

CSCI 6360: Parallel Computing Lecture Summary - 3

Anirban Das (dasa2@rpi.edu)

January 23, 2018

Amdahl's Law in the Multicore Era

In this paper, authors Hill and Marty talks about the applicability of 'Amdahl's Law' model on today's multicore / multiprocessor systems. The authors have also put forward their own model, to complement the Amdahl's model, where they have focused on the performance of the entire chip rather than looking at performance of individual cores.

Amdahl's law is a formula which gives us the theoretical speedup of the execution time obtained, as a result of enhancement of the system resources, more specifically if the code is run in processors in parallel. So basically there are three ways in which one can speed up execution of the program, run serial part faster i.e. make $(1 - f)$ bigger, or make s bigger, or make f bigger i.e. make the code more parallelizable. However, the law is originally defined only on single core processors working in parallel. It also assumes that there is no scheduling overhead. It further assumes that the fraction of the code that is parallelizable is fixed.

Therefore it is not applicable on the multicore processors. The authors, have introduced the parameter r to address this issue, where resources of r base core equivalents (BCE) can be combined together to get powerful core with sequential performance $perf(r)$. The authors then talks about different ways of allocating this resources in this framework: symmetric (all cores have same cost), asymmetric (all cores might not have same cost) and dynamic (dynamically combining cores as needed).

Performance measures based on the modified model revealed that for smaller systems with 16BCE, distributing the BCEs in many weaker cores i.e. increasing parallelism increases performance than that with few strong cores. Asymmetric multicore performance is better than the symmetric case, where it is possible to assign the few stronger core to the sequential tasks at the same time maintaining enough weaker cores to execute the parallel part of the program. For denser systems, like the 256BCE system, in almost all the values of f , there is a sweet spot, where the speedup reaches maximum between the two extreme values of f .

It is very difficult to distribute parts of code having equal load in a fixed asymmetric configuration of BCEs, but the authors have acknowledged the omission of such overhead and scheduling cost in their performance model. They have assumed the load balancing cost as 0.

However, hypothetically, the dynamic multicore chips can offer best performance among all with identical $perf(r)$. This, as the authors point out, is mainly because in these chips it is possible to allocate the resources together to form stronger cores or free them for parallel execution on the fly.

Overall this was a very insightful paper giving importance on the fact that depending on the workload the performance of multicore systems as per Amdahl's Law paradigm might vary. It suggests that with better models and better software/ hardware paradigms, such speedup potentials as described in the paper can be achieved.