配

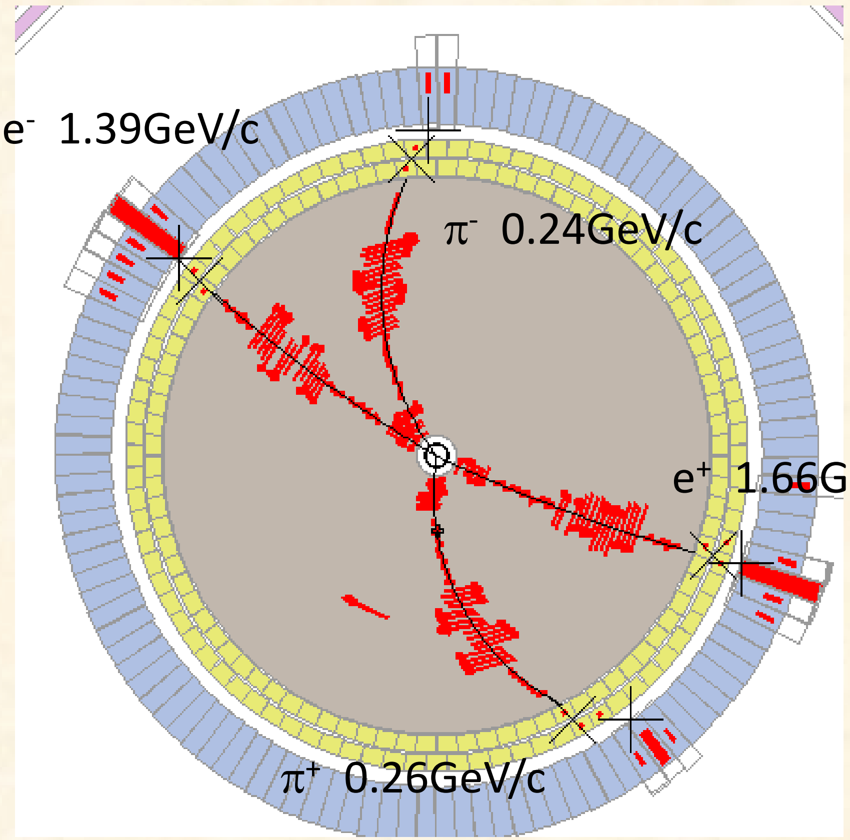
顶点测量

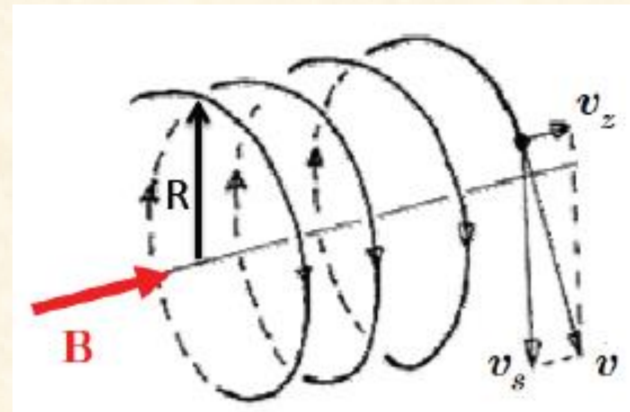

Primary
Vertex
L_{xy}
d
Secondary
Vertex


LHCb
B⁺→J/Ψ K⁺
XY Projection
[mm]
Tracks from primary vertex
Primary vertex
B⁺
μ⁺
B decay vertex
J/Ψ
K⁺
μ⁻
0.5
0.4
0.3
0.2
0.1
0.2
0.4
0.6
0.8
1
1.2
[mm]
Tracks from PV are forced to come from PV

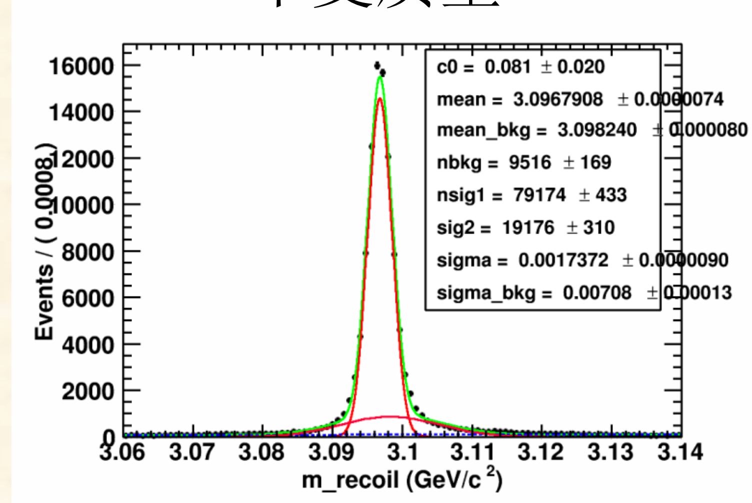
动量测量

$e^+e^- \rightarrow \pi\pi J/\psi,\ J/\psi \rightarrow e^+e^-$



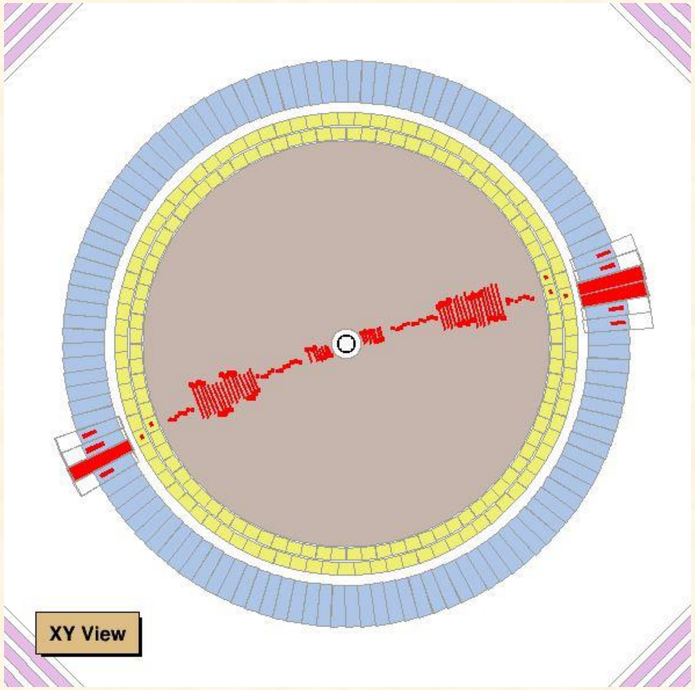


不变质量

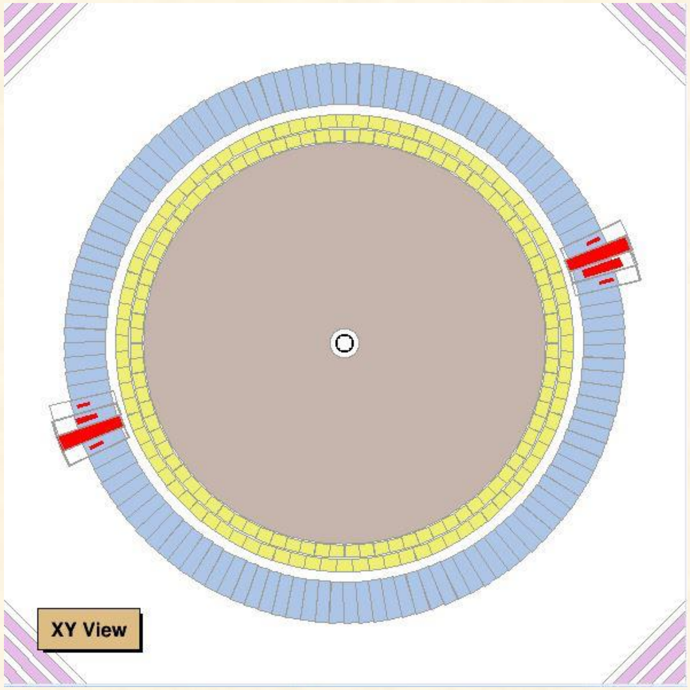


径迹外推和匹配

e+e- → e+e-

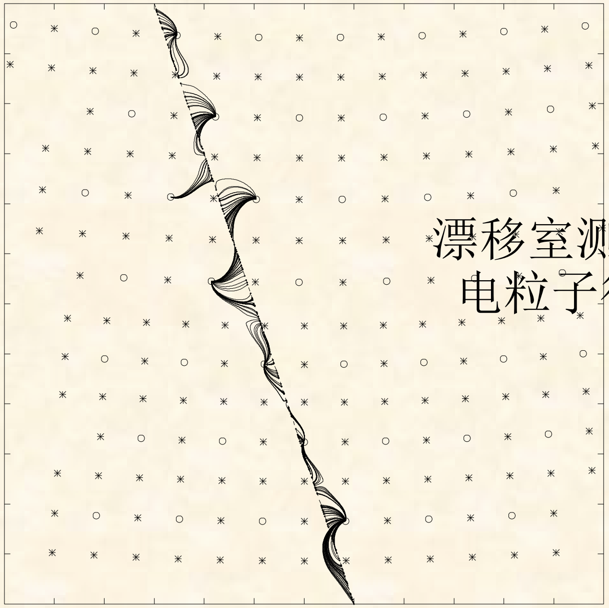


e+e- → γγ



径迹探测器

- 气体类
  - 稻草管
  - 多丝正比室
  - 漂移室
  - 时间投影室
  - GEM探测器
- 半导体类
  - 硅微条
  - 硅像素

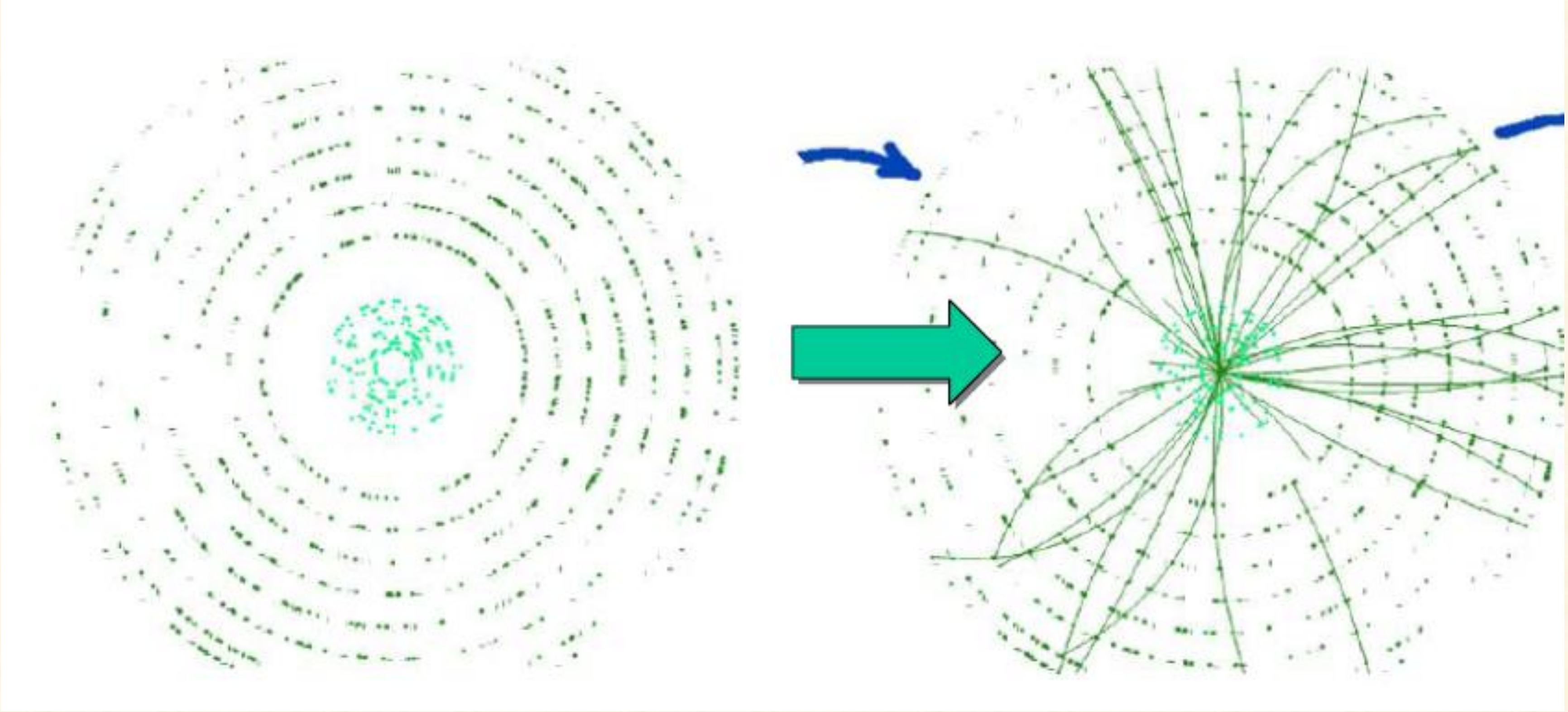


漂移室测量带电粒子径迹

径迹重建

- 径迹寻找(Track Finding)
  - 寻找可能的入射粒子径迹以及属于该径迹的击中
- 径迹拟合(Track Fit)
  - 通过拟合确定径迹参数及误差矩阵

径迹寻找



Connecting The Dots -- a workshop on tracking

CONNECTING THE DOTS 2018
4TH INTERNATIONAL WORKSHOP
20-22 MARCH 2018
UNIVERSITY OF WASHINGTON, SEATTLE, USA

Connecting The Dots 2018

20-22 March 2018
University of Washington Seattle
US/Pacific timezone

Search...

- Overview
- Scientific Programme
- Timetable
- Seattle tracking hackathon
  - TrackML hackathon registration
- Registration
- Participant List
- Contribution List

This is a workshop on track reconstruction and other problems in pattern recognition in sparsely sampled data. The workshop is intended to be inclusive across other disciplines wherever similar problems arise. The main focus will be on pattern recognition and machine learning problems that arise e.g. in the reconstruction of particle tracks or jets in high energy physics experiments.

This 2018 edition is the 4th of the Connecting The Dot series (see CTD2015 Berkeley, CTD2016 Vienna, WIT/CTD2017 LAL-Orsay).

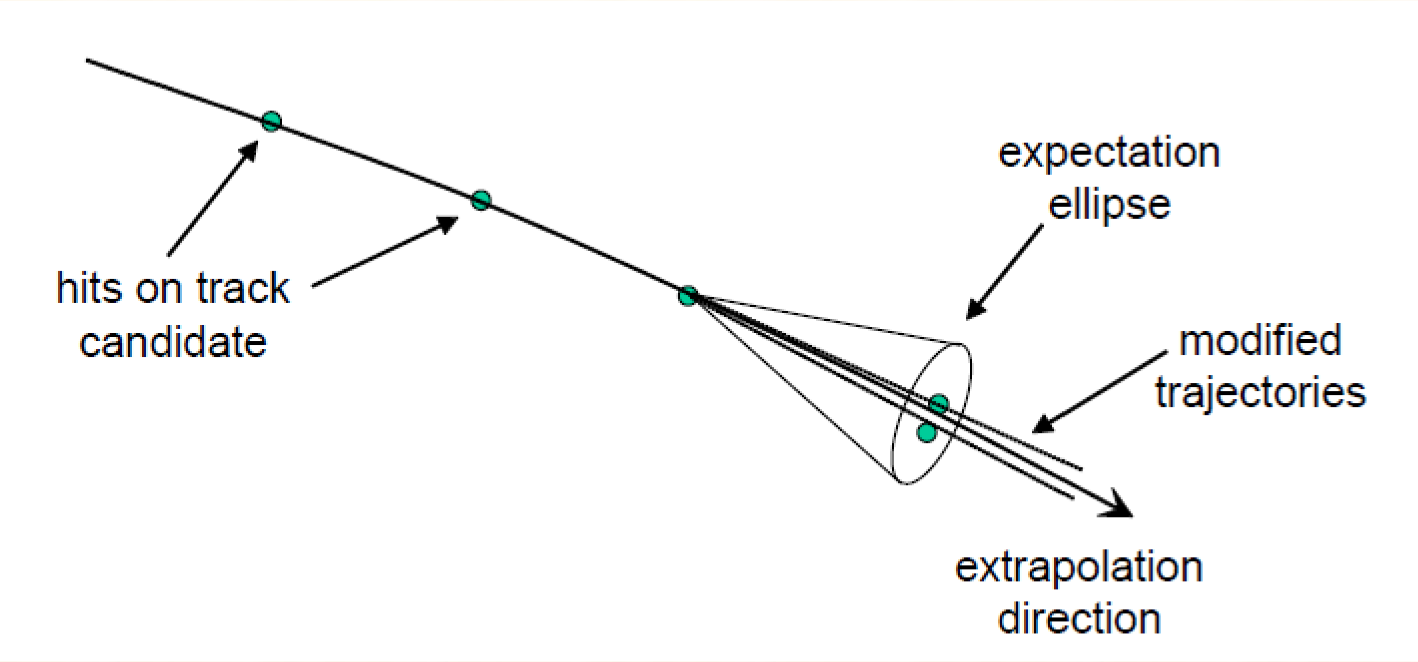
The workshop is plenary sessions only, with a mix of invited talks and accepted contributions. There will also be a Poster session.

Wifi is available on site, eduroam credentials, from your institution or CERN, are recommended (but not mandatory)

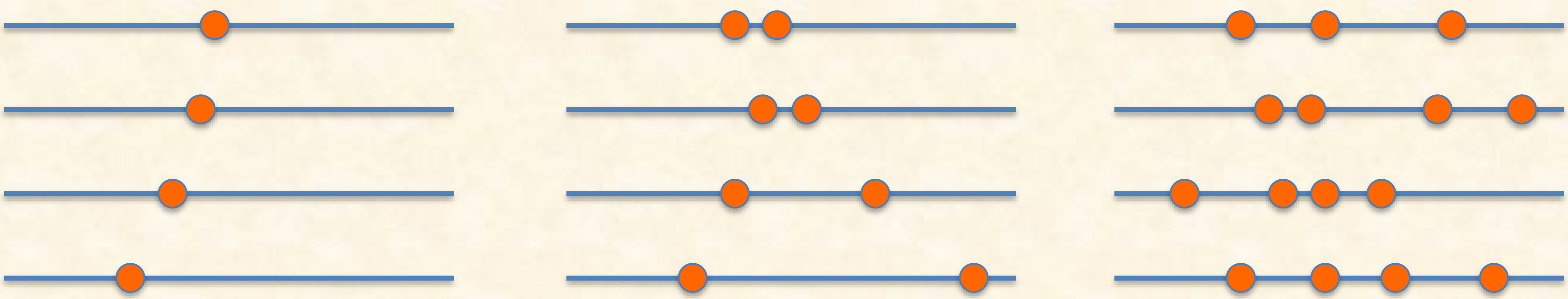
寻迹，怎么寻？

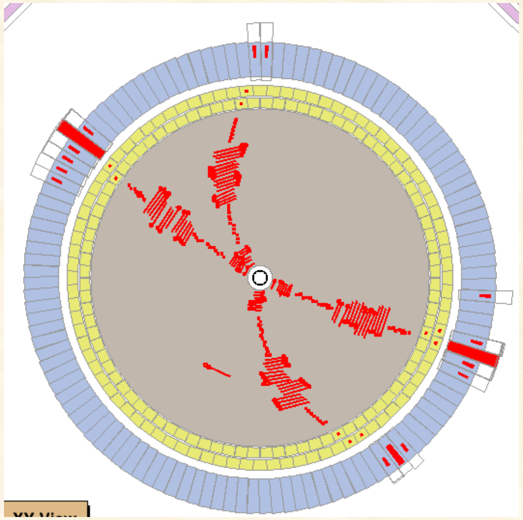
- 径迹跟踪：

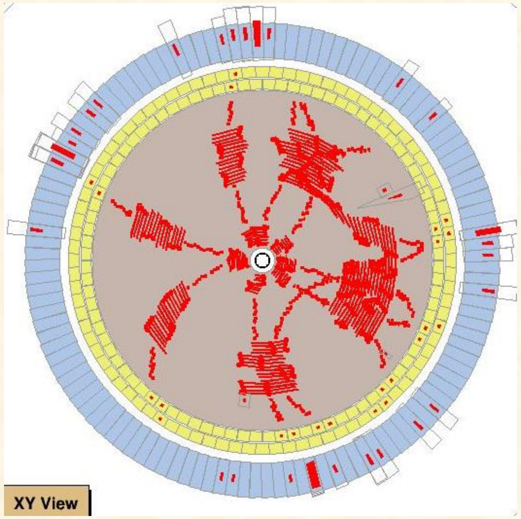
  候选种子击中 → 预测 → 加入新的测量点

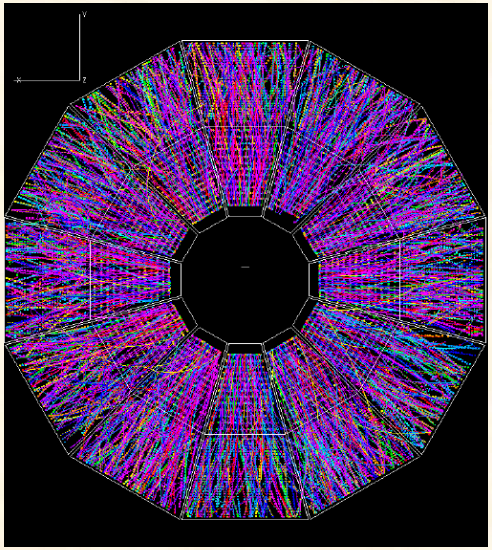


问题：随着测量点的增加，组合数急剧上升，计算时间急剧增长





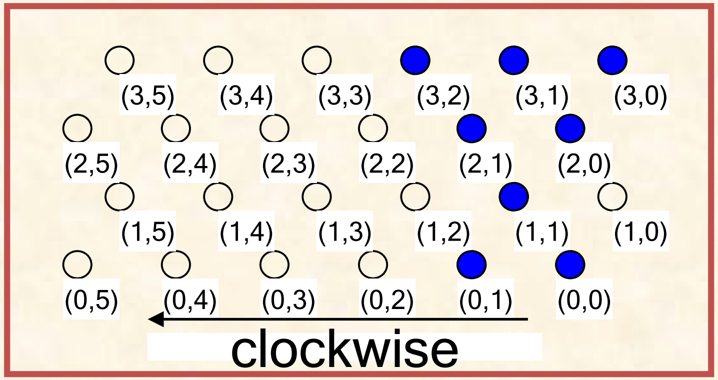




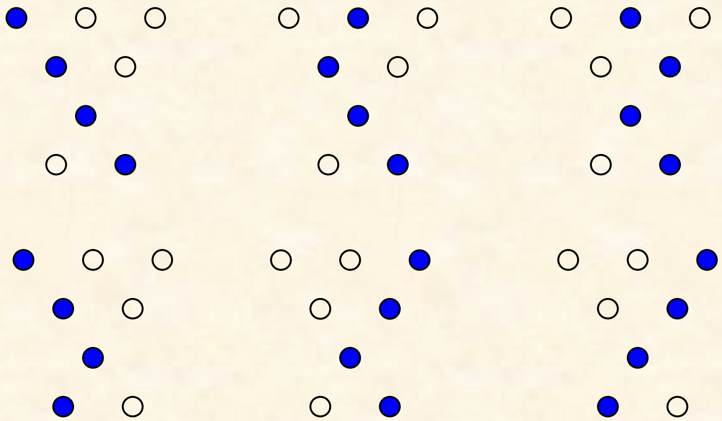
寻迹方法－模版匹配

- 构造模版字典，通过查字典的方式找出径迹段

Cells for pattern



4-hit pattern dictionary



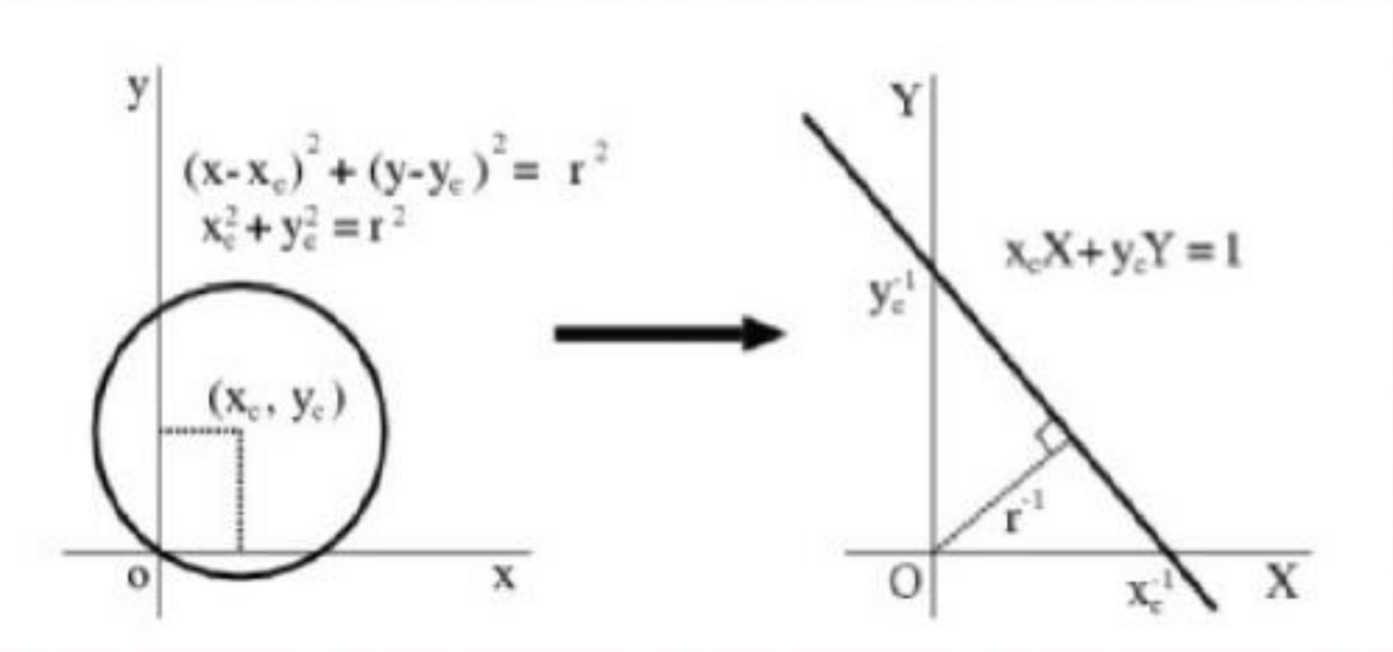
寻迹方法 – 数学变换

- 共形变换（Conformal mapping）
- 勒让德变换（Legendre transform）
- 霍夫变换（Hough transform）

共形变换

- 通过共形变换，圆径迹变换为直线

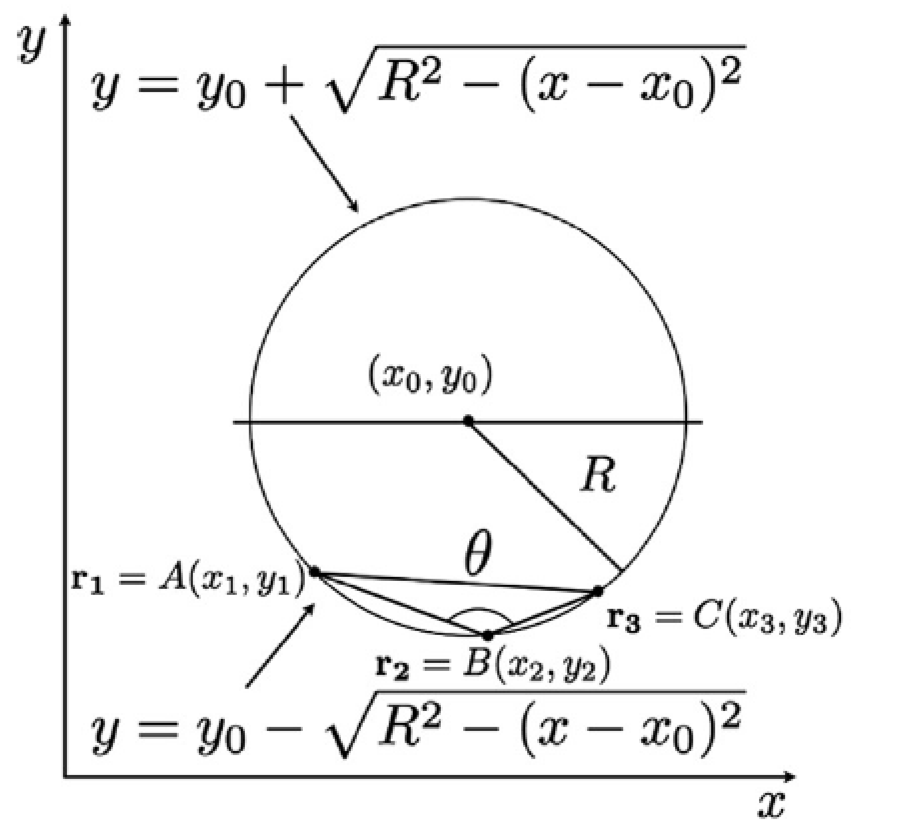
圆方程: $(x-x_c)^2 + (y-y_c)^2 = r^2$ ⟶ $X = 2x/(x^2+y^2)$, $Y = 2y/(x^2+y^2)$

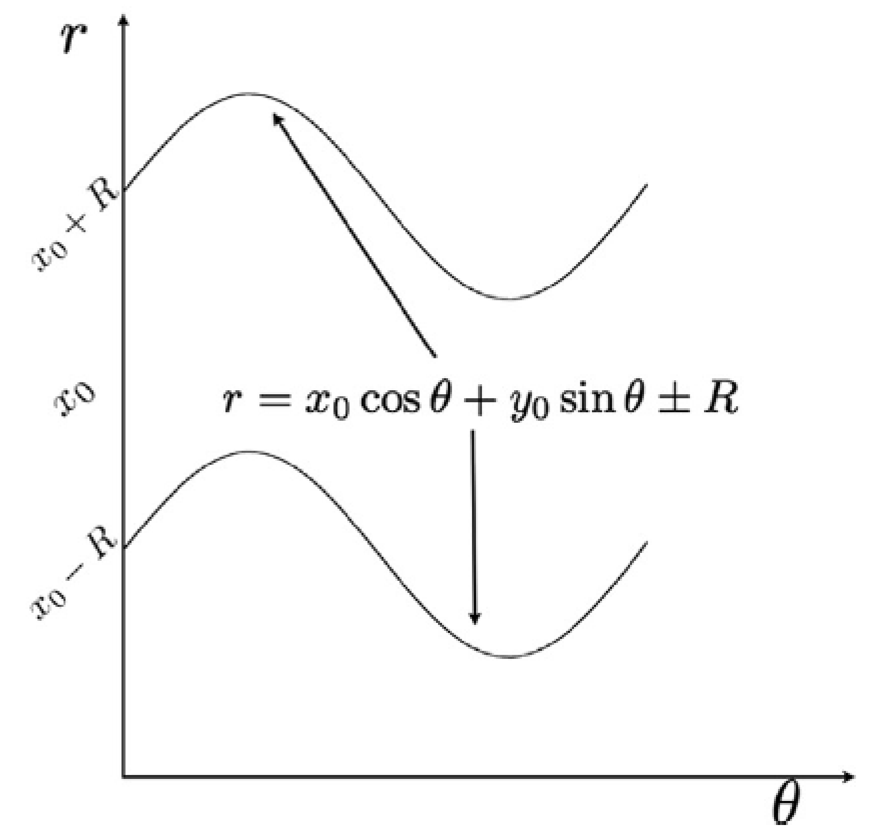


勒让德变换

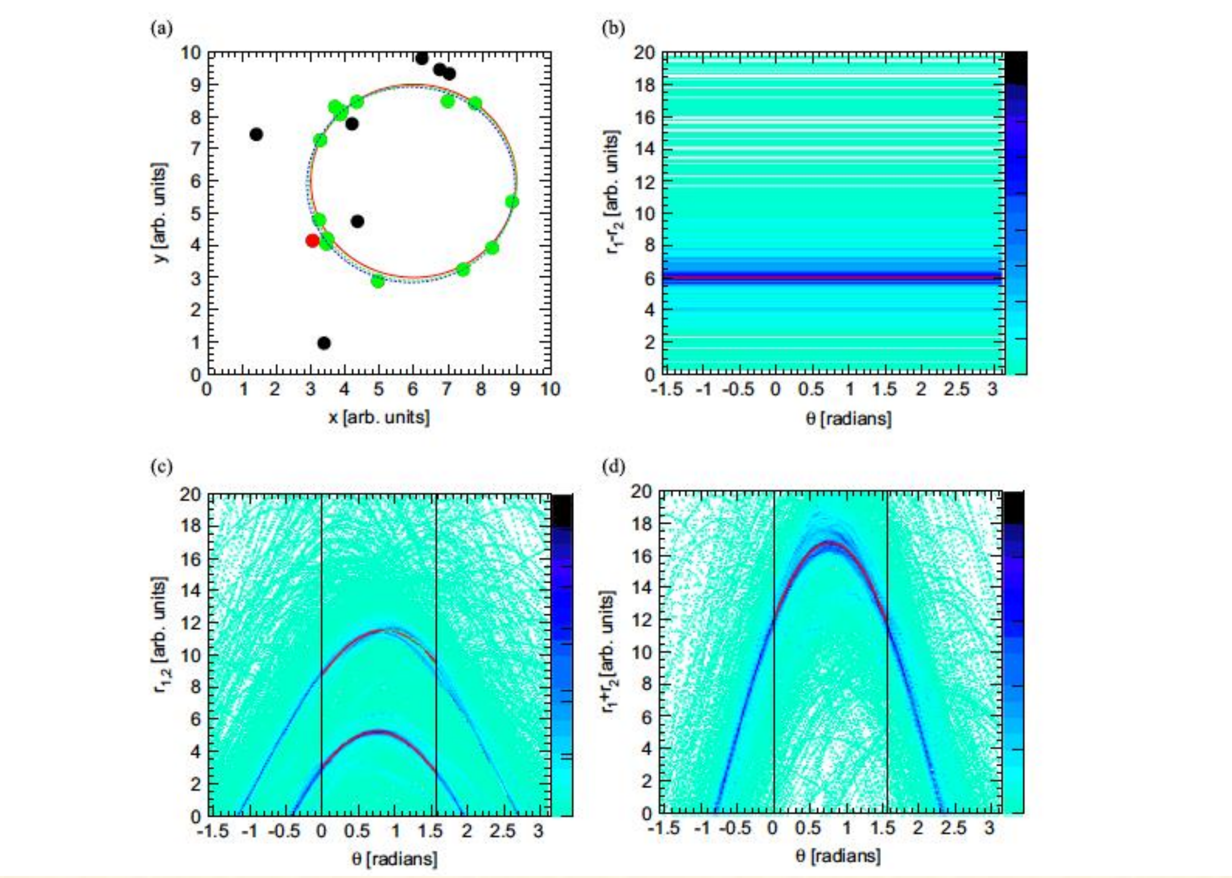
NIM A 745(2014) 16-23

Real space　　Legendre transformation space



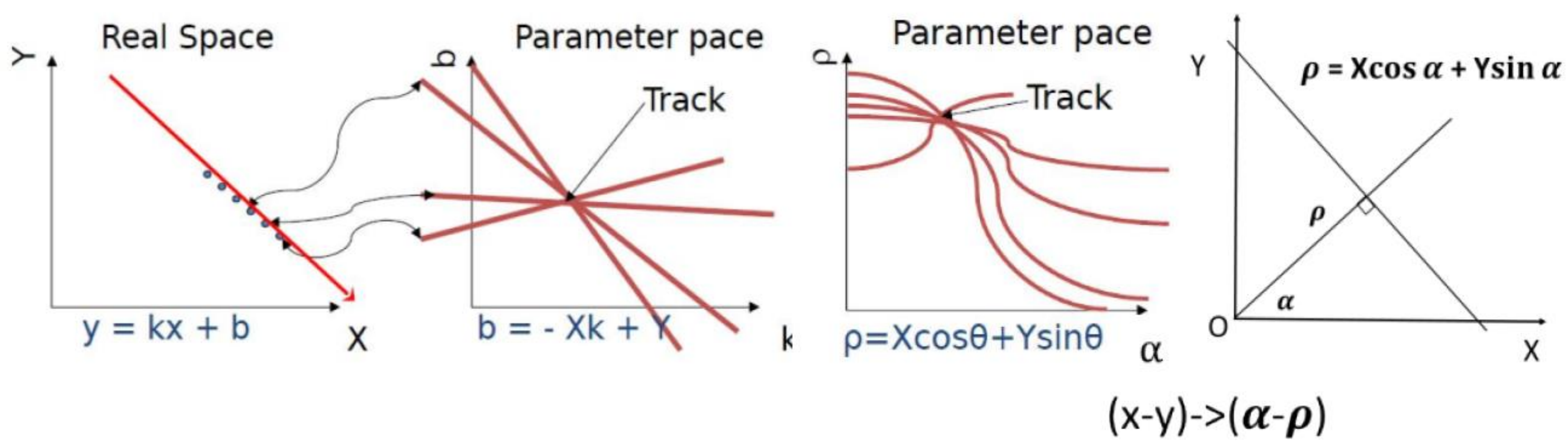


- A circle can be estimated by every three datapoints
- Each circle is transformed to the Legendre space
- The maxima in Legendre space are used for extraction of the original circle parameters



Hough变换

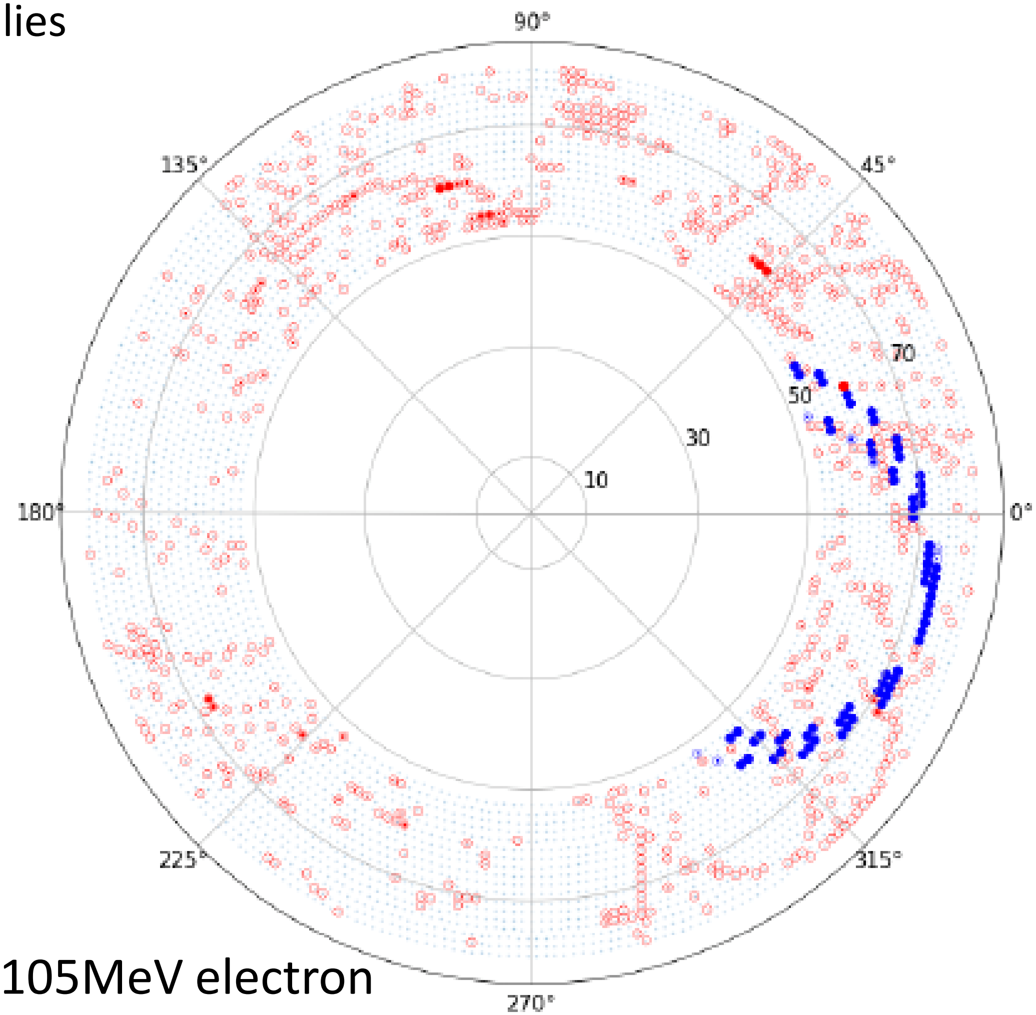
- R->0的Legendre变换
- Transform a point in real space to a line or a curve in parameter space
- Points rest on a line in real space ←→lines or curves in Hough space



Example: circular Hough transform in Comet experiment

From Ewen Lawson Gillies

- Open circles are original hit locations
- Signal Hits and Background Hits are scaled to the output of the neighbour level GBDT.



Detector watches for 105MeV electron

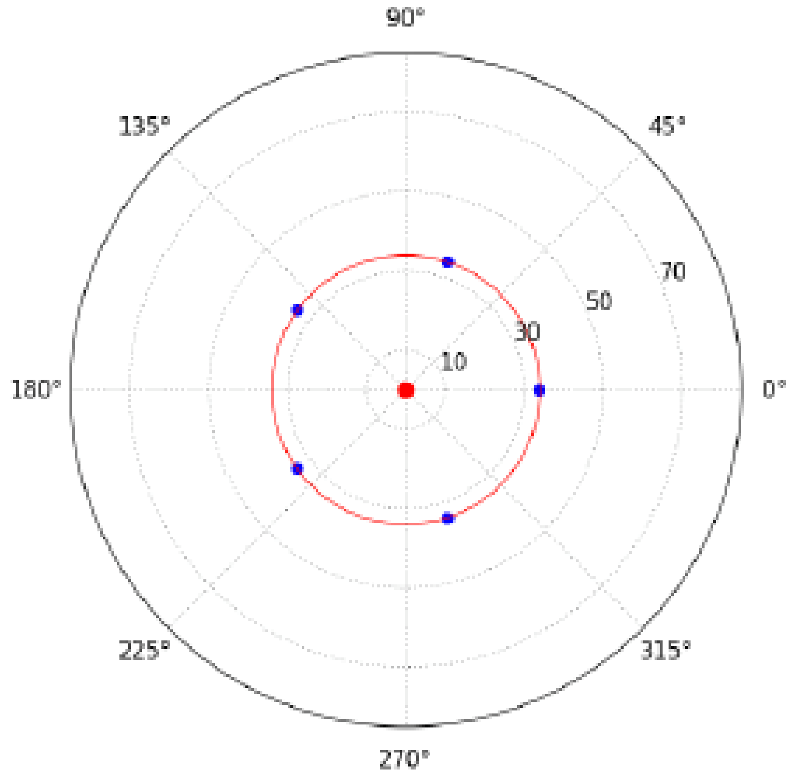


Figure 1: Points in (x, y) space, blue, thought to be on a circle, red, whose centre lies at the origin, orange.

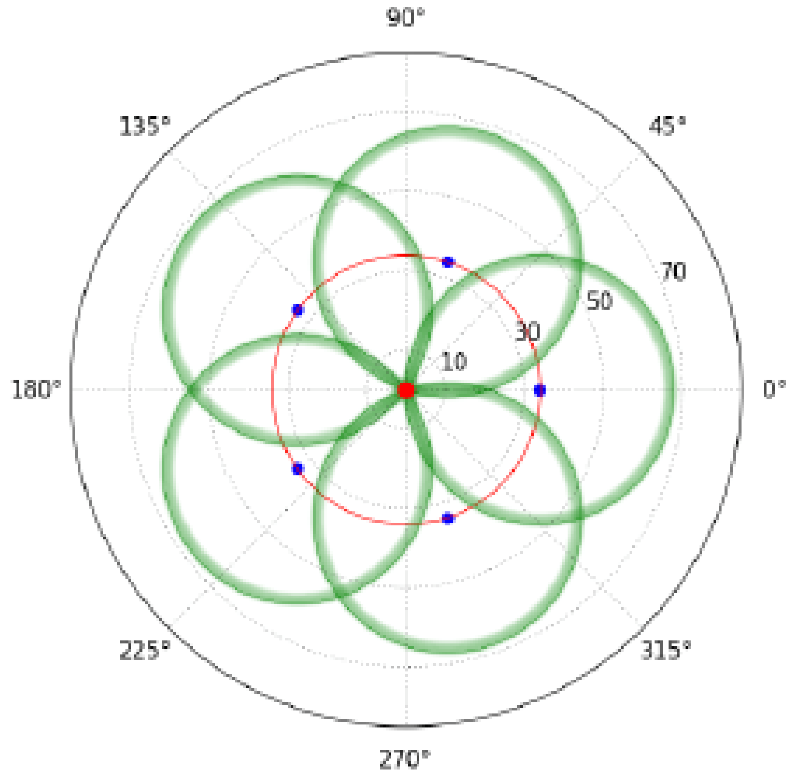
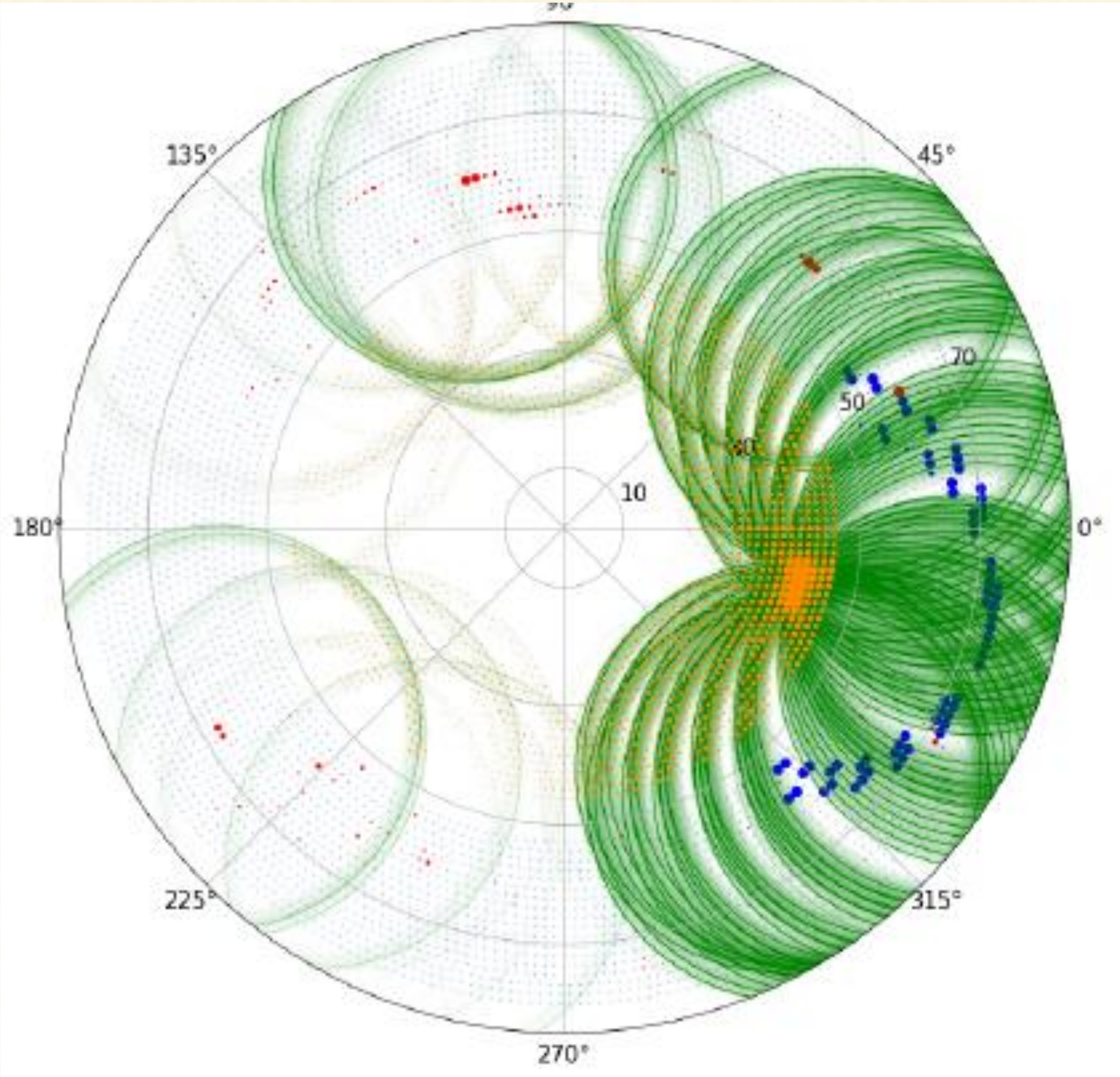


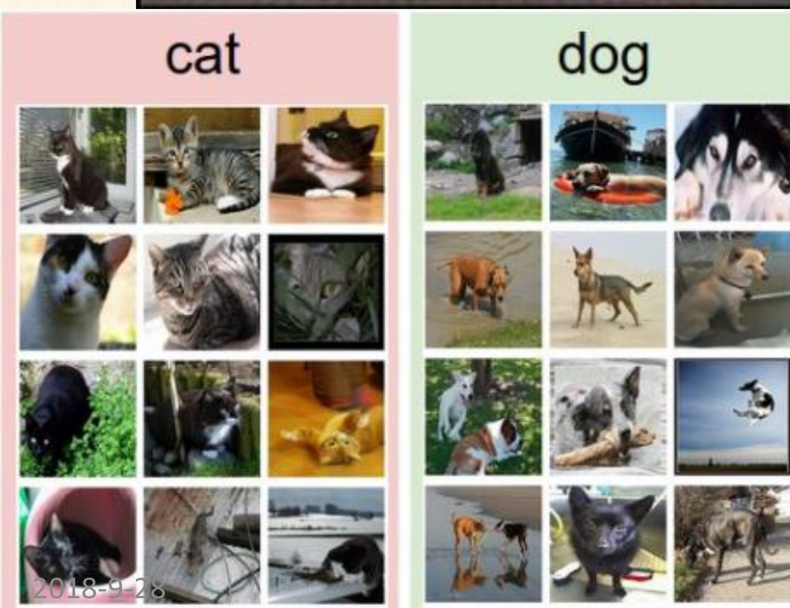
Figure 2: A mapping from the points in (x, y) space, blue, to possible circle centers in (a, b) space, green.

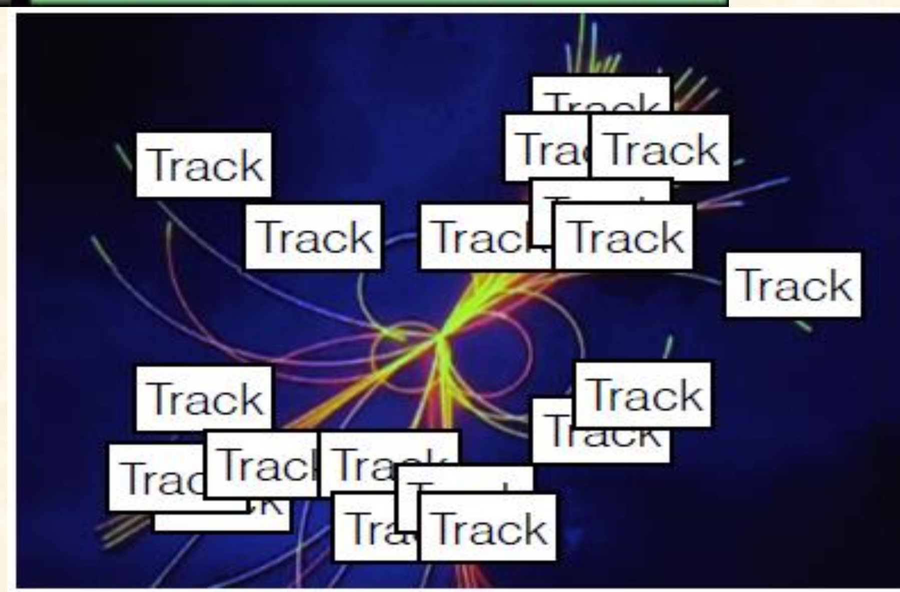


135°
45°
70
50
40
10
180°
0°
225°
315°
270°

寻迹方法 -- 机器学习


点击查看源网页
sky
boat
tree
building
person
fence
mbike
water
ground


cat
dog


Track
Track
Track
Track
Track
Track
Track
Track
Track

机器学习在重建中的应用

- 排除噪声
  - BESIII上正在研究
- CGEM探测器中的Cluster重建
  - Liu Beijiang’ talk in CHEP2018
- 结合Hough变换进行寻迹
  - 如Comet实验中研究的GBDT方法

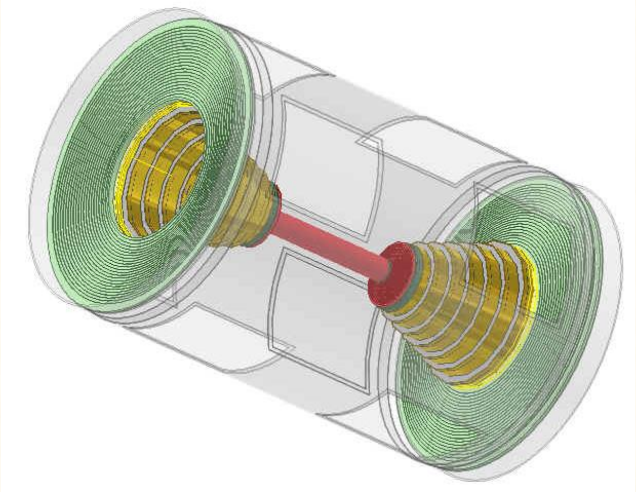
径迹拟合

- 全局最小二乘拟合
  - 将径迹视为标准螺旋线，快速地得到径迹参数
- 分段拟合
  - 便于考虑多次散射、磁场不均匀、能量损失等效应，如Kalman filter方法

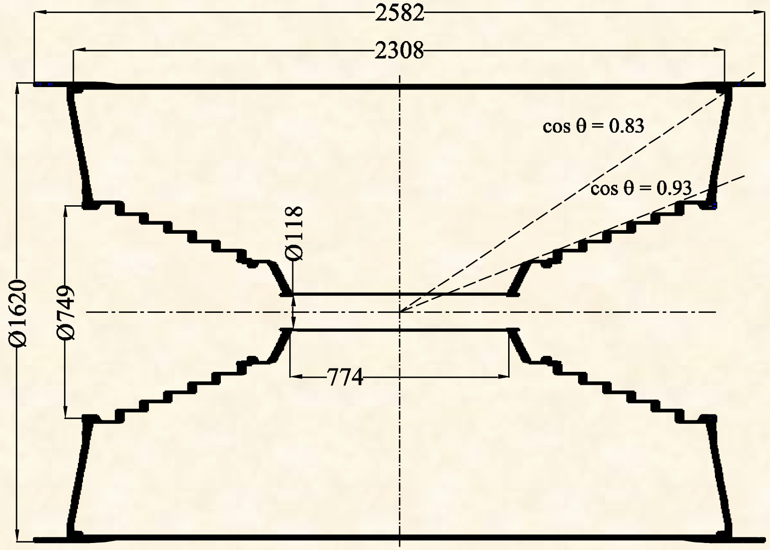
BESIII实验中的径迹重建

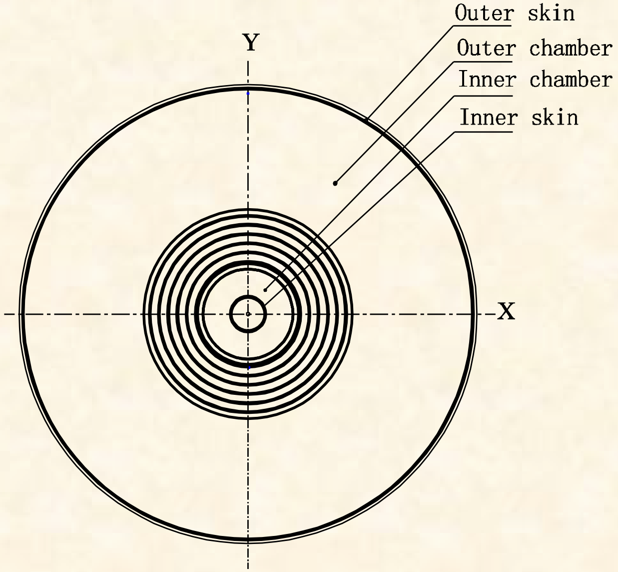
- 采用漂移室进行带电粒子的径迹探测
- 设计要求
  - 具有尽可能大的立体角覆盖，并为加速器超到插入铁提供空间
  - 高的径迹探测和重建效率
  - 好的动量分辨 （ 0.5%@1GeV/c ）
  - 好的dE/dx分辨（ 6 ~ 7% ）
  - 能在1T磁场下正常工作
  - 能在高计数率条件下正常工作，并能有较长的寿命

BESIII漂移室总体结构

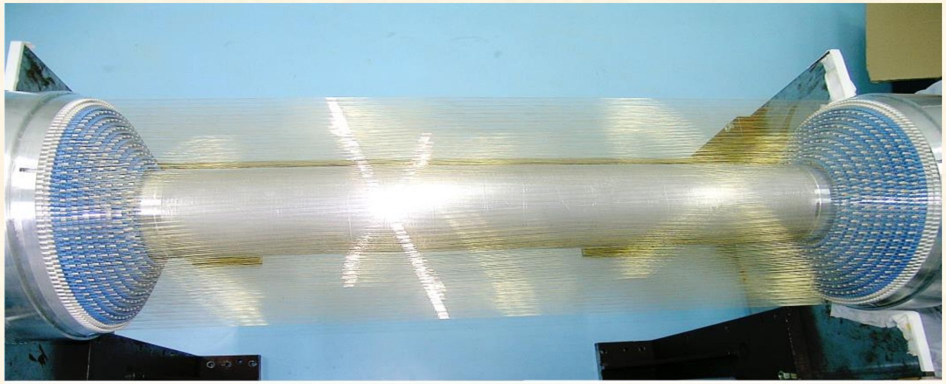


- 2.6m long cylindrical chamber
- It consists of an inner chamber and an outer chamber
- With an inner radius of 59mm and an outer radius of 810mm
- Filled with He/C_3H_8(60/40)





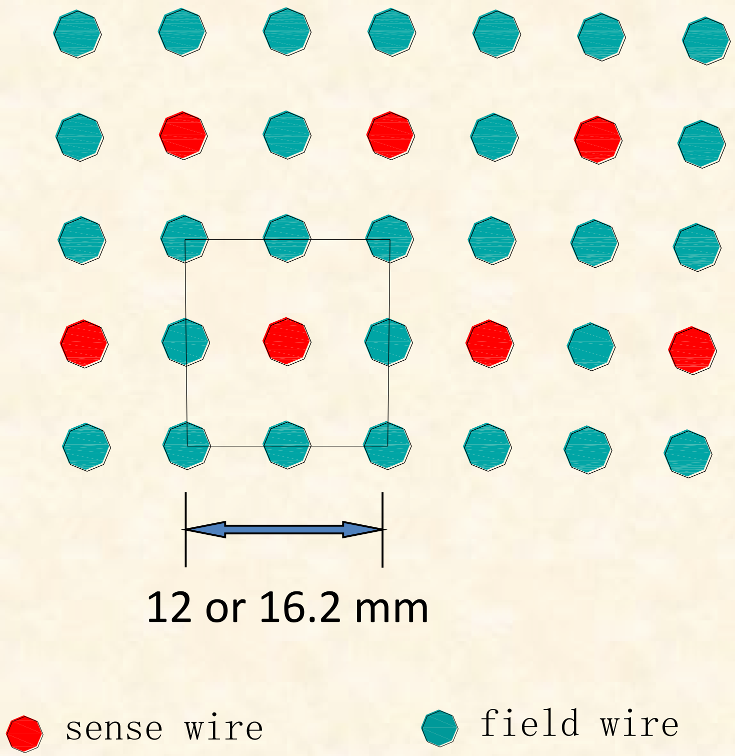
内室



- Sense wires:
  - ϕ25μm gold-plated tungsten
  - at a potential of 2100V for the inner chamber and 2200V for the outer chamber
- Field wires:
  - ϕ110μm gold-plated aluminum
  - at ground

外室

单元结构



- Square cell
- 6796 sense wires and 21884 field wires.
- The average half-cell size is 6mm for the inner chamber and 8.1mm for the outer chamber
- Half cell staggering to resolve the left-right ambiguity

丝层设置

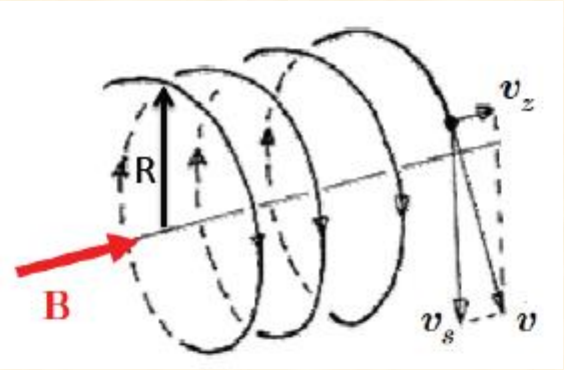
- 43 sense layers

  The inner most 8 layers are located in the inner chamber and the other 35 layer are in the outer chamber.

Layer	Number of cells per layer	type	Stereo angle（degree）
1 — 4	40/44/48/56	stereo（–）	2.9 ~ 3.3
5 — 8	64/72/80/80	stereo（+）	3.4 ~ 3.9
9 — 20	76/88/100/112/128/140	axial	—
21 — 24	160	stereo（–）	2.4 ~ 2.7
25 — 28	176	stereo（+）	2.7 ~ 3.1
29 — 32	208	stereo（–）	3.0 ~ 3.3
33 — 36	240	stereo（+）	3.3 ~ 3.6
37 — 40	256	axial	—
41 — 43	288	axial	—

动量测量

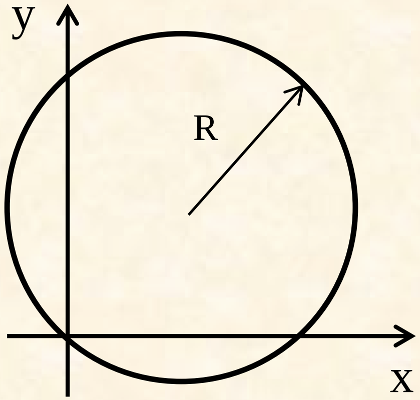
入射带电粒子在均匀磁场B下，
其飞行轨迹为螺旋线



在x-y(或r-ϕ)平面上测量粒子的横动量(P_T)

$$R = \frac{P_T}{0.3qB}$$

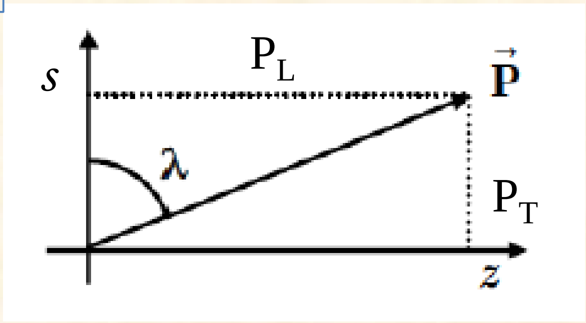
R: m
P_T: GeV/c
B: T
q: 粒子电荷，以电子的电荷为单位



在s-z平面上测量dip angle (λ)

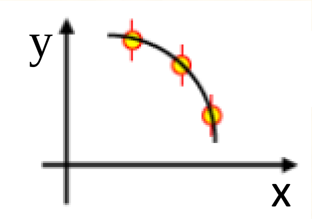
s为粒子飞行长度

$P = P_T/\cos\lambda$



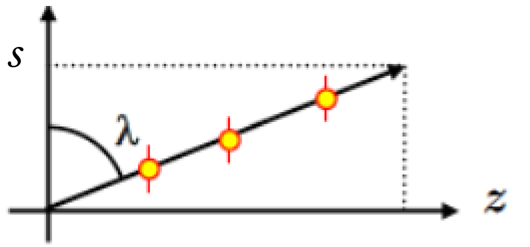
位置测量

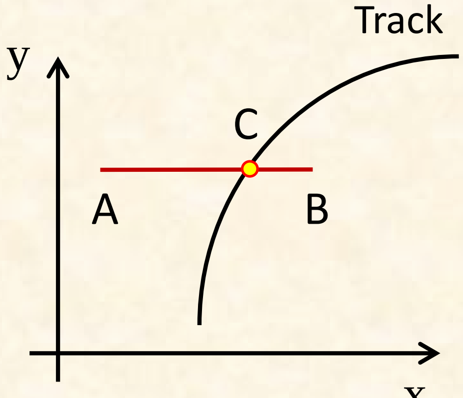
直丝 (axial wire)： x-y平面的测量

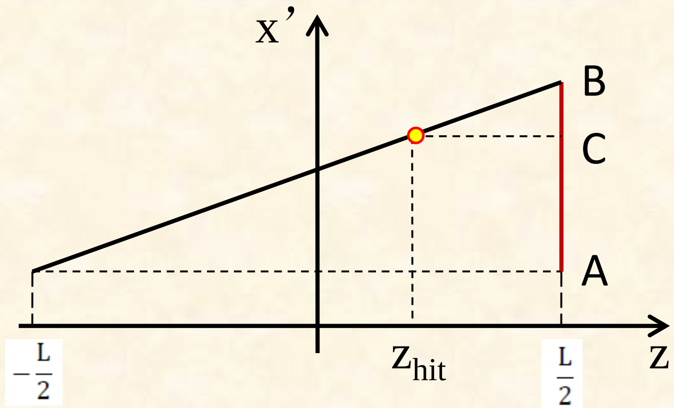


斜丝 (stereo wire)： x-y平面及z向测量

丝与z轴的夹角称为倾斜角(ε)







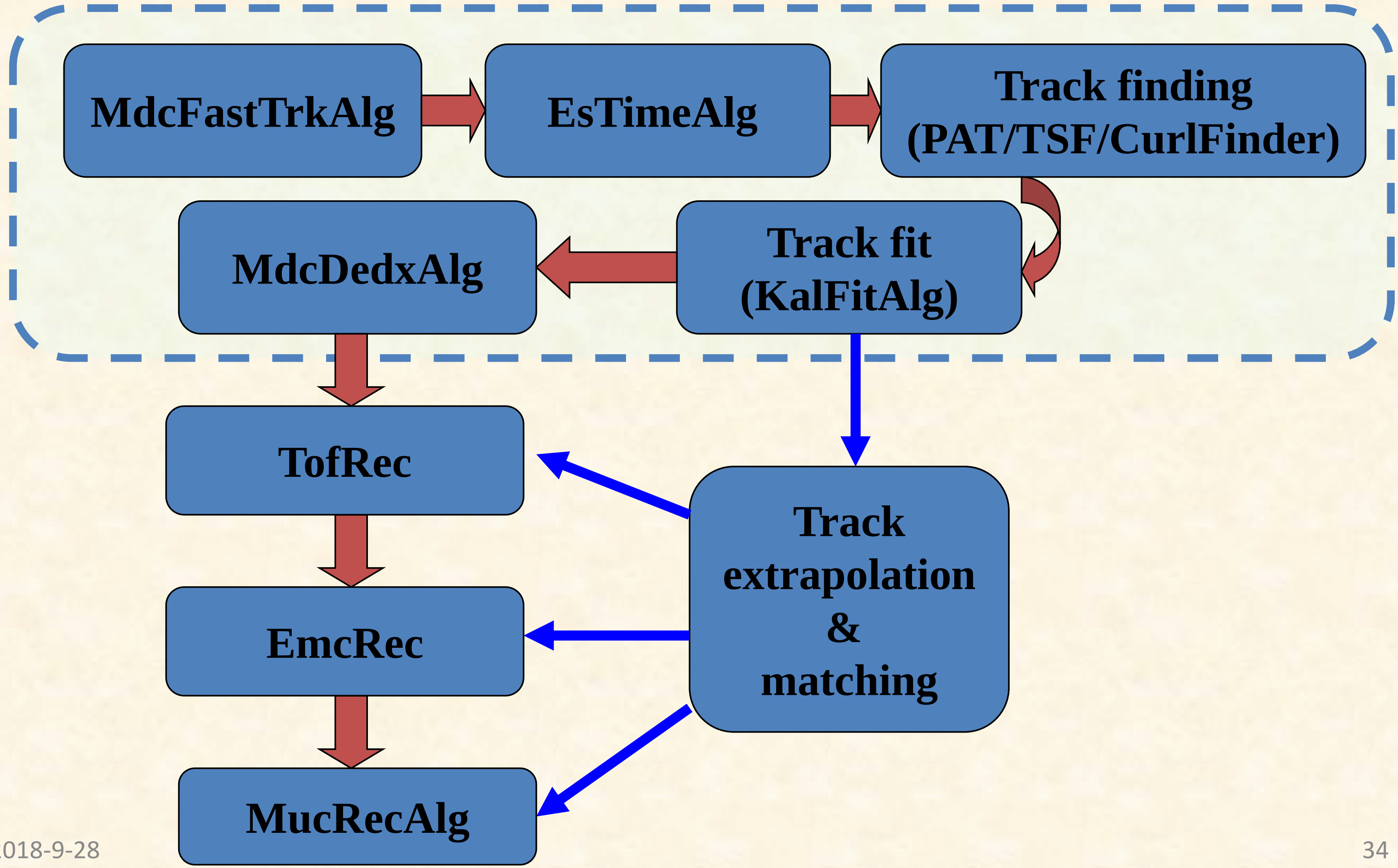
$$z_{hit} = \frac{L_{AC}}{\tan\varepsilon} - \frac{L}{2}$$

ε较小 $\sigma(L_{AC}) \approx \sigma x$

$$\sigma z \approx \sigma x / \tan\varepsilon$$

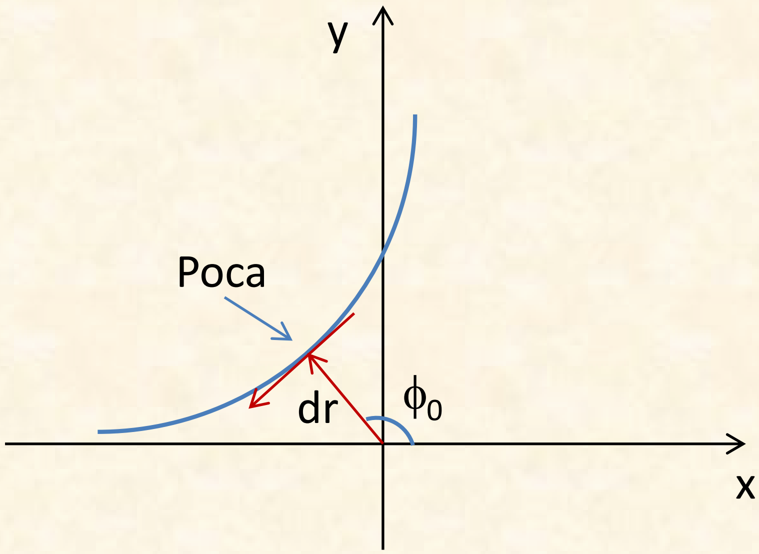
如果σx=120μm, ε=50mrad → σz=2.4mm

重建流程


MdcFastTrkAlg
EsTimeAlg
Track finding
(PAT/TSF/CurlFinder)
MdcDedxAlg
Track fit
(KalFitAlg)
TofRec
Track
extrapolation
&
matching
EmcRec
MucRecAlg

径迹参数

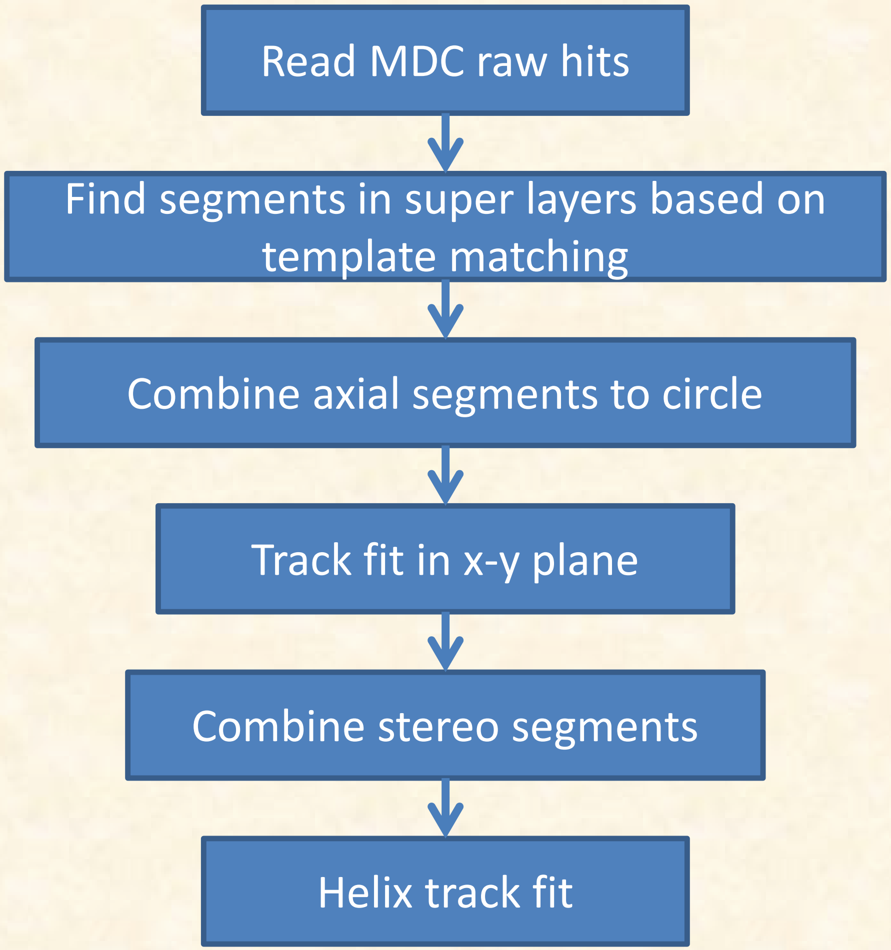
- dr
  - Signed distance in x-y plane from the pivot to the helix
- ϕ_0
  - Azimuth angle of the Poca (point of closest approach in the x-y plane)
- κ
  - q/Pt
- dz
  - z position of the Poca
- $\tan\lambda$
  - Tangent of the dip angle



径迹重建算法

- 径迹寻找
  - PAT algorithm： based on template matching
  - TSF algorithm: conformal transformation
  - CurlFinder algorithm: low momentum tracking
  - Hough transform: a complementary tracking algorithm
- 径迹拟合
  - Global least-squares fit
  - Track fit with Kalman filter

PAT algorithm

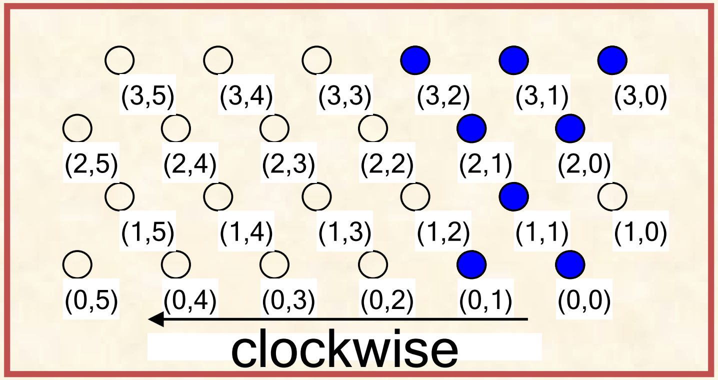


Segment Finding Method

- Tracking in the MDC is based on ***segments***
- Search for ***patterns*** in the super layer

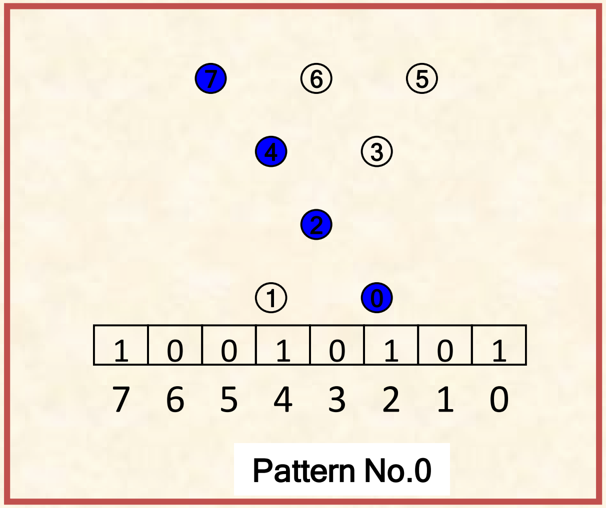
An example of the pattern

Cells for pattern

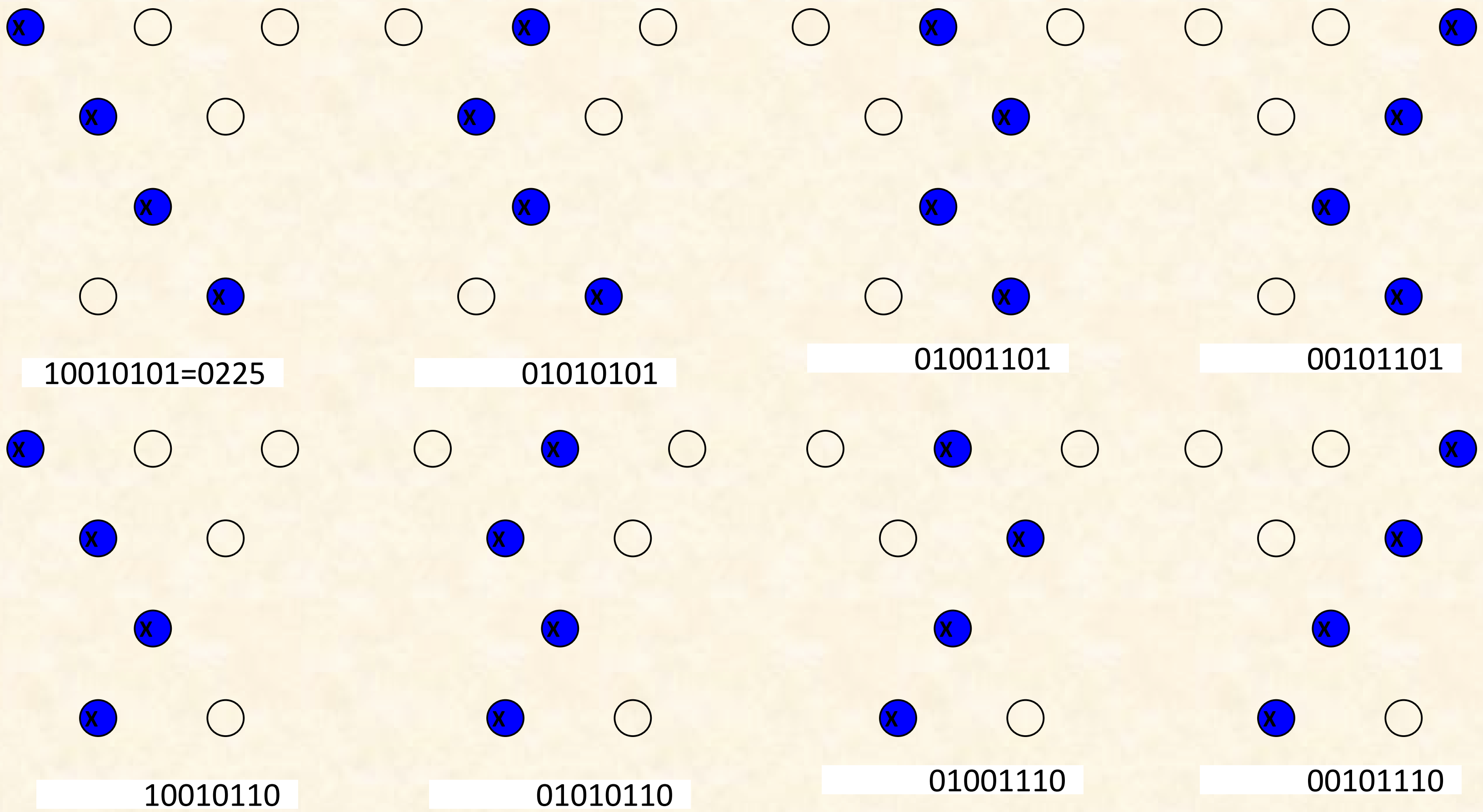


We have 8 4-hit patterns
and 20 3-hit patterns

One segment pattern



4-hit patterns



Combination of segments

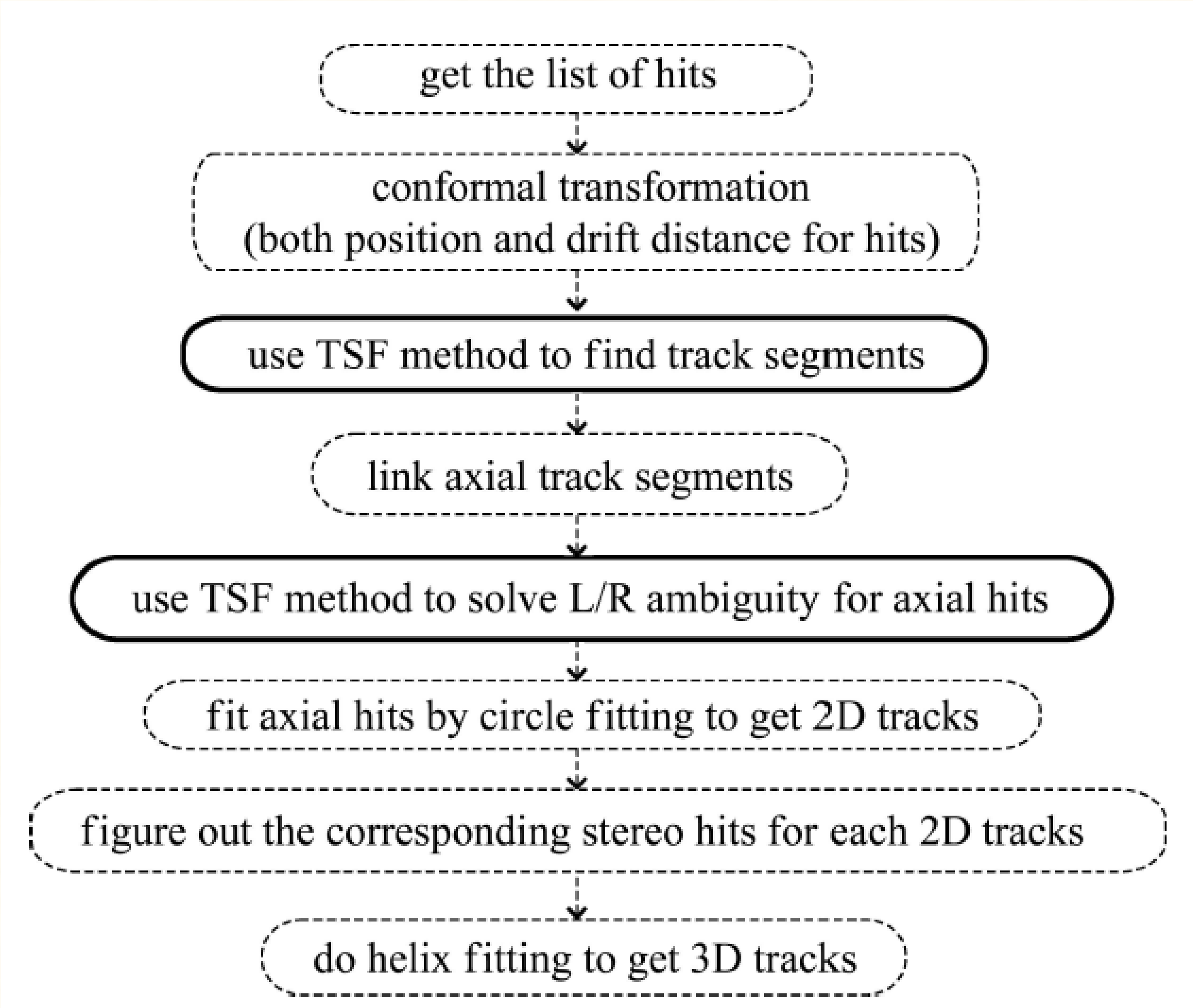
- Link axial segments

$$\begin{aligned} \phi_0 &= \bar{\phi} - arcsin(\kappa r_{sl}) \\ \kappa &= \frac{\bar{\kappa}}{\sqrt{1 + (r_{sl}\bar{\kappa})^2}} \end{aligned}$$

- Link stereo segments

$$\begin{aligned} cot\theta &= (\kappa_{ax} - \kappa_{seg})(-\sqrt{1 - \kappa^2 R^2})(\frac{2L_w}{\alpha}) \\ &= (\frac{\kappa_{sl}}{\sqrt{1 - \kappa_{sl}^2 r_{sl}^2}} - \kappa_{seg})(-\sqrt{1 - \kappa^2 r_s l^2})(\frac{2L_w}{\alpha}) \\ z_0 &= z - cot\theta l \end{aligned}$$

TSF algorithm



Conformal transform

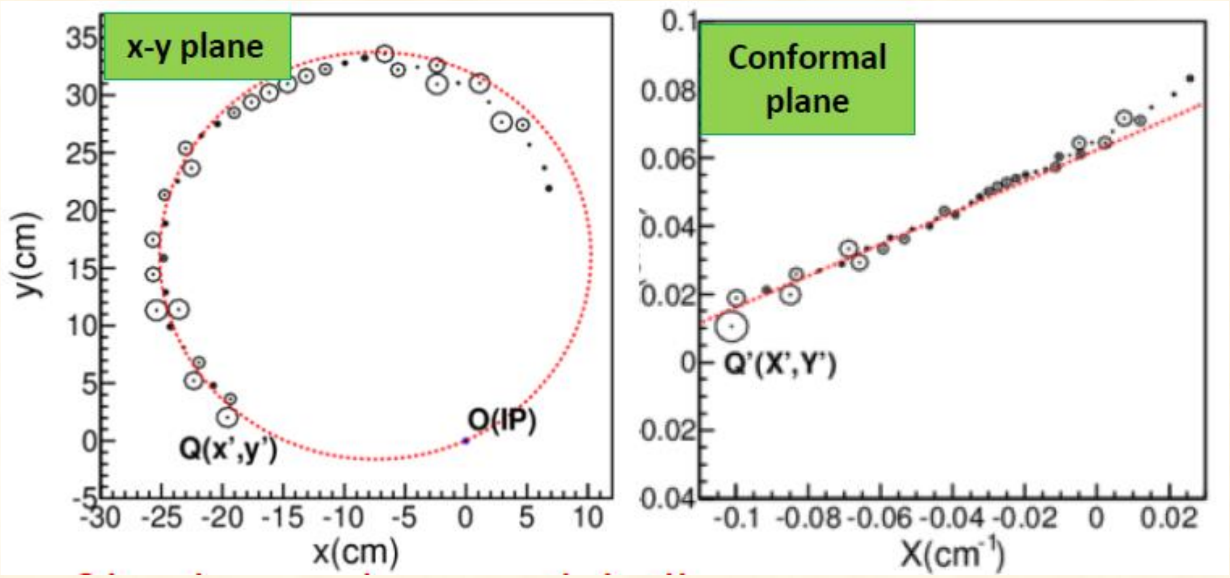
- A hit position (x, y) in x-y plane is transformed into a position (X, Y) in the conformal plane.

  $X = 2x/(x^2+y^2)$, $Y = 2y/(x^2+y^2)$

- A circle which passes through the origin (0, 0) is transformed into a line.

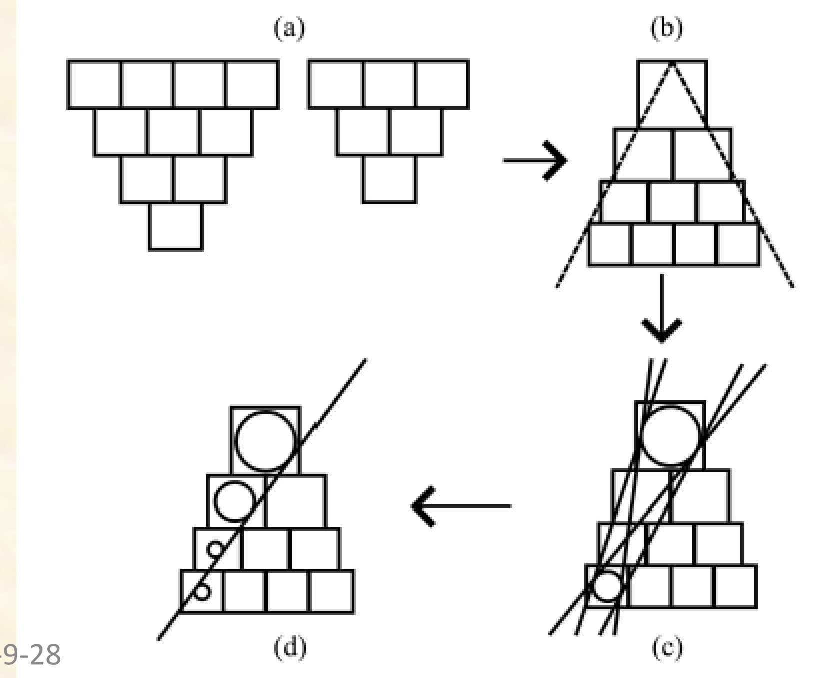
  $x_c X + y_c Y = 1$

  (x_c, y_c) is the center of the circle



Track segment finding

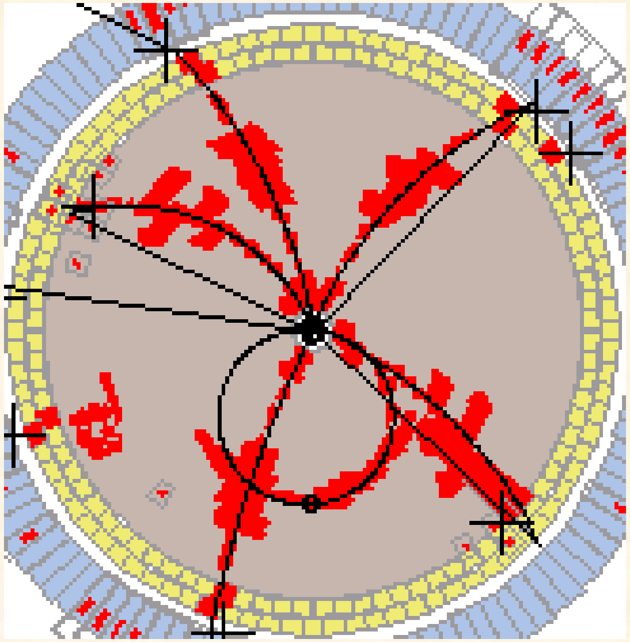
- If the hit belongs to the track, its drift circle in the conformal plane is tangent to the line of the track
- Left-right ambiguity is kept after conformal transform
- Require that at least 3 layer are fired in each super layer



a) Patterns
b) Four-layer pattern in the conformal plane
c) Four candidates of track segment
d) The best candidate after matching the middle layers

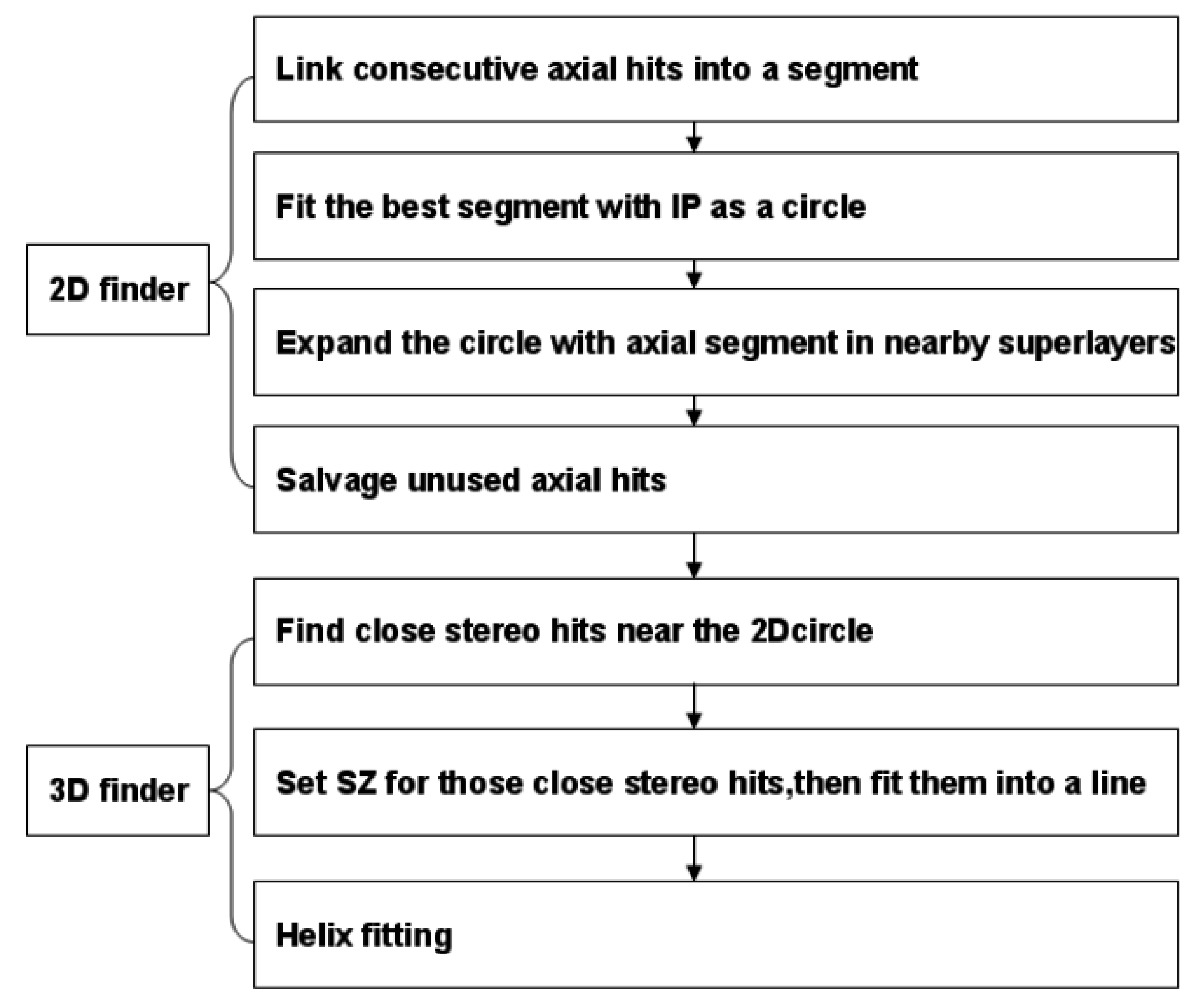
CurlFinder algorithm

- An algorithm for low momentum track finding



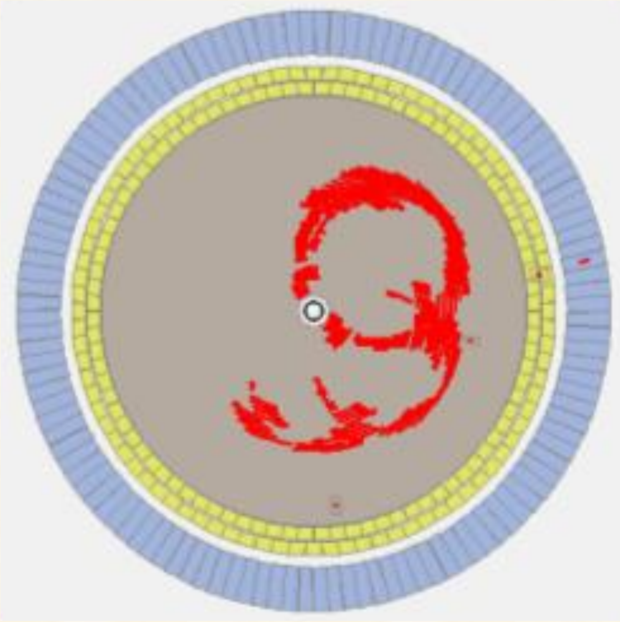
Super layer	Stereo/ axial	Radius (cm)	p_T (MeV/c)
1	stereo	11.5	17
2	stereo	16.2	24
3	axial	24.6	37
4	axial	31.0	47
5	axial	37.5	56
6	stereo	44.8	67
7	stereo	51.4	77
8	stereo	57.9	87
9	stereo	64.2	96
10	axial	71.6	107
11	axial	77.1	116
Outer tube	--	81.0	122

Flow diagram

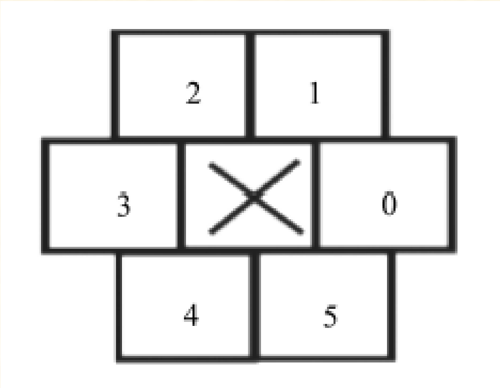


Axial segment finding

- More than one hit in the same layer
- Find consecutive hits in each axial super layer
  - Fired cell → find neighbor → find neighbor's neighbor …

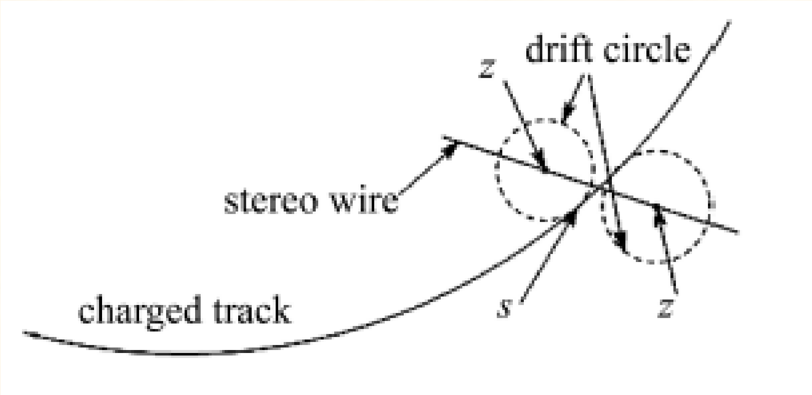


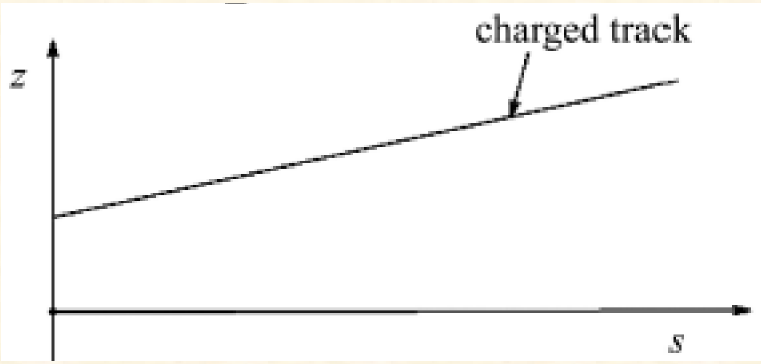
Definition of neighbor

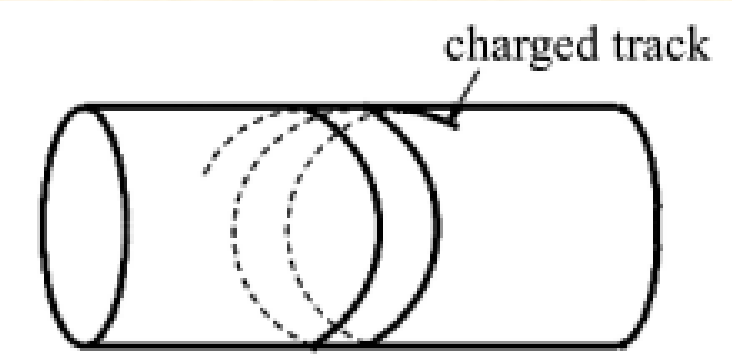


3-D track finding

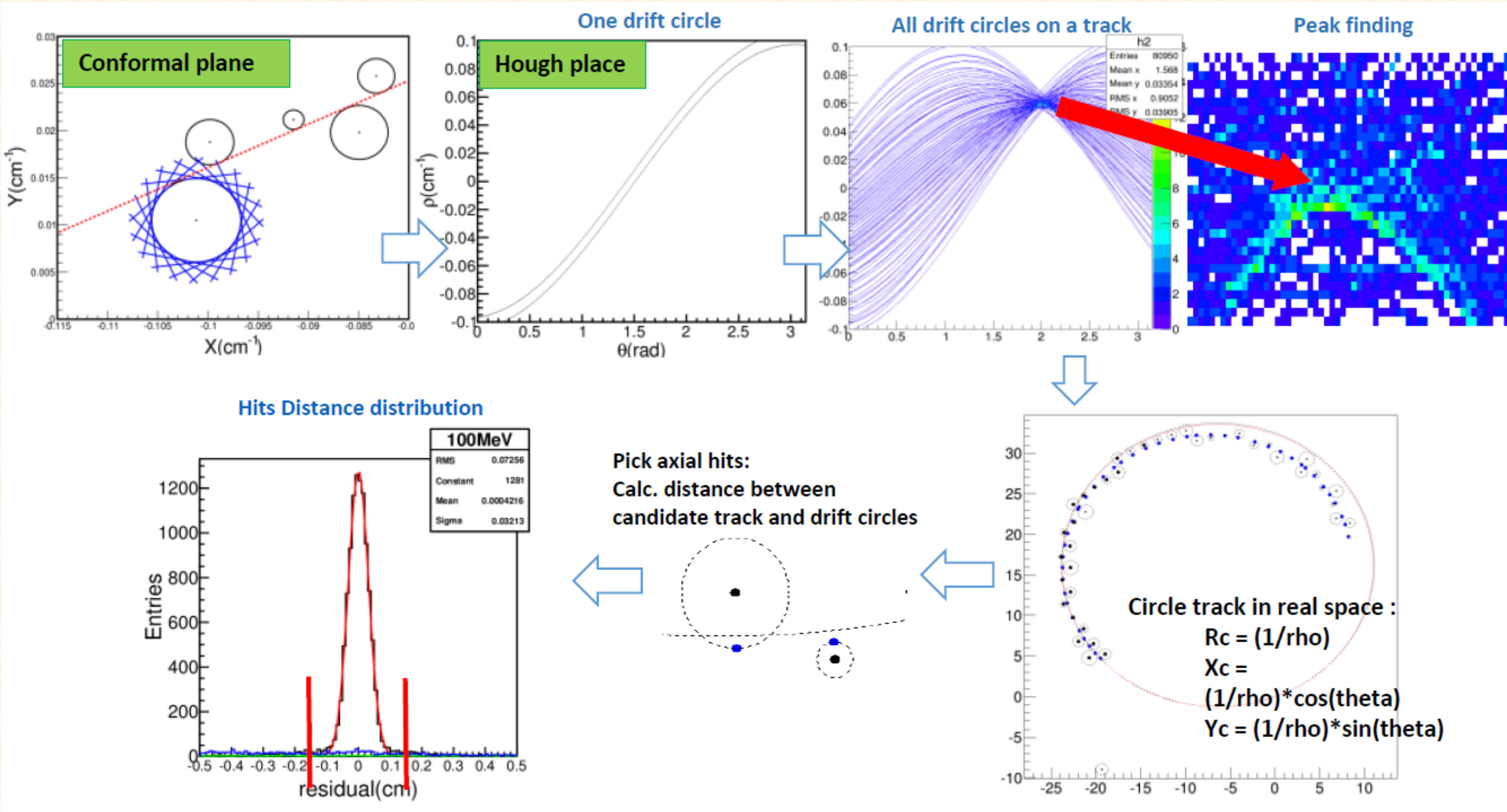
- Find stereo hits near the circle
- Line fit in s-z plane
- Helix fitting







Hough transform



全局的最小二乘拟合

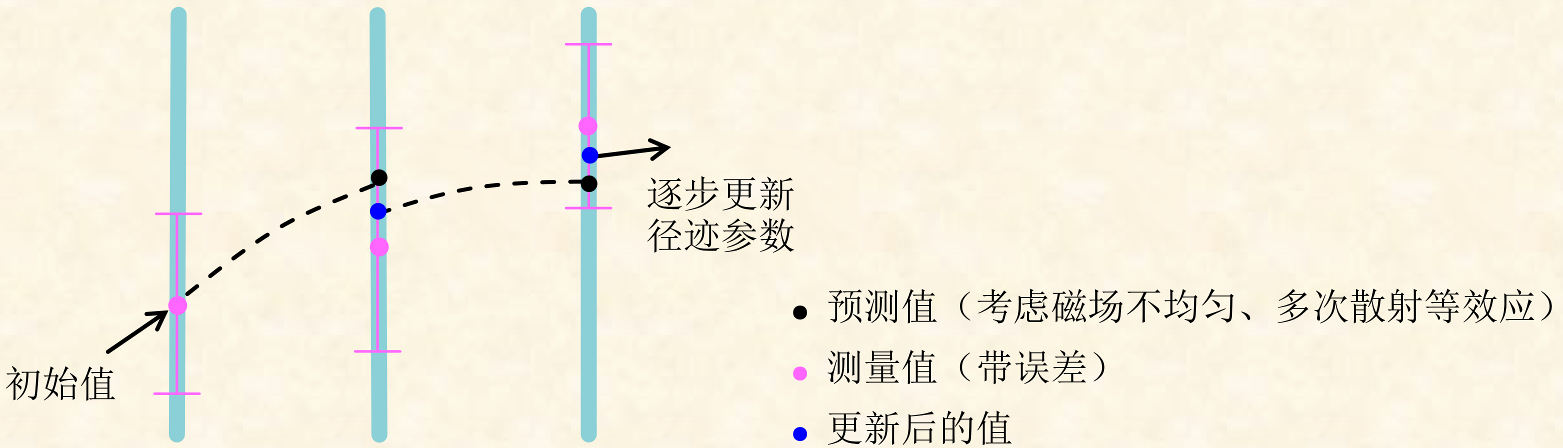
- Least squares fitting without consideration of NUMF, multiple scattering, energy loss

$$\chi^2 = \sum_{i=1}^{N_{hit}} \frac{(d_{meas}^{(i)} - d_{track}^{(i)})^2}{\sigma_i^2}$$

- d_{meas} is the measured drift distance
- d_{track} i.e. doca, distance of closest approach

基于Kalman滤波的径迹拟合

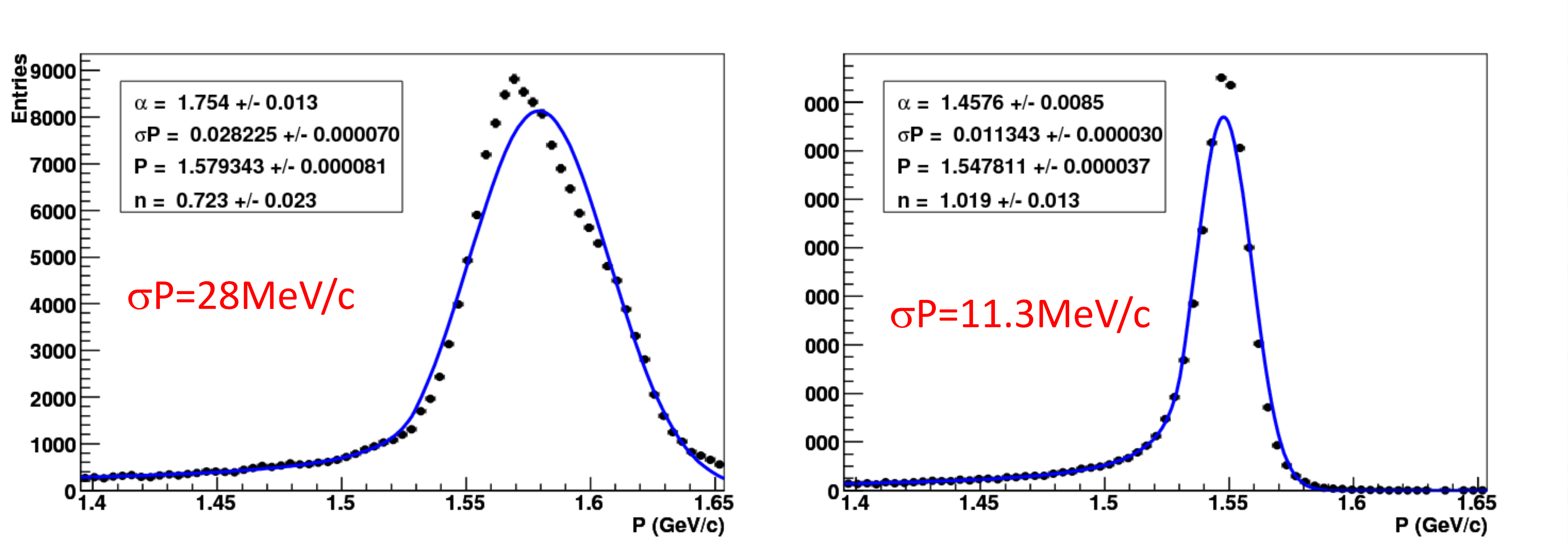
- Track parameters obtained by in iterations: …→prediction→filter with a hit→update→…
- Considered effects: nonuniform magnetic field, energy loss and multiple scattering
- 5 hypothesis: e, μ, π, K, p
- Inwards filter: provide track parameters at IP
- Outwards filter: provide track parameters seed for extrapolation (expected TOF, reconstruction with TOF/EMC/MUC)



两种径迹拟合的对比

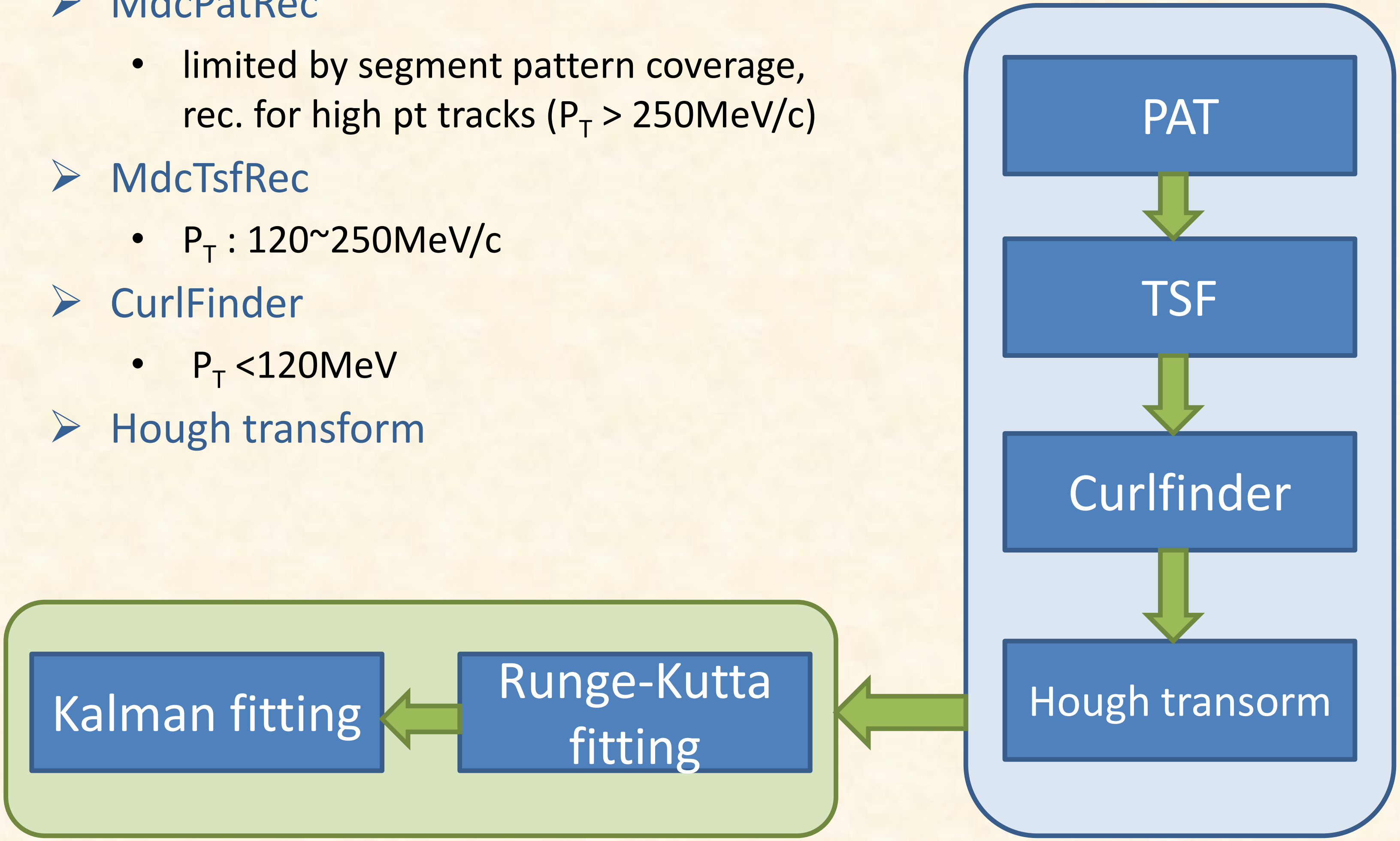
全局拟合结果

Kalman滤波拟合结果



BESIII Drift Chamber Tracking

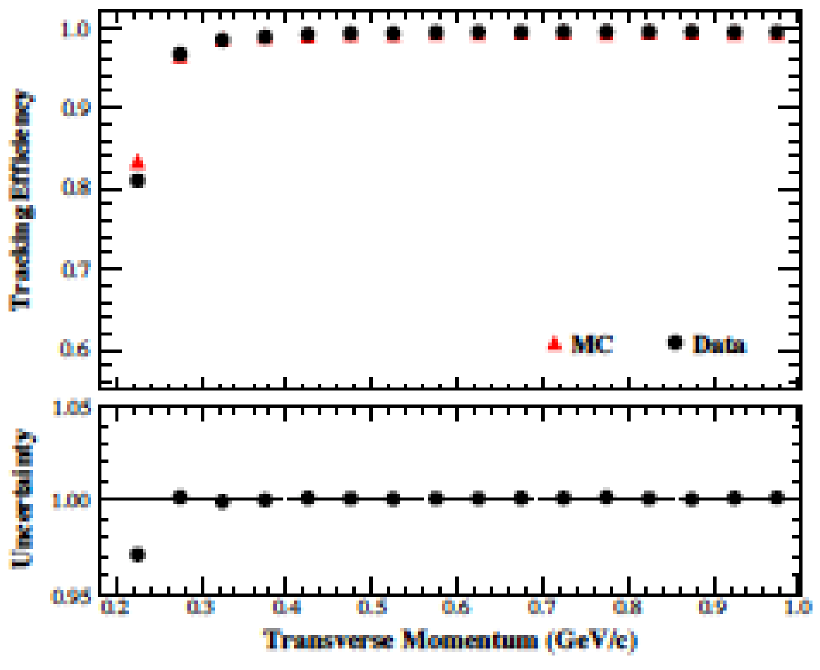
- MdcPatRec
  - limited by segment pattern coverage, rec. for high pt tracks (P_T > 250MeV/c)
- MdcTsfRec
  - P_T : 120~250MeV/c
- CurlFinder
  - P_T <120MeV
- Hough transform

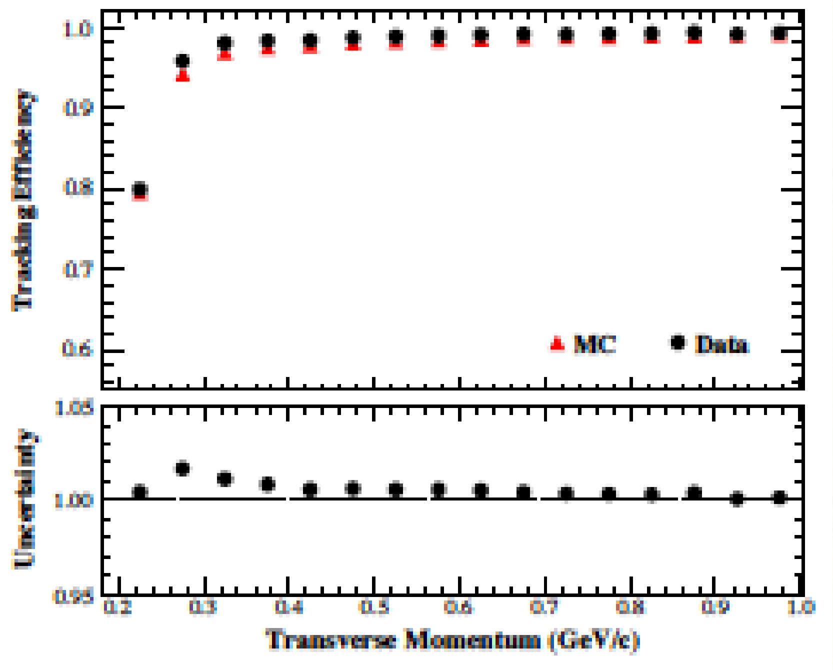


Tracking efficiency with P_T

J/psi → ppππ， without Hough transform

proton anti-proton

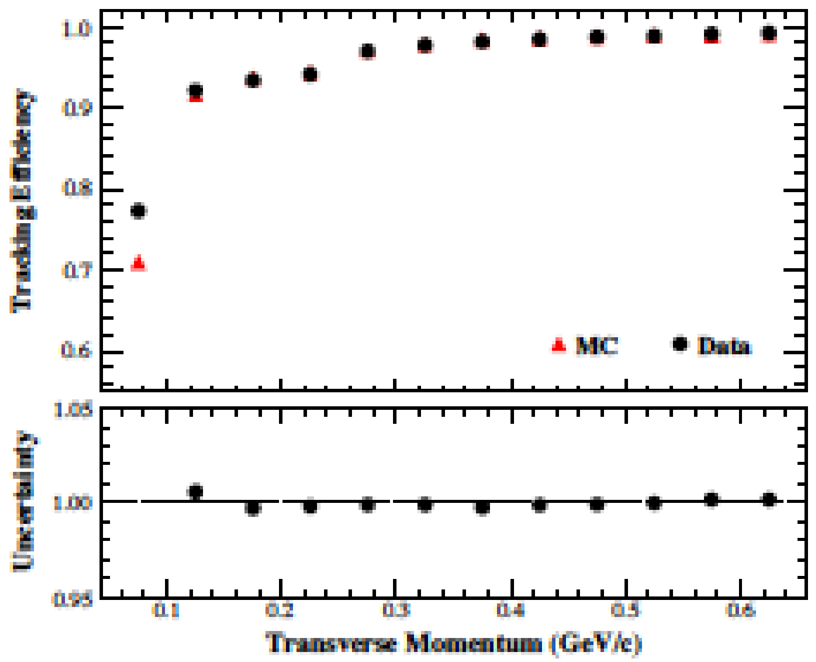


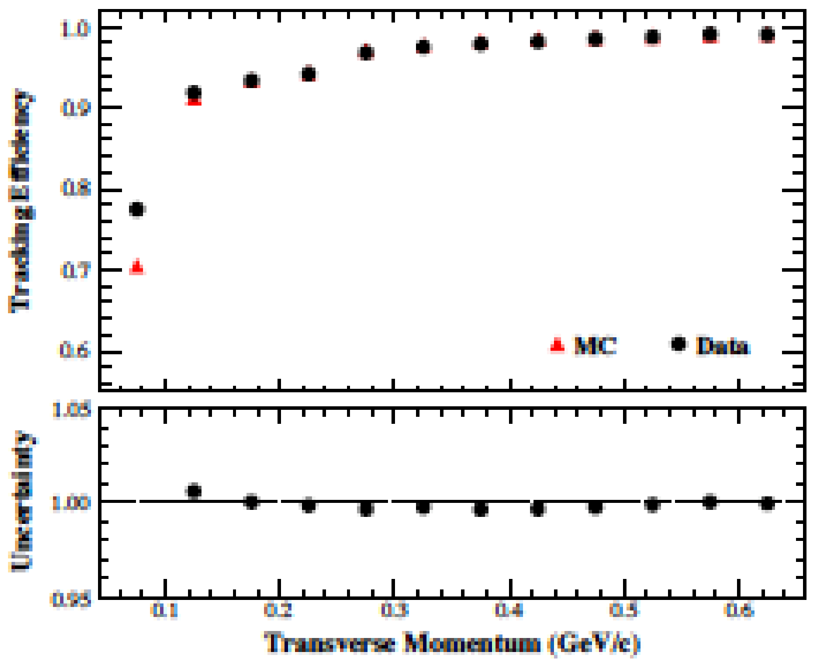


Tracking efficiency with P_T

J/psi → ppππ， without Hough transform

π+ π-





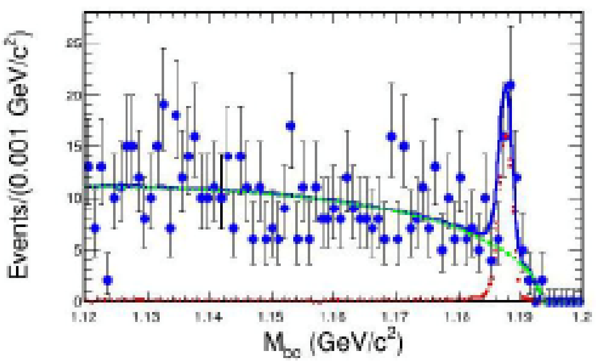
Tracking efficiency after Hough tracking

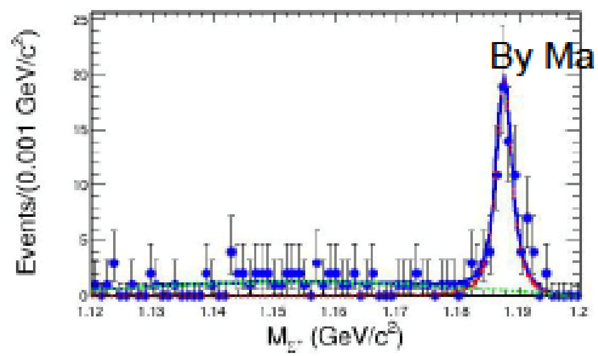
Ds Signal Yields slightly improved after using HOUGH tracking

By H.L. Ma and S.F. Zhang

From 2.386 data, number of signal increased ~5% after using HOUGH

By Ma X.N and Zhou X.R.





Monte-Carlo Λ_c from 4.6GeV

	Efficiency(%)	relative eff. increase after HOUGH
$\Lambda_c^+ \to \Sigma^+\omega$ (mBC 2.282~2.292)	9.491	2.89
$\Lambda_c^+ \to \Sigma^+\Phi$ (mBC 2.275~)	2.571	1.98
$\Lambda_c^+ \to p^+\Phi\pi^0$ (mBC 2.275)	6.213	3.30
$\Lambda_c^+ \to p^+K^-e^+\nu_e$ (mBC 2.275)	24.808	2.5

总结

- 带电粒子径迹重建是高能物理实验数据处理的重要环节
- 寻迹方法
  - 模版匹配、数学变换、机器学习
- BESIII实验中的径迹重建
  - 径迹寻找：多种方法相结合
  - 径迹拟合：Kalman滤波方法

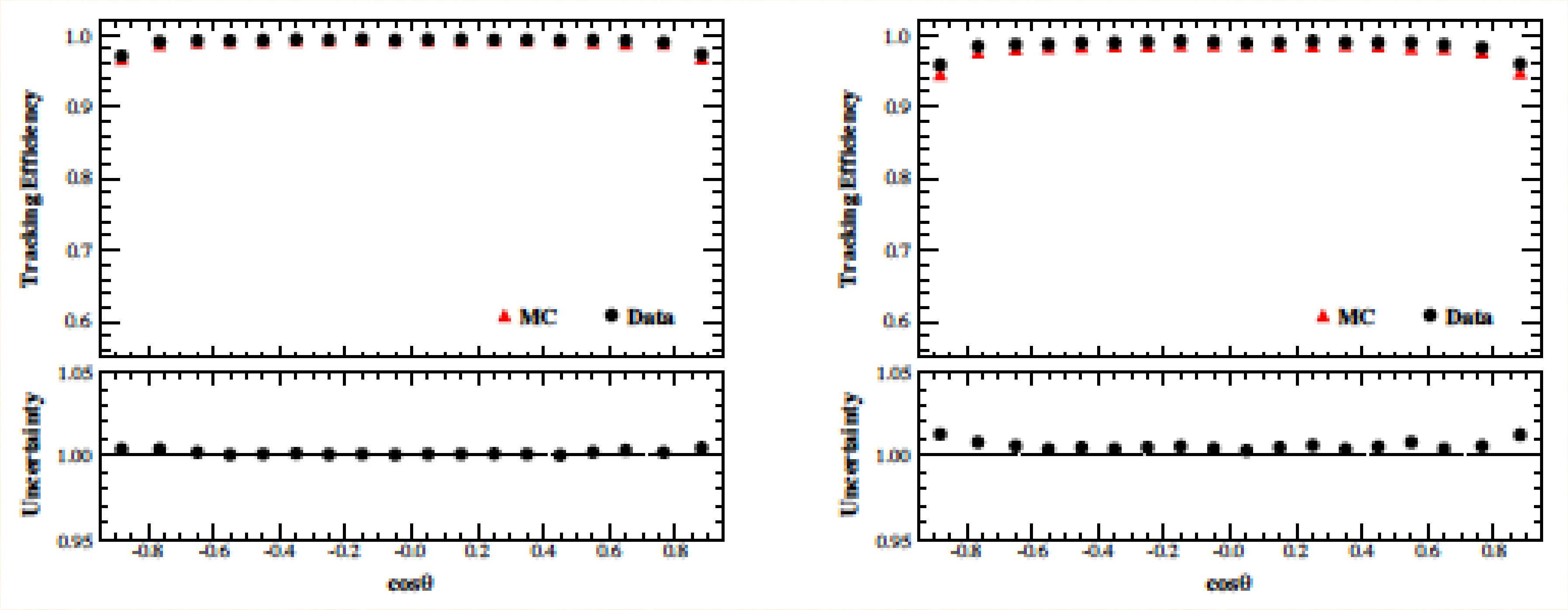
谢谢！

Backup

Tracking efficiency with $\cos\theta$

J/psi → $pp\pi\pi$， without Hough transform

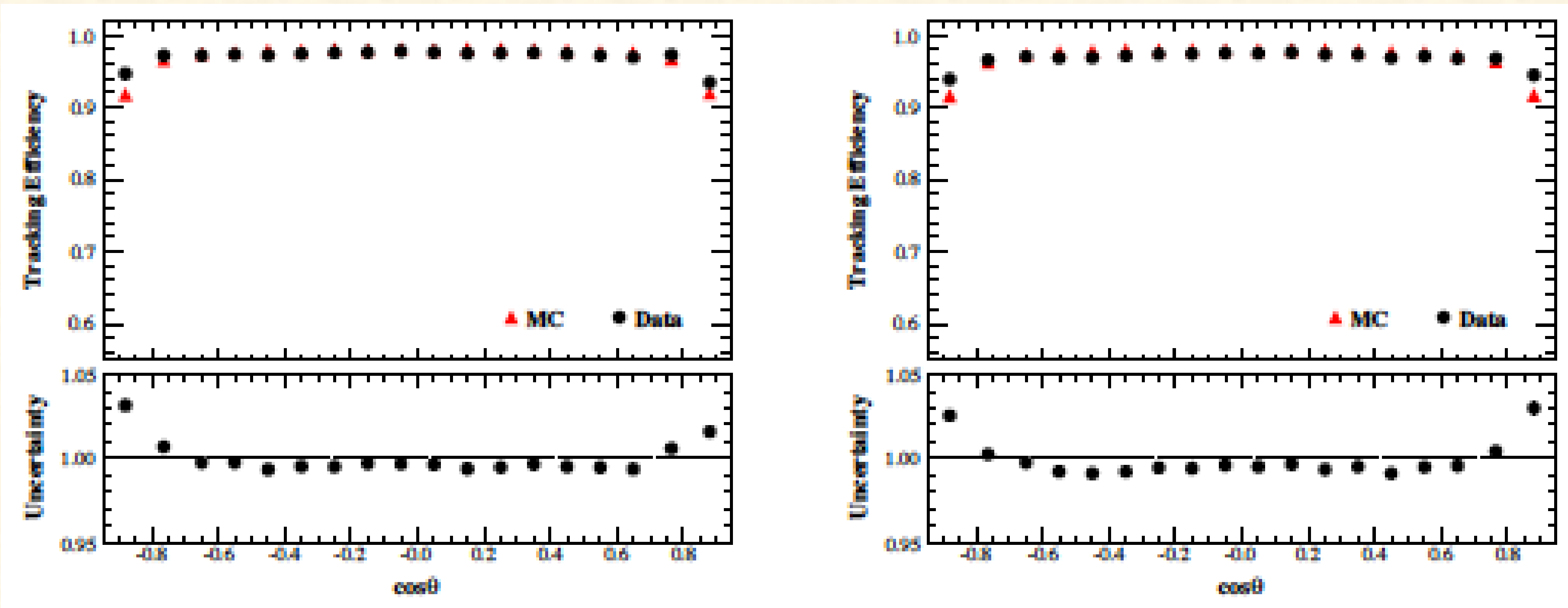
proton　　　　　　　　anti-proton



Tracking efficiency with cosθ

J/psi → pp$\pi\pi$， without Hough transform

π+ π-



Main References Contact

GENFIT

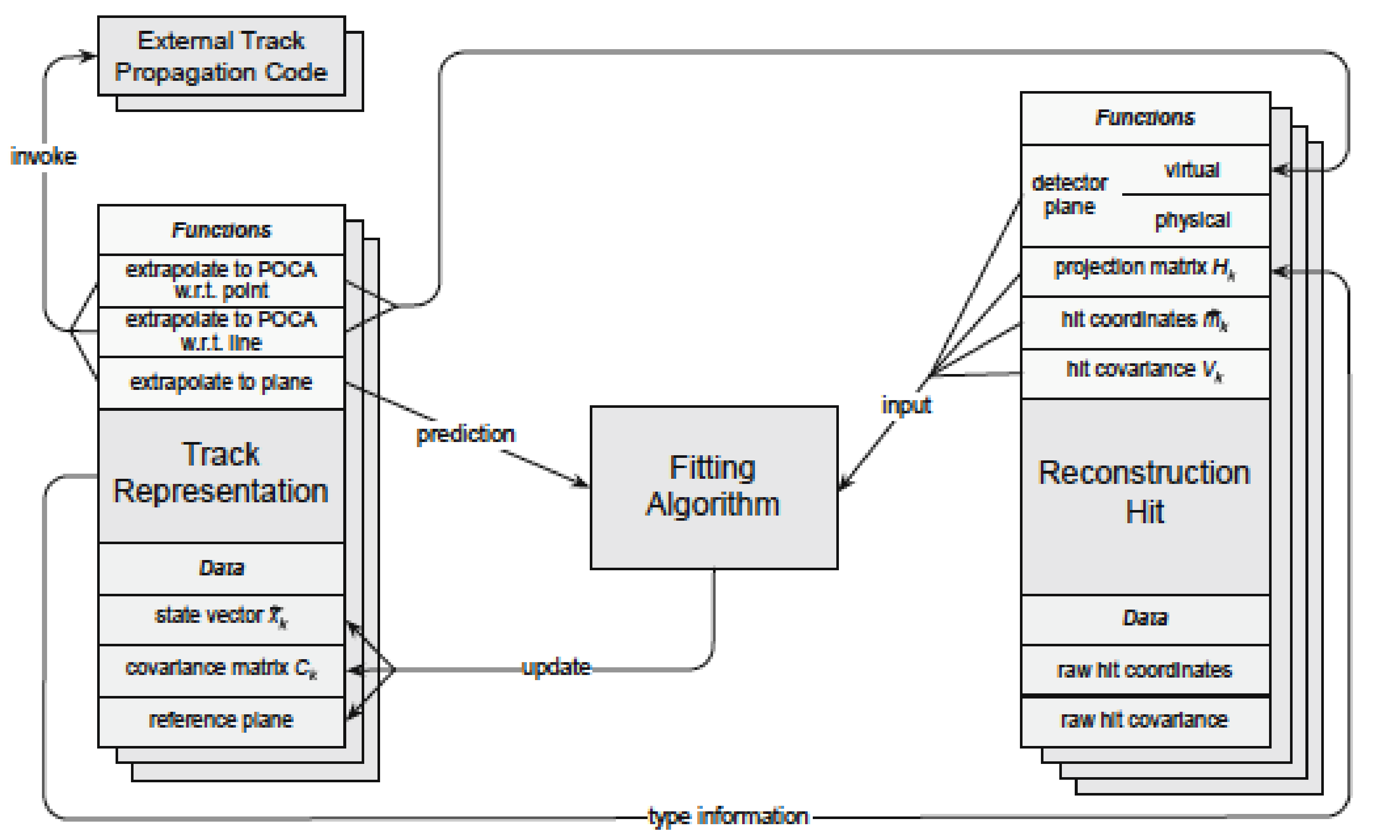
a generic toolkit for track reconstruction for experiments in particle and nuclear physics

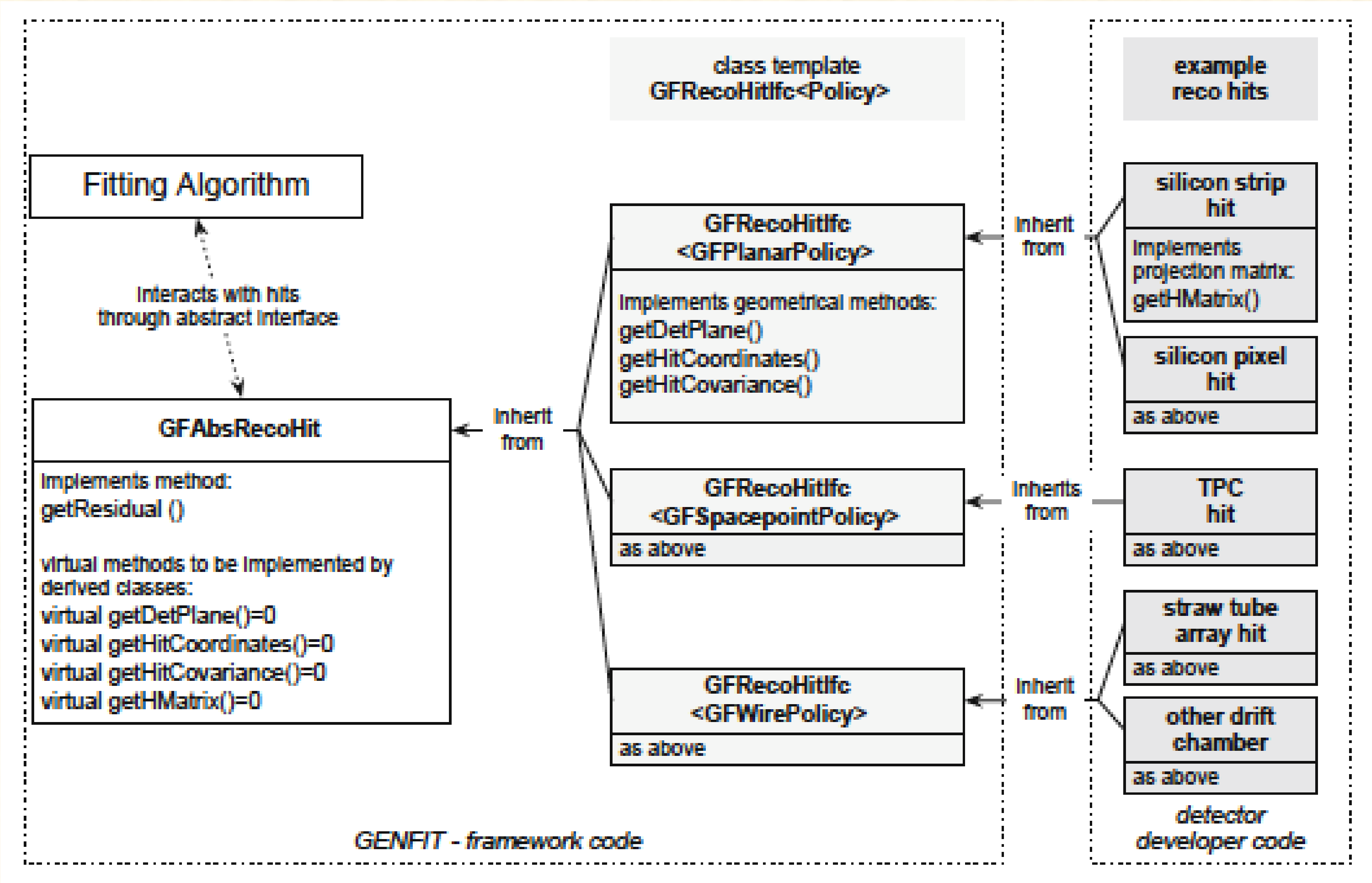
GENFIT is an experiment-independent framework for track reconstruction for particle and nuclear physics. It consists of three modular components:

- **Track fitting algorithms**
  Currently, GENFIT contains a Kalman Filter and a Deterministic Annealing Filter. Other algorithm modules can be added easily.
- **Track representations**
  These modules hold the data of track track parameters and can perform extrapolations of these parameters. GENFIT is distributed with two well-tested track representations.
  Existing track extrapolation codes can be interfaced in a very straightforward way in this framework, using their native geometry and magnetic field interfaces.
- **Reconstruction hits**
  The hit dimensionality and the orientation of planar tracking detectors can be chosen freely. GENFIT is especially useful for tracking systems which include detectors which do not measure the passage of particles on predefined planes, like TPCs or wire-based drift chambers. The concept of so-called virtual detector planes provides a simple mechanism to use these detector hits in a transparent way without any geometrical simplifications.

GENFIT has been developed in the framework of the PANDA experiment at FAIR, Darmstadt, Germany. It is also used in the Belle-2, Fopi, and GEM-TPC experiments.
GENFIT is implemented in C++ and makes extensive use of object-oriented design. It is available here at sourceforge under the LGPL v3. There is a mailing list, which you can register for at https://lists.sourceforge.net/lists/listinfo/genfit-forum. A paper on GENFIT has been published in NIM A (preprint). Recent developments are described in another preprint.

http://genfit.sourceforge.net/Main.html


External Track Propagation Code
invoke
Functions
extrapolate to POCA w.r.t. point
extrapolate to POCA w.r.t. line
extrapolate to plane
Track Representation
Data
state vector $\vec{x}_k$
covariance matrix C_k
reference plane
prediction
Fitting Algorithm
update
input
Functions
detector plane
virtual
physical
projection matrix H_k
hit coordinates $\vec{m}_k$
hit covariance V_k
Reconstruction Hit
Data
raw hit coordinates
raw hit covariance
type information


class template
GFRecoHitIfc<Policy>
example
reco hits
Fitting Algorithm
interacts with hits
through abstract interface
GFRecoHitIfc
<GFPlanarPolicy>
Implements geometrical methods:
getDetPlane()
getHitCoordinates()
getHitCovariance()
inherit
from
silicon strip
hit
implements
projection matrix:
getHMatrix()
silicon pixel
hit
as above
GFAbsRecoHit
Implements method:
getResidual ()
virtual methods to be implemented by
derived classes:
virtual getDetPlane()=0
virtual getHitCoordinates()=0
virtual getHitCovariance()=0
virtual getHMatrix()=0
inherit
from
GFRecoHitIfc
<GFSpacepointPolicy>
as above
inherits
from
TPC
hit
as above
straw tube
array hit
as above
GFRecoHitIfc
<GFWirePolicy>
as above
inherit
from
other drift
chamber
as above
GENFIT - framework code
detector
developer code