Scientific and High-Performance Computing at FAIR

Ivan Kisel
Goethe University Frankfurt am Main
FIAS Frankfurt Institute for Advanced Studies
- $10^7$ 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
Reconstruction Challenge in CBM

First level event selection will be done in a processor farm with ≈ 60 000 CPU equivalent cores fed with data from the event (time slices) building network.
Stages of Event Reconstruction

1. Track Finder
   - Conformal Mapping
   - Hough Transformation
   - Track Following + Kalman Filter
   - Cellular Automaton + Kalman Filter

2. Track Fitter
   - Kalman Filter

3. Ring Finder (Particle ID)
   - Hough Transformation
   - Elastic Neural Net

4. Short-Lived Particle Finder
   - Kalman Filter
Global Methods: Conformal Mapping + Histogramming

Global methods are especially suitable for fast tracking in projections

**Example:** Collider experiment with a solenoid, where tracks are circular trajectories

**Conformal Mapping:**
Transform circles into straight lines

\[
\begin{align*}
u &= \frac{x}{x^2+y^2} \\
v &= -\frac{y}{x^2+y^2}
\end{align*}
\]

**Histogram:**
Collect a histogram of azimuth angles \( \phi \)
Find peaks in the histogram
Collect hits into tracks
Global Methods: Hough Transformation

Measurement Space

\[ y = a \times x + b \]

Parameter Space

\[ b = -x \times a + y \]
Local Methods: Kalman Filter for Track Following

Kalman Filter

1. Initialization
2. Prediction
3. Correction

Detector layers

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

Position, direction and momentum

State vector

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.

KF Block-diagram

KF as a recursive least squares method

\( KF \) Track Fitter

\( KF \) Track Finder

Seed Planes

Detector layers

Hits

Correction

(r, C)

\( r \) – Track parameters

\( C \) – Covariance matrix

Initialization

Prediction

Precision

KF Track Fitter

KF Track Finder
Local Methods: Cellular Automaton as Track Finder

0. Hits

1. Segments

2. Counters

3. Track Candidates

4. Tracks

Detecter layers

Hits

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.
Many-Core CPU/GPU Architectures

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

Intel/AMD CPU

- Future systems are heterogeneous

ATI/NVIDIA GPU

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

Intel Xeon Phi

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

IBM Cell

- General purpose RISC processor (PowerPC)
- 8 co-processors (SPE, Synergistic Processor Elements)
- 128-bit wide SIMD units
Many-core HPC: Cores, Threads and Vectors

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

Cores and Threads realize the task level of parallelism

Vectors (SIMD) = data level of parallelism

Fundamental redesign of traditional approaches to data processing is necessary
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**

1. **Initialization**
   - Initial estimates for $r_0$ and $C_0$

2. **Prediction**
   - Prediction step
   - $r_k$, $C_k$

3. **Correction**
   - Filtering step
   - State estimate $r_n$, Error covariance $C_n$

**KF as a recursive least squares method**

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.

**State vector**

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

**Position, direction and momentum**

$$r, C$$

**Detector layers**

**KF Block-diagram**

- **Initialization**
- **Prediction**
- **Correction**
- **Detector layers**

**KF as a recursive least squares method**

1. **Initialization**
   - Initial estimates for $r_0$ and $C_0$

2. **Prediction**
   - Prediction step
   - $r_k$, $C_k$

3. **Correction**
   - Filtering step
   - State estimate $r_n$, Error covariance $C_n$

**KF as a recursive least squares method**

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.
CBM: Kalman Filter 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

Track Propagation:
- Runge-Kutta
- Analytic Formula

Implementations

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

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

Precision:
- single
- double

Strong many-core scalability of the Kalman filter library
CBM: Cellular Automaton Track Finder

Efficient and reliable event reconstruction
CBM: CA Track Finder at High Track Multiplicity

A number of minimum bias events is gathered into a group, which is then treated by the track finder as one event.

1 minimum bias event
\[ <N_{\text{reco}}^\text{1mbias} > = 109 \]

5 minimum bias events
\[ <N_{\text{reco}}^\text{5mbias} > = 572 \]

100 minimum bias events
\[ <N_{\text{reco}}^\text{100mbias} > = 10340 \]
CBM: CA Track Finder Efficiency and Time vs. Track Multiplicity

Stable reconstruction efficiency and time as a second order polynomial up to 100 minimum bias events in a group
CBM: 4D Event Building with CA Track 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 original events.
CBM: Particle Finder for Physics Analysis and Selection

Tracks: $e^\pm$, $\mu^\pm$, $\pi^\pm$, $K^\pm$, $p^\pm$

Open-charm:
- $D^0 \rightarrow \pi^+ K^-$
- $D^0 \rightarrow \pi^+ \pi^+ \pi^- K^-$
- $\bar{D}^0 \rightarrow \pi^- K^+$
- $D^0 \rightarrow \pi^- \pi^- \pi^+ K^+$
- $D^+ \rightarrow \pi^+ \pi^+ K^-$
- $D^- \rightarrow \pi^- \pi^- K^+$
- $D^+_s \rightarrow \pi^+ K^+ K^-$
- $D^-_s \rightarrow \pi^- K^+ K^-$
- $\Lambda_c \rightarrow \pi^+ K^- p$

Strange particles:
- $K^0_s \rightarrow \pi^+ \pi^-$
- $\Lambda \rightarrow p \pi^-$
- $\bar{\Lambda} \rightarrow \pi^+ p^-$

Strange resonances:
- $K^{*0} \rightarrow K^+ \pi^-$
- $\bar{K}^{*0} \rightarrow \pi^- K^+$
- $\Lambda^* \rightarrow p K^-$
- $\bar{\Lambda}^* \rightarrow p^+ K^+$

Light vector mesons:
- $\rho \rightarrow e^- e^+$
- $\rho \rightarrow \mu^- \mu^+$
- $\omega \rightarrow e^- e^+$
- $\omega \rightarrow \mu^- \mu^+$
- $\phi \rightarrow e^- e^+$
- $\phi \rightarrow \mu^- \mu^+$

Charmonium:
- $J/\Psi \rightarrow e^- e^+$
- $J/\Psi \rightarrow \mu^- \mu^+$

Multi-strange:
- $\Xi^- \rightarrow \Lambda \pi^-$
- $\Xi^+ \rightarrow \bar{\Lambda} \pi^+$
- $\Omega^- \rightarrow \Lambda K^-$
- $\Omega^+ \rightarrow \bar{\Lambda} K^+$

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

Multi-strange resonances:
- $\Xi^{*0} \rightarrow \Xi^- \pi^+$
- $\Xi^{*0} \rightarrow \Xi^+ \pi^-$
- $\Omega^{*-} \rightarrow \Xi^- \pi^+ K^-$
- $\Omega^{*+} \rightarrow \Xi^+ K^+$

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

Ivan Kisel, Uni-Frankfurt, FIAS
3rd International Conference on New Frontiers in Physics, Crete, 29.07.2014
CBM Standalone First Level Event Selection (FLES) Package

The first version of the FLES package is vectorized, parallelized, portable and scalable.
Towards CBM FLES Demonstrator

- LOEWE CSC (FIAS, Frankfurt)
- Green Cube (GSI, Darmstadt)
- FAIR-Russia HPC Cluster (ITEP, Moscow)

From cores to the CBM FLES farm with 60,000 cores

Graph showing the relationship between number of cores and number of events per second with the following specifications:
- 10^3 Events/s
- 2.2 \times 10^5 events/s
- AMD Opteron 6272, 2.1 GHz

Number of Cores
0 500 1000 1500 2000 2500 3000

Number of Nodes
0 10 20 30 40 50 60 70 80 90

19
PANDA (FAIR): CA Track Finder in STT and MVD

10 primary tracks with pt = 1 GeV/c

<table>
<thead>
<tr>
<th></th>
<th>STT</th>
<th>STT+MVD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Efficiency</td>
<td>97.2</td>
<td>99.3</td>
</tr>
<tr>
<td>Clone</td>
<td>1.8</td>
<td>9.2</td>
</tr>
<tr>
<td>Ghost</td>
<td>2.5</td>
<td>2.5</td>
</tr>
<tr>
<td>Tracks/event</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Time, ms/event</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>

Combine experience of CBM and PANDA
Consolidate Efforts: Common Reconstruction Package

- Create a common FAIR reconstruction package
- Use also in running experiments (ALICE, STAR, …)
- Optimize for many-core CPU/GPU/Phi architectures
- Hide parallelism and keep the traditional form of physics analysis
Summary

With increasing complexity of online data analysis
HPC computing becomes a detector sub-system in HEP experiments