LONGITUDINAL SPACE CHARGE EFFECTS IN THE CLIC DRIVE BEAM

R.L. Lillestol*, S. Doebert, A. Latina and D. Schulte, CERN, Geneva, Switzerland
E. Adli and K.N. Sjobak, University of Oslo, Norway

Abstract

The CLIC main beam is accelerated by rf power generated from a high-intensity, low-energy electron drive beam. The accelerating fields are produced in Power Extraction and Transfer Structures, and are strongly dependent on the drive beam bunch distribution, as well as other parameters. We investigate how longitudinal space charge affects the bunch distribution and the corresponding power production, and discuss how the bunch length evolution can affect the main beam. We also describe the development of a Particle-in-Cell space charge solver which was used for the study.

INTRODUCTION

In the Compact Linear Collider (CLIC) scheme [1], the main beams are accelerated by rf fields extracted from two high-intensity, low-energy drive beams, before being brought into collision. These fields are extracted in Power Extraction and Transfer Structures (PETS), which are passive microwave structures with a fundamental mode at 12 GHz. In order to keep a high luminosity for the main beams, strict tolerances are necessary for several parts of the machine. One factor that affects the rf fields sent to the main beam is the drive beam bunch length. Assuming perfect bunch phase, the power produced in a single PETS scales as $P \propto F_b^2(\lambda)$, where $F_b(\lambda)$ is the single-bunch form factor. For a Gaussian bunch it can be written

$$F_b(\lambda) = F_b(\sigma_z) = \exp\left[-\frac{1}{2}(\sigma_z 2\pi f_b/c)^2\right], \quad (1)$$

where σ_z is the bunch length, f_b is the bunch frequency and c is the speed of light in vacuum.

The nominal drive beam bunch length is 1 mm. The bunch length tolerance can be calculated from a specification for the main beam of maximum 1 % luminosity loss, which requires an energy spread of $\frac{\Delta E}{E} < 7 \times 10^{-4}$. This leads to a tolerance for a coherent bunch length change of 1.1 %, and a tolerance for an incoherent bunch length change of 3.3 %.

Even though the CLIC drive beams are relativistic ($\gamma \in [470, 4700]$ in the decelerator), there has been some concern that longitudinal space charge in the high-intensity bunches may violate the bunch length tolerance. This has previously not been studied for the CLIC scheme. In this paper we describe the development of a simulation code and its preliminary results.

* reidar.lunde.lillestol@cern.ch

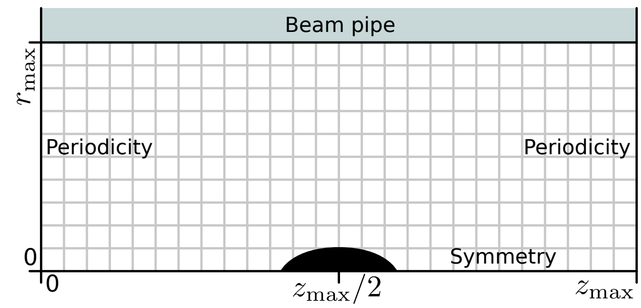

Figure 1: The system that is modeled. A bunch is placed on axis in an axisymmetric r-z grid. The maximum z value $z_{\max}$ is equal to the bunch separation.

MODEL

A common method for simulating space charge and similar effects in simulation codes is using the Particle-in-Cell framework (PIC). Because it is a well-known scheme that is intuitive and relatively straightforward to implement, PIC was chosen as the framework. It is foreseen to implement the space charge code into the existing tracking code Placet [2], and PIC should be compatible with the particle tracking in that code.

A full 3D PIC simulation requires a large amount of memory (e.g., fieldsolver matrices require $n_x^2 n_y^2 n_z^2$ elements, where $n_{x,y,z}$ is the number of grid points in each dimension) and extensive computation time. A 1D longitudinal code would have long-range Coulomb forces (the force is constant and does not fall off with distance) and overestimates the longitudinal force for particles outside the axis. A compromise was therefore chosen with a 2D code with cylindrical coordinates (r,z). This model – shown in Figure 1 – assumes axisymmetry, which is a simplification but applicable to the fairly round beams in the decelerator FODO lattice. Each code iteration follows a fairly standard PIC algorithm:

1. Define a grid for one bunch in the beam frame, based on its current energy in the lab frame. Because of length contraction the longitudinal coordinates scale with the Lorentz γ. A fixed number of grid points is used every iteration.

2. Assign charge with a linear, cloud-in-cell weighting to grid points near the particles.

3. Calculate charge densities on the grid points based on the volumes they represent.

ISBN 978-3-95450-122-9

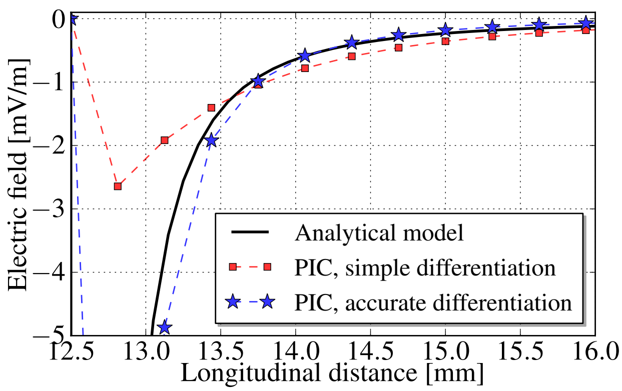

Figure 2: Longitudinal electric field from a point charge, where the PIC code is compared to an analytical model. For this grid size we clearly see the difference between the two types of numerical differentiation.

4. Solve Poisson's equation to obtain the potential ϕ everywhere on the grid.
5. Calculate electric fields from $\vec{E} = -\nabla\phi$. Magnetic fields are ignored since we solve in the beam frame.
6. Assign electric fields to each particle, in a linear cloud-in-cell manner.
7. Calculate particle kicks (i.e., update velocities and positions) from Lorentz' equation $\vec{F} = q\vec{E}$ and by using a leap-frog algorithm.

During the development it was found that normal 2-point numerical derivatives did not give accurate solutions for certain grid sizes and particle distributions. Therefore more accurate derivatives are used, and we write Poisson's equation for a free space grid point as

$$\begin{aligned}\nabla^2\phi\big|_{i,j} \approx\ & \frac{1}{r_i}\frac{\phi_{i-2,j} - 8\phi_{i-1,j} + 8\phi_{i+1,j} - \phi_{i+2,j}}{12\Delta_r} \\ &+ \frac{-\phi_{i-2,j} + 16\phi_{i-1,j} - 30\phi_{i,j} + 16\phi_{i+1,j} - \phi_{i+2,j}}{12\Delta_r^2} \\ &+ \frac{-\phi_{i,j-2} + 16\phi_{i,j-1} - 30\phi_{i,j} + 16\phi_{i,j+1} - \phi_{i,j+2}}{12\Delta_z^2} \\ =\ & -\frac{\rho_{i,j}}{\epsilon_0},\end{aligned} \tag{2}$$

where $\phi_{i,j}$ is the potential and $\rho_{i,j}$ is the charge density at coordinates $(r = r_i, z = z_j)$, ϵ_0 is the electric permittivity in vacuum and $\Delta_{r,z}$ is the grid length in each dimension. Figure 2 shows a comparison between the two numerical differentiation schemes. In this example a coarse grid in the r dimension shows that the more accurate differentiation clearly outperforms the simpler version.

A sketch of the system that is simulated is shown in Figure 1. The bunch is situated at $r = 0$, and this axis has a Neumann boundary condition with $\frac{\partial}{\partial r}\phi = 0$ to represent symmetry. Longitudinally, the grid spans from half the distance to the previous bunch, to half the distance to the next bunch. To eliminate any effects of bunch phase we use periodic boundary conditions longitudinally. A bunch offset from the longitudinal center would therefore not have an impact on the simulation. At the final boundary, which is the beam pipe, the potential is fixed to $\phi = 0$.

IMPLEMENTATION

The code has been written in Octave. The tracking code Placet [2] has an interface to Octave, and at a later stage this will make an implementation of the space charge code into Placet easier. Octave also offers a good environment for development.

Matrices used for the fieldsolver are stored in sparse format, which offers very good compression since the matrices have at most 9 diagonals. In this way we can use larger grids without running into memory problems. We solve for the potential using incomplete LU factorization, which was found to be the fastest method for large grids, and which operates on sparse matrices.

Several steps have been made to optimize the code for speed. However, since the energy changes along the decelerator, the grid (in the beam frame) and the field solver matrices must be remade every timestep, including the incomplete LU factorization. To increase the speed of this bottleneck, the chosen solution is to store normalized sparse matrices, which are scaled each iteration. Still, each iteration takes around 1.5 s for the conditions normally used (51×61 grid size and 10'000 macro particles).

BENCHMARKING

The code results have been compared to a number of analytical models. Mainly we have compared the electric fields, which agree reasonably well. One example is shown in Figure 2, which shows the longitudinal electric field close to a point particle. Another example is shown in Figure 3, which shows the longitudinal electric field from a bunch with a Gaussian longitudinal distribution with all particles placed on the axis transversally. In this particular analytical model we integrated a Gaussian distribution numerically to obtain the field everywhere on the axis.

For a small number of grid points in the r dimension and certain distributions like the Gaussian, the code has been found to underestimate the longitudinal field. To get a sufficiently accurate solution, we therefore need at least 60 points for a converging solution. This effect can be seen in Figure 3, where the field from the 41×51 grid ($r \times z$, in red) is underestimated (even though it converges in the Gaussian tails).

FIRST RESULTS

Some preliminary results have been obtained. However, since the code is not yet implemented in Placet, quadrupole focusing and beam dynamics other than space charge is not included. Therefore the simulations are quite simplified, and should be seen as the first stage of a study.

One simulation treated the decelerator as a long drift space, where the beam energy changed stepwise from 2.4 to 0.24 GeV from the 1492 PETS in the lattice. We used

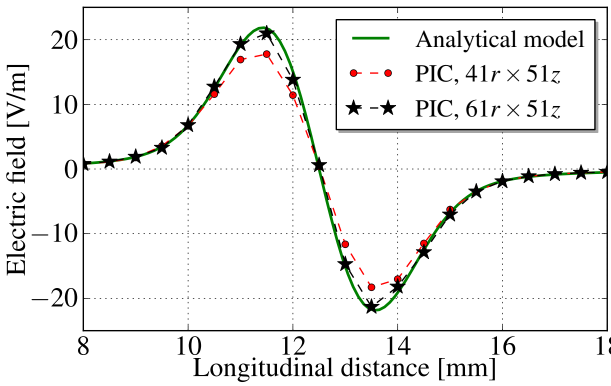

Figure 3: Longitudinal electric field from a bunch that is Gaussian distributed longitudinally and placed on axis transversally. 10'000 macroparticles were used in the PIC code and the result is compared to an analytical model. It is necessary to use at least 60 grid points radially to ensure convergence.

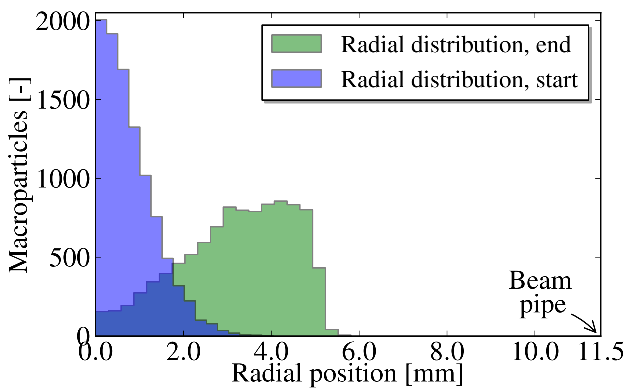

Figure 4: Radial bunch distribution before and after going through the CLIC decelerator (without quadrupole focusing).

nominal starting parameters, a 61×51 grid and 10'000 macro particles. The particles also started out stationary, which means zero emittance and energy spread. With these conditions the bunch length barely increased, with a factor $(1 + 6 \times 10^{-7})$. This is well within the coherent bunch length tolerance of 1.1 %, and has a negligible effect on the PETS power production. The space charge effect is more noticable transversally, where the beam occupies half the aperture as shown in Figure 4. However, when quadrupole focusing is included later we expect it to correct the transverse space charge defocusing, and also increase the longitudinal effect.

A similar simulation was performed for the Test Beam Line [3] in the CLIC Test Facility 3, which is a prototype CLIC decelerator. In this simulation the beam was decelerated from 120 to 60 MeV with 16 PETS, and here we also got a negligible bunch lengthening of a factor $(1 + 5 \times 10^{-5})$.

FUTURE PROSPECTS

It is foreseen to implement the space charge code into the Placet code. One approach for how it could work inside Placet, is that each time the tracker encounters a lattice element, it should estimate a number of timesteps that is needed for numerical convergence. This is based on the length of the current lattice element and the stability criterion [4]

$$\delta_t \leq \frac{1}{c} \frac{1}{\sqrt{\frac{1}{\Delta_r^2} + \frac{1}{\Delta_z^2}}}, \tag{3}$$

where δ_t is the timestep. The calculated positions and velocities at the end of the lattice element can then be interleaved with the other effects calculated in Placet.

After being merged with Placet, it will be possible to benchmark the code with more advanced models using other existing codes.

Adding the space charge effect to Placet will force it to run considerably slower, particularly because the field solver matrices are updated every iteration. Therefore it will likely be an effect that can be used by demand.

CONCLUSION

A PIC code for space charge has successfully been developed in Octave, and benchmarked with some analytical models. Simulations have been performed for the CLIC decelerator without quadrupole focusing, and showed a negligible bunch lengthening. In the future it is foreseen to implement the space charge code in Placet, and this will allow simulations closer to the real case. In particular the goal is to verify that longitudinal space charge is not an issue for PETS power production and for luminosity loss in CLIC.

ACKNOLEDGEMENTS

The authors want to thank Steven M. Lund of the Lawrence Berkeley and Lawrence Livermore National Laboratories, for helpful discussions and tips during the code development.

REFERENCES

[1] M. Aicheler et al., *A Multi-TeV Linear Collider Based on CLIC Technology: CLIC Conceptual Design Report*, Geneva, 2012.

[2] `https://savannah.cern.ch/projects/placet/`.

[3] R. Lillestol et al., "Experimental Results from the Test Beam Line in the CLIC Test Facility 3", these proceedings.

[4] A. Taflove and M.E. Brodwin, "Numerical Solution of Steady-State Electromagnetic Scattering Problems Using the Time-Dependent Maxwell's Equations", IEEE Trans. Microwave Theory & Techniques MTT-23, 623, 1975.

ISBN 978-3-95450-122-9