Undersubscription: An Underutilized Factor in High-Performance Computing

LIBERTY UNIVERSITY

Reid Prichard and Dr. Wayne Strasser

Background

- High Performance Computing (HPC) uses multiple computers working together to solve challenging computational problems.
- It can take **weeks or months** and cost **tens or hundreds of thousands of dollars** to perform a scientific simulation.
- Consequently, even a small increase in efficiency can save weeks of time and tens of thousands of dollars.
- Despite the huge motivation to get the most out of HPC hardware, one substantial factor remains the topic of little discussion: undersubscription.

Introduction

- As shown in Figure 1, each machine in an HPC cluster has a certain number of CPU cores it can use to perform calculations.
- Intuitively, you would think that using all available cores would be the fastest way to complete a calculation. If you are paying for the whole machine, you should use the whole machine – right?
- Surprisingly, the answer to this question is often "no."
- Leaving some cores unused is called "undersubscription," and existing research shows that doing so can improve speed.
- Chadha et al. identified three mechanisms by which undersubscription might cause a speedup:
  - Limited scalability. You might expect that, as you continue to add cores, each new core would do just as much work as the previous ones.
  - A bottleneck in shared resources. CPU cores must communicate with memory stored locally and stored on other machines. These links have a limited bandwidth; if this link is saturated, it becomes a limiting factor.
  - Dynamic frequency scaling. It requires more power to run a processor faster. However, thermal and electrical constraints limit the amount of power than can flow through a CPU. As a result, leaving some cores idle means that the rest can operate at a higher speed.
- In this research, we explore the effects of undersubscription on a type of simulation called Computational Fluid Dynamics (CFD).
- We demonstrate that in some cases it can be beneficial to leave up to half of a machine's cores unused, reaping speed boosts of up to 100%.

Methods

- Benchmarking data were obtained by running the Computational Fluid Dynamics software, ANSYS Fluent, on cloud-based HPC hardware.
- Three different CFD models [6,7,8] were tested, and each of those three models was tested with varying computational rigor.
- Three different hardware types and up to 2000 CPU cores were tested.
- These three hardware types differ in crucial ways:
  - Different local and nonlocal memory bandwidth per core.
  - Different numbers of cores per node.
  - Different dynamic frequency scaling behavior.

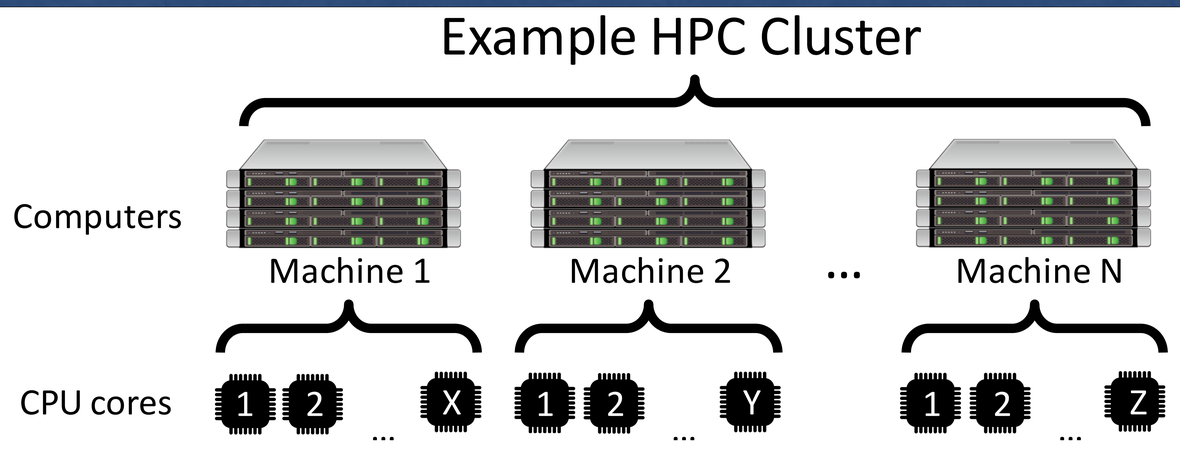

Figure 1: An illustration of HPC hierarchy.

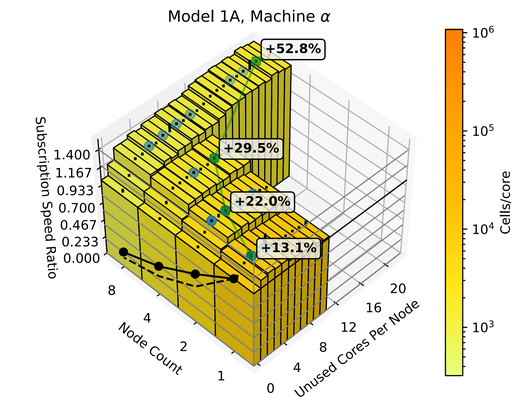

Figure 2: Undersubscription behavior for the smallest model with one machine type. Taller bars are better. With this combination, undersubscription increased speed by over 50% with the largest node count.

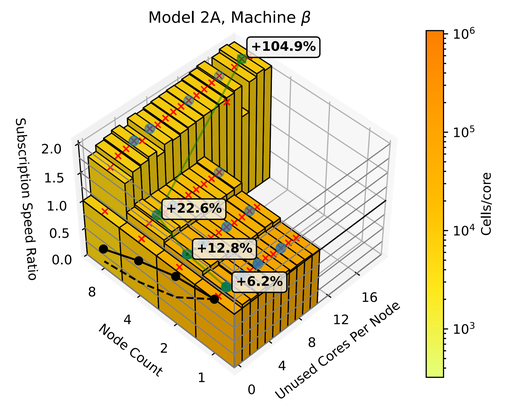

Figure 3: Undersubscription behavior for a different model with another machine type. With 8 nodes, undersubscription more than doubled speed.

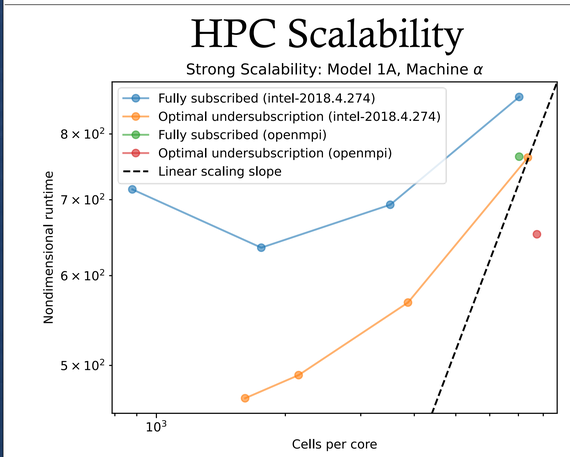

Figure 4: A depiction of scalability behavior with and without undersubscription. Ideally, this would be a straight line parallel to the dashed line in the plot.

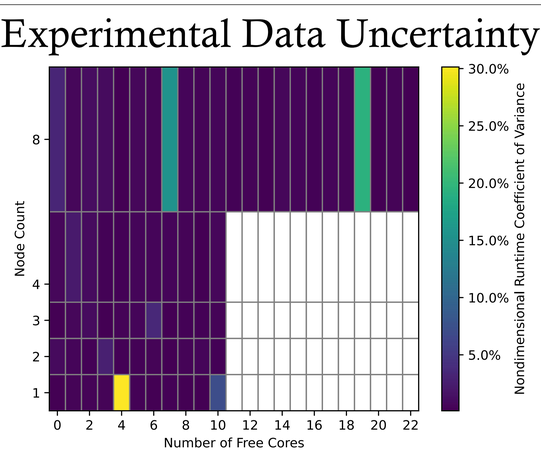

Figure 5: The coefficient of variance for the data shown in Figure 2.

Results

- We found that undersubscription was beneficial in all cases.
- With few nodes, the speedup was minimal: in some cases, less than 1%. At larger node counts, we commonly saw 10-50% speedup, with some cases exceeding 50%.
- As node count increases, the optimum number of free cores monotonically increases. Sometimes this change is gradual, and sometimes it jumps sharply.
- We found that even core counts tend to perform better. This is likely a consequence of the hardware tested consisting of dual-socket machines.
- We found that undersubscription qualitatively changed the scalability of the system. Where scalability became negative with fully-subscribed nodes, it remained positive with undersubscription.
- Scatter in data was minimal. Except for a few outliers, coefficient of variance remained under 10% for all tests.

Conclusions & Future Work

- It is a groundbreaking, novel discovery to learn the immense impact undersubscription can have on HPC performance.
- Furthermore, the trends we discovered are useful to inform optimization testing:
  - The optimum number of free cores increases monotonically as node count increases.
  - Even core counts generally outperform odd core counts.
- It would be even more useful to be able to predict the optimum undersubscription without requiring testing.
- Moving forward, we hope to quantify the patterns seen in undersubscription behavior to allow prediction of the optimum number of free cores.

References

1] Chadha, G., Mahlke, S., and Narayanasamy, S. When less is more (LIMO):controlled parallelism for improved efficiency. In Proceedings of the 2012 international conference on Compilers, architectures and synthesis for embedded systems, ACM, pp. 141–150.

[2] Heirman, W., Carlson, T. E., Van Craeynest, K., Hur, I., Jaleel, A., and Eeckhout, L. Undersubscribed threading on clustered cache architectures. In 2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA), IEEE, pp. 678–689.

[3] Schonherr, J. H., Richling, J., and Heiss, H.-U. Dynamic teams in OpenMP. In 2010 22nd International Symposium on Computer Architecture and High Performance Computing, IEEE, pp. 231–237.

[4] Schwarzrock, J., de A. Rocha, H. M. G., Beck, A. C. S., and Lorenzon, A. F. Effective exploration of thread throttling and thread/page mapping on NUMA systems. In 2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), IEEE, pp. 239–246.

[5] Wang, W., Davidson, J. W., and Soffa, M. L. Predicting the memory bandwidth and optimal core allocations for multi-threaded applications on large-scale NUMA machines. In 2016 IEEE International Symposium on High Performance Computer Architecture (HPCA), IEEE, pp. 419–431.

[6] Wilson, D. M., and Strasser, W., 2021, "SMART ATOMIZATION: IMPLEMENTATION OF PID CONTROL IN BIOSLUDGE ATOMIZER," *Proceeding of 5-6th Thermal and Fluids Engineering Conference (TFEC)*, Begellhouse, Virtual, pp. 1149–1158.

[7] Strasser, W., 2021, "TOWARD ATOMIZATION FOR GREEN ENERGY: VISCOUS SLURRY CORE DISRUPTION BY FEED INVERSION," Atomiz Spr, **31**(6), pp. 23–43.

[8] Turman, E. M., and Strasser, W., 2021, "REVEALING ETHYLENE HOT SPOTS IN LOW DENSITY POLYETHYLENE REACTOR," *Proceeding of 5-6th Thermal and Fluids Engineering Conference (TFEC)*, Begellhouse, Virtual, pp. 705–721.