doi:10.11887/j.cn.202405023

http://journal. nudt. edu. cn

通过包络面重构的大规模粒子并行绘制算法

王华维^{1,2*},艾志玮^{1,2},曹 轶^{1,2}

(1. 北京应用物理与计算数学研究所 计算物理重点实验室,北京 100088;

2. 中物院高性能数值模拟软件中心,北京 100088)

摘 要:针对大规模粒子高表现可视化需求,提出基于包络面重构的大规模粒子并行绘制算法。该算法 以连续曲面的形式表示,绘制大规模粒子的团簇表面及其物理量分布。对算法进行了分布式并行化,从而可 以通过大规模并行来处理亿以上规模的粒子数据。在算法实现上,还解决了并行计算时的块间裂缝问题,并 提出了快速查找邻域粒子的方法,同时,基于可见性对粒子数据进行剔除,提高了绘制效率。由此,可以通过 带光照效果的光滑曲面来高表现展示大规模粒子数据中的团簇结构及其物理量分布。实验结果表明,该算 法在 512 核上可在 5 s 内完成上亿粒子的绘制,并行效率可达 60%。该算法已成功应用到大规模并行非平衡 分子动力学模拟等实际模拟应用中。

关键词:粒子可视化;包络面;距离场;分布式并行化;可见性剔除 中图分类号:TP391 文献标志码:A 文章编号:1001-2486(2024)05-219-09



Parallel rendering algorithm for large-scale particles by wrapping surface reconstruction

WANG Huawei^{1,2*}, AI Zhiwei^{1,2}, CAO Yi^{1,2}

(1. Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing 100088, China;

2. CAEP Software Center for High Performance Numerical Simulation, Beijing 100088, China)

Abstract: A parallel rendering algorithm based on wrapping surface reconstruction was proposed for large-scale particles in distributed environments so as to visualize the particles in high quality. In the algorithm, particle clusters were represented and then rendered in the form of a series of continuous surfaces, where the distribution of the physical variable was also shown. The algorithm was parallelized in distributed environments, thus more than a hundred million particles can be visualized using a lot of processing cores. In terms of algorithm implementation, the issue of inter-block cracks during parallel computation was be solved, and the method for rapidly finding adjacent particles was presented. Meanwhile, based on visibility culling, the particle data was filtered and thus the rendering efficiency was improved. As a result, smooth surfaces with lighting can be used to expressively exhibit inner structures and physical variable distributions of particle clusters for large-scale particles. Experiment results demonstrate that using the proposed algorithm, the rendering of more than 100 million particles is realized in 5 seconds on 512 processing cores with about 60% parallel efficiency. The proposed algorithm has been successfully applied to practical simulation applications such as massively parallel non-equilibrium molecular dynamics simulations.

Keywords: particle visualization; wrapping surface; distance field; distributed parallelization; visibility culling

粒子模拟出现在材料、工程力学、电磁环境、 激光聚变、裂变能源等多个应用领域中,是科学计 算的重要形式之一。其中一类粒子模拟方法是对 连续物质空间抽取样本,利用一系列样本粒子来 模拟物质的运动变化规律,然后通过统计或积分 分析物质的宏观性质。这类模拟方法的优点,即 非常适合模拟物质的畸变、断裂、破碎等大变形问题,常用方法包括分子动力学、光滑粒子流体动力 学、物质点法等。在这类模拟方法中,人们不断增 加粒子的抽样密度以实现更高的计算精度,由此 输出急剧膨胀的粒子数据集,大规模粒子数据给 可视化系统带来了极大的挑战。

收稿日期:2022-05-12

基金项目:国家重点研发计划资助项目(2017YFB0202203)

^{*}第一作者:王华维(1976—),男,湖北云梦人,副研究员,博士,硕士生导师,E-mail:wang_huawei@iapcm.ac.cn

引用格式:王华维,艾志玮,曹轶. 通过包络面重构的大规模粒子并行绘制算法[J]. 国防科技大学学报, 2024, 46(5): 219-227.

Citation: WANG H W, AI Z W, CAO Y. Parallel rendering algorithm for large-scale particles by wrapping surface reconstruction [J]. Journal of National University of Defense Technology, 2024, 46(5): 219 – 227.

在目前的可视化系统中,常常用球体表示一个个粒子,并以离散化球面的方式送入 OpenGL 绘制。根据离散化精度的不同,一个球面可能对应生成 10~1000 个 OpenGL 绘制图元。对于数千万甚至数亿粒子,绘制中 OpenGL 需要处理的图元量可达到数亿至数百亿以上,这对目前的并行绘制系统来说是难以承受的。

在面对大规模粒子时,离散球面绘制形式不 仅交互速度慢,而且大量小球极度混杂、相互遮 挡,难以高质量展现模拟结果中的三维结构,不能 满足材料破坏、力学毁伤等研究中的可视化需求。 此外,粒子模拟的原连续物质虽然可能畸变、断裂 甚至破碎,但在空间中仍然呈现宏观上的复杂形 状与结构特征,也需要不同于全离散化小球的可 视化表现形式。因此,目前亟须发展针对大规模 粒子的高表现可视化算法,以清晰展现粒子模拟 中的三维结构特征及其演化发展规律。

一个主要的对策,就是对粒子团簇的外表面 进行曲面重构,通过绘制外表面的方法来表达粒 子团簇的空间结构。这类方法以基于距离场的团 簇表面重构方法为代表^[1-3],目前已可生成光顺 的团簇表面几何,可视化表现力较高,然而其算法 实现或者是串行的,或者是在一个 CPU 的多核或 者单 GPU 上基于共享内存并行的,限于内存容 量,难以处理大规模粒子数据。

单机/单结点或者单 GPU 的处理能力是有限 的,难以适应规模快速增长的粒子数据可视化需 求,此外,在部分国产高性能计算机上也没有配置 GPU,因而,面向超大规模粒子数据,发展可扩展 的分布式并行粒子绘制算法是很有必要的。为 此,本文提出了基于包络面重构的大规模粒子并 行绘制算法。该算法以一系列连续曲面的形式表 示大规模粒子数据中的一个个团簇,并在团簇表 面上绘制出粒子数据的物理量值分布。本文对算 法进行了分布式并行化,从而可以通过大规模并 行来处理上亿甚至更大规模的粒子数据。同时, 本文从多个方面对算法进行了优化,以提升大规 模粒子的绘制处理效率。

1 相关工作

最常用的粒子绘制方法,就是将粒子看作三 维点球,将其表面离散成三角形网格,绘制出真实 感光照效果。该方法适合于绘制少量的粒子数 据,有时也用于粗略绘制出大规模粒子的空间分 布形态。针对大规模粒子绘制效率低的问题, Wang 等^[4]提出了一个基于不可见剔除的筛选绘 制方法,可剔除大量的被遮挡粒子,显著提升绘制 效率。另外,研究者也在不断提升粒子的绘制效 果,例如可基于 OSPRay 的光线跟踪方法^[5],绘制 出具有真实感光影效果的粒子场景。然而,绘制 大规模粒子时,视野中粒子极多,每个粒子占有屏 幕区域极小,不仅难以呈现高级光影效果,其绘制 也是极其费时的。

用离散球面表示一个个粒子,在处理大规模 粒子时,其绘制效果难以自然形成团簇整体上的 光滑表面效果。因而,引入体绘制方法来融合绘 制粒子数据,形成比较连续的空间形态。抛雪球 法将每个粒子看成具有深度、颜色但无几何连接 的点元,投影到屏幕空间进行融合与插补^[6]。 Wang 等^[7]改进了该方法,在反向过程中优化梯 度设计,正向过程中规则化点元分布,取得了很好 的点云表面绘制效果。Fraedrich 等^[8]则利用插 值核函数从粒子集合重构数据场,进而实施光线 投射体绘制,可有效展现粒子集合中的三维体特 征。在此基础上, Hochstetter 等^[9]进一步通过双 向自适应采样,提升了光线投射体绘制的速度,以 快速绘制动态变化的粒子集合。然而,体绘制生 成的团簇边界常常显得比较模糊,而且与视点相 关,不利于对团簇结构做进一步分析。

近年来,基于团簇外表面重构的粒子可视化 方法得到了更多的关注。这类方法可生成连续的 团簇外表面几何,也可导入高级绘制工具,由此来 高表现展示粒子团簇的内部结构和空间分布。在 这类方法中,生成表面的光顺效果、对复杂拓扑结 构的适应性以及算法处理的速度是主要考虑的 问题。

第一种方法是液滴表面生成方法^[10],该方法 将粒子看成一个个的液滴球体,当粒子靠近时液 滴自然融合,由此逐步形成越来越复杂的液体形 态。该方法对少量粒子效果很好,但在处理大规 模粒子表示的平整曲面时,例如一块平面、锥面或 球面等,由于每个粒子附近都有明显的小球体隆 起感,因而生成曲面光顺性差,视觉上负面效应明 显。虽然 Müller 等^[11]通过密度平均方法降低了 这种效应,但无法从根本上消除平整表面上的隆 起感问题。

第二种方法是对粒子数据重构标量场,通 过提取隐式曲面的方法来获取粒子团簇包络 面。这里,标量场重构计算常常是影响表面质 量和计算时间的关键因素。Zhu和Bridson^[1]将 带符号距离场从单粒子球推广到了大量粒子的 团簇,通过从带符号距离场中提取等值面的方 法来构造、生成粒子团簇的外表面,但该方法用 于凹的粒子团簇则容易出错。Solenthaler等^[2] 通过在距离场中加入校正因子来处理这个错误,可较好地适应粒子团簇的复杂结构,生成质 量较高的团簇外表面。

针对标量场重构, Akinci 等^[3] 基于 OpenMP 对重构算法进行了并行化,取得了较高的曲面重 构速度。基于三层自适应格式, Akinci 等^[12] 又对 算法进行了优化, 在高曲率区域用高分辨率网格 保持精度, 在低曲率区域则用低分辨率网格减少 计算量。基于 GPU 加速, Huang 等^[13]和 Du 等^[14] 对曲面重构算法做了进一步的扩展与应用。Wu 等^[15]则基于两层网格优化了曲面重构算法, 粗层 使用哈希方法索引曲面粒子, 细层实施标量场重 构, 由此加快了重构速度。近年来, Yang 和 Gao^[16]对曲面重构算法的全过程进行了并行化与 优化, 重构速度提升了一个量级。然而, 上述实现 都是在一个 CPU 的多核或者单 GPU 上基于共享 内存完成的, 由于资源有限, 难以处理大规模粒子 数据。

在标量场重构中,Yu和Turk^[17]还引入了各向异性核函数,其方向基于主成分分析计算,但该方法需要邻域粒子搜索,因而非常耗时,而且仍会出现凹区域出错问题。Wang等^[18]进一步将基于各向异性核的曲面重构引入粒子多相流模拟,以抽取由水平集定义的流体表面和多相界面,得到了高质量曲面重构效果。在各向异性核曲面重构中,Chen等^[19]提出了检测外围曲面顶点的方法,可只在曲面外围区域构造标量场,由此节省了一半的计算时间。

进一步工作还集中于边界粒子优化处理方 面,用于提高曲面重构的精度和效率,包括基于空 间哈希网格的重构策略^[20]、边界粒子自适应重采 样技术^[21]、粒子预处理缺陷修复技术^[22]、基于区 间分析的边界粒子检测方法^[23]、流体界面的层次 边界检测方法^[24]、改进的基于邻域粒子分布各向 异性粒子变换方法^[25]、基于离散指示函数的流体 粒子表面重建方法^[26]等。

2 算法流程

针对大规模粒子高表现可视化需求,本文提 出了基于包络面重构的大规模粒子并行绘制算 法。该算法对三维空间离散点集生成拓扑二维的 连续曲面,对多个点集团簇则对应生成一系列连 续曲面。首先,大规模粒子数据是分块存储在并 行文件系统上的,本文基于可见性剔除只读入可

见区域中的各个粒子数据块(patch)。为了避免 计算区域交叠造成距离场计算的不一致性,本文 将计算区域重新划分给各个处理器核,并重新分 发粒子数据,各处理器核收集落于增广区域之内 的所有粒子。进而,各处理器核对所负责的计算 区域构造均匀网格分块,并根据局域粒子位置关 系计算一个带符号距离场。计算中,为了快速查 询局域粒子,本文让粒子与网格关联并有序组织。 为了节省计算开销,将靠近目标曲面的粒子判定 为曲面粒子,曲面粒子附近的网格标记为计算有 效网格,后续只对有效网格计算距离场并抽取等 值面。然后,利用移动立方体(marching cubes, MC)方法^[27]从距离场网格中抽取值为0的隐式 曲面,即为所得曲面。最后,基于就近法则,将粒 子上的物理量赋值给所得曲面,即可基于云图方 法完成后续的连续曲面光照绘制。该算法的流程 如图1所示。下文将具体介绍该算法中的主要关 键技术。





3 粒子分布式载入

3.1 粒子可见性剔除

本文算法的输入是大规模粒子模拟输出的结 果数据,这些数据常常是分块存储于并行文件系 统中,分块数目可以达到上万个甚至更多。本文 首先基于可见性对粒子数据进行筛选剔除,利用 OpenGL 的视景锥对数据场进行粗筛,将落在视 景锥之外因而对用户不可见的数据块剔除(见 图2)。OpenGL使用视景投影变换将三维空间中 的几何元素映射到屏幕像素区域,只有位于视景 锥之内的几何元素才可能最终出现在屏幕上。那 些整个位于视景锥之外的数据块,包括由此重构 出来的包络面特征数据,对观察者来说是不可见 的,因而,在读数据时无须读取它们,在绘制时更 不需要处理它们。





具体地,本文基于合约机制^[28]实现大规模粒 子数据的可见性剔除。首先,从元数据信息获取 数据块序列的空间包围盒,构建一棵空间区间树, 并根据当前绘制窗口的视点参数构建视景锥。其 次,从根节点开始,递归处理空间区间树:当某节 点的包围盒全部在视景锥之外因而对视景锥不可 见时,则该节点之下的所有叶子(数据块)均不可 见,停止遍历其子节点;否则,该节点的包围盒与 视景锥有交,继续对该节点的子节点进行可见性 判断,如此递归下去,直至到达叶子。最终得到所 有与视景锥有交的数据块列表,将它们的读入需 求写入合约,即可在可视化流程的下行执行阶段 实际只读入这些数据块。

3.2 粒子重新分发

在分布式并行环境下,本文基于"数据并行" 原则,将经过可见性剔除之后的数据块均衡分配 给各个处理器核,一般采用连续分配策略。如果 数据块数目为 M,处理器核数目为 N,则连续分配 策略将把数据块列表划分为 N 段,依次把数目为 M/N 的数据块分配给处理器核 P₀,P₁,…,P_{N-1}。 然后,各个处理器核按照所分配的数据块号,实际 读入各个数据块,并送入后续处理流程。 在后续处理中,各个处理器核对读入的数据 块构造带符号距离场 φ(r)并分块抽取等值面 φ(r) =0。为了保证所得等值面不重不漏,分布 式计算中每个隐函数数据场(带符号距离场)分 块应是空间中不重叠,为了算法高效,本文要求各 个分块的空间包围盒不重叠。然而,从文件系统 读入的粒子初始分块不能保证这一点,不同分块 的粒子子集可能纠缠交错,致使其空间包围盒重 叠,见图3。因而,本文需要对粒子数据进行重新 划分,并分发给每个处理器核。



图 3 初始数据中两个分块(不同颜色标出) Fig. 3 Two initial data patches (in different colors)

具体地,首先并行收集全体粒子的实际空间 包围盒,将它递归划分为多个长方体子区域,指定 为各核的计算区域。然后,将该子区域在各个维 度的两侧各扩大 R(粒子影响半径)的距离,并对 各自增广区域收集落于其内的所有粒子,见图4。 这样,后续计算各个子区域带符号距离场时,其局 域的所有粒子皆在本地,由此保证区域边界附近 的计算结果在所有处理器核上是一致的。这样, 分布式计算的数据场以及后面的等值面都能保证 分块附近的一致性。



图 4 子区域粒子收集二维示意图 Fig. 4 Sketch map for collecting particles in an extended region

粒子数据重新分发的优点在于,即使初始数 据块的数目少于处理器核数,或者甚至只有一个 数据块,都可以重新划分子区域,并均衡地分配给 各个处理器核,同时,由于对子区域进行了扩展, 使得各个子区域的距离场重构与等值面抽取都可 以独立进行,无须通信中间结果,也没有块间裂缝 问题。

4 粒子有效管理

4.1 粒子有序组织

在距离场计算中,对距离场每个节点都有寻 找附近粒子的需要。若每次都遍历所有粒子,判 断并找寻距离小于阈值的粒子,则这样的计算复 杂度就太高了。因而,本文先对各个粒子,计算其 所在单元的编号;然后依据单元编号,对所有粒子 进行排序(使用快排算法,复杂度为O(nlgn)); 接着,对每个单元,本文记录其内粒子的最小序号 $k = k_{min}$,若其粒子的最大序号为 k_{max} ,则下一个单 元记录的最小序号是 k_{max} +1,即使它是一个空单 元(不包含粒子)。由此,对任何一个单元*i*,立即 知道其内的粒子序号为 $\{k | k_i \leq k < k_{i+1}\}$,其中 k_i 和 k_{i+1} 分别是单元*i*和下一个单元*i*+1中记录的 粒子最小序号。

粒子有序组织之后,在具体计算一个网格节 点的带符号距离场标量值时,本文会根据影响半 径 R 确定附近的相关单元(对均匀网格而言,复 杂度为 O(1)),然后根据单元访问其内的所有粒 子,由此可以找到所有的局域粒子,避免了每次都 对全部粒子遍历找距离满足阈值者。

4.2 标记曲面粒子

为计算区域构造一个光滑颜色场[11],即

$$C(\mathbf{r}) = \sum_{j} \frac{m_{j}}{\rho_{j}} W(\|\mathbf{r} - \mathbf{r}_{j}, h\|) \qquad (1)$$

其中,**r**为位置向量,**r**_j为位置**r**影响区域内的 所有粒子,**W**(·,h)为影响半径为h的一个核 函数, m_j 和 ρ_j 分别是粒子的质量和密度。该颜 色场在粒子团簇边界处变化剧烈,因而其梯度 较大,也就是说,当一个粒子处的梯度大于某 个阈值,即

$$\|\nabla C(\mathbf{r}_i)\| > l \tag{2}$$

则它一定位于边界附近,称为曲面粒子。此处,本 文取 W(s,h) = g(s/h),其中

$$g(s) = \max(0, (1 - s^2)^3)$$
 (3)

同时,令所有粒子都有相同的质量和密度,所 以不妨设 $m_j = 1$, $\rho_j = 1$ 。于是,梯度计算公式为

$$\nabla C(\mathbf{r}) = \sum_{j} g'(\|\mathbf{r} - \mathbf{r}_{j}\|/h) \frac{\mathbf{r} - \mathbf{r}_{j}}{h\|\mathbf{r} - \mathbf{r}_{j}\|} \quad (4)$$

根据这些公式,将曲面粒子标记出来,然后将 曲面粒子附近的网格判定为计算有效网格(见 图5),也就是说,后续只对曲面粒子附近的网格 节点计算上述距离场,也只在曲面粒子附近进行 等值面抽取计算,由此减少计算开销。下文中,梯 度阈值 *l* 使用经验值 0.125,用户也可根据粒子的 疏密程度进行交互调节。





5 粒子团簇包络面重构

5.1 带符号距离场计算

均,即

引入带符号距离场标量场^[1-2],它将单个粒 子球面的距离场推广到大量粒子的集合,由此即 可用值为0的隐式曲面描述粒子团簇的包络面。 带符号距离场标量场的重构质量是决定团簇表面 生成质量的关键。具体地,首先在计算区域布置 一套均匀网格,然后在每一个节点位置计算带符 号距离场值。带符号距离场在一个网格节点r位 置的值按以下公式计算:

$$\phi(\mathbf{r}) = \|\mathbf{r} - \bar{\mathbf{r}}\| - rf \tag{5}$$

其中,r为平均粒子半径或者光滑粒子的平均距离,f是一个校正因子,则

$$f = \begin{cases} 1 & EV_{\max} < t_{low} \\ \gamma^3 - 3\gamma^2 + 3\gamma & \ddagger \psi \end{cases}$$
(6)

其中, $\gamma = \frac{t_{high} - EV_{max}}{t_{high} - t_{low}}$, $t_{high} = 3.5$, $t_{low} = 0.4$, EV_{max} 为 $\nabla_r(\bar{r})$ 的最大特征值, \bar{r} 是局域粒子的加权平

$$\bar{\boldsymbol{r}} = \frac{\sum_{j} \boldsymbol{r}_{j} g(\|\boldsymbol{r} - \boldsymbol{r}_{j}\|/R)}{\sum_{j} g(\|\boldsymbol{r} - \boldsymbol{r}_{j}\|/R)}$$
(7)

其中, *R* 为带符号距离场计算中的粒子影响半径, *g*(*s*)为加权平均的核函数。即

$$g(s) = \max(0, (1 - s^{2})^{3})$$
(8)

$$\nabla_{r}(\bar{r}) \not \to \uparrow 3 \times 3 \text{ in } \# \mathbb{R}, \mathbb{C} \text{ in } \to \uparrow \neg \mathbb{R} \not \to f \\ \frac{\partial \bar{r}^{n}}{\partial r^{m}} = \frac{\sum_{j} r_{j}^{n} (r^{m} - r_{j}^{m}) p(\|\mathbf{r} - \mathbf{r}_{j}\|/R)}{\sum_{j} g(\|\mathbf{r} - \mathbf{r}_{j}\|/R)} - \frac{\sum_{j} r_{j}^{n} g(\|\mathbf{r} - \mathbf{r}_{j}\|/R) \sum_{j} (r^{m} - r_{j}^{m}) p(\|\mathbf{r} - \mathbf{r}_{j}\|/R)}{\left[\sum_{j} g(\|\mathbf{r} - \mathbf{r}_{j}\|/R)\right]^{2}}$$

$$\frac{\operatorname{He}_{r}(\bar{r}) = (\bar{r}^{1}; \bar{r}^{2}, \bar{r}^{3})^{\mathrm{T}}; \mathbf{r} = (r^{1}, r^{2}, r^{3})^{\mathrm{T}}; \mathbf{r}_{j} = (r_{j}^{1}, r_{j}^{2}, r_{j}^{3})^{\mathrm{T}}; \mathbf{r}_{j} = (r_{j}^{1}, r_{j}^{2}, r_{j}^{3})^{\mathrm{T}}, p(s) = -6 \times (1 - s^{2})^{2}/R^{2}; \nabla_{r}(\bar{r}) = \left(\frac{\partial \bar{r}^{n}}{\partial r^{m}}\right)_{3 \times 3}, n, m \in \{1, 2, 3\}_{\circ}$$

5.2 粒子量赋值曲面

· 224 ·

当带符号距离场计算完毕后,即可用 MC 方 法抽取值为0 的等值面,也即对大规模输入粒子 各个团簇外表面构造连续曲面。MC 方法为经典 的等值面抽取方法,在此无须赘述。

以上计算只是得到了粒子团簇包络面的几何 信息。显然,原来粒子一般是带有物理量的,表征 了粒子所在位置的物理属性,这一属性有时也需 要表现在绘制曲面上。为此,本文对所得曲面的 每一个顶点赋上物理值。具体地,本文以近似广 度优化顺序扫描顶点周围的网格单元,找到其中 的粒子,并比较距离最小者,由此快速得到局域最 近粒子,将该粒子的物理量赋给该曲面顶点。注 意,在之前粒子数据核间分发时,这些物理量也需 一并包含。

至此,针对大规模粒子数据,既构造了几何包 络面,又赋上了物理量,因而可以基于云图方式绘 制该粒子数据,并通过光滑曲面光照效果来提升 其表现力。

6 实验结果

将本文算法应用到大规模并行非平衡分子动 力学模拟,该模拟研究分析衰减冲击载荷下金属 熔化破碎机理。模拟输出的数据包含约600万粒 子,在浪潮集群一个结点24个处理器核上完成绘 制,如图6所示。由于物理空间是一个非常细长 的区域,此处只给出局部放大图,以便更好观察绘 制效果。本文对两种方法进行了比较,一是传统 的粒子球绘制方法,一是本文基于包络面绘制方 法。从图中可以看出,基于包络面绘制方法可以 清晰展现熔化金属破碎后形成的各种团簇外表面 形态,而且,通过透明化绘制(见图7)可以更好地 观察熔化金属由连续体断裂破碎至液态颗粒时三 维空间复杂分布。相比较而言,粒子球绘制方法



则因大量小球极度簇拥,团簇表面的方向杂乱无 章,因而绘制效果显得暗淡无光,无法自然呈现清 晰光洁的团簇表面,又因不支持透明绘制而无法 看清被遮挡的内部结构。

为了测试大规模粒子绘制性能,本文将该算 法用于大体系动态破坏过程分子动力学实际模拟 结果的可视化,该模拟输出结果中粒子数目达到 1.2亿,包含103个时间步,时变数据总量达到 736 GB。并行性能测试分别使用了 32、64、128、 256 和 512 个处理器核,测试结果见图 8、图 9 和 图 10。本文测试了粒子包络面并行绘制算法的 时间开销以及采取可见性剔除方法后的时间开 销,图8展示了并行实现中可见性剔除带来的性 能提升效果,从图中可以看出,在使用不同数目处 理器核的情况下,可见性剔除均可以提升9倍以 上的绘制速度。在512 核上,可见性剔除带来的 性能提升,将粒子绘制时间降低到了5s以内。 同时,本文也测试了算法的并行可扩展性,如图9 和图 10 所示。当处理器核数不断增加时,绘制时 间随之快速下降,如图9所示,本文相应计算了算



图 8 并行实现中可见性剔除带来的性能提升 Fig. 8 Rendering performance enhanced based on visibility culling in the parallel testing





parallel rendering algorithm



图 10 并行绘制算法的并行效率

Fig. 10 Parallel efficiencies of the proposed distributed parallel rendering algorithm

法的并行效率,如图 10 所示,从图中看出,本文算 法的并行效率在 60% 以上,这显示了算法良好的 并行可扩展性。

尽管如此,本文算法仍然存在一定的不足。 从图 10 可以看出,当处理器核数不断增加的时候,并行效率虽然在 60% 之上,但其数值还是在 不断地下降着。并行效率下降的原因,除了读数 据文件、消息传递接口标准(message passing interface, MPI)聚合通信等的扩展性较差,还在于 增广区域带来的计算量增加。当处理器核数增加 时,各核处理的子区域持续变小,但仍然要在各个 维度两侧增加 *R* 的扩展区域,这使得总计算量实 际在不断增加,影响了算法的可扩展性。

扩展区域的负面影响在图 8 所示的可见性剔除效果中也有所体现。图 8 显示,可见性剔除带来的性能优化效果随核数增加而下降。这是由于使用了可见性剔除后,各核上处理子区域变小了,但由于始终带着宽度 R 的扩展区域,使得可见性剔除前后计算量的比值打了折扣。随着处理器核的增加,这个折扣后的比值越来越小。例如,理想情况下,可见性剔除前后一个核上子区域分别设为 $D \times D \times D$ 和 $d \times d \times d$,D > d,实际计算量为 $(D+2R)^3 \pi (d+2R)^3$,若核数增加 8 倍,对应各维度子区域减半,则实际计算量分别变为 $(D/2 + 2R)^3 \pi (d/2 + 2R)^3$,显然

$$\frac{D+2R}{d+2R} > \frac{D+4R}{d+4R} = \frac{D/2+2R}{d/2+2R}$$
(9)

所以

$$\left(\frac{D+2R}{d+2R}\right)^{3} > \left(\frac{D/2+2R}{d/2+2R}\right)^{3}$$
 (10)

若核数继续增加,则该比值持续变小。计算量比 值的减小,一定程度上造成了加速比的减小。

7 结论与展望

针对大规模粒子高表现可视化需求,本文提 出了基于包络面重构的大规模粒子并行绘制算 法,该算法从带符号距离场中提取,绘制粒子团簇 的外表面。在分布式并行实现时,该算法首先基 于可见性对粒子数据进行剔除,只处理可见区域 内的重构计算,由此节省了大量的内存和时间开 销;其次,对增广区域完成粒子收集与距离场计 算,避免包络面并行生成时的块间裂缝问题。进 一步,各核在计算带符号距离场时,令粒子与网格 关联并有序组织,以快速查询邻域粒子。在抽取 曲面几何后,基于就近法则,将粒子上的物理量赋 值给所得曲面。由此,即可基于曲面光照效果来 高表现展示粒子团簇的空间结构及其物理量值 分布。

该算法已成功应用到分子动力学等大规模实际模拟,对上亿粒子进行高表现可视化,可有效表现清晰光洁的粒子团簇表面,并通过透明化显示出内部/背后被遮挡的结构,绘制效果显著好于传统的粒子球绘制,有力支撑了相关应用模拟结果的后处理可视分析。通过高效并行实现策略,该算法取得了 60% 的并行效率,在 512 核上实现了上亿粒子5 s 以内完成绘制。

本文算法仍存在一定的不足,未来可能在以 下几个方面开展工作,以进一步提高算法的执行 速度与并行可扩展性:

1)动态负载平衡策略。针对团簇特征的空间任意性,探讨曲面粒子和有效网格核间分布的不均衡问题,设计任务再划分与重分配策略,平衡计算和通信开销,提升不均衡分布特征下的并行绘制效率。

2)稀疏区域剔除。并行分配任务将粒子数 据包围盒划分为子区域时,剔除掉空白或稀疏粒 子区域,对于其中的孤立粒子,直接生成包络球 面,由此减小计算开销。

3)混合并行实现。本文算法目前是进程并 行,可考虑结点间进程、结点内线程的混合并行实 现机制,由此降低子区域的划分个数,从而减小 MPI聚合通信的开销以及子区域扩展部分的计 算量。

4)粒子数据下采样。针对超大规模粒子数据,可考虑进行下采样处理,基于采样后的代表性粒子来重构距离场,由此降低算法的处理 开销。

参考文献(References)

- ZHU Y N, BRIDSON R. Animating sand as a fluid [C]// Proceedings of the ACM SIGGRAPH 2005 Papers, 2005: 965-972.
- [2] SOLENTHALER B, SCHLÄFLI J, PAJAROLA R. A unified particle model for fluid-solid interactions [J]. Computer Animation and Virtual Worlds, 2007, 18(1): 69-82.
- [3] AKINCI G, IHMSEN M, AKINCI N, et al. Parallel surface reconstruction for particle-based fluids [J]. Computer Graphics Forum, 2012, 31(6): 1797 - 1809.
- [4] WANG H W, XIAO L, CAO Y, et al. Visibility-cullingbased geometric rendering of large-scale particle data [C]// Proceedings of the International Conference on Virtual Reality and Visualization (ICVRV), 2016.
- [5] WALD I, JOHNSON G P, AMSTUTZ J, et al. OSPRay: a CPU ray tracing framework for scientific visualization [J].
 IEEE Transactions on Visualization and Computer Graphics, 2017, 23(1): 931 - 940.
- [6] PFISTER H, ZWICKER M, VAN BAAR J, et al. Surfels: surface elements as rendering primitives [C]//Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, 2000.
- WANG Y F, SERENA F, WU S H, et al. Differentiable surface splatting for point-based geometry processing [J].
 ACM Transactions on Graphics, 2019, 38(6): 230.
- [8] FRAEDRICH R, AUER S, WESTERMANN R. Efficient high-quality volume rendering of SPH data[J]. IEEE Trans Vis Comput Graph, 2010, 16(6): 1533 – 1540.
- [9] HOCHSTETTER H, ORTHMANN J, KOLB A. Adaptive sampling for on-the-fly ray casting of particle-based fluids [C]// Proceedings of High Performance Graphics, 2016: 129 - 138.
- [10] BLINN J F. A generalization of algebraic surface drawing [J]. ACM Transactions on Graphics, 1982, 1(3): 235 – 256.
- [11] MÜLLER M, CHARYPAR D, GROSS M. Particle-based fluid simulation for interactive applications [C]//Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, 2003.
- [12] AKINCI G, AKINCI N, OSWALD E, et al. Adaptive surface reconstruction for SPH using 3-level uniform grids [C]// Proceedings of the 21st International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, 2013.
- [13] HUANG C, ZHU J, SUN H Q, et al. Efficient fluids simulation and rendering on GPU [C]//Proceedings of the 12th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry, 2013.
- [14] DU S C, KANAI T. GPU-based adaptive surface reconstruction for real-time SPH fluids [C]//Proceedings of Conference on Computer Graphics, Visualization and Computer Vision, 2014.
- [15] WU W, LI H P, SU T Y, et al. GPU-accelerated SPH fluids surface reconstruction using two-level spatial uniform

grids[J]. The Visual Computer, 2017, 33: 1429-1442.

- [16] YANG W C, GAO C Y. A completely parallel surface reconstruction method for particle-based fluids [J]. The Visual Computer, 2020, 36: 2313 – 2325.
- [17] YU J H, TURK G. Reconstructing surfaces of particle-based fluids using anisotropic kernels [J]. ACM Transactions on Graphics, 2013, 32(1): 5.
- [18] WANG X K, BAN X J, ZHANG Y L, et al. Anisotropic surface reconstruction for multiphase fluids [C]//Proceedings of the International Conference on Cyberworlds, 2017.
- [19] CHEN Q R, ZHANG S, ZHENG Y. Enhanced narrow band surface reconstruction with anisotropic kernel [J]. Computers & Graphics, 2022, 102: 280 – 288.
- [20] CHEN Q R, ZHANG S, ZHENG Y. Parallel realistic visualization of particle-based fluid [J]. Computer Animation and Virtual Worlds, 2021, 32: e2019.
- [21] SANDIM M, OE N, CEDRIM D, et al. Boundary particle resampling for surface reconstruction in liquid animation [J]. Computers & Graphics, 2019, 84: 55-65.
- [22] CHEN Q R, ZHANG S, ZHENG Y. Particle pre-processing for particle-based fluid surface reconstruction [C]// Proceedings of the 14th International Symposium on Visual

Information Communication and Interaction, 2021.

- [23] SANDIM M, PAIVA A, DE FIGUEIREDO L H. Simple and reliable boundary detection for meshfree particle methods using interval analysis[J]. Journal of Computational Physics, 2020, 420: 109702.
- [24] OLIVEIRA F, PAIVA A. Narrow-band screen-space fluid rendering[J]. Computer Graphics Forum, 2022, 41(6): 82-93.
- [25] LI H J, REN H X, WANG C, et al. A unified anisotropic particle-based ocean wave simulation framework for marine simulator visual system [J]. IEEE Access, 2020, 8: 126370-126384.
- [26] INCAHUANACO F, PAIVA A. Surface reconstruction method for particle-based fluids using discrete indicator functions[J]. Computers & Graphics, 2023, 114: 26-35.
- [27] LORENSEN W E, CLINE H E. Marching cubes: a high resolution 3D surface construction algorithm [J]. ACM SIGGRAPH Computer Graphics, 1987, 21(4): 163 – 169.
- [28] CHILDS H, BRUGGER E, BONNELL K, et al. A contract based system for large data visualization [C]//Proceedings of the IEEE Visualization, 2005.