TESLA Report 2004-13

Calculation of 3D Space-Charge Fields of Bunches of Charged Particles by Fast Summation

Gisela Pöplau[1⋆], Daniel Potts[2], and Ursula van Rienen[1]

[1] Institute of General Electrical Engineering, Rostock University, D-18051 Rostock, Germany
gisela.poeplau@etechnik.uni-rostock.de
ursula.van-rienen@etechnik.uni-rostock.de
[2] Institute of Mathematics, University of Lübeck, D- Lübeck, Germany
potts@math.uni-luebeck.de

Summary. The fast calculation of space-charge fields of bunches of charged particles in three dimensional space is a demanding problem in accelerator design. Since particles of equal charge repel each other due to space-charge forces, it is difficult to pack a high charge in a small volume. For this reason, the calculation of space-charge forces is an important part of the simulation of the behaviour of charged particles in these machines. As the quality of the charged particle bunches increases, so do the requirements for the numerical space-charge calculations.

In this paper we develop a new fast summation algorithm for the determination of the electric field generated by N charged particles. Applying the nonequidistant Fast Fourier Transform (NFFT) the fast summation requires only $O(N \log N)$ operations. The numerical test cases confirm this behaviour.

1 Introduction

Recent developments in the field of charged particle accelerator research make high demands on numerical simulations. Among the simulation problems of particle dynamics is the three dimensional calculation of Coulomb repulsion, so-called space-charge fields, of bunches containing millions of particles.

Widely used methods for the calculation of these space-charge fields are the particle-mesh method and the particle-particle method [5]. The particle-mesh method,

⋆ supported by a research grant from DESY, Hamburg

based on solving Poisson's equation for the electrostatic potential, is typically much faster than the particle-particle method. Furthermore, it provides better numerical results for sufficiently "smooth" distributed particles. Progress in the particle-mesh method has been achieved with the construction of nonequispaced adaptive grids and the development of multigrid Poisson solvers for grids with large aspect ratio [8, 7]. The computational effort of the resulting algorithm scales linearly with the number of particles for a wide range of particle distributions.

Although the particle-mesh method provides good results for most real life simulations [8], it is on the edge of the requirements for the simulation of very short bunches present in rf-photoguns based on femtosecond excitation lasers. Also problematic are simulations of high peak current bunches with a long tail as present after the compression stage of the Tesla Test Facility (TTF), a novel linear accelerator recently under development and construction at DESY in Hamburg [1]. In both cases, the main difficulty is the fact that to keep computational costs and memory consumption at an acceptable level, a very high aspect-ratio mesh needs to be constructed resulting in the degradation of the convergence behaviour of the Poisson solver [7].

Motivated by the above-mentioned problems with the particle-mesh method we deal in this paper with the development of a new fast calculation technique for the particle-particle model. The particle-particle method calculates the self-induced field E generated by N charged particles with the superposition principal. Let the ℓ-th particle have the charge q_ℓ and the position r_ℓ ($\ell = 1,\dots,N$) and let ε_0 denote the dielectric constant, then

$$E(r) = \frac{1}{4\pi\varepsilon_0} \sum_{\ell=1}^{N} q_\ell \frac{r - r_\ell}{\|r - r_\ell\|^3}, \qquad r, r_\ell \in \mathbb{R}^3,\ r \neq r_\ell,\ \ell = 1,\dots,N. \tag{1}$$

The direct summation which requires $O(N^2)$ operations is either very time consuming or causes large simulation errors due to the restricted number of particles. This essentially eliminates its applicability to real life simulations, unless the computation is the restriction to 2 D models [13]. In order to make large scale problems tractable it is essential to compute these interactions efficiently. A number of algorithms have been proposed for this purpose. The fast multipole method (FMM) has been one of the most successful, especially for nonuniform particle distributions (see [14] and references therein). Our new method is fully 3 D and based on the nonequidistant Fast Fourier Transform (NFFT) [6], hereby reducing the computational from $O(N^2)$ to $O(N \log N)$. Although this is still slower compared to the best particle-mesh methods, it could prove to be advantageous for ultra-short and 'TTF'-like bunches because no mesh needs to be constructed.

In the next chapter we develop the main principles of the fast summation by NFFT and present an algorithm for the computation of (1). Finally the numerical experiments in section 3 show that the fast summation technique provides the values for the field with an acceptable numerical error in much shorter simulation time compared to direct methods.

2 Fast Summation at Nonequispaced Knots by NFFTs

The fast computation of special structured discrete sums similar to (1) is a frequently appearing task in the study of particle models [3, 4, 14]. The new fast summation technique we develop in this paper is based on a method first presented in [9].

The fast computation of E at the positions r_j ($j = 1,\dots,N$) is performed for the two sums

$$E(r_j) = \frac{1}{4\pi\varepsilon_0}\left(r_j \sum_{\substack{\ell=1\\ j\neq\ell}}^{N} \frac{q_\ell}{\|r_j - r_\ell\|^3} - \sum_{\substack{\ell=1\\ j\neq\ell}}^{N} q_\ell \frac{r_\ell}{\|r_j - r_\ell\|^3} \right) \tag{2}$$

in the following way: As suggested in [9] we use a separation of the knots r_j and r_ℓ by Fourier expansions. More precisely, we split the function $1/\|x\|^3$ into the sum $1/\|x\|^3 \approx \mathcal{K}_{\mathrm{NE}} + \mathcal{K}_{\mathrm{R}}$. Thereby the function $\mathcal{K}_{\mathrm{NE}}$ is supposed to have small support with $\operatorname{supp}\mathcal{K}_{\mathrm{NE}} = \{x \in \mathbb{R}^3;\ \|x\| \le \varepsilon_I\}$. It can be considered as the near field approximation of $1/\|x\|^3$. Further the function $\mathcal{K}_{\mathrm{R}}$ is chosen as a smooth 1–periodic function also referred to as the regularisation of $1/\|x\|^3$. The construction of $\mathcal{K}_{\mathrm{R}}$ is somewhat technical so we don't give it at this place. It is needed for the computation of the discrete Fourier coefficients b_k defined by

$$b_k := \frac{1}{n^3} \sum_{j\in I_n} \mathcal{K}_{\mathrm{R}}(j/n)\, \mathrm{e}^{-2\pi \mathrm{i} jk/n} \tag{3}$$

where k runs over the finite index set $I_n := \{-n/2,\dots,n/2-1\}^3$. A detailed description can be found in [9] for the one dimensional case which can be straightforward applied to the three dimensional problem.

Next, we approximate the smooth function $\mathcal{K}_{\mathrm{R}}$ by the discrete finite Fourier sum $\mathcal{K}_{\mathrm{RF}}$ given by

$$\mathcal{K}_{\mathrm{R}} \approx \mathcal{K}_{\mathrm{RF}} = \sum_{k\in I_n} b_k\, \mathrm{e}^{2\pi \mathrm{i} k\cdot}. \tag{4}$$

Then, $1/\|x\|^3$ is replaced by $1/\|x\|^3 \approx \mathcal{K}_{\mathrm{RF}} + \mathcal{K}_{\mathrm{NE}}$. Using the outstanding property $\mathrm{e}^{2\pi\mathrm{i}(r_j - r_\ell)} = \mathrm{e}^{2\pi\mathrm{i} r_j}\, \mathrm{e}^{-2\pi\mathrm{i} r_\ell}$, we obtain the desired separation of r_j and r_ℓ by

$$\frac{1}{\|r_j - r_\ell\|^3} \approx \sum_{k\in I_n} b_k\, \mathrm{e}^{2\pi\mathrm{i} k r_j}\, \mathrm{e}^{-2\pi\mathrm{i} k r_\ell} + \mathcal{K}_{\mathrm{NE}}(r_j - r_\ell)$$

and finally

$$\begin{aligned} \hat{\alpha}_j &:= \sum_{\substack{\ell=1\\ j\neq\ell}}^{N} \frac{\alpha_\ell}{\|r_j - r_\ell\|^3} \\ &\approx \sum_{k\in I_n} b_k \left(\sum_{\ell=1}^{N} \alpha_\ell\, \mathrm{e}^{-2\pi\mathrm{i} k r_\ell} \right) \mathrm{e}^{2\pi\mathrm{i} k r_j} + \sum_{\substack{\ell=1\\ j\neq\ell}}^{N} \alpha_\ell\, \mathcal{K}_{\mathrm{NE}}(r_j - r_\ell) - \alpha_j \sum_{k\in I_n} b_k. \end{aligned} \tag{5}$$

The expression in the inner brackets can be computed by a multivariate $\mathrm{NFFT}^{\mathrm{T}}(n)$, where $\mathrm{NFFT}^{\mathrm{T}}$ denotes the transposed version of the NFFT [6]. This is followed by

n^3 multiplications with b_k and completed by a multivariate NFFT(n) to compute the outer sum with the complex exponentials. By construction the function $\mathcal{K}_{\mathrm{NE}}$ has a small support such that the summation can be done very efficiently. The approximation of (5) is used in (2) with $\alpha_\ell = q_\ell$ and $\alpha_\ell = q_\ell r_\ell$, respectively. Applying the recently developed fast Fourier transform for nonequispaced data (NFFT) (see [11] and references therein), we come up with a fast summation algorithm. This NFFT summation requires for "sufficiently uniformly distributed" points r_ℓ only $O(N \log N)$ arithmetic operations and can be simply implemented using the public domain NFFT toolbox (see e.g. [6]). Note that the NFFT itself is based on the approximation of functions by translates of one function, which is taken as a Kaiser–Bessel function in our numerical computations. In summary we obtain the following

Algorithm:

Precomputation:

i) Computation of $(b_k)_{k \in I_n}$ by (3).

ii) Computation of $K_{\mathrm{NE}}(r_j - r_\ell)$ for all $(j = 1, \ldots, N)$ and $\ell \in I_{\varepsilon_{\mathrm{I}}}^{\mathrm{NE}}(j)$, where $I_{\varepsilon_{\mathrm{I}}}^{\mathrm{NE}}(j) := \{\ell \in \{1, \ldots, N\} : \|r_j - r_\ell\| < \varepsilon_{\mathrm{I}}\}$.

1. For $k \in I_n$ compute by four multivariate NFFT$^{\mathrm{T}}$(n)s

$$\hat{q}_k := \sum_{\ell=1}^{N} q_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k r_\ell}, \quad \hat{r}_k := \sum_{\ell=1}^{N} q_\ell r_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k r_\ell}.$$

2. For $k \in I_n$ compute the products $d_k := \hat{q}_k b_k \in \mathbb{C}$.
3. For $j = 1, \ldots, N$ compute by a multivariate NFFT(n)

$$f_{\mathrm{RF}}(r_j) := r_j \sum_{k \in I_n} d_k \, \mathrm{e}^{2\pi \mathrm{i} k r_j}.$$

4. For $k \in I_n$ compute the products $d_k := \hat{r}_k b_k \in \mathbb{C}^3$.
5. For $j = 1, \ldots, N$ compute by three multivariate NFFTs(n)

$$f_{\mathrm{RF}}(r_j) := f_{\mathrm{RF}}(r_j) - \sum_{k \in I_n} d_k \, \mathrm{e}^{2\pi \mathrm{i} k r_j}.$$

6. For $j = 1, \ldots, N$ compute the near field sums

$$f_{\mathrm{NE}}(r_j) = r_j \sum_{\ell \in I_{\varepsilon_{\mathrm{I}}}^{\mathrm{NE}}(j)} q_\ell \mathcal{K}_{\mathrm{NE}}(r_j - r_\ell) - \sum_{\ell \in I_{\varepsilon_{\mathrm{I}}}^{\mathrm{NE}}(j)} q_\ell r_\ell \mathcal{K}_{\mathrm{NE}}(r_j - r_\ell).$$

7. For $j = 1, \ldots, N$ compute the near field corrections

$$\tilde{E}(r_j) = \frac{1}{4\pi\varepsilon_0} \left(f_{\mathrm{NE}}(r_j) + f_{\mathrm{RF}}(r_j) \right).$$

Note, that usually the field values are requested at same the locations as the location of the particles. But the algorithm can also evaluate field values at other points the number of which has not to be in coincidence with the number of particles (see [9]).

3 Numerical Results

The algorithms for the fast summation have been implemented in C and tested on an AMD Atlon xp1800+ 512MB RAM, SuSe-Linux 8.0 using double precision arithmetic. Throughout our experiments we have applied the NFFT/NFFT$^{\mathrm{T}}$ package [6] with Kaiser–Bessel functions, oversampling factor $\rho = 2$ and several bandwidth parameters n which will be specified in the examples. Further the NFFT/NFFT$^{\mathrm{T}}$ algorithms require the parameters p (guarantees the smoothness of $\mathcal{K}_{\mathrm{R}}$ up to the derivative of order $p-1$) and m (controls the accuracy of interpolation by the Kaiser–Bessel functions) which are for the fast summation chosen as $p = 2$ and $m = 2$. Note, that the fast summation method suggested in (5) was first proposed for the univariate case in [9] and for the bivariate case in [10] (see also [2]). There error estimates are proved to obtain clues about the choice of the involved parameters. For a numerical comparison with the fast multipole method in 2D see [10]. With the algorithm for the calculation of the electric field we extend these methods to $\mathbb{R}^3$.

As numerical test we used a spherical bunch uniformly filled with charged particles. The total charge of the sphere has been kept with $Q = -1$ nC. Thus the particles are assumed to posses the charge $q_i = q = -1/N$ nC $(i = 1,\ldots,N)$, where N denotes the number of particles in the sphere. These particles are also regarded as macro-particles representing the distribution of all particles (for instance electrons) in a bunch. The uniform particle distributions have been generated with the tracking code GPT (General Particle Tracer) [12] by means of Hammersley sequences. These sequences provide pseudo random numbers such that distance between two particles does not become too small. The advantage of such generated distributions is represented in Fig. 1 where the numerical error is compared to particle distributions generated with straightforward computed random numbers.

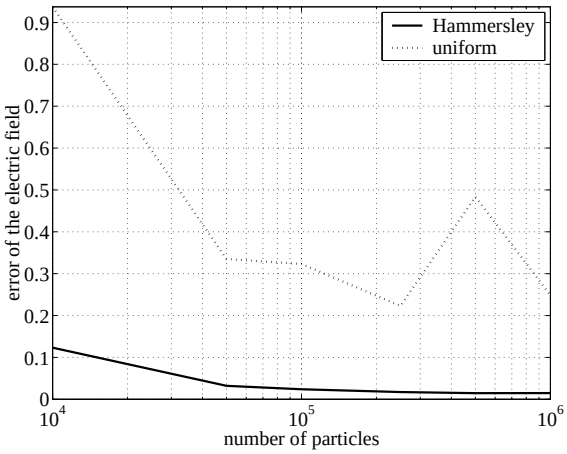

Fig. 1. The error $\mathcal{E}_2$(theo,fast)) of the electric field for particle distributions generated by Hammersley sequences and by straightforward computed random numbers, respectively.

The fast summation technique is not restricted to the calculation of the discrete sum (1) but can be applied to a great variety of discrete sums appearing in the study of particle models. In order to demonstrate the efficiency of our new method with a more simple discrete sum we start with the calculation of the potential φ caused by N charged particles with charge q given by

$$\varphi(r_j) = \frac{1}{4\pi\varepsilon_0} \sum_{\substack{\ell=1 \\ j\neq\ell}}^{N} \frac{q}{\|r_j - r_\ell\|}, \qquad (r_j \in \mathbb{R}^3).$$

The fast summation strategy described in section 2 can be easily adapted to the above discrete sum. Since a sphere uniformly filled with an increasing number of particles of equal charge gets more and more close to a sphere with charge $Q = \sum_{i=1}^{N} q$, we compare the results of the summation to the analytically known potential given by

$$\varphi_{\text{theo}}(r_j) = \frac{Q}{4\pi\varepsilon_0} \left(\frac{3}{2} - \frac{\|r_j\|}{2R^2} \right), \qquad \|r_j\| \leq R,$$

where R denotes the radius of the sphere.

We have investigated the numerical error

$$\mathcal{E}_2(\text{a,b}) = \left(\sum_{k=1}^{N} |\varphi_{\text{a}}(r_k) - \varphi_{\text{b}}(r_k)|^2 \right)^{1/2} \left(\sum_{k=1}^{N} |\varphi_{\text{a}}(r_k)|^2 \right)^{-1/2},$$

where a and b represent the different techniques for the computation either of the potential or the electric field (slow: straightforward summation, fast: fast summation, theo: analytical solutions). Similarly the computational time for the straightforward summation and for the fast summation based on (5) is denoted by t_{slow} and t_{fast}, respectively.

N	n	t_{slow}	t_{fast}	$\mathcal{E}_2$(theo,slow)	$\mathcal{E}_2$(theo,fast)	$\mathcal{E}_2$(slow,fast)
10000	64	6.680e+00	2.310e+00	2.586e-03	2.544e-03	1.206e-04
50000	64	1.777e+02	7.140e+00	1.018e-03	9.755e-04	9.654e-05
100000	64	7.092e+02	1.770e+01	5.630e-04	5.283e-04	1.002e-04
250000	128	4.470e+03	4.821e+01	2.952e-04	2.584e-04	1.125e-04
500000	128	1.756e+04	8.951e+01	2.043e-04	1.647e-04	1.103e-04
1000000	128	7.024e+04*	2.257e+02		1.079e-04	

Table 1. Computational time and the error $\mathcal{E}_2$ for the potential φ, *estimated.

The numerical experiments documented in Table 1 show that we obtain with our fast algorithm the same errors as with the straightforward (slow) summation but with an numerical effort of only $O(N \log N)$. Hereby the parameters of the NFFT are chosen such that the approximation error is less than the simulation error. Depending on the number of particles the Fourier sum (4) has been computed as NFFT(n) with

$n = 32$, $n = 64$ and $n = 128$, respectively (see Tables 1 and 2). The star * means that the running time of the direct evaluation is obtained by extrapolation. Note that the straightforward evaluation of the potential φ with $N = 5 \cdot 10^6$ requires more that 10 days (see Fig. 2). Finally we test the algorithm for the computation of the electrostatic field suggested in section 2. It is well known that the field of a charged sphere is given by

$$E_{\text{theo}}(r_j) = \frac{Q}{4\pi\varepsilon_0}\left(\frac{r_j}{R^3}\right), \qquad \|r_j\| \le R.$$

Here we consider the error

$$\mathcal{E}_2(\text{a,b}) = \left(\sum_{k=1}^{N} \|E_{\text{a}}(r_k) - E_{\text{b}}(r_k)\|^2\right)^{1/2} \left(\sum_{k=1}^{N} \|E_{\text{a}}(r_k)\|^2\right)^{-1/2}.$$

Table 2 represents the results of the related numerical simulations. Figure 2 compares the performance of the fast summation algorithm with NFFT to the direct slow summation. It shows that the NFFT summation scales with $O(N \log N)$. Hence this

N	n	t_{slow}	t_{fast}	$\mathcal{E}_2$(theo,slow)	$\mathcal{E}_2$(theo,fast)	$\mathcal{E}_2$(slow,fast)
10000	32	7.580e+00	3.680e+00	1.232e-01	1.232e-01	1.068e-03
50000	64	1.930e+02	2.185e+01	3.204e-02	3.205e-02	5.765e-04
100000	64	7.710e+02	4.810e+01	2.393e-02	2.394e-02	4.662e-04
250000	128	5.781e+03	1.635e+02	1.716e-02	1.718e-02	5.462e-04
500000	128	2.312e+04*	2.699e+02		1.446e-02	
1000000	128	9.245e+04*	6.031e+02		1.468e-02	

Table 2. Computational time and the error $\mathcal{E}_2$ for the electric field E, *estimated.

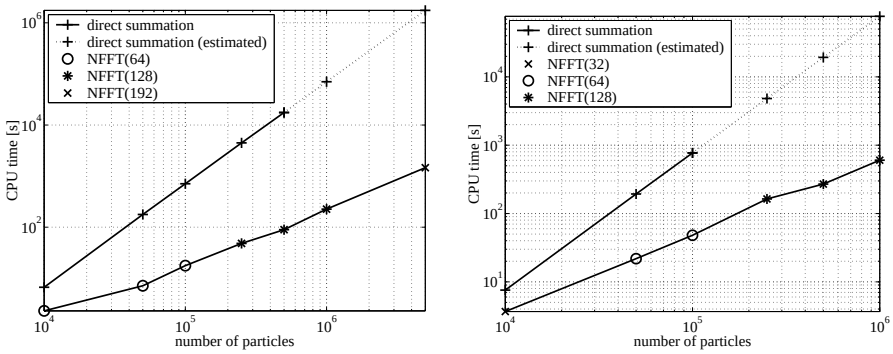

Fig. 2. Performance of the fast NFFT-algorithm compared to the direct summation: computation of φ (left), computation of E (right).

new summation technique enables the computation of fully 3 D particle-particle interactions in real life applications.

4 Acknowledgment

The authors like to thank the physicists Marieke de Loos (Pulsar Physics, the Netherlands) and Bas van der Geer (TU Eindhoven, the Netherlands) for supporting the work with both a lot of helpful comments related to accelerator physics and a lot of space-charge simulations performed with the tracking code GPT (General Particle Tracer) [12].

References

1. DESY, Hamburg, Germany, `tesla.desy.de`. *DESY-TTF*.
2. M. Fenn and G. Steidl. Fast NFFT based summation of radial functions. *Sampling Theory in Signal and Image Processing*, 3:1 – 28, 2004.
3. L. Greengard. *The Rapid Evaluation of Potential Fields in Particle Systems*. MIT Press, Cambridge, 1988.
4. L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73:325 – 348, 1987.
5. R. Hockney and J. Eastwood. *Computer Simulation Using Particles*. Institut of Physics Publishing, Bristol, 1992.
6. S. Kunis and D. Potts. NFFT, Softwarepackage, C subroutine library. `http://www.math.uni-luebeck.de/potts/nfft`, 2002 – 2004.
7. G. Pöplau, U. van Rienen, S. van der Geer, and M. de Loos. Multigrid algorithms for the fast calculation of space-charge effects in accelerator design. *IEEE Transactions on Magnetics*, 40(2):714–717, 2004.
8. G. Pöplau, U. van Rienen, S. van der Geer, and M. de Loos. Fast calculation of space charge in beam line tracking by multigrid techniques. In *Proceedings of the 4th Conference on Scientific Computing in Electrical Engineering (SCEE–2002)*, Mathematics in Industry. Springer-Verlag, to appear.
9. D. Potts and G. Steidl. Fast summation at nonequispaced knots by NFFTs. SIAM *J. Sci. Comput.*, 24:2013 – 2037, 2003.
10. D. Potts, G. Steidl, and A. Nieslony. Fast convolution with radial kernels at nonequispaced knots. *Numer. Math.*, 98:329 – 351, 2004.
11. D. Potts, G. Steidl, and M. Tasche. Fast Fourier transforms for nonequispaced data: A tutorial. In J. J. Benedetto and P. J. S. G. Ferreira, editors, *Modern Sampling Theory: Mathematics and Applications*, pages 247 – 270, Boston, 2001. Birkhäuser.
12. Pulsar Physics, De Bongerd 23, 3762 XA Soest, The Netherlands, `www.pulsar.nl/gpt`. *General Particle Tracer (GPT)*, release 2.60 edition.
13. S. van der Geer and M. de Loos. *The General Particle Tracer Code. Design, implementation and application.* PhD thesis, TU Eindhoven, 2001.
14. L. Ying, G. Biros, and D. Zorin. A kernel-independent adaptive fast multipole method in two and three dimensions. *J. Comput. Pysics*, to appear.