TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI



**VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

BÀI TẬP LỚN MÔN

TÍNH TOÁN HIỆU NĂNG CAO

**Đề tài: Phương trình nhiệt Diffusion trong hệ tọa độ 2 chiều**

|  |  |
| --- | --- |
| **Họ và tên:** | **Lê Đức Lộc** |
| **SHHV:** | **20211033M** |

Hà Nội - 2023

**Mục Lục**

[1.1 Phương trình nhiệt Diffusion 2D 4](#_Toc148575497)

[1.2 Điều kiện ban đầu của bài toán 5](#_Toc148575498)

[1.3 Phương pháp giải theo lập trình tuần tự 5](#_Toc148575499)

[1.4 Phương pháp lập trình song song MPI 6](#_Toc148575500)

**Danh mục hình ảnh**

[Hình 1: Điều kiện nhiệt độ ban đầu 5](#_Toc148575573)

[Hình 2: Kết quả sau khi chạy chương trình bằng phương pháp lập trình MPI 7](#_Toc148575574)

## Phương trình nhiệt Diffusion 2D

Phương trình nhiệt Diffusion là một phương trình đạo hàm riêng mô tả quá trình truyền nhiệt trong một vật thể hay một vùng không gian. Trong không gian 2 chiều, phương trình nhiệt Diffusion có dạng:



Trong đó, u(x,y,t) là nhiệt độ tại điểm (x,y) vào thời điểm t, và k là hệ số truyền nhiệt của vật liệu. Phương trình này có thể được giải bằng phương pháp phân tách biến số, phương pháp Fourier hay phương pháp hữu hạn.

Giải pháp của phương trình này cho biết sự thay đổi của nhiệt độ theo thời gian và không gian, dựa trên các điều kiện ban đầu và biên cho trước.

Phương trình diffusion 2D khi được xấp xỉ dưới dạng rời rạc có thể được viết như sau:

A math equation with numbers and symbols

Description automatically generated

Đây là một phương pháp xấp xỉ rất phổ biến cho phương trình diffusion 2D, sử dụng phương pháp sai phân hữu hạn. Phương pháp này cho phép bạn tính toán giá trị của U tại mỗi điểm lưới và mỗi bước thời gian dựa trên giá trị của U tại các điểm lưới lân cận và thời điểm trước đó.

Với trường hợp Δx = Δy, phương trình trên có thể viết lại dưới dạng như sau:

A math equation with black text

Description automatically generated

Từ phương trình trên, ta có thể xây dựng hàm tính giá trị nhiệt độ tại điểm (x,y) và thời gian t như sau

void FD(float \*C, float \*dC)

{

int i, j;

float c,u,d,l,r;

for ( i = 0 ; i < m ; i++ )

for ( j = 0 ; j < n ; j++ )

{

c = \*(C+i\*n+j);

u = (i==0) ? \*(C+i\*n+j) : \*(C+(i-1)\*n+j);

d = (i==m-1) ? \*(C+i\*n+j) : \*(C+(i+1)\*n+j);

l = (j==0) ? \*(C+i\*n+j) : \*(C+i\*n+j-1);

r = (j==n-1) ? \*(C+i\*n+j) : \*(C+i\*n+j+1);

\*(dC+i\*n+j) = (D/(dx\*dx))\*(u+d+l+r-4\*c);

}

}

## Điều kiện ban đầu của bài toán

Kích thước của không gian m = 30 hàng, n = 30 cột

Thời gian chạy: 2s

Bước nhảy thời gian: 0,01

Bước thay đổi x: 0,1

Hằng số Delta D: 0,1

Nhiệt độ được phân bố thành một hình chữ nhật trong ma trận C có nhiệt độ cao (80.0) và phần còn lại có nhiệt độ thấp (25.0) như hình dưới

A black and white chart

Description automatically generated

Hình : Điều kiện nhiệt độ ban đầu

## Phương pháp giải theo lập trình tuần tự

Để giải bài toán theo phương pháp tuần tự, chương trình cần thực hiện các bước sau:

* Cấp phát bộ nhớ cho ma trận C và dC.
* Gọi hàm KhoiTao để khởi tạo ma trận nhiệt độ ban đầu
* Thực hiện vòng lặp chính để tính toán sự thay đổi nhiệt độ trong khoảng thời gian T. Trong vòng lặp này, hàm FD được gọi để tính toán sự thay đổi nhiệt độ, và sau đó, nhiệt độ mới được cập nhật tại mỗi điểm trong lưới.

## Phương pháp lập trình song song MPI

* Hàm KhoiTao: Hàm này được sử dụng để khởi tạo nhiệt độ ban đầu trong lưới. Nhiệt độ được thiết lập thành 80.0 trong một khu vực hình chữ nhật giữa lưới và 25.0 trong các phần còn lại.
* Hàm FD: Đây là hàm tính toán sự thay đổi nhiệt độ dựa trên phương trình truyền nhiệt và các điểm lân cận. Hàm này tính toán sự thay đổi nhiệt độ tại mỗi điểm trong lưới, dựa trên nhiệt độ hiện tại và giá trị nhiệt độ của các điểm lân cận.
* Chương trình sử dụng phương pháp sai phân hữu hạn (finite difference) để giải phương trình dẫn nhiệt.
  + Phương pháp này xấp xỉ đạo hàm bậc hai của nhiệt độ theo không gian bằng công thức: laplacian = (l + r + u + d - 4 \* c) / (dx \* dx)

Trong đó l, r, u, d là nhiệt độ của các điểm bên trái, bên phải, bên trên và bên dưới điểm hiện tại, c là nhiệt độ của điểm hiện tại, dx là khoảng cách giữa hai điểm liền kề.

* + Đạo hàm bậc nhất của nhiệt độ theo thời gian được xấp xỉ bằng công thức: dT = D \* laplacian \* dt

Trong đó D là hệ số dẫn nhiệt, dt là khoảng thời gian giữa hai bước lặp. Nhiệt độ mới của mỗi điểm được tính bằng cách cộng dT với nhiệt độ cũ.

* Hàm main: Hàm chính của chương trình, nơi chúng ta thực hiện các bước chính của mô phỏng nhiệt độ. Cụ thể:
  + Gọi MPI\_Init để khởi tạo MPI và lấy thông tin về số lượng tiến trình (processes) và ID của từng tiến trình.
  + Dựa trên số lượng tiến trình, tính toán rows\_per\_process (số hàng của lưới mà mỗi tiến trình quản lý), start\_row (hàng đầu tiên mà tiến trình này quản lý), và end\_row (hàng cuối cùng mà tiến trình này quản lý).
  + Cấp phát bộ nhớ cho hai ma trận T và new\_T. T là ma trận nhiệt độ hiện tại và new\_T là ma trận để lưu trạng thái nhiệt độ mới.
  + Gọi hàm KhoiTao để khởi tạo nhiệt độ ban đầu trong lưới.
  + Bắt đầu vòng lặp thời gian trong Ntime bước.
  + Trong mỗi bước thời gian, tiến trình sử dụng hàm FD để tính toán sự thay đổi nhiệt độ và lưu kết quả vào ma trận new\_T.
  + Sau đó, tiến trình sử dụng MPI\_Allgather để tổng hợp dữ liệu từ tất cả các tiến trình. Dòng này đảm bảo rằng mỗi tiến trình có thông tin về nhiệt độ tại tất cả các hàng của lưới.
  + Cuối cùng, nếu tiến trình có ID là 0 (tiến trình gốc), nó sẽ ghi kết quả vào một tệp "result.txt" và hiển thị nhiệt độ trên màn hình. Nếu không, các tiến trình khác không thực hiện các bước này.
  + Chương trình giải phóng bộ nhớ và kết thúc MPI bằng MPI\_Finalize.
  + Chương trình này mô phỏng sự thay đổi nhiệt độ trong lưới 2D theo thời gian và sử dụng tính toán song song để tăng hiệu suất. Kết quả nhiệt độ được lưu vào tệp "result.txt"
* Kết quả sau khi chạy chương trình

A red and yellow gradient with a white background

Description automatically generated with medium confidence

Hình : Kết quả sau khi chạy chương trình bằng phương pháp lập trình MPI

* Môi trường chạy thực nghiệm: sử dụng máy ảo chạy hệ điều hành ubuntu

A screenshot of a computer

Description automatically generated

Hình : Môi trường chạy thực nghiệm