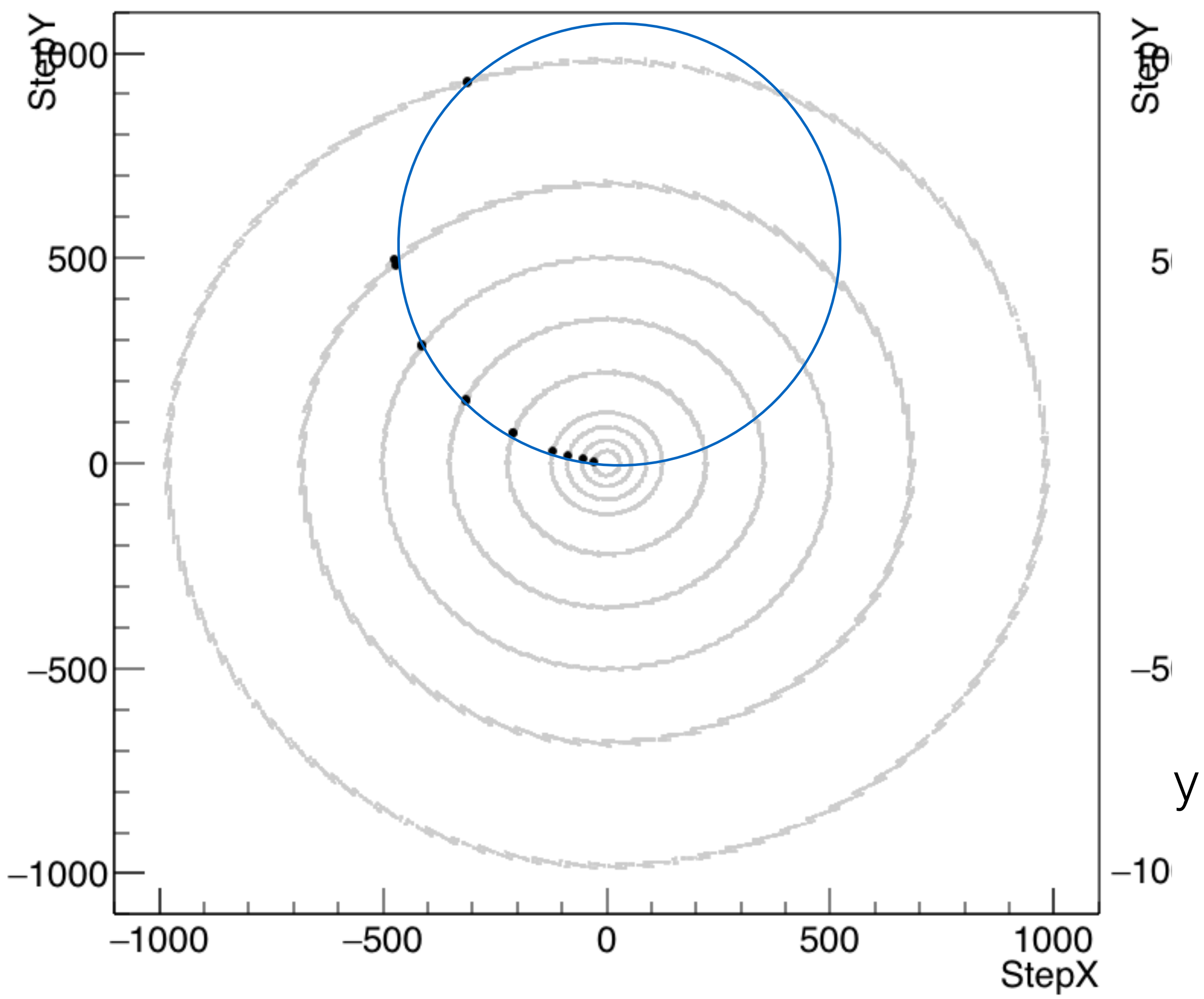
Track Reconstruction

in High Energy Physics &
in the era of Machine Learning/Data Science

A. Salzburger (CERN)

Big European Bubble Chamber at CERN, photograph http://www.cern.ch

Introduction Charged particles



hits from 1 particle

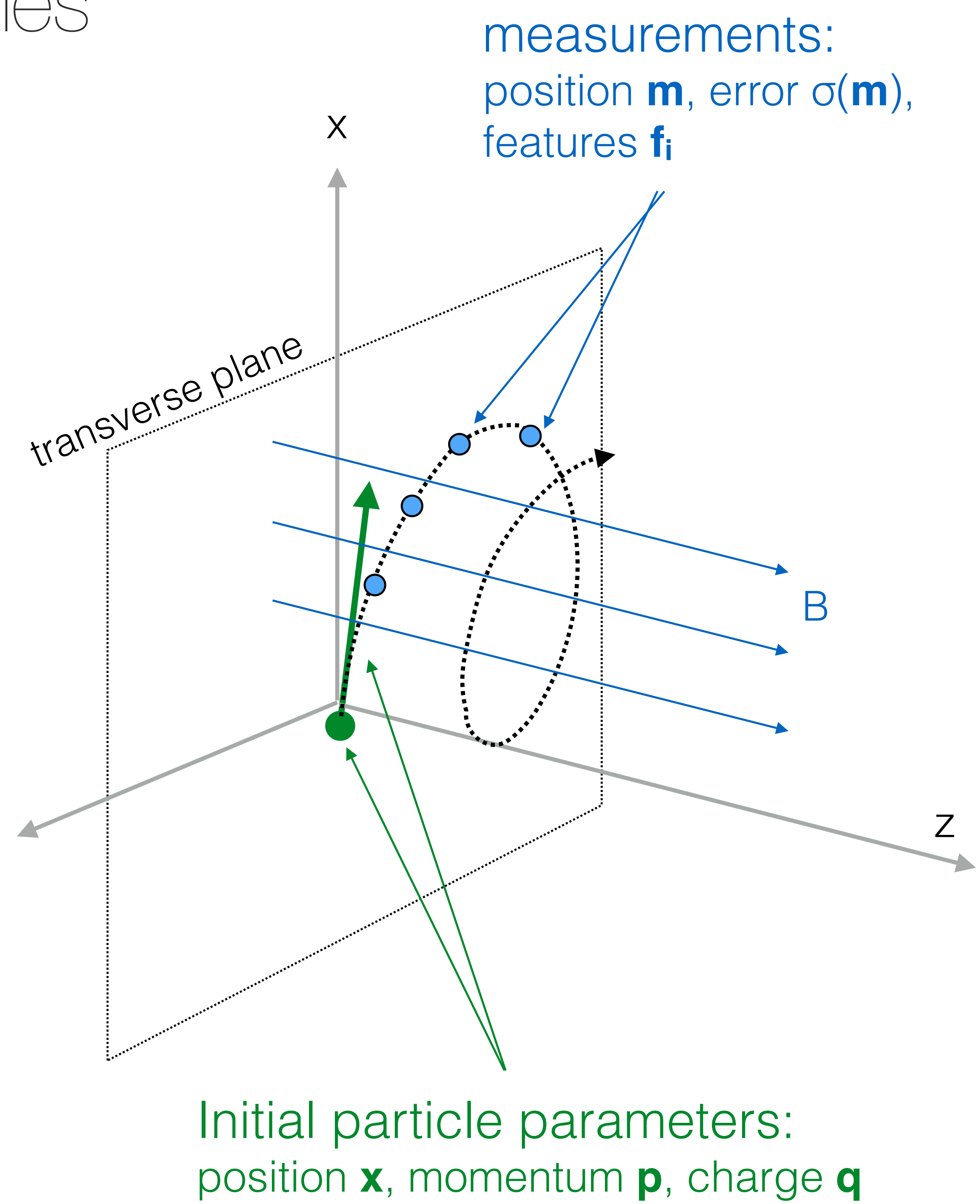


Illustration:
Hits (in transverse view) created in a tracking detector with constant magnetic field without interaction with the detector material (left), A schematic view of a particle moving in a constant magnetic field (right).

Introduction Charged particles in the detector

Particle trajectories can not be directly measured and have to be reconstructed from localisations.

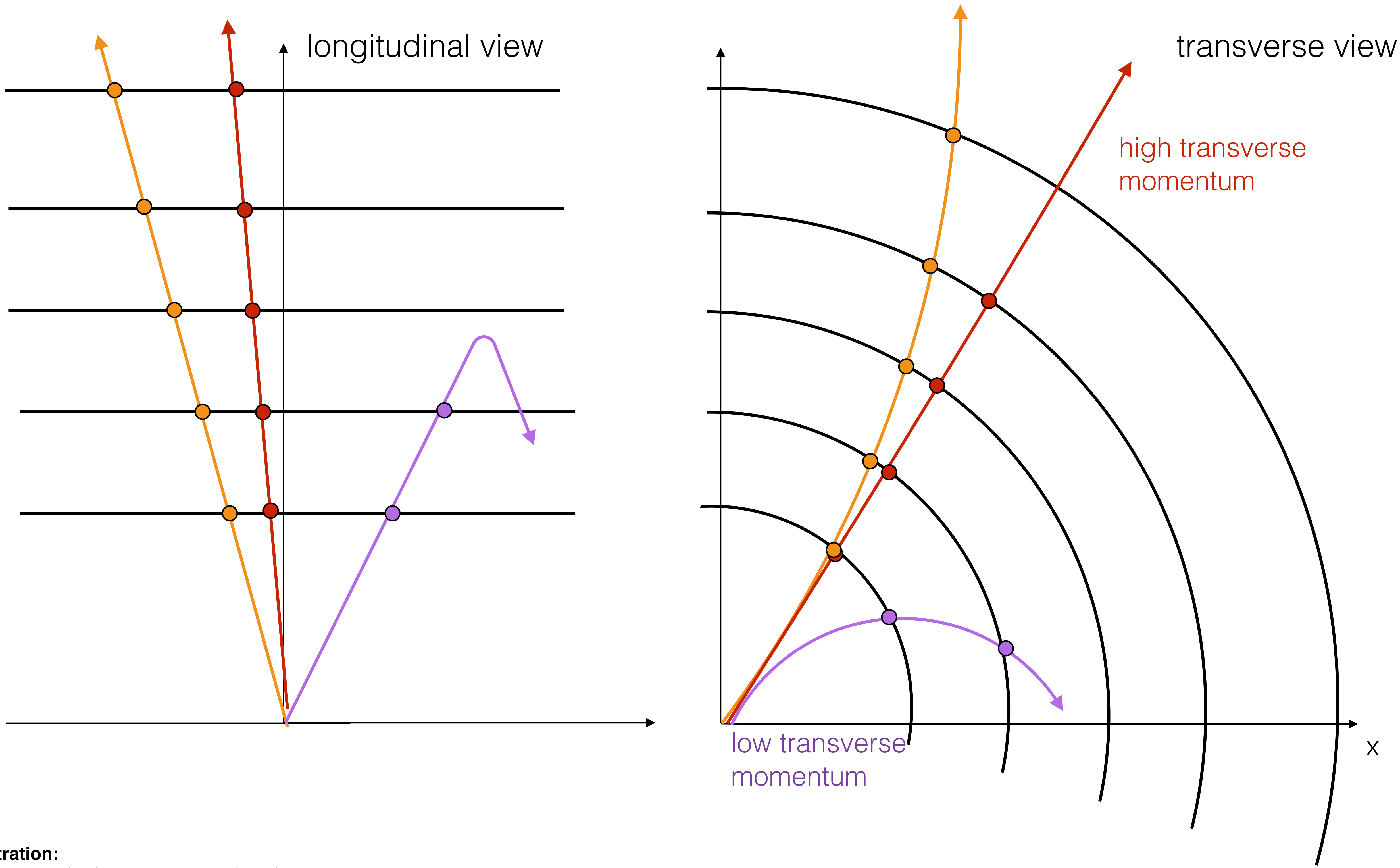


Illustration:
Longitudinal (left) and transverse (right) schematic of charged particles in a test detector.

Introduction Tracking Detectors

Tracking detectors are innermost detection devices, closest to the beam interaction region:

- measure trajectory and origin of
- charged particles

track reconstruction:
trajectory measurement gives access to kinematic information of the charged particle:
origin, momentum, charge

vertex reconstruction:
association of particles to an origin allows for grouping of particles and subsequent event reconstruction

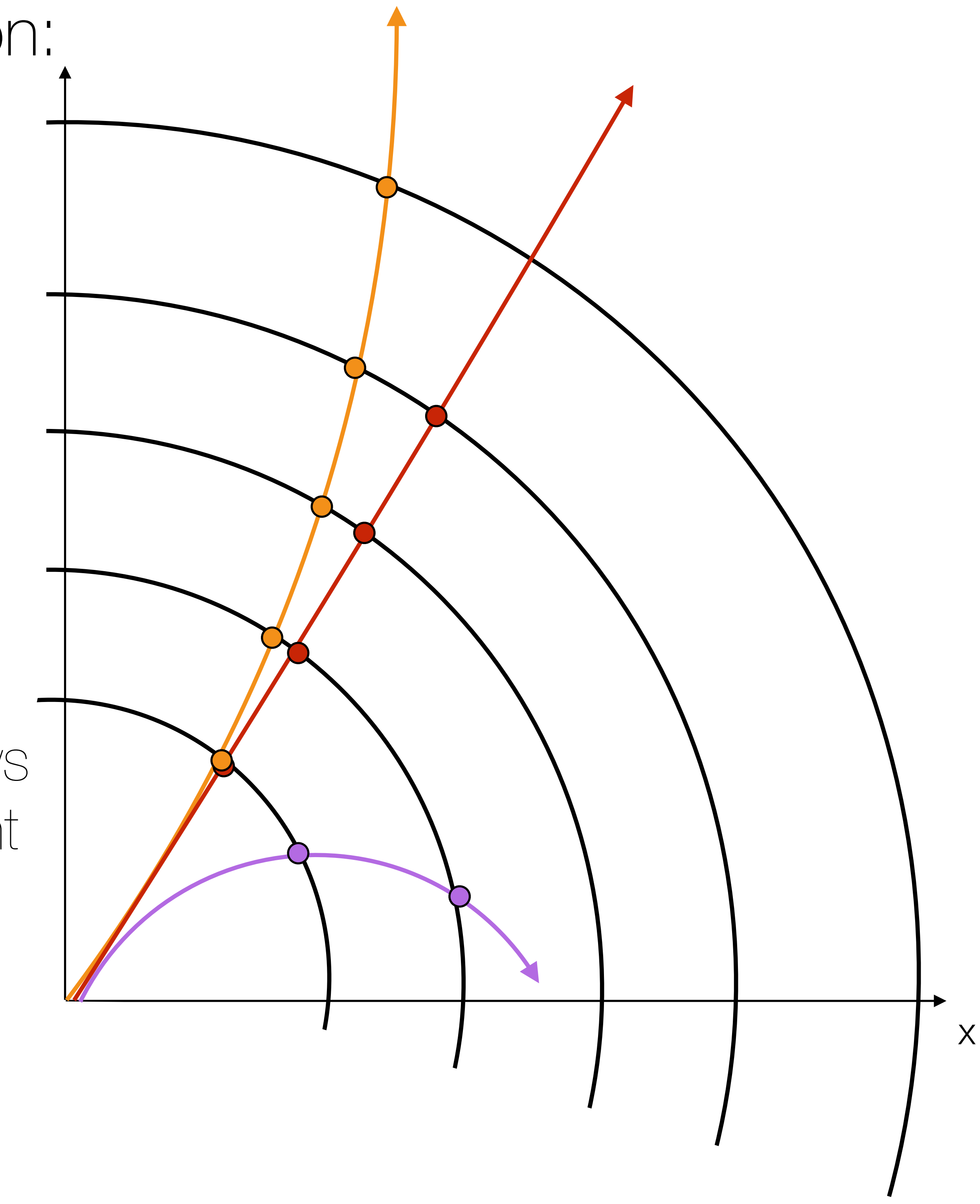


Illustration:
Transverse (right) schematic of charged particles in a test detector.

Introduction Tracking detectors

Typical setup of a Tracking detector

- very precise innermost tracker: silicon pixel detector
- several additional detection layers, e.g. silicon strip detectors
- embedded in a magnetic field for particle bending (momentum measurement)
- hermetic coverage, highly efficient, radiation tolerant

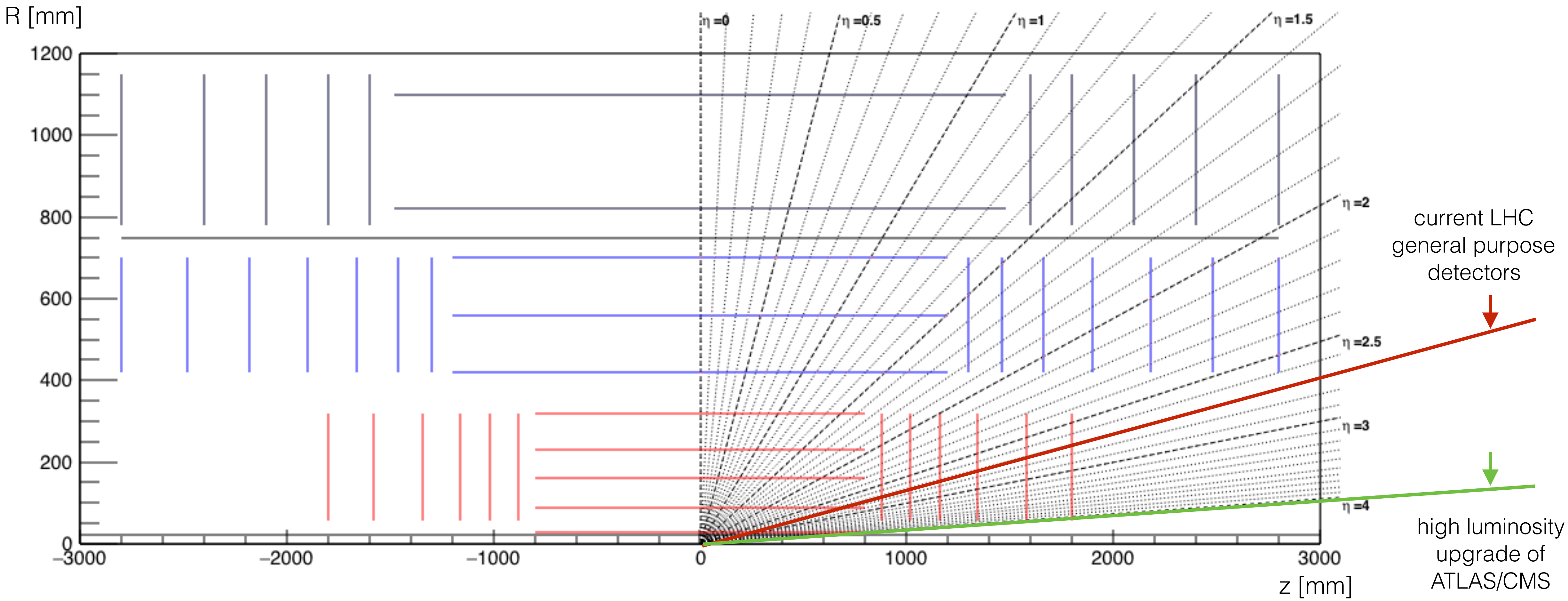


Illustration:
Longitudinal view of a schematic Tracking detector used for the Tracking ML challenge with a central barrel and endcap system.

Introduction Tracking detectors - pixel detector

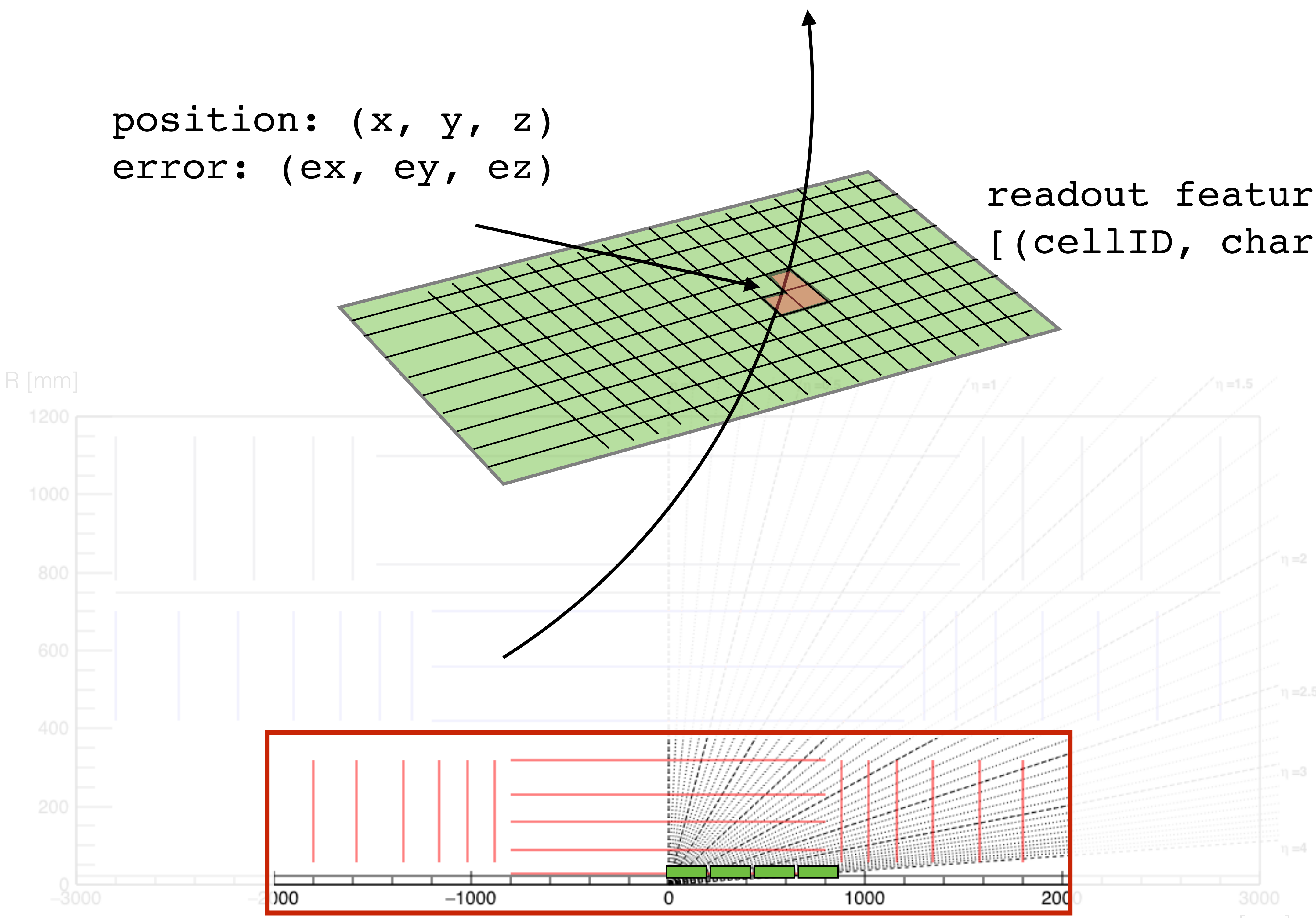
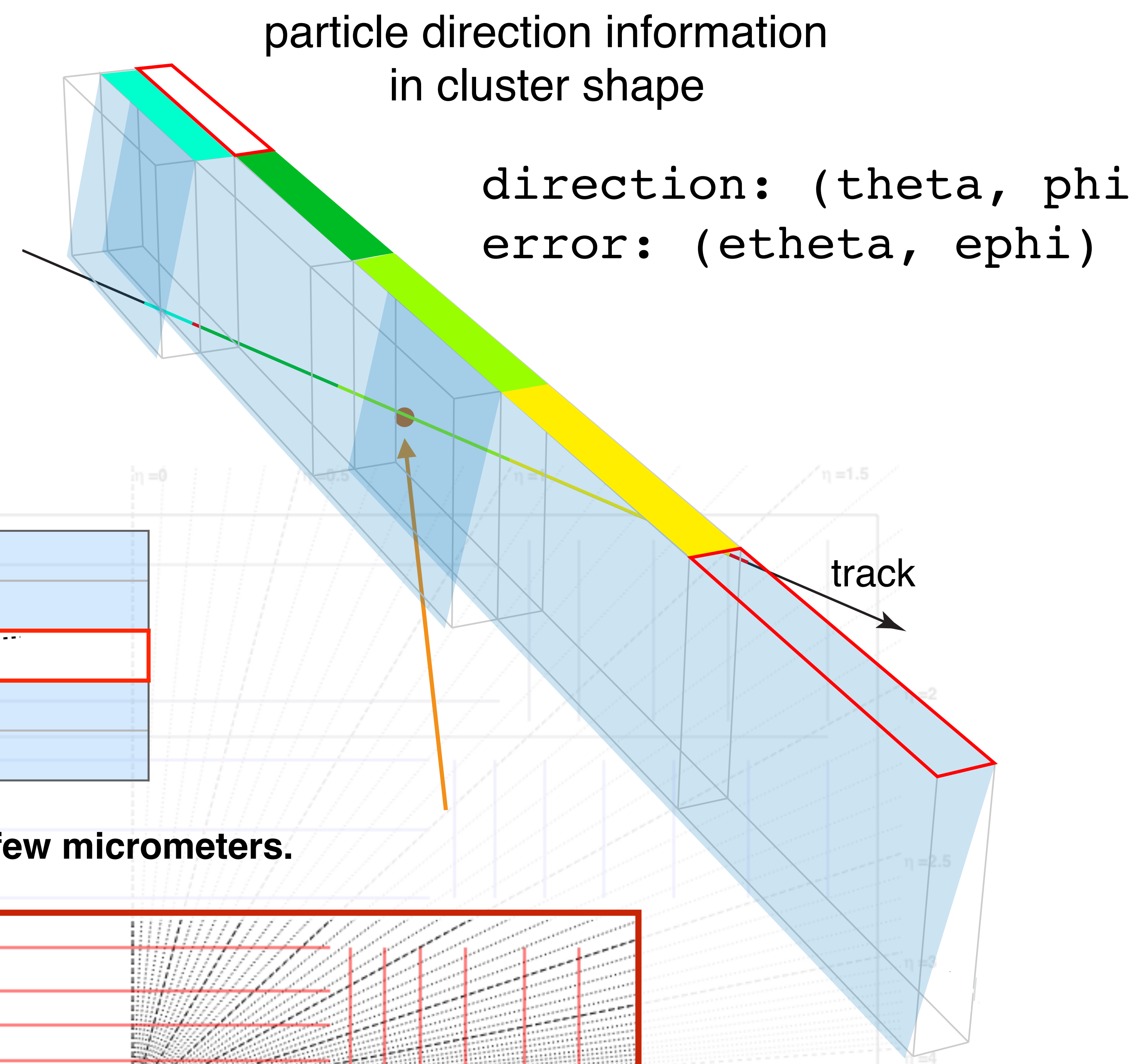
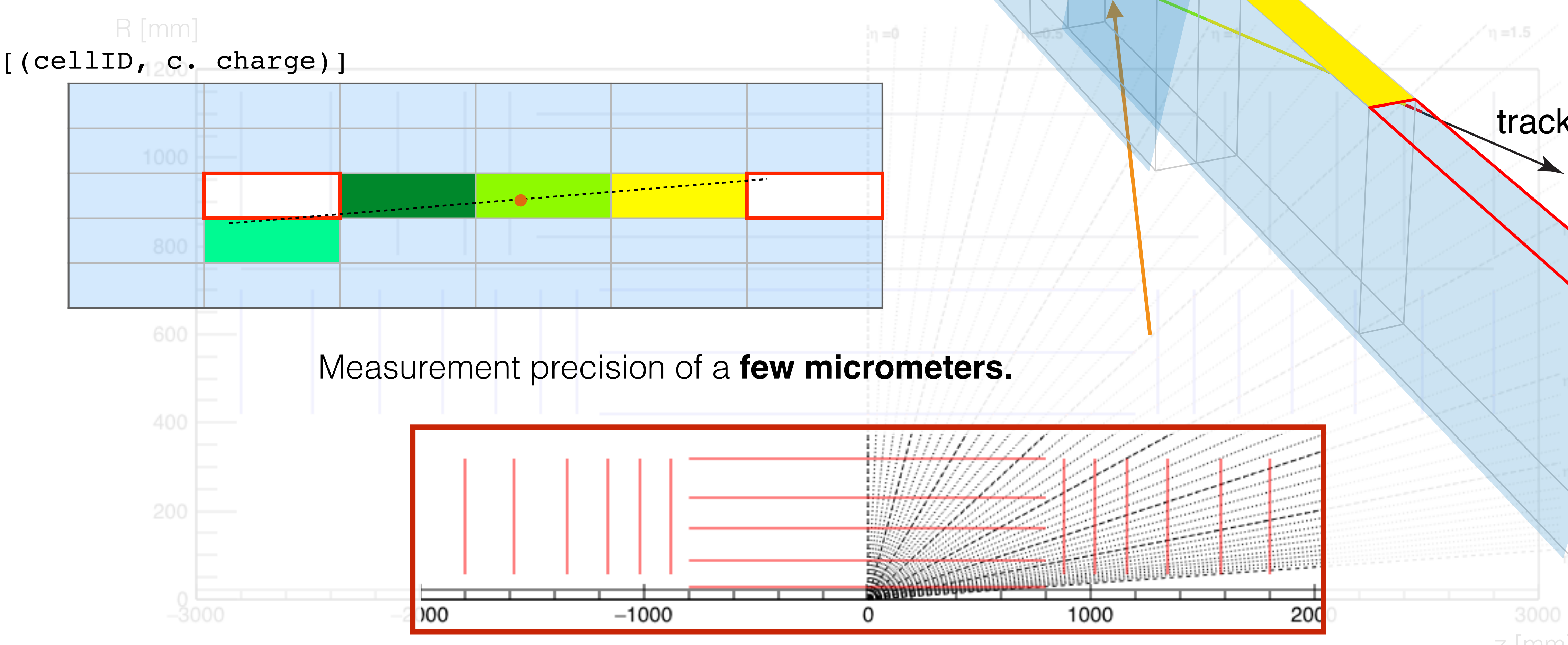


Illustration:
Longitudinal view of a schematic Tracking detector with a central barrel and endcap system. Zoom into the pixel system build from planar sensors.

Introduction Tracking detectors - pixel detector

Multiple cells hit can be used to increase measurement precision

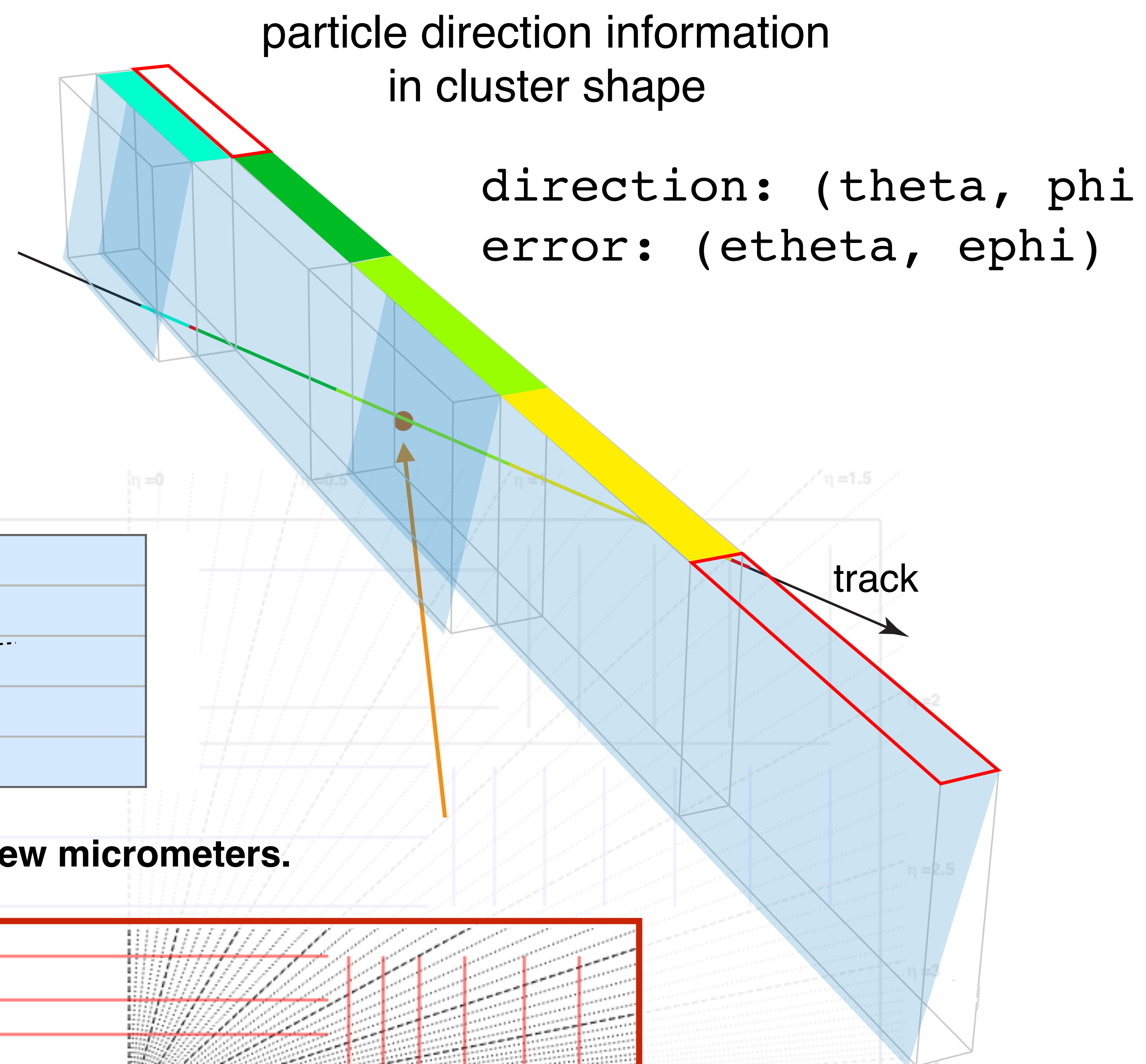


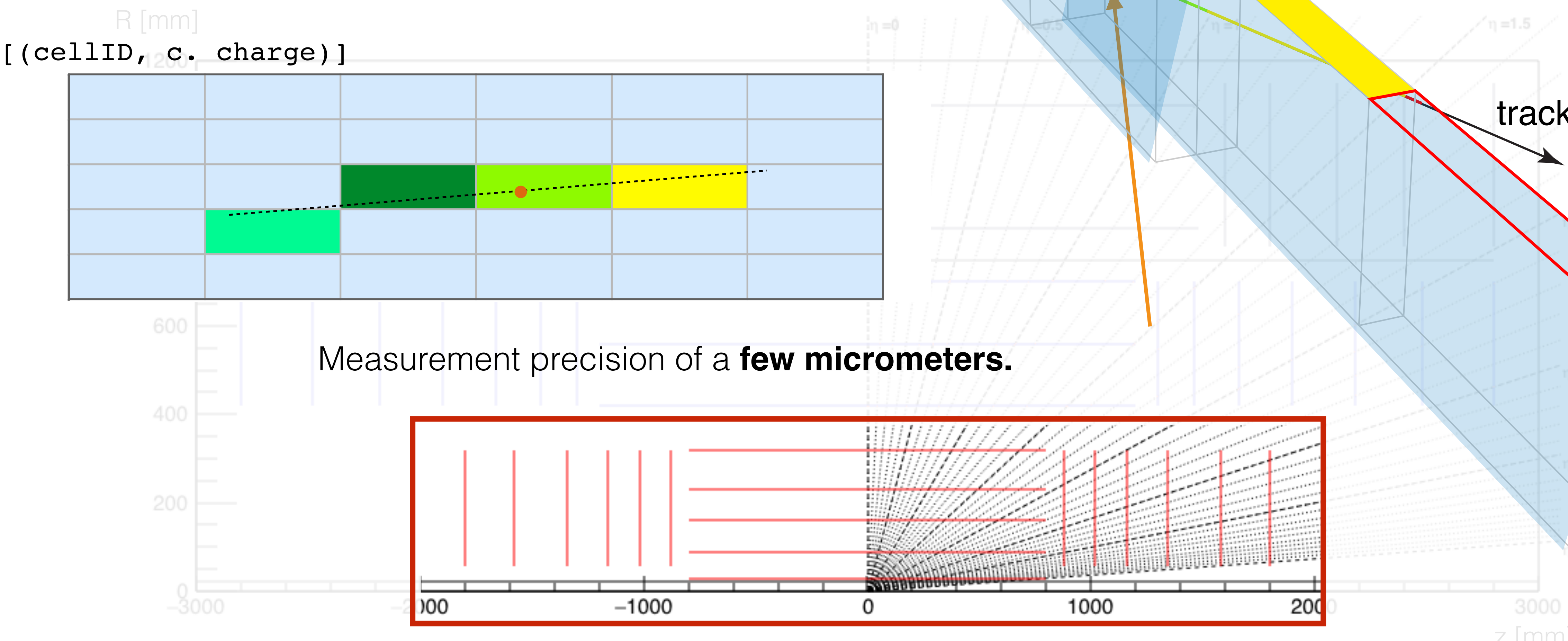


Illustrations:
A particle passing through a pixel silicon sensor: it provides a track localisation and some information about the track angle.

Introduction Tracking detectors - pixel detector

Multiple cells hit can be used to increase measurement precision





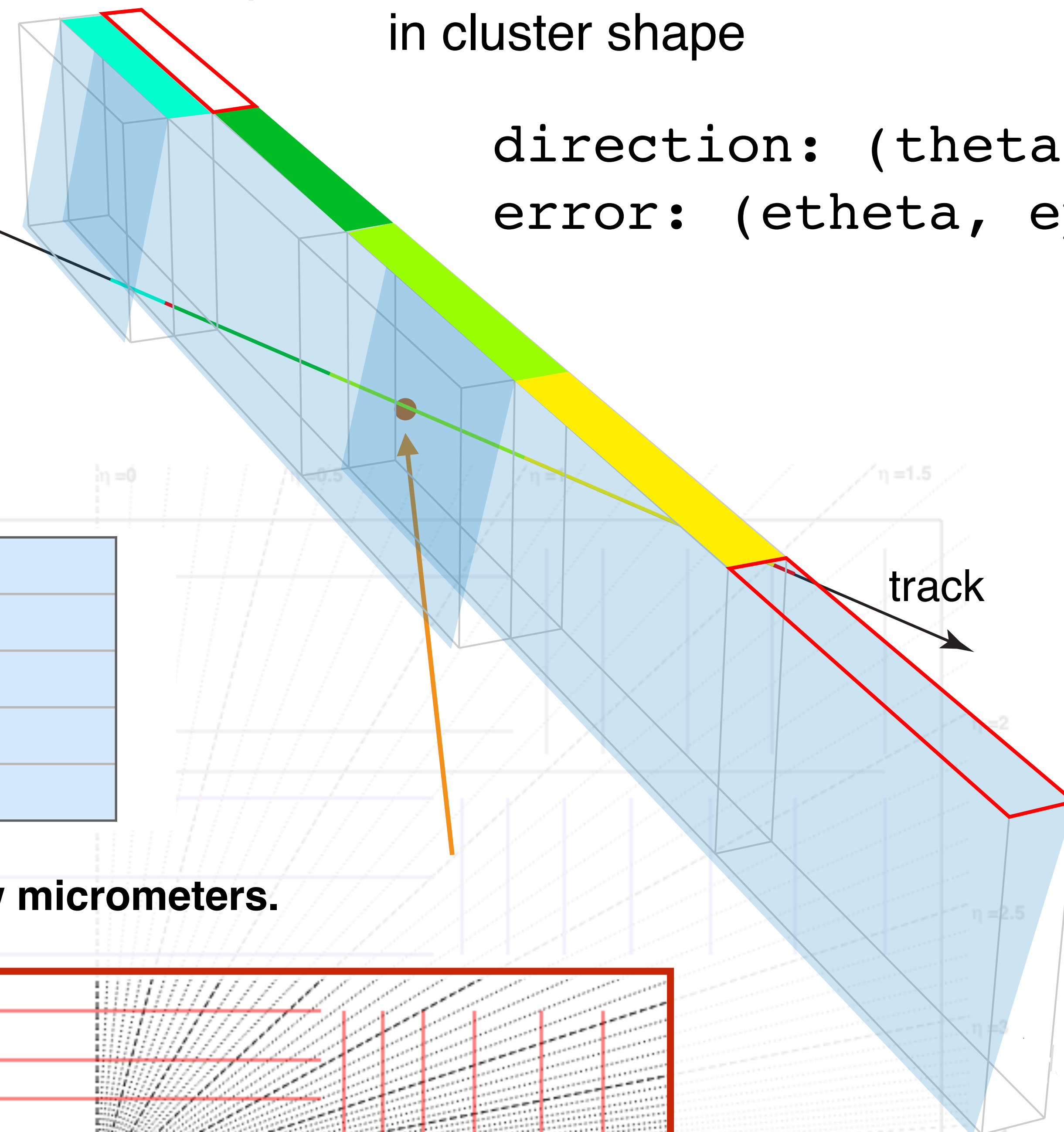
Illustrations:
A particle passing through a pixel silicon sensor: it provides a track localisation and some information about the track angle.

Introduction Tracking detectors - pixel detector

Multiple cells hit can be used to increase measurement precision

particle direction information in cluster shape

```
direction: (theta, phi)
error: (etheta, ephi)
```

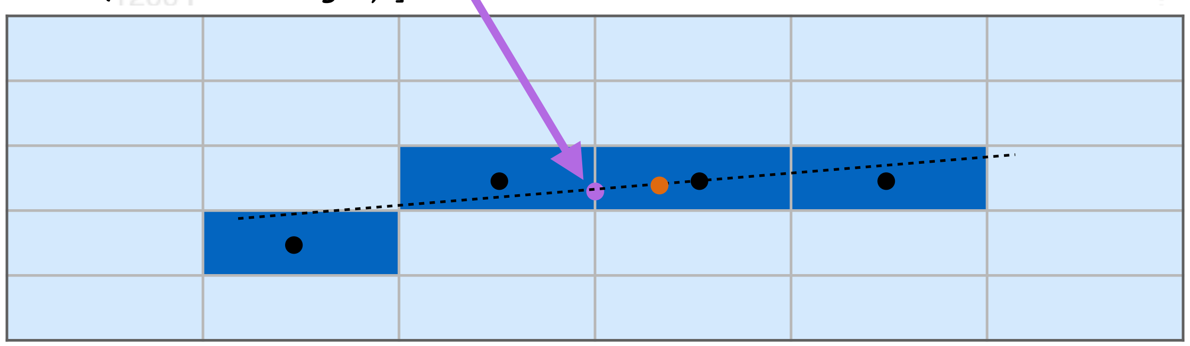


the binary approach:

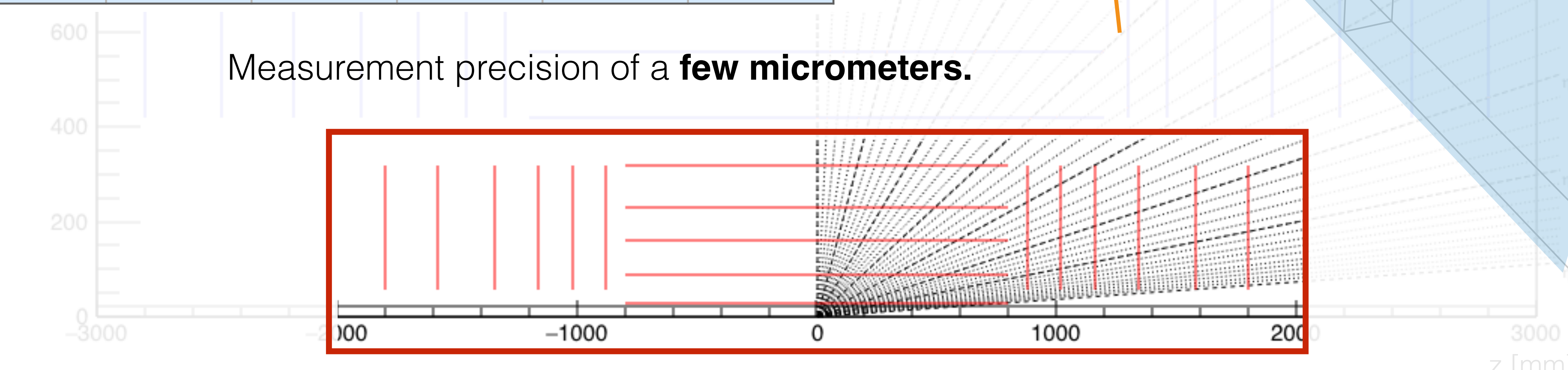
i-th pixel position

measurement $\boxed{\mathbf{m}} = \frac{1}{N} \sum_{i=1,N} \boxed{l_i}$

```
[(cellID, c. charge)]
```



Measurement precision of a **few micrometers.**



Illustrations:
A particle passing through a pixel silicon sensor: it provides a track localisation and some information about the track angle.

Introduction Tracking detectors - pixel detector

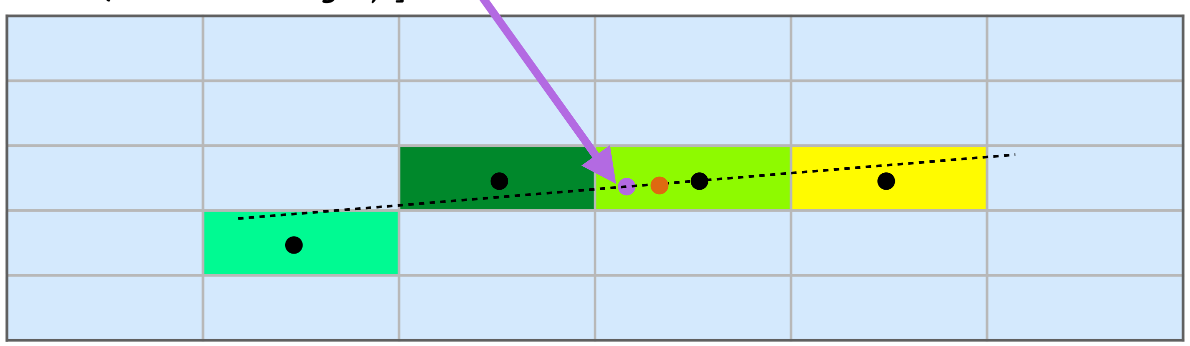
Multiple cells hit can be used to increase measurement precision

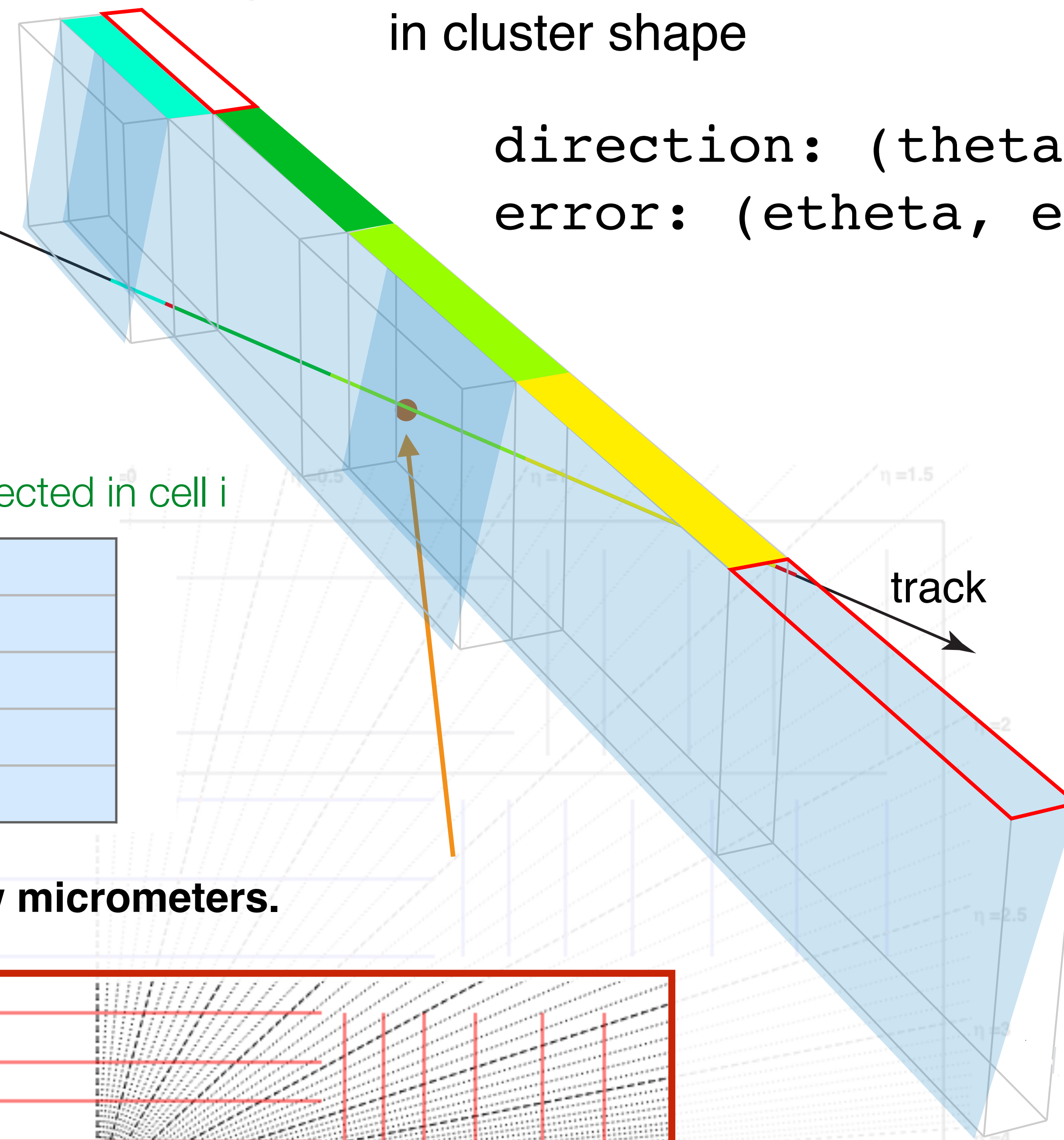
the charge-weighted approach :

$$\mathbf{m} = \frac{1}{\sum_{i=1,N} q_i} \sum_{i=1,N} q_i l_i$$

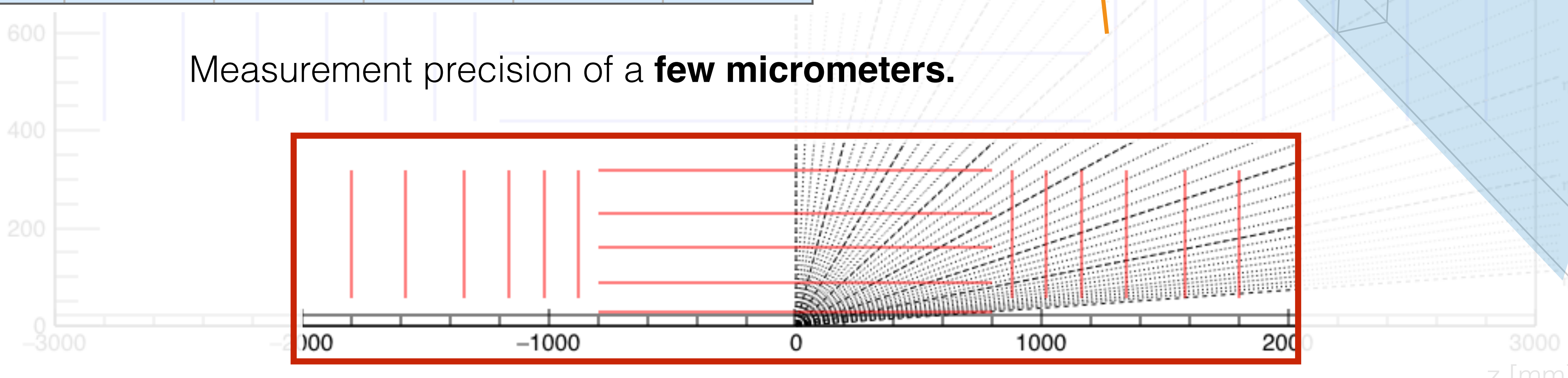
charge collected in cell i

`[(cellID, c. charge)]`





Measurement precision of a **few micrometers.**

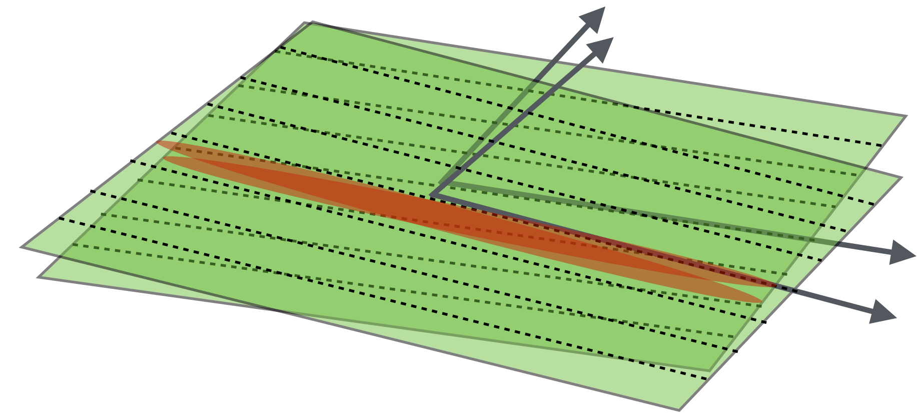


Illustrations:
A particle passing through a pixel silicon sensor: it provides a track localisation and some information about the track angle.

Introduction Tracking detectors - strip detector

Strip detector is less precise

- often built with a double module structure to achieve a 3D measurement



Measurement precision of a **few tens of micrometers.**

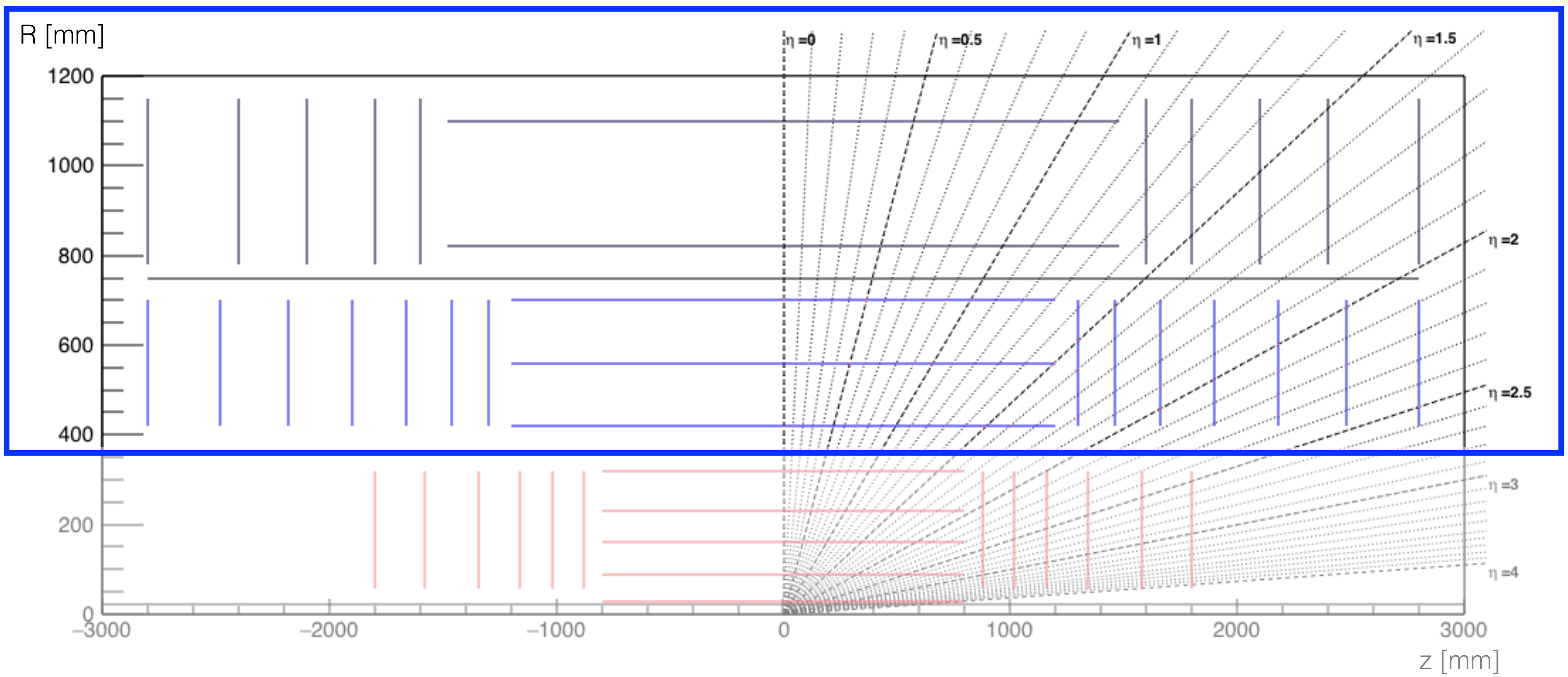
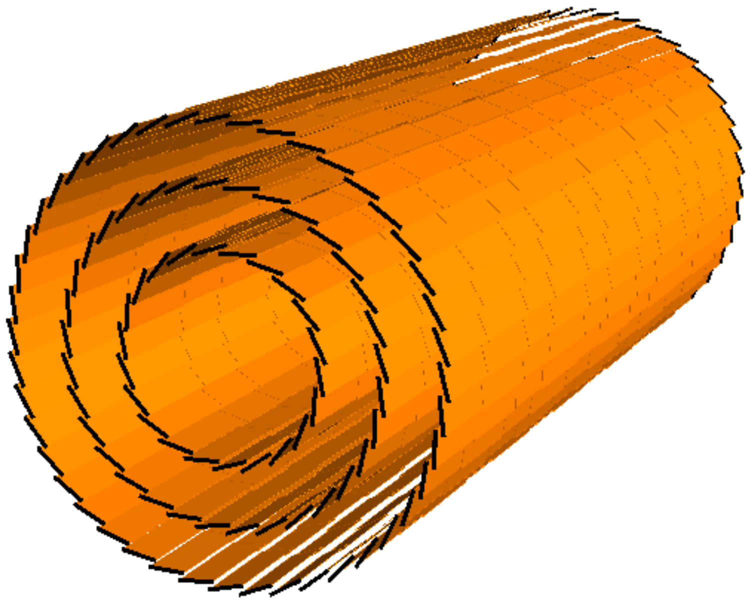


Illustration:
Longitudinal view of a schematic Tracking detector with a central barrel and endcap system.

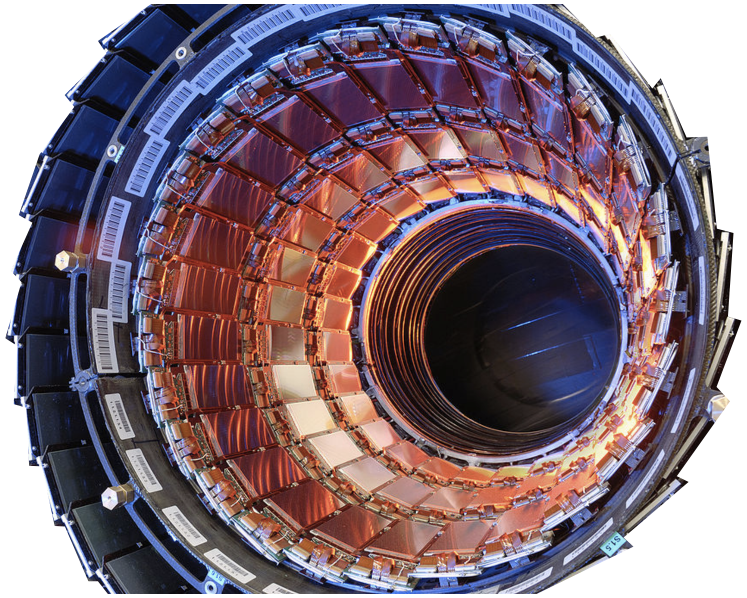
Introduction Tracking detectors

Detector material is the main source of process noise

- despite significant efforts to build the most light-weight detectors



ideal



real

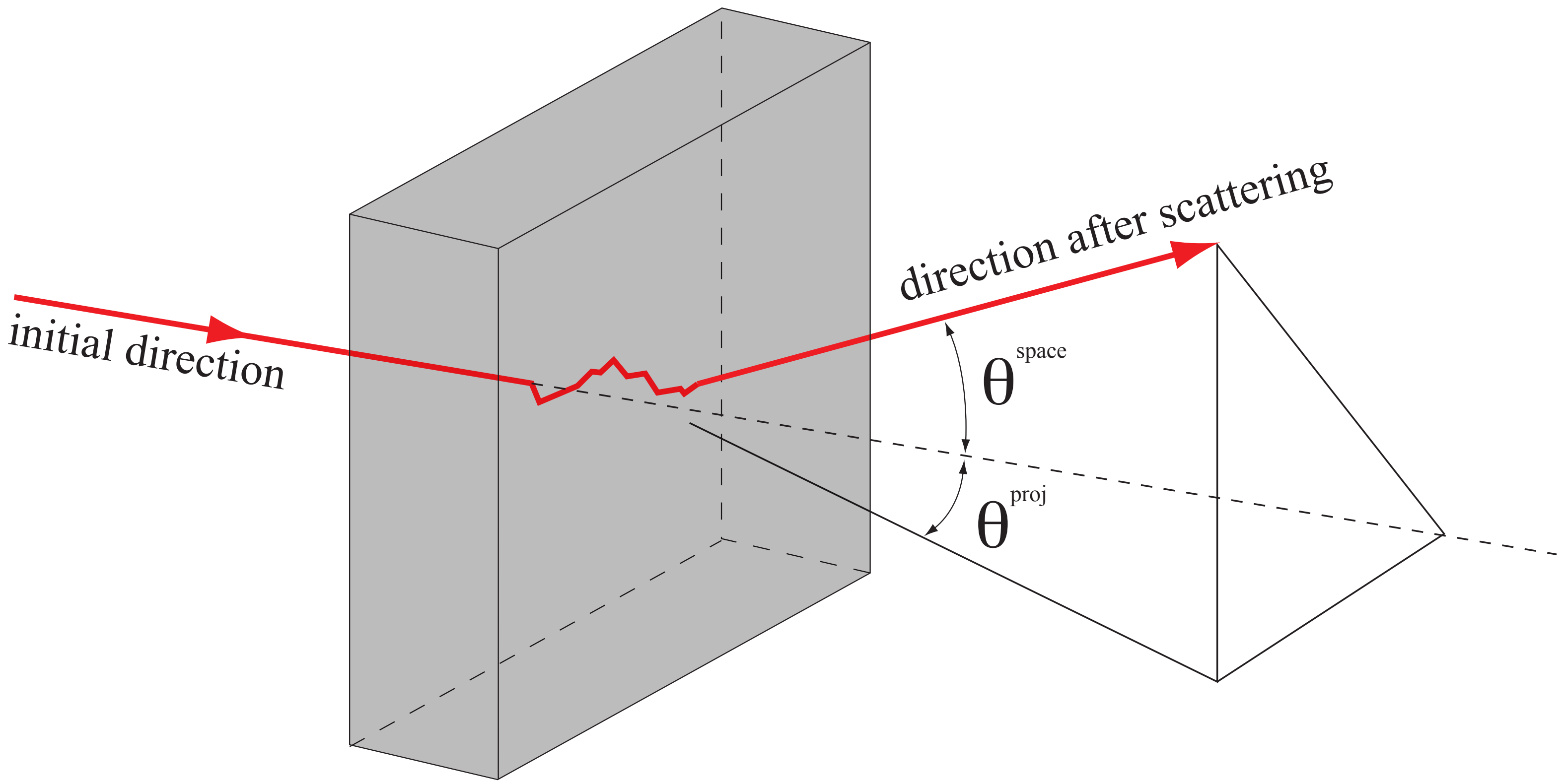
Particles interact with the detector material

- introduces different types and levels of disturbance (process noise)

Multiple Coulomb Scattering

Charged particle undergoes multiple coulomb scattering when passing through material

- net deflection: $\mathrm{Var}(\theta) = 0$
- almost Gaussian process noise (except for single large angle scattering)



$$\sigma_{ms}^{proj} = \frac{13.6\ \mathrm{MeV}}{\beta c p} Z \sqrt{t/\mathrm{X}_0}\,[1 + 0.038 \ln{(t/\mathrm{X}_0\,)}]$$

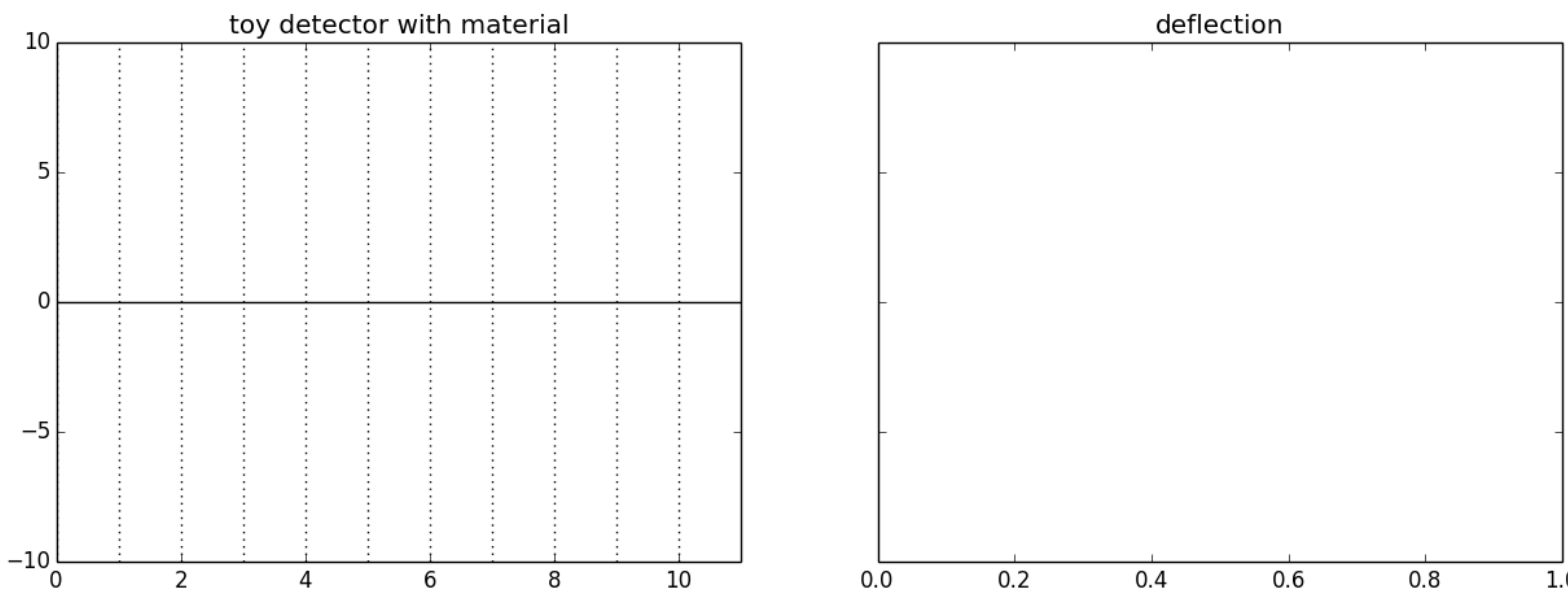
inverse proportional to momentum, i.e. low momentum particles scatter more !

Illustration:
Passage of a charged particle through dense matter resulting in a significant deflection of the particle direction.

Multiple Coulomb Scattering Effects

Passage of particle through detector material

- deflects the initial particle direction
- is inverse proportional to the particle momentum
- adds almost gaussian process noise to the measurement

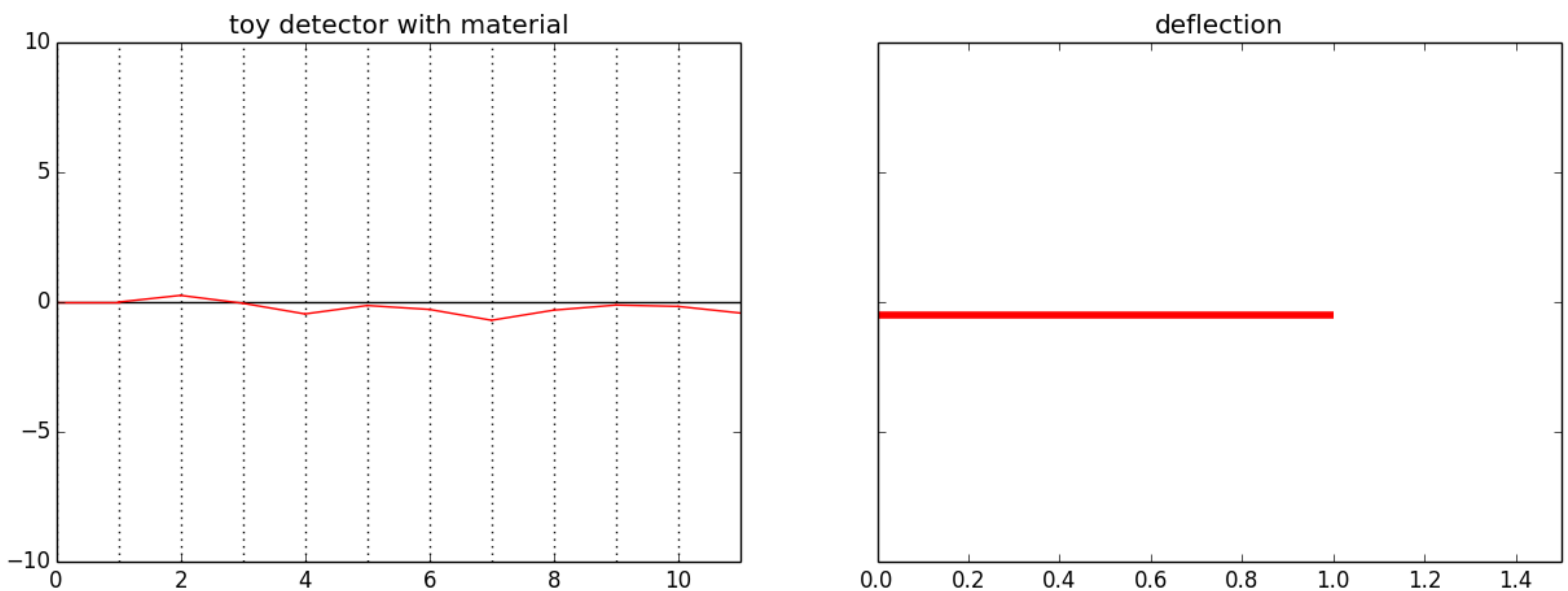


Toy Model:
Scattering emulation by passage through ten layers of material resulting in a core gaussian distribution.

Multiple Coulomb Scattering Effects

Passage of particle through detector material

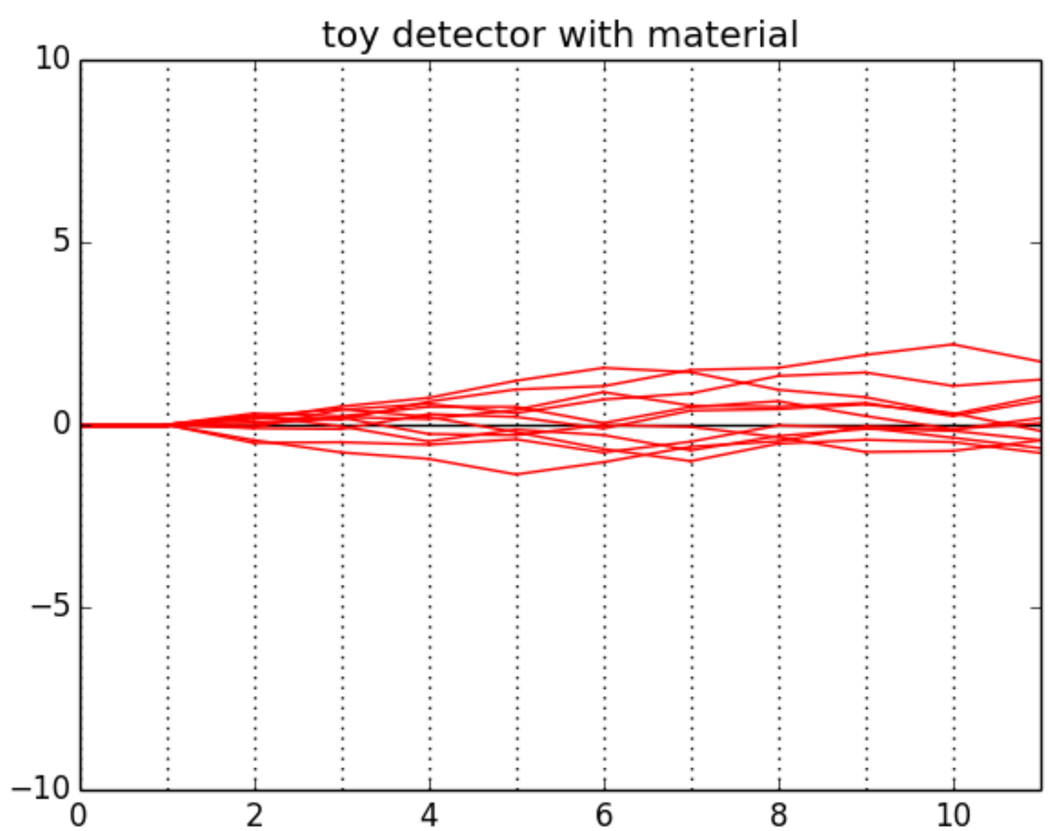
- deflects the initial particle direction
- is inverse proportional to the particle momentum
- adds almost gaussian process noise to the measurement

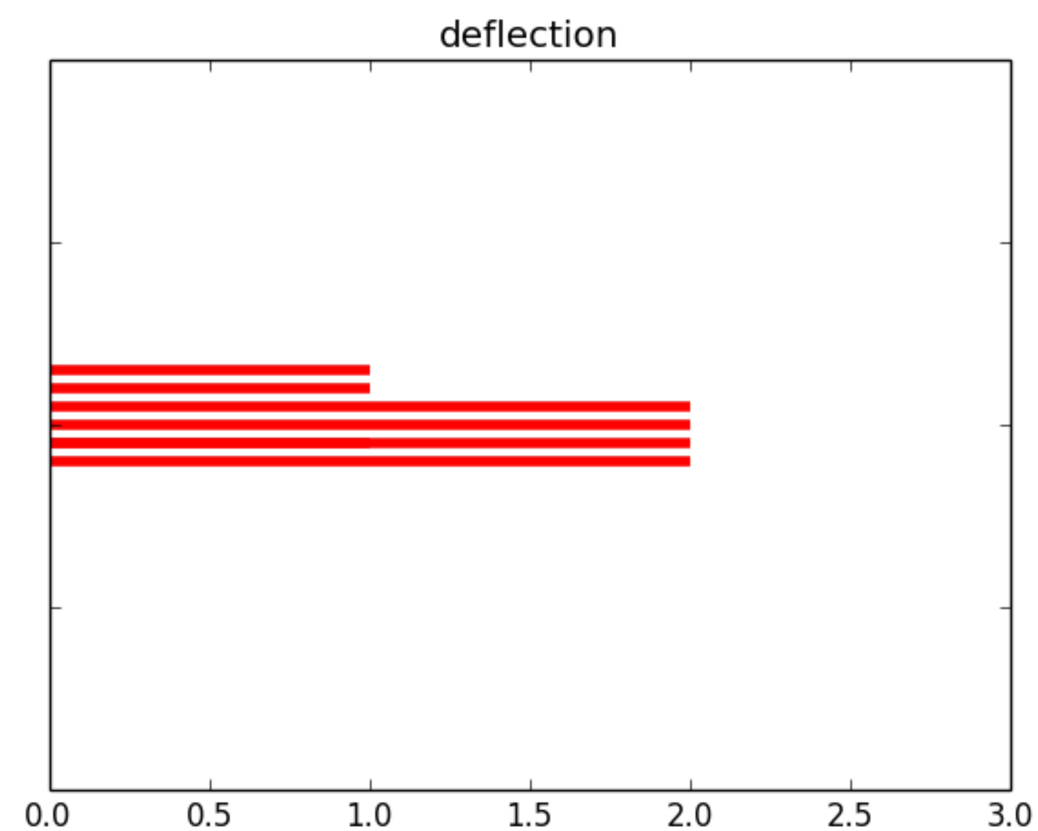


Toy Model:
Scattering emulation by passage through ten layers of material resulting in a core gaussian distribution.

Multiple Coulomb Scattering Effects

Passage of particle through detector material
- deflects the initial particle direction
- is inverse proportional to the particle momentum
- adds almost gaussian process noise to the measurement



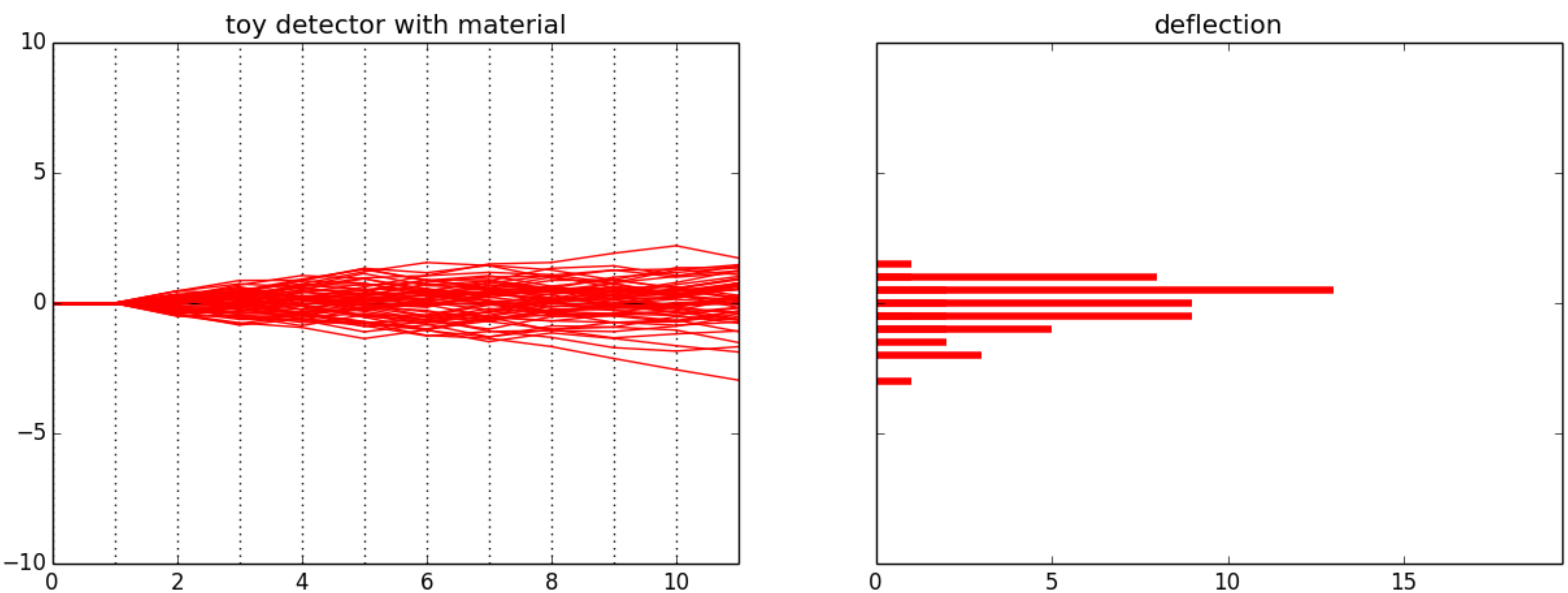


Toy Model:
Scattering emulation by passage through ten layers of material resulting in a core gaussian distribution.

Multiple Coulomb Scattering Effects

Passage of particle through detector material

- deflects the initial particle direction
- is inverse proportional to the particle momentum
- adds almost gaussian process noise to the measurement

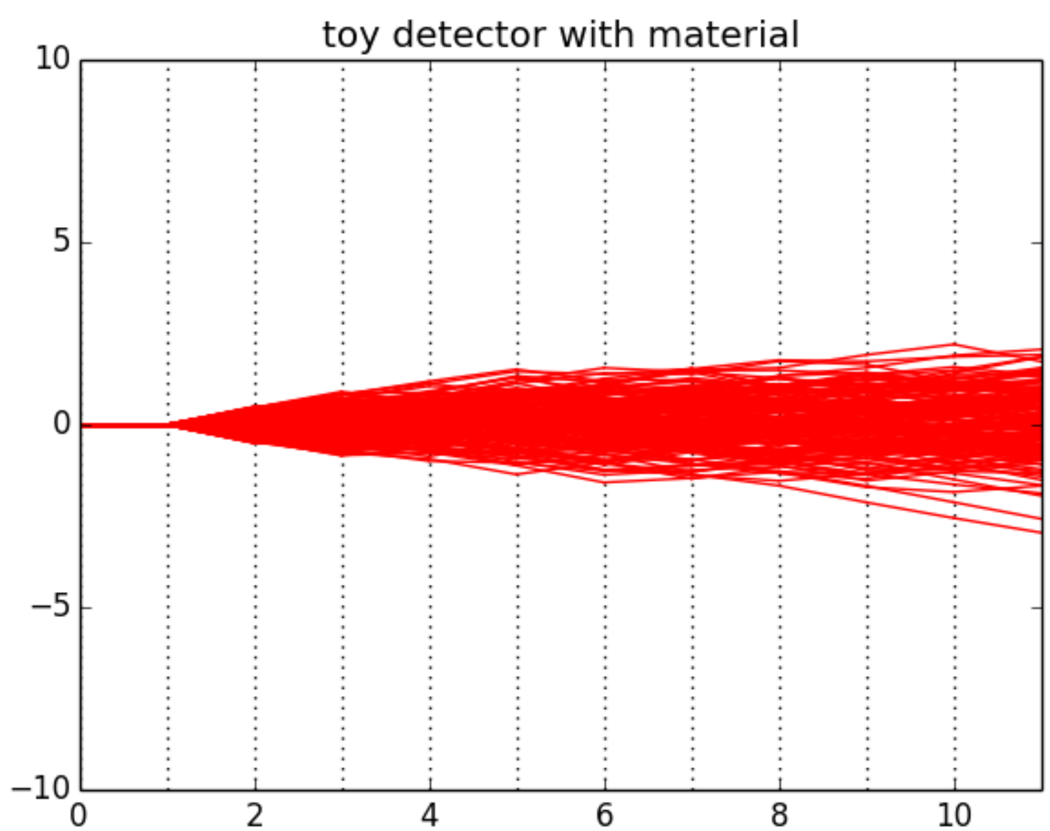


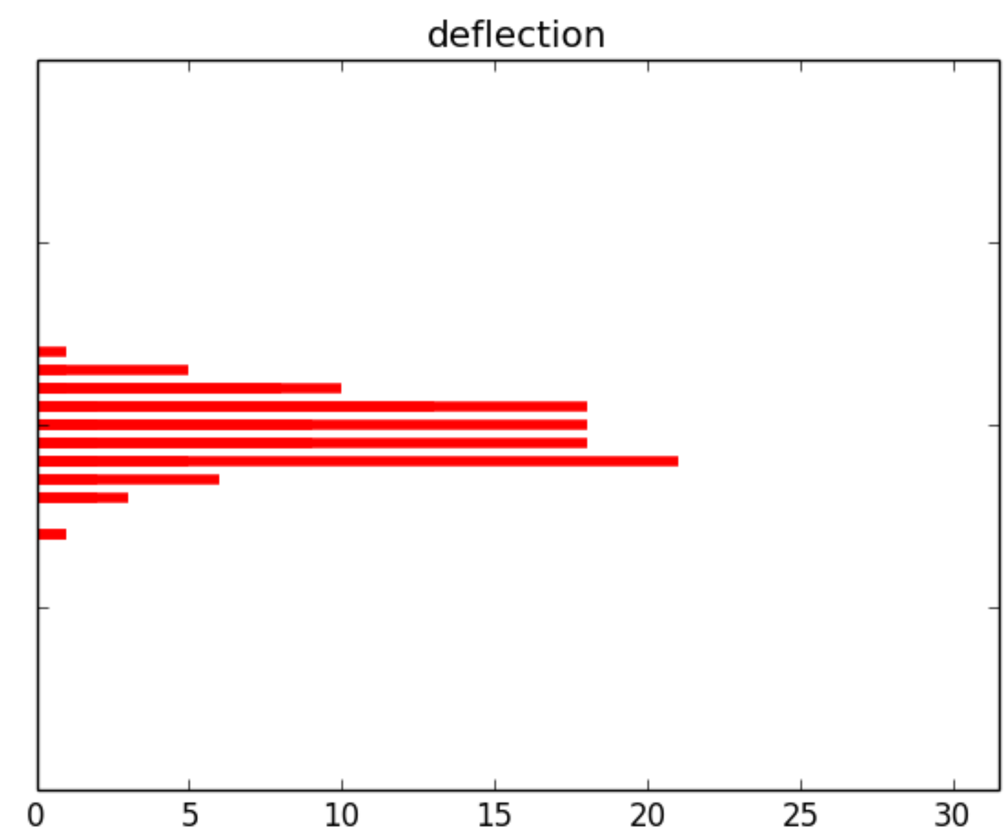
Toy Model:
Scattering emulation by passage through ten layers of material resulting in a core gaussian distribution.

Multiple Coulomb Scattering Effects

Passage of particle through detector material

- deflects the initial particle direction
- is inverse proportional to the particle momentum
- adds almost gaussian process noise to the measurement



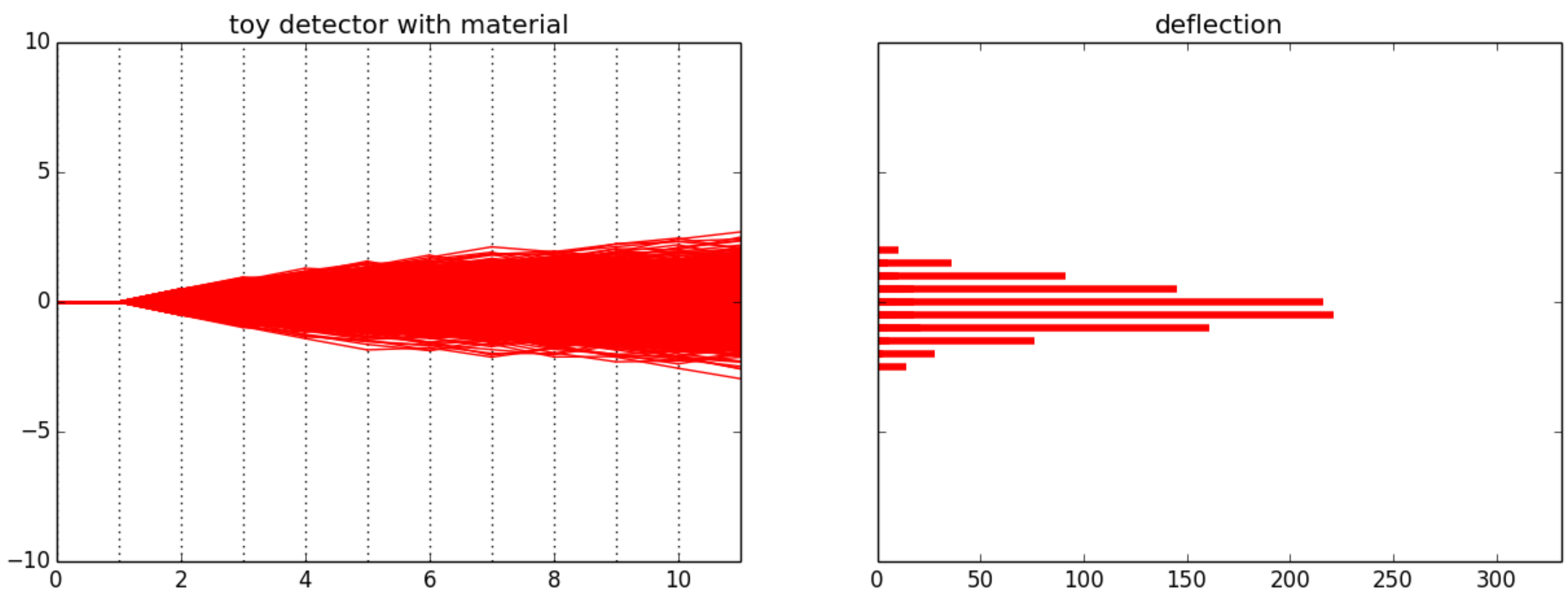


Toy Model:
Scattering emulation by passage through ten layers of material resulting in a core gaussian distribution.

Multiple Coulomb Scattering Effects

Passage of particle through detector material

- deflects the initial particle direction
- is inverse proportional to the particle momentum
- adds almost gaussian process noise to the measurement

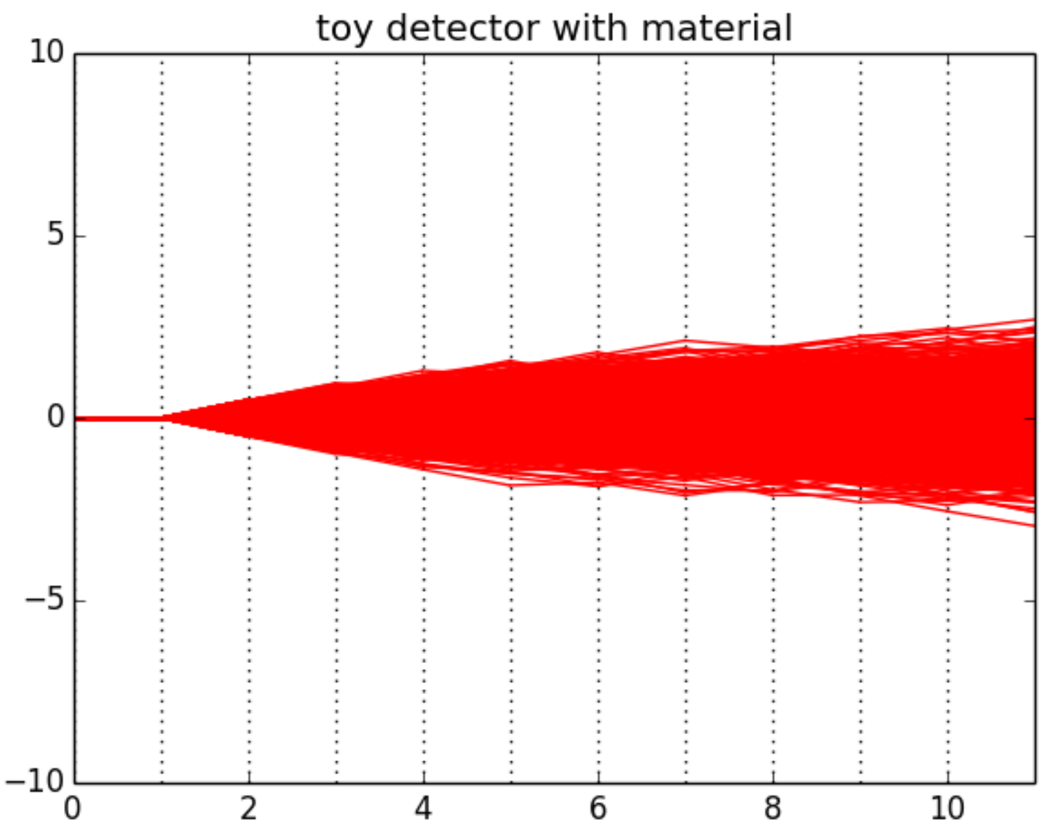


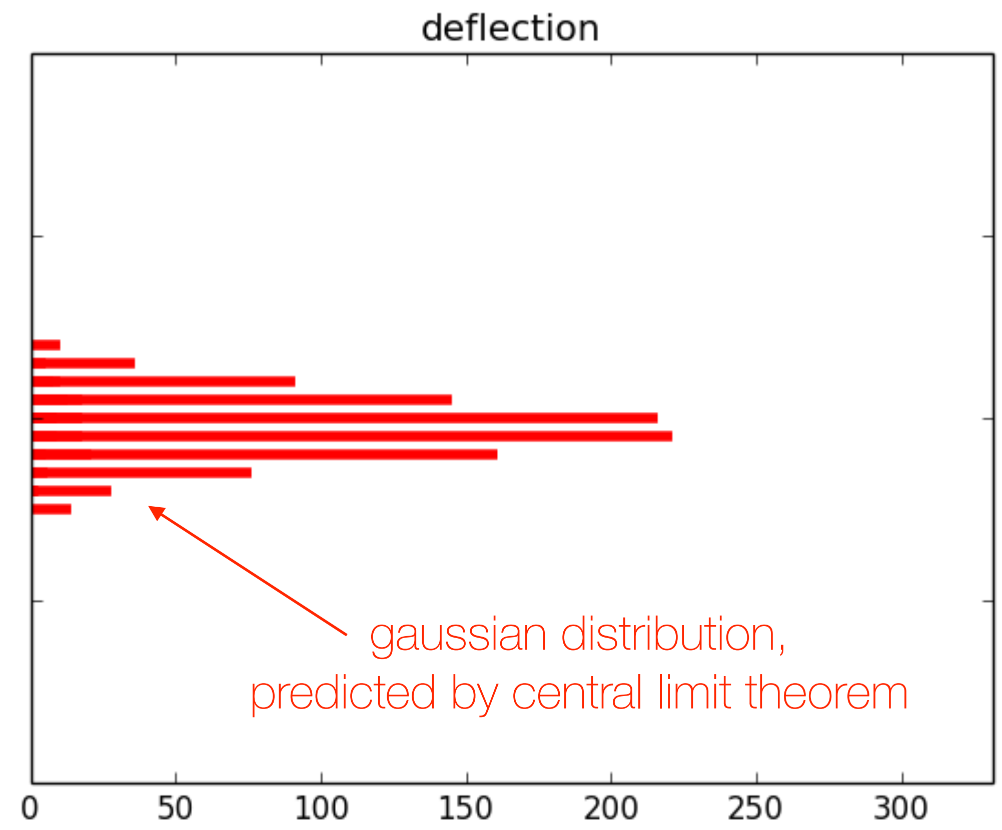
Toy Model:
Scattering emulation by passage through ten layers of material resulting in a core gaussian distribution.

Multiple Coulomb Scattering Effects

Passage of particle through detector material

- deflects the initial particle direction
- is inverse proportional to the particle momentum
- adds almost gaussian process noise to the measurement



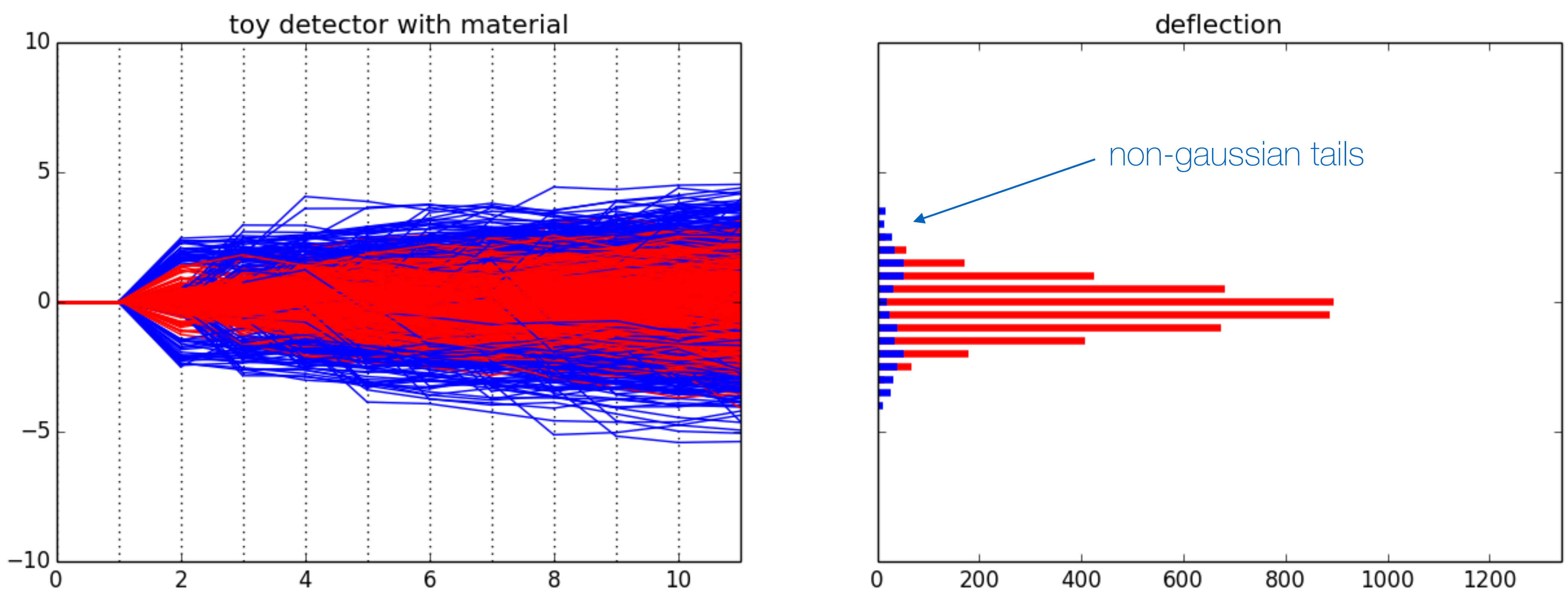


Toy Model:
Scattering emulation by passage through ten layers of material resulting in a core gaussian distribution.

Multiple Coulomb Scattering Effects

Passage of particle through detector material

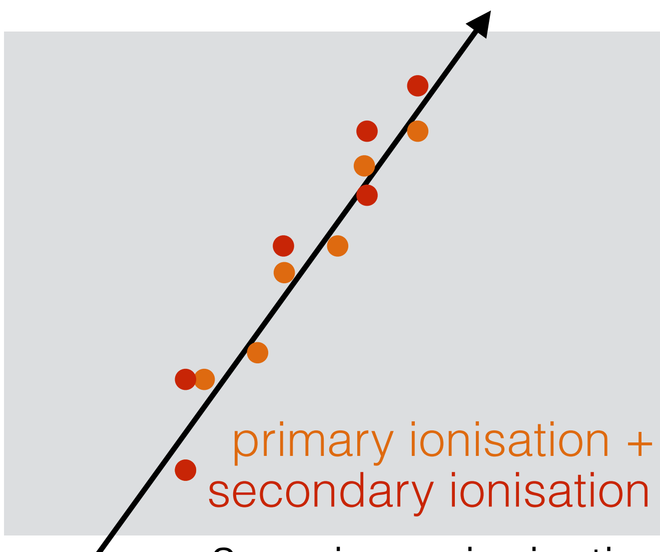
- deflects the initial particle direction
- is inverse proportional to the particle momentum
- adds almost gaussian process noise to the measurement



Toy Model:
Scattering emulation by passage through ten layers of material resulting in a core gaussian distribution.

Energy loss Effects

Ionisation

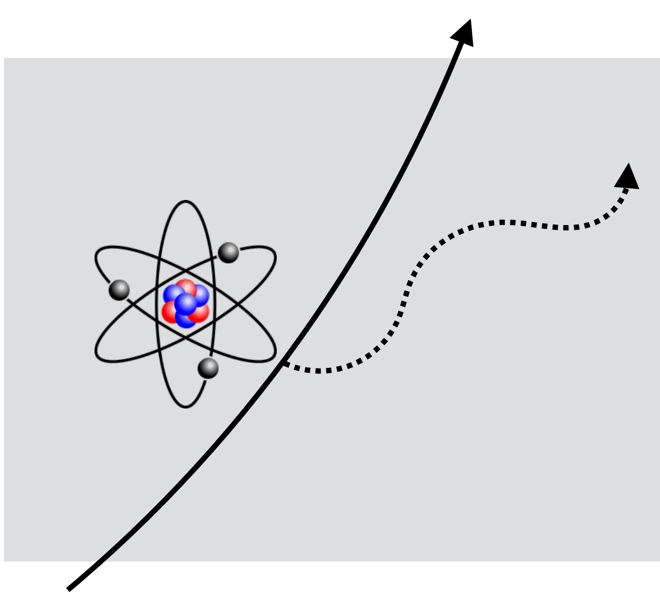


≈ 3 x primary ionisation

$$(dE/dx)_{ion} = \alpha^2 2\pi N_a \lambda_e^2 \frac{Z m_e}{A\beta^2}\left[\ln\frac{2 m_e \beta^2\gamma^2 E'_m}{I^2(Z)} - 2\beta^2 + 1/4\frac{E'^2_m}{E^2} - \delta\right]$$

N_a	$=$	$6.023 \cdot 10^{23}$, Avogadro's number
$Z,\ A$		atomic number and weight of the traversed medium
$m,\ m_e$		rest masses of the particle and the electron
β	$=$	p/E, where p is the particle momentum
γ	$=$	E/m
λ_e	$=$	$3.8616 \cdot 10^{-11}$ cm is the Compton wavelength of the electron
$I(Z)$		the mean ionisation potential of the medium,
E'_m		the maximum energy transferable to the electrons of the medium with $E'_m = 2m_e \frac{p^2}{m_e^2 + m^2 + 2m_e\sqrt{p^2+m^2}}$
δ		density correction.

Bremsstrahlung



$$(dE/dx)_{rad} = 4\alpha N_A \frac{z^2 Z^2}{A}\left(\frac{1}{4\pi\epsilon_0}\frac{e^2}{mc^2}\right)^2 E\ \ln\frac{183}{Z^{\frac{1}{3}}} \propto \frac{E}{m^2}$$

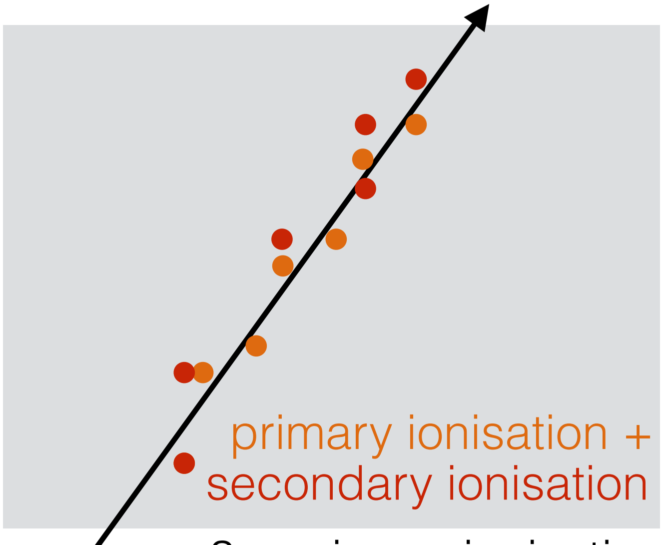
$$(dE/dx)_{rad} = -E_i/X_0$$

$$X_0 = \frac{A}{4\alpha N_A\ Z^2 r_e^2\ \ln\frac{183}{Z^{\frac{1}{3}}}}$$

radiation length

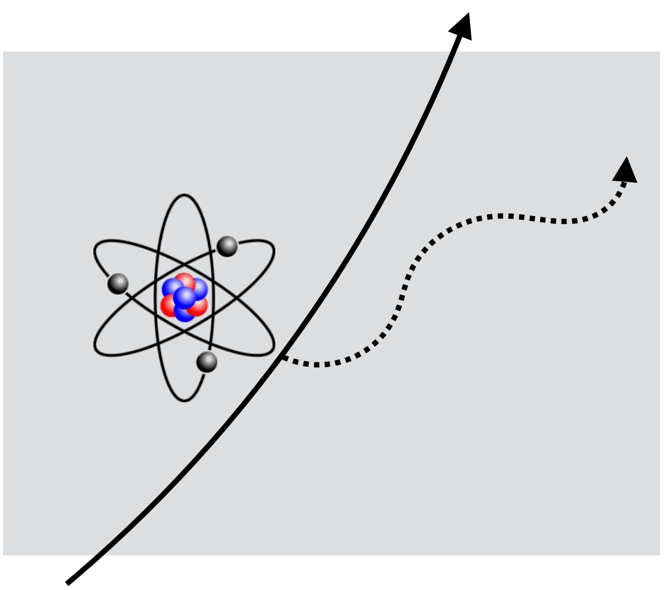
Energy loss Effects

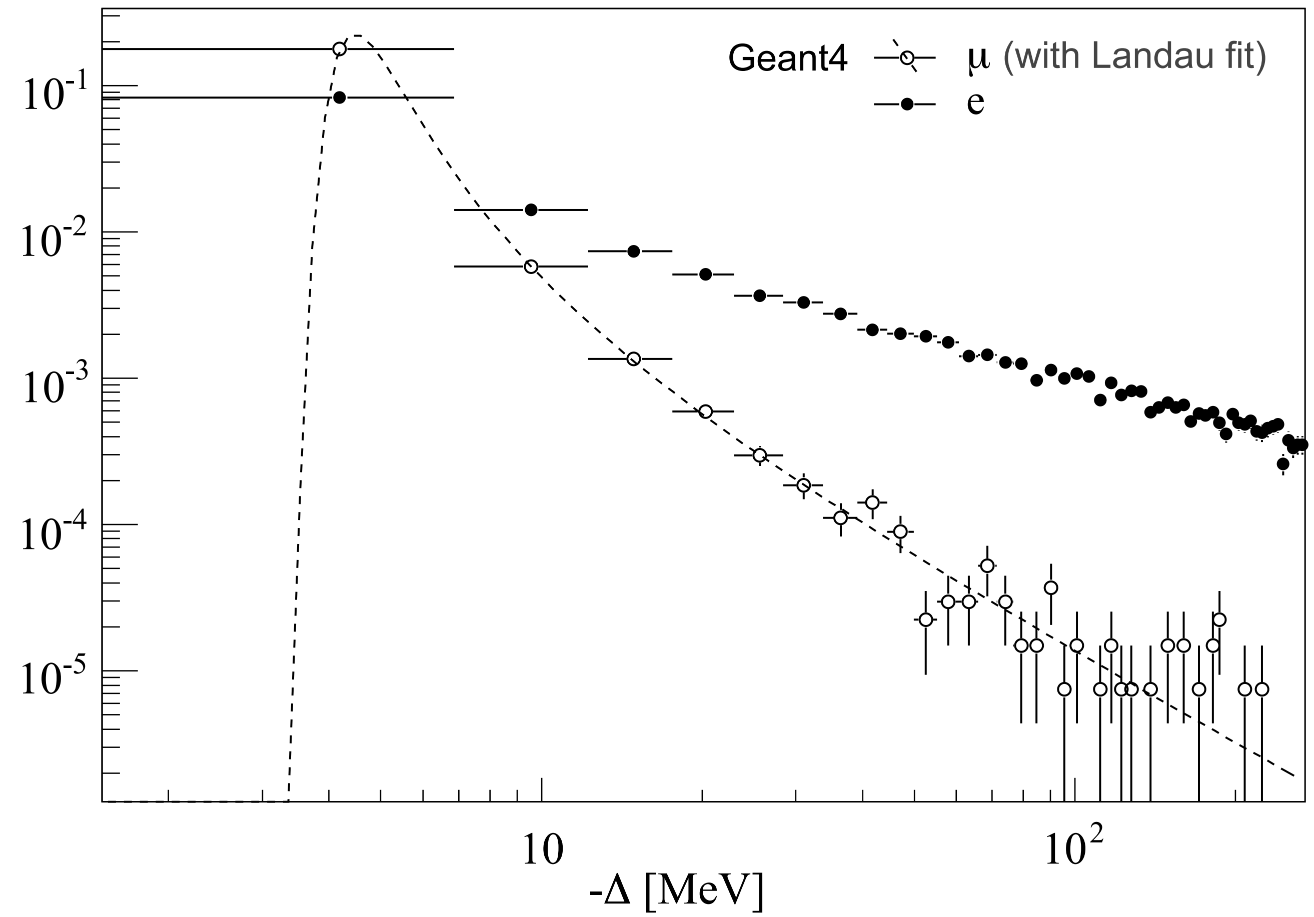
Ionisation



≈ 3 x primary ionisation

Bremsstrahlung

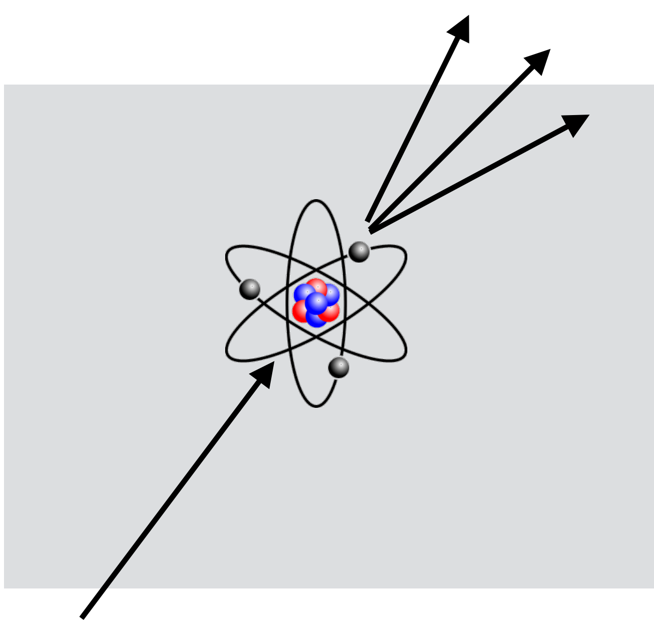




Hadronic interactions Effects

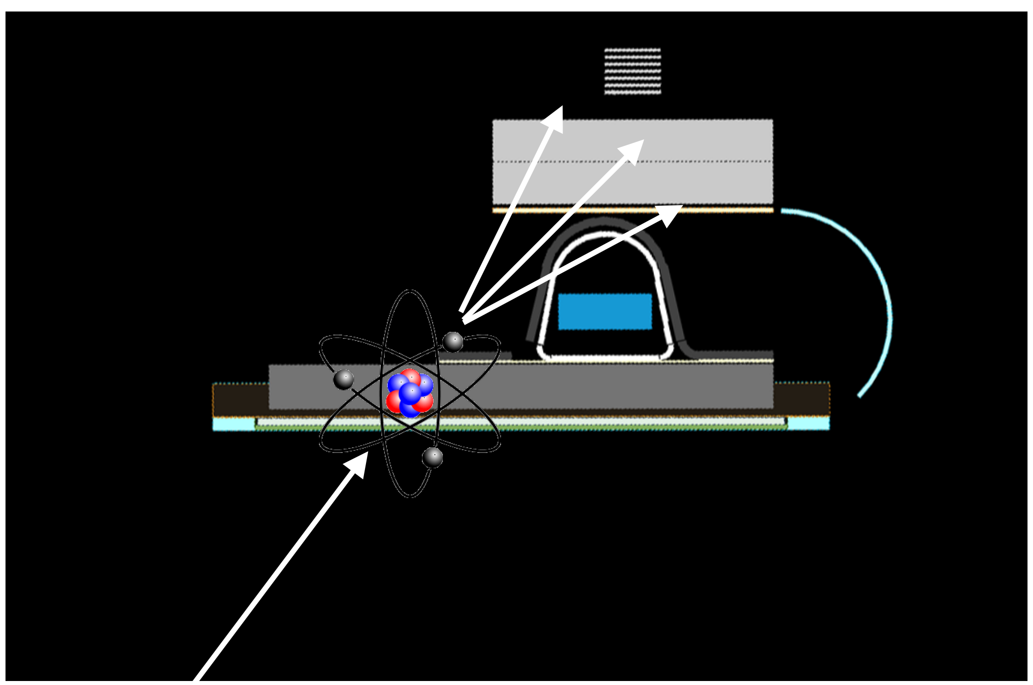
Vast majority of charged particles from p-p interactions are hadrons
- those interact with the nuclei of the detector material
- usually leads to the destruction of the particle and is the main source of inefficiency in track reconstruction

Nuclear interactions



- there are many different processes that can happen in hadron-nucleus interactions
- resulting shower has hadronic, but also EM shower components
- nuclear interaction length defined as the mean path length Λ_0 by which the number of charged particles is traversing through matter is reduced by $1/e$

Track reconstruction at LHC



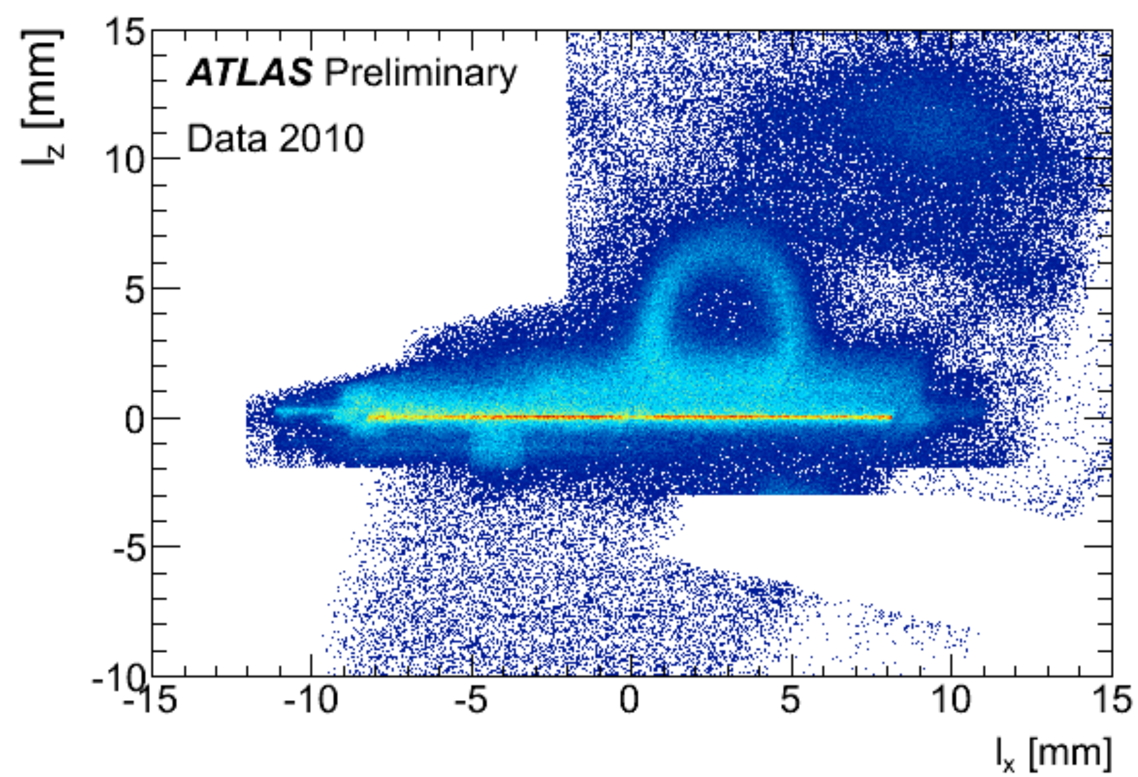


Figure:
ATLAS pixel model as described in simulation (left), tomography from vertices built from tracks for hadronic interactions (right)

Introduction Physics

The reason for the data race …

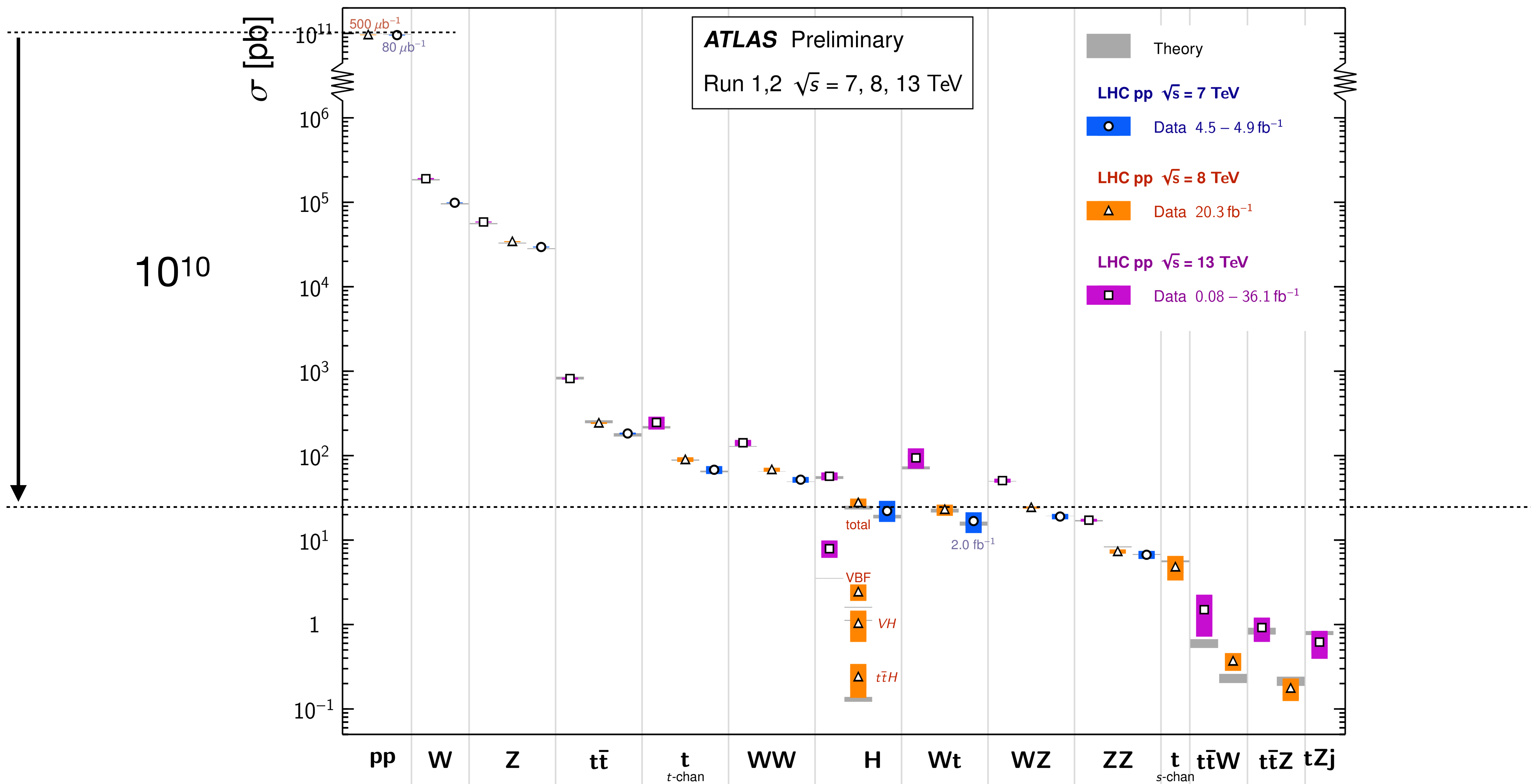


Figure:
Standard Model cross sections measured with the ATLAS experiment and compared to theoretical predictions, July 2017

Introduction Physics

Maximise (re)search potential by maximise total number of collisions

- filter (trigger) "interesting" events
- <u>increase the number of collisions per beam crossing</u>: event pile-up

Event pile-up

- when proton bunches collide multiple p-p interactions take place
- most of them are "uninteresting" p-p interactions, hoping for one interesting event

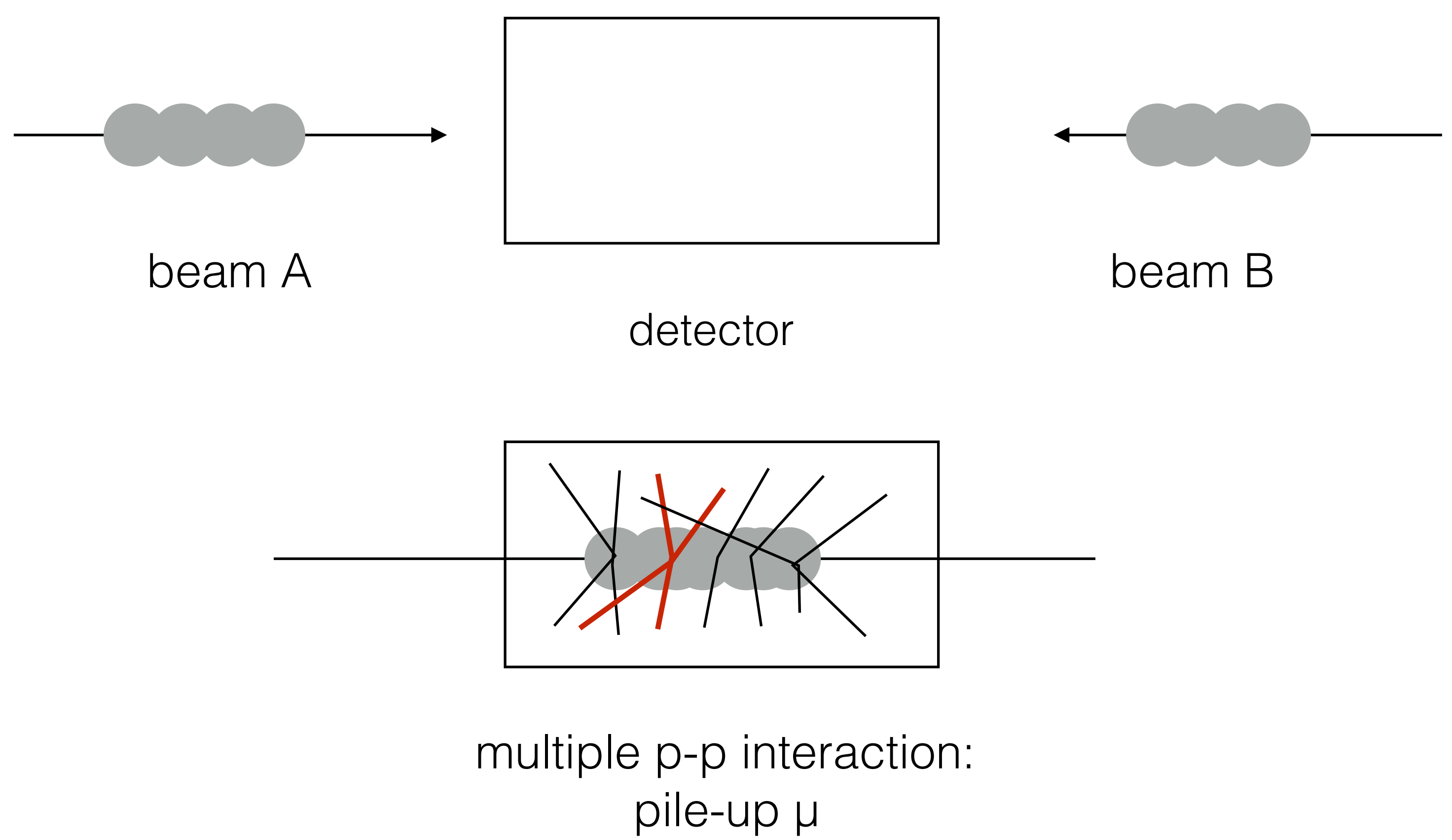
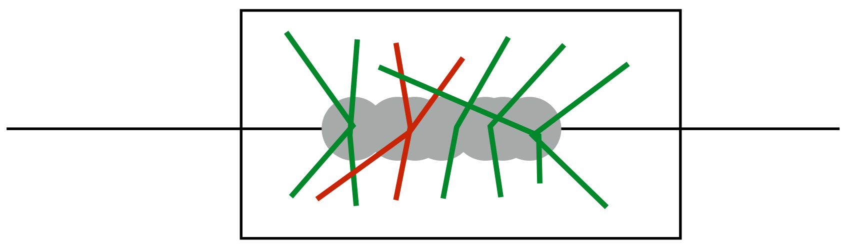


Illustration:
Simplified illustration of pile-up in a detector by colliding bunches of protons.

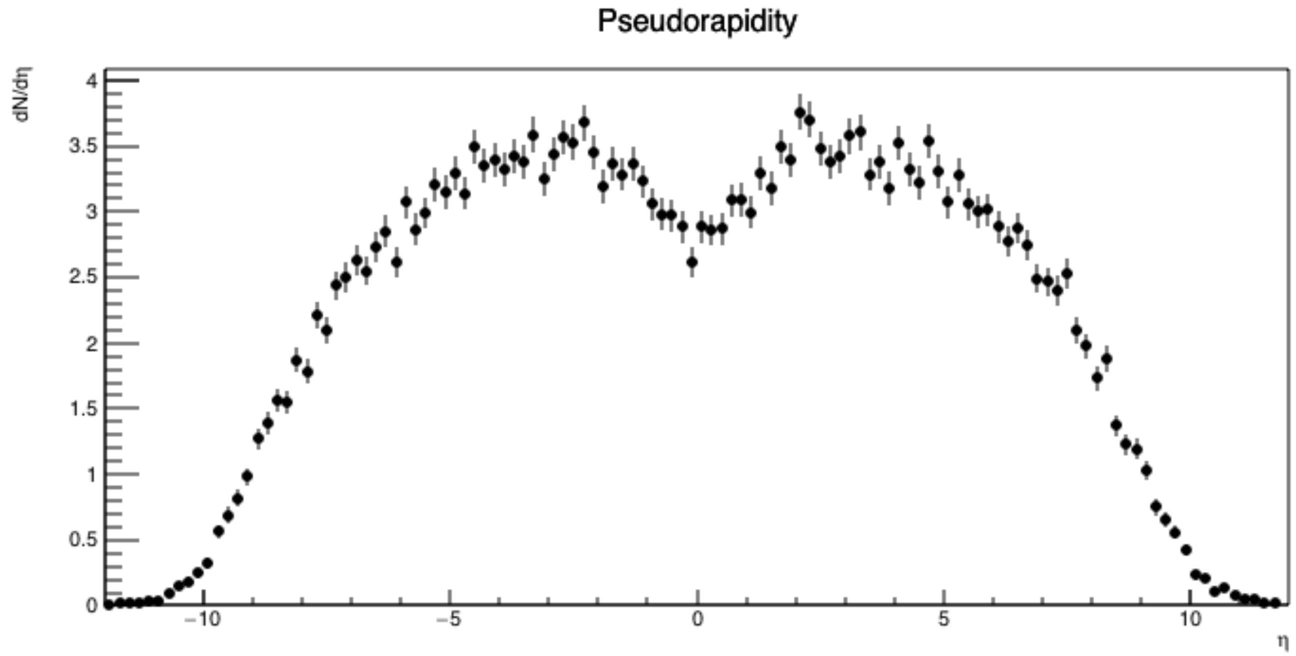
Introduction Physics

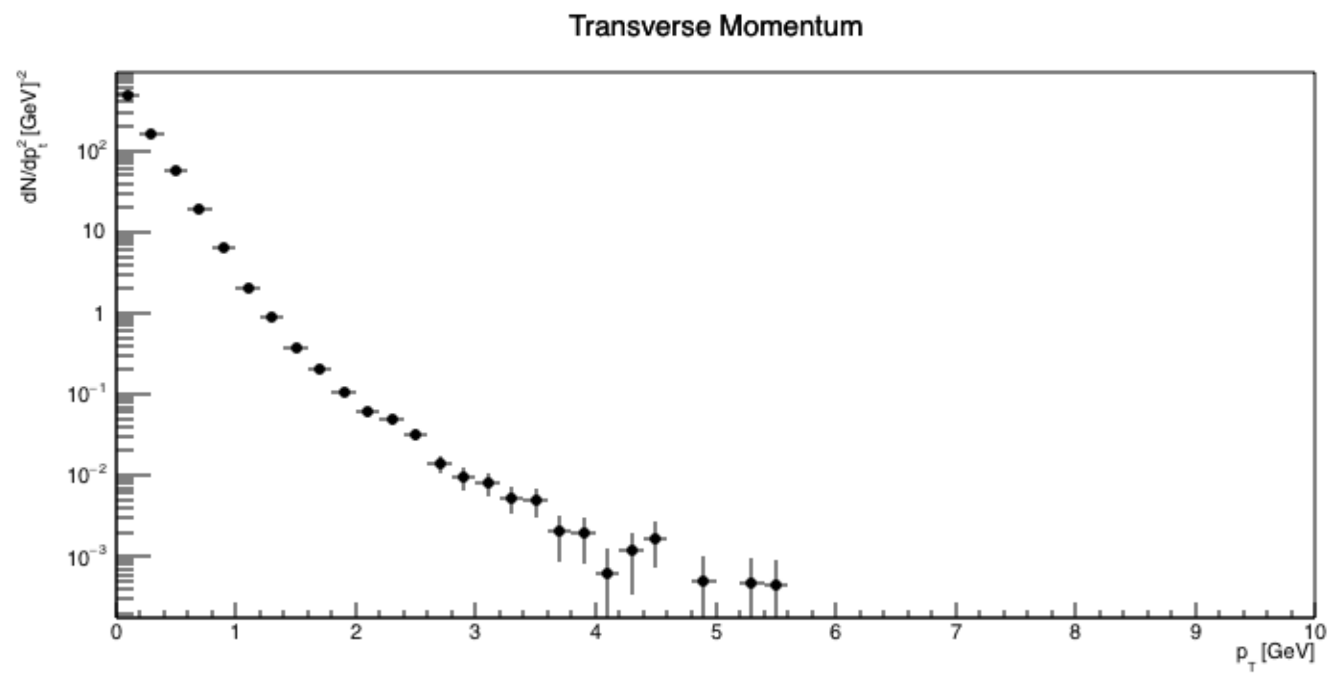


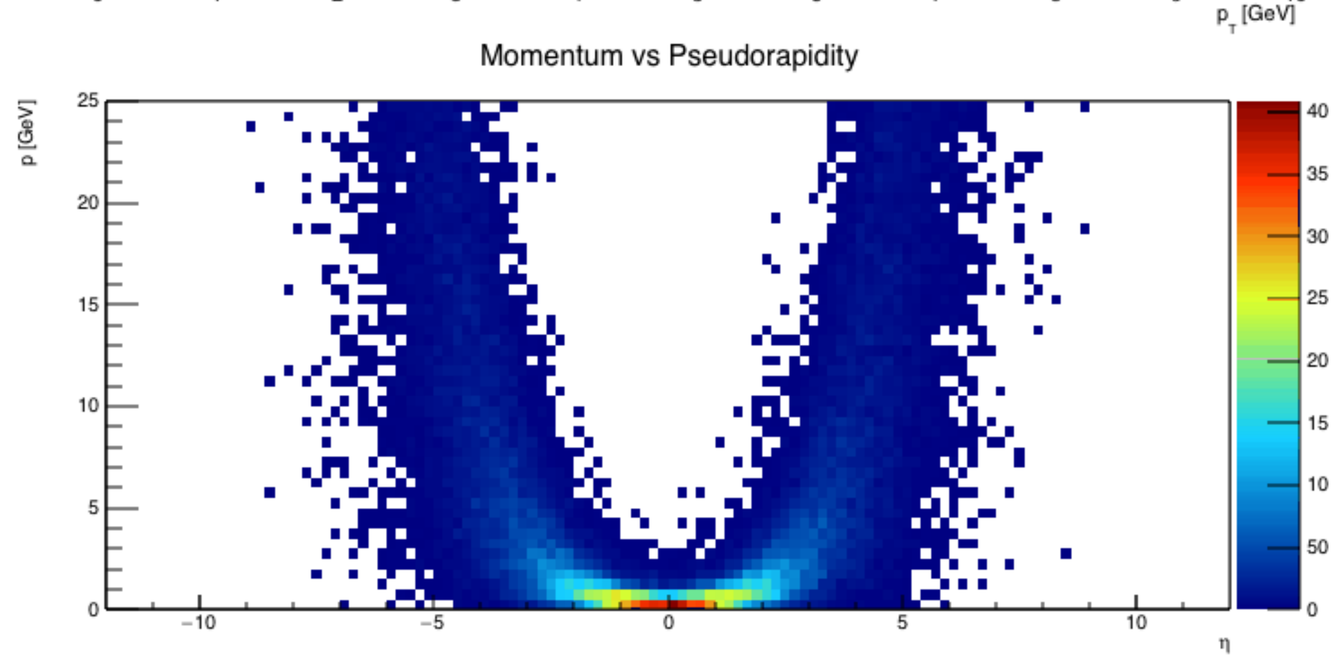
Vast majority of interaction are "uninteresting"
- yet to understand the event we spend a lot of time in understanding them

Majority of particles have low/mid momentum
- has consequences on the interaction with detector

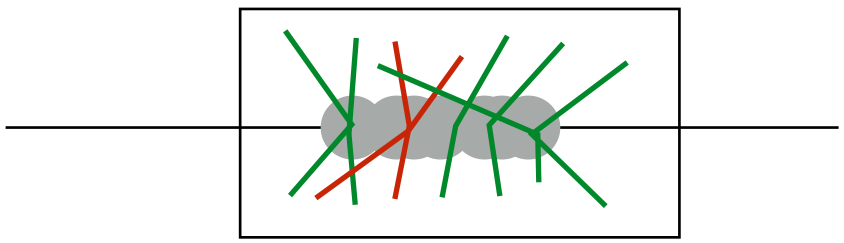
Plots:
Kinematic parameters of charged particles from p-p interactions as simulated with the PYTHIA8 generator.







Introduction Physics

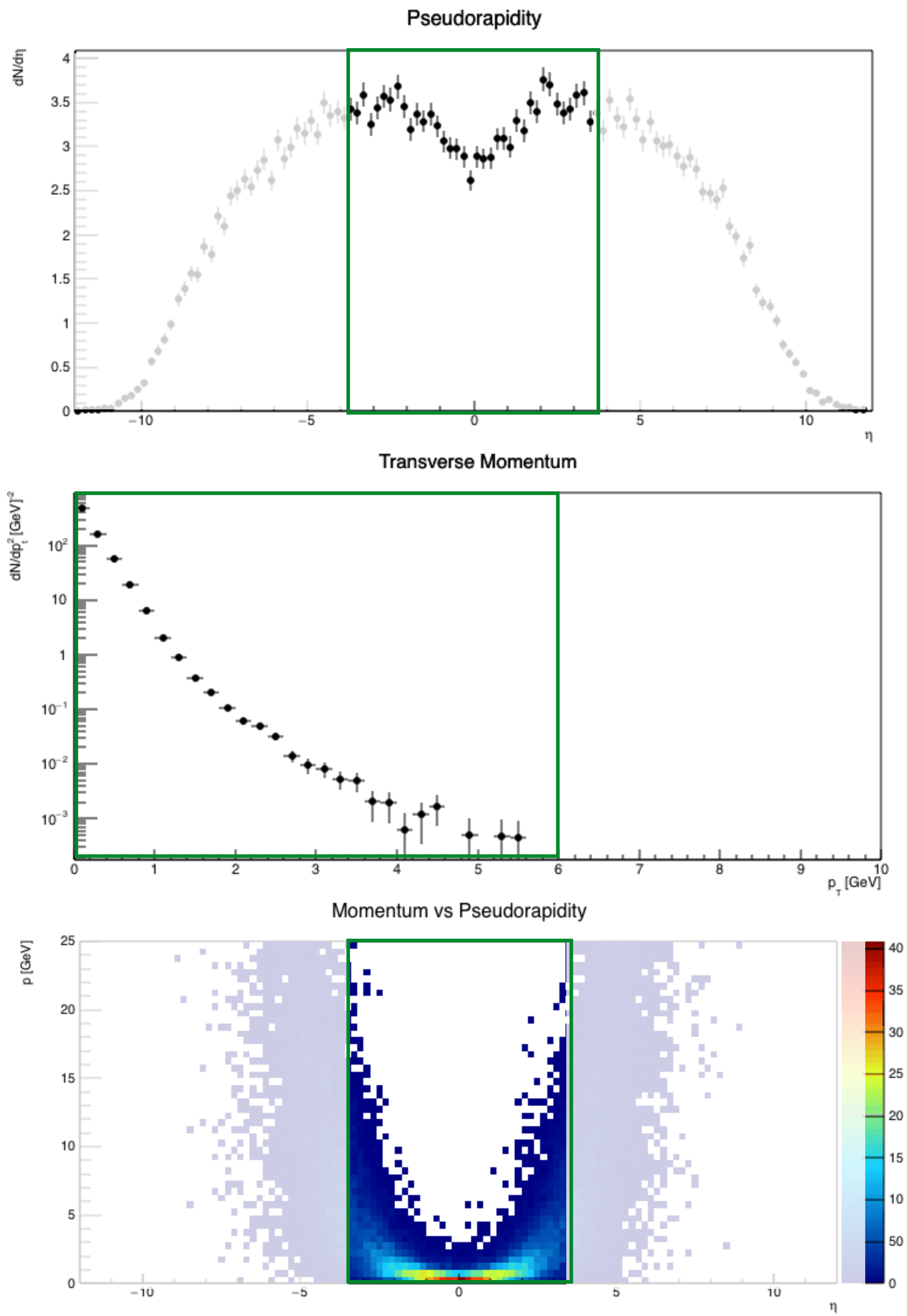


Vast majority of interaction are "uninteresting"

- yet to understand the event we spend a lot of time in understanding them

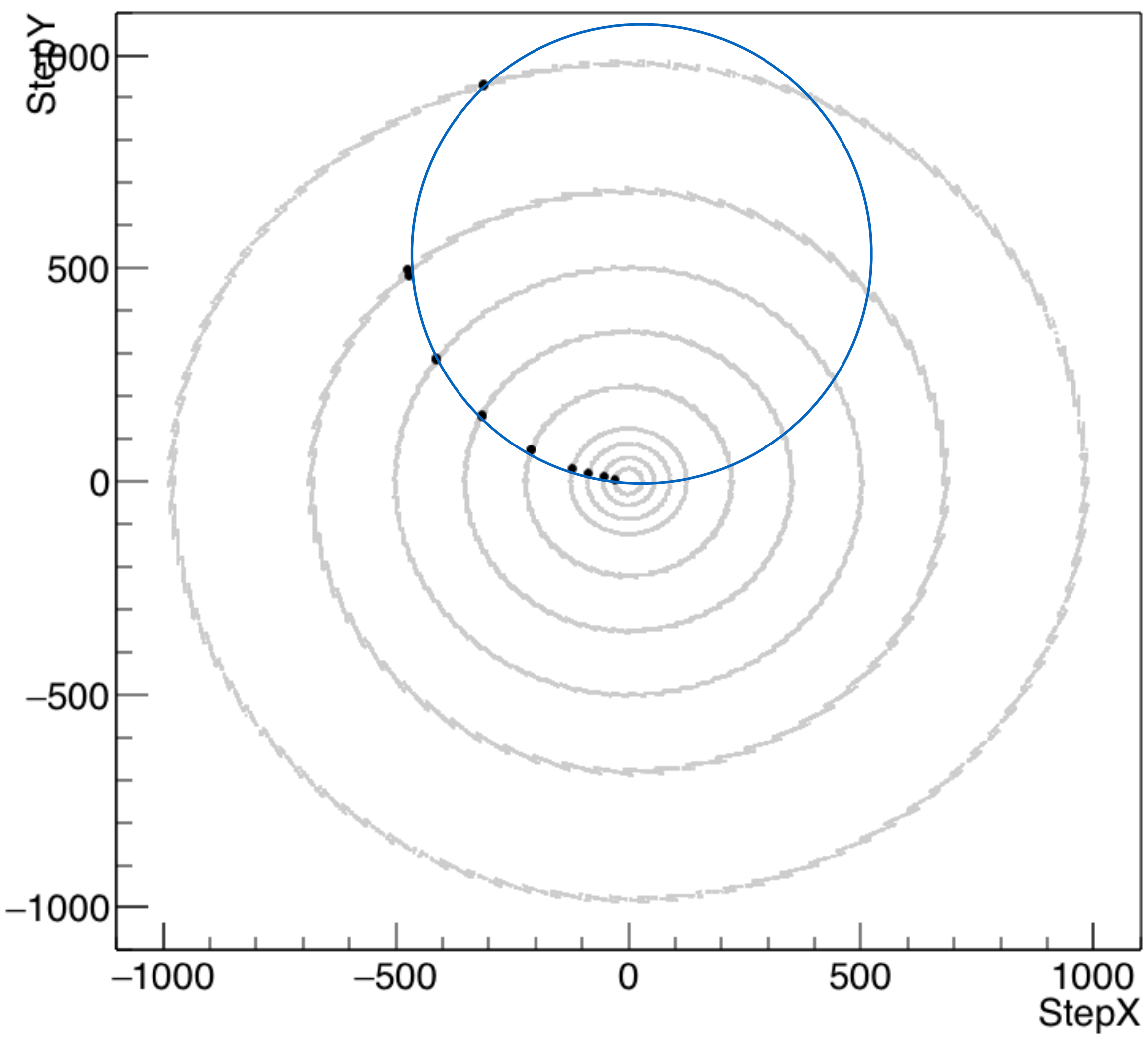
Majority of particles have low/mid momentum

- has consequences on the interaction with detector



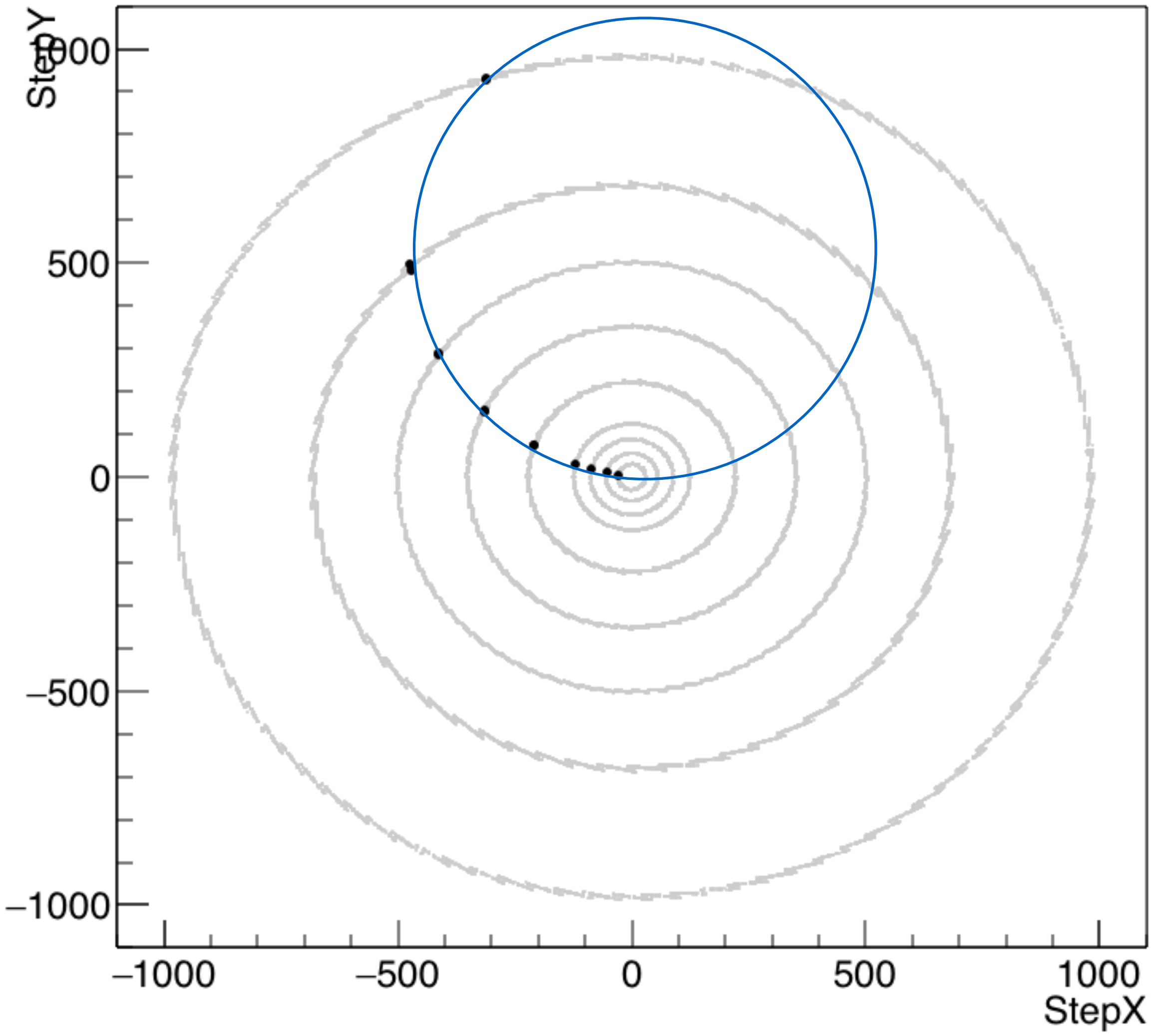
Plots:
Kinematic parameters of charged particles from p-p interactions as simulated with the PYTHIA8 generator.

Introduction Charged particles

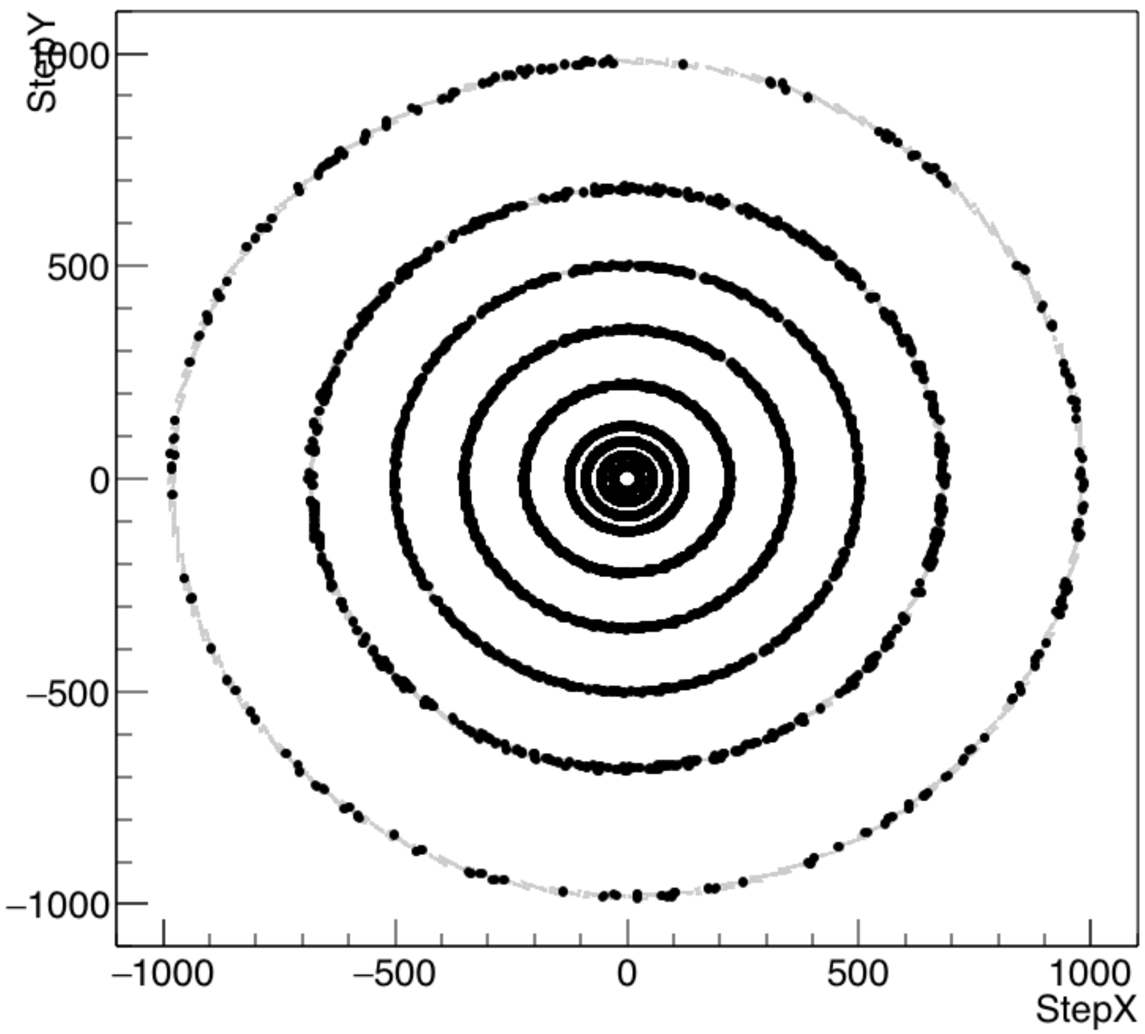


hits from 1 particle

Introduction Charged particles



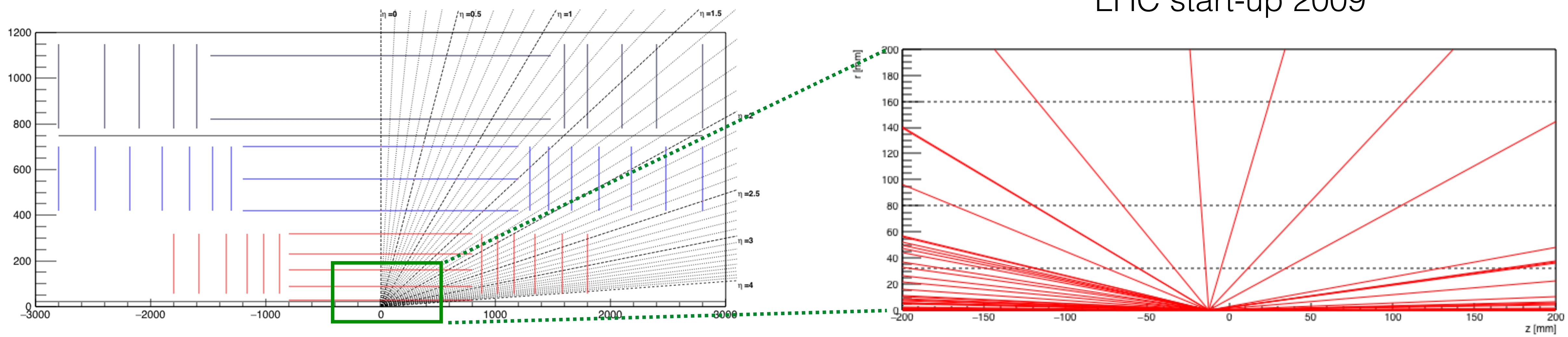
hits from 1 particle

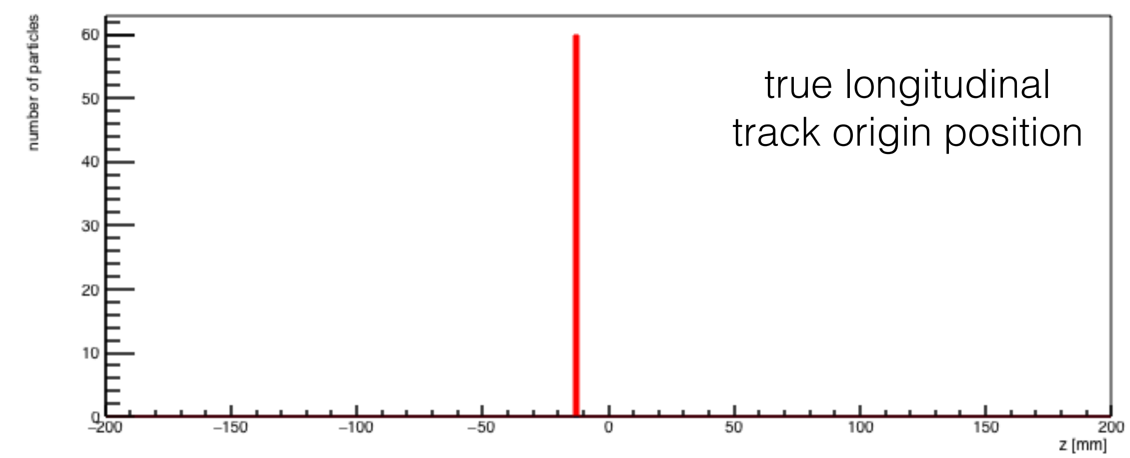


fraction of hits
from particles
in 200 pile-up events

Investigation Pile-up

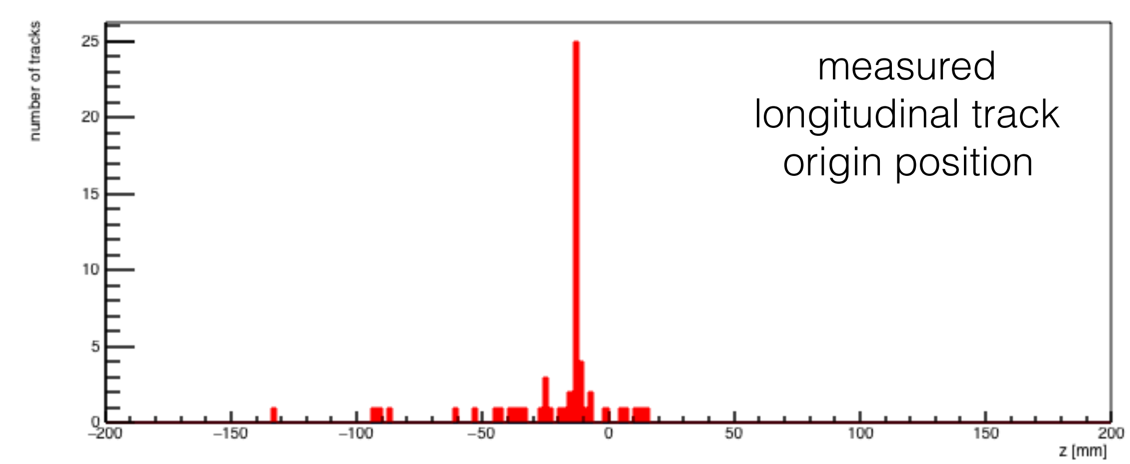
1 p-p collision
LHC start-up 2009

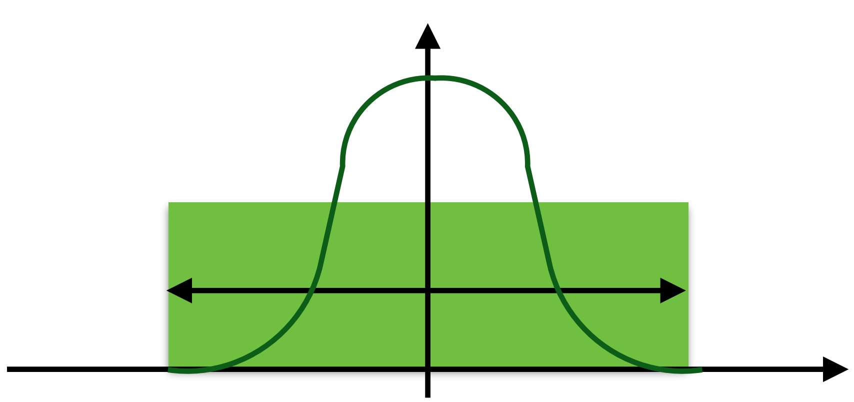




Event pile-up has follows a beam profile around the interaction region

- currently gaussian with σ ~ 55mm
- future possible flat profile

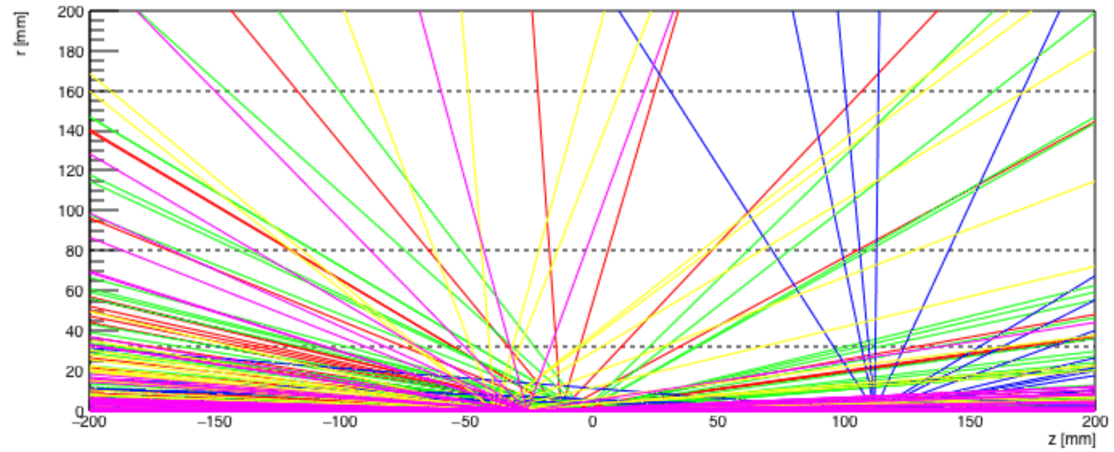


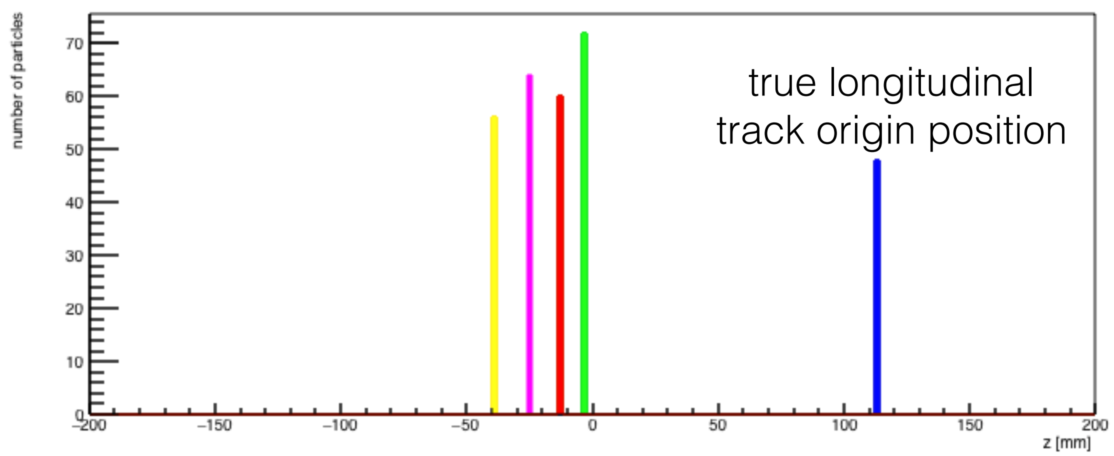


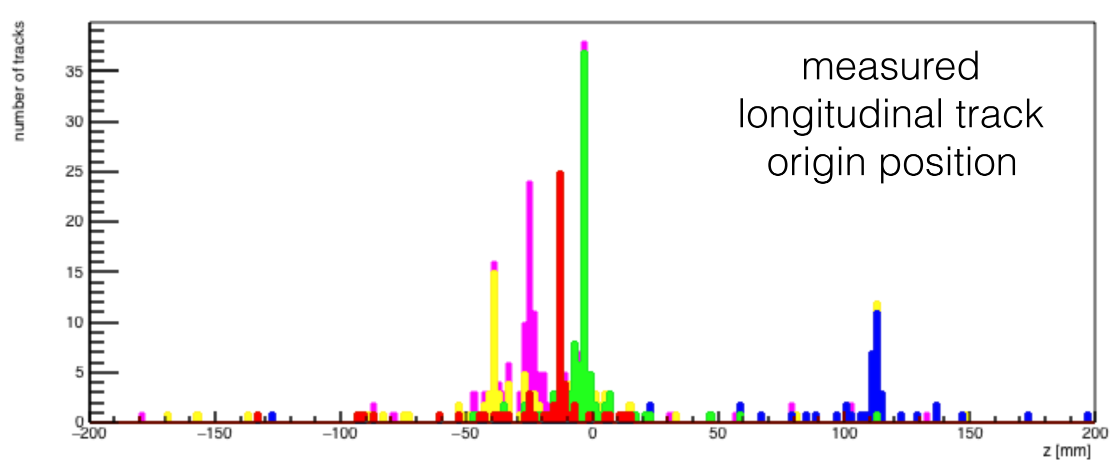
Figures:
Tracking detector (left top), longitudinal view of tracks emerging from one vertex (right top), true and measured longitudinal track origin (right), illustration of beam profile (left bottom),

Investigation Pile-up

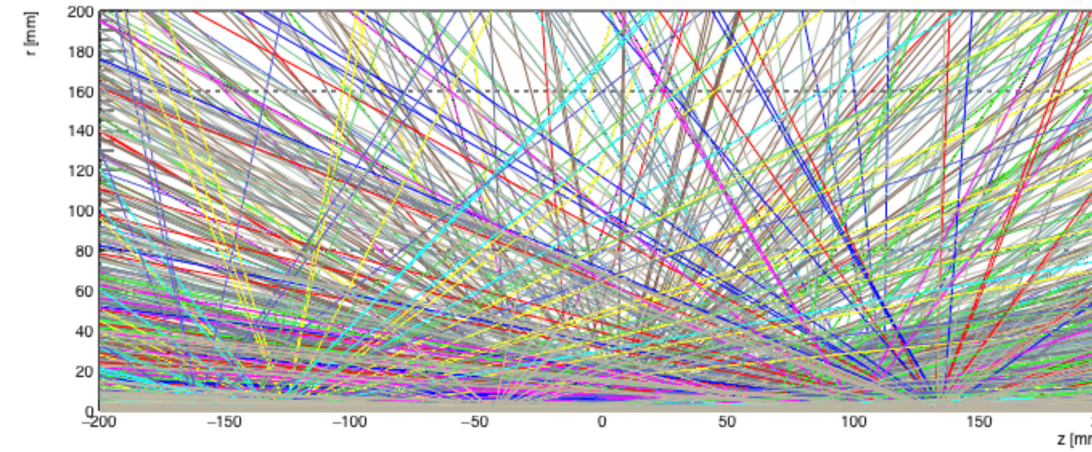
5 p-p collisions
LHC early Run-1 2010

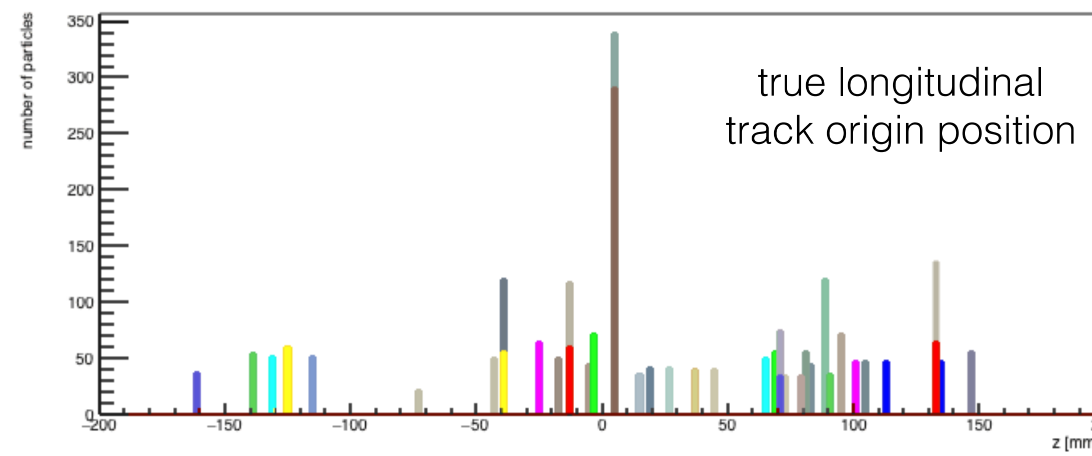


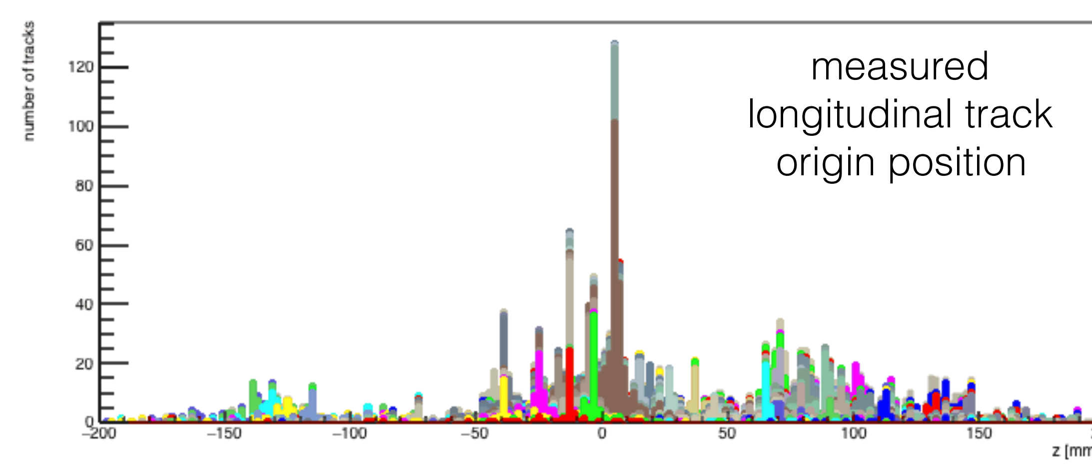




40 p-p collisions
LHC early Run-2 2015/16



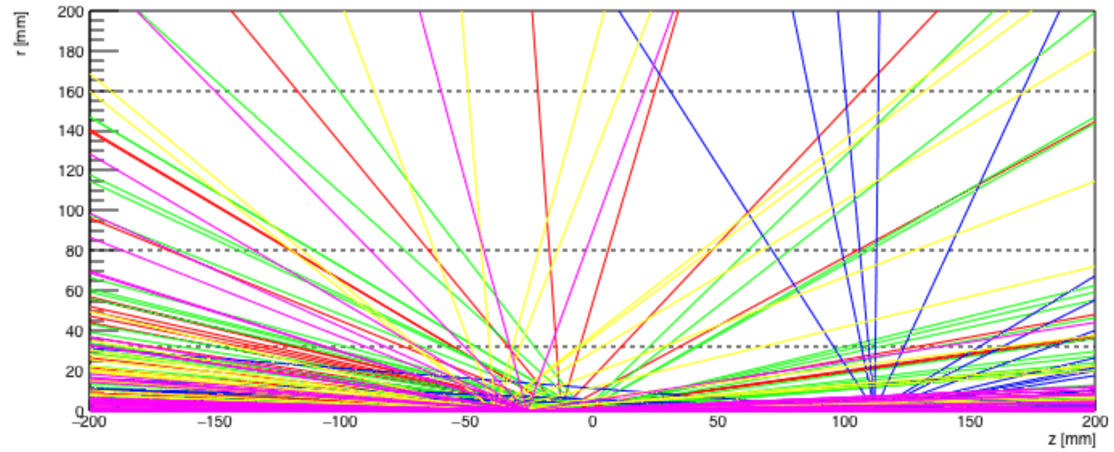


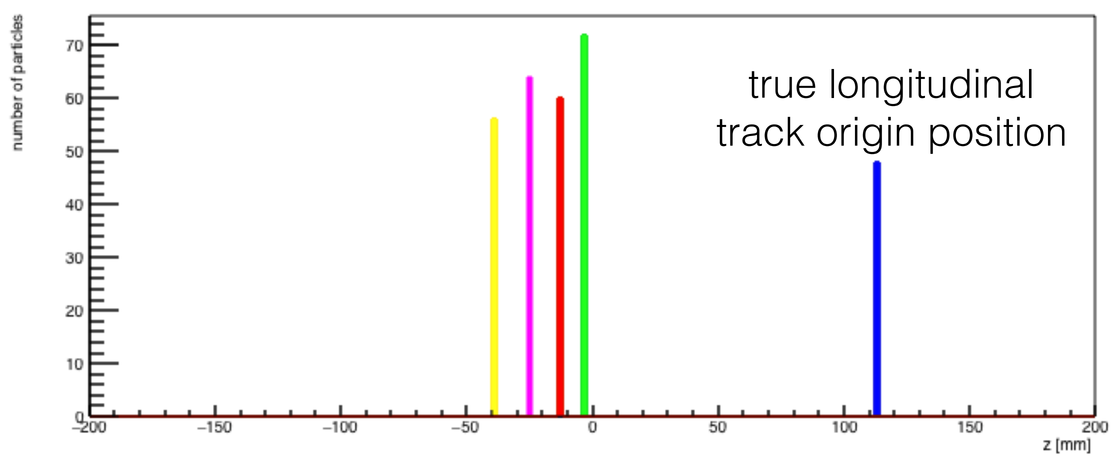


Figures:
Longitudinal view of tracks emerging from one vertex (top), true (middle) and measured longitudinal track origin (bottom), for 5 proton-proton collisions (left), and 40 proton-proton collisions (right).

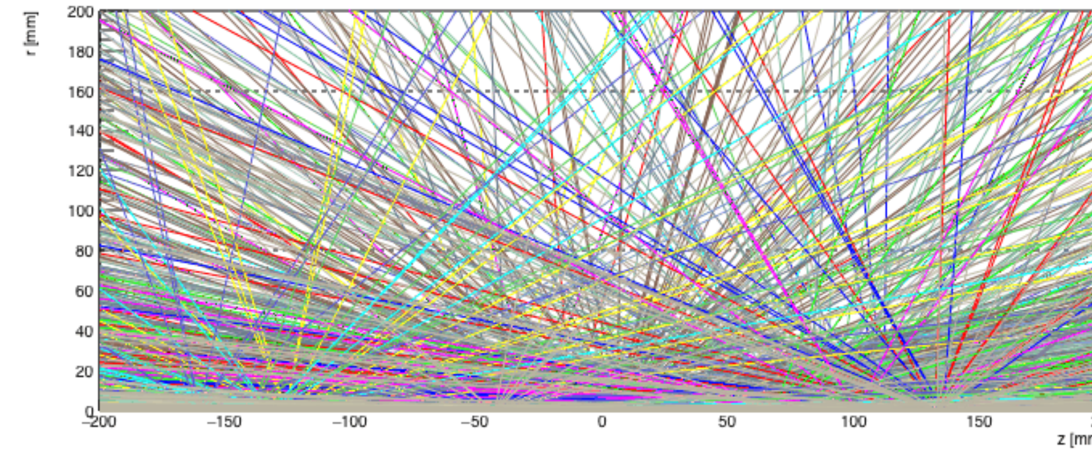
Investigation Pile-up

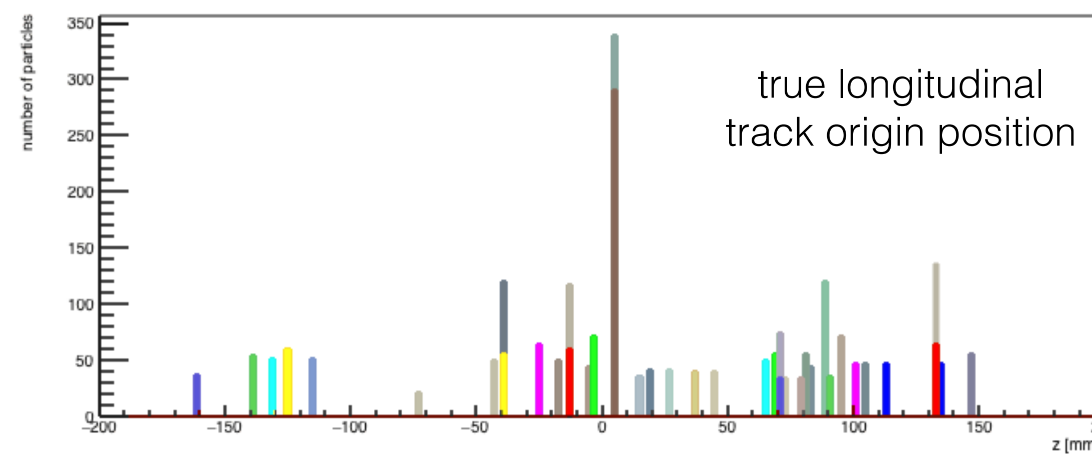
5 p-p collisions
LHC early Run-1 2010





40 p-p collisions
LHC early Run-2 2015/16





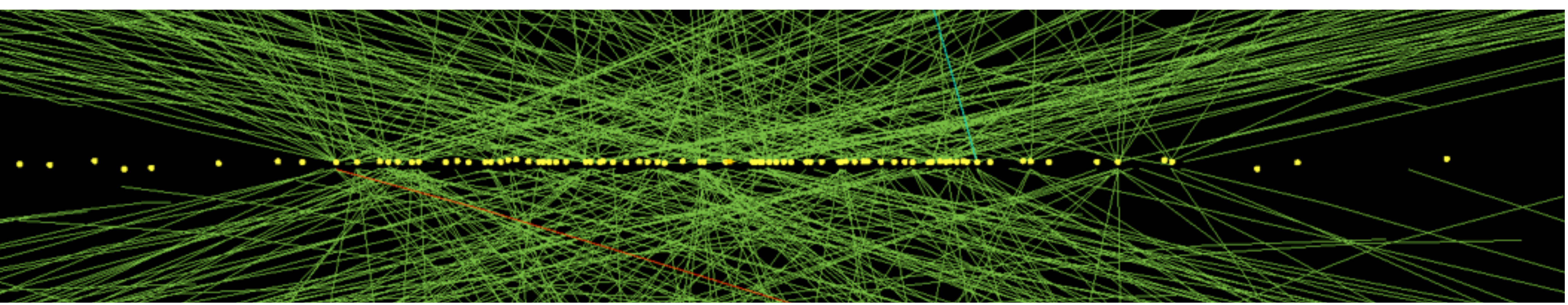
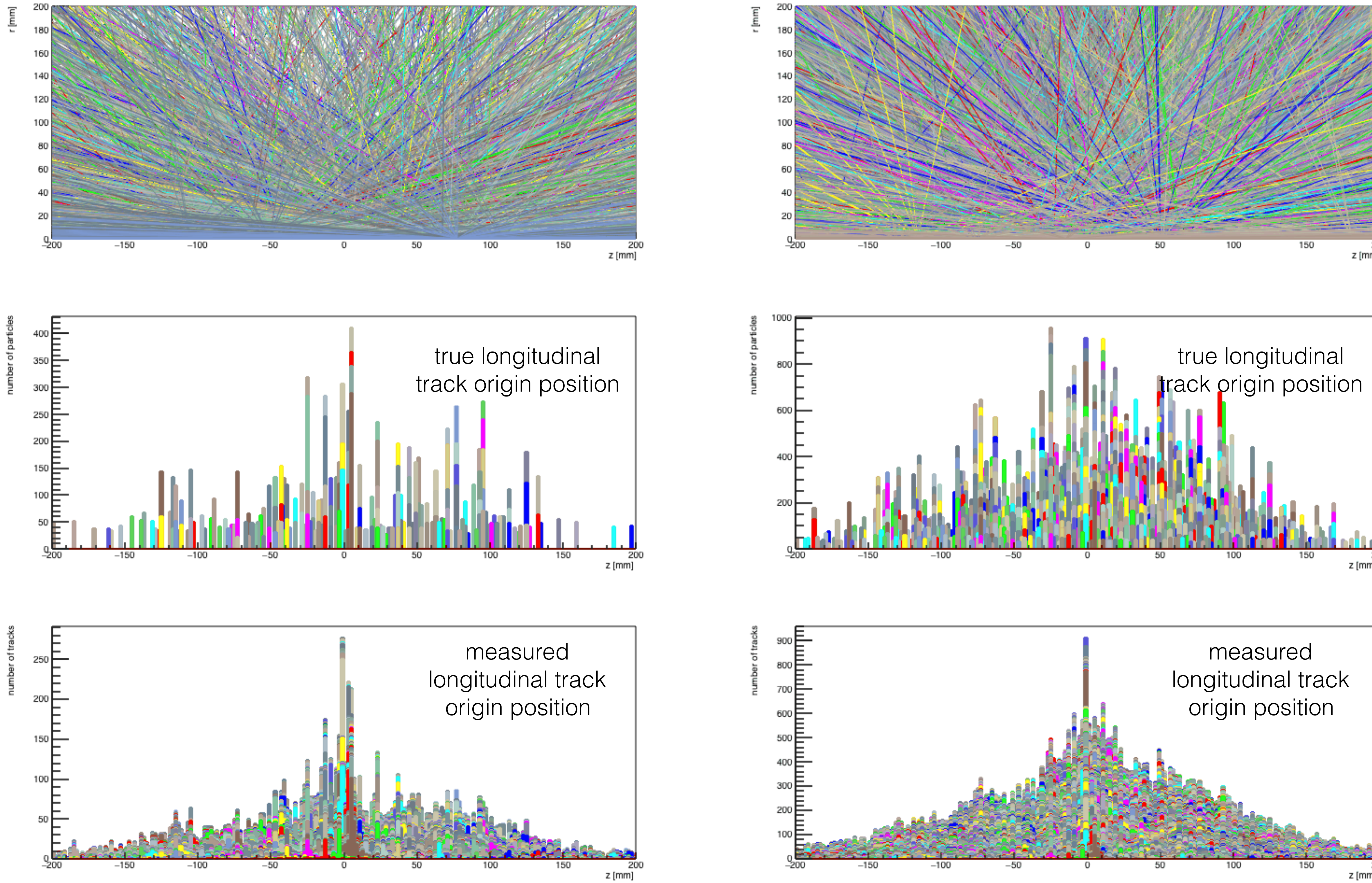


Figure:
Reconstructed vertices in a recorded collisions with the CMS experiment.

Investigation Pile-up

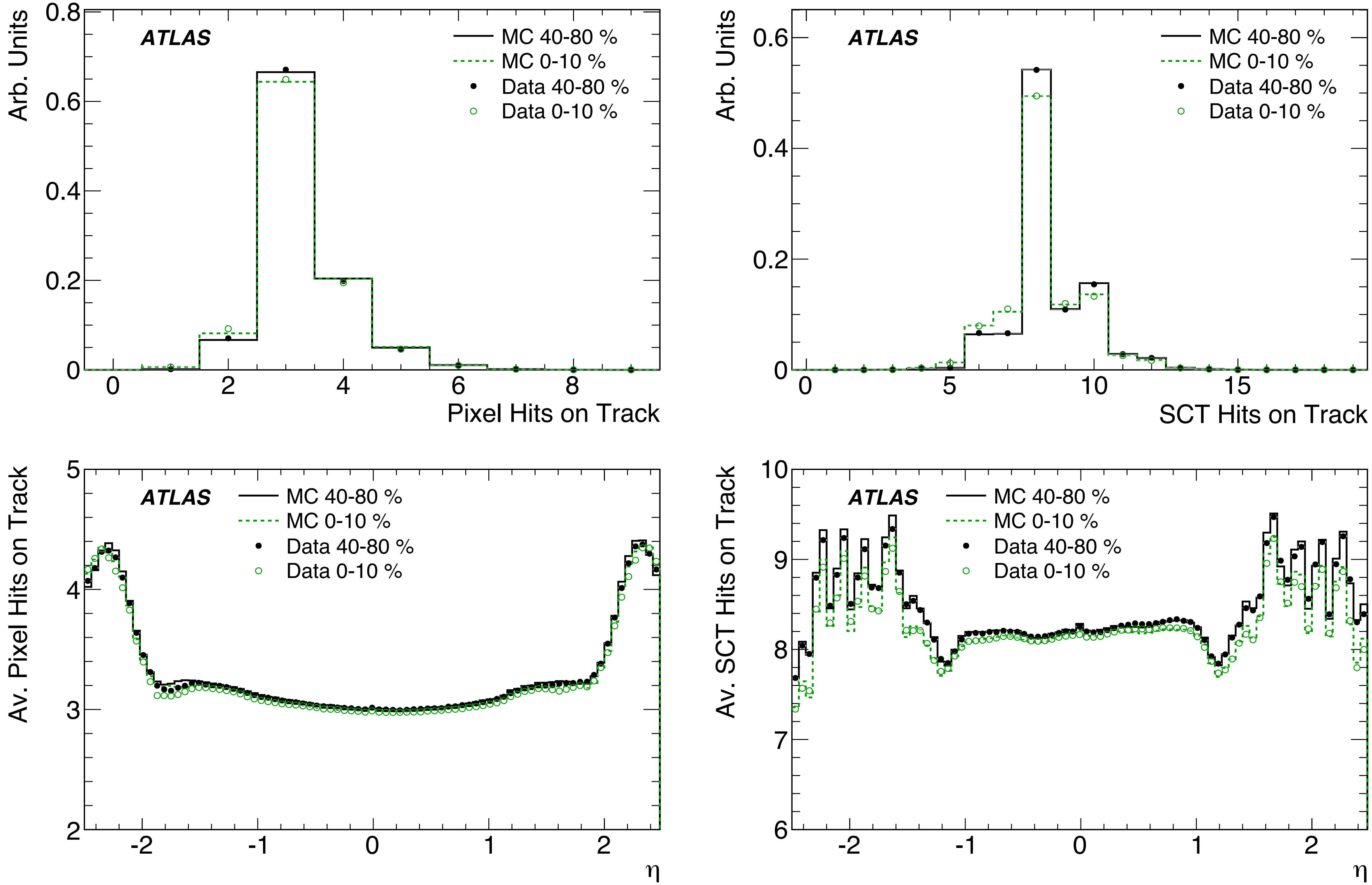
200 p-p collisions
HL-LHC conditions

1000 p-p collisions
5ns scenario for FCC-hh



Figures:
Longitudinal view of tracks emerging from one vertex (top), true (middle) and measured longitudinal track origin (bottom), for 200 proton-proton collisions (left), and 1000 proton-proton collisions (right).

Track reconstruction at LHC with high particle multiplicities

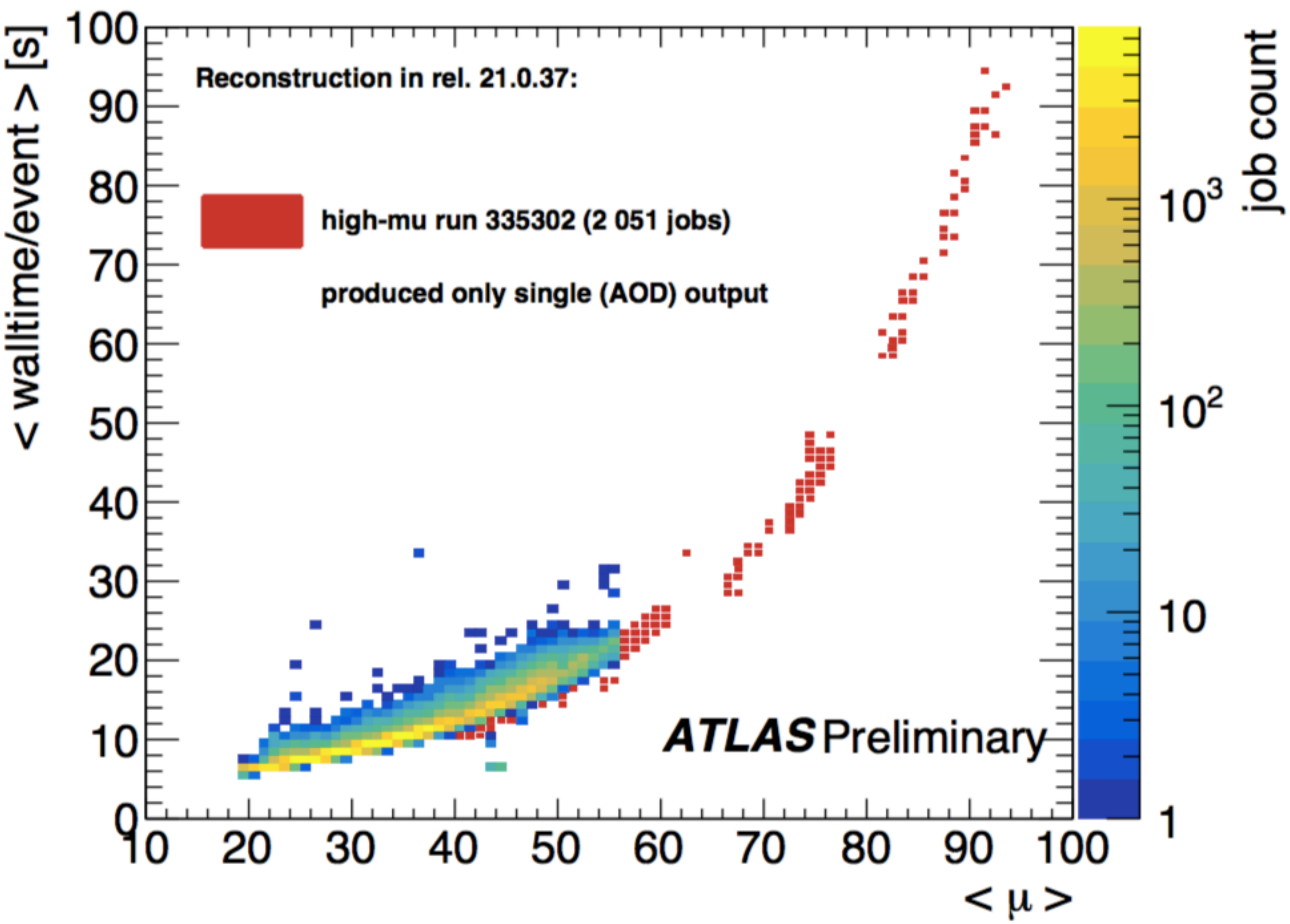


Figures:
Hits on track statistics in the Pixel and strip (SCT) detector of ATLAS, comparing low multiplicity and high multiplicity event classes.

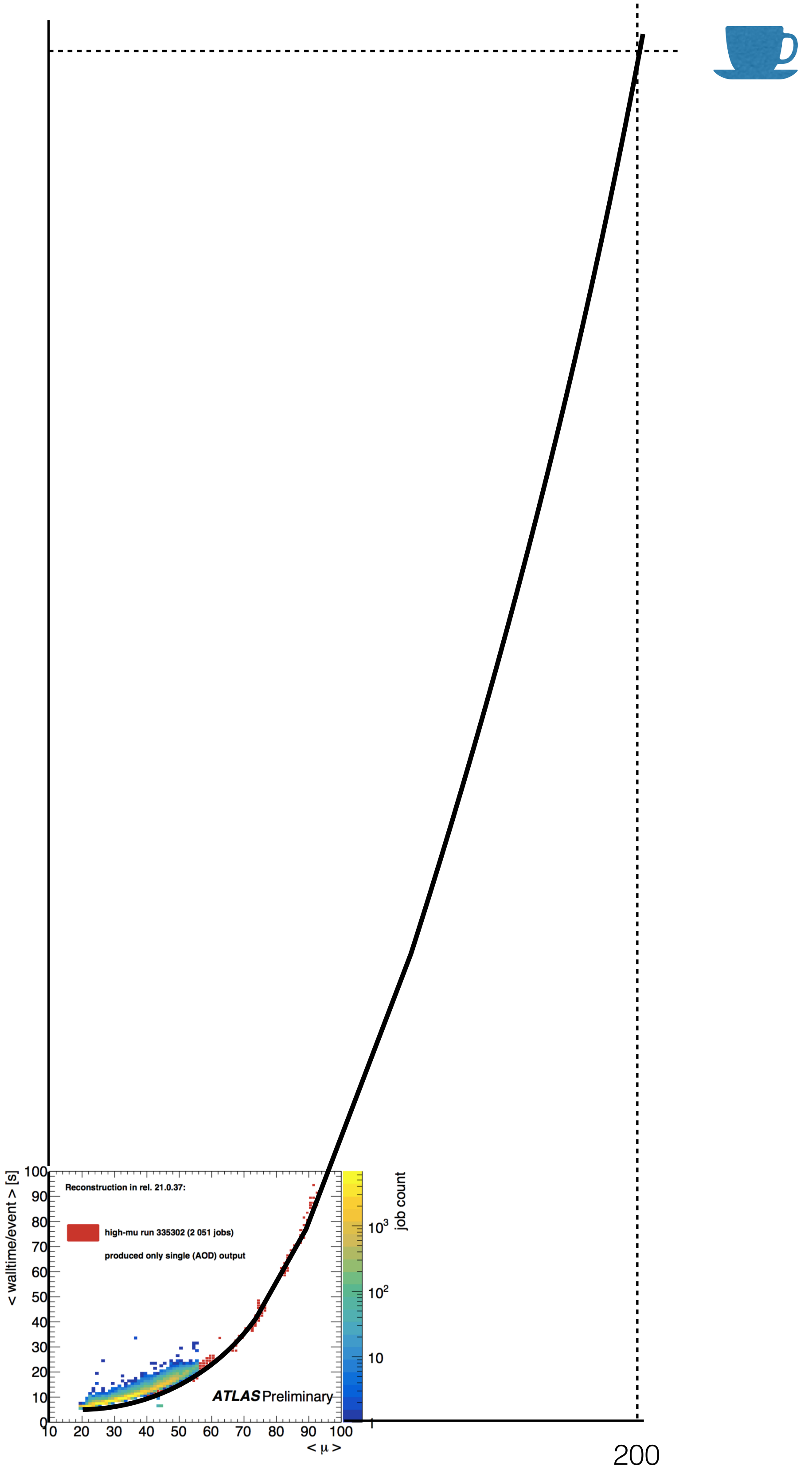
Track reconstruction CPU consumption

Track reconstruction is a combinatorial problem

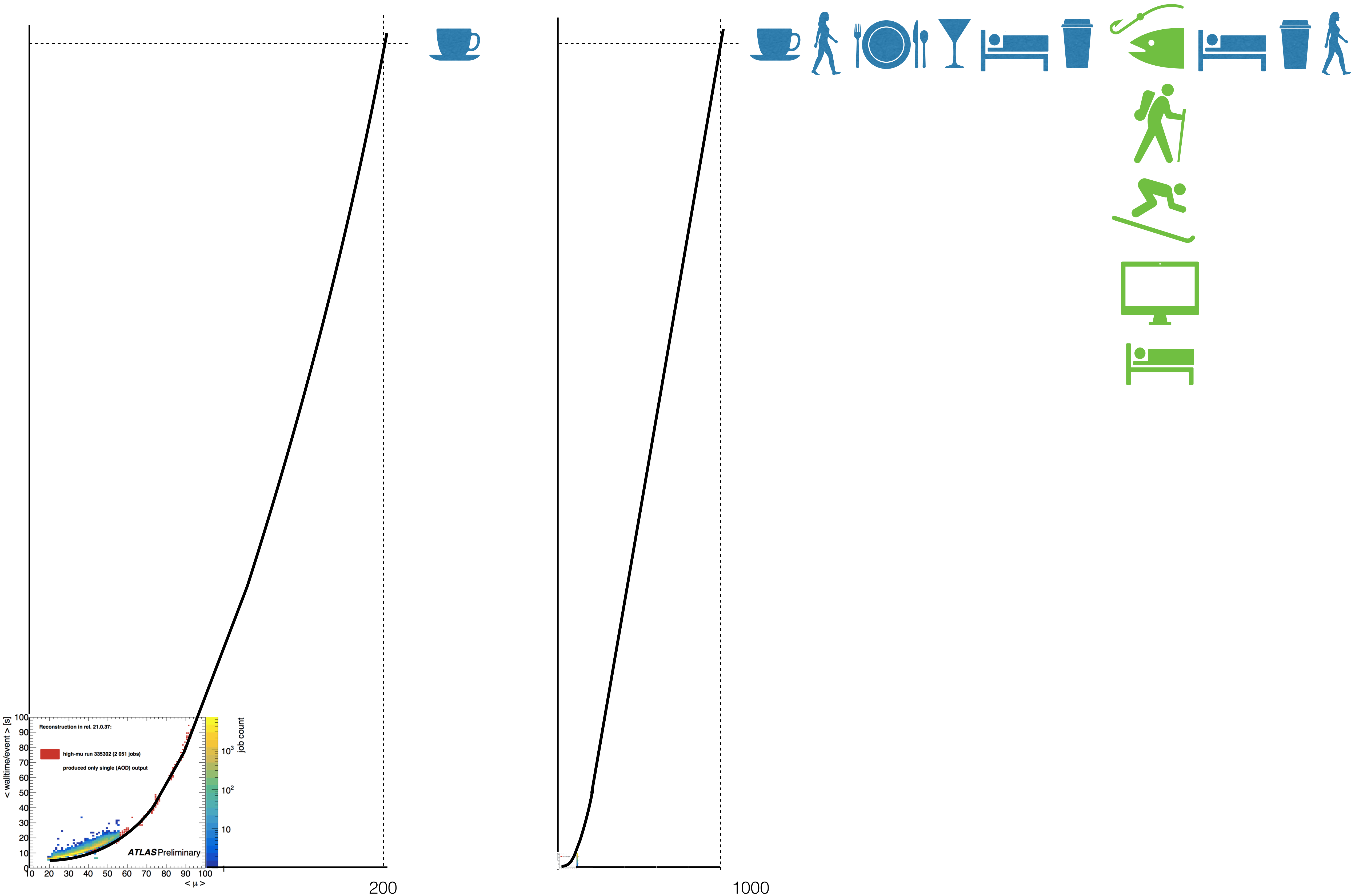
- naturally dominates the reconstruction time of hadron colliders
- LHC experiments have massively invested into code/SW optimistion



Track reconstruction Extrapolation HL-LHC



Track reconstruction Extrapolation HL-LHC / FCC-hh



Track finding Basics

Starting from measurements (with given accuracy) on detector layers

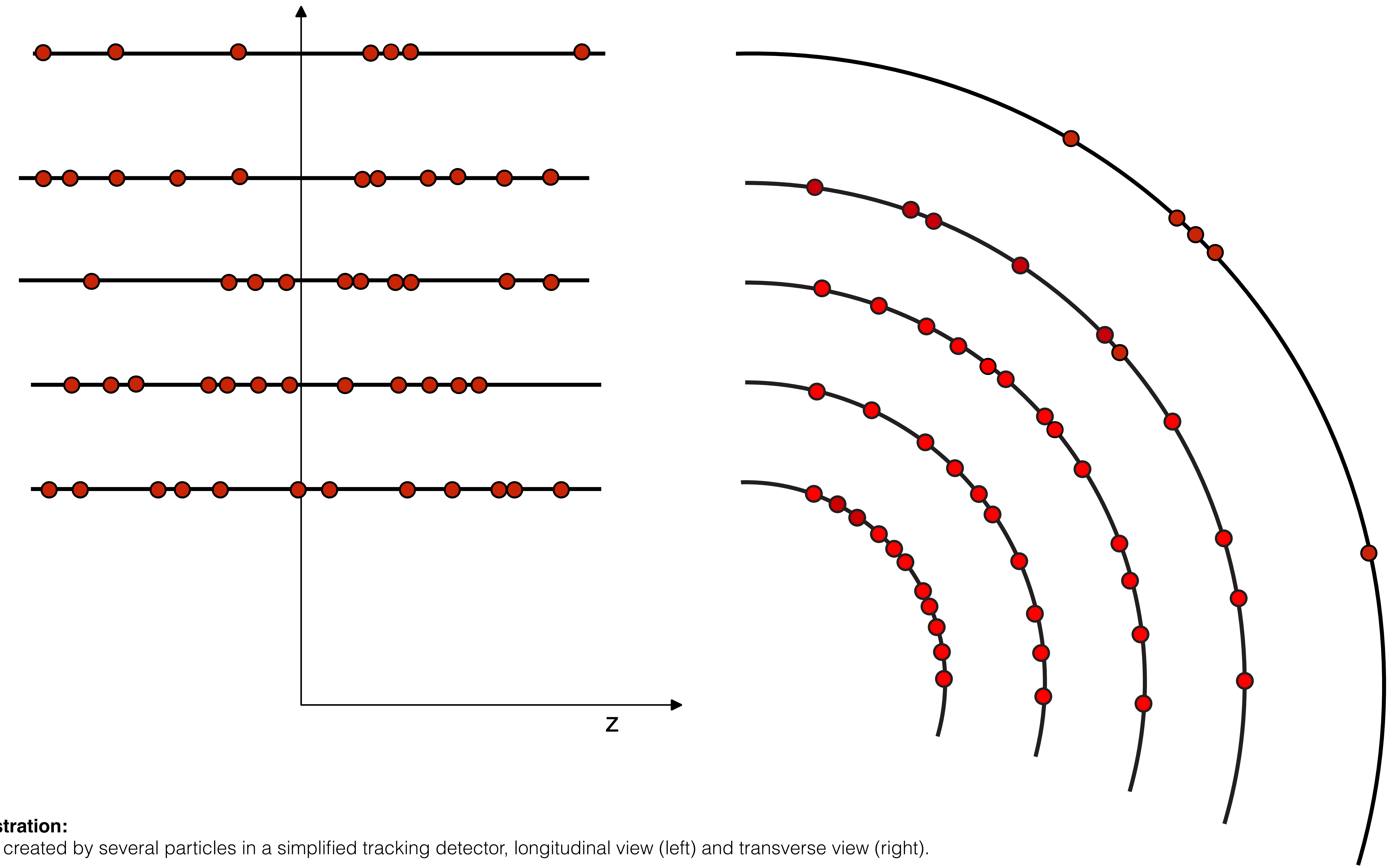


Illustration:
Hits created by several particles in a simplified tracking detector, longitudinal view (left) and transverse view (right).

Track finding pattern recognition methods

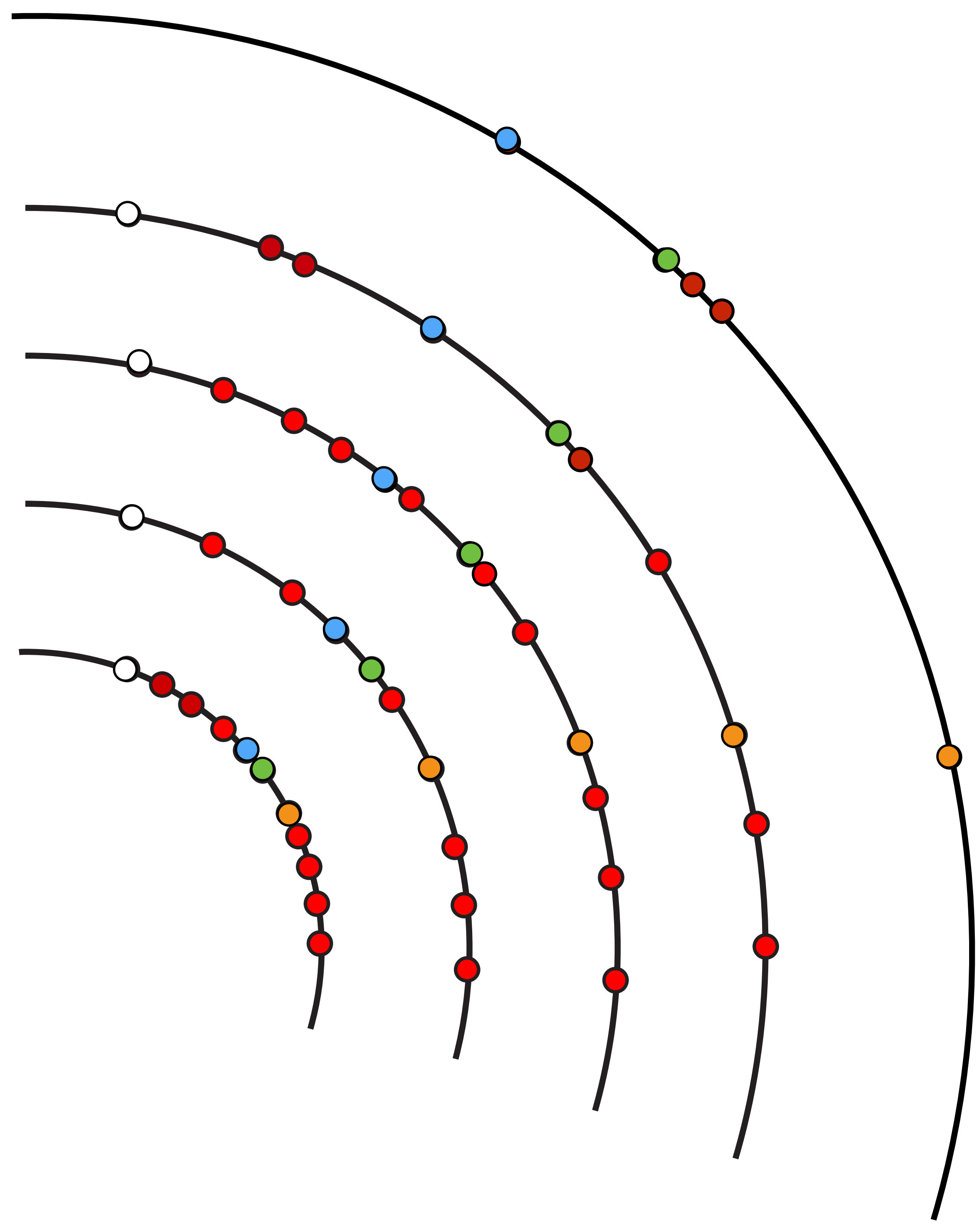


Illustration:
Hits created by several particles in a simplified tracking detector, transverse view.

Track finding a clustering problem

This is a classical clustering/unsupervised learning problem

- find sets of hits that *belong* together

what defines
this relationship ?

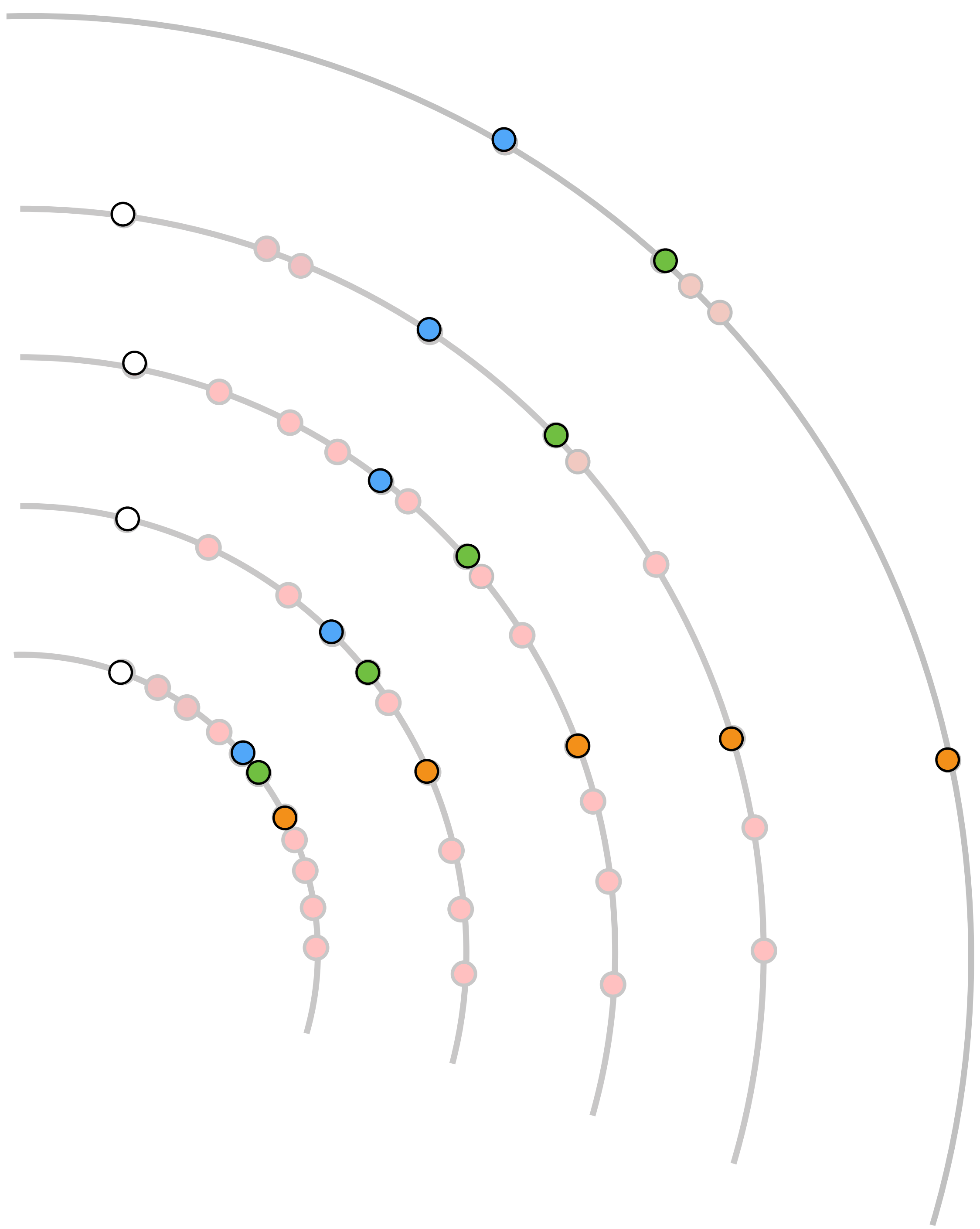


Illustration:
Hits created by several particles in a simplified tracking detector, transverse view.

A historical view (of a part of London)

Illustration:
Parts of the map of the 1854 cholera outbreak in London's Soho district by Dr. John Snow.

ML clustering

a classical clustering problem

- find sets of hits that *belong* together

What defines a *belonging* relation ?

- simple solution here:
  "clustering by euclidian distance"
  (distance measure)

Example: k-means algorithm

$$\underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2 = \underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} |S_i| \operatorname{Var} S_i$$

▲ … infected water pump (highly correlated with cluster centers)

Illustration:
Detail of the map of the 1854 cholera outbreak in London's Soho district by Dr. John Snow.

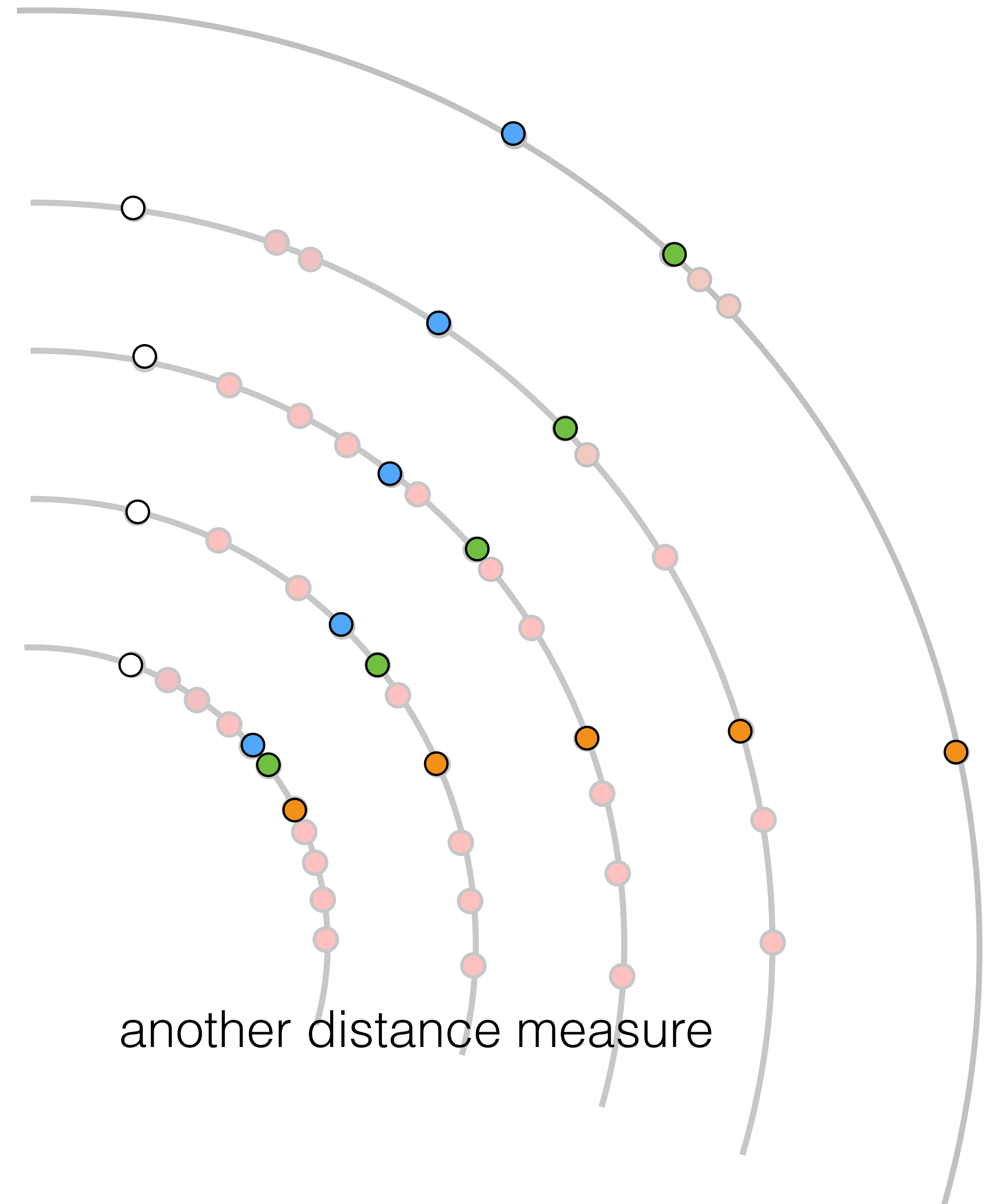
Track finding a clustering problem

This is a classical clustering problem

- find sets of hits that *belong* together

need some
domain knowledge

euclidian distance measure



another distance measure

https://fusiontables.google.com/DataSource?docid=1HsIb_r4gYYmIz8y_UE1h-X8yUtAYW2INy99BR_c#map:id=3

Global pattern recognition Conformal mapping

Conformal mapping : Hough transform

- transform your track hits from the x, y space into a more appropriate space
- let's assume that particles come from the interaction region + solve in the transverse direction

$$\mathbf{q} = (d_0, z_0, \phi, \theta, q/p)$$

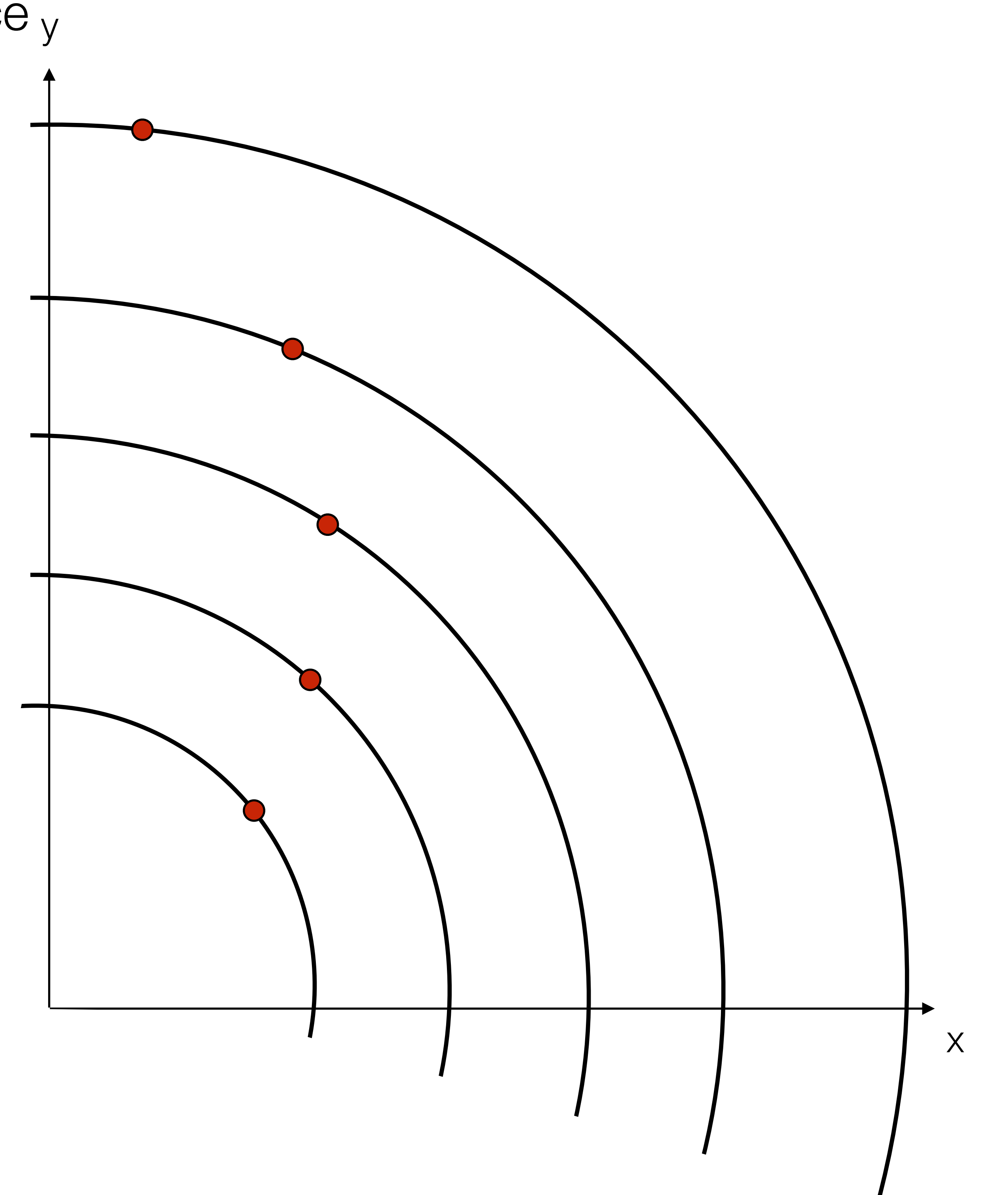


Illustration:
Hits created by one particles in a simplified tracking detector, transverse view.

Global pattern recognition Conformal mapping

Conformal mapping : <u>Hough transform</u>

- transform your track hits from the x, y space into a more appropriate space
- let’s assume that particles come from the interaction region + solve in the transverse direction

$$\mathbf{q} = (d_0, z_0, \phi, \theta, q/p)$$

$$\downarrow$$

$$\mathbf{q} = (\cancel{d_0}, \cancel{z_0}, \phi, \cancel{\theta}, q/p_T)$$

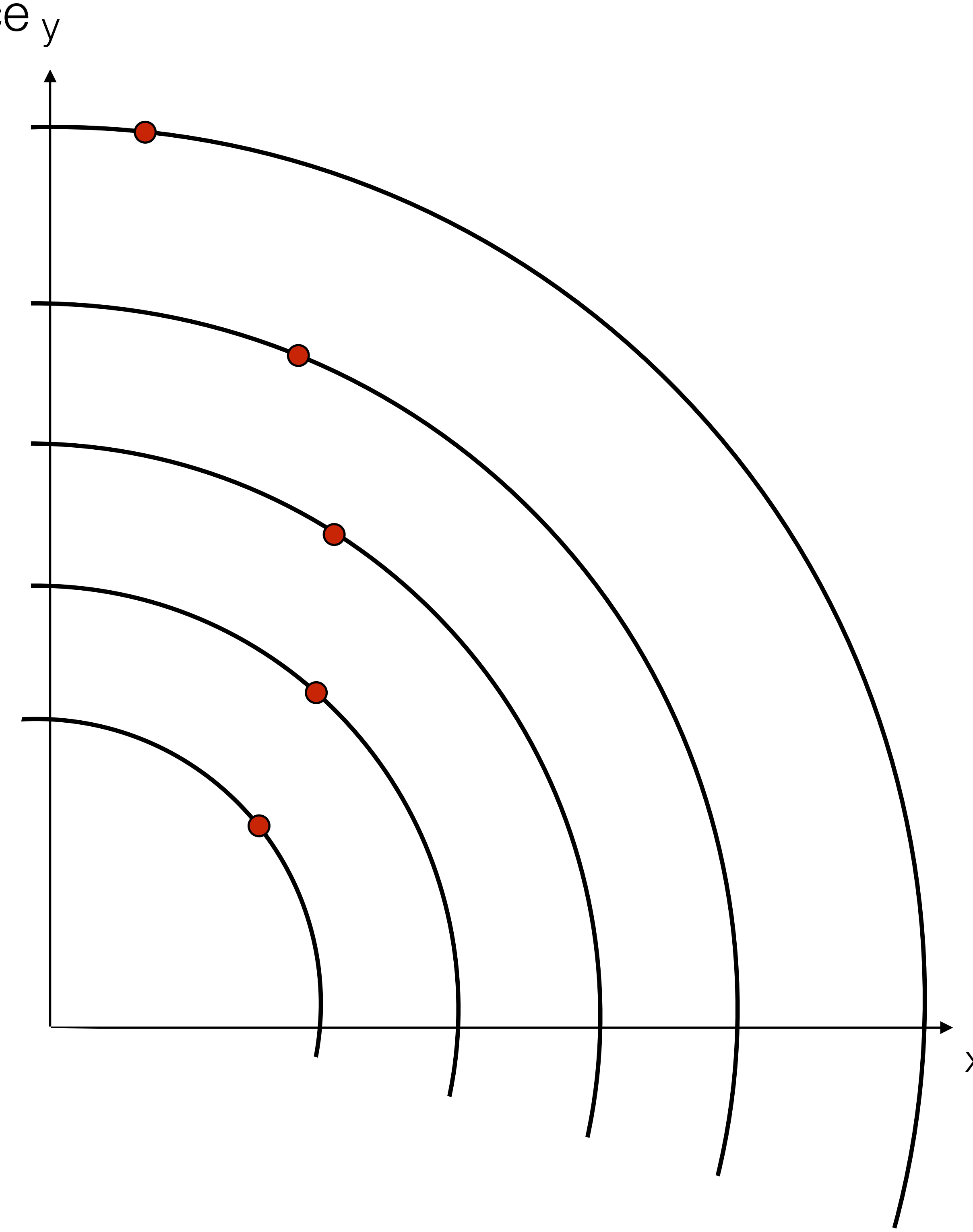
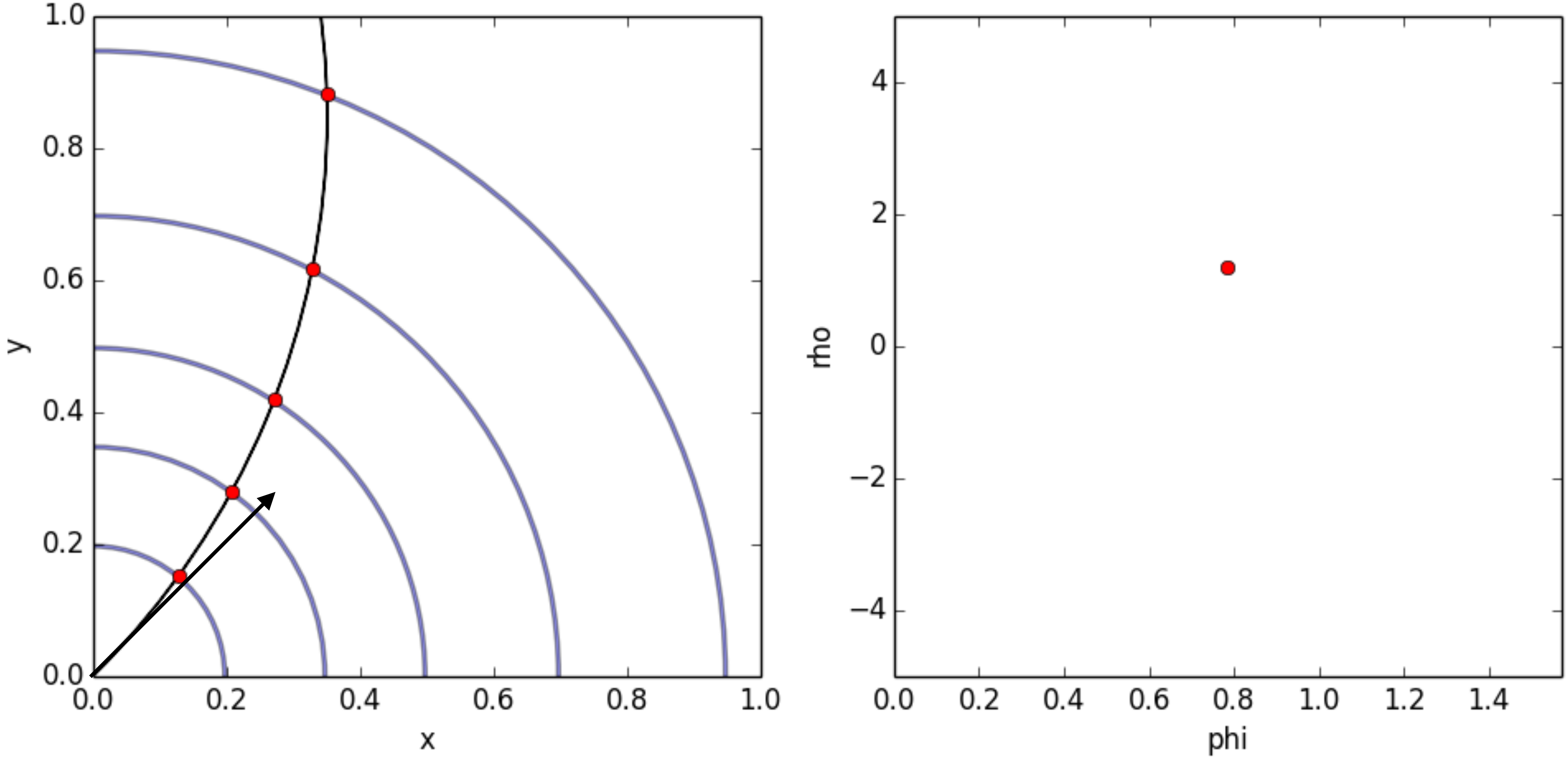


Illustration:
Hits created by one particles in a simplified tracking detector, transverse view.

34

Conformal mapping techniques Toy example



phi … emission angle in transverse direction at origin
rho … radius of the helix in transverse plane

Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: perfect solution in the hough space.

Conformal mapping techniques Toy example

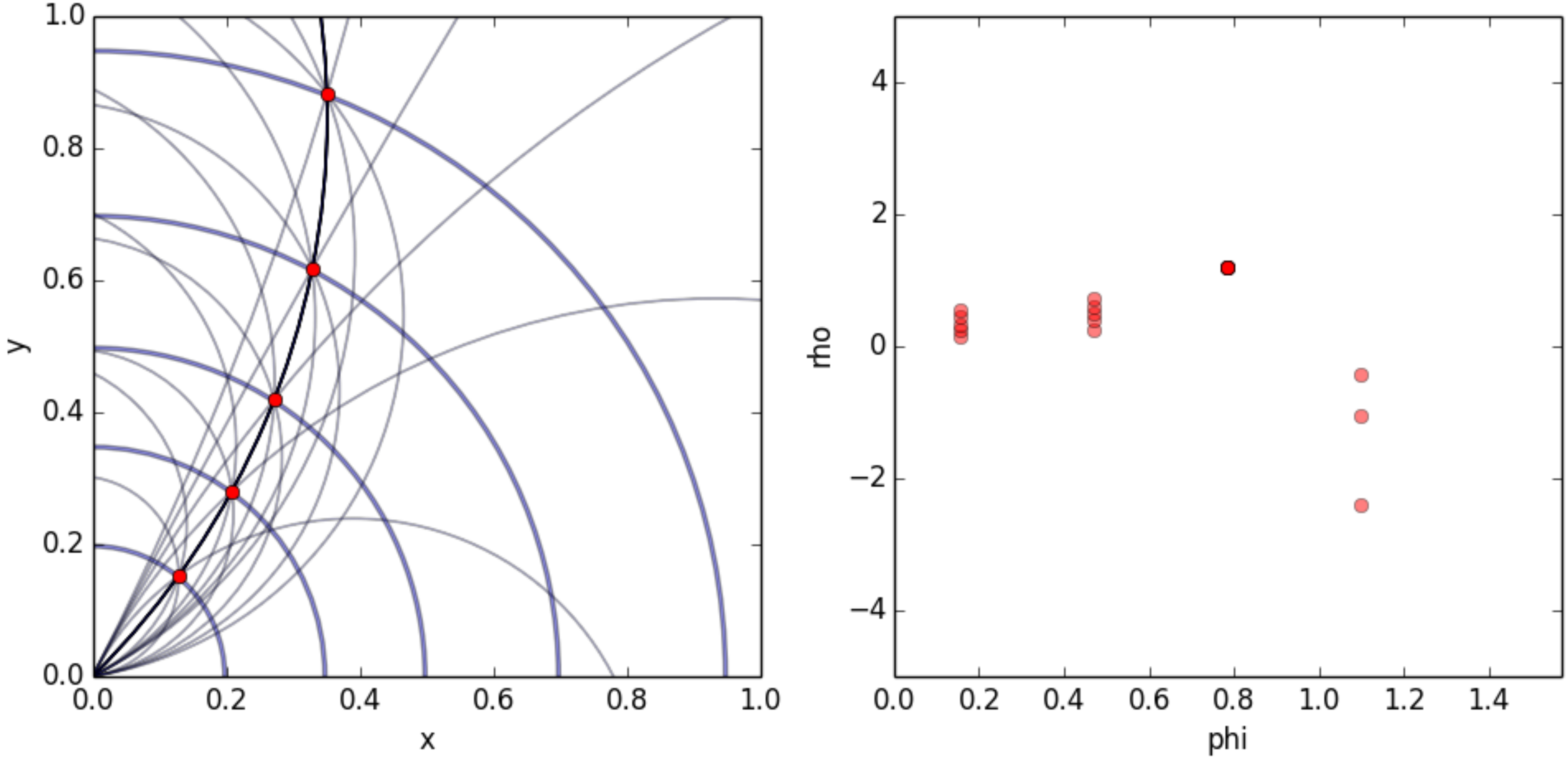
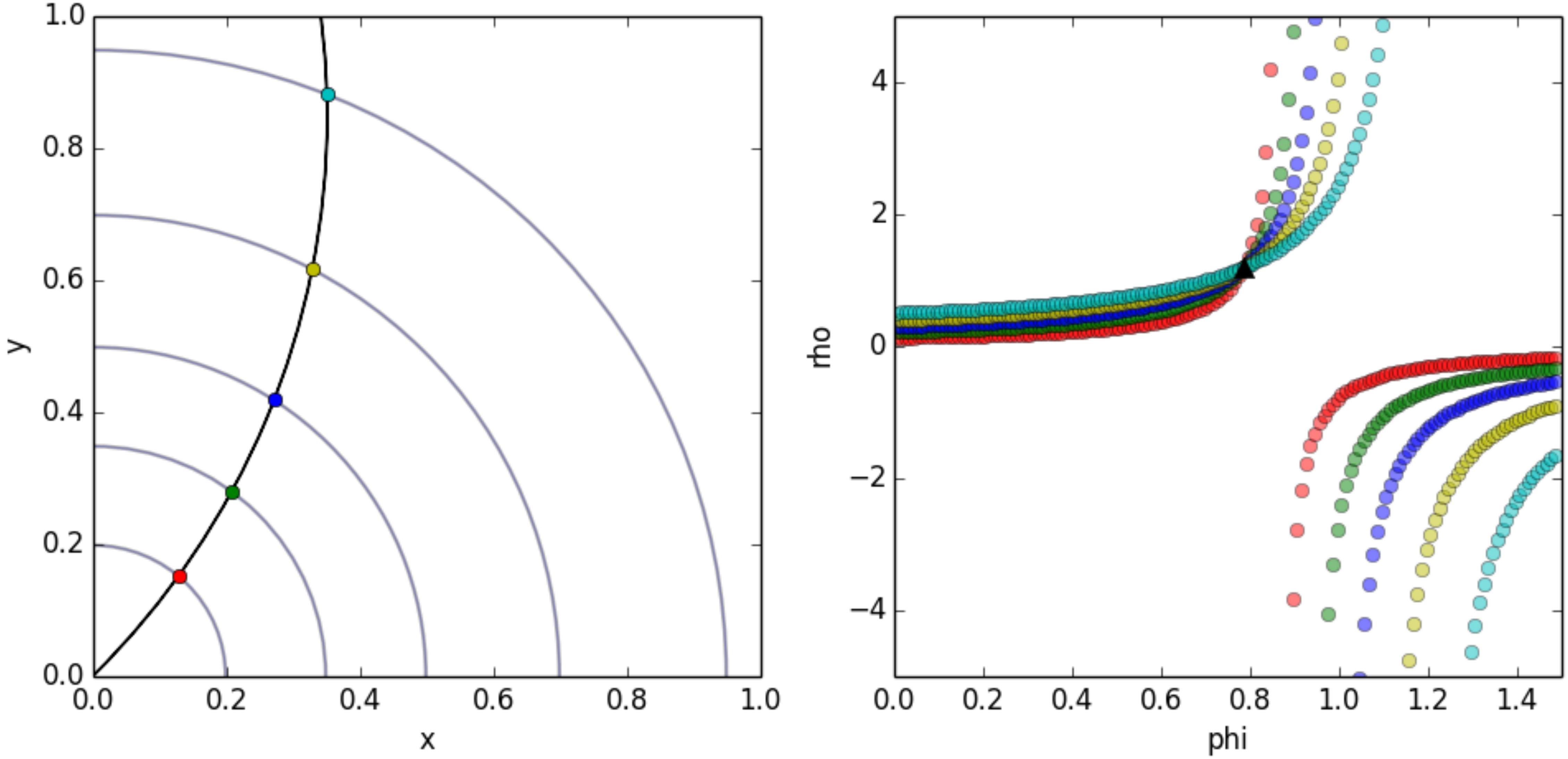


Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses..

Conformal mapping techniques Toy example

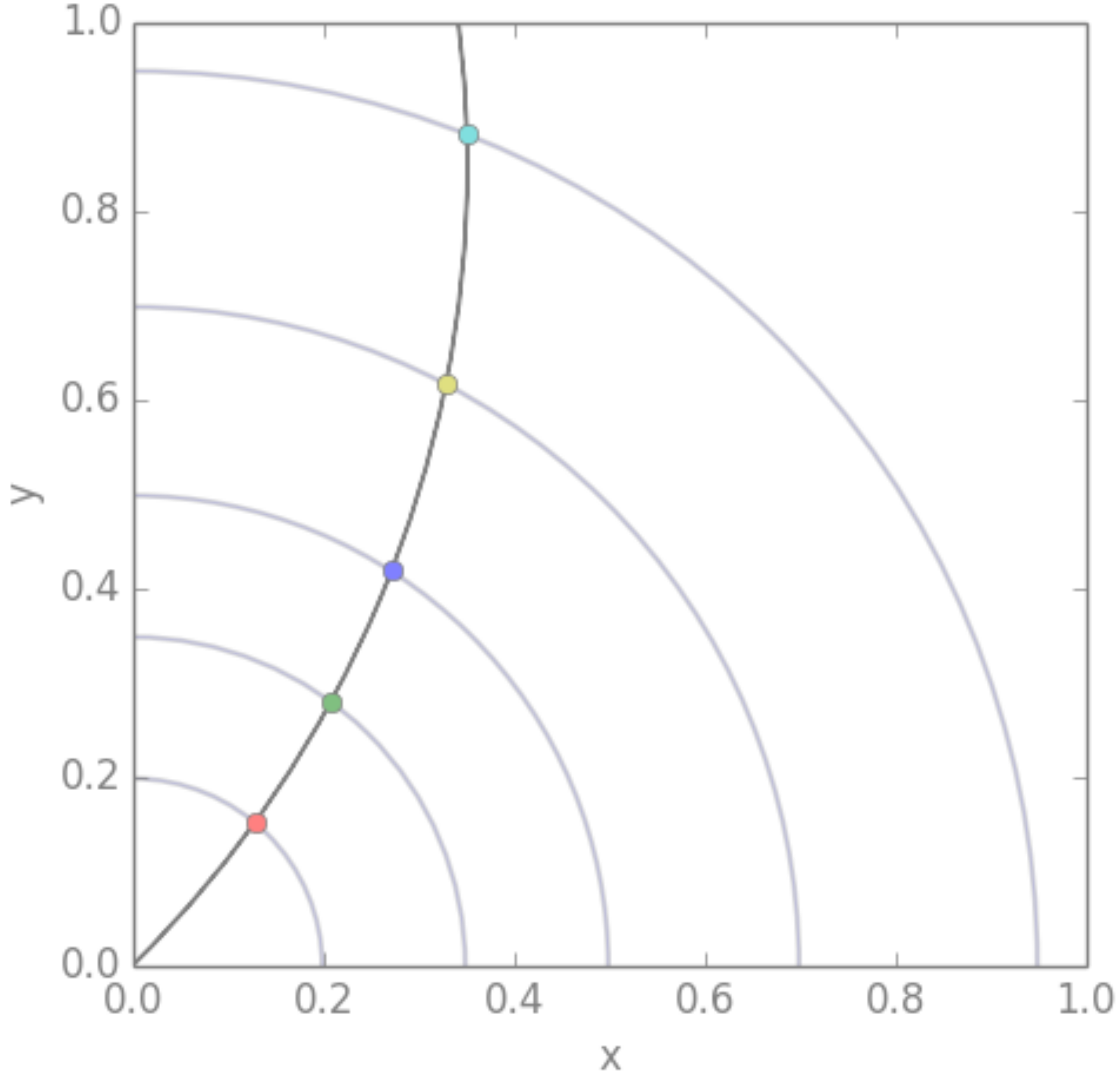


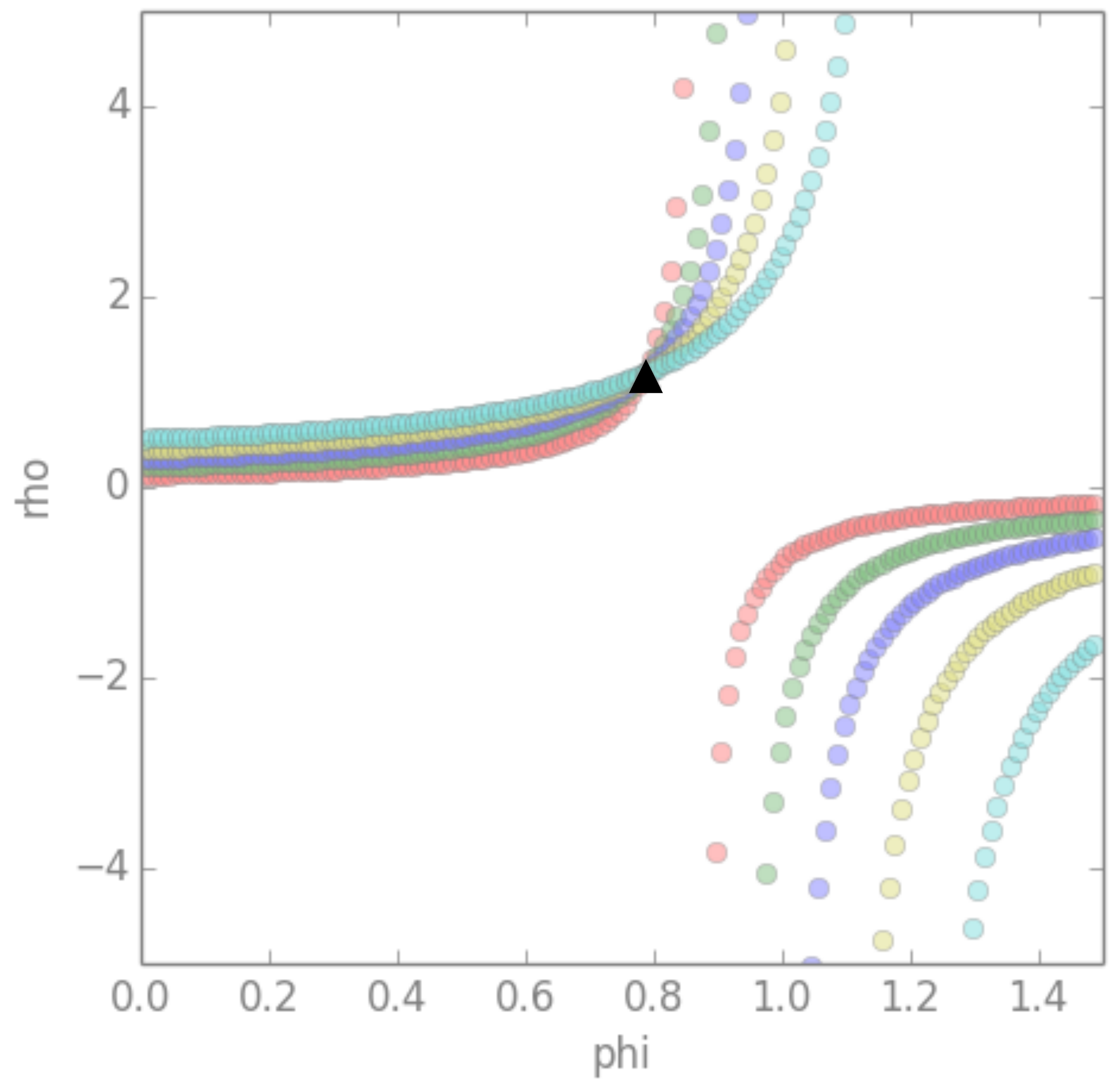
▲ … common solution compatible for all hits in (x,y) space

Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

ML clustering/unsupervised learning

here is our
domain knowledge to inject !

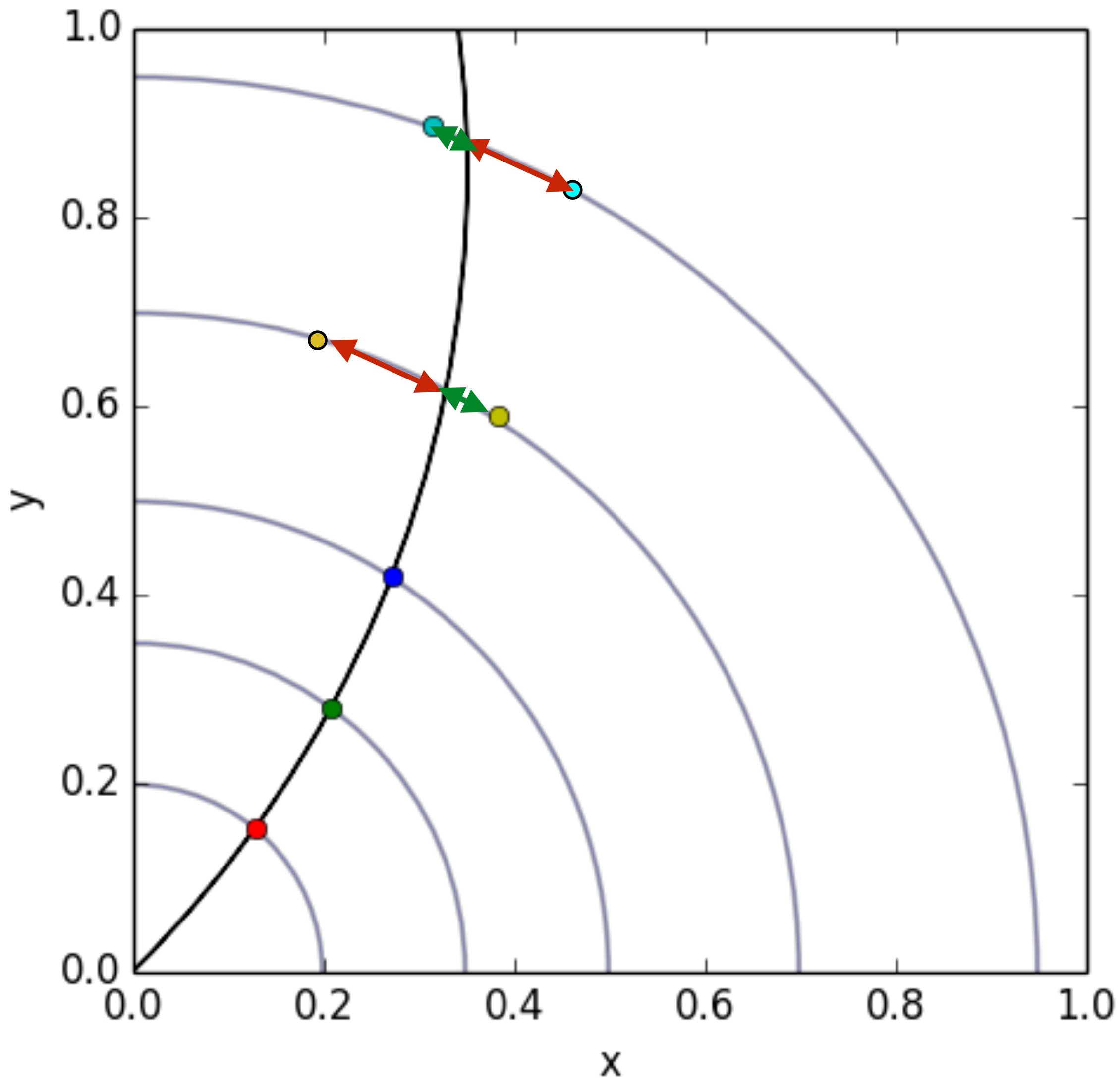




▲ … clusters in Hough space

Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

ML clustering/unsupervised learning



What defines a *belonging* relation ?

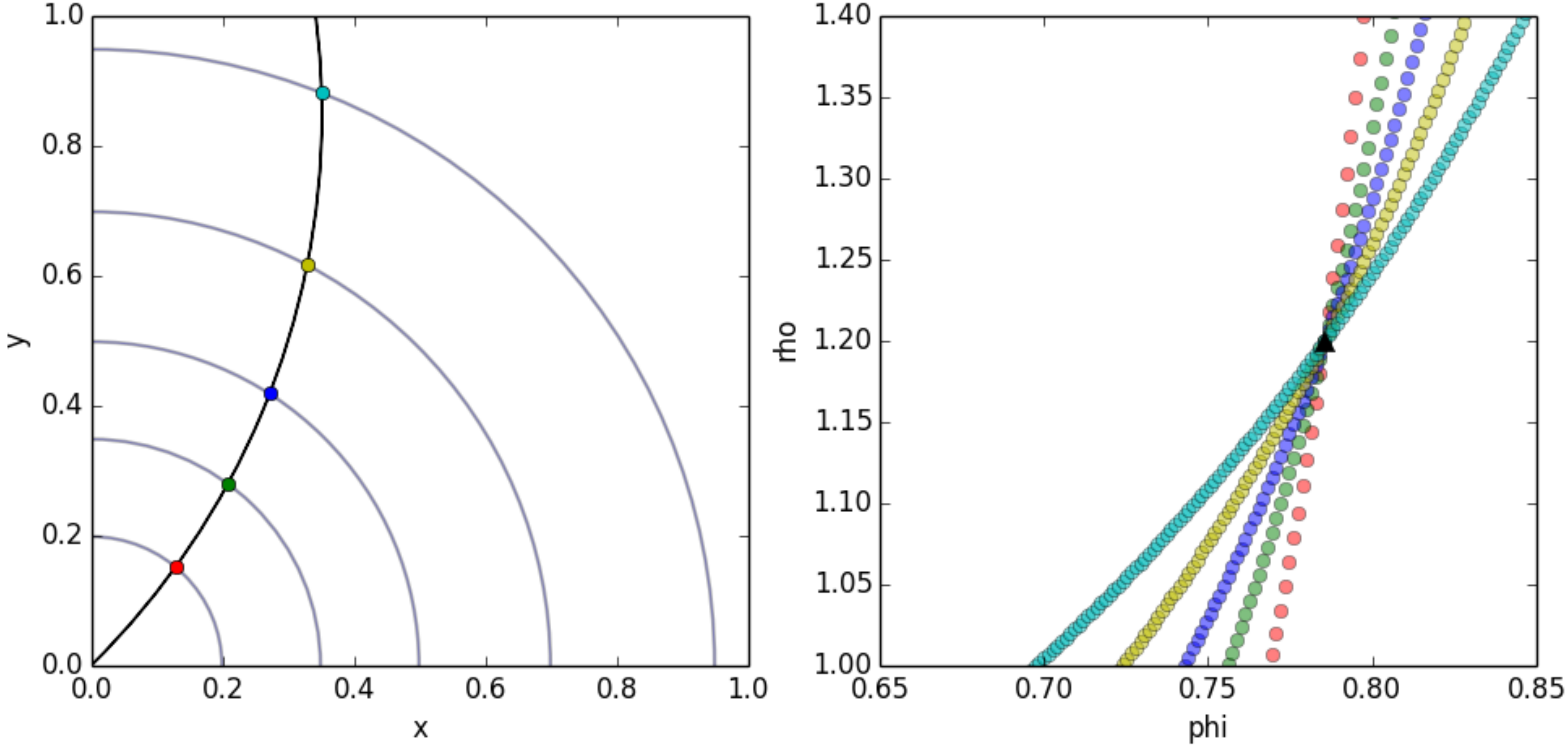
- possible solution here: distance measure as distance to the assumed solution

- additional domain knowledge: two hits on one module are not allowed

Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

Conformal mapping techniques Toy example

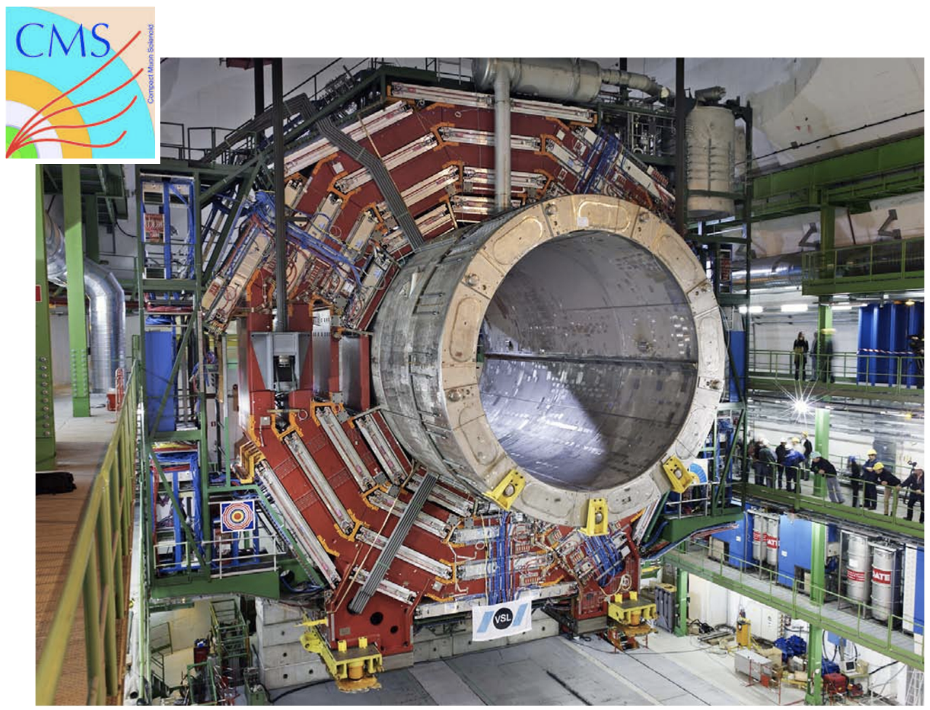
Optimal: no noise & analytical mapping

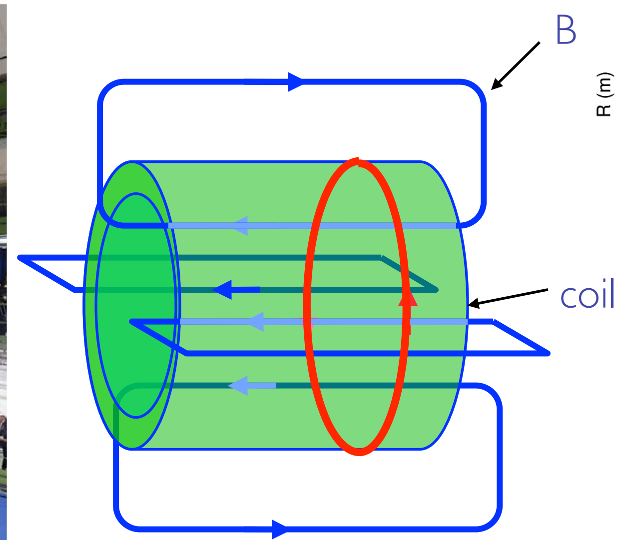


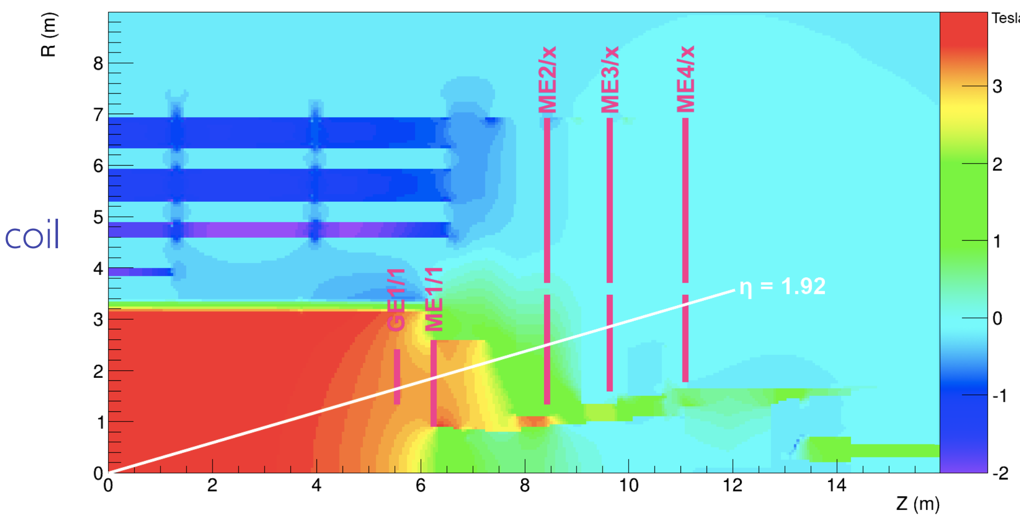
▲ … common (=true) solution compatible for all hits in (x,y) space

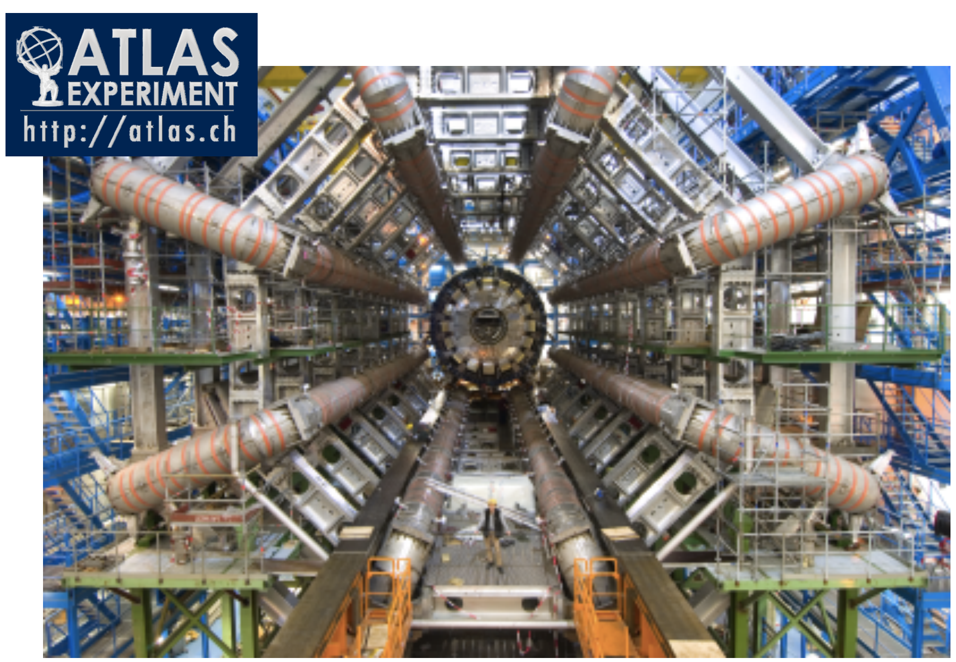
Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

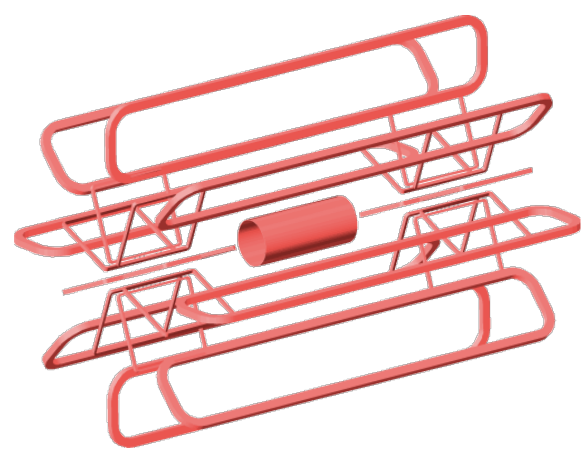
Trajectories Magnetic field maps

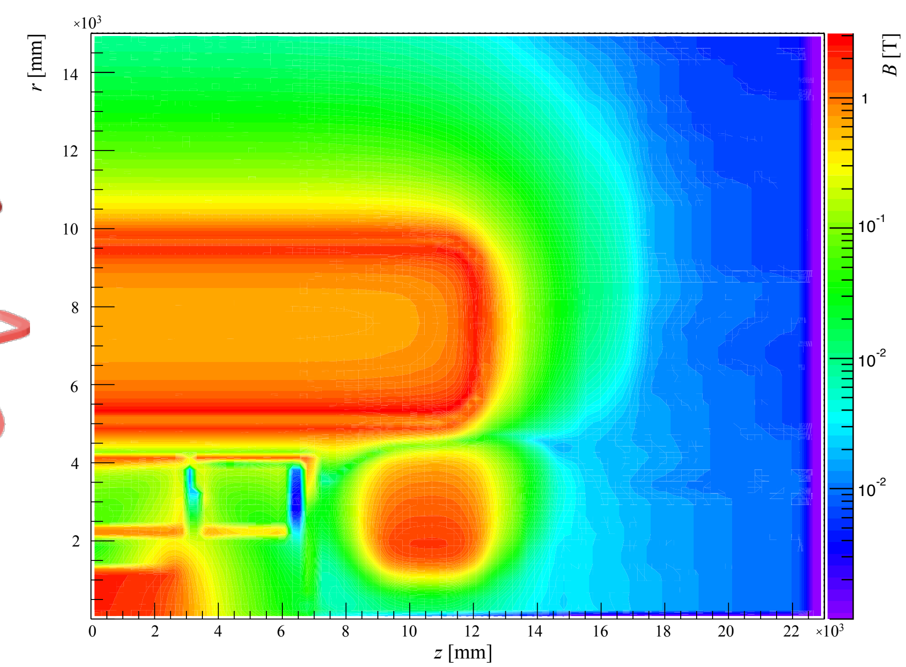










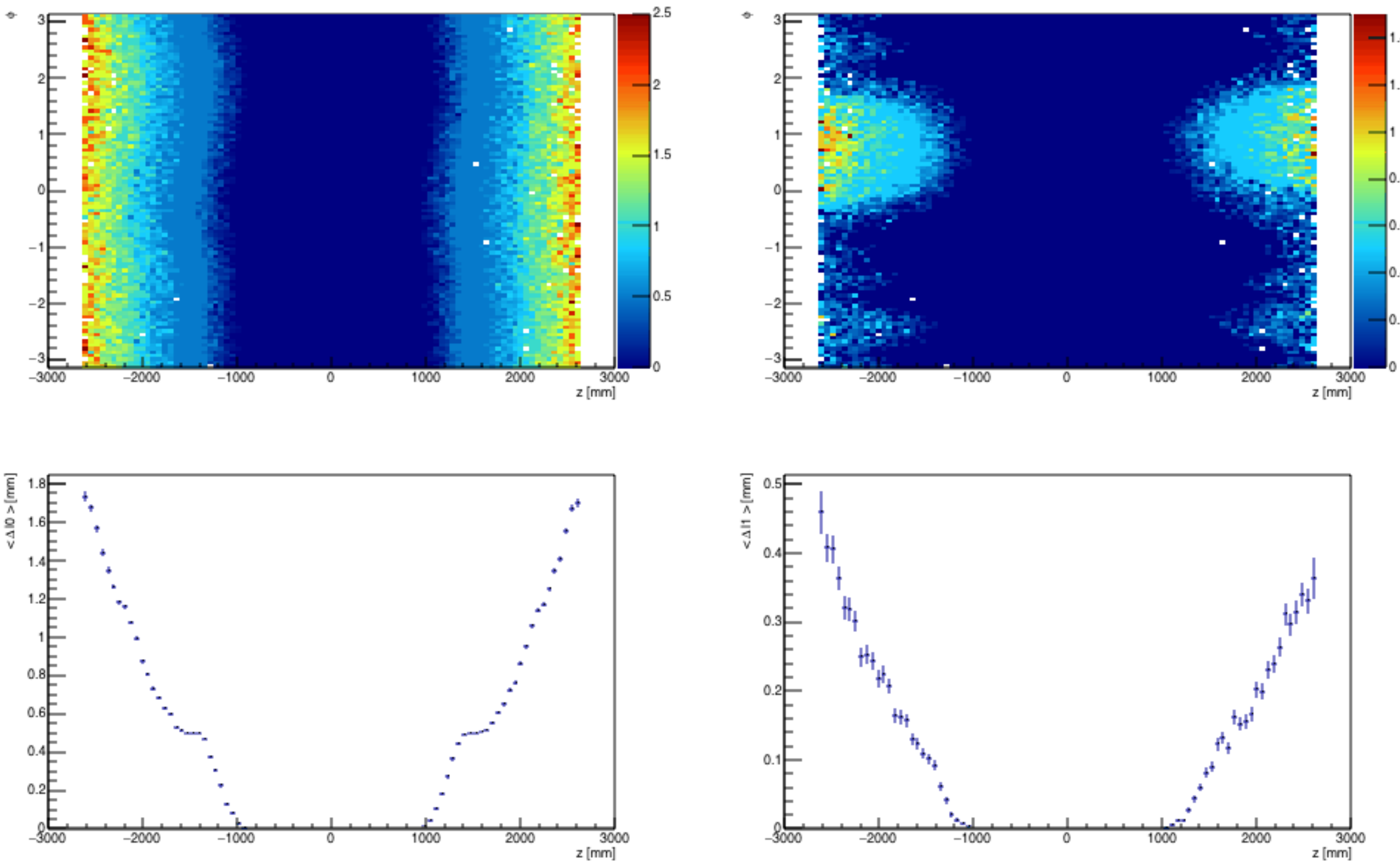


Illustrations:
Manget system of the CMS (top) and ATLAS (bottom) experiment.

Trajectories Magnetic field maps

Difference between homogenous and ATLAS magnetic field

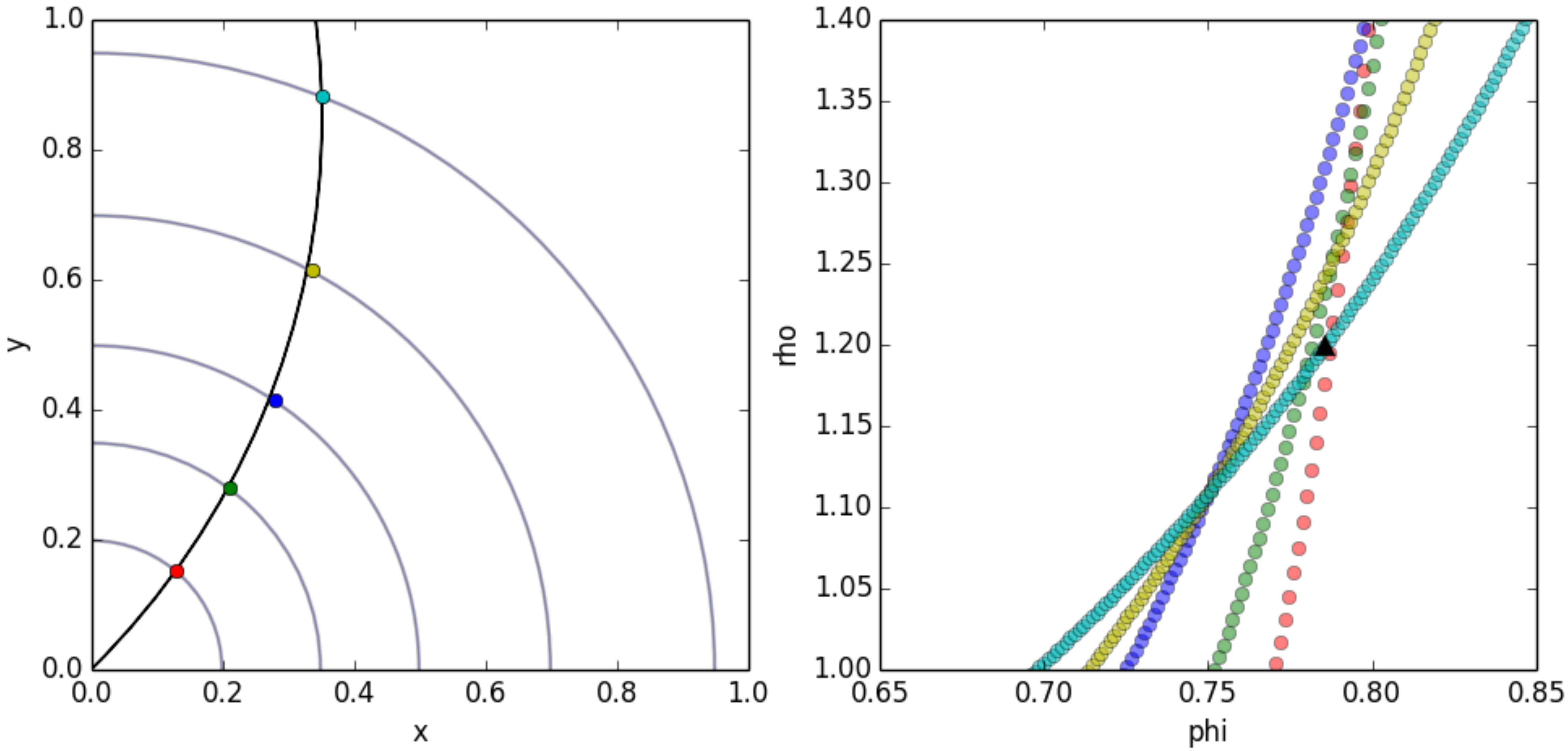
- middle layer of detector: measurement accuracy 0.05 mm, difference O(1) mm



Plots:
Difference in transverse (top) and longitudinal (bottom) extrapolation between a perfect helical and the actual ATLAS solenoidal magnetic field.

Conformal mapping techniques Toy example

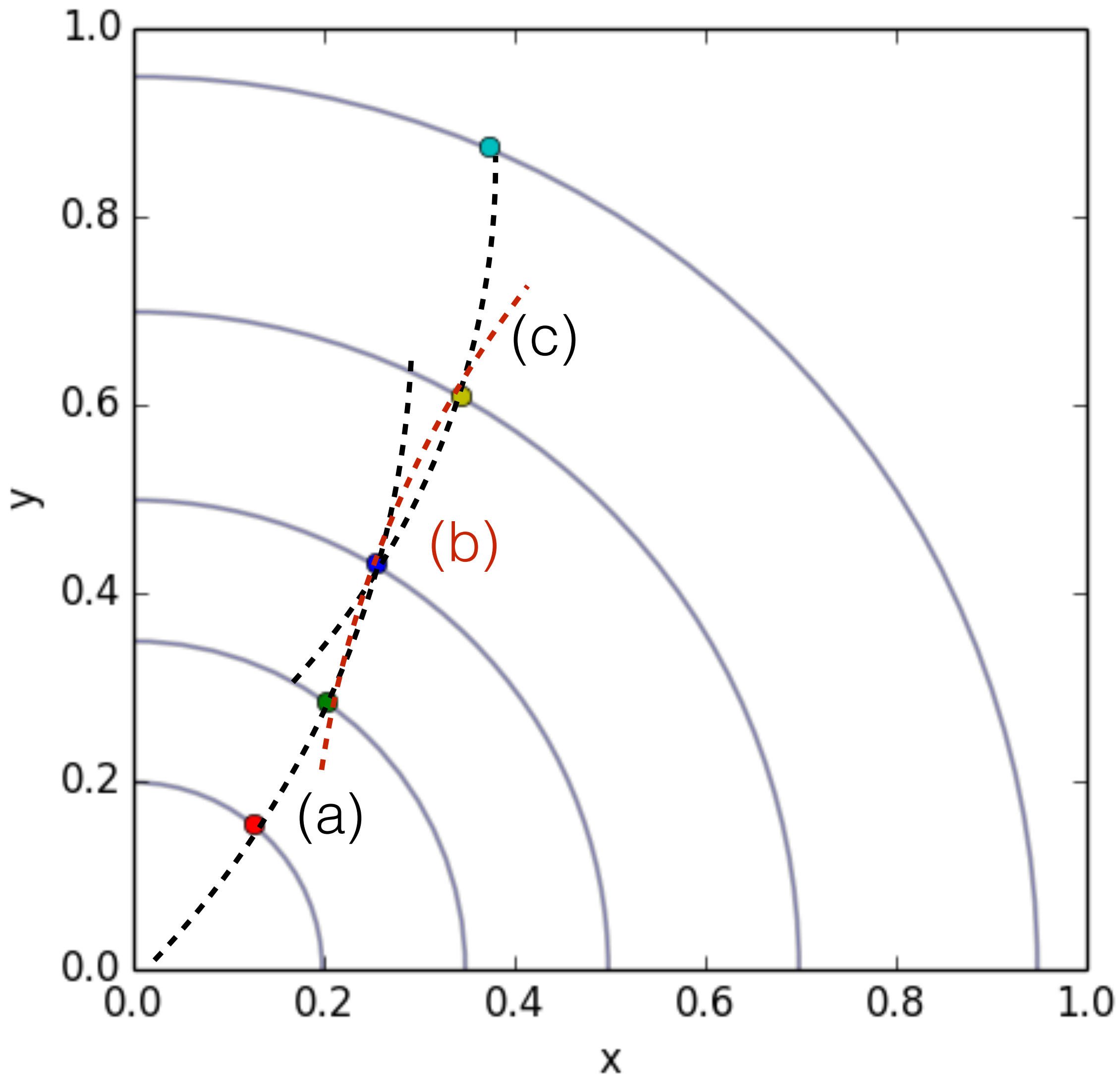
Gaussian random position smearing of 0.01 units in tangential direction



▲ … true solution of the particle emission angle and bending radius

Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

ML clustering/unsupervised learning



The assumed solution can be subject to modification

- may differ for different subsets of in the cluster
- may be updated with more information added to the cluster (as for any clustering algorithm)

Clustering with adaptive distance measure

$$\underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2 = \underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} |S_i| \operatorname{Var} S_i$$

Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

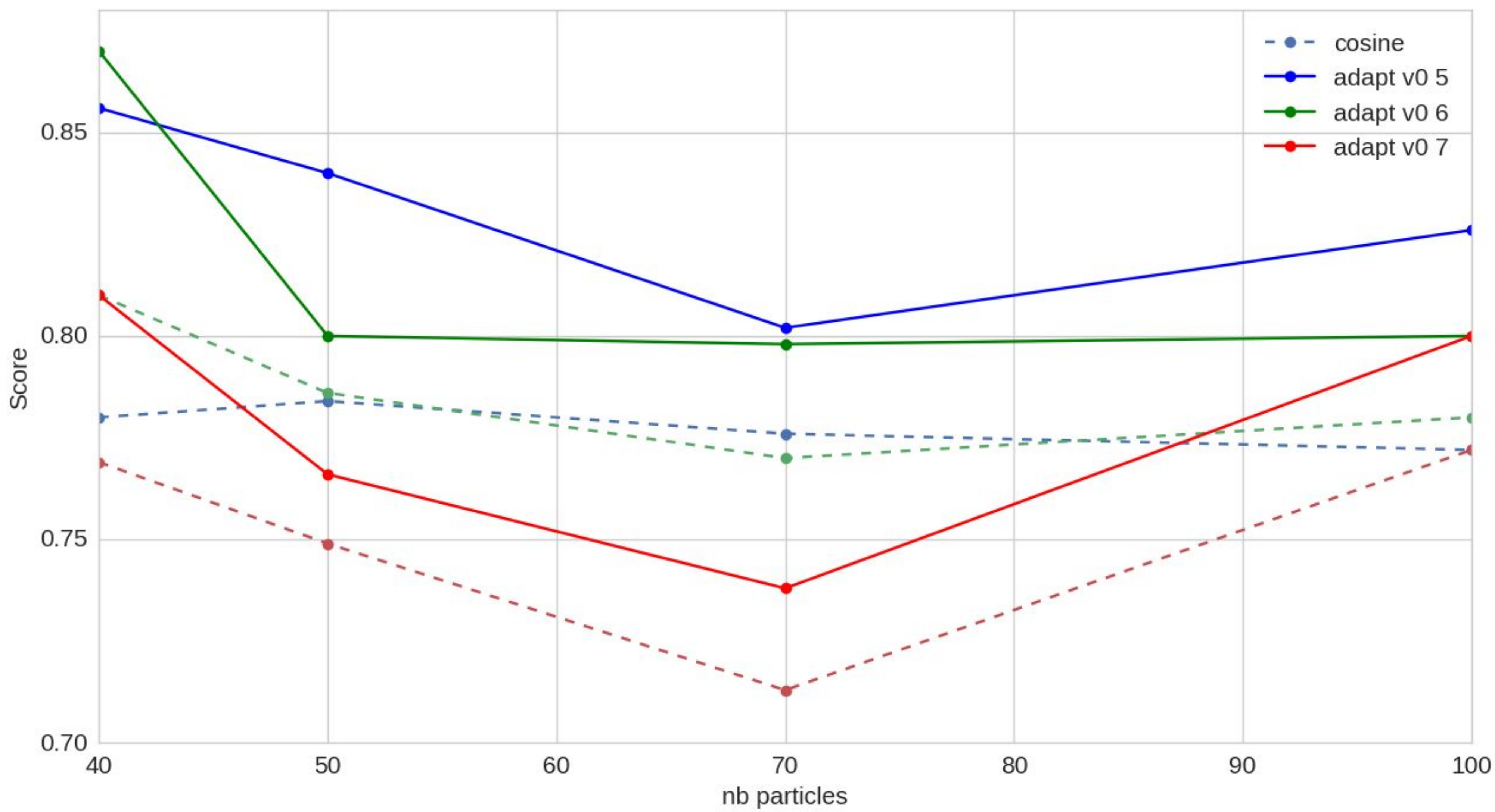
ML clustering/unsupervised learning

Cluster algorithm with *adaptive distance measure*

- distance to current track hypothesis (= cluster center in mapping space) calculated:

  transverse plane (circle)

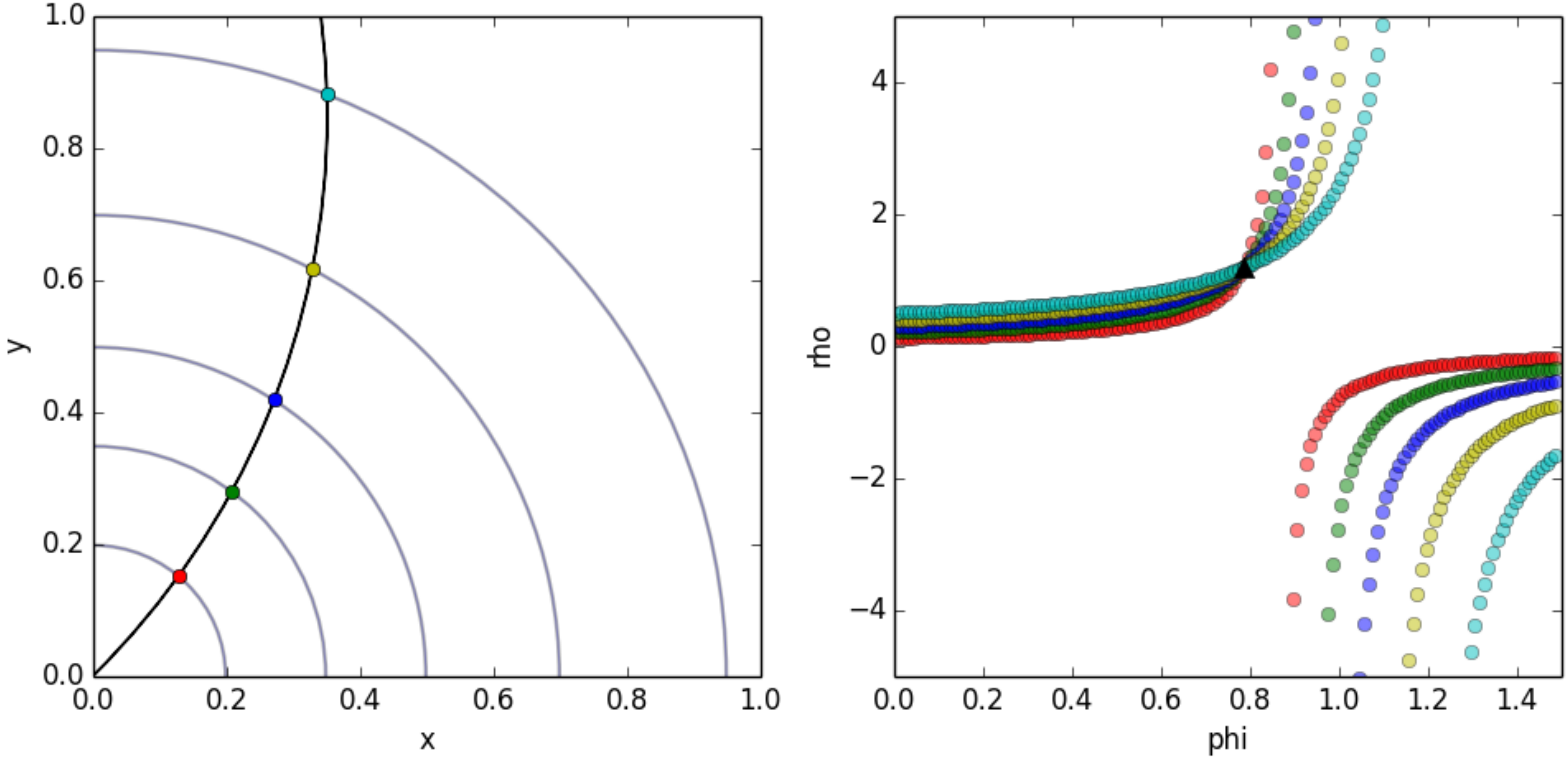
  longitudinal plane (line)



Plot:
Improvement over cosine distance measure with adaptive distance clustering, courtesy of S. Amrouche.

Conformal mapping techniques Toy example

1 particle, no smearing applied.

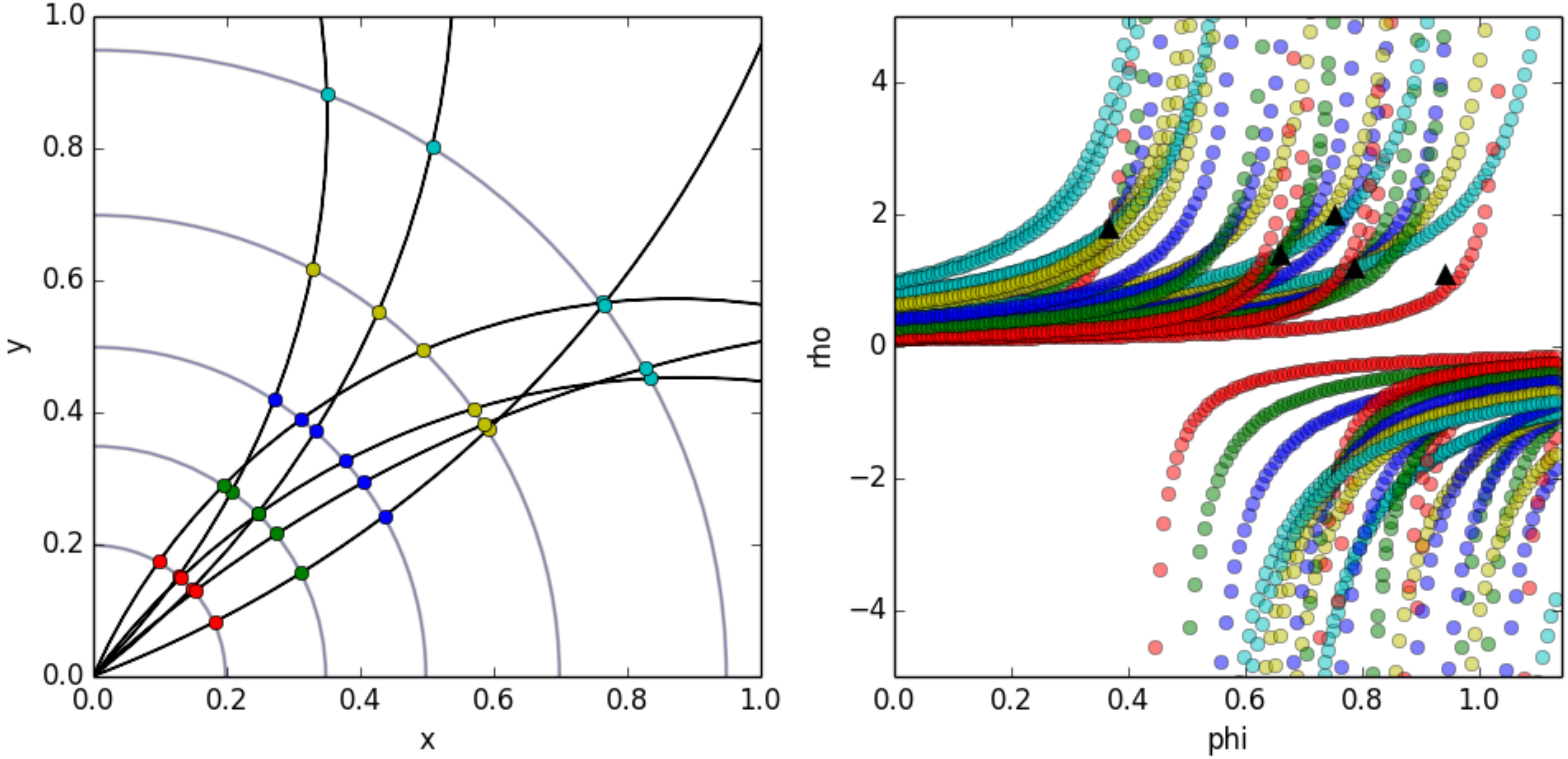


▲ … common/true solution of the particle emission angle and bending radius

Illustration:
Hits created by one particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

Conformal mapping techniques Toy example

6 particles, no smearing applied.

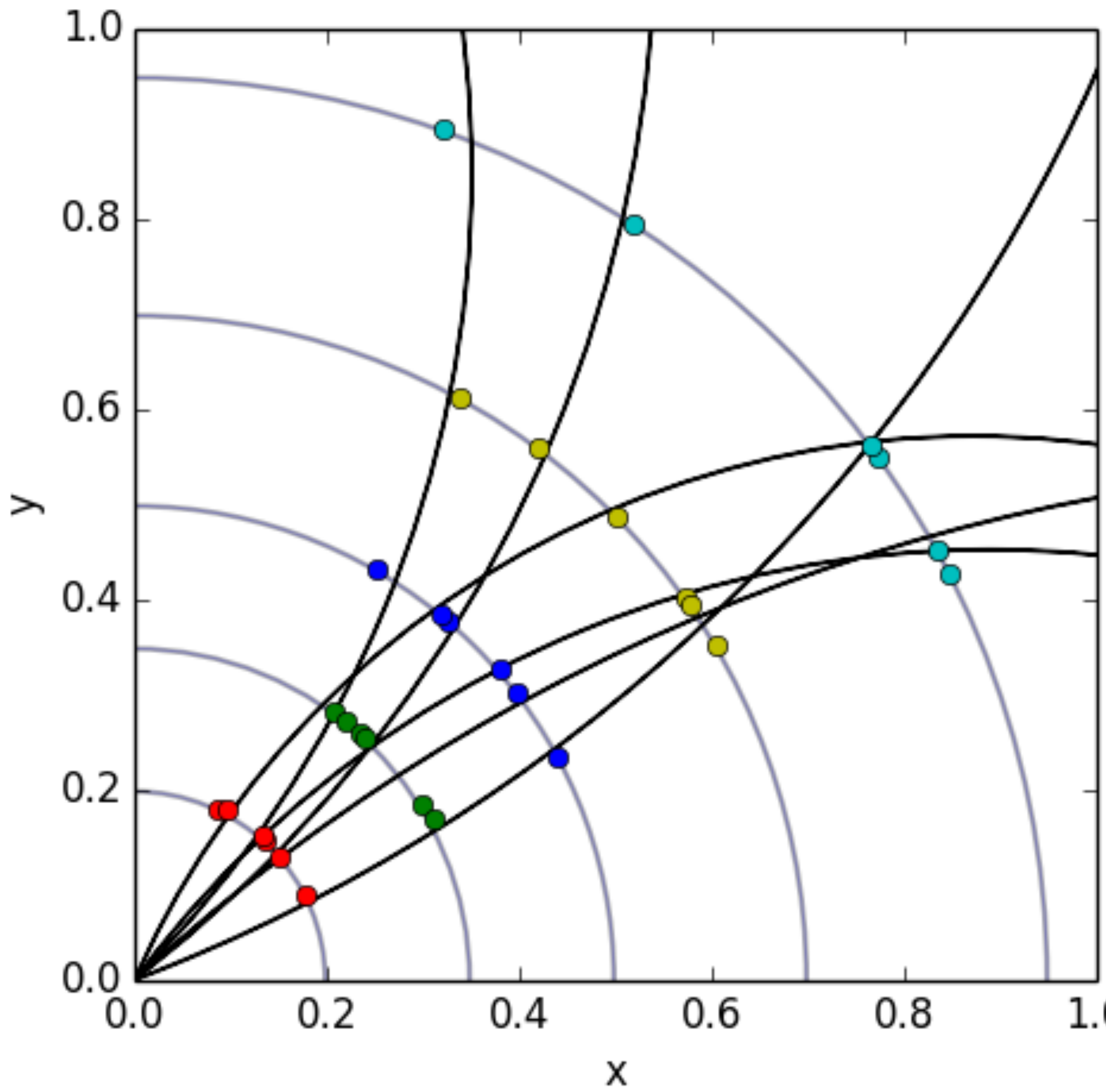


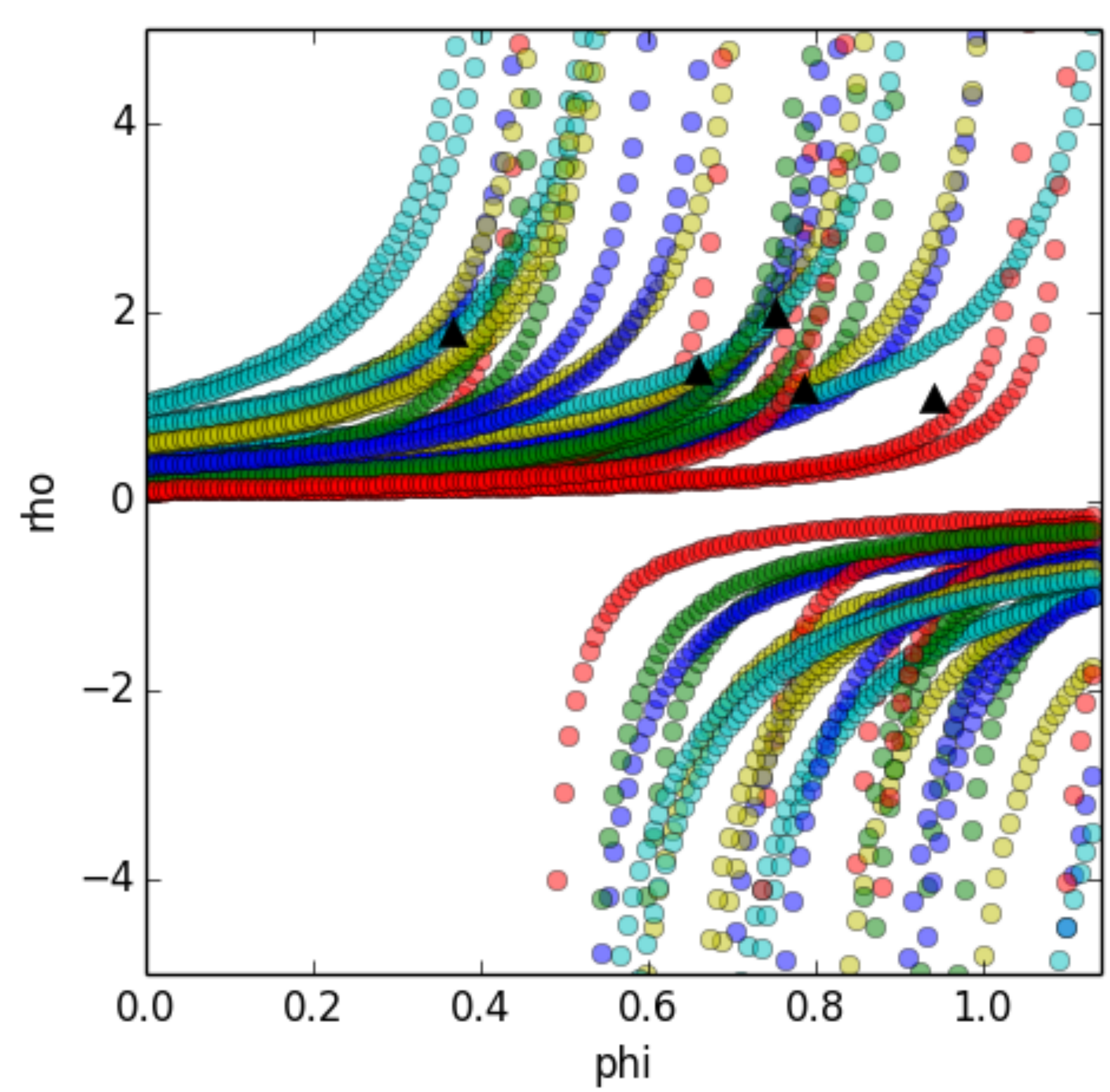
▲ … common/true solutions of the particles emission angles and bending radii

Illustration:
Hits created by six particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

Conformal mapping techniques Toy example

6 particles, smearing applied.





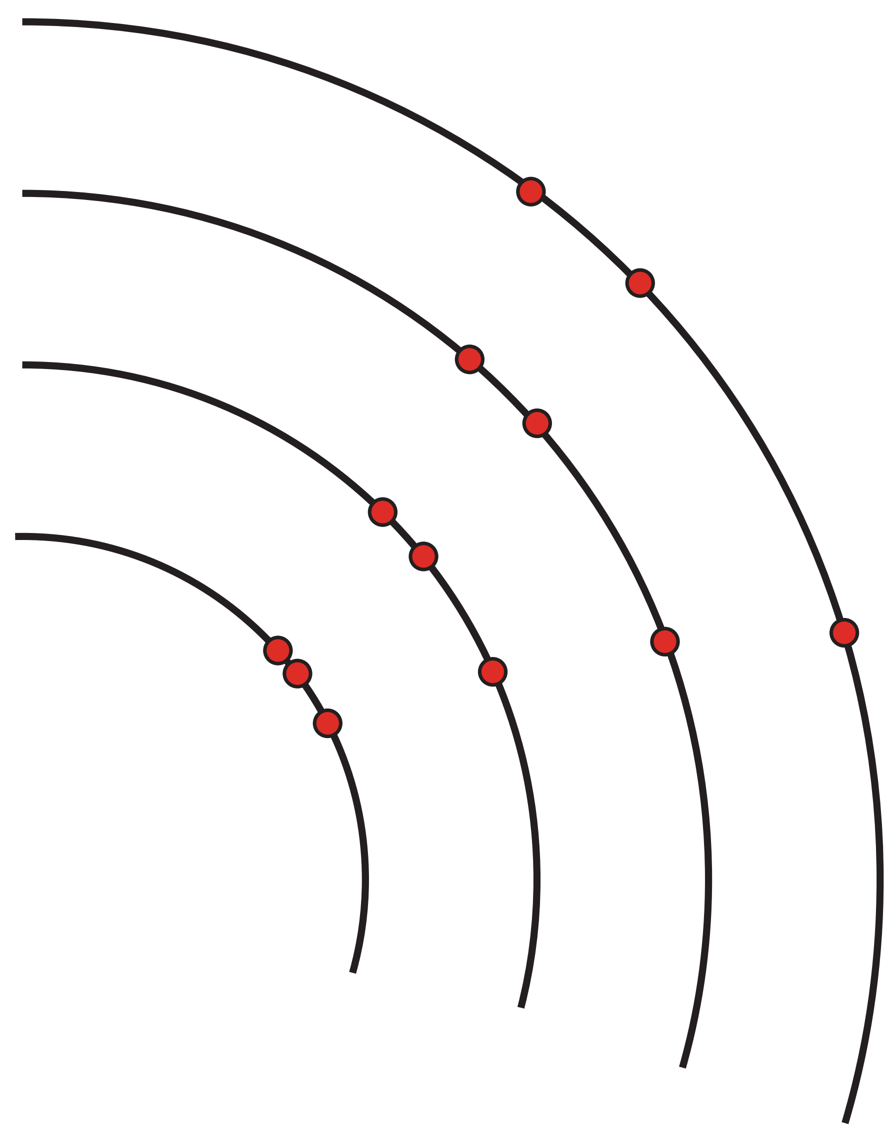
▲ … true solutions of the particles emission angles and bending radii

Illustration:
Hits created by six particles in a simplified tracking detector, transverse view (left). Right: scan through hough space with different hypotheses for all hits.

Track seeding and following Basics

Start of many track finding algorithms is the building of track seeds

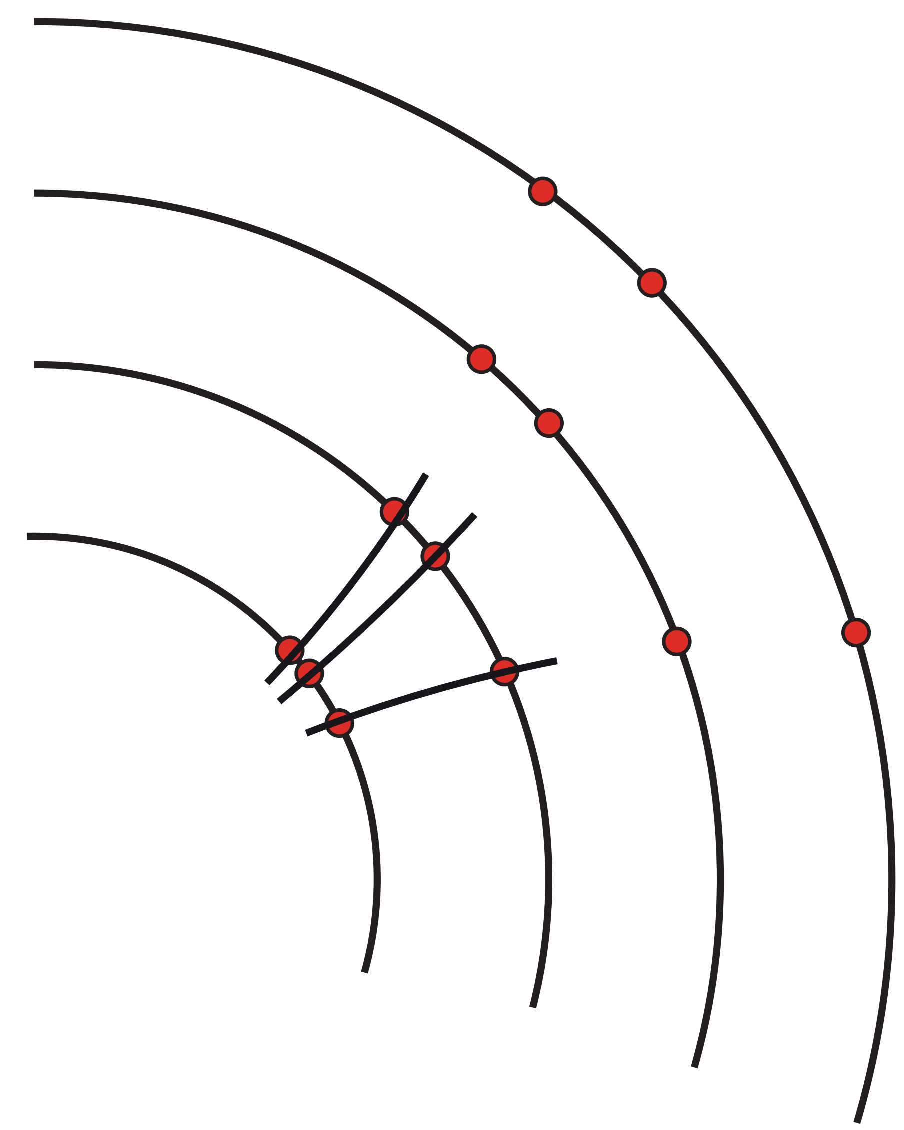
- groups of 2 or 3 measurements that are compatible with a crude track hypothesis
- seeds are used to build roads to find track candidates



Track seeding and following Basics

Start of many track finding algorithms is the building of track seeds

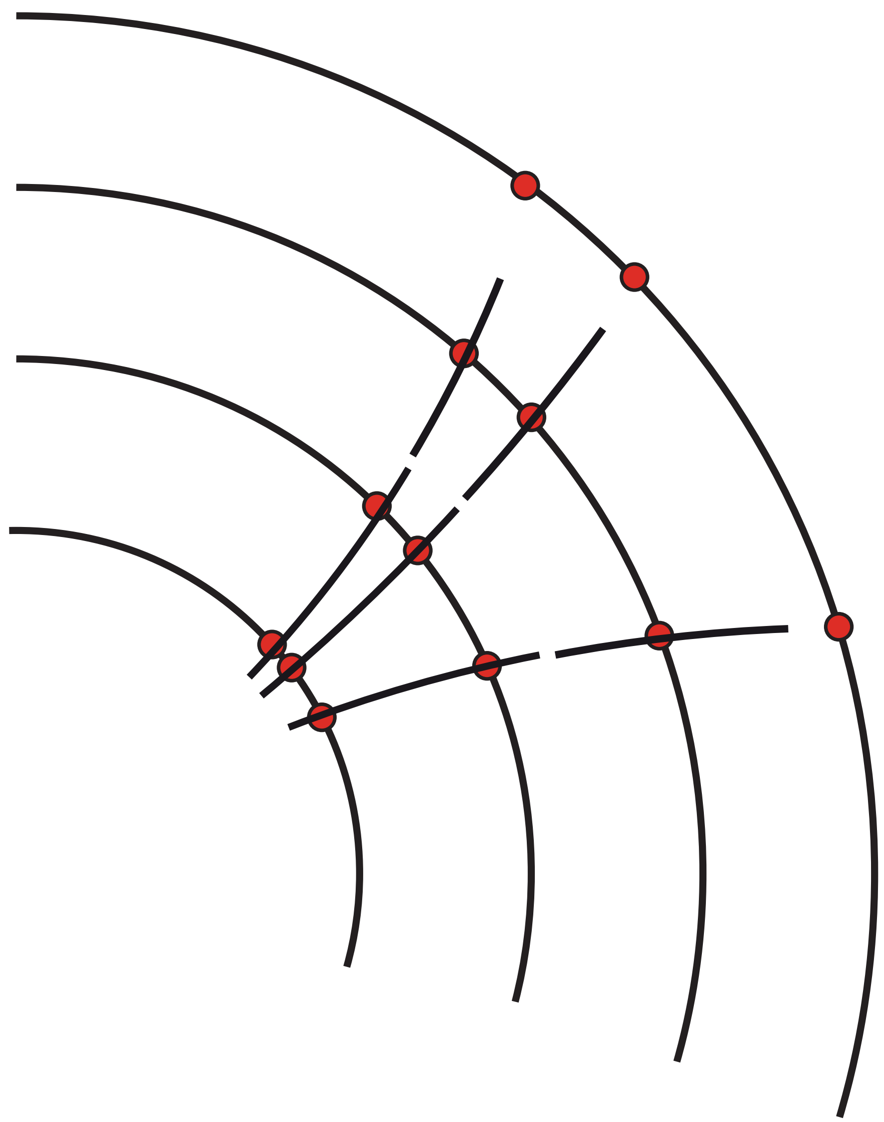
- groups of 2 or 3 measurements that are compatible with a crude track hypothesis
- seeds are used to build roads to find track candidates



Track seeding and following Basics

Start of many track finding algorithms is the building of track seeds

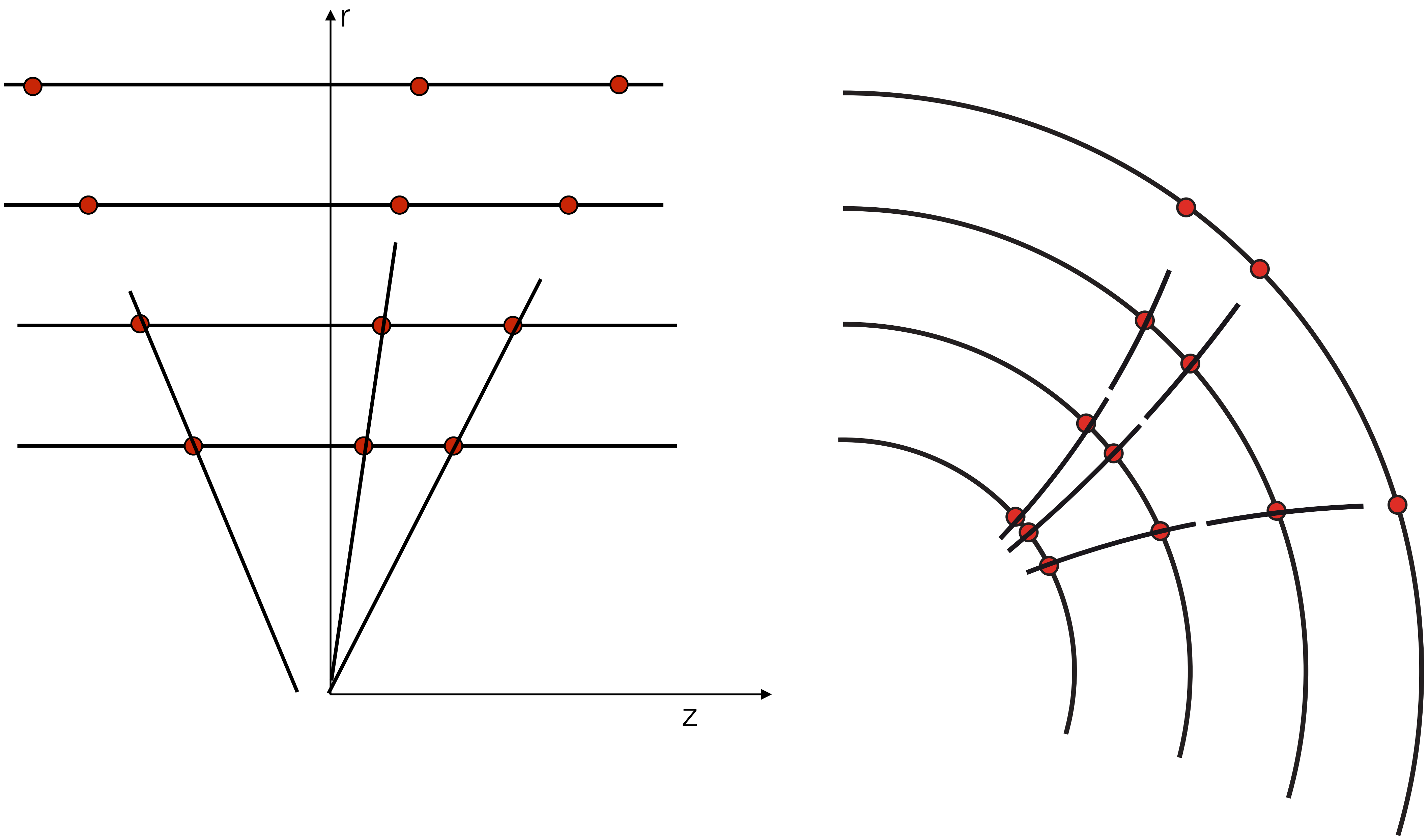
- groups of 2 or 3 measurements that are compatible with a crude track hypothesis
- seeds are used to build roads to find track candidates



Track seeding and following Basics

Start of many track finding algorithms is the building of track seeds

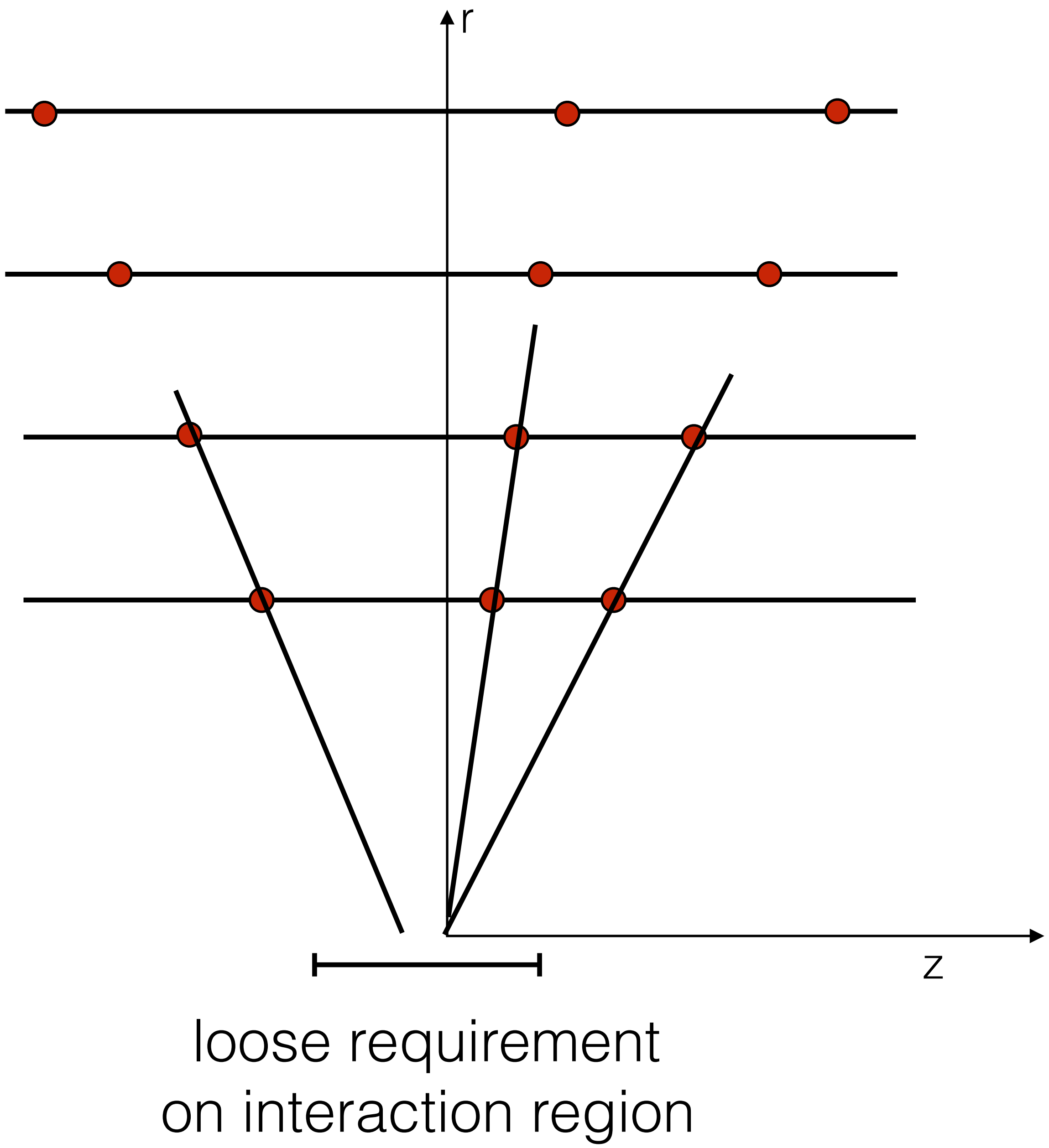
- groups of 2 or 3 measurements that are compatible with a crude track hypothesis
- seeds are used to build roads to find track candidates

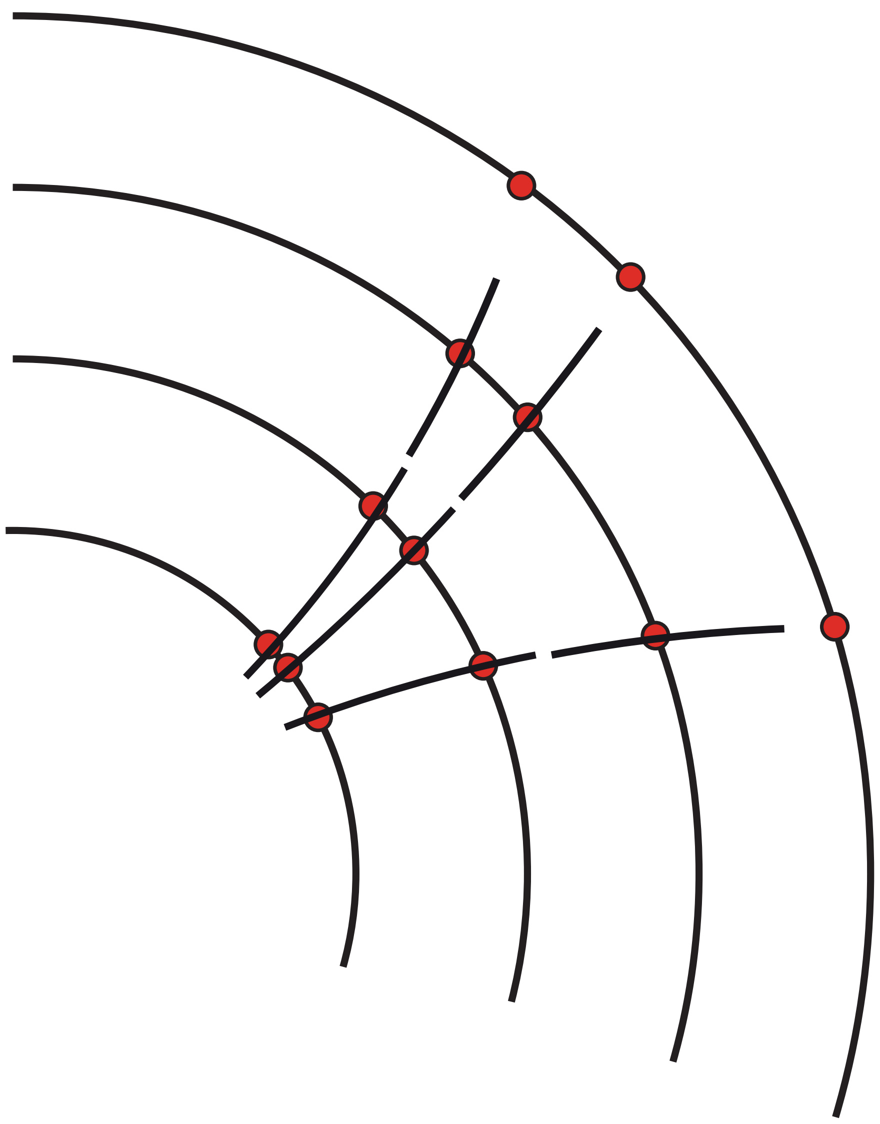


Track seeding and following Basics

Start of many track finding algorithms is the building of track seeds

- groups of 2 or 3 measurements that are compatible with a crude track hypothesis
- seeds are used to build roads to find track candidates

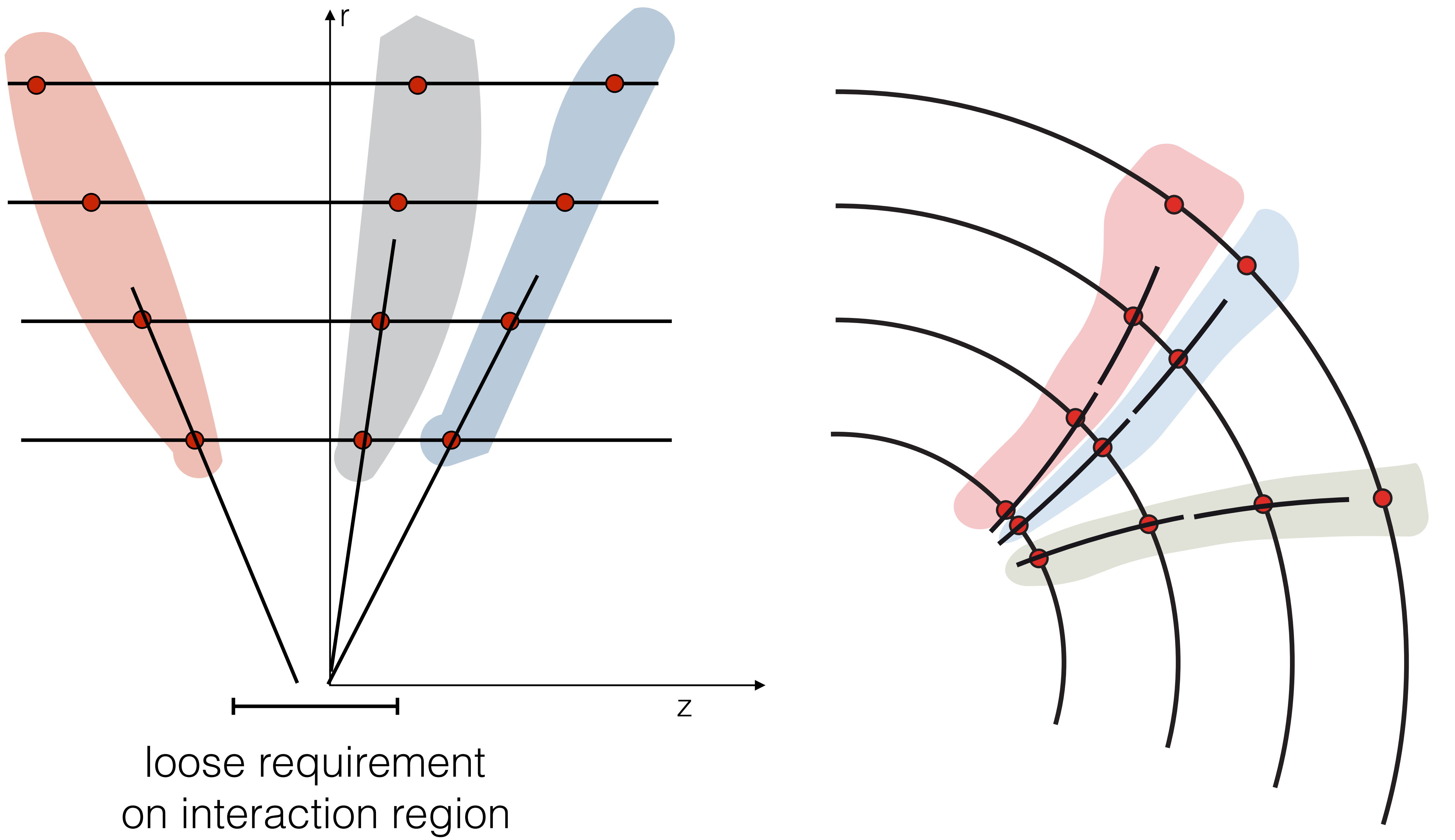




Track seeding and following Basics

Start of many track finding algorithms is the building of track seeds

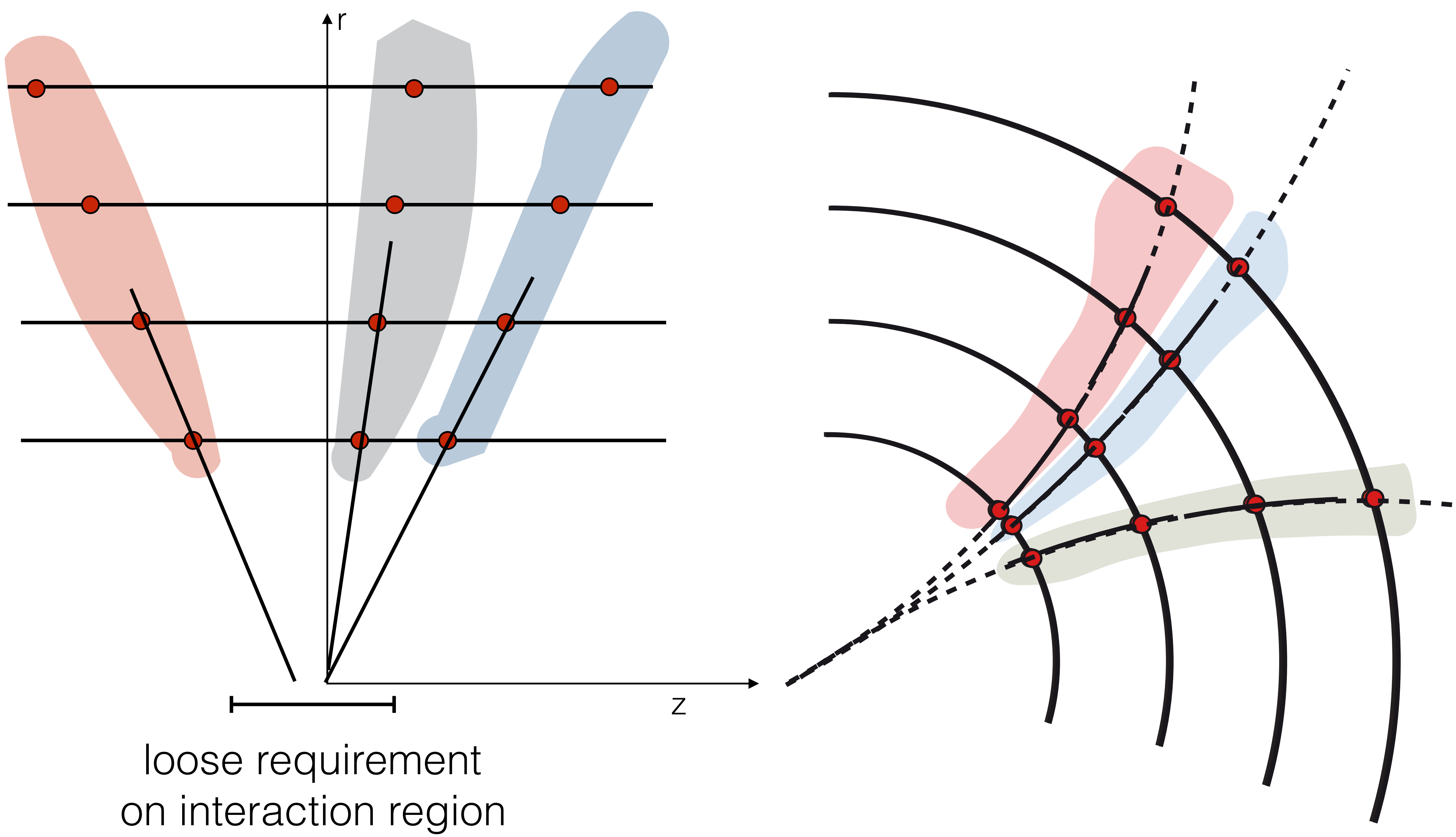
- groups of 2 or 3 measurements that are compatible with a crude track hypothesis
- seeds are used to build roads to find track candidates



Track seeding and following Basics

Start of many track finding algorithms is the building of track seeds

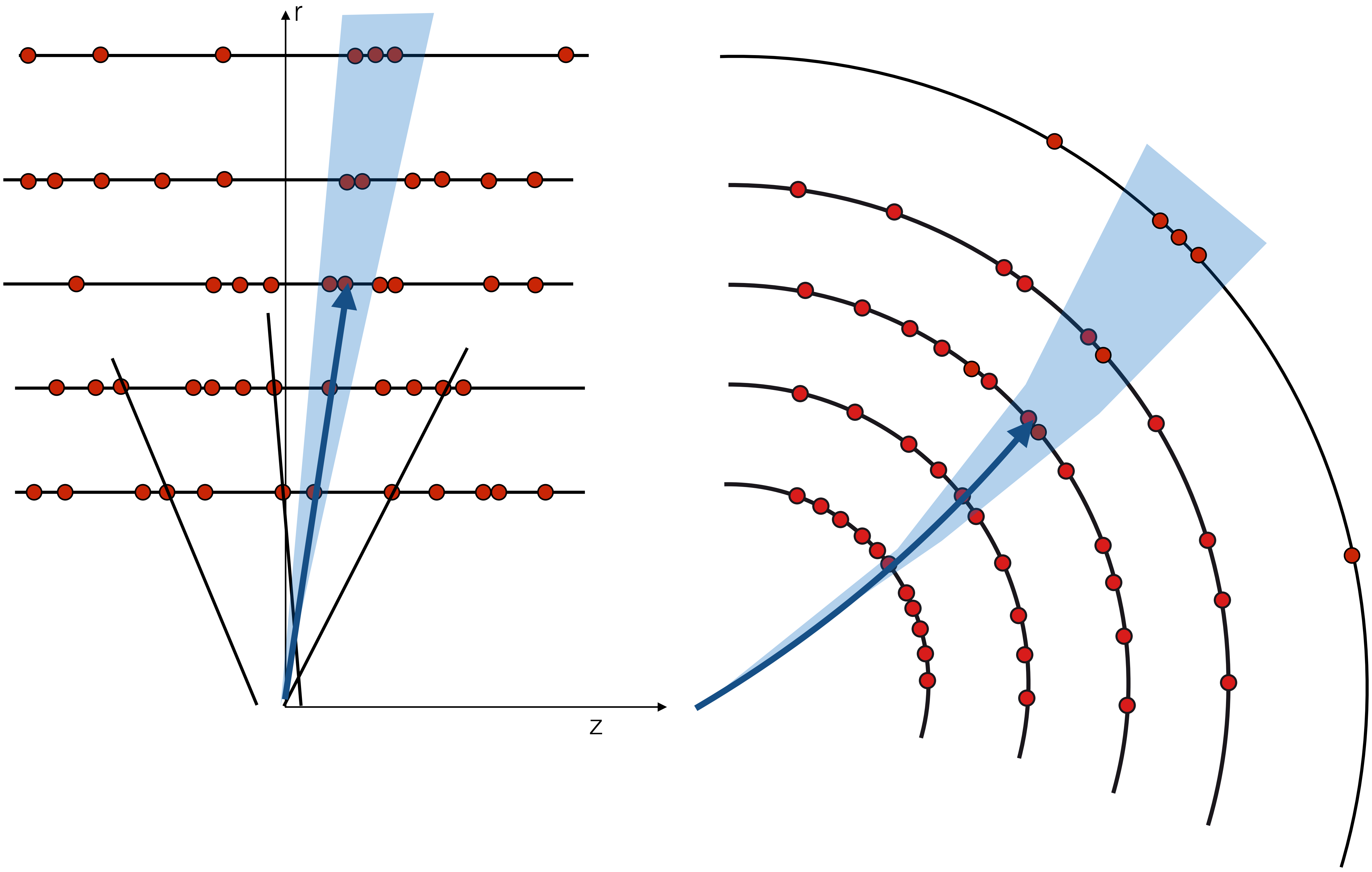
- groups of 2 or 3 measurements that are compatible with a crude track hypothesis
- seeds are used to build roads to find track candidates



Track following combinatorial filter

Dense environments create problems for the progressive filter

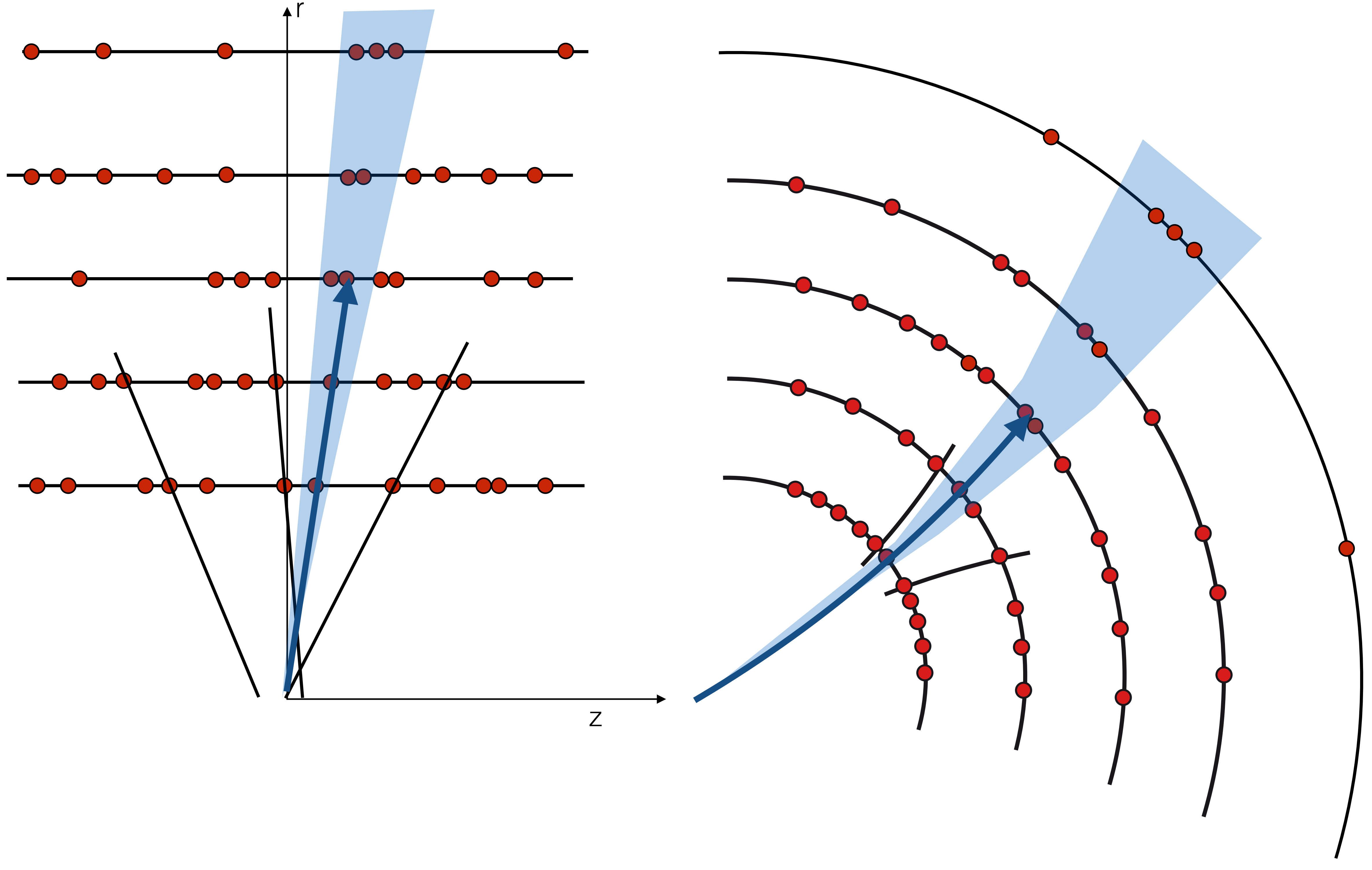
there may not always be one obvious path to be followed:



Track following combinatorial filter

Dense environments create problems for the progressive filter

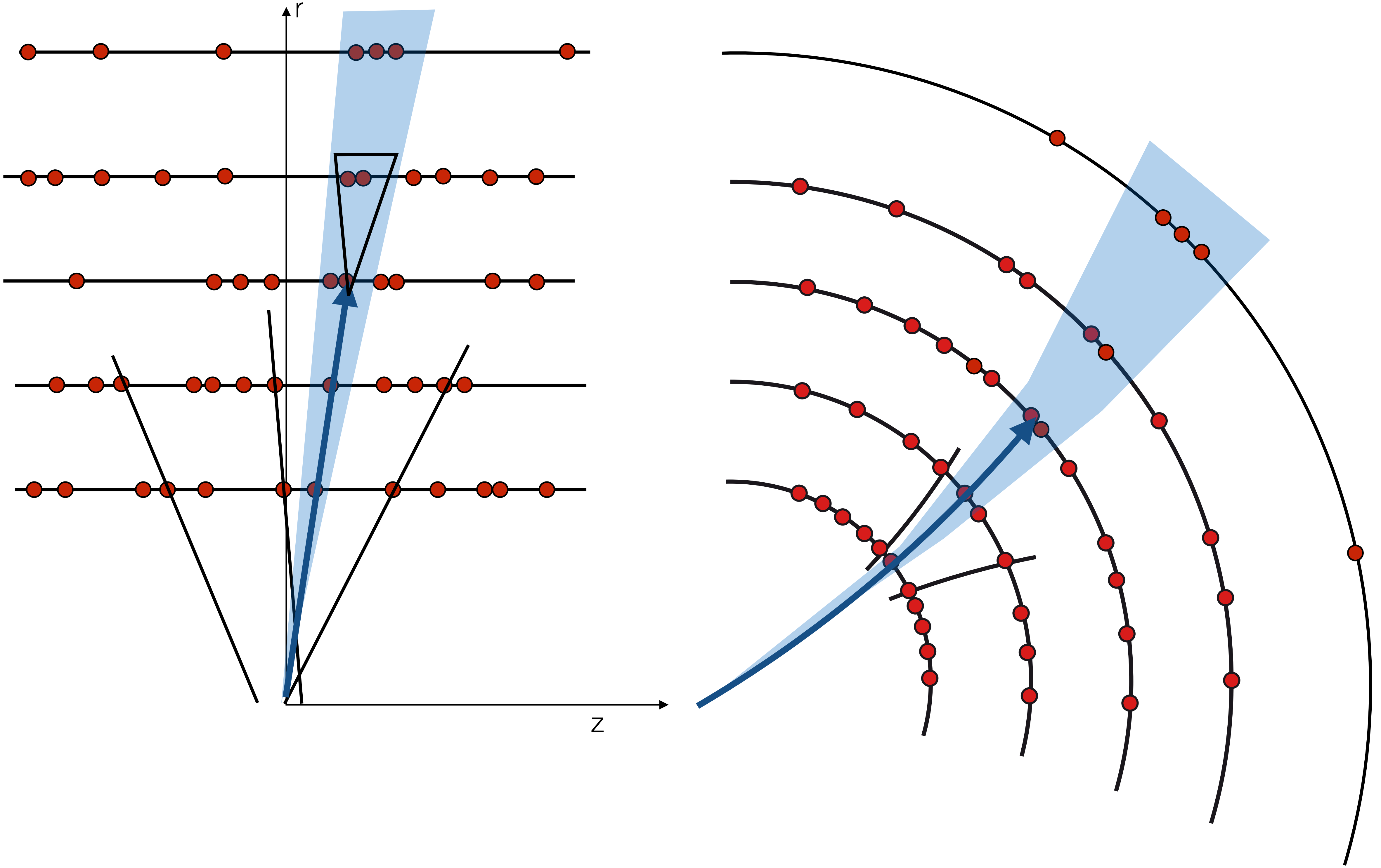
there may not always be one obvious path to be followed:



Track following combinatorial filter

Dense environments create problems for the progressive filter

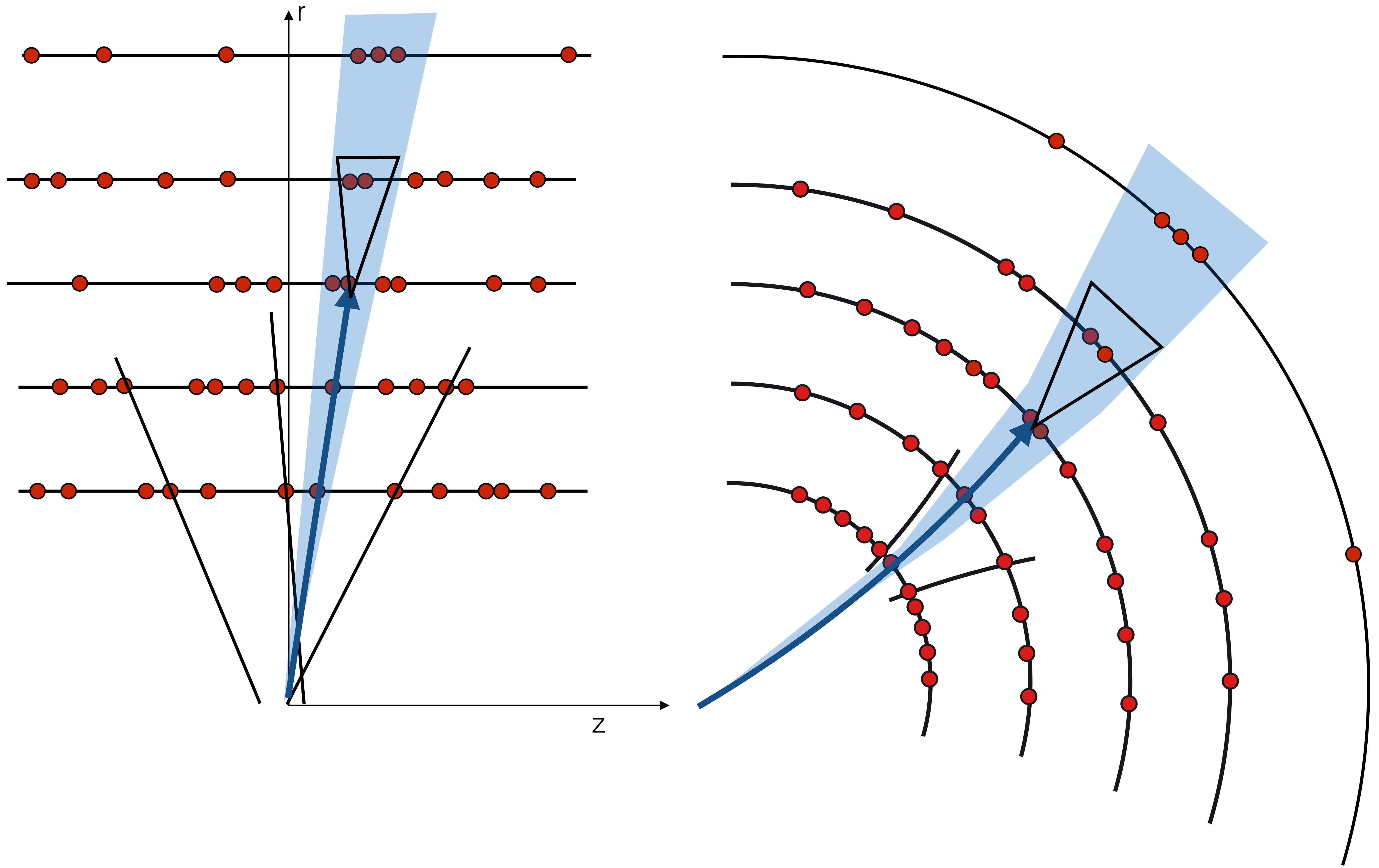
there may not always be one obvious path to be followed:



Track following combinatorial filter

Dense environments create problems for the progressive filter

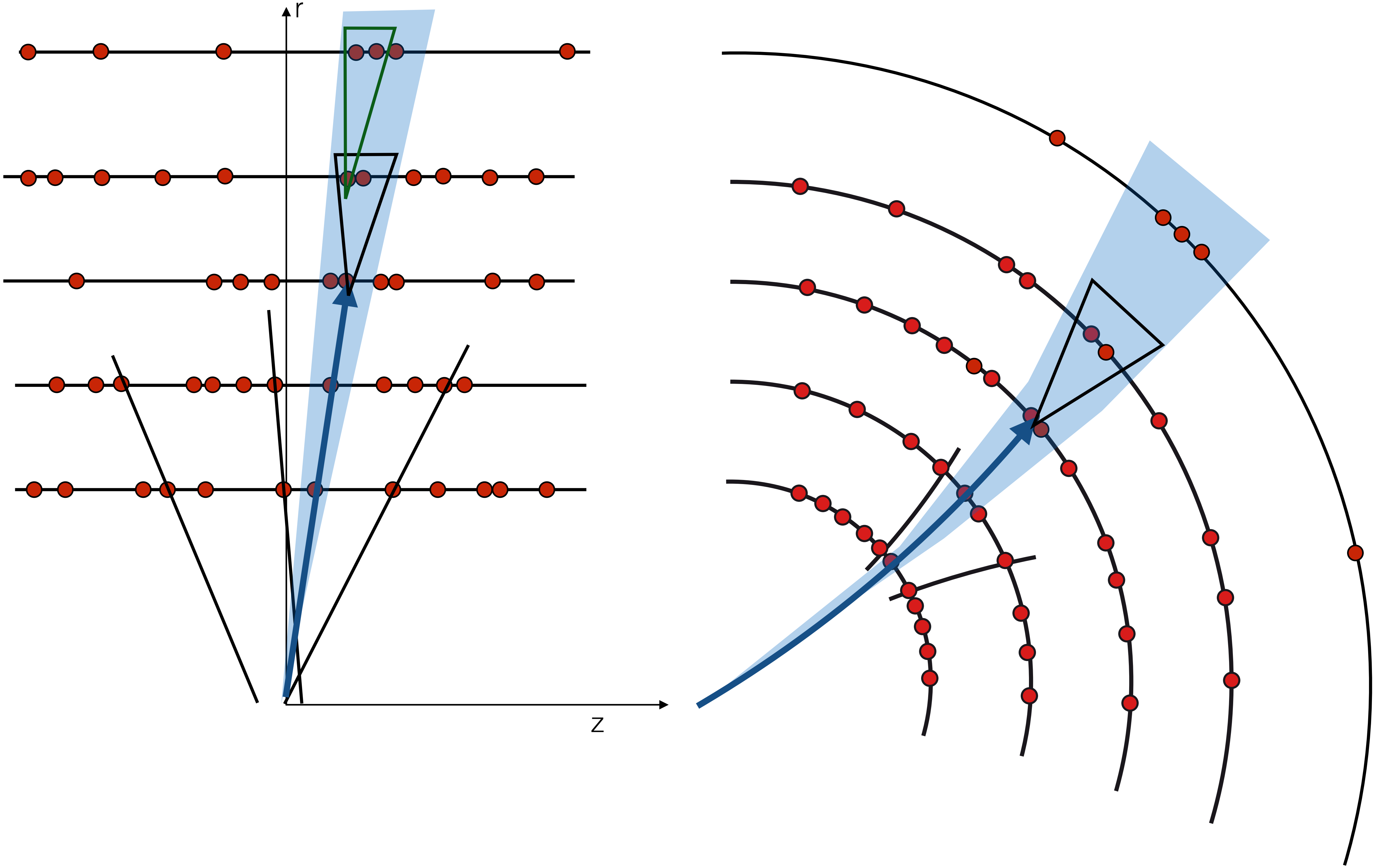
there may not always be one obvious path to be followed:



Track following combinatorial filter

Dense environments create problems for the progressive filter

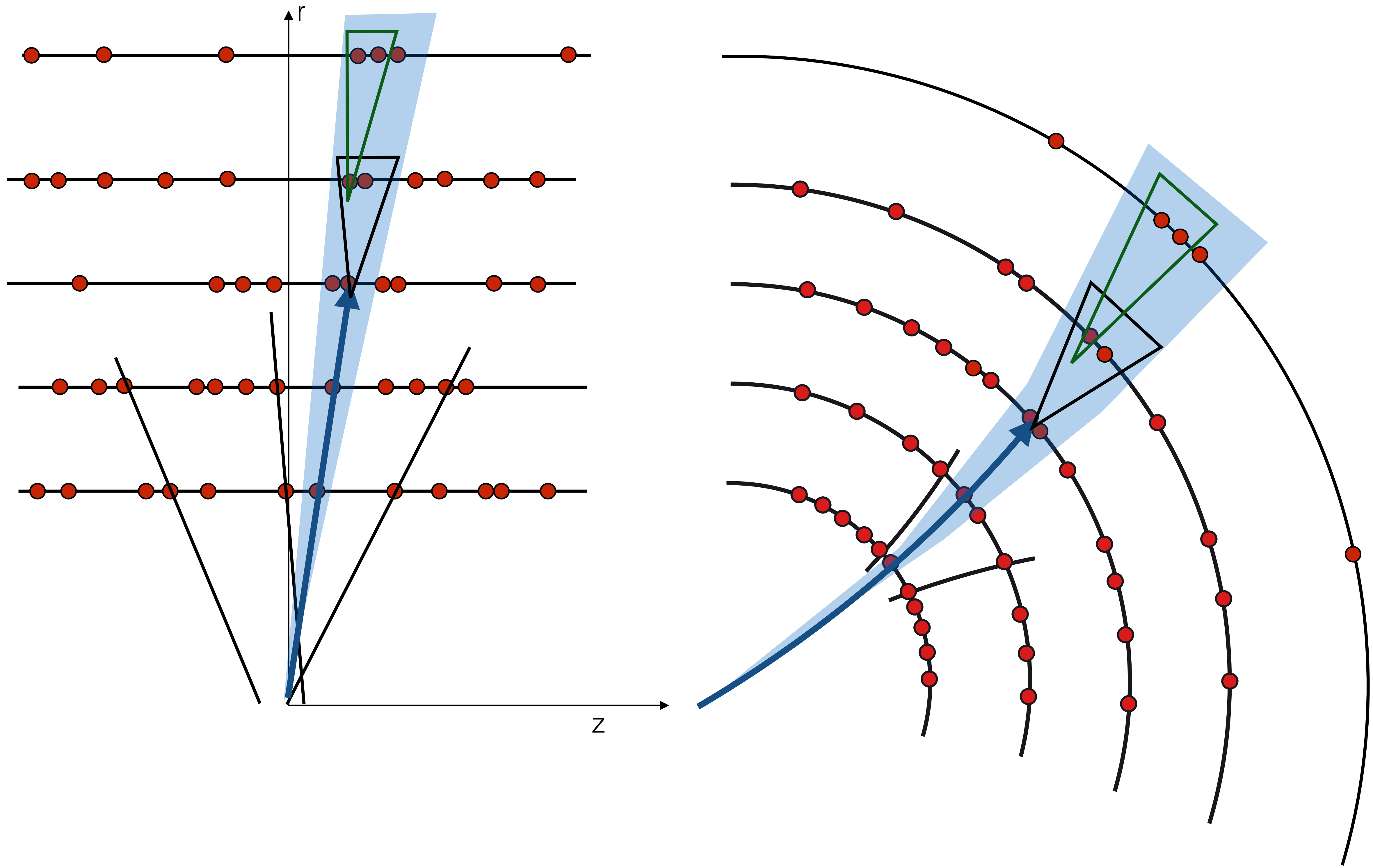
there may not always be one obvious path to be followed:



Track following combinatorial filter

Dense environments create problems for the progressive filter

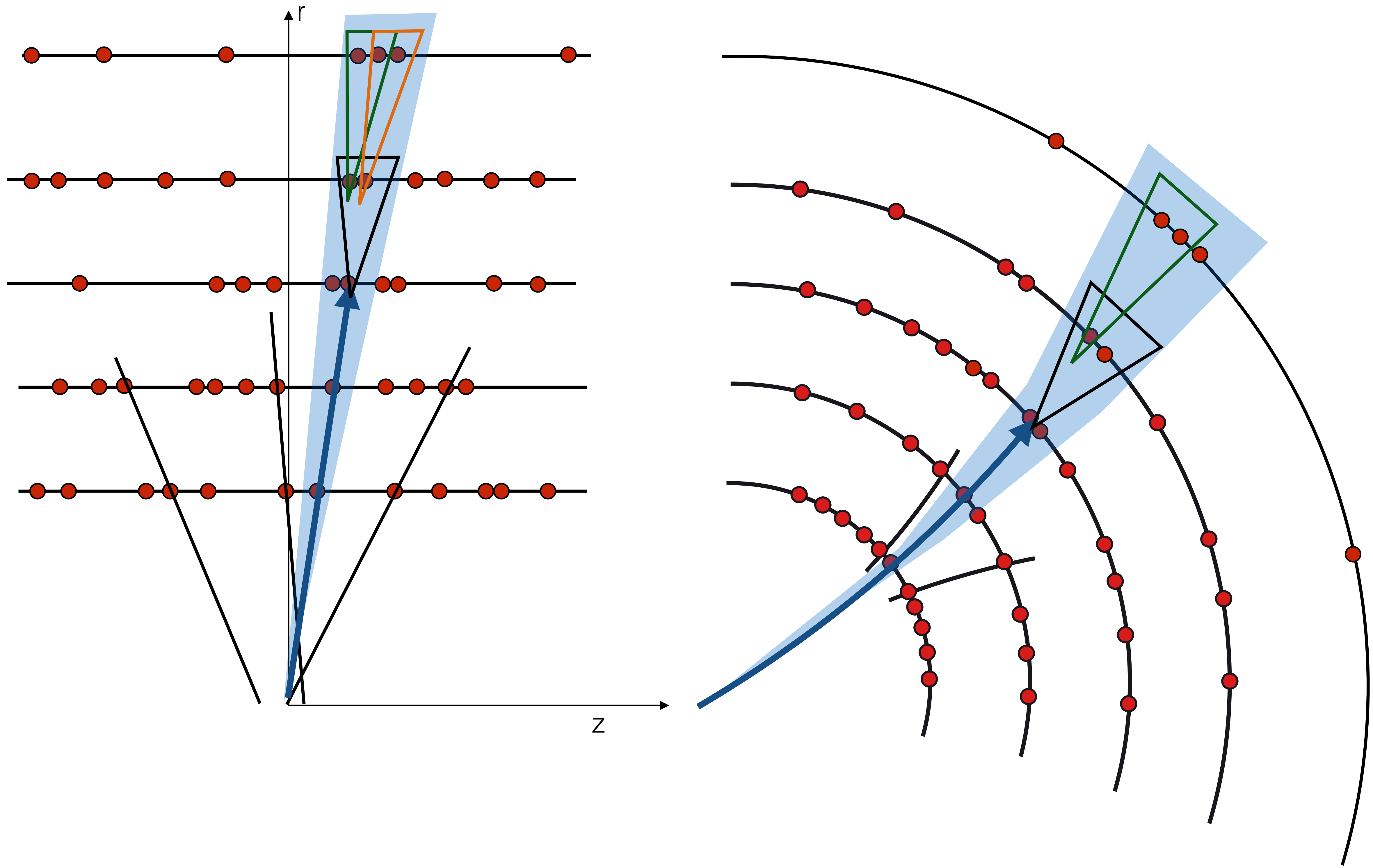
there may not always be one obvious path to be followed:



Track following combinatorial filter

Dense environments create problems for the progressive filter

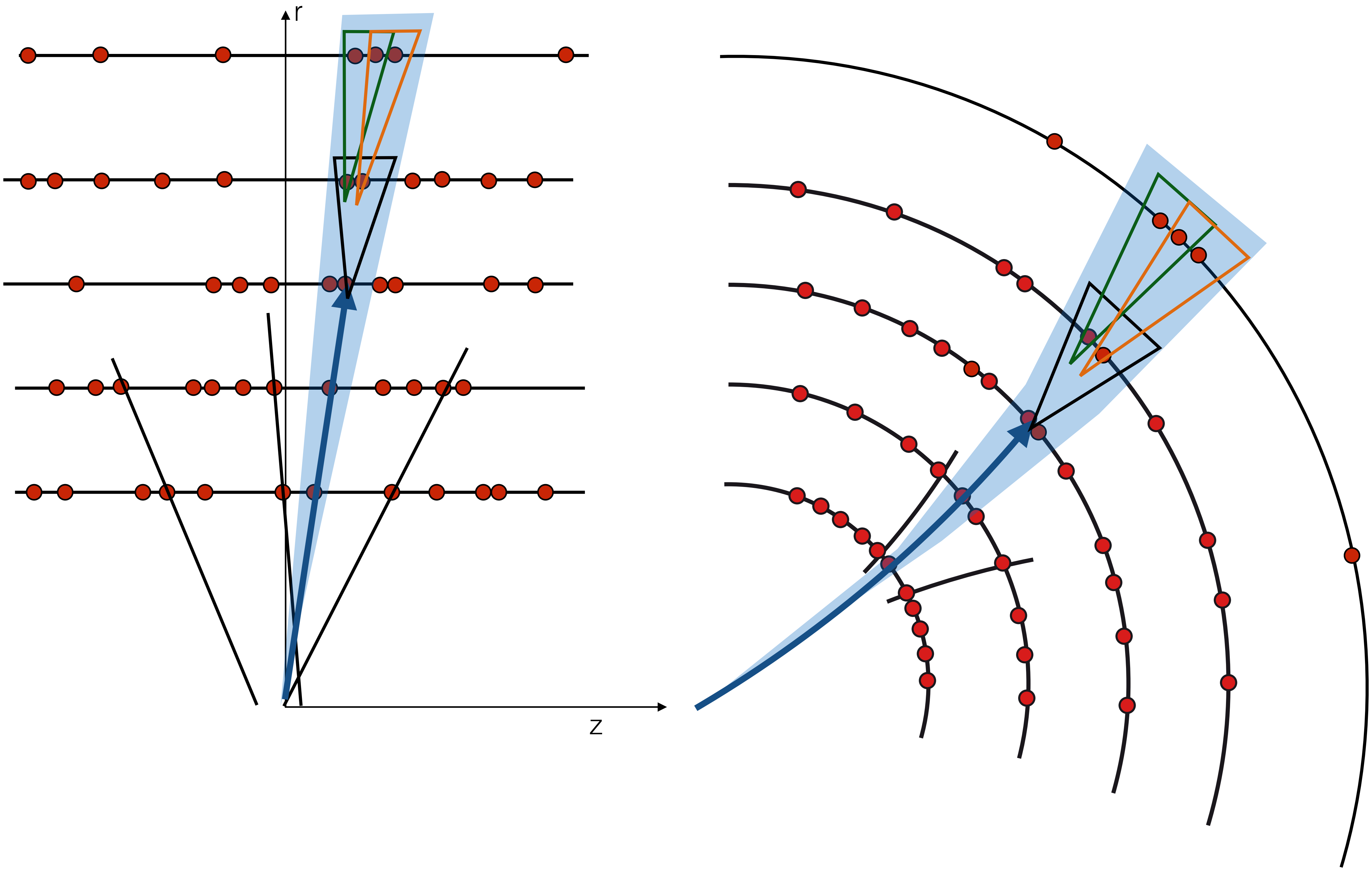
there may not always be one obvious path to be followed:



Track following combinatorial filter

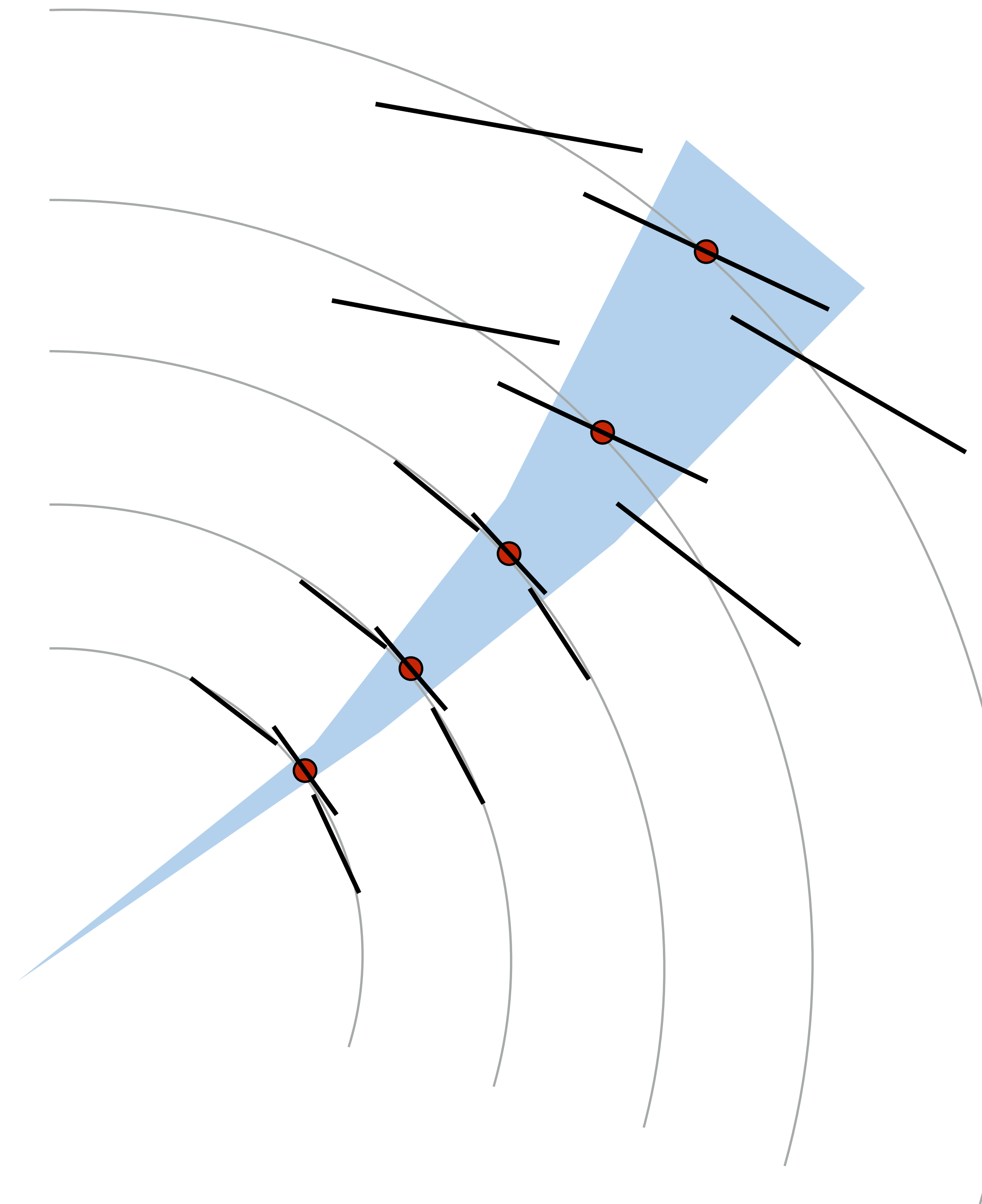
Dense environments create problems for the progressive filter

there may not always be one obvious path to be followed:



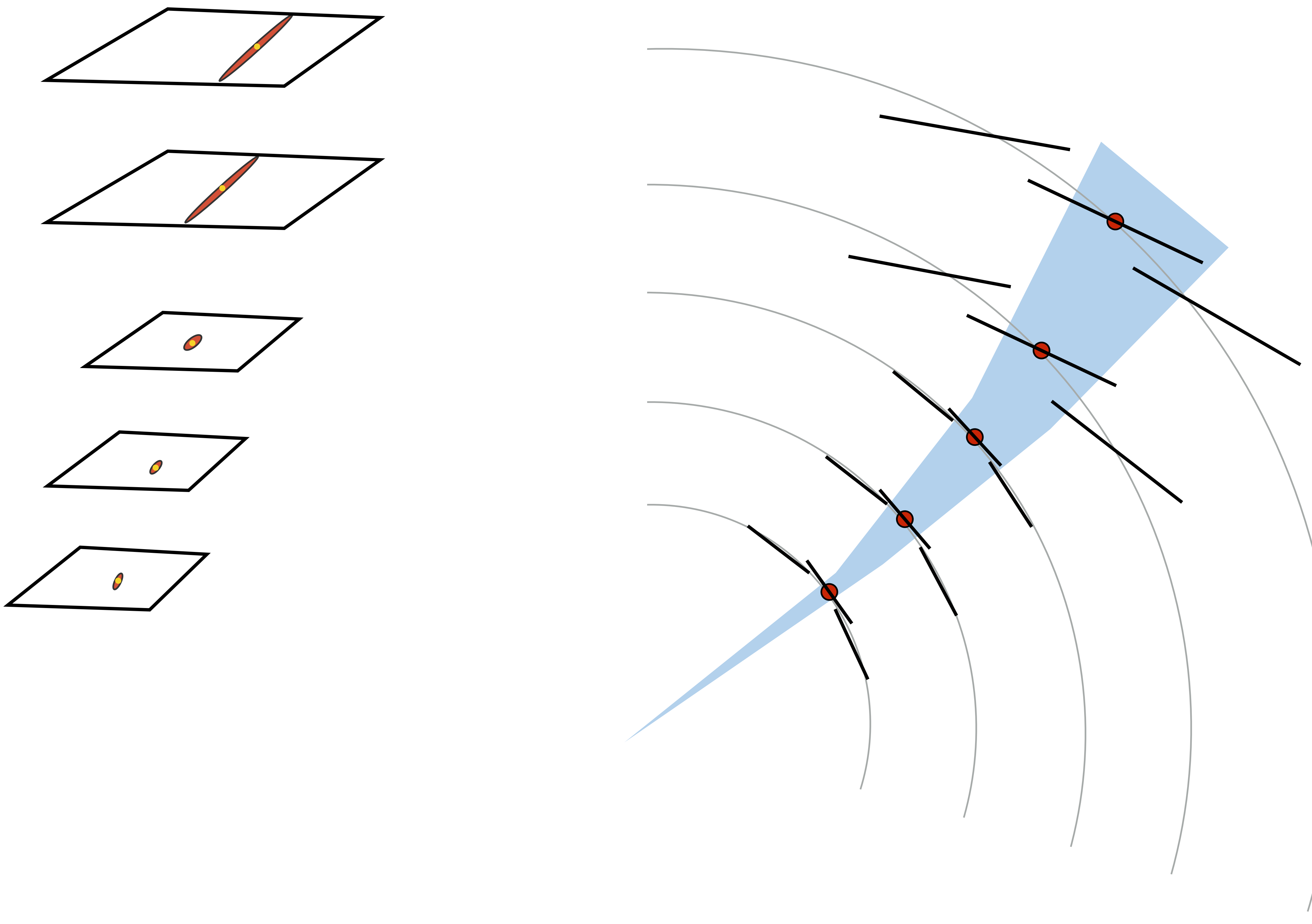
Track fitting parameter estimation

At the end we are only interested in the track parameters.



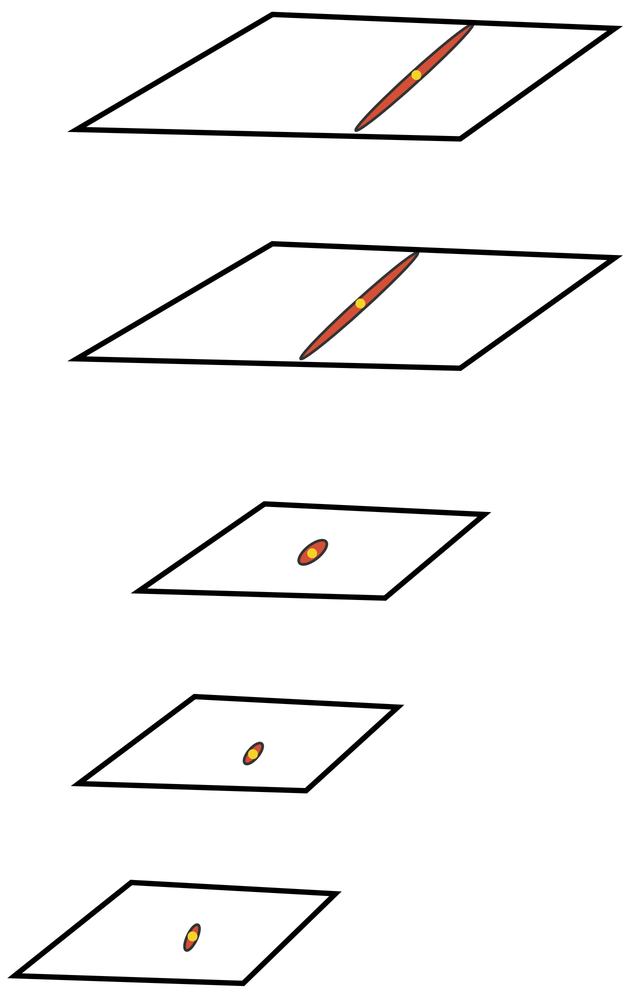
Track fitting parameter estimation

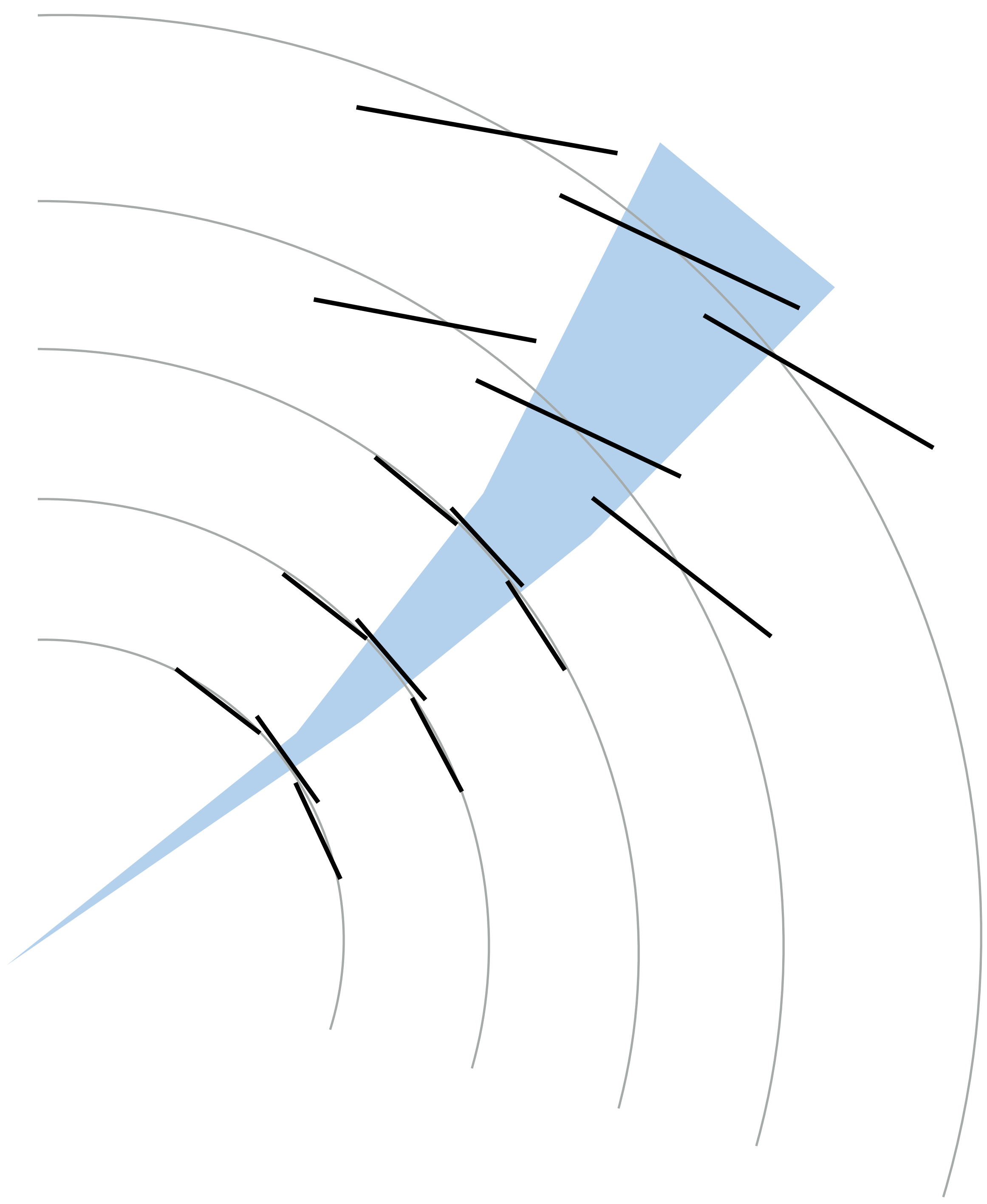
At the end we are only interested in the track parameters.



Track fitting parameter estimation

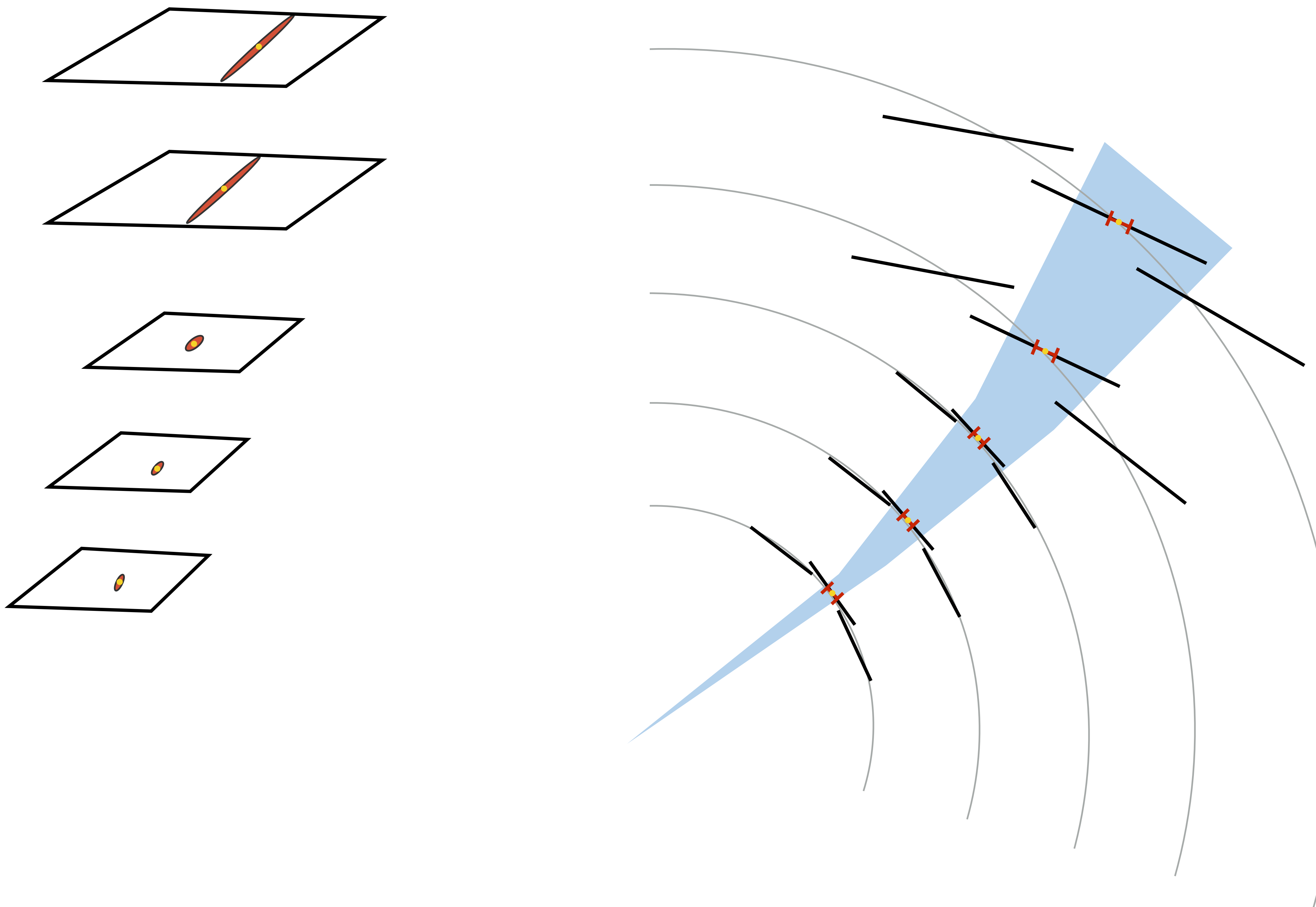
At the end we are only interested in the track parameters.





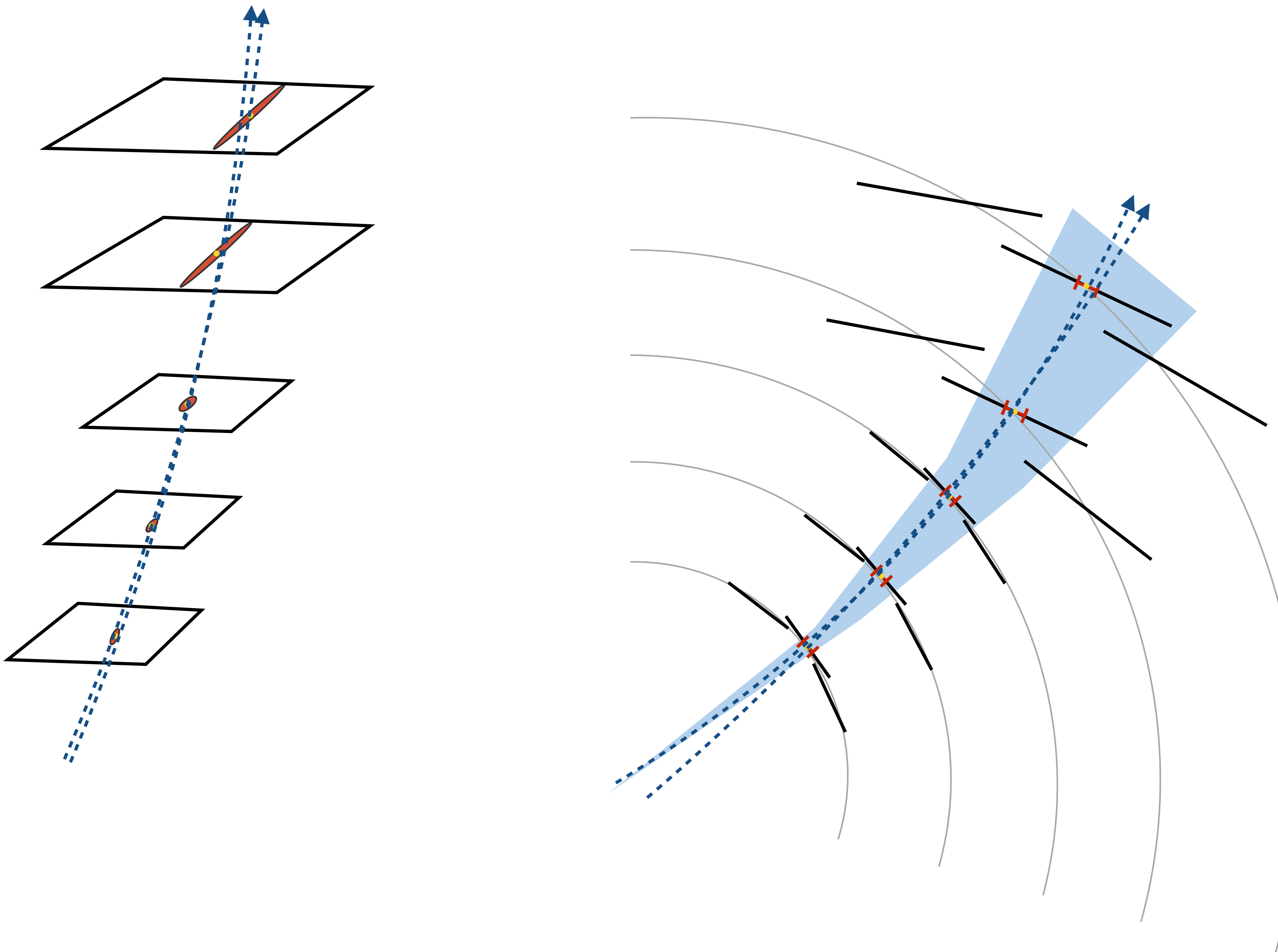
Track fitting parameter estimation

At the end we are only interested in the track parameters.



Track fitting parameter estimation

At the end we are only interested in the track parameters.



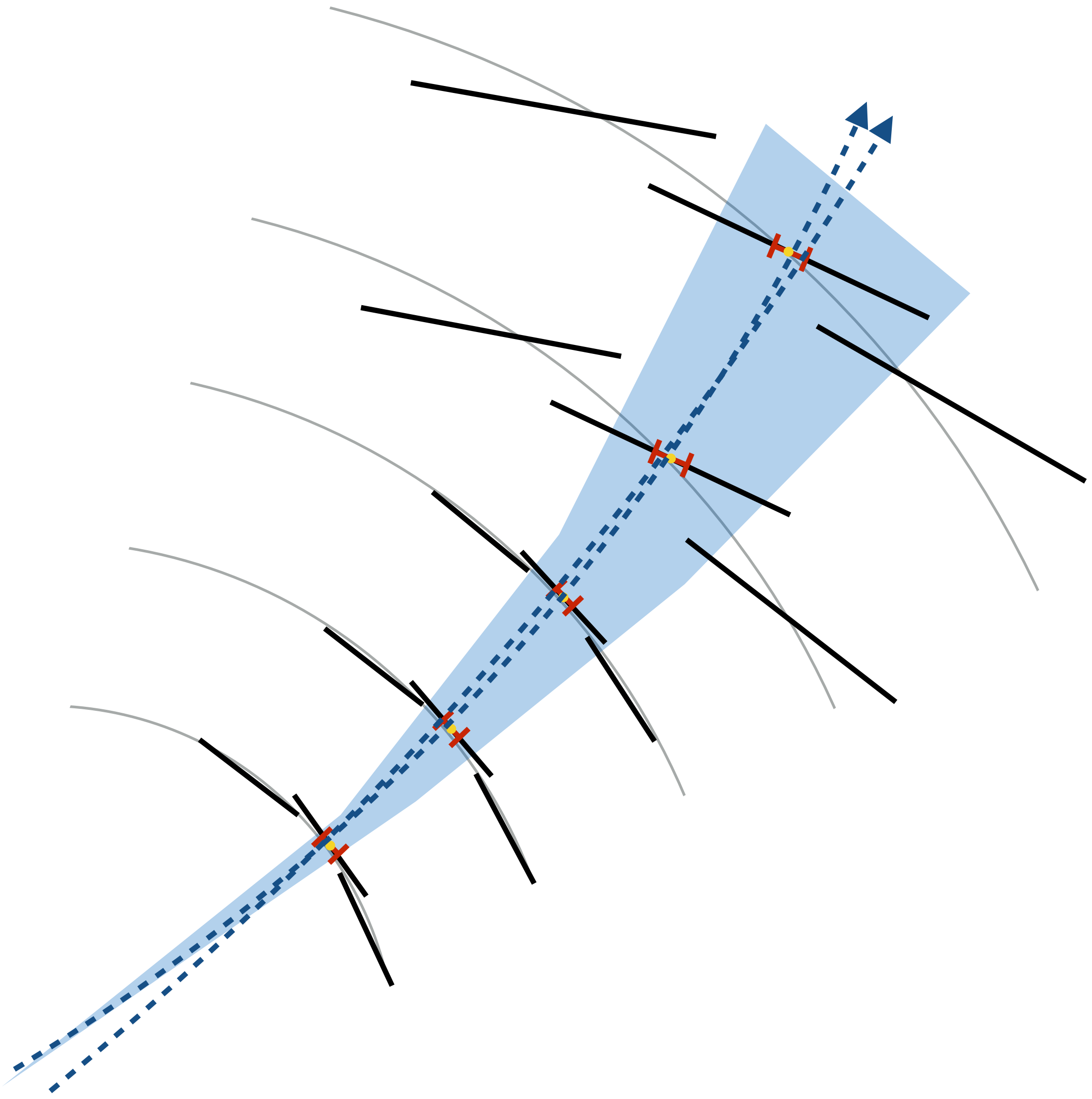
Track fitting global least squares fit

- a classical least squares estimator problem !

$$\chi^2 = \sum_k \Delta m_k^T G_K^{-1} \Delta m_k \quad \text{with} \quad \Delta m_k = m_k - d_k(\mathbf{q})$$ and $\boldsymbol{G_k}$ the covariance of measurement $\mathbf{m}_k$

d_k including transport of $\mathbf{q}$ to measurement layer k
and measurement mapping function

$$\boldsymbol{d}_k = \boldsymbol{h}_k \circ \boldsymbol{f}_{k|k-1} \circ \cdots \circ \boldsymbol{f}_{2|1} \circ \boldsymbol{f}_{1|0}$$



Track fitting global least squares fit

- a classical least squares estimator problem !

$$\chi^2 = \sum_k \Delta m_k^T G_K^{-1} \Delta m_k$$ with $$\Delta m_k = m_k - d_k(\mathbf{q})$$ and $\boldsymbol{G_k}$ the covariance of measurement $\mathbf{m}_k$

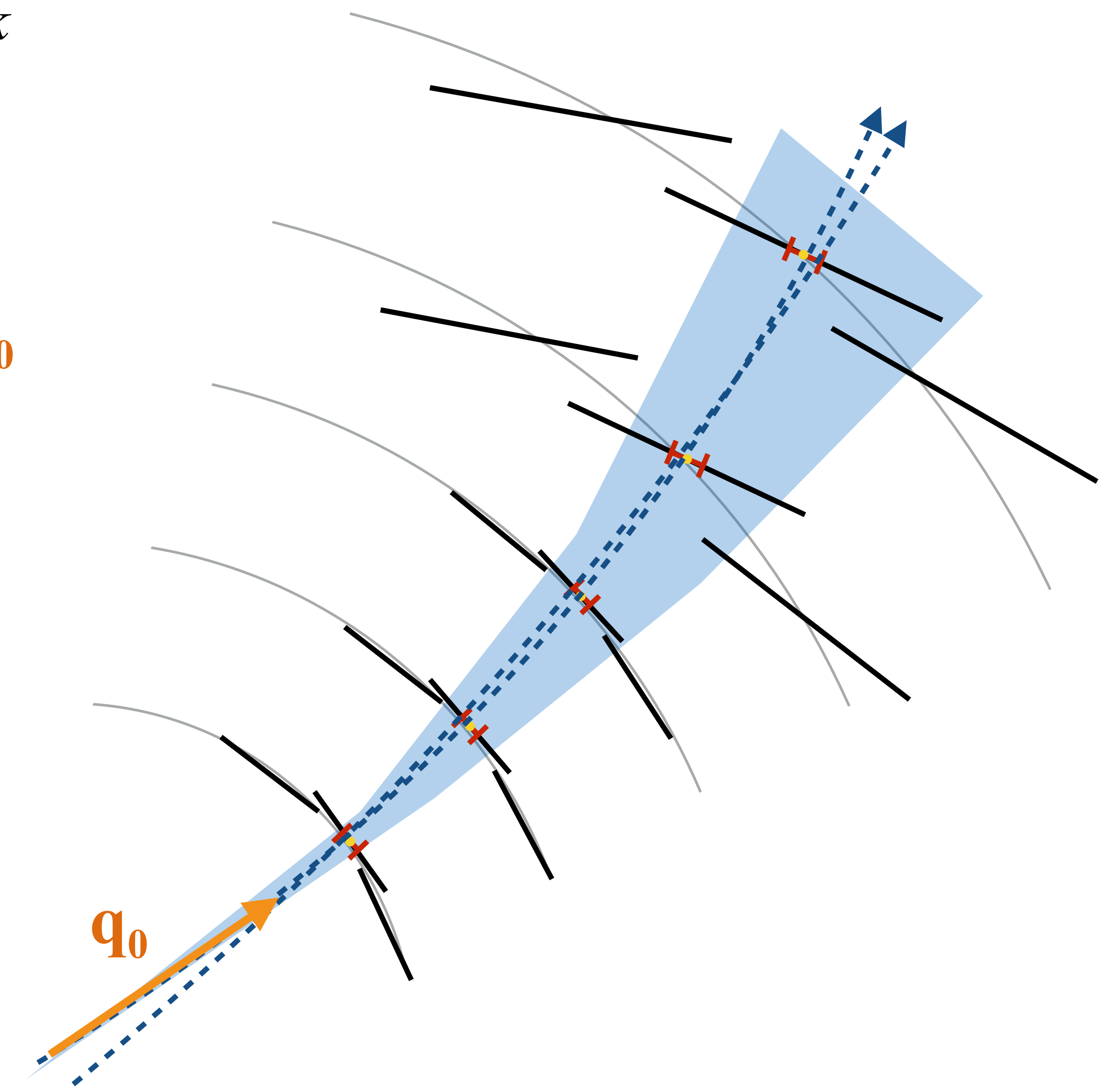
d_k including transport of $\mathbf{q}$ to measurement layer k
and measurement mapping function

$$\boldsymbol{d}_k = \boldsymbol{h}_k \circ \boldsymbol{f}_{k|k-1} \circ \cdots \circ \boldsymbol{f}_{2|1} \circ \boldsymbol{f}_{1|0}$$

linearise the problem, starting from an initial state $\mathbf{q_0}$

$$d_k(\mathbf{q_0} + \delta\mathbf{q}) \cong d_k(\mathbf{q_0}) + D_k \cdot \delta\mathbf{q}$$

with Jacobian $$\boldsymbol{D}_k = \boldsymbol{H}_k \boldsymbol{F}_{k|k-1} \cdots \boldsymbol{F}_{2|1} \boldsymbol{F}_{1|0}$$



Track fitting global least squares fit

- a classical least squares estimator problem !

$$\chi^2 = \sum_k \Delta m_k^T G_K^{-1} \Delta m_k \quad \text{with} \quad \Delta m_k = m_k - d_k(\mathbf{q})$$ and $\boldsymbol{G_k}$ the covariance of measurement $\mathbf{m}_k$

d_k including transport of $\mathbf{q}$ to measurement layer k
and measurement mapping function

$$\boldsymbol{d}_k = \boldsymbol{h}_k \circ \boldsymbol{f}_{k|k-1} \circ \cdots \circ \boldsymbol{f}_{2|1} \circ \boldsymbol{f}_{1|0}$$

linearise the problem, starting from an initial state $\mathbf{q_0}$

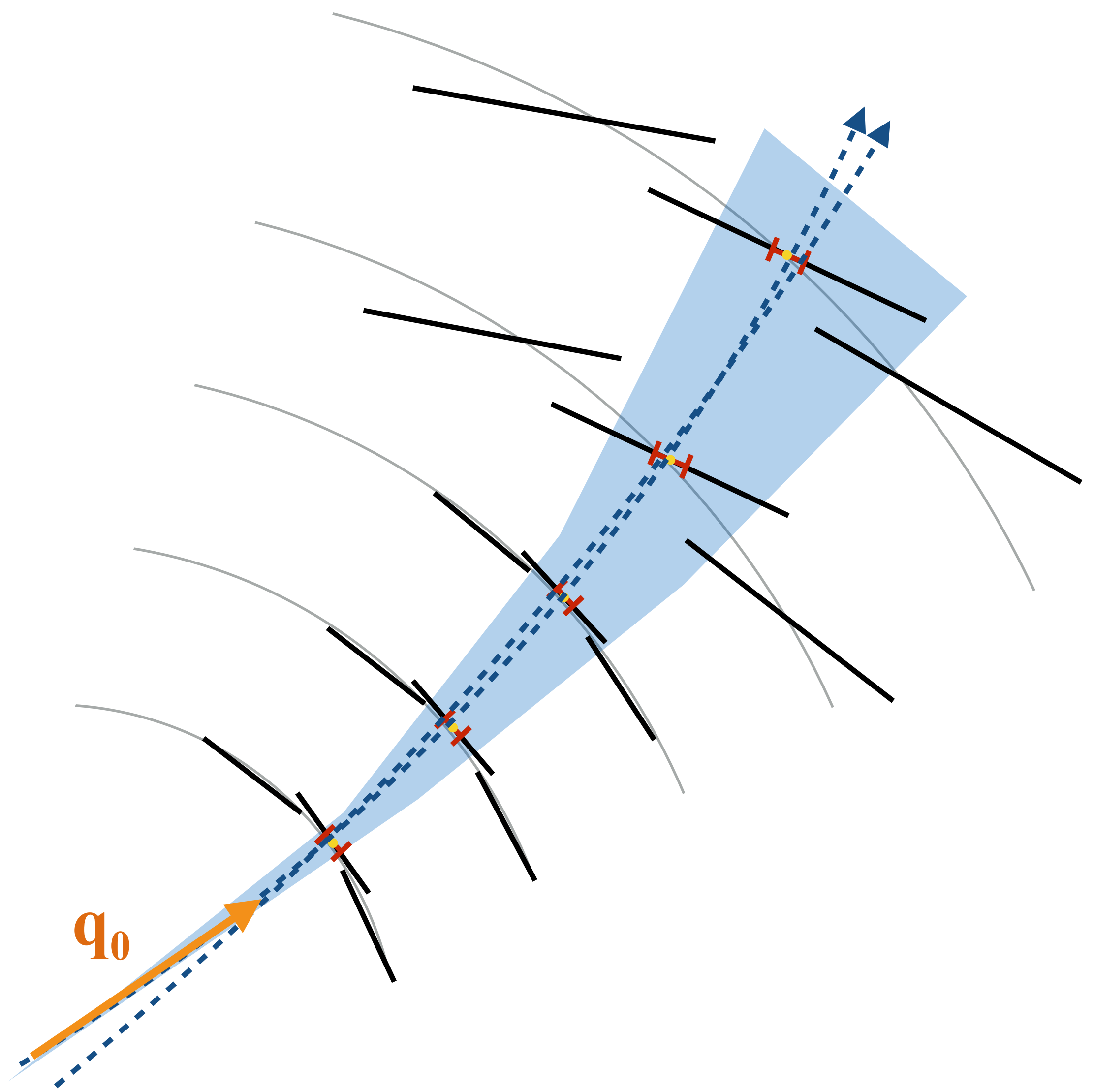
$$d_k(\mathbf{q_0} + \delta\mathbf{q}) \cong d_k(\mathbf{q_0}) + D_k \cdot \delta\mathbf{q}$$

with Jacobian $$\boldsymbol{D}_k = \boldsymbol{H}_k \boldsymbol{F}_{k|k-1} \cdots \boldsymbol{F}_{2|1} \boldsymbol{F}_{1|0}$$

find the global minimum: $$\frac{\partial \chi^2}{\partial \mathbf{q}} \stackrel{!}{=} 0$$

$$\partial\mathbf{q} = \left(\sum_k D_k^T G_k^{-1} D_k\right)^{-1} \sum_k D_k^T G_k^{-1}\left(m_k - d_k(\mathbf{q_0})\right)$$

$$C = \left(\sum_k D_k^T G_k^{-1} D_k\right)^{-1}$$



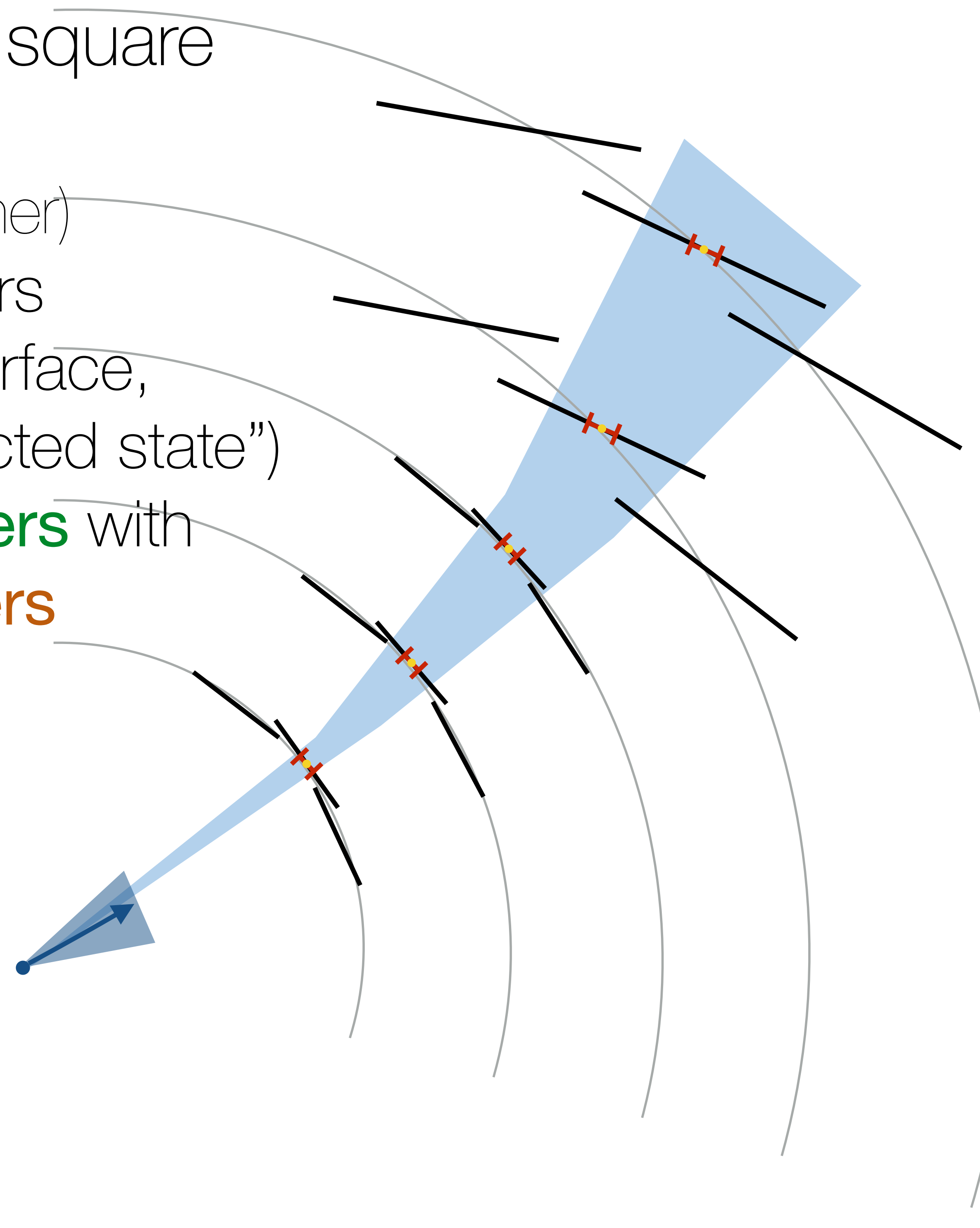
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with measurement to **updated parameters** ("filtered state")



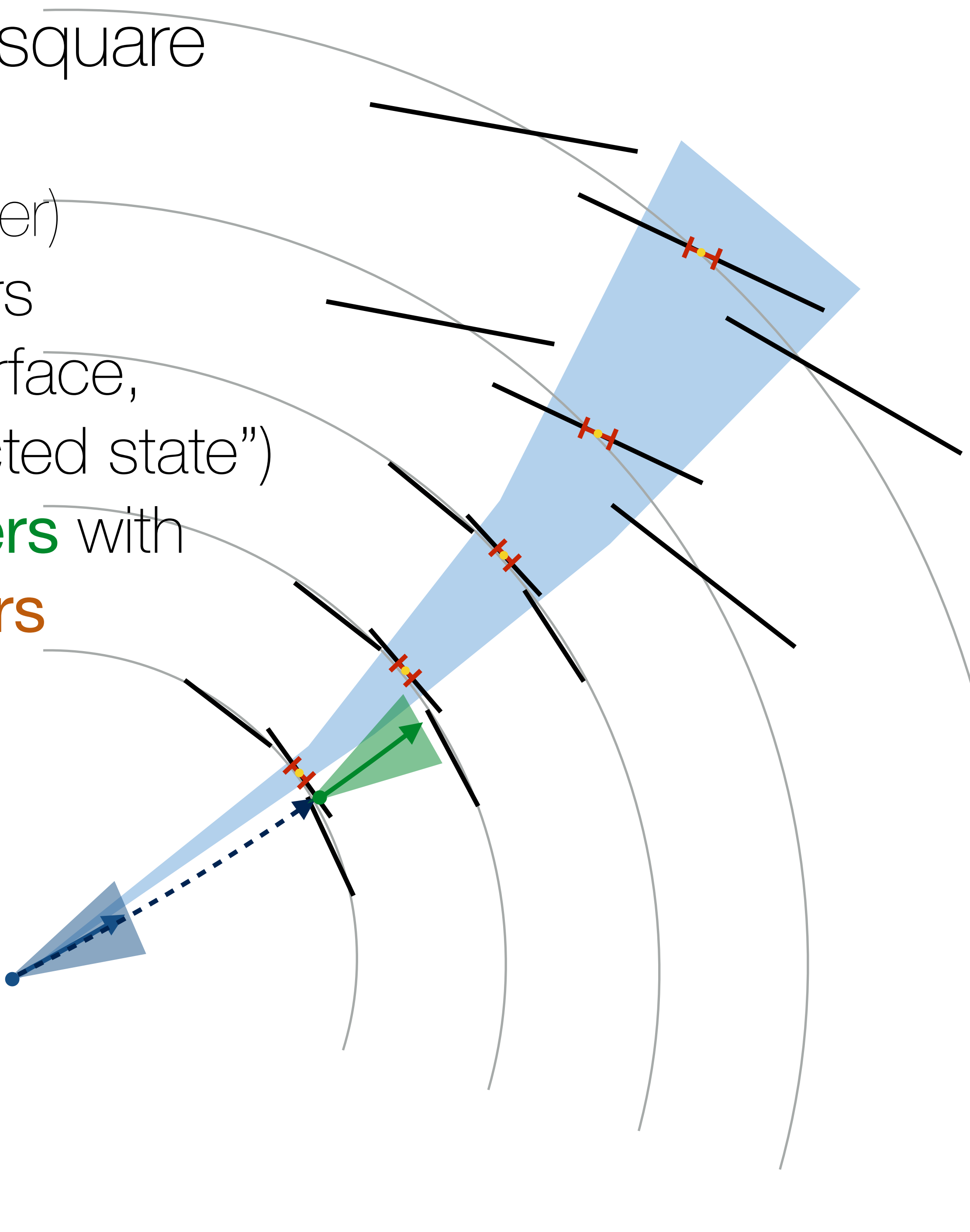
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



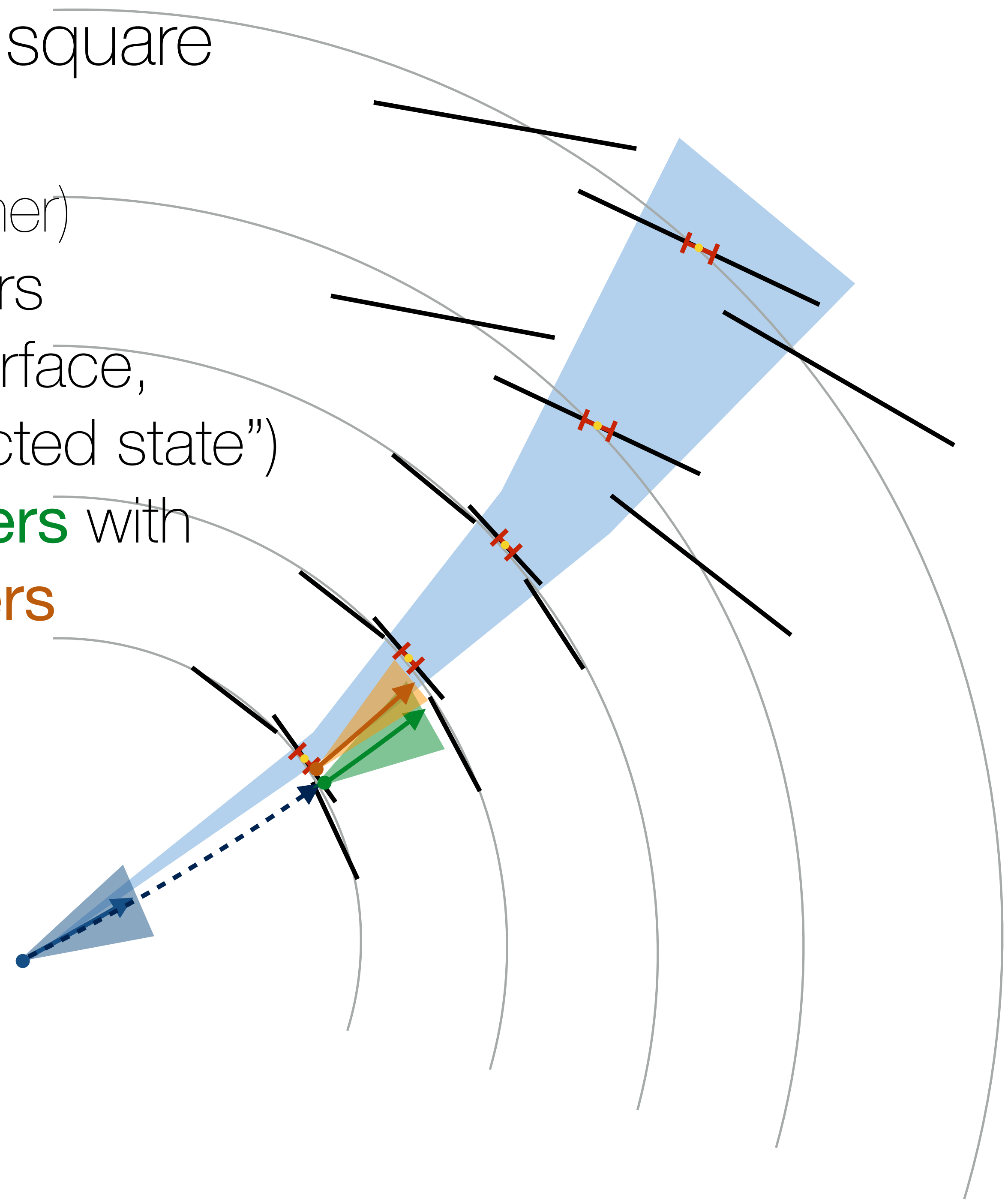
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



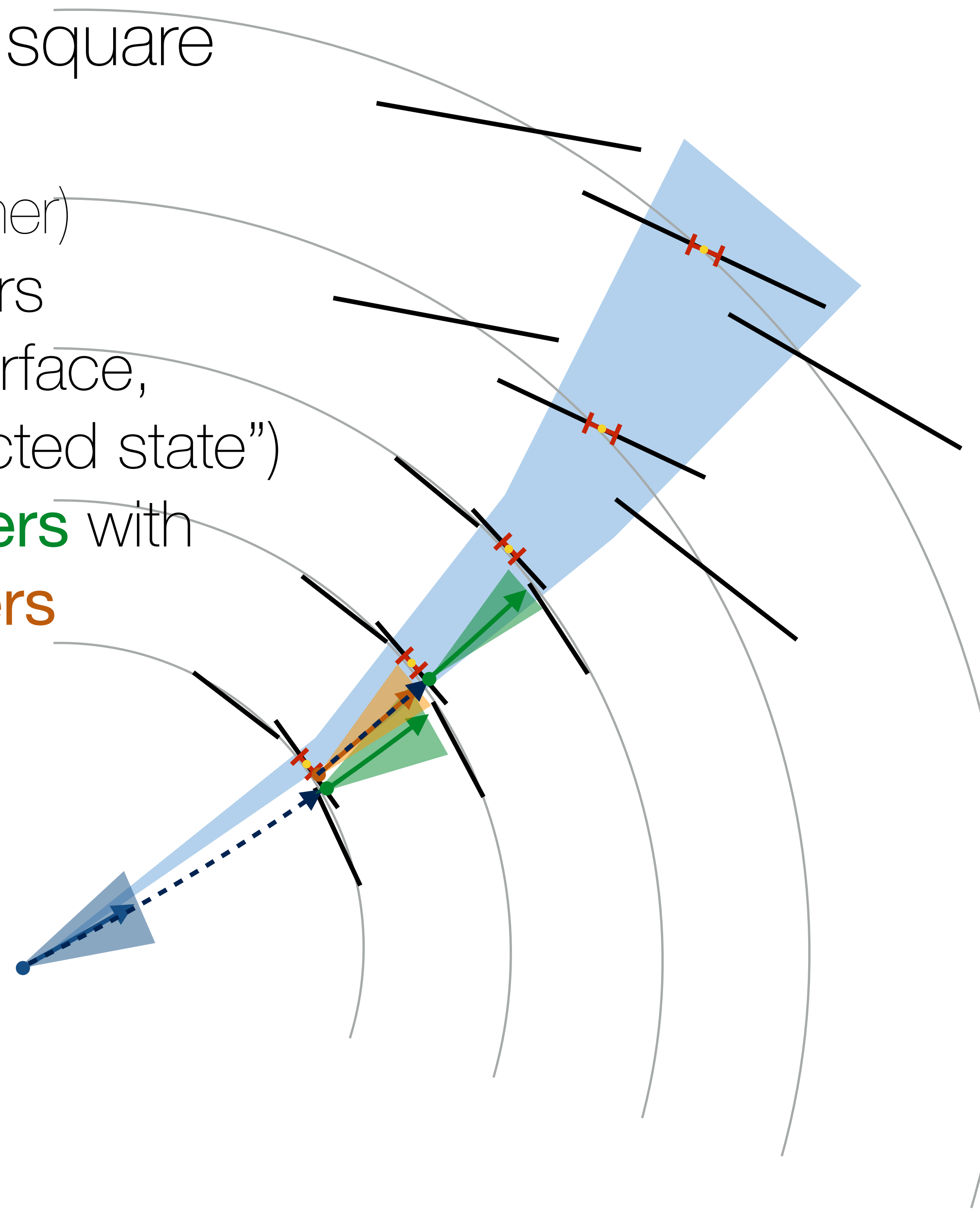
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** (“predicted state”)
- combine/update **predicted parameters** with **measurement** to **updated parameters** (“filtered state”)



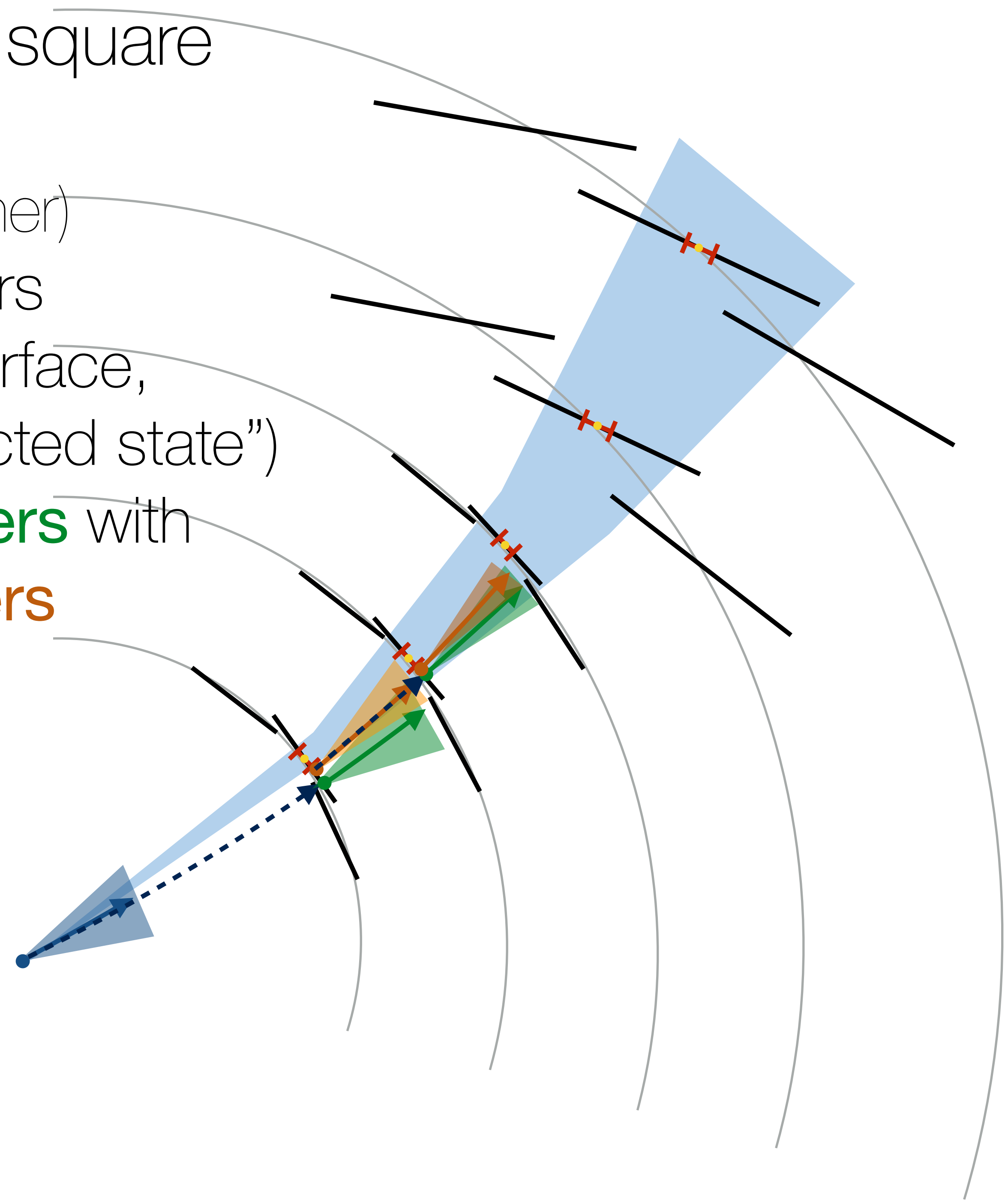
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



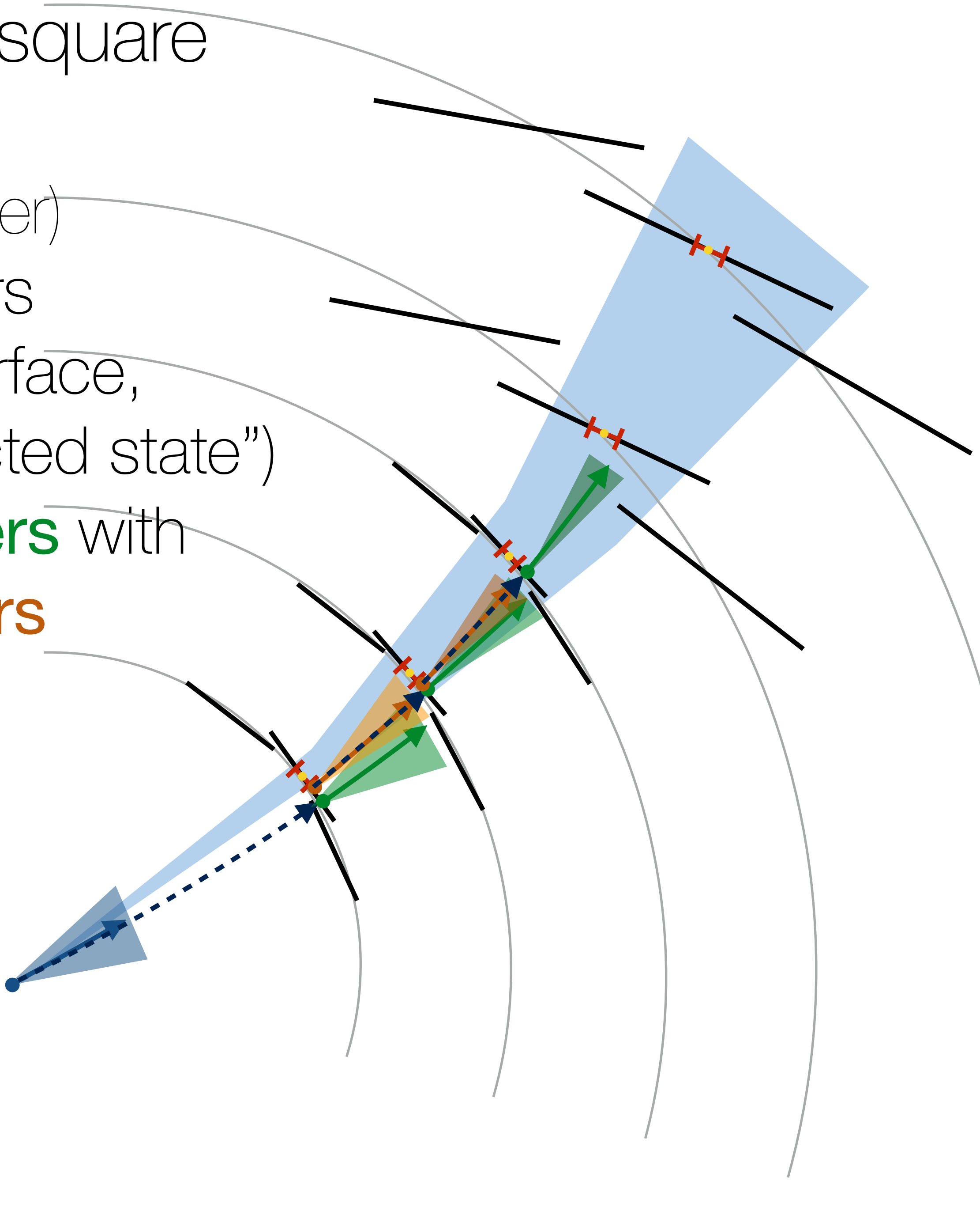
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



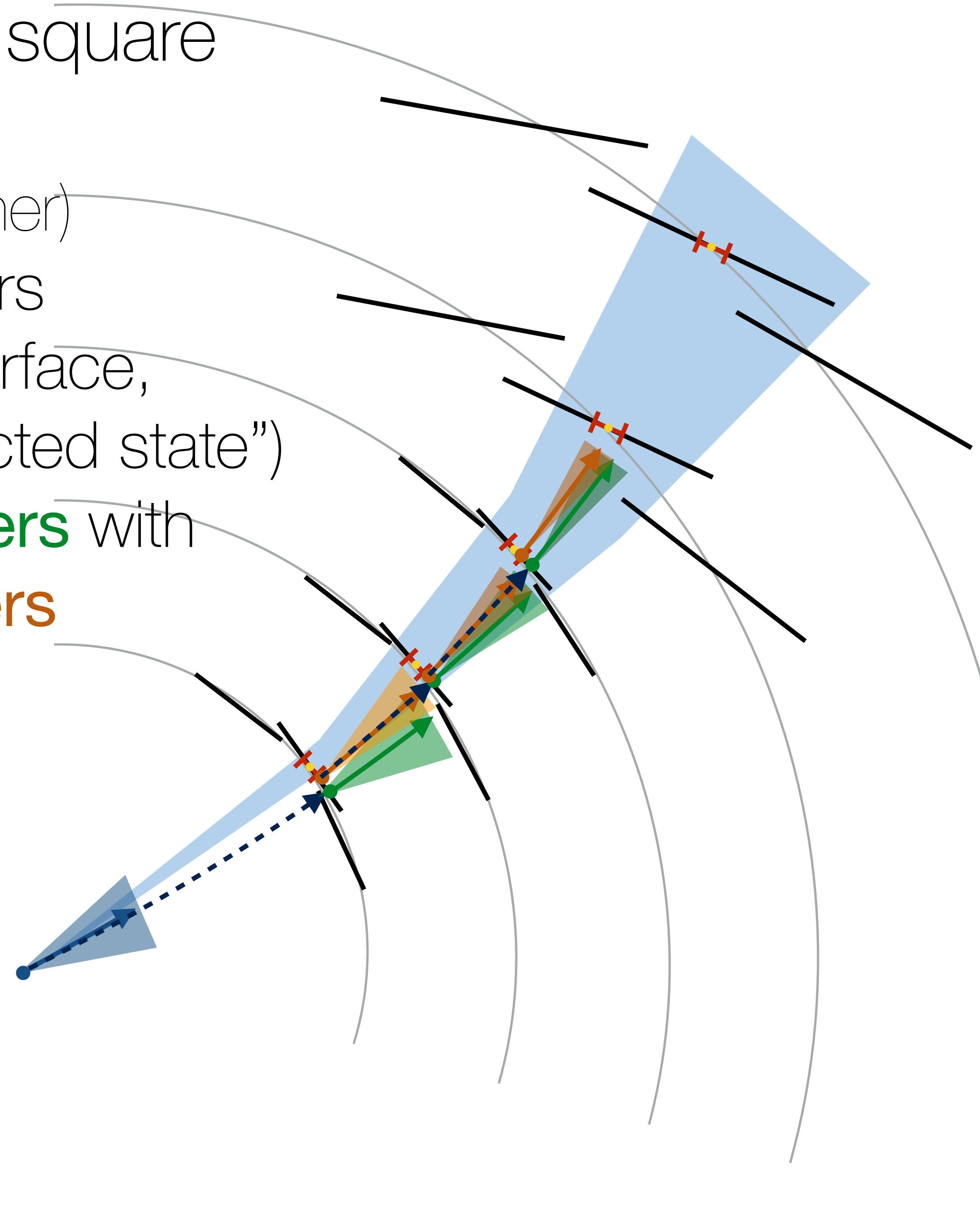
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



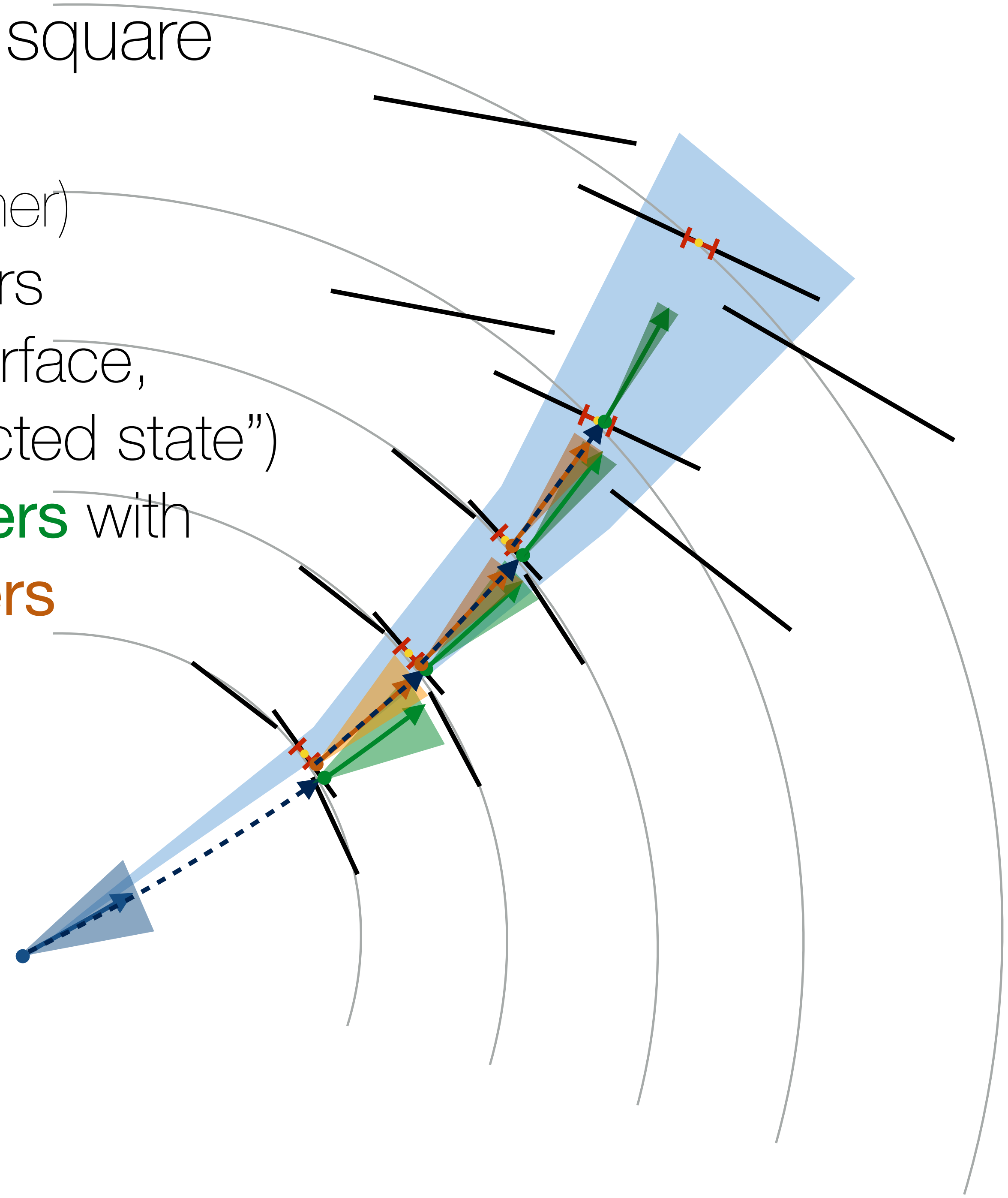
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



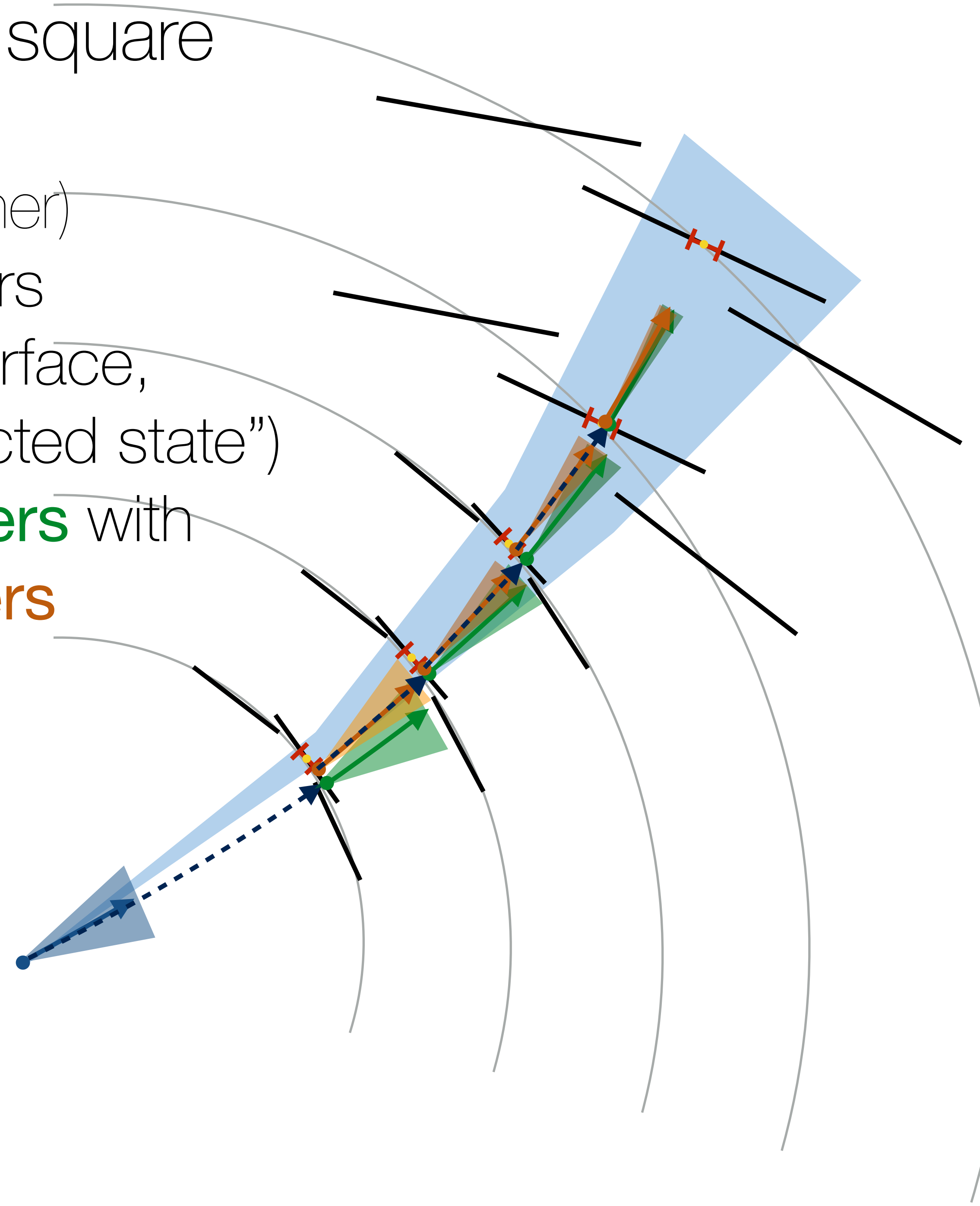
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

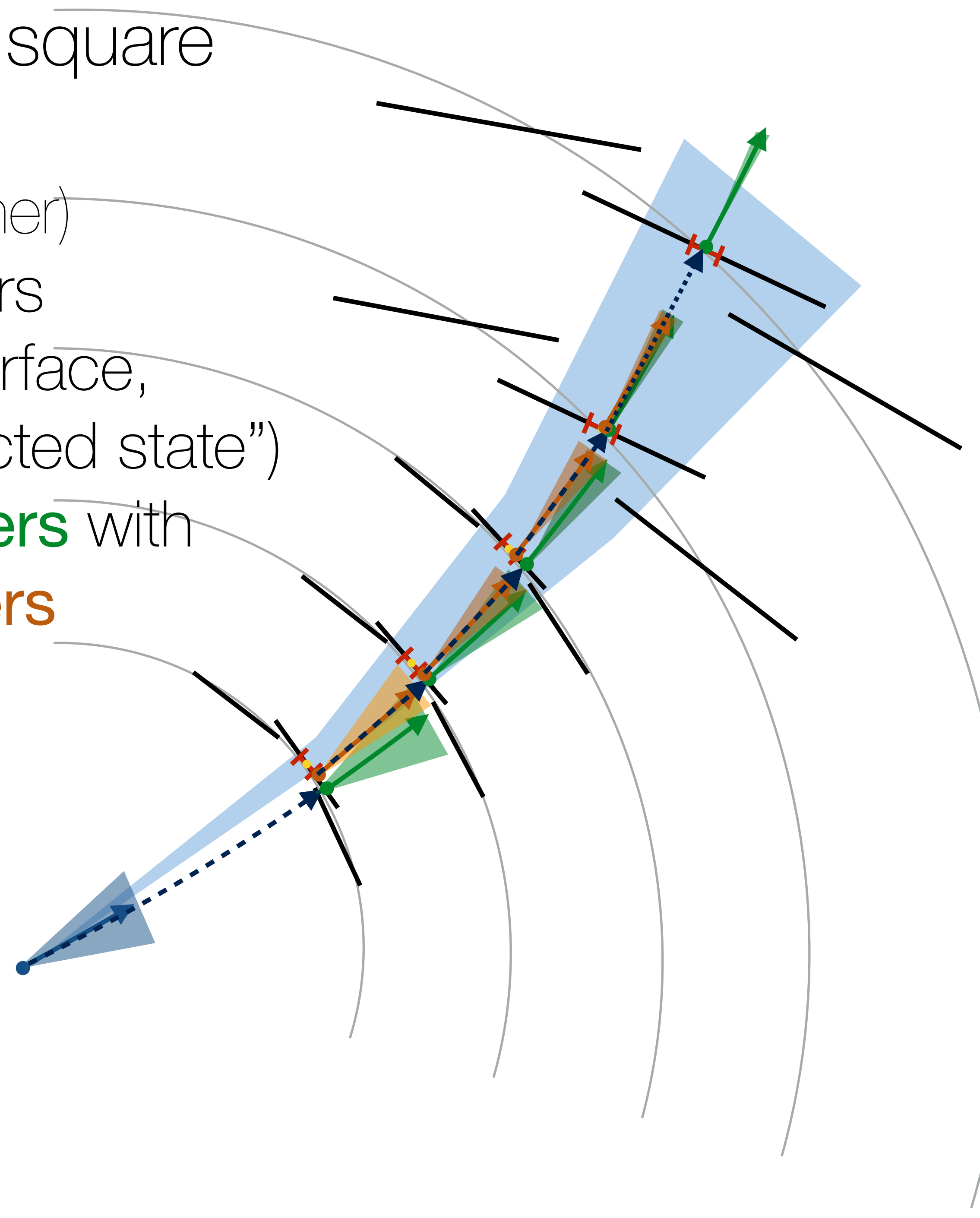
- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion
- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation
- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



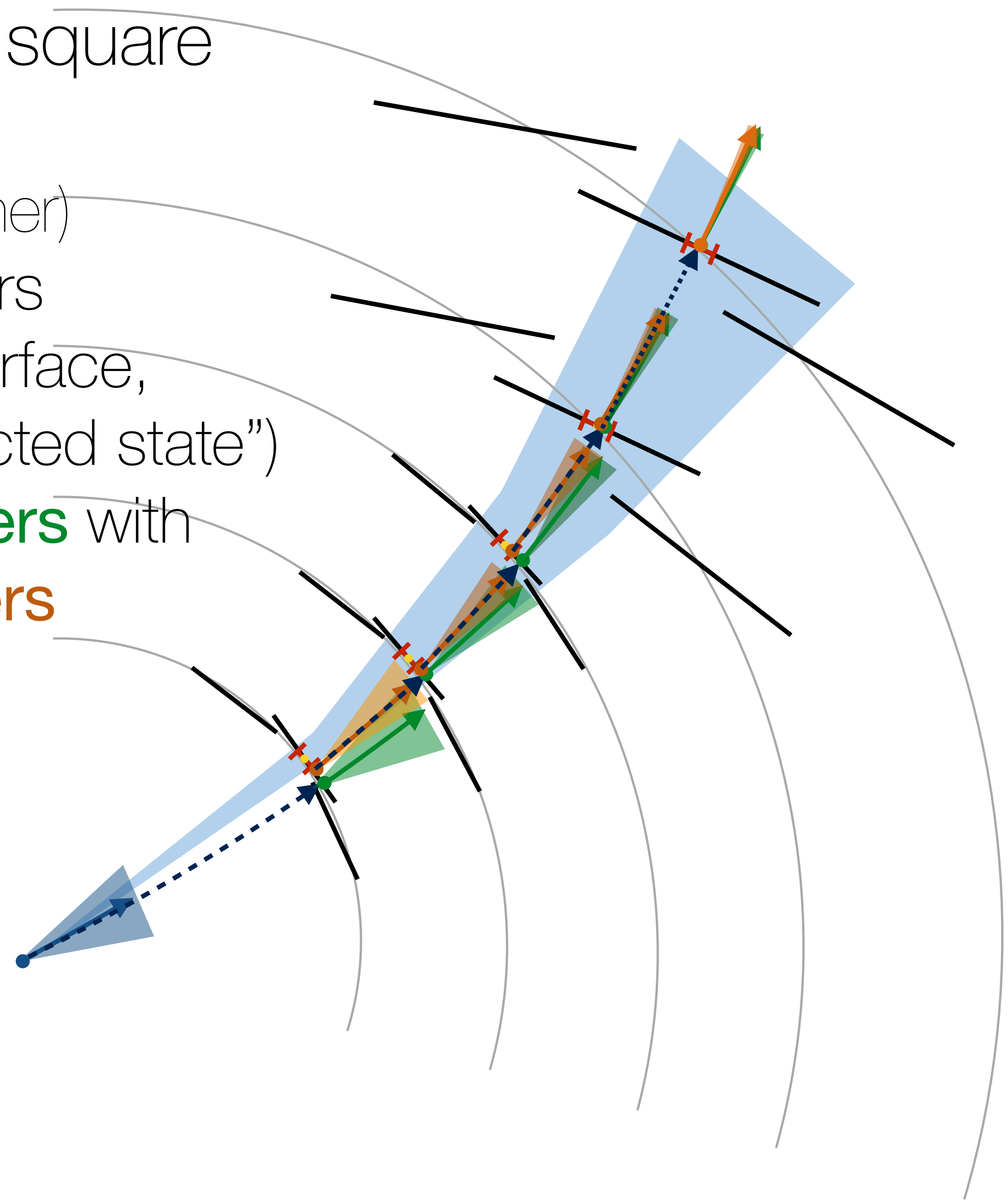
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



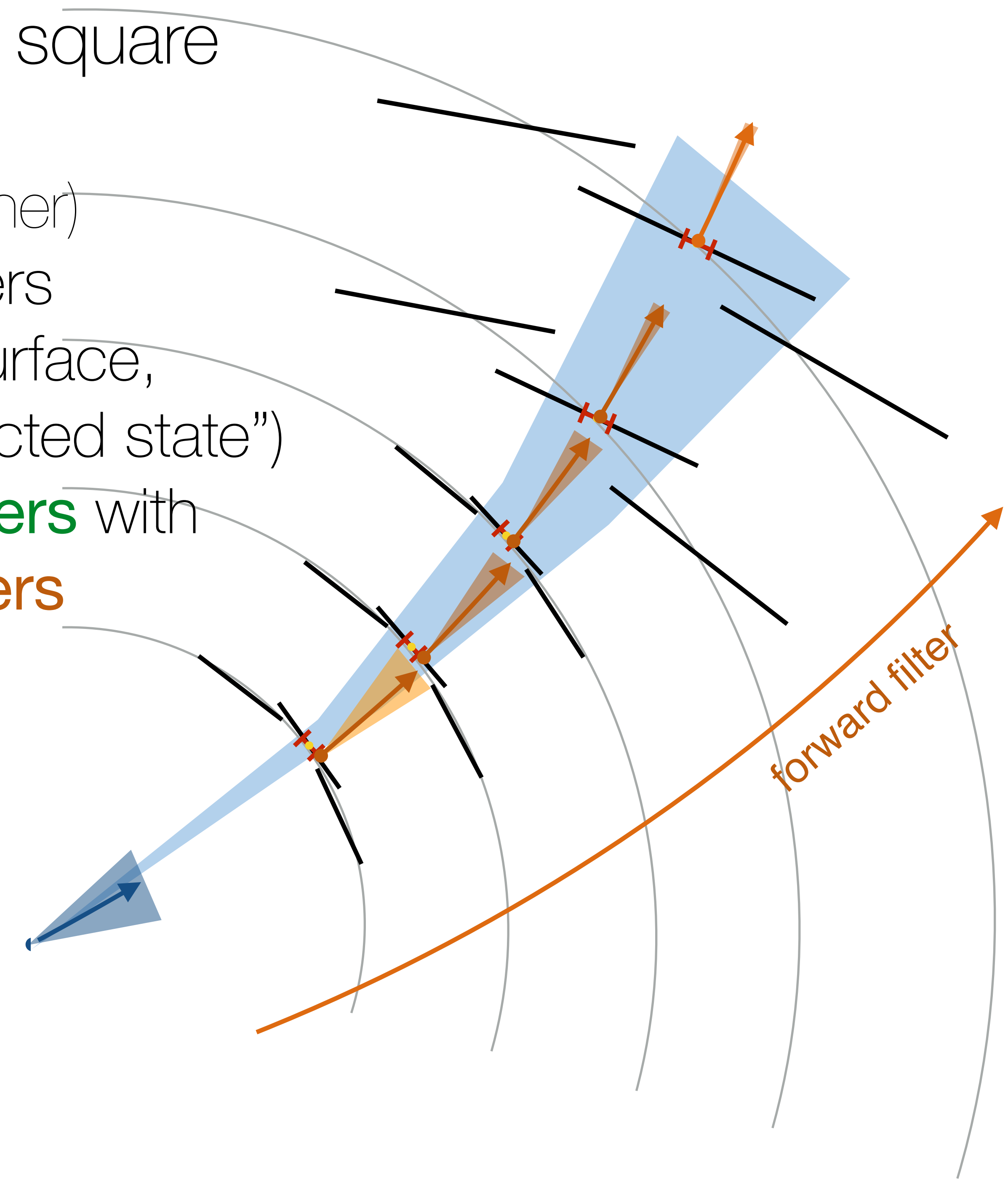
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



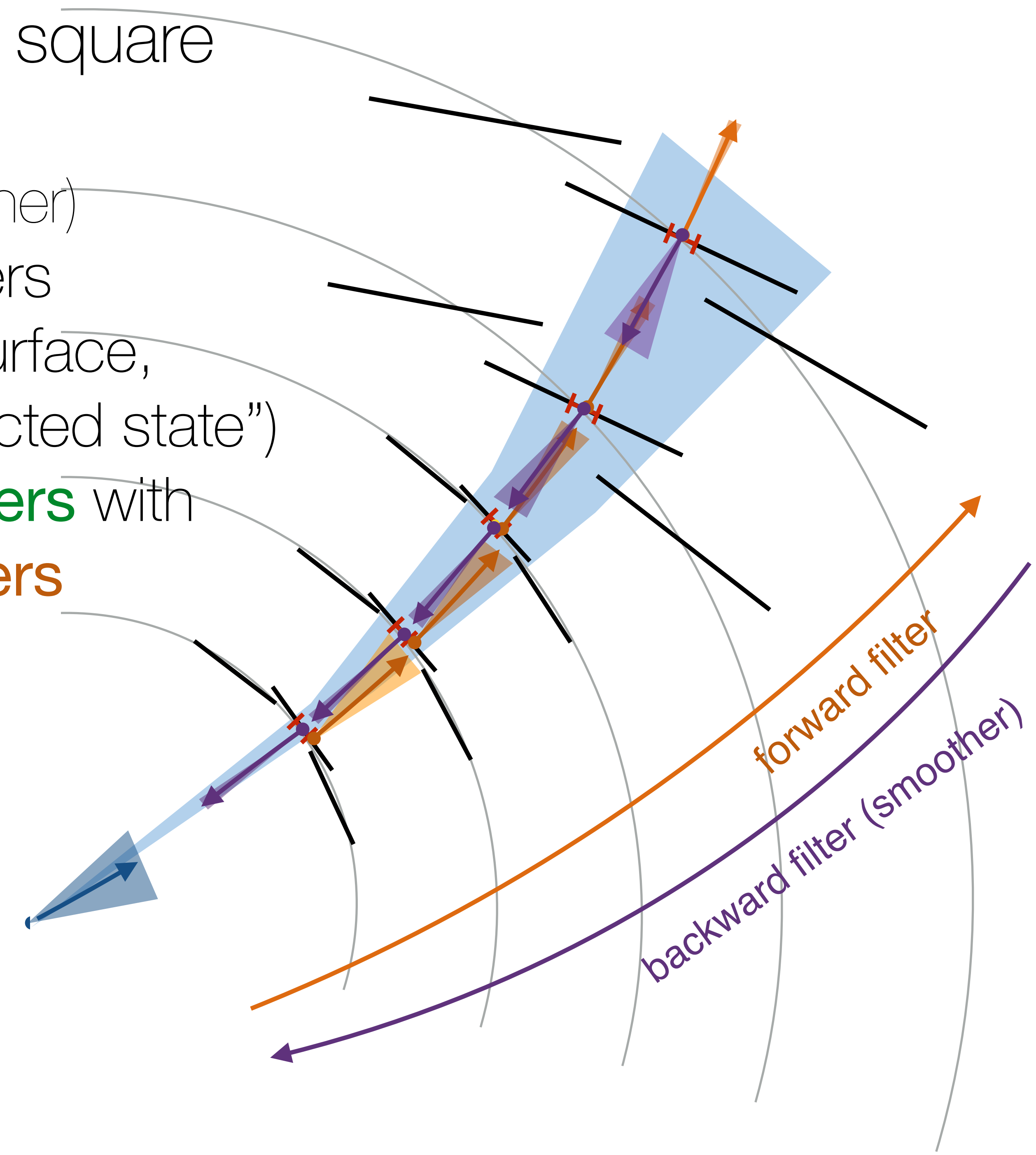
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with measurement to updated parameters ("filtered state")



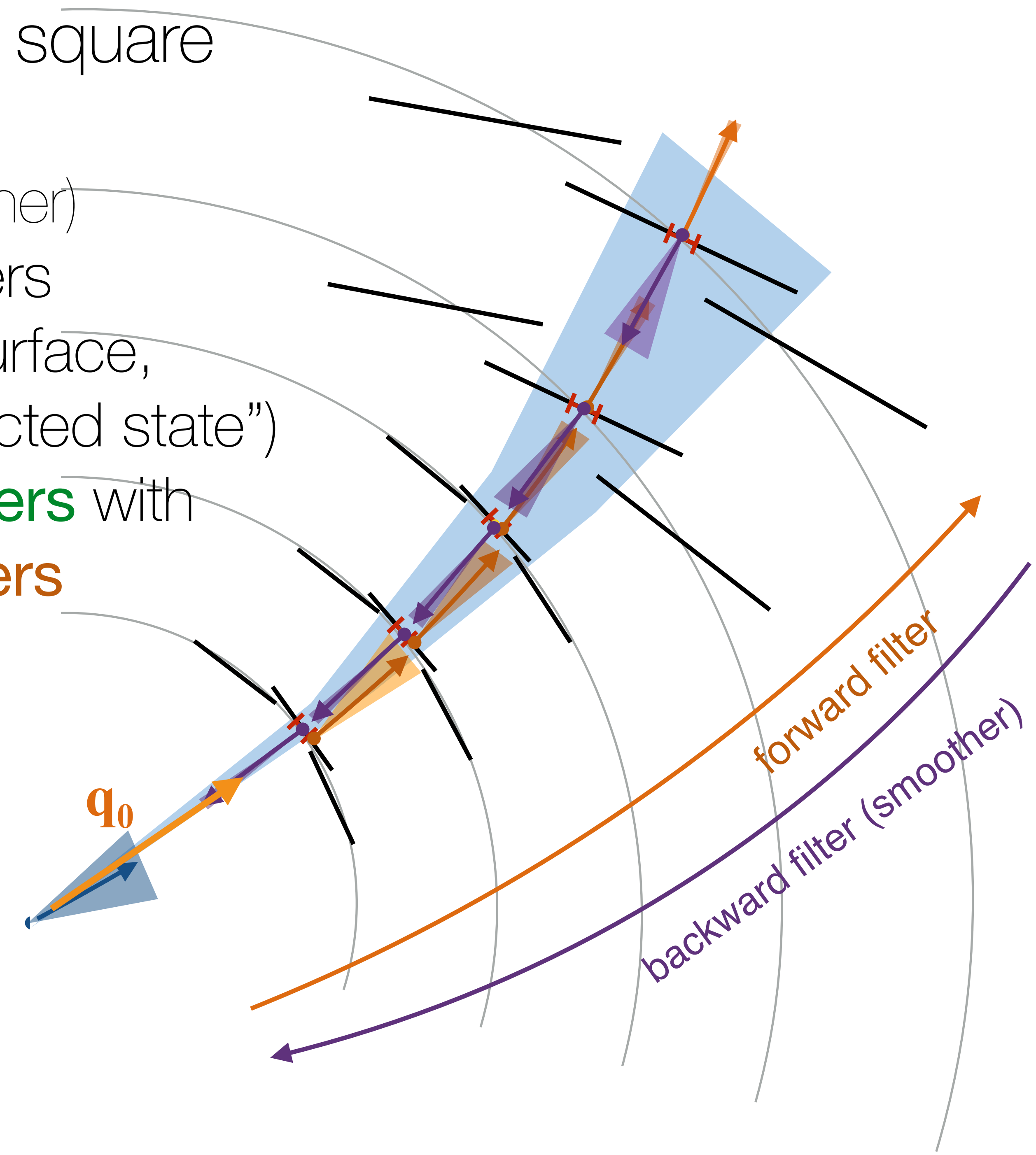
Track fitting Kalman filtering

offers an alternative solution to the large matrix inversion

- initially developed by I. Kalman to track missiles
- for HEP pioneered by Billoir and R. Fruehwirth

performs a progressive way of least square estimation

- equivalent to a χ^2 fit (if run with a smoother)
- start with **transport** of track parameters (and covariances) to measurement surface, create **predicted parameters** ("predicted state")
- combine/update **predicted parameters** with **measurement** to **updated parameters** ("filtered state")



Track fitting Kalman filter expressed in maths

let's assume the *k-th* filter step

- propagate parameters and **covariances** from *k-1* to *k* adding noise $\boldsymbol{Q}_k$

$$\boldsymbol{q}_{k|k-1} = \boldsymbol{f}_{k|k-1}(\boldsymbol{q}_{k-1|k-1})$$

$$\boldsymbol{C}_{k|k-1} = \boldsymbol{F}_{k|k-1}\boldsymbol{C}_{k-1|k-1}\boldsymbol{F}^{\mathrm{T}}_{k|k-1} + \boldsymbol{Q}_k$$

- update the prediction with measurement

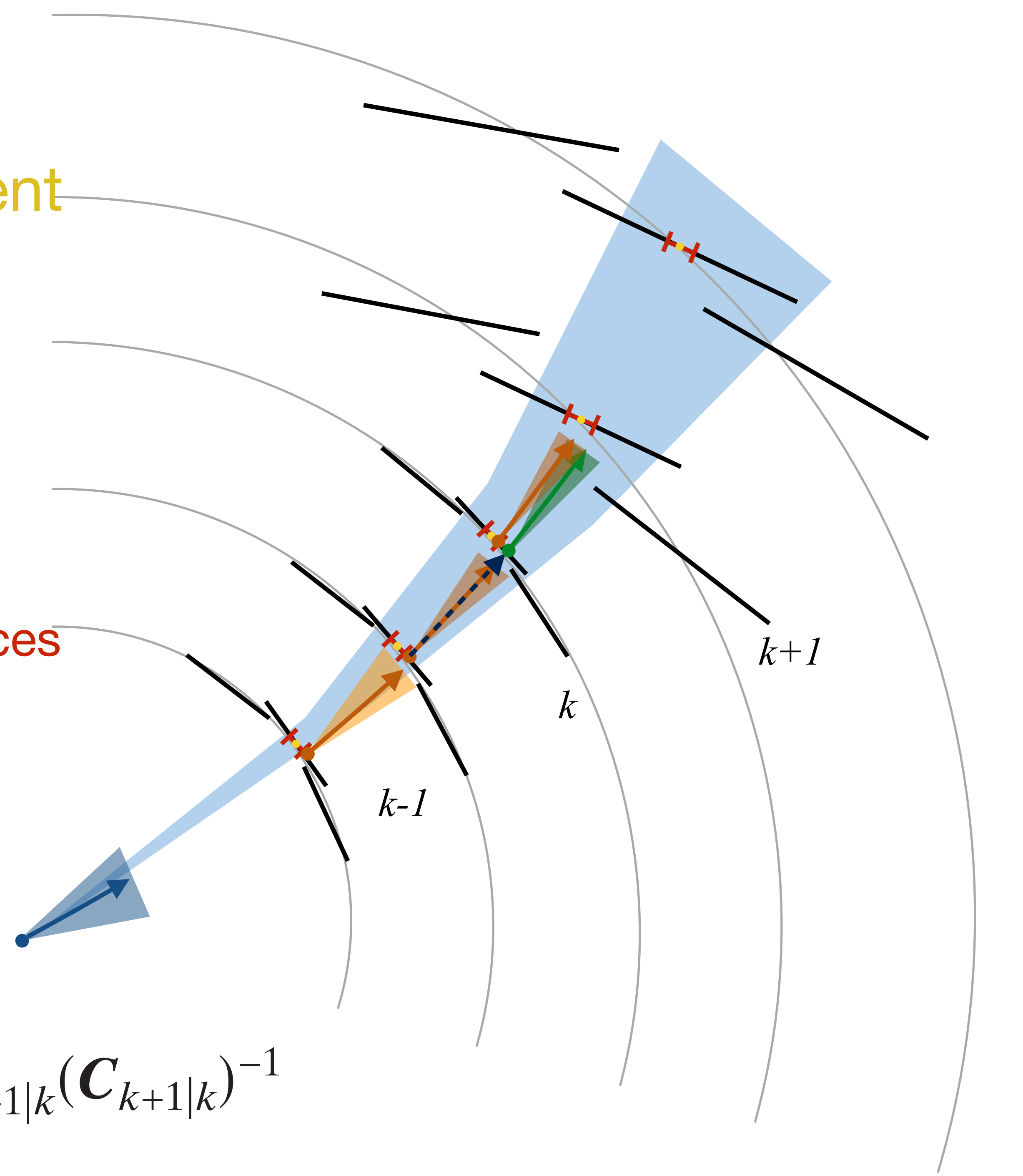
$$\boldsymbol{q}_{k|k} = \boldsymbol{q}_{k|k-1} + \boldsymbol{K}_k[\boldsymbol{m}_k - \boldsymbol{h}_k(\boldsymbol{q}_{k|k-1})]$$

$$\boldsymbol{C}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{C}_{k|k-1}$$

with gain matrix $\boldsymbol{K}_k$:

$$\boldsymbol{K}_k = \boldsymbol{C}_{k|k-1}\boldsymbol{H}^{\mathrm{T}}_k(\boldsymbol{G}_k + \boldsymbol{H}_k\boldsymbol{C}_{k|k-1}\boldsymbol{H}^{\mathrm{T}}_k)^{-1}$$

mapping measurement covariances



run the smoother from *k+1* to *k*

$$\boldsymbol{q}_{k|n} = \boldsymbol{q}_{k|k} + \boldsymbol{A}_k(\boldsymbol{q}_{k+1|n} - \boldsymbol{q}_{k+1|k})$$

$$\boldsymbol{C}_{k|n} = \boldsymbol{C}_{k|k} - \boldsymbol{A}_k(\boldsymbol{C}_{k+1|k} - \boldsymbol{C}_{k+1|n})\boldsymbol{A}^{\mathrm{T}}_k$$

with smoother gain matrix $\boldsymbol{A}_k$: $\boldsymbol{A}_k = \boldsymbol{C}_{k|k}\boldsymbol{F}^{\mathrm{T}}_{k+1|k}(\boldsymbol{C}_{k+1|k})^{-1}$

Track fitting Least squares estimator

Global χ^2 fitter and Kalman filter are least squares estimators that rely on gaussian errors:

$\boldsymbol{G_k}$ the covariance of measurement $\mathbf{m}_k$

$\boldsymbol{Q}_k$ the noise addition due to material effects (Kalman filter)

$\sum_i \delta\theta_i^T Q_i^{-1} \delta\theta_i$ χ^2 contribution from scattering angles (χ^2 fitter)

Track fitting Least squares estimator

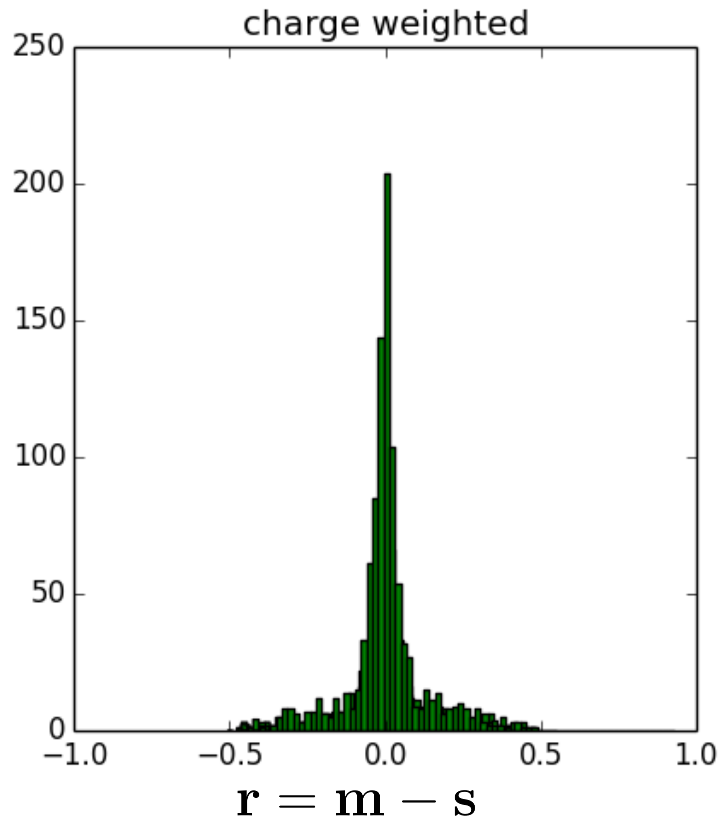
Global χ^2 fitter and Kalman filter are least squares estimators that rely on gaussian errors:

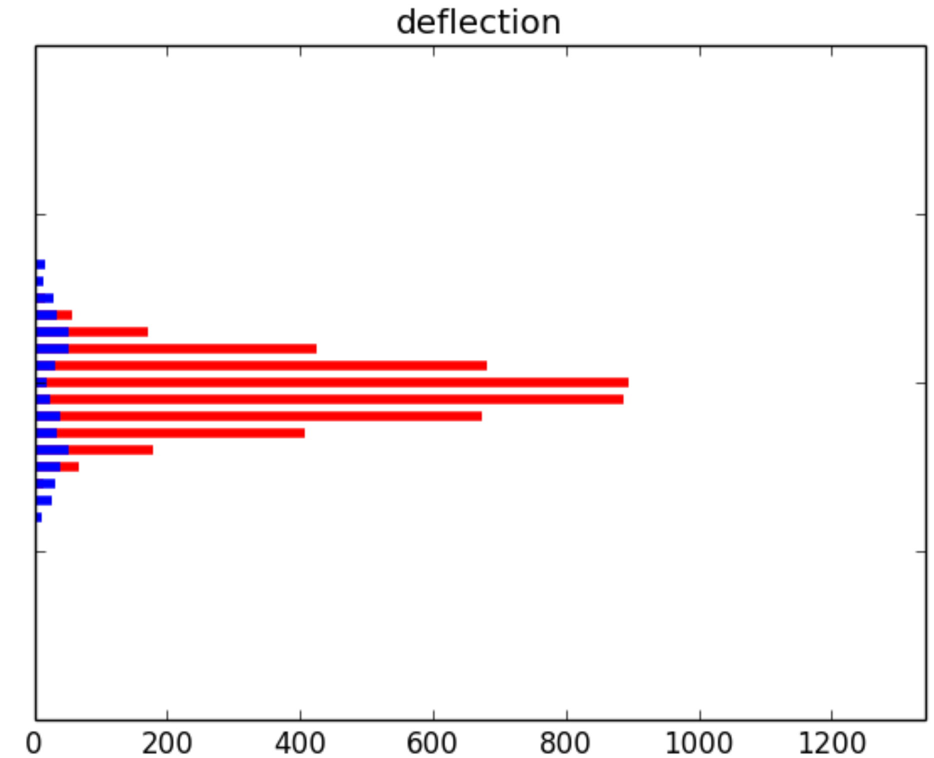
$\boldsymbol{G_k}$ the covariance of measurement $\mathbf{m}_k$

$\boldsymbol{Q}_k$ the noise addition due to material effects (Kalman filter)

$\sum_i \delta\theta_i^T Q_i^{-1} \delta\theta_i$ χ^2 contribution from scattering angles (χ^2 fitter)

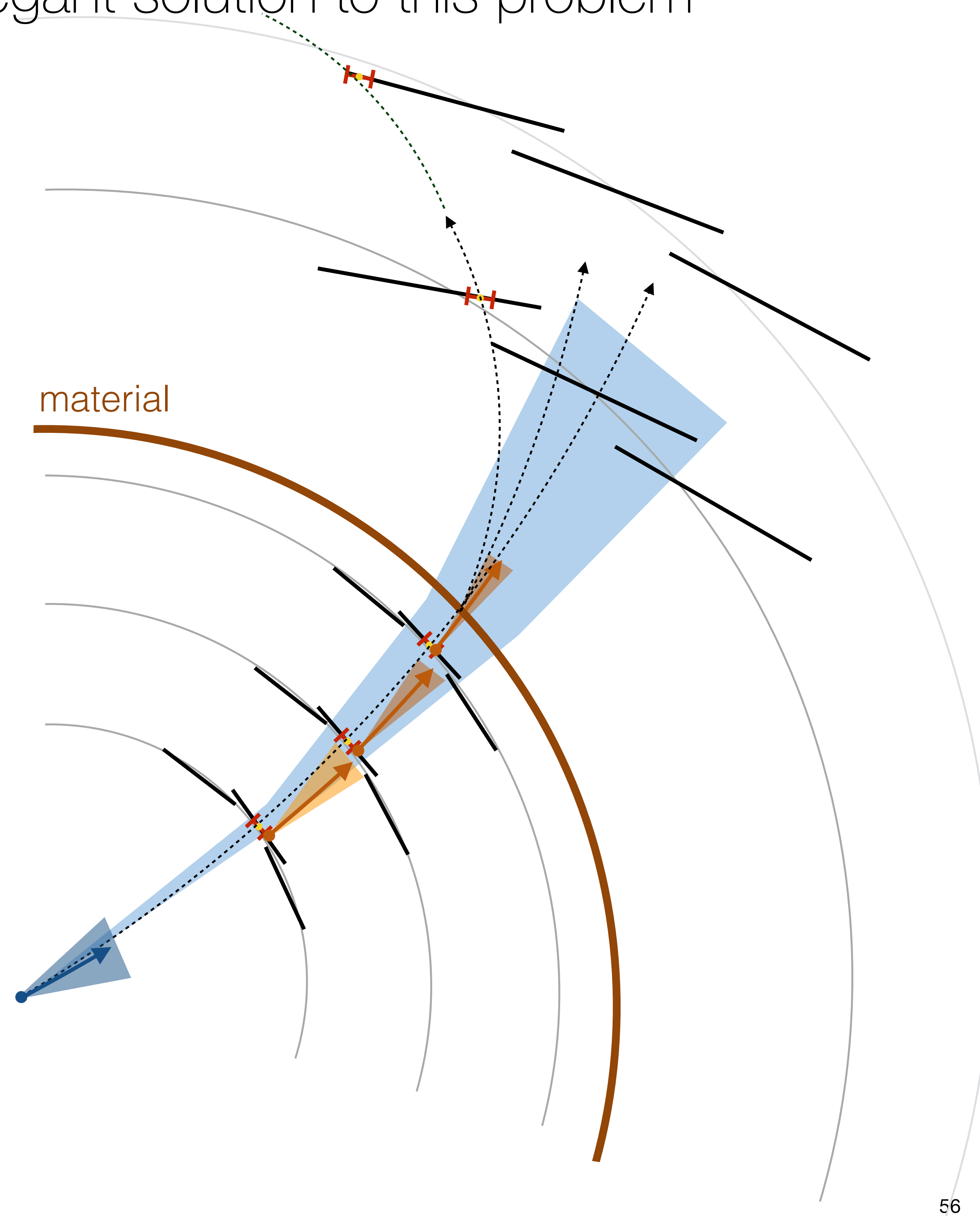
neither of them are !

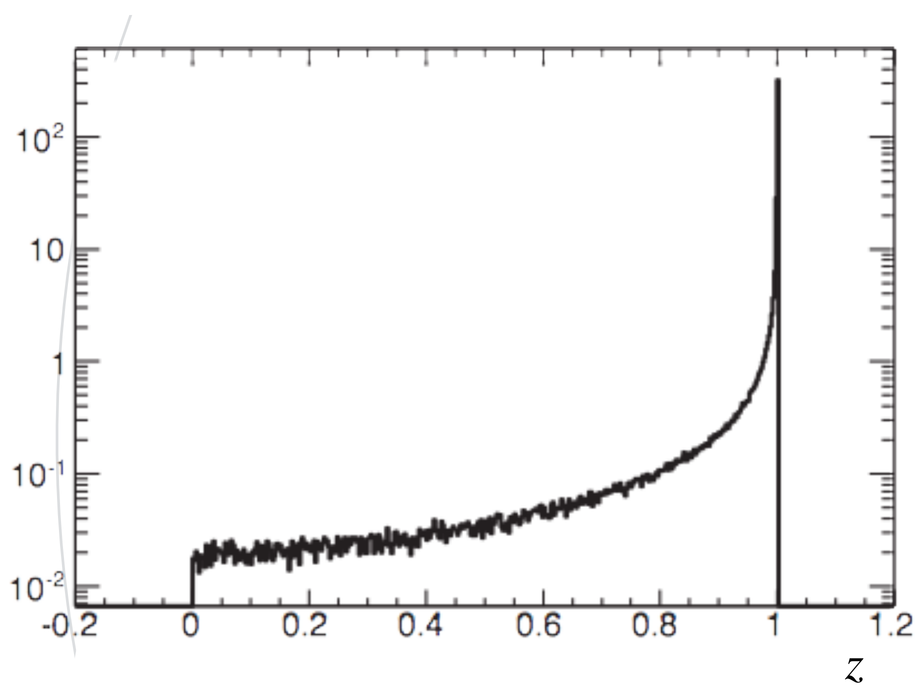




Track fitting Bremsstrahlung tail for electrons

Kalman filter formalism offers a very elegant solution to this problem

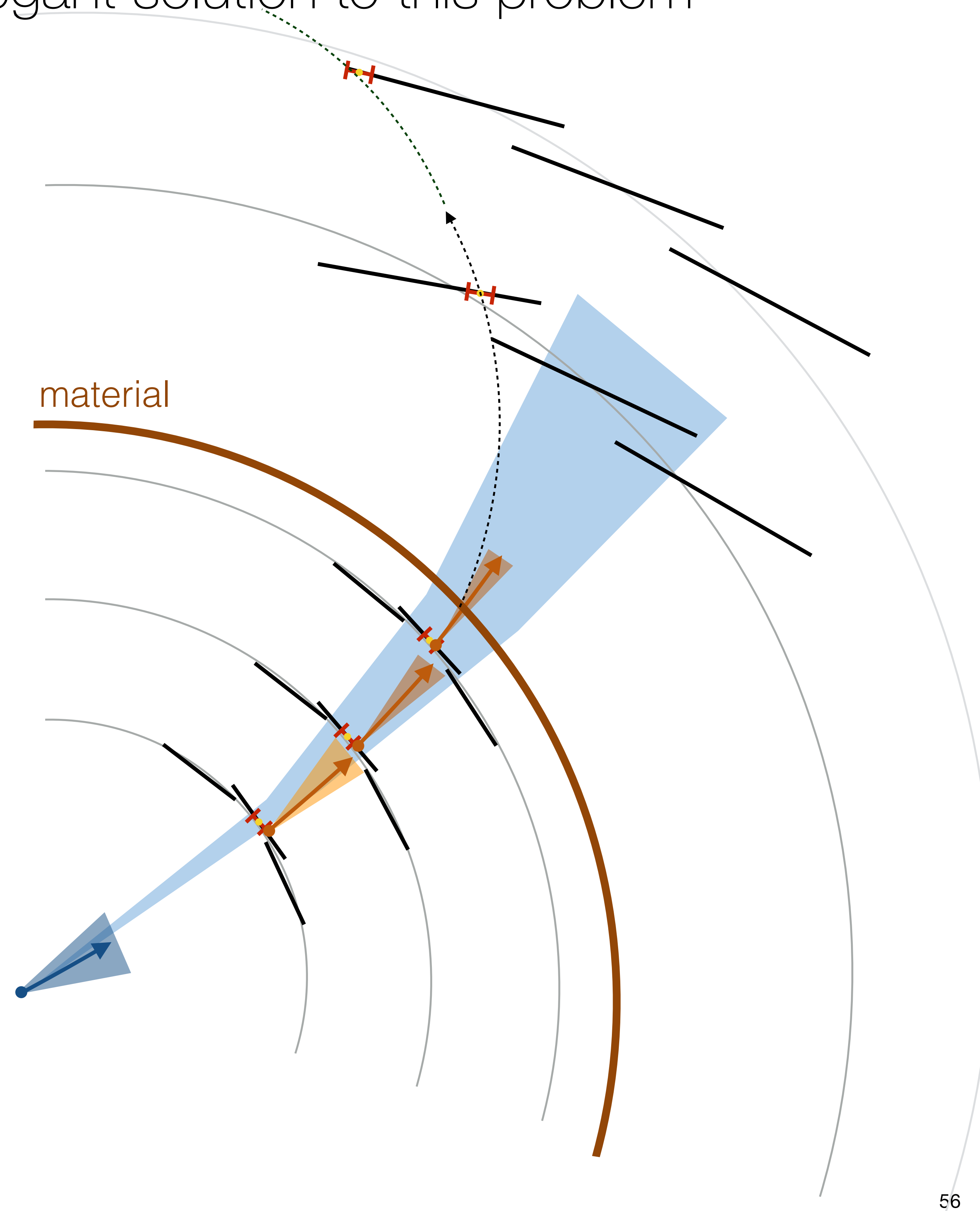


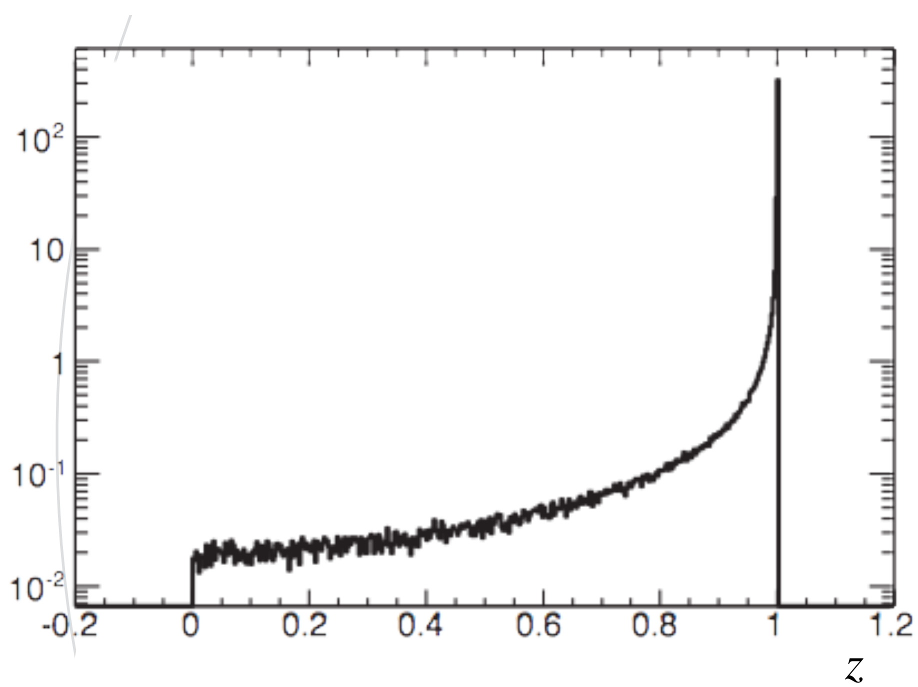


$$z = E_i/E_f \qquad f(z) = \frac{[-\ln z]^{c-1}}{\Gamma(c)}$$

Track fitting Bremsstrahlung tail for electrons

Kalman filter formalism offers a very elegant solution to this problem

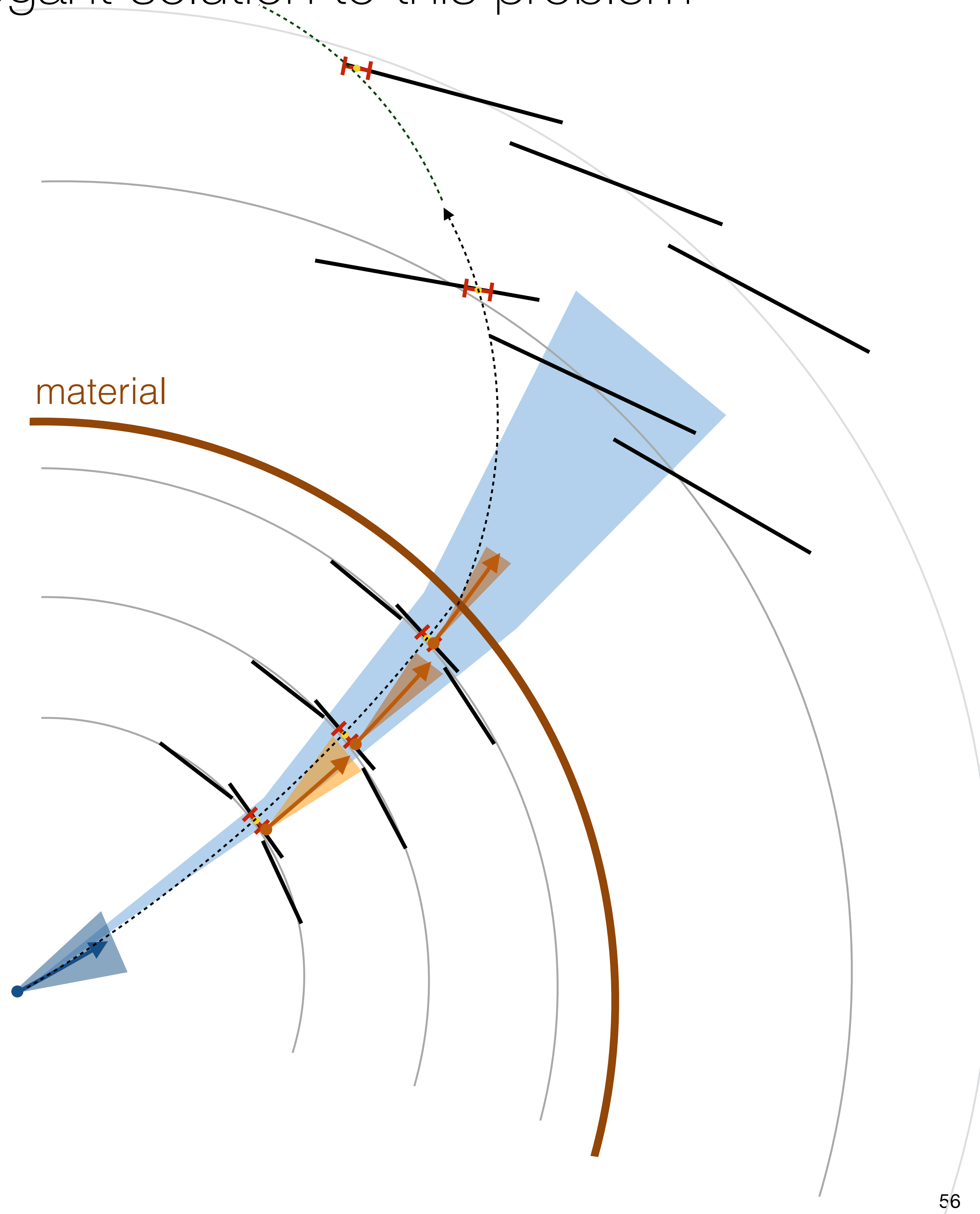


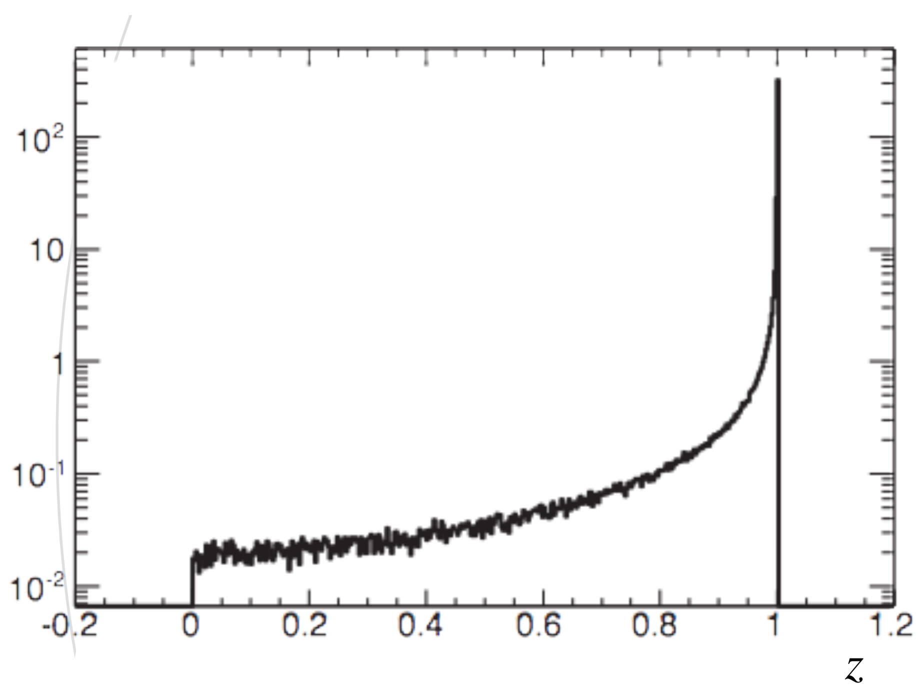


$z = E_i/E_f$ $\qquad f(z) = \frac{[-\ln z]^{c-1}}{\Gamma(c)}$

Track fitting Bremsstrahlung tail for electrons

Kalman filter formalism offers a very elegant solution to this problem



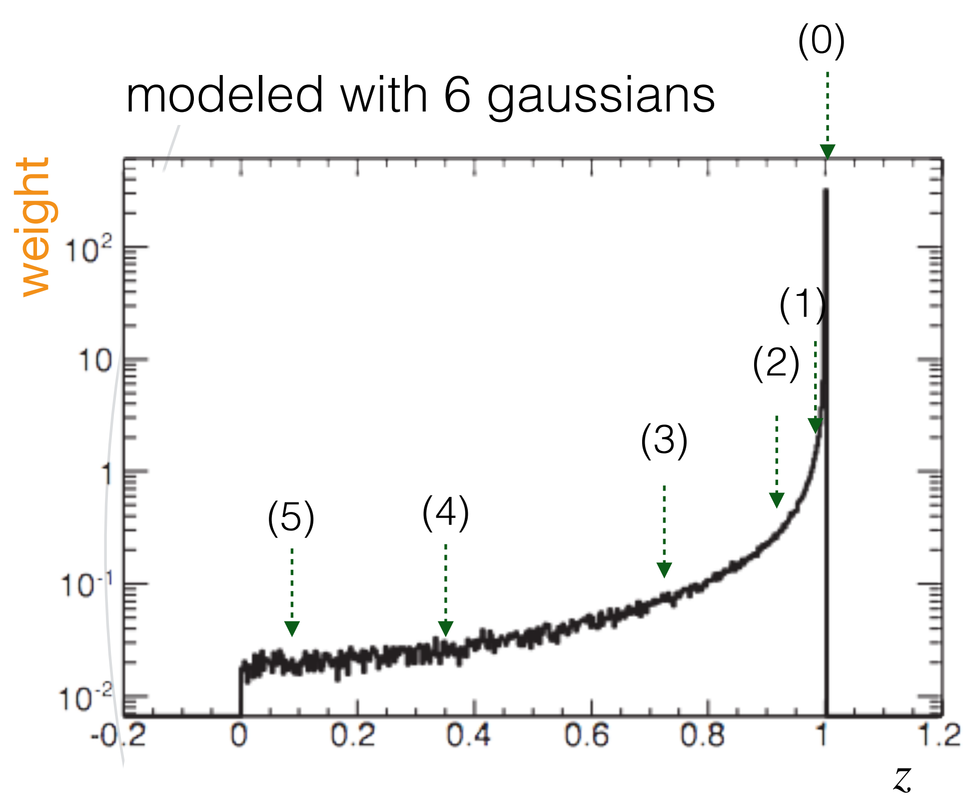


$$z = E_i/E_f \qquad f(z) = \frac{[-\ln z]^{c-1}}{\Gamma(c)}$$

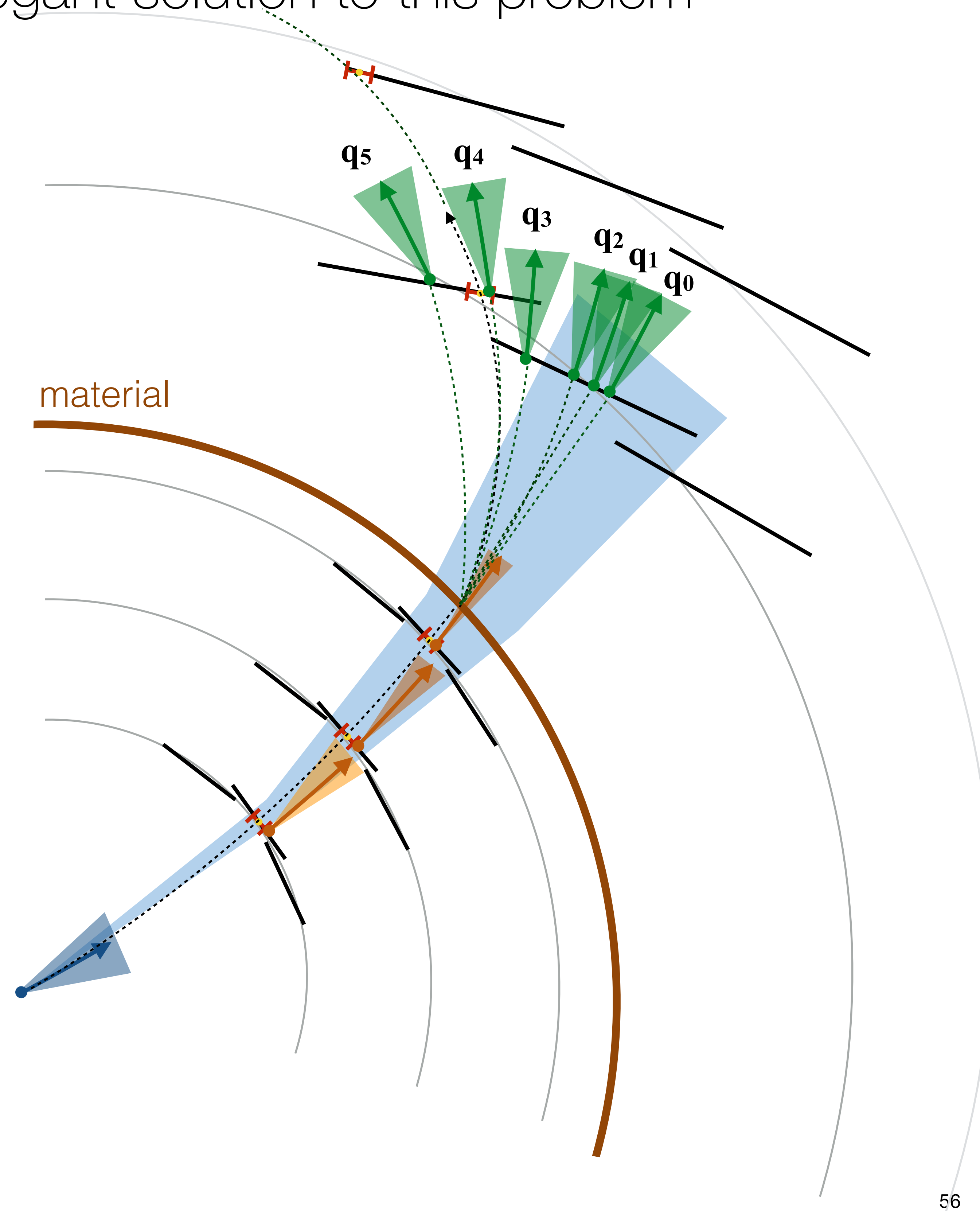
Track fitting Bremsstrahlung tail for electrons

Kalman filter formalism offers a very elegant solution to this problem

- fork the Kalman filter at the material layer into multiple components with weights and propagate them individually



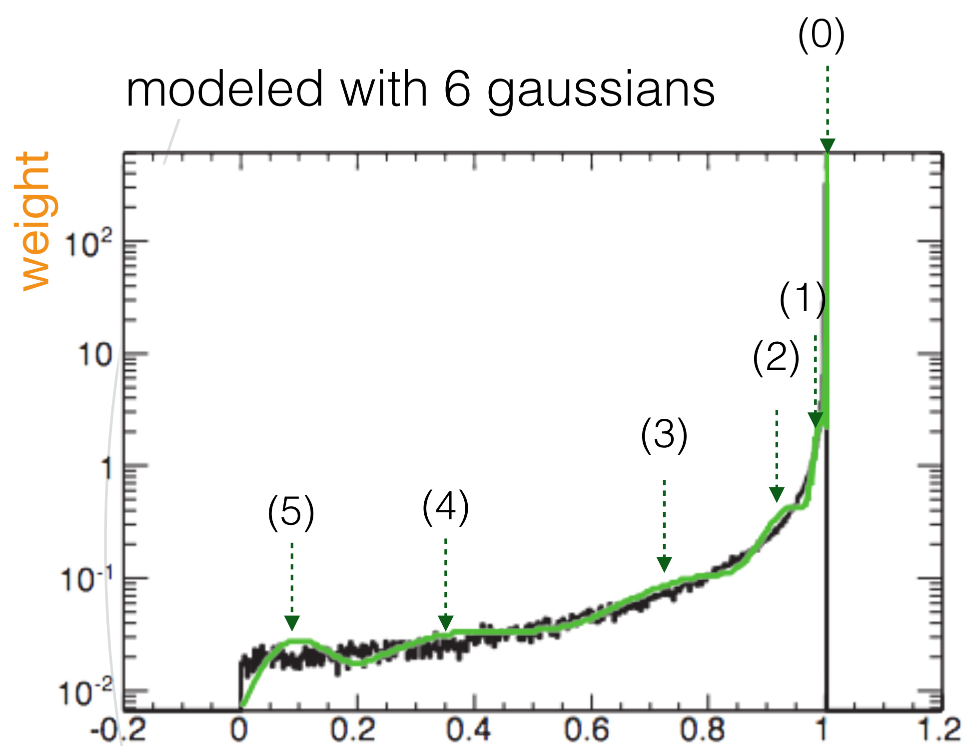
$$z = E_i/E_f \qquad f(z) = \frac{[-\ln z]^{c-1}}{\Gamma(c)}$$



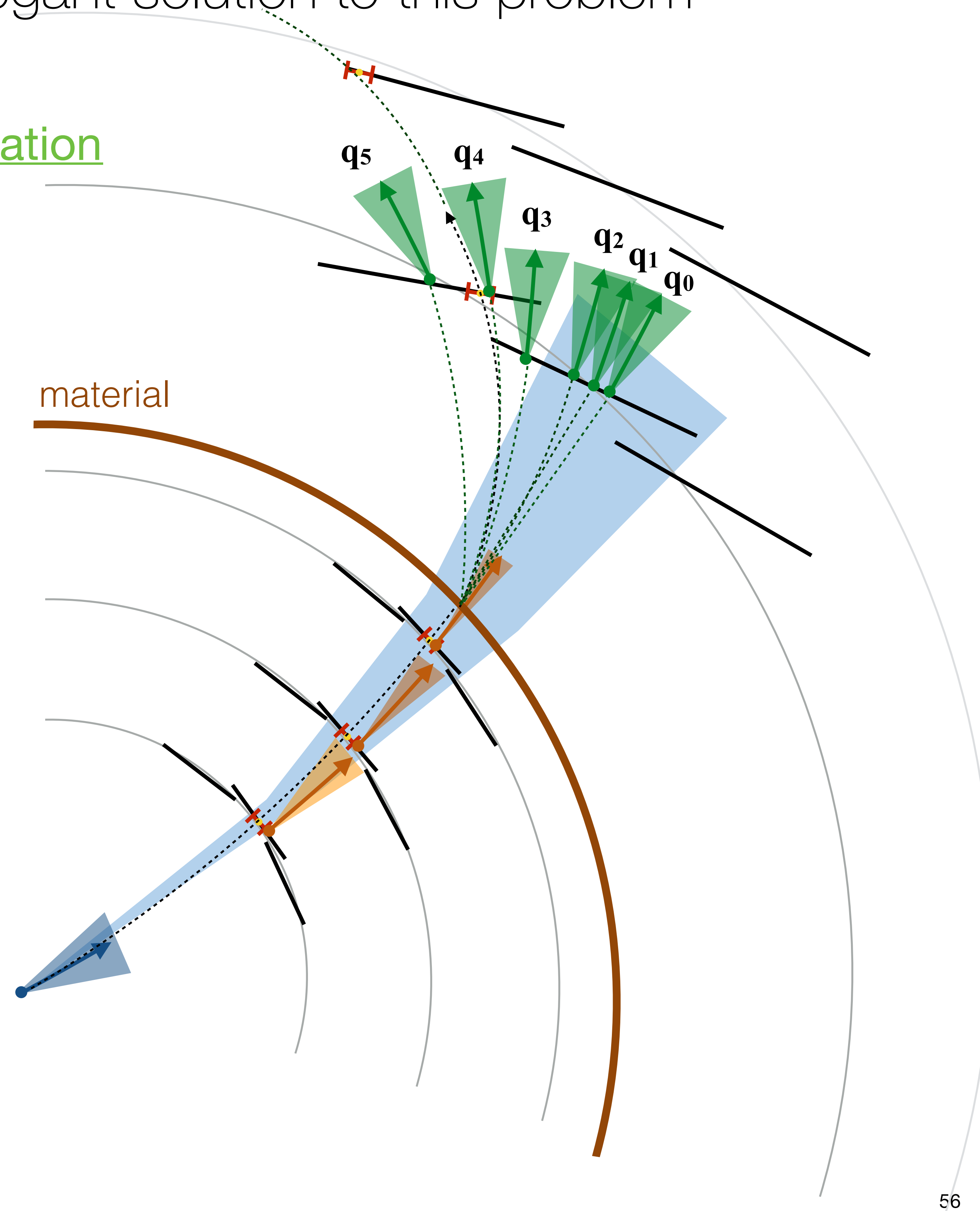
Track fitting Bremsstrahlung tail for electrons

Kalman filter formalism offers a very elegant solution to this problem

- modeling of non-gaussian noise through multivariant (gaussian) approximation
- fork the Kalman filter at the material layer into multiple components with weights and propagate them individually



$z = E_i/E_f$ $\quad f(z) = \frac{[-\ln z]^{c-1}}{\Gamma(c)}$

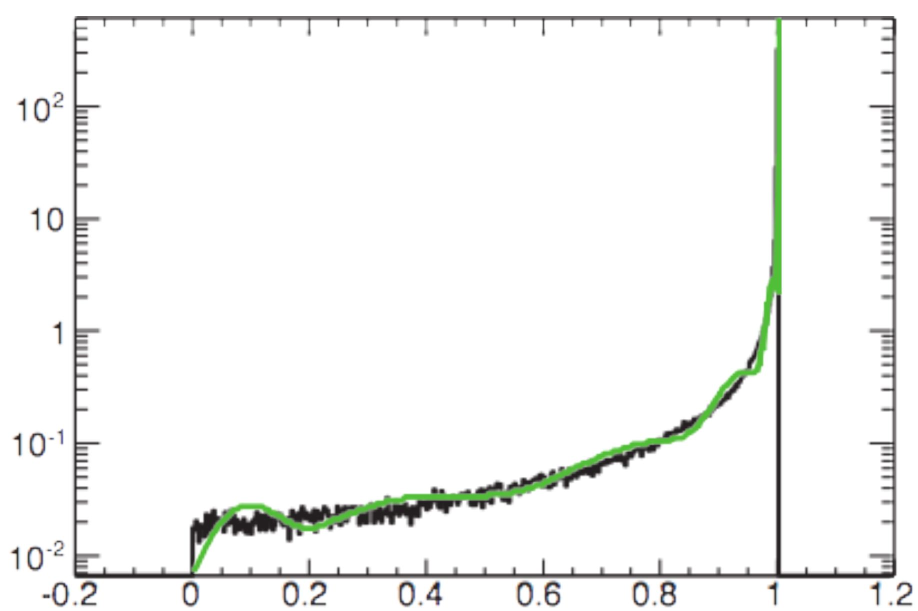


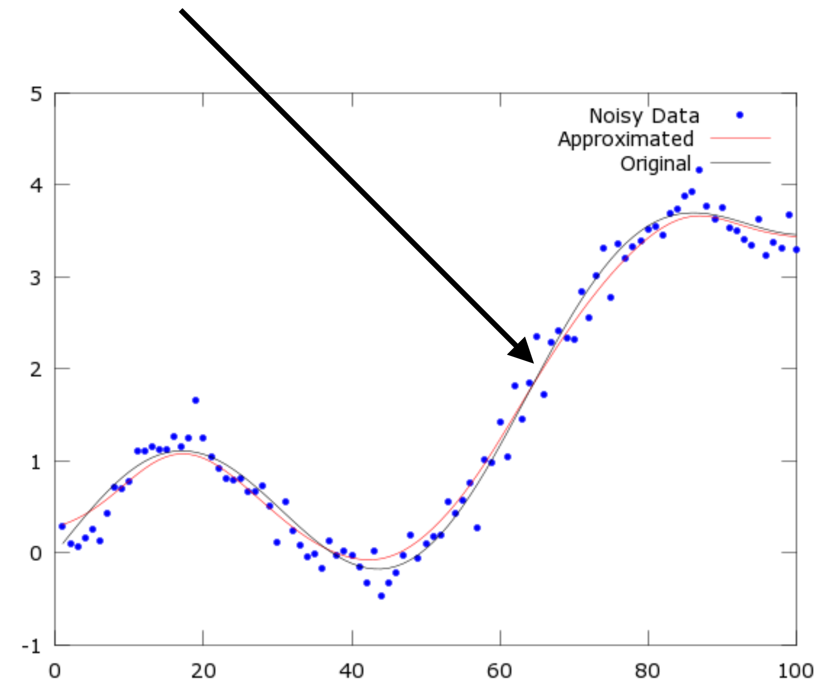
Track fitting electron fitting / non-gaussian noise

Electron classification is an obvious playground for ML
- PID is a standard field for NNs/BDTs

(Electron) track fitting ?
- obviously ML can be used to fit a non-linear system
- the fit function has to make sense, though, it has to behave like an electron

Idea is not to find the function that fits my measurements best !

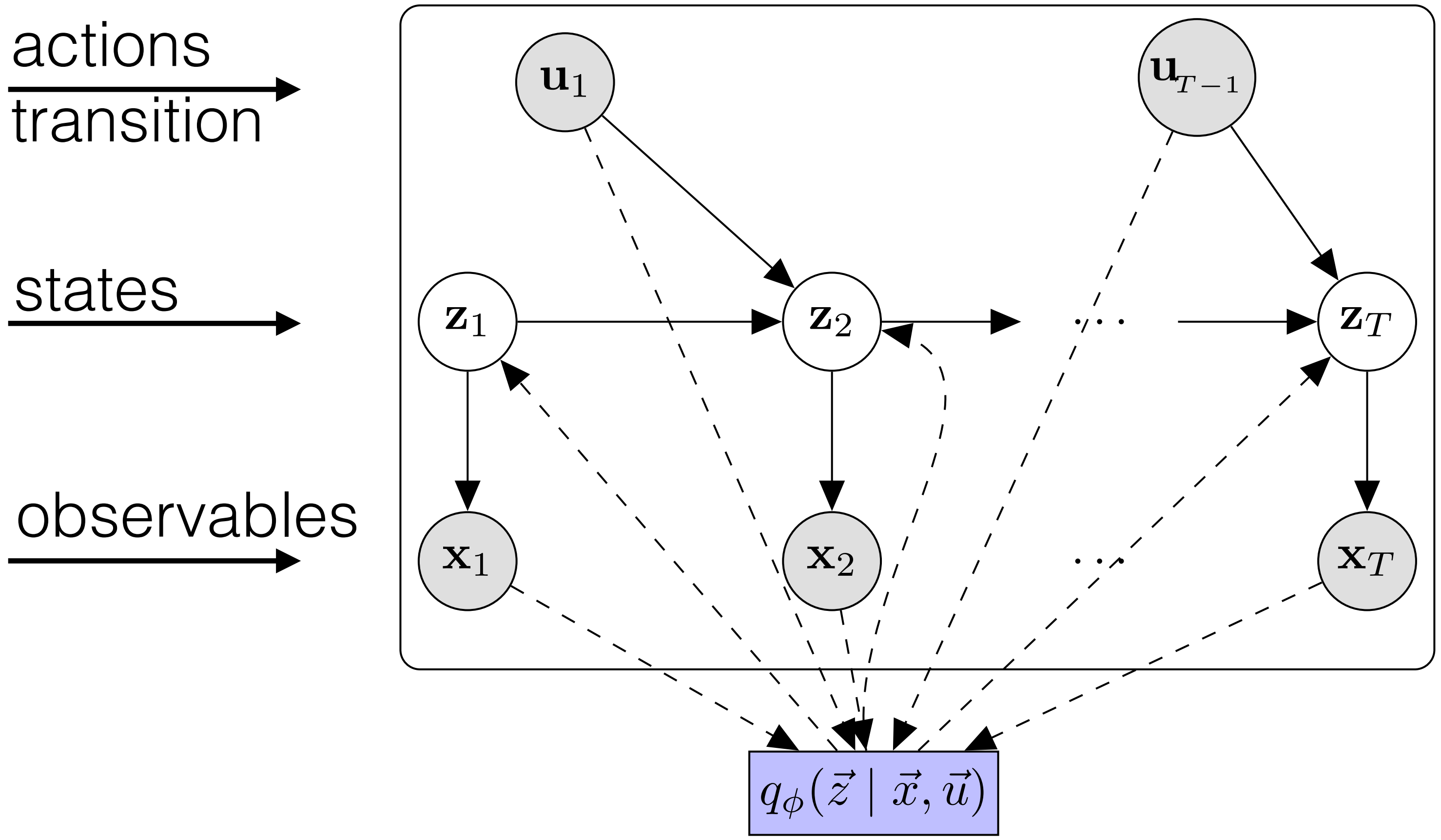




Track fitting Deep Kalman Filter

Kalman Filter is per se a linear dynamical system

- the GSF is a multivariate, but still linear dynamical system
- Extended Kalman filter, e.g. is an extension with a non-linear transition
- idea: using NN to describe non-linear transition function: Deep Kalman Filter



$$z_1 \sim \mathcal{N}(\mu_0; \Sigma_0)$$
$$z_t \sim \mathcal{N}(G_\alpha(z_{t-1}, u_{t-1}, \Delta_t), S_\beta(z_{t-1}, u_{t-1}, \Delta_t))$$
$$x_t \sim \Pi(F_\kappa(z_t)).$$

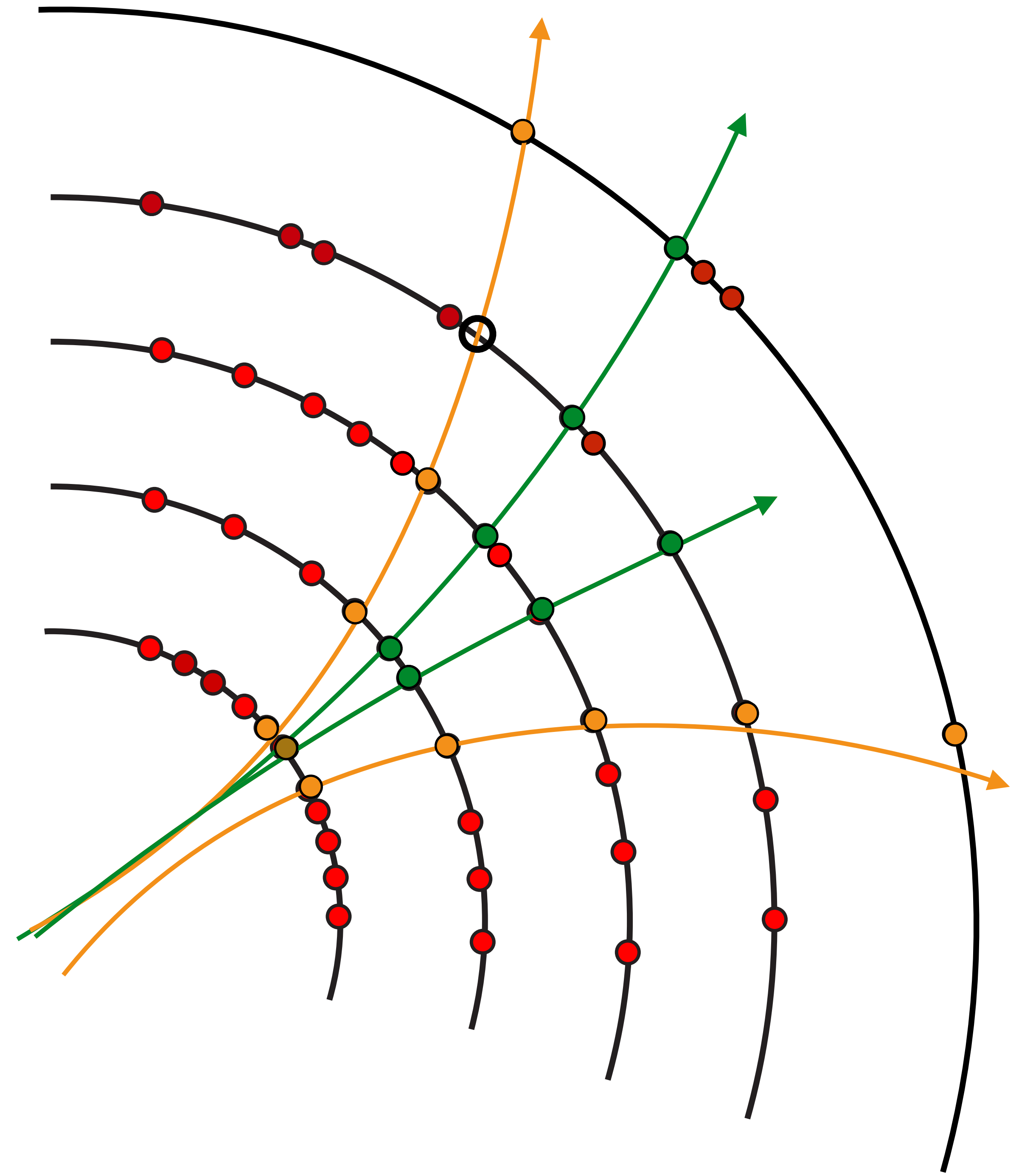
Optimize *jointly* over generative model $p_\theta(\vec{x}|\vec{u})$
and variational approximation $q_\phi(\vec{z}|\vec{x}, \vec{u})$ and learning via stochastic back-progation.

See:
Rahul G. Krishnan Uri Shalit David Sontag , arXiv:1511.05121v2

Track ranking What is a good track ?

Some of the characteristics can only be checked after all track candidates are found

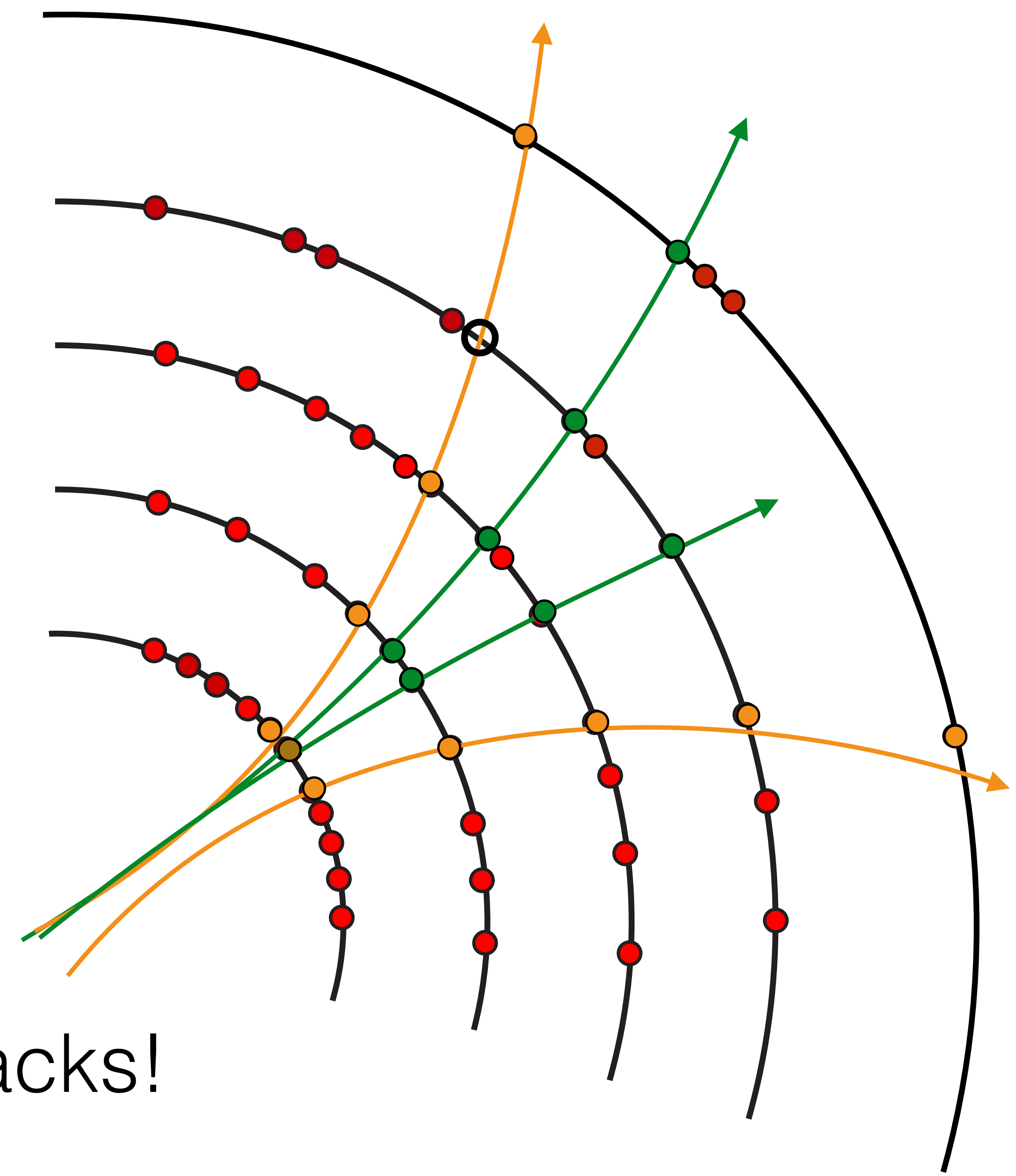
good track	**not so good track**
many compatible hits	short tracks
completeness	holes
uniqueness	shared hits
low χ^2/ndf	bad fit quality, outliers
small impact parameter (for primaries)	
clusters are compatible	



Track ranking What is a good track ?

Some of the characteristics can only be checked after all track candidates are found

good track	**not so good track**
many compatible hits	short tracks
completeness	holes
uniqueness	shared hits
low χ^2/ndf	bad fit quality, outliers
small impact parameter (for primaries)	
clusters are compatible	

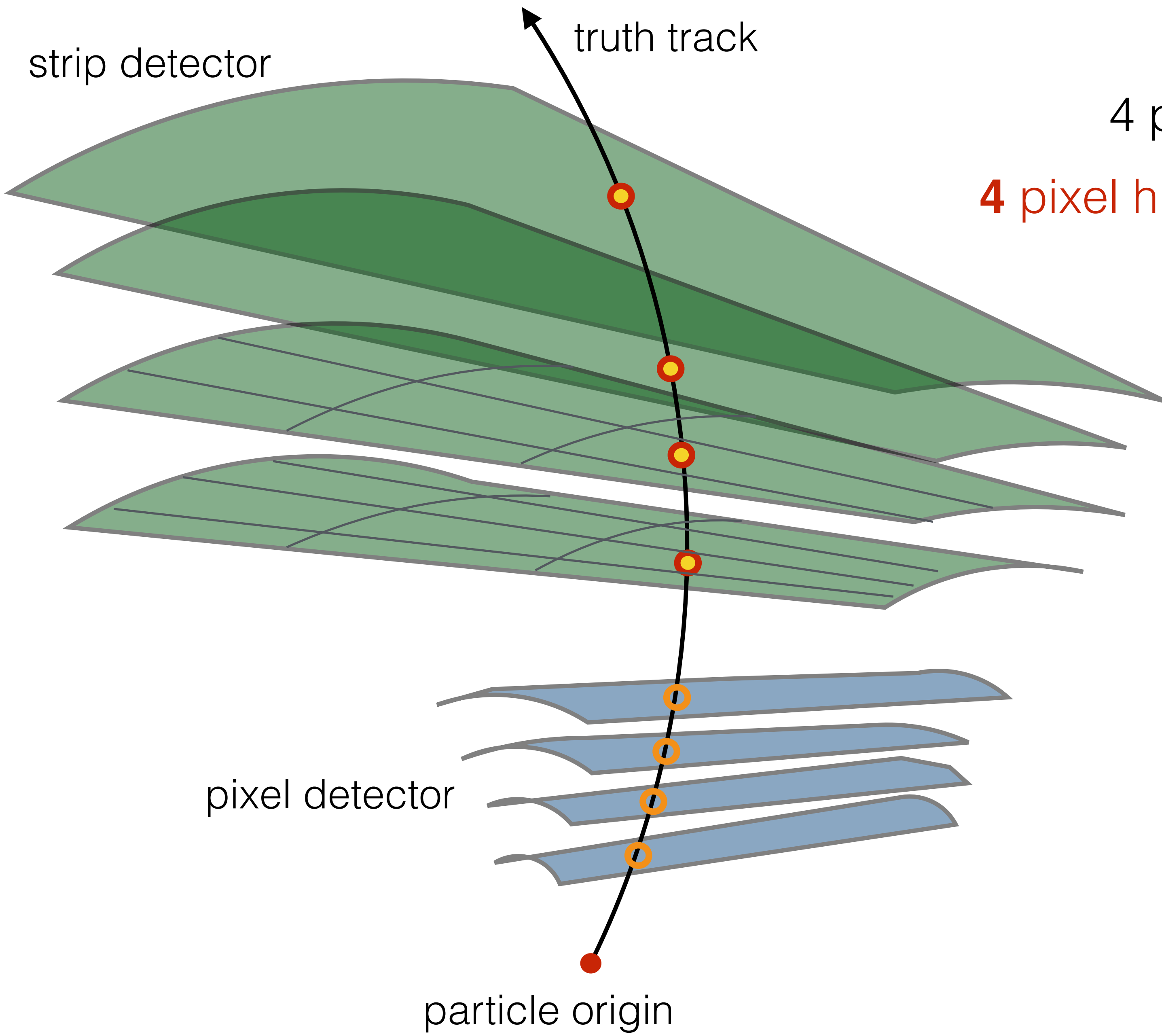


give scores and rank the tracks!

Track ranking A perfect track

There is no unique truth matching to define a found track

we use truth matching per hits



4 pixel hits, 4 strip hits created

4 pixel hits, **4** strip hits found and assigned

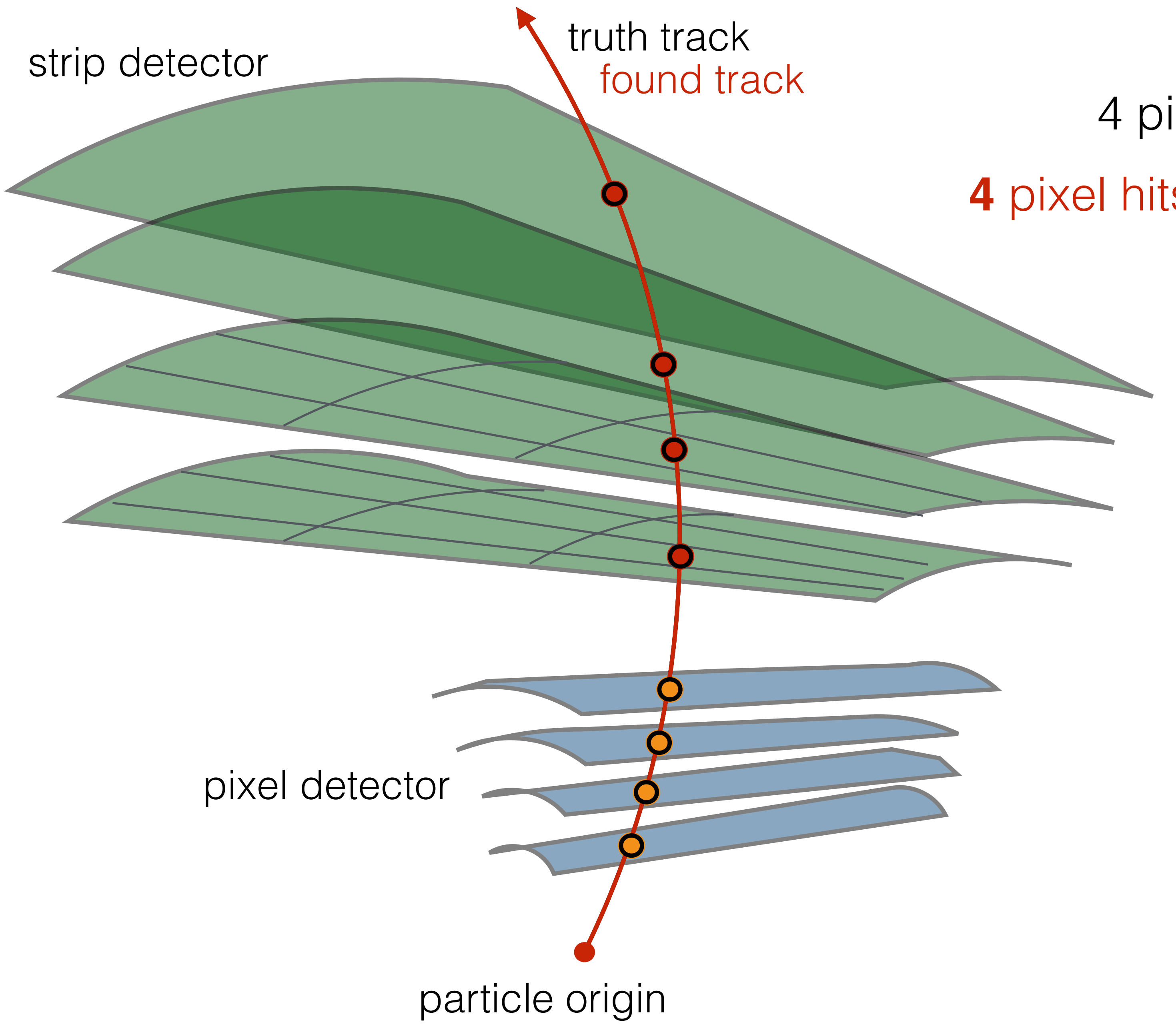
you can't do better,
score = 1

Illustration:
Track scoring, a perfect track with all hits assigned correctly.

Track ranking A perfect track

There is no unique truth matching to define a found track

we use truth matching per hits



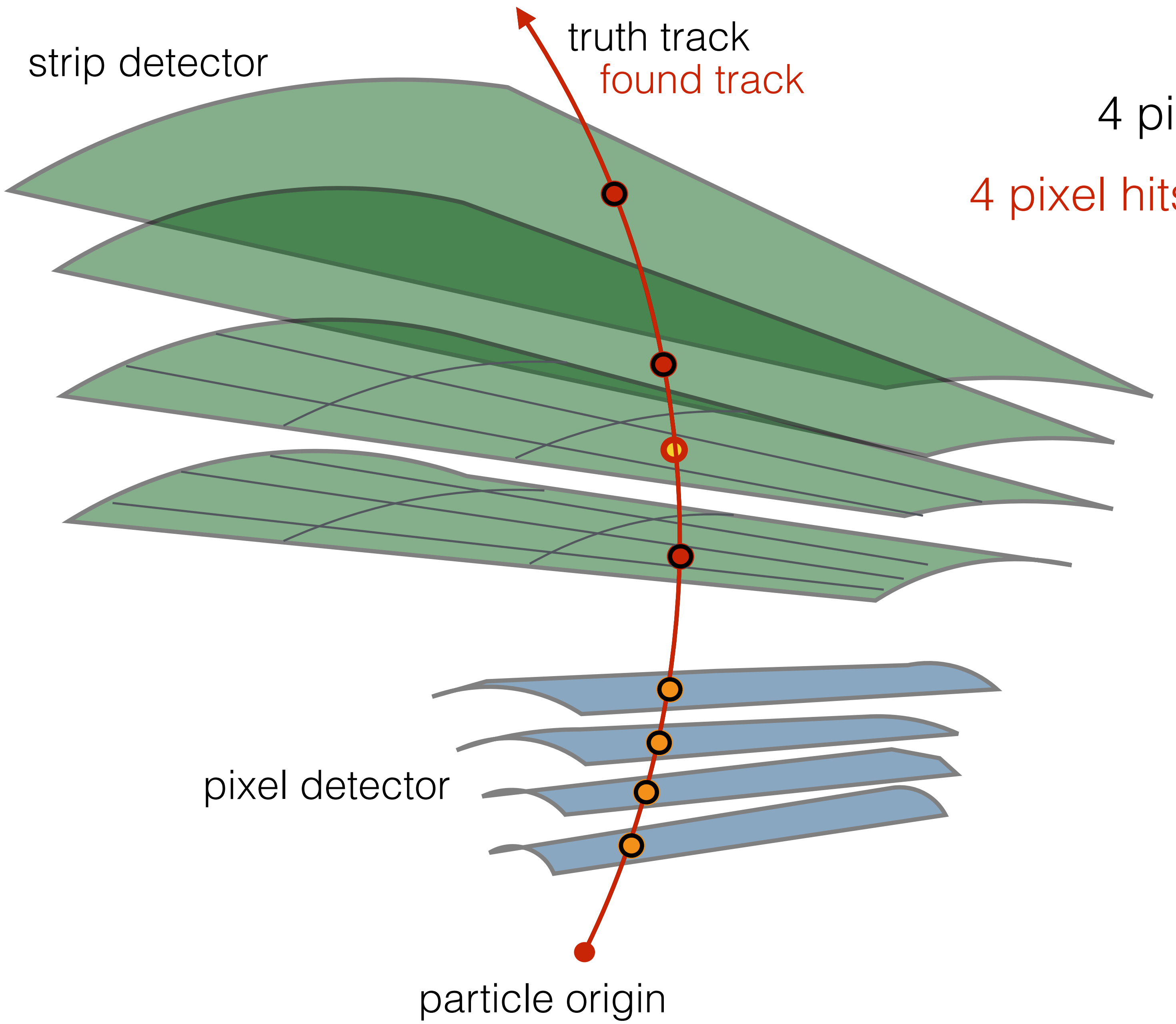
4 pixel hits, 4 strip hits created

4 pixel hits, **4** strip hits found and assigned

you can't do better,
score = 1

Illustration:
Track scoring, a perfect track with all hits assigned correctly.

Track ranking A good track



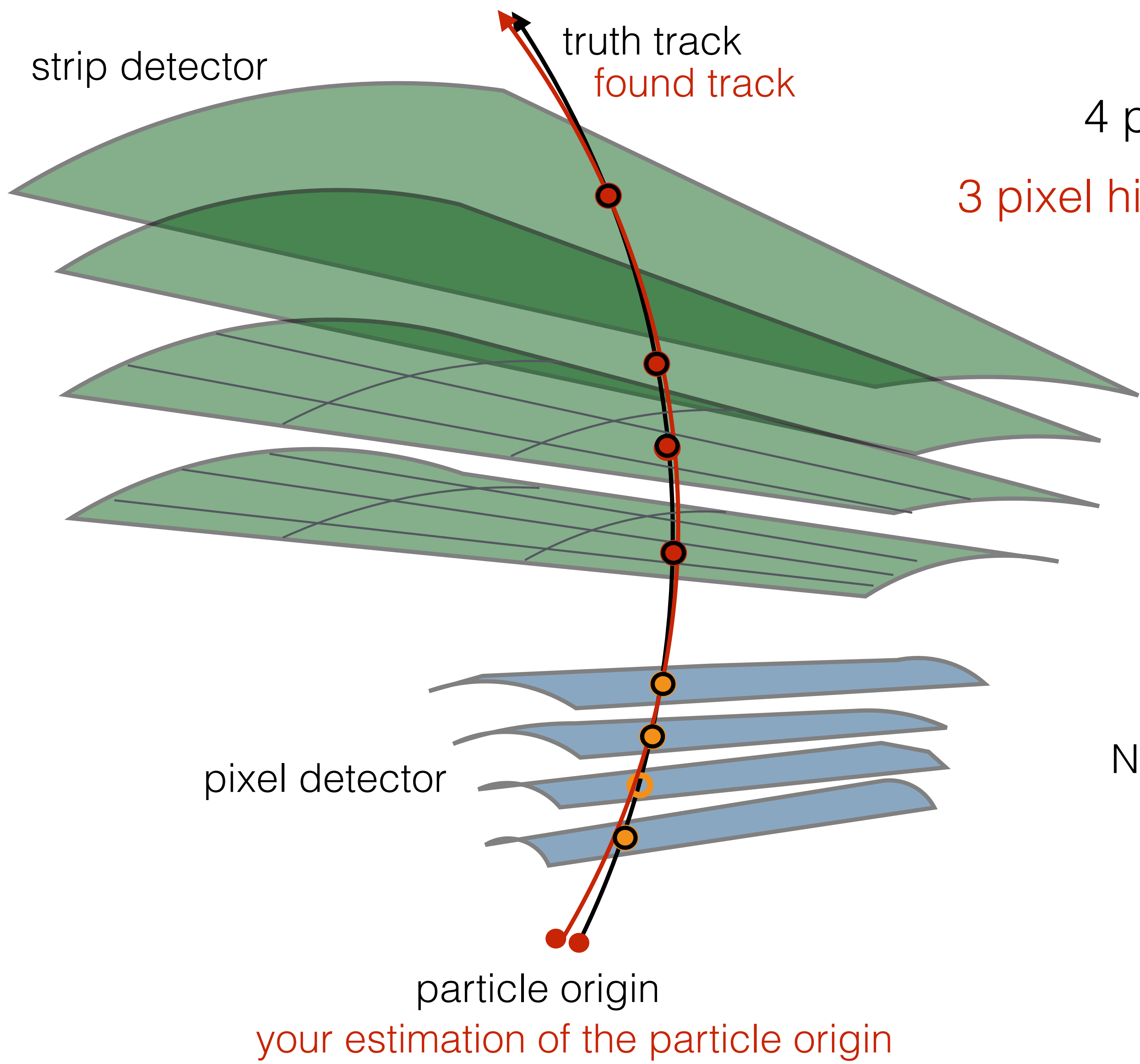
4 pixel hits, 4 strip hits created

4 pixel hits, 3 strip hits found and assigned

that's an ok track,
you got 7 out of 8,
naive score = 7/8 = 0.875

Illustration:
Track scoring, a good track with all but one hit assigned correctly.

Track ranking Another good track



4 pixel hits, 4 strip hits created

3 pixel hits, 4 strip hits found and assigned

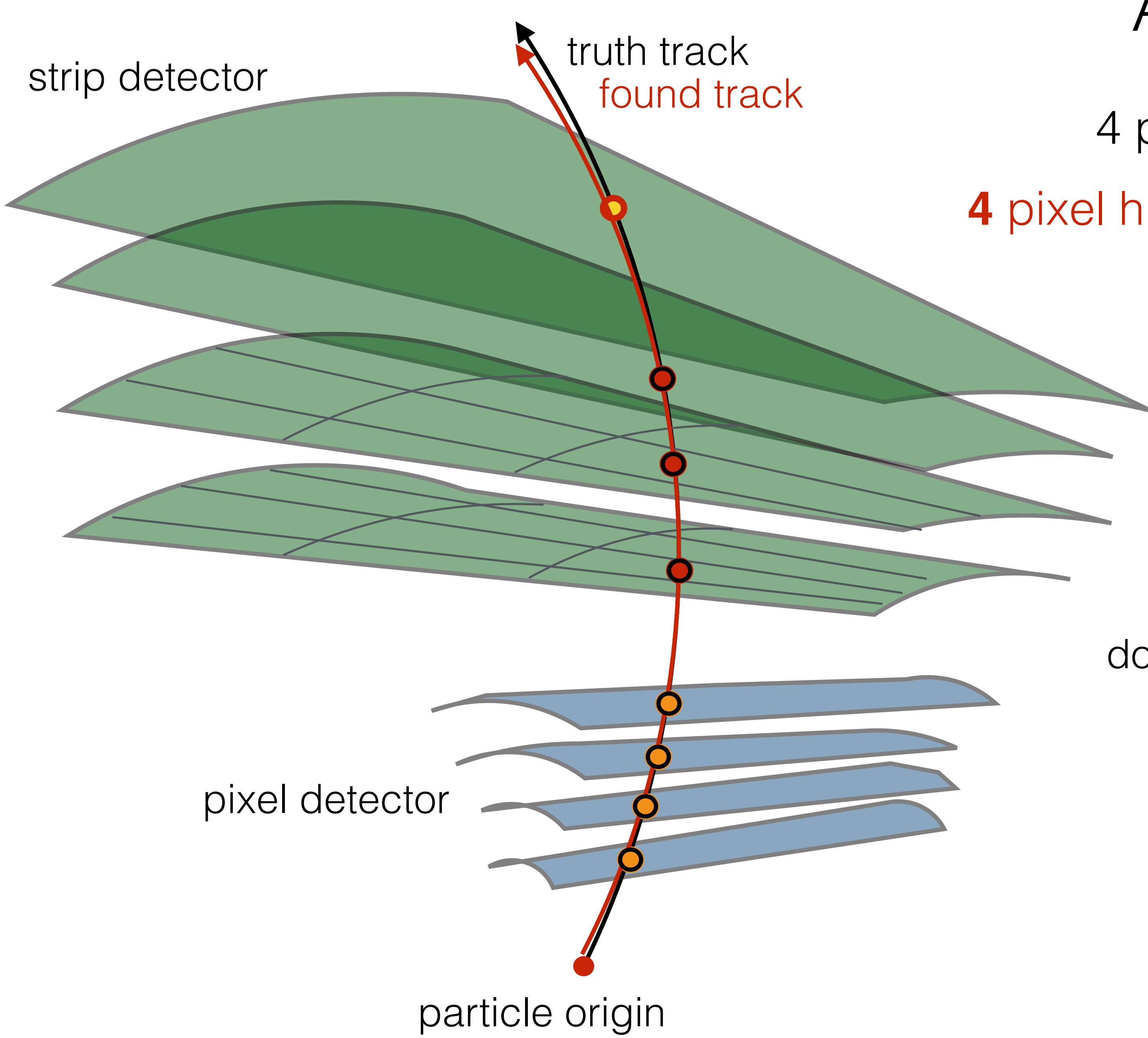
that's an ok track,
you got 7 out of 8,
naive score = 7/8 = 0.875

does a pixel hit weigh the same
as a strip hit ?

NOT if we want to measure primary
particles !!!

Illustration:
Track scoring, a good track with all but one hit assigned correctly, resulting in a slight mis-measurement of the impact parameter.

Track ranking Another good track



Another good track

4 pixel hits, 4 strip hits created

4 pixel hits, **3** strip hits found and assigned

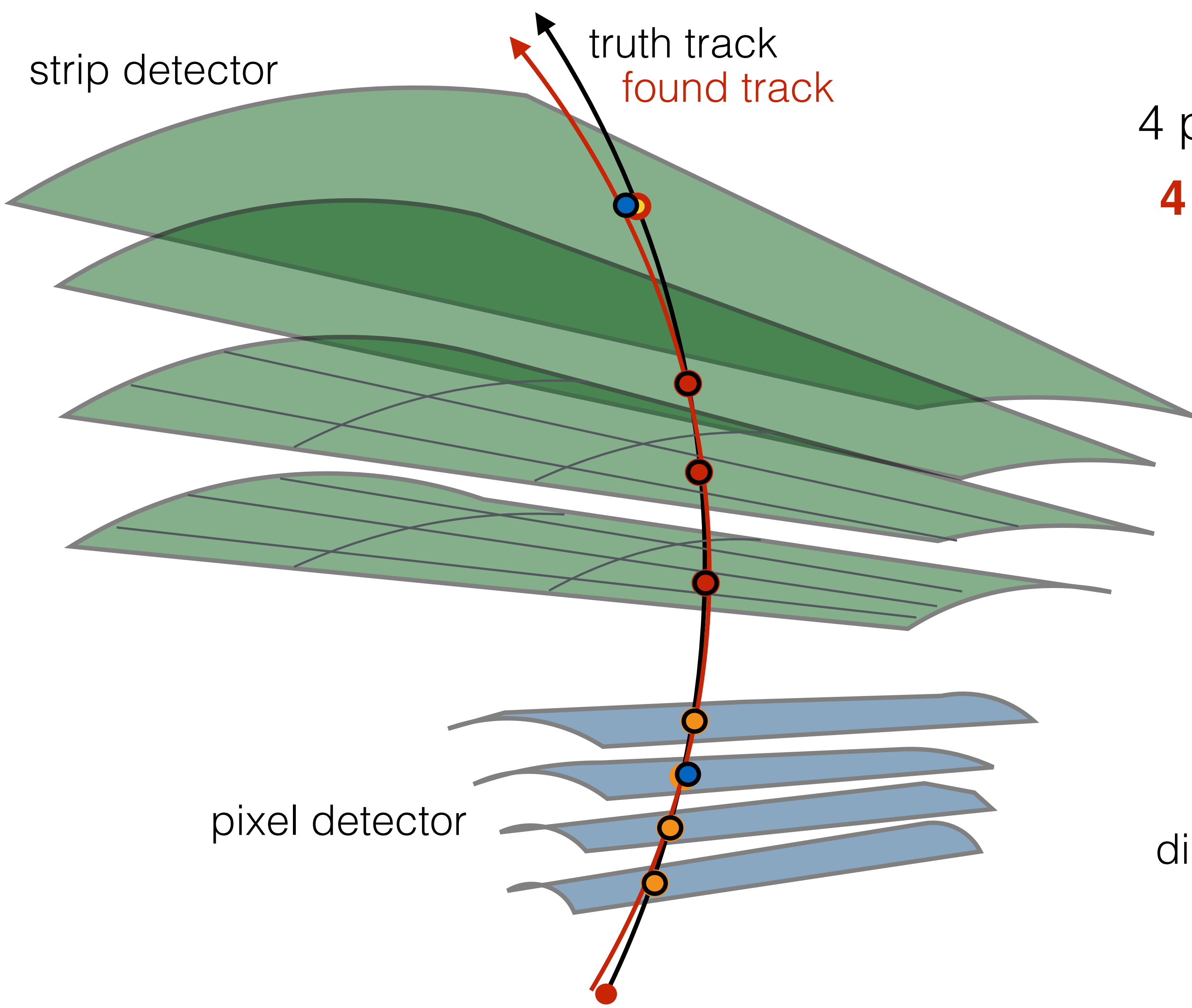
that's an ok track,
you got 7 out of 8,
naive score = 7/8 = 0.875

does a hit at the end weigh the same
as a strip hit ?

NOT if we want to
measure the momentum
precisely !!!

Illustration:
Track scoring, a good track with all but one hit assigned correctly, resulting in a slightly wrong momentum estimation,.

Track ranking A distorted track



4 pixel hits, 4 strip hits created

4 pixel hits, **4** strip hits found
2 wrongly associated

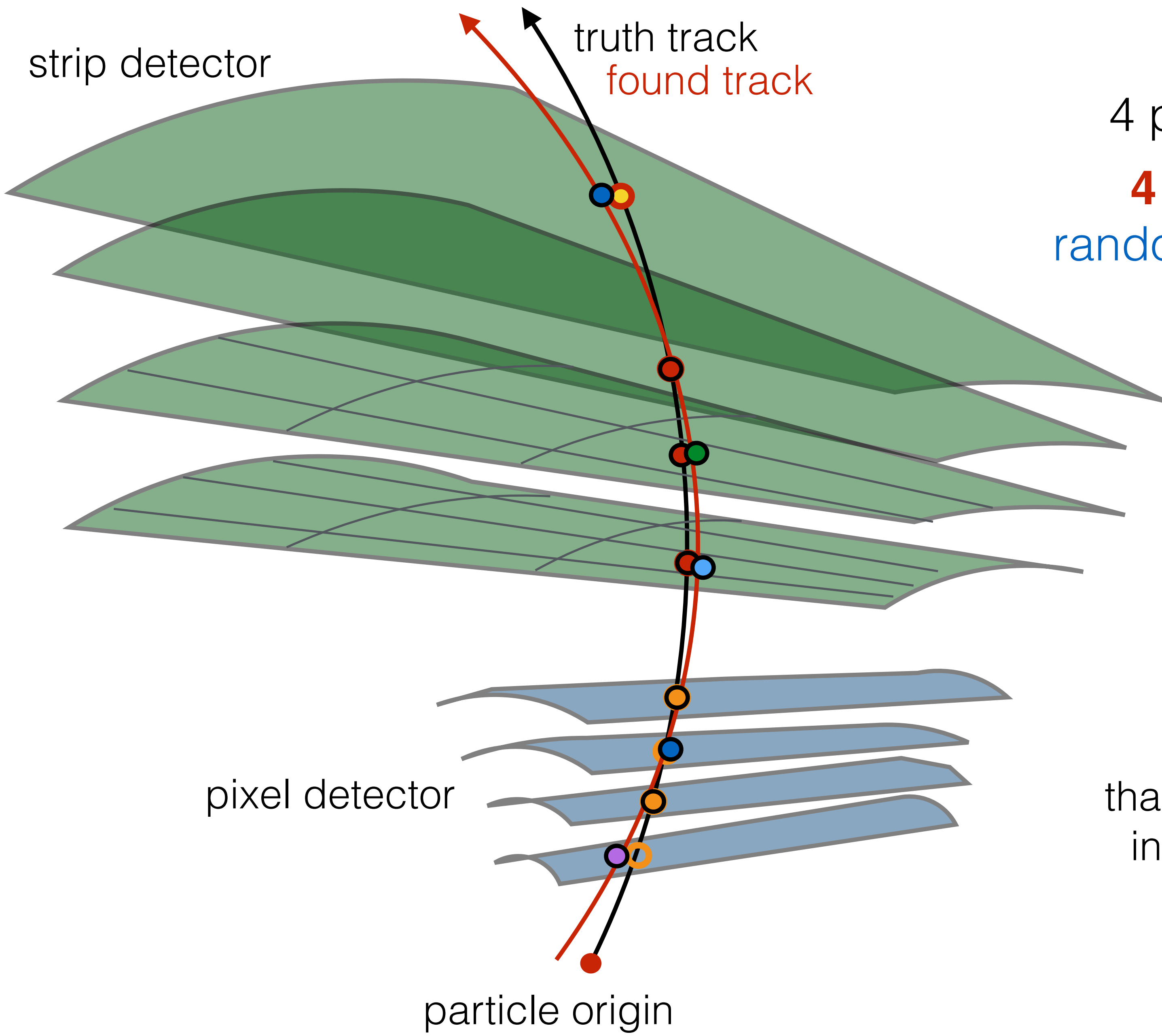
that's not very good
you got 6 out of 8,
naive score = 6/8 = 0.75

your track is rather distorted

did you really measure the particle ?

Illustration:
Track scoring, a rather distorted track. Is this still good ?

Track ranking A ghost (fake) track



4 pixel hits, 4 strip hits created

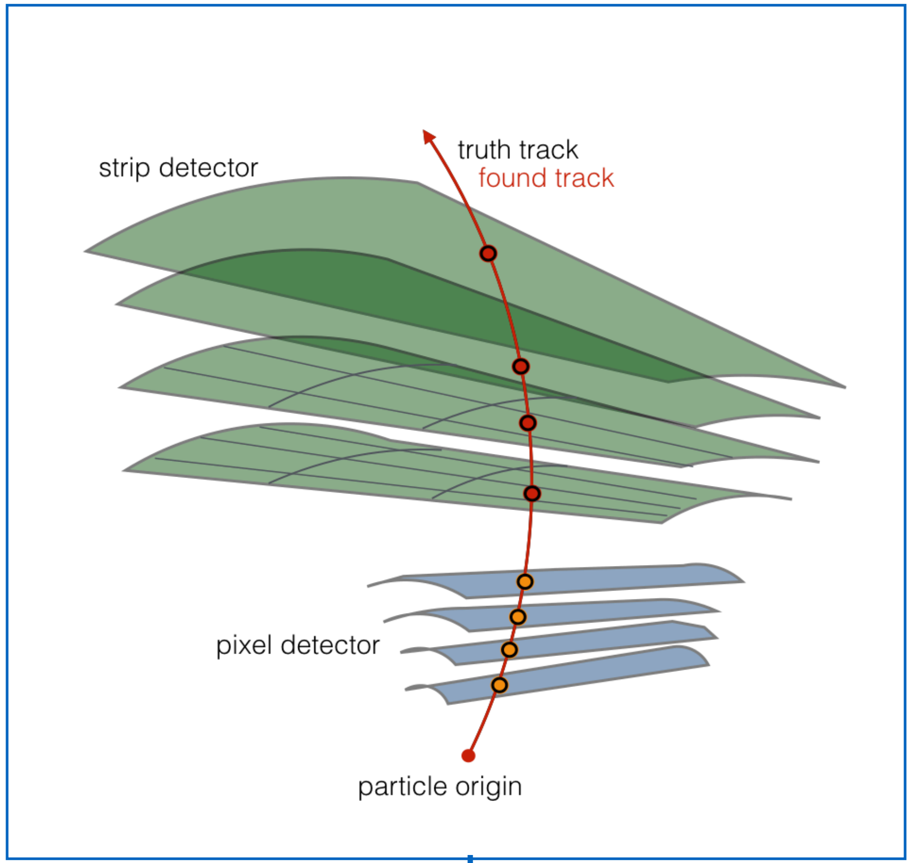
4 pixel hits, **4** strip hits found
randomly associated (3 associated)

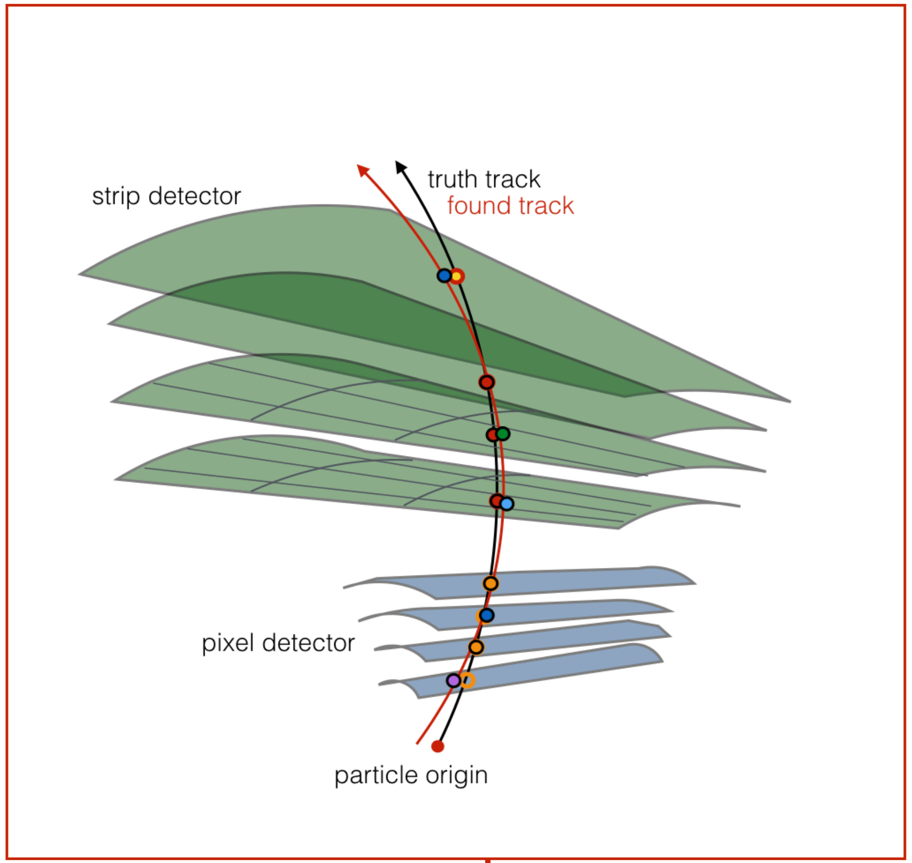
that's garbage
you got 3 out of 8,
naive score = 3/8 = 0.375

your track is **a ghost**

that should not even give you a score !
in fact, it should count as score = -1

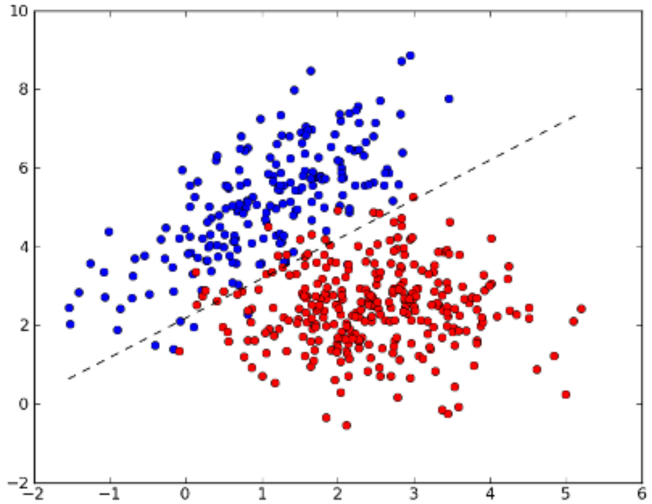
ML Track classification





nhits, nholes, chi2, cluster feature, etc …

nhits, nholes, chi2, cluster feature, etc …



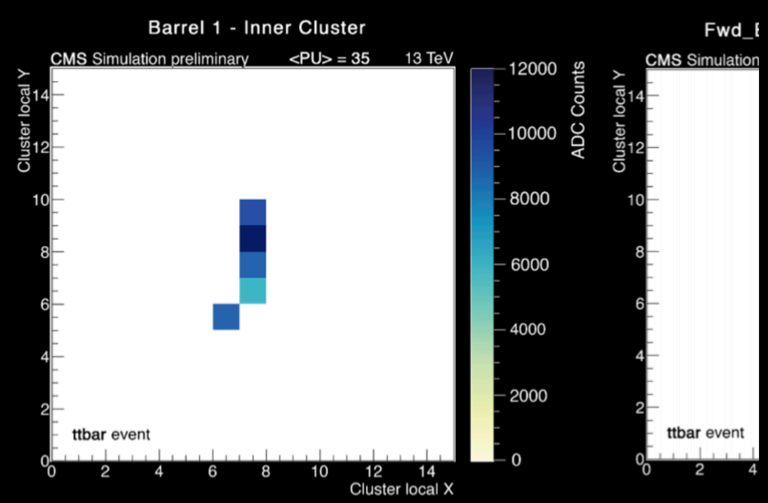
BDT, NN

Illustration:
Track scoring as a classification problem.

ML Track/seed classification

Deep learning tracks – Track seeding

Track building demands a lot of computational resources, so one should choose carefully which seeds to use. Convolutional Neural Networks have shown[2] good results in rejecting seeds that do not correspond to real tracks by comparing the shapes of the hit clusters used in the seed.



[2] A. Di Florio: Convolutional Neural Network for Track Seed Filtering at the CMS HL
https://indico.cern.ch/event/567550/contributions/2638698/

Deep learning tracks – Track quality and classification

Fitted tracks pass through a classifier that rejects fake tracks not corresponding to a real particle. Use of deep neural networks (DNN) instead of boosted decision trees (BDT) as classifiers improves efficiency and reduces the fraction of fake tracks.

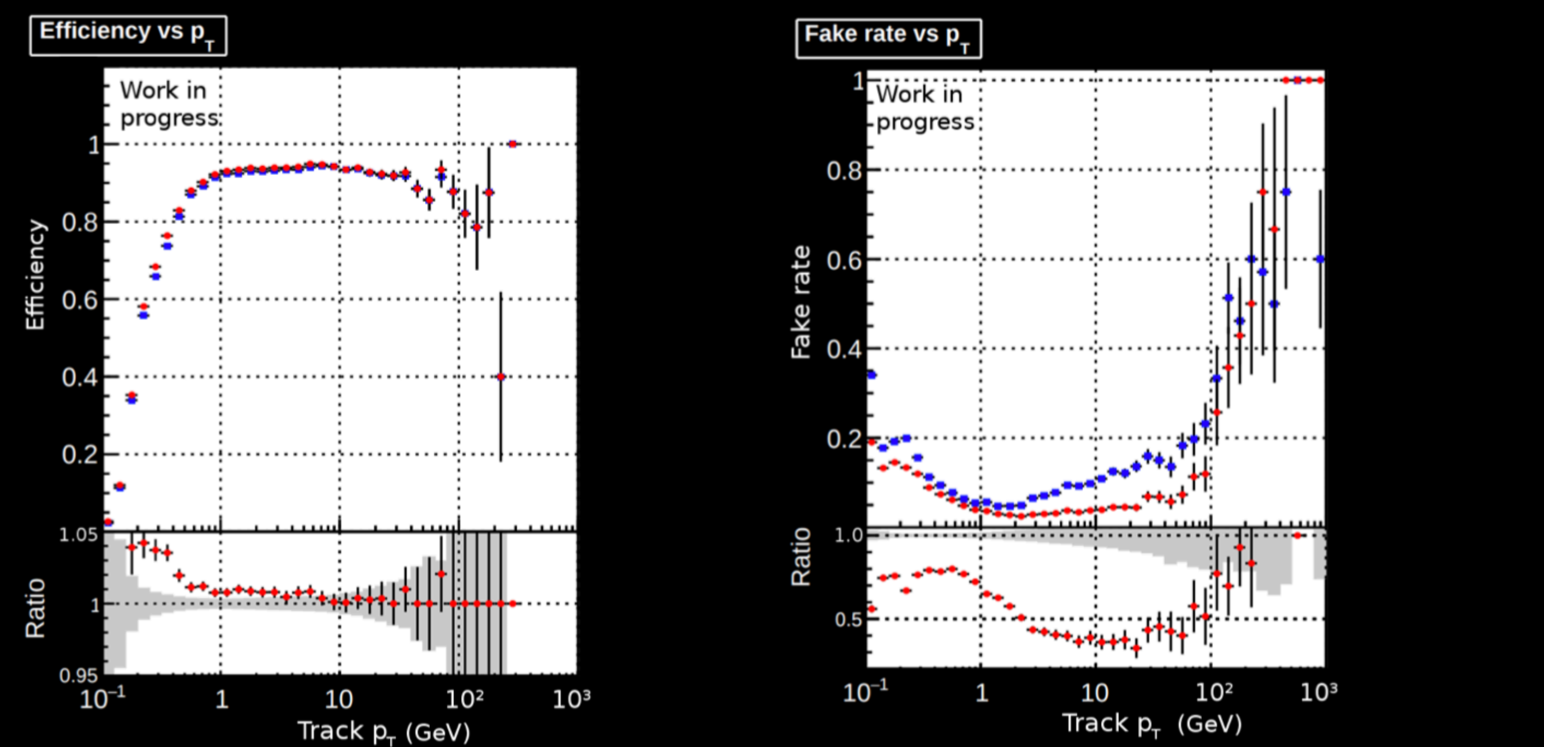


Figure 5: Comparison of DNN and BDT classification on simulated $t\bar{t}$ events with pile-up 50 in CMS. Efficiency (left) and fake rate (right) as a function of track p_T

8

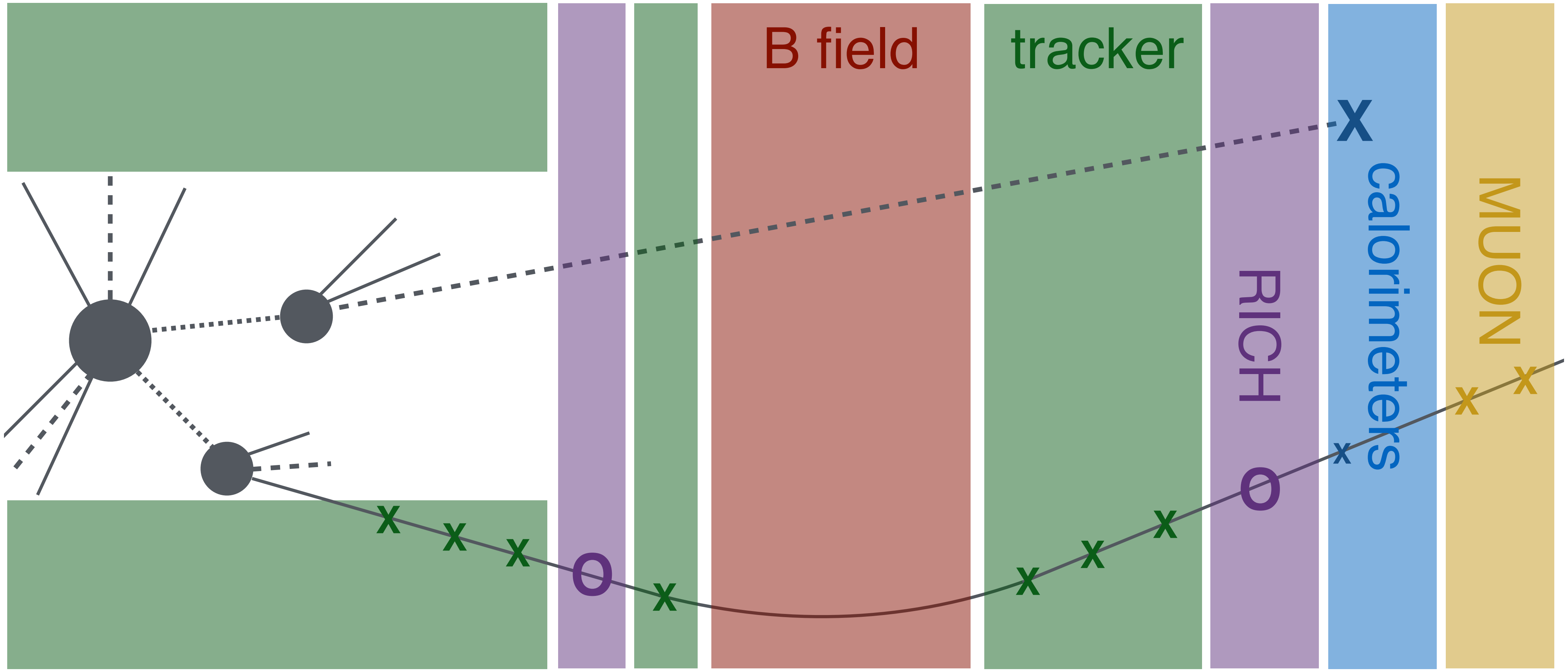
ML Track classification

Track classification is a perfect playing field for ML

- supervised learning application
- ATLAS had a ML based scoring in early 2000s (never used in production)

Fake/ghost track identification

- so called "fake killer" from LHCb
- NN implementation based on hit and hole statistics



Source:
Mike Williams, https://erez.weizmann.ac.il/pls/htmldb/f?p=101:58:::NO:RP:P58_CODE,P58_FILE:5410,Y

ML Track classification

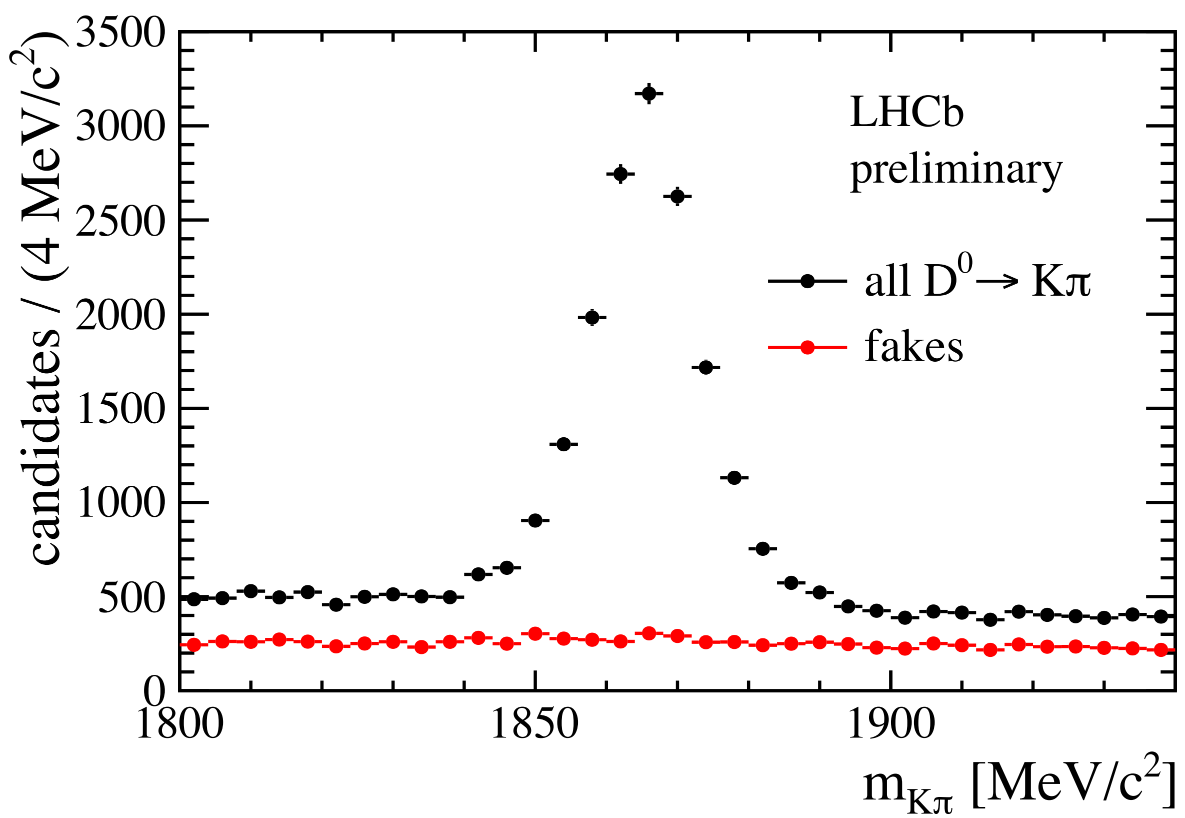
Track classification is a perfect playing field for ML

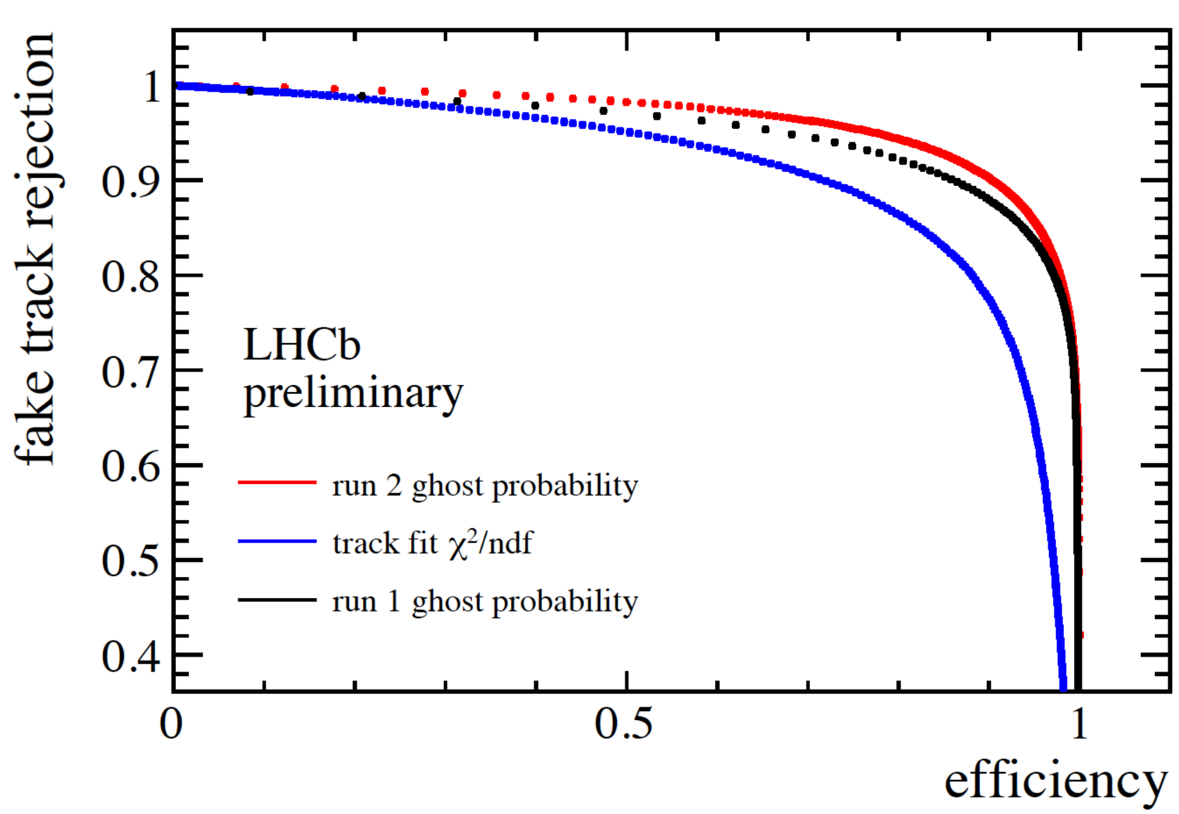
- supervised learning application
- ATLAS had a ML based scoring in early 2000s (never used in production)

Fake/ghost track identification

- so called "fake killer" from LHCb
- NN implementation based on hit and hole statistics

LHCb-PUB-2017-011





Source:
Mike Williams, https://erez.weizmann.ac.il/pls/htmldb/f?p=101:58:::NO:RP:P58_CODE,P58_FILE:5410,Y

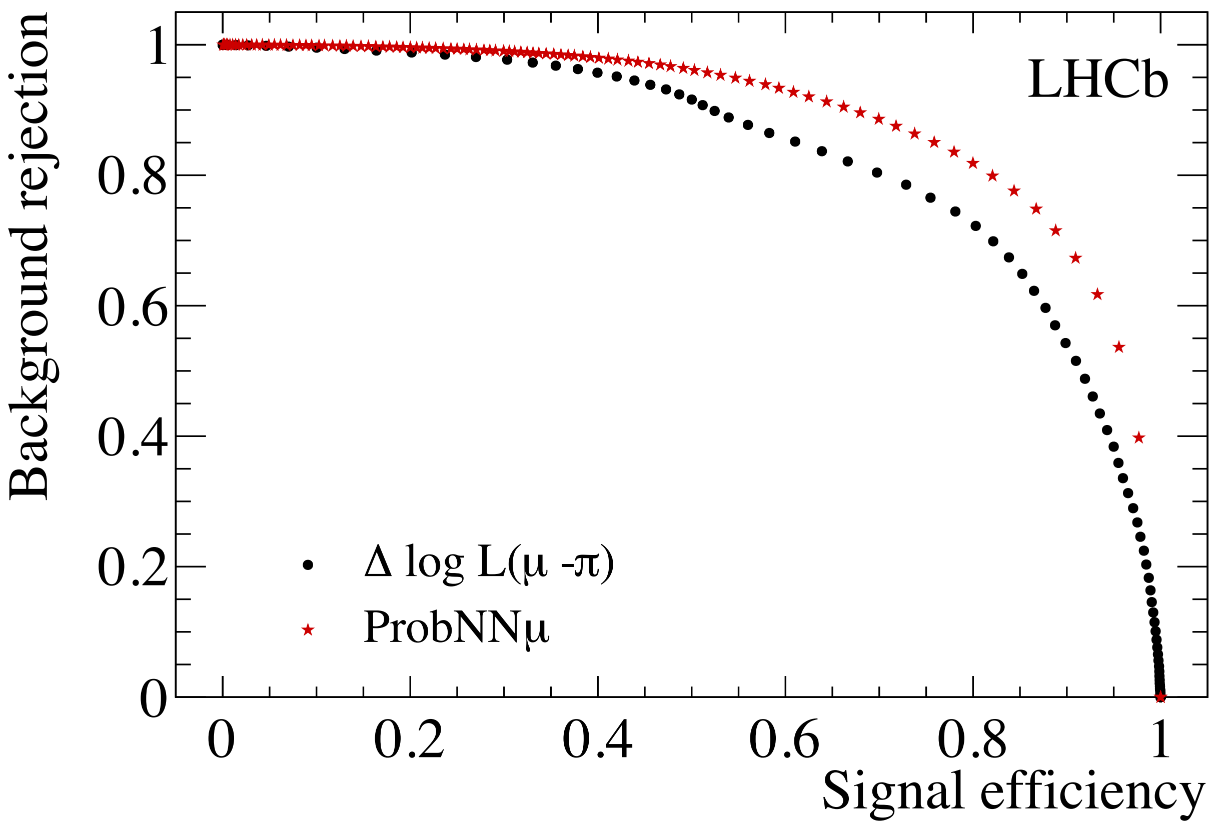
ML Track classification

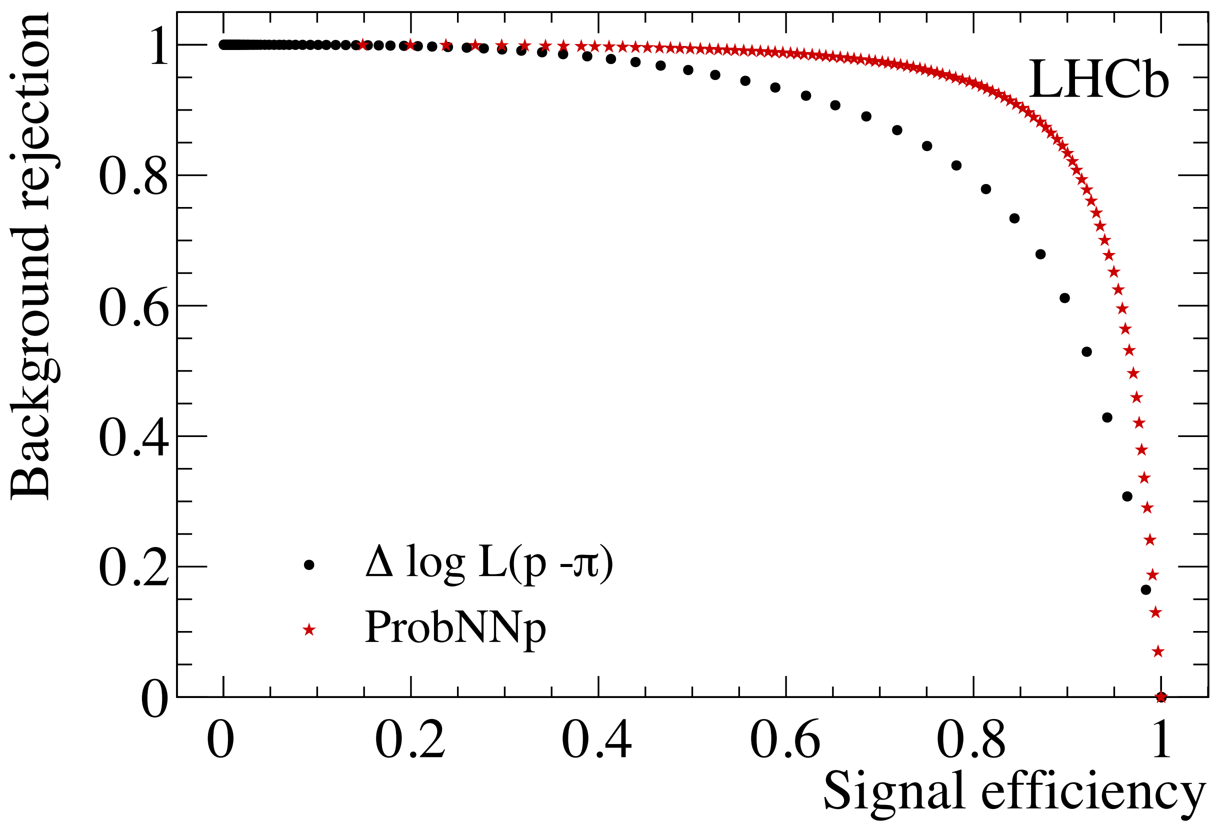
Track classification is a perfect playing field for ML

- supervised learning application
- ATLAS had a ML based scoring in early 2000s (never used in production)

Also, μ/p vs pion identification

- single hidden layer NN trained to perform PID classification





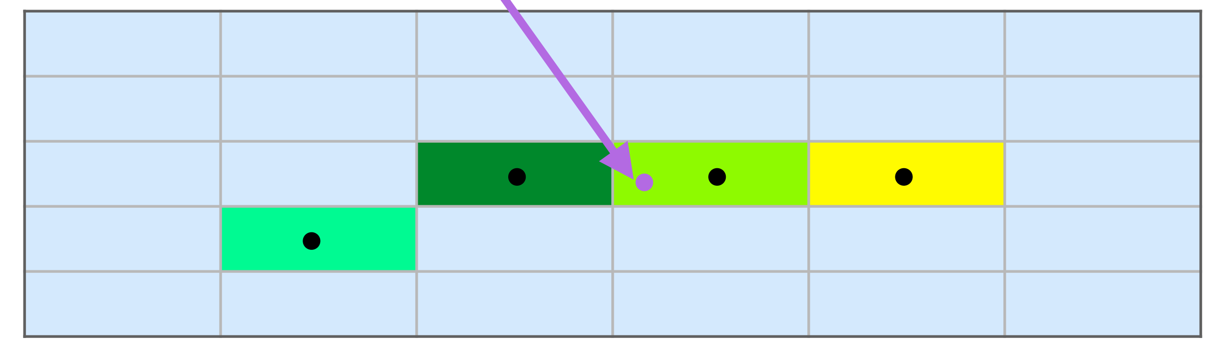
Source:
Mike Williams, https://erez.weizmann.ac.il/pls/htmldb/f?p=101:58:::NO:RP:P58_CODE,P58_FILE:5410,Y

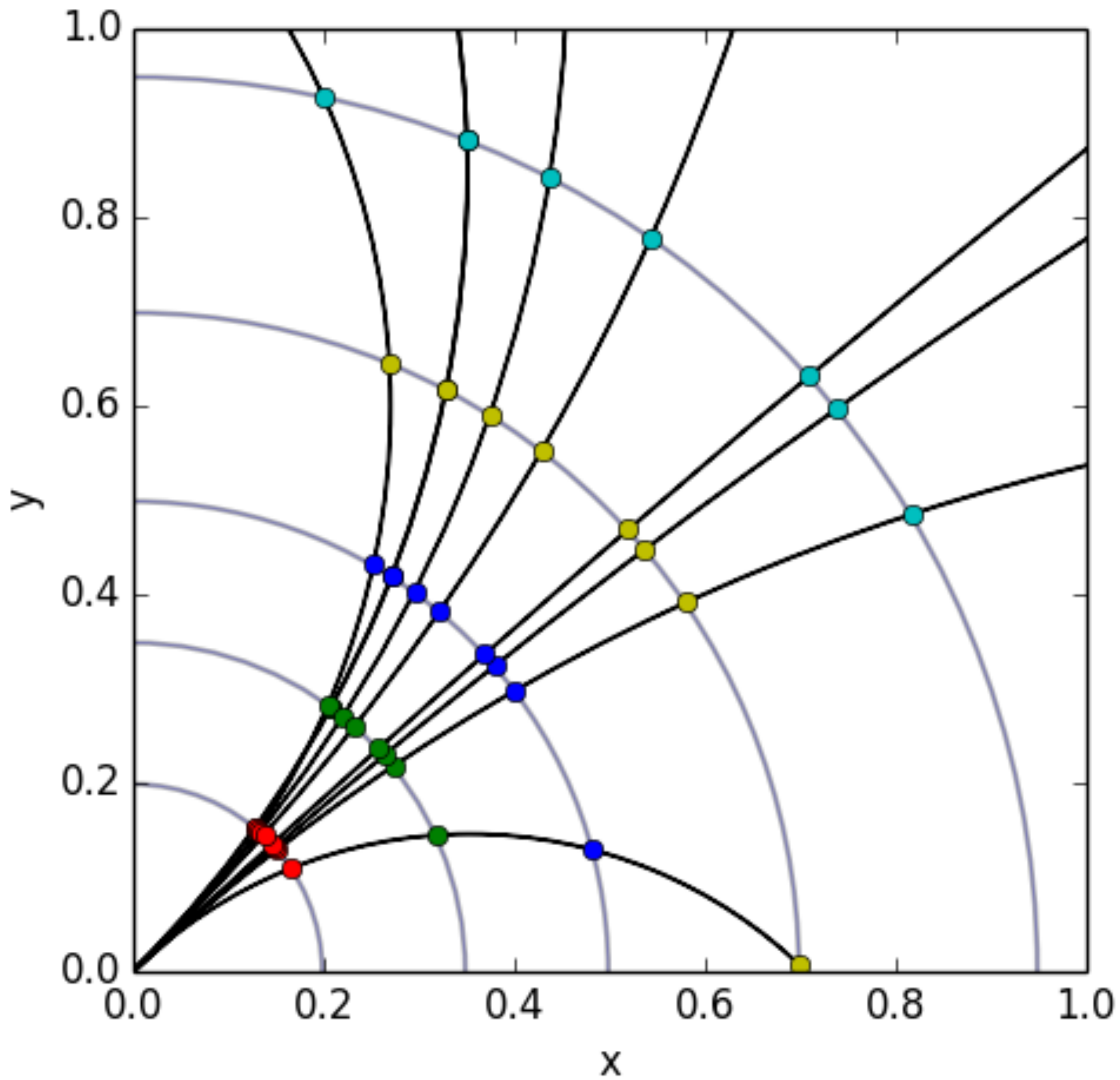
Physics Jet structures

the charge-weighted approach :

$$\mathbf{m} = \frac{1}{\sum_{i=1,N} q_i} \sum_{i=1,N} q_i l_i$$

charge collected in cell





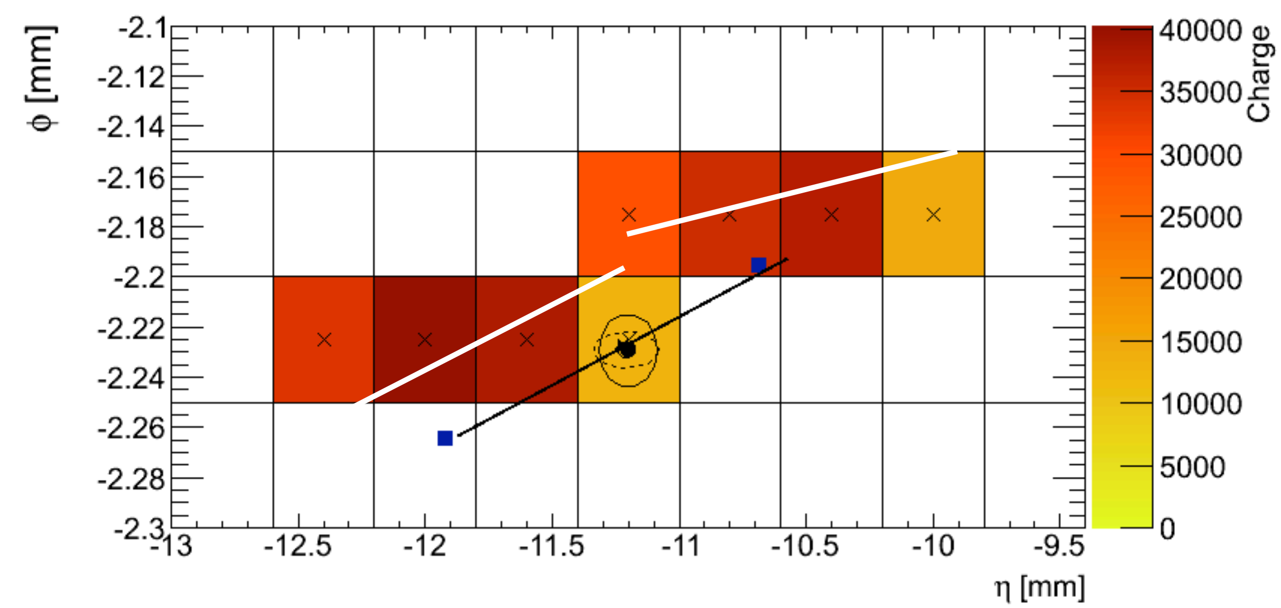
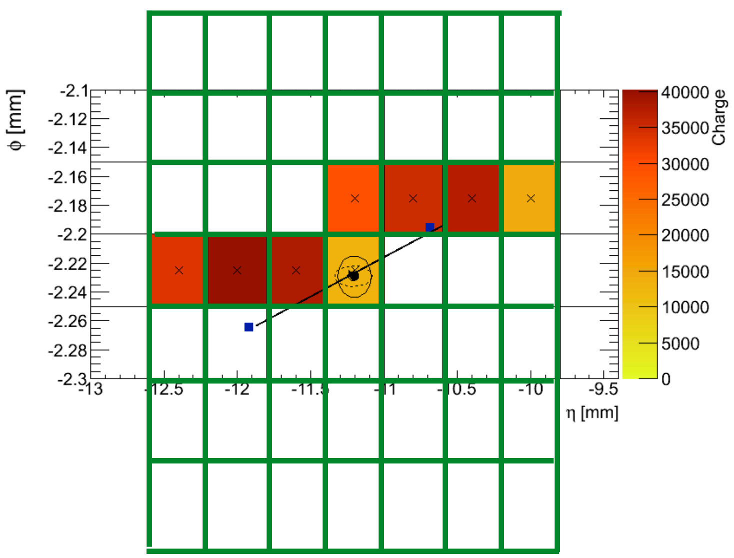


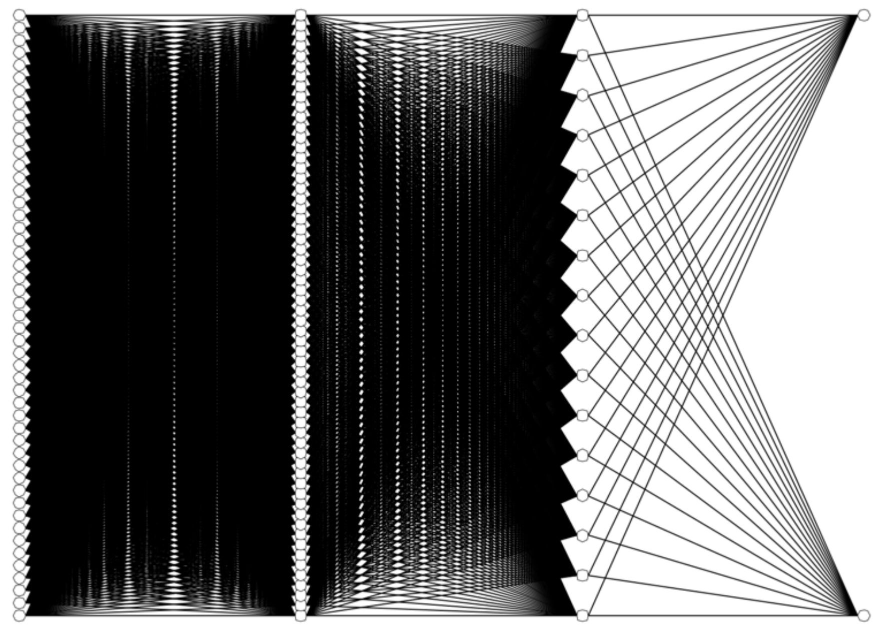
Illustration:
Hits created by a particles jet in a simplified tracking detector, transverse view (left). Illustration of cluster merging (right).

ML Dense environments - shared cluster splitting with ANN

ATLAS pioneered a solution for identifying and eventually even split shared clusters

7x7 pixel charge matrix → 49 input nodes → 2 hidden layers → 2 output layers





→ cluster compatible with 1 particle

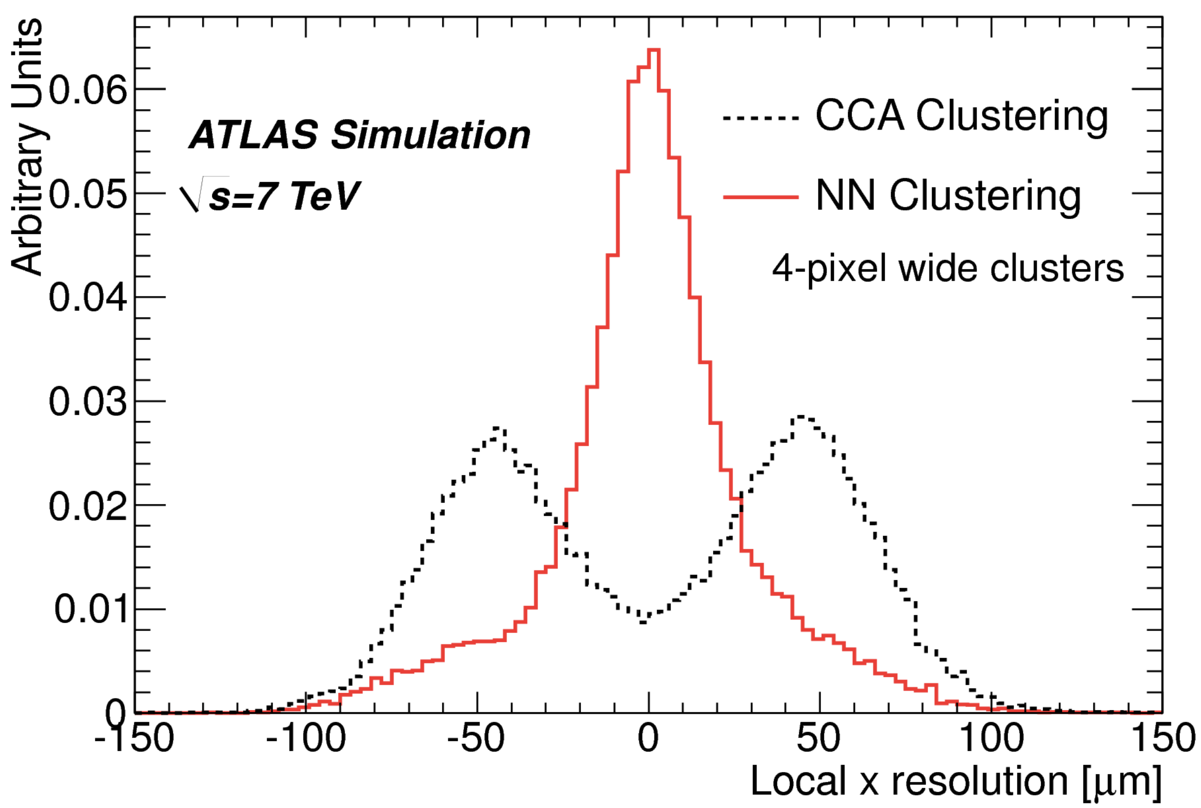
→ cluster compatible with 2 and more particles

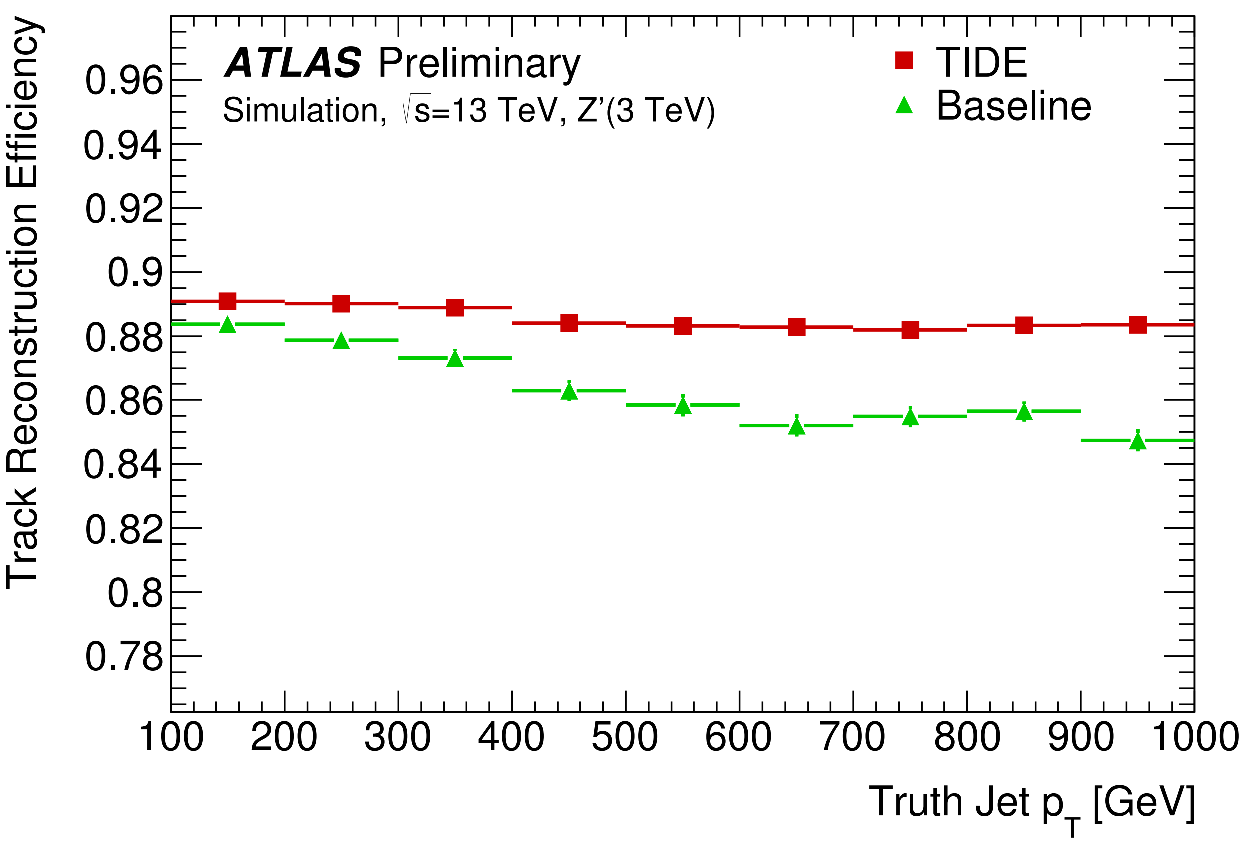
- training an artificial neural network (ANN) with test data from Monte Carlo simulation
- output interpreted as an a posteriori probability
- second set of ANN to estimate particle intersections with sensor & error

See:
The optimization of the ATLAS Track Reconstruction in Dense environments, ATLAS-PHYS-PUB-2015-006

ML Dense environments - shared cluster splitting with ANN

ATLAS pioneered a solution for identifying and eventually even split shared clusters





- regains almost flat reconstruction efficiency in jet cores
- similar performance on data although trained on MC
- what will happen with significant radiation damage in the silicon ?

See:
The optimization of the ATLAS Track Reconstruction in Dense environments, ATLAS-PHYS-PUB-2015-006

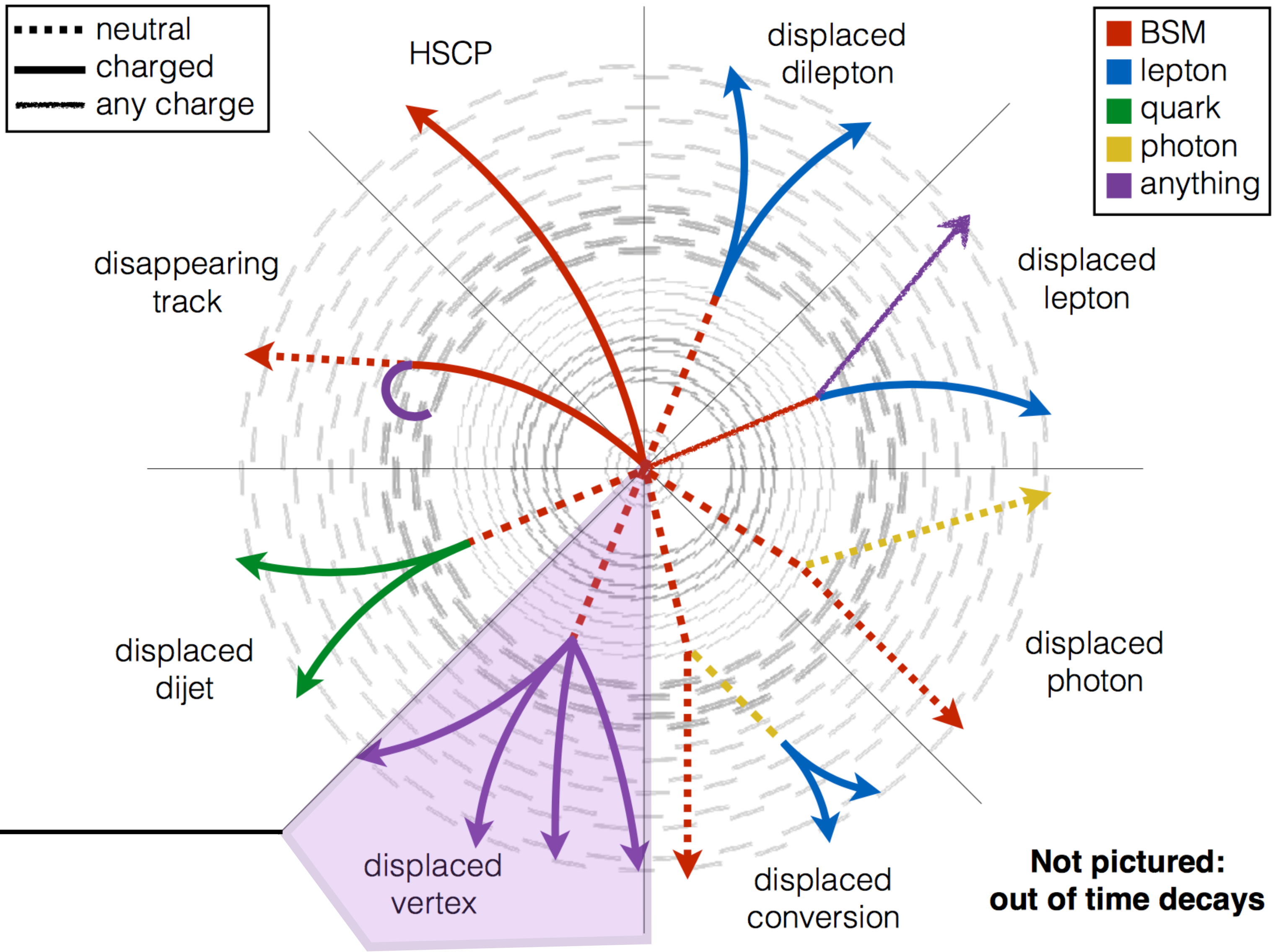
ML Special signatures - environment detection

Special event topologies may need dedicated track reconstruction

- those are usually more CPU intensive
- not feasible to run them on full scan event

Potential

- can we use ML to classify regions ?

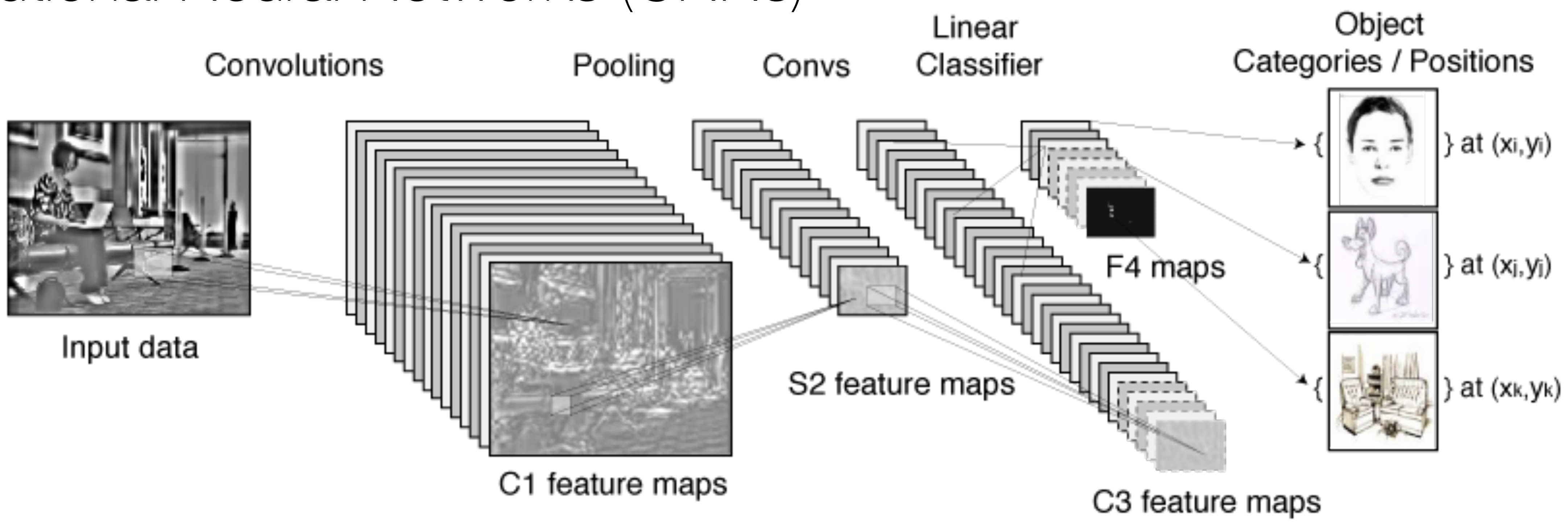


run dedicated displaced vertex tracking in this region:

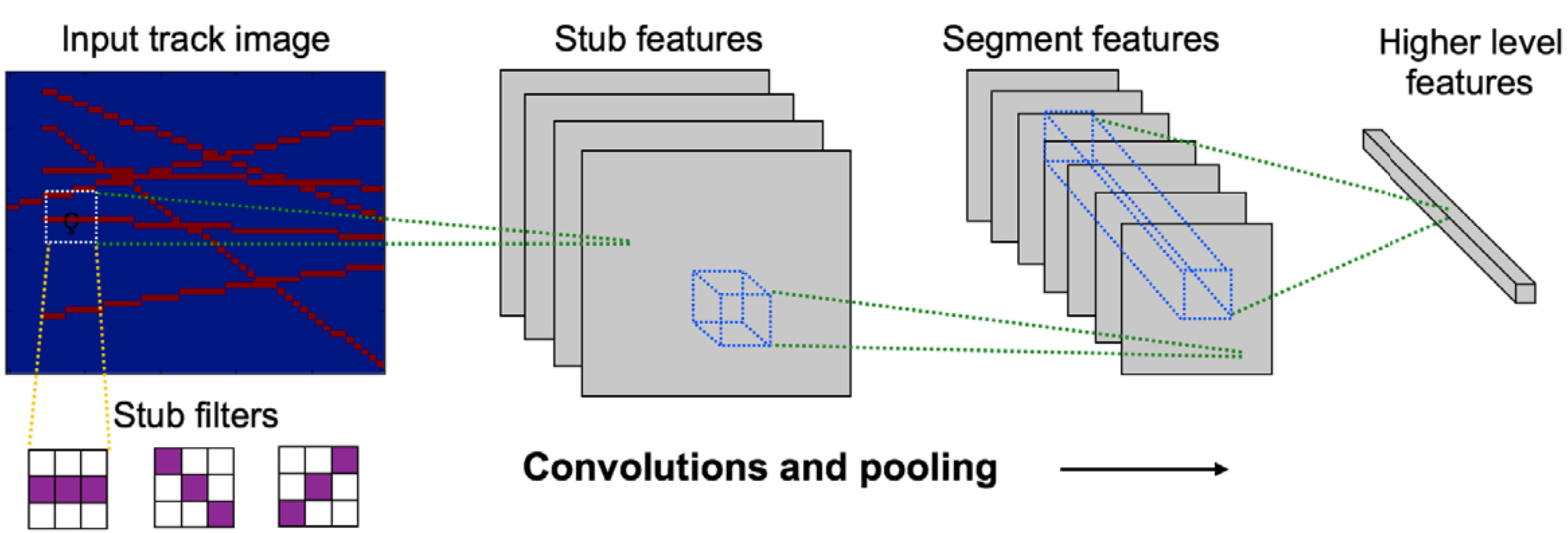
- allow for large impact parameter
- allow for less hits on track

ML End-to-end approach

Convolutional Neural Networks (CNNs)



- for track reconstruction ?

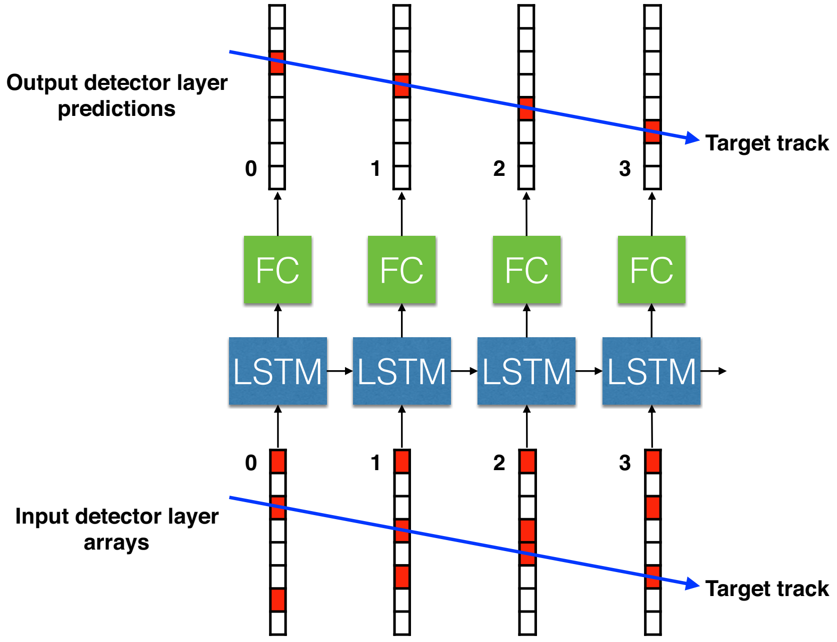


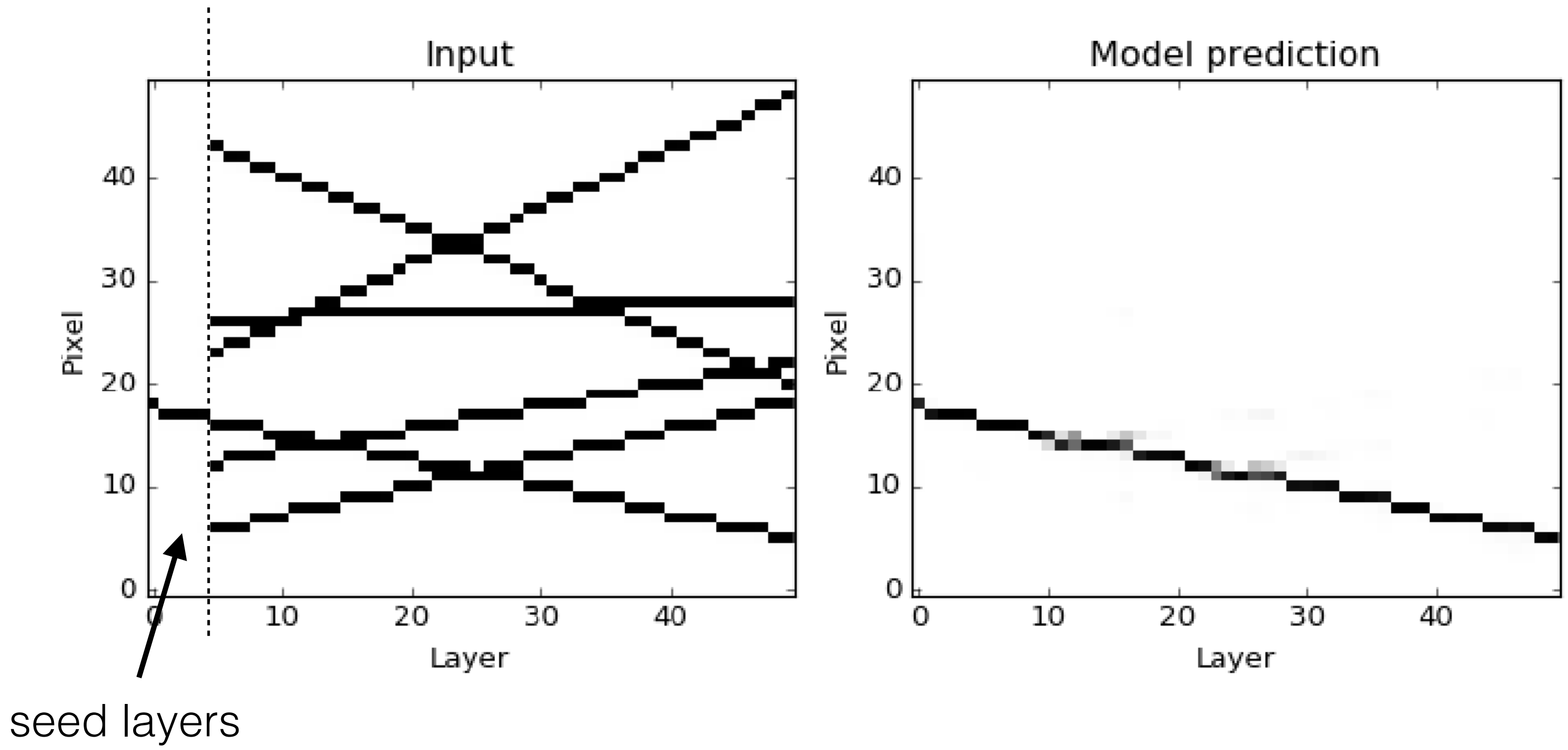
See:
Farrel S. et al, The HEP.TrkX Project: deep neural networks for HL-LHC online and offline tracking, EPJ Web of Conferences **150**, 00003 (2017)

ML End-to-end approach

Recurrent Neural Networks (RNNs)

- e.g. Long Short Term Memory (LSTM) network
- state estimator very similar to the Kalman filter:
  single fully connected (FC) layer with activation,
  uses input to update hidden state that can be used for prediction on target layer,
  i.e. it predict which pixel belongs to the track





See:
Farrel S. et al, The HEP.TrkX Project: deep neural networks for HL-LHC online and offline tracking, EPJ Web of Conferences **150**, 00003 (2017)

Upcoming Tracking challenge hosted on kaggle

- training and test dataset for a mockup detector in HL-LHC environment
  - *particle properties (ID and kinematics)*
  - *created hits and features*
  - *link map <{hit,feature}, particle ID}*
- provide scoring function to rate potential solutions

Stage 1 - Feb/Mar 2018:
optimise track finding score

Stage 2 - Q2/3 2018:
optimise track finding time

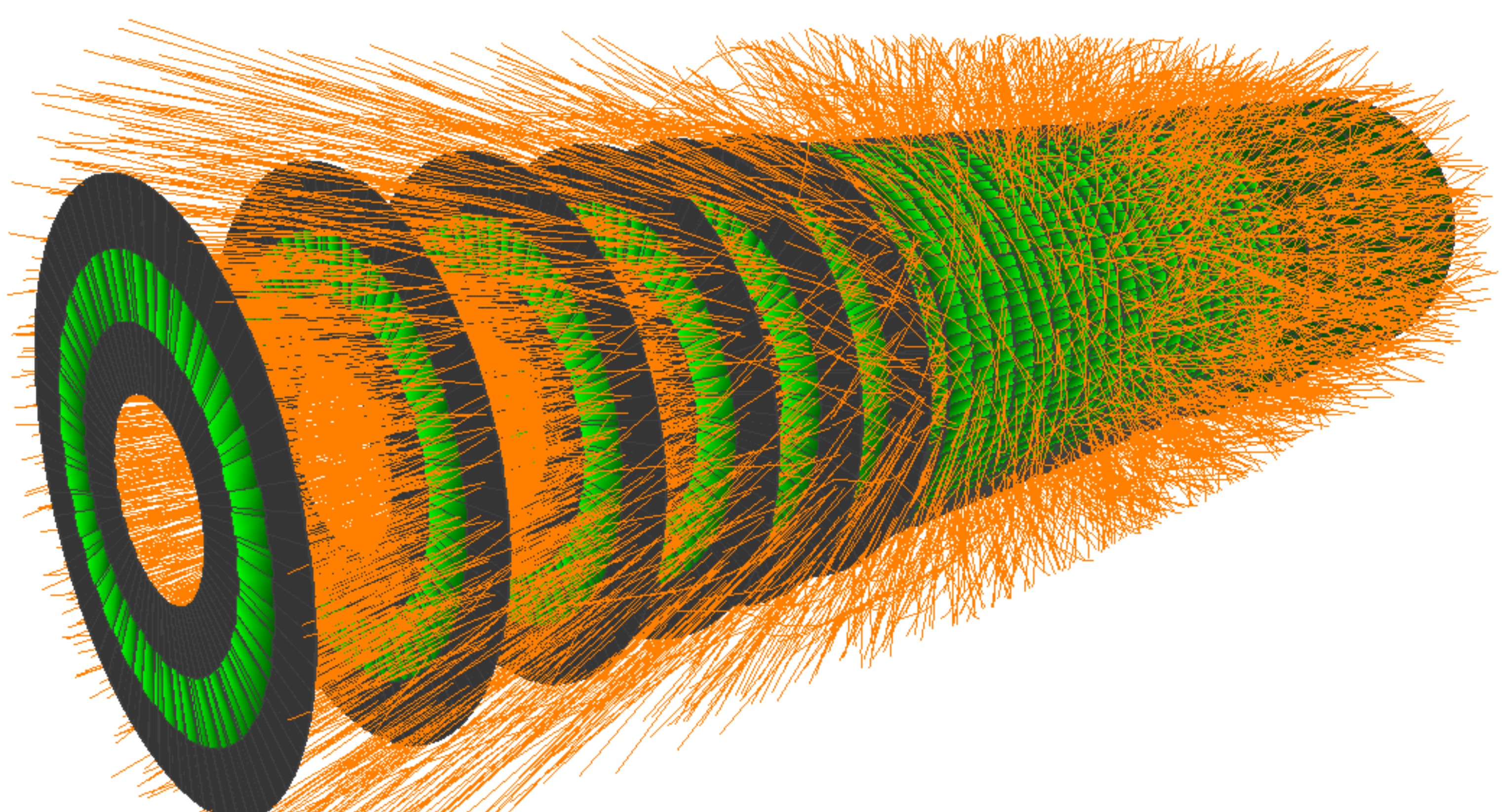
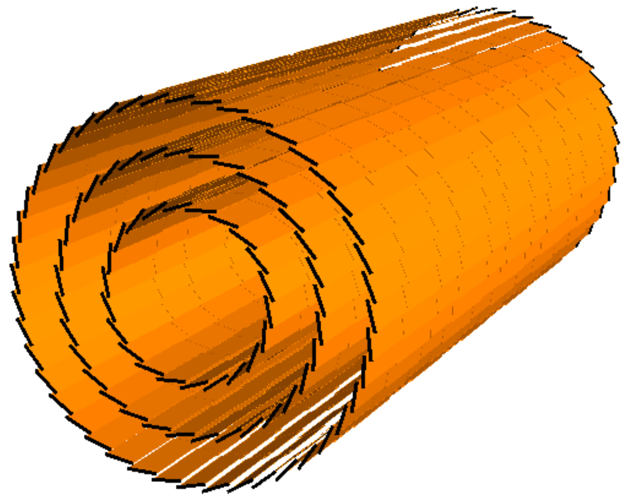
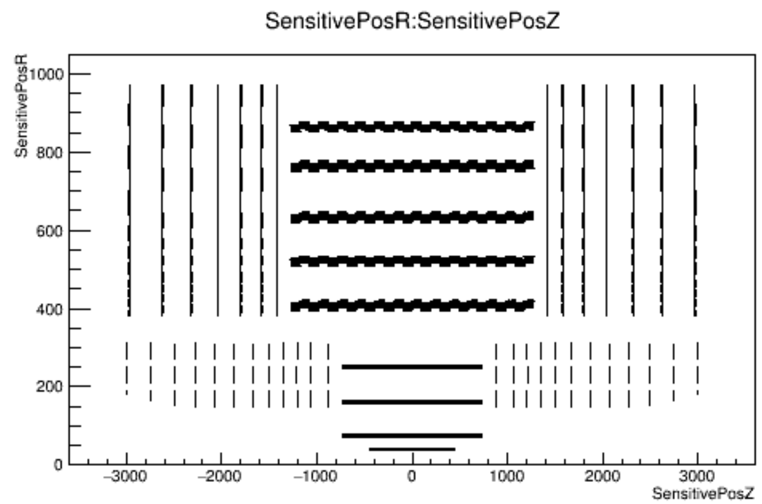
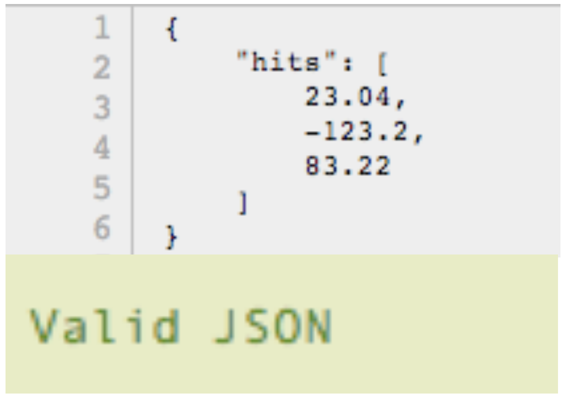


Illustration:
Botton left: simulated event with very high event pileup (μ = 1000) showing only particles with transverse momentum higher than 250 MeV.



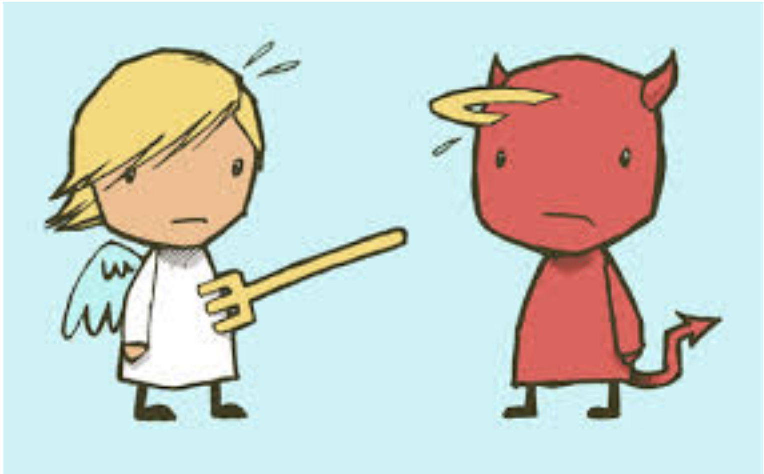


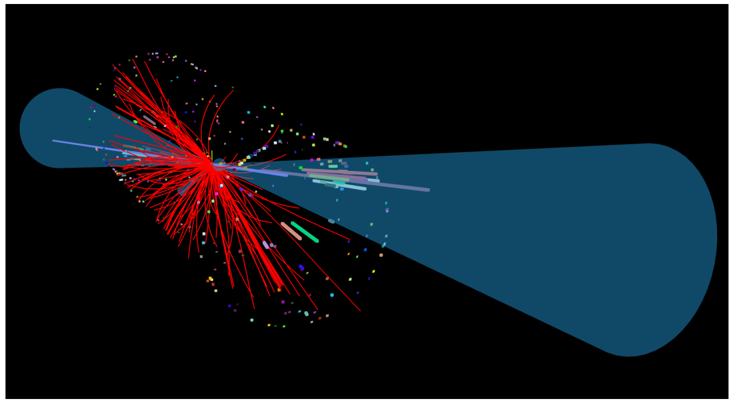


detector geometry
planar barrel/EC type detector
pixel/strip system

simulation
with the possibility to
simplify where possible

event data
easily readable,
platform independent





well defined goal
what is success
and how we measure it

visualisation
of geometry,
hits & found tracks

different categories
for different
solutions

Summary Conclusion

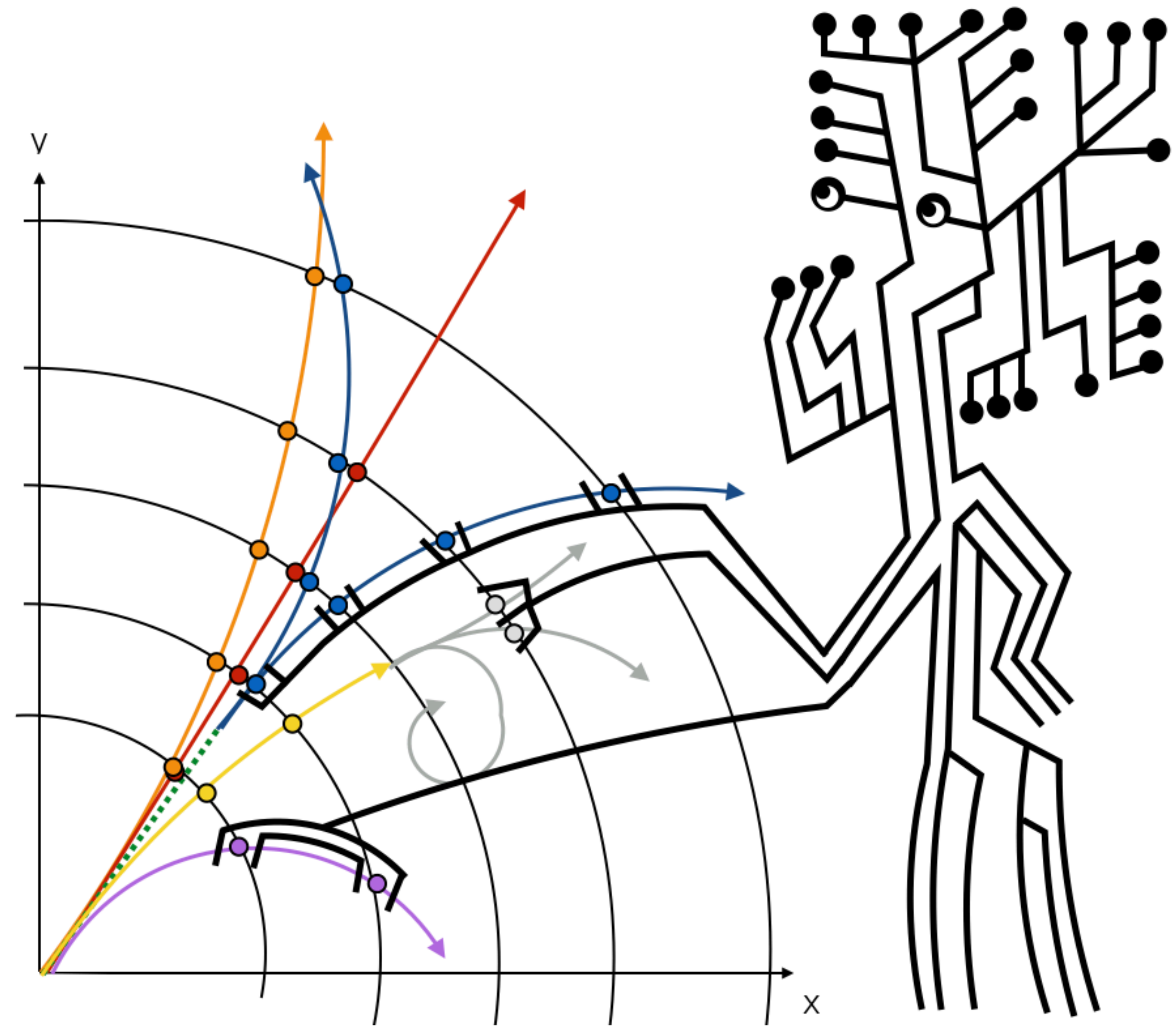
Track reconstruction is a natural playing field for ML/DS

- it's also not new to our field
  (we may just have labeled it differently)
- unsupervised learning: clustering
- supervised learning: classification
- interference

Recent boost in ML

- we should (and will) profit from it
- we will have to learn some new
  language (AuC vs. Integral)

Watch out for the Tracking
ML challenge on kaggle

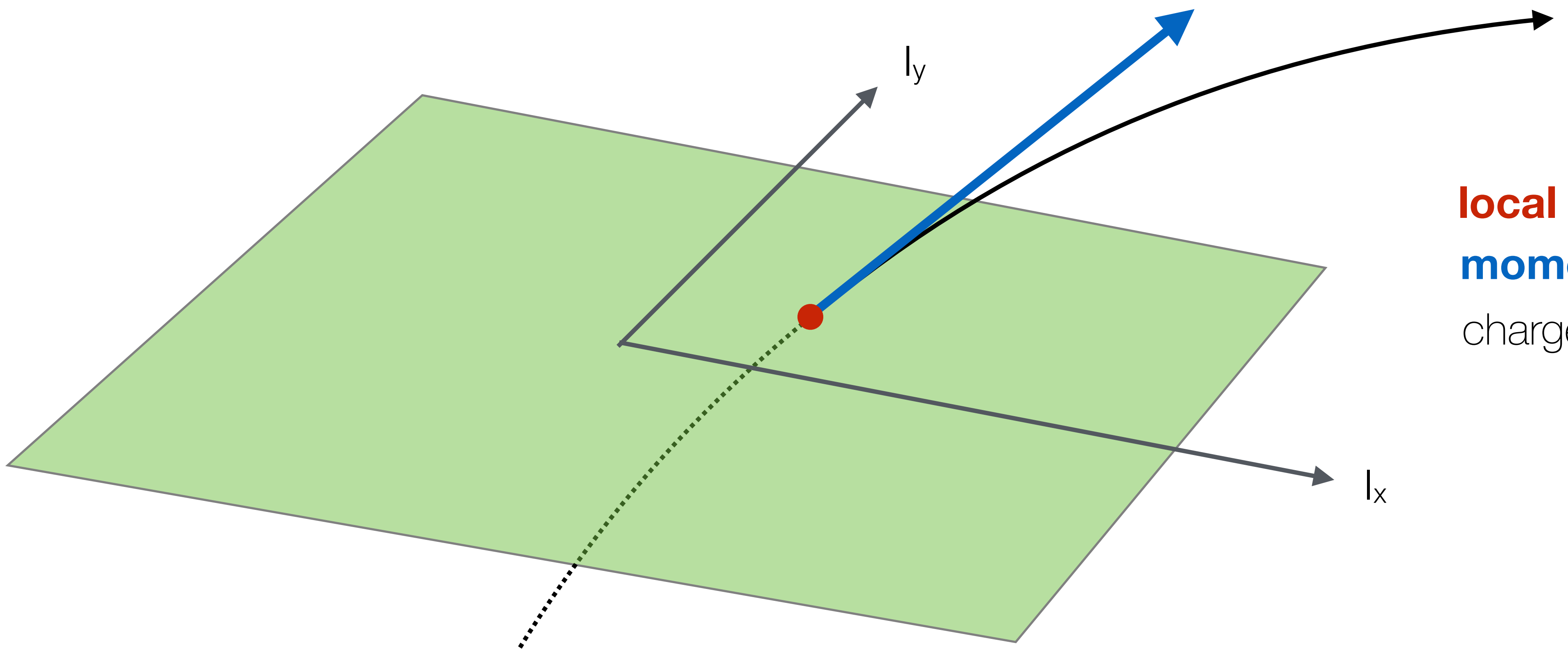


Backup Slides

Definitions Track parameterisation

Charged particle trajectory parameterisation

- five parameters needed to describe a trajectory localisation on a surface



local position on surface

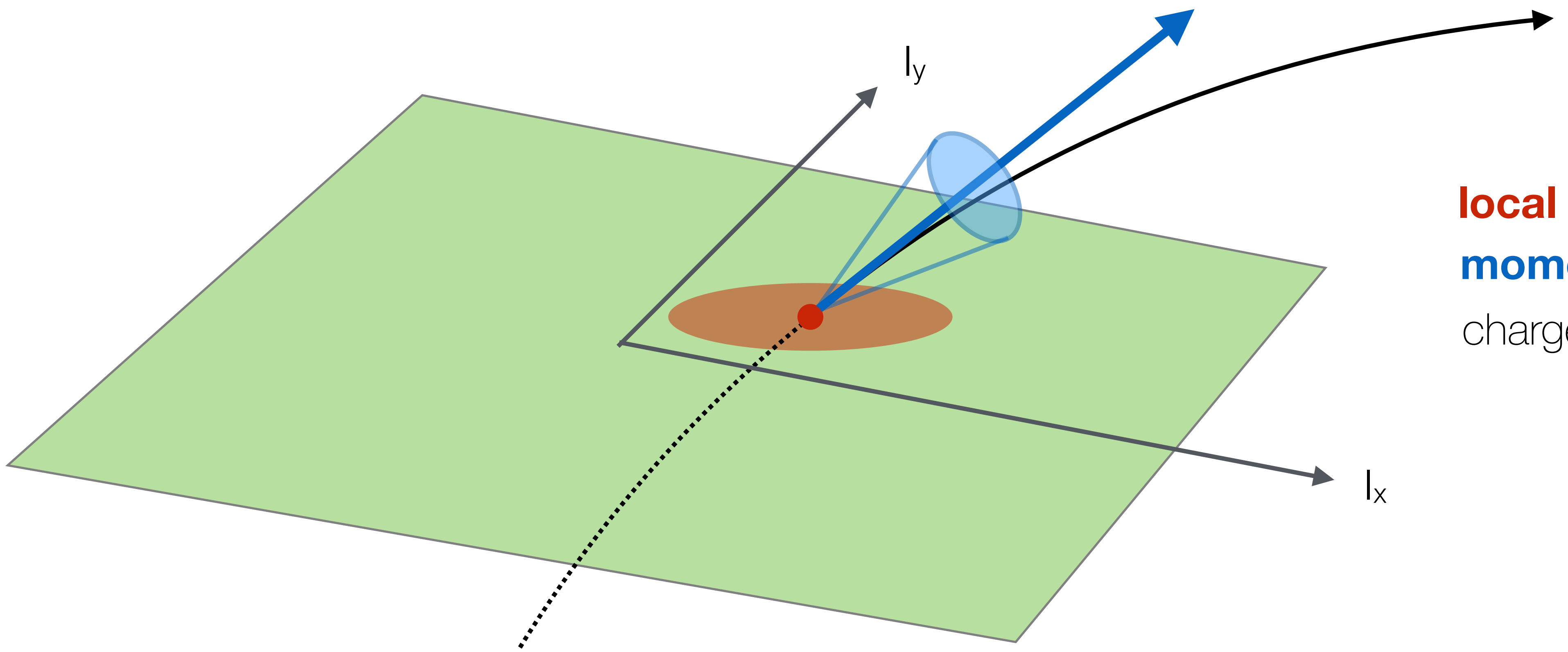
momentum

charge

$$\mathbf{q} = (l_1, l_2, \phi, \theta, q/p)$$

Definition Track parameterisation

Obviously, every measurement has associated errors



local position on surface

momentum

charge

$$\mathbf{q} = (l_1, l_2, \phi, \theta, q/p)$$

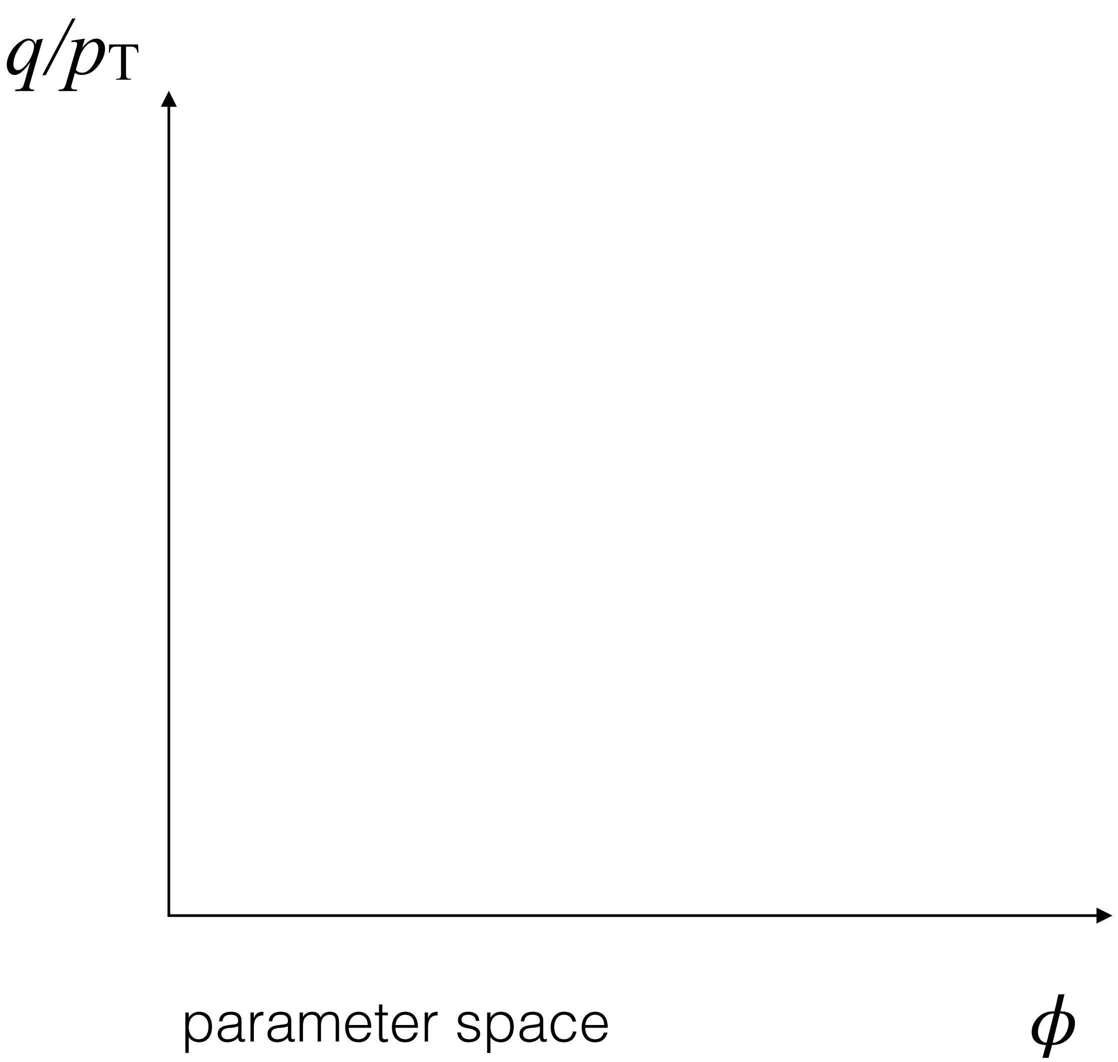
$$\mathbf{C} = \begin{pmatrix} \sigma^2(l_1) & cov(l_1,l_2) & cov(l_1,\phi) & cov(l_1,\theta) & cov(l_1,q/p) \\ . & \sigma^2(l_2) & cov(l_2,\phi) & cov(l_2,\theta) & cov(l_2,q/p) \\ . & . & \sigma^2(\phi) & cov(\phi,\theta) & cov(\phi,q/p) \\ . & . & . & \sigma^2(\theta) & cov(\theta,q/p) \\ . & . & . & . & \sigma^2(q/p) \end{pmatrix}$$

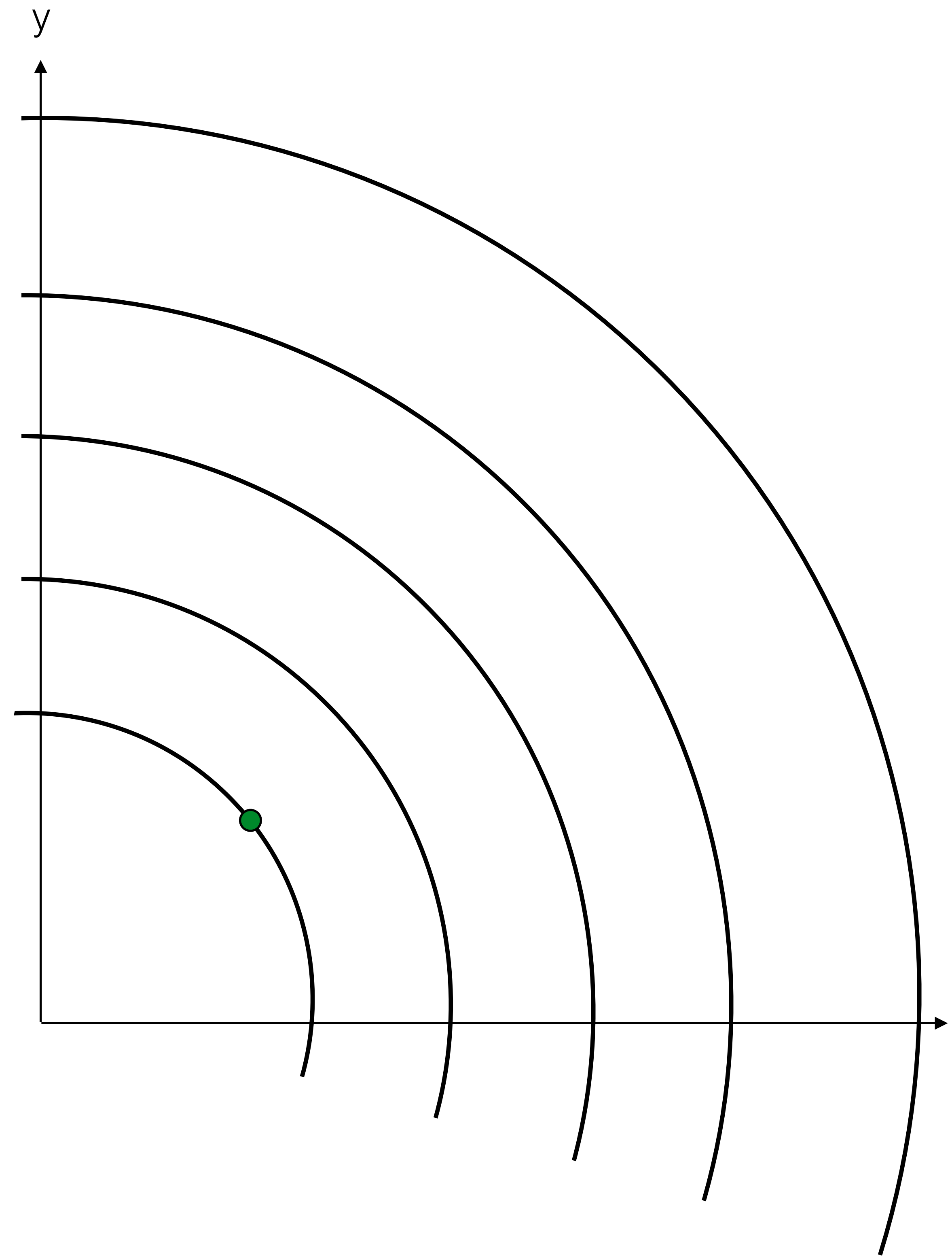
Conformal mapping techniques

Hough transform

- transform your track hits in the x, y space

$$\mathbf{q} = (d_0, z_0, \phi, \theta, q/p_{\mathrm{T}})$$



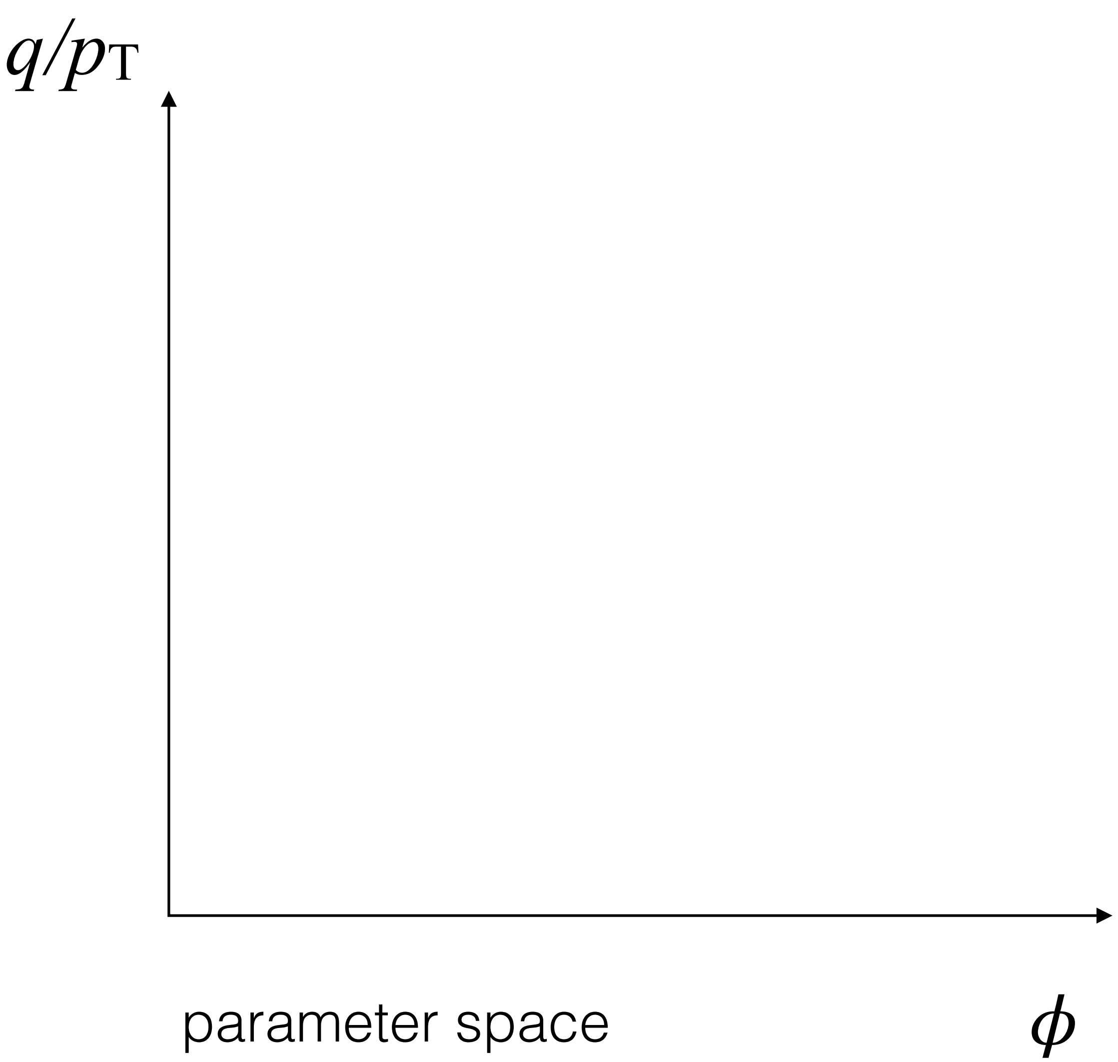


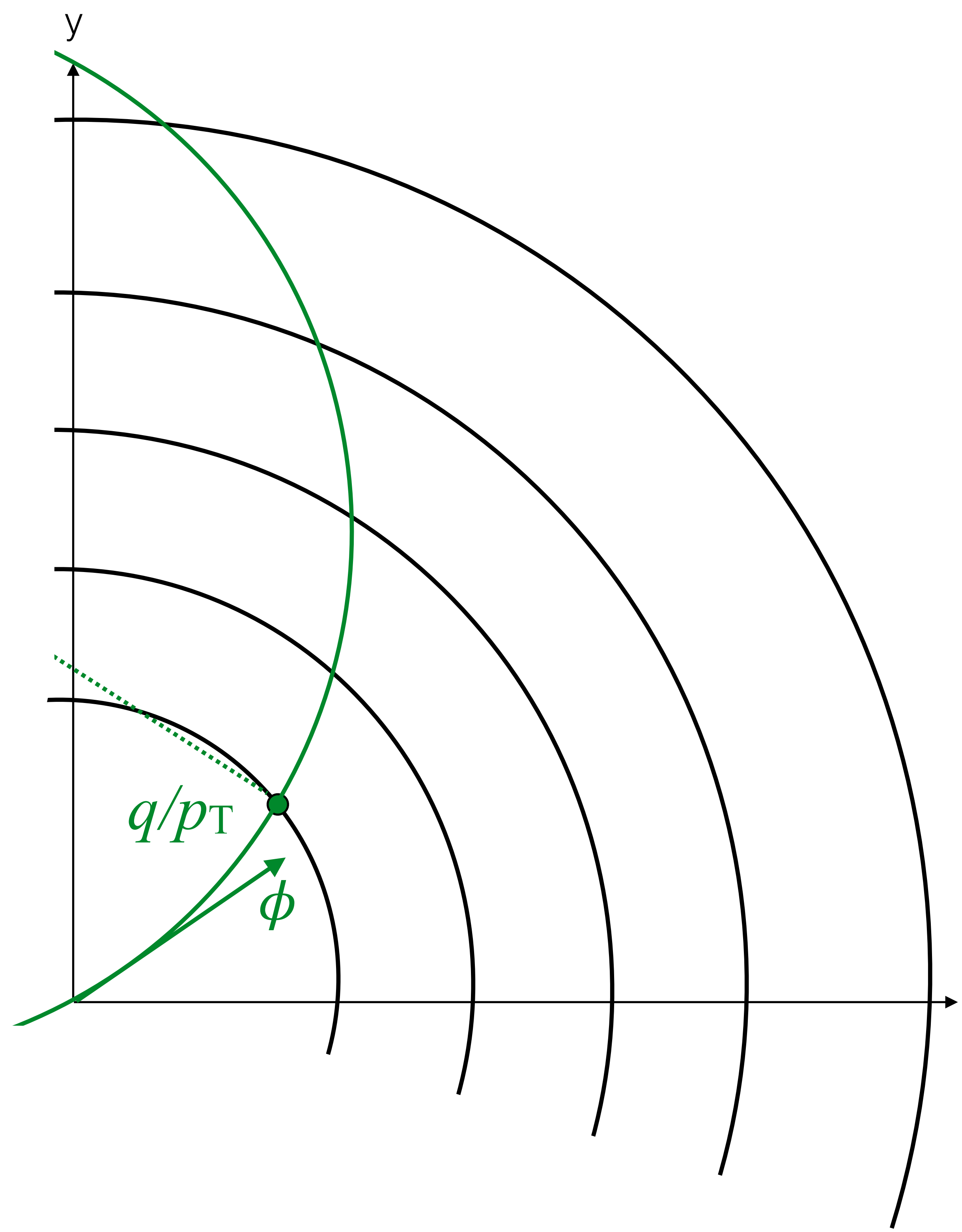
Conformal mapping techniques

Hough transform

- transform your track hits in the x, y space

$$\mathbf{q} = (d_0, z_0, \phi, \theta, q/p_T)$$



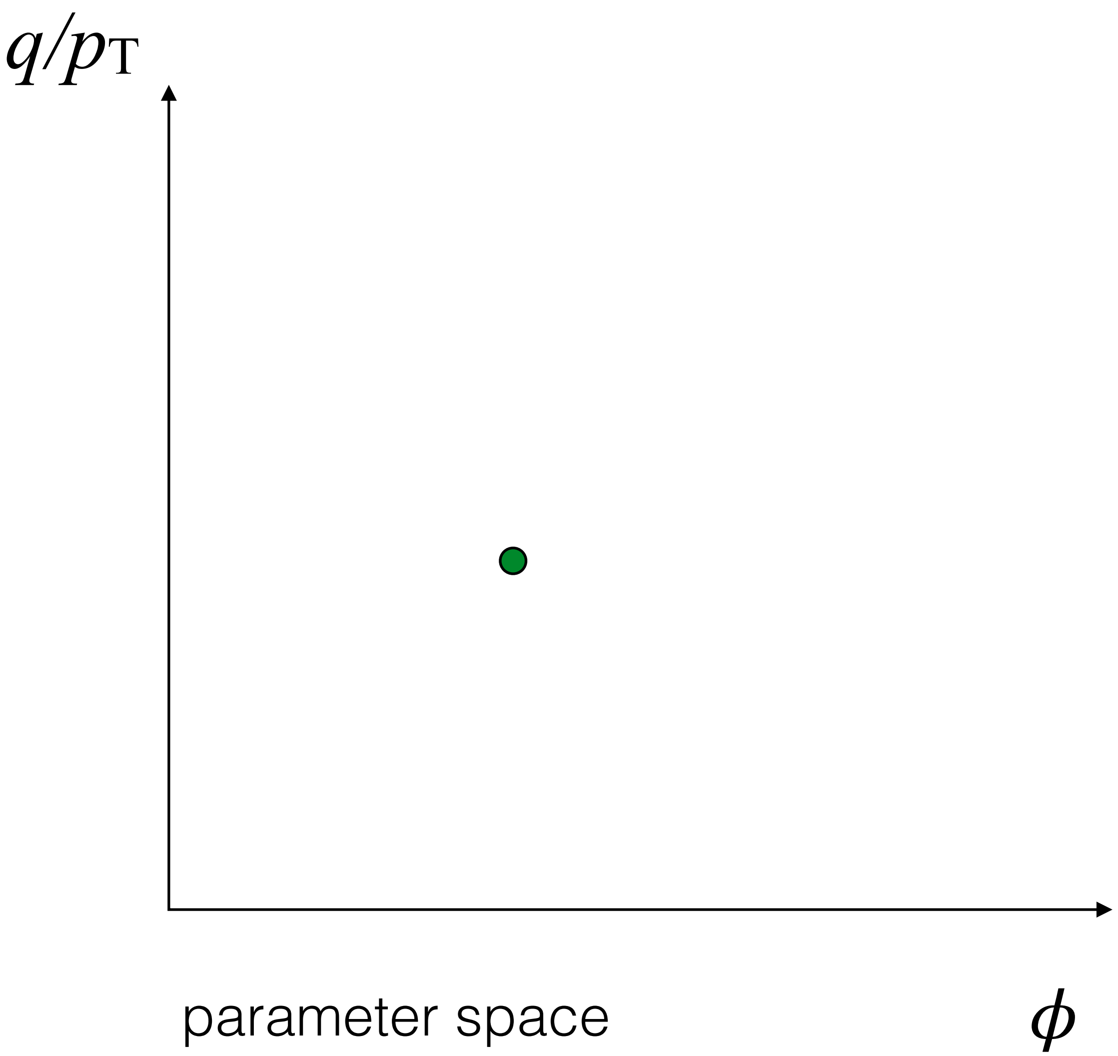


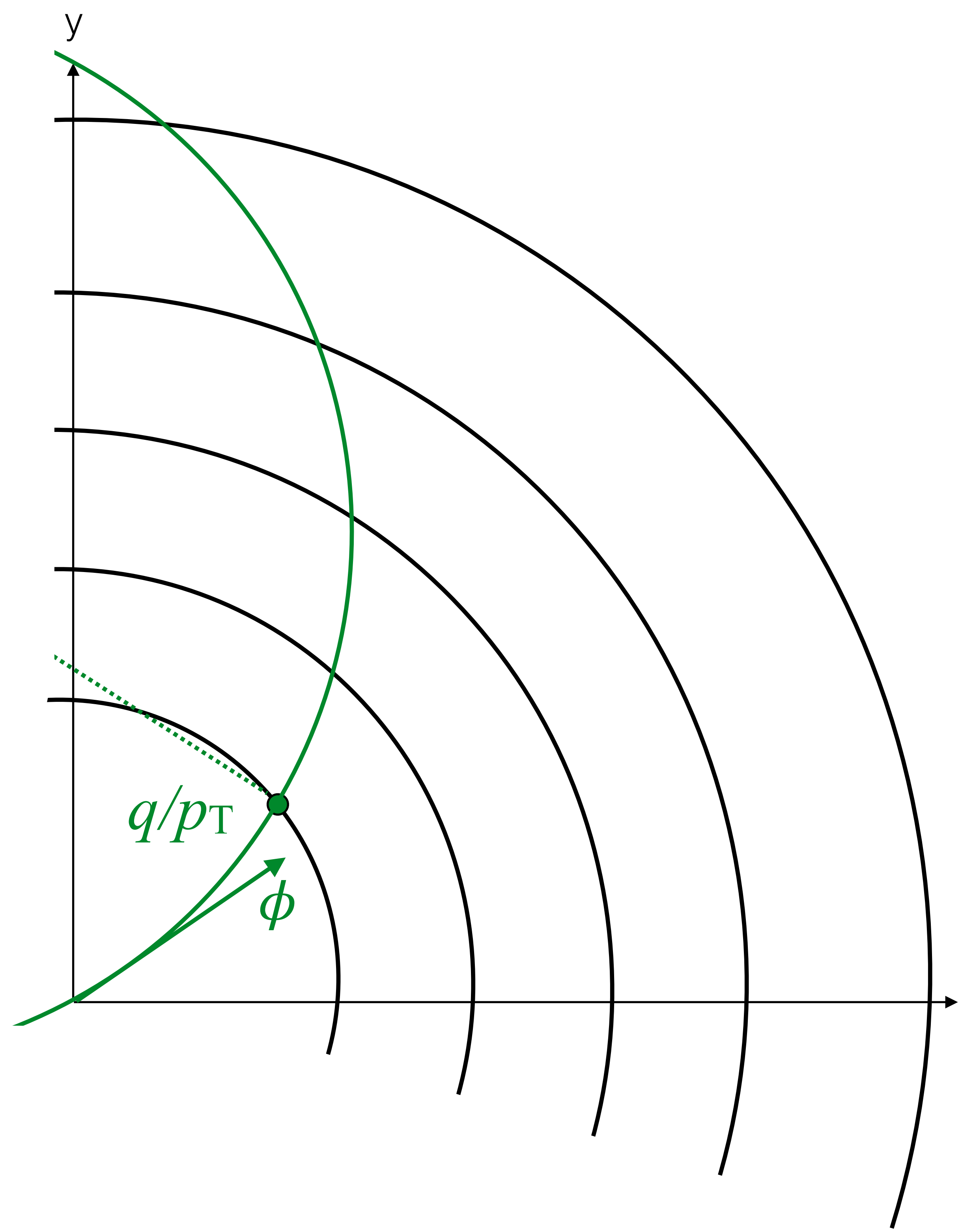
Conformal mapping techniques

Hough transform

- transform your track hits in the x, y space

$$\mathbf{q} = (\cancel{d_0}, \cancel{z_0}, \phi, \cancel{\theta}, q/p_{\mathrm{T}})$$



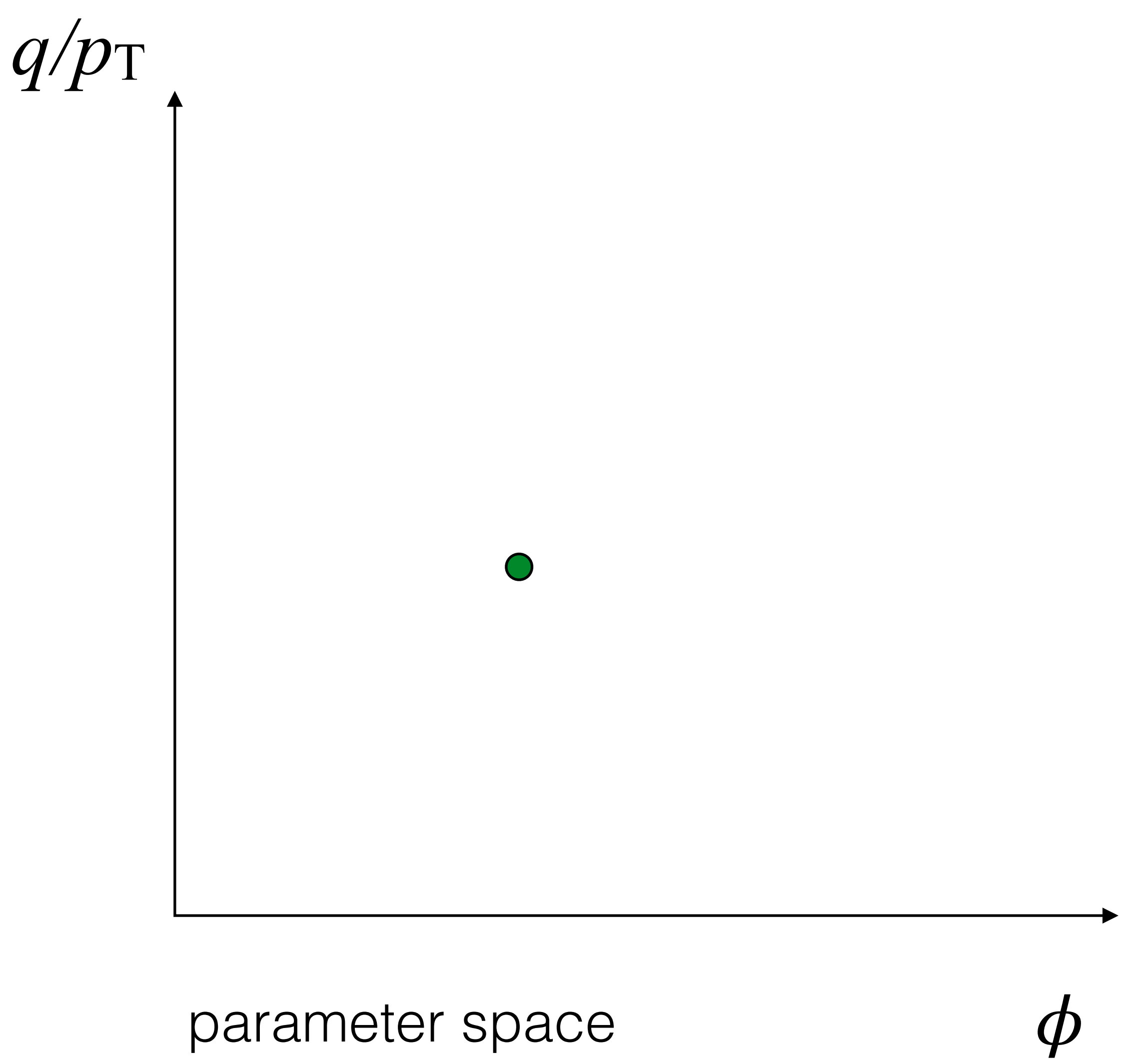


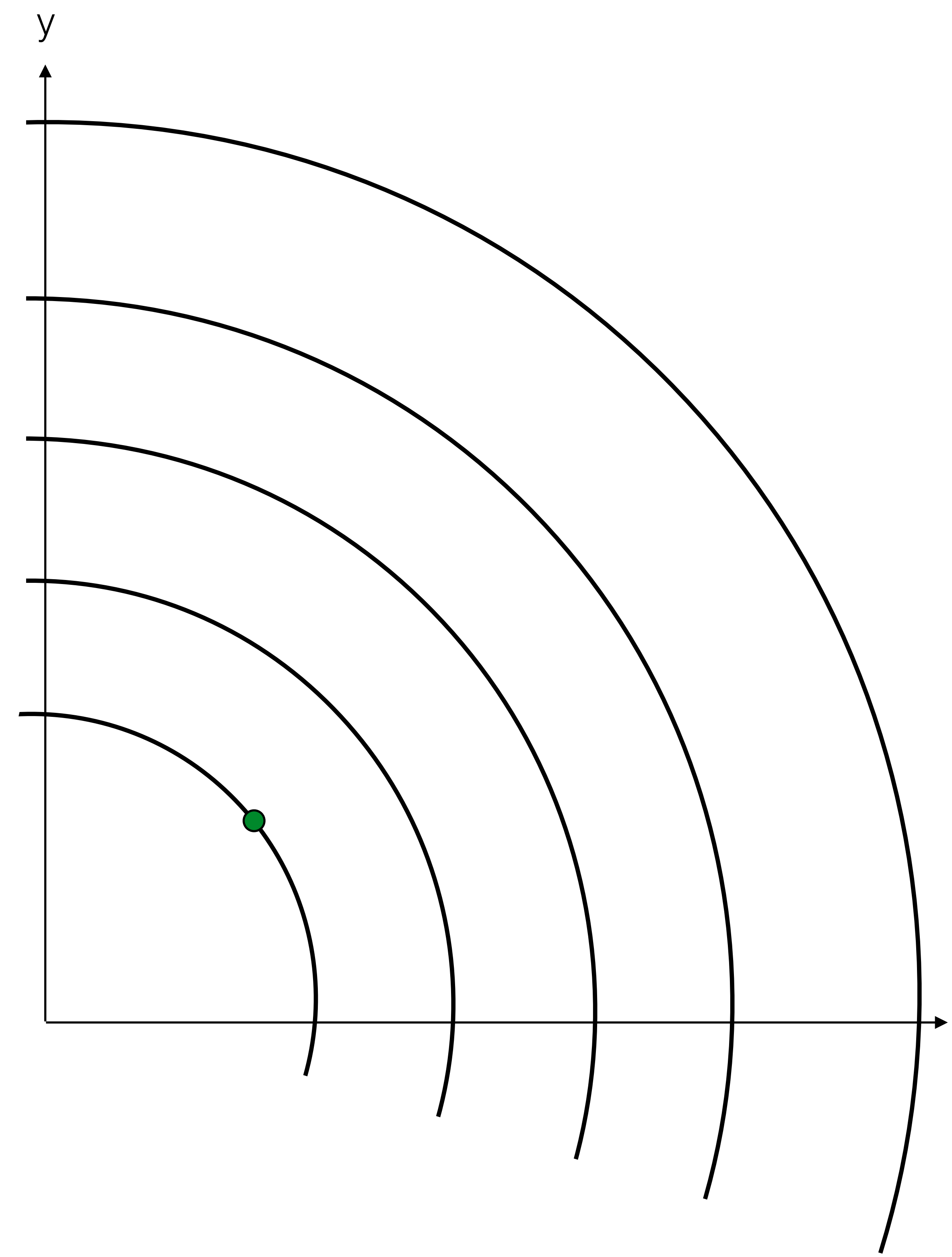
Conformal mapping techniques

Hough transform

- transform your track hits in the x, y space

$$\mathbf{q} = (\cancel{d_0}, \cancel{z_0}, \phi, \cancel{\theta}, q/p_{\mathrm{T}})$$



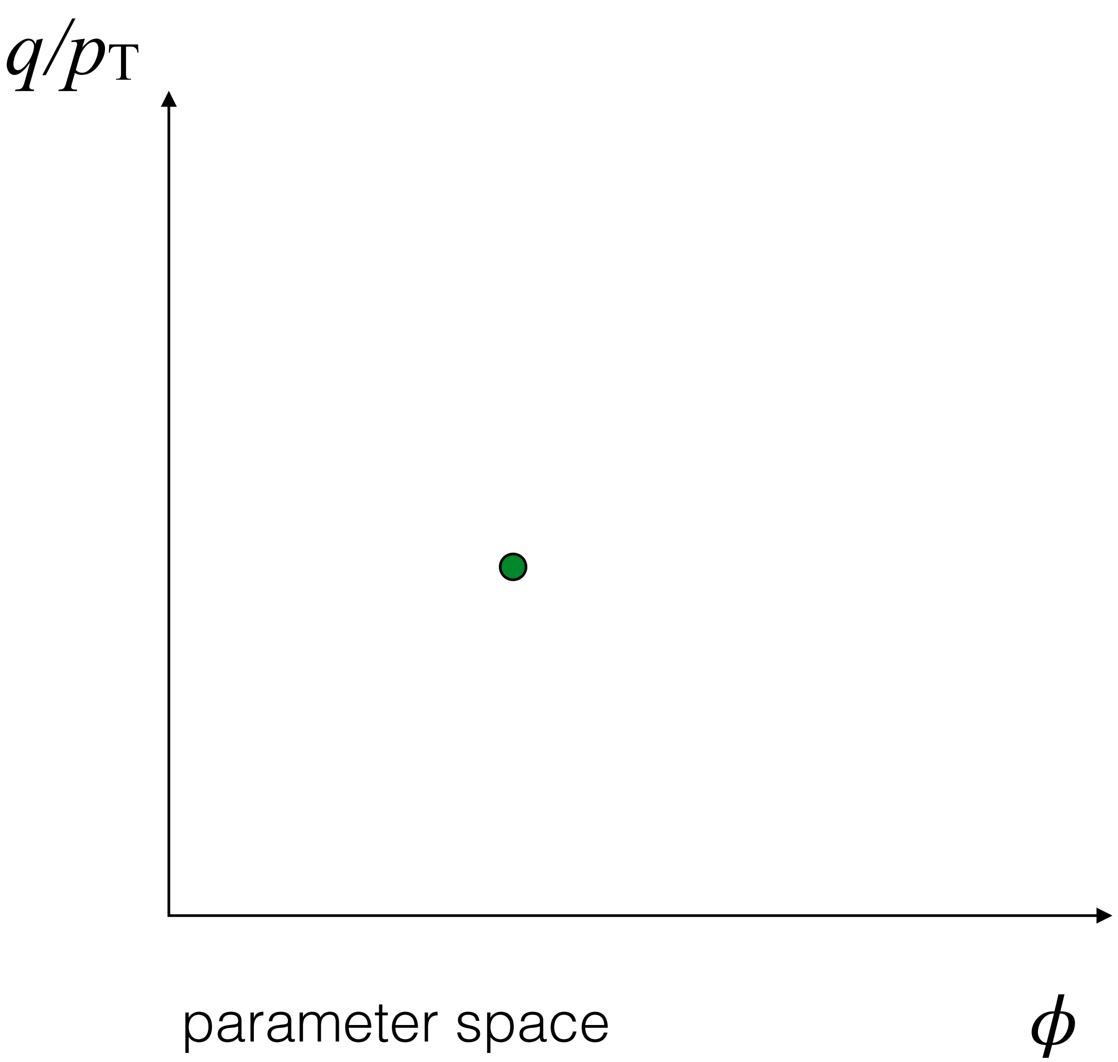


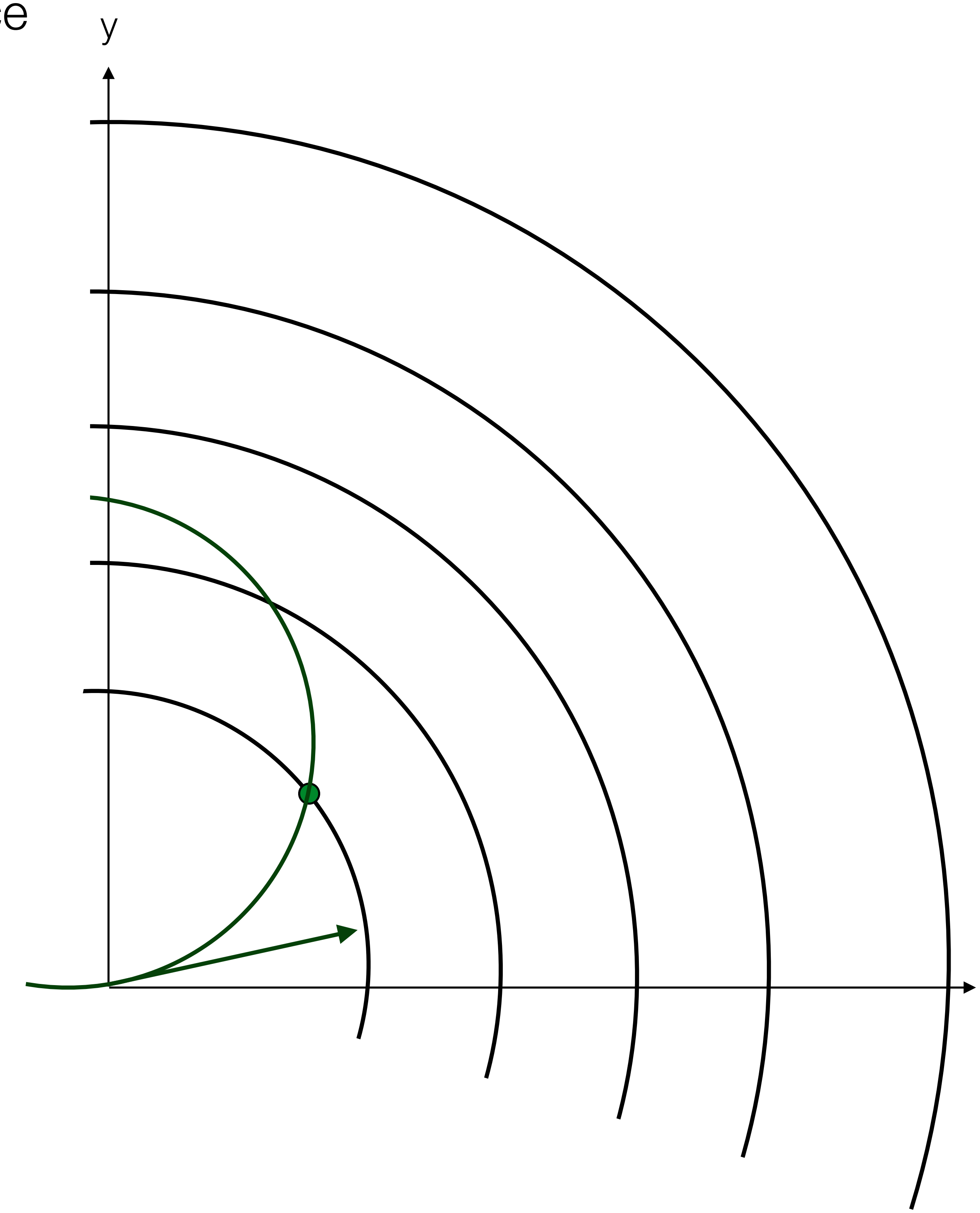
Conformal mapping techniques

Hough transform

- transform your track hits in the x, y space

$$\mathbf{q} = (d_0, z_0, \phi, \theta, q/p_T)$$



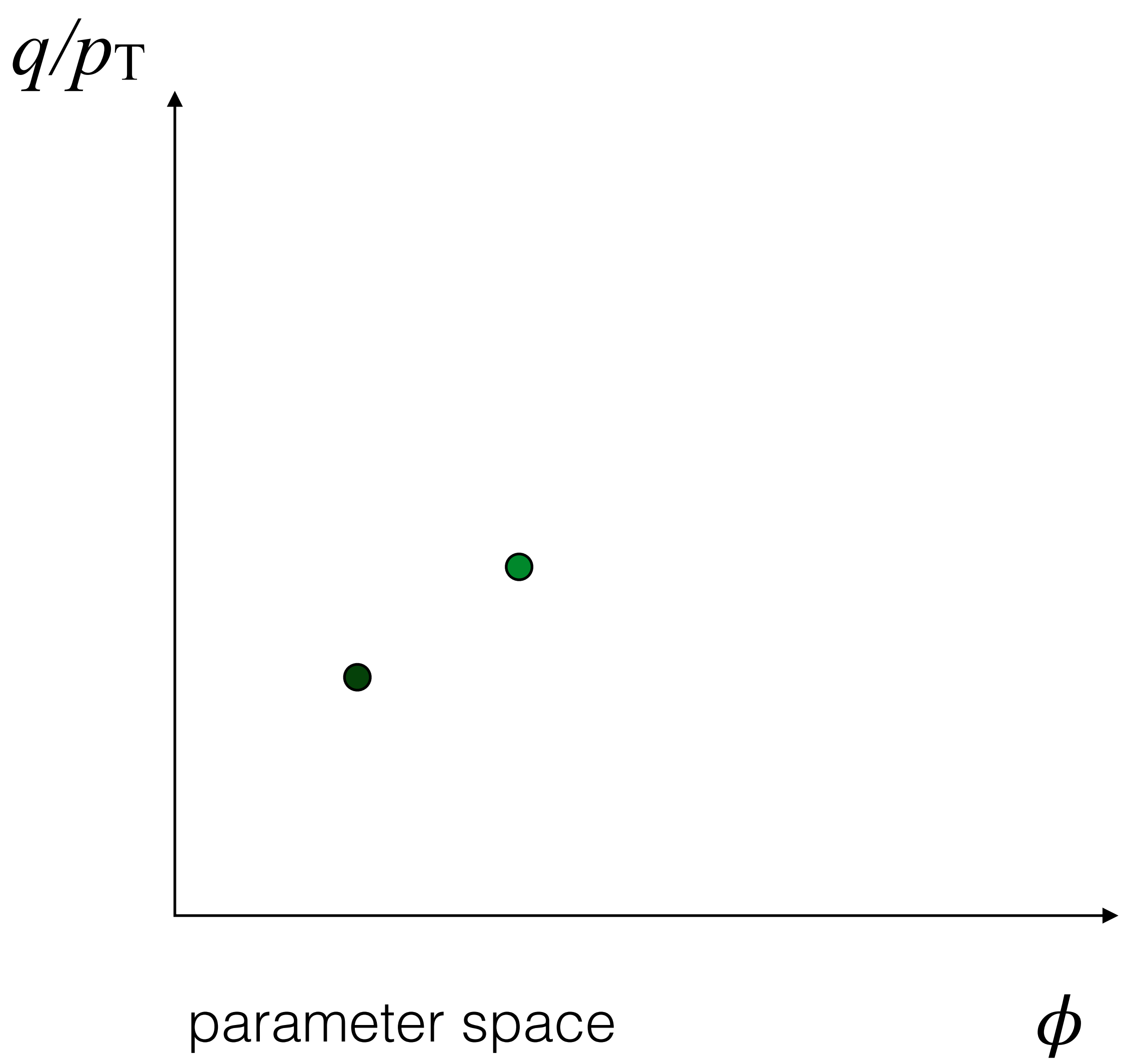


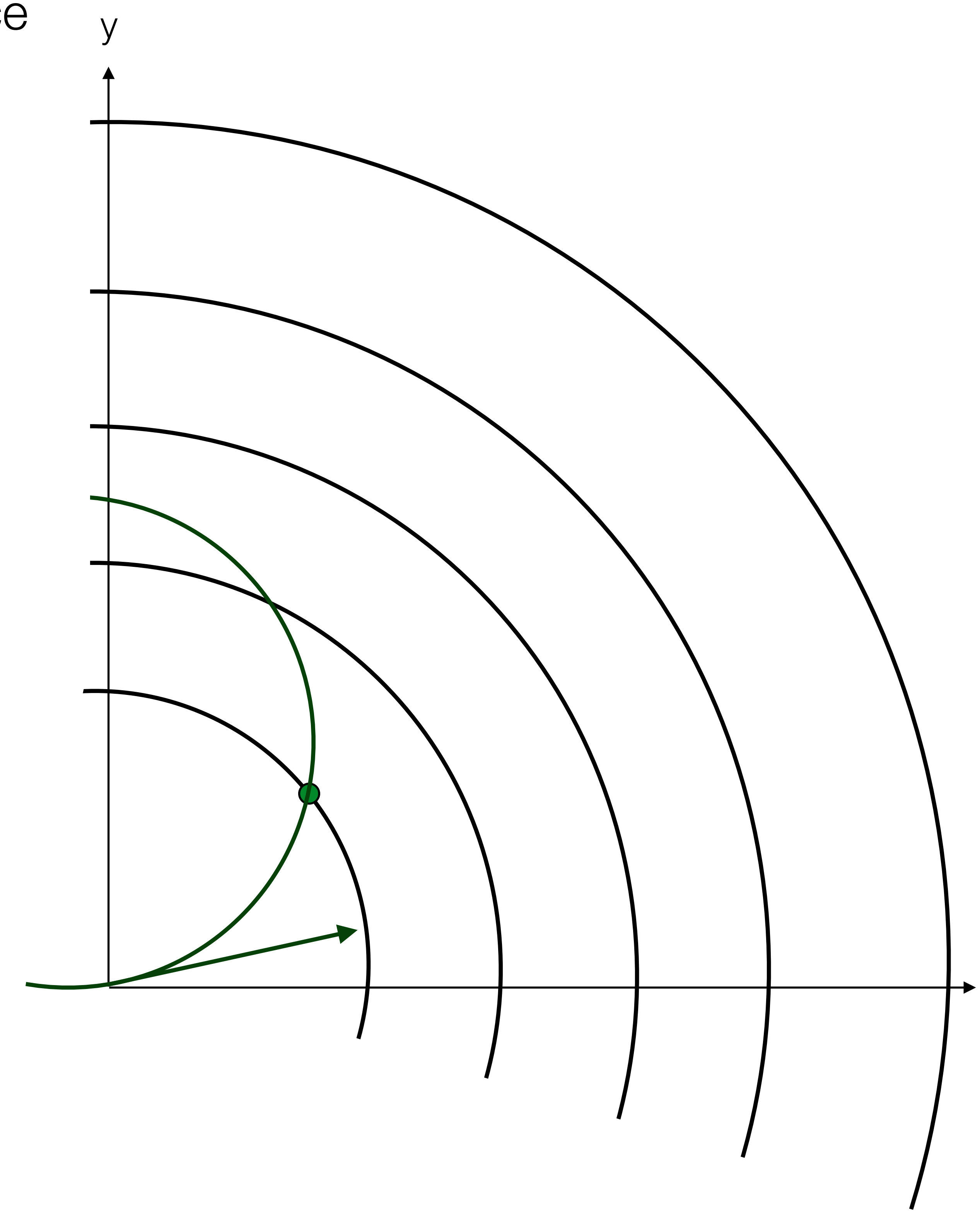
Conformal mapping techniques

Hough transform

- transform your track hits in the x, y space

$$\mathbf{q} = (\cancel{d_0}, \cancel{z_0}, \phi, \cancel{\theta}, q/p_{\mathrm{T}})$$



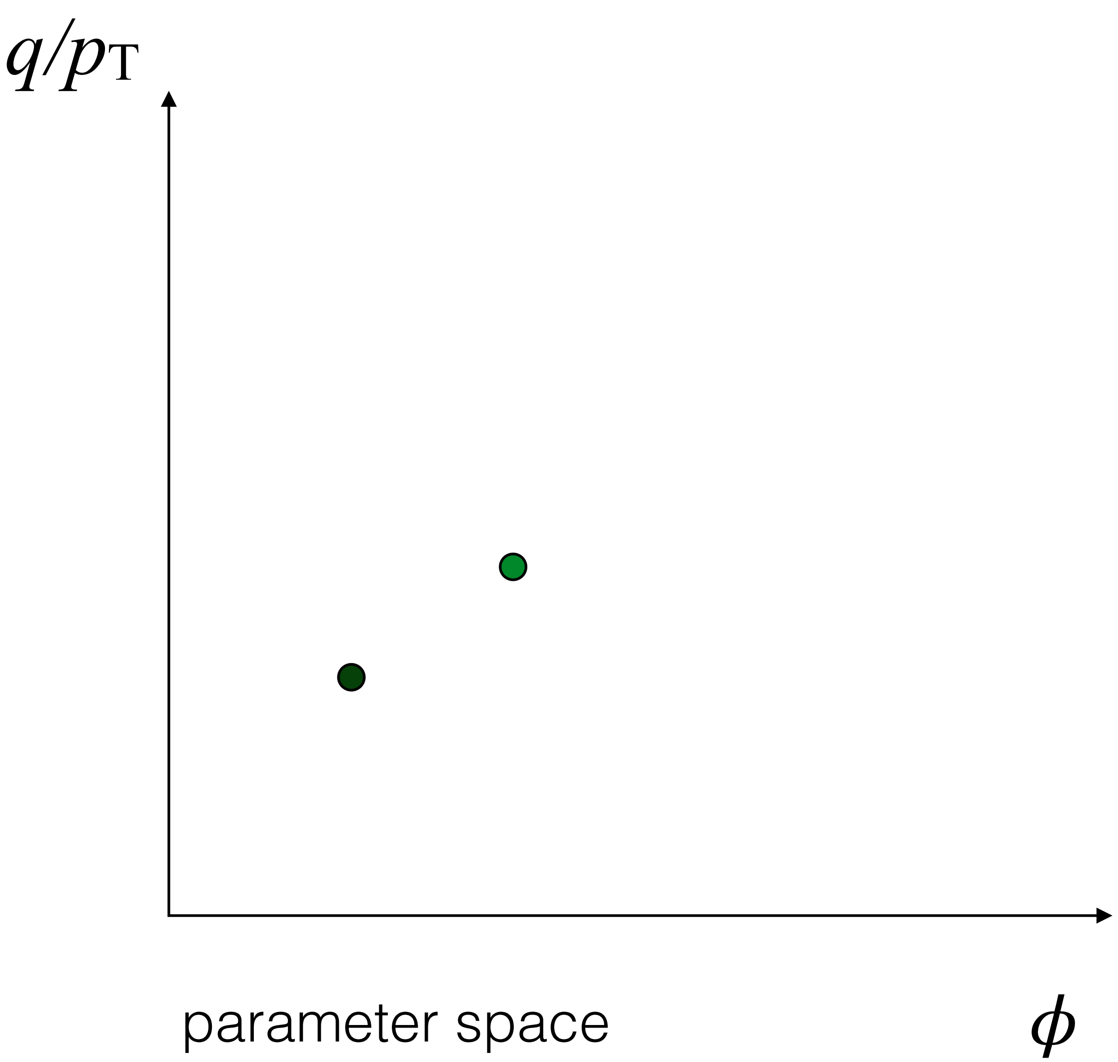


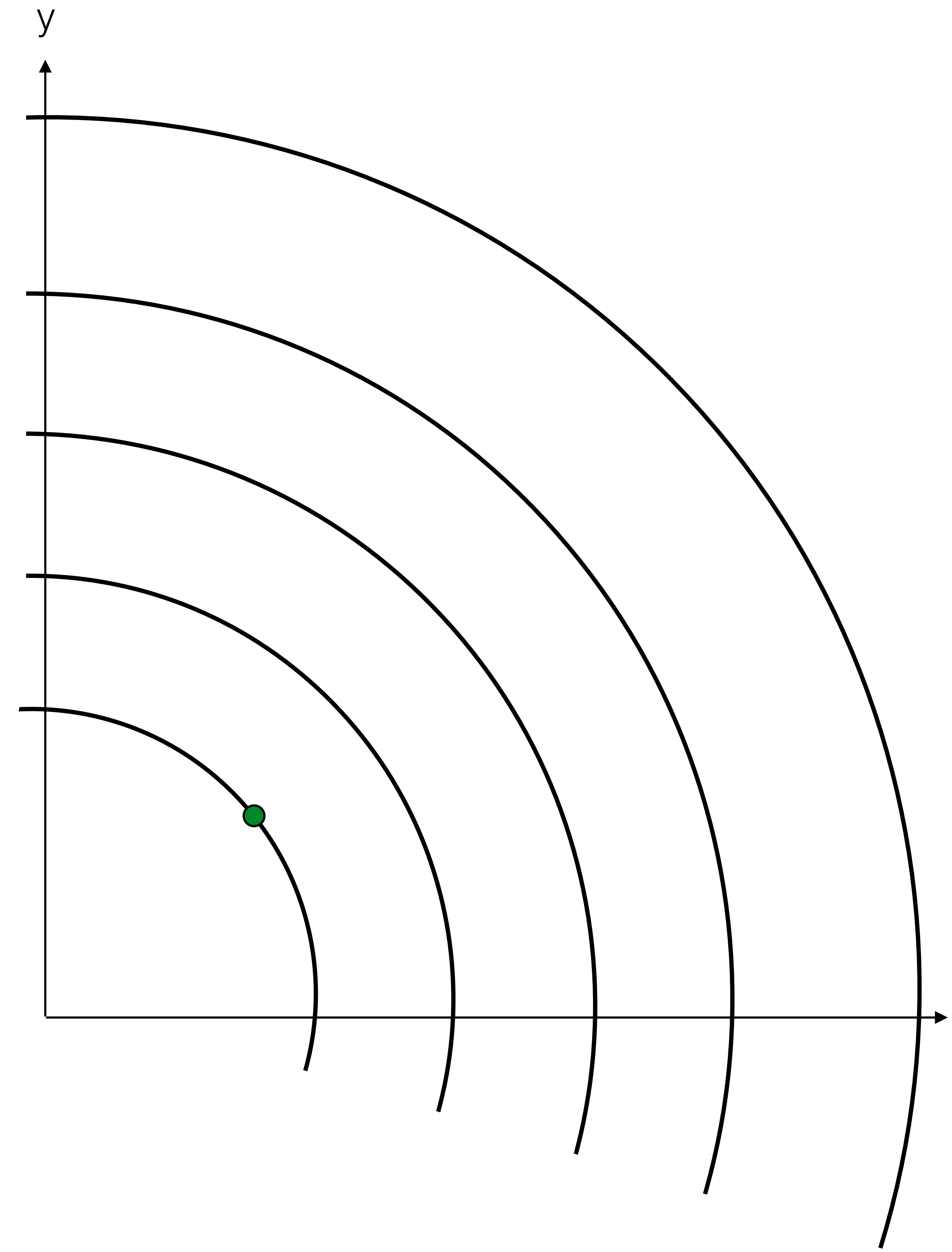
Conformal mapping techniques

<u>Hough transform</u>

- transform your track hits in the x, y space

$$\mathbf{q} = (\cancel{d_0}, \cancel{z_0}, \phi, \cancel{\theta}, q/p_{\mathrm{T}})$$



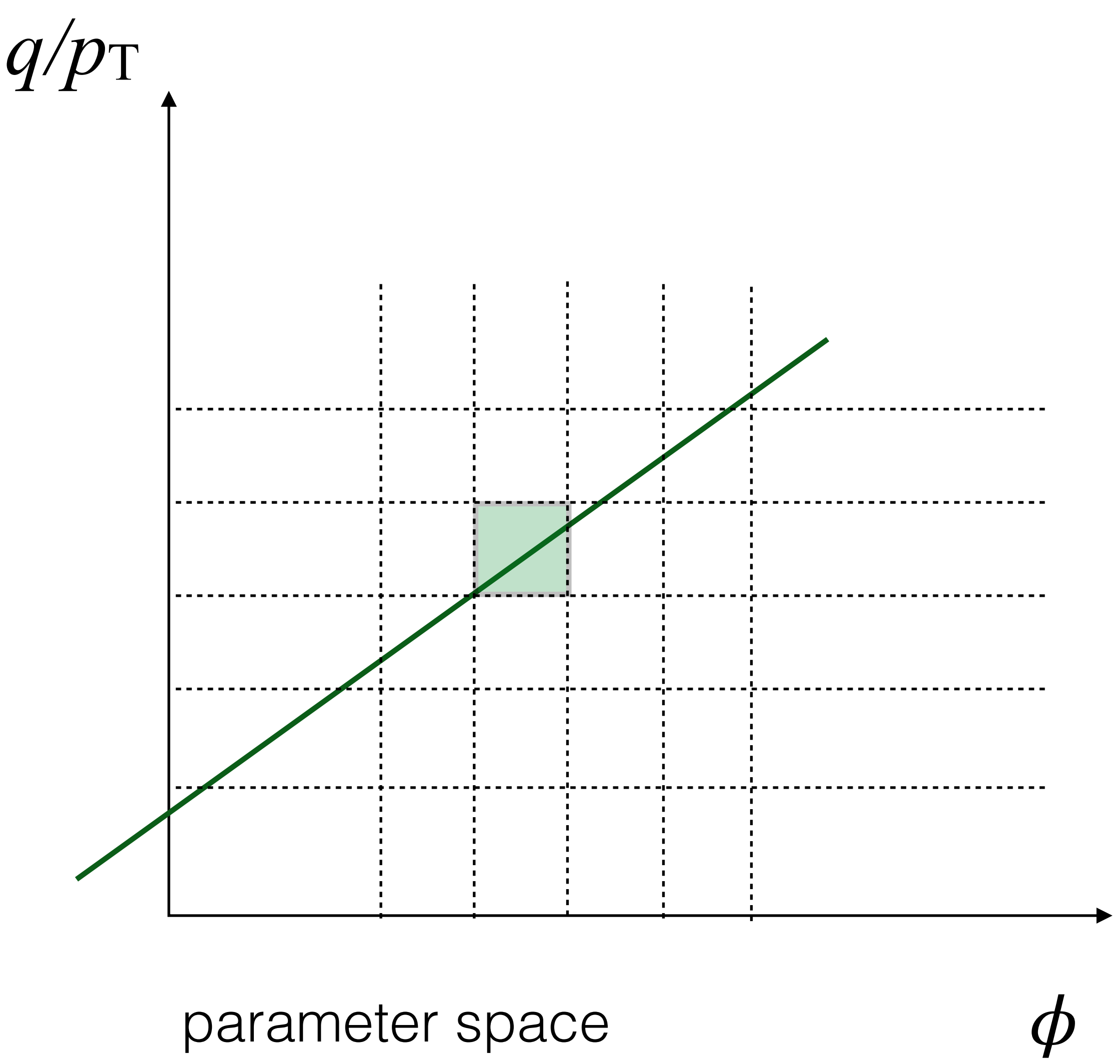


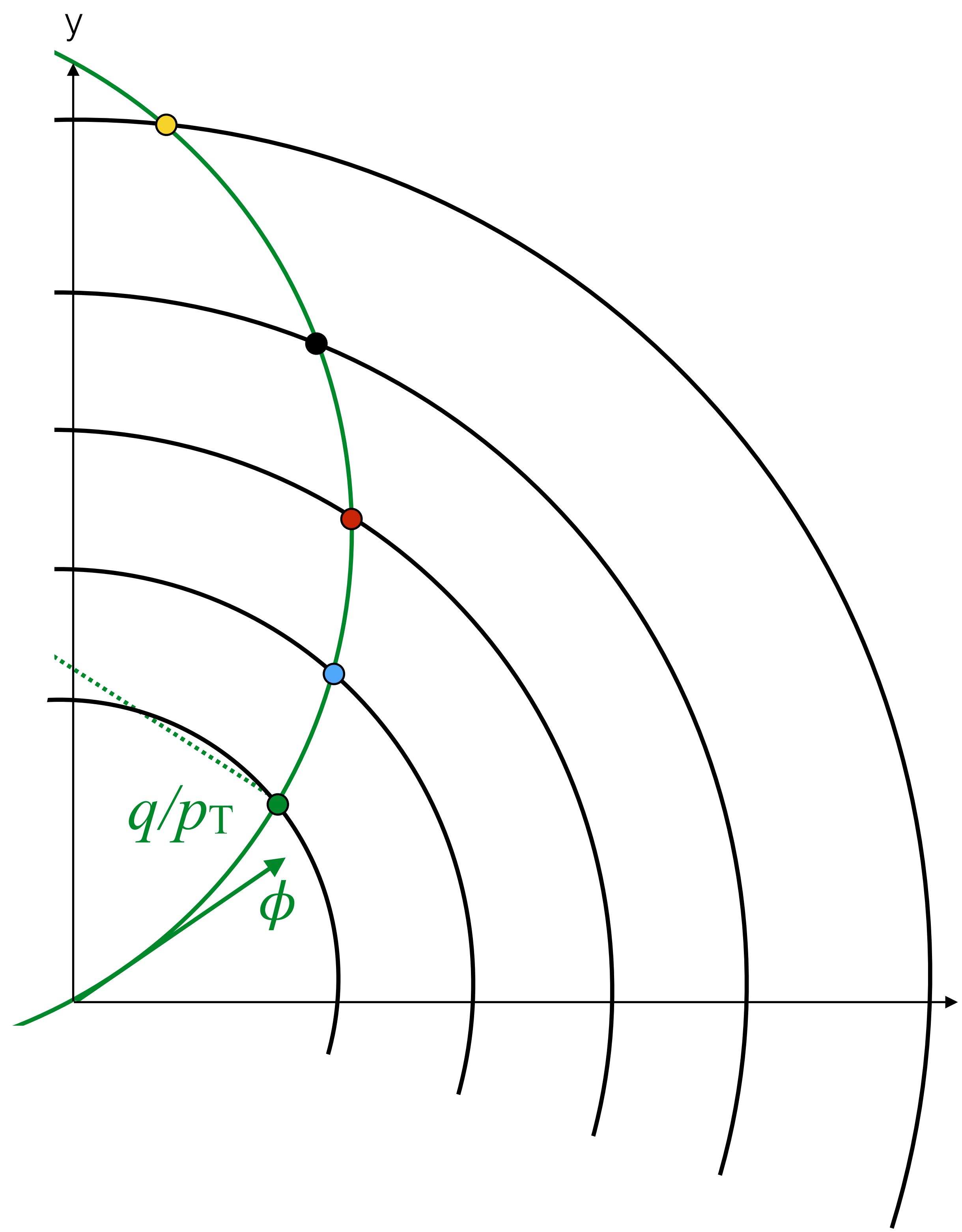
Conformal mapping techniques

Hough transform

- transform your track hits in the x, y space

$$\mathbf{q} = (\cancel{d_0}, \cancel{z_0}, \phi, \cancel{\theta}, q/p_{\mathrm{T}})$$



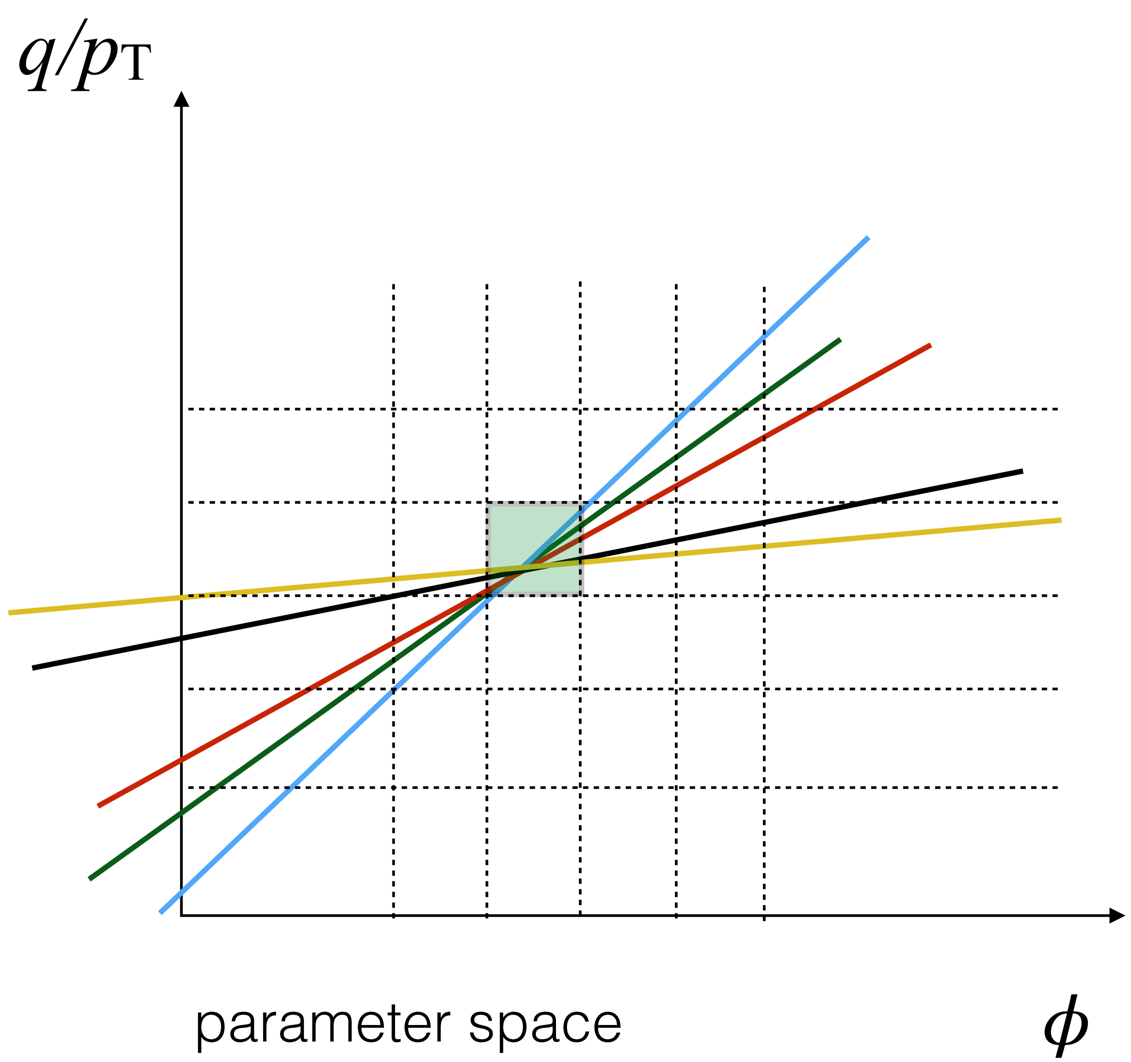


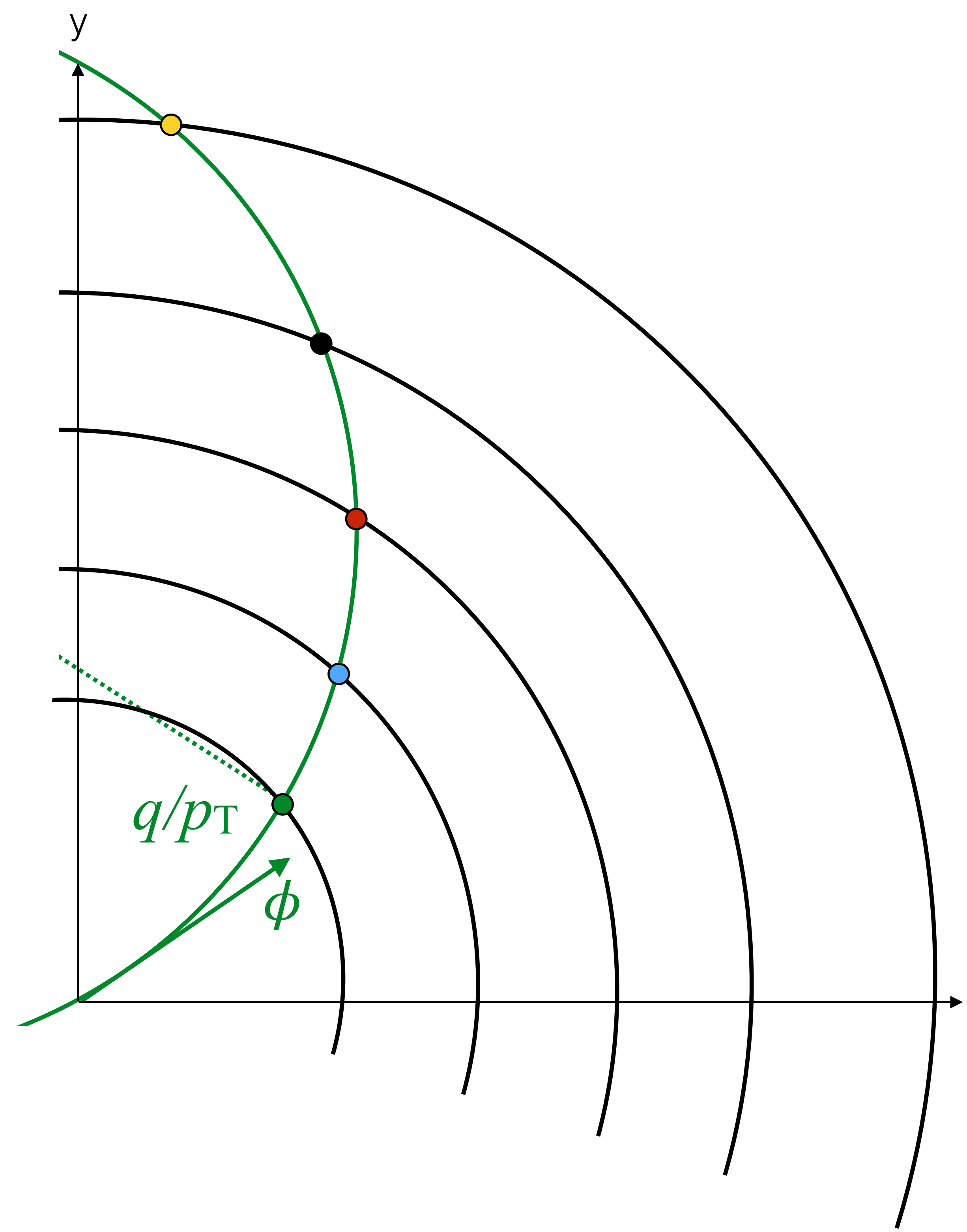
Conformal mapping techniques

Hough transform

- transform your track hits in the x, y space

$$\mathbf{q} = (\cancel{d_0}, \cancel{z_0}, \phi, \cancel{\theta}, q/p_{\mathrm{T}})$$

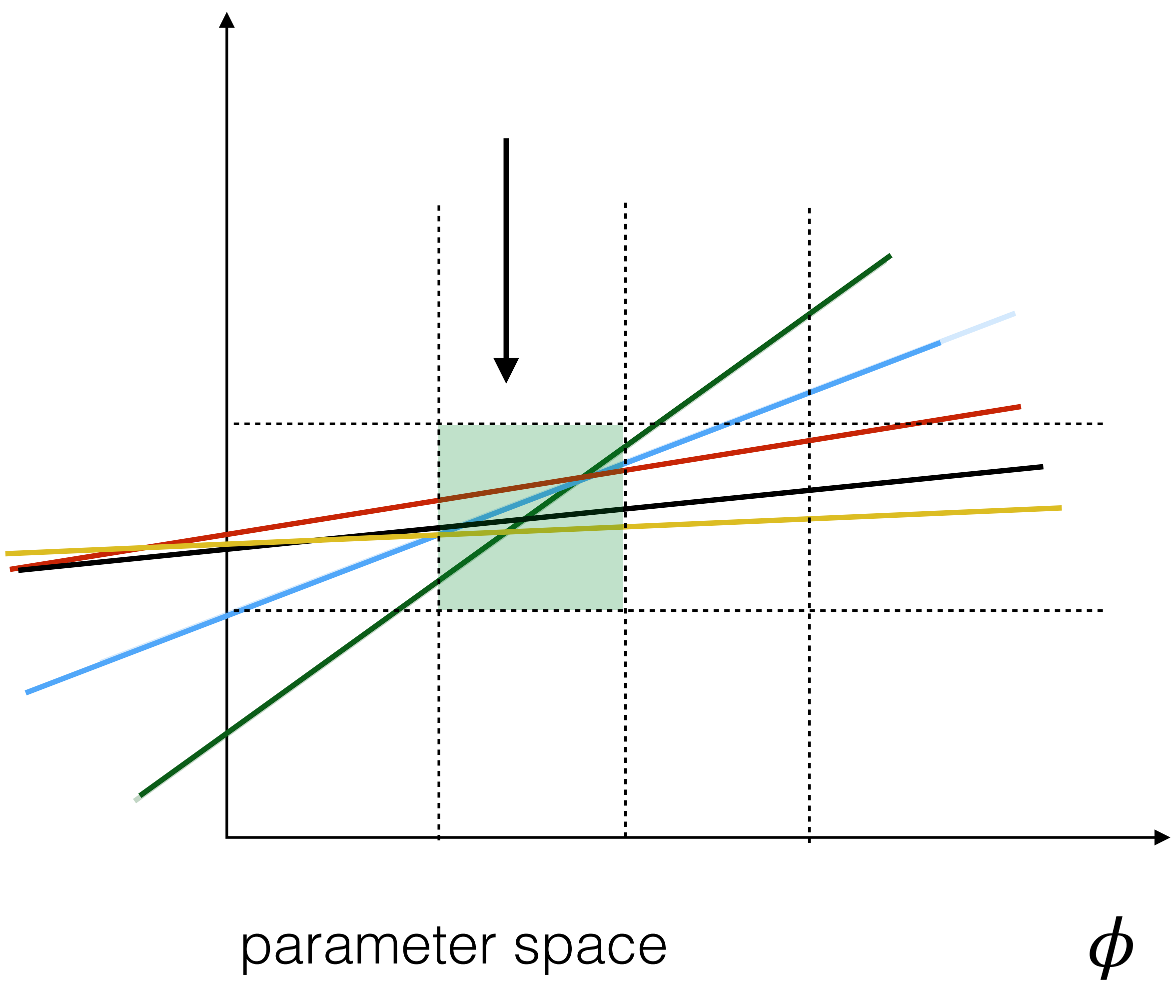


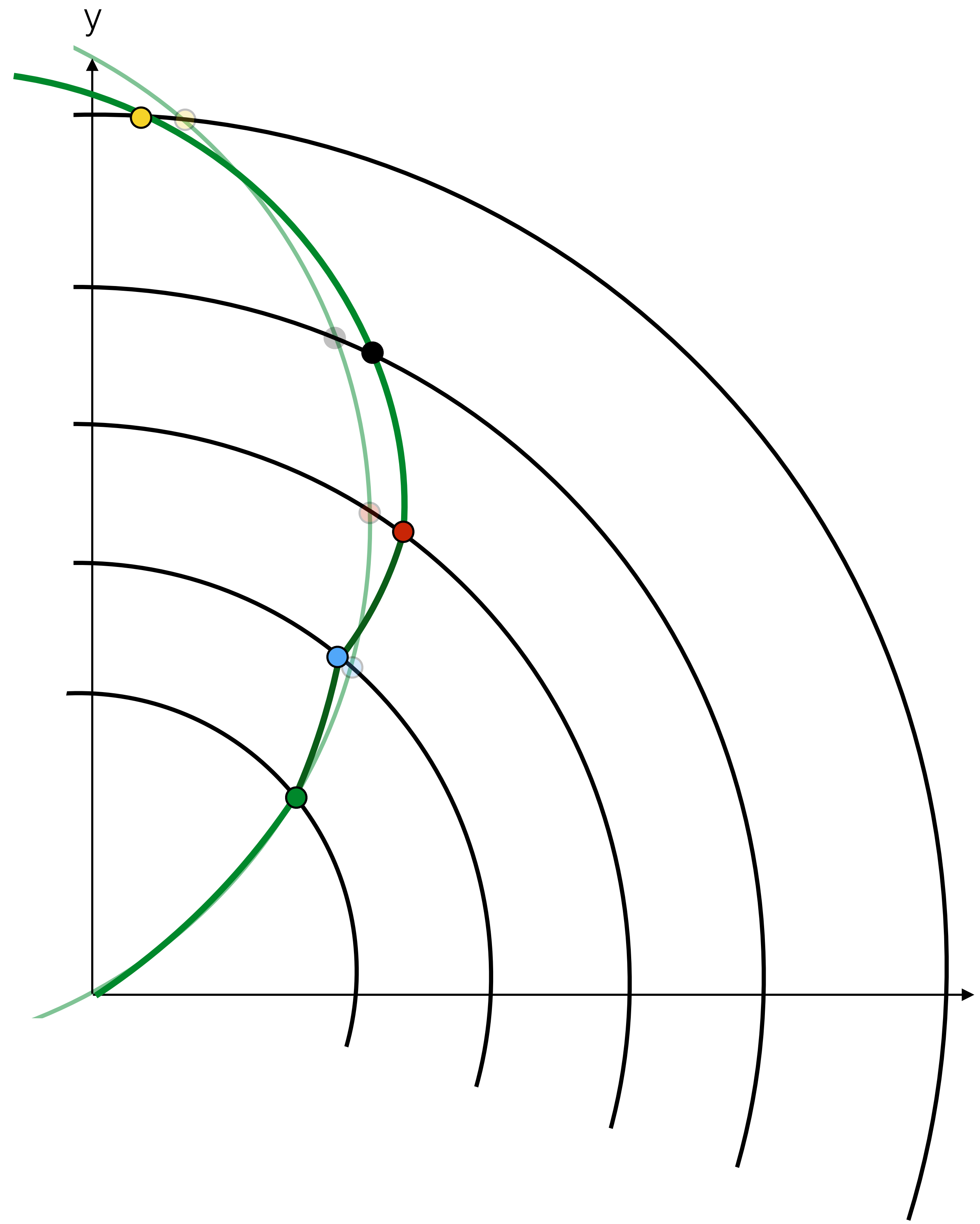


Conformal mapping techniques Problems

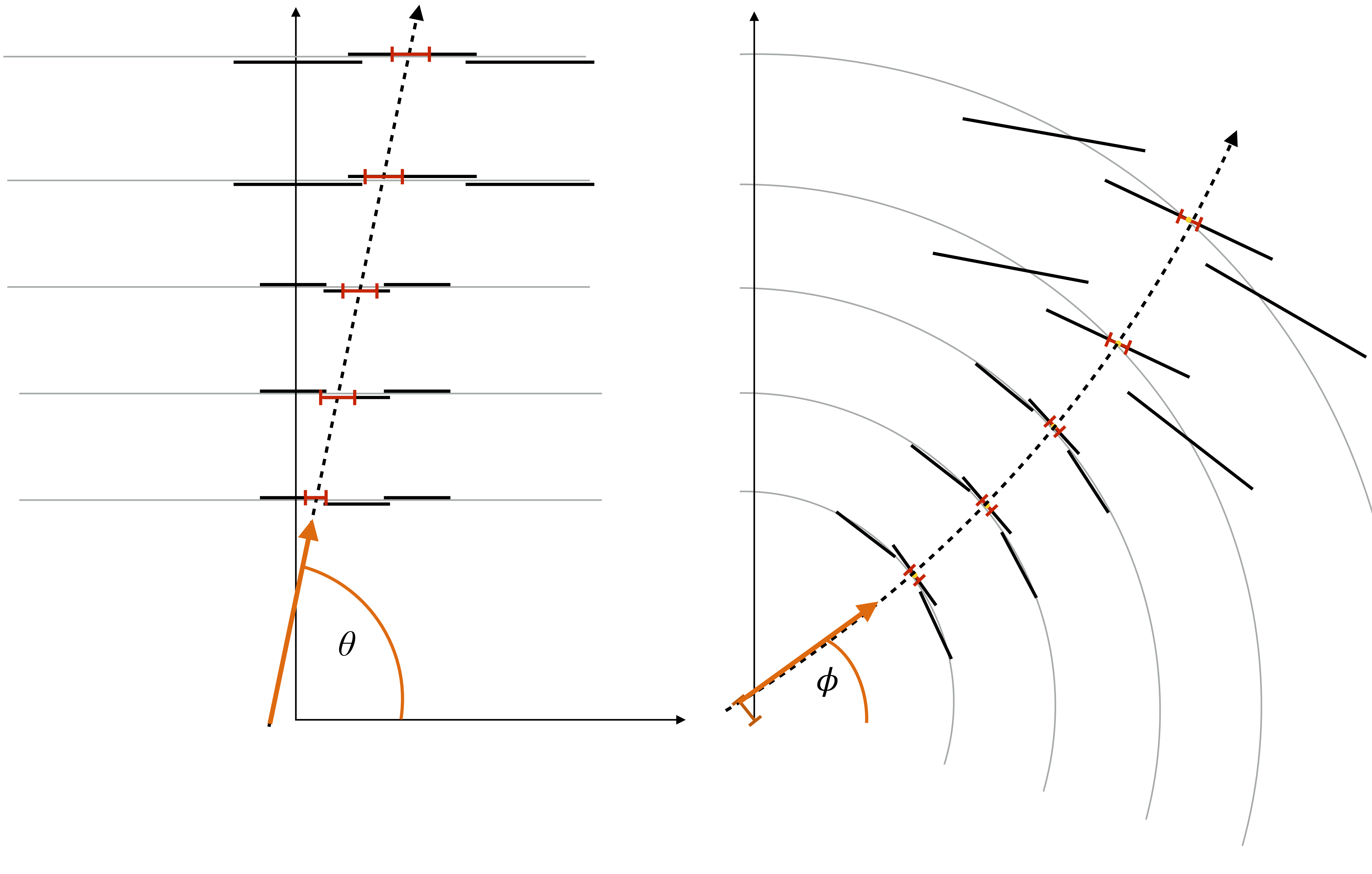
Granularity of grouping in Hough space needs to be adapted to process noise

- scattering effects are very harmful for the precision

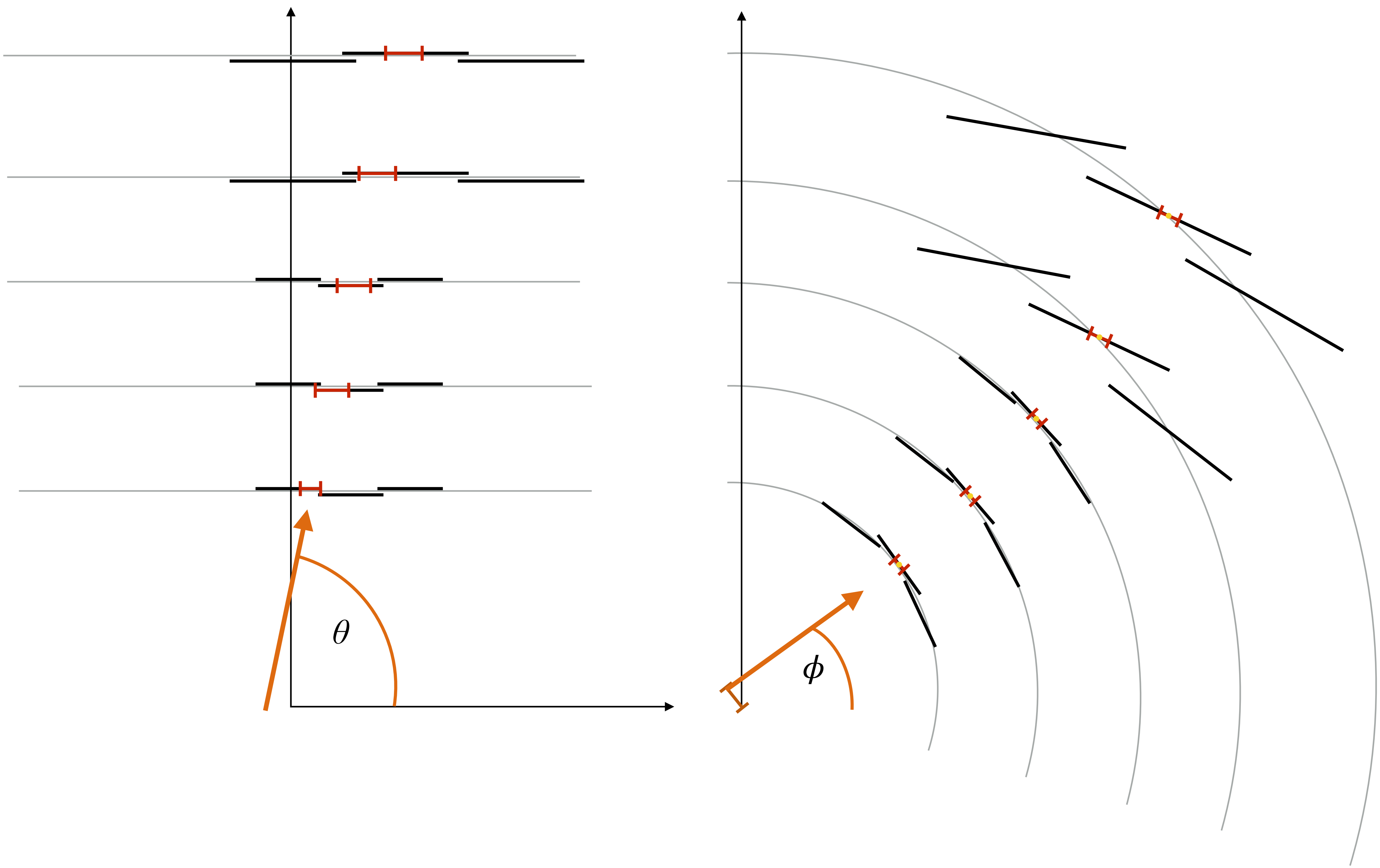




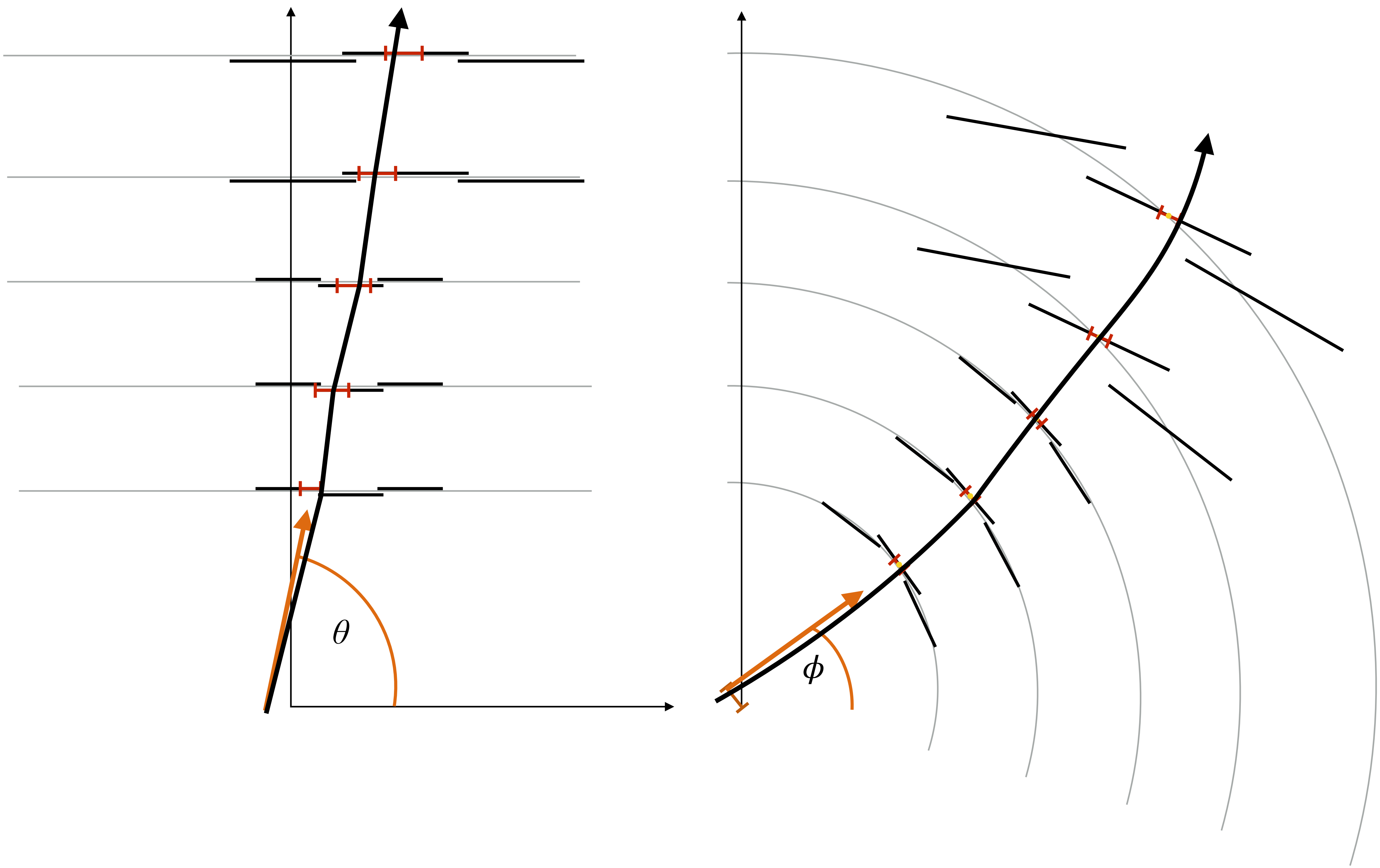
2D world A particle through a toy detector



2D world A particle through a toy detector



2D world A particle through a toy detector



Track fitting Least squares minimisation

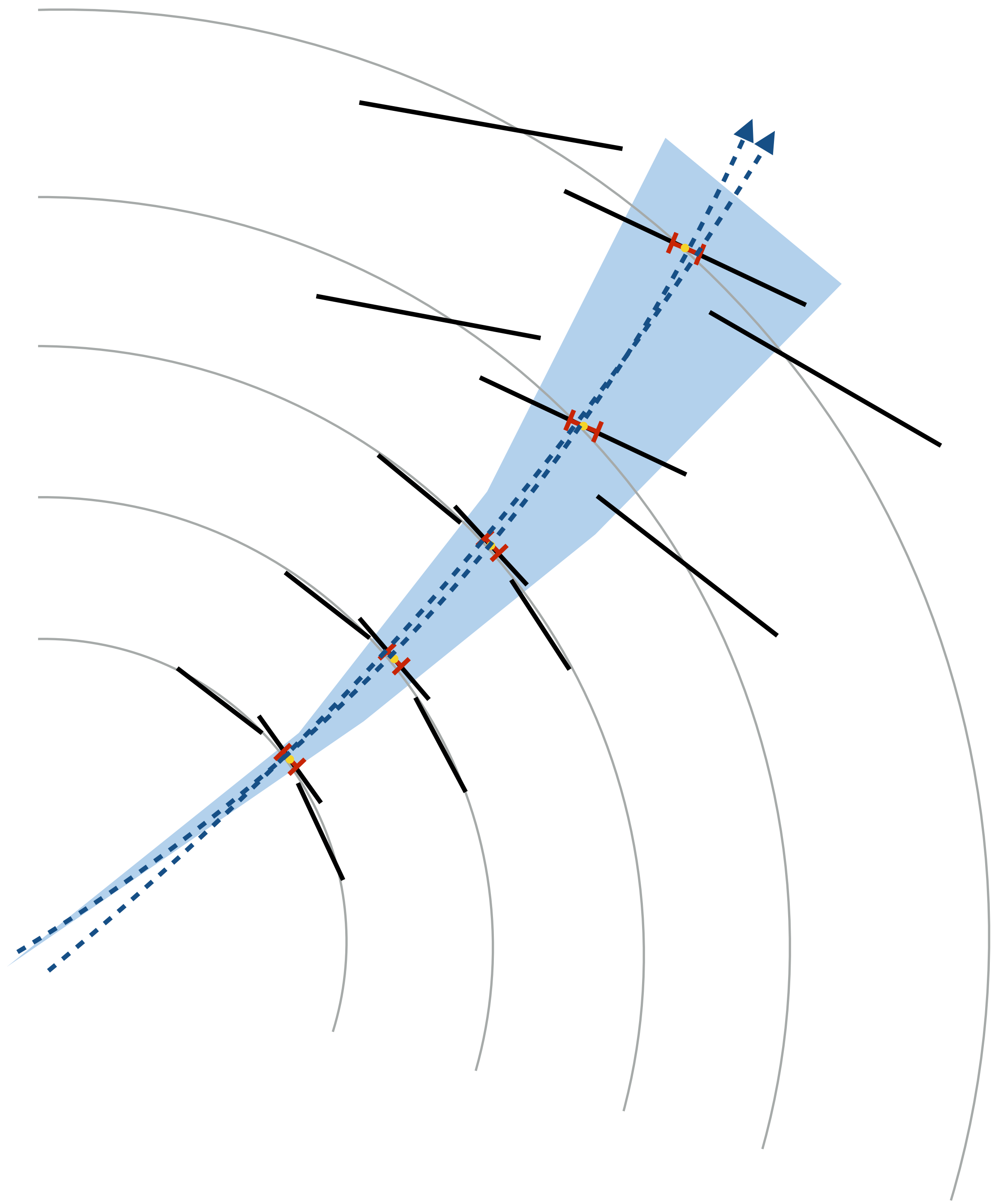
a classical least squares estimator problem !

$$\chi^2 = \sum_k \Delta m_k^T G_K^{-1} \Delta m_k$$ with $$\Delta m_k = m_k - d_k(\mathbf{q})$$ and $\boldsymbol{G_k}$ the covariance of measurement $\mathbf{m}_k$

d_k including transport of $\mathbf{q}$ to measurement layer k

and measurement mapping function

$$\boldsymbol{d}_k = \boldsymbol{h}_k \circ \boldsymbol{f}_{k|k-1} \circ \cdots \circ \boldsymbol{f}_{2|1} \circ \boldsymbol{f}_{1|0}$$



Track fitting Least squares minimisation

a classical least squares estimator problem !

$$\chi^2 = \sum_k \Delta m_k^T G_K^{-1} \Delta m_k$$ with $$\Delta m_k = m_k - d_k(\mathbf{q})$$ and $\boldsymbol{G_k}$ the covariance of measurement $\mathbf{m}_k$

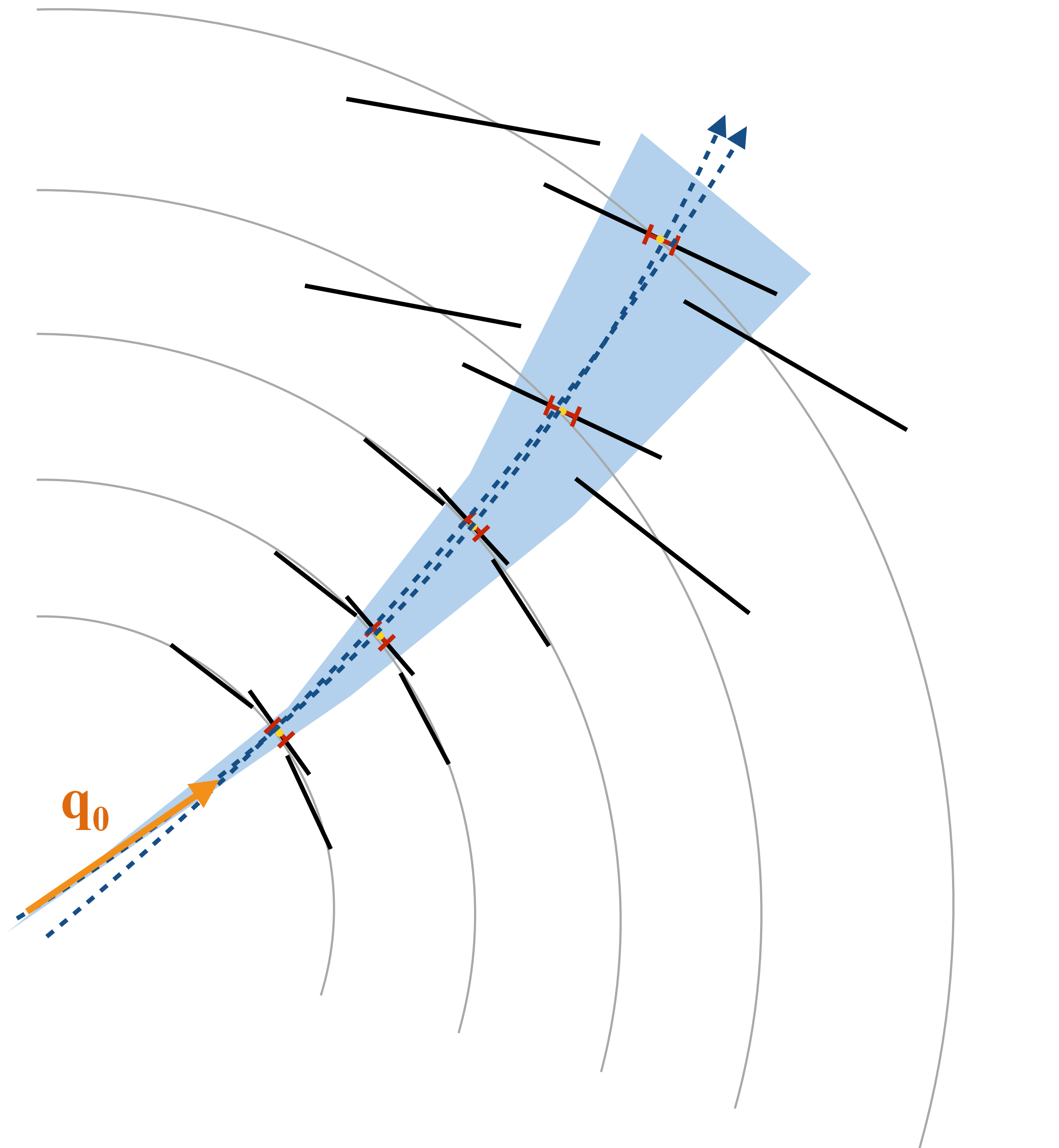
d_k including transport of $\mathbf{q}$ to measurement layer k and measurement mapping function

$$\boldsymbol{d}_k = \boldsymbol{h}_k \circ \boldsymbol{f}_{k|k-1} \circ \cdots \circ \boldsymbol{f}_{2|1} \circ \boldsymbol{f}_{1|0}$$

linearise the problem, starting from an initial state $\mathbf{q_0}$

$$d_k(\mathbf{q_0} + \delta\mathbf{q}) \cong d_k(\mathbf{q_0}) + D_k \cdot \delta\mathbf{q}$$

with Jacobian $$\boldsymbol{D}_k = \boldsymbol{H}_k \boldsymbol{F}_{k|k-1} \cdots \boldsymbol{F}_{2|1} \boldsymbol{F}_{1|0}$$



Track fitting Least squares minimisation

a classical least squares estimator problem !

$$\chi^2 = \sum_k \Delta m_k^T G_K^{-1} \Delta m_k$$ with $$\Delta m_k = m_k - d_k(\mathbf{q})$$ and $\boldsymbol{G_k}$ the covariance of measurement $\mathbf{m}_k$

d_k including transport of $\mathbf{q}$ to measurement layer k and measurement mapping function

$$\boldsymbol{d}_k = \boldsymbol{h}_k \circ \boldsymbol{f}_{k|k-1} \circ \cdots \circ \boldsymbol{f}_{2|1} \circ \boldsymbol{f}_{1|0}$$

linearise the problem, starting from an initial state $\mathbf{q_0}$

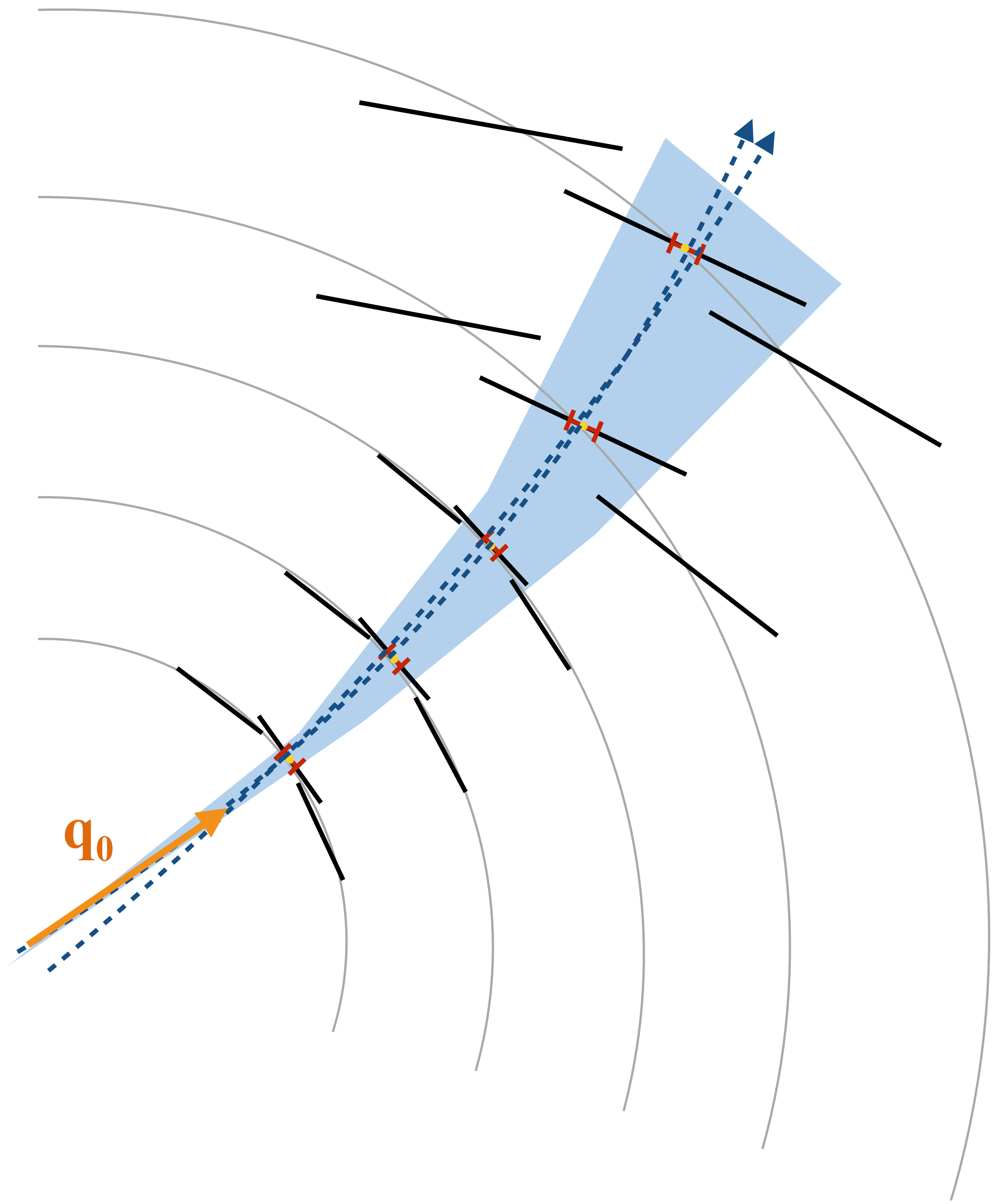
$$d_k(\mathbf{q_0} + \delta\mathbf{q}) \cong d_k(\mathbf{q_0}) + D_k \cdot \delta\mathbf{q}$$

with Jacobian $$\boldsymbol{D}_k = \boldsymbol{H}_k \boldsymbol{F}_{k|k-1} \cdots \boldsymbol{F}_{2|1} \boldsymbol{F}_{1|0}$$

find the global minimum: $$\frac{\partial \chi^2}{\partial \mathbf{q}} \stackrel{!}{=} 0$$

$$\partial\mathbf{q} = \left(\sum_k D_k^T G_k^{-1} D_k\right)^{-1} \sum_k D_k^T G_k^{-1} (m_k - d_k(\mathbf{q_0}))$$

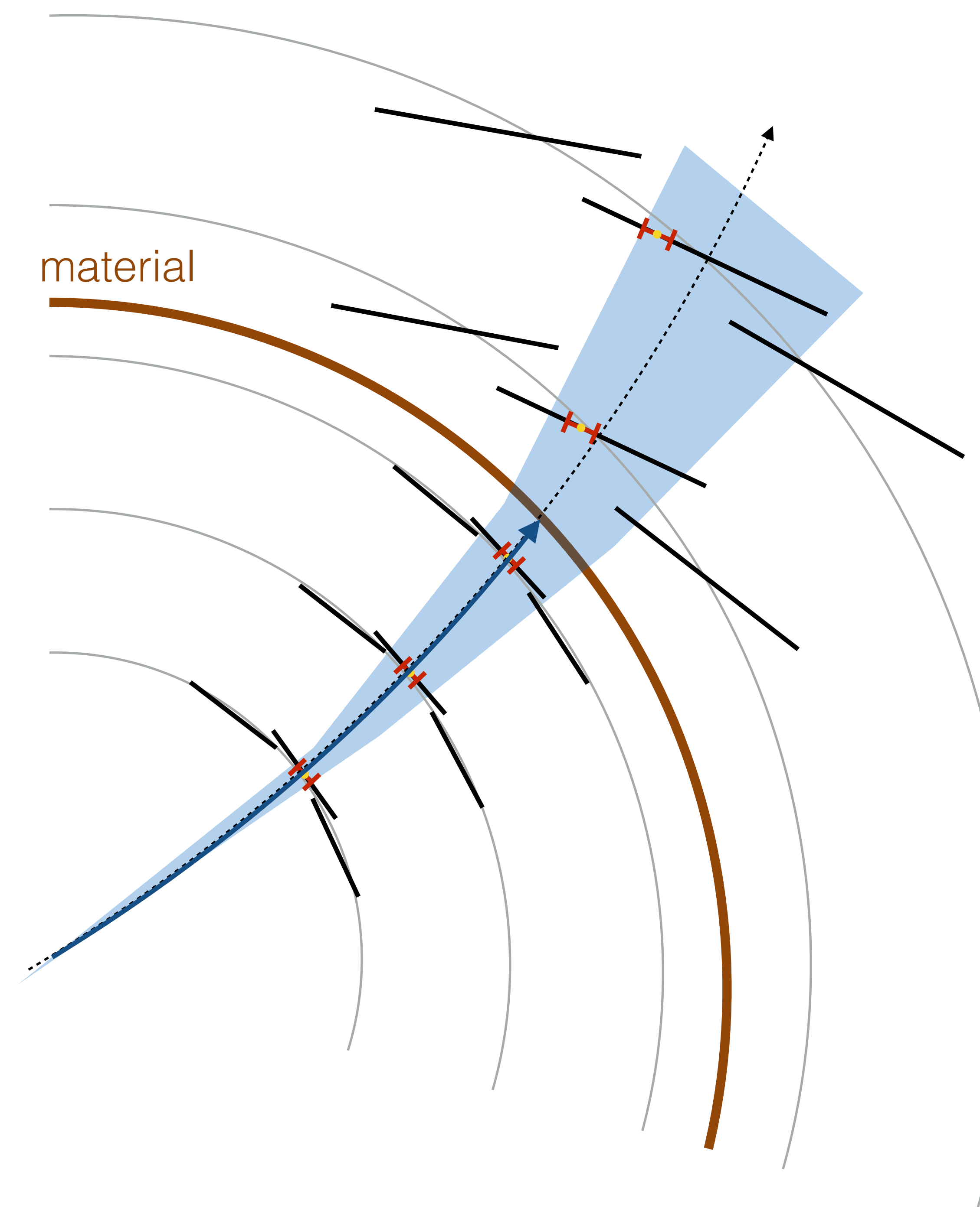
$$C = \left(\sum_k D_k^T G_k^{-1} D_k\right)^{-1}$$



Track fitting Least squares minimisation with material

in reality the particle gets deflected by **material**

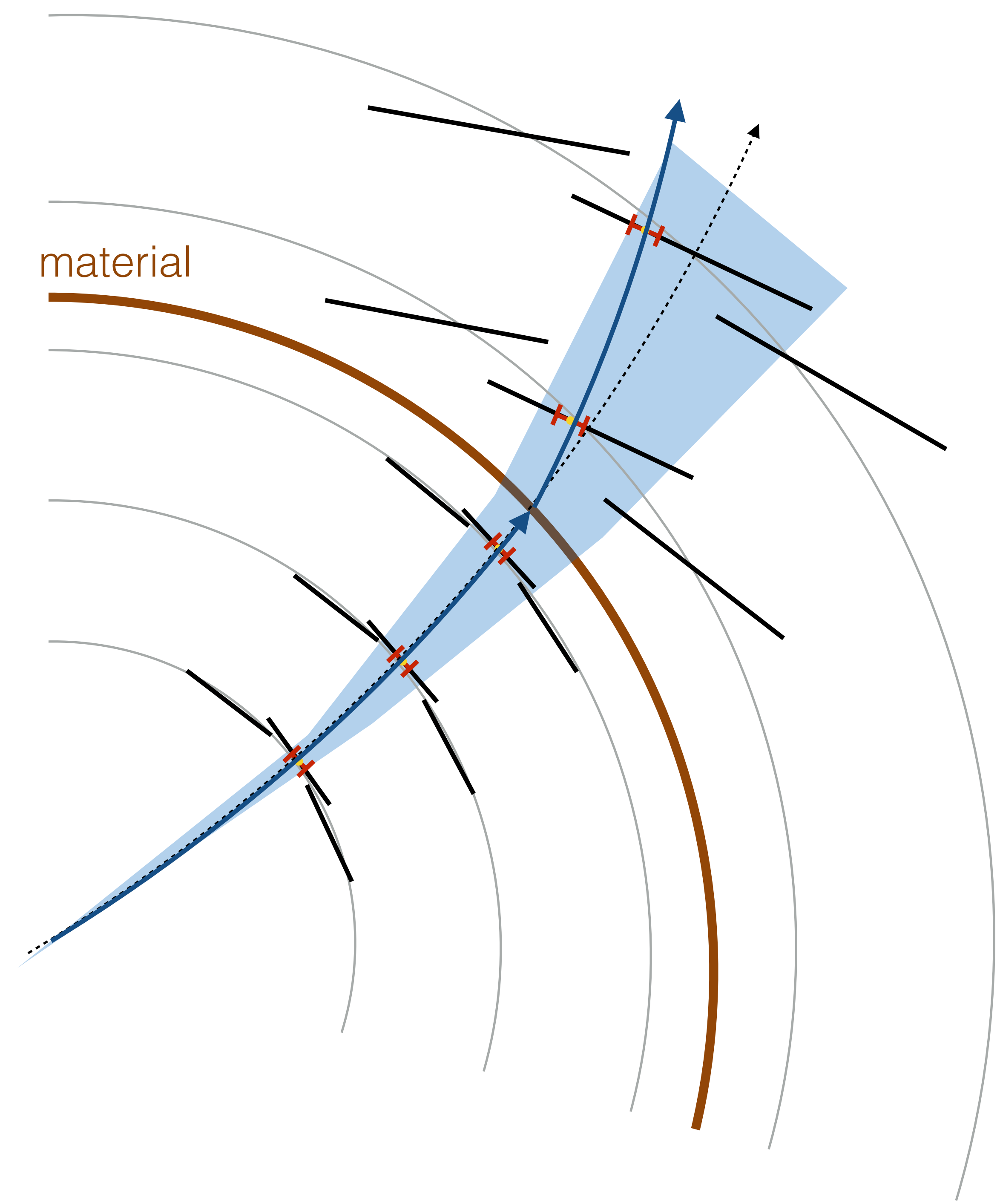
- multiple coulomb scattering



Track fitting Least squares minimisation with material

in reality the particle gets deflected by **material**

- multiple coulomb scattering



Track fitting Least squares minimisation with material

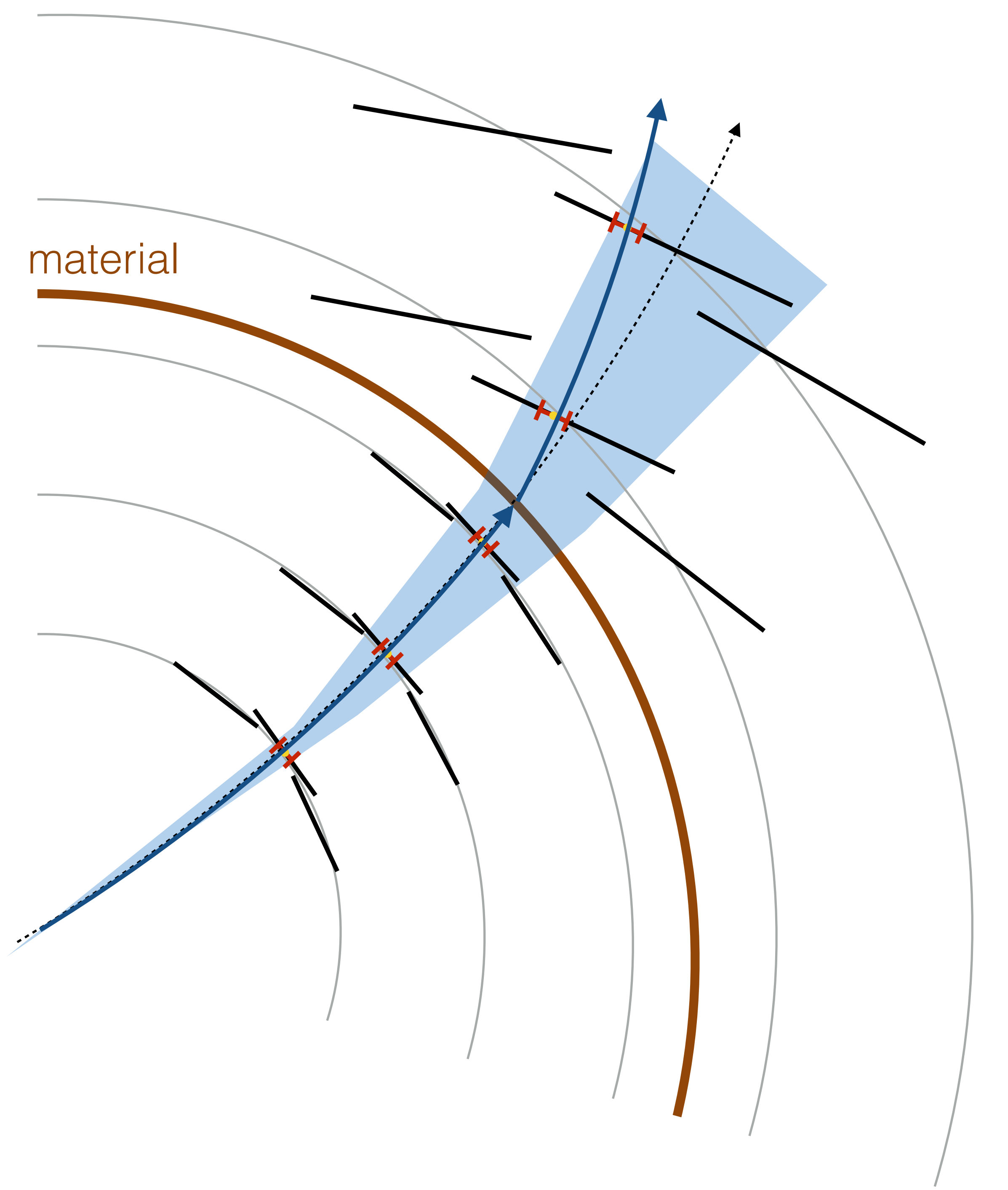
in reality the particle gets deflected by **material**

- multiple coulomb scattering

- modification of the χ^2 function

$$\chi^2 = \sum_k \Delta m_k^T G_K^{-1} \Delta m_k + \sum_i \delta\theta_i^T Q_i^{-1} \delta\theta_i$$

with: $\Delta m_k = m_k - d_k(\mathbf{q}, \delta\theta_i)$



Track fitting Least squares minimisation with material

in reality the particle gets deflected by **material**

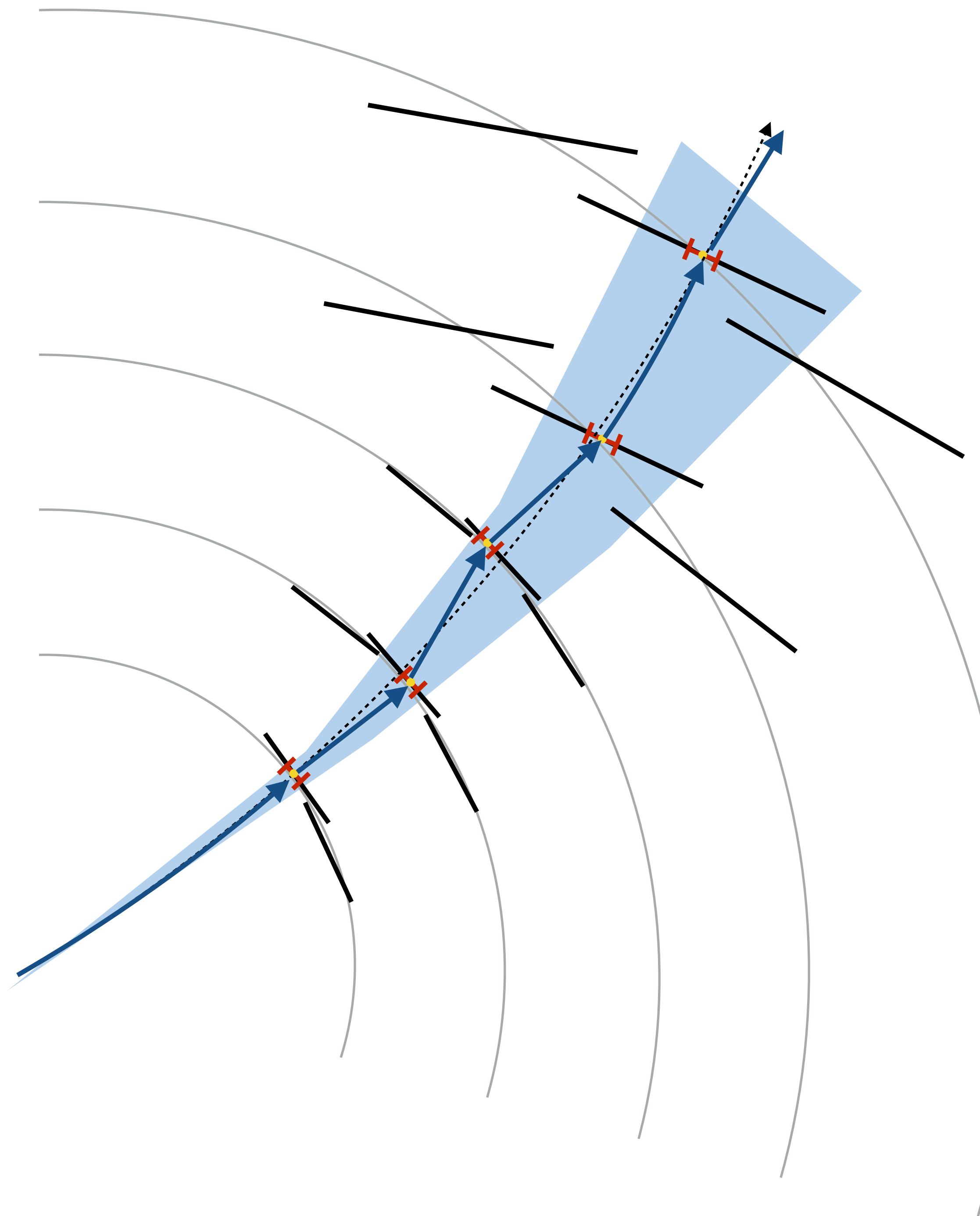
- multiple coulomb scattering

- modification of the χ^2 function

$$\chi^2 = \sum_k \Delta m_k^T G_K^{-1} \Delta m_k + \sum_i \delta\theta_i^T Q_i^{-1} \delta\theta_i$$

with: $$\Delta m_k = m_k - d_k(\mathbf{q}, \delta\theta_i)$$

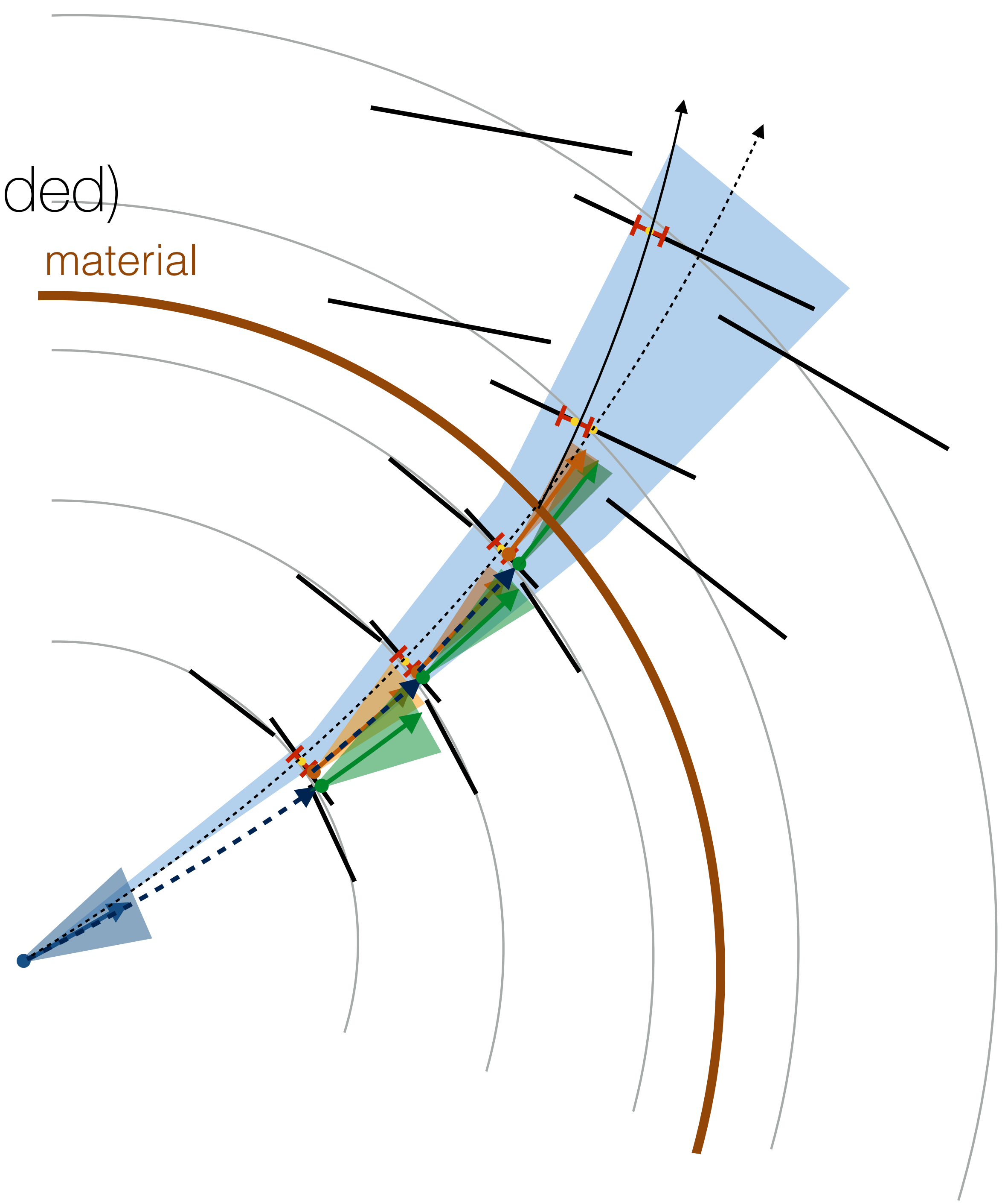
- every layer is a **material layer**
  - creates a computational problem: matrix inversion of huge matrix to find the χ^2 minimum



Track fitting Kalman filter with material

when crossing a material layer

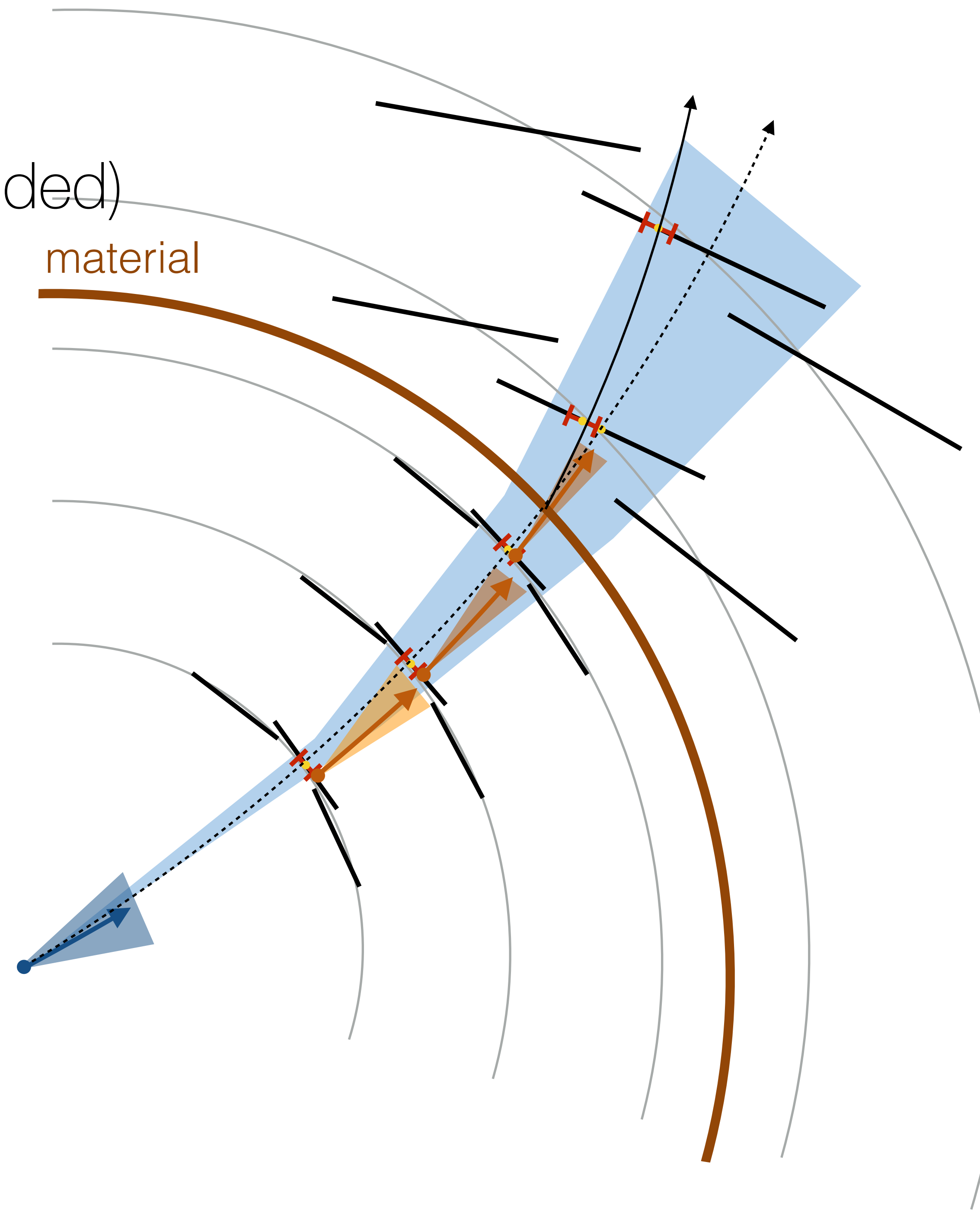
- increase covariance by "noise" term according to the amount of material crossed
  (scattering has expected mean of 0)
- energy loss is applied deterministically
  (additional noise term for straggling added)



Track fitting Kalman filter with material

when crossing a material layer

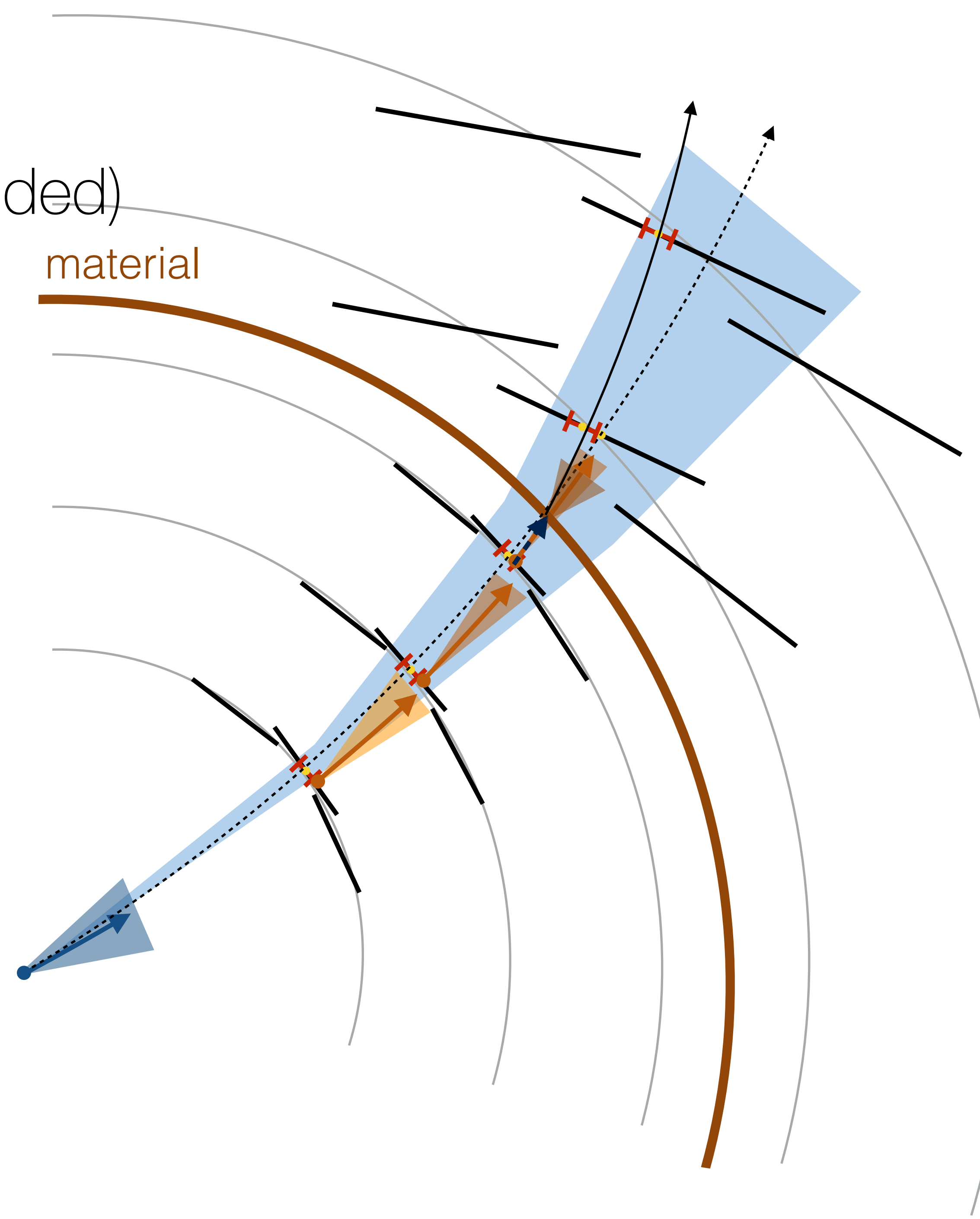
- increase covariance by "noise" term according to the amount of material crossed
  (scattering has expected mean of 0)
- energy loss is applied deterministically
  (additional noise term for straggling added)



Track fitting Kalman filter with material

when crossing a material layer

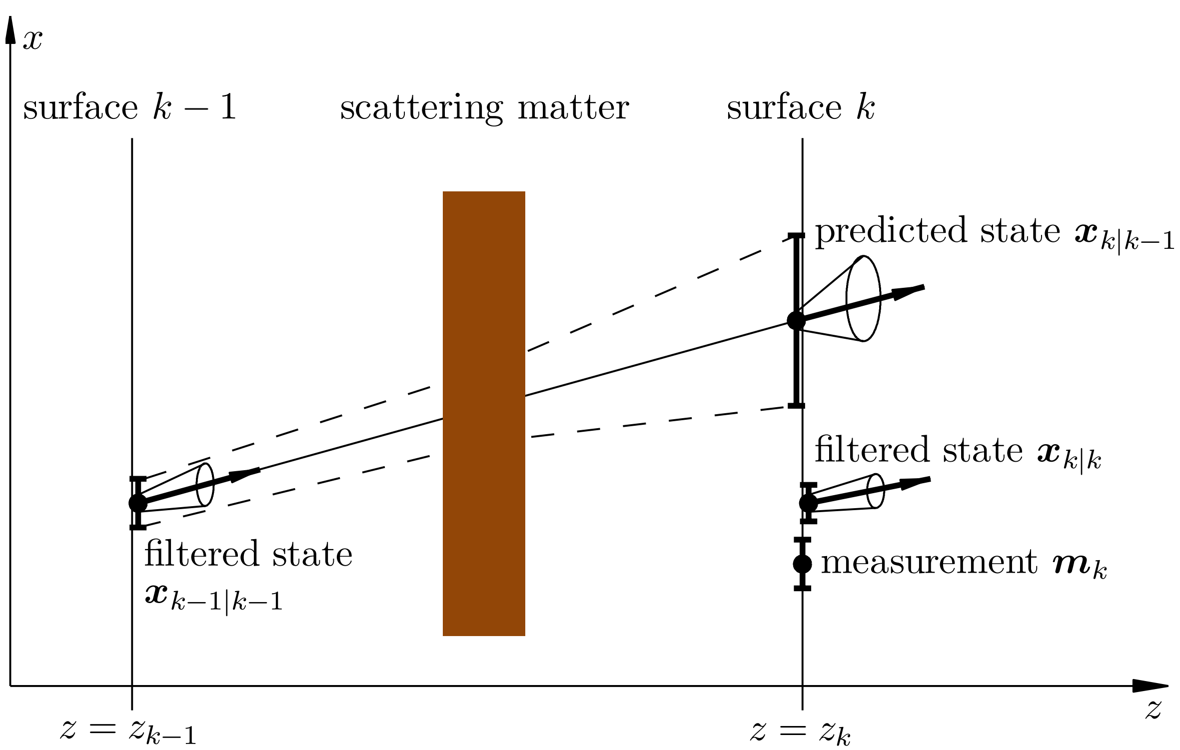
- increase covariance by "noise" term according to the amount of material crossed
  (scattering has expected mean of 0)
- energy loss is applied deterministically
  (additional noise term for straggling added)

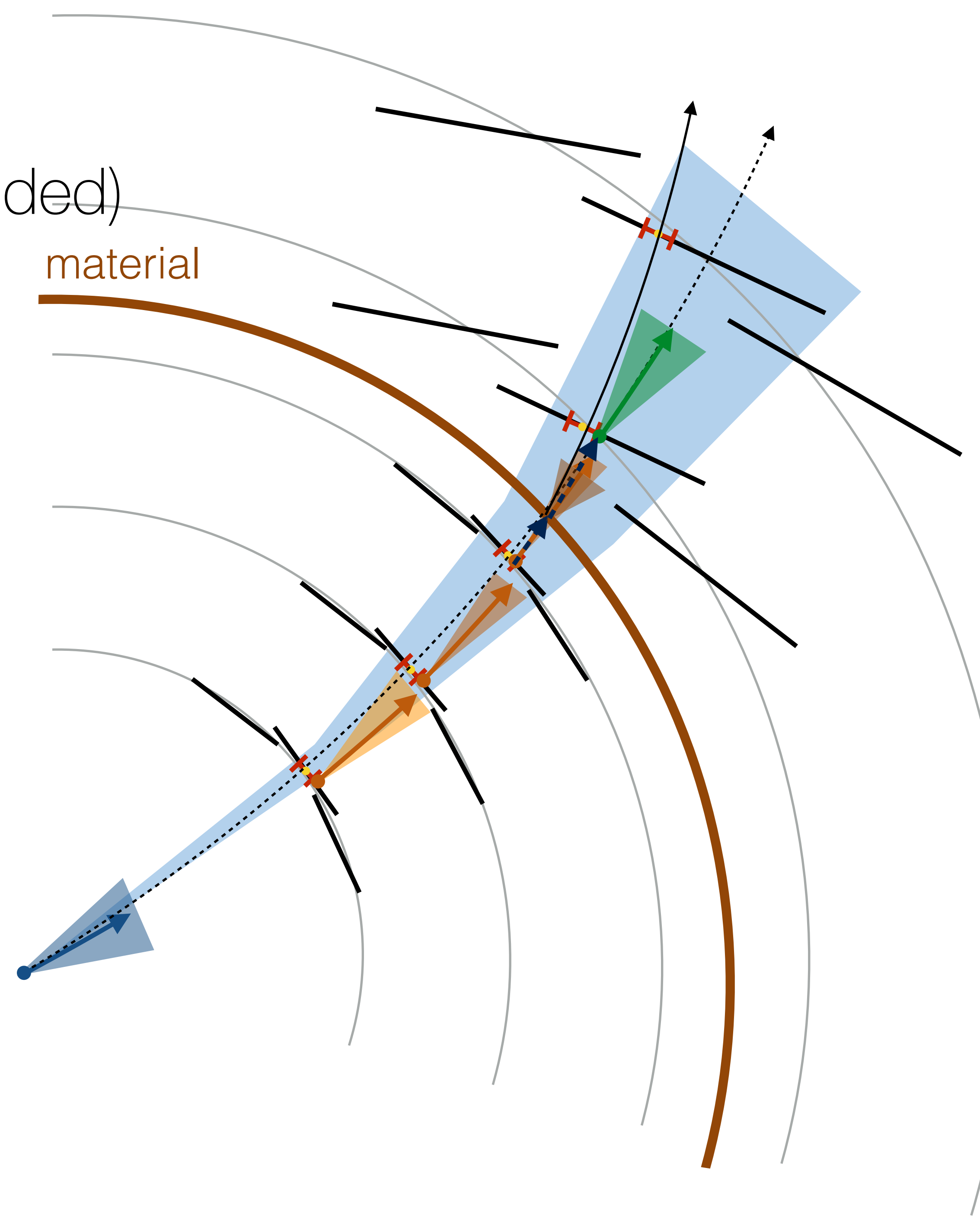


Track fitting Kalman filter with material

when crossing a material layer

- increase covariance by "noise" term according to the amount of material crossed
  (scattering has expected mean of 0)
- energy loss is applied deterministically
  (additional noise term for straggling added)





Track fitting Kalman filter expressed in maths

let's assume the *k-th* filter step

- propagate parameters and **covariances** from *k-1* to *k* adding noise $\boldsymbol{Q}_k$

$$\boldsymbol{q}_{k|k-1} = \boldsymbol{f}_{k|k-1}(\boldsymbol{q}_{k-1|k-1})$$

$$\boldsymbol{C}_{k|k-1} = \boldsymbol{F}_{k|k-1}\boldsymbol{C}_{k-1|k-1}\boldsymbol{F}^{\mathrm{T}}_{k|k-1} + \boldsymbol{Q}_k$$

- update the prediction with measurement

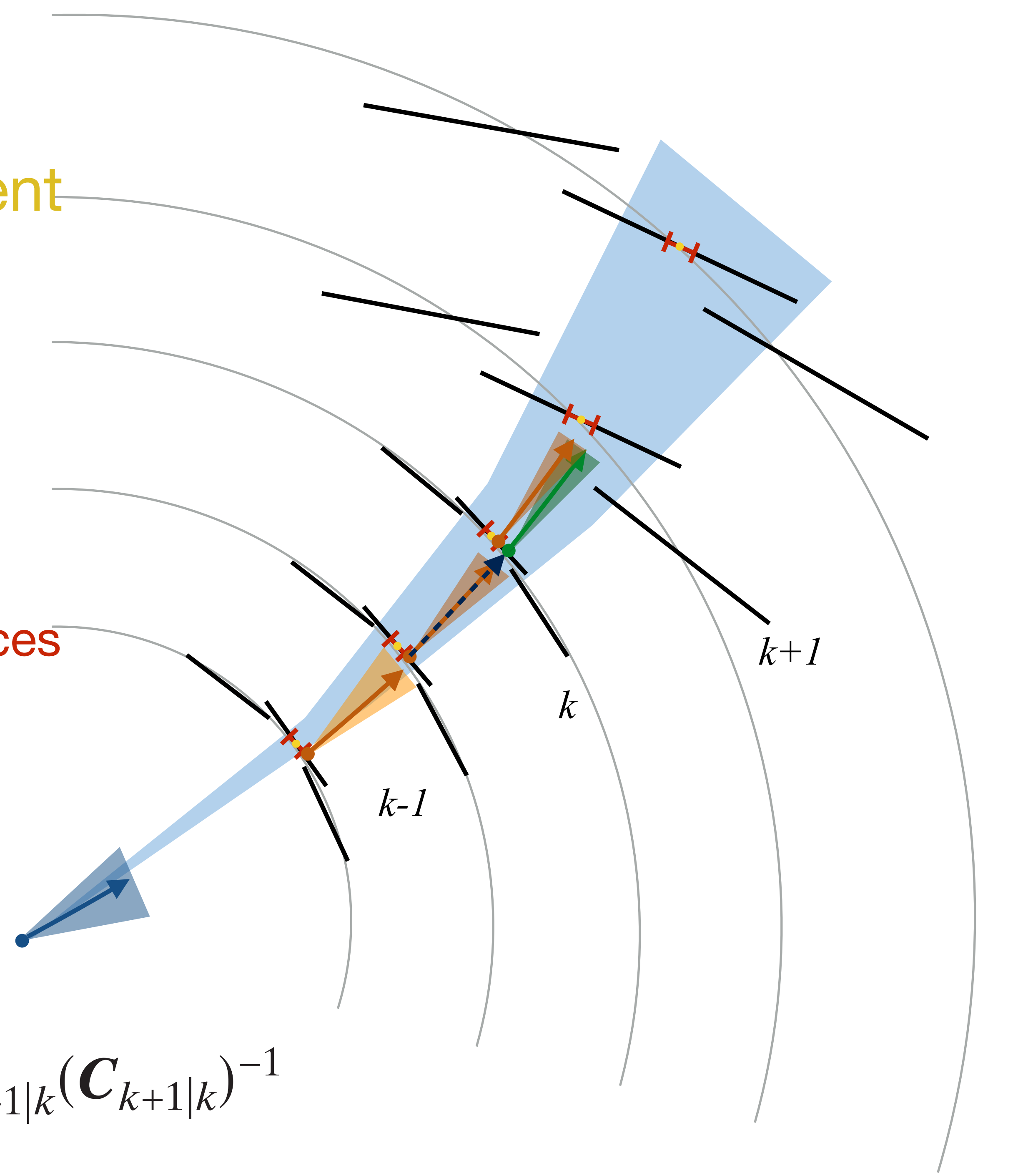
$$\boldsymbol{q}_{k|k} = \boldsymbol{q}_{k|k-1} + \boldsymbol{K}_k[\boldsymbol{m}_k - \boldsymbol{h}_k(\boldsymbol{q}_{k|k-1})]$$

$$\boldsymbol{C}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{C}_{k|k-1}$$

with gain matrix $\boldsymbol{K}_k$:

$$\boldsymbol{K}_k = \boldsymbol{C}_{k|k-1}\boldsymbol{H}^{\mathrm{T}}_k(\boldsymbol{G}_k + \boldsymbol{H}_k\boldsymbol{C}_{k|k-1}\boldsymbol{H}^{\mathrm{T}}_k)^{-1}$$

mapping measurement covariances



run the smoother from *k+1* to *k*

$$\boldsymbol{q}_{k|n} = \boldsymbol{q}_{k|k} + \boldsymbol{A}_k(\boldsymbol{q}_{k+1|n} - \boldsymbol{q}_{k+1|k})$$

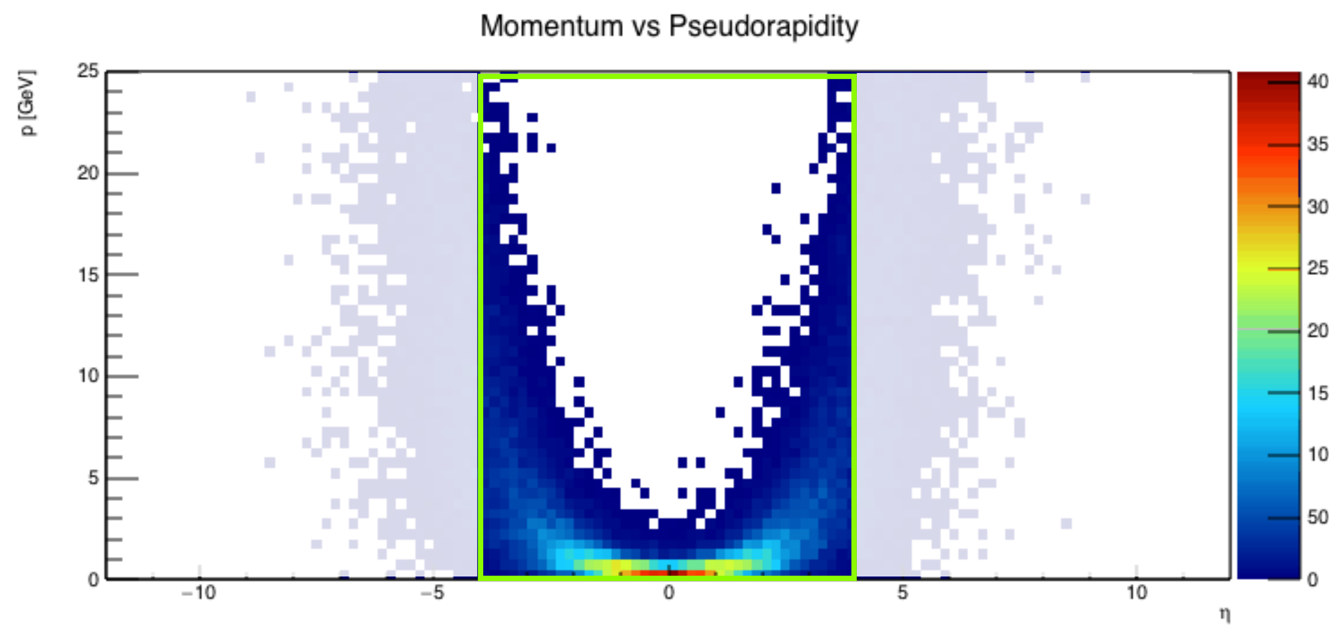
$$\boldsymbol{C}_{k|n} = \boldsymbol{C}_{k|k} - \boldsymbol{A}_k(\boldsymbol{C}_{k+1|k} - \boldsymbol{C}_{k+1|n})\boldsymbol{A}^{\mathrm{T}}_k$$

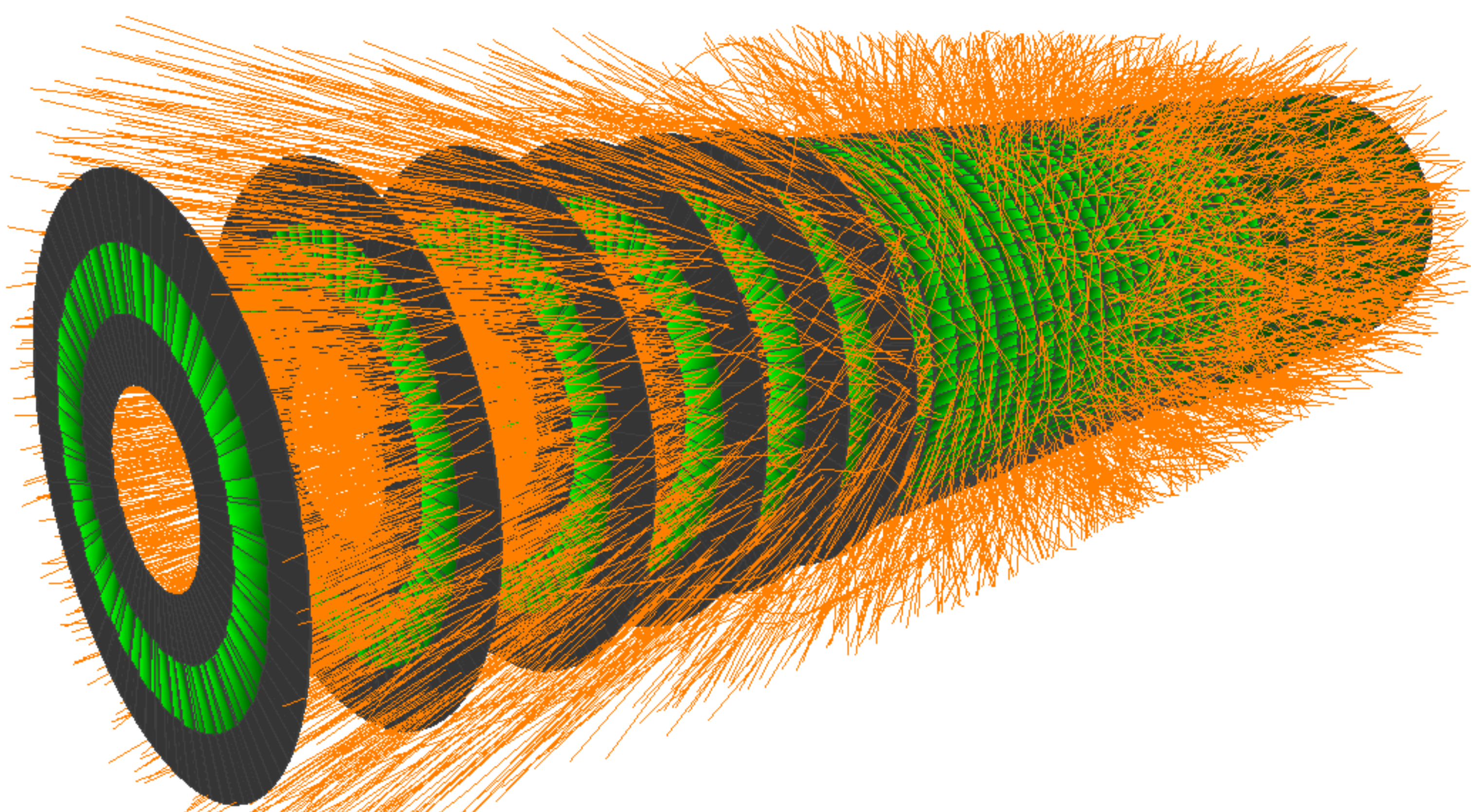
with smoother gain matrix $\boldsymbol{A}_k$: $\boldsymbol{A}_k = \boldsymbol{C}_{k|k}\boldsymbol{F}^{\mathrm{T}}_{k+1|k}(\boldsymbol{C}_{k+1|k})^{-1}$

Track reconstruction at the HL-LHC

HL-LHC environment

- Detector coverage to |η| < 4
  most particles are of <u>low/mid</u> momentum and heavily affected by detector material
- Expected pile-up of <μ> ~ 200
  spread out over a luminous region





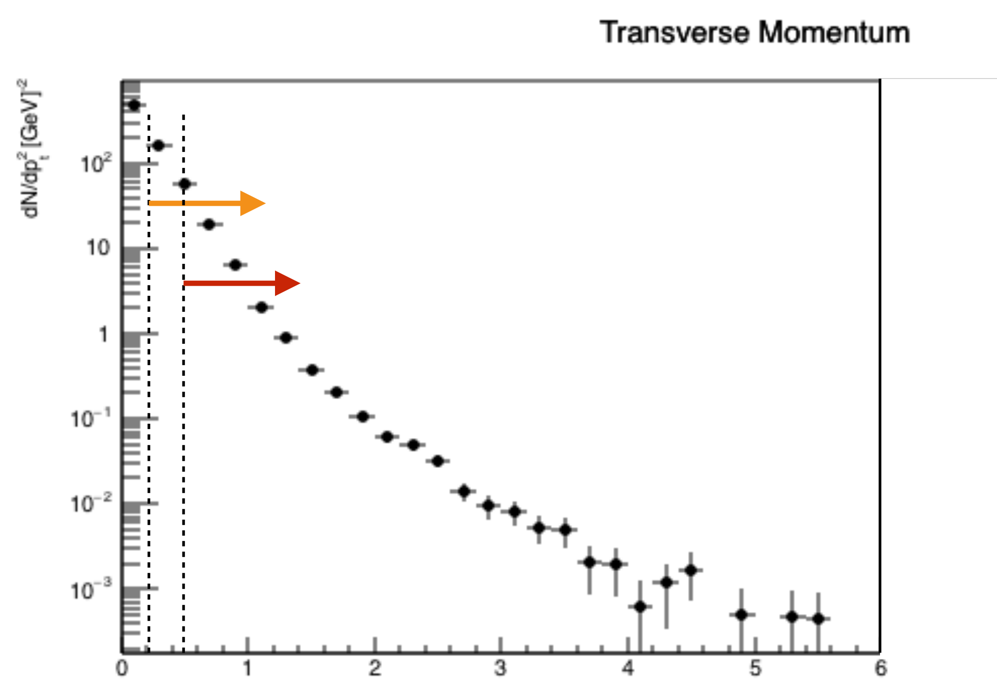
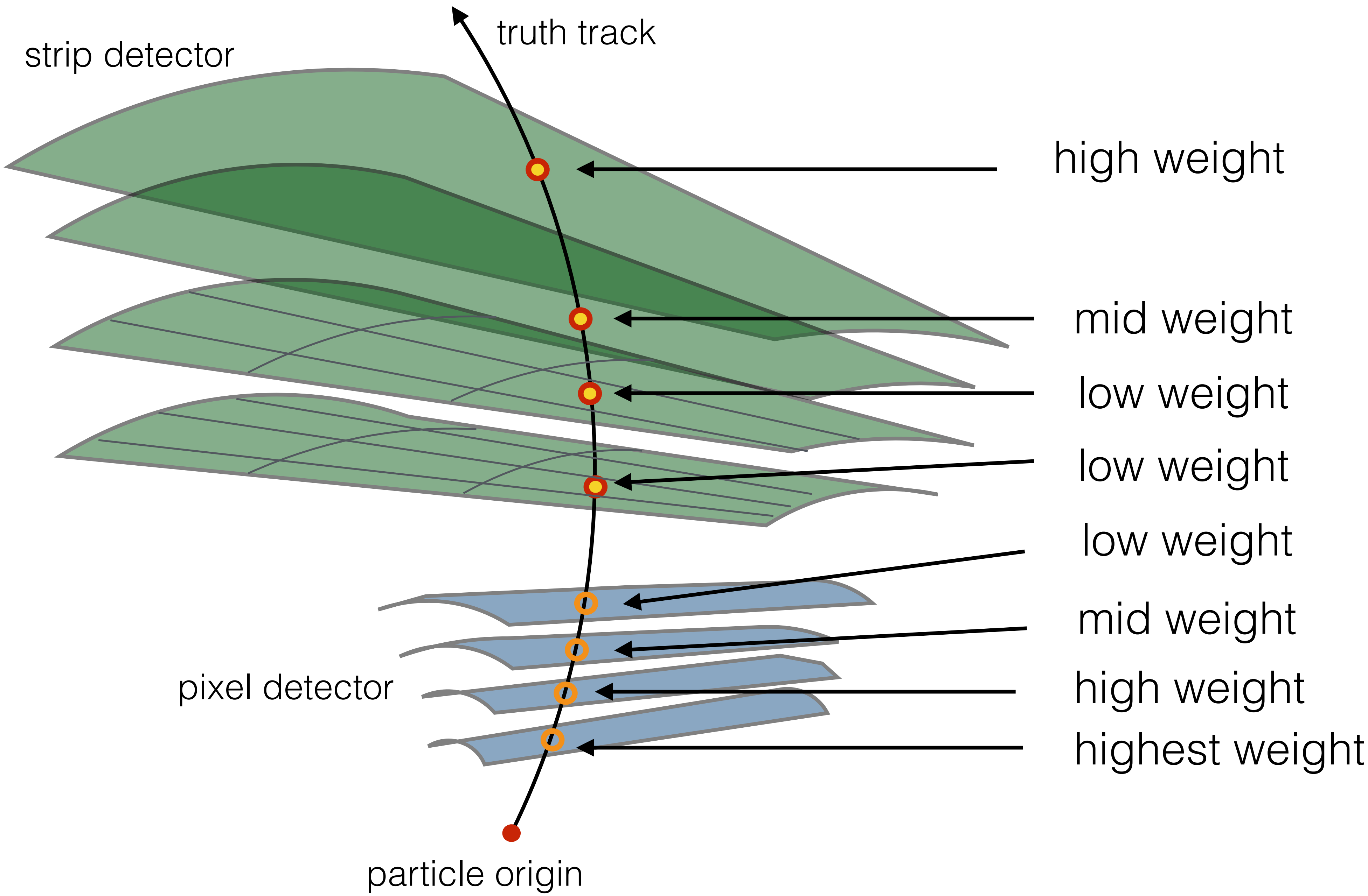


Illustration:
Top right: momentum spectrum for charged particles inside the pseudo rapidity window of |η| < 4.
Bottom right: transverse momentum spectrum of simulated particles, display cut, possible reconstruction cut.
Botton left: simulated event with very high event pileup (μ = 1000) showing only particles with transverse momentum higher than 250 MeV.

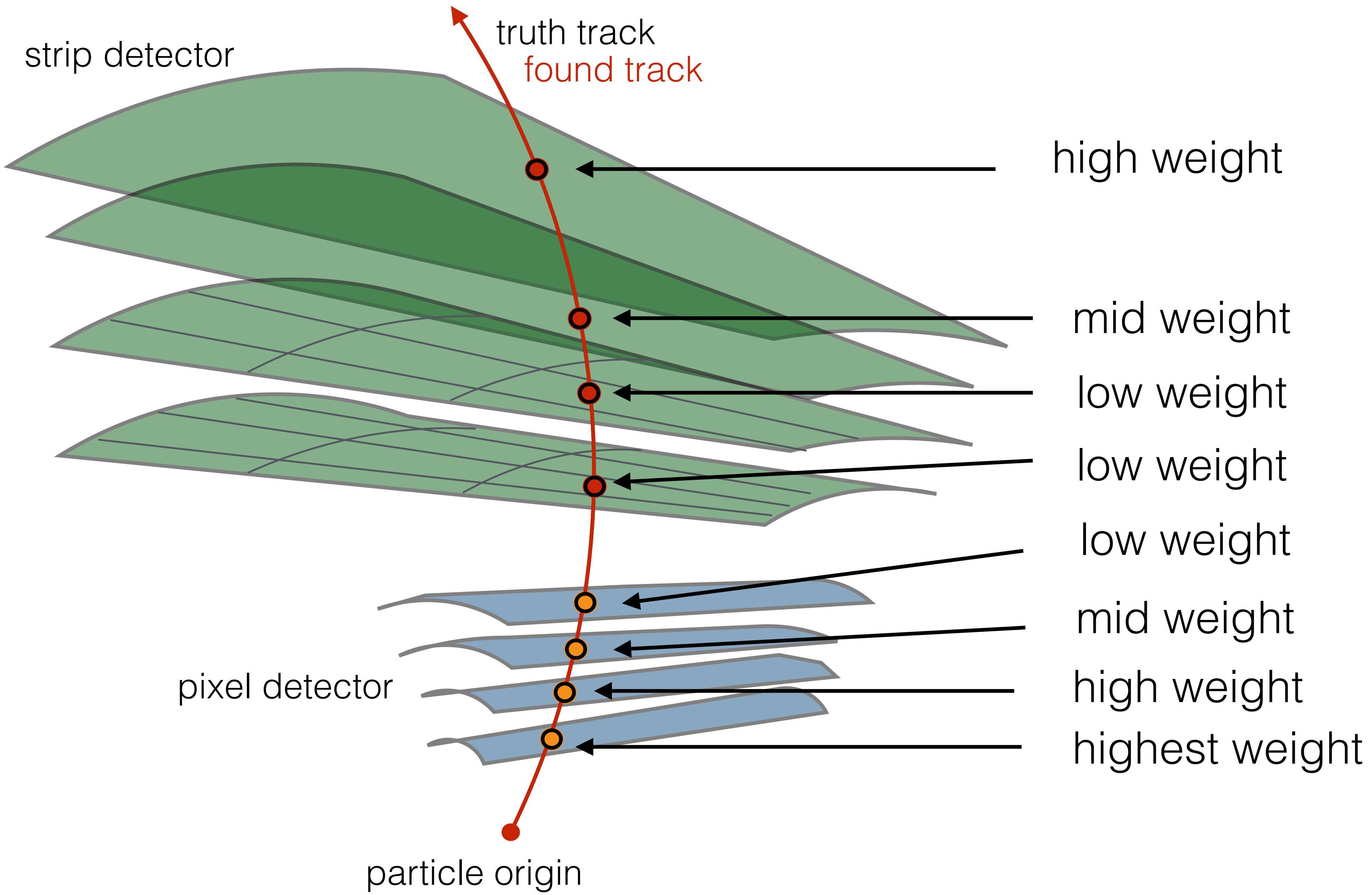
Track ranking proposed setup

Weighted track score

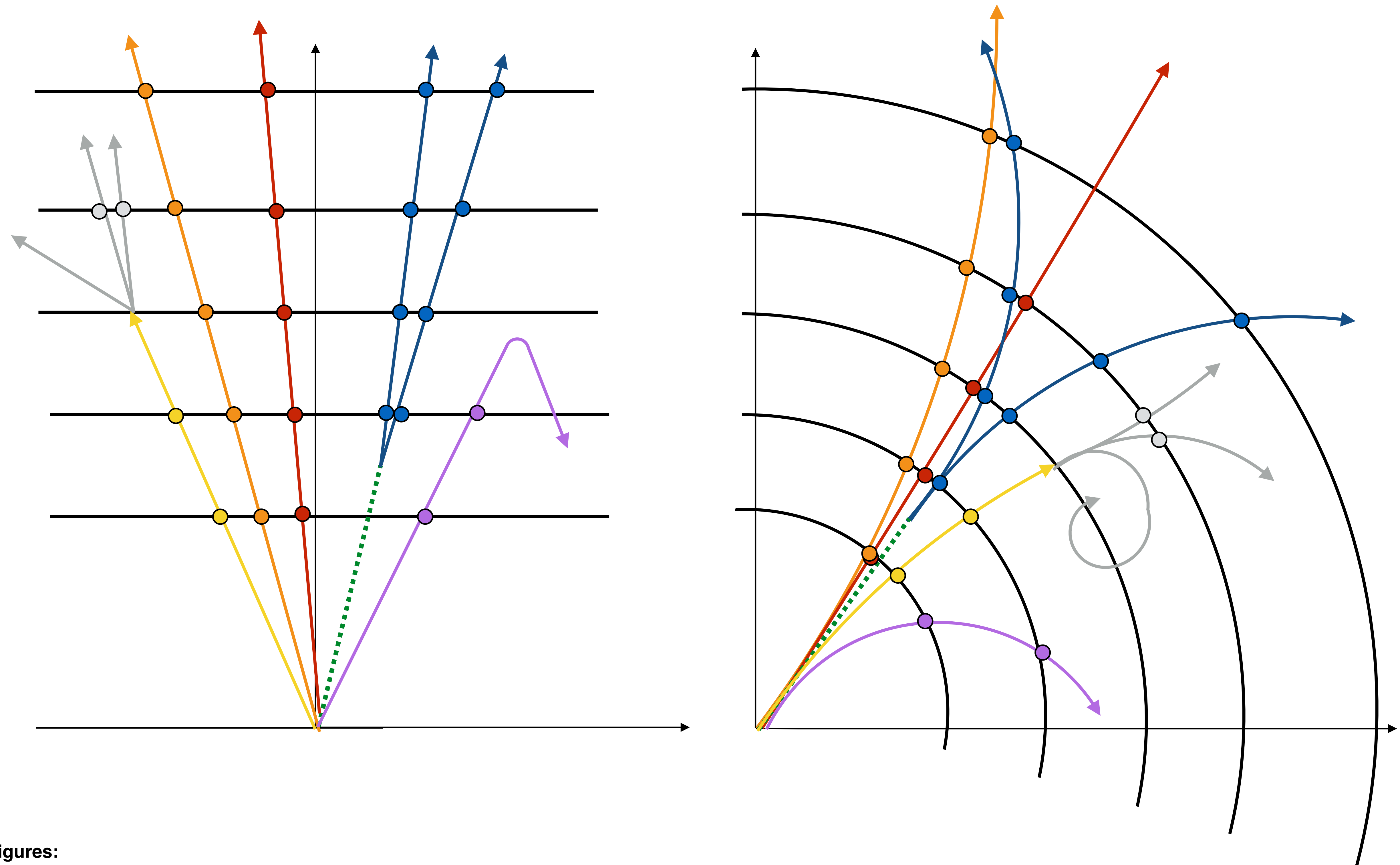


Track ranking proposed setup

Weighted track score

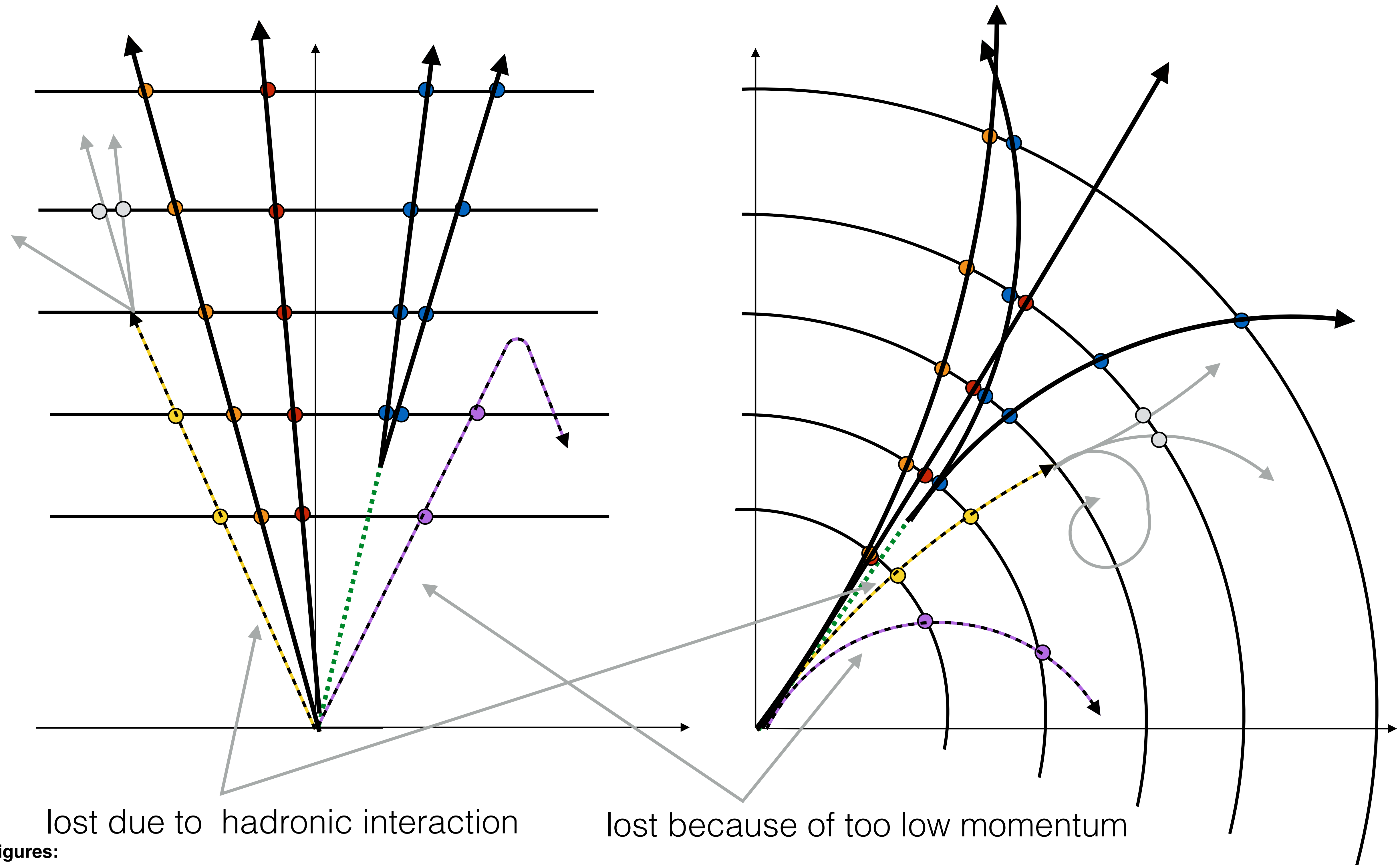


Summary Particles in tracking detectors



Figures:
Longitudinal (left) and transverse (right) view of particles in a detector with material interactions.

Summary Particles in tracking detectors



Figures:
Longitudinal (left) and transverse (right) view of particles in a detector with material interactions.