arXiv:2304.09079v1 [stat.CO] 17 Apr 2023

# 一种用于拉格朗日随机方法、在三维非结构网格中计算粒子轨迹的时间步长鲁棒算法

**G. Balvet**[*,1,2], **J.-P. Minier**[1,2], **C. Henry**[3], **Y. Roustan**[2], **M. Ferrand**[1,2]

[*] 联系地址：guilhem.balvet@edf.fr[1] EDF R&D，流体力学、能源与环境部，Quai Watier 6号，78400，沙图，法国 [2] CEREA，´法兰西岛国立桥路学校[3] 蔚蓝海岸大学，Inria，法国国家科学研究中心，索菲亚安提波利斯，法国

数字对象标识符：10.1515/mcma-2023-2002

**摘要**

本文旨在提出一种时间步长鲁棒的逐单元积分方法，用于粒子/网格拉格朗日随机方法中三维非结构网格上的粒子轨迹计算。其核心思想是动态更新时间积分中使用的平均场：对每个粒子，将时间步长划分为若干子时间步，使每个子步都对应粒子在单元中的停留时间。这样可以减小空间离散误差。鉴于模型具有随机性，关键在于推导停留时间估计值时不预知维纳过程的未来。为此，新算法为每个随机粒子配备一个虚拟粒子，其平均条件行为可提供无统计偏差的停留时间预测。经过一致性检查后，本文通过两个具有代表性的测试案例验证了该新算法：统计均匀流中的粒子扩散，以及非均匀流中的粒子动力学。

MSC 2010 class: 76M35

**关键词：拉格朗日随机建模、粒子-网格概率密度函数、时间积分、三维非结构网格中的轨迹、时间分裂方法、预见误差**

# 1 引言

顾名思义，拉格朗日概率密度函数（PDF）方法是对一组变量的概率密度函数进行模拟，这些变量经过筛选，用于描述单相流或分散两相流的相关信息。由拉格朗日概率密度函数可直接推导出欧拉概率密度函数，从而获得所关注的统计矩。在仅关注统计量的弱形式下，拉格朗日概率密度函数方法等价于粒子随机模型。因此，这些方法表现为针对一组附着于每个粒子的变量所建立的随机方程，这组变量构成粒子状态向量（通常包括粒子位置、速度及其他所关注的变量）。这些方法在湍流反应性含颗粒流的建模与模拟方面具有显著优势。具体而言，当局部源项以附着于每个粒子的变量的已知表达式给出时（前提是这些变量包含在粒子状态向量中），无论局部源项多么复杂或非线性，这些方法都能在不作近似的情况下处理它们。因此，概率密度函数方法能够解决与非线性局部项的平均值相关的闭合问题，例如化学源项，或粒子运动方程中的阻力。同时，拉格朗日视角确保了对流得到准确处理。这使得这些方法非常适合模拟多分散非均匀两相流以及单相反应流，同时也可作为惰性单相湍流的有效模型。

拉格朗日概率密度函数模型于20世纪80年代和20世纪90年代开始发展，主要由Pope及其合作者 [11, 33, 31, 34, 35]完成，后来由Minier及其合作者 [24, 37, 21, 23, 16, 17, 18]将其扩展到弥散型两相流情形。多年来，人们持续致力于分析随机模型 [38, 16, 17] ，并提出有助于构建这些模型的实用指南 [20]。对于弥散型湍流两相流中所见流体速度的随机模型而言，情况尤其如此；这是一个仍在持续研究的课题 [13]（另请参见 [18]中近期提出的一般方法）。与此同时，随着Pope及多位合作者 [11, 34, 32, 10, 14]对广义朗之万模型（GLM）的构建，单相湍流方面的研究似乎已进入较为成熟的阶段。在数值研究方面，也有专门工作致力于分析与数值实现相关的各种问题，而数值实现通常是在粒子/网格方法的背景下进行的 [42, 29]。在此阶段，值得提醒的是，许多此类发展都是在研究代码中完成的，主要着眼于验证新的随机模型，因此使用的是相对简单几何形状中的结构化网格。据作者所知，对于复杂真实域中的粒子追踪方法，以及这些方法与适当时间积分格式结合使用的问题，关注相对较少（例如用于环境用途）。

为积分描述粒子动力学的随机微分方程（SDE）， [29, 22]中提出了一种专门的时间积分数值格式。该格式利用粒子速度方程中存在的弛豫项，首先考虑包含指数项的积分形式，然后对这些积分表达式进行离散化，而不是直接离散随机微分方程。因此，这种格式被称为指数数值格式。需要注意的是，由于该格式假设存在弛豫项，因此并不适用于任意随机微分方程系统；不过，它对于湍流中粒子动力学的许多模型方程仍具有重要价值。这种指数格式具有一些有趣的性质：按照构造，当所有平均场和时间尺度均为常数时（即随机微分方程为线性方程时），它对应于精确解；同时，它保持显式且无条件稳定（这得益于其公式中的指数因子）。此外，对随机积分进行仔细离散化后，即使弛豫项中一个或多个时间尺度远小于时间步长，也能正确再现相应的渐近状态。一个具有典型实际意义的例子是，该格式能够准确捕捉扩散区间，无需切换到其他数值格式。关于这种指数格式的深入描述与分析，可参见一篇全面的综述；有兴趣的读者可查阅相关文献以了解详情 [29]。

时间积分指数格式的无条件稳定性，加上其能够捕捉正确渐近极限的特性，使得实际模拟中可以采用大时间步。然而，在这种情况下，粒子通常会跨越一段距离，在此期间，其模型动力学方程中涉及的平均场（例如流体平均速度场）可能发生显著变化。然而，秉承欧拉格式的思路，时间一阶指数格式仅保留粒子初始位置处这些平均场的取值。值得注意的是，即使是基于预测校正步骤扩展的二阶格式，也存在同样的局限性：它仅保留平均场的两个取值（分别位于粒子初始位置和预测位置处）。当这些平均场的梯度较大时，这会导致空间

离散误差，从而降低数值格式的整体精度，以及从颗粒动力学模拟中提取的相关统计量的精度。因此，需要开发新方法，将粒子跟踪与时间积分数值格式结合起来，以适用于复杂几何结构和非均匀流。

在此背景下，本文旨在开发一种新的颗粒追踪算法，以降低指数格式的空间离散误差，尤其是针对粒子可能跨越多个单元的大时间步。其基本思想是引入与粒子在所经过各单元中的停留时间相对应的子时间步，执行逐单元积分。这样便可在每个时间步内沿粒子轨迹动态更新平均场和局部时间尺度。从本质上说，如果采用 $P_0$ 插值（其中平均场在单元内被视为常数），这种新的时间积分方法旨在使指数数值格式获得精确解。然而，尽管采用附着于颗粒的局部子时间步这一思路看似相当直接，并且可直接应用于每条粒子轨迹，但由于用于构建粒子轨迹的动力学模型具有随机性，估计粒子在给定单元内的停留时间实际上是一个棘手问题。停留时间的估计确实必须遵循伊藤对随机积分的定义。这对应于非预见性要求，而本文详述的新算法满足该要求。需要注意的是，逐单元积分的思想已由 [36] 提出，但此前仅在结构化网格上给定的规定速度场中进行了测试（因此未涉及显式速度随机模型）。因此，本研究可以视为在随机模型背景下（即处理非预见性要求）以及针对非结构化网格对该工作的扩展。

本文结构安排如下。第2节首先回顾拉格朗日概率密度函数（或粒子随机）模型的理论背景，其中介绍流体粒子的极限情形以及所采用的特定朗之万模型。第3节概述当前的数值格式，以揭示空间离散误差。随后在第4节详细介绍新算法，并在第5节通过两个测试案例进行验证。最后在第6节得出结论。

# 2 理论　　湍流随机模型的理论背景

## 2.1 湍流的 PDF模型

弥散型湍流两相流包含离散单元（即“粒子”），它们由湍流流体流动输运。为简化起见，本文聚焦于粒子动力学，不考虑热效应、粒子半径或质量的变化以及粒子碰撞效应。对此类系统建模时，一种合适的建模框架是采用拉格朗日随机观点。在这种方法中，建立一组随机微分方程（SDEs），用以描述构成颗粒状态向量 $\mathbf{Z}_p$ 的粒子附着变量随时间的演化。这里，状态向量取为 $\mathbf{Z}_p = (\mathbf{X}_p, \mathbf{U}_p, \mathbf{U}_s)$，其中 $\mathbf{X}_p$ 为粒子位置，$\mathbf{U}_p$ 为其速度，$\mathbf{U}_s$ 为所见流体速度。该组随机微分方程为

$$\mathrm{d}X_{p,i} = U_{p,i}\,\mathrm{d}t\ , \tag{1a}$$

$$\mathrm{d}U_{p,i} = \frac{U_{s,i} - U_{p,i}}{\tau_p}\,\mathrm{d}t + A_{p,i}\,\mathrm{d}t\ , \tag{1b}$$

$$\mathrm{d}U_{s,i} = (\text{stochastic model})\ . \tag{1c}$$

在公式 **(1)** 中，所见流体速度 $\mathbf{U}_s$ 定义为在粒子位置处采样的瞬时流体速度，$\mathbf{U}_s(t) = \mathbf{U}(t, \mathbf{X}_p(t))$，其中 $\mathbf{U}(t, \mathbf{x})$ 是流体速度场。需要注意的是，对于非完全解析的湍流中的离散粒子而言，无法根据流体速度场所提供的有限信息（通常为其一阶和二阶矩）获得所见流体速度。这意味着 $\mathbf{U}_s$ 需要作为一个独立的附着于粒子的变量引入（参见 [21, 16, 17] 中的详细讨论）。在公式 **(1b)** 的右端（RHS）可以看到，$\mathbf{U}_s$ 出现在阻力的表达式中，而项 $\mathbf{A}_p$ 包含作用于离散粒子的可能附加力（例如重力）。该阻力用粒子弛豫时间尺度 $\tau_p$ 表示，其定义为

$$\tau_p = \frac{\rho_p}{\rho}\frac{4\,d_p}{3\,C_D\,|\,\mathbf{U}_s - \mathbf{U}_p\,|}\ , \tag{2}$$

其中，$\rho$ 和 $\rho_p$ 分别为流体和粒子密度，$d_p$ 为粒子直径，$C_D$ 为阻力系数。该时间尺度表示粒子速度调节至所见局部流体速度所需的典型时间

并且是粒子惯性的度量。在斯托克斯区间内，当 $\mathrm{Re}_p \leq 1$（其中，$\mathrm{Re}_p$ 为颗粒雷诺数，定义为 $\mathrm{Re}_p = |\mathbf{U}_r|\, d_p/\nu$，其中 $\mathbf{U}_r = \mathbf{U}_s - \mathbf{U}_p$，$\nu$ 为流体运动分子黏度）时，该条件有效，阻力系数为 $C_D = 24/\mathrm{Re}_p$。在这种情况下，粒子弛豫时间尺度由斯托克斯公式给出：

$$\tau_p = \frac{\rho_p}{\rho}\frac{d_p^2}{18\nu_f}. \tag{3}$$

对于 $\mathrm{Re}_p$ 的一般取值，阻力系数通过经验关联式获得，例如 [5, 3]

$$C_D = \begin{cases} \dfrac{24}{\mathrm{Re}_p}\left[1 + 0.15\mathrm{Re}_p^{0.687}\right] & \text{if } \mathrm{Re}_p \leq 1000, \\ 0.44 & \text{if } \mathrm{Re}_p \geq 1000. \end{cases} \tag{4}$$

值得回顾的是，由类似式（1）这样的方程建模的随机颗粒，必须被视为相应概率密度函数的样本，因此，这种描述也可称为两相概率密度函数模型 [21, 17, 20]。在当前的概率密度函数框架中，我们基本上处理的是一个双时间单粒子拉格朗日概率密度函数，由此推导出单时间单点欧拉概率密度函数，从而可以直接提取矩方程 [21, 17, 32]。这意味着，一旦写出随机模型（如式（1）中的模型），我们就可以直接得到相应的离散拉格朗日 $F^L_{p,N}$以及由此得到的欧拉 $F^E_{p,N}$ 质量密度函数（MDF），其形式为

$$F^L_{p,N}(t\,;\,\mathbf{Y}_p, \mathbf{V}_p, \mathbf{V}_s) = \sum_{i=1}^{N} m_p^{(i)}\, \delta(\mathbf{Y}_p - \mathbf{X}_p^{(i)}(t))\, \delta(\mathbf{V}_p - \mathbf{U}_p^{(i)}(t))\, \delta(\mathbf{V}_s - \mathbf{U}_s^{(i)}(t))\ , \tag{5}$$

$$F^E_{p,N}(t, \mathbf{x}\,;\,\mathbf{V}_p, \mathbf{V}_s) = F^L_{p,N}(t\,;\,\mathbf{Y}_p = \mathbf{x}, \mathbf{V}_p, \mathbf{V}_s)\ , \tag{6}$$

其中，$m_p$ 为颗粒质量，$N$ 为定义域中的随机颗粒数量，而$\mathbf{Y}_p$、$\mathbf{V}_p$ 和$\mathbf{V}_s$ 分别是与随机变量$\mathbf{X}_p(t)$、$\mathbf{U}_p(t)$和$\mathbf{U}_s(t)$相对应的样本空间值。随后，可以直接推导出任意颗粒变量 $H_p(t; \mathbf{V}_p, \mathbf{V}_s)$ 的平均值 $\langle H_p\rangle(t,\mathbf{x})$ 的场值，

$$\alpha_p(t,\mathbf{x})\, \rho_p \langle H_p\rangle(t,\mathbf{x}) = \int H_p(t; \mathbf{V}_p, \mathbf{V}_s) F_p^E(t, \mathbf{x}; \mathbf{V}_p, \mathbf{V}_s)\, d\mathbf{V}_p\, d\mathbf{V}_s\ , \tag{7}$$

其中，$\alpha_p(t,\ \mathbf{x})$ 是平均颗粒体积分数，通过适当的概率归一化约束 [21, 30] 定义。在数值模拟中，这些理论表达式意味着，在给定位置 $\mathbf{x}$ 周围的一个小体积内，平均值是通过对该体积内存在的 $N^p_{\mathbf{x}}$ 粒子进行系综平均，或进行 Favre平均（或质量加权平均）来估算的

$$\langle H_p\rangle(t,\mathbf{x}) \simeq \frac{\displaystyle\sum_{i=1}^{N^p_{\mathbf{x}}} m_p^{(i)} H_p(t; \mathbf{V}_p^{(i)}(t), \mathbf{V}_s^{(i)}(t))}{\displaystyle\sum_{i=1}^{N^p_{\mathbf{x}}} m_p^{(i)}}\ . \tag{8}$$

出于其他地方 [21], 所述的原因，流体所见的速度通常用朗之万模型表示，其形式如下（详见 [21, 23, 16, 20] 中的论述）

$$\mathrm{d}U_{s,i} = -\frac{1}{\rho}\frac{\partial\langle P\rangle}{\partial x_i}\,\mathrm{d}t + \left(\langle U_{p,j}\rangle - \langle U_j\rangle\right)\frac{\partial\langle U_i\rangle}{\partial x_j}\mathrm{d}t + G^*_{ij}\left(U_{s,j} - \langle U_j\rangle\right)\mathrm{d}t + B^*_{ij}\,\mathrm{d}W_j\ . \tag{9}$$

其中，$P$ 是流体压力，$\mathbf{dW}$ 是维纳过程的增量（更多详情见第 2.2 节）。在式 (9) 中，用于封闭矩阵 $G^*_{ij}$ 和 $B^*_{ij}$ 的表达式可能相当复杂，完整公式请参阅相关文献 [21, 16]。正如这些文献所述，$\mathbf{U}_s$ 的随机模型是在用于模拟流体粒子的模型基础上扩展构建的。从式 (1) 中的一般弥散两相模型出发，通过令颗粒时间尺度 $\tau_p$ 趋于零，即可得到流体极限情形（也称为示踪颗粒极限）。

在本研究中，我们主要关注复杂几何结构中一般非均匀流的粒子跟踪，因此重点放在粒子位置 $\mathbf{X}_p$ 上。为了突出第4节所介绍的新算法的基本要素，所采用的用于模拟粒子速度的随机模型选择为

尽可能简单，以避免在后续推导中处理繁琐的表达式。基于这些原因，从现在起，我们仅考虑湍流中流体粒子的极限情形，并选择最简单的朗之万模型来描述这些流体粒子的速度。然而，本节回顾的一般背景旨在表明，新算法确实适用于比本文其余部分所采用的随机公式大得多的一类随机公式。

**单相湍流的 PDF模型** 如前所述，流体粒子的动力学基于最简单的朗之万模型，即简化朗之万模型（SLM）[32, 9]。粒子状态向量现在为 $\mathbf{Z} = (\mathbf{X},\mathbf{U})$，其演化方程写为

$$\mathrm{d}X_i = U_i \mathrm{d}t, \tag{10a}$$

$$\mathrm{d}U_i = -\frac{1}{\rho}\frac{\partial \langle P\rangle}{\partial x_i}(t,\mathbf{X}(t))\,\mathrm{d}t - \frac{U_i - \langle U_i\rangle(t,\mathbf{X}(t))}{T_L(t,\mathbf{X}(t))}\mathrm{d}t + \sqrt{C_0\epsilon(t,\mathbf{X}(t))}\,\mathrm{d}W_i\ . \tag{10b}$$

公式（10b）的右侧涉及多个平均场值，包括湍流耗散率 $\epsilon$、流体湍动能 $k$ 和拉格朗日积分时间尺度 $T_L$。所有这些平均场都应在粒子附近的局部位置进行评估，因此明确写出了这种局部依赖性。拉格朗日积分时间尺度定义为:

$$T_L = \left(\frac{1}{\frac{1}{2}+\frac{3}{4}C_0}\right)\frac{k}{\epsilon}\ , \tag{11}$$

其中 $C_0$ 为所谓的柯尔莫哥洛夫常数。随后可以按照上文概述的 PDF形式主义进行推导，结果表明，就流体速度的统计量而言，SLM模型直接对应于一种二阶湍流模型，即 Rotta模型 [35, 32]。此外，我们只考虑不可压缩流。因此，由于我们为每个流体粒子分配相同的质量 $\delta m$ ，这意味着粒子浓度必须保持均匀，以满足不可压缩条件。

## 2.2 随机积分提示

本文仍将流体粒子的 SLM 作为参考模型，尽管如上所述，第4节中将要介绍的数值推导可以通过采用 [29] 中提出的扩展数值积分格式，扩展到分散型湍流两相流的情况。当前的研究思路也适用于更一般的随机模型，在这些模型中，粒子状态向量不仅包括粒子位置和速度，还包括附加变量 $\mathbf{Z} = (\mathbf{X},\ \mathbf{U},\ \mathbf{Z}')$，其中 $\mathbf{Z}'$ 表示附加到每个随机粒子上的附加变量，用于处理更复杂的物理问题（在本文其余部分中，我们令 $\mathbf{Z}' = \varnothing$）。此外，所建模的粒子速度方程也可以采用不同于 Langevin 模型的其他随机模型，但我们假设其形式为一般的随机扩散过程，这意味着包含白噪声项（请注意，在这种情况下，时间积分所采用的数值格式必须进行相应调整，参见第4.2.1节）。

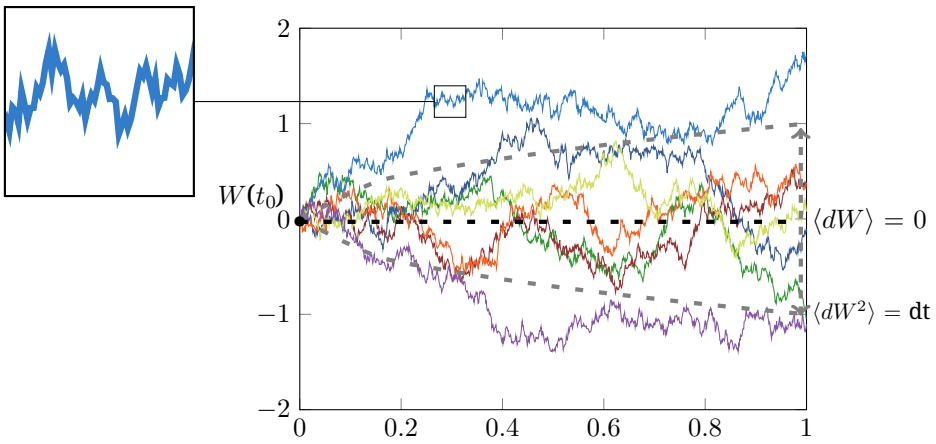
不过，有必要简要回顾一下随机积分的定义以及非预见性公式这一重要问题。为此，我们考虑一个粒子状态向量 $\mathbf{Z} = (Z_i)$ 的一般随机扩散过程，其形式为

$$\mathrm{d}Z_i = \underbrace{A_i\left(t,\mathbf{Z},\langle\mathcal{F}(t,\mathbf{Z})\rangle\right)\mathrm{d}t}_{\text{deterministic Part}} + \underbrace{B_{ij}\left(t,\mathbf{Z},\langle\mathcal{G}(t,\mathbf{Z})\rangle\right)\mathrm{d}W_j}_{\text{random Part}}\ . \tag{12}$$

在公式（12）中，$\mathbf{A} = (A_i)$ 和 $\mathbf{B} = (B_{ij})$ 分别表示漂移向量和扩散矩阵。它们采用尽可能一般的形式，其中包含对均值的可能依赖，这种依赖由两个泛函项表示，即 $\mathcal{F}$ 和 $\mathcal{G}$，因此公式（12）构成了 McKean型随机扩散过程（见 [16, 17]）。维纳过程 $\mathbf{W}$ 是一个连续时间马尔可夫过程，其增量服从方差为 $\mathrm{d}t$ [28, 38] 的独立高斯中心分布。尽管其轨迹是连续的，但在任意区间内都极为不规则，甚至具有无限总方差（见图1中的示意图），这意味着不能应用经典积分规则。

公式（12）中的随机微分方程表述实际上是恰当数学表达式的一种简写，即其积分形式

$$Z_i(t) = Z_i(t_0) + \int_{t_0}^{t} A_i(s,\mathbf{Z}(s),\mathcal{F}(s,\mathbf{Z}(s)))\,\mathrm{d}s + \underbrace{\int_{t_0}^{t} B_{ij}(s,\mathbf{Z}(s),\mathcal{G}(s,\mathbf{Z}(s)))\,\mathrm{d}W_j(s)}_{\text{stochastic integral}}\ , \tag{13}$$




Fig图 1：条件维纳过程的一些实现，从时间 $t = 0$ 时的 0 开始 .

其中，必须对随机积分进行仔细定义。为此，需采用 Itˆo 对非预见过程的定义。使用区间 $[t_0; t]$ 的划分 $[t_k; t_{k+1}]$, $k = 1, \ldots, N$，即可得到 Itˆo 意义下的随机积分：

$$\int_{t0}^{t} B_{ij}(s, \mathbf{Z}(s), \mathcal{F}(s, \mathbf{Z}(s)))\, \mathrm{d}W_s = \text{ms-}\lim_{N\to\infty} \sum_{k=1}^{N} B_{ij}(t_k, \mathbf{Z}(t_k), \mathcal{F}(t_k, \mathbf{Z}(t_k)))\, (W_j(t_{k+1}) - W_j(t_k)) \ , \qquad (14)$$

其中，极限必须理解为均方意义下的极限（因为不可能逐轨迹收敛）。该定义的选择使随机积分的两个基本性质对于任意足够光滑的函数 $\Psi_1$ 和 $\Psi_2$ 始终成立（此处省略向量索引和函数依赖关系，以保持记号简洁）：

- $\left\langle \int_{t_0}^{t} \Psi_1(\mathbf{Z}(s)) \mathrm{d}W(s) \right\rangle = 0 \ ,$
- $\left\langle \left( \int_{t_0}^{t_2} \Psi_1(\mathbf{Z}(u)) \mathrm{d}W(u) \right) \left( \int_{t_1}^{t_3} \Psi_2(\mathbf{Z}(v)) \mathrm{d}W(v) \right) \right\rangle = \int_{t_1}^{t2} \langle \Psi_1(\mathbf{Z}(s)) \Psi_2(\mathbf{Z}(s)) \rangle \mathrm{d}s \ , \ t_0 \le t_1 \le t_2 \le t_3 \ .$

随机积分定义的非预见性所带来的一个关键结果是，公式（12）中随机过程 $\mathbf{Z}$ 在一个小时间增量 $\Delta t$ 内的条件均值增量仅由漂移项决定，这意味着我们有

$$\langle \Delta Z_i \,|\, \mathbf{Z}(t_0) = \mathbf{z}_0 \rangle = A_i(t_0, \mathbf{z}_0, \mathcal{F}(t_0, \mathbf{z}_0))\, \Delta t \ . \qquad (15)$$

These relations play a central role in the development of the new algorithm in Section 4.

## 3 Current numerical algorithm

To obtain numerical solutions of the SDEs in Eqs. (10), three issues need to be addressed:

1) How do we calculate the mean fields at particle positions?

2) How do we integrate in time the SDEs?

3) How do we extract statistics from the ensemble of particles?

这三个彼此独立的方面分别对应三种误差来源：第一种是由于在粒子位置处对平均场进行近似表达而产生的空间离散误差；第二种是由于所选积分方案在离散时间更新粒子变量而产生的时间误差；第三种则与蒙特卡罗方法固有的统计误差有关。在一个

完整的数值 formulation 中，这些误差可能相互影响，甚至会诱发第四种误差，即偏差误差。然而，出于下文所述的原因，在当前语境中并不存在这种偏差误差，因此我们可以放心地集中讨论上述三种误差。

尽管专门针对这些问题开展的数值分析研究相对较少，但至少有两项详细研究对其进行了深入分析 [42, 29], ，读者可参阅这些研究以获取更多细节。在本节中，我们仅回顾当前算法的关键方面，并着重讨论空间离散误差，从而为第4节所述的发展奠定基础。

## 3.1 混合FV/PDF方法与粒子-网格实现

对于流体粒子，一种初步方法是直接从粒子模拟中提取公式（10）中的平均场，然后将其反馈至计算中。这可以通过使用辅助网格，根据粒子系综计算这些统计量来实现，也可以采用以每个粒子位置为中心的局部核估计。这类数值公式称为PDF独立代码，其优点之一是从构造上具有一致性；另一方面，由于使用有限数量的粒子而产生的固有统计噪声可能会导致确定性偏差误差 [42, 29]。然而，在本研究中，我们选择采用一种混合公式，将粒子模拟与经典FV（有限体积）求解器耦合，从而形成所谓的混合FV/PDF方法。这类公式的一个优点是，提供给粒子求解器的平均场不再含有统计噪声，因此可以避免偏差误差。另一方面，对于流体粒子而言，这种混合公式会立即引发一致性问题，因为例如，平均流体速度场被模拟了两次（一次在FV求解器中，另一次则作为位于给定小体积内的流体粒子的平均值）。正如多项研究 [16, 4], 所讨论的，随后将对应于同一湍流模型的FV求解器与粒子求解器进行耦合就十分重要。例如，对于公式（10）中采用的当前简单朗之万模型，FV求解器中的湍流模型必须是Rotta模型，其常数应与公式（10b）中 $C_0$ 的取值保持一致。这些问题已在多篇论文 [35, 16, 32, 4] 中进行了详细讨论，本文不再重复。选择这类混合公式的另一个原因是，正如第2节所指出的，我们有意将当前研究扩展至离散粒子的情形；在这种情形下，例如无法从粒子模拟中提取流体平均速度场（所见流体速度的统计特性不同于流体速度的统计特性 [21, 16, 17]），因此这类混合公式具有特殊意义。需要注意的是，尽管本文未涉及这一点，但在粒子求解器中引入底层网格也有助于模拟粒子碰撞效应 [39, 40]。

实际上，如图2所示，混合FV/PDF耦合公式被转化为粒子/网格模拟；在每次迭代中，首先运行FV求解器，然后将所需的平均场提供给网格上的粒子求解器。这类混合公式的难点在于，FV求解器中的平均场是在确定的空间位置处已知的（对于同位离散，通常位于网格中每个单元的中心）。然而，粒子求解器中的流体示踪粒子连续分布在定义域内，其位置极少与单元中心位置重合。因此，我们还需要解决一个问题：如何将网格上的平均场值插值到粒子位置。在本研究中，我们采用了最简单的方案，即将相同的单元中心平均值赋给位于该单元内的所有粒子：这相当于在粒子位置进行 $P_0$ 插值。尽管这意味着粒子穿过两个相邻单元之间的界面时，平均量的分布会不连续，但数值实现简单这一优点弥补了这一不足。尤其是，即便对于复杂几何结构中的非结构化网格（有时由多个重叠网格构成），该方法仍然有效且易于应用；在这类情况下，线性插值等更先进的方法（例如在相邻单元中心之间进行线性插值）很快就会变得极其复杂，几乎难以实现。

Based on this hybrid formulation and using the $P_0$ interpolation assumption, the particle equations, cf Eqs. (10), can be expressed as:

$$\mathrm{d}X_i = U_i \,\mathrm{d}t, \tag{16a}$$

$$\mathrm{d}U_i = -\frac{1}{\rho}\left[\frac{\partial\langle P\rangle}{\partial x_i}\right]_0 (t,\mathbf{X}(t))\mathrm{d}t - \frac{U_i - [\langle U_i\rangle]_0(t,\mathbf{X}(t))}{[T_L]_0(t,\mathbf{X}(t))}\mathrm{d}t + \sqrt{C_0\,[\epsilon]_0(t,\mathbf{X}(t))}\mathrm{d}W_i\ , \tag{16b}$$

其中，$[.]_0$ 表示 $P_0$ 插值，为简洁起见，后文将省略。在这种混合有限体积（FV）/概率密度函数（PDF）表述中，使用同一网格从特定时刻位于给定单元内的粒子子集中提取粒子统计量。这对应于最近网格点（NGP）方法，其中

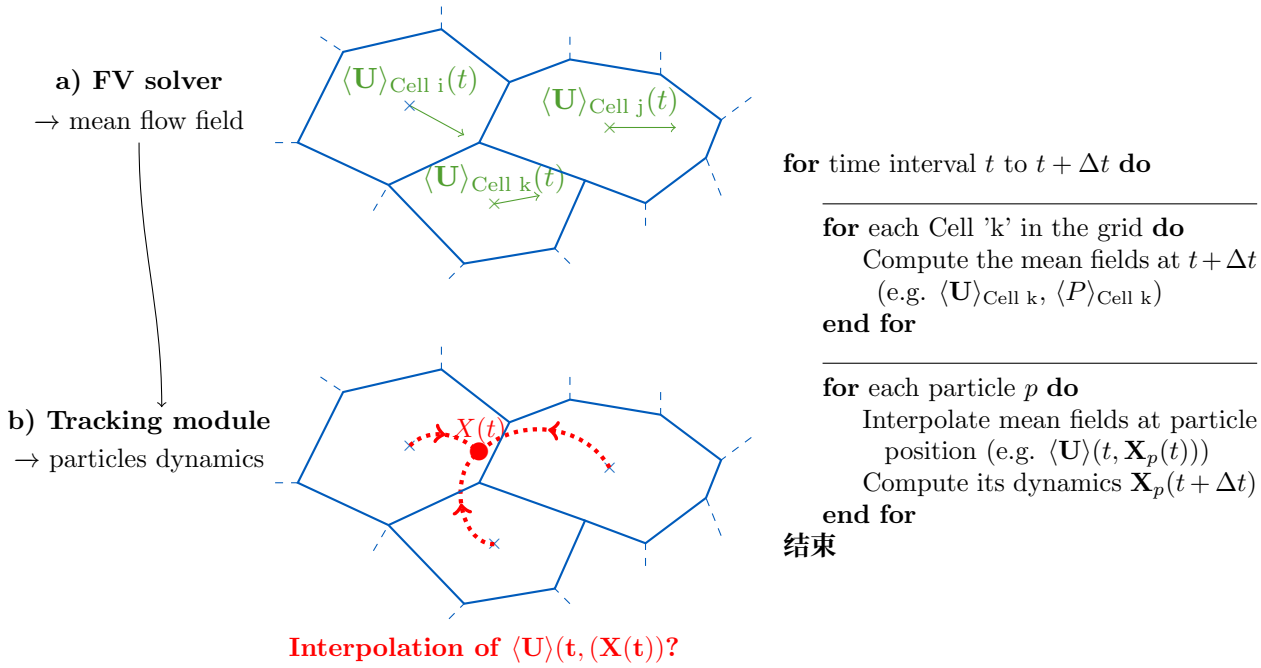



图2：混合FV/PDF方法原理示意图：对于每个时间步长，首先运行FV求解器，以在网格上提供所需的平均场；随后，采用PDF方法求解粒子动力学，这需要使用插值方法来确定粒子位置处的平均场值。

与每个粒子相关联的全部统计权重都归属于其所在的单元，而不考虑粒子在该单元中的位置。实际上，这种NGP方法与用于将网格信息传递给粒子的 $P_0$ 插值相一致，可视为其逆向操作，即从粒子传递信息到网格（参见 [42, 29]中的讨论）。

在此阶段，有必要针对在粒子位置处插值平均场这一问题作几点说明。在采用粒子-网格方法实现的独立PDF模型中，插值格式与粒子统计量的提取方式表现为两个伴随算子 [42, 29]。这意味着，例如，如果使用线性插值格式获得插值后的平均场值，则必须选择单元内云（CIC）方法，以从粒子集合中推导统计量。在结构化网格中，这对应于一种经典方法 [12]。然而，在非结构化网格中，实现这种一致的方法很快就会变得复杂。在现有的流体粒子混合FV/PDF公式中，由于粒子统计量不会被注入粒子动力学的随机模型，这两个算子是解耦的。因此，人们很容易考虑采用比 $P_0$更精确的插值格式，因为这可能有助于减小总体空间离散误差。然而，在非结构化网格中，实现这类格式仍然很棘手。此外，只有当高阶插值能够应用于粒子在一个时间步长内穿过的不同单元时，采用高阶插值才基本上具有益处。从这个意义上说，插值问题似乎与单元间积分格式的开发互为补充。下文将重点构建这样一种单元间积分方法。因此，为了简化处理，我们选择保留最简单的情况，即 $P_0$ 插值格式。

Therefore, questions (1) and (3) have been addressed and we can turn our attention to the question (2) and how time integration of the SDEs is performed.

### 3.2 Current numerical scheme for particle transport

确定粒子位置处的平均场后，下一步是设计一种数值格式，用于预测构成粒子状态向量 $\mathbf{Z} = (\mathbf{X}, \mathbf{U})$ 的流体粒子属性。这通常会引入时间步长 $\Delta t$ 的概念，用于对粒子运动方程进行积分，并对时刻 $t^n = n\Delta t$的 $\mathbf{Z}$ 进行离散近似，记为 $\mathbf{Z}^n$。这些离散近似 $\mathbf{Z}^n$ 通过以下方式获得

连续更新（即，根据 $\mathbf{Z}^n$ 计算 $\mathbf{Z}^{n+1}$ ），使用方程（16）右侧的积分形式。在当前混合方法的框架下，当前算法包含两个步骤:

- 一种时间积分格式，可利用粒子位置处进入公式（16）中随机微分方程的平均场已知值，获得$\mathbf{Z}^{n+1}$ 由 $\mathbf{Z}^n$ ；
- 一种轨迹算法，用于确定空间域中每个粒子的位置。混合FV/PDF方法需要这第二个步骤，因为必须根据用于流体相计算的网格，确定每个粒子所在的单元。为了在时间积分过程的下一步更新各粒子位置处平均场的值，这些信息是必需的。

I下面简要回顾当前算法在各步骤中的具体内容 .

### 3.2.1 用于预测粒子状态向量的时间积分格式

用于对随机微分方程进行积分的数值格式（参见公式（16））已在先前的论文中加以描述 [29, 22]„ 这些论文全面介绍了其推导过程和主要特性（尤其参见 [29]中对构建该格式所涉及不同步骤的完整描述）。因此，这里仅回顾其关键方面以及由此得到的公式。

**要求**时间积分格式根据以下准则开发（更多详情见 [29, 22]）：

- 数值格式为显式格式（出于简便性考虑）；
- 数值格式无条件稳定（当时间步长无法减小时，这一点尤为重要，例如在混合方法中，为满足潜在的稳定性判据）；
- 当进入所建方程的平均场和时间尺度保持恒定时，数值格式对应于精确解;
- 当时间步长远大于物理时间尺度 [29] 时，数值格式应在极限情况下捕捉正确的物理行为。

这些要求背后的主要原因是，混合 FV-PDF 公式中使用的时间步长通常对流体求解器和粒子求解器是相同的。对于非定常流动，这是必须的，因为必须在同一时刻获得流体和粒子的属性。因此，由于时间步长通常由流体求解器设定（以便在给定网格上正确计算流场），它并不总能针对粒子求解器进行减小或调整（而这在独立 PDF 方法中是可以做到的）。在平均流体速度可能因区域不同而发生剧烈变化的复杂流动中，这意味着我们无法控制粒子在一个时间步长内穿过的单元数量。此外，当流体时间尺度也存在较大差异时，正确捕捉扩散极限十分重要，而不应因此限制整个粒子集合的时间步长 [29]。

**所选格式** 当前时间积分格式的基本原则是，首先写出随机微分方程的积分形式（假设各项性质恒定），由此可直接得到指数形式的解析解。在一般情况下，平均场和时间尺度并非常量，而是在空间和时间上变化，此时的思路是在积分形式中将它们的值冻结在时间步长开始时，而不是在随机微分方程中采用欧拉类格式（详细说明见 [29]）。由于采用了指数因子形式，因此能够自动保证无条件稳定性。尽管采用相同方法也完全可以构造二阶格式（见 [29]），但以下我们仅讨论一阶形式，因为我们主要关注的是此类格式中相关平均场的预测

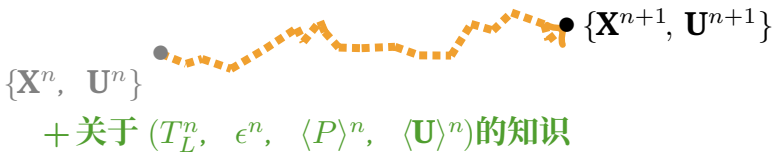



图 3：示意图说明了采用当前数值格式进行时间积分的过程：在下一次迭代中，粒子位置$X_i^{n+1}$ 和速度 $U_i^{n+1}$ 根据上一次迭代时粒子的位置和速度 $n$ 以及位置 $X_i^n$处的流体特性计算得出。

因此，得到的流体粒子离散数值格式为：

$$X_i^{n+1} = X_i^n + U_i^n T_L^n \left(1 - \exp\left(-\frac{\Delta t}{T_L^n}\right)\right) + C_i^n T_L^n \left(\Delta t - T_L^n \left(1 - \exp\left(-\frac{\Delta t}{T_L^n}\right)\right)\right) + \underbrace{\sqrt{\frac{C_0 \epsilon^n (T_L^n)^3}{2}} \frac{\left(1 - \exp\left(-\frac{\Delta t}{T_L^n}\right)\right)^2}{\sqrt{1 - \exp\left(-2\frac{\Delta t}{T_L^n}\right)}} \zeta_i^U + \sqrt{C_0 \epsilon^n (T_L^n)^2 \left(\Delta t - 2T_L^n \frac{1 - \exp\left(-\frac{\Delta t}{T_L^n}\right)}{1 + \exp\left(-\frac{\Delta t}{T_L^n}\right)}\right)} \zeta_i^X}_{I_i^X(\Delta t)}, \tag{17a}$$

$$U_i^{n+1} = U_i^n \exp\left(-\frac{\Delta t}{T_L^n}\right) + C_i^n T_L^n \left(1 - \exp\left(-\frac{\Delta t}{T_L^n}\right)\right) + \underbrace{\sqrt{C_0 \epsilon^n \frac{T_L^n}{2} \left(1 - \exp\left(-2\frac{\Delta t}{T_L^n}\right)\right)} \zeta_i^U}_{I_i^U(\Delta t)}. \tag{17b}$$

这两个项 $\zeta_i^X$和 $\zeta_i^U$对应于从归一化高斯分布 $\mathcal{N}(0,\ 1)$中抽取的独立随机数。如式（17）所示，这些随机数参与两个相关随机积分的乔列斯基分解，即 $I_i^U(\Delta t)$ 和 $I_i^X(\Delta t)$（说明见 [29]）。

如图 3 所示，时间积分格式利用时间步长开始时的数值（$\mathbf{X}^n,\ \mathbf{U}^n$）以及位置 $\mathbf{X}^n$处的局部流体特性，预测下一时间步长的粒子状态向量（$\mathbf{X}^{n+1},\ \mathbf{U}^{n+1}$）。

### 3.2.2 用于空间位置确定的轨迹算法

时间积分格式需要了解给定时刻粒子位置处的流体平均场（例如流体速度、湍流耗散率），这些平均场已在给定网格上定义。因此，时间积分格式必须辅以轨迹算法，以确定每个粒子的位置并为其分配相应的单元。

**假设** 轨迹算法必须满足以下要求：

- 首先，由于采用NGP方法，轨迹算法应能够确定每个粒子所在的单元。

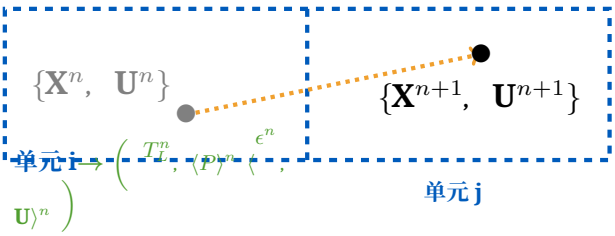
- 其次，轨迹算法必须适用于通用三维非结构网格，其中各单元均围绕其中心呈星形分布。之所以作出这一选择，是因为计算平均流场时通常使用此类网格。

- 第三，轨迹算法假设粒子在每个时间步长内进行自由飞行运动。这意味着，在每个时间步长内，粒子沿直线从其初始位置 $X_i^n$移动到最终位置 $X_i^{n+1}$ 。

**所选算法** 针对前述要求，我们采用了一种基于连续邻居搜索的算法。此类跟踪算法仅利用粒子的初始位置和最终位置来确定粒子所在的新单元（另见图4）。随后，通过搜索轨迹向量（连接初始位置和最终位置）与当前单元各面的交点，确定新单元。为此，我们采用M¨oller-Trumblemore算法 [26] ，该算法用于确定向量与三角形子面之间的交点。然后重复这一过程，以检查最终位置是否位于新单元内，或轨迹向量是否穿过了另一个面。附录A介绍了该跟踪算法的详细信息和验证结果。

保留了此类跟踪算法，因为已有研究表明，它们对于非结构化网格十分高效（例如， [15] 中的有限差分方法或 [27, 41]中的有限体积方法）。此外，对于非结构化网格而言，这些跟踪算法比简单的定位算法更为合适（后者仅利用最终位置的信息来确定位置）。

此时，值得一提的是，对于追踪算法过程中穿过的每个面，都可以考虑附加条件。特别是，当穿过的面对应于物理边界时，可以添加特定的边界条件：例如，当粒子到达出口边界时，可以将其从模拟中移除，而边界条件则可以应用于壁面（感兴趣的读者可参阅 [6, 25, 2]）。同样，也可以添加特定条件，以正确处理平移周期性和/或旋转周期性。




**对于**时间区间 $n$ 至 $n+1$ **执行**积分 $\mathbf{X}^{n+1}$绘制向量 $\delta\mathbf{X} = \mathbf{X}^{n+1} - \mathbf{X}^n$**执行**确定 $\delta\mathbf{X}$ 是否通过一个面离开单元 更新当前单元 增加子迭代次数 $m$**当**穿过一个面 跟踪最终积分位置**结束循环**

图4：整个时间步执行一次积分后所使用的当前轨迹算法：使用连续邻居搜索算法跟踪粒子在时间步结束时所在的单元。

# 4 基于逐单元积分的新算法

在当前混合FV/PDF方法中，针对平均场值的 $P_0$-插值假设意味着，只要粒子在整个时间步内始终停留在起始单元中，第 3.2 节所述的指数格式不仅稳定，而且能够为粒子状态向量提供精确解。然而，在这类混合公式中，时间步长通常由平均流体流动计算所决定，因此未必总能缩短时间步长以满足这一判据。上述情况在弥散两相流模拟中尤为常见，其中离散粒子的惯性可能跨越多个数量级。本文仅研究流体粒子，但这一总体情况表明，我们面临这样一种情形：在给定时间步内，粒子可能穿越多个单元。欧拉格式中固有的假设（因此本文采用的指数格式也同样如此）只保留在时间步开始时，也就是在粒子的初始位置处计算的平均场值和时间尺度。在非均匀流中，平均流场量可能发生变化，因此会引入空间离散误差，如图 5 所示。

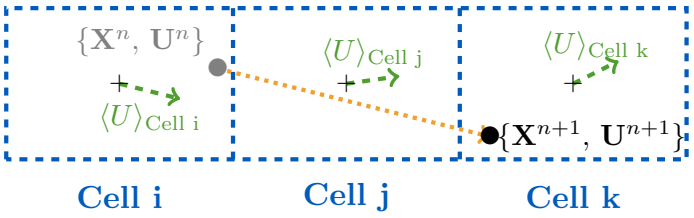



Figure 5: Sketch illustrating the issue with the current numerical scheme over large time steps in non-uniform flows: the particle can cross many cells in a single time step, meaning that the fluid characteristics (including velocities) in the intermediate cells are not taken into account in its trajectory.

**目标** 本文旨在开发一种新算法，使其在大时间步下，即使粒子可能穿越多个单元，也能在非均匀流中保持精度。为此，我们在保留相似假设/要求的同时，对第 3.2 节所述的数值格式进行扩展：

a) it should be explicit;

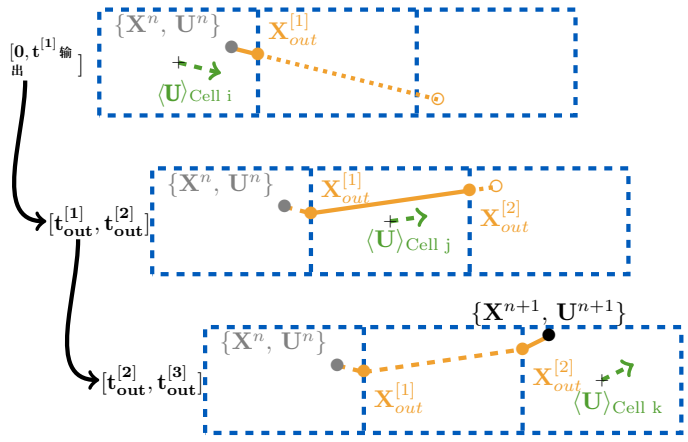
b) it should be unconditionally stable with respect to the time step;

c) it should be exact for constant flow mean fields (and thus within a given cell in line with the $P_0$ interpolation);

d) it should take into account spatial variations of flow mean fields encountered over large time steps.

### 4.1 Leading Principle: a cell-to-cell integration for large time steps

**原理** 为提高非均匀流中的数值精度，并适用于大时间步的情况，其思路是将当前的指数格式扩展为单元间积分。如图6所示，主导原理在于将时间步长划分为多个子迭代：每当粒子离开一个单元时，积分便停止，因此每个子迭代对应于粒子在一个单元内的运动。通过这种方式，可以直接考虑粒子每次进入新单元时流动特性的变化。由于需要检测粒子何时穿过两个相邻单元之间的界面，因此每个子迭代均应用以下两个步骤：**(1)** 时间积分格式和**(2)** 轨迹算法。当时间步长足够小时，粒子可在整个时间步内保持在同一单元中，此时新格式与前文所述格式相同。




**对于**时间区间 $n$ 至 $n+1$ **执行执行**获取相对相交时间 $\theta_{[m+1]}$积分直到该面 更新单元和剩余时间 增加子迭代 $m$**当**穿过一个面**结束对于**

Figure 6: Sketch illustrating the principle of a cell-to-cell integration for large time steps and the corresponding algorithm.

**其他要点**与第 3.2 节所述的两步过程相比，新的逐单元积分引出了另外两个问题：

- 粒子何时离开单元?
- 粒子从单元的何处离开?

这意味着必须修改轨迹算法，因为它只能提供新单元的信息（简单追踪）。新的轨迹算法应提供离开时间 $t_{\text{out}}$ 和离开位置 $\mathbf{X}_{\text{out}}$的信息。事实上，要计算粒子在时间步长剩余部分内的运动，确实需要这两项额外信息。事实上，如图6所示，粒子穿过面后的运动由一个时间积分格式给出，该格式从$\mathbf{X}_{\text{out}}$ 开始，在剩余时间 $\Delta t - t_{\text{out}}$内进行积分。从这个意义上说，本文的算法借鉴了 [36]提出的单元间积分，同时将其扩展，以满足 Itô 意义下随机积分的非预见性要求（下一段将对此进行讨论）。

**非预见性估计的关键问题** 实现逐单元算法的原理看似简单。朴素公式确实会认为，从时刻 $t^n$ 的粒子位置 $\mathbf{X}^n$ 出发，我们应用指数格式（参见公式（17）），预测时刻 $t^{n+1}$ 的 $\mathbf{X}^{n+1}$，然后使用轨迹算法确定 $\mathbf{X}_{\text{out}}$ 以及离开该单元的时间 $t_{\text{out}}$ （见图6上方的图）。然而，无论采用何种几何方法来确定这些量，显然，所得时间 $t_{\text{out}}$ 都取决于随机数 $\boldsymbol{\zeta}$；在公式（17）中，这些随机数表示随机模型中维纳过程的归一化增量，参见公式（10）。将每次子迭代得到的不同扩散系数（例如图6所示的三次子迭代）组合起来时，所得扩散系数便会成为维纳过程未来值的函数，因为随机数$\boldsymbol{\zeta}$ 表示 $\mathbf{W}^{n+1} - \mathbf{W}^n$ 的归一化值。换言之，这种看似简单却会产生误导的方法，将得到预见维纳过程未来的扩散系数。这直接违反了 Itô 意义下随机积分的定义，从而导致虚假的总体漂移和扩散数值（类似分析见 [19]）。

**Assumptions** Drawing on the aforementioned analysis, the new numerical scheme is designed so as to respect the following additional assumptions/constraints:

i) It should respect the non-anticipation rule with respect to the Wiener process;

ii) The trajectory algorithm is based on a neighbor search;

iii) The trajectory algorithm considers free-flight motion of particles within each sub-iteration.

## 4.2 Non-anticipating estimations of intermediate time steps using virtual partners

To obtain an estimate of the fraction of time spent by a particle in a given cell that remains independent of the future of the Wiener process entering the stochastic terms, a virtual partner is associated to each particle within each time step of the computation. This means that at the beginning of each time step $t = t^n$, this virtual partner starts from the same location as the real particle which is considered but moves in a deterministic manner (to be precised below) based only on the particle variables at time $t^n$ and local mean-field values. The fraction of time spent by this virtual partner in the corresponding cell is therefore, by construction, free of any statistical dependence with the Wiener process driving the random terms and provides an estimate for the time spent by the real particle within that cell.

下文首先在第4.2.1节介绍该算法的详细内容，然后在第4.2.2节展示该算法与当前积分方案的一致性。

#### 4.2.1 新算法概述

如前所述，新算法基于大时间步下的逐单元积分。这意味着每次时间迭代都分为一系列子迭代，每个子迭代对应粒子在给定单元中的运动。因此，该算法结合了若干连续的时间积分步骤和轨迹步。在每个时间步中，轨迹步利用虚拟伙伴来确定粒子所在的单元，并估算离开时间和位置。

实际上，该算法由以下步骤组成，如图7所示：

步骤1 基于粒子的初始位置$\mathbf{X}^{[0]}$ 以及该单元内流体平均场的知识，确定性估计时间步长结束时的粒子位置 $\Delta t$，记为$\mathbf{X}^{[1]}$, 。请注意，上标 $[m]$ 表示单元间积分中的子迭代编号 $m$ 。这是通过将指数时间积分格式中的随机积分置零实现的，由此得到$\mathbf{X}^{[1]}$的下列方程：

$$\widehat{\mathbf{X}^{[1]}} = \mathbf{X}^{[0]} + \mathbf{U}^{[0]} T_L^{[0]} \left\{ 1 - \exp\left( -\frac{\Delta t}{T_L^{[0]}} \right) \right\} + C_i^{[0]} T_L^{[0]} \left( \Delta t - T_L^{[0]} \left\{ 1 - \exp\left( -\frac{\Delta t}{T_L^{[0]}} \right) \right\} \right) \ . \qquad (18)$$

需要注意的是，$\mathbf{X}^{[1]}$ 对应于在时间 $\Delta t$、给定初始位置为$\mathbf{X}^{[0]}$,（即$\mathbf{X}^{[1]} = \langle \mathbf{X}(\Delta t) \ | \mathbf{X}(0) = \mathbf{X}^{[0]} \rangle$）条件下粒子位置的条件均值。因此，如果我们在时间 $t = 0$将若干粒子释放于$\mathbf{X}^{[0]}$ ，那么 $\mathbf{X}^{[1]}$ 表示这些释放粒子云在时间 $t = \Delta t$的质心位置。

步骤2 然后，应用基于自由飞行假设的轨迹算法，以确定粒子是否离开单元或留在单元内。为避免任何前瞻问题，轨迹算法应用于虚拟伙伴，而非真实粒子。假设虚拟伙伴在其初始位置（初始化为粒子位置 $\mathbf{X}^{[0]} = \mathbf{X}^{[0]}$）与$\mathbf{X}^{[1]}$之间沿直线运动。轨迹算法详见附录A。其结果可区分以下两种情况：

步骤2a 虚拟伙伴离开单元。在这种情况下，轨迹算法会提供虚拟粒子离开单元所经的面以及离开时间 $\theta^{[1]}\Delta t$ （参见附录A.1.2中对轨迹算法的扩展）。根据自由飞行假设，可直接根据离开时间推导出该虚拟伙伴的离开位置$\mathbf{X}^{[1]}$, 。此情况将触发下一次子迭代，以计算粒子在剩余时间内的运动（参见步骤3a）。

步骤2b 虚拟伙伴留在单元内。在这种情况下， $\theta^{[1]} = 1$，且虚拟伙伴在时间步长结束时的位置等于首次估算的位置$\mathbf{X}^{[1]} =\mathbf{X}^{[1]}$。此情况不会触发新的子迭代，单元间积分将停止（参见步骤3b）。

step-3 The position $\mathbf{X}^{[1]}$ and velocity $\mathbf{U}^{[1]}$ of the particle at the end of this sub-iteration are then computed using the full time-integration scheme, i.e., including the stochastic integrals (see Eq. (17)). The only issue is to determine the amount of time that the particle has spent in the cell. From step-2, there are two possibilities:

step-3a If the virtual partner exits the cell (step-2a), the elapsed time retained for the prediction of ($\mathbf{X}^{[1]}$, $\mathbf{U}^{[1]}$) is taken as being equal to the fraction of time computed for the virtual partner $\theta^{[1]}\Delta t$.
然后，我们必须继续计算剩余时间 $\Delta t^{[1]} =(1 - \theta^{[1]})\Delta t$ 内粒子的运动。按照这种逐单元算法的思路，可通过重复前述三个步骤并更新各步骤所需的信息来实现这一点（见图7中的子迭代2和3）。在第二次子迭代的步骤1中，从粒子位置 $\mathbf{X}^{[1]}$ 和粒子速度 $\mathbf{U}^{[1]}$ 出发，估计新时间步长 $\Delta t^{[1]}$ 结束时的条件平均粒子位置 $\mathbf{X}^{[2]}$ 。此时，用于估计 $\widehat{\boldsymbol{X}}^{[2]}$ 的平均场为 ($T_L(t, \widetilde{\boldsymbol{X}}^{[1]})$, $C_i(t, \widetilde{\boldsymbol{X}}^{[1]})$)，取自虚拟伙伴所在的单元。随后，应用步骤2计算虚拟伙伴的轨迹，假设它从上一个离开位置 $\mathbf{X}^{[1]}$ 出发并朝向估计得到的条件平均位置 $\mathbf{X}^{[2]}$ 移动。值得注意的是，起始位置

取为上一个离开位置，从而使虚拟伙伴的轨迹保持连续。随后，根据步骤2中得到的 $\theta_2$ 估计值，在这次子迭代的步骤3中计算位置 $\mathbf{X}^{[2]}$ 和速度 $\mathbf{U}^{[2]}$ 。

实际上，这个三步过程会重复进行，直到虚拟伙伴到达一个能够在整个剩余时间步长内停留其中的单元 $\Delta t^{[m]}$ （请注意，上标 $[m]$ 对应于子迭代编号 $m$）。

步骤3b 如果虚拟粒子停留在该单元中（步骤2b），则经过的时间实际上等于剩余时间步长 $\Delta t^{[m]}$ （即 $\theta^{[m]} = 1$）。虚拟伙伴的位置等于步骤1中最后一次$\frown\!\!\!\smile$ $\frown\!\!\!\smile$估算的位置$\mathbf{X}^{[m]}$ =$\mathbf{X}^{[m]}$。在这种情况下，单元间迭代积分便停止。

最后一步 当虚拟伙伴最终到达一个能够在剩余时间内停留其中的单元时，我们便得到整个时间步长结束时的粒子位置$\mathbf{X}(t+\Delta t)$。然而，如图7所示，由于随机项的影响，粒子最终可能位于与虚拟伙伴不同的单元中。因此，我们采用最后轨迹算法：追踪虚拟伙伴最后位置$\mathbf{X}^{[m]}$到粒子位置$\mathbf{X}(t+\Delta t)$之间的单元变化。这个最后轨迹步骤确保粒子与其在时间步长结束时所在的正确单元相关联。这样可以确保下一个时间步长的位置不会产生误差。

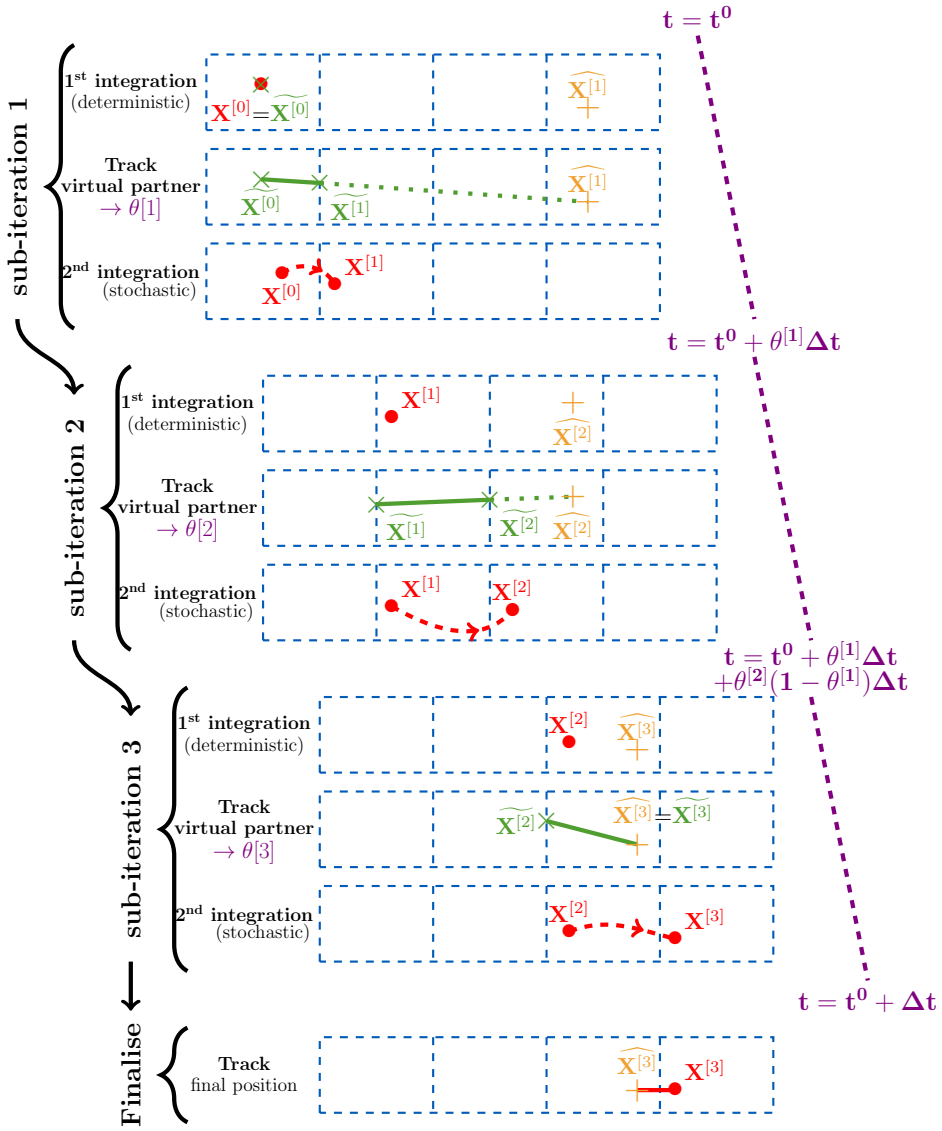
这里有几点需要说明。乍看之下，可以认为，虚拟伙伴 $\mathbf{X}^{[m]}$ 的位置$\frown\!\!\!\smile$与上一次子迭代时粒子的位置 $\mathbf{X}(t+\Delta t)$ 之间的差异表明算法存在错误。然而，需要注意的是，这种差异源于计算粒子位置时所考虑的随机项，而虚拟伙伴的位置并未考虑这些随机项（虚拟伙伴的位置是完全确定性的）。在此需要强调的是，在计算的每个时间步长内，虚拟伙伴的作用主要是提供粒子在每个单元中所花费时间的信息，并确定粒子在该时间步长内可以穿过的相邻单元，同时确保避免预见性问题。这进一步支持了采用基于连续邻居搜索的追踪算法（见第3.2.2节），因为该算法能够自然地提供粒子在一个时间步长内连续穿过的单元信息（不同于简单的定位算法）。在算法的每次子迭代中，该虚拟伙伴的增量表示条件平均位移，因此在这一意义上仍与所建模的底层物理过程保持一致。因此，每个虚拟伙伴都是计算（真实）粒子运动的一个辅助量。此外，我们基本上关注的是开发弱近似，这意味着从统计意义上捕捉粒子动力学。因此，真正重要的是通过蒙特卡罗方法获得从粒子动力学中提取的统计量的准确估计（例如平均浓度、平均速度），也就是从粒子系综中获得这些估计。换言之，对单个粒子变量（位置以及速度）的预测进行改进时，始终应从统计意义上加以评估。为此，在本文余下部分，我们将转而关注这类粒子统计量所呈现的行为。

#### 4.2.2 时间步长分解的一致性分析

在第5节考虑适当的确认测试案例之前，重要的是检查新算法是否满足第4节开头提出的判据 (c)。

**验证过程** 为此，我们考虑所有平均场和时间尺度均为常数的情形。在这种情况下，我们已经知道，当前数值格式（参见式 (17)，其中 $T_L$、 $C_i$和 $\epsilon$ 现在均为常数）在弱意义下给出精确解。接着，我们考虑新算法，并假设在时间 $\theta\Delta t$已进行一次子迭代（推广到多次子迭代显而易见），这意味着粒子状态向量$\mathbf{Z}^{n+1}$ 现在是通过两次迭代（一次针对 $\theta\Delta t$ ，另一次针对 $(1-\theta)\Delta t$）由$\mathbf{Z}^n$ 得到的。接下来要检查的是，即使在弱意义下，这两个预测值仍然相同。

下文中，我们省略方向指标 $i$ ，因为这基本上是一维表述。从给定的时刻 $t^n$ 的粒子状态向量值 $\mathbf{Z}^n$ 出发，单次迭代（即当前算法）得到的预测值用指标 [1], 表示，而两次子迭代（即新算法）得到的预测值则用指标 [2]表示。




**对于**时间区间 $n$ 至 $n+1$，**执行**

执行 (a) 从 $\mathbf{X}^{[m]}$ 积分 $\mathbf{X}^{[m+1]}$，使用 $I^Z = 0$ (b) 使用向量 $\mathbf{X}^{[m]}\mathbf{X}^{[m+1]}$ 计算离开点 $\mathbf{X}^{[m+1]}$ 使用 $\theta^{[m+1]} = |\mathbf{X}^{[m]}\ \mathbf{X}^{[m+1]}|\ \mathbf{X}\ \mathbf{X}\ \mathbf{X}$ 计算离开时间 $\theta^{[m+1]}dt^{[m]}$ (c) 使用离开时间 $\theta^{[m+1]}\mathrm{d}t^{[m]}$ 和随机数 ($\zeta^Z$)，从 $\mathbf{X}^{[m]}$ 积分 $\mathbf{X}^{[m+1]}$ $^{[m+1]}$(d) 更新单元、流场和剩余时间 $\mathrm{d}t^{[m+1]} = (1-\theta^{[m+1]})\mathrm{d}t^{[m]}$增加子迭代次数 $m$当（穿过一个面时）追踪最终积分位置结束

Figure 7: Summary of the proposed algorithm.

Predictions in one iteration are expressed directly by Eqs. (17), which are rewritten here as

$$(X^{n+1})^{[1]} = X^n + U^n T_L \left(1 - \exp\left(-\frac{\Delta t}{T_L}\right)\right) + C T_L \left[\Delta t - T_L \left(1 - \exp\left(-\frac{\Delta t}{T_L}\right)\right)\right] + (I^X)^{[1]}(\Delta t)\ , \quad \text{(19a)}$$

$$(U^{n+1})^{[1]} = U^n \exp\left(-\frac{\Delta t}{T_L}\right) + C T_L \left[(1 - \exp\left(-\frac{\Delta t}{T_L}\right)\right] + (I^U)^{[1]}(\Delta t). \quad \text{(19b)}$$

The two stochastic 积分由两个相关的中心高斯随机变量表示 s

$(I^X)^{[1]}(\Delta t)$以及$(I^U)^{[1]}(\Delta t)$，它们完全由其二阶矩矩阵 [29 确定 ]

$$\left\langle \left((I^U)^{[1]}(\Delta t)\right)^2 \right\rangle = C_0\epsilon\frac{T_L}{2}\left(1-\exp\left(-2\frac{\Delta t}{T_L}\right)\right) , \tag{20a}$$

$$\left\langle (I^U)^{[1]}(\Delta t)\,(I^X)^{[1]}(\Delta t) \right\rangle = C_0\epsilon\frac{T_L^2}{2}\left(1-\exp\left(-\frac{\Delta t}{T_L}\right)\right)^2 , \tag{20b}$$

$$\left\langle \left((I^X)^{[1]}(\Delta t)\right)^2 \right\rangle = C_0\epsilon T_L^2\left\{\Delta t-\frac{T_L}{2}\left[1-\exp\left(-\frac{\Delta t}{T_L}\right)\right]\left[3-\exp\left(-\frac{\Delta t}{T_L}\right)\right]\right\} . \tag{20c}$$

在进行两次迭代的情况下，首先使用公式（19），通过标记为$(I^U)^{[2]}(\theta\Delta t)$和$(I^X)^{[2]}(\theta\Delta t)$的两个相关高斯随机变量，对$\mathbf{Z}(\theta\Delta t)$进行估计；然后在剩余时间区间$(1-\theta)\Delta t$ 上执行第二次迭代，以获得$\mathbf{Z}^{n+1}$ 的预测，此次迭代使用另外两个标记为$(I^U)^{[2]}((1-\theta)\Delta t)$和$(I^X)^{[2]}((1-\theta)\Delta t)$的相关高斯随机变量。注意，$\left[(I^U)^{[2]}((1-\theta)\Delta t)、(I^X)^{[2]}((1-\theta)\Delta t)\right]$与$\left[(I^U)^{[2]}(\theta\Delta t)、(I^X)^{[2]}(\theta\Delta t)\right]$相互独立。将两者结合可得

$$\begin{aligned}(X^{n+1})^{[2]} = X^n + U^nT_L\left(1-\exp\left(-\frac{\Delta t}{T_L}\right)\right) + CT_L\left[\Delta t-\exp\left(-\frac{\Delta t}{T_L}\right)\right] \\ + \underbrace{(I^X)^{[2]}(\theta\Delta t) + (I^X)^{[2]}((1-\theta)\Delta t) + T_L(I^U)^{[2]}(\theta\Delta t)\left(1-\exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right)\right)}_{(\widetilde{I}^X)^{[2]}(\theta,\Delta t)},\end{aligned} \tag{21a}$$

$$\begin{aligned}(U^{n+1})^{[2]} = U^n\exp\left(-\frac{\Delta t}{T_L}\right) + CT_L\left(1-\exp\left(-\frac{\Delta t}{T_L}\right)\right) \\ + \underbrace{(I^U)^{[2]}(\theta\Delta t)\exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right) + (I^U)^{[2]}((1-\theta)\Delta t)}_{(\widetilde{I}^U)^{[2]}(\theta,\Delta t)}.\end{aligned} \tag{21b}$$

The variance of $(\widetilde{I}^U)^{[2]}(\theta,\Delta t)$ is easily calculated and is:

$$\left\langle \left((\widetilde{I}^U)^{[2]}(\theta,\Delta t)\right)^2 \right\rangle = \left\langle \left((I^U)^{[2]}(\theta\Delta t)\right)^2 \right\rangle \exp\left(-\frac{2(1-\theta)\Delta t}{T_L}\right) + \left\langle \left((I^U)^{[2]}((1-\theta)\Delta t)\right)^2 \right\rangle \tag{22a}$$

$$= C_0\epsilon\frac{T_L}{2}\left[\left(1-\exp\left(-\frac{2\theta\Delta t}{T_L}\right)\right)\exp\left(-\frac{2(1-\theta)\Delta t}{T_L}\right)+1-\exp\left(-\frac{2(1-\theta)\Delta t}{T_L}\right)\right] \tag{22b}$$

$$= C_0\epsilon\frac{T_L}{2}\left[1-\exp\left(-\frac{2\Delta t}{T_L}\right)\right] \tag{22c}$$

$$= \left\langle \left((I^U)^{[1]}(\Delta t)\right)^2 \right\rangle, \tag{22d}$$

where the last equality comes from Eq. (20a). Similarly, the variance of $(\widetilde{I}^X)^{[2]}(\theta,\Delta t)$ is obtained as:

$$\begin{aligned}\left\langle \left((\widetilde{I}^X)^{[2]}(\theta,\Delta t)\right)^2 \right\rangle \quad &= T_L^2\left[1-\exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right)\right]^2\left\langle \left((I^U)^{[2]}(\theta\Delta t)\right)^2 \right\rangle \\ &+2T_L\left[1-\exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right)\right]\left\langle \left((I^X)^{[2]}(\theta\Delta t)(I^U)^{[2]}(\theta\Delta t)\right) \right\rangle \\ &+\left\langle \left((I^X)^{[2]}(\theta\Delta t)\right)^2 \right\rangle + \left\langle \left((I^X)^{[2]}((1-\theta)\Delta t)\right)^2 \right\rangle ,\end{aligned} \tag{23}$$

while the covariance is

$$\begin{aligned}\left\langle \left((\widetilde{I}^X)^{[2]}(\theta,\Delta t)(\widetilde{I}^U)^{[2]}(\theta,\Delta t)\right) \right\rangle = T_L\left[1-\exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right)\right]\exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right)\left\langle \left((I^U)^{[2]}(\theta\Delta t)\right)^2 \right\rangle \\ +\exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right)\left\langle \left((I^X)^{[2]}(\theta\Delta t)(I^U)^{[2]}(\theta\Delta t)\right) \right\rangle + \left\langle \left((I^X)^{[2]}((1-\theta)\Delta t)(I^U)^{[2]}((1-\theta)\Delta t)\right) \right\rangle .\end{aligned} \tag{24}$$

The formulas in Eqs. (20) can then be applied, using either $\theta\Delta t$ or $(1-\theta)\Delta t$ as the proper time interval, and injected in Eqs. (23) and (24). Tedious but straightforward calculations show then that we have

$$\left\langle \left( (\widetilde{I}^X)^{[2]}(\theta,\Delta t) \right)^2 \right\rangle = \left\langle \left( (I^X)^{[1]}(\Delta t) \right)^2 \right\rangle \tag{25a}$$

$$\left\langle \left( (\widetilde{I}^X)^{[2]}(\theta,\Delta t)(\widetilde{I}^U)^{[2]}(\theta,\Delta t) \right) \right\rangle = \left\langle (I^U)^{[1]}(\Delta t)(I^X)^{[1]}(\Delta t) \right\rangle , \tag{25b}$$

结合公式（22d），可证明 $\left[(\widetilde{I}^X)^{[2]}(\theta,\ \Delta t),\ (\widetilde{I}^U)^{[2]}(\theta,\ \Delta t)\right]$ 与 $\left[(I^X)^{[1]}(\Delta t),\ (I^U)^{[1]}(\Delta t)\right]$在统计上等价，因此公式（21）与公式（19）相同。

**重要说明** 即使 $\theta$ 是$\mathbf{Z}^n$ 、平均场和时间尺度（此处为 $C$ 和 $T_L$）的函数，这些计算仍然成立。不过，需要强调的是， $\theta$ 与表示随机积分的随机变量相互独立，这一点对于上述验证至关重要；否则，这些性质便不再成立。事实上，在这种不合理的情况下，我们会得到:

$$\left\langle \exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right)(I^{Z_1})^{[2]}(\theta\Delta t)(I^{Z_2})^{[2]}(\theta\Delta t) \right\rangle \neq \left\langle \exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right) \right\rangle \left\langle (I^{Z_1})^{[2]}(\theta\Delta t)(I^{Z_2})^{[2]}(\theta\Delta t) \right\rangle, \tag{26}$$

其中 $Z_1$ 和 $Z_2$ 为 $X$ 或 $U$，且:

$$\left\langle \exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right) \right\rangle \neq \exp\left(-\frac{(1-\theta)\Delta t}{T_L}\right). \tag{27}$$

这反映了本节中反复指出的非预见性问题 .

完成这一验证测试案例后，我们现在可以考察确认测试案例。下一节将对此进行讨论。

# 5 数值结果

确认第4节开头提出的所有标准均已满足后，新算法现已通过验证。为此，该算法已在开源CFD求解器 Code Saturne 中实现 [1]，并在两种情况下将数值结果与解析结果进行比较: -

- 第一种情况旨在检验：当所有平均场和时间尺度均为常数（即均匀平均流）时，通过蒙特卡洛方法从粒子附着变量中获得的统计量是否与解析表达式一致。第5.1节通过定常均匀各向同性湍流中的点源粒子弥散来进行这项检验。

- 第二种情况对应于一种简单但具有挑战性的非均匀流，其中包含弯曲流线，可用于评估新算法在复杂情况下的准确性；在这类情况下，经典粒子跟踪算法往往会遇到局限性（见第 5.2 节）。

## 5.1 Validation in a uniform flow

Drawing on the verification test case carried out in Section 4.2.2, the aim is twofold: first, to check that, with the new proposed time-splitting method, numerical results are exact when flow properties are constant in time and space; second, to bring out discrepancies induced when using an anticipation method even in such a simple case. To that end, the dispersion of particles released from a point source is analyzed in a stationary homogeneous isotropic turbulent flow. This choice is motivated by the fact that, in such a context, analytical formulas can be derived [22].

In the following, we start by describing the system considered, including physical aspects (e.g., flow characteristics), as well as numerical ones (e.g., spatial discretization, time step) and theoretical results. Then, numerical results are validated with respect to these theoretical expressions.

### 5.1.1 所考虑的系统：均匀各向同性湍流中的点源弥散

**Physical aspects** The case studied consists of a point-source (fluid) particle dispersion in a stationary homogeneous isotropic turbulent flow. This means that the flow is uniform and stationary within the domain considered and that it is periodic in all directions. Here, we have imposed a velocity based on the turbulent dispersion $U_\alpha = \sqrt{\langle u_\alpha^2 \rangle} = 1\,\mathrm{m\,s^{-1}}$ and a Lagrangian time scale $T_L = 1\,\mathrm{s}$.

**数值方面** 使用计算流体力学软件 Code Saturne 进行了模拟。通过施加额外的人工强迫来维持系统内恒定的能量水平，从而设定平均流场 [32]。这使流场的耗散率保持恒定，等于 $\epsilon = \frac{2}{C_0}\frac{U_\alpha^2}{T_L}$，其中均匀定常流动的 $T_L$ 适当闭合为 $T_L = 4k/(3C_0\epsilon)$ [32, 21]，且 $k = 3/2U_\alpha^2$，朗之万模型扩散系数中的 Kolmogorov 常数为 $C_0$。

获得流体平均场后，在方箱中心注入 100 000 个流体粒子。使用第4节所述的新算法计算这些粒子的轨迹。定义域（方箱）的总尺寸设置得足够大，以确保每次模拟期间粒子都不会到达边界。

**理论方面**如第4.2.2节所示，当所有平均场和时间尺度均为常数时，该算法是精确的。借助 Itô 演算，可以直接推导出方差 $\langle X^2 \rangle$ 和 $\langle U^2 \rangle$ 以及协方差 $\langle XU \rangle$ 随时间变化的解析公式，其中以 $t_0 = 0$ 作为粒子释放的初始时刻。根据这些解析解，我们可以得到两个极限情形：弹道极限（即 $t \ll T_L$ 时）和扩散极限（即 $t \gg T_L$ 时）。 [22], 中已对此进行了推导，得到这两个极限情形下的以下表达式：

| Ballistic limit case ($t \ll T_L$) | | 扩散极限情形（$t \gg T_L$） | |
|---|---|---|---|
| $\langle X^2 \rangle \sim C_0 \dfrac{\epsilon t^2 T_L}{2} = U_\alpha^2 t^2,$ | (28a) | $\langle X^2 \rangle \sim C_0 \epsilon T_L^2 t = 2U_\alpha^2 T_L t,$ | (29a) |
| $\langle XU \rangle \sim \quad C_0 \epsilon\, t^2 \quad = 2U_\alpha^2 \dfrac{t^2}{T_L},$ | (28b) | $\langle XU \rangle \sim \dfrac{C_0 \epsilon T_L^2}{2} = U_\alpha^2 T_L,$ | (29b) |
| $\langle U^2 \rangle \sim \quad C_0 \epsilon\, t \quad = 2U_\alpha^2 \dfrac{t}{T_L}.$ | (28c) | $\langle U^2 \rangle \sim \dfrac{C_0 \epsilon T_L}{2} = U_\alpha^2.$ | (29c) |

值得一提的是，在扩散区间，唯一具有物理意义的统计量是 $\langle X^2 \rangle$，因为粒子瞬时速度不再影响粒子位置演化方程。这对应于快变量消去的概念（此处 $U$ 成为可消去的快变量）；标准教材 [8] 以及先前从理论角度 [21, 17] 和数值角度 [29]开展的研究中，都对此进行了详尽讨论。速度二阶矩的常数值仍保留其典型的动能含义（这表明我们处理的实际上是与某种恒温器接触的粒子），但相关性 $\langle XU \rangle$的物理意义较弱，因此也予以考察，因为这两项结果都有助于讨论蒙特卡罗估计中的统计噪声。

### 5.1.2 验证

下面我们检查，在以下两种情况下，无论调用时间分裂算法的次数如何，数值结果都与解析公式一致：

- 第一组模拟侧重于弹道状态。为此，时间步长取为 $\Delta t = 0.05 T_L$。考虑了两种空间离散：
  $\Delta x = U_\alpha \Delta t/50$和 $\Delta x = 10 U_\alpha \Delta t$。这两种网格尺寸经过仔细选择，使得第一种（分别为第二种）对应于一个时间步长内的粒子平均位移远小于（分别远大于）网格尺寸 $\Delta x$ 的情况。模拟运行的总时间等于 $6T_L$。

  Figures 8A-8C display the second-order moments as a function of the dimensionless time $t^* = t/T_L$, comparing results obtained with the new algorithm to analytical formulas. It can be seen that the outcomes of the new algorithm remain exact for both cases. This validates the new algorithm (which

has been activated on average 50 times every time step in the case where $\Delta x = U_\alpha \Delta t/50$). Small differences between results obtained with the cell-to-cell algorithm and analytical values are simply related to statistical errors using Monte Carlo methods (due to a finite number of fluid particles), as discussed below.

如图 8D–8F 所示，使用预判方法（例如第 4.1 节中概述的朴素公式）会得到截然不同的结果。这一点从结果对单元尺寸的强烈依赖性中尤为明显，因为单元尺寸确实决定了应用时间分裂算法的频率。单元尺寸足够大时（也就是说，新算法很少被调用），结果大致仍可接受；而采用较小的单元尺寸时（即使用 $\Delta x = U_\alpha \Delta t/50$），数值预测会迅速恶化，从而导致严重误差。

- 在扩散极限情形下还进行了一组模拟。为此，时间步长取为 $\Delta t = 200 T_L$。为评估数值结果是否独立于新算法的应用次数，还考虑了两种空间离散： $\Delta x = U_\alpha \Delta t/20$和 $\Delta x = 2.5\ U_\alpha \Delta t$. 模拟总时长为 $24\,000\ T_L$。

  图9A至图9C展示了二阶矩随时间的演化，表明无论在一个时间步长内使用时间分裂算法的次数是多少，新算法都能提供准确的结果。在图9A中可以看到，$\langle X^2 \rangle$ 从一开始就呈线性演化，这正是扩散区间中的应有表现；采用非预判方法时，其斜率也得到了正确再现。这说明粒子弥散系数得到了准确捕捉。如上文所述，考察图9B中的 $\langle U^2 \rangle$ 和图9C中的 $\langle XU \rangle$，有助于凸显蒙特卡罗方法所固有的统计噪声。对于 $\langle U^2 \rangle$，蒙特卡洛估计量的方差不随时间变化，因为 $\mathrm{Var}(U^2) = 2\langle U^2 \rangle^2$，99%置信区间由图9B和图9E中所示的两条水平直线表示。然而，对于相关性 $\langle XU \rangle$，估计量的方差是时间的函数，因为 $\mathrm{Var}(XU)(t) = (C_0\epsilon)^2 T_L^4/2\,(1 + t/T_L)$，因此，限定99%置信区间的包络线会随时间增加，如图9C和图9F所示。再次注意，$\langle XU \rangle$ 所产生的不断增加的噪声水平仅仅是一个可观测量，不会对粒子模拟产生反馈作用。

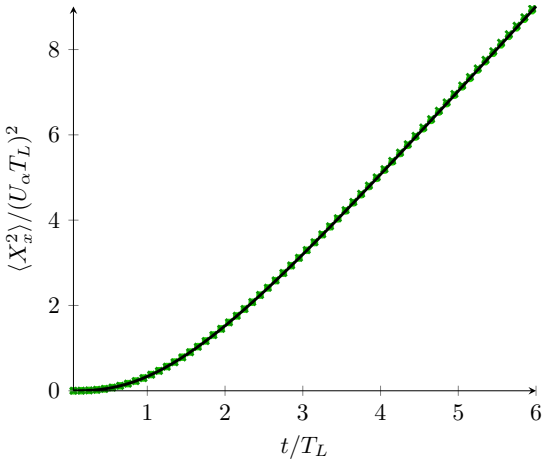
  另一方面，如图 9D 所示，预判方法（例如第 4.1 节概述的朴素公式）无法再现正确的弥散系数，尤其是在频繁调用时间分裂算法时（在图 9D 中，这对应于平均每次迭代跨越 20 个单元的情况）。为完整起见，我们还在图 9E 中展示了 $\langle U^2 \rangle$，并在图 9F 中展示了相关性 $\langle UX \rangle$，这进一步证实，预判方法所得结果虽会波动，但其波动中心是错误的平均值。

此时需要强调的是，当前结果是在一维弥散情形下获得的。然而，由于每个方向均被独立处理，这些结果可以直接推广到三维弥散情形。此外，与规则一维笛卡尔网格相比，使用三维非结构网格也获得了类似的行为（详情见附录 A.2）。

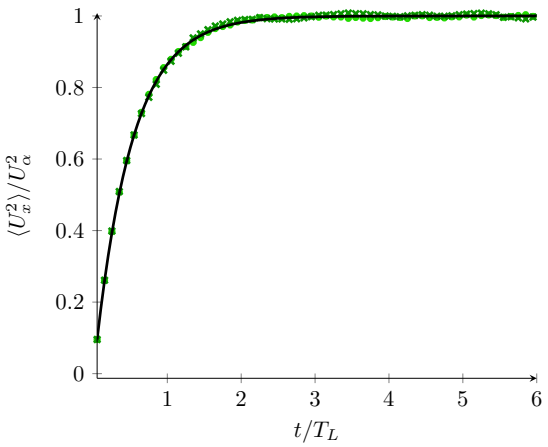
## 5.2 非均匀流中的验证

在均匀流中验证了新算法后，接下来将在非均匀流情况下对该算法进行验证。为此，考虑一种层流，使算法时间积分部分中的随机项等于零。这确保了时间积分部分具有确定性，从而可以检验所计算的非均匀流中轨迹是否精确。此外，选择圆柱库埃特流是因为它是最简单的非均匀流之一。在这种情况下，预计每个流体粒子确实会围绕圆柱体的旋转中心做纯圆周运动（见第 5.2 节）。

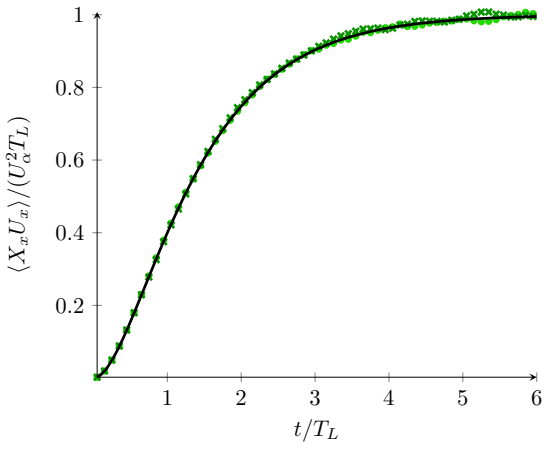
下文首先介绍所考虑的系统，包括物理方面（例如旋转圆柱体的几何形状、流动特性）和数值方面（例如空间离散、时间步长）。随后评估新轨迹算法的精度。这一验证分为两个步骤：首先，验证单个粒子的运动确实遵循纯圆周运动；其次，验证粒子浓度的统计量在整个模拟时间内保持不变，而与所采用的时间步长无关。



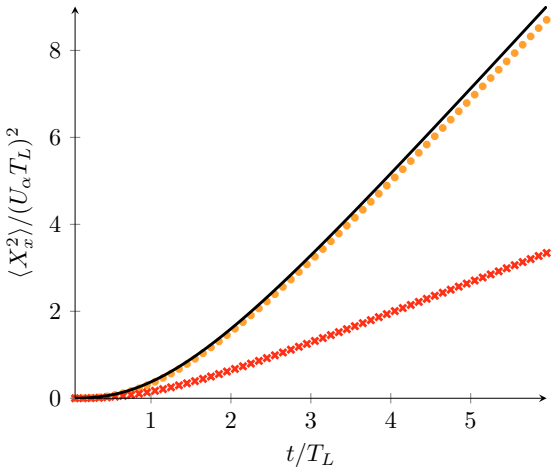

(A) $\langle XX\rangle$ using the proposed method.



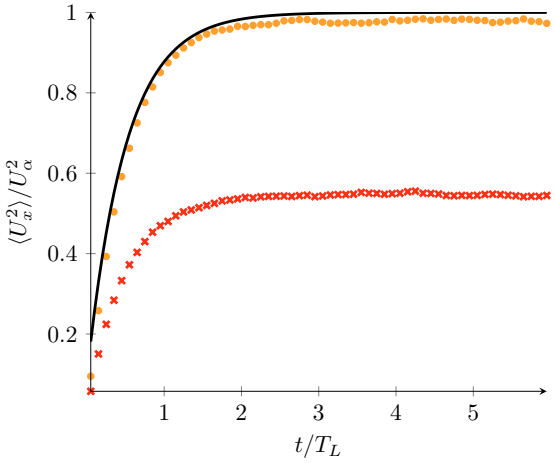

(B) $\langle UU\rangle$ using the proposed method.



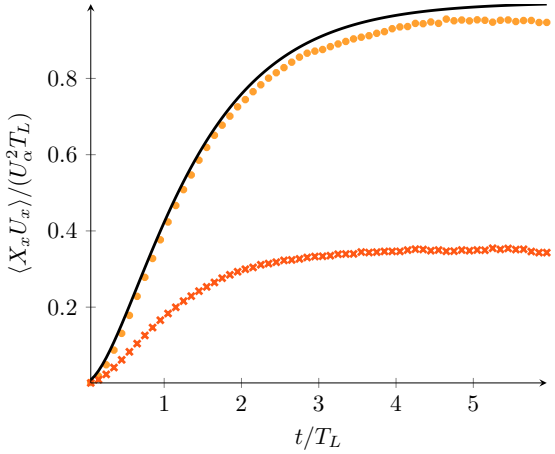

(C) $\langle XU\rangle$ using the proposed method.




(D) $\langle XX\rangle$ using an anticipating method.

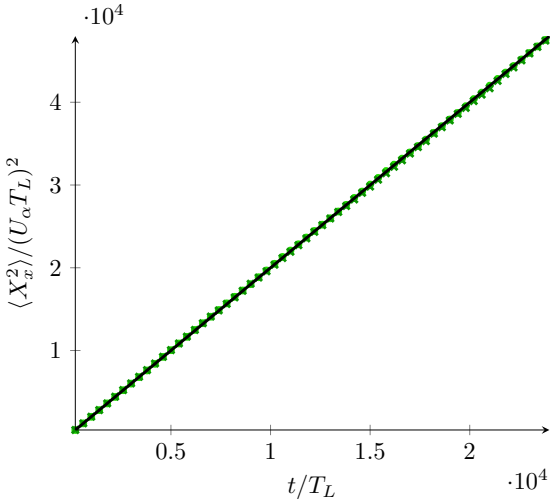



(E) $\langle UU\rangle$ using an anticipating method.

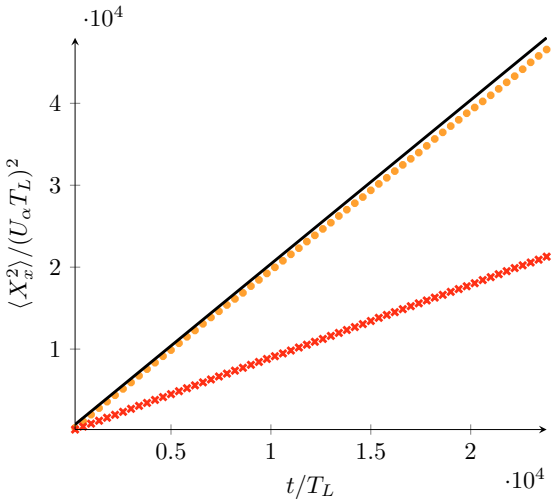



(F) $\langle XU\rangle$ using an anticipating method.

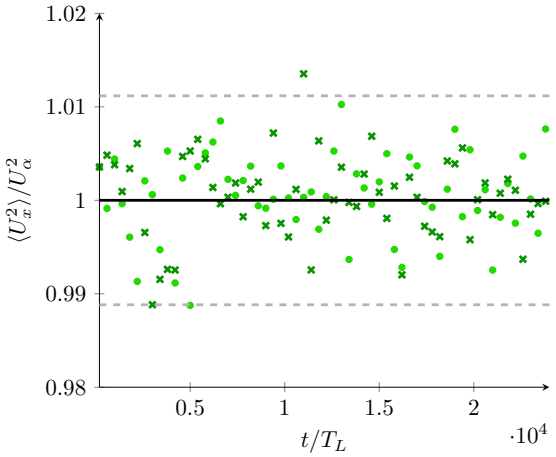
图8：在时间步长为 $\Delta t = 0.05T_L$的弹道极限情况下，$\langle XX\rangle$、$\langle UU\rangle$、$\langle UX\rangle$随无量纲时间的演化。考虑了两种空间细化：$\times$（分别为 $\bullet$）对应于单元尺寸为 $\Delta x = U_\alpha \Delta t/50$（分别为 $10\ U_\alpha \Delta t$）的模拟。左侧为解析解（黑色直线）与采用预见方法获得的数值结果的比较。右侧为使用新算法得到的误差。虚线对应于99%置信区间的包络线（解析公式）。



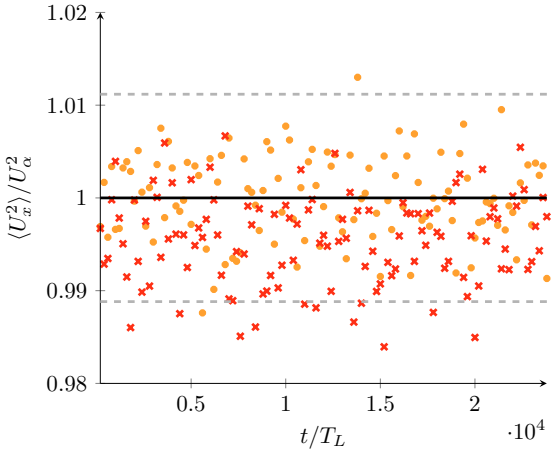

(A) $\langle XX\rangle$ using the proposed method.



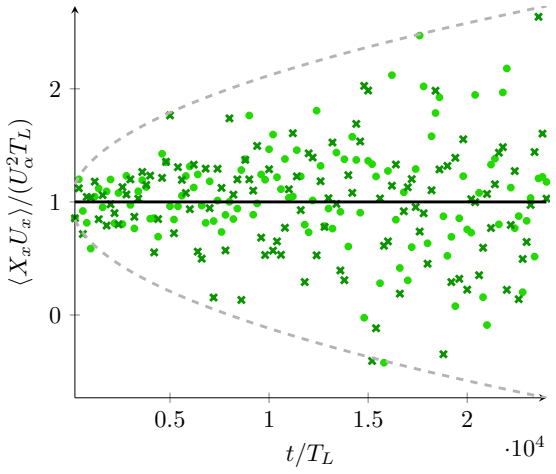

(D) $\langle XX\rangle$ using an anticipating method.



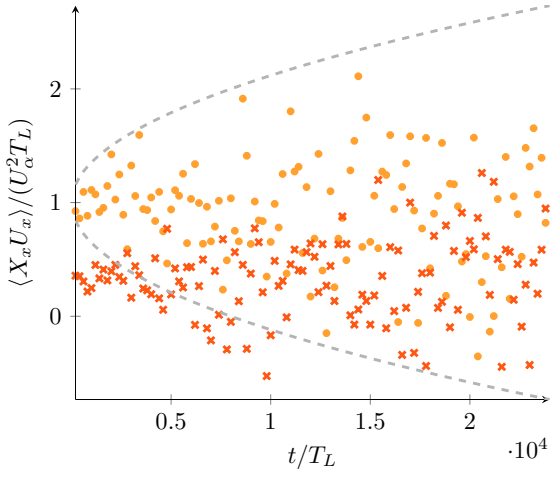

(B) $\langle UU\rangle$ using the proposed method.




(E) $\langle UU\rangle$ using an anticipating method.




(C) $\langle XU\rangle$ using the proposed method.




(F) $\langle XU\rangle$ using an anticipating method.

图9：在扩散极限情形下，$\langle XX\rangle$, $\langle UU\rangle$, $\langle UX\rangle$随无量纲时间的演化，时间步长为 $\Delta t = 200T_L$。考虑两种空间细化：× （分别对应以单元尺寸为 $\Delta x = U_\alpha \Delta t/20$ 的模拟），以及 •（分别对应以单元尺寸为 2.5 $U_\alpha \Delta t$的模拟）。左图为解析解（黑色直线）与采用预见方法得到的数值结果的比较。右图为采用新算法得到的误差。虚线表示99%置信区间的包络线（解析公式）。

### 5.2.1 所考虑的系统：层流圆柱库埃特流

**物理参数** 此处研究的情形为层流圆柱库埃特流。它由两个圆柱体之间的流体流动组成：内圆柱半径为 $r_{in} = 1\mathrm{m}$，以给定的角速度 $\omega_\theta(r_{in}) = 1\mathrm{s}^{-1}$ 旋转，而外圆柱半径为 $r_{out} = 2\,\mathrm{m}$，保持静止。这意味着流体被限制在分隔两个圆柱体的环隙内，该环隙厚度为 $\delta r = 1\mathrm{m}$。流体的运动黏度设为 $1\mathrm{m}^2\mathrm{s}^{-1}$，因此基于这些量计算得到的雷诺数等于 1（即处于层流区）。

在此配置下，流体进行圆柱运动。因此，速度场是单向的，并且使用圆柱坐标 $\mathbf{e}_\theta$ 表示时，其解析解为：

$$U = U_\theta(r) = \langle U_\theta \rangle(r) = \frac{\omega_\theta(r_{in})}{\dfrac{r_{in}}{r_{out}} - \dfrac{r_{out}}{r_{in}}} \left( \frac{r}{r_{out}} - \frac{r_{out}}{r} \right) \tag{30}$$

其中 $r$ 是距旋转中心的距离。

**数值参数** 使用 CFD 软件 Code Saturne 在 360° 多面体网格上求解纳维–斯托克斯方程，以开展模拟：该网格在方位角方向上包含 360 个单元（即方位角离散角度为 $\Delta\theta = 1°$），在径向上包含 21 个单元。模拟采用恒定时间步长，其取值范围为 0.055 s 至 200 s。选择这一取值范围是为了评估算法精度与时间步长的关系，尤其是在时间步较大、粒子平均位移远大于网格尺寸的情况下。

通过应用第4节所述算法，追踪层流圆柱库埃特流中的流体粒子。拉格朗日积分时间 $T_L$ 设为 0 s，以强制流动保持层流（即所有随机积分均为零）。

### 5.2.2 数值结果的准确性

新算法通过以下方式进行验证：首先，检查单个粒子的轨迹是否遵循纯圆柱运动；其次，检查径向上的初始粒子浓度是否随时间保持不变。下文中的所有结果均使用定义为 $t^+ = t\,\omega_\theta(r_{in})/(\Delta\theta)$ 的无量纲时间绘制。这意味着，无量纲时间步长 $\Delta t^+$ 表示内壁附近（沿方位角方向）的粒子平均位移与网格尺寸之比。

**单个粒子轨迹的结果** 层流圆柱库埃特流中流体粒子的运动预计将遵循纯圆形轨迹。这意味着，在整个模拟过程中，每个粒子都保持与旋转中心的恒定距离 $r$。模拟使用 1.024 s 的时间步长运行了 400 次迭代。这意味着，内圆柱附近的粒子平均位移等于网格尺寸 $\Delta\theta \times r_{in}$ 的 55 倍，而外圆柱附近则等于网格尺寸 $\Delta\theta \times r_{out}$ 的 0.5 倍。

Figure 10 displays the trajectory of two particles: one is initially located very close to the inner cylinder while the second one is initially located in the bulk. The trajectory (left-hand plot) clearly shows that the new algorithm does lead to a circular motion. These results confirm the accuracy of the new algorithm even for large values of the time step. This is further supported by the time-evolution of the radius $r$ (right-hand plot). In fact, the results are not distinguishable from the theoretical expectations, even for the particle initially close to the inner cylinder (which has circled the cylinder close to 60 times by the end of the simulation).

图10还展示了使用参考算法获得的结果，该算法未考虑单元间积分。可以看到，半径随时间增加的幅度会随着时间步长的增大而增大。这是由于在使用大时间步时未进行单元间积分所产生的误差。图11说明了该误差的来源：空间离散和 $P_0$ 插值意味着每个粒子在一个单元内经历恒定速度。当粒子穿过一个面时，如果速度未立即更新，则该粒子将在剩余的时间步内以其在前一个单元中遇到的速度继续运动。由于这一先前的流体速度相对于当前新单元中的径向方向存在轻微偏移，在未进行单元间积分的情况下会产生误差，并且该误差与时间步长成正比。

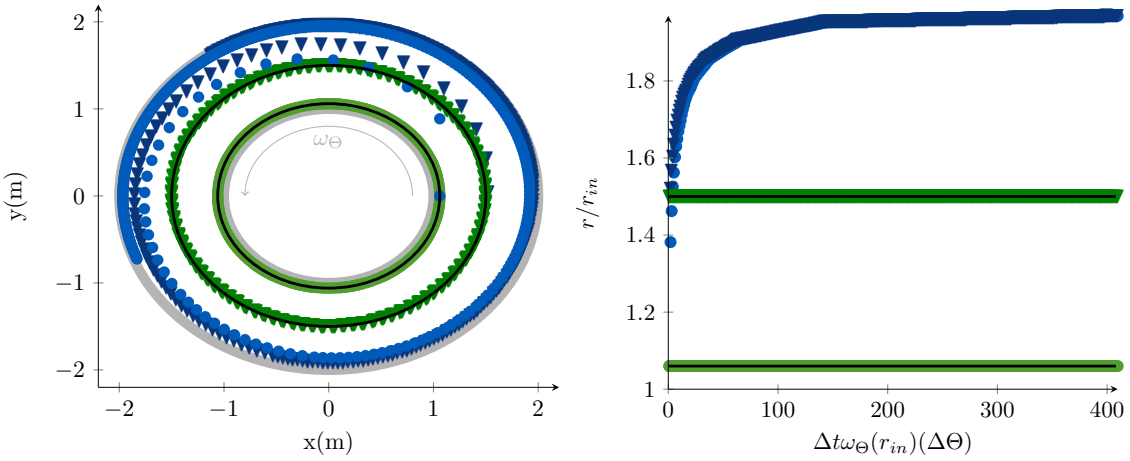



图10：恒定时间步长 $\Delta t^+ = 55$下，粒子轨迹（左）及半径 $r$ 随无量纲时间 $t^+$ 变化的对应演变（右）。追踪了两个粒子：一个最初位于壁面附近（●），另一个位于区域中心（▼）。对于每个粒子，新算法（绿色）得到的结果与预期轨迹（橙色虚线）一致，而参考格式（蓝色）得到的结果则受到伪漂移效应的影响（导致粒子在外圆柱附近区域聚集）。

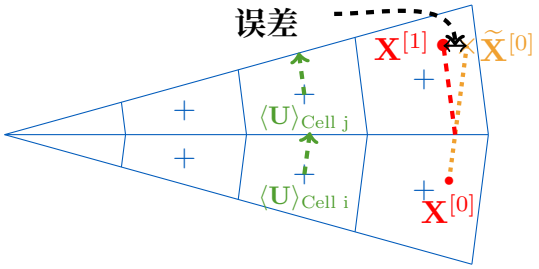



图11：示意图展示了不进行单元间积分的算法中数值误差的来源。随着时间推移，不进行单元间积分会导致粒子在外圆柱附近累积（见图12）。

**粒子浓度的结果** 现在我们将注意力转向大量流体粒子被跟踪的模拟。在这种情况下，我们关注的统计量是沿径向的粒子浓度。事实上，由于每个流体粒子都沿圆形轨道运动，粒子浓度应始终保持不变。基于这一结果，我们在此验证：只要流体粒子在初始时空间分布均匀，粒子浓度就会在空间中保持均匀。

数值模拟中，初始时有200,000个流体粒子均匀分布在空间中。随后，通过计算沿径向分布的21个单元中各自的粒子数浓度来分析浓度。图12展示了浓度随距离 $r$ 的变化，其中浓度已用初始浓度归一化。可以看出，当无量纲时间步长为 $\Delta t^+ \simeq 5.5$ 时，在整个模拟时段内，整个定义域中的浓度始终保持恒定。与此同时，参考算法（即不进行单元间积分）所得结果表明，粒子会趋于在外部区域累积，导致该区域的浓度随时间不断增加（较长时间后的稳态将表现为所有粒子都位于外圆柱处）。

To further assess the accuracy of the new algorithm, the error between the numerical value and the theoretical value of the number concentration has been computed within each of the 21 regions. The average error over the whole domain and simulation time (here 41 s) is displayed in Figure 13 as a function of the dimensionless time step $\Delta t^+$. It confirms that the new algorithm provides accurate results regardless of the time step used, i.e., both when the average displacement is smaller or greater than the grid size. The small (constant) error with the new algorithm comes from the statistical noise due to Monte Carlo methods. Meanwhile, the results obtained without cell-to-cell integration clearly show an increasing numerical error

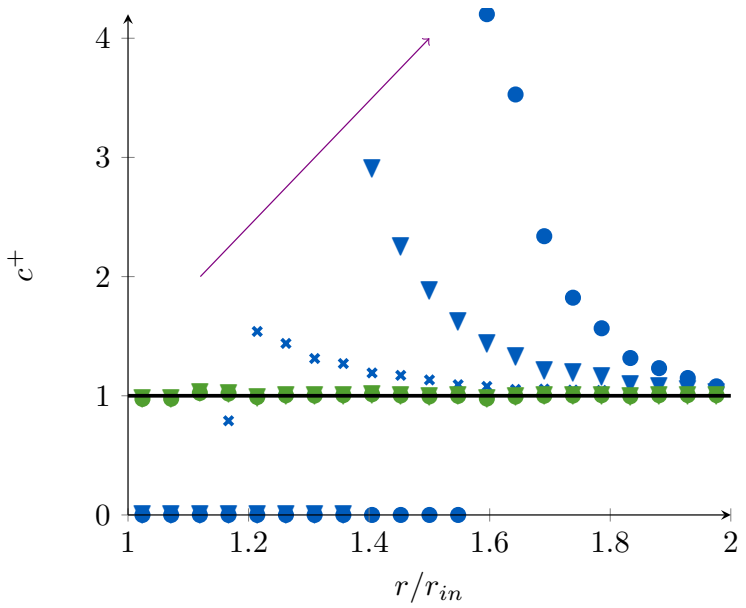



图12：无量纲浓度 $c^+$ （浓度以初始浓度归一化）随半径 $r$ 的变化。图中展示了过渡状态下多个模拟时间的结果： $t^+$=275 （×）、 $t^+$=1100 （▼）和 $t^+$=2200 （•）。时间步长保持恒定，使得 $\Delta t^+ \simeq 5.5$。新算法（绿色）得到的结果与解析结果（黄色）一致，而参考轨迹（蓝色）得到的结果则导致粒子向外圆柱聚集，且这种聚集随时间增加（箭头所示）。

with increasing time steps.

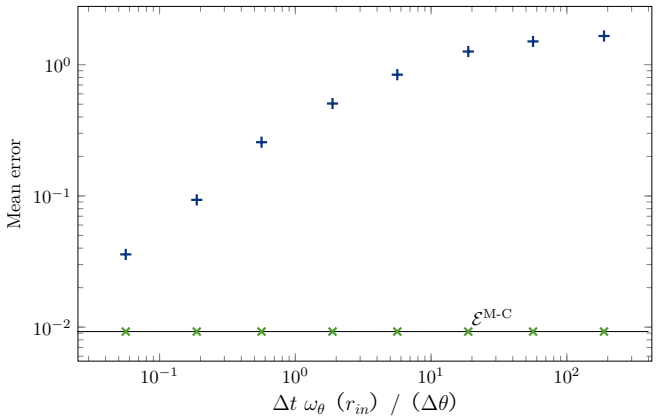



图13：使用参考算法（+）和新的逐单元算法（×）计算得到的粒子浓度平均误差，在时刻 $t$ =41s 随时间步长的变化。这些结果证实，参考算法会导致伪漂移（误差随时间步长增大而增加）；而新算法则始终保持准确，不受时间步长影响（恒定的非零误差源于蒙特卡罗方法固有的统计噪声）。

# 6 结论与展望

本文提出了一种新的逐单元算法，用于在将拉格朗日随机方法与平均场方法相耦合的混合方法框架下进行粒子跟踪。单元间积分将时间步长划分为若干个子迭代，每个子迭代对应粒子在一个单元内的运动。这意味着，每当粒子穿过一个面并离开一个单元时，其运动便会停止；此时将更新平均场，以计算粒子在该时间步长剩余时间内的粒子位移。这种分解方式能够自然地处理粒子每次进入新单元时平均场的变化。鉴于模型的随机性，还必须满足一个额外约束：在估计粒子在每个单元中的停留时间时，不得进行任何预先推断。实际上，一种朴素公式是先预测粒子在一个时间步长内的位移，然后再尝试推导其在所穿过单元中的停留时间。然而，这样得到的停留时间估计会成为驱动扩散项的维纳过程未来状态的函数，从而直接违背伊藤意义下随机积分的基本定义。为避免这一陷阱及其可能导致的虚假数值漂移和扩散值，必须仅依据每个子时间步开始时的数值进行谨慎估计。在当前算法中，这是通过引入虚拟伙伴的概念实现的：该虚拟伙伴的运动由条件均值增量控制，并用于在一个时间步长内穿过的不同单元中提供不存在统计偏差的停留时间估计。

基于这两个概念（逐单元积分和非预见性），新算法采用了一个三步流程。第一步是计算时间步长结束时粒子位置的确定性估计。然后，应用基于自由飞行假设的轨迹算法，以确定粒子是离开还是留在单元内。如果粒子留在单元内，则计算时间步长结束时粒子的位置和速度。否则，轨迹算法会提供离开时间和离开位置的信息。随后，从已知的最后一个离开位置开始，针对剩余时间步长再次执行整个三步流程。重复这一过程，直至粒子在时间步长结束时到达最终位置 $\Delta t$ （即粒子确实留在单元内）。

首先要分析新算法在恒定平均场下是否仍然精确。对于更一般的非均匀流，由于停留时间现在只能近似求得，因此时间积分格式不能视为精确；但可以保证整体格式遵循非预见性约束，并且研究表明，与当前用于非均匀流中粒子跟踪的算法相比，该格式有显著改进。通过考察两个测试案例中的数值预测，进一步验证了这一点：均匀平均流中的粒子扩散，以及非均匀流中的粒子动力学。这两个案例证实了新算法的精度有所提高，即使采用大时间步也不例外。

此处值得强调的是，尽管我们研究了流体粒子和一个简单的朗之万模型，但同一方法也可直接应用于分散型湍流两相流；更一般地说，还可应用于类似的随机动力学模型，这些模型需要在复杂几何结构和网格中追踪粒子。请注意，第5节所举的两个例子中，一个涉及层流，另一个涉及高雷诺数湍流。然而，本文的方法（即时间分裂算法）并不限于这些情形，只要采用恰当的动力学模型及其相应的数值格式，就可应用于任何雷诺数下的流动。此外，这种新算法为开发此类混合方法的精细算法铺平了道路，这些混合方法将拉格朗日方法与平均场方法相结合。在本文中，每当粒子进入新的单元时，我们都会进行一次单元间积分。另一方面，这种时间分裂格式也可以仅在相邻单元间的平均场差异显著时应用。因此，可以利用测试函数量化粒子轨迹上平均场量的变化，并且仅当变化超过给定阈值时才应用这种时间分裂方法，从而设计出一种精细算法。

# A 三维非结构网格粒子追踪算法的详细说明

本附录介绍本研究中使用的轨迹算法。该算法能够在带有扭曲面的三维完全非结构网格中追踪粒子的运动。附录 A.1 首先介绍追踪算法，其中包括原始追踪算法的说明，以及第 A.1.1 节中对穿面事件的检测。随后，在附录 A.1.2 中扩展该算法，以计算粒子穿过某个面时的位置和离开时间。最后，在附录 A.2 中，通过比较在简单非均匀流中使用各种三维非结构网格获得的数值结果，对该算法进行验证。

## A.1 邻居搜索算法的原理

该算法基于逐次邻居搜索。这意味着，通过遍历相邻单元，确定粒子当前所在的单元。此类算法需要三项信息：(a) 粒子的初始位置 $\mathbf{X}_O = \mathbf{X}^n$，(b) 粒子最初所在的对应单元，以及 (c) 粒子的目标位置 $\mathbf{X}_D = \mathbf{X}^{n+1}$（如图4所示）。其原理是，假设粒子在点 $\mathbf{X}_O$ 和$\mathbf{X}_D$之间做自由飞行运动，判断粒子是否离开当前单元。具体步骤如下：

1. 计算当前单元中哪些面与直线 ($\mathbf{X}_O\mathbf{X}_D$) 相交；
2. 检查交点是否位于直线向量 $\mathbf{X}_O\mathbf{X}_D = \mathbf{X}_D - \mathbf{X}$ 上 $_O$.

因此，关键问题是找到一种稳健的方法，检测位移向量与任意一个面之间的交点。最重要的是防止粒子在计算域中永久丢失。因此，该方法采用可在不同单元间重复执行的布尔基本测试，以处理病态情况，例如向量 $\mathbf{X}_O\mathbf{X}_D$ 穿过某个面的边（精确到机器精度）。

### A.1.1 检测穿过扭曲面的穿面事件的方法

检测穿面事件的方法基于将每个面分解为一组三角形子面（另见图17）。每个三角形子面由面的一条有向边（由两个相邻顶点 $\mathbf{X}_i$ 和 $\mathbf{X}_j$构成）以及该面的重心 $\mathbf{X}_f$构成。将面如此分解，可确保所处理的每个三角形子面都是平面的，因此即使对于扭曲网格（即顶点不在同一平面上的面），该方法也易于实施。

将一个面分解为一组三角形子面后，使用一种 M¨oller–Trumblemore 算法 [26]来检测向量与每个平面子面之间的交点。其原理是将每个子面的顶点投影到一个定向平面上，该平面垂直于位移向量 $\mathbf{X}_O\mathbf{X}_D$ ，并经过 $\mathbf{X}_O$ ，记作（$\mathbf{X}_O$，$\mathbf{X}_O\mathbf{X}_D^{\perp}$）。接着，要判断直线（$\mathbf{X}_O\mathbf{X}_D$）是否穿过子面，只需检查点 $\mathbf{X}_O$ 是否位于该子面在此平面上的投影所构成的三角形内（如图16所示，上标（.)$^{\dagger}$ 表示在垂直于位移方向的平面上的投影）。为此，我们采用一组三个基本布尔测试，每个测试都用于验证点 $\mathbf{X}_O$ 是否位于某条投影边的“正确一侧”。

**基本布尔测试**对于构成投影子面的三条边（即$\mathbf{X}_f^{\dagger}\mathbf{X}_i^{\dagger}$、$\mathbf{X}_f^{\dagger}\mathbf{X}_j^{\dagger}$和$\mathbf{X}_i^{\dagger}\mathbf{X}_j^{\dagger}$），我们必须验证点$\mathbf{X}_O$是否位于投影边的正确一侧。为此，我们在这里采用简单的逻辑测试。为清楚起见，考虑投影平面上连接两点$\mathbf{X}_\alpha\mathbf{X}_\beta$ 的一条有向边（其中，$_\alpha$ 和$_\beta$ 是子面的两个顶点的索引）。在这种情况下，逻辑测试$\mathcal{L}_{\alpha}^{\text{edge}}{}_{,\,\beta}$ 表示：

$$\mathcal{L}_{\alpha,\beta}^{\text{edge}} = \begin{cases} true & \text{if } (\mathbf{X}_\alpha\mathbf{X}_\beta \wedge \mathbf{X}_\alpha\mathbf{X}_O)\cdot\mathbf{X}_O\mathbf{X}_D > 0. \\ false & \text{otherwise.} \end{cases} \tag{31}$$

如图14所示，这个基本布尔测试可以判断点$\mathbf{X}_O$位于投影直线（$\mathbf{X}_\alpha^\dagger\mathbf{X}_\beta^\dagger$）左侧的半平面内（真）还是右侧的半平面内（假）。值得注意的是，由于该测试中的不等式是严格不等式，因此在有向直线上的行为是不对称的。事实上，如果点$\mathbf{X}_O$在机器精度下位于这条直线上，测试$\mathcal{L}_{\alpha,\beta}^{\text{edge}}$返回$false$。换言之，我们任意规定直线（$\mathbf{X}_\alpha^\dagger\mathbf{X}_\beta^\dagger$）属于闭右半平面，而不属于开左半平面。因此，布尔测试$\mathcal{L}_{\alpha,\beta}^{\text{edge}} \bigcup \text{not}(\mathcal{L}_{\alpha,\beta}^{\text{edge}})$构成定义域的一个划分。这意味着，对于给定的位移向量$\mathbf{X}_O\mathbf{X}_D$，只要顶点$\alpha$和$\beta$的排序固定，共享同一条边的两个面进行$\mathcal{L}_{\alpha,\beta}^{\text{edge}}$计算的结果就是可复现的。在本研究中，我们规定顶点按以下顺序排序：先$\mathbf{X}_f$，再$\mathbf{X}_i$，最后$\mathbf{X}_j$（其中$i<j$）。

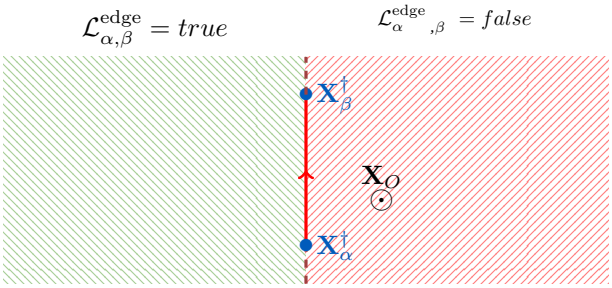



图14：示意图展示了基本布尔测试$\mathcal{L}_{\alpha,\beta}^{\text{edge}}$的工作原理：点（$\mathbf{X}_0$可以位于有向直线（$\mathbf{X}_\alpha^\dagger\mathbf{X}_\beta^\dagger$）的左侧或右侧。此处，图示为从平面（$\mathbf{X}_0$,$\mathbf{X}_O\mathbf{X}_D^\perp$）上方观察，这意味着基本布尔测试为$true$位于图形左侧。请注意，该直线属于闭半平面（即，$false$以红色表示）。

图14还表明，仅凭这一基本布尔测试不足以确定点$\mathbf{X}_O$是否属于投影三角形子面。事实上，根据向量$\mathbf{X}_i^\dagger\ \mathbf{X}_j^\dagger$的方向，以及该点位于平面上方还是下方，点可能位于直线的右侧或左侧。虽然借助上一段引入的排序后的顶点，向量的方向现已固定，但要处理第二个问题，我们还必须引入平面方向的概念。为此，我们依据子面的方向，并引入一个额外的逻辑测试，以判断位移向量$\mathbf{X}_O\mathbf{X}_D$是否与有向子面对齐。若对齐则返回 true，否则返回 false。该条件表示为：

$$\mathcal{A}_{f,i,j}^{\text{face}} = \begin{cases} true & \text{if } (\mathbf{X}_f\mathbf{X}_i \wedge \mathbf{X}_f\mathbf{X}_j)\cdot\mathbf{X}_O\mathbf{X}_D > 0. \\ false & \text{otherwise.} \end{cases} \tag{32}$$

通过结合两个逻辑测试$\left(\mathcal{L}_{i,j}^{\text{edge}} \equiv \mathcal{A}_{f,i,j}^{\text{face}}\right)$（其中$\equiv$表示两个布尔测试具有相同的值），我们能够确定粒子相对于面的方向位于直线$\mathbf{X}_i^\dagger\ \mathbf{X}_j^\dagger$的哪一侧。

**组合基本布尔测试** 为了确定交点$\mathbf{X}_i$是否位于投影面内，我们需要将这些基本逻辑测试结合起来。如图15所示，当满足三个条件时，交点属于投影面。这些条件取决于子面相对于位移向量的方向：

- 当位移向量与子面方向一致时（即$\mathcal{A}_{面f、i、j} = true$），如果按照排序后的顶点（即$\mathbf{X}_f^\dagger\mathbf{X}_i^\dagger$、$\mathbf{X}_i^\dagger\mathbf{X}_j^\dagger$和$\mathbf{X}_j^\dagger\mathbf{X}_f^\dagger$）进行遍历，则该点位于每条投影边的左侧。这意味着必须满足以下三个条件：第一，$\mathcal{L}_{边f、i} = true$；第二，$\mathcal{L}_{边f、j} = false$（由于其方向相反）；第三，$\mathcal{L}_{边i、j} = true$。这种情况显示在图15的左侧。

- When the displacement vector is not aligned with the sub-face orientation (i.e., $\mathcal{A}_{f,i,j}^{\text{face}} = false$), the point is located to the right of each of the projected edges provided that we follow the sorted vertices

[1]This is the case for instance for a sub-face seen from one cell or from the neighboring one.

(namely $\mathbf{X}_f^\dagger\mathbf{X}_i^\dagger$, $\mathbf{X}_i^\dagger\mathbf{X}_j^\dagger$ and $\mathbf{X}_j^\dagger\mathbf{X}_f^\dagger$). This means that the following three conditions have to be met: first, $\mathcal{L}^{\text{edge}}_{f,i} = false$; second, $\mathcal{L}^{\text{edge}}_{f,j} = true$ (due to its reverse orientation); third, $\mathcal{L}^{\text{edge}}_{i,j} = false$. This case is displayed in the RHS of Figure 15.

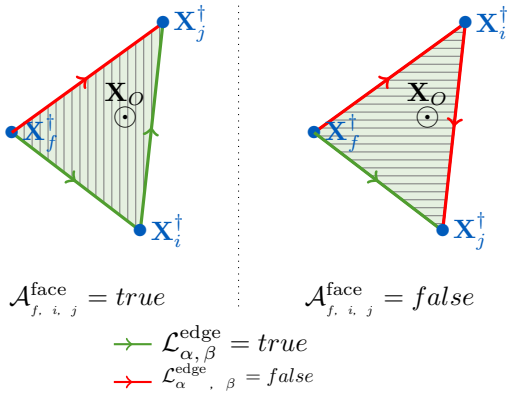



图15：用于测试对齐的布尔测试示意图 $\mathcal{A}^{\text{face}}_{f,i,j}$。该测试用于确定位移向量 $\mathbf{X}_O\mathbf{X}_D$ 是否与有向子面 $\mathbf{X}_f\mathbf{X}_i$ $\mathbf{X}_j$ 对齐（若对齐，则返回 $true$）。示意图展示了两种可能的情况：左侧中，位移向量与子面方向对齐；右侧中，二者未对齐。在每种情况下，如果点 $\mathbf{X}_O$ 位于所有有向边的正确一侧（即 $\mathbf{X}_f\mathbf{X}_i$、$\mathbf{X}_f\mathbf{X}_j$ 和 $\mathbf{X}_i\mathbf{X}_j$，其中 $i < j$），则该点位于投影子面内。逻辑测试 $\mathcal{L}^{\text{edge}}_{\alpha,\beta} = true$ 根据其结果显示（绿色 $= true$ 和红色 $= false$）。

总而言之，如果满足以下条件，则交点位于给定的子面内。

$$\begin{aligned} &\left(\mathcal{L}^{\text{edge}}_{i,j} \equiv \mathcal{A}^{\text{face}}_{f,i,j}\right) \\ \text{and}\quad &\left(\mathcal{L}^{\text{edge}}_{f,i} \equiv \mathcal{A}^{\text{face}}_{f,i,j}\right) \\ \text{and}\quad &\left(\text{not}(\mathcal{L}^{\text{edge}}_{f,j}) \equiv \mathcal{A}^{\text{face}}_{f,i,j}\right) = true. \end{aligned} \tag{33}$$

其中，符号 $\equiv$ 表示“等价”算子（即，当两个布尔变量的值相同时，该条件为真）。对每个子面都执行此项测试。该算法如图16所示（其中交点位于右侧面上）。

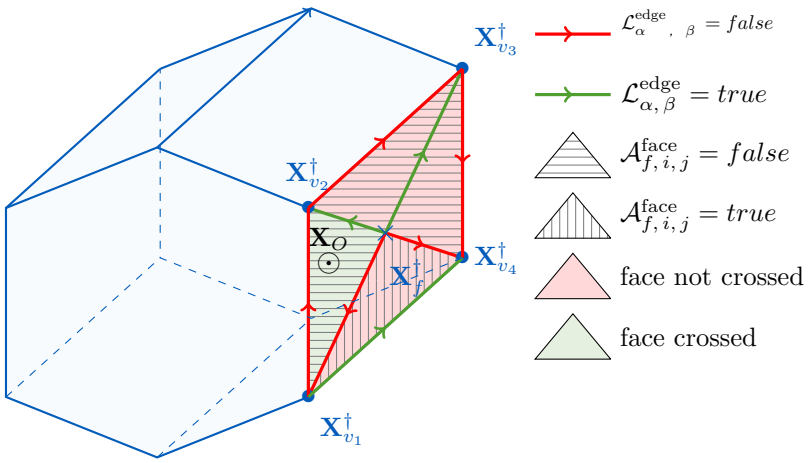



Figure 16: Sketch illustrating how the tracking algorithm determines if a particle displacement from $\mathbf{X}_O$ to $\mathbf{X}_D$ crosses a face. Here, the particle exits the cell through the face on the right. As a result, when the edges of this face are projected on the plane normal to the displacement vector (here with superscripts †), the various logical tests $\mathcal{L}^{\text{edge}}_{\alpha,\beta}$ and $\mathcal{A}^{\text{face}}_{f,i,j}$ confirm that one of the triangular sub-face is detected as an exit face (green color) while the three other sub-faces are not (red color).

对于扭曲面，直线（$\mathbf{X}_O\mathbf{X}_D$）可能多次穿过同一个面。如图17所示。在这种情况下，算法会监测直线（$\mathbf{X}_O\mathbf{X}_D$）穿过该面的次数。如果次数为偶数，则表示粒子仍处于当前单元内（它离开后又重新进入了该单元）。如果次数为奇数，则表示粒子通过此面离开该单元。

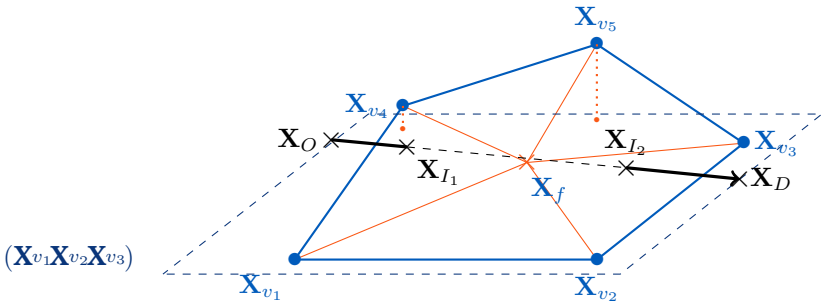



图17：示意粒子位移从 $\mathbf{X}_O$ 到 $\mathbf{X}_D$ 穿过一个扭曲面：可以看出，粒子通过 $\mathbf{X}_{I_1}$ 离开单元（该点属于子面（$\mathbf{X}_f, \mathbf{X}_{v_1}, \mathbf{X}_{v_4}$））， 并通过 $\mathbf{X}_{I_2}$ 重新进入单元（该点属于子面（$\mathbf{X}_f, \mathbf{X}_{v_2}, \mathbf{X}_{v_3}$））。当前算法会遍历所有子面，并统计每个子面被穿过的次数，因此能够自然地处理这种情况：偶数次表示粒子仍处于当前单元内，奇数次表示粒子离开当前单元。

### A.1.2 交叉时间和位置的估算方法

如第4.2.1节所述，新算法不仅需要包含粒子的单元信息，还需要交叉时间和位置的信息。这意味着必须扩展前文所述的轨迹算法，以提供这些信息。

确定直线（$\mathbf{X}_O\mathbf{X}_D$）确实穿过一个子面后，可以根据自由飞行假设估算到达该子面所需的相对时间 $\theta = t_{\text{cross}}/\Delta t$ 。结果为：

$$\theta = \frac{\mathbf{X}_O\mathbf{X}_f \cdot (\mathbf{X}_f\mathbf{X}_i \wedge \mathbf{X}_f\mathbf{X}_j)}{\mathbf{X}_O\mathbf{X}_D \cdot (\mathbf{X}_f\mathbf{X}_i \wedge \mathbf{X}_f\mathbf{X}_j)}. \tag{34}$$

由于我们仅在此前已确定直线确实穿过该面的情况下应用此方程，因此该方程定义良好。实际上， $\theta$ 的值还提供了额外信息。当 $\theta$ 为负时，这意味着交点是定向轴（$\mathbf{X}_O\mathbf{X}_D$）的进入点。当 $\theta$ 为正时，这意味着交点是离开点。随后统计定向轴（$\mathbf{X}_O\mathbf{X}_D$）进入（$n_{in}$）和离开（$n_{out}$）的子面数量。然后利用这些数量检查粒子是否确实位于正确的单元中。当直线（$\mathbf{X}_O\mathbf{X}_D$）进入该单元所穿过的子面数量等于其离开该单元所穿过的子面数量，且这两个数量均不为零（即 $n_{in} = n_{out} > 0$）时，粒子位于正确的单元中。

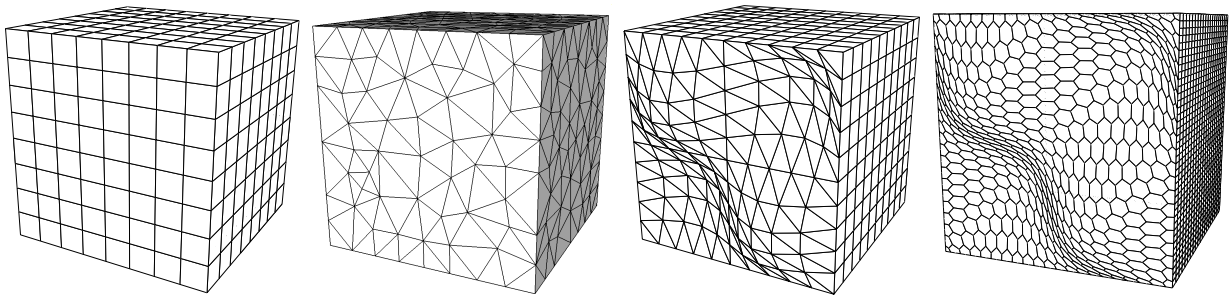
If the value of $\theta$ defined in Eq. (34) is in the interval $\theta \in [0, 1[$, the particle does actually cross a face during the time step. The position of the particle at the intersection is then simply given using a simple linear interpolation (this linear interpolation is coherent with the assumption of free-flight motion):

$$\mathbf{X}_I = \mathbf{X}_O + \theta \times \mathbf{X}_O\mathbf{X}_D. \tag{35}$$

When a particle leaves a cell after crossing one face several times (as in wrapped faces), the exit time is considered equal to the largest value in the range $[0, 1[$ (i.e., the last exit time).

## A.2 Validation on 3-D unstructured meshes

The trajectory algorithm has been tested using various meshes obtained from the FVCA6 benchmark test cases [7]. These meshes were selected to be representative of a range of different meshes, going from a



(A) Hexa mesh. (B) Tetra mesh. (C) PrT mesh. (D) PrG mesh.

Figure 18: Type of mesh used in the present case, which span a range of possible configurations (from simple Cartesian mesh without wrapped faces to highly distorted meshes with wrapped faces.

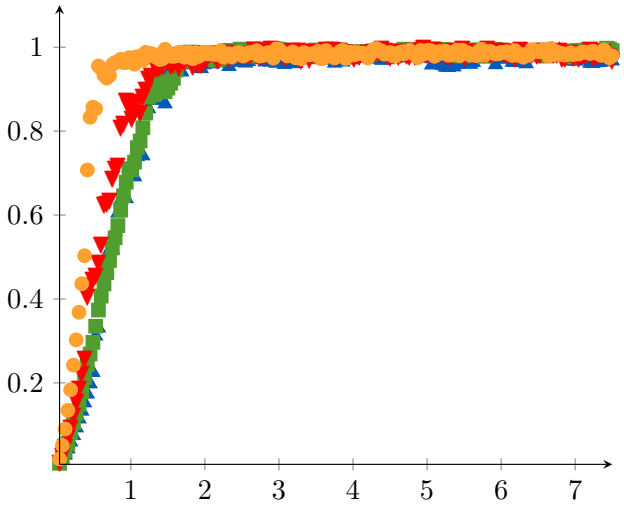



Figure 19: Maximum dimensionless distance between the particles and their center of gravity using the Hexa mesh (■), the Tetra mesh (▲), the PrT mesh (▼) and the PrG mesh (●. All maximum dimensionless distances are smaller or equal to 1, meaning that the particles stay within the physical domain.)

regular Cartesian mesh to a highly distorted mesh with different refinements. The four meshes used here are displayed in Figure 18.

用于验证的案例实际上对应于第5.1.1节所述的均匀流。它由均匀各向同性湍流中的点源弥散组成。为确保没有粒子丢失，我们跟踪粒子与单元中心（点源）之间的距离。为了得到一个不受网格影响、具有代表性的量，所跟踪的量是无量纲距离 $d^*$。其定义为 $d^* = \frac{\|\mathbf{x}_c - \mathbf{x}_p\|}{\max_{X \in \Omega_c} \|\mathbf{x}_c - \mathbf{x}\|}$，其中 $\mathbf{X}_c$ 为单元的重心，$\mathbf{X}_p$ 为粒子的位置，

而 $\Omega_c$ 为单元 $c$所定义的定义域。当粒子被正确追踪时，该距离始终小于 1。然而，如果粒子穿过的某个面被漏检，粒子就会永久丢失，因为它可能会不受限制地继续运动。 [2] 在这种情况下，粒子与单元中心之间的距离可能会发散，并变得远大于 1。

Results obtained with various meshes are compared using a time scale made dimensionless using the Lagrangian time scale $t^* = t/T_L$ (which is constant for all meshes considered). The results are displayed in Figure 19: we can see that, with 100 000 particles dispersed initially from the point source, the maximum distance to the cell center converges toward 1 but remains always smaller than unity. This proves that the current algorithm is tractable even for 3-D unstructured meshes.

As in Section 5.1, we can also analyze the results obtained for the different second order moments. We focus here on verifying that the results are consistent regardless of the mesh used using a given time step (the role of the time step has been detailed in Section 5.1). The results are displayed in Figure 20: it can be

[2] If the current cell associated to a particle is not correct, the algorithm would not detect other face-crossing events.

seen that all numerical results match the analytical values regardless of the mesh used in such cases. This confirms the accuracy of the algorithm even when 3-D unstructured meshes are used. At this stage, it is also worth noting that 1-D simulation provide the same results as 3-D ones since each of the three directions can be treated independently of the other ones (this is actually a typical characteristic of homogeneous isotropic turbulence).

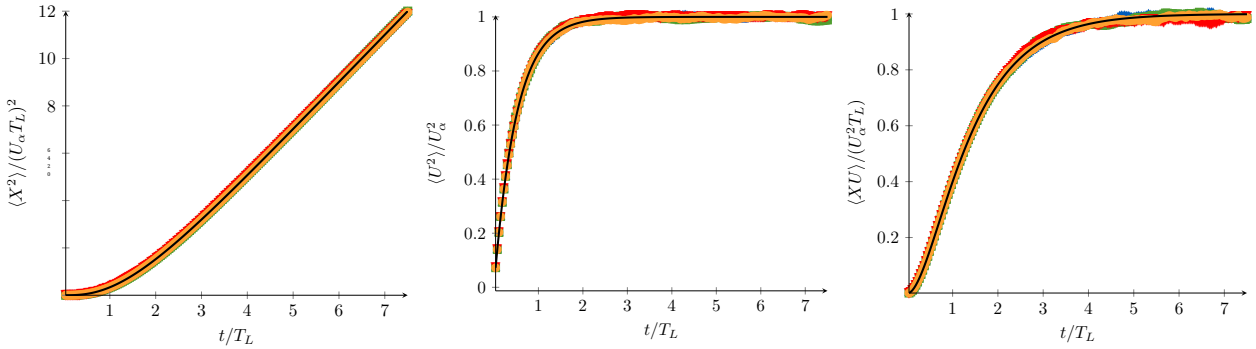



图20：弹道极限情况下点源弥散的 $\langle XX\rangle$, $\langle UU\rangle$,$\langle XU\rangle$ 演变。比较解析解（黑色直线）与采用新算法在不同网格上得到的数值结果。无论使用何种网格，数值结果都与解析解一致。

## 致谢

G. Balvet通过ANRT获得了EDF-CIFRE合同（编号2020/1387）提供的资金支持。作者感谢EDF R&D和CEREA实验室提供基础设施，使其能够使用计算资源。

## References

[1] F. Archambeau、N. M´echitoua 和 M. Sakiz。Code Saturne：用于计算湍流不可压流的有限体积代码——工业应用。《有限体积国际期刊》，1(1)，2004 年。

[2] M. L. Bahlali、C. Henry 和 B. Carissimo。关于弥散的混合欧拉-拉格朗日随机模型中的充分混合条件和一致性问题。《边界层气象学》，174(2)：275–296，2020 年。

[3] C. E. Brennen 和 C. E. Brennen。多相流基础。剑桥大学出版社，2005 年。

[4] S. Chibbaro 和 J.-P. Minier。关于混合欧拉-拉格朗日方法处理多相流的一致性说明。《国际多相流杂志》，37(3)：293–297，2011 年。

[5] R. Clift、J. R. Grace 和 M. E. Weber。《气泡、液滴与粒子》。多佛土木与机械工程系列。多佛出版社，2005 年。

[6] T. D. Dreeben 和 S. B. Pope。湍流中概率密度函数方法的壁面函数处理。《流体物理学》，9(9)：2692–2703，1997 年。

[7] R. Eymard、G. Henry、R. Herbin、F. Hubert、R. Kl¨ofkorn 和 G. Manzini。一般网格上各向异性扩散问题离散化格式的三维基准测试。载于《复杂应用的有限体积方法 *VI*：问题与展望》，第 895–930 页。Springer，2011 年。

[8] C. W. Gardiner。《随机方法手册》，第 3 卷。Springer 柏林，1985 年。

[9] D. Haworth 和 S. Pope。自相似湍流自由剪切流的概率密度函数建模研究。《流体物理学》，30(4):1026–1044，1987。

[10] D. C. Haworth。湍流反应流概率密度函数方法的进展。能源与燃烧科学进展，36(2):168–259，2010。[11] D. C. Haworth 和 S. B. Pope。湍流的广义朗之万模型。《流体物理学》，29(2):387–405，1986。[12] R. W. Hockney 和 J. W. Eastwood。使用粒子的计算机模拟。CRC出版社，1988。[13] A. Innocenti、R. O. Fox 和 S. Chibbaro。密集区间湍流含颗粒通道流的拉格朗日概率密度函数模型。《流体物理学》，33(5):053308，2021。[14] A. Innocenti、N. Mordant、N. Stelzenmuller 和 S. Chibbaro。壁面约束湍流中加速度的拉格朗日随机建模。流体力学杂志，892，2020。[15] R. L¨ohner 和 J. Ambrosiano。用于非结构网格的向量化粒子追踪器。《计算物理学杂志》，91(1):22–31，1990。[16] J.-P. Minier。关于湍流多分散两相反应流的拉格朗日随机方法。能源与燃烧科学进展，50:1–62，2015。[17] J.-P. Minier。多分散湍流两相流的统计描述。物理学报告，665:1–122，2016。[18] J.-P. Minier。为湍流分散两相流中的粒子动力学构建一致概率密度函数模型的方法。《流体物理学》，33(2):023312，2021。[19] J.-P. Minier、R. Cao 和 S. Pope。对 Li 和 Modest 所著文章“混合有限体积/概率密度函数蒙特卡罗方法中结构/非结构网格上的有效粒子追踪方案”的评论。《计算物理学杂志》，186(1):356–358，2003。[20] J.-P. Minier、S. Chibbaro 和 S. B. Pope。单相和分散两相湍流粒子模拟的拉格朗日随机模型构建指南。《流体物理学》，26(11):113303，2014。[21] J.-P. Minier 和 E. Peirano。湍流多分散两相流的概率密度函数方法。物理学报告，352(1-3):1–214，2001。[22] J.-P. Minier、E. Peirano 和 S. Chibbaro。拉格朗日两相流建模中的随机微分方程所用的弱一阶和二阶数值格式。蒙特卡罗方法与应用，9(2):93–133，2003。[23] J.-P. Minier、E. Peirano 和 S. Chibbaro。基于朗之万方程的多分散两相流概率密度函数模型及其在钝体气固流中的应用。《流体物理学》，16(7):2419–2431，2004。[24] J.-P. Minier 和 J. Pozorski。基于统计物理学原理推导湍流概率密度函数模型。《流体物理学》，9(6):1748–1753，1997。[25] J.-P. Minier 和 J. Pozorski。概率密度函数方法中的壁面边界条件及其在湍流通道流中的应用。《流体物理学》，11(9):2632–2644，1999。[26] T. M¨oller 和 B. Trumbore。快速、最小存储的射线-三角形求交。图形工具杂志，2(1):21–28，1997。[27] M. Muradoglu 和 S. Pope。求解概率密度函数湍流模型方程的局部时间步进算法。AIAA杂志，40(9):1755–1763，2002。.[28] H. Ottinger。聚合物流体中的随机过程：开发模拟算法的工具与示例。施普林格柏林海德堡，1996。[29] E. Peirano、S. Chibbaro、J. Pozorski 和 J.-P. Minier。多分散湍流两相流的平均场/概率密度函数数值方法。能源与燃烧科学进展，32(3):315–371，2006。[30] E. Peirano 和 J.-P. Minier。多分散湍流两相流的概率形式体系与模型层级。《物理评论E》，65(4):046301，2002。

[31] S. Pope。速度-耗散概率密度函数模型在非均匀湍流中的应用。《流体物理学*A*：流体动力学》，3(8):1947–1957，1991。[32] S. Pope。《湍流》。剑桥大学出版社，2000。[33] S. Pope 和 Y. Chen。湍流中的速度-耗散概率密度函数模型。《流体物理学*A*：流体动力学》，2(8):1437–1449，1990。[34] S. B. Pope。湍流的拉格朗日概率密度函数方法。《流体力学年度评论》，26(1):23–63，1994。[35] S. B. Pope。关于湍流随机拉格朗日模型与二阶矩闭合之间关系的研究。《流体物理学》，6(2):973–985，1994。[36] P. P. Popov、R. McDermott 和 S. B. Pope。用于粒子跟踪的高精度时间推进算法。《计算物理学杂志》，227(20):8792–8806，2008。[37] J. Pozorski 和 J.-P. Minier。基于朗之万方程的拉格朗日湍流弥散模型研究。《国际多相流杂志》，24(6):913–945，1998。[38] K. K. Sabelfeld 和 N. A. Simonov。《随机场与随机拉格朗日模型：湍流和多孔介质中的分析与应用》。Walter de Gruyter，2012。[39] D. P. Schmidt 和 C. Rutland。新的液滴碰撞算法。《计算物理学杂志》，164(1):62–80，2000。[40] H. Sigurgeirsson、A. Stuart 和 W.-L. Wan。考虑碰撞的粒子-场模拟算法。《计算物理学杂志》，172(2):766–807，2001。[41] S. Subramaniam 和 D. Haworth。在三维非结构变形网格上用于湍流混合与燃烧的概率密度函数方法。《发动机研究国际期刊》，1(2):171–190，2000。[42] J. Xu 和 S. Pope。湍流反应流中概率密度函数/蒙特卡罗方法数值精度的评估。《计算物理学杂志》，152(1):192–230，1999。