



Characterisation of an electron beam profile monitor for the FLASHForward target chamber

Frederic Van Assche

DESY Hamburg, Germany & Ghent University, Belgium

September 10, 2015

Abstract

The FLASHForward experiment is a future plasma-wakefield accelerator experiment. One component is a high precision electron beam profiling system placed in front of the plasma cell. This is implemented as a scanning component (a wire, knife edge, ...) being moved through the beam while detectors downstream pick up the scattered radiation, giving an integrated profile of the beam. As this beam will be a strongly focused microbeam of a size comparable to the scanner component geometry, the observed signal will contain a convolution of scanner component geometry and beam profile. This report explores the simulation of measurements for a range of scanner components, and the performance of the novel fitting models required to extract the beam parameters of interest.

Contents

1	Introduction	3
1.1	The FLASHForward experiment	3
1.2	Goal of this project	3
2	The Geant4 simulation code	5
2.1	Code modifications	5
2.2	Remark on random number quality in simulation code	5
2.3	Included physics processes	8
2.4	Basic geometry	9
2.5	Generated data	9
2.6	Computing infrastructure	11
3	The Gauss-Newton NLLS fitting algorithm	11
4	Simulation results	12
4.1	Geometry	12
4.1.1	Example curves for the different scanner components	13
4.1.2	Different scanner component materials	15
4.1.3	Behaviour with very thick scanning components	16
4.2	Fitting performance	19
5	Conclusions	24
5.1	Simulated measurement results	24
5.2	Fitting algorithms	24
5.3	Future improvements and goals	24
6	Acknowledgements	25

1 Introduction

1.1 The FLASHForward experiment

FLASHForward is a future plasma-wakefield acceleration experiment to be performed as part of a new beamline splitting off from the FLASH accelerator at DESY Hamburg [1]. This experiment is an example of a hybrid beam-driven plasma-wakefield accelerator, where a laser shot into a gas target ionises this gas to the point of becoming a plasma. The electron beam provided by FLASH then injects the driver bunch into this plasma, finally accelerating a witness bunch of electrons from the ionised gas to energies of order GeV over just centimeter distances.

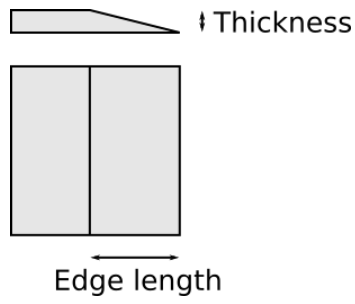
It is clear that both the ionising laser and the electron beam need a tightly constrained profile and good alignment. For this purpose, two beam profiling systems will be placed as close to the front of the plasma cell as possible. The electron beam profiler will be implemented as a scanning component (a wire, knife edge, ...) being moved through the beam by a hexapod platform, while beam loss monitors downstream pick up the intensity of the generated secondary radiation.

The electron beam will be a focused microbeam at this point, on the order of micrometers across. This means that the traditional approach to a wire or knife edge scanner beam profiler no longer applies, as the beam size will be equal or smaller than the geometrical features of the scanner component. The signal detected by the BLMs will thus show a convolution of the beam profile and the scanner component geometry, which requires more complicated fitting algorithms and models if one wants to extract useful beam parameters from these curves.

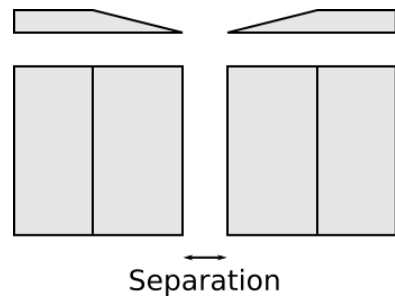
Figure 1 shows sketches of the two scanning components with a more complicated geometry. The knife edge (and slit scanner) can have a non-zero *edge length*. For a slit scanner, the *separation* length can also be varied.

1.2 Goal of this project

At the start of this project, a basic simulation of a wire scanner and knife edge scanner component was available using the Geant4 toolkit. Similarly, a Gauss-Newton non-linear least squares fitting algorithm was implemented in MATLAB, including fitting functions for knife edge, wire and slit scanners. The goal was to extend and improve the simulation code where possible, to the point where a set of simulations could be performed to validate the performance and output of the fitting algorithms.



(a) Sketch of a knife edge component. The edge length can be zero, in which case it is a simple box shape.



(b) A slit scanner is modeled as two mirrored knife edges with non-zero separation.

Figure 1: Sketches of a knife edge and slit scanner component. A profile view is shown on top, and a view along the beam axis on bottom.

2 The Geant4 simulation code

The simulations used in this project are based on Geant4 [2]. This toolkit allows one to simulate the passage of particles through matter and is widely used in detector and accelerator physics, as well as medical applications where radiation is involved.

The code available at the start of the project created a minimal geometry of the target chamber and scanner components, and included the basic tools needed to get simulated measurements for a range of configurations.

2.1 Code modifications

- The code was taken from the original single-threaded execution, to multi-threaded operation.
- A slit scanner component (roughly the inverse of a wire scanner) was added to the available components, and the flat knife edge and linear knife edge were merged into one.
- A command interface was added to allow changing the scanner component material between different runs, both using a material's name, or by specifying elements and their mass fractions.
- Energy deposition and secondary particle generation were added to the measured observables.
- A general cleanup and simplification of the code was performed, and several smaller bugs were fixed along the way.

2.2 Remark on random number quality in simulation code

While the exact requirements will vary, simulation code will generally depend on high quality (pseudo)random number generators. It is always important to verify the correct operation of these random number generators before production use of a simulator. Switching code from single- to multi-threaded execution can add additional complexities to proper random number generation.

During the course of this project, an issue was identified and corrected with regards to the random number generators provided by Geant4. The behaviour of the toolkit was in clear conflict with its documentation. The first few batches of simulations raised suspicions by showing very similar “features” in their randomness, shown in figure 2. After running a set of simulations with identical parameters and obtaining perfectly identical data for each simulation in the set, it became clear that the random number generator was not being seeded properly.

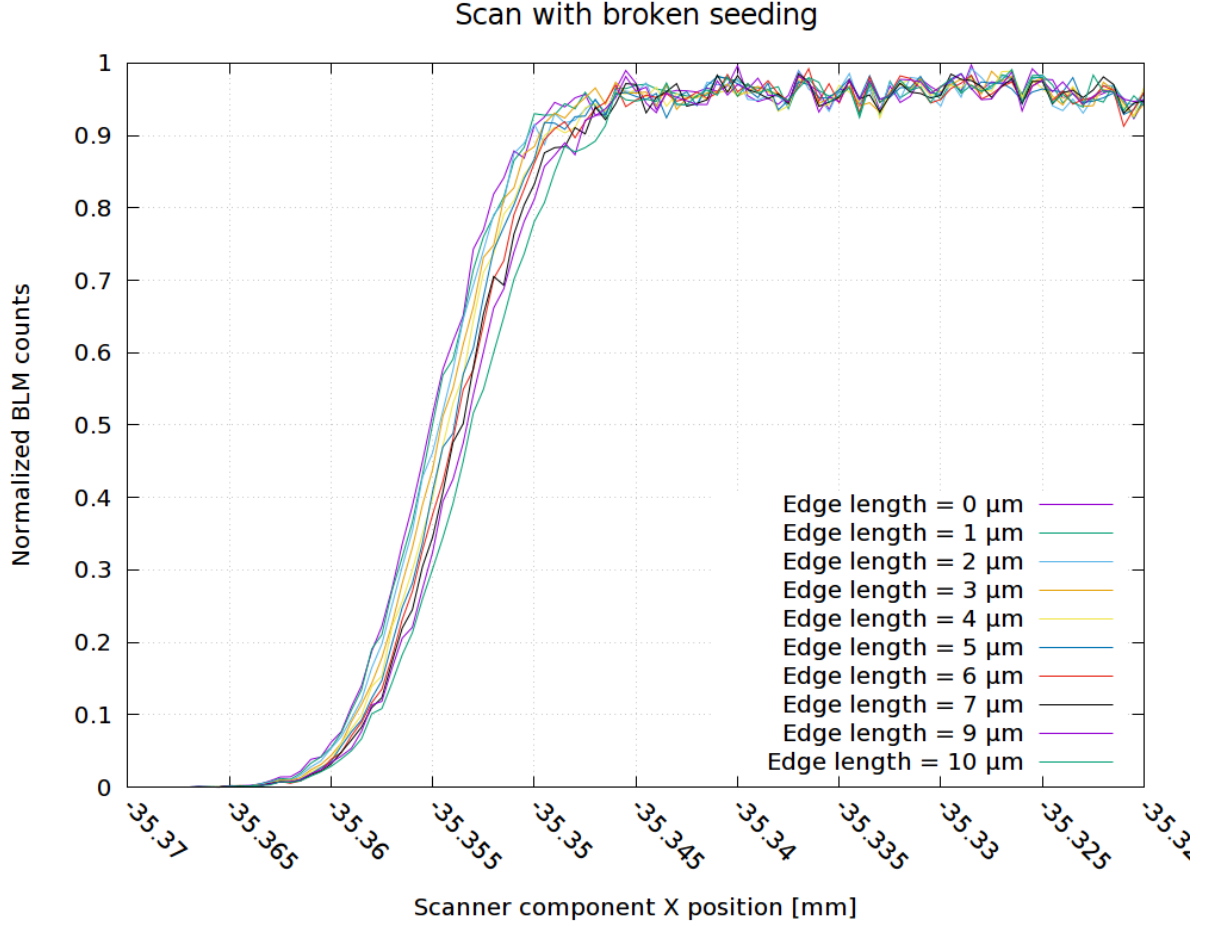


Figure 2: Example set of simulations where each one uses the exact same string of random numbers. The right part of all curves show stronger similarities than expected from random data.

Attempting to set a random seed through the appropriate Geant4 API had no effect on the actual random numbers. The code was modified to use the 24-bit RANLUX engine [3] provided by the C++11 standard for primary event generation. The internal engine used by the Geant4 toolkit was unchanged causing the internal processes (ionisation, decay branches, ...) to still use the same seed every simulation. However, proper randomisation of the primary events already greatly improved the overall simulation quality, shown in figure 3.

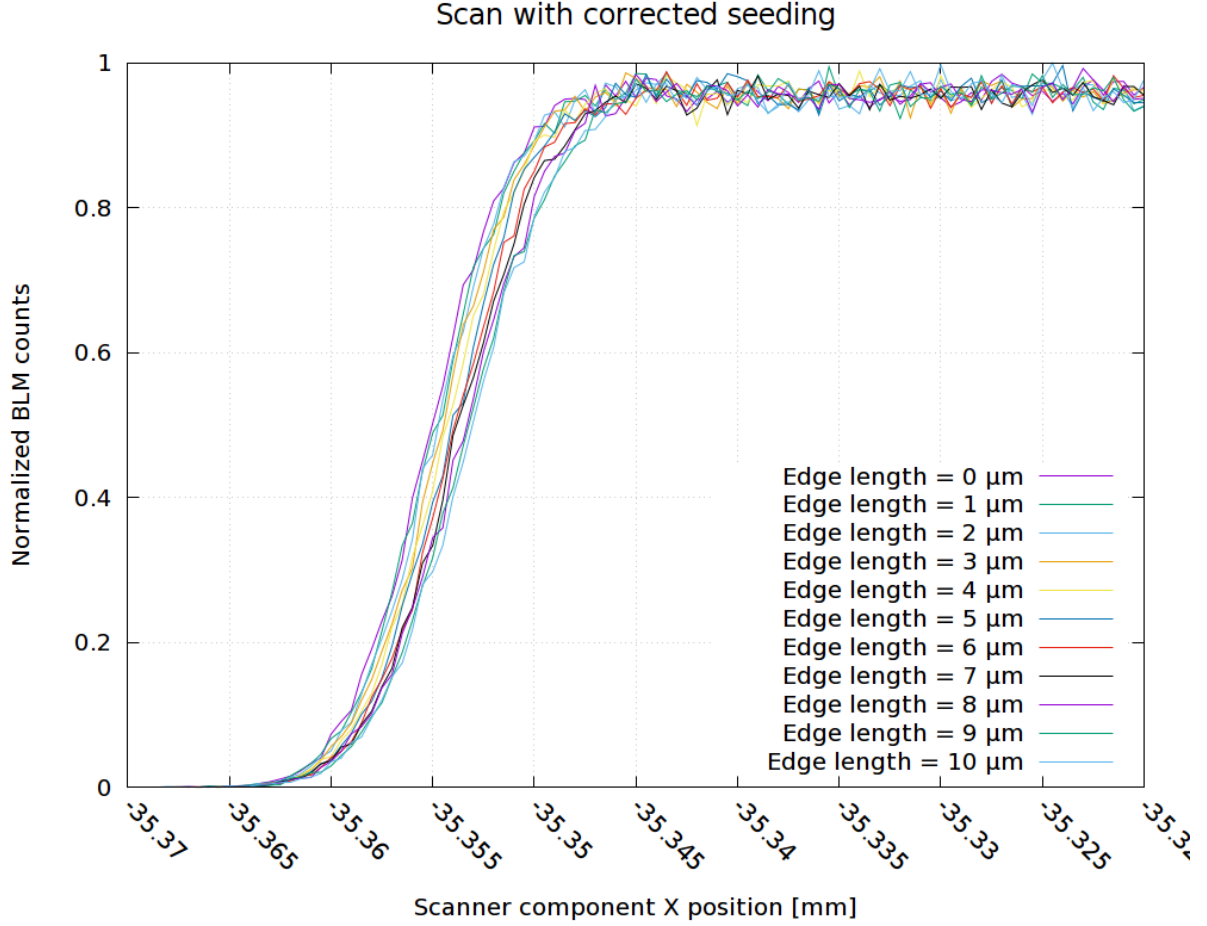


Figure 3: Example set of simulations with proper random number generator seeding.

To seed a random number generations, two possible choices are:

- Seeding every simulation using a completely random seed obtained through other means.
- Using a predictable but unique seed for every simulation. This allows to easily repeat a previous simulation, for example to visualise specific events of interest inside one run.

For this code the first option was chosen, and a hardware true random number generator¹ provided the initial seed of every simulation. Repeat simulations with the same seed were not as important as the code only outputs summed values of all events, and digging down to specific events is not important. If this would be desirable in the future, this could trivially be added by including the seed in the output and allowing a seed to be specified when setting up a simulation.

¹Often implemented through sampling of entropy gained from user input, network chatter and free-floating audio inputs.

2.3 Included physics processes

Only electrons, positrons and photons were included in the simulation. All hadronic processes were neglected. Of the electromagnetic physics processes offered by Geant4, the photoelectric effect, compton scattering, gamma conversion, bremsstrahlung and general ionisation processes were used. The more precise Penelope model was used for bremsstrahlung, ionisation and photoelectric effect, in large part because it is the only available model when positrons are to be simulated. The remaining processes used the standard Geant4 models, which are valid for the 1 GeV energy of the electron beam.

These choices were validated using a separate toy simulation, where electrons were launched at a thin steel target. The angular distribution and particle populations were observed for different combinations of physics processes and models. More than 99.9% of all observed particles were electrons, positrons or photons. Variation of the angular distribution of the produced secondary radiation was not observed between more complete sets of processes, and the simplifications employed for the simulations during this project. Figure 4 shows the difference in radial profile between an EM-only simulation, and one also including hadronic processes. Table 1 shows an overview of the two tests. The difference in generated particle populations for otherwise identical conditions was found to be negligible. Energy deposition inside the steel target was also found to vary for less than one percent.

Limiting the simulations to electromagnetic processes and only the three most relevant particles provided a good tradeoff between precision and computational complexity.

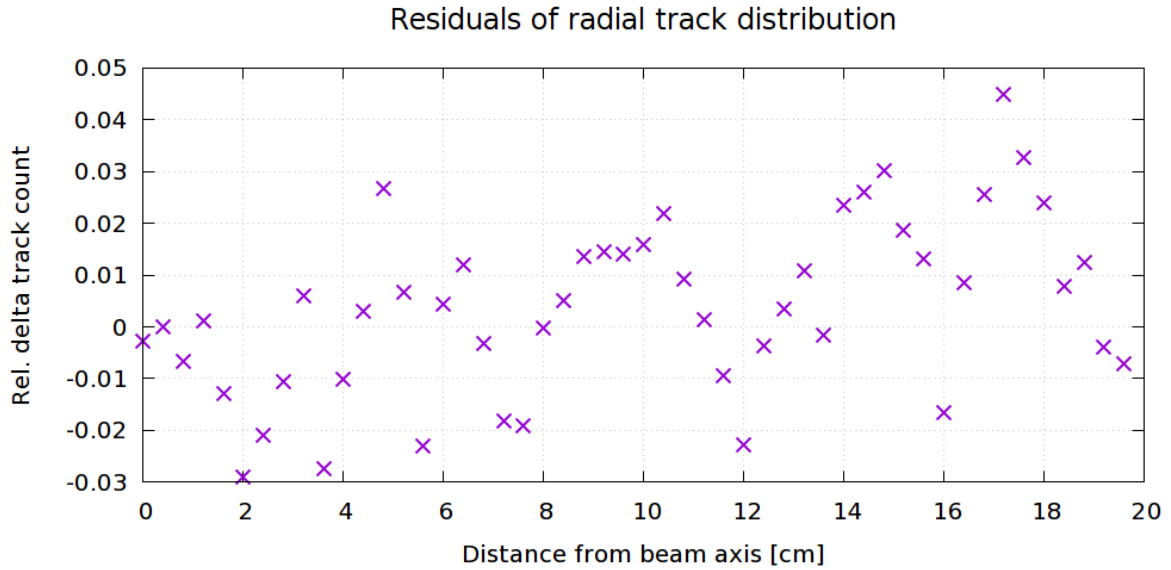


Figure 4: Relative difference in radial profile of the tracks going through a disc-shaped region downstream of a fixed steel target. One simulation only uses electromagnetic processes, the other uses a more complete EM + hadronic setup.

Particle type	EM-only	EM + Hadr.	$\Delta_{relative}$
γ	621 642	623 536	2.4×10^{-3}
e^-	136 568	136 426	-1.8×10^{-4}
e^+	27 990	28 107	1.4×10^{-4}
p	0	5	6.4×10^{-6}
n	0	46	5.9×10^{-5}
π	0	2	2.5×10^{-6}
others	0	16	2.0×10^{-5}
Σ	786 200	788 138	2.5×10^{-3}

Table 1: Simulation of a toy setup using two different sets of physics processes. The rightmost column shows the difference divided by the total count of the EM-only set.

2.4 Basic geometry

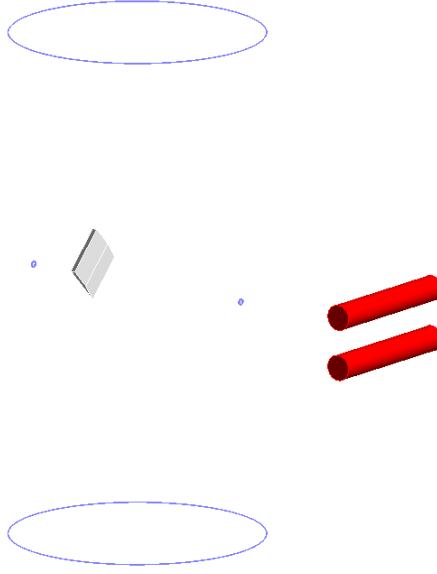
The simulated geometry is shown in figure 5. The target chamber and beam pipes are modeled as stainless steel cylinders containing a vacuum. The volume outside the chamber contains air at normal pressure and density.

The beam loss monitors are simulated as vacuum cylinders, mainly keeping track of the amount of particles passing through them. It is possible to replace this material to more closely match actual detectors, for example using scintillators. This will however increase the computational complexity, and was not necessary given the scope of the project.

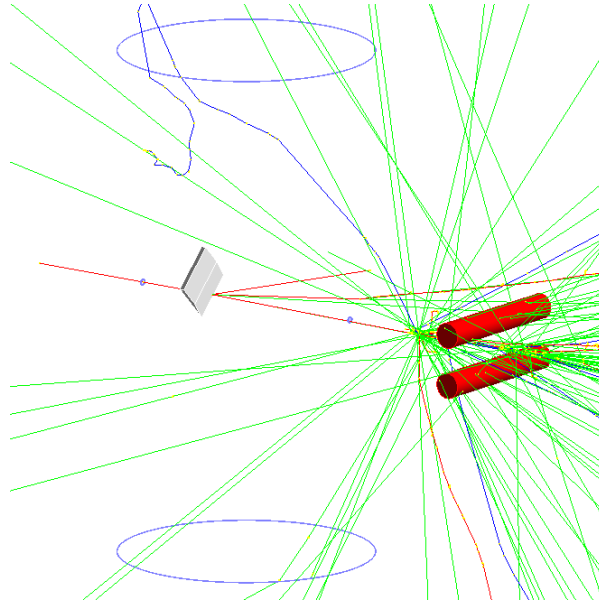
2.5 Generated data

The code provides the following output for each simulation:

- A description of the beam parameters for the simulation.
- A short description of the chosen scanner component and its geometry.
- In a filename of choice, a user specified textual description of the simulation.
- The main data file with one line for each simulation step, providing:
 - Scanner component position (x, y and z)
 - Energy deposited in the detectors
 - Particle tracks crossing through the detectors
 - Detector positions of both detectors (x, y and z)
 - Energy deposited in the scanner component
 - Number of secondary particles generated in the scanner component



(a) Simulation geometry.



(b) Simulation geometry including simulated event.

Figure 5: The geometry of the simulation. The red cylinders are the beam loss detectors. The grey scanning component is pictured in the middle. The circles on top and bottom indicate the target chamber, modeled as a simple stainless steel cylinder with vacuum interior. The smaller circles to the left and right of the scanner component show the interface between target chamber and beampipes. The beam enters the target chamber from the left.

2.6 Computing infrastructure

A typical simulation ran for roughly four hours on a modern quad-core machine. Many hundreds of these simulations were run during the course of the project, so a way was needed to dramatically cut down the simulation time.

The BIRD cluster [4] allowed – depending on the cluster load – running over a hundred independent simulations in parallel, providing more than a hundredfold reduction in time.

3 The Gauss-Newton NLLS fitting algorithm

As the eventual goal is to deduce the beam profile from measurements taken by the beam profiler, the algorithms performing this step needed to be tested on the simulated measurements obtained through the Geant4 code. The fitting process has previously been implemented in MATLAB and is able to use the simulation output as input.

A varying number of model parameters (beam width, beam intensity, knife edge geometry, wire scanner diameter, ...) need to be optimised according to the data and measurement errors. The Gauss-Newton algorithm is a good method to solve a non-linear least squares problem, and is used here to link the different models (knife edge, wire scanner, ...) to the simulations by minimising the χ^2 .

This algorithm and a set of fitting functions were implemented previously (see [5] and [6]).

4 Simulation results

4.1 Geometry

By visualising the simulated events, it became apparent that most of the scattering responsible for the radiation hitting the detectors is not caused directly by the scanning component itself, but happens instead when the narrow cone of scattered radiation hits the beam pipe wall. This is shown in figure 6.

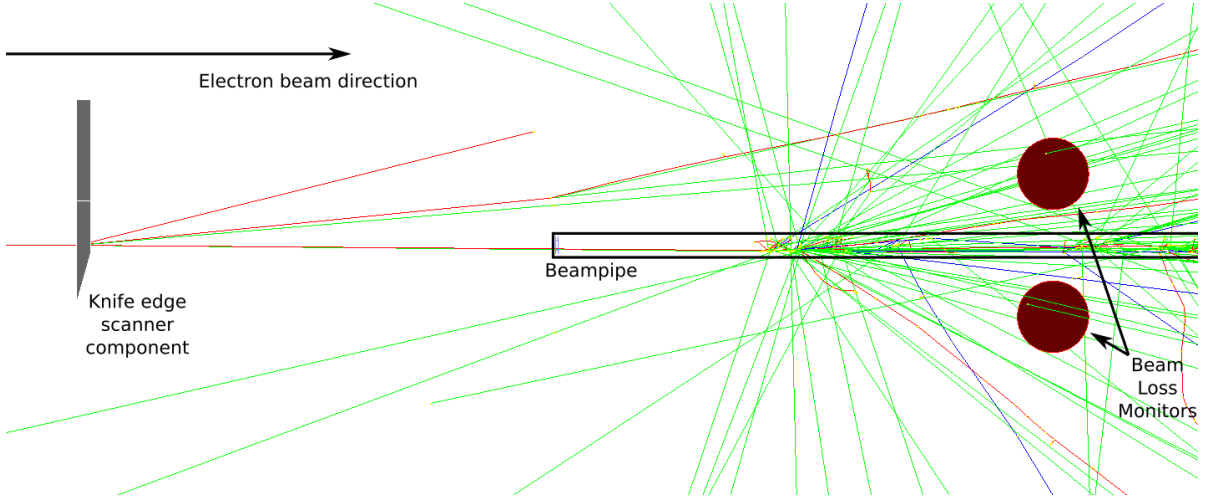


Figure 6: A very narrow cone of radiation leaves the scanner component on the left. After travelling some distance, these particles hit the beam pipe and cause additional heavy generation of scattered secondaries.

This result would follow from the fact that the scanning component usually has a thickness on the order of one millimeter. The incoming electrons thus move through relatively little material. In contrast, the particles hitting the long beampipe at a grazing angle travel through the beampipe material for centimeters to tens of centimeters, giving many more scattering targets.

This means that care must be taken not to oversimplify the geometry of the beam pipe and target chamber beyond the beam profiler interaction point, and it certainly can not be removed completely. Simulations which exclude the target chamber and beam pipe parts of the geometry generate almost no signal in the detectors.

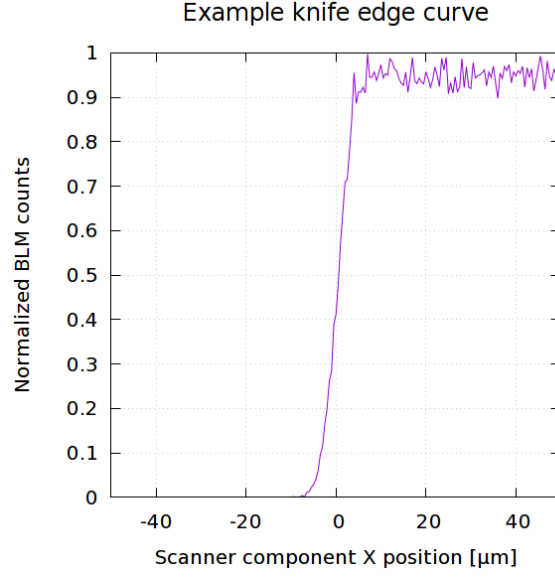


Figure 7: Simulation of a knife edge scanning component. The knife edge is 0.5 mm thick and has zero edge length.

4.1.1 Example curves for the different scanner components

When moving a knife edge component through the electron beam and measuring the number of tracks seen in the BLMs, a curve shown in figure 7 is observed. The slit scanner shows a very similar curve in figure 8. One can clearly see the two mirrored knife edge-like curves, centered around the beam position.

The wire scanner is modeled as a simple cylinder with small radius. Its example curve is shown in figure 9.

All example curves have been simulated in identical conditions, using a gaussian beam with a width of 4 μm. The BLM counts have been normalised to the maximum counts seen over the combined three curves. Figure 9 shows clearly that the wire scanner will give a signal almost two orders of magnitude lower than provided by the other two components in otherwise identical simulations.

The amount of material traversed plays a large role in the amount of scattering observed, so a wire diameter equal to the thickness of the other components (0.5 mm) would be needed to get a peak signal of comparable strength. Given beam widths on the order of micrometers, a wire scanner this big would no longer be a thin wire relative to the beam. It would in essence be a modified knife scanner with a geometry that would require a more challenging model to fit the measurements back to a beam profile.

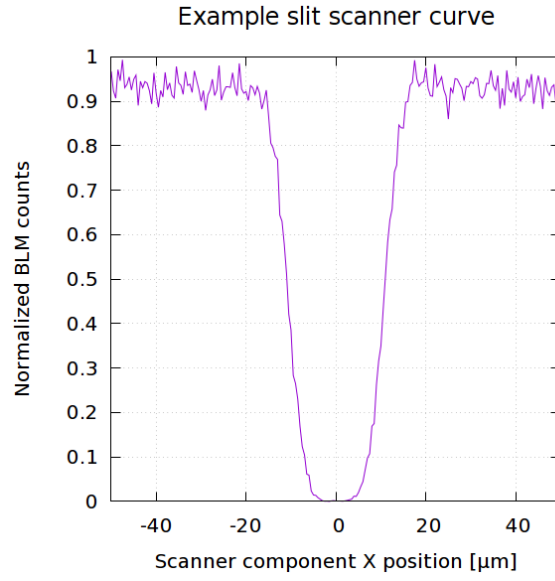


Figure 8: Simulation of a slit scanner component. The slit scanner is 0.5 mm thick, has zero edge length and a separation of $30\text{ }\mu\text{m}$.

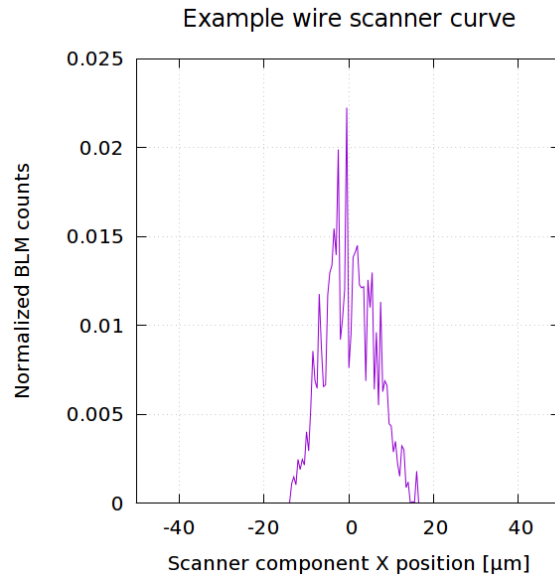


Figure 9: Simulation of a wire scanner component. The wire has a diameter of $30\text{ }\mu\text{m}$.

4.1.2 Different scanner component materials

Figure 10 shows curves produced using identical knife edges of various materials. It is clear that heavier materials such as metals and tungsten produce much more scattering than plastics.

As energy deposition in the scanner component is also simulated and stored, it can be of interest to look at this data for different materials. Table 2 shows the energy deposition per bunch for the simulated materials. Copper has a thermal conductivity roughly one order of magnitude better than that of stainless steel and twice as good as aluminum, yet the energy deposition into these three materials differs far less. By simulating realistic operational parameters, this energy deposition data could be used to select appropriate classes of materials depending on the heat load.

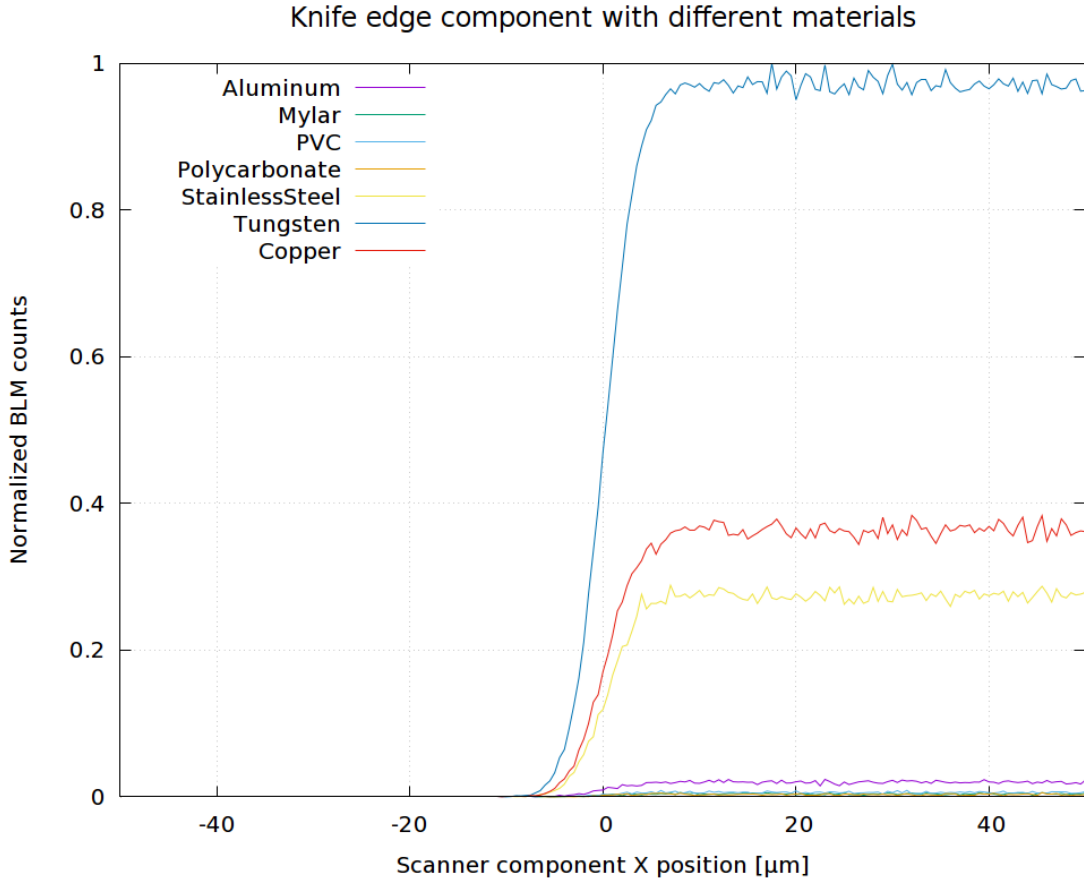


Figure 10: Knife edge simulations using different materials.

Material	eDep [J/bunch]
Tungsten	1.6×10^{-3}
Copper	7.4×10^{-4}
Stainless Steel	6.6×10^{-4}
Aluminum	2.3×10^{-4}
Mylar	1.3×10^{-4}
PVC	1.2×10^{-4}
Polycarbonate	1.1×10^{-4}

Table 2: Energy deposition per 0.39 nC electron bunch in a knife edge for different materials.

4.1.3 Behaviour with very thick scanning components

When simulating knife edge scanning components differing only in thickness, two specific effects were observed in the data. Such a set of simulations is plotted in figure 11.

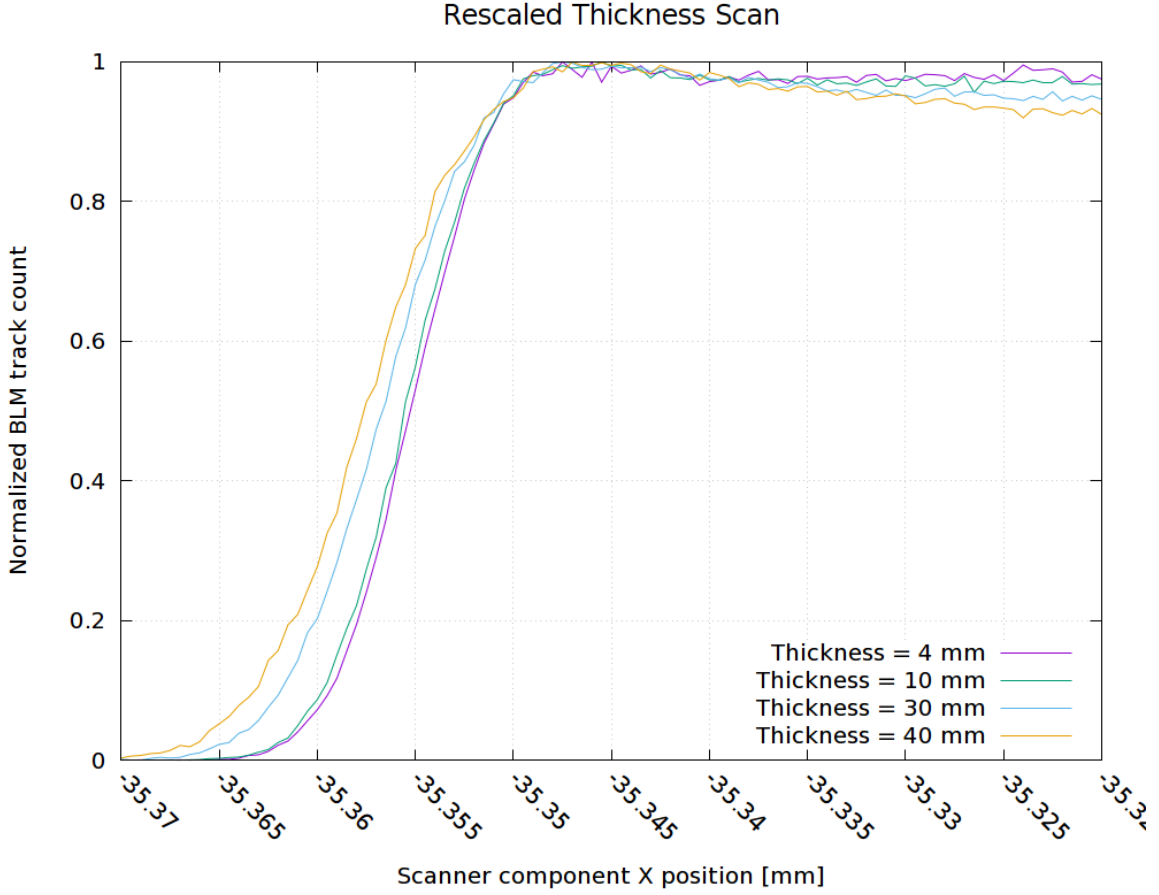


Figure 11: Simulations of a straight knife edge at various thickness values.

Firstly, the curves show an earlier increase in signal for an otherwise unchanged geometry. This can be explained by looking at the way the focused electron beam is simulated. The geometry of a focused gaussian beam is roughly simulated by using two gaussian profiles of different width, one at the interaction point, and one at the “magnet” point. Two locations are randomly generated in each of these planes according to the appropriate gaussian distributions, and a particle is shot along a trajectory connecting both points. This gives a beam that looks like a cone when viewed from the side. Beyond a certain thickness, the scanner component’s corners will start hitting the beam while its face is still some distance away from the interaction point, scattering part of the beam and generating a signal earlier than expected. This effect is sketched in figure 12. This essentially gives rise to an “effective edge length” different from zero, as the beam will see a different thickness of material for changing scanner component positions. When simulating using simple straight-line beams with constant gaussian profile, this effect disappears entirely. This supports the explanation put forth here.

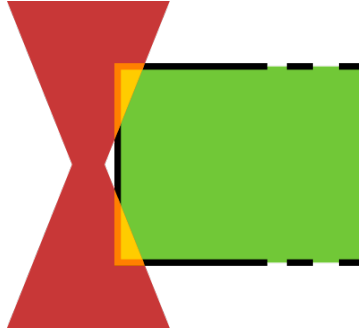


Figure 12: Simplified sketch showing the effective edge length mechanism at large scanner component thickness. The area of overlap between beam and scanner component gives rise to a signal before the component is expected to start intersecting the beam at the normal interaction point.

The second observed effect in figure 11 shows up on the right of the curves, where they no longer reach a stable plateau for thicker scanning components. Instead, a decrease is observed when the beam strikes the component progressively farther from the edge of the material. As absorption of radiation is a function of distance travelled inside a material, this phenomenon can likely be explained using the sketch in figure 13. The leftmost radiation cone produced by an electron striking the scanner component near its edge will have part of it escape the material along the edge. The cone on the right hits the material a good distance from the edge, thus the particles have to travel through the entire thickness of the material.

This effect should be present at any thickness, but will become more noticeable as the thickness of the scanner component approaches the order of the stopping distance of electrons in the material.

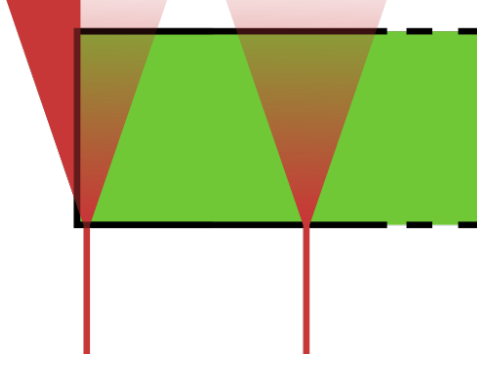


Figure 13: Sketch showing the difference in particle absorption when an electron strikes the scanning component at different distances from the edge. This sketch greatly simplifies the actual behaviour inside the material as the cone will widen due to additional scattering and slowing down of the particles. The general principle should however be valid.

Both these effects will place limits on the maximum thickness one can use in case of a knife edge scanner. The complexity of a model necessary to fit the measurements would otherwise become very complicated. The strongest limitation will likely come from the interplay between beam geometry and scanner component thickness, and not from absorption inside the material as a fairly large thickness is required before absorption effects become apparent. From this simulation, these non-linear effects are expected to appear for thicknesses beyond one or two millimeters, though simulations of different materials and more realistic beam profiles for FLASHForward may be desired in the future.

4.2 Fitting performance

Simulations of a knife edge component were performed for a threefold parameter scan. The edge length was varied linearly from 0 μm to 100 μm in steps of 10 μm , the beam width from 4 μm to 10 μm in steps of 1 μm and the knife edge thickness from 0.5 mm to 2 mm in steps of 0.5 mm. This provided 308 simulated measurements which could then be used to check the performance of the knife edge fitter for various geometries.

Fitting was done by keeping curve minimum, maximum (or intensity), beam position and beam width as free parameters. Knife edge length was assumed to be known exactly at this point, and was fixed to its actual simulated value.

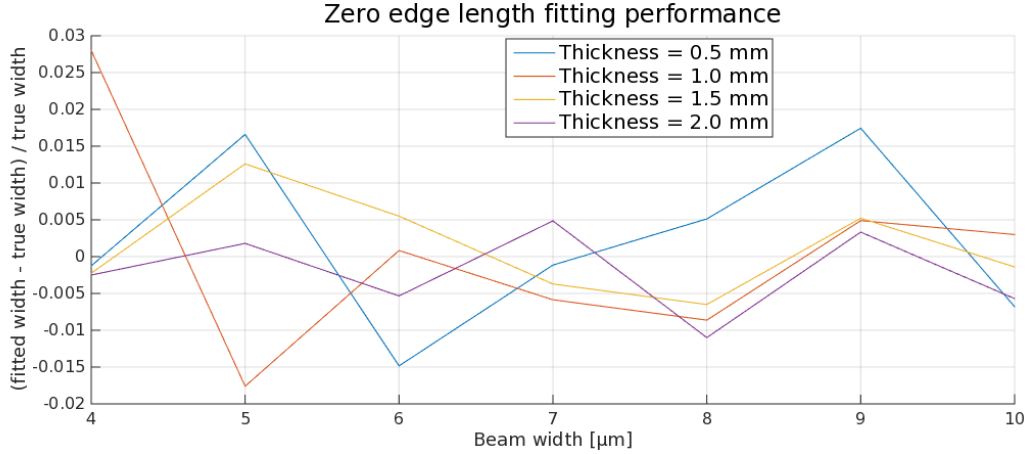


Figure 14: Fitting performance for zero edge length. The relative difference of the fitted value from the true value is shown for four different scanner thicknesses.

Performance in the case of zero edge length was found to be very good, with the beam width provided by the fitter consistently deviating by less than 3 % from the true value for all simulated beam widths and knife edge thicknesses, as shown in figure 14.

Knife edges with non-zero thickness posed more of a challenge to the fitting algorithms. The shape of the knife edge is convoluted with the gaussian profile of the beam, requiring the fitting functions to model this profile accurately.

When fitting the simulated data, it proved to be very difficult to match the shape of the model curves to the simulated curves. Only after using double the actual edge length did the fits start converging across the entire dataset. The residuals of measurements with non-zero edge length then still point to a poor reproduction of the actual curves, and the beam width is grossly overestimated, up to five times the true width in the worst case.

Selecting a poor performing measurement and manually adjusting all model parameters in an attempt to reproduce the measured curve prove to be difficult to impossible. The simulated data has a shape that is impossible to perfectly replicate using the current

model. The parameters are all sufficiently independent and do not allow producing identical curves by varying two parameters simultaneously. An example of one such result of manual fitting is shown in figure 15.

These results suggest that in the case of a non-zero edge length, the model does not describe the physical behaviour well enough to allow a high quality fit.

Figure 16 and figure 17 show the relative difference between fitted and true beam width in function of edge length and simulated beam width respectively. Increasing edge length progressively worsens estimation of beam width through fitting, while increasing the beam width improves the result.

Combining both the worsening performance with increased edge length, and the improving performance with increased beam width leads to the plot shown in figure 18. This figure shows a steadily worsening fit performance with increased edge length to beam width ratio.

In light of the good fitting performance in the zero edge length case, it would seem that the beam width is correctly modeled as a parameter. Attention should then be drawn to the modeling of the knife edge geometry as the actual error in the fitting functions.

A “zero edge length” geometry would actually be better defined as a small edge length to beam width ratio. Given figure 18, it then becomes possible to define a cutoff ratio for a given maximum error. For example, a maximum relative error of 5% would here correspond to a maximum edge length to beam width ratio of 1.5.

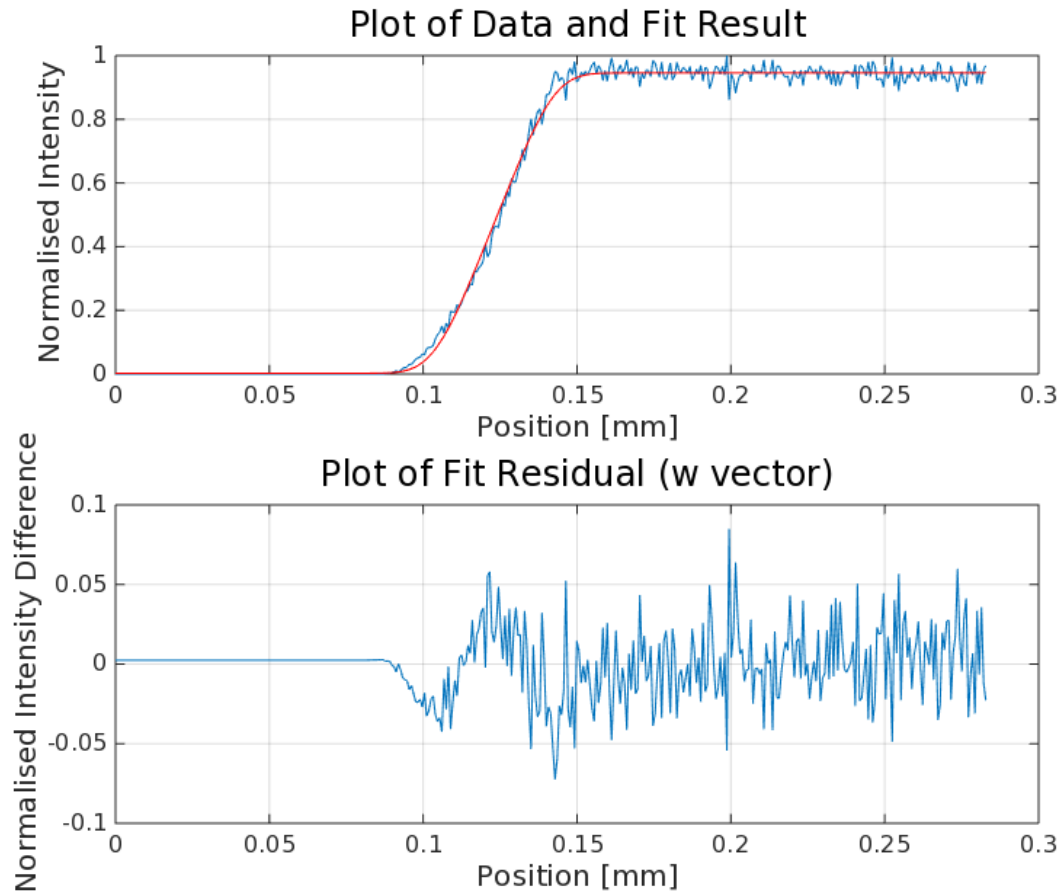


Figure 15: Manual fitting of the model function to a simulated measurement. The actual beam width is $4\mu\text{m}$, but has to be overestimated to $7.2\mu\text{m}$. The actual agreement between model curve and data curve is still quite poor, as can be seen from the residuals.

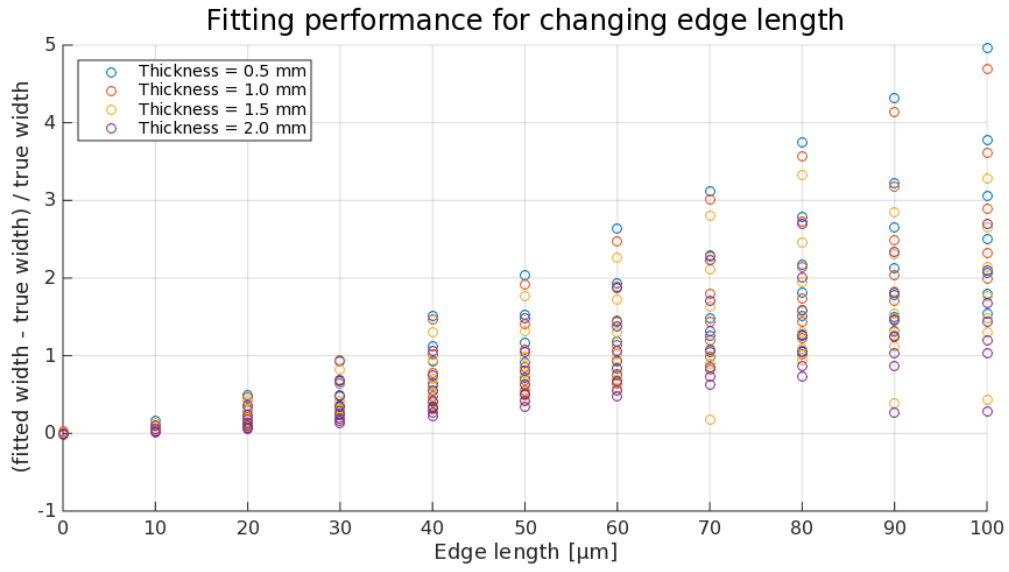


Figure 16: Fitting performance in function of changing edge length. A clear increase is visible, with beam width reproduction quickly getting worse with increasing edge length.

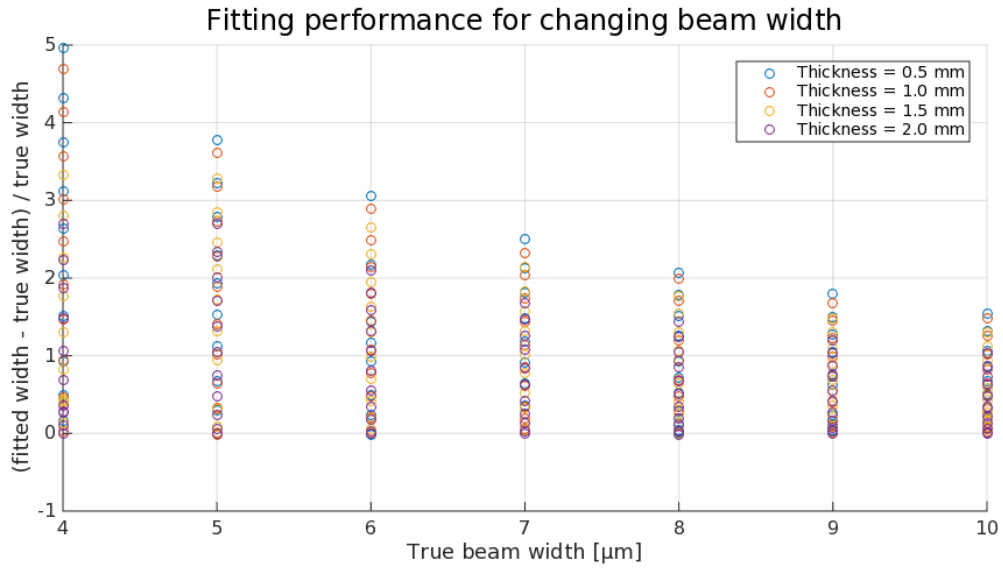


Figure 17: Fitting performance in function of changing beam width. Wider beams improve the fits significantly.

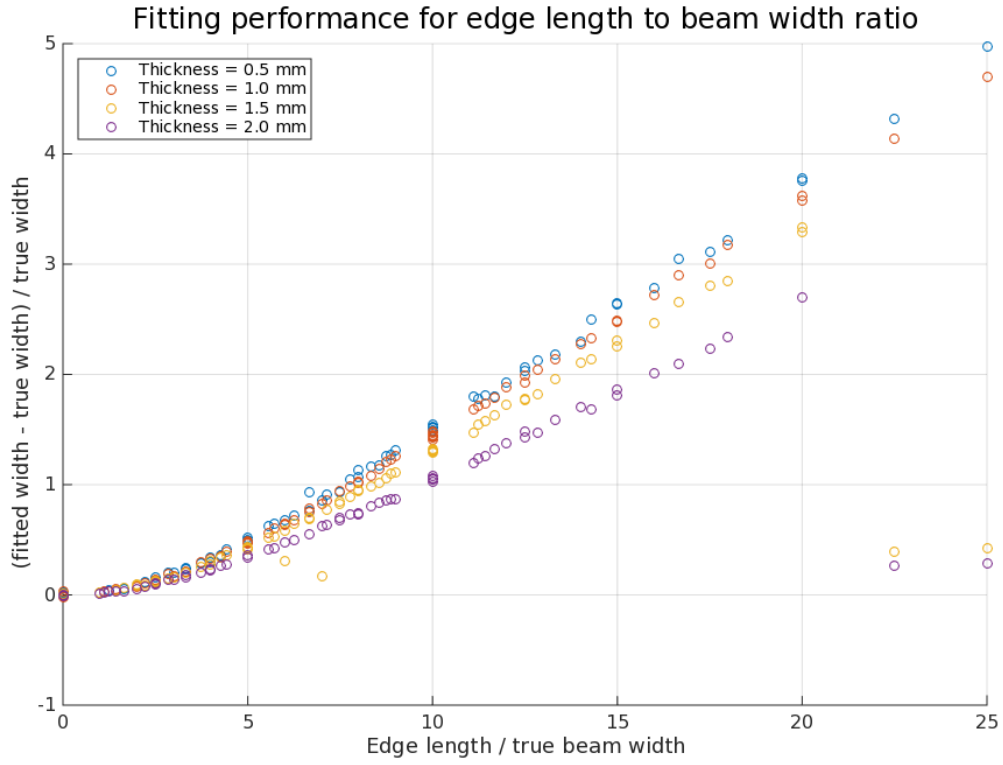


Figure 18: Fitting performance in function of edge length to beam width ratio. The extreme outliers are caused by non-converging fits.

5 Conclusions

5.1 Simulated measurement results

In section 4.1 it was shown that the radiation detected by the BLMs is not coming directly from the scanner component itself. The component sends out a narrow cone of secondary radiation, which then hits the beampipe at a grazing angle, producing lots of secondaries. This suggests that more realistic simulations need careful thought about what geometries are simulated downstream of the scanning component.

The interplay between beam geometry and scanner component geometry can lead to some fairly unexpected effects as shown in section 4.1.3. While these effects will not be very relevant for the FLASHForward experiment at the currently planned geometries and beam parameters it is still important to know of their existence, and to have insight into their causes.

From figure 9 it would seem necessary to handle a wire scanner in a slightly different way due to the lower signal it would generate. It might be necessary to employ a slower scanning speed, allowing longer integration times in the BLMs.

5.2 Fitting algorithms

The fitting model used for knife edge scanners appears to have some shortcomings whenever the edge length of a knife edge is non-zero. Performance for zero edge length knife edges appears to be adequate. Such a model gives the convolution of the scanning component geometry and the beam profile (assumed gaussian). It is likely that the slope of the knife edge's edge is not correctly modeled. One possible reason would be that the assumption of linearity in signal detected versus length of material traversed is incorrect.

5.3 Future improvements and goals

At the time of writing this report, the most obvious way forward is to investigate the cause of the poor fitting performance for knife edge scanners with non-zero edge length.

In parallel, one could also perform a similar set of simulations using a wire scanner, as its fitting model should be fairly independent from that of the knife edge and might yield better results. Similarly, the newly added slit scanner component has not been thoroughly tested yet.

The electron beam is currently simulated as an approximation of a focused gaussian beam. This could be improved by using more realistic electron position and momentum distributions obtained through the appropriate simulations. The Geant4 toolkit and the simulation code's current setup allow this to be implemented fairly trivially. Similarly,

the geometry of the target chamber only includes the most basic components. The actual structure will be far more complex. Verifying if any insights gained from the simple simulation still hold when a more complete environment is present might be worth the increased computational complexity.

Once the fitting models are producing desirable results and constraints on the scanner component geometry have been set, the facilities to simulate different materials and register energy deposition into the scanner could provide further refinement of acceptable solutions. Depending on material choice and volume, heat load could be an important limiter.

The current simulation code can take advantage of traditional shared memory multi-threading. To obtain better coverage and utilisation of the BIRD cluster (or similar facilities), it might be desirable to also implement a distributed memory parallelism scheme such as OpenMPI [7].

One other easy addition to the simulation would be controlled “fuzzing” of the scanner component position. A real beam will not be perfectly stable, and there will be some error on the actual positions achieved by the hexapod system. A simple way to add gaussian noise to the scanner component position after a movement could provide another way to test the robustness of the fitting algorithms.

6 Acknowledgements

I would like to thank the FLA group at the DESY Hamburg campus for their warm welcome and positive working environment; in particular my supervisor John Dale for his guidance and support during the course of this project. I would also like to thank the organisers of the summer student programme, as well as the DESY institute, for providing the support and infrastructure needed to enable my stay as a summer student.

References

- [1] A. Aschikhin et al. The FLASHForward Facility at DESY. 2015.
- [2] Geant4: A toolkit for the simulation of the passage of particles through matter. <https://geant4.web.cern.ch/geant4/>.
- [3] Martin Lüscher. A Portable high quality random number generator for lattice field theory simulations. *Comput. Phys. Commun.*, 79:100–110, 1994.
- [4] Grid Engine GE (BIRD/NAF2). <http://bird.desy.de/info/index.html.en>.
- [5] John Dale. Flashforward drive electron beam profiling with wires and knife edges in the experimental chamber. Technical report, DESY, 2014.
- [6] D.E. Wells and E.J. Krakiwsky. *The Method Of Least Squares*. Geodesy and Geomatics Engineering University Of New Brunswick, 1971.
- [7] Open MPI: Open Source High Performance Computing. <http://www.open-mpi.de/>.