

Задание практикума: Параллельная реализация решения уравнения теплопроводности на плоскости

Сальников А.Н.

1 Общее описание

Требуется написать параллельную программу с применением технологий MPI и OpenMP, которая позволяет решать задачу Коши[1] для уравнения теплопроводности в двумерной прямоугольной области. (Описанно формулами (1).)

$$\begin{cases} \frac{\partial u(x,y,t)}{\partial t} = a^2 \left(\frac{\partial^2 u(x,y)}{\partial x^2} + \frac{\partial^2 u(x,y)}{\partial y^2} \right) + heat(x,y,t) \\ u(x,y,t_0) = init(x,y) \\ u(x,y,t)|_{x,y \in G} = border(x,y,t) \end{cases} \quad (1)$$

Данная задача, заменяется разностной задачей на сетке (2).

$$\begin{aligned} & \frac{u(x,y,t+\Delta t) - u(x,y,t)}{\Delta t} = \\ & = a^2 \left(\frac{u(x,y,t) - 2 \cdot u(x+\Delta x,y,t) + u(x+2 \cdot \Delta x,y,t)}{\Delta x^2} + \frac{u(x,y,t) - 2 \cdot u(x,y+\Delta y,t) + u(x,y+2 \cdot \Delta y,t)}{\Delta y^2} \right) + \\ & + heat(x,y,t) \end{aligned} \quad (2)$$

Предполагается, что область делится на набор равномерных отрезков по каждому изменению с шагами Δx и Δy . Мы знаем состояние в начальный момент времени, задаётся функцией $init(x,y)$, которую можно вычислить дискретно, то есть задать начальное состояние системы матрицей. Нам известен закон поведения границы области во времени. Это означает, что в тот момент, когда мы будем подходить к границе в программе, для точек границы необходимо будет вызвать функцию $border(x,y,t)$, внутри которой значение можно будет вычислить по некоторому непрерывному закону.

2 Параметры передаваемые программному коду

Для обеспечения моделирования распространения тепла программе необходимо передать следующие параметры.

- шаг по измерению X – собственно Δx .
- шаг по измерению Y – собственно Δy .
- шаг по времени – собственно Δt .

- **промежуток сброса в файл** – через какие промежутки времени в терминах времени модели, а не астрономического времени машины на которой проводится математическое моделирование, необходимо сбрасывать результаты в файл.
- **файл с начальной матрицей** – начальное состояние системы.
- **префикс имён файлов с результатами** – имя файла задаётся в формате *префикс_модельное_время*.
- **число потоков** – Число OpenMP потоков, которое придётся на один узел кластера.
- **функция для рабты с границей** – выбор функции для вычисления границы из некоторого списка.
- **функция источников тепла** – выбор одного из вариантов для функции $heat(x, y, t)$

Все параметры можно поместить в специальный конфигурационный файл, а так же можно задать параметры по умолчанию. Если параметров передаются не через конфигурационный файл, то весьма желательно, чтобы для их разбора была использована функция `getopt` или `getoptlong`.

3 Список программ и файлов

Итак, необходимо написать:

1. последовательную программу
2. параллельную MPI программу, которая допускает запуск себя в режиме, когда есть только многопоточность и MPI не задействован.
3. набор файлов, в которых находятся реализации функций *heat* и *border*. (Предусмотреть возможность указания функций, которые обнуляют границу и не вносят дополнительного тепла). В идеале реализовать их подгрузку как `.so/.dll` объектов с использованием функции `dlopen`, но на Bluegene не будет работать.
4. Makefile, которым всё это компилируется и, возможно, запускаются тесты на вычислительном кластере.

4 Эксперименты и графики

Собственно построить те же графики, что и в предыдущих заданиях, однако здесь необходимо задействовать MPI-IO, и построить графики с учётом накладных расходов связанных с записью в файл.

Список литературы

- [1] Страничка в википедии про уравнение теплопроводности: https://en.wikipedia.org/wiki/Heat_equation