

gwsc: a script to run QSGW calculation

gwscがQSGW計算実行スクリプトである。

QSGW計算は、複数のfortran実行ファイルを呼び出して実行される。

ファイルGWinputが読み込まれる。

Usage

Usage: gwsc -np NP [-np2 NP2] [--phispinsym] [--gpu] [--mp] nloop extension
[Options]

-np NP

MPI並列数を指定する。

-np2 NP2

GPU版を使用する場合のみ指定する。GPUで実行される計算のMPI並列数を指定する。

通常は使用できるGPU数を指定する。

--gpu

GPU版を使用する場合のみ指定する。

--mp

GPU-MP版(混合精度)を使用する場合のみ指定する。計算精度に注意すること。

nloop

QSGWのイテレーション数を指定する。

extension

ctrl ファイルの拡張子を指定する。

Options

追加のオプションを指定する。

追加オプションは、全ての実行ファイルの実行時引数となる。以下追加のオプションのリスト。

またlmfへのオプション-vsoc=1などもここに書く。

--keepwv

--gpu を指定した場合に自動で付け加わる。

自己エネルギー(関連部分)を計算する際に、遮蔽クーロン相互作用の行列要素をメモリ上に保持する。

GPU計算ではファイルIO, データ転送が特に律速になるが, それを回避するため, ただし十分なGPUおよびCPUメモリが必要となる。

--nb=X

- Xは整数 --nb=4のように指定する。
遮蔽クーロン相互作用(W)計算hrcxq or hrcxq_gpu で使用される。分極関数のMPB基底並列数を指定する。
GPU計算においてhrcxq(_gpu)計算でメモリ不足になる場合に使用する。--np2 で指定した並列数を割り切れる値を入れる必要がある。

--nwpara=X

- Xは整数 --nwpara=2のように指定する。
相関部分の自己エネルギー計算hsfp0_sc --job=2 or hsfp0_sc_gpu --job=2 で使用される。 ω 積分の並列数を指定する。
--keepwv 使用時(GPU版ではデフォルトで使用される) __WVR.X (X=1,...)ファイルがメモリに乗りきらずメモリ不足になる場合に使用する。
--np2 で指定した並列数を割り切れる値を入れる必要がある。

--tetwtk

指定すると, 分極関数を計算する際に, 結合状態間の四面体重みをメモリ上に保持しない。 k 点が多い計算でメモリ不足になる場合に使用する。

--skipGS

lmf --jobgw=1 で使用される。

GW計算ではDFT計算(lmf)で得られた波動関数をlmfとは異なる基底関数で展開しなおす。再展開後の波動関数についての規格直交化をスキップする場合に指定する。

通常は指定する必要はないが、lmf --jobgw=1計算が遅い場合には指定することによって計算の高速化が期待される。

- ecaljでは有限のqで誘電関数が計算できる—このとき分母分子のキャンセレーションが起こるため、波動関数の直行性が正確である必要があり、そのときには--skipGSを使うべきでない。

--normcheck

lmf --jobgw=1 で使用される。GW計算で使用する波動関数の規格直交性を確認したいときに使用する。

--ntqxx

This fix the number of bands to calculate self energy at the first iteration for each q point in the IBZ.

In principle, the number is determined by

Cautions

- QPU.[number]runをチェックして、number回のQSGWイテレーションが終了している、と認識する。
(初期状態から実行したいときはすべての*run*ディレクトリ、ファイルを消すこと)。
- QSGW.[number]runディレクトリには、QSGWのnumber回目の結果rst,sigm(加えてatmpnu,ctrl,GWinput)が格納されており、これを用いてバンドプロットなどができる。

GWinput

GWinput sets the computation conditions for GW/QSGW calculations.

mkGWinputによる自動生成

- GWinput は 自動生成することができ、殆どのパラメータはデフォルト設定で利用できる。

自動生成コマンド

```
mkGWinput $target
```

[!INFO]

mkGWinput はpython スクリプトだが内部でecalj packageに含まれている実行ファイルを実行する。
そのため ~/bin ディレクトリ(ecalj package のバイナリinstall ディレクトリ)へのPATHが通っている必要がある。

これにより GWinput.tmp が生成される。

```
cp GWinput.tmp GWinput
```

PROF

とし, GWinput を用意する。

Format

- !で開始する行はコメント
- keyword value(s) の形式であり記載順序は問わないが, 同じkeywordを複数記述しないようにする

Parameters in GWinput

生成されたGWinputに, パラメータの意味は記載されていますが, 以下に重要な変数と記載がない変数について記述します。

n1n2n3

\$GW\$自己エネルギー計算におけるk点の数。ctrl file に記載するk点とは異なる点を指定できる。計算コストは, このk点数の2乗に比例することに注意。

- **type** : integer list

We set number of k points for self-energy in GWinput. These can be 1/2 or 2/3 of k points specified in ctrl, in order to reduce computational time.

The 6x6x6 k points is feasible setting for ZB structure (2 atoms per cell). That is, $6 \times 6 \times 6 \times 2 = 432 \sim k \text{ points} \times \text{atom number}$.

For example, when we try 8 atoms per cell, 4x4x4 or 3x3x3 is fine because $4 \times 4 \times 4 \times 8 \sim 400$.

For metallic systems, larger is fine, but limited by computational time. See takao kotani's papers, for example, <https://doi.org/10.1103/PhysRevB.93.075125>

Good news is that we don't need to use so many k points as in ctrl.

In cases, 4x4x4 for ZB is not so bad --- roughly speaking, this is a low limit for publication. This means 3x3x3x8 (for 8 atom case) is not so bad. ($3 \times 3 \times 3 \times 8 > 4 \times 4 \times 4 \times 2$).

See our examinations shown in <https://doi.org/10.7566/JPSJ.83.094711> and <http://doi.org/10.7567/JJAP.55.051201>

KeepEigen

波動関数をメモリに保持するかどうか。波動関数はCPHI およびGEIGというファイルで出力される。それらが1MPIが使用できるメモリに対して, 同等もしくはそれ以上である場合は必ず`.false.`にする。 $\$k\$$ 点や軌道数が多い場合は`.false.`にする。

- **type**: boolean
- **default**: `.true.`

MEMnmbatch

バッチサイズ:単位GB。1MPIが使用できるメモリサイズの3分の1から4分の1程度を記載すると良い。`zmel_max_size` とほぼ同様な意味をもつ。自己エネルギーの計算で使用される。大規模系では大きい方が計算は早くなるが、メモリ不足で計算が落ちる場合は小さくする。

- **type**: float
 - **default**: 2.0
- 系に対して値が小さいと"sxcf_fal2_count.sc. Too small memory for nmbatch mechanism. Enlarge GWinput MEMnmbatch" との出力が`stdout.xxxx.hsfp0_sc.mode.yyy`に出力され計算が終了する場合がある。その場合は大きくする。

zmel_max_size

バッチサイズ:単位GB。1MPIが使用できるメモリサイズの3分の1から4分の1程度を記載すると良い。`MEMnmbatch` とほぼ同様な意味をもつが, $\$W\$$ の計算で使用される。大規模系では大きい方が計算は早くなるが、メモリ不足で計算が落ちる場合は小さくする。

- **type**: float
- **default**: 1.0

KeepPpb

MT内の積基底とMT内波動関数基底の行列要素(ppb変数)をメモリに保持するかどうか。通常はデフォルト値 `.true.` で良いが, 使用メモリ量を軽減させたいときに `.false.` とする。

- **type:** boolean
- **default:** `.true.`

GaussianFilterX0

密度応答関数の計算に用いられる振動数におけるスミアリングパラメータ: 単位 a.u.^2 (Hartree 2)。QSGW計算が不安定な場合(固有値が振動するなど)の際に用いる。

- **type:** float
- **default:** 0
- **examples:** GaussianFilterX0 0.0001

QpGcut_psi

波動関数の平面波基底のカットオフエネルギー: 単位 Ryd。 `lmf` 計算で用いるPMT手法とは波動関数の基底関数が異なることに注意。原子間領域の波動関数は全て平面波で記述される。

- **type:** float
- **default:** 4.0
波動関数のIPWのカットオフctrl内のpwemaxとも関係する。pwemaxとしては2,3を試すことが多い(単位Ryd)。

QpGcut_cou

積基底の平面波部分のカットオフエネルギー: 単位 Ryd

- **type:** float
- **default:** 3.0

emax_sigm

フェルミ準位から測った, 計算する準粒子状態の自己エネルギーの最大値: 単位 Ryd。QSGW計算では自己エネルギーはポテンシャル構築に使用されるため小さくするとポテンシャルの精度も下がる。

- **type:** float
- **default:** 3.0

High resolution energy mesh near Ef for metal

For metal, it may be better to set

```
HistBin_dw      1d-5 ! 1d-5 is fine mesh (good for metal?) !(a.u.)
BinWidth along real axis at omega=0.
HistBin_ratio 1.03 ! 1.03 maybe safer. frhis(iw)= b*(exp(a*(iw-1))-1),
where a=ratio-1.0 and dw=b*a
```

Smaller lcutmx to reduce computational time

To reduce the computational time, we reduce number of MPB (mixed product basis).

One is lcutoff of Product Basis (PB) within MT. Use lcutmx=2 for oxygen or something (s,p block atoms). Thus it is like

```
lcutmx(atom) = maximum l-cutoff for the product basis.  =4 is required
for atoms with valence d, like N
4 4 4 2 2 2
```

in the section of GWinput.

- NOTE: we know that lcutmx =6 is required for 4f systems.

Smaller IPW related part to reduce computational time

To reduce computational time, we may reduce MPB coming from IPS, reduce the size of IPW for psi, reduce emax_sigm and pwemax as follows.

- QpGcut_cou is for the Interstitial plane wave (IPW) for MPB.
- QpGcut_psi is for expansion of eigenfunctions.
- emax_sigm is the upper cutoff (relative to the Fermi energy) to calculate self energy.
- pwemax (in ctrl) is the APW basis cutoff for the eigenfunction.

To reduce computational time, we may use

```
QpGcut_psi 3.0
QpGcut_cou 2.5
emax_sigm 2.0
```

PROF

In addition, we may use

```
pwemax=2 (in ctrl file).
```

This is for the APWs in the band calculation.

We sometimes use this setting. It is better to check how the numerical results are affected.