Final Report: Super Instruction Architecture for Scalable Parallel Computations

DE-FG02-09ER25913/DE-SC0002565

PI: Beverly Sanders

University of Florida

Date of Report: 12/2/2013

Period Covered 9/1/2009- 8/31/2014

Executive Summary

The most advanced methods for reliable and accurate computation of the electronic structure of molecular and nano systems are the coupled-cluster techniques. These high-accuracy methods help us to understand, for example, how biological enzymes operate and contribute to the design of new organic explosives.

The ACES III software provides a modern, high-performance implementation of these methods optimized for high performance parallel computer systems, ranging from small clusters typical in individual research groups, through larger clusters available in campus and regional computer centers, all the way to high-end petascale systems at national labs, including exploiting GPUs if available. ACES III calculates the simultaneous repulsions and attractions of all charged particles in these molecules to unprecedented accuracy.

The basis for the performance of ACES III resides in the underlying super instruction architecture (SIA), which is a framework for parallel programming. One part of the SIA is a domain specific parallel programming language call SIAL which was designed to support computations involving tensor algebra on extremely large multi-dimensional arrays. The other part is the parallel runtime system that manages much of the complexities involved in writing large scale parallel programs in SIAL.

The goal of this project was to enhance the SIA, both the underlying parallel computing framework and the scientific capabilities of ACES III, and use the system for scientific investigations. The project was collaboration between Computer Scientists and Computational Chemists.

Accomplishments

Software

ACES III

During the course of the project, the Super Instruction Architecture, ACES III, and supporting tools were enhanced in several ways and provided in several releases of the software.

- Enhancements to the SIAL language.
 - New language features for dealing with subblocks. This is important for handling high dimensional (>4) arrays. [SC10]
 - Support for execution on GPUs. A programmer can add directives to a SIAL program to indicate regions of code that should be executed on a GPU, and the required data movement. All of the primitive operations on blocks supported by SIAL can be executed on a GPU. [IJPP]
- The SipMap tool for performance modeling. [IPDPS]
 - This tool takes a SIAL program and automatically generates a performance model based on the SIAL source and data from a small reference run on the target architecture. This allows reasonable estimates of overall performance time and can also be used in various ways to tune SIAL programs for particular systems.
- New options for static work sharing that improve utilization on very large machines.
- Implementation of several new computational chemistry methods.

ACES 4.

During 2013, it became painfully clear that the software base of the SIP, the SIA's runtime system, had reached a limit in our ability to adapt it, and in particular, was preventing us from using the ideas of the SIA in domains other than computational chemistry. Using lessons learned in ACES III, we embarked on the implementation of ACES 4.

ACES 4 keeps the basic approach of the SIA with the DSL SIAL and support for block-oriented programming. It includes enhancements to the SIAL programming language, a new compiler for SIAL, and a new design and implementation of the SIP. An important goal was to ensure that the system is architected for sustainability and will be a usable test bed for new ideas.

Compared with ACES III, ACES IV offers the following enhancements:

SIAL compiler

- The new SIAL compiler is implemented using a modern compiler generator tool (LPG), which takes an annotated grammar and automatically generates a scanner, parser, and

the abstract syntax tree. The use of software development tools means that simple changes to the SIAL programming language can typically be implemented and tested in a couple of hours.

- More sophisticated static type checking that moves some checking from runtime to compile time.
- Improved byte-code format that reduces the size of SIAL object files by two thirds while including more symbolic information to enhance debugging and output. The new object files are portable and need not be generated on the same system that will read and execute them.

SIAL language enhancements

- More flexible and robust treatment of the way SIAL programs obtain their input data. In ACES 4, SIAL programs declare predefined variables and arrays, and the initial values of these variables are read from a file in a portable format that need not be generated on the same system as is executing the SIAL program. Compared with ACES III, this has the effect of decoupling a significant amount of sequential, legacy computational chemistry code (from ACES II, the sequential predecessor of ACES III) from the SIA.
- Improved and simplified annotations for GPU usage. Data movement between the host and a GPU is handled automatically by the SIP; the only necessary annotations are to delineate regions of code that should be executed on a GPU.
- Improved handling of special super instructions. Special super instructions are user provided computational kernels written in an appropriate programming language (C/C++, Fortran and/or CUDA)
- Numerous small enhancements that improve usability and maintenance.

SIP enhancements

- The ACES 4 SIP is implemented in C/C++ rather than a mixture of (mostly) Fortran and C. User provided special super instructions may be implemented in any language including C/C++, Fortran, and CUDA. While there are ample technical reasons to prefer C++ over Fortran for the SIP, it will also reduce the learning curve for Computer Science students who want to participate in the project.
- Improved management of distributed blocks.
- Internal data structures organized to facilitate multi-threading and efficient checkpointing.
- Implementation of basic tensor operations such as contraction can be performed in parallel using multiple cores (utilizing OpenMP) on a node, or GPUs if available.

- While ACES III was designed specifically for clusters and cannot be run without at least two MPI processes available, ACES 4 is structured to allow both efficient single node and cluster versions, each with and without GPU support.

Status

The compiler and single node versions of the SIP, with GPU support are already functional. Modifications to existing ACES III SIAL programs for ACES 4 can be done automatically although most will undergo some changes to exploit new features of SIAL. Existing user defined super instructions need to be modified to use a different interface to the SIP; this is a fairly straightforward task. We plan to host Aces 4 on GitHub in order to make it easier for others to contribute to the system.

Scientific Applications

In our work to begin to unravel some of the details of nucleic acid bases,

Notable applications of ACES III to large molecules are illustrated by the three papers by Szalay, et. al. on excited states of nucleic acid bases. After an initio study of the individual

bases, which provides predicted spectra compared to experiment . .

The nucleic acid bases were studied in extended environments and eventually, as stacked and parallel groupings. A double stack of the guanine-cytosine pair resulted in the fully ab initio treatment of 58 atoms, 216 electrons, and 1154 basis functions in a equation-of-motion (EOM)- coupled-cluster calculation for excited states. The figure also shows the calculations' scaling up to 60,000 processors.

Cytosine-Guanine Watson-Crick stacked dimer

1154 basis functions
58 atoms
216 electrons

Other work dealt with the accuracy of EOM-CCSD(T), a proposed non-iterative triples correction that offers vastly faster applications than a more complete EOM-CCSDT-3 or CC3 calculation. For some 24 molecules whose excited states have been carefully studied, EOM-CCSD(T) has error from those reference calculations of only 0.025 eV.

EOM-CCSD(T) vs. EOM-CCSDT-3

Reference results, Sauer, Schreiber, Silva-Junior, Thiel, JCTC (2009).

In a new effort, a new solvation model has been created within ACES III that unlike all other solvation models, is entirely quantum mechanical. The method exploits the weak interactions between the solvating units and between those units and the solute, in a way that allows all calculations to be done with no global scaling. This permits very large numbers of units to be included at almost no computational cost. The units to be described will include rare gas atoms, as in matrix isolation spectroscopy and in He-droplet spectroscopy in particular; other weakly interacting units like CO₂, N₂, or H-bonded units like water. The goal is to be able to study all properties of the solute like its spectroscopy under the influence of its medium.

ACES III has also been used to study the design of new organic explosives a task that requires high-accuracy methods. Our simulations calculate the simultaneous repulsions and attractions of all charged particles in these molecules to unprecedented accuracy. These systems have been the subject of numerous previous studies, but all were inconclusive due to the accuracy challenges of these systems.

Existing explosives, including RDX, are known empirically, but the specifics of how they work are not understood. We have succeeded in establishing the chemical play-by-play of the atoms to know how the explosives move from stable organic molecules through detonation. In calculating the various repulsions/attractions, we found the most likely energetic pathway. We now have insights as to how we can adjust the explosive yield as well as the shock-sensitivity to detonation.

Products and Technology Transfer

Publications

- B. A. Sanders, E. Deumens, V. Lotrich, M. Ponton. Refactoring a Language for Parallel Computational Chemistry, , Second Workshop on Refactoring Tools, Nashville, Tennessee, Oct. 19-23, 2008 (2008)
- V. Lotrich, N. Flocke, M. Ponton, B. A. Sanders, E. Deumens, R. J. Bartlett, A. Perera . An Infrastructure for Scalable Parallel Programs in Computational Chemistry, , ICS 09: 23rd International Conference on Supercomputing, ACM, 523-524 (2009), doi: <http://doi.acm.org/10.1145/152275.1542361>
- V. F. Lotrich, J. M. Ponton, A. S. Perera, E. Deumens, R. J. Bartlett, B. A. Sanders Super instruction architecture for petascale electronic structure software: the story, , Mol. Phys, special volume "Electrons, Molecules, Solids, and Biosystems: Fifty Years of the Quantum Theory Project", 108(21) 3323-3330 (2010)
- Beverly Sanders, Rod Bartlett, Erik Deumens, Victor Lotrich, and Mark Ponton, A block-oriented language and runtime system for tensor algebra with very large arrays, Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE Computer Society, Washington DC, ISBN 978-1-4244-7558-9 (2010).
- Erik Deumens, Victor F. Lotrich, Ajith Perera, Mark J. Ponton, Beverly A. Sanders, and Rodney, J. Bartlett . Software design of ACES III with the super instruction architecture, , Wiley Interdisciplinary Reviews – Computational Molecular Science, 1 895-901 (ISSN 1759-0884) DOI: 10.1002/wcms.77 (2011)
- Nakul Jindal, Victor Lotrich, Erik Deumens, Beverly Sanders. SIPMaP: A Tool for Modeling Irregular Parallel Computations in the Super Instruction Architecture, , 27th IEEE International Parallel and Distributed Processing Symposium, May 20-24, 1569670009 http://www.ipdps.org/ipdps2013/2013_advance_program.html (2013)
- Nakul Jindal, Victor Lotrich, Erik Deumens, Beverly Sanders, Exploiting GPUs with the Super Instruction Architecture, Int. J. Par. Progr., accepted for publication (2013)
- R. Molt, T. Watson, V.F. Lotrich, and R.J. Bartlett, "RDX geometries, excited states, and revised energy ordering of conformers via MP2 and CCSD(T) methodologies: Insights into decomposition mechanism," J. Phys. Chem. A 115, 884-890 (2011).
- P. Szalay, T. Watson, A. Perera, V. Lotrich, and R.J. Bartlett, "Benchmark studies on the building blocks of DNA. 1. Superiority of couple cluster methods in describing the excited states of nucleobases in the Franck-Condon region," J. Phys. Chem. A 116 (25), 6702-6710 (2012).

- P. Szalay, T. Watson, A. Perera, V. Lotrich, G. Fogarasi, and R.J. Bartlett, "Benchmark studies on the building blocks of DNA. 2. Effect of biological environment on the electronic excitation spectrum of nucleobases," J. Phys. Chem. A 116 (35), 8851-8860 (2012).
- P. Szalay, T. Watson, A. Perera, V. Lotrich, and R.J. Bartlett, "Benchmark studies on the building blocks of DNA. 3. Watson-Crick and stacked base pairs," J. Phys. A 117 (15), 3149-3157 (2013).
- T.J. Watson, Jr., V. Lotrich, P. Szalay, A. Perera, and R.J. Bartlett, "Benchmarking for perturbative triple-excitations in EE-EOM-CC methods," J. Phys. Chem. A 117, 2569-2579 (2013).
- R.W. Molt, Jr., R.J. Bartlett, T. Watson, Jr. and A. Bazanté, "Conformers of CL-20 explosive and ab initio refinement using perturbation theory: Implications to detonation mechanisms," J. Phys. Chem. A, 10.1021/jp305443h (2012).
- R.W. Molt, Jr., A. Bazanté, T. Watson, Jr., and R.J. Bartlett, "Pragmatic ab initio prediction of enthalpies of formation for large molecules: accuracy of MP2 geometries and frequencies using CCSD(T) correlation energies," J. Mol. Model. 10.1007/s00894-012-1663-1 (2012).
- Robert W Molt, Victor Lotrich, and Rodney J. Bartlett. The mechanism of rdx decomposition using ccscd(t). in preparation.
- Robert W Molt, Victor Lotrich, and Rodney J. Bartlett. Relative energies of cytochrome p450 intermediates using coupled cluster many-body methodologies. in preparation.

Web Site and Software

ACES III is open source software which can be obtained from <http://www.qtp.ufl.edu/ACES/>

Networks or collaborations fostered

- Sustainable Software Innovation Institute

The PI (Sanders) and co-PI (Deumens) are senior personnel in the Sustainable Software Innovation Institute for Computational Chemistry and Materials Modeling (S2I2C2M2), an interdisciplinary collaborative project funded by the National Science Foundation to attack the problems of software complexity and education in computational chemistry that present obstacles to long-term sustainability and extension of the software on ever-changing high-performance hardware.

- Blue Waters

As part of and NSF PRAC award, ACES III, was ported to Blue Waters with the help of the NCSA staff and used for the RDX calculations mentioned earlier. Blue Waters was also used to evaluate the GPU enhanced version of ACES III.