



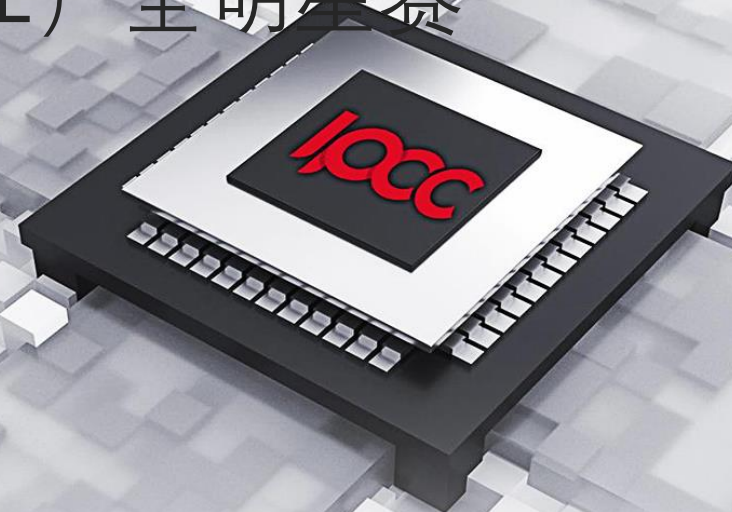
Association for
Computing Machinery

ipccc ACM中国
国际并行计算挑战赛

AMD

全国大学生高性能计算超级联赛（HPC-PL）全明星赛 暨第三届ACM中国-国际并行计算挑战赛开幕赛

汇报人：赵展 2022年5月22日





目录 CONTENTS

01. 个人简介

02. 应用程序运行的硬件环境和软件环境

03. 优化方法

04. 程序运行结果



Association for
Computing Machinery



ACM中国
国际并行计算挑战赛



个人简介



大家好，我是山东大学软件工程专业的赵展。非常荣幸能够参加这个活动，和大家进行交流。我在大三的并行计算课程初次接触到高性能领域，并在课程刘卫国教授（后面有幸跟着老师读研）的带领下参加了20，21两年的PAC比赛。

软硬件环境

作业运行的硬件环境为BSCC-A5 超算系统，队列为 amd_256 队列，amd_256 队列的计算节点配置是： AMD 7452@2.35GHz，256G 内存，32核，共可用两个节点，节点间采用 ib 网互连。



```
vendor_id      : AuthenticAMD
cpu family     : 23
model          : 49
model name     : AMD EPYC 7452 32-Core Processor
stepping       : 0
microcode      : 0x830104d
cpu MHz        : 2345.596
cache size     : 512 KB
physical id    : 1
siblings       : 32
core id        : 31
cpu cores      : 32
apicid         : 95
initial apicid : 95
fpu            : yes
fpu_exception : yes
cpuid level    : 16
wp             : yes
```

运行环境Linux版本为3.10.0-

1160.11.1.el7.x86_64，操作系统为CentOS

7.9.2009。

优化方法

替换编译器及编译参数

使用openmp并行

位运算优化If判断

优化访存，减少拷贝



优化方法

替换编译器及编译参数

GCC 4.8.5  GCC 10.2.0

-march=native

-Ofast

使用OpenMP并行

由于迭代间存在依赖，在迭代内部进行并行

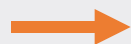
OpenMP

```
for (int iter = 0; iter < max_iter; ++iter) {  
    printf("Iter %d...\n", iter);  
    for (int i = 1; i < N - 1; ++i) {
```

优化方法

位运算优化if判断

```
if (a[i][j] == 1) {  
    if (cnt == 2 || cnt == 3) {  
        tmp[i][j] = 1;  
    } else {  
        tmp[i][j] = 0;  
    }  
} else { // a[i][j] == 0  
    if (cnt == 3) {  
        tmp[i][j] = 1;  
    } else {  
        tmp[i][j] = 0;  
    }  
}
```



```
tmp[i][j] = (!(cnt^3))||((a[i][j]&1)&&(!(cnt^4)));
```

优化方法

减少拷贝，优化访存

```
for (int i = 0; i < N; ++i) {  
    for (int j = 0; j < N; ++j) {  
        a[i][j] = tmp[i][j];  
    }  
}
```

重复利用a数组和tmp数组，取消这次拷贝

运行结果

版本	运行时间(MS)	加速比
初始版本	109007.755	---
更换编译器及 编译参数	8061.762	13.52X
OpenMP并行	342.265	318.48X
位运算	63.761	1709.63X
减少拷贝	39.754	2742.06X



感谢观看

THANKS FOR WATCHING

