IJMDRR
E- ISSN –2395-1885
ISSN -2395-1877

Research Paper
Impact Factor - 2.262
Peer Reviewed Journal

A COMPUTER PACKAGE FOR MOLECULAR DYNAMICS COMPUTATIONS - AN EXAMPLE FOR THE USAGE OF COTS COMPONENTS

N. Gnanasankaran* S. Natarajan**
**Department of Computer Science, KristuJayanti College, Bangalore.*
***Department of Physics, Madurai Kamaraj University, Madurai.*

Abstract
Commercial Off-The-Shelf (COTS) technology is of recent origin and often used in developing SW for usage in different disciplines of science and technology. Almost without exception, every software-relatedendeavor will utilize a significant percentage of COTSsoftware components. In this contribution, as a case-study, a SW package (AMBER) used for computations related to Molecular Dynamics (MD) is considered for detailed study.

Keywords: COTS, Molecular Dynamics, AMBER.

1. INTRODUCTION

Commercial Off-The-Shelf (COTS) technology is widely used in many industries and also in scientific computing. In this paper, a case study is considered about the usage of COTS in a software package used in the field of molecular dynamics. Numerous software packages are available for MD calculations in physics, chemistry and other branches of science. Some of the popular packages are: AMBER, CHARMM, Gaussian, and GROMACS. AMBER is considered for the case study since it possesses reasonable computational capabilities.

Almost without exception, every software-related endeavor will utilize a significant percentage of COTS software components. The application of COTS components in crystallographic software was evaluated by us earlier. In another contribution, the case of the mathematical software viz., SCILAB was taken up as a case study. The SW GROMACS was scrutinized in one more paper [Gnanasankaran et al. 2013A-C]. In this paper, the case of the SW AMBER is taken up for detailed study.

2. ENERGY MINIMIZATION AND MOLECULAR DYNAMICS

In computational chemistry, energy minimization (also called energy optimization or geometry optimization) methods are used to compute the equilibrium configuration of molecules and solids. Stable state of molecular systems corresponds to global and local minimum on their potential energy surface. Starting from a non-equilibrium molecular geometry, energy minimization employs the mathematical procedure of optimization to move atoms so as to reduce the net forces (the gradients of potential energy) on the atoms until they become negligible.

Like MD and Monte-Carlo approaches, periodic boundary conditions have been allowed in energy minimization methods, to make small systems. A well-established algorithm of energy minimization can be an efficient tool for molecular structure optimization.

Unlike molecular dynamics simulations, which are based on Newtonian dynamic laws and allow calculating atomic trajectory with kinetic energy, molecular energy minimization does not include the effect of temperature, and hence the trajectories of atoms during the calculation do not really make any physical sense, i.e. we can only obtain a final state of the system that corresponds to a local minimum of potential energy. From a physical point of view, this final state of the system corresponds to the configuration of atoms when the temperature of the system is approximately zero, e.g. as shown in Fig. 1, if there is a cantilevered beam vibrating between positions 1 and 2 around an equilibrium position 0 with an initial kinetic motion, whether we start with the state 1, the state 2 or any other state between these two positions, the result of energy minimization for this system will always be the state 0.

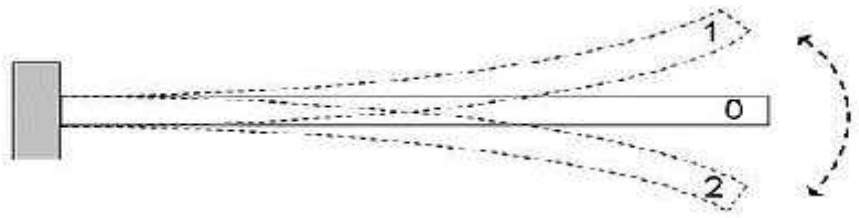

Fig. 1 Schematics of a cantilevered beam vibrating between two positions

The algorithms of gradient methods are the most popular methods for energy minimization. The basic idea of gradient methods is to move atoms by the total net forces acting on them. The force on atoms is calculated as the negative gradient of total potential energy of system, as follows:

$$F(r_i) = -\vec{\nabla}_{r_i} U^{tot}, \quad i = 1, \ldots, N \quad \text{..(1)}$$

where $\mathbf{r}_i$ is the position of atom i and U^{tot} is the total potential energy of the system.

An analytical formula of the gradient of potential energy is preferentially required by the gradient methods. If not, one needs to calculate numerically the derivatives of the energy function. In this case, the Powell's direction set method or the downhill simplex method can generally be more efficient than the gradient methods.

2.1 Simple gradient method

Here we have a single function of the potential energy to minimize with 3N independent variables, which are the 3 components of the coordinates of N atoms in our system. We calculate the net force on each atom (F) at each iteration step *t*, and we move the atoms in the direction of F with a multiple factor *k*. *k* can be smaller at the beginning of calculation if we begin with a very high potential energy. Note that similar strategy can be used in molecular dynamics for reducing the probability of divergence problems at the beginning of simulations.

$$r_i^t = r_i^{t-1} + \kappa \cdot F(r_i), \quad i = 1, \ldots, N \quad \text{..(2)}$$

We repeat this step in the above equation $t = 1, 2, \ldots$ until **F** reaches to zero for every atom. The potential energy of the system goes down in a long narrow valley of energy in this procedure.

Despite that it is as well called “steepest descent”, the simple gradient algorithm is in fact very time-consuming if we compare it to the nonlinear conjugate gradient approach, it is therefore known as a not very good algorithm. However, its advantage is its numerical stability, i.e., the potential energy can never increase if we take a reasonable *k*. Thus, it can be combined with a conjugated gradient algorithm for solving the numerical divergence problem when two atoms are too close to each other.

2.2 Nonlinear conjugate gradient method

The conjugate gradient algorithm includes two basic steps: adding an orthogonalvector to the direction of research, and then move them in another direction nearly perpendicular to this vector. These two steps are as well known as: step on the valley floor and then jump down. Fig. 2 shows a highly simplified comparison between the conjugated and the simple gradient on a 1D energy curve.

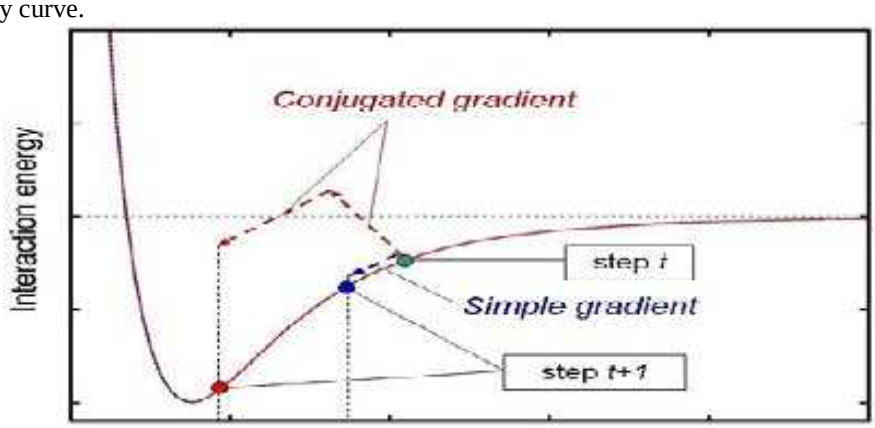

Fig. 2 Comparison between two gradientalgorithms on a simple 2D energy curve

In this algorithm, we minimize the energy function by moving the atoms as follows,

$$r_i^t = r_i^{t-1} + \kappa \cdot h_i^t, \quad i = 1, \ldots, N \quad \text{..(3)}$$

where

IJMDRR
E- ISSN –2395-1885
ISSN -2395-1877

$$h_i^t = F\left(r_i^t\right) + \gamma_i^{t-1} h_i^{t-1} \quad \text{..(4)}$$

and*gamma* is updated using the Fletcher-Reeves formula as:

$$\gamma_i^{t-1} = \frac{F\left(r_i^t\right) \cdot F\left(r_i^t\right)}{F\left(r_i^{t-1}\right) \cdot F\left(r_i^{t-1}\right)} \quad \text{..(5)}$$

Here we note that *gamma* can also be calculated by using the Polak-Ribiere formula, however, it is less efficient than the Fletcher-Reeves one for certain energy functions. At the beginning of the calculation (when $t = 1$), we can make the search direction vector '$h0 = 0$.

This algorithm is very efficient. However, it is not quite stable with certain potential functions, i.e. it sometimes can step so far into a very strong repulsive energy range (e.g. when two atom are too close to each other), where the gradient on this point is almost infinite. It can directly result a typical data-overrun error during the calculation. For resolving this problem, we can combine the conjugated gradient algorithm with the simple one. Fig. 3 shows the schematics of this combined predicting algorithm. We note for implementation that the steps 2 and 5 can be combined to one single step.

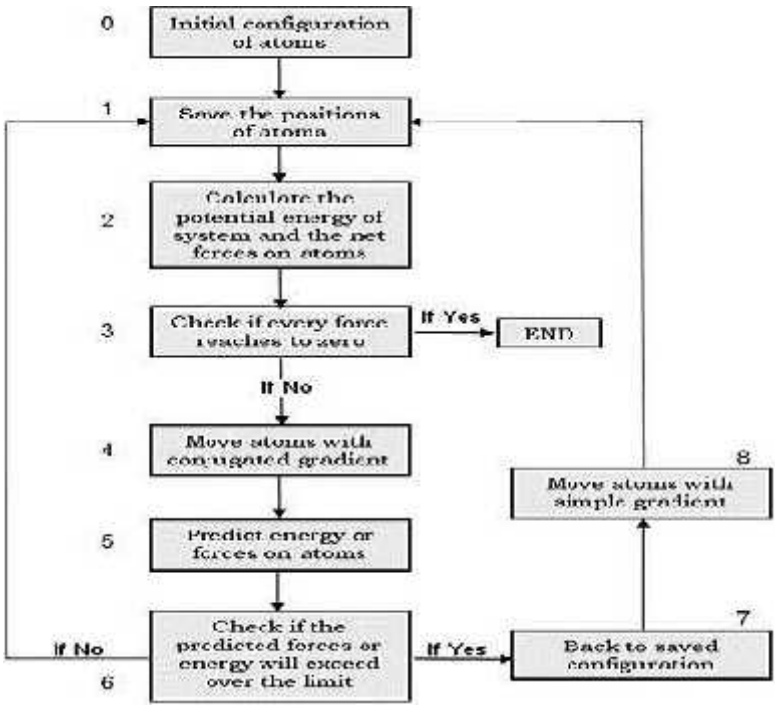

Fig. 3 Schematics of a computational energy minimization procedure

2.3 Boundary Conditions

The atoms in our system can have different degrees of freedom. For example, in the case of a tube suspended over two supports, we need to fix certain number of atoms N* at the tube ends during the calculation. In this case, it is enough not to move these N* atoms in the step 4 or 8 in Figure 3, but we still calculate their interaction with other atoms in the steps 2 and 5. i.e. from the mathematical point of view, we change the total number of variables in the energy function from 3N to 3N-3N* using the boundary condition, by which the values of these 3N* unknown variables are taken as known constants. Note that one can even fix atoms in only one or two directions in this way.

Moreover, one can equally add other boundary conditions to the minimized energy function, such as adding external forces or external electric fields to the system. In these cases, the terms in potential energy function will be changed but the number of variables remains constant.

IJMDRR
E- ISSN –2395-1885
ISSN -2395-1877

An example of the application of the energy minimization method in molecular modeling in Nano science is shown in Fig. 4.

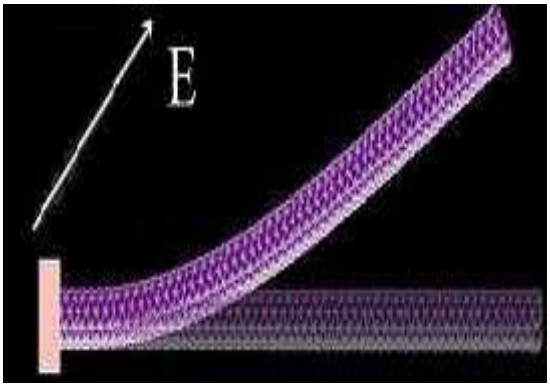

Fig. 4 Electrostatic deflections of a carbon nanotube in an electric field

3. SOFTWARES FOR MD DYNAMICS CALCULATIONS

Numerous software packages are available for MD calculations useful in physics, chemistry and other branches of science. Some of the popular packages are: AMBER (Cornell et al., 1995), Auto Dock (http://autodock.scripps.edu), CHARMM (Brooks et al., 1983), GAUSSIAN (Hehre et al., 1970; Young, 2001), GROMACS (www.gromacs.org), Materials Studio(www.msc.psu.edu), VASP (Kresse et al., 2007), etc. In the following sections, some of the features of the software AMBER are furnished.

3.1 AMBER (Assisted Model Building with Energy Refinement)

AMBER (an acronym for Assisted Model Building with Energy Refinement) is a family of force fields for molecular dynamics of biomolecules originally developed by the late Peter Kollman's group at the University of California, San Francisco (Cornell et al., 1995). AMBER is also the name for the molecular dynamics software package that simulates these force fields. It is maintained by an active collaboration between David Case at Rutgers University, Tom Cheatham at the University of Utah, Tom Darden at NIEHS, Ken Merz at Florida, Carlos Simmerling at Stony Brook University, Ray Luo at UC Irvine, and Junmei Wang at Encysive Pharmaceuticals.

3.2 Force field

The term "AMBER force field" generally refers to the functional form used by the family of AMBER force fields. This form includes a number of parameters; each member of the family of AMBER force fields provides values for these parameters and has its own name.

3.3 Functional form

The functional form of the AMBER force field (Cornell et al., 1995) is

$$V(r^N) = \sum_{\text{bonds}} \frac{1}{2} k_b (l - l_1)^2 + \sum_{\text{angles}} \frac{1}{2} k_\theta (\theta - \theta_0)^2$$

$$+ \sum_{\text{torsions}} \frac{1}{2} V_n [1 - \cos(n\omega - \gamma)] + \sum_{j=1}^{N-1} \sum_{i=j+1}^{N} \left\{ \epsilon_{ij} \left[\left(\frac{r_{0ij}}{r_{ij}} \right)^{12} - 2 \left(\frac{r_{0ij}}{r_{ij}} \right)^{6} \right] + \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \right\} \quad \text{..(6)}$$

Note that despite the term force field, this equation defines the potential energy of the system; the force is the derivative of this potential with respect to position.

The meanings of right hand side terms are:

First term (summing over bonds): represents the energy between covalently bonded atoms. This harmonic (ideal spring) force is a good approximation near the equilibrium bond length, but becomes increasingly poor as atoms separate.

Second term (summing over angles): represents the energy due to the geometry of electron orbitals involved in covalent bonding.

IJMDRR
E- ISSN –2395-1885
ISSN -2395-1877

Third term (summing over torsions): represents the energy for twisting a bond due to bond order (e.g. double bonds) and neighboring bonds or lone pairs of electrons. Note that a single bond may have more than one of these terms, such that the total torsional energy is expressed as a Fourier series.

Fourth term (double summation over *i* and *j*): represents the non-bonded energy between all atom pairs, which can be decomposed into van der Waals (first term of summation) and electrostatic (second term of summation) energies.

The form of the van der Waals energy is calculated using the equilibrium distance (r_{0ij}) and well depth (ε). The factor of 2 ensures that the equilibrium distance is r_{0ij}. The energy is sometimes reformulated in terms of σ, where $r_{0ij} = 2^{1/6}(\sigma)$, as used e.g. in the implementation of the softcore potentials.

The form of the electrostatic energy used here assumes that the charges due to the protons and electrons in an atom can be represented by a single point charge (or in the case of parameter sets that employ lone pairs, a small number of point charges).

4. SOFTWARE

The AMBER software suite provides a set of programs for applying the AMBER forcefields to simulations of bio-molecules. It is written in Fortran 90 and C with support for most major Unix-like systems and compilers. Development is conducted by a loose association of mostly academic labs. New versions are generally released in the spring of even numbered years; AMBER 10 was released in April 2008. The software is available under a site-license agreement, which includes full source, currently priced at US$400 for non-commercial and US$20,000 for commercial organizations.
Amber is distributed in two parts: AmberTools15 and Amber14. One can use AmberTools15 without Amber14, but not *vice versa*. See below for information on how to obtain Amber14.

4.1 Technical Details of AMBER

1. The programs here are mostly released under the GNU General Public License (GPL). A few components are included that are in the public domain or which have other, open-source, licenses.
2. AmberTools is distributed in source code format, and must be compiled in order to be used. one will need C, C++ and Fortran90 compilers.
3. The distribution contains a Reference Manual in pdf format. You can also download the PDF version of the manual, if you want to see if AmberTools might meet your needs.
4. If you already have AmberTools14, you do not need to download anything. Simply type "./update_amber --upgrade" in your AMBERHOME directory. (Follow the instructions: you will need to type this twice.) You then need to re-run the configure script and re-compile.
5. If you are a new user, you should Download AmberTools15. Follow the installation instructions in the Reference Manual.
6. The Amber14 package builds on AmberTools15 by adding the **pmemd** program, whichresembles the **sander** (molecular dynamics) code in AmberTools, but provides (much) better performance on multiple CPUs, and dramatic speed improvements on GPUs. In this release, more features from *sander* have been added to *pmemd* for both CPU and GPU platforms, including performance improvements, and support for extra points, multi-dimension replica exchange, a Monte Carlo barostat, ScaledMD, Jarzynski sampling, explicit solvent constant pH, GBSA, and hydrogen mass repartitioning. Support is also included for the latest Kepler, Titan and GTX7xx GPUs.

5. COMPUTATIONS INVOLVED

It is easy to understand that one has to go through several computations, step by step to arrive at the minimum energy structure of the molecule, using the data available. Suitable computer programs have been written and are available for the different steps of the calculations. The modern software package, AMBER, incorporates all the above programs in a unique way and helps the scientists in arriving at the minimum energy configuration. No one knows the details of the programs used in the above packages and they remain black boxes, but help the scientists.

6. CONCLUSIONS

It is easy to visualize the usage of Commercial Off-The-Shelf (COTS) technology in the above example. The modern packages are very versatile in their functioning and the quality is not compromised in utilizing the above software package. Hence, the above example could be considered as a positive usage of COTS components, in designing a software package for the usage of a group of scientists.

REFERENCES

1. Brooks B. R, Bruccoleri R. E, Olafson B. D, States D. J, Swaminathan S and Karplus M. CHARMM: A program for macromolecular energy, minimization and dynamics calculations. Journal of Computational Chemistry, 4 (1983) 187–217. doi: 10.1002/jcc.540040211.
2. Cornell W.D, Cieplak P, Bayly C.I, Gould I.R, Merz K.M. Jr, Ferguson D.M, Spellmeyer D.C, Fox T, Caldwell J.W, Kollman P.A, A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids and Organic Molecules, J. Am. Chem. Soc.117 (1995) 5179–5197. doi:10.1021/ja00124a002.
3. N.Gnanasankaran, S.Natarajan, K.Iyakutti, K.Alagarsamy, Acase study of the applications of COTS Components in a Molecular Dynamics software, Lecture notes on Software Engineering, Vol 1(2), Pages 141-143, May 2013A.
4. N. Gnanasankaran, S. Natarajan, K. Alagarsamy, and K. Iyakutti,"Application of COTS components in crystallography," *BritishJournal of Mathematics & Computer Science*, 2012.Vol 3(2), Pages 99 – 107, March 2013B
5. N. Gnanasankaran, S. Natarajan, K. Alagarsamy, and K. Iyakutti,"Commercial Off-The-Shelf (COTS) components in softwareengineering: The software package SCILAB," *International Journal ofComputer Technology & Applications*, Vol 4(1), Pages 68 – 71,2013C.
6. Hehre W. J, Lathan W. A, Ditchfield R, Newton M. D and Pople J. A, Gaussian 70 Quantum Chemistry Program Exchange, Program No. 237 (1970).
7. http://autodock.scripps.edu.
8. http://www.vasp.at/.
9. http://cms.mpi.univie.ac.at/vasp/.
10. http://www.gaussian.com/g_tech/g_ur/m_citation.htmKresse G, Marsman M and Furkmuller J, Vienna Ab-initio Simulation Package, University of Wien, Austria (2007).
11. www.gromacs.org.
12. www.msc.psu.edu.
13. Young D, Computational Chemistry, Wiley-Interscience, Appendix A. A.2.4 (2001) 336.