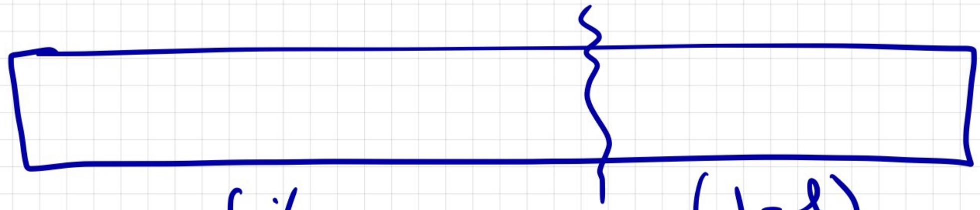
Theoretical Performance Computation

Amdahl's Law

- Serialization limits Performance
- Amdahl's law is an observation that the speed-up one gets from parallelizing the code is limited by the remaining serial part.
- Any remaining serial code will reduce the possible speed-up
- This is why it's important to focus on parallelizing the most time consuming parts, not just the easiest.



→ f

→ n = number of parallel processing unit.

$$\text{Speedup} = \frac{\text{Time on sequential}}{\text{Time on } /\!/ \text{ ex.}} = \frac{\cancel{T}}{\frac{f*\cancel{T}}{n} + (1-f)\cancel{T}}$$

$$= \frac{1}{\frac{f + (1-f)*n}{n}} = \boxed{\frac{n}{f + (1-f)n}}$$

~~1000~~

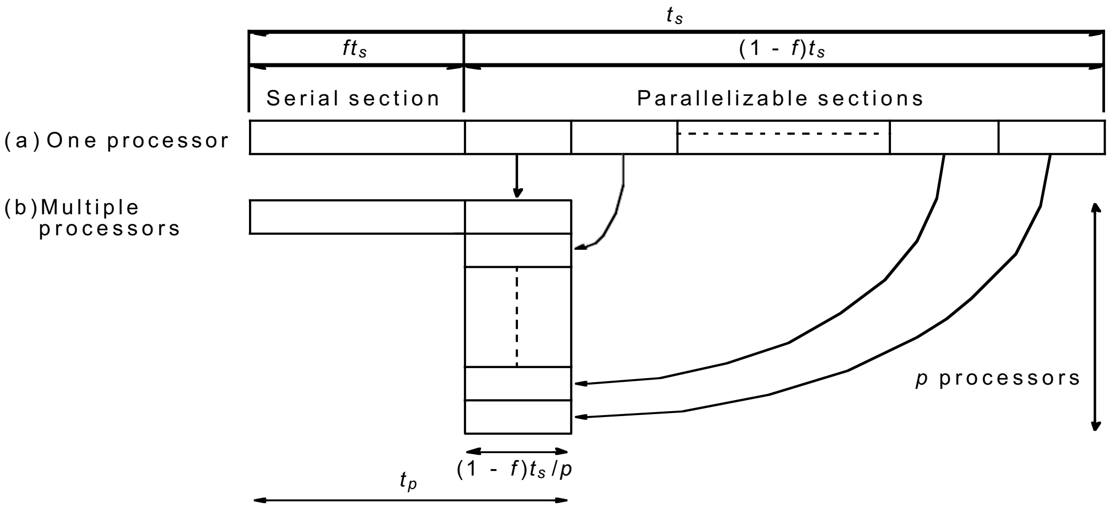
16

1000

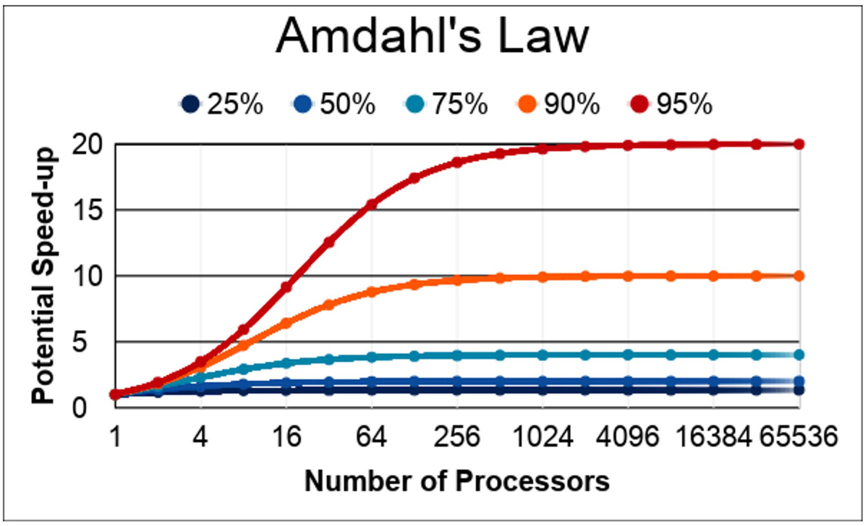
0.96 + 0.04 * 16

~~40 * 0.96~~

Amdahl's Law



Amdahl's Law



Amdahl's Law

- f =fraction of program (algorithm) that is serial and cannot be parallelized
  - Data setup
  - Reading/writing to a single disk file
- Speedup factor is given by:

$$T_s = fT_s + (1-f)T_s$$

$$T_p = fT_s + \frac{(1-f)T_s}{n}$$

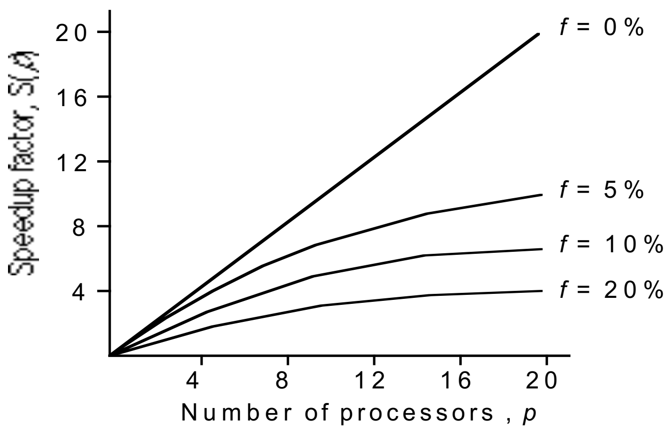
$$S(n) = \frac{T_s}{fT_s + \frac{(1-f)T_s}{n}} = \frac{n}{1+(n-1)f}$$

$$\lim_{n \to \infty} = \frac{1}{f}$$

Note that as n → ∞, the maximum speedup is limited to 1/f.

Speedup Against Number of Processors

- Even with infinite number of processors, maximum speedup limited to 1/f .
- Example: With only 5% of computation being serial, maximum speedup is 20, irrespective of number of processors.



Example of Amdahl's Law (1)

- Suppose that a calculation has a 4% serial portion, what is the limit of speedup on 16 processors?
  - 16/(1 +(16 – 1)*.04) = 10
  - What is the maximum speedup?
    - 1/0.04 = 25

Example of Amdahl's Law (2)

- 95% of a program's execution time occurs inside a loop that can be executed in parallel. What is the maximum speedup we should expect from a parallel version of the program executing on 8 CPUs?

$$\psi \leq \frac{1}{0.05 + (1 - 0.05)/8} \cong 5.9$$

Example of Amdahl's Law (3)

- 20% of a program's execution time is spent within inherently sequential code. What is the limit to the speedup achievable by a parallel version of the program?

$$\lim_{p \to \infty} \frac{1}{0.2 + (1 - 0.2)/p} = \frac{1}{0.2} = 5$$

Example of Amdahl's Law (4)

- What's the maximum speed-up that can be obtained by parallelizing 50% of the code?
- (1 / 100% - 50%) = (1 / 1.0 - 0.50) = 2.0X
- What's the maximum speed-up that can be obtained by parallelizing 25% of the code?
- (1 / 100% - 25%) = (1 / 1.0 - 0.25) = 1.3X
- What's the maximum speed-up that can be obtained by parallelizing 90% of the code?
- (1 / 100% - 90%) = (1 / 1.0 - 0.90) = 10.0X