Parallel Computing:

PAR-Q1: Define the following technical terms:
(Be thorough and general. An example is not a definition.)

Cluster (in high-performance resp. big-data computing)

Ans: Aggregate LOTS of computers are called Clusters l Need scalable parallel algorithms l Need to exploit multiple levels of parallelism l Fault tolerance (Only in Big data computing, not in HPC)

Parallel work (of a parallel algorithm)

Ans: Parallel work Wa (n) of algorithm A on an input of size n, maximum number of instructions performed by all procs during execution of A, where in each (parallel) time step as many processors are available as needed to execute the step in constant time.

Parallel speed-up

Ans: the factor by how much faster we can solve a problem with p

processors than with 1 processor, usually in range (0…p)

Communication latency (for sending a message from node Pi to node Pj)

Ans: The total time taken for sending a message from node Pi to node Pj is called communication latency. High latency favors larger transfer block sizes (cache lines, memory pages, file blocks, messages)

Temporal data locality

Ans: Temporal locality – re-access same data element multiple times within a short time interval

Dynamic task scheduling

Ans: Each newly created task is dispatched at runtime to an available worker processor

PAR-Q2: Explain the following parallel algorithmic paradigm: Parallel Divide-and-Conquer.

Ans: Parallel Divide-and-Conquer: l Recursive calls can be done in parallel. l Parallelize, if possible, also the divide and combine phase. l Switch to sequential divide-and-conquer when enough parallel tasks have been created.

PAR-Q3: Discuss the performance effects of using large vs. small packet sizes in streaming.

Ans: when the packet is large the communication latency is high. In case of small packet, the idle time for the processors is high.

PAR-Q4: Why should servers (cluster nodes) in datacenters that are running I/O-intensive tasks (such as file/database accesses) get (many) more tasks to run than they have cores?

Ans: As parallel programing involves many processors the waiting time for processors will be high if I/O intensive tasks get fewer tasks to run than they have cores.

PAR-Q5: In skeleton programming, which skeleton will you need to use for computing the maximum element in a large array? Sketch the resulting pseudo code (explain your code).

Ans: the skeleton is MapReduce and the code ….see first question of lab one.

PAR-Q6: Describe the advantages/strengths and the drawbacks/limitations of high-level parallel programming using algorithmic skeletons.

Ans: Advantages/Strengths: Abstraction, hiding complexity (parallelism and low-level programming). Enforces structuring, restricted set of constructs. Parallelization for free. Easier to analyze and transform.

Drawbacks/Limitations: Requires complete understanding and rewriting of a computation. Available skeleton set does not always fit. May lose some efficiency compared to manual parallelization.

PAR-Q7: Derive Amdahl's Law and give its interpretation.

Ans. Page 8

PAR-Q8: What is the difference between relative and absolute parallel speed-up? Which of these is expected to be higher?

Ans. Page 7

PAR-Q9: The PRAM (Parallel Random Access Machine) computation model has the simplest-possible parallel cost model. Which aspects of a real-world parallel computer does it represent, and which aspects does it abstract from?

Ans. The PRAM cost model has only 1 machine-specific parameter: the number of processors, and it abstracts from scheduling overhead.

PAR-Q10: Which property of streaming computations makes it possible to overlap computation with data transfer?

Ans. Streaming applies pipelining to processing of large (possibly, infinite) data streams from or to memory, network or devices, usually partitioned in fixed-sized data packets, this property of streaming makes it possible to overlap computation with data transfer.

MapReduce

MR-Q1: A MapReduce computation should process 12.8 TB of data in a distributed file with block (shard) size 64MB. How many mapper tasks will be created, by default? (Hint: 1 TB (Terabyte) = 10^12 byte).

Ans. 200,000 mappers. (10^12*12.8/64*10^6)