A Robust Grid Computing Architecture on Heterogeneous Mobile Devices for Scientific Computing

Logan Garbarini

Background & Introduction

Mobile computers in the form of cell phones, tablets, gaming consoles, and Personal Digital Assistants (PDAs) are becoming increasingly more powerful and are beginning to rival computers in computational power. Now, many people have several of these powerful devices. However, their hardware and software is vastly different from that of traditional general-purpose computers. If this power could be harnessed, a person's computational resources could be increased substantially (Büsching *et al.*, 2012). Furthermore, these same types of devices and software are starting to be used in more conventional roles. Currently, grid computing applications are designed to run on conventional thick OS platforms like Windows, Mac OS X, or Linux. By harnessing the power of the ever-increasing number of devices into a massively parallel architecture, small organizations or individuals could foreseeably create a fast heterogeneous compute cluster simply by using the multitude of devices at their disposal (Fadika *et al.*, 2012).

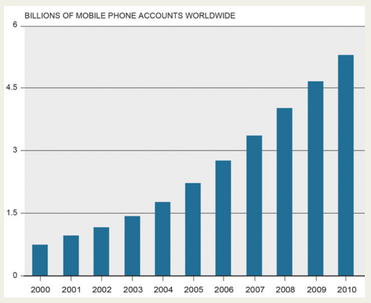


Figure 1: Mobile Phone Adoption Worldwide
Photo Credit: MIT Technology Review

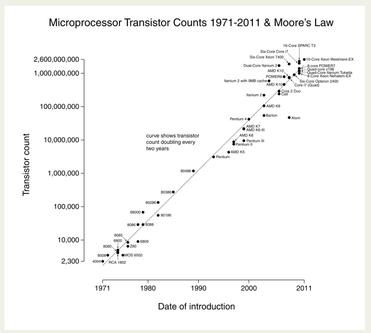
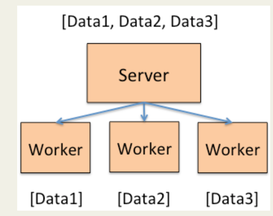


Figure 2: Actual Transistor Count vs. Moore's Law
Photo Credit: Wikimedia Commons

Methods



This experiment used a combination of the Ruby on Rails Web Application Framework and the Java Programming language. These two pieces of software are used for the Server and the Worker systems respectively. The server is in charge of taking the tasks and splitting them into jobs which can be sent to the workers via HTTP in accordance with the REST principles. The workers will then process the jobs and return them through HTTP. Once all the workers have returned all the jobs, the server must process the Job data into the output for a job.

To test this architecture, a heterogeneous set of Android based mobile devices will be used to run a simple integration program in a parallel manner. The integration will be done numerically using Simpson's Rule(*Equation 1*) to generate an approximation of the integral of the function. The partition size *h* that will be used is 0.00001. The speed of integration from 0 to 100, 200, 300, 400 and 500 will be tested.

$$\int_a^b f(x)\,dx \approx h\left[\frac{1}{3}f(x_0) + \frac{4}{3}f(x_1) + \frac{2}{3}f(x_2) + \cdots + \frac{4}{3}f(x_{n-1}) + \frac{1}{3}f(x_n)\right]$$

Equation 1: Simpson's rule was used to test the compute grid

Discussion and Results

The data show that the compute cluster is capable of running the test computations and running through the data and returning a result. The grid is capable of handling progressively larger task sizes so that there will not be a noticeable degradation in speed. Also, the data show that the more workers, the faster the computation can run.

The data show that the computer grid can be used to create a logical computer that can reach better computational speeds that could be achieved with a single device. This investigation shows that the software architecture and systems are capable of handling multiple devices allowing the server device to be freed up for other computations(single core CPU usage <5%).

Once the computations were run, the data was plotted onto a graph(Figure 4) which showed how the Run Time scaled linearly with the number of jobs. Also, the same graph shows the differences in the rate of computations between the different data runs. All three trials showed that the data had a direct linear relationship with $p < 0.02$.

Figure 5 shows how increasing the number of nodes could reduce the job run time. Unfortunately, it was not possible to use more devices because of budget constraints. Also, the Android Virtual Devices that had been planned on being used ran far too slowly to do any kind of real world simulation. The data that was collected showed that there was a very high correlation ($p < 0.02$) when using an exponential regression. This relationship shows how more devices can increase the speed of the compute cluster.

Although the desktop computer is much faster at small values of N, when N is increased the trend shows that a compute cluster of about 14 devices will overtake a single computer and begin to reduce the computation time.

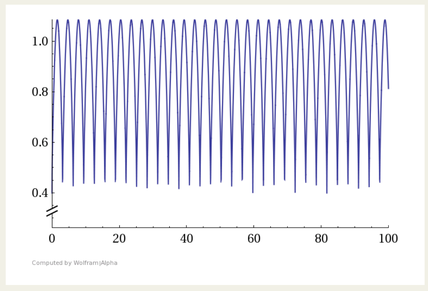


Figure 3: A graph showing the function being integrated
Graph computed by: WolframAlpha

$$\int \frac{e\sqrt{|sin(x)|}}{\sqrt{2\pi}}$$

Equation 2: The function integrated using the Compute Grid and Simpson' Rule

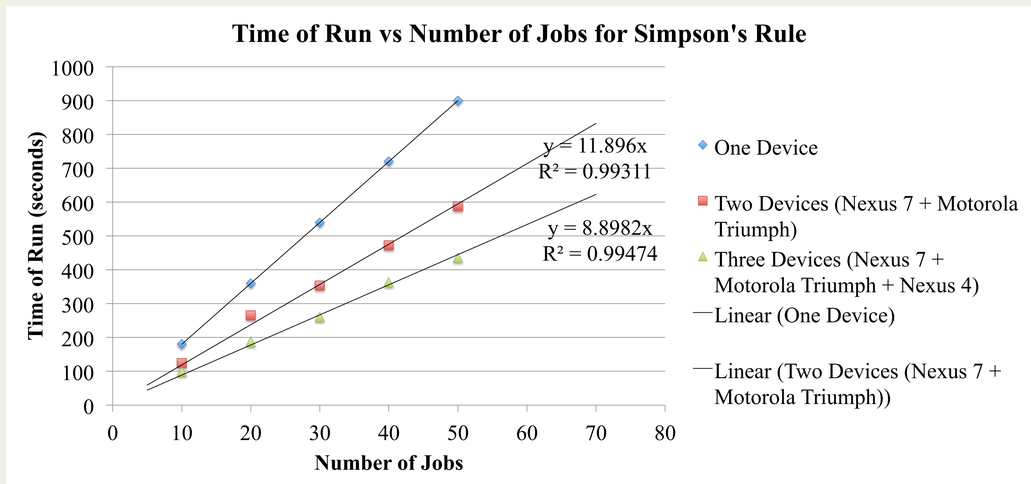


Figure 4: This graph shows how the number of workers has a larger effect on run time as the number of Jobs increases. It also shows that the run time of jobs scales linearly with the number of jobs

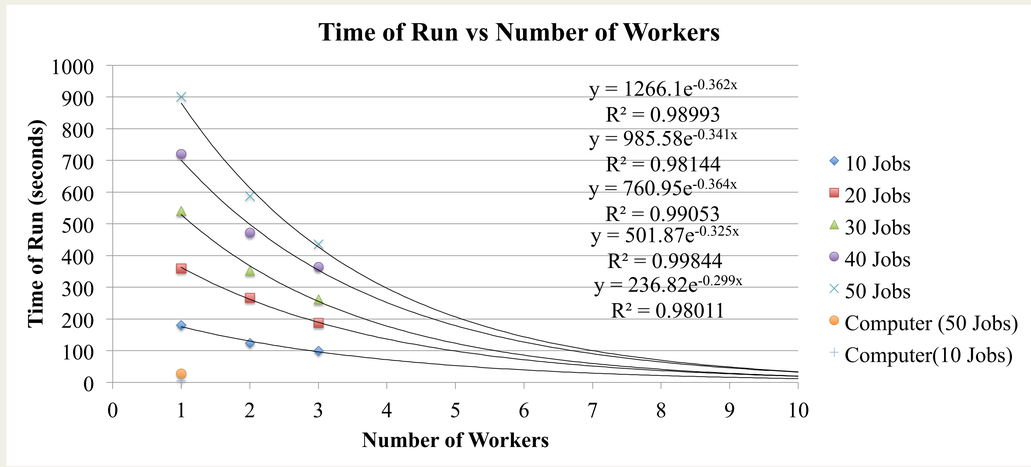


Figure 5: This graph shows how the number of workers affects the run time of the computation. Exponential regression was used for the trend lines.

Conclusion

The ever improving speed and availability of inexpensive Android based mobile devices will allow a new set of mobile grid applications that can harness the power of conventional computers as well as new mobile devices.

This investigation shows that grid computing on mobile phones is possible and provides a proof of concept architecture for running such a grid. This architecture shows that the processing can scale quite linearly and the size of the job does not factor in to the compute time.

Additionally, the data support the conclusion that the compute power of the cluster can be easily increased to handle larger jobs. The data also show that with enough workers, the computational power of the grid could exceed that of a desktop computer.

Mobile grids offer a promising opportunity to improve computation speed using currently existing devices. While a few mobile devices could not rival a desktop or laptop, combining more of them together could yield a very powerful grid. Such grids would allow developing countries to pool resources to have more powerful computers than conventionally possible. Finally, as these devices become more and more powerful and are used in more conventional desktop environments, the applications of these grids will expand further.

Further Research

While this research does a decent job showing how mobile devices could help create a more powerful mobile device landscape, there is much more room for additional research. First, more devices could have been used to examine the information in more depth. Due to budget constraints it was only possible to test with three devices. This limited the ability to extrapolate the data and look for trends.

Second, the application could be written to be multithreaded. During the experiment, it was observed that each multicore device was only using one core to do the computation. If the grid could be written to handle more tasks concurrently, the speed of the grid could have been substantially improved.

To improve the performance even further and to take advantage of mobile processing technology, GPU hardware acceleration could be employed to speed up certain types of computations. This would allow the grid to harness the very powerful graphics processors onboard many newer mobile devices.

An additional improvement that could be made to improve the data would be to try different tasks on the grid to see if certain tasks run faster than others. For example, a Monte Carlo Simulation could be run or image processing techniques could be used.

References

Busching, F., Schildt, S., & Wolf, L. (2012). DroidCluster: Towards Smartphone Cluster Computing – The Streets are Paved with Potential Computer Clusters. In *2012 32nd International Conference on Distributed Computing Systems Workshops (ICDCSW)* (pp. 114 – 117). Presented at the 2012 32nd International Conference on Distributed Computing Systems Workshops (ICDCSW). doi:10.1109/ICDCSW.2012.59

Fadika, Zacharia, Dede, E., Hartog, J., & Govindaraju, M. (2012). MARLA: MapReduce for Heterogeneous Clusters. In *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)* (pp. 49–56). Washington, DC, USA: IEEE Computer Society. doi:10.1109/CCGrid.2012.135

File:Transistor Count and Moore's Law - 2011.svg. (n.d.). In *Wikipedia, the free encyclopedia*. Retrieved from http://en.wikipedia.org/wiki/File:Transistor_Count_and_Moore%27s_Law_-_2011.svg

Mancini, E., Wainer, G., Al-Zoubi, K., & Dalle, O. (2012). Simulation in the Cloud Using Handheld Devices. In *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)* (pp. 867–872). Washington, DC, USA: IEEE Computer Society. doi:10.1109/CCGrid.2012.65

Schwartz, E. (n.d.). The Mobile Device Is Becoming Humankind's Primary Tool (Infographics Feature) | MIT Technology Review. *MIT Technology Review*. Retrieved February 27, 2013, from http://www.technologyreview.com/news/421826/the-mobile-device-is-becoming-humankinds-primary-tool-infographics-feature/

Wolfram Alpha LLC. 2010. Wolfram|Alpha. http://www.wolframalpha.com/input/?i=exp%28sqrt%28abs%28sin%28x%29%29%29%29%2Fsqrt%282*PI%29+0+to+100 Retrieved February 18, 2013