Eur. Phys. J. D **41**, 629–639 (2007)
DOI: 10.1140/epjd/e2006-00255-3

THE EUROPEAN PHYSICAL JOURNAL D

Modeling of clusters by a molecular dynamics model using a fast tree method

G.M. Petrov[a] and J. Davis[b]

Naval Research Laboratory, Plasma Physics Division, 4555 Overlook Ave SW, Washington, DC 20375, USA

Received 19 May 2006 / Received in final form 29 September 2006
Published online 22 November 2006 –

Abstract. The dynamics of clusters irradiated by a high-intensity ultrashort pulse laser has been studied using a fully relativistic three-dimensional Molecular Dynamics Model. A fast three-dimensional tree algorithm for computing the electrostatic force has been developed and compared with the conventional particle-particle method. The particle-particle method requires computation time, which scales as $O(N_p^2)$, and it is faster for small number of particles $N_p < 10^3$. In the opposite case of relatively large ensemble of particles $N_p > 10^3$, the preferred method is the tree algorithm whose computation time scales as $O(N_p \log N_p)$. The tree algorithm has been benchmarked against the particle-particle method for clusters composed of xenon and deuterium atoms and its accuracy and computation time have been analyzed. The optimum free parameter of the tree method has been determined to be $\theta \approx 0.5$. We addressed the effects of boundary conditions by studying the contribution of adjacent clusters to the total electromagnetic force exerted on individual particles. We found that the adjacent clusters play a minor role in the overall cluster dynamics.

PACS. 52.38.-r Laser-plasma interactions – 52.38.Ph X-ray, gamma-ray, and particle generation – 52.50.Jm Plasma production and heating by laser beams

1 Introduction

Theoretical studies of laser-cluster interaction have advanced considerably in the past ten years. Initially, the model of choice was based on hydrodynamic approach [1–3], but alternative models such as Monte Carlo classical particle-dynamics simulations [4,5], particle-in-cell (PlC) [6–13] and Molecular Dynamics (MD) [14–31] were employed as well. Both PIC and MD have been proven to be reliable and robust simulation tools and have been widely used for simulations of plasmas. In a MD simulation, the classical equations of motion for the positions, velocities, and accelerations of all particles are integrated in time using a finite-difference algorithm and the dynamical trajectories given by the Newton's equations of motion are calculated. In essence, both PIC and MD are used to advance particles by solving the particle equations of motion, which, for charged particles, requires the computation of the electromagnetic force acting on each particle. On the computational side, the main difference between them lies in the particular numerical implementation. In the MD approach the force on a particle is computed as a sum over individual two-body interactions. For a collection of N_p particles the direct evaluation of all pair inter-particle forces requires $O(N_p^2)$ operations. In the PIC method, the field is computed on a grid of cells, and the force on the particles is interpolated from the fields on the grid. The computation time scales linearly with the number of grid points, and, in general, leads to $O(N_p)$ operations. In reality, MD calculations are reasonable for only the smallest systems with $N \leq 10^3$, and for larger systems the problem poses severe numerical difficulties. The computational cost grows so quickly, that models with more than 10^5 particles become impractical. The considerably more CPU time devoted to calculate the inter-particle forces is the main limitation of the MD models.

The other difference between PIC and MD is the use of computational grid. PIC employs a grid to extrapolate charges from particles to grid nodes and forces from grid nodes to particles, while the MD does not require a grid at all. The grid imposes limitations such as artificial boundaries on the system of particles, even if such boundaries do not exist. Grids require spatial resolution: for example, it is customary for plasma simulations to resolve the Debye length. Sampling of fields and forces is another concern. Some portions of the calculation may be undersampled, while others are oversampled. As a result, various features involving sharp gradients such as shocks and fronts may be poorly resolved. The number of particles used in simulations is also grid-dependent: one must allow at least a

[a] e-mail: george.petrov@nrl.navy.mil
[b] e-mail: jack.davis@nrl.navy.mil

few particles per cell in order to ensure adequate statistics. Therefore, in one or two dimensions many cells and particles per cell can be readily used and adequate grid resolution achieved, but in three dimensions the necessary number of grid points and number of particles may become quite large. The grid adds to the overall complexity of the problem, as the force computation requires the solution of the Poisson equation. Grids are used as a computational convenience, but they themselves introduce other difficulties obscuring the aim of the calculation. In contrast, gridless methods offer a large dynamic range in ion charge and spatial resolution and do not impose geometrical restrictions on the system to be studied. Gridless methods are better suited when studying inhomogeneous systems with high-density contrast or large asymmetries. Therefore, an extension of MD models to large number of particles is needed to compete with PIC models.

Both MD models and PIC have been employed to model high-intensity laser cluster interactions. Unlike other problems related to laser-plasma or laser-solid interaction, the MD approach has a clear advantage for the problem at hand. The limited number of particles in a cluster (10^3–10^5), combined with the numerous advantages of gridless calculations makes MD the perfect simulation tool for laser cluster interactions. Modeling of rare gas clusters with $\sim 10^4$ atoms was recently done [29], but larger clusters containing $\sim 10^6$ atoms need to be modeled. As an example, we can point out the explosion of deuterium clusters relevant to table-top nuclear fusion. High fusion rate and neutron yield can be achieved only from large clusters, containing, perhaps, a million atoms. Following so many particles may be on the verge of or beyond the capability of present-day desktop computers. Not surprisingly, a remedy for the N_p^2 problem in MD has been sought. In 1986 Barnes and Hut [32] introduced a hierarchical tree algorithm, whose computation time scales with the number of particles as $O(N_p \log N_p)$. The force calculation is improved by lumping particles at "large" distance to a given individual macroparticle. The larger the distance, the larger the clumps of lumped particles. Thus, on average, the tree code achieves $O(\log N_p)$ operations per particle. But the gain in speed entails reduced accuracy of the force calculation due to lumping of particles. In principal, for large number of particles ($>10^3$) the tree algorithm offers a significant reduction in computing cost compared to the conventional particle-particle method at the expense of a small, but controllable error in the force computation. The N_p^2 nature of the problem in MD models resembles that of the Fourier transform. Before the Fast Fourier Transform (FFT) was invented, there was little interest in the Discrete Fourier Transform since it is inherently a N_p^2 problem. The FFT, which requires $O(N_p \log N_p)$ operations, changed dramatically the fate of the Discrete Fourier Transform since FFT can process large amount of (multidimensional) data on a digital computer. Analogously, the tree algorithm can boost the capabilities of MD models.

There are different variations of the original tree algorithm, in fact, fast tree algorithms have been created prior to that of Barnes and Hut [33]. We adopted the original Barnes and Hut algorithm because of its simplicity, flexibility, and clear and unambiguous construction. The objective of the present article is to present the hierarchical tree algorithm and its application to laser-cluster interactions. It should be mentioned that MD models using the tree algorithm have already been successfully applied to similar problems, such as laser-cluster [18,25,26,31] and laser-solid [34] interactions. In Section 2 we describe in detail the physical and computational issues involved in modeling clusters. Section 3 is devoted to the classical Barnes and Hut tree algorithm. Section 4 is left for results and discussions. The results refer to medium-size clusters (50–100 Å initial radius) composed of xenon and deuterium. Also in this section, the tree algorithm is compared to the particle-particle method and the accuracy and computation time of the tree algorithm have been analyzed.

2 Formulation of the problem

2.1 Initial cluster configuration

We consider a single rare gas cluster irradiated by a high intensity sub-picosecond laser pulse. The cluster has a spherical form with initial radius R_0. The cluster parameters are derived adopting the liquid drop model. The number of atoms per cluster is $N = (R_0/R_w)^3$, where $R_w = \sqrt[3]{3M/(4\pi\rho)}$ is the Wigner-Seitz radius and M and ρ are the mass and liquid density of the gas. In our idealized model clusters are located at equal intervals $R_{ic} \approx 20R_0$ [29]. The cluster is located at the origin of the coordinate system and the 3D computational box has equal sides extending from $-R_{ic}/2$ to $R_{ic}/2$. The simulations commence by initializing particles position and velocity. A sphere with radius R_0 is filled with tightly packed neutral particles, arranged in a random fashion. Outside the sphere is empty space extending up to the cluster boundaries $\pm R_{ic}/2$. The particles occupy a very small faction of the computational box, typically $\sim$1%.

For computational purposes we deal with macroparticles instead of individual particles. A macroparticle represents n identical particles (atoms, electrons or ions) with charge qn, mass mn and size $r_0 n^{1/3}$, where q, m and r_0 are the individual particle charge, mass and size. Different species may be lumped differently, i.e. may have different n, but all macroparticles of the same type have equal mass and charge. The parameter n is carefully chosen to satisfy two conditions. On one hand, it should be large enough so that the number of macroparticles in the system N_p is not too big. On the other hand, n must be limited to $\sim 10^2$ in order to prevent macroparticles from attaining excessively large charge resulting in too strong binary Coulomb interactions [29], which are unphysical by nature. An appropriate number is adopted from the trade-off between the two. We consider only two types of macroparticles: electrons and ions.

2.2 Relativistic equations of motion

The charged macroparticles move according to their relativistic equations of motion

$$\frac{d\vec{p}_i}{dt} = \vec{F}_i^{laser} + \vec{F}_i^{Coulomb} \tag{1a}$$

$$\vec{v}_i = \frac{\vec{p}_i/m_i}{\sqrt{1+|\vec{p}_i|^2/(m_i c)^2}} \tag{1b}$$

$$\frac{d\vec{r}_i}{dt} = \vec{v}_i. \tag{1c}$$

$\vec{F}_i^{laser}$ is the external force applied on particle i due to the electromagnetic field of the laser, $F_i^{Coulomb}$ is the total Coulomb force applied on particle i arising from all charged particles, $\vec{p}_i$, $\vec{r}_i$, $\vec{v}_i$, m_i and q_i are the relativistic momentum, coordinate, velocity, mass and charge of the ith macroparticle respectively, and c is the speed of light. Equation (1) refers only to charged particles: for simplicity we assume that neutral atoms are immobile.

The force has been artificially split in two parts: external and Coulomb. The first one is known in advance, while the second is a function of the particles position and changes dynamically during the calculations. The external electromagnetic force

$$\vec{F}_i^{laser} = q_i\left(\vec{E}(t) + \vec{v}_i \times \vec{B}(t)\right) \tag{2}$$

in equation (1a) originates from the laser. The laser pulse is defined by its wavelength λ and intensity $I(t) = I_0 \exp\left(-(t-t_0)^2/\tau^2\right)$, where I_0 is the peak laser intensity, and t_0 and τ are given parameters. The laser electric and magnetic fields $\vec{E} = (E_0(t)\cos(\omega t - ky),\ 0,\ 0)$ and $\vec{B} = (0,\ 0,\ -B_0(t)\cos(\omega t - ky))$ have amplitudes $E_0(t) = \sqrt{2I(t)/\varepsilon_0 c}$ and $B_0(t) = E_0(t)/c$, respectively. The parameter ε_0 is the permittivity of free space, $k = 2\pi/\lambda$ is the wave vector, and λ and ω are the laser wavelength and frequency, respectively.

The Coulomb force is the most computationally intensive part of the MD model. The Coulomb force exerted on macroparticle i from all other macroparticles j is [34]

$$\vec{F}_i^{Coulomb}(\vec{r}) = q_i \sum_{j\neq i} \frac{q_j}{4\pi\varepsilon_0 R_{ij}^3}\vec{r}_{ij}, \tag{3}$$

where $\vec{r}_{ij} = \vec{r}_i - \vec{r}_j$. Equation (3) does not use a pure Coulomb force since we are dealing with point charges and close encounters pose numerical problems. As is customary in MD simulations, the Coulomb law is modified by including the softening parameter r_s, which is assumed to be equal to the size of macroparticle i. With this modification the effective distance between particles i and j takes form $R_{ij} = \sqrt{|\vec{r}_{ij}|^2 + r_s^2}$. The simple expression (3) for the binary interaction between macroparticles has been chosen in order to make the computation of the Coulomb force for the particle-particle and tree methods compatible.

2.3 Boundary conditions

The boundary conditions may be open [28] or periodic [29,30]. For open boundary conditions no limitations on the particle positions are imposed and the particles can go anywhere in the free space. Periodic boundary conditions simulate an infinite medium. They require formal cluster boundaries (computational box), which are defined through the intercluster distance:

$$-R_{ic}/2 \le x \le R_{ic}/2, \tag{4a}$$

$$-R_{ic}/2 \le y \le R_{ic}/2, \tag{4b}$$

$$-R_{ic}/2 \le z \le R_{ic}/2. \tag{4c}$$

The aim of the periodic boundary conditions is to mimic the influence of other clusters adjacent to the cluster under consideration. This is achieved by (i) imposing that any macroparticle leaving the computational box reappears on the opposite side with the same velocity, and (ii) the electromagnetic field produced by the charged particles from adjacent clusters is added to the Coulomb and laser fields (details will be given in Sect. 3.2).

2.4 Cluster dynamics

The cluster dynamics involves a variety of elementary processes for free electrons, neutral atoms and charged ions. The elementary processes taken into account in our model can be defined compactly as follows:

$$\text{optical field ionization: } X^{k+} \overset{OFI}{\rightarrow} X^{(k+1)+} + e \text{ (inner)} \tag{5a}$$

$$\text{collisional ionization: } X^{k+} + e \text{ (inner)} \rightarrow X^{(k+1)+} + 2e \text{ (inner)} \tag{5b}$$

$$\text{outer ionization: } e \text{ (inner)} \rightarrow e \text{ (outer)} \tag{5c}$$

$$\text{electron recapture: } e \text{ (outer)} \rightarrow e \text{ (inner)}. \tag{5d}$$

Ionization is accounted for via optical field (5a) and collisional ionization (5b). The newly created electrons by both processes (5a) and (5b) are placed inside the cluster as inner electrons with randomly generated coordinates and zero initial velocity. The ion charge of all ions is increased equally by an appropriate amount. Processes (5c) and (d) are unique to clusters. Clusters have boundaries and some of the free electrons remain inside the cluster, while others leave their parent cluster, making it positively charged. "Inner electrons" are those within the cluster. Mathematically this condition is written as $|\vec{r}_k(t)| \le R(t)$, where $|\vec{r}_k|$ is the distance of the kth electron to the origin of the coordinate system and

$$R(t) = \sqrt{2\sum_{i=1}^{N}\left(x_i^2(t) + y_i^2(t) + z_i^2(t)\right)/N} \tag{6}$$

is the cluster radius. The sum is only over atoms or ions. The factor two is introduced to roughly match the cluster

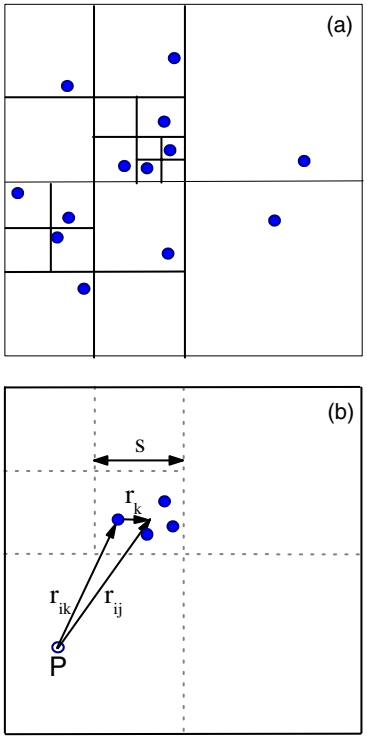

Fig. 1. (Color online) Schematics of the tree formation (a) and force computation (b). The distance to cell j participating in (8) is $d = |r_{ij}|$. The typical values of the free parameter are $0.1 \leq \theta \leq 1$.

radius at time $t = 0$. Electrons for which $|\vec{r}_k(t)| > R(t)$ are called "outer electrons". Outer ionization is the process in which an inner electron leaves the parent cluster and becomes an outer electron. The reverse process of electron recapture is also possible. It can occur when an outer electron is attracted to the positively charged cluster through Coulomb forces or due to cluster expansion. It should be noted that while processes (5a, 5b) create new electrons and increase the average ion charge, processes (5c, 5d) only reassign macroparticles as inner or outer electrons.

3 The hierarchical tree method

3.1 The Barnes and Hut algorithm

In 1986 Barnes and Hut [32] introduced a hierarchical tree algorithm for computing long-range (electromagnetic or gravitational) particle-particle forces. The tree method is of the type "divide-and-conquer" and it is based on a partition of the space enclosing the particles [34–39]. The whole space (root) is divided by cells of equal volume, called branches. In three-dimensions the number of branches is eight, while in two dimensions it is four. Then each cell is subdivided in the same manner until cells, called leaves, contain either one or zero particles. This process is illustrated in Figure 1a in two dimensions. With the division of space accomplished, the computation of forces on each particle is carried out. It starts with the root, working through the branches toward the leaves. If the cell dimension s and the distance d from the particle to the "center-of-mass" of the cell are such that

$$s/d \leq \theta, \tag{7}$$

the cell's lumped contribution is added to the force on the particle. Otherwise, the daughter cells of the branch containing particles are examined for contributions to the force. Ultimately, if none of the branches qualify for force contribution, the force is calculated from the leaves. The advantage of this approach is that a distant group of particles may require examining just a few branches, while close encounters are handled automatically by adding the force from individual particles (leaves). The free parameter θ can be fine tuned to meet the desired requirements for accuracy and computation time. The number of computations per macroparticle scales on average as $O(\log N_p)$, a great improvement compared to the direct particle-particle method, which scales as $O(N_p)$. The overall computation cost scales as $O(N_p \log N_p)$ [35,39], which makes it attractive and computationally feasible for large ensemble of particles.

The Coulomb force is computed from (3), but unlike the particle-particle method we now deal with cells, which may contain more than one particle. An appropriate method for handling this problem is the so-called Multipole Expansion Method. The aim of this method is to provide high accuracy by roughly accounting for the particle distribution in the cell.

The potential at point P, where macroparticle i is located, can be expressed as a sum of the potentials of all individual macroparticles in cell j (Fig. 1b)

$$\begin{aligned}\Phi_i^{Coulomb}(\vec{r}_{ij}) &= \sum_{k=1}^{n_j} \Phi(\vec{r}_{ik}) = \sum_{k=1}^{n_j} \Phi(\vec{r}_{ij} - \vec{r}_k) \\ &\cong \sum_{k=1}^{n_j} (\Phi(\vec{r}_{ij}) - \vec{r}_k \nabla\Phi(\vec{r}_{ij}) + ...)\end{aligned} \tag{8}$$

where $\vec{r}_{ik}$ is the vector from macroparticle i to the individual particle k in the cell and $\vec{r}_{ij} = (x_{ij}, y_{ij}, z_{ij})$ is the vector from macroparticle i to the "center of mass" of cell j, defined as $\vec{r}_{ij} = \vec{r}_i - \vec{r}_j$. After (or during) the tree construction the total charge $q_j = \sum_{k=1}^{n_j} q_k$, the "center of mass" $\vec{r}_j = \sum_{k=1}^{n_j} |q_k|\vec{r}_k / \sum_{k=1}^{n_j} |q_k|$ and dipole moment $\vec{d}_j = \sum_{k=1}^{n_j} q_k \vec{r}_k$ for each cell j containing n_j particles are calculated. The total Coulomb force on a macroparticle i calculated from all qualifying cells is $\vec{F}_i(\vec{r}) = -\nabla_r \Phi_i^{Coulomb}(\vec{r})$. Explicit expressions for the monopole, dipole and quadrupole terms can be found in reference [39]. We use only the first two terms of the multipole expansion, the monopole and dipole moments:

$$\vec{F}_i = q_i \sum_j \left(\frac{q_j}{4\pi\varepsilon_0 R_{ij}^3} \vec{r}_{ij} + \frac{3\vec{r}_{ij} \cdot \vec{d}_j}{4\pi\varepsilon_0 R_{ij}^5} \vec{r}_{ij} - \frac{1}{4\pi\varepsilon_0 R_{ij}^3} \vec{d}_j \right). \tag{9}$$

Similar to the particle-particle method, the effective distance to cell j takes form $R_{ij} = \sqrt{|\vec{r}_{ij}|^2 + r_s^2}$. Note that for each cell j the force is now expressed with respect to the "center of mass" of the corresponding cell. The first term in (9), the monopole contribution from cell j, includes the total charge of the cell and has the same form as that of an individual macroparticle. The second and

third terms, the dipole moment, accounts for the particle distribution in the cell through the parameter $\vec{d}_j$. As seen from equation (9) the inclusion of the dipole moment to the force computation needs only a little computational overhead and improves the accuracy of the electrostatic force at low computational cost.

The free parameter θ plays a key role in the computations. It is related to both the accuracy of the Coulomb force computation and computation time. It offers great flexibility, yet it must be carefully chosen. It is problem dependent, but the typical range of values for θ is well known. When the main issue is accuracy, θ must be decreased. High accuracy can be achieved for small $\theta = 0.1$–0.2. Values of θ of order of unity allow a significant reduction of computation time, if the inaccuracy can be tolerated. To compensate for the loss of accuracy, one can include higher order terms in the expansion (8). So both θ and the number of terms of the multipole expansion can be tuned to meet the desired accuracy and computation time. Such flexibility is undoubtedly a great advantage of the tree code.

3.2 Boundary conditions

We now return to the boundary conditions discussed in the previous section. Equation (9) can be used to evaluate the electrostatic force from neighboring clusters. The calculated force is approximate due to the truncation of expansion (8), but fortunately, even such a crude approximation provides sufficient accuracy. In what follows we will present a simple and straightforward way to calculate the electrostatic force from neighboring clusters. The only assumption we make is that a neighboring cluster is exactly the same as the cluster under consideration and that it is located at a distance equal to one intercluster distance R_{ic}. One can add the force from this cluster without partitioning it and looking into the detailed structure of particle distribution by using equation (9). The trick is to add the force from "root", but shifted one intercluster distance ($r_{ij} \rightarrow r_{ij} + R_{ic}$). The root properties are already known from the tree construction. Adding a neighboring cluster as one cell is seemingly inaccurate, but since the cluster is quasineutral, its charge is zero ($q_{j=root} \equiv 0$) and the leading term drops out. Only the dipole moment of the expansion contributes to the force, which decreases very rapidly with increasing distance from point P. Therefore, the contribution from neighboring clusters is expected to be relatively small. The computational cost of including the boundary conditions is discussed in Section 4, but it is obviously marginal since the properties of the "root" have already been calculated.

3.3 Pros and cons of the tree algorithm

The tree construction is unique and unambiguous, it requires no assumption about particle distribution and can be constructed at each time step. The total number of nodes (branches plus leaves) is proportional to the number of macroparticles N_p participating in the computations. In our simulations the number of nodes is $\sim 1.5 N_p$. The computed force is not exact, but the algorithm allows the error to be estimated and analyzed. The error can be controlled via the free parameter θ and the number of terms of the multipole expansion.

The program realization of the tree algorithm is not very complicated, particularly if using a computer language such as C++ or FORTRAN 90, which allow recursion. The tree construction and traversal can be neatly organized using recursive subroutines. Non-recursive implementation is also possible. Details of the tree algorithm, error analysis and computer implementation are given elsewhere [39]. It should be pointed out that the tree algorithm can be parallelized and thus used on massively parallel computers.

4 Results and discussions

In this section results both for the particle-particle and tree method are presented and compared. Our primary goal is to study the impact of the free parameter θ on the accuracy and computation time of the tree algorithm. Numerical experiments with different values of θ showed that with increasing θ the computation time decreases, but the accuracy deteriorates. For small θ only small branches containing few particles are added to the Coulomb force (9). This results in high accuracy, but since more branches must be included, the computation time increases as well. In the limiting case $\theta \rightarrow 0$ the tree method approaches the accuracy of the particle-particle method, but it exceeds it in time because of the additional computational overhead of the tree construction and traversal. It is certainly not wise to use too small a value of θ, since the advantage of the tree method to use large chunks of particles is lost. Intermediate values of θ (~ 0.5) sacrifice a little bit of the accuracy, but improve significantly the computation time. Larger values of θ, of the order of unity, would be even more advantageous in terms of computation time, but they may introduce significant error in the force calculation. This is to be expected, since for $\theta = 1$ the force is calculated at a distance equal to the cell size and the particle distribution inside the cell may greatly affect the result. In general, one should choose the largest value of θ, for which the error can be tolerated. Studies of gravitational problems and some limited studies of plasma physics problems indicate that θ should be between 0.1 and 1. The Coulomb force is a long-range force just as the gravitational one, and we expect θ to be in the same range. We performed sets of calculations for different values of θ between 0.25 and 1 and compared with the exact (particle-particle) solution. The particular values of θ we chose are 0.25, 0.5 and 1. We applied the model to medium size xenon and deuterium clusters. The latter is of interest for modeling fusion reactions and neutron yield in D-D processes. The laser and cluster parameters, as well as useful computational parameters are given in Table 1. Details of the cluster dynamics for xenon clusters at the same conditions have been recently published [29].

Table 1. Laser, cluster and computational parameters for Figures 2–12.

	xenon	deuterium
Cluster parameters		
Cluster radius R_0 [Å]	50	100
Wigner-Seitz radius R_w [Å]	2.73	1.70
Intercluster distance R_{ic} [Å]	1000	2000
Number of atoms per cluster N	6144	2.04×10^5
Laser parameters		
Peak laser intensity I_0 [W/m^2]	1×10^{20}	1×10^{20}
Peak laser electric field [V/m]	2.8×10^{11}	2.8×10^{11}
Laser wavelength [nm]	800	800
Laser pulse duration [fs]	400	400
Laser pulse center [fs]	200	200
Laser FWHM [fs]	125	125
Computational parameters		
Time step dt [attosecond]	1	0.5
Number of ion/atom macroparticles	100	500
Number of ions/atoms per macroparticle n	61.4	408
Size of an ion/atom r_0 [Å]	1	1
Size of an ion/atom macroparticle [Å]	3.9	7.3
Number of electron macroparticles	860	407
Number of electrons per macroparticle n	100	500
Size of an electron r_0 [Å]	1	1
Size of an electron macroparticle [Å]	4.6	8.0
Total number of macroparticles N_p	960	907

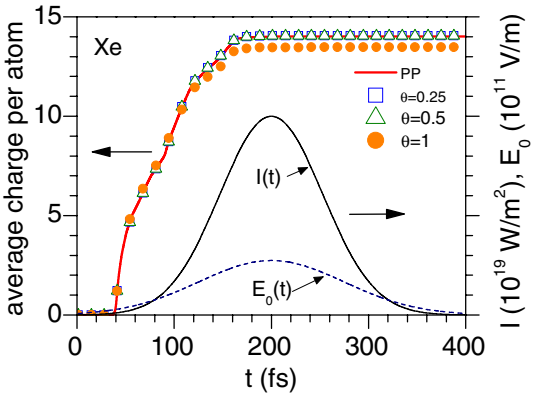

Fig. 2. (Color online) Average charge per atom vs. time for Xe clusters calculated with the particle-particle method and tree algorithm with $\theta = 0.25$, $\theta = 0.5$ and $\theta = 1$. The conditions are given in Table 1.

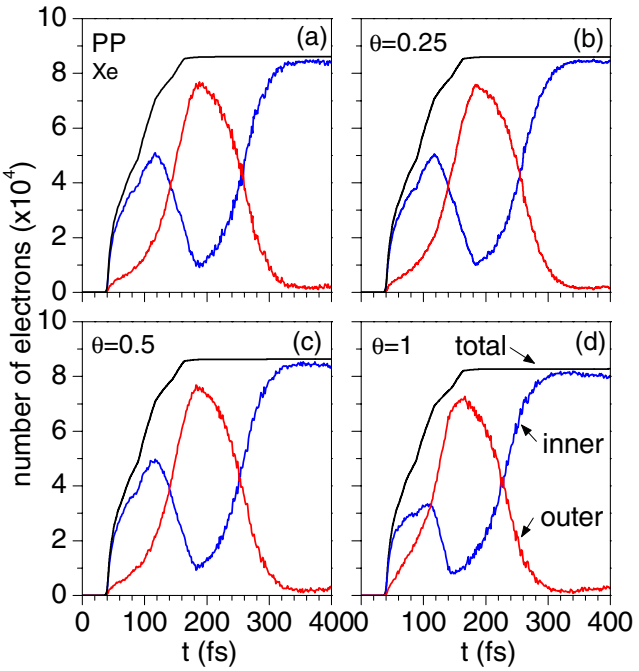

Fig. 3. (Color online) Number of inner electrons, outer electrons and total number of free electrons for Xe clusters vs. time. Particle-particle method (a); tree algorithm with $\theta = 0.25$ (b); $\theta = 0.5$ (c); and $\theta = 1$ (d).

4.1 Xenon

The laser intensity and the envelope of the laser electric fields are shown in Figure 2. The averaged charge per atom, also shown in Figure 2, is zero up to time $t < 40$ fs. There are no ionizations as the laser intensity is too low. For later times optical field and collisional ionization increase the averaged charge per atom. The results for all parameters of θ (symbols), even for $\theta = 1$, are in excellent agreement with the exact solution (solid line). The production of new electrons is well represented for relatively large values of the parameter θ. This is illustrated in Figure 3, which displays the total number of electrons, the number of inner and outer electrons for the same values of θ. For $\theta \leq 0.5$ the results from the tree simulations are identical to the exact ones. For $\theta = 1$ the number of inner electrons slightly deviates from the exact solution. Three-dimensional snapshots of

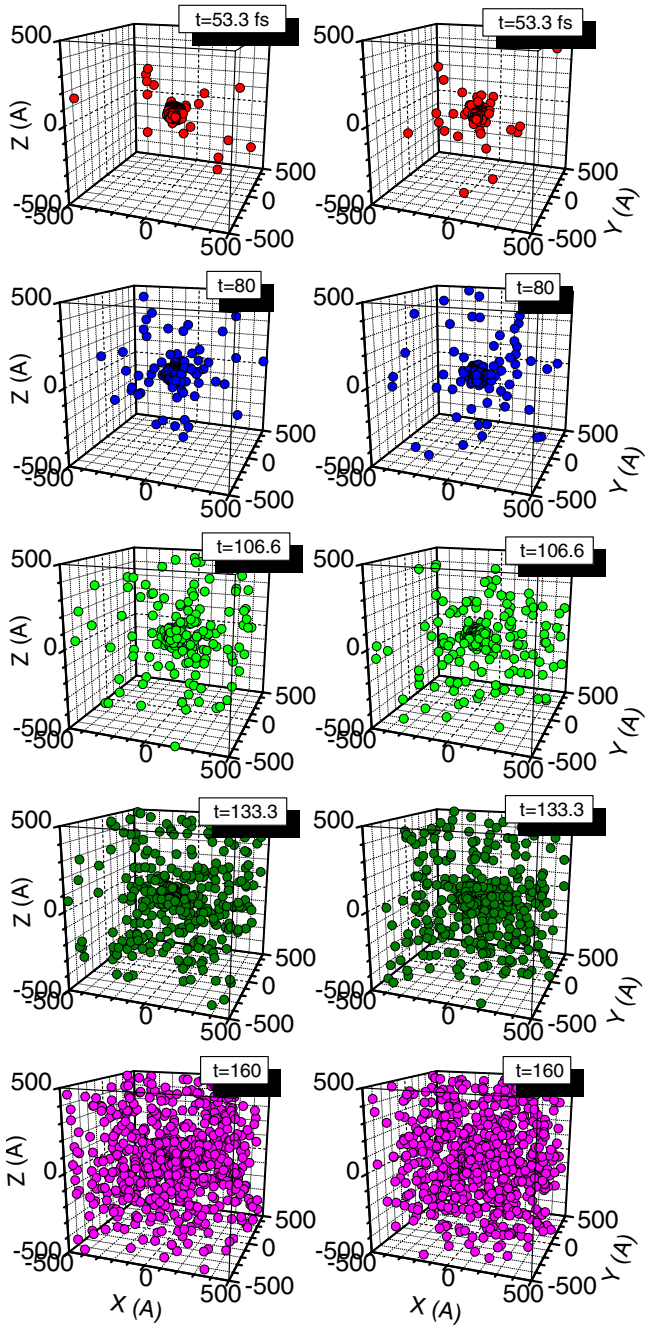

Fig. 4. (Color online) Positions of the electron macroparticles calculated with the particle-particle method (left) and tree algorithm with $\theta = 0.5$ (right).

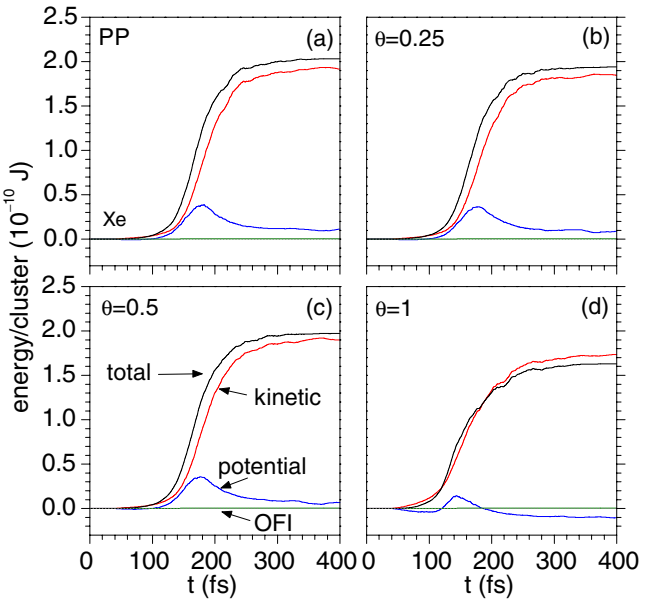

Fig. 5. (Color online) Energy balance terms for Xe clusters calculated with the particle-particle method (a); tree algorithm with $\theta = 0.25$ (b); $\theta = 0.5$ (c); and $\theta = 1$ (d).

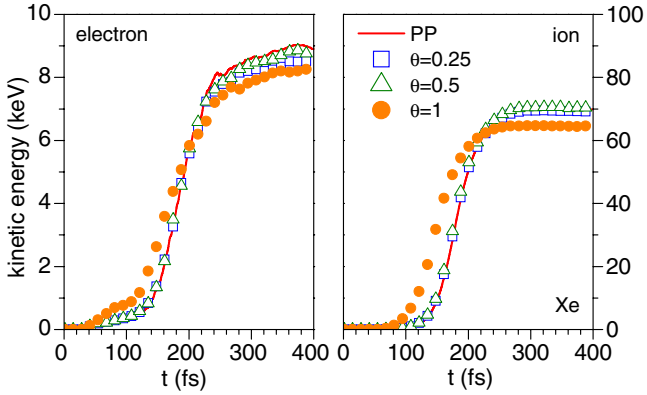

Fig. 6. (Color online) Mean electron energy (a) and ion energy (b) for Xe clusters vs. time calculated with the particle-particle method and tree algorithm with $\theta = 0.25$, $\theta = 0.5$ and $\theta = 1$.

the electron macroparticles positions calculated with the particle-particle method and tree algorithm with $\theta = 0.5$ are given in Figure 4. Both simulation runs have been done with exactly the same input parameters, including the initial distribution of macroparticles discussed in Section 2.1. It is interesting to observe that the tree algorithm leads to slightly different particle trajectories than the (exact) particle-particle method. This is the case for only a small fraction of the macroparticles, but the occasional individual deviation from the exact trajectories has limited impact on the "global" properties (Figs. 3, 5–7). The energy balance is of paramount importance. It provides valuable information not only for the amount of energy absorbed, but how this energy is partitioned among various channels. The absorbed energy per cluster is denoted in Figure 5 as "total". The tree algorithm is very accurate for values of the fitting parameter not exceeding 0.5. For $\theta = 1$ neither the absorbed energy, nor the kinetic and potential energies are accurately represented by the tree algorithm. In particular, the potential energy is slightly negative, while the exact solution is slightly positive. This is a consequence of the poor representation of the Coulomb forces by the tree method when the field is calculated at a distance comparable to the cell size. Most of the energy ends up as kinetic energy of both electrons and ions. It would be of interest to compare the accuracy of the tree algorithm for electrons and ions separately. Both particles interact through

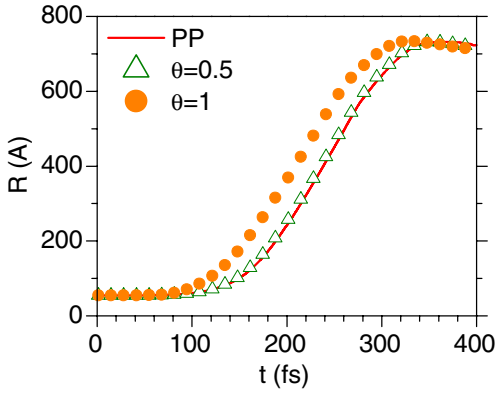

Fig. 7. (Color online) Cluster radius (6) vs. time for Xe clusters calculated with the particle-particle method and tree algorithm with $\theta = 0.5$ and $\theta = 1$.

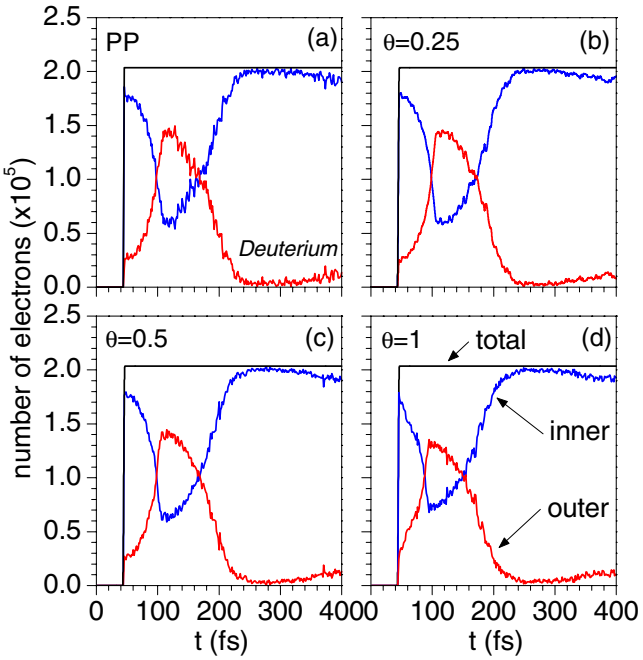

Fig. 8. (Color online) Number of inner electrons, outer electrons and total number of free electrons for D clusters vs. time. Particle-particle method (a); tree algorithm with $\theta = 0.25$ (b); $\theta = 0.5$ (c); and $\theta = 1$ (d).

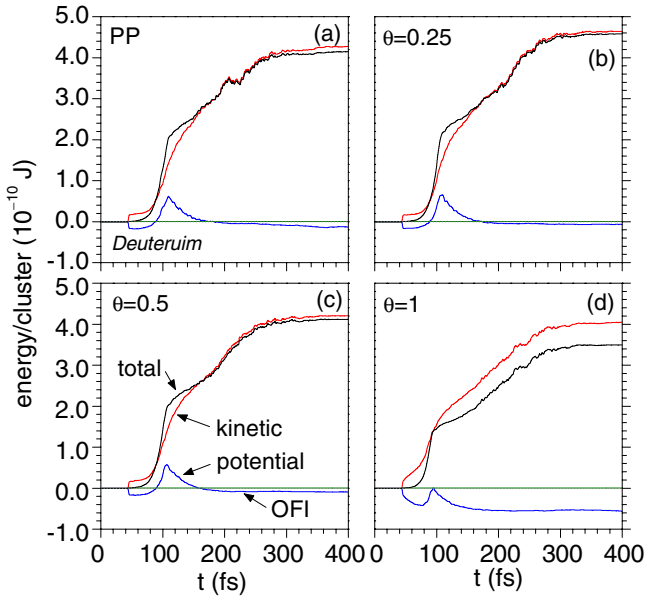

Fig. 9. (Color online) Energy balance terms for D clusters calculated with the particle-particle method (a); tree algorithm with $\theta = 0.25$ (b); $\theta = 0.5$ (c); and $\theta = 1$ (d).

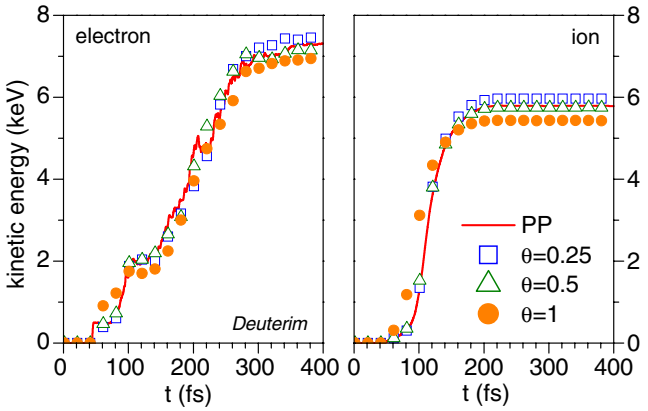

Fig. 10. (Color online) Mean electron energy (a) and ion energy (b) for Xe clusters vs. time calculated with the particle-particle method and tree algorithm with $\theta = 0.25$, $\theta = 0.5$ and $\theta = 1$.

Coulomb forces, but the ions are much heavier and sluggish. The impact of θ on both the mean electron and ion energies (Fig. 6) is similar to those in the previous figures: $\theta = 0.5$ provides fairly good accuracy, while for $\theta = 1$ the accuracy deteriorates. The last cluster parameter under consideration is the time evolution of the cluster radius, which is shown in Figure 7. The cluster retains its initial radius for ∼150 fs, but then it rapidly expands due to the large amount of absorbed energy, which results in large internal pressure. Generally, the laser energy is deposited into the cluster on a sub-ps time scale, much shorter than it can be dissipated by thermal conduction. An overheating and large pressure gradient is developed, which causes the cluster expansion. The cluster expansion is aided by a Coulomb explosion; the net positive charge of the cluster causes the ions to repel each other and the ensemble of ions expands. The contribution of both mechanisms (hydrodynamics and Coulomb explosion) is comparable. The results for the largest value of θ we used, $\theta = 1$, are satisfactory.

4.2 Deuterium

The results for deuterium are shown in Figures 8–11. Figure 8 displays the total number of electrons, the number of inner and outer electrons for the particle-particle method and $\theta = 0.25,\ 0.5,\ 1$. Figure 9 displays the energy balance,

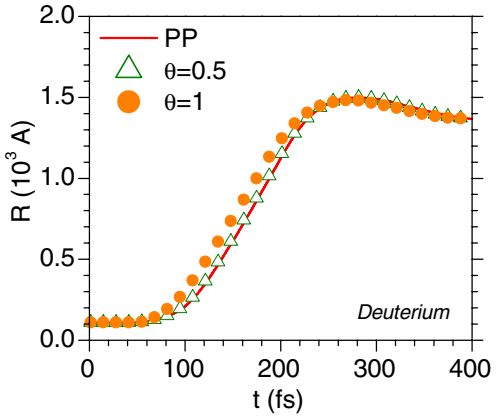

Fig. 11. (Color online) Cluster radius (6) for D clusters vs. time calculated with the particle-particle method and tree algorithm with $\theta = 0.5$ and $\theta = 1$.

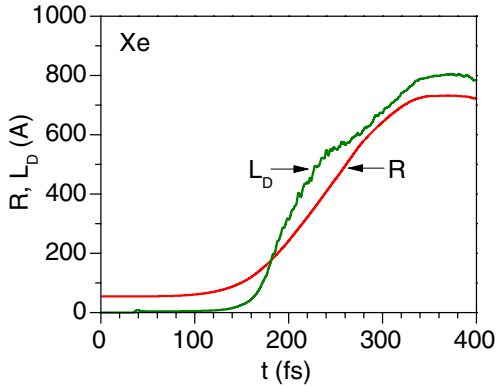

Fig. 12. (Color online) Debye length and cluster radius vs. time for Xe clusters.

Figure 10 displays the mean electron and ion energies, and Figure 11 displays the cluster radius. Qualitatively, the variation of θ produces the same effect: the results for $\theta \leq 0.5$ are accurate, while those for $\theta = 1$ have somewhat limited accuracy. However, depending on the application, even results with $\theta = 1$ may be acceptable. For example, the interest in deuterium clusters is primarily for studying fusion processes, and as seen in Figure 10, the mean ion energy at the end of the laser pulse for $\theta = 1$ is accurate to within ∼10%.

4.3 Impact of the boundary conditions

The method of incorporating boundary conditions was briefly discussed in the previous section. From a physical standpoint, the boundary conditions treat adjacent clusters as a source of additional electromagnetic field, which must be added to the laser field and the field generated inside the cluster. First, with the help of Figure 12 we shall make a quick quantitative analysis of the contribution from neighboring clusters. The analysis is done for Xe clusters. During the cluster expansion the Debye length is comparable to the cluster radius. Since the Coulomb force dies on a distance comparable to the Debye length, the

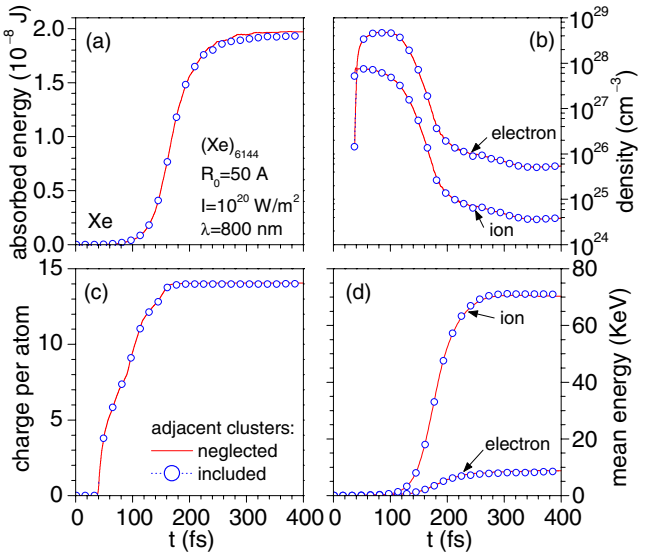

Fig. 13. (Color online) Absorbed energy per cluster (a), electron and ion density (b), average charge per atom (c) and mean electron and ion energy (d) vs. time for Xe clusters with initial radius of 50 Å, subject to laser radiation with intensity 10^{16} W/cm^2 and wavelength 800 nm. Computation algorithm: tree method with $\theta = 0.5$, solid lines: impact of adjacent clusters neglected, open symbols: impact of adjacent clusters included.

electrostatic field of individual clusters is well "shielded" and has negligible impact on its neighbors. Therefore, one would expect that the field produced by charged particles from adjacent clusters is small compared to the field generated inside the cluster. To verify this statement, we performed an additional computation in which the electrostatic field from adjacent clusters is added to the Coulomb and laser fields as discussed in Section 3.2. Though we do not compare the fields directly, we derive information regarding the impact of adjacent clusters by comparing some fundamental cluster parameters. The absorbed energy per cluster, electron and ion density and mean energy differ by ∼5%, and the charge per atom by ∼1–2%. From Figure 13 we conclude that the contribution of adjacent clusters to the total electromagnetic field is indeed small and can be omitted.

4.4 Analysis of the computation time

We made a series of runs for xenon clusters with different number of macroparticles N_p, varying from ∼500 to ∼3500. We verified that the computation time of the particle-particle method scales as $O\left(N_p^2\right)$, while that for the tree method scales as $O\left(N_p \log N_p\right)$ (Fig. 14). The break is at $N_p \approx 1000$, about the same as found by other authors [38,39]. The reason for the break location is well-known: the tree algorithm is indeed more efficient in calculating the forces since the number of cells used for computing the force is on average less than the number of macroparticles, but the overhead involved in

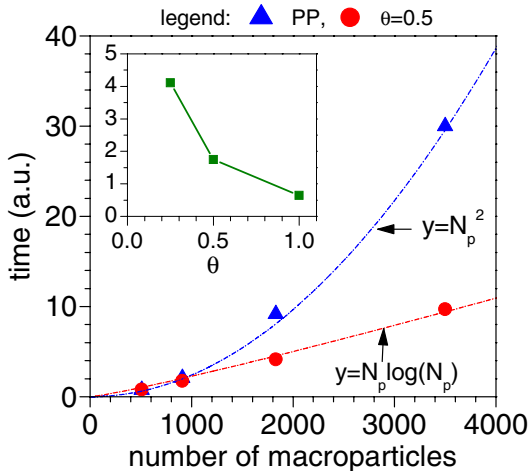

Fig. 14. (Color online) Computation time for xenon clusters vs. number of particles N_p. Particle-particle method — full triangles, tree method with $\theta = 0.5$ — full cycles. The insert shows the computation time vs. θ.

the tree construction and traversal, and computing the dipole moments can be fairly large. The tree algorithm becomes more efficient than the direct particle-particle force summation for number of macroparticles $N_p > 1000$. Given the small time step (∼1 attosecond), even today the computation time for $N_p \approx 1000$ is a challenge, but increase of computer power and/or multithread programming will definitely favor the tree method and its variations. The computation time in our simulations scales as $1/\theta$ (Fig. 14). From Figures 2–11 we determined the optimum value of the parameter θ by comparing the basic dynamic properties of the cluster such as ionization, energy absorption, and cluster expansion. We conclude that clusters can be successfully modeled using the tree algorithm without loss of accuracy for values of the fitting parameter $\theta \approx 0.5$.

We tracked the computation time spent in various sections of the tree code. For Xe clusters with $N_p \approx 10^3$ macroparticles the bulk of the computation time (∼89%) is used for computing the Coulomb force (9). The fraction of time spent in other sections of the code is as follows: tree construction and traversal — 4.5%, boundary conditions — 2.5%, auxiliary computations related to particles (mean electron and ion energy, Electron and Ion Energy Distribution Functions, etc.) — 3%, initialization and file output — 1%.

5 Conclusion

A fast three-dimensional tree algorithm for computing the electrostatic force acting upon charged particles in laser-cluster interactions has been developed. The method has been employed as part of a fully relativistic three-dimensional Molecular Dynamics Model. Results from the tree algorithm have been compared with the conventional particle-particle method for clusters composed of xenon and deuterium. The accuracy and computation time of the tree algorithm has been analyzed. We found that for large numbers of macroparticles, exceeding one thousand, the tree algorithm is computationally advantageous. The optimum free parameter of the tree method has been determined to be $\theta \approx 0.5$ for different gases (with high and low Z), cluster sizes and laser intensities, both relativistic and non-relativistic. We addressed the effects of boundary conditions by studying the contribution of adjacent clusters to the total electromagnetic force exerted on individual particles. We found that the adjacent clusters play a minor role in the overall cluster dynamics.

We would like to thank Quentin Saulter and ONR for their support. Also, a portion of the work was supported by NRL under the 6.1 Basic Physics Program.

References

1. T. Ditmire, T. Donnelly, A.M. Rubenchik, R.W. Falcone, M.D. Perry, Phys. Rev. A **53**, 3379 (1996)
2. H.M. Milchberg, S.J. McNaught, E. Parra, Phys. Rev. E **64**, 056402 (2001)
3. E. Parra, I. Alexeev, J. Fan, K.Y. Kim, S.J. McNaught, H.M. Milchberg, J. Opt. Soc. Am. B **20**, 118 (2003)
4. C. Rose-Petruck, K.J. Schafer, K.R. Wilson, C.P.J. Barty, Phys. Rev. A **55**, 1182 (1997)
5. K. Ishikawa, T. Blenski, Phys. Rev. A **62**, 063204 (2000)
6. T. Esirkepov, R. Bingham, S. Bulanov, T. Honda, K. Nishihara, F. Pegoraro, Laser Part. Beams **18**, 503 (2000)
7. M. Eloy, R. Azambuja, J.T. Mendonca, R. Bingham, Phys. Plasmas **8**, 1084 (2001)
8. M. Eloy, R. Azambuja, J.T. Mendonca, R. Bingham, Phys. Scripta **T89**, 60 (2001)
9. Y. Kishimoto, T. Masaki, T. Tajima, Phys. Plasmas **9**, 589 (2002)
10. P. Greschik, H.-J. Kull, Laser Part. beams **22**, 137 (2004)
11. T. Taguchi, T.M. Antonsen Jr, H.M. Milchberg, Phys. Rev. Lett. **92**, 205003 (2004)
12. C. Jungreuthmayer, M. Geissler, J. Zanghellini, T. Brabec, Phys. Rev. Lett. **92**, 133401 (2004)
13. Y. Fukuda, Y. Kishimoto, T. Masaki, K. Yamakawa, Phys. Rev. A **73**, 031201(R) (2006)
14. T. Ditmire, Phys. Rev. A **57**, R4094 (1998)
15. I. Last, J. Jortner, Phys. Rev. A **60**, 2215 (1999); I. Last, J. Jortner, Phys. Rev. A **62**, 013201 (2000)
16. J. Kou et al., J. Chem. Phys. **112**, 5012 (2000)
17. H. Wabnitz et al., Nature **420**, 482 (2002)
18. U. Saalmann, J.-M. Rost, Phys. Rev. Lett. **91**, 223401 (2003)
19. C. Siedschlag, J.-M. Rost, Phys. Rev. Lett. **89**, 173401 (2002); C. Siedschlag, J.-M. Rost, Phys. Rev. Lett. **93**, 043402 (2004)
20. I. Last, J. Jortner, J. Chem. Phys. **120**, 1336 (2004); I. Last, J. Jortner, J. Chem. Phys. **120**, 1348 (2004); I. Last, J. Jortner, J. Chem. Phys. **121**, 3030 (2004)
21. D. Bauer, Appl. Phys. B **78**, 801 (2004)
22. D. Bauer, J. Phys. B: At. Mol. Opt. Phys. **37**, 3085 (2004)
23. Z. Jurek, G. Oszlanyi, G. Faigel, Europhys. Lett. **65**, 491 (2004)

24. A. Heidenreich, I. Last, J. Jortner, Eur. Phys. J. D **35**, 567 (2005)
25. C. Jungreuthmayer, L. Ramunno, J. Zanghellini, T. Brabec, J. Phys. B **38**, 3029 (2005)
26. U. Saalmann, J.-M. Rost, Eur. Phys. J. D **36**, 159 (2005)
27. S.V. Fomichev, D.F. Zaretsky, D. Bauer, W. Becker, Phys. Rev. A **71**, 013201 (2005)
28. G.M. Petrov, J. Davis, A.L. Velikovich, P.C. Kepple, A. Dasgupta, R.W. Clark, A.B. Borisov, K. Boyer, C.K. Rhodes, Phys. Rev. E **71**, 036411 (2005)
29. G.M. Petrov, J. Davis, A.L. Velikovich, P. Kepple, A. Dasgupta, R.W. Clark, Phys. Plasmas **12**, 063103 (2005)
30. G.M. Petrov, J. Davis, Phys. Plasmas **13**, 033106 (2006)
31. U. Saalmann, J. Mod. Opt. **53**, 173 (2006)
32. J. Barnes, P. Hut, Nature **324**, 446 (1986)
33. A.W. Apel, SIAM J. Sci. Stat. Comput. **6**, 85 (1985)
34. P. Gibbon, F.N. Beg, E.L. Clark, R.G. Evans, M. Zepf, Phys. Plasmas **11**, 4032 (2004)
35. L. Hernquist, Comp. Phys. Comm. **48**, 107 (1988)
36. J. Ambrosiano, L. Greengard, V. Rokhlin, Comp. Phys. Comm. **48**, 117 (1988)
37. J. Makino, J. Comp. Phys. **87**, 148 (1990)
38. S.L.W. McMillan, S.J. Aarseth, Astrophys. J. **414**, 200 (1993)
39. S. Pfalzner, P. Gibbon, *Many Body Tree Methods in Physics* (Cambridge University Press, New York, 1996)