The KFParticle Package for Vertexing and Particle Finding

Ivan Kisel[1,2,3], Igor Kulakov [1,2,3],
Iourii Vassiliev [1,3] , Maksym Zyzak[1,2,3]

1 – Goethe-Universität Frankfurt, Frankfurt am Main, Germany
2 – Frankfurt Institute for Advanced Studies, Frankfurt am Main, Germany
3 – GSI Helmholtzzentrum für Schwerionenforschung GmbH, Darmstadt, Germany

Fourth International Workshop for Future Challenges
in Tracking and Trigger Concepts

30.11.2012

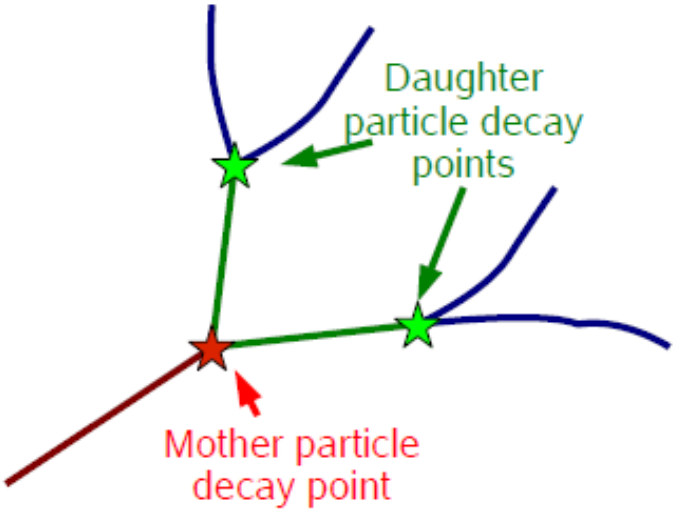
Content

- Concept and mathematics of the KFParticle package
- KF Particle finder:
  - Block-diagram
  - Options
  - Tests and results
  - Many core scalability
- Unified KFParticle:
  - Test and results
  - New functionality development
- Summary and Plans

Concept of KFParticle

Concept:

- Mother and daughter particles have the same state vector and are treated in the same way
- Geometry independent
- Kalman filter based



State vector — Position, momentum and energy

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

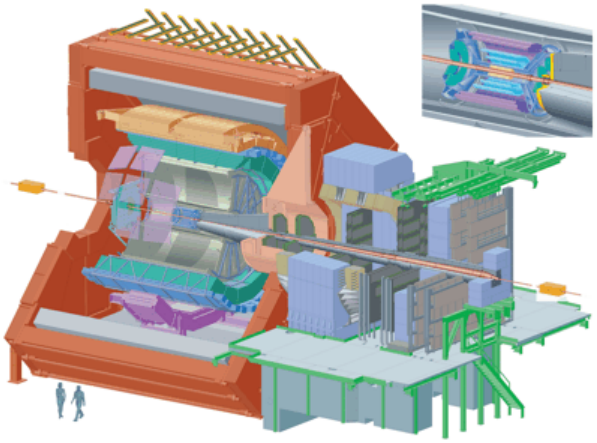
Functionality of the package:

- Construction of the particles from tracks or another particles
- Decay chains reconstruction
- Transport of the particles
- Simple access to the particle parameters and their errors
- Calculation of the distance to point

Experiments

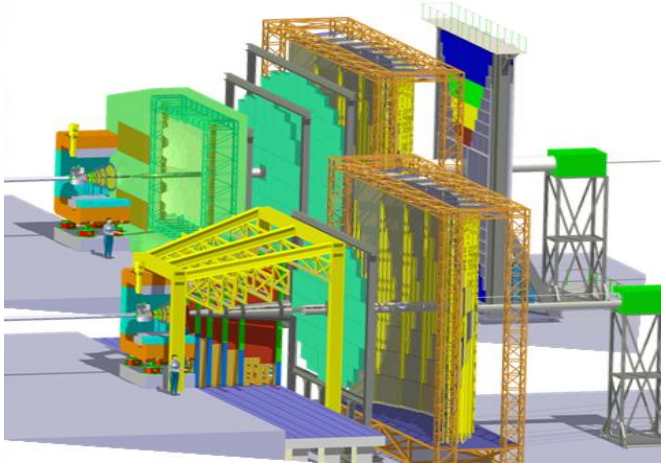
KFParticle is developed based on the ALICE and CBM experiments. In progress in the STAR experiment

ALICE (CERN, Switzerland) – a collider experiment



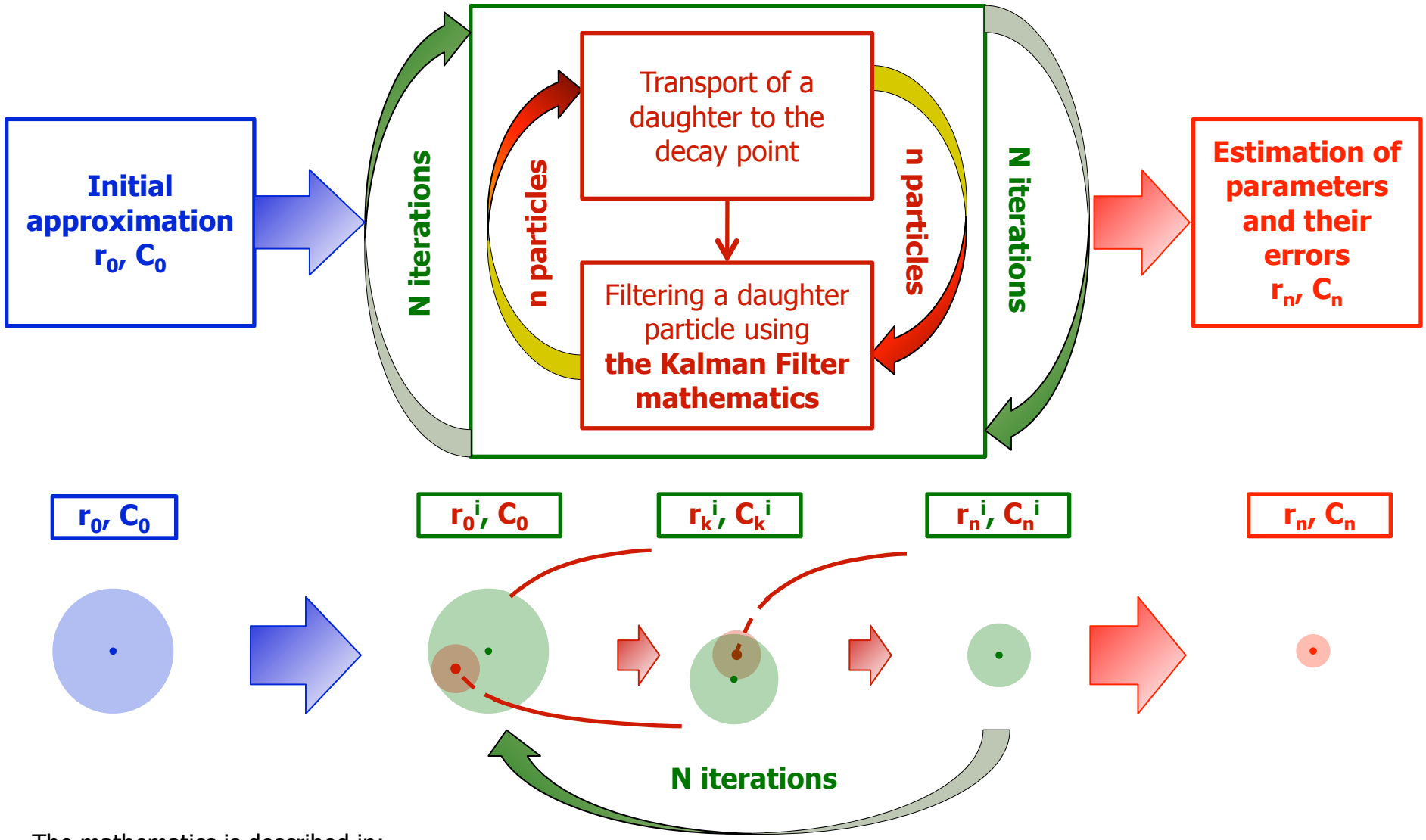
- Few 1000s charged particles/collision
- High statistic is collected – a speed of short-lived particles reconstruction is important for the physics analysis

CBM (FAIR, Germany) – a fixed-target experiment



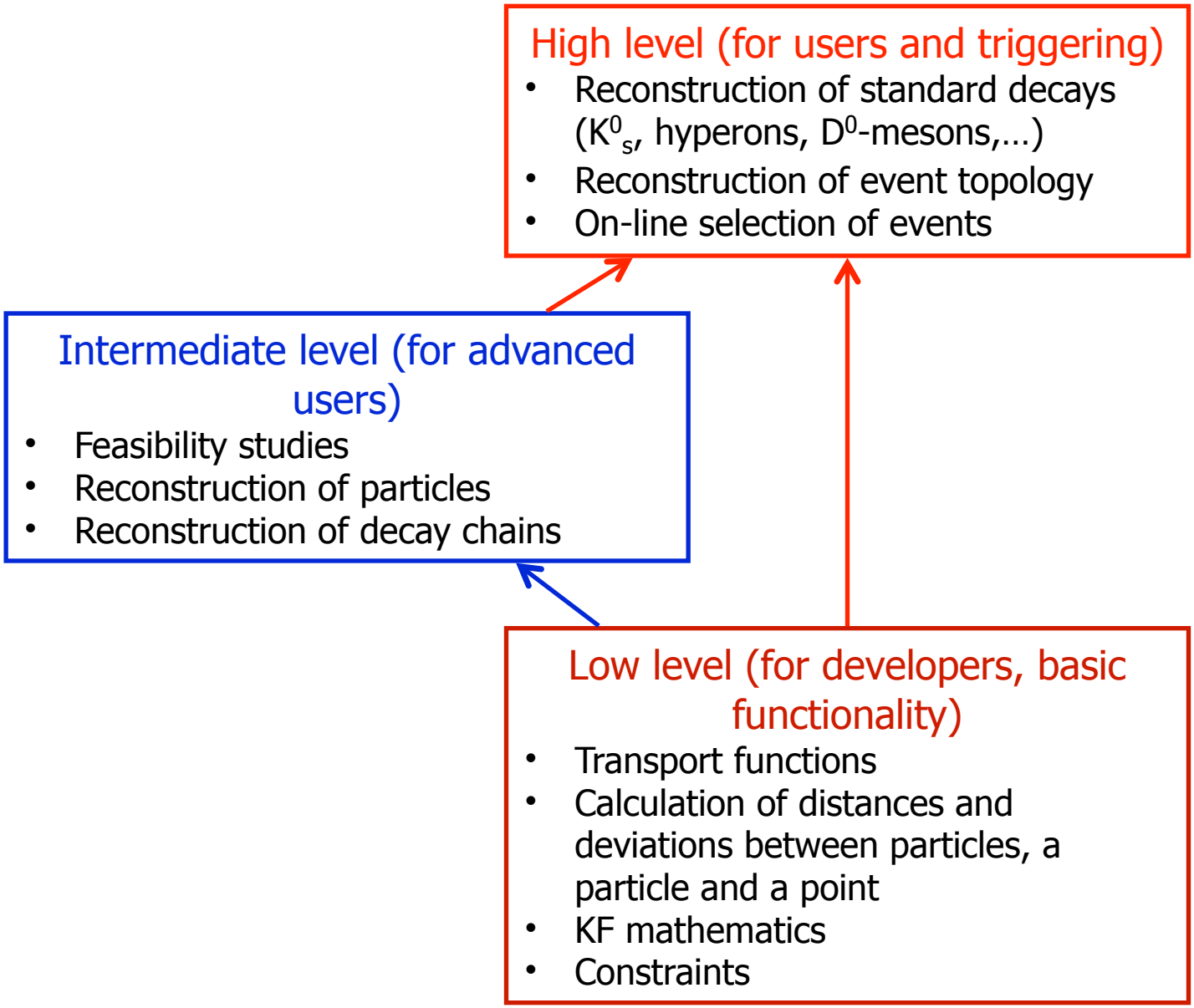
- Up to 1000 charged particles/collision
- Non-homogeneous magnetic field
- 10^7 AuAu collisions/sec
- Reconstruction of the full event topology is required in the first level trigger
- The speed and efficiency of the reconstruction is crucial

KFParticle Algorithm

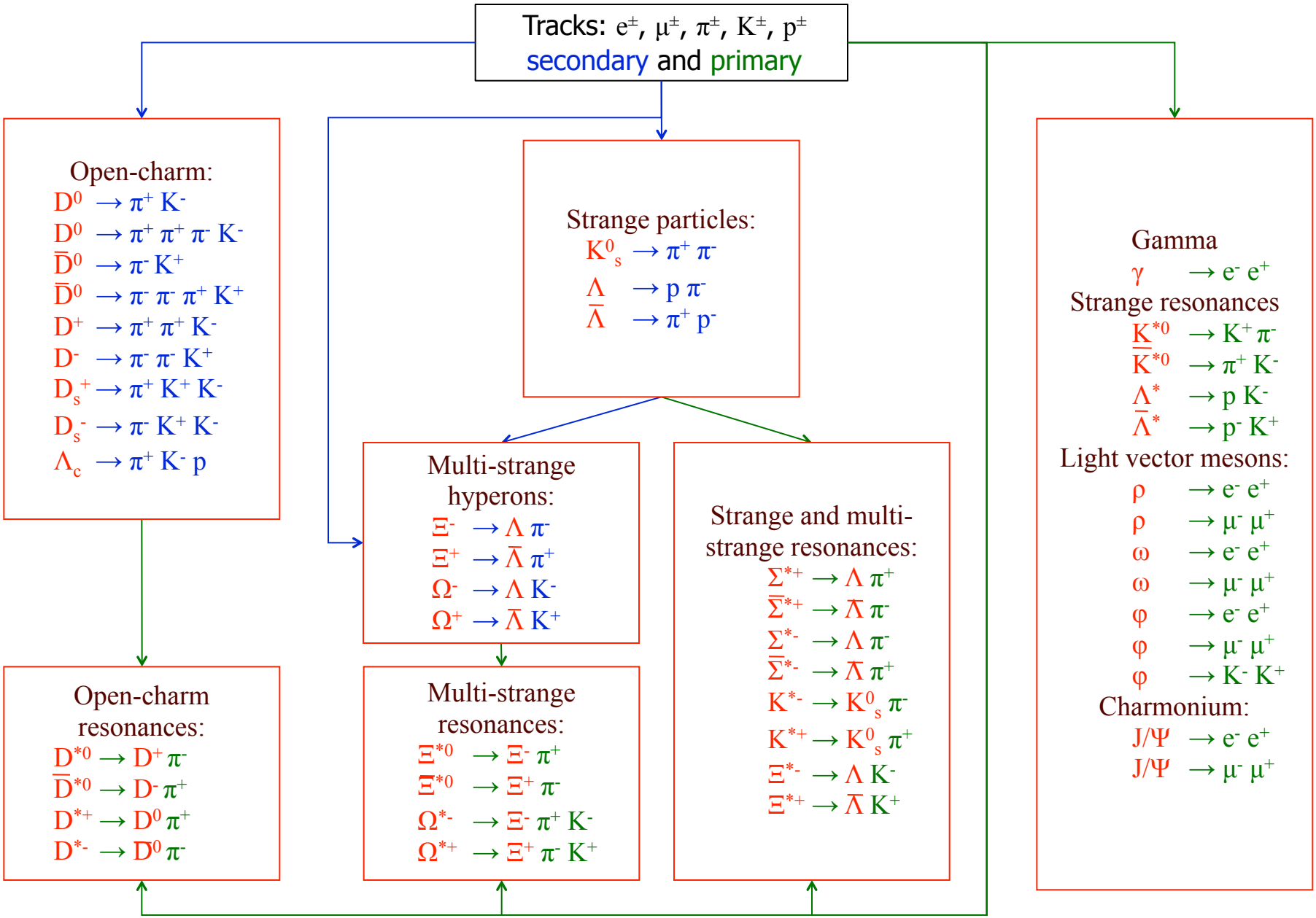


The mathematics is described in:
S. Gorbunov and I. Kisel, Reconstruction of decayed particles based on the Kalman filter.
CBM-SOFT-note-2007-003, 7 May 2007

KFParticle Development: Structure Complication

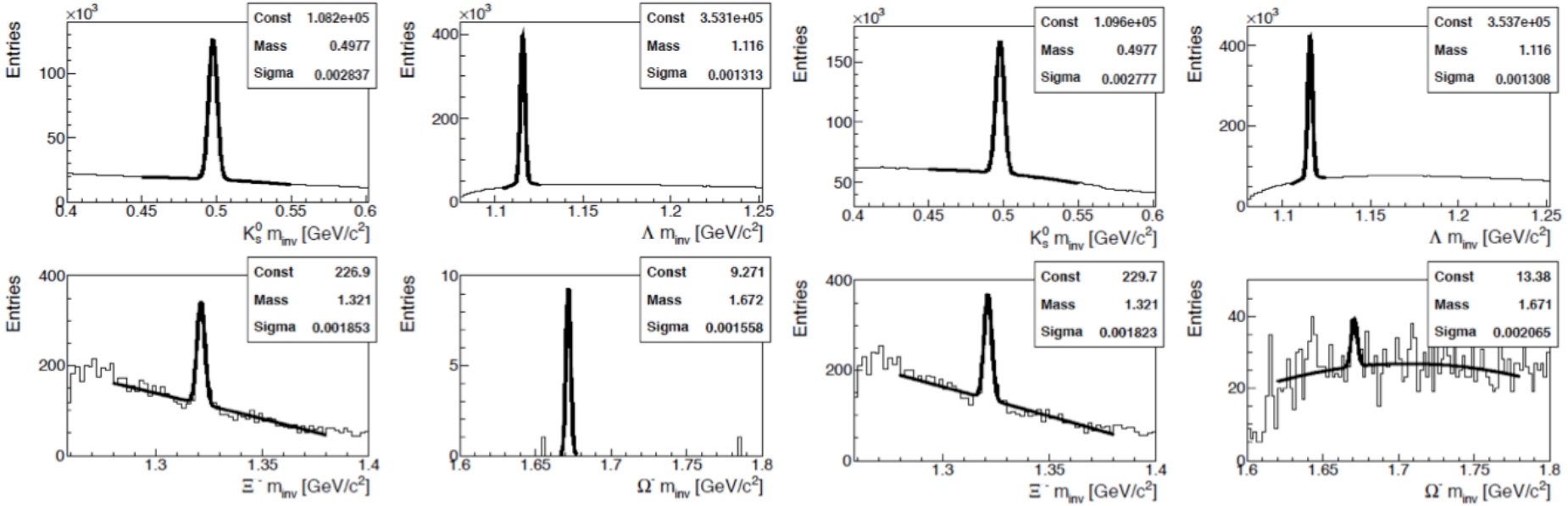


KF Particle Finder



KF Particle Finder. Options

- In the KF Particle Finder two options are foreseen: reconstruction with or without PID.
- Due to the huge combinatorial background only strange particles and hyperons are reconstructed without PID information.
- The algorithm has been developed with MC PID information.
- PID information dramatically increase S/B ratio for rare particles:



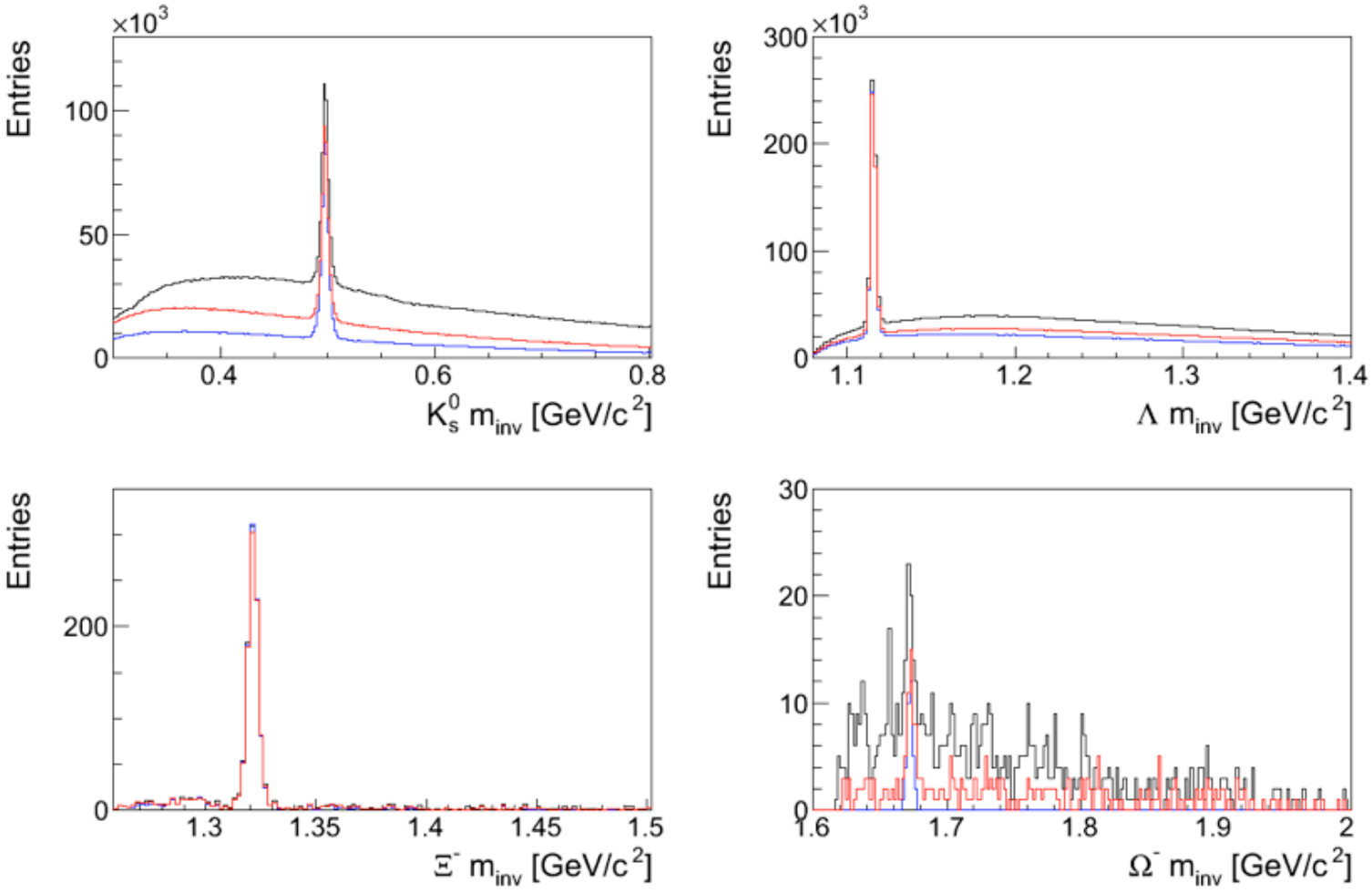
With MC PID

Without MC PID

240 mbias AuAu kevents at 25 AGeV

- The package allows to change easily values of cuts.

PID Options in KF Particle Finder

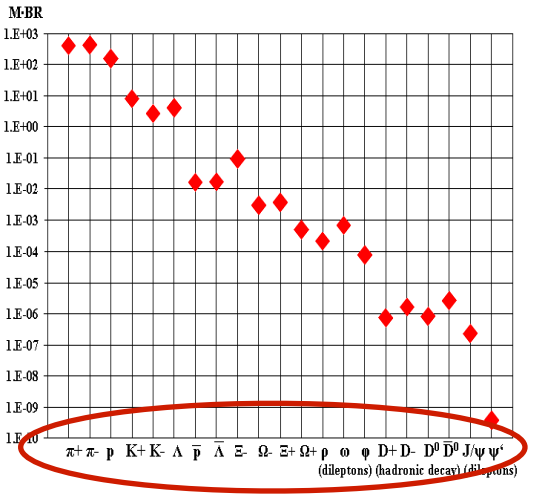


240 mbias AuAu kevents at 25 AGeV

- The possibility to use TOF PID has been added.
- Test **without PID** with **TOF PID** and **MC PID** have been performed.
- PID allows to decrease background significantly already for hyperon decays.

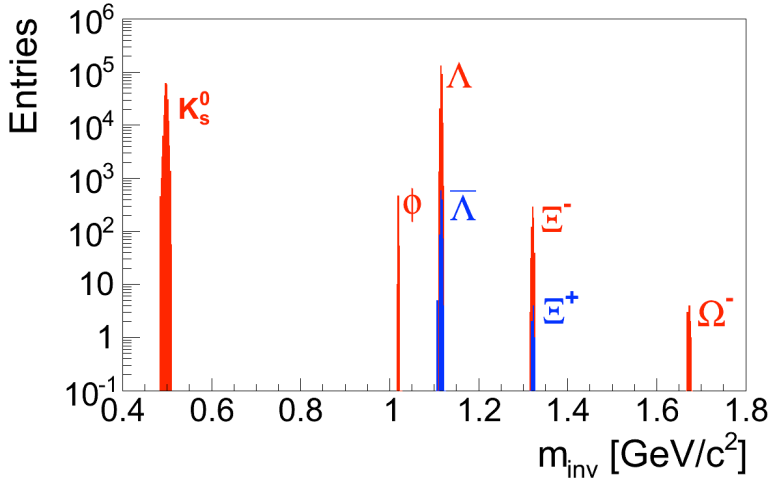
Characteristics of KF Particle Finder

Multiplicities times branching ratio, heavy ion collisions, statistical model



All decays are included to KF Particle Finder

Signal of the found particles



central AuAu collisions at 25 AGeV

The speed of the package:

- central AuAu collisions at 25 AGeV – 10.5 ms/event
- minbias AuAu collisions at 25 AGeV – 1.5 ms/event

Efficiencies of the KF Particle Finder (job summary)

Particle	:	Eff	Ghost	BackGr	N Ghost	N BackGr	N Reco	N Clone		N MC
Kshort	:	0.249	0.972	0.015	18155772	284536	242696	254	|	972992
Lambda	:	0.201	0.972	0.014	18155772	257777	269527	181	|	1341971
Lambda b	:	0.213	0.972	0.028	18155772	526299	1187	0	|	5568
Xi-	:	0.023	0.969	0.001	22934	25	708	0	|	30198
Xi+	:	0.026	1.000	0.000	21842	1	9	0	|	348
Omega-	:	0.020	0.955	0.044	8869	411	10	0	|	506
Omega+	:	0.000	0.999	0.001	9391	6	0	0	|	11

UrQMD events, central AuAu collisions at 25 AGeV, 80 kEvents, w/o PID

Temporary definitions:

Ghost – combinatorial background

BackGr – physical background

Clone – particle, reconstructed more then 1 time

Reco – correctly reconstructed

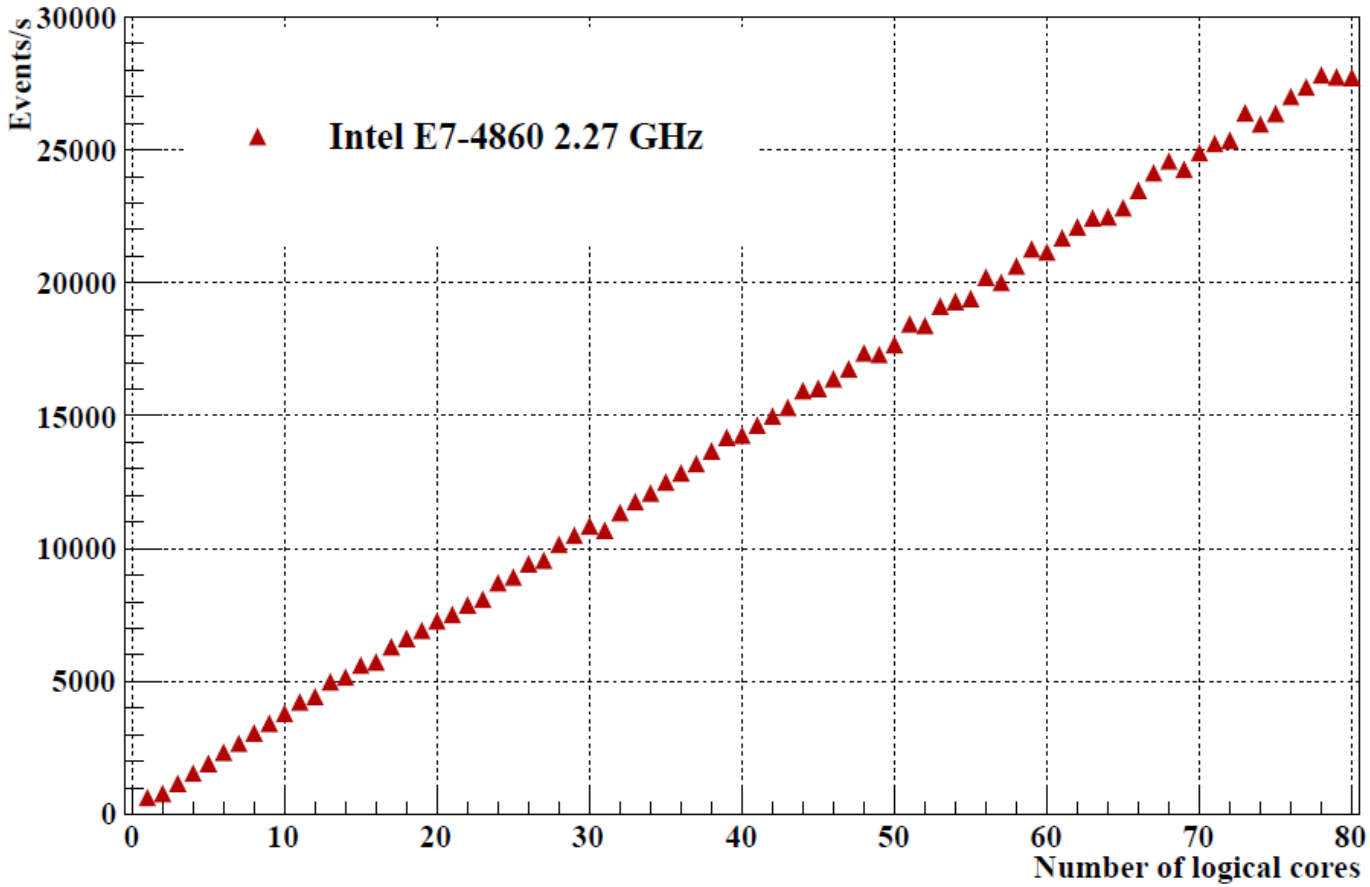
Feasibility Studies

- The feasibility studies for different systems have been performed using KF Particle Finder.
- UrQMD events have been used.
- The results using MC PID information are:

Systems	Particles
pC, pAu events at 10-30 GeV	strange particles, strange hyperons and anti-hyperons, Σ^* and K^*-resonances, φ ($K^- K^+$)
CC events at 10 and 25 AGeV	strange particles, strange hyperons and anti-hyperons, Σ^* and K^*-resonances, φ ($K^- K^+$)
AuAu events at 11 and 25 AGeV	strange particles, strange hyperons and anti-hyperons, φ ($K^- K^+$)

Scalability on Many-core System

- The KF Particle Finder has been parallelized using Intel TBB.
- The KF Particle Finder shows linear scalability on many-core machines (the scalability on a computer with 80 cores is shown).



Given n threads each filled with 1000 events,
run them on specific n logical cores, 1 thread per 1 core.
AuAu mbias events at 25 AGeV

Unification of ALICE and CBM KFParticle

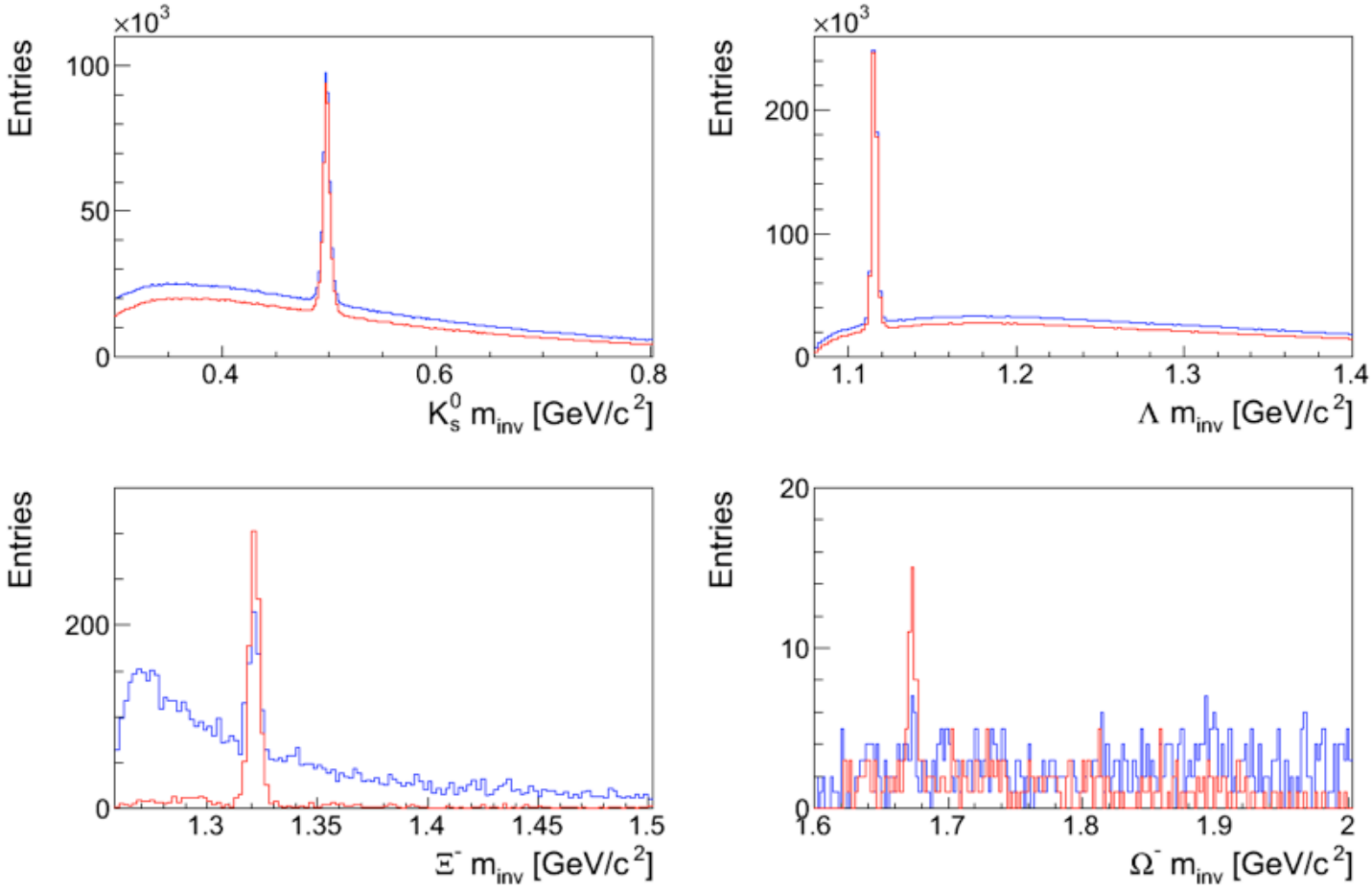
- The ALICE and CBM KFParticle have been unified.
- The unified package contains all the functionality from both versions.
- During unification the implementation of the functionality has been improved.
- The package is fully vectorized.

Functionality in ALICE and CBM

Functions	ALICE	CBM
Construct, SetMassConstraint, SetProductionVertex, SetVtxGuess	+	+
GetMass, GetMomentum, GetDecayLength, GetLifeTime	+	+
GetDecayLengthXY, GetPhi, GetR	+	+
Extrapolate, TransportToProductionVertex(), TransportToDecayVertex()	+	+
TransportToPoint, TransportToVertex, TransportToParticle, TransportToDS,	+	+
GetDStoPoint	+	+
GetDStoParticle, GetDStoParticleXY, GetDistanceFromVertex, GetDistanceFromVertexXY, GetDistanceFromParticle, GetDistanceFromParticleXY, GetDeviationFromVertex, GetDeviationFromVertexXY, GetDeviationFromParticle, GetDeviationFromParticleXY	+	+
GetAngle, GetAngleXY, GetAngleRZ	+	+
SubtractFromVertex, SubtractFromParticle, ConstructGamma	+	+
SetNoDecayLength, +=, -=	+	+
KF Particle Finder	-	+

Functionality become more and more advanced

KF Particle Finder: Unified vs. CBM



240 mbias AuAu kevents at 25 AGeV

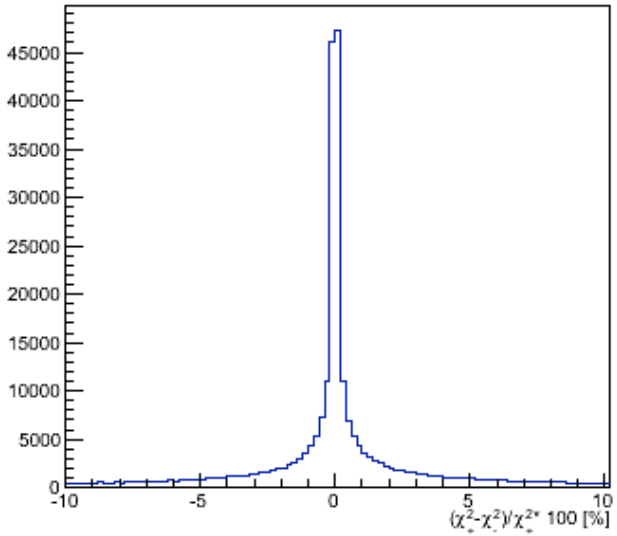
- The Unified and CBM KF Particle Finders have been compared.
- Improvements in the implementation allow to increase the efficiency of hyperon reconstruction by factor of 2 and dramatically decrease the background.

New Functionality. SubtractFromParticle

- New function for subtraction of a daughter from a mother particle has been added.
- The code can be written with += and -= operators:

```
KFParticleSIMD mother;
mother += d1;
mother += d2;
mother.SetVtxGuess(mother.X(), mother.Y(), mother.Z()); //for the stabilization
mother += d3;
mother -= d4;
mother += d5;
```

- The numerical stability in single precision is under investigation:



Summary and Plans

Summary

- The KFParticle package is a particle reconstruction package with a rich functionality. The functionality becomes more and more advanced.
- The KF Particle Finder has been developed based on the SIMDized KFParticle package. About 50 particles (decay channels) are included.
- The algorithm shows high speed (1.5 ms per mbias AuAu event at 25 AGeV) and efficiency, shows linear scalability on many-core systems.
- The ALICE and CBM KFParticle have been unified. The package shows increased stability and performance.

Future plans

- KF Particle Finder for ALICE.
- Increase the functionality of the package, create the KFParticle library.
- Add user code to KF Particle Finder and help to speed it up.
- Add adaptive methods (DAF, PDAF, etc.) to KFParticle.
- Implement using parallel languages (ArBB and OpenCL), implement on GPUs.