# Track and Vertex Reconstruction on GPUs for the Mu3e Experiment

*Dorothea vom Bruch*[1] *for the Mu3e collaboration*

[1]Physikalisches Institut, Universität Heidelberg, Im Neuenheimer Feld 226, 69120 Heidelberg, Germany

**DOI:** will be assigned

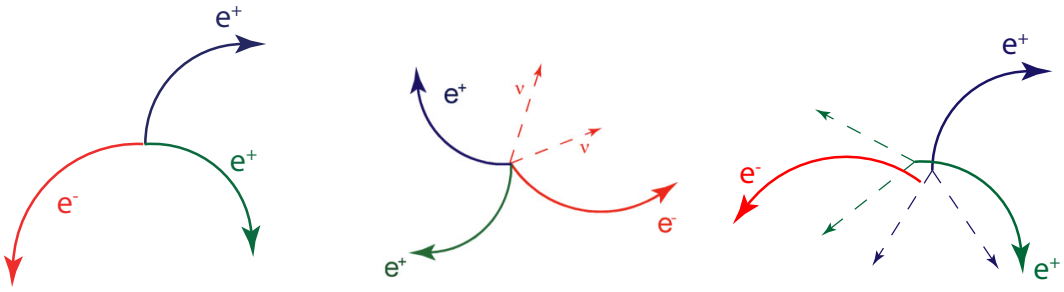
The Mu3e experiment searches for the lepton flavour violating decay $\mu \to eee$, aiming at a branching ratio sensitivity of $10^{-16}$. A high precision tracking detector combined with timing detectors will measure the momenta, vertices and timing of the decay products of more than $10^9$ muons/s stopped in the target. The trigger-less readout system will deliver about 100 GB/s of data. The implementation of a 3D tracking algorithm on a GPU is presented for usage in the online event selection. Together with a vertex fit this will allow for a reduction of the output data rate to below 100 MB/s.

## 1 The Mu3e experiment

The Mu3e experiment [1] searches for the lepton flavour violating decay $\mu \to eee$. Within the standard model, this process is allowed via neutrino oscillations. It is however suppressed to a branching fraction below $10^{-54}$. If lepton flavour violation is observed in the charged lepton sector, this is a clear indication for new physics. Many models beyond the standard model, such as supersymmetry, grand unified models or the extended Higgs sector predict lepton flavour violation at a level to which future detectors are sensitive. The current limit on the $\mu \to eee$ branching fraction was set by the SINDRUM experiment to $10^{-12}$ [2]. The Mu3e experiment aims to improve this limit by four orders of magnitude and to reach a sensitivity of $10^{-16}$ at 90 % CL.

To reach this sensitivity level, the distinction between signal and background events is crucial. In the Mu3e experiment, muons will be stopped in a target and decay at rest. A signal event consists of two positrons and one electron originating from one single vertex as shown in Figure 1a. They are coincident in time and the momentum sum is zero. The total energy of the event is equal to the rest mass of the muon.

One source of background is a radiative muon decay with internal conversion $\mu^+ \to e^+e^-e^+\bar{\nu_\mu}\nu_e$. This process is shown in Figure 1b. Here, the decay products are also coincident in time and have a single vertex, however the momenta do not add up to zero and the energy does not equal the muon rest mass. Combinatorial background stems from two ordinary muon decays $\mu^+ \to e^+\nu_e\bar{\nu_\mu}$ taking place close to each other in space and time together with an additional electron from photon conversion, Bhabha scattering etc. (see Figure 1c). When both the $e^+$ and $e^-$ from Bhabha scattering are detected, only one additional muon decay is required and the probability of misreconstruction as signal event is higher.




(a) Signal event (b) Internal conversion (c) Combinatorial background

Figure 1: Comparison of signal and background events.

Momentum, timing and vertex resolution are therefore the key parameters for the distinction between signal and background events. To achieve the desired sensitivity, Mu3e aims for a momentum resolution $< 0.5$ MeV/c, a timing resolution of 100 ps and a vertex resolution of $< 200$ $\mu$m. These requirements guide the design of the Mu3e detector.

### 1.1 Detector design

Since the muons decay at rest in the target, the momentum of the decay electrons is at maximum half the muon mass (53 MeV/c). In this energy range, the momentum resolution is dominated by multiple Coulomb scattering whose variance is inversely proportional to the momentum. Therefore the design of the Mu3e experiment is aimed at minimizing the material budget. Ultralight mechanics will be used for construction and a pixel tracking detector is built from high voltage monolithic active pixels sensors [3, 4, 5] thinned to 50 $\mu$m, with a pixel size of $80\times80$ $\mu$m$^2$. In addition, scintillating tiles and fibres are included for precise timing information. The muons are stopped on the surface of a hollow double cone target, leading to a spread in the vertex locations. A magnetic field of 1 T is applied so that the tracks are bent and recurling tracks are detected by outer detector regions. This allows for a more precise momentum measurement. A schematic of the detector is shown in figure 2. To reach the sensitivity of $10^{-16}$ within a reasonable time, muons at high rates are desired which will be provided by the Paul Scherrer Institut. At the current beamlines, up to $10^8$ $\mu$/s are available. A future high intensity muon beam line (HiMB) could deliver in excess of $2 \cdot 10^9$ $\mu$/s.

## 2 Readout scheme

A triggerless readout is foreseen for the detector. At the above mentioned rates this results in a data rate of about 100 GB/s. Consequently, an online selection is required to cope with this amount of data and reduce the rate by a factor of 1000. A schematic of the readout is shown in Figure 3. Zero-suppressed data are sent from the pixel sensors, the fibres and the tiles to front-end FPGAs via flex print cables. The FPGAs merge the data and sort it into time slices of 50 ns which are then transferred via optical links to the readout boards. Each readout board receives the information from each sub-component of the detector and then sends the complete detector information for one time slice to one computer of the filter farm via optical links. With

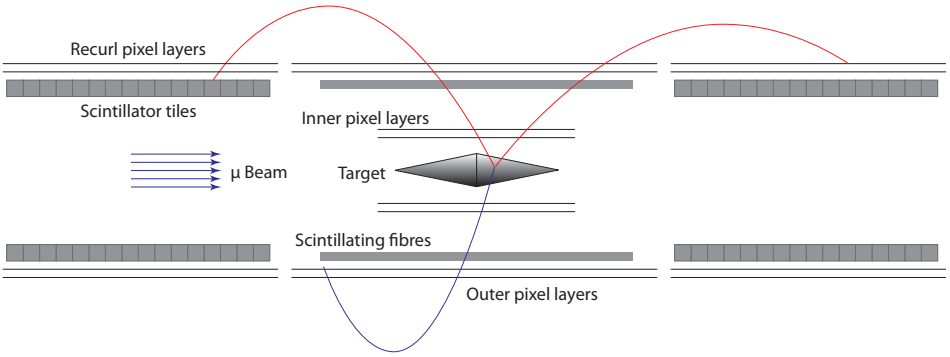



Figure 2: Schematic of the Mu3e detector design.

this procedure, each PC analyzes different time slices of full detector information. The GPUs of the filter farm PCs perform the online event selection by searching for 3 tracks originating from one single vertex.

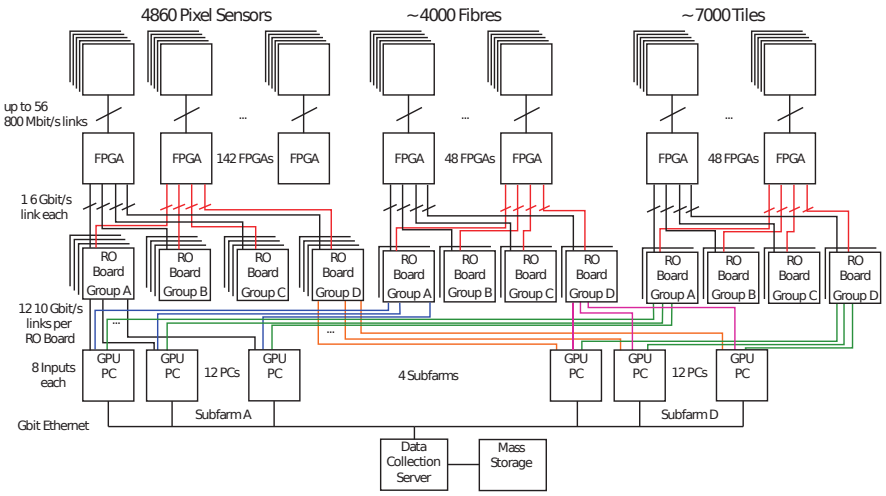



Figure 3: Readout scheme of the Mu3e detector design. [6]

# 3 Multiple scattering fit

Tracks are reconstructed based on the hits in the pixel detectors using a 3D tracking algorithm for multiple scattering dominated resolution [7, 8]. Triplets of subsequent hits in three layers are selected and multiple scattering is assumed at the middle hit of the triplet. In the momentum

region of interest the position resolution of the pixel detector ($\sigma_{\text{pixel}} = 80/\sqrt{12}$ $\mu$m) is small compared to the effects of multiple scattering.

Figure 4 shows one triplet with the azimuthal scattering angle $\Phi_{MS}$ in the x-y plane on the left and the polar scattering angle $\Theta_{MS}$ in the s-z plane on the right, where s is the 3D path length. The combined $\chi^2 = \frac{\phi^2_{MS}}{\sigma^2_{MS}} + \frac{\theta^2_{MS}}{\sigma^2_{MS}}$ is minimized during the non-iterative fitting procedure. The variances of the scattering angles are obtained from multiple scattering theory. Several triplets grouped together form one track.

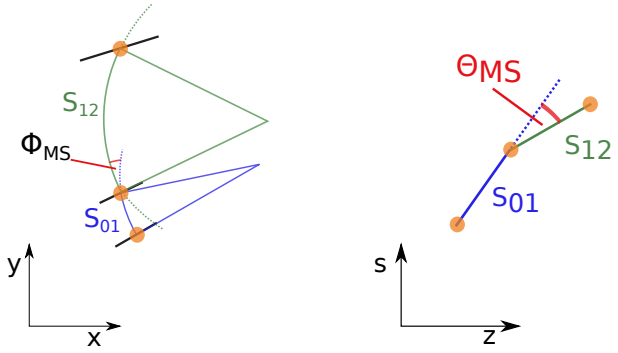



Figure 4: Sequence of three hits forming a triplet in the x-y plane on the left, and the s-z plane on the right with respective scattering angles.

# 4 Fit implementation on GPU

For online event selection, a basic version of the multiple scattering fit runs on the GPUs of the filter farm PCs. It has been implemented on Nvidia's GeForce GTX 680 using the CUDA environment. Triplets are constructed from hits in the first three detector layers. The number of possible hit combinations to form a triplet is proportional to $n_1 \times n_2 \times n_3$ where $n_i$ is the number of hits in layer i. The main steps of the processing are:

- Sorting the hit array with respect to the z-coordinate
- Geometric filtering on ordered hits
- Triplet fitting and selection

Notice that in the final readout mechanism, the hit arrays will be sorted by the FPGAs in the PCs and geometrical selection cuts will be applied. The preselected arrays will then be copied via direct memory access to the GPUs to perform the fitting step. Currently the CPU sorts the hit arrays with respect to the z-coordinate, then the sorted hit arrays are copied to the GPU global memory.

A preliminary kernel implementation gathers the hit filtering and triplet fitting. The selection cuts require proximity in the x-y plane and in z for pairs of hits in layers one and two, and two and three respectively. If these cuts were passed, the fitting procedure is applied and hits of triplets are saved in global memory with an atomic function given a successful fit completion and a certain $\chi^2$.

For the first kernel implementation, a 2D-grid is set up with $n_1 \times n_2$ grid dimensions, each kernel loops over the $n_3$ hits in layer three. Since the geometrical selection cuts and fit completion cuts introduce branch divergence, 87 % of the threads are idle in this configuration.

Consequently, an alternative implementation with two separate kernels was tested. Within the *geometric kernel*, only the geometry cuts are applied and triplets of hits passing these cuts are saved into an array in global memory. The grid dimensions for this kernel are the same as described for the first implementation. Then, the *fit kernel*, which fits and selects hit triplets, is launched on the number of selected triplets using a 1D-grid associated with 128-block size. Consequently, one fit is performed per thread without any divergence. In addition, since the block dimension is now a multiple of the warp size (32), no inherently idle threads are launched in the grid.

In terms of run time, there is no significant improvement with the two kernel version compared to one kernel. However, only tasks for which the GPU is optimized are now performed in the fitting kernel, so this is one step towards the final readout and selection chain of the filter farm.

# 5 Performance

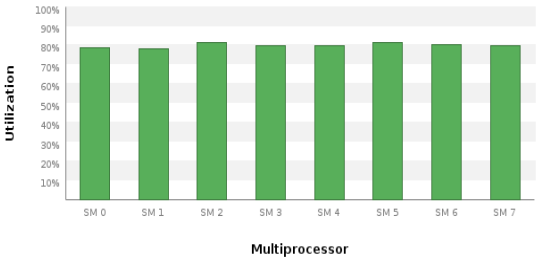



Figure 5: Compute utilization of the streaming multiprocessors on the GTX 680 for the fitting kernel of the two kernel fit implementation.

Next we will focus on the triplet fitting performance, since it will be the processing step deployed on GPUs.
Using Nvidia's Visual Profiler tool, the performance of the different versions was studied. The compute utilization of the streaming multiprocessors averages around 80 % for the fitting kernel of the two kernel version, as shown in Figure 5.

The block size has been chosen to optimize the number of warps per streaming multiprocessor: 128 threads per block take advantage of all 64 warps in a streaming multiprocessor (see Figure 6 on the left). Since the fitting kernel requires 29 registers per thread, the kernel implementation can make optimal use of these 64 warps per streaming multiprocessor (see Figure 6 on the right). Therefore, the compute efficiency is not limited by the block size or register count.

Currently, $1.4 \cdot 10^{10}$ triplets/s are processed. This measurement is based on the wall time of both CPU and GPU, so that all sorting and selection times are included. Most of the time is spent on the first kernel applying the selection cuts. Therefore, further improvement is expected when the pre-selection is outsourced to the FPGAs on the readout boards. Similarly, the ratio of copying data from CPU to GPU compared to computing time (40 %) will improve once more selection criteria are applied before transferring the data to the PC.

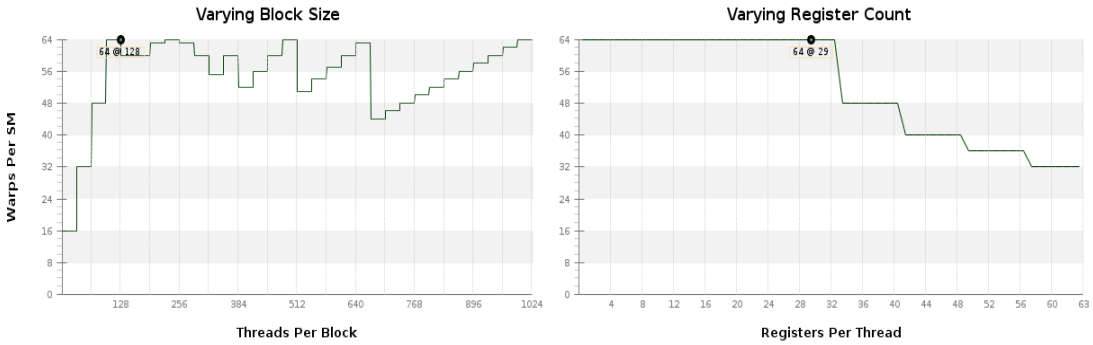



Figure 6: Warps per Streaming Multiprocessor (SM) as a function of the block size (left) and the register count (right).

# 6 Summary and outlook

To achieve a sensitivity of $10^{16}$ in the measurement of the $\mu \to eee$ branching ratio, high rates on the order of $10^9 \mu/s$ are required, resulting in a data rate on the order of 100 GB/s in the detector.

The implementation of the triplet fit on the GTX 680 can currently process $1.4 \times 10^{10}$ triplets/s. For a muon rate of $10^8$, about $10^{12}$ hit combinations are expected per second in the first three layers, therefore requiring 10 - 100 GPU computers in the filter farm. Faster filtering is required for the higher rate of $10^9 \mu/s$. This can be achieved by sorting the data on FPGAs and further improving the performance of the fit, and as a result, reduce by a factor of 1000 the data rate. In addition, a new vertex fit [9] will be implemented on the GPU as well in order to apply the selection criteria for a signal of 3 tracks originating from one single vertex.

# References

[1] A. Blondel et al. Research Proposal for an Experiment to Search for the Decay $\mu \to eee$. *ArXiv e-prints*, January 2013.

[2] W. Bertl et al. Search for the decay $\mu^+ \to e^+e^+e^-$. *Nucl. P*, B 260(1):1 – 31, 1985.

[3] N. Berger et al. A Tracker for the Mu3e Experiment based on High-Voltage Monolithic Active Pixel Sensors. *Nucl. Instr. Meth. A*, 732:61–65, 2013.

[4] I. Perić. A novel monolithic pixelated particle detector implemented in high-voltage CMOS technology. *Nucl.Instrum.Meth.*, A582:876, 2007.

[5] I. Perić et al. High-voltage pixel detectors in commercial CMOS technologies for ATLAS, CLIC and Mu3e experiments. *Nucl.Instrum.Meth.*, A731:131–136, 2013.

[6] S. Bachmann et al. The proposed trigger-less tbit/s readout for the mu3e experiment. *Journal of Instrumentation*, 9(01):C01011, 2014.

[7] A. Schöning et al. A multiple scattering triplet fit for pixel detectors. *publication in preparation.*

[8] M. Kiehn. Track fitting with broken lines for the mu3e experiment. Diploma thesis, Heidelberg University, 2012.

[9] S. Schenk. A Vertex Fit for Low Momentum Particles in a Solenoidal Magnetic Field with Multiple Scattering. Master's thesis, Heidelberg University, 2013.