臺灣大學

chapter 7 exercises

7.1 Using the definition of speedup presented in section 7.2, prove that there exists a P_0 such that $P > P_0 \Rightarrow \psi(n,P) < \psi(n,P_0)$.

Assume $K(n,P) = C \log P$

$$\text{Speedup} = \frac{\text{Sequential execution time}}{\text{Parallel execution time}}$$

$$\psi(n,P) \le \frac{\sigma(n) + \varphi(n)}{\sigma(n) + \varphi(n)/P + K(n,P)}$$

$$\varphi(n)/P + C\log P > \varphi(n)/P_0 + C\log P_0 \quad \text{, when } P > P_0$$

$$\varphi(n)/P - \varphi(n)/P_0 > C\log P_0 - C\log P$$

$\because P > P_0 \quad , \therefore \log P > \log P_0 \; , \; C(\log P_0 - \log P) < 0$

$$\frac{1}{P_0} > \frac{1}{P} \quad , \; \varphi(n)\left(\frac{1}{P} - \frac{1}{P_0}\right) < 0$$

then

$$C(\log P_0 - \log P) < \varphi(n)\left(\frac{1}{P} - \frac{1}{P_0}\right) < 0$$

代表因為溝通而增加的時間消耗比可平行化部分所減少的計算時間還多。

$$C\left(\log \frac{P_0}{P}\right) < \varphi(n)\left(\frac{P_0 - P}{PP_0}\right) < 0$$

No.
Date:

7.2 Starting with the definition of efficiency presented in Section 7.2, Prove that $p' > p \Rightarrow \varepsilon(n,p') \leq \varepsilon(n,p)$

$$\text{Efficiency} = \frac{\overset{= \text{SET}}{\text{Sequential execution time}}}{\text{Processors used} \times \underset{\text{PET}}{\text{Parallel execution time}}} = \frac{\text{Speedup}}{p}$$

$$\varepsilon(p) = \frac{SET}{p \times PET(p)}, \quad \varepsilon(p') = \frac{SET}{p' \times PET(p')}$$

$$\varepsilon(p') \leq \varepsilon(p), \quad \frac{SET}{p' \times PET(p')} \leq \frac{SET}{p \times PET(p)}$$

$$\Rightarrow \frac{1}{p' \times PET(p')} \leq \frac{1}{p \times PET(p)}$$

$$\Rightarrow \frac{p}{p'} \leq \frac{PET(p')}{PET(p)}$$

then

ε must ≤ 1 and ≥ 0, so $0 \leq \frac{SET}{p' \times PET(p')} \leq \frac{SET}{p \times PET(p)} \leq 1$

$$\Rightarrow 0 \leq \frac{p}{p'} \leq \frac{PET(p')}{PET(p)} \leq 1$$

$$\Rightarrow \frac{p}{p'} \leq 1 \Rightarrow p \leq p'$$

If $p' \neq p$

then $p' > p$ when $\varepsilon(n,p') \leq \varepsilon(n,p)$ #

Date:

7.3 Estimate the Speedup achievable by the parallel reduction algorithm developed in Section 3.5 on 1, 2, ... 16 processors. Assume $n = 100000$, $\chi = 10$ nanoseconds and $\lambda = 100\ \mu sec$.

$$\text{Speedup} = \frac{\text{sequential execution time}}{\text{parallel execution time}}$$

$$\text{sequential execution time} = (n-1)\chi$$

$$\text{parallel execution time} = (\lceil n/p \rceil - 1)\chi + \lceil \log p \rceil (\lambda + \chi)$$

$$\text{Speedup} = \frac{(n-1)\chi}{(\lceil n/p \rceil - 1)\chi + \lceil \log p \rceil (\lambda + \chi)}, \quad n = 10^5,\ \chi = 10^{-8}\ sec,\ \lambda = 10^{-4}\ sec$$

speedup = 1.000 $p=1$

= 1.6666 $p=2$

= 2.3077 $p=3$

= 2.2222 $p=4$

= 2.4999 $p=5$

= 2.7272 $p=6$

= 2.9166 $p=7$

= 3.0768 $p=8$

= 2.4323 $p=9$

= 2.4999 $p=10$

= 2.5580 $p=11$

= 2.6086 $p=12$

= 2.6529 $p=13$

= 2.6922 $p=14$

= 2.7271 $p=15$

= 2.1620 $p=16$

hint if $n = 10^7$

數字很漂亮

Date:

7.4 Benchmarking of a sequential program reveals that 95 percent of the execution time is spent inside functions that are amenable to parallelization. What is the maximum speedup we could expect from executing a parallel version of this program on 10 processors?

Amdahl's Law

$$\psi \le \frac{1}{f+(1-f)/p}, \quad f = 1 - 0.95 = 0.05$$

$$\psi \le \frac{1}{0.05+(0.95)/p}, \quad p = 10$$

$$\psi \le \frac{1}{0.05 + 0.095}$$

maximum $\psi = \frac{1}{0.145} \cong 6.89655$

ψ

7.5 For a problem size of interest, 6 percent of the operations of a parallel program are inside I/O functions that are executed on a single processor. What is the minimum number of processors needed in order for the parallel program to exhibit a speedup of 10?

Amdahl's Law

$\psi \le \dfrac{1}{f+(1-f)/p}$, $f = 0.06$

$\psi = 10$, $p = ?$

$10 \le \dfrac{1}{0.06+0.94/p}$

$\Rightarrow \dfrac{1}{10} \ge 0.06+0.94/p \Rightarrow 0.1 \ge 0.06+0.94/p \Rightarrow 0.04 \ge 0.94/p$

$\Rightarrow \dfrac{1}{p} \le \dfrac{0.04}{0.94} \Rightarrow p \ge \dfrac{0.94}{0.04}$, $p \ge 23.5$

$\therefore$ minimum number of processors needed is $\underline{24}$ #

No.
Date:

7.6 What is the maximum fraction of execution time that can be spent performing inherently sequential operations if a parallel application is to achieve a speedup of 50 over its sequential counterpart?

Amdahl's Law

$$50 \le \frac{1}{f + \frac{1-f}{p}}$$

if $p \to \infty$

$$50 \le \frac{1}{f} \Rightarrow f \le \frac{1}{50} \Rightarrow \underline{f \le 0.02}$$ #

ate:

7.7 Shauna's parallel program achieves a speedup of 9 on 10 processors. What is the maximum fraction of the computation that may consist of inherently sequential operations?

Amdahl's Law

$$\psi \le \frac{1}{f + \frac{(1-f)}{10}} \quad , \quad \psi = 9$$

$$9 \le \frac{1}{f + \frac{(1-f)}{10}} \Rightarrow f + \frac{1}{10} - \frac{f}{10} \le \frac{1}{9} \Rightarrow \frac{9}{10} f \le \frac{1}{90}$$

$$\Rightarrow f \le \frac{1}{81}$$, maximum fraction $\cong 0.012345679$ //

7.8 Brandon's parallel program execute in 242 second on 16 processors. Through benchmarking he determines that 9 second is spend performing initializations and cleanup on one processor. During the remaining 233 seconds all 16 processors are active. What is the scaled speedup achieved by Brandon's program?

$$S = \frac{\sigma(n)}{\sigma(n) + \frac{\varphi(n)}{p}} \quad , \quad \sigma(n) = 9 \text{ second}, \quad \frac{\varphi(n)}{p} = 233 \text{ second}$$

$$S = \frac{9}{242}$$

$$\psi = P + (1-P)S \quad , \quad P = 16$$

$$\psi = 16 + (-15)\frac{9}{242}$$

$$\cong 15.442$$

7.9 Courtney benchmarks one of her parallel programs executing on 40 processors. She discovers that it spends 99 percent of its time inside parallel code. What is the scaled speedup of the program?

$s = 1 - 0.99 = 0.01$

$p = 40$, $\psi = p - (p-1)s$

$\psi = 40 - (40-1)0.01$

$= 40 - 0.39 = 39.61$

No.
Date: ☆

7.10 The execution times of six parallel programs, labeled I - VI, have been benchmarked on 1, 2, ... 8 processors. The following table presents the speedups achieved by these programs.

For each of these programs, choose the statement that best describes its likely performance on 16 processors:

A: The speedup achieved on 16 processors will probably be at least 40 persent higher than the speedup achieved on eight processors.

B: The speedup achieved on 16 processors will probably be less than 40 persent higher than the speedup achieved on eight processors, due to the large serial component of the computation.

C: The speedup achieved on 16 processors will probably be less than 40 persent higher than the speedup achieved on eight processors, due to the increase in overhead as processors are added.

	Speedup					
Processors	I	II	III	IV	V	VI
1	1.00	1.00	1.00	1.00	1.00	1.00
2	1.67	1.89	1.89	1.96	1.74	1.94
3	2.14	2.63	2.68	2.88	2.30	2.82
4	2.50	3.23	3.39	3.67	2.74	3.65
5	2.78	3.68	4.03	4.46	3.09	4.42
6	3.00	4.00	4.62	5.22	3.38	5.15
7	3.18	4.22	5.15	5.93	3.62	5.84
8	3.33	4.35	5.63	6.25	3.81	6.50
16	4.66	6.09	7.88	8.75	5.33	9.10

Date:

serial fraction $e = \frac{1/\psi - 1/p}{1 - 1/p}$

processors	I	II	III	IV	V	VI
1	–	–	–	–	–	–
2	0.1976	0.0582	0.0582	0.0204	0.1494	0.0309
3	0.2009	0.0703	0.0597	0.0208	0.1522	0.0319
4	0.2000	0.0795	0.0600	0.0300	0.1533	0.0320
5	0.1996	0.0897	0.0602	0.0303	0.1545	0.0328
6	0.2000	0.1000	0.0597	0.0299	0.1550	0.0330
7	0.2002	0.1098	0.0599	0.0301	0.1556	0.0331
8	0.2003	0.1199	0.0601	0.0400	0.1571	0.0330
⋮						
16	0.1621	0.1085	0.0687	0.0552	0.1333	0.0505
	↓	↓	↑	↑	↓	↑

e如果下降代表16顆處理器無法達到8顆處理器speedup的1.4倍。

B	C	A	A	C	A
固定在0.2附近	上升很快			緩慢上升	

Date:

7.11 Both Amdahl's Law and Gustafson-Barsis's Law are derived from the same general speedup formula. However, when increasing the number of processors p, the maximum speedup predicted by Amdahl's Law converges on $1/f$, while the speedup predicted by Gustafson-Barsis's Law increase without bound. Explain why this is so.

Amdahl's Law

$$f = \frac{\sigma(n)}{\sigma(n)+\varphi(n)} \quad , \quad \psi \le \frac{1}{f+\frac{1-f}{p}}$$

如果 f 保持不變，且 p 不斷上升至 ∞ 則 $\frac{1-f}{p} \to 0$

$\therefore \psi \le \frac{1}{f}$

Gustafson's Law

$$s = \frac{\sigma(n)}{\sigma(n)+\frac{\varphi(n)}{p}} \quad , \quad \psi \le s + (1-s)p$$

如果 s 保持不變，且 p 不斷上升，因為 s 內有 p 的存在，所以 p 上升的同時，n 也會上升。

而

$\psi \le s + (1-s)p$ 因為 s 為固定，所以會隨著 p、n 上升而不斷上升。

7.12 Given a problem to be solved and access to all the processors you care to use, can you always solve the problem within a specified time limit? Explain your answer.

如果要解決一個相同的問題，隨著限制的時間越來越少，則為了要達成條件處理器(P)也必須越來越多。

但一個問題必定包含可分工的部分(φ)與不可分工的部分(σ)和溝通的時間(κ)，

而 $\sigma+\varphi+\kappa$ 就是解決問題所需的大概時間。

隨著處理器的上升可分工的部分所耗時間則變成 $\frac{\varphi}{p}$，

並且 κ 也可能隨著 P 上升而增加，假設考慮 κ 保持不變，

則 $(\sigma+\varphi+\kappa)-(\sigma+\frac{\varphi}{p}+\kappa) \Rightarrow \varphi-\frac{\varphi}{p}$

而 $\frac{\varphi}{p}$ 表明了隨著 P 上升減少的耗時並不會隨著 P 而線性增加，

反而會越來越小，並且當 P 大到一定程度時可能幾乎成為定值。

如果再將隨著 P 上升而增加的溝通時間 κ 算進來可能還會耗費更多的時間在解相同的問題。

Date

7.13 Let $n \geq f(p)$ denote the isoefficiency relation of a parallel system and $M(n)$ denote the amount of memory required to store a problem of size n. Use the scalability function to rank the parallel systems shown below from most scalable to least scalable.

a. $f(p) = Cp$ and $M(n) = n^2$

b. $f(p) = C\sqrt{p}\log p$ and $M(n) = n^2$

c. $f(p) = C\sqrt{p}$ and $M(n) = n^2$

d. $f(p) = Cp\log p$ and $M(n) = n^2$

e. $f(p) = Cp$ and $M(n) = n$

f. $f(p) = p^C$ and $M(n) = n$. Assume $1 < C < 2$

g. $f(p) = p^C$ and $M(n) = n$. Assume $C > 2$.

(a) $(cp)^2/p = c^2 p$

(b) $(c\sqrt{p}\log p)^2/p = c^2\log^2 p$

(c)

$(c\sqrt{p})^2/p = c^2$

(d)

$(cp\log p)^2/p = c^2 p\log^2 p$

(e)

$cp/p = c$

(f) $p^c/p = p^{c-1}$, $1<c<2$

$1 < p^{c-1} < p$

(g)

$p^c/p = p^{c-1}$, $c>2$.

$p^{c-1} > p^1$

most scalable $\longrightarrow$ least scalable

$c=1.5$, $p=50$

$e \to c \to f \to b \to a \to d$

$c=2.5$, $p=50$

$e \to c \to b \to a \to g \to d$