

## Background

1. Biologically important macromolecules, such as proteins, are within the scope of electronic-structure calculation by using FMO because of the improvement of computational resources.
2. GPUs are successfully applied for many kinds of problems in high performance computation. Advanced Vector Extensions (AVX) of Intel SandyBridge CPU would be promising alternative.

## Purpose of this Research

Based on Yasuda [2008], We develop a software library (XA-CUDA-QM, XA-AVX-QM) to accelerate famous QM softwares (GAMESS-US etc...) by using NVIDIA GPU and Intel SandyBridge. We show the excellent performance of XA-CUDA-QM/XA-AVX-QM together with FMO.

## Coding Policy

1. Total energy and the gradient of under FMO approximation.
2. Most time-consuming steps (the evaluation of Coulomb, Hartree-Fock exchange, and exchange-correlation matrices...) are executed on GPUs. Remaining parts are calculated on CPUs.
3. Use single precision arithmetic as much as possible for the sake of effective speedup, without degradation of accuracy.
4. Take special care for task scheduling and host-GPU communication time to keep as many CPUs and GPUs busy.
5. Modular design applicable to many quantum chemistry softwares.

## Algorithms

### Electrostatic potential (Coulomb or J-matrix) :

Far-field contribution: fast multipole method (FMM) by CPU.

Near-field: two-electron integrals and the J-matrix in terms of primitive Hermite Gaussian basis are evaluated by GPU (Direct J Engine).

$$J_{ab} = \sum_{cd} (ab|cd) D_{cd} = \sum_p E_{ab}^p \sum_q (p|q) D_q$$

Schwarz upper bounds are used to reduce # of ERIs to be calculated, but we don't use integral symmetry ( e.g.  $(p|q)=(q|p)$  ). Gauss-Rys quadrature is used to evaluate ERIs because of small memory footprint. The communication time between CPU and GPU to send / receive was only about 10 % of GPU computation time.

$$p, q, Dq, Jp$$

The upper bound of each ERI is first evaluated and ERIs which are small in magnitude are calculated on GPU with single-precision. The rests (10% of total count of ERIs) are calculated on CPU with double precision.

### Hartree-Fock exchange (K-matrix) :

Uncontracted McMurchie-Davidson algorithm and full integral symmetry are used.

### Energy Gradient :

The same methods as stated above. Note ERIs for higher angular momentum are required.

### Environmental Electrostatic Potential (ESP):

J-matrix algorithm without FMM. This is the mainly time-consuming step of FMO.

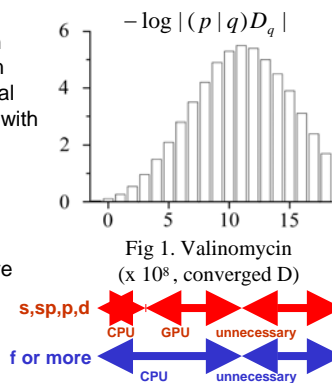


Fig 1. Valinomycin (x 10<sup>8</sup>, converged D)

## PRISM host code was re-written to use AVX as much as possible.

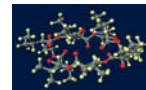
Four shell quartets are packed in an AVX vector and the 1- and 2-electron transformations in PRISM are fully vectorized. Some parts (evaluation of Boys function and summation of ERIs to Fock) are partially vectorized.

## Results & Discussion

**Benchmark Environment :** Intel Core i5 2500 @ 3.30GHz(SandyBridge 4core) + GTX580 x 1 + DDR3 6GB, Intel Composer XE (12.0) + MKL 10.3

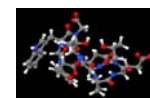
**Table 1 : Ab-initio calculation time of Valinomycin (GAMESS 2010 R3, RHF/3-21G, mainly Direct SCF)**

Software	Time(sec)	Energy[a.u.]
Original	476.24	-3750.9205018139
Original + XA-AVX-QM	149.69	-3750.9205017267



**Table 2-1 : FMO2 monomer energies of each residue of Chignolin (GAMESS 2010 R3 FMO-RHF/6-31G, a.u.)**

Residue	E(CPU)	Δ E(AVX+CUDA)
1(GLY)	-80.222749536	1.0E-09
2(TYR)	-534.769290097	5.0E-09
3(ASP)	-417.441523331	0.0E-09
4(PRO)	-307.413920526	5.0E-09
5(GLU)	-456.476084655	-1.0E-08
6(THR)	-344.370404179	1.4E-08
7(GLY)	-191.594835091	-1.2E-08
8(THR)	-344.375192996	1.0E-08
9(TRP)	-590.759583790	3.0E-09
10(GLY)	-393.749078255	1.0E-08



1. The value of dipole moment is exactly the same.
2. Total Monomer Energy  
E(Original) : -3661.172662455  
E(AVX+CUDA) : -3661.172662482  
Δ E : 2.7 × 10<sup>-8</sup> a.u.
3. Calculation Time (sec) :  
Original : 913.01  
AVX+CUDA : 472.45

**Table 2-2 : Calculation time of FMO2 environmental electrostatic potential of Chignolin (sec)**

	Time(Original)	Time(AVX+CUDA)
ESP(CUDA : XA-CUDA-QM)	519.856	107.824
Direct SCF(AVX : XA-AVX-QM)	307.177	266.144

We achieved 2x acceleration for total time by AVX and CUDA than by a CPU, and 5x acceleration for ESP alone. We expect much more acceleration for larger proteins because the calculation of ESP consumes predominant time (over 80%).

The regular SCF calculations of small but many fragments explain the rest of computational time. The direct SCF method is unfavorable for them because fragments are so small that we can store all ERIs in a host memory. On the other hand, GPGPU must use direct SCF because of the host-GPU data transfer cost. Hence, we developed fine-tuned ERI evaluation routine for host CPUs. Direct SCF acceleration has also achieved by this AVX tuning.

## Summary

A software library (XA-CUDA-QM, XA-AVX-QM) developed in this study made Hartree-Fock quantum chemical calculation with GAMESS FMO about 2 times faster than the original one. The total energy and the dipole moments were essentially the same. GPGPU calculation of environmental electrostatic potential becomes 5 times faster than multi-core CPUs calculation. AVX were found to be useful to accelerate the host-side ERI evaluation.

## Future Theme

1. CUDA acceleration of ERI including f or higher basis.
2. Fine tuning of Hartree-Fock Exchange of GPU-side, integral code of host-side.
3. Add-on of DFT algorithm to GAMESS-US by AVX and CUDA
4. Parallel CPU + GPGPU hybrid computation of FMO method