

Optimum Choice of NGP, CIC and QS Algorithms in One Dimensional Electrostatic Particle Simulations

Ali Peiravi

Ferdowsi University of Mashhad,
Department of Electrical Engineering, School of Engineering, Mashhad IRAN
Telephone number: (0098) 511-881-5100
Ali_peiravi@yahoo.com

Abstract: Several algorithms of different order and complexity are available for particle simulations. In this study, the actual results of measurements of self heating times as a measure of accuracy of a single species one dimensional electrostatic simulation of thermal plasmas are presented using zero order weightings (NGP), linear weighting (CIC), and quadratic spline (QS) weighting. Optimal range of parameters is found to be along $v_t \Delta t / \Delta x \approx 1.5$ for zero order weightings and $v_t \Delta t / \Delta x \approx 0.5$ for linear and quadratic spline weighting. Considerable increase in self heating time is achieved through truncation in k-space. The results are useful in the optimum choice of simulation algorithm and indicate that using a higher order simulation algorithm is highly recommended, especially when k-space truncation is used. [Journal of American Science 2010;6(10):623-628]. (ISSN: 1545-1003).

Key words: Plasma simulation, self heating times, choice of simulation algorithm, NGP, CIC, QS

Nomenclature

T_e =electron temperature
 T_i =ion temperature
 κ =Boltzman's constant
 m_e =electron mass
 m_i =ion mass
 ω_p =plasma frequency
 λ_D =Debye length
 Δx =simulation grid size
 Δt =simulation time step
 n_0 =particle density
 N_D =number of particles in a Debye length
 N_C =number of particles in a grid spacing
 τ_h =self-heating time
 v_t =electron thermal temperature
 k_{\max} =maximum allowable mode
 k_{last} =last mode kept

1. Introduction

The choice of which algorithm to use in particle simulations involving many particles is highly important since it determines both the computational expense and the accuracy involved since each algorithm naturally introduces some errors due to discretization of both time and space.

Particle simulations have been carried out to study the behavior of electron beams in vacuum tubes and

have later developed into the simulations of both electrostatic and electromagnetic plasmas.

Linear interpolation or other higher order interpolation schemes are almost always used in particle-in-cell simulation because of their lower noise characteristics and accuracy relative to the nearest-grid-point method. The higher order interpolation schemes are chosen because of their optimal performance, balancing a smaller number of particles against more computer operations per particle per time step. However, this is not always the case. Parker (2002) presented large-scale gyrokinetic particle simulations, where sometimes nearest-grid-point interpolation is used with results virtually identical to those of linear interpolation using the same number of particles. He presented a comparison and analysis of nearest-grid-point and linear interpolation schemes showing why nearest-grid-point interpolation can be optimal.

Li et al. (2009) proposed improved particle-in-cell (PIC) algorithms including volume weighting cloud-in-cell model, geometry profile considered explosive electron emission model and divergence error diffused perfectly matched layer boundary in order to increase the precision of the algorithm and decreased the numerical noise. Their model resulted in the development of a user-friendly, 2.5-dimensional PIC code called UNIPIC to simulate high power microwave

source devices and used for their geometry optimization in x-y, z-r and r- ϕ coordinate systems. Results of simulation of a relativistic magnetron, magnetically insulated line oscillator, virtual cathode oscillator have all shown good physics image and beam-wave interaction characteristics.

In the electrostatic simulations, the model consists of charged moving particles experiencing both the force due to themselves and due to the externally applied fields. Maxwell's equations and Newton's Lorentz equations are both applied to follow the motion of these particles. Simulations are done both in discrete space and time. Thus, the models used affect the accuracy and stability of the simulations. The simplest model used is the zero order particle and field weighting called nearest-grid-point (NGP). A better weighting would be first order weighting that smoothes the density and field fluctuations at the expense of accessing two grid points for each particle twice each step. This is called cloud-in-cell (CIC). A higher order weighting is quadratic spline (QS) that further rounds off the roughness of the particle shape.

The non-physical self heating times of a one dimensional electrostatic simulation of thermal plasma are presented using zero order weighting or the nearest-grid-point (NGP), linear weighting cloud-in-cell (CIC-PIC), and quadratic spline weighting (QS) in a momentum conserving code called ES1 that was written around 1972 (Birdsall and Langdon, 2005). It is noteworthy that even energy conserving codes show self-heating for $\Delta t \rightarrow 0$. In such codes, energy is conserved only in the limit of zero for Δt . ES1 is a "particle-in-cell" simulation of the Vlasov Eq. for "1-Dimensional" (in space, plus the corresponding velocity dimension) periodic problems. One may use ES1 to develop insight into plasma behavior and the properties of the Vlasov equation, including surprising nonlinear results. ES1 tracks several thousand individual particles in phase space. Thus, it effectively solves a Klimontovich-Dupree Equation which in the limit of a very large number of particles should converge to the equivalent problem of solving the Vlasov Eq. for a smooth $f(x,v,t)$.

It has been found that the temperature of thermal plasmas increases linearly with time. Self heating time τ_h is defined as the time taken for the thermal energy (T or v^2) of the system to double in value. A one dimensional electrostatic plasma model consisting of a

mobile electron species and immobile neutralizing ion background was used in this study. For this system, this is the time in which the average kinetic energy of an electron increases by $0.5\kappa T$. This increase in energy is numerical in origin and of a stochastic nature. It arises due to the fluctuations in the force due to the presence of finite grids in space and time. Therefore, the self heating time strongly depends on Δt and Δx . A Maxwellian velocity loader with first and second moment correction was used (Gitomer 1971).

Hockney (1971) empirically obtained the self-heating times for a two dimensional plasma with ions and electrons using $T_e = T_i$, $m_i / m_e = 64$ with further refinements by Hockney et al. (1974).

Quadratic spline weightings were added to the scope in this study and our results indicate that self heating times are longest for $v_t \Delta t / \Delta x \approx 3/2$ for NGP and $v_t \Delta t / \Delta x \approx 1/2$ for CIC and QS.

2. Choice of $\omega_p \Delta t$ in the determination of self heating time τ_h

Results of simulations carried out with ES1 for the typical growth in time of the thermal energy indicate that the energy increases linearly with time for $\omega_p \Delta t \leq 0.6$. This implies a random process. Therefore, this study will be restricted to this range of values of $\omega_p \Delta t$. The growth observed in thermal energy for larger values of $\omega_p \Delta t$ is like t^n with $n > 1$. This implies some other form of growth as yet unexplained. The same change in growth pattern was observed for the thermal energy for $\omega_p \Delta t \leq 0.6$ using both CIC-PIC and QS weightings. Thus, $\omega_p \tau_h$ was determined from similar histories with unity slope. In obtaining τ_h , the zero in time and the initial thermal energies were assigned to the beginning of growth linear in time.

3. Dependence of τ_h on $n_0 \lambda_D, n_0 (\lambda_D + \Delta x)$

The results of simulations for self-heating times are shown in Fig. 1 in units of electron plasma frequency $\omega_p \tau_h$ vs. $N_D = n_0 \lambda_D$ and $N_C + N_D = n_0 (\Delta x + \lambda_D)$ where n_0 is varying. Here N_C denotes the number of particles in a grid, spacing, N_D denotes the number of particles in a Debye length, n_0 is the particle density, Δx is the simulation grid size and λ_D is the Debye length.

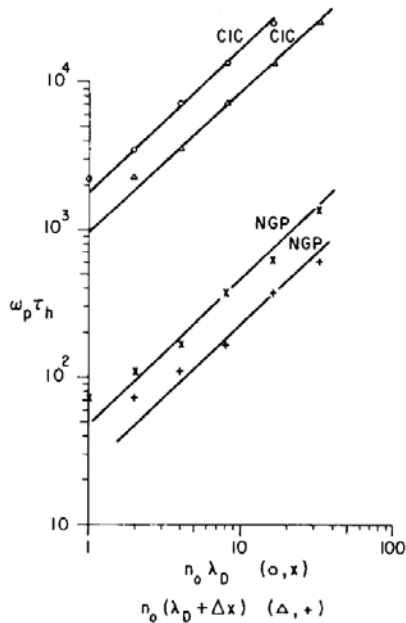


Fig. 1 Self heating times vs $n_0 \lambda_D, n_0 (\lambda_D + \Delta x)$ for NGP, CIC showing linear dependence

Fig. 2 shows the results of simulations for self-heating times plotted as electron plasma frequency $\omega_p \tau_h$ vs. $N_C + N_D$ for two different ratios of $\lambda_D / \Delta x = 0.5$ and $\lambda_D / \Delta x = 2.0$ for NGP. Similar results were observed for CIC.

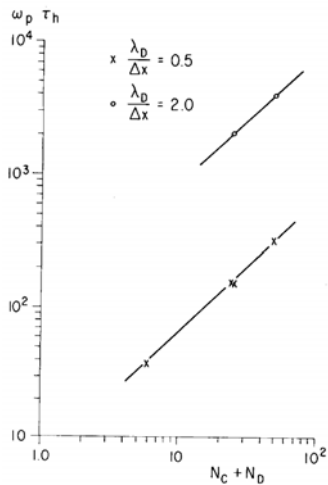


Fig. 2 Self heating times vs $N_C + N_D$ for different ratios of $\lambda_D / \Delta x$ indicating a linear dependence of $\omega_p \tau_h$ on $N_C + N_D$ for fixed $\lambda_D / \Delta x$ for NGP

4. Dependence of $\lambda_D / \Delta x$ and optimum choice of $v_t \Delta t / \Delta x$

The self heating times plotted as electron plasma

frequency $\omega_p \tau_h$ divided by $N_C + N_D$ vs. $\lambda_D / \Delta x$ are shown in Fig. 3, 4 and 5 for different values of $\omega_p \Delta t$. The dashed line drawn through the different graphs indicates $v_t \Delta t / \Delta x \approx 1.5$ for NGP and $v_t \Delta t / \Delta x \approx 0.5$ for CIC and QS. The longest heating times occur at about these values of $v_t \Delta t / \Delta x$.

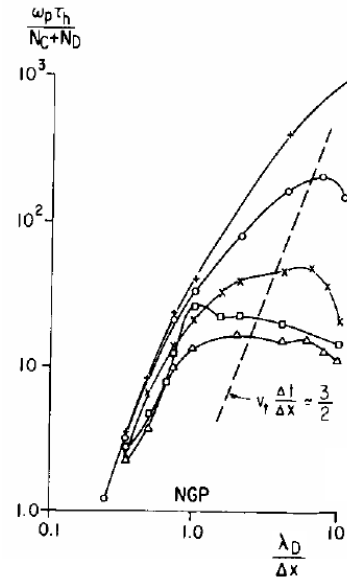


Fig 3. Self heating times divided by $N_C + N_D$ vs. $\lambda_D / \Delta x$ for several values of $\omega_p \Delta t$ for NGP

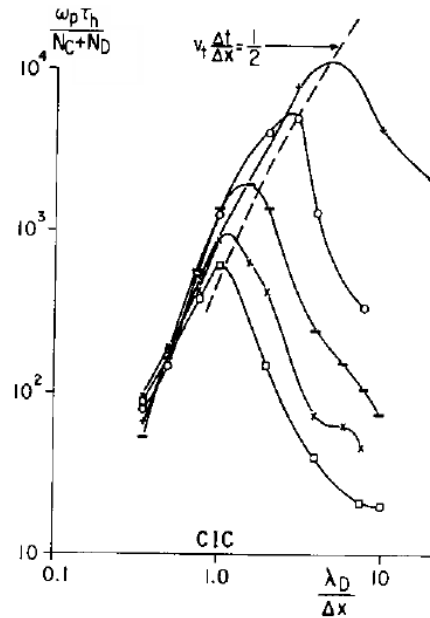


Fig 4 Self heating times divided by $N_C + N_D$ vs. $\lambda_D / \Delta x$ for several values of $\omega_p \Delta t$ for CIC

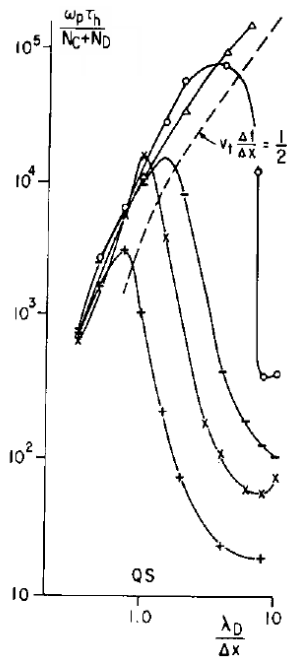


Fig 5 Self heating times divided by $N_C + N_D$ vs. $\lambda_D / \Delta x$ for several values of $\omega_p \Delta t$ for QS

5. Comparison of results and the gain of going to a higher order plasma simulation algorithm

As a comparison of the different algorithms studied, ratios of self-heating times of quadratic-spline (QS) to nearest-grid-point (NGP) and cloud-in-cell (CIC) to nearest-grid-point (NGP) are plotted in Figs. 6, 7 and 8 for $\omega_p \Delta t = 0.1, 0.2, 0.3$, respectively.

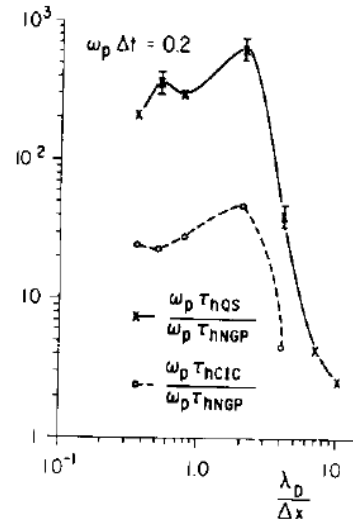
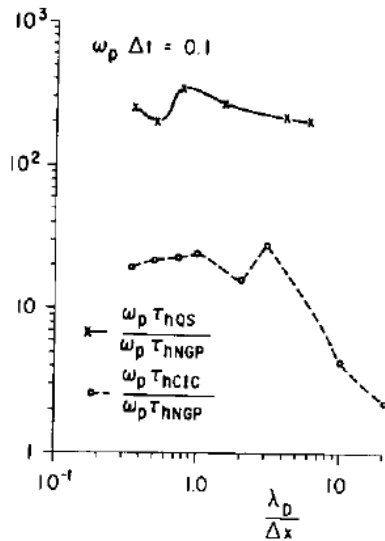
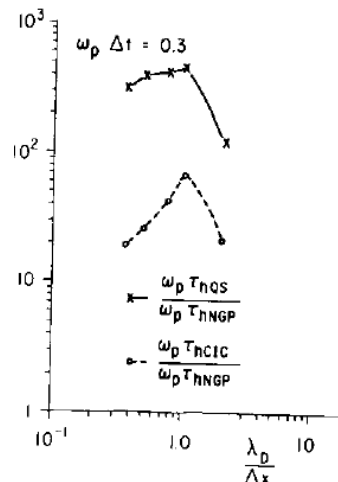


Fig. 6 Ratios of self-heating times vs. $\lambda_D / \Delta x$ for $\omega_p \Delta t = 0.1$

Fig. 7 Ratios of self-heating times vs $\lambda_D / \Delta x$ for $\omega_p \Delta t = 0.2$

Fig. 8 Ratios of self-heating times vs $\lambda_D / \Delta x$ for $\omega_p \Delta t = 0.3$

These figures indicate that CIC self heating times are as much as 70 times longer than NGP, and that the self heating times of QS is as much as 650 times longer than NGP. Such increases in heating times come at the expense of much longer computational times. Actual measurement of the cost of running the plasma simulations per time step on a computer at the Lawrence Livermore Laboratory show that $T=5, 11.6$ and 24 microseconds/particle/time step for NGP, CIC and QS, respectively. Hence, we need a measure of accounting for this cost. Therefore, the gain of using a higher order weighting scheme for the simulations can be defined as follows:



$$Gain = \frac{\text{increase in self heating time}}{\text{increase in computer simulation time}}$$

Going through the optimal path, we will get gains as shown in Table 1. Note that the gains of using higher order weightings, i.e. going from NGP to CIC or going from CIC to QS are roughly one order of magnitude, much less than the gains in the self heating times τ_h .

Table 1 –A comparison of gains in going to a higher order simulation algorithm indicating ratios of increase in self-heating times or reduced error in energy over increase in cost determined on a computer at the Lawrence Livermore Laboratory

$\omega_p \Delta t$	CIC/NGP Gain	QS/NGP Gain	QS/CIC Gain
0.1	11.9	72.9	6.1
0.2	20.8	135.4	6.5
0.3	30.4	91.6	3.0

6. Increase in gain due to k-space smoothing

The self heating times of one dimensional thermal plasma simulations can be considerably increased by smoothing the charge density in k-space. The smoothing factor used was simple Fourier space truncation, where all the modes beyond k_{last} are dropped as shown in Fig. 9.

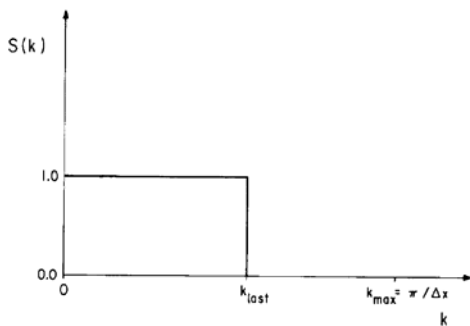


Fig 9. Smoothing factor used in k-space truncation

Fig. 10 shows the self-heating times plotted as electron plasma frequency $\omega_p \tau_h$ vs. k_{max} / k_{last} for the different schemes used by keeping everything fixed and varying k_{last} for each scheme. The gain in self-heating time due to k-space truncation is almost proportional to k_{max} / k_{last} for NGP and is close to but not quite proportional to $(k_{max} / k_{last})^2$ for CIC and $(k_{max} / k_{last})^3$ for QS. Thus, k-space truncation further increases the gain of CIC/NGP, QS/NGP and QS/CIS.

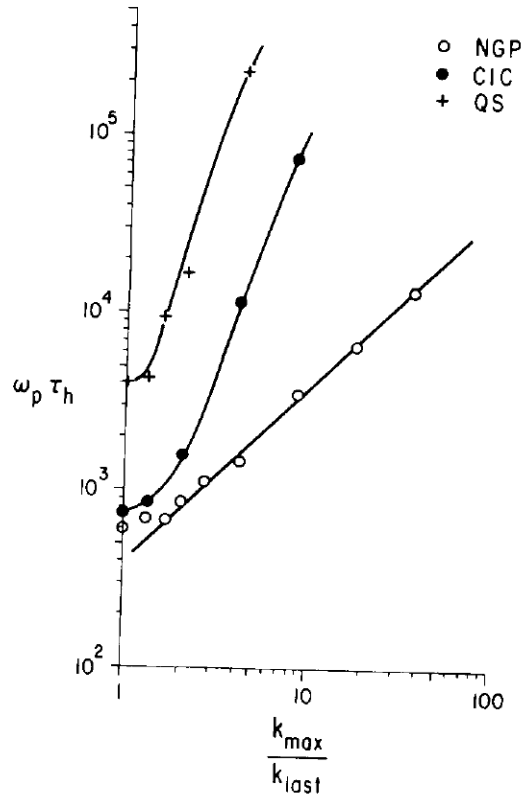


Fig. 10 Self-heating times plotted as electron plasma frequency $\omega_p \tau_h$ vs. k_{max} / k_{last} for NGP, CIC and QS.

7. Conclusion

The self heating times of one dimensional electrostatic plasma simulation were found to be the longest for $v_i \Delta t / \Delta x \approx 1.5$ for NGP and $v_i \Delta t / \Delta x \approx 0.5$ for CIC and QS. Roughly speaking, the quadratic spline self heating times are one order of magnitude longer than for cloud-in-cell, while cloud-in-cell self heating times are one order of magnitude longer than the nearest-grid-point algorithm, considering both the gain in heating times and the increased computational cost. Smoothing by Fourier space truncation considerably increases the self-heating times and this increase is roughly proportional to $(k_{max} / k_{last})^{n+1}$ where n is the order of the weighting scheme, i.e. n=0 for NGP, n=1 for CIC and n=2 for QS. Therefore, using a higher order algorithm is highly recommended, especially when k-space truncation is used.

Acknowledgement

I am highly indebted to Professor C. K. Birdsall for the support and advice for this study. I am also grateful of the Vice Chancellor of Research and Technology of the Ferdowsi University of Mashhad for the grant project.

References

- [1] Parker, S. E. (2002), "Nearest-grid-point interpolation in gyrokinetic particle-in-cell simulation," *Journal of Computational Physics*, Vol. 178, No. 2, pp. 520-532.
- [2] Li, Y., Wang, H., Liu, C., Zhang, D., Wang, J., Wang, Y., (2009), "2.5-dimensional electromagnetic particle-in-cell code-UNIPIC for high power microwave simulations," *Qiangjiguang Yu Lizishu High Power Laser and Particle Beams* 21 (12), pp. 1866-1870.
- [3] Birdsall, C. K., Langdon, A. B., (2005), *Plasma Physics via Computer Simulation*, Institute of Physics, Series in Plasma Physics, Taylor & Francis Group, 2005.
- [4] Gitomer, S. (1971), "Comments on numerical simulation of the Weibel instability in one and two dimensions," *Phys. Fluids*, Vol. 14, July 1971, pp.1591-1592.
- [5] Hockney, R. W., (1971), "Measurement of collisions and heating times in a two dimensional thermal plasma," *J. Comp. Phys.*, Vol. 8, Aug. 1971, pp.19-44.
- [6] Hockney, R. W., Goel, S. P., Eastwood, J. W. (1974), "Quiet high resolution computer models of a plasma," *Journal of Comp. Phys.*, Vol. 14, (Feb. 1974), pp.148-158.

3/1/2010