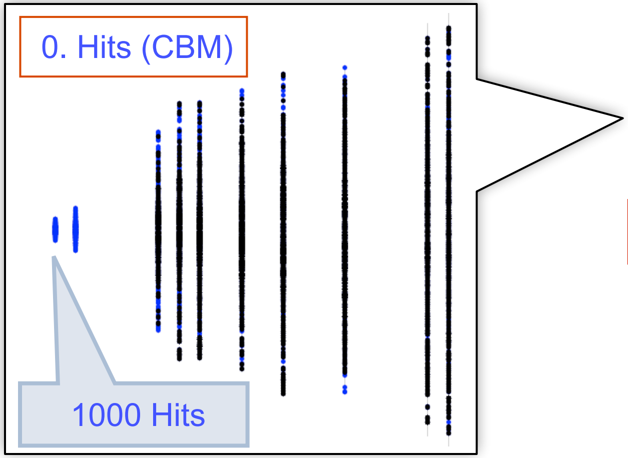
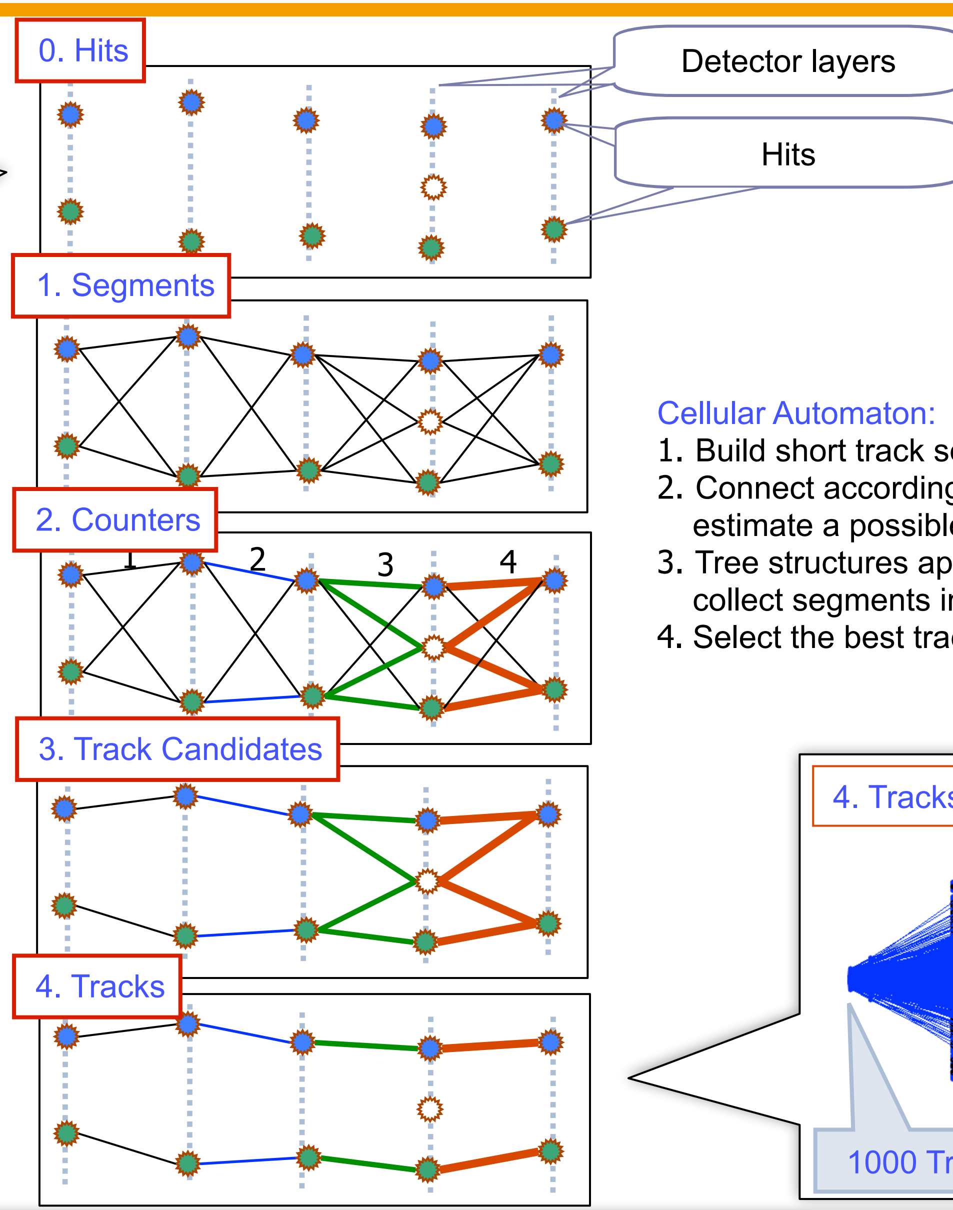
Summary of Event Reconstruction with Cellular Automaton, Kalman Filter and KF Particle

I. Kisel, M. Pugach and M. Zyzak

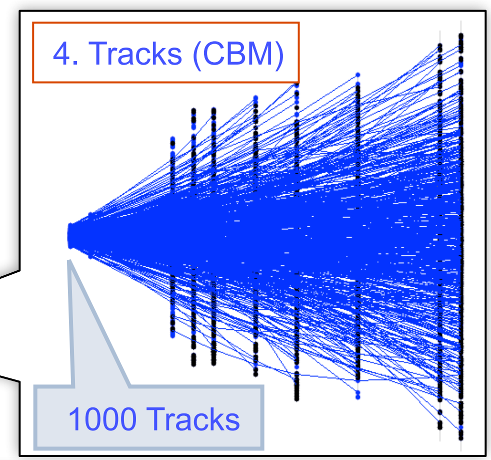
Uni-Frankfurt, FIAS, GSI

Cellular Automaton as Track Finder





Cellular Automaton:
1. Build short track segments.
2. Connect according to the track model, estimate a possible position on a track.
3. Tree structures appear, collect segments into track candidates.
4. Select the best track candidates.



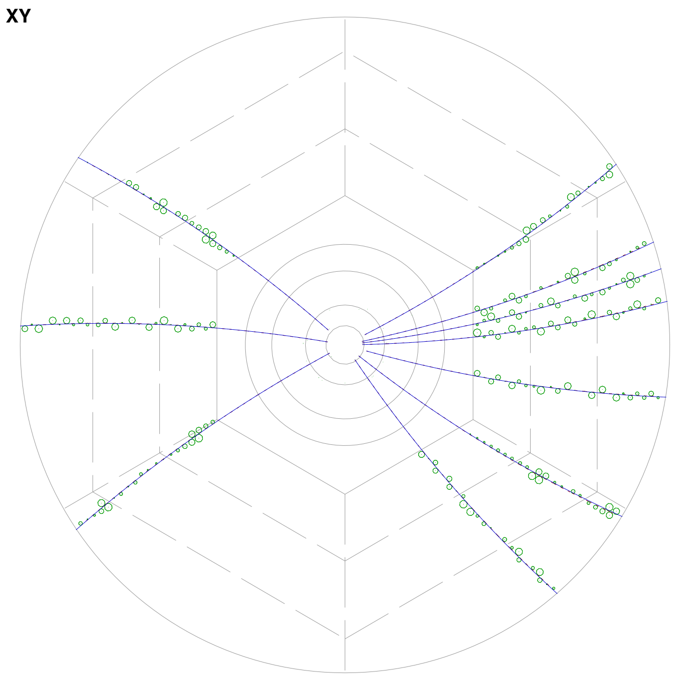
Cellular Automaton:
- local w.r.t. data
- intrinsically parallel
- extremely simple
- very fast

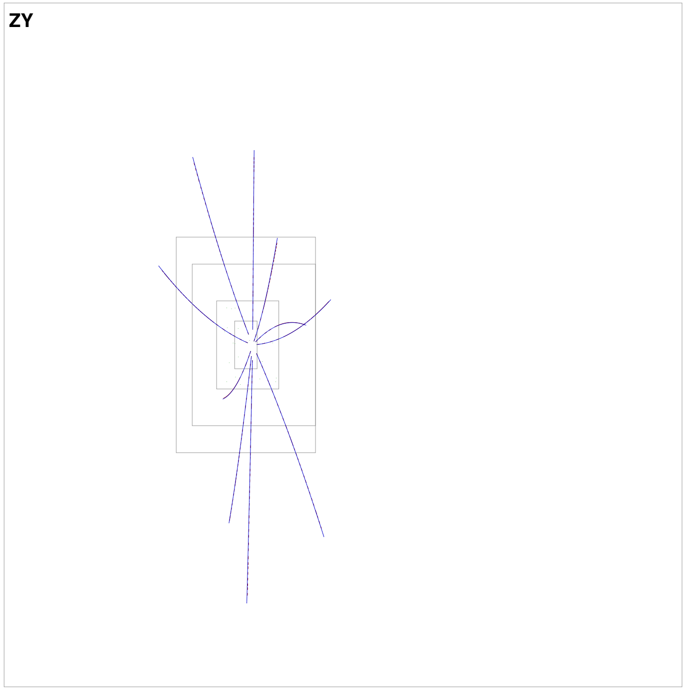
Perfect for many-core CPU/GPU !

Useful for complicated event topologies with large combinatorics and for parallel hardware

CA Track Finder in STT and MVD

10 primary tracks with pt = 1 GeV/c





	STT	STT+MVD
Efficiency	97.2	**99.3**
Clone	1.8	9.2
Ghost	2.5	2.5
Tracks/event	10	10
Time, ms/event	5	7

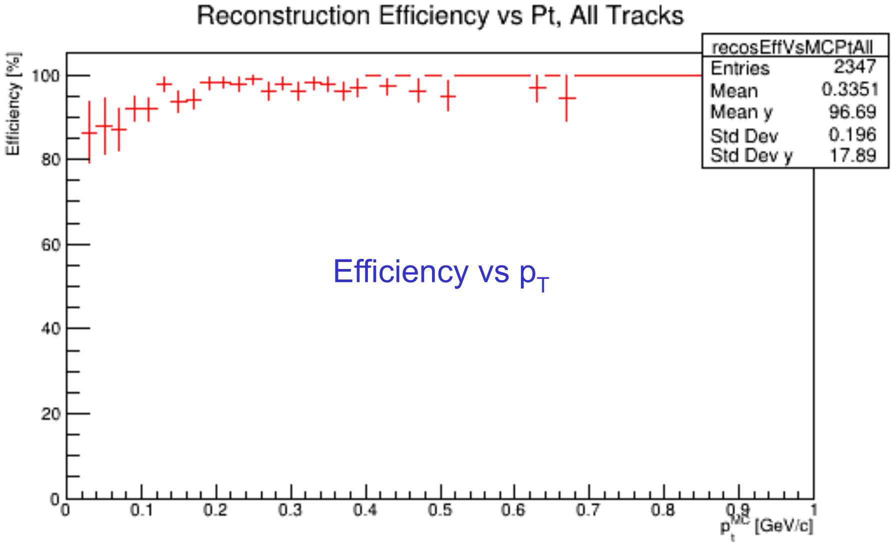
	STT+MVD
Efficiency	100.0
Clone	4.8
Ghost	1.5
Tracks/event	10
Time, ms/event	**1**

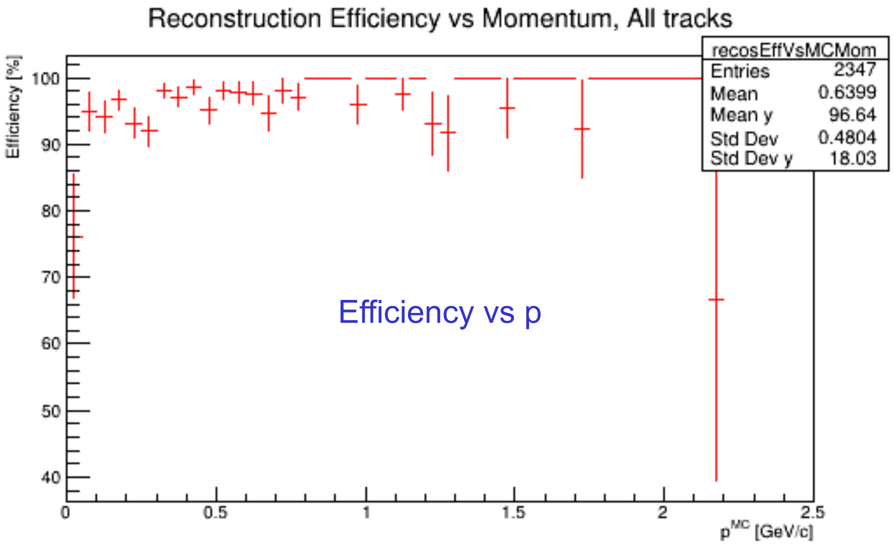
Igor Kulakov, 11.09.2013

Sergey Gorbunov, 10.12.2014

Tracking Efficiency

Efficiency, %	Eff. new
High-p primary	98.1
High-p secondary	93.3
High-p set	97.7
Low-p primary	97.7
Low-p secondary	93.0
Low-p set	96.2
All set	**96.6**

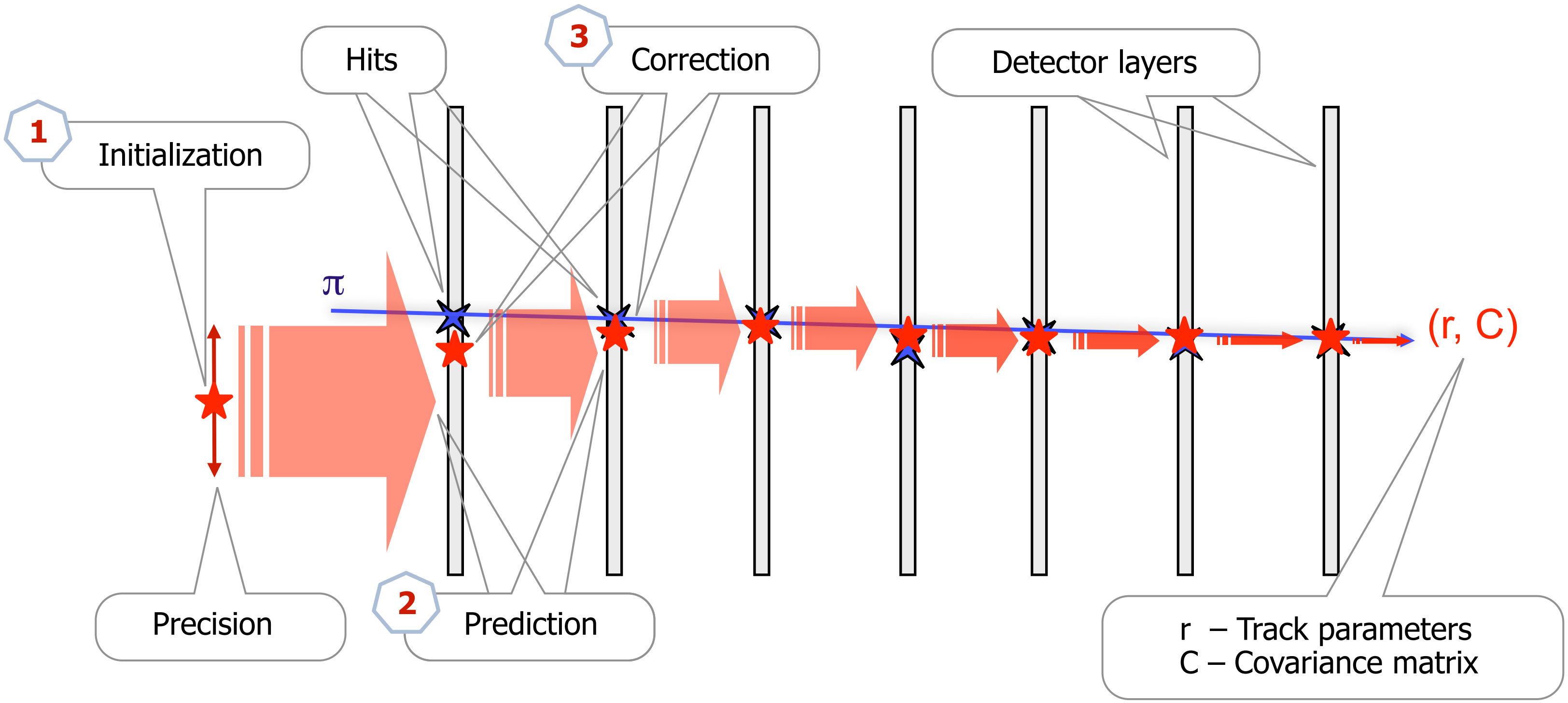




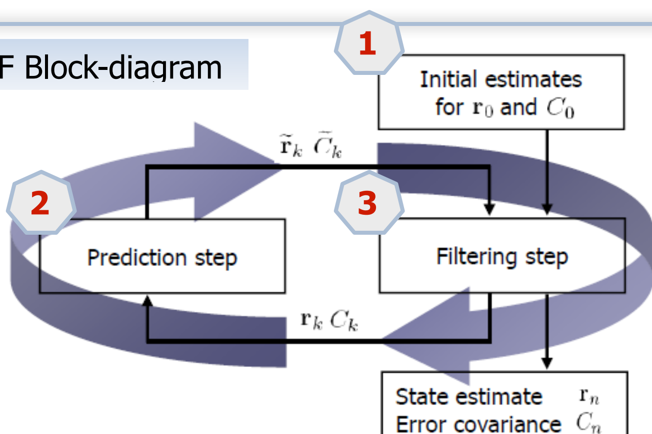
Irina Zivko, 07.06.2017

Kalman Filter based Track Fit

Track fit: Estimation of the track parameters at one or more hits along the track – Kalman Filter (KF)



KF Block-diagram



KF as a recursive least squares method

State vector

Position, direction and momentum

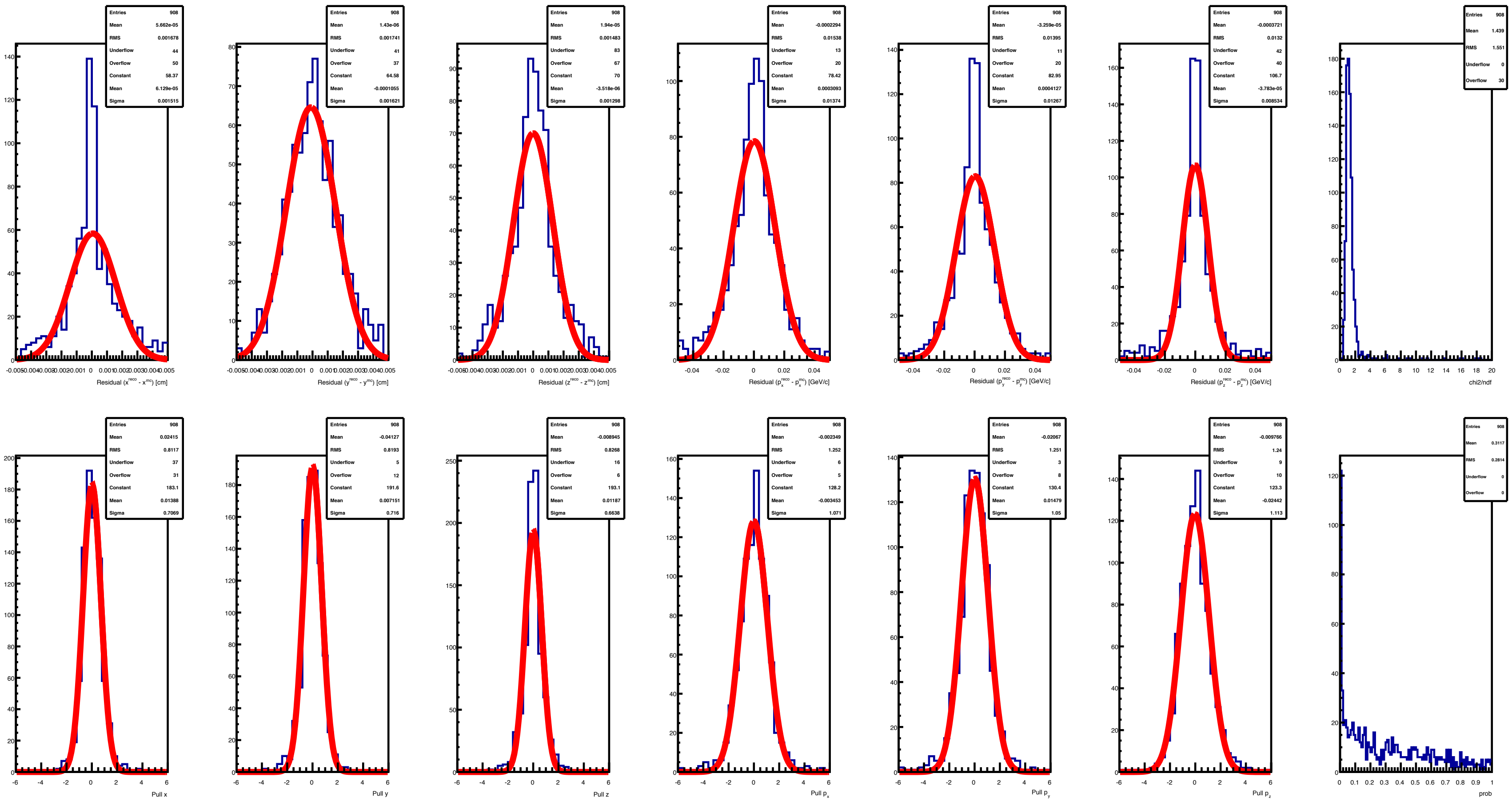
$$r = \{\, x, y, z, p_x, p_y, p_z \,\}$$

Kalman Filter:

1. Start with an arbitrary initialization.
2. Add one hit after another.
3. Improve the state vector.
4. Get the optimal parameters after the last hit.

Nowadays the Kalman Filter is used in almost all HEP experiments

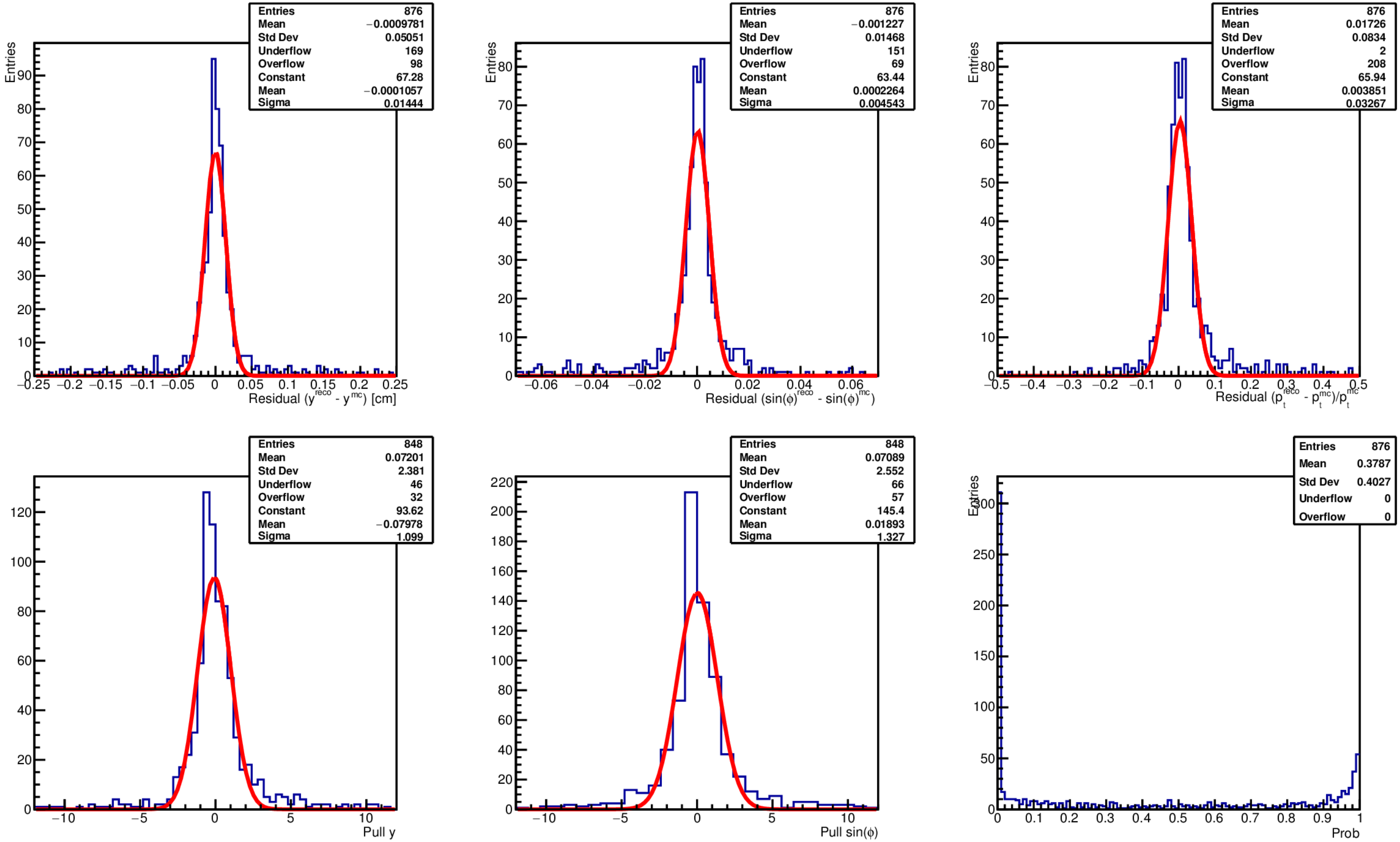
KF Fit Quality (STT+MVD)



	x, µm	y, µm	z, µm	p_x, MeV/c	p_y, MeV/c	p_z, MeV/c
Residual	15	16	13	14	13	9
Pull	0,7	0,7	0,7	1,1	1,1	1,1

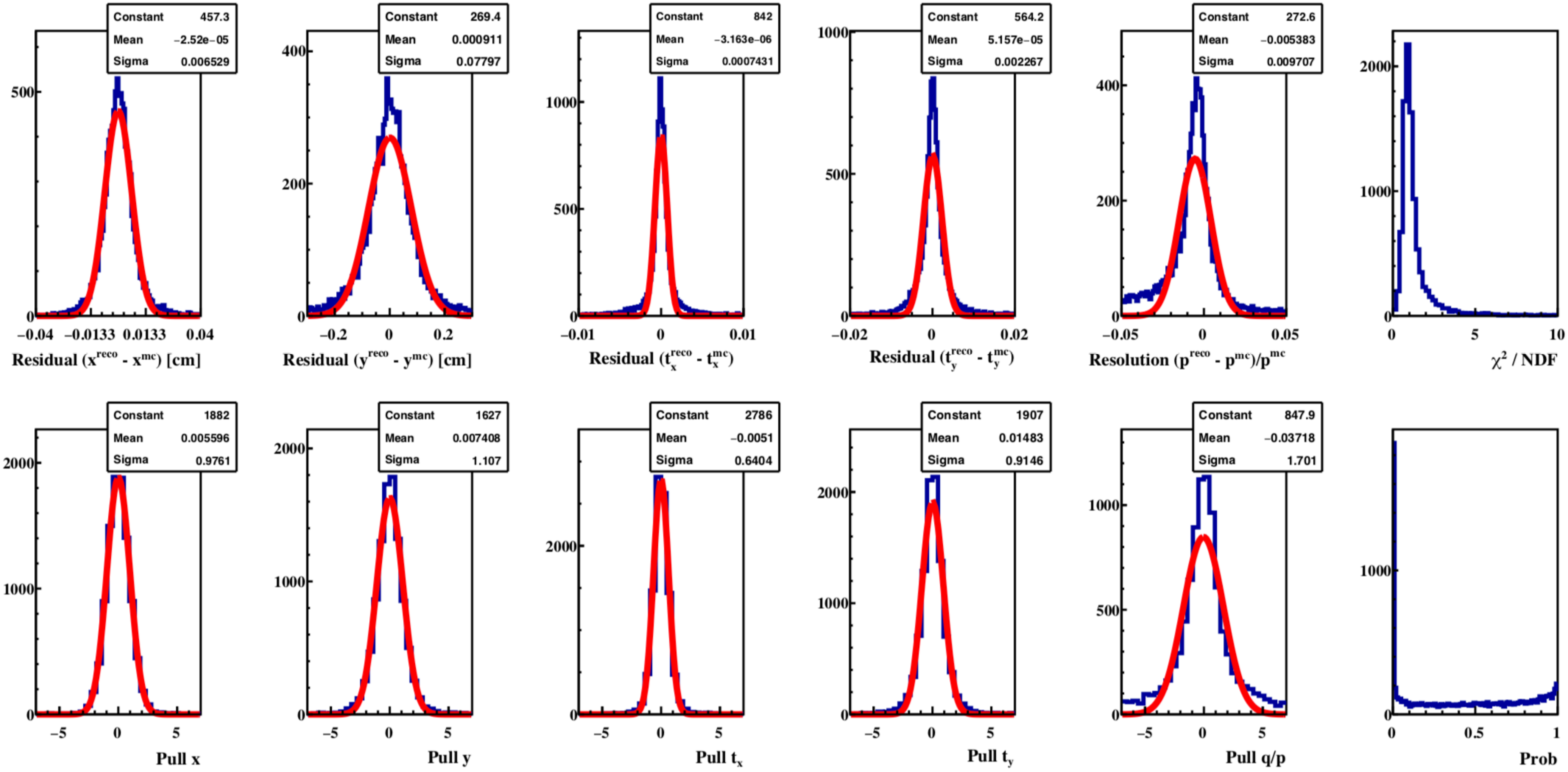
Igor Kulakov, Maksym Zyzak, 25.06.2013

KF Residuals and Pulls (STT+MVD)



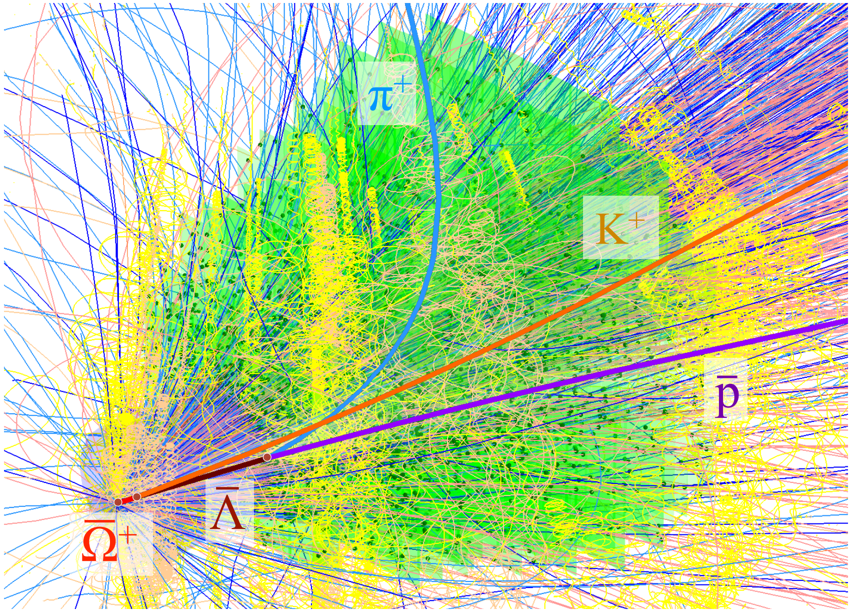
Irina Zivko, 07.06.2017

KF Residuals and Pulls (FTS)



Mykhailo Pugach, Irina Zivko, 05.09.2017

KF Particle: Reconstruction of short-lived Particles



Simulated AuAu collision at 25 AGeV

$$\bar{\Omega}^+ \leftarrow \bar{\Lambda}\, K^+$$
$$\bar{\Lambda} \rightarrow \bar{p}\, \pi^+$$

$$\mathbf{r} = \{\, x, y, z, p_x, p_y, p_z, E \,\}$$

State vector

$$C = \langle \mathbf{r}\mathbf{r}^T \rangle = \begin{bmatrix} \sigma_x^2 & C_{xy} & C_{xz} & C_{xp_x} & C_{xp_y} & C_{xp_z} & C_{xE} \\ C_{xy} & \sigma_y^2 & C_{yz} & C_{yp_x} & C_{yp_y} & C_{yp_z} & C_{yE} \\ C_{xz} & C_{yz} & \sigma_z^2 & C_{zp_x} & C_{zp_y} & C_{zp_z} & C_{zE} \\ C_{xp_x} & C_{yp_x} & C_{zp_x} & \sigma_{p_x}^2 & C_{p_xp_y} & C_{p_xp_z} & C_{p_xE} \\ C_{xp_y} & C_{yp_y} & C_{zp_y} & C_{p_xp_y} & \sigma_{p_y}^2 & C_{p_yp_z} & C_{p_yE} \\ C_{xp_z} & C_{yp_z} & C_{zp_z} & C_{p_xp_z} & C_{p_yp_z} & \sigma_{p_z}^2 & C_{p_yE} \\ C_{xE} & C_{yE} & C_{zE} & C_{p_xE} & C_{p_yE} & C_{p_zE} & \sigma_E^2 \end{bmatrix}$$

Covariance matrix

```
KFParticle Lambda(P, Pi);                    // construct anti Lambda
Lambda.SetMassConstraint(1.1157);            // improve momentum and mass
KFParticle Omega(K, Lambda);                 // construct anti Omega
PV -= (P; Pi; K);                            // clean the primary vertex
PV += Omega;                                 // add Omega to the primary vertex
Omega.SetProductionVertex(PV);               // Omega is fully fitted
(K; Lambda).SetProductionVertex(Omega);      // K, Lambda are fully fitted
(P; Pi).SetProductionVertex(Lambda);         // p, pi are fully fitted
```

Features:

- KF Particle class describes particles by the state vector and the covariance matrix.
- Covariance matrix contains essential information about tracking and detector performance.
- The method for mathematically correct usage of covariance matrices is provided by the KF Particle package based on the Kalman filter (KF).
- Heavy mathematics of KF requires fast and vectorised algorithms.
- Mother and daughter particles are treated in the same way.
- The natural and simple interface allows two reconstruct easily complicated decay chains.
- The package is geometrically independent and can be adapted to different experiments (CBM, ALICE, STAR).

KF Particle provides a simple and very efficient approach to physics analysis

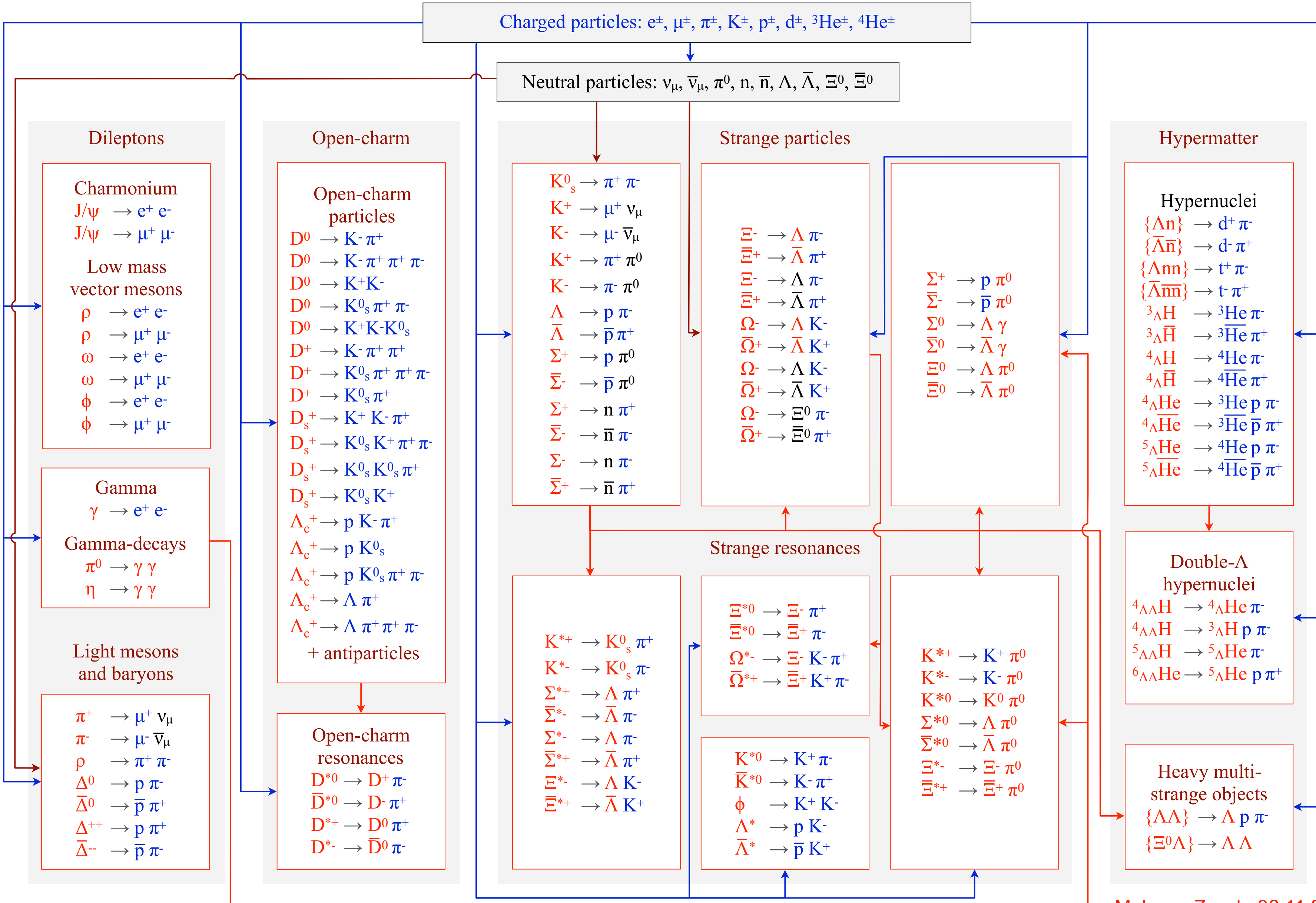
KF Particle: Functionality

Functions	CBM	PANDA	ALICE	STAR
Construction of mother particles	+	+	+	+
Addition and subtraction of the daughter particle to (from) the mother particle	+	+	+	+
+= and -= operators	+	+	+	+
Accessors to the physical parameters (mass, momentum, decay length, lifetime, rapidity, etc)	+	+	+	+
Transport: to an arbitrary point, to the decay and production points, to another particle, to a vertex, on the certain distance	+	+	+	+
Calculation of a distance: to a point, to a particle, to a vertex	+	+	+	+
Calculation of a deviation: from a point, from a particle, from a vertex	+	+	+	+
Calculation of the angle between particles	+	+	+	+
Constraints: on mass, on a production point, on a decay length	+	+	+	+
KF Particle Finder	+	+	+	+

Exactly the same package in all four experiments: CBM, PANDA, ALICE and STAR

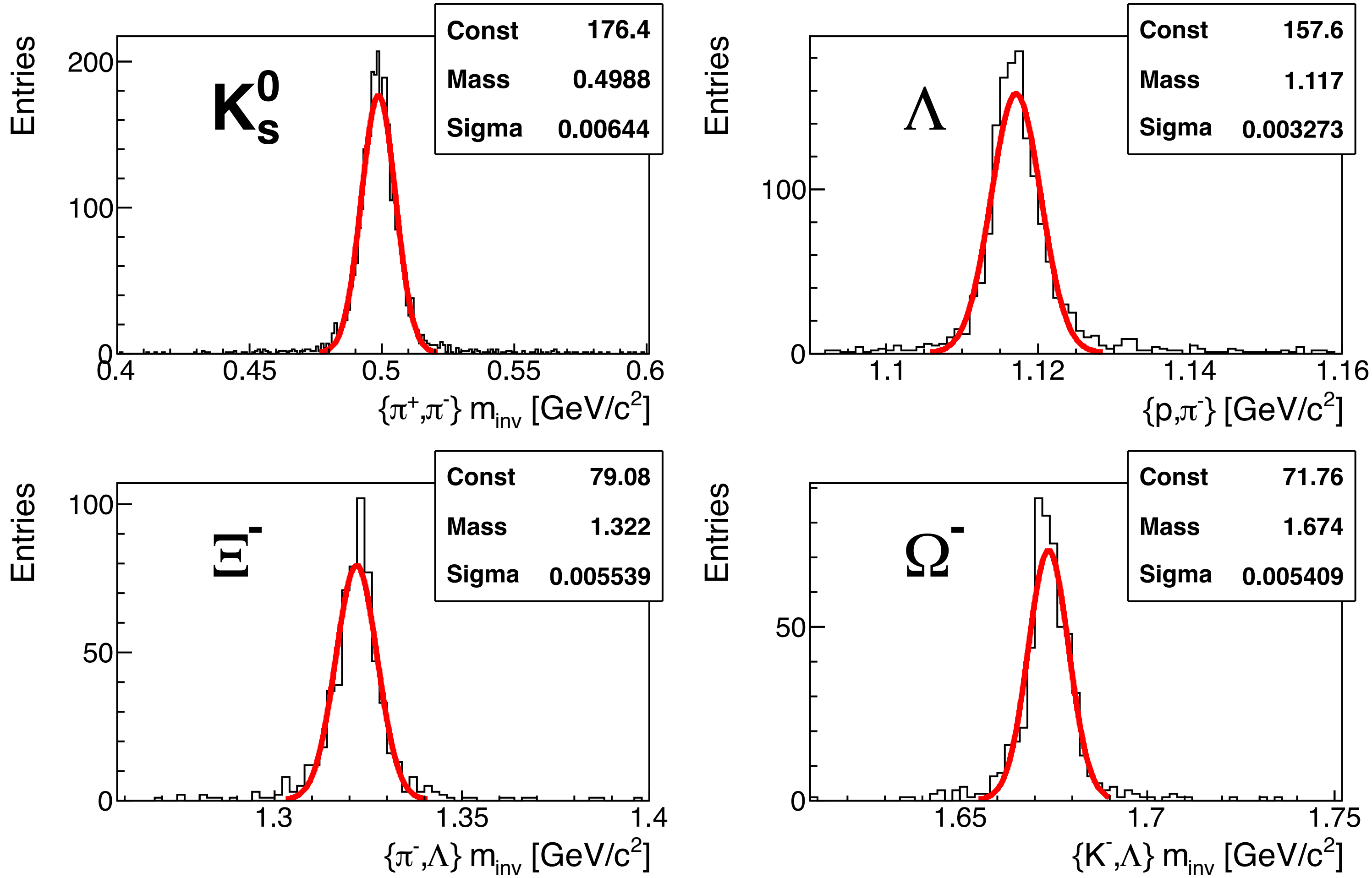
Maksym Zyzak, 09.06.2015

KF Particle Finder for Physics Analysis and Selection



Maksym Zyzak, 06.11.2018

PANDA: Reconstruction of Strange Particles

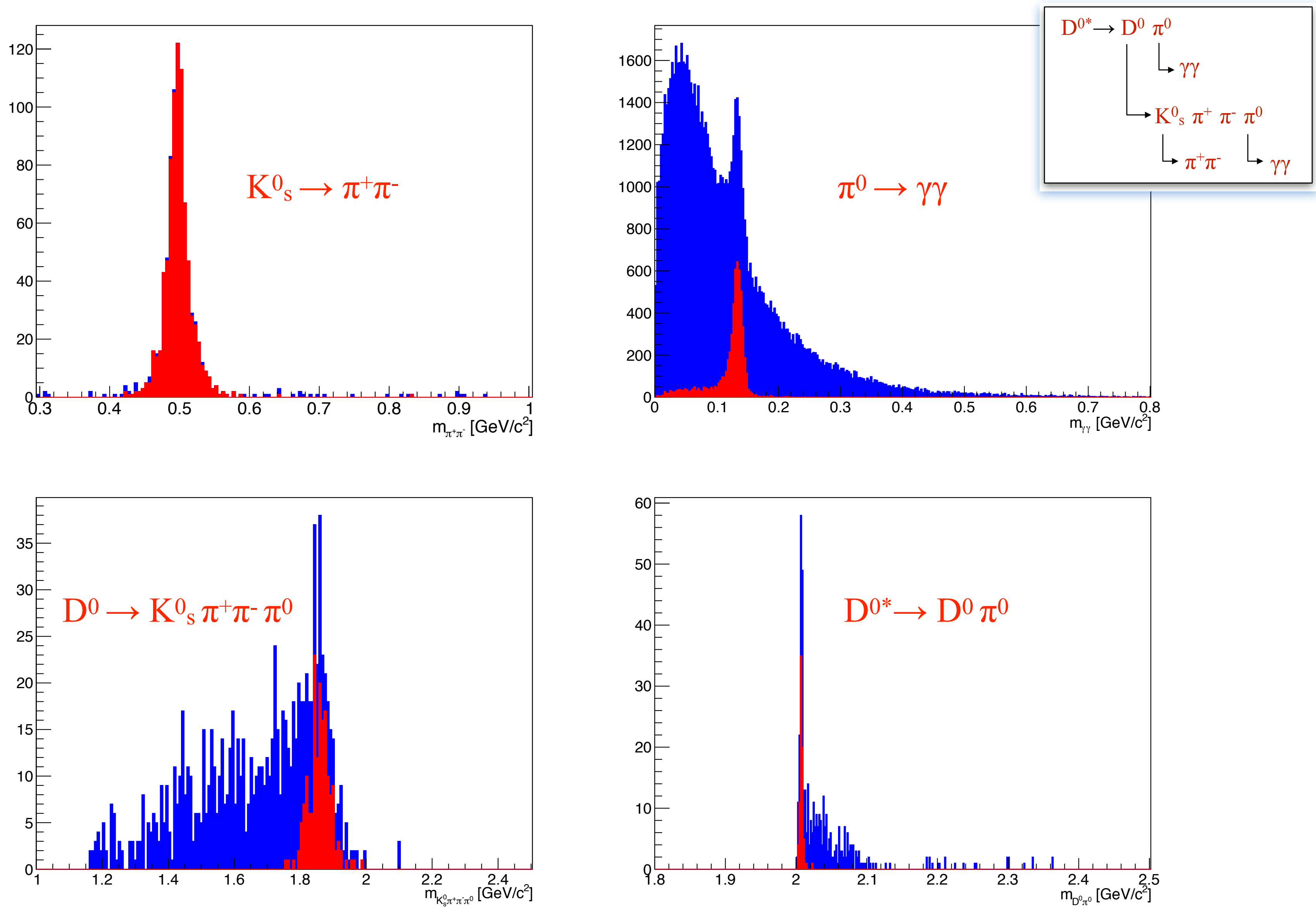


Particle	p_t, GeV/c	Efficiency	Particle	p_t, GeV/c	Efficiency
K^0_s	1	56,2 %	Λ	1	32 %
Ξ^-	2	14,1 %	Ω^-	2	17,1 %

10000 signal events, Ideal track finder, MC primary vertex

Maksym Zyzak, 11.03.2014

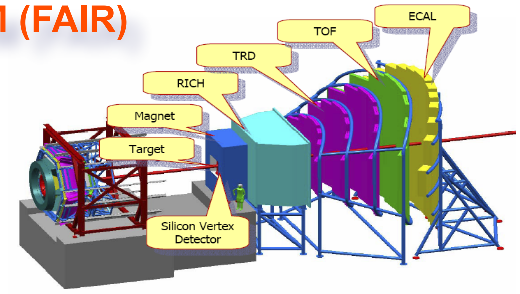
KF Particle: Reconstruction of Decay Chains



Maksym Zyzak, 09.06.2015

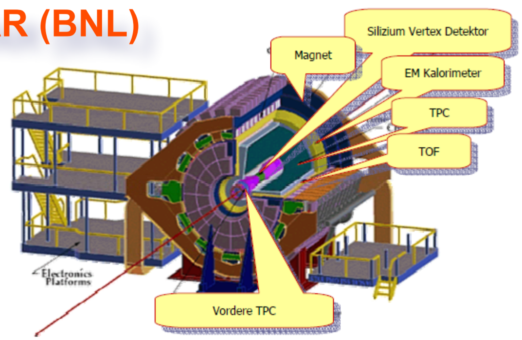
Concluding Remarks

CBM (FAIR)



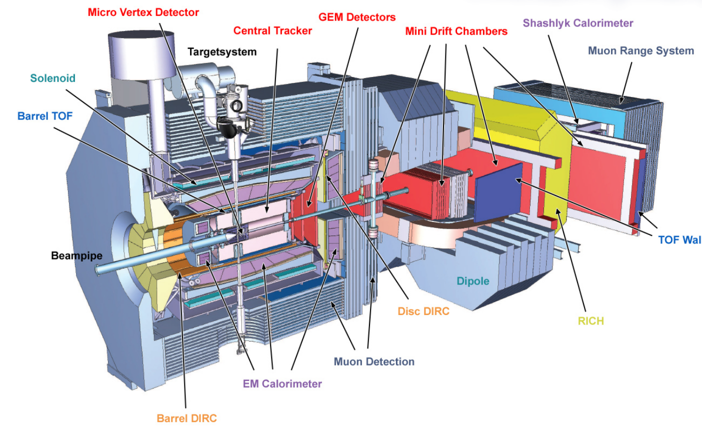
STS CA Track Finder
KF Track Fitter
KF Particle Finder

STAR (BNL)



HFT CA Track Finder
KF Particle Finder

PANDA (FAIR)



1. The idea was to share experience between different experiments and thus to develop a common reconstruction package for detectors with different technologies and geometries.
2. The Cellular Automaton for searching for particle trajectories and the Kalman Filter to estimate their parameters have a high level of intrinsic parallelism for their efficient implementation on modern and future many-core HPC architectures.
3. The event reconstruction in real time requires to redesign offline algorithms for their fast and reliable online operation, and for the maximum utilisation of the online HPC farm. With the CA track finder we have reached the reconstruction speed of 1 ms/event/core.
4. The KF Particle Finder package with more than 100 decay channels implemented (100 μs/core/decay) is a common platform, both for the offline physics analysis and for the real-time express analysis.