

NWEN 303 Assignment 3

Daniel Braithwaite

September 21, 2016

1 Question 1

1.1 a

A weather simulation model consists of a number of equations. These equations can be separated and solved independently, therefore a parallel system would be well suited. However we could also think of it as a concurrent system. In this situation we could be running multiple simulations for different locations at the same time.

1.2 b

Concurrent would be the best choice for this problem. The server will receive a number of requests from users, we want to be able to serve these requests at the same time. Perhaps this could also be seen as a Parallel system i.e. We have the task of serving all requests and we are breaking this big task up into a number of smaller tasks (each smaller task for responding to one request). However it doesn't seem natural to say that all the requests are one big task as we don't start with all the tasks we just receive them over time.

1.3 c

A distributed system is a natural fit here. We have a number of different physical parts (the different airlines websites/servers) and they are connected by some network (the internet) so the booking system can present all the prices/flights in one place.

1.4 d

latency is defined by the time taken for one bit to travel from the source to the destination.

bandwidth is defined by the number of bits that can travel between the source and destination in a given time period.

1.5 e

The speed up is defined as $speedup = \frac{T(1)}{T(N)}$. Therefore we have the speed up of the program as $speedup = \frac{T(1)}{T(10)} = \frac{1}{2} = 0.5$
This sub optimal speedup could be caused by the overhead of the parallelization combined with the program not properly utilizing the parallelization.

2 Question 2

The key difference between the two is where the memory is. With a SMP system we will have a number of identical processors connected to a single shared memory resource, this would mean that the time to look up something in memory should be the same across all the processes. Compared to NUMA system where each processor will have its own memory resource, all the processes are linked by a communication network. So all the memory is still accessible from each of the processors. However this means that if a processor is looking something up in its local memory its faster, however looking something up in another processes memory would take longer.

A NUMA system would be more effective in a situation where processes where mostly only looking at there local memory, and a SMP system would likely work better if there was a lot of sharing between the processes.

3 Question 3

P1's request is likely to take less time as it is accessing its own memory resource so it doesn't have to make any request over the network linking the processors. Where as P3 will have to make a request over the network as it is trying to access P2's memory.

4 Question 4

Say we have a NUMA system with three modules. If P1 wants to read to P3's memory then the following happens

1. R1 sends request
2. R3 sees request gets data from M3.
3. Puts data on network
4. R1 reads the data to P1

So to write the process would be the same just in reverse. Say P1 wants to write to P3, so R1 would start by sending the data to R3 which would then read the data and save it to M3. In ether case we are doing only one read and one write to the network so no most likely it wouldn't take any longer.

5 Question 5

We must wait for the pipeline to be full of rows, for a given node to receive a column vector and for that node to pass the column on before work can begin. Work can only begin at time t_7 in the example. As this is the first point that a part of the output is computed. Before time t_7 the only thing happening was information being passed down the pipeline.

6 Question 6

We start by passing rows of the first matrix through the pipeline, until the pipeline is full, at this stage the coordinator will be feeding rows from the top of the matrix into the pipeline. When passing the rows down the pipeline we want to do it so the last worker has the first row to enter the pipeline, this makes reconstructing the final solution easier (as we can just append every row leaving the pipeline to the bottom of the solution).

Then just like before the coordinator will pass down the columns of the second matrix with each worker computing the product before passing the column on. Once all the computations have been finished we pass the resultant rows to the coordinator. Because of the way we filled the pipeline as the coordinator empties it we can just append each of the rows to the bottom of our total solution. Then we repeat this process for the rest of the rows.