IMPROVEMENTS OF THE TRACKING CODE ASTRA FOR DARK CURRENT STUDIES AT FLASH

L. Fröhlich*, University of Hamburg and DESY, Hamburg, Germany,
S. Meykopff, DESY, Hamburg, Germany

Abstract

At the Free Electron Laser in Hamburg (FLASH), the activation of components due to dark current emitted by the gun has become a serious problem. To improve the understanding of dark current transport in the linac, we have used extensive tracking simulations. To reduce the required amount of computing time, we have used a novel parallelized version of the Astra tracking code. We present the main parallelization scheme and investigations on the scalability of the code. We also introduce a new library for the description of complex three-dimensional aperture models. Finally, a brief overview of the simulation results and an evaluation of possible remedies of the activation problem are given.

INTRODUCTION

The FLASH linac accelerates electron bunches from a photoinjector to beam energies of up to 700 MeV [1]. The bunch length is reduced in two magnetic chicanes, BC2 and BC3, at energies of 127 MeV and 380 MeV. In front of the undulator, a four-stage collimation system consisting of two transverse and two energy collimators removes particles deviating excessively from the designated reference orbit and momentum (Fig. 1).

Because it is operated at a high electric field amplitude of 40–44 MV/m, the normal conducting RF gun is a major source of dark current. Measurements with a Faraday cup have shown an average current of 200–300 µA exiting the structure during the RF pulse. A substantial part of it is picked up by the first superconducting acceleration module ACC1 and transported through the linac, leading to increased losses and activation of components along the machine. This has become a problem especially near the first dipole magnets of bunch compressor BC2, where the narrow vacuum chamber intercepts a major fraction of the incoming dark current. Equivalent dose rates above 16 mSv/h due to activated material have been measured in this place, requiring an increased radiation protection effort [2].

To investigate possible remedies for this problem, the transport of dark current through the linac has been studied with particle tracking simulations. The space charge tracking code *Astra* [3] was the natural choice for this task because a good model of the FLASH injector was already available for it [4]. Originally covering only the initial 14 m of the machine, this model has been extended to include the complete beamline of about 250 m length. The scope of these simulations required substantial changes to the code in order to reduce execution times as well as the development of a new aperture modelling language capable of describing the complex geometries found in the accelerator.

* lars.froehlich@desy.de

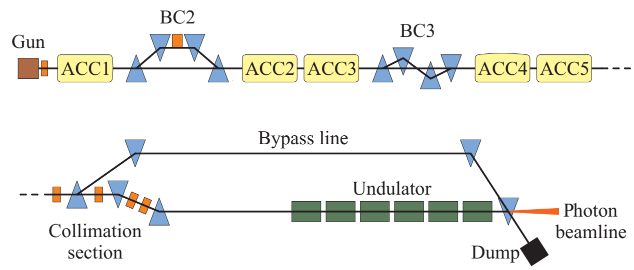

Figure 1: Schematic of the FLASH linac. The five acceleration modules ACC1-5 are shown in yellow, dipole magnets in blue, and collimators in orange color.

In this paper, we provide only a rough outline of simulation setup and results; details are found in [5]. Instead, we focus on the computational aspects of the work.

SIMULATION SETUP

The main simulation is divided into three steps:

1. Beam tracking: A bunch of 10^5 macroparticles is extracted from the cathode by photoemission and tracked to the dump. The step is iterated several times to match the beta function to the design optics.
2. Dark current tracking: 10^6 macroparticles are extracted from the cathode by field emission according to the Fowler-Nordheim model [6] and tracked to the dump. The phase space is saved in intervals of 8 cm, resulting in about 100 GB of data.
3. Aperture check: The saved phase space data are compared against a three-dimensional aperture model of the machine with a separate tool.

While space charge effects have to be included in the tracking of the beam in step 1, they are negligible in step 2 due to the low charge density of the dark current. The number of macroparticles required in step 2 is determined by the desired precision of the results. With a million macroparticles, relative dark current losses down to the level of 10^{-5} – 10^{-4} can be simulated.

We found the conventional single-processor version of *Astra* to be ill-suited for a work load of this magnitude; a single run would take several days to complete on the

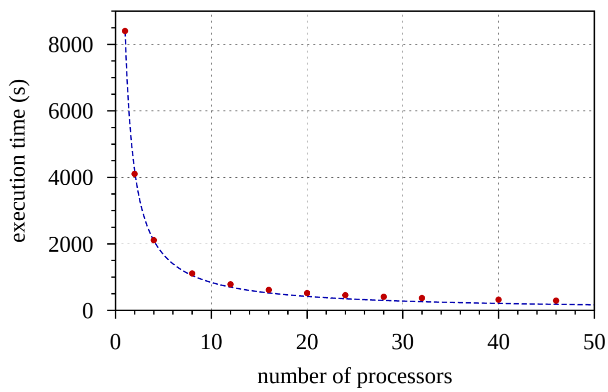

Figure 2: Parallel performance benchmark. The dotted line indicates the best theoretically possible execution time based on the measurement for a single processor, i.e. with assumed full scalability.

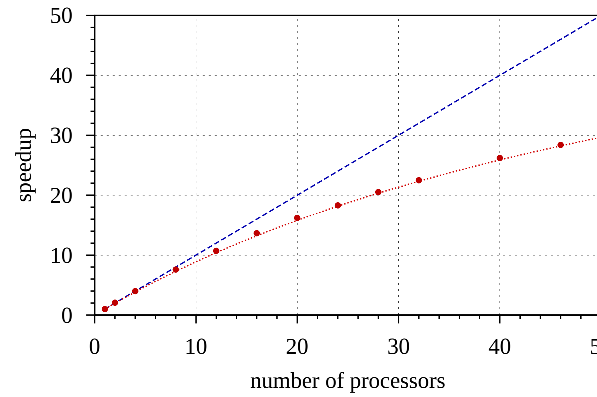

Figure 3: Speedup achieved by running Astra on multiple processors. The dashed line indicates the ideal case of full scalability, the dotted line corresponds to Amdahl's law with a sequential code fraction of $\alpha = 1.4\,\%$.

fastest available computer. Instead, we decided to make use of parallel computing with a new version of the code.

PARALLELIZATION OF THE ASTRA TRACKING CODE

Astra provides several particle-in-cell (PIC) algorithms to calculate the space charge field generated by the macroparticles. The most frequently used one employs a cylindrically symmetric grid in which the single cells are addressed by a radial and a longitudinal bin number, r and l. The code iterates over all N macroparticles to determine the charge present in each cell, Q_{rl}. Afterwards, the main tracking loop iterates over all particles again, for each one summing up the contributions of all partial charges Q_{rl} to the local space charge field as well as external electric and magnetic fields. The forces exerted on each particle are integrated with a fourth-order Runge-Kutta algorithm [7].

Many parallel PIC codes divide the computation work by distributing the grid cells among the available processors, which is especially advantageous in the case of grids with high resolution and with computationally demanding field calculations. However, Astra's cylindrically symmetric algorithm is frequently used with coarse grids (under 1000 cells), and the field calculation is comparatively simple due to the absence of complicated boundary conditions. Investigations with a code profiler also confirm that only a minor part of the execution time is spent in the space charge routines.

On the basis of these considerations, it was decided to achieve parallelization by distributing the macroparticles among the processors. In a run with N particles and P processors, the initial assignment would follow the pattern:

process 0:	particles	$0 \ldots N/P-1$
process 1:	particles	$N/P \ldots 2N/P-1$
$\ldots$		
process N:	particles	$(P-1)N/P \ldots N-1$

To exploit this form of data parallelism, we have chosen a classical "single program multiple data" (SPMD) approach; the existing serial source code has been extended by calls to the MPICH2 message passing interface library [8]. It is therefore possible to run the executable on a single processor as usual, or to spawn multiple processes within the framework of an existing MPICH2 installation.

Parallel Performance

Figure 2 shows the measured execution times on a cluster with 23 nodes of two CPUs each. Shared memory is used for communication between the twin CPUs, gigabit ethernet for communication between the nodes. The input file used for benchmarking contains the complete model of the FLASH linac, but simulates only the first 20 m of the machine with 10^5 macroparticles including space charge effects. There are no emittance calculations or file outputs except for the saving of one phase space file at the end of tracking.

The speedup ξ achieved by using P processors is defined as the ratio of parallel to sequential execution time, $\xi(P) = \Delta t(P)/\Delta t(1)$. Figure 3 shows that the measured speedup increasingly deviates from ideal behavior with increasing number of processors—the maximum achieved on the cluster is $\xi(46) = 28.4$. This behavior is described well by *Amdahl's law* [9] which states that if a program can be separated into a parallelizable part and a sequential part that takes a fraction α of the total execution time, the speedup by running it on P processors is given by

$$\xi(P) = \left(\alpha + \frac{1-\alpha}{P}\right)^{-1} .$$

By fitting to the data, we obtain a sequential fraction of $\alpha = 1.4\,\%$ for this benchmark case. The maximum possible speedup is then given by

$$\lim_{P\to\infty} \xi(P) \approx 71 .$$

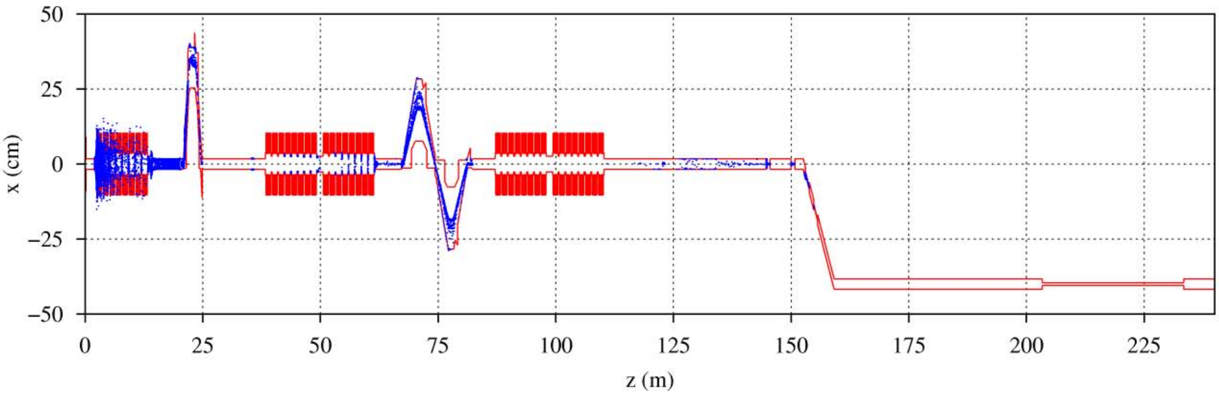

Figure 4: Horizontal section of the FLASH aperture model. The blue dots mark the positions of lost dark current particles. Losses in the gun region are not shown.

However, it is obvious that these figures depend strongly on the specific application—generally, a higher number of macroparticles will increase the speedup, and a higher number of sequential operations like file saving or emittance calculations will decrease it.

APERTURE MODELLING

To obtain reliable information about dark current losses from a tracking simulation, a good knowledge of the apertures along the machine is required. It is also necessary to have tools that can reproduce this geometry in an adequate way. Unfortunately, the built-in capabilities for modelling apertures in Astra and in most tracking codes are limited to basic shapes like circular openings or parallel plates.

To facilitate a coherent and exact description of aperture models, we have developed the portable *ApertureLib* library. It is written in C++ and allows integration to C and Fortran projects by a set of wrapper functions. The library allows to read aperture models from XML files like the following:

```
<aperture-list>
  <circle z="0" name="drift">
    <r>0.017</r>
  </circle>
  <include z="2.4" name="acc. module">
    <filename>acc_module.xml</filename>
  </include>
</aperture-list>
```

In this example, two aperture elements are specified along with their longitudinal positions in the machine. The first one describes a circular aperture with a radius of 1.7 cm, the second one includes another XML file that defines the geometry of cavities and vacuum chambers inside a cryomodule.

A wide variety of element types is available, allowing to model even complicated geometries:

- Primitive profiles (circle, ellipse, rectangle, . . .), rotatable to any direction in space
- Elements delimited by an arbitrary number of planes, specified by position and normal vectors
- Combination of other apertures by a logical AND or OR, e.g. to model alternative beamlines or new shapes like semicircles
- Inclusion of other XML files as “building blocks” for repetitive structures
- Inclusion of existing Astra aperture files
- Import of 2-dimensional CAD drawings in DXF format for complex planar geometries

In addition to the library, a set of tools has been created that allows validation and inspection of aperture models as well as generation of plots and checking of phase space files against the model.

With this toolkit, the geometry of the FLASH vacuum chambers shown in Fig. 4 was set up and compared against the phase space files from the previous Astra simulations.

SIMULATION RESULTS

The simulation shows that about 70 % of the emitted dark current are lost in the region from the gun cavity to the first cavity of module ACC1. Many particles arrive there on the wrong phase of the RF wave and are reflected or lost in the module. By scaling to the measured value of ∼250 µA at a Faraday cup behind the gun, we can conclude that roughly 100 µA of dark current are transported through ACC1 (Fig. 5).

After passing the acceleration module, the dark current has reached an energy of about 127 MeV. Since this is well above the giant nuclear resonance threshold, losses may lead to activation of accelerator beamline and components. As expected, the entrance of bunch compressor BC2 with its low vacuum chamber of only 8 mm height intercepts the biggest part of the remaining current, more than 50 µA. The simulation predicts other hot spots at the exit of ACC1, at the entrance and exit of BC3, and at the collimators, which is qualitatively confirmed by the measured radiation levels in these locations [2].

We can also estimate the remaining amount of dark cur-

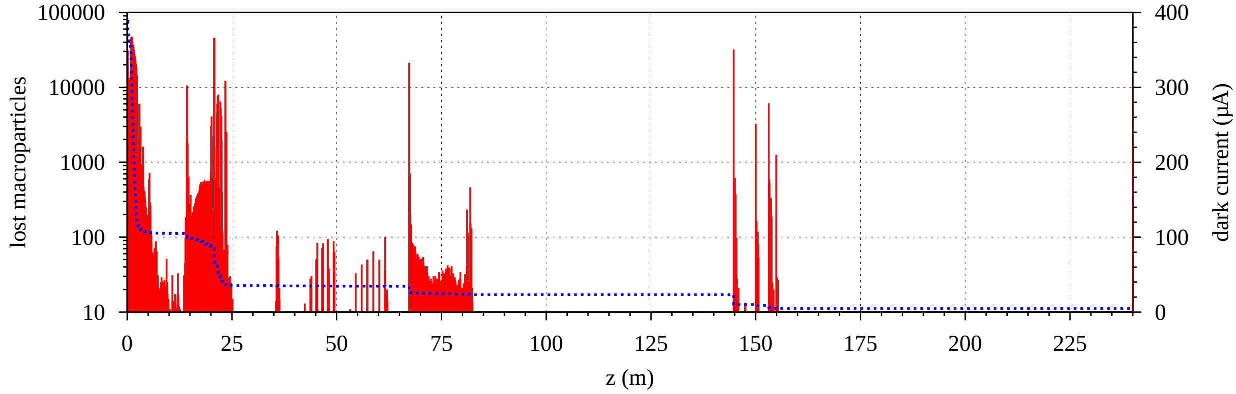

Figure 5: Simulated dark current losses along the FLASH linac. *Red (logarithmic scale):* lost number of macroparticles per phase space file. *Blue (linear scale):* remaining gun dark current.

rent traversing the FEL undulator to be about 5 µA. With the machine set to default optics, it oscillates between rms beam sizes of 250 µm and 1.5 mm in both planes while the beam is smaller than 100 µm. Therefore it is easy to lose some dark current in the undulator chamber by wrong beam steering. Calculations with the particle transport code Fluka [10, 11] have shown that a complete loss of the dark current would deposit a local dose rate of 30 Gy/minute in the permanent magnets.

To reduce the amount of gun dark current that is transported through module ACC1, it has been proposed to increase the drift length between gun and module by 30 cm [12], and to insert a collimator with a circular aperture of 8 mm diameter behind the gun. From simulations with our model, we expect a reduction of transported dark current by about two thirds.

CONCLUSION

The investigation of dark current transport and losses with detailed tracking simulations is computationally demanding. We found the parallelized version of the Astra code well suited for this task. It has been demonstrated that simulation times can be reduced drastically by distributing the work load over several processors. The scalability of the code makes it well suited for multicore-/multiprocessor machines or small clusters with up to 64 processors.

In order to set up a detailed three-dimensional aperture model of the FLASH linac, we have developed the portable C++ library *ApertureLib* and a set of necessary tools. We have used this toolkit to compare phase space files generated by the Astra runs against the aperture model. This allowed us to check variations of the model quickly without having to repeat the tracking. However, the library has also been integrated into Astra as an optional component and can be used directly during the tracking run.

The simulation has provided an insight to the beam dynamics of dark current transport in the linac and allowed to evaluate possible counter measures for the problem of beamline activation. Based on these data, we can predict a reduction of dark current losses at bunch compressor BC2 to about one third of their former amount by the proposed relocation of the gun and the use of a transverse collimator.

ACKNOWLEDGEMENTS

We highly appreciate K. Flöttmann's assistance with the original Astra source code. We also wish to thank him, B. Beutner, and J.-H. Han for many helpful discussions.

REFERENCES

[1] The TESLA Test Facility FEL team, "SASE FEL at the TESLA Facility, Phase 2", TESLA-FEL 2002-01, June 2002, Hamburg.

[2] A. Leuschner, DESY, private communication.

[3] K. Flöttmann, "Astra—A Space Charge Tracking Algorithm", available at http://www.desy.de/~mpyflo/.

[4] K. Flöttmann, E. Schneidmiller, Y. Kim, private communication.

[5] L. Fröhlich, "Dark Current Transport in the FLASH Linac", PAC'07, June 2007, Albuquerque, USA, pp. 956–958.

[6] R. H. Fowler, L. Nordheim, "Electron Emission in Intense Electric Fields", Proc. R. Soc. A, 119, May 1928, London, pp. 173–181.

[7] K. Atkinson, "An Introduction to Numerical Analysis", 2nd ed., Wiley, 1989.

[8] MPICH2, an implementation of the message-passing interface, http://www-unix.mcs.anl.gov/mpi/mpich2/.

[9] G. Amdahl, "Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities", Proc. AFIPS, (30), pp. 483–485, 1967.

[10] A. Ferrari, P. R. Sala, A. Fassò, et al., "Fluka: A Multi-Particle Transport Code", CERN-2005-00X, August 2005, Geneva.

[11] A. Fassò, A. Ferrari, S. Roesler, et al., "The Physics Models of FLUKA: Status and Recent Developments", CHEP'03, March 2003, La Jolla.

[12] J.-H. Han, K. Flöttmann, "Dark Current Collimation and Modified Gun Geometry for the European X-Ray FEL Project", FEL'06, August 2006, Berlin, pp. 579-582.