

Fonones y espectroscopía Raman en semiconductores bidimensionales.

casimir

March 28, 2021

Contents

Abstract

Los materiales bidimensionales (2D) como el grafeno son de gran interés tanto por sus propiedades físicas exclusivas como por sus aplicaciones potenciales. El estudio de la dinámica de la red cristalina (fonones) de estos materiales es un requisito previo para entender su estabilidad estructural y propiedades térmicas, así como sus propiedades de transporte y ópticas.

Este Trabajo de Fin de Grado consiste en la computación de los modos vibracionales de materiales semiconductores 2D y su correlación con los observables relevantes para la interpretación de los experimentos de dispersión de luz.

La redacción del TFG todavía se encuentra en una versión muy preliminar, de hecho es más un bloc de notas que lo que espero que sea el TFG final.

Para redactar este trabajo estoy usando **org-mode** (junto con `org-babel`, `org-ref`, ...): un modo de edición del editor *Emacs*, que, entre otras muchas cosas, permite usar de manera nativa \LaTeX escribir un mismo fichero tanto el *texto* cómo el *código* (en muchos lenguajes, permitiendo también metaprogramación ...), pudiendo insertar si así se desea la salida de los programas en el mismo documento, y exportar el resultado final a varios formatos (en particular \LaTeX) mostrando u ocultando los bloques con el código en el pdf final.

1 Lecturas y apuntes de repaso

Links a los documentos, no es la bibliografía.

La bibliografía va en el fichero TFG-Casimir.bib y la gestiono (así como las referencias) con el paquete org-ref: entre otras cosas añade la referencia en formato bibtex automáticamente al fichero .bib y copia el pdf a mi biblioteca (renombrando el fichero) simplemente arrastrando el pdf al buffer de emacs donde tenemos abierto este fichero. Podemos también insertar manualmente ('M-x crossref-lookup') el título del artículo, lo busca en internet, lo baja si es posible (se pueden añadir fuentes no oficiales), y nos genera la entrada bibtex.

1.1 Bibliografia basica

- Phonons Theory and Experiments
- wirtz2004
- Symmetry constraints on phono dispersion in graphene
- Bond softening in monolayer graphite formed on transition-metal carbide surfaces
- Phonon dispersions in graphene sheet and single-walled carbon nanotubes
- Matriz de fuerzas, interpretacion, simetrias ... (pag 29)

1.2 Repaso de Estado Sólido

- Tema 1 de Estado Sólido
- Tema2. Vibraciones atómicas en cristales

2 TODO Introduccion

Por ahora solo algunas ideas que hay que desarrollar.

La idea que tengo es explicar el modelo de Born y Von karman (1921), tal y como viene explicado en Brüesch, 1982, pagina 4.

Luego explicar que haciendo uso de la aproximación adiabática, ibid. página 6, podemos, siempre que sea válida tal aproximación -explicar cuando lo es y cuando deja de serlo-, considerar los nucleos ionicos y los electrones de valencia como constituyentes independientes del sólido, de manera que podemos escribir la energía potencial (o el Hamiltoniano) como una suma de las distintas contribuciones, y que para

Pasar a describir la aproximación armónica: escribir el potencial, las ecuaciones de movimiento de Lagrange, y pasar de estas ecuaciones de movimiento en el espacio de posiciones al problema de valores propios de la matriz dinámica en el espació de vectores de onda.

Una vez concluida la introducción teórica general, valida para cualquier cristal, escribir el potencial separando el "stretching" y "bendig" y escribir la matriz dinámica, teniendo en cuenta que debido a la simetria del cristal la matriz de constantes de fuerza (y su transformada de Fourier, la matriz dinámica), debe cumplir ciertas relaciones (forma de la matriz, elementos nulos, relación entre elementos, ...), Aizawa et al., 1990 (en el apendice). Hay que hacer notar que para expresar la matriz de constantes de fuerza en coordenadas cartesianas hay que hacer el correspondiente cambio de coordenadas desde las radiales-transversales.

Las vibraciones reticulares están regidas por las fuerzas que experimentan los átomos cuando se desplazan de su posición de equilibrio. La primera hipótesis es que cada átomo tiene una posicion de equilibrio en el cristal, y consideraremos que estos átomos vibran con una amplitud pequeña (en comparación con la distancia interatómica) alrededor de su posición de equilibrio, de manera que el sólido se encuentra en estados que corresponden a lo que macroscópicamente se conoce como *la región de comportamiento elástico lineal*, donde se verifica la ley de Hooke.

Podremos por tanto aproximar la energía potencial de interacción por el termino armónico de su desarollo en serie de potencias del desplazamiento **desarrollar como a partir de la energia potencial del cristal \mathcal{U} , usando una aproximacion armonica, obtenemos la ecuacion secular para los fonones..**

Un fonón es un modo de vibración cuantizado que tiene lugar en una red cristalina. El estudio de los fonones juega un papel muy importante en la física del estado sólido porque los fonones tienen una gran importancia en muchas propiedades físicas de los sólidos. Los fonones son una versión mecano-cuantica de los modos normales de vibración de la mecánica clasica, donde cada parte de la red oscila con la misma frecuencia. Estos modos normales son importantes porque cualquier movimiento vibracional de la red puede describirse como una superposición de modos normales de distinta frecuencia, en este sentido son la base de las vibraciones de la red.

2.1 TODO Matriz dinámica

La matriz dinámica es una cantidad central en la dinámica reticular: las frecuencias de los fonones (vibraciones reticulares cuantificadas) se calculan a partir de los valores propios de la matriz dinámica:

$$\sum_{\alpha'} D_{\alpha\alpha'} \cdot \vec{e}_{\alpha'}(\vec{q}) = \omega^2 \vec{e}_{\alpha}(\vec{q}) \quad (1)$$

Por lo tanto, las frecuencias ω como función de vector de ondas \vec{q} del fonón son solución de la ecuación secular:

$$\det \left| \frac{1}{\sqrt{M_{\alpha} M_{\alpha'}}} D_{\alpha\alpha'}^{ij}(\vec{q}) - \omega^2(\vec{q}) \right| \quad (2)$$

donde M_{α} es la masa del átomo α y la matriz dinámica viene definida por:

$$D_{\alpha,\alpha'}^{i,j} = \frac{\partial^2 E}{\partial u_{\alpha}^{*i}(\vec{q}) \partial u_{\alpha'}^j(\vec{q})} \quad (3)$$

donde u_{α}^i representa el desplazamiento del átomo α en la dirección i .

La segunda derivada de la energía de la ecuación ?? corresponde al cambio en la fuerza que actúa sobre el átomo α' en la dirección j cuando se desplaza el átomo α en la dirección i

$$D_{\alpha\alpha'}^{ij}(\vec{q}) = \frac{\partial}{\partial u_{\alpha}^{*i}} F_{\alpha'}^j(\vec{q}) \quad (4)$$

Puesto que el cálculo de los modos de vibración por primeros principios empieza por establecer la geometría del cristal en equilibrio, vamos a comprobar que con los datos proporcionados que el BN monolayer se trata de un cristal bidimensional de base diatómica, cuya celda unidad viene dada por (datos proporcionados):

$$\vec{a}_1 = a(1, 0); \quad \vec{a}_2 = a \left(-\frac{1}{2}, \frac{\sqrt{3}}{2} \right) \quad (5)$$

```
import numpy as np
from numpy import array, sqrt, sort, vdot, pi, arccos
from numpy.linalg import norm
import pandas as pd
import matplotlib
from matplotlib import pyplot as plt
```

```
a=1
a_1=np.array([a,0])
a_2=np.array([-a/2,sqrt(3)*a/2])
```

Podemos comprobar que efectivamente se trata de una celdilla hexagonal, pues los dos vectores base forman un ángulo de $2\pi/3$ rad

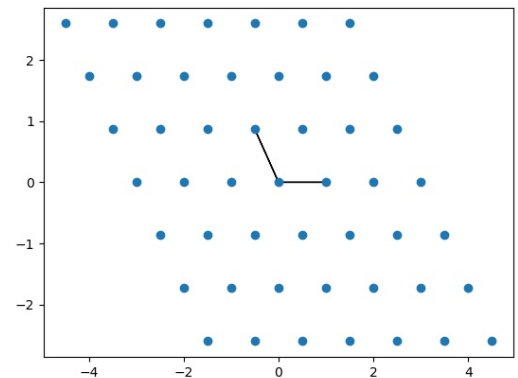
Numeraremos las celdillas unidad con un índice vectorial $\vec{l} = (l_1, l_2)$, notemos que aunque es habitual representar la celdilla con un índice entero n es más sencillo y facilita los cálculos (así como determinar a que celdilla no referimos) el uso de un índice vectorial. Las posiciones de los nudos son $\vec{R}_{\vec{l}} = l_1 \vec{a}_1 + l_2 \vec{a}_2$.

Visualizamos una región de la red hexagonal, con los correspondientes nudos (que no átomos), así como la correspondiente celda unidad,

```
def R_l(l_1,l_2):
    return l_1*a_1+l_2*a_2

rednudos=array([R_l(l_1,l_2) for l_1 in range(-3, 4)
                 for l_2 in range(-3,4)])

x = rednudos[:,0]
y = rednudos[:,1]
plt.plot(x,y,"o")
ax = plt.axes()
ax.arrow(R_l(0,0)[0],R_l(0,0)[1],
         R_l(1,0)[0],R_l(1,0)[1])
ax.arrow(R_l(0,0)[0],R_l(0,0)[1],
         R_l(0,1)[0],R_l(0,1)[1])
plt.savefig("Graficas/Reddenudos.jpg")
plt.close()
```



Hacer figuras de la red recíproca, primera zona de Brillouin (y puntos especiales, Gamma, K, K', M).

Para calcular los modos de vibración por primeros principios debemos determinar primero las posiciones atómicas de equilibrio en la celda unidad **nota: proporcionadas como datos**

Los átomos están situados en:

$$\begin{aligned}\vec{R}_B &= \frac{1}{3}\vec{a}_1 + 2\vec{a}_2 \\ \vec{R}_N &= \frac{2}{3}\vec{a}_1 + \frac{1}{3}\vec{a}_2\end{aligned}\tag{6}$$

```
R_B=1/3*a_1+2/3*a_2
R_N=2/3*a_1+1/3*a_2
```

Las posiciones de equilibrio de los átomos de la base respecto de su nudo son \vec{R}_α^0 , con $\alpha = 1, 2$, puesto que la base tiene 2 átomos, el 1 hará referencia a los átomos de B y 2 a los de átomos de N (notemos que aunque los átomos fuesen idénticos tendríamos que especificar a que átomo de la base nos referimos, puesto que no ocupan posiciones equivalentes).

Las posiciones de equilibrio de los átomos: $\vec{R}_{\alpha,\vec{l}} = \vec{R}_{\vec{l}} + \vec{R}_\alpha^0$ así como los desplazamientos atómicos: $\vec{u}_{\alpha,\vec{l}}$ quedarán por tanto identificados por medio de dos índices.

Pasamos ahora a representar la red de átomos:
#Posiciones de equilibrio de los átomos

```
def R_alpha_l(alpha,l_1,l_2):
    if alpha == 1:
        return l_1*a_1+l_2*a_2+R_B

    elif alpha == 2:
        return l_1*a_1+l_2*a_2+R_N

    else:
        print("Error, alpha solo puede ser 1 o 2 ")

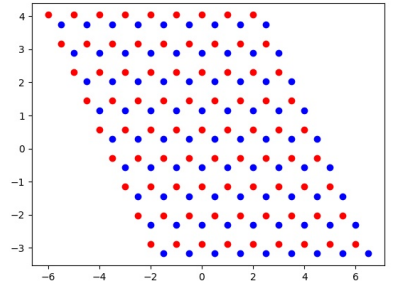
AtomosB=array([R_alpha_l(1,l_1,l_2) for l_1 in range(-4, 5)
               for l_2 in range(-4,5)])

AtomosN=array([R_alpha_l(2,l_1,l_2) for l_1 in range(-4, 5)
               for l_2 in range(-4,5)])

xB = AtomosB[:,0]
yB = AtomosB[:,1]
plt.plot(xB,yB,"o",color="red")

xN = AtomosN[:,0]
yN = AtomosN[:,1]
plt.plot(xN,yN,"o",color="blue")

plt.savefig("Graficas/Reddeatomos.jpg")
plt.close()
```



Las dimensiones del cristal son $L_1 = N_1 a_1$ y $L_2 = N_2 a_2$, donde N_i ($i = 1, 2$) es el número de celdillas en la dirección \vec{a}_i . El cristal tiene $N = N_1 N_2$ celdillas unidad primitivas y $2N$ átomos.

La idea básica es que si la base tiene r átomos entonces debemos plantear y resolver las ecuaciones de movimiento de los r átomos de la base de la celdilla $\vec{0}$, por lo tanto en el caso que

estamos estudiando debemos resolver 2 ecuaciones vectoriales de movimiento: una para el átomo de B y la otra para el de N .

Falta reescribir completamente y ampliar mucho la introducción teórica

Notar que aunque el cristal sea bi-dimensional, los átomos de este pueden vibrar en las 3 direcciones espaciales.

Explicar que podemos tratar por un lado las vibraciones en el plano del cristal y por otro las vibraciones perpendiculares a este, ya que se trata de vibraciones completamente desacopladas.

Que la ecuación secular tiene $3N_\nu = 6$ soluciones que describen las 6 ramas de la relación de dispersión, es decir, las 6 frecuencias características de los 6 modos normales de vibración de vector de onda \vec{q} . Se cumple que el número total de modos normales de vibración coincide con el triple del número total de átomos, es decir, *con el de grados de libertad de movimiento de los átomos*

Puesto que la energía potencial es una función cuadrática Falkovsky, 2008

3 TODO Desarrollo del trabajo

3.1 Clasificación de los vecinos

Puesto que debemos determinar cuales son las posiciones de equilibrio de los átomos más cercanos a los átomos de la celda $\vec{0}$ antes que nada genero un array con los datos que voy a necesitar, ordenando las filas de manera creciente a la distancia a cada uno de los 2 átomos de la celda $l = \vec{0}$ hasta los cuartos vecinos, y guardando el array con la información como un DataFrame de pandas, que facilita mucho la manipulación de los datos.

```
from sympy import *
import pandas as pd

## Parametros de la red, de la celdilla y del cristal
a=Symbol('a', real=True, positive=True)
q_x=Symbol('q_x', real=True); q_y=Symbol('q_y', real=True)
q=Matrix([q_x,q_y])
a_1=Matrix([a,0]); a_2=Rational(1,2)*Matrix([-a,sqrt(3)*a])
R_B=Rational(1,3)*a_1+Rational(2,3)*a_2; R_N=Rational(2,3)*a_1+Rational(1,3)*a_2

## Masas de los átomos, frecuencia, ...
M_B, M_N, omega=symbols("M_B, M_N, omega") #masa de los átomos de Boro y N.
def masa(alpha):
    if alpha == 1:
        return M_B

    elif alpha == 2:
        return M_N

    else:
        print("Error, alpha sólo puede se 1 o 2")

## Vector R_l (vector de traslación primitivo)
def R_l(l_1,l_2):
    return l_1*a_1+l_2*a_2

## Vector de posición de los átomos del cristal (en equilibrio)
def R_alpha_l(alpha,l_1,l_2):
    if alpha == 1:
        return l_1*a_1+l_2*a_2+R_B

    elif alpha == 2:
        return l_1*a_1+l_2*a_2+R_N

    else:
        print("Error, alpha solo puede ser 1 o 2 ")

## Vector unitario que une uno de los átomos en la celdilla 0 con el átomo considerado
def R_hat(alphaprima,alpha,l_1,l_2):
    if (R_alpha_l(alpha,l_1,l_2)-R_alpha_l(alphaprima,0,0)).norm()>0:
        return (R_alpha_l(alpha,l_1,l_2)-R_alpha_l(alphaprima,0,0))/(R_alpha_l(alpha,l_1,l_2)
```

```

-R_alpha_l(alphaprima,0,0)).norm()

else:
    return (R_alpha_l(alpha,l_1,l_2)-R_alpha_l(alphaprima,0,0))

def fase(l_1,l_2):
    return exp(I*q.dot(R_l(l_1,l_2)))

## Finalmente construyo un DataFrame de pandas con la información necesaria para
## identificar a los primeros, segundos, ... vecinos, según su distancia a cada uno
## de los átomos de la celdilla unidad
def propiedades_atomos(l_1, l_2):
    return [(k, m, i, j, R_hat(k,m,i,j),fase(i,j), (R_alpha_l(m,i,j)-R_alpha_l(k,0,0)).norm()),
            for k in [1,2] for m in [1,2] for i in range(-l_1,l_1+1) for j in range(-l_2,l_2+1)]

columnas = [r"$\alpha\prime$",r"$\alpha$",r"$l_1$", r"$l_2$",r"$\hat{R}_{\nu\prime,\nu,\vec{l}}$",
            'Fase', 'Distancia']

def Atomos(l_1, l_2):
    return pd.DataFrame(propiedades_atomos(l_1,l_2),columns=columnas).sort_values(
        ['Distancia',r"$\alpha\prime$"], ascending=[True, True])
## Mostramos el dataframe como una tabla en formato \LaTeX.
Atomos(2,2).head(26).to_latex(escape=False,float_format="{:0.4f}".format,index=False)

```

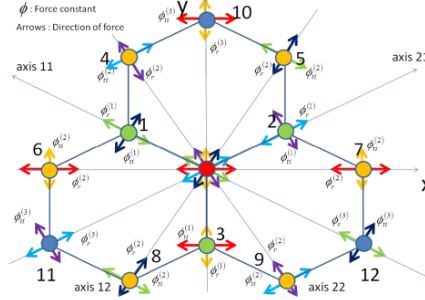
α'	α	l_1	l_2	$\hat{R}_{\nu\prime,\nu,\vec{l}}$	Fase	Distancia
1	1	0	0	[0, 0]	1	0
2	2	0	0	[0, 0]	1	0
1	2	-1	0	$[-\sqrt{3}/2, -1/2]$	$\exp(-I^*a^*q_x)$	$\sqrt{3}/3$
1	2	0	0	$[\sqrt{3}/2, -1/2]$	1	$\sqrt{3}/3$
1	2	0	1	[0, 1]	$\exp(I^*(a^*q_x/2 + \sqrt{3} * a * q_y/2))$	$\sqrt{3}/3$
2	1	0	-1	[0, -1]	$\exp(I^*(a^*q_x/2 - \sqrt{3} * a * q_y/2))$	$\sqrt{3}/3$
2	1	0	0	$[-\sqrt{3}/2, 1/2]$	1	$\sqrt{3}/3$
2	1	1	0	$[\sqrt{3}/2, 1/2]$	$\exp(I^*a^*q_x)$	$\sqrt{3}/3$
1	1	-1	-1	$[-1/2, -\sqrt{3}/2]$	$\exp(I^*(a^*q_x/2 - \sqrt{3} * a * q_y/2))$	1
1	1	-1	0	[-1, 0]	$\exp(-I^*a^*q_x)$	1
1	1	0	-1	$[1/2, -\sqrt{3}/2]$	$\exp(I^*(a^*q_x/2 - \sqrt{3} * a * q_y/2))$	1
1	1	0	1	$[-1/2, \sqrt{3}/2]$	$\exp(I^*(-a^*q_x/2 + \sqrt{3} * a * q_y/2))$	1
1	1	1	0	[1, 0]	$\exp(I^*a^*q_x)$	1
1	1	1	1	$[1/2, \sqrt{3}/2]$	$\exp(I^*(a^*q_x/2 + \sqrt{3} * a * q_y/2))$	1
2	2	-1	-1	$[-1/2, -\sqrt{3}/2]$	$\exp(I^*(a^*q_x/2 - \sqrt{3} * a * q_y/2))$	1
2	2	-1	0	[-1, 0]	$\exp(-I^*a^*q_x)$	1
2	2	0	-1	$[1/2, -\sqrt{3}/2]$	$\exp(I^*(a^*q_x/2 - \sqrt{3} * a * q_y/2))$	1
2	2	0	1	$[-1/2, \sqrt{3}/2]$	$\exp(I^*(-a^*q_x/2 + \sqrt{3} * a * q_y/2))$	1
2	2	1	0	[1, 0]	$\exp(I^*a^*q_x)$	1
2	2	1	1	$[1/2, \sqrt{3}/2]$	$\exp(I^*(a^*q_x/2 + \sqrt{3} * a * q_y/2))$	1
1	2	-1	-1	[0, -1]	$\exp(I^*(-a^*q_x/2 - \sqrt{3} * a * q_y/2))$	$2*\sqrt{3}/3$
1	2	-1	1	$[-\sqrt{3}/2, 1/2]$	$\exp(I^*(-3^*a^*q_x/2 + \sqrt{3} * a * q_y/2))$	$2*\sqrt{3}/3$
1	2	1	1	$[\sqrt{3}/2, 1/2]$	$\exp(I^*(a^*q_x/2 + \sqrt{3} * a * q_y/2))$	$2*\sqrt{3}/3$
2	1	-1	-1	$[-\sqrt{3}/2, -1/2]$	$\exp(I^*(a^*q_x/2 - \sqrt{3} * a * q_y/2))$	$2*\sqrt{3}/3$
2	1	1	-1	$[\sqrt{3}/2, -1/2]$	$\exp(I^*(3^*a^*q_x/2 - \sqrt{3} * a * q_y/2))$	$2*\sqrt{3}/3$
2	1	1	1	[0, 1]	$\exp(I^*(a^*q_x/2 + \sqrt{3} * a * q_y/2))$	$2*\sqrt{3}/3$

3.2 Matriz de constantes de fuerza y matriz dinámica

A parte de identificar los primeros, segundos, ... vecinos, necesitamos conocer la matriz de constantes de fuerza que corresponde a la interacción de cada átomo de la celdilla unidad con su n-esimo vecino.

Vamos a suponer (por simplificar) que un desplazamiento longitudinal (radial, que estará contenido en el plano del cristal) o transversal (tangencial, sea en el plano o perpendicular al plano) solo genera una fuerza radial o transversal.

Dibujar un esquema tipo



3.2.1 Primeros vecinos

Para hacer más explícito el método que he seguido se muestran las matrices de constantes de fuerza para los primeros vecinos del boro (y que son átomos de nitrógeno).

Dibujar el boro, y sus vecinos

```
PrimerosVecinosBoro= Atomos(1,1)[(Atomos(1,1)['Distancia']<0.9) &\
(Atomos(1,1)['Distancia']>0) & (Atomos(1,1)[r"$\alpha\prime$"]==1)]
PrimerosVecinosBoro.to_latex(escape=False)
```

α'	α	l_1	l_2	$\hat{R}_{\nu',\nu,\vec{l}}$	Fase	Distancia	
10	1	2	-1	0	$[-\sqrt{3}/2, -1/2]$	$\exp(-I^*a^*q_x)$	$\sqrt{3}/3$
13	1	2	0	0	$[\sqrt{3}/2, -1/2]$	1	$\sqrt{3}/3$
14	1	2	0	1	$[0, 1]$	$\exp(I^*(-a^*q_x/2 + \sqrt{3} * a * q_y/2))$	$\sqrt{3}/3$

Podemos observar que para el átomo de nitrógeno de la celdilla $\vec{l} = (0, 1)$ la correspondiente matriz de constantes de fuerza puede escribirse (en coordenadas cartesianas) como:

```
phi_1r__BN,phi_1ti__BN,phi_1to__BN=symbols('phi_1r__BN,phi_1ti__BN,phi_1to__BN')
```

```
Phi_10__BN=Matrix([[phi_1ti__BN,0,0],[0,phi_1r__BN,0],[0,0,phi_1to__BN]])
print(r"\begin{equation}\Phi_1^{\sim\{BN\}}(0,1)=", latex(Phi_10__BN),r"\end{equation}")
```

$$\Phi_1^{BN}(0,1) = \begin{bmatrix} \phi_{1ti}^{BN} & 0 & 0 \\ 0 & \phi_{1r}^{BN} & 0 \\ 0 & 0 & \phi_{1to}^{BN} \end{bmatrix} \quad (7)$$

donde ϕ_r hace referencia a la constante de fuerza en la dirección radial (*bond stretching*) y ϕ_{ti} y ϕ_{to} a las constantes de fuerza en dirección tangencial (*bond bending*) dentro y fuera de plano, respectivamente.

Puesto que los otros dos primeros vecinos del boro son átomos exactamente iguales que este, y se encuentran a la misma distancia, podemos calcular sus respectivas matrices de fuerza simplemente rotando esta matriz:

$$\Phi(\vec{R}_{\vec{l}}) = U(\vec{R}_{\vec{l}})^{-1} \Phi_1^{BN}(0,1) U(\vec{R}_{\vec{l}}) \quad (8)$$

Donde $U(\vec{R}_{\vec{l}})$ es la matriz de rotación para llevar el átomo de nitrógeno que esta en la celdilla $\vec{l} = (0,1)$ a la posición que ocupan cada uno de los otros primeros vecinos en sus respectivas celdillas.

Por lo tanto tenemos que para el átomo situado en $\vec{l} = (-1,0)$

```
def U(theta):
    return Matrix([[cos(theta), sin(theta), 0], [-sin(theta), cos(theta), 0], [0, 0, 1]])

# Para el átomo que ocupa la posición l_1=-1,0
def Phi_11_BN(theta):
    return U(-theta)*Phi_10_BN*U(theta)
```

```
print_latex(Matrix([Phi_11_BN(2*pi/3)[i,j].factor() for j in range(3) \
for i in range(3)]).reshape(3,3))
```

$$\begin{bmatrix} \frac{3\phi_{1r}^{BN} + \phi_{1ti}^{BN}}{4} & \frac{\sqrt{3}(\phi_{1r}^{BN} - \phi_{1ti}^{BN})}{4} & 0 \\ \frac{\sqrt{3}(\phi_{1r}^{BN} - \phi_{1ti}^{BN})}{4} & \frac{\phi_{1r}^{BN} + 3\phi_{1ti}^{BN}}{4} & 0 \\ 0 & 0 & \phi_{1to}^{BN} \end{bmatrix} \quad (9)$$

Y para el átomo situado en $\vec{l} = (0,0)$ la matriz de constantes de fuerza es:

```
print_latex(Matrix([Phi_11_BN(-2*pi/3)[i,j].factor() for j in range(3) \
for i in range(3)]).reshape(3,3))
```

$$\begin{bmatrix} \frac{3\phi_{1r}^{BN} + \phi_{1ti}^{BN}}{4} & -\frac{\sqrt{3}(\phi_{1r}^{BN} - \phi_{1ti}^{BN})}{4} & 0 \\ -\frac{\sqrt{3}(\phi_{1r}^{BN} - \phi_{1ti}^{BN})}{4} & \frac{\phi_{1r}^{BN} + 3\phi_{1ti}^{BN}}{4} & 0 \\ 0 & 0 & \phi_{1to}^{BN} \end{bmatrix} \quad (10)$$

Notemos que las vibraciones fuera de plano, en la dirección z no están acopladas a las interplanares.

Por lo tanto, la contribución a matriz dinámica que tenemos por parte de la interacción de estos primeros vecinos es:

```
Phi_1_BN = Phi_10_BN*fase(0,1) + Phi_11_BN(2*pi/3)*fase(-1,0) + Phi_11_BN(-2*pi/3)*fase(0,0)
Phi_1_BN_xy = Matrix([Phi_1_BN[i,j].factor().simplify() for i in range(2) \
for j in range(2)]).reshape(2,2)
Phi_1_BN_zz = Phi_1_BN[2,2].factor().simplify()
```

$$\begin{bmatrix} \frac{\left(3\phi_{1r}^{BN} e^{iaqx} + 3\phi_{1r}^{BN} + \phi_{1ti}^{BN} e^{iaqx} + 4\phi_{1ti}^{BN} e^{\frac{ia(qx + \sqrt{3}qy)}{2}} + \phi_{1ti}^{BN}\right) e^{-iaqx}}{4} & \frac{\sqrt{3}(1 - e^{iaqx})(\phi_{1r}^{BN} - \phi_{1ti}^{BN}) e^{-iaqx}}{4} \\ \frac{\sqrt{3}(1 - e^{iaqx})(\phi_{1r}^{BN} - \phi_{1ti}^{BN}) e^{-iaqx}}{4} & \frac{\left(\phi_{1r}^{BN} e^{iaqx} + 4\phi_{1r}^{BN} e^{\frac{ia(qx + \sqrt{3}qy)}{2}} + \phi_{1r}^{BN} + 3\phi_{1ti}^{BN} e^{iaqx} + 3\phi_{1ti}^{BN}\right) e^{-iaqx}}{4} \end{bmatrix} \quad (11)$$

Mientras que la única componente no nula de la fila y columna 3 de esta matriz dinámica (la componente zz) es:

$$\phi_{1to}^{BN} \left(e^{iaqx} + e^{\frac{ia(qx + \sqrt{3}qy)}{2}} + 1 \right) e^{-iaqx} \quad (12)$$

De manera análoga, para los primeros vecinos del átomo de nitrógeno

```

PrimerosVecinosNitrogeno= Atomos(1,1)[(Atomos(1,1)['Distancia']<0.9) & \
(Atomos(1,1)['Distancia']>0) & (Atomos(1,1)[r"$\alpha\prime$"]==2)]
PrimerosVecinosNitrogeno.to_latex(escape=False)

```

	α'	α	l_1	l_2	$\hat{R}_{\nu',\nu,\vec{l}}$	Fase	Distancia
21	2	1	0	-1	$[0, -1]$	$\exp(I^*(a^*q_x/2 - \sqrt{3} * a * q_y/2))$	$\sqrt{3}/3$
22	2	1	0	0	$[-\sqrt{3}/2, 1/2]$	1	$\sqrt{3}/3$
25	2	1	1	0	$[\sqrt{3}/2, 1/2]$	$\exp(I^*a^*q_x)$	$\sqrt{3}/3$

Por simetría, la matriz de constantes de fuerza para la interacción entre el átomo de nitrógeno y su primer vecino situado en $\vec{l} = (0, -1)$ es igual a la que hemos visto antes para el átomo de boro en la celdilla $\vec{l} = (0, 0)$ y su primer vecino (un átomo de nitrógeno) en la celdilla $\vec{l} = (0, 1)$:

Y por lo tanto, la contribución a matriz dinámica que tenemos por parte de la interacción de estos primeros vecinos es:

```

Phi_10__NB=Phi_10__BN
def Phi_11__NB(theta):
    return U(-theta)*Phi_10__NB*U(theta)

Phi_1__NB=Phi_10__NB*fase(0,-1)+Phi_11__NB(2*pi/3)*fase(1,0)+Phi_11__NB(-2*pi/3)*fase(0,0)
Phi_1__NB_xy=(Matrix([Phi_1__NB[i,j].factor() for i in range(2) \
for j in range(2)]).reshape(2,2))
Phi_1__NB_zz=Phi_1__NB[2,2].factor()

```

La parte que da la interacción dentro del plano:

$$\begin{bmatrix} \frac{3\phi_{1r}^{BN}e^{iaq_x} + 3\phi_{1r}^{BN} + 4\phi_{1ti}^{BN}e^{\frac{iaq_x}{2}}e^{-\frac{\sqrt{3}iaq_y}{2}} + \phi_{1ti}^{BN}e^{iaq_x} + \phi_{1ti}^{BN}}{4} & \frac{\sqrt{3}(\phi_{1r}^{BN} - \phi_{1ti}^{BN})(e^{iaq_x} - 1)}{4} \\ \frac{\sqrt{3}(\phi_{1r}^{BN} - \phi_{1ti}^{BN})(e^{iaq_x} - 1)}{4} & \frac{4\phi_{1r}^{BN}e^{\frac{iaq_x}{2}}e^{-\frac{\sqrt{3}iaq_y}{2}} + \phi_{1r}^{BN}e^{iaq_x} + \phi_{1r}^{BN} + 3\phi_{1ti}^{BN}e^{iaq_x} + 3\phi_{1ti}^{BN}}{4} \end{bmatrix} \quad (13)$$

Mientras que la interacción fuera de plano viene dada por el elemento de matriz:

$$\phi_{1to}^{BN} \left(e^{\frac{iaq_x}{2}} e^{-\frac{\sqrt{3}iaq_y}{2}} + e^{iaq_x} + 1 \right) \quad (14)$$

Debemos hacer notar que *como la energía potencial es una función cuadrática de los desplazamientos atómicos $u_i^B(\vec{R}_i)$ $u_i^N(\vec{R}_i)$ la matriz de constantes de fuerza tiene la forma $\phi_{ij}^{BN}(\vec{R}_i) = \phi_{ji}^{NB}(-\vec{R}_i)$, y su transformada de Fourier, es decir, la matriz dinámica, es una matriz Hermítica (ibid.),*

3.2.2 Segundos vecinos y terceros vecinos

Pro ahora las contribuciones a la matriz dinámica de los segundos y terceros vecinos.

```
SegundosVecinosBoro= Atomos(1,1)[(Atomos(1,1)['Distancia']<1.1) &\
(Atomos(1,1)['Distancia']>0.9) & (Atomos(1,1)[r"$\alpha\prime$"]==1)]
##SegundosVecinosBoro.to_latex(escape=False)
```

Tenemos 6 segundos vecinos para cada uno de los dos átomos de la celdilla unidad y en este caso las interacciones son entre el mismo tipo de átomo. Fijandonos en el átomo de boro situado en la celdilla $\vec{l} = (1,0)$ podemos escribir la matriz de constantes de fuerza como:

$$\Phi_2^{BB}(1,0) = \begin{pmatrix} \phi_{2r}^{(BB)} & 0 & 0 \\ 0 & \phi_{2ti}^{(BB)} & 0 \\ 0 & 0 & \phi_{2to}^{(BB)} \end{pmatrix} \quad (15)$$

mientras que para el átomo de nitrógeno la matriz de constantes de fuerza para el átomo situado en la celdilla $\vec{l} = (1,0)$ es:

$$\Phi_2^{NN}(1,0) = \begin{pmatrix} \phi_{2r}^{(NN)} & 0 & 0 \\ 0 & \phi_{2ti}^{(NN)} & 0 \\ 0 & 0 & \phi_{2to}^{(NN)} \end{pmatrix} \quad (16)$$

y por tanto, la contribución a la matriz dinámica debida a estas interacciones será:

```
phi_2r__BB,phi_2ti__BB,phi_2to__BB=symbols('phi_2r__BB,phi_2ti__BB,phi_2to__BB')
# Para el átomo que ocupa la posición l=1,0
Phi_20__BB=Matrix([[phi_2ti__BB,0,0],[0,phi_2r__BB,0],[0,0,phi_2to__BB]])
def Phi_2l__BB(theta):
    return U(-theta)*Phi_20__BB*U(theta)
```

```
Phi_2__BB=Phi_20__BB*fase(1,0)+Phi_2l__BB(pi/3)*fase(1,1)+Phi_2l__BB(-pi/3)*fase(0,-1) \
+Phi_2l__BB(pi)*fase(-1,0)+Phi_2l__BB(2*pi/3)*fase(0,1)+Phi_2l__BB(-2*pi/3)*fase(-1,-1)
#Phi_2__BB=Matrix([Phi_2__BB[i,j].rewrite(cos).simplify() for i in range(3)] \
# for j in range(3))).reshape(3,3)
```

```
SegundosVecinosNitrogeno= Atomos(1,1)[(Atomos(1,1)['Distancia']<1.1) &\
(Atomos(1,1)['Distancia']>0.9) & (Atomos(1,1)[r"$\alpha\prime$"]==2)]
##SegundosVecinosNitrogeno.to_latex(escape=False)
phi_2r__NN,phi_2ti__NN,phi_2to__NN=symbols('phi_2r__NN,phi_2ti__NN,phi_2to__NN')
Phi_20__NN=Matrix([[phi_2ti__NN,0,0],[0,phi_2r__NN,0],[0,0,phi_2to__NN]])
# Para el átomo que ocupa la posición l=1,0
def Phi_2l__NN(theta):
    return U(-theta)*Phi_20__NN*U(theta)
```

```
Phi_2__NN=Phi_20__NN*fase(1,0)+Phi_2l__NN(pi/3)*fase(1,1)+Phi_2l__NN(-pi/3)*fase(0,-1) \
+Phi_2l__NN(pi)*fase(-1,0)+Phi_2l__NN(2*pi/3)*fase(0,1)+Phi_2l__NN(-2*pi/3)*fase(-1,-1)
#Phi_2__NN=Matrix([Phi_2__NN[i,j].rewrite(cos).simplify() for i in range(3)] \
# for j in range(3))).reshape(3,3)
```

Mientras que para los terceros vecinos:

```
TercerosVecinosBoro= Atomos(1,1)[(Atomos(1,1)['Distancia']<sqrt(21)/3) &\
(Atomos(1,1)['Distancia']>1) & (Atomos(1,1)[r"$\alpha\prime$"]==1)]
```

```
TercerosVecinosNitrogeno= Atomos(1,1)[(Atomos(1,1)['Distancia']<sqrt(21)/3) &\
(Atomos(1,1)['Distancia']>1) & (Atomos(1,1)[r"$\alpha\prime$"]==2)]
```

```
phi_3r__BN,phi_3ti__BN,phi_3to__BN=symbols('phi_3r__BN,phi_3ti__BN,phi_3to__BN')
phi_3r__NB,phi_3ti__NB,phi_3to__NB=symbols('phi_3r__BN,phi_3ti__BN,phi_3to__BN')
```

```
# Para el átomo de N de la celdilla l=-1,-1
```

```
Phi_30__BN=Matrix([[phi_3ti__BN,0,0],[0,phi_3r__BN,0],[0,0,phi_3to__BN]])
```

```
# Para el átomo de B de la celdilla l=1,1
```

```
Phi_30__NB=Matrix([[phi_3ti__NB,0,0],[0,phi_3r__NB,0],[0,0,phi_3to__NB]])
```

```
def Phi_3l__BN(theta):
```

```
    return U(-theta)*Phi_30__BN*U(theta)
```

```
def Phi_3l__NB(theta):
```

```
    return U(-theta)*Phi_30__NB*U(theta)
```

```
Phi_3__BN=Phi_30__BN*fase(-1,-1)+Phi_3l__BN(2*pi/3)*fase(1,1)+Phi_3l__BN(-2*pi/3)*fase(-1,1)
```

```
#Phi_3__BN=Matrix([Phi_3__BN[i,j].rewrite(cos).simplify() for i in range(3) \
```

```
#for j in range(3)]).reshape(3,3)
```

```
Phi_3__NB=Phi_30__NB*fase(1,1)+Phi_3l__NB(2*pi/3)*fase(-1,-1)+Phi_3l__NB(-2*pi/3)*fase(1,-1)
```

```
#print_latex(Phi_3__NB)
```

```
#Phi_3__NB=Matrix([Phi_3__NB[i,j].rewrite(cos).simplify() for i in range(3) for j in range(3)
```

Debemos tener en cuenta en este punto que las constantes de fuerza de fuerza asociadas a la interaccion de un átomo **??consigo mismo o con la totalidad del cristal??** las excluimos con la ayuda de las condiciones impuestas por la invarianza respecto a la traslacion del cristal en su totalidad en las direcciones x/z (ibid.).

3.2.3 Matriz dinámica del cristal de BN

Tenemos por lo tanto que la matriz dinámica que obtenemos considerando hasta los terceros vecinos es

```
Dsup=Phi_2__BB.col_insert(3,Phi_1__BN+Phi_3__BN);
```

```
Dinf=(Phi_1__NB+Phi_3__NB).col_insert(3,Phi_2__NN)
```

```
D=Dsup.row_insert(3,Dinf)
```

```
def Matriu_com_a_Taula(Matriu,m,n):
```

```
    s = r"\begin{tabular}{|c|c|c|}\hline"
```

```
    s += r"$i$ & $j$ & $D_{i,j}$ \\\hline"
```

```
    for i in range(m):
```

```
        for j in range(n):
```

```
            s += " $ %d $ & $ %d $ & $ %s $" %(i+1, j+1, latex(Matriu[i,j]))
```

```
            s += r"\\ \hline"
```

```
    s += r"\end{tabular}"
```

```
    return s
```

```
Matriu_com_a_Taula(D,6,6)
```

[illegible]

3.2.4 Ajuste a los datos experimentales

Podemos evaluar la matriz dinámica en el punto Γ ($q_x = 0, q_y = 0$).

```
from periodictable import B, N, constants
u=constants.atomic_mass_constant
D_Gamma=D.subs([(q_x,0),(q_y,0),(M_B,B.mass*u),(M_N,N.mass*u)])
latex(Matrix([D_Gamma[i,j].factor() for i in range(6) for j in range(6)]).reshape(6,6))
```

$$\begin{bmatrix} 3(\phi_{2r}^{BB} + \phi_{2ti}^{BB}) & 0 & 0 & \frac{3(\phi_{1r}^{BN} + \phi_{1ti}^{BN} + \phi_{3r}^{BN} + \phi_{3ti}^{BN})}{2} & 0 & 0 \\ 0 & 3(\phi_{2r}^{BB} + \phi_{2ti}^{BB}) & 0 & 0 & \frac{3(\phi_{1r}^{BN} + \phi_{1ti}^{BN} + \phi_{3r}^{BN} + \phi_{3ti}^{BN})}{2} & 0 \\ 0 & 0 & 6\phi_{2to}^{BB} & 0 & 0 & 3(\phi_{1to}^{BN} + \phi_{3to}^{BN}) \\ \frac{3(\phi_{1r}^{BN} + \phi_{1ti}^{BN} + \phi_{3r}^{BN} + \phi_{3ti}^{BN})}{2} & 0 & 0 & 3(\phi_{2r}^{NN} + \phi_{2ti}^{NN}) & 0 & 0 \\ 0 & \frac{3(\phi_{1r}^{BN} + \phi_{1ti}^{BN} + \phi_{3r}^{BN} + \phi_{3ti}^{BN})}{2} & 0 & 0 & 3(\phi_{2r}^{NN} + \phi_{2ti}^{NN}) & 0 \\ 0 & 0 & 3(\phi_{1to}^{BN} + \phi_{3to}^{BN}) & 0 & 0 & 6\phi_{2to}^{NN} \end{bmatrix} \quad (17)$$

References

- Aizawa, T. et al. (1990). “Bond Softening in Monolayer Graphite Formed on Transition-Metal Carbide Surfaces”. In: *Physical Review B* 42.18, pp. 11469–11478. DOI: 10.1103/physrevb.42.11469. URL: <https://doi.org/10.1103/physrevb.42.11469>.
- Brüesch, Peter (1982). *Phonons: Theory and Experiments I*. Springer Series in Solid-State Sciences. Springer Berlin Heidelberg, nil. DOI: 10.1007/978-3-642-81781-6. URL: <https://doi.org/10.1007/978-3-642-81781-6>.
- Falkovsky, L.A. (2008). “Symmetry Constraints on Phonon Dispersion in Graphene”. In: *Physics Letters A* 372.31, pp. 5189–5192. DOI: 10.1016/j.physleta.2008.05.085. URL: <https://doi.org/10.1016/j.physleta.2008.05.085>.