Machine Learning in Charged Particle Tracking

Hammers & Nails – Machine Learning & HEP
Weizmann Institute – July 19-28 2017

Pre-amble

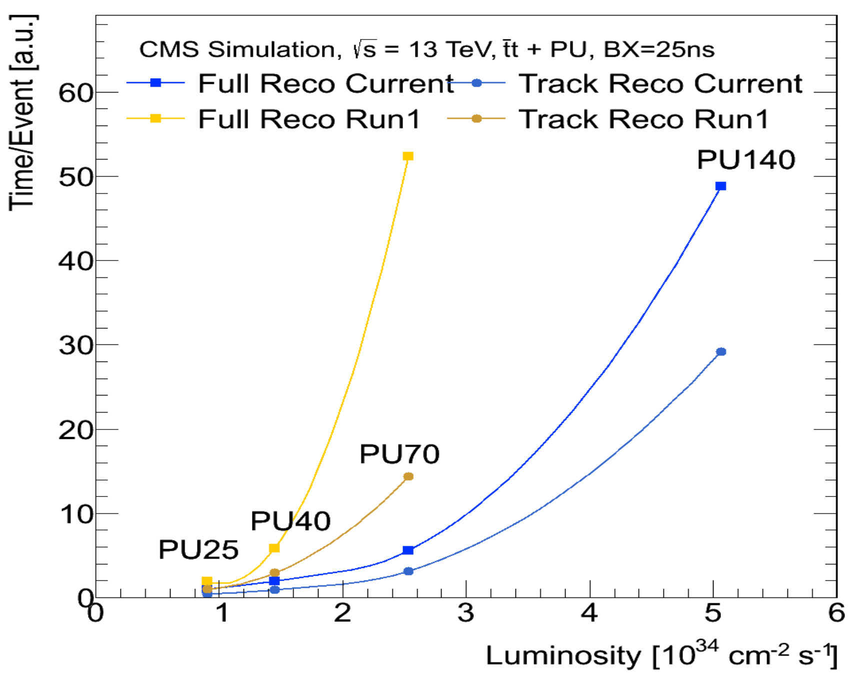
- Assumes the details of the physics of charged particle trajectory through detector is covered.
- Collection of applications of machine learning for tracking-like problems.
- Covering some of the technical details, underlying the challenges and prospects.
- Covering LHC and Neutrino experiments. Many thanks to LHCb : P. Seyfert, ATLAS : D. Rousseau, R. Jansky, A. Salzburger, S. Amrouche, ALICE : M. Floris, R. Shahoyan, CMS : V. Innocente, Neutrino : R. Sulej, A Farbin, HEP.TrkX group for material and input.

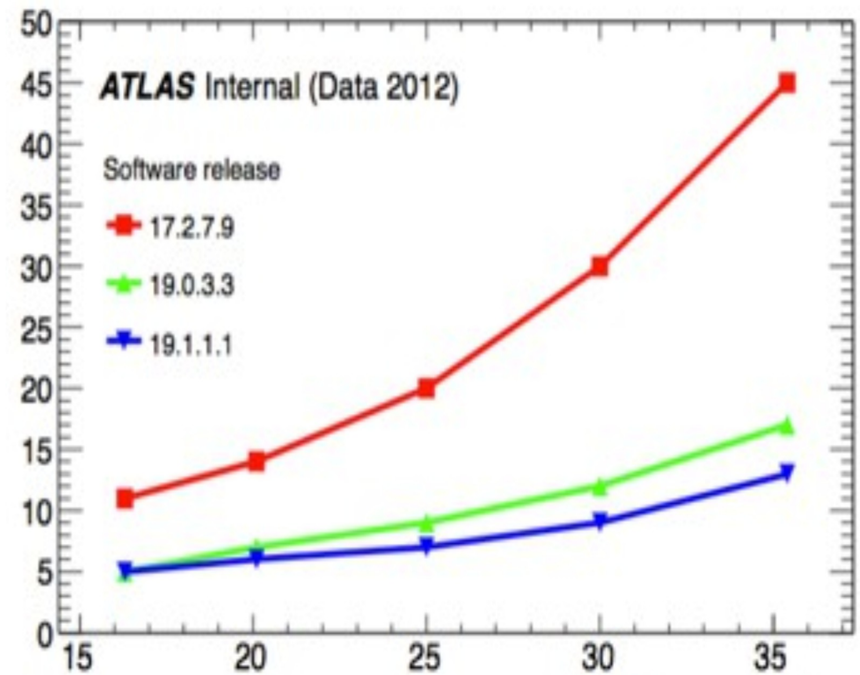
Cost of Tracking

- Charged particle track reconstruction is one of the **most CPU consuming task** in event reconstruction

- **Optimizations (to fit in computational budgets) mostly saturated**

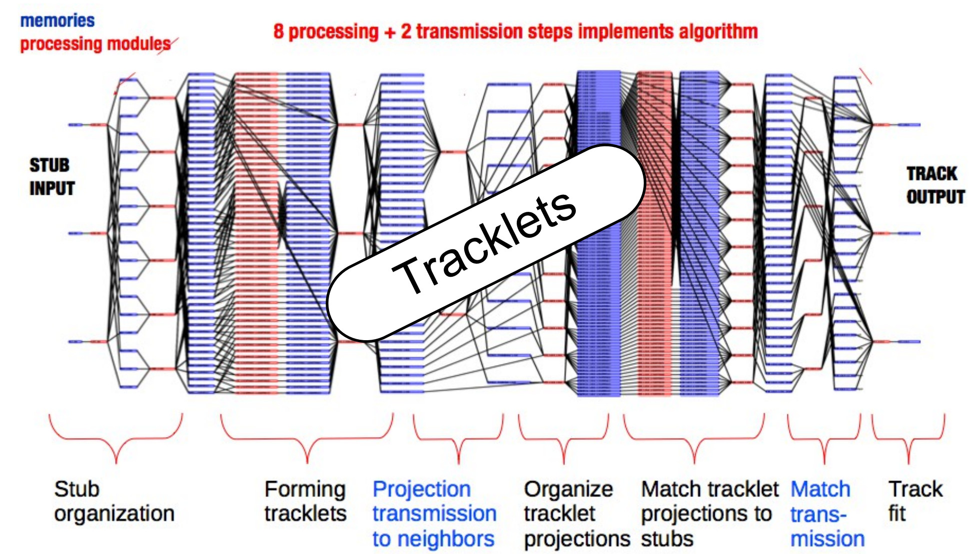
- Large fraction of CPU required in the HLT. **Cannot perform tracking inclusively at CMS and ATLAS.**

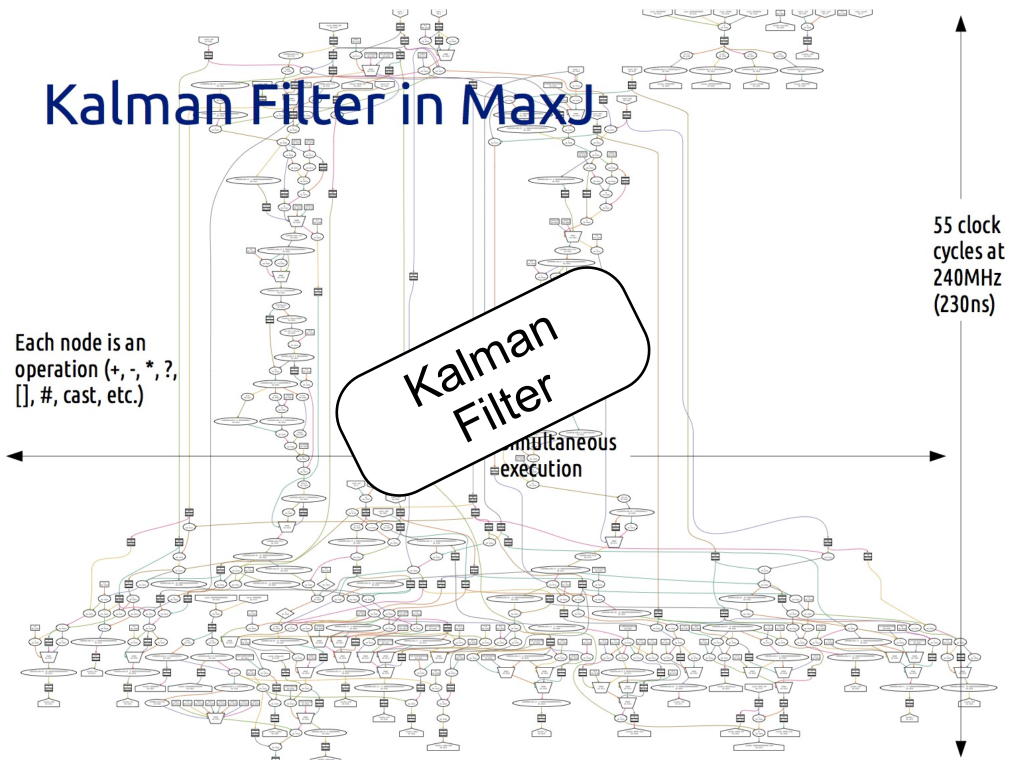


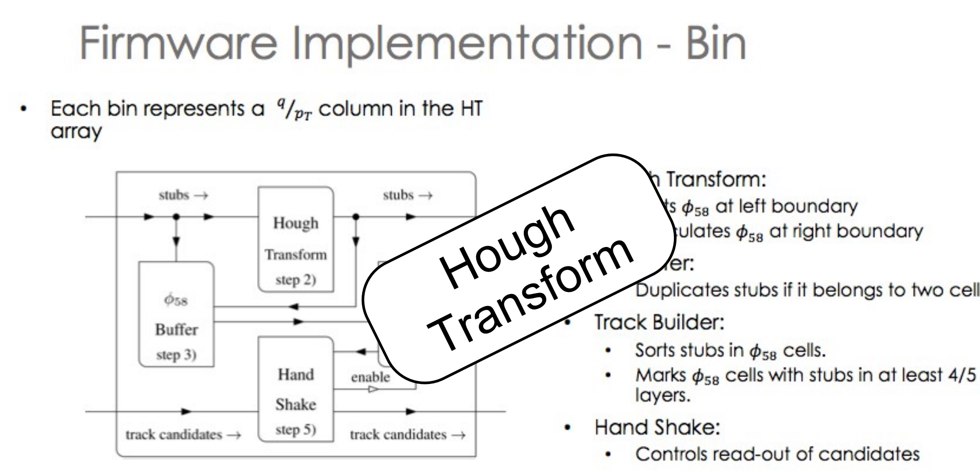


Fast Hardware Tracking

- Track trigger implementation for Trigger upgrades development on-going
- Several approaches investigated
- **Dedicated hardware is the ke**y to fast computation.
- **Not applicable for offline** processing unless by adopting heterogeneous hardware.







See https://ctdwit2017.lal.in2p3.fr/

Outline

I. Challenges and similarities with machine learning

II. Applications of machine learning for tracking

Part I

Similarities and Challenges

- Particle tracking is an active field in data science
- Making a track is called pattern recognition
- Tracking data is much sparser than regular images
- Tracking device may have up to 10M of channels
- Underlying complex geometry of sensors
- Unstable detector geometry ; alignment
- Not the regular type of sequences
- Defining an adequate cost function
- A solution must be performant during inference

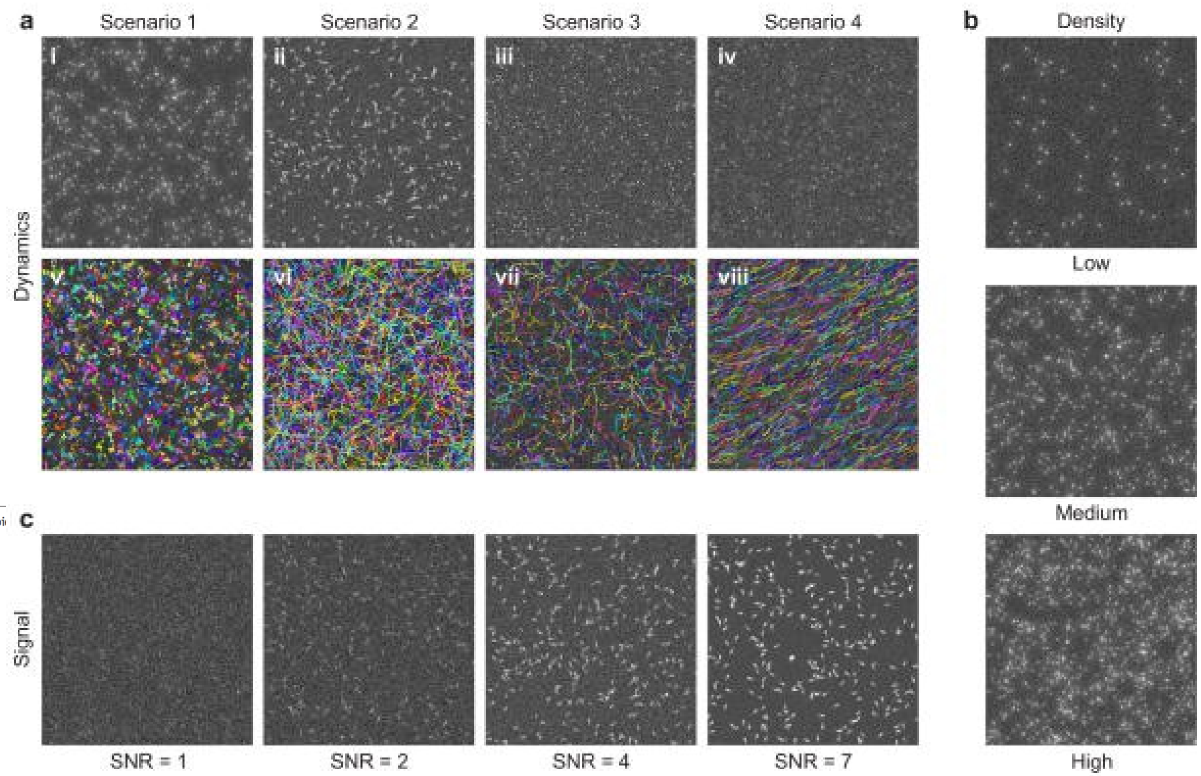
Particle Tracking in Biology

https://www.ncbi.nlm.nih.gov/pubmed/24441936

Table 1 | Participating teams and tracking methods

		Detection			Linking				
Method	**Authors**	**Prefilter**	**Approaches**	**Remarks**	**Principle**	**Approaches**	**Remarks**	**Dim.**	**Refs.**
1	I.F. Sbalzarini Y. Gong J. Cardinale	–	M, C	Iterative intensity-weighted centroid calculation	Combinatorial optimization	MF, MT, GC	Greedy hill-climbing optimization with topological constraints	2D & 3D	32
2	C. Carthel S. Coraluppi	Disk	M, T	Adaptive local-maxima selection	Multiple hypothesis tracking	MF, MT, MM	Motion models are user specified (near-constant position and/or velocity)	2D & 3D	33,34
3	N. Chenouard F. de Chaumont J.-C. Olivo-Marin	Wavelets	M, T	Maxima after thresholding two-scale wavelet products	Multiple hypothesis tracking	MF, MT, MM, GC	Motion models are user specified (near-constant position and/or velocity)	2D & 3D	35–37
4	M. Winter A.R. Cohen	Gaussian, median and morphology	M, T, C	Adaptive Otsu thresholding	Multitemporal association tracking	MF, MT, GC	Post-tracking refinement of detections	2D & 3D	38,39
5	W.J. Godinez K. Rohr	Laplacian of Gaussian or Gaussian fitting	M, T, F, C	Either thresholding + centroid or maxima + Gaussian fitting	Kalman filtering + probabilistic data association	MF, MM	Interacting multiple models using	2D & 3D	29,40
6	Y. Kalaidzidis	Windowed floating mean background subtraction	T, F	Lorentzian function fitting to structures above noise level	Dynamic programming				
7	L. Liang J. Duncan H. Shen Y. Xu	Laplacian of Gaussian	M, T, F	Gaussian mixture model fitting	Multiple hypothesis tracking				
8	K.E.G. Magnusson J. Jaldén H.M. Blau	Deconvolution	M, T, F	Watershed-based clump splitting and parabola fitting	Viterbi algorithm on state-space representation				
9	P. Paul-Gilloteaux	Laplacian of Gaussian or Gaussian filtering	M, T, F	Either maxima with pixel precision (2D) or thresholding + Gaussian fitting (3D)	Nearest neighbor + global optimization				
10	P. Roudot C. Kervrann F. Waharte	Structure tensor	T, F	Histogram-based thresholding and Gaussian fitting	Gaussian template matching				
11	I. Smal E. Meijering	Wavelets	M, F, C	Gaussian fitting (round particles) or centroid calculation (elongated particles)	Sequential multiframe assignment				
12	J.-Y. Tinevez S.L. Shorte	Difference of Gaussian	M, T, F	Parabolic fitting to localized maxima	Linear assignment problem				
13	J. Willemse K. Celler G.P. van Wezel	Gaussian and top hat	T, C	Watershed-based clump splitting	Nearest neighbor				
14	H.-W. Dan Y.-S. Tsai	Gaussian, Wiener and top hat	T, C	Morphological opening–based clump splitting	Nearest neighbor + Kalman filtering				

See **Supplementary Note 1** for further details on methods 1–14. Dim, dimensionality. Detection approaches: M, maxima detection; T, thresholding; F, fitting; C, centroi
GC, gap closing.



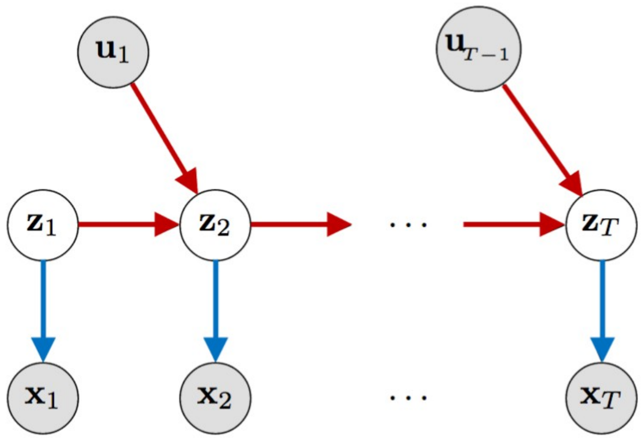
Deep Kalman Filter

Deep Kalman filters

Actions u_t
(e.g., prescribing a medication, performing a surgery)

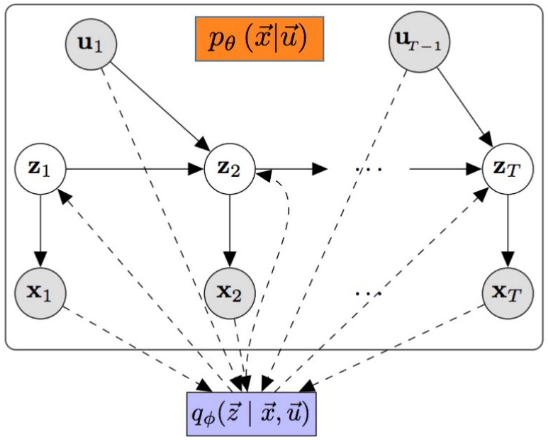
Patient latent state $z_t \in \mathbb{R}^d$

Observations x_t: Lab test results, diagnosis codes, etc.



Optimize *jointly* over generative model $p_\theta(\vec{x}|\vec{u})$ and variational approximation $q_\phi(\vec{z}|\vec{x},\vec{u})$

Stochastic backpropagation
(Rezende et al. 2014, Kingma & Welling, 2014)



Initial state: $z_1 \sim \mathcal{N}(\mu_0, \Sigma_0)$

Action-transition: $z_t \sim \mathcal{N}\left(G_\alpha(z_{t-1}, u_{t-1}), S_\beta(z_{t-1}, u_{t-1})\right)$

Emission: $x_t \sim \Pi(F_\kappa(z_t))$

Uri Shalit at DSHEP2016
https://indico.hep.caltech.edu/indico/conferenceDisplay.py?confId=102

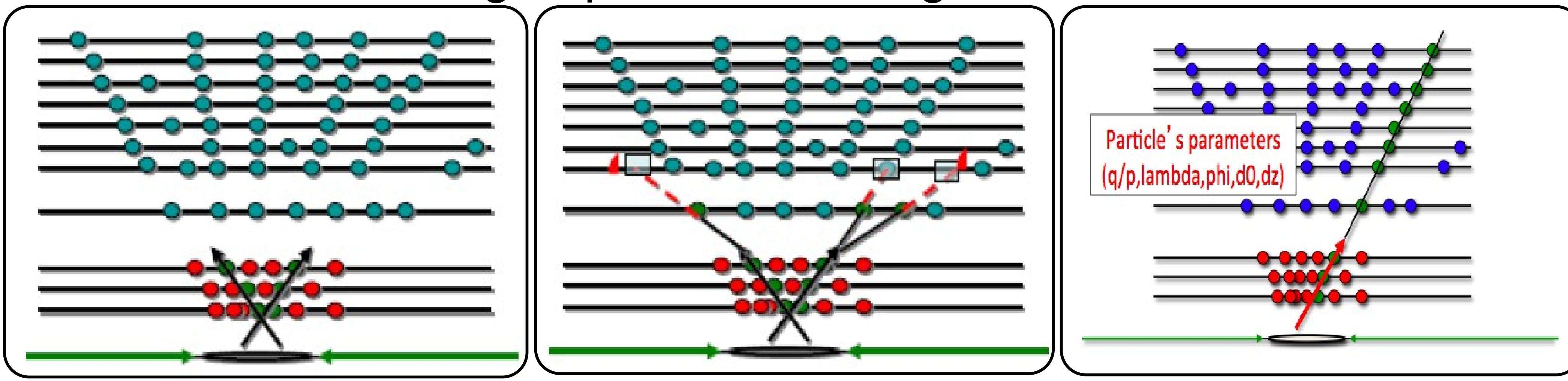
Kalman Filter in Ballistic

- Available methods to track multiple objects using kalman filters
- Deal with “splitting objects”
- Deal with crossing trajetories
- More complexe KF, more computationally intensive ...

Undisclosed contribution during DS@HEP 2016

Pattern Recognition or not

HEP charged particle tracking in a nutshell

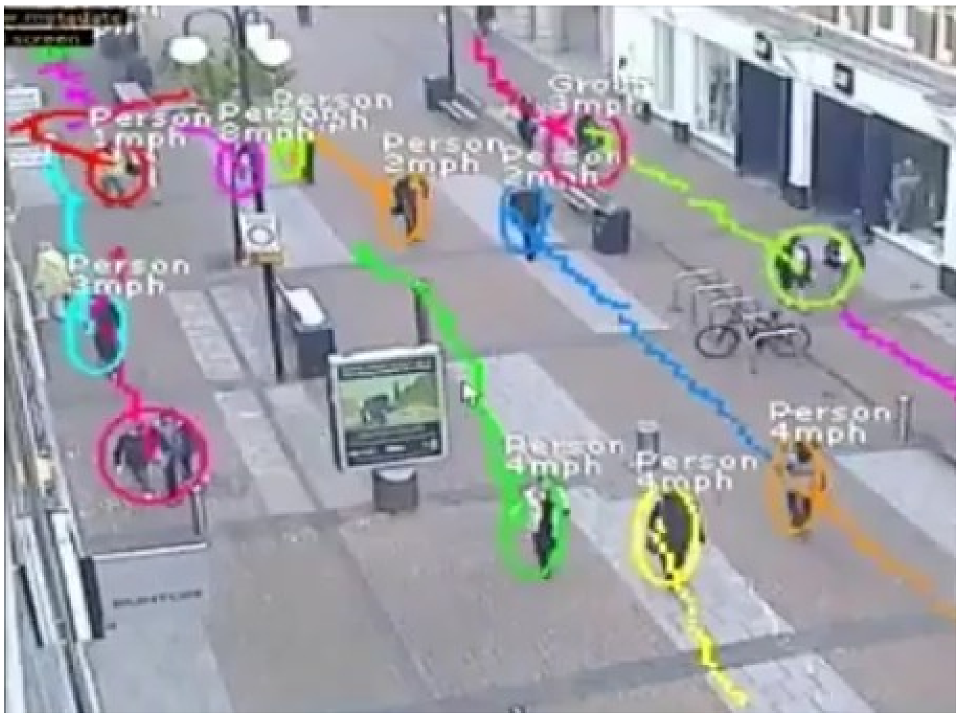


Seeding | Track Building | Track Fitting

- Track building ≡ pattern recognition HEP jargon
- Finding the list of hits belonging to a track ...
- Finding the pattern of hits left by a charged particle in the detector ...

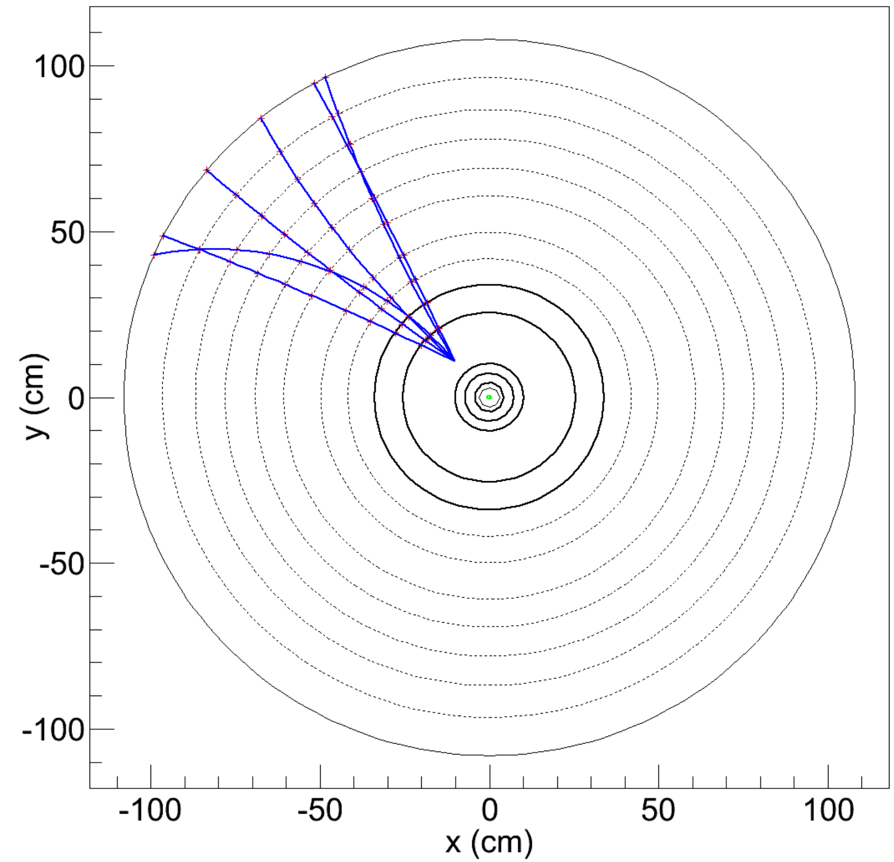
- Not the "usual" data science pattern recognition

Data sparsity

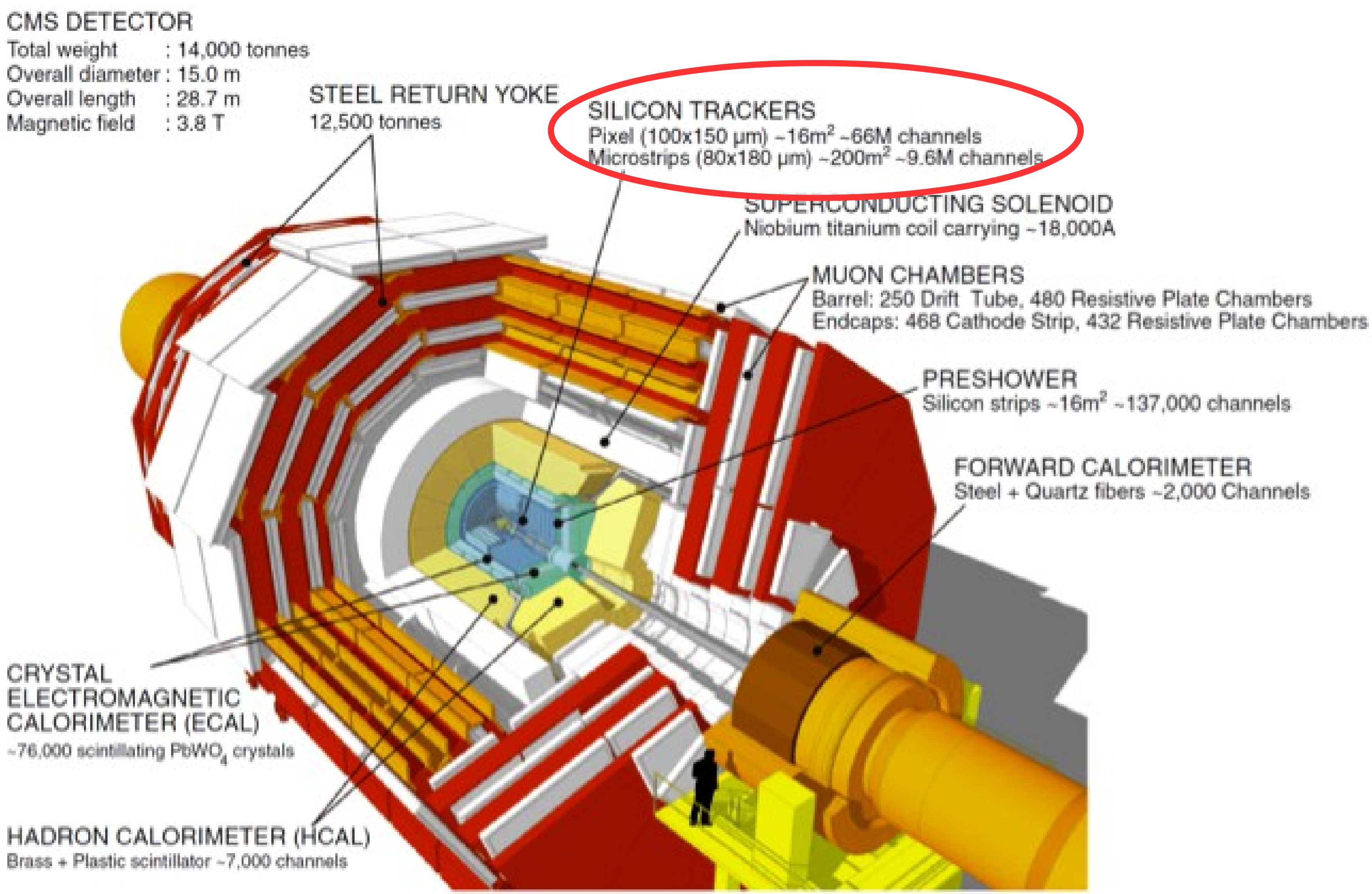


https://privacysos.org/

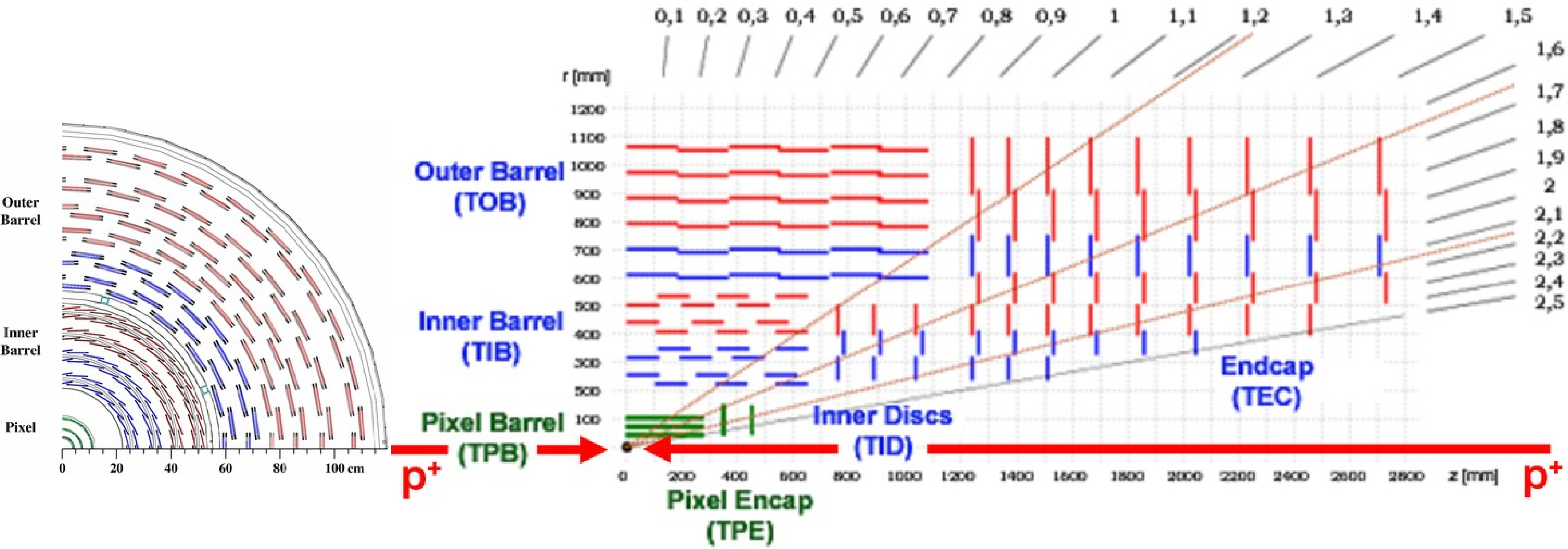
≠



High Dimensionality

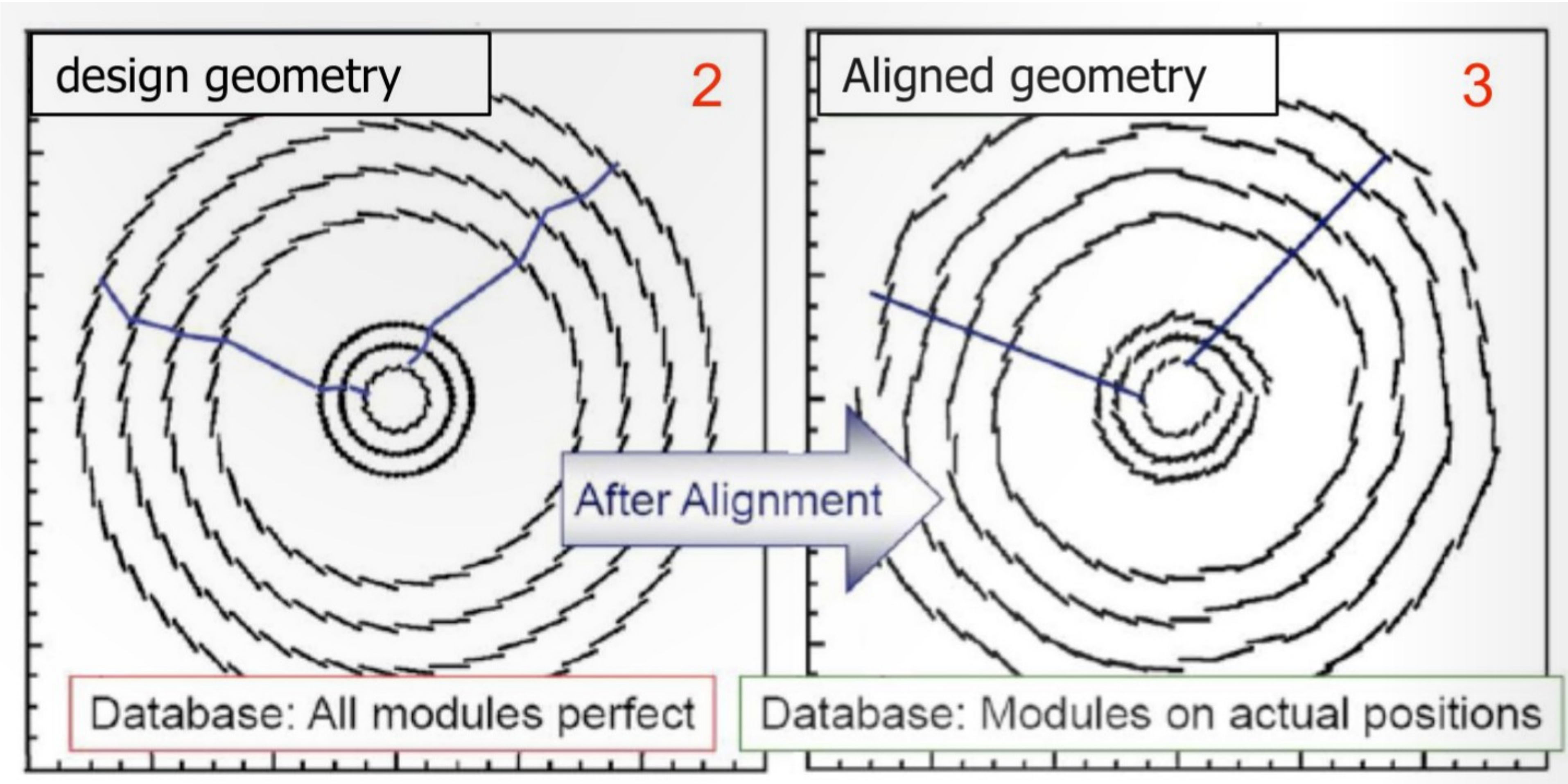


Complex Geometry



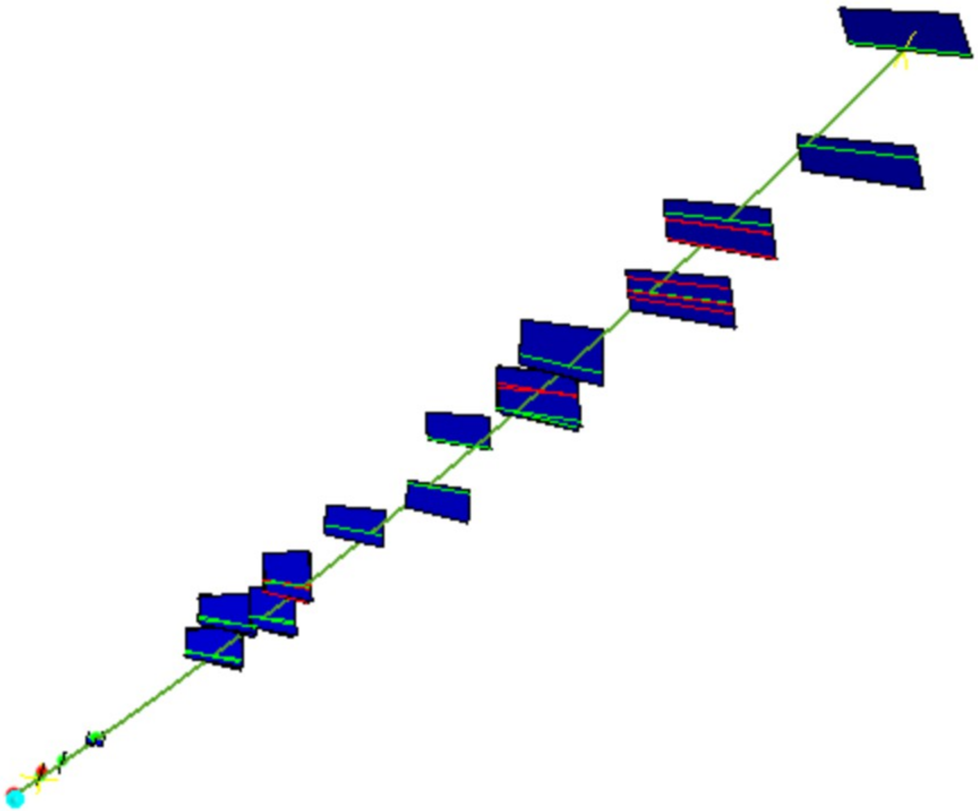
Not the typical data geometry for data science

Mis-aligned Geometry



Mechanical stress (magnetic field, cooling, ...)
does modify the geometry in time

Hit Sequencing



- Hits leave on modules, modules leave on layer, layers are traverse along time.
- "Natural" ordering when trying a hit fitting
- Not so "natural" when doing track building, and hit combinatorics

Figure(s) of Merit(s)

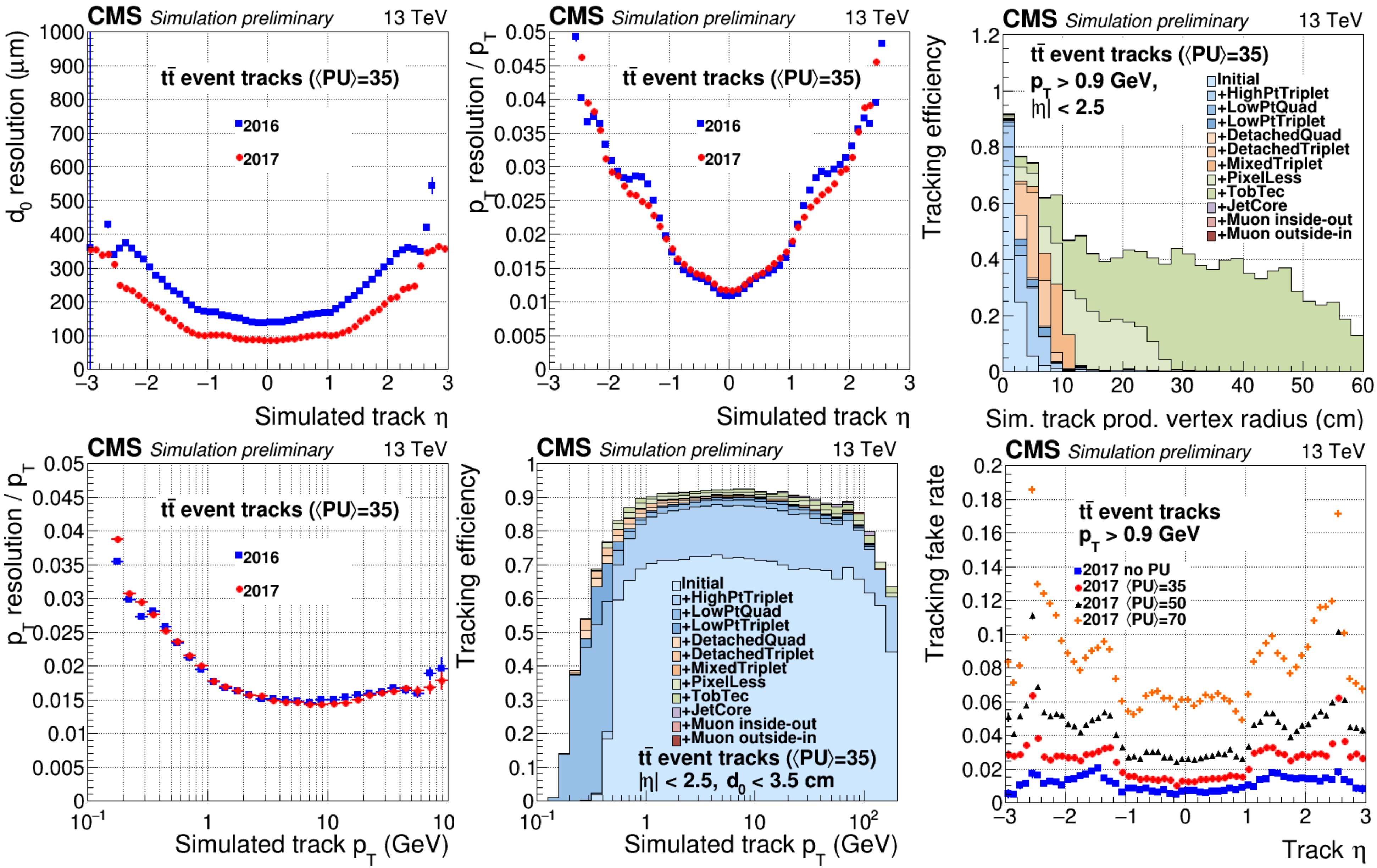


Figure of Merit

- A combination of resolution, fake rate, efficiency, ...

➔ Tracking has been improved within a given a method (CKF+CTF) and within processing time constraints

- Not all tracks are equal. Not all features matter

➔ High dimensional cost function

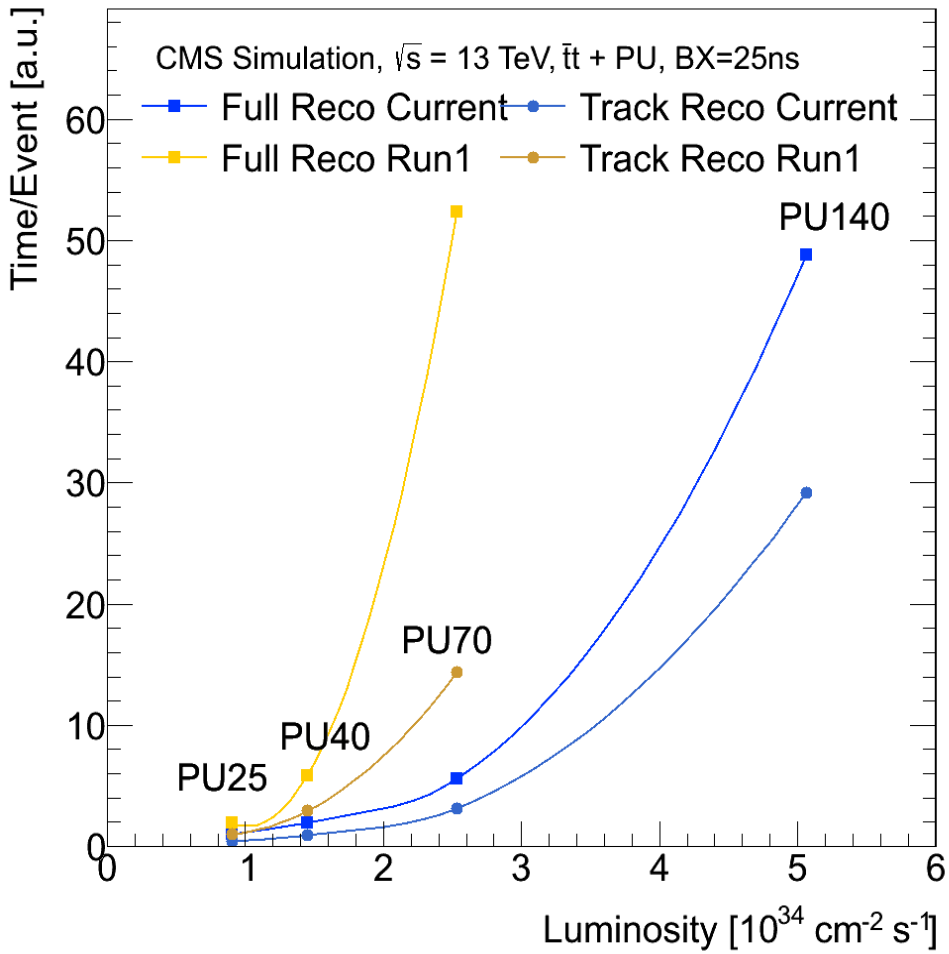
- No golden metric for “tracking” in a general purpose detector

➔ Things would be done differently, if the purpose was different

- Remember the breaking point is computation requirement

➔ Not something that folds in a cost function ...

Computation Performance



- Is this N^2 N^3, ... e^N, ...
- No actual complexity estimation to my knowledge
- PU200 is far off the chart.
- Memory consumption not necessarily an issue

Where ML Can Fit

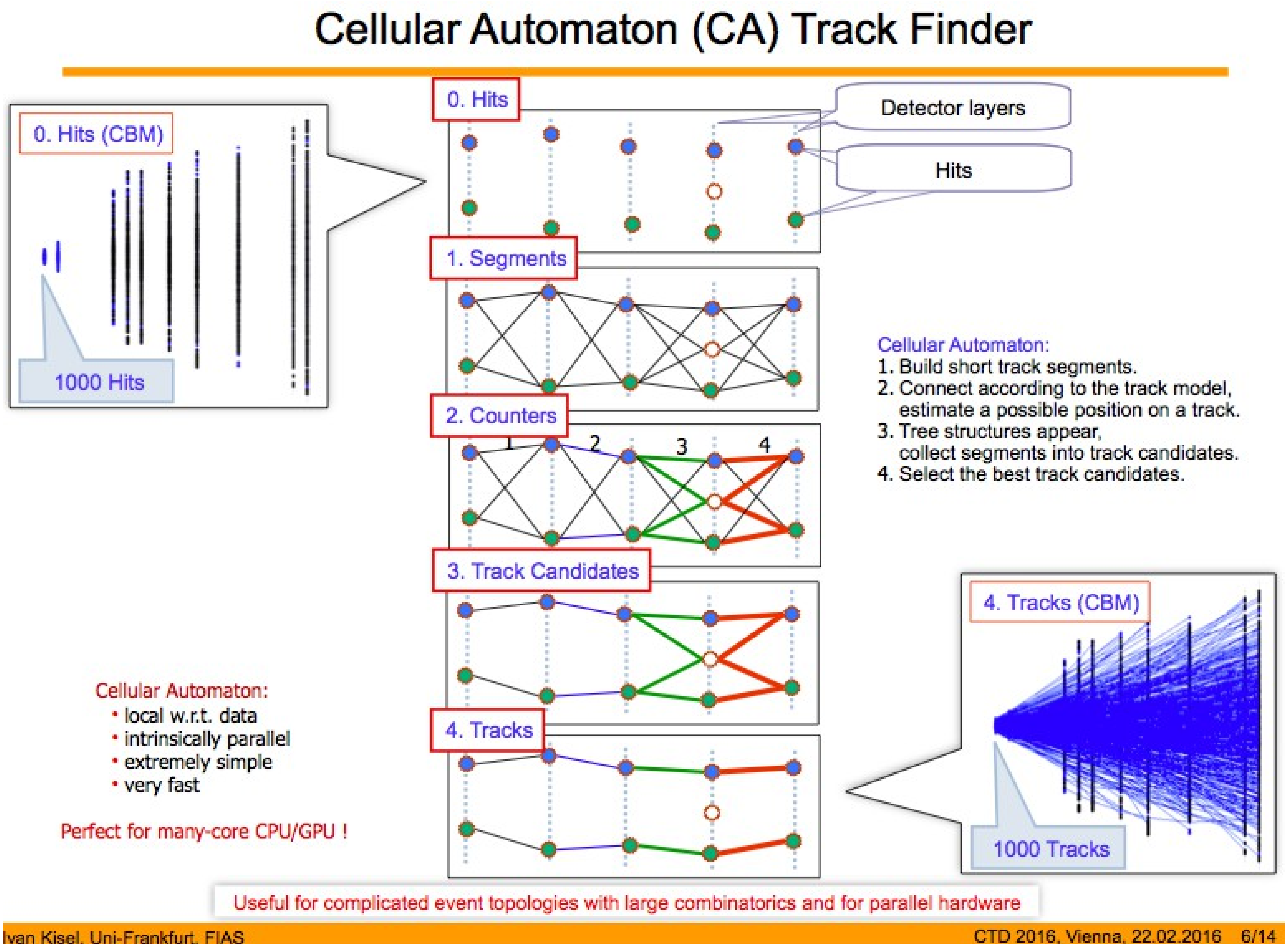
- Signal de-noising (less hit, less combinatorics)
- Making of clusters of hits (less merged, less ambiguity)
- Hits quality (less noise, less combinatorics)
- Seed making (faster composition of tracklets)
- Seed cleaning (less seed, less track making)
- KF pattern recognition
  - In the transport, the update to the new state: deep KF
  - Selecting the best hit candidate
- Pattern recognition
  - Seeded track making
  - Un-seeded track making
- Track fitting
  - Track parameters regression
  - Track parameter reconstruction
- Any combination with other alternative methods (see next slides)
- Any new idea from this workshop ...

Is ML the Only Way Out

No !

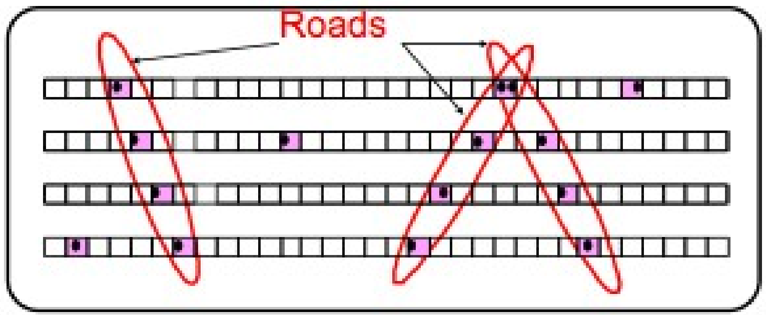
Material from the Connecting the dots Workshop and Intelligent Tracker Workshop Series
https://indico.cern.ch/event/577003/

Cellular Automaton at CBM



Associative Memory

Main feature: avoid hit combinatorics

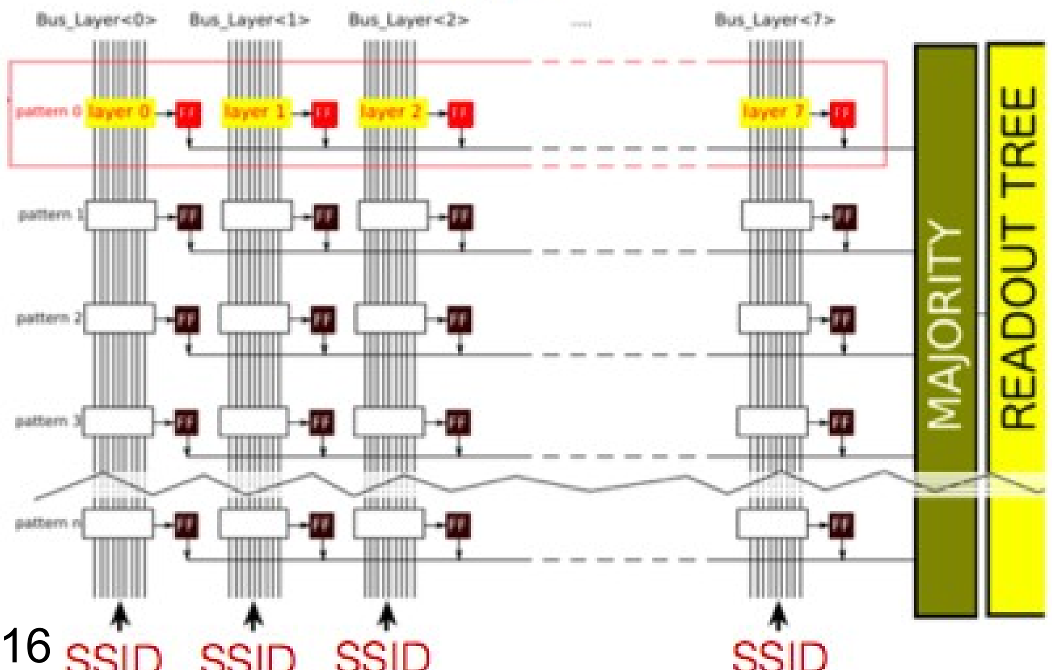


Configuration

- The combinations of resolution-downgraded stubs (SSIDs) generated by the tracks are stored inside the AM chip

Operation

- The SSIDs of each tracker layers are sent to the AM input buses
- The matched road addresses are sent as AM chip output



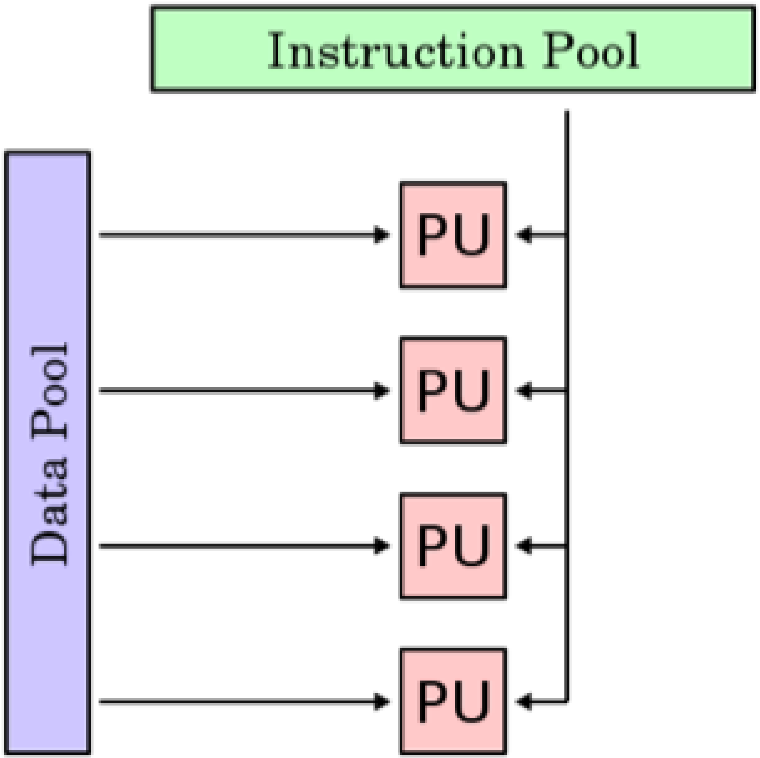
Giacomo Fedi, CTD 2016

Parallelism

- KF tracking cannot ported in straightforward way to run in parallel
- Need to exploit two types of parallelism with parallel architectures

- **Vectorization**
  - Perform the same operation at the same time in lock-step across different data
  - **Challenge: branching** in track building - exploration of multiple track candidates per seed

- **Parallelization**
  - Perform different tasks at the same time on different pieces of data
  - **Challenge: thread balancing** – splitting the workload evenly is difficult as track occupancy in the detector not uniform on a per event basis



Vectorization

Hough Transform

Hough algorithm

Discretised maximum likelihood optimisation over

$$L(n|\{x_i\}) = \sum_i \int \mathrm{d}n\, \delta(d(n, x_i))$$

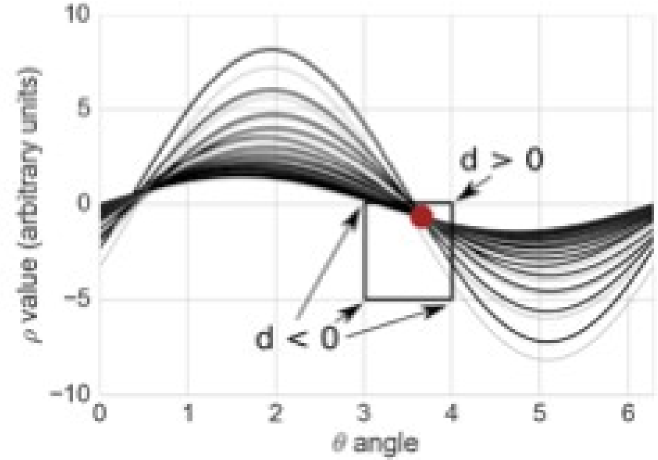
where d is the distance measure of track to hit.
Typically carried out as

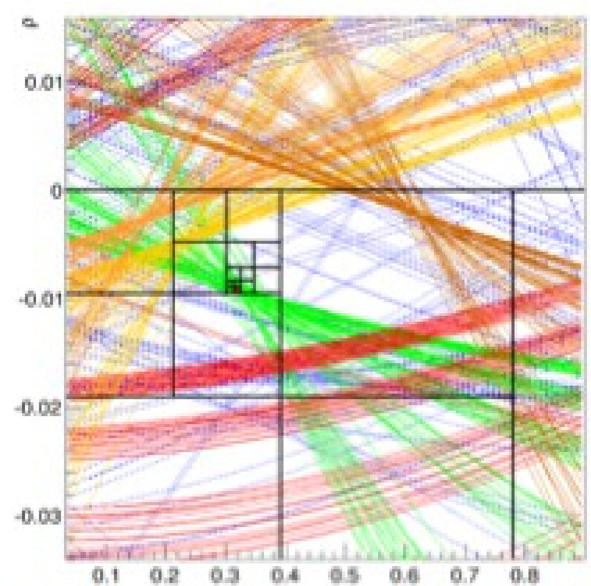
- grid search
- *Fast Hough* bisecting each dimension

over small volumes dn of the parameter space evaluating only the signs of d on the edges.

Refinements

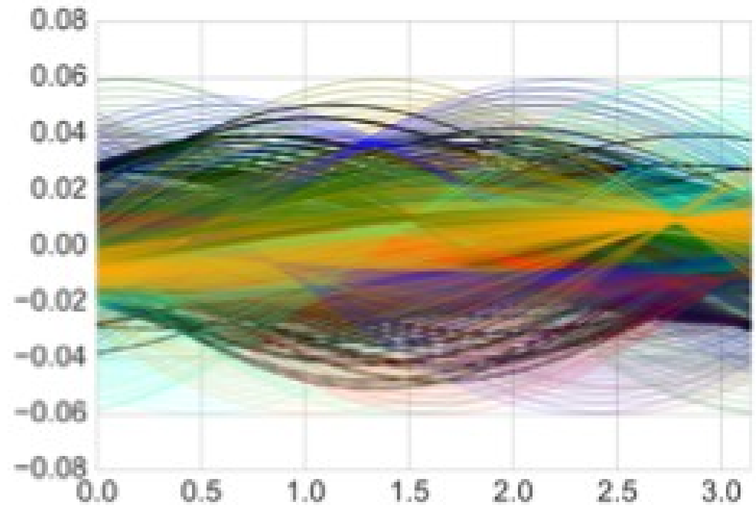
- Weighting of hits versus tracks e.g. on distance d or prior distributions
- Priorisation of search areas
- Overlapping volumes





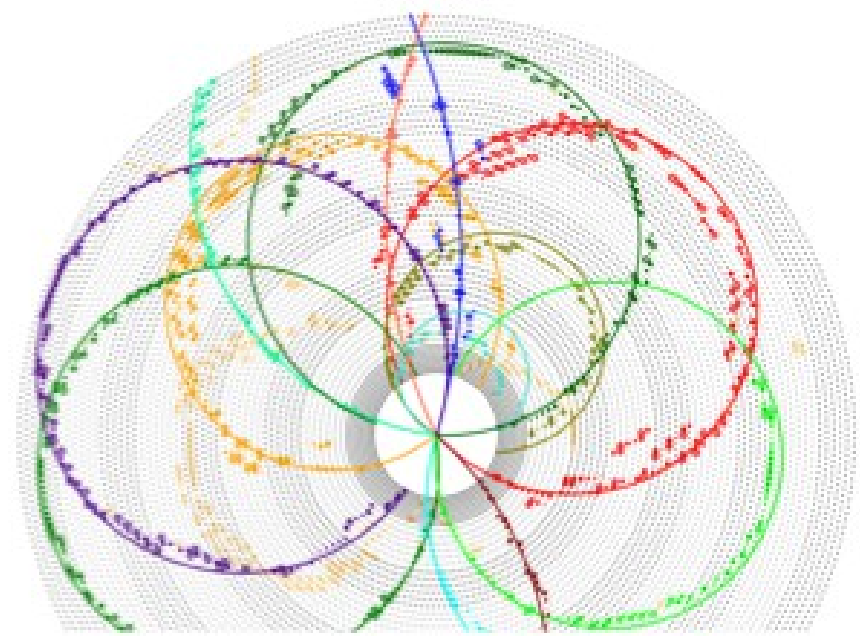
Oliver Frost on behalf of the Belle II collaboration | DESY | 2016-02-22 | Page 4 / 23

Hough Transform



Fast hough search in axial layers

- Working on hits or complete segments
- Multiple passes with priority on high momenta
- Sequential lowering of weight threshold
- Quadtree for structure keeping results of earlier passes
- Sliding bin overlaps pulling towards density centers
- On the fly merging of closeby tracks
- Cleaning and recycling of hits

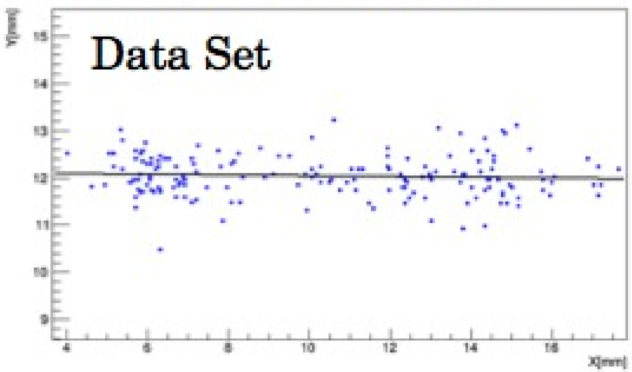


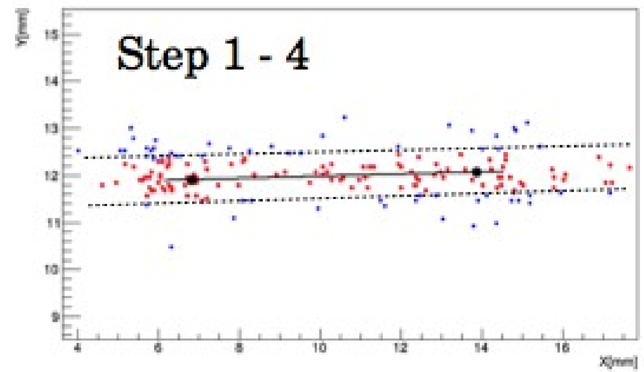
Oliver Frost on behalf of the Belle II collaboration | DESY | 2016-02-22 | Page 19 / 23

RANSAC

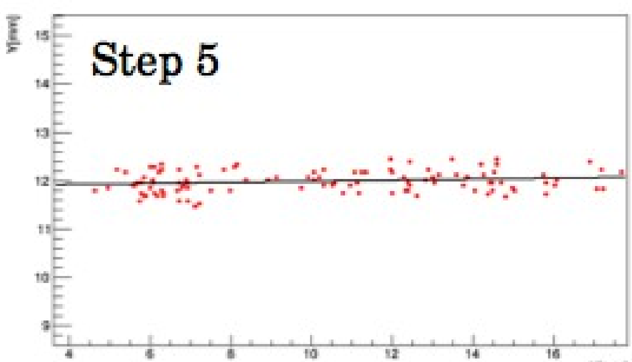
ILC – TPC: RANSAC

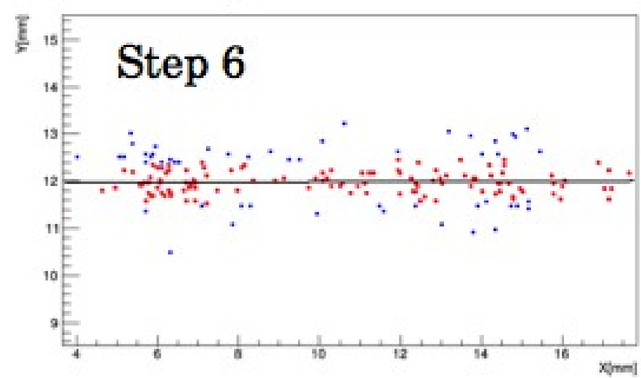
1) Choosing two random hits
2) Define straight line based on these two hits
3) Calculate the distance of all other hits to the line





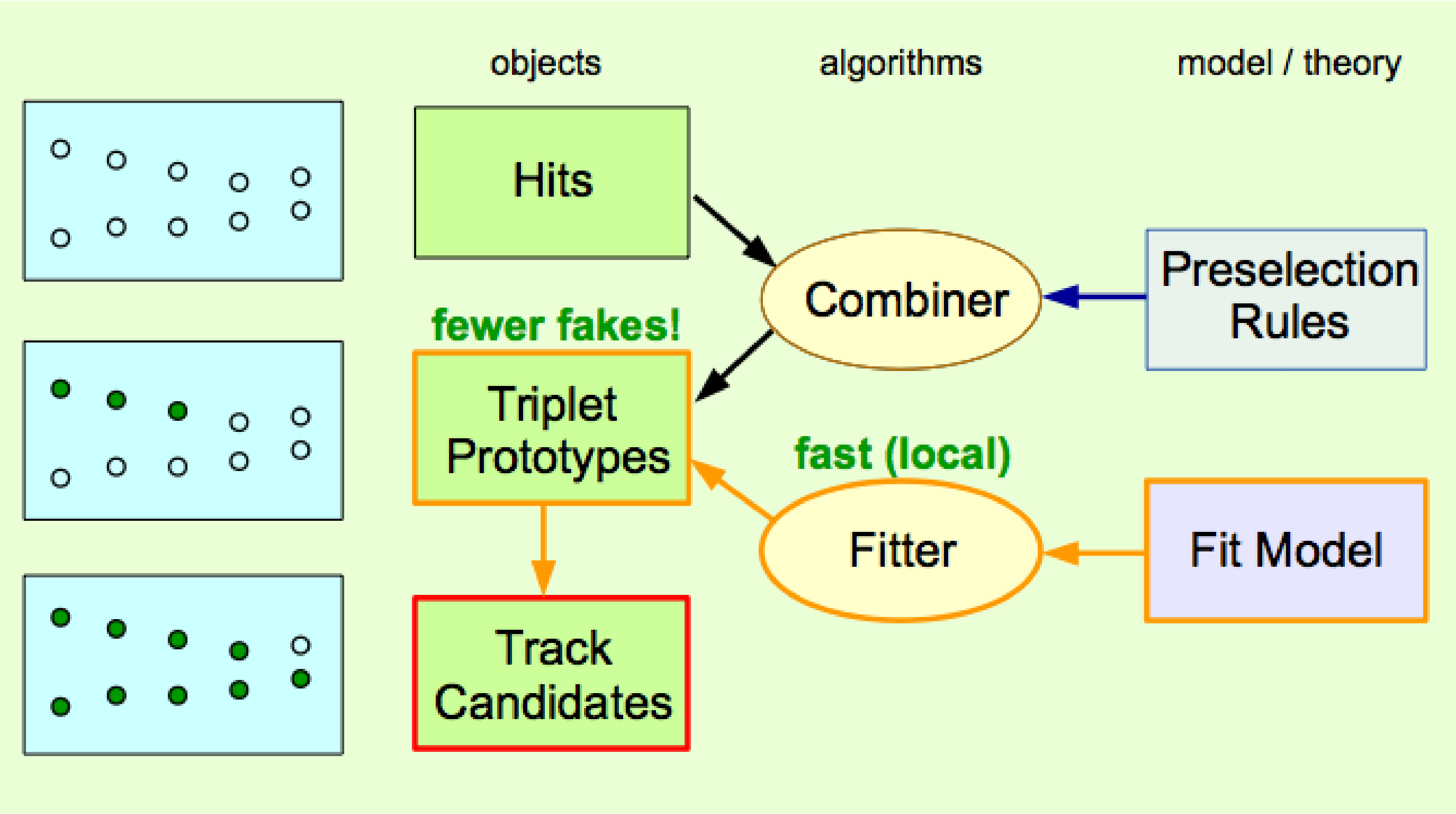
4) Collect those points with the distance less than the tolerance
5) If there are enough points in collection then a solution is found. Fit the line using all points in collection.
6) Recalculate inliers and outliers points based on this fit and then refit again based on new inlier points.
7) Repeat step 1 to 6 N times.
8) If more than one solution is found take solution with smallest chi square.





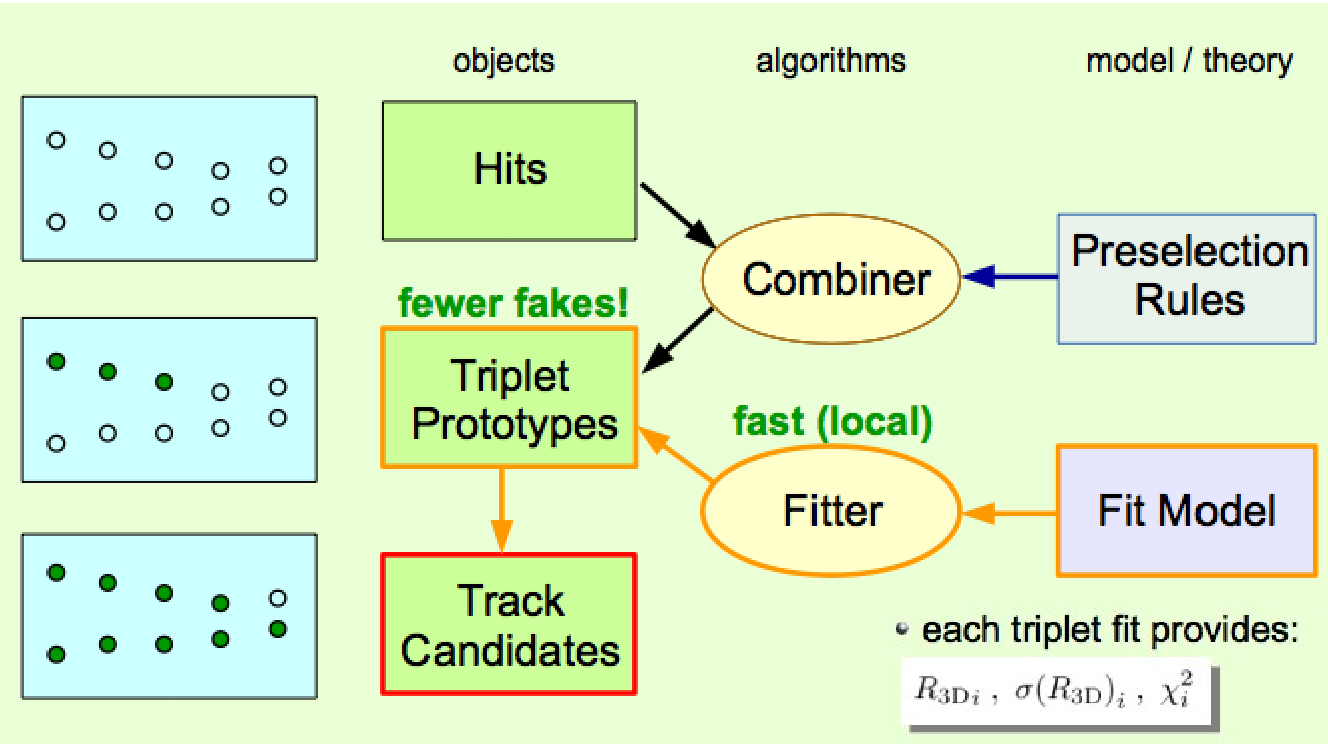
UNIVERSITÄT SIEGEN

Broken Lines



A.Schöning, Heidelberg University

Broken Lines



- each triplet fit provides:

$$R_{3D\,i}\ ,\ \sigma(R_{3D})_i\ ,\ \chi^2_i$$

- averaging of R_{3D} (w/o bias correction)

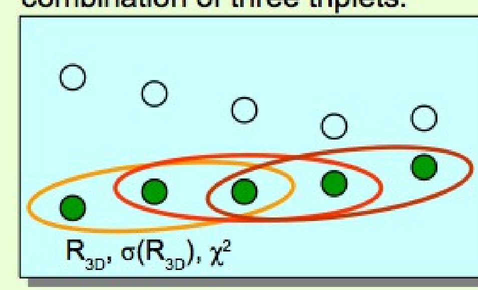
$$\overline{R_{3D}} = \sum_i^{n_{hit}-2} \frac{R_{3D,i}}{\sigma_i(R_{3D})^2} \Big/ \sum_i^{n_{hit}-2} \frac{1}{\sigma_i(R_{3D})^2}$$

- combined χ^2

$$\chi^2_{comb} = \sum_{i=\text{triplet}} \chi^2_i + \frac{(R_{3D,i} - \overline{R_{3D}})^2}{\sigma_i(R_{3D})^2}$$

→ **extremely simple and fast**
→ **allows to check consistency of triplets "on the fly"**
→ **final selection: $\chi^2_{comb} < \chi^2_{cut}$**

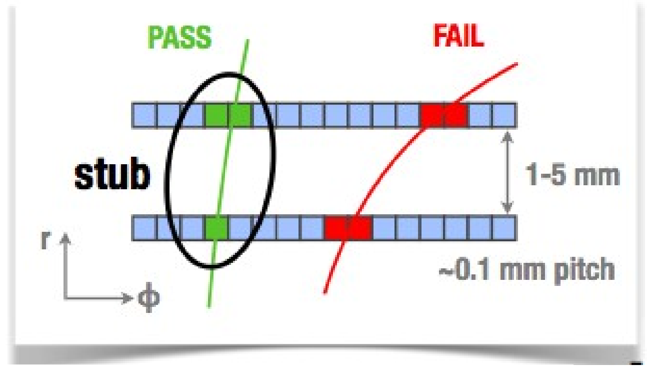
combination of three triplets:

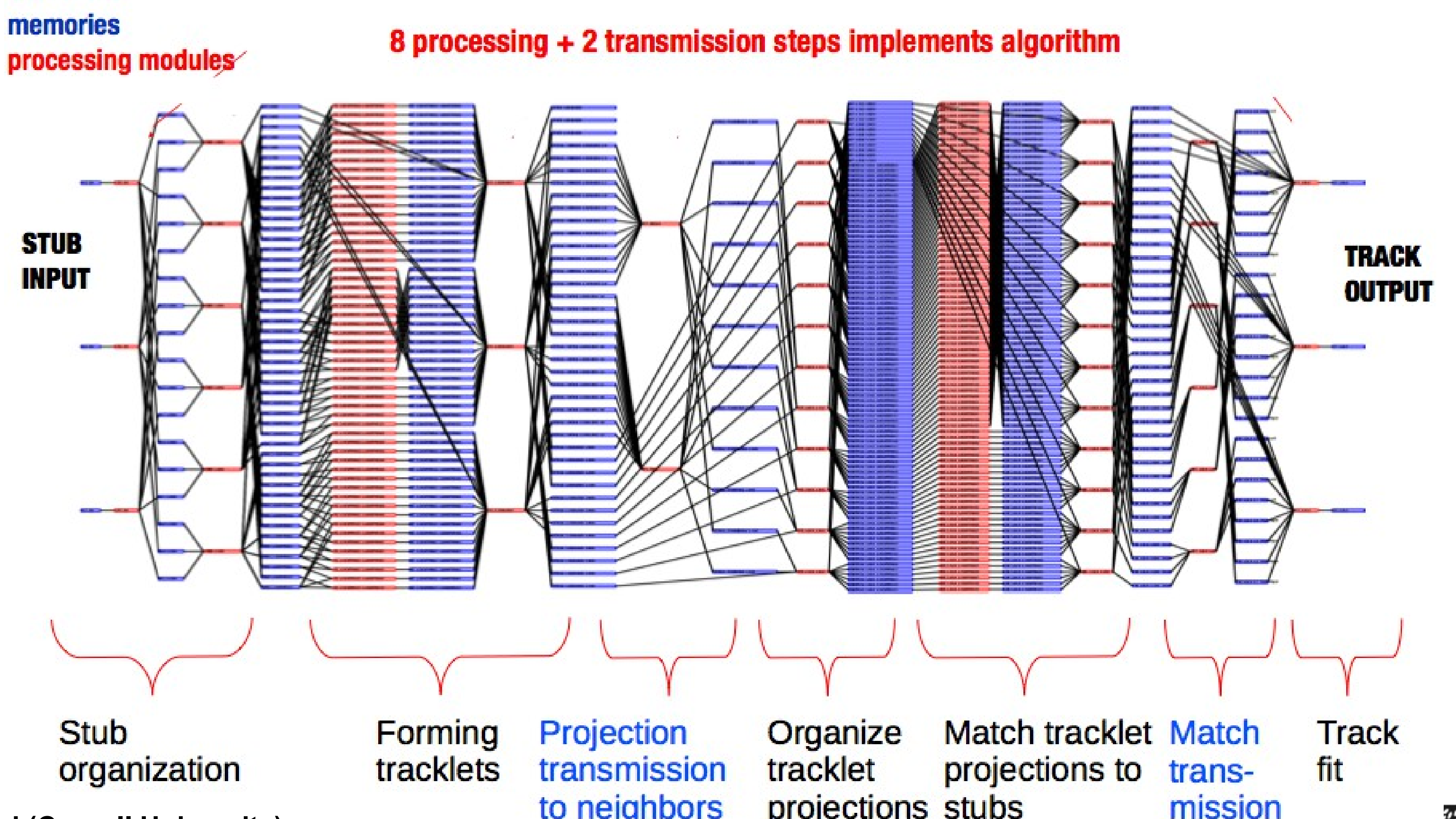


A.Schöning, Heidelberg University

Tracklets

- **Stubs:** Correlated pairs of clusters, consistent with ≥ 2 GeV track
  - For minbias, rejects ~95% of tracks
  - Stubs form input to track finding





Part II

ML in Tracking

- Hopfield network
- Tracking in dense environment in atlas
- Activity segmentation in TPC test beam
- Activity segmentation for neutrino flavor ID
- Muon decay point identifier
- TPC space-charge distortion
- Track selection in LHCb
- Tracker hit cluster selection in LHCb
- Non-parametric functional regression
- Seeded track finding in simplified model
- Track parameters estimation using LSTM
- Pattern recognition with sequence-2-sequence

Hopfield Network

$$E = -\frac{1}{2}\left(\sum_{i,j} w_{ij} S_i S_j - 2\sum_i \theta_i S_i\right).$$

- Not a neural network per say
- Fully-connected graph
- Connections pruned based on an enery minimisation model

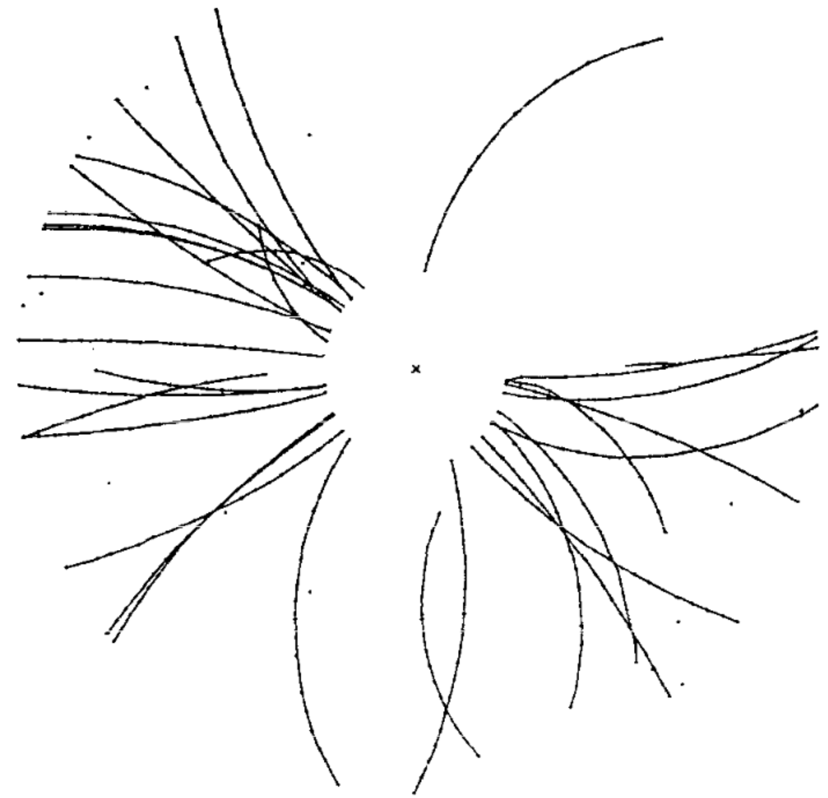
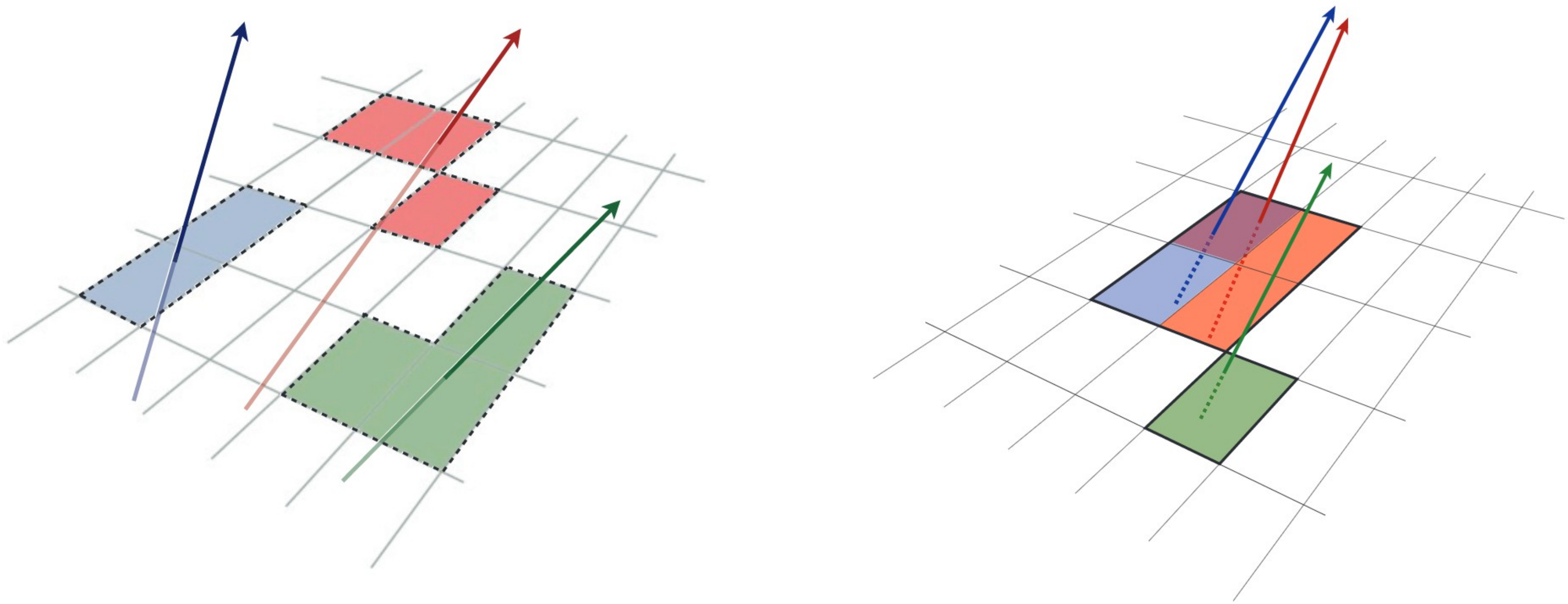


Fig. 4. Tracks in the ALEPH TPC reconstructed with a Hopfield net [13].

https://link.springer.com/chapter/10.1007/3-540-61510-5_1

Tracking in Dense Environment

Tracking In Dense Environment



(a) Single-particle pixel clusters

(b) Merged pixel cluster

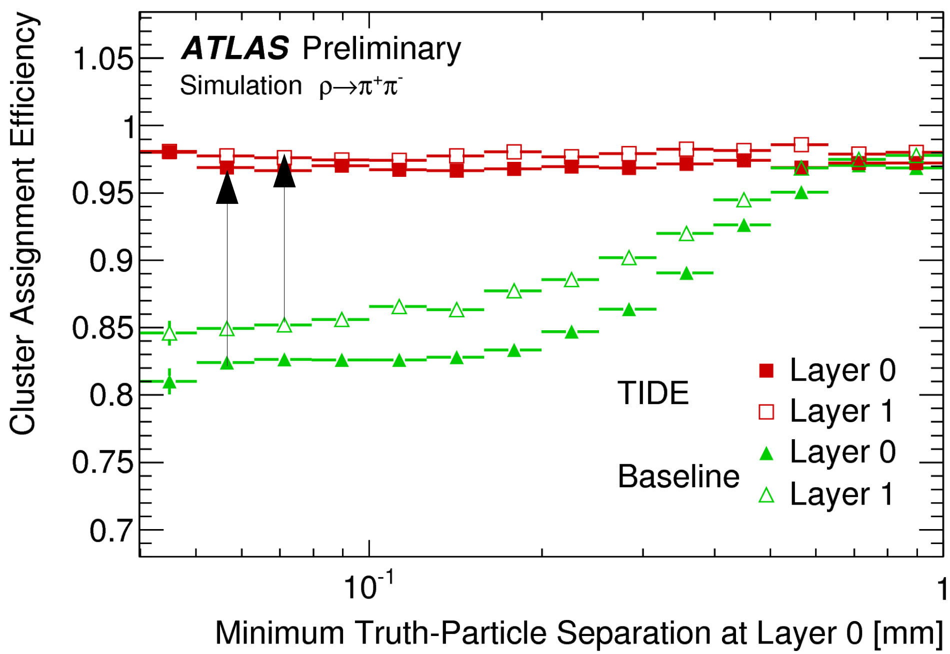
Converging tracks are likely in boosted jets and jets dense of charged particles.
Degraded performance

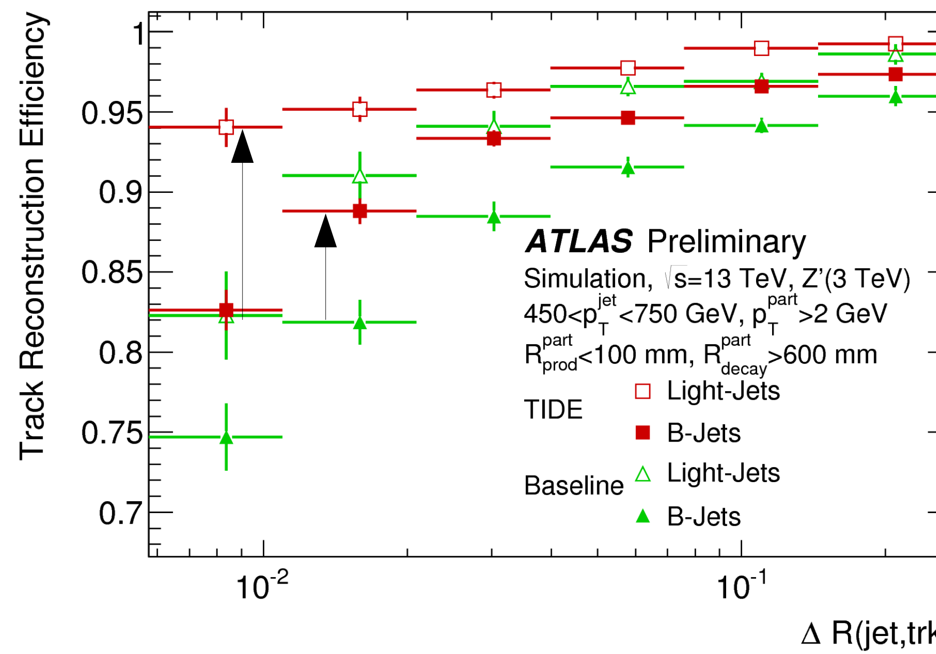
https://arxiv.org/abs/1704.07983

Cluster Splitting

Feed forward NN in three stages
- Determines the category 1-track, 2-tracks, 3-tracks
- Determines the n-crossing positions regression
- Determines the uncertainties as a multi-bin categorization

2 hidden layers fully connected NN with batch norm





ATL-PHYS-PUB-2015-006

Non-parametric Track Finding

Non Parametric Functional Kernels

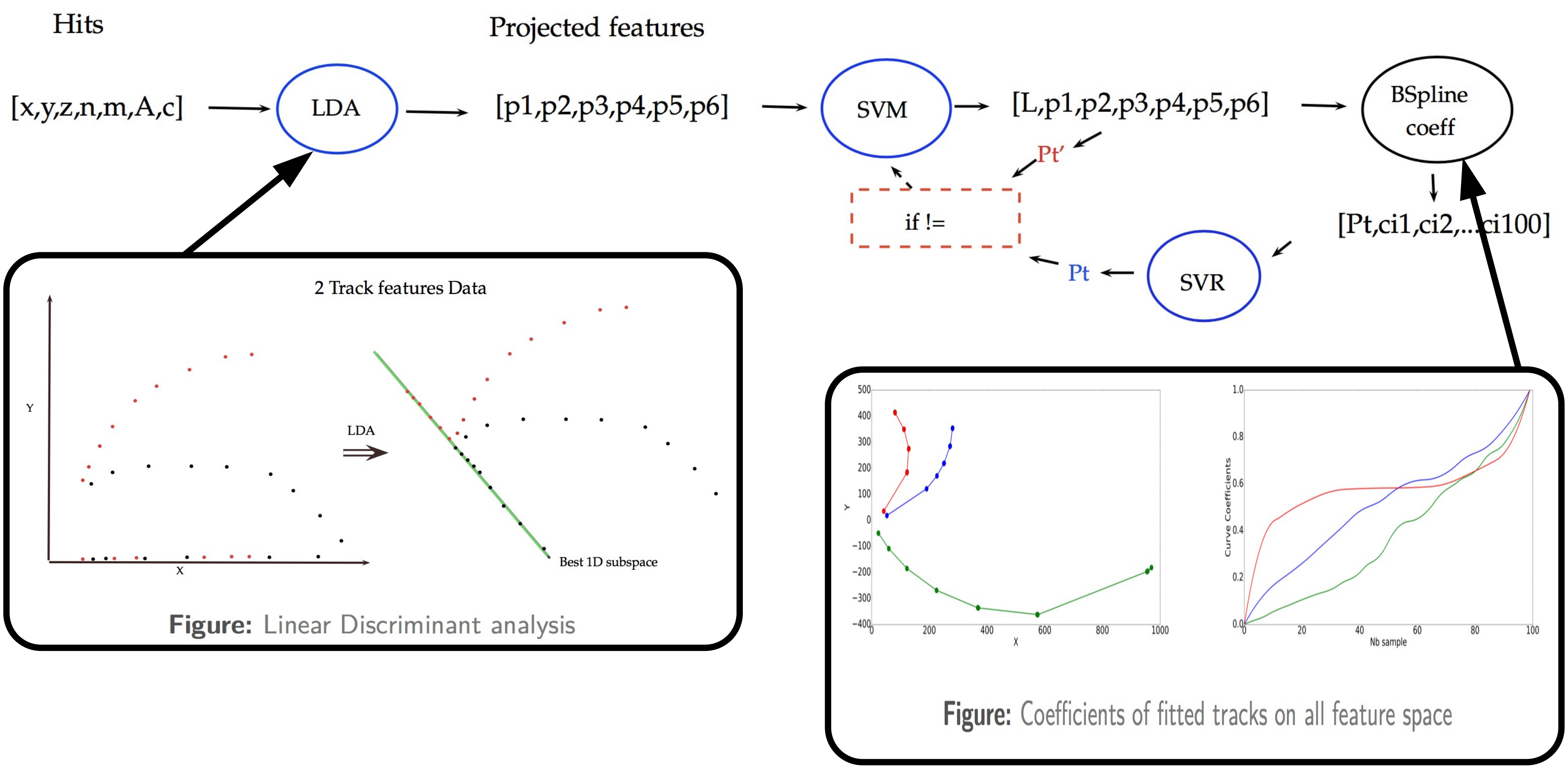


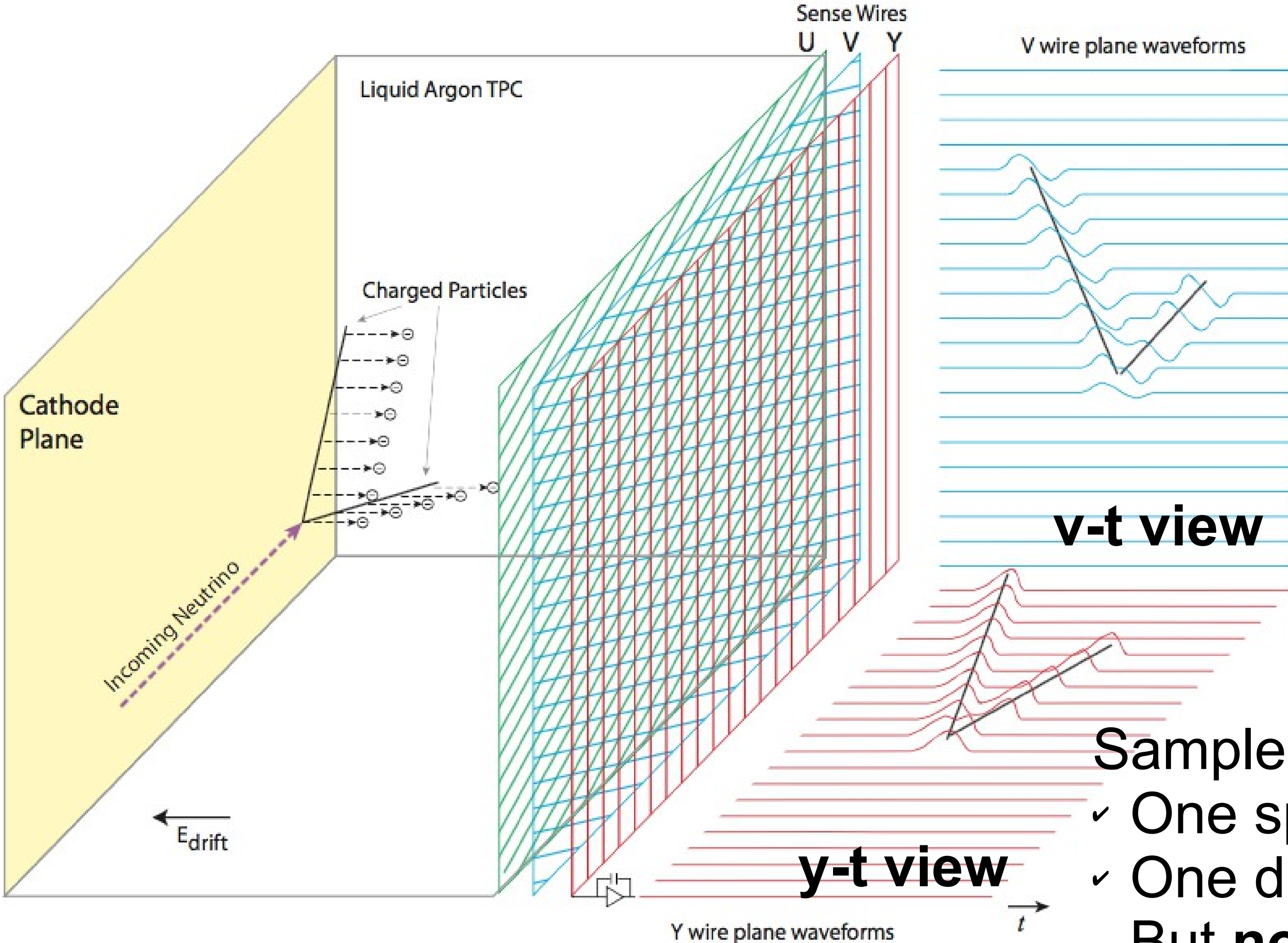
Figure: Linear Discriminant analysis

Figure: Coefficients of fitted tracks on all feature space

Work in progress : S. Amrouche. T. Golling, A. Salzburger, J. Pilz
https://indico.cern.ch/event/577003/contributions/2444883/

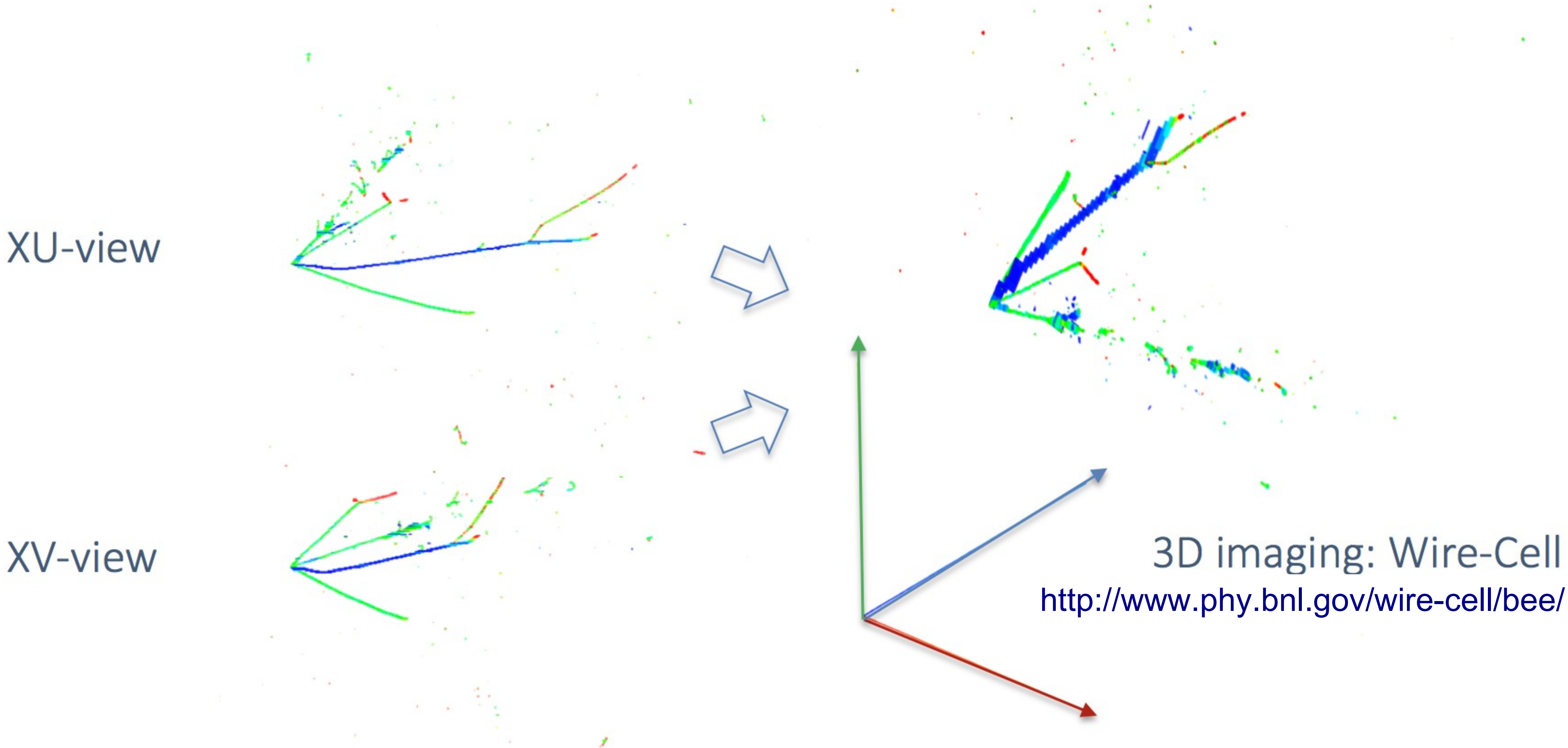
Tracking-like Neural Net Application in Time Projection Chambers

Time Projection Chamber

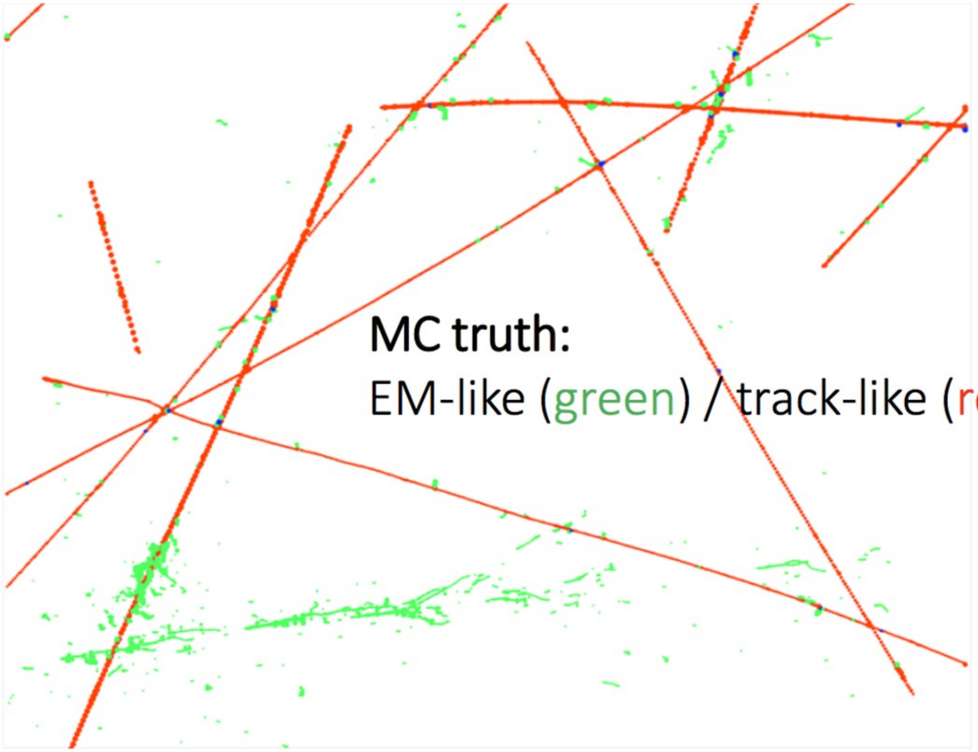


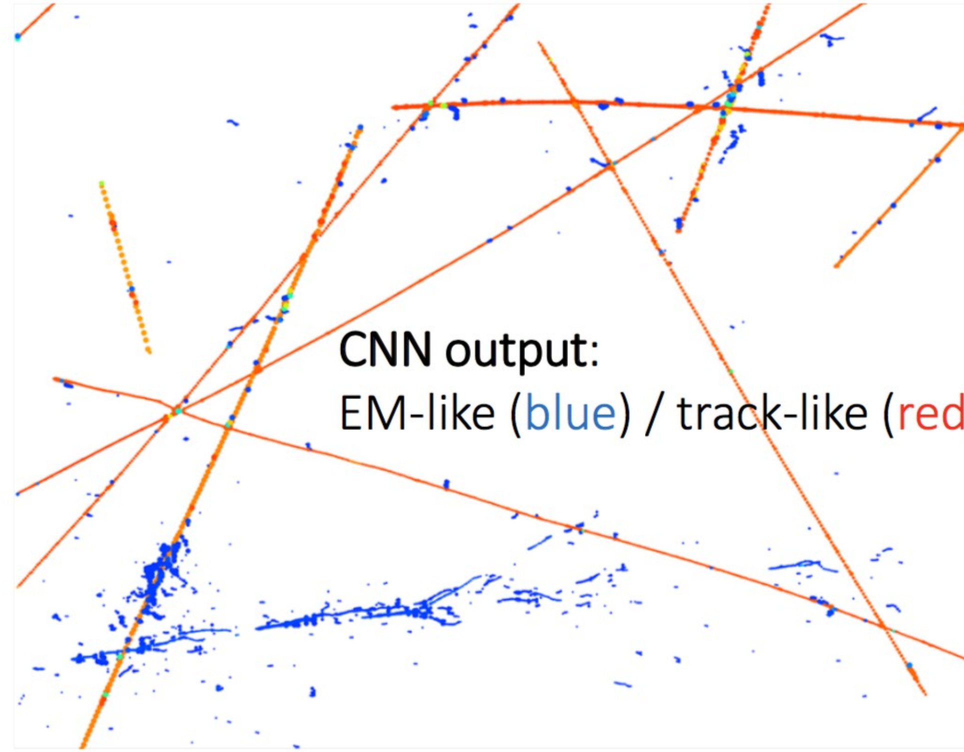
Sample ≡ 2x2D images
- ✓ One space dimension
- ✓ One dimension **is time.** But **not the time** of the particle decay

TPC 2x2D to 3D



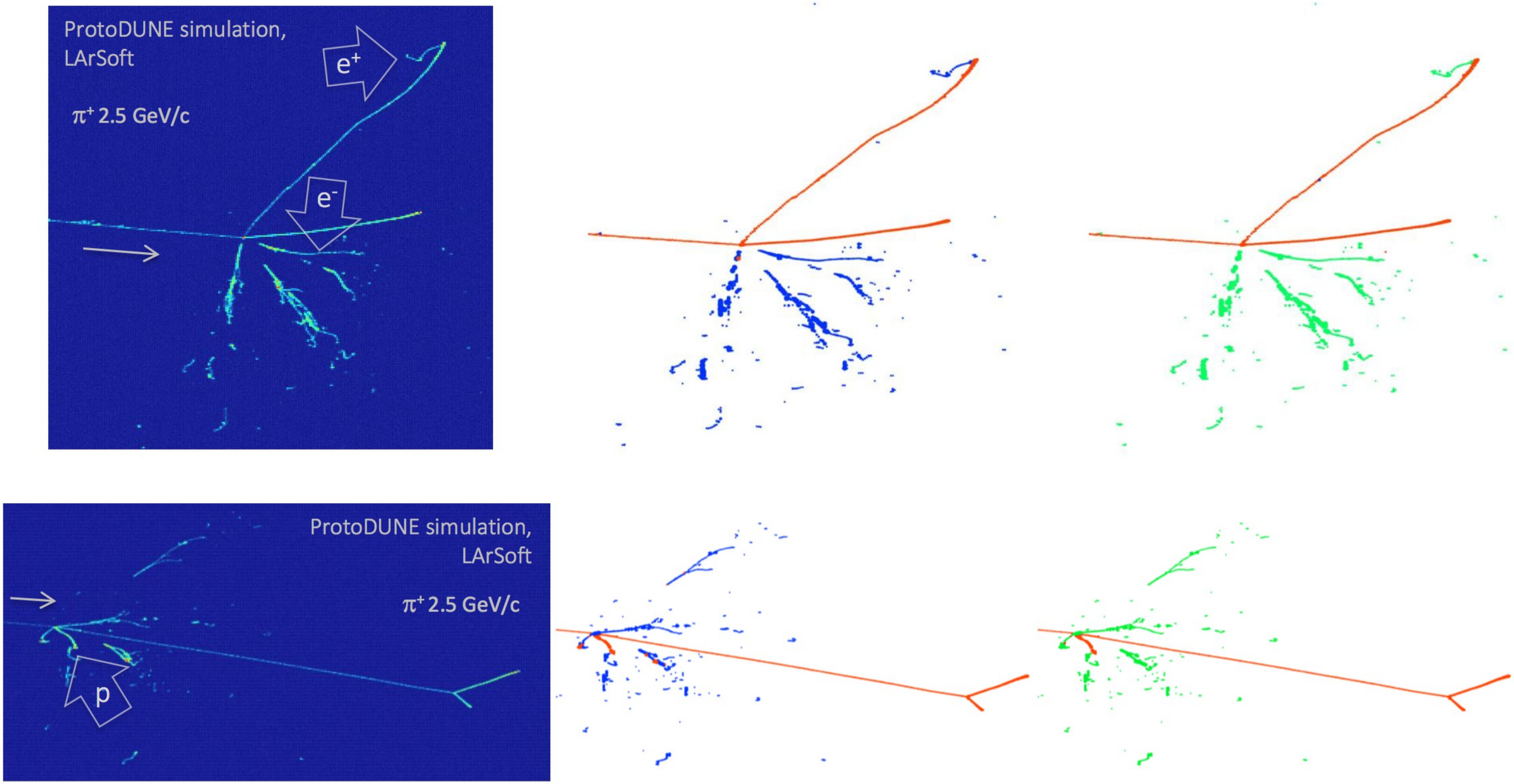
TPC Activity Segmentation





- Challenge to code explicitly
- Almost text-book example of de-noising AE
- Achieved with CNN

Flavor Segmentation

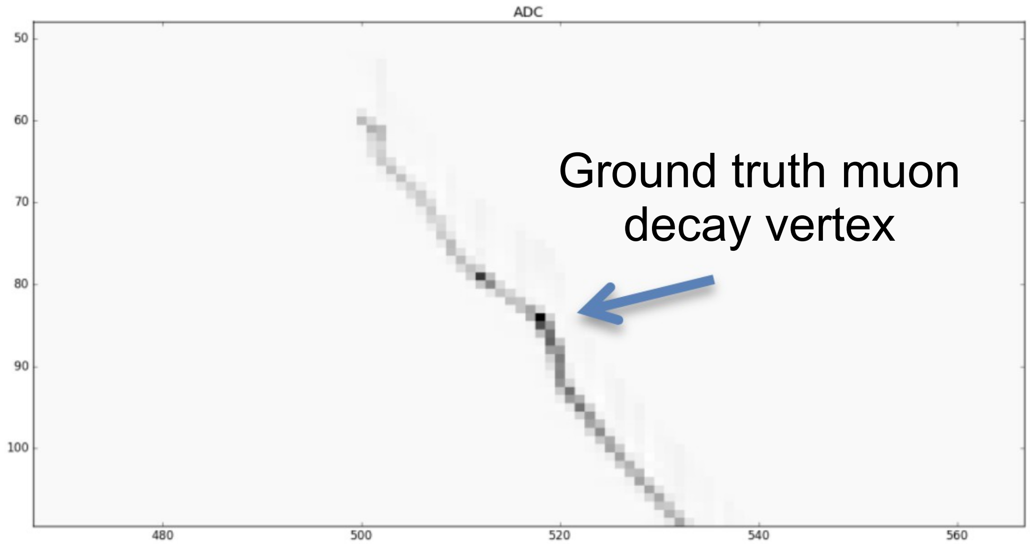


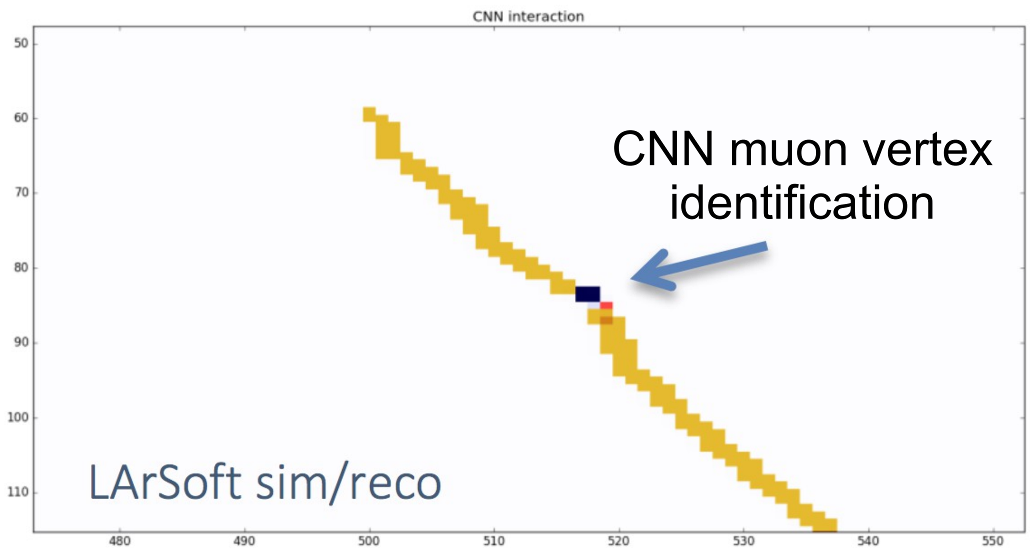
input: 2D ADC

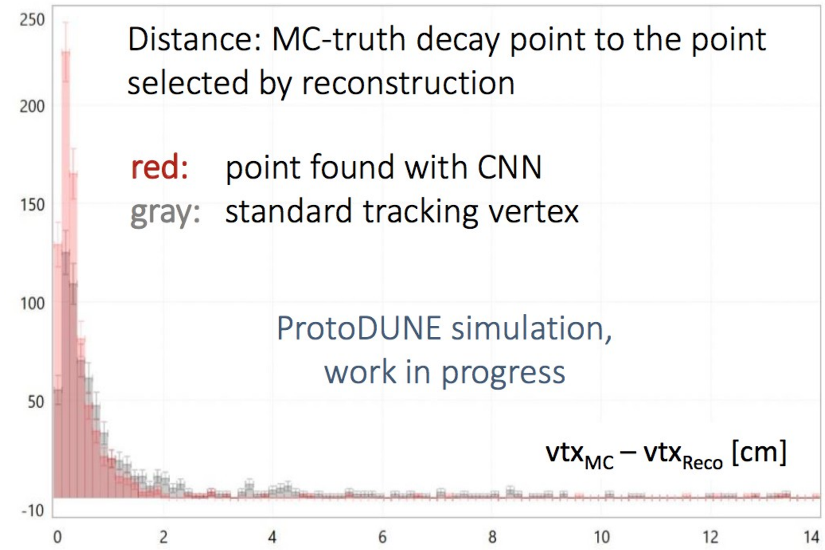
CNN output:
EM-like (blue) / track-like (red)

MC truth:
EM-like (green) / track-like (red)

Decay Point Identifier



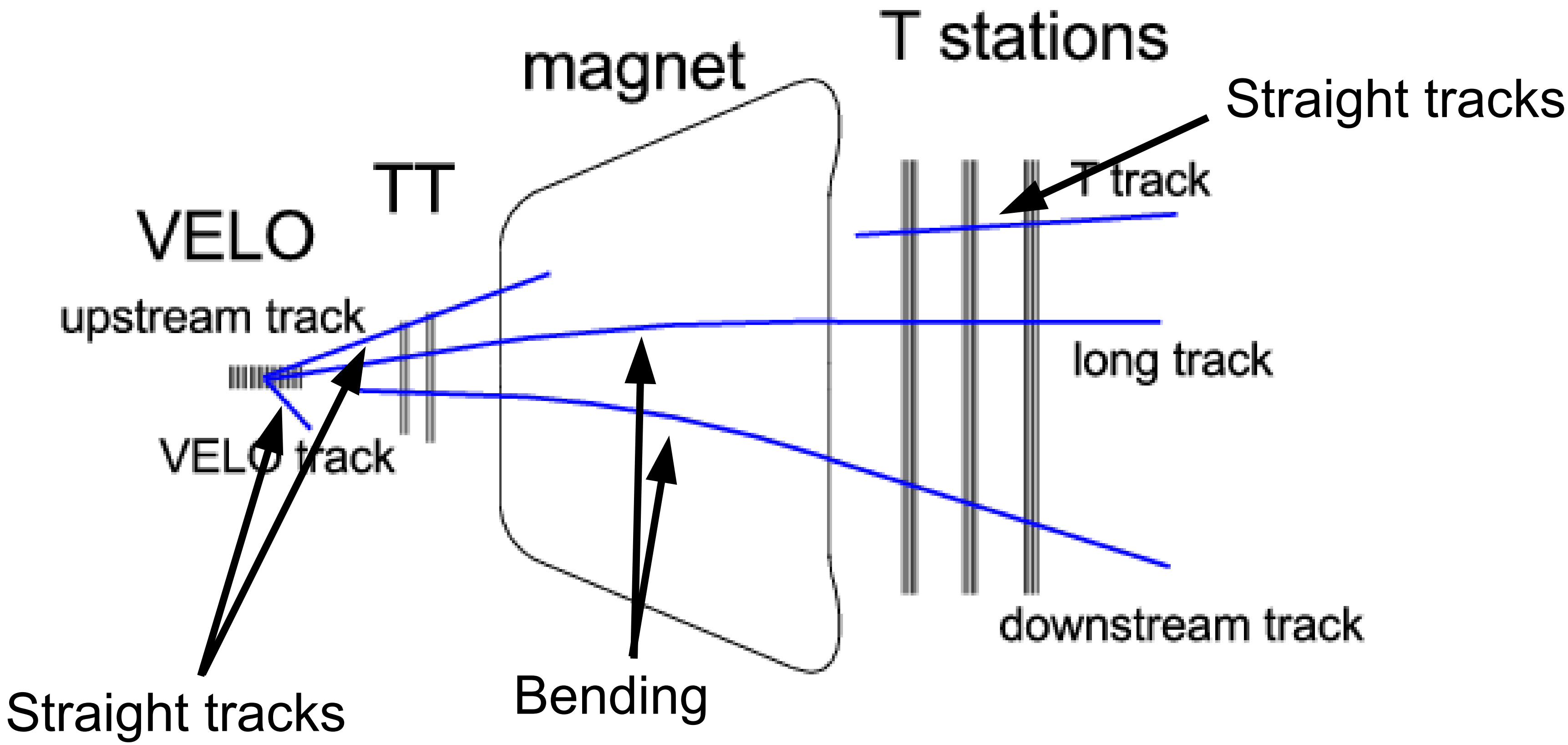




- CNN slightly outperform the classical approach
- Much less complication in programming the vertex finding

Online Tracking in LHCb

Tracking in LHCb



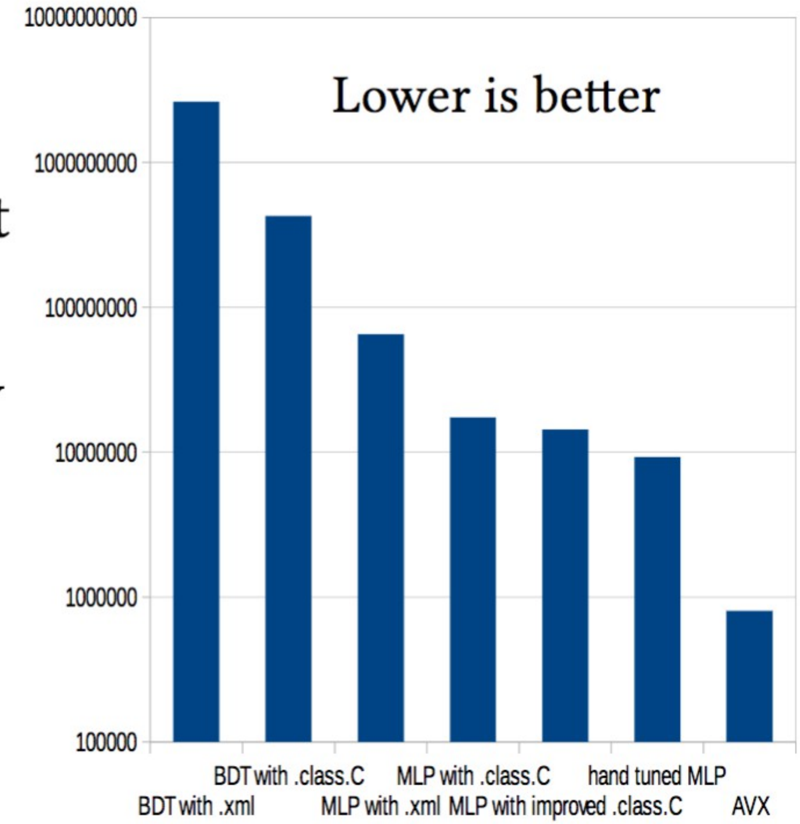
Computational bottleneck is in the final track fit involving a kalman filtering procedure

Forward tracking : velo tracks extrapolated to T-stations

Downstream tracking : tracks in the T-stations extrapolated back to TT

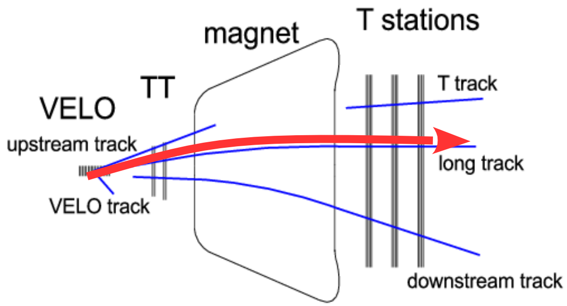
Fast Neural Net

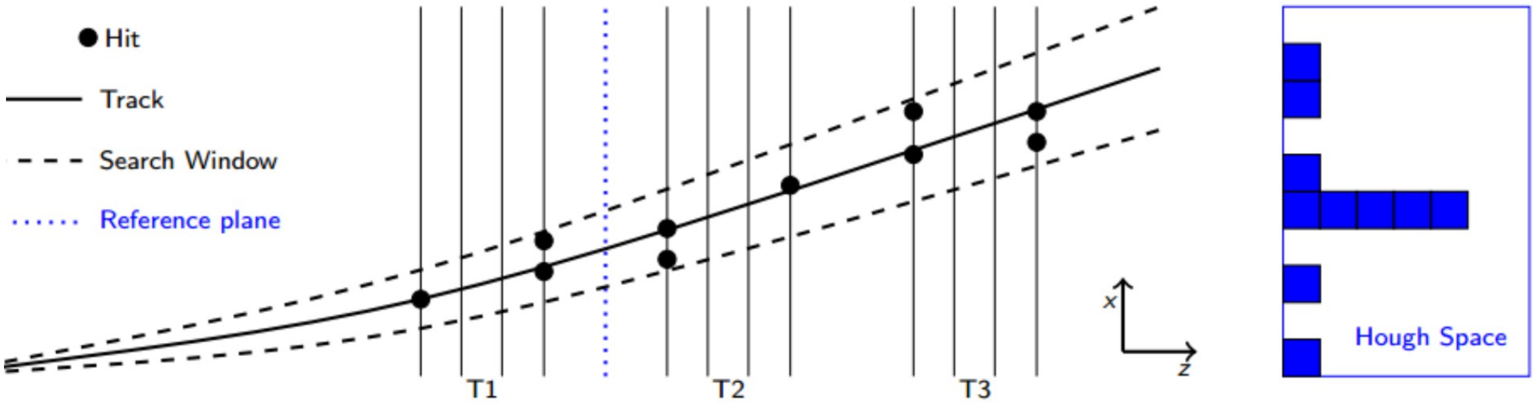
- Ghost probability must be computed fast (numbers for TMVA)
  - Neural network faster than BDT (40x)
  - Compile network instead of loading at runtime (4x)
  - Tune auto-generated network code by hand (2x)
  - Faster network activation function (uncharted, 4x)
- Drop support for >5yr old CPUs (10x)
  → Make auto-generated code better?



S. Stahl at DS@HEP 2016
https://indico.hep.caltech.edu/indico/conferenceDisplay.py?confId=102

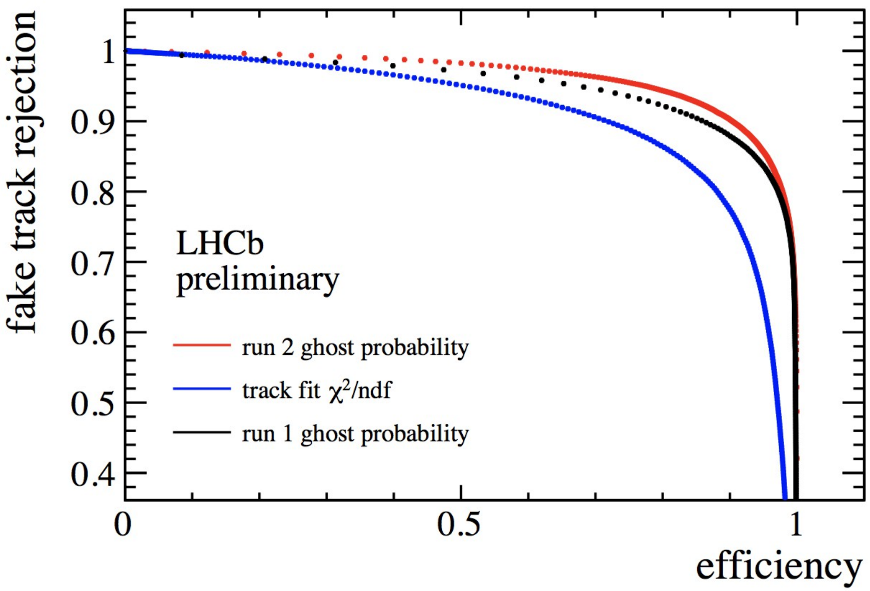
Forward Tracking



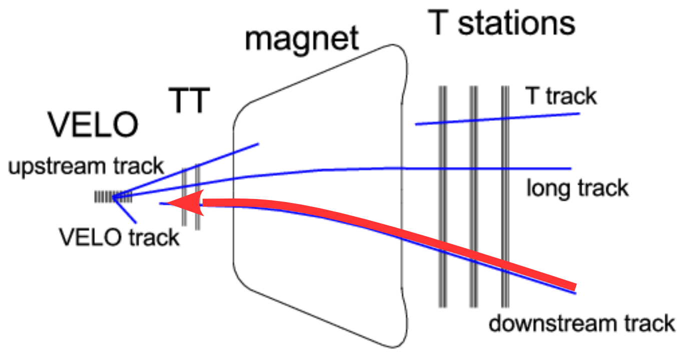


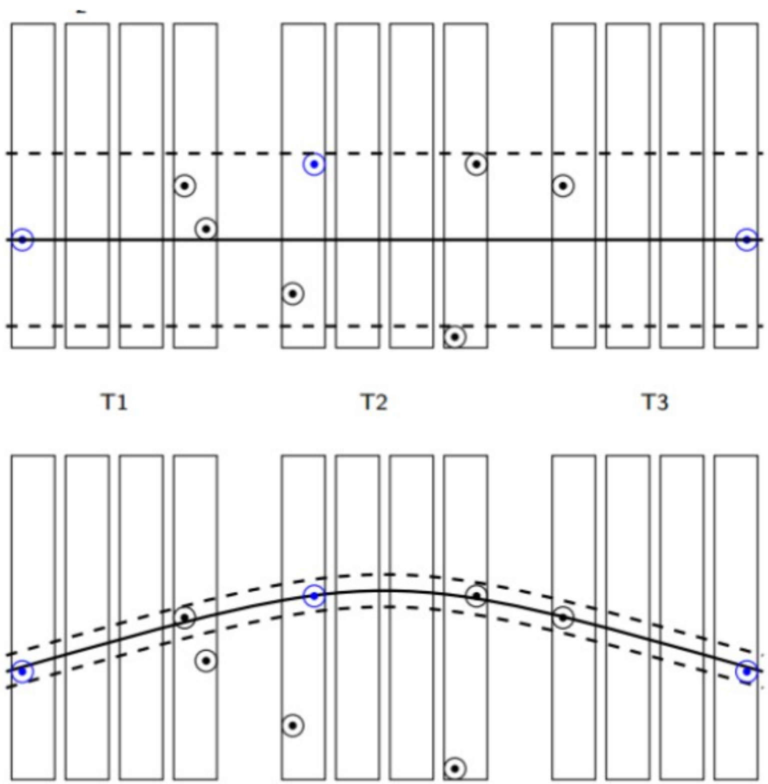
Forward tracking : long track reconstructions

- NN classifier to distinguish good and bad clusters in the hough space
- NN classifier to select good from bad tracks



Downstream Tracking





Backward track reconstructions

- NN classifier to distinguish good and bad T-seed (Use of the bonsai BDT https://arxiv.org/abs/1210.6861)
- NN classifier to select good from bad final tracks

Seeded Track Candidate Making

https://heptrkx.github.io/

HEP.TrkX Project

https://heptrkx.github.io/

- Pilot project funded by DOE ASCR and COMP HEP
- Part of HEP CCE
- Mission
  - Explore deep learning techniques for track formation

- People
  - **LBL** : Paolo Calafiura, Steve Farrell, Mayur Mudigonda, Prabhat
  - **FNAL** : Giuseppe Cerati, Lindsey Gray, Jim Kowalkowski, Panagiotis Spentzouris, Aristeidis Tsaris
  - **Caltech** : Dustin Anderson, Josh Bendavid, Pietro Perona, Maria Spiropulu, Jean-Roch Vlimant, Stephan Zheng

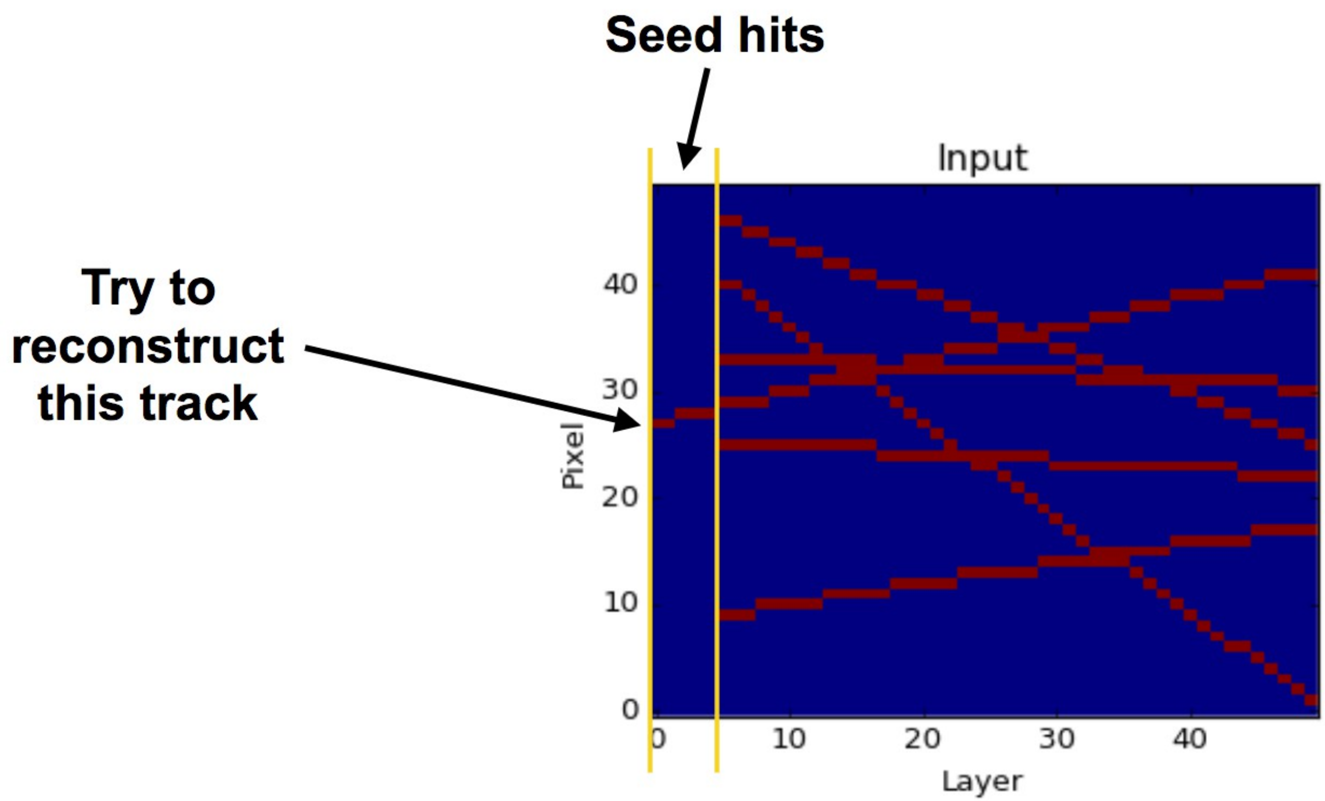
Scene Labeling

Farabet et al. ICML 2012, PAMI 2013

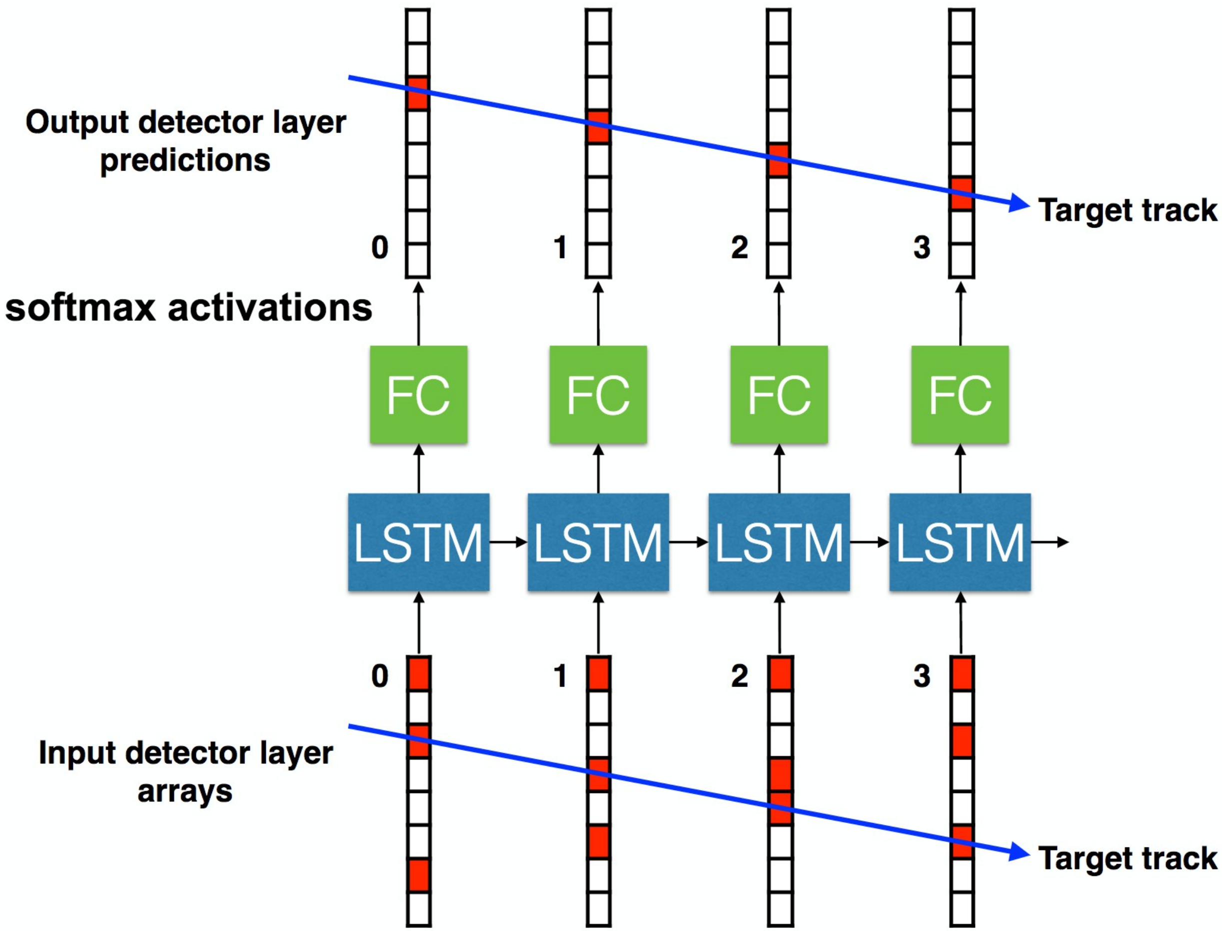
- Assign hits to track candidates

Seeded Pattern Prediction

- Hits on first 3 layers are used as seed
- Predict the position of the rest of the hits on all layers

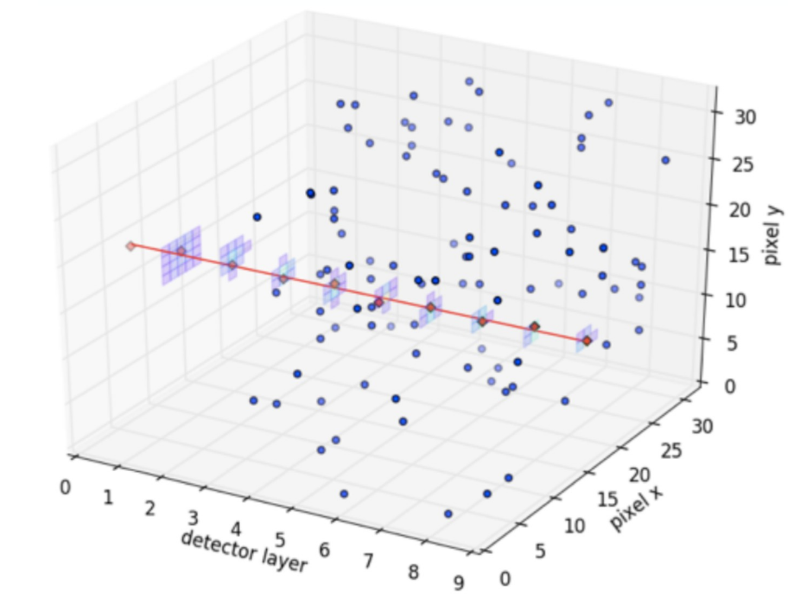


LSTM ≡ Kalman Filter

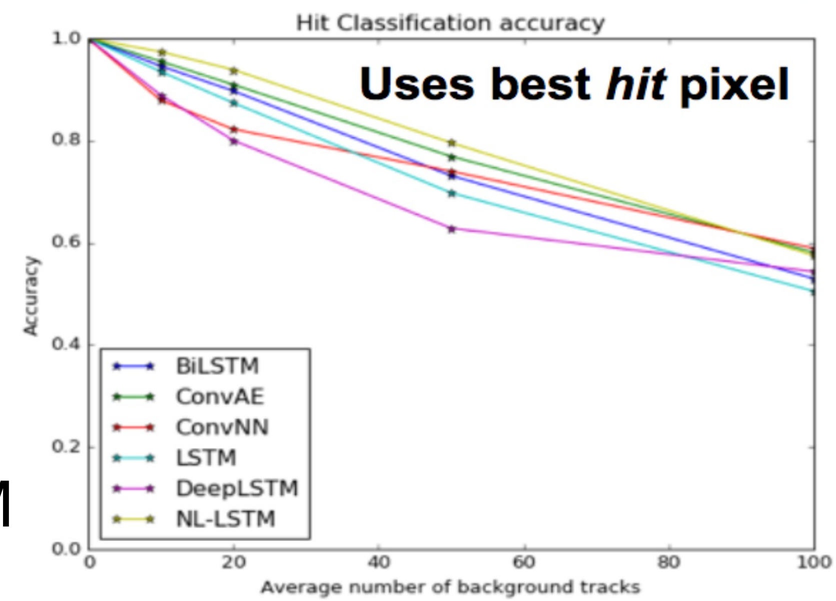


Seeded Pattern Recognition Insights

- For a simplified track models, predicting the track pattern from the seed works
  - In 2D and 3D
  - With some level of noise
  - With other tracks present
  - On layers with increasing number of pixels

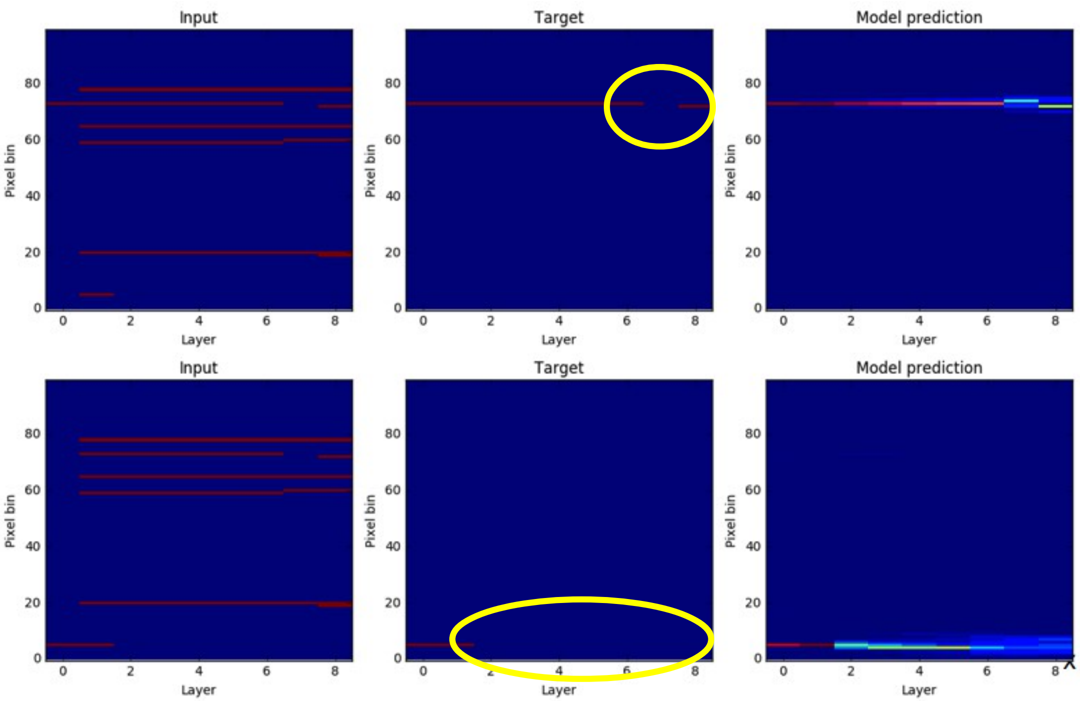


- Several other architectures tried
  - Convolutional neural nets (no LSTM)
  - Convolutional auto-encoder
  - Bi-directional LSTM
  - Prediction on next layer with LSTM

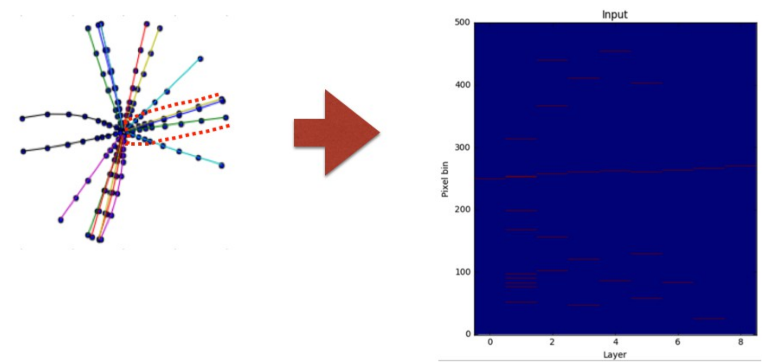


Tracking RAMP at CtD

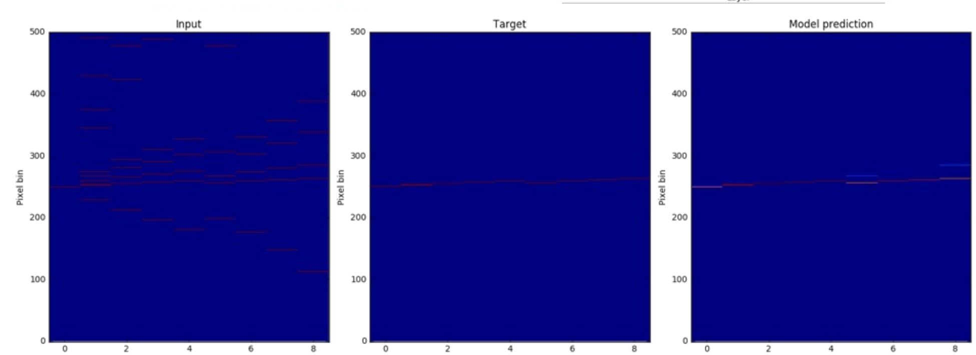
S. Farrell : Best solution in the Machine Learning category
https://indico.cern.ch/event/577003/contributions/2509988/



- Increased granularity in “road”
- LSTM for hit assignment
- 95% efficiency



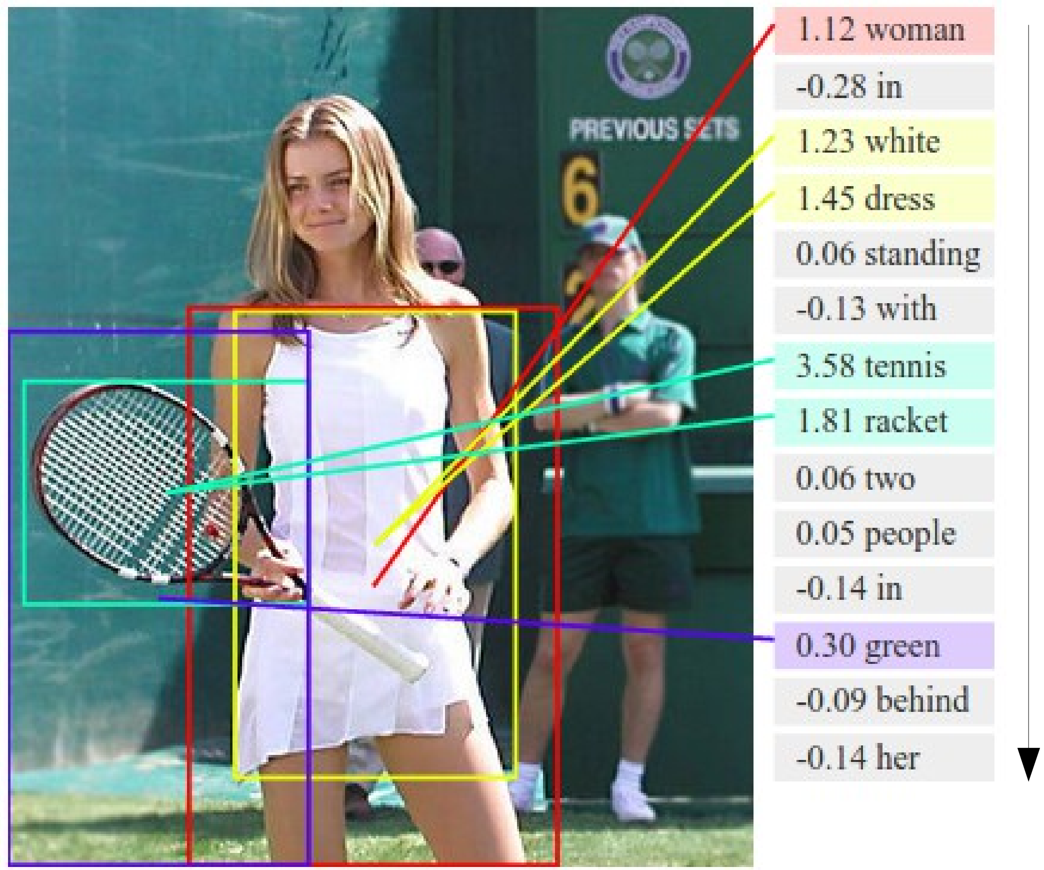
- Down-sampling layer to 100 bins
- LSTM for hit assignment
- 92% efficiency
- Robust to holes and missing hits



Track Parameters Measurement

https://heptrkx.github.io/

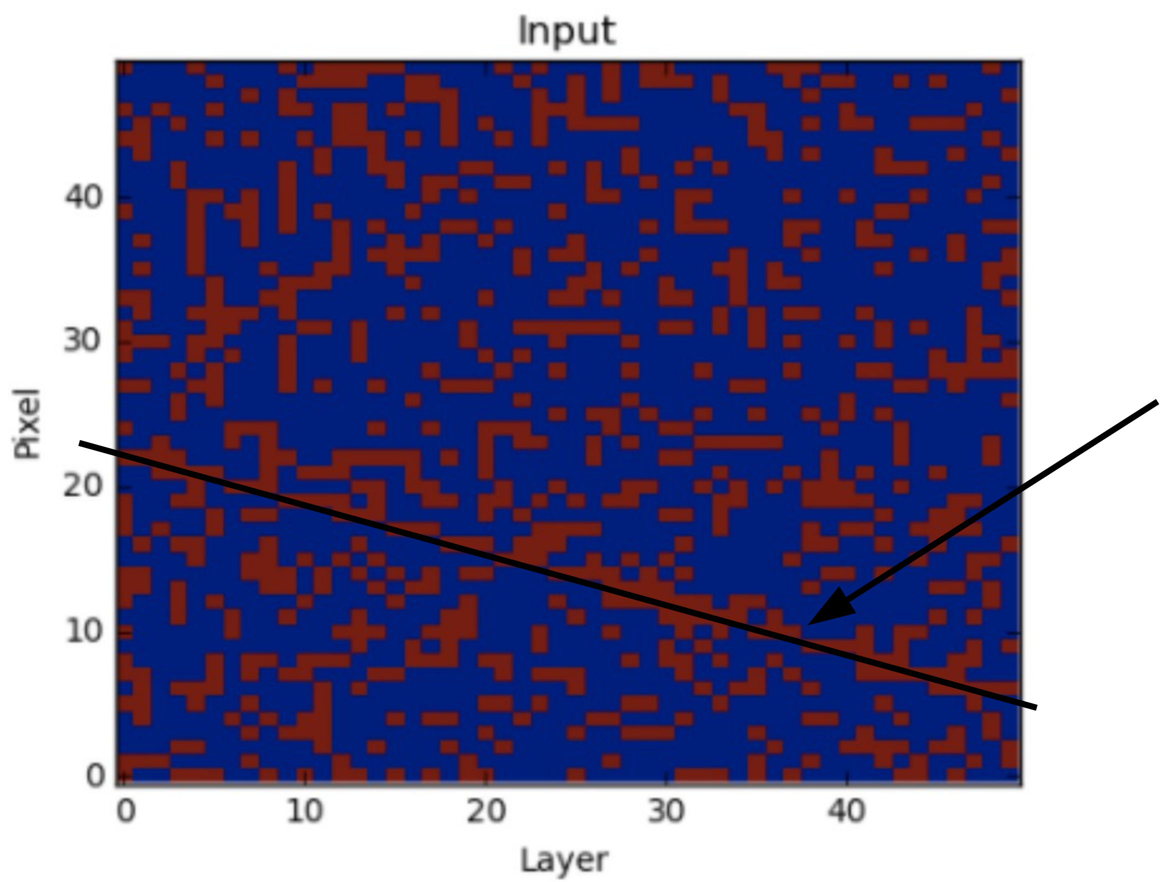
Scene Captioning



Karpathy, Fei-Fei, CVPR 2015

➔ Compose tracks explanation from image

Track Parameter Estimation

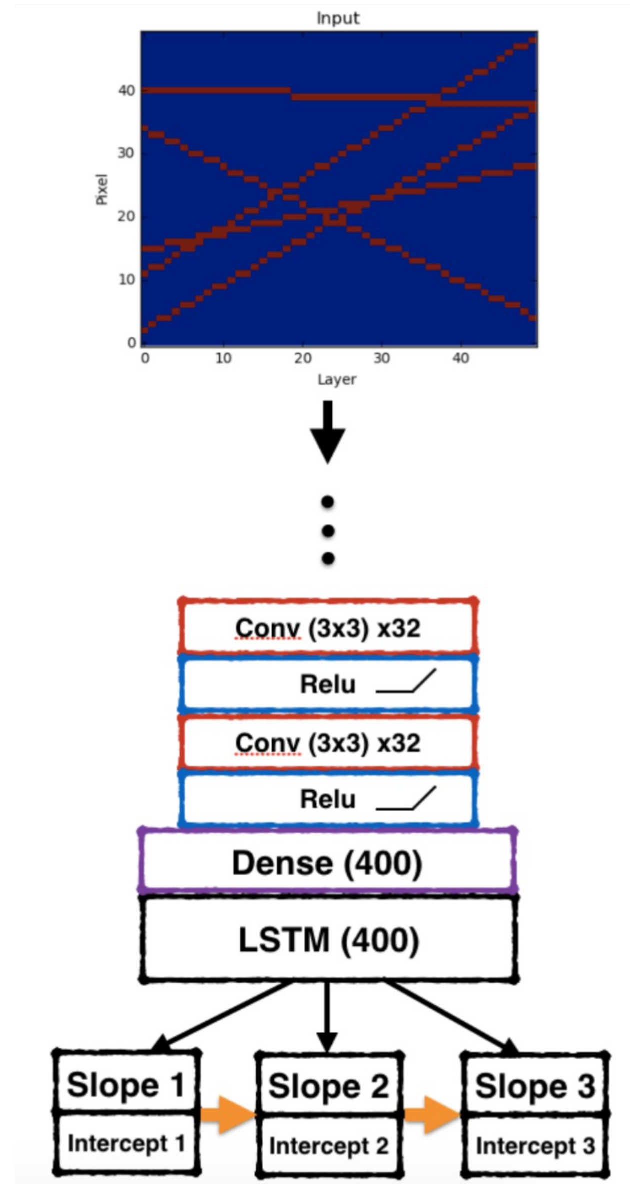


Try to predict the slope and intersect of this track

Multi-Track Prediction with LSTM

- Hit pattern from multiple track processed through convolutional layers

- LSTM Cell runs for as many tracks the model can predict.

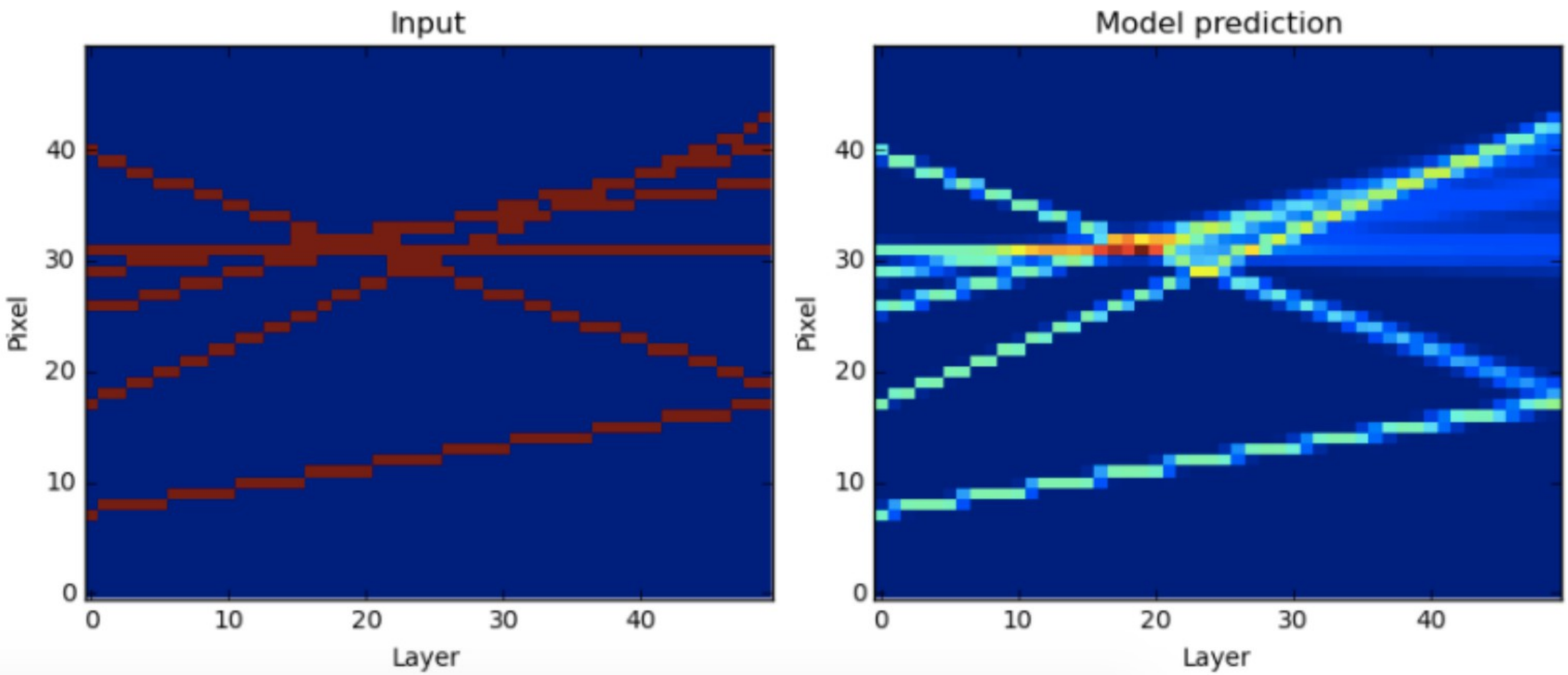


Prediction Track Covariance

Model is modified to predict a covariance matrix for which there is no ground truth, but is used with the modified loss function

$$L(\boldsymbol{x}, \boldsymbol{y}) = \log |\boldsymbol{\Sigma}| + (\boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}))^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}))$$

Track Parameters Uncertainty



Representation of track slope, intersect and respective uncertainties

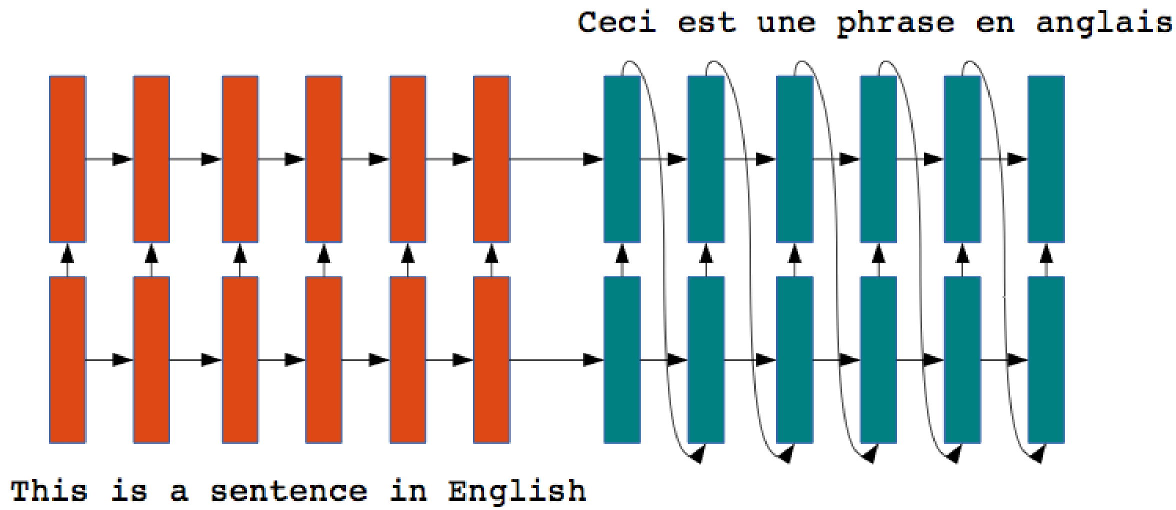
Pattern Recognition / Seeding

https://heptrkx.github.io/

Text Translation

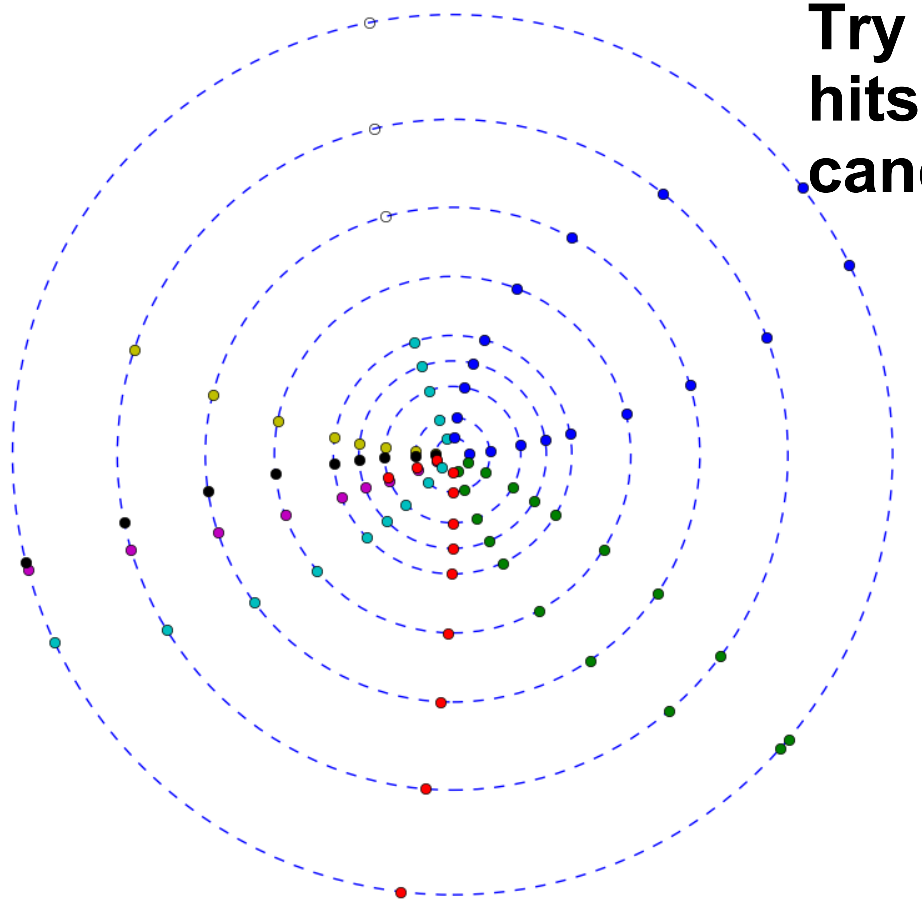
[Sutskever et al. NIPS 2014]

- Multiple layers of very large LSTM recurrent modules
- English sentence is read in and encoded
- French sentence is produced after the end of the English sentence
- Accuracy is very close to state of the art.



→ From sequence of hits on layer to sequence of hits on track

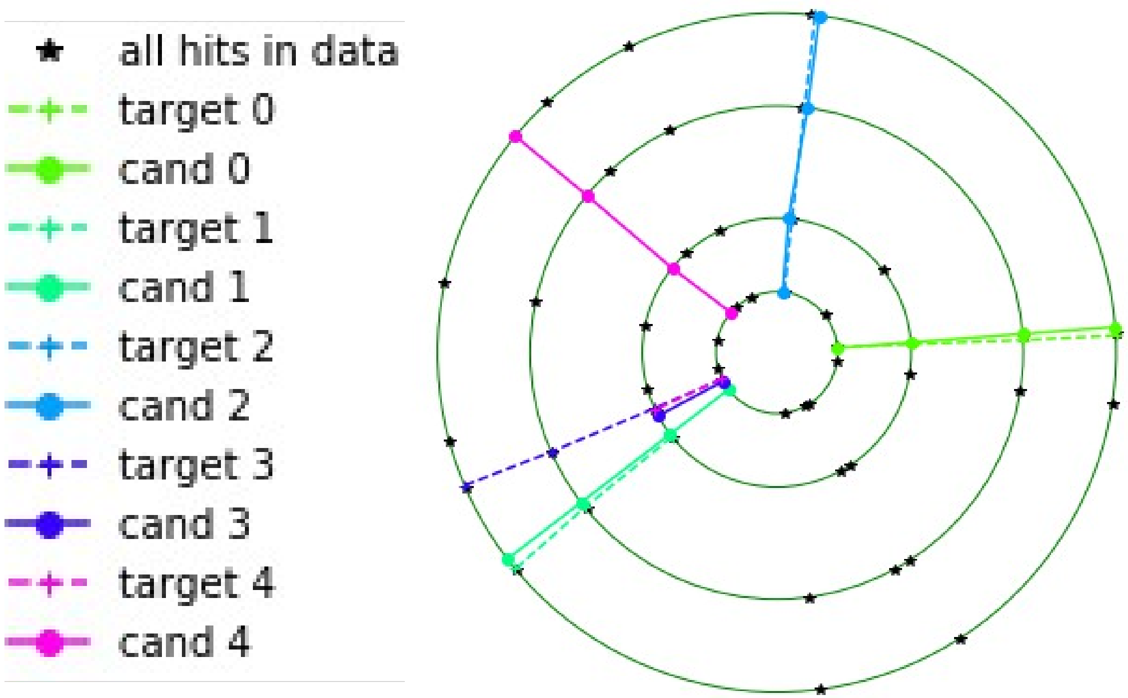
Pattern Recognition



Try to assemble hits into track candidates.

Pattern Recognition with LSTM

- Input sequence of hits per layers (one sequence per layer)
  - One LSTM cell per layer
- Output sequence of hits per candidates
  - Final LSTM runs for as many candidates the model can predict



- Still work in progress
- Restricted to 4 layers (with seeding in mind)
- Work to some extend

Summary

- Charged particle tracking in High Energy Physics is a complex task.

- Machine learning is already applied at several levels to cope with the task complexity.
  - Software : JetNet, TMVA, keras, ...
  - Methods : MLP, BDT, bBDT

- More computationally efficient ways to perform tracking would be welcome.

- Active R&D in tracking (using CNN, RNN, LSTM, ...) can benefit from input and suggestions. Get in touch and get involved !

Backup

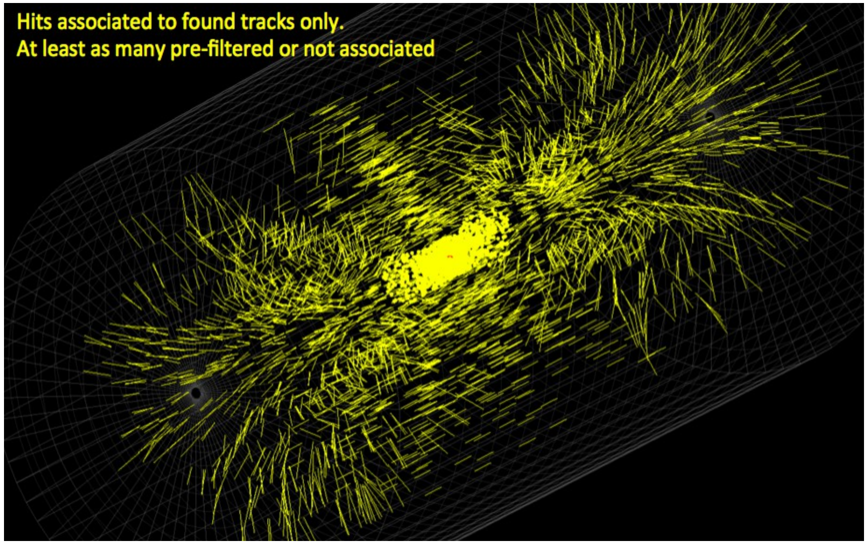
HEP.TrkX Project

https://heptrkx.github.io/

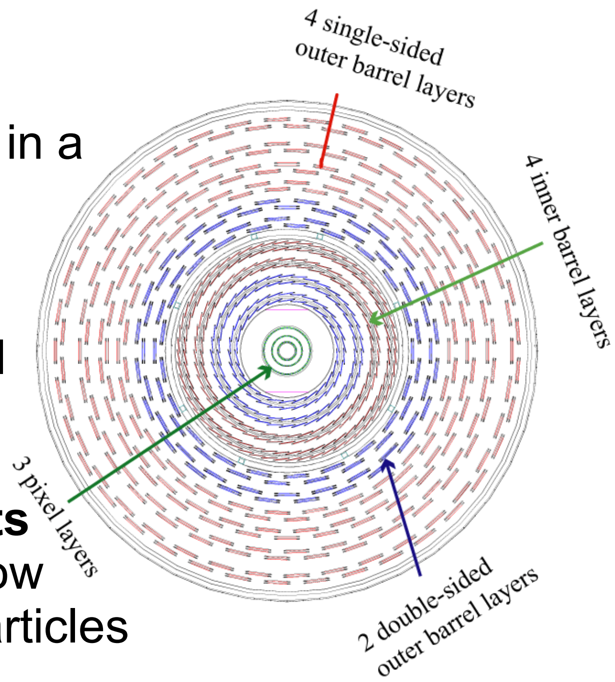
- Pilot project funded by DOE ASCR and COMP HEP
- Part of HEP CCE
- Mission
  - Explore deep learning techniques for track formation

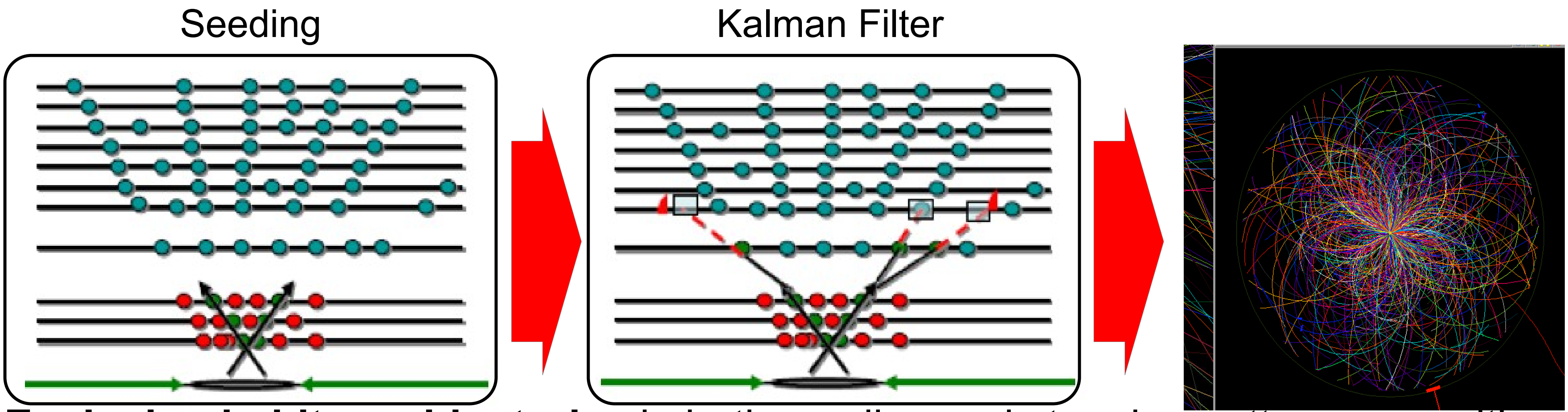
- People
  - **LBL** : Paolo Calafiura, Steve Farrell, Mayur Mudigonda, Prabhat
  - **FNAL** : Giuseppe Cerati, Lindsey Gray, Jim Kowalkowski, Panagiotis Spentzouris, Aristeidis Tsaris
  - **Caltech** : Dustin Anderson, Josh Bendavid, Pietro Perona, Maria Spiropulu, Jean-Roch Vlimant, Stephan Zheng

In a Nutshell



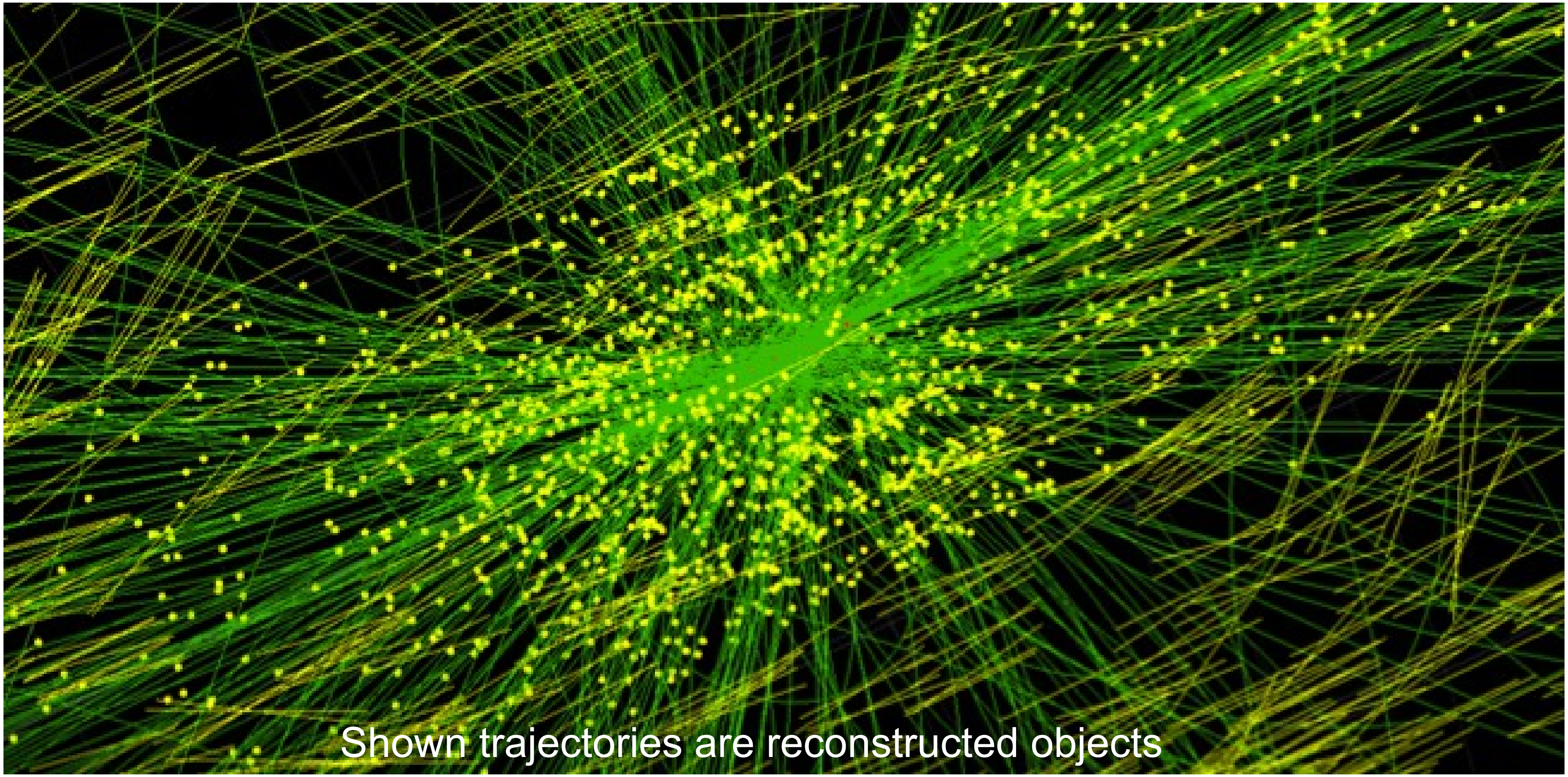
- Particle trajectory bended in a solenoid magnetic field
- Curvature is a proxy to momentum
- Particle ionize silicon pixel and strip throughout several concentric layers
- **Thousands of sparse hits**
- Lots of hit pollution from low momentum, secondary particles





- **Explosion in hit combinatorics** in both seeding and stepping pattern recognition
- **Highly time consuming task** in extracting physics content from LHC data

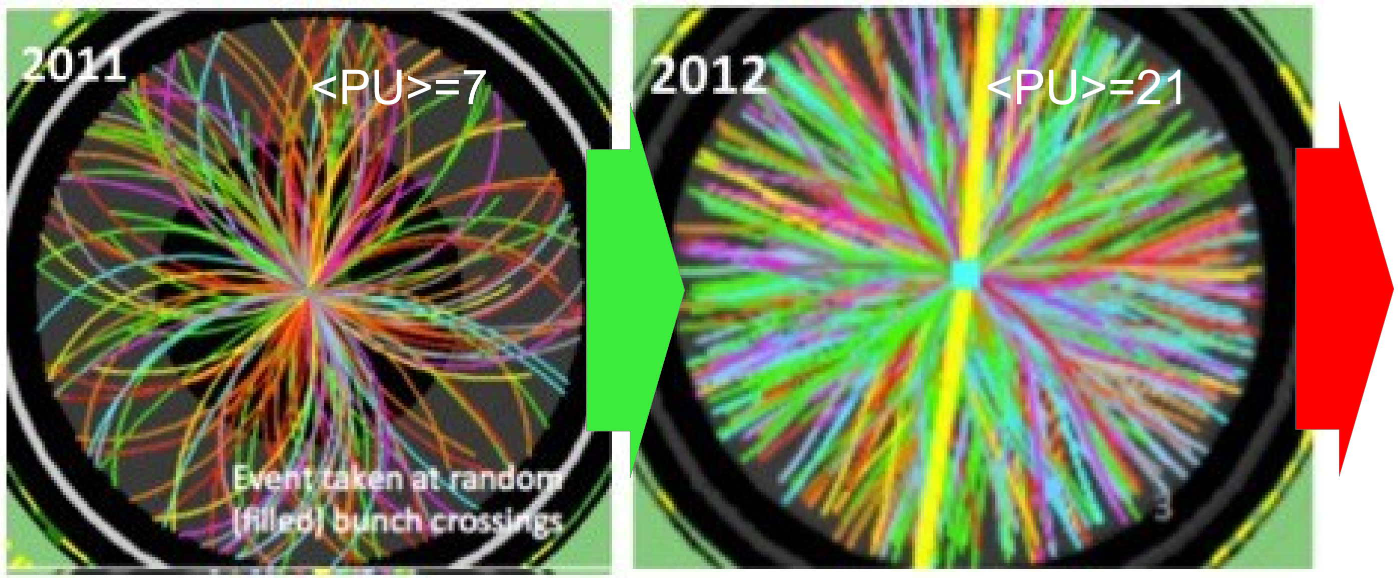
Complexity and Ambiguity



The future is with **x10 more hits**

High Luminosity LHC
The Challenge

HL-LHC Challenge

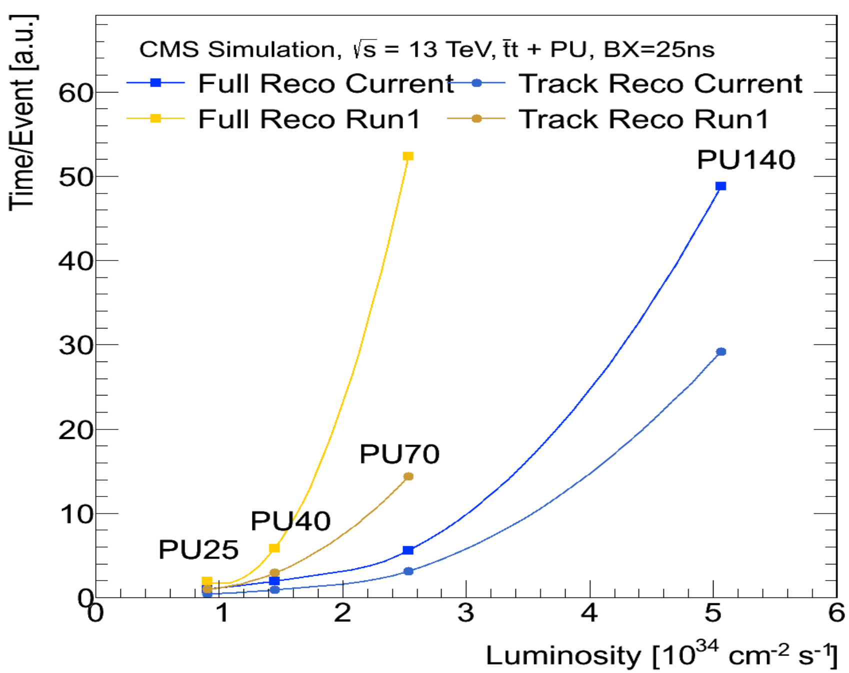


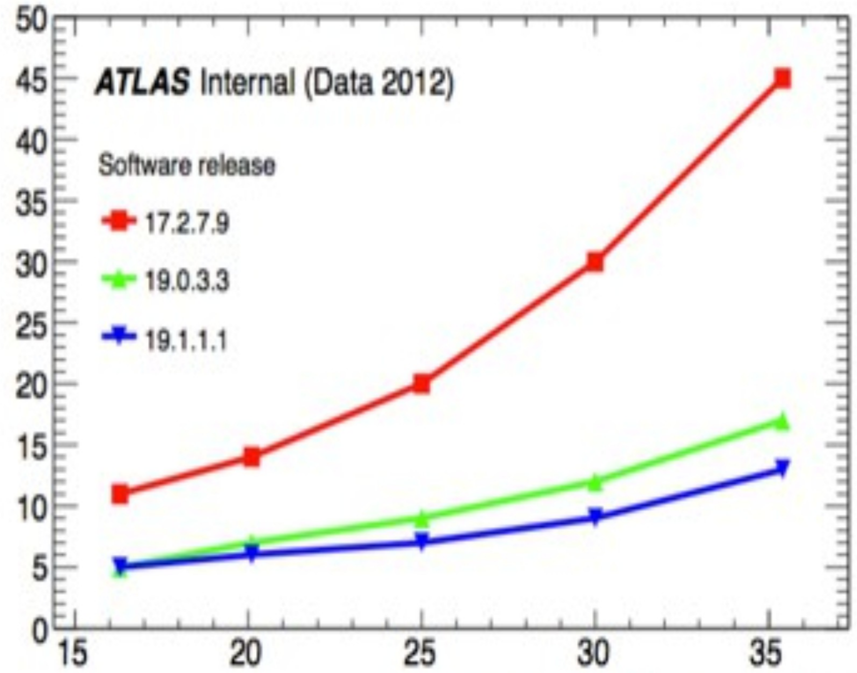
<PU>=140-200
10x more hits
Circa 2025

- CPU time extrapolation into HL-LHC era far **surpasses growth in computing budget**
- **Need for faster algorithms**
- Approximation allowed in the trigger

Cost of Tracking

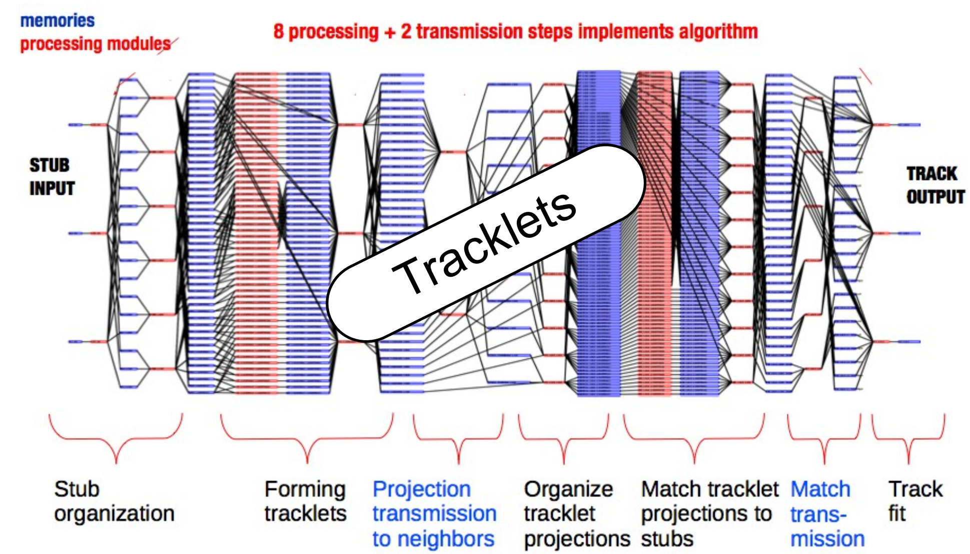
- Charged particle track reconstruction is one of the **most CPU consuming task** in event reconstruction
- **Optimizations (to fit in computational budgets) mostly saturated**
- Large fraction of CPU required in the HLT. **Cannot perform tracking inclusively at CMS and ATLAS.**

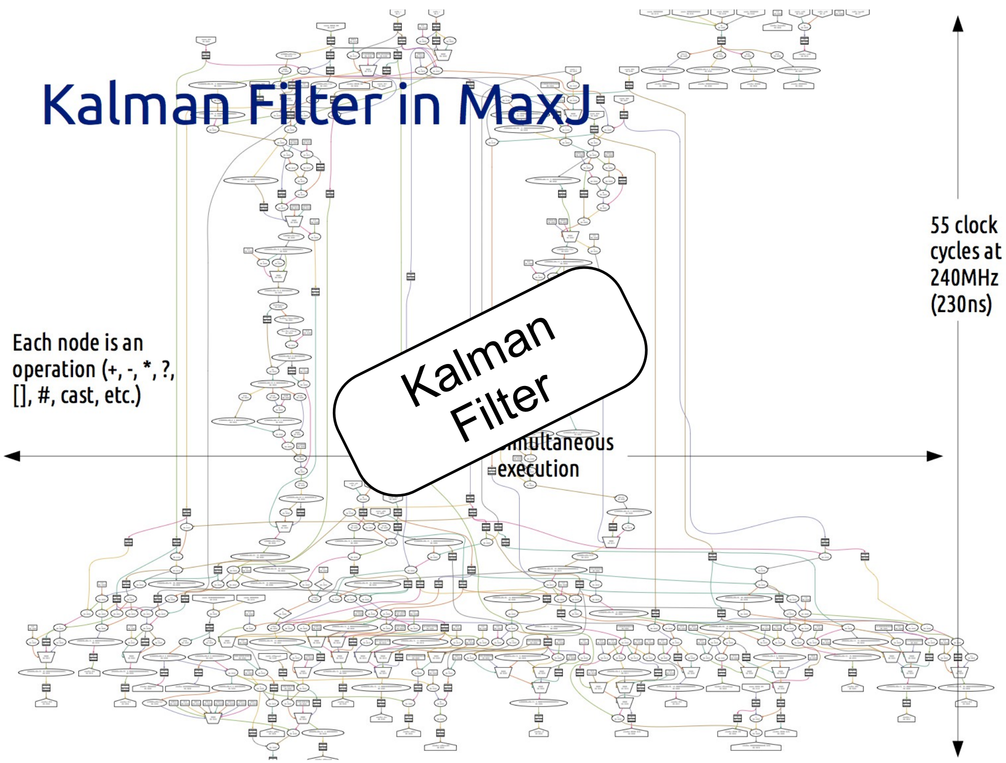


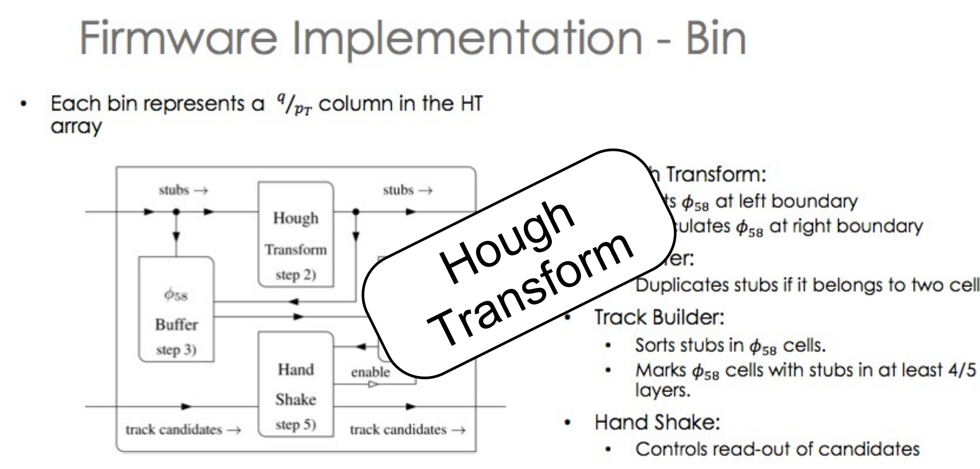


Fast Hardware Tracking

- Track trigger implementation for Trigger upgrades development on-going
- Several approaches investigated
- **Dedicated hardware is the ke**y to fast computation.
- **Not applicable for offline** processing unless by adopting heterogeneous hardware.







See https://ctdwit2017.lal.in2p3.fr/

Bottom Line

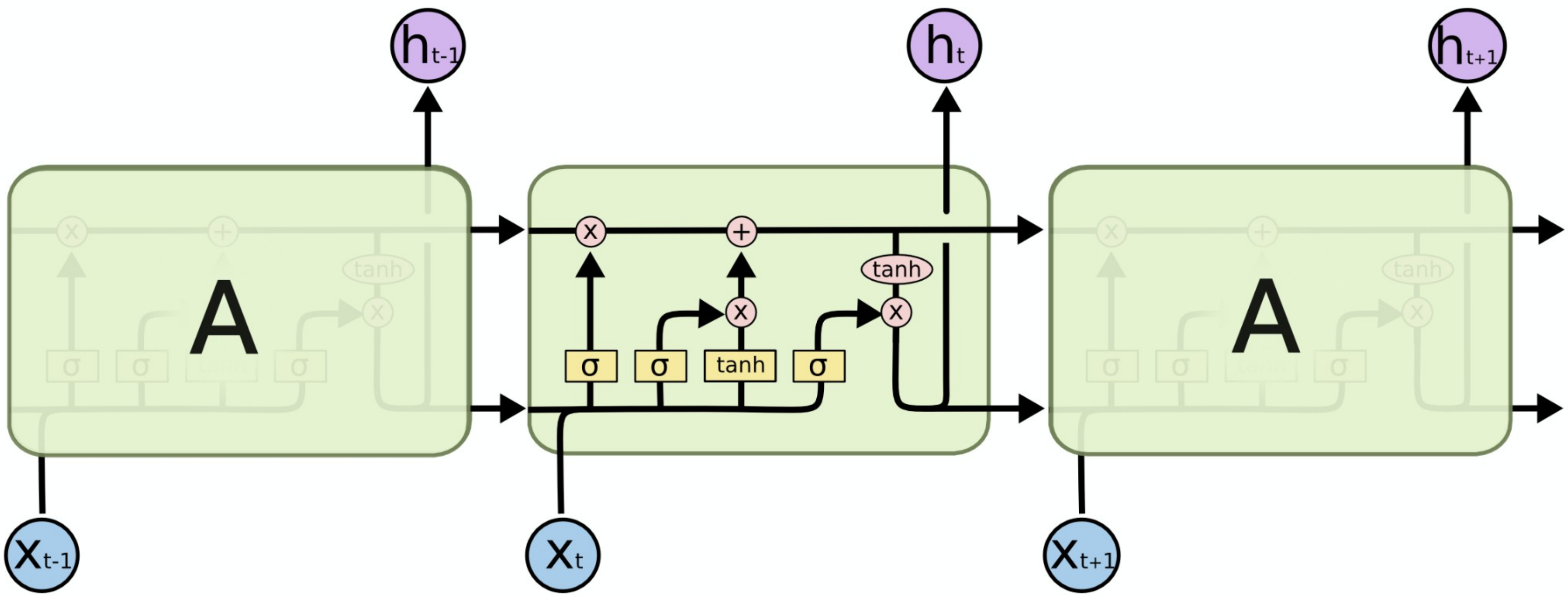
Current algorithms for tracking are highly performant physics-wise and scale badly computation-wise

Faster implementations are possible with dedicated hardware

Think outside the box for new methods

Long Short Term Memory - LSTM

Breakthrough in sequence processing by carrying over an internal state, “memory” of the previous items in the sequence, allowing for long range correlation



http://colah.github.io/posts/2015-08-Understanding-LSTMs/

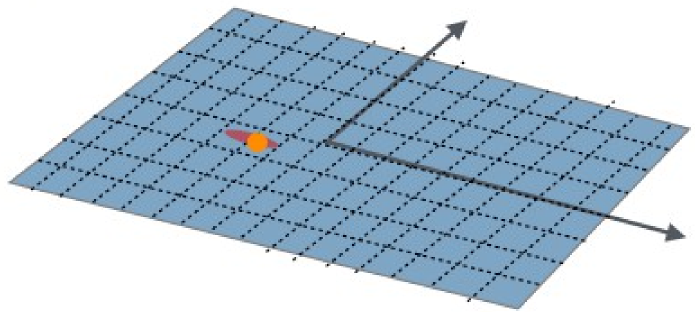
Tracking **Not** In a Nutshell

Several Times

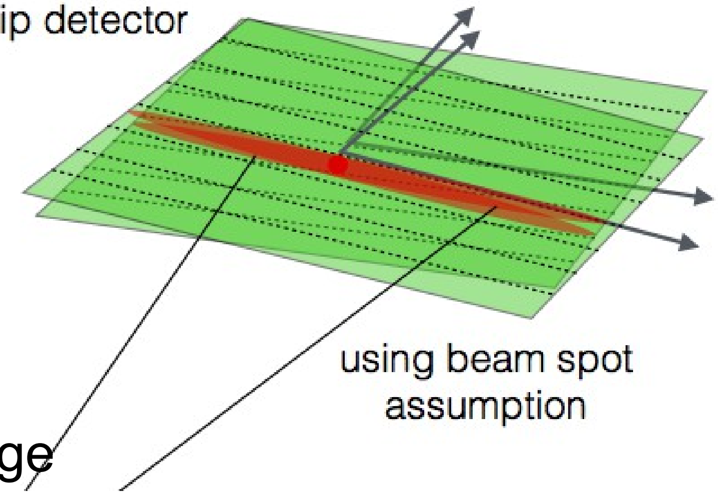
- **Hits preparation**
- **Seeding**
- **Pattern recognition**
- **Track fitting**
- **Track cleaning**

Hit Preparation

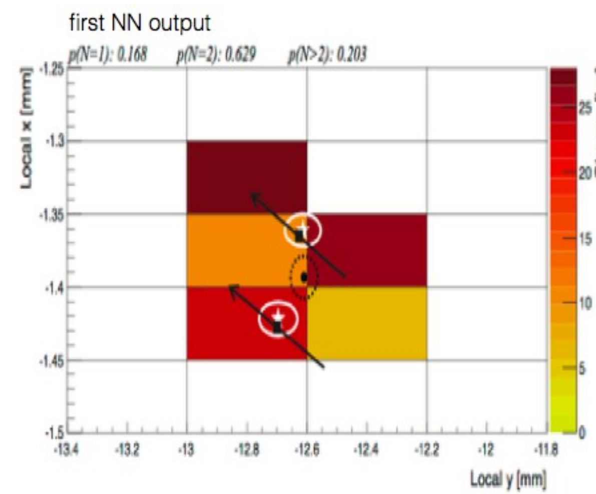
pixel detector



strip detector

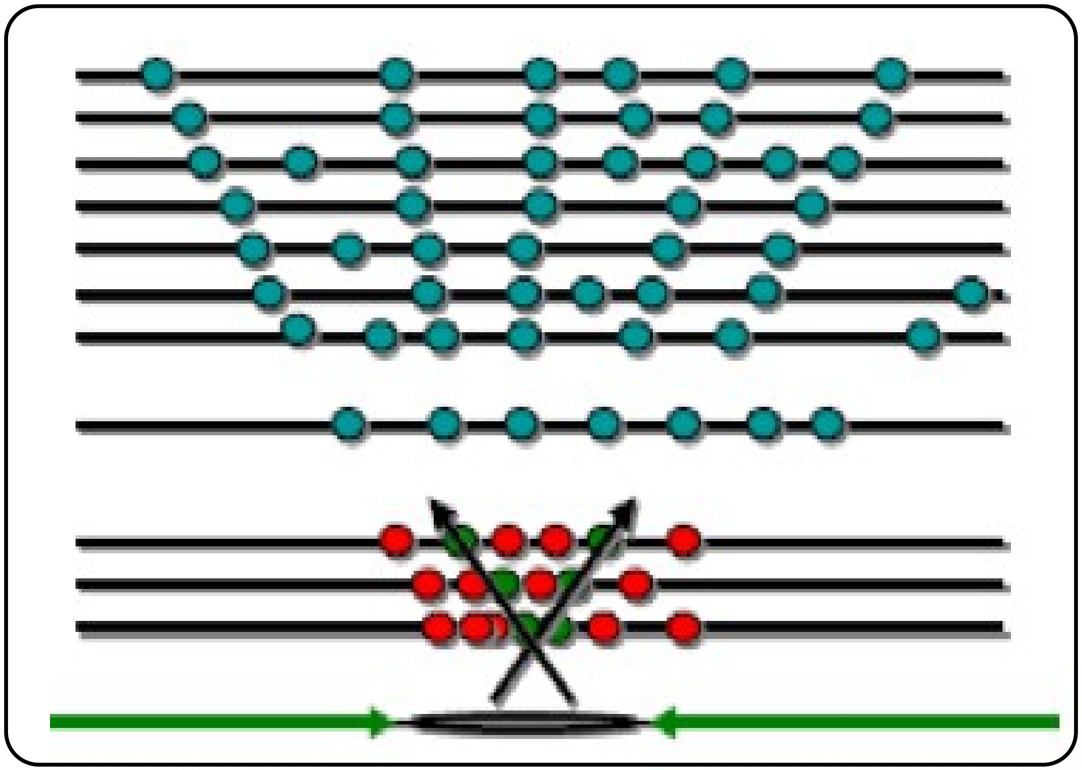


- Calculate the hit position from barycenter of charge deposits
- Use of neural net classifier to split cluster in ATLAS
- Access to trajectory local parameter from cluster shape
- Remove hits from previous tracking iterations
- HL-LHC design include double layers giving more constraints on the local trajectory parameters



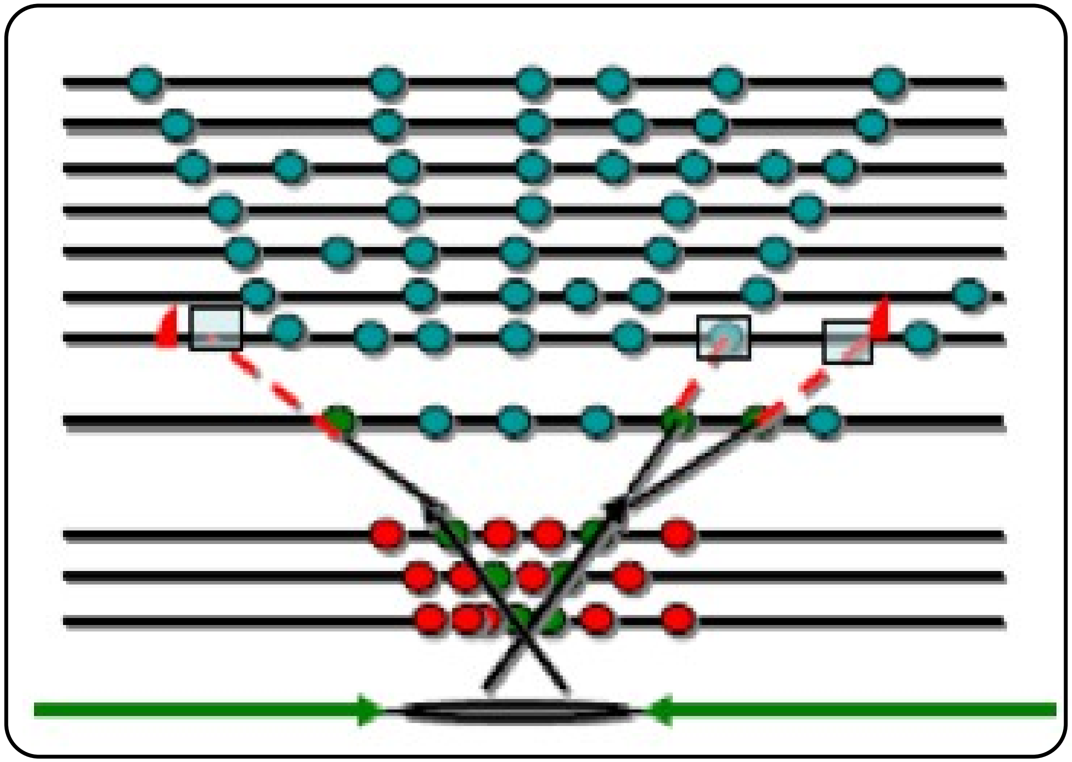
Example of cluster split

Seeding



- Combinatorics of 2 or 3 hits with tight/loose constraints to the beam spot or vertex
- Seed cleaning/purity plays in an important in reducing the CPU requirements of sub-sequent steps
  - ➔ Consider pixel cluster shape and charge to remove incompatible seeds
- Initial track parameters from helix fit

Pattern Recognition



- Use of the Kalman filter formalism with weight matrix
- Identify possible next layers from geometrical considerations
- Combinatorics with compatibles hits, retain N best candidates
- No smoothing procedure
- Resilient to missing modules
- Hits are mostly belonging to one track and one track only
- Hit sharing can happen in dense events, in the innermost part
- Lots of hits from low momentum particles

Kalman Filter

- Trajectory state propagation done either
  - ✓ Analytical (helix, fastest)
  - ✓ Stepping helix (fast)
  - ✓ Runge-Kutta (slow)
- Material effect added to trajectory state covariance
- Projection matrix of local helix parameters onto module surface
  - → Trivial expression due to local helix parametrisation
- Hits covariance matrix for pixel and stereo hits properly formed
  - ✗ Issue with strip hits and longitudinal error being non gaussian (square)

$$K_k = C_{k|k-1} H_k^{\top} (V_k + H_k C_{k|k-1} H_k^{\top})^{-1}$$

$$p_{k|k} = p_{k|k-1} + K_k (m_k - H_k p_{k|k-1})$$

$$C_{k|k-1} = (I - K_k H_k) C_{k|k-1}$$

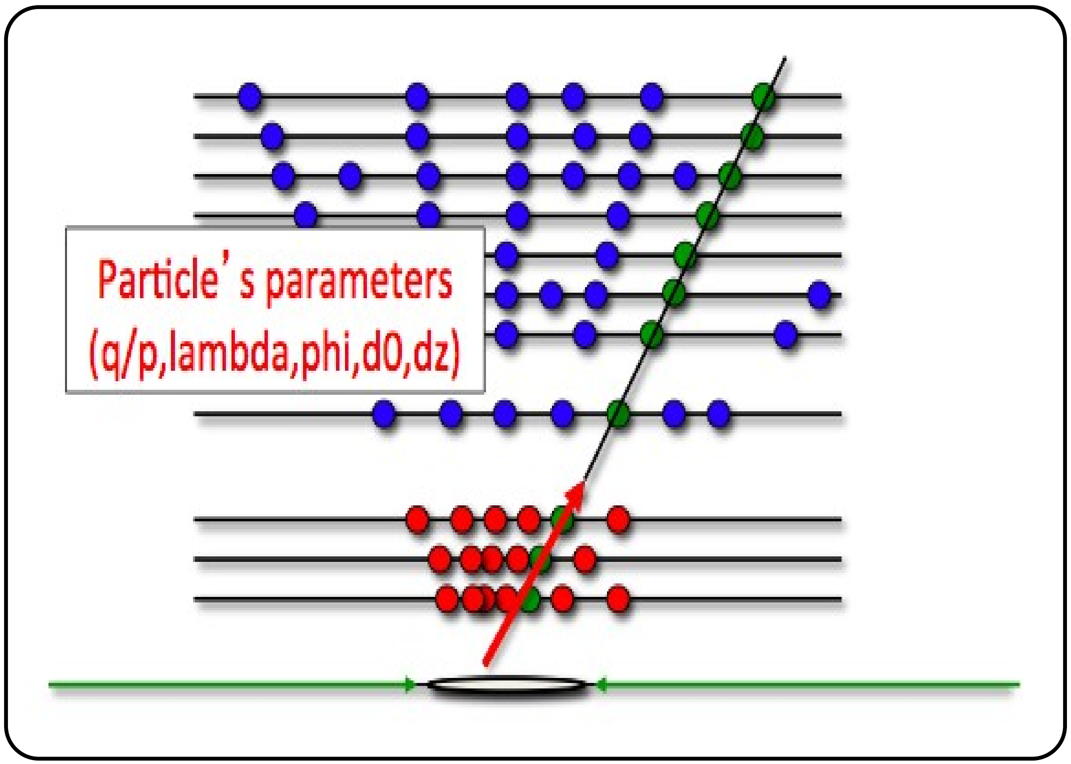
H_k is the projection matrix

V_k is the hit covariance matrix

$p_{i|j}$ is the trajectory state at i given j

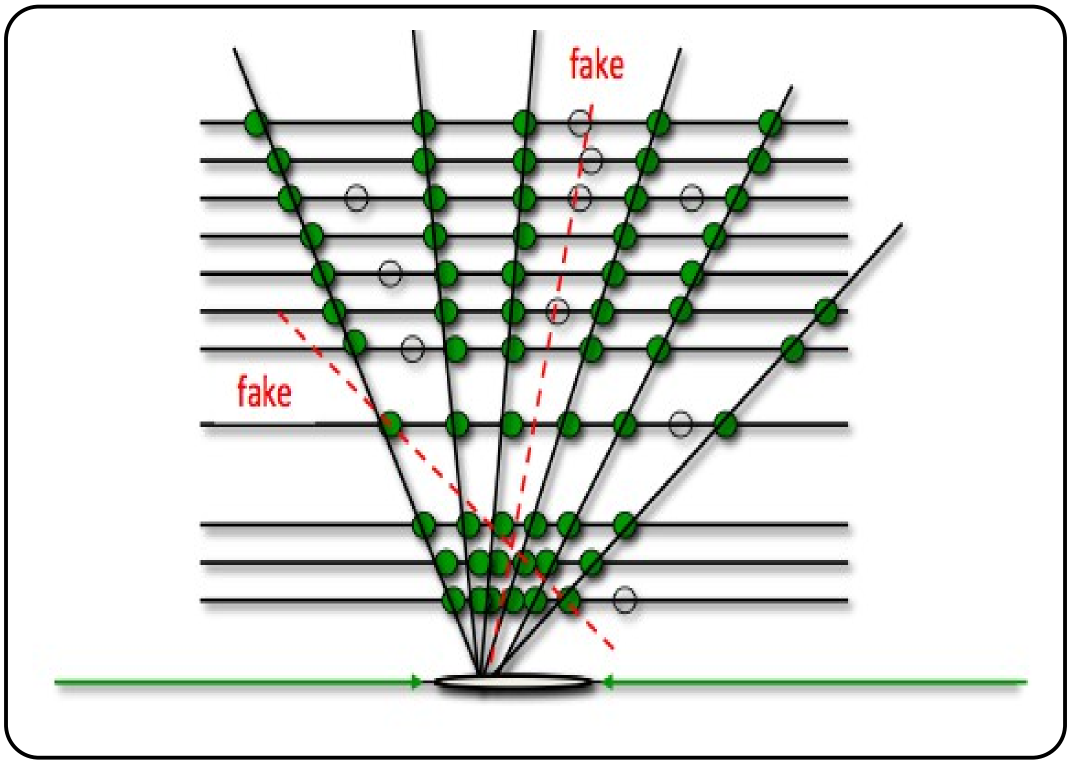
$C_{i|j}$ is the trajectory state covariance matrix at i given j

Track Fitting



- Use of the Kalman filter formalism with weight matrix
- Use of smoothing procedure to identify outliers
- Field non uniformity are taken into account
- Detector alignment taken into account

Cleaning, Selection



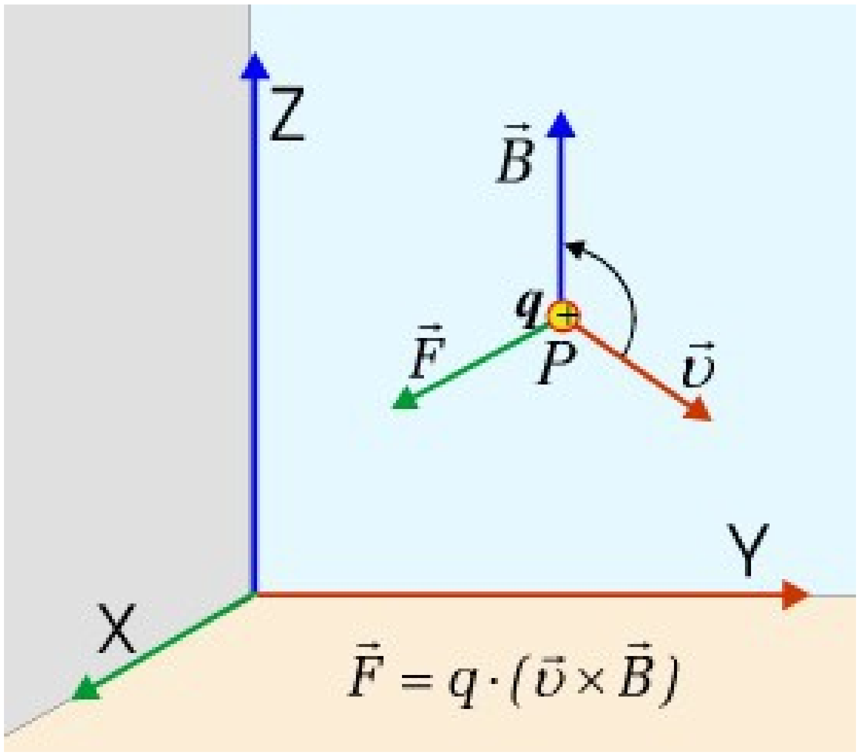
- Track quality estimated using ranking or classification method
  ➔Use of MVA
- Hits from high quality tracks are remove for the next iterations where applicable

A Charged Particle Journey

First order effect : electromagnetic elastic interaction of the charge particle with nuclei (heavy and multiply charged) and electrons (light and single charged)

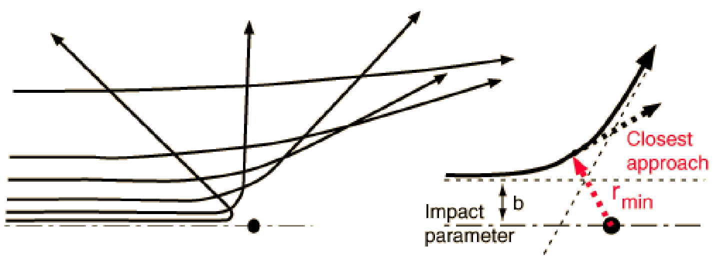
Second order effect : inelastic interaction with nuclei.

Magnetic Field

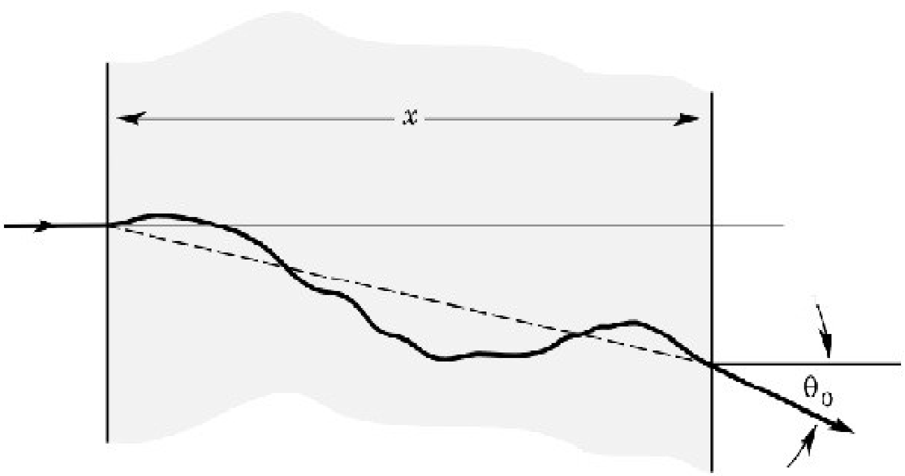


- Magnetic fieldB acts on charged particles in motion : Lorentz Force
- The solution in uniform magnetic field is an helix along the field : 5 parameters
- Helix radius proportional to the component of momentum perpendicular to B
- Separate particles in dense environment
- ➔ Bending induces radiation : bremsstrahlung
- ➔ The magnetic field has to be known to a good precision for accurate tracking of particle

Multiple Scattering



- **Deflection on nuclei** (effect from electron are negligible)
- Addition of scattering processes
- Gaussian approximation valid for substantial material traversed



Gaussian Approximation

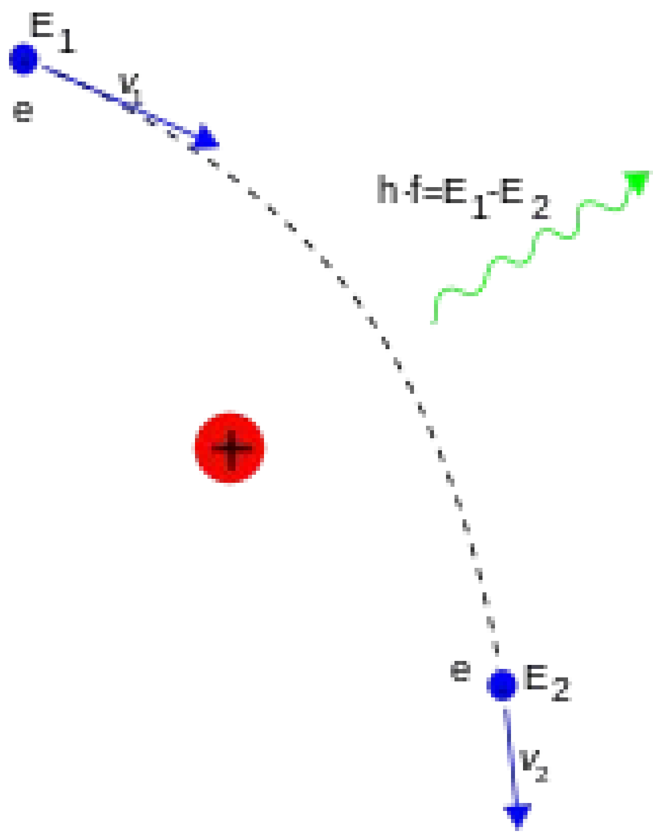
$$\theta^2 = \left(\frac{13.6MeV}{\beta c p}\right)^2 * \frac{x}{X_0}$$

β -particle velocity

ρ – material density

P - particle momenta

Bremsstrahlung



- Electromagnetic radiation of charged particles under acceleration due to nuclei charge
- Significant at low mass or high energy
- Discontinuity in energy loss spectrum due to photon emission and track curvature

→ Can be observed as kink in the trajectory or presence of collinear energetic photons

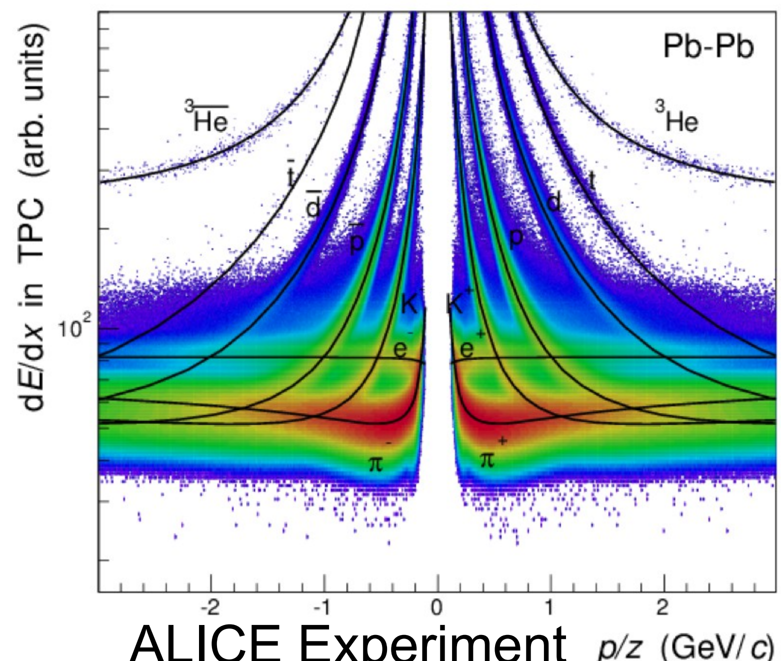
Energy Loss

- Momentum transfer to electrons when traversing material (effect of nuclei is negligible

$$dE/dx = k_1 \frac{Z}{A}\frac{1}{\beta^2}\rho\left(\ln\left(\frac{2m_e c^2\beta^2}{I(1-\beta^2)}\right)-\beta^2-\frac{\delta}{2}\right)$$

β -particle velocity
ρ – material density
Z - atomic number of absorber
A – mass number of absorber
I – mean excitation energy
δ – density effect correction factor – material dependent and β dependent

- Energy loss at low momentum depends on mass : can be used as mass spectrometer



ALICE Experiment

Summary on Material Effects

- Collective effects can be estimated statistically and taken into account in how they modify the trajectory

- Bremstrahlung and nuclear interactions significantly distort trajectories

Scene Labeling

From talk of LeCunn at CERN

Scene Labeling

LeCunn Seminar at CERN

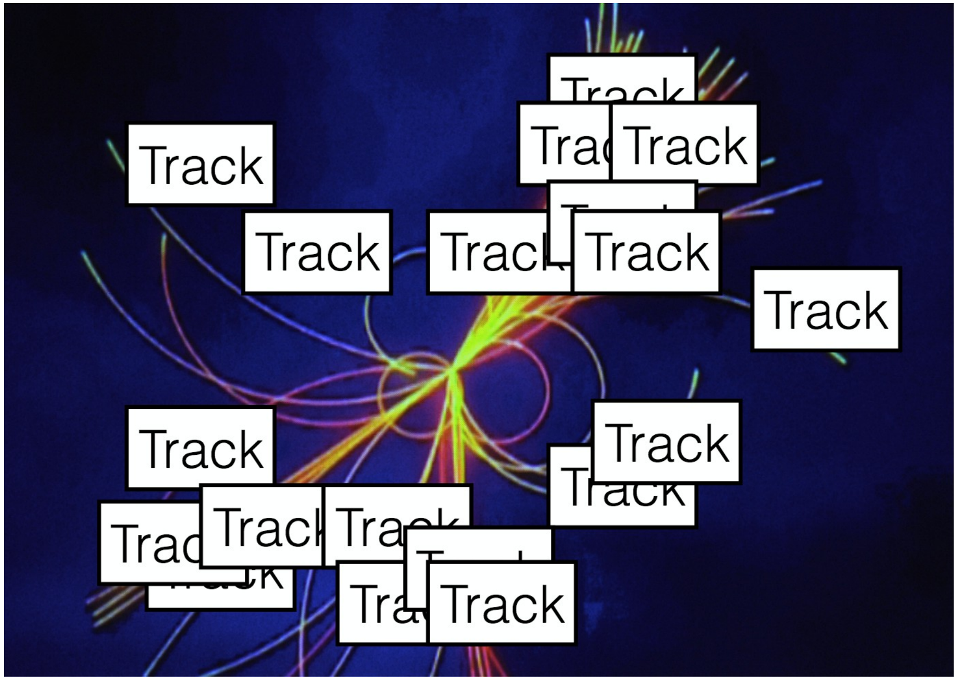


Photo by Pier Marco Tacca/Getty Images