Measurements of performance of hardware and general purpose classical molecular dynamics simulation software

José O. Sotero-Esteva
Universidad de Puerto Rico en Humacao,
Call box 860, Humacao, Puerto Rico 00791
jose.sotero@upr.edu

ABSTRACT

This note presents different measurements of hardware and software performance in classical molecular dynamics (CMD) simulations from 2001 through 2010 obtained from published literature and the internet. Opinion articles by CMD researchers point out that tools developed during that decade to set-up CMD simulations barely increased human productivity. Massively parallel hardware and CMD software running on that hardware performance has increased a thousand-fold during the same period. The analysis supports the need for better software tools for set-up and analysis of these types of simulations.

INTRODUCTION

Currently, performing molecular dynamics simulations, and molecular modelling in general, is one of the main objectives for building the most powerful supercomputers along with codebreaking, weather studies, nuclear physics and modelling the early universe. The performance of this type of hardware has been documented by a group of HPC researchers who have maintained since 1993 a web site and publication (`top500.org`) that compiles the biannual lists of the top 500 computer systems in the world[11]. In those lists one can find mostly national research and defence laboratories at the top few dozens positions and mostly industries and research universities through the rest of the list.

On the software side, general purpose CMD simulators such as CHARMM [2], DL_POLY [10], NAMD [7] , AMBER [9] , GROMACS [1] and LAMMPS [8] have accumulated twenty years of development that have made them very reliable and very hard to improve in terms of efficient CPU and GPU usage. Today many of the results based on CMD simulations are obtained using general purpose simulators.

CMD review papers point out that "*One of the more time-consuming aspects of conducting a molecular simulation is generating an initial equilibrated system.*"[6] Assembling simulations is a difficult and long process. The user has to modify manually files with hundreds or thousands of lines in which the coordinates, type and force fields parameters of thousands of atoms in the system needs to be specified.

Version	Date	time	speedup	system
2.2	Sep 2000	67.2	1	PSC TCS1
2.3	Aug 2001	19.2	3.5	NCSA Titan
2.4	Mar 2002	8	8.4	PSC LeMieux
2.5	Oct 2003	2.5	26.88	NAMD 2.6
2.6	Mar 2008	1.5	44.8	Cambridge Xeon
2.7	Oct 2010	0.67	100.8	TACC Lonestar

Table 1. Times of the best NAMD's benchamrk performances and speedup with respecto to version 2.2.

This technical report quantifies performance gains in software and hardware used for CMD simulation based on data that can be found on the internet as well as data generated by our group. Based on this analysis the case is made for the need of more research focussed on the usability of the software and the rigorous application of proven software engineering techniques with the purpose of decreasing the costs and improving the human productivity of this endeavor.

SOFTWARE SPEEDUP

For the purposes of this analysis NAMD will serve as a representative of the development of general purpose CD simulators. Its developers have been publishing the result of benchmarks and they keep them available as new versions become available [5] since version 2.2. NAMD's performance on systems available at the time of the release of each version are measured benchmark called ApoA1. The benchmark system consists of a 92,224 atoms system (a solvated protein). The configuration includes the core integrator, a scaled 1-4 exclusion, neighbors lists with a 12Å cutoff and a 12Å switch distance, PME and a periodic full electrostatic computation.

Table 1 presents how NAMD's performance progressed during the last decade. The date of the user manual for that version was taken as the release date of the version. The actual release date of the software might have been earlier or later but the assumption used here is that they are relatively small and uniform throughout de development. The best running time for a 1 ns simulation was estimated from each graph presented in the performance report. Speedups are computed relatively to version 2.2's time.

HARDWARE SPEEDUP

The peak speed in Tera FLOPS of the top-rated and the 500th supercomputers from 2000 through 2010 was obtained from

Date	500th (speedup)	top500 (speedup)
Nov. 2000	0.08 (1.0)	12.3 (1.0)
Jun. 2001	0.10 (1.2))	12.3 (1.0)
Jun. 2002	0.21 (2.7)	41.0 (3.3)
Nov. 2003	0.56 (7.2)	41.0 (3.3)
Jun. 2004	1.08 (13.8)	41.0 (3.3)
Jun. 2005	2.00 (25.6)	183.5 (14.9)
Jun. 2006	5.60 (71.8)	367.0 (29.9)
Jun. 2007	4.60 (59.0)	367.0 (29.9)
Jun. 2008	17.90 (229.5)	1375.8 (112.0)
Jun. 2009	23.70 (303.8)	1456.7 (118.5)
Nov. 2010	57.50 (737.2)	4701.0 (382.6)

Table 2. Speeds and speedups w/r to year 200 in TFLOPS in the top500.org lists.

Date	NAMD's system (speedup)
Sep 2000	0.3 (1.0)
Aug 2001	1.0 (3.0)
Mar 2002	6.0 (17.7)
Oct 2003	15.3 (44.8)
Mar 2008	28.1 (82.3)
Oct 2010	55.5 (162.6)

Table 3. Speeds and speedups w/r to year 200 in TFLOPS of NAMD systems.

the WWW site `top500.org`. Table 2 summarizes the data gathered for those systems. The same data for computers reported in NAMD's benchmark reports (table 1) were obtained as well as shown in table 3. On the years were NAMD versions were released the top500 list for the month closest to the release was chosen. June was selected for other years. Some of the systems were NAMD's benchmark was tested have been upgraded a few times, for example *NAMD Titan*. It was assumed that NAMD's was benchmarked using the same systems that appeared on the chosen lists.

DISCUSSION

Figure 1 shows the comparison between NAMD's speedup and of the hardware were the ApoA1 benchmark were run. Both speedups are very similar suggesting that the core algorithms in the CMD simulator maintained essentially the same efficiency. Changes to those core components have may have been limited to fine-tuning to the new architectures. Significant developments in NAMD during this period have been made as expansion of it's capabilities. Those are not used by the ApoA1 test. Also, the ApoA1 test may have become too small for the newer systems and it may not be fully measuring the hardware and software capability improvements.

Figure 2 shows how the performance increase of supercomputers in general along with NAMD's test systems during the decade. Top systems have maintained a 100 factor advantage over the 500th place. The fact that NAMD's benchmarks do not show the same exponential performance increase trend as supercomputers in general hardware is attributable to the hardware were the tests were executed, not to NAMD itself. If these data were to be extrapolated, simulations performed with NAMD would had speedup increases similar to the hardware. We stress our conjecture that NAMD is representative of all other general purpose simulators in this respect.

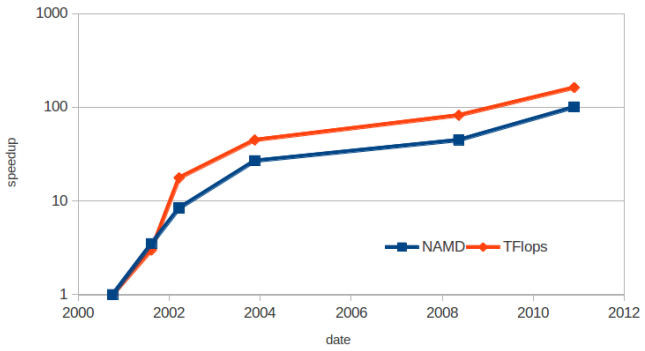

Figure 1. Speedups of NAMD and of NAMD's benchmarked systems.

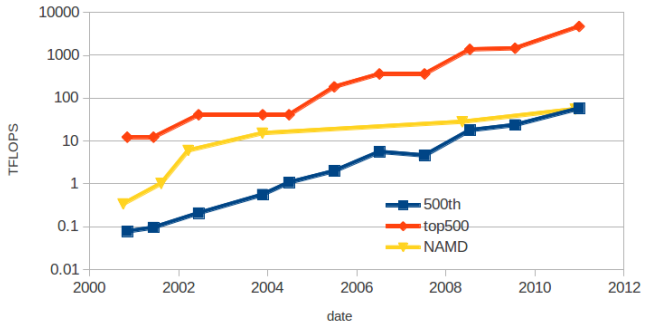

Figure 2. Speeds of the top and 500th fastest computers according to top500.org and systems used for NAMD's benchmarks.

Recent developments in GPGPU technology have made possible systems that claim to be "personal supercomputers". *Tabonuco* is the name of a system in our laboratory with a dual Intel(R) Xeon(R) CPU E5-2665 0 @ 2.40GHz and four NVIDIA Tesla C2075 GPGPUs running NAMD version 2.9 64 bits multicore CUDA. The ApoA1 benchark was run on this system. The result is shown along with the NAMD 2.8 benchmarks in figure 3. Cores in Tabonuco refer to GPU cores (448 per device) while others are CPU cores. The result puts Tabonuco close to the range of the supercomputers and clusters used for NAMD's benchmarks in 2010. Current NVIDIA Tesla GPU units are claimed to be more than twice as fast as the C2050 [3, 4].

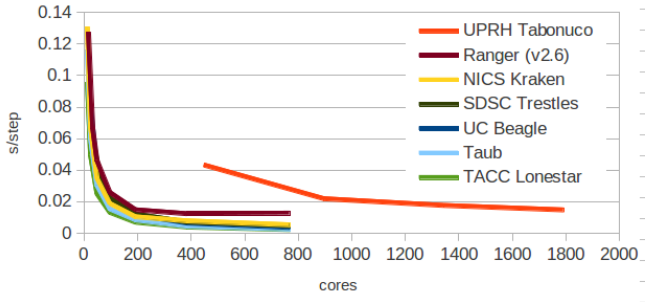

Figure 3. ApoA1 benchmark in a 4xC20050 Tesla system (Tabonuco) and NAMD 2.8 benchmarks. Tabonuco's cores refer to GPU cores.

CONCLUSIONS

During the last decade hardware performance as well as the execution of CMD simulations with general purpose CMD simulators in that hardware has increased a thousand-fold. The opinion of at least some CMD scientist is that the set-up of such simulations did not show nearly such an improvement. A scientists making a simulation of the size of cutting-edge simulations of that period, which may still be useful today, is likely to be spending more time on the set-up than waiting for the results to come out. The analysis supports the need for research on the usability of current CMD software and the development software tools for set-up and analysis of these types of simulations from the point of view of software engineering.

ACKNOWLEDGMENTS

This work has been supported by the *National Science Foundation* under Grant NSF-DMR-0934195, *PENN-UPRH Partnership for Research and Education in Materials* (PREM).

REFERENCES

1. Berendsen, H., van der Spoel, D., and van Drunen, R. Gromacs: A message-passing parallel molecular dynamics implementation. *Comput. Phys. Comm. 91*, 1-3 (1995), 43–56.
2. Brooks, B., Bruccoleri, R., Olafson, D., States, D., Swaminathan, S., and Karplus, M. Charmm: A program for macromolecular energy, minimization, and dynamics calculations. *J. Comput. Chem. 4* (1983), 187–217.
3. Corporation, N. *TESLA C2050 / C2070 GPU Computing Processor Supercomputing at 1/10th the Cost*, 2010. Accessed Sept. 17, 2013 at `http://www.nvidia.com/docs/IO/43395/NV_DS_Tesla_C2050_C2070_jul10_lores.pdf`.
4. Corporation, N. *Tesla Kepler GPU Accelerators*, 2012. Accessed Sept. 17, 2013 at `http://www.nvidia.com/content/tesla/pdf/Tesla-KSeries-Overview-LR.pdf`.
5. for Advanced Science, B. I., and Technology. Namd performance.
6. Maginn, E. J. From discovery to data: What must happen for molecular simulation to become a mainstream chemical engineering tool. *AIChE J. 55*, 6 (2009), 1304–1310.
7. Nelson, M. T., Humphrey, W. F., Gursoy, A., Dalke, A., Kale, L. V., Skeel, R. D., and Schulten, K. NAMD: A parallel, object-oriented molecular dynamics program. *Int. J. Supercomput. Ap. 10*, 4 (1996).
8. Plimpton, S. Fast parallel algorithms for short-range molecular dynamics. *J Comp Phys 117*, 1-19 (1995).
9. Salomon-Ferrer, R., Case, D., and R.C.Walker. An overview of the amber biomolecular simulation package. *WIREs Comput. Mol. Sci. 3* (2013), 198–210.
10. Smitha, W., Yonga, C., and Rodgerb, P. Dl_poly: Application to molecular simulation. *Mol. Simul. 28*, 5 (2002), 385–471.
11. Strohmaier, E., and Meuer, H. W. Supercomputing: What have we learned from the top500 project? *Computing and Visualization in Science 6*, 4 (2004), 227–230.