Performance comparison of pthread, CUDA and openMP

CS4032D Assignment
Report

Submitted by
Dev Kumar (B170514CS)
Pritam Patel (B170418CS)
Balla Raviteja (B170115CS)

Under the Guidance of
DR. Saidalvi Kalady
Assistant Professor

Department of Computer Science and Engineering National Institute of Technology Calicut Calicut, Kerala, India - 673 601

April, 2021

Problem statement :- Calculating PI Value In Different Parallel Framework.

Preliminaries

$$\int_0^1 \frac{1}{1+x^2}dx = arctanx\Big|_0^1 = \frac{\pi}{4} \quad \Rightarrow \quad \pi = 4 \times \int_0^1 \frac{1}{1+x^2}dx$$

We use Riemann Sum with Midpoint Rule to calculate the integration, which includes loops that may be optimized using parallel computing.

Frameworks

PThread

Used pthread as the parallel framework to calculate Pi.

OpenMP

Use OpenMP as the parallel framework to calculate Pi.

CUDA

Used CUDA to optimize the parallel computing process, which must be run under CUDA environment. i.e., at least a nVidia card and nvcc must be installed to compile and run the code.

Experiment

All experiments are carried out under Linux with nvcc and nVidia cards installed.

And we choose 2^30 as the STEP_NUM for all frameworks.

Parallel parameters are listed below:

Framework	Parameters
OpenMP	16 Threads
PThread	16 Threads
CUDA	512 Threads / 64 Blocks

Results

Cuda Program:-

```
Configuring device...
1 CUDA Capable device(s) detected
Start calculating in optimized kernel function...
PI = 3.1416046619415283 with error 0.0000119209289551
Time elapsed : 0.009735 seconds.
```

OpenMP Program:-

```
Start calculating...
PI = 3.14159265359088334 with error 0.0000000000010902
Time elapsed : 2.988628 seconds.
```

PThread Program:-

```
Start calculating with 16 threads...
PI = 3.9647432709085915 with error 0.8231506173187984
Time elapsed : 1.967153 seconds.
```

Comparative Analysis:-

Framework	Parameters	Error	Time taken
CUDA	512 Threads / 64 Blocks	0.0000119209289551	0.009735 seconds
OpenMP	16 Threads	0.0000000000010902	2.988628 seconds
PThread	16 Threads	0.8231506173187984	1.967153 seconds

Conclusion

Among all the frameworks CUDA computes the value of π fastest because it uses 32768 threads in parallel and OpenMP is slowest because it computes the value for more precision digits as compare to any of the frameworks and PThread is gives highest error of all the frameworks because it computes the value for less precision digits as compare to any of the framework.

Contribution

Balla Raviteja has prepared and presented the slides of Introduction to CUDA, CUDA features- standard math functions and written the code of pthread which is a pthread.c file.Pritam Patel has prepared and presented the slides of atomic operations in CUDA, memory –CUDA malloc, CUDA free, CUDA memcopy and written the code of openmp which is a openmp.c file.Dev Kumar has prepared and presented the slides of CUDA graphics, CUDA function declaration-CUDA vector addition and written the code of cuda which is a cuda.ipynb file. Report and readme file is made in a group by all of us.