Performance Implications Of Cloud Sourcing Scientific Applications

**Robert L. Cloud *a*, Michael J. Hannon *b*, Suman Silwalc, R. Steven Wingod**

*a,b,c,d The University of Alabama at Birmingham, Birmingham AL, USA, 35294*

Abstract

High-Performance scientific computing applications simulate natural phenomena on a large scale using mathematical models.  Historically, these applications have been primarily targeted towards tightly controlled computing environments composed of massive quantities of commodity hardware components bound together with high performance interconnects (Beowulf type clusters).  The most powerful of these clusters are extremely expensive, and access to their computational time is rare and valuable, but they enable the solution of previously intractable problems.

Recently, a new environment for scientific computing has emerged with the advancement of the Internet economy, which requires large server farms that often have excess storage and computational capacity.  Organizations, such as Amazon.com, have created tools that enable one to leverage the spare capacity of these resources for, e.g., scientific endeavors.  This development, known as cloud computing, enables more democratic access to high performance computational resources; investigators now do not have the burden of purchasing a cluster for their research, but rather can pay for compute instances on demand.

 Our research analyzes the performance gradients of the cloud environments relative to the traditional cluster through the means of two scientific applications. First, we use a well-known discretization of Laplace's equation and measure the time required for it to converge with known boundary conditions on both the cluster and in the cloud. Furthermore, we will attempt to implement an inverse modeling application with experimental data and use this as an example of practically using cloud based resources to further scientific research.

***Keywords:*** *Scientific Computing, Cloud Sourcing, Parallel Computing.*

|  |  |
| --- | --- |
|  |  |

### Introduction

1.1. High Performance Computing

1.2. Cloud Computing

1.3. Application

*1.3.1 Laplace Equation*

*1.3.2 Pressure Pulse Decay*

### 2. Background

2.1. High Performance Computing

2.2. Cloud Computing

2.3. Application

*2.3.1 Laplace Equation*

*2.3.2 Pressure Pulse Decay*

### Methodology

3.1. High Performance Computing

3.2. Cloud Computing

3.3. Application

*3.3.1 Laplace Equation*

*3.3.2 Pressure Pulse Decay*

### 4. Conclusion

### Future Work

Acknowledgments

References

Behnel, S., Bradshaw, R., Citro, C., Dalcin, Lisandro, Seljebotn, D. S., & Smith, K. (2011, March). Cython: The Best of Both Worlds. *Computing in Science & Engineering*, *13*(2), 31-39. doi:10.1109/MCSE.2010.118

Brebner, P. (2011). Performance and cost assessment of cloud services. *Service-Oriented Computing*. Retrieved from http://www.springerlink.com/index/444475421L1X1482.pdf

Dalcin, L, Paz, R, Storti, M, & Delia, J. (2008). MPI for Python: Performance improvements and MPI-2 extensions. *Journal of Parallel and Distributed Computing*, *68*(5), 655-662. doi:10.1016/j.jpdc.2007.09.005

Dalcín, L., Paz, Rodrigo, & Storti, Mario. (2005). MPI for Python. *Journal of Parallel and Distributed Computing*, *65*(9), 1108-1115. doi:10.1016/j.jpdc.2005.03.010

Kloeckner, A. (2010). Scripting GPUs with PyOpenCL. *Scipy 2010*. Retrieved from http://conference.scipy.org/scipy2010/slides/tutorials/andreas\_kloeckner\_pyopencl.pdf

Masini, S., & Bientinesi, P. (2010). High-Performance Parallel Computations using Python as High-Level Language. *EuroPar 2010*. Retrieved from http://www.aices.rwth-aachen.de:8080/aices/preprint/documents/AICES-2010-08-01.pdf

Rabenseifner, R., Hager, G., & Jost, G. (2009). Hybrid MPI/OpenMP parallel programming on clusters of multi-core SMP nodes. *Parallel, Distributed and Network-based Processing, 2009 17th Euromicro International Conference on* (pp. 427–436). IEEE. Retrieved from http://ieeexplore.ieee.org/xpls/abs\_all.jsp?arnumber=4912964

Seljebotn, D. S. (2009). Fast numerical computations with Cython. *Proceedings of the 8th Python in Science Conference (SciPy 2009)* (pp. 15-22).

Wilbers, I. M., Langtangen, H. P., & Ødegård, Å. (2009). Using Cython to Speed up Numerical Python Programs. In H. I. Andersson (Ed.), *Proceedings of MekIT* (Vol. 9, pp. 495-512). Retrieved from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Using+Cython+to+Speed+up+Numerical+Python+Programs#0