Universidade de Aveiro

Mestrado em Engenharia de Computadores e Telemática

Arquitecturas de Alto Desempenho

Performance Evaluation

Academic year 2021/2022 Exercises adapted from *Computer Architecture: a Quantitative Approach*

1. Your company is trying to decide between purchasing Opteron or Itanium 2 processors for it new computer system. You have analyzed the applications to be run there and you have concluded that 60% of time the system will be running applications similar to `wupwise`, 20% of the time applications similar to `ammp` and 20% of the time applications similar to `apsi`.

SPECfp2000 execution times (s) for Sun Ultra 5 (reference computer) and execution times and SPECRatios for AMD Opteron and Intel Itanium 2

Benchmark	Ultra 5 exec time (s)	Opteron exec time (s)	SPECRatio	Itanium 2 exec time (s)	SPECRatio	Opteron/Itanium 2 exec time	Itanium 2/Opteron SPECRatio
wupwise	1 600	51,5	31,06	56,1	28,53	0,92	0,92
ammp	2 200	136,0	16,14	132,0	16,63	1,03	1,03
apsi	2 600	150,0	17,36	231,0	11,27	0,65	0,65
...							
geo mean			20,86		27,12	1,30	1,30

1.1. If your selection criterion was just based on overall SPEC performance for SPECfp2000, which processor would you choose and why?

1.2. What is the weighted average of execution time ratios for this particular mix of applications in the Opteron and in the Itanium 2?

1.3. What is the speed up of the Opteron over the Itanium 2?

2. The company you work for has bought a IntelCore i5 dual-core processor and your boss has challenged you to optimize the code for the processor. You are supposed to run two independent applications on it whose resource requirements are unequal. The first application requires 80% of the resources of the computer system, while the second only requires 20%. Assume that when you parallelize a portion of the code, the speed up for that portion is 2.

2.1. Given that 40% of the first application is parallelizable, how much speed up would this application observe if run in isolation?

2.2. Given that 99% of the second application is parallelizable, how much speed up would this application observe if run in isolation?

2.3. Given that 40% of the first application is parallelizable, how much overall speed up would you observe if you parallelized it and run it together with the second application?

2.4. Given that 99% of the second application is parallelizable, how much overall speed up would you observe if you parallelized it and run it together with the first application?

2.5. Given that 40% of the first application and 99% of the second application are parallelizable, how much overall speed up would you observe if you parallelized both of them and run them together?

3. When parallelizing an application, the maximum speed up that can be attained is equal to the number of processors running the application. This value, however, is limited by two factors: the amount of the application that can be run concurrently and the communication cost among the processors. Amdahl's law takes into account the former, but not the latter.

   3.1. What is the speed up with N processors if as much as 80% of the application is concurrent and the communication cost is ignored?

   3.2. What is the speed up with 8 processors if, for every processor added, the communication overhead is increased by 0.5% of the original execution time?

   3.3. What is the speed up with 8 processors if, every time the number of processors is doubled, the communication overhead is increased by 0.5% of the original execution time?

   3.4. What is the speed up with N processors if, every time the number of processors is doubled, the communication overhead is increased by 0.5% of the original execution time?

   3.5. Write the general equation that solves the following problem: What is the number of processors that gives rise to the highest speed up for the execution of an application in which P% of the original execution time is concurrent and, every time the number of processors is doubled, the communication overhead is increased by 0.5% of the original execution time?

4. In a warehouse-scale computer system such as that used by Amazon or eBay, a single failure in a node computer does not cause the entire system to crash. Instead, the total number of requests that can be served at any one time will be just reduced.

Unavailability costs for some businesses in a warehouse-scale computer, assuming that downtime is distributed uniformly

		Annual losses with downtime of		
Business	**Downtime cost per hour**	**1% (87.6 hrs/yr)**	**0.5% (43.8 hrs/yr)**	**0.1% (8.8 hrs/yr)**
brokerage operations	\$ 6 450 000	\$ 565 000 000	\$283 000 000	\$ 56 500 000
home shopping channel	\$113 000	\$ 9 900 000	\$ 4 900 000	\$ 1 000 000
catalog sales center	\$ 90 000	\$ 7 900 000	\$ 3 900 000	\$ 800 000
airline reservation center	\$ 89 000	\$ 7 900 000	\$ 3 900 000	\$ 800 000

   4.1. If a company has 10 000 computers, forming a warehouse-scale computer system, each with a MTTF of 35 days and a MTTR of 1 day, and if it only experiences catastrophic failure when at least 1/3 of its computer nodes fail, what is the MTTF of the warehouse-scale computer?

   4.2. Suppose it costs an extra \$200 per computer node to double its MTTF. Would it be a sound business decision to do this? Present your reasoning.

   4.3. The table above illustrates the downtime cost for some businesses assuming that the cost is invariant throughout the year. For retailers, however, the Christmas season is the most profitable (and therefore is the most costly time to loose sales). If a catalog sales center has twice as much traffic in the fourth quarter as in any other quarter, what is the average cost of downtime per hour during both the fourth quarter and the rest of the year?