# 龙讯教程

## 电声耦合矩阵计算

Contributed by 叶耀坤

modified at 2021.11.5

support@pwmat.com

本module功能：使用wannier函数（WFs）计算电声耦合矩阵

核心计算目标：电声耦合矩阵，即 $\left\langle \psi_{mk_1} \left| \frac{\partial H}{\partial v_{bq}} \right| \psi_{nk_2} \right\rangle$ 其中，k1+q=k2

m和n为电子能带指标，k1和k2为电子k点指标，b为声子模指标，q为声子q点指标

计算步骤（大纲）：

1. 做JOB=SCF，构造WFs
2. 利用PWmat的声子模块，计算声子相关信息
3. 做displacement，将WFs写入G空间，做JOB=HPSI，用于后续计算电声耦合矩阵
4. 计算电声耦合矩阵

```
calc_EPC     calc_EPW.x    displacement     G-wann.x    make_wann
calc_EPC.x   degenerate    DISPLACEMENT.x   in_wann.x   ref
calc_EPW     degenerate.x  G-wann           Makefile
```

**wannier functions (WFs) 理论简介**

紧束缚近似:

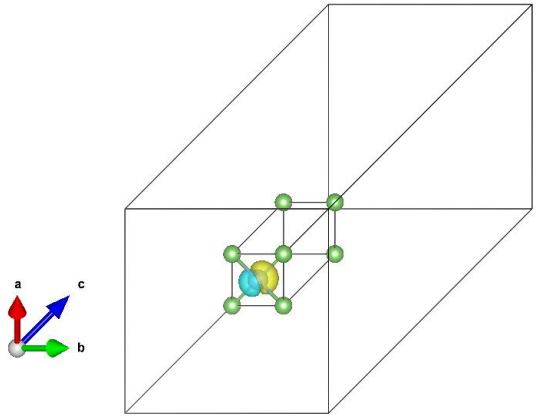
$$\psi(r) = \frac{1}{\sqrt{N}} \sum_{R_l} e^{ik \cdot R_l} \phi_i(r - R_l)$$

“Localize”

$$\int \phi_i^*(r - R_p)\phi_i(r - R_l)\, dr = \delta_{pl}$$

**将WFs选取为展开基组**

$$\psi_{nk}(r) = \sum_{mR} e^{ikR} U_{mnk}^{+} W_m(r - R)$$




U矩阵的作用是使其满足规范性

WFs “局域”示意图

$$\psi_{nk}(r) = \sum_{mR} e^{-ikR} U^{+}_{mnk} W_m(r+R)$$

```
eikr
W90: -4.637370263141959E-005  1.189060446887973E-004
TRY: -4.637654542870407E-005  1.189041929697664E-004
W90:  1.901298509084546E-005 -9.379986966689191E-005
TRY:  1.900823337425177E-005 -9.380265281941672E-005
W90:  5.497568708536781E-005 -3.819735701769107E-005
TRY:  5.496871701513405E-005 -3.819429447103448E-005
```

将WFs加回bloch state的对比

$$\psi_{nk}(r) = \sum_{mR} e^{-ikR} U^{+}_{mnk}\, W_m(r+R)$$

$$\left\langle \psi_{mk_1} \left| \frac{\partial H}{\partial \nu_{bq}} \right| \psi_{nk_2} \right\rangle = \left\langle \sum_{m' R'_e} e^{-ik_1 \cdot R'_e}\, U^{+}_{mm'k_1} W_{m'}(r+R'_e) \left| \sum_{\kappa R_p} \frac{\partial H}{\partial \tau^{bq}_{\kappa R_p}} \frac{\partial \tau^{bq}_{\kappa R_p}}{\partial \nu_{bq}} \right| \sum_{n R_e} e^{-ik_2 \cdot R_e} U^{+}_{nn'k_2} W_{n'}(r+R_e) \right\rangle$$

$$\frac{\partial \tau^{bq}_{\kappa R_p}}{\partial \nu_{bq}} \rightarrow \frac{\mu'_{bq\kappa}(r)}{\sqrt{M(\kappa)}}\, e^{iqR_p}$$

$$= \sum_{R'_e, R_e, R_p} \sum_{m'n'\kappa} \frac{\mu'_{bq\kappa}(r)}{\sqrt{M(\kappa)}} e^{ik_1 R'_e - ik_2 R_e + iqR_p}\, U_{mm'k_1} U^{+}_{nn'k_2} \left\langle \boldsymbol{W_{m'}(r+R'_e)} \left| \frac{\boldsymbol{\partial H}}{\boldsymbol{\partial \tau^{bq}_{\kappa R_p}}} \right| \boldsymbol{W_{n'}(r+R_e)} \right\rangle$$

$$= \sum_{R'_e - R_p, R_e - R_p} \sum_{m'n'\kappa} \frac{\mu'_{bq\kappa}(r)}{\sqrt{M(\kappa)}} e^{ik_1 (R'_e - R_p)}\, e^{-ik_2 (R_e - R_p)} \sum_{R_p} e^{i(k_1 - k_2 + q) R_p}\, U_{mm'k_1} U^{+}_{nn'k_2} \left\langle \boldsymbol{W_{m'}(r+R'_e)} \left| \frac{\boldsymbol{\partial H}}{\boldsymbol{\partial \tau^{bq}_{\kappa R_p}}} \right| \boldsymbol{W_{n'}(r+R_e)} \right\rangle$$

$$\sum_{R_p} e^{i(k_1 - k_2 + q) R_p} \rightarrow \delta_{k_1 - k_2 + q}$$

$$= \sum_{R'_e, R_e} \sum_{m'n'\kappa} \frac{\mu'_{bq\kappa}(r)}{\sqrt{M(\kappa)}} e^{ik_1 R'_e} e^{-ik_2 R_e}\, U_{mm'k_1} U^{+}_{nn'k_2} \left\langle \boldsymbol{W_{m'}(r+R'_e)} \left| \frac{\boldsymbol{\partial H}}{\boldsymbol{\partial \tau^{bq}_{\kappa R_p}}} \right| \boldsymbol{W_{n'}(r+R_e)} \right\rangle \qquad (\boldsymbol{R_p = 0})$$

$$= sum_i sum_{xyz} \sum_{R_p} \sum_{R_l} f_{bq}(i, xyz) e^{-ik_1 R_p} e^{ik_2 R_l} \delta_{-k_1 + k_2 + q} U^{*}_{mm'} U_{nn'} \qquad (R = 0)$$

$$\left\langle \boldsymbol{W_{m'}(r - R_p)} \left| \frac{\boldsymbol{\partial H}}{\boldsymbol{\partial R}} \right| \boldsymbol{W_{n'}(r - R_l)} \right\rangle$$

# 理解phonon mode

Viewpoint Paper

# First principles phonon calculations in materials science

Atsushi Togo [a,b], Isao Tanaka [a,b,c,*]

[a] Center for Elements Strategy Initiative for Structure Materials (ESISM), Kyoto University, Sakyo, Kyoto 606-8501, Japan
[b] Department of Materials Science and Engineering, Kyoto University, Sakyo, Kyoto 606-8501, Japan
[c] Nanostructures Research Laboratory, Japan Fine Ceramics Center, Atsuta, Nagoya 456-8587, Japan

As found in text books [27–30], dynamical property of atoms in the harmonic approximation is obtained by solving eigenvalue problem of dynamical matrix $D(\mathbf{q})$,

$$D(\mathbf{q})\mathbf{e}_{\mathbf{q}j} = \omega^2_{\mathbf{q}j}\mathbf{e}_{\mathbf{q}j}, \quad \text{or} \quad \sum_{\beta\kappa'} D^{\alpha\beta}_{\kappa\kappa'}(\mathbf{q}) e^{\beta\kappa'}_{\mathbf{q}j} = \omega^2_{\mathbf{q}j} e^{\alpha\kappa}_{\mathbf{q}j}, \tag{3}$$

with

$$D^{\alpha\beta}_{\kappa\kappa'}(\mathbf{q}) = \sum_{l'} \frac{\Phi_{\alpha\beta}(0\kappa, l'\kappa')}{\sqrt{m_\kappa m_{\kappa'}}} e^{i\mathbf{q}\cdot[\mathbf{r}(l'\kappa') - \mathbf{r}(0\kappa)]}, \tag{4}$$

where $m_\kappa$ is the mass of the atom $\kappa$, $\mathbf{q}$ is the wave vector, and $j$ is the band index. $\omega_{\mathbf{q}j}$ and $\mathbf{e}_{\mathbf{q}j}$ give the phonon frequency and polarization vector of the phonon mode labeled by a set $\{\mathbf{q}, j\}$, respectively.

解dynamical matrix

本征值 $\omega^2_{qj}$对应频率

本征矢 $e^{\alpha\kappa}_{qj}$ 对应极化矢量

两者一起给出一个phonon mode_{q,j}
q是声子q点，j是声子3N个支，N为原子数

1. 计算SCF，生成wannier函数（WFs）

① 准备好输入文件，进行一次scf运算，参考etot.input如下：

```
1   4
IN.ATOM    = atom.config
IN.PSP1    = Al.SG15.PBE.UPF
IN.PSP2    = As.SG15.PBE.UPF
JOB = SCF
MP_N123 = 4 4 4 0 0 0 2
XCFUNCTIONAL = PBE
PWSCF_OUTPUT = T
OUT.FORCE = T
```

注意：这里构造的初始k点网络需要与后面的WFs的supercell指标一致，因此MP_N123这里一定要关闭对称性，以及一定要打开PWSCF_OUTPUT = T 以及根据计算资源与计算结果准确度平衡考虑，推荐使用4*4*4

② 在①任务结束的目录下，运行 in_wann.x 程序，辅助生成WFs的输入文件（in_wann.x由本module提供）

```
[YaokunYe@mgr alas]$ in_wann.x
 **------------------------------------------------------**
 ** JOB DONE!                                            **
 ** Please check file wannier90.win and wannier90.pw2wan!**
 **------------------------------------------------------**
```

注意：in_wann.x程序需要本目录下有atom.config文件提供原子位置信息，以及需要有①中MP_N123生成的OUT.KPT文件提供k点信息，如果缺少这两个文件的话，程序会报错，请根据提示修改相关内容

```
[YaokunYe@mgr alas]$ in_wann.x
 File doesn't exist! Make sure you have file 'OUT.KPT'!
 Program goes wrong!
```

## ③ 完善wannier90.win和wannier90.pw2wan输入文件

对于wannier90.win, 需要填写的内容有：

```
num_wann = 8
num_bands = 8
num_iter = 100

exclude_bands = 9-18
```

num_wann是产生WFs的个数，需要与下边projections的个数一致，计算方法为：
num_wann =$\sum_i atom(i)$上的WFs projections

其中，l=0,1,2,3对应于s, p ,d, f, 对应的projections数量是1,3,5,7（杂化情况可参考wannier90 user guide及原子轨道信息分析，此处不做展开）

```
begin projections
!Pleae Choose Tight Binding Basis Carefully!
f=0,0,0:l=0;l=1
f=0.25,0.25,0.25:l=0;l=1
end projections
```

因此在本例中，num_wann = Al原子对应的(1+3)以及As原子对应的(1+3)，即num_wann = 8

本module中，均采用num_bands = num_wann

exclude_bands为选取WFs对应bands后排除掉的bands，建议值为num_wann+1至scf计算时设置的num_band值（可在etot.input中设置或使用默认值，REPORT中可以查到）

③ 完善wannier90.win和wannier90.pw2wan输入文件

如计算体系无需考虑spin，请直接使用in_wann.x协助生成的wannier90.pw2wan

对于wannier90.win, 需要填写的内容有：

```
wannier_plot = T
wannier_plot_supercell = 4
```

```
mp_grid 4 4 4
```

wannier_plot = T 打开这个开关即可
wannier_plot_supercell = 4，构造WFs时的超胞大小
根据Marzari的理论，构造的超胞大小应该与原始的k点网络大小一样，即我们使用的4*4*4的k点网络，就应该构造4*4*4的超胞

至此wannier90.win文件已经填写完成，可以开始运算

④依次输入以下命令（请使用我们重新编译过的wannier90.x，由本module提供）：
wannier90.x -pp wannier90
pw2wannier90.x < wannier90.pw2wan
wannier90.x wannier90

```
[YaokunYe@mgr alas]$ ls wannier90*
wannier90_00001.xsf  wannier90_00005.xsf  wannier90.amn  wannier90.nnkp    wannier90.wout
wannier90_00002.xsf  wannier90_00006.xsf  wannier90.chk  wannier90.pw2wan
wannier90_00003.xsf  wannier90_00007.xsf  wannier90.eig  wannier90_u.mat
wannier90_00004.xsf  wannier90_00008.xsf  wannier90.mmn  wannier90.win
```

上述流程顺利完成后，可以看到目录下已经生成WFs（要注意查看WFs是否局域）

2. 使用pwmat声子模块计算声子，得到mesh.yaml文件

```
DIM = 2 2 2
PRIMITIVE_AXIS = 1.0 0.0 0.0  0.0 1.0 0.0  0.0 0.0 1.0
BAND = 0.0 0.0 0.0  0.5 0.0 0.0
BAND_LABELS = \Gamma  X
BAND_POINTS = 101
FC_SYMMETRY = .TRUE.
FREQUENCY_CONVERSION_FACTOR = 15.633302
MP = 4 4 4
DOS = .TRUE.
FPITCH = 0.1
SIGMA = 0.1
EIGENVECTORS = .TRUE.
GAMMA_CENTER = .TRUE.
MESH_SYMMETRY = .FALSE.
```

k点与之前保持一致

声子网格与k点网格一致

需要在phonon_std的band_dos.conf文件中加上这两行参数

```
[YaokunYe@mgr old]$ ls mesh.yaml*
mesh.yaml  mesh.yaml.dat
```

得到声子信息文件mesh.yaml和mesh.yaml.dat，将其存放在SCF任务完成的目录下

3. 在SCF的目录下，运行DISPLACEMENT.x做displacement,得到disp-xxx文件夹和运行脚本

```
[YaokunYe@mgr module-test]$ ~/DISPLACEMENT.x
There are   2  atoms in the unit cell
Atoms in unit cell are:
 1  33  0.0000000000000000  0.0000000000000000  0.0000000000000000
 2  13  0.2500000000000000  0.2500000000000000  0.2500000000000000
Program will produce    13  sub-directories
Num_wann:   8
Super cell grid is:   4   4   4

 Doing  As   atom!

 Move towards x axis
 Creating Supercell
 Move towards -x axis
 Creating Supercell
 Move towards y axis
 Creating Supercell
 Move towards -y axis
 Creating Supercell
 Move towards z axis
 Creating Supercell
 Move towards -z axis
 Creating Supercell

 Doing  Al   atom!

 Move towards x axis
 Creating Supercell
 Move towards -x axis
 Creating Supercell
 Move towards y axis
 Creating Supercell
 Move towards -y axis
 Creating Supercell
 Move towards z axis
 Creating Supercell
 Move towards -z axis
 Creating Supercell


Created  13  files


 **-------------------------------------------------------**
 ** JOB DONE  ^_^ . disp-xxx files have been created.     **
 **           Please check these folders!                 **
 **-------------------------------------------------------**
```

```
[YaokunYe@mgr module-test]$ ls -d disp-0*
disp-000  disp-002  disp-004  disp-006  disp-008  disp-010  disp-012
disp-001  disp-003  disp-005  disp-007  disp-009  disp-011
```

```
[YaokunYe@mgr module-test]$ ls *.sh
check-run.sh  ln-IN.WG.sh  run-disp.sh  run-hpsi.sh
```

./run-disp.sh，计算disp-xxx文件夹里的SCF任务

注：为保证脚本可以方便直接使用，请将SCF计算脚本的后缀命名为.pbs

进入disp-000文件夹，运行G-wann.x，将WFs写入G空间

```
[YaokunYe@mgr disp-000]$ ~/G-wann.x
 JOB 'WFs into G-space' Starts!
 Supercell grid is:           4           4           4
 Num_wann:           8
 **------------------------------------------------------------**
 ** Gspace WFs have been written in "OUT.WG" for all (ir1,ir2,ir3)  **
 **------------------------------------------------------------**
```

退出disp-000文件夹到上层文件夹
./ln-IN.WG.sh，会准备好hpsi文件夹里的文件

```
var=/share2/YaokunYe/ele-ph-module/ref/alas/disp-000
NUM_BAND=512
```

./run-hpsi.sh，在disp-xxx文件夹里计算JOB=HPSI

```
[YaokunYe@mgr hpsi]$ ls
aaa.pbs          As.SG15.PBE.UPF  etot.input    IN.VR  OUT.HPSI  OUT.NONSCF  OUT.SYMM
Al.SG15.PBE.UPF  atom.config      final.config  IN.WG  OUT.KPT   output      REPORT
```

计算完成后会得到OUT.HPSI文件，在后续计算电声耦合时需要用到

# check-run.sh

```
[YaokunYe@mgr test]$ ./check-run.sh
disp-000
 total computation time (sec)=         219
disp-001
 total computation time (sec)=         217
disp-002
 total computation time (sec)=         222
disp-003
 total computation time (sec)=         250
disp-004
 total computation time (sec)=         243
disp-005
 total computation time (sec)=         221
disp-006
 total computation time (sec)=         251
disp-007
 total computation time (sec)=         248
disp-008
 total computation time (sec)=         215
disp-009
 total computation time (sec)=         243
disp-010
 total computation time (sec)=         243
disp-011
 total computation time (sec)=         244
disp-012
 total computation time (sec)=         245
```

```
[YaokunYe@mgr test]$ ./check-run.sh
disp-000
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:23 OUT.HPSI
disp-001
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:23 OUT.HPSI
disp-002
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:24 OUT.HPSI
disp-003
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:24 OUT.HPSI
disp-004
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:23 OUT.HPSI
disp-005
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:23 OUT.HPSI
disp-006
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:24 OUT.HPSI
disp-007
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:24 OUT.HPSI
disp-008
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:27 OUT.HPSI
disp-009
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:23 OUT.HPSI
disp-010
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:24 OUT.HPSI
disp-011
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:24 OUT.HPSI
disp-012
-rw-rw-r-- 1 YaokunYe YaokunYe 1019883676 Jul 26 20:24 OUT.HPSI
```

在做完run-disp.sh和run-hpsi.sh系列任务时，可以使用./check-run.sh
检查每个子文件夹是否顺利计算完成

4. 回到SCF的目录下，运行calc_EPW.x，得到epw.out文件

```
[YaokunYe@mgr module-test]$ ~/calc_EPW.x
 calc_EPW.x starts!
 Num_atom:            2
 Num_wann:            8
 Num_super:            4           4           4
Program calculated      1/12
Program calculated      2/12
Program calculated      3/12
Program calculated      4/12
Program calculated      5/12
Program calculated      6/12
Program calculated      7/12
Program calculated      8/12
Program calculated      9/12
Program calculated     10/12
Program calculated     11/12
Program calculated     12/12
 Time used:   80.29237     (s)

 **------------------------------------**
 **--------calc_EPW.x DONE-------------**
 **------------------------------------**
```

程序会提示当前进度

因为电声耦合矩阵数量巨大
如无必要，不需要每次都遍历所有指标输出
因此我们最后准备一个epc.input的输入文件来确定输出范围
每次更改epc.input里的参数时，只需要重新运行calc_EPC.x即可

$$\left\langle \psi_{mk_1} \left| \frac{\partial H}{\partial v_{bq}} \right| \psi_{nk_2} \right\rangle$$

```
5 5
5 5
1 3
1 3
```

第一行为公式中m的取值范围
第二行为公式中n的取值范围
第三行为公式中k1的取值范围
第四行为公式中k2的取值范围
公式中b和q的取值范围为遍历

```
[YaokunYe@mgr module-test]$ ~/calc_EPC.x
 JOB calc_EPC.x starts!
 Num_wann:    8
 READ <Wp_dH/dR_Wq> from epw.out
 Checking WFs symmetry...OK!
 Total k1*k2*q is:       262144
 Paired (k1,q,k2):         4096
 Input information:    5  5  5  5  1  3  1  3
 Doing EPC calculation...
 Time used:    2.313307     (s)

 **------------------------------------------------------------**
 ** JOB DONE  ^_^ . <psi_mk1|dH/dv_bq|psi_nk2> has been written **
 ** Please check the file "OUT.EPC"                            **
 **------------------------------------------------------------**
```

运行calc_EPC.x，得到OUT.EPC文件，完成计算

OUT.EPC文件中的信息

$$\left\langle \psi_{mk_1} \left| \frac{\partial H}{\partial \nu_{bq}} \right| \psi_{nk_2} \right\rangle$$

前三行：满足k1+q=k2的k1、q、k2坐标

后续为3N，N为原胞原子数个声子支对应的电声耦合矩阵信息

```
k1:  0.0000000E+00  0.0000000E+00  0.0000000E+00
q:  0.0000000E+00  0.0000000E+00  0.0000000E+00
k2:  0.0000000E+00  0.0000000E+00  0.0000000E+00
 5   5   1   1   1   1   -7.06119  -7.06119   0.00000        0.00000000       0.00000000       0.00000000
 5   5   1   2   1   1   -7.06119  -7.06119   0.00000        0.00000000       0.00000000       0.00000000
 5   5   1   3   1   1   -7.06119  -7.06119   0.00000        0.00000000       0.00000000       0.00000000
 5   5   1   4   1   1   -7.06119  -7.06119  46.15522       -0.00001993       0.00000000       0.22525045
 5   5   1   5   1   1   -7.06119  -7.06119  46.15522       -0.00002653      -0.00000000       0.29986604
 5   5   1   6   1   1   -7.06119  -7.06119  46.15522        0.00001875      -0.00000000       0.21192353

k1:  0.0000000E+00  0.0000000E+00  0.0000000E+00
q:  0.0000000E+00  0.0000000E+00  0.2500000
k2:  0.0000000E+00  0.0000000E+00  0.2500000
 5   5   1   1  17   2   -7.06119  -6.45454   7.42228       -0.00000343      -0.00000084       0.09958246
 5   5   1   2  17   2   -7.06119  -6.45454   7.42228        0.00000053       0.00000068       0.02414633
 5   5   1   3  17   2   -7.06119  -6.45454  18.65086       -0.00579563       0.00126490     105.45862827
 5   5   1   4  17   2   -7.06119  -6.45454  45.52458        0.00000415       0.00000051       0.04754462
 5   5   1   5  17   2   -7.06119  -6.45454  45.52458       -0.00000050      -0.00000220       0.02568016
 5   5   1   6  17   2   -7.06119  -6.45454  49.54365       -0.01239736       0.00642492     152.30729738

k1:  0.0000000E+00  0.0000000E+00  0.0000000E+00
q:  0.0000000E+00  0.0000000E+00  0.5000000
k2:  0.0000000E+00  0.0000000E+00  0.5000000
 5   5   1   1  33   3   -7.06119  -5.55401   8.33448        0.00000195      -0.00000009       0.05190413
 5   5   1   2  33   3   -7.06119  -5.55401   8.33448       -0.00000011       0.00000022       0.00650775
 5   5   1   3  33   3   -7.06119  -5.55401  26.75983        0.00446475      -0.00024131      66.36128164
 5   5   1   4  33   3   -7.06119  -5.55401  45.43849       -0.00000162       0.00000001       0.01842305
 5   5   1   5  33   3   -7.06119  -5.55401  45.43849       -0.00000020       0.00000013       0.00273123
 5   5   1   6  33   3   -7.06119  -5.55401  49.45714       -0.01605756       0.00086792     175.55968079
```

如图所示，第一列和第二列对应m和n的序号
第三、四、五、六列依次为k1、b、q、k2的序号
第七列和第八列为k1、k2对应的本征能量(eV)
第九列为声子频率换算过来的能量(meV)
第十列和第十一列为电声耦合矩阵的实部和虚部(原子单位)
最后一列为电声耦合矩阵的模(meV)

用户如有需要，也可以使用degenerate.x
将输出的OUT.EPC“简并化”

```
[YaokunYe@mgr module-test]$ ~/degenerate.x
 **-------------------------------------------------------------**
 ** JOB DONE! Please check the file "NEW-OUT.EPC"               **
 **-------------------------------------------------------------**
```

此处“简并化”是指，如果电声耦合矩阵的k1、k2、q的能量都相同，则认为它简并

然后将它的模做一个算术平均，如图所示

```
k1:     0.0000     0.0000     0.2500
q:     0.0000     0.0000     0.2500
k2:     0.0000     0.0000     0.5000
  14    1       -6.4545      -5.5540       7.4223       0.0808
  14    2       -6.4545      -5.5540       7.4223       0.0808
  14    3       -6.4545      -5.5540      18.6509      50.8612
  14    4       -6.4545      -5.5540      45.5246       0.0394
  14    5       -6.4545      -5.5540      45.5246       0.0394
  14    6       -6.4545      -5.5540      49.5436     181.0789
```

```
k1:     0.0000     0.2500     0.0000
q:     0.0000     0.2500     0.0000
k2:     0.0000     0.5000     0.0000
  42    1       -6.4545      -5.5540       7.4223       0.0808
  42    2       -6.4545      -5.5540       7.4223       0.0808
  42    3       -6.4545      -5.5540      18.6509      50.8612
  42    4       -6.4545      -5.5540      45.5246       0.0394
  42    5       -6.4545      -5.5540      45.5246       0.0394
  42    6       -6.4545      -5.5540      49.5436     181.0789
```

和EPW的对比

输出内容：$\left\langle \psi_{mk_1} \left| \frac{\partial H}{\partial v_{bq}} \right| \psi_{nk_2} \right\rangle$

体系：AlAs

非简并电子态：m=n=5

| phonon mode | freq | EPW | PWMAT |
|---|---|---|---|
| 1 | 7.4222 | 0.0000 | 0.0207 |
| 2 | 7.4222 | 0.0000 | 0.0186 |
| 3 | 18.6508 | 50.5914 | 50.5965 |
| 4 | 45.5245 | 0.0000 | 0.0142 |
| 5 | 45.5245 | 0.0000 | 0.0173 |
| 6 | 49.5436 | 183.4980 | 181.5947 |