



天津大学
Tianjin University

《并行计算》实验指导书

2022-2023 学年第二学期

指导老师：于策

实验指导：孙超 李琨 张雅洁 原雨豪

《并行计算》实验指导书

2022-2023 学年第二学期

一、实验要求及评分标准

本课程实验目的为提升学生对并行计算的理解认识，培养学生编写基本并行程序的能力，加深对多线程(Phthread)和多进程(MPI)并行编程的理解认识。

实验课程需要上交实验报告，报告评分标准如下：

实验	内容要求	评分比例	占总分比例
实验一	实验内容	10%	10%
	实验原理	10%	
	程序流程图	30%	
	实验结果及分析	40%	
	实验总结	10%	
实验二	实验内容	10%	30%
	实验原理	10%	
	程序流程图	30%	
	实验结果及分析	40%	
	实验总结	10%	
实验三	实验内容	10%	30%
	实验原理	10%	
	程序流程图	30%	
	实验结果及分析	40%	
	实验总结	10%	
实验四	实验内容	10%	30%
	实验原理	20%	
	设计实现	30%	
	实验结果	30%	
	实验总结	10%	

所有实验的编程语言要求使用 C/C++，实验原理包括：**实验数学计算模型**和**实现方法**；实验结果及分析应包括：**实验结果数据**、**加速比曲线**和**实验结果分析**。

二、实验环境介绍及使用方法

1. 集群登录及所需软件

国家超级计算天津中心提供了集成客户端——青索客户端，使用手册见：<https://www.nscctj.cn/file/青索安装与入门手册.pdf>，以下为常规使用所用软件：

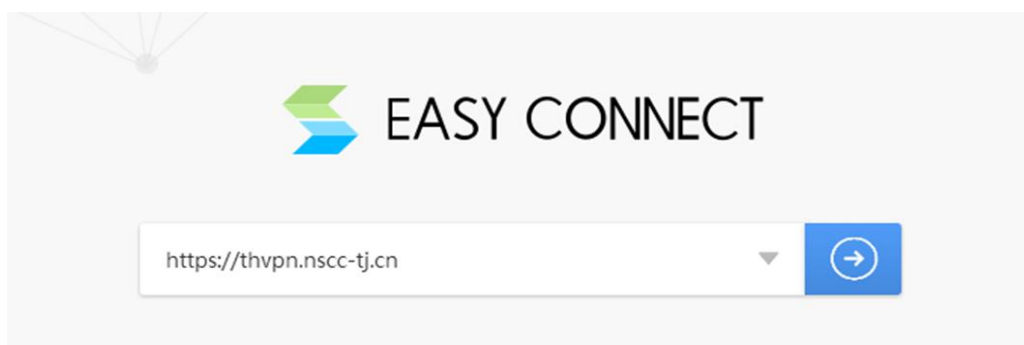
- Easy Connect（VPN 登陆）
 - 校内下载地址：<https://vpn.tju.edu.cn/com/EasyConnectInstaller.exe>
- SSH（命令行远程登陆命令）
 - Win10 可直接使用 ssh 命令
 - Win7 需要安装 OpenSSH，官方网站：<https://www.mls-software.com/opensshd.html>
- WinSCP（远程文件传输工具）
 - 官方网站：<https://winscp.net/eng/index.php>

实验环境不支持图形界面。

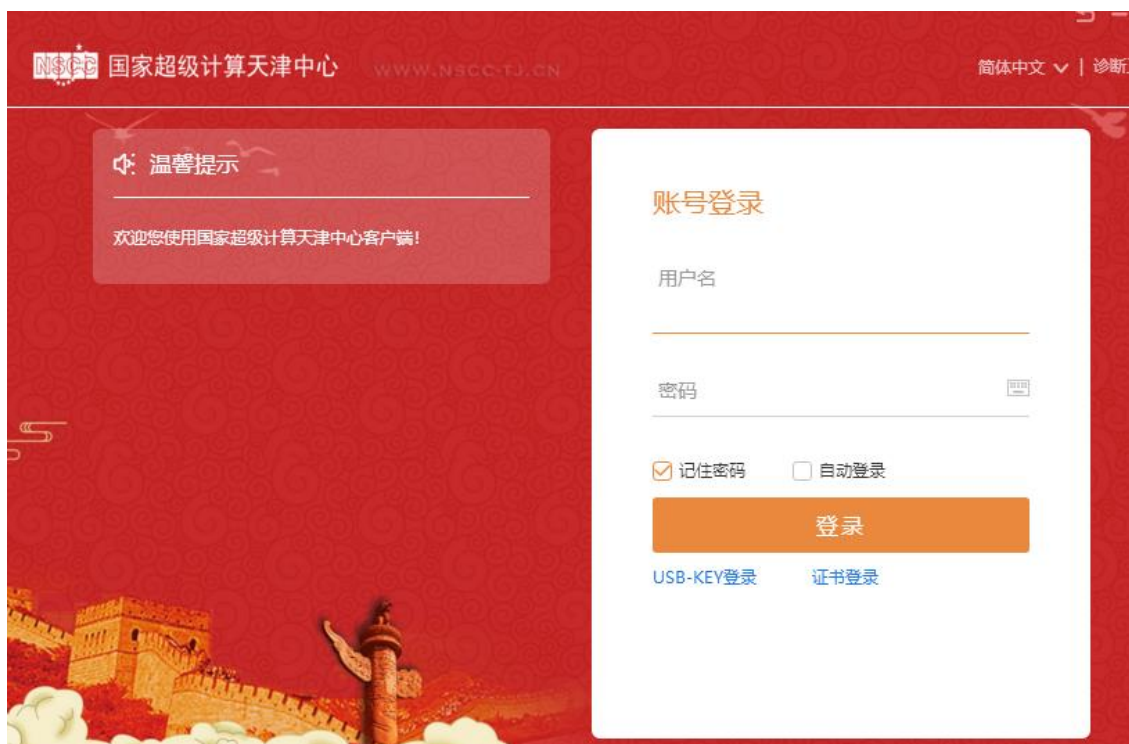
PS：Linux 可以安装 Easy Connect（<https://vpn.tju.edu.cn> 可下载）后直接使用 OpenSSH 进行登录，Mac OS 也可以使用对应的 ssh 命令登录链接到集群。

2. 登陆 VPN

- 登录 VPN 方式如图所示



在输入框中输入：<https://thvpn.nscctj.cn>，点击右侧箭头连接国家超级计算天津中心 VPN。



在输入框中输入 VPN 登陆的用户名和密码（另行通知），点击登陆按钮等待连接成功。

3. 登录集群

- 登录集群方式如图所示：



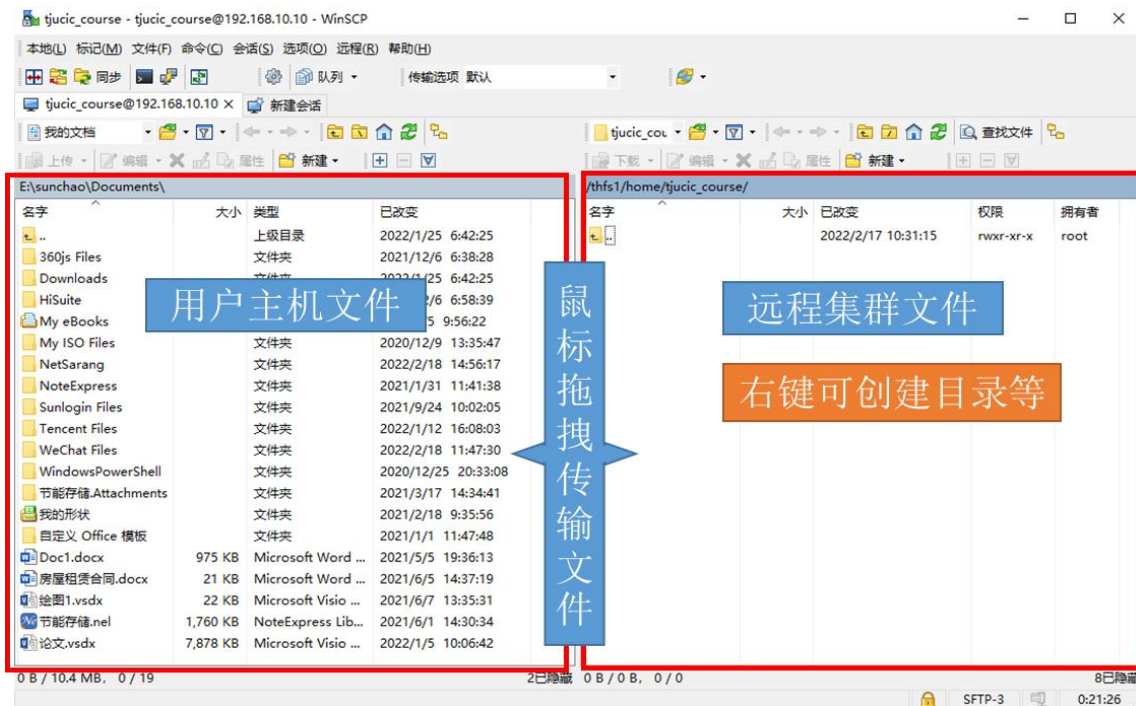
在提示符后输入：ssh USERNAME@192.168.10.10，其中，需要将 USERNAME 替换为自己的账户名称（另行通知），然后按回车键登录系统。首次登陆会提示密钥指纹信息，输入 yes 继续连接，之后根据提示输入密码（无回显），按回车键确认，待提示欢迎信息后就可以正式使用了。

- 文件传输：

打开 WinSCP 软件，弹出登陆界面，如图所示：



输入主机名、用户名、密码，点击登陆按钮可以打开文件传输界面，通过鼠标拖动操作即可实现文件传输。



3. 常用 Linux 命令

- ls: 列出当前文件夹下文件。如: ls -al
- mkdir: 新建文件夹。如: mkdir data
- cd: 切换工作文件夹。如: cd data/

- pwd: 查看当前文件夹绝对路径
- rm: 删除文件或文件夹（需要加上 -r 参数）
- time: 获取程序运行时间
- passwd: 修改登录密码
- exit: 退出登录

4. 实验环境

i. 系统参数

- 国家超级计算天津中心定制操作系统
- 使用国产飞腾处理器
- 天河自主高速互联网络 (400GB/s)
- 单核理论性能 (双精度) 9.2 GFlops
- 单节点理论性能 (双精度) 588.8 GFlops

ii. 编译环境

- GCC 9.3.0: gcc, g++, gfort 等
- OpenMPI 4.1.1: mpicc, mpic++等

iii. 示例

- g++ -pthread -o test.o test.cpp
- g++ -fopenmp -o test.o test.cpp
- mpic++ -o test.o test.cpp

注: 运行 MPI 命令前需加载 OpenMPI 环境:

- module load openmpi

5. 使用任务队列

i. 同步执行, 可用于小规模测试。程序执行结束前中断连接会导致程序中止。

- 测试串行程序示例(test.o):
yhrun -p thcp1 -n 1 ./test.o
- 测试多线程程序示例(test.o), 使用 8 个核:
yhrun -p thcp1 -n 1 -c 8 ./test.o
- 测试多进程程序示例(test.o), 使用 2 个节点, 共 8 个核, 每节点 4 个核:
yhrun -p thcp1 -N 2 -n 8 ./test.o

ii. 异步执行, 常规使用方法。通过提交任务实现, 提交任务后可随时退出。

- 步骤 1: 编写任务脚本。脚本编写及参数设置可参考 SLURM 调度系统:

<https://slurm.schedmd.com/sbatch.html>

- 串行程序脚本示例:

```
test.sh

#!/bin/bash

time yhrun -p thcp1 -n 1 ./test.o &> run.log
```

- 多线程程序脚本示例:

```
test.sh

#!/bin/bash

time yhrun -p thcp1 -n 1 -c 8 test.o &> run.log
```

- 多进程程序脚本示例:

```
test.sh

#!/bin/bash

module load openmpi

time yhrun -p thcp1 -N 2 -n 8 test.o &> run.log
```

- 步骤 2: 提交任务

- 提交串行程序任务:

```
yhbatch -p thcp1 -n 1 ./test.sh
```

- 提交多线程程序任务:

```
yhbatch -p thcp1 -n 1 ./test.sh
```

- 提交多进程程序任务, 使用 2 个节点, 共 8 个核, 每节点 4 个核:

```
yhbatch -p thcp1 -N 2 -n 8 ./test.sh
```

- 步骤 3: 查看/删除任务

- 查看任务列表, 找到相应 `jobid`:

```
yhq
```

- 删除任务:

```
yhcancel jobid
```

6. 任务结果查看

- 任务运行结果文件名称默认保存格式为: `slurm-jobid.out`。

如: `slurm-418101.out`

- 可以在任务脚本中添加输出重定向, 输出到自定义文件中。

如: `time yhrun -p thcp1 -n 1 ./test.o &> run.log`

三、实验题目

实验一：多线程计算 PI 值

本实验不对输入数据做特殊规定。**要求：Pthread 并行化实现。**

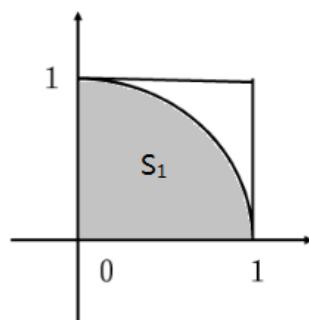
i. 积分法

$$\pi = \int_0^1 \frac{4}{1+x^2} dx \approx \sum_{0 \leq i \leq N} \frac{4}{1 + (\frac{i+0.5}{N})^2} \times \frac{1}{N}$$

ii. 概率方法

如右图，在正方形中随机的投 n 个点，若有 m 个落入圆弧内，则：

$$\frac{m}{n} \approx \frac{S_1}{1} = \frac{\pi}{4}$$



iii. 幂级数计算方法

$$\pi = 4 \times \arctan(1) = 4 \times (1 - \frac{1}{3} + \frac{1}{5} - \dots + \frac{(-1)^{n+1}}{2n-1} - \dots)$$

➤ 编译与运行

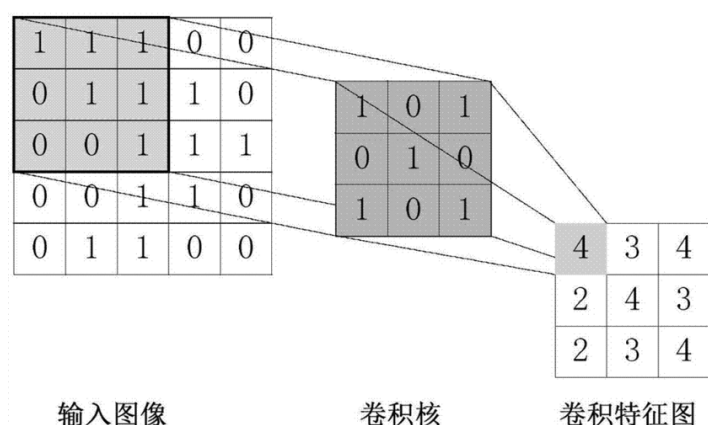
- 编译命令：`g++ -pthread -o test.o test.cpp`
- 同步执行命令：`time yhrun -p thcp1 -n 1 -c 8 ./test.o`
- 脚本示例：

```
test.sh
#!/bin/bash
time yhrun -p thcp1 -n 1 -c 8 test.o &> run.log
```

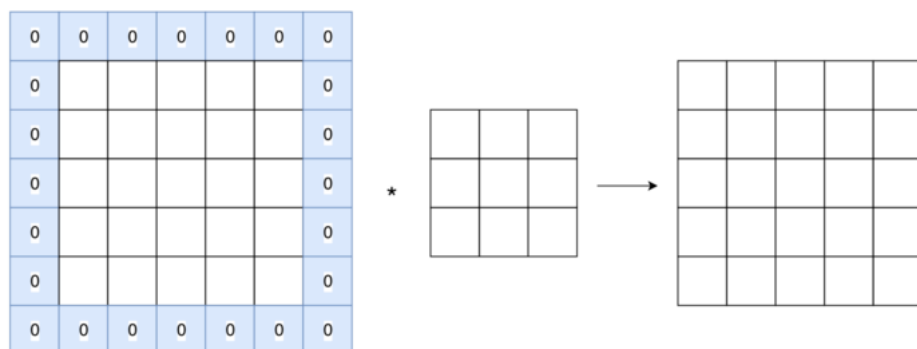
- 提交任务：`yhbatch -p thcp1 -n 1 ./test.sh`

实验二：多线程计算卷积

卷积是一种积分变换的数学方法，广泛应用于通信、物理、图像处理等领域。图像处理中，卷积操作就是卷积核（过滤器 / Filter）在原始图像中进行滑动得到特征图的过程，如图所示。



为了使卷积后的特征图的大小与原始图像相同，需要在原始图像的周围添加 0 像素，这个操作叫做零填充，此操作还能加强对图像周边像素信息的利用。



本实验的输入需包含代表图像的二维矩阵和一个卷积核，需要进行 N 次卷积处理，输出最终结果。**要求：Pthread 并行化实现。**

i. 串行算法

Begin

```

for w in 1..N
  for w in 1..W (img_width)
    for h in 1..H (img_height)
      for x in 1..K (filter_width)
        for y in 1..K (filter_height)
          output(w, h) += input(w+x, h+y) * filter(x, y)
        endfor
      endfor
    endfor
  endfor
endfor

```

End

ii. 并行算法

划分方法可参考课程中的 Jacobi 迭代，将原始图像划分成 p （线程数）个子块，每个线程处理一个子块，进行 N 次卷积计算，计算中需要注意多个线程计算过程的同步。

➤ 编译与运行

- 编译命令: `g++ -pthread -o test.o test.cpp`
- 同步执行命令: `time yhrun -p thcp1 -n 1 -c 8 ./test.o`
- 脚本示例:

```
test.sh

#!/bin/bash
time yhrun -p thcp1 -n 1 -c 8 test.o &> run.log
```

- 提交任务: `yhbatch -p thcp1 -n 1 ./test.sh`

实验三：多进程计算卷积

本实验针对实验二问题，采用 MPI 编程模型实现卷积计算。

划分方法可参考课程中的 Jacobi 迭代，将原始图像划分成 p （进程数）个子块，每个进程处理一个子块，进行 N 次卷积计算，计算中每一个进程都要向相邻的进程发送数据，同时从相邻的进程接收数据。

➤ 编译与运行

- 编译命令: `mpic++ -o test.o test.cpp`
- 加载 MPI 环境: `module load openmpi`
- 同步执行命令: `time yhrun -p thcp1 -N 2 -n 8 ./test.o`
- 脚本示例:

```
test.sh

#!/bin/bash
module load openmpi
time yhrun -p thcp1 -N 2 -n 8 test.o &> run.log
```

- 提交任务: `yhbatch -p thcp1 -N 2 -n 8 ./test.sh`

实验四：多级并行化计算卷积

本实验针对实验二问题，采用 MPI+OpenMP 编程模型实现卷积计算。节点间采用 MPI，节点内采用 OpenMP。需要制定多层划分策略。

➤ 编译与运行

- 编译命令: `mpic++ -fopenmp -o test.o test.cpp`
- 加载 MPI 环境: `module load openmpi`

- 同步执行命令: `time yhrun -p thcp1 -N 2 -n 2 -c 4 ./test.o`
- 脚本示例:

test.sh
<pre>#!/bin/bash module load openmpi time yhrun -p thcp1 -N 2 -n 2 -c 4 test.o &> run.log</pre>

- 提交任务: `yhbatch -p thcp1 -N 2 -n 2 ./test.sh`