

高性能计算应用实践

Lab7 实验报告

朱祉睿 220110501 大二秋

1.实现方案

在主进程中初始化 AB 矩阵，通过广播发送给其他四个进程，每个进程解决矩阵乘的四分之一(将矩阵 C 分为四块)，然后再通过点对点通信把结果返回主进程，主进程合并结果。

2.代码截图

```

#include <stdio.h>
#include "mpi.h"
typedef struct
{
    int a;
    int b;
    int c;
    int f;
}para;
int main(int argc, char *argv[])
{
    int numprocs, myid, source;
    int count = 2;
    double C[4] = {0};
    double D[4] = {0};
    para d;
    double A[4];
    double B[4];
    MPI_Status status;

    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &myid);
    MPI_Comm_size(MPI_COMM_WORLD, &numprocs);
    if (myid == 0)
    {
        A[0] = -1.000000e+00;
        A[1]=-9.167380e-01;
        A[2]= -9.980292e-01;
        A[3]= -6.467147e-01;
        B[0] = -2.707955e-01;
        B[1] = -8.154047e-01,
        B[2]= -8.173388e-01;
        B[3] =-2.556555e-02;
    }
}

```

```

MPI_Bcast(A, count * count, MPI_DOUBLE, 0, MPI_COMM_WORLD);
MPI_Bcast(B, count * count, MPI_DOUBLE, 0, MPI_COMM_WORLD);
if(myid == 1 ){
    d.a=1;
    d.b= count / 2;
    d.c=1;
    d.f=count / 2;
}
if(myid==2){
    d.a=count / 2 + 1;
    d.b= count;
    d.c=1;
    d.f= count / 2;
}
if(myid==3){
    d.a=1;
    d.b= count / 2;
    d.c=count / 2 + 1;
    d.f=count;
}
if(myid==4){
    d.a=count / 2 + 1;
    d.b=count;
    d.c=count / 2 + 1;
    d.f=count;
}

```

```

if (myid != 0)
{
    for (int i = d.a - 1; i < d.b; i++)
    {
        for (int j = d.c - 1; j < d.f; j++)
        {
            for (int p = 0; p < count; p++)
            {
                C[i * count+j] += A[i * count + p] * B[p * count + j];
// printf("%d %d %d %e\n",i,j,p,A[i * count + p] * B[p * count + j]);
            }
        }
    }
    MPI_Send(C, count * count, MPI_DOUBLE, 0, 99, MPI_COMM_WORLD);
}

```

```

if(myid ==0 )
{
    for (source = 1; source < numprocs; source++)
    {
        MPI_Recv(D, count * count, MPI_DOUBLE, source,
                99, MPI_COMM_WORLD, &status);
        for (int i = 0; i < count * count; i++)
        {
            if (D[i] != 0)
            {
                C[i] = D[i];
            }
        }
        for (int i = 0; i < count * count; i++)
        {
            printf("%e ", C[i]);
        }
    }
    MPI_Finalize();
    return 0;
}

```

3.碰到的问题

(1) 编译显示变量 A, B 矩阵未声明, 解决: 在 MPI_INIT 前声明, 问题: 不能在各个进程中分别声明变量吗?

(2) 怎么在不同进程中传输自己定义的结构体数据类型?