Homework 2

Hai Zhu

due date: September 20, 2017

1. **Question**
A collection of independent tasks are going to be running on some parallel system. Each task has probability 2/3 to take 1 hour to finish on a single computer, and probability 1/3 to take 2 hours to finish. Compute efficiency and speedup compared to a single computer for the following cases:

   (a) 2 tasks and 2 computers, where each computer solves one task.
   (b) 4 tasks and 2 computers, where each computer is given 2 tasks. One task runs immediately after another.
   (c) 4 tasks and 4 computers, where each computer solves one task.

   Solution

   (a) Serial expected time can be computed directly

$$\text{SerTime}(2) = 2 \times (\frac{2}{3} + \frac{1}{3} \times 2) = \frac{8}{3} \tag{1}$$

   And for the case of parallel expected time, two things could happen, and the probabilities are as follows:

$$\begin{cases} P(\text{two tasks all finish in 1 hour}) = & (\frac{2}{3})^2 \\ P(\text{one of them finishes in 2 hours}) = & 1-(\frac{2}{3})^2 \end{cases}$$

   Therefore the expected parallel time is

$$\text{ParTime}(2,2) = 1 \times (\frac{2}{3})^2 + 2 \times (1 - (\frac{2}{3})^2) = \frac{14}{9} \tag{2}$$

   Efficiency and speedup can be calculated accordingly:

$$\begin{aligned} \text{Efficiency}(2,2) &= \frac{\text{SerTime}(2)}{2 \times \text{ParTime}(2,2)} = \frac{6}{7} \\ \text{Speedup}(2,2) &= \frac{\text{SerTime}(2)}{\text{ParTime}(2,2)} = \frac{12}{7} \end{aligned} \tag{3}$$

   (b) Serial expected time can be computed directly

$$\text{SerTime}(4) = 4 \times (\frac{2}{3} + \frac{1}{3} \times 2) = \frac{16}{3} \tag{4}$$

And for the case of parallel expected time, three things could happen, and the probabilities are as follows:

$$\begin{cases} P(\text{parallel time is 2 hours}) = & (\frac{2}{3})^4 \text{ (every task finishes in 1 hour)} \\ P(\text{parallel time is 4 hours}) = & (\frac{1}{3})^2 \times 2 - (\frac{1}{3})^4 \\ & \text{(at least 2 tasks on same computer finish in 2 hours)} \\ P(\text{parallel time is 3 hours}) = & 1 - \text{(previous pribabilities)} \end{cases}$$

Therefore the expected parallel time is

$$\text{ParTime}(4,2) = 2(\frac{2}{3})^4 + 4[(\frac{1}{3})^2 2 - (\frac{1}{3})^4] + 3[1 - (\frac{2}{3})^4 - ((\frac{1}{3})^2 2 - (\frac{1}{3})^4)] = \frac{244}{81} \quad (5)$$

Efficiency and speedup can be calculated accordingly:

$$\begin{aligned} \text{Efficiency}(4,2) &= \frac{\text{SerTime}(4)}{2 \times \text{ParTime}(4,2)} = \frac{54}{61} \\ \text{Speedup}(4,2) &= \frac{\text{SerTime}(4)}{\text{ParTime}(4,2)} = \frac{108}{61} \end{aligned} \quad (6)$$

(c) Serial expected time is the same as last question

$$\text{SerTime}(4) = \frac{16}{3} \quad (7)$$

And for the case of parallel expected time, three things could happen, and the probabilities are as follows:

$$\begin{cases} P(\text{two tasks all finish in 1 hour}) = & (\frac{2}{3})^4 \\ P(\text{one of them finishes in 2 hours}) = & 1 - (\frac{2}{3})^4 \end{cases}$$

Therefore the expected parallel time is

$$\text{ParTime}(4,4) = 1 \times (\frac{2}{3})^4 + 2 \times (1 - (\frac{2}{3})^4) = \frac{146}{81} \quad (8)$$

Efficiency and speedup can be calculated accordingly:

$$\begin{aligned} \text{Efficiency}(4,4) &= \frac{\text{SerTime}(4)}{4 \times \text{ParTime}(4,4)} = \frac{54}{73} \\ \text{Speedup}(4,4) &= \frac{\text{SerTime}(4)}{\text{ParTime}(4,4)} = \frac{216}{73} \end{aligned} \quad (9)$$

Add-on Question

What happens asymptotically, as $n \to \infty$, if there are 2 computers and n tasks, and each given $n/2$ of the tasks? What happens asymptotically if there are n computers and n tasks?

Solution

If there are 2 computers, and they are running $n/2$ tasks each independently, we could think of this as running $n/2$ tasks in serial on a single computer. And therefore the expected time should be approximately

$$\text{ParTime}(n, 2) \approx \text{SerTime}(n/2) = \frac{2n}{3} \tag{10}$$

Therefore

$$\begin{aligned} \text{Efficiency}(n, 2) &= \frac{\text{SerTime}(n)}{2 \times \text{ParTime}(n, 2)} \to 1 \text{ as } n \to \infty \\ \text{Speedup}(n, 2) &= \frac{\text{SerTime}(n)}{\text{ParTime}(n, 2)} \to 2 \text{ as } n \to \infty \end{aligned} \tag{11}$$

If there are n computers and n tasks, and each computer gets one task to run, two cases could happen

$$\begin{cases} P(\text{all tasks all finish in 1 hour}) = & (\frac{2}{3})^n \\ P(\text{one of them finishes in 2 hours}) = & 1 - (\frac{2}{3})^n \end{cases}$$

$$\text{ParTime}(n, n) = (\frac{2}{3})^n + 2 \times (1 - (\frac{2}{3})^n) \to 2 \text{ as } n \to \infty \tag{12}$$

Therefore we can compute the efficiency and speedup

$$\begin{aligned} \text{Efficiency}(n, n) &= \frac{\text{SerTime}(n)}{n \times \text{ParTime}(n, n)} \to \frac{2}{3} \text{ as } n \to \infty \\ \text{Speedup}(n, n) &= \frac{\text{SerTime}(n)}{\text{ParTime}(n, n)} \to \frac{2n}{3} \text{ as } n \to \infty \end{aligned} \tag{13}$$

2. **Question**

Start with the serial program

```
1      initialize
2      for i = 0 to n-1
3          statement 3
4          for j = 0 to n-1
5              statement 5
6          end for j
7      end for i
8      finalize
```

Suppose this has been parallelized for a distributed memory machine with p processors. Lines 1, 3, 5 and 8 each take a constant amount of time. All processors execute 1, 8, and the outer i-loop, with lines 2, 3, and 7 together taking $\Theta(n)$ time. The inner j-loop is perfectly parallelized, i.e., each processor does n/p iterations. However, the parallelization adds a communication step between lines 6 and 7, taking $\Theta(p)$ time.

(a) How should p grows as a function of n to minimize the time?

(b) Suppose for a given $0 < x < 1$ you want to achieve efficiency $\geq x$ by scaling. Given x, what is the fastest that p can increase with n and still achieve efficiency x, or is it impossible to achieve arbitrarily high efficiency?

Solution

(a) Assume line 1 takes constant c_1 time, line 3 takes constant c_3 time, line 5 takes constant c_5 time, and line 8 takes constant c_8 time. Further we assume the communication step takes $c_c p$ time. Then

$$\begin{aligned} \text{ParTime}(n,p) &= c_1 + n(c_3 + c_5\frac{n}{p} + c_c p) + c_8 \\ &= c_1 + c_8 + c_3 n + c_5\frac{n^2}{p} + c_c np \end{aligned} \tag{14}$$

Let's assume $p = \Theta(n^\alpha) = n^\alpha$ (the extra constant can be absorbed into c_5 and c_c). Then

$$\text{ParTime}(n,p) = c_1 + c_8 + c_3 n + c_5 n^{2-\alpha} + c_c n^{1+\alpha} \tag{15}$$

To avoid parallel time growing too fast as $n \to \infty$, we need to balance the power in n in the last two terms in the above expression. Therefore we require $2-\alpha = 1+\alpha$. And this turns out to be

$$\alpha = \frac{1}{2} \tag{16}$$

And this would minimize the time asymptotically, as we scale $n \to \infty$. And the leading power of run time as $n \to \infty$ would be $\Theta(n^{3/2})$.

(b) We can compute the serial time directly

$$\text{SerTime}(n) = c_1 + n(c_3 + c_5 n) + c_8 = c_1 + c_8 + c_3 n + c_5 n^2 \tag{17}$$

Therefore we can get efficiency

$$\begin{aligned} \text{Efficiency}(n,p) &= \frac{\text{SerTime}(n)}{p \times \text{ParTime}(n,p)} \\ &= \frac{c_1 + c_8 + c_3 n + c_5 n^2}{p(c_1 + c_8 + c_3 n + c_5\frac{n^2}{p} + c_c np)} \\ &= \frac{c_1 + c_8 + c_3 n + c_5 n^2}{(c_1 + c_8)p + c_3 np + c_5 n^2 + c_c np^2} \end{aligned} \tag{18}$$

For any given $0 < x < 1$, if we want to achieve efficiency $\geq x$ by scaling, let's look at the dominant terms in the numerator and denominator. As long as p doesn't grow as fast as $\sqrt{n}$, the dominant term will both be $c_5 n^2$. And therefore we can achieve arbitrarily high efficiency by scaling. If $p = \Theta(\sqrt{n}) = c_p n$, Efficiency $\to \frac{c_5}{c_5 + c_c c_p^2} < 1$ as $n \to \infty$.

So the fastest that p can increase with n should be less than $\sqrt{n}$ in order to achieve efficiency x, where x is given $0 < x < 1$.