

# Optimización de parámetros en cálculo SCF

```
scp espresso@192.168.1.16:TutorialQ-E/Practicas/pract3.pdf •  
evince pract3.pdf &
```

## **Ejemplos en la distribución de Quantum-ESPRESSO**

Los ejemplos de espresso barren la mayor parte de las capacidades del paquete.

Cambiase al directorio de Quantum-ESPRESSO y

```
cd BLABLABLA/espresso-4.0.4
ls
cd examples/
less README
```

Lea rapidamente el README. Ahora vamos a correr el ejemplo 01.

**Nota:** Si ha instalado el espresso en un directorio fuera de la cuenta de usuario, por ej. en /usr/local, entonces no podrá escribir nada dentro de ese directorio. Entonces, copie el espresso a su directorio de usuario.

**Ejemplo de como copiarlo (antes vea la nota roja):**

```
cp -r /usr/local/espresso-4.0.4 $HOME/TutorialQ-E/.
```

Si tiene limitaciones de disco (p.e. Virtual Box), solamente, mueva en lugar de copiar.

```
mv -r /usr/local/espresso-4.0.4 $HOME/TutorialQ-E/.
```

y arregle la linea la variable PATH en ~/.bashrc

## Ejemplo 01

```
cd example01
less README      (infórmese del contenido del ejemplo)
./run_example
```

Examinemos lo que hizo run\_example. Para esto abra una nueva terminal

y desde el directorio example01 haga

```
cd results
ls
less si.scf.david.in
```

Opcional: Vea en el archivo INPUT\_PW.txt la explicación de cada una de las keywords utilizadas.

#Visualizacion de la estructura

```
xcrysden --pwi si.scf.david.in
```

Si no se ven los enlaces Si-Si, ir al menu Modify/Atomic radius y aumentar

chemical connectivity factor a 1.5.

Jugar un poco con la estructura

#Veamos como un mínimo error (frecuente) del input cambia la estructura.

```
gedit si.scf.david.in
```

Cambiar **ATOMIC\_POSITIONS** -> **ATOMIC\_POSITIONS bohr**

visualizar de nuevo. ¿Cual es el error?

```
ls $HOME/tmp/ # Vea lo que hay en el directorio outdir
```

Editamos si.scf.david.in, borremos bohr para dejarlo en el estado original y cambiemos : **outdir='.'**,

Visualicemos con xcrysden para asegurarnos.

Ahora corramos manualmente pw.x

```
pw.x <si.scf.david.in >si.scf.david.out2 &
```

El signo & permite correr el background. Inspeccione la salida con  
`less si.scf.david.out2`

## Calculo de bandas (en example01).

Etapas del cálculo de bandas

**Calculo SCF -> Calculo de bandas -> postprocesamiento y gráficos.**

**Paso 1.** Cálculo SCF. `pw.x < input.scf.in > output.scf.out`

Se obtiene  $E$ ,  $n(\mathbf{r})$ , salvados en el outdir/prefix.save

**&CONTROL**

```
calculation = 'scf' ,  
restart_mode = 'from_scratch' ,  
  outdir = './' ,  
  pseudo_dir = '.',  
  prefix = 'si' ,
```

.....

Esto ya lo hicimos. Verifique el si.scf.david.in

verifique que outdir = '.'

**Paso 2.** Manteniendo constante la  $n(\mathbf{r})$  (esto es un cálculo no autoconsistente) obtenida en el cálculo SCF, calcular las energías  $\epsilon_{n,k}$  a lo largo de una serie de puntos  $\mathbf{k}$ .

Copiar el archivo input.scf.in en uno nuevo input.bands.in y hacerle algunos cambios

```
calculation = 'scf' , -> calculation = 'bands' ,  
nbnd -> nbnd = Numero-de-bandas-que-desea-obtener.  
K_POINTS automatic -> K_POINTS  
Npuntos. Vea el INPUT_PW.txt
```

Vea, en el ejemplo, la diferencia entre los dos archivos

```
diff si.band.david.in si.scf.david.in
```

Luego veremos como generar los k-points on XCRYSDEN.



Demostremos como se usa XCRYDEN para generar los puntos k.

Q: How to construct a PWscf's k-path for band-structure calculation with XCRYSDEN ?

• A:

–select menu:

• Tools-->k-path Selection

–select a k-path by mouse-clicking

–specify how many k points to be

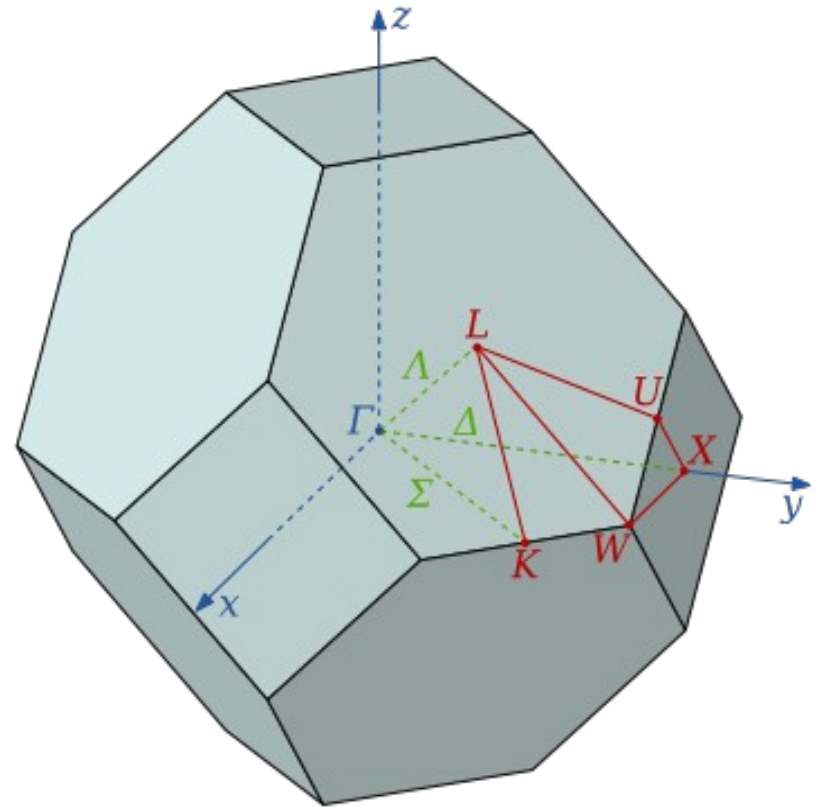
calculated (Ex: 100)

–save the file with .pwscf extension

(**BEWARE: .pwscf extension is mandatory to get the pw.x formatted k-list ,otherwise the file is not written )**

Haga la secuencia

Gamma -> X -> U -> L -> Gamma



### Paso 3. Postprocesamiento

Ahora vea si.bands.in . Este es un input para el programa de postprocesamiento, que se lee los autovalores de Kohn-Sham, les realiza un analisis de simetria (ver si.bands.out) y los guarda en un fichero de datos si.band. si.bands.in es así:

```
&inputpp  
  prefix='silicon',  
  outdir='.'  
  filband='si.band' ! aqui dice donde se escribe  
  lsym=.true.,  
/
```

**bands.x < si.bands.in > si.bands.out**

Vea si.bands.out

Luego lea si.bands, que tiene los datos de las bandas

El formato de si.band es ms o menos claro

```
&plot nbnd= 8, nks= 28 / # Hay 8 bandas y 28 ptos k
          0.000000 0.000000 0.000000 # kx,ky,kz (en que
unidades? vea INPUT_PW.txt )
-5.810 6.255 6.255 6.255 8.822 8.822 8.822 9.723
# energia de las 8 bandas
```

Aqui esta toda la informacion. Para obtener un grafico usemos el utilitario plotband.x.

```
plotband.x
```

El programa le pide datos. Escribale los datos despues de cada ">"

```
Input file > si.band
```

```
Reading      8 bands at    28 k-points
```

```
Range: -5.8100 16.4070eV Emin, Emax > -6,20
```

```
high-symmetry point:  0.0000 0.0000 0.0000
```

```
.....
```

```
output file (xmgr) > si.bands.xmgr
```

```
lines nrap 1 5
```

```
.....
```

```
bands in xmgr format written to file si.bands.xmgr
```

```
output file (ps) > si.bands.ps
```

```
Efermi > 6.2549 (es el valor del HOMO. Encuentre este dato en el  
output del calculo SCF si.david.scf.out)
```

```
deltaE, reference E (for tics) 1.0 6.255
```

```
bands in PostScript format written to file
```

```
si.bands.ps
```

Puede hacer la misma corrida si hace este input file llamado si.plotband.in con el siguiente contenido

```
si.band  
-6.0 20  
si.bands.xmgr  
si.bands.ps  
6.2549  
1.0 6.255
```

y ejecuta

```
plotband.x < si.plotband.in
```

Nota: 6.2549 el maximo de la banda de valencia. Eso se sabe examinando si.scf.david.out o si.band. Si el material es metalico, pw.x lo escribe en la salida. Si es aislante como el Si, hay que buscar manualmente el maximo. En todo caso, el efecto de esto es establecer el 0 de energias, que convencionalmente se pone en el maximo de la banda de valencia o en el nivel de Fermi.

Visualizamos las bandas con

**display si.bands.ps**

(prefiero usar gv en lugar de display, pero no esta instalado. Creo que puede usar evince también.)

**Use su editor preferido para editar el archivo postscript y ponerle las letras griegas.**

**Si quiere usar otro programa, los datos de cada banda estan en si.bands.xmlgr.\***

**Si no le satisface plotbands.x (como a mi), para plotear las bandas usted puede hacer su propio programita que lea si.band, ya ESPRESSO le hizo la parte de resolver autoconsistentemente las ecs de KS y el calculo no autoconsistente de los autovalores.**

En example05 puede ver el mismo proceso del ploteo de las bandas del Si. Lea en run\_example de ese directorio.

#Entendamos lo que hizo run\_example

desde example01/results

```
cd ../
```

```
less run_example
```

run\_example es un archivo de comandos (script) del shell sh

Ver en especial las secciones

```
./environment_variables
```

```
# required executables and pseudopotentials
```

```
# check for executables
```

```
for diago in david cg ; do
```

```
    cat > si.scf.$diago.in << EOF
```

```
$PW_COMMAND < si.scf.$diago.in > si.scf.$diago.out
```

Luego vea la forma en que se usa si.band.david.in

En breve, si.band.david.in es el input para un calculo no

autoconsistente en que pw.x, lee

la densidad de carga calculada antes, (calculation='scf') y calcula los

autovalores de Kohn-Sham para cada pto k especificado siguiendo

una linea en la 1ra ZB.

## Conclusión

Calcular bandas con Q-ESPRESSO es bastante trabajoso. Se agradecería un programa interactivo que lea los parametros básicos y mande a correr pw.x las dos veces, bands.x, y plotbands.x

! Eso es precisamente lo que hace el script example05/run\_example !  
(se puede hacer mas simple)

Reformulo mi íntimo deseo. Se agradecería un extensión a pwgui para hacer el script al estilo de run\_example. Tal vez me anime a escribirle un email a Tone Kokalj.



Red inalámbrica: bituos

Password: 6839725308

Numeros IP 192.168.1.X (X el mismo que antes, a partir de 50).

Gateway 192.168.1.1

Router: 192.168.1.253

para verificar que hay red, dar

**ping 192.168.1.253**

## Paso 0. Optimización de parámetros.

Usaremos un script para optimizar los parametros de cutoff y k-point, basados en el calculo del Si. En el ejemplo anterior no estan optimizados para que haya convergencia.

```
cd TutorialQ-E/Practicas/
```

```
scp -r espresso@192.168.1.16:TutorialQ-E/Practica3/ .
```

```
ls
```

```
cd Cutoff
```

Ejecute el programa

Examine el si-cutoff.sh para entenderlo un poco

```
./si-cutoff.sh
```

y vea la salida en cutoff.dat

Vera que faltan numeros. Investigue por qué. Lea el script si-cutoff.dat y encuentre como se llaman los archivos de salida de pw.x. Abra uno de esos archivos y vea el error que da. !Falta el pseudopotencial!

Busquelo en la distribucion de espresso-4.0.4 y cópielo al directorio ./

Repita el cálculo.

## Convergencia de las fuerzas y la energía

En el cutoff.dat las fuerzas son todas 0. Esto ocurre porque todos los átomos están en posiciones de alta simetría.

Desplace ligeramente ( $\sim 0.01 a_0$ ) el primer átomo de Si y corra de nuevo el script. Vea la convergencia de las fuerzas. Compare los tiempos de cálculo ?Por qué es mas lento?

A partir de los resultados en cutoff.dat, **determine el parámetro ecutwfc necesario para obtener la energía total convergida en 0.01 eV/atomo y el ecutwfc para obtener la fuerza convergida en 0.01 eV/angstrom.** Recuerde que pw.x usa Ry como unidad de energía y Ry/bohr para fuerzas.

## Convergencia respecto a los puntos k

```
cd ../K-points
```

Edite el archivo `si-kpoints.sh`

Modifíquelo para estudiar la convergencia respecto a los k-points. Use el `ecutwfc` que determinó en el ejercicio anterior. La sección `K_POINTS` debe quedar

```
K_POINTS automatic
```

```
  ${Kp} ${Kp} ${Kp} 1 1 1
```

donde  $N$  es la variable que se hace variar en el script. Póngale valores entre 1 y 30. en el script se

```
for Kp in 5 10 15 20 25 30;done
```

```
.....
```

```
K_POINTS automatic
```

```
 ${Kp} ${Kp} ${Kp} 1 1 1
```

```
.....
```

```
done
```

Despues de correr exitosamente el script, determine el valor de  $N$  que permite la convergencia de las energías y fuerzas con el mismo criterio que se usó para optimizar ecutwfc.

Repita en cálculo usando

```
K_POINTS automatic  
N N N 0 0 0
```

Determine el valor de  $N$  que da la misma precisión que el  $N$  determinado antes. ¿Cual es mayor?

Lo que hemos hecho es comparar dos formas de grillas de puntos  $k$ .

$$\vec{k} = \frac{n_1 + \alpha_1}{N_1} \vec{b}_1 + \frac{n_2 + \alpha_2}{N_2} \vec{b}_2 + \frac{n_3 + \alpha_3}{N_3} \vec{b}_3, \quad \alpha_i = 0 \text{ si } 0 \ 0 \ 0 \text{ ó } 0.5 \text{ si } 1 \ 1 \ 1.$$

Ahora puede repetir el example01 usando los valores optimizados.

Ver individualmente

**example05: carga y otras propiedades**

example08: Densidad de Estados

Basados en el ejemplo8, podemos calcular manualmente la densidad de estados del Si. En el run\_examples podemos ver las ordenes

```
$PW_COMMAND < ni.scf.in > ni.scf.out  
$PW_COMMAND < ni.dos.in > ni.dos.out  
$DOS_COMMAND < ni.dos2.in > ni.dos2.out
```

Esto significa que el proceso es así

- 1) Cálculo SCF con pw.x, (ni.scf.in)
- 2) Cálculo no SCF con pw.x, input ni.dos.in
- 3) Cálculo con dos.x: si.dos2.in

Despues el run\_example hace otros cálculos.

Corra el ejemplo

```
./run_example
```

```
cd results
```

```
ls
```

Si ve muchos \*.in y \*.out y los \*.out no muestran errores, todo esta bien.

Veamos la diferencia entre los inputs calculo SCF y el no SCF

```
diff ni.dos.in ni.scf.in |more
```

```
2c2,3
```

```
<      calculation='nscf'
```

```
---
```

```
>      calculation='scf'
```

```
>      restart_mode='from_scratch',
```

```
10,11c11,12
```

```
<      ecutwfc = 24.0, ecutrho = 288.0, nbnd=8,
```

```
<      occupations='tetrahedra'
```

```
---
```

```
>      ecutwfc = 24.0, ecutrho = 288.0,
```

```
>      occupations='smearing', smearing='methfessel-paxton',
```

```
degauss=0.02
```

```
21,22c22,83
```



```
< K_POINTS {automatic}
< 12 12 12 0 0 0
---
> K_POINTS
> 60
> 0.0625000 0.0625000 0.0625000 1.00
> 0.0625000 0.0625000 0.1875000 3.00
```

Entonces, el ni.scf.in y el ni.dos.out se diferencian en:

1) calculation='scf' --> calculation='nscf'  
2) occupations='smearing', smearing='methfessel-paxton', degauss=0.02 -->>  
occupations='tetrahedra'

3) K\_POINTS  
> 60  
> 0.0625000 0.0625000 0.0625000 1.00  
> 0.0625000 0.0625000 0.1875000 3.00

--->

```
K_POINTS {automatic}
< 12 12 12 0 0 0 (en el calculo de DOS se usa una grilla mas densa)
```

Hagamos el proceso para el silicio.

En el directorio Practica3/DOS encontrara ficheros de entradas que debe arreglar para obtener la DOS

La secuencia es

```
pw.x < si.scf.in |tee si.scf.out
```

```
pw.x <si.dos.in |tee si.dos.out
```

```
dos.x <si.dos2.in|tee si.dos2.out
```

La DOS queda en el archivo silicon.dos, para graficarlo de el comando gnuplot

le aparece un programa interactivo con el prompt gnuplot>

Tiene que dar comandos. Para graficarlo se dan las siguientes lineas de comando

```
gnuplot> set style data lines
```

```
gnuplot> plot "silicon.dos" u 1:2,"silicon.dos" u 1:3
```

## **Ejercicio STM**

Copiar el dir Practica4/3-data-analysis.\*

```
evince 3-data-analysis.pdf &  
tar xzf 3-data-analysis.tgz  
cd Ex1-STM
```

Siga las instrucciones del PDF.

Ayudese del example16.

## **Tarea evaluativa para entregar el lunes (si quiere una evaluación)**

Realizar el calculo autoconsistente de la densidad de carga y la energia total para uno de los siguientes materiales simples:Al,Cu,Fe,Ag,Na,Ca,...

Debe realizar los siguientes pasos.

- 0) Busque el pseudopotencial para uno de los metales mencionados.
- 1) Busque la estructura cristalina del material en alguna fuente confiable: libro de fisica de solidos, [www.google.cl](http://www.google.cl), [www.webelements.com](http://www.webelements.com), base de datos cristalografica como ICSD, [prola.aps.org](http://prola.aps.org), [scholar.google.cl](http://scholar.google.cl), etc.
- 3) Haga el fichero de input, busque los pseudopotenciales o hagalos, visualice la estructura.
- 4) Determine los parametros optimos para su cálculo, especialmente cutoffs de funcion de onda y de densidad, k-points.
- 5) Haga un grafico de energia vs volumen (+-20% alrededor del volumen experimental) y ajustelo con la ecuacion de Birch-Murnaghan.

$$E(V) = E_0 - \frac{9}{16}B_0 \left[ (4 - B'_0) \frac{V_0^3}{V^2} - (14 - 3B'_0) \frac{V_0^{7/3}}{V^{4/3}} + (16 - 3B'_0) \frac{V_0^{5/3}}{V^{2/3}} \right]$$

- 6) Haga un reporte de sus resultados, junto con un archivo zip o tar.gz con los inputs y outputs.

## Optimización de Geometría

Ejercicio 1. example03: Geometria

```
cd BLABLABLA/espesso-4.0.4/examples/example03  
./run_example
```

Visualice las estructuras finales con xcrysden

```
xcrysden --pwo co.rx.out
```

**Utilice la opcion de reducir a 0D y vea la diferencia con no usarla.  
Haga animacion del proceso de relajacion estructural.**

```
xcrysden --pwo a1001.mm.out
```

Utilice la opcion de reducir a 2D y vea las diferencias con no utilizarla.  
Haga una animacion del proceso de relajacion.

Vea las diferencias entre los dos metodos para relajar la geometria.

```
diff a1001.rx.in a1001.mm.in
```

Vea los tiempos de calculo en la salida de pw.x.

## Sistemas con degeneración. Efectos de entropía.

Ejercicio 2.

example11: Oxigeno aislado (atomo)

Hagamos algunas modificaciones para permitir otras ocupaciones o para dejar que el programa encuentre las ocupaciones optimas.

Si se usa smearing, lease lo siguiente

Lo correcto es hacer varios calculos con distintos degauss y extrapolar al valor degauss=0. Se sugiere usar dgauss=0.02,0.01,0.005, plotear las energias y extroplar hasta 0. Una funcion lineal debe ser suficiente para extrapolar.

Vea smearing contrib. (-TS) en la salida de pw.x.