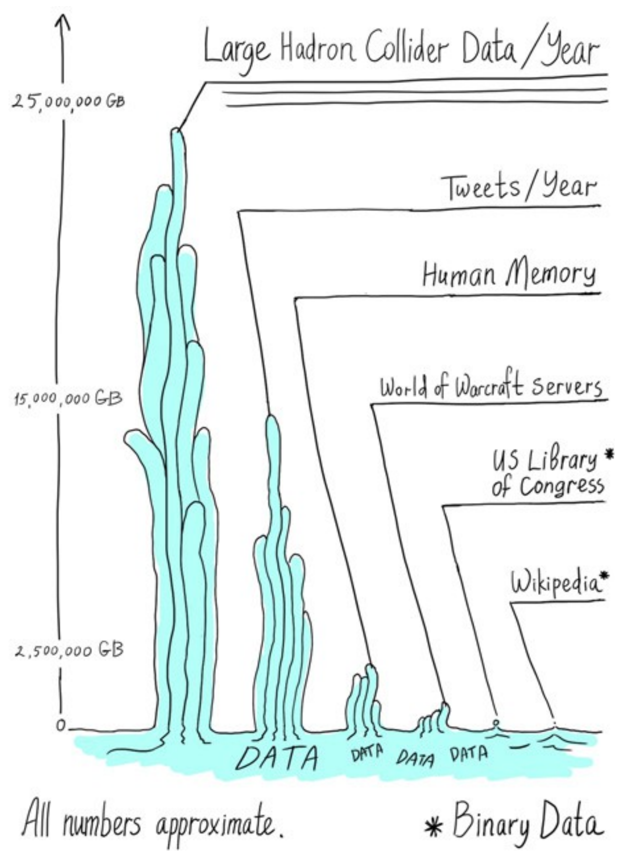
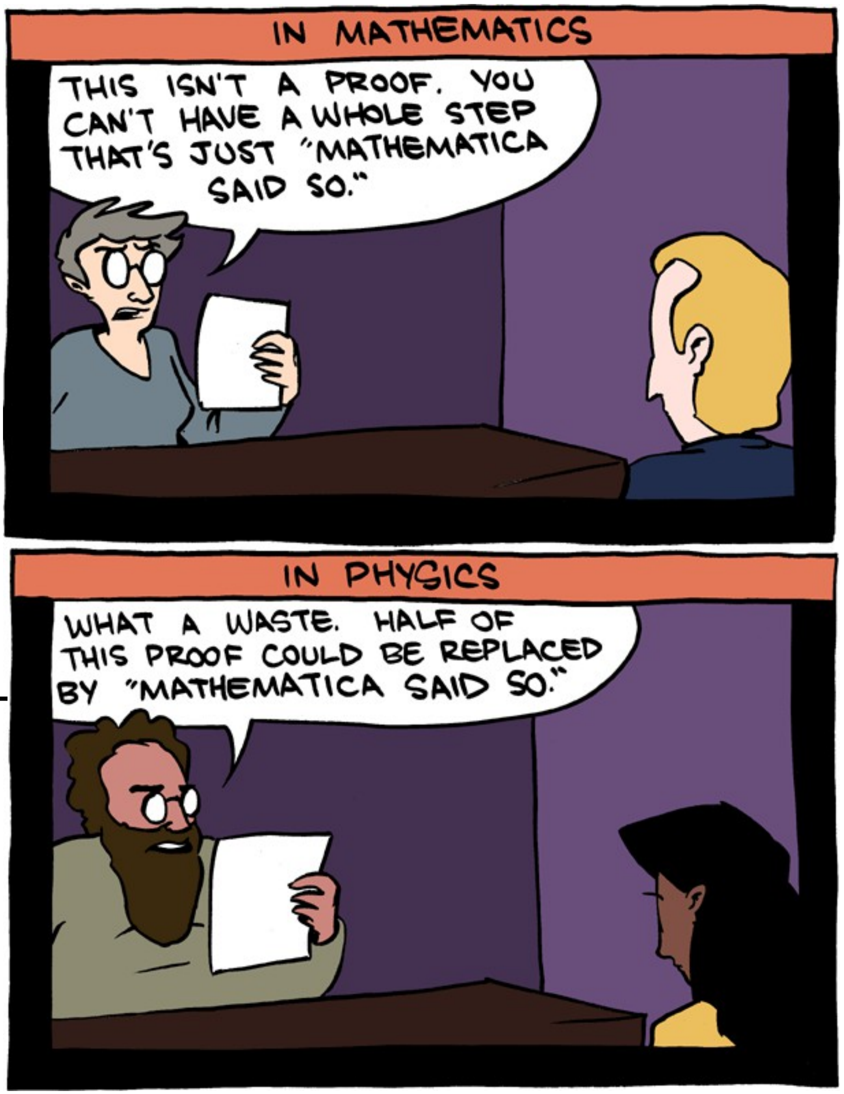
Data Analysis in Experimental Particle Physics



Different languages



From http://www.smbc-comics.com

- Second time in my life that I give a talk to a mixed audience of physicists and mathematicians
  - But the first time was 17 years ago...
- Please don't hesitate to stop me if my jargon is unintelligible
- If question comes from the Math side, large probability that I will not even understand the question; please be patient with me

Standard Model (SM) in a nutshell

Known elementary particles, and their inter-relationships:

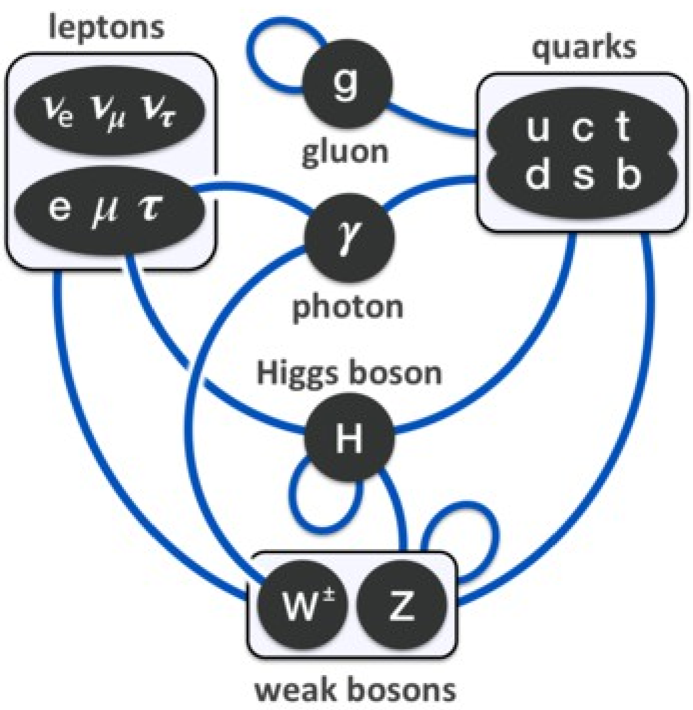
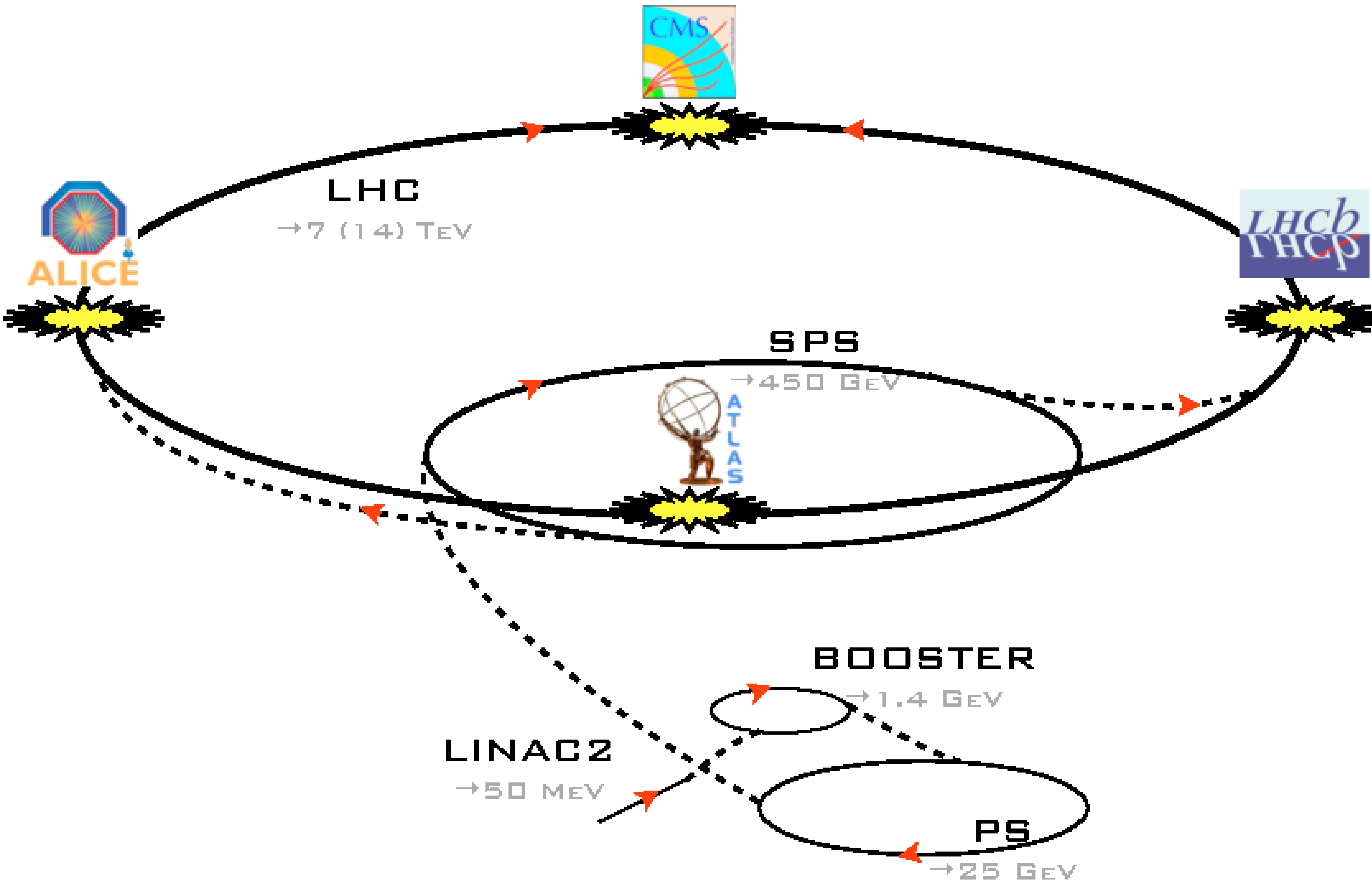


Image source: wikipedia

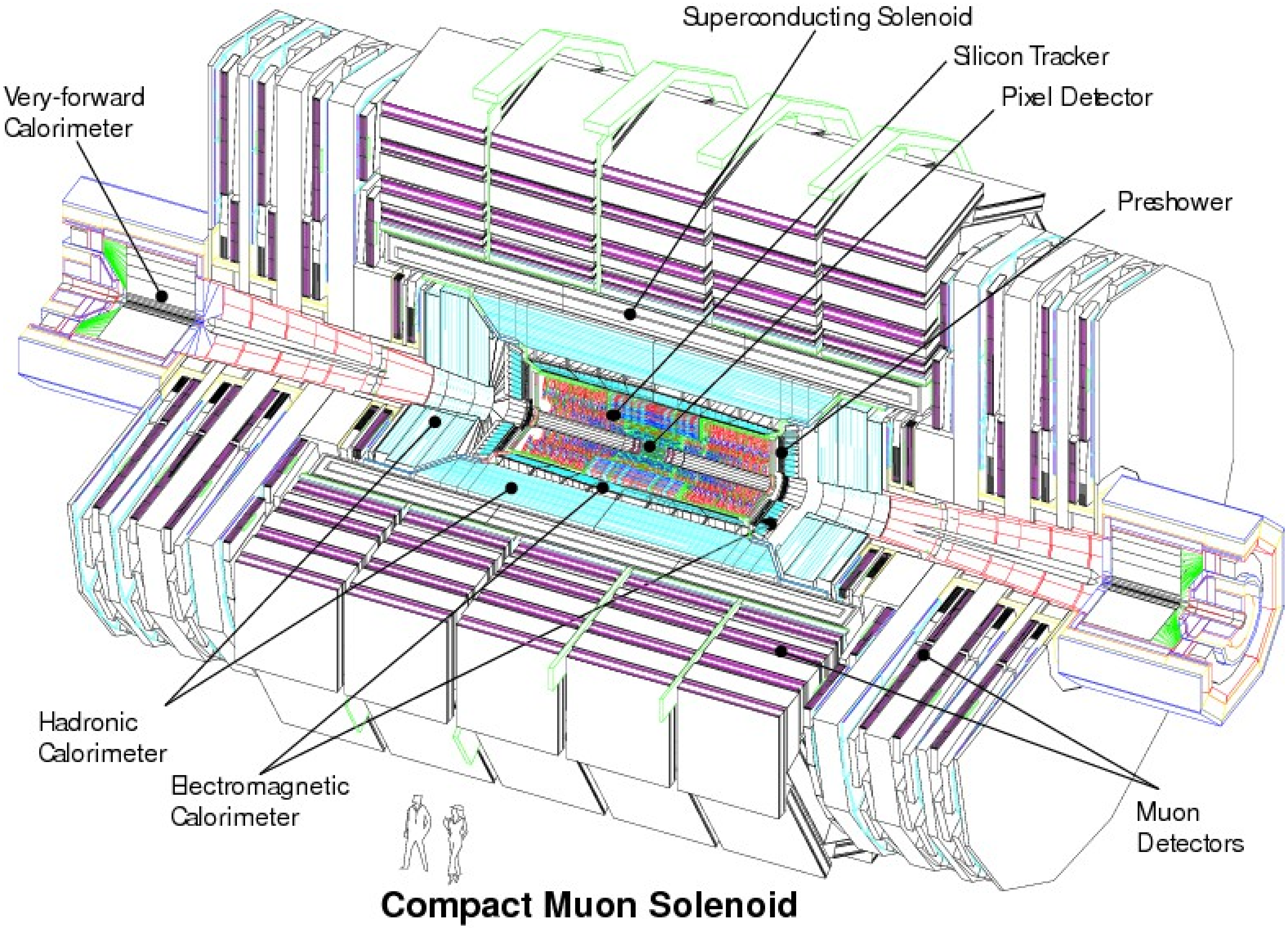
(Gravity is not part of the picture)

The Large Hadron Collider



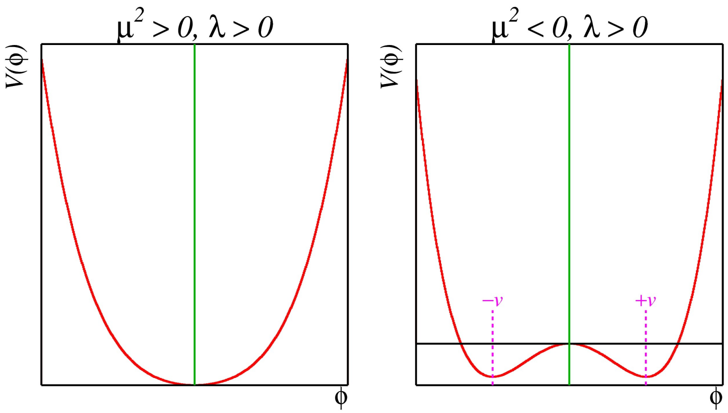
Picture taken from https://inspirehep.net/record/1119569

CMS

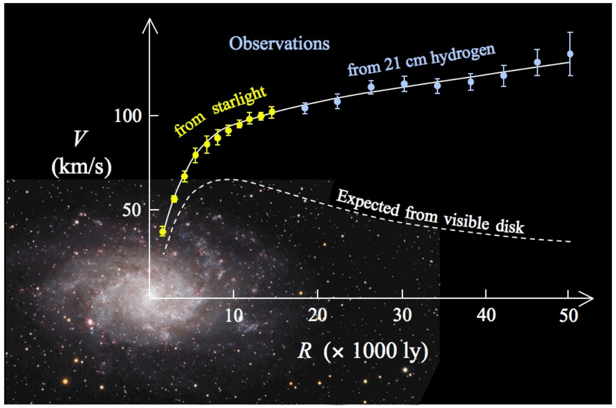

Superconducting Solenoid
Silicon Tracker
Very-forward Calorimeter
Pixel Detector
Preshower
Hadronic Calorimeter
Electromagnetic Calorimeter
Muon Detectors
Compact Muon Solenoid

Some big questions that LHC was designed to address

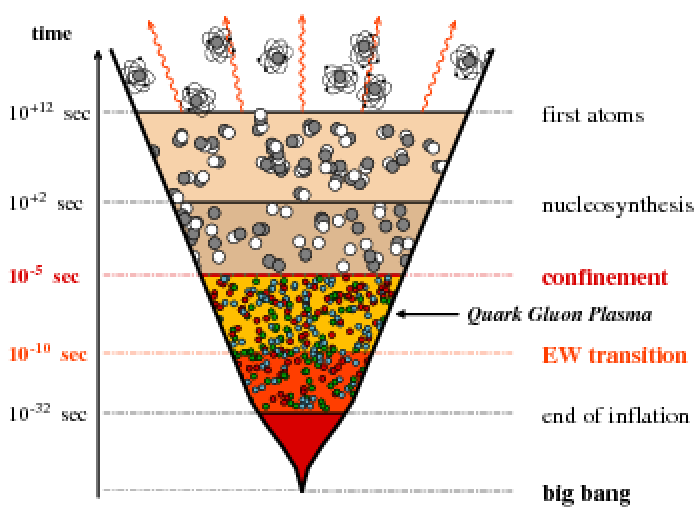
Origin of mass



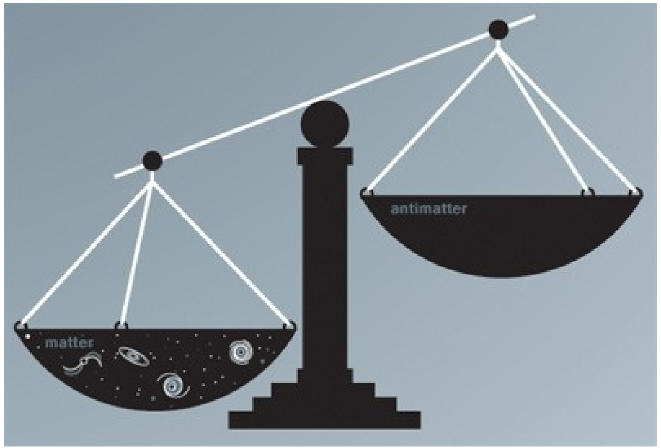
Nature of Dark Matter



Behaviour of early Universe



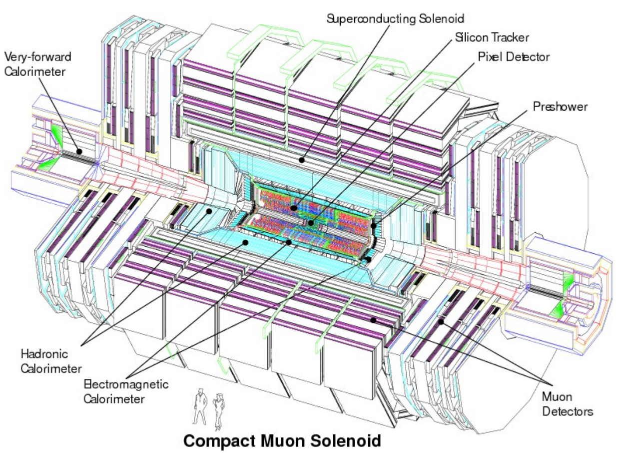
Reason for matter / antimatter imbalance



LHC check-list

- Confirmation (or not) of the Brout-Englert-Higgs mechanism, i.e., explaining where the mass of elementary particles is coming from
  - Done
- Check if Dark Matter is made of new particles; if so, study them
  - Trying hard
- Study the hot medium that filled the early Universe
  - Very advanced (using special LHC runs with heavy ions as projectiles)
- Explain the matter/anti-matter imbalance of the Universe
  - Trying hard
- Search for additional particles, forces, dimensions of space
  - Due to LHC data, some of the most promising theories of 2010 are seriously in trouble in 2017 (still alive if "fine tuned" a bit)
- Precisely measure the properties of the known particles
  - Unprecedented precision achieved almost everywhere

How did we do that, in practice?



data

knowledge

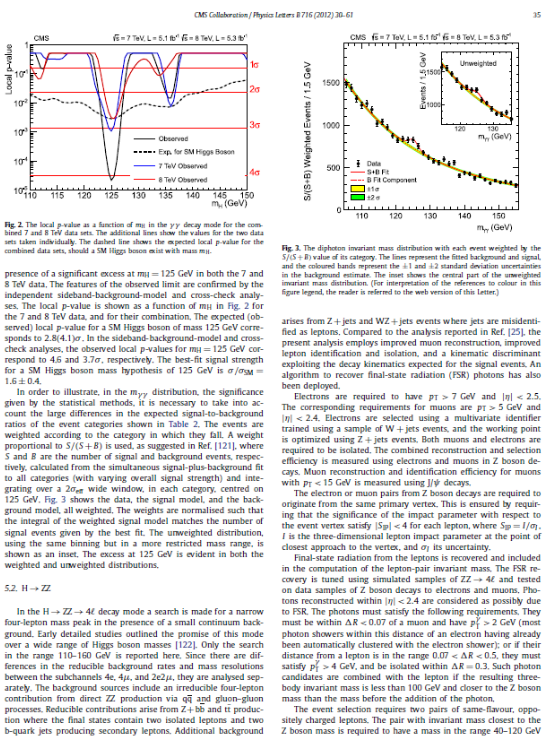

CMS Collaboration / Physics Letters B 716 (2012) 30–61 35

Fig. 2. The local p-value as a function of m_H in the $\gamma\gamma$ decay mode for the combined 7 and 8 TeV data sets. The additional lines show the values for the two data sets taken individually. The dashed line shows the expected local p-value for the combined data sets, should a SM Higgs boson exist with mass m_H.

presence of a significant excess at $m_H = 125$ GeV in both the 7 and 8 TeV data. The features of the observed limit are confirmed by the independent sideband-background-model and cross-check analyses. The local p-value is shown as a function of m_H in Fig. 2 for the 7 and 8 TeV data, and for their combination. The expected (observed) local p-value for a SM Higgs boson of mass 125 GeV corresponds to $2.8(4.1)\sigma$. In the sideband-background-model and cross-check analyses, the observed local p-values for $m_H = 125$ GeV correspond to 4.6 and 3.7σ, respectively. The best-fit signal strength for a SM Higgs boson mass hypothesis of 125 GeV is $\sigma/\sigma_{SM} = 1.6 \pm 0.4$.

In order to illustrate, in the $m_{\gamma\gamma}$ distribution, the significance given by the statistical methods, it is necessary to take into account the large differences in the expected signal-to-background ratios of the event categories shown in Table 2. The events are weighted according to the category in which they fall. A weight proportional to $S/(S+B)$ is used, as suggested in Ref. [121], where S and B are the number of signal and background events, respectively, calculated from the simultaneous signal-plus-background fit to all categories (with varying overall signal strength) and integrating over a $2\sigma_{eff}$ wide window, in each category, centred on 125 GeV. Fig. 3 shows the data, the signal model, and the background model, all weighted. The weights are normalised such that the integral of the weighted signal model matches the number of signal events given by the best fit. The unweighted distribution, using the same binning but in a more restricted mass range, is shown as an inset. The excess at 125 GeV is evident in both the weighted and unweighted distributions.

5.2. $H \to ZZ$

In the $H \to ZZ \to 4\ell$ decay mode a search is made for a narrow four-lepton mass peak in the presence of a small continuum background. Early detailed studies outlined the promise of this mode over a wide range of Higgs boson masses [122]. Only the search in the range 110–160 GeV is reported here. Since there are differences in the reducible background rates and mass resolutions between the subchannels 4e, 4μ, and $2e2\mu$, they are analysed separately. The background sources include an irreducible four-lepton contribution from direct ZZ production via $q\bar{q}$ and gluon–gluon processes. Reducible contributions arise from $Z + b\bar{b}$ and $t\bar{t}$ production where the final states contain two isolated leptons and two b-quark jets producing secondary leptons. Additional background

Fig. 3. The diphoton invariant mass distribution with each event weighted by the $S/(S+B)$ value of its category. The lines represent the fitted background and signal, and the coloured bands represent the ±1 and ±2 standard deviation uncertainties in the background estimate. The inset shows the central part of the unweighted invariant mass distribution. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this Letter.)

arises from Z + jets and WZ + jets events where jets are misidentified as leptons. Compared to the analysis reported in Ref. [25], the present analysis employs improved muon reconstruction, improved lepton identification and isolation, and a kinematic discriminant exploiting the decay kinematics expected for the signal events. An algorithm to recover final-state radiation (FSR) photons has also been deployed.

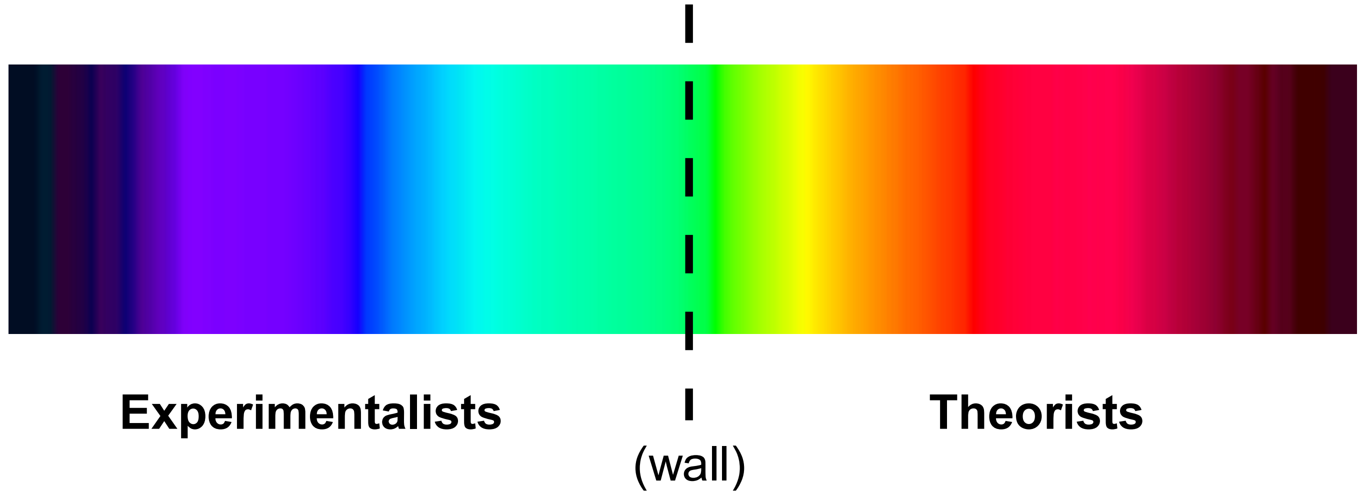
Electrons are required to have $p_T > 7$ GeV and $|\eta| < 2.5$. The corresponding requirements for muons are $p_T > 5$ GeV and $|\eta| < 2.4$. Electrons are selected using a multivariate identifier trained using a sample of W + jets events, and the working point is optimized using Z + jets events. Both muons and electrons are required to be isolated. The combined reconstruction and selection efficiency is measured using electrons and muons in Z boson decays. Muon reconstruction and identification efficiency for muons with $p_T < 15$ GeV is measured using J/ψ decays.

The electron or muon pairs from Z boson decays are required to originate from the same primary vertex. This is ensured by requiring that the significance of the impact parameter with respect to the event vertex satisfy $|S_{IP}| < 4$ for each lepton, where $S_{IP} = I/\sigma_I$, I is the three-dimensional lepton impact parameter at the point of closest approach to the vertex, and σ_I its uncertainty.

Final-state radiation from the leptons is recovered and included in the computation of the lepton-pair invariant mass. The FSR recovery is tuned using simulated samples of $ZZ \to 4\ell$ and tested on data samples of Z boson decays to electrons and muons. Photons reconstructed within $|\eta| < 2.4$ are considered as possibly due to FSR. The photons must satisfy the following requirements. They must be within $\Delta R < 0.07$ of a muon and have $p_T^\gamma > 2$ GeV (most photon showers within this distance of an electron having already been automatically clustered with the electron shower); or if their distance from a lepton is in the range $0.07 < \Delta R < 0.5$, they must satisfy $p_T^\gamma > 4$ GeV, and be isolated within $\Delta R = 0.3$. Such photon candidates are combined with the lepton if the resulting three-body invariant mass is less than 100 GeV and closer to the Z boson mass than the mass before the addition of the photon.

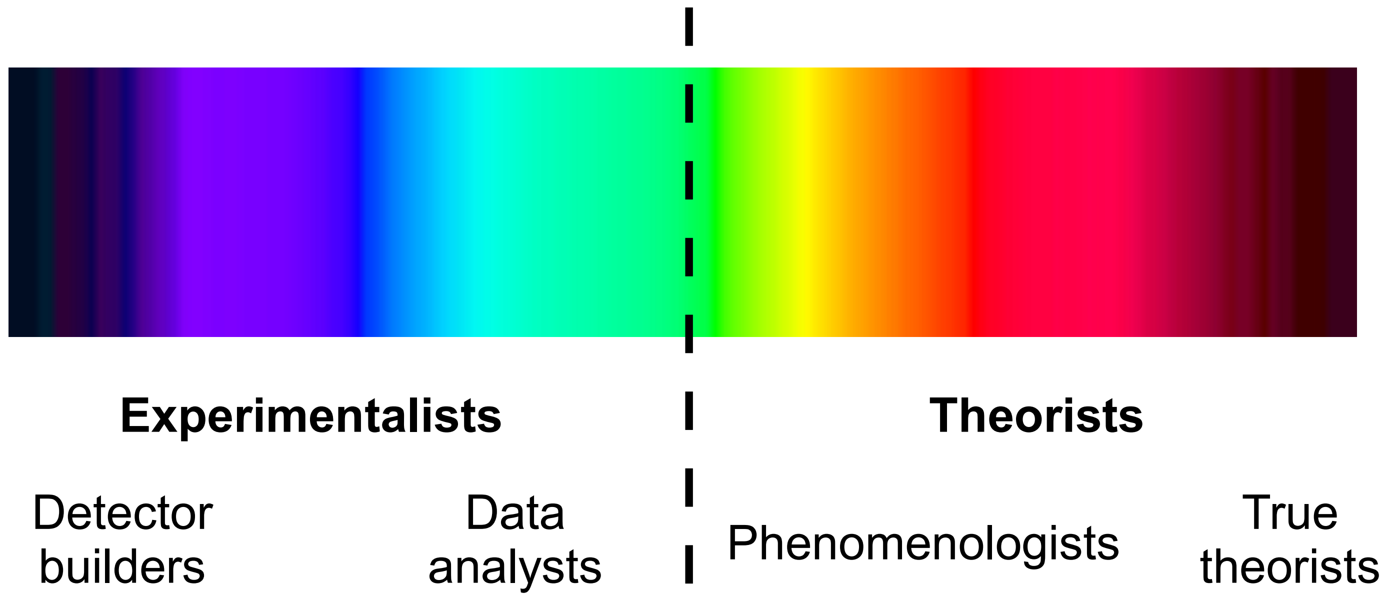
The event selection requires two pairs of same-flavour, oppositely charged leptons. The pair with invariant mass closest to the Z boson mass is required to have a mass in the range 40–120 GeV

Human spectrum of Particle Physics



- This distinction emerged in Physics around the early XX century
  - Enrico Fermi (1901-1954) was probably the last physicist able to be at once a *great* theorist and a *great* experimentalist
  - Nowadays, at least in Particle Physics, hopping from one side of the spectrum to the other is very unusual
- But:
  - Exp-theory collaborations still happen (rarely)
  - Undergraduate training is almost the same

Human spectrum of Particle Physics



- Unlikely that a *detector builder* and a *true theorist* will coauthor a paper or even attend the same conference (unless it's a big one)
- *Data analysts* and *phenomenologists* usually attend the same conferences and sometimes co-author a paper
- A 100% *detector builder* and a 100% *data analyst* usually coauthor hundreds of papers, but they may not talk to each other for years!
  - The split happened recently (80's?), with the start of Big Science
  - However, having both skills is still considered an asset for career: perception that the ideal experimental physicist is a bit of everything

Data analysis in 1937

Raw data:

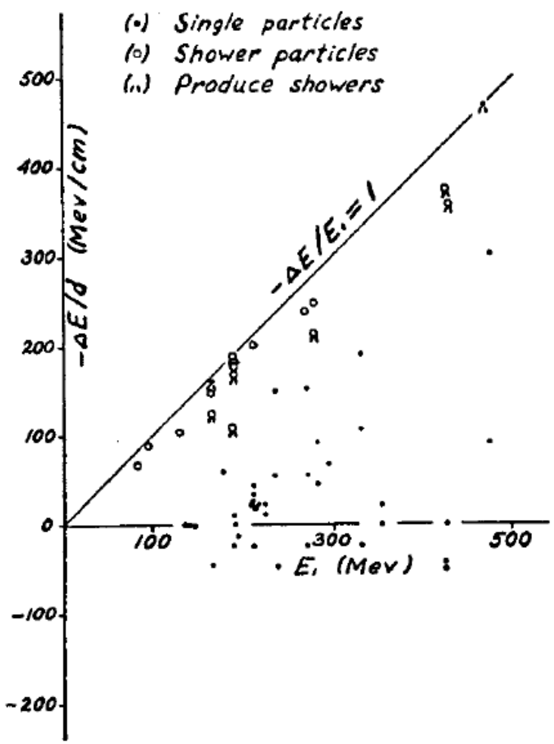


Figure 1: Energy loss in 1 cm of platinum.

Reduced data:

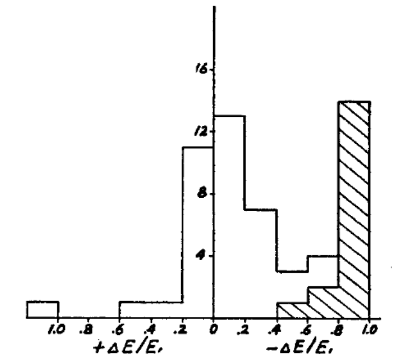


Figure 2: Distribution of fractional losses in 1 cm of platinum.

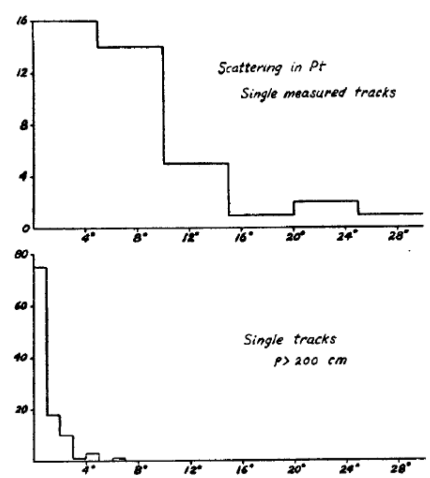


Figure 3: Scattering distributions in 1 cm of platinum.

The Money Plot:

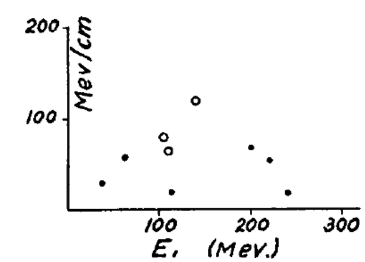


Figure 4: Early measurements of energy loss in 0.7-1.5 cm of Pb. Dots indicate single particles; circles, shower particles.

Mathematical tools used:

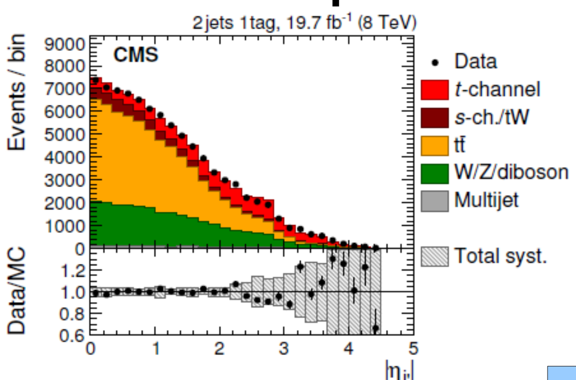
+, -, ×, ÷

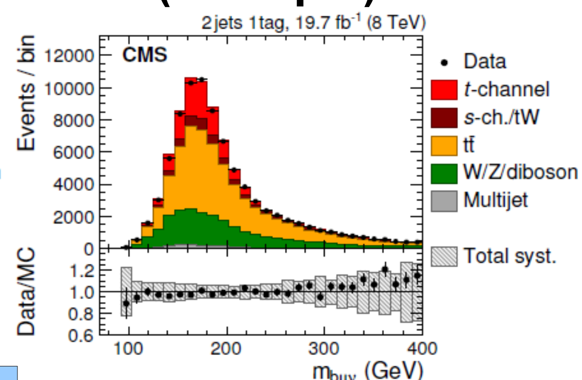
Level of physics insight required: huge

S.Neddemeyer & C.Anderson, Phys.Rev. 51 (1937) 884
(Proof that cosmic rays contained new particles)

Data analysis in 2017

Some complicated features (example):

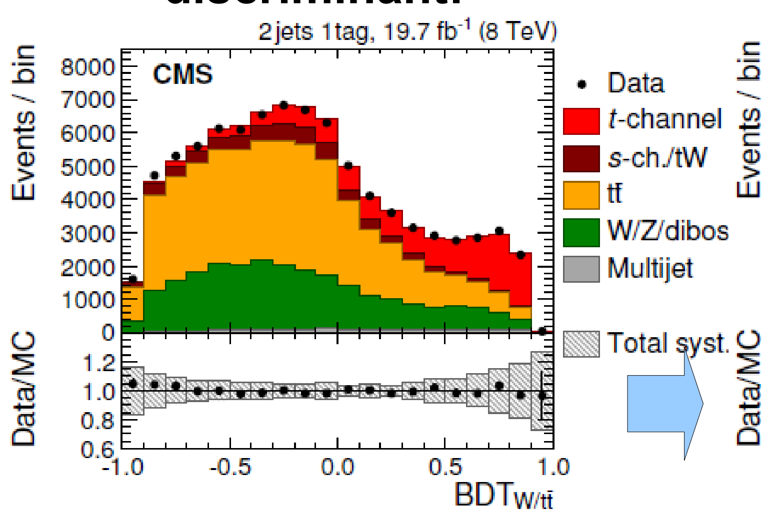




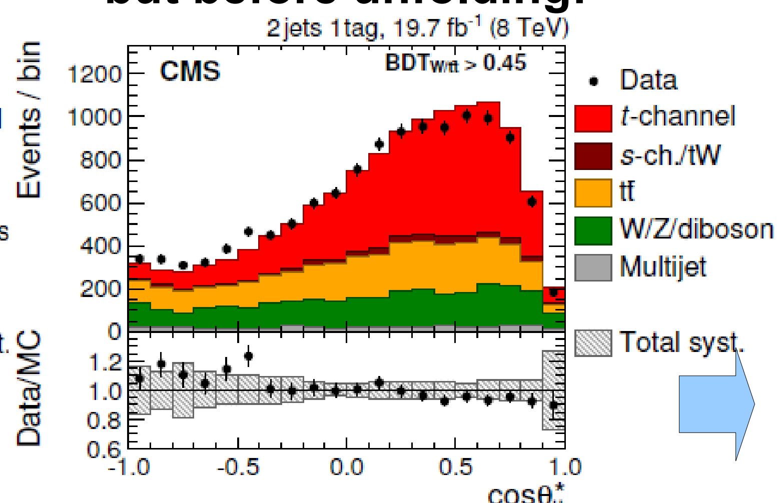
Mathematical tools used:

- Boosted Decision Trees
- Maximum Likelihood fit with nuisance marginalization
- Matrix inversion with Tikhonov regularization

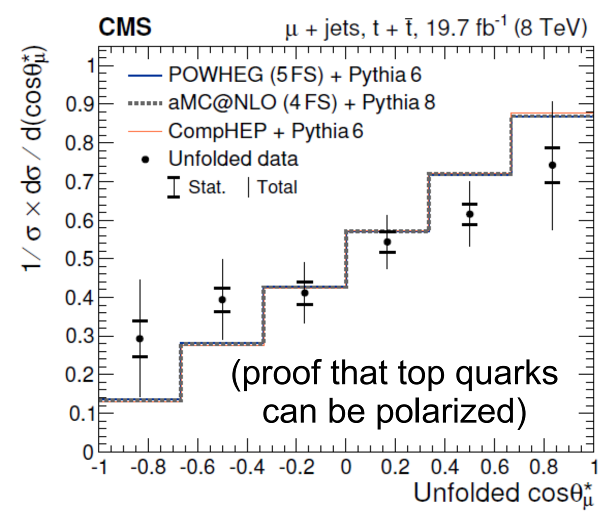
Machine-learned discriminant:



After machine-learning and fit, but before unfolding:

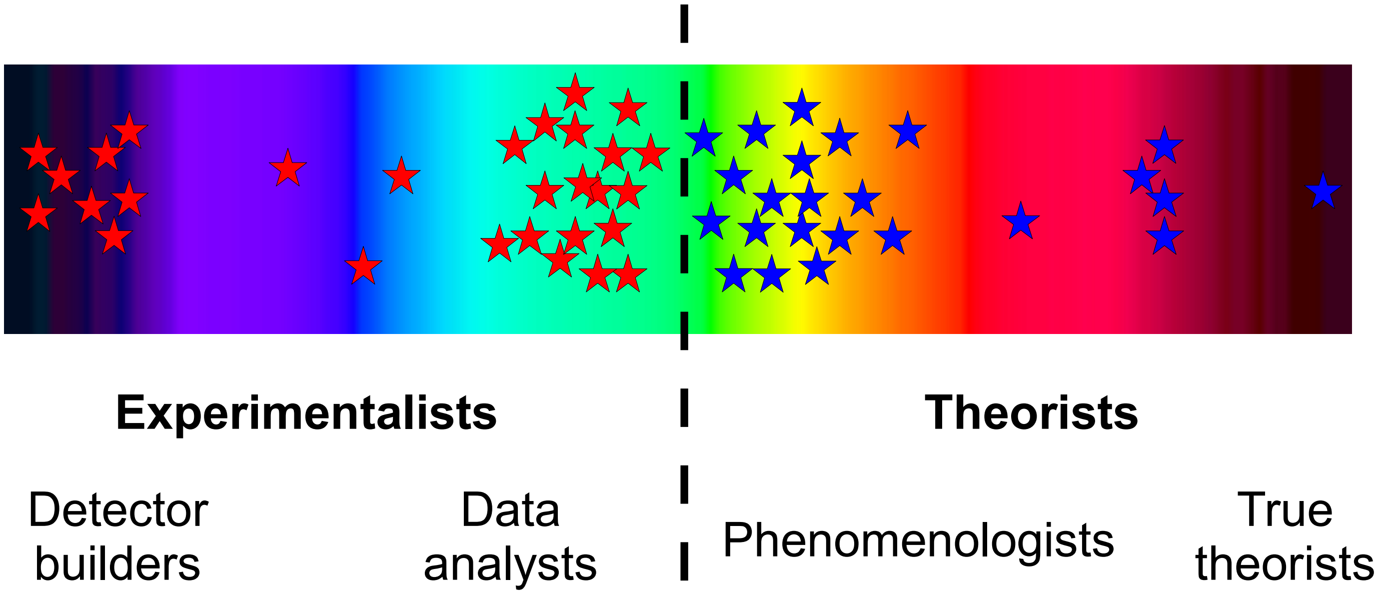


The Money Plot:



CMS Coll. (M.Komm, AG, et ~2000 al.), JHEP 04 (2016) 073

CP3 spectrum



- Qualitative / subjective classification based on https://cp3.irmp.ucl.ac.be/Members (from professors to grad students)
- Bottom line: the subject of *data analysis* is very relevant at CP3, directly (experimental side) or indirectly (theory side)

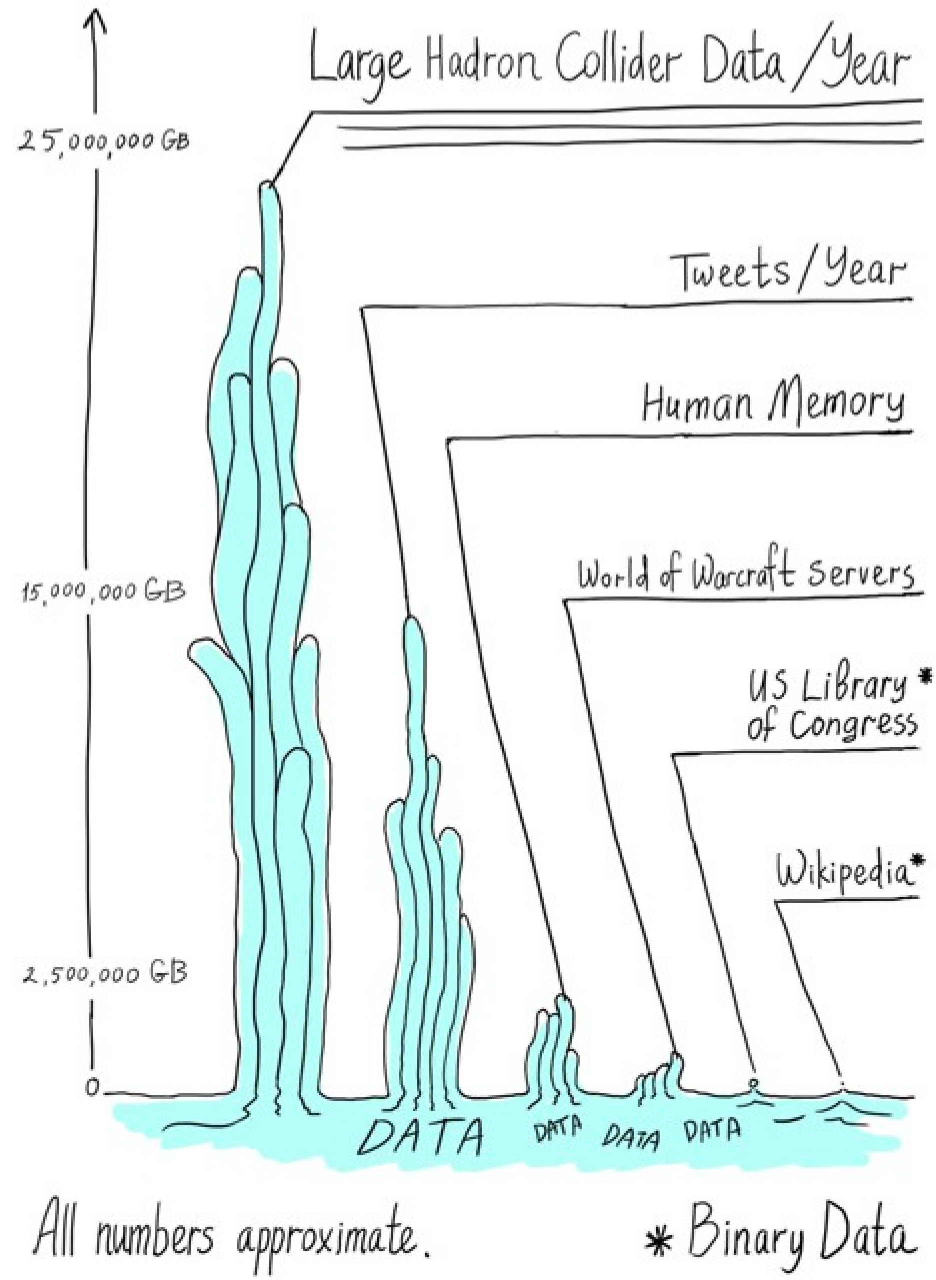
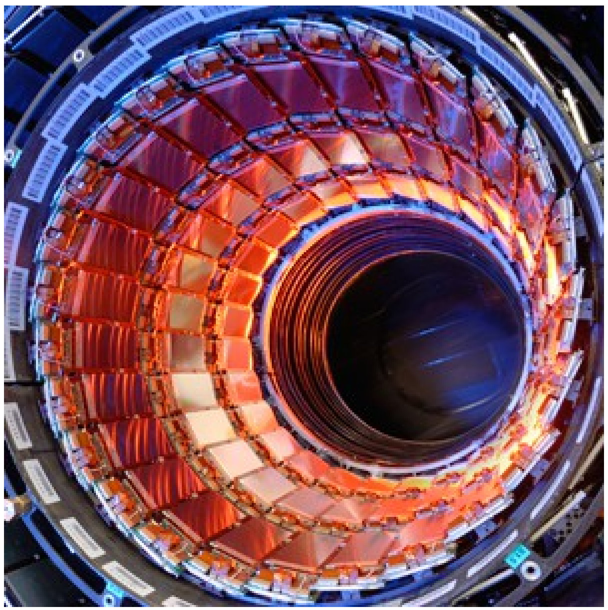
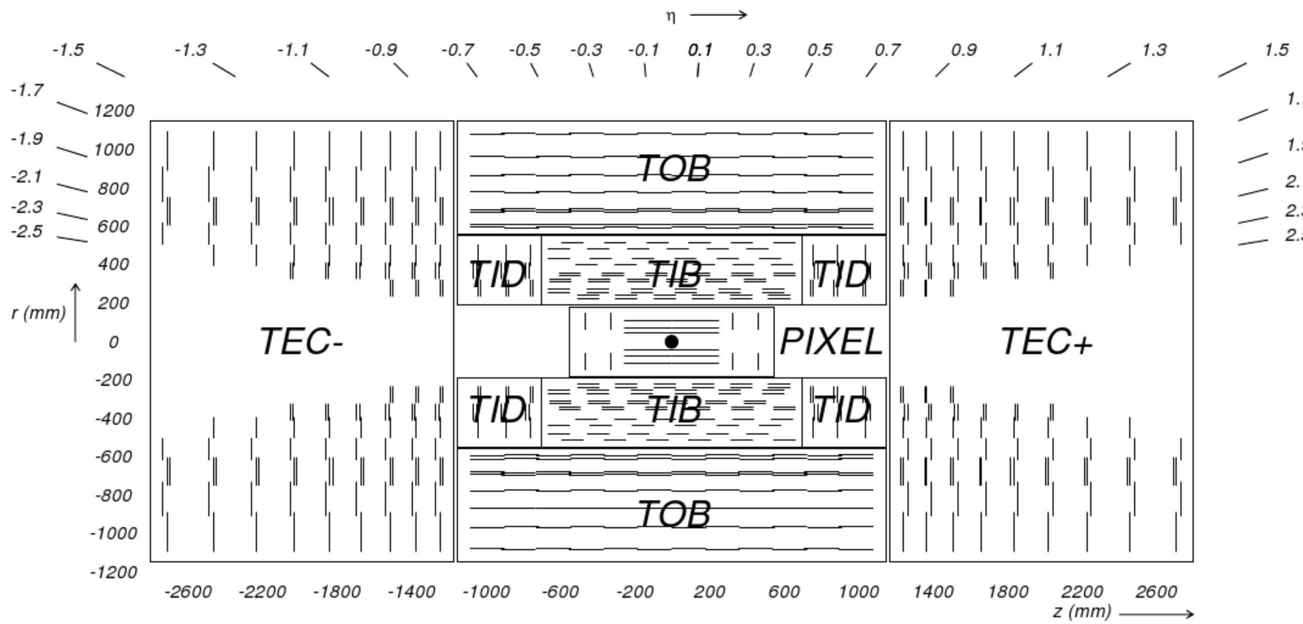


Image from http://www.symmetrymagazine.org/article/august-2012/particle-physics-tames-big-data

Example: the CMS inner tracker



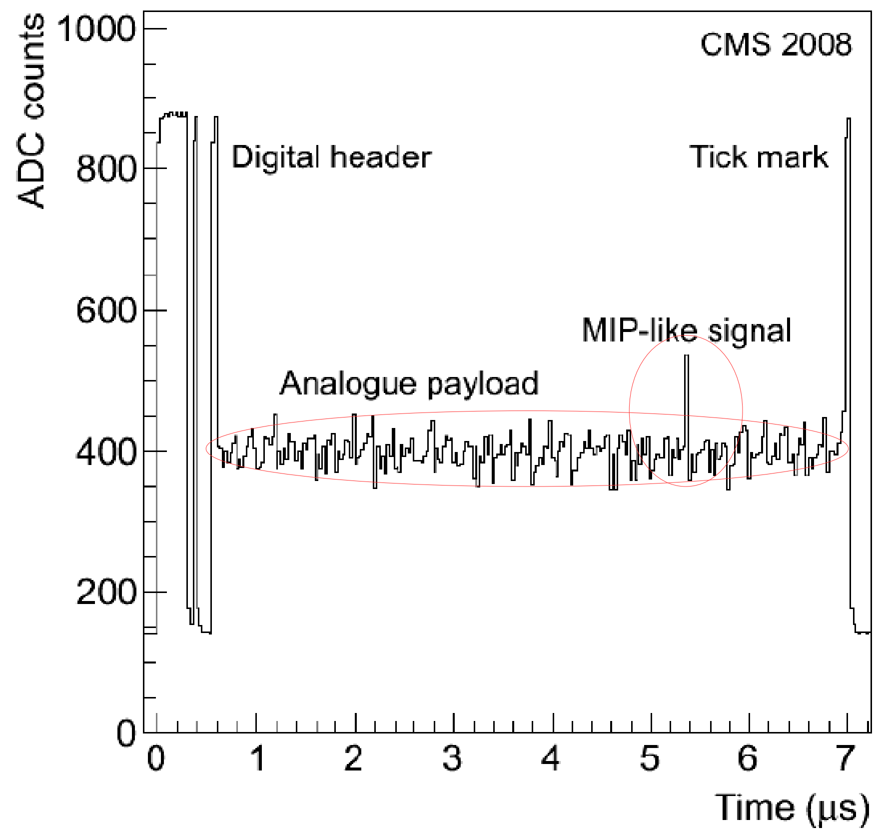


- Innermost part of CMS; a cylinder of 1.2 m radius (CMS: 7.5 m)
- Electrically charged particles (and only them) give a signal each time they cross one of its layers
- Each layer is made of several modules, each module has hundreds of sensitive units (pixels or microstrips) with spatial resolution of O(0.1 mm)
- Its volume is only a fraction of all CMS, but it dominates the size of its raw data with its 77 millions of sensitive units

Raw data from the tracker

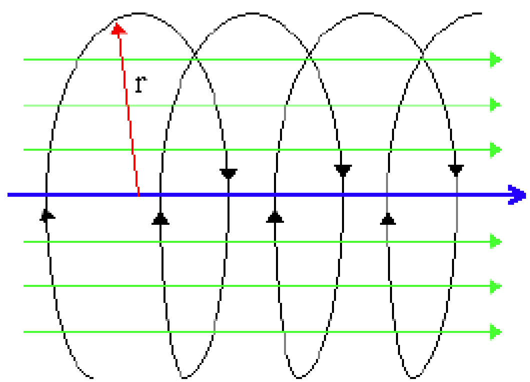
Example from one of the two technologies employed in the CMS tracker (microstrips):

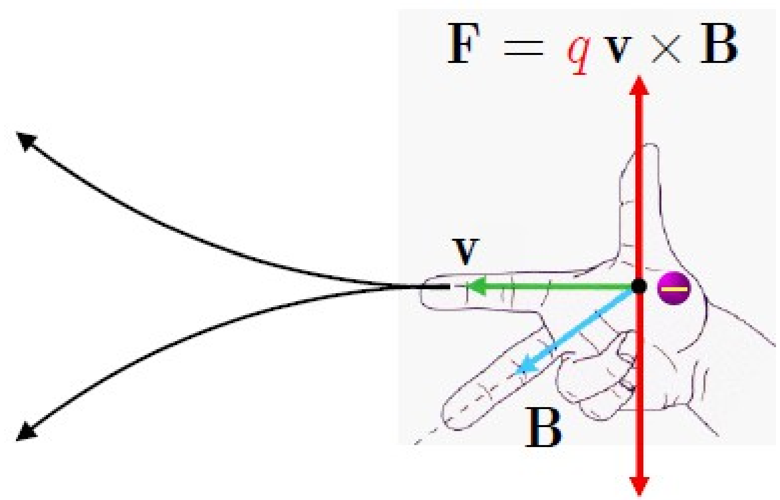
- A block of 128 microstrips is read-out by a single chip
- This chip sends as output a *data-frame* (see figure)
- Fluctuating part: electronic noise
- Passage of a particle gives a signal that sticks out of that noise: a *hit*
- (Can also give signal in a few consecutive strips; they get clustered into a single *hit*)
- From then on, we only process the *hits* and ignore other microstrips
- This is just an example of *data reduction*



Data-frame; each bin between header and tick mark corresponds to the position of a strip

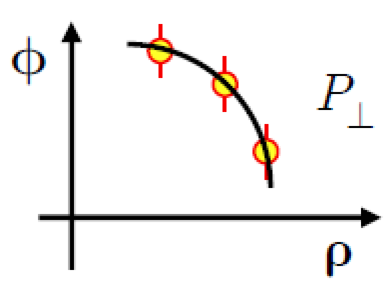
More data reduction: Tracking



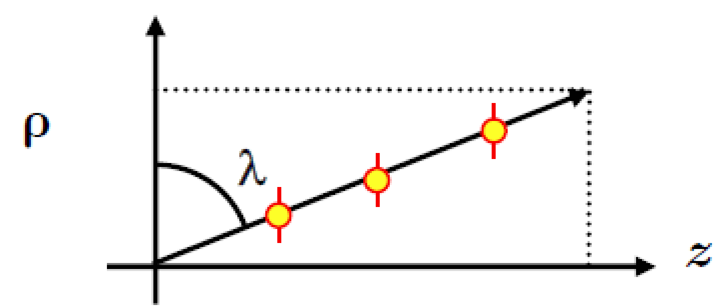


Solenoidal field along z: deflection in x-y (or $\rho-\phi$) plane

We sample the trajectory in a discrete number of crossings with the detector; from those crossings we must infer the trajectory

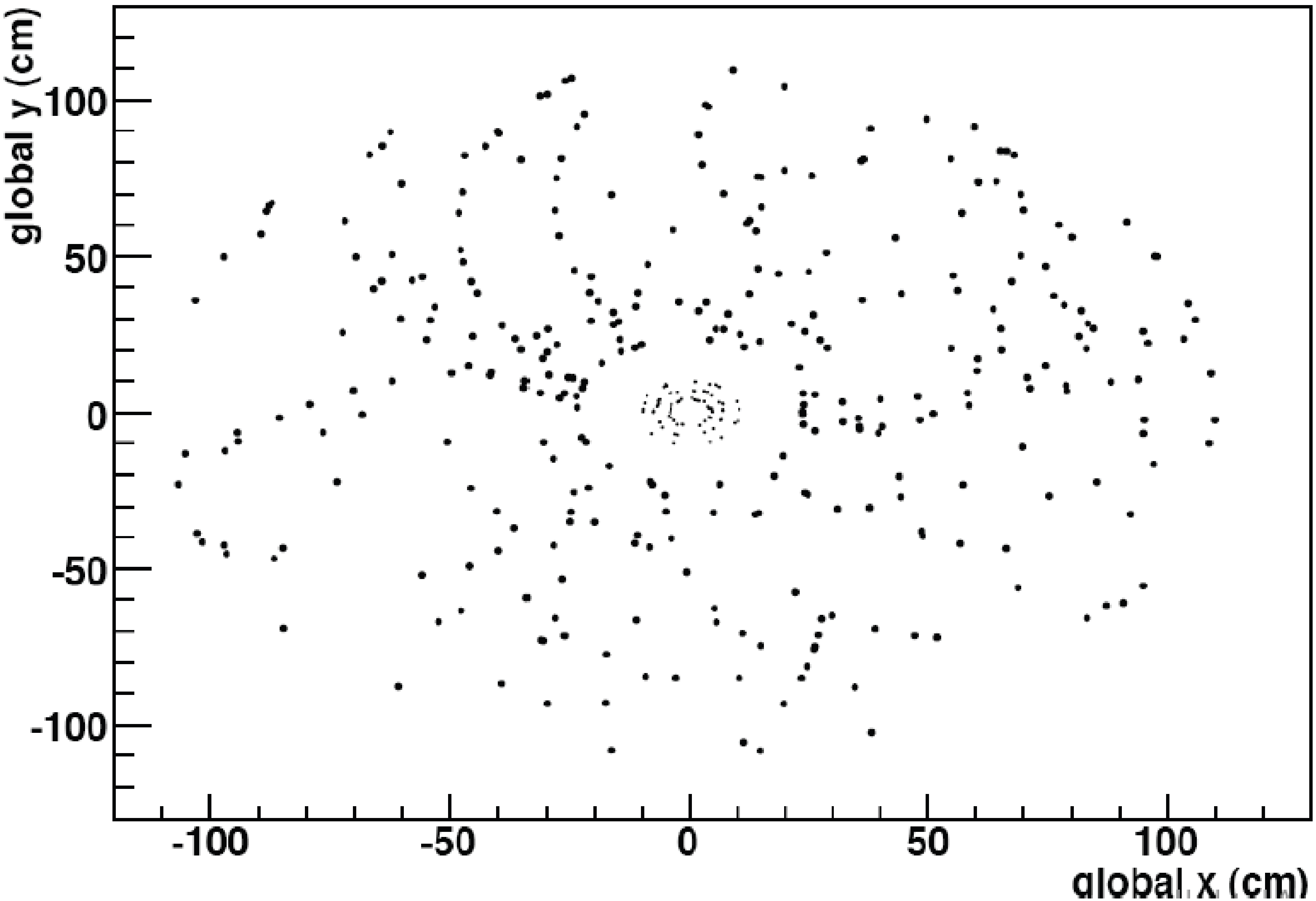


$$P_{\perp} = P \cos\lambda = 0.3BR$$



Find the track

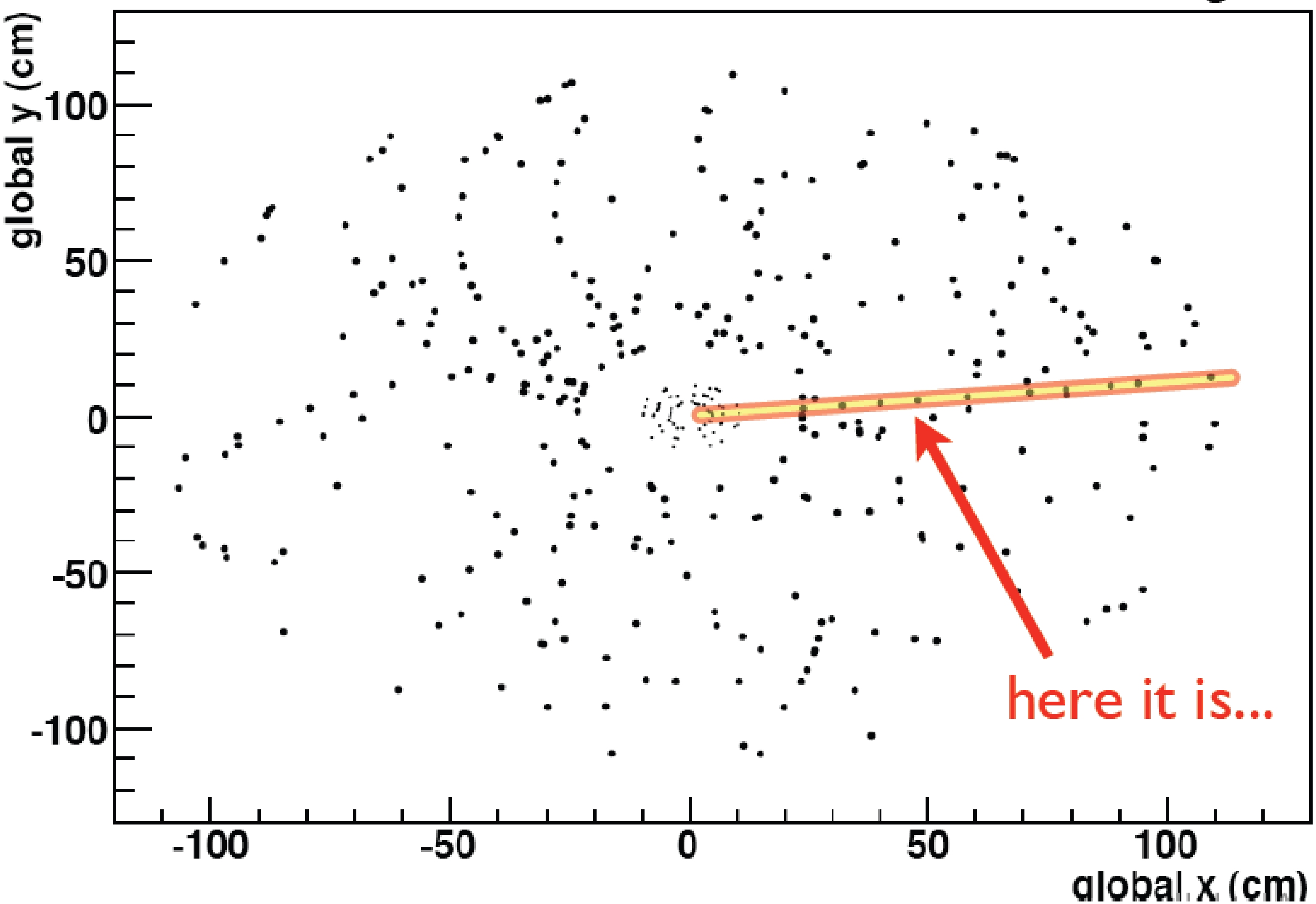
cf Aaron Dominguez



Where is the 50 GeV track? (Hint: it is very straight)

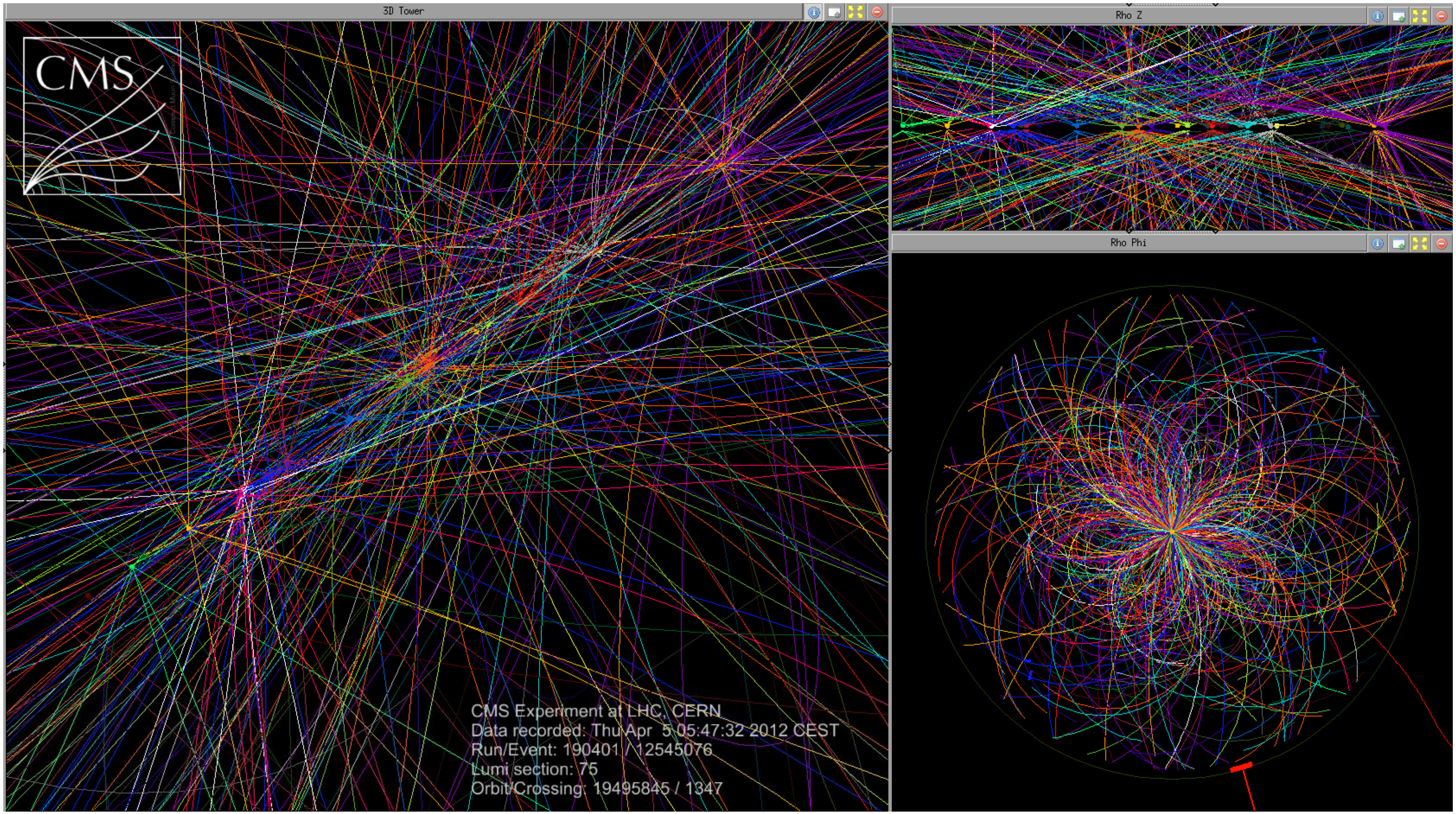
Find the track

cf Aaron Dominguez



These data are from Tevatron, a past accelerator operating at ~1/7 of LHC energy

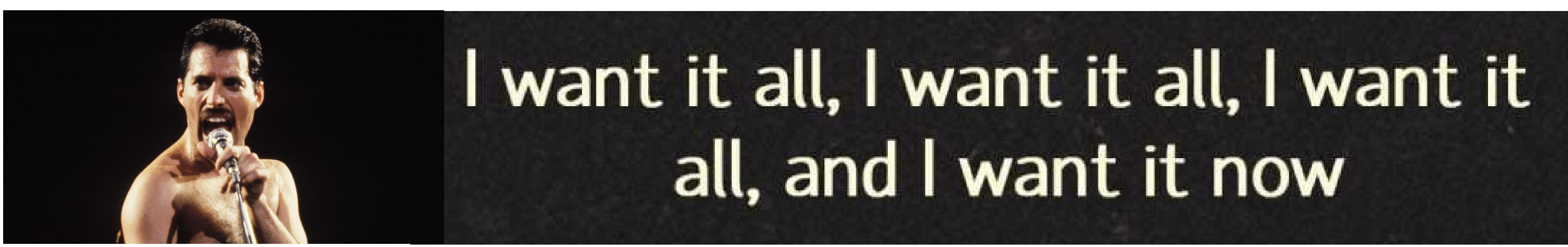
Tracking at LHC



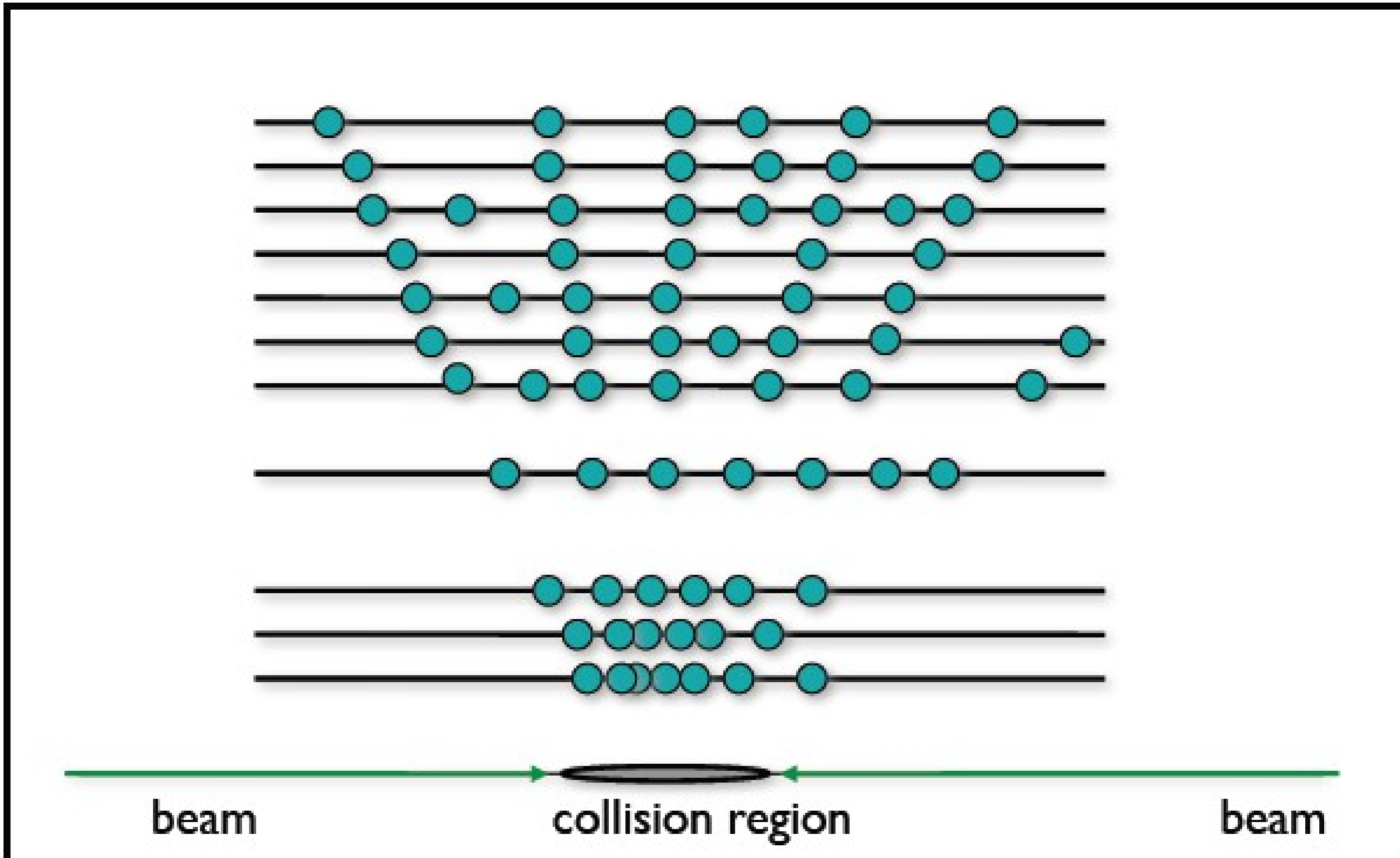
LHC achieves large intensities by very dense proton bunches (large number of protons, small volume) ⇒ several proton-proton interactions during each bunch crossing (*pileup*)

What we need

- We need track-finding to be *efficient*
  - We cannot afford to miss many tracks
- We need the track sample to be very *pure*
  - We cannot afford to contaminate the sample with many fakes (i.e., wrong hit combinations)
- And it has to be fast
- To summarize:

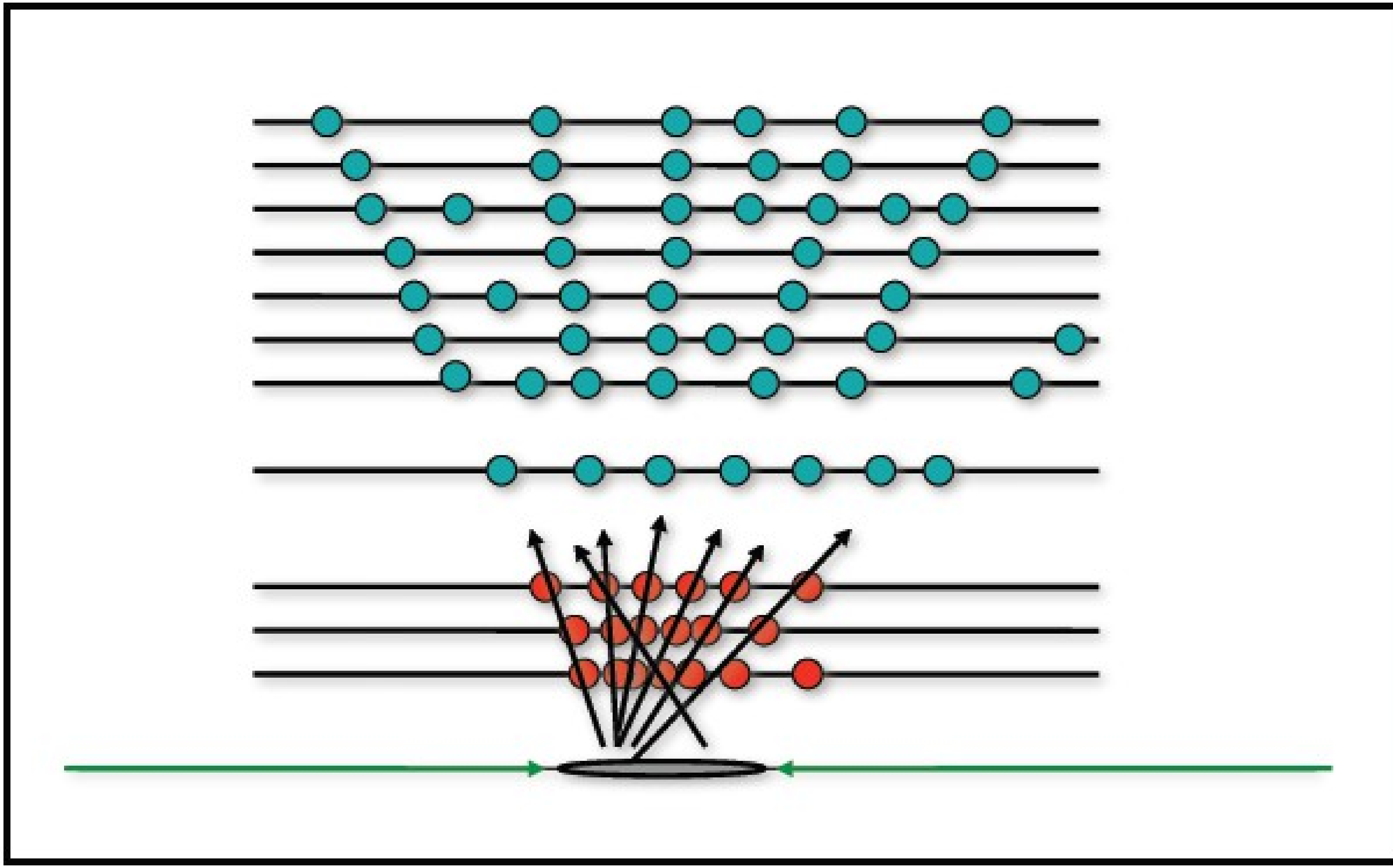


After local data reduction



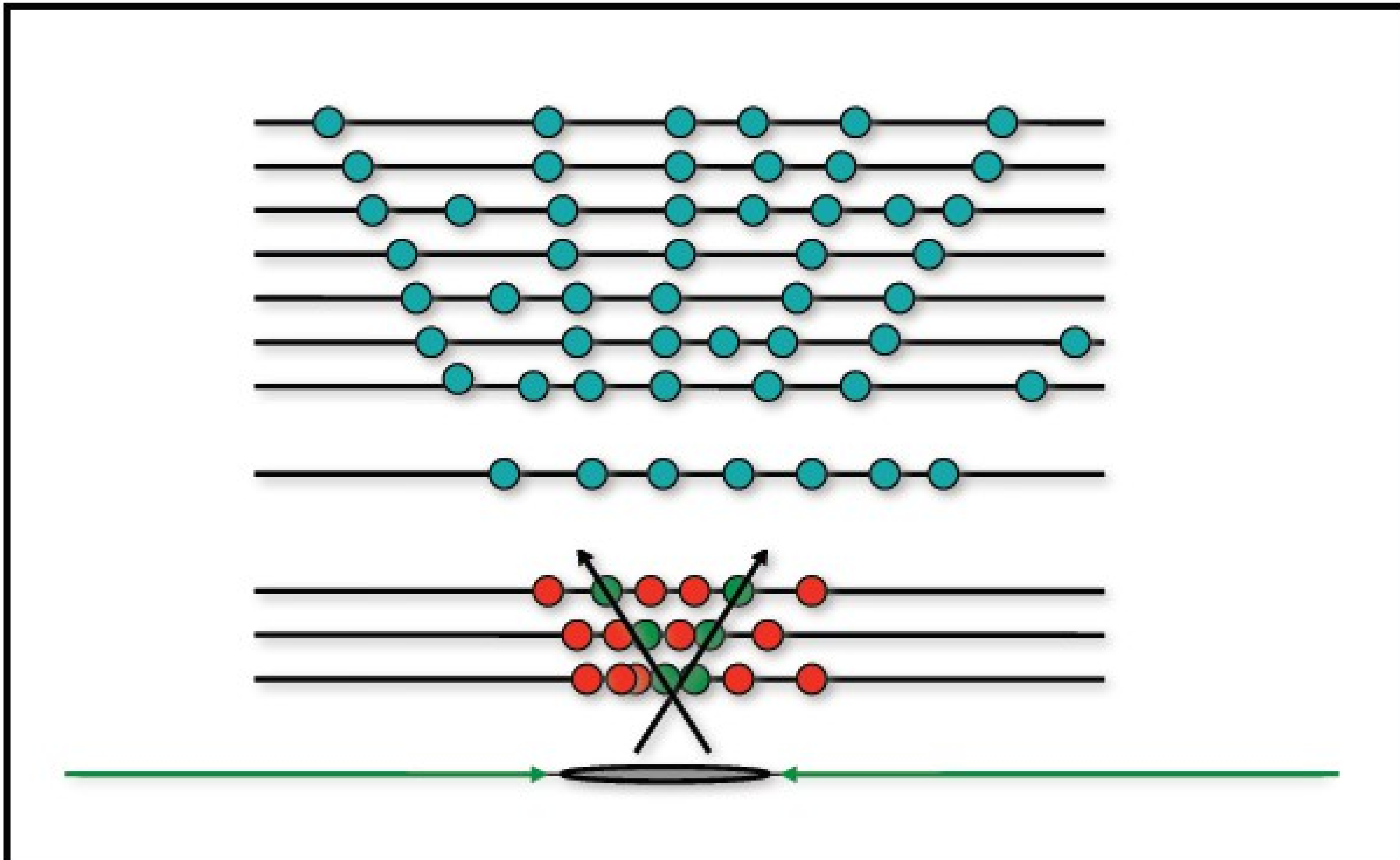
We start from a collection of hits, associated to a position and an uncertainty

Seeding



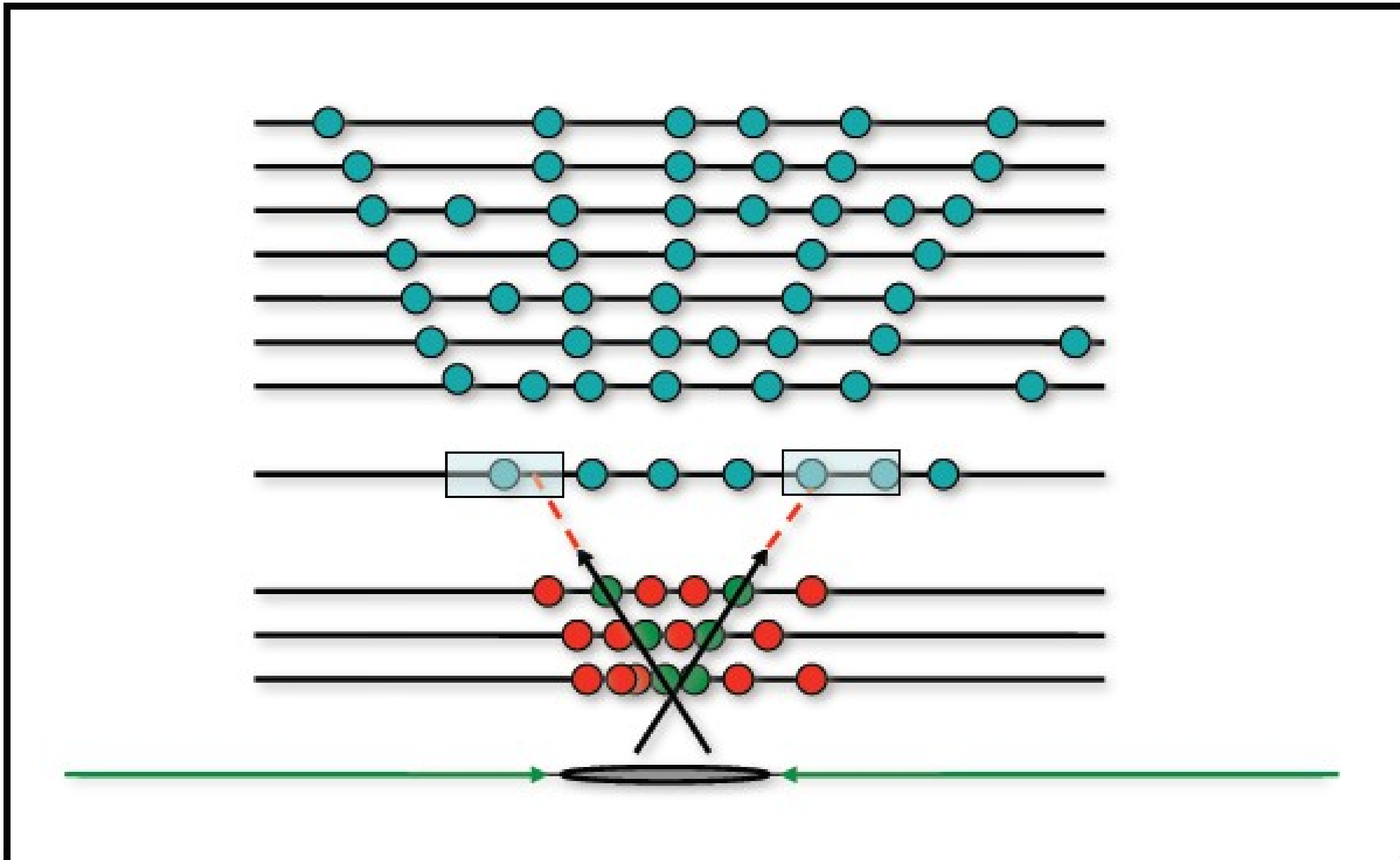
Fast fit to get initial trajectory, trying all combinations of hits in a small subset of layers

Trajectory building



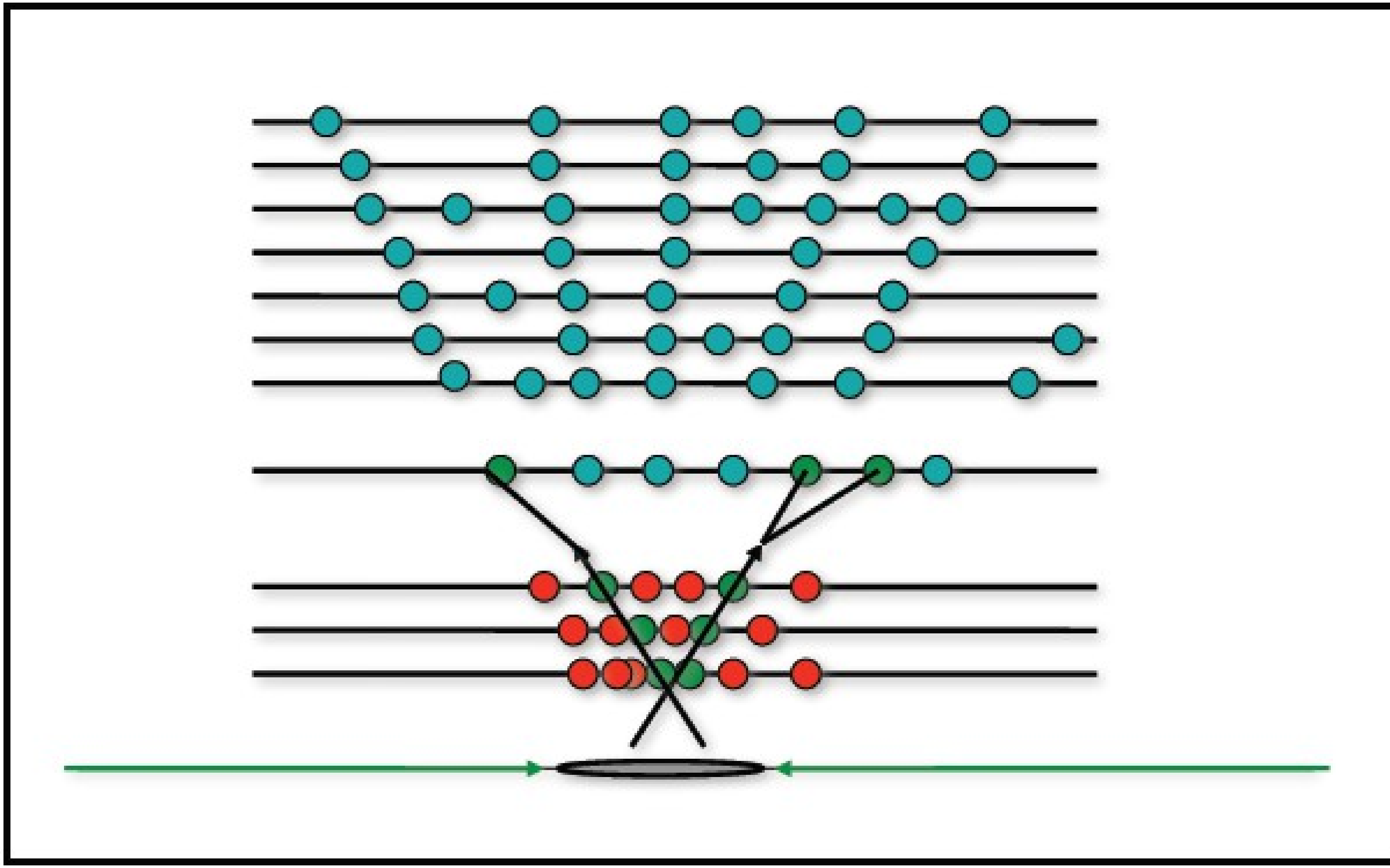
For illustration, let's consider these two seeds and let's see how trajectories are built from there.

Trajectory building

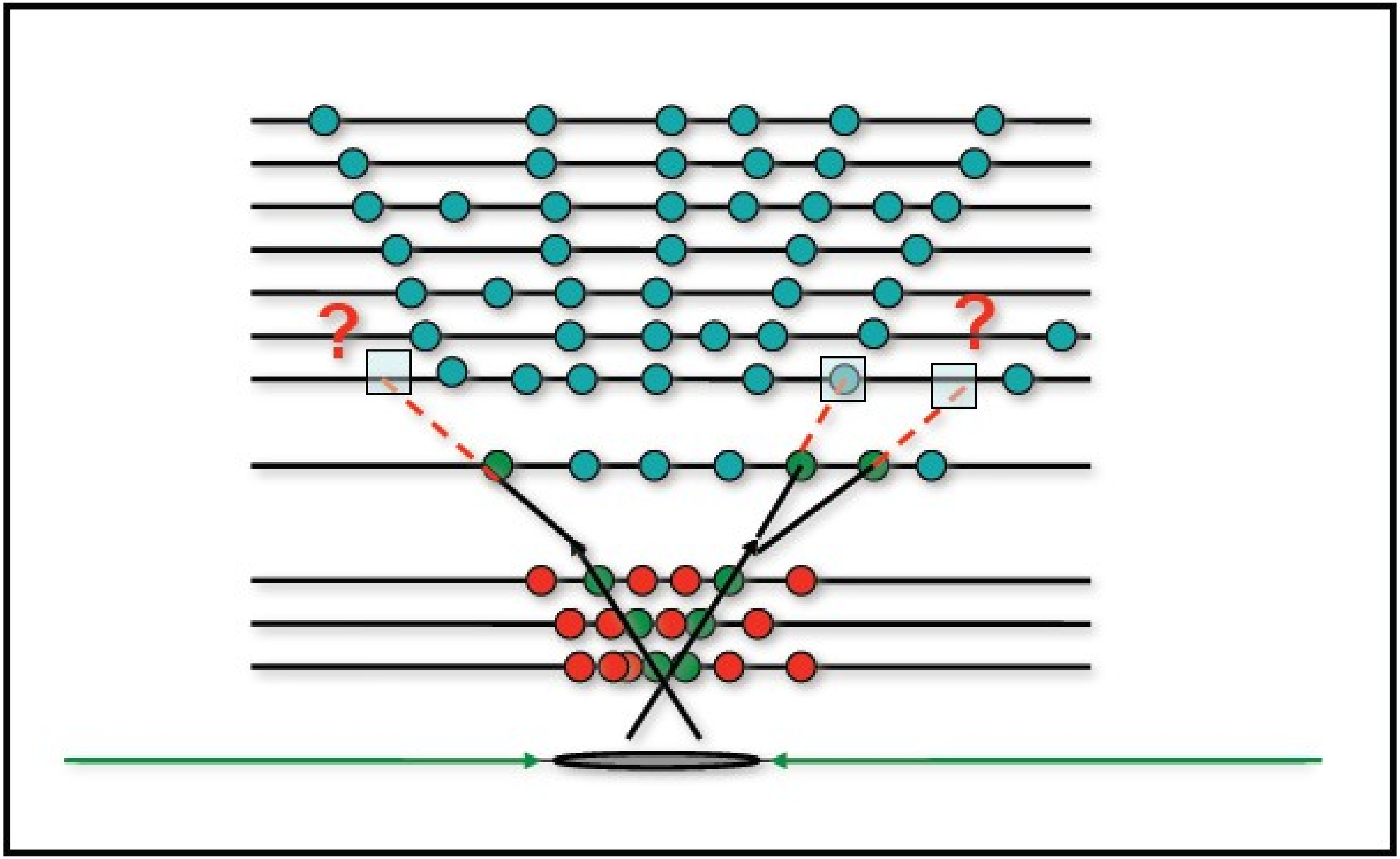


Trajectory is propagated from layer to layer taking into account the uncertainties on the hit positions, energy loss, multiple scattering.
The Kalman Filter technique is employed in CMS
(I have backup slides if you are interested)

Trajectory building

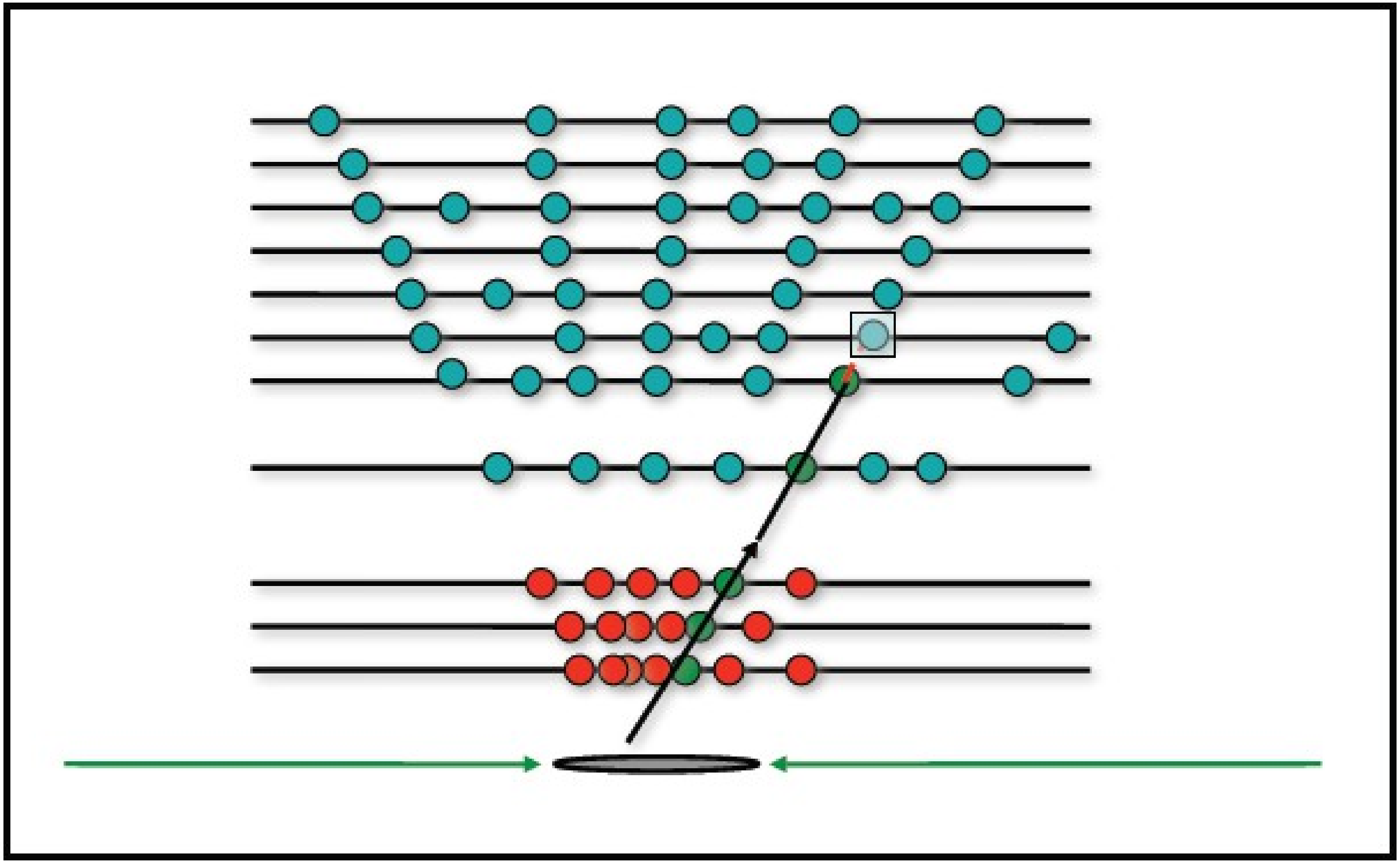


Trajectory building

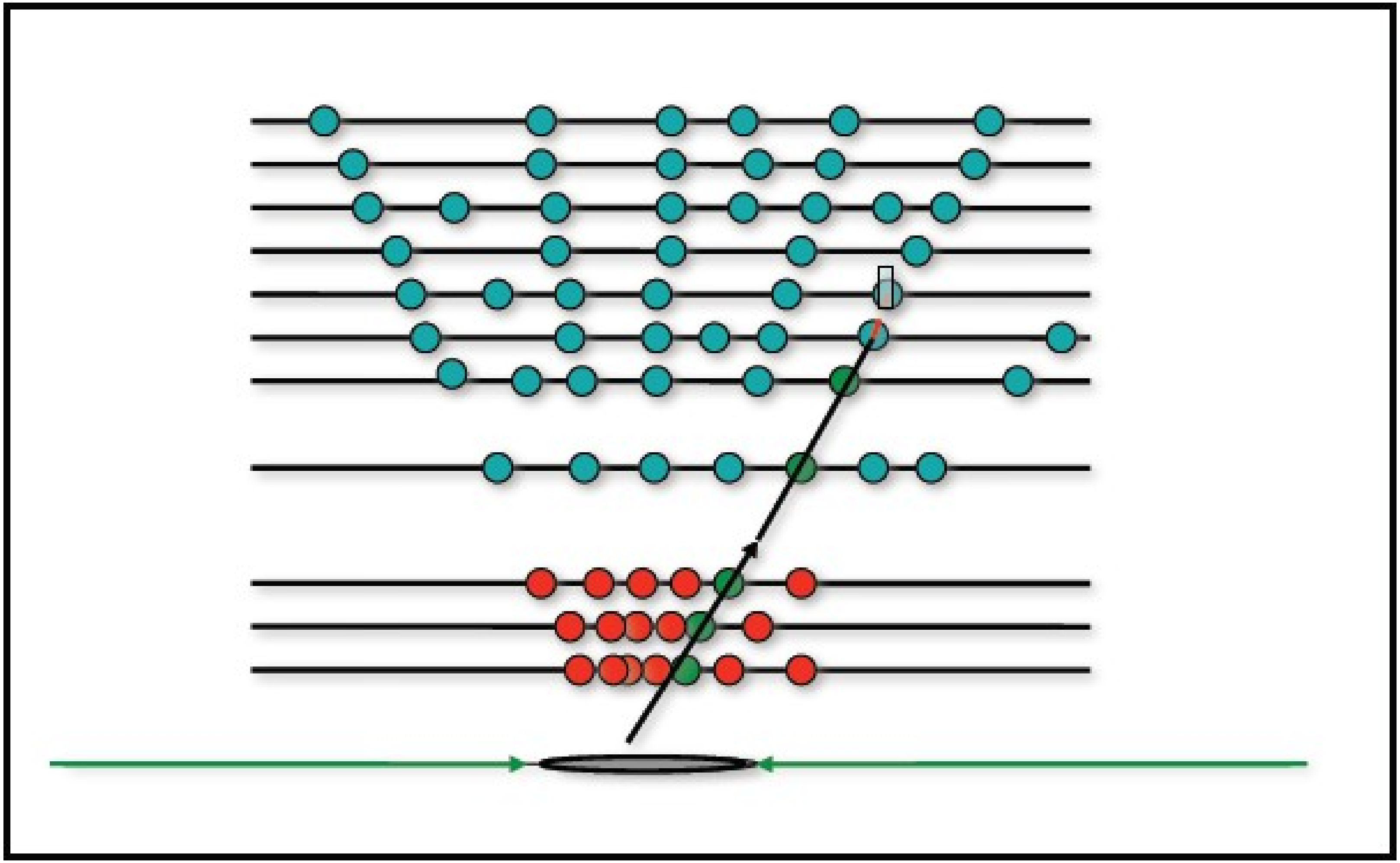


When no hits are found, track is probably fake

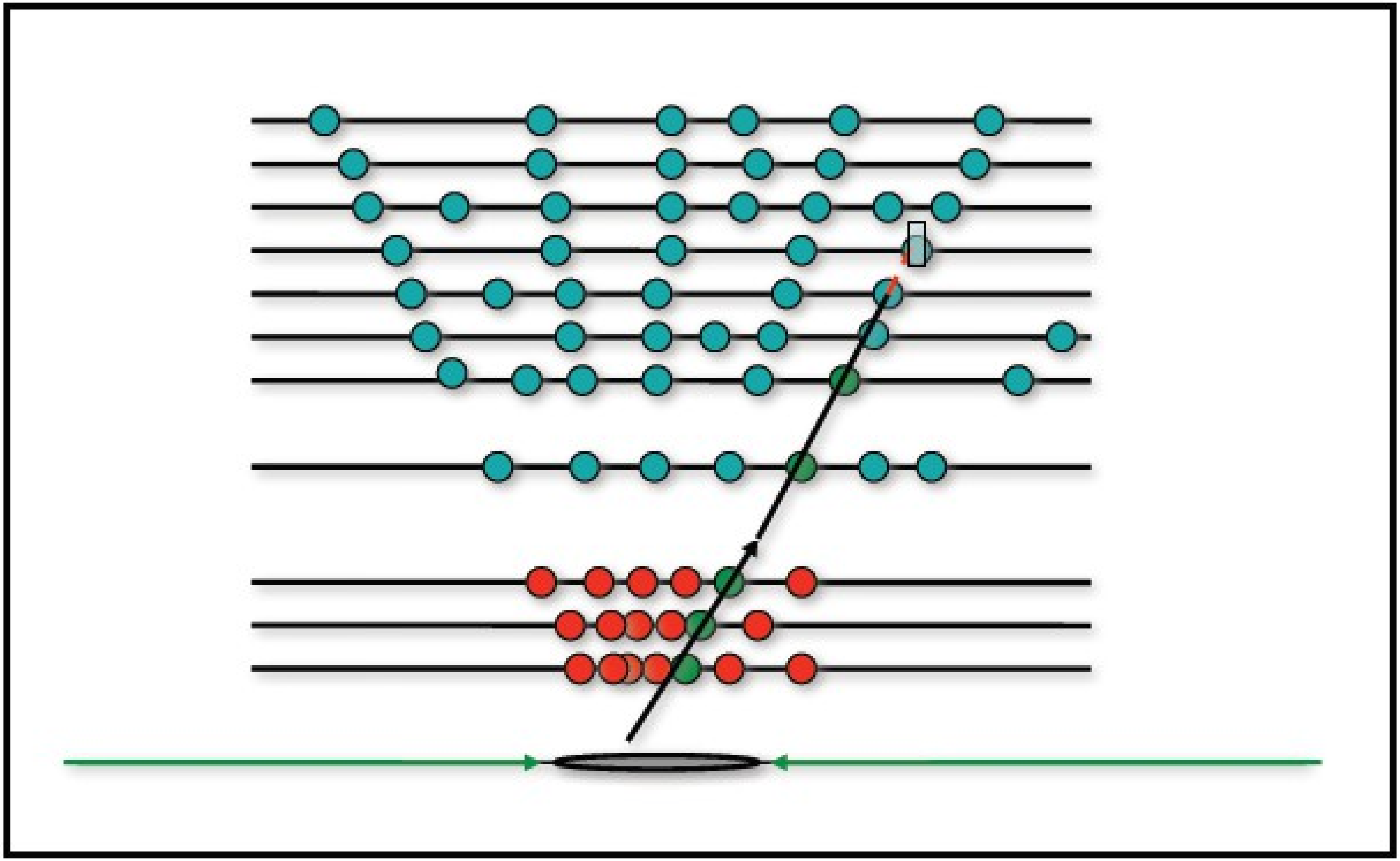
Trajectory building



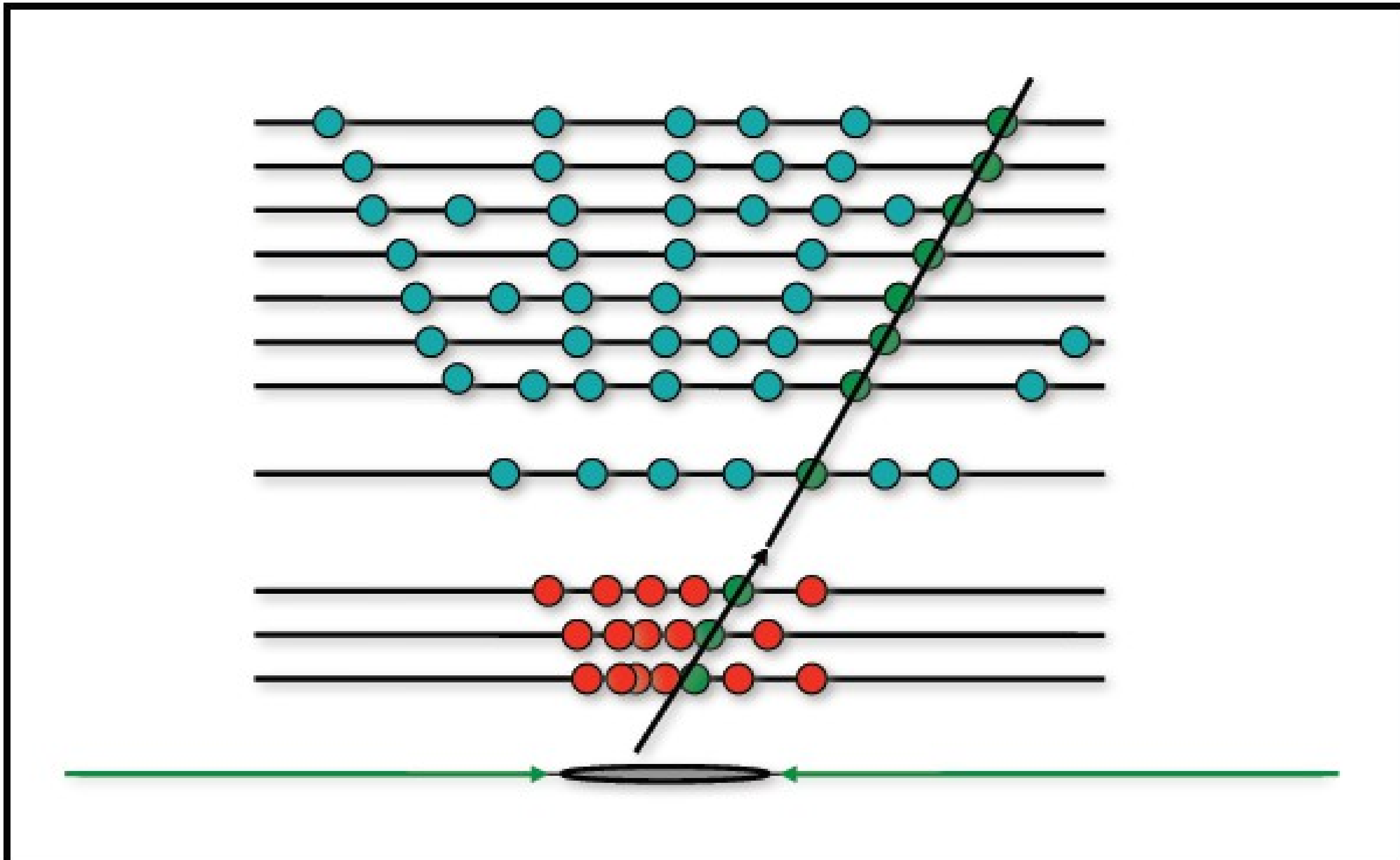
Trajectory building



Trajectory building

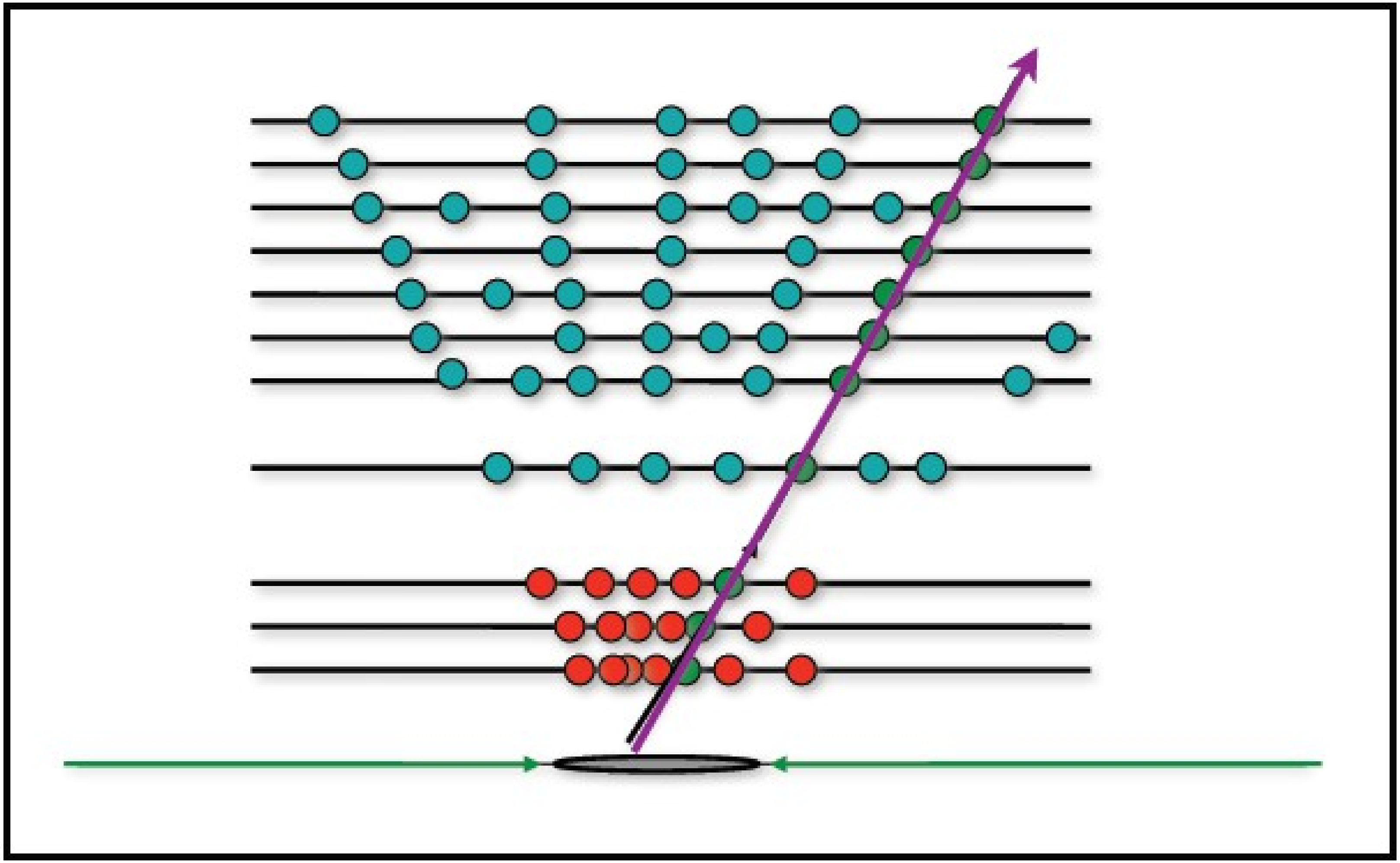


Trajectory building



Now we have a track, and a preliminary estimate of its parameters; but this estimate can be biased by the constraints that we applied to reduce combinatorics, hence a final fit must be done.

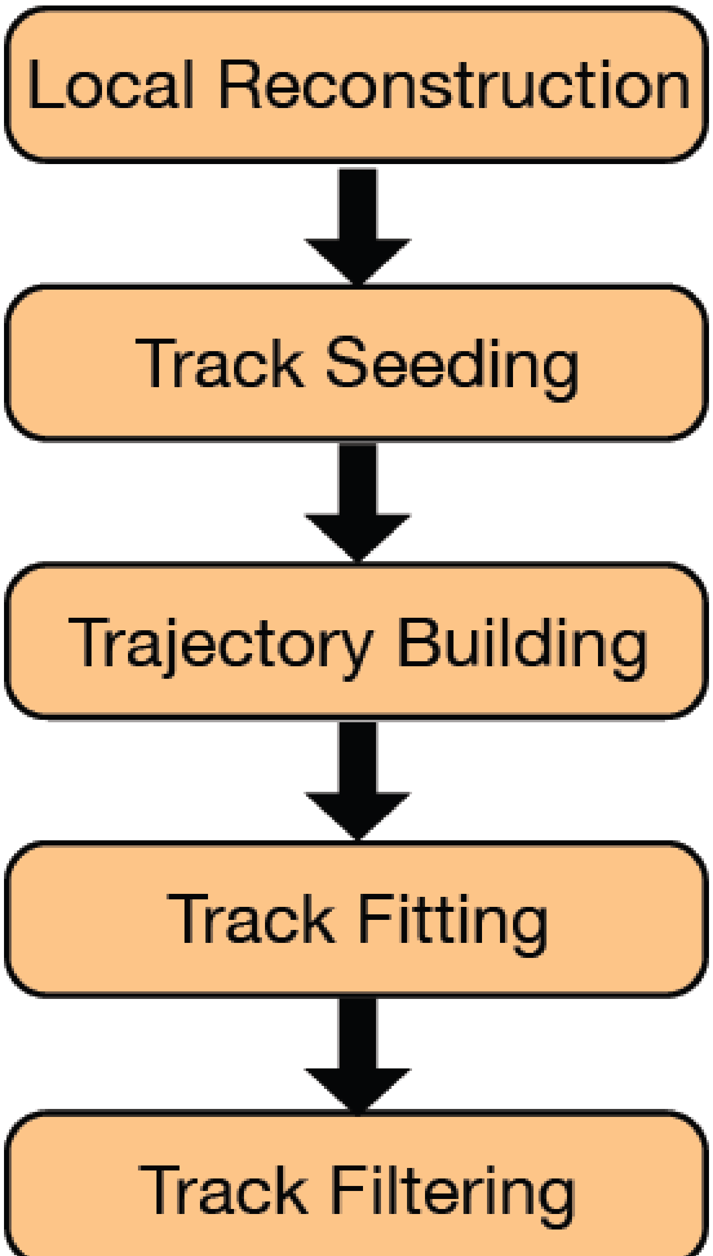
Track fitting



Final fit to the hits, to get a better estimate of the trajectory

If χ^2 of the fit is poor, ignore this track

In summary



Channels (strips, pixels, …) giving signal are clustered into “**hits**”

A minimal number of hits (or, in special cases, information from another detector) is used for an **initial estimate of track direction**

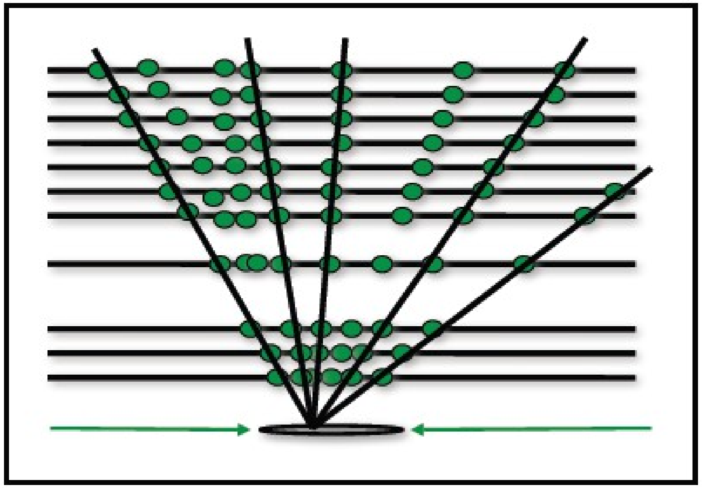
Pattern recognition step: all available hits are used to infer the particle trajectory

Final estimate of the track parameters using the full set of associated hits

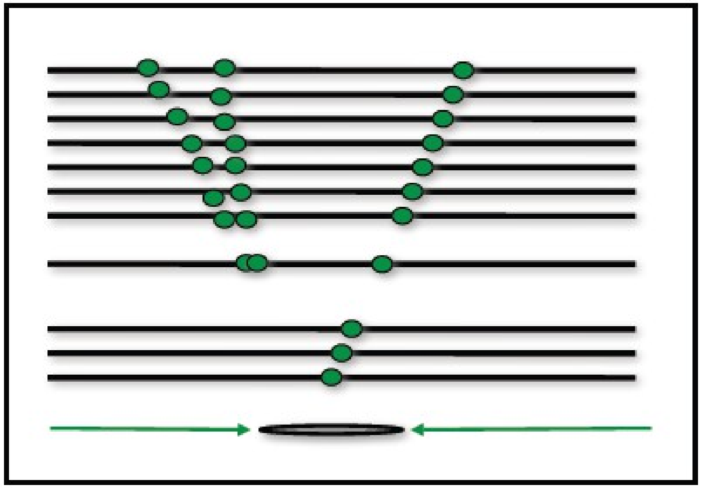
Removal of low-quality tracks, likely to be fakes

Iteratively

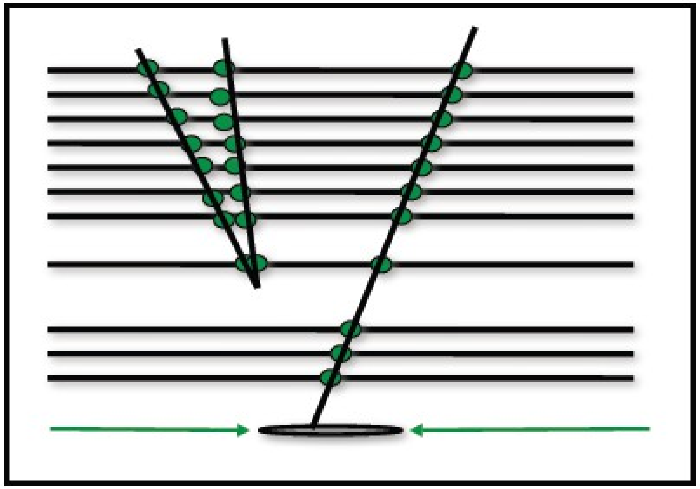
N-th step:



Remove associated hits:

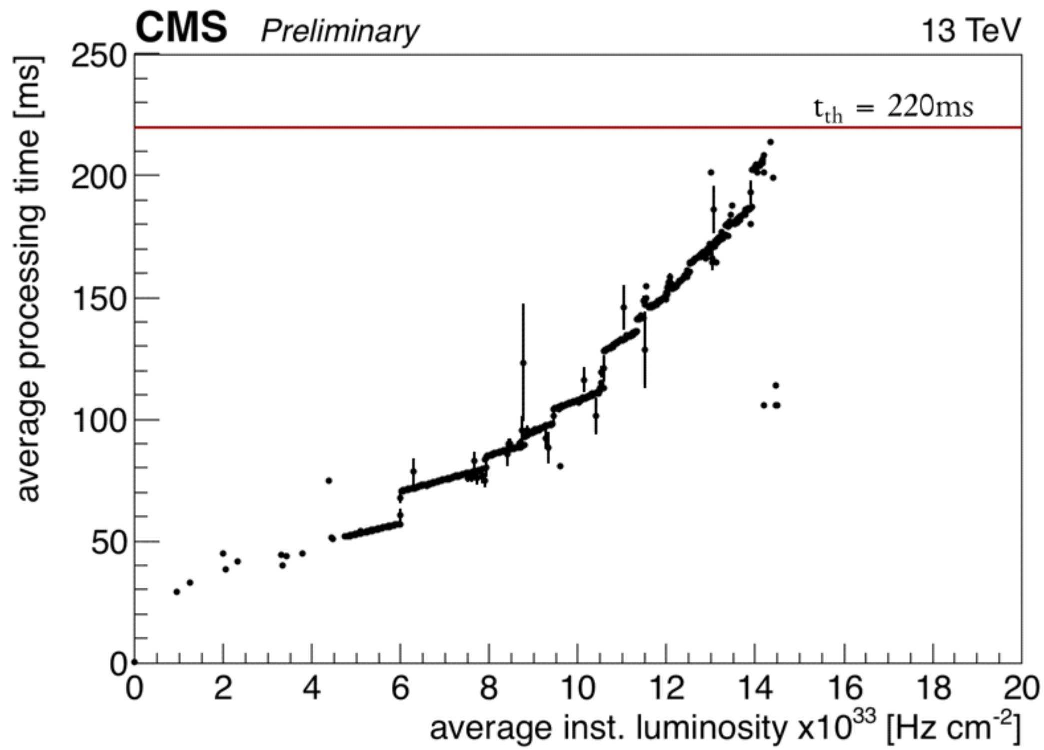


(N+1)-th step:



So far so good, but...

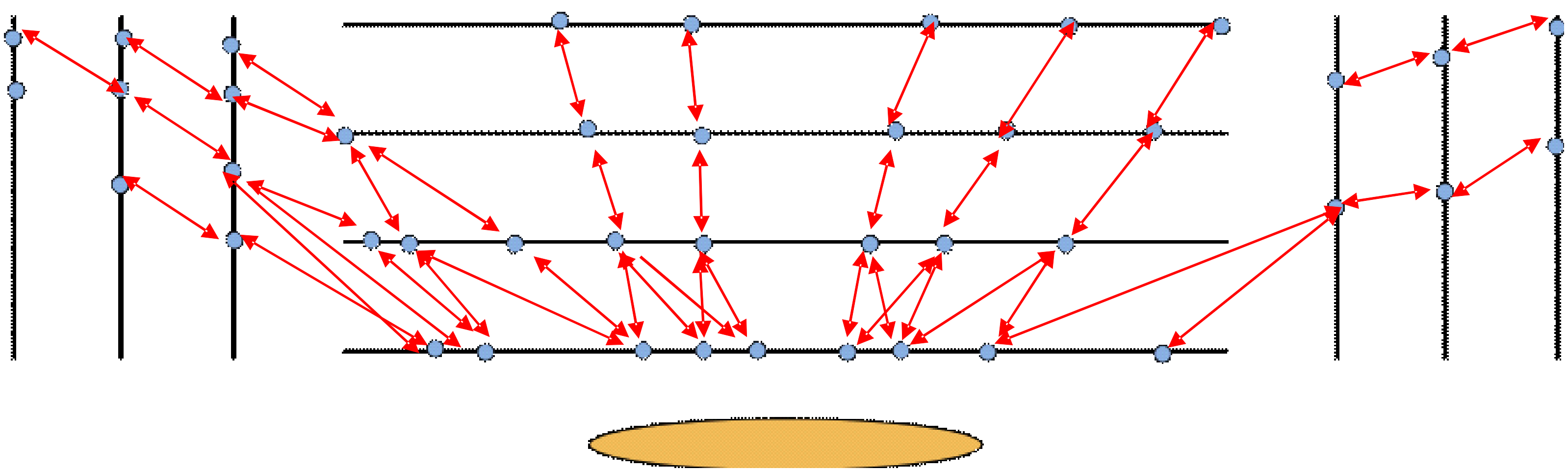
- Run II of LHC (started in 2015): large jump in collision frequency, more pileup, also more particles produced at each interaction because of larger collision energy
- And it will become much much worse in the future runs of LHC (dubbed High-Luminosity LHC)
- The "seeding step", in particular, scales very badly with increasing multiplicities
- And since 2017 we have one more layer in the detector, making combinatorics worse



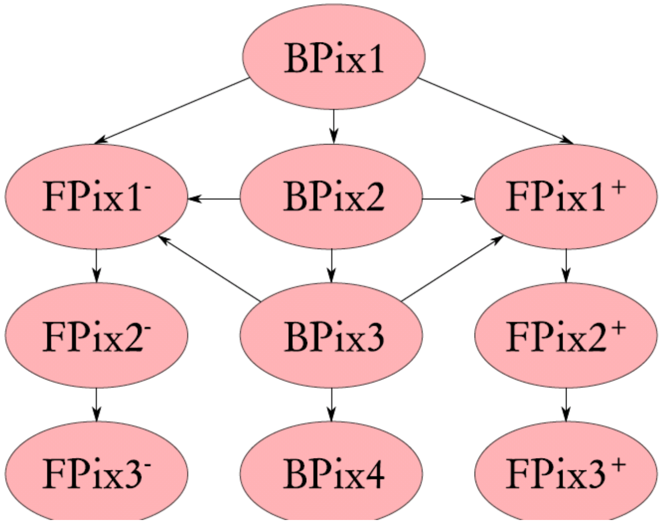
Plot for the High Level Trigger (HLT) during 2016 data taking.
Timing is dominated by tracking, although HLT runs a simplified version.
HL-LHC: 10 time more luminosity

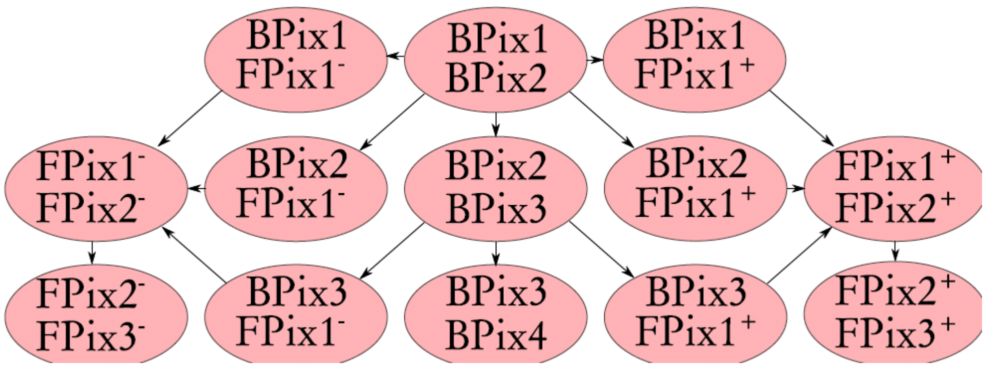
Cellular Automata (CA)

- Solution for seeding, chosen by CMS starting from 2017 operations
- A graph of all the possible connections between layers is created
- Doublets ("cells") are created for each pair of layers
- Fast computation of the compatibility between two connected cells
- No knowledge of the world outside adjacent neighboring cells required, making it easy to parallelize

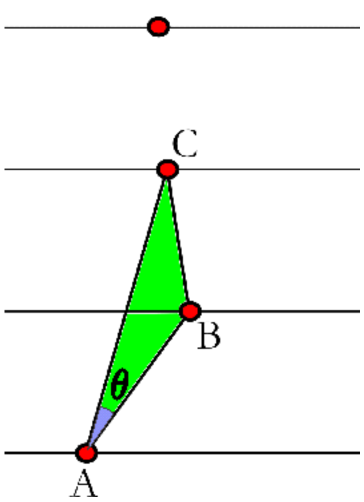


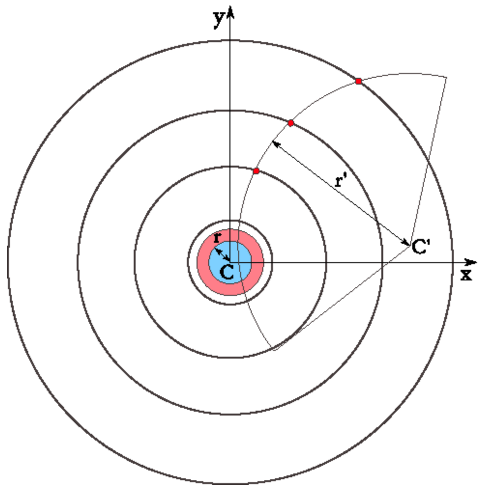
Cellular Automata (CA)





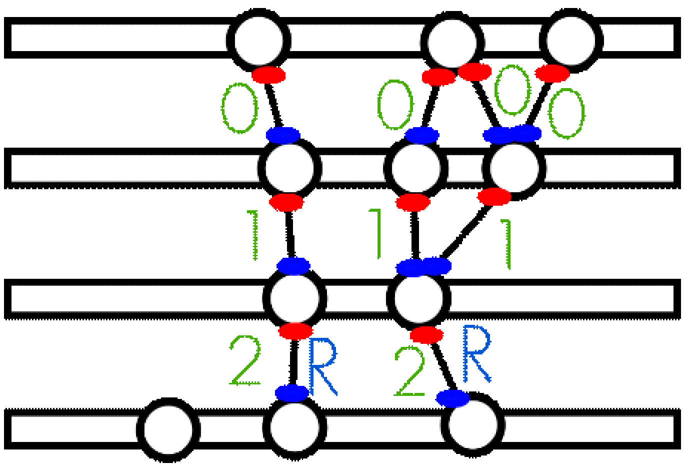
Degree of compatibility between hits is checked in r-z and x-y views:





Cellular Automata (CA)

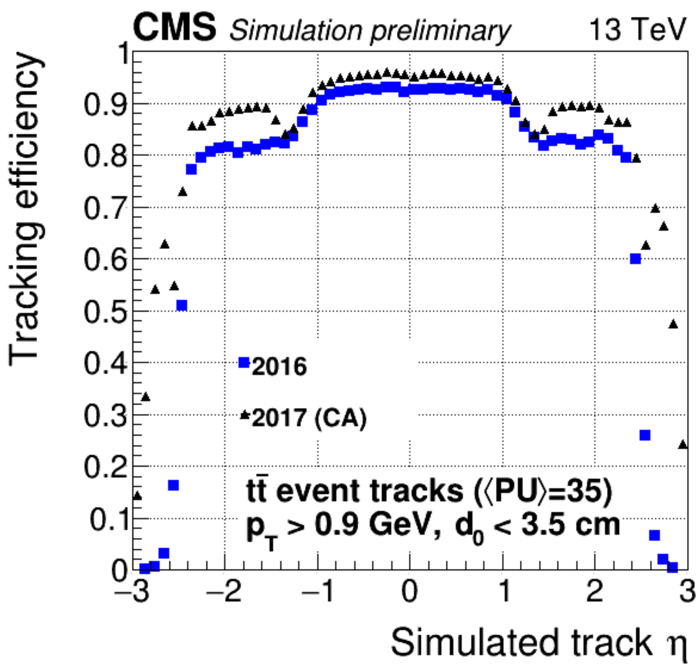
- If two cells satisfy all the compatibility requirements they are said to be neighbors and their state is set to 0
- In the evolution stage, their state increases in discrete generations if there is an outer neighbor with the same state
- At the end of the evolution stage the state of the cells will contain the information about the length
- If one is interested in quadruplets, there will be surely one starting from a state 2 cell, pentuplets from state 3, etc.

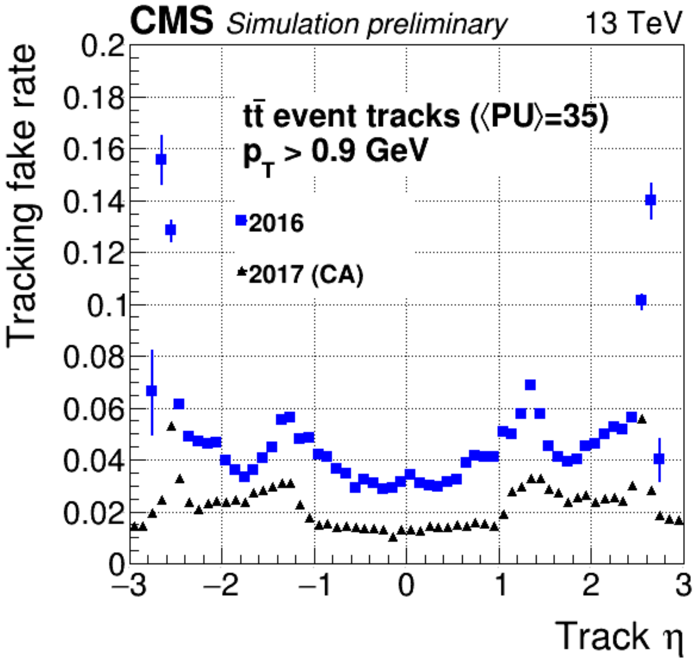


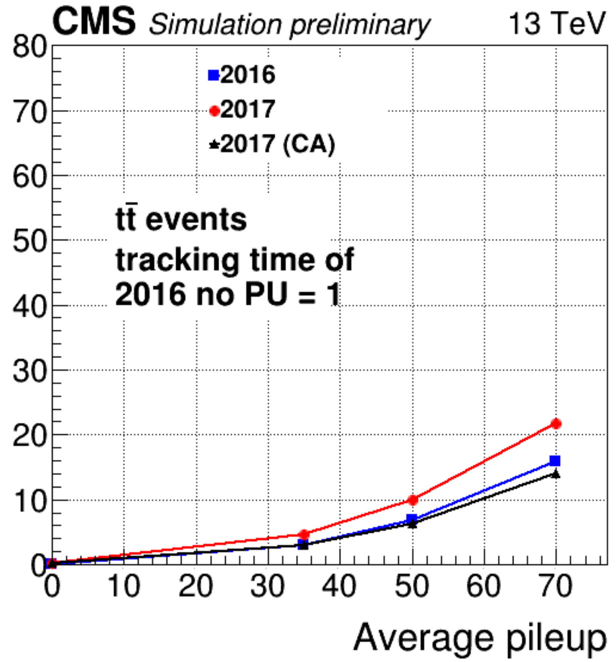
What we need, and where we stand

- We need track-finding to be efficient
- We need the track sample to be very pure
- And it has to be fast

Automaton works well

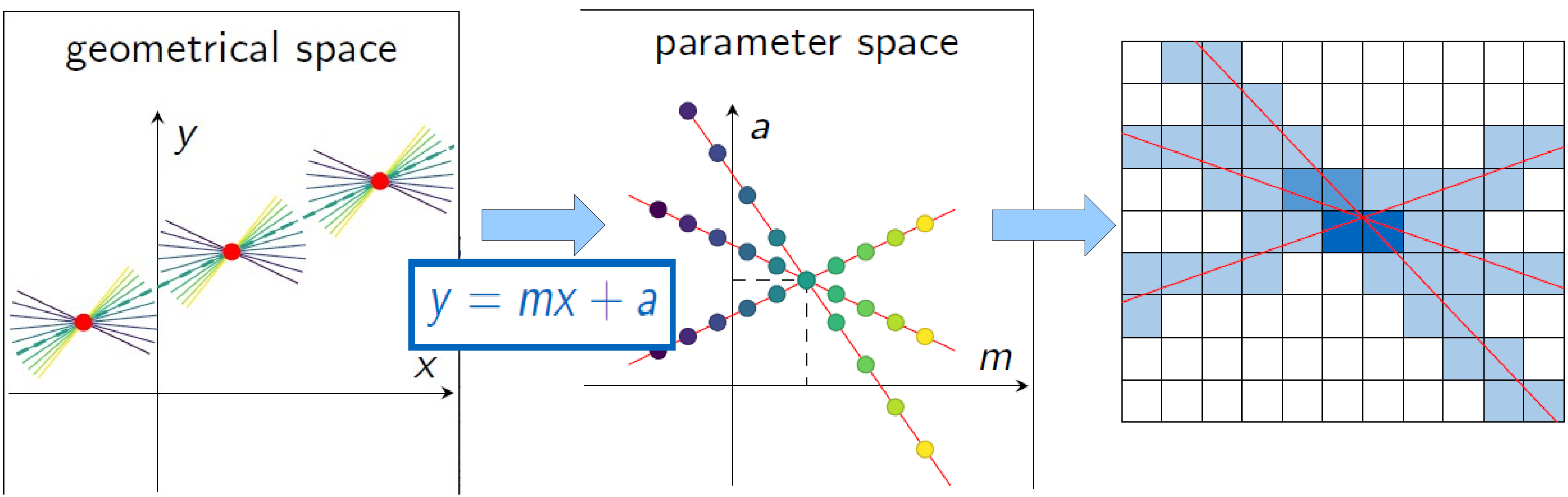






Hough Transform

- The methods seen so far are all *local*: a global fit would be better but too slow: very large combinatorics of hits, and very large error matrices because errors are correlated across the whole trajectory
- Novel idea: use Hugh Transforms in tracking, following example from digital image processing



Points are hits;
Lines are possible trajectories;
Dashed line is true one

Points are trajectories;
Point where the lines intersect gives the best-fit trajectory

Reduce dimensionality by binning and choose by majority vote

Resolving ambiguity: Graphs

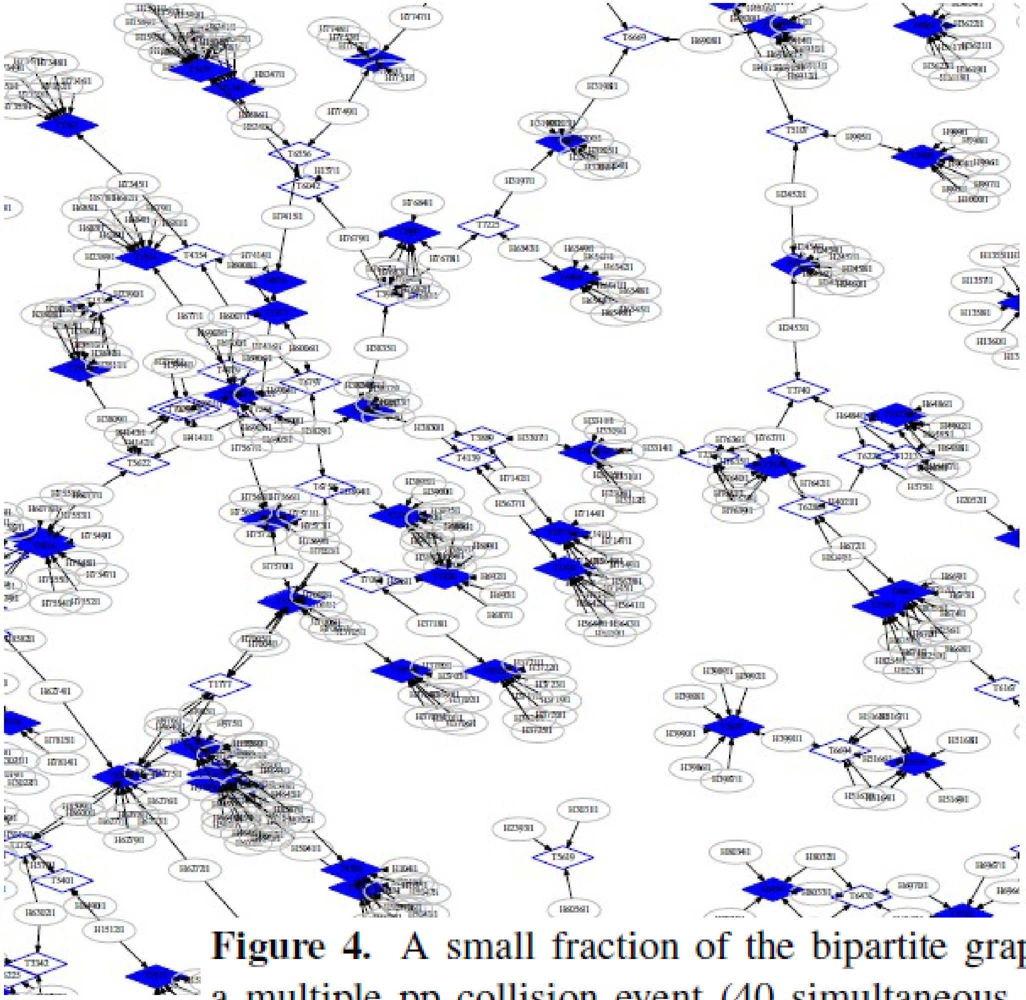


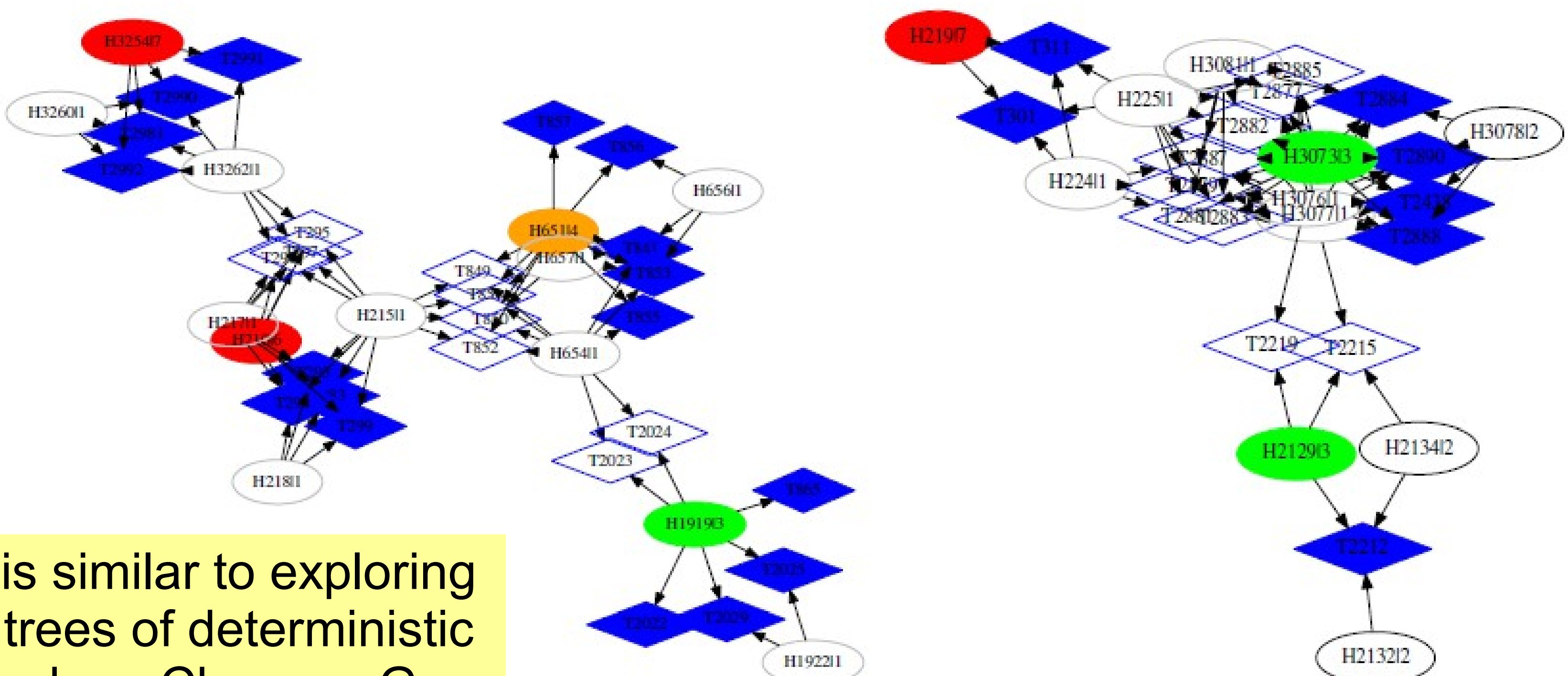
Figure 4. A small fraction of the bipartite graph of hits (ellipses) and track candidates (blue diamonds) for a multiple pp collision event (40 simultaneous inelastic pp collisions). Directed arrows, graph edges, show potential hits-to-track candidate assignments.

- Several track candidates may share some hits (the larger the multiplicity, the more often that happens)
- Our goal is to assign all hits to at most one track
- Most methods are local; not easy to consider total merit of all tracks (sum of χ^2)
- To keep total CPU time manageable, recent proposal based on graph theory

Ferenc Sikler, proceedings in preparation from this talk:
https://indico.cern.ch/event/577003/contributions/2415235/attachments/1424172/2183976/sikler_denseTracking_ctdwit17.pdf

Resolving ambiguity: Graphs

Figure 5. Example minigraphs obtained after the removal of all bridges and articulation points from a bipartite graph of hits and track candidates, in the case of a multiple pp collision event (40 simultaneous inelastic pp collisions). The colors of contracted hits (ellipses) refer the number of hits with identical role (red – 6 or more, orange – 4 or 5, green – 3, white – 1 or 2). Filled track candidates (blue diamonds) indicate true tracks, while the others (open diamonds) show candidates where one or more hits are not in place.



Process is similar to exploring decision trees of deterministic games such as Chess or Go

Ferenc Sikler, proceedings in preparation from this talk:
https://indico.cern.ch/event/577003/contributions/2415235/attachments/1424172/2183976/sikler_denseTracking_ctdwit17.pdf

Further data reduction: from particles to "jets"

Quarks are never naked, but we need to undress them

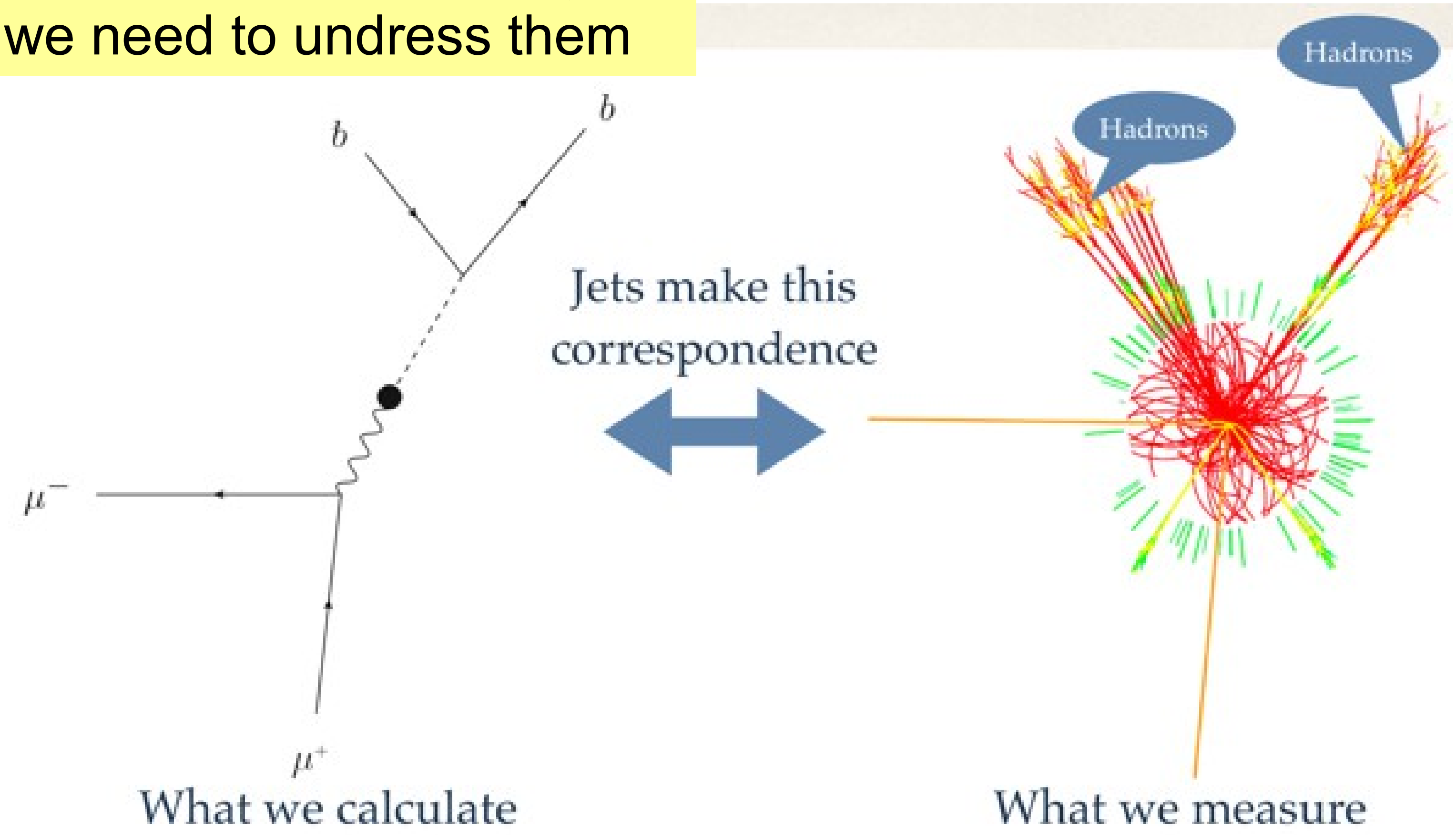
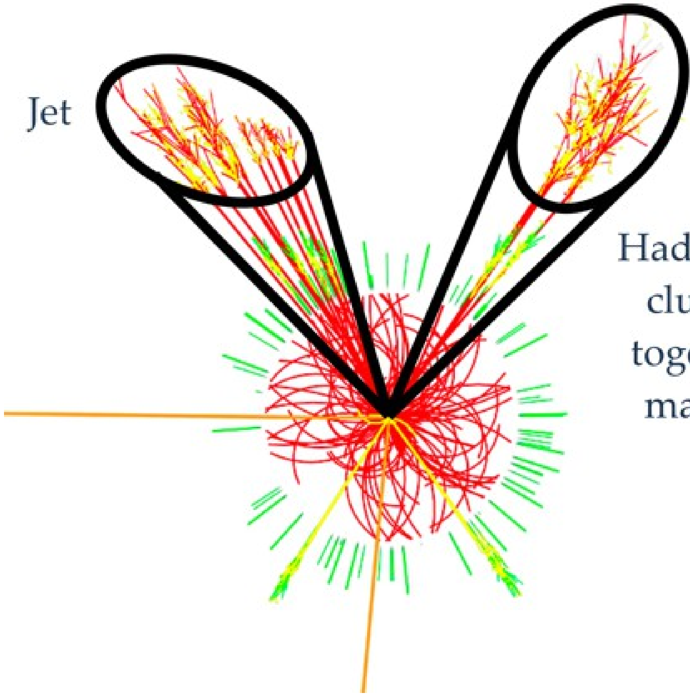


Illustration credit: Flip Tanedo

How to build jets

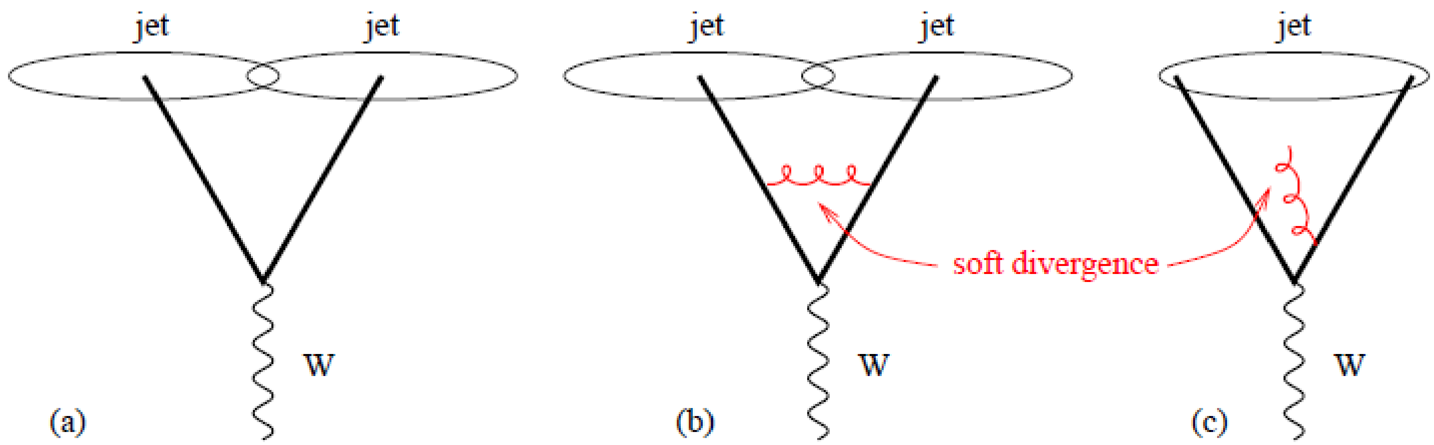


Two popular ways:

- Cone-based algorithms:
  - Find a set of seeds (e.g., the highest-energy particles in the event)
  - Sum vectors of all particles in a cone around the seeds
  - Use those sum vectors as new seeds, and repeat until convergence
- Clustering algorithms:
  - Calculate distances d_{ij} (according to some metrics) between particles i and j, for all i,j, and distance d_{iB} between particle i and the beam axis
  - If $d_{ij}<d_{iB}$, combine i+j; else, call i a jet

Illustration credit: Flip Tanedo

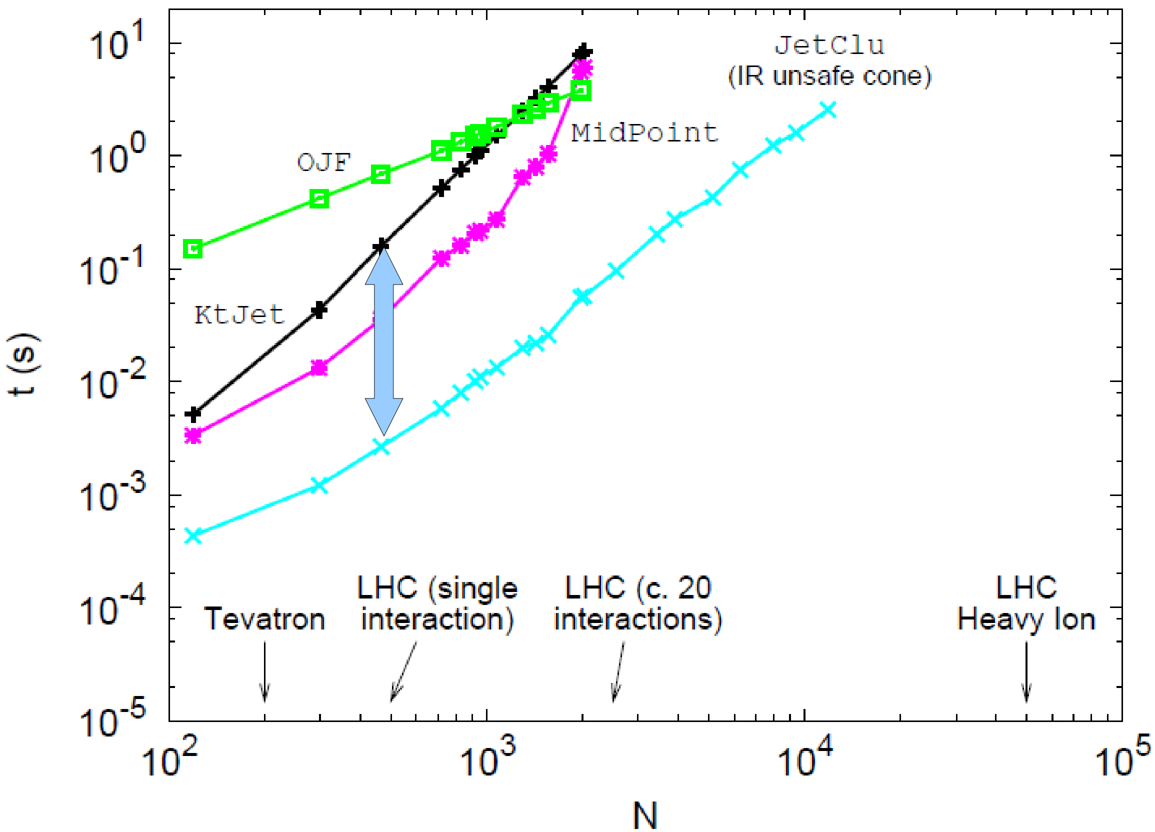
Infra-red (IR) stability



From G.Salam, arXiv:0906.1833 [hep-ph], Eur.Phys.J.C67 (2010) 637

A jet algorithm is said to be IR-unstable if the addition of a low-momentum particle (with arbitrarily low momentum) can change the outcome of the jet finding, making the theory-experiment comparison quite ill-defined

Fast and wrong, or right and slow?



- The blue curve is for a cone algorithm
  - IR-unstable...
  - ...but a lot faster
- The black curve is for a clustering algorithm
  - IR-stable...
  - ...but much slower
  - Gets worse as N grows: finding minimal value of d_{ij}, d_{iB} is a $O(N^2)$ operation done N times
  - *(Really?)*

Jet finding with Voronoi cells

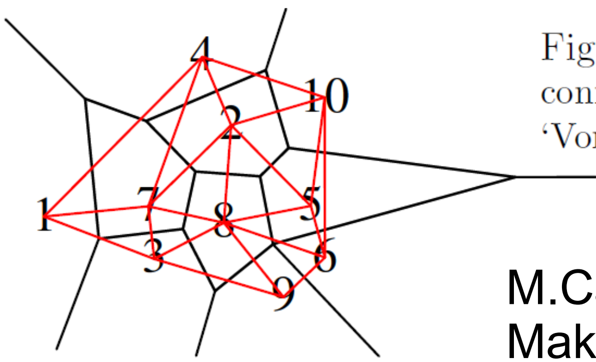
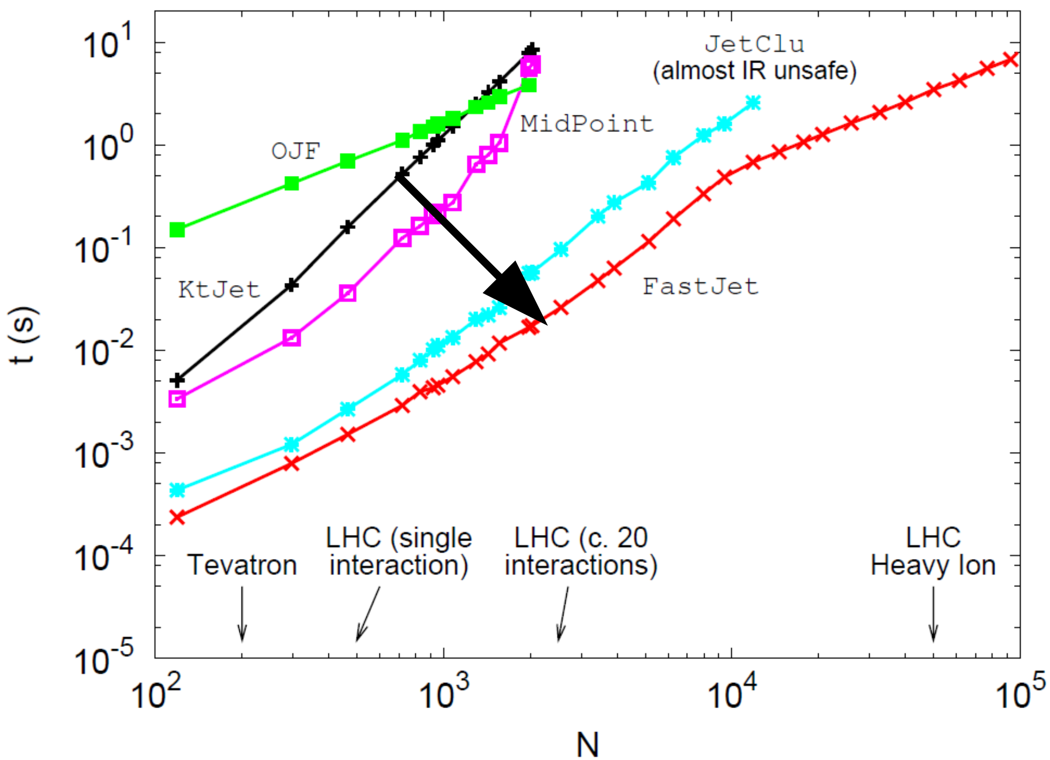


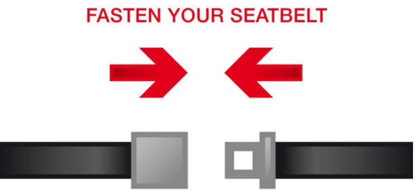
Figure 1: The Voronoi diagram for ten random points. The Delaunay triangulation (red) connecting the ten points is also shown. In this example the points 1, 4, 2, 8 and 3 are the 'Voronoi' neighbours of 7, and 3 is its nearest neighbour.

M.Cacciari, G.Salam, arXiv:hep-ph/0512210, Phys.Lett.B641 (2006) 57
Making use of work by Dirichlet (1850) and Voronoi (1908)

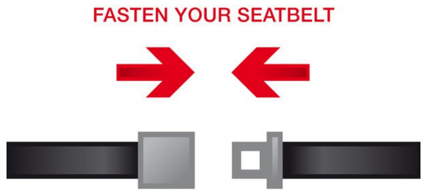


$O(N^3)$ became $O(N \ln N)$

Example of a *Paradigm Shift*: since a decade, nobody uses cone algorithms anymore



Higher level analysis

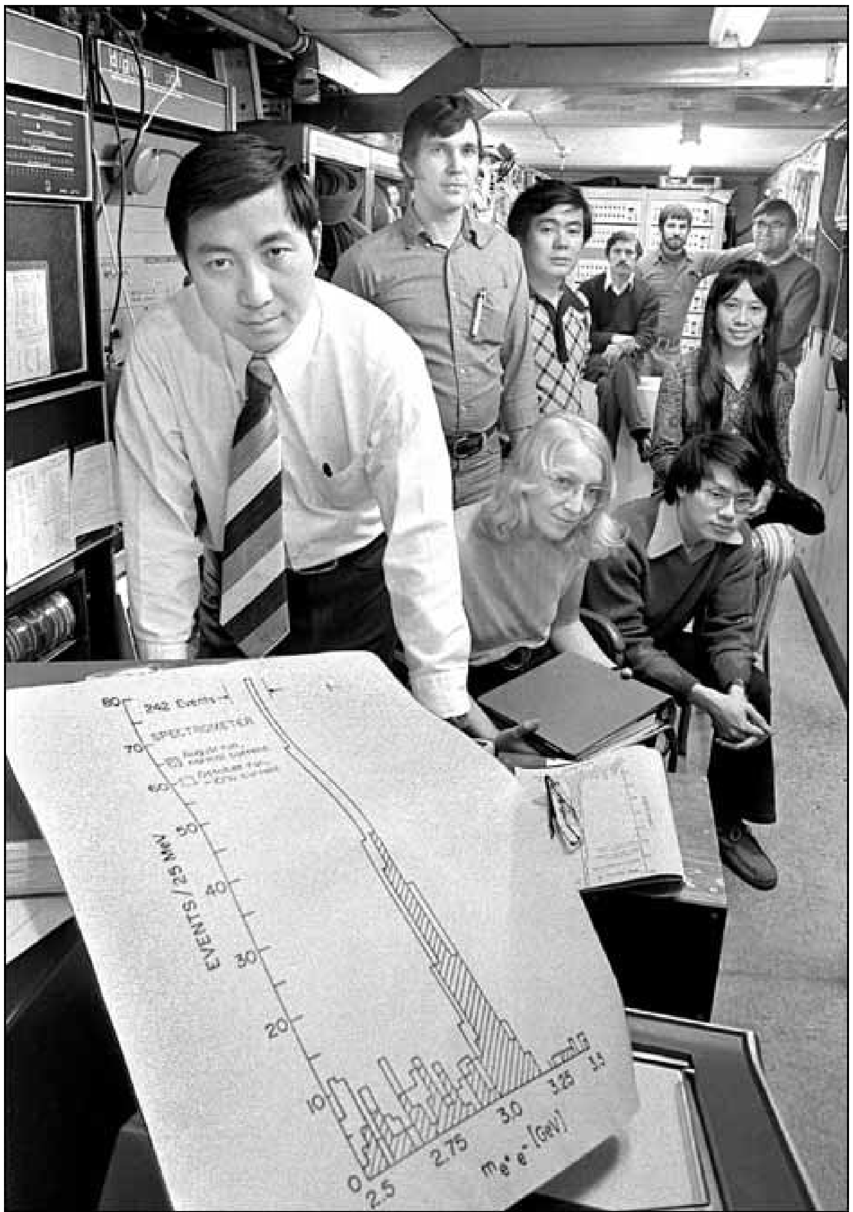


- All that we have seen so far is run centrally in CMS
- Now that the complexity of the problem is reduced to a small set of jets and other high-level objects (e, μ, τ, γ), you can start the very last bit of analysis, e.g., your PhD thesis
- It may look very different, depending on the question you want to address, e.g.:
  - Search for a new particle, for which you have a model
  - Search for new particles, as model-independently as possible
  - Measure a certain quantity, and compare it with models
  - Measure a certain quantity, for which there is no expectation (e.g., a fundamental parameter of Nature)
  - Measure the distribution of a quantity versus another one

Hypothesis testing

- Quantify the agreement of data with a null hypothesis H_0 (e.g., the Standard Model)
  - In case we only test H_0, methods may resemble to what is elsewhere called Anomaly Detection
  - See Alessia Saggio's seminar (pharmaceutical shop anomalies): https://agenda.irmp.ucl.ac.be/conferenceDisplay.py?confId=2558
- Or quantify which one is best between H_0 or H_1, e.g.:
  - H_0 = only backgrounds exist, and behave as in SM
  - H_1 = like H_0 but also the Higgs exists and behaves as in SM
- Or select which sub-set of $\{H_i\}$ is consistent with data
  - $\{H_i\}$ is often a continuum, e.g.: m=10.0±1.0 GeV, meaning that 9.0<m<11.0 GeV is the 68% confidence interval for m

Anomaly detection, the way we prefer it



Dream of every particle physicist:

- Study a simple feature of data, e.g., some invariant mass
- Find a spectacular anomaly with a clear interpretation, e.g., a peak rising from a smooth background
- Get a Nobel Prize

(Or at least get it awarded to your boss, or to some theorist who predicted it.)

Anomaly detection, the tough way

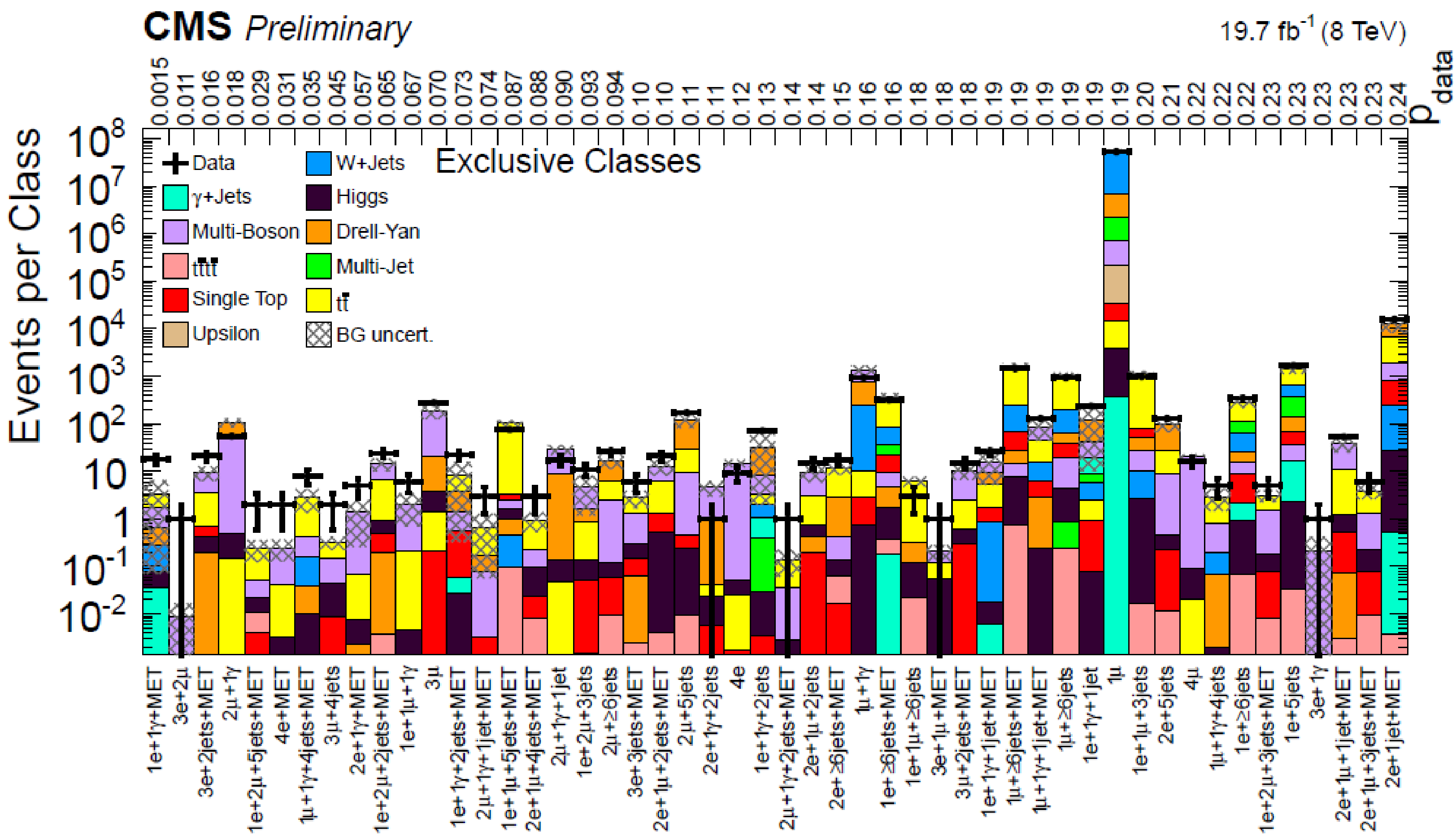


Figure 2: The 50 most significant exclusive event classes, considering only the total number of events.

CMS Coll., CMS-PAS-EXO-14-016

Anomaly detection, the tough way

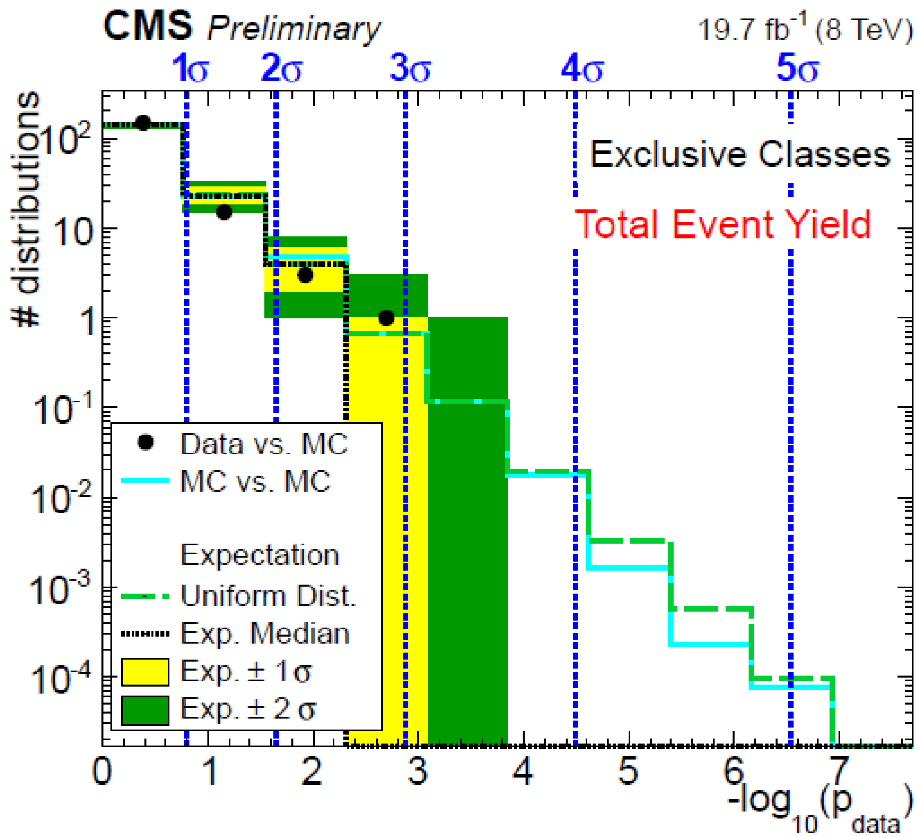


Figure 5: Distribution of p-values for exclusive event classes in the scan of total event yield. Black markers represent the measured data compared to the SM MC expectation. The histogram labeled "MC vs. MC" represents the comparison of the SM MC expectation to pseudo-data generated under the SM-only hypothesis. As a further comparison, the expectation from the uniform distribution is given, where the individual components are explained in Sec. 3.5. In the first bin 148 distributions are observed, with $139^{+4}_{-5}(1\sigma)^{+9}_{-10}(2\sigma)$ expected from the SM.

MC: Monte Carlo

(If you don't know what σ and p-value are, just ask, I have a backup slide)

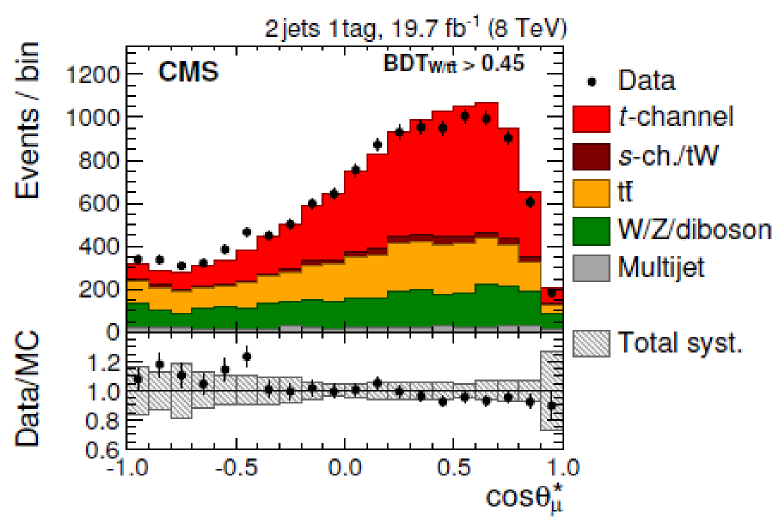
CMS Coll., CMS-PAS-EXO-14-016

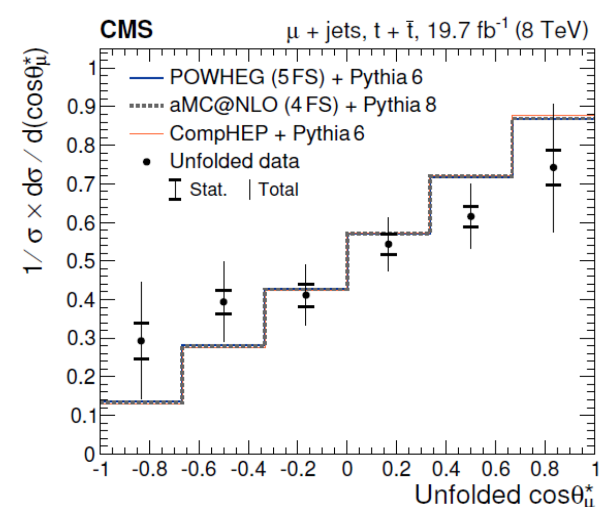
"Unfolding", i.e., ill-posed linear inversion problems

In some cases, useful to report some data distribution after correcting for the experimental nuisances such as:

- Bias from selection
- Resolution
- Background

CMS Coll. (M.Komm, AG, et ~2000 al.), JHEP 04 (2016) 073





In principle, a trivial matrix equation: **y = Mx + b**

x/y: unfolded/observed data (histogram); M: bin-by-bin migration matrix; b: background

"Unfolding", i.e., ill-posed linear inversion problems

In principle, a trivial matrix equation: **y = Mx + b**, but stochastic noise affects y, M and b. Nasty things can happen upon inversion:

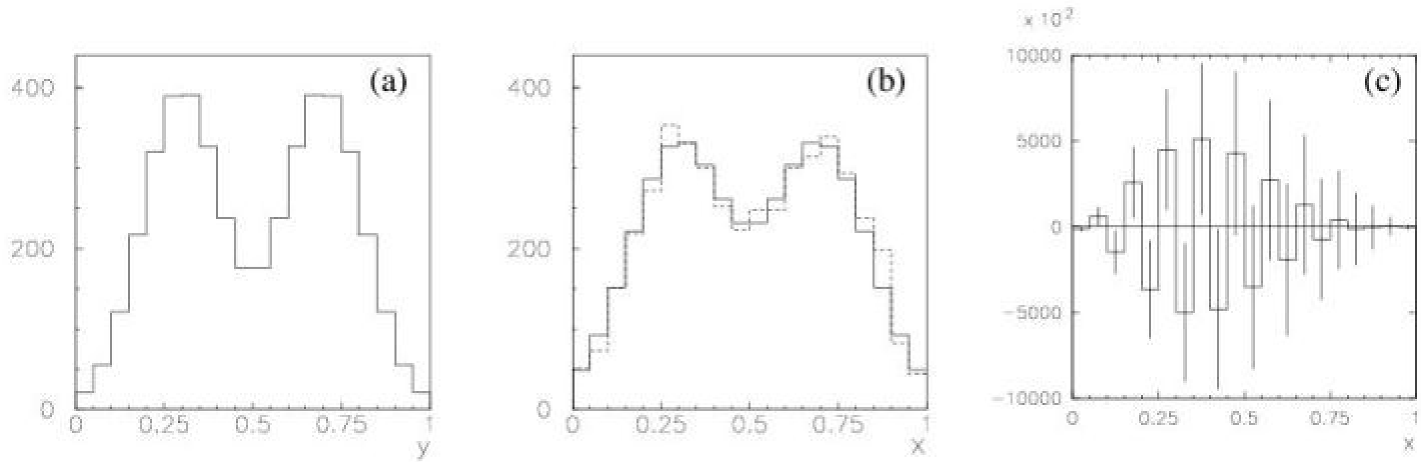


Fig. 2: Attempt to unfold using matrix inversion: (a) the 'true histogram', (b) the observed histogram $\mathbf{n}$ (dashed) and corresponding expectation values $\boldsymbol{\nu}$ (solid), (c) the estimators $\hat{\boldsymbol{\mu}}$ based on equation (8).

Illustrative plots from G.Cowan, Conf.Proc. C0203181 (2002) 248-257 [link]

Problem solved by *regularization* (e.g., add curvature-dumping term)

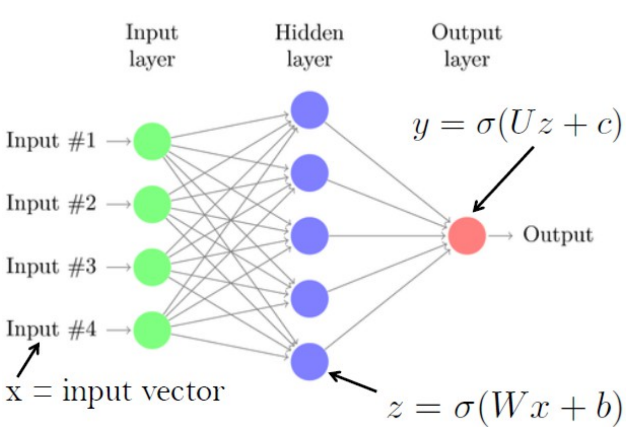
Highly controversial problem in HEP because any conceivable regularization introduces a bias towards expectation

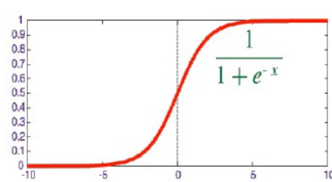
More in my blog posts on unfolding here: [1] - [2] - [3] - [4]

Machine Learning (ML)

- Some particle physicists started using ML techniques in the 90s, typically facing resistance by old-schoolers who were afraid of delegating physics intuition to „black boxes“
- Nowadays, Neural Networks (NN) and Boosted Decision Trees (BDT) are very standard tools, widely used in LHC analyses
- Probably because most „low hanging fruits“ have been reaped already, and what remains are the toughest cases

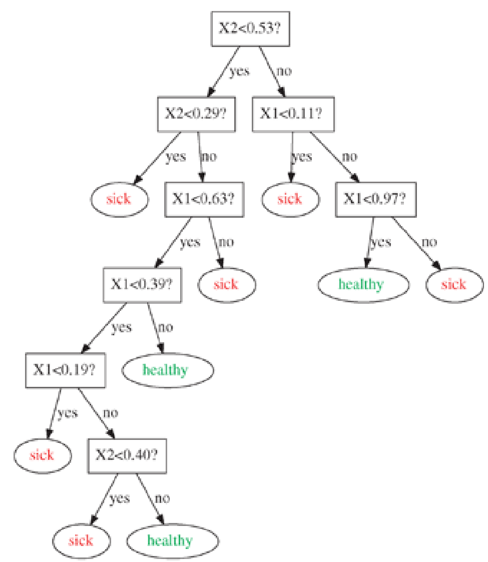
Neural Network:

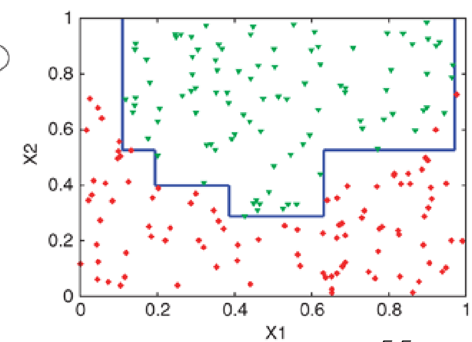




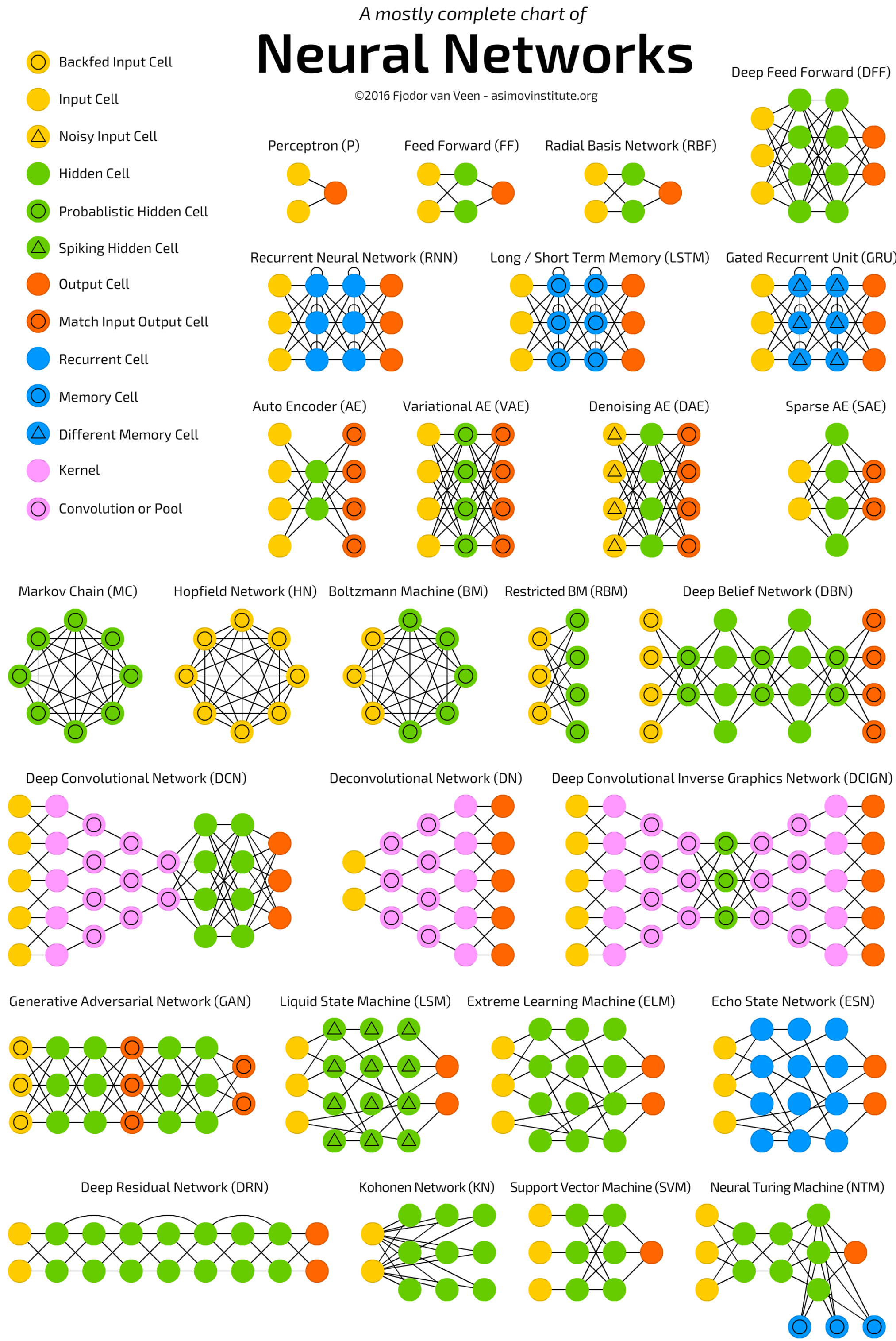
σ(x) = sigmoid function
is the *Activation Function*

Decision Tree:

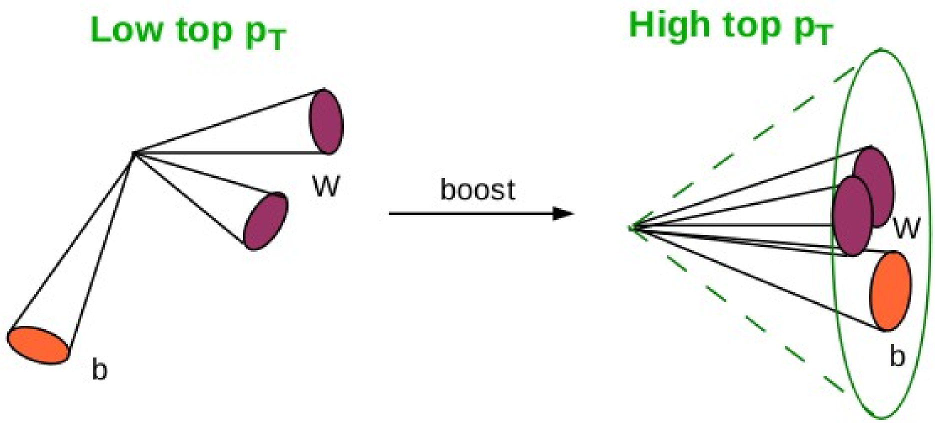




CP3 is a node of the European Training Network **AMVA4NewPhysics**, whose mission is to explore the suitability of novel ML methods for particle physics http://www.asimovinstitute.org/neural-network-zoo/

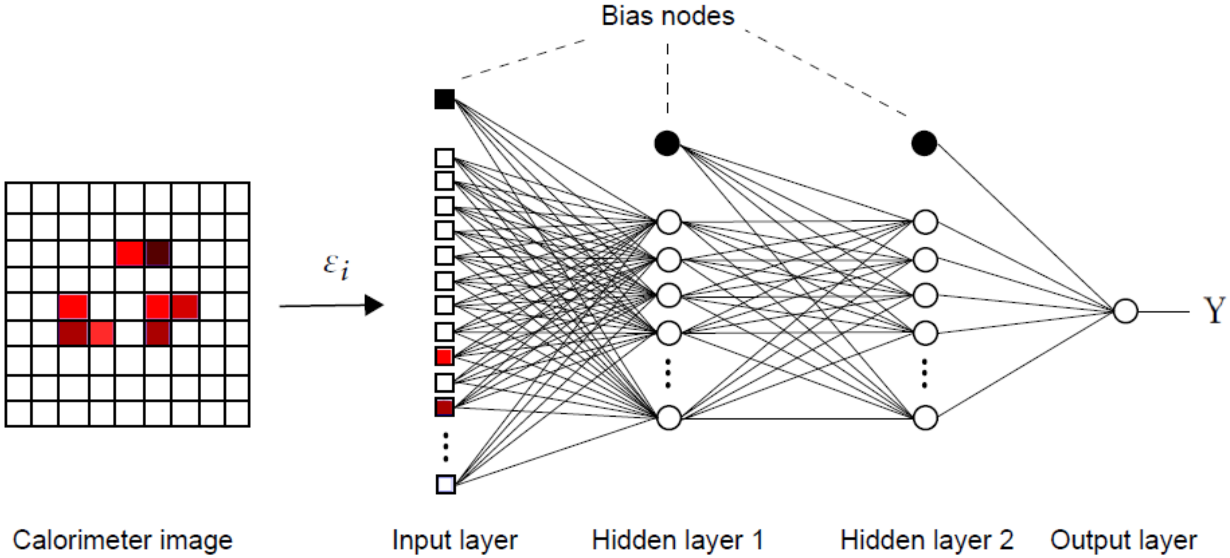


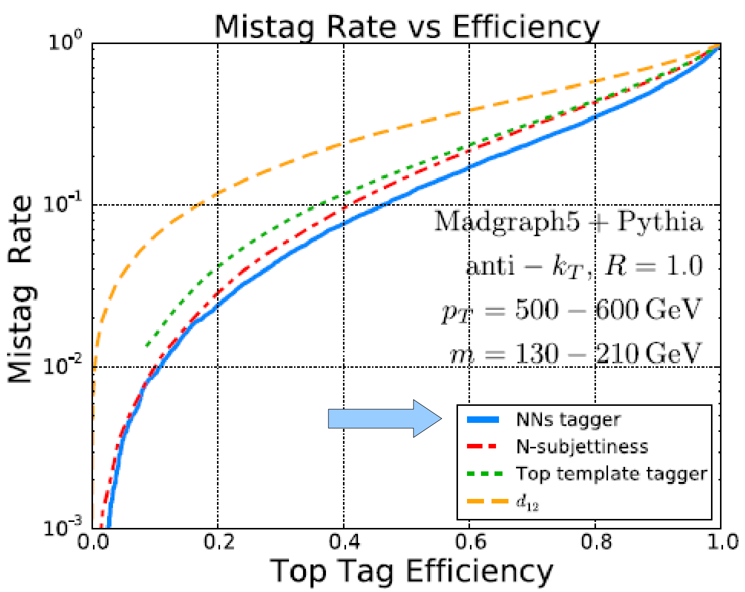
Identication of boosted top quarks



From http://www.quantumdiaries.org/2012/08/05/boost/

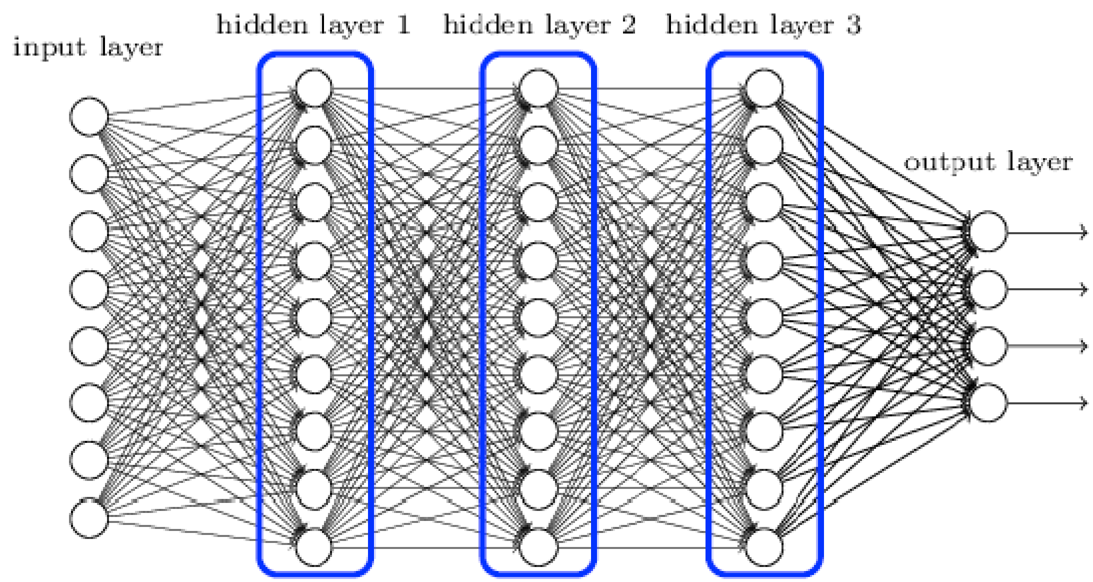
- At very high momentum, decay products tend to form a single jet
- Several algorithms discriminate those cases from the background using sub-clustering inside jets
- What about using digital image recognition methods, e.g.:





Mihailo Backovic et al, arXiv:1501.05968 [hep-ph], JHEP 1507 (2015) 086

Deep Learning



Basic idea: learn multiple levels of representations that correspond to different levels of abstraction

Computationally tough, but suitable for parallelization ($\Rightarrow$ GPUs)

CMS Coll. (O. Bondu, S. Brochet, C. Delaere, M. Delcourt, B. Francois, V. Lemaître, M. Vidal Marono, S. Wertz, et al.), CMS-PAS-HIG-17-006

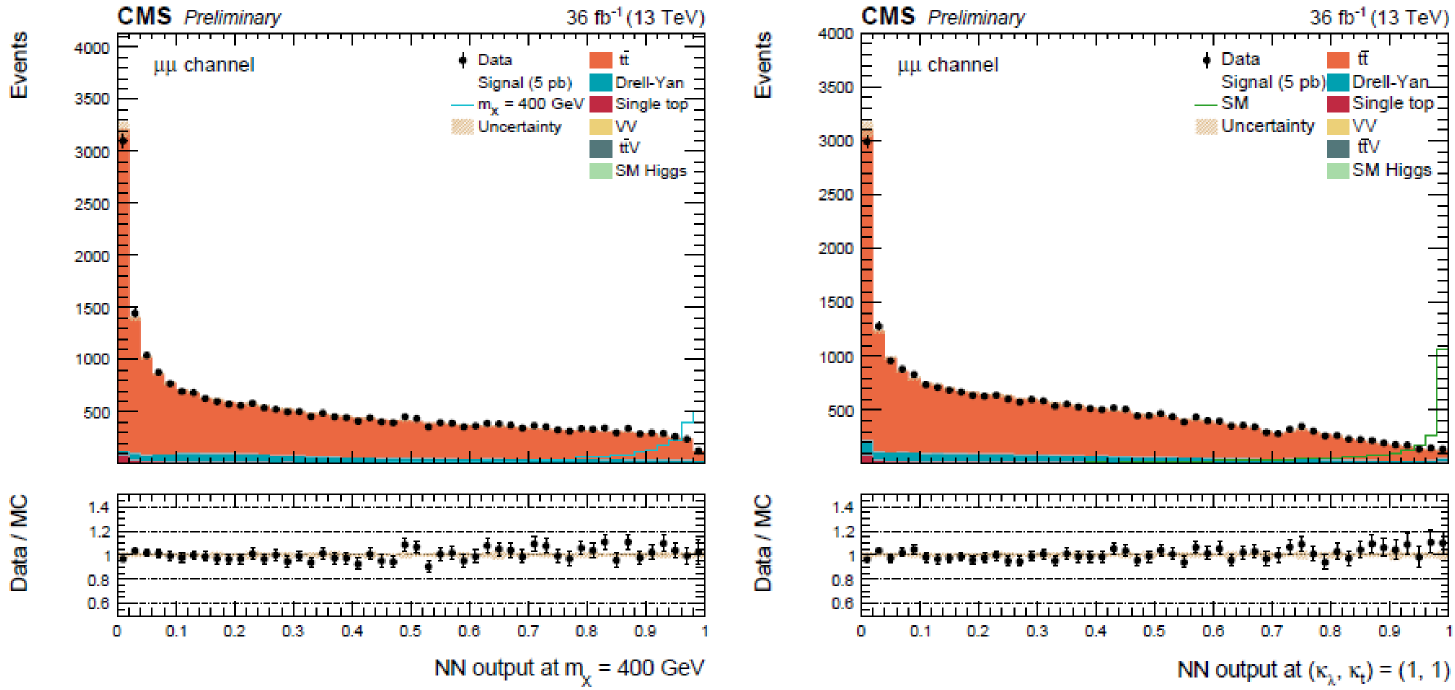
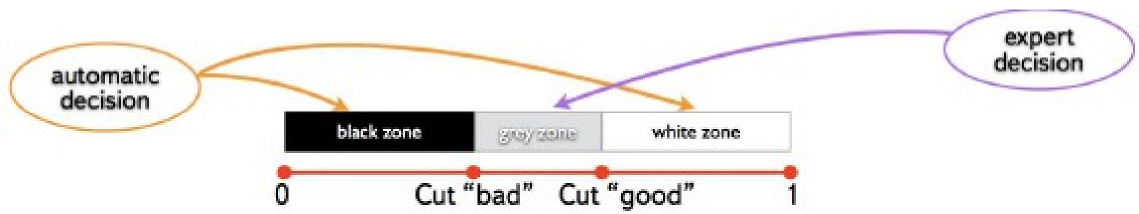


Figure 4: The DNN output distribution for data and simulated events after requiring all selection cuts, for the $\mu^+\mu^-$ channel. Output values towards 0 are background-like, while output values towards 1 are signal-like. The resonant DNN output (left) evaluated at $m_X = 400$ GeV and the non-resonant DNN output (right) evaluated at $\kappa_\lambda = 1$, $\kappa_t = 1$. The various signal hypotheses displayed have been scaled to a cross-section of 5 pb.

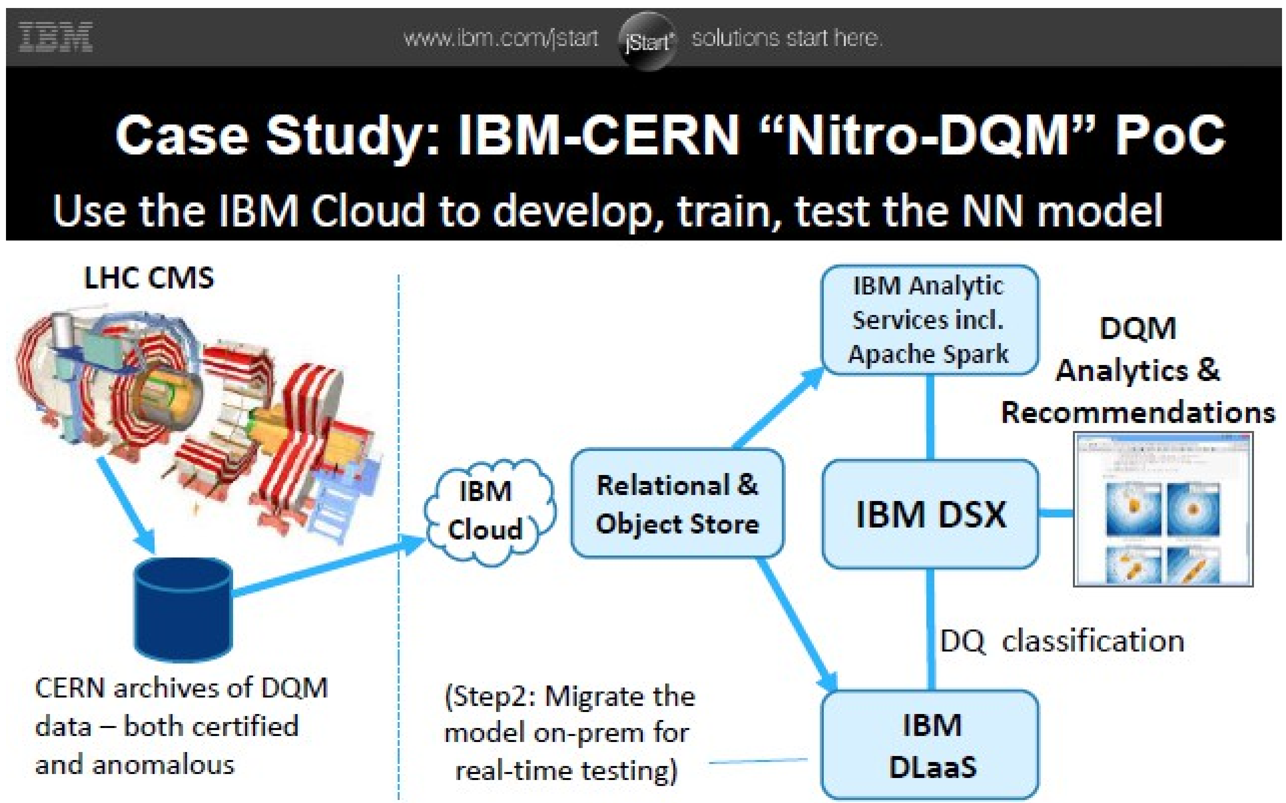
- Methodological novelty: mass and coupling hypotheses are inputs to the Deep NN, which is trained for all hypotheses at once
- Therefore, the algorithm learns how to interpolate
- Trained on a GPU, order of magnitude gain in wall-clock time

Another kind of anomaly detection:
Data Quality Monitoring (DQM)

Null hypothesis: data are ok



Goal: minimize grey zone, save time of humans



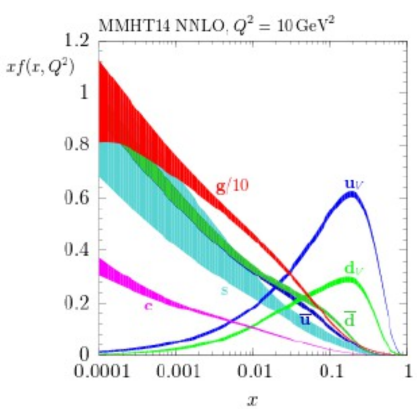
Cartoon / slide from Yandex / IBM speakers at the Inter-experimental Machine Learning Working Group Workshop on Machine Learning

Ultimate hypothesis testing

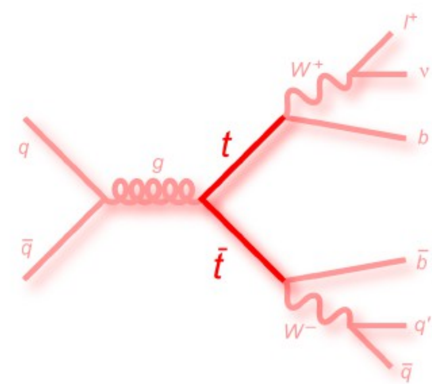
that we call Matrix Element Method (MEM)

To discriminate between competing hypotheses, full likelihood of each of them is computed with Monte Carlo from A to Z

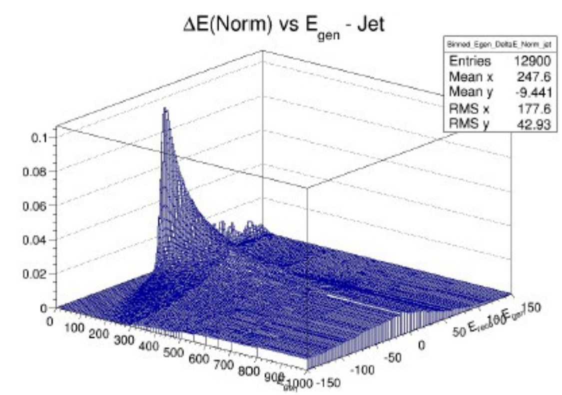
$$L(\mathrm{x}|\alpha) = \frac{1}{\sigma_\alpha} \int \mathrm{d}\Phi(y)\mathrm{d}x_1\mathrm{d}x_2 f(x_1)f(x_2)|M_\alpha(y, x_1, x_2)|^2 W(\mathrm{x}|y)$$



PDF



Matrix Element



Transfer Function

Computationally very heavy ⇒ parallelization

Case-by-case implementation is painful ⇒ general tool

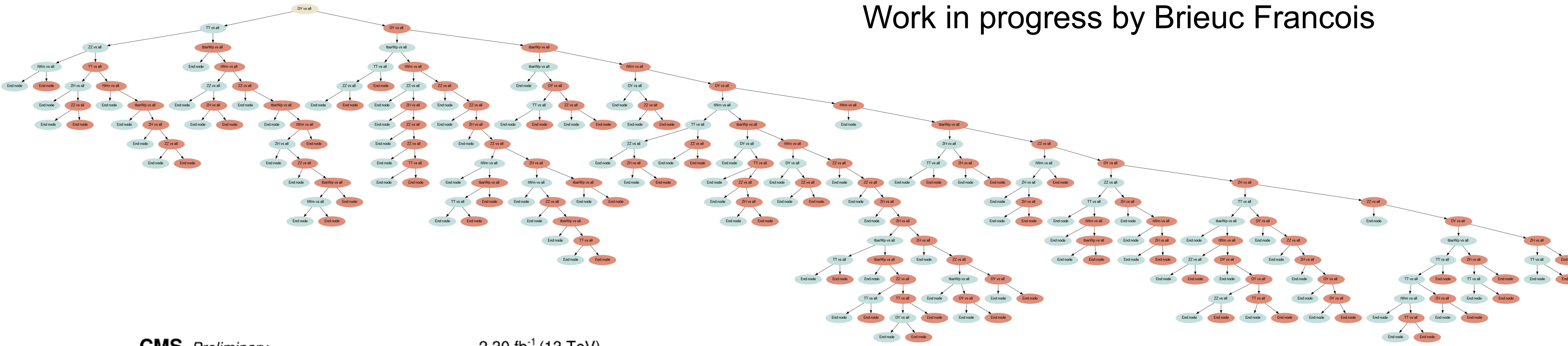
MoMEMta project at CP3, successor of MadWeight

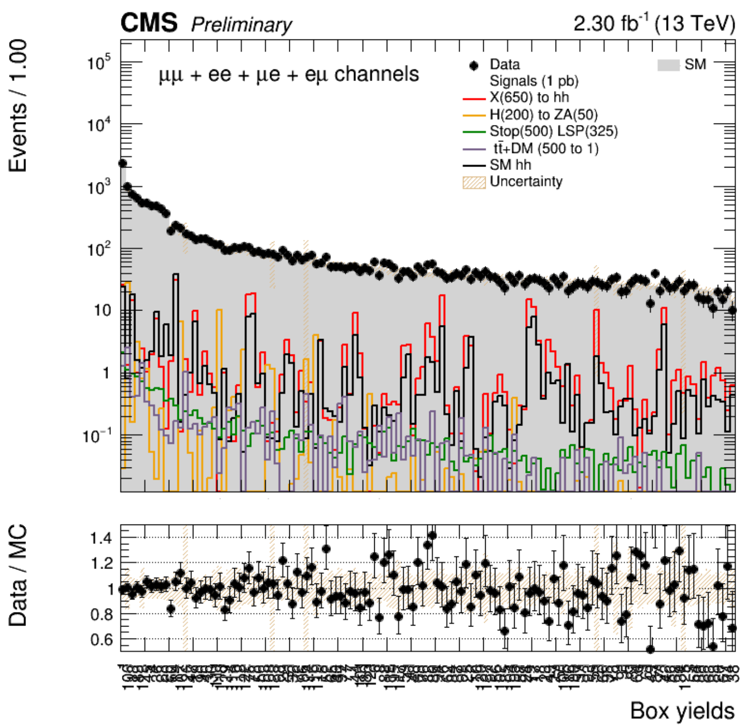
Example of experiment-theory collaboration

Illustration from A.Saggio

MEM + decision trees

Work in progress by Brieuc Francois



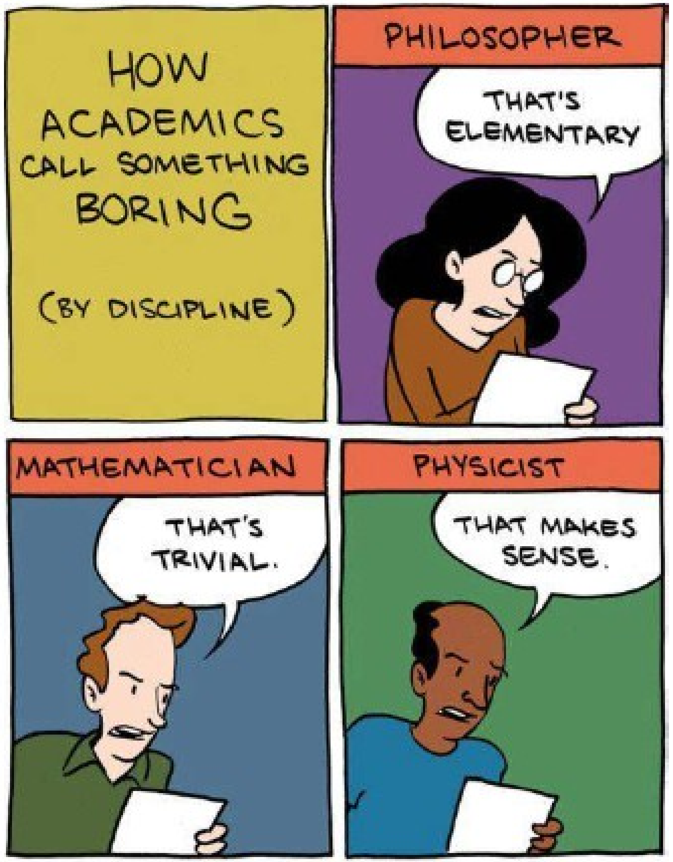


Recursively separate one of the backgrounds against the others, using the MEM weights under the respective hypotheses.

The choice of which background to separate at each step is based on the maximization of the purity improvement (for this specific bkg)

Conclusions

- Whether you are from M or P, I hope this talk was not too boring for you!
  - Where the definition of *boring* depends on the discipline, as the figure explains



From http://www.smbc-comics.com

Thanks for your attention

Image credit: www.holidaysuites.be

...and thanks to my sources:

- Marco Rovere
  - Who clarified the role of CA in CMS tracking and pointed me to the *Connecting The Dots* talks
- Ferenc Sikler
  - Who sent me his preliminary article draft about solving the tracking decision trees with graph theory methods
- Boris Mangano
  - Who made the pedagogical cartoons on CMS tracking and explained me several practical tracking issues
- Fosco Loregian and Michael Weiss
  - Who gave me "math feedback" on an early draft
- I also stole material from F. Ragusa, F. Pantaleo, S. Neuhaus, F. Tanedo, G. Salam, M.Kagan, A.Saggio

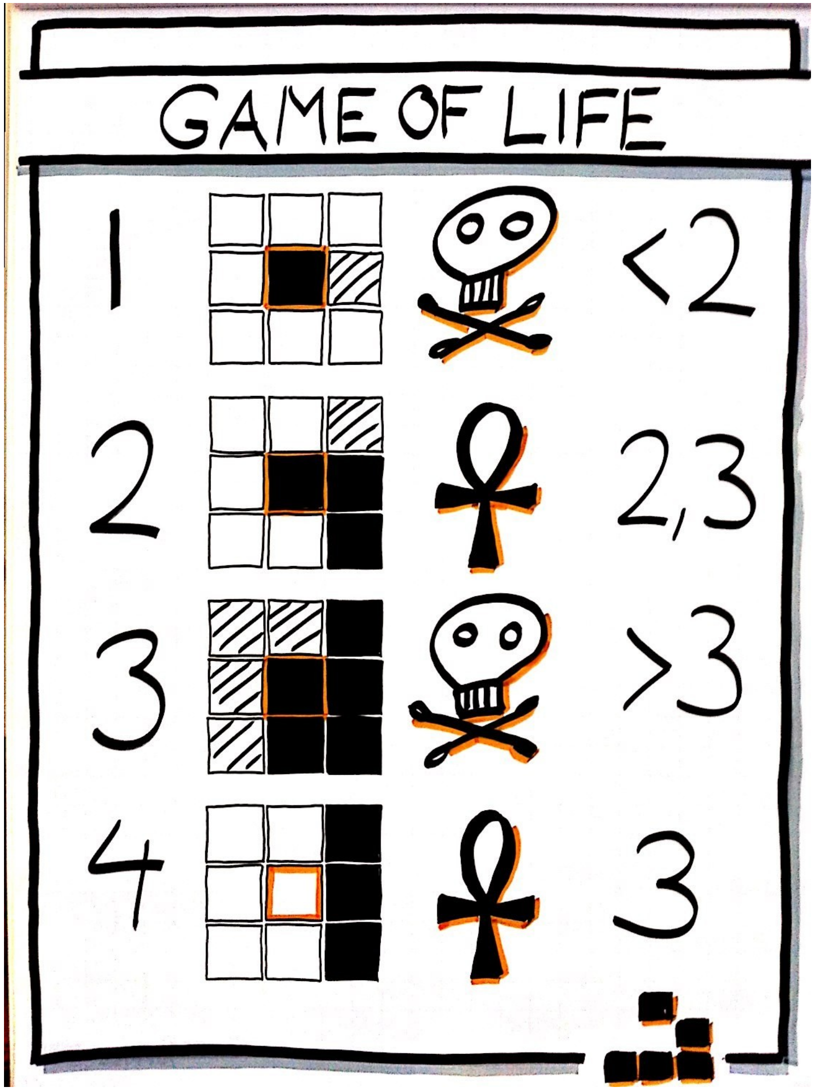
Statistical significance

- Given some data X and a suitable test statistic T one starts with the p-value, i.e. the probability of obtaining a value of T at least as extreme as the one observed, if H0 is true.
- p can be converted into the corresponding number of "sigma," i.e. standard deviation units from a Gaussian mean. This is done by finding x such that the integral from x to infinity of a unit Gaussian N(0,1) equals p:

$$\frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{-\frac{t^2}{2}}dt = p$$

Slide partially copied from T.Dorigo

Cellular Automata (CA)



In general, a CA consists of a regular grid of cells, each in a finite number of states.

For each cell, a set of cells called its neighborhood is defined.

An initial state (time t = 0) is selected by assigning a state for each cell.

The new state of each cell depends from the current states of the cell and its neighborhood.

Famous example: Conway's Game of Life

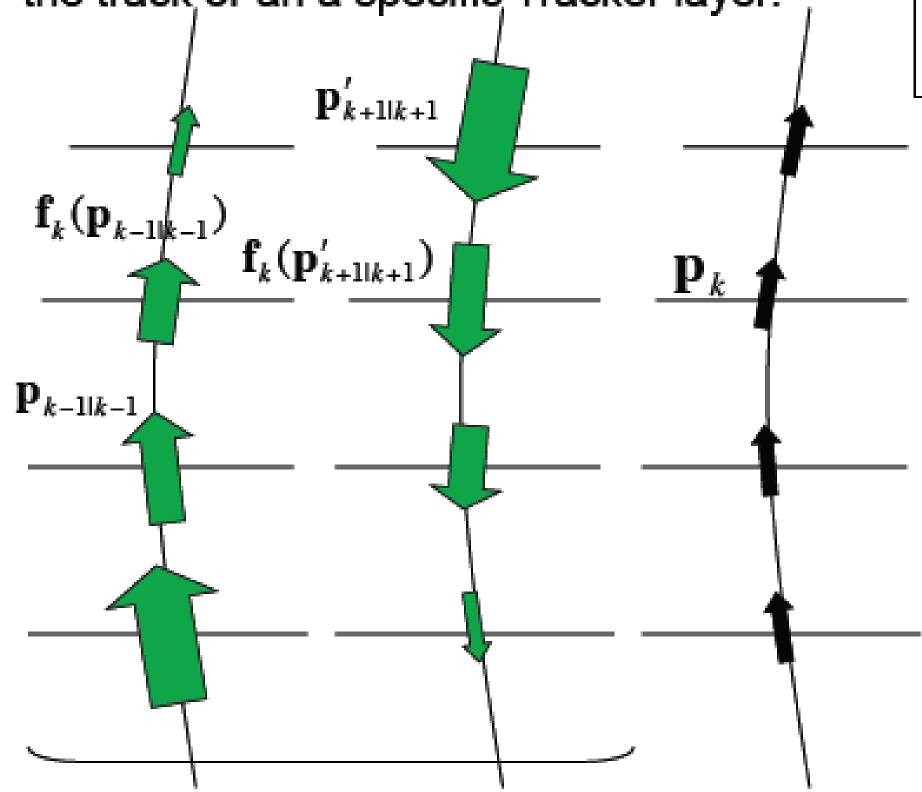
Picture from https://qualityswdev.com/2011/07/31/conways-game-of-life-in-scala/

Kalman Filter in tracking

Once the set of hits that defines a trajectory are identified, the next step is to fit the positions of the hits to extract the particle parameters: at vertex, at the end of the track or an a specific Tracker layer:

$$\mathbf{p}_k = \mathbf{p}_{k|k} \oplus \mathbf{f}_k(\mathbf{p}'_{k+1|k+1})$$

A statistically correct *weighted mean:* **Kalman *smoother***



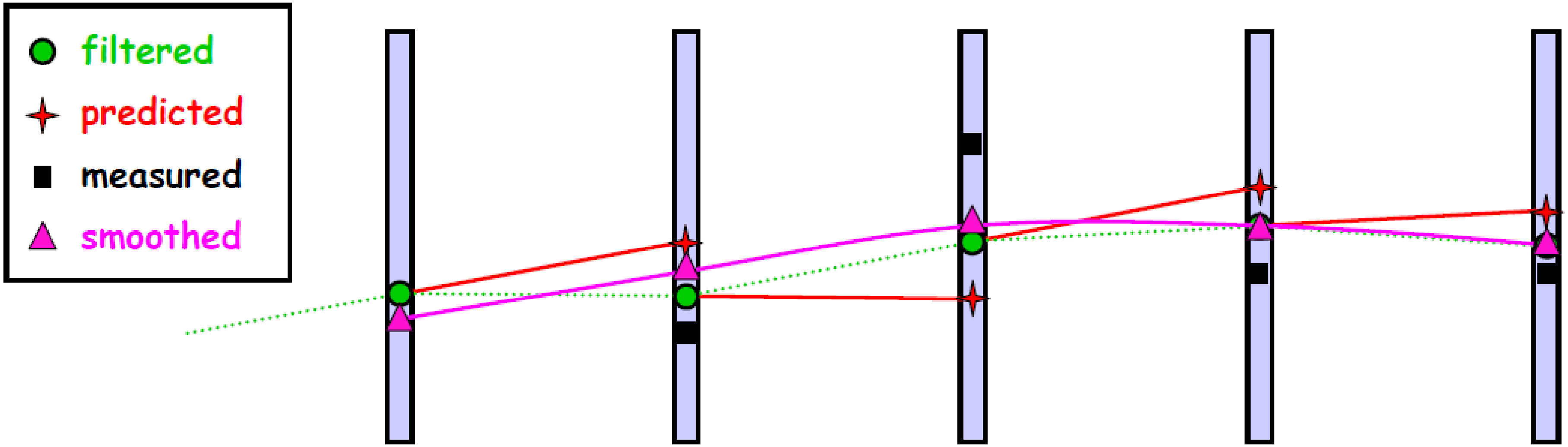
Both in-out filter and out-in one are run

$\mathbf{p}_{k\vert k}$	Contains information from measurements: 1,2,..,k
$\mathbf{p}'_{k+1\vert k+1}$	Contains information from measurements: n,n-1,…,k+1
$\mathbf{p}_k$	Contain the full information. All measurement from 1 to n are used.

All math details in book from R.Fruhwirth: “Data Analysis Techniques for High-Energy Physics”

Slide from Boris Mangano

Kalman Filter



to start the Kalman Filter we need a seed

the position on the next plane is predicted

the measurement is considered

prediction and measurement are merged (filtered)

then new prediction … measurement …

filtering … prediction … measurement …

filtering … prediction … measurement …

The filtered trajectory

The smoothed trajectory

filtered trajectory

predicted trajectory

smoothed trajectory

Kalman Filter

The filtering is nothing but a weighted average of the

- new measurement y_n
- the prediction y_p

$$y_f = \frac{\frac{1}{\sigma_p^2} y_p + \frac{1}{\sigma_n^2} y_n}{\frac{1}{\sigma_p^2} + \frac{1}{\sigma_n^2}}$$

$$\boxed{y_f = \frac{\sigma_n^2}{\sigma_p^2 + \sigma_n^2} y_p + \frac{\sigma_p^2}{\sigma_p^2 + \sigma_n^2} y_n}$$

Clearly if the new measurement has a very large error

$\sigma_n \to \infty \qquad y_f \to y_p$ — measurement ignored

If the prediction has a large error (for example large multiple scattering)

$\sigma_p \to \infty \qquad y_f \to y_n$ — prediction ignored

The effect of multiple scattering, or any other stochastic effect, can be handled in the prediction

The advantages of this procedure are

- is an iterative procedure
- not necessary to invert large matrices
- is a local procedure: at any step the estimate at the given plane is the best that make use of the prevoius measurements

Kalman Filter

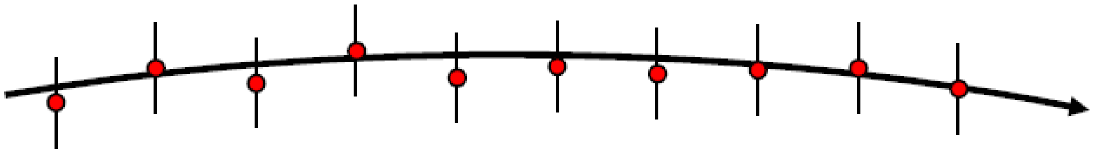
The consequence is that if you want the optimal measurement at the origin you have to start the filter from the end of the track

After all the measurements have been used (filtered) it is possible possible to build a procedure that

- uses the (stored) intermediate results of the filter
- gives the best parameter estimation at any point

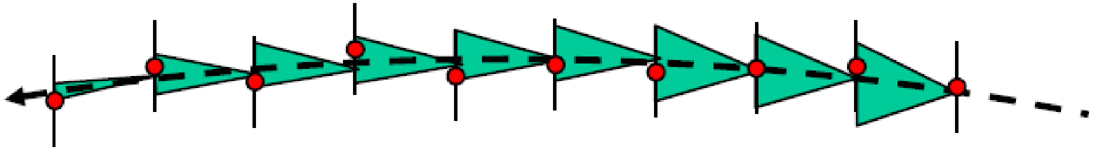
This is the smoother

production vertex



direction of flight →

production vertex



← direction of filter

Kalman Filter

Applications of Kalman Filter:

- navigation
- radar tracking
- sonar ranging
- satellite orbit computation
- stock prize prediction

It is used in all sort of fields

- Eagle landed on the moon using KF
- Gyroscopes in airplanes use KF

Usually the problem is to estimate a state of some sort and its uncertainty

- location and velocity of airplane
- track parameters of charged particles in HEP experiments

However we do not observe the state directly

We only observe some measurements from sensors which are noisy:

- radar tracking
- charged particle tracking detectors

As an additional complication the state evolve in time with is own uncertainties: process stochastic noise

- deviation from trajectory due to random wind
- multiple scattering

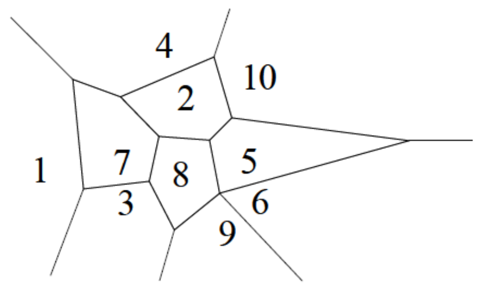
In case of tracking in HEP instead of time we can consider the evolution of the track parameter at the discrete layers where the detectors perform the measurement

Tracking timing: 2016 vs 2017, standard vs CA, CPU vs GPU

Algorithm	**time per event [ms]**
2016 Pixel Tracks	29.3 ± 13.1
Triplet Propagation	72.1 ± 25.7
GPU Cellular Automaton	1.2 ± 0.9
CPU Cellular Automaton	14 ± 6.2

Jet finding with Voronoi diagrams

Finding Geom Nearest Neighbours



Given a set of vertices on plane (1...10) a *Voronoi diagram* partitions plane into cells containing all points closest to each vertex

Dirichlet '1850, Voronoi '1908

A vertex's nearest other vertex is always in an adjacent cell.

E.g. GNN of point 7 will be found among 1,4,2,8,3 (it turns out to be 3)

Construction of Voronoi diagram for N points: $N \ln N$ time — Fortune '88

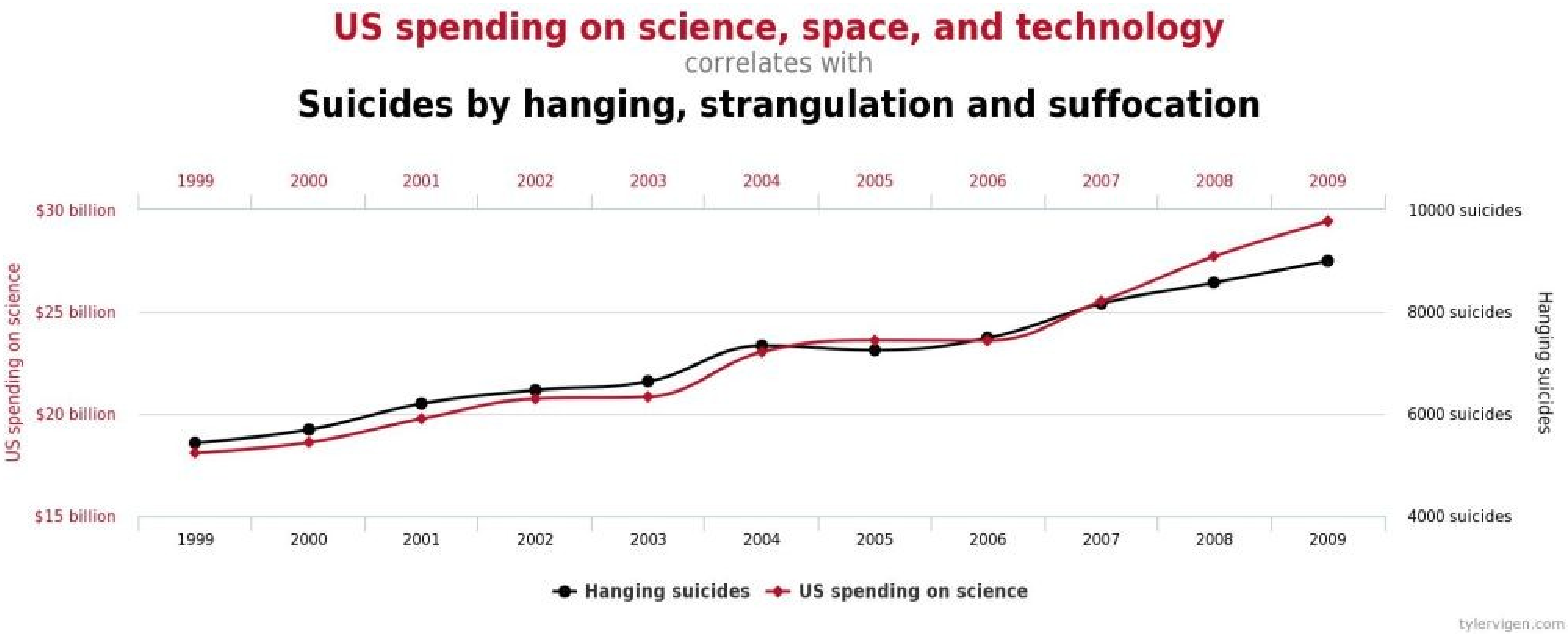
Update of 1 point in Voronoi diagram: $\ln N$ time

Devillers '99 [+ related work by other authors]

Convenient C++ package available: **CGAL** — http://www.cgal.org

Slide from Gavin Salam

Correlation is not causation



Source: http://tylervigen.com/spurious-correlations