Computer Physics Communications 182 (2011) 1919–1923

Contents lists available at ScienceDirect

Computer Physics Communications

www.elsevier.com/locate/cpc

Removal of spurious self-interactions in particle–mesh methods

V. Ballenegger [a,*], J.J. Cerdà [b], C. Holm [b]

[a] Institut UTINAM, Université de Franche-Comté, UMR 6213, 16, route de Gray, 25030 Besançon cedex, France
[b] Institute for Computational Physics, Universität Stuttgart, Pfaffenwaldring 27, 70569 Stuttgart, Germany

ARTICLE INFO

Article history:
Received 29 July 2010
Accepted 28 January 2011
Available online 15 February 2011

Keywords:
Particle–mesh methods
Accuracy
Conservation law
SPME
P3M

ABSTRACT

We derive an analytic formula for subtracting the spurious self-forces in particle–mesh methods that use the analytical differentiation scheme, such as the Smooth Particle Mesh Ewald (SPME) method and the Particle–Particle Particle–Mesh (P3M) method with analytical differentiation. The impact of the self-forces on the accuracy of the particle–mesh methods is investigated, and it is shown that subtracting them can improve the accuracy of the calculation for some choices of the method's parameters. It is also suggested to subtract exactly the approximate, mesh-computed, self-energy of each particle, replacing them by the exact value. Subtracting in this way the self-energy and self-force of each particle not only improves the accuracy, but also reduces the violation of momentum and energy conservation in particle–mesh methods with analytical differentiation.

1. Introduction

The computation of long-range forces in a many-particle system is a demanding task that scales a priori as $O(N^2)$ with the number N of particles. Particle–mesh methods reduce the complexity of this problem to $O(N \log N)$ by discretizing the system onto a mesh and by taking advantage of the Fast Fourier Transform (FFT) algorithm to solve the Poisson equation in Fourier space, where it reduces to a simple multiplication by the Green function. For higher efficiency, the pair interaction is moreover decomposed into a short-range and a long-range parts, with the short-range interactions computed in real-space, as in the Ewald method [1]. The Particle–Particle Particle–Mesh (P3M) method [2] and the Smooth Particle Mesh Ewald (SPME) method [3] are two such particle–mesh methods that are widely used to compute Coulomb or gravitational forces in computer simulations.

In particle–mesh methods, various routes have been proposed to compute forces from the mesh-based potential:

- differentiation in real space by finite differences,
- differentiation in real space using the exact gradient of the assignment function used to interpolate the charge density onto/from the mesh,
- differentiation in Fourier space by multiplying the potential by $i\boldsymbol{k}$.

The second route is referred to as the analytical differentiation (AD) scheme, and it is used in the optimal energy-conserving scheme discussed in Hockney and Eastwood's book [2]. The third route, called the $i\boldsymbol{k}$-differentiation or force-interpolation scheme, corresponds to calculate the electric (or gravitational) field at each mesh point, and to interpolate this vectorial "force" field back to the particle positions. The finite-differences scheme was favored by Hockney and Eastwood in their implementation of the P3M method, but the other schemes, implemented first in the PME method [4] (for $i\boldsymbol{k}$-differentiation) and in the SPME method [3] (for AD differentiation), proved to be efficient and were later also implemented in the P3M method [5,6].

Each scheme has its own merits and drawbacks. The $i\boldsymbol{k}$-differentiation scheme is the most accurate one but also the most computationally expensive one, as it requires 3 inverse FFTs to transform the vectorial electric field mesh back to real space (compared to only one inverse FFT of the potential mesh in the other schemes). The higher accuracy of this scheme permits however the use of a coarser mesh, and accepting the two additional Fourier transforms can be competitive on desktop computers (this route becomes however less attractive in parallel distributed-memory computers because of the global communications required by the FFT's) [7]. As shown by the thorough analysis of particle–mesh methods in Hockney and Eastwood's book, the $i\boldsymbol{k}$-differentiation and finite-differences schemes conserve momentum but not energy, while analytical differentiation conserves energy (in the limit of small time steps) but not momentum [2, §5.3.3, §5.5 and §7.6]. In the AD scheme, a correction to all forces must be applied to conserve at least the center-of-mass momentum [3]. This correction has unfortunately the collateral effect of breaking the exact conservation of energy (a mass-weighted correction reduces this drawback but does not remove it entirely) [8].

* Corresponding author. Tel.: +33 3 81 66 64 79.
E-mail address: vincent.ballenegger@univ-fcomte.fr (V. Ballenegger).

0010-4655/$ – see front matter
doi:10.1016/j.cpc.2011.01.026

When a local violation of momentum conservation can be accepted, for example when the system is coupled to a thermostat, analytical differentiation can be the most efficient route since it requires less operations than the other schemes (especially when compared to a finite difference operator of high order). The AD scheme is commonly used in constant-temperature simulations of condensed matter, plasmas and soft matter systems, including large-scale biophysical simulations.

The main drawback of analytical differentiation is of course its violation of momentum conservation, which arises both because the pair force (computed via the mesh) does not obey the action–reaction principle, and because each particle is subjected to a self-force that depends on its position relative to the mesh. It has been proposed recently to subtract the self-forces by using a tabulation of the self-force determined at the beginning of a simulation [9,10]. After a brief reminder of the AD scheme (Section 2), we derive in Section 3 analytical formulae for the self-force and self-energy, for any position of a particle within a mesh cell. These formulae provide a better understanding of the self-interactions, and can be used to subtract them directly without having to first tabulate them. The accuracy gain obtained by subtracting self-forces in the analytically differentiated P3M method and the SPME method is discussed in Section 4.

2. The analytical differentiation scheme

We consider a system made up of N charges q_i at positions $\boldsymbol{r}_i$ in a periodic box of dimensions L_1, L_2, L_3. The box is assumed to be orthorhombic, and its volume is $V = L_1L_2L_3$. The particle–mesh methods use a mesh $\mathbb{M}$ of dimensions N_1, N_2, N_3. The lattice spacings are $h_1 = L_1/N_1$, $h_2 = L_2/N_2$, $h_3 = L_3/N_3$ and the grid nodes are $\boldsymbol{r}_{\boldsymbol{n}} = (h_1n_1, h_2n_2, h_3n_3)$, $n_i = 0 \ldots N_i$. Formally, the energy of the system is approximated on the mesh by

$$E_{\mathrm{PM}} = \frac{1}{2} \sum_{\boldsymbol{r}_{\boldsymbol{n}}, \boldsymbol{r}_{\boldsymbol{n}'} \in \mathbb{M}} q_{\boldsymbol{r}_{\boldsymbol{n}}} q_{\boldsymbol{r}_{\boldsymbol{n}'}} G(\boldsymbol{r}_{\boldsymbol{n}} - \boldsymbol{r}_{\boldsymbol{n}'}) \tag{1}$$

where $G(\boldsymbol{r}_{\boldsymbol{n}})$ is the lattice Green function in real space for periodic boundary conditions. The charges at each mesh point are computed via an assignment function $W(\boldsymbol{r})$,

$$q_{\boldsymbol{r}_{\boldsymbol{n}}} = \sum_{i=1}^{N} q_i W(\boldsymbol{r}_i - \boldsymbol{r}_{\boldsymbol{n}}), \tag{2}$$

where $W(\boldsymbol{r})$ is the Hockney–Eastwood charge assignment function [11,2], equal to a product of three one-dimensional centered B-splines of order P [12]. In the P3M method, the lattice Green function is adjusted so that Eq. (1) gives the best possible discrete approximation to the energy of the original system in continuous space, while the lattice Green function in the SPME method differs from the optimal one [12]. In the analytical differentiation scheme, the forces are obtained by differentiating analytically the approximate energy (1):

$$\begin{aligned} \boldsymbol{F}_i^{\mathrm{PM}} &= -\nabla_i E_{\mathrm{PM}} \\ &= -\sum_{\boldsymbol{r}_{\boldsymbol{n}}, \boldsymbol{r}_{\boldsymbol{n}'} \in \mathbb{M}} (\nabla_i q_{\boldsymbol{r}_{\boldsymbol{n}}}) q_{\boldsymbol{r}_{\boldsymbol{n}'}} G(\boldsymbol{r}_{\boldsymbol{n}} - \boldsymbol{r}_{\boldsymbol{n}'}) \\ &= -q_i \sum_{\boldsymbol{r}_{\boldsymbol{n}} \in \mathbb{M}} \Phi_{\boldsymbol{r}_{\boldsymbol{n}}} \nabla_i W(\boldsymbol{r}_i - \boldsymbol{r}_{\boldsymbol{n}}) \end{aligned} \tag{3}$$

where we used the fact that $G(\boldsymbol{r}_{\boldsymbol{n}})$ is even and introduced the mesh-based potential

$$\Phi_{\boldsymbol{r}_{\boldsymbol{n}}} = \sum_{\boldsymbol{r}_{\boldsymbol{n}'} \in \mathbb{M}} q_{\boldsymbol{r}_{\boldsymbol{n}'}} G(\boldsymbol{r}_{\boldsymbol{n}} - \boldsymbol{r}_{\boldsymbol{n}'}). \tag{4}$$

The particle–mesh force (3) is hence obtained by interpolating the mesh-based potential back to the particle position using the gradient of the charge assignment function. The convolution (4) for the potential mesh is computed in practice as a multiplication in Fourier space by the reciprocal lattice Green function $\widetilde{G}(\boldsymbol{k}_{\boldsymbol{n}})$, the result of which is transformed back to real space to compute the forces via (3). The expression of the optimal $\widetilde{G}(\boldsymbol{k}_{\boldsymbol{n}})$ for the analytically differentiated P3M method can be found in Ref. [6], while the corresponding expression for the SPME method is derived in Ref. [13]:

$$\widetilde{G}_{\mathrm{SPME}}(\boldsymbol{k}_{\boldsymbol{n}}) = \frac{\hat{\phi}(\boldsymbol{k}_{\boldsymbol{n}})}{(\sum_{\boldsymbol{m} \in \mathbb{Z}^3} \hat{U}(\boldsymbol{k}_{\boldsymbol{n} + \underline{\mathbf{N}} \cdot \boldsymbol{m}}))^2} \tag{5}$$

where $\underline{\mathbf{N}} = \mathrm{diag}(N_1, N_2, N_3)$ is a diagonal matrix made with the mesh dimensions and $\hat{\phi}(\boldsymbol{k})$ is the pair interaction between the particles in Fourier space. When the pair interaction is decomposed according to the Ewald method [1,14], $\phi(\boldsymbol{r})$ corresponds to the Coulomb interaction of a point charge interacting with a Gaussian charge distribution of width α^{-1}: $\hat{\phi}(\boldsymbol{k}) = \frac{4\pi}{k^2} \exp(-k^2/(4\alpha^2))$. The complementary interaction $\psi(\boldsymbol{r}) \equiv 1/r - \phi(\boldsymbol{r}) = (1 - \mathrm{erf}(\alpha r))/r$ is short-ranged and computed in real space by direct pair-wise summation. The Ewald screening length α^{-1} is a free parameter that impacts, with an exponential dependence, the accuracy of the method, but not its computational cost; it has to be fine-tuned for getting the best accuracy (see Section 4). The wavevector $\boldsymbol{k}_{\boldsymbol{n}}$ belongs to the finite reciprocal mesh $\widetilde{\mathbb{M}} = \{\boldsymbol{k}_{\boldsymbol{n}} = n_1 \frac{2\pi}{L_1} \hat{\boldsymbol{e}}_1 + n_2 \frac{2\pi}{L_2} \hat{\boldsymbol{e}}_2 + n_3 \frac{2\pi}{L_3} \hat{\boldsymbol{e}}_3\}$ where $n_\beta = -N_\beta/2 + 1, -N_\beta/2 + 2, \ldots, N_\beta/2$, $\beta = 1, 2, 3$. The function $\hat{U}(\boldsymbol{k}) = \hat{W}(\boldsymbol{k})/(h_1h_2h_3)$ is equal to the Fourier transform of the B-spline assignment function $W(\boldsymbol{r})$ of order P divided by the volume of a mesh cell:

$$\hat{U}(\boldsymbol{k}_{\boldsymbol{n}}) = \left(\frac{\sin(\pi n_1/N_1)}{\pi n_1/N_1} \frac{\sin(\pi n_2/N_2)}{\pi n_2/N_2} \frac{\sin(\pi n_3/N_3)}{\pi n_3/N_3} \right)^P. \tag{6}$$

3. Expressions for the self-force and self-energy

The self-force is an artefact caused by the analytical differentiation scheme which breaks a symmetry: the forward and backward mapping of the charges onto/from the mesh is not performed using the same charge assignment function, since its gradient is used in the force interpolation (3) [2, p. 151]. Cerutti et al. [9] measured the self-force at various points in a mesh cell, and showed that the self-force, along direction $\beta = 1, 2, 3$, on a charge q located at $\boldsymbol{r}$ is well described by a Fourier sine series

$$F_\beta^{\mathrm{self}}(\boldsymbol{r}) \simeq q^2 \sum_{n=1}^{\infty} a_\beta^{(n)} \sin(n 2\pi s_\beta), \quad s_\beta = \frac{r_\beta}{h_\beta}. \tag{7}$$

Note that the self-force is periodic over distances h_β, as expected. More generally, the self-force can be expanded as

$$F_\beta^{\mathrm{self}}(\boldsymbol{r}) = q^2 \sum_{\boldsymbol{m} \in \mathbb{Z}^3} b_\beta^{(\boldsymbol{m})} \sin\big(2\pi (m_1 s_1 + m_2 s_2 + m_3 s_3)\big). \tag{8}$$

We prove in Appendix A that, for an orthorhombic simulation cell, the Fourier coefficients $b_\beta^{(\boldsymbol{m})}$ are given by

$$b_\beta^{(\boldsymbol{m})} = \frac{2\pi m_\beta}{h_\beta} \frac{1}{2V} \sum_{\boldsymbol{k}_{\boldsymbol{n}} \neq 0} \widetilde{G}(\boldsymbol{k}_{\boldsymbol{n}}) \sum_{\boldsymbol{m}' \in \mathbb{Z}^3} \hat{U}(\boldsymbol{k}_{\boldsymbol{n} + \underline{\mathbf{N}} \cdot \boldsymbol{m}'}) \hat{U}(\boldsymbol{k}_{\boldsymbol{n} + \underline{\mathbf{N}} \cdot (\boldsymbol{m}' + \boldsymbol{m})}). \tag{9}$$

We note that these coefficients are odd in $\boldsymbol{m}$: $b_\beta^{(-\boldsymbol{m})} = -b_\beta^{(\boldsymbol{m})}$. The coefficients $a_\beta^{(n)}$ in expansion (7) are therefore given by $a_\beta^{(n)} =$

$2b_\beta^{(n\hat{\boldsymbol{e}}_\beta)}$, or, fully explicitly,

$$a_\beta^{(n)} = \frac{2\pi m_\beta}{h_\beta}\frac{1}{V}\sum_{\boldsymbol{k_n}\neq 0}\widetilde{G}(\boldsymbol{k_n})\sum_{\boldsymbol{m'}\in\mathbb{Z}^3}\hat{U}(\boldsymbol{k_n}+\underline{\mathbf{N}}\cdot\boldsymbol{m'})\hat{U}(\boldsymbol{k_n}+\underline{\mathbf{N}}\cdot(\boldsymbol{m'}+\boldsymbol{m})) \tag{10}$$

where $\boldsymbol{m}$ is fixed to $\boldsymbol{m} = n\hat{\boldsymbol{e}}_\beta$ with $\hat{\boldsymbol{e}}_\beta$ the unit vector in direction β. The general formula (9) for the coefficient $b_\beta^{(\boldsymbol{m})}$ can be used to compute terms beyond the decoupled approximation (7), for example terms varying like $\sin(2\pi(s_1+s_2))$, $\sin(2\pi(s_1+s_3))$ or $\sin(2\pi(s_2+s_3))$.

The self-interaction energy $E_{\mathrm{MS}}(\boldsymbol{r})$ of a charged particle located at $\boldsymbol{r}$ can also be expressed as a Fourier series. As shown in Appendix A, we have

$$E_{\mathrm{MS}}(\boldsymbol{r}) = q^2\sum_{\boldsymbol{m}\in\mathbb{Z}^3} c^{(\boldsymbol{m})}\cos\big(2\pi(m_1s_1+m_2s_2+m_3s_3)\big) \tag{11}$$

with coefficients given by

$$c^{(\boldsymbol{m})} = -\frac{1}{2V}\sum_{\boldsymbol{k_n}\neq 0}\widetilde{G}(\boldsymbol{k_n})\sum_{\boldsymbol{m'}\in\mathbb{Z}^3}\hat{U}(\boldsymbol{k_n}+\underline{\mathbf{N}}\cdot\boldsymbol{m'})\hat{U}(\boldsymbol{k_n}+\underline{\mathbf{N}}\cdot(\boldsymbol{m'}+\boldsymbol{m})). \tag{12}$$

$E_{\mathrm{MS}}(\boldsymbol{r})$ is the approximate (mesh-computed) "Madelung self-energy" of a particle (see [15]), defined as the sum of the interaction energy of the particle with itself ("Ewald" self-energy) and with its own periodic images (Madelung energy). Note that $\boldsymbol{F}^{\mathrm{self}}(\boldsymbol{r}) = -\nabla E_{\mathrm{MS}}(\boldsymbol{r})$. The self-force is therefore a consequence of the approximate space-dependent self-energy and of the analytical differentiation scheme.

4. Numerical results and discussion

Our test system is made up of $N = 800$ charged particles (400 carry a positive and 400 a negative unit charge) distributed at random in a cubic simulation box of length $L = 20$ units. Forces are measured in units of $\mathcal{C}^2/\mathcal{L}^2$ where the unit of charge $\mathcal{C}$ and the unit of length $\mathcal{L}$ are arbitrary ($\mathcal{C}$ and $\mathcal{L}$ could be for example the electronic charge and one Ångström, or a solar mass and a parsec in a cosmological simulation). If the unit of length is multiplied by a scale factor $s > 0$ ($\mathcal{L}' = s\mathcal{L}$), forces are multiplied by a factor s^{-2} and the density of the system is multiplied by factor s^{-3} (i.e. it is reduced if $s > 1$). By applying such a scaling to our results, the density of our test system (0.1 particle/$\mathcal{L}^3$) can easily be converted into any other value. The results obtained for our test system are therefore representative for charged systems characterized by a uniform random charge distribution of arbitrary density. We chose the same reference density 0.1 particle/$\mathcal{L}^3$ as in Ref. [12] but work with a system size twice larger for better statistics and for allowing larger real-space cutoffs (note that results for $L = 10$, $N = 100$, mesh size $M = N_1N_2N_3 = 32^3$ and real-space cutoff $r_c = 4$ are equivalent by extensivity to results for $L = 20$, $N = 800$, mesh size $M = 64^3$ and the *same* cutoff $r_c = 4$). We varied the mesh size from $M = 16^3$ to 64^3, the real-space cutoff from $r_c = 2$ to 9 and the spline interpolation order P from 3 to 7.

Fig. 1 shows the accuracy gain brought by subtracting the self-forces in the P3M algorithm with analytical differentiation, as a function of the Ewald splitting parameter α, for various choices of the mesh size and of the real-space cutoff distance r_c. The spline interpolation order is fixed to $P = 5$, but similar curves are found for other values of P. Analogous results are obtained for the SPME method (data not shown). When $r_c = 9$, the optimal accuracy of the P3M method is obtained when $\alpha \simeq 0.3$ for $M = 16^3$ and $\alpha \simeq 0.4$ for $M = 64^3$. The optimal value of α shifts to larger values when the real-space cutoff is reduced. For $r_c = 3$, the optimal value of α is for example $\alpha_{\mathrm{opt}} \simeq 0.83$ for $M = 32^3$ and $P = 5$. With parameters in this range, the accuracy of the forces is improved by about 30% when the self-forces are subtracted.

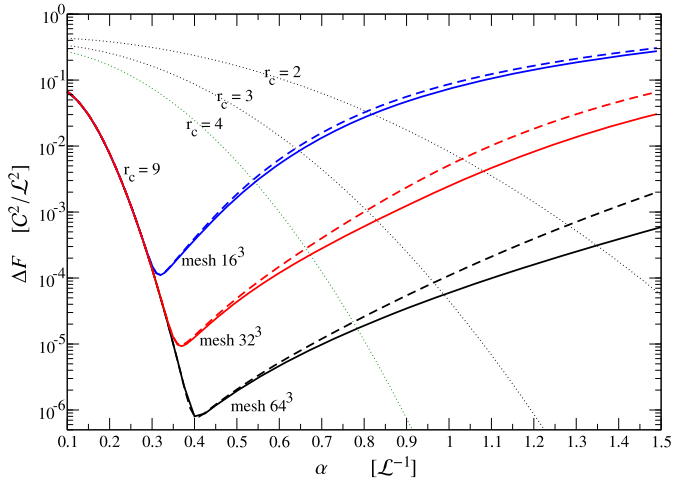

Fig. 1. The root-mean-square accuracy $\Delta F = \sqrt{\frac{1}{N}\sum_i (\boldsymbol{F}_i - \boldsymbol{F}_i^{\mathrm{exact}})^2}$ of the P3M forces (computed with analytical differentiation) is shown for our uniform test system (see text) as a function of the Ewald parameter α for different mesh sizes (from 16^3 to 64^3) and different real-space cutoff distances r_c (from 2 to 9). The spline interpolation order is set to $P = 5$. The accuracies of the particle–particle part of the force calculation for cutoffs $r_c = 2, 3$ and 4 are shown as dotted lines. Solid lines represent the accuracy of the particle–mesh forces obtained when self-forces are subtracted, while dashed lines correspond to the accuracy with self-forces included.

Table 1
Fourier coefficients $b_\beta^{(\boldsymbol{m})}$ for the self-force along direction $\beta = 1$ computed using Eq. (9) for a cubic box of side $L = 20$, mesh size $M = 32^3$, Ewald parameter $\alpha = 0.83$ and spline interpolation order $P = 5$. The self-force $F^{\mathrm{self}}_{\beta=1}(r_1, r_2, r_3)$ does not depend solely on particle coordinate r_1; its dependence on coordinates r_2 and r_3 is described by Fourier coefficients with vector $\boldsymbol{m} = (m_1, m_2, m_3)$ not purely along direction 1.

m_1	m_2	m_3	Fourier coefficient ($\mathcal{L}^{-2}$)
1	0	0	1.706×10^{-3}
2	0	0	1.528×10^{-4}
3	0	0	4.198×10^{-5}
4	0	0	1.722×10^{-5}
...			
1	±1	0	1.960×10^{-6}
1	0	±1	1.960×10^{-6}
2	±1	0	1.682×10^{-7}

When using a large cutoff $r_c \gtrsim 9$ (and hence a small α), no gain in accuracy is obtained by subtracting the self-forces: the dominant source of error is then not the self-forces, but errors in the interparticle forces. It is therefore worthy to subtract the self-forces only when one uses a relatively small cutoff in the simulation. The choice of the cut-off is dictated in part by the particular system under study (i.e. whether there are other short-range interactions to be computed in the pair-wise summation in real space) and by the respective timings, for various combination of parameters r_c, M and P, of the particle–particle part and of the particle–mesh part of the P3M (or SPME) method. Decreasing the cut-off makes indeed the particle–particle part of the algorithm faster, but the particle–mesh calculation has to be able to deliver the required accuracy sufficiently quickly. At the end, the cutoff is determined from the optimal set of parameters $\{r_c, M, P\}$ that provides the prescribed accuracy with the shortest possible computational time. Small cut-offs are usually present in simulations of dense (charged) systems.

We list in Table 1 the first few Fourier coefficients $b_\beta^{(\boldsymbol{m})}$ for the self-force in direction $\beta = 1$, calculated with Eq. (9) for parameter values in the region where the subtraction of the self-force brings a substantial increase in accuracy. Thanks to the rapid decay of the coefficients, the Fourier series of the self-force can be truncated essentially after the first two terms, in agreement with the findings of Ref. [9]. The self-force $F^{\text{self}}_{\beta=1}(r_1, r_2, r_3)$ does not depend solely on particle coordinate r_1. Its dependence on the other coordinates is described by Fourier coefficients with vector $\boldsymbol{m} = (m_1, m_2, m_3)$ not purely along direction 1. As these terms are found to be smaller than the uncoupled terms $a_\beta^{(n)}$ for $n \leqslant 4$, they can safely be neglected, unless one wants to calculate the self-force to a very high precision.

Though particle–mesh algorithms with analytical differentiation don't conserve momentum, they do conserve energy in the limit of small time steps. The subsequent correction of forces to enforce conservation of the center-of-mass momentum breaks down however this exact conservation. To preserve it as well as possible, one has to subtract not only the self-forces, but also the interaction energy $E_{\text{MS}}(\boldsymbol{r})$ of each particle with itself and with its periodic images, which is given by formula (11). The exact Madelung self-energy of a particle (which is a constant) can then be added back so that particles have the right self-energy associated to the periodic boundary conditions. In this way, the energy (1) of the discretized system is modified in a way consistent with the subtraction of the self-forces to maintain the equality $\boldsymbol{F}_i = -\nabla_i E_{\text{PM}}$. A correction of this kind has been introduced in Ref. [16] to remove, *on average*, the bias of the P3M energies that is caused by the approximate (mesh computed) self-interactions. Correcting exactly these self-interactions at the level of each particle will improve the accuracy of the computed energies beyond the average correction of Ref. [16].

In summary, we have shown that the accuracy of particle–mesh calculations can be enhanced, especially for small cut-offs, by removing (or correcting in the case of the energy) the self-interaction of each particle. Our analytic expressions for the self-interactions, namely the self-force associated to the analytical differentiation scheme and the self-energy, can be evaluated during a simulation at virtually no computational cost. A further benefit of subtracting the self-interactions is that it reduces the amount of non-conservation of momentum (and energy) in particle–mesh methods with analytical differentiation.

Appendix A. Proof of formulae (8)–(9) for the self-force

Let $\boldsymbol{F}(\boldsymbol{r}_1, \boldsymbol{r}_2)$ be the particle–mesh force felt by a test particle with unit charge at $\boldsymbol{r}_1$ due to a particle with unit charge at $\boldsymbol{r}_2$ when analytical differentiation is used. That function can be expressed as a Fourier series in both variables

$$\boldsymbol{F}(\boldsymbol{r}_1, \boldsymbol{r}_2) = \frac{1}{V^2} \sum_{\boldsymbol{k}_1, \boldsymbol{k}_2} \hat{\boldsymbol{F}}(\boldsymbol{k}_1, \boldsymbol{k}_2)\, e^{i(\boldsymbol{k}_1 \cdot \boldsymbol{r}_1 + \boldsymbol{k}_2 \cdot \boldsymbol{r}_2)} \tag{A.1}$$

where the Fourier coefficients are given by [10]

$$\hat{\boldsymbol{F}}(\boldsymbol{k}_1, \boldsymbol{k}_2) = -i\boldsymbol{k}_1 V \hat{U}(\boldsymbol{k}_1)\hat{U}(\boldsymbol{k}_2)\widetilde{G}\big(\boldsymbol{k}_1^{\text{G}}\big) \sum_{\boldsymbol{m} \in \mathbb{Z}^3} \delta_{\boldsymbol{k}_1 + \boldsymbol{k}_2 + \boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}}}. \tag{A.2}$$

Vector $\boldsymbol{k}_1^{\text{G}}$ is defined as in Ref. [10] – that is as vector $\boldsymbol{k}_1$ folded back into the first Brillouin zone $\widetilde{\mathbb{M}}$; matrix $\underline{\mathbf{N}}$ and vector $\boldsymbol{k}_{\boldsymbol{n}}$ (here for $\boldsymbol{n} = \underline{\mathbf{N}} \cdot \boldsymbol{m}$) are defined in Section 2 of the main text. The self-force is obtained by setting $\boldsymbol{r}_1 = \boldsymbol{r}_2$ in (A.1):

$$\begin{aligned}
&\boldsymbol{F}^{\text{self}}(\boldsymbol{r}) \\
&\quad = \boldsymbol{F}(\boldsymbol{r}, \boldsymbol{r}) \\
&\quad = \frac{1}{V} \sum_{\boldsymbol{k}_1, \boldsymbol{k}_2} (-i\boldsymbol{k}_1)\hat{U}(\boldsymbol{k}_1)\hat{U}(\boldsymbol{k}_2)\widetilde{G}\big(\boldsymbol{k}_1^{\text{G}}\big) \sum_{\boldsymbol{m}} \delta_{\boldsymbol{k}_1 + \boldsymbol{k}_2 + \boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}}} e^{i(\boldsymbol{k}_1 + \boldsymbol{k}_2) \cdot \boldsymbol{r}} \\
&\quad = \frac{1}{V} \sum_{\boldsymbol{k}_1} (-i\boldsymbol{k}_1)\widetilde{G}\big(\boldsymbol{k}_1^{\text{G}}\big)\hat{U}(\boldsymbol{k}_1) \sum_{\boldsymbol{m}} \hat{U}(\boldsymbol{k}_1 + \boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}}) e^{-i\boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}} \cdot \boldsymbol{r}} \\
&\quad = -\frac{1}{V} \sum_{\boldsymbol{k}_1} \boldsymbol{k}_1 \widetilde{G}\big(\boldsymbol{k}_1^{\text{G}}\big)\hat{U}(\boldsymbol{k}_1) \sum_{\boldsymbol{m}} \hat{U}(\boldsymbol{k}_1 + \boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}}) \sin(\boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}} \cdot \boldsymbol{r})
\end{aligned} \tag{A.3}$$

where the last equality holds because $\widetilde{G}(\boldsymbol{k})$ and $\hat{U}(\boldsymbol{k})$ are even functions of vector $\boldsymbol{k}$. Since $\boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}} \cdot \boldsymbol{r} = 2\pi(m_1 s_1 + m_2 s_2 + m_3 s_3)$ with $s_\beta = r_\beta / h_\beta$, the self-force is periodic over distances h_β and we can write

$$\boldsymbol{F}^{\text{self}}(\boldsymbol{r}) = \sum_{\boldsymbol{m}} \boldsymbol{b}^{(\boldsymbol{m})} \sin\big(2\pi(m_1 s_1 + m_2 s_2 + m_3 s_3)\big) \tag{A.4}$$

where

$$\boldsymbol{b}^{(\boldsymbol{m})} \equiv -\frac{1}{V} \sum_{\boldsymbol{k}_1} \boldsymbol{k}_1 \widetilde{G}\big(\boldsymbol{k}_1^{\text{G}}\big)\hat{U}(\boldsymbol{k}_1)\hat{U}(\boldsymbol{k}_1 + \boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}}). \tag{A.5}$$

Decomposing the wavevector $\boldsymbol{k}_1$ as $\boldsymbol{k}_1 = \boldsymbol{k}_{\boldsymbol{n}} + \boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}'}$ with $\boldsymbol{k}_{\boldsymbol{n}} = \boldsymbol{k}_1^{\text{G}}$ the folded position into the first Brillouin zone $\widetilde{\mathbb{M}}$ and $\boldsymbol{m}' \in \mathbb{Z}^3$, we find that the term with $\boldsymbol{k}_{\boldsymbol{n}}$ gives a contribution to the self-force that vanishes by symmetry, since $\boldsymbol{k}_{\boldsymbol{n}}$ is odd while $G(\boldsymbol{k}_{\boldsymbol{n}})$ is even and

$$f^{(\boldsymbol{m})}(\boldsymbol{k}_{\boldsymbol{n}}) \equiv \sum_{\boldsymbol{m}' \in \mathbb{Z}^3} \hat{U}(\boldsymbol{k}_{\boldsymbol{n} + \underline{\mathbf{N}} \cdot \boldsymbol{m}'})\hat{U}(\boldsymbol{k}_{\boldsymbol{n} + \underline{\mathbf{N}} \cdot \boldsymbol{m}' + \underline{\mathbf{N}} \cdot \boldsymbol{m}}) \tag{A.6}$$

is an even function of $\boldsymbol{k}_{\boldsymbol{n}}$ for any fixed vector $\boldsymbol{m}$. This result corresponds to the absence of self-force in the case of $i\boldsymbol{k}$-differentiation [in that scheme the particle–mesh pair force is indeed given by (A.2) with $\boldsymbol{k}_1$ replaced by $\boldsymbol{k}_1^{\text{G}}$]. Eq. (A.5) reduces therefore to

$$\boldsymbol{b}^{(\boldsymbol{m})} = -\frac{1}{V} \sum_{\boldsymbol{k}_{\boldsymbol{n}} \in \widetilde{\mathbb{M}}} \widetilde{G}(\boldsymbol{k}_{\boldsymbol{n}}) \sum_{\boldsymbol{m}' \in \mathbb{Z}^3} \boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}'} \hat{U}(\boldsymbol{k}_{\boldsymbol{n} + \underline{\mathbf{N}} \cdot \boldsymbol{m}'})\hat{U}(\boldsymbol{k}_{\boldsymbol{n} + \underline{\mathbf{N}} \cdot \boldsymbol{m}' + \underline{\mathbf{N}} \cdot \boldsymbol{m}}). \tag{A.7}$$

Since $\boldsymbol{b}^{(\boldsymbol{m})} = -\boldsymbol{b}^{(-\boldsymbol{m})}$, we can rewrite the previous expression as

$$\boldsymbol{b}^{(\boldsymbol{m})} = \frac{\boldsymbol{b}^{(\boldsymbol{m})} - \boldsymbol{b}^{(-\boldsymbol{m})}}{2} = \boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}} \frac{1}{2V} \sum_{\boldsymbol{k}_{\boldsymbol{n}}} \widetilde{G}(\boldsymbol{k}_{\boldsymbol{n}}) f^{(\boldsymbol{m})}(\boldsymbol{k}_{\boldsymbol{n}}) \tag{A.8}$$

where symmetries were used to factor out the vector $\boldsymbol{k}_{\underline{\mathbf{N}} \cdot \boldsymbol{m}}$. This formula for coefficient $\boldsymbol{b}^{(\boldsymbol{m})}$ is equivalent to expression (9) given in the main text.

Formulae (11)–(12) for the Madelung self-energy $E_{\text{MS}}(\boldsymbol{r})$ of a particle can be obtained by integrating term by term the Fourier series (A.4) for the self-force. Alternatively, it can also be deduced directly by noticing that $E_{\text{MS}}(\boldsymbol{r})$ is given by the first line of Eq. (A.3) in which factor $(-i\boldsymbol{k}_1)$ is replaced by $\frac{1}{2}$.

References

[1] P.P. Ewald, Ann. Phys. 369 (1921) 253–287.
[2] R.W. Hockney, J.W. Eastwood, Computer Simulations Using Particles, McGraw–Hill, New York, 1981.
[3] U. Essmann, L. Perera, M. Berkowitz, T. Darden, H. Lee, L. Pedersen, J. Chem. Phys. 103 (1995) 8577–8593.
[4] T. Darden, D. York, L. Pedersen, J. Chem. Phys. 98 (1993) 10089–10092.
[5] R. Ferrell, E. Bertschinger, J. Mod. Phys. 5 (1994) 933.
[6] H. Stern, K. Calkins, J. Chem. Phys. 128 (2008) 214106.
[7] H. Wang, F. Dommert, C. Holm, J. Chem. Phys. 133 (2010) 034117.
[8] R. Skeel, D. Hardy, J. Phillips, J. Comput. Phys. 225 (2007) 1–5.

[9] D. Cerutti, R. Duke, T. Darden, T. Lybrand, J. Chem. Theory Comput. 5 (2009) 2322–2338.
[10] A. Neelov, C. Holm, J. Chem. Phys. 132 (2010) 234103.
[11] J.W. Eastwood, Computational Methods in Classical and Quantum Physics, Advance Publications Limited, 1976, pp. 206–228.
[12] M. Deserno, C. Holm, J. Chem. Phys. 109 (1998) 7678.
[13] V. Ballenegger, J.J. Cerdà, C. Holm, A simple error estimate for the SPME method, J. Chem. Theory Comput., submitted for publication..
[14] D. Frenkel, B. Smit, Understanding Molecular Simulation, 2nd ed., Academic Press, San Diego, 2002.
[15] J.J. Cerdà, V. Ballenegger, O. Lenz, C. Holm, J. Chem. Phys. 129 (2008) 234104.
[16] V. Ballenegger, J.J. Cerdà, O. Lenz, C. Holm, J. Chem. Phys. 128 (2008) 034109.