Performance analysis

Jeremy Iverson

College of Saint Benedict & Saint John’s University

Performance analysis

- how do we reason about parallel algorithms?
- how can we compare two algorithms and determine which is better?
- how do we measure improvement?

Performance metrics

- execution time (T_p)
- speedup (S)
- efficiency (E)
- cost (C)

Execution time

Serial (T_s)

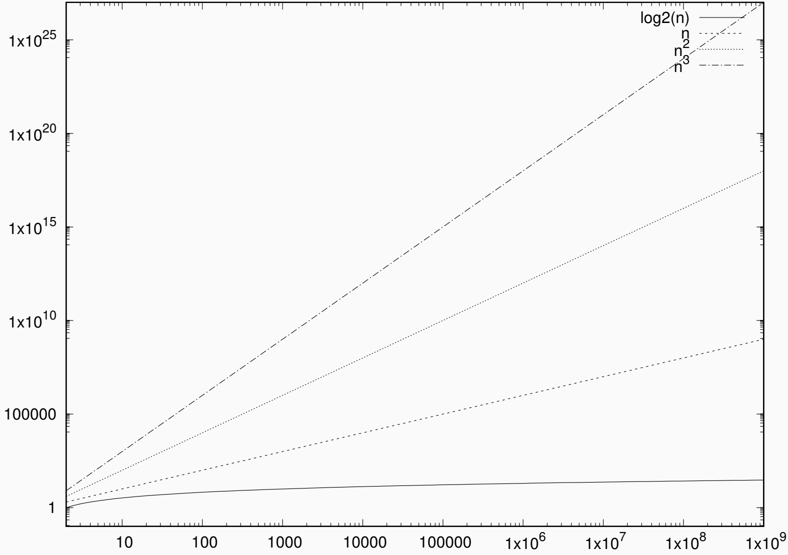
- time elapsed between beginning and end of execution

Parallel (T_p)

- time elapsed between beginning of execution and the moment the last processing element finishes execution

- Adding numbers
- Dot-product
- Matrix-vector multiplication
- Matrix-matrix multiplication

Execution time



Speedup

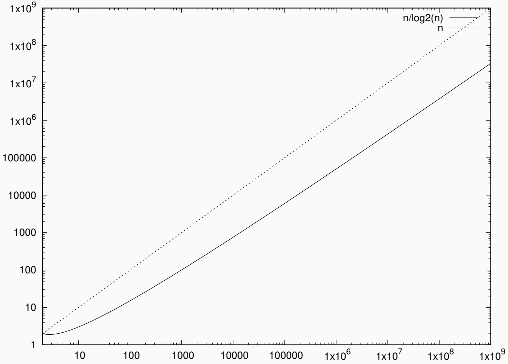
Speedup ($S = T_s / T_p$)

- the ratio of time taken to solve a problem on a single processing element to the time required to solve the same problem on a parallel computer with p processing elements

Speedup

Speedup ($S = T_s/T_p$)

- the ratio of time taken to solve a problem on a single processing element to the time required to solve the same problem on a parallel computer with p processing elements



Efficiency

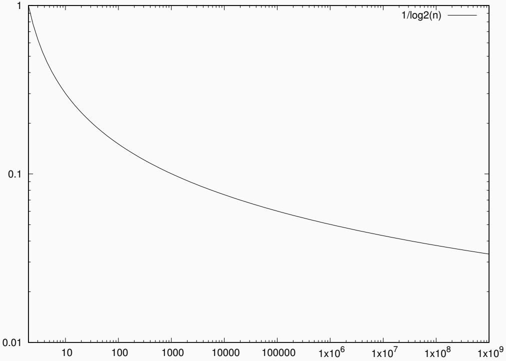
Efficiency ($E = S/p$)

- the ratio of speedup to the number of processing elements — the fraction of time for which a processing element is usefully employed

Efficiency

Efficiency ($E = S/p$)

- the ratio of speedup to the number of processing elements — the fraction of time for which a processing element is usefully employed



Cost

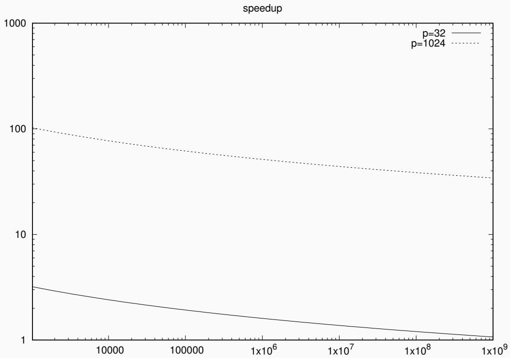
Cost ($C = pT_p$)

- the sum of the time spent by all processing elements solving the problem
- *cost-optimal* if $C = T_s$

Cost

Cost ($C = pT_p$)

- the sum of the time spent by all processing elements solving the problem
- *cost-optimal* if $C = T_s$



Exercise — vector addition

$p = n$ — *not cost-optimal*

- $T_p = \Theta(\log n)$
- $S = \Theta(\frac{n}{\log n})$
- $E = \Theta(\frac{1}{\log n})$
- $C = \Theta(n \log n)$

$p > n$ — too many processing elements, use less

$p < n$ — ?

Exercise — vector addition

$p = n$ — *not cost-optimal*

- $T_p = \Theta(\log n)$
- $S = \Theta(\frac{n}{\log n})$
- $E = \Theta(\frac{1}{\log n})$
- $C = \Theta(n \log n)$

$p > n$ — too many processing elements, use less

$n < p$ — *not cost-optimal?*

- $T_p = \Theta(\frac{n}{p} + \log p)$
- $S = \Theta(\frac{n}{\frac{n}{p} + \log p})$
- $E = \Theta(\frac{n}{n + \log p})$
- $C = \Theta(n + p \log p)$

Exercise — vector addition

$p = n$ — *not cost-optimal*

- $T_p = \Theta(\log n)$
- $S = \Theta(\frac{n}{\log n})$
- $E = \Theta(\frac{1}{\log n})$
- $C = \Theta(n \log n)$

$p > n$ — too many processing elements, use less

$n < p$ — *cost-optimal iff* $n = \Theta(p \log p)$

- $T_p = \Theta(\frac{n}{p} + \log p)$
- $S = \Theta(\frac{n}{\frac{n}{p} + \log p})$
- $E = \Theta(\frac{n}{n + \log p})$
- $C = \Theta(n + p \log p)$