

# PLACET: A PROGRAM TO SIMULATE DRIVE BEAMS

D. Schulte, CERN, 1211 Geneva, Switzerland

## Abstract

In the Compact Linear Collider (CLIC), the 30GHz RF-power used to accelerate the main beam will be obtained by decelerating a drive beam. The stability of this high-intensity low-energy beam is crucial, especially in the decelerator where the energy spread in the bunch train is large and perturbing fields are strong. A program is presented that simulates the drive beam deceleration. It takes the high group velocities of longitudinal and transverse wakefields into account, together with damping of the transverse mode. The effects of a possible non-uniformity of the decelerating field can also be studied. In addition, different beam-based alignment schemes can be simulated.

## 1 INTRODUCTION

The program PLACET (Program for Linear Accelerator Correction Efficiency Tests) was developed to simulate the different beam lines in CLIC [1]. It allows to track the beam and to perform beam-based alignment in the presence of transverse and longitudinal effects, such as wakefields, phase errors, misalignment of beam-line elements, ground motion, the earth's magnetic field and beam jitter. One can also simulate the results of different beam-based alignment schemes. The program is still under development, which results in some changes to the interface from time to time. Here, the part necessary to simulate the decelerator will be presented; it is relatively stable, and to the knowledge of the author, no equivalent programs exist.

The program is written in C and provides an extended version of the tcl-shell wish [2]. This allows the interface to have the full flexibility of a macro language without having to invent yet another one. It also makes access to graphics and to graphical interfaces very easy. A more complete documentation of the program exists [3]. Here, only a few basics will be shown as examples.

Tracking 1000 bunches with 21 slices each through 334 PETS (power extraction and transfer structures) with five steps per structure takes about 400s on a 300 MHz-Pentium. The program requires less than 10 MB in this case.

## 2 STRUCTURE MODEL

The decelerator structures are assumed to have a single longitudinal and a single transverse mode [4, 5]. Each mode is described by a loss factor  $k_0$ , a wavelength  $\lambda$  and a group velocity  $\beta$ , and for the transverse mode also by a damping factor  $Q$ . The wakefield produced by a driving particle at

position  $s_0$  is seen by a witness particle flying at a distance  $z$  behind, to be at the position  $s = s_0 + z\beta/(1 - \beta)$ . In the simulation, a particle is tracked through the structure in several steps. For each step, the average field is calculated and used for the tracking. The total deceleration is exact, one can thus vary the number of steps to verify convergence of the transverse effects without being confused by longitudinal effects.

For the transverse field, also the attenuation due to the damping is taken into account. Once the particle has caught up with the emitted wakefield  $w_0$ , its strength  $W(z)$  is calculated as

$$W(z) = W_0 \exp\left(\frac{1}{1 - \beta_{\perp}} \frac{\pi z}{\lambda Q}\right). \quad (1)$$

If the structures have no rotational symmetry, the longitudinal field varies as a function of the radius  $r$  and the azimuthal angle  $\phi$ . If the structure has  $m_s$ -fold symmetry, the field can be expressed in a multi-pole expansion:

$$E_{\parallel}(z, r, \phi) = \sum_{i=0}^{\infty} 2k_i \frac{r^{m_s i}}{a^{m_s i}} \cos(2\pi z/\lambda) \cos(m_s i \phi). \quad (2)$$

Here,  $a$  is the structure radius and  $z$  is the position of the particle within the beam. In one of the symmetry planes one has  $\phi = 0$ . The non-uniformity of the longitudinal field also causes a transverse kick [6]. Both effects are included in the program, where  $k_0$  is defined via the longitudinal  $R'/Q$  and an arbitrary number of coefficients  $k_i/k_0$  can be specified.

## 3 COMMAND FILES

In order to make the simulations one has first to define the structure properties and to set up the lattice, then to create the beams.

### 3.1 Defining the Structure Properties

All structures in the beam line are assumed to have the same properties, which are defined with a simple command. In the following examples, the six-waveguide structure from reference [6] is used with an updated transverse wakefield and damping factor [4, 5]. The wavelengths are  $\lambda_{\parallel} = \lambda_{\perp} = 10$  mm and the group velocities are  $\beta_{\parallel} = 0.59$  and  $\beta_{\perp} = 0.7$ . The longitudinal loss factor is  $R'/Q = 25.4 \Omega/\text{m}$  (circuit convention). The transverse wakefield has an amplitude defined by  $W_0 = 125 \text{ V}/(\text{pCm}^2)$  and a damping factor of  $Q = 70$  (more conservative than in

the reference). The structure radius is  $a_0 = 15$  mm, the symmetry  $m_s = 6$  and the coefficients in equation 2 are  $k_1/k_0 = 1.6$  and  $k_2/k_0 = 0.65$ .

The structure length and azimuthal angle can be specified when the structure is placed into the beam line.

### 3.2 Defining the Lattice

All elements are placed on girders. The command `Girder` appends an empty girder to the beamline. Drifts, BPMs (beam position monitors), structures, quadrupoles, dipoles etc. can be positioned on this last girder using the commands `Drift`, `Bpm`, `DecCavity`, `Quadrupole` and `Dipole`, respectively. The example below creates the decelerator lattice, except for the initial and final matching.

```
for {set i 1} {$i < 167} {incr i} {
  Girder
  Drift -l 0.0925
  Bpm -l 0.03
  Quadrupole -l 0.14 -s [expr 0.87*$e]
  Drift -l 0.0925
  DecCavity -l 1.3 -phi 0
  Drift -l 0.0925
  Bpm -l 0.03
  Quadrupole -l 0.14 \
    -s [expr -0.87*($e-0.5*$de)]
  Drift -l 0.0925
  DecCavity -l 1.3 -phi 30
  set e [expr $e - $de]
}
```

This lattice consists of 167 FODO-cells with a phase advance of about  $90^\circ$  per cell. Each cell also contains two 1.3 m long PETS, each of which feeds three main linac structures. The parameter `-l` defines the length of each element in m. The variable `e` contains the initial beam energy and `de` the maximum deceleration per FODO-cell. The quadrupole strengths are scaled in order to lead to the same focal length for the lowest energy in the beam, all along the beam line. Finally the beam line is fixed using the command `SetBeamline`.

### 3.3 Defining the Beams

The beams are defined using the command `DriveBeam`. A beam consists of bunches, with each bunch cut longitudinally into slices. Each slice is represented by at least one macro particle. To simulate an initial energy spread of the beam, several macro particles are used per slice with the same position in  $z$  but different energy and charge. In case of a linear lattice, the tracking is done in second order, so not only the centres of the macro particles but also the corresponding beam matrices are tracked.

If a field non-uniformity should be taken into account, the beam is represented by macro particles that are distributed over the phase space ellipse. Since the lattice is not linear, only first-order tracking is used.

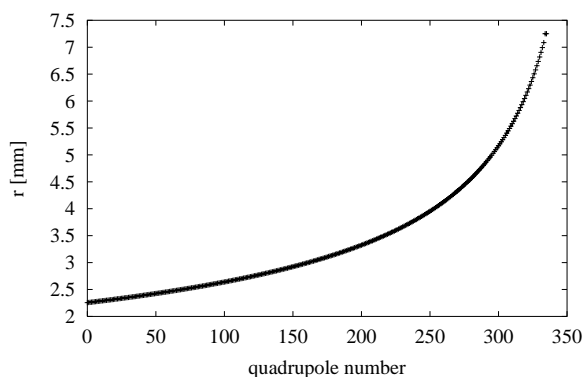


Figure 1: The 4-sigma beam envelope along the decelerator. A perfect beam line and an on-axis beam are assumed.

Beam loading compensation in the main linac requires that in the first part of the pulse, the bunch charge [7] or the longitudinal bunch position [8, 9] are varied. The command `DriveBeam` therefore accepts a list which defines, for each bunch: in which bucket it is, its charge and the longitudinal position with respect to the centre of its bucket.

In the following simulation the beam has no ramp and consists of one bunch with  $7.9 \times 10^{10}$  particles every 2 cm. The initial energy is  $E_0 = 1.7$  GeV, the bunch length  $\sigma_z = 400 \mu\text{m}$  and the emittances are  $\gamma\epsilon_x = \gamma\epsilon_y = 150 \mu\text{m}$ .

## 4 EXAMPLES

After the lattice and the beams are defined, the actual simulation can start. As simple examples, the calculation of the beam envelope, the effect of transverse beam jitter, and one beam-based alignment method are presented, before giving one graphics example.

### 4.1 Beam Envelope

To find the  $4\sigma$ -envelope (Fig. 1) of a beam one can use the command `TestNoCorrection`:

```
TestNoCorrection -beam beam0 \
  -survey Zero -emitt_file emitt.dat
```

Here, `beam0` is the name of the beam to be tracked and `Zero` stands for a survey without any misalignments. The results will be stored in the file `emitt.dat`. In the program the envelope contains all particles with the following conditions at the start of the beamline:

$$4^2 \geq \left(\frac{x}{\sigma_x}\right)^2 + \left(\frac{x'}{\sigma_{x'}}\right)^2 \quad \text{and} \quad 4^2 \geq \left(\frac{y}{\sigma_y}\right)^2 + \left(\frac{y'}{\sigma_{y'}}\right)^2$$

### 4.2 Beam Jitter

In order to simulate beam jitter, the beam is put in an offset position in phase space at the start of the beamline. It is then tracked as before. The simulation of a beam with an initial offset  $\Delta y = \sigma_y$  shows only a moderate increase of the maximum offset of any slice with respect to the case when transverse wakefields are neglected, see Fig. 2.

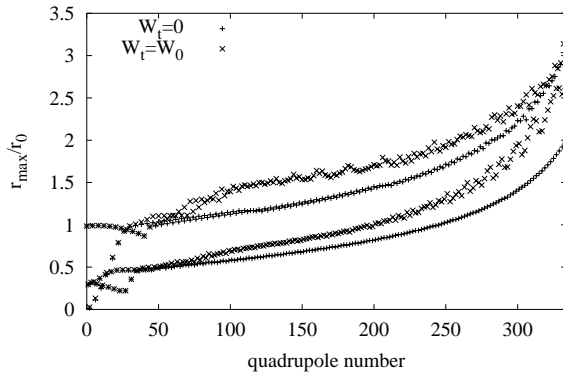


Figure 2: Amplification of an initial offset along the decelerator with and without transverse wakefields. The two lines for each case correspond to focusing and defocusing quadrupoles.

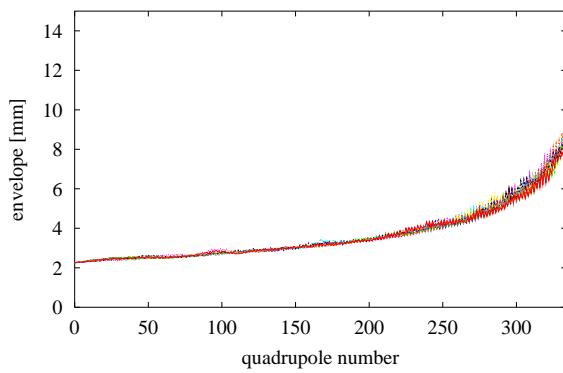


Figure 3: Envelopes of 4-sigma beams in a decelerator after beam-based alignment. Ten different machines were simulated.

### 4.3 Beam-Based Alignment

The effect of the initial misalignment and the beam-based correction can also be simulated. The initial alignment errors are set with the command `SurveyErrorSet` to be sampled from a Gaussian with a sigma of  $100 \mu\text{m}$ . First, the BPMs are aligned with the ballistic method [10] using the main beam after the damping ring at an energy  $E = 1.98 \text{ GeV}$ . Then the quadrupoles are aligned using the full drive beam and a one-to-one correction. Both steps can be performed using the single command `TestBallisticCorrection` which is similar to `TestNoCorrection`, except that more parameters can be specified. The resulting envelopes of ten different machines all stay well within the acceptance, as can be seen in Fig. 3.

### 4.4 Graphics

The tk toolkit allows simple implementation of graphics and graphical interfaces. Figure 4 shows an example window in which the bunches can be viewed while passing the decelerator. By clicking on the appropriate button one can stop or restart the tracking. With the slider, the window can be moved along the bunch train. Clicking on a bunch slice prints its transverse position.

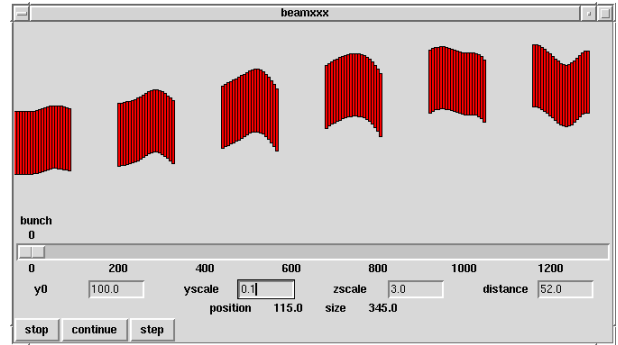


Figure 4: The window that shows the bunch train during its passage through the decelerator.

With another command the beamline can be viewed after beam-based alignment. This allows one to visualise the effect of the algorithms.

It is also possible to store the position of beamline elements or the beam in files that can be processed with any plotting program.

## 5 CONCLUSION

A program is presented that allows one to simulate efficiently the drive beam decelerator of CLIC. It can easily be applied to other frequencies and parameters. While the development of the program is continuing, it is available from the author. Simulations of the six-waveguide structure, using updated parameters, show that beam jitter will not be enhanced very much by the wakefields and that beam-based alignment works very well.

## REFERENCES

- [1] J.-P. Delahaye. *CERN/PS 99-005 (LP)* (1999).
- [2] J. K. Ousterhout. "Tcl and the Tk Toolkit." (1994).
- [3] D. Schulte. "The Tracking Code PLACET". Draft at <http://home.cern.ch/dschulte/placet.html>.
- [4] I. Syratchev and M. Luong. These proceedings.
- [5] G. Carron, A. Millich, L. Thorndahl. *CERN/PS 98-012 (LP)* (1998).
- [6] A. Millich, A. Riche and D. Schulte. *PAC 1999 and CERN/PS 99-028 (LP)* (1999).
- [7] L. Thorndahl. *CLIC-Note 291* (1995).
- [8] R. Corsini, J.-P. Delahaye and I. Syratchev. *CLIC-Note 408* (1999).
- [9] D. Schulte. *CLIC-Note 434* (2000).
- [10] T. O. Raubenheimer and D. Schulte. *PAC 1999 and CERN/PS 99-024 (LP)* (1999).