

COMP464 - Homework 2 – Report

Bruno G M Correa – bcorrea@luc.edu (brunogmc@gmail.com)

There are two main goals that we may be pursuing when we try to parallelize code. In one of them we try to reduce the amount of time taken to run a program. This is called strong scaling. In the other one we try to maintain constant the amount of time taken to run a program by proportionally increasing the number of processors (or threads). This is called weak scaling.

Considering the information gathered during many executions of the N-body problem, I think we can consider that it has a really good strong parallel scalability. 98% of efficiency could be achieved when running the program with 10000 bodies and scaling up the number of threads from 1 to 8. As we can see in the table, the efficiency drops when we try to use more than 8 threads.

We can notice from the charts that when we have a small number of bodies, the process of parallelizing the code does not bring us a better efficiency. Instead, the run-time gets worse.

In the case of weak scaling efficiency, the results are not good if we double the number of bodies when doubling the number of threads. This happens because the serial run-time for the N-body algorithm used in this benchmark scales as n^2 (where n is the number of bodies). This means that doubling the number of bodies without increasing the number of threads should lead to a four times slower run-time. So, to achieve a constant run-time when doubling the number of threads we should increase the number of bodies by a factor of $\sqrt{\text{\# of threads}}$. The difference between the efficiency of these two cases can be viewed in the charts. As expected, the weak efficiency is much better when we use the square root factor to define the number of bodies to be used.

Summarizing, we could say that this algorithm scales well but it is dependent upon the number of threads and bodies. There is a strange behavior that drops the efficiency when we use more than 8 threads in the Stampede machine and I was not able to discover what lead us to this behavior, though.

Sources:

Class resources

<https://www.cs.rit.edu/~ark/bcbd/ch09.pdf> pg. 9-2

<https://www.cs.rit.edu/~ark/bcbd/ch10.pdf> pg. 10-2

https://www.sharcnet.ca/help/index.php/Measuring_Parallel_Scaling_Performance