

# A hybrid Fast Multipole Method for cosmological $N$ -body simulations

Qiao Wang

Key Laboratory for Computational Astrophysics, National Astronomical Observatories, Chinese Academy of Sciences, Beijing 100101, China; [qwang@nao.cas.cn](mailto:qwang@nao.cas.cn)

School of Astronomy and Space Science, University of Chinese Academy of Sciences, Beijing 100049, China

Received 2020 April 24; accepted 2020 June 24

**Abstract** We investigate a hybrid numerical algorithm aimed at large-scale cosmological  $N$ -body simulation for on-going and future high precision sky surveys. It makes use of a truncated Fast Multiple Method (FMM) for short-range gravity, incorporating a Particle Mesh (PM) method for long-range potential, which is applied to deal with extremely large particle number. In this work, we present a specific strategy to modify a conventional FMM by a Gaussian shaped factor and provide quantitative expressions for the interaction kernels between multipole expansions. Moreover, a proper Multipole Acceptance Criterion for the hybrid method is introduced to solve potential precision loss induced by the truncation. Such procedures reduce the amount of computation compared to an original FMM and decouple the global communication. A simplified version of code is introduced to verify the hybrid algorithm, accuracy and parallel implementation.

**Key words:** methods: numerical — cosmology: theory — large-scale structure of universe

## 1 INTRODUCTION

In the early Universe, extremely hot and dense baryons and photons are strongly interacting prior to the recombination. The relic of the fluctuation of photons is imprinted on the Cosmic Microwave Background Radiation which can be detected at the radio band, as the Universe is expanding (Planck Collaboration 2018). Meanwhile, the initial fluctuation of mass increased due to the gravitational collapse that formed planets, galaxies, cluster halos and large scale structure of the Universe at present (Peebles & Yu 1970). Several primary cosmic probes can extract information locked in the mass distribution, such as mass function of cluster counting, the sound horizon of Baryon Acoustic Oscillations as a standard ruler, power spectrum of cosmic gravitational lensing, etc (Kaiser 1987; Eisenstein & et al. 2005). The evidence combined with various data from sky surveys (Rozo et al. 2010; Abbott et al. 2018) supports a modern picture of cosmology with two mysterious components, dark matter and dark energy, whose natures are still puzzles of standard physics (DES Collaboration 2019).

The next generation sky surveys will reveal the dark side of the Universe by utilizing distant galaxies

and quasars as cosmic probe tracers, including DESI<sup>1</sup>, EUCILD<sup>2</sup> and LSST<sup>3</sup>. One crucial step is to generate a simulated catalog of those tracers. However, it is not trivial to produce the related observable for galaxy formation which involves complicated astrophysical processes and nonlinear evolution (Cole et al. 2000; Berlind & Weinberg 2002; Kitaura et al. 2016; Guo et al. 2011). Mock tracers, such as the emission line galaxies residing in smaller structures and at an earlier epoch, require underlying cosmological simulations with unprecedented resolution and volume to understand their mask effects, cosmic variance, redshift uncertainties, etc. for the next generation surveys. Various techniques are developed for large-scale simulations carried out on top supercomputers with a trillion particles.

The movement of particles in an expanding background can be modeled by equivalent Newtonian gravity with periodic boundary conditions. One natural solver of this model is referred to as the Particle Mesh (PM) method (Hockney & Eastwood 1988) by convolution of the Green function of gravity. The convolution has cost of  $\mathcal{O}(N \log N)$ , calling Fast Fourier Transformation (FFT)

---

<sup>1</sup> <https://www.desi.lbl.gov/>

<sup>2</sup> <https://www.euclid-ec.org/>

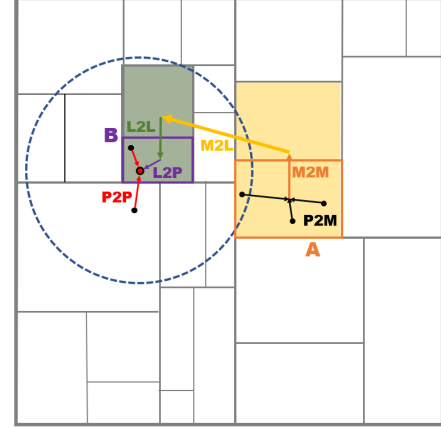
<sup>3</sup> <https://www.lsst.org/>

two times. However, PM can only deal with the scale above the computing grid so that it needs a compensated sub-grid gravity solver, such as a truncated Particle-Particle (PP) direct summation. The pioneer cosmological simulation based on PP+PM method, so called  $P^3M$ , identified the structure of the cosmic web (Efstathiou et al. 1985; Jing & Suto 2002). Once the system has apparently condensed, it will fail to reduce to  $\mathcal{O}(N^2)$  due to the domination of PP interactions.

Another solver with cost of  $\mathcal{O}(N \log N)$  is introduced by Barnes & Hut (1986). They make use of an octal tree to organize the particles. Details about the source cells can be neglected, since it attracts a well-separated particle like the gravity of mass points. Thus, any particle is just concerned with the cells or particles “near enough”, determined by an opening angle to control the precision of the acceleration. Warren (2013) modified a tree code (2HOT) to run on Graphics Processing Units (GPUs) and Bédorf et al. (2014) optimized *Bonsai* (Portegies Zwart et al. 2013) to achieve performance of 24.77 Pflops for the Milky Way simulation.

Tree code is less sensitive to the particle clustering than a PM code but the PM method is stable and rapid for a regular and periodic mesh. A hybrid TreePM method can combine the merits from both methods (Bagla 2002; Bagla & Ray 2003). The Millennium Simulation Project (Springel et al. 2005) was carried out with over  $10^{10}$  particles by the parallel TreePM code of Gadget-2 (Springel 2005) over ten years ago. The idea of TreePM is also effective on a heterogeneous platform, such as HACC (Habib et al. 2016, 2013). Recently, Ishiyama et al. (2012) ran a trillion particle cosmological simulation on K computer.

To face the challenge of high precision, the scale of simulation is increasing with the capability of supercomputers. That requires a faster algorithm to deal with a unprecedented particle number. Fast Multipole Method (FMM) has a nearly linear computational complexity of  $\mathcal{O}(N)$ . It was originally introduced by Greengard & Rokhlin (1987). Cheng et al. (1999) extended it into three dimensions. Similar to the Tree method, FMM also builds a tree but computes the interactions between cells. The early FMM works in spherical polar coordinates. It has also been successfully accelerated on GPU devices, such as ExaFMM<sup>4</sup> (Gumerov & Duraiswami 2008; Yokota & Barba 2011). Its precision is controlled by the order of expansions. Moreover, an implementation in Cartesian coordinates may be more suitable for astrophysical simulation with an acceptance criterion based on opening angle, instead of “children of parent’s brother” (Dehnen 2000, 2002).



**Fig.1** Schematic diagram for FMM. The gravitational potential at a certain point (red point with black circle) in Region A is generated by a group of particles in region B. The multipole expansion is computed by P2M in Region A. Then the information is successively propagated via interaction kernels, M2M, M2L, L2L and L2P. The gravity from the nearby particles can be directly accumulated by P2P. The blue dashed circle signifies the cutoff radius.

For some mass dependent criterion, its complexity can approach  $\mathcal{O}(N^{0.86})$  by relying on a dual tree traversal. Recently, Potter et al. (2017) reported that the application of cosmological simulation is able to involve 2 trillion particles by software PKDGRAV-3 on supercomputer Piz Daint.

This paper is organized as follows. In Section 2, we briefly introduce the fundamental approach for FMM, then we investigate the combination of FMM and PM method, two current methods, for cosmological simulation. The details of our algorithm will be discussed in Section 2.2. A corresponding Multipole Acceptance Criterion (MAC) is taken into account in Section 2.3. We measure the reduction of kernel interactions via the hybrid method compared to a conventional one in Section 3. In Section 4, we introduce a parallel implementation to verify the algorithm. Finally, we summarize in the last section.

## 2 ALGORITHM

### 2.1 Fast Multipole Method

In FMM, all particles are organized into a tree and the finest tree cells (or tree nodes) are always a series of particle packs, which are also referred to as leaves. In Figure 1, we employ an Orthogonal Recursive Bisection (ORB) tree and set a maximum limit on the particle number in leaves. The particles including parent cell are almost equally divided into two offspring cells down to leaves.

The gravitational potential at a point in region B induced by the particles in region A can be estimated

<sup>4</sup> <http://www.bu.edu/exafmm/>

via a series of interaction kernels between the multipole coefficients. We follow the mathematical notations in Dehnen (2014) and summarize the relevant equations of FMM method in Cartesian coordinates. The multipole coefficients  $\mathcal{M}$  of region  $A$  are computed by the particles

$$\mathcal{M}_{\mathbf{m}}(\mathbf{z}_A) = \sum_{a \in A} \mu_a \frac{(-1)^m}{\mathbf{m}!} (\mathbf{x}_a - \mathbf{z}_A)^{\mathbf{m}}, \quad (1)$$

where  $\mathbf{z}_A$  is the geometric (or mass) center of region  $A$ ,  $\mu_a$  is mass of the particle labeled by  $a$  at the position of  $\mathbf{x}_a$  and the integer vector  $\mathbf{n}$  denotes  $(n_x, n_y, n_z)$ . One can shift a multipole expansion from  $\mathbf{z}$  to  $\mathbf{z} + \mathbf{x}$  by the summation of

$$\mathcal{M}_{\mathbf{m}}(\mathbf{z} + \mathbf{x}) = \sum_{\mathbf{n}=0}^{\mathbf{m}} \frac{\mathbf{x}^{\mathbf{n}}}{\mathbf{n}!} \mathcal{M}_{\mathbf{m}-\mathbf{n}}(\mathbf{z}). \quad (2)$$

Given a Green function of  $\psi(\mathbf{z}_b - \mathbf{z}_a)$ , the coefficients of local expansion  $\mathcal{L}$  of potential  $\Psi$  at the center of region  $B$  of  $\mathbf{z}_B$  are determined by the  $\mathcal{M}$  at  $\mathbf{z}_a$  by equation

$$\mathcal{L}_{\mathbf{n}}(\mathbf{z}_B) = \sum_{|\mathbf{m}|=0}^{p-|\mathbf{n}|} \mathcal{M}_{\mathbf{m}}(\mathbf{z}_A) \mathcal{D}_{\mathbf{n}+\mathbf{m}}(\mathbf{z}_B - \mathbf{z}_A), \quad (3)$$

where  $\mathcal{D}_{\mathbf{n}} \equiv \nabla^{\mathbf{n}} \psi$  is a *traceless* operator. Newtonian gravity or electrostatic force can be expressed by a traceless displacement tensor  $\bar{\mathbf{r}}^{\mathbf{n}}$  multiplied by the prefactors of

$$\tilde{f}_{(n)}(r) = (-1)^n \frac{(2n-1)!!}{r^{2n+1}}. \quad (4)$$

Similarly, the multipole  $\mathcal{L}$  can be shifted by

$$\mathcal{L}_{\mathbf{n}}(\mathbf{z} + \mathbf{x}) = \sum_{|\mathbf{m}|=0}^{p-|\mathbf{n}|} \frac{\mathbf{x}^{\mathbf{m}}}{\mathbf{m}!} \mathcal{L}_{\mathbf{m}+\mathbf{n}}(\mathbf{z}). \quad (5)$$

Thus, the potential at  $\mathbf{x}_b$  is approximated by

$$\Psi(\mathbf{x}_b) = \sum_{|\mathbf{n}|=0}^p \frac{1}{\mathbf{n}!} \mathcal{L}_{\mathbf{n}}(\mathbf{z}_B) (\mathbf{x}_b - \mathbf{z}_B)^{\mathbf{n}}. \quad (6)$$

For short, the multipole expansion for the source is labeled by ‘multipole’ and the expansion at the sink (or target) place is labeled by ‘local’. Then the abbreviations of the kernels are as follows: particle-to-particle is referred as to P2P, particle-to-multipole is P2M, multipole-to-multipole is M2M, multipole-to-local is M2L, local-to-local is L2L and local-to-particle is L2P. The multipole and local expansion coefficients are included in all tree cells.

The above equations can be utilized to transmit the information on gravity from one particle group to another. The gravity of a certain particle in the purple sink (targeted) leaf induced by the orange source leaf can be computed through a series of successive kernels. The multipole coefficients  $\mathcal{M}$  of an (orange) leaf are

determined by the discrete mass points in the cell (by using Eq. (1)), which describe mass distribution in source cells or leaves. Then the multipole expansion coefficients in a parent cell are computed by its offspring cells recursively (utilizing Eq. (2)). This procedure is also referred to as upward pass.

The kernel M2L computes the local expansion coefficients of gravitational potential from the multipole of source cells by Equation (3). When two cells are well separated from each other, the M2L kernel is called to compute the interactions from yellow to green cells in this illustration. The local expansion coefficients are passed level-by-level downward (applying Eq. (5)) till a leaf is met. The gravity and potential of the targeted particles (red point with black circle) is calculated by the local expansion about the center of purple leaf, using Equation (6).

The nearby particles surrounding the target are too close to be dealt with by the above kernels and they must be directly accumulated by their interaction. Since a direct  $N$ -body summation is  $\mathcal{O}(N^2)$ , it can be quite time consuming. The procedure is referred to as P2P, which actually can be considered as interactions between particle packs in our implementation as well.

## 2.2 Combination of Particle-Mesh and Fast Multipole Method

Analogous to TreePM, the essence of Particle-Mesh and Fast Multipole Method (PM-FMM) is also to split gravity into two parts by scale. A truncated short-range part of gravity is computed by FMM and a smoothed long-range part is by PM method. A combination of two parts must be equivalent with an original inverse-square law at the split scale, by fine tuning its splitting function. Bagla (2002) suggests a Gaussian form as transition function for TreePM and we generalize it to the PM-FMM in this section.

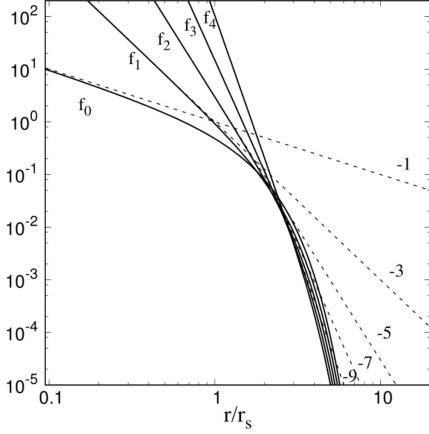
Specifically, the convolution for PM with a grid size of  $\Delta_g$  needs a Gaussian function  $\exp(-k^2/4k_s^2)/\sqrt{\pi}$  as a filter to suppress the Green function of the Poisson Equation, where  $k_s$  is the wave number of split scale of  $r_s \sim 1.2\Delta_g$ .

Correspondingly, the short range gravity must be properly modified to compensate underestimation from the PM method. As a result, the potential  $r^{-1}$  is modified by error function as

$$\phi(r) = f_{(0)}(r) = \frac{1}{r} \operatorname{erfc}\left(\frac{r}{2r_s}\right), \quad (7)$$

and an inverse-square law is modified by

$$g(r) = f_{(1)}\mathbf{r} = -\frac{\mathbf{r}}{r^3} \left[ \operatorname{erfc}\left(\frac{r}{2r_s}\right) + \frac{r}{r_s\sqrt{\pi}} \exp\left(-\frac{r^2}{4r_s^2}\right) \right]. \quad (8)$$



**Fig. 2** Reduction effect. All prefactors are truncated at the cutoff radius  $\sim 5r_s$ .

Moreover, the operator  $\mathcal{D}_n = \nabla^n[\text{erfc}(|\mathbf{r}|/r_s)\phi_N]$  for a truncated potential is employed for computation of FMM, where  $\phi_N$  is Newtonian potential.

For higher orders, the M2L kernel in the truncated algorithm can be implemented by minimum substitutions of the original prefactor  $\tilde{f}_{(n)}$  (Eq. (4)) by the following ones without tildes:

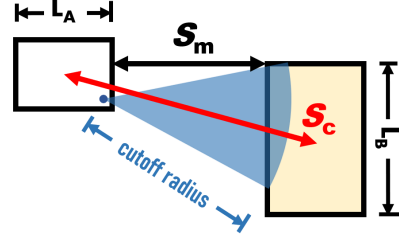
$$\begin{aligned} f_{(2)}(r) &= \frac{3}{r^5} \text{erfc}\left(\frac{r}{2r_s}\right) + \frac{1}{\sqrt{\pi}} \exp\left(-\frac{r^2}{4r_s^2}\right) \\ &\quad \times \left[ \frac{3}{r_s r^4} + \frac{1}{2r_s^3 r^2} \right], \\ f_{(3)}(r) &= -\frac{15}{r^7} \text{erfc}\left(\frac{r}{2r_s}\right) - \frac{1}{\sqrt{\pi}} \exp\left(-\frac{r^2}{4r_s^2}\right) \\ &\quad \times \left[ \frac{15}{r_s r^6} + \frac{5}{2r_s^3 r^4} + \frac{1}{4r_s^5 r^2} \right], \\ f_{(4)}(r) &= \frac{105}{r^9} \text{erfc}\left(\frac{r}{2r_s}\right) + \frac{1}{\sqrt{\pi}} \exp\left(-\frac{r^2}{4r_s^2}\right) \\ &\quad \times \left[ \frac{105}{r_s r^8} + \frac{35}{2r_s^3 r^6} + \frac{7}{4r_s^5 r^4} + \frac{1}{8r_s^7 r^2} \right]. \end{aligned}$$

Figure 2 demonstrates the comparison of two kinds of prefactors. It is apparent that all interactions are truncated at the radius of  $5 \times r_s$ , a cutoff radius  $R_{\text{cutoff}}$ , so that gravity at a short range is negligible beyond that scale. For P2P kernel, only  $f_{(1)}$  is needed.

One can compute the prefactor of any order  $p$  by equation

$$\begin{aligned} (-1)^p r_s^{2p+1} f_{(p)}(x) &= \frac{(2p-1)!!}{x^{2p+1}} \text{erfc}\left(\frac{x}{2}\right) \\ &\quad + \sum_{q=1}^p \frac{2^{q-p} (2p-1)!!}{(2p-2q+1)!!} \frac{e^{-x^2/4}}{\sqrt{\pi} x^{2q}}, \end{aligned} \quad (9)$$

where  $x \equiv r/r_s$ , or a more effective approach is to calculate all prefactors in successive orders via a recurrence



**Fig. 3** MAC.  $L_A$  is side length of sink (targeted) tree cell and  $L_B$  is the source one. The red arrow is the separation  $S_c$  between the centers of two nodes and  $S_m$  is minimum distance between boundaries of tree cells. Two boxes are still physically relevant despite  $S_c$  being beyond the cutoff radius.

form of

$$-r_s^2 x^2 f_{(p+1)}(x) = (2p-1)f_{(p)}(x) + \frac{e^{-x^2/4}}{2^{p-1}\sqrt{\pi}r_s^{2p-1}}. \quad (10)$$

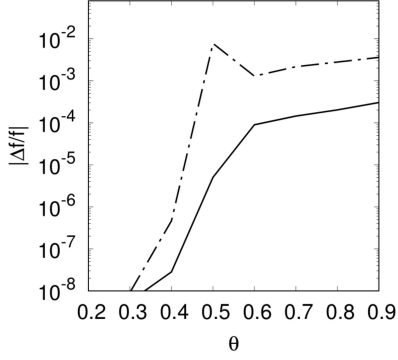
The hybrid algorithm is illustrated in Figure 1. The dashed circle signifies a cutoff radius. The contribution from FMM is localized within the cutoff radius.

### 2.3 Multipole Acceptance Criterion and Error Estimation

A dual tree traversal method can complete all kernel computations by one tree walking. For a local tree, the traversal begins with root-root pair but it can begin with any pair of cells. If two cells are “well separated”, an M2L kernel will be employed to compute the local multipole of sink cell induced by the source one. Otherwise one of two cells, usually a larger one, needs to be opened. The traversal recursively proceeds in the opened cell until walking to the end of a tree. The interaction between two “nearby” leaves must use a P2P kernel.

Therefore, the definition of “well separated” will be influenced by MAC. One can define the geometric relation by the parameter of *opening angle*  $\theta = L/S$ , where  $L$  is the length of cell and  $S$  is the separation between two cells. It is still free to choose those lengths. In PKDGRAV3, [Potter et al. \(2017\)](#) utilize the opening radius  $RO = b_{max}/\theta$  as a criterion, where  $b_{max}$  is from the mass center of the cell. [Dehnen \(2014\)](#) introduces a mass-dependent acceptance criterion, which suppresses the cost of  $\mathcal{O}(N^{0.86})$  less than a linear complexity. To demonstrate this approach, we choose the maximum side length of a source cell as  $L$ , the separation  $S_c$  is the distance between the geometric center of two cells and  $S_m$  is minimum distance between two cells (see Fig. 3).

Besides opening angle, our acceptance criterion needs to be adjusted for effects caused by truncation and it is summarized into three items:



**Fig. 4** Relation of error to opening angle. Relative error of gravity or acceleration is calculated via truncated FMM up to hexadecapole. The solid curve traces the rms error and dash-dotted curve signifies the maximum error of all particles.

- All interactions between cell-cell, cell-body and body-body are neglected when the minimum separation  $S_m$  is larger than the cutoff radius;
- Call M2L kernel to compute local expansion coefficients while opening angle  $\theta \leq \theta_{\text{MAC}}$ . Otherwise, open the larger one of two cells of interest;
- Enforce opening the cells if the separation is at the range of the transition region, even if the 2nd condition is already met, where transition region is defined by  $S_c > R_{\text{cutoff}} > S_m$ .

Our opening angle is defined as  $\theta \equiv L_B/S_c$ , where  $L_B$  is the maximum side length of a source cell. It is apparent that neighboring cells must be always opened according to the 2nd item.

The first item causes an essential improvement in the hybrid method. The additional 3rd item is due to truncated design. Figure 2 shows that the term  $r^{-1}$  is dominated beyond the truncated scale but the higher order multipoles still contribute the expansion coefficients of local gravitational field in the traditional FMM. However in our method, the multipoles at any order, here from  $f_0$  to  $f_4$ , are suppressed within the cutoff radius. Therefore no information can be propagated out the cutoff barrier no matter how many orders are considered. The error of short-range gravity also fails to be suppressed by a stricter  $\theta_{\text{MAC}}$  because the shadow region in Figure 3 is always ignored and lost due to the modification of prefactors, despite also contributing short-range gravity. The 3rd item is motivated to fix it and guarantee the higher multipole can influence the cells beyond the scale of cutoff radius.

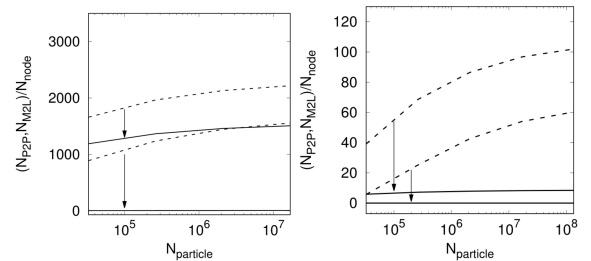
Figure 4 plots the relative error by the new MAC with the orders up to hexadecapole. The solid curve traces the root-mean-square (rms) of relative error and the dash-dotted curve signifies the envelop of maximum error. The error of gravity in our criterion with the third item can be

controlled well. It drops as the opening angle decreases. But the relation of error to opening angle changes. It breaks the power law of error with  $\theta^p$ , say  $\theta^4$  in hexadecapole, presented in Dehnen (2014).

The relation between accuracy and efficiency depends on the choice of control parameters. But it is difficult to derive the optimized parameters from a theoretical analysis. The order of FMM can be independently determined, which is constrained by the machine memory. The memory that cells consume doubles for every order that FMM adds. Some applications without a memory limit employ the FMM up to 10th order with a huge equivalent opening angle. We practically set the order of FMM up to octupole or hexadecapole, since cosmological simulations are usually pressed for memory. Correspondingly, it keeps sufficient statistics to set the opening angle from 0.3 to 0.4 for regular cosmological simulations. One can decrease the opening angle to improve accuracy and cause a sharp enlargement in the amount of computation. Therefore the parameters must be finely adjusted in accordance with computing power, environments and accuracy tolerance.

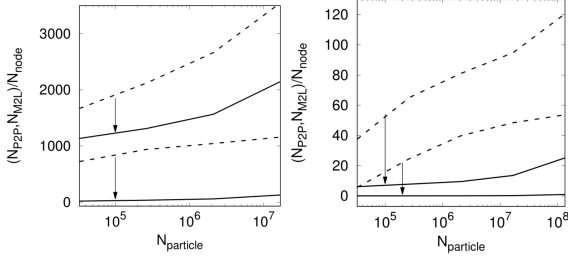
### 3 COMPARISON OF COMPUTATIONAL COST

In the previous section, we describe an algorithm for decoupling long-range and short-range gravity with a proper modification of MAC. Correspondingly, in this section we measure and confirm that the amount of computation is reduced. There are two extreme density distributions that affect cosmological evolution. In the high-redshift epoch, the density contrast is tiny. The computing box is almost uniformly filled with mass particles. With the evolution of the Universe, the initial seeds of structures grow and highly clustered structures are formed, including filaments and halos. Most particles fall into condensed regions.



**Fig. 5** Uniform density distribution. The dashed curves mark the counts of kernel P2P and M2L for a conventional FMM and the solid curves signify truncated FMM in this work. Each leaf (particle pack) only holds one particle in the *left panel*; Each leaf holds 32 particles at most in the *right panel*. Therefore, the length of tree cells is shorter than the *left panel*.





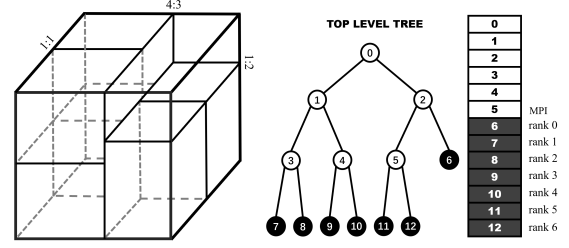
**Fig. 6** Highly clustered density distribution, which is the most condensed case from a  $\Lambda$ CDM cosmological simulation run at redshift  $z = 0$ . The dashed curves mark the interaction counts of kernel P2P and M2L for a conventional FMM and the solid curves signify truncated FMM in the work. Each leaf only contains one particle in the left panel; Each leaf contains 32 particles at most in the right panel.

The dashed curves correspond to the conventional case and the solid curves are in this work. Arrows demonstrate the reductions in the amount. The upper curves are for the P2P kernel and lower are M2L. There is apparent improvement, since the mean separation of most cells is beyond the cutoff radius so that the computation of M2L in original FMM vanishes for the uniform distribution, whose cost is close to the  $P^3M$  method in the uniform case.

Figure 5 demonstrates the analyses of the uniform case. The most time-consuming kernels of M2L and P2P are directly influenced by the truncated MAC. The run duration depends on the implementation but the count of interaction does not. It only depends on the choice of MAC. In the left panel of Figure 5, each leaf (particle pack) only holds one particle at most, and 32 particles are in the right panel. The count of interaction kernels determines the depth of tree and spatial configuration. The tree contains less cells in the right panel than right one in the same particle distribution.

Figure 6 demonstrates similar analyses for the cosmic structure distribution of the Universe at present (redshift  $z = 0$ ). The notation is the same as in Figure 5. As expected, the total count is larger than in the uniform case, because more particles are constricted into smaller cells so that more cells are opened by the acceptance criterion. The mean separation of cells shrinks below the cutoff radius, but the hybrid method still works to reduce the kernel count.

Compared with original FMM, kernel interactions of M2L and P2P are reduced in our truncated one. However an additional PM exists in our method. Theoretically, PM has cost of  $\mathcal{O}(N \log N)$ , which is worse than FMM, but a PM method can usually be more effectively implemented to save the overall duration. On the contrary, the original FMM makes use of Ewald summation to deal with



**Fig. 7** The domain decomposition and top level tree.

a periodic boundary condition (Gumerov & Duraiswami 2014). It is not needed in this hybrid method.

#### 4 PARALLEL IMPLEMENTATION OF ALGORITHM

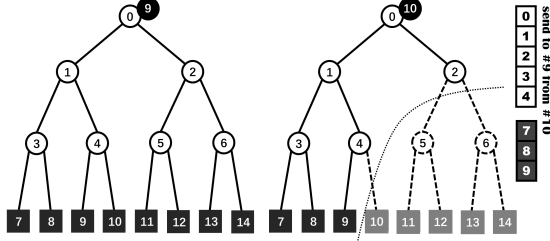
This PM-FMM method is employed by the cosmological  $N$ -body simulation code *photoNs-2*, which is designed for massively parallel cosmological simulations<sup>5</sup>. Its first version (Wang et al. 2018) adopts a parallel Tree-PM method and the interactions between particles and tree cells are arranged into a task pool, which is suitable for the optimization, especially on a heterogeneous platform (Makino 2004; Hamada et al. 2009; Wang et al. 2018; Iwasawa et al. 2018). This second version updates the short-range gravitational solver by truncated FMM described in the previous Section 2, instead of the tree method.

In the new version, the domain decomposition returns to an ORB tree across the computing nodes, such as computing sockets or processors with a shared memory. In addition, all domains and their upper nodes construct a top level tree that needs be stored in all computing nodes. Thus, the domain cell is the finest cell in the top level tree but a root cell for local essential tree (LET). The particles are also organized into a  $k$ -d tree in every computing domain so that a distributed global ORB tree is constructed. Figure 7 illustrates a domain decomposition for an almost uniform particle distribution. As an example of seven processes, the ORB tree firstly distributes the particles along the  $x$ -axis by the fraction of 4:3 in order to balance particle number, then four processes take charge of the left 4/7 volume of the box and another three processes do the right 3/7.

The upward pass of P2M and M2M firstly runs on the local tree. When all local upward passes are complete, the parallel M2M, M2L and L2L are made for the top level tree. But the inter-domain M2L and P2P still requires information across the other domains. In *photoNs-2*, we actively send local tree cells and leaves to the domain that

<sup>5</sup> A simplified parallel MPI+openMP version can be downloaded at <https://github.com/nulllike/photoNs-2.0> to test the hybrid algorithm.

may need it. Using our MAC, a traversal with respect to the closest boundary of target domain will find all potential cells and leaves involved in the target domain. For instance,



**Fig. 8** The local tree. The tree cells (0-4) and the particle packs (7-9) need to send to the target domain (Domain 9) in this case.

there are two local trees which are mounted on domain 9 and 10, respectively in Figure 8. Cell 2 may be accepted by MAC with respect to the “right side” boundary of domain 9 so that nodes 5 and 6 can be safely ignored (see Fig. 8), as are leaves from 11 to 14. The separation between leaf 10 and boundary may be larger than the cutoff radius so that leaf 10 is also ignored. Then a segment of subtree including all necessary information is arranged from the local tree of domain 10.

Extra memory to *send* and *receive* must be allocated to exchange the segment of cells and leaves for the communications. After those parallel operations, a local downward pass of L2L and L2P can be executed.

In this version, M2L and P2P operators are the most time-consuming two kernels for gravity calculation. For the criterion “children of parent’s brother”, the number of operations of M2L and P2P only depends on the length of the tree. But for the MAC in our implementation, it also depends on the clustering of particles. We also rely on a double-buffering task pool to improve the concurrency, for P2P and M2L. As the mass particles collapse into the potential well, the density contrast in the simulation box is wildly different from place to place. Therefore we estimate the workload for a single domain by counting the total number of M2L and P2P operations to determine how to redistribute the particles in the next time step. It is similar with the strategy of the code GreeM (Ishiyama et al. 2009; Ishiyama et al. 2012). Such a feedback strategy usually can control the workload imbalance within 15%.

Practically, this hybrid method is designed for massively parallel supercomputers with over  $10^4$  computing nodes or sockets. Its PM method needs to call the FFT subroutine two times at every single synchronized time step. A conventional FFT library, such as FFTW (Frigo & Johnson 2005), decomposes a mesh into a series of slices along a certain direction. It fails if the number of processes is larger than the number of sides of the mesh,

but it exactly happens in a cosmological simulation. In this version, we employ a Fortran library with a pencil decomposition, 2DECOMP&FFT (Li & Laizet 2010), for the PM. As a test, a simulation with a  $\sim 5000^3$  grid is carried out by over 20 000 processes.

## 5 SUMMARY AND DISCUSSION

In this paper, we investigate a hybrid method for the massive application of cosmological simulations. In the epoch of precision cosmology, FMM with complexity  $\mathcal{O}(N)$  is a key method to run high resolution simulations on supercomputers and a traditional PM method still contributes to decoupling the gravity and dealing with the periodic boundary condition. The hybrid algorithm of FMM with PM keeps the benefit of gravity splitting and decreases the amount of computations.

Specifically, we modified the operators of the truncated FMM for short-range gravity and provide a general form to compute the prefactor of multipoles. We focus on a Gaussian-type truncation. Because its sharp splitting is proven by the TreePM method, in principle, one can choose another truncated function, instead of the exponential form. A polynomial function can be calculated more efficiently for the numerical mathematical library than an exponential one. The modifications of their prefactors can be generated by a similar procedure. The method in this work is different from the Particle Mesh Multipole Method (PMMM), which calls  $(p + 1)^2$  FFT to directly compute the multipole coefficients (Nitadori 2014) or Fourier Transform on Multipoles (FTM) method (Ong et al. 2004). We do not use a Fourier transformation to calculate the coefficients of multipole expansion but rather gravitational potential.

Moreover, MAC needs be modified for two additional conditions. One is for truncation of long-range interaction and the other is for controlling the accuracy. The count of kernel interactions is determined by the details of implementation of FMM and MAC. A conventional FMM has a linear stability and Dehnen (2002) reports a better stability by implementing a mass dependent MAC. In this work, the hybrid method we demonstrate costs more than  $\mathcal{O}(N)$  but less than  $\mathcal{O}(N \log N)$ . The reduction of kernel computation is due to decoupling the long-range force so that such a hybrid method can robustly work for other kinds of traversal and tree construction as well.

Finally, the current and next generation supercomputers provide a powerful numerical platform to run the massive simulations with unprecedented resolutions and simulation boxes, which usually are composed of tens of thousands of computing nodes and various heterogeneous accelerators and many-core architectures. Besides the pressure of memory, I/O band and storage of snapshot,

it requires appropriate algorithms designed for massive concurrency to face the challenges of software scalability and computing performance, especially on heterogeneous devices. The  $N$ -body applications exchange an enormous number of particles among processes so that the communication strategy becomes an essential issue. The other trend is to employ a more efficient method, such as  $\mathcal{O}(N)$ , to deal with the extreme amount of force computation. The method we proposed provides an option to calculate the force efficiently and decouple global communication in the meantime. The eventual performance of applications depends on the algorithm and the implementation of programming details. Here, we release a fundamental version of code to verify the precision and validity of the hybrid algorithm, which is expected to be optimized on high performance computers.

**Acknowledgements** We gratefully thank the anonymous referee for the helpful comments. We acknowledge the support from the National Key Program for Science and Technology Research and Development (2017YFB0203300) and the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No. XDC01040100.

## References

- Abbott, T. M. C., Abdalla, F. B., Annis, J., & et al. 2018, *MNRAS*, 480, 3879
- Bagla, J. S. 2002, *Journal of Astrophysics and Astronomy*, 23, 185
- Bagla, J. S., & Ray, S. 2003, *New Astron.*, 8, 665
- Barnes, J., & Hut, P. 1986, *Nature*, 324, 446
- Bédorf, J., Gaburov, E., Fujii, M. S., et al. 2014, in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC '14* (Piscataway, NJ, USA: IEEE Press), 54
- Berlind, A. A., & Weinberg, D. H. 2002, *ApJ*, 575, 587
- Cheng, H., Greengard, L., & Rokhlin, V. 1999, *Journal of Computational Physics*, 155, 468
- Cole, S., Lacey, C. G., Baugh, C. M., & Frenk, C. S. 2000, *MNRAS*, 319, 168
- Dehnen, W. 2000, *The Astrophysical Journal Letters*, 536, L39
- Dehnen, W. 2002, *Journal of Computational Physics*, 179, 27
- Dehnen, W. 2014, *Computational Astrophysics and Cosmology*, 1, 1
- DES Collaboration. 2019, *Phys. Rev. Lett.*, 122, 171301
- Efstathiou, G., Davis, M., White, S. D. M., & Frenk, C. S. 1985, *ApJS*, 57, 241
- Eisenstein, D. J., & et al. 2005, *ApJ*, 633, 560
- Frigo, M., & Johnson, S. G. 2005, in *Proceedings of the IEEE*, 216
- Greengard, L., & Rokhlin, V. 1987, *Journal of Computational Physics*, 73, 325
- Gumerov, N. A., & Duraiswami, R. 2008, *J. Comput. Phys.*, 227, 8290
- Gumerov, N. A., & Duraiswami, R. 2014, *Journal of Computational Physics*, 272, 307
- Guo, Q., White, S., Boylan-Kolchin, M., et al. 2011, *MNRAS*, 413, 101
- Habib, S., Morozov, V., Frontiere, N., et al. 2013, in *SC '13 Proceedings of SC13: International Conference for High Performance Computing, Networking, Storage and Analysis*, Article id. 6, 6
- Habib, S., Pope, A., Finkel, H., et al. 2016, *New Astron.*, 42, 49
- Hamada, T., Narumi, T., Yokota, R., et al. 2009, in *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis, SC '09* (New York, NY, USA: ACM), 62:1
- Hockney, R. W., & Eastwood, J. W. 1988, *Computer Simulation Using Particles*
- Ishiyama, T., Fukushige, T., & Makino, J. 2009, *PASJ*, 61, 1319
- Ishiyama, T., Nitadori, K., & Makino, J. 2012, in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, SC '12* (Los Alamitos, CA, USA: IEEE Computer Society Press), 5:1
- Iwasawa, M., Wang, L., Nitadori, K., et al. 2018, in *Computational Science – ICCS 2018* (Cham: Springer International Publishing), 483
- Jing, Y. P., & Suto, Y. 2002, *ApJ*, 574, 538
- Kaiser, N. 1987, *MNRAS*, 227, 1
- Kitaura, F.-S., Rodríguez-Torres, S., Chuang, C.-H., et al. 2016, *MNRAS*, 456, 4156
- Li, N., & Laizet, S. 2010, *Cray User Group 2010 Conferences*, Edinburgh
- Makino, J. 2004, *Publications of the Astronomical Society of Japan*, 56, 521
- Nitadori, K. 2014, *arXiv e-prints*, arXiv:1409.5981
- Ong, E. T., Lee, K. H., & Lim, K. M. 2004, *International Journal for Numerical Methods in Engineering*, 61, 633
- Peebles, P. J. E., & Yu, J. T. 1970, *ApJ*, 162, 815
- Planck Collaboration. 2018, *arXiv e-prints*, arXiv:1807.06209
- Portegies Zwart, S., McMillan, S. L. W., van Elteren, E., Pelupessy, I., & de Vries, N. 2013, *Computer Physics Communications*, 184, 456
- Potter, D., Stadel, J., & Teyssier, R. 2017, *Computational Astrophysics and Cosmology*, 4, 2
- Rozo, E., Wechsler, R. H., Rykoff, E. S., et al. 2010, *ApJ*, 708, 645
- Springel, V. 2005, *MNRAS*, 364, 1105
- Springel, V., White, S. D. M., Jenkins, A., et al. 2005, *Nature*, 435, 629
- Wang, Q., Cao, Z.-Y., Gao, L., et al. 2018, *RAA (Research in Astronomy and Astrophysics)*, 18, 062
- Warren, M. S. 2013, *arXiv:1310.4502*
- Yokota, R., & Barba, L. A. 2011, *arXiv:1110.2921*