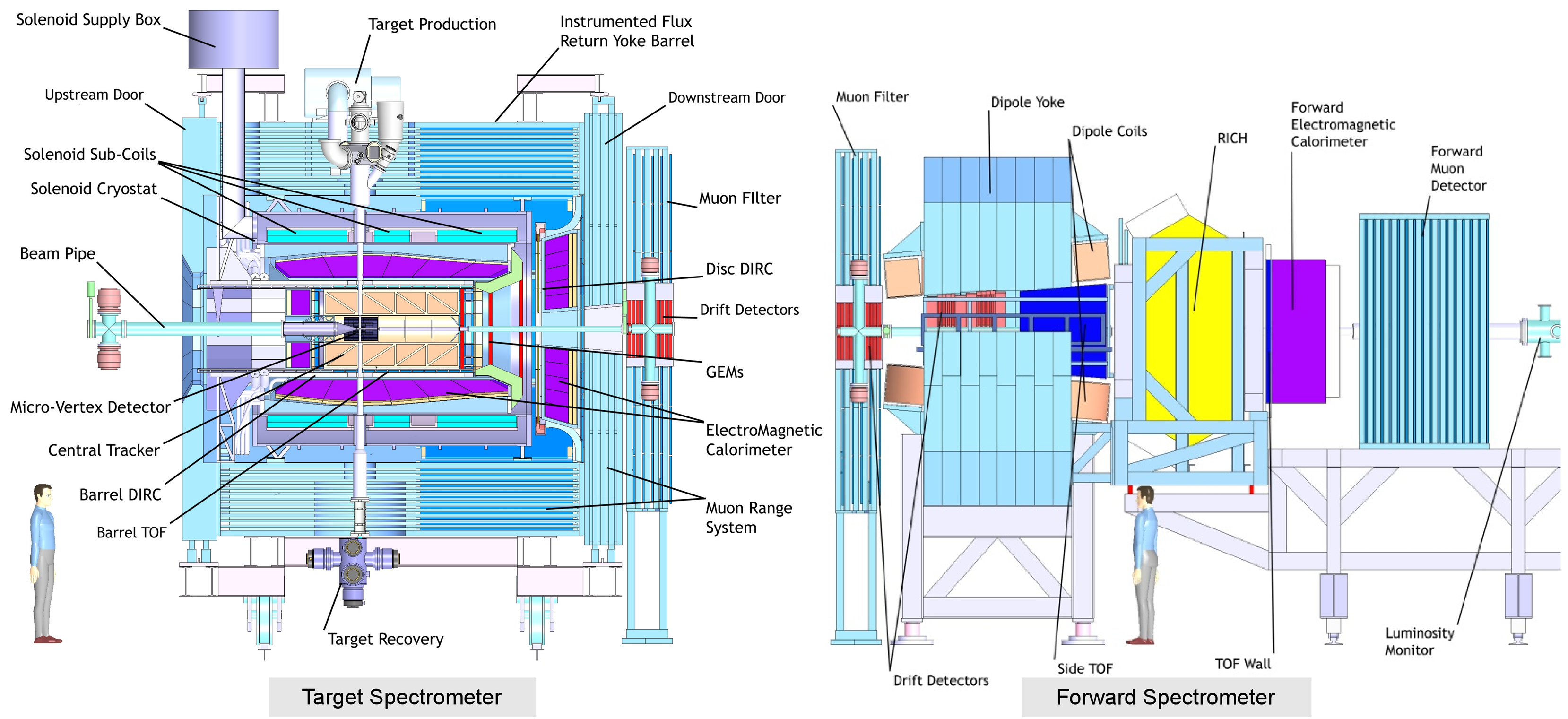
GPU Online Tracking for PANDA

Andreas Herten, Institut für Kernphysik, Forschungszentrum Jülich GmbH

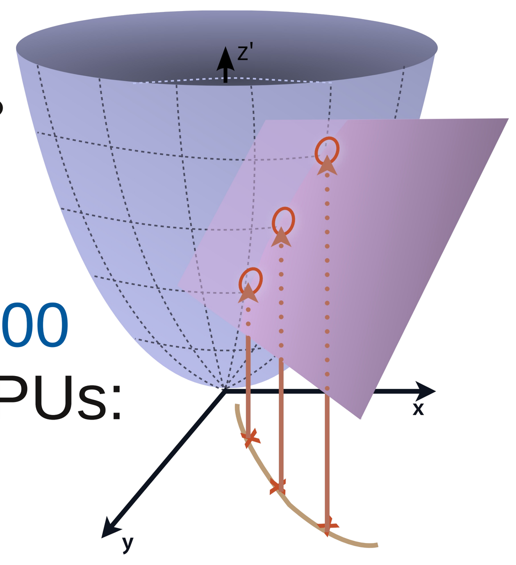
The PANDA Experiment



Currently being built for FAIR (Darmstadt) • Antiprotons colliding with hydrogen • Phase-space cooled beam (1.5 - 15 GeV/c) • Physics: Open charm, exotics

Tracking detectors: MVD & STT

Algorithm: Riemann Track Finder

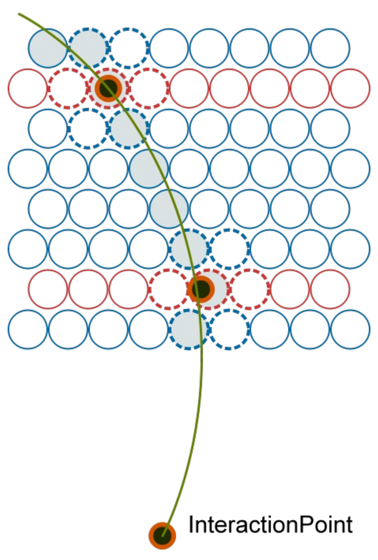


Transform 2D hit points onto Riemann paraboloid • Create intersecting plane • Tranform back

Iterative problem → analytical problem

First implementation for GPU • Speed: 500 µs/event • Specific modifications for GPUs: Non-recursiveness • Flat structure

Algorithm: Triplet Finder

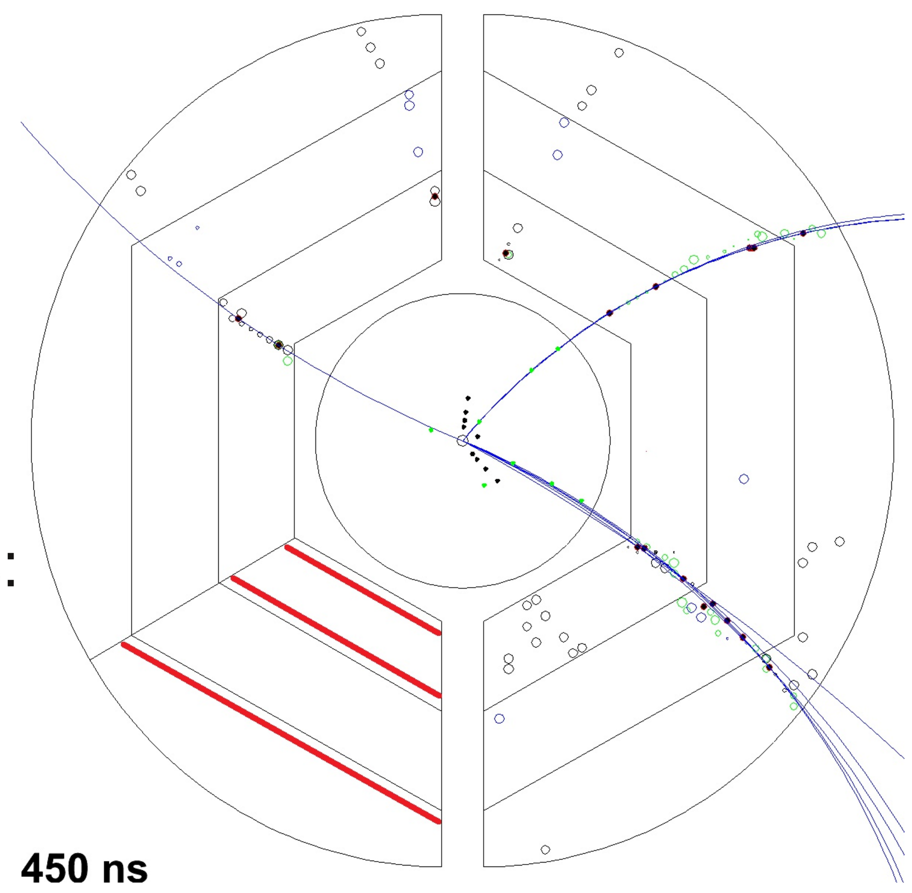


Algorithm specially designed for PANDA's STT • Concept: Use only limited number of straws (pivot straws) to calculate circle • Fast, data streaming algorithm • Possibility to determine event start time t_0

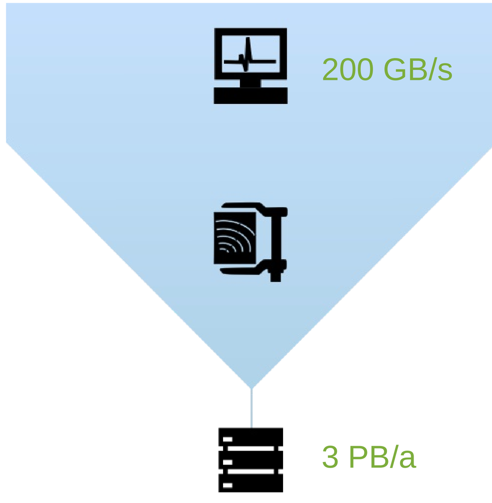
Stages of Triplet Finder:

- Triplet Finding: Use hit of pivot straw together with two adjacent hits to calculate virtual hit (Triplet)
- Circle Calculation: Use Triplet together with second Triplet and (0,0) point for circle generation
- Hit Association: associate all STT hits lying on circle to form track candidate

Right: Snapshot at 450 ns run time. Red lines: Pivot straws



Online Tracking

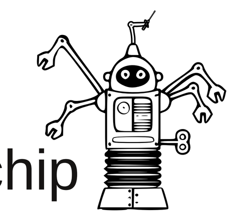


PANDA: Rare physical events • Signal & background similar • Instead of hardware-level trigger: Sophisticated online event filter • Online tracking needed

GPUs - Graphics Processing Units

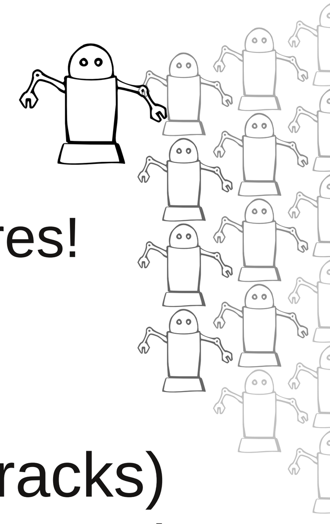
GPGPU: Using GPUs for massively parallel computing • Programming language: CUDA C (NVIDIA)

CPU core



- Sophisticated chip
- Flexibel & Fast

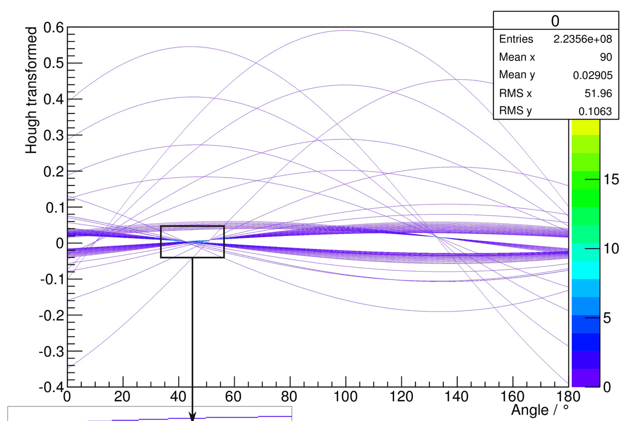
GPU core

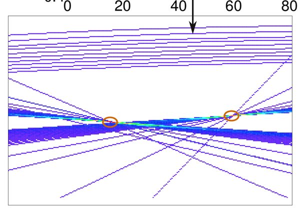


- Simple
- Not as fast
- But: Many cores!

Algorithm: Hough Transform

Easily parallelizable method to find straight lines (tracks) connecting points (hits) • Invented for photograph digitalization of bubble chamber experiments





Needed: Conformal Mapping (circle tracks line tracks)

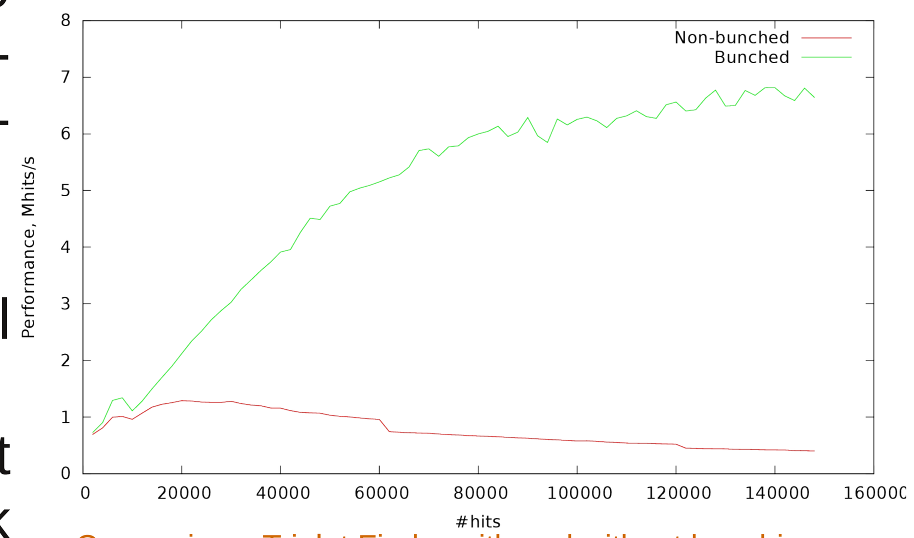
For hit points (x_i,y_i): solve equation $r_{ij} = x_i \cos(\alpha_j) + y_i \sin(\alpha_j) + \rho_i$, α_j selected at j values between 0° and 360°, ρ: drift radius of STT straws • Histogram: (α,r) bin with highest multiplicity is most probable track parameters

GPU versions: Using plain CUDA (incl. Dynamic Parallelism) and Thrust • Speed: 500 µs/event

GPU implementation of Triplet Finder together with **NVIDIA Application Lab** of Jülich Supercomputing Centre

Introduction of GPU-specific optimization:

- Bunching: Always load GPU with amount of hits occupying it the most (best performance) • As a wrapper class (for other online algorihms) • Decoupling of physical event structure • Bunch size: 2 µs (=3200 STT hits) • Reduction of algorithmic complexity ($O(n^2)$ to $O(n)$)
- Different methods of kernel launching
- Hit association: Only test subset of hits against track candidate
- Simplification: No trigonometric & square root functions in code



Comparison: Triplet Finder with and without bunching

Performance: 20 µs/event on single modern GPU

Conclusion & Outlook

- High performance computing challenge at physics
- Different algorithms in research, different stages
- Triplet Finder: Together with NVIDIA Application Lab • Processes 6 10^6 hits/s; needed: 10^9 hits/s
- Using GPUs for PANDA's online tracking seams feasible

Future: Triplet Finder performance test • Further development of other algorithms • Large-scale tests with physics • Data transport to GPU

HELMHOLTZ | ASSOCIATION