

# APPLICATIONS OF THE GENERAL PARTICLE TRACER CODE

S.B. van der Geer, M.J. de Loos, Pulsar Physics,  
Flamingostraat 24, 3582 SX Utrecht, The Netherlands

## Abstract

The General Particle Tracer (GPT) code provides a new 3D simulation package to study charged particle dynamics in electromagnetic fields. Because of its modern implementation, GPT can be conveniently customized without compromising its ease of use, accuracy or simulation speed.

The most common use of GPT is accelerator and beam line design, especially the calculation of non-linear 3D space-charge effects. A typical example is the study of the effect of different bunch charges in a bend system.

One of the advanced GPT features is the possibility to solve additional differential equations while tracing particles. Using this mechanism, the generated EM wave power spectra in an undulator and the effect of beam-loading in a traveling wave linac are calculated self-consistently with the particle trajectories.

GPT is currently being applied to the design of the energy recovery system of the 2 MeV, 12 A Free Electron Maser under construction at FOM-Rijnhuizen. The required 99.8 % beam recovery and the initial energy spread of 300 keV ensure this to be a challenging GPT project.

## 1 INTRODUCTION

Particle tracking is a very powerful tool in the design of accelerators or beam lines. Computers however used to be too slow to trace the amount of particles needed to obtain good statistics in a reasonable amount of computing time. Therefore the particle tracking method was often abandoned and especially matrix/optical methods became popular. These alternative methods however have their limitations. Especially when space-charge effects are important, they are inaccurate or can not be used at all.

Because of enormous advances in computer technology, it is currently possible to track thousands of particles through complex electromagnetic fields taking all 3D effects and space-charge forces into account. This offers much more accurate and reliable results than matrix codes. The General Particle Tracer (GPT [1]) offers all of these capabilities in a very user friendly and easy to customize package. Due to the general character of the code and its flexibility, GPT is suited for a large number of purposes.

The main features of GPT, new additions to the package and selected cases that highlight the capabilities of GPT are presented.

## 2 THE GPT CODE

### 2.1 A brief description

The purpose of the General Particle Tracer software package [2] is to aid in the design of accelerators and beam lines by using modern, particle tracking techniques. GPT is a 3D particle tracer developed to simulate charged particle dynamics in electromagnetic fields.

The GPT inputfile language supports the use of variables, functions and algebraic expressions, thus giving the user great flexibility. Automatic parameter scanning is also incorporated. GPT has a large set of standard elements, i.e. sector magnets, quadrupoles, solenoids and accelerating structures. Additional elements can easily be developed and exchanged with other users. Elements can be positioned anywhere and with any orientation in 3D space. The effect of fringe fields can also be simulated.

GPT tracks macro particles through the electromagnetic fields and offers complete freedom in the initial particle distribution. The space-charge forces are calculated from full 3D relativistic particle-particle interaction [3]. To ensure both accuracy and speed, an embedded fifth order Runge-Kutta method with adaptive stepsize control [4] is used to solve the equations of motion of the particles in the time domain. Additional differential equations can be solved while tracking the particles, ensuring self-consistent results.

The data-analysis and plotting capabilities of GPT complete the possibilities of the package. GPT is written in ANSI-C for portability and runs on UNIX platforms and PC's. Furthermore a user interface with on-line help and graphical output is available for Windows 95/NT.

### 2.2 Different types of particles

GPT has always been capable of simulating any type of particle. Electrons, positrons and ions can all be simulated. In addition, the new GPT kernel is capable of simulating different types of particles simultaneously and can calculate all 3D space-charge effects among the particles.

Particle sets can be used to sort different types of particles. For example electron and positron beams can be simulated simultaneously. The merging and colliding of various beams can be simulated.

### 2.3 Field maps

Field maps are a convenient tool to import electromagnetic fields, for example obtained with a Poisson solver, into GPT. Electrostatic and magnetostatic field maps for 2D, 2.5D and 3D problems can be imported. The specified fields must lie on an equidistant rectangular grid, allowing the code to immediately find the enclosing square or cube. All interpolations are performed bilinearly to further improve the simulation speed. Furthermore, standing and traveling wave resonant cavities can also be simulated.

## 3 APPLICATIONS

To demonstrate a few of the capabilities of the GPT package the following cases are examined: Basic beam line and accelerator design, simulation of a free-electron laser and the highly specialized design of a depressed collector.

### 3.1 Beam line design

One of the main applications of GPT is beam line design. It is especially useful for high accuracy simulations including space-charge. This example presents a beam injection system, which in its normal mode is isochronous [5]. The system uses a minimum number of magnetic components, 3-4 dipoles, 2 quadrupole triplets and 6 sextupoles.

Figure 1 shows the evolution of the rms-bunchlength in the system. The simulations are performed for different settings of the bunch charge, hereby affecting the space-charge forces. Without space-charge, the simulations agree with the predictions of the TRANSPORT code [6]. Space-charge has a large effect on the bunchlength in the second part of the system. However, due to the careful positioning of the various beam line components, the final increase remains restricted to a few percent.

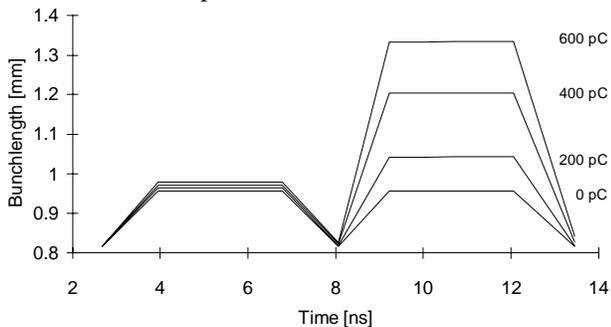


Figure 1: The effect of space-charge on the evolution of the bunchlength in an isochronous bend system.

An important consideration in the design of this type of system is the effect of energy spread in the initial beam. Figure 2 shows the influence of energy spread on the rms-bunchlength, at the exit of the system. The various lines represent different bunch charges.

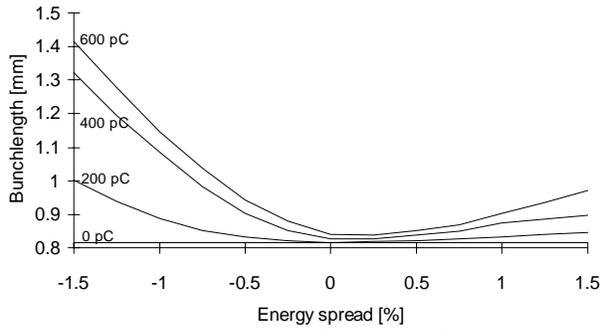


Figure 2: Bunchlength as function of initial energy spread at the exit of an isochronous bend system. The corresponding bunch charges are indicated.

### 3.2 Accelerator design

The FELIX accelerator consists of a standing wave prebuncher and a traveling wave buncher followed by two RF-linac sections [7]. Because of the low energy of the electrons emitted by the gun (100 keV), space-charge effects are very important in the first section of the beam line.

A free-electron laser (FEL) employing a RF-linac accelerator as injector, is very sensitive to the amplitude and phase of the RF power applied to different parts of the injector. The microbunches need to be reproduced exactly alike and at the same intervals to build up the radiation in the optical cavity. Furthermore the shift in average energy during a macropulse due to imperfect stability of the RF sources should be small compared to the energy spread within a single microbunch.

In FELIX the buncher and linac are fed from a common source. Figure 3 shows the final energy as function of the RF phase. The different lines represent the difference in operating phase of the linac in degrees from the top of the wave. Between  $-5^\circ$  and  $-10^\circ$  the output energy is almost independent of the input RF phase. This operating phase will produce the greatest micropulse to micropulse reproducibility.

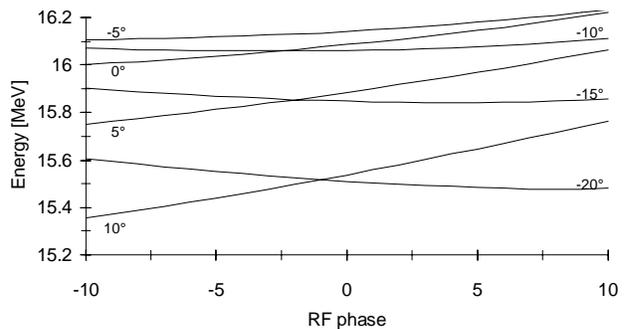


Figure 3: Output energy after the buncher and linac of FELIX as function of RF phase for various operating points of the linac.

GPT has the possibility to solve additional user-specified differential equations while tracking the particles. Using this option, the beam-loading field in the constant gradient FELIX linac is calculated from the

particle trajectories in the above results. This method ensures that the beam-loading wave has the correct phase.

### 3.3 Free Electron Lasers

The complete FOM Fusion FEM [8] beam line has been simulated using the GPT code. During the commissioning of the first part of the beam line, the settings predicted by the simulations were proven to be accurate within a few percent.

Additional differential equations for the amplitude and phase of the various longitudinal and transverse modes in the waveguide of the FEM are solved while tracking the particles. This enables multi-mode, multi-frequency examination of the modes in the cavity and their influence on the particle trajectories. The method was used to optimize both beam transport and output power.

Figure 4 shows the energy of the individual particles going through the step-tapered undulator of the FEM. The particles, on average, lose energy to build up output power, at the cost of an introduction of an energy spread of 300 keV.

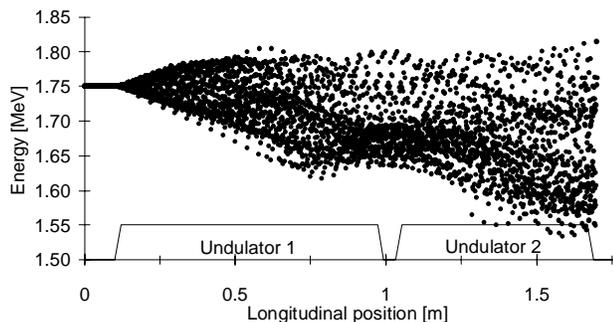


Figure 4: Snap shot of the energy of individual particles of the beam on flowing through the FEM undulators. The loss in average energy is contributed to the radiation.

### 3.4 FEM depressed collector design

Currently the GPT code is being used to design the 2 MV electrostatic decelerator and depressed collector of the FOM Fusion FEM [8]. The design is complicated due to the required 99.8% collection efficiency, the space-charge forces of the 12 A continuous beam and the energy spread of 300 keV generated by the undulator.

The electrostatic decelerator has been modeled in GPT using analytical expressions. The cylindrical symmetric electrostatic fields of the depressed collector [9] are imported into GPT using a 2D field map. In the simulation shown in Figure 5, an additional external magnetic field generated by the fields of two solenoids positioned above and below the collector are used to minimize the backstream of electrons. The asymmetric bending scheme requires a 3D treatment, even for a cylindrical symmetric collector.

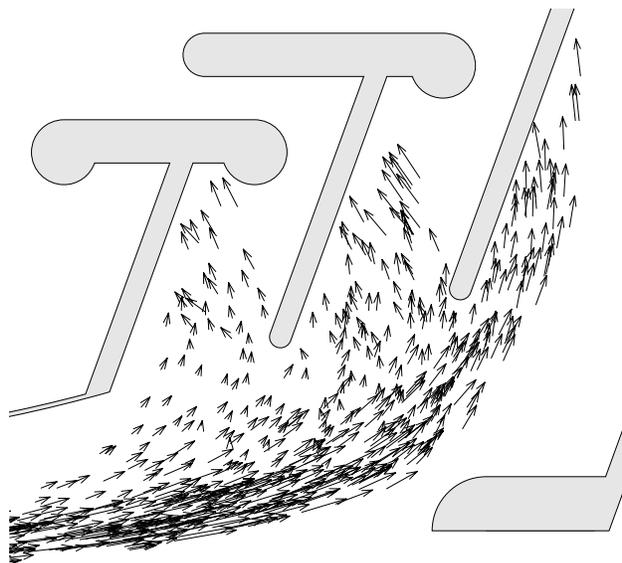


Figure 5: Primary electrons in the FEM depressed collector. The tail of the vector is proportional to the energy of the particle. The energy of the incoming beam ranges from 75 keV to 425 keV.

In the actual collector the beam will be rotated slowly by two sets of coils to homogeneously distribute the dissipation over the collector plates. The analysis of the effect of scattered primaries and secondaries is scheduled to take place in the upcoming months.

## 4 CONCLUSION

The high accuracy, the possibility to add custom code, the large number of standard elements and the user friendly interface, all make GPT an attractive analysis tool. Also the possibility to add self-consistent differential equations makes GPT a powerful tool for the design of accelerators and beam lines, and even FEL's. The current status of the GPT project can be found on the web at <http://www.pulsar.nl/gpt>.

## 5 ACKNOWLEDGEMENTS

The authors gratefully thank Prof. W.A. Gillespie and C.A.J. van der Geer for their contributions to this paper.

## 6 REFERENCES

- [1] M.J. de Loos, et al, Proc. 5<sup>th</sup> Eur. Part. Acc. Conf., Sitges, (1996) pp. 1241.
- [2] GPT User Manual, Pulsar Physics, Flamingostraat 24, 3582 SX Utrecht, The Netherlands.
- [3] Ch. Bourat, CGR-MeV, Thesis, (1988).
- [4] W.H. Press, et al., Numerical Recipes in C, Cambridge Univ. Press, 2<sup>nd</sup> edition, (1992) pp. 714.
- [5] W. A. Gillespie, Nucl. Inst. Meth. Phys. Res., A 374, (1996) pp. 395-400.
- [6] K.L. Brown, et al, CERN Report 80-04, (1980).
- [7] P.W. van Amersfoort, et al, Nucl. Inst. Meth. Phys. Res., A 318, (1992) pp. 42.
- [8] A. van der Linden, et al., Proc. 4<sup>th</sup> Eur. Part. Acc. Conf, London, (1994) pp. 849.
- [9] A. Smirnov, Private communication.