September 2014, Volume 2
Proceeding of 3rd International Conference on Computation for Science and Technology, ISSN xxxx-yyyy

Acceleration of norm-conserving Pseudopotential Plane-Wave-Based DFT Calculation on GPU using CUDA

Feradi Fathurahman[1], Enggar Alfianto[1*], Hermawan K. Dipojono[2], Muhamad. A. Martoprawiro[1]

[1]*Department of Computational Science, FMIPA, Bandung Institute of Technology, Jalan Ganesha 10, Bandung 40132, Indonesia*
[2]*Department of Engineering Physics, Bandung Institute of Technology, Jalan Ganesha 10, Bandung 40132, Indonesia.*

Received: 18 September 2014 / Accepted: 30 November 2014

Abstract:
In present study, acceleration of density functional theory calculation using norm-conserving pseudopotential and plane wave (NCPP-PW) basis set has been performed. It did not use or parallelize commonly program packages (such as ABINIT, VASP, PWSCF, etc.) but propose prototypical program to carry out self-consistent field calculations to solve Kohn-Sham equation and focus on Hamiltonian diagonalization part by using CUDA to utilize a graphical processing unit (GPU) accelerator. The results showed that acceleration up to 10 times speed-ups for certain type of GPU, namely NVidia GTX 460, for three systems: 8 silicon atoms in cubic unit cell (small), 10 water molecules in a box (medium), and 64 silicon atoms in 2 × 2 × 2 cubic supercell (large).

Key words: *CUDA, Density Functional Theory, GPU, hamiltonian diagonalization, norm-conserving pseudopotential, plane wave basis set*

Introduction

Calculation of many-electron systems using norm-conserving pseudopotential plane wave based on DFT calculation possesses a deficiency. This deficiency causes long computational time in processing of Hamiltonian diagonalization. Long computational time is caused by iteration process that uses just a single processor. Due to this reason, a queue system to calculate the process is needed.

To accelerate computational time, parallelization method is performed. This method allows calculation using many processors simultaneously. The need of computers with more than one processor to do parallelization process is inevitable. However, multi-processor computers are relatively expensive, and moreover, its technological advances are rather slow.

Market demand on game industry, lead to the development of graphical cards with multiple core. The graphical cards with multiple processors known as GPU have shown significant development in its technology and its price are relatively lower than the traditional CPU with multicore processors.

Knowing the advantages has motivated us to do parallel calculation in GPU. Parallelization using GPU needs a special language that is CUDA. By using CUDA language, the normal calculation in processor can be transferred to GPU.

Experimental

The basic structure of calculation in this work is taken from FHI98MD [1] code. This code is designed to investigate the material properties of large system. It is based on an iterative approach to obtain electronic ground state. Iterative approach using Norm-Conserving pseudopotential has a general form as presented by Klinman and Bylander [2] to describe the potential of nuclei and core electrons acting on the valence electron. The exchange and correlation are described by GGA or LDA. The equations of motion (EoM) of the nuclei are integrated using molecular dynamic approximation [1].

The algorithm needs some modifications for parallel programming application. The modified code from now on is called PSPW_DFT. The first modification is performed on the Hamiltonian diagonalization which is done by using block-Davidson and Locally Optimal Block Preconditioned conjugate gradient (LOBPCG). The selection of this method is based on the fact that the Hamiltonian matrices can be generated by iterative processes. To guarantee speed-up, a precondition model is employed. The preconditioned model was initially introduced by Kerker [6] then implemented in KSSOLV [7].

***Corresponding author:** Enggar Alfianto,
E-mail: enggar@s.itb.ac.id

The algorithm of this code written by:

Algorithm the FHI98MD

- Read the input file.
- Calculate several variable that depend on the variables were specified by user.
- Read the pseudopotential file.
- Generate G- and G+k- vectors and also variables related to them.
- Calculate structure factor.
- Calculate form factor of local pseudopotentials.
- Calculate form factor of non-local pseudopotentials.
- Guess initial charge density ρ^0
- Calculate ion-ion interaction energy.
- SCF loop
  - Diagonalize Hamiltonian for all K-points using LOBPCG method. So that we can obtain N_{pairs} lowest eigenpairs.
  - From these eigenpairs evaluate new wave function coefficients and electron density ρ^i, where i designates current iteration number.
  - Evaluate energy components from this new electron density.
  - Check convergence using total energy difference.
  - Evaluate next input using charge density mixing

The calculation result from the block-Davidson method and LOBPCG is a Hamiltonian form implemented into wave function $\phi_{(i,k)}$ [4,6]. Action of Hamiltonian $\hat{H}$ to wave function $\phi_{(i,k)}$ should be written as equation 1.

$$\langle G+k|\hat{H}|\phi_{i,k}\rangle = \langle G+k|\hat{T_e}+\hat{V}^{loc}+V_{PS}^{nl}|\phi_{i,k}\rangle \quad (1)$$

With kinetic contribution reads equation 2.

$$\langle G+k|\hat{T_e}|\phi_{i,k}\rangle = \frac{1}{2}|G+k|^2_{c_{i,G+K}} \quad (2)$$

Local potential contribution reads equation 3.

$$\langle G+k|\hat{V}^{loc}|\phi_{i,k}\rangle = \frac{1}{\Omega}\int_{\Omega} V^{loc}(r)u_{i,k}(r)\exp[-iG\cdot r]. \quad (3)$$

And local potential contribution reads equation 4.

$$\langle G+k|\hat{V}^{nl}|\phi_{i,k}\rangle = \sum_{I_s,I_a}\sum_{l,m} w_{I_s}^{I} f_{i,I_s,I_a}^{l,m}(k)\exp\left[iG\cdot\tau_{I_s,I_a}\right]\Phi_{PS,I_s}^{l,m}(G), \quad (4)$$

A convergence criterion of 10-6 Ha is set for a test in order to know the number of iterations needed. The next step is parallelization process which is decomposed into three steps. First, the parallelization in K point, followed by parallelization on electronic state and parallelization on spatial plane wave decomposition from FFT grid. The third part of parallelization is included on diagonalization Hamiltonian step [8].

Measurement of computational time is conducted by using standard C++ library called clock(). As a measure of performance a speed up parameter is used, which is defined as the ratio between the time required to execute by CPU and the one by CPU+GPU.

Implementation of the Hamiltonian on the wave function using CUDA is performed on the vector G and FFT grid. The next step is the implementation of parallelization to block-Davidson method/LOBPCG [4]. On this process the BLAS package is usually being used, but in CUDA environment CUBLAS is used instead.

Results and Discussion

The first testing process is conducted to compare the result generated by the convergent iteration of PSPW_DFT with a widespread program the Abinit [10]. Systems that were used for doing some test are: H_2O, CO_2, N_2, CH_4, SiH_4, C_8H_{18}.

Figure 1 shows that the convergence calculation of PSPW_DFT is in good agreement with that from ABINIT calculation. PSPW_DFT is then modified through implementation of parallel computation with CUDA language to parallelize it using GPU. Figure 2 shows good results for the speed up process / time acceleration. The measurements of computational time are performed for 3 different jobs: diagonalization, SCF, and others that include but not limited to reading the input and output. When GPU is used, acceleration of calculations gains an increase by 3.59% for a small system, 6.44% for medium and 10.05% for a system of large systems.

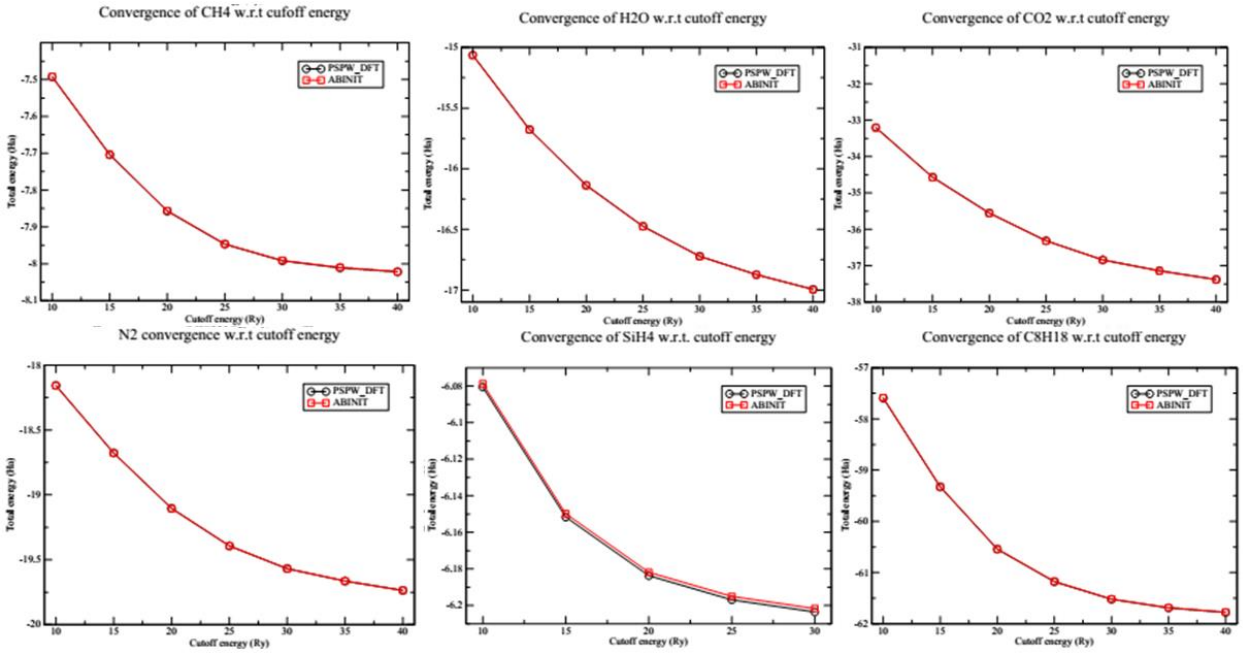

Figure 1. Comparison of convergence calculation between PSPW_DFT and Abinit using many systems.

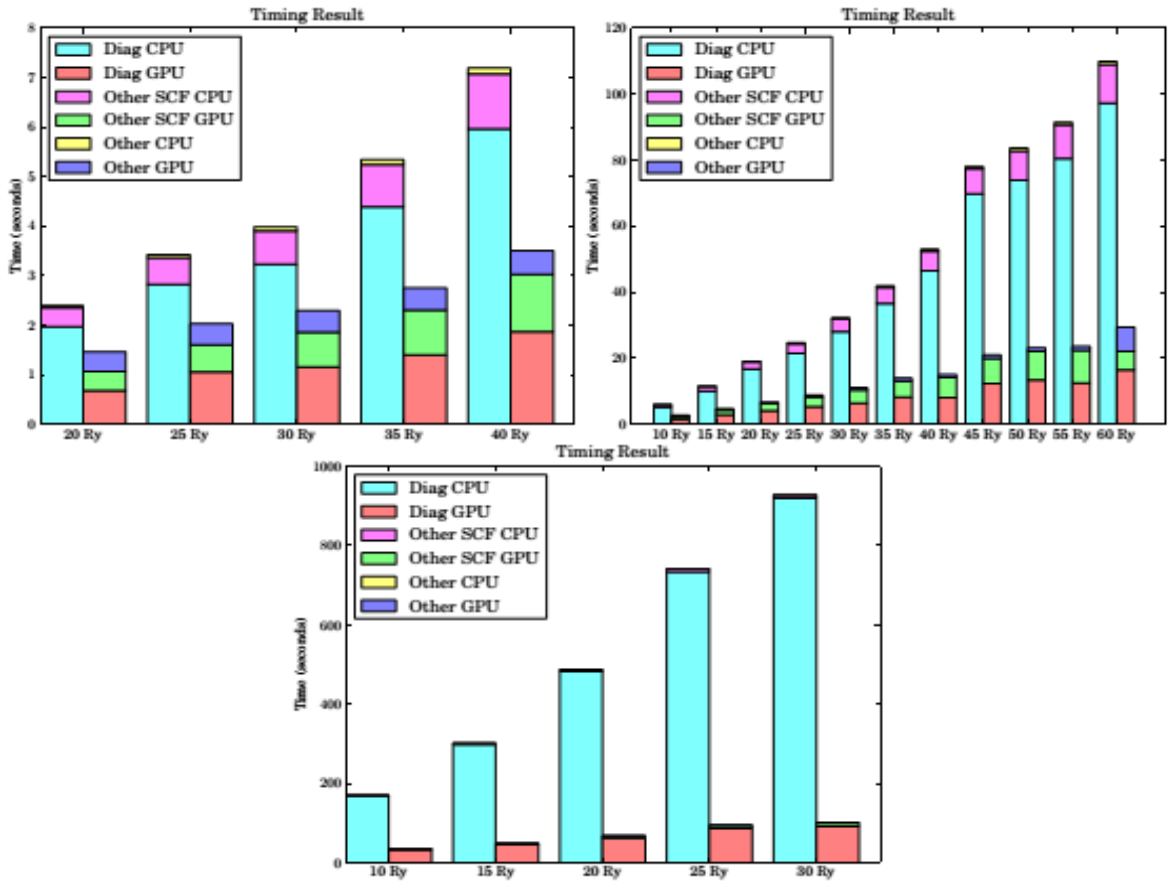

Figure 2. Comparison of computational time between GPU + CPU and CPU only for system Si8, 10 H2O and Si64.

Conclusions

PSPW_DFT plane wave pseudopotensial based code has been implemented and performed in GPU. Significant speed-up in computational time is achieved. The speed up is proportional with the size of the calculated systems.

Acknowledgement

The authors thank to the Directorate General of Higher Education, Ministry of Education and Culture of the Republic of Indonesia (DIKTI) for the financial assistance. Comments and suggestions from Dr. Kemal Agusta is gratefully acknowledge.

References

[1] M. Bockstedte, A. Kley, J. Neugenbauer, and M. Scheffler, Density functional theory calculation for poly-atomic systems: electronic structure, static and elastic properties and *ab-initio* molecular dynamics, *Comp. Phys. Comm*., **107**,1997, 187-222.

[2] L. Kleinman and D.M. Bylander, Efficacious form for model pseudopotentials, *Phys. Rev. Lett*., **48**, 1982, 1425-1428.

[3] R.E. Davidson, The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices, *J. Comput. Phys*., **17**, 1975, 87-94.

[4] G.P. Kerker, Efficient iteration scheme for self-consistent pseudopotential calculations, *Phys. Rev. B*, **23**, 1981, 3082-3084.

[5] C. Yang, J. C. Meza, B. Lee, and L. Wang, KSSOLV – a MATLAB toolbox for solving the Kohn-Sham equations, *ACM Trans. Math. Softw*., **36** (10), 2009, 1-10:35.

[6] F. Bottin, S. Leroux, A. Knyazev, dan G. Zerah, Large-scale ab initio calculations based on three levels of parallelization, *Comp. Mat. Sci*., **42**(2), 2008, 329-336.

[7] X. Gonze et. al, ABINIT: First-principles approach to material and nano system properties, *Comp. Phys. Comm*., **180**(12), 2009, 2582-2615.