BigDataX 2015: Data-Intensive Computing Application Projects

Mike Wilde, University of Chicago

The PIs have identified various applications from many disciplines that demonstrate characteristics of big-data applications. [1, 20]. These applications cover a wide range of domains, from astronomy, physics, astrophysics, pharmaceuticals, bioinformatics, biometrics, neuroscience, medical imaging, chemistry, climate modeling, economics, and data analytics.

David Keyes identified in a recent keynote presentation reasons why today's computational scientists want performance: resolution, fidelity, dimension, artificial boundaries, ***parameter inversion, optimal control, uncertainty quantification, and the statistics of ensembles*** [15]. A decade ago or earlier, it was recognized that applications composed of large numbers of tasks may be used as an driver for numerical experiments that may be combined into an aggregate method [16]. In this proposal, we have decided to focus on several applications, such as Protein Structure, OOPS [18], NAMD, DOCK, and Montage. These are all applications that collaborators Wilde and Wozniak at UChicago are already working with, and would make ideal testbeds for undergraduate research projects.

Protein Structure Application: As protein targets increase in size, the need for next generation HPC computing becomes increasingly important to model. Current prediction methods have limited accuracy even for proteins on the order of 100 residues when homology-based information is minimal. To fold larger and multi-domain proteins, statistical sampling becomes a limiting factor. As a result, the folding and docking processes require extra sampling rounds, which will greatly increase the amount of computation power required. Furthermore, this entire class of calculations of protein structure (see Figure 2) is generally going to require a much larger number of loosely coupled calculations rather than a few massively parallel calculations. The identification of the protein sequences in a genome has transformed biological studies. But it is only the starting point. Amino acid sequence codes for structure, which often determines function. At the next level, protein association is integral to signaling networks which control and orchestrate cellular processes. Our long-term goal is the ability to take a set of genes identified in a biological process and provide the structure and function of the proteins, as well as identify their interactions and signaling connections, including the positive and negative feedback one interaction has on another. Petascale computation with adequate storage resources and bandwidth are critical for this far-reaching goal.

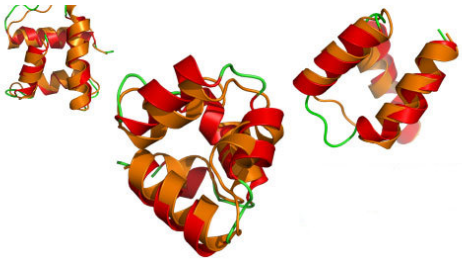

Figure 1: Development and validation methods to predict protein structure using homology-free approaches

OOPS – the Open Protein Simulator – is a suite of C++ programs for the prediction of the structure of proteins with minimal use of information derived from sequence similarity or homology to other proteins. OOPS derives its speed and accuracy from the use of a "C_β" model, an accurate statistical potential, and a search strategy involving iterative fixing of structure in multiple "rounds" of folding. Since OOPS uses minimal homology information and a reduced representation, its success depends crucially on describing the "protein physics" correctly. Great effort has been devoted to the energy function, e.g., interactions are conditional on backbone geometry and the relative orientation of side chains. In 2009, its accuracy exceeded available all-atom potentials, and it has been significantly improved since then. Our homology-free 2^o and 3^o structure predictions for small proteins rival or exceed homology-based methods with (expensive) explicit side chains, engendering optimism for continued progress. OOPS is currently executed in parallel on modest computing clusters using an ad-hoc Python script that submits independent parallel jobs, analyzes results, and orchestrates the iteration of a parallel solution to a folding problem. We propose to adapt OOPS to effectively utilize the hybrid architecture through Xeon Phi coprocessors or NVIDIA Tesla GPGPUs. We would therefore be enabling OOPS to be applied to challenging folding targets and providing the requisite number of docking simulations (10^4 – 10^6) of challenging binding targets. OOPS's performance, accuracy, capabilities and usability will be enhanced and will benefit from live scientific usage and testing on this high TFLOP computing platform.

NAMD/DOCK/BLAST: There are a wide variety of applications from the general field of biology, which are an excellent fit for MTC, due to the "ensemble" nature of many biochemical simulation studies. **NAMD** is a molecular dynamics program designed for high performance simulations of large biomolecular systems on parallel computers, which may be run as part of a statistical ensemble [21]. In the pharmaceutical domain, there are applications that screen **KEGG** [17] compounds and drugs against important metabolic protein targets using **DOCK6** [**Error! Reference source not found.**] to simulate the "docking" of small molecules, or ligands, to the "active sites" of large macromolecules of known structure called "receptors". The parameter space is large, totaling to more than one billion computations that have a large variance of execution times from seconds to hours, with an average of 10 minutes. The entire parameter space would require over 22,600 CPU years, or over 25 days on a petaflop supercomputer [23], but is most easily expressed as an MTC run. In bioinformatics, Basic Local Alignment Search Tool (**BLAST**) BLAST [22][20] is a family of tools for comparing primary biological sequence information. A BLAST search enables one to compare a query sequence with a library or database of sequences [19].

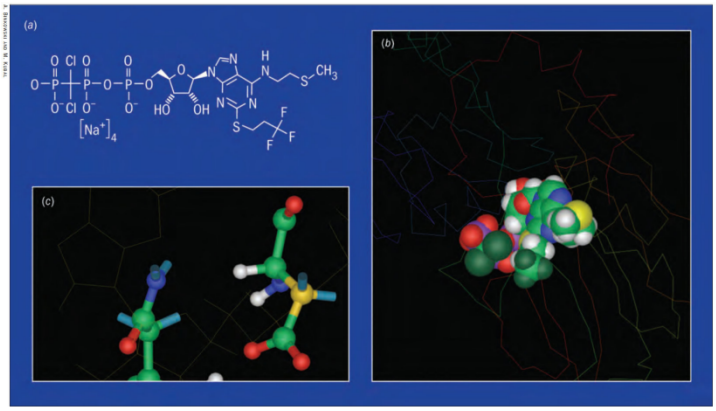

Figure 3: Molecular docking simulations for drug discovery through parallel scripting. (a) 2D representation of best scoring compound, KEGG drug compound D03361, against protein NAD kinase. (b) Spherical representation of D03361, Cangrelor tetrasodium, docked in pocket of NAD kinase (only showing backbone of protein as wireframe). (c) Close-up of NAD kinase (backbone removed) with a side-chain carbon atom of residue ASP209 in yellow binding to an oxygen of D03361 (gold wire frame), and residue ASN115 interacting with core rings of D03361.

Montage/AstroPortal: The **MONTAGE** [9, **Error! Reference source not found.**] (see Figure 4) application is a national virtual observatory project [10] that stitches tiles of images of the sky from various sky surveys (e.g. **SDSS** [8]) into a photorealistic single image; it is well-studied for MTC [22]. The "**AstroPortal**" [7] offers a stacking service of astronomy images from the Sloan Digital Sky Survey (SDSS) dataset (currently at 10TB with over 300 million objects) [8] using grid resources. In the physics domain, the **CMS** detector being built to run at **CERN's Large Hadron Collider** [11] is expected to generate over a petabyte of data per year. Supporting applications that can perform a wide range of analysis of the **LHC** data will require novel support for data intensive applications and many independent analysis tasks.

Figure 2: DSS pleiades mosaic generated by Montage; Inseok Song, University of Georgia - Digitized Sky Survey

There are many other applications we have identified to be a great fit for big data, and in many cases, we have already begun work on implementing these applications with Swift. Due to space limitation, we simply list them and their references here. In **Earth Systems**, we have **CIM-EARTH** [4, 5]. In **Physical Chemistry**, we have Monte Carlo search on the **Deem** dataset [6]. In **Neuroscience**, we have **CNARI** [14] and **fMRI/AIR/FMRIB/SPM2** [2]. In **Economics**, we have **MARS** [3]. In **Social Learning**, we have **computer simulations** [12] and **Tournaments** [13].

All of these applications are an excellent fit for undergraduate student summer projects. The UChicago mentors have in place a wide array of collaborators from the various science domains, which can provide students with interesting applications with real datasets towards achieving real science breakthroughs. These projects will not require any specific science domain knowledge from the students in order to be successful.

1 Mentor

Michael Wilde is a software architect in the Mathematics and Computer Science Division, Argonne National Laboratory, and a Senior Fellow of the University of Chicago/Argonne National Laboratory Computation Institute. His research focus is the application of parallel scripting to enhance scientific productivity by making parallel and distributed computing systems easier to use. He also conducts research into data provenance to record and query the history and metadata of scientific computations and datasets. His work centers on development and application of the Swift parallel scripting language, http://swift-lang.org.

2 References

[1] I. Raicu, I. Foster, Y. Zhao. "Many-Task Computing for Grids and Supercomputers", Invited Paper, IEEE Workshop on Many-Task Computing on Grids and Supercomputers (MTAGS08), 2008

[2] The Functional Magnetic Resonance Imaging Data Center, http://www.fmridc.org/, 2007

[3] D. Hanson. "Enhancing Technology Representations within the Stanford Energy Modeling Forum (EMF) Climate Economic Models", Energy and Economic Policy Models: A Reexamination of Fundamentals, 2006

[4] D. Bernholdt, S. Bharathi, D. Brown, K. Chanchio, M. Chen, A. Chervenak, L. Cinquini, B. Drach, I. Foster, P. Fox, J. Garcia, C. Kesselman, R. Markel, D. Middleton, V. Nefedova, L. Pouchard, A. Shoshani, A. Sim, G. Strand, and D. Williams, "The Earth System Grid: Supporting the Next Generation of Climate Modeling Research", Proceedings of the IEEE, 93 (3), p 485-495, 2005

[5] J. Elliott, I. Foster, K. Judd, E. Moyer, and T. Munson. CIM-EARTH: Philosophy, Models, and Case Studies, Feb 2010. Argonne National Laboratory, Mathematics and Computer Science Division preprint ANL/MCSP1710-1209

[6] M.W. Deem, R. Pophale, P.A. Cheeseman, and D.J. Earl. Computational discovery of new zeolite-like materials. The Journal of Physical Chemistry C, 113:21353–21360, 2009

[7] I. Raicu, I. Foster, A. Szalay, G. Turcu. "AstroPortal: A Science Gateway for Large-scale Astronomy Data Analysis", TeraGrid Conference (TG06), 2006

[8] SDSS: Sloan Digital Sky Survey, http://www.sdss.org/, 2008

[9] G.B. Berriman. "Montage: a Grid Enabled Engine for Delivering Custom Science-Grade Image Mosaics on Demand", SPIE Conference on Astronomical Telescopes and Instrumentation. 2004

[10] US National Virtual Observatory (NVO), http://www.us-vo.org/index.cfm, 2008

[[11] CERN's Large Hadron Collider, http://lhc.web.cern.ch/lhc, 2008

[[12] L. Rendell, L. Fogarty, and K. N. Laland. Rogers' paradox recast and resolved: Population structure and the evolution of social learning strategies. Evolution, 64(2):534–548, 2009

[[13] L. Rendell, R. Boyd, D. Cownden, M. Enquist, K. Eriksson, M. W. Feldman, L. Fogarty, S. Ghirlanda, T. Lillicrap, and K. N. Laland. Why copy others? Insights from the social learning strategies tournament. Science, 328(5975):208–213, April 9, 2010

[[14] Computational Neuroscience Applications Research Infrastructure,

http://www.ci.uchicago.edu/wiki/bin/view/CNARI/WebHome, 2008

[[15] D. Keyes. Exaflop/s, seriously!, 2010. Keynote lecture for Pan-American Advanced Studies Institutes Program (PASI), Boston University

[16] D. Abramson, J. Giddy, L. Kotler. High performance parametric modeling with Nimrod/G: Killer application for the global grid. In Proc. International Parallel and Distributed Processing Symposium, 2000

[17] KEGG's Ligand Database: http://www.genome.ad.jp/kegg/ligand.html, 2008

[18] PL protein library, http://protlib.uchicago.edu/, 2008

[19] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, D.J. Lipman. “Basic Local Alignment Search Tool”, J Mol Biol 215 (3): 403–410, 1990

[20] Justin M. Wozniak and Michael Wilde. Case studies in storage access by loosely coupled petascale applications, Proc. Petascale Data Storage Workshop at SC, 2009.

[21] Justin M. Wozniak, Michael Wilde, and Daniel S. Katz , JETS: Language and system support for many-parallel-task workflows J. Grid Computing 11(3), 2013.

[22] Zhao Zhang, Daniel Katz, Justin M. Wozniak Michael Wilde, Ian T. Foster, MTC Envelope: Defining the capability of large scale computers in the context of parallel scripting applications, Proc. HPDC 2013

[23] IBM BlueGene/P (BG/P), http://www.research.ibm.com/bluegene/, 2008