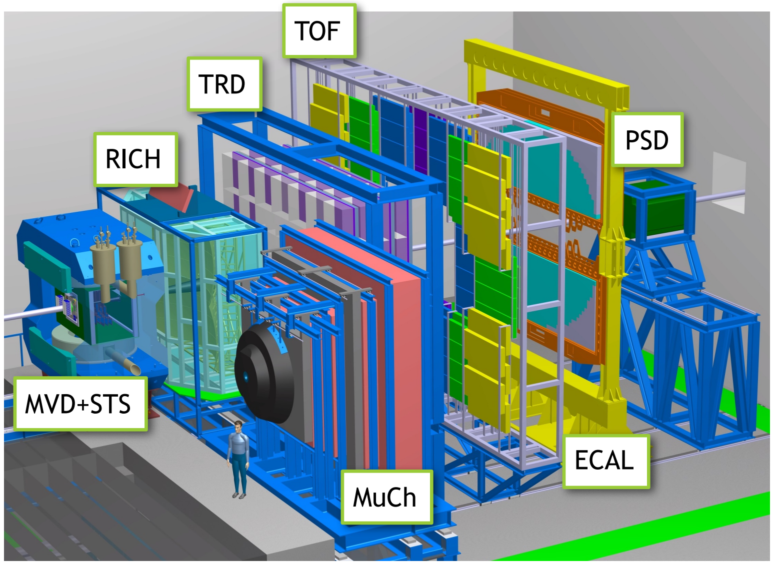
Application and improvement of the Cellular Automaton track finder in the TPC detector

G.Kozlov[1,2,3], I.Kisel[1,2,4]

[1]Goethe University, Frankfurt, Germany
[2]FIAS, Frankfurt, Germany
[3]JINR LIT, Dubna, Russia
[4]GSI, Darmstadt, Germany

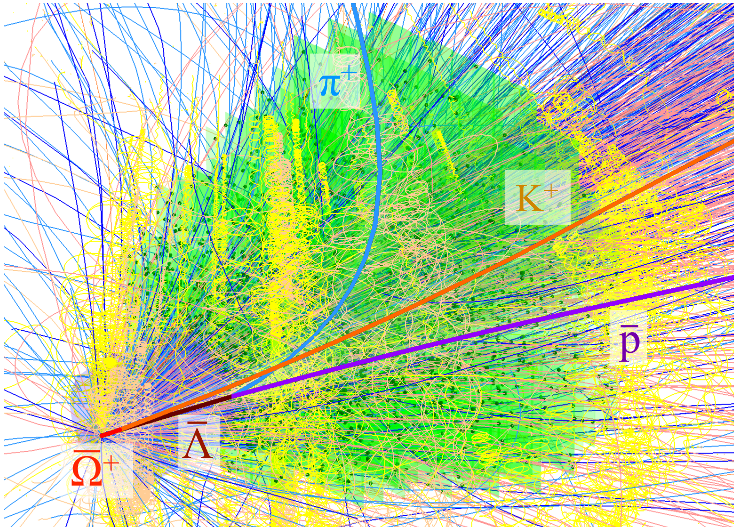
20.03.2019

Challenges in CBM

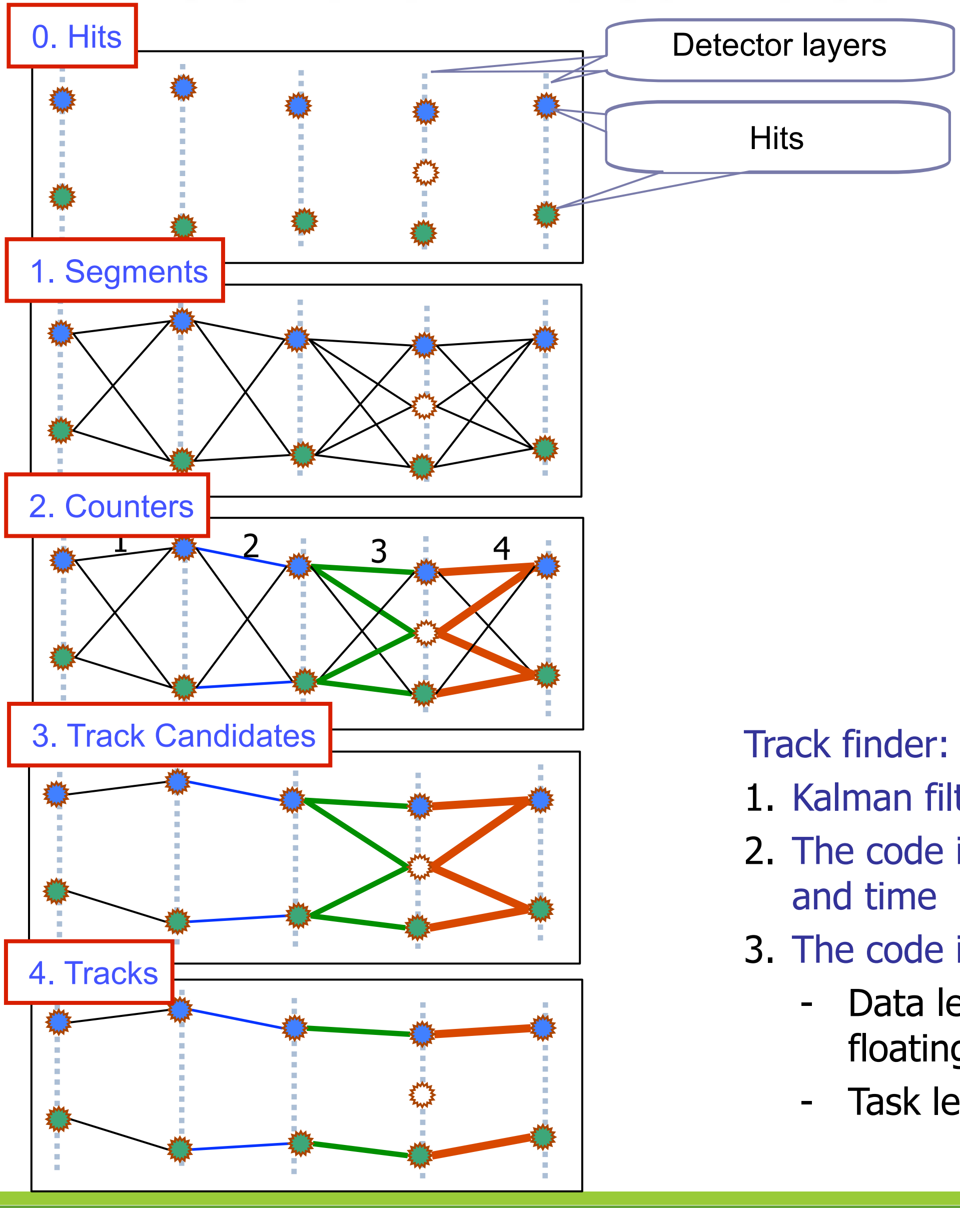


- CBM — a future fixed-target heavy-ion experiment at FAIR, Darmstadt, Germany.
- Interaction rate 10^5-10^7 collisions per second.
- Up to 1000 charged particles/collision.
- Free streaming data.
- No hardware triggers.
- On-line time-based event reconstruction and selection is required in the first trigger level.

- On-line reconstruction at the on-line farm with 60.000 CPU equivalent cores.
- High speed and efficiency of the reconstruction algorithms are required.
- The algorithms have to be highly parallelised and scalable.
- CBM event reconstruction: Kalman Filter and Cellular Automaton.



Cellular Automaton 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:

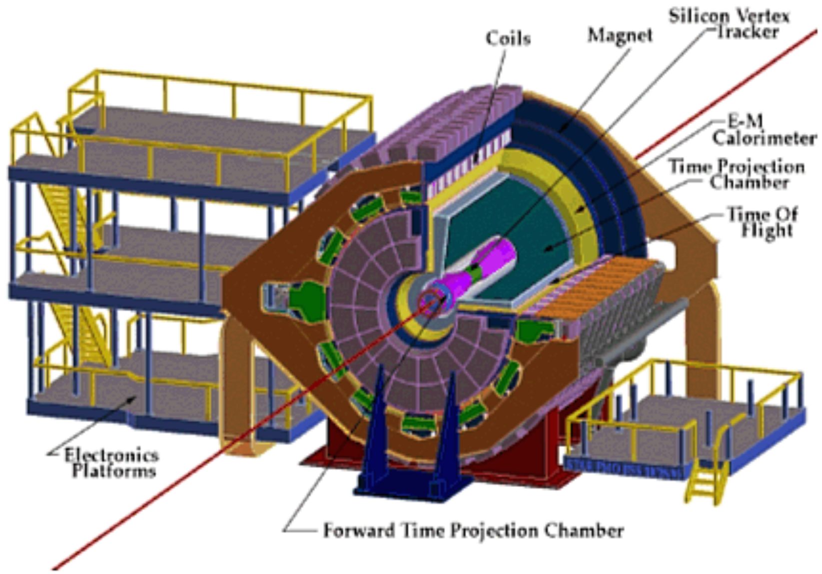
- local w.r.t. data
- intrinsically parallel
- extremely simple
- very fast

Perfect for many-core CPU/GPU !

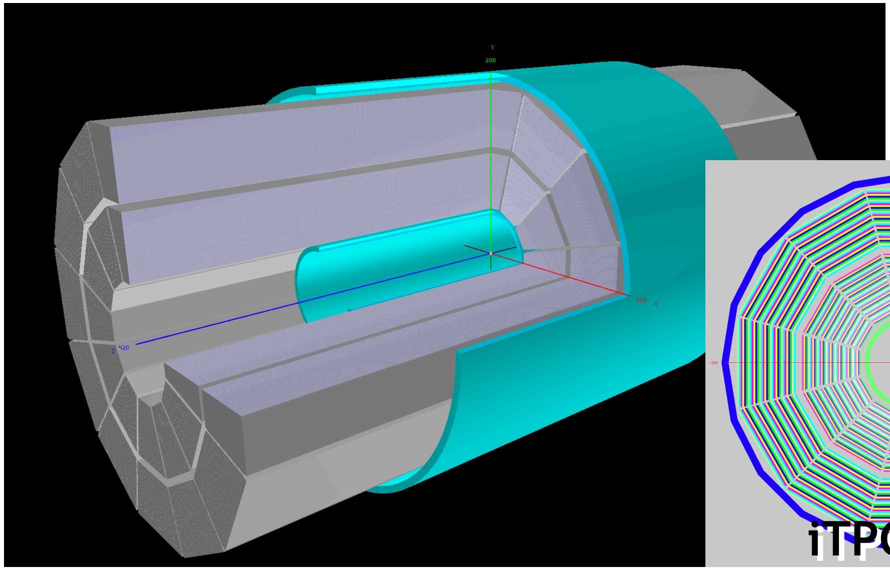
Track finder:

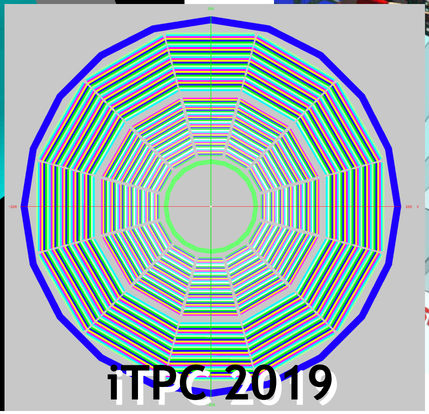
1. Kalman filter for track segments fit
2. The code is optimised with respect to both efficiency and time
3. The code is parallelised
   - Data level (SIMD instructions, 4 single-precision floating point calculations in parallel)
   - Task level (ITBB, parallelisation between cores)

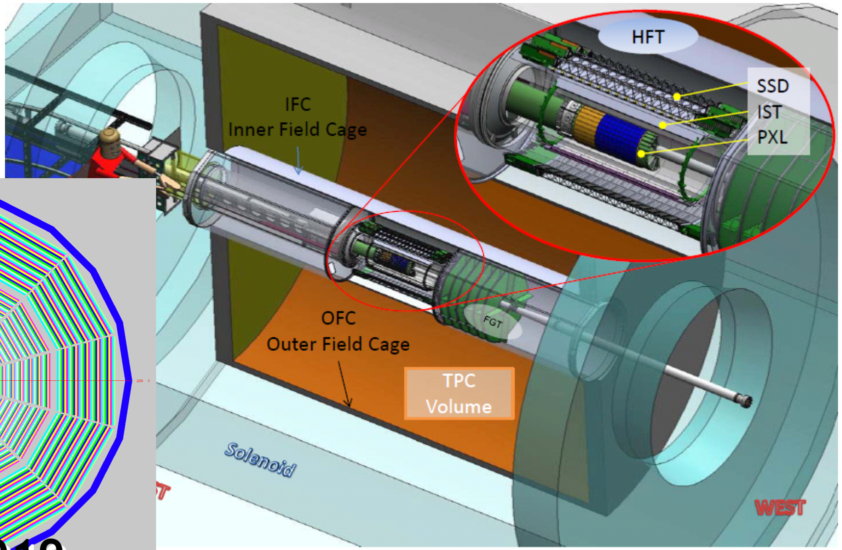
STAR at BNL



- Collider experiment at RHIC, BNL
- Up to 200 AGeV Au-Au collisions
- Main detector – Time Projection Chamber (TPC)
- Standard Sti track reconstruction is based on track following
- Increased RHIC luminosity
- Upgrade the reconstruction algorithms for:
    - vectorization
    - multi-threading
    - many-core systems
- Study of the CA tracking algorithm within FAIR Phase 0

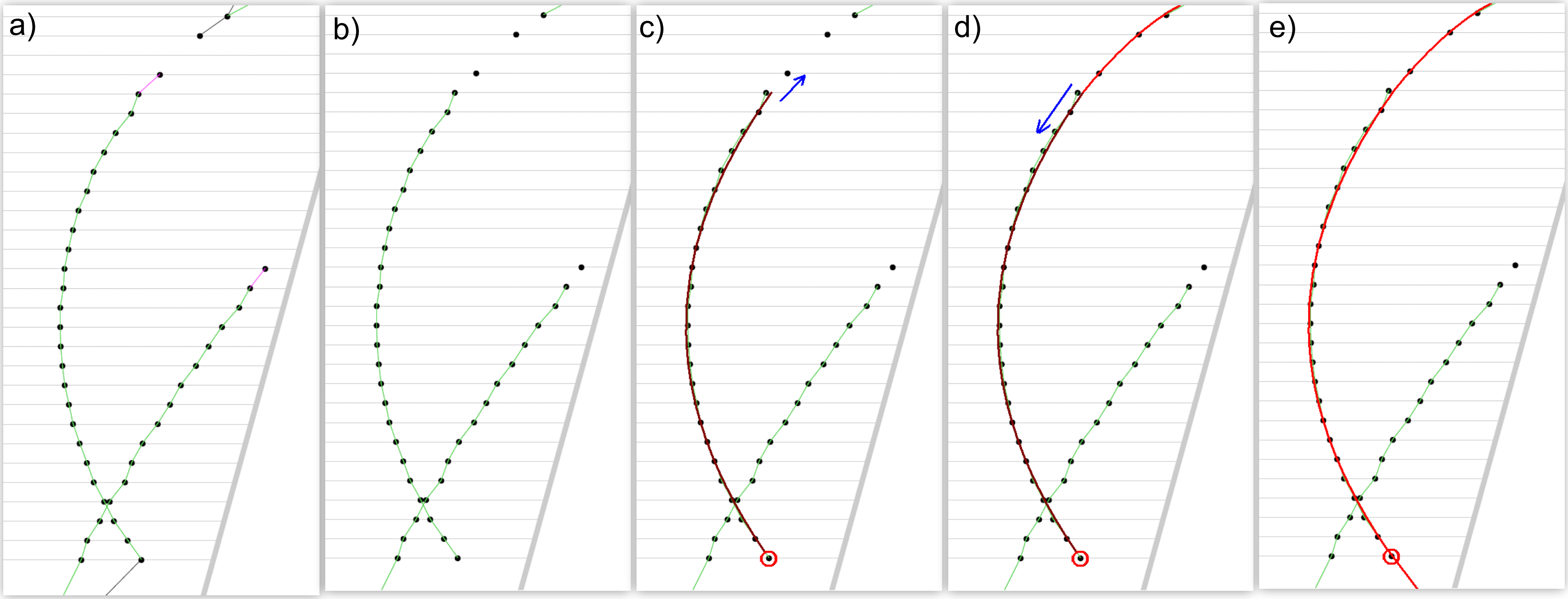






CA track reconstruction in STAR TPC

1. Reconstruction of track segments in each TPC sector:
   a) Find and link neighbouring hits
   b) Clean links
   c) Create segments by fitting chains and adding outer hits
   d) Refit tracks and add inner hits
   e) Selection of tracks
2. Merge sector tracks into TPC global tracks.



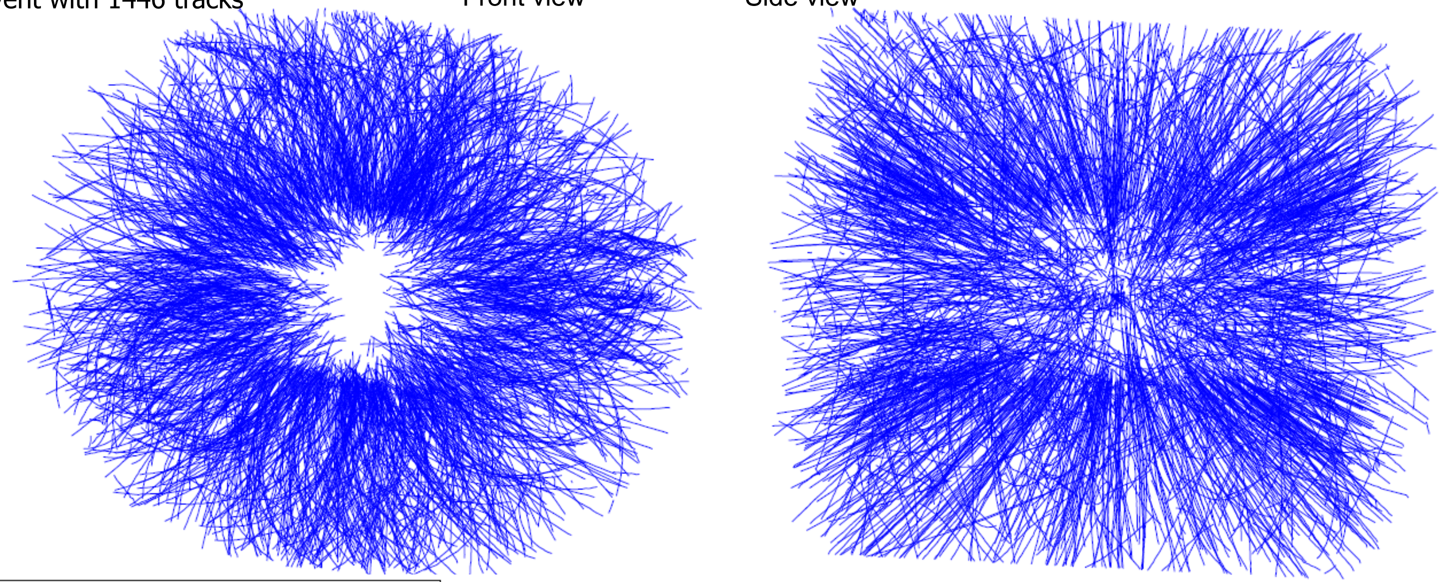
Based on the ALICE HLT TPC CA Track Finder

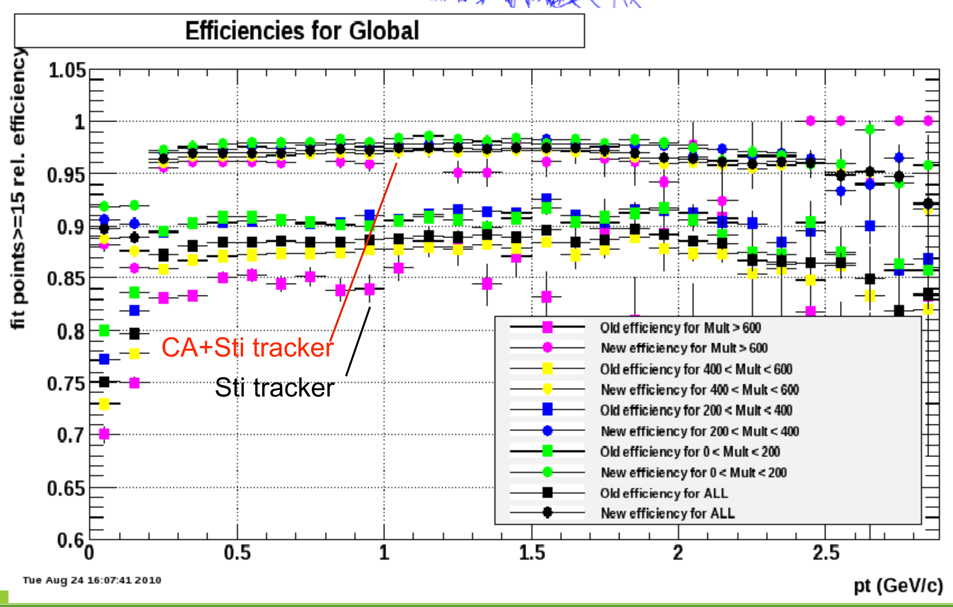
STAR TPC CA Track Finder

Au-Au event with 1446 tracks

Front view

Side view

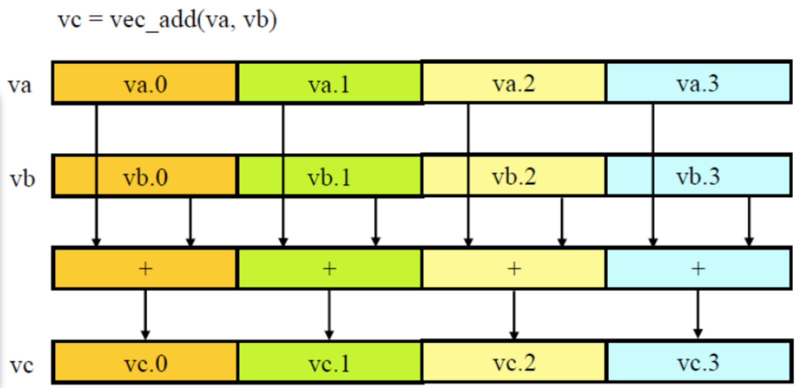




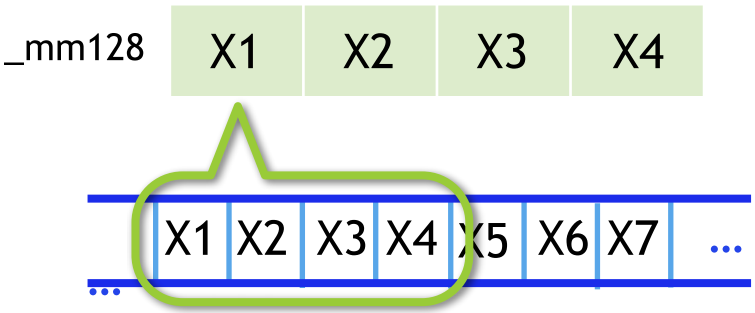
- Stable at high track multiplicity
- Higher tracking efficiency
- About 10 time faster than the track following based Sti track finder

Vectorisation

- **SIMD** intrinsics with **Vc** headers are used.
- Faster calculations with the same hardware.
- Optimal for streaming calculations.

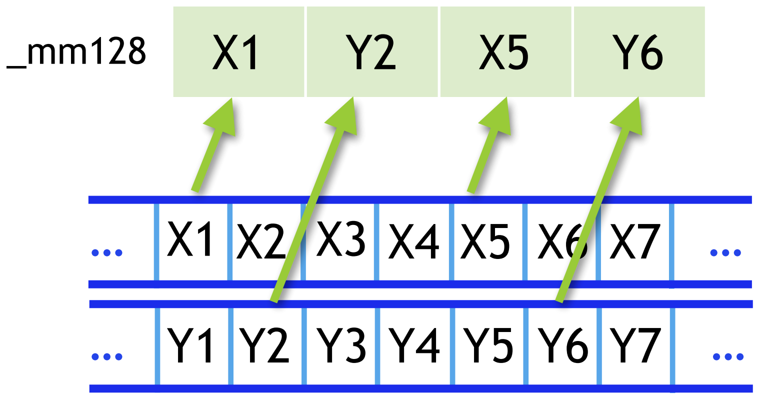


Sequential memory access



- All data in one cash line.
- No overhead for long SIMD registers.
- Maximal speed up.
- Algorithm?

Random memory access



- Data copying from different cash lines.
- More overhead for long SIMD registers.
- Reduced speed up.

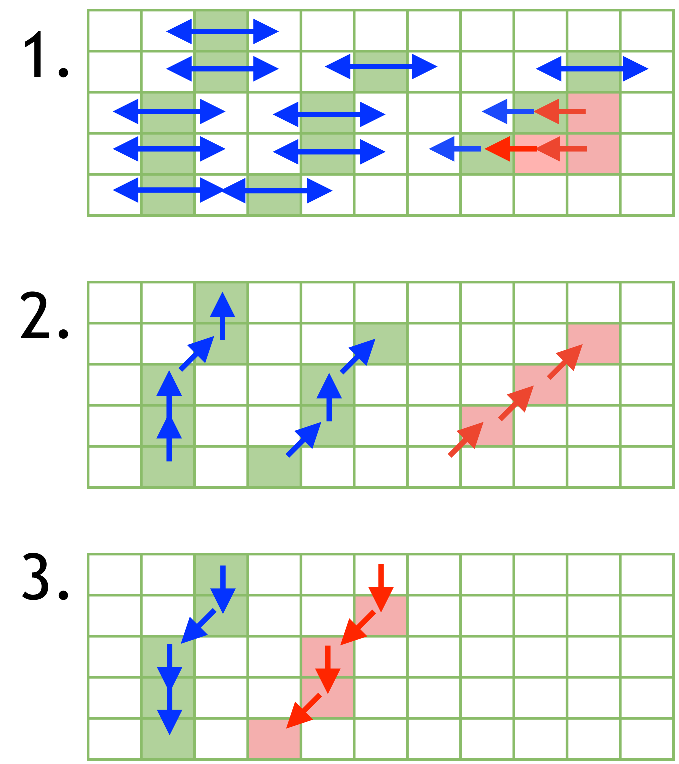
"Game of Life"

The Game of Life is a cellular automaton devised by the British mathematician John Horton Conway in 1970. Its evolution is determined by its initial state and a rule set, requiring no further input.

Same evolution rules for all cells -> no combinatoric -> streaming calculations

In TPC:

- 3D cell structure for every 5 rows
- Cell, containing a hit, is "alive"
- 3 waves of evolution:
    1. Same row - no direct neighbours;
    2. Down to up - lower neighbour exists;
    3. Up to down - upper neighbour exists.
- Combine hits from every corridor into 5-hit segments and pack them in SIMD vectors
- Fit segments
- Store good segments



“Game of Life” and SIMD intrinsics

- Evolution of every cell does not depend on each other.
- Evolution rules are the same for every cell at one step.
- Cell states can be stored in aligned arrays.

- Ready for SIMDization
- Optimal for sequential memory access
- Scalable for register sizes

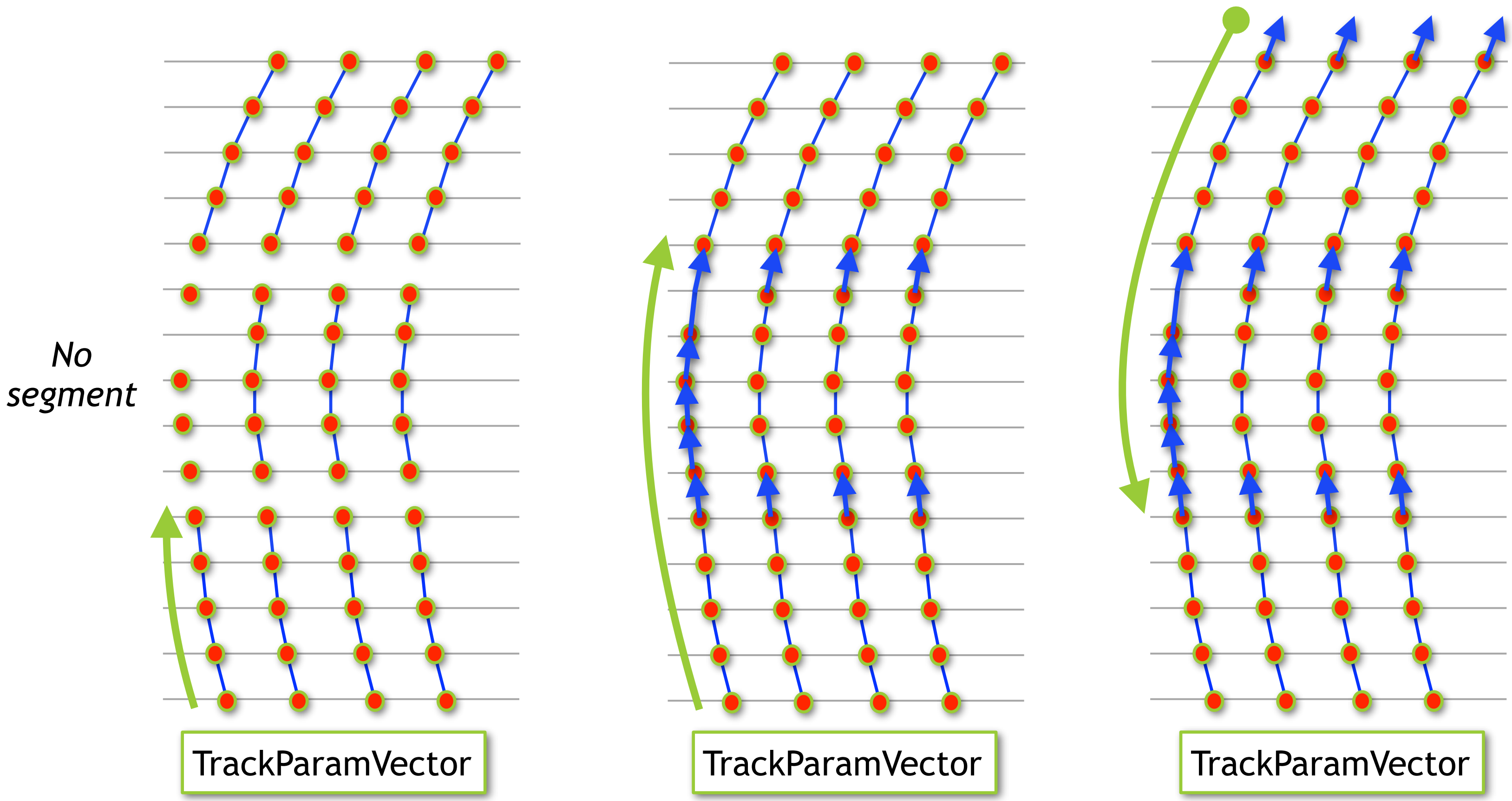
Problem:
Empty cells

- Waste of memory
- Loss of speed up

- Pack data into int8 (char) and int16 (short) arrays.
- Use SIMD intrinsics for int8 elements.
- Maximal speed up factor - 16 (SSE).
- Real speed up factor depends on the track multiplicity.

Tracklet construction

1. Fit segments in the vector mode
2. Extrapolate tracks to the next stations moving to the next segments. Try to catch additional hits if there is no segment.
3. Refit the track back, trying to add more hits in the inner rows.



Results

A new segment constructor:

- Low efficiency for short tracks and tracks with small XZ angle
- Vectorizable and scalable
- Will be used as a first step for the standard CA track finder to improve its calculation speed and scalability

Efficiency

	New	Standard
Long tracks	83.0 %	99.2 %
Reco tracks/ev	298	355
All tracks	55.8 %	93.3 %
Reco tracks/ev	386	653
Clone	5.9 %	10.6 %
Ghost	6.2%	13.7 %

All tracks: $p \geq 0.05$ GeV/c
Long tracks: nHits > 30
Ghost: purity < 90%

Calculation time

	New		Standard tracking	
	Scalar	SIMD	Scalar	SIMD
Sector tracking	17.7 ms	8.1 ms	24.7 ms	16.2 ms
- Chains/Segments	14.2 ms	5.5 ms	15.7 ms	12.2 ms
- Tracklets	3.5 ms	2.6 ms	7.0 ms	3.8 ms

Intel Xeon X5550 at 2.7GHz

Summary

- A Cellular Automaton based track finder in the CBM STS and STAR TPC detectors is fast and efficient even in the case of high track multiplicity.
- A “Game of Life” procedure for track segment finding improves the capability of SIMDization of the tracking algorithm.
- A new segment finding approach should be used as a first step for the standard CA tracking procedure to decrease the combinatoric level and to speed up the algorithm without losing the reconstruction efficiency.

Plans

- Adapt the algorithm for AVX and MIC intrinsics.
- Incorporate the new algorithm into the standard CA approach.