

Лабораторная работа № 6. Метод частиц в ячейках

1 апреля 2019 г.

1 Краткое описание метода

Основываясь на книге (Вшивков В.А., Григорьев Ю.Н., «Численные методы "частицы-в-ячейках"») приведем описание основной идеи метода. Пусть задача записана в абстрактной операторной форме:

$$\frac{\partial q}{\partial t} + Aq = 0 \quad (1)$$

Здесь $q(t, u)$ - вектор-функция со значениями в R^n , u – вектор независимых переменных с областью изменения в R^m , $0 < t < \infty$.

В реальных задачах роль уравнения 1 выполняет кинетическое уравнение (уравнение Власова или Больцмана) или уравнения гидродинамики. Решение задачи 1 представляется в виде следующей интерполяционной формулы:

$$q = \sum_{j=1}^N Q_j R(u, u_j(t)) \quad (2)$$

Этот переход называют разбиением среды на модельные частицы. Функция $R(u, v)$ называется ядром (или форм-фактором) модельной частицы. Эта функция описывает распределение некоторого признака (массы, заряда, скорости, ...) в рамках одной частицы. Далее, если представить функцию q в виде:

$$q = \sum_{j=1}^N Q_j R(r, r_j(t)) \delta(p - p_j(t)) \quad (3)$$

Где r – радиус-вектор частицы, p – импульс частицы, то можно показать, что решение уравнения 1 тождественно решению следующей динамической системы:

$$\frac{\partial r_j}{\partial t} = p_j, \dots \frac{\partial p_j}{\partial t} = F(r_j, p_j) \quad (4)$$

Здесь F – вектор обобщенного поля.

Исключительно важно отметить, что переход к модельным частицам не означает замены реальной физической системы, где число частиц – молекул, атомов, ионов порядка числа Авогадро ($N_A = 6.02 \times 10^{23}$) на некую упрощенную систему из значительно меньшего количества таких же частиц, но «более крупного размера» (даже для самых больших расчетов на суперЭВМ N на данный момент не превышает 10^{12}). Система уравнений 4 является не более чем математическим формализмом для решения уравнения 1, т.е. на данном этапе никакого нарушения математической строгости не происходит, система уравнений 4 точно так же описывает физический процесс, как и исходное уравнение 1.

2 Математическая постановка задачи. Одномерная электростатическая задача

Далее рассмотрим наиболее простую математическую постановку для метода частиц в ячейках, а именно одномерную электростатическую задачу, т.е. на некотором отрезке от 0 до L_x движутся заряженные частицы плазмы – электроны с зарядом -1 и массой 1 и ионы с зарядом 1 и массой M (в единицах массы электрона, для водорода $M = 1836$). То, что набор заряженных частиц в данном случае именуется плазмой, означает, что средняя плотность положительных и отрицательных частиц, как и полное количество модельных частиц того и другого сорта, равны. Пусть количество модельных частиц будет равно N и пусть в пространстве, где они находятся, введена равномерная сетка с количеством узлов N_x . Шаг сетки:

$$h_x = L_x / N_x$$

Временная эволюция описанной выше системы частиц описывается

системой уравнений Власова-Пуассона:

$$\begin{aligned}\frac{\partial f_{i,e}}{\partial t} + v \frac{\partial f_{i,e}}{\partial r} - F_{i,e} \frac{\partial f_{i,e}}{\partial p} &= 0, \\ F_{i,e} &= q_{i,e} E, E = \nabla \varphi \\ \Delta \varphi &= \rho\end{aligned}\tag{5}$$

Здесь $f_{i,e}(x_k, v)$ – функция распределения отдельно ионов и электронов, имеющая смысл количества частиц данного сорта с определенным значением скорости v в данной ячейке сетки с номером k , F – сила, действующая на частицы, E – электрическое поле, φ – потенциал электрического поля. Поскольку рассматривается одномерная задача, то все векторные величины могут быть записаны как скалярные. Плотность заряда $\rho(x_k)$ вычисляется суммированием по частицам:

$$\rho(x_k) = \sum_{j=1}^N q_j R(x_i, u_j(t)), k = 1 \dots N_x$$

Конкретный вид форм-фактора частиц будет рассмотрен ниже. Здесь

$$x_k = k * h_x, \quad k = 1, \dots, N_x$$

2.1 Алгоритм вычислений для метода частиц в ячейках

Завершая математическую постановку задачи, необходимо описать последовательность вычислений:

1. Задается распределение частиц – начальные значения координат и скоростей: самый простой начальный вариант: распределение частиц по пространству равномерное, скорости все нулевые
2. Далее на каждом временном шаге:
 - (a) Вычисление плотности заряда по частицам
 - (b) Вычисление потенциала поля с помощью решения уравнения Пуассона
 - (c) Вычисление значений электрического поля в каждой ячейке

(d) Расчет движения модельных частиц

Обычно наиболее затратным (до 90 % времени) является шаг 2-d, расчет движения модельных частиц, соответственно, основные усилия по оптимизации должны быть направлены именно на этот этап алгоритма.

2.2 Задание 1

Напишите программу, реализующую начальное распределение частиц. Для размещения координат и скоростей частиц используйте одномерные массивы одинарной точности, число частиц, $N = 1000$. Результатом распечатанное равномерное распределение координат частиц.

3 Методы интегрирования уравнений движения модельных частиц

Уравнения движения модельных частиц в приведенной выше математической постановке по виду совпадают с обычными уравнениями движения заряженных частиц в электрическом поле:

$$\frac{\partial v}{\partial t} = \frac{q}{m} E, \quad \frac{\partial r}{\partial t} = v \quad (6)$$

Наиболее простым (и тем не менее, часто используемым в реальных расчетах) методом решения этих уравнений является схема с перешагиванием (англ. leapfrog scheme):

$$\frac{r^{n+1} - r^n}{\tau} = v^{n+1/2}, v^{n+1/2} = \frac{q}{m} E^{n+1/2} \quad (7)$$

Здесь n обозначает временной слой, так что фактическое время $t = n\tau$, где τ - временной шаг, m - масса модельной частицы. Электрическое поле здесь должно быть взято в месте расположения частицы, оно вычисляется интерполяцией значений поля в двух соседних узлах:

$$E(x) = \frac{x - x_{k-1}}{h} E_k + \frac{x - x_k}{h} E_{k-1}, x_{k-1} < x < x_k \quad (8)$$

Напомним, что заряд q в безразмерном варианте может быть либо 1, либо -1, масса m равна 1 для электронов и 1836 для ионов водорода (в тестовых расчетах часто используют частицы меньшей массы, например $m = 10$ или $m = 100$).

3.1 Форм-факторы метода частиц

Важнейшим этапом метода частиц в ячейках является вычисление плотности заряда на сетке на основе координат модельных частиц, т.е. передача данных о распределении заряда в области с частиц на сетку. Для этого необходимо придать конкретный вид функции R – форм-фактору модельной частицы. С точки зрения реализации на GPU речь идет прежде всего о том, между каким количеством узлов сетки распределяется заряд частицы (1, 2 или более), и соответственно, будут или не будут возникать конфликты при записи в узлы сетки, как это будет показано далее.

Теоретически, чем больше узлов используется для распределения заряда частицы, как говорят, чем выше порядок форм-фактора, тем более гладким получается распределение плотности и вычисленное по нему поле, и все решение в целом, но на практике это не всегда так. Поэтому основным инструментом для повышения качества (гладкости, точности и т.д.) решения является повышение количества модельных частиц. Известна оценка уровня численных шумов κ (в процентах) от среднего количества модельных частиц в ячейке сетки N_{cell} :

$$\kappa \propto \frac{1}{\sqrt{N_{cell}}},$$

Таким образом, 100 модельных частиц в ячейке дает уровень шумов порядка 10 %.

4 Форм-фактор NGP

Форм-фактор NGP (англ. Nearest Grid Point, ближайший узел сетки). Означает, что заряд всех частиц суммируется в ближайшем к ним узле сетки:

$$\rho(x_k) = \frac{1}{N} \sum_{j=1}^N q_j, x_{k-1/2} < x_j < x_{k+1/2}, k = 1, \dots, N_x \quad (9)$$

множитель $1/N$ вводится для того, чтобы при увеличении количества модельных частиц значение плотности в узлах не возрастало.

4.1 Задание 2.

Реализуйте расчет движения и вычисление плотности по частицам, храня их в виде одного массива координат и одного массива скоростей, используйте форм-фактор NGP. Результат задания – время выполнения этой операции (движения и вычисление плотности) для всех частиц. Измерение времени проводите с помощью функции `gettimeofday`. Так как вычисление поля еще не рассматривалось, при выполнении заданий этого раздела задайте электрическое поле в виде синуса: $E(x) = \sin(2\pi/N_x * x)$. Результат - динамика движения частиц в течение нескольких десятков (напр. 50) временных шагов. Выводите

- координаты частиц по шагам в файлах (один шаг - один файл)
- значения плотности во всех узлах сетки

и используйте в дальнейшем эти файлы для контроля правильности параллельного счета. Также необходимо визуализировать динамику, например, с помощью программы `gnuplot`, или Excel, etc.

4.2 Задание 3.

Реализуйте параллельный расчет движения и вычисление плотности по частицам с использованием MPI, разделяя частицы между MPI-процессами по номеру в массиве (например, для двух процессов, 0-я частица - 0й процесс, 1-я - 1-й процесс, 2 - 0-й процесс и т.д.). Для выполнения этого задания используйте большое число частиц (более 1 млн.). Ожидаемый результат - ускорение, линейное по числу процессов.

4.3 Задание 4.

Реализуйте параллельный расчет движения и вычисление плотности по частицам с использованием MPI, разделяя частицы между MPI-процессами по координате (, т.е. методом разделения области на части, как это делалось в лабораторной работе 4). При этом необходимо обратить внимание на правильный расчет плотности в узлах сетки, которые находятся на границе раздела по процессорам. Для выполнения этого задания используйте большое число частиц (более 1 млн.). Ожидаемый результат - ускорение, линейное по числу процессов (сравнить с предыдущим).