
PHASE/0 Manual

リリース 2021.01

PHASE システム研究会

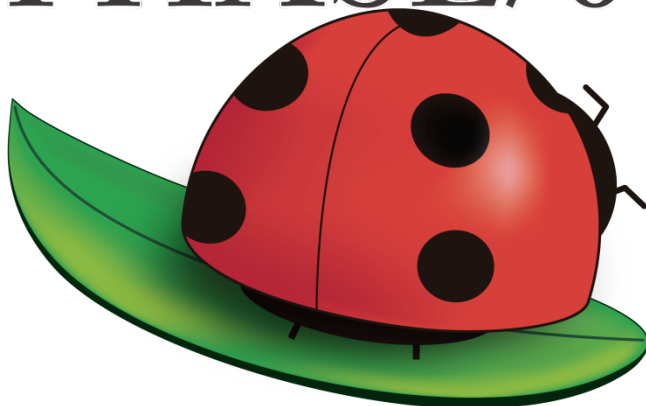
2021 年 09 月 06 日

Contents:

第 1 章	第一原理電子状態計算ソフトウェア PHASE/0 2021 ユーザーマニュアル	1
第 2 章	2021.01 版新機能	3
2.1	帯電欠損状態の評価	3
2.2	Methfessel-Paxton 法によるスメアリング	16
2.3	高精度な局所状態密度	18
2.4	limited-memory BFGS 法による構造最適化	20
2.5	高精度な XPS 計算	23
2.6	機械学習によるニューラルネットワークポテンシャルの作成	33
	関連図書	45

第1章 第一原理電子状態計算ソフトウェア PHASE/0 2021 ユーザーマニュアル

PHASE/0



12.0.01 (2021/08)

<https://azuma.nims.go.jp>

PHASE System

COPYRIGHT of the program codes

Copyright (C) of the original version: Hideki Katagiri, Koichi Kato, Tsuyoshi Miyazaki, Yoshitada Morikawa, Hideaki Sawada, Toshihiro Uchiyama, Tsuyoshi Uda, Takahiro Yamasaki.

Copyright (C) of the developed version by the national projects FSIS, RSS21, and RISS has been managed by the Institute of Industrial Science (IIS), the University of Tokyo.

The Institute of Industrial Science (IIS) has a right to distribute the program set developed from the original version as a free software.

HISTORY

The original version of this set of the computer programs "PHASE" was developed by the members of the Theory Group of Joint Research Center for Atom Technology (JRCAT), based in Tsukuba, in the period 1993-2001. The names of the contributors to the original version are Hideki Katagiri, K. Kato, T. Miyazaki, Y. Morikawa, H. Sawada, T. Uchiyama, T. Uda and T. Yamasaki. These contributors has agreed with that the Institute of Industrial Science (IIS), the University of Tokyo, distributes this program as a free software.

Since 2002, this program set had been intensively developed as a part of the following national projects supported by the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan; "Frontier Simulation

Software for Industrial Science (FSIS)" from 2002 to 2005, "Revolutionary Simulation Software (RSS21)" from 2006 to 2008. "Research and Development of Innovative Simulation Software (RISS)" from 2008 to 2013. These projects is lead by the Center for Research on Innovative Simulation Software (CISS), the Institute of Industrial Science (IIS), the University of Tokyo.

Since 2013, this program set has been further developed centering on PHASE System Consortium.

The activity of development of this program set has been supervised by Takahisa Ohno.

CONTACT ADDRESS

PHASE System Consortium

E-mail: phase_system@nims.go.jp URL <https://azuma.nims.go.jp>

* When distributing the software "PHASE" duplications, the user must attach the full text in this file.

第2章 2021.01 版新機能

2.1 帯電欠損状態の評価

2.1.1 概要

PHASE/0 では、電荷非中性状態を計算する際、周期的境界条件によるエネルギー発散を抑えるために一様な背景電荷を考慮して処理しています。この処理により、全エネルギーは周期的境界条件による背景電荷間の相互作用を含んだ値となります。このため、例えば、帯電した原子欠損などを含む半導体の生成エネルギーは、実際の値とは異なるものとなります。これを補正する手法として、近年、FNV 法 [Freysoldt09] が注目されており、対応するプログラム (`sxdefectalign`, `CoFFEE` など) がいくつか公開されています。PHASE/0 2021 年版以降、これらのプログラムに対するインターフェースが配備され、CUBE ファイル形式で出力することができるようになりました。また、これらのプログラムを利用せずに補正量を評価する仕組みも実装されています。さらに、異なる荷電状態間の生成エネルギーを比較した図を描画するための Python スクリプトが付属します。

2.1.2 理論

電荷 q をもつ欠陥 (D) の生成エネルギーは

$$E_{\text{form}}^{\text{D},q}(E_{\text{F}}) = E_{\text{DFT}}^{\text{D},q} - E_{\text{DFT}}^{\text{bulk}} - \sum_i n_i \mu_i + q(E_{\text{VBM}} + E_{\text{F}}) + E_{\text{corr}}^q \quad (2.1)$$

で書けます。 $E_{\text{DFT}}^{\text{D},q}$ および $E_{\text{DFT}}^{\text{bulk}}$ は DFT 計算における全エネルギー値で、前者は帯電欠陥、後者は非帯電・無欠陥の完全結晶 (Pristine) に対応します。 n_i は完全結晶からの元素 i の原子数の増減、 μ_i はその化学ポテンシャルです。 E_{VBM} は完全結晶の価電子帯上端のエネルギー、 E_{corr}^q は帯電量に依存する補正項です。

FNV 法

FNV 法では、 E_{corr}^q を 2 つの寄与に分けて考えます [Freysoldt09]。

$$E_{\text{corr}}^q = E_{\text{PC}}^q - q\Delta V_{\text{far}} \quad (2.2)$$

第 1 項は、孤立した点電荷 q の静電エネルギーと、一様な背景電荷を加えて周期的境界条件を課した際の静電エネルギーの差です。

$$E_{\text{PC}}^q = E_{\text{isolated}}^q - E_{\text{periodic}} \quad (2.3)$$

この値は、スーパーセルのサイズ・誘電率から、DFT 計算とは無関係に決定されます。一方、第 2 項は、potential alignment 補正項で、静電ポテンシャルの差

$$\Delta V(\mathbf{r}) = V_{\text{DFT}}^{D,q}(\mathbf{r}) - V_{\text{DFT}}^{\text{bulk}}(\mathbf{r}) - V_{\text{PC}}^q(\mathbf{r}) \quad (2.4)$$

を、欠陥から十分に離れた位置で評価した値です。ここで、 V_{PC}^q は、静電エネルギーに対応する静電ポテンシャルです。

外部解析プログラム

sxdefectalign、CoFFEE などの解析プログラムは、以下の入力を要求します。

- 結果の位置情報
- 誘電率の値
- $V_{\text{DFT}}^{D,q}(\mathbf{r})$ や $V_{\text{DFT}}^{\text{bulk}}$ の空間分布。たとえば CUBE 形式。

出力されるのは以下の情報です。

- E_{PC}^q の値
- $\Delta V(\mathbf{r})$ の値。各格子ベクトル方向に対するプロット。他の格子ベクトル方向については面平均をする。欠陥からの距離に対して、原子位置近傍で平均した値を出力することもある。

既知の問題として、以下が挙げられます。

- sxdefectalign では、欠陥からの距離に対するプロットが正常に出力されないことがある。
- CoFFEE では、各格子ベクトル方向に対するプロットも、ユーザーが python スクリプトを、都度書く必要がある。また、欠陥からの距離に対するプロットは得られない。
- CoFFEE では、 E_{PC}^q の値は、セルサイズを変えて複数計算し、fitting の結果として得る。このため、ユーザーの手間が多い。

extended FNV 法

Kumagai ら [Kumagai14] の extended FNV 法では、 E_{PC}^q は

$$E_{\text{PC}}^q = -\frac{q}{2} \left[\sum_{\mathbf{R}_i \neq 0} \frac{q}{\sqrt{|\varepsilon|}} \frac{\text{erfc}(\sqrt{\mathbf{R}_i \varepsilon^{-1} \mathbf{R}_i})}{\sqrt{\mathbf{R}_i \varepsilon^{-1} \mathbf{R}_i}} - \frac{\pi q}{\Omega \gamma^2} + \sum_{\mathbf{G}_i \neq 0} \frac{4\pi q}{\Omega} \frac{\exp(-\mathbf{G}_i \varepsilon \mathbf{G}_i / 4\gamma^2)}{\mathbf{G}_i \varepsilon \mathbf{G}_i} - \frac{2\gamma q}{\sqrt{\pi|\varepsilon|}} \right] \quad (2.5)$$

で表されます。ここで、 γ は適当に選んだ収束パラメータです。また、 ε は誘電率テンソル、 Ω は系の体積です。一方、ポテンシャル V_{PC}^q は、

$$V_{\text{PC}}^q = \sum_{\mathbf{R}_i \neq 0} \frac{q}{\sqrt{|\varepsilon|}} \frac{\text{erfc}(\sqrt{\mathbf{R}_i \varepsilon^{-1} \mathbf{R}_i})}{\sqrt{\mathbf{R}_i \varepsilon^{-1} \mathbf{R}_i}} - \frac{\pi q}{\Omega \gamma^2} + \sum_{\mathbf{G}_i \neq 0} \frac{4\pi q}{\Omega} \frac{\exp(-\mathbf{G}_i \varepsilon \mathbf{G}_i / 4\gamma^2)}{\mathbf{G}_i \varepsilon \mathbf{G}_i} \exp[i\mathbf{G}_i(\mathbf{r} - \mathbf{R}_d)] \quad (2.6)$$

で表されます。ここで、 \mathbf{R}_d は欠陥の位置です。

V_{far} の評価に関しては、以下の手順をとります。まず、各原子サイトにおける $\Delta V(\mathbf{r})$ を計算します。このうち、欠陥からの距離 R_{WS} より遠方にあるデータを平均し、 ΔV_{far} とします。なお、 R_{WS} は、欠陥間の最短距離の半分とします。(2.5), (2.6) による計算を用いることによって sxdefectalign, CoFFEE などを用いずとも (自動的に) E_{corr}^q の評価が可能です。

2.1.3 使い方

外部解析プログラムの使用

sxdefectalign や CoFFEE に受け渡す静電ポテンシャルを CUBE 形式で出力するには、入力パラメーター ファイルを以下のように設定します。

```
postprocessing{
  electrostatic_potential{
    sw_write_electrostatic_pot = on
    unit = Rydberg                ! { Rydberg | Hartree | eV }
  }
}
```

単位系は unit で指定し、Rydberg、Hartree、eV の選択が可能です。default 値は eV です。

結果は electrostatic_pot.cube ファイルに出力されます。ヘッダー部に単位系が表示され、以降、各軸方向の分割数、メッシュの刻み幅、各メッシュ点における値と続きます。

```
Calculated by phase
Local+Hartree potential in Rydberg
215      0.0000      0.0000      0.0000
180      0.180912      0.000000      0.000000
180      0.000000      0.180912      0.000000
180      0.000000      0.000000      0.180912
33 33.000000      2.396007      2.396052      2.396080
(後略)
```

帯電欠陥、及び完全結晶の静電ポテンシャルを出力しておくようにします (sxdefectalign を利用する場合には、unit は Rydberg にしておく必要があります。)。以下に、これらの CUBE ファイルを用いた sxdefectalign の実行例を示します。

```
sxdefectalign --charge 3 --eps 12.88
--vdef ../electrostatic_pot.cube --vref ../bulk/electrostatic_pot.cube
--center 0.0,0.0,0.0 --relative
--ecut 270 --qe > Log
```

系の帯電量を q としたとき、charge には $-q$ を渡します。eps には誘電率の値を指定します。vdef 及び vref には、帯電欠陥及びバルクにおける静電ポテンシャルファイルを、それぞれ指定します。center には欠陥位置の内部座標を与えます。ecut には nfnp.data で指定した cutoff_cd を Ry 単位で渡します。

sxdefectalign の Log ファイルの末尾には、以下のような出力がされます。

```
Defect correction (eV): 0.828347 (incl. screening & alignment)
```

ここで表示されているのは、値 (式 (2.3) 参照) であり、potential alignment の寄与は含まれていません。Log ファイル以外には、vline-eV-an.dat ($n=0, 1, 2$) 及び vAtoms.dat が出力されます。前者は、式 (2.4) の各結晶軸方向に対する、ポテンシャルの 1 次元プロファイルを出力したものです。一方、後者は、各原子の欠陥位置からの距離、及び原子上でのポテンシャルの値を出力したものです。ただし、vAtoms.dat は正常に出力されないことが多々あるようです。

extended FNV 法

まず帯電なし・欠陥なしの bulk の計算を行います。入力パラメーターファイルは以下のように設定します。

```
postprocessing{
  electrostatic_potential{
    sw_write_electrostatic_pot = on
  }
}
```

結果は `elecspot_bin.data` ファイルが出力されます。静電ポテンシャルの G 空間での値をバイナリ形式で出力しているため、外部ソフトと連携することはできません。ファイル名は、`F_ELECPOT_BIN` により変更できます。

```
&fnames
F_ELECPOT_BIN = './elecspot_bin.data'
/
```

ついで帯電欠陥の計算を行います。入力パラメーターファイルは以下のように設定します。

```
postprocessing{
  electrostatic_potential{
    sw_write_electrostatic_pot = on
  }
  charged_defect{
    correction{
      sw_calc_extfnv_correction = on          ! default : off
      dielectric_constant{
        exx = 12.88
        eyy = 12.88
        ezz = 12.88
      }
      position{
        x = 0.0, y = 0.0, z = 0.0
      }
    }
  }
}
```

`charged_defect` ブロック内の `correction` ブロックで、`sw_calc_extfnv_correction = on` を設定します。また、`dielectric_constat` ブロック内で誘電率テンソルの値、`position` ブロック内で欠陥の内部座標を指定します。欠陥以外の場合には、内部座標ではなく、原子番号 (`atom_id`) による指定も可能です (例: `atom_id= 2`)。さらに、参照する bulk の静電ポテンシャルファイル名を `F_ELECPOT_BIN_REF` で指定します。

```
&fnames
F_ELECPOT_BIN_REF = './bulk/elecspot_bin.data'
F_ELECPOT_BIN = './elecspot_bin.data'
/
```

結果は `defect_pot_correction.direction_n` ($n=1,2,3$)、`defect_pot_correction.atoms` に出力されます。前者は、式 (2.4) の各結晶軸方向に対する、ポテンシャルの 1 次元プロファイルを出力したものです。一方、後者は、各原子の欠陥位置からの距離、及び原子上でのポテンシャルの値を出力したものです。

以下に、`defect_pot_correction.direction_1` の出力例を示します。

```
# dist. (Ang), pot_diff, Vpc, pot_diff -Vpc (eV)
0.00000 -0.13205 -0.03907 -0.09299
0.09573 -0.13743 -0.03902 -0.09841
(後略)
```

第 2 列及び第 3 列は、それぞれ式 (2.4) の $V_{\text{DFT}}^{D,q}(\mathbf{r}) - V_{\text{DFT}}^{\text{bulk}}(\mathbf{r})$ 及び $V_{\text{PC}}^q(\mathbf{r})$ に対応します。第 4 列は $\Delta V(\mathbf{r})$ に対応します。なお、符号については、電子の電荷が負であることを考慮しています。

以下に、`defect_pot_correction.atoms` の出力例を示します。

# no.,	dist. (Ang),	pot_diff,	Vpc,	pot_diff -Vpc (eV)
1	2.1961	-0.0550653932	-0.2936706923	0.2386052991
2	3.9534	-0.1647261198	-0.1693351641	0.0046090443
(中略)				
# Correction energy (eV):		0.7827480523		

なお、最終行に表示されている “ Correction energy ” は、生成エネルギー図作成に必要な E_{corr}^q です。

生成エネルギー図の作成 1

欠陥ごとの生成エネルギー図の作成方法を説明します。生成エネルギー図作成には calc_defect_formation_energy.py スクリプトを用います。

平衡状態のとき、化学ポテンシャルは

$$\begin{aligned}\mu_{\text{Ga}} + \mu_{\text{As}} &= \mu_{\text{GaAs}}^0, \\ \mu_{\text{Ga}} &\leq \mu_{\text{Ga}}^0, \\ \mu_{\text{As}} &\leq \mu_{\text{As}}^0\end{aligned}\quad (2.7)$$

を満たします。ここで、 μ_{Ga}^0 及び μ_{As}^0 は、それぞれ Ga 及び As 単体の 1 原子あたりのエネルギーです。また、 μ_{GaAs}^0 は、GaAs 結晶の 2 原子あたりのエネルギーです。特に、Ga-rich 極限の場合は、

$$\mu_{\text{Ga}} = \mu_{\text{Ga}}^0, \mu_{\text{As}} = \mu_{\text{GaAs}}^0 - \mu_{\text{Ga}}^0 \quad (2.8)$$

となり、As-rich 極限では

$$\mu_{\text{Ga}} = \mu_{\text{GaAs}}^0 - \mu_{\text{As}}^0, \mu_{\text{As}} = \mu_{\text{As}}^0 \quad (2.9)$$

となります。これら化学ポテンシャルの情報は、以下で説明する calc_defect_formation_energy.py スクリプトの入力に記述する必要があるものです。

例えば、GaAs のスーパーセルの中に Ga 欠損を導入した計算を行い、帯電量 $q = -3, -2, -1, 0$ の計算結果が得られているとします。生成エネルギー図作成のために、以下のようなファイルを作成します。ファイル名に制限はありません。以下の例では tmp1.in とします。

```
&VEM          #(eV)
6.00887

&band_gap      #(eV)
1.424

&Chemical_potentials #(Ha)
Ga -138.9838873703      # mu_GaAs_bulk -mu_As
As -87.9848825305

&Defects        #elements, number ( negative==vacancy, positive==impurity )
Ga -1

&Host_supercell_energy #(Ha)
-7263.0006368270

&Defective_supercell_energy # charge_state(q) and energy (Ha)
-3 -7123.2688494462
-2 -7123.4909799361
-1 -7123.7111092854
0 -7123.9292947570

&Correction_energy # charge_state(q) and energy (eV)
-3 1.1593520601
-2 0.6178614391
-1 0.2315198082
0 0.0
```

以下、用語の説明をします。

表 2.1: キーワードの説明

キーワード	単位	意味
&VBM	eV	E_{VBM}
&band_gap	eV	バンドギャップ値
&Chemical_potentials	Hartree	化学ポテンシャル μ_i 。系を構成する全元素について記述する。
&Defects		導入した欠陥
&Host_supercell_energy	Hartree	$E_{\text{DFT}}^{\text{bulk}}$
&Defective_supercell_energy	Hartree	$E_{\text{DFT}}^{D,q}$. 計算した全荷電状態について q が小さい順に記述する。
&Correction	eV	E_{corr}^q . 計算した全荷電状態について q が小さい順に記述する。

以下の要領で calc_defect_formation_energy.py スクリプトを実行します。

```
python3 calc_defect_formation_energy.py input
[-o OUTFILE]
[--emin EMIN] [--emax EMAX] [--de DE]
[--vmin VMIN] [--vmax VMAX]
[--image_format IMAGE_FORMAT]
```

括弧内は省略可能なオプションで、その意味は以下のとおりです。

表 2.2: キーワードの説明

引数	意味	デフォルト値
-o	出力するファイルの名称	result
--emin	エネルギー EF の最小値	-1.0
--emax	エネルギー EF の最大値	6.0
--de	エネルギー EF の刻み幅	0.01
--vmin	生成エネルギーの表示範囲の最小値	-5.0
--ymax	生成エネルギーの表示範囲の最大値	5.0
--image_format	可視化画像の形式 (png/eps)	png

実行例を以下に示します。

```
python3 calc_defect_formation_energy.py tmp1.in -o result1
```

上述のコマンドを実行した結果、生成されるファイルは result1.qdep、result1.min、result1.gnu、result1.png です。以下に、result1.qdep 及び result1.min の一部を示します。前者には、各帯電状態における生成エネルギー

ギーの、フェルミエネルギー依存性が出力されています。後者は、各フェルミエネルギー値における最小の生成エネルギー値が出力され、ファイル末尾に Charge Transition Level が追記されています。

result1.qdep ファイルの内容

```
# Formation energy
# Ef (eV)    q=-3    q=-2    q=-1    q=0
-1.000000    6.48430    4.90715    3.53961    2.37978
(後略)
```

result1.min ファイルの内容

```
# Formation energy
# Ef (eV)    min
-1.000000    2.37978
(中略)
6.000000 -14.51570

# Charge    Transtion level [eV]
#-2/-3      0.57715
#-1/-2      0.36754
# 0/-1      0.15983
```

result1.gnu は可視化のための gnuplot 用ファイル、result1.png は gnuplot の出力です。

生成エネルギー図の作成 2

興味ある全ての欠陥構造について、[生成エネルギー図の作成 1](#) の作業が終了しているものとします。ここでは、これらをまとめた図の作成を行うため、以下のようなファイルを作成します。ファイル名に制限はありません。以下の例では gather1.in とします。

```
#
# title    filename ( excluding ".min" )
#
&List
Vac_Ga    Vacancy_Ga/result1
Vac_As    Vacancy_As/result1
Ga_As     Ga_for_As/resultaa
As_Ga     As_for_Ga/resultaa

&band_gap    #(eV)
1.424
```

以下、用語の説明をします。

表 2.3: キーワードの説明

ワード	単位	意味
&List		欠陥の名称、及び計算の出力 (生成エネルギー図の作成 1 指定した outfile 名)
&band_gap	eV	バンドギャップ値

以下の要領で plot_multiple_defect_formation_energy.py スクリプトを実行します。

```
python3 plot_multiple_defect_formation_energy.py input
[-o OUTFILE]
[--emin EMIN] [--emax EMAX]
[--vmin VMIN] [--vmax VMAX]
[--image_format IMAGE_FORMAT]
[--keypos_h KEYPOS_H] [--keypos_v KEYPOS_V]
```

括弧内は省略可能なオプションで、その意味は以下のとおりです。なお、EMIN、EMAX 値は、[生成エネルギー図の作成 1](#) の指定と揃えた方がよいです。

表 2.4: plot_multiple_defect_formation_energy.py で省略可能なオプション一覧

引数	意味	デフォルト値
-o	出力するファイルの名称	result_all
--emin	エネルギー EF の最小値	なし
--emax	エネルギー EF の最大値	なし
--vmin	生成エネルギーの表示範囲の最小値	なし
--ymax	生成エネルギーの表示範囲の最大値	なし
--image_format	可視化画像の形式 (png/eps)	png
--keypos_h	凡例の水平位置 (left/center/right)	right
--keypos_v	凡例の垂直位置 (top/center/bottom)	top

以下に実行例を示します。

```
python3 plot_multiple_defect_formation_energy.py gather1.in
```

実行すると result_all.gnu 及び result_all.png が生成されます。前者は gnuplot 用ファイルで、後者はこれを可視化したものです。

2.1.4 例題

概要

GaAs 64 原子をホストとして、帯電欠陥の計算を行った例を紹介します。計算条件は以下の通りです。なお、GaAs の格子定数は、基本格子で最適化を行いました。補正エネルギーは、PHASE/0 に実装されている extended FNV 法により評価しました。

表 2.5: 帯電欠陥の評価で使用した計算条件

平面波カットオフ [Ry]	30.0
電荷密度カットオフ [Ry]	270.0
k 点サンプリング	Monkhorst-Pack (2 × 2 × 2)
交換相関相互作用	GGAPBE, PAW
単位胞の 1 辺 [Å]	11.48882
SCF 収束条件	[Ha/atom] 1.0E-8
力の収束条件	[Ha/bohr] 2.0E-4
擬ポテンシャル	Ga_ggapbe_paw_02.pp, As_ggapbe_paw_02.pp
(補正項の計算で使用する)	誘電率 12.88

<https://www.microwaves101.com/encyclopedias/gallium-arsenide>

例題のディレクトリー構成および計算のながれ

本例題は複数の計算を行い、その結果をとりまとめ、スクリプトで処理することによって結果が得られる仕組みになっています。ここではディレクトリーの構成と計算のながれについて説明します。

必要な入力ファイルはサンプルディレクトリーの下で defectq/GaAs 以下のサブディレクトリーに配置されています。以下のようなディレクトリー構成になっています。

defectq/GaAs 以下には Preparation ディレクトリーと GaAs_64_lattice_opt ディレクトリーが存在します。

Preparation ディレクトリー

単体の As, Ga, GaAs の格子最適化の入力ファイルが納められたディレクトリーです。bulk_As, bulk_Ga, bulk_GaAs ディレクトリーがあり、格子最適化の入力ファイルが納められています。結果得られるエネルギーと (2.8) (2.9) から化学ポテンシャルをもとめ、スクリプト入力ファイルの &Chemical_potentials に記述します。

GaAs_64_lattice_opt ディレクトリー

各種生成エネルギー計算の入力ファイルが格納されています。ベースとなる結晶は格子最適化によってもとまった格子定数から作成した 64 原子系です。GaAs_64_lattice_opt ディレクトリーは、さらに以下のようなサブディレクトリー群が存在します。

表 2.6: GaAs_64_lattice_opt 以下のディレクトリー構成

ディレクトリー名	説明
Pristine	欠損のない結晶の入力ファイルが納められたディレクトリー。
Ga_for_As	As を Ga で置換した欠陥構造の入力ファイルが納められたディレクトリー。
As_for_Ga	Ga を As で置換した欠陥構造の入力ファイルが納められたディレクトリー。
As_interstitial_As4	As が最近接原子となる位置に入り込んだ As interstitial の入力ファイルが納められたディレクトリー。
As_interstitial_Ga4	Ga が最近接原子となる位置に入り込んだ As interstitial の入力ファイルが納められたディレクトリー。
Ga_interstitial_As4	As が最近接原子となる位置に入り込んだ Ga interstitial の入力ファイルが納められたディレクトリー。
Ga_interstitial_Ga4	Ga が最近接原子となる位置に入り込んだ Ga interstitial の入力ファイルが納められたディレクトリー。

それぞれのディレクトリーにはさらに q_ q ディレクトリーが存在します。ここで q は電荷をあらわす数値です。

- Pristine の場合は q_0 と q_0.2 ディレクトリーが存在します。 $E_{\text{VBM}} = \frac{E(N) - E(N - \Delta N)}{\Delta N}$ (今の場合 $\Delta N = 0.2$) という関係から E_{VBM} をもとめ、スクリプト入力の &VBM に記述します。また q_0 のエネルギーをスクリプト入力の &Host_supercell_energy に記述します。
- ほかの欠損に対応するディレクトリーでは、エネルギーの計算結果をスクリプト入力の &Defective_supercell_energy に記述します。また、q_0 以外の計算では extended FNV 法による補正エネルギー E_{corr}^q の計算がなされます。結果は defect_pot_correction.atoms ファ

イルの末尾に記録されるので、その計算値をスクリプト入力の&Correction に記述します (q=0 の項には 0 を記述します)。

各欠損ディレクトリーにおいて calc_defect_formation_energy.py スクリプトの入力を作成し、実行することによって各欠損の生成エネルギー図を作成することができます。またすべての欠陥のデータを集約し [生成エネルギー図の作成 2](#) の手続きをふむことによってすべての欠陥の結果をまとめた生成エネルギー図を作成することができます。

補正エネルギー比較

以下に、Ga 欠損 (q=-3) における補正エネルギーについて、sxdefectalign と PHASE/0 実装の extentend FNV 法による評価の比較を示します。両者がおおよそ一致していることが分かります。

表 2.7: Ga 欠損 (q=-3) における補正エネルギーの内訳 (単位 : eV)

	sxdefectalign	PHASE/0
Epc	1.24251	1.24272
dV	-0.00731	-0.02779
Ecorr	1.22058	1.15935

生成エネルギー図の作成 1.

以下では、Ga 欠損に対して、2 つの極限における入力を作成し可視化します。以降、Ga-rich 極限及び As-rich 極限の入力を示します。

calc_defect_formation_energy.py 用の入力 (Ga-rich 極限における Ga 欠損; サンプルディレクトリーの defectq/GaAs_64_lattice_opt/Vacancy_Ga/cond_Ga-rich.in)

```
&VBM          #(eV)
6.00887

&band_gap     #(eV)
1.424

&Chemical_potentials #(Ha)
Ga -138.9586691142      #mu_Ga
As -88.0101007866      #mu_GaAs -mu_Ga

&Defects      #elements, number ( negative==vacancy, positive==impurity )
Ga -1

&Host_supercell_energy #(Ha)
-7263.0006368270

&Defective_supercell_energy # charge_state(q) and energy (Ha)
-3 -7123.2688494462
-2 -7123.4909799361
-1 -7123.7111092854
0 -7123.9292947570

&Correction energy # charge_state(q) and energy (eV)
-3 1.1593520601
-2 0.6178614391
-1 0.2315198082
0 0.0
```

calc_defect_formation_energy.py 用の入力 (As-rich 極限における Ga 欠損; サンプルディレクトリーの defectq/GaAs_64_lattice_opt/Vacancy_Ga/cond_As-rich.in)


```

&VBM          #(eV)
6.00887

&band_gap      #(eV)
1.424

&Chemical_potentials #(Ha)
Ga  -138.9838873703  # mu_GaAs_bulk -mu_As
As   -87.9848825305   #mu_As

&Defects        #elements, number ( negative==vacancy, positive==impurity )
Ga  -1

&Host_supercell_energy #(Ha)
-7263.0006368270

&Defective_supercell_energy # charge_state(q) and energy (Ha)
-3 -7123.2688494462
-2 -7123.4909799361
-1 -7123.7111092854
0  -7123.9292947570

&Correction_energy # charge_state(q) and energy (eV)
-3 1.1593520601
-2 0.6178614391
-1 0.2315198082
0 0.0

```

つぎのコマンドを実行すると、生成エネルギーの図 (result-Ga-rich.png 及び result-As-rich.png) が出来ます。結果を 図 2.1, 図 2.2 に示します。

calc_defect_formation_energy.py の実行 (Ga 欠損)

```

python3 calc_defect_formation_energy.py
cond_Ga_rich.in -o result_Ga_rich --emin -0.5 --emax 2.0

python3 calc_defect_formation_energy.py
cond_As_rich.in -o result_As_rich --emin -0.5 --emax 2.0

```

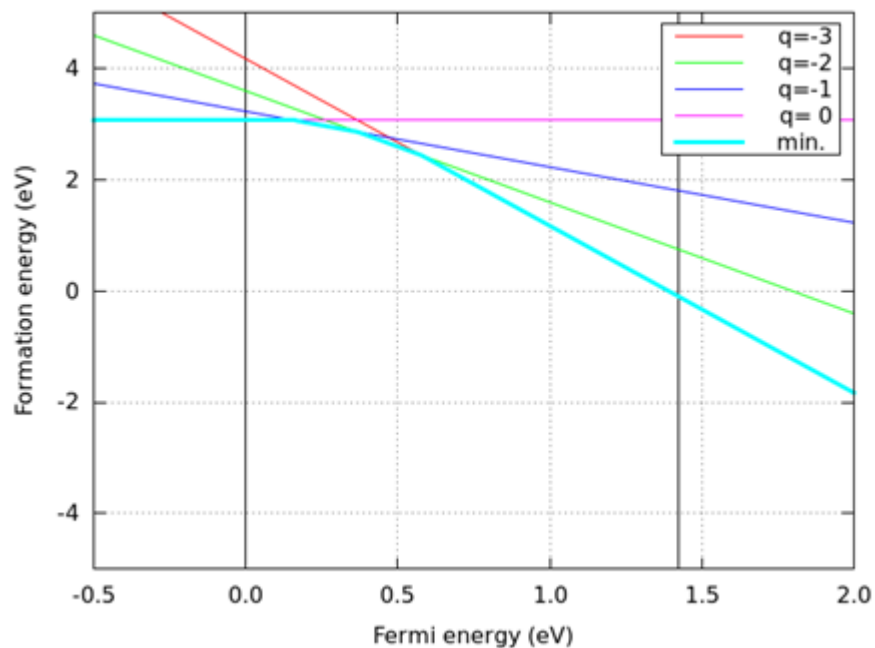


図 2.1: Ga-rich limit (ファイル名: result-Ga-rich.png)

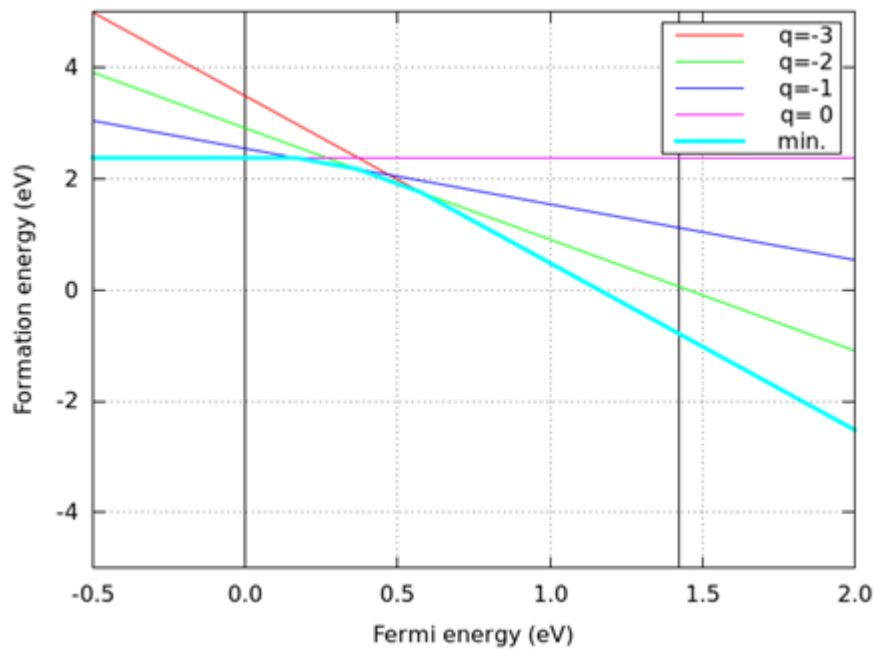


図 2.2: As-rich limit (ファイル名 : result-As-rich.png)

生成エネルギー図の作成 2.

以下で、種々の欠陥構造に対する生成エネルギーをまとめた図を作成します。Ga-rich 及び As-rich 極限のファイルが、それぞれ、defectq/GaAs_64_lattice_opt の下の gather_Ga-rich.in 及び gather_As-rich.in にあります。後者は、つぎに紹介する入力の result-Ga-rich を result-As-rich に置換したものです。なお、欠陥の種類としては、文献 [Broberg18] 記載のものを選択しました。

gather_Ga-rich.in の内容

```
#
# title  filename ( excluding ".min" )
#
&List
Vac_Ga      Vacancy_Ga/result-Ga-rich
Vac_As      Vacancy_As/result-Ga-rich
Ga_As       Ga_for_As/result-Ga-rich
As_Ga       As_for_Ga/result-Ga-rich
Ga_i_As4    Ga_interstitial_As4/result-Ga-rich
As_i_As4    As_interstitial_As4/result-Ga-rich
Ga_i_Ga4    Ga_interstitial_Ga4/result-Ga-rich
As_i_Ga4    As_interstitial_Ga4/result-Ga-rich

&band_gap      #(eV)
1.424
```

つぎのコマンドを実行すると、生成エネルギーの図 (results-Ga-rich.png 及び results-As-rich.png) が出来ます。結果を図 2.3 および 図 2.4 に示します。

```
python3 plot_multiple_defect_formation_energy.py
gather_Ga-rich.in -o results_Ga-rich --emin -0.2 --emax 2.0 --vmin -4.0 --vmax 5.0
--keypos_h left --keypos_v bottom
python3 plot_multiple_defect_formation_energy.py
gather_As-rich.in -o results_As-rich --emin -0.2 --emax 2.0 --vmin -4.0 --vmax 5.0
--keypos_h left --keypos_v bottom
```

文献 [Broberg18] と同様の生成エネルギー図が得られました (生成エネルギーの絶対値や Charge Transition Level は、完全には一致しません)。

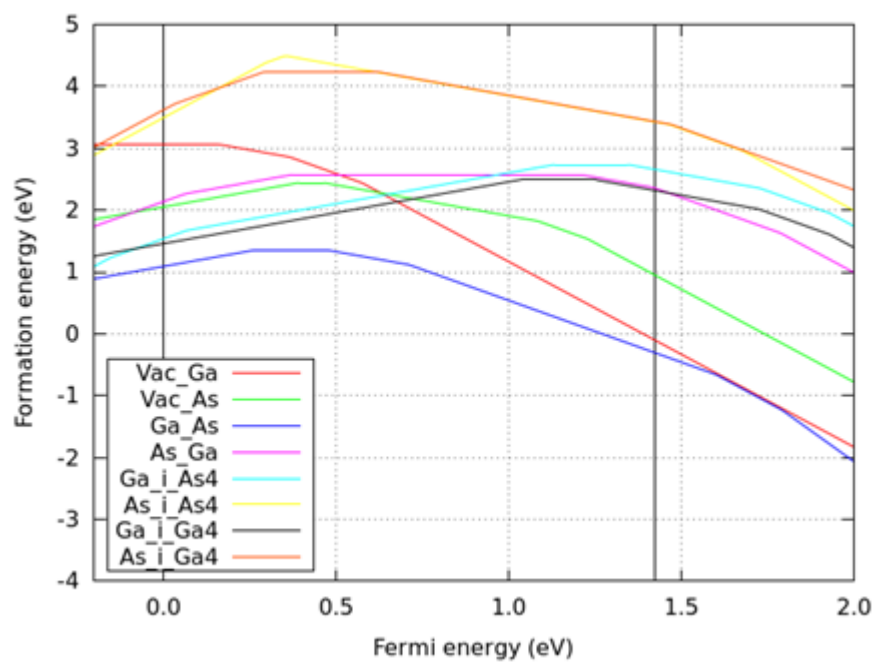


図 2.3: Ga-rich limit (ファイル名 : results-Ga-rich.png)

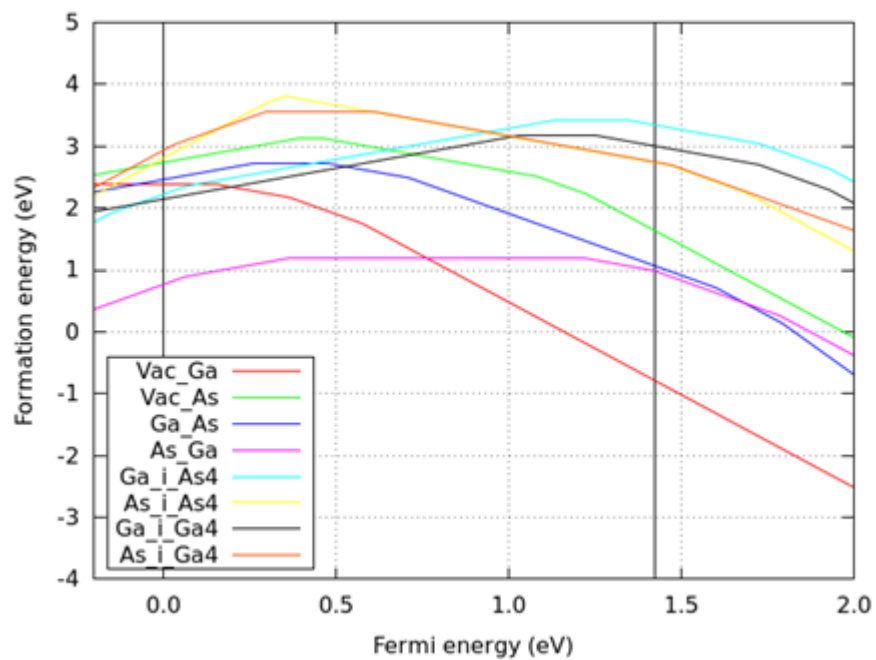


図 2.4: As-rich limit (ファイル名 : results-As-rich.png)

2.2 Methfessel-Paxton 法によるスミアリング

2.2.1 概要

PHASE/0 には parabolic 法, 四面体法などのスミアリング手法が備わっています。バージョン 2021 以降, これに Methfessel-Paxton 法 [Methfessel89] が加わりました。Methfessel-Paxton 法は, 負の占有数を許容するかわりに大きなスミアリング幅を利用しても全エネルギーなどの重要な計算結果がほとんど変化しない, という特徴を持ったスミアリング手法です。

2.2.2 Methfessel-Paxton 法について

基本の理論

スミアリング手法とは, δ 関数および階段関数を数値的に安定でなめらかな関数で近似する手法ということです。Methfessel-Paxton 法においては, δ 関数および階段関数を以下のように近似します。

$$D(x) = \sum_{n=0}^N [A_n H_{2n} e^{-x^2}]$$

$$S(x) = \frac{1}{2} (1 - \operatorname{erf}(x)) \quad (2.10)$$

$A_n = \frac{(-1)^n}{n! 4^2 \sqrt{\pi}}$, H_n は次数 n のエルミート多項式です。実際の計算においては, 準位 ik における固有値を ε_{ik} フェルミエネルギー ε_F スミアリング幅を σ とすると x は $x_{ik} = \frac{(\varepsilon_{ik} - \varepsilon_F)}{\sigma}$ という形で用いられます。また, エントロピー項は以下のように評価することができます [Kresse96]

$$S_N = \sigma \sum_{ik} \frac{1}{2} A_N H_{2N}(x_{ik}) e^{-x_{ik}^2} \quad (2.11)$$

図 2.5 に, $N=6$ までの $D(x), S(x)$ を示します。

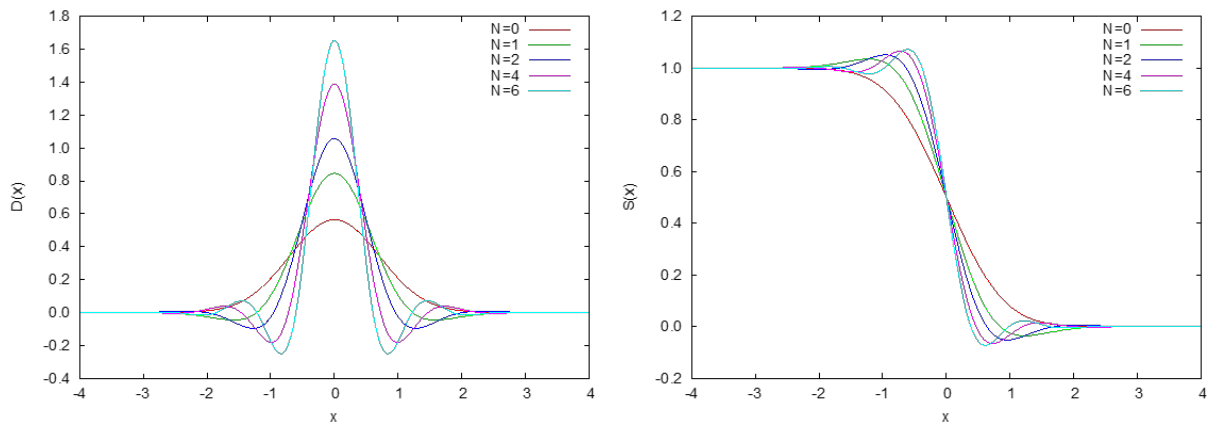


図 2.5: $N=6$ までの $D(x)$ および $S(x)$

図 2.5 から明らかなように, Methfessel-Paxton 法の場合フェルミエネルギー付近 ($x=0$ 付近) の電子の占有数が負になる場合があります。これは明らかに物理的ではありませんが, 最終的な全エネルギーの計算値などに影響を及ぼすことはありません。

フェルミエネルギーについて

通常フェルミエネルギーは電荷の数が合うように二分法によって求めますが、Methfessel-Paxton 法においては負の占有状態が発生するため通常の二分法ではフェルミエネルギーが一意に求まらない可能性があります。そこで、最低エネルギーから少しずつエネルギーを増やし、電子数を超えるエネルギーを見つけてから通常の二分法に移行する、という特殊な手続きでフェルミエネルギーを求めます。具体的な手続きは下記の通り。

1. スメアリングに利用するスメアリング幅と Methfessel-Paxton 法の次数からエネルギー幅 δ を設定します。 $\delta = a\sigma/N$. a は入力パラメーターファイルで設定可能な調整値で、PHASE/0 のデフォルト値は 10 です。
2. 探索を開始する最低エネルギー値を設定します。求まっている固有値の最低値でもよいですが、保守的すぎるので $e_{\min} + (e_{\max} - e_{\min})/2n$ という値を採用します。
3. δ を最低エネルギーに加算していきフェルミエネルギーとし、結果得られる仮の電荷が本来の電荷を上回った段階でそのエネルギーを上限值とします。
4. 以上の処理によって下限と上限がエネルギー幅の範囲で決定するので、ここからは通常の二分法に移行し、フェルミエネルギーを求めます。

2.2.3 使い方

スメアリングの設定は、入力パラメーターファイルの accuracy ブロックの下に smearing ブロックにおいて行います。Methfessel-Paxton 法の設定例を以下に記述します。

```
accuracy{
  smearing{
    method = methfessel_paxton
    width = 0.01 hartree
    methfessel_paxton{
      order = 2
    }
  }
}
```

method に methfessel_paxton を指定すると (meth でも可) スメアリング手法が Methfessel-Paxton 法になります。width においてスメアリング幅をエネルギーの単位で指定します。Methfessel-Paxton 法の詳細設定は methfessel_paxton ブロック (もしくは meth ブロック) において行います。ここでは以下のような設定を施すことができます。

表 2.8: Methfessel-Paxton 法の詳細設定

タグ名	説明
order	Methfessel-Paxton 法の次数を指定します。デフォルト値は 2 です。
esearch	フェルミエネルギーの探索を、通常の二分法ではなく特殊なエネルギー探索手法で行うかどうかを指定する真偽値です。デフォルト値は on.
esearch_factor	esearch が on の場合のエネルギー刻みのファクター。デフォルト値は 10.

2.2.4 例題

サンプルディレクトリーの smearing 以下に例題が配置されています。FCC-Al と BCC-Fe の例題が格納されています。それぞれの結晶について、mp2 mp4 mp6 parabolic の 3 つのディレクトリーがあり、それぞれ $N=4$ および 6 の場合の Methfessel-Paxton 法を適用した入力例と従来の parabolic 法を適用した例題に対応します。さらにその下のサブディレクトリーには、sigma0.0001(スミアリング幅を 0.0001 hartree にした場合) と sigma0.001(スミアリング幅を 0.001 hartree にした場合) の例題が配置されています。これらの例題から得られる全エネルギーの計算結果を次の表にまとめます (表中のエネルギーはすべて hartree 単位)

表 2.9: Methfessel-Paxton 法および従来法によって得られる全エネルギー。

	$\sigma = 0.0001 \text{ hartree}$	$\sigma = 0.001 \text{ hartree}$
Al, mp2	-0.2552794	-0.2552805
Al, mp4	-0.2552794	-0.2552801
Al, mp6	-0.2552794	-0.2552801
Al, parabolic	-0.2552797	-0.2552832
Fe, mp2	-21.993246	-21.993248
Fe, mp4	-21.993246	-21.993248
Fe, mp6	-21.993246	-21.993247
Fe, parabolic	-21.993247	-21.993259

表の数値から、Methfessel-Paxton 法は parabolic 法と比較して全エネルギーがスミアリング幅に依存しにくいことが分かります。

2.3 高精度な局所状態密度

2.3.1 概要

原子分割局所状態密度とは、空間を原子ごとの領域に分割し、その領域における電子状態密度をもとめる計算機能です。積算状態密度のフェルミエネルギーにおける値を調べることによって、対象原子の電子数をもとめるという用途に用いることも可能です。

原子分割局所状態密度は、波動関数および電荷密度の実空間における FFT メッシュを空間上で最も近い原子に割り当て、その寄与分を加算することによって計算します。このような方法の場合、実空間の FFT メッシュと原子位置の関係によって対称性から等価な原子間でも割り当たるメッシュ数などが異なり、結果が微妙に異なる場合があります。言い換えると、原子配置を単位胞に対してどのように定義するかによって計算結果が変化します。たとえば、対称性から等価なはずの原子の電子数が互いに異なる値となってしまう場合があります。この振る舞いを改善するため、2021 年版以降電荷密度の割り当てを実空間 FFT メッシュではなく原子中心メッシュに切り替えて評価する方法を用いることができます。このようにするとメッシュは常に同じように各原子に割り当てられるため、原子配置を単位胞に対してどう定義するかによって結果が左右されなくなります。原子中心メッシュは原子ごとに定義され、その値は実空間 FFT メッシュ上の値の三次元線形補間によって求められます。

2.3.2 使い方

原子中心メッシュによる局所状態密度計算の指定は、postprocessing ブロックにおける ldos ブロックにおいて行います。以下のような指定を行います。

```
postprocessing{
  dos{
    sw_dos = on
    method = t
  }
  ldos{
    sw_aldos = on
    aldoses{
      sw_atom_centered_mesh = on
      atom_centered_mesh_factor = 1
    }
  }
}
```

postprocessing ブロックに dos ブロックを作ると、全状態密度の設定を行うことができます。局所状態密度の計算の基本設定は全状態にならう形式になっているので、ここでの設定は局所状態密度にもあてはまります。ldos ブロックにおいて局所状態密度計算の設定を、さらにその下の aldoses ブロックにおいて原子分割局所状態密度計算の設定を行います。aldoses ブロックにおいて新たに利用できるようになった設定項目は下記の通り。

- sw_atom_centered_mesh もしくは sw_ac_mesh : on とすることによって、原子分割局所状態密度計算のメッシュが実空間 FFT メッシュから原子中心メッシュに切り替わります。デフォルト値は off.
- atom_centered_mesh_factor もしくは ac_mesh_factor : 原子中心メッシュはデフォルトの振る舞いではその“濃さ”は実空間 FFT メッシュと同じですが、ここで指定する係数分増やすことも可能です。たとえば 2 とすると、3 方向のメッシュ数がそれぞれ 2 倍となります。その結果計算量は 8 倍となる点は注意が必要です。デフォルト値は 1.

2.3.3 例題

Si 結晶およびグラファイトの例題がサンプルディレクトリーの dos_band/aldoses_by_acmesh 以下にあります。Si 結晶の例題はディレクトリー Si2 以下、グラファイトの例題はディレクトリー graphite 以下に配置されており、それぞれ acmesh と fftmesh フォルダが存在し、前者が原子中心メッシュを用いた局所状態密度計算、後者が FFT メッシュを用いた局所状態密度計算の入力ファイルが格納されています。

いずれの結晶も、精度の高い結果が得られるよう比較的大きなカットオフエネルギー (80 Rydberg) を採用しています。また、精密な状態密度を得るため四面体法を用いる設定を施しています。すなわち、入力パラメーターファイルには以下のような設定が施されています。

```
accuracy{
  cutoff_wf = 80 rydberg
  ...
  ksampling{
    method = mesh
    ...
  }
  smearing{
    method = t
  }
}
```

それぞれの最下層のディレクトリーにおいて計算を実行すると、SCF 計算のあと状態密度の計算が行われ、結果が dos.data ファイルに記録されます。原子分割局所状態密度の計算結果は ALDOS num_atom = aid という文字列からはじまる行以降に記録されます。ここで aid は原子の ID で、各原子の入力ファイルにお

ける定義順に対応します。各原子の電子数はフェルミエネルギーにおける積算状態密度の値です。dos.data ファイルは (スピンを考慮していない場合) 4 カラム目のデータがフェルミエネルギーを原点としたエネルギー, 6 カラム目が積算状態密度に対応するので, 4 カラム目が 0 となる行の 6 カラム目の値がその原子の電子数に対応することになります。たとえば以下のようなデータが dos.data に記録されている場合, 1 番目の原子の電子数はおおよそ 3.9803 となります。

```

...
...
ALDOS      num_atom =      1
No.    E(hr.)    dos(hr.)    E(eV)    dos(eV)    sum
   6    -0.33730    0.0000000000    -11.955933    0.0000000000    0.0000000000
  16    -0.33630    0.0000000000    -11.928722    0.0000000000    0.0000000000
...
...
4366    0.09870    0.6019928210    -0.091764    0.0221228201    3.9797782305
4376    0.09970    0.2103606325    -0.064553    0.0077306078    3.9801773503
4386    0.10070    0.0555466368    -0.037342    0.0020413005    3.9802911964
4396    0.10170    0.0056133904    -0.010130    0.0002062882    3.9803167566
4406    0.10270    0.0000000000    0.017081    0.0000000000    3.9803179066
4416    0.10370    0.0000000000    0.044293    0.0000000000    3.9803179066
...
...

```

以下, この例題によって得られる Si 結晶およびグラファイトのある原子の電子数を報告します。

表 2.10: 本例題によって得られる電子数

Si, fftmesh	Si, acmesh	C, fftmesh	C, acmesh
3.9803	4.0000	4.0537	3.9992

いずれの例題もすべての原子は対称性から等価なため, 電子数としては 4 という値が得られるはずです。表の値から明らかなように, 原子中心メッシュとともに FFT メッシュの結果よりも正しい解に近い結果が得られています。

2.4 limited-memory BFGS 法による構造最適化

2.4.1 概要

PHASE/0 には構造最適化手法として quench 法, CG 法, GDIIS 法, BFGS 法, FIRE 法などが搭載されています。バージョン 2021.01 以降, 文献 [Hjorth17] の limited-memory BFGS 法 (l-BFGS 法) が選択肢に加わりました。通常の l-BFGS 法のほか, 文献において提案されている前処理を施すこともできるようになっています。

2.4.2 l-BFGS 法について

limited-memory BFGS 法とは, BFGS 法を効率化した最適化手法です。ヘッセ行列をメモリーに明示的に持つのではなく, iteration ごとに構築していくことに特徴を持つ手法です。ヘッセ行列をあらわに持たないため, メモリー要求が小さいことが名称の limited memory の由来です。そのアルゴリズムの擬コードは,

以下のように記述することができます。

$$\begin{aligned}
 s_{k-1} &= x_k - x_{k-1} \\
 y_{k-1} &= g_k - g_{k-1} \\
 \rho_k &= \frac{1}{s_k y_k} \\
 q &= g_k \\
 \text{do } i &= k-1, k-2, \dots, k-m \\
 \alpha_i &= \rho_i s_i q \\
 q &= q - \alpha_i y_i \\
 \text{enddo} \\
 \gamma_k &= \frac{s_{k-1} y_{k-1}}{y_{k-1} y_{k-1}} \\
 H_k^0 &= \gamma_k I \\
 z &= q H_k^0 \\
 \text{do } i &= k-m, k-m+1, \dots, k-1 \\
 \beta_i &= \rho_i y_i z \\
 z &= z + s_i (\alpha_i - \beta_i) \\
 \text{enddo} \\
 x_{k+1} &= x_k - \alpha_k z
 \end{aligned} \tag{2.12}$$

ここで x_k, g_k 回目の iteration における座標データとエネルギーの座標微分, H_k^0 は初期ヘッセ行列の推定値に相当する行列です。その行列要素は (2.12) のように $H_k^0 = \gamma_k I$ と計算することが一般的ですが, これを以下のような前処理行列で置き換えることが文献では提案されています。

$$\begin{aligned}
 P_{ij} &= -\mu \exp \left(-A \left(\frac{r_{ij}}{r_{nn}} - 1 \right) \right), r_{ij} < r_{\text{cut}}, \\
 &0, r_{ij} > r_{\text{cut}}
 \end{aligned} \tag{2.13}$$

ここで $A, \mu, r_{nn}, r_{\text{cut}}$ は任意に設定できるパラメーターです。文献によると A はあまり結果を左右しないようです。文献でも採用されている 3 という値をデフォルト値としています。 r_{nn} は最近接間距離の最大値であり, 原子配置から自動的に決まる値です。 r_{cut} はカットオフ距離に相当します。典型的には r_{nn} の 2 倍という値にります。最も結果を左右するパラメーターが μ です。このパラメーターは小さい方が速い収束が見込めますが, 小さくし過ぎるとロバスト性が損なわれてしまいます。PHASE/0 のデフォルトの振る舞いとしては, 前処理行列の行列要素の大きさが結果としておおむね通常の l-BFGS 法の γ_k と同程度の値になるような値が採用されます。

2.4.3 使い方

構造最適化の設定は, 入力パラメーターファイルの structure_evolution ブロックにおいて行います。l-BFGS 法を用いる場合, 典型的には下記のような指定になります。

```

structure_evolution{
  method = lbfgs
  lbfgs{
    c_iteration2gdiis=3
    c_forc2gdiis = 0.05
    gdiis_box_size = 6
  }
}

```

(次のページに続く)

(前のページからの続き)

```

    initial_method = cg2
    maxstep = 0.1
  }
}

```

タグ `method` に `lbfgs` を指定すると l-BFGS 法を利用することができます。Limited-memory BFGS 法 (および通常の BFGS 法, GDIIS 法) の詳細設定は, `structure_evolution` ブロックの下の `lbfgs` ブロック (`bfgs`, `gdiis` も可) において行うことができます。lbfgs ブロック (もしくは `bfgs`, `gdiis` ブロック) において設定できる主なパラメーターは下記の通り。

表 2.11: l-BFGS 法の詳細設定

タグ名	説明
<code>c_iteration2gdiis</code>	lbfgs 法, bfgs 法, gdiis 法は計算の最初期から適用されるわけではなく, はじめは閾値を満たすまでは別の最適化手法が用いられます。閾値を満たしたあと, このタグの指定の回数経てから lbfgs 法 (もしくは bfgs 法, gdiis 法) に移行します。デフォルト値は 3.
<code>c_forc2gdiis</code>	lbfgs 法, bfgs 法, gdiis 法へ移行する際の閾値を力の単位で指定します。デフォルト値は 0.05 hartree/bohr
<code>gdiis_box_size</code>	lbfgs 法, bfgs 法, gdiis 法の履歴の大きさ。デフォルト値は 6.
<code>initial_method</code>	lbfgs 法, bfgs 法, gdiis 法へ移行する前の構造最適化手法を指定します。デフォルト値は <code>cg2</code> .
<code>maxstep</code>	lbfgs 法において, 最適化 1 回で原子が動ける距離の最大値を長さの単位で指定します。デフォルト値は 0.1 bohr. この設定値は lbfgs 法の場合のみ利用できます。

多くのタグ名が GDIIS 法にちなんだものとなっているのは, 本機能が GDIIS 法の履歴活用手法を用いているためです。

Limited-memory BFGS 法は, 上述の通り前処理行列を作用させることができます。その設定は, 下記の要領で行います。

```

structure_evolution{
  method = lbfgs
  ...
  sw_prec = on
  prec{
    A = 3.0
    mu = -1
  }
}

```

`structure_evolution` ブロックにおいて `sw_prec = on` とすると前処理が有効になります。前処理の詳細設定は `prec` ブロックにおいて行います。設定できるパラメーターは下記の通り。

表 2.12: l-BFGS 法の詳細設定

タグ名	説明
A	(2.13) のパラメーター A . デフォルト値は 3.
mu	(2.13) のパラメーター μ . 負の値を指定すると、履歴に応じた最適値が自動的に計算されます。正の値の場合はその値がそのまま利用されます。デフォルト値は-1.

2.4.4 例題

PHASE/0 の構造最適化の例題は、サンプルディレクトリーの structural_evolution の下に用意されています。すべての例題について quench, bfgs, cg2, gdiis の例題が用意されていますが、ここに lbfgs ディレクトリーを追加しました。PHASE/0 付属の例題を用いて、CG2, BFGS, l-BFGS 法が収束するまでに要した ionic iteration の回数を以下に報告します。

表 2.13: l-BFGS 法の詳細設定

系	CG2	BFGS	l-BFGS
Si(001) 面	84	102	63
SiO ₂ 結晶	15	15	11
TiO ₂ 結晶	21	31	17
ジクロロシクロヘキサン	77	60	45

この表から、おおむね l-BFGS 法が高速であることが分かります。必ずよい結果が得られるとは限りませんが、多くの問題でここで見たような傾向が見られており、l-BFGS 法は有力なオプションであると言えます。

2.5 高精度な XPS 計算

2.5.1 概要

2020 年版までの PHASE/0 では、XPS・XANES のピーク位置が実験値と異なることがありました。原因として、内殻電子の相対論の効果、スピン分裂効果が考えられます。前者については、CIAO を改変し相対論的效果を含む (DIRAC 型の) 運動エネルギーを出力可能なようにしました。後者については、PHASE/0 の opencore 法を改変し、その効果を取り込めるようにしました。これによって、SiC、AlN、Li₇Ti₅O₁₂ 結晶の内殻軌道の結合エネルギーについて、従来法に比べて実験値との差が小さくなることなどを確認しています。

2.5.2 理論

相対論的運動エネルギー

Kohn-Sham 方程式は、

$$(T + V) \psi_i = \varepsilon_i \psi_i \quad (2.14)$$

で表されます。 T 及び V はそれぞれ、運動エネルギー及びポテンシャル演算子、また ε_i 及び ψ_i は固有値・固有波動関数です。ここで、運動エネルギー演算子は、相対論的効果の取り入れ方により、いくつかの形状が提案されています [Lenthe96]。

$$\begin{aligned} T^{\text{NR}} &= \frac{1}{2} \mathbf{p}^2, \\ T^{\text{ZORA}} &= \sigma \cdot \mathbf{p} \frac{c^2}{2c^2 - V} \sigma \cdot \mathbf{p}, \\ T^{\text{DIRAC}} &= \sigma \cdot \mathbf{p} \frac{c^2}{2c^2 + \varepsilon_i - V} \sigma \cdot \mathbf{p}. \end{aligned} \quad (2.15)$$

T^{NR} は非相対論、 T^{ZORA} は ZORA (Zero Order Regular Approximation)、 T^{DIRAC} は Dirac 相対論によるものです。ZORA の場合は、scalar relativistic 項とスピン軌道相互作用項に分解できます。

$$\begin{aligned} T^{\text{ZORA}} &= T_{\text{sc}}^{\text{ZORA}} + T_{\text{SOC}}^{\text{ZORA}}, \\ T_{\text{sc}}^{\text{ZORA}} &= \mathbf{p} \frac{c^2}{2c^2 - V} \mathbf{p}, T_{\text{SOC}}^{\text{ZORA}} = \frac{c^2}{(2c^2 - V)^2} \sigma \cdot (\nabla V \times \mathbf{p}) \end{aligned} \quad (2.16)$$

Dirac の場合、上式にて V を $V - \varepsilon_i$ と読み替えばよく、

$$\begin{aligned} T^{\text{DIRAC}} &= T_{\text{sc}}^{\text{DIRAC}} + T_{\text{SOC}}^{\text{DIRAC}}, \\ T_{\text{sc}}^{\text{DIRAC}} &= \mathbf{p} \frac{c^2}{2c^2 - V - \varepsilon_i} \mathbf{p}, T_{\text{SOC}}^{\text{DIRAC}} = \frac{c^2}{(2c^2 - V - \varepsilon_i)^2} \sigma \cdot (\nabla V \times \mathbf{p}) \end{aligned} \quad (2.17)$$

となります。ただし、擬ポテンシャル作成時は scalar-relativistic で解く必要があり、スピン軌道相互作用による効果は考慮されません。このため、CIAO が擬ポテンシャルファイルに出力する、内殻電子の運動エネルギーは scalar-relativistic 項のみで評価します。

さて、内殻電子の運動エネルギーは

$$E_{\text{kin}}^{\text{core}} = \sum_i^{\text{core}} \langle \psi_i | V_{\text{sc}}^{\text{DIRAC}} | \psi_i \rangle \quad (2.18)$$

で表されます。上記は DIRAC の場合ですが、ZORA の場合も同様です。最後に、CIAO ではポテンシャルは球対称として扱うので、波動関数は動径方向成分のみをもちます。このため、運動エネルギー演算子は以下のように展開できます。

$$\begin{aligned} T_{\text{sc}}^{\text{ZORA}} \psi_i(r) &= -\frac{\partial V}{\partial r} \frac{c^2}{(2c^2 - V)^2} \frac{\partial \psi_i}{\partial r} - \frac{c^2}{2c^2 - V} \Delta \psi_i, \\ T_{\text{sc}}^{\text{DIRAC}} \psi_i(r) &= -\frac{\partial V}{\partial r} \frac{c^2}{(2c^2 + \varepsilon_i - V)^2} \frac{\partial \psi_i}{\partial r} - \frac{c^2}{2c^2 + \varepsilon_i - V} \Delta \psi_i, \\ \Delta \psi_i(r) &= \frac{1}{r} \frac{\partial^2}{\partial r^2} (r \psi_i) - \frac{l(l+1)}{r^2} \psi_i \end{aligned} \quad (2.19)$$

内殻電子のスピン

CIAO では、擬ポテンシャルはスピン中性で作成します。内殻電子についても同様であり、この時の内殻電子密度を ρ_{core} とします。ここで、ある軌道 (n,l) におけるスピン占有数 $f_{n,l}^{\uparrow}$ 及び $f_{n,l}^{\downarrow}$ が変わり、 $f_{n,l}^{\uparrow} \neq f_{n,l}^{\downarrow}$ になったとします。ただし、 $f_{n,l}^{\uparrow} + f_{n,l}^{\downarrow}$ は擬ポテンシャル作成時と同じであるとします。このとき、各スピンの内殻電子密度は以下のように表されます。

$$\begin{aligned}\rho_{\text{core}}^{\uparrow}(r) &= \rho_{\text{core}}(r) + \frac{1}{2}\Delta\rho(r) \\ \rho_{\text{core}}^{\downarrow}(r) &= \rho_{\text{core}}(r) - \frac{1}{2}\Delta\rho(r)\end{aligned}\tag{2.20}$$

ここで、

$$\Delta\rho(r) = \left(f_{n,l}^{\uparrow} - f_{n,l}^{\downarrow}\right) \left|\phi_{n,l}^{\text{AE}}(r)\right|^2\tag{2.21}$$

はスピン密度の差で、波動関数 $\phi_{n,l}^{\text{AE}}$ はスピン状態により不変と仮定しています。このスピン密度変化により影響を受けるエネルギーは、交換相関相互作用 (XC) エネルギー

$$\tilde{E}_{\text{XC}}(\rho) - E_{\text{XC}}^{\text{PS}}(\rho) + E_{\text{XC}}^{\text{AE}}(\rho)\tag{2.22}$$

です。第 1 項は非 PAW 項、第 2、3 項は PAW 由来の項です。なお、スピン密度の和は不変であるので、Hartree ポテンシャルは変更を受けません。よって、XC ポテンシャルのみ $\Delta\rho(r)$ を考慮します。

XPS のピークエネルギーの計算方法

非金属の固体の XPS におけるピークエネルギーは、以下の結合エネルギー E_{B} に対応します [Ozaki17]。

$$E_{\text{B}} = E_{\text{final}}(N-1) - E_{\text{initial}}(N) + \mu_0\tag{2.23}$$

ここで、 N は系の電子数です。 μ_0 は系の化学ポテンシャルで、gap のある系では、価電子帯上端 (VBM) から伝導体下端 (CBM) まで取りえます。特に、真性半導体では、バンドギャップを E_g として、

$$\mu_0 = E_{\text{VBM}} + \frac{1}{2}E_g\tag{2.24}$$

となります。なお、VBM のエネルギー E_{VBM} は

$$E_{\text{VBM}} = \frac{E_{\text{initial}}(N) - E_{\text{initial}}(N - \Delta N)}{\Delta N}\tag{2.25}$$

で計算できます。また、帯電状態を周期的境界条件で計算するので、その補正法の 1 つとして、式 (2.2) の ($q=1$) を加えることもできます。

$$E_{\text{B}} = E_{\text{final}}(N-1) - E_{\text{initial}}(N) + \mu_0 + E_{\text{corr}}^1\tag{2.26}$$

一方、系が金属の場合には、

$$E_{\text{B}} = E_{\text{final}}(N) - E_{\text{initial}}(N)\tag{2.27}$$

となります。

なお、固体ではなく分子の場合は、

$$E_{\text{B}} = E_{\text{final}}(N-1) - E_{\text{initial}}(N) + E_{\text{corr}}^1\tag{2.28}$$

となります。

2.5.3 使い方

相対論的運動エネルギー

相対論的運動エネルギーを出力するための CIAO の入力は下記の通り。

```
# PAW
sw_paw      1

# CORE ELECTRON INFO
sw_with_dipole_cor2val 1
method_ekin_core 1          ! default:0 (非相対論), 1:DIRAC, 2: ZORA
```

method_ekin_core 1 或いは 2 で、内殻電子の相対論的運動エネルギーを出力します。なお、“sw_with_dipole_cor2val 1” は、これまでと同様、内殻電子の XPS・XANES 計算には必須です。

“method_ekin_core 1” の場合の 3 つのスイッチの設定は、以下の入力でも可能です。

```
sw_write_core_full 1
```

結果は gncpp2 ファイル (擬ポテンシャルファイル) に出力されます。

```
CORE ENERGY CONTRIB
 0.319190836204E+02    kin          ! 内殻電子の運動エネルギー
-0.677766126155E+02    ion
 0.697542582590E+01    hartr
```

内殻電子のスピン

ある原子の内殻電子が励起された状態を考える場合、該当する軌道に正孔を入れた擬ポテンシャルを使う必要があります。例えば、1s 軌道励起の場合は 1s 軌道に正孔、2p 軌道励起の場合は 2p 軌道に正孔を入れます。これらの原子は、内殻電子が開殻となるので、スピン自由度をもちます。内殻電子のスピンを扱うには、sw_opencore = on とします。デフォルトでは、式 (2.22) の全項でスピンを考慮しますが、sw_xc_opencore_ae_only = on とすると、第 3 項 (AE 部分) のみ考慮します。価電子が磁気モーメントをもつ磁性材料では、spin_orientation の指定により、内殻電子と (原子近傍の) 価電子の磁気モーメントの向きを、平行・半平行のいずれかに指定できます。ただし、SCF iteration で価電子の磁気モーメントが反転したりすると、内殻電子も追従するため、収束性が悪くなる場合があります。そこで、sw_fix_core_spin_pol = on とすると、初期磁気モーメントの向きに固定することが出来ます。

内殻電子のスピンを考慮するための入力例。

```
accuracy{
  paw = on
  core_electrons{
    sw_opencore = on          ! default :off
    sw_xc_opencore_ae_only = on ! default :off
    spin_orientation = anti_parallel ! anti_parallel or parallel (default)
    sw_fix_core_spin_pol = on   ! default: off
  }
}
postprocessing{
  corelevels{
    sw_calc_core_energy = on      ! XPS 計算に必要
  }
}
```

また、s 軌道以外から励起する場合には、内殻電子のスピン軌道相互作用により、XPS のピークが 2 つに分裂します。例えば、2p 軌道の場合には、2p_{1/2} と 2p_{3/2} に分裂します。この場合、以下のようにすると、両者の全エネルギー値を計算することが出来ます。

内殻電子準位のスピン軌道分裂を考慮するための入力例。

```
postprocessing{
  corelevels{
    sw_calc_core_energy = on
    corehole{
      atom_id = 1      ! 励起する原子
      orbital = 2p      ! 励起する軌道
    }
  }
}
```

(内殻電子のエネルギーを含む) 系の全エネルギーは、core_energy.data に出力されます。sw_calc_core_energy = on とした場合には、core_energy.data 末尾に、スピン軌道分裂したエネルギー値が追記されます。(2.23) (2.26) (2.28) などの結合エネルギーの計算における E_{final} や E_{initial} はここに記録された値を利用します。

core_energy.data の出力例

```
# Etotal (Core+Valence)
-18558.3245482622

# Etotal (Core+Valence+Soc_corehole)
# J = 3/2:      -18558.3331659845
# J = 1/2:      -18558.3073128174
```

2.5.4 例題

例題の計算結果を紹介します。XPS の例題はサンプルディレクトリーの下の XPS ディレクトリーに配置されています。用いる擬ポテンシャルは [相対論的運動エネルギー](#) において説明した sw_write_core_full の値を 1 として作成した擬ポテンシャルで、それぞれの例題の pseudo サブディレクトリーに配置されています。

4H-SiC 結晶 (C 1s)

4H-SiC 結晶の例は XPS/4H-SiC 以下に配置されています。そのディレクトリー構成は下記の通りです。

表 2.14: 4H-SiC (C 1s) XPS 計算のディレクトリー構造

第一階層	第二階層	説明
Final		終状態の入力ファイル
	q_0_no_opencore	中性・内殻スピンなし
	q_0_with_opencore	中性・内殻スピンあり
	q_1_no_opencore	電荷+1・内殻スピンなし
	q_1_with_opencore	電荷+1・内殻スピンあり
Initial		初期状態の入力ファイル
	q_0	中性
	q_0.2	電荷+0.2
	q_0.5	電荷+0.5
	q_1	電荷+1

計算条件は以下のとおりです。なお、スーパーセルの格子定数は、自動最適化により得られた値 ($a = 3.108 \text{ \AA}$ 、 $c = 10.170 \text{ \AA}$) を $3 \times 3 \times 3$ 倍しました。

表 2.15: 4H-SiC (C 1s) XPS 計算に用いた条件

平面波カットオフ [Ry]	30.0
電荷密度カットオフ [Ry]	270.0
k 点サンプリング	Monkhorst-Pack $2 \times 2 \times 2$
交換相関相互作用	GGAPBE, PAW
SCF 収束条件 [Ha/atom]	1.0E-8
内殻電子のスピン	sw_xc_opencore_ae_only = on
CIAO の入力	sw_write_core_full 1
(補正項計算で使用する) 誘電率	9.76 (ab 面内), 10.32 (c 軸)

<https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781118313534.app3>

計算結果は以下の通りです。

表 2.16: 4H-SiC (C 1s) の XPS 計算結果 (単位: eV)。式 (2.26) により評価。

	内殻電子のスピン考慮なし	内殻電子のスピン考慮あり
$E_{\text{final}}(N-1) - E_{\text{initial}}(N)$	279.258	272.309
E_{corr}^1	0.237	0.237
$E_{\text{VBM}} (\Delta N = 0.2)$	9.307	
E_g	2.435	
E_B	289.975	283.026
実験値	283 +/- 0.8 ^{a)}	

a) <http://www.xpsfitting.com/2012/01/silicon.html>

内殻電子のスピンを考慮することにより、実験値に近い結合エネルギーが得られています。なお、式 (2.27) を用いた場合、 E_B は以下のようになりました。

表 2.17: 4H-SiC (C 1s) の XPS 計算結果 (単位: eV)。式 (2.27) により評価。

	内殻電子のスピン考慮なし	内殻電子のスピン考慮あり
E_B	290.815	283.864

w-AIN 結晶 (N 1s)

w-AIN 結晶 (N 1s) の例は XPS/w-AIN 以下に配置されています。そのディレクトリー構成は下記の通りです。

表 2.18: w-AlN 結晶 (N 1s) XPS 計算のディレクトリー構造

第一階層	第二階層	説明
Final_N1s		終状態の入力ファイル
	q_0_no_opencore	中性・内殻スピンなし
	q_0_with_opencore	中性・内殻スピンあり
	q_1_no_opencore	電荷+1・内殻スピンなし
	q_1_with_opencore	電荷+1・内殻スピンあり
Initial		初期状態の入力ファイル
	q_0	中性
	q_0.2	電荷+0.2
	q_0.5	電荷+0.5
	q_1	電荷+1

計算条件は以下のとおりです。なお、スーパーセルの格子定数は、自動最適化により得られた値 ($a = 3.140 \text{ \AA}$ 、 $c = 5.040 \text{ \AA}$) $3 \times 3 \times 2$ 倍しました。

表 2.19: w-AlN 結晶 (N 1s)

平面波カットオフ [Ry]	30.0
電荷密度カットオフ [Ry]	270.0
k 点サンプリング	Monkhorst-Pack $2 \times 2 \times 2$
交換相関相互作用	GGAPBE, PAW
SCF 収束条件 [Ha/atom]	1.0E-8
内殻電子のスピン	sw_xc_opencore_ae_only = on
CIAO の入力	sw_write_core_full 1
(補正項計算で使用する) 誘電率	8.23 (ab 面内), 9.74 (c 軸)

<https://materialsproject.org/materials/mp-661/>

表 2.20: w-AlN (N 1s) の XPS 計算結果 (単位: eV)。式 (2.26) により評価。

	内殻電子のスピン考慮なし	内殻電子のスピン考慮あり
$E_{\text{final}}(N-1) - E_{\text{initial}}(N)$	395.772	387.703
E_{corr}^1	0.310	0.310
$E_{\text{VBM}} (\Delta N = 0.2)$	6.735	
E_g	4.404	
E_B	405.019	396.550
実験値	397.4 ^{a)}	

a) [Mahmood03]

内殻電子のスピンを考慮することにより、実験値に近い結合エネルギーが得られています。なお、式 (2.27) を用いた場合、 E_B は以下のようになりました。

表 2.21: AlN (N 1s) の XPS 計算結果 (単位: eV)。式 (2.27) により評価。

	内殻電子のスピン考慮なし	内殻電子のスピン考慮あり
E_B	406.976	398.889

w-AlN 結晶 (Al 2p_{3/2})

w-AlN 結晶 (Al 2p_{3/2}) の例は XPS/w-AlN 以下に配置されています。そのディレクトリー構成は下記の通りです。

表 2.22: w-AlN 結晶 (Al 2p_{3/2}) XPS 計算のディレクトリー構造

第一階層	第二階層	説明
Final_Al2p		終状態の入力ファイル
	q_0_no_opencore	中性・内殻スピンなし
	q_0_with_opencore	中性・内殻スピンあり
	q_1_no_opencore	電荷+1・内殻スピンなし
	q_1_with_opencore	電荷+1・内殻スピンあり
Initial		初期状態の入力ファイル
	q_0	中性
	q_0.2	電荷+0.2
	q_0.5	電荷+0.5
	q_1	電荷+1

計算条件は以下のとおりです。なお、スーパーセルの格子定数は、自動最適化により得られた値 ($a = 3.140$ Å、 $c = 5.040$ Å) $3 \times 3 \times 2$ 倍しました。

表 2.23: w-AlN 結晶 (Al 2p_{3/2})

平面波カットオフ [Ry]	30.0
電荷密度カットオフ [Ry]	270.0
k 点サンプリング	Monkhorst-Pack $2 \times 2 \times 2$
交換相関相互作用	GGAPBE, PAW
SCF 収束条件 [Ha/atom]	1.0E-8
内殻電子のスピン	sw_xc_opencore_ae_only = on
CIAO の入力	sw_write_core_full 1
(補正項計算で使用する) 誘電率	8.23 (ab 面内), 9.74 (c 軸)

<https://materialsproject.org/materials/mp-661/>

表 2.24: w-AlN (Al 2p3/2) の XPS 計算結果 (単位: eV)。式 (2.26) により評価。

	内殻電子のスピン考慮なし	内殻電子のスピン考慮あり
$E_{\text{final}}(N-1) - E_{\text{initial}}(N)$	64.980	64.212
E_{corr}^1	0.327	0.327
$E_{\text{VBM}} (\Delta N = 0.2)$	6.735	
E_g	4.404	
E_B	74.244	73.476
実験値	73.3 ^{a)}	

a) [Mahmood03]

内殻電子のスピンを考慮することにより、実験値に近い結合エネルギーが得られています。なお、式 (2.27) を用いた場合、 E_B は以下のようにになりました。

表 2.25: w-AlN (Al 2p3/2) の XPS 計算結果 (単位: eV)。式 (2.27) により評価。

	内殻電子のスピン考慮なし	内殻電子のスピン考慮あり
E_B	76.221	75.446

fcc Pt 結晶 (Pt 4f)

fcc PtN 結晶 (Pt 4f) の例は XPS/Pt 以下に配置されています。そのディレクトリー構成は下記の通りです。

表 2.26: fcc Pt 結晶 (Pt 4f) XPS 計算のディレクトリー構造

第一階層	第二階層	説明
Final		終状態の入力ファイル
	q_0_no_opencore	中性・内殻スピンなし
	q_0_with_opencore	中性・内殻スピンあり
Initial		初期状態の入力ファイル
	q_0	中性

計算条件は以下のとおりです。なお、スーパーセルの格子定数は、自動最適化により得られた値 ($a = 3.963 \text{ \AA}$) を $2 \times 2 \times 2$ 倍しました。

表 2.27: fcc-Pt 結晶 (Pt 4f) XPS 計算に用いた条件

平面波カットオフ [Ry]	30.0
電荷密度カットオフ [Ry]	270.0
k 点サンプリング	Monkhorst-Pack $2 \times 2 \times 2$
交換相関相互作用	GGAPBE, PAW
SCF 収束条件 [Ha/atom]	1.0E-8
内殻電子のスピン	sw_xc_opencore_ae_only = on
CIAO の入力	sw_write_core_full 1

表 2.28: fcc Pt (Pt 4f 7/2) の XPS 計算結果 (単位: eV)。式 (2.27) により評価。

	内殻電子のスピン考慮なし	内殻電子のスピン考慮あり
E_B	70.673	70.380
実験値	71 ^{a)}	

a) <http://techdb.podzone.net/xps/index.cgi?element=Pt>

表 2.29: fcc Pt (Pt 4f 5/2) の XPS 計算結果 (単位: eV)。式 (2.27) により評価。

	内殻電子のスピン考慮なし	内殻電子のスピン考慮あり
E_B	74.333	74.040
実験値	71 ^{a)}	

a) <http://techdb.podzone.net/xps/index.cgi?element=Pt>

なお、CIAO において内殻電子の運動エネルギーを非相対論で評価した擬ポテンシャルを用いた場合、4f 軌道の結合エネルギー (4f7/2 と 4f5/2 の重みつき平均) は 83.052 eV となり、実験値と離れてしまいます。

O₂ 分子 (O 1s)

表 2.30: O₂ 分子 (O 1s XPS 計算のディレクトリー構造

第一階層	説明
final_spin_antiparallel	終状態・スピン反平行
final_spin_parallel	終状態・スピン平行
initial	始状態

計算条件は以下のとおりです。

表 2.31: O₂ 分子 (O 1s) XPS 計算に用いた条件

平面波カットオフ [Ry]	30.0
電荷密度カットオフ [Ry]	270.0
k 点サンプリング	Γ 点のみ
交換相関相互作用	GGAPBE, PAW
単位胞の 1 辺 [Å]	14.0
SCF 収束条件 [Ha/atom]	1.0E-8
内殻電子のスピン	sw_xc_opencore_ae_only = on
CIAO の入力	sw_write_core_full 1
(補正後計算で使用する) 誘電率	1.00

真空中

表 2.32: O₂ 分子 (O 1s) XPS 計算結果 (単位: eV)。式 (2.28) により評価。

	S=1/2	S=3/2
spin orientation の指定	anti parallel	parallel
$E_{\text{final}}(N-1) - E_{\text{initial}}(N)$	542.000	541.297
E_{corr}^1	1.436	1.439
E_B	543.436	542.736
実験値 ^{a)}	544.2	543.1

a) https://t-ozaki.issp.u-tokyo.ac.jp/vps_pao_core2019/O/index.html and references therein.

2 つのスピン状態とも、実験値に近い値が得られました。

2.6 機械学習によるニューラルネットワークポテンシャルの作成

2.6.1 概要

ニューラルネットワークポテンシャルとは、凝縮系のポテンシャルエネルギー表面を近似することのできる学習モデルです [Behler07]。その学習は、PHASE/0 などの第一原理計算の結果を教師データとして行われます。PHASE/0 には、`ænet` [Artrith16] や `n2p2` [Singraber19] などのソフトウェアと連携し、ニューラルネットワークポテンシャルを作成することができる機能が備わっています。

2.6.2 使い方

教師データの作成

以下に説明するような手続きによって、n2p2 および ænet の教師データとして用いることのできるデータを出力することができます。

入力パラメーターファイルに以下のような設定を施します。

```
structure_evolution{
  ...
  ...
  nnp_output{
    sw_nnp_output = on
    filetype = all
    frequency = 100
  }
}
```

Structure_evolution ブロックにおいて nnp_output ブロックを作成し、機械学習ポテンシャル用出力の設定を行います。sw_nnp_output = on とすると機械学習ポテンシャル用出力が行われます。filetype で出力するファイルの種類を指定することができます。XSF を指定すると ænet で利用できる XSF 形式、n2p2 を指定すると n2p2 で利用できる形式、all を指定すると XSF と n2p2 形式双方が出力されます。frequency によってサンプリングの頻度を指定します。このパラメーターのデフォルト値は 100 です。ファイル名のデフォルト値は、n2p2 が input.data、XSF が nfdynm xxxxx .xsf です。ここで xxxxx は MD ステップ数です。n2p2 は学習データが一つのファイルに記録されるのに対し、ænet はスナップショット 1 つにつき 1 つの XSF が必要です。

file_names.data ファイルにおいては、n2p2 および XSF 形式のファイルのファイル名を指定することができます。デフォルト値はそれぞれ input.data および nfdynm xxxxx .xsf です。

ファイル ポ イ ン ター	説明
F_N2P2	n2p2 の教師データの出力先。デフォルト値は input.data
F_XSF	ænet の教師データの出力先。ここで指定した文字列の拡張子に相当する部分の直前に MD のイテレーション数が付加されたファイル名が採用される。デフォルト値は nfdynm.xsf

サンプル時の収束判定条件の切り替え

教師データをサンプルする際、相関が強くなりすぎないような程度間隔をあけて行うことが一般的ですが、サンプルするステップ以外のステップの重要性は低いと考えられます。そこで、サンプルするステップ以外は甘い収束判定で計算を行うことによって計算時間の短縮をはかることができます。このような処理を実現するには、以下のような設定を入力パラメーターファイルに施します。

```
accuracy{
  ...
  scf_convergence{
    delta_total_energy = 1e-4 Hartree
    delta_total_energy_sampling = 1.e-9
    succession = 1
  }
}
```

accuracy ブロックの scf_convergence ブロックにおいて SCF 計算の収束判定条件の設定は行われるが、ここで delta_total_energy_sampling キーワードを使ってサンプリングの時のみに用いる収束判定条件を指定することができます。この例では、通常の SCF 計算では $1e-4$ hartree という収束判定条件が採用されますが、サンプリングステップの場合 $1e-9$ という収束判定条件が採用されます。

教師データの形式

教師データのファイル形式について説明します。

input.n2p2 ファイル

input.n2p2 ファイルは n2p2 プログラムで利用できる教師データ形式です。以下の形式で原子配置と対応するエネルギーが記録されます。

```
begin
comment iteration      100
lattice      5.6022996655      0.0000000000      0.0000000000
lattice      0.0000000000      9.7034676599      0.0000000000
lattice      0.0000000000      0.0000000000      14.7489157721
atom  5.054 4.636 1.407 Li  0.00 0.00 -0.141 -0.0743 0.1568
...
energy -25706.79549229032
charge 0.0
end
begin
...
end
```

begin と end の間に 1 つのデータが記録されます。comment から始まる行でコメントを記述できます。lattice から始まる行で格子ベクトルの指定を行います。atom から始まる行で原子の情報を記述します。2 カラム目から 4 カラム目に原子の xyz 座標を指定し、5 カラム目に元素名を指定します。6, 7 カラム目は未使用領域ですが、適当な数値を記述する必要があります。8 カラム目から 10 カラム目で原子間力を指定します。原子間力は学習に直接利用されるわけではありませんが、結果得られる機械学習ポテンシャルから得られる原子間力との差を評価し、誤差を推測する、という使われ方がなされる場合があります。energy から始まる行でエネルギーを指定します。charge から始まる行で系の総電荷を指定します。ただし charge 指定による情報は現バージョンの n2p2 では利用されないようです。利用される単位は単位はエネルギーが eV、距離が Å、原子間力が eV/Å です。

n2p2 の入力として利用するには、データは input.data というファイルに記録されている必要があります。複数のディレクトリーにまたがって行った計算の結果を活用したい場合、これらを cat コマンドとリダイレクトなどを利用して一つのファイルにまとめる必要があります。

```
$ cat A/input.data > input.data
$ cat B/input.data >> input.data
$ cat C/input.data >> input.data
...
```

xsf

xsf は XCrysDen などにおいて標準的に用いられる座標データ形式です。ænet はこの形式を教師データ形式として採用しています。XSF においては、以下の形式で原子配置と対応するエネルギーが記録されます。

```
# total energy =      -25711.2660427032 eV
CRYSTAL
PRIMVEC
```

(次のページに続く)

(前のページからの続き)

```

5.6022996655      0.0000000000      0.0000000000
0.0000000000      9.7034676599      0.0000000000
0.0000000000      0.0000000000      14.7489157721
PRIMCOORD
  84      1
Li  5.0229472700  5.0300845743  1.8014586046 -0.1368952902 -0.0902290564 -0.4716359333
...
```

#から始まる行は XSF 形式ではコメント行とみなされますが、`ænet` は # total energy という文字列がある場合はエネルギーが指定される行とみなし、= の後の数値をエネルギー値として採用します。CRYSTAL 行を記述することによってこのあと結晶のデータが記録されていることを指定します。PRIMVEC 行を記述することによってこのあとの 3 行が *a* 軸、*b* 軸、*c* 軸の格子ベクトルの指定であることを指定します。PRIMCOORD を記述することによって続く行が原子座標や原子間力の指定に利用されることを指定します。まずは原子数が指定されます。この例では 84 1 となっており、84 原子系であることが指定されています。その次の 1 という数値は `ænet` では利用しない情報ですが XSF 形式の様式を満たすために必要です。その次の行以降が原子の情報です。1 カラム目が元素名、2 カラム目から 4 カラム目が原子位置、5 カラム目から 7 カラム目が原子間力です。単位は n2p2 の場合と同様エネルギーが eV、距離が Å、原子間力が eV/Å です。

2.6.3 ニューラルネットワークポテンシャル作成例

ここでは、 α Quartz 結晶を例に、ニューラルネットワークポテンシャルの作成方法について説明します。

用いた結晶

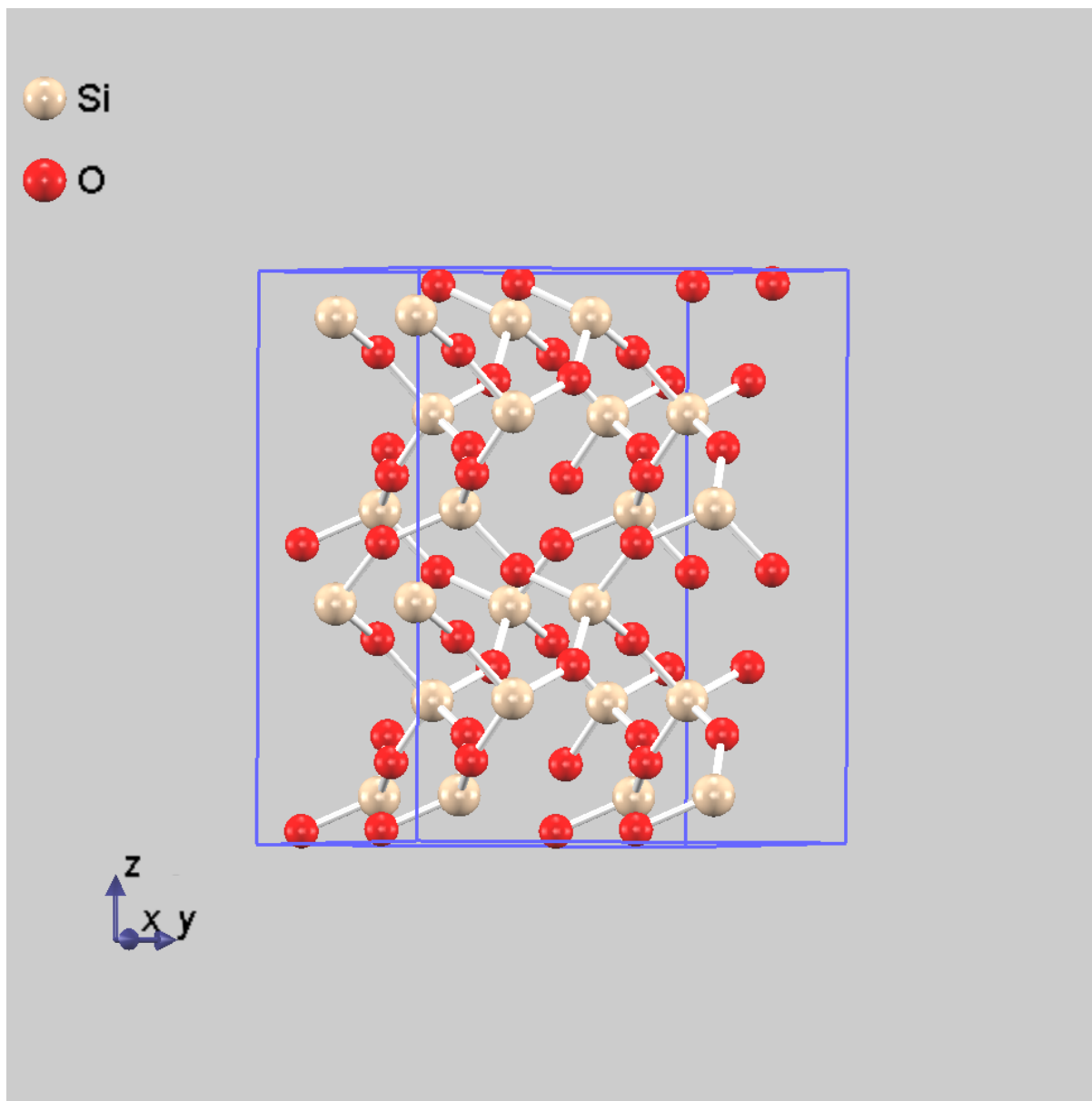
六方晶の α Quartz 結晶を直方晶に取り直し、さらに *a* 軸と *c* 軸をそれぞれ 2 倍とするスーパーセルを作成しました。合計 72 原子の系です。その初期配置は [図 2.6](#) に示す通り。

第一原理計算

[図 2.6](#) の結晶を初期構造として、温度 500K、800K、1200K で定温の第一原理計算分子動力学シミュレーションを実施しました。入力ファイルと教師データのアーカイブはサンプルディレクトリーの下の `nnp/SiO2/fpmd` 以下のサブディレクトリーにあります。温度以外の主要な計算条件は下記の通り。

表 2.33: 第一原理分子動力学の主要な計算条件

計算条件	値
カットオフエネルギー	25 Rydberg
k 点サンプリング	Γ 点のみ
収束判定条件	1e-9 hartree
サンプリング頻度	10 MD ステップに一度
総 MD ステップ数	10000
教師データ出力	XSf 形式

図 2.6: 72 原子 α Quartz 結晶

このシミュレーションにおいてはサンプリング頻度が 10 MD ステップに一度と比較的高いため、収束判定条件を切り替える機能は用いていません。

ニューラルネットワークポテンシャルの作成の準備

得られた教師データを用いてニューラルネットワークポテンシャルを作成する方法を説明します。用いるアプリケーションは `ænet` です。`ænet` の詳細については [ウェブサイトの情報](#) を参照してください

まずは、サンプルディレクトリーの下に教師データのアーカイブ（ファイル名 `xsf.tar.bz2`）が配置されているので、それを解凍します。

```
$ cd samples/nnp/SiO2/fpmd/500K
$ tar -jxvf xsf.tar.bz2
$ ...
$ cd ../800K
$ ...
```

成功すると、各ディレクトリーにファイル名が `nfdynm.xxxxx.xsf` の 1000 個の `xsf` が作成されるはずです。

また、`ænet` のウェブサイトから最新版のアーカイブをダウンロードし、コンパイルしてください。成功すれば、`ænet` のインストールディレクトリーの下に `bin` ディレクトリーに `generate.x-*`, `train.x-*`, `predict.x-*` の 3 つのバイナリーが作成されるはずです（ここで `*` は `ænet` のバージョンやコンパイル環境を識別する文字列；以降 `*` は省略）

最後に、古典分子動力学シミュレーター LAMMPS [Plimpton95] から `ænet` で作れるニューラルネットワークポテンシャルを用いる場合、LAMMPS の `ænet` インターフェース [Mori20] を [作者のウェブサイト](#) からダウンロードし、指示に従って LAMMPS に組み込んでください。

教師データの変換

教師データとしての原子配置は、必ずしも機械学習に適したものではないので、それを機械学習に適した形式（記述子）に変換する必要があります。このような変換を行うプログラムが `generate.x` です。`generate.x` を実行するために必要な入力 `generate.x` の全体の振る舞いを制御するコントロールファイル（`generate.in`）と元素ごとの記述子の構築の仕方を指定するセットアップファイルです。サンプルデータは `samples/nnp/SiO2/aenet/generate` の下にあります。

`generate.in` ファイルの内容は、典型的には下記のようになります。

```
OUTPUT SiO.train

TYPES
2
Si -107.171 | eV
O -434.109 | eV

SETUPS
Si Si.fingerprint.stp
O O.fingerprint.stp

FILES
3000
../fpmd/800K/nfdynm000772.xsf
../fpmd/800K/nfdynm000038.xsf
...
...
```

- OUTPUT に `generate.x` の結果を出力するファイルを指定します。この例では `SiO.train` です。

- TYPES において原子種の指定を行います。まず利用する元素数を指定し（この例では2）、一行に1元素ずつ元素名と原子の参照エネルギーを指定します。この例では、シリコン結晶と酸素分子の1原子あたりのエネルギーを採用しました。
- SETUPS に元素ごとのセットアップファイルを 元素名 セットアップファイル という形式で指定します。
- FILES において教師データを指定します。まず教師データの数指定し、1行に1ファイル教師データが記録されたファイルのパスを指定します。

セットアップファイルの内容は、典型的には下記ようになります。

```
ATOM Si
ENV 2
Si
0
RMIN 0.85d0
BASIS type=Chebyshev
radial_Rc = 6.0 radial_N = 16 angular_Rc = 6.0 angular_N = 8
```

- ATOM 元素名とし、対象元素を指定します。
- ENV N とし、その元素と相互作用し得る元素の数を指定します。さらに、具体的にどの元素が相互作用し得るかを1行につき1元素名という形式で指定します。
- RMIN によって最小距離を指定します。
- BASIS において記述子の種類を指定します。type=Chebyshev とすると Chebyshev 展開を用いて記述子を構築します [Artrith17] この場合、後段の radial_Rc と radial_N によって動径方向のカットオフ距離と分割数、angular_Rc と angular_N によって角度方向のカットオフ距離と分割数を指定します。

generate.in ファイルとセットアップファイルが準備できたら、generate.x を実行します。

```
$ generate.x generate.in > generate.log
```

この処理の結果、generate.in において OUTPUT キーワードで指定したファイルに結果が記録されます。このファイルを学習用のディレクトリー（今の例の場合 samples/nnp/SiO2/aenet/train）にコピーします。

ニューラルネットワークポテンシャルの学習

generate.x によって変換した座標データとエネルギーのセットを用いて train.x による学習を行うことができます。そのコントロールファイル (train.in) は、典型的には以下のように記述します。

```
TRAININGSET SiO.train
TESTPERCENT 10
ITERATIONS 5000
MAXENERGY 1.0
METHOD
bfgs
NETWORKS
! atom network hidden
! types file-name layers nodes:activation
Si Si.10t-10t.nn 2 10:tanh 10:tanh
O O.10t-10t.nn 2 10:tanh 10:tanh
```

- TRAININGSET で generate.x によって作成されたファイルを指定します。

- TESTPERCENT で教師データの学習ではなくテストに利用するパーセンテージを指定します。
- ITERATIONS 学習は iterative に行われますが、その iteration の最大回数を指定します。
- METHOD: 学習の手法を指定します。bfgs のほか、Levenberg-Marquardt 法などを利用することができます。bfgs 法は設定が必要なパラメーターが少なく、ロバストな最適化が可能のため推奨の方法です。
- NETWORKS においてベースとなるニューラルネットワークを指定します。1 行につき 1 元素の指定を行います。まず元素名を記述し、つぎにニューラルネットワークの重みを記録するファイルを指定します。さらに隠れ層の数を指定し、最後に隠れ層の数分ノード数と activation function (階段関数の近似) の形式を指定します。

train.x を以下の要領で実行します。

```
$ mpiexec -n 8 train.x train.in > train.log
```

このコマンドは MPI を介して 8 並列で実行する例です。具体的なコマンドはシステムによって異なる可能性があります。

ログは標準出力に出力されるので、この例ではリダイレクトされ train.log というファイルに出力されます。train.log には様々な情報が記録されます。学習がうまくいっているかどうかの確認は以下のように Training process 以降のデータを参照するようにしてください。

```
-----
Training process
-----
Weight optimization for 5000 epochs using the Limited Memory BFGS method.
Sampling type          : sequential

epoch |-----TRAIN-----| |-----TEST-----|
      | MAE      <RMSE>      | MAE      <RMSE>      |
0      | 2.055153E-01  2.392261E-01 | 2.059784E-01  2.414336E-01 <
1      | 2.055153E-01  2.392261E-01 | 2.059784E-01  2.414336E-01 <
2      | 1.338985E-01  1.499288E-01 | 1.358931E-01  1.509523E-01 <
3      | 4.876623E-02  7.448498E-02 | 4.891127E-02  7.381589E-02 <
4      | 3.806436E-02  5.418523E-02 | 3.679157E-02  5.251901E-02 <
...
...
```

特に TEST の MAE (mean absolute error), RMSE (root mean square error) の値に注目し、これが初期値に比べ小さくなっている epoch を採用するようにしてください。

ネットワークの重みづけファイル(分子動力学シミュレーションなどにおいて利用するファイル)は train.in において指定した文字列に、学習の iteration 回数が付与されたファイル名(たとえば Si.10t-10t.nn-00100 など)のファイルに出力されます。

ニューラルネットワークポテンシャルの検証

ænet 付属のプログラム predict.x を用いることによって、作成したニューラルネットワークポテンシャルの検証を行うことができます。predict.x は xsf とニューラルネットワークポテンシャルの重みファイルを入力とし、エネルギーや原子間力の計算と構造最適化を行うことができます(predict.x でできることはこれだけなので、本格的な活用のためには LAMMPS インターフェースなどを利用することが推奨されます)

predict.x のコントロールファイル(predict.in)は典型的には以下のようになります(このサンプル入力 samples/nnp/SiO2/ænet/predict 以下にあります)

```

TYPES
2
Si
0

NETWORKS
Si Si.10t-10t.nn
O O.10t-10t.nn

FORCES

FILES
3000
../fpmd/800K/nfdynm000772.xsf
../fpmd/800K/nfdynm000038.xsf
...

```

- TYPES の次の行で原子種の数指定します。それに続く行で元素名を記述します。
- NETWORKS に続く行で、元素名 ニューラルネットワークポテンシャルの重みファイル名 という形式でニューラルネットワークポテンシャルの重みファイルを指定します。
- 原子間力を計算する場合、FORCES というキーワードを記述します。
- FILES キーワードとそれに続く行において、generate.in と同じ形式で対象の原子配置を記述した XSF を指定することができます。この部分がない場合、対象の原子配置ファイルは predict.x 実行時に引数指定する。

predict.x は generate.x, train.x と同じように実行することができます。

```
$ predict.x predict.in > predict.log
```

結果は標準出力に出力されるので、上述のコマンド例の場合 predict.log ファイルに記録されます。たとえば以下のような内容になります。

```

-----
Parallel run
-----

Number of processes : 1
...
...
...
Cartesian atomic coordinates (input) and corresponding atomic forces:

```

	x (Ang)	y (Ang)	z (Ang)	Fx (eV/Ang)	Fy (eV/Ang)	Fz (eV/Ang)
Si	9.649365	8.409237	9.994378	0.235924	0.525635	-1.368195
Si	1.351787	1.864069	0.963064	2.655693	2.032425	-0.024373
Si	1.296394	6.654412	2.411131	1.116420	0.752235	1.365680
O	8.774874	1.120195	9.374453	-0.116155	1.088577	-0.440955
...						
...						

```

Cohesive energy      :      -184.66378181 eV
Total energy         :     -23593.99978181 eV
Mean force (must be zero) :    -0.000000      0.000000      0.000000
...

```

図 2.7 に predict.x で計算したエネルギーと第一原理計算によって得られたエネルギーをプロットした図を示します。この図では、完全な学習ができていた場合はすべての点が斜め 45 度の直線にのります。学習が完璧であることはないの直線からのバラつきはあるものの、直線からの乖離は原子あたりおおむね 1 meV 以下であり、よく学習できていることが分かります。

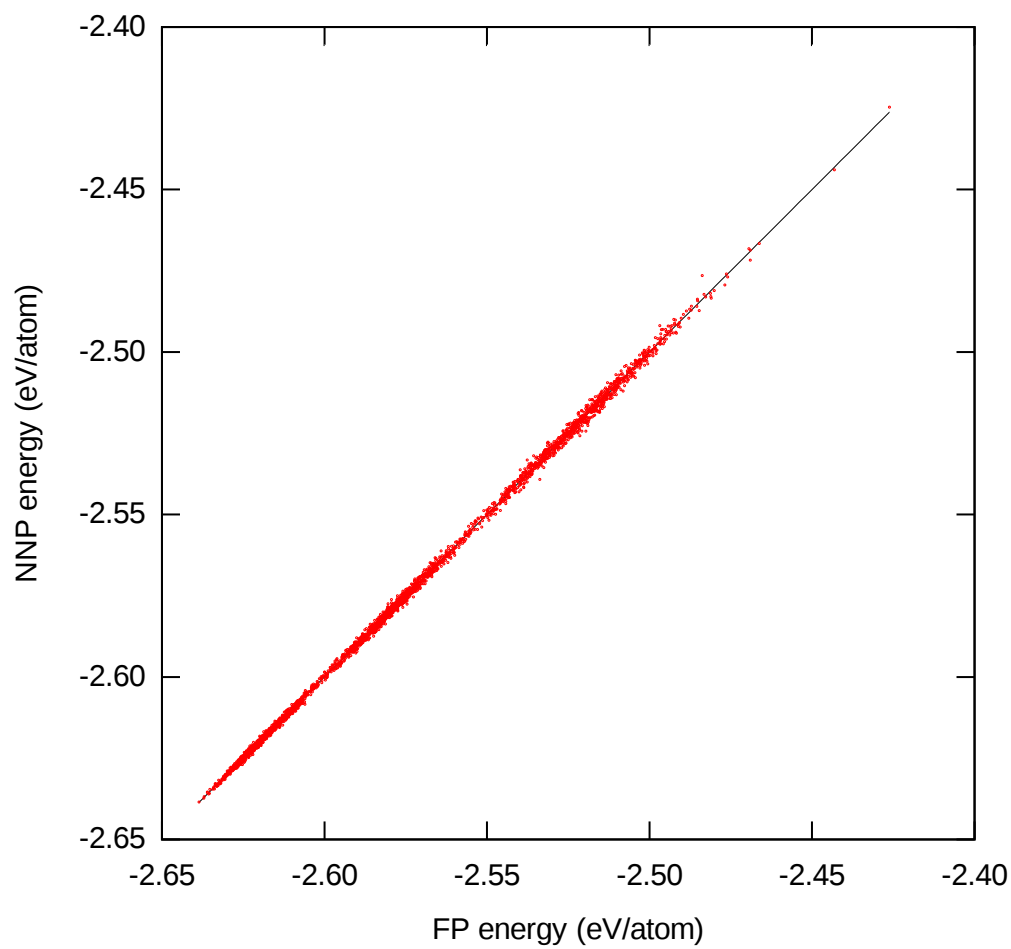


図 2.7: 第一原理計算とニューラルネットワークポテンシャルによって得られるエネルギーの比較

ænet インターフェースを組み込んだ LAMMPS の使い方

ニューラルネットワークポテンシャルを用いて LAMMPS による計算を行うには、まず重みファイル (Si.10t-10t.nn-xxxxx など) のファイルを実行ディレクトリーにコピーします。その際、ファイル名末尾の “-xxxxx” の部分は取り除いたファイル名にします。

LAMMPS の入力スクリプトは通常通り用意します。単位は metal, atom_style は atomic を用います。すなわち、以下のような指定は必須です。

```
units metal
atom_style atomic
```

ポテンシャル指定部分には以下のような文字列を指定します。

```
pair_style      ænet
pair_coeff      * * v00 Si O 10t-10t.nn Si O
```

- pair_style ænet によって ænet のニューラルネットワークポテンシャルを利用することを指定します。
- pair_coeff によってニューラルネットワークの重みファイルと元素のマッピングを行います。
 - * * という文字列によってすべての元素間が対象であることを指定します。
 - v00 によって ænet プログラムが出力するファイルを利用することを指定します。
 - つづく Si O 10t-10t.nn は Si.10t-10t.nn, O.10t-10t.nn の 2 つのファイル指定の短縮形です。
 - 最後の Si O は元素へのマッピングで、1 つ目のファイルは Si, 2 つ目のファイルは O に対応する重みファイルであることを意味します。
- 初期座標データファイルは、LAMMPS の atom_style atomic で用いられる形式で準備します。

図 2.8 に、ニューラルネットワークポテンシャルと PHASE/0 による 500K および 800K における α Quartz の分子動力学シミュレーションのトラジェクトリーから得られた二体分布関数および結合角分布関数を比較した結果を示します。図 2.8 より、得られたニューラルネットワークポテンシャルは第一原理計算の結果をよく再現するものであることが分かります。

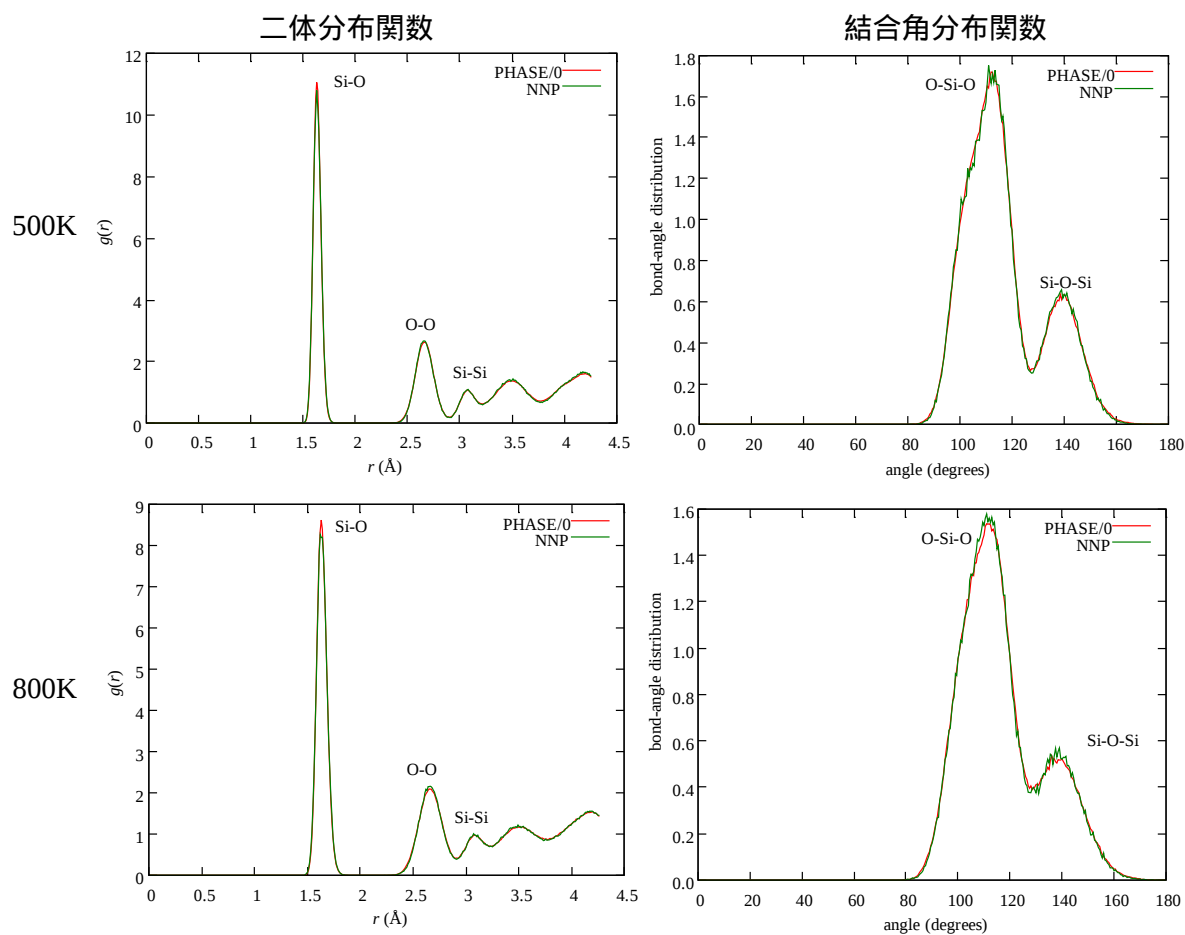


図 2.8: 500K および 800K における二体分布関数と結合角分布関数の比較。赤線：PHASE/0 による第一原理分子動力学シミュレーション，緑線：ニューラルネットワークポテンシャルによる古典分子動力学シミュレーション。

関連図書

- [Freysoldt09] C. Freysoldt, J. Neugebauer, and C. G. Van de Walle, Phys. Rev. Lett. 102 (2009) 016402.
- [Kumagai14] Y. Kumagai and F. Oba, Phys. Rev. B 89 (2014) 195205.
- [Broberg18] D. Broberg, B. Medasani, N. Zimmermann, A. Canning, M. Haranczyk, M. Asta, and G. Hautier,, Comput. Phys. Commun. 226 (2018) 165.
- [Methfessel89] Methfessel and A. T. Paxton, Phys. Rev. B 40 (1989) 3616.
- [Kresse96] G. Kresse and J. Furthmüller, Computational Materials Science 6 (1996) 15.
- [Hjorth17] Ask Hjorth Larsen et al. J. Phys.: Condens. Matter 29 (2017) 273002.
- [Lenthe96] <https://www.scm.com/wp-content/uploads/vlenthe.pdf>
- [Ozaki17] T. Ozaki and C. Lee, Phys. Rev. B 118 (2017) 026401.
- [Mahmood03] A. Mahmood, R. Machorro, S. Muhl, J. Heiras, F. F. Castillon, M. H. Farias, and E. Andrade, DIAM RELAT MATER 12 (2003) 1315.
- [Behler07] J. Behler and M. Parrinello, Phys. Rev. Lett. 98, 146401 (2007)
- [Artrith16] N. Artrith and A. Urban, Comput. Mater. Sci. 114 (2016) 135-150.
- [Singraber19] Singraber, A.; Behler, J.; Dellago, C. J. Chem. Theory Comput. 15 (2019) 1827–1840.
- [Plimpton95] S. Plimpton, J. Comp. Phys. 117 (1995) 1-19.
- [Mori20] H. Mori and T. Ozaki, Physical Review Materials 4 (2020) 040601.
- [Artrith17] N. Artrith, A. Urban, and G. Ceder, Phys. Rev. B 96 (2017) 014112.