Modeling nano-scale carbon materials. Electronic structure on GPU

Jacek Jakowski[1], Stephan Irle[2], Keiji Morokuma[3]

[1]National Institute for Computational Sciences, University of Tennessee, Oak Ridge, TN, USA, [2]Nagoya University, Nagoya 464-8602, Japan, [3]Fukui Institute for Fundamental Chemistry, Kyoto University, Kyoto 6006-8103, Japan

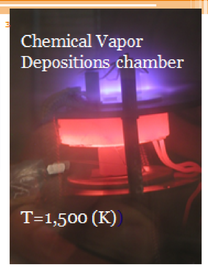
Abstract

In direct molecular dynamics methods the nuclei are moving classically, using Newton equation of motion, on the quantum mechanical potential energy surface obtained from electronic structure calculations. This potential energy surface has to be recalculated every time the nuclei change their position. In order to obtain a meaningful trajectory one has to perform several thousands of electronic structure calculations. We demonstrate in this work (a) how the electronic structure can be propagated without diagonalization in our Liouville-von Neumann molecular dynamics scheme, (b) how to cast our problem to GPU in a code that combines Fortran, CUBLAS library and CUDA kernels.

Motivation

Synthesis:
- Chemical vapor deposition
- Arc discharge
- Laser ablation

Nanotechnology, 21, 115301 (2010)



- Condensed phase, harsh environments
- Needs for computational models:
  - Coupling between electrons and nuclei (non- adiabaticity)
  - Include electronically excited states during dynamics
  - Include interaction with laser pulses of different shapes
  - Charge transport, nanoelectronics
  - Applicable for large systems
- Framework:
  - Time-dependent Schroedinger Equation

Computational method

❑ Electronic structure

We employ the density functional tight binding framework to describe electronic structure:

$$E = \sum_i 2 f_i \langle \phi_i | H_{core} | \phi_i \rangle + \frac{1}{2} \sum_{A,B}^{Atom} \gamma^{AB} \Delta q^A \Delta q^B + \sum_{A>B}^{Atom} E_{rep}^{AB}$$

where f_i is an occupation number (typically 0 or 1). Last expression can be rewritten in the form

$$E = Tr[H_{core} P] + \frac{1}{2} Tr[G(P) P] + \sum_{A>B}^{Atom} E_{rep}^{AB}$$

In which P is a reduced one-electron density matrix obtained from molecular orbitals coefficients C, such that $P = C f C^T$.

❑ Direct Dynamics: Quantum dynamics

We perform Liouville-von Neumann molecular dynamic[JCP 130, 224106,(2009)]

$$i\hbar \frac{dP(t)}{dt} = [F(t), P(t)]$$

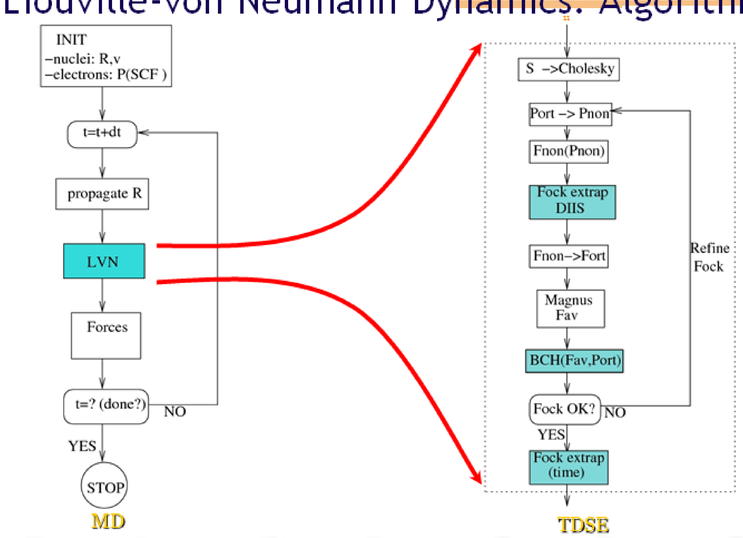
where F(t) is Fock matrix and P is a density matrix. The time evolution of density matrix can be obtained from

$$P(t_1) = P(t_0) - it[\bar{F}, P] - \frac{\Delta t^2}{2!}[\bar{F},[\bar{F},P]] + i\frac{\Delta t^3}{3!}\left[\bar{F},[\bar{F},[\bar{F},P]]\right] + \frac{\Delta t^4}{4!}\left[\bar{F},[\bar{F},[\bar{F},[\bar{F},P]]]\right] \ldots$$

where $\bar{F} = \frac{1}{2}[F(t_0) + F(t_1)]$ and $t_1 = t_0 + \Delta t$.

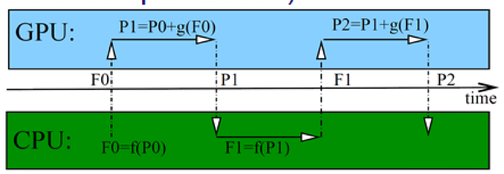
The propagation of density matrix is performed in Cholesky orthogonalized basis set.

Liouville-von Neumann Dynamics: Algorithm



Hybrid CPU+GPU implementation:

Objectives:
- Get the calculations done faster
- connect GPU code with main program Fortran code (Gaussian, legacy electronic structure codes)
- Use existing, efficient libraries whenever possible (let others do optimization)

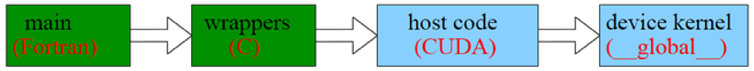


GPU ←→ CPU memory transfer:
- Density (P): one transfer GPU → CPU
- Fock (F): one transfer CPU → GPU

GPU implementation

- CUBLAS for matrix multiplication (very efficient)
- CUDA kernel for transpose
- C code that calls kernels
- Wrappers in C to bind with the top layer Fortran code (build as a static library)

```
nvcc  transpose.cu      (nvcc also uses pgCC)
pgCC –c  jacek.C  -o libjacek.o
pgf90  program.F  libjacek.o  –lcudart –lcublas
```



Some lessons learned

❑ Rethink algorithm
- often can be redesign in terms of matrix multiplication very well optimized
- problem size, scaling

❑ Padding important
data size may affect performance

❑ Data pattern - exploit if possible

❑ Use libraries (use existing and create your own to save future work)

❑ CPU vs TESLA:
- CPU code:
  2xIntel Quad Xeon 2.5GHz + multithreaded MKL
- CPU+GPU:
  C1060 (30 DP & 240SP cores) 1.3GHz+ CUBLAS & CUDA

❑ Speedup: CPU vs GPU is ~ 2 for DP , ~10 for SP

References

[1] J. Jakowski, K. Morokuma, "Liouville-von Neumann molecular dynamics", J. Chem. Phys. 130, 224106 (2009)

[2] J. Jakowski, S. Irle, K. Morokuma, "Quantum Chemistry: Propagation of electronic structure on GPU" in "GPU computing Gems" ed. Wen-Mei-Hwu, ch. 5, pp.59-74, Morgan Kauffmann , 2011

Acknowledgments

Financial support is provided in parts (JJ) by NSF SC/TN–EPSCOR grant #EPS-0919436, Computational resources within TG-DMR10047 on kraken at NICS.