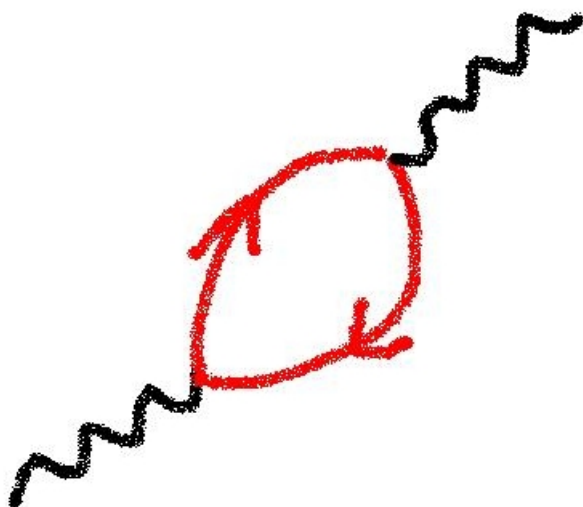


RESPACK マニュアル

中村 和磨

(九州工業大学大学院基礎科学研究系)

平成 29 年 10 月 20 日



目 次

1	はじめに	4
1.1	プログラム	4
1.2	単位系	4
1.3	ソースコードのダウンロード	4
1.4	ソースコードのコンパイル	4
1.5	計算ディレクトリ	5
2	RESPACK 入力ファイルの生成	6
2.1	xTAPP から RESPACK への接続手続き	6
2.2	QUANTUM ESPRESSO から RESPACK への接続手続き	7
3	wannier	8
3.1	インプットファイル	8
3.1.1	¶m_wannier	8
3.1.2	¶m_interpolation	10
3.1.3	¶m_visualization	10
3.2	用意すべきバンド計算データファイル	11
3.3	ジョブの実行	11
3.4	計算結果	11
4	chiqw	14
4.1	インプットファイル (¶m_chiqw)	14
4.2	用意すべきバンド計算データファイル	15
4.3	ジョブの実行	15
4.4	計算結果	16
5	calc_w3d および calc_j3d	18
5.1	インプットファイル (¶m_calc_int)	18
5.2	用意すべきバンド計算データファイル	18
5.3	ジョブの実行	19
5.4	計算結果	19
6	まとめ: AI 計算インプット	22
7	La ₂ CuO ₄ (制限 RPA を用いた相互作用評価)	23
8	TiO ₂ (局所座標を用いるワニエ関数計算)	28
9	Si (B 行列を用いるワニエ関数計算)	30
10	SrVO ₃ (制限 RPA を用いた相互作用評価)	32
11	(TMTSF) ₂ PF ₆ (制限 RPA を用いた相互作用評価)	33
12	計算 q 点を指定して計算したい場合 (chiqw)	34
13	実空間ワニエ関数を指定して計算したい場合 (wannier)	34

14 ファイルフォーマット (読込形式)	35
15 ライセンス	37
16 引用法	37

1 はじめに

1.1 プログラム

RESPACK は物質の相互作用パラメータを評価するソフトウェアであり、3 つのプログラムからなる:

1. 誘電関数計算 (chiqw)
2. ワニエ関数計算 (wannier)
3. 電子間相互作用計算 (直接積分: calc_w3d, 交換積分: calc_j3d)

これら 3 つのプログラムは独立しており、各プログラムが

- インプットファイル
- 計算に必要なファイル群

を必要とする。RESPACK はノルム保存型擬ポテンシャル+平面波基底を用いるバンド計算に対応しており、計算前にあらかじめバンド計算が実行されていなければならない。バンド計算の出力ファイルから RESPACK の入力ファイルを自動生成するスクリプトが、現在 xTAPP と QUANTUM ESPRESSO に対して用意されている。本マニュアルでは、第一原理バンド計算終了後 RESPACK により有効相互作用パラメータを評価するまでの手続きを、Al の相互作用パラメータ評価を例にとり具体的に説明する。制限 RPA 法を用いた計算例として La_2CuO_4 についても説明する。できるだけ具体例が多い方がよいと考え、Si, TiO_2 , SrVO_3 , 有機化合物 $(\text{TMTSF})_2\text{PF}_6$ などのインプットも提示する。

1.2 単位系

プログラムは原子単位で記述されるが、入出力においては、慣習の為、 \AA , Ry, eV, あるいは格子座標系などを用いる場合がある。

1.3 ソースコードのダウンロード

現在、公式ウェブサイト <https://sites.google.com/view/kazuma7k6r> 内に最新版のソースコード一式が置かれている。このサイトから `RESPACK.tar.gz` をダウンロードする。

1.4 ソースコードのコンパイル

ソースコード `RESPACK.tar.gz` を自身のマシンに移しファイルを展開する。展開ディレクトリ RESPACK 内には (i) 計算例のインプットを納めたディレクトリ `sample`, (ii) ソースコードを納めたディレクトリ `src`, (iii) 計算を補完するために必要なスクリプトを納めたディレクトリ `util`, (iv) 本ソフトウェアライセンスを納めたディレクトリ `GPL` がある。`src` に移ると 3 つのソースコードを納めたディレクトリ `calc_int`, `chiqw`, `wannier` がある。各ディレクトリの中に関係プログラムと `Makefile` が用意されている。実行したいプログラムのディレクトリに移り `make` を実行する。一例として `chiqw` の場合ならここまでの作業は以下となる。

```
> tar -zxvf RESPACK.tar.gz
> cd RESPACK/
> ls
GPL/ sample/ src/ util/
> cd src/
> ls
calc_int/ chiqw/ wannier/
> cd chiqw/
> ls
Makefile est_NTK.f90  m_fft3d.f90      est_nwx2.f90 m_rdinput.f90
chiqw.F  gen_grid.f90 m_tetrahedron.F chiqw_sub.F  inv.F90
config.h invZGE.F90
> make
> ls calc_chiqw
calc_chiqw
```

上記の Makefile がコンパイルスクリプトであり make 終了後, 実行形式のファイル calc_chiqw が生成される. こうした作業は, 他のプログラムでも共通である.

1.5 計算ディレクトリ

このチュートリアルでは, 計算を実行するディレクトリをカレントディレクトリとし, この位置を ./ と記す. 計算ディレクトリの下に作られるディレクトリは ./dir-xxx のように記す. 上記のソースコードのコンパイルを経て生成した実行形式のファイルをカレントディレクトリに移動させる. ここまでが計算の準備段階となる.

2 RESPACK 入力ファイルの生成

RESPACK はバンド計算を行って得た計算結果を用いて誘電関数等の計算を行うプログラムパッケージであり、バンド計算があらかじめ実施されていることを前提とする。本章では、汎用性バンド計算ソフトウェア xTAPP および QUANTUM ESPRESSO を用いたバンド計算から RESPACK への接続手続きについて記す。バンド計算についての詳細は各ソフトウェアの公式マニュアルに譲るが、ここでは計算に必要な点にのみ焦点を当て実務的手続きを記す。

2.1 xTAPP から RESPACK への接続手続き

xTAPP バンド計算終了後 RESPACK 計算の入力ファイルを生成させる。ディレクトリ RESPACK/util/xtapp2respack に移り wfn2respack.F90 をコンパイルする:

```
> ./COMPILE.wfn2respack.csh
```

ディレクトリ内に wfn2respack が生成される。wfn2respack と同ディレクトリ内にあるスクリプトファイル xtapp2respack.sh の二つが必要ファイルである。バンド計算を実施したディレクトリに戻りこれらを同ディレクトリにコピーする。同ディレクトリにはバンド計算出力である波動関数ファイル (@@@.wfn) と構造ファイル (@@@.str) がある (拡張子前の @@@ は xTAPP では通常、物質名を指定する)。これらのデータから RESPACK 計算に必要な入力データを生成する。スクリプト実行後、ディレクトリ ./dir-wfn が作られ、この中に dat.xxx という名称のファイルが 8 つ生成される。例えば、Al 計算の場合はバンド計算を実施したディレクトリにて以下のように行う。

```
> ./xtapp2respack.sh -b ./wfn2respack Al
> ls ./dir-wfn
dat.bandcalc dat.eigenvalue dat.kg          dat.lattice
dat.nkm       dat.sample-k    dat.symmetry dat.wfn
```

生成される各ファイルは以下である。なお各ファイルのフォーマットは 14 章を参照されたい。

-
1. ./dir-wfn/dat.bandcalc (バンド計算情報)
 2. ./dir-wfn/dat.sample-k (サンプル k 点)
 3. ./dir-wfn/dat.nkm (展開 G 数)
 4. ./dir-wfn/dat.eigenvalue (エネルギー固有値)
 5. ./dir-wfn/dat.kg (逆格子ベクトル)
 6. ./dir-wfn/dat.wfn (波動関数)
 7. ./dir-wfn/dat.lattice (基本並進ベクトル)
 8. ./dir-wfn/dat.symmetry (対称性)
-

以上までが xTAPP バンド計算から RESPACK 計算前までの手続きである。

2.2 QUANTUM ESPRESSO から RESPACK への接続手続き

QUANTUM ESPRESSO バンド計算終了後 RESPACK 計算の入力ファイルを生成させる。手続きは、ディレクトリ RESPACK/util/qe2respack 内の README.txt に纏められている。同ディレクトリに移り、以下の手順で作業を行う:

1. RESPACK/util/qe2respack 下の Makefile の編集

具体的には Makefile 内の変数 QEROOT を Quantum ESPRESSO iotk toolkit を使うためのルートディレクトリへのセット, LAPACK へのリンク部分の編集等である (詳しくは同ディレクトリ下の README.txt 参照されたい)。

2. RESPACK/util/qe2respack 下にて make 実行

qe2respack が生成される。

3. RESPACK/util/qe2respack 下の qe2respack.sh の編集

具体的には qe2respack.sh 内の変数 qe2respack_root を qe2respack.sh および qe2respack が置かれているディレクトリにセットする。

4. RESPACK 用入力ファイルを納めたディレクトリ ./dir-wfn を生成したいディレクトリ (任意) にてスクリプト実行:

```
> bash qe2respack.sh ./outdir/prefix.save/
```

ここで ./outdir/prefix.save/ は QUANTUM ESPRESSO バンド計算の出力が納められているディレクトリである。

この一連の手続きを実行後、ディレクトリ ./dir-wfn が手順 4 を実行したディレクトリの下に作られ、この中に dat.xxx という名称の RESPACK 計算に必要なファイル群 [計 8 つ (2.1 参照)] が生成される。以上までが QUANTUM ESPRESSO バンド計算から RESPACK 計算前までの手続きである。

3 wannier

準備終了後 RESPACK 計算を実行する。まずワニエ関数計算 (wannier) の手続きを記述する。バンド計算には xTAPP を用いた。バンド計算の条件であるが、 k 点サンプリングを $6 \times 6 \times 6$ (既約 k 点数 16), 波動関数カットオフを 36 Ry, 全バンド数を 50 (占有バンド数 1, 部分占有バンド数 1, 非占有バンド数 48) であり, フェルミ準位上約 120 eV 上までのバンドを包含) にとった。

3.1 インプットファイル

Al のワニエ関数計算を実施するためのインプットファイル input.in について解説する。インプットファイルは Fortran のネームリスト形式を用いる。ワニエ関数計算に関わるインプットは 3 つのネームリストから構成される [¶m_wannier (ワニエ関数計算), ¶m_interpolation (ワニエ内挿バンド計算), ¶m_visualization (実空間ワニエ関数計算)]。以下各ネームリストについて説明する。

3.1.1 ¶m_wannier

ネームリスト ¶m_wannier はワニエ関数計算のための変数リストである。変数は自由に並べて良い。ネームリストで定義される変数は定義しない場合はデフォルト値にセットされる。左列がプログラムに読み込ませる変数, 右の ! 以降が変数の説明である。説明部に n があるものが変数指定を必ず必要とするパラメータで, 入力をお忘れした場合エラーとなる。o となっているものはオプションなパラメータで, 入力を省いた場合デフォルト値が設定される。

```
&param_wannier
N_wannier=9,                !n: 求めたいワニエ軌道の数
Lower_energy_window=-10.0d0, !n: エネルギーウィンドウ下限 (eV)
Upper_energy_window= 35.0d0, !n: エネルギーウィンドウ上限 (eV)
N_initial_guess=9,          !n: 初期ガウシアンの個数
icell=0,                    !o: 結晶格子情報
EPS_SPILLAGE=1.0d-4,         !o: こぼれ汎関数最小化における収束閾値 (au)
EPS_SPREAD=1.0d-6,          !o: 広がり汎関数最小化における収束閾値 (au)
DAMP=0.1d0,                 !o: こぼれ汎関数最小化における収束因子
MAX_STEP_LENGTH=4.0d0,      !o: 広がり汎関数最小化におけるステップ長
set_inner_window=F,         !o: インナーウィンドウの設定有無 (下参照)
Lower_inner_window=10.0d0,   !o: エネルギーインナーウィンドウ下限 (eV)
Upper_inner_window=35.0d0,   !o: エネルギーインナーウィンドウ上限 (eV)
flg_initial_guess_direct=0,  !o: 初期ガウシアンを表示する座標系 (下参照)
flg_BMAT=0/                 !o: 初期ガウシアン of 線形結合係数の読み込み (下参照)
s  0.2d0 0.0d0 0.0d0 0.0d0 !n: 軌道型, 軌道指数, ガウシアン位置 (格子座標) s@A1
px  0.2d0 0.0d0 0.0d0 0.0d0 !n: px@A1
py  0.2d0 0.0d0 0.0d0 0.0d0 !n: py@A1
pz  0.2d0 0.0d0 0.0d0 0.0d0 !n: pz@A1
dxy 0.2d0 0.0d0 0.0d0 0.0d0 !n: dxy@A1
dyz 0.2d0 0.0d0 0.0d0 0.0d0 !n: dyz@A1
dz2 0.2d0 0.0d0 0.0d0 0.0d0 !n: dz2@A1
dzx 0.2d0 0.0d0 0.0d0 0.0d0 !n: dzx@A1
dx2 0.2d0 0.0d0 0.0d0 0.0d0 !n: dx2-y2@A1
```


¶m_wannier では `N_wannier`, `Lower_energy_window`, `Upper_energy_window`, `N_initial_guess` の四つが必要パラメータである。 `Lower_energy_window` と `Upper_energy_window` はエネルギーウィンドウを指定する変数であり、このエネルギー範囲からワニエ関数を `N_wannier` 個作成する。図 1 にバンド分散とエネルギーウィンドウの関係を示す。エネルギーウィンドウは下限が -10 eV, 上限が $+35$ eV に設定された。このウィンドウ内のバンドから 9 つのワニエ軌道 (`N_wannier=9`) が作られる。

`N_initial_guess` が初期ガウシアン個数を指定する変数であり、この計算では 9 つ用意する (`N_initial_guess=9`)。このネームリストに引き続いて入力する初期ガウシアン情報も必要入力である。具体的には「s 0.2d0 0.0d0 0.0d0 0.0d0」から「dx2 0.2d0 0.0d0 0.0d0 0.0d0」の部分がこれに相当する。この計算では初期ゲスとして Al サイト上に置かれた s, px, py, pz, dxy, dyz, dz2, dzx, dx2-y2 型の 9 つのガウス関数を用いる。ここで x, y, z はブラベー格子を表現する際にとった直交座標系である。ここまでが必要パラメータである。

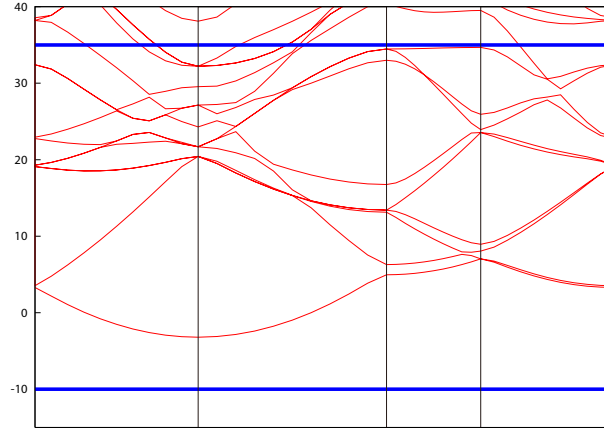


図 1: Al のバンド構造。エネルギーウィンドウは、下限が -10 eV (`Lower_energy_window=-10.0`), 上限が $+35$ eV (`Upper_energy_window=35.0`) に設定された。このウィンドウ内のバンドから 9 つのワニエ軌道 (`N_wannier=9`) が 9 つの初期ガウシアン (`N_initial_guess=9`) からつくられる。

`icell` は結晶格子の指定変数である。 `icell=0` (デフォルト) の場合は自動サーチである。 `EPS_SPILLAGE`, `EPS_SPREAD`, `DAMP`, `MAX_STEP_LENGTH` は計算の収束に関わるパラメータである。与えられている値がデフォルト値である。

`set_inner_window` はインナーエネルギーウィンドウを指定するパラメータであり、設定する場合は `Lower_inner_window` と `Upper_inner_window` を指定する。これらのパラメータを指定すると、インナーエネルギーウィンドウ領域内のバンドは必ずオリジナルバンドと一致するようにワニエ関数が構築される。この計算では `set_inner_window=F` なのでインナーウィンドウを設定せず、ゆえに `Lower_inner_window` と `Upper_inner_window` を設定しても計算の中では用いられない。これらのパラメータについては La_2CuO_4 計算 (第 7 章) で詳述する。

`flg_initial_guess_direc` はガウス関数を表示させる x, y, z を直交座標ではなく自ら設定した局所座標に変更する場合の設定変数である (この場合は `flg_initial_guess_direc=1` とする)。これについては第 8 章で TiO_2 のインプットを例にとり説明する。

`flg_BMAT` はワニエ関数の初期ゲスをつくる手続きに関する変数である。 `flg_BMAT=1` であれば、ガウシアンを線形結合して以下のように初期ゲスをつくる。

$$w_j^{in}(\mathbf{r}) = \sum_i B_{ij} g_i(\mathbf{r})$$

本計算のように flg_BMAT=0 であれば初期ガウシアン線形結合をしないことを意味し、B 行列は単位行列である。flg_BMAT=1 を設定した場合、重ね合わせの係数 B_{ij} は初期ガウシアン情報に引き続いて読み込まれる。これについては第 9 章で Si のインプットを参照して説明する。

3.1.2 ¶m_interpolation

ネームリスト ¶m_interpolation はワニエ内挿バンドを描画する際の入力変数のリストを納める。内挿バンド計算は必ず実施されるのでネームリスト内変数は必須入力パラメータであり、指定を忘れた場合はエラーとなる。以降の部分に変数説明を纏める。

```
&param_interpolation
N_sym_points=5,      !n: 計算ラインを構成する対称 k 点数
Ndiv=40/              !o: 対称 k 点間の分割数 (オプションパラメータ)
0.500 0.500 0.500 !n: 対称 k 点; SK_sym_pts(1:3,1); L
0.000 0.000 0.000 !n: 対称 k 点; SK_sym_pts(1:3,2); G
0.500 0.000 0.500 !n: 対称 k 点; SK_sym_pts(1:3,3); X
0.500 0.250 0.750 !n: 対称 k 点; SK_sym_pts(1:3,4); W
0.500 0.500 0.500 !n: 対称 k 点; SK_sym_pts(1:3,5); L
```

このネームリストではワニエ内挿バンドを描く際のブリルアンゾーン内の対称 k 点の個数 $N_{\text{sym_points}}$ を指定し、ネームリスト下にその対称 k 点の座標 (逆格子座標系での成分) を与える。ふたつの対称 k 点をグリッド分割する際の分割数 N_{div} はオプションパラメータでありデフォルト値は 40 である。バンド計算にて求めたオリジナルバンド図とワニエ内挿バンド図を比較して、後者が良く前者を再現できているかチェックするとよい。

3.1.3 ¶m_visualization

ネームリスト ¶m_visualization は実空間ワニエ関数を描画するための入力変数リストであり、計算終了後に可視化ソフト VESTA (<http://jp-minerals.org/vesta/jp/>) で描画させるための実空間ワニエ関数ファイルが得られる。

```
&param_visualization
flg_vis_wannier=1,!o: 実空間ワニエ関数を計算 (しない: 0, する:1)
N_write_wannier=9,!o: 求めたい実空間ワニエ関数の数
ix_vis_min=-1,      !o: ワニエ関数の描画範囲 (a1 方向始点格子)
ix_vis_max= 1,      !o: ワニエ関数の描画範囲 (a1 方向終点格子)
iy_vis_min=-1,      !o: ワニエ関数の描画範囲 (a2 方向始点格子)
iy_vis_max= 1,      !o: ワニエ関数の描画範囲 (a2 方向終点格子)
iz_vis_min=-1,      !o: ワニエ関数の描画範囲 (a3 方向始点格子)
iz_vis_max= 1/      !o: ワニエ関数の描画範囲 (a3 方向終点格子)
```

実空間ワニエ関数を計算したい場合は flg_vis_wannier=1 とセットする。デフォルトは 0 であり計算を実施しない。N_write_wannier は求めたい実空間ワニエ関数の個数で、デフォルトでは N_{wannier} に設定される。 N_{wannier} より少ない数を指定することもでき、この場合はこのネームリストの下に描画したいワニエ関数の番号を $N_{\text{write_wannier}}$ 個だけ記述する (第 13 章参照)。

3.2 用意すべきバンド計算データファイル

計算に必要なファイルは2章で生成したファイルである.

-
1. `./dir-wfn/dat.bandcalc` (バンド計算情報)
 2. `./dir-wfn/dat.sample-k` (サンプル k 点)
 3. `./dir-wfn/dat.nkm` (展開 G 数)
 4. `./dir-wfn/dat.eigenvalue` (エネルギー固有値)
 5. `./dir-wfn/dat.kg` (逆格子ベクトル)
 6. `./dir-wfn/dat.wfn` (波動関数)
 7. `./dir-wfn/dat.lattice` (基本並進ベクトル)
 8. `./dir-wfn/dat.symmetry` (対称性)
-

3.3 ジョブの実行

ジョブは以下のように実行する. `sh` 系での入力コマンドは以下である.

```
> export OMP_NUM_THREADS=16
> export MKL_NUM_THREADS=16
> ./calc_wannier < input.in > LOG.wannier
```

`calc_wannier` はワニエ関数計算実行ファイルである. カレントディレクトリに置かれた `input.in` と `./dir-wfn` 内に置かれたファイル群のもとジョブは実行される. このジョブは `openmp` ジョブである (16 スレッドを用いる). 計算の標準出力は `LOG.wannier` に出力される.

3.4 計算結果

ジョブが実行されると計算ディレクトリの下に `dir-wan` がつくられ, その中に計算結果がファイルに出力される. 出力ファイルは以下の通りである.

-
1. `./dir-wan/dat.h_mat_r` (実空間トランスファー行列)
 2. `./dir-wan/dat.iband` (ワニエ内挿バンド)
 3. `./dir-wan/dat.umat` (ユニタリー行列)
 4. `./dir-wan/dat.wan` (ワニエ関数を平面波展開したときの展開係数)
 5. `./dir-wan/dat.wan-center` (ワニエ関数の中心位置)
 6. `./dir-wan/dat.ns-nb` (ワニエ関数計算で考慮したバンド情報)
 7. `./dir-wan/dat.wan-realspace-001.grd-dat.wan-realspace-009.grd` (実空間ワニエ関数)
-

8. ./dir-wan/dat.frmsf (フェルミ面データ)

`dat.h_mat_r` は実空間トランスファー行列でありワニエ軌道による Kohn-Sham ハミルトニアンの行列要素 $\langle w_{i0} | h_{KS} | w_{jR} \rangle$ を納める。格子 \mathbf{R} の範囲は a_1 方向に沿って $-\text{Na}1:\text{Na}1$, a_2 方向に沿って $-\text{Na}2:\text{Na}2$, a_3 方向に沿って $-\text{Na}3:\text{Na}3$ である ($\text{Na}1$, $\text{Na}2$, $\text{Na}3$ の値はバンド計算のサンプル k 点数によって自動的に決定される)。

このプログラムでは格子 \mathbf{R} の i 番目のワニエ関数は以下のように定義される。

$$w_{i\mathbf{R}}(\mathbf{r}) = \frac{1}{\sqrt{N}} \sum_{\mathbf{k}} e^{-i\mathbf{k}\mathbf{R}} \sum_{\alpha=N_s(\mathbf{k})}^{N_b(\mathbf{k})} U_{\alpha i}^{\mathbf{k}} \psi_{\alpha\mathbf{k}}(\mathbf{r}), \quad (1)$$

ここで N は Monkhorst-Pack メッシュ k 点の総数である。 $N_s(\mathbf{k})$ および $N_b(\mathbf{k})$ は、ワニエ関数をつくるために用いたバンドの始点の番号および個数である (プログラム内で自動算出される)。 $N_s(\mathbf{k})$, $N_b(\mathbf{k})$ を納めたものが `dat.ns-nb`, 変換行列 $\{U_{\alpha i}^{\mathbf{k}}\}$ を納めたものが `dat.umat` である。これらの量は k 点に依存した値であることに注意されたい。 $\psi_{\alpha\mathbf{k}}(\mathbf{r})$ はブロッホ波動関数であり以下のように定義される。

$$\psi_{\alpha\mathbf{k}}(\mathbf{r}) = \frac{1}{\sqrt{N}} \sum_{\mathbf{G}} c_{\mathbf{k}+\mathbf{G},\alpha} \frac{1}{\sqrt{\Omega}} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}, \quad (2)$$

ここで Ω は計算セルの体積である。ワニエ関数 $w_{i\mathbf{R}}(\mathbf{r})$ とブロッホ関数 $\psi_{\alpha\mathbf{k}}(\mathbf{r})$ はともに全空間 (結晶体積の体積分) で規格化されている。いまホームセル ($\mathbf{R}=0$) のワニエ関数は (2) を (1) に代入すると以下のように書かれる。

$$w_{i0}(\mathbf{r}) = \frac{1}{N} \sum_{\mathbf{k}} \sum_{\mathbf{G}} \tilde{c}_{\mathbf{k}+\mathbf{G},i} \frac{1}{\sqrt{\Omega}} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}, \quad (3)$$

ここで,

$$\tilde{c}_{\mathbf{k}+\mathbf{G},i} = \sum_{\alpha=N_s(\mathbf{k})}^{N_b(\mathbf{k})} U_{\alpha i}^{\mathbf{k}} c_{\mathbf{k}+\mathbf{G},\alpha}, \quad (4)$$

$\{w_{i0}(\mathbf{r})\}$ を納めたものが `dat.wan-realspace-@@@.grd` ($@@@$ は出力ワニエ関数の番号 i), $\{\tilde{c}_{i\mathbf{k}+\mathbf{G}}\}$ を納めたものが `dat.wan` である。

`dat.iband` はワニエ内挿バンドである。gnuplot にて以下のようにプロットする (図 2)。

```
gnuplot> plot 'dir-wan/dat.iband' w line
```

`dat.wan-realspace-@@@.grd` は実空間ワニエ関数であり VESTA の読み込み形式で出力される。図 3 が VESTA での出力結果であるが、`dat.wan-realspace-@@@.grd` をそのまま VESTA の描画ウィンドウにドラッグするだけでこの図が作成できる。

`dat.frmsf` はフェルミ面データでありフェルミ面描画ソフト FermiSurfer の読み込み形式で出力される。図 4 が FermiSurfer での出力結果であるが、FermiSurfer がインストールされていれば `dat.frmsf` をクリックするだけでこの図が作成できる。

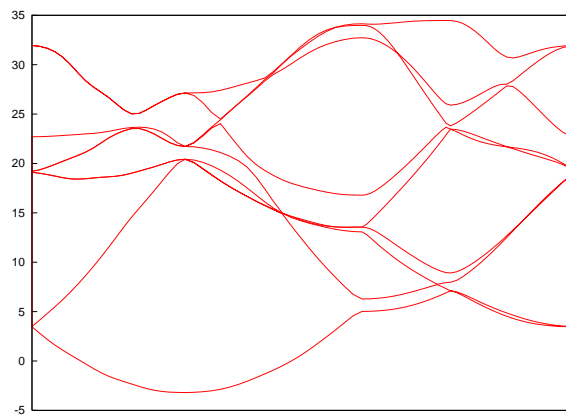


図 2: ワニエ内挿バンド. 縦軸の単位は eV.

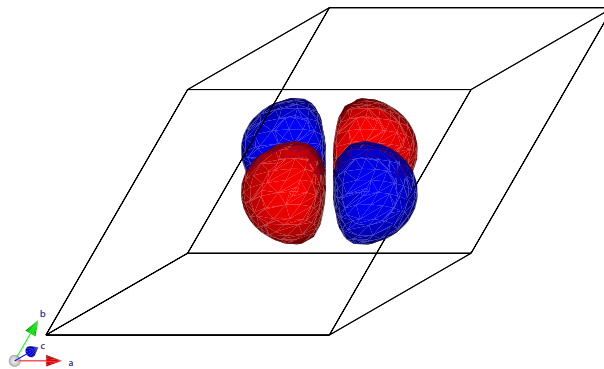


図 3: 実空間ワニエ関数. 第 9 番目の $d_{x^2-y^2}$ ワニエ関数 (dat.wan-realspace-009.grd) を出力している.

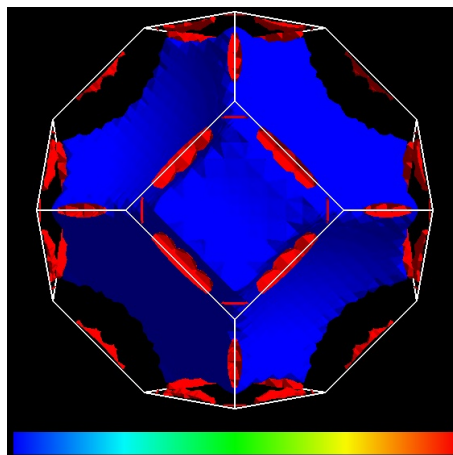


図 4: フェルミ面描画ソフト FermiSurfer によるフェルミ面. この計算ではサンプル k 点は $21 \times 21 \times 21$ の密なものを用いている.

4 chiqw

4.1 インプットファイル (¶m_chiqw)

以下が AI の誘電関数計算を実施するためのインプットファイル `input.in` である。¶m_chiqw がネームリストのグループ名であり、この行以下に変数が並ぶ。左が変数、右が変数の簡単な説明である。¶m_chiqw の場合すべての変数がオプションパラメータであり (説明部に o), 変数値はデフォルト設定されるので、ユーザーは入力を省略できる。下の例においてはデフォルト値が提示されている。

¶m_chiqw	
Ecut_for_eps=3.6d0,	!o: 分極関数のカットオフ (Ry)
Num_freq_grid=70,	!o: 計算周波数の総数
N_CALC_BAND=50,	!o: 分極計算で考慮されるバンドの総数
MPI_num_proc_per_qcomm=6,	!o: コミュニティ当たりの MPI の総数
MPI_num_qcomm=1,	!o: コミュニティの総数
MPI_io_rank=0,	!o: 標準出力させる MPI の番号
shift_ef=0.0d0,	!o: 人為的なフェルミ準位のシフト値 (eV)
Max_excitation_energy=200.0d0,	!o: 計算で考慮する最大励起エネルギー (eV)
Green_func_delt=0.1d0,	!o: テトラヘドロン計算でのスメアリング値 (eV)
ttrhdrn_dmna=1.0d-3,	!o: テトラヘドロン計算での縮退判定パラメータ 1 (eV)
ttrhdrn_dmnr=1.0d-3,	!o: テトラヘドロン計算での縮退判定パラメータ 2 (eV)
flg_cRPA=0,	!o: 制限 RPA オプション (0:通常 RPA, 1:制限 RPA)
flg_calc_type=0	!o: 0:all-q-calc, 1:gamma-only, 2:バルクジョブ

Ecut_for_eps は分極関数のカットオフエネルギー (Ry 単位) であり、デフォルトではバンド計算で用いた波動関数のカットオフエネルギーの 1/10 にセットされる (今の場合波動関数カットオフが 36 Ry なので分極関数カットオフを 3.6 Ry としている)。Num_freq_grid は分極関数の計算周波数総数であり、デフォルトでは 70 点のロググリッドが生成される。N_CALC_BAND は分極計算で考慮するバンド数であり、デフォルトではバンド計算で計算されたバンド数が設定される。このバンド数から、最大励起エネルギー E_{max} が決まる。計算周波数は分極関数のローレンツテイルを考慮して $2E_{max}$ までとられる。

MPI_num_proc_per_qcomm, MPI_num_qcomm, MPI_io_rank は MPI 計算のためのパラメータであり、上記の設定値の場合、6 MPI 計算が想定され、出力はマスター MPI が担当する。Green_func_delt, ttrhdrn_dmna, ttrhdrn_dmnr はテトラヘドロン計算のためのパラメータであり、すべてデフォルト値が用意されている。shift_ef は人為的にフェルミ準位をシフトさせて chiqw 計算を実施したい場合に設定する (単位は eV)。Max_excitation_energy は考慮する励起エネルギーを制限したい場合に設定する (単位は eV)。

flg_cRPA は制限 RPA 計算を行うか否かを指定する変数である。デフォルトでは通常 RPA 計算を行うので flg_cRPA=0 である。制限 RPA 計算を行う場合 (flg_cRPA=1), 二つの方針がある。ひとつは排除分極をワニエ関数によって規定するもの、もうひとつはエネルギーウィンドウを指定して排除分極を規定するものである。デフォルトでは前者を採用する (後者のオプションを用いたいユーザは連絡されたい)。前者の制限 RPA 計算の場合 chiqw 計算の前に wannier 計算 (第 3 章) を実行しておかねばならない。制限 RPA 計算については、後述の La_2CuO_4 計算 (第 7 章) の中で詳述する。

また、計算 q 点を指定して計算するモードが用意されている (`flg_calc_type=2`). 詳細については第 12 章で述べる.

4.2 用意すべきバンド計算データファイル

計算に必要なファイルは 2 章で生成した 8 つのファイルである.

-
1. `./dir-wfn/dat.bandcalc` (バンド計算情報)
 2. `./dir-wfn/dat.sample-k` (サンプル k 点)
 3. `./dir-wfn/dat.nkm` (展開 G 数)
 4. `./dir-wfn/dat.eigenvalue` (エネルギー固有値)
 5. `./dir-wfn/dat.kg` (逆格子ベクトル)
 6. `./dir-wfn/dat.wfn` (波動関数)
 7. `./dir-wfn/dat.lattice` (基本並進ベクトル)
 8. `./dir-wfn/dat.symmetry` (対称性)
-

`chiqw` 計算のためのバンド計算データは必ず Monkhorst-Pack メッシュ上に切られた k 点の下でのデータでなくてはならない. 既約 k 点に対して求められたデータ `dat.sample-k`, `dat.nkm`, `dat.eigenvalue`, `dat.kg`, `dat.wfn` は `dat.symmetry` の対称性情報から Monkhorst-Pack メッシュ上の全 k 点データに変換される.

4.3 ジョブの実行

ジョブは以下のように実行する. `sh` 系での入力コマンドは以下である.

```
> export OMP_NUM_THREADS=16
> export MKL_NUM_THREADS=16
> mpirun -np 6 ./calc_chiqw < input.in > LOG.chiqw
```

`calc_chiqw` は誘電関数計算実行ファイルである. `calc_chiqw` は `RESPACK/src/chiqw` 内でコンパイルして作成する (1.4 節参照). カレントディレクトリに置かれた `input.in` と `./dir-wfn` 内に置かれたファイル群のもとジョブは実行される. このジョブはハイブリッド並列ジョブである (各 MPI プロセスが 16 スレッドからなり, 6 MPI プロセスにて実行される. `¶m_chiqw` の `MPI_num_proc_per_qcomm`, `MPI_num_qcomm` を省略すると `MPI_num_proc_per_qcomm=6`, `MPI_num_qcomm=1` と自動設定される). `input.in` は標準入力として読み込まれ, 計算の標準出力 (標準出力はデフォルトでは MPI のマスターが担当する) は `LOG.chiqw` に出力される.

4.4 計算結果

ジョブが実行されると計算ディレクトリの下に `dir-eps` がつくられ、そこに周波数グリッド情報 (`dat.wgrid`) と `chiqw` 計算の基本情報 (`dat.chiqwcalc`) が出力される。さらにその下に q 点番号を付されたディレクトリ `qxxx` が生成され、その中に計算結果がファイル出力される。計算 q 点の総数 (すなわち生成される `qxxx` ディレクトリの総数) は既約 q 点の総数であり、今回の AI 計算では 16 である。各ディレクトリ `qxxx` 内のファイルに関して、例えば 16 番目の q 点の場合、ディレクトリ `q016` の中に以下の出力ファイルが生成される。

-
1. `./dir-eps/dat.wgrid` (周波数グリッド)
 2. `./dir-eps/dat.chiqwcalc` (`chiqw` 計算情報)
 3. `./dir-eps/q016/dat.log.400` (計算終了フラグ)
 4. `./dir-eps/q016/dat.epsqw.600` (逆誘電関数データ)
 5. `./dir-eps/q016/dat.epsqw.600001-dat.epsqw.600070` (各周波数の逆誘電行列対角項)
-

`dat.log.400` は計算終了フラグである。`dat.epsqw.600` には逆誘電行列 $\{\epsilon_{\mathbf{G}\mathbf{G}'}^{-1}(\mathbf{q}, \omega)\}$ がバイナリ形式で収容される。`dat.epsqw.600000` 番台のファイルは周波数グリッドの数だけ逆誘電行列対角項データを納める。ファイルの第一列目が $|\mathbf{q} + \mathbf{G}|$ (単位は bohr^{-1})、第二列目が $\epsilon_{\mathbf{G}\mathbf{G}}^{-1}(\mathbf{q}, \omega_1)$ の実部、第三列目が虚部である。今回のジョブ (`Num_freq_grid=70`) の場合は `dat.epsqw.600001` から `dat.epsqw.600070` まで生成される。たとえば `dat.epsqw.600001` には周波数グリッドの第 1 番目 ω_1 の誘電行列対角項 $\epsilon_{\mathbf{G}\mathbf{G}}^{-1}(\mathbf{q}, \omega_1)$ が収容され、`dat.epsqw.600070` には周波数グリッドの第 70 番目 ω_{70} の誘電行列対角項 $\epsilon_{\mathbf{G}\mathbf{G}}^{-1}(\mathbf{q}, \omega_{70})$ が収容される。

全既約 q 点の逆誘電行列対角データ $\epsilon_{\mathbf{G}\mathbf{G}}^{-1}(\mathbf{q}, \omega)$ をプロットするには以下のように行う。ここでは $\omega_1 = 0$ の結果についてプロットする。

```
> cat dir-eps/q0*/dat.epsqw.600001 > epsq0
> gnuplot
gnuplot> plot 'epsq0' u 1:2, 'epsq0' u 1:3
```

電子エネルギー欠損スペクトル $[-\text{Im}\epsilon_{00}^{-1}(0, \omega)]$ の出力については以下の手続きを行う。まず `grep` コマンドで `dat.epsqw.600*` 内の文字列 `'0000000'` ($\mathbf{q}=\mathbf{0}$, $\mathbf{G}=\mathbf{0}$ の成分) を抜き出し、次に `paste` コマンドで周波数データ `dat.wgrid` と結合して `eels` を作成する。このファイルの第 6 列をプロットすると図 6 のようなスペクトルが描画される (第 5 列は実部である)。具体的手続きは以下のとおりである。

```
> grep '0000000' dir-eps/q0*/dat.epsqw.600* > dat
> sed -e '1,1d' dir-eps/dat.wgrid > grd (先頭行を省いた周波数データ生成)
> paste grd dat > eels
> gnuplot
gnuplot> plot 'eels' u 1:5, 'eels' u 1:6
```

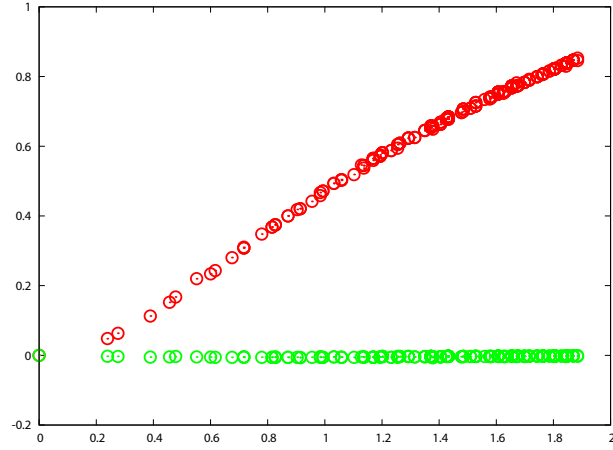


図 5: 全既約 q 点の逆誘電行列対角データ $\epsilon_{\mathbf{GG}}^{-1}(\mathbf{q}, \omega_1)$ の結果 (赤が実部, 緑が虚部). 横軸は $|\mathbf{q} + \mathbf{G}|$ (bohr $^{-1}$) である. この結果では $\omega_1 = 0$ の結果がプロットされている.

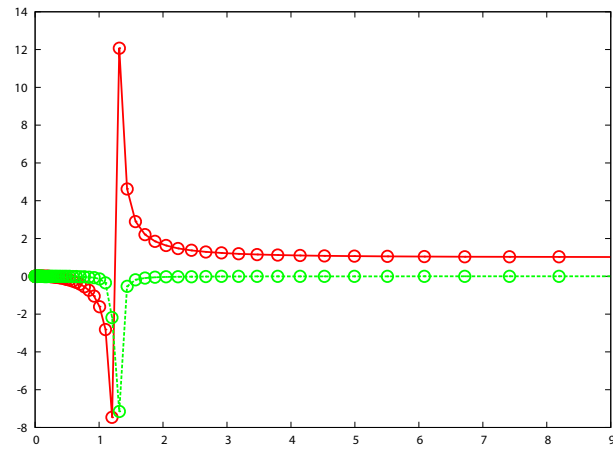


図 6: eels $[\epsilon_{00}^{-1}(\Gamma, \omega)]$ の結果 (赤が実部, 緑が虚部). 横軸は ω (hartree) である. 約 1 hartree の位置に大きなピーク (緑) が現れているが, これが Al のプラズモン吸収による強度である.

5 calc_w3d および calc_j3d

直接積分と交換積分の評価は各々別実行形式ファイルにて実行されるが (直接積分計算が calc_w3d, 交換積分計算が calc_j3d), これらは共通のインプットとファイルで実施できる. 積分評価は誘電関数計算とワニエ関数計算を終えた後に行わなければならない.

5.1 インプットファイル (¶m_calc_int)

以下が相互作用積分計算 (calc_w3d, calc_j3d) を実施するためのインプットファイル input.in である. ネームリスト ¶m_calc_int 内のすべての変数に対してデフォルト値が用意されているのでユーザーは原則的に入力手続きをする必要はない.

```
&param_calc_int
calc_ifreq=1, !o: 計算出力させたい周波数
ix_int_min=0, !o: 交換積分の計算範囲 (a1 方向始点格子) (calc_j3d のみ)
ix_int_max=0, !o: 交換積分の計算範囲 (a1 方向終点格子) (calc_j3d のみ)
iy_int_min=0, !o: 交換積分の計算範囲 (a2 方向始点格子) (calc_j3d のみ)
iy_int_max=0, !o: 交換積分の計算範囲 (a2 方向終点格子) (calc_j3d のみ)
iz_int_min=0, !o: 交換積分の計算範囲 (a3 方向始点格子) (calc_j3d のみ)
iz_int_max=0, !o: 交換積分の計算範囲 (a3 方向終点格子) (calc_j3d のみ)
```

相互作用計算を実行する際に重要なことは, この計算に入る前に既約 q 点全ての逆誘電行列を計算しておくことである. 既約 q 点でのデータが不十分では Monkhorst-Pack メッシュ上の全データを生成できないからである. RESPACK では, ./dir-eps/qxxx 内に計算終了フラグ dat.log.400 の有無を調べ, 無い場合は計算を止める.

5.2 用意すべきバンド計算データファイル

計算は以下のファイルを用いる.

-
1. ./dir-wfn/dat.bandcalc (バンド計算情報)
 2. ./dir-wfn/dat.sample-k (サンプル k 点)
 3. ./dir-wfn/dat.nkm (展開 G 数)
 4. ./dir-wfn/dat.kg (逆格子ベクトル)
 5. ./dir-wfn/dat.lattice (基本並進ベクトル)
 6. ./dir-wfn/dat.symmetry (対称性)
 7. ./dir-eps/dat.wgrid (周波数グリッド)
 8. ./dir-eps/dat.chiqwcalc (chiqw 計算情報)
 9. ./dir-eps/qxxx/dat.epsqw.600 (誘電関数データ)
 10. ./dir-wan/dat.wan (ワニエ関数を平面波展開したときの展開係数)

11. ./dir-wan/dat.wan-center (ワニエ関数の中心位置)

5.3 ジョブの実行

ジョブは以下のように実行する. sh 系での入力コマンドは以下である.

```
> export OMP_NUM_THREADS=16
> export MKL_NUM_THREADS=16
> ./calc_w3d (or ./calc_j3d) < input.in > LOG.W3d (or LOG.J3d)
```

calc_w3d あるいは calc_j3d は相互作用計算実行ファイルである. カレントディレクトリに置かれた input.in とディレクトリ ./dir-wfn, ./dir-eps, ./dir-wan 内に置かれたファイルのもとジョブは実行される. このジョブは openmp ジョブである (16 スレッドを用いる). calc_w3d の標準出力は LOG.W3d に, calc_j3d の標準出力は LOG.J3d に出力される.

5.4 計算結果

ジョブが実行されると計算ディレクトリの下に dir-intW および dir-intJ がつくられ, その中に計算結果がファイルとして出力される. dir-intW の中に calc_w3d の結果が, dir-intJ の中に calc_j3d の結果が収容される. 出力ファイルは以下の通りである.

-
1. ./dir-intW/dat.Wmat [遮蔽直接積分, 式 (5)]
 2. ./dir-intW/dat.Vmat [裸の直接積分, 式 (7)]
 3. ./dir-intW/dat.VvsR.001 - dat.VvsR.009 ($V(r)$)
 4. ./dir-intW/dat.WvsR.001 - dat.WvsR.009 ($W(r)$)
 5. ./dir-intW/dat.UvsE.001-001 - dat.UvsE.009-009 ($U(\omega)$)
 6. ./dir-intJ/dat.Jmat [遮蔽交換積分, 式 (6)]
 7. ./dir-intJ/dat.Xmat [裸の交換積分, 式 (8)]
 8. ./dir-intJ/dat.JvsE.001-001 - dat.JvsE.009-009 ($J(\omega)$)
-

遮蔽直接積分および遮蔽交換積分は以下のように定義される.

$$W_{ij}(\mathbf{R}, \omega) = \int_V d\mathbf{r} \int_V d\mathbf{r}' w_{i0}^*(\mathbf{r}) w_{i0}(\mathbf{r}) W(\mathbf{r}, \mathbf{r}', \omega) w_{j\mathbf{R}}^*(\mathbf{r}') w_{j\mathbf{R}}(\mathbf{r}'), \quad (5)$$

$$J_{ij}(\mathbf{R}, \omega) = \int_V d\mathbf{r} \int_V d\mathbf{r}' w_{i0}^*(\mathbf{r}) w_{j\mathbf{R}}(\mathbf{r}) W(\mathbf{r}, \mathbf{r}', \omega) w_{j\mathbf{R}}^*(\mathbf{r}') w_{i0}(\mathbf{r}'). \quad (6)$$

積分は結晶体積 V の体積分を意味する. これらの積分はホームセル ($\mathbf{R}=0$) の i 番目のワニエ関数とセル \mathbf{R} の j 番目のワニエ関数の間の遮蔽クーロン相互作用 $W(\mathbf{r}, \mathbf{r}', \omega)$ の行列要素として計算される.

$W(\mathbf{r}, \mathbf{r}', \omega)$ ではなく裸のクーロン相互作用 $v(\mathbf{r}, \mathbf{r}') = \frac{1}{|\mathbf{r} - \mathbf{r}'|}$ の場合は、裸の直接積分および交換積分である。

$$V_{ij}(\mathbf{R}) = \int_V d\mathbf{r} \int_V d\mathbf{r}' w_{i0}^*(\mathbf{r}) w_{i0}(\mathbf{r}) v(\mathbf{r}, \mathbf{r}') w_{j\mathbf{R}}^*(\mathbf{r}') w_{j\mathbf{R}}(\mathbf{r}'), \quad (7)$$

$$X_{ij}(\mathbf{R}) = \int_V d\mathbf{r} \int_V d\mathbf{r}' w_{i0}^*(\mathbf{r}) w_{j\mathbf{R}}(\mathbf{r}) v(\mathbf{r}, \mathbf{r}') w_{j\mathbf{R}}^*(\mathbf{r}') w_{i0}(\mathbf{r}). \quad (8)$$

dat.Wmat, dat.Jmat, dat.Vmat, dat.Xmat の計算データは $N_{\text{wannier}} \times N_{\text{wannier}}$ の行列としてセル \mathbf{R} ごとに出力される。直接積分はすべての \mathbf{R} に対して計算されるが、交換積分は計算が重いので、自らが指定した格子の範囲 $ix_intJ_min \sim iz_intJ_max$ でのみ計算される。

dat.WvsR.xxx は静的遮蔽直接積分 $[W_{ij}(\mathbf{R}, 0)]$ の距離依存性である。ホームセル ($\mathbf{R}=0$) のワニエ軌道 i と相互作用する全てのワニエ軌道との間の直接積分が計算される。デフォルトでは $\omega = 0$ が計算されるが (calc_ifreq=1), この変数を変えることで有限 ω の結果を出力できる。今回の Al 計算の場合 $N_{\text{wannier}}=9$ なので 9 つのファイルが出力される。遮蔽効果を見るために dat.VvsR.xxx を重ねてプロットするとよい。出力は以下のようにする (図 7 が出力結果):

```
gnuplot> plot 'dir-intW/dat.VvsR.001', 'dir-intW/dat.WvsR.001'
```

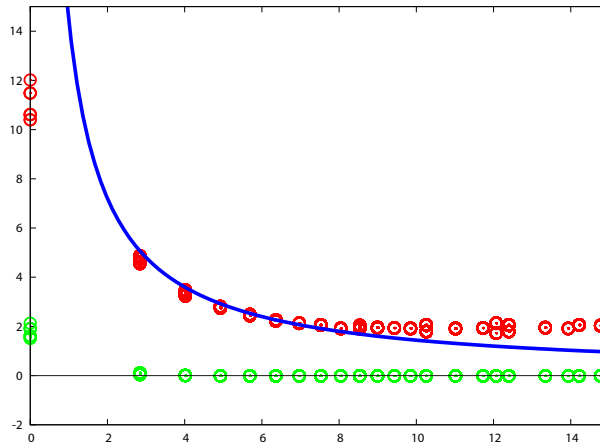


図 7: 裸直接積分 (赤) と静的有効直接積分 (緑) の距離依存性. 横軸はワニエ関数 w_{i0} と $w_{j\mathbf{R}}$ の間の距離 (単位はÅ), 縦軸の単位は eV である. この結果は $i=001$ のものである. 裸直接積分は $\frac{1}{R}$ で減衰し (青), 静的有効直接積分は金属遮蔽のためオンサイトでしか値を持っていない.

オンサイト遮蔽直接相互作用 $[W_{ii}(0, \omega)]$ の周波数依存性は以下のようにプロットする (図 8 が出力結果). この出力は $W_{11}(0, \omega)$ であるが, 本プログラムではワニエ軌道 i と j のすべてのペアに対して $W_{ij}(0, \omega)$ が計算される.

```
gnuplot> plot 'dir-intW/dat.UvsE.001-001' u 1:3, 'dir-intW/dat.UvsE.001-001' u 1:4
```

オンサイト遮蔽交換相互作用 $[J_{ij}(0, \omega)]$ の周波数依存性は以下のようにプロットする (図 9 が出力結果). この出力では $J_{12}(0, \omega)$ であるが, 本プログラムではワニエ軌道 i と j のすべてのペアに対して $J_{ij}(0, \omega)$ が計算される.

```
gnuplot> plot 'dir-intJ/dat.JvsE.001-002' u 1:3, 'dir-intJ/dat.JvsE.001-002' u 1:4
```

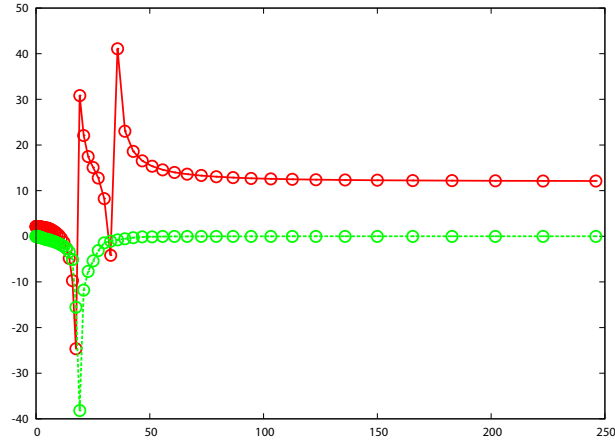


図 8: オンサイト遮蔽直接積分の周波数依存性. 赤が実部, 緑が虚部である. 横軸は周波数 ω (eV), 縦軸の単位は eV. 約 20 eV 付近にプラズモン励起に由来する極構造が見て取れる.

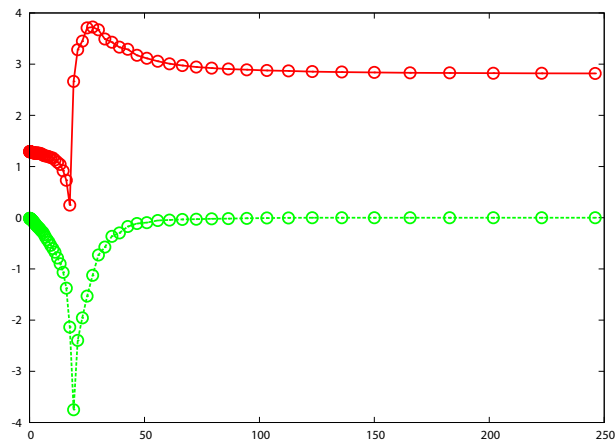


図 9: オンサイト遮蔽交換積分の周波数依存性. 赤が実部, 緑が虚部である. 横軸は周波数 ω (eV), 縦軸の単位は eV. この場合も約 20 eV 付近に極構造があるが, 全体的なエネルギースケールは直接積分の 1/10 程度である.

6 まとめ: AI 計算インプット

以上までをまとめて AI に対する RESPACK 計算のインプット全体を示す。以下が省略可能変数をすべて省略したインプットファイル `input.in` である。wannier 計算にだけ必要パラメータがあり、それ以外はすべてデフォルト設定であることが分かる。

```
&param_wannier
N_wannier=9, !n: 求めたいワニエ軌道の数
Lower_energy_window=-10.0d0, !n: エネルギーウィンドウ下限 (eV)
Upper_energy_window= 35.0d0, !n: エネルギーウィンドウ上限 (eV)
N_initial_guess=9/ !n: 初期ガウシアン個数
s 0.2d0 0.0d0 0.0d0 0.0d0 !n: 軌道型, 軌道指数, ガウシアン位置 (格子座標) s@A1
px 0.2d0 0.0d0 0.0d0 0.0d0 !n: px@A1
py 0.2d0 0.0d0 0.0d0 0.0d0 !n: py@A1
pz 0.2d0 0.0d0 0.0d0 0.0d0 !n: pz@A1
dxy 0.2d0 0.0d0 0.0d0 0.0d0 !n: dxy@A1
dyz 0.2d0 0.0d0 0.0d0 0.0d0 !n: dyz@A1
dz2 0.2d0 0.0d0 0.0d0 0.0d0 !n: dz2@A1
dzx 0.2d0 0.0d0 0.0d0 0.0d0 !n: dzx@A1
dx2 0.2d0 0.0d0 0.0d0 0.0d0 !n: dx2-y2@A1

&param_interpolation
N_sym_points=5/ !n: 計算ラインを構成する対称 k 点数
0.500 0.500 0.500 !n: 対称 k 点; SK_sym_pts(1:3,1); L
0.000 0.000 0.000 !n: 対称 k 点; SK_sym_pts(1:3,2); G
0.500 0.000 0.500 !n: 対称 k 点; SK_sym_pts(1:3,3); X
0.500 0.250 0.750 !n: 対称 k 点; SK_sym_pts(1:3,4); W
0.500 0.500 0.500 !n: 対称 k 点; SK_sym_pts(1:3,5); L

&param_visualization
/

&param_chiqw
/

&param_calc_int
/
```

このデモンストレーションでは計算時間短縮のため計算条件は甘くとられており、収束のためには条件を厳しくとる必要がある。ワニエ関数は主に k 点数と波動関数カットオフに依存する。誘電関数は非占有バンド数、誘電関数カットオフに依存する。AI 計算の場合、 k 点数 $10 \times 10 \times 10$ 、波動関数カットオフ 64 Ry、誘電関数カットオフ 10 Ry、非占有バンド数 50 程度にとるのが良いだろう。制限 RPA の場合、一般的には非占有バンド数はフェルミ準位より数 10 eV 程度上のバンドまでを含めるようにとるとよい。

7 La_2CuO_4 (制限 RPA を用いた相互作用評価)

前までで RESPACK を用いた相互作用評価の手続きを述べた。この章では別の例として、遷移金属酸化物 La_2CuO_4 の相互作用評価について述べる。とくに wannier 計算におけるインナーエネルギーウィンドウの設定や chiqw 計算における制限 RPA オプションについて記す。バンド計算 (xTAPP を使用) の条件であるが、 k 点サンプリングを $6 \times 6 \times 6$ (既約 k 点数 30), 波動関数カットオフを 81 Ry, 全バンド数を 60 (占有バンド数 26, 部分占有バンド数 1, 非占有バンド数 33) であり, フェルミ準位から約 15 eV 上までのバンドを包含) にとった。

この計算では制限 RPA 計算を行うのでネームリスト ¶m_chiqw の変数 flg_cRPA を 1 にセットする。計算を軽くするため, 分極関数カットオフを 3 Ry としている (デフォルトなら 8.1 Ry)。制限 RPA の場合, 排除分極の定義にワニエ関数を用いるので wannier 計算を chiqw 計算より先に実施しておく必要がある。ゆえに計算の順は wannier \rightarrow chiqw \rightarrow calc_w3d (calc_j3d) となる。以下が La_2CuO_4 (底心正方格子) の RESPACK 計算のインプット input.in である。このインプットでは wannier, chiqw, calc_w3d (calc_j3d) の各インプットがひとつにまとめられており, デフォルト設定パラメータはすべて省略されている。

```
&param_chiqw
Ecut_for_eps=3.0d0, !分極関数のカットオフ (Ry)
flg_cRPA=1/          !制限 RPA オプション (0:通常 RPA, 1:制限 RPA)

&param_wannier
N_wannier=1,          !求めたいワニエ軌道の数
Lower_energy_window=10.0d0, !エネルギーウィンドウ下限 (eV)
Upper_energy_window=15.6d0, !エネルギーウィンドウ上限 (eV)
N_initial_guess=1,    !初期ガウシアン個数
set_inner_window=T,   !インナーエネルギーウィンドウの設定有無
Lower_inner_window=12.60d0, !エネルギーインナーウィンドウ下限 (eV)
Upper_inner_window=13.40d0/ !エネルギーインナーウィンドウ上限 (eV)
dx2 0.2d0 0.0d0 0.0d0 0.0d0 !dx2-y2@Cu

&param_interpolation
N_sym_points=6/      !計算ラインを構成する対称 k 点数
0.000 0.000 0.000 !対称 k 点; G
0.000 0.000 0.500 !対称 k 点; X
0.250 0.250 0.250 !対称 k 点; P
0.000 0.500 0.000 !対称 k 点; N
0.000 0.000 0.000 !対称 k 点; G
0.500 0.500 -0.500 !対称 k 点; Z

&param_visualization
flg_vis_wannier=1/

&param_calc_int
/
```

この wannier 計算ではエネルギーウィンドウを Lower_energy_window=10.0 eV から

Upper_energy_window=15.6 eV に設定し, 銅原子の位置に中心をもつ $d_{x^2-y^2}$ ガウシアンを初期ゲスとしたワニエ関数をつくる (dx2 0.2d0 0.0d0 0.0d0 0.0d0). また, この計算ではインナーウィンドウが設定される (set_inner_window=T). インナーウィンドウは Lower_inner_window=12.6 eV から Upper_inner_window=13.4 eV に設定され, このインナーウィンドウ内のバンドをワニエ内挿バンドは正確に再現する. 通常インナーウィンドウは金属のフェルミ準位 (この計算では 12.97 eV) を挟むように設定することが多い.

得られたワニエ内挿バンドと実空間ワニエ関数を図 10 および 11 に示す. 図 12 および 13 に制限 RPA および 通常 (制限なし) RPA による静的逆誘電関数の波数依存性を示す. 図 14 および 15 に制限 RPA および 通常 (制限なし) RPA による電子エネルギー欠損スペクトル (eels) を示す. 図 16 に直接積分の距離依存性を示す. 赤が裸直接積分, 緑が制限 RPA 直接積分, 青が通常 (制限なし) RPA 直接積分である. 図 17 および 18 に制限 RPA および 通常 (制限なし) RPA によるオンサイト遮蔽直接積分の周波数依存性を示す.

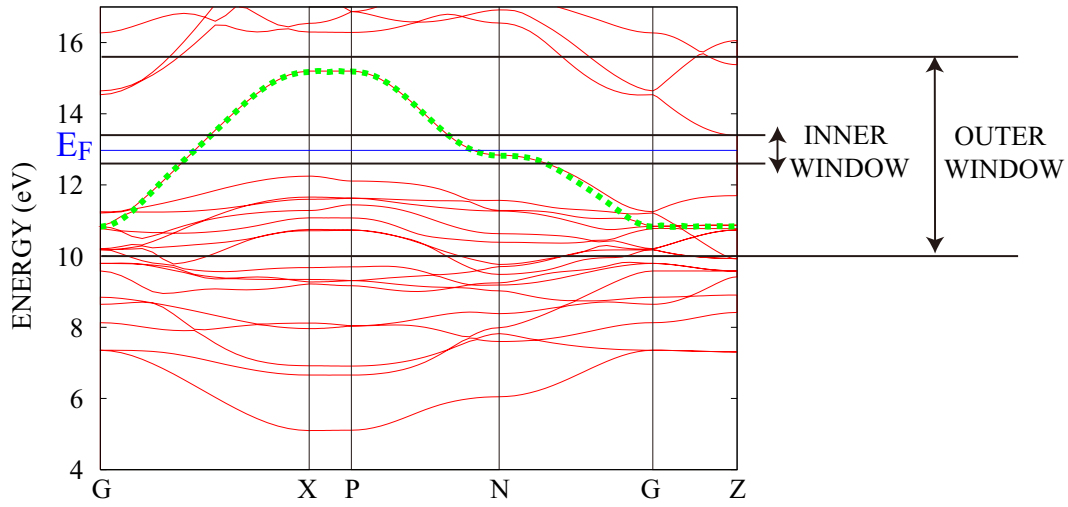


図 10: ワニエ内挿バンド. 縦軸の単位は eV. エネルギーウィンドウは下限が 10 eV (Lower_energy_window=10.0), 上限が 15.6 eV (Upper_energy_window=15.6) に設定された. このウィンドウ内のバンドから 1 つのワニエ軌道 (N_wannier=1) が 1 つの初期ガウシアン (N_initial_guess=1) からつくられた. インナーエネルギーウィンドウは Lower_inner_window=12.6 eV, Upper_inner_window=13.4 eV に設定された.

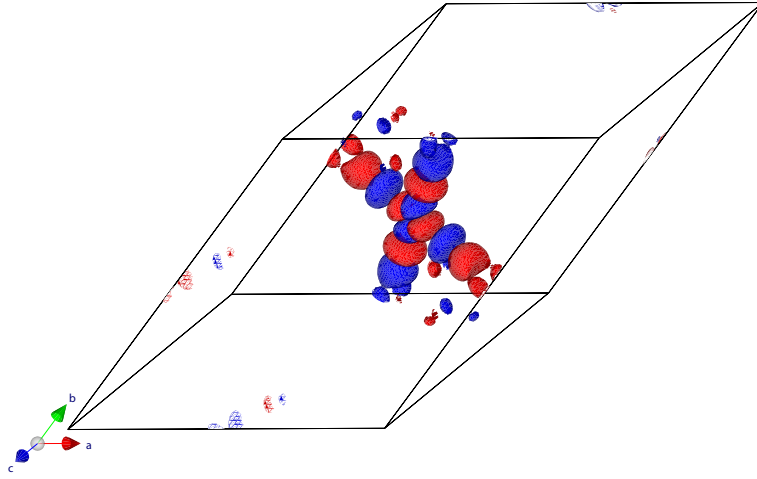


図 11: 実空間ワニ関数.

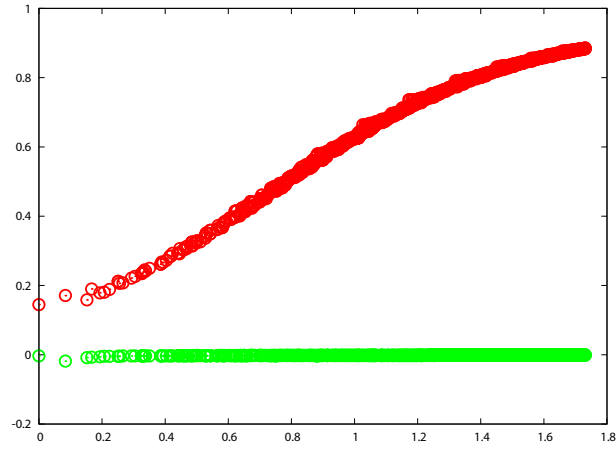


図 12: 制限 RPA 計算による全既約 q 点の逆誘電行列対角データ $\epsilon_{\mathbf{G}\mathbf{G}}^{-1}(\mathbf{q}, \omega_1)$ の結果 (赤が実部, 緑が虚部). 横軸は $|\mathbf{q} + \mathbf{G}|$ (bohr $^{-1}$) である. この結果では $\omega_1 = 0$ の結果がプロットされている.

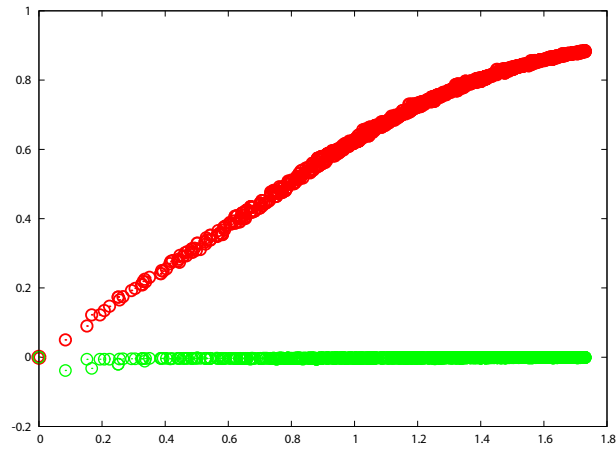


図 13: 通常 RPA (制限なし RPA) 計算による全既約 q 点の逆誘電行列対角データ $\epsilon_{\mathbf{G}\mathbf{G}}^{-1}(\mathbf{q}, \omega_1)$ の結果 (赤が実部, 緑が虚部). 横軸は $|\mathbf{q} + \mathbf{G}|$ (bohr $^{-1}$) である. この結果では $\omega_1 = 0$ の結果がプロットされている.

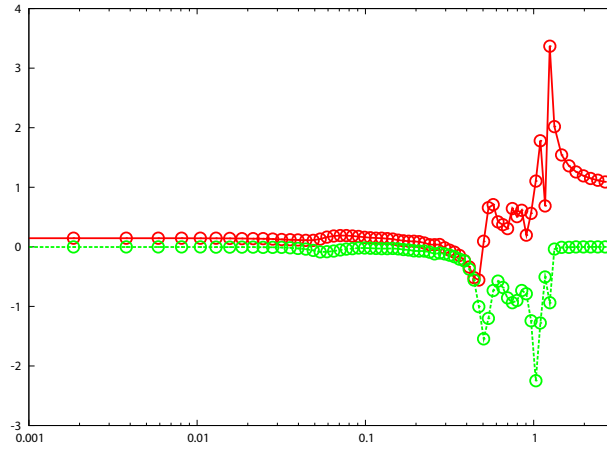


図 14: 制限 RPA 計算による eels $[\epsilon_{00}^{-1}(\Gamma, \omega)]$ の結果 (赤が実部, 緑が虚部). 横軸は周波数 ω (hartree) の対数である.

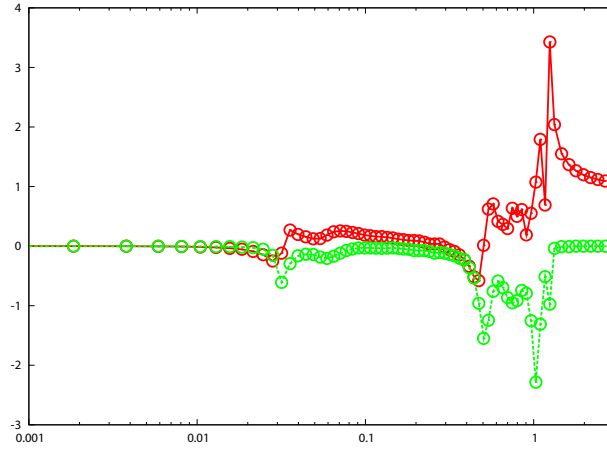


図 15: 通常 RPA (制限なし RPA) 計算による eels $[\epsilon_{00}^{-1}(\Gamma, \omega)]$ の結果 (赤が実部, 緑が虚部). 横軸は周波数 ω (hartree) の対数である.

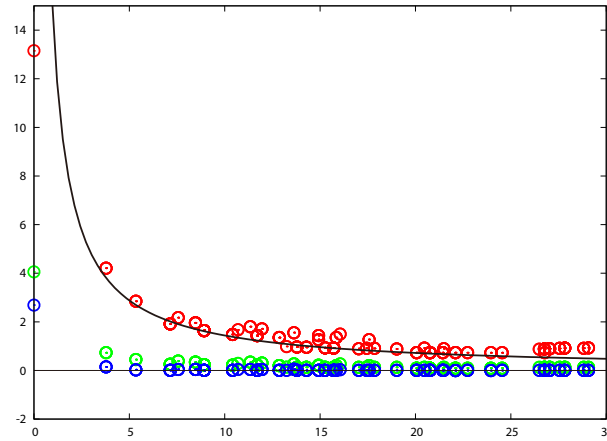


図 16: 直接積分の距離依存性 [裸直接積分 (赤), 制限 RPA 直接積分 (緑), 通常 RPA (制限なし RPA) 直接積分 (青)]. 横軸はワニエ関数 w_{i0} と w_{jR} の間の距離 (単位はÅ), 縦軸の単位は eV である. 裸直接積分は $\frac{1}{R}$ で減衰する. 制限 RPA 直接積分は有限の相互作用をオフサイトに持っている. 通常 RPA (制限なし RPA) 直接積分は金属遮蔽のためオンサイトでしか値を持っていない.

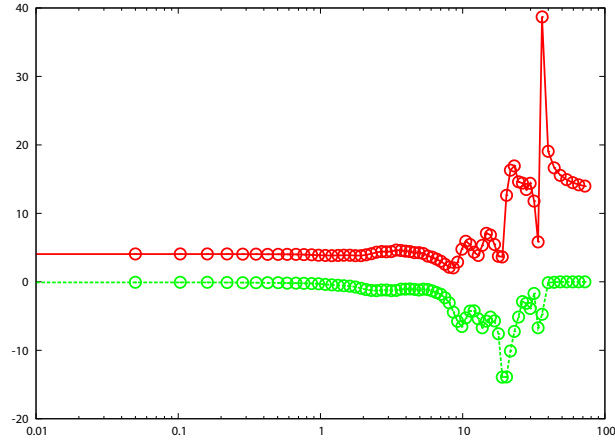


図 17: 制限 RPA 計算による オンサイト遮蔽直接積分の周波数依存性. 赤が実部, 緑が虚部である. 横軸は周波数 ω (eV) の対数である. 縦軸の単位は eV.

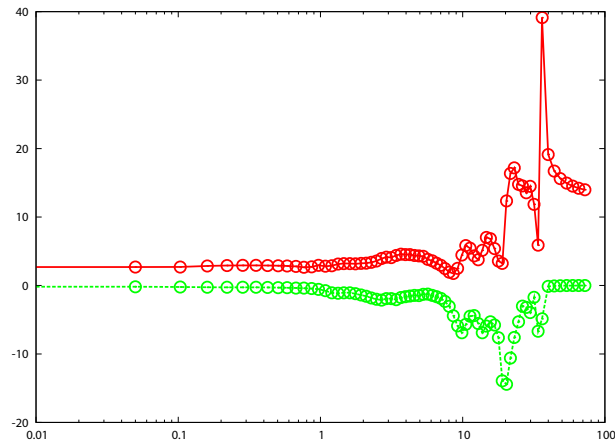


図 18: 通常 RPA (制限なし RPA) 計算による オンサイト遮蔽直接積分の周波数依存性. 赤が実部, 緑が虚部である. 横軸は周波数 ω (eV) の対数である. 縦軸の単位は eV.

8 TiO₂ (局所座標を用いるワニエ関数計算)

局所座標に沿って初期ガウシアンを設置してワニエ関数を構築する計算を, TiO₂ (ルチル構造) の t_{2g} ワニエ関数構築を例にとって説明する. 以下がこの計算のための input.in である. バンド計算 (xTAPP を使用) の条件であるが, k 点サンプリング $6 \times 6 \times 6$ (既約 k 点数 64), 波動関数カットオフ 64 Ry である.

```
&param_wannier
N_wannier=6,                !求めたいワニエ軌道の数
LOWER_energy_window= 7.0d0, !エネルギーウィンドウ下限 (eV)
UPPER_energy_window=11.0d0, !エネルギーウィンドウ上限 (eV)
flg_initial_guess_direc=1,  !初期ガウシアンを表示座標
N_initial_guess=6/         !初期ガウシアンの個数
dxy 0.20 0.00 0.00 0.00  0.46 -0.46 0.76 0.46 -0.46 -0.76  0.71 0.71 0.00 !dxy@Ti1
dyz 0.20 0.00 0.00 0.00  0.46 -0.46 0.76 0.46 -0.46 -0.76  0.71 0.71 0.00 !dyz@Ti1
dzx 0.20 0.00 0.00 0.00  0.46 -0.46 0.76 0.46 -0.46 -0.76  0.71 0.71 0.00 !dzx@Ti1
dxy 0.20 0.50 0.50 0.50 -0.46 -0.46 0.76 0.46  0.46  0.76 -0.71 0.71 0.00 !dxy@Ti2
dyz 0.20 0.50 0.50 0.50 -0.46 -0.46 0.76 0.46  0.46  0.76 -0.71 0.71 0.00 !dyz@Ti2
dzx 0.20 0.50 0.50 0.50 -0.46 -0.46 0.76 0.46  0.46  0.76 -0.71 0.71 0.00 !dzx@Ti2

&param_interpolation
N_sym_points=5/ !計算ラインを構成する対称 k 点数
0.50 0.00 0.50  !対称 k 点 M
0.00 0.00 0.50  !対称 k 点 Z
0.00 0.00 0.00  !対称 k 点 G
0.50 0.00 0.00  !対称 k 点 X
0.00 0.50 0.00  !対称 k 点 Y

&param_visualization
FLG_VIS_WANNIER = 1/
```

この wannier 計算では t_{2g} バンドを含むようにエネルギーウィンドウを Lower_energy_window=7.0 eV から Upper_energy_window=11.0 eV に設定した. またこの計算ではネームリスト変数 flg_initial_guess_direc が 1 にセットされた. これにより初期ガウシアンの表示座標は局所座標系に変更される. このフラッグを立てるもと, 初期ガウシアンの基本情報に加えて表示座標を規定する局所座標を追加入力する. 具体的には以下のように入力する.

```
dxy 0.20 0.50 0.50 0.50 -0.46 -0.46 0.76  0.46 0.46 0.76 -0.71 0.71 0.00
```

上は dxy ガウシアン [軌道指数 0.2 (a.u.)] を格子座標として (0.50, 0.50, 0.50) の位置に置き, さらにこのガウシアンを局所座標 (L_x, L_y, L_z) に沿って設定する場合の入力例である. ここで各局所座標は以下のとおりである.

$$\begin{aligned} L_x &= -0.46\mathbf{e}_x - 0.46\mathbf{e}_y + 0.76\mathbf{e}_z, \\ L_y &= +0.46\mathbf{e}_x + 0.46\mathbf{e}_y + 0.76\mathbf{e}_z, \\ L_z &= -0.71\mathbf{e}_x + 0.71\mathbf{e}_y + 0.00\mathbf{e}_z, \end{aligned}$$

ここで e_x, e_y, e_z はバンド計算実施時に選んだ直交座標系を表す. (L_x, L_y, L_z) は TiO_6 正八面体ユニットが規定する局所座標として定義されている. この場合の (L_x, L_y, L_z) と (e_x, e_y, e_z) は図 19 のようになる. 局所座標の成分についてはユーザーは自分自身で計算系の構造情報から値を算出しなければならない. 今回の場合は TiO_6 正八面体ユニットの Ti と隣接 O の座標から算出するプログラムを自作して求めた.

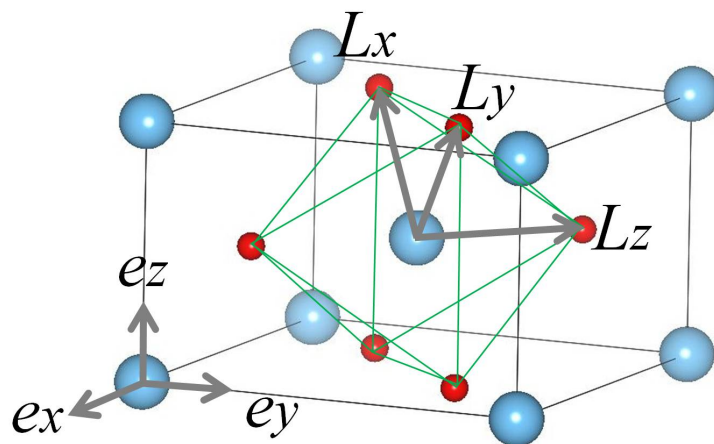


図 19: ルチル型 TiO_2 ワニエ計算における局所座標および直交座標の定義. 緑枠が TiO_6 正八面体ユニットである.

図 20 が得られた d_{xy} 型実空間ワニエ関数であり, 図 19 で定義された局所座標系に基づいて d_{xy} 軌道が得られていることが見て取れる.

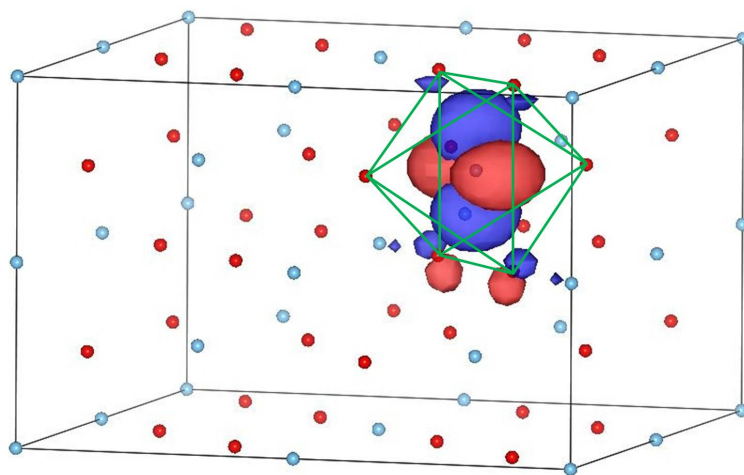


図 20: ルチル型 TiO_2 ワニエ計算によって得られた局所座標表示での d_{xy} 実空間ワニエ関数.

9 Si (B 行列を用いるワニエ関数計算)

初期ガウシアンを一次結合させてワニエ関数を構築する場合のインプットを Si を例にとって説明する. 以下がこの計算のための `input.in` である. バンド計算 (xTAPP) の条件であるが, k 点サンプリング $6 \times 6 \times 6$ (既約 k 点数 16), 波動関数カットオフ 36 Ry である.

```
&param_wannier
N_wannier=8,                !求めたいワニエ軌道の数
Lower_energy_window=-7.0d0, !エネルギーウィンドウ下限 (eV)
Upper_energy_window=17.0d0, !エネルギーウィンドウ上限 (eV)
N_initial_guess=8,          !初期ガウシアン個数
flg_BMAT=1/                 !0:BMAT=unit matrix, 1:reading BMAT
s  0.5d0 0.00d0 0.00d0 0.00d0 !s@Si1
px 0.5d0 0.00d0 0.00d0 0.00d0 !px@Si1
py 0.5d0 0.00d0 0.00d0 0.00d0 !py@Si1
pz 0.5d0 0.00d0 0.00d0 0.00d0 !pz@Si1
s  0.5d0 0.25d0 0.25d0 0.25d0 !s@Si2
px 0.5d0 0.25d0 0.25d0 0.25d0 !px@Si2
py 0.5d0 0.25d0 0.25d0 0.25d0 !py@Si2
pz 0.5d0 0.25d0 0.25d0 0.25d0 !pz@Si2
0.50 -0.50  0.50 -0.50 0.00  0.00  0.00  0.00 !B_MAT(ig=1,iw=1:8)
0.50  0.50 -0.50 -0.50 0.00  0.00  0.00  0.00 !B_MAT(ig=2,iw=1:8)
0.50 -0.50 -0.50  0.50 0.00  0.00  0.00  0.00 !B_MAT(ig=3,iw=1:8)
0.50  0.50  0.50  0.50 0.00  0.00  0.00  0.00 !B_MAT(ig=4,iw=1:8)
0.00  0.00  0.00  0.00 0.50  0.50 -0.50  0.50 !B_MAT(ig=5,iw=1:8)
0.00  0.00  0.00  0.00 0.50 -0.50  0.50  0.50 !B_MAT(ig=6,iw=1:8)
0.00  0.00  0.00  0.00 0.50  0.50  0.50 -0.50 !B_MAT(ig=7,iw=1:8)
0.00  0.00  0.00  0.00 0.50 -0.50 -0.50 -0.50 !B_MAT(ig=8,iw=1:8)

&param_interpolation
N_sym_points=5/ !計算ラインを構成する対称 k 点数
0.500 0.500 0.500 !対称 k 点 L
0.000 0.000 0.000 !対称 k 点 G
0.500 0.000 0.500 !対称 k 点 X
0.500 0.250 0.750 !対称 k 点 W
0.500 0.500 0.500 !対称 k 点 L

&param_visualization
flg_vis_wannier=1/
```

この計算では Si の 8 つの sp^3 型ワニエ関数を 8 つの s 型および p 型ガウス関数の一次結合より構築する. 一次結合の係数である B 行列情報は初期ガウシアン情報に続いて入力される. B 行列成分についてはユーザーは自分自身で計算系の構造情報から値を算出しなければならない. 今回の場合は Si の共有結合に沿う方向から成分を算出した. B 行列のサイズは 8×8 であり, 行が

ガウス関数の並び, 列がワニエ軌道の並びを表す. B 行列の成分 B_{ij} は以下のように定義される.

$$w_j^{in}(\mathbf{r}) = \sum_i B_{ij} g_i(\mathbf{r})$$

ここで g_i が i 番目のガウス関数, w_j^{in} が j 番目のワニエ関数のイニシャルゲスである. B 行列の入力フォーマットは以下のとおりである.

```
do ig=1,N_initial_guess
  read(5,*) (B_MAT(ig,iw),iw=1,N_wannier) !real(8)
enddo
```

なお `reading_bmat_format=1` をネームリストに追加すると, B 行列の行と列を入替えて入力できる. この場合, 行がワニエ関数, 列がガウス関数である.

```
do iw=1,N_wannier
  read(5,*) (B_MAT(ig,iw),ig=1,N_initial_guess)
enddo
```

図 21 が得られた sp3 型実空間ワニエ関数である.

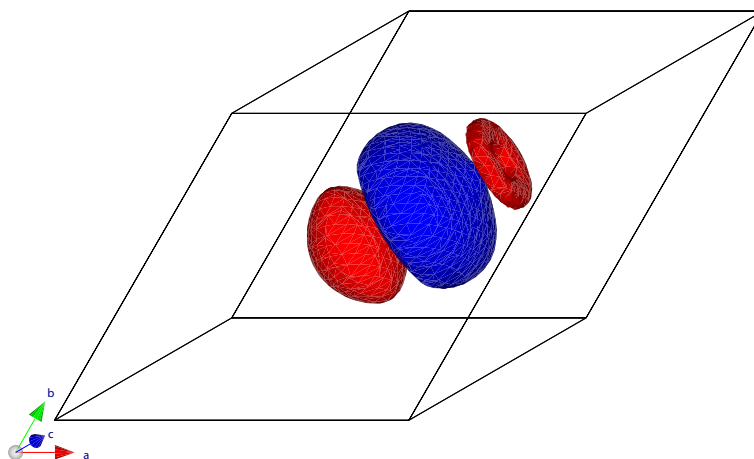


図 21: Si の sp3 型実空間ワニエ関数.

10 SrVO₃ (制限 RPA を用いた相互作用評価)

下は制限 RPA を用いた SrVO₃ の t_{2g} ワニエ関数による相互作用行列要素計算のためのインプットである。

```
&param_chiqw
Ecut_for_eps=5.0d0,!分極関数のカットオフ (Ry)
flg_cRPA=1/          !制限 RPA オプション (0:通常 RPA, 1:制限 RPA)

&param_wannier
N_wannier=3,          !求めたいワニエ軌道の数
Lower_energy_window=6.5d0, !エネルギーウィンドウ下限 (eV)
Upper_energy_window=9.7d0, !エネルギーウィンドウ上限 (eV)
N_initial_guess=3/    !初期ガウシアン の 個数
dxy 0.5d0 0.5d0 0.5d0 0.5d0 !dxy@V
dyz 0.5d0 0.5d0 0.5d0 0.5d0 !dyz@V
dzx 0.5d0 0.5d0 0.5d0 0.5d0 !dzx@V

&param_interpolation
N_sym_points=5/    !計算ラインを構成する対称 k 点数
0.500 0.500 0.500 !対称 k 点 R
0.000 0.000 0.000 !対称 k 点 G
0.500 0.000 0.000 !対称 k 点 X
0.500 0.500 0.000 !対称 k 点 M
0.000 0.000 0.000 !対称 k 点 G

&param_visualization
/

&param_calc_int
/
```

このインプットは xTAPP を念頭に変数設定されている。エネルギーウィンドウの設定値はバンド計算に依存するので注意されたい。上記のインプットは sample に納められた簡易版であるが、収束を得たいなら k 点数 $10 \times 10 \times 10$, 波動関数カットオフ 100 Ry, 誘電関数カットオフ 15 Ry, 総バンド数 100 程度にとるのが良いだろう。SrVO₃ ではプラズモン励起が複数観測されるので、周波数グリッドもできるだけ密に取った方がよい。具体的には Num_freq_grid=150 程度を設定するとよい。遷移金属は局在度が強く、相互作用パラメータの定量性は擬ポテンシャルに影響されることがあるので注意されたい (できるだけ硬めの擬ポテンシャルを使う方がよい)。

11 (TMTSF)₂PF₆ (制限 RPA を用いた相互作用評価)

下は制限 RPA を用いた 有機化合物 (TMTSF)₂PF₆ の HOMO ワニエ関数による相互作用行列要素計算のためのインプットである。計算は物性研究所システム B で行った。コミュニティを 6 MPI で構築すると、12 コミュニティの並列ジョブである (計 72 MPI)。1 MPI は 24 スレッドで構成される。計算時間は約 100 分であった。バンド計算には xTAPP を用いた。

```
&param_chiqw
flg_cRPA=1,          !制限 RPA オプション (0:通常 RPA, 1:制限 RPA)
MPI_num_qcomm=12/    !MPI コミュニティの総数

&param_wannier
N_wannier=2,          !求めたいワニエ軌道の数
Lower_energy_window=2.7d0,    !エネルギーウィンドウ下限 (eV)
Upper_energy_window=4.5d0,    !エネルギーウィンドウ上限 (eV)
FLG_BMAT=1,           !0:BMAT=unit matrix, 1:reading BMAT
N_initial_guess=6/     !初期ガウシアン の 個数
px 0.20 0.26635 0.45950 0.44800 !px@TMTSF 分子 1 の中心サイト
py 0.20 0.26635 0.45950 0.44800 !py@TMTSF 分子 1 の中心サイト
pz 0.20 0.26635 0.45950 0.44800 !pz@TMTSF 分子 1 の中心サイト
px 0.20 0.73365 0.54050 0.55200 !px@TMTSF 分子 2 の中心サイト
py 0.20 0.73365 0.54050 0.55200 !py@TMTSF 分子 2 の中心サイト
pz 0.20 0.73365 0.54050 0.55200 !pz@TMTSF 分子 2 の中心サイト
0.9995851478 0.0000000000 !B_MAT(ig=1,iw=1:2)
0.0012540916 0.0000000000 !B_MAT(ig=2,iw=1:2)
-0.0287742877 0.0000000000 !B_MAT(ig=3,iw=1:2)
0.0000000000 0.9996715835 !B_MAT(ig=4,iw=1:2)
0.0000000000 -0.0140153561 !B_MAT(ig=5,iw=1:2)
0.0000000000 -0.0214544857 !B_MAT(ig=6,iw=1:2)

&param_interpolation
N_sym_points=5/      !計算ラインを構成する対称 k 点数
0.000 0.500 0.000 !対称 k 点 Y
0.000 0.000 0.000 !対称 k 点 G
0.500 0.000 0.000 !対称 k 点 X
0.500 0.500 0.000 !対称 k 点 M
0.000 0.000 0.000 !対称 k 点 G

&param_visualization
/

&param_calc_int
/
```

12 計算 q 点を指定して計算したい場合 (chiqw)

下は計算 q 点を指定して chiqw 計算を実行する際のインプットである。この場合はネームリスト ¶m_chiqw の変数 flg_calc_type を 2 にセットする。この指定のもと、計算したい q 点の総数を 5 とし (n_calc_q=5), ネームリストの下に計算したい q 点の番号を列挙する。この番号はサンプル既約 k 点ベクトルを納めたファイル dat.sample-k の k 点の並びに順じていることに注意されたい。この例では 4, 8, 14, 22, 27 番目の 5 個の q 点が計算対象となる。

```
&param_chiqw
flg_calc_type=2, !0:all-q-calc, 1:gamma-only, 2:bulkjob mode
n_calc_q=5/      !計算したい  $q$  点の総数
4 8 14 22 27    !計算したい  $q$  点の番号
```

13 実空間ワニエ関数を指定して計算したい場合 (wannier)

下は プログラム wannier において、実空間ワニエ関数を指定して計算する場合のインプットである。この場合はネームリスト ¶m_visualization の変数 N_write_wannier を求めたい個数にセットする。この指定のもと、ネームリストの下に求めたいワニエ関数の番号を列挙する。この例では 1, 5, 8 番目の 3 つのワニエ関数が実空間ワニエ計算の対象となる。

```
&param_visualization
N_write_wannier=3/ !求めたい実空間ワニエ関数の数
1 5 8             !求めたい実空間ワニエ関数の番号
```

14 ファイルフォーマット (読込形式)

1. dat.bandcalc (バンド計算情報)

```
OPEN(117,FILE='./dir-wfn/dat.bandcalc')
read(117,*) Ecut_for_psi !波動関数カットオフエネルギー (Ry) real(8)
read(117,*) FermiEnergy !フェルミエネルギー (au) real(8)
read(117,*) Etot !全エネルギー (au) real(8)
```

2. dat.sample-k [サンプル k 点ベクトル (既約 k 点)]

```
OPEN(101,FILE='./dir-wfn/dat.sample-k')
read(101,*) Nk_irr !既約 k 点数 integer
do ik=1,Nk_irr
  read(101,*) (SKI(i,ik),i=1,3) !既約サンプル k 点 real(8)
enddo
```

3. dat.eigenvalue [バンドエネルギー (既約 k 点)]

```
OPEN(111,FILE='./dir-wfn/dat.eigenvalue')
rewind(111)
read(111,*) NBAND !バンド数
do ik=1,Nk_irr
  do ib=1,NBAND
    read(111,*) E_EIGI(ib,ik) !バンドエネルギー (au) real(8)
  enddo
enddo
```

4. dat.nkm [展開 G 数 (既約 k 点)]

```
OPEN(132,FILE='./dir-wfn/dat.nkm-nw')
rewind(132)
do ik=1,Nk_irr
  read(132,*) NGI(ik) !展開 G 数 integer
enddo
```

5. dat.kg [展開 G ベクトル (既約 k 点)]

```
OPEN(104,FILE='./dir-wfn/dat.kg')
do ik=1,Nk_irr
  read(104,*) NG_for_psi !既約 ik 点での展開 G 数 integer
  do ig=1,NG_for_psi
    read(104,*) (KGI(i,ig,ik),i=1,3) !既約 ik 点での展開 G ベクトル integer
  enddo
enddo
```

6. dat.wfn [波動関数 (既約 k 点)]

```
OPEN(102,FILE='./dir-wfn/dat.wfn',FORM='unformatted')
read(102) ncomp !成分数
do ik=1,Nk_irr
  do ib=1,NBAND
    read(102)((C0(ic,i,ib,ik),ic=1,ncomp),i=1,NGI(ik))!波動関数 complex(8)
  enddo
enddo
!現バージョンでは, ncomp=1 にしか対応していない
```

7. dat.lattice (基本並進ベクトル)

```
OPEN(105,FILE='./dir-wfn/dat.lattice')
read(105,*) a1(1), a1(2), a1(3) !a1 vector (bohr 単位) real(8)
read(105,*) a2(1), a2(2), a2(3) !a2 vector (bohr 単位) real(8)
read(105,*) a3(1), a3(2), a3(3) !a3 vector (bohr 単位) real(8)
```

8. dat.symmetry (対称性, ++逆空間で作用する表現であること++)

```
OPEN(100,FILE='./dir-wfn/dat.symmetry')
read(100,*) nsymq !対称性数 integer
read(100,*) nnp !部分並進 integer
do iop=1,nsymq
  read(100,*) ((rg(i,j,iop),i=1,3),j=1,3) !回転対称性 integer
  read(100,*) (pg(i,iop),i=1,3) !部分並進対称性 integer
enddo
```

15 ライセンス

本ソフトウェアのプログラムパッケージおよびソースコード一式は GNU General Public License version 3 (GPL v3) に準じて配布されています.

16 引用法

RESPACK を引用する際は, 以下の論文を引用して頂ければと思います.

1. K. Nakamura, Y. Nohara, Y. Yoshimoto, Y. Nomura, Phys. Rev. B **93**, 085124 (2016).
2. K. Nakamura, Y. Yoshimoto, T. Kosugi, R. Arita, M. Imada, J. Phys. Soc. Jpn **78**, 083710 (2009).
3. K. Nakamura, R. Arita, M. Imada, J. Phys. Soc. Jpn **77**, 093711 (2008).
4. Y. Nohara, S. Yamamoto, T. Fujiwara, Phys. Rev. B **79**, 195110 (2009).
5. T. Fujiwara, S. Yamamoto, Y. Ishii, J. Phys. Soc. Jpn. **72**, 777 (2003).