Analyzing Scalability of Parallel Algorithm and Architectures

Vipin Kumar and Anshul Gupta
Objectives of this paper

- 1. Survey various metrics
- 2. Relations between these metrics
- 3. Critically assess these metrics
- 4. Directions for future research
Typical metrics

- Amdahl's law
- Isoefficiency function
- Gustafson
- Zorbas
- Isospeed
- Sizeup
Definition

- $W$ is problem size
- $T_p(W,p)$ is Parallel execution time on $p$ processors for $W$ problem size
- $T_s(W)$ is Serial execution time for $W$ problem size
- $T_0(W,p)$ is Overhead
- $T_0 = pT_p - T_s$
- Efficiency $E = T_s/pT_p = 1/(1+T_0/T_s)$
Amdahl's law

- If $s$ is serial fraction in an algorithm.
  Then the maximum speedup is bounded by $1/s$. 
Isoefficiency

- Solve fixed size problem. Efficiency decrease when processor increase.
- If the problem size increase on fixed number of processors, then the efficiency increase.
- If the system is 'scalable parallel system'
  Efficiency $E$ can be maintained at some fixed value for increasing of number of processors and problem size.
Isoefficiency

- $f_E(p)$ is Isoefficiency function
- $f_E(p)$ return problem size that can maintain efficiency $E$ on number of $p$ processors.
- If $f_E(p)$ grows exponential, then it's poorly scalable
- If $f_E(p)$ grows linearly, then it's highly scalable
**Gustafson**

- Scaled speedup
- Speedup obtained when the problem size is increased linearly with the number of processors.
- If speedup curve is good (close to linear)
then the system is scalable system.
• Gustafson

• Two ways to increase problem size
  1. the size of memory be used.
  2. the size of problem growths with $p$ subject to an upper bound on execution time.
This metric considers the system is ideally scalable if the system's overhead function remains constant when the problem size is increased sufficiently fast with respect to the number of processors.
Isospeed

- Isospeed measure trying to find problem size $W'$, for the average unit speed of computation remain constant when number of processors increased to $p'$ from $p$
- $\text{isospeed}(p, p') = \frac{p'W}{pW'}$
- For perfectly parallelizable algorithm with no communication $\text{isospeed}(p, p') = 1$
- $W' = \frac{p'W}{p}$
Sizeup

- Unfair measure
- Sizeup is ratio of
  Size of problem solved on the parallel computer
  -----------------------------------
  Size of problem solved on the sequential computer in fixed time
• Is there exists one measure that is better than all others?

• NO
Examples

- If problem size fixed, and one is trying to solve by increase number of processors, in this case, try to find optimal number of processors, Amdahl's law is good measure.

- If number of processor is fixed, then use isoefficiency function to find best problem size is good measure.
Hardware Factor

- Improve technology in only one direction may not be a wise idea.
- Increasing speed of the processors alone without improving the communication speed will result in diminishing returns in terms of overall speedup and efficiency.
Thoughts

- How to define problem size
- How to increase problem size
Ideas

- Gustafson's suggestion:
  1. memory
  2. execution time on sequential computer

- Others:
  1. If there are iterations, increase iteration.
Conclusion

- No single scalability metric would be better than all others
- Different measures will be useful in different contexts and further research is needed along several directions
- Hardware cost factors in the scalability analysis is important but still very preliminary