A TIME-ADAPTIVE MESH APPROACH FOR THE SELF-CONSISTENT SIMULATION OF PARTICLE BEAMS *

S. Schnepp**, E. Gjonaj, T. Weiland, Technische Universitaet Darmstadt,
Institut fuer Theorie Elektromagnetischer Felder, Schlossgartenstr. 8, 64289 Darmstadt, Germany

Abstract

The high spatial resolution needed in self-consistent particle simulations can be achieved using adaptive mesh refinement techniques (AMR). In this paper the code SMOVE which is based on such an adaptive mesh is presented. It allows for accurate simulations with a high spatial resolution in the vicinity of the particle bunches. Furthermore, it offers the possibility of simulating long accelerator structures of several meters length. The code is tested and validated using the RF electron gun of the Photo Injector Test Facility at DESY Zeuthen (PITZ) as an example. The evolution of various beam parameters along the gun is compared with the results obtained by different beam dynamics codes.

INTRODUCTION

In many applications the self-consistent simulation of charged particle beams is necessary. Well-known codes like the MAFIA TS modules [1] use a fixed computational grid which has to resolve the bunch adequately. Therefore, they suffer from enormous memory consumption. A remedy to this limitation is the application of adaptive mesh refinement techniques. However, since their application in Finite-Difference methods in time-domain is critical concerning numerical instability, usually problem-matched but static meshes are used.

The simulation of long accelerator structures, is a particularly difficult task due to diverse aspects. Regarding the geometrical dimensions of the bunches in a photo injector and the injector section itself the problem has an extreme multi-scale character. Considering the PITZ injector [2] as an example, the laser pulse has a longitudinally flat-top time profile with a FWHM length of 20 ps; 2 ps rise and fall time, respectively. The emitted bunches evolve to a length of approximately 10 mm whereas the booster cavity, accelerating the particles to relativistic energies, is located at a position of 2.4 m behind the cathode. Besides the small mesh step size needed for resolving the extensions of such short bunches, the requirements on spatial grid resolution are further increased because of the high-frequency fields being excited by the bunch. Since these fields experience Lorentz-contraction in the direction of bunch motion their spectrum is dominated by frequencies in the THz range. Thus, resolving the complete structure with an adequate mesh step size leads to an enormous number of mesh cells which cannot be handled with commonly available computational resources.

A possible approach for solving the problem is the application of a time-adaptive mesh technique. Starting with a rather coarse base grid, a local refinement is applied in a small region containing the particles. The topology of the grid is automatically adapted as the particles move within the computational domain. A related idea has been proposed in [3] for the calculation of wake potentials for short bunches.

TIME-ADAPTIVE MESH REFINEMENT

Data structure

In order to provide efficient and flexible ways of manipulating the computational grid, the refinement process is organized in a hierarchical manner. Local refinement at a given position is provided by bisectioning of existing, large cells of the underlying grid. The data structures which describe this procedure adequately, are binary trees growing from each cell of the base grid. The mesh step size Δ in such a refinement process decreases according to

$$\Delta = \Delta_{\text{base}}/2^N, \tag{1}$$

with N being the level of refinement.

Each transition between regions of different mesh step sizes causes numerical reflection since the numerical phase velocity depends on the resolution. The complex reflection coefficient $\underline{r}$ depends on the jump of the grid step size at the transition boundary. For the transition of a homogeneous plane wave from a region 1 to a region 2 the reflection coefficient is given by

$$\underline{r} = -\frac{\frac{\Delta_1}{\Delta_2}\left(e^{-\mathrm{i}\alpha_2}-1\right)+\left(e^{+\mathrm{i}\alpha_1}-1\right)+\phi}{\frac{\Delta_1}{\Delta_2}\left(e^{-\mathrm{i}\alpha_2}-1\right)+\left(e^{-\mathrm{i}\alpha_1}-1\right)+\phi} \quad \text{with} \tag{2}$$

$$\alpha_1 = \beta_1\Delta_1,\ \alpha_2 = \beta_2\Delta_2,\ \phi = \omega^2\frac{\Delta_1}{2\,c^2}(\Delta_1+\Delta_2). \tag{3}$$

Δ_1 and Δ_2 denote the step sizes in two regions and β_1 and β_2 the corresponding numerical propagation constants [4]. Figure 1 shows the dependency of the reflection amplitude on the refinement level N for various grid resolutions.

In order to minimize such numerical reflection effects the difference in the refinement levels of every two neighboring cells N_1 and N_2 is limited to one in the longitudinal as well as in the transversal directions. In other words, only grid transitions of aspect ratio 1:2 are allowed. The refinement level, therefore, smoothly decreases from the most

* Work supported by HGF under contract VH-FZ-005
** schnepp@temf.tu-darmstadt.de

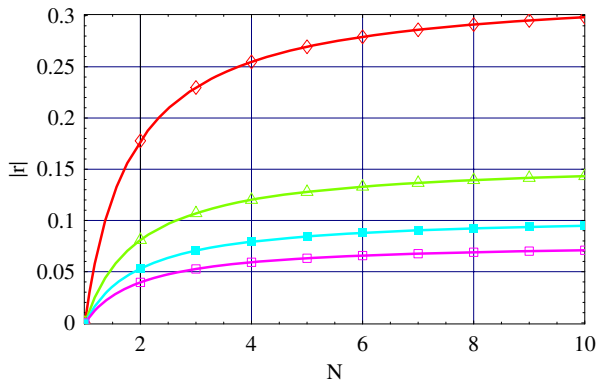

Figure 1: Reflection coefficient vs. refinement level. The reflection coefficient strongly increases for small changes in the step size. (Grid points per wave length λ: ◇ ∼ 5, △ ∼ 10, ■ ∼ 15, □ ∼ 20)

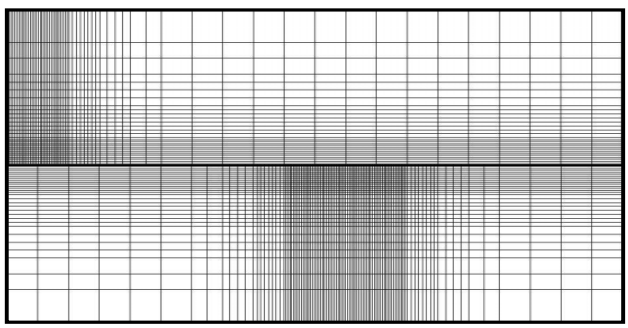

Figure 2: Snapshots of the adaptively refined computational grid. The upper half corresponds to the mesh during emission. The lower half depicts the dynamically refined mesh at a later time instant.

refined region to the coarse parts of the grid. While the particles move within the computational domain the underlying data structure is automatically updated by expanding or shrinking the binary trees representing the mesh refinement. In figure 2 the adaptive mesh refinement procedure is illustrated. The maximum level of grid refinement in the longitudinal as well as the transversal directions is equal to $N = 4$ which corresponds to a refinement by a factor of $2^4 = 16$ of the base grid.

Field Interpolation Scheme

The solution of Maxwell's equations for the discretizations shown in figure 2 is obtained by applying the *Finite Integration Technique* (FIT) [5, 6]. However, in order to apply the method, field components have to be interpolated to their new position after each adaptation of the grid. Since interpolations within explicit time domain methods are critical concerning accuracy and stability, several interpolation techniques have been tested.

Linear interpolation is easy to implement and fast in execution. However, a linear interpolation of the high-frequency fields in the vicinity of the bunch is not very accurate. Therefore, the usage of higher order interpolating functions such as splines was investigated.

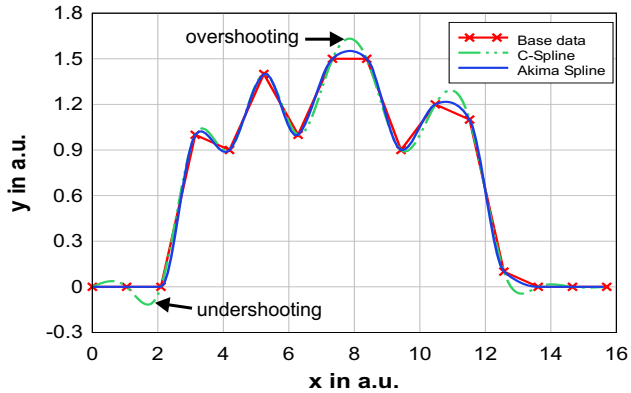

Figure 3: Comparison of a C-Spline and an Akima-Spline interpolation. The Akima-Spline shows less overshooting. As an effect of the slope limiting it is smoothed out to a constant at both ends of the curve.

Since many splines show significant overshooting and oscillatory behavior a class of modified subsplines is utilized. The coefficients of the Akima-subsplines can be calculated explicitly without solving a system of equations. Additionally, the slope-limiting technique [7] can be applied. Using this technique the spline is modified in such a way that alternating gradients in neighboring supporting intervals are flattened. This effect can be seen best at both ends of the Akima-spline in figure 3. More details on this topic are given in [8].

Dispersion properties

For long range particle simulations, low dispersion and a low level of related numerical noise have to be attained in order to achieve accurate results. This can be realized by using a high spatial resolution. Along with the spatial resolution the numerical phase velocity increases, approaching the physical speed of light. The code SMOVE [9] works on the basis of the time-adaptive mesh algorithm described above. Below, its dispersion properties are investigated for different refinement levels.

First, the numerical phase velocity of a sinusoidal wave was determined. The excitation of the wave takes place on one boundary of the computational domain. As the wave propagates, the region of the refined grid is adaptively enlarged in order to cover the area in between the exciting boundary and the front of the traveling wave. In figure 4 (top) the results are given together with a fitted curve and the physical phase velocity normalized by the speed of light in vacuum c_0. Level zero corresponds to the non-refined base grid, resolving one wavelength by five grid points ($\Delta_0 = \lambda/5$). Hence, a refinement level of seven corresponds to a resolution of $\Delta_7 = \Delta_0/(2^7) = \lambda/640$.

The reduction of noise was investigated using a traveling wave of Gaussian shape. Starting from a resolution of eight points, the refinement level was increased until only

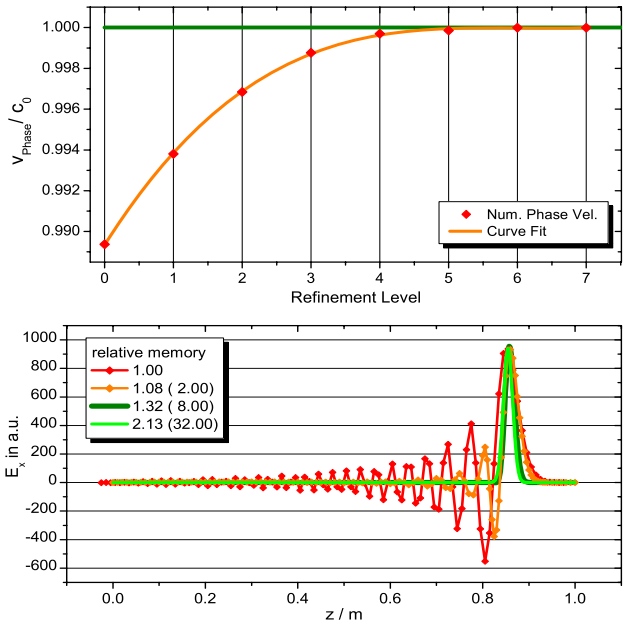

Figure 4: The numerical phase velocity increases with the refinement level and approaches the speed of light in vacuum (top). Gaussian wave for the refinement levels 0, 1, 3 and 5. In the legend, the memory needs for the time-adaptive mesh approach and a static grid of equal resolution (in brackets) are given (bottom).

spurious noise was observed after a distance of five meters. Figure 4 (bottom) shows the result for different refinement levels. The most remarkable property is the memory consumption. In order to achieve an accurate transmission using the time-adaptive mesh approach, memory requirements are increased by a factor of ≈ 2.13 compared to the base grid. For a static grid of the same resolution, however, this factor is 32.

Material discretization

For accurate self-consistent simulations including all effects, the geometry of accelerator components also has to be taken into account. This is realized by assigning material properties to the cells of the computational mesh according to the physical model. Since the accuracy of the shape representation has a strong influence on the accuracy of the results, it should be as precise as possible, especially in the vicinity of the bunch. Therefore, within the code SMOVE, the resolution of the material discretization increases in conjunction with the refinement of the grid.

The geometry is represented using a triangulation of the surfaces. This allows for a precise description independent of the size of the computational cells. In the current implementation, a cell is filled with the material of the component its center point is located in which results in a staircase approximation of the material distribution. This procedure is independent of the geometrical dimensions of the computational cells.

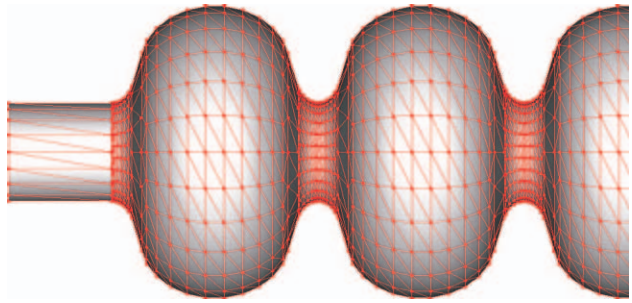

Figure 5: Surface triangulation of a piece of beam tube and 2.5 TESLA-like cavities.

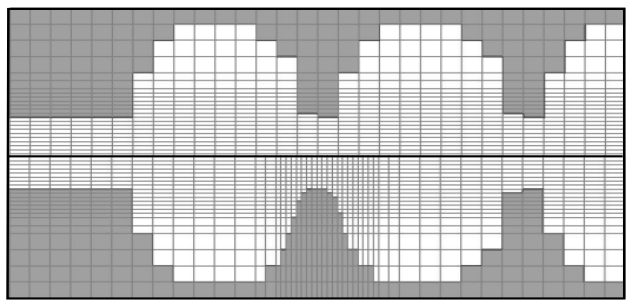

Figure 6: Computational meshes and material discretizations on the basis of the above surface triangulation. Along with the adaptive mesh refinement, the material discretization becomes more accurate. The mesh cells in grey are filled completely with perfectly conducting material.

In figure 5 a surface triangulation of TESLA-like cavities is shown. This triangulation is the basis for the material discretizations shown in figure 6. The meshes have been built for demonstration purposes only. The resolution is, therefore, very coarse. In the upper part of figure 6 the material distribution in the base grid is shown. In the lower part, a region of the mesh was refined. Despite the rather coarse resolution, the representation of the structure in this region is more accurate.

EXAMPLES

The code is tested and validated using the PITZ gun as an example. In figure 7 the evolution of the transversal RMS beam size in tracking and in self-consistent mode are given. For comparison the results obtained with the code ASTRA [10] are shown in the same diagram. In [8] first results for the transverse beam size in self-consistent mode were given. Recently, a more accurate scheme [11] for the particle-to-grid interpolation has been implemented. The results are in good agreement with those given in [12] using a *Discontinuous Galerkin Finite Element* (DG-FEM) approach in time domain as well as those obtained using ASTRA. However, in contrast to the DG-FEM code and SMOVE, ASTRA does not take the geometry of the enclosing accelerator into account.

The RMS beam size in the longitudinal direction (fig. 8

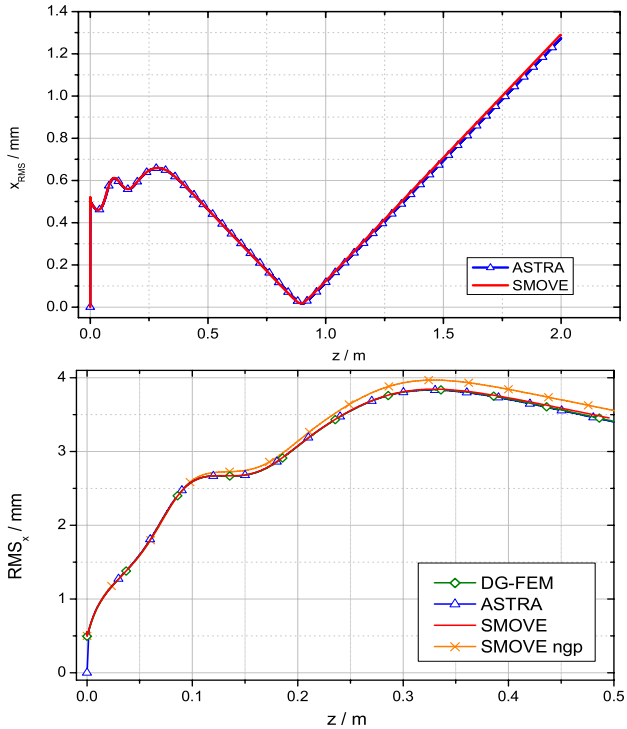

Figure 7: Transverse RMS beam size calculated in tracking mode (top) and in self-consistent mode (bottom).

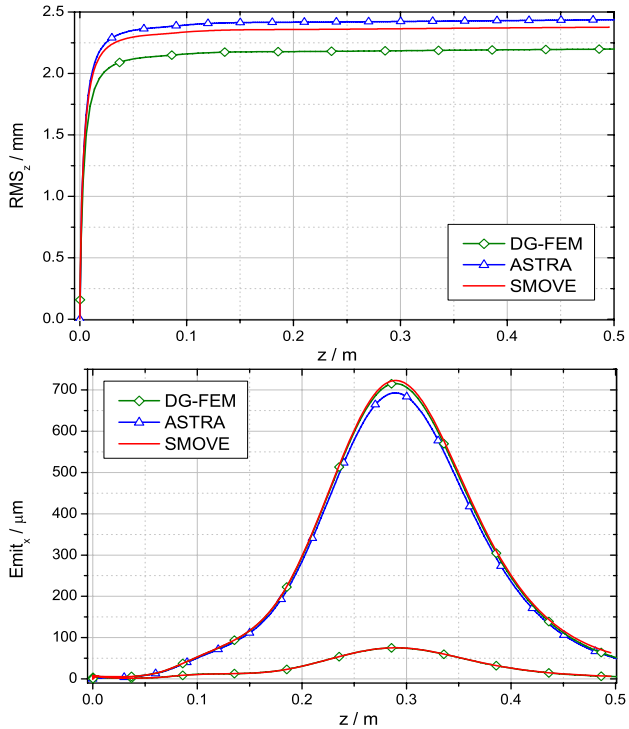

Figure 8: RMS bunch length (top) and transverse RMS emittance (bottom). In the upper curves the RMS emittance is given in normalized form.

top) however, shows differences for all codes. In [13] it was stated that longitudinal resolutions down to 20 μm are necessary in the vicinity of the cathode to cover all aspects of the particle motion in that low-energetic regime. Indeed, variations of the longitudinal mesh step size do have an impact on the computed bunch length. For the evolution of the transverse beam emittance the codes also yield slightly different results whereas the results of SMOVE and DG-FEM coincide. Further investigations have to be done before a final statement on the accuracy of these results can be made.

CONCLUSIONS

A new approach to self-consistent simulations of charged particles including geometry effects was presented. It is based on a time-adaptive mesh which is automatically refined in the vicinity of particles. Due to the considerable reduction of memory demands it is well-suited for the application to long accelerator structures. Results obtained utilizing the code SMOVE, based on this method, were presented and show good agreement to those attained using other codes. In order to achieve an even higher resolution in dedicated regions, like the vicinity of the cathode, an additional static mesh refinement is a work in process.

REFERENCES

[1] CST GmbH, Bad Nauheimer Str. 19, 64289 Darmstadt

[2] A. Oppelt, "The Photo Injector Test Facility at DESY Zeuthen: Results of the First Phase", Proc. LINAC 2004

[3] M. Dehler, T. Weiland, "Calculating Wake Potentials for Ultra Short Bunches", AIP Vol. 297, 1993

[4] P. Thoma, "Zur numerischen Loesung der Maxwellschen Gleichungen im Zeitbereich", Darmstädter Diss.30, 1997

[5] T. Weiland, "A Discretization Method for the Solution of Maxwell's Equations for Six-Component Fields", Int. J. Elec. and Comm. (AEÜ), Vol. 31 (1977), pp 116 - 120.

[6] T. Weiland, "On the Numerical Solution of Maxwell's Equations and Applications in Accelerator Physics", Particle Accelerators Vol. 15 (1984), pp 245-292.

[7] C. Laney, "Computational Gas Dynamics", Cambridge University Press, 1998, pp 158

[8] S. Schnepp, E. Gjonaj and T. Weiland, "Development of a Self-Consistent Particle-in-Cell (PIC) Code using a Time-Adaptive Mesh Technique", EPAC, 2006, Edinburgh

[9] SMOVE – A Code for Self-Consistent Particle Beam Simulations over Long Distances, "pdf-Documentation", http://www.temf.de/downloads/SMOVEdoc.pdf

[10] ASTRA, http://www.desy.de/∼mpyflo/

[11] J. Villasenor, O. Buneman, "Rigorous charge conservation for local electromagnetic field solvers", Comp. Phys. Comm. 69 pp 306-316, 1992

[12] E. Gjonaj, T. Lau, S. Schnepp and T. Weiland, "Accurate modelling of charged particle beams in linear accelerators", *Submitted to New J. of Phys.*

[13] M. Zhang, P. Schuett, "TESLA FEL Gun Simulations with PARMELA and MAFIA", CAP, 1996, Williamsburg