**Parallel Programming Exercise 4– 12**

|  |  |
| --- | --- |
| **Author:** | 李明軒 ([b08505012@ntu.edu.tw](mailto:b08505012@ntu.edu.tw)) |
| **Student ID** | B08505012 |
| **Department** | Engineering Science and Ocean Engineering |

(If you and your team member contribute equally, you can use (co-first author), after each name.)

# Problem and Proposed Approach

(Brief your problem, and give your idea or concept of how you design your program.)

Problem:

Write a parallel program that apply Simpson`s rule.

Concept:

Distribute the data and use MPI\_Reduce to calculate the result.

# Theoretical Analysis Model

(Try to give the time complexity of the algorithm, and analyze your program with iso-efficiency metrics)

time complexity: Θ(logp+n/p)

Dvide 1 into n interval. P=number of processor.

Iso-efficiency: n>Cplogp

T(n,1)= Θ(n), To(n,p)=pk=Θ(plogp)

M(n)=n, M(Cplogp)/p=Clogp

# Performance Benchmark

(Give your idea or concept of how you design your program.) (寫在第一題)

Using the estimated time provided in the textbook.

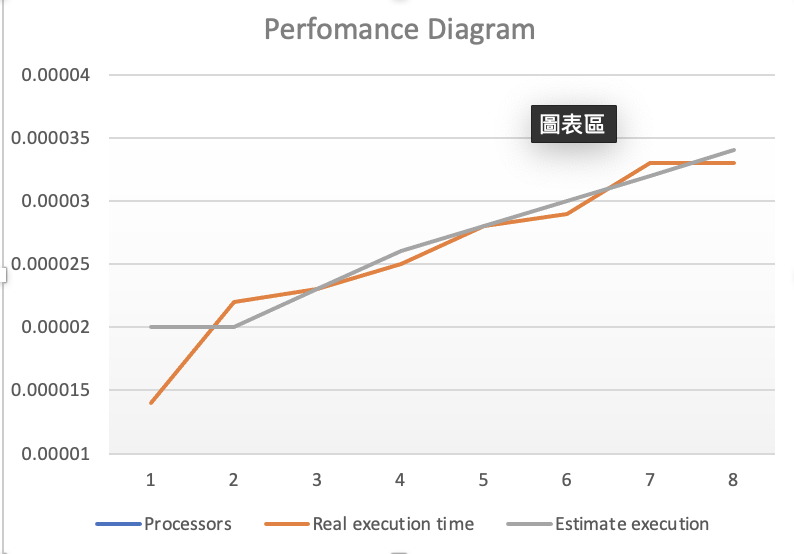
Estimate execution time : λ(logp)+χ(n/p)

λ: 0.000015, n = 50

χ: 0.0000004

Table . The execution time

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Processors | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Real execution time | 0.000014 | 0.000022 | 0.000023 | 0.000025 | 0.000459 | 0.000029 | 0.000033 | 0.000033 |
| Estimate execution time | 0.000020 | 0.000020 | 0.000023 | 0.000026 | 0.000028 | 0.000030 | 0.000032 | 0.000034 |
| Speedup | 1 | 1 | 0.86956 | 0.76923 | 0.71428 | 0.66667 | 0.62500 | 0.588235 |
| Karp-flatt metrics | X | X | 1.22501 | 1.400002 | 1.500014 | 1.6 | 1.7 | 1.800001 |



# Conclusion and Discussion

1. What is the speedup respect to the number of processors used?

: As the processor increase the speedup decrease. The reason might be the data is too small and λ(logp) dominate the time.

1. How can you improve your program further more

: Using Parallel program isn`t a good idea here, I think we can just use one processor.

1. How does the communication and cache affect the performance of your program?

: Most of time are spending on communicating.

1. How does the Karp-Flatt metrics and Iso-efficiency metrics reveal?

: e increase as the processor increase, and from the formula we know the program has pretty good scalability.

**Appendix(optional):**

