

磨矿破碎过程粒度分布的分布式参数蒙特卡洛动力学 模拟及加速方法

卢绍文¹ 蔚润琴² 崔玉洁³

摘要 本文针对磨矿破碎过程, 提出一种分布式参数蒙特卡洛动力学方法的粒度分布预测模型和模拟算法. 该算法采用了分段思想, 将磨机沿着轴向分为若干个虚拟的子磨机; 根据破裂、前向和后向移动三类微观事件定义了倾向函数和系统状态矩阵, 并设计了分布式算法的调度策略. 此外, 针对蒙特卡洛动力学算法效率低的问题, 提出了基于 τ -leap 的磨矿过程分布式参数蒙特卡洛模拟加速算法. 为了解决分布式参数更新过程中状态不一致的问题, 创新性地提出了一种基于缓冲区的同步方法. 通过对仿真案例的分析表明, 本文提出的分布式参数蒙特卡洛动力学算法具有较高的精度, 提出的基于 τ -leap 的加速算法能够显著提高计算效率, 同时保持较好的精度.

关键词 磨矿破碎过程, 分布式参数模型, 粒度分布, 蒙特卡洛动力学模拟, 模拟加速

引用格式 卢绍文, 蔚润琴, 崔玉洁. 磨矿破碎过程粒度分布的分布式参数蒙特卡洛动力学模拟及加速方法. 自动化学报, 2019, 45(9): 1655–1665

DOI 10.16383/j.aas.c180020

A Distributed Parameter Kinetic Monte Carlo Simulation Algorithm of Grinding Process and Its Acceleration

LU Shao-Wen¹ YU Run-Qin² CUI Yu-Jie³

Abstract In this paper, we propose a prediction model for the particle size distribution of grinding process and a kinetic Monte Carlo simulation algorithm. The algorithm is based on the idea of dividing the mill into several virtual grinding machines along the axial direction. The preference function and the system state matrix are defined according to three kinds of microcosmic events, breakage, forward movement and back movement, and the scheduling strategy of distributed algorithm is designed. In addition, in view of the low efficiency of the kinetic Monte Carlo algorithm, a Monte Carlo simulation acceleration algorithm based on τ -leap is proposed. In order to solve the problem of inconsistent state when updating the distributed parameters, a new buffer based method is proposed. Case study based on simulation shows that the distributed parameter Monte Carlo dynamic algorithm proposed in this paper has high precision, and the proposed algorithm based on τ -leap can significantly improve the computational efficiency while maintaining good accuracy.

Key words Grinding process, distributed parameter model, particle size distribution, kinetic Monte Carlo simulation, simulation acceleration

Citation Lu Shao-Wen, Yu Run-Qin, Cui Yu-Jie. A distributed parameter kinetic Monte Carlo simulation algorithm of grinding process and its acceleration. *Acta Automatica Sinica*, 2019, 45(9): 1655–1665

选矿过程中, 颗粒破碎是在外力作用下使物料由大变小的过程, 使得原料能够满足后续的选别作

业对物料的粒度要求, 便于物料中 useful 矿物与脉石解离. 颗粒破碎研磨是高耗能工业过程, 运行优化的主要目标是在满足工艺要求的情况下降低能耗, 并使产品粒度稳定^[1–3]. 粒度分布是破裂过程的关键工艺指标, 它是不同粒度颗粒在总物料中占比. 粒度分布过大不能满足工艺要求, 但是如果物料磨得过细会导致能耗过高. 粒度分布可以通过激光粒度仪在线检测, 但是这类仪表成本高, 而且受到生产环境干扰影响, 需经常校准. 目前通常采用离线筛析, 其主要问题是滞后大, 不能满足在线运行优化的要求. 由于破碎过程的主要扰动来自于原料的粒度、硬度的变化, 利用破碎过程模拟技术来预测在当前的生产条件下最终的粒度分布, 可通过运行操作优化来

收稿日期 2018-01-10 录用日期 2018-07-02
Manuscript received January 10, 2018; accepted July 2, 2018
国家自然科学基金 (61833004, 61473071) 资助
Supported by National Natural Science Foundation of China (61833004, 61473071)

本文责任编辑 吴立刚
Recommended by Associate Editor WU Li-Gang
1. 东北大学流程工业综合自动化国家重点实验室 沈阳 110819 2. 东北大学信息科学与工程学院 沈阳 110819 3. 东北大学秦皇岛分校控制工程学院 秦皇岛 066004
1. State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819
2. College of Information Science and Engineering, Shenyang 110819 3. School of Control Engineering, Northeastern University at Qinhuangdao, Qinhuangdao 066004

补偿干扰^[4-6]。此外, 经过验证的模拟模型还可以用于检验工艺设计, 对于优化磨矿生产过程控制和设计都具有重要应用价值^[7-10]。

破碎过程粒度分布的模拟主要有三类方法。第一类是稳态模拟方法^[11], 该方法假设生产的边界条件不变, 将粒度分布的变化近似为线性过程, 主要用于初期的工艺设计, 不能用于在线指导操作。由于磨矿过程机理极为复杂, 难以建立精确的数学模型, 一般采用唯像学建模法^[11]。20 世纪 40 年代, 利用统计原理研究磨机内物料粒度分布, 提出了选择函数和破裂函数两个概念^[12]。其中选择函数描述颗粒发生破裂的概率, 破裂函数刻画颗粒破碎后得到的子颗粒的粒度分布。Austin 等经过实验给出了选择函数和破裂函数的经验公式。文献 [13] 提出矩阵动力学模型与总体平衡模型, 为磨矿的研究开辟了一个新的领域。文献 [14] 提出磨矿的动态矩阵理想模型, 进一步推动了磨矿数学模型的发展。基于能级关系的 Bond 模型是比较经典的球磨机模型, 但这种模型得不到各粒级的质量分数。上述这些模型主要是稳态模型, 无法刻画颗粒破碎过程中粒度分布的演变。

第二类方法为动态模拟方法。磨矿过程的动态模型需要建立粒度分布随时间变化的过程模型, 输入物料的性质、操作条件, 模型输出每一时刻粒度分布的模拟值作为输出。该方法主要采用总量平衡原理建立粒度分布的主方程模型^[15]。通过磨矿粒度分布的群体平衡动态模型, 记录磨机中不同位置各个粒径的颗粒每时每刻的状态, 从而描述出磨矿过程中物料在时间和空间上的粒度分布。但该模型一般是积分微分方程形式, 多数情况下难以获得解析解^[16]。

第三类是动力学模拟方法, 它是一种随机模拟方法^[17-18], 它用随机过程来刻画颗粒的破碎过程, 能够得到颗粒破碎过程的演化过程。蒙特卡洛动力学模拟方法 (以下简称 MC 方法) 是常用的随机模拟方法, 它通过模拟每一次破碎事件的发生, 记录研磨过程中粒度分布的动态变化情况, 较真实模拟了物理过程。文献 [17-18] 提出了基于 MC 方法的颗粒破碎过程模拟算法。但是在颗粒破碎过程中, 大粒径颗粒会破碎为多个小粒级的颗粒, 因此随着仿真时间的推进, 需要模拟的颗粒总数以指数速度增加, 导致计算量快速增大, 效率降低。针对这一问题, Khalili 等和 Smith 等提出了一种定数量 MC (Constant-number Monte Carlo, CNMC) 方法^[19-20], 该算法在处理颗粒的凝并和破碎事件时, 通过随机从系统中增加或移除颗粒, 目的是使系统总的颗粒样本数保持不变, 从而提高了原有 MC 方法的效率。赵海波等提出了一种多重 MC (Multi

Monte Carlo, MMC) 方法, 通过引入加权的“虚拟颗粒”的概念, 将实际颗粒由一颗或者几颗属性相近的虚拟颗粒代表并赋予一个数目转换权值, 该值就是该虚拟颗粒所代表的当地实际颗粒的个数, 模拟时通过调整数目转换权值来保证在发生凝并和破裂事件之后虚拟颗粒总数目仍然保持不变。卢绍文在 CNMC 基础上^[21] 提出一种带精度反馈和速度控制的动力学模拟算法 (ED-KMC-AC), 能够在满足一个给定的精度阈值约束下, 最大地提高模拟速度。这些方法能够加快 MC 模拟的速度, 其本质上是通过人为减少系统内颗粒数目实现的, 这将降低模拟的准确程度。Gillepsie 在文献 [22] 中提出了 τ -leap 近似加速算法, 他假设磨矿过程中多个颗粒破碎信息可以合并, 不需要每一次事件都更新系统, 通过一个调节因子来控制系统更新的粒度, 从而从总体上调节模拟的速度和精度。

上述模型都属于集总参数模型, 即假设磨机是一个质量分布均匀的质点, 物料在磨机内处处均匀。但实际工业生产中, 工业磨机的尺寸日趋大型化, 物料在磨机内的驻留时间更长, 物料性质的空间分布难以忽略, 集总参数模拟方法已经不能准确刻画物料在磨机腔体内的驻留和变化^[23]。针对这个问题, Kis 等^[23] 提出了分布参数的磨矿过程模型。该模型不仅研究磨机径向方向的破裂, 同时也考虑了磨机轴向方向物料的移动, 考虑了对流前移和扩散后移两种颗粒消失事件。该方法将磨机分为若干段虚拟的子磨机, 从而得到每一段子磨机在连续磨矿过程中物料的变化情况, 最后合并得到产品的粒度分布。但是基于该方法的模型需要采用数值近似方法进行求解, 由于模型方程非常复杂, 不能保证解的存在。此外, 该方法只能解出粒度分布随时间变化的近似解, 无法描述分布参数下磨矿模拟过程中颗粒的破裂和移动的随机特性。针对这些问题, 本文提出一种基于分布式参数破碎过程模拟的粒度分布终点预报方法。我们采用分布参数模拟方法, 将磨机在轴向分段为若干串级连接的虚拟的子磨机, 由于每个子磨机的轴长足够小, 可以假设认为每段子磨机内物料为充分混合状态, 从而将连续磨矿过程看作多个串级连接的批次子磨过程, 但这种方法面临计算代价过大的问题。为了提高模拟效率, 在批次磨矿过程的蒙特卡洛动力学模拟方法基础上, 给出一种模拟加速方法。

1 相关算法及存在的问题

磨机内物料发生破碎的过程很难精确建模, 只能采用概率的方法, 即将颗粒的破碎看成是一个随机事件, 通过人为控制实验条件下进行大量实验获得的实验样本, 采用统计方法得到不同条件下破碎事

件发生的概率. 将颗粒的破碎过程看作是一个随机过程, 通过实验我们可以获得两方面重要特性, 一个是破碎的速率, 一个是颗粒破碎后获得的子颗粒的分布. 前者假设颗粒破碎的发生符合泊松分布, 破碎事件发生的间隔时间服从指数分布, 通过选择函数的形式给出. 后者通过物理实验获得, 它主要与物料的硬度和材质相关, 通过破裂函数的形式给出. 但是颗粒破碎速率与当前的粒度分布、浓度等多种因素有关. 这些因素在生产过程中是不断变化的, 如果再考虑模型的分布参数, 模型的形式异常复杂, 难以获得解析解.

1.1 集总参数的颗粒破碎蒙特卡洛模拟算法

蒙特卡洛 (MC) 模拟方法是一种随机数值算法, 主要利用随机抽样原理实现对任意积分过程的数值模拟. 目前采用蒙特卡洛方法模拟颗粒破碎过程, 主要是利用蒙特卡洛抽样方法来模拟时变的颗粒破碎概率分布以及破裂函数. 蒙特卡洛方法分事件驱动和时间驱动两种, 它们的数值模拟能力是等价的^[24]. 以下给出时间驱动蒙特卡洛模拟算法.

算法 1. 时间驱动蒙特卡洛颗粒破碎算法

- 1) REPEAT
- 2) 反函数法求下一破裂事件步长
- 3) 舍选法选取被破裂颗粒
- 4) 反函数确定破裂子颗粒的粒度
- 5) 系统状态更新
- 6) UNTIL 模拟时间达到最大

假设磨机内破碎事件的发生满足马尔科夫特性, 粒径为 d 的颗粒发生破碎事件的速率由选择函数 $S(d)$ 定义, 事件间隔时间服从指数分布. 从当前时刻 t 到下一破裂事件发生的时间间隔 Δt 可通过反函数法得到:

$$\Delta t = \frac{\ln(1 - R_1)^{-1}}{S_b} \quad (1)$$

其中, R_1 为 $(0, 1)$ 区间随机数, S_b 为所有计算区域内颗粒发生破碎事件的总速率. 采用蒙特卡洛舍选法^[17] 确定被破裂颗粒所在粒度, 生成 $(0, 1)$ 区间随机数 R_2 , 在 1 到 N 中随机选择一个粒度 j , 若下式成立, 则 j 为发生被破裂颗粒所在粒度:

$$R_2 < \frac{S_j}{S_{\max}}, \quad S_{\max} = \max(S_i), \quad i = 1, \dots, N \quad (2)$$

其中, S_i 为粒度 i 的选择函数. 由于破裂函数一般不随生产条件变化, 仍采用反函数法确定破裂产生的子颗粒的粒径, 生成 $(0, 1)$ 区间随机数 R_3 , 在粒度 j 到 N 随机选择粒度 k , 如果下式成立, 则 k 为

破裂产生的子颗粒的粒度.

$$R_3 \leq B(k, j) \quad (3)$$

其中, $B(k, j)$ 为破裂函数的累积量. 最后, 更新当前各个粒度颗粒数量的系统状态向量 $X(t)$ 以及各个粒度颗粒的破碎速率 S 和总破碎速率 S_b .

1.2 分布式参数的颗粒破碎过程离散差分数值算法

算法 1 介绍的蒙特卡洛算法属于集总参数算法, 主要用于批次磨的模拟. 对于连续磨过程, 物料从磨机入口到出口的传递过程中物料颗粒在轴向上不一定分布均匀, 所以集总参数模型不能精确模拟连续磨过程. 针对这一问题, Kis 等^[23] 提出一种考虑物料在磨机中轴向分布的基于离散差分方法的连续磨粒度分布动态模型. 如图 1 所示, 首先将磨机在轴向上等分为 J 段子磨机, 分别为 $y_1, y_2, \dots, y_j, \dots, y_J$; 径向上分为 N 个粒度, 分别为 $x_1, x_2, \dots, x_i, \dots, x_N$. 假设每段子磨机内物料颗粒分布均匀, 对子磨机利用批次磨集总参数模拟方法对每个子磨机同时进行模拟. 而与批次磨略有不同的地方是每段子磨机各个粒度的颗粒数目的变化不仅来自于径向上颗粒之间的破裂过程, 同时也是相邻子磨机之间颗粒相互移动的结果.

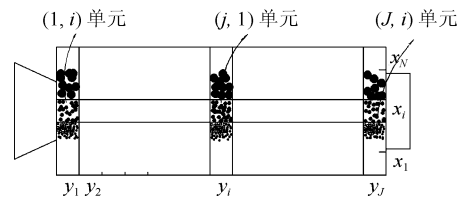


图 1 离散径向分布参数磨机模型示意图

Fig. 1 Illustration of the distributed parameter grinding mill model

以任意中间段第 j 段子磨机的 i 粒度为例分析磨机中间段的物料变化, 称矩形区域 $[y_{j-1}, y_j][x_{j-1}, x_j]$ 为 (j, i) 单元, $u(y_j, x_i, t_n)$ 代表该单元的质量, 另外定义 $m(y, x, t_n)$ 为物料在 t_n 时刻的密度, 则

$$u(y_j, x_i, t_n) = \int_{y_{j-1}}^{y_j} \int_{x_{i-1}}^{x_i} m(y, x, t_n) dx dy \quad (4)$$

该单元物料的质量包括四小部分: 1) 该单元经过破裂及先后移动后剩余的质量; 2) 上段子磨机移动对流; 3) 下段子磨机移动逆流; 4) 径向上其他粒径比较大的粒度破裂为粒度颗粒. 则该单元在下一

时刻粒度分布可以由以下离散方程表示:

$$u(y_j, x_i, t_{n+1}) = (1 - dt - dtS(\bar{x}_i))u(y_j, x_i, t_n) + dtV_F u(y_{j-1}, x_i, t_n) + dtV_B u(y_{j+1}, x_i, t_n) + \sum_{k=1}^i dtP_{i,k} u(y_j, x_k, t_n) \quad (5)$$

其中, dt 表示模拟步进时间, V_F 表示物料单位时间向前移动比例, V_B 表示物料单位时间向后移动比例, $P_{k,i}$ 表示单位时间 i 粒级的颗粒破裂到 k 粒级的比例.

以 $(1, i)$ 单元为例分析入料口物料的质量变化, 与中间段相比该单元物料不存在来自上一段子磨机的移动对流, 但有给料注入物料. 因此:

$$u(y_1, x_i, t_{n+1}) = (1 - dt - dtS(\bar{x}_i))u(y_1, x_i, t_n) + dtV_B u(y_{j+1}, x_i, t_n) + \sum_{k=1}^i dtP_{i,k} u(y_j, x_k, t_n) + u(\bar{x}_i)m(\bar{x}_i) \quad (6)$$

最后, 以 (J, i) 单元为例分析出口处物料的质量变化. 与中间段相比, 出口处不存在下段子磨机逆流移动而来的物料, 因此:

$$u(y_J, x_i, t_{n+1}) = (1 - dt - dtS(\bar{x}_i))u(y_J, x_i, t_n) + dtV_B u(y_{j+1}, x_i, t_n) + \sum_{k=1}^i dtP_{i,k} u(y_j, x_k, t_n) \quad (7)$$

如上, 通过求解方程得到每一段子磨机中物料颗粒的动态变化过程, 从而模拟整个磨机物料的粒度分布的变化情况.

1.3 存在的问题

颗粒破碎蒙特卡洛模拟算法的主要优点是它能够准确地模拟物料随时间变化的过程, 但也存在两方面缺点: 1) 基于集总参数的蒙特卡洛动力学算法无法反映物料性质在磨机内部的差异及动态变化, 因此不适于大型磨机以及连续作业过程的模拟; 2) 这类算法通常需要较大的计算资源, 而且模拟的速度随着计算的进行不断衰减, 原因见文献 [18]. Kis 等^[23] 提出磨机轴向的离散化近似来解决这一问题. 该算法采用时间差分得到差分方程, 然后不断计算步进时间内物料质量的变化, 从而预测模拟到达研磨设定时间的模拟结果. Kis 算法的主要问题在于选择的差分步进时间必须满足一定的约束条件, 而该约束条件是随着模拟不断变化的, 确定的步进时间可能导致约束条件在模拟过程中被破坏.

步进时间的大小直接影响着物料质量的变化程度, 因此步进时间的大小对于模拟结果具有较大的影响. 以式 (5) 为例: 等式右边第一项表示该粒级内剩余物料质量, 即该粒级内的颗粒发生移动或者破裂事件后剩余的物料质量. 由于剩余物料质量不能为负值, 因此应有:

$$1 - dtV_F - dtV_B - dtS(\bar{x}_i) \geq 0 \quad (8)$$

令

$$\tau'(i) = \frac{1}{uh + \frac{2D}{h^2} + S(\bar{x}_i)}, \quad i = 1, \dots, N \quad (9)$$

进而可得到时间间隔的约束条件:

$$dt \leq \min_{i \in [1, \dots, N]} \tau'(i) \quad (10)$$

该算法时间步长的选择对于数值结果的精度影响至关重要, 当分别取不同的步进时间会得到不同的累积粒度分布曲线, 并且差别显著. 在差分方法中, 步进时间的选择需要事先人为决定, 导致其并不一定能够满足约束条件 (10), 这时模拟计算的结果是没有物理意义的. 另一方面, 离散差分方法虽然能够得到连续磨过程内物料的质量变化, 并求得产品的粒度分布, 但该方法结果属于固定的数值解, 模拟结果与试验次数无关, 因此它只是反映了连续磨过程的终点期望结果. 而在连续磨过程中, 颗粒的破裂与移动具有随机性, 固定意义上的数值解无法描述连续磨矿的随机过程, 以及各个粒级内颗粒的微观变化过程.

2 分布式参数磨矿模拟算法

2.1 分布式参数蒙特卡洛动力学模拟算法

离散差分的方法虽然通过求解方程能够得到整个磨机物料的粒度分布情况, 但该方法不能准确描述颗粒的动态变化过程. 而且磨机体积趋向于越来越大, 里面的物料颗粒数目比较庞大, 不同时间点各个位置各个粒级颗粒的破碎都是随机的, 所以本文在离散差分方法的基础上提出一种基于随机算法的分布式参数磨矿过程模拟方法.

本算法的思路借鉴离散差分算法, 将磨机分为多个串级连接的子磨机, 如图 1 所示. 轴向上对磨机按编号从 1 到 j 段子磨机. 对于每段子磨机, 径向上划分为 N 个粒级, 每个粒级用代表粒径 d_i 来表示. 并假设子磨机内物料均匀混合, 即子磨机内的粒度分布处处相同; 颗粒在子磨机之间移动是在破裂之后发生的. 对于子磨机内的颗粒破裂过程, 我们假设轴向的粒度差异可以忽略, 因此仍采用算法 1 给出的时间驱动蒙特卡洛模拟方法. 根据分段思想, 相

邻的子磨机间有物料的相互传递. 因此, 子磨机内颗粒数目的增加是由大颗粒破裂或者子磨机间颗粒的流入而导致的. 从微观角度来观察, 磨机内颗粒可能发生三种事件: 向前移动事件, 是由对流和物料本身的循环流动引起; 向后移动事件, 是由扩散作用引起; 破裂事件, 是由磨机研磨作用而导致颗粒破裂至比其粒度小的粒度内. 将上述三种事件统称为颗粒消失事件.

首先, 针对上述离散模型, 定义新的系统状态矩阵 $X_{N \times J}$, 其中 $x(i, j)$ 表示第 j 个子磨机内第 i 个粒级的颗粒数量. 由于分布式参数模型中倾向函数的计算需要靠考虑颗粒的对流前移、扩散后移及破裂三种颗粒消失事件, 新定义的倾向函数反映这三种事件发生的总速率, 定义如下:

$$a = \sum_{i=1}^n x(i, j)(V_F + S(i) + V_B) \quad (11)$$

对此又分为三种情况: 给料段 ($j = 1$) 存在对流前移和破裂 2 类事件; 中间段 ($j \in \{2, \dots, J - 1\}$) 存在前移、后移和破裂 3 类事件; 排料段 ($j = J$) 存在对流后移和破裂 2 类事件.

其次, 根据倾向函数计算步进时间. 由于磨机被等分为 J 子磨机, 不同子磨机分别采用 MC 方法进行模拟计算, 由于步进时间的选取只与子磨机局部状态相关, 因此导致全局时间不一致. 针对这一问题, 我们每次计算得到 J 个不同的步进时间, 并取其中的最小值作为全局系统的推进步长.

$$dt = \min_{i \in \{1, \dots, J\}} \{a^{-1}(i)\} \quad (12)$$

其中对于发生消失事件的子磨机, 采用算法 1 计算当前子磨机的粒度分布. 由于增加了前向流动和后向流动两种新的事件, 因此在用舍选法选择消失颗粒时, 需要根据式 (11) 定义的倾向函数来计算.

综上, 算法的主要步骤如下:

算法 2. 颗粒破碎的分布式参数蒙特卡洛动力学模拟算法

- 1) REPEAT
- 2) 利用式 (11) 计算倾向函数
- 3) 确定下一颗粒消失事件发生的子磨机并根据式 (12) 确定全局时间步长
- 4) 对发生消失事件的子磨机, 采用算法 1 计算粒度分布
- 5) 系统状态更新
- 6) UNTIL 模拟时间达到最大

2.2 基于 τ -leap 的磨矿过程分布式参数蒙特卡洛模拟加速算法

算法 2 实现的连续破碎过程模拟能够精确刻画

每一个子磨机的动态变化过程, 进而得到整个磨机中不同时间不同位置各个时间点的粒度分布. 但由于该方法需要记录每一次事件的发生, 计算量极大, 只能应用于小体系的模拟. 针对该问题, 本文在算法 2 基础上提出一种基于 τ -leap 的模拟加速方法. τ -leap 方法主要广泛应用于化学反应过程的快速模拟. 这里将 τ -leap 算法引入颗粒破碎过程, 通过将模拟时间划分为若干时间片作为步进时间, 同时假设在每个步进时间内有多个反应事件同时发生. 即不必每次事件都更新系统状态, 而是根据步进时间片内多个事件的累积效应对系统状态进行更新, 从而提高模拟速度.

在分布式参数模拟的框架下采用 τ -leap 方法, 就必须解决计算中系统状态不一致的问题. 在算法 2 中, 子磨机系统内的颗粒破碎和移动均被视为该颗粒的消失事件. 由于 τ -leap 加速方法并不是针对每个事件更新系统状态, 将其应用于分布式参数模型时, 不同子磨机的状态和时间推进速度并不一致, 这就带来子系统的状态同步和时间的同步问题. 此外, MC 模拟过程中需要假设当前子系统的状态是独立的, 但是在时间段 $(t, t + \tau)$ 内各类事件的累积效应还需要考虑子磨机之间存在的物料流动, 所以各个磨机之间的状态又是相关的. 这就导致 τ -leap 算法的思想并不能直接应用在分段离散模型中.

针对上述问题, 本文提出一种基于缓冲区的解决方法. 考虑到移动事件和破裂事件直接的独立性, 我们对每个子磨机系统开辟两个状态同步缓冲区: 前向移动缓冲区和后向移动缓冲区, 如图 2 所示. 在计算过程中, 将前向移动和后向移动的颗粒分别暂存在缓冲区, 等到达下一个同步时间点 $t + \tau$ 后, 再将缓冲区的颗粒与相邻子磨机进行交换. 本质上, 这相当于将前/后向的移动事件和破裂事件分别进行处理. 相应地, 在进行破裂事件的处理中, 将子磨机和其附属的缓冲区视为一个独立的批次磨过程, 并更新状态转移矩阵和倾向函数.

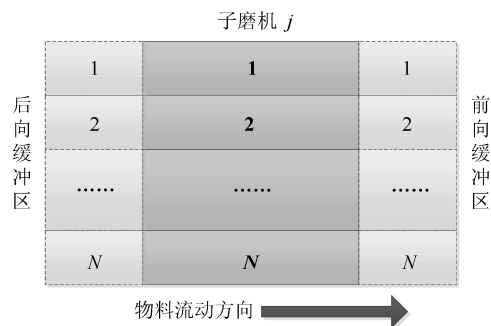


图 2 分段子磨机缓冲区示意图

Fig. 2 The buffer zones for sub-grinding mill

加速算法对每个子磨机的模拟计算采用基于

τ -leap 的批次 MC 模拟. 子磨机 j 的步进时间由以下公式计算:

$$\tau_j = \min_{m \in [1, \dots, M]} \left\{ \epsilon a_0(x) \left| \sum_{i=1}^N \xi_i(x) \frac{\partial a(x)}{\partial x} \right|^{-1} \right\} \quad (13)$$

其中, M 为消失事件的种类, a_0 为当前倾向函数之和, x 为系统状态, ϵ 为 τ -leap 算法的速度调节参数. 对于时间同步的问题, 由于 τ -leap 算法的主要作用是加快模拟速度, 因此步进时间的同步采用与算法 2 相反的策略, 即选取各个子磨机中的最大步长作为全局系统的步进同步时间点. 这样做的目的是尽可能发挥加速的性能.

$$dt = \max_{i \in [1, \dots, J]} \{\tau_i\} \quad (14)$$

加速算法的主要步骤如下所示:

算法 3. 颗粒破碎的分布式参数蒙特卡洛动力学 τ -leap 算法

1) **REPEAT**

2) 利用式 (11) 计算倾向函数

3) 确定下一颗粒消失事件发生的子磨机并根据式 (14) 确定全局时间步长

4) 对发生消失事件的子磨机, 采用基于 τ -leap MC 算法计算粒度分布, 其中 τ 值由式 (13) 计算

5) 系统状态更新

6) **UNTIL** 模拟时间达到最大

3 算法性能的比较分析

3.1 速度和精度的性能指标

动力学模拟算法的性能主要从精度和速度两个方面来衡量. 本文研究主要采用文献 [21] 所提出的两个量化指标用于以下的分析和讨论. 首先定义稳态模拟速度 (Transient simulation speed, TSS) 来度量磨矿过程的模拟速度. TSS 越大说明模拟速度越快. 表达式为:

$$TSS = \frac{dt_i}{t_i} \quad (15)$$

其中, dt_i 表示第 i 次与第 $i-1$ 次破裂事件之间模拟时间间隔, t_i 表示实际的计算时间.

其次, 定义平均粒径加权变异系数 (Weighted squared coefficient of variation, WSCV) 来度量磨矿过程的模拟精度, WSCV 越小说明模拟精度越好.

表达式为:

$$WSCV(x, t) = \sum_{i=1}^N M \frac{m_i^2(x, t)}{2} \times \left(x_i - S(i)x(i) + \sum_{j=1}^{i-1} b_{ij}S(j)x(j) \frac{v(j)}{v(i)} \right) \quad (16)$$

其中, $v(i)$ 代表第 i 粒级矿物颗粒的体积, $v(j)$ 代表第 j 粒级矿物颗粒的体积.

3.2 模拟计算实验

本文分别采用离散差分方法、MC 方法以及 τ -leap 加速方法对磨矿过程进行模拟实验. 模型的初始参数设置如下: 磨机的长度 $L = 4.4$ m, 磨机分段数 $J = 10$, 研磨时间 $T = 2$ min, 扩散协同系数 $D = 0.005$, 对流速度 $u = 0.065$. 为了便于比较, 模型中的破裂函数、选择函数参数以及给料粒度取值与文献 [21] 相同, 这里不再赘述.

3.2.1 算法精度比较

1) 平均粒径

图 3 所示为磨机出口物料各粒级代表粒径的平均值随时间的变化曲线 (其中 $\epsilon = 0.001$). 从图中可见磨机中物料的平均粒径随时间逐渐变小, 这反映了磨机中较大粒径的颗粒不断破裂为较小粒径颗粒的物理过程. 从图中可以看到 MC 模拟方法和基于 τ -leap 的模拟加速算法计算得到的平均粒度高度吻合, 并不存在系统性的差别. 统计两种算法的相对误差, 最大误差为 4.3×10^{-2} , 最小误差为 2.13×10^{-6} ,

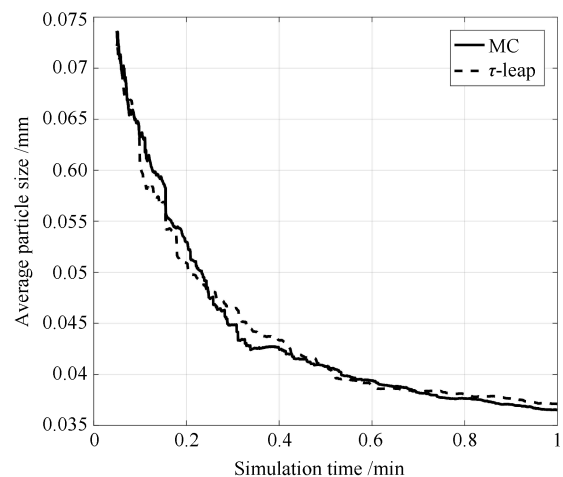


图 3 分布参数 MC 算法 (图中简称 MC) 和 τ -leap 加速算法计算得到的平均粒径随时间变化趋势

Fig. 3 The averaged particle size over simulation time for distributed parameter MC algorithm and the τ -leap accelerated algorithm

平均误差为 8.19×10^{-4} 。值得指出的是, 在时刻约 0.05 min 之前, 可以观察到模型计算得到的排料为零。这是因为分布式参数模型考虑了磨机轴向物料分布, 物流从给料口传递到排料口之前磨机没有物料排出。这是与集总参数模型最显著的不同。

理论和实验结果表明基于 MC 动力学的模拟算法具有较高的精度^[25]。我们以分布参数 MC 算法为对比的基准, 对比 τ -leap 加速算法和离散差分数值算法的计算结果, 见图 4。其中每个子图中绘制了该子磨机模型的输入-输出的累积粒度分布曲线。进而, 将计算结果分为两组比较: 第 1 组为分布参数

MC 算法和离散差分算法的比较, 第 2 组为分布参数 MC 算法和 τ -leap 加速算法的比较, 结果见表 1。从图 4 和表 1 中的统计数据可看出, 两组算法的相对误差稳定在 1% 以下, 且未表现出明显的误差变化趋势。从第 1 组结果可以看出, 分布式参数 MC 算法和离散差分数值算法的最终计算结果的差距较小。第 2 组结果表明, 采用 τ -leap 算法对 MC 模拟加速后, 仍然确保了较高的最终计算精度。

2) 平均绝对误差

τ -leap 方法确保准确性的充要条件是倾向函数在一段时间内不发生明显变化。倾向函数的变化可

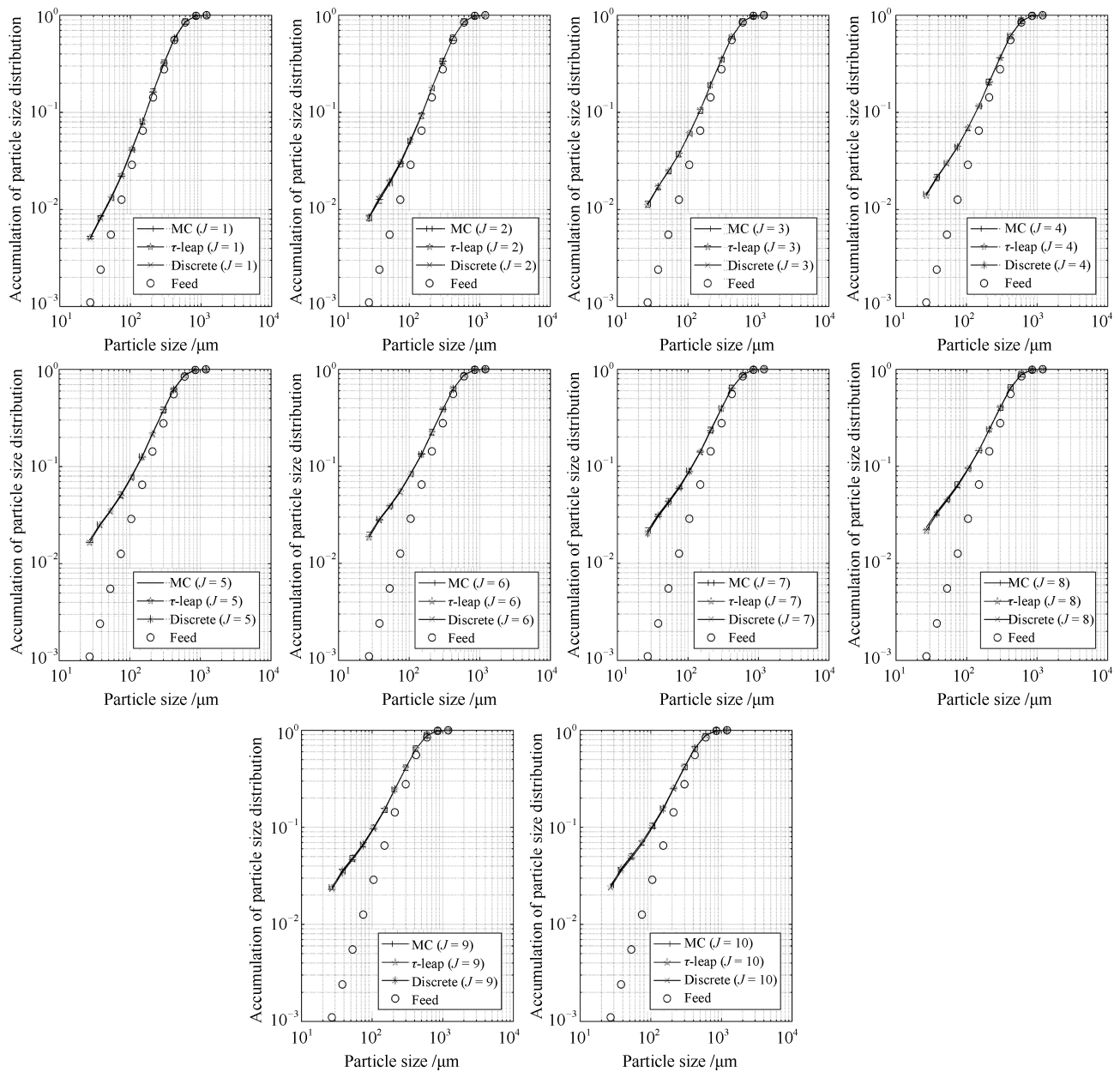


图 4 各子磨机系统的输入和输出物料的累积粒度分布曲线对比

Fig. 4 The cumulative particle size distribution of the input and output of each sub-mill

表 1 离散差分数值算法和 τ -leap 方法相对于分布参数 MC 算法的相对误差比较

Table 1 The relative errors of the discrete numerical algorithm and the τ -leap algorithm with the MC algorithm

子磨机编号		1	2	3	4	5	6	7	8	9	10
第 1 组 ($\times 10^{-3}$)	最大误差	8.6	3.8	4.9	7.2	8.6	7.2	5.8	6.7	5.4	5.7
	平均误差	2.2	0.7	1.0	2.1	2.9	1.2	2.2	2.7	1.9	2.7
第 2 组 ($\times 10^{-3}$)	最大误差	4.4	3.3	7.3	7.2	1.1	7.5	6.4	1.1	9.5	6.7
	平均误差	1.0	1.2	1.7	1.8	3.1	2.2	2.6	2.9	1.4	1.8

通过参数 ϵ 反映. 增大 ϵ 相当于增大 τ -leap 方法充要条件的判断阈值, 进而可以更大幅度地加快模拟速度, 但同时也会带来更大的误差; 反之, 加速幅度降低, 但保持较高的精度.

我们对 ϵ 分别取 1×10^{-4} 、 1×10^{-3} 、 1×10^{-2} , 将 τ -leap 加速算法得到结果与分布式 MC 方法的结果进行对比. 考虑到颗粒粒度是多个粒级的分布来刻画, 因此需要计算每个粒级的粒度分布的绝对差. 为了便于观察, 我们定义平均绝对误差 M_p 如下式所示:

$$M_p(t) = \frac{\|p_{mc}(t) - p_{\tau}(t)\|_1}{N} \quad (17)$$

其中, $p_{mc}(t)$ 为 t 时刻下分布式 MC 方法的粒度分布输出, $p_{\tau}(t)$ 是基于 τ -leap 的加速方法得到的 t 时刻的输出, $\|\cdot\|_1$ 为 1 范数.

表 2 列出了 ϵ 分别取 1×10^{-4} 、 1×10^{-3} 、 1×10^{-2} , 不同时段内求得的绝对误差 M_p 的平均值. 表中数据的折线图如图 5 所示. 可以看出, ϵ 值越小, M_p 值越小, 即 τ -leap 方法与 MC 方法的差异程度越小. 说明 ϵ 取值越小, τ -leap 方法与 MC 方法越接近, 相应的 τ -leap 方法越准确. 这与预期吻合, 即模拟结果的误差 M_p 值与 ϵ 的取值呈正比关系. 此外, 还可以观察到计算误差随着仿真的推进越来越小的明显趋势. 这一点的原因将在第 3.2.2 节分析.

3) 加权变异系数

式 (16) 定义的加权变异系数是对过程随机性的衡量指标, 它能够反映模拟算法对于微观动力学

过程的模拟精度. 表 3 所示为出口段子磨机输出在采用 τ -leap 的加速方法后, 其加权变异系数 WSCV 与分布式 MC 方法之误差在不同时间段内的值, 其中列出了 τ -leap 方法下不同 ϵ 取值得到的结果. 可以看出两种方法的 WSCV 值均会随着模拟过程减小, 这主要是由于破裂过程导致系统内整体颗粒个数逐渐增加, 即式 (16) 右边括号内减去的 $S(i)x(i)y$ 一项. 但是, WSCV 的减小速度会越来越慢, 这是因为系统内的颗粒的数量增加到一定数目以后, 颗粒粒度的分布范围收窄, 从而部分抵消了增加颗粒的

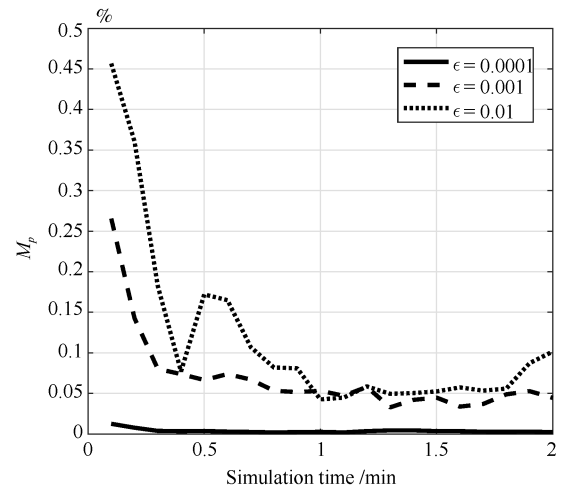


图 5 不同 ϵ 值所对应的 M_p 值随时间变化图

Fig. 5 The values of M_p during the simulation under different ϵ

表 2 基于 τ -leap 的加速算法在不同 ϵ 取值下与分布式参数 MC 算法的 M_p 的相对误差 (%)

Table 2 The relative difference of M_p (%) of the τ -leap acceleration algorithm comparing with the MC algorithm

t (min)	0~0.1	0.1~0.2	0.2~0.3	0.3~0.4	0.4~0.5	0.5~0.6	0.6~0.7	0.7~0.8	0.8~0.9	0.9~1.0
M_p ($\epsilon = 1 \times 10^4$)	0.0125	0.0077	0.0038	0.0034	0.0037	0.0029	0.0025	0.0019	0.0023	0.0026
M_p ($\epsilon = 1 \times 10^3$)	0.2654	0.1426	0.0802	0.0738	0.0665	0.0738	0.0671	0.0533	0.0516	0.0532
M_p ($\epsilon = 1 \times 10^2$)	0.4565	0.3605	0.1825	0.0770	0.1720	0.1649	0.1071	0.0819	0.0809	0.0426
t (min)	1.0~1.1	1.1~1.2	1.2~1.3	1.3~1.4	1.4~1.5	1.5~1.6	1.6~1.7	1.7~1.8	1.8~1.9	1.9~2.0
M_p ($\epsilon = 1 \times 10^4$)	0.0019	0.0035	0.0044	0.0044	0.0035	0.0035	0.0027	0.0026	0.0028	0.0023
M_p ($\epsilon = 1 \times 10^3$)	0.0471	0.0580	0.0328	0.0420	0.0448	0.0337	0.0369	0.0488	0.0528	0.0445
M_p ($\epsilon = 1 \times 10^2$)	0.0447	0.0586	0.0495	0.0504	0.0524	0.0574	0.0534	0.0559	0.0872	0.1015

表 3 基于 τ -leap 的加速算法在不同 ϵ 取值下与分布式参数 MC 算法的 M_p 的相对误差 (%)

Table 3 The relative difference of M_p (%) of the τ -leap acceleration algorithm comparing with the MC algorithm

t (min)	0~0.1	0.1~0.2	0.2~0.3	0.3~0.4	0.4~0.5	0.5~0.6	0.6~0.7	0.7~0.8	0.8~0.9	0.9~1.0
WSCV ($\epsilon = 1 \times 10^4$)	27.6599	16.4219	5.5694	6.4539	6.3902	2.0711	1.6268	0.8461	1.4014	2.8565
WSCV ($\epsilon = 1 \times 10^3$)	29.3654	19.5643	12.4057	14.1440	8.9499	9.0685	7.7018	5.3212	4.5702	4.0914
WSCV ($\epsilon = 1 \times 10^2$)	30.5564	20.5641	15.235	16.2165	8.9873	8.1839	8.4892	6.1973	5.0565	5.9391
t (min)	1.0~1.1	1.1~1.2	1.2~1.3	1.3~1.4	1.4~1.5	1.5~1.6	1.6~1.7	1.7~1.8	1.8~1.9	1.9~2.0
WSCV ($\epsilon = 1 \times 10^4$)	2.2858	2.6192	3.0302	3.5939	3.3510	5.3169	2.5186	2.1569	1.7443	1.1591
WSCV ($\epsilon = 1 \times 10^3$)	6.6934	6.8302	6.3439	6.9364	1.6810	2.7825	3.3907	4.0133	5.6812	3.2316
WSCV ($\epsilon = 1 \times 10^2$)	7.2165	7.6127	7.3752	7.9085	2.3725	2.6708	4.3917	5.4931	6.3914	3.9081

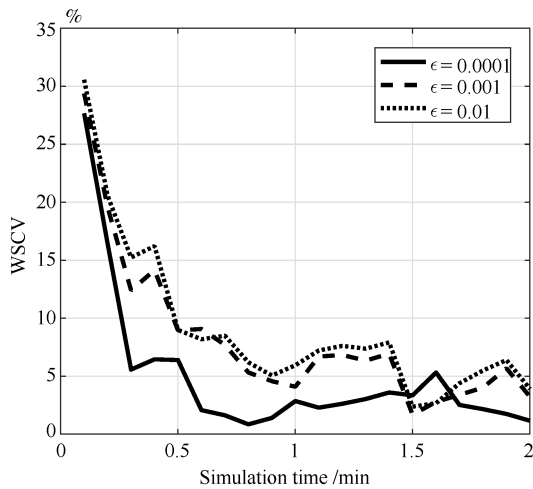


图 6 基于 τ -leap 的加速算法在不同 ϵ 取值下的 WSCV 与分布式参数 MC 算法相对误差

Fig. 6 The relative difference of WSCV of the τ -leap acceleration algorithm comparing with the MC algorithm

数目对于 WSCV 减小的贡献, 即式 (16) 右边括号内的求和项. 具体分析过程参见^[21].

此外, 由于本文所描述的对象为“冷车”启动, 初始时磨机中并没有待研磨物料, 因此在模拟初期颗粒数目比较少, 使得倾向函数较小, 当颗粒发生消失事件时, 倾向函数产生很大的波动, 破坏了 τ -leap 条件. 所以, 可以从图中看到, τ -leap 方法与 MC 方法相比的 WSCV 结果在模拟初始时存在较大的差距, 但随着模拟时间的进行, 由于系统内的颗粒总数的增多, 两种方法的差距越来越小. 这说明: 1) 基于 τ -leap 的加速模拟算法的结果会随着模拟时间不断逼近相应的 MC 模拟结果; 2) 在相对较短的模拟周期内, 二者的相对误差较大. 以上两点趋势可以从图 6 中更明显地看出.

3.2.2 τ -leap 算法的加速性能

本节主要考察基于 τ -leap 算法的加速性能. 主要根据式 (15) 定义的稳态模拟速度指标 TSS 来分析. 由于磨机中颗粒的移动和破裂的随机性导致 TSS 曲线波动较大. 所以对每个 0.1 分钟模拟时间

段内的瞬时 TSS 求平均值, 结果见表 4. 可以看到采用 τ -leap 加速方法能够明显提高模拟计算的速度, 如图 7 所示. 其中, 调节因子 ϵ 可以用于调节加速的性能. 例如, 当 ϵ 取 0.01 时, 基于 TSS 计算的加速比最大可达 227 倍, 最小也可以取得 30 倍的加速. 减小 ϵ 会降低加速比, 极端情况下, 如当 ϵ 取 0.0001 时, τ -leap 几乎退化成 MC 算法. 此外, 可以看到 τ -leap 的加速比随着模拟过程越来越大. 这是由于 MC 和 τ -leap 算法本身的特点导致的. 由文献 [21] 的分析可知, 基于 MC 的颗粒破裂动力学模拟的速度随着系统中颗粒数目的增加而衰减; 而 τ -leap 算法的效率几乎不受系统颗粒数目的影响. 这解释了我们看到的加速比越来越大的现象.

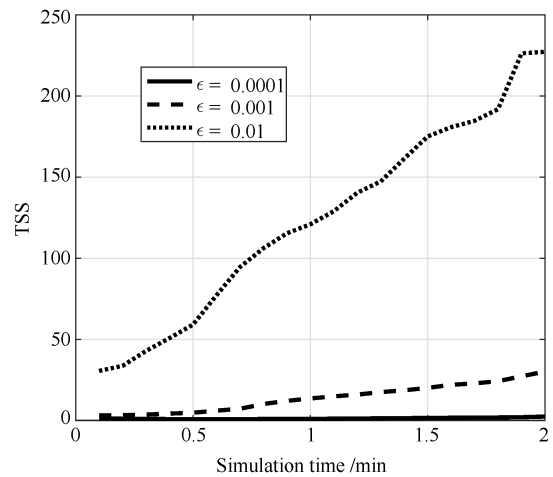


图 7 基于 τ -leap 的加速算法相对于分布式参数 MC 算法的 TSS 加速倍数

Fig. 7 The magnitude of speedup achieved by τ -leap over the MC algorithm

4 结论

磨矿是选矿过程中最重要的工艺过程之一, 对于选矿过程的运行品质至关重要. 而物料的粒度分布是反映磨矿过程中颗粒破碎情况的直接工艺指标. 本文以磨矿过程为背景, 介绍了两种颗粒研磨过程

表 4 基于 τ -leap 的加速算法相对于分布式参数 MC 算法的 TSS 加速倍数
Table 4 The magnitude of speedup achieved by τ -leap over the MC algorithm

t (min)	0~0.1	0.1~0.2	0.2~0.3	0.3~0.4	0.4~0.5	0.5~0.6	0.6~0.7	0.7~0.8	0.8~0.9	0.9~1.0
TSS ($\epsilon = 1 \times 10^4$)	1.2654	1.0361	1.0256	0.8022	0.8122	0.8222	0.8322	0.8461	0.8522	0.9578
TSS ($\epsilon = 1 \times 10^3$)	3.0340	3.2120	3.6136	4.1904	4.7486	6.0206	7.1760	10.1046	12.0586	13.5427
TSS ($\epsilon = 1 \times 10^2$)	30.5564	33.6722	42.8775	50.7719	59.0955	77.1530	94.5627	106.1508	115.3813	120.9510
t (min)	1.0~1.1	1.1~1.2	1.2~1.3	1.3~1.4	1.4~1.5	1.5~1.6	1.6~1.7	1.7~1.8	1.8~1.9	1.9~2.0
TSS ($\epsilon = 1 \times 10^4$)	1.0695	1.1649	1.2973	1.3904	1.5310	1.6498	1.7270	1.8265	1.9938	2.3590
TSS ($\epsilon = 1 \times 10^3$)	14.8307	15.9210	17.4453	18.5544	20.0657	21.9578	22.8312	24.1852	27.2161	30.1375
TSS ($\epsilon = 1 \times 10^2$)	129.0026	140.3950	147.3712	161.3876	175.1282	180.8352	184.7492	191.9528	226.3620	227.326

的动力学模拟算法. 首先提出一种分布式参数的连续磨蒙特卡洛动力学模拟算法, 进而针对该算法效率较低的问题, 提出了基于 τ -leap 的模拟加速算法. 本文的主要贡献是:

1) 提出分布式参数蒙特卡洛动力学模拟算法. 将磨机分为多个串级连接的子磨机. 对于每段子磨机, 径向上划分为若干粒级, 假设子磨机内物料均匀混合而且颗粒在子磨机之间移动是在破裂之后发生的. 那么子磨机内的破裂过程的模拟采用批次磨的时间驱动蒙特卡洛模拟方法实现. 根据分段思想, 相邻的子磨机间有物料的相互传递. 因此, 子磨机内颗粒数目的增加是由大颗粒破裂或者子磨机间颗粒的流入而导致的. 从微观角度来观察, 磨机内颗粒可能发生三种事件: 向前移动事件, 是由对流和物料本身的循环流动引起; 向后移动事件, 是由扩散作用引起; 破裂事件, 是由磨机研磨作用而导致颗粒破裂至比其粒级小的粒级内. 将上述三种事件统称为颗粒消失事件. 以此为基础定义系统状态矩阵、倾向函数和基于蒙特卡洛的模拟计算方法.

2) 基于 τ -leap 的磨矿过程分布式参数蒙特卡洛模拟加速算法. 分布式参数蒙特卡洛动力学模拟算法能够精确刻画每一个子磨机的动态变化过程, 进而得到整个磨机中不同时间不同位置各个时间点的粒度分布. 但由于该方法需要记录每一次事件的发生, 计算量极大, 只能应用于小体系的模拟. 针对这个问题, 我们提出基于 τ -leap 的模拟加速方法. 将模拟时间划分为数个时间片作为步进时间, 同时假设在每个步进时间内有多个反应事件同时发生. 即不必每次事件都更新系统状态, 而是根据步进时间片内多个事件的累积效应对系统状态进行更新, 从而提高模拟速度. 为了解决状态不一致的问题, 我们创新性地提出了一种基于缓冲区的解决方法. 考虑到移动事件和破裂事件直接的独立性, 对每个子磨机系统开辟两个状态同步缓冲区: 前向移动缓冲区和后向移动缓冲区. 在计算过程中, 将前向移动和后向移动的颗粒分别暂存在缓冲区, 等到达下一个

同步时间点, 再将缓冲区的颗粒与相邻子磨机进行交换. 本质上, 这相当于将前/后向的移动事件和破裂事件分别进行处理. 相应地, 在进行破裂事件的处理中, 将子磨机和其附属的缓冲区视为一个独立的批次磨过程, 并更新状态转移矩阵和倾向函数.

对本文工作加以总结如下:

1) 本文提出径向离散化的模型框架来解决磨机连续磨过程的分布式参数蒙特卡洛动力学模拟问题, 从仿真结果来看, 算法能够精确地刻画颗粒的随机破碎和径向方向的流动; 但是蒙特卡洛动力学模拟的计算速度远大于差分数值算法.

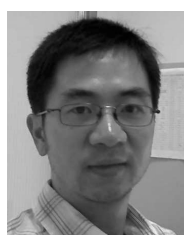
2) 针对分布式参数蒙特卡洛动力学算法的速度问题, 提出基于 τ -leap 加速算法来解决这一问题. 通过仿真实验以及算法精度与速度的比较分析可知, 基于 τ -leap 的方法在本质上是一种对蒙特卡洛动力学算法的精度和速度的平衡机制, 通过调节参数 ϵ 可以实现速度和精度的取舍.

3) 本文所提出的方法, 可以用于球磨机或棒磨机的分布参数动力学模拟.

References

- Chai T Y. Optimal operational control for complex industrial processes. *The International Federation of Automatic Control*, 2012, 722–731
- Wang X L, Wang Y L, Yang C H, Xu D G, Gui W H. Hybrid modeling of an industrial grinding-classification process. *Powder Technology*, 2015, **279**(7): 75–85
- Zhou P, Chai T Y, Wang H. Intelligent optimal-setting control for grinding circuits of mineral processing process. *IEEE Transactions on Automation Science and Engineering*, 2009, **6**(4): 730–743
- Dai W, Zhou P, Zhao D Y, Lu S W, Chai T Y. Hardware-in-the-loop simulation platform for supervisory control of mineral grinding process. *Powder Technology*, 2016, **288**(Supplement C): 422–434
- Lu S W, Zhou P, Chai T Y, Dai W. Modeling and simulation of whole ball mill grinding plant for integrated control. *IEEE Transactions on Automation Science and Engineering*, 2014, **1**(4): 1004–1019

- 6 Zhou P, Lu S W, Yuan M, Chai T Y. Survey on higher-level advanced control for grinding circuits operation. *Powder Technology*, 2016, **288**(Supplement C): 324–338
- 7 Ballantyne G R, Powell M S. Benchmarking comminution energy consumption for the processing of copper and gold ores. *Minerals Engineering*, 2014, 65: 109–114
- 8 Hua Cheng-Cheng, Wang Hong, LU Shao-Wen, Wang Hong. Knowledge automation-oriented brain cognitive feature and control effect analysis of operator in mineral grinding process. *Acta Automatica Sinica*, 2017, **43**(11): 1898–1907 (化成城, 王宏, 卢绍文, 王宏. 面向知识自动化的磨矿系统操作员脑认知特征与控制效果的相关分析. *自动化学报*, 2017, **43**(11): 1898–1907)
- 9 Pani A K, Mohanta H K. Soft sensing of particle size in a grinding process: Application of support vector regression, fuzzy inference and adaptive neuro fuzzy inference techniques for online monitoring of cement fineness. *Powder Technology*, 2014, **264**(9): 484–497
- 10 Zhang W, You C F. Numerical approach to predict particle breakage in dense flows by coupling multiphase particle-in-cell and monte carlo methods. *Powder Technology*, 2015, **283**(10): 128–136
- 11 King R P. Modeling and Simulation of Mineral Processing Systems. *Butterworth-Heinemann, Oxford, Second Edition*, 2001
- 12 Lynch A J, Morrison R D. Simulation in mineral processing history, present status and possibilities. *The Journal of The South African Institute of Mining and Metallurgy*, 1999, **99**(4): 283–288
- 13 Ramkrishna D. Analysis of population balance iv: The precise connection between Monte Carlo simulation and population balances. *Chemical Engineering Science*, 1981, **36**(7): 1203–1209
- 14 Berthiaux H. Analysis of grinding processes by markov chains. *Chemical Engineering Science*, 2000, **55**(19): 4117–4127
- 15 Hasseine A, Senouci S, Attarakih M, Bart H J. Application of two analytical approaches for the solution of the population balance equations: Particle breakage process. *Chemical Engineering and Technology*, 2015, **38**(9): 1574–1584
- 16 Su Jun-Wei, Gu Zhao-Lin, Yun X X. Advances of solution methods of population balance equation for disperse phase system. *Scientia Sinica Chimica*, 2010, **40**(2): 144–160 (苏军伟, 顾兆林, Yun X X. 离散相系统群体平衡模型的求解算法. *中国科学. 化学*, 2010, **40**(2): 144–160)
- 17 Lu Shao-Wen, Yu Ce. A fast Monte Carlo algorithm for dynamic simulation of particle size distribution of grinding processes. *Acta Automatica Sinica*, 2014, **40**(9): 1903–1911 (卢绍文, 余策. 磨矿粒度动态过程的一种快速 Monte Carlo 仿真方法. *自动化学报*, 2014, **40**(9): 1903–1911)
- 18 Mishra B K. Monte Carlo Method for the Analysis of Particle Breakage, chapter 15. *Elsevier*, 2007, 637–660
- 19 Khalili S, Lin Y L, Armaou A, Matsoukas T. Constant number Monte Carlo simulation of population balances with multiple growth mechanisms. *AIChE Journal*, 2010, **56**(12): 3137–3145
- 20 Yu M J, Lin J Z, Cao J J, Seipenbusch M. An analytical solution for the population balance equation using a moment method. *Particuology*, 2015, **18**: 194–200
- 21 Lu S W. Acceleration of kinetic Monte Carlo simulation of particle breakage process during grinding with controlled accuracy. *Powder Technology*, 2016, **301**(Supplement C): 186–196
- 22 Gillespie D T. Stochastic simulation of chemical kinetics. *Annual Review of Physical Chemistry*, 2007, **58**(1): 35–55
- 23 Kis P B, Mihalyko C, Lakatos B G. Discrete model for analysis and design of grinding mill-classifier systems. *Chemical Engineering and Processing*, 2006, **45**(5): 340–349
- 24 Battaile C C. The kinetic Monte Carlo method: Foundation, implementation, and application. *Computer Methods in Applied Mechanics and Engineering*, 2008, **197**(41-42): 3386–3398
- 25 Liu X, Lu S W. Acceleration of the dynamic simulation of grinding particle size distribution based on tau-leap method. In: Proceedings of the 11th World Congress on Intelligent Control and Automation (WCICA). Shenyang, China: 2014, 772–776



卢绍文 东北大学流程工业综合自动化国家重点实验室教授. 于 2006 年获得伦敦大学皇后玛丽学院电子工程学博士学位. 主要研究方向为多尺度随机建模方法, 模拟软件设计和数据可视化方法. 本文通信作者.

E-mail: lusw@mail.neu.edu.cn

(LU Shao-Wen Professor at the

State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University. He received his Ph.D. degree in electronic engineering from the Queen Mary University of London. His research interest covers multi-scale modeling, working with both stochastic simulation and visualization methods. Corresponding author of this paper.)



蔚润琴 东北大学信息科学与工程学院硕士生, 2012 年获太原理工大学自动化专业学士学位. 主要研究方向为蒙特卡洛仿真. E-mail: 18842300547@163.com

(YU Run-Qin Master student at

Information Science and Engineering College, Northeastern University. She received her bachelor degree in automation from Taiyuan University of Technology in 2012. Her research interest covers Monte Carlo simulation methods.)



崔玉洁 东北大学秦皇岛分校控制工程学院副教授. 主要研究方向为机电一体化, 机器人技术.

E-mail: cyjhjn@163.com

(CUI Yu-Jie Associate professor

at the Control Engineering College, Northeastern University at Qinhuangdao. Her research interest covers mechanical electronics and robotics.)