Band structure of optical lattices

J.R.

November 20, 2016

1 Physical considerations

The optical lattice potential arises from the dipole force, which itself emerges from the light shift or AC stark effect. This affects a variation in the (metastable) ground state of the confined atoms proportional to the intensity of the light field, and the atoms will be subject to a force such that this internal potential energy is minimized. Optical lattices are produced by retroreflected laser beams, forming a standing wave. For a red-detuned trap, the light-shifted potential is minimized at the nodes of the standing wave. Furthermore, a real optical lattice formed by Gaussian beams will see a spatial variation in light intensity, adding a slow quadratic term to the potential. We will ignore this, however, and the initial treatment will follow the spirit of first-order methods in solid state physics.

2 Bloch's theorem

Bloch's theorem is interesting in a historical sense; the periodic potential is an extremely versatile model and was one of the earliest exactly solved models. It sits in the suite of highly symmetric potentials that form the basis of much analysis in theoretical physics. The theorem was born of the consideration of eigenfunctions of the interaction-free electron Hamiltonian in crystals. We concern ourselves not with electrons in crystals but with Helium atoms in optical lattices, but much of the structure is transferrable. For a first approximation we will ignore atom-atom interactions, and treat the optical lattice as infinitely periodic (although a lattice of finite extent is sufficient to prove and apply the Bloch theorem). The Bloch theorem is in fact a generalization of the Floquet theorem, which concerns itself with one-dimensional systems. There is active research into the structure of operators associated with Floquet systems as they represent periodically driven quantum systems.

For now we state the theorem plainly, deferring proof and detailed discussion for another time.

Theorem. *For a Hamiltonian $\hat{H} = \frac{p^2}{2m} + U(x)$ defined on a lattice, where $U(x) = U(x+R)$ for any R in the Bravais lattice, the eigenfunctions can be written in*

the form $\phi_{n,q}(x) = e^{iq\cdot x}u_{n,q}(x)$, where x is any position in the lattice; n is the band index, and q is an element of the dual lattice, and $u(r) = u(R + R)$.

3 Analytical reduction

Our mission is to find a concise expression for the Bloch Hamiltonian in an optical lattice such that we may numerically diagonalize the operator. From Bloch's theorem we have the time-independent Schrödinger equation;

$$\begin{aligned} H\phi_{n,q}(x) &= E_{n,q}\phi_{n,q}(x) \\ He^{iq\cdot x}u_{n,q}(x) &= E_{n,q}e^{iq\cdot x}u_{n,q}(x) \end{aligned} \tag{1}$$

We note that the exponential functions $e^{iq\cdot x}$ form an orthogonal basis for the Hilbert space of single-atom eigenfunctions. Given that we can represent the eigenfunctions as linear combinations of these basis vectors it is naturally our goal to find these explicitly, and to find their associated eigenvalues. In order to do this we must find an explicit representation of the Hamiltonian with respect to this basis.

First we turn to the kinetic operator;

$$\begin{aligned} \frac{p^2}{2m}\phi_{n,q}(x) &= -\frac{\hbar^2}{2m}\nabla^2(e^{iq\cdot x}u_{n,q}(x)) \\ &= -\frac{\hbar^2}{2m}\nabla(iqe^{iq\cdot x}u_{n,q}(x) + e^{iq\cdot x}\nabla u_{n,q}(x)) \\ &= -\frac{\hbar^2}{2m}(-q^2e^{iq\cdot x}u_{n,q}(x) + 2iqe^{iq\cdot x}\nabla u_{n,q}(x) + e^{iq\cdot x}\nabla^2 u_{n,q}(x)) \\ &= \frac{\hbar^2}{2m}(q - i\nabla)^2\phi_{n,q}(x) \end{aligned} \tag{2}$$

The quantity $\hbar q$ is known as the crystal momentum, and is characteristic to the periodic potential just as the free particle momentum is to the uniform potential. Indeed, the periodic potential reduces to the flat potential when the periodic variation is identically zero, wherein the reciprocal lattice contains only the zero vector, and the action of the momentum operator reproduces the dynamics of a free particle.

To express the potential, consider a perfectly periodic infinite lattice with in one dimension, wherein

$$V(x) = V_l \sin^2(kx), \tag{3}$$

with the light wavenumber $k = 2\pi/\lambda$, and V_l is in units of the recoil energy $E_r = \frac{\hbar^2k^2}{2m}$. The lattice constant is given by $a = \lambda/2 = \pi/k$.

We note that the periodicity allows us to consider the Fourier transform of the potential in the primitive cell, $V(x) = \sum_\mu V_\mu e^{2ki\mu x}$. The lattice constant $a = \lambda/2 = \pi/k$ sets the fundamental spatial frequency $2\pi/a = 2k$. Moreover, $e^{iq'\cdot x} = e^{i(q+K)\cdot x} = e^{iq\cdot x}$ for any q', where K is an element of the dual lattice and q is a dual vector in the first Brillouin zone. Thus we can restrict the sum to

the set of allowed reciprocal vectors in the first Brillouin zone. For the potential in question we have

$$\begin{aligned} V_\mu &= \int_0^{\lambda/2} e^{-2ik\mu x} V(x) dx \\ &= \int_0^{\lambda/2} e^{-2ik\mu x} V_l \sin^2(kx) dx \\ &= -\frac{V_l}{4} \int_0^{\lambda/2} e^{-2ik\mu x} (e^{2ikx} + e^{-2ikx} - 2) dx \\ &= -\frac{V_l}{4} \int_0^{\lambda/2} (e^{2ik(1-\mu)x} + e^{-2ik(1+\mu)x} - 2e^{-2ik\mu x}) dx \\ \implies V_1 = V_{-1} &= -\frac{V_l}{4}; V_0 = \frac{V_l}{2} \end{aligned} \tag{4}$$

It remains to find the action of the potential and kinetic operators on the eigenfunctions ϕ. We first establish that for the periodic functions $u(x)$, we can take the Fourier transform

$$u_{n,q}(x) = \sum_\nu c_{n,q}(\nu) e^{2ik\nu x} \tag{5}$$

Then we may proceed by evaluating the following expressions;

$$\begin{aligned} V\phi_{n,q}(x) &= \sum_{\mu,\nu} V_\mu e^{2ki(\nu+\mu)x} c_{n,q}(\nu) e^{iq\cdot x} \\ \frac{p^2}{2m}\phi_{n,q}(x) &= \frac{\hbar^2}{2m}(q - i\nabla)^2 \sum_\nu c_{n,q}(\nu) e^{2ik\nu x} e^{iq\cdot x}. \end{aligned} \tag{6}$$

We are now equipped to find the spectrum of H and the corresponding stationary states. We begin again with the time-independent Schrödinger equation;

$$\begin{aligned} H\phi_{n,q}(x) &= E_{n,q}\phi_{n,q}(x) \\ &= (\frac{p^2}{2m} + V(x))\phi_{n,q}(x) \\ E_{n,q}u_{n,q}(x)e^{iq\cdot x} &= (\frac{\hbar}{2m}(q - i\nabla)^2 u_{n,q}(x) + \sum_{\mu,\nu} V_\mu e^{2ki\mu x} u_{n,q}(x)) e^{iq\cdot x} \end{aligned} \tag{7}$$

As this is identically true for each basis function $e^{iq\cdot x}$, we have

$$E_{n,q}u_{n,q}(x) = (\frac{\hbar^2}{2m}(q - i\nabla)^2 u_{n,q}(x) + \sum_{\mu,\nu} V_\mu e^{2ki\mu x} u_{n,q}(x))$$

$$E_{n,q}\sum_\nu c_{n,q}(\nu)e^{2ik\nu x} = \frac{\hbar^2}{2m}\sum_\nu(((q+2k\nu)^2 + \frac{V_l}{2})c_{n,q}(\nu) - \frac{V_l}{4}\sum_{\mu=1,-1} V_\mu c_{n,q}(\nu-\mu))e^{2ik\nu x}$$

$$\implies E_{n,q}c_{n,q}(\nu) = \frac{\hbar^2}{2m}(((q+2k\nu)^2 + \frac{V_l}{2})c_{n,q}(\nu) - \frac{V_l}{4}\sum_{\mu=1,-1} V_\mu c_{n,q}(\nu-\mu)) \tag{8}$$

Where the second step is made by changing the indexing in the potential series and collecting summation indices. Hence we must solve the eigenvalue equation given by the Fourier relation $\sum_j H_{\mu,\nu}c_{n,q}\nu = E_{n,q}c_{n,q}(\mu)$, where

$$H_{\nu,\nu} = E_r(\frac{q}{k} + 2\nu)^2 + \frac{V_l}{2}; H_{\nu,\nu\pm1} = -\frac{V_l}{4} \tag{9}$$

We note that the trivial gauge transformation $V \to V - \frac{V_l}{2}I$ allows us to set V_0 to an 0 without altering the physics of the problem, reducing the diagonal terms to $E_r(\frac{q}{k} + 2\nu)^2$.

4 Computational methods

The Hamiltonian may now be diagonalized easily by numerical methods. To fix the number of terms in the Fourier series, we examine the convergence of the eigenvalues and the power spectra of each of the eigenfunctions, as displayed in figure 1.

While the plots in figure 1 are not exhaustive they cover the realm of lattice parameters that are expected to be relevant to our experiment. Thus for our purposes terminating the Fourier series at $|\mu| = 5$ is generally sufficient.With this in mind, we proceed to numerically diagonalize the Hamiltonian above with 5 Fourier modes. We show the bandwidths of the ground and first two excited states in figure 2, and the reduced-zone representation of the dispersion relation in figure 3. The components of the eigenvectors obtained by numerical diagonalization are the Fourier coefficients of the dual representation of the Bloch eigenfunctions. Using these, we perform an inverse Fourier transform and reconstruct the wavefunctions of the Bloch atoms in lattices of varying depths, the results of which are shown in figure 4.

5 Moving forwards

Evidently there's more to cover here. The next step is to transform to the Wannier representation and recover the localized description of the independent

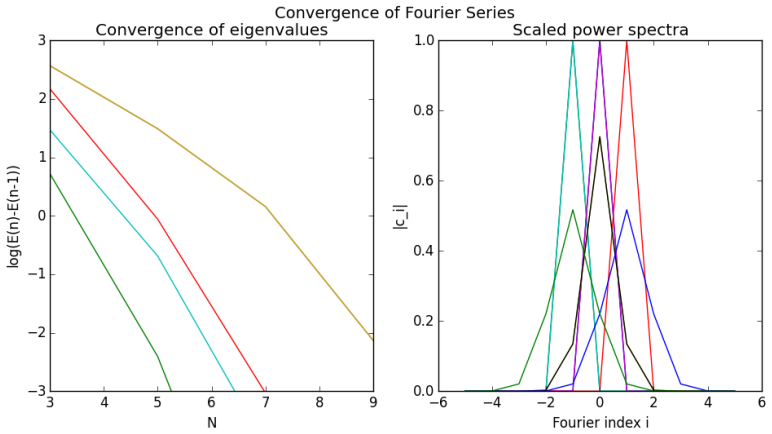

Figure 1: (a) Relative difference of eigenvalues vs. number of Fourier terms. The lower curve is for the ground state of a flat lattice, and the upper is for an excited state of a lattice of depth $30E_r$. (b) Scaled power spectra of eigenfunctions for varying lattice parameters, with depths between 0 and 30 E_r, including some excited states and states of nonzero crystal momentum.

atom wavefunctions. This will allow direct calculation of the tunnelling time and on-site interaction energy. Following this it is prudent to include the global harmonic confinement, which will give rise to a radially-dependent shift in the dispersion relation. An extension to 3D is natural. Following this we may make further considerations of classic solid-state properties with experimental relevance, such as the Fermi surface and the density of states. Incorporation of atomic interactions is the next logical step. The endgame is to produce a tractable model of the physical parameters of the lattice with fewer unrealistic idealizations. This will assist with lattice calibration and control, and provide a computational benchmark with which to compare empirical results.

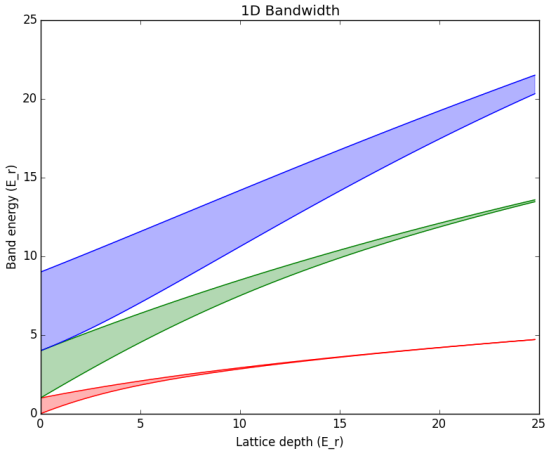

Figure 2:

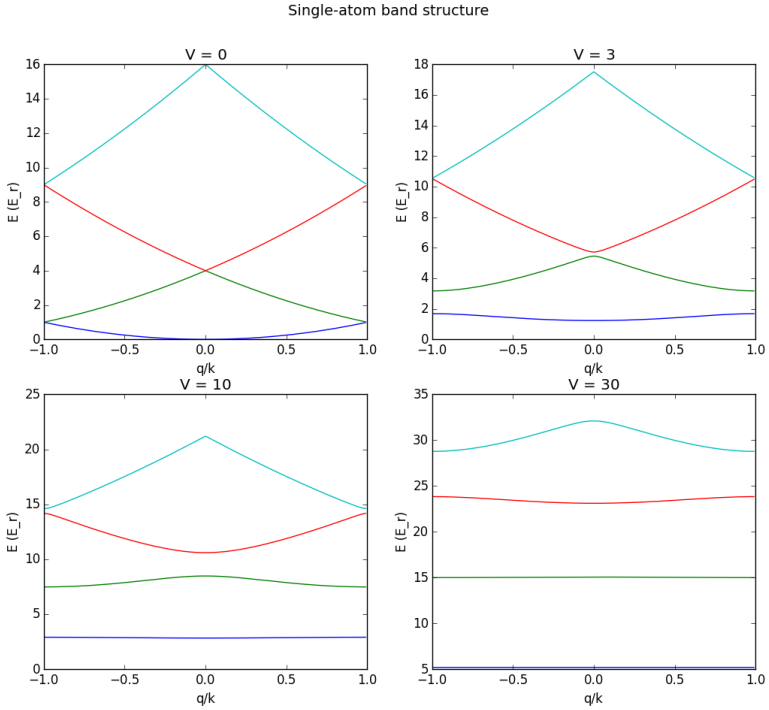

Figure 3:

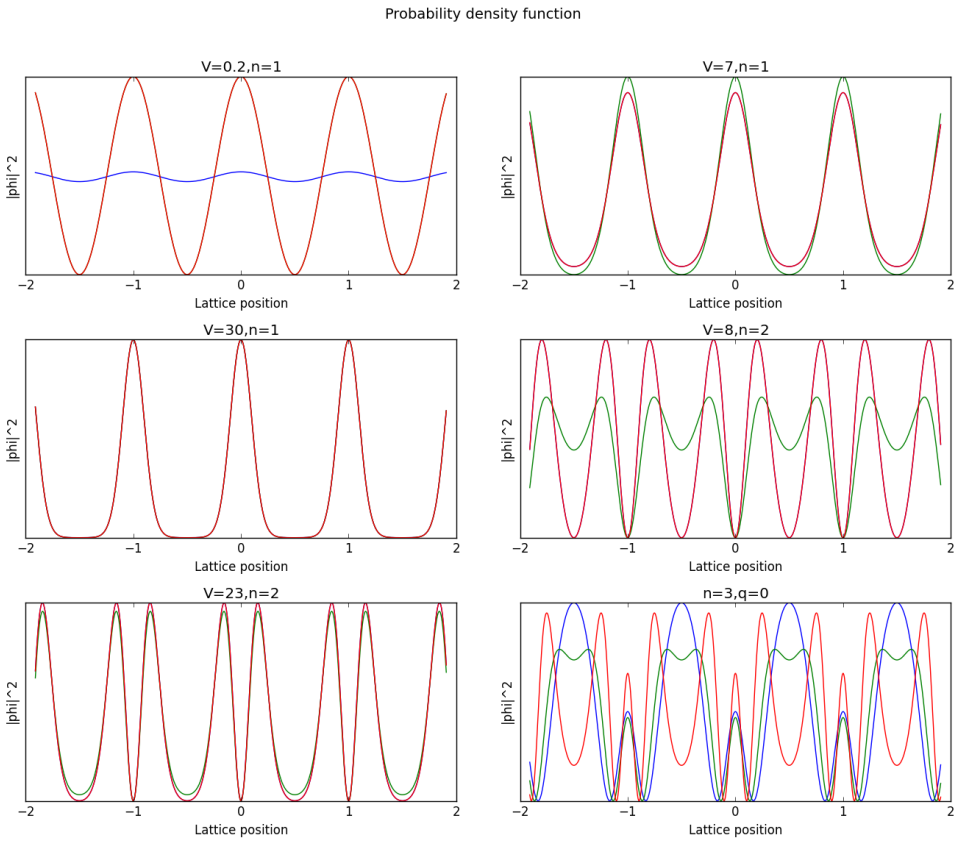

Figure 4: (a-e) Probability density functions for varying lattice depths, for q=0 (blue), q=1 (green) and q=2 (red). (f) Second excited state for lattice depths V=8 (blue), 15 (green), 30 (red).

6 Python code

Below are the numerical methods employed to produce the figures above. The plotting elements have been pruned for brevity.

```
def ev(q,V,N):
#Returns the eigenvalues and eigenvectors of H for given
    q and V with N terms either side of 0 in the Fourier
    series.
        D = numpy.diagflat([(2*l+q)**2+V/2.0 for l in
            range(-N,N+1)]) #Diagonal terms
        l = [-V/4.0 for x in range(0,2*N)] #Off-diagonal
            terms
        L = numpy.diagflat(l,1) + numpy.diagflat(l,-1)
        H = L + D
        return nl.eigh(H)

def econv(q,V): Plots the incremental change in
    eigenvalues as a function of the number of terms in
    the Fourier expansion. The result is used to justify
    the number of terms required for sufficient
    convergence of the band structure.
        XX = range(1,5)
        plt.plot([2*n+1 for n in XX],[numpy.log(abs(ev(q,
            V,n)[0][0]-ev(q,V,n-1)[0][0])) for n in XX])


def seespec(q,V,N,L):
#Visualizes the power spectrum of the L-th Bloch
    wavefunction. The result is used to set the bandwidth
    required to reconstruct the wavefunctions.
        Vex = ev(q,V,N)[1]
        plt.plot([x-N for x in range(0,2*N+1)],map(abs,
            Vex[:,L]**2))


def plotbandstructure(V,N):
#Generates the band structure plot for a given lattice
    depth to 2N+1 terms
        Q = numpy.arange(-1,1,0.01)
        bands = numpy.asmatrix([ev(q,V,N)[0] for q in Q])
        plt.plot(Q,bands[:,0],Q,bands[:,1],Q,bands[:,2],Q
            ,bands[:,3]) #Plot the band structure


def plotbandwidths(N):
```

```
#Plots bandwidths as a function of V to 2n+1 terms
        Q = numpy.arange(-1,1,0.1)
        V = numpy.arange(0,25,0.2)
        mins=[[min([ev(q,v,N)[0][i] for q in Q]) for v in
            V] for i in [0,1,2]]
        maxs=[[max([ev(q,v,N)[0][i] for q in Q]) for v in
            V] for i in [0,1,2]]
        plt.plot(V, mins[0], V, maxs[0],color='red')
        plt.fill_between(V,mins[0],maxs[0],color='red',
            alpha='0.3')

def ploteigfun(V,Q,N,b):
#Visualizes the probability density of the b-th band
    provided by the Bloch functions.
        I = [i for i in range(0,2*N+1)]
        X = numpy.arange(-6,6,0.01)
        nv = ev(Q,V,N)[1] [:,b]/(math.pi)
        phimodsquare =
            [abs(cmath.exp(complex(0,1.0)*Q*x)*numpy.sum
                ([nv[n]*cmath.exp(2*complex(0,1)*(n-0.5*(
                len(I)-1))*x) for n in I]))**2 for x in X]
        plt.plot([x/math.pi for x in X],phimodsquare)
```