Event Reconstruction
with High-Performance Computing

Ivan Kisel

Goethe University Frankfurt am Main
FIAS Frankfurt Institute for Advanced Studies
GSI Helmholtz Center for Heavy Ion Research

- 10^7 collisions/second
- A very high track density
- 1000 charged particles/collision
- A non-homogeneous magnetic field
- The first plane has only 5 cm diameter
- The silicon detector is only 1 m long

CBM experiment at FAIR/GSI
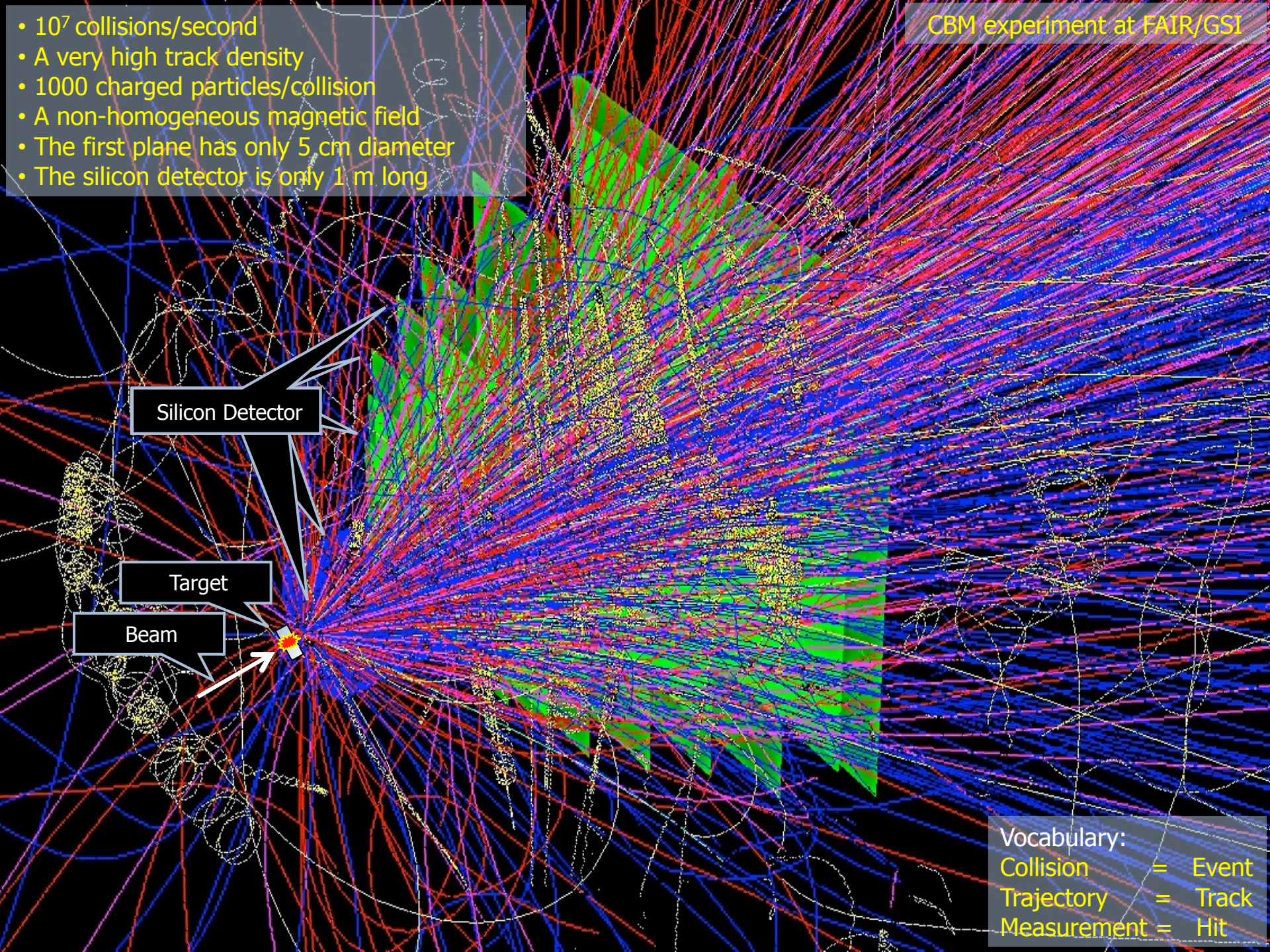
Silicon Detector

Target

Beam

Vocabulary:
Collision = Event
Trajectory = Track
Measurement = Hit

HEP Experiments: Collider and Fixed-Target

Beam → ← Beam

Beam → Target
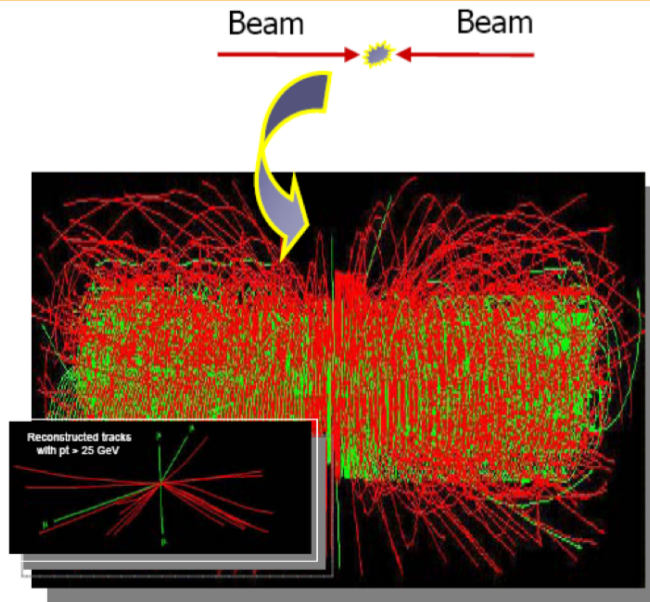
Inelastic collisions
$10^7 - 10^9$

10^{11}

Signal events
$10^2 - 10^{-2}$

Reconstructed tracks with pt > 25 GeV
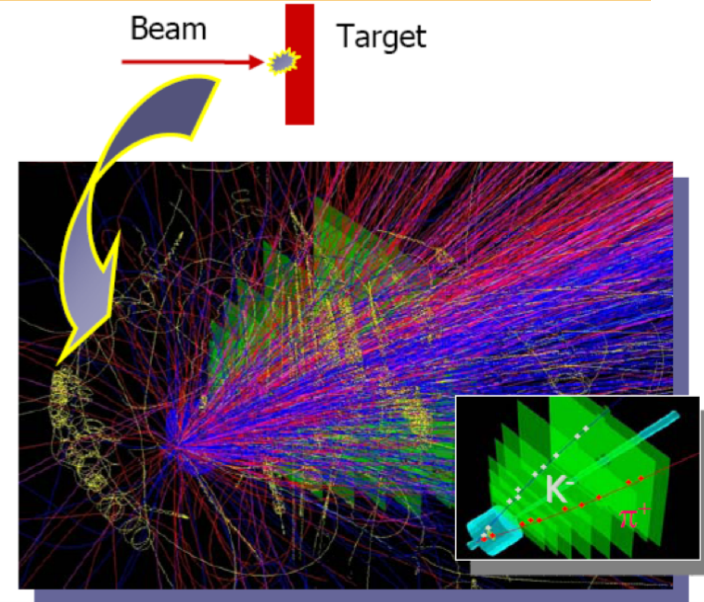
High energy = high density + high rate

ALICE (CERN)

Magnet
PID
TPC
Inner Tracker
Muon Chambers

CBM (FAIR/GSI)

ECAL
TOF
TRD
RICH
Magnet
Target
Silicon Vertex Detector
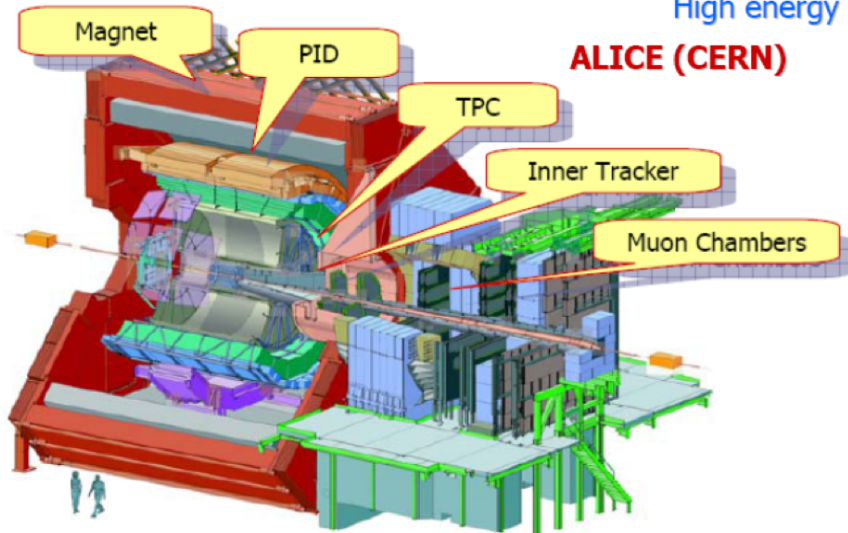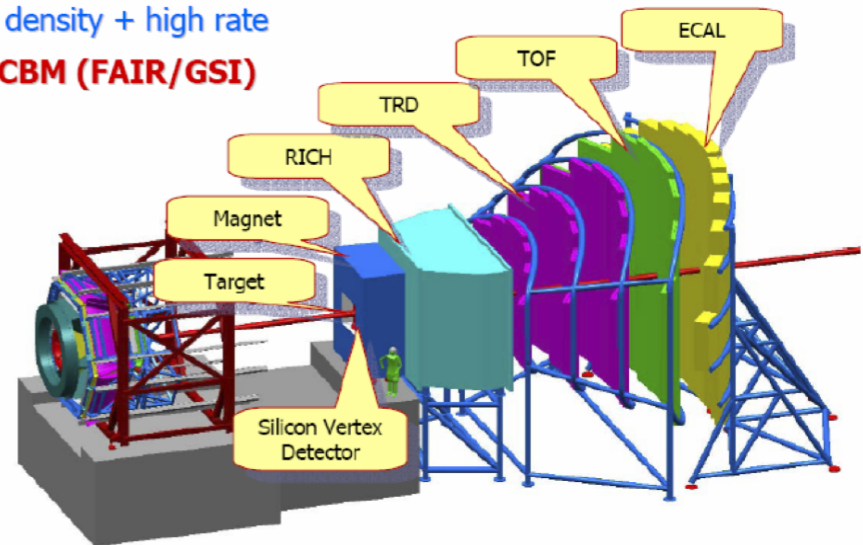
HEP Experiments: select interesting physics on-line

Modern detectors consist of many different pieces of equipment which test for different aspects of an event. These many components are arranged in such a way that we can obtain the most data about the particles spawned by an event. In this way we can figure out the type of particle (PID) based on where that particle appeared in the detector.

Tracking Chambers: The inner region of the detector is filled with highly segmented sensing devices of various kinds, so that charged particle trajectories can be very accurately determined.

Electromagnetic Calorimeter: This device measures the total energy of e^+, e^-, and photons. These particles produce showers of e^+/e^- pairs in the material. The e^-'s (or e^+'s) are deflected by the electric fields of atoms, causing them to radiate photons. The photons then make e^+/e^- pairs, which then radiate photons, etc. The number of final e^+/e^- pairs is proportional to the energy of the initiating particle.

Hadron Calorimeter: This device measures the total energy of hadrons. The hadrons interact with the dense material in this region, producing a shower of charged particles. The energy that these charged particles deposit is then measured.

Magnet: The path of a charged particle curves in a magnetic field. The radius of curvature and direction tell the momentum and the sign of the charge.

Muon Chambers: Only muons and neutrinos get this far. The muons are detected, but the weakly interacting neutrinos escape. The presence of neutrinos can be inferred by the "missing" energy.
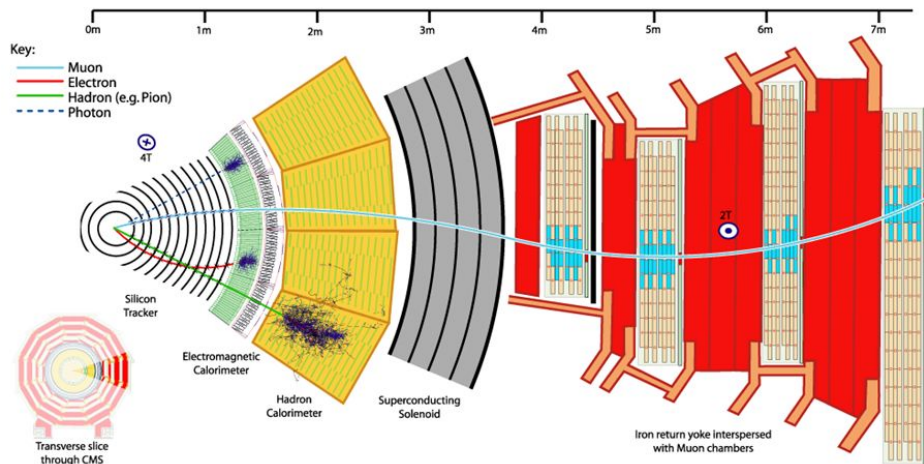
Reconstruction Challenge in CBM at FAIR/GSI

(1) Collision

(2) Detection
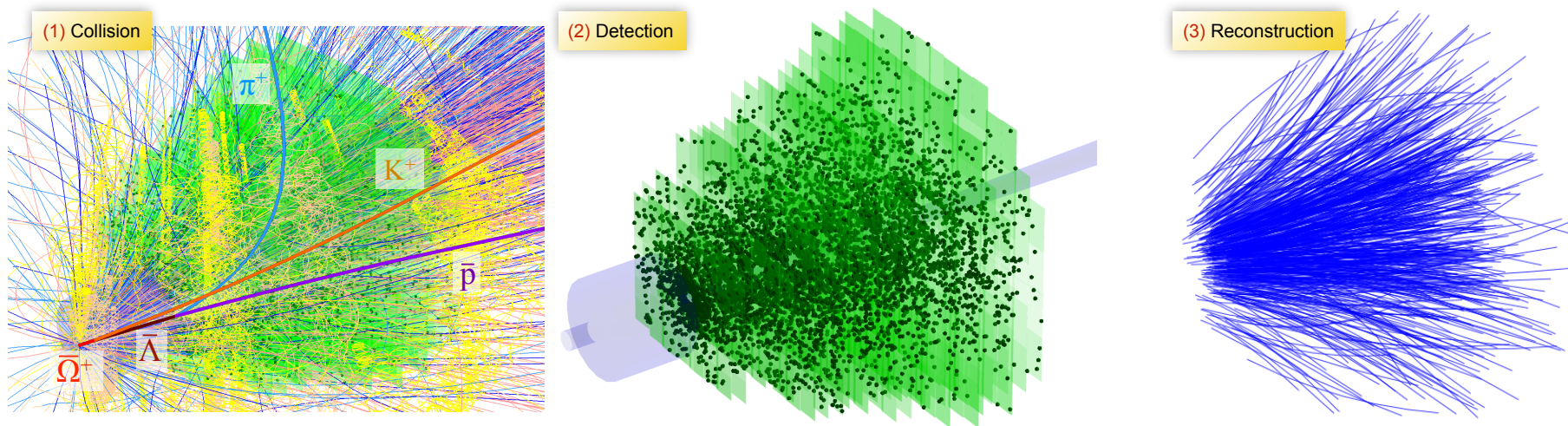
(3) Reconstruction

- Future fixed-target heavy-ion experiment
- 10^7 Au+Au collisions/sec
- ~ 1000 charged particles/collision
- Non-homogeneous magnetic field
- Double-sided strip detectors (85% fake space-points)

Full event reconstruction will be done
on-line at the First-Level Event Selection (FLES) and
off-line using the same FLES reconstruction package.

Cellular Automaton (CA) Track Finder
Kalman Filter (KF) Track Fitter
KF short-lived Particle Finder

All reconstruction algorithms are vectorized and parallelized.

ECAL

TOF

TRD

RICH

Magnet

Target

Silicon Vertex
Detector
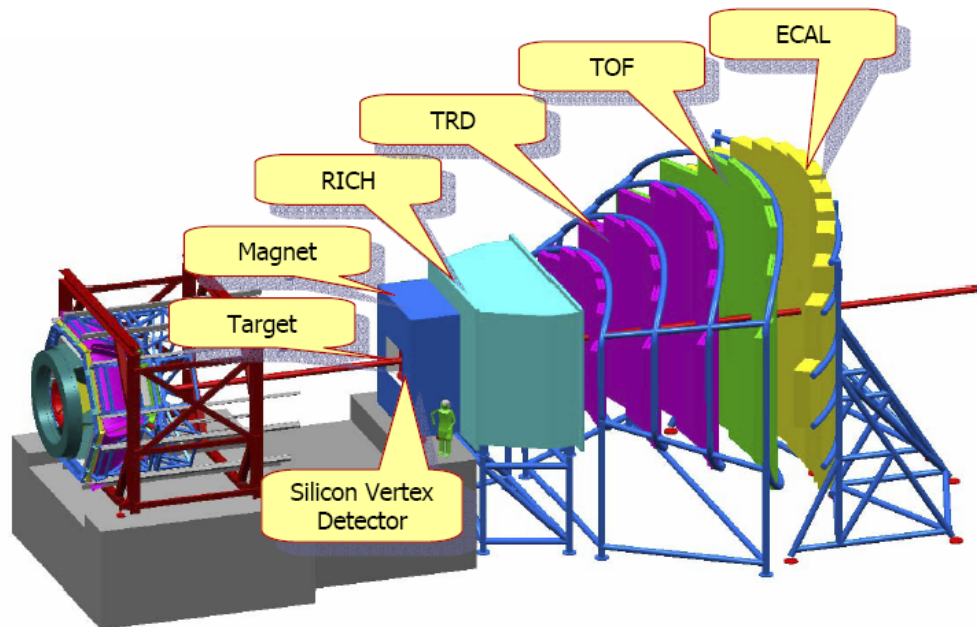
Stages of Event Reconstruction

1 **Track Finder**
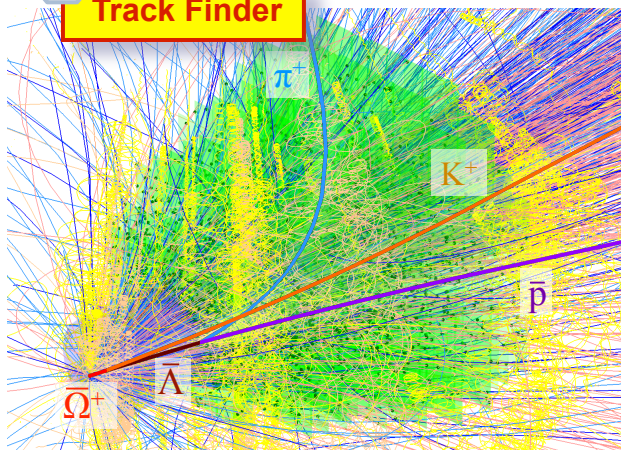
- Conformal Mapping
- Hough Transformation
- Track Following
- Cellular Automaton

2 **Track Fitter**

- Kalman Filter

3 **Short-Lived Particles Finder**
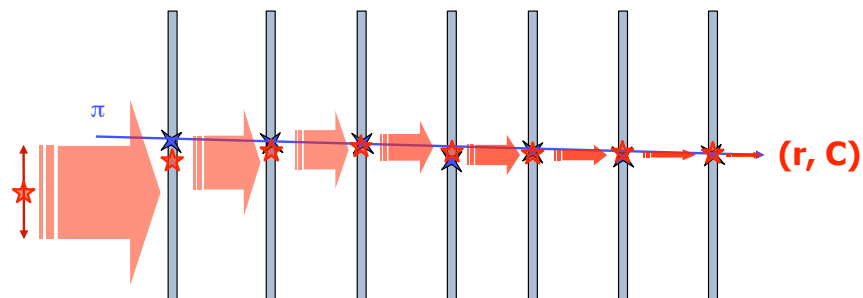
- Kalman Filter

4 **Online Physics Analysis**
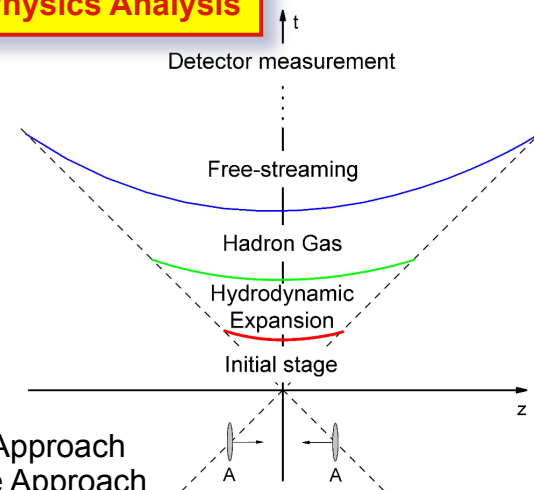
- Direct Approach
- Inverse Approach

Track Finding: Rubik's Cube in CBM
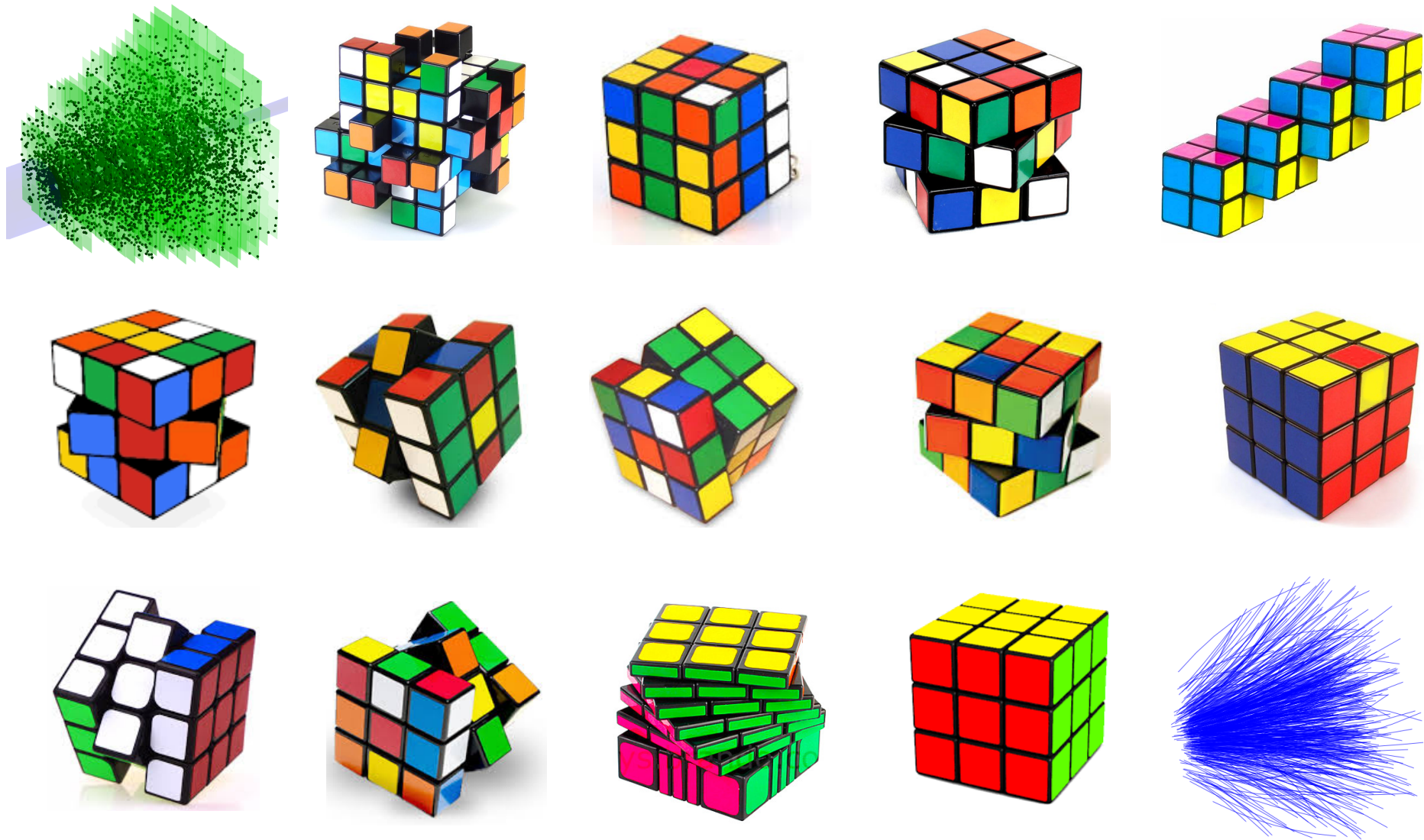
The world record times are: 4.73 seconds/Rubik's cube by Feliks Zemdegs (Australia) and
0.0000045 seconds/CBM cube by FAIR-Russia Research Center HPC (Moscow)

Conformal Mapping:
Transform circles into straight lines
$$u = x/(x^2+y^2)$$
$$v = -y/(x^2+y^2)$$
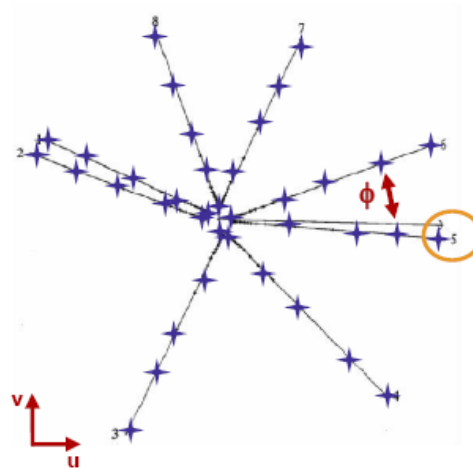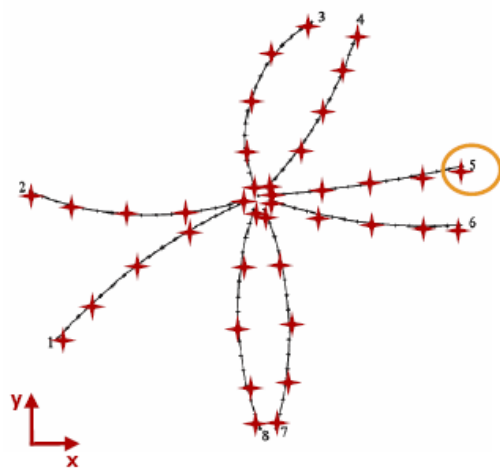
Histogram:
Collect a histogram of azimuth angles ϕ
Find peaks in the histogram
Collect hits into tracks
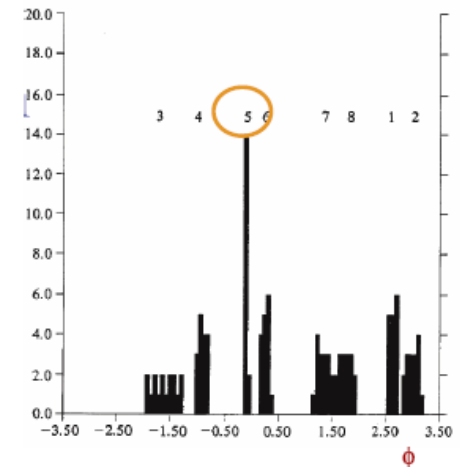
Strong features:
- Impressive visual simplification of the problem
- Each step is easy to implement in hardware
- This results in a fast algorithm
- ...

Weak points:
- Needs to know exact position of the interaction point
- Gives approximate track parameters only
- One has to collect hits into track candidates
- Finds only primary tracks
- No secondary (physics) tracks found
- ...

Useful implemented in hardware and for very simple event topologies only

Measurement Space

$$y = a*x + b$$

Parameter Space

$$b = -x*a + y$$
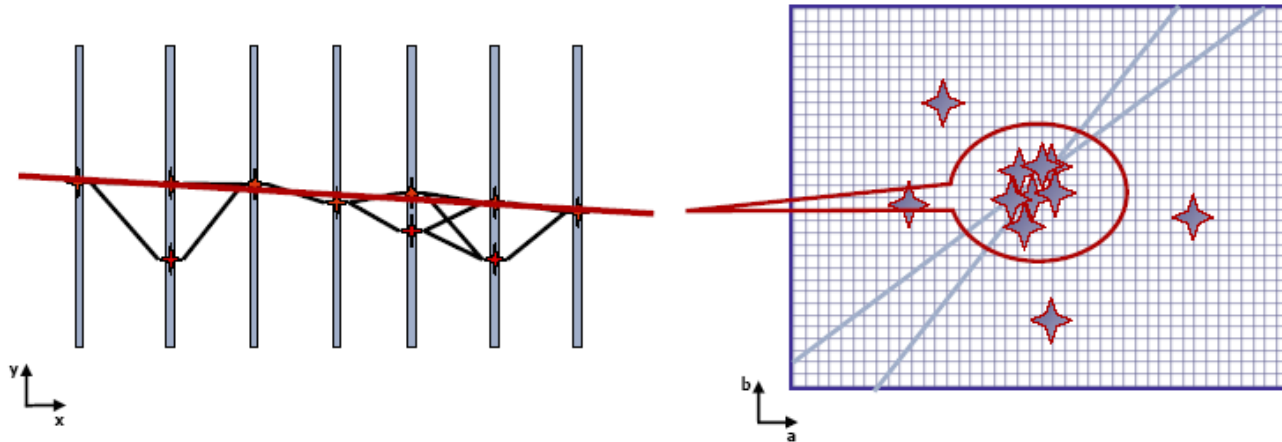
Strong features:
- Generalization of the histogramming method
- Easy to implement in hardware
- This results in a fast algorithm
- ...

Weak points:
- Histogramming provides only track parameters etc.
- Needs a global track model
- Therefore, appropriate for simple magnetic fields only
- Does not include multiple scattering -> only fast tracks
- Histogramming needs access to main memory -> slow
- 5D histogramming (x, y, tx, ty, q/p) needs a lot of memory
- Precise tracking requires even more memory
- ...

Useful implemented in hardware and for simple event and trigger topologies

measurements

detectors

π

(r, C)

track parameters and errors

Initial estimates for R_0 and Γ_0

$\tilde{R}_{k+1}, \tilde{\Gamma}_{k+1}$

Prediction step

Filtering step

$\hat{R}_k, \hat{\Gamma}_k$

State estimate $\hat{R}$
Error covariance $\hat{\Gamma}$

KF Fit

KF Find

Seeding Planes
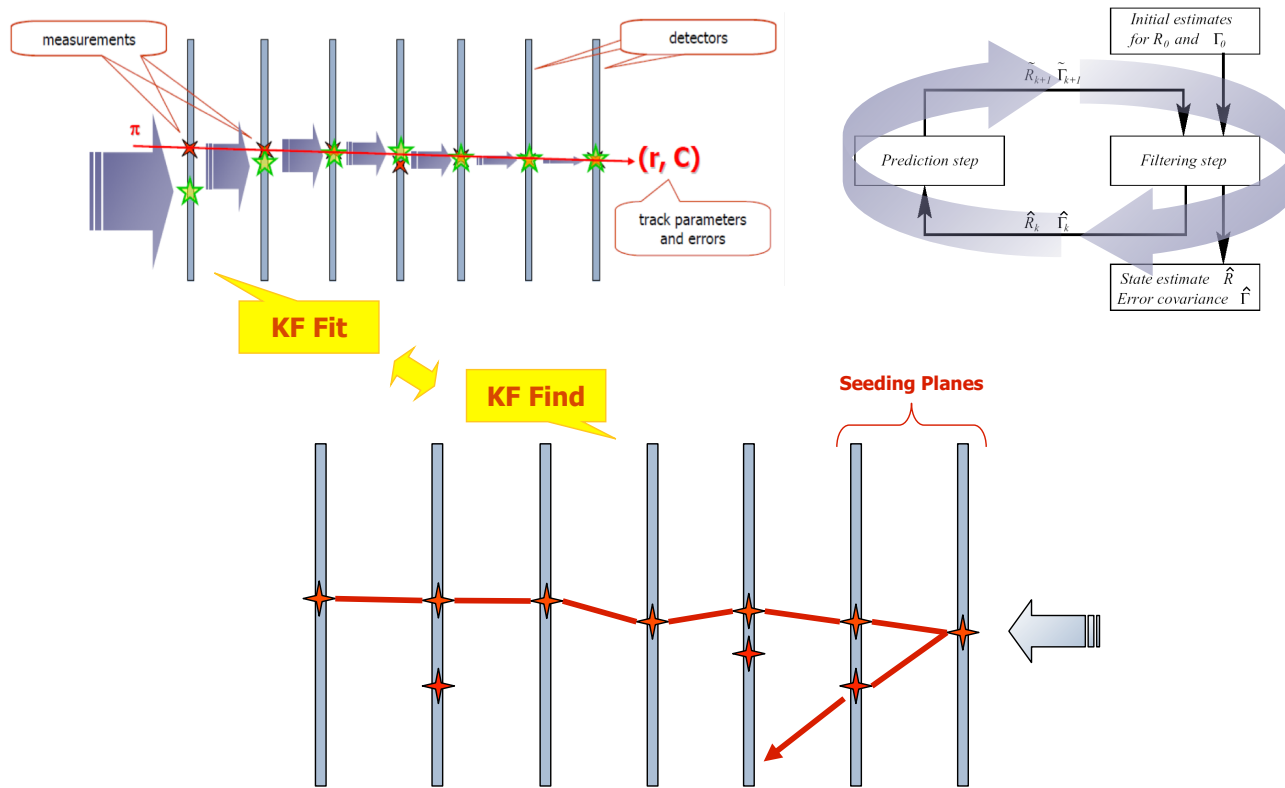
Strong features:
- Psychologically easy to accept hit by hit track finding
- Combined track finder and fitter based on KF
- Development of a new experiment starts with an ideal MC track finder and a realistic KF track fitter, therefore the next step to a realistic track finder is obvious – KF
- ...

Weak points:
- Based on a single track approach
- Needs seeding (starting short track segments)
- A final track finding efficiency is limited by the seeding efficiency
- It is also limited by the efficiency of the seeding detector planes
- How many inefficient detector planes can be tolerated?
- Complicated KF calculations, then discarding track candidates
- Too early competition decision to accept/discard a track candidate
- ...

Useful for relatively simple event topologies and as a second after the ideal track finder
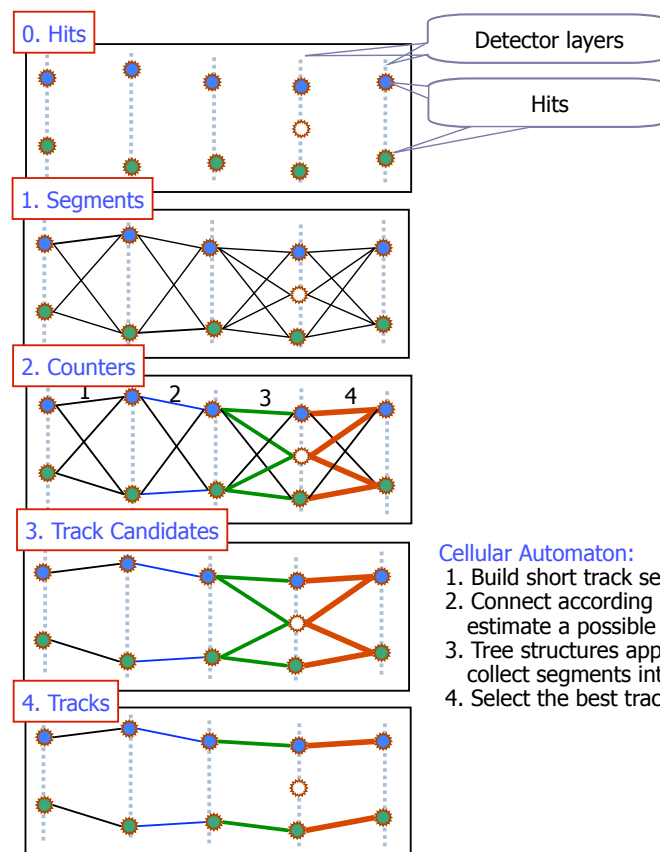
1

0. Hits

Detector layers

Hits

1. Segments

2. Counters

1   2   3   4

3. Track Candidates

4. Tracks

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.

Strong features:
- Local relations -> simple and fast calculations
- Local relations -> parallel algorithm
- Staged implementation: hits -> segments -> tracks
- Track finder efficiency monitoring at each stage
- Track competition at the global level
- Includes the KF fitter, if necessary, for high track densities
- Detector inefficiency problem is located out the combinatorics
- ...

Weak points:
- Not easy to understand a parallel algorithm (Game of Life)
- Currently implementations on sequential computers
- Parallel hardware is coming now
- ...

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

STAR TPC CA Track Finder

Au-Au event with 1446 tracks   Front view   Side view

Efficiencies for Global

CA+Sti tracker
Sti tracker

| Old efficiency for Mult > 600 |
| New efficiency for Mult > 600 |
| Old efficiency for 400 < Mult < 600 |
| New efficiency for 400 < Mult < 600 |
| Old efficiency for 200 < Mult < 400 |
| New efficiency for 200 < Mult < 400 |
| Old efficiency for 0 < Mult < 200 |
| New efficiency for 0 < Mult < 200 |
| Old efficiency for ALL |
| New efficiency for ALL |

Tue Aug 24 16:07:41 2010
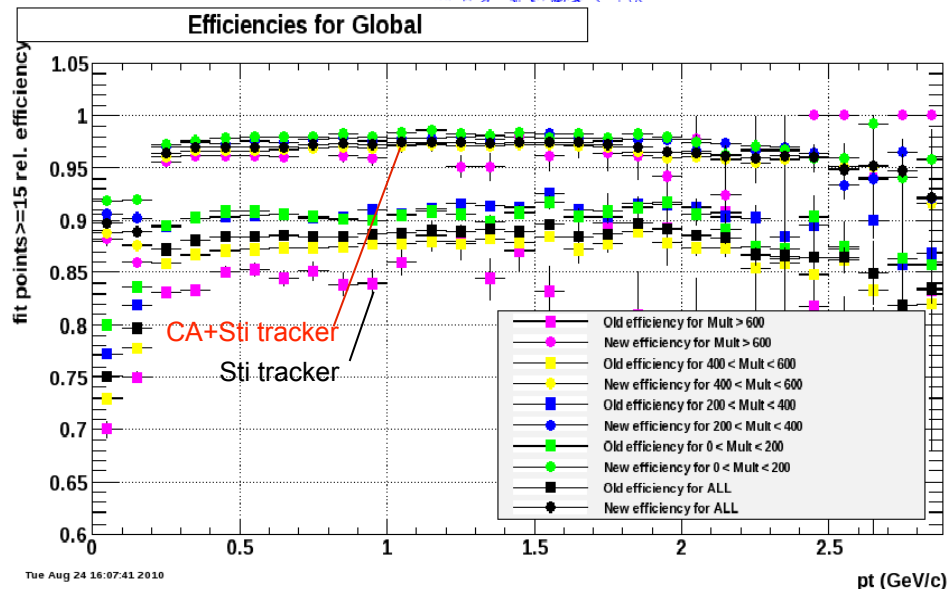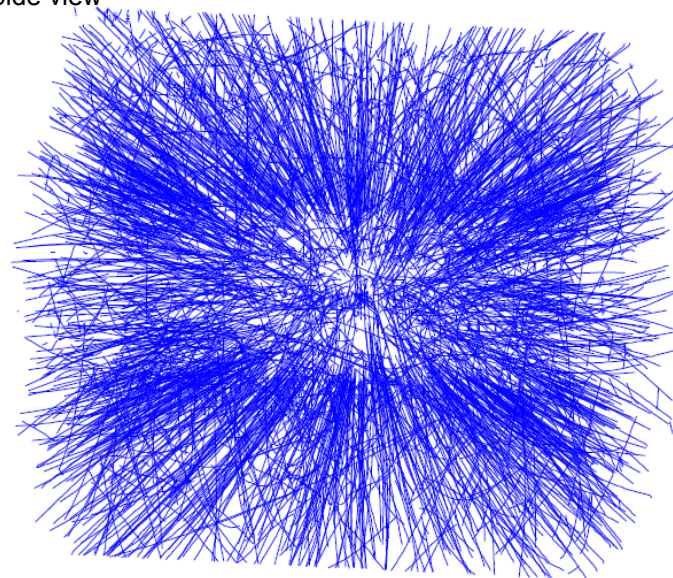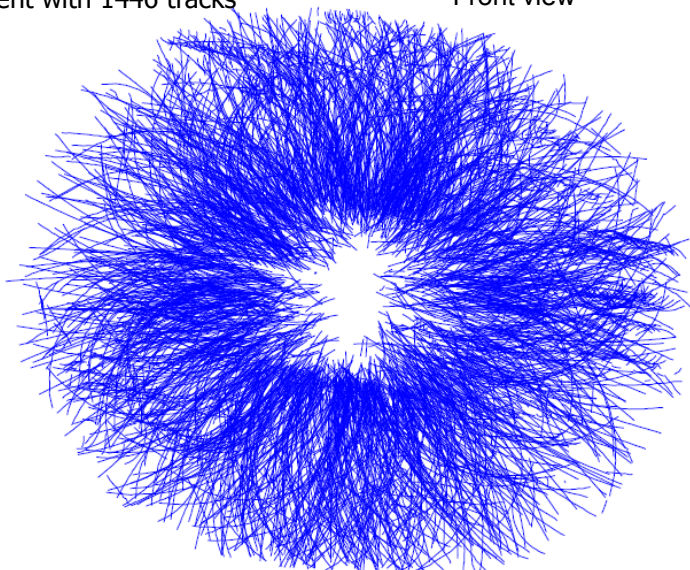
Efficiency and ratio, %	
Ref Set	96.6
All Set	88.6
Clone	10.6
Ghost	12.6
Tracks/ev	659
Time/ev, ms	47

All set: p ≥ 0.05 GeV/c
Reference set: p ≥ 1 GeV/c
Ghost: purity < 90%

The CA track finder is more stable w.r.t. track multiplicity and is ~10 times faster than the TF based Sti track finder.

Y2004_AuAu200
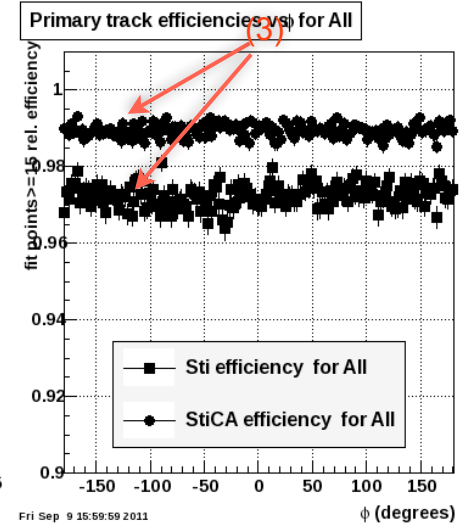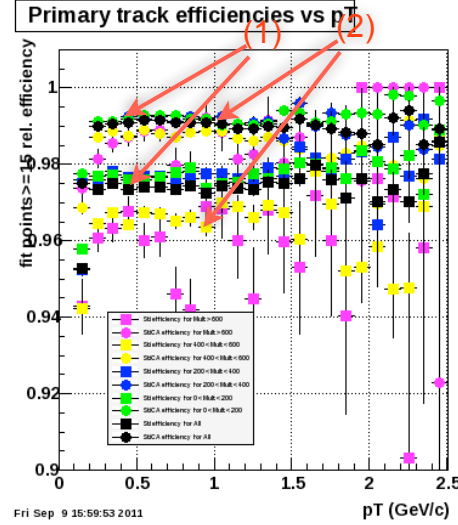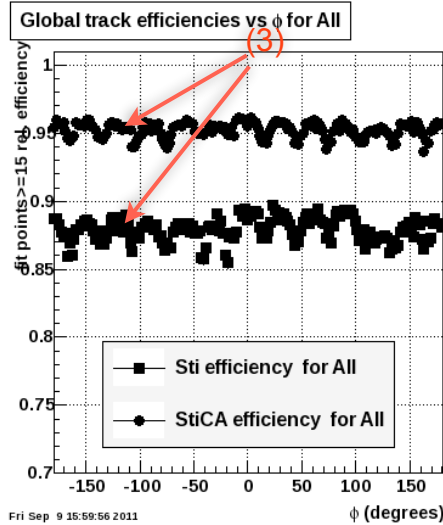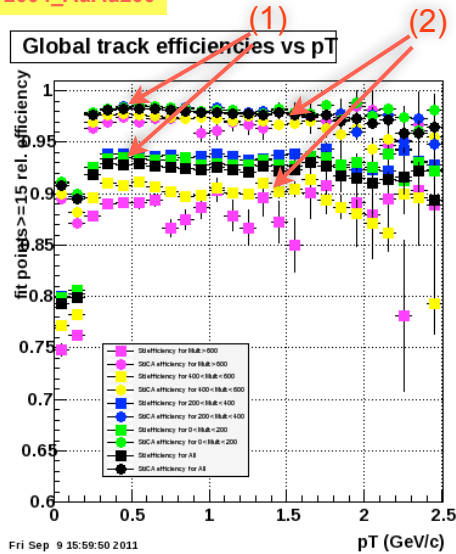
Y2010_AuAu200
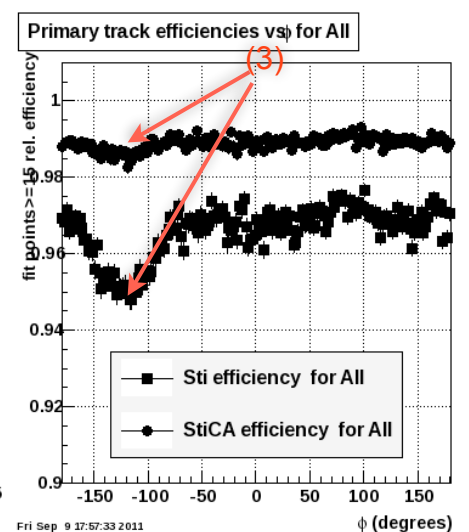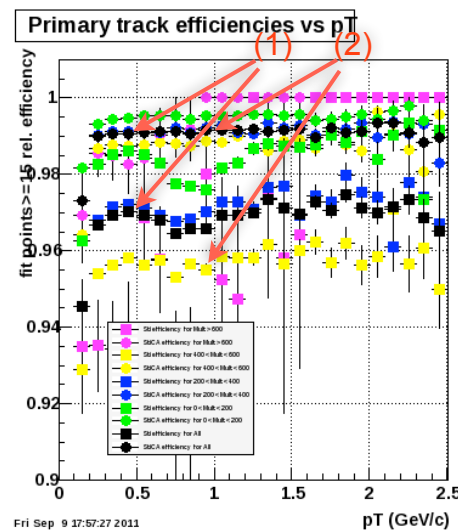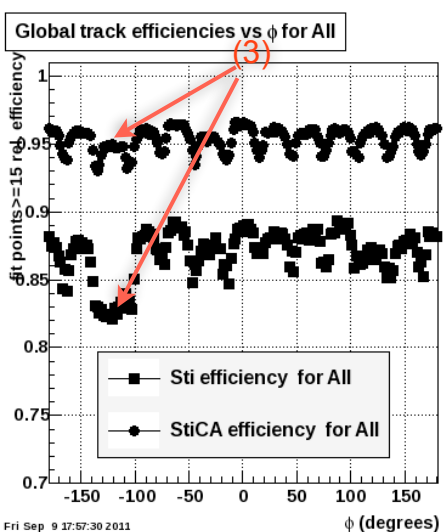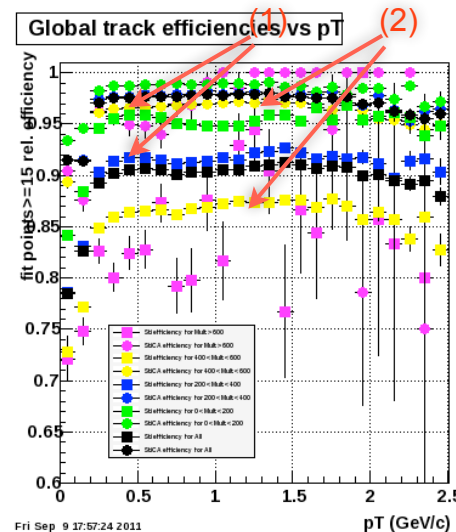
CA+Sti has (1) a higher efficiency, is (2) stable w.r.t. track multiplicity and (3) robust w.r.t. detector inefficiency.
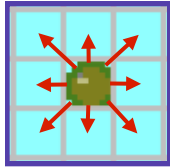
Each **cell** has 8 neighboring cells, 4 adjacent orthogonally, 4 adjacent diagonally. The **rules** are:
Survival: Every counter with 2 or 3 neighboring counters survives for the next generation.
Death: Each counter with 4 or more neighbors dies from overpopulation, with 1 neighbor or none dies from isolation.
Birth: Each empty cell adjacent to exactly 3 neighbors is a birth cell.
It is important to understand that all births and deaths occur *simultaneously*.

Martin Gardner, Mathematical games: The fantastic combinations of John Conway's new solitaire game „Life", Sci. Amer., 223 (1970) 120-123
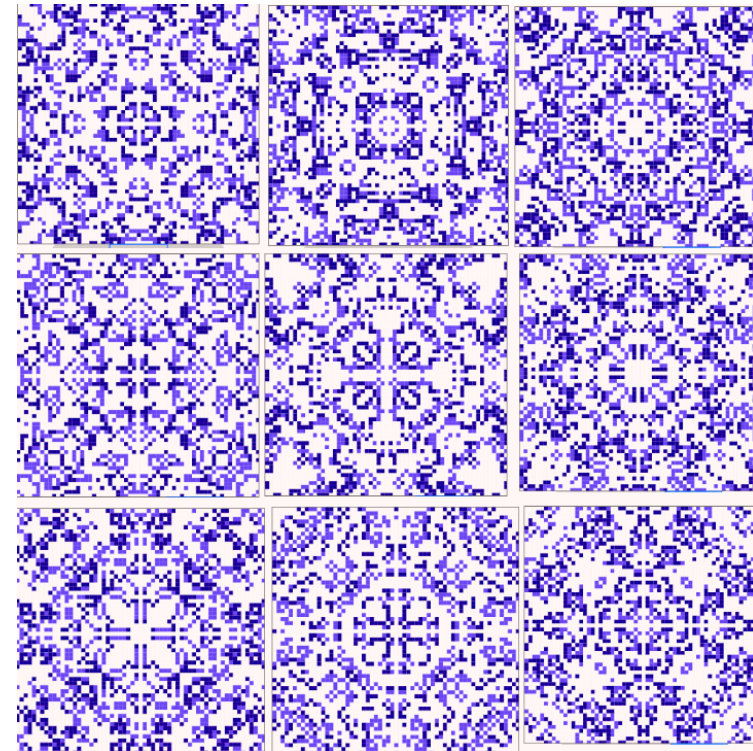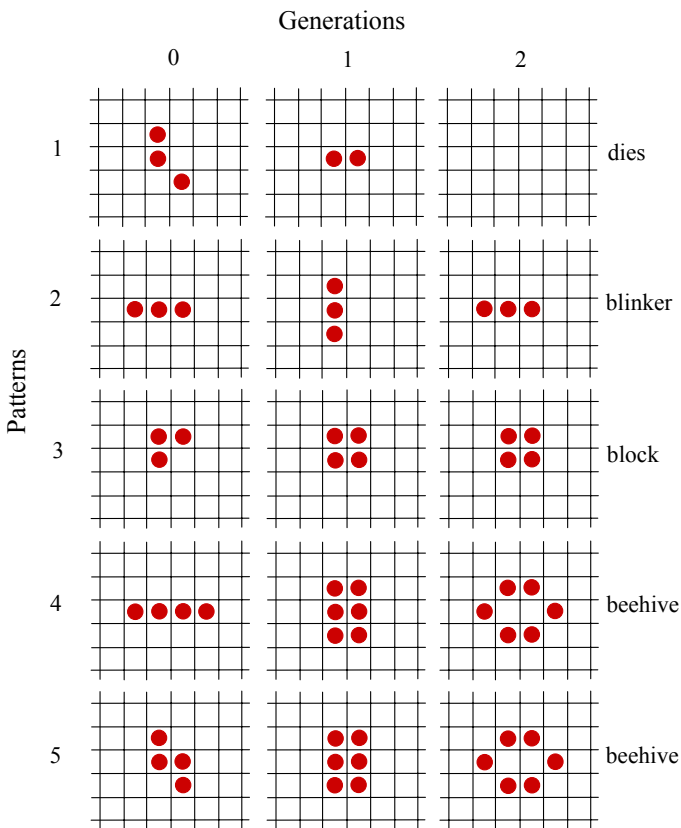
Cellular Automaton as Track Finder

1

0. Hits (CBM)

1000 Hits

0. Hits

Detector layers

Hits

$$Eff^{reco} = N_{tr}^{reco}/N_{tr}^{MC}$$

1. Segments

2. Counters

1   2   3   4

3. Track Candidates

4. Tracks

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.

4. Tracks (CBM)

1000 Tracks

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

Perfect for many-core CPU/GPU !

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

CBM CA Track Finder

(1) high-momentum primary tracks
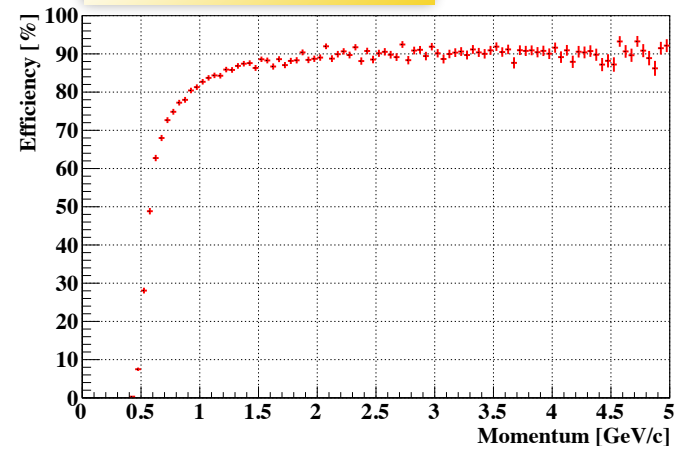
Track category	Eff, %
All tracks	70.4
Primary high-p	94.9
Primary low-p	56.8
Secondary high-p	49.7
Secondary low-p	13.0
Clone level	0.3
Ghost level	0.3
MC tracks found	103
Time, ms/ev	4

(3) secondary tracks
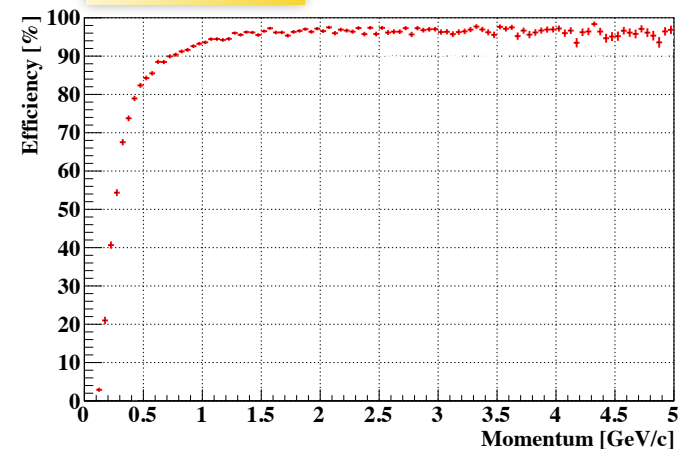
Track category	Eff, %
All tracks	89.2
Primary high-p	97.5
Primary low-p	92.4
Secondary high-p	86.6
Secondary low-p	54.7
Clone level	1.0
Ghost level	5.5
MC tracks found	131
Time, ms/ev	7

(2) low-momentum primary tracks
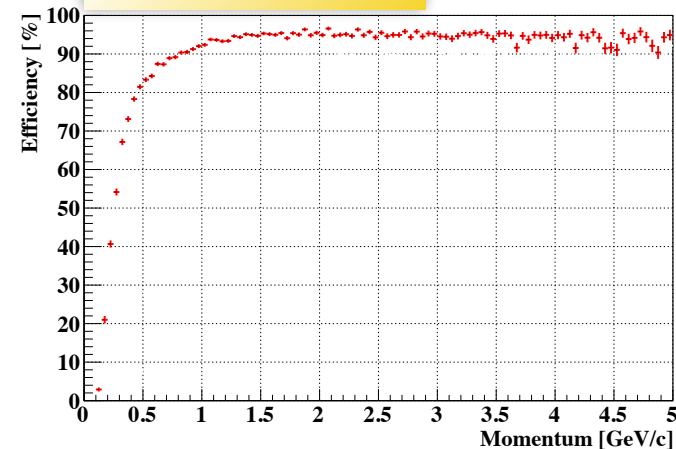
Track category	Eff, %
All tracks	87.8
Primary high-p	95.8
Primary low-p	91.4
Secondary high-p	84.5
Secondary low-p	54.2
Clone level	0.9
Ghost level	5.6
MC tracks found	129
Time, ms/ev	6

(4) broken tracks
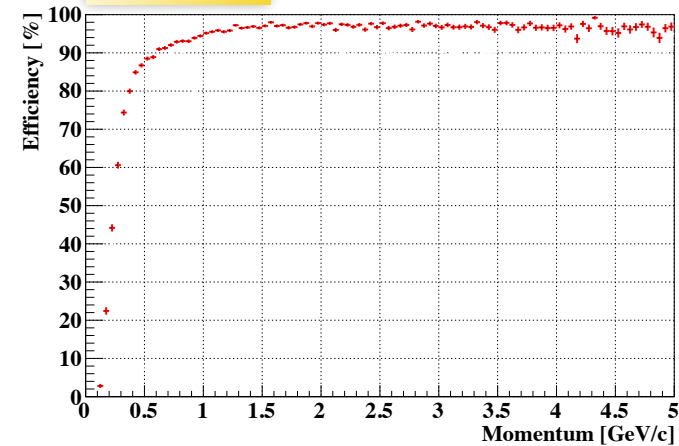
Track category	Eff, %
All tracks	90.9
Primary high-p	97.5
Primary low-p	92.6
Secondary high-p	91.1
Secondary low-p	63.8
Clone level	1.0
Ghost level	5.9
MC tracks found	134
Time, ms/ev	8

Efficient and stable event reconstruction

CBM CA Track Finder
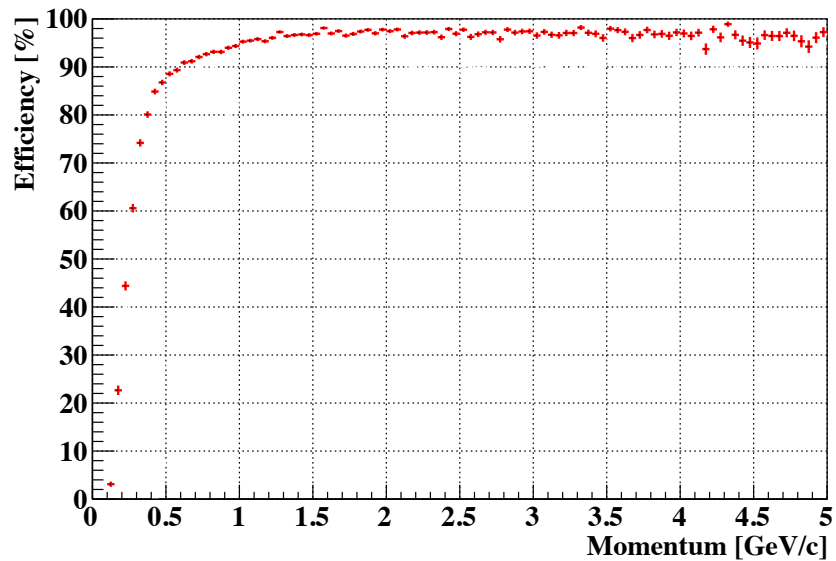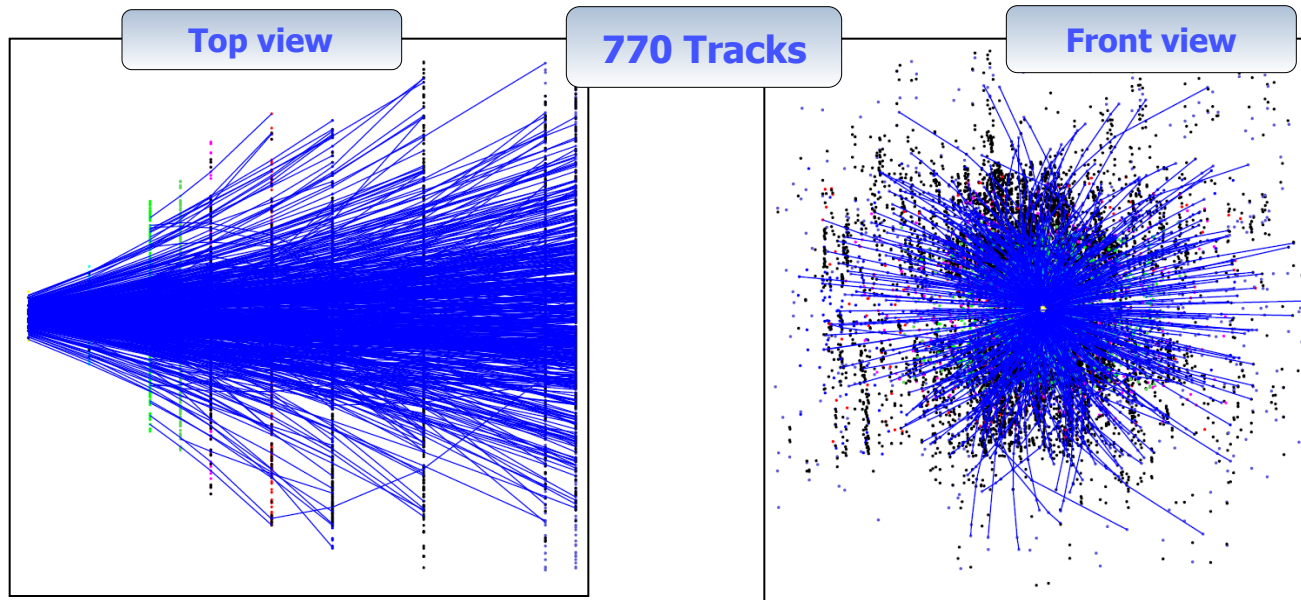
Top view

770 Tracks

Front view

Track category	Eff, %
All tracks	90.9
Primary high-p	97.5
Primary low-p	92.6
Secondary high-p	91.1
Secondary low-p	63.8
Clone level	0.4
Ghost level	5.9
MC tracks found	134
Time, ms/ev	10

Efficient and stable event reconstruction

CA Track Finder at High Track Multiplicity

A number of minimum bias events is gathered into a group (super-event), which is then treated by the CA track finder as a single event
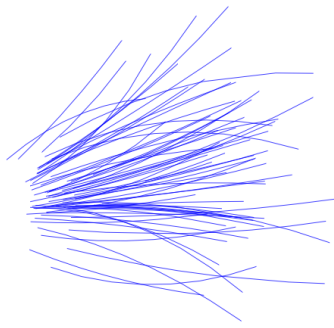
1 mbias event, $<N_{reco}> = 109$ 5 mbias events, $<N_{reco}> = 572$ 100 mbias events, $<N_{reco}> = 10340$

Number of Events

Efficiency, %

- ▲ **LongRefPrim**
- ▲ **RefPrim**
- ▲ **RefSec**
- ▲ **RefSet**
- ◆ **AllSet**
- ▼ **ExtraPrim**
- ▼ **ExtraSec**
- ▼ **ExtraSet**
- ● **Ghost**

MC Tracks

Au+Au mbias events at 25 AGeV

$$T = a \cdot N^2_{MC} + b \cdot N_{MC} + c$$

Time [s]

MC Tracks
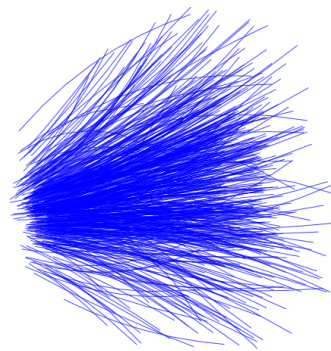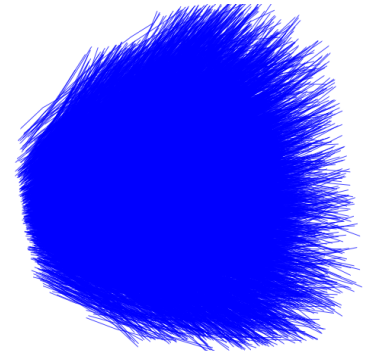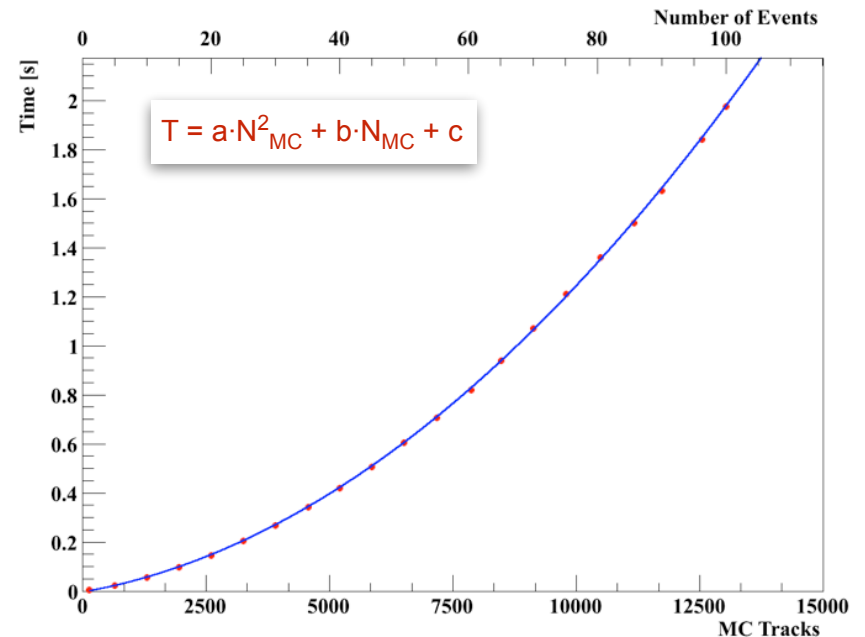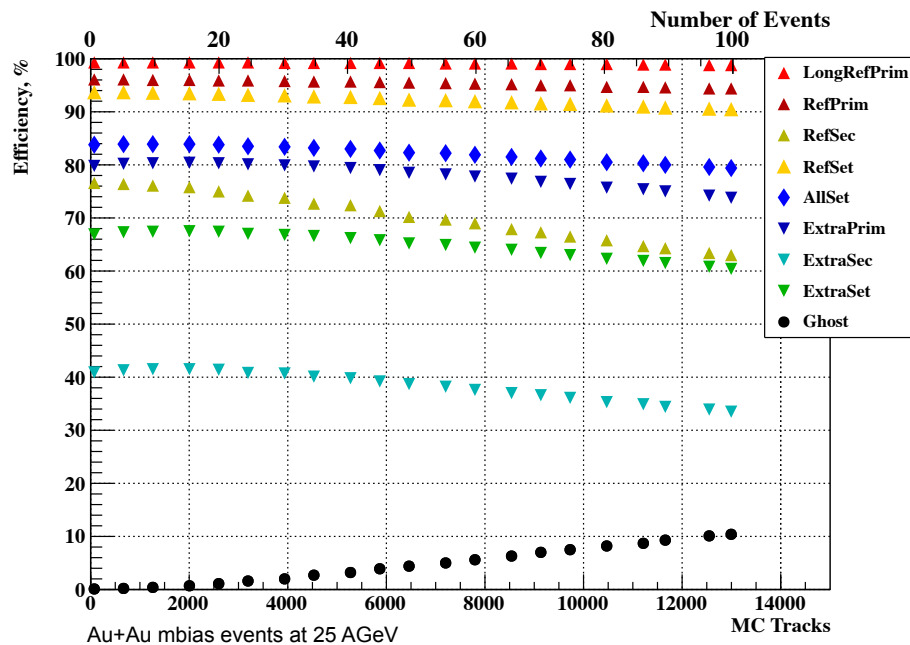
Stable reconstruction efficiency and time as a second order polynomial w.r.t. to track multiplicity

~~Event Builder~~ Collision Finder

- The beam in the CBM will have no bunch structure, but continuous.
- Measurements in this case will be 4D (x, y, z, t).
- Reconstruction of time slices rather than events will be needed.
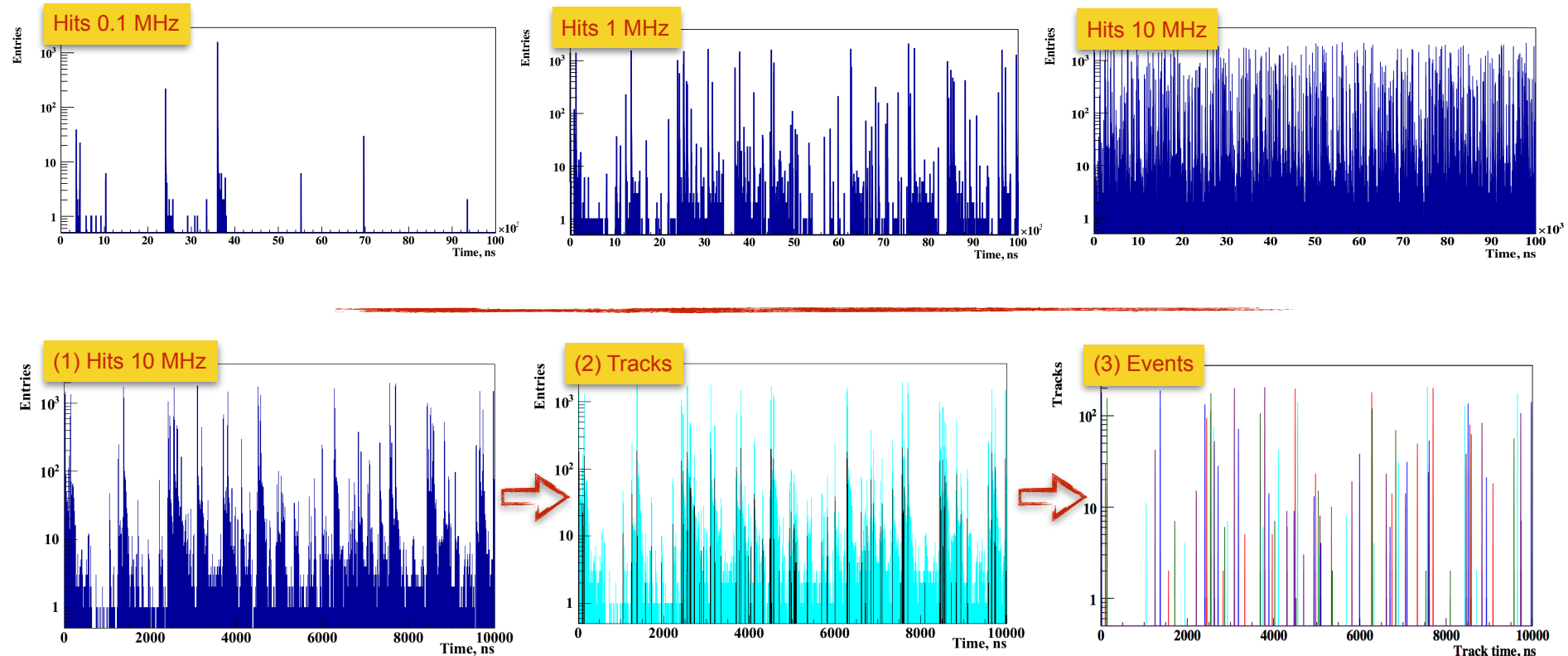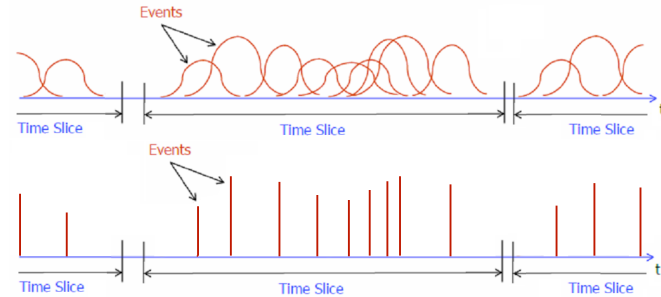
Reconstructed tracks clearly represent groups, which correspond to the primary collisions

Many-core HPC: Cores, Threads and Vectors

HEP experiments work with high data rates, therefore need High Performance Computing (HPC) !

Vectors (SIMD) = data level parallelism
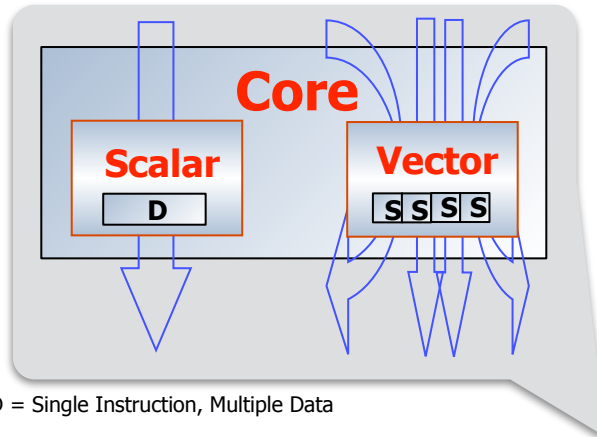
SIMD = Single Instruction, Multiple Data
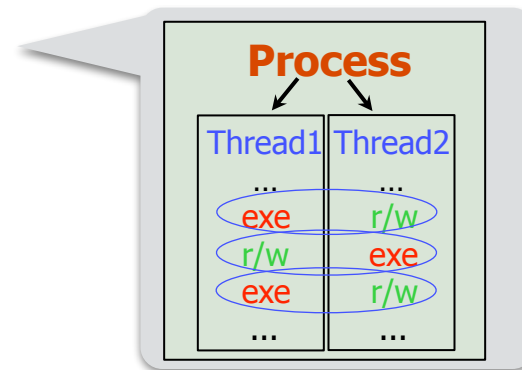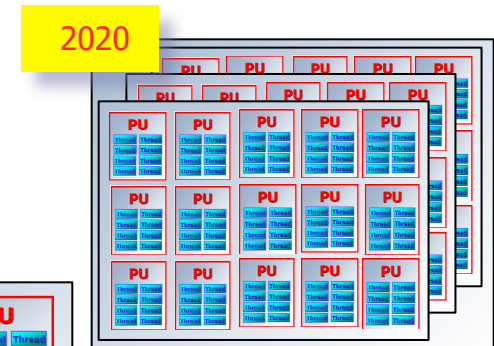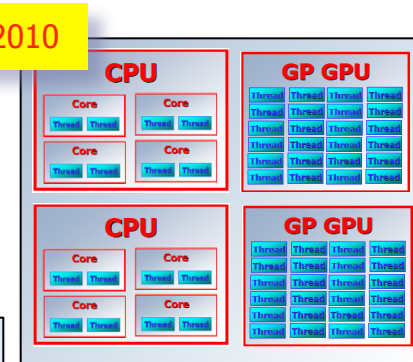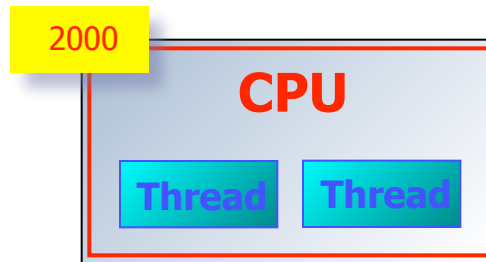
Threads

Cores

HPC   Vectors

Cores and Threads = task level parallelism

Fundamental redesign of traditional approaches to data processing is necessary

Many-Core CPU/GPU Architectures

Intel/AMD CPU

4xXX cores

Core Core Core Core

Processor Graphics

System Agent & Memory Controller

including Display, DMI and Misc. I/O

Shared L3 Cache

Memory Controller I/O

Math

Memory

Nvidia/ATI GPU

XXXX cores

L2 Cache
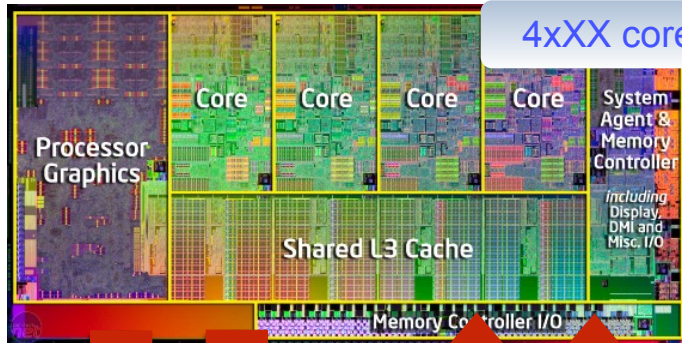
- Optimized for low latency access to cached data sets
- Control logic for out-of-order and speculative execution

- Optimized for data-parallel, throughput computation
- More transistors dedicated to computation

Parallelism

Math

Memory

#Cores

Stability

Memory

Intel Phi

60 cores

VECTOR IA CORE VECTOR IA CORE ... VECTOR IA CORE IA CORE

INTERPROCESSOR NETWORK

COHERENT CACHE COHERENT CACHE ... COHERENT CACHE COHERENT CACHE

COHERENT CACHE COHERENT CACHE ... COHERENT CACHE COHERENT CACHE

INTERPROCESSOR NETWORK

FIXED FUNCTION LOGIC

MEMORY and I/O INTERFACE

IBM Cell

1+8 cores

Interrupt Controller

SPE SPE SPE SPE

Local Store 256KB (x4)

Power PC Processing Element (PPE)

Memory Flow Controller (MFC) (x4)

Memory Controller

System Memory Rambus XDR

64-bit PPC 2-way SMT VMX

25GB/s (x5)

Element Interconnect Bus (EIB) 200GB/s

L1 Cache

25GB/s

Memory Flow Controller (MFC) (x4)

25GB/s 35GB/s

I/O Controller

I/O Device

- Many Integrated Cores architecture announced at ISC10 (June 2010)
- Based on the x86 architecture
- Many-cores + 4-way multithreaded + 512-bit wide vector unit

- General purpose RISC processor (PowerPC)
- 8 co-processors (SPE, Synergistic Processor Elements)
- 128-bit wide SIMD units

Future systems are heterogeneous. Fundamental redesign of traditional approaches to data processing is necessary

The Kalman filter is a recursive estimator – only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state.

mean value over n measurements

n

$$\mu_n = \frac{1}{n} \sum_{i=1}^{n} x_i$$

previous estimation   new measurement

n+1

$$\mu_{n+1} = \frac{1}{n+1} \sum_{i=1}^{n+1} x_i = \frac{n}{n+1} \left(\frac{1}{n} \sum_{i=1}^{n} x_i + \frac{1}{n} x_{n+1} \right) = \frac{n}{n+1} \mu_n + \frac{1}{n+1} x_{n+1} = \mu_n + \frac{1}{n+1} (x_{n+1} - \mu_n)$$

mean value over n+1 measurements

weight   correction

For this work, U.S. President Barack Obama rewarded Rudolf Kálmán with the National Medal of Science on October 7, 2009.

Radar applications

FAN BEAM

REFLECTOR
ANTENNA

R_1 R_2

Θ_{AZ}

AZIMUTH ANGLE

R
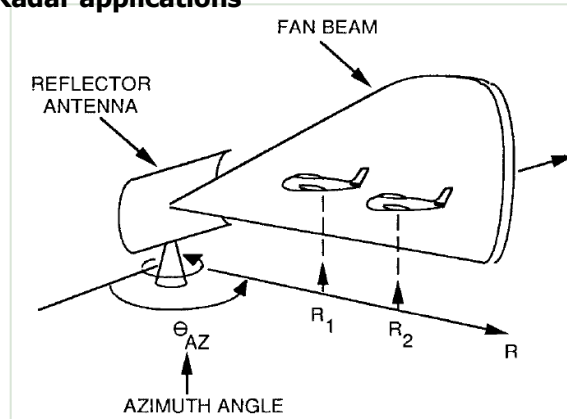
state vector:

r = { x, y, z, v_x, v_y, v_z }

covariance matrix:

$$C = \begin{Bmatrix} \sigma^2_x & & & & & \\ & \sigma^2_y & & & \cdots & \\ & & \sigma^2_z & & & \\ & & & \sigma^2_{vx} & & \\ & \cdots & & & \sigma^2_{vy} & \\ & & & & & \sigma^2_{vz} \end{Bmatrix}$$

Apollo Flight Journal

December 21, 1968. The Apollo 8 spacecraft has just been sent on its way to the Moon.
003:46:31 Collins: Roger. At your convenience, would you please go P00 and Accept? We're going to update to your W-matrix.

Kalman Filter based Track Fit

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

3   Correction

1   Initialization

Hits

Detector layers

π

(r, C)

Precision

2   Prediction

r – Track parameters
C – Covariance matrix

KF Block-diagram

1   Initial estimates for r_0 and C_0

$\tilde{r}_k$ $\tilde{C}_k$

2   Prediction step

3   Filtering step

r_k C_k

State estimate r_n
Error covariance C_n
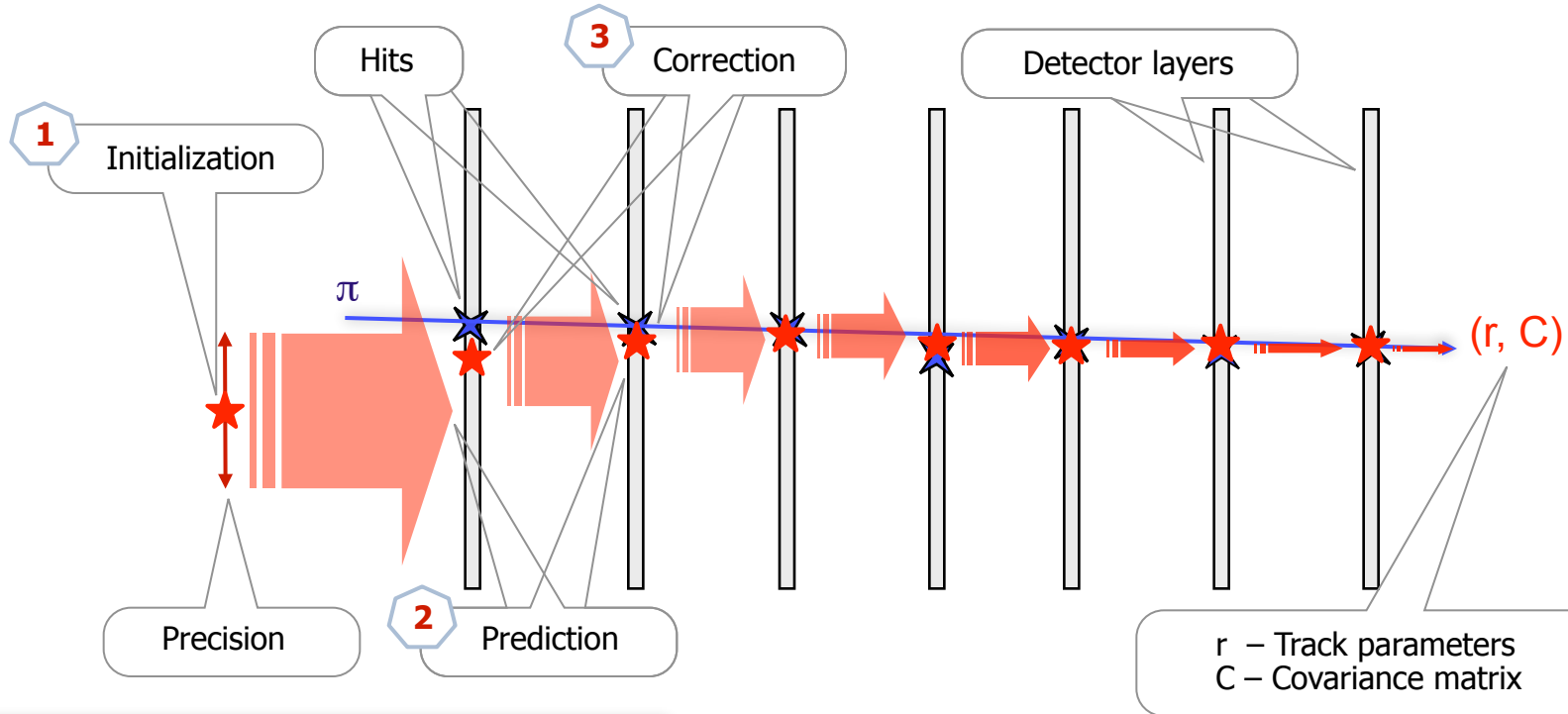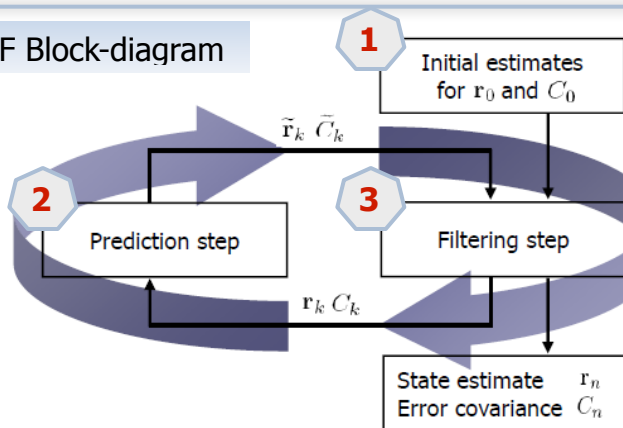
KF as a recursive least squares method

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

Kalman Filter Track Fit Quality

CBM track parameters: $r = \{ x, y, t_x, t_y, q/p \}$ position, tg of slopes, charge over momentum

residuals

$$\rho_x = x_{reco} - x_{mc}$$

pulls

$$P(x) = \frac{\rho_x}{\sqrt{C_{xx}}}$$
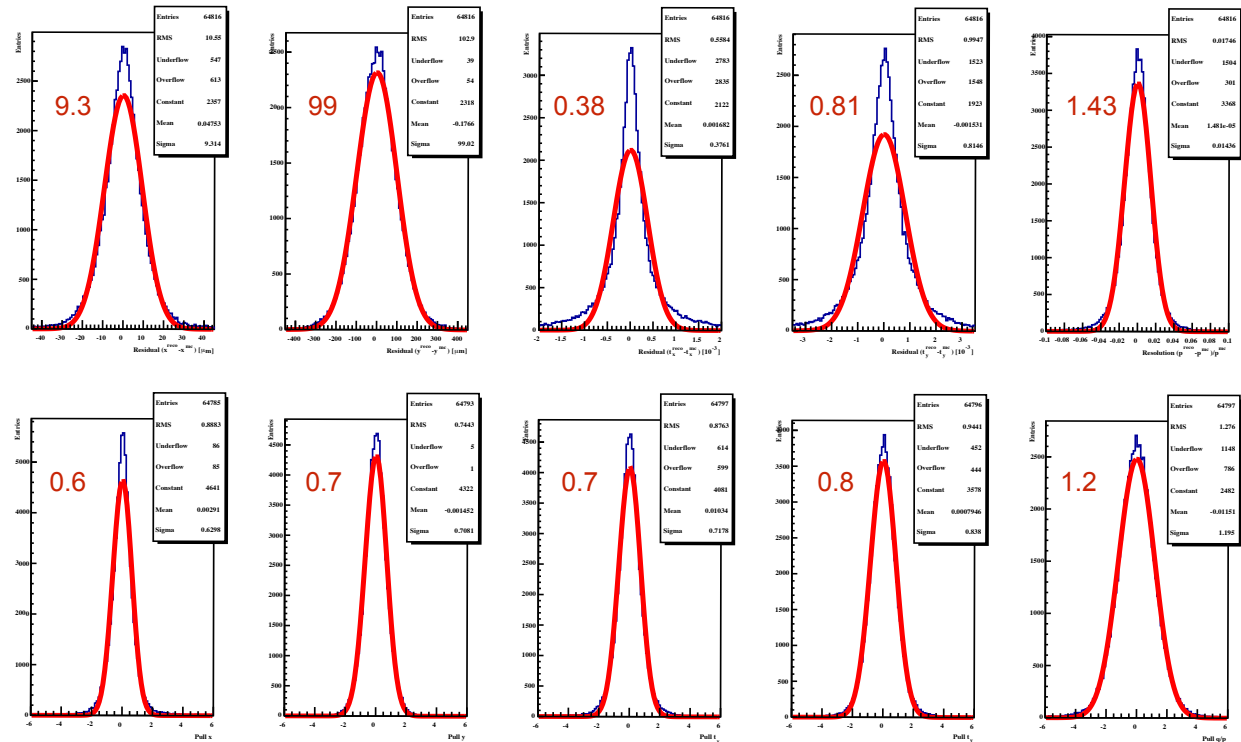
resolution					pull widths				
x, µm	y, µm	t_x, 10^{-3}	t_y, 10^{-3}	p, %	x	y	t_x	t_y	q/p
9.3	99	0.38	0.81	1.43	0.6	0.7	0.7	0.8	1.2

Input hits parameters: ρ x = 8.4, ρ y = 91,   Pull x = 0.63, Pull y = 0.69

Illustrative slide, 2012!

Kalman Filter Track Fit on Cell

Stage	Description	Time/track	Speedup
	Initial scalar version	12 ms	–
1	Approximation of the magnetic field	240 μs	50
2	Optimization of the algorithm	7.2 μs	35
3	Vectorization	1.6 μs	4.5
4	Porting to SPE	1.1 μs	1.5
5	Parallelization on 16 SPEs	0.1 μs	10
	Final simdized version	0.1 μs	120000

Intel — Stages 1, 2, 3

Cell — Stages 4, 5

10000x faster on any PC

Comp. Phys. Comm. 178 (2008) 374-383

The KF speed was increased by 5 orders of magnitude
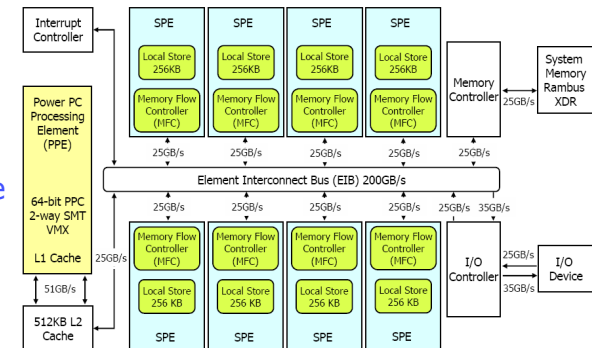
blade11bc4 @IBM, Böblingen:
2 Cell Broadband Engines, 256 kB LS, 2.4 GHz

Motivated by, but not restricted to Cell !

Kalman Filter (KF) Track Fit Library

Kalman Filter Methods

Kalman Filter Tools:
- KF Track Fitter
- KF Track Smoother
- Deterministic Annealing Filter

Kalman Filter Approaches:
- Conventional DP KF
- Conventional SP KF
- Square-Root SP KF
- UD-Filter SP
- Gaussian Sum Filter
- 3D (x,y,z) and 4D (x,y,z,t) KF

Track Propagation:
- Runge-Kutta
- Analytic Formula

Detector Types:
- Pixel
- Strip
- Straw-Tube
- Time Projection Camber

Implementations

Vectorization (SIMD):
- Header Files
- Vc Vector Classes
- ArBB Array Building Blocks
- OpenCL

Parallelization (many-cores):
- Open MP
- ITBB
- ArBB
- OpenCL

Precision:
- single precision SP
- double precision DP

Conventional KF DP vs. SP
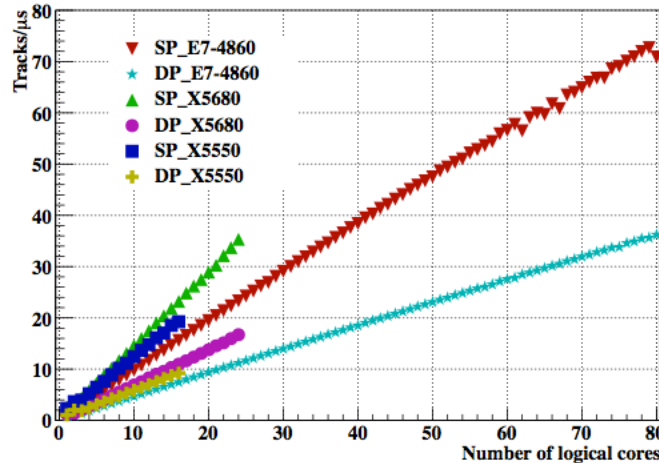
Conventional KF RK4 vs. Analytical
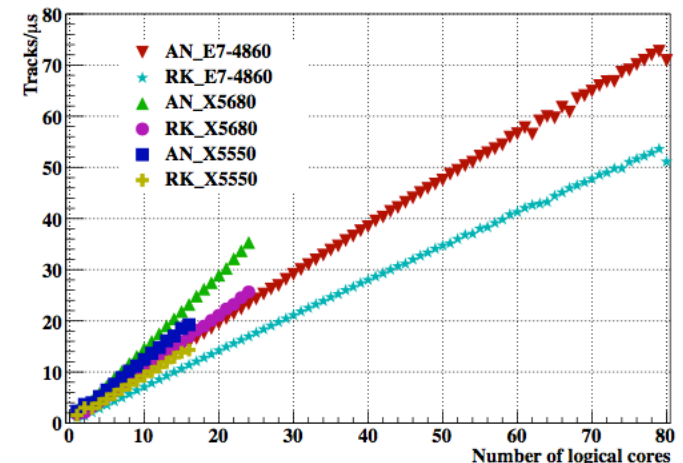
Square-Root KF
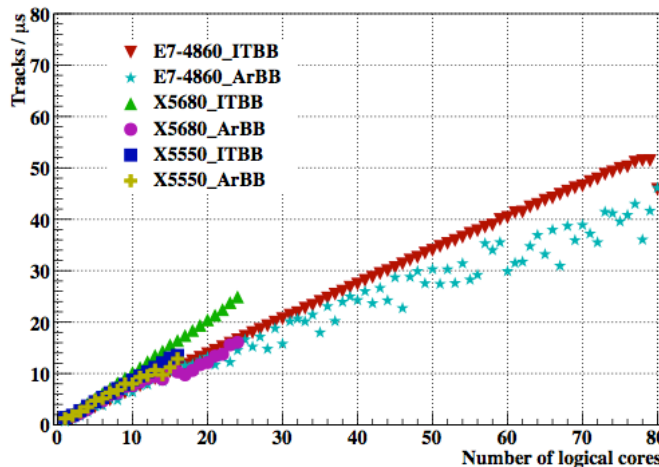
UD KF
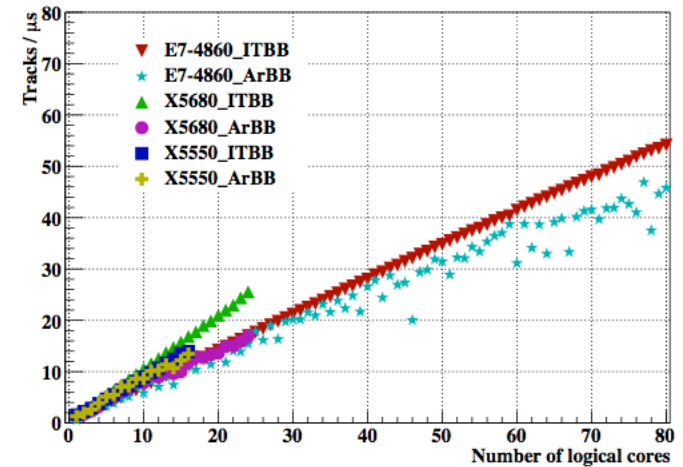
Strong many-core scalability of the Kalman filter library

with I. Kulakov, H. Pabst* and M. Zyzak (*Intel)

Full Portability of the KF Track Fit

85 tracks/μs

Intel Xeon E7-4860, 2.26 GHz

Tracks/μs vs *Number of Logical Cores*

192 tracks/μs

Intel Xeon Phi 7120, 1.2 GHz

Tracks/μs vs *Number of Logical Cores*

115 tracks/μs

Nvidia GTX 480, 700 MHz

Tracks/μs vs *Local Item Size*

372 tracks/μs

AMD Radeon HD 7970, 925 MHz

Tracks/μs vs *Local Item Size*
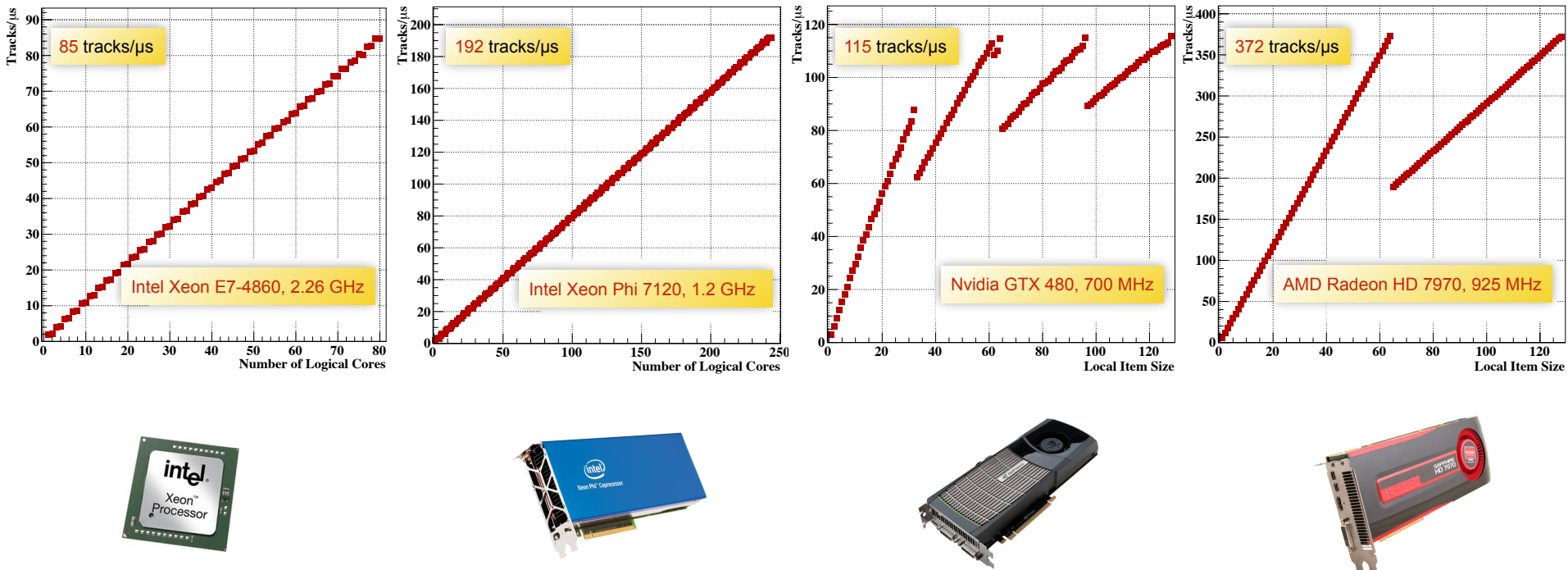
- **Scalability** with respect to the **number of logical cores** in a CPU is one of the most important parameters of the algorithm.
- The scalability on the **Intel Xeon Phi** coprocessor is **similar** to the **CPU**, but running **four threads per core instead of two**.
- In case of the **graphics cards** the set of tasks is divided into **working groups** of size *local item size* and **distributed among compute units** (or streaming multiprocessors) and the **load of each compute unit** is of the particular **importance**.

Single node KF Track Fit performance: 2*CPU+2*GPU = 10^9 tracks/s = (100 tracks/event)* 10^7 events/s = 10^7 events/s

Fast, precise and portable Kalman filter library

$$f = 4 \text{ sockets} \times 8 \text{ cores} \times 1.3 \text{ threads} \times 8 \text{ SIMD} \approx 300,$$

which is already equivalent to a moderate computer farm with scalar single-core CPUs.

Location		Architecture	(Nodes·)sockets·cores·threads·SIMD	Data streams
CERN	Switzerland	AMD 6164HE	4·12·1·4	192
GSI	Germany	Intel E7-4860	4·10·2·4	320
ITEP	Russia	AMD 6272	100·(2·16·1·4)	12 800
FIAS	Germany	Intel E5-2600+Intel Phi 7120	2·8·2·8+2·61·4·16 **8 064**	256+7 808 **177 408**
BNL	USA	Intel E5-2680+Intel Phi 5110P	22·(2·12·2·8+2·60·4·16)	8 448+168 960

Tab. 5: List of some heterogeneous HPC nodes, used in our investigations.

In order to investigate the HPC hardware and to develop efficient algorithms we use different nodes and clusters in several high-energy physics centers over the worlds (see Tab. 5) ranging from dozens to thousand of cores.

11

Parallelization becomes a standard in the CBM experiment

Simulated AuAu collision at 25 AGeV

$$\overline{\Omega}^+ \rightarrow \overline{\Lambda}\, K^+$$
$$\qquad\quad \hookrightarrow \overline{p}\, \pi^+$$

Concept:

- Mother and daughter particles have the same state vector and are treated in the same way
- Reconstruction of decay chains
- Kalman filter based
- Geometry independent
- Vectorized
- Uncomplicated usage

Functionality:

- Construction of short-lived particles
- Addition and subtraction of particles
- Transport
- Calculation of an angle between particles
- Calculation of distances and deviations
- Constraints on mass, production point and decay length
- KF Particle Finder

Reconstruction of decays with a neutral daughter
by the missing mass method:


```
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
```

KF Particle provides a simple and direct approach to physics analysis (used in CBM, ALICE and STAR)

KF Particle Finder block-diagram

Charged particles: $e^\pm$, $\mu^\pm$, $\pi^\pm$, $K^\pm$, $p^\pm$, $d^\pm$, $^3He^\pm$, $^4He^\pm$

Neutral particles: ν_μ, $\overline{\nu}_\mu$, π^0, n, $\overline{n}$, Λ, $\overline{\Lambda}$, Ξ^0, $\overline{\Xi}^0$

Dileptons

Charmonium
$J/\psi \rightarrow e^+ e^-$
$J/\psi \rightarrow \mu^+ \mu^-$

Low mass
vector mesons
$\rho \rightarrow e^+ e^-$
$\rho \rightarrow \mu^+ \mu^-$
$\omega \rightarrow e^+ e^-$
$\omega \rightarrow \mu^+ \mu^-$
$\phi \rightarrow e^+ e^-$
$\phi \rightarrow \mu^+ \mu^-$

Gamma
$\gamma \rightarrow e^+ e^-$
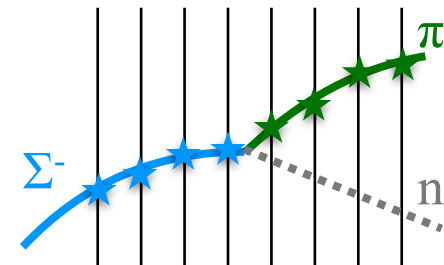
Gamma-decays
$\pi^0 \rightarrow \gamma \gamma$
$\eta \rightarrow \gamma \gamma$

Light mesons
and baryons
$\pi^+ \rightarrow \mu^+ \nu_\mu$
$\pi^- \rightarrow \mu^- \overline{\nu}_\mu$
$\rho \rightarrow \pi^+ \pi^-$
$\Delta^0 \rightarrow p \pi^-$
$\overline{\Delta}^0 \rightarrow \overline{p} \pi^+$
$\Delta^{++} \rightarrow p \pi^+$
$\overline{\Delta}^{--} \rightarrow \overline{p} \pi^-$

Open-charm

Open-charm
particles
$D^0 \rightarrow K^- \pi^+$
$D^0 \rightarrow K^- \pi^+ \pi^+ \pi^-$
$\overline{D}^0 \rightarrow K^+ \pi^-$
$\overline{D}^0 \rightarrow K^+ \pi^+ \pi^- \pi^-$
$D^+ \rightarrow K^- \pi^+ \pi^+$
$D^- \rightarrow K^+ \pi^- \pi^-$
$D_s^+ \rightarrow K^+ K^- \pi^+$
$D_s^- \rightarrow K^+ K^- \pi^-$
$\Lambda_c^+ \rightarrow p K^- \pi^+$
$\overline{\Lambda}_c^- \rightarrow \overline{p} K^- \pi^+$

Open-charm
resonances
$D^{*0} \rightarrow D^+ \pi^-$
$\overline{D}^{*0} \rightarrow D^- \pi^+$
$D^{*+} \rightarrow D^0 \pi^+$
$D^{*-} \rightarrow \overline{D}^0 \pi^-$

Strange particles

$K^0_s \rightarrow \pi^+ \pi^-$
$K^+ \rightarrow \mu^+ \nu_\mu$
$K^- \rightarrow \mu^- \overline{\nu}_\mu$
$K^+ \rightarrow \pi^+ \pi^0$
$K^- \rightarrow \pi^- \pi^0$
$\Lambda \rightarrow p \pi^-$
$\overline{\Lambda} \rightarrow \overline{p} \pi^+$
$\Sigma^+ \rightarrow p \pi^0$
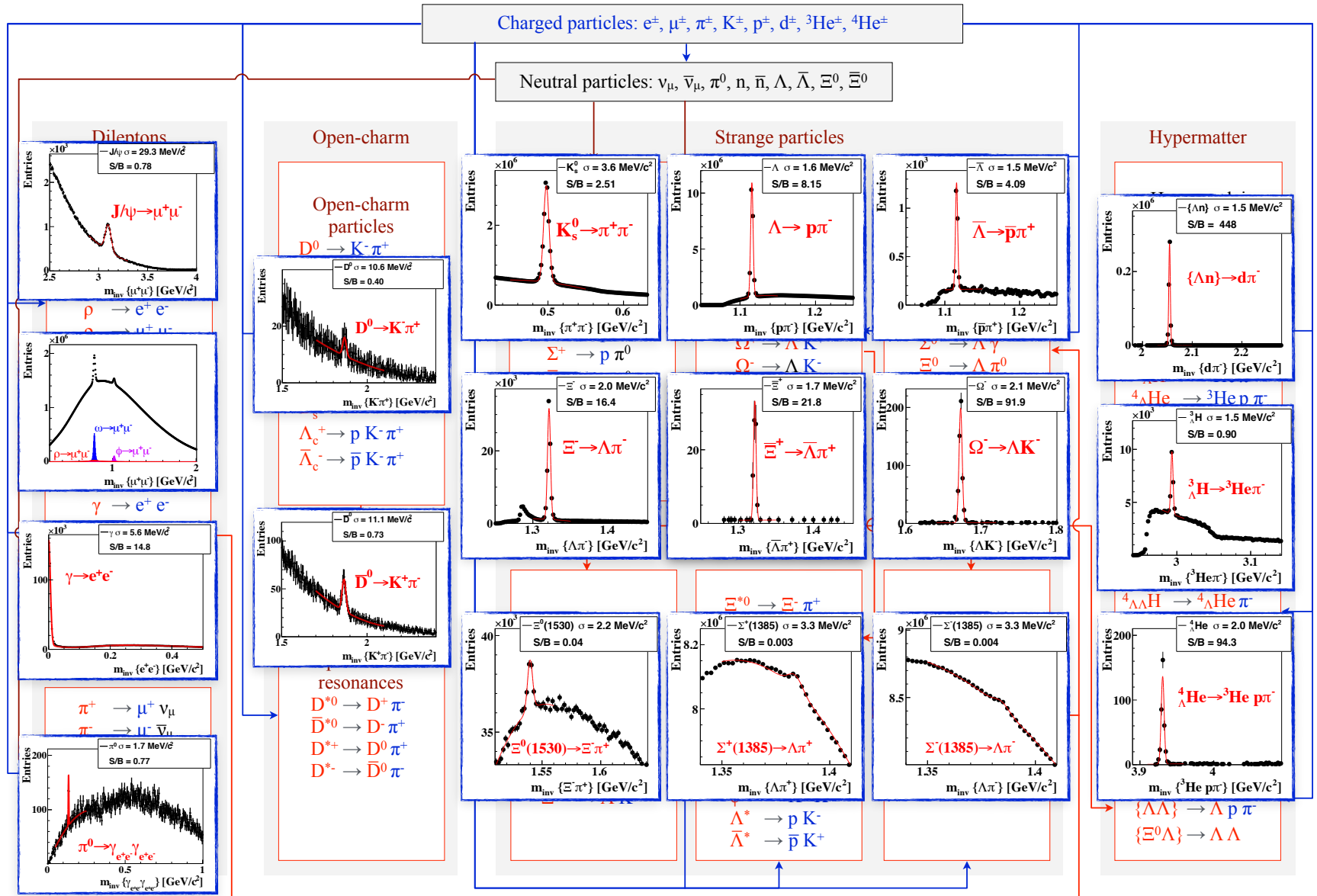$\overline{\Sigma}^- \rightarrow \overline{p} \pi^0$
$\Sigma^+ \rightarrow n \pi^+$
$\Sigma^- \rightarrow \overline{n} \pi^-$
$\Sigma^- \rightarrow n \pi^-$
$\overline{\Sigma}^+ \rightarrow \overline{n} \pi^+$

$\Xi^- \rightarrow \Lambda \pi^-$
$\overline{\Xi}^+ \rightarrow \overline{\Lambda} \pi^+$
$\Xi^- \rightarrow \Lambda \pi^-$
$\overline{\Xi}^+ \rightarrow \overline{\Lambda} \pi^+$
$\Omega^- \rightarrow \Lambda K^-$
$\overline{\Omega}^+ \rightarrow \overline{\Lambda} K^+$
$\Omega^- \rightarrow \Lambda K^-$
$\overline{\Omega}^+ \rightarrow \overline{\Lambda} K^+$
$\Omega^- \rightarrow \Xi^0 \pi^-$
$\overline{\Omega}^+ \rightarrow \overline{\Xi}^0 \pi^+$

$\Sigma^+ \rightarrow p \pi^0$
$\overline{\Sigma}^- \rightarrow \overline{p} \pi^0$
$\Sigma^0 \rightarrow \Lambda \gamma$
$\overline{\Sigma}^0 \rightarrow \overline{\Lambda} \gamma$
$\Xi^0 \rightarrow \Lambda \pi^0$
$\overline{\Xi}^0 \rightarrow \overline{\Lambda} \pi^0$

Strange resonances

$K^{*+} \rightarrow K^0_s \pi^+$
$K^{*-} \rightarrow K^0_s \pi^-$
$\Sigma^{*+} \rightarrow \Lambda \pi^+$
$\overline{\Sigma}^{*-} \rightarrow \overline{\Lambda} \pi^-$
$\Sigma^{*-} \rightarrow \Lambda \pi^-$
$\overline{\Sigma}^{*+} \rightarrow \overline{\Lambda} \pi^+$
$\Xi^{*-} \rightarrow \Lambda K^-$
$\overline{\Xi}^{*+} \rightarrow \overline{\Lambda} K^+$

$\Xi^{*0} \rightarrow \Xi^- \pi^+$
$\overline{\Xi}^{*0} \rightarrow \overline{\Xi}^+ \pi^-$
$\Omega^{*-} \rightarrow \Xi^- K^- \pi^+$
$\overline{\Omega}^{*+} \rightarrow \overline{\Xi}^+ K^+ \pi^-$

$K^{*0} \rightarrow K^+ \pi^-$
$\overline{K}^{*0} \rightarrow K^- \pi^+$
$\phi \rightarrow K^+ K^-$
$\Lambda^* \rightarrow p K^-$
$\overline{\Lambda}^* \rightarrow \overline{p} K^+$

$K^{*+} \rightarrow K^+ \pi^0$
$K^{*-} \rightarrow K^- \pi^0$
$K^{*0} \rightarrow K^0 \pi^0$
$\Sigma^{*0} \rightarrow \Lambda \pi^0$
$\overline{\Sigma}^{*0} \rightarrow \overline{\Lambda} \pi^0$
$\Xi^{*-} \rightarrow \Xi^- \pi^0$
$\overline{\Xi}^{*+} \rightarrow \overline{\Xi}^+ \pi^0$

Hypermatter

Hypernuclei
$\{\Lambda n\} \rightarrow d^+ \pi^-$
$\{\overline{\Lambda}\overline{n}\} \rightarrow d^- \pi^+$
$\{\Lambda nn\} \rightarrow t^+ \pi^-$
$\{\overline{\Lambda}\overline{nn}\} \rightarrow t^- \pi^+$
$^3_\Lambda H \rightarrow {}^3He \pi^-$
$^3_\Lambda \overline{H} \rightarrow {}^3\overline{He} \pi^+$
$^4_\Lambda H \rightarrow {}^4He \pi^-$
$^4_\Lambda \overline{H} \rightarrow {}^4\overline{He} \pi^+$
$^4_\Lambda He \rightarrow {}^3He p \pi^-$
$^4_\Lambda \overline{He} \rightarrow {}^3\overline{He} \overline{p} \pi^+$
$^5_\Lambda He \rightarrow {}^4He p \pi^-$
$^5_\Lambda \overline{He} \rightarrow {}^4\overline{He} \overline{p} \pi^+$

Double-Λ
hypernuclei
$^4_{\Lambda\Lambda} H \rightarrow {}^4_\Lambda He \pi^-$
$^4_{\Lambda\Lambda} H \rightarrow {}^3_\Lambda H p \pi^-$
$^5_{\Lambda\Lambda} H \rightarrow {}^5_\Lambda He \pi^-$
$^4_{\Lambda\Lambda} He \rightarrow {}^5_\Lambda He p \pi^+$

Heavy multi-
strange objects
$\{\Lambda\Lambda\} \rightarrow \Lambda p \pi^-$
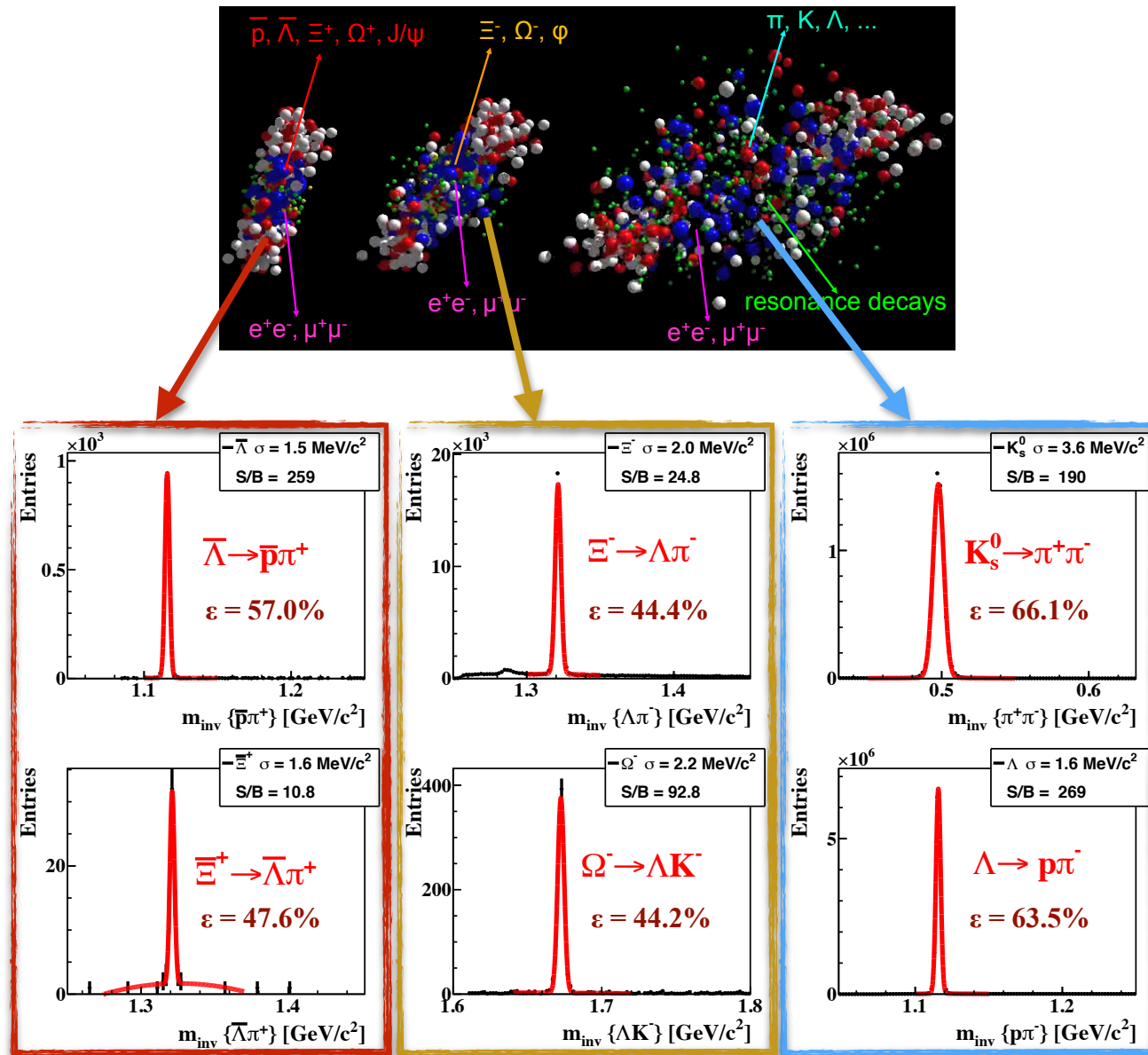$\{\Xi^0\Lambda\} \rightarrow \Lambda \Lambda$

(mbias: 1.4 ms; central: 10.5 ms)/event/core

KF Particle Finder for Physics Analysis and Selection

Clean Probes of Collision Stages

$\overline{p}, \overline{\Lambda}, \Xi^+, \Omega^+, J/\psi$ Ξ^-, Ω^-, φ $\pi, K, \Lambda, ...$

$e^+e^-, \mu^+\mu^-$ $e^+e^-, \mu^+\mu^-$ resonance decays $e^+e^-, \mu^+\mu^-$

$\overline{\Lambda}$ σ = 1.5 MeV/c² , S/B = 259

$$\overline{\Lambda} \rightarrow \overline{p}\pi^+$$

ε = 57.0%

$m_{inv}\{\overline{p}\pi^+\}$ [GeV/c²]

$\overline{\Xi}^+$ σ = 1.6 MeV/c² , S/B = 10.8

$$\overline{\Xi}^+ \rightarrow \overline{\Lambda}\pi^+$$

ε = 47.6%

$m_{inv}\{\overline{\Lambda}\pi^+\}$ [GeV/c²]

Ξ^- σ = 2.0 MeV/c² , S/B = 24.8

$$\Xi^- \rightarrow \Lambda\pi^-$$

ε = 44.4%

$m_{inv}\{\Lambda\pi\}$ [GeV/c²]

Ω^- σ = 2.2 MeV/c² , S/B = 92.8

$$\Omega^- \rightarrow \Lambda K^-$$

ε = 44.2%

$m_{inv}\{\Lambda K^-\}$ [GeV/c²]

K_s^0 σ = 3.6 MeV/c² , S/B = 190

$$K_s^0 \rightarrow \pi^+\pi^-$$

ε = 66.1%

$m_{inv}\{\pi^+\pi^-\}$ [GeV/c²]

Λ σ = 1.6 MeV/c² , S/B = 269

$$\Lambda \rightarrow p\pi^-$$

ε = 63.5%

$m_{inv}\{p\pi^-\}$ [GeV/c²]

AuAu, 10 AGeV, 3.5M central UrQMD events, MC PID

Online physics analysis = online extraction of medium properties in heavy-ion collisions

Direct approach

Theoretical model (T, μ,V, σ,...) $\longrightarrow$ $\longleftarrow$ $\omega_p \dfrac{dN}{dp^3}$ (observable)

Inverse approach
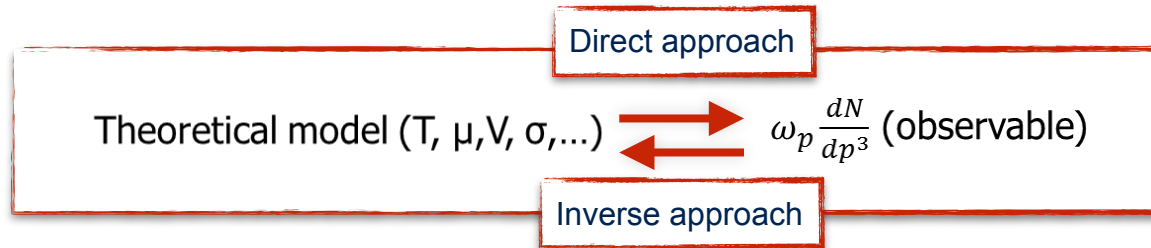
Motivation:
Direct way:
- determination of physical properties of QCD matter created in HIC (temperature, flow, phase transitions, …),
- obtain limits of applicability of different models
Inverse way:

Stages of collision   **Models for different stages**
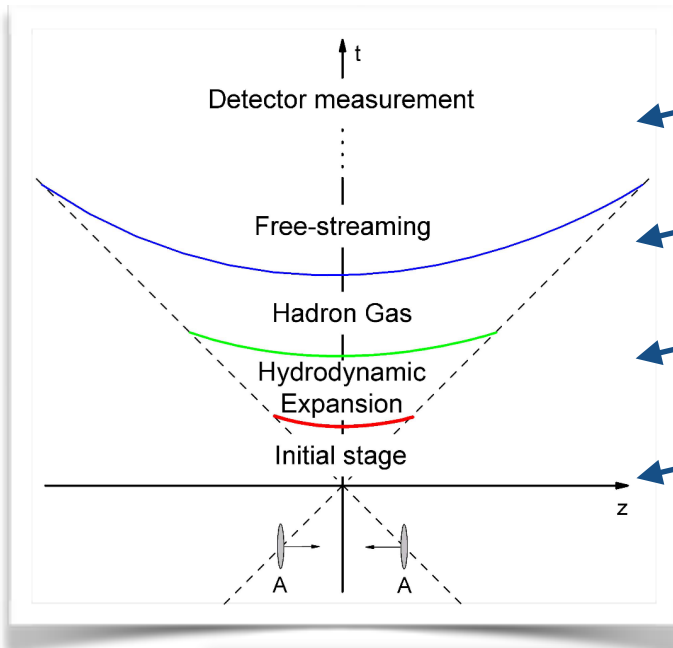
Final momentum spectrum (Blast-Wave, Tsallis, …)

Statistical-thermal models for chemical freeze-out
(ideal hadron gas, Van der Waals hadron gas, Hagedorn states, …)

Relativistic hydrodynamics (ideal, viscous; (0+1)D, (1+1)D, (3+1)D, …)

Initial stage (Glauber, CGC, …)

A package to extract the parameters of theoretical models in CBM experiment is implemented

CBM Online Physics Analysis

Direct approach

Theoretical model (T, μ,V, σ,...) ⇄ $\omega_p \frac{dN}{dp^3}$ (observable)

Inverse approach
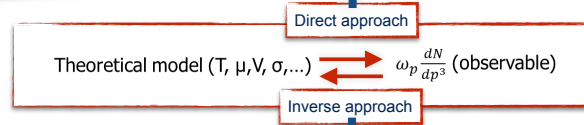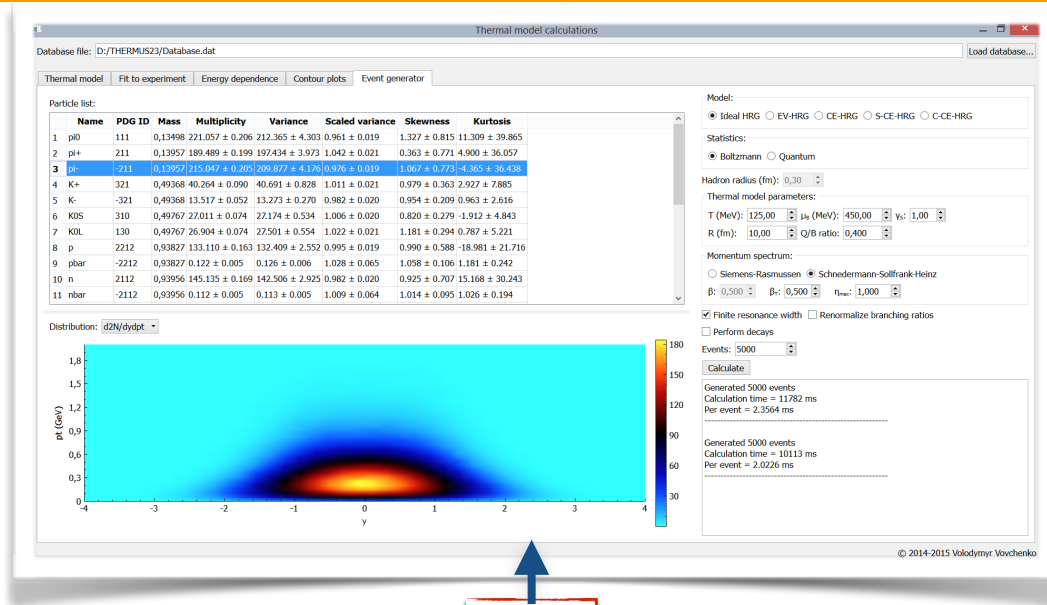
E.-by-E. extraction of T and μ_B (HRG)
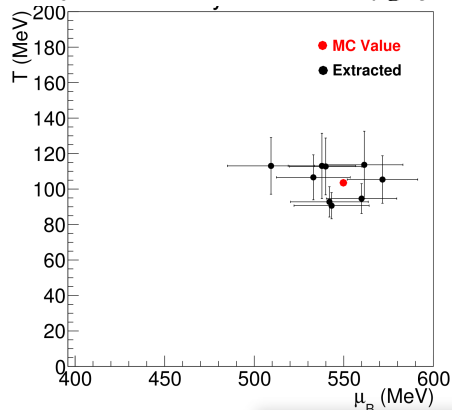
E.-by-E. yield estimate incl. acceptance (Blast-Wave)
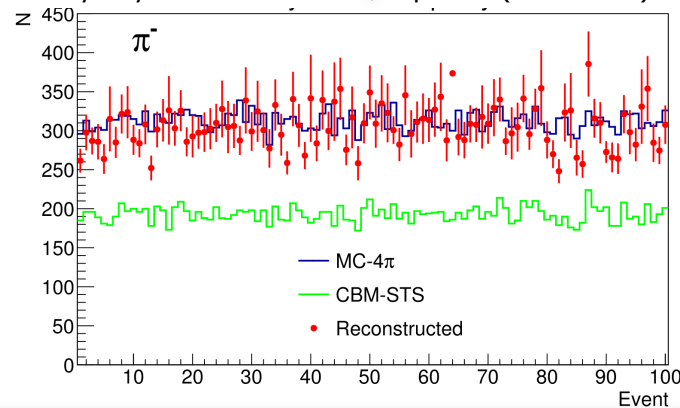
E.-by-E. impact parameter (Glauber)
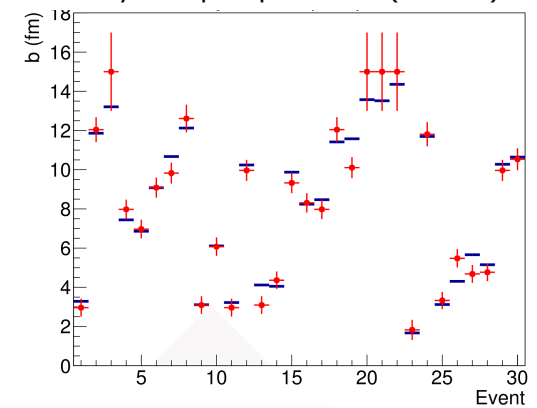
A package to extract the parameters of theoretical models in CBM experiment is implemented

CBM Standalone First Level Event Selection (FLES) Package
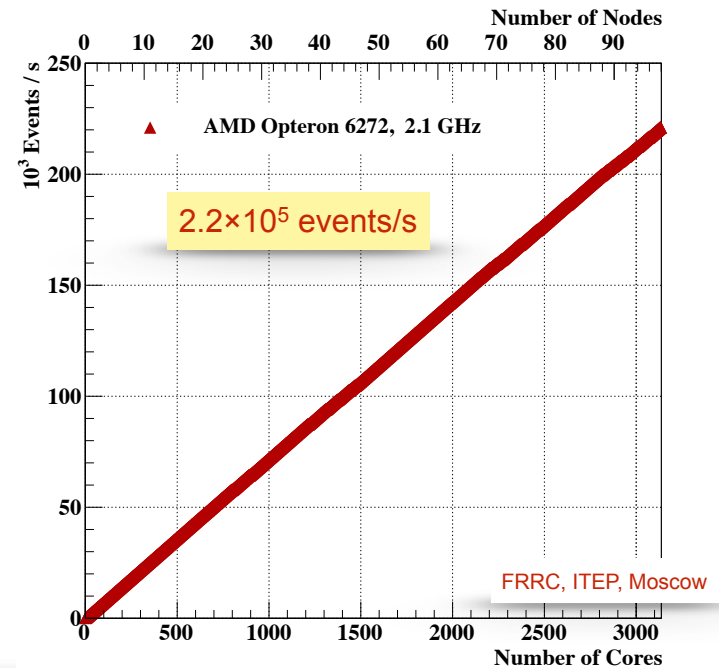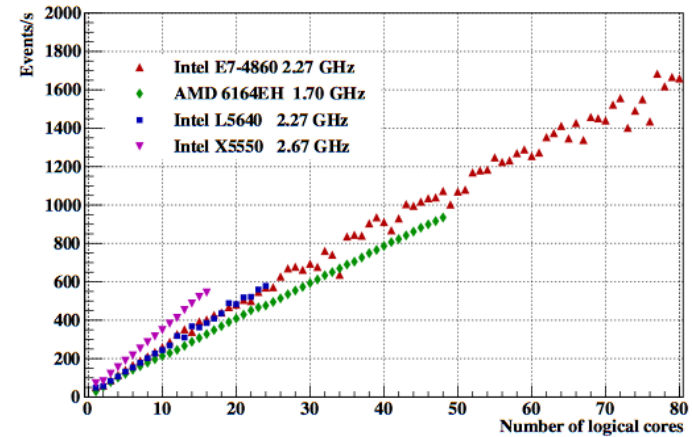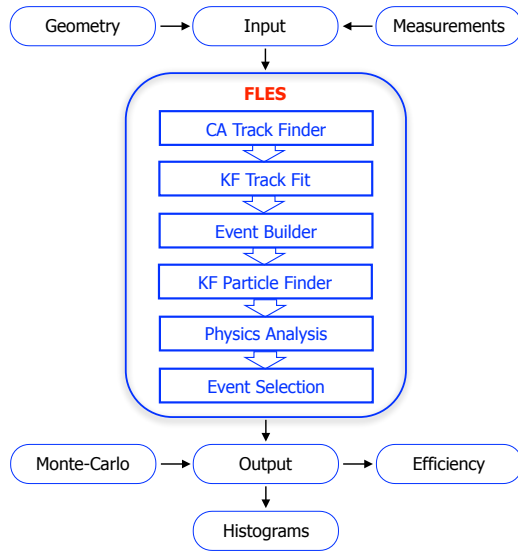
The FLES package is vectorized, parallelized, portable and scalable up to 3 200 CPU cores

Summary: to be HPC Efficient - Consolidate Efforts
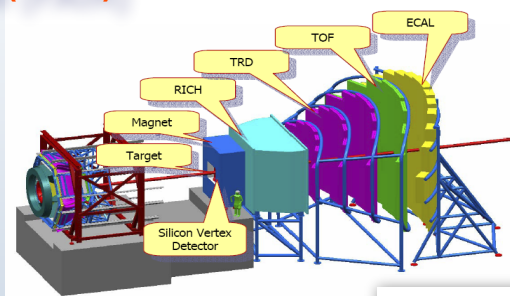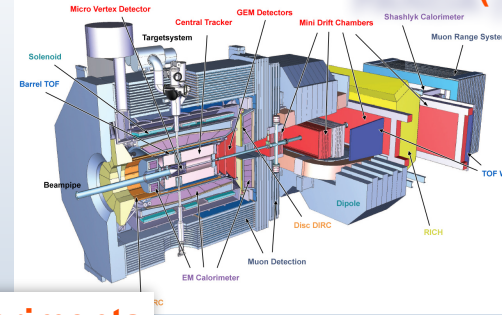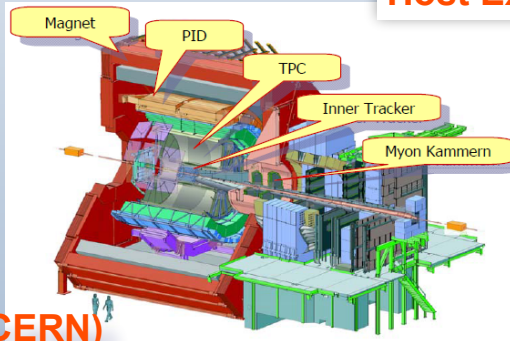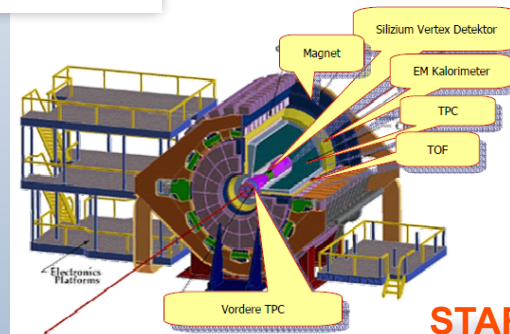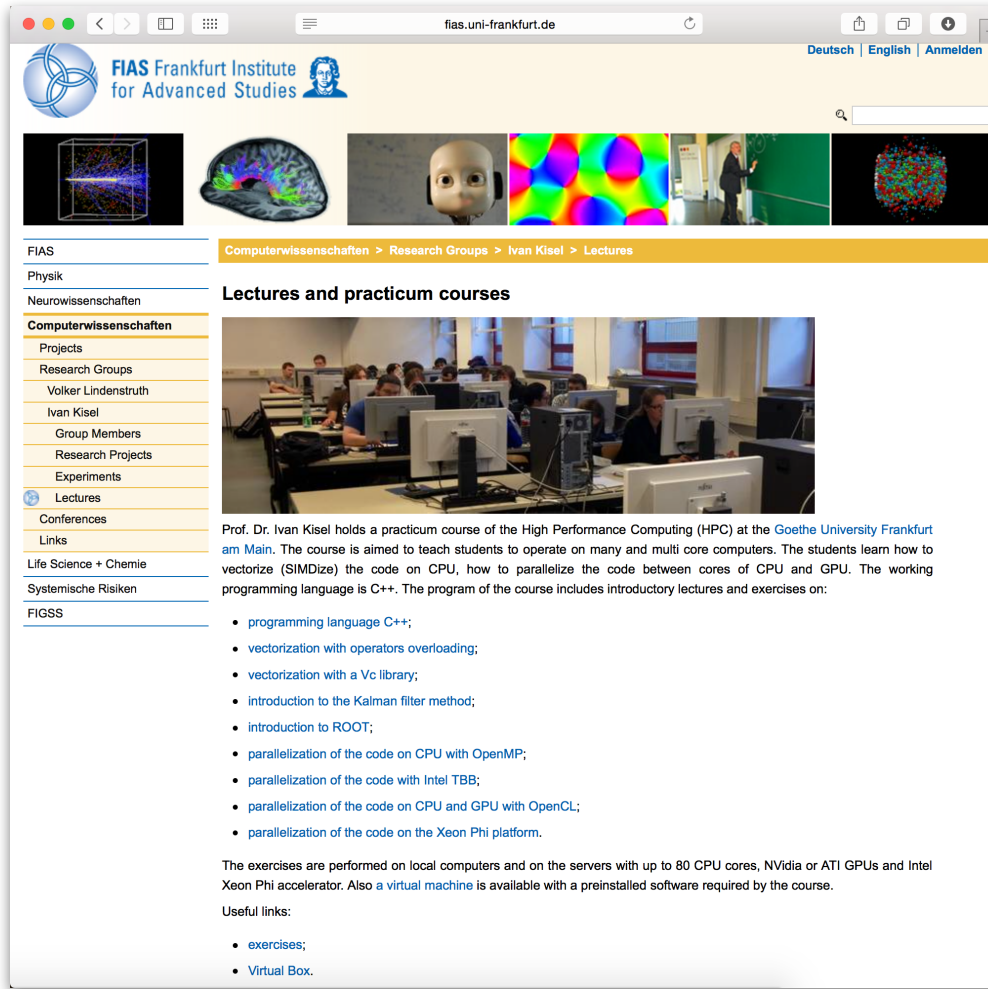
Consolidate efforts of:

- Physicists
- Mathematicians
- Computer scientists
- Developers of // languages
- Many-core CPU/GPU manufacturers

HPC Practical Course at the Goethe University Frankfurt

http://fias.uni-frankfurt.de/de/cs/kisel/lectures/

Conclusion

The future is parallel.
The future is parallel.
The future is parallel.
The future is parallel.