

# 基于时驱硬球算法与格子玻尔兹曼方法的颗粒流体系统直接数值模拟

周国峰<sup>①②</sup>, 王利民<sup>①\*</sup>, 王小伟<sup>①\*</sup>, 熊勤钢<sup>①②</sup>, 葛蔚<sup>①</sup>

① 中国科学院过程工程研究所, 多相复杂系统国家重点实验室, 北京 100190;

② 中国科学院研究生院, 北京 100049

\* 联系人, E-mail: lmwang@home.ipe.ac.cn, xwwang@home.ipe.ac.cn

2010-12-08 收稿, 2011-01-28 接受

国家重大科研装备研制项目(ZDYZ2008-2)、国家科技重大专项(2008ZX05014-003-006HZ)、中国科学院知识创新工程项目(KGCX2-YW-124)和科技部中澳国际合作项目(2007DFA41320)资助

**摘要** 实现了一种直接数值模拟颗粒流体系统的耦合算法, 颗粒间相互作用由时驱硬球算法描述, 而流体的控制方程采用格子玻尔兹曼方法求解, 流固耦合用浸入运动边界法实现. 该方法使用欧拉网格求解流场, 拉格朗日网格跟踪颗粒, 避免了非结构化贴体网格方法需要重新划分网格的问题. 通过模拟两个圆形颗粒在黏性流体中的沉降过程, 成功地复现了经典的 Drafting-Kissing-Tumbling(DKT)过程, 验证了耦合算法的有效性.

## 关键词

时驱硬球算法  
格子玻尔兹曼方法  
浸入运动边界法  
颗粒流体系统  
直接数值模拟

颗粒流体系统广泛存在于自然界和工业过程, 如沙尘暴、泥石流、血细胞在血管中的流动、流化床反应器、气力输送等<sup>[1]</sup>, 涉及复杂的多相流动过程. 传统的多相流数值模拟方法, 如双流体模型和颗粒轨道模型, 难以获得颗粒流体间相互作用的细节信息. 为此, 必须进行计算网格尺度远小于颗粒特征尺度的直接数值模拟(direct numerical simulation, DNS). 随着计算能力的提高和并行算法研究的深入, 这种方法的可行性与重要性正逐渐显露.

Hu 等人<sup>[2]</sup>、Johnson 和 Tezduyar<sup>[3,4]</sup>采用非结构化网格的有限元方法直接模拟了大量二维和三维的刚性颗粒在牛顿流体中运动. 该算法基于任意拉格朗日欧拉(arbitrary lagrangian eulerian, ALE)技术, 使用动网格算法处理时变的流体域. ALE 方法需要重新生成网格, 消耗了大量的计算资源. Glowinski 等人<sup>[5]</sup>和 Pan 等人<sup>[6]</sup>用基于分布式拉格朗日乘子的虚拟区域方法直接模拟了类似体系. 他们的有限元计算使用固定的结构化网格, 避免了网格重新生成问题, 但不

能有效处理颗粒间的碰撞<sup>[7]</sup>. 上述报道都是针对液固系统, 对更高密度比的气固系统, 麻景森等人<sup>[8,9]</sup>用宏观拟颗粒方法(macro-scale pseudo particle modeling, MaPPM)实现了对包含 1024 个颗粒的悬浮系统的模拟, 考察了非均匀结构对曳力的影响, 最近, 熊勤钢等人<sup>[10]</sup>用大规模 GPU 并行, 计算了 10 万以上的固体颗粒. 但因为相互作用的邻近粒子数较多并需要动态搜索、更新, 所以计算量相当大.

格子玻尔兹曼方法(lattice Boltzmann method, LBM)是一种新兴的流体力学计算方法, 具有高效、内在的并行性和鲁棒性, 1990 年被 Ladd 等人<sup>[11,12]</sup>成功应用于颗粒流体悬浮体系的模拟. 该方法通过修正的反弹格式实现运动边界上的无滑移条件, 根据动量变化计算相间作用力, 成功地模拟了液固悬浮. 在 Ladd 的方法中, 固体颗粒内部也被流体粒子占据, 因此该方法只适用固体密度大于流体密度的问题, 对此, Aidun 等人<sup>[13]</sup>通过避免使用颗粒内部流体的方法来解决. Qi 等人<sup>[14]</sup>在 Ladd 方法的基础上,

提出了满足局部质量守恒的方法, 并以此模拟了二维矩形槽道中的多颗粒沉降. 然而, Ladd 的方法用规则的边界点来近似运动边界, 模拟边界与物理边界不重合, 有时会导致流固间作用力的计算产生振荡. Feng 和 Michaelides<sup>[15]</sup>将允许颗粒边界变形的浸入边界法<sup>[16]</sup>(immersed boundary method, IBM)引入了 LBM, 以处理流固耦合, 但对颗粒刚性系数的选择还缺乏严格的标准.

Noble 和 Torczynski<sup>[17]</sup>提出了一种新的流固耦合方式, 称为浸入运动边界法<sup>[18]</sup>(immersed moving boundary method). 它通过在演化方程中加入附加碰撞项, 以实现运动边界上的无滑移条件, 同时保留了 LBM 演化方程局部性的优点. Cook 等人<sup>[19]</sup>利用该方法, 并对颗粒间相互作用采用离散单元模型<sup>[20]</sup>(discrete element model, DEM), 结合 LBM 模拟了黏结颗粒在流体侵蚀下的脱落过程. Feng 等人<sup>[18]</sup>在此基础上, 将大涡模拟<sup>[21]</sup>(large eddy simulation, LES)引入 LBM, 并耦合 LBM 和 DEM 成功模拟了高雷诺数的颗粒流体系统. 与 Ladd 方法不同, 浸入运动边界法通过引入格子固含率实现了不同类型格子光滑过渡, 使得该边界处理方法有亚网格精度<sup>[19]</sup>. 本文使用浸入运动边界法实现流固耦合.

颗粒间相互作用大体有两类处理方法<sup>[1]</sup>: 软球模型和硬球模型. 软球模型含有弹性、阻尼以及滑移的力学机制, 采用时间驱动算法以相同的时间步长推进. 这涉及耗时的积分运算, 其间引入的数值误差可能引起计算失稳. 硬球模型认为颗粒间的作用是瞬时的二体碰撞, 根据动量守恒定理确定碰撞后的状态. 硬球模型通常采用事件驱动算法, 即模拟体系按碰撞事件发生的先后顺序演化, 其时间步长不固定, 不利于统计分析; 同时该算法的每一步都要更新事件列表, 其串行的本质使之难以获得良好的并行效果<sup>[22]</sup>; 此外, 事驱硬球模型处理密集颗粒流时, 容易发生碰撞坍塌<sup>[23]</sup>, 导致模拟体系停止演化. 对此, Hopkins 等人<sup>[24]</sup>提出了兼备硬球模型数值精度高和软球模型易于并行、统计的时间驱动的硬球算法. 而在葛蔚和李静海<sup>[25]</sup>提出的拟颗粒模拟(pseudo-particle modeling, PPM)中, 颗粒和离散表征流体的拟颗粒从物理模型上就处理为时驱演化的可重叠的硬球, 成功地实现了微尺度气固流化床的直接数值模拟. 欧阳洁和李静海<sup>[26,27]</sup>发展的模拟循环流化床的颗粒轨道模型中, 固体颗粒间的相互作用

也采取了类似的计算策略, 效果良好. 鉴于此, 本文中对颗粒间的相互作用也采用时驱硬球算法. 最近, 王利民等人<sup>[28]</sup>首次报道了该耦合算法和它在流态化模拟中取得的初步结果, 本文将详细介绍该耦合算法的实现细节及其有效性的验证.

## 1 耦合算法

本文实现了一种基于时驱硬球算法与格子玻尔兹曼方法的直接数值模拟算法. 离散颗粒相的计算采用时驱硬球算法, 而流体的控制方程用格子玻尔兹曼方法求解, 颗粒与流体间的相互作用通过 Noble 和 Torzynski 提出的浸入运动边界法实现. 首先简要介绍这 3 种方法的相关内容. 同时, 考虑到浸入运动边界法实现的关键在于格子固含率的求解, 我们还详细地给出了 2 种具体求解方法.

### 1.1 时驱硬球算法

时驱硬球算法假定颗粒间的相互作用是瞬时的二体碰撞, 根据碰撞前后动量守恒和碰撞的恢复系数确定碰撞后的状态, 模拟体系的演化按固定的时间步长向前推进. 在每个时间步长内包含两个过程: 自由运动和硬球碰撞. 首先, 在该时间步长内, 颗粒的运动满足牛顿运动方程, 即

$$\begin{aligned} m \frac{dV}{dt} &= F, \\ I \frac{d\omega}{dt} &= T. \end{aligned} \quad (1)$$

$m$  和  $I$  分别代表颗粒的质量和转动惯量,  $V$  和  $\omega$  为质心的平动速度和转动角速度,  $F$  和  $T$  为颗粒受到的外力和外力矩; 在该时间步长, 如果任何两个颗粒之间的距离小于它们的半径之和, 且仍在相互靠近, 即同时满足  $|P_1 - P_2| < (r_1 + r_2)$  和  $(P_1 - P_2)_2 \cdot (V_1 - V_2) < 0$ , 则它们以硬球方式碰撞, 如图 1 所示.

碰撞后颗粒的速度  $V_1'$  和  $V_2'$  分别为

$$\begin{aligned} V_1' &= V_1 - \frac{(1+e)m_2}{m_1+m_2} \frac{(P_1 - P_2)_2 \cdot (V_1 - V_2)}{|P_1 - P_2|^2} (P_1 - P_2), \\ V_2' &= V_2 + \frac{(1+e)m_1}{m_1+m_2} \frac{(P_1 - P_2)_2 \cdot (V_1 - V_2)}{|P_1 - P_2|^2} (P_1 - P_2). \end{aligned} \quad (2)$$

式中,  $e$  为碰撞的恢复系数,  $P_1$  和  $P_2$  分别为 2 个颗粒的位置. 模拟体系用时驱算法驱动, 因此允许颗粒间有微量的重叠.

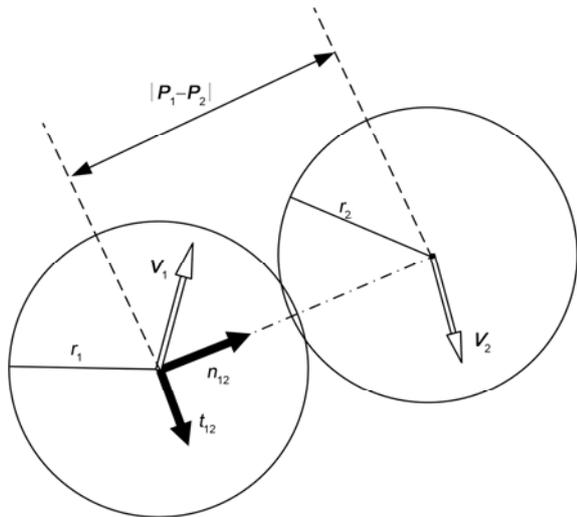


图1 两个颗粒间的碰撞分析

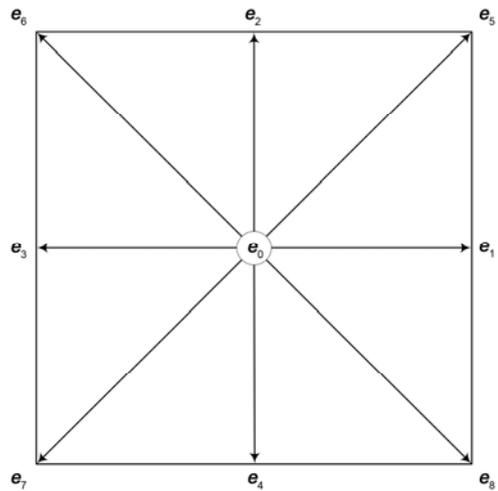


图2 D2Q9 格子模型

### 1.2 LBM 理论基础

LBM 起源于格子气自动机, 以分子动理论和统计力学为基础, 能够准确地复现宏观的 N-S 方程, 因而被广泛应用于流体流动模拟<sup>[29]</sup>.

LBM 中, 流动区域被划分成规则的格点, 各方向的空间间隔都为  $h$ , 流体相用分布在格点上的流体粒子代表. 由于速度空间也是离散的, 流体粒子只能沿规定的方向迁移至邻近格点, 不同的格子模型规定了不同的离散速度. 如图 2, 本文采用 D2Q9 格子模型, 该模型分布函数的演化方程为

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) = f_i(\mathbf{x}, t) + \frac{1}{\tau} [f_i^{\text{eq}}(\rho, \mathbf{v}) - f_i(\mathbf{x}, t)]. \quad (3)$$

$f_i(\mathbf{x}, t)$  为  $t$  时刻在格点  $\mathbf{x}$  处  $\mathbf{e}_i$  方向的粒子分布函数,  $\tau$  为无量纲弛豫时间,  $\Delta t$  为时间间隔,  $f_i^{\text{eq}}(\rho, \mathbf{v})$  为  $t$  时刻  $\mathbf{e}_i$  方向的平衡分布函数. 演化方程包含碰撞和迁移两个基本步骤, 对应的表达式分别为

$$\bar{f}_i(\mathbf{x}, t) = f_i(\mathbf{x}, t) + \frac{1}{\tau} [f_i^{\text{eq}}(\rho, \mathbf{v}) - f_i(\mathbf{x}, t)]. \quad (4)$$

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) = \bar{f}_i(\mathbf{x}, t). \quad (5)$$

式中  $\bar{f}_i(\mathbf{x}, t)$  是碰撞后的粒子分布函数.

在低  $Ma$  数下, 通过对 Maxwell-Boltzmann 平衡分布方程的二阶 Taylor 展开, 可以推导出粒子的平衡分布函数<sup>[30]</sup>

$$f_i^{\text{eq}}(\rho, \mathbf{v}) = \omega_i \rho \left( 1 + \frac{\mathbf{e}_i \cdot \mathbf{v}}{C_s^2} + \frac{1}{2} \frac{(\mathbf{e}_i \cdot \mathbf{v})^2}{C_s^4} - \frac{1}{2} \frac{\mathbf{v} \cdot \mathbf{v}}{C_s^2} \right) \quad (6)$$

$i = 0, 1, \dots, 8.$

式中,  $\omega_i$  为权系数, 其值为

$$\omega_i = \begin{cases} 4/9 & i = 0, \\ 1/9 & i = 1, 2, 3, 4, \\ 1/36 & i = 5, 6, 7, 8. \end{cases} \quad (7)$$

LBM 求解的原始变量是格点上的粒子分布函数, 流体的宏观密度与速度可由粒子分布函数求得, 具体关系式为

$$\rho(\mathbf{x}, t) = \sum_{i=0}^8 f_i(\mathbf{x}, t), \quad (8)$$

$$\rho \mathbf{v}(\mathbf{x}, t) = \sum_{i=0}^8 \mathbf{e}_i f_i(\mathbf{x}, t).$$

$\rho, \mathbf{v}$  分别表示格点上的密度和速度. 格点上的压力通过理想气体状态方程与密度相关联,

$$p = C_s^2 \rho. \quad (9)$$

式中,  $C_s = 1/\sqrt{3}$  是格子声速. 流体的运动学黏性系数  $\nu$  为

$$\nu = \frac{1}{3} \left( \tau - \frac{1}{2} \right) \frac{h^2}{\Delta t}. \quad (10)$$

通过 Chapman-Engskog 展开可证明<sup>[31]</sup>, D2Q9 格子模型对应的宏观方程是二维可压缩 N-S 方程, 时间和空间均为二阶精度<sup>[29]</sup>. 理论上 LBM 只能用于可压缩流动的模拟. 为了将其用于不可压缩流动的模拟, 可借鉴经典计算流体力学 (computational fluid dynamics, CFD) 中处理不可压缩流动的人工压缩法思想<sup>[32]</sup>, 即把不可压缩流动当成拟可压缩流动. 但要使 LBM 的解收敛于不可压缩流动的 N-S 方程, 流动的  $Ma$  数必须非常小, 至少应小于 0.1.

### 1.3 流固耦合

(1) 浸入运动边界法. 颗粒流体系统中, 颗粒受到流体施加的作用力, 流体在颗粒表面须满足无滑移边界条件. 因此, 准确地模拟颗粒流体系统还要正确地处理流固耦合. 本文采用 Noble 和 Torzynski<sup>[17]</sup> 提出的浸入运动边界法实现, 该方法引入了格子控制体(图 3(a))和格子固含率  $\varepsilon_s$ (图 3(b))的概念. 格子固含率定义为控制体被颗粒占据的面积与格子控制体面积的比, 其大小表征了格子与颗粒间相互作用的强弱. 浸入运动边界法不需要用规则的格子来近似表示固体颗粒, 避免了由此引起作用力的数值振荡. 值得说明的是, 虽然浸入边界法和浸入运动边界法都是通过流体的控制方程中加入一种体积力来实现流固耦合, 但它们的具体实现手段是截然不同的. 浸入边界法假设颗粒的边界是可变形的, 再依据应力-形变的关系确定流固间作用力<sup>[15]</sup>. 浸入运动边界法则是由运动颗粒与当地流体间的动量差确定该作用力.

加入考虑固体颗粒对流场影响的附加碰撞项后, 分布函数的演化方程为<sup>[17]</sup>

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) = f_i(\mathbf{x}, t) + \frac{1}{\tau} (1 - \beta(\varepsilon_s, \tau)) \times (f_i^{\text{eq}}(\rho, \mathbf{v}) - f_i(\mathbf{x}, t)) + \beta(\varepsilon_s, \tau) \Omega_i^s. \quad (11)$$

$\beta(\varepsilon_s, \tau)$  和  $\Omega_i^s$  的表达式分别为<sup>[17]</sup>

$$\beta(\varepsilon_s, \tau) = \frac{\varepsilon_s(\tau - 0.5)}{(1 - \varepsilon_s) + (\tau - 0.5)}. \quad (12)$$

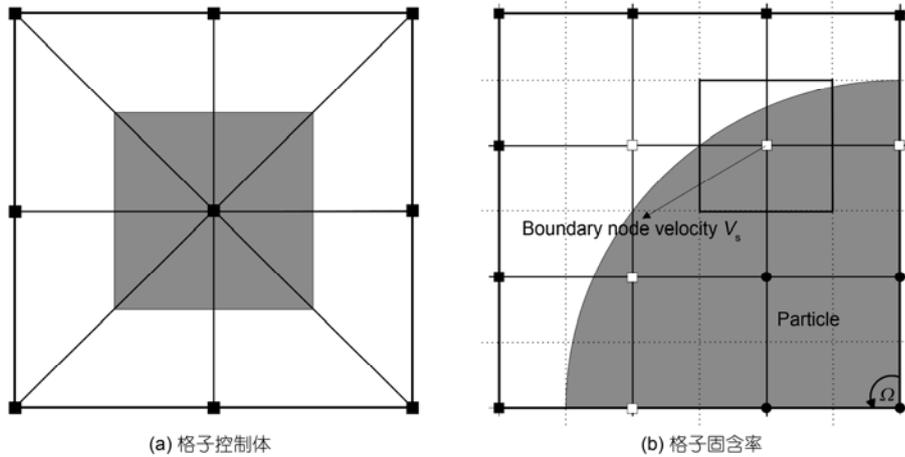


图 3 Noble 和 Torzynski 的浸入运动边界法<sup>[17]</sup>

$\Omega_i^s = f_{-i}(\mathbf{x}, t) - f_i(\mathbf{x}, t) + f_i^{\text{eq}}(\rho, \mathbf{V}_s) - f_{-i}^{\text{eq}}(\rho, \mathbf{v}). \quad (13)$   
 式中,  $-i$  表示与  $i$  相反的方向, 如与 1 相反的方向为 3(参考图 2).  $\mathbf{v}$  是  $t$  时刻格点  $\mathbf{x}$  的速度,  $\mathbf{V}_s$  为固体颗粒在当前格点  $\mathbf{x}$  处的速度. 假定颗粒质心位置、平动速度和角速度分别用  $\mathbf{x}_c$ ,  $\mathbf{V}_c$  和  $\boldsymbol{\omega}$  表示, 则  $\mathbf{V}_s$  可表示为

$$\mathbf{V}_s = \mathbf{V}_c + \boldsymbol{\omega} \times (\mathbf{x} - \mathbf{x}_c). \quad (14)$$

修正后的演化方程可使得被颗粒覆盖的格点碰撞后的速度等于颗粒在该格点的速度, 从而实现无滑移边界条件. 浸入运动边界法可以直接计算出运动颗粒对流体的作用力, 根据牛顿第三定律, 由此也可以得到流体对颗粒的作用力. 图 4 标识了控制体被某个颗粒覆盖或部分覆盖的所有格点. 加和这些格点碰撞前后的动量变化, 可以得到流体对该颗粒的作用力<sup>[19]</sup>  $\mathbf{F}_{p \rightarrow f}$ , 即

$$\mathbf{F}_{p \rightarrow f} = \frac{h^2}{\Delta t} \sum_{j=1}^n \left( \beta_j \sum_{i=1}^8 \Omega_i^s \mathbf{e}_i \right). \quad (15)$$

其中,  $n$  是被颗粒覆盖的格点个数, 相应的力矩为

$$\mathbf{T}_{p \rightarrow f} = \frac{h^2}{\Delta t} \sum_{j=1}^n \left( (\mathbf{x}_j - \mathbf{x}_c) \times \beta_j \sum_{i=1}^8 \Omega_i^s \mathbf{e}_i \right). \quad (16)$$

式中  $\mathbf{x}_j$  是第  $j$  个被覆盖格点的位置.

综上所述, 耦合算法的计算流程为

- (i) 在每个时间步的起点  $t = t_n$ , 已知流场信息和颗粒的位置、平动速度及角速度;
- (ii) 根据颗粒位置信息, 判断格子是否被颗粒覆盖, 并加以标记(参考图 4). 对被颗粒部分覆盖的格点, 计算其控制体内的固含率;

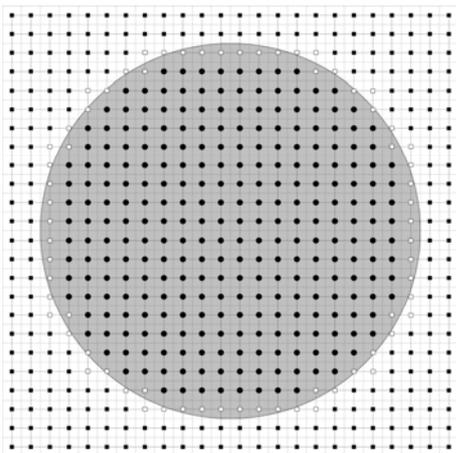


图4 颗粒周围格子的标识

■未被覆盖的格子; □被部分覆盖的格子; ●完全被覆盖的格子

(iii) 用 LBM 求解下一时刻的流场, 即  $t_{n+1}$  时刻的流场;

(iv) 由  $t_{n+1}$  时刻的流场信息, 计算颗粒此时受到的力和力矩;

(v) 由上一步得到的颗粒受力信息, 更新颗粒的速度和位置, 得到颗粒在  $t_{n+1}$  时刻的位置和速度;

(vi) 判断颗粒在  $t_{n+1}$  是否存在碰撞, 计算颗粒间相互作用;

(vii) 以  $t_{n+1}$  时刻的信息作为输入信息, 依次重复上述所有步骤, 直至达到计算要求。

(2) 格子固含率的计算. 上述讨论表明, 格子固含率的计算是浸入运动边界法中的关键步骤. 颗粒运动后, 格子固含率会随之发生变化, 因此每步都要重新计算. 当颗粒数很多时, 格子固含率的计算会占据大量的计算资源, 因此准确高效地求解格子固含率显得尤其重要. 为此我们提出了分类求解法. 先计算方形格子控制体的 4 个顶点到颗粒中心的距离, 根据距离与颗粒半径的大小关系判定顶点在颗粒内部还是外部, 并依此对颗粒与格子控制体的不同相交情况进行分类. 对每个顶点而言, 都有两种情况: 在颗粒内部或外部, 4 个顶点的组合理论上共有 16 种情况. 去除 4 个顶点同时位于颗粒内部或外部这两种情况, 再排除左下角和右上角位于颗粒内部、左上角和右下角位于颗粒外部或左下角和右上角位于颗粒外部、左上角和右下角位于颗粒内部这两种不可能发生的情况, 还剩下 12 种相交情况, 具体如图 5. 用该方法求格子固含率时, 首先判断格子控制体与圆形颗

粒属于哪一类相交方式, 再调用相应的公式求格子控制体被颗粒占据的面积, 即可得格子固含率. 以图 5(a) 为例, 记颗粒与格子控制体的两个交点分别为  $A(R \cos \theta_2, R \sin \theta_2)$ ,  $B(R \cos \theta_1, R \sin \theta_1)$ , 则相交区域的面积为

$$S = R^2 \left( \frac{\theta_2 - \theta_1}{2} - \frac{\sin 2\theta_2 + \sin 2\theta_1}{4} + \sin \theta_1 \cos \theta_2 \right). \quad (17)$$

另外, Cook<sup>[33]</sup>还提出了一种采样统计方法: 根据求解准确度的需要, 用不同数目均匀分布的点离散格子控制体, 然后根据离散点到颗粒中心的距离判断它是否在颗粒内部, 位于颗粒内部的离散点个数占总的离散点的比例即为格子固含率. 显然, 该方法的计算量与采用的离散点数目成正比, 因此, 在保证一定准确度时应使用尽可能少的点. 图 6 为颗粒半径为  $R=30.4$ ,  $Re_p=1.0$  时, 单圆柱颗粒绕流的流动方向受力和计算时间随不同离散精度的变化情况. 模拟的网格数为  $128 \times 128$ , 水平流动方向为周期边界, 上下边界是自由运动的平板, 其速度为  $u_w = 0.01$  保持不变,  $\tau$  取 2.324, 图中的受力是用  $\rho u_w$  无量纲化后的值. 实际离散格子控制体时,  $x$  方向和  $y$  方向的离散点数目相等, 图中横坐标就是其中任意方向离散点数目. 随着离散点目数的增加, 计算时间单调增加, 流动方向受力在离散点数目等于 65 时达到稳定, 即  $R=30.4$  时的最佳离散点数目. 此外, 我们还考察了  $R=10.4, 60.8$  时的情况, 它们最佳离散点数目分别为 125 和 31, 发现最佳离散点数目随着颗粒半径的增大而减小. 这种变化趋势在其它参数条件下也可以观察到, 因此可以作为一种实用的点数选取方法.

比较两种求解方法, 采样统计法更为简单, 程序也容易实现和并行化, 但需要高准确度时计算量较大. 分类求解法则更为直观, 计算量少, 能准确求解, 但实现相对困难些. 本文采用了前一种方法计算.

## 2 算法的验证

为了验证耦合算法的有效性, 我们模拟了两个圆形颗粒在充满黏性流体的封闭通道中的自由沉降过程, 以期复现经典的“Drafting-Kissing-Tumbling”(DKT)过程<sup>[34]</sup>.

### 2.1 二维 Drafting-Kissing-Tumbling 过程模拟

Patankar 等人<sup>[35]</sup>和 Feng 等人<sup>[15]</sup>分别用虚拟区域法(fictitious domain method, FDM)和浸入边界法耦合

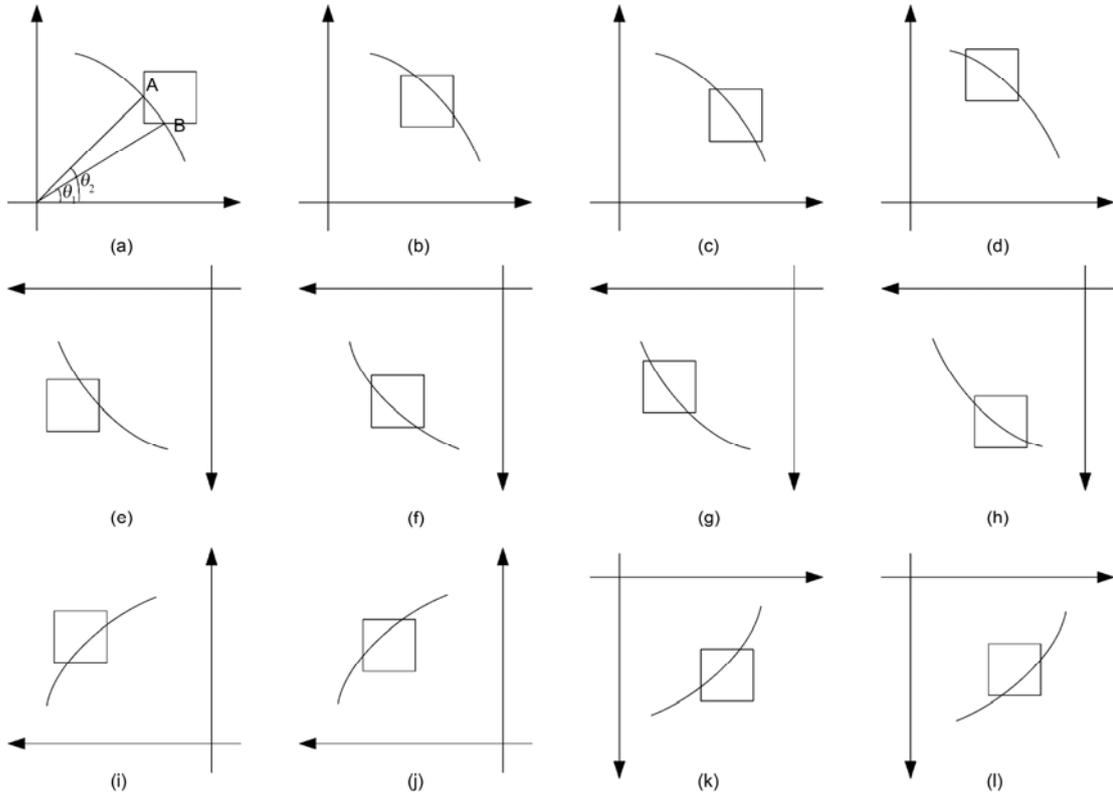


图 5 圆形颗粒与方形格子控制体 12 类相交情况的分类

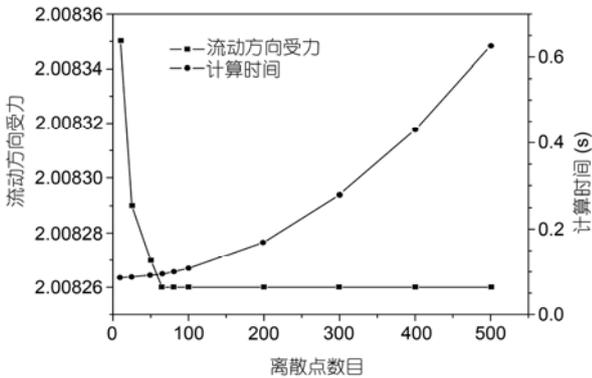


图 6 计算准确度和计算时间随离散精度的变化

LBM(immersed boundary-lattice Boltzmann model, IB-LBM)研究了这个问题. 本文模拟参数的设置与 Feng 等人<sup>[15]</sup>一致. 封闭通道的宽为 2 cm, 长为 8 cm. 通道中流体的运动学黏度为  $1.0 \times 10^{-6} \text{ m}^2/\text{s}$ , 密度为  $1000 \text{ kg/m}^3$ . 圆形颗粒的密度为  $1010 \text{ kg/m}^3$ , 半径是 0.1 cm. 两个颗粒的中心位置分别为(0.999 cm, 7.2 cm)和(1.0 cm, 6.8 cm), 参考点为通道的左下角. 位置高的颗粒记为 1, 另一个为 2. 颗粒从静止开始在重力作用下自由沉降, 重力加速度为  $9.8 \text{ m/s}^2$ . 固体

颗粒的恢复系数  $e$  取 0.8, 与颗粒与壁面间相互作用为镜面反弹.

为了将上述物理量用格子单位表示, 需要选取合适的参考量. 取  $h$  为 0.01 cm, 无量纲弛豫时间  $\tau$  为 0.65, 则时间步长  $\Delta t$  为  $5.0 \times 10^{-4} \text{ s}$ , 参考质量取每个格子所对应的质量, 即  $1.0 \times 10^{-5} \text{ kg}$ . 定下这三个参考量后, 就可以把所有的物理量转化为无量纲量, 具体见表 1.

图 7(a)给出了不同时刻颗粒在通道中的位置, 图 7 (b)为颗粒间距离随时间的变化曲线. 为了更加详细地阐明 DKT 过程中两颗粒的运动细节, 我们还给出了模拟过程中两颗粒位置和速度随时间的变化情况, 并与 FDM 和 IB-LBM 的模拟结果进行了对比, 如图 8 和 9 所示.

从图中我们可以清楚地观察到 DKT 过程. 模拟初期, 由于颗粒对流体的影响还未充分发展, 相互间的影响微弱, 它们保持恒定距离自由沉降. 随着时间的推移, 颗粒对流体的影响逐渐扩展, 并开始影响对方的运动状态. 从  $t = 0.6 \text{ s}$  开始, 颗粒 1 在颗粒 2 形成的尾涡作用下, 加速向颗粒 2 运动, 即 Drafting 过

表1 二维模拟参数设置

	$H$ (m)	$W$ (m)	$\rho f$ (kg/m <sup>3</sup> )	$\nu$ (m <sup>2</sup> /s)	$\rho s$ (kg/m <sup>3</sup> )	$R$ (m)	$m_s$ (kg)	$I$ (kg m <sup>2</sup> )	$g$ (m/s <sup>2</sup> )	$\tau$
有量纲	0.08	0.02	1000	$1.0 \times 10^{-6}$	1010	0.001	$3.17 \times 10^{-3}$	$1.59 \times 10^{-9}$	9.8	0.65
无量纲	800	200	1	0.05	1.01	10	317.14	$1.59 \times 10^4$	$2.45 \times 10^{-2}$	0.65

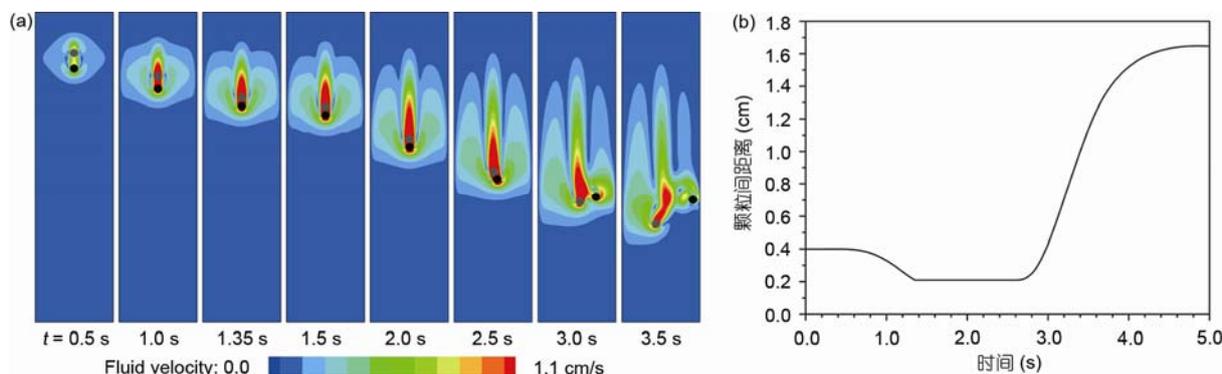


图7 二维 Drafting-Kissing-Tumbling 过程模拟  
(a) 不同时刻的流场图; (b) 颗粒间距离随时间的变化

程, 在图7(b)中表现为颗粒间的距离在这一阶段逐渐减小, 竖直方向的相对速度则不断增大, 如图9(b)所示.  $t = 1.35$  s时, 两颗粒间的距离接近颗粒的半径之和(0.2 cm), 紧密接触形成二元体, 即所谓的 Kissing. 此后颗粒间的作用力开始起作用, 其中心逐渐偏离通道的中间位置, 如图8(a)所示. 形成的二元体很不稳定, 它会迅速地翻转, 在图7(a)中表现为二元体的空间构型不断变化, 即DKT的 Tumbling 过程.  $t = 3.0$  s时, 两颗粒的空间相对位置完全翻转过来, Tumbling 过程结束. 随后, 两颗粒逐渐分离, 相互间的影响随

之减弱, 竖直方向的速度趋于稳定, 如图9(b)所示.

将我们的模拟结果与 FDM(Patankar 等人)和 IB-LBM(Feng 等人)的结果比较, 发现在 Tumbling 开始前( $t = 1.35$  s), 本文的模拟结果与文献的结果符合得很好; Tumbling 开始后, 模拟结果有所差别, 但定性的物理图景是完全一致的. 目前对 DKT 过程的物理机制还没有共识<sup>[35]</sup>, 但普遍认为流体力学不稳定性的作用是非常关键的<sup>[34]</sup>, 因此, 其模拟结果对不同数值方法的敏感性也不难理解了.

此外, 为了考察时间步长的影响, 我们还模拟了

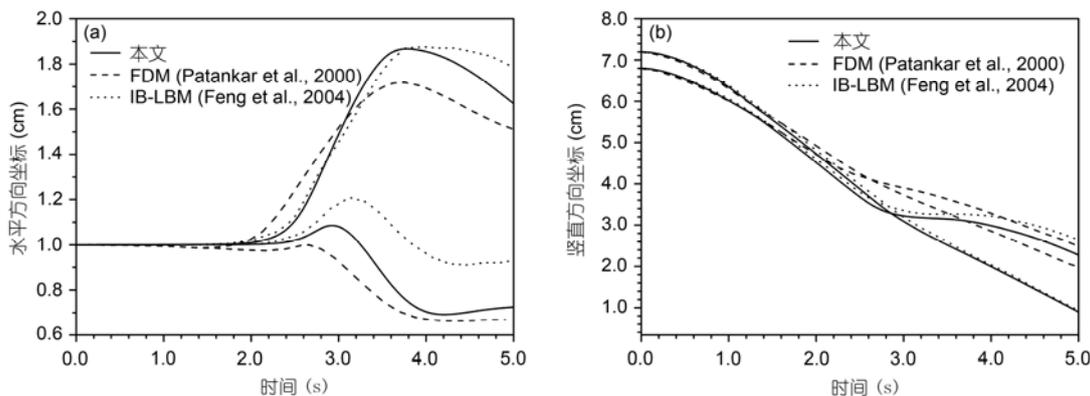


图8 不同算法模拟 DKT 过程中颗粒位置变化的比较  
(a) 水平方向坐标比较; (b) 竖直方向坐标比较

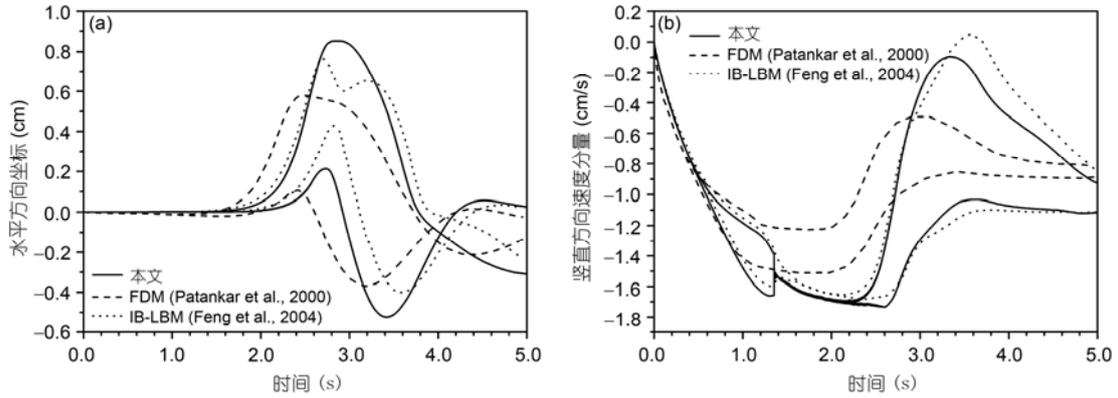


图9 不同算法模拟 DKT 过程中的颗粒速度变化的比较  
(a) 水平方向速度比较; (b) 竖直方向速度比较

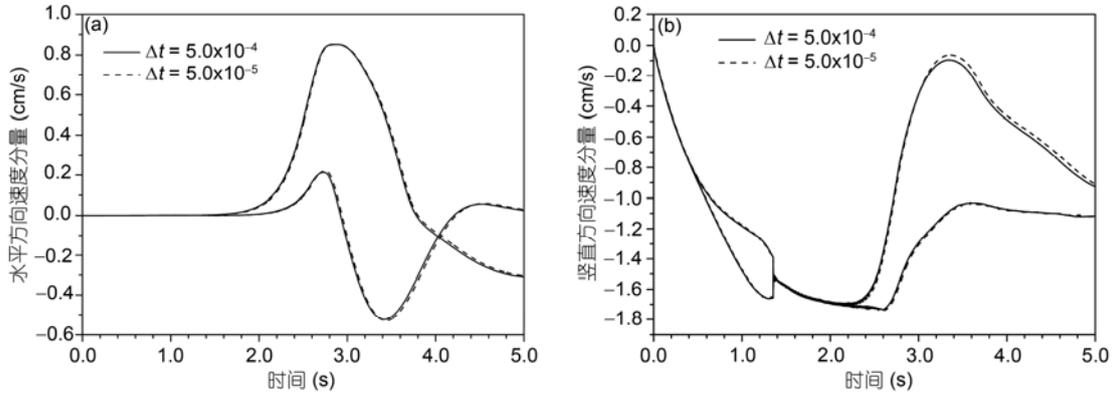


图10 不同时间步长模拟结果比较  
(a) 水平方向速度比较; (b) 竖直方向速度比较

$\Delta t$  等于  $5.0 \times 10^{-5}$  s 的情况, 如图 10 所示. 从图中我们可以看出, 两个不同时间步长的模拟结果几乎没有差别.

### 2.2 三维 Drafting-Kissing-Tumbling 过程模拟

为了进一步验证耦合算法的有效性, 我们模拟了更为接近实际的三维问题. 模拟区域为宽 2 cm, 长 2 cm, 高 8 cm 的封闭矩形通道, 通道中流体的运动学黏度为  $1.0 \times 10^{-6}$  m<sup>2</sup>/s, 密度为 1000 kg/m<sup>3</sup>. 球形颗粒的密度为 1010 kg/m<sup>3</sup>, 半径是 0.1 cm. 2 个固体颗粒的中心位置分别为 (1.0, 7.5, 1.0 cm) 和 (1.01, 7.2, 1.0 cm), 分别记为颗粒 1 和颗粒 2. 封闭矩形通道的流体开始时静止, 颗粒从静止开始在重力作用下自由沉降, 重力加速度为 9.8 m/s<sup>2</sup>, 矩形通道壁面处流体满足无滑移条件, 速度始终为零. 固体颗粒的恢复系数

$e$  取 0.8. 取  $h$  为 0.01 cm, 无量纲弛豫时间  $\tau$  为 0.65, 则时间步长  $\Delta t$  为  $5.0 \times 10^{-4}$  s, 参考质量取每个晶格所对应的质量, 即  $1.0 \times 10^{-9}$  kg, 三维模拟参数设置如表 2 所示.

图 11 给出了不同时刻颗粒在通道中的运动轨迹, 从 2 个颗粒在静止流体中沉降的三维模拟结果中, 我们清楚地观察到期待的 Drafting-Kissing-Tumbling 现象确实发生.

图 12 和图 13 还给出了模拟过程中两颗粒位置和速度随时间的变化情况. 与前面讨论的二维情况相比, 三维模拟和二维模拟定性上基本相似, 基于最大颗粒沉降速度的颗粒雷诺数都约为 25.

对所有的全分辨直接数值模拟方法仔细研究, 发现其难点在于解决短程相互作用力<sup>[36]</sup>, Dance 等人<sup>[37]</sup>研究表明模拟结果对短程作用力模型非常敏感.

表2 三维模拟参数设置

	$H$ (m)	$L$ (m)	$W$ (m)	$\rho_f$ (kg/m <sup>3</sup> )	$\nu$ (m <sup>2</sup> /s)	$\rho_s$ (kg/m <sup>3</sup> )
有量纲	0.08	0.02	0.02	1000	$1.0 \times 10^{-6}$	1010
无量纲	800	200	200	1	0.05	1.01
	$R$ (m)	$m_s$ (kg)	$I$ (kg m <sup>2</sup> )	$g$ (m/s <sup>2</sup> )	$\tau$	
有量纲	0.001	$4.23 \times 10^{-6}$	$1.67 \times 10^{-12}$	9.8	0.65	
无量纲	10	4228.53	$1.67 \times 10^5$	$2.45 \times 10^{-4}$	0.65	

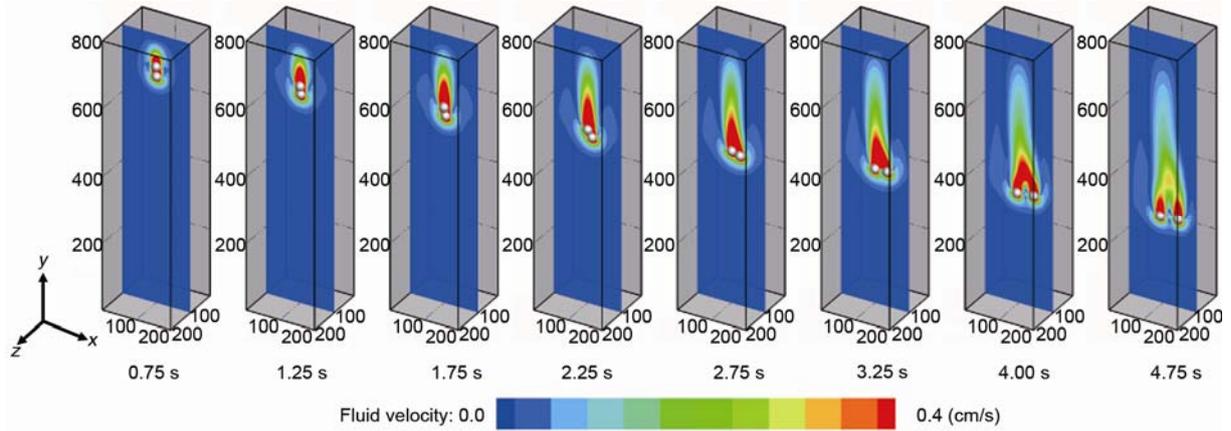


图11 三维 Drafting-Kissing-Tumbling 过程模拟

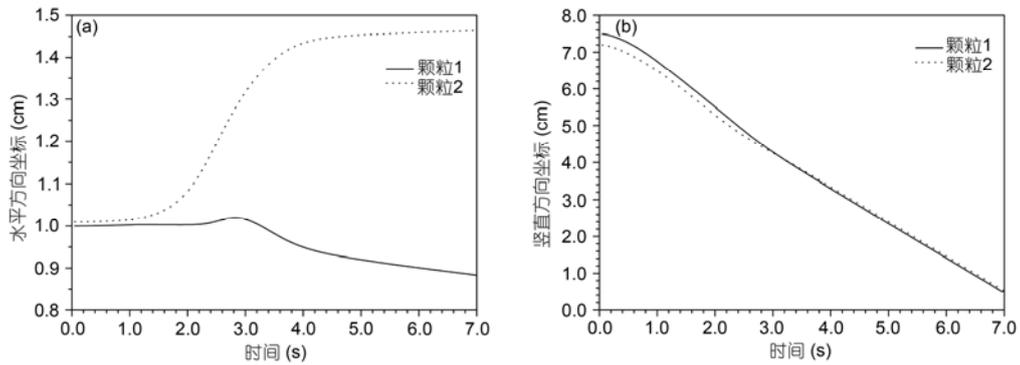


图12 三维 DKT 过程模拟中颗粒位置的变化

(a) 水平方向坐标; (b) 垂直方向坐标

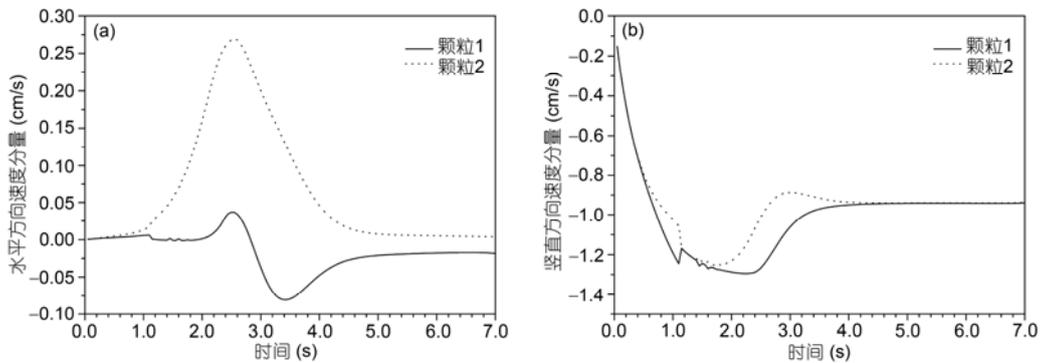


图13 三维 DKT 过程模拟中颗粒速度的变化

(a) 水平方向速度; (b) 垂直方向速度

Drafting-Kissing-Tumbling 过程是经典的多相流体力学现象, 固体颗粒间的相互作用被处理为时驱硬球算法的新直接数值模拟算法成功地复现了经典的 DKT 过程, 揭示了该直接数值模拟算法的合理性.

### 3 结论

本文针对颗粒流体系统实现了一种新的直接数值模拟算法——时驱硬球算法处理固体颗粒的相互作用, LBM 求解流体运动的控制方程, 流固耦合使用 Noble 和 Torzynski 提出的浸入运动边界法. 采用该耦合算法分别二维和三维模拟了 2 个颗粒在充满黏性流体的封闭通道中的自由沉降, 成功地复现了经

典的 DKT 过程, 验证了算法的有效性.

本文只模拟了两个颗粒在流体中的运动, 但该算法可以直接应用于多颗粒系统的模拟. 将时驱硬球算法与传统 CFD 方法结合应用于多颗粒流体系统的模拟时, 为避免颗粒的无量纲速度过大而造成颗粒间重叠过大和遗漏实际发生的碰撞<sup>[1]</sup>, 可能要采用非常小的时间步长, 此时其计算量会相当大. 而在与 LBM 耦合时, 由于计算效率的提高, 这样的时间步长是可以接受的, 也是采用人工压缩法所必须的. 此时, 流体演化的物理时间步长较小, 颗粒在一步内的位移远远小于其半径, 不会出现严重的颗粒重叠和碰撞遗漏.

### 参考文献

- 1 李静海, 欧阳洁, 高士秋, 等. 颗粒流体复杂系统的多尺度模拟. 北京: 科学出版社, 2005
- 2 Hu H H, Joseph D D, Crochet M J. Direct simulation of fluid particle motions. *Theor Comp Fluid Dyn*, 1992, 3: 285–306
- 3 Johnson A A, Tezduyar T E. Simulation of multiple spheres falling in a liquid-filled tube. *Comput Method Appl M*, 1996, 134: 351–373
- 4 Johnson A A, Tezduyar T E. 3D simulation of fluid-particle interactions with the number of particles reaching 100. *Comput Method Appl M*, 1997, 145: 301–321
- 5 Glowinski R, Pan T W, Hesla T I, et al. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow. *J Comput Phys*, 2001, 169: 363–426
- 6 Pan T W, Joseph D D, Bai R, et al. Fluidization of 1204 spheres: Simulation and experiment. *J Fluid Mech*, 2002, 451: 169–191
- 7 Zhu H P, Zhou Z Y, Yang R Y, et al. Discrete particle simulation of particulate systems: Theoretical developments. *Chem Eng Sci*, 2007, 62: 3378–3396
- 8 Ma J, Ge W, Wang X, et al. High-resolution simulation of gas-solid suspension using macro-scale particle methods. *Chem Eng Sci*, 2006, 61: 7096–7106
- 9 Ma J, Ge W, Xiong Q, et al. Direct numerical simulation of particle clustering in gas-solid flow with a macro-scale particle method. *Chem Eng Sci*, 2009, 64: 43–51
- 10 Xiong Q, Li B, Chen F, et al. Direct numerical simulation of sub-grid structures in gas-solid flow-GPU implementation of macro-scale pseudo-particle modeling. *Chem Eng Sci*, 2010, 65: 5356–5365
- 11 Ladd A J C. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *J Fluid Mech*, 1994, 271: 285–309
- 12 Ladd A J C. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results. *J Fluid Mech*, 1994, 271: 311–339
- 13 Aidun C K, Lu Y, Ding E. Direct analysis of particulate suspensions with inertia using the discrete Boltzmann equation. *J Fluid Mech*, 1998, 373: 287–311
- 14 Qi D. Lattice-Boltzmann simulations of particles in non-zero-Reynolds-number flows. *J Fluid Mech*, 1999, 385: 41–62
- 15 Feng Z G, Michaelides E E. The immersed boundary-lattice Boltzmann method for solving fluid-particles interaction problems. *J Comput Phys*, 2004, 195: 602–628
- 16 Peskin C S. Numerical analysis of blood flow in the heart. *J Comput Phys*, 1977, 25: 220–252
- 17 Noble D R, Torzynski J R. A lattice-Boltzmann method for partially saturated computational cells. *Int J Mod Phys C*, 1998, 9: 1189–1201
- 18 Feng Y T, Han K, Owen D R J. Coupled lattice Boltzmann method and discrete element modelling of particle transport in turbulent fluid flows: Computational issues. *Int J Numer Meth Eng*, 2007, 72: 1111–1134
- 19 Cook B K, Noble D R, Williams J R. A direct simulation method for particle-fluid systems. *Eng Computation*, 2004, 21: 151–168
- 20 Cundall P A, Strack O D L. A discrete numerical model for granular assemblies. *Geotechnique*, 1979, 29: 47–65
- 21 Sagaut P. *Large Eddy Simulation for Incompressible Flows: An Introduction*. Berlin: Springer Verlag, 2006

- 22 Marín M. Event-driven hard-particle molecular dynamics using bulk-synchronous parallelism. *Comput Phys Commun*, 1997, 102: 81–96
- 23 Van der Hoef M A, van Sint Annaland M, Deen N G, et al. Numerical simulation of dense gas-solid fluidized beds: A multiscale modeling strategy. *Annu Rev Fluid Mech*, 2008, 40: 47–70
- 24 Hopkins M A, Louge M Y. Inelastic microstructure in rapid granular flows of smooth disks. *Phys Fluids A: Fluid Dynamics*, 1991, 3: 47
- 25 Ge W, Li J. Pseudo-particle approach to hydrodynamics of particle-fluid systems. In: Kwauk M, Li J, eds. *Proceedings of the 5th International Conference on Circulating Fluidized Bed*. Beijing: Science Press, 1996. 260–265
- 26 Ouyang J, Li J. Particle-motion-resolved discrete model for simulating gas-solid fluidization. *Chem Eng Sci*, 1999, 54: 2077–2083
- 27 Ouyang J, Li J. Discrete simulations of heterogeneous structure and dynamic behavior in gas-solid fluidization. *Chem Eng Sci*, 1999, 54: 5427–5440
- 28 Wang L, Zhou G, Wang X, et al. Direct numerical simulation of particle-fluid systems by combining time-driven hard-sphere model and lattice Boltzmann method. *Particuology*, 2010, 8: 379–382
- 29 Chen S, Doolen G D. Lattice Boltzmann method for fluid flows. *Annu Rev Fluid Mech*, 1998, 30: 329–364
- 30 Qian Y H, Orszag S. Lattice BGK models for the Navier-Stokes equation: Nonlinear deviation in compressible regimes. *Europhys Lett*, 1993, 21: 255–259
- 31 Frisch U, Hasslacher B, Pomeau Y. Lattice-gas automata for the Navier-Stokes equation. *Phys Rev Lett*, 1986, 56: 1505–1508
- 32 Chorin A J. A numerical method for solving incompressible viscous flow problems. *J Comput Phys*, 1967, 2: 12–26
- 33 Cook B K. A numerical framework for the direct simulation of solid-fluid systems. Doctor Dissertation. Cambridge: Massachusetts Institute of Technology, 2001
- 34 Fortes A F, Joseph D D, Lundgren T S. Nonlinear mechanics of fluidization of beds of spherical particles. *J Fluid Mech*, 2006, 177: 467–483
- 35 Patankar N A, Singh P, Joseph D D, et al. A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows. *Int J Multiphas Flow*, 2000, 26: 1509–1524
- 36 Wang L P, Rosa B, Gao H, et al. Turbulent collision of inertial particles: point-particle based, hybrid simulations and beyond. *Int J Multiphase Flow*, 2009, 35: 854–867
- 37 Dance S L, Climent E, Maxey M R. Collision barrier effects on the bulk flow in a random suspension. *Phys Fluid*, 2004, 16: 828–831

## Direct numerical simulation scheme for particle-fluid systems based on a time-driven hard-sphere and the lattice Boltzmann method

ZHOU GuoFeng<sup>1,2</sup>, WANG LiMin<sup>1</sup>, WANG XiaoWei<sup>1</sup>, XIONG QinGang<sup>1,2</sup> & GE Wei<sup>1</sup>

<sup>1</sup>State Key Laboratory of Multiphase Complex Systems, Institute of Process Engineering, Chinese Academy of Sciences, Beijing 100190, China;

<sup>2</sup>Graduate University of the Chinese Academy of Sciences, Beijing 100049, China

A direct numerical simulation scheme for particle-fluid systems has been proposed and implemented. The discrete particle phase is described using a time-driven hard-sphere, and the governing equations of the continuous fluid phase are resolved using the lattice Boltzmann method. Particle-fluid coupling is realized using the immersed moving boundary method. This method modifies the collision operator with the force-term dependence of the solid volume fraction of the partially filled fluid cells, which ensures a no-slip boundary condition. The proposed scheme is based on structured grids. This eliminates the need for re-meshing the domain, which is necessary in unstructured grid-based methods. It was found that the proposed scheme successfully reproduces the classical Drafting-Kissing-Tumbling process in the sedimentation of two circular particles in a viscous fluid, and the simulation results are in agreement with previously reported results. Combined, these results validate the proposed scheme.

**direct numerical simulation, time-driven hard-sphere, lattice Boltzmann method, immersed moving boundary method, particle-fluid systems**

doi: 10.1360/972010-1687