**Parallel Programming Exercise 4 – 8**


| | |
|---|---|
| **Author:** | 林子傑 (r10525069@ntu.edu.tw) |
| **Student ID** | R10525069 |
| **Department** | Engineering Science and Ocean Engineering |

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


## 1 Problem and Proposed Approach

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

Problem: 找出 1 到 n 所有數對 (I,I+2) 滿足 I 和 I+2 都是質數。

Proposed Approach:每個 processor 負責 in/p 到 (i+1)n/p-1 的範圍，如果邊界是偶數，就往外 +1 或 -1，利用迴圈判斷每個奇數是不是質數，如果連續兩個奇數都是質數，代表成功找到一組數對，最後用 reduction 將所有 processor 的找到的對數進行加總。

## 2 Theoretical Analysis Model

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

Sequential algorithm complexity：$\Theta(n\sqrt{n})$

Parallel computational complexity：$\Theta(n\sqrt{n}/p)$

Parallel communication complexity：$\Theta(\log p)$

Parallel overhead：$To(n, p) = \Theta(p \log p)$

Iso-efficiency relation：$p >= Cp \log p$

$M(n)=1$

$M(Cp \log p)/p = C \log p /p$

## 3 Performance Benchmark

(Give your idea or concept of how you design your program.)

The time to determine whether a number x is prime：$\chi$

Sequential execution time：$n\chi$

Parallel：

The computation time for each process: $\chi \lceil n/p \rceil$

Necessary communication time：$\lambda$

Parallel execution time： $\chi \lceil n/p \rceil + \lambda$

Table 1. The execution time

| Processors | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Real execution time | 0.171128 | 0.105235 | 0.073259 | 0.056167 | 0.045521 | 0.038271 | 0.032821 | 0.028995 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Estimate execution time | 0.15 | 0.075 | 0.05 | 0.0375 | 0.03 | 0.025 | 0.021429 | 0.01875 |
| Speedup | | 1.626151 | 2.335931 | 3.046771 | 3.75932 | 4.47148 | 5.213979 | 5.901983 |
| Karp-flatt metrics | | 0.229898 | 0.142142 | 0.104288 | 0.082507 | 0.068368 | 0.057091 | 0.050782 |

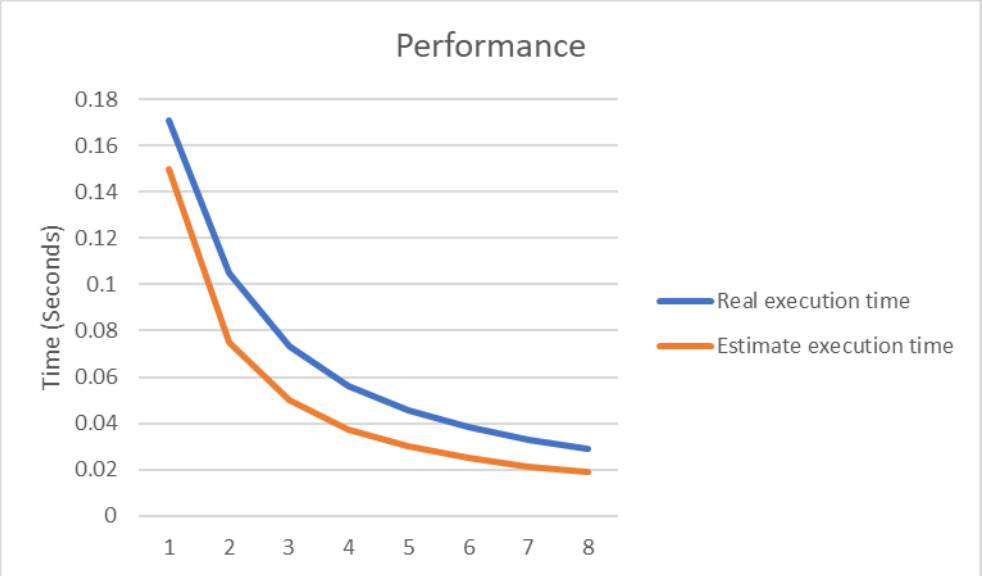


Figure 1. The performance of diagram

## 4 Conclusion and Discussion

(Discuss the following issues of your program

1. What is the speedup respect to the number of processors used?
2. How can you improve your program further more
3. How does the communication and cache affect the performance of your program?
4. How does the Karp-Flatt metrics and Iso-efficiency metrics reveal?

)

從 speedup 的數據來看，當 processor 增加，speedup 的數據也會增加，本問題適合用平行計算。

從 Iso-efficiency metrics 顯示出這個程式有很好的 Scalability。

## Appendix(optional):

(If something else you want to append in this file, like picture of life game)