Adaptive Mesh Refinement Design

Wenchong Huang

School of Mathematical Sciences,
Zhejiang University.

Mar. 8th, 2024

研究背景

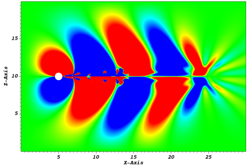


图 1: 在密度分层流体中小球拖曳造成的密度扰动.

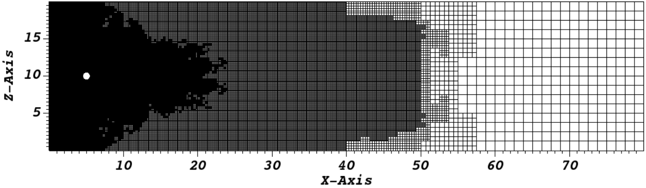


图 2: Li 等人在捕捉小球拖曳尾涡时使用的网格.

捕捉尾涡要非常细的网格，但全局用这么细的网格跑不动，加并行也没用。组里的代码目前不支持自适应，我们要解决这个问题。

AMR 简介

上世纪 90 年代, 美国 Lawrence Berkeley 国家实验室开发了一套支持基于块状加密的 AMR 网格与并行计算的基础软件设施库 Chombo[1,2] .

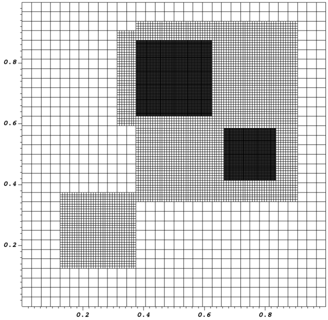


图 3: 基于块状加密的 AMR 示意图.

用户可以选择物理区域中的若干个块状区域进行加密, 每一个加密块在内存中都是连续的. 用户还可以自定义加密率.

AMR 简介

上世纪 90 年代, 美国 Lawrence Berkeley 国家实验室开发了一套支持基于块状加密的 AMR 网格与并行计算的基础软件设施库 Chombo[1,2] .

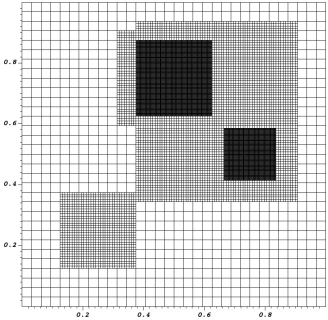


图 3: 基于块状加密的 AMR 示意图.

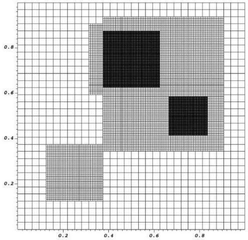
用户可以选择物理区域中的若干个块状区域进行加密, 每一个加密块在内存中都是连续的. 用户还可以自定义加密率.

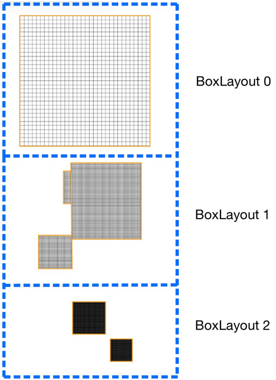
但是, Chombo 的代码实现过于老旧, 不符合现代编程理念. 另外, 它将不规则边界当成一段一段的线段,[3] 这导致它在不规则区域中不能做到高阶精度.

AMR 底层设计——存储加密区域的布局

组内程序： 用一个 Box 存储一个矩形区域.

我们的设想： 用一个 BoxLayout 封装同一层级的的所有块状加密区域，每个区域是一个 Box，再用 std::vector<BoxLayout> 存储整个区域.





需要解决的问题： BoxLayout 内部用什么数据结构存储多个 Box 更高效？当我要找一个网格时，需要快速知道它在哪个 Box 里，如果每次找都要遍历一个层级里的所有 Box，

组内程序：用一个 `Tensor` 存储一个矩形区域中的所有数据.

我们的设想：用一个 `LevelData` 封装同一层级的的所有数据，`LevelData` 内有多个 `Tensor`，`Tensor` 与 `Box` 一一对应。再用 `AllLevelData` 存储全部的数据.

`LevelData` 内部用数组存储多个 `Tensor`，当要找某个网格数据时，先通过 `BoxLayout` 的成员方法定位到 `Box` 编号然后在对应的 `Tensor` 里找就行。

AMR 底层设计——存储算子

组内程序： 用一个 `SpatialOperator` 的派生类存储算子，计算时先全部按标准格式计算，然后左乘一个稀疏矩阵来校正不规则格点．稀疏矩阵在算子的构造函数中预处理．

不能直接推广的原因： `LevelData` 并不是 `Tensor`， 无法左乘稀疏矩阵，而且算子的离散格式会用到不同层级的数据．

我们的设想： 设计一个 `FillGhost` 类， 它将每个 `Box` 向外延拓两格， 并且填充 Ghost Cell．然后算子全部用标准格式计算．`SpatialOperator` 及其派生类在架构层面不需要改太多．

AMR 中的高阶虚拟单元填充方法

要将 AMR 推广到高阶精度，首先需要解决粗细网格交界处的虚拟单元填充问题．我们参考 Zhang 的工作[4]．以下图所示简单的二维情形为例，白色是粗网格区域，灰色是细网格区域，对于黑色三角形所示的细网格虚拟单元，我们可以用圆点所示的粗网格积分平均来拟合一个四阶多项式（当圆点落在灰色区域内部时，指的是对应四个细网格积分平均的平均值），从而计算细网格虚拟单元的取值．

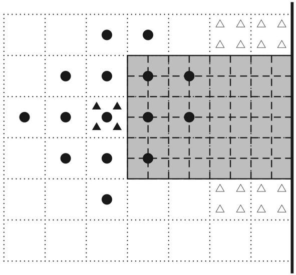


图 4: 粗细网格交界处虚拟单元的填充示意图[4]

不规则边界的处理

在非规则边界附近填充虚拟单元时，需要拟合一个高阶多项式．具体而言，我们知道一些点 $\mathbf{x}_1, \ldots, \mathbf{x}_n$，以及这些点处的函数值 $y_1, \ldots, y_n$．我们需要确定一个二元多项式 $p \in \mathbb{P}_k[x, y]$，使得

$$\sum_{i=1}^{n} |p(\mathbf{x}_i) - y_i|^2 \tag{1}$$

取得最小值．这是一个加权最小二乘问题，与一维情形很不一样，要想使得该问题有唯一解，并不能随意取 $\binom{k+2}{2}$ 个点，而应该按照某种规则来选择点．另外，这些点必须选在网格点上，在不规则边界附近亦是如此，这使得这个问题更加复杂．适定格点生成算法 (Poised Lattice Generation, PLG)[5] 解决的就是如何选择这些点的问题．

我们需要设计一种基于殷集表示[6] 与 PLG 算法的不规则边界处理方法．文章[7]已经做了相关工作，我们需要让它能够支持并行计算与自适应网格．

预期目标与进度安排

中期检查之前，完成自适应网格底层的设计，实现虚拟单元的填充，使程序能够根据任一场函数 $\mathbf{u}$ 在网格单元的积分平均来计算出 $\frac{\partial u_i}{\partial x_i}$、$\nabla \cdot \mathbf{u}$、$\Delta\mathbf{u}$ 在网格单元的积分平均.

预期目标与进度安排

中期检查之前，完成自适应网格底层的设计，实现虚拟单元的填充，使程序能够根据任一场函数 $\mathbf{u}$ 在网格单元的积分平均来计算出 $\frac{\partial u_i}{\partial x_i}$、$\nabla \cdot \mathbf{u}$、$\Delta\mathbf{u}$ 在网格单元的积分平均.

毕设结题之前完成 Dirichlet 边界条件、Neumann 边界条件下椭圆方程的四阶有限体积并行自适应求解器，能够支持任意复杂的不规则区域. 进行充分的数值实验，并验证四阶收敛性.

预期目标与进度安排

中期检查之前，完成自适应网格底层的设计，实现虚拟单元的填充，使程序能够根据任一场函数 $\mathbf{u}$ 在网格单元的积分平均来计算出 $\frac{\partial u_i}{\partial x_i}$、$\nabla \cdot \mathbf{u}$、$\Delta \mathbf{u}$ 在网格单元的积分平均.

毕设结题之前完成 Dirichlet 边界条件、Neumann 边界条件下椭圆方程的四阶有限体积并行自适应求解器，能够支持任意复杂的不规则区域. 进行充分的数值实验，并验证四阶收敛性.

未来一年内完成无穿透、无滑移边界条件下，二维不可压 Navier-Stokes 方程的四阶有限体积投影方法求解器，能够支持任意复杂的物理区域，能够支持自适应网格与 CPU 并行计算. 实现代码并测试卡门涡街等具有代表性的算例，对时空一致四阶收敛性进行数值验证.

[1] P. Colella, D. Graves, T. Ligocki, D. Martin, D. Modiano, D. Serafini, and B. Straalen, "Chombo software package for amr applications design document," tech. rep., 01 2009.

[2] P. Colella, D. Graves, T. Ligocki, D. Martin, and B. Straalen, "Amr godunov unsplit algorithm and implementation," tech. rep., 01 2003.

[3] P. Colella, D. Graves, T. Ligocki, D. Modiano, and B. Straalen, "Ebchombo software package for cartesian grid, embedded boundary applications," tech. rep., 02 2000.

[4] Q. Zhang, "High-order, multidimensional, and conservative coarse–fine interpolation for adaptive mesh refinement," *Computer Methods in Applied Mechanics and Engineering - COMPUT METHOD APPL MECH ENG*, vol. 200, pp. 3159–3168, 10 2011.

[5] Q. Zhang and Z. Li, "Poised lattice generation in $\mathbb{Z}^d$ for multivariate polynomial interpolation." 2022.

[6] Q. Zhang and Z. Li, "Boolean algebra of two-dimensional continua with arbitrarily complex topology," *Mathematics of Computation*, vol. 89, pp. 2333–2364, 05 2020.

[7] 黎至轩, "二维不规则区域上不可压 navier-stokes 方程的四阶有限体积方法," Master's thesis, 浙江大学, Apr. 2021.

[8] K. Devlin, "The millennium problems: The seven greatest unsolved mathematical puzzles of our time," 01 2002.

[9] Q. Zhang, "Gepup: Generic projection and unconstrained ppe for fourth-order solutions of the incompressible navier–stokes equations with no-slip boundary conditions," *Journal of Scientific Computing*, vol. 67, pp. 1134–1180, 06 2016.

[10] L. Chen, "iFEM: An integrated finite element methods package in MATLAB," tech. rep., 2009.

[11] D. Arndt, W. Bangerth, M. Bergbauer, M. Feder, M. Fehling, J. Heinz, T. Heister, L. Heltai, M. Kronbichler, M. Maier, P. Munch, J.-P. Pelteret, B. Turcksin, D. Wells, and S. Zampini, "The `deal.II` library, version 9.5," *Journal of Numerical Mathematics*, vol. 31, no. 3, pp. 231–246, 2023.

[12] Q. Zhang, H. Johansen, and P. Colella, "A fourth-order accurate finite-volume method with structured adaptive mesh refinement for solving the advection-diffusion equation," *SIAM Journal on Scientific Computing*, vol. 34, pp. B179–B201, 04 2012.

Thank You