

実空間第一原理電子状態計算コード RSDFT講習会

東京大学大学院工学系研究科物理工学専攻
岩田潤一

京都産業大学理学部物理科学科
内田和之

Mar 8, 2016 @柏

内容

- 第一原理計算とは？
- 密度汎関数法(Density Functional Theory)
- 実空間差分擬ポテンシャル法(RSDFT)
- 実習

第一原理電子状態計算とは？

第一原理(電子状態)計算：何を計算するのか？

- 多電子系シュレーディンガ一方程式の最小固有値「基底状態のエネルギー」を計算する
- 正攻法では計算できないので密度汎関数法(DFT)を経由する
- 実験に合わせるためのパラメータが無い



基底状態エネルギー以外にDFTで計算できること

- 安定な原子配置
(基底状態エネルギーを 原子座標の関数 $E_0(\mathbf{R}_{\text{ion}})$ として最小化する)
- 電子密度 (化学結合の様子)
- バンド構造 (1電子のエネルギー,運動量)
- 時間依存密度汎関数法、摂動論
(励起状態、ダイナミクス、応答関数、etc.)

DFTの理論的側面

密度汎関数理論まとめ①

電子密度(N電子系)

$$\rho(\mathbf{r}) = \sum_{i=1}^N |\phi_i(\mathbf{r})|^2$$

P. Hohenberg and W. Kohn, Phys. Rev. 136 (1964) B864.
W. Kohn and L. J. Sham, Phys. Rev. 140 (1965) A1133.

(1998年 W. Kohn ノーベル化学賞)

系の全エネルギー(密度の汎関数)

$$E[\{\phi_i\}] = \sum_{i=1}^N \int d\mathbf{r} \phi_i^*(\mathbf{r}) \left(-\frac{1}{2} \nabla^2 \phi_i(\mathbf{r}) \right) \quad \text{運動エネルギー}$$

$$+ \int d\mathbf{r} \rho(\mathbf{r}) v_{ion}(\mathbf{r}) \quad \text{電子-イオン(原子核)間相互作用}$$

$$+ \frac{1}{2} \int d\mathbf{r} \int d\mathbf{r}' \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \quad \text{電子間クーロン相互作用}$$

$$+ E_{XC}[\rho] \quad \text{交換-相關エネルギー (上記3項で記述できない効果を全て押し込んだもの)}$$

最小値が E_0

最小値を与える電子密度が $\rho_0(\mathbf{r})$

省略してるがイオン同士の
相互作用エネルギーも入る
(Ewald和)

交換-相關エネルギー項に対する近似関数形

局所密度近似(LDA) : J. P. Perdew and A. Zunger, Phys. Rev. B23 (1981) 5048.

一般化勾配近似(GGA) : J. P. Perdew, K. Burke, and, Y. Ernzerhof, Phys. Rev. Lett. 77 (1996) 3865.

密度汎関数理論まとめ②

エネルギー最小化

$$\frac{\delta E[\phi]}{\delta \phi_i^*(\mathbf{r})} = 0 \quad \left[\begin{array}{l} \text{constraint} \\ \int d\mathbf{r} \phi_i^*(\mathbf{r}) \phi_i(\mathbf{r}) = 1 \end{array} \right] \quad \rightarrow \quad \text{系の最安定状態(基底状態)}$$

変分方程式: Kohn-Sham方程式

W. Kohn and L. J. Sham, Phys. Rev. 140 (1965) A1133.

$$\left(-\frac{1}{2} \nabla^2 + v_{KS}(\mathbf{r}) \right) \phi_i(\mathbf{r}) = \varepsilon_i \phi_i(\mathbf{r})$$

電子密度 $\rho(\mathbf{r}) = \sum_{i=1}^N |\phi_i(\mathbf{r})|^2$

$$v_{KS}(\mathbf{r}) = \int d\mathbf{r}' \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} + \frac{\delta E_{XC}}{\delta \rho(\mathbf{r})} + v_{ion}(\mathbf{r})$$

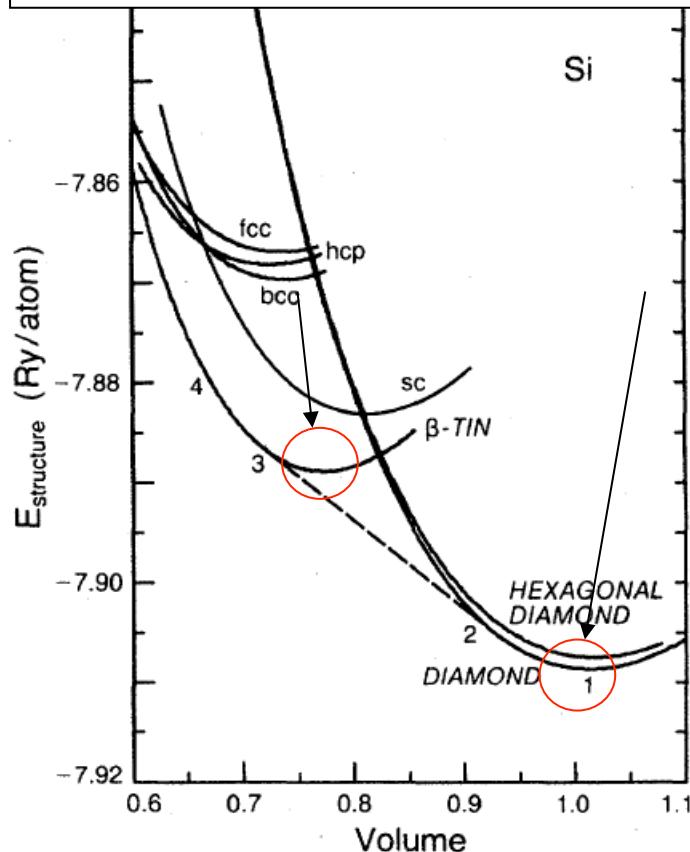
ポテンシャル

- 汎関数最小化 \Leftrightarrow Kohn-Sham方程式を解く(等価)
- 固有値の小さい方から N 個の固有対を求める
- 非線形固有値問題(行列が固有ベクトルに依存)
- $3N$ 次元の問題(Schrödinger) \rightarrow 3次元の問題 $\times N$ 個(DFT)

密度汎関数法における近似の精度

Si結晶

- ・ダイヤモンド構造が最安定
- ・体積を小さくする(圧縮する)と
別の構造(β -スズ構造)の方が安定になる



M. T. Yin and M. L. Cohen
Phys. Rev. B26, 5668 (1982).

単純な近似で

$$E_{XC}[\rho] = E_X[\rho] + E_C[\rho]$$

局所密度近似(LDA)による交換汎関数

$$E_X[\rho] = -C_X \int d\mathbf{r} \rho^{5/3}(\mathbf{r})$$

$$\nu_X[\rho](\mathbf{r}) = \frac{\delta E_X[\rho]}{\delta \rho(\mathbf{r})} = -\frac{5}{3} C_X \rho^{1/3}(\mathbf{r})$$

高い定量性

Si(ダイヤモンド構造)

	DFT計算	実験値
格子定数(Å)	5.37	5.41
体積弾性率(Mb)	0.977	0.988

LDAで良く記述できない物性も多々ある

→新しい汎関数

→DFTを越える枠組み(Green関数、波動関数法)

DFTの数値計算的側面

平面波基底関数展開法 —FFTが必須—

Plane-wave expansion:

$$\phi(\mathbf{r}) = \sum_{\mathbf{G}} f(\mathbf{G}) e^{i\mathbf{G} \cdot \mathbf{r}}$$

Kohn–Sham eq.:

$$\frac{1}{2}G^2f(\mathbf{G}) + \sum_{\mathbf{G}'} v(\mathbf{G} - \mathbf{G}')f(\mathbf{G}') = \varepsilon f(\mathbf{G})$$

Matrix representation

$$\left(\begin{array}{c} H_{\text{GG'}} \\ \end{array} \right) \left(\begin{array}{c} \vec{f} \\ \end{array} \right) = \varepsilon \left(\begin{array}{c} \vec{f} \\ \end{array} \right)$$

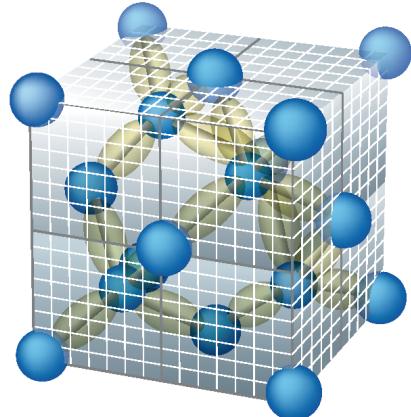
Fast Fourier Transform

$O(N \log N)$

→ ハミルトン演算毎にFFTが必要

→しかしFFTは並列化に不利(全対全通信が発生)

Real-Space finite-difference pseudopotential method



$$\left(-\frac{1}{2} \nabla^2 + v_{ion} + \int d\mathbf{r}' \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} + \frac{\delta E_{XC}[\rho]}{\delta \rho(\mathbf{r})} \right) \phi_j(\mathbf{r}) = \varepsilon_j \phi_j(\mathbf{r})$$

→ Kohn-Sham equation is solved in discretized space

J. R. Chelikowsky *et al.*, Phys. Rev. B50, 11355 (1994).
J.-I. Iwata *et al.*, J. Comp. Phys. 229, 2339 (2010).

- Derivatives → (higher-order) finite difference $\frac{\partial^2}{\partial x^2} \phi(\mathbf{r}) \approx \sum_{m=-6}^6 C_m \phi(x + m\Delta, y, z)$
- Integrals → summation over grid points $\int \phi_m^*(\mathbf{r}) \phi_n(\mathbf{r}) d\mathbf{r} \approx \sum_{i=1}^{N_{grid}} \phi_m^*(\mathbf{r}_i) \phi_n(\mathbf{r}_i) \Delta V$
- Ionic potentials → Pseudopotentials $-\frac{1}{r} \longrightarrow \hat{v}_{ion}$

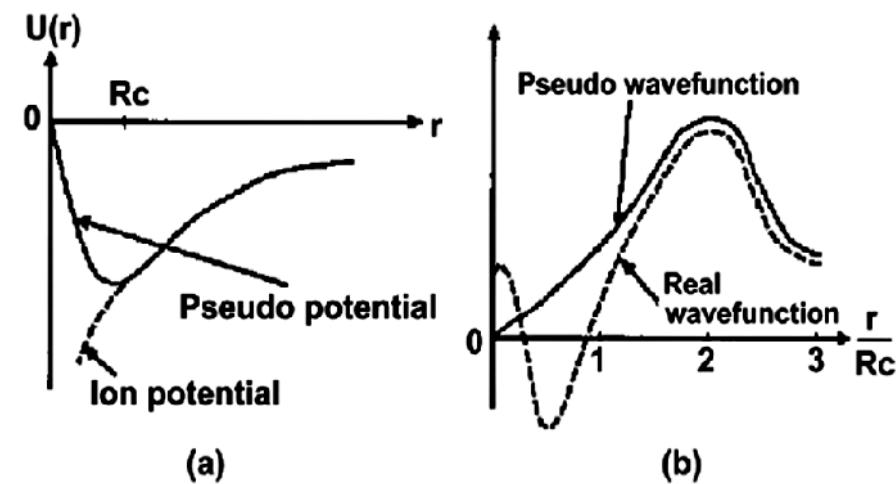
擬ポテンシャル①

- 化学的に活性な価電子状態だけ扱う
- ノードを持たない滑らかな波動関数を扱う
- 計算に都合の良いように作った偽原子だが、価電子状態については本物と変わらない性質を持つ
 - 本物とエネルギー固有値が同じ
 - 散乱の性質が同じ(外から突付いて本物と区別が付かない)

計算する固有状態の本数が減る

粗いグリッド(低いカットオフ波数)

ノルム保存、ウルトラソフト



擬ポテンシャル: 局所+非局所演算子

$$\left(-\frac{1}{2} \nabla^2 + v_H[\rho](\mathbf{r}) + v_{XC}[\rho](\mathbf{r}) + v_{ext}(\mathbf{r}) \right) \phi_n(\mathbf{r}) = \varepsilon_n \phi_n(\mathbf{r}) \quad v_{ext}(\mathbf{r}) = \sum_{I=1}^{N_{atom}} \frac{Z_I}{|\mathbf{r} - \mathbf{R}_I|}$$

$$\left(-\frac{1}{2} \nabla^2 + v_H[\rho](\mathbf{r}) + v_{XC}[\rho](\mathbf{r}) + v_{loc}^{PP}(\mathbf{r}) + \hat{v}_{nloc}^{PP} \right) \phi_n(\mathbf{r}) = \varepsilon_n \phi_n(\mathbf{r})$$

非局所演算子

$$\hat{v}_{nloc}^{PP} \phi_n(\mathbf{r}) = \sum_{a=1}^{N_{atom}} p_a(\mathbf{r}) \int_{\Omega_a} d\mathbf{r}' p_a(\mathbf{r}') \phi_n(\mathbf{r}')$$

$$p_a(\mathbf{r}) \begin{cases} = 0 & |\mathbf{r}| > \Omega_a \\ \neq 0 & |\mathbf{r}| \leq \Omega_a \end{cases} \quad \text{原子 } a \text{ を中心とするある小さい半径の内側} \\ \text{でのみ値を持つ}$$

アルゴリズム①

部分空間反復法(共役勾配法)
(Rayleigh-Ritz法)

問題

$$\left(-\frac{1}{2}L + v_H[\rho] + v_{XC}[\rho] + v_{loc}^{PP} + \hat{v}_{nloc}^{PP} \right) \vec{\phi}_n = \varepsilon_n \vec{\phi}_n$$

N_{grid} 次元ハミルトニアンの固有状態を固有値の小さい方から $\textcolor{teal}{N} \ll N_{grid}$ 本求めよ。

初期ベクトル(求めたい固有状態の本数($\textcolor{teal}{N}$ 本)分用意する)

$$\{\vec{\phi}_1, \vec{\phi}_2, \dots, \vec{\phi}_N\}$$

共役勾配法によるReyleigh商の最小化

$$\varepsilon_n(\vec{\phi}_n) = \frac{\langle \vec{\phi}_n | h_{KS} | \vec{\phi}_n \rangle}{\langle \vec{\phi}_n | \vec{\phi}_n \rangle}$$

$$\vec{\phi}_n = \vec{\phi}_n + \alpha \vec{p}_n$$

という形でベクトル(波動関数)を更新していく

具体的アルゴリズムは
Numerical Recipes
「最適化」と名の付く本
にある。ただしこの時点で
本気の最小化はしない
(反復3~4回でやめる)

この部分に他のアルゴリズムを用いることもできる
(最小残差法、Chebyshevフィルタ法、etc.)
そういうのをひっくるめて「部分空間反復法」と呼ぶ

アルゴリズム②

Gram-Schmidt 直交化

$$\vec{\phi}'_n = \vec{\phi}_n - \sum_{m=1}^{n-1} \vec{\phi}_m \langle \vec{\phi}_m | \vec{\phi}_n \rangle \quad \langle \vec{\phi}'_m | \vec{\phi}'_n \rangle = \delta_{mn} \quad O(N_{\text{grid}} N^2)$$

部分対角化

ここまで得点で求まっている $\{\vec{\phi}'_1, \vec{\phi}'_2, \dots, \vec{\phi}'_N\}$ を基底関数とみなして最適な解を求める。

$$h_{m,n} = \langle \vec{\phi}'_m | h_{KS} | \vec{\phi}'_n \rangle \quad O(N_{\text{grid}} N^2)$$

$$\vec{\phi}_n = \sum_{i=1}^N c_i \vec{\phi}_i \quad (Ritz \text{ vectors}) \quad O(N_{\text{grid}} N^2)$$

$$\begin{pmatrix} & \\ h_{N \times N} & \end{pmatrix} \begin{pmatrix} \vec{c}_i \\ \vdots \\ \vec{c}_i \end{pmatrix} = \varepsilon_i \begin{pmatrix} \vec{c}_i \\ \vdots \\ \vec{c}_i \end{pmatrix} \quad O(N^3)$$

ScaLAPACK, Eigen-k

$\{\vec{\phi}_1, \vec{\phi}_2, \dots, \vec{\phi}_N\}$ ← 一次はこれを入力にして再び共役勾配法へ

密度・ポテンシャルの更新

セルフコンシスティントな解を求めるには

$$\rho_{OUT}[\rho_{IN}] = \rho_{IN}$$

という方程式を解けばよい

Pulay法

$$\rho_{IN}^{guess} = \sum_i \alpha_i \rho_{IN}^{(i)}$$

反復の各ステップで密度をとっておく
IN と OUT の差がなるべく小さくなるように係数を決める

$$\rho_{OUT}^{guess} = \sum_i \alpha_i \rho_{OUT}^{(i)}$$

Broyden法もよく使われる

密度 ρ_{IN}

ポテンシャル

Kohn-Sham方程式

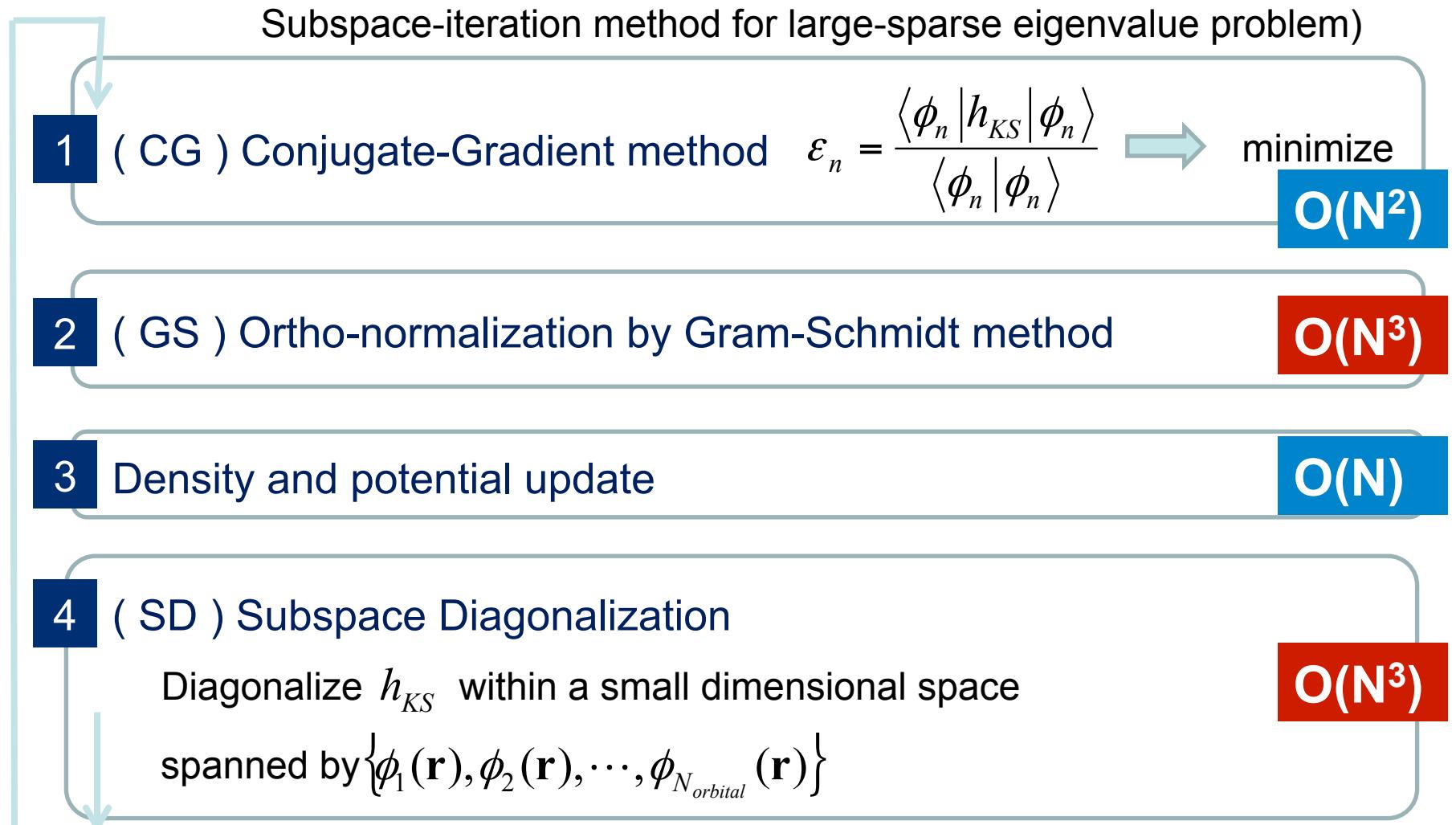
波動関数

密度 ρ_{OUT}



Flow chart of the self-consistent procedure

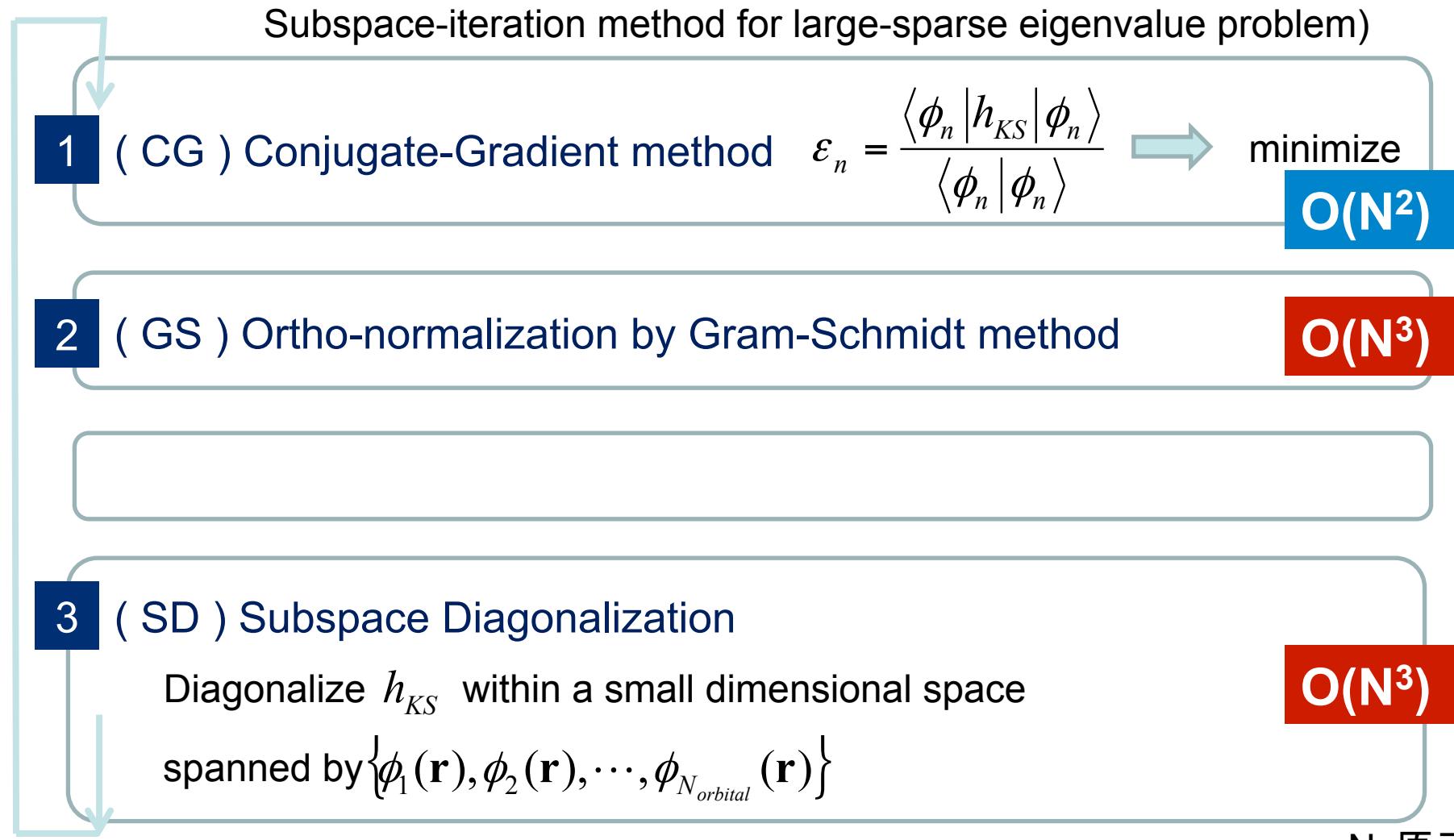
格子点数(10万～1000万点)の次元の行列の固有ベクトルを
下から電子数(1千～10万)本分求める



N: 原子数

Flow chart (Sweep)

擬ポテンシャルのデータから作る初期ポテンシャルや、SCFポテンシャルを初期ポテンシャルとして、ポテンシャルを更新しないで計算を行う（バンド計算、初期波動関数）



RS-DFTの高速化のポイント（行列×行列にしてBLAS 3を使う）

$O(N^3)$ ボトルネック部のBLAS3化（筑波大 高橋大介先生）

Si4096, computational time and performance (PACS-CS 256 CPUs)

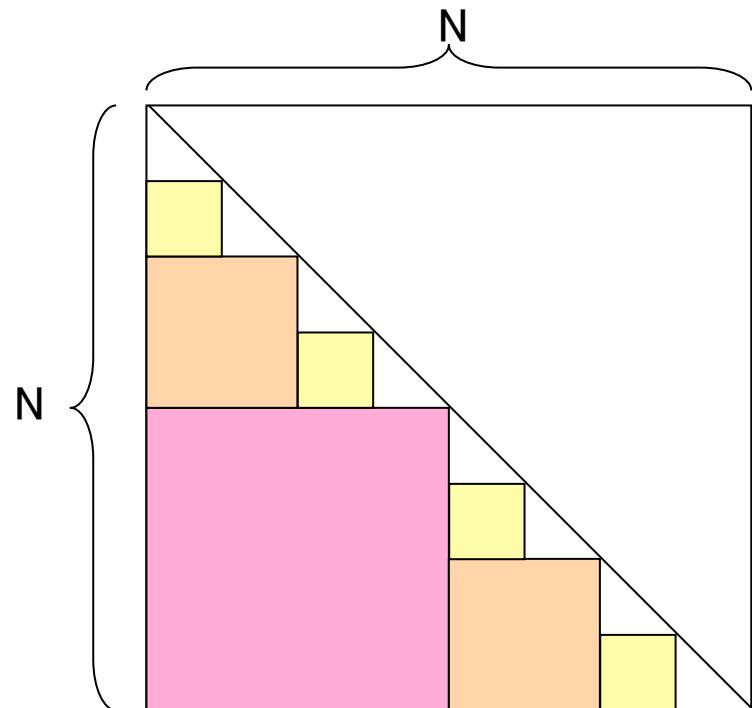
	Time (sec.)	GFLOPS/node
Old algorithm	661	0.7
New algorithm	111	4.3

Theoretical peak : 5.6 GFLOPS/node

$O(N^3)$ bottle necks of DFT calculations can be performed
with **80~90%** of the theoretical peak performance !

ψ_8

再帰的アルゴリズム



・なるべく大きい四角を取って
「[行列 × 行列 \(BLAS 3\)](#)」で計算する
のが効率が良い。

・残った三角部分を素直に
「[行列 × ベクトル \(BLAS 2\)](#)」
で計算するよりも、三角の中に再帰的
に小さい四角を取って「[行列 × 行列](#)」
で計算する方が効率が良い。

Self-Consistent解 → 全エネルギー

$$E[\{\phi_i\}] = \sum_{i=1}^N \int d\mathbf{r} \phi_i^*(\mathbf{r}) \left(-\frac{1}{2} \nabla^2 \phi_i(\mathbf{r}) \right)$$

$$+ \int d\mathbf{r} \rho(\mathbf{r}) v_{ion}(\mathbf{r}) + \sum_{i=1}^N \int d\mathbf{r} \phi_i^*(\mathbf{r}) \hat{v}_{ion}^{nonlocal} \phi_i(\mathbf{r})$$

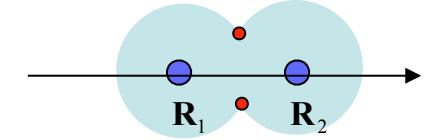
$$+ \frac{1}{2} \int d\mathbf{r} \int d\mathbf{r}' \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}$$

$$+ E_{XC}[\rho]$$

省略してるがイオン同士の
相互作用エネルギーも入る
(Ewald和)

全エネルギーのイオン座標に関する微分

$$\begin{aligned} \frac{\partial E[\{\phi_n\}, \{\mathbf{R}_I\}]}{\partial R} &= \int d\mathbf{r} \frac{\partial v_{loc}^{PP}(\mathbf{r})}{\partial R} \rho(\mathbf{r}) + \sum_{n=1}^N \int d\mathbf{r} \phi_n^*(\mathbf{r}) \frac{\partial \hat{v}_{nloc}^{PP}}{\partial R} \phi_n(\mathbf{r}) \\ &+ \sum_{n=1}^N \left\langle \frac{\partial \phi_n}{\partial R} \left| \left(-\frac{1}{2} \nabla^2 + \nu + \hat{v}_{nloc}^{PP} \right) \right| \phi_n \right\rangle + c.c. \\ &+ \frac{\partial E_{Ewald}}{\partial R} \end{aligned}$$



$$\left(-\frac{1}{2} \nabla^2 + \nu[\rho](\mathbf{r}) + \hat{v}_{nloc}^{PP}(\mathbf{r}) \right) \phi_n(\mathbf{r}) = \epsilon_n \phi_n(\mathbf{r})$$

Kohn-Sham方程式が解けているとして
そのときの波動関数で上の式を計算すると

$$\frac{\partial E[\{\phi_n\}, \{\mathbf{R}_I\}]}{\partial R} = \int d\mathbf{r} \frac{\partial v_{loc}^{PP}(\mathbf{r})}{\partial R} \rho(\mathbf{r}) + \sum_{n=1}^N \int d\mathbf{r} \phi_n^*(\mathbf{r}) \frac{\partial \hat{v}_{nloc}^{PP}}{\partial R} \phi_n(\mathbf{r}) + \frac{\partial E_{Ewald}}{\partial R}$$

擬ポテンシャルの微分は本計算に入る前に簡単に用意できるため、
DFT+擬ポテンシャル法は力の計算が非常に簡単に実行できる

$$-\frac{\partial E[\{\phi_n\}, \{\mathbf{R}_I\}]}{\partial \mathbf{R}_I} = \mathbf{F}_I$$

→ 系が I 番目のイオンに及ぼす力
(Hellman-Feynman 力)

原子構造最適化

DFTの計算→エネルギー最小化問題

$$E[\rho] \xrightarrow{\text{最小化}} E_0 = E[\rho_0]$$

正確には「ある与えられた原子配置」でのエネルギー最小化をやっていることになる

$$E\left[\rho, \mathbf{R}_1^{ini}, \mathbf{R}_2^{ini}, \dots, \mathbf{R}_{N_{atom}}^{ini}\right] \xrightarrow{\text{最小化}} E_0\left(\mathbf{R}_1^{ini}, \mathbf{R}_2^{ini}, \dots, \mathbf{R}_{N_{atom}}^{ini}\right) = E\left[\rho_0, \mathbf{R}_1^{ini}, \mathbf{R}_2^{ini}, \dots, \mathbf{R}_{N_{atom}}^{ini}\right]$$

$$E_0\left(\mathbf{R}_1^{ini}, \mathbf{R}_2^{ini}, \dots, \mathbf{R}_{N_{atom}}^{ini}\right) \geq E_0\left(\mathbf{R}_1^{2nd}, \mathbf{R}_2^{2nd}, \dots, \mathbf{R}_{N_{atom}}^{2nd}\right) \quad \text{となるように原子配置を更新}$$

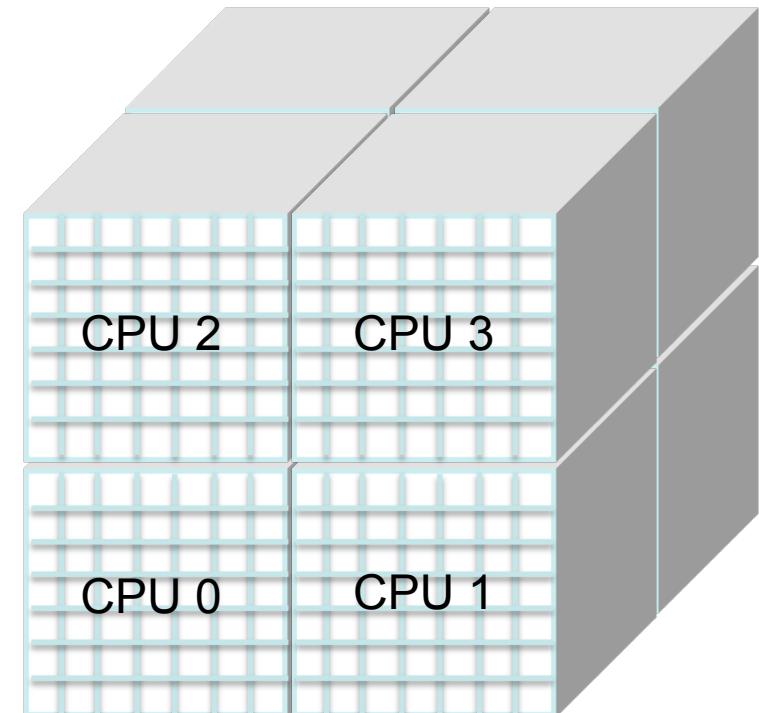
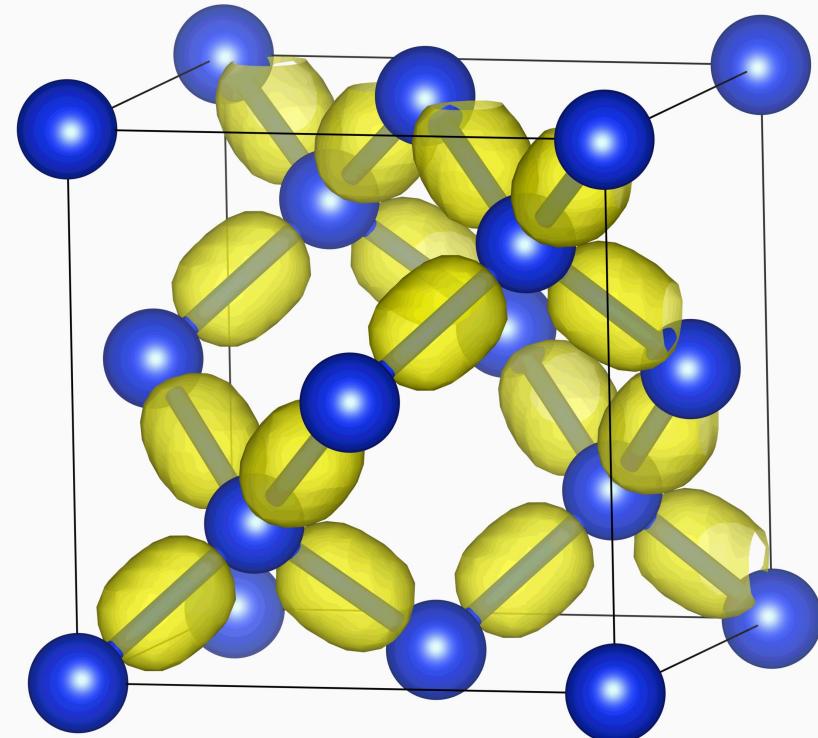
DFT計算 → 原子配置更新 → DFT計算 → …(エネルギー最小まで繰り返す)

→ 原子構造最適化

RSDFTの並列化

RSDFDの並列化

Si結晶のユニットセル(価電子密度の等値面)



$$\rho(\mathbf{r}) = \sum_{n=1}^N |\phi_n(\mathbf{r})|^2$$

MPI並列

KS 方程式 (finite-difference eq.)

$$\left(-\frac{1}{2} \nabla^2 + v_s[\rho](\mathbf{r}) + \hat{v}_{nloc}^{PP}(\mathbf{r}) \right) \phi_n(\mathbf{r}) = \varepsilon_n \phi_n(\mathbf{r})$$

Higher-order finite difference

$$\frac{\partial^2}{\partial x^2} \psi_n(x, y, z) \approx \sum_{m=-6}^6 C_m \psi_n(x + m\Delta x, y, z)$$

`MPI_ISEND, MPI_IRECV`

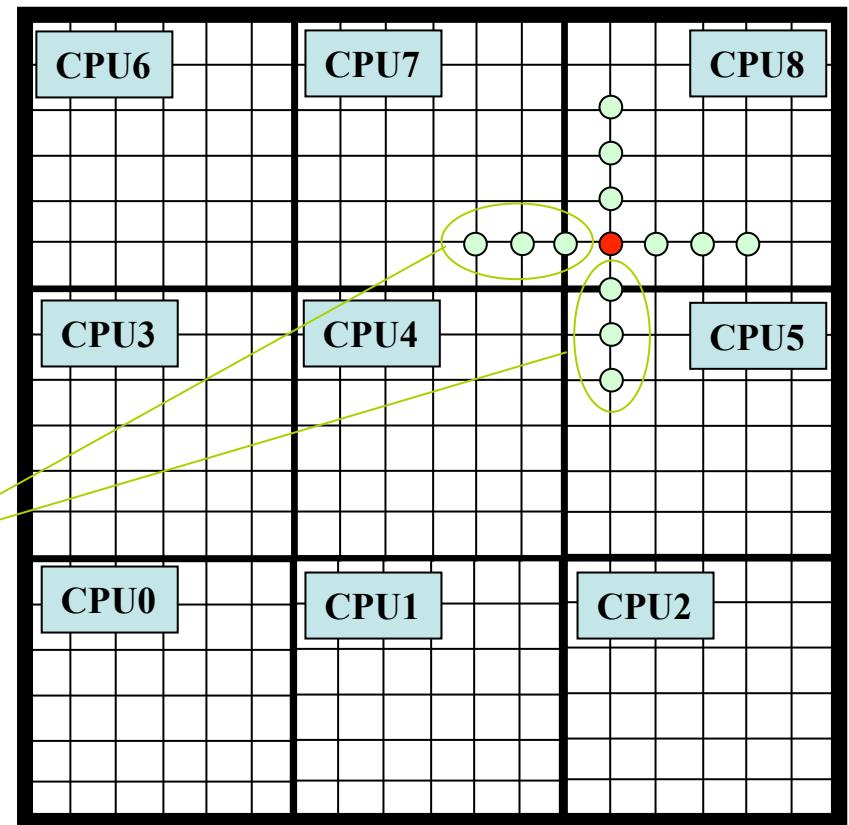
(隣接通信)

Integration

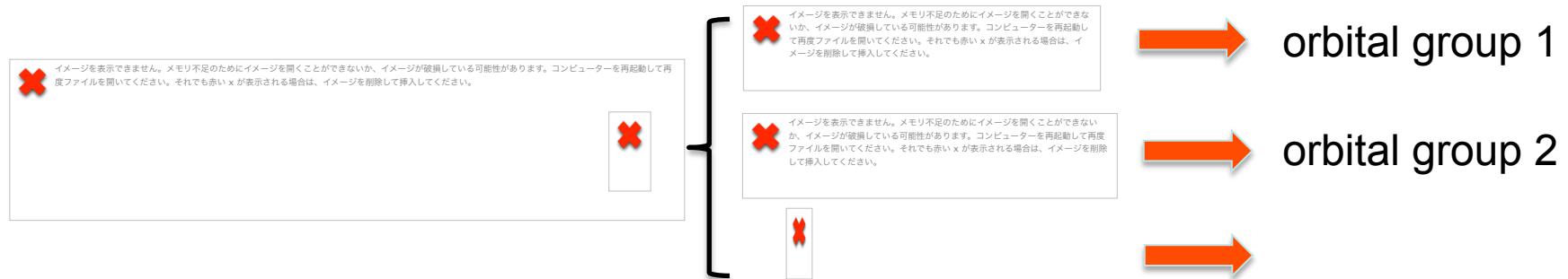
$$\int \psi_m(\mathbf{r}) \psi_n(\mathbf{r}) d\mathbf{r} \approx \sum_{i=1}^{Mesh} \psi_m(\mathbf{r}_i) \psi_n(\mathbf{r}_i) \Delta x \Delta y \Delta z$$

`MPI_ALLREDUCE`

3D grid is divided by several regions for parallel computation.

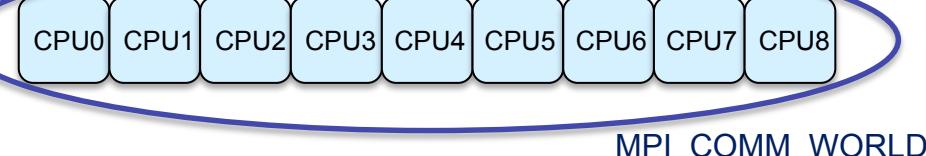


Orbital parallel(Band parallel)



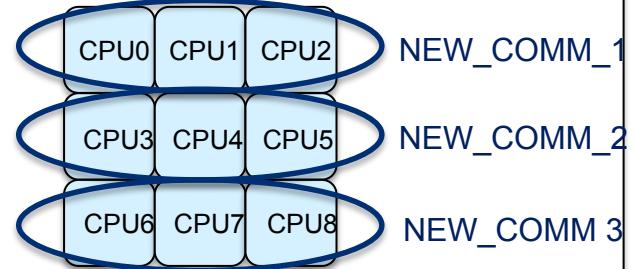
Grid parallel

- N_{grid} is divided by 9 CPUs
- Each CPU handles all orbitals



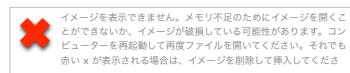
Grid & Orbital parallel

- N_{grid} is divided by 3 CPUs
- Each CPU handles 1/3 of orbitals



K computer has more than 80,000 CPUs

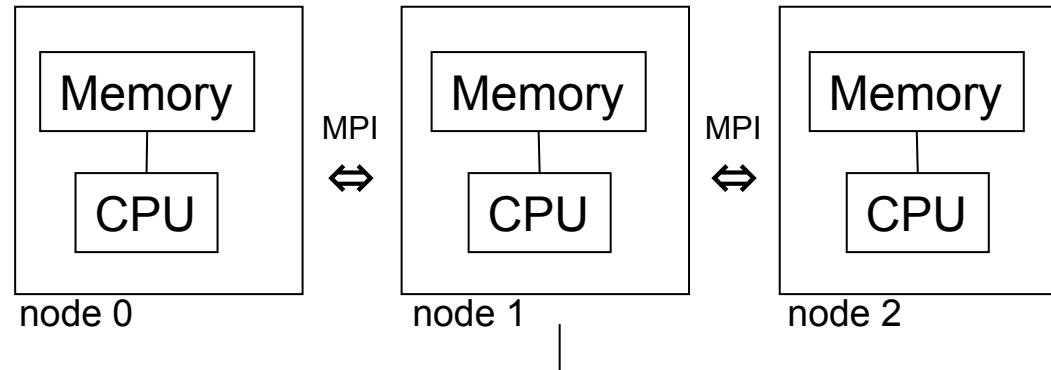
Cost of *MPI_ALLREDUCE* :



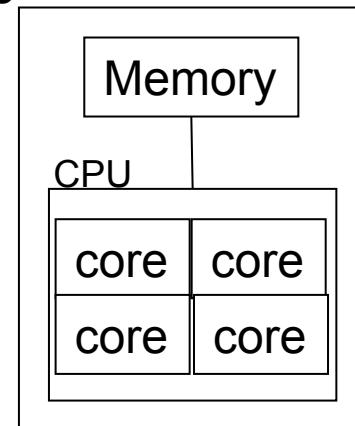
Cost of global comm. is much reduced

分散メモリ, 共有メモリ, マルチコア

Distributed-Memory architecture

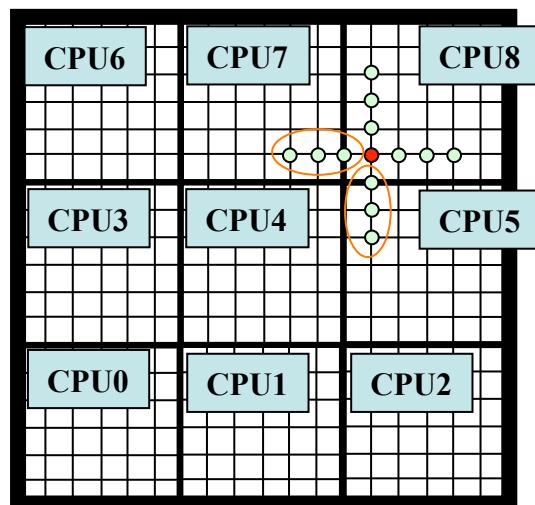
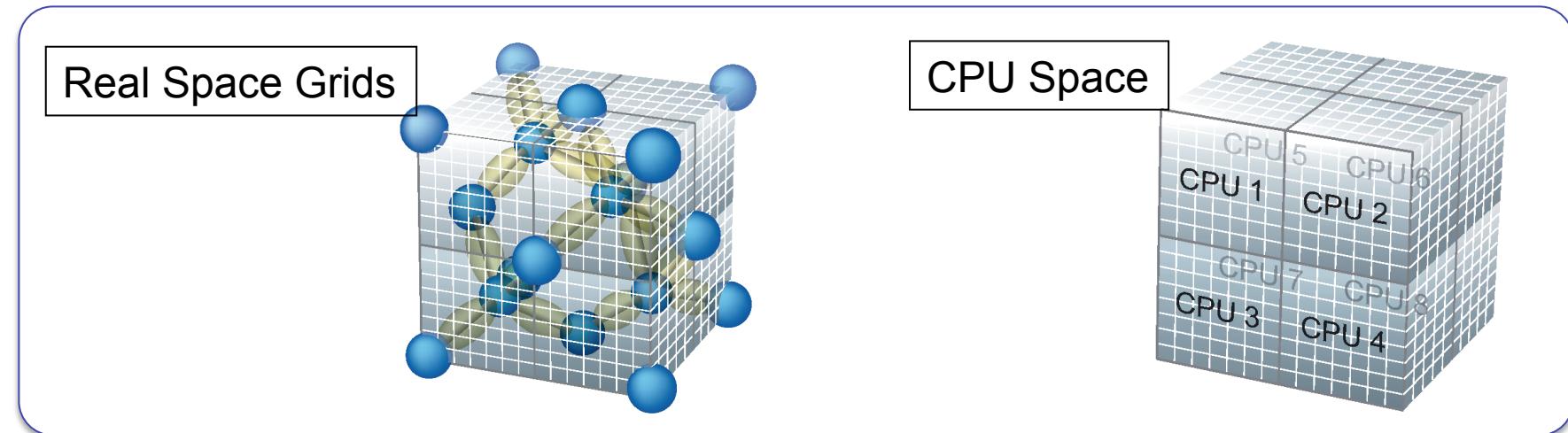


Shared-Memory architecture



← この中もMPIで並列化できる
(flat-MPI)

Grid parallel (MPI & OpenMP)

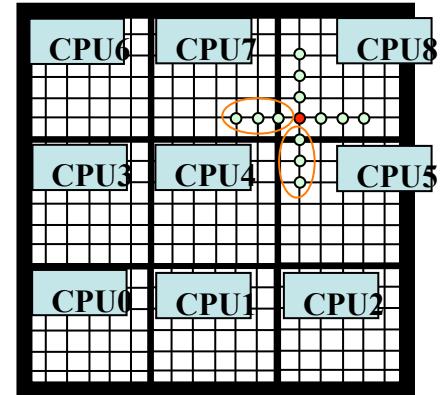


- ◆ MPI (Message-Passing Interface) library
 - MPI_ISEND, MPI_IRecv → finite-difference calc.
 - MPI_ALLREDUCE → global summation

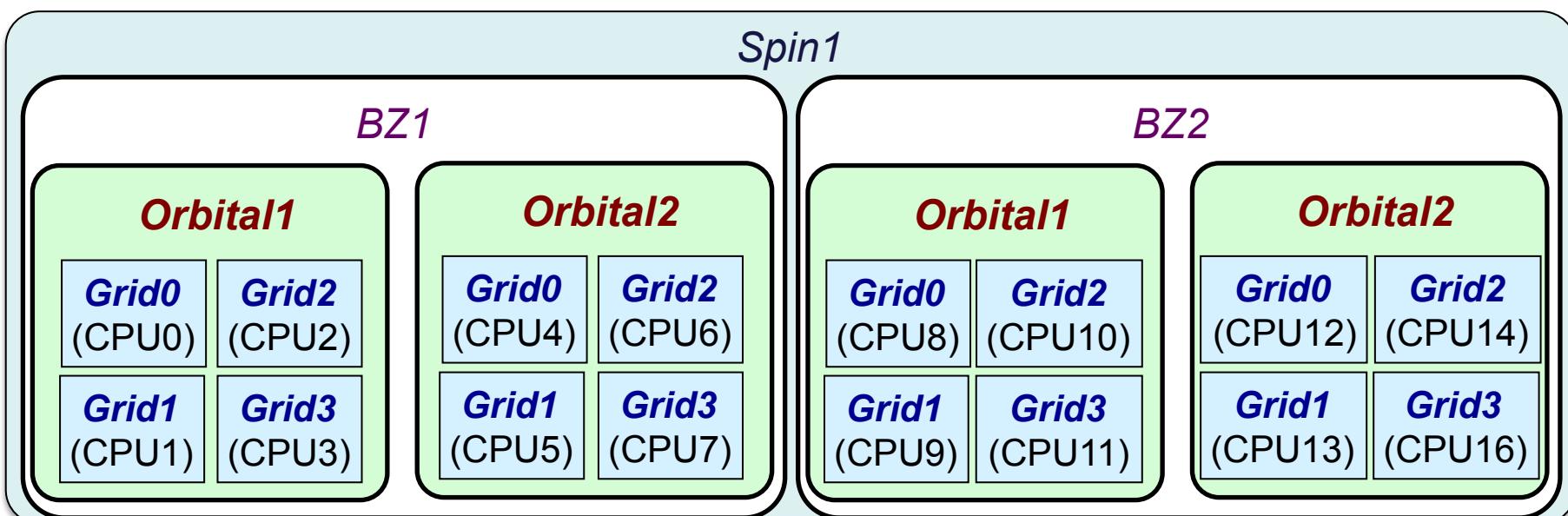
- ◆ OpenMP
 - Further grid parallelization (within each node) is performed by multi-thread parallel

Grid, band, k, spin Parallelization

- ◆ MPI (Message-Passing Interface) library
 - MPI_ISEND, MPI_RECV → finite-difference calc.
 - MPI_ALLREDUCE → global summation
- ◆ OpenMP
 - Further grid parallelization (within each CPU) is performed by thread parallelization



Example of CPU allocation for 4-grid, 2-orbital, 2-k, 1-spin parallelization

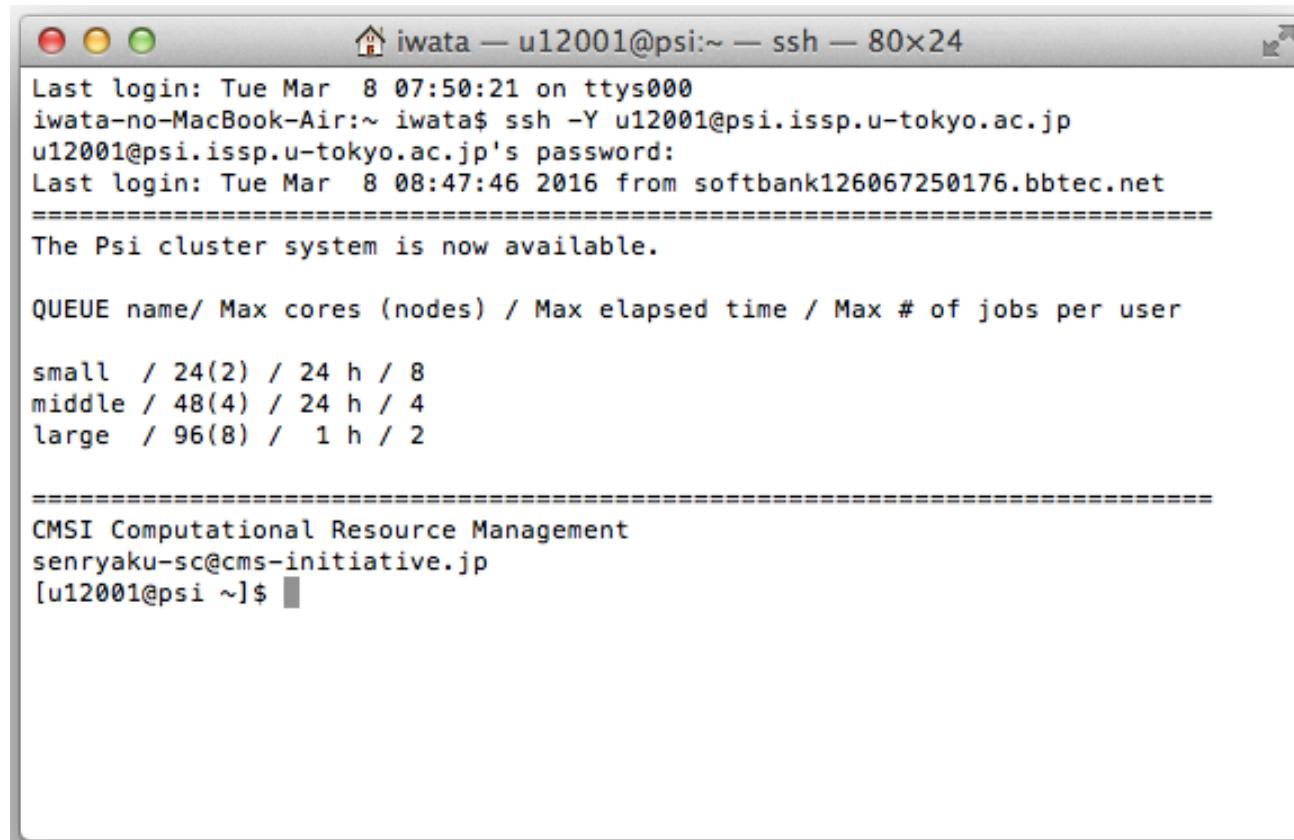


RSDFTの利用

計算機へのログイン方法

ターミナルソフトを起動
(Windowsの場合 puttyなどをインストールする必要があるかもしれない)

```
ssh -Y ユーザー名@psi.issp.u-tokyo.ac.jp
```



The screenshot shows a terminal window titled "iwata — u12001@psi:~ — ssh — 80x24". The window contains the following text:

```
Last login: Tue Mar  8 07:50:21 on ttys000
iwata-no-MacBook-Air:~ iwata$ ssh -Y u12001@psi.issp.u-tokyo.ac.jp
u12001@psi.issp.u-tokyo.ac.jp's password:
Last login: Tue Mar  8 08:47:46 2016 from softbank126067250176.bbtec.net
=====
The Psi cluster system is now available.

QUEUE name/ Max cores (nodes) / Max elapsed time / Max # of jobs per user

small  / 24(2) / 24 h / 8
middle / 48(4) / 24 h / 4
large  / 96(8) / 1 h / 2

=====
CMSI Computational Resource Management
senryaku-sc@cms-initiative.jp
[u12001@psi ~]$
```

プログラムのダウンロード

- <https://github.com/j-iwata/RSDFT>

j-iwata / RSDFT

Code Issues 0 Pull requests 0 Pulse Graphs

No description or website provided.

8 commits 1 branch 0 releases 0 contributors

Branch: master New pull request New file Find file HTTPS https://github.com/j-iwata/RSDFT Download ZIP

		Latest commit faa2458 22 hours ago
doc	RSDFT (ver.1.2.0)	4 days ago
examples	RSDFT (ver.1.1.0)	15 days ago
input	RSDFT (ver.1.1.0)	15 days ago
pseudopotential	RSDFT (ver.1.2.1)	22 hours ago
src	RSDFT (ver.1.2.1)	22 hours ago
utility	RSDFT (ver.1.1.0)	15 days ago
.gitignore	RSDFT (ver.1.0.3)	a month ago
HISTORY	RSDFT (ver.1.2.1)	22 hours ago
LICENSE	RSDFT (ver.1.0.0)	2 months ago
README.md	RSDFT (ver.1.2.1)	22 hours ago
README.md		

もし git に馴染みがあれば

```
$ git clone https://github.com/j-iwata/RSDFT
```

でも取得できます。

gitの管理機能(変更履歴を残したり、間違えたら元に戻したり)
が使えるので、慣れていればこちらの方が便利

コンパイル

- 1) cd src/
- 2) cp make.inc.org make.inc
- 3) make.inc を編集し
- 4) コンパイラ, プリプロセッサ, ライブラリを指定する([ユーザーガイド参照](#))
- 5) make
- 6) 実行ファイル名 `rsdft.x`

必要なファイル

- fort.1 (基本パラメータ)
 - fort.970 (原子座標)
 - 擬ポテンシャルファイル
-
- fort.2 (孤立系計算の場合)

原子座標

- fort.970 というファイルに記述
- XYZ形式, CIF形式も読み込む
- 格子ベクトルをこちらに書くこともできる(fort.1と両方に書いた場合は fort.1 を優先する)

原子座標の例(RSDFTのXYZと格子座標)

N₂ 分子

RSDFT-XYZ形式 (格子座標形式と同様にユニットセルの情報を書く事もできる)

XYZ

1	2	7 /	原子種数	原子数	原子番号(原子種数だけ並べる)
1	0.000000000000	0.000000000000	1.02	1 /	原子種番号 座標(単位は bohr) 力の拘束
1	0.000000000000	0.000000000000	-1.02	1 /	

RSDFT-格子座標形式

AX	20.000	/ ax		
A1	1.00	0.00	0.00	/ aa(1:3,1)
A2	0.00	1.00	0.00	/ aa(1:3,2)
A3	0.00	0.00	1.00	/ aa(1:3,3)

AA

1	2	7 /		
1	0.000000000000	0.000000000000	0.051	1
1	0.000000000000	0.000000000000	-0.051	1

格子座標

$$\mathbf{a}_1 = \begin{pmatrix} a_{1x} \\ a_{1y} \\ a_{1z} \end{pmatrix} \quad \mathbf{a}_2 = \begin{pmatrix} a_{2x} \\ a_{2y} \\ a_{2z} \end{pmatrix} \quad \mathbf{a}_3 = \begin{pmatrix} a_{3x} \\ a_{3y} \\ a_{3z} \end{pmatrix}$$

$$\mathbf{R}_{atom} = \alpha \mathbf{a}_1 + \beta \mathbf{a}_2 + \gamma \mathbf{a}_3$$

原子单位
1 (a.u.) = 0.529177 (\AA)

原子座標の例(XYZ, CIF)

XYZ形式 (C₆₀)

```
60
Fullerene
C 1.89038 -0.32037 -3.41132
C 1.38778 1.03202 -3.42850
C -0.06240 0.96979 -3.45699
C -0.44318 -0.44456 -3.45638
C 0.77758 -1.24455 -3.42397
C 3.03726 -0.66070 -2.61255
C 2.07017 2.03660 -2.64609
C -0.79950 1.98887 -2.69734
C -1.64150 -0.85522 -2.69471
C 0.80357 -2.47441 -2.64349
C 3.70200 0.33480 -1.82737
: : : :
```

CIF形式 (Al₂O₃)

```
#=====
# CRYSTAL DATA
#
_pd_phase_name          'Al2O3'
_cell_length_a           4.759(1)
_cell_length_b           4.759(1)
_cell_length_c           12.991(5)
_cell_angle_alpha         90
_cell_angle_beta          90
_cell_angle_gamma         120
_symmetry_space_group_name_H-M   'R -3 c'
_symmetry_Int_Tables_number      167

loop_
_symmetry_equiv_pos_as_xyz
'x, y, z'
'-x, -y, -z'
'y, x-y, z'
....
```

座標の単位はÅ
ユニットセルの情報は fort.1 に書く

ユニットセル(格子ベクトル)
の情報も含まれている

実習① Siのバンド構造 (SCFポテンシャル)

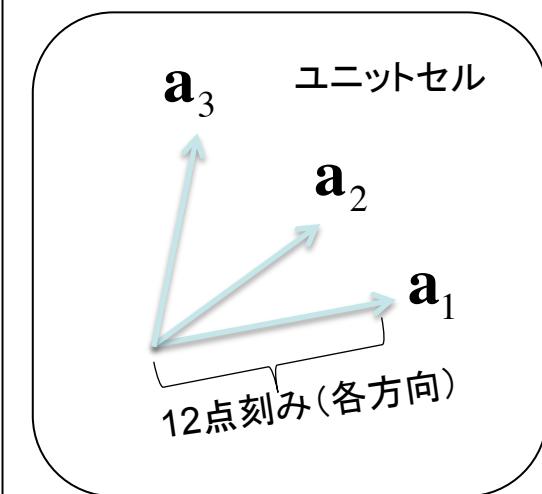
[examples/Si2/band_calc_step_by_step/step1_scf/fort.1](#)

```
# RSDFT
XCtype 'LDAPZ81'          / XCtype
AX 10.261                   / ax
A1 0.00 0.50 0.50           / aa(1:3,1)
A2 0.50 0.00 0.50           / aa(1:3,2)
A3 0.50 0.50 0.00           / aa(1:3,3)
Nband 8                      / Nband
NK 8                         / nk
MMM1 5 5 5                   / mmm(1:3,2)
MMM2 2 2 2                   / mmm(1:3,2)
NGRID 12 12 12               / Ngrid(1:3)
PP 2 'Si_psv.dat'           / file_ps(1)
scfconv 1.d-15                / scf_conv
PROCS 1 1 1 1 8 1            / np_2d(1:6)
Diter 100                     / Diter
Nsweep 5                      / Nsweep
IC 0                          / IC
OC 2                          / OC
ETLIMIT 10000.d0              / etime_limit
SWSCF 1                        / iswitch_scf
INFOLEVEL 1

SWBAND 0                      / iswitch_band
BAND 7 24 20 1.d-7 100        / nbk mb_band mb2_band esp_conv_tol maxiter_band
1.0 1.0 0.75 0.0 0.5 0.75 0.75 0.5
0.0 0.5 0.375 0.0 0.5 0.375 0.25 0.0
0.0 0.5 0.375 0.0 0.5 0.375 0.5 0.5
30 5 10 10 10 5 5
```

fort.970

1	2	/ MKI, MI		
1	0.00	0.00	0.00	1 / Kion(1), asi(1:3,1)
1	0.25	0.25	0.25	1 / Kion(2), asi(1:3,2)



Brillouin Zone サンプリング

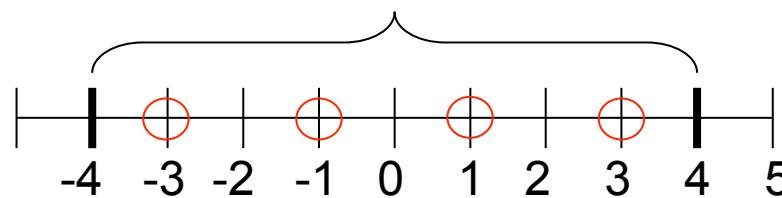
8
5 5 5
2 2 2

/ nk
/ mmm(1:3,1)
/ mmm(1:3,2)



○の付いた点をサンプル

第一Brillouin Zone



INFOLEVEL (標準出力の情報量)

- = 0 (または指定なし)
 - 最小限の情報のみ
- = 1
 - 固有値を少し(占有/非占有を5本ずつ程度)表示

実習① Siのバンド構造(バンド計算)

examples/Si2/band_calc_step_by_step/step2_band/fort.1 (変更部のみ)

```
Nsweep 0           / Nsweep
IC 2             / IC
OC 0             / OC
ETLIMIT 10000.d0 / etime_limit
SWSCF 0          / iswitch_scf

SWBAND 1          / iswitch_band
BAND 7 24 20 1.d-7 100 / nbk mb_band mb2_band esp_conv_tol maxiter_band
1.0 1.0 0.75 0.0 0.5 0.75 0.75 0.5
0.0 0.5 0.375 0.0 0.5 0.375 0.25 0.0
0.0 0.5 0.375 0.0 0.5 0.375 0.5 0.5
30 5 10 10 10 5 5
```

7区間を直線で結び、各区間を
30, 5, 10, 10, 10, 5, 5
の刻みでk点をサンプルし、
各k点でsweep計算を行う

$$\mathbf{k} = \alpha \mathbf{b}_1 + \beta \mathbf{b}_2 + \gamma \mathbf{b}_3$$

計算が完了すると
band_eigv(各k点、バンドでの固有値)
band_dedk(そのk微分(ベクトル))
というファイルができる

実習① Siのバンド構造(バンドのプロット)

[examples/Si2/band_calc_step_by_step/step3_plot/](#)

utility/band2gp.f90 をコンパイル

band_eigv があるディレクトリでそれを実行

plot_band, fort.2001, fort.2002, … というファイルができる

gnuplotの実行

\$ gnuplot

gnuplot> load 'plot_band'

N₂分子（スーパーセル計算）

examples/N2/supercell/scf_and_opt/fort.1

```
# RSDFT
XCtype 'LDAPZ81'           / XCtype
NGRID 40 40 40              / Ngrid(1:3)
PP 2 'N_psv.dat'            / file_ps(1)
scfconv 1.d-15                / scf_conv
PROCS 1 2 2 1 1 1            / np_2d(1:6)
Diter 100                    / Diter
Nsweep 5                     / Nsweep
IC 0                         / IC
OC 3                         / OC
ETLIMIT 1.d10                 / etime_limit
INFOLEVEL 1                   / info_level
SWSCF 1                       / iswitch_scf

SWOPT 1                       / iswitch_opt
ATOMOPT1 50 8 5               / ncycl, most, nrfr
ATOMOPT2 0.5d0 1.d-10 5.d-4 1.d-1 / okatom,
eeps, feps, decr
ATOMOPT3 100                  / diter_opt

TIM
0.0 0.0 0.0
0.0 0.0 0.0
0.0 0.0 1.0
```

fort.970

AX	20.000	/ ax
A1	1.00 0.00 0.00	/ aa(1:3,1)
A2	0.00 1.00 0.00	/ aa(1:3,2)
A3	0.00 0.00 1.00	/ aa(1:3,3)
AA		
	1 2 7 /	
1	0.00000000000000	0.00000000000000 0.051 1
1	0.00000000000000	0.00000000000000 -0.051 1

初期ボンド長 2.04 bohr (1.08 Å)
(実験値 1.098 Å)

力のx,y,z成分に行列を掛けて力の方向を制限

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} F_x \\ F_y \\ F_z \end{pmatrix}$$

N₂分子（孤立系境界条件計算）

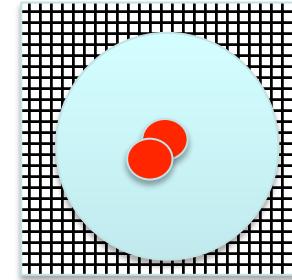
[examples/N2/rsmol/scf_and_opt/fort.1](#)

```
# RSDFT
SYStype 1           / SYStype
XCtype 'LDAPZ81'    / Xctype
```

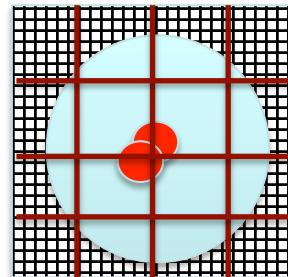
[examples/N2/rsmol/scf_and_opt/fort.2](#)

```
1      / Box_Shape
0.5 10.0 / Hsize Rsize
1      / eqdiv
```

実数版でしか動かない！
(-D_DRSDFT_をつけて再コンパイル)



球の外で波動関数 0
という境界条件



eqdiv=1 は球内のグリッド点数が
なるべく均等になるように並列化する
(等間隔に分割すると左図のようになる)

イオン化のような荷電状態の計算がスーパーセルよりも直接的に実行できる
(スーパーセル計算はユニットセルが常に中性である必要があるので、人為的な一様背景電荷を導入する)

その他のサンプル

Si64原子セル中の单原子空孔

構造最適化～バンドアンフォールディング

(Si2原子のプリミティブセルのBZでバンドを描く)

```
SWBAND 1           / iswitch_band
ETLIMIT 5.d10      / etime_limit
ATOMOPT1 100 10 5   / ncycl, most, nrfr
ATOMOPT2 0.5d0 1.d-10 5.d-4 1.d-1 / okatom,
eeps, feps, decr
ATOMOPT3 100       / diter_opt

BAND 1 160 140 1.d-7 100 /
0.0 0.0 /
0.0 0.5 /
0.0 0.5 /      プリミティブセルでバンドを各設定を書く
50

BANDUF
10.16          “BANDUF”の後にプリミティブセル
0.0 0.5 0.5    のユニットセルの情報を書く
0.5 0.0 0.5    (これらがあると unfoldingのバンド計算に)
0.5 0.5 0.0
```

計算が完了すると band_ufld というファイルができる。

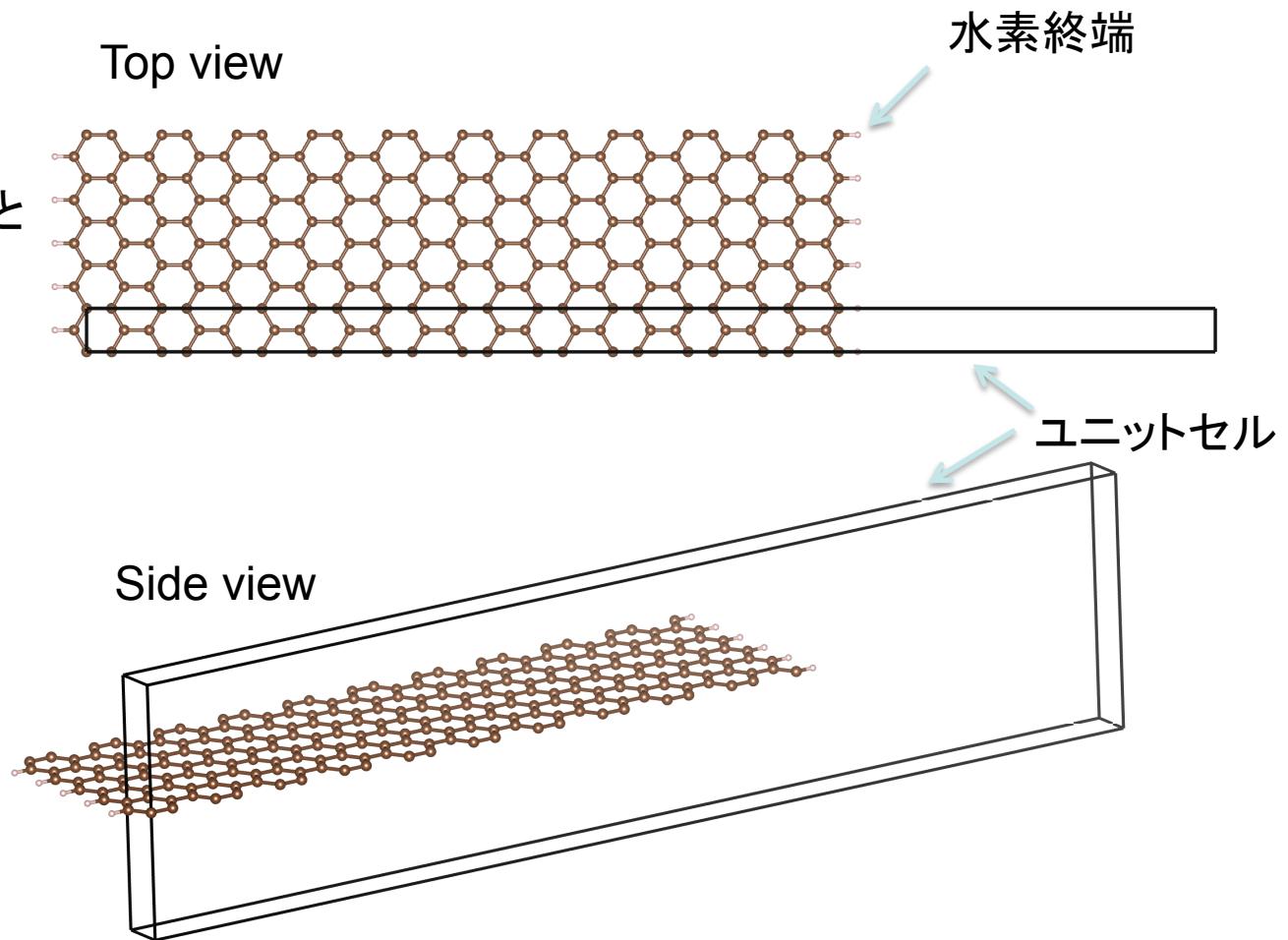
utility/ufld2gp.f90 をコンパイル&実行

gnuplotを起動し、plot_ufld を load

グラフェンナノリボン(ジグザグエッジ)

examples/GNR/antiferro
examples/GNR/ferro

初期スピン配置を変えると
異なる解に収束する



初期 спин配置の指定(fort.980)

```
# RSDFT
XCtype 'LDAPZ81'
Nspin 2
Ndspin -1.0
Nfixed 500
NK 32           / nk
MMM1 32 17 32  / mmm(1:3,2)
MMM2 32 2 32   / mmm(1:3,2)
NGRID 312 12 72 / Ngrid(1:3)
PP 2 'C_psv.dat' / ippform(1) file_ps(1)
PP 2 'H_psv.dat' / ippform(1) file_ps(1)
scfconv 1.d-15  / scf_conv
ekbt 1.d-3      /
PROCS 4 1 2 1 8 1 / np_2d(1:6)
Diter 300        / Diter
Nsweep 100       / Nsweep
IC 0             / IC
OC 2             / OC
SWSCF 1          / scf
SWOPT 0          / opt
SWBAND 0         / band
ETLIMIT 1.d10    / etime_limit

BAND 1 120 100 1.d-7 200
0.0 0.0 0.5
0.0 0.5 0.0
0.0 0.0 0.0
50 50
```

別ファイル(fort.980)で指定する
スピン配置を初期スピンとする

fort.980

```
4 4 1.0d0
42 42 -1.0d0
```

(fort.970に記述された順番で)

4番目～4番目までの原子のスピン↑↓差を 1.0
42番目～42番目までの原子のスピン↑↓を -1.0
指定の無い原子についてはスピン差0.0

HSE 汎関数

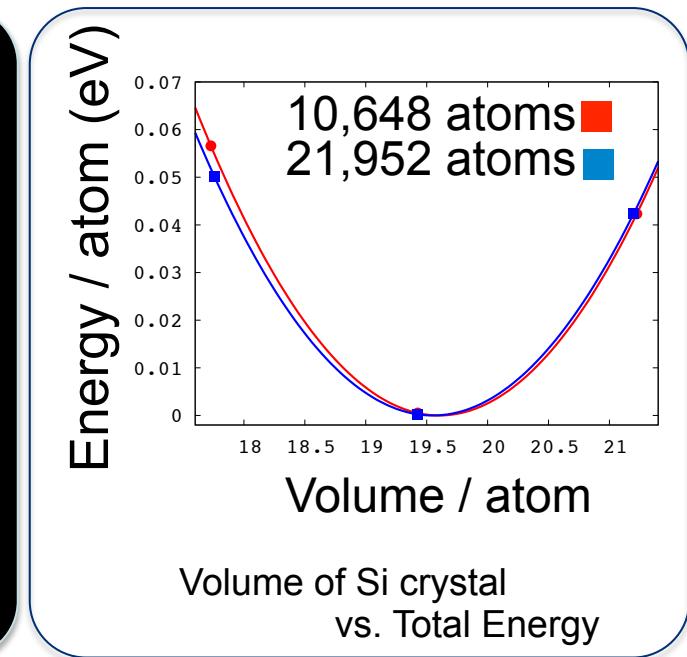
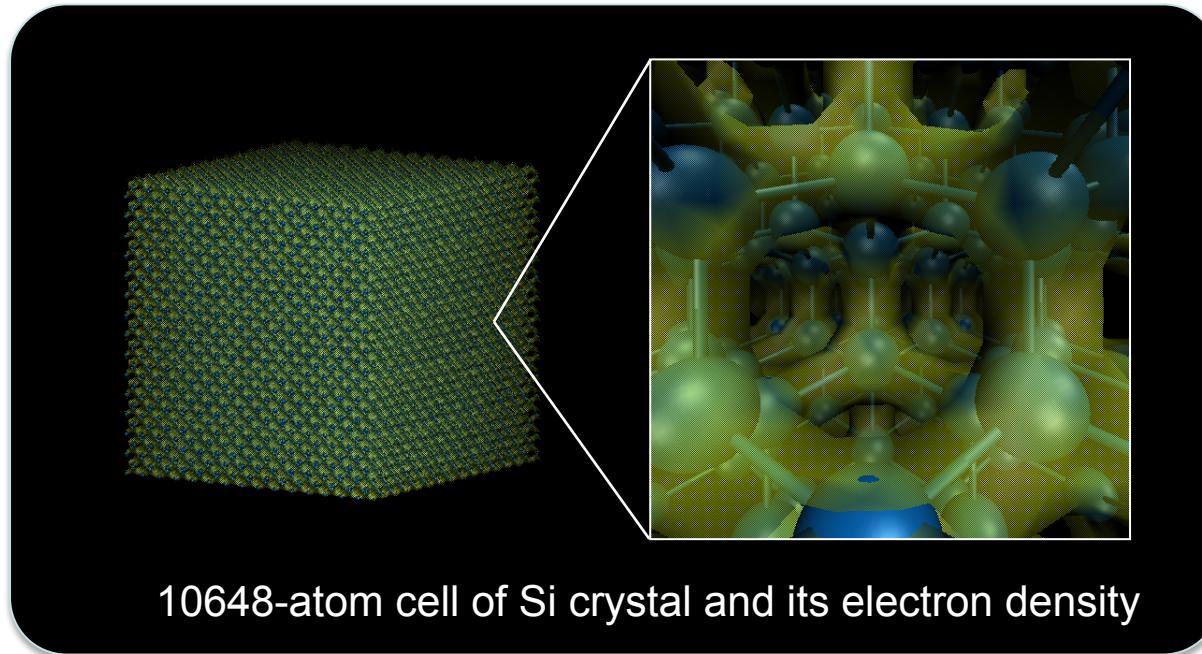
examples/Si2_HSE/

Si2原子のハイブリッド汎関数(HSE06)によるバンド計算

LDA, GGA に比べて計算が非常に重い
FFTの性能が重要になるので **-D_FFTF** を付けて再コンパイル
(fftwが使えるならこちらを使ってもよい)

FFTE
<http://www.ffte.jp>

格子定数の最適化



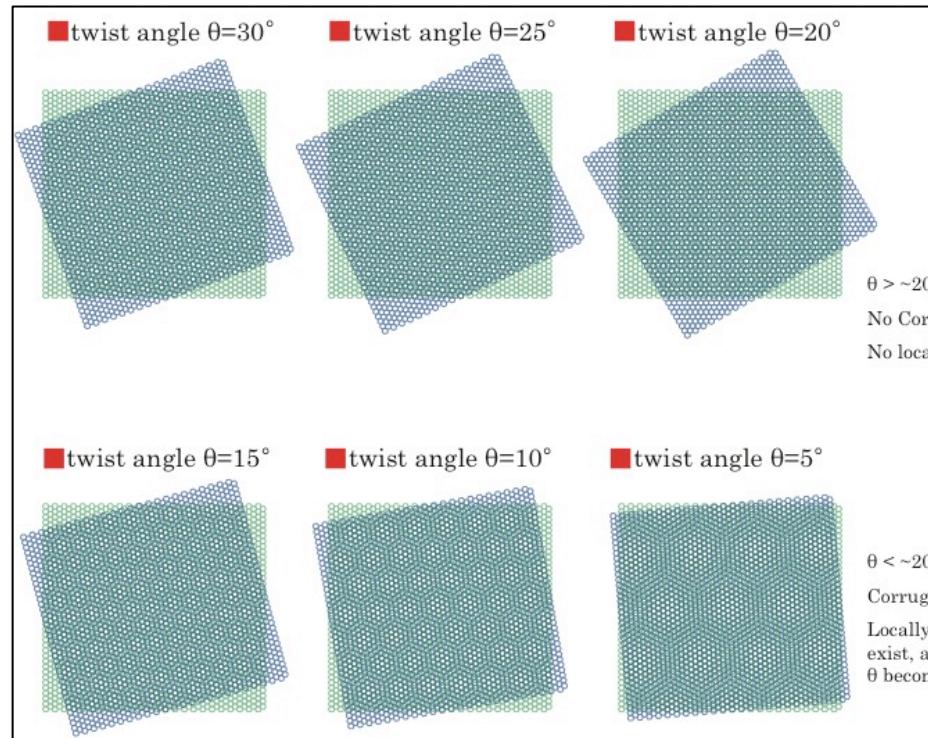
LDA = Local Density Approximation (exchange-correlation functional)

	LDA (21,952 atoms)	LDA (10,648 atoms)	LDA (2 atoms)	Expt.
Lattice constant	5.39Å	5.39Å	5.39Å	5.43Å

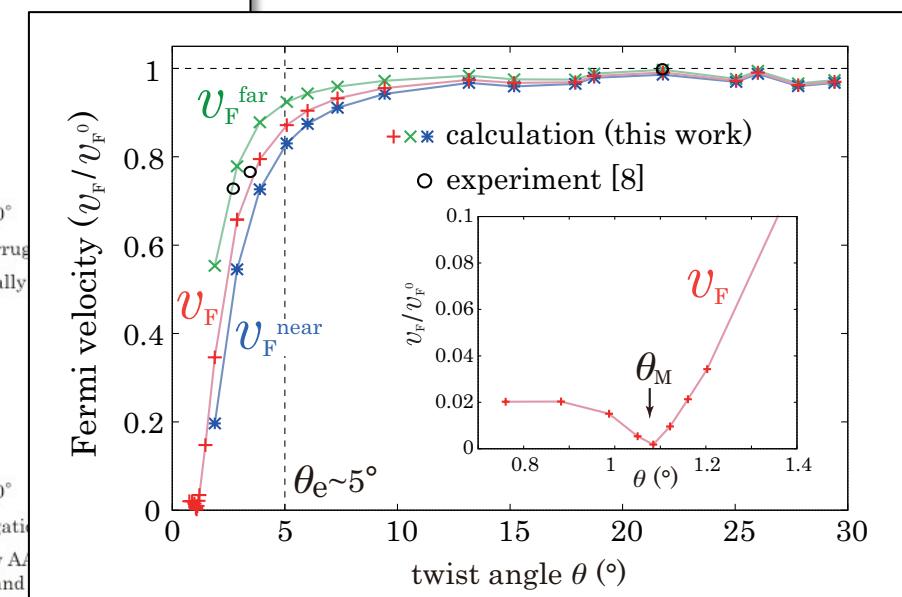
計算時間の目安

（横軸は計算時間、縦軸は問題の大きさ）

twisted bilayer graphene



K. Uchida et al., Phys. Rev. B (2014)

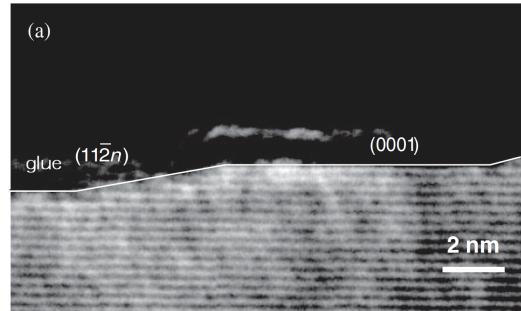


Elapsed time for 1 step of SCF loop at K computer (841 nodes)

atoms	SCF	CG	GS	SD
22,708	2980	446	710	1452

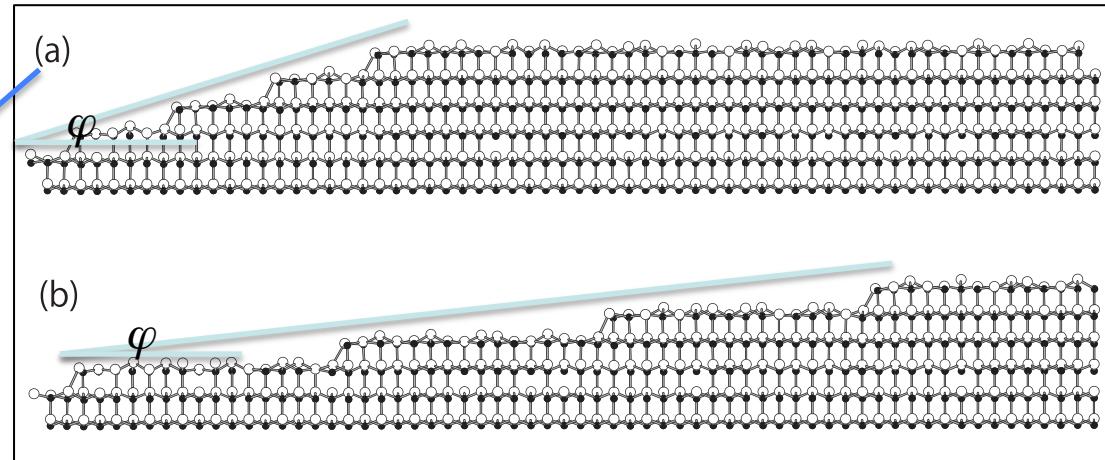
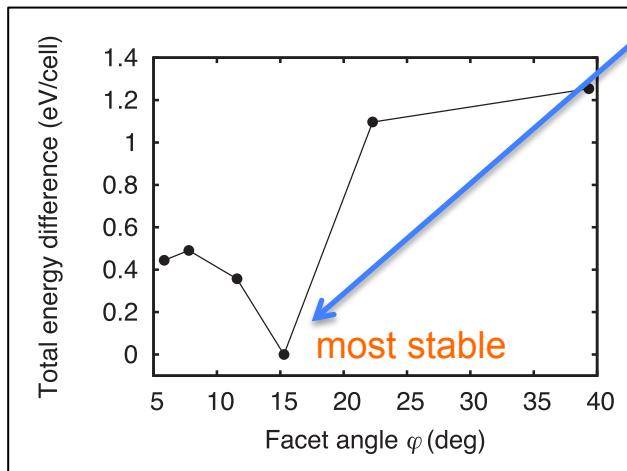
single k point
real*8

magic angle in nanotacets on sic(0001) vicinal surface



TEM image
M. Fujii and . Tanaka
Phys. Rev. Lett. 99, 06102 (2007)

K. Sawada, J.I, A. Oshiyama
Appl. Phys. Lett. 104, 051605 (2014).



Elapsed time for 1 step of SCF loop at K computer (960 nodes)

atoms	SCF	CG	GS	SD
968	29.55	6.38	4.08	6.41

4 k points
complex*16

複素数になると計算量は
実数のときの4倍になる

full-node test @K

system : Si nanowire (107,292 atoms)
grid : $576 \times 576 \times 180 = 59,719,680$
band : 230,400

nodes: 82,944 (10.6 PFLOPS peak)
cores : 663,552

Elapsed time for 1 step of SCF iteration

	SCF	CG $O(N^2)$	GS $O(N^3)$	SD $O(N^3)$	SD (mate)	SD (Eigen)	SD (rotation)
time (s)	2931	160	946	1797	525	493	779
PFLOPS	5.48	0.06	6.70	5.32	6.15	0.01	8.14
peak%	51.67	0.60	63.10	50.17	57.93	1.03	76.70



計算時間の見積もり

$$\begin{aligned} \text{格子点数 } M_g &\propto \text{原子数 } N \\ \text{バンド数 } M_b &\propto \text{原子数 } N \end{aligned}$$

あるグリッド点数、バンド数での計算時間がわかっているとき

$$\text{計算時間} \propto M_g \times M_b^2$$

計算時間はグリッド点数の1乗、バンド数の2乗に比例

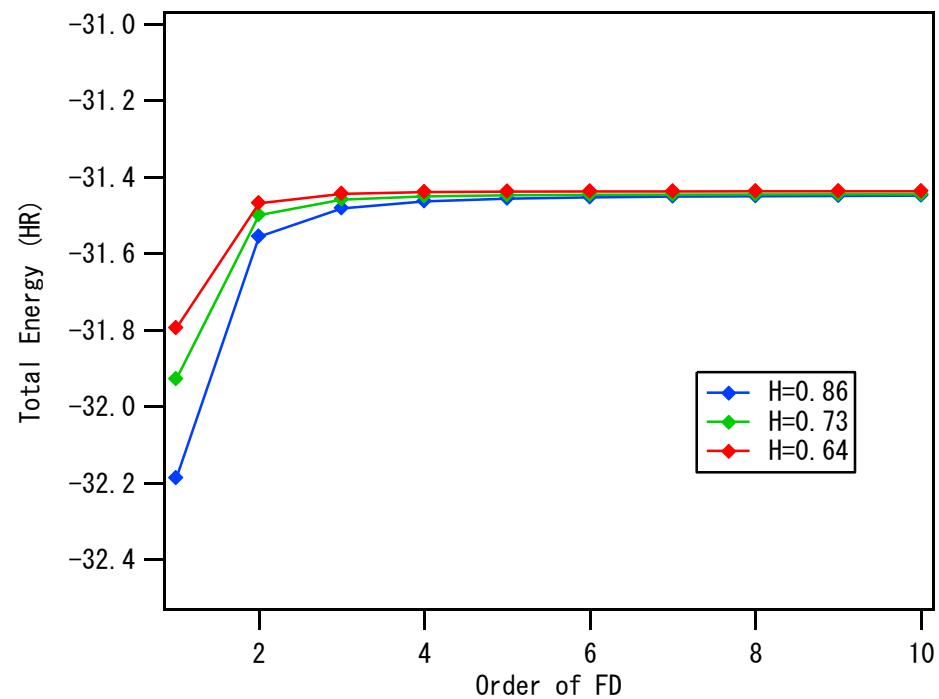
付録

（付録）

差分公式の次数

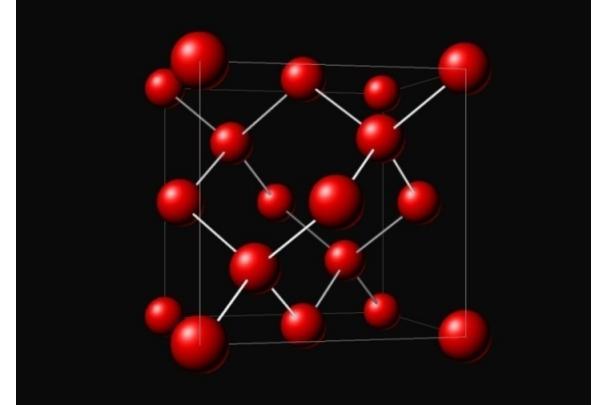
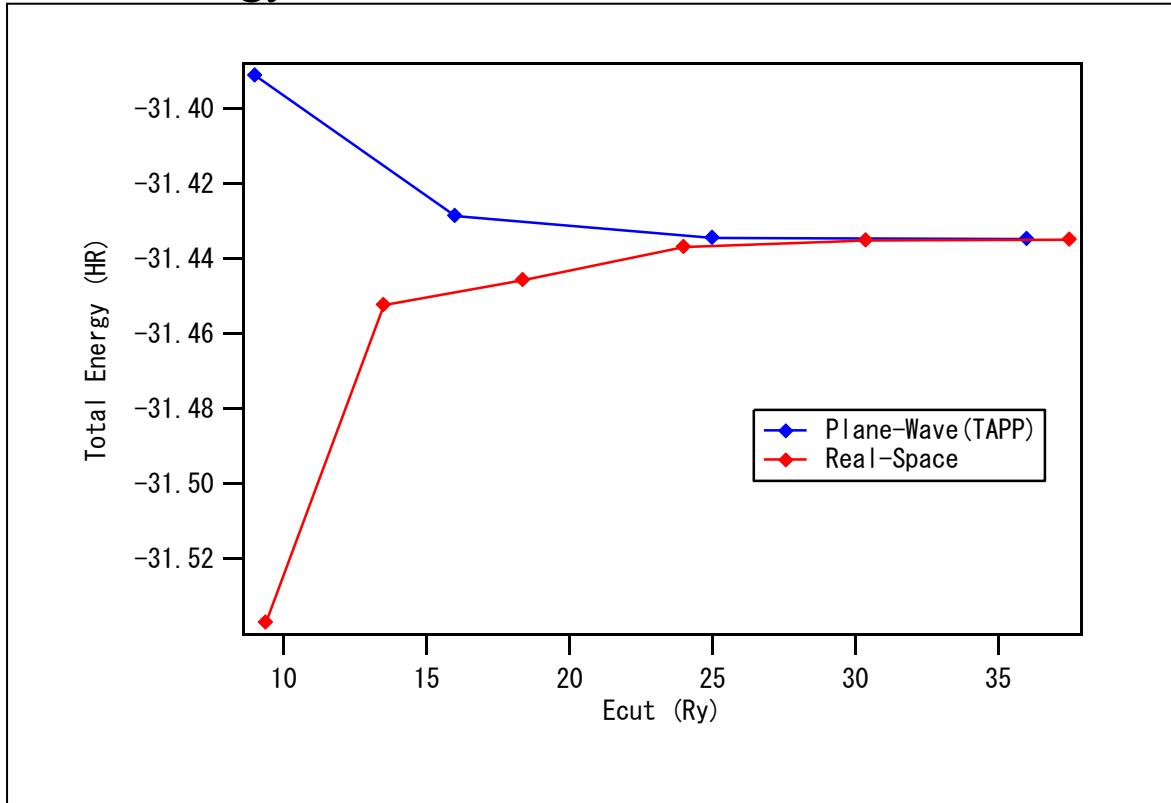
$$\frac{\partial^2}{\partial x^2} \phi(x, y, z) \approx \sum_{m=-M}^M C_m \phi(x + m\Delta x, y, z)$$

Total energy of 8-atom Si



Real-Space vs Plane-Wave Method

Total energy of 8-atom Si



Real-Space

Lattice constant 5.37 \AA

Bulk Modulus 0.977 (Mb)

Plane-Wave (TAPP)

Lattice constant 5.37 \AA

Bulk Modulus 0.972 (Mb)

Experiment

5.41 \AA

0.988 (Mb)

A pflops computer - *K* computer -



Total Compute Node (CPU)	: 88,128
Total CPU Core	: 705,024
Total Peak Performance	: 11.28 PFLOPS
Total Memory	: 1.34 PiB

Full node test @ K

Table 2. Distribution of computational costs for an iteration of the SCF calculation of the modified code.

Procedure block	Execution time (s)	Computation time (s)	Communication time (s)				Performance (PFLOPS/%)
			Adjacent/grids	Global/grids	Global/orbitals	Wait/orbitals	
SCF	2903.10	1993.89	61.73	823.02	12.57	11.89	5.48/51.67
SD	1796.97	1281.44	13.90	497.36	4.27	—	5.32/50.17
MatE/SD	525.33	363.18	13.90	143.98	4.27	—	6.15/57.93
EigenSolve/SD	492.56	240.66	—	251.90	—	—	0.01/1.03
RotV/SD	779.08	677.60	—	101.48	—	—	8.14/76.70
CG	159.97	43.28	47.83	68.85	0.01	—	0.06/0.60
GS	946.16	669.17	—	256.81	8.29	11.89	6.70/63.10

The test model was a SiNW with 107,292 atoms. The numbers of grids and orbitals were $576 \times 576 \times 180$, and 230,400, respectively. The numbers of parallel tasks in grids and orbitals were 27,648 and three, respectively, using 82,944 compute nodes. Each parallel task had 2160 grids and 76,800 orbitals.

Y. Hasegawa *et al.*, International Journal of High Performance Computing Applications, to be published

「京」時代のRSDFTによる第一原理電子状態計算の状況

10,000～数万原子規模の計算を何度も実行する
100,000原子規模の計算をやろうと思えばできる