

# Berechnung der Phononen Dispersionsrelation am Beispiel eines FCC Gitters

## Inhaltsverzeichnis

1) FCC-Gitter .....	2
2) Federmodell .....	3
3) Dispersionsrelation.....	4
4) Auswahl der $k$ Vektoren.....	6
5) Programm im Matlab .....	7
i) Hauptprogramm.....	7
ii) Solver für Dispersions-Relation .....	8
iii) Ergebnis.....	9

## 1) FCC-Gitter

Im Folgenden soll gezeigt werden, wie aus einer bekannten Gitterkonfiguration eine Dispersionsrelation berechnet und dargestellt werden kann. Da, wie jeder weiß, durch ein Beispiel alles klarer wird, berechnen wir die Dispersionsrelation eines FCC Gitters.

Um irgendetwas berechnen zu können, müssen wir uns im Klaren sein, wie unser Gitter aussieht.

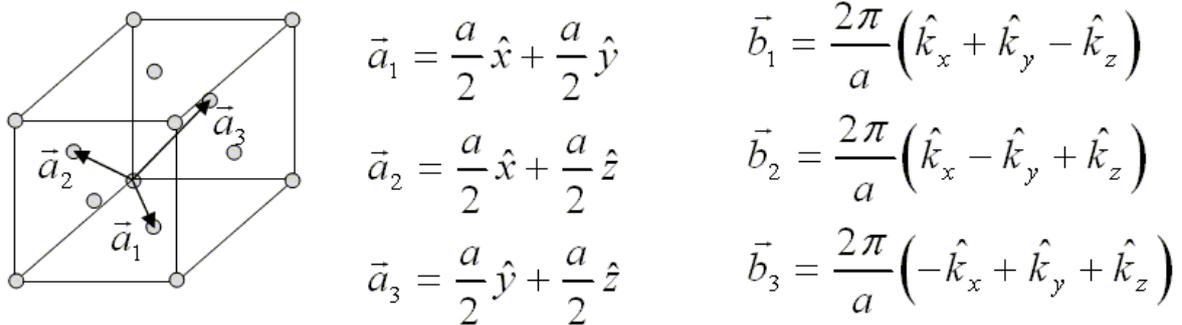


Abb. 1: FCC Conventional Unit Cell mit Basisvektoren und Vektoren im reziproken Raum.

In Abb. 1 sieht man die Conventional Unit Cell des FCC-Gitters. Dummerweise ist in dieser Darstellung nicht sofort ersichtlich wieviele direkte Nachbarn ein Gitterpunkt hat. Daher ist eine andere Visualisierung des Gitters sinnvoller.

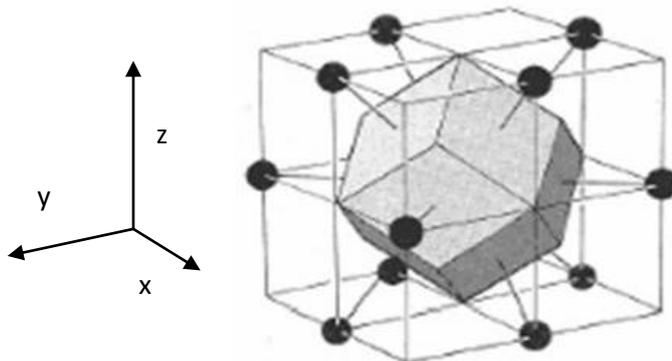


Abb. 2: FCC-Wigner-Seitz-Cell

In Abb.2 ist die Wigner-Seitz-Cell des FCC-Gitters eingezeichnet. Im Zentrum befindet sich (hier nicht sichtbar) ein Gitterpunkt. Rundherum sind alle direkten Nachbarn des Gitterpunkts eingezeichnet. Es sind 12.

Die Koordinaten lauten:

$l,m,n$	$x,y,z$
1,0,0	$a/2,a/2,0$
0,1,0	$a/2,0,a/2$
0,0,1	$0,a/2,a/2$
-1,0,0	$-a/2,-a/2,0$
0,-1,0	$-a/2,0,-a/2$
0,0,-1	$0,-a/2,-a/2$
0,1,-1	$a/2,-a/2,0$
1,0,-1	$a/2,0,-a/2$
1,-1,0	$0,a/2,-a/2$
0,-1,1	$-a/2,a/2,0$
-1,0,1	$-a/2,0,a/2$
-1,1,0	$0,-a/2,a/2$

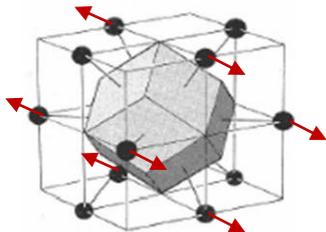
## 2) Federmodell

Wie üblich in physikalischen Aufgabestellungen ist es nun an der Zeit komplexe Sachverhalte, wie komplizierte Atombindungen zu vergessen und die Dinge zu vereinfachen. Wir nehmen an, dass alle Massen gleich groß sind und diese mit Federn gleicher Federkonstante untereinander verbunden sind. Ein Atom ist also mit 12 Federn an seine Nachbarn gebunden. Jedes dieser Nachbaratome kann in x,y,z-Richtung ausgelenkt werden. Das Auslenken der Nachbaratome bewirkt eine Kraft auf das zentrale Atom und diese gilt es zu berechnen.

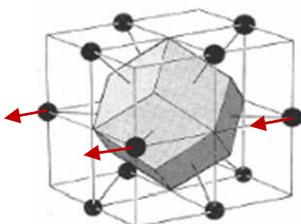
Es macht Sinn zuerst nur die x-Komponente der Kraft zu betrachten.

Folgende Terme tragen merklich zur Kraft bei:

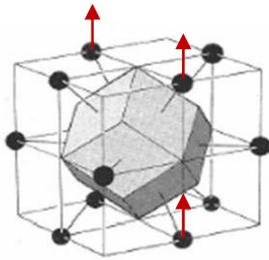
8 Terme von der x-Auslenkung der 8 Atome, die nicht in der gleichen y-z-Ebene liegen.



4 Terme von der y-Auslenkung der 4 Atome, die in der gleichen x-y Ebene liegen.

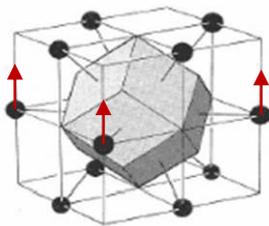


4 Terme von der z-Auslenkung der 4 Atome, die in der gleichen x-z Ebene liegen.



Alle Pfeile können natürlich auch in die entgegengesetzte Richtung zeigen.

Nun taucht die berechnete Frage auf, warum all die anderen möglichen Atombewegungen nicht berücksichtigt werden. Der Grund dafür liegt darin, dass alle betrachteten Auslenkungen sehr klein sind, sodass viele Terme einfach vernachlässigt werden können. Werden beispielsweise die Atome in der gleichen x-y Ebene in z-Richtung ausgelenkt, so steht der Vektor der Auslenkung normal auf die Feder und aufgrund der sehr kleinen Auslenkung ist die Kraft in x-Richtung nahezu Null.



Wer trotzdem skeptisch ist kann sich die geometrischen Verhältnisse aufzeichnen und wird erkennen, dass alle oben nicht genannten Terme sehr viel weniger beitragen als die genannten und somit ignoriert werden können.

Für die y- und z-Komponente der Kraft finden sich aufgrund der Symmetrie ähnliche Ergebnisse.

### 3) Dispersionsrelation

Jetzt, wo wir die wirkenden Kräfte auf ein Atom kennen, können wir das Problem mit Newton'scher Mechanik formulieren. Die Auslenkung des Atoms  $l_{mn}$  nennen wir  $u_{l_{mn}}$ . Dann folgt:

$$m \frac{d^2 u_{l_{mn}}^x}{dt^2} = F_{l_{mn}}^x$$

$$m \frac{d^2 u_{l_{mn}}^y}{dt^2} = F_{l_{mn}}^y$$

$$m \frac{d^2 u_{l_{mn}}^z}{dt^2} = F_{l_{mn}}^z$$

Die x-Komponente der Kraft (16 Terme laut oben) ist dann:

$$\begin{aligned}
 F_{\Delta mn}^x = \frac{C}{\sqrt{2}} & \left[ \left( \mathcal{U}_{i+1mn}^x - \mathcal{U}_{\Delta mn}^x \right) + \left( \mathcal{U}_{i-1mn}^x - \mathcal{U}_{\Delta mn}^x \right) + \left( \mathcal{U}_{\Delta m+1n}^x - \mathcal{U}_{\Delta mn}^x \right) + \left( \mathcal{U}_{\Delta m-1n}^x - \mathcal{U}_{\Delta mn}^x \right) \right. \\
 & + \left( \mathcal{U}_{i+1mn-1}^x - \mathcal{U}_{\Delta mn}^x \right) + \left( \mathcal{U}_{i-1mn+1}^x - \mathcal{U}_{\Delta mn}^x \right) + \left( \mathcal{U}_{\Delta m+1n-1}^x - \mathcal{U}_{\Delta mn}^x \right) + \left( \mathcal{U}_{\Delta m-1n+1}^x - \mathcal{U}_{\Delta mn}^x \right) \\
 & + \left( \mathcal{U}_{i+1mn}^y - \mathcal{U}_{\Delta mn}^y \right) + \left( \mathcal{U}_{i-1mn}^y - \mathcal{U}_{\Delta mn}^y \right) - \left( \mathcal{U}_{\Delta m+1n-1}^y - \mathcal{U}_{\Delta mn}^y \right) - \left( \mathcal{U}_{\Delta m-1n+1}^y - \mathcal{U}_{\Delta mn}^y \right) \\
 & \left. + \left( \mathcal{U}_{\Delta m+1n}^z - \mathcal{U}_{\Delta mn}^z \right) + \left( \mathcal{U}_{\Delta m-1n}^z - \mathcal{U}_{\Delta mn}^z \right) - \left( \mathcal{U}_{i+1mn-1}^z - \mathcal{U}_{\Delta mn}^z \right) - \left( \mathcal{U}_{i-1mn+1}^z - \mathcal{U}_{\Delta mn}^z \right) \right]
 \end{aligned}$$

Der Faktor  $\frac{1}{\sqrt{2}} = \cos(45^\circ)$  kommt daher, dass die Gitterpunkte im  $45^\circ$  Winkel zueinander versetzt stehen und nur die auf die x-Achse projizierte Federkraft wirkt. Als Lösung des Gleichungssystems nehmen wir harmonische Schwingungen an. Für die x-Komponente der Auslenkung erhält man dann:

$$\mathbf{u}_{lmn}^x = \mathbf{u}_{\vec{k}}^x * e^{i(\bar{a}_1 * \vec{k} + m\bar{a}_2 * \vec{k} + n\bar{a}_3 * \vec{k} - \omega t)} = \mathbf{u}_{\vec{k}}^x * e^{i\left(\frac{(l+m)k_x}{2} + \frac{(l+n)k_y}{2} + \frac{(m+n)k_z}{2} - \omega t\right)}$$

Wobei  $\vec{a}_i$  die Basisvektoren des Gitters sind und  $\vec{k}$  der betrachtete Vektor im reziproken Raum.

Diese Lösungen setzt man nun in die Bewegungsgleichungen ein und erhält daraus folgende Matrix:

$$M(k_x, k_y, k_z) = \begin{bmatrix}
 4 - \cos\left[\frac{(k_x + k_y) a_0}{2}\right] - \cos\left[\frac{(k_x - k_y) a_0}{2}\right] - \cos\left[\frac{(k_x + k_z) a_0}{2}\right] - \cos\left[\frac{(k_x - k_z) a_0}{2}\right] & -\cos\left[\frac{(k_x + k_y) a_0}{2}\right] + \cos\left[\frac{(k_x - k_y) a_0}{2}\right] & -\cos\left[\frac{(k_x + k_z) a_0}{2}\right] + \cos\left[\frac{(k_x - k_z) a_0}{2}\right] \\
 -\cos\left[\frac{(k_x + k_y) a_0}{2}\right] + \cos\left[\frac{(k_x - k_y) a_0}{2}\right] & 4 - \cos\left[\frac{(k_x + k_y) a_0}{2}\right] - \cos\left[\frac{(k_x - k_y) a_0}{2}\right] - \cos\left[\frac{(k_y + k_z) a_0}{2}\right] - \cos\left[\frac{(k_y - k_z) a_0}{2}\right] & -\cos\left[\frac{(k_y + k_z) a_0}{2}\right] + \cos\left[\frac{(k_y - k_z) a_0}{2}\right] \\
 -\cos\left[\frac{(k_x + k_z) a_0}{2}\right] + \cos\left[\frac{(k_x - k_z) a_0}{2}\right] & -\cos\left[\frac{(k_y + k_z) a_0}{2}\right] + \cos\left[\frac{(k_y - k_z) a_0}{2}\right] & 4 - \cos\left[\frac{(k_x + k_z) a_0}{2}\right] - \cos\left[\frac{(k_x - k_z) a_0}{2}\right] - \cos\left[\frac{(k_y + k_z) a_0}{2}\right] - \cos\left[\frac{(k_y - k_z) a_0}{2}\right]
 \end{bmatrix}$$

Wobei gilt:

$$-\omega^2 m \mathbf{u}_{lmn} = C * M * \mathbf{u}_{lmn}$$

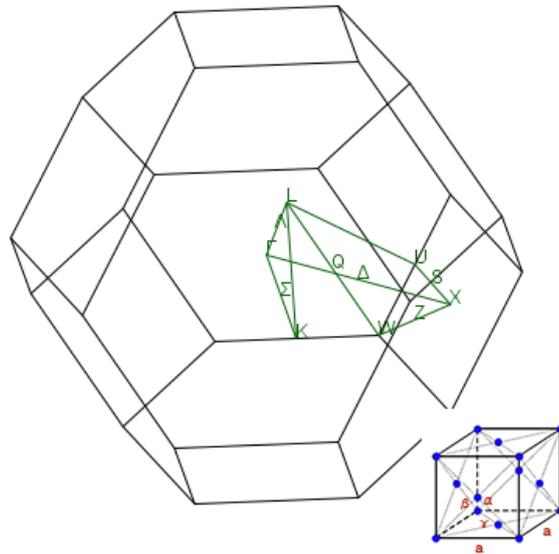
Die Cosinus-Terme erhält man aus der Relation:  $\cos(x) = \frac{e^{ix} + e^{-ix}}{2}$

Um diese Gleichung zu lösen, müssen wir die Eigenwerte  $\lambda$  der Matrix für betrachtete k-Vektoren bestimmen und erhalten dann eine Dispersionsrelation:

$$\omega = \sqrt{-\frac{\lambda * C}{m}}$$

#### 4) Auswahl der $\vec{k}$ Vektoren

Als nächstes müssen wir wählen welche  $\vec{k}$  Vektoren wir in unserer Berechnung verwenden wollen. Hierzu betrachten wir die erste Brillouin-Zone des FCC-Gitters:



Wollen wir nun beispielsweise die Dispersionsrelation entlang der Symmetrielinie  $\Delta$  betrachten, so brauchen wir einen Vektor  $\vec{k}$  in Richtung  $\overline{\Gamma X}$ , dessen Länge wir von 0 bis zur Brillouin-Zonengrenze variieren. Nun taucht die Frage auf, wie weit die Brillouin-Zonengrenze von einem Punkt entfernt ist. Dies ist ein rein geometrisches Problem und sollte durch skizzieren und rechnen gelöst werden können.

Für FCC ergeben sich folgende Beziehungen:

$$\vec{k} = u\vec{b}_1 + v\vec{b}_2 + w\vec{b}_3 \quad : \quad (u, v, w)$$

Symmetry points $(u, v, w)$	$[k_x, k_y, k_z]$	Point group
$\Gamma: (0, 0, 0)$	$[0, 0, 0]$	m3m
$X: (0, 1/2, 1/2)$	$[0, 2\pi/a, 0]$	4/mmm
$L: (1/2, 1/2, 1/2)$	$[\pi/a, \pi/a, \pi/a]$	$\bar{3}m$
$W: (1/4, 3/4, 1/2)$	$[\pi/a, 2\pi/a, 0]$	$\bar{4}2m$
$U: (1/4, 5/8, 5/8)$	$[\pi/2a, 2\pi/a, \pi/2a]$	mm2
$K: (3/8, 3/4, 3/8)$	$[3\pi/2a, 3\pi/2a, 0]$	mm2

$$\overline{\Gamma L} = \frac{\sqrt{3}\pi}{a}, \quad \overline{\Gamma X} = \frac{2\pi}{a}, \quad \overline{\Gamma K} = \frac{3\pi}{\sqrt{2}a}$$

Symmetry lines	Point group
$\Delta: (0, v, v) \quad 0 < v < 1/2$	4mm
$\Lambda: (w, w, w) \quad 0 < w < 1/2$	3m
$\Sigma: (u, 2u, u) \quad 0 < u < 3/8$	mm2
$S: (2u, 1/2+2u, 1/2+u) \quad 0 < u < 1/8$	mm2
$Z: (u, 1/2+u, 1/2) \quad 0 < u < 1/4$	mm2
$Q: (1/2-u, 1/2+u, 1/2) \quad 0 < u < 1/4$	2

Eine drehbare erste Brillouin-Zone des FCC Gitters mit allen Symmetriepunkten findet sich auch unter <http://lamp.tu-graz.ac.at/~hadley/ss1/bzones/fcc.php>.

Im Weiteren wollen wir den Vektor  $\vec{k}$  entlang  $\overline{\Gamma X}$ ,  $\overline{XW}$ ,  $\overline{WK}$ ,  $\overline{K\Gamma}$  und  $\overline{\Gamma L}$  variieren. Die Spitze des Vektors  $\vec{k}$  läuft dabei also entlang der Symmetrielinien.

## 5) Programm im Matlab

Wir sind nun an einem Punkt angelangt, an dem wir nur noch Rechenarbeit vor uns haben. Um alle Rechnungen schnell abzuwickeln und die Dispersionsrelation auch graphisch auszuwerten zu können empfiehlt sich das Erstellen eines Computerprogramms. Dies kann natürlich jeder selbst programmieren oder, und das werden die meisten vorziehen, man verwendet ein Programm, das bereits existiert.

Im Rahmen der Vorlesung „Solid State Physics Fundamentals“ wurde als „Contribution to the course“ ein solches Programm von Philipp Thaler erstellt. Da es nicht einfach zu verstehen ist, werde ich versuchen es im Folgenden durch zahlreiche Kommentare im Programm verständlich machen. Ziel ist es aber auch, dass der Leser dieses Dokuments selbstständig in der Lage ist das Programm so umzuschreiben, dass die Dispersionsrelation eines anderen Gitters (z.B. BCC) ermittelt werden kann. Die notwendigen Änderungen im Programm sind nur minimal.

### i) Hauptprogramm

```
% Parameter

a = 4.49;          % Länge der konventionellen Einheitszelle
C = 200;          % Federkonstante in N/m
m = 0.344*10^-24; % Masse in kg
l=pi/a;
% Position des betrachteten Atoms und seiner Nachbarn
% FCC:
an = a/2 * [0,0,0; 1,1,0; 1,0,1; 0,1,1; -1,-1,0; -1,0,-1; 0,-1,-1; 1,-1,0; 1,0,-1; 0,1,-1; -1,1,0; -1,0,1; 0,-1,1];
% BCC:
% an = a/2 * [0,0,0; 1,1,1; 1,-1,1; 1,-1,-1; 1,1,-1; -1,1,1; -1,-1,1; -1,-1,-1; -1,1,-1];
% SC:
% an = a/2 * [0,0,0; 1,0,0; -1,0,0; 0,1,0; 0,-1,0; 0,0,1; 0,0,-1];

% Erzeugung der Wellenvektoren
nk = 1000; %Schritte bis zur brillioun zongrenze. Je größer nk ist, desto glatter
           %wird die Kurve.

%Die folgenden Vektoren sind die betrachteten k Vektoren. Die Spitze der Vektoren laufen von
einem Symmetriepunkt zum nächsten

k_GAMMA_X=[linspace(0,0,nk);linspace(0,2*pi/a,nk);linspace(0,0,nk)];
k_X_W=[linspace(0,pi/a,nk);linspace(2*pi/a,2*pi/a,nk);linspace(0,0,nk)];
k_W_K=[linspace(pi/a,3*pi/(2*a),nk);linspace(2*pi/a,3*pi/(2*a),nk);linspace(0,0,nk)];
k_K_GAMMA=[linspace(3*pi/(2*a),0,nk);linspace(3*pi/(2*a),0,nk);linspace(0,0,nk)];
k_GAMMA_L=[linspace(0,pi/a,nk);linspace(0,pi/a,nk);linspace(0,pi/a,nk)];

%Schließlich werden die k Vektoren verkettet um nacheinander abgearbeitet
%werden zu können.
k=[k_GAMMA_X k_X_W k_W_K k_K_GAMMA k_GAMMA_L];

%Der Vektor kk wird für die X-Achse des Plots verwendet und setzt sich aus
%den Weglängen zwischen den Symmetriepunkten zusammen.
kk=[linspace(0,2*pi/a,nk), 2*pi/a+linspace(0,pi/a,nk),
2*pi/a+pi/a+linspace(0,pi/(sqrt(2)*a),nk),
```

```

2*pi/a+pi/a+pi/(sqrt(2)*a)+linspace(0,3*pi/(sqrt(2)*a),nk),
2*pi/a+pi/a+pi/(sqrt(2)*a)+3*pi/(sqrt(2)*a)+linspace(0,sqrt(3)*pi/a,nk)];

%Berechnung der w-Werte. Es kann bis zu 3 Lösungen für w pro k-Vektor
%geben.
w = NaN(3,size(k,2));
for i = 1:size(k,2)
    k_aktuell = [k(1,i) k(2,i) k(3,i)]; %Erstellung des betrachteten k-Vektors
    w(:,i) = disp_solver(an, m, C, k_aktuell)/sqrt(C/m); %Der Faktor 1/sqrt(C/m)
    %wurde nur hinzugefügt, damit für w kleine Werte entstehen. Man erhält
    %w also in Einheiten von sqrt(C/m). Ohne den Faktor erhält man Werte im
    %Terahertzbereich.
end

figure(1)
plot(kk, w(1,:), 'b-', kk, w(2,:), 'b-', kk, w(3,:), 'b-')
hold on
%Anschließend werden noch die Punkte Gamma, X, W, K, Gamma,L durch rote Linien
%eingezeichnet.
ymax = round(10*1.2*max(w(:)))/10;
plot([0,0],[0,ymax], 'r-')
plot([2*pi/a,2*pi/a],[0,ymax], 'r-')
plot([2*pi/a+pi/a,2*pi/a+pi/a],[0,ymax], 'r-')
plot([2*pi/a+pi/a+pi/(sqrt(2)*a),2*pi/a+pi/a+pi/(sqrt(2)*a)],[0,ymax], 'r-')

plot([2*pi/a+pi/a+pi/(sqrt(2)*a)+3*pi/(sqrt(2)*a),2*pi/a+pi/a+pi/(sqrt(2)*a)+3*pi/(sqrt(2)*a) ]
,[0,ymax], 'r-')

xlim([kk(1) kk(end)])
ylim([0 ymax])

hold off

```

## ii) Solver für Dispersions-Relation

```

% solver für das Eigenwertproblem von 3d Dispersions-Relationen
function w = disp_solver(an, m, C, k)
% Richtungen der Nachbarn
ar = an(2:end,:);
arnorm = zeros(size(ar,1),1);
for n = 1:size(ar,1)
    arnorm(n) = norm(ar(n,[1,2,3]));
    ar(n,[1,2,3]) = ar(n,[1,2,3])./arnorm(n);
end
%Berechnung der Matrix laut Herleitung
for n = 1:size(ar,1)
    % x-direction, sum(delta_u):
    delta_uxx = (ar(:,1).*ar(:,1)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
    delta_uxy = (ar(:,1).*ar(:,2)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
    delta_uxz = (ar(:,1).*ar(:,3)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
    % y-direction, sum(delta_u):
    delta_uyx = (ar(:,2).*ar(:,1)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
    delta_uyy = (ar(:,2).*ar(:,2)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
    delta_uyz = (ar(:,2).*ar(:,3)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
    % z-direction, sum(delta_u):
    delta_uzx = (ar(:,3).*ar(:,1)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
    delta_uzy = (ar(:,3).*ar(:,2)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
    delta_uzz = (ar(:,3).*ar(:,3)).*(exp(i*sum(an(2:end,:).*repmat(k,size(ar,1),1),2))-1);
end
M = -
real([sum(delta_uxx,1),sum(delta_uxy,1),sum(delta_uxz,1);sum(delta_uyx,1),sum(delta_uyy,1),sum
(delta_uyz,1);sum(delta_uzx,1),sum(delta_uzy,1),sum(delta_uzz,1)]);
% eigs(M) = w^2 * m/C
%Berechnung von w laut hergeleiteter Formel
w = real(sqrt(eigs(M)*C/m));

```

### iii) Ergebnis

Führt man obiges Programm aus sollte die Dispersionsrelation geplottet werden.

