

# Manual for DFT+*GW*+BSE and the related matters

Jinyuan Wu

February 6, 2023

# Contents

<b>1</b>	<b><i>GW</i> and BSE</b>	<b>4</b>
1.1	Preliminaries	4
1.1.1	Details in diagrammatics	4
1.1.1.1	Infinitesimals	4
1.1.1.2	The expression of $\Sigma$ and $W$	5
1.1.1.3	About “antiparticles”	6
1.1.1.4	The direction of momentum on the interaction line	7
1.1.2	“Quantitative” Feynman rules	9
1.1.2.1	In the free space	9
1.1.2.2	In a crystal	10
1.1.2.3	Diagrammatic components as tensors	13
1.1.3	Quasiparticles	13
1.1.3.1	Spectral density	14
1.1.4	Light-matter interaction	14
1.1.4.1	Interaction Hamiltonian	14
1.1.4.2	How to capture absorption	15
1.2	Overview of <i>GW</i> +BSE	15
1.2.1	What is <i>GW</i>	15
1.2.1.1	<i>GW</i> is screened Hartree-Fock approximation	15
1.2.1.2	Infinitesimal displacement of time in <i>GW</i>	16
1.2.1.3	<i>GW</i> compared with the Hartree-Fock approximation	16
1.2.1.4	BSE with the same order of approximation	17
1.2.2	The dielectric matrix $\epsilon$	17
1.2.2.1	Frequency-dependent form	17
1.2.2.2	The static limit and the generalized plasmon-pole model	18
1.2.2.3	The static subspace approach	20
1.2.3	The self-energy matrix $\Sigma$	20
1.2.3.1	COHSEX approximation	20
1.3	Accuracy of <i>GW</i>	20
1.3.1	Coverage	20
1.3.2	Diagonal or not	20
1.3.3	Self-consistent or not	20
1.3.4	On so-called failure of <i>GW</i> and convergence issues	21
1.4	The QuantumESPRESSO-BerkeleyGW ecosystem	21
1.4.1	Overview of the pipeline	21
1.4.2	Relativistic effects	21
1.4.3	Input and output of <b>pw</b>	21
1.4.4	<b>epsilon</b>	21
1.4.4.1	Procedure and speed	21
1.4.4.2	Divergence problems when $\mathbf{q} \rightarrow 0$	22
1.4.4.3	Frequency dependence of $\epsilon$	22
1.4.4.4	Console output	22
1.4.4.5	Other output files	23
1.4.5	Systems of units	23

<b>2</b>	<b>Tight-binding models</b>	<b>24</b>
2.1	Wannier functions and tight-binding models	24
2.2	The cRPA approach to obtain effective models	24
<b>3</b>	<b>Carrying out calculations</b>	<b>26</b>
3.1	Details in installation	26
3.1.1	QuantumESPRESSO	26
3.2	Standard operation procedures	26
3.2.1	Avoid data pollution	26
3.2.2	Finding the structure	26
3.2.2.1	From open data	26
3.2.2.2	Comparing existing structures for the same material	27
3.2.3	Insulator DFT+GW+BSE	27
3.2.3.1	The DFT stage	27
3.2.3.2	The GW stage	28
3.2.3.3	The BSE stage	29
3.2.4	Metal DFT+GW+BSE	29
3.2.5	Band plot	30
3.2.5.1	DFT level: <code>k-path</code>	30
3.2.5.2	GW level: <code>inteqp</code>	30
3.2.5.3	GW level: using <code>WFN_outer</code>	31
3.2.5.4	BSE level	31
3.2.6	Wannier functions and tight-binding models	31
3.2.6.1	<code>wannier90</code> for DFT	31
3.2.6.2	GW level: <code>sig2wan</code>	31
3.2.7	Self-consistent GW	31
3.2.7.1	Energy self-consistent calculation in GW	32
3.2.7.2	Eigenstate self-consistent calculation in GW	32
3.2.8	Topological invariants with <code>z2pack</code>	32
3.2.9	Band projection	32
3.3	Performance tricks	32
3.3.1	Parallelization	32
3.3.2	Choosing cutoff energies wisely	32
3.3.3	<code>pseudobands</code>	33
3.4	Convergence tests	33
<b>4</b>	<b>Trouble shooting</b>	<b>34</b>
4.1	Unexpected units	34
4.1.1	Band energy output of <code>pw.x</code>	34
4.2	Trouble shooting in MPI	34
4.2.1	<code>srun: fatal: Can not execute</code>	34
4.2.2	<code>error parsing parameters</code>	34
4.2.3	Each process is run serially and doesn't communicate with others	34
4.2.4	<code>nsufficient virtual memor</code>	34
4.3	Trouble shooting in Python	34
4.3.1	<code>AttributeError: 'Dataset' object has no attribute 'value'</code>	34
4.4	Trouble shooting in QuantumEspresso	35
4.4.1	<code>Intel MKL FATAL ERROR: Cannot load symbol MKLMPI_Get_wrappers.</code>	35
4.4.2	Program frozen	35
4.4.3	<code>Error in routine allocate_fft (1): wrong ngms</code>	35
4.4.4	<code>Error reading attribute index : expected integer , found *</code>	35
4.4.5	<code>cdiaghg (159): eigenvectors failed to converge</code>	35
4.4.6	<code>Error in routine cdiaghg (1052): problems computing cholesky</code>	35
4.4.7	<code>Error in routine set_occupations (1): smearing requires a vaklue for gaussian broadening (degauss)</code>	35
4.4.8	<code>Error in routine splitwf (36197): wrong size for pwt</code>	35
4.4.9	<code>Error in routine PW2BGW(19):input pw2bow</code>	35
4.4.10	<code>Error in routine PW2BGW (19): input_pw2bgw</code>	35

4.4.11	stress for hybrid functionals not available with pools . . . . .	35
4.4.12	Error in routine projwave (1): Cannot project on zero atomic wavefunctions! . . . . .	36
4.5	Trouble shooting in epsilon and sigma . . . . .	36
4.5.1	Selected number of bands breaks degenerate subspace. . . . .	36
4.5.2	WFN ifmin/ifmax fields are inconsistent . . . . .	36
4.5.3	Segmentation fault: address not mapped to object at address . . . . .	36
4.5.4	eqpcor mean-field energy mismatch . . . . .	37
4.5.5	ERROR: occupations (ifmax field) inconsistent between WFN and WFNq files. . . . .	37
4.5.6	ERROR: Unexpected characters were found while reading the value for the keyword . . . . .	37
4.5.7	forrtl: severe (24): end-of-file during read, unit -5, file Internal List-Directed Read . . . . .	37
4.5.8	ERROR: Inconsistent screening, truncation, or q0 vector . . . . .	37
4.5.9	cannot use metallic screening with q=0 . . . . .	37
4.5.10	ERROR: genwf mpi: No match for rkq point . . . . .	37
4.5.11	ERROR: Missing bands in file eqp_co.dat . . . . .	37
4.5.12	forrtl: severe (71): integer divide by zero . . . . .	38
4.5.13	ERROR: screened Coulomb cutoff is bigger than epsilon cutoff . . . . .	38
4.6	Trouble shooting in wannier90 and pw2wannier90 . . . . .	38
4.6.1	w90_wannier90_readwrite_read: mismatch in WTe2.eig . . . . .	38
4.6.2	WTe2.amn has not the right number of bands . . . . .	38
4.6.3	forrtl: severe (174): SIGSEGV, segmentation fault occurred . . . . .	38
4.7	Trouble shooting in kernel and absorption . . . . .	38
4.7.1	ERROR: Inconsistent symmetry treatment of the fine and shifted grids with the momentum operator . . . . .	38
4.8	Checklist for unexpected results . . . . .	38
4.8.1	Band symmetry higher than the space group shown at the beginning of bands.out . . . . .	38
4.8.2	Band structure looks very far from the literature . . . . .	38
4.8.3	Band plot is empty . . . . .	39
4.8.4	Band plot is not continuous . . . . .	39
4.8.5	The size of band gap . . . . .	39
4.8.6	SOC effects are too strong . . . . .	39
4.8.7	When we get a semimetal in the DFT step but it should be an insulator . . . . .	39

# Chapter 1

## $GW$ and BSE

### 1.1 Preliminaries

#### 1.1.1 Details in diagrammatics

This section briefly goes through some tricky aspects of Feynman diagram techniques that may seem puzzling when we do concrete calculations.

##### 1.1.1.1 Infinitesimals

There are some infinitesimals in Feynman rules that are often ignored. The first is about the illdefinedness of  $\mathcal{T} \langle c(t)c^\dagger(0) \rangle$  when  $t = 0$ . We want to make the propagator to be the particle number (so that if we evaluate the tadpole diagram, we get the Hartree term). Therefore, the contribution of an electron line is

$$\begin{aligned} \longrightarrow_k &:= \mathcal{T} \langle c_{\mathbf{k}}(t-0^+)c_{\mathbf{k}}^\dagger(0) \rangle \\ &= \int \frac{d\omega}{2\pi} e^{-i\omega(t-0^+)} \underbrace{\frac{i}{\omega - \xi_{\mathbf{k}} + i0^+ \text{sgn}(\xi_{\mathbf{k}})}}_{iG_0(\omega, \mathbf{k})} = \int \frac{d\omega}{2\pi} e^{-i\omega t} e^{i\omega 0^+} iG_0(\omega, \mathbf{k}). \end{aligned} \quad (1.1)$$

In the notation used here,  $G(t, 0)$  is proportional to  $\mathcal{T} \langle c(t)c^\dagger(0) \rangle$  and therefore is not well-defined when  $t = 0$ ; the infinitesimal therefore is introduced explicitly by adding the  $e^{i\omega 0^+}$  factor; some may *define* that  $iG$  is the correct propagator, and the  $e^{i\omega 0^+}$  factor or the small time displacement is embedded into the definition of  $G$ .

The necessity of this  $e^{i\omega 0^+}$  factor can also be seen by explicitly doing the integration: when  $t = 0$ , if we ignore the  $e^{i\omega 0^+}$  factor, we get

$$\int \frac{d\omega}{2\pi} \frac{i}{\omega - \xi_{\mathbf{k}} + i0^+ \text{sgn}(\xi_{\mathbf{k}})}.$$

This integral is not zero, but we want it to be zero when  $\xi_{\mathbf{k}} > 0$ , so we have to add a  $e^{i\omega 0^+}$  factor to make the integrand approaches zero quickly enough in the upper plane, so we can construct an integration contour in the upper plane, in which there is no pole, and

$$\int_{|\omega|=R \gg 1} e^{i\omega 0^+} \frac{d\omega}{2\pi} \frac{i}{\omega - \xi_{\mathbf{k}} + i0^+ \text{sgn}(\xi_{\mathbf{k}})} = 0.$$

Note that an interaction line should also receive the same mini-regularization outlined for Green functions, because it's obtained by integrating out some intermediate states. For bare Coulomb interaction this is not needed, because we don't have  $\omega$  dependence in the potential, and it makes no sense to discuss the poles when we change  $\omega$ . It does make sense to talk about retardation in the relativistic origin of Coulomb interaction: the Coulomb interaction is mediated

by virtual photons, and is therefore proportional to the off-shell (i.e.  $\omega \rightarrow 0$ ) limit of the photon propagator, which has  $\omega^2 - \mathbf{q}^2 + i0^+$  as the denominator, and we get

$$V(q) = \frac{4\pi e^2}{\mathbf{q}^2 - \omega^2 - i0^+}. \quad (1.2)$$

Also, for screened Coulomb interaction, the correct retardation is important, because now something looking like (1.2) appears again.

The  $1^+$  in  $\Sigma(1, 2) = iW(1^+, 2)G(1, 2)$  seems to be from the infinitesimal in the electronic Green function instead of the infinitesimal in the Coulomb interaction line (Section 1.2.1.2).

#### 1.1.1.2 The expression of $\Sigma$ and $W$

In this section I only consider how many imaginary units there are in front of Green functions, self energies, etc. Normalization factors like  $2\pi$  or  $V$  involved in summation of  $\mathbf{r}$  or  $\mathbf{k}$  are not considered.

The self-energy correction is visualized as the follows:

$$\text{Double line} = \text{Single line} + \text{Single line} \times \text{Loop}, \quad (1.3)$$

and from it we have

$$iG = iG_0 + iG_0 iG \times \text{Loop}.$$

It's then a good idea to define

$$-i\Sigma = \text{Loop}, \quad (1.4)$$

because in this case, we have

$$G = G_0 + GG_0\Sigma, \quad (1.5)$$

and therefore

$$\underbrace{\omega - \xi_{\mathbf{k}}^0}_{1/G_0} = \underbrace{\omega - \xi_{\mathbf{k}}}_{1/G} + \Sigma, \quad (1.6)$$

which agrees with the definition of the self energy as the shift of single-particle energy from the free dispersion.

It should be noted that the derivations above are mainly about how to set the position of  $i$  correctly: I left the  $e^{i\omega 0^+}$  factor (which can be found in (1.1)) out. After taking these factors into account, we find (1.5) becomes

$$e^{i\omega 0^+} iG = e^{i\omega 0^+} iG_0 + e^{i\omega 0^+} iG_0 \times \text{Loop} \times e^{i\omega 0^+} iG. \quad (1.7)$$

We can eliminate the common  $ie^{i\omega 0^+}$  factor (note that the magnitude of  $0^+$  doesn't matter, so a finite product of  $e^{i\omega 0^+}$  collapses into a single  $e^{i\omega 0^+}$ ), and thus we get (1.4) again; but note that

for the same “collapse of finite product of  $e^{i\omega 0^+}$ ” reason outlined above, the  $e^{i\omega 0^+}$  factor in the RHS of (1.4), if any, can also be removed.

Similarly, we define the corrected interaction line as

$$-iW = \left\{ \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} \right\}, \quad (1.8)$$

because in this way, when there is no interaction corrections, we have

$$W = \frac{e^2}{r} =: v. \quad (1.9)$$

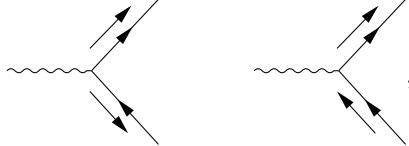
### 1.1.1.3 About “antiparticles”

The directions of momentum lines indicate whether the particles are created or annihilated. When the momentum arrow goes against the arrow on a line, we say this line is an antiparticle line. But here is a puzzle: we know there is no such thing as positrons in condensed matter physics, so what does “antiparticle” mean?

Here the problem lies on what it means to be the antiparticle of a kind of particle. In particle physics, when we do the particle-antiparticle transformation to an electron state whose polarization is one of the Dirac basis, a  $\psi^+$  particle is flipped into a  $\psi^-$  particle and vice versa. In condensed matter physics, the label of  $\psi^+$  and  $\psi^-$  (or  $\phi$  and  $\chi$  as people often call them:  $\psi = (\phi, \chi)$ ) is no longer there: the  $\chi$  modes in the Dirac field have already been integrated out. So electrons in condensed matter physics don’t really have antiparticles in the context of high energy physics.

Indeed, if we are still dealing with scattering problems in the non-relativistic limit, the antiparticle lines don’t appear at all! And similar to the case in QED (which can be checked in Peskin (A.6)), no separate momentum arrows parallel to the internal lines are needed: When calculating the propagator, the processes of both “an electron traveling forward” and “a hole traveling backward” are automatically covered together.

The antiparticle lines only appear when there are electrons in the ground state, which usually indicates there is a  $-\mu N$  term in the Hamiltonian so having some preexisting electrons lower the energy further, and this differs with the scattering case only in the rules pertaining to the external lines. For external lines, we now have diagrams like the following:



because now it’s possible to annihilate a preexisting electron in the ground state, but for internal lines, momentum labels can still be directly attached to the internal lines. This can be also seen by reckoning how Feynman rules are derived: the series we obtain by expanding  $e^{-iHt}$  contains field operators, not single creation or annihilation operators, and after Wick expansion, the correlation functions we get are all like  $\langle \bar{\psi}\psi \rangle$ , and of course an annihilation operator appearing in the expression of the out state in terms of the ground state can be contracted with a creative operator in  $e^{-iHt}$ , and this is visualized as an “antiparticle” external line with an outward momentum line. So here, the “particle-antiparticle transformation” is just swapping  $c_{\mathbf{k}}$  and  $c_{\mathbf{k}}^\dagger$  – this operation is still legit in condensed matter physics, because it doesn’t involve the  $\chi$  field; of course, the operation doesn’t create that kind of antiparticle in high energy physics.

No real modification happens to the propagator when there are electrons in the ground state. We have

$$\int_{-\infty}^{\infty} e^{i\omega t} dt \mathcal{T} \langle c_{\mathbf{k}}(t) c_{\mathbf{k}}^\dagger(0) \rangle = \frac{i}{\omega - \epsilon_{\mathbf{k}} + \mu}, \quad (1.10)$$

which can be straightforwardly obtained by looking at

$$H = \sum_{\mathbf{k}} \epsilon_{\mathbf{k}} c_{\mathbf{k}}^\dagger c_{\mathbf{k}} - \mu N \quad (1.11)$$

without doing any calculation.

Now we have to face the tough question: if antiparticle lines are there when there is a Fermi ball in the ground state, then why poles corresponding to antiparticles (whatever they are) are absent in the propagator? The answer is, for a  $\mathbf{k}$  on an antiparticle line appearing in diagrams, the corresponding pole can indeed be understood as a pole of an antiparticle: for an antiparticle line with momentum  $\mathbf{k}$ ,  $\mathbf{k}$  has to be under the Fermi surface in the ground state, so  $\omega_{\mathbf{k}} = \epsilon_{\mathbf{k}} - \mu < 0$ , and the point  $\omega = \omega_{\mathbf{k}}$  thus may be understood as an antiparticle pole. But here is a rather strong antisymmetry between particles and antiparticles: in external lines, when particles appear ( $\mathbf{k}$  over Fermi surface), antiparticles never appear; when antiparticles appear ( $\mathbf{k}$  below Fermi surface), particles never appear. The spectrum of electrons is split into two halves: for the part over the Fermi surface, only particles are visible, while for the part below the Fermi surface, only antiparticles are visible.

This means we can do away with antiparticle lines. By defining

$$b_{\mathbf{k}} = \begin{cases} c_{\mathbf{k}}, & \epsilon_{\mathbf{k}} > \mu, \\ c_{\mathbf{k}}^{\dagger}, & \epsilon_{\mathbf{k}} < \mu, \end{cases} \quad (1.12)$$

for  $\epsilon_{\mathbf{k}} < \mu$ , we have

$$\int_{-\infty}^{\infty} e^{i\omega t} dt \mathcal{T} \langle b_{\mathbf{k}}(t) b_{\mathbf{k}}^{\dagger}(0) \rangle = \frac{i}{\omega - \mu + \epsilon_{\mathbf{k}}}, \quad (1.13)$$

and now all poles have positive energies. It's also easy to replace  $c$  operators in all interaction vertices with  $b$  operators, so now, in the theory in terms of  $b$  operators, there is no antiparticle poles or Feynman diagrammatic antiparticle lines. Indeed,  $b$  operators give the true free excitation spectrum in a system with a non-zero chemical potential.

For  $\epsilon_{\mathbf{k}} < \mu$ ,  $b_{\mathbf{k}}^{\dagger}$  is said to *create* a **hole**. The energy of a hole is still positive: the energy of a state with a hole with momentum  $\mathbf{k}$  is

$$\sum_{\mathbf{k}' \neq \mathbf{k}, \epsilon_{\mathbf{k}'} < \mu} \epsilon_{\mathbf{k}'} - \mu(N-1),$$

and compared with the ground state, the energy of the hole is

$$\begin{aligned} E &= \sum_{\mathbf{k}' \neq \mathbf{k}, \epsilon_{\mathbf{k}'} < \mu} \epsilon_{\mathbf{k}'} - \mu(N-1) - \left( \sum_{\epsilon_{\mathbf{k}'} < \mu} \epsilon_{\mathbf{k}'} - \mu N \right) \\ &= \mu - \epsilon_{\mathbf{k}} > 0. \end{aligned} \quad (1.14)$$

So, we may say a hole is the antiparticle of an electron, but when we talk about the former, the latter is just a part of the background. Unlike the case in particle physics, where the electron and the positron are definitely two things, the hole and the electron are basically two *representations* of the *same* thing. (But this doesn't make talking about "annihilation between an electron and a hole" nonsense, because in an annihilation-between-electron-and-hole process, the  $n$  and  $\mathbf{k}$  numbers of the electron and the hole are different, so there is no problem of double counting the same mode in two representations, etc.) Once we choose the picture in which there are holes, the electron modes corresponding to the holes should be considered.

#### 1.1.1.4 The direction of momentum on the interaction line

##### Box 1.1: Convention when defining Fourier transformation

In this note, when we define the Fourier transformation for a function, we always do the integral over the *variable name*. That's to say, since we have

$$W(\omega) = \int dt e^{i\omega t} W(t),$$

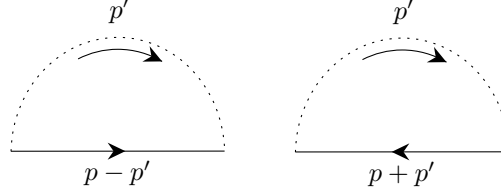
the Fourier integral of  $W(-t)$  is

$$\int dt e^{i\omega t} W(-t).$$



Thus, we can say “changing  $t$  to  $-t$  means changing  $\omega$  to  $-\omega$ ”. This convention makes doing actual calculation easier.

What is demonstrated in the last section, essentially, is “changing the direction of momentum line attached to a propagator in a diagram doesn’t create a new diagram”. The same works for the bare interaction line: changing the 4-momentum from  $p$  to  $-p$  doesn’t create a new diagram. Thus, the two diagrams below are actually *one* diagram, and only one of them should be calculated, or otherwise we have double counting:



If we write down the expressions of these diagrams, we get

$$\int \frac{d^4 p'}{(2\pi)^4} G(p + p') W(p'),$$

and

$$\int \frac{d^4 p'}{(2\pi)^4} G(p - p') W(p') = \int \frac{d^4 p'}{(2\pi)^4} G(p + p') W(-p').$$

To make the two diagrams equal to each other, what we need is  $W(p) = W(-p)$  ... or do we? It should be noted that if we replace  $W(\mathbf{p}, \omega)$  with  $W(-\mathbf{p}, -\omega)$ , the physical rules of the system *always* remains the same, as long as we have translational symmetries for time and space (if  $\mathbf{p}$  is a lattice momentum, then we only need translational symmetry for the lattice). This is *not* due to the time reversal symmetry! Instead, this is due to the fact that  $W(\mathbf{p}, \omega)$  only makes sense in

$$S_{\text{int}} = \sum_{p_1, p_2, q} c_{p_1+q}^\dagger c_{p_2-q}^\dagger W(q) c_{p_2} c_{p_1}.$$

By manipulating the variables, we easily find

$$\begin{aligned} S_{\text{int}} &= \sum_{p_1, p_2, q} c_{p_1-q}^\dagger c_{p_2+q}^\dagger W(-q) c_{p_2} c_{p_1} \\ &= \sum_{p_1, p_2, q} c_{p_2+q}^\dagger c_{p_1-q}^\dagger W(-q) c_{p_1} c_{p_2} \\ &= \sum_{p_1, p_2, q} c_{p_1+q}^\dagger c_{p_2-q}^\dagger W(-q) c_{p_2} c_{p_1}, \end{aligned}$$

where in the first line, we change  $q$  into  $-q$ , and in the second line, we swap the order of the operators, and get a  $(-1)^2 = 1$  factor, and in the third line, we swap the dummy variables  $p_1$  and  $p_2$ . Thus we can see  $W(q)$ , although not necessarily *equal* to  $W(-q)$ , is always *equivalent* to  $W(-q)$ .  $W(q) = W(-q)$  is of course true for the unscreened Coulomb interaction, and this directionless feature eventually originates from the directionless feature of the photon propagator; but it doesn’t have to be true if we want to replace  $W(q)$  with  $W(-q)$ : indeed, this means if  $W(q) \neq W(-q)$ , maybe it’s a good idea to *symmetrize*  $W(q)$ .

Some additional comments are needed when we put the time variable in the frequency space but keep the spatial coordinates in the real space. The  $W(q)$ -equivalent-to- $W(-q)$  conclusion now is that  $W(\mathbf{r}, \mathbf{r}', \omega)$  equals to  $W(\mathbf{r}', \mathbf{r}, -\omega)$ . We don’t need to add a minus sign to  $\mathbf{r}$  or  $\mathbf{r}'$ , but we need to swap them, so that

$$\int d^3 \mathbf{r} e^{-i\mathbf{p} \cdot (\mathbf{r} - \mathbf{r}')} W(\mathbf{r}', \mathbf{r}, -\omega) = W(-\mathbf{p}, -\omega),$$

while

$$\int d^3 \mathbf{r} e^{-i\mathbf{p} \cdot (\mathbf{r} - \mathbf{r}')} W(\mathbf{r}, \mathbf{r}', \omega) = W(\mathbf{p}, \omega).$$

The question, then, is what corresponds to the time reversal symmetric. Although the time reversal operation doesn't change  $\mathbf{r}$  and  $t$ , it changes the *order* of the coordinates: thus,  $W(\mathbf{r}, \mathbf{r}', \omega)$  equals to  $W(\mathbf{r}', \mathbf{r}, \omega)$ . Note that time reversal symmetry doesn't change  $\omega$ , if we swaps  $\mathbf{r}$  and  $\mathbf{r}'$ : this can be explicitly verified for the Green function. We have

$$\psi(\mathbf{x}, t) \xrightarrow{T} i\psi^\dagger(\mathbf{x}, -t), \quad (1.15)$$

and therefore

$$\begin{aligned} \mathcal{T} \langle \psi(\mathbf{x}, t) \psi^\dagger(\mathbf{x}', 0) \rangle &\xrightarrow{T} \mathcal{T} \langle i\psi^\dagger(\mathbf{x}, -t) i\psi(\mathbf{x}', 0) \rangle \\ &= -\mathcal{T} \langle \psi^\dagger(\mathbf{x}, -t) \psi(\mathbf{x}', 0) \rangle \\ &= \mathcal{T} \langle \psi(\mathbf{x}', 0) \psi^\dagger(\mathbf{x}, -t) \rangle. \end{aligned} \quad (1.16)$$

When the time translational symmetry is present, the last line becomes  $\mathcal{T} \langle \psi(\mathbf{x}', t) \psi^\dagger(\mathbf{x}, 0) \rangle$ . This can be explained quite intuitively: after time reversal operation, a process in which the electron moves from  $\mathbf{x}'$  to  $\mathbf{x}$  becomes a process in which the electron from  $\mathbf{x}$  to  $\mathbf{x}'$ , costing exactly the same amount of time. Thus after time reversal operation, we need to replace  $G(\mathbf{x}, t; \mathbf{x}', 0)$  with  $G(\mathbf{x}', t; \mathbf{x}, 0)$ , and therefore we need to replace  $G(\mathbf{r}, \mathbf{r}', \omega)$  with  $G(\mathbf{r}', \mathbf{r}, \omega)$ .

You may ask: so why can't we just replace  $t$  with  $-t$ ? We can, actually: but now we *shouldn't* change the order of  $\mathbf{r}$  and  $\mathbf{r}'$ ! Let's have a look at what it means to change  $t$  to  $-t$ :

$$\begin{aligned} W(\mathbf{r}, t; \mathbf{r}', 0) &\xrightarrow{t \rightarrow -t} W(\mathbf{r}, -t; \mathbf{r}', 0) \\ &\simeq W(\mathbf{r}', 0; \mathbf{r}, -t) \\ &= W(\mathbf{r}', t; \mathbf{r}, 0). \end{aligned}$$

The second line comes from the aforementioned fact that the initial and end labels in  $W$  can be swapped, which may change the value but doesn't influence the final result; the third line comes from time translational symmetry. So we see keeping  $\mathbf{r}$  and  $\mathbf{r}'$  untouched and changing  $t$  to  $-t$  is equivalent to keeping  $t$  untouched and swap  $\mathbf{r}$  and  $\mathbf{r}'$ . Both representations of the time reversal operation are good: but remember only to do one of them – don't do both of them!

In the  $(\mathbf{r}, \mathbf{r}', \omega)$  representation, if we take the first representation of the time reversal operation, we have

$$W(\mathbf{r}, \mathbf{r}', \omega) \xrightarrow{T} W(\mathbf{r}', \mathbf{r}, \omega), \quad (1.17)$$

while if we take the second representation, we get

$$W(\mathbf{r}, \mathbf{r}', \omega) \xrightarrow{T} W(\mathbf{r}, \mathbf{r}', -\omega). \quad (1.18)$$

Note that  $W(\mathbf{r}, \mathbf{r}', \omega)$  is always equivalent to  $W(\mathbf{r}', \mathbf{r}, -\omega)$ , but the former is only equivalent to  $W(\mathbf{r}', \mathbf{r}, \omega)$  when we have time reversal symmetry.

The above discussion is about a time reversal symmetry in a single term in the interaction Hamiltonian. It's possible to have a term that doesn't have time reversal symmetry in  $H_{\text{int}}$ , but then the complex conjugate of the aforementioned complex term appears in another diagram, and changing the direction of the momentum line merely swaps two diagrams.

## 1.1.2 “Quantitative” Feynman rules

The above discussions are all pretty loose; in this section I will deal with Feynman diagrams more “quantitatively” and rigorously.

### 1.1.2.1 In the free space

TODO:  $n$  interaction line  $\Rightarrow n$  independent momenta, not all of which are on Coulomb interaction lines

A large benefit of this formalism is it lifts the burden to worry about normalization concerning  $\delta$ -functions: no  $\delta$ -function appears in the rules now!

$$v(\mathbf{q}) = \frac{4\pi e^2}{q^2}. \quad (1.19)$$

### 1.1.2.2 In a crystal

There are several things that need attention concerning Feynman rules in condensed physics:

- The propagator is no longer  $1/(\omega - \mathbf{k}^2/2m)$ , but  $\propto \sum_{\mathbf{k}} \psi_{n\mathbf{k}}(\mathbf{r}) \psi_{n\mathbf{k}}^*(\mathbf{r}')/(\omega - \xi_{n\mathbf{k}})$ ;
- The crystal momentum  $\mathbf{k}$  appears above. To carry out Feynman diagrammatic calculations in an actual computer, a cutoff on the density of  $\mathbf{k}$  points is needed (essentially, a cutoff on the size of the system  $V$ ). This means we should replace  $\int d^3\mathbf{k}/2\pi$  with properly normalized  $\sum_{\mathbf{k}}$ , where  $\mathbf{k}$  goes over the discretely infinite  $\mathbf{k}$ -grid.
- Usually we limit all momenta to the **first Brillouin zone (1BZ)**, and therefore a sum over an unbounded momentum variable  $\mathbf{k}$  or  $\mathbf{q}$  has to be recast into  $\sum_G \sum_{\mathbf{k}} f(\mathbf{k} + \mathbf{G})$ .

Below I describe a set of Feynman rules that are compatible with the notation in [10]. Note that there is no controversy over normalization in the *real space* Feynman diagrams: although the inner structure of  $G(1, 2)$  has changed in the presence of the crystal potential, its relation with other diagrammatic components as abstract entities is never changed. Thus, starting from the space-in-real-space-and-time-in-frequency-space formalism may be a good idea: the free-space momentum formalism however is an important reference for normalization constants. In the space-in-real-space-and-time-in-frequency-space formalism, in a diagram containing  $n$  Coulomb interaction lines, the variables summed over are  $n$  frequency variables, and  $2n$  space variables, and we just need to do

$$\int \frac{d\omega_1}{2\pi} \cdots \int \frac{d\omega_n}{2\pi} \int d^3\mathbf{r}_1 \cdots \int d^3\mathbf{r}_{2n}. \quad (1.20)$$

We choose a normalization scheme for  $\psi_{n\mathbf{k}}$  such that

$$c_{n\mathbf{k}}^\dagger = \int d^3\mathbf{r} \underbrace{e^{i\mathbf{k}\cdot\mathbf{r}} u_{n\mathbf{k}}(\mathbf{r})}_{\psi_{n\mathbf{k}}} \psi^\dagger(\mathbf{r}), \quad \{c_{n\mathbf{k}}, c_{n'\mathbf{k}'}^\dagger\} = \delta_{nn'} \delta_{\mathbf{k}\mathbf{k}'}, \quad (1.21)$$

where  $\psi^\dagger(\mathbf{r})$  is the complex conjugate of the non-relativistic electron field operator. This leads to the normalization condition

$$\int d^3\mathbf{r} \psi_{n\mathbf{k}}^*(\mathbf{r}) \psi_{n'\mathbf{k}'}(\mathbf{r}) = \delta_{nn'} \delta_{\mathbf{k}\mathbf{k}'}, \quad (1.22)$$

and

$$G^0(\mathbf{r}, \mathbf{r}', \omega) = \sum_{n, \mathbf{k}} \frac{\psi_{n\mathbf{k}}(\mathbf{r}) \psi_{n\mathbf{k}}^*(\mathbf{r}')}{\omega - \xi_{n\mathbf{k}} + i0^+ \text{sgn}(\omega)}. \quad (1.23)$$

In the free-electron case, we expect

$$\psi_{n\mathbf{k}}(\mathbf{r}) = \frac{1}{\sqrt{V}} e^{i(\mathbf{k} + \mathbf{G}_n) \cdot \mathbf{r}}, \quad (1.24)$$

and therefore (note that the only possibility for  $\mathbf{G}_n + \mathbf{k}$  to be equal to  $\mathbf{G}_{n'} + \mathbf{k}'$  is for  $\mathbf{G}_n$  to be  $\mathbf{G}_{n'}$  and  $\mathbf{k}$  to be  $\mathbf{k}'$ )

$$\text{LHS of (1.22)} = \frac{1}{V} \int d^3\mathbf{r} e^{i\mathbf{r} \cdot (-\mathbf{G}_n - \mathbf{k} + \mathbf{G}_{n'} + \mathbf{k}')} = \frac{1}{V} V \delta_{\mathbf{G}_n + \mathbf{k}, \mathbf{G}_{n'} + \mathbf{k}'} = \delta_{nn'} \delta_{\mathbf{k}\mathbf{k}'},$$

and

$$\begin{aligned} \text{RHS of (1.23)} &= \frac{1}{V} \sum_{n, \mathbf{k}} e^{i(\mathbf{k} + \mathbf{G}_n) \cdot (\mathbf{r} - \mathbf{r}')} = \frac{1}{V} \sum_{\text{unbounded } \mathbf{k}} e^{i\mathbf{k} \cdot (\mathbf{r} - \mathbf{r}')} \\ &= \int \frac{d^3\mathbf{k}}{(2\pi)^3} e^{i\mathbf{k} \cdot (\mathbf{r} - \mathbf{r}')} =: \text{LHS of (1.23)}, \end{aligned}$$

so (1.22) and (1.23) go back to the free-space case.

Now we consider what happens when we encounter a Coulomb interaction line. The structure in which a Coulomb interaction line is connected to four electron lines gives rise to the following factor in the interpretation of the diagram:

$$M = \int d^3\mathbf{r} \int d^3\mathbf{r}' iG(\mathbf{r}_4, \mathbf{r}, \omega_1) iG(\mathbf{r}_3, \mathbf{r}', \omega_2) iG(\mathbf{r}', \mathbf{r}_2, \omega_3) iG(\mathbf{r}, \mathbf{r}_1, \omega_4) (-i)v(\mathbf{r} - \mathbf{r}'). \quad (1.25)$$

Note that here  $\omega_4$  is just a shorthand of  $\omega_1 + \omega_2 - \omega_3$  and this condition is *not* imposed by a  $\delta$ -function factor according to the rules listed above for the free-space case (if we want to make this expression really symmetric, we should add a  $2\pi\delta(\sum \omega)$  factor, and let, say,  $\int d\omega_4/2\pi$  explicitly impose the energy conservation condition for us). Inserting (1.23) into the above expression, and using the definition

$$v(\mathbf{r} - \mathbf{r}') = \int \frac{d^3\mathbf{p}}{(2\pi)^3} e^{i\mathbf{p}\cdot(\mathbf{r}-\mathbf{r}')} v(\mathbf{p}), \quad (1.26)$$

we have

$$\begin{aligned} M &= \psi_4(\mathbf{r}_4)\psi_3(\mathbf{r}_3)\psi_2^*(\mathbf{r}_2)\psi_1^*(\mathbf{r}_1) \\ &\times \int \frac{d^3\mathbf{q}}{(2\pi)^3} \sum_{1,2,3,4} \frac{i}{\omega_1 - \xi_4 + i0^+ \text{sgn}(\omega_1)} \frac{i}{\omega_1 - \xi_3 + i0^+ \text{sgn}(\omega_2)} \\ &\times \frac{i}{\omega_1 - \xi_2 + i0^+ \text{sgn}(\omega_3)} \frac{i}{\omega_1 - \xi_1 + i0^+ \text{sgn}(\omega_4)} \\ &\times \int d^3\mathbf{r} \psi_4^*(\mathbf{r})\psi_1(\mathbf{r})e^{i\mathbf{q}\cdot\mathbf{r}} \int d^3\mathbf{r}' \psi_3^*(\mathbf{r}')\psi_2(\mathbf{r}')e^{-i\mathbf{q}\cdot\mathbf{r}'} v(\mathbf{q}), \end{aligned} \quad (1.27)$$

where the label 1, 2, etc. mean  $(n_1, \mathbf{k}_1), (n_2, \mathbf{k}_2)$ , etc., which *doesn't* contain  $\omega$  variables. Since  $V$  is always large enough, we have

$$\int \frac{d^3\mathbf{q}}{(2\pi)^3} = \frac{1}{V} \sum_{\mathbf{G}} \sum_{\mathbf{q}},$$

and eventually we get

$$\begin{aligned} M &= \psi_4(\mathbf{r}_4)\psi_3(\mathbf{r}_3)\psi_2^*(\mathbf{r}_2)\psi_1^*(\mathbf{r}_1) \\ &\times \frac{1}{V} \sum_{\mathbf{G}} \sum_{\mathbf{q}} \sum_{1,2,3,4} \frac{i}{\omega_1 - \xi_4 + i0^+ \text{sgn}(\omega_1)} \frac{i}{\omega_1 - \xi_3 + i0^+ \text{sgn}(\omega_2)} \\ &\times \frac{i}{\omega_1 - \xi_2 + i0^+ \text{sgn}(\omega_3)} \frac{i}{\omega_1 - \xi_1 + i0^+ \text{sgn}(\omega_4)} \\ &\times \langle 4 | e^{i(\mathbf{q}+\mathbf{G})\cdot\mathbf{r}} | 1 \rangle \langle 3 | e^{-i(\mathbf{q}+\mathbf{G})\cdot\mathbf{r}} | 2 \rangle v(\mathbf{q} + \mathbf{G}). \end{aligned} \quad (1.28)$$

The structure of this expression reveals the Feynman rules for a crystal:

- For each external electron line, according to its direction, write down  $\psi(\mathbf{r})$  (outgoing) or  $\psi^*(\mathbf{r})$  (incoming).
- For each inner electron line, i.e. propagator, write down

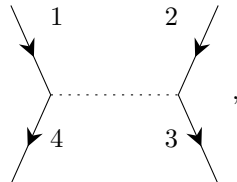
$$\overrightarrow{n, k} = \frac{i}{\omega - \xi_{n\mathbf{k}} + i0^+ \text{sgn}(\omega)} =: iG_{n\mathbf{k}}^0(\omega), \quad (1.29)$$

where  $k = (\omega, \mathbf{k})$ .

- For each Coulomb interaction line, write

$$\dots \overset{q, \mathbf{G}}{\dots} \dots = -i \frac{1}{V} v(\mathbf{q} + \mathbf{G}) \quad (1.30)$$

When an interaction line appears, enforce the momentum and energy conservation laws by hand. That's so say, when you see



we need to replace  $k_4$  by  $k_1 + k_2 - k_3$  manually: we just do a find-and-replace operation in the interpretation of this diagram. We *don't* integrate over  $k_4$ :  $k_4$  has already been replaced in every part of the interpretation of this diagram and is not an integration variable.

- For each vertex – here I mean the two-electron-line-one-dotted-line structure, not the four-electron-line structure (i.e. not the diagrammatic component corresponding to  $H_{\text{int}}$ ), we write  $\langle \text{out} | e^{\pm i(\mathbf{q} + \mathbf{G})} | \text{in} \rangle$ :

$$\begin{array}{c}
 n', k \\
 \searrow \\
 \text{---} \xrightarrow{q, \mathbf{G}} \text{---} \\
 \nearrow \\
 n, k - q
 \end{array} = \langle n, \mathbf{k} - \mathbf{q} | e^{-i(\mathbf{q} + \mathbf{G}) \cdot \mathbf{r}} | n', \mathbf{k} \rangle, \quad (1.31)$$

and

$$\begin{array}{c}
 n', k \\
 \searrow \\
 \text{---} \xleftarrow{q, \mathbf{G}} \text{---} \\
 \nearrow \\
 n, k + q
 \end{array} = \langle n, \mathbf{k} + \mathbf{q} | e^{i(\mathbf{q} + \mathbf{G}) \cdot \mathbf{r}} | n', \mathbf{k} \rangle. \quad (1.32)$$

The sign is decided by the direction of  $\mathbf{q}$ . The direction of  $\mathbf{q}$  doesn't matter: we *don't* sum over the two directions. The value of  $\mathbf{q}$  is determined by the values of free momentum variables; if  $\mathbf{q}$  is not selected as a free momentum variable, it shouldn't be integrated over.

Now we compare the above Feynman rules with the momentum space Feynman rules in Section 1.1.2.1. Note that here we distribute the space and time parts of  $(2\pi)^4$  differently: the space part,  $(2\pi)^3$ , becomes  $1/V$ , because we replace  $\int d^3\mathbf{q}$  by  $\sum_{\mathbf{G}} \sum_{\mathbf{q}}$ .

Since there are  $n$  free momentum variables and  $n$  Coulomb interaction lines, we can attribute the  $1/V$  factor to the interaction line, and we can also attribute it to the sum over  $\mathbf{q}$ . In [10], the  $1/V$  factor comes together with  $v(\mathbf{q} + \mathbf{G})$ . Indeed, they define

$$v(\mathbf{q} + \mathbf{G}) = \frac{4\pi e^2}{V|\mathbf{q} + \mathbf{G}|^2}. \quad (1.33)$$

The time part,  $2\pi$ , now comes with integrations of  $\omega$ 's. Thus, a ring diagram should be interpreted as

$$\begin{array}{c}
 n, k \\
 \curvearrowright \\
 \text{---} \quad \text{---} \\
 \curvearrowleft \\
 m, q - k
 \end{array} = \int \frac{d\omega}{2\pi} \sum_{m,n} iG_n^0(\mathbf{k}, \omega) iG_m^0(\mathbf{q} - \mathbf{k}, \omega_0 - \omega), \quad (1.34)$$

where  $q = (\omega_0, \mathbf{q})$ .

Another way to make sense of the Feynman rules in this section is to write down the Coulomb interaction Hamiltonian in the crystal momentum representation before deriving Feynman rules. The interaction Hamiltonian – the Coulomb repulsion Hamiltonian – is

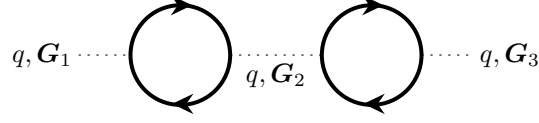
$$H = \sum_{1,2,3,4} c_4^\dagger c_3^\dagger H_{4321} c_2 c_1, \quad (1.35)$$

$$H_{4321} = \frac{1}{2} \int d^3\mathbf{r} \int d^3\mathbf{r}' \langle 4 | \mathbf{r} \rangle \langle 3 | \mathbf{r}' \rangle \langle \mathbf{r}' | 2 \rangle \langle \mathbf{r} | 1 \rangle v(\mathbf{r} - \mathbf{r}'), \quad (1.36)$$

Now again invoking the Fourier transformation from  $v(\mathbf{r} - \mathbf{r}')$  to  $v(\mathbf{q})$ , and discretize the integration over  $\mathbf{q}$ , the  $1/V$  factor appears again. The  $1/2$  prefactor is canceled in the same way it's canceled in the free-space case.

### 1.1.2.3 Diagrammatic components as tensors

Both the Coulomb interaction line and the band electron propagator line contains one discrete index. And there are actually *two* indices: for an interaction-corrected Coulomb line, the two edges of a  $W$  line can have different  $\mathbf{G}$  vectors, as is clearly shown below:



For the renormalized electron Green function, the same applies.

Note that for a free-space electron, the band index label  $n$  is essentially  $\mathbf{G}_n$ , but once the crystal potential and the self-energy is added, electrons can be scattered from one  $\mathbf{G}$  to another. One may want to diagonalize the effective Hamiltonian, but now the band index after diagonalization no longer corresponds to the index of  $\mathbf{G}$  vectors.

Thus, the  $\mathbf{G}$ - and  $n$ -summations mentioned in Section 1.1.2.2 can be recast into matrix forms:

$$\xrightarrow{k} = iG_{\mathbf{k}}^0(\omega) := i[\delta_{nn'}G_{n\mathbf{k}}^0(\omega)]_{nn'}, \quad (1.37)$$

$$\cdots \overset{q}{\cdots} = -iv(q) := -i[\delta_{\mathbf{G}\mathbf{G}'}v(\mathbf{q} + \mathbf{G})]_{\mathbf{G}\mathbf{G}'}, \quad (1.38)$$

and

$$\begin{array}{c} k \\ \swarrow \\ \text{---} \xrightarrow{q} \text{---} \\ \searrow \\ k+q \end{array} = M := [\langle n, \mathbf{k} + \mathbf{q} | e^{i(\mathbf{q} + \mathbf{G}) \cdot \mathbf{r}} | n' \mathbf{k} \rangle]_{nn' \mathbf{G}}, \quad (1.39)$$

$$\begin{array}{c} k \\ \swarrow \\ \text{---} \xrightarrow{q} \text{---} \\ \searrow \\ k+q \end{array} = M^* = [\langle n' \mathbf{k} | e^{-i(\mathbf{q} + \mathbf{G}) \cdot \mathbf{r}} | n, \mathbf{k} + \mathbf{q} \rangle]_{nn' \mathbf{G}}. \quad (1.40)$$

How to contract the indices can be done by following the standard quantum mechanic convention: the index of what happens later appears on the left side.

### 1.1.3 Quasiparticles

In the context of *ab initio* calculations, the term quasiparticle usually means renormalized band electrons and holes. Bosonic modes are just called “excitations”, and non-electron-like fermionic modes like spinons are simply absent (we are studying weakly correlated systems, anyway), so this terminology creates no confusion.

In most *ab initio* tasks, we expect the single-electron Green function to look like a “broadened” version of the free electron Green function. In this case, since  $G \sim 1/E - H$ , we say  $H$  is the single quasiparticle Hamiltonian, and the poles of the single-electron Green function are said to reflect the dispersion curve  $\varepsilon_{n\mathbf{k}}$  of quasiparticles. The correction to the  $\varepsilon_{n\mathbf{k}}$  is the most salient influence of Coulomb repulsion.

This however doesn’t mean the spectrum of the system *only* contains quasiparticles: for example, as is demonstrated in RPA of the empty-lattice model, we always have the plasmon mode besides quasiparticles; in insulators, the bound state of an electron and a hole is called an **exciton**, again something not a quasiparticle but a legit component of the system’s spectrum.

Even for quasiparticles, the effective single-electron energy is still dependent to the particle number distribution, and quasiparticle states have finite lifetime – they are not eigenstates of the system, after all. It should be noted in the full quantum treatment of the system, even when  $T = 0$ , we can still talk about damping: many-body wave functions containing a single quasiparticle are not strictly eigenstates of the system, so if we start with a single quasiparticle state, we will find that because of unitarity, probabilistic weight is “dissipated” into states we are not interested in or even can’t describe. A physical picture is that quantum fluctuation makes scattering possible when  $T = 0$ .

### Box 1.2: Quantum fluctuation

A more rigorous formulation of the idea is the follows: “quantum fluctuation” is a fancy word for “non-commuting terms in Hamiltonian”. Each term has to be, of course, meaningful in a basis that has direct physical meaning: in the basis of the eigenstates of the total Hamiltonian, nothing is non-commutative; but in the basis of, say, Fock states, we have two terms in the total Hamiltonian, the single electron term and the Coulomb repulsion term, each of which has direct meaning but doesn’t commute with the other. This non-commutation means an eigenstate of the system can never be a Fock state, but a mix of Fock states. Thus  $\langle \text{quasiparticle } n, \mathbf{k} | \text{ground state} \rangle \neq 1$ , and an initial state containing one quasiparticle always evolves away, and we say “quantum fluctuation enables scattering”.

The **Fermi liquid theory** describes the behavior of these (semi-)well-defined, bare electron or hole-like quasiparticles, if any. TODO: derive the follows

$$E = \varepsilon_{n\mathbf{k}} n_{n\mathbf{k}} + f n n. \quad (1.41)$$

The relation between diagrammatic components and the  $\varepsilon, f$  functions are given in [9].

#### 1.1.3.1 Spectral density

#### 1.1.4 Light-matter interaction

##### 1.1.4.1 Interaction Hamiltonian

The non-relativistic minimal coupling is

$$H = \frac{(\mathbf{p} - q\mathbf{A})^2}{2m} = \frac{(\mathbf{p} + e\mathbf{A})^2}{2m}, \quad (1.42)$$

and therefore the full light-matter interaction Hamiltonian is

$$H_{\text{light-matter}} = \frac{e\mathbf{p} \cdot \mathbf{A}}{m} + \frac{e^2 \mathbf{A}^2}{2m}. \quad (1.43)$$

Usually, the double-photon process is ignored, and we get

$$H_{\text{light-matter}} = \frac{e\mathbf{p} \cdot \mathbf{A}}{m}. \quad (1.44)$$

Note that here  $\mathbf{p}$  is the original momentum operator, i.e.  $-i\nabla$ .

The dipole approximation is

$$H_{\text{light-matter}} = -\mathbf{d} \cdot \mathbf{E}. \quad (1.45)$$

It can be derived when the effect of the outside electromagnetic field is predominantly electrostatic, and the system in question is restricted to a relatively small region. With the above two approximations, we can attribute the light-matter interaction to

$$H_{\text{light-matter}} = q\varphi \approx \text{const} + q\mathbf{r} \cdot \nabla\varphi \simeq \underbrace{q\mathbf{r}}_{-\mathbf{d}} \cdot \mathbf{E}. \quad (1.46)$$

Of course, when magnetic response of the system is important, we also need to add a magnetic dipole interaction term, etc.

Another light-matter interaction Hamiltonian is

$$H_{\text{light-matter}} = e\mathbf{v} \cdot \mathbf{A}. \quad (1.47)$$

The problem with this Hamiltonian is, when the potential term is not  $V(\mathbf{r})$ , we have

$$\mathbf{v} = \frac{1}{i}[\mathbf{r}, H] = \frac{\mathbf{p}}{m} + \frac{1}{i} \underbrace{[\mathbf{r}, V]}_{\neq 0}. \quad (1.48)$$

TODO: then why it works?

#### 1.1.4.2 How to capture absorption

Absorption is usually modeled by the imaginary part of  $\epsilon$ , and it's usually calculated by Fermi golden rule. There seems to be a contradiction here: Fermi golden rule looks “discrete”, while  $\text{Im } \epsilon$  gives us a continuous damping.

Recall how we treat spontaneous radiation using a quantum jump formalism: damping here is introduced by two (related) factors, the first being  $\text{Im } H_{\text{eff}}$ , the second being the quantum jump channels, and once the shapes of the two terms in the master equation are determined, the strengths of which are “equal” to some extent.

Now  $\text{Im } \epsilon$  is about  $\text{Im } H_{\text{eff}}$ , while Fermi golden rule is about the quantum jump channels, so it's indeed correct to infer  $\text{Im } \epsilon$  from Fermi golden rule, using a procedure like, say, comparing the amount of light absorbed calculated from  $\text{Im } \epsilon$  and from Fermi golden rule. There is no double counting in this procedure: *both*  $\text{Im } H_{\text{eff}}$  and  $\Gamma$  are needed for a full account of dissipation.  $H_{\text{eff}}$  continuously reduces the possibility to see the system staying in its original state, while quantum jump channels “confirm” that indeed the system decays to a lower energy state when the norm of the wave function has decreased considerably.

In the phenomenological model of spontaneous radiation, we first write down an  $H_{\text{eff}}$ , and then find the correct corresponding quantum jump channels to make the theory unitary, while here, we first calculate quantum jump channels (i.e. scattering) and then fit the  $H_{\text{eff}}$  according to the strength of scattering.

## 1.2 Overview of $GW+BSE$

### 1.2.1 What is $GW$

#### 1.2.1.1 $GW$ is screened Hartree-Fock approximation

In short,  $GW$  means to consider the Hartree term and the screened Fock term



$$(1.49)$$

as self-energy diagrams. Thus, apart from the Hartree term, we have

$$\Sigma = iGW, \quad (1.50)$$

where  $G$  is the renormalized Green function, and  $W$  is the renormalized (i.e. screened) Coulomb interaction, which is

$$W = \frac{v}{\epsilon} = \dots + \dots \bigcirc \dots + \dots \bigcirc \dots \bigcirc \dots + \dots, \quad (1.51)$$

$$\epsilon = 1 - \dots \bigcirc \dots. \quad (1.52)$$

Compared with the exact Hedin equations, no vertex correction is taken into account here in  $GW$ .





TODO:

- Why we say  $\Sigma = V_{\text{xc}} + \Sigma - V_{\text{xc}}$ ? (see [7] p. 1271)
- What should be symmetric? (11) and (12) in [7]???

### 1.2.2.1 Frequency-dependent form

$$q \cdots \cdots \text{loop} \cdots \cdots q = i\chi(q) \quad (1.56)$$

Now we have (every letter in the expression below is a matrix, so pay attention to the order)

$$W(q) = v(q) + v(q)\chi(q)v(q) + v(q)\chi(q)v(q)\chi(q)v(q) + \cdots = (1 - v(q)\chi(q))^{-1}v(q).$$

$$\epsilon(q) = 1 - v(q)\chi(q), \quad W(q) = \epsilon(q)^{-1}v(q). \quad (1.57)$$

$$(v(q)\chi(q))_{\mathbf{G}\mathbf{G}'} = \sum_{\mathbf{G}''} v(q + \mathbf{G}'') \delta_{\mathbf{G}\mathbf{G}''} \chi_{\mathbf{G}''\mathbf{G}'}(q).$$

$$\epsilon_{\mathbf{G}\mathbf{G}'}(q) = \delta_{\mathbf{G}\mathbf{G}'} - v(q + \mathbf{G})\chi_{\mathbf{G}\mathbf{G}'}(q). \quad (1.58)$$

$$[\epsilon_{GG'}(q)]_{GG'}^{-1} v(q + G').$$

$$\begin{aligned}
i\chi_{\mathbf{G}\mathbf{G}'}(\mathbf{q}, \omega) &= q, G \cdots \cdots \begin{array}{c} \text{---} n, k+q \text{---} \\ \text{---} n', k \text{---} \end{array} \cdots \cdots q, G' \\
&= - \int \frac{d\omega'}{2\pi} \sum_{\mathbf{k}} \sum_{n, n'} \frac{i}{\omega' - \xi_{n'\mathbf{k}} + i0^+ \text{sgn}(\xi_{n'\mathbf{k}})} \frac{i}{\omega + \omega' - \xi_{n, \mathbf{k}+\mathbf{q}} + i0^+ \text{sgn}(\xi_{n, \mathbf{k}+\mathbf{q}})} \\
&\quad \times M_{nn'}(\mathbf{k}, \mathbf{q}, \mathbf{G}) M_{nn'}^*(\mathbf{k}, \mathbf{q}, \mathbf{G}'),
\end{aligned} \tag{1.59}$$

where the minus sign comes from the closed electron loop, and to make the final expression agree with [7], I choose a rather strange direction of the  $\mathbf{q}$  arrow. The bands and wave functions are all renormalized ones.

The RHS of the equation can be evaluated straightforwardly: when  $\xi_{n'\mathbf{k}} < 0$ ,  $\xi_{n,\mathbf{k}+\mathbf{q}} > 0$ , we have

$$\begin{aligned} & \int \frac{d\omega'}{2\pi} \frac{1}{\omega' - \xi_{n'\mathbf{k}} + i0^+ \operatorname{sgn}(\xi_{n'\mathbf{k}})} \frac{1}{\omega + \omega' - \xi_{n,\mathbf{k}+\mathbf{q}} + i0^+ \operatorname{sgn}(\xi_{n,\mathbf{k}+\mathbf{q}})} \\ &= \int \frac{d\omega'}{2\pi} \frac{1}{\omega' - \xi_{n'\mathbf{k}} - i0^+} \frac{1}{\omega + \omega' - \xi_{n,\mathbf{k}+\mathbf{q}} + i0^+} \\ &= \frac{2\pi i}{2\pi} \frac{1}{\omega + \xi_{n'\mathbf{k}} + i0^+ - \xi_{n,\mathbf{k}+\mathbf{q}} + i0^+}, \end{aligned}$$

where in the third line we complete the integral contour in the upper plane. Similarly, when  $\xi_{n'\mathbf{k}} > 0$ ,  $\xi_{n,\mathbf{k}+\mathbf{q}} < 0$ , we have

$$\begin{aligned} & \int \frac{d\omega'}{2\pi} \frac{1}{\omega' - \xi_{n'\mathbf{k}} + i0^+ \operatorname{sgn}(\xi_{n'\mathbf{k}})} \frac{1}{\omega + \omega' - \xi_{n,\mathbf{k}+\mathbf{q}} + i0^+ \operatorname{sgn}(\xi_{n,\mathbf{k}+\mathbf{q}})} \\ &= \int \frac{d\omega'}{2\pi} \frac{1}{\omega' - \xi_{n'\mathbf{k}} + i0^+} \frac{1}{\omega + \omega' - \xi_{n,\mathbf{k}+\mathbf{q}} - i0^+} \\ &= \frac{2\pi i}{2\pi} \frac{1}{-\omega + \xi_{n,\mathbf{k}+\mathbf{q}} + i0^+ - \xi_{n'\mathbf{k}} + i0^+} \end{aligned}$$

When  $\xi_{n'\mathbf{k}}$  and  $\xi_{n,\mathbf{k}+\mathbf{q}}$  are both positive or negative, the integral is zero. Thus, we find

$$\begin{aligned} \chi_{GG'}(\mathbf{q}, \omega) &= \sum_{\mathbf{k}} \sum_{n,n'} M_{nn'}(\mathbf{k}, \mathbf{q}, \mathbf{G}) M_{nn'}^*(\mathbf{k}, \mathbf{q}, \mathbf{G}') \\ &\times \left( \frac{\theta(-\xi_{n'\mathbf{k}})\theta(\xi_{n,\mathbf{k}+\mathbf{q}})}{\omega + \xi_{n'\mathbf{k}} - \xi_{n,\mathbf{k}+\mathbf{q}} + i0^+} + \frac{\theta(\xi_{n'\mathbf{k}})\theta(-\xi_{n,\mathbf{k}+\mathbf{q}})}{-\omega + \xi_{n,\mathbf{k}+\mathbf{q}} - \xi_{n'\mathbf{k}} + i0^+} \right). \end{aligned} \quad (1.60)$$

TODO: occupied, or empty:

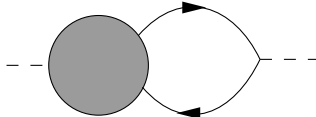
Note that in [7], the normalization of  $\chi$  is the same as here, but the definition of  $v(\mathbf{q})$  doesn't include the  $1/V$  factor. So (8) in [7] is wrong: it leads to a ridiculous result: as we increase the density of the  $\mathbf{k}$ -grid,  $\chi$  becomes larger and larger!

### 1.2.2.2 The static limit and the generalized plasmon-pole model

In the  $\omega \rightarrow 0$  limit, we have

$$\begin{aligned} \chi_{GG'}(\mathbf{q}, \omega = 0) &= \sum_{\mathbf{k}} \sum_{n,n'} M_{nn'}(\mathbf{k}, \mathbf{q}, \mathbf{G}) M_{nn'}^*(\mathbf{k}, \mathbf{q}, \mathbf{G}') \\ &\times \left( \frac{\theta(-\xi_{n'\mathbf{k}})\theta(\xi_{n,\mathbf{k}+\mathbf{q}})}{\varepsilon_{n'\mathbf{k}} - \varepsilon_{n,\mathbf{k}+\mathbf{q}}} + \frac{\theta(\xi_{n'\mathbf{k}})\theta(-\xi_{n,\mathbf{k}+\mathbf{q}})}{\varepsilon_{n,\mathbf{k}+\mathbf{q}} - \varepsilon_{n'\mathbf{k}}} \right). \end{aligned} \quad (1.61)$$

A frequently used way to obtain some degree of frequency dependence is the **generalized plasmon pole model (GPP)**. The logic is like this. We know  $D = \epsilon E$ ; although in this note  $\epsilon$  relates the bare *scalar potential* and the screened *scalar potential*, the physical picture is the same. This means  $1/\epsilon$  can be seen as the response (in the Feynman diagrammatic sense; I'm not saying  $1/\epsilon$  is a response function in linear response theory; indeed its pole structure is the same with time-ordered Green function, not a response function) of  $E$  to an externally introduced charge, which looks like this:



So of course,  $1/\epsilon$  can also be seen as unity plus the sum of Green functions of electron density modes. We haven't introduced any approximation yet, so now let's introduce a rather strong approximation:  *$1/\epsilon - 1$  only contains the plasmon mode, the dispersion relation of which is completely flat.* We know this approximation is far from accurate: we always have electron-hole pairs and also excitons besides the plasmon, and the dispersion relation of the plasmon is not strictly flat; but anyway let's just take this as the starting point: in cases where what's important

concerning  $\epsilon$  is only a handful of parameters obtained from it (like  $\omega_p$ ), our rather coarse model should work well. In other parts of condensed matter physics, we encounter similar rather coarse models, like the Debye model, which however are (semi-)quantitative.

The structure of  $1/\epsilon$ , therefore, becomes

$$\frac{1}{\epsilon} - 1 = \text{const.} \times \frac{1}{\omega^2 - \omega_0^2 + i0^+}, \quad (1.62)$$

where  $\omega_0$  is the “plasmon frequency” which is assumed to be the same when we change the plasmon wave vector  $\mathbf{q}$ . This equation can be rewritten into

$$\frac{1}{\epsilon} - 1 = \frac{\Omega^2}{2\tilde{\omega}} \left( \frac{1}{\omega - \tilde{\omega} + i0^+} - \frac{1}{\omega + \tilde{\omega} - i0^+} \right). \quad (1.63)$$

This is just the GPP; we call it *generalized* plasmon pole model, because  $\tilde{\omega}$  isn’t necessarily the plasmon frequency obtained with the free-electron approximation:

$$\omega_p = \sqrt{\frac{ne^2}{m\epsilon_0}}. \quad (1.64)$$

When  $\tilde{\omega} = \Omega = \omega_p$ , we just get the Drude-like

$$\epsilon = 1 - \frac{\omega_p^2}{\omega^2}. \quad (1.65)$$

The next question is how to find  $\Omega$  and  $\tilde{\omega}$ . Of course we need two constraints, and the first is of course obtained from (1.61). The second constraint is the  $f$ -sum rule: TODO Putting the two constraints together, we are able to fix  $\epsilon(\omega)$  once  $\epsilon(0)$  is found by calculating (1.61).

The GPP also justifies why we need to write  $G(1, 2)W(1^+, 2)$ : in the frequency domain, this is equivalent to adding a  $e^{i0^+\omega}$  factor, where  $\omega$  is the frequency on the Coulomb interaction line. Thus the  $GW$  self-energy is proportion to the following factor:

$$\int d\omega' e^{i0^+\omega'} \frac{1}{\omega + \omega' - \xi_{n\mathbf{k}} + i0^+ \text{sgn}(\xi_{n\mathbf{k}})} \left( \frac{1}{\omega' - \tilde{\omega} + i0^+} - \frac{1}{\omega' + \tilde{\omega} - i0^+} \right).$$

When  $\xi_{n\mathbf{k}} > 0$ , we find only the pole of the  $1/(\omega' + \tilde{\omega} - i0^+)$  term contributes to the integral, and the result is proportional to

$$\frac{1}{\omega - \tilde{\omega} - \xi_{n\mathbf{k}}},$$

which means the singularity is achieved when  $\xi_{n\mathbf{k}} = \omega - \tilde{\omega}$ . Note that  $\xi_{n\mathbf{k}}$  may be seen as the energy of the inner electron line, and  $\omega$  is the input energy, so this means the energy of the inner electron is the input energy minus  $\tilde{\omega}$ , which is correct according to the physical picture of screening. When  $\xi_{n\mathbf{k}} < 0$ , all terms contribute to the integral, but now the residue of the pole of  $\xi_{n\mathbf{k}} = \omega - \tilde{\omega}$  cancels with the residue of the pole of the electron propagator, and the result is proportional to

$$\frac{1}{\xi_{n\mathbf{k}} - \omega - \tilde{\omega}}.$$

This means  $\xi_{n\mathbf{k}}$  is  $\omega$  – the input energy – plus  $\tilde{\omega}$ . But note that in this case  $\xi_{n\mathbf{k}} < 0$ , so the above expression means  $\xi_{n\mathbf{k}}$  is less negative than the input energy, and thus, the excited hole is closer to the Fermi surface, so the energy of the hole represented by the inner electron propagator is *lower* than  $|\omega| = -\omega$ , which is the energy of the input hole. In both case, the  $e^{i0^+\omega}$  factor gives the correct physical picture. The origin of this factor is discussed in Section 1.2.1.2.

### 1.2.2.3 The static subspace approach

## 1.2.3 The self-energy matrix $\Sigma$

### 1.2.3.1 COHSEX approximation

## 1.3 Accuracy of $GW$

### 1.3.1 Coverage

$GW$  proves to be accurate enough for most weakly-correlated systems. TODO: any counter-examples? Recently, it's also applied successfully to systems like polymers, nano-wires and molecules.

### 1.3.2 Diagonal or not

We know in the momentum space, we have

$$E_{n\mathbf{k}}^{\text{QP}} = E_{n\mathbf{k}}^0 + \Sigma_{n\mathbf{k}}(E^{\text{QP}}). \quad (1.66)$$

Here since  $\Sigma$  depends on the corrected propagator,  $E_{n\mathbf{k}}^{\text{QP}}$  enters its expression. The cost of  $GW$  calculation means we need to first do a DFT calculation and feed this as the input of the  $GW$  package (the former usually mysteriously called the “mean field” step, though we may also say  $GW$  is a mean-field method; on the other hand, in principle – though of course not in practice – DFT is able to decide everything about the system), so (1.66) now is

$$E_{\mathbf{k}}^{\text{QP}} = E_{\mathbf{k}}^{\text{KS}} + \Sigma_{\mathbf{k}}(E^{\text{QP}}) - \Sigma^{\text{KS}}. \quad (1.67)$$

Here  $\Sigma^{\text{KS}}$  is the so-called DFT self-energy, i.e. the Hartree potential plus the exchange-correlation potential. Note that here I don't insert band indices into the equation, because  $\Sigma_{\mathbf{k}}$  may mix different bands together, and (1.67) is an equation about matrices, essentially a single-electron Schrodinger equation. Its first order approximation is

$$E_{n\mathbf{k}}^{\text{QP}} = E_{n\mathbf{k}}^{\text{KS}} + \langle \psi_{n\mathbf{k}}^{\text{KS}} | \Sigma(E_{n\mathbf{k}}^{\text{QP}}) - \Sigma^{\text{KS}} | \psi_{n\mathbf{k}}^{\text{KS}} \rangle. \quad (1.68)$$

TODO: when doing iterative calculation,  $V_{\text{H}}$  may no longer be the same between DFT and  $GW$ ?? So when doing iterative calculation, should we calculate “the  $GW$   $V_{\text{H}}$ ”?

### 1.3.3 Self-consistent or not

There are three iterative schemes. The first is the eigenvalue self-consistent scheme: It's just a self-consistent solver of (1.68). In this case, we don't need off-diagonal elements, because they are not used in (1.68). This scheme is mentioned in Section 3.3 in [7]. The second scheme takes the change of eigenstates into account, and thus iteratively solves (1.67). In this case we need to take non-diagonal elements seriously [1, 8]. In the third scheme, the form of  $\Sigma$  itself is changed: Recall that we need an **epsilon** step to calculate  $\epsilon$  and thus the screened interaction potential  $W$ , and  $\Sigma = iGW$ . This in general is not recommended, because we know  $GW$  tends to widen the band gap, and sometimes as we iteratively update the band gap, it becomes too large. The origin of this overestimation of band gap is that  $GW$  neglects the vertex, so iterative  $GW$  only leads us towards the more and more inaccurate way.

The non-self-consistent  $G_0W_0$  calculation proves to be a better choice empirically, if the initial DFT input is of good quality – and here there is another empirical observation that sometimes LDA functional together with  $G_0W_0$  provides better results. Still, the argumentation provided above only explains why iterative  $GW$  is bad, but doesn't explain why one-shot  $GW$  is good. In other words, we need to know how certain factors in the one-shot  $GW$  scheme somehow makes up for the missing vertex correction.

One possible form of the vertex is the electron-hole interaction, which is calculated by solving the BSE. Now an empirical fact is LDA tends to give the same band gap as BSE, leading to a pretty good one-shot approximation.

TODO: [15]

The question, then, is why LDA in some cases works as well as BSE. The reason for this is because of the relation between the derivative discontinuity in DFT and electron-hole interaction kernel TODO: the relation with [13]

### 1.3.4 On so-called failure of $GW$ and convergence issues

Some (weak-correlated, of course) materials are claimed to be impossible to be characterized correctly using  $GW$ , or at least  $G^0W^0$ . [14] refutes such a claim, at least for ZnO.

The root for this seems to be poor convergence test: people often use insufficient number of bands, etc.

See <https://www.nerisc.gov/assets/Uploads/ConvergenceinBGW.pdf>

## 1.4 The QuantumESPRESSO-BerkeleyGW ecosystem

### 1.4.1 Overview of the pipeline

TODO: whether and how BerkeleyGW supports HSE06. See <https://groups.google.com/a/berkeleygw.org/g/help/c/I>

Note that the division of labor is different in the  $GW$  step and the BSE step. The **sigma** program doesn't really do diagonalization, so building  $\Sigma$  and finding quasiparticle energies are done in one step, which is implemented in **sigma**. On the other hand, diagonalization *is* needed for BSE, so building the kernel – counterpart of  $\Sigma$  – is done in one step (**kernel**), while diagonalizing it is done in another step (**absorption**).

### 1.4.2 Relativistic effects

BerkeleyGW supports SOC calculation, with the so-called fully relativistic mode and the scalar relativistic mode; I say “so-called” because all the calculations are still done in the framework of Schrodinger equation, with relativistic effects being introduced by adding perturbative terms like  $\mathbf{L} \cdot \mathbf{S}$  or  $\mathbf{p}^4$  [3]. On the other hand, when it comes to calculating the inner structure of atoms (which, in condensed matter physics, is usually seen when we generate our own pseudopotentials), being “fully relativistic” means using *Dirac equation*.

### 1.4.3 Input and output of pw

### 1.4.4 epsilon

#### 1.4.4.1 Procedure and speed

What **epsilon** does, as its name implies, is to calculate  $\epsilon$  – and since  $\epsilon$  is used to find  $W$ , we need  $\epsilon^{-1}$ . The relative equations are (8-10) in [7]; relative discussions can be found in Section 1.2.2. There are three steps in **epsilon**:

1. Calculate  $M_{nn'}(\mathbf{k}, \mathbf{q}, \mathbf{G})$ ;
2. Summing over  $n, n', \mathbf{k}$ ;
3. Finding  $\epsilon^{-1}$ .

With a fixed accuracy requirement, the time cost of first step is  $\sim N^3 \log N$ , where  $N$  is the number of atoms per unit cell. The  $\mathbf{k}$  and  $\mathbf{q}$  points are given by the  $\mathbf{q}$ -grid given in **epsilon.inp** and the  $\mathbf{k}$ -grid in the wave function files, so they are fixed and are not a part of the scaling. With the cutoff energy fixed, the size of the  $\mathbf{G}$ -grid is proportional to  $V$ , which is in turn proportional to  $N$  (the distance between atoms is roughly fixed, and therefore the more atoms we have, the larger the unit cell is). With a fixed accuracy standard, the required numbers of occupied bands and empty bands are all  $\sim N$ , so the number of matrix elements of  $M_{nn'}(\mathbf{k}, \mathbf{q}, \mathbf{G})$  scales as  $N^3$ . For each matrix element, we need to calculate  $\langle n, \mathbf{k} + \mathbf{q} | e^{i(\mathbf{q} + \mathbf{G}) \cdot \mathbf{r}} | n', \mathbf{k} \rangle$ . Note that the matrix inside contains only  $\mathbf{r}$ , and the expression therefore can be evaluated as

$$\int d^3\mathbf{r} \phi_{n, \mathbf{k} + \mathbf{q}}^*(\mathbf{r}) e^{i(\mathbf{q} + \mathbf{G}) \cdot \mathbf{r}} \phi_{n', \mathbf{k}}(\mathbf{r}),$$

and the scale of the calculation needed is proportional to  $V$  and again  $N$ . In practice, we use the  $\mathbf{G}$  representation to calculate the matrix element, and again the time cost is  $\sim N$ . (Note that the two estimations are equivalent: by saying the time cost is proportional to  $V$ , we implicitly imply the absolute spatial resolution is fixed, which, in other words, means we fix the cutoff energy.) So naively, the time cost is  $\sim N^4$ . Fortunately the matrix element  $M_{nn'}(\mathbf{k}, \mathbf{q}, \mathbf{G})$  with

fixed  $n, n', \mathbf{G}$  can be evaluated by fast Fourier transformation, so eventually, the time cost scales like  $N^3 \log N$ .

The time cost of the second step, in a serial program, scales like  $N^4$ . We sum over  $n$  and  $n'$ , each of which has  $\sim N$  values. And we need to calculate  $\chi_{\mathbf{G}\mathbf{G}'}$ , where the values of  $\mathbf{G}$  and  $\mathbf{G}'$  are all roughly proportional to  $N$ . So the final scaling of the time cost is  $N^4$ . This however can be parallelized, and eventually, in a well optimized parallelized package, the time cost scales like  $N^2$ .

The time cost of the third step – the matrix inversion step – scales like  $N^3$ .

#### 1.4.4.2 Divergence problems when $\mathbf{q} \rightarrow 0$

When calculating  $\epsilon_{\mathbf{G}\mathbf{G}'}(\mathbf{q}, \omega = 0)$ , we notice that when  $\mathbf{G} = \mathbf{G}' = 0$ , the matrix element diverges as  $\mathbf{q} \rightarrow 0$ . For an insulator, we have

$$\begin{aligned} \epsilon_{00}(\mathbf{q} \rightarrow 0, \omega = 0) &\propto -\frac{1}{q^2} \chi_{00}(\mathbf{q} \rightarrow 0, \omega = 0) \\ &\propto \sum_{n \text{ occupied}, n' \text{ empty}} -\frac{1}{q^2} |\langle n\mathbf{k} | 1 + i(\mathbf{q} + \mathbf{G}) \cdot \mathbf{r} + \dots | n'\mathbf{k} \rangle|^2 \\ &\propto \text{const.} \times \frac{q^2}{q^2}. \end{aligned} \quad (1.69)$$

Here the first term in the Taylor expansion of  $e^{i(\mathbf{q}+\mathbf{G})\cdot\mathbf{r}}$  vanishes because of orthogonality conditions. We see  $\epsilon_{00}(\mathbf{q} \rightarrow 0, \omega = 0)$  has a definite value.

For a metal, some bands are both occupied and empty, so we can no longer use the orthogonality conditions, and  $\epsilon_{00}(\mathbf{q} \rightarrow 0, \omega = 0)$  scales like  $C/q^2$ . TODO: whether this is a numerical artifact or not To decide the constant  $C$ , very fine description of the Fermi surface is needed, so we need a very fine  $\mathbf{k}$ -grid. On the other hand, we don't really need many bands, because for the metal  $\epsilon_{00}(\mathbf{q} \rightarrow 0, \omega = 0)$ , most of the relevant transitions are inter-band ones.

#### 1.4.4.3 Frequency dependence of $\epsilon$

The  $\epsilon_{\mathbf{G}\mathbf{G}'}(\mathbf{q}, \omega)$  matrix has frequency dependence. The fastest way to handle this is to calculate the  $\omega = 0$  case only and then find the plasmon frequency, and then, the  $\epsilon_{\mathbf{G}\mathbf{G}'}(\mathbf{q}, \omega)$  curve can be fitted using sum rules. TODO

The spectral function – used in ARPES simulation – can't be accurately obtained just by the above mentioned method. Thus, **full frequency** treatments are needed. A trick called **static subspace** can speed up the process. TODO

#### 1.4.4.4 Console output

The console output of `epsilon` has the following structure:

1. Initialization
2. Iterating over the  $\mathbf{q}$ -grid. For each  $\mathbf{q}$ -point,
  - (a) The output starts with something like

```
=====
13:59:21   Dealing with q =  0.000000  0.500000  0.000000      5 / 35
=====
```

```
This is a regular non-zero q-point.
```

- (b) Then we can see lines about Rank of the polarizability matrix, BLACS processor grid, and Number of k-points in the irreducible BZ(q) (nrk).
- (c) Now we enter the first step mentioned in Section 1.4.4.1. The start line looks like

```
Started calculation of matrix elements with 324 transition(s) at
↪ 13:59:21.
```

- (d) TODO: what's the corresponding step of the following line:

```
Started building polarizability matrix with 320 processor(s) at
↪ 13:59:30.
```

#### 1.4.4.5 Other output files

The `epsilon` program generates the files mentioned in [this page](#). Note that `eps0mat.h5` is *not*  $\epsilon$  from DFT orbitals! In steps after `epsilon`, screening obtained from DFT orbitals are not used; we only use  $\epsilon$  from *GW* orbitals. The file `eps0mat.h5` is the  $\epsilon$  around the  $\Gamma$  point. Thus, in principle, we can calculate `eps0mat.h5` and `epsmat.h5` in two runs, and indeed this is the case when we deal with a metallic system (Section [3.2.4](#)).

#### 1.4.5 Systems of units

TODO



## Chapter 2

# Tight-binding models

TODO: from ab initio to models

### 2.1 Wannier functions and tight-binding models

For an insulator, we can construct exponentially decaying Wannier functions if and only if the Chern numbers are zero, and thus all insulators with inversion symmetry have exponentially decaying Wannier functions [5]

[12] introduces **maximally localized Wannier functions**. Although we usually say Wannier functions are Fourier transformations of Bloch functions, the fact that we have a gauge freedom in the global phase of each Bloch state – which may depend on  $\mathbf{k}$  – means Wannier functions are strongly non-unique. We want Wannier functions to be as localized as possible.

The standard of localization is the **localization functional**

$$\Omega = \sum_n (\langle \mathbf{r}^2 \rangle_n - \langle \mathbf{r} \rangle_n^2). \quad (2.1)$$

It's possible to use density overlap or Coulomb self-interaction or something else to decide how localized a wave function is. The gauge-dependent part of the functional is

$$\tilde{\Omega} = \sum_n \sum \text{TODO} \quad (2.2)$$

It contains a derivative  $\nabla_{\mathbf{k}}$ , and the usual numerical format is used.

Suppose we choose  $J$  bands (**frozen window**) and a set of smooth trial orbitals  $g_n(\mathbf{r})$ . We define

$$|\phi_{n\mathbf{k}}\rangle = \sum_{m=1}^J |\psi_{m\mathbf{k}}\rangle \langle \psi_{m\mathbf{k}} | g_n \rangle. \quad (2.3)$$

Now the arbitrary phase factors all cancel out. Picking up  $g_n$  requires chemical intuition: if, say, a band mainly consists of electrons from a  $\text{sp}^2$  chemical bond, then we should let  $g_n$  to be the  $\text{sp}^2$  orbitals.

### 2.2 The cRPA approach to obtain effective models

Suppose we already have identified the bands that cause strongly correlated effects and are now ready to write an effective theory for them. Below, I use the subscript  $d$  for the bands considered and  $r$  for the rest. Then we have [2]

$$W = (1 - W_r P_d)^{-1} W_r, \quad W_r = (1 - v P_r)^{-1} v, \quad (2.4)$$

where  $P_d$  is the polarization *within* the  $d$  subspace, and  $P_r$  is the polarization involving at least one state in the  $r$  space (and may also involve a state in  $d$  subspace). The exact form of  $P_d$  is hard to obtain, because the  $d$  bands are strongly correlated and therefore usual Feynman diagram resummation schemes fail for them, but  $r$  bands are dispersive and extended enough

for a diagrammatic resummation scheme like  $GW$  to work. Thus, the part concerning only  $r$  electrons in  $P_r$  can be reliably obtained using, say, RPA; the hopping between  $d$  bands and  $r$  bands is more tricky, TODO: why RPA for them is acceptable

## Chapter 3

# Carrying out calculations

List of technical issues:

- Band mode of BerkeleyGW
- Wannier “oscillation”
- 

### 3.1 Details in installation

#### 3.1.1 QuantumESPRESSO

What I write below is about QuantumESPRESSO 6.7Max; the names of compilation options and tags may change, but the overall idea is the same.

To enable OpenMP, the option `--enable-openmp` has to be added when we run `./configure`. MPI parallelization however can’t be obtained with adding this tag only: we need a MPI-wrapped compiler, as well as appropriately linked libraries like ScaLAPACK. The last time I tried this on Cori, the key point is to load module `impi` (i.e. IntelMPI) before running `./configure`. It seems following [this documentation page](#) complicates matters (but probably they will update this page, and it then becomes useful; anyway, what to be remembered is we need MPI compilers to get the fully parallelized version).

Whether an MPI compiler is present can be checked by looking at `make.inc`; alternatively, you may pay attention to whether the message `WARNING: parallel and serial compiler are the same` appears in the output of `make`.

### 3.2 Standard operation procedures

#### 3.2.1 Avoid data pollution

The follows are strongly recommended whenever modifying files:

- Read out loud the *directory* of the file being modified. Are you mistakenly modify files in another directory?
- When copying file from a folder to another, similarly read out loud the name of the source and the target.
- If a file is going to be overwritten, make sure it’s really no longer needed. Whenever you have doubts, make backups.
- 

#### 3.2.2 Finding the structure

##### 3.2.2.1 From open data

Sometimes the atomic positions are given with the help of Wyckoff positions: an example can be found in [\[11\]](#). In this case, the number of atomic positions given is *less* than the total numbers

of atoms in a primitive unit cell. The difference between the two can be found according to the multiplicity of the Wyckoff positions involved – in the case of [11], all atomic positions given are in the 2a position, the multiplicity of which is two, and 6 atomic positions are given, so we have  $2 \times 6 = 12$  atoms per primitive cell. But then the primitive cell mentioned in the experimental paper may be a primitive cell in the bulk while it's possible that we only want a layer in the bulk material (i.e. we are dealing with the monolayer version), and then we should only pick the atoms that are close enough to each other in the  $z$  direction.

### 3.2.2.2 Comparing existing structures for the same material

One good idea is to first find some high symmetry points (the inversion center, the reflection mirror, etc.) and then compare the relative relations of these positions in the two structures.

## 3.2.3 Insulator DFT+GW+BSE

### 3.2.3.1 The DFT stage

1. Do a `scf` calculation in `1-scf`.
2. Do a `bands` calculation in `2.1-wfn`. This step includes:
  - (a) Create a `the-suffix-you-set.save` folder in `2.1-wfn`, and link `data-file-schema.xml` and `charge-density.dat` from `1-scf` into this folder. These are files required for a `bands` calculation.
  - (b) Run

```
data-file2kgrid.py --kgrid nx ny nz the-suffix-you-set.save/
↪ data-file-schema.xml kgrid.inp
```

to create `kgrid.inp`, which describes how to create a  $k$ -grid with size `nx ny nz`. This can't be done with options in QuantumESPRESSO's `KPOINTS` section because QuantumESPRESSO and BerkeleyGW have different tolerance for symmetry.

- (c) Run

```
kgrid.x kgrid.inp kgrid.out kgrid.log
```

to obtain `kgrid.out`. The content in `kgrid.out` will be used as the `KPOINTS` section for the input file of `pw`.

- (d) Preparing the `bands.in` file, which is the input file of `pw.x`. Do the following checklist:
  - Whether calculation is `bands`.
  - Whether `pseudo_dir` is correct.
  - Whether `nbnd` is set to, say, 1000.
  - Whether `lspinorb = .true.` and `noncolin = .true.` are set for an SOC run.
- (e) Run `pw2bgw.x` in `2.1-wfn`. Do the following checklist:
  - This step should be done with *exactly the same* parallelization setting with `pw.x`.
  - The `wfng_nk1`, `wfng_nk2`, `wfng_nk3` parameters should be set to `nx`, `ny`, `nz` mentioned above. (This item needs double check especially when `pw2bgw.inp` comes from another run.)
  - Whether `rhog_flag` is `.true..`
  - Whether `vxc_flag` is `.true..`
  - Whether `wfng_flag` is `.true..`

3. Do a `bands` calculation in `2.2-wfnq`. The steps are similar to `2.1-wfn`:

- (a) Linking files from `1-scf`.
- (b) Run

```
data-file2kgrid.py --kgrid nx ny nz --qshift qx qy qz the-
↪ suffix-you-set.save/data-file-schema.xml kgrid.inp
```

to get the `kgrid.inp` file. Here `qx qy qz` is a small displacement used to regularize Coulomb interaction at  $\mathbf{q} = 0$ . A common choice is `0 0 0.001`; when dealing with a 2D material, choose `0 0.001 0`, because with `cell_slab_truncation` open in the `epsilon.x` step, non-zero  $z$  components of  $k$ -points are forbidden.

- (c) Run `kgrid.x`.
- (d) Preparing the `bands.in` file.
- (e) Run `pw2bgw.x`. Do the following checklist:
  - This step should be done with *exactly the same* parallelization setting with `pw.x`.
  - The `wfng_nk1`, `wfng_nk2`, `wfng_nk3` parameters should be set to `nx`, `ny`, `nz` mentioned above.
  - The `wfng_dk1`, `wfng_dk2`, `wfng_dk3` parameters should be set to `wfng_nk1 × qx`, etc. (If `kshift` is used, it also should be added to `wfng_dk1`, etc.)

Note that this doesn't mean the displacement imposed to the  $k$ -grid is `wfng_nk1 × qx`: the displacement is still `qx qy qz`. Here the `wfng_dk1`, `wfng_dk2`, `wfng_dk3` are conventional parameters used in Monkhorst-Pack grids, and `wfng_dk1 = 0.5` means the grid is shifted towards  $x$  direction by half a *grid step* – and therefore the displacement is  $0.5 \times 1 / \text{wfng\_nk1}$  in the crystal coordinates. Now we understand why we need to set `wfng_dk1` to `wfng_nk1 × qx`. Indeed, below is a part of the header of a WFN file in 2.2-wfnq:

```
k-grid:    24  24   1
k-shifts:    0.000000    0.024000    0.000000
[ifmin = lowest occupied band, ifmax = highest occupied band, for
  ↪ each spin]
      Index      Coordinates (crystal)      Weight    Number of
      ↪ G-vectors      ifmin      ifmax
      1      0.000000    0.001000    0.000000    0.001736
      ↪                               36275      1      120
```

It can be seen that the first  $k$ -point is displaced 0.001 in the  $y$  direction, and the `k-shifts` parameter corresponding to the  $y$  direction is 0.024; since the size of the grid in the  $y$  direction is 24, the displacement instructed by the latter is  $0.024/24 = 0.001$ , exactly the displacement recorded in the first  $k$ -point.

### 3.2.3.2 The GW stage

1. Do a `epsilon` calculation in 1-`epsilon`. The steps are listed below:

- (a) Linking files. Come to 1-`epsilon` and do the follows:

```
ln -sf ../2.1-wfn/WFN
ln -sf ../2.2-wfnq/WFN ./WFNq
```

- (b) Prepare `epsilon.inp`. Do the follow checklist:

- Whether we are setting `qpoints` instead of `kpoints`.
- Whether there is an `end` line of the `qpoints` block.
- Whether each line of the `qpoints` block is in the format (see [here](#))

```
qx qy qz 1 is_q0
```

- Especially, whether the line corresponding to the  $\Gamma$  point has `is-q0 = 1`.
- If we are dealing with a 2D material, add `cell_slab_truncation`.
- Set `epsilon_cutoff` to, say, 10; the exact value is to be decided by convergence tests.
- Set `number_bands` to the highest *total* number of bands allowed by `degeneracy_check.x` (Section 4.5.1).

2. Do a `sigma` calculation in 2-`sigma`. The steps are listed below:

- (a) Link necessary files:

```
ln -sf ../2.1-wfn/vxc.dat
ln -sf ../2.1-wfn/RH0
ln -sf ../2.1-wfn/WFN ./WFN_inner
ln -sf ../1-epsilon/epsmat.h5
ln -sf ../1-epsilon/eps0mat.h5
```

- (b) Prepare `sigma.inp`. Do the following checklist:

- Whether we are setting `kpoints` instead of `qpoints`. (This time it's not `qpoints`!)
- Whether there is an `end` line of the `kpoints` block.

- Whether each line of the `kpoints` block is in the format

```
kx ky kz 1
```

- If we are dealing with a 2D material, add `cell_slab_truncation`.
- Set `number_bands` to the same value in `epsilon.inp`.
- Set `band_index_min` and `band_index_max`. The bands between the two are corrected by (1.66), and others are not.

### 3.2.3.3 The BSE stage

1. Do a `kernel` calculation in `3-bse`. The steps are listed below:

- (a) Link necessary files.

```
ln -sf ../1-epsilon/WFN ../WFN_co
ln -sf ../1-epsilon/epsmat.h5
ln -sf ../1-epsilon/eps0mat.h5
```

- (b) Prepare `kernel.inp`. Do the following checklist:

- Whether the following lines are there:

```
use_symmetries_coarse_grid
```

- Whether `number_val_bands` and `number_cond_bands` are specified.
- If we are dealing with a 2D material, whether `cell_slab_truncation` is added.

2. Do an absorption step.

- (a) Build a denser  $\mathbf{k}$ -grid and repeat 2.1-wfn and 2.2-wfnq steps.

- (b) Link necessary files.

```
ln -sf ../2.1-wfn/WFN ../WFN_co
ln -sf ../2.2-wfnq/WFN ../WFNq_co
ln -sf ../2.1-wfn-dense/WFN ../WFN_fi
ln -sf ../2.2-wfnq-dense/WFN ../WFNq_fi
ln -sf ../1-epsilon/epsmat.h5 ./
ln -sf ../1-epsilon/eps0mat.h5 ./
ln -sf ../3-kernel/bsema.h5 ./
```

The `bsema` and `msexmat` files are missing if HDF5 format is used to generate the output files – the documentation of BerkeleyGW says we need the files but it’s wrong: When `bsema.h5` is present, the program runs well.

- (c) Prepare `absorption.inp`. Do the following checklist:

- Whether the following lines are in `absorption.inp`:

```
use_symmetries_fine_grid
use_symmetries_coarse_grid
```

- Whether

```
number_cond_bands_coarse
number_cond_bands_fine
number_val_bands_coarse
number_val_bands_fine
```

are specified.

- 

### 3.2.4 Metal DFT+GW+BSE

For metals, the  $\mathbf{q}$ -displacement technique can no longer be used. The working procedure now is

1. Do a `scf` calculation in `1-scf`.
2. Do a `bands` calculation in `2.1-wfn`.
3. Do `2.2-wfn0` with a *finer*  $\mathbf{k}$ -grid, still *without* any `qshift` displacement.
4. Do a `epsilon` calculation in `1-epsilon`. The steps are listed below:
  - (a) Link files according to

```
ln -sf ../2.1-wfn/WFN
ln -sf ../2.1-wfn/WFN ./WFNq
```

The 2.2-wfnq step is not needed, because we are not going to deal with the  $\Gamma$  point in this step.

(b) Preparing `epsilon.inp`. Do the following checklist:

- Whether we are setting `eqpoints` instead of `kpoints`.
- Make sure the  $\Gamma$  point is *not* included in the `qpoints` block.
- Whether each line of the `qpoints` block is in the format

```
qx qy qz 1 0
```

5. Do a `epsilon` calculation in 1.2-epsilon0.

(a) Link files according to

```
ln -sf ../2.2-wfn0/WFN
ln -sf ../2.2-wfn0/WFN ./WFNq
```

Now the outputs of the 2.1-wfn step is not used.

(b) Prepare `epsilon.inp`. Do the following checklist:

- Whether we are setting `eqpoints` instead of `kpoints`.
- The *only point* included in the `qpoints` block should be the non-zero  $k$ -point with smallest length in the  $k$ -grid used in 2.2-wfn0.
- Whether the  $\Gamma$  point is in the format

```
qx qy qz 1 2
```

## 3.2.5 Band plot

### 3.2.5.1 DFT level: $k$ -path

To plot the bands, just do a `bands` calculation where the `K_POINTS` block is in `crystal_b` mode.

The results can be processed by the `bands.x` utility. An example of the input file:

```
&BANDS
prefix = 'WTe2'
outdir = './'
filband = 'WTe2_bands.dat'
lsigma(1) = .true.
lsigma(2) = .true.
lsigma(3) = .true.
lsym = .false.
/
```

Here the `lsigma` options are only available for a full SOC calculation. A script used to read the output file can be found [here](#).

Note that when plotting the band structure, the Fermi energy is to be found in `scf.out` or in a `nscf` run. It won't appear in a `bands` run.

### 3.2.5.2 GW level: `inteqp`

The GW level bands can be obtained by the `inteqp` program. Note that this shouldn't be used for a system considered metallic by DFT (Section 4.8.4). The standard operation procedure is listed here:

1. Go to `4-path` – the folder responsible for the DFT level calculation – and perform a `pw2bgw` run needed to create a `WFN` file. Note that if we are not dealing with a  $k$ -grid, `wfng_nk1`, `wfng_nk2`, and `wfng_nk3` should be skipped.
2. Create a folder within `1-epsilon/`, and go into it.
3. Link the necessary files:

```
ln -sf ../../2.1-wfn/WFN ./WFN_co
ln -sf ../../4-path/WFN ./WFN_fi
cp ../../1-epsilon/eqp1.dat ./eqp_co.dat
```

4.

### 3.2.5.3 GW level: using WFN\_outer

It's possible to do the same thing in Section 3.2.5.1 with BerkeleyGW: we can

### 3.2.5.4 BSE level

## 3.2.6 Wannier functions and tight-binding models

Tight-binding models are a more compact way to show the band structure.

### 3.2.6.1 wannier90 for DFT

1. The first step is to perform a fresh QuantumESPRESSO run. **wannier90** requires a full Monkhorst–Pack grid, and to create a **bands.in** file, (don't forget to add the **utility** folder to the **PATH** environment variable) just prepare a **bands.in** file without the **K-POINTS** block and run

```
kmesh.pl 20 20 1 >> bands.in
```

Replace 20 20 1 by the **k**-grid size you want. The number of bands in this DFT run *shouldn't* be too large, or otherwise **pw2wannier90** will be extremely slow.

2. Prepare a **prefix.win** file, where **prefix** is the QuantumESPRESSO prefix used. Examples of this input file can be found by searching “wannier90 2007 workshop” and downloading the tutorial files coming together with the web page about this event – currently it's [here](#). Do the following checklist:
  - Is **num\_bands** plus the number of excluded bands the same as **nbnd** in the QuantumESPRESSO run (Section 4.6.1)? If you don't want all valence bands in the DFT run, always decide how many bands to include *before* doing any **wannier90.x** run; changing parameters in **prefix.win** after a **wannier90.x** run may lead to inconsistencies.

3. Run

```
wannier90.x -pp prefix.win
```

to get **prefix.nnkp**.

4. Run **pw2wannier90.x**. The input file looks like

```
&inputpp
outdir      = './'
prefix      = 'WTe2'
seedname    = 'WTe2'
scdm_proj   = .true.
write_amn   = .true.
write_mmn   = .true.
write_unk   = .true.
/
```

This step should finish in several minutes. If it takes too long, reduce **nbnd** in the QuantumESPRESSO step and redo the whole procedure.

5. Run

```
wannier90.x prefix.win
```

### 3.2.6.2 GW level: sig2wan

The **sig2wan** code can extract a Wannier interpolated band structure from output files of the **sigma** step. An example can be found [here](#). The routine is almost the same as **wannier90** for DFT, except for one thing: after running **pw2wannier90.x**, we need to replace the **perfix.eig** file generated by **pw2wannier90.x** by the **prefix.eig** file generated by **sig2wan**.

## 3.2.7 Self-consistent GW

Self-consistent schemes in *GW* are listed in Section 1.3.3. In this section, I talk about how to do them.



### 3.2.7.1 Energy self-consistent calculation in *GW*

The `epsilon` code has an option called `eqp_corrections`, which takes the

### 3.2.7.2 Eigenstate self-consistent calculation in *GW*

Eigenstate self-consistent *GW* – in other words, diagonalizing  $\Sigma$  instead of taking into account only the diagonal elements – is realized by `scGWtool.py`. The procedure is listed below:

1. Perform a DFT run, with *non-diagonal* VXC in the `pw2bgw` step. An example of the part of `pw2bgw.in` concerning VXC:

```
vxcg_flag = .true.  
vxc_diag_nmax = 1000  
vxc_offdiag_nmax = 1000
```

Note that here we are using VXC instead of `vxc.dat` as the output format of  $V_{xc}$ . Whenever weird errors occur with the `pw2bgw` step, redo the `bands` step and do `pw2bgw` in exactly the same parallelization environment.

2. Run `sigma` with the options

```
sigma_matrix -1 0  
dont_use_vxc.dat
```

and rename the output file `sigma_hp.log` to `sigma_hp_col.log`.

3. Run `sigma` with the options

```
sigma_matrix -2 0  
dont_use_vxc.dat
```

and rename the output file `sigma_hp.log` to `sigma_hp_row.log`. TODO: understand what's going on in this step; the explanation involving “the lower triangle and the upper triangle” doesn't seem to agree with the official documentation

4. Rename (or link, or whatever) the `WFN_inner.h5` file used in the last steps as `wfn_old.h5`. If the last two steps are not done with the HDF5 format (i.e. they are done with the Fortran binary format), a `wfn2hdf.x` run is needed.
5. Run `scGWtools.py` to build and diagonalize the quasiparticle Hamiltonian. The output files are `vxc_new.dat` and `wfn_new.h5`.
6. Rename the output files into `vxc.dat` and `WFN.h5`. Run `sigma` with `use_wfn_hdf5` and remove the `dont_use_vxc.dat` option.

### 3.2.8 Topological invariants with `z2pack`

The error `ValueError: The given WCC are not degenerate Kramers pairs at the edges of the surface` occurs because one topological band and a trivial band appear in the `bands` option of `z2pack.tb.System`.

### 3.2.9 Band projection

## 3.3 Performance tricks

### 3.3.1 Parallelization

TODO: Do more MPI tasks already result in faster speed?

### 3.3.2 Choosing cutoff energies wisely

The cutoff energies, especially the one in the *GW* step, of course should be large enough, but not too large: if in a benchmark test, a smaller cutoff energy gives almost the same result compared to a higher cutoff energy, then the smaller cutoff energy should be used unless we have reasons against this practice.

### 3.3.3 pseudobands

```
wfn2hdf.x BIN WFN WFN.h5
pseudobands.py WFN.h5 WFN0.h5 0.7 0.02
hdf2wfn.x BIN WFN0.h5 WFN0
```

Using `pseudobands` breaks the norm conserving condition. Therefore, in `epsilon.inp`, we need to add `dont_check_norms`.

TODO: how it works; maybe [6] may provide some hints.

## 3.4 Convergence tests

Here, the term **convergence test** means to make sure the parameters that control the size of the problem are large enough. The parameters included are:

- The  $\mathbf{k}$ - (and therefore  $\mathbf{q}$ -)grid this is defined in the 2.1-`wfn` step;
- The cutoff energy in DFT;
- The number of empty bands given to `epsilon`;
- The cutoff energy in `epsilon`;
- TODO: what's number in CH summation? Possibly just `number_bands`.

TODO: problem: do high bands require more  $\mathbf{G}$  vectors, or do low bands require more  $\mathbf{G}$  vectors?

List of theoretical problems:

- Wavefunction cutoff only affects bare exchange significantly, can be treated separately; large  $\mathbf{G}$ 's only contribute to bare exchange.
- 

More theoretical reading is needed to understand <https://www.nersc.gov/assets/Uploads/ConvergenceinBGW.pdf>

# Chapter 4

## Trouble shooting

### 4.1 Unexpected units

#### 4.1.1 Band energy output of `pw.x`

When invoked in the `bands` or `nscf` modes, `pw.x` gives a list of  $\mathbf{k}$ -points and the corresponding band energies. The unit of the  $\mathbf{k}$  points is the momentum space version of `alat`, which is *not* based on the crystal coordinates.

### 4.2 Trouble shooting in MPI

#### 4.2.1 `srun: fatal: Can not execute`

This error may come from an error in compilation, but it can also occur because the directory to the program is misspelt, or an environment variable involved in the directory is not defined, etc.

#### 4.2.2 `error parsing parameters`

This sometimes occurs because you misspell the name of a program to be `mpiruned`.

#### 4.2.3 Each process is run serially and doesn't communicate with others

A possibility is when compiling the program to be launched in parallel, you choose the serial version (Section 3.1.1).

#### 4.2.4 `nsufficient virtual memor`

Try to reduce the number of MPI processes per node and increase the number of OpenMP threads per MPI process.

### 4.3 Trouble shooting in Python

#### 4.3.1 `AttributeError: 'Dataset' object has no attribute 'value'`

This error comes from an update of `hdf5` which deprecated the `.value` attribute. To solve the problem, either downgrade the library or change the statements using `.value`. This error is known in `scGWtool.py`.

## 4.4 Trouble shooting in QuantumEspresso

**4.4.1** Intel MKL FATAL ERROR: Cannot load symbol MKLMPI\_Get\_wrappers.

### 4.4.2 Program frozen

Check whether too much resource is given to a simple task.

**4.4.3** Error in routine `allocate_fft (1): wrong ngms`

I'm still not quite sure what causes this error, but it seems to be related to parallelization: in a run with 2240 MPI tasks, the error occurred, but when I used 320 MPI tasks, the error disappeared. The error can occur with `pw.x` or `bands.x`.

**4.4.4** Error reading attribute index : expected integer , found \*

This error occurs when we use a pseudopotential that is obtained by converting another pseudopotential in a different format (see [here](#)). Usually we don't need to "correct" it.

**4.4.5** `cdiaghg (159): eigenvectors failed to converge`

Usually by changing `diagonalization` to `cg`, this can be solved; `cg` is more stable but much slower.

**4.4.6** Error in routine `cdiaghg (1052): problems computing cholesky`

This also seems to be a convergence problem that can be solved by changing `diagonalization` to `cg`.

**4.4.7** Error in routine `set_occupations (1): smearing requires a vaklue for gaussian broadening (degauss)`

This happens whenever smearing is used but the smearing parameter is not set. Note that this is *not* restricted to Gaussian smearing: all smearing schemes are controlled by the `degauss` parameter, and when this parameter is not set, the error occurs.

**4.4.8** Error in routine `splitwf (36197): wrong size for pwt`

Usually this occurs when `pw2bgw` is redone (`pw2bgw` deletes intermediate files, making another `pw2bgw` run impossible). This also appears in cases similar to Section 4.4.3. A complete `bands-pw2bgw` run has to be redone.

**4.4.9** Error in routine `PW2BGW(19):input pw2bow`

Usually this occurs when something else happens between a `bands` run and a `pw2bgw` run for it. A complete `bands-pw2bgw` run has to be redone.

**4.4.10** Error in routine `PW2BGW (19): input_pw2bgw`

This can occur when you misspell one option in `pw2bgw.in`.

**4.4.11** `stress for hybrid functionals not available with pools`

As the error message implies, turning `tstress` to `.false.` (or simply deleting anything about this option) solves the problem.

#### 4.4.12 Error in routine projwave (1): Cannot project on zero atomic wavefunctions!

This occurs when running `projwfc.x`. It occurs when atomic wave functions are not shipped together with the pseudopotentials used. This is the case for some types of pseudopotentials, including ONCV TODO: full list.

## 4.5 Trouble shooting in epsilon and sigma

### 4.5.1 Selected number of bands breaks degenerate subspace.

Run `degeneracy_check.x` WFN to see degeneracy-allowed number of bands. This error occurs when one band in a degeneracy subspace is considered but others are not. Also, the `band_index_min` and `band_index_max` parameters shouldn't be too close to `vxc_diag_min` and `vxc_diag_max`, or the error occurs.

### 4.5.2 WFN ifmin/ifmax fields are inconsistent

The full message is

```
WFN ifmin/ifmax fields are inconsistent:
- there is a valence state above the middle energy
- there is a conduction state below the middle energy
Possible causes are:
(1) Your k-point sampling is too coarse and cannot resolve the Fermi energy.
    Try to carefully inspect your mean-field energies, and consider using a
    ↪ finer
k-grid.
(2) You are using eqp.dat and the QP corrections change the character of some
    ↪ s
tates
    from valence<->conduction. In this case, you should use another mean-field
    ↪ the
ory
    that gives the same ground state as your GW calculation.
(3) You are running integp, but you are either shifting the Fermi energy or
    ↪ usi
ng
    restricted transformation.
```

The direct causes of this error are already listed above. But it takes some time to see what is the deeper reason, and how to solve it:

- Sometimes when the `occupation` option in the 2.1-wfn and 2.2-wfnq steps is not correct. If `fixed` is used for a metal, for example, some positions that should be a part of a hole Fermi pocket are occupied by electrons, and therefore the highest occupied state has higher energy than the lowest unoccupied state. This usually means the smearing scheme needs to be changed.
- When you shift the occupation in a WFN file (Section 4.8.7), but forget to change the `ifmax` dataset correspondingly, this error will also occur, regardless of how you change the `eqp.dat`.

### 4.5.3 Segmentation fault: address not mapped to object at address

The root of this error differs from case to case.

If we see

```
q-pt      2: Head of Epsilon      =   NaN      NaN
q-pt      2: Epsilon(2,2)         =   NaN      NaN
```

usually this means a “divided-by-zero” error occurs. This may occur when we incorrectly use the insulator procedure to calculate a metallic system (as long as DFT thinks the system is metallic, the error has the potential to occur, regardless of whether the system is metallic after GW correction), often regardless of the smearing type.

#### 4.5.4 eqpcor mean-field energy mismatch

This error happens when we try to do an eigenvalue self-consistent calculation, and `epsilon` finds the DFT energies given in `eqp.dat` are different from the energies in `WFn`. This sometimes is a technical problem (the Rydberg energy definitions used in QuantumESPRESSO and BerkeleyGW are slightly different), and can be solved by increasing `TOL_eqp` in the source code of BerkeleyGW. The error may also be reported when the DFT energies in `eqp.dat` are mistakenly changed (we should only change the column corresponding to the corrected energy).

#### 4.5.5 ERROR: occupations (ifmax field) inconsistent between WFn and WFnq files.

```
ERROR: occupations (ifmax) inconsistent between WFn and WFnq files.  
Remember that you should NOT use WFnq for metals and graphene.
```

#### 4.5.6 ERROR: Unexpected characters were found while reading the value for the keyword

This usually happens when the input file contains a line like

```
number_bands = 148
```

while the correct format is

```
number_bands 148
```

#### 4.5.7 forrtl: severe (24): end-of-file during read, unit -5, file Internal List-Directed Read

This usually happens when we should write `qp`points but actually write `kp`points (or the opposite).

#### 4.5.8 ERROR: Inconsistent screening, truncation, or q0 vector

The full error message may be

```
ERROR: the input truncation flag indicates that the Coloumb interaction v(q0)  
diverges for q0->0. However, you have q0 exactly zero.  
You should always specify a *nonzero* q0->0 vector unless you have 0D  
truncation, i.e., spherical or box truncation.
```

```
ERROR: Inconsistent screening, truncation, or q0 vector
```

This arises when we are dealing with a metal but forget to add `screening_metal` to `sigma.inp`.

```
ERROR: cannot use metallic screening with q0 = 0.  
You should either specify a nonzero q0->0 vector or use another screening  
→ flag.
```

#### 4.5.9 cannot use metallic screening with q=0

As the name implies, when doing a metallic calculation, we need to make sure that the  $\mathbf{q} \rightarrow 0$  point is the smallest *non-zero* point in the dense  $\mathbf{k}$ -grid (Section 3.2.4).

#### 4.5.10 ERROR: genwf mpi: No match for rkq point

#### 4.5.11 ERROR: Missing bands in file eqp\_co.dat

This happens when doing `inteqp`. The origin of the error is just its name implies. Note that this error sometimes occurs for unnoticed reasons:

- When the pseudopotential is changed, the number of bands may change, because the orbitals included in the pseudopotentials change. Now if `band_index_max` and `band_index_min` are set according to the old pseudopotentials, and the Fermi energy happens to fall outside of the range between the two, the error occurs.

#### 4.5.12 forrtl: severe (71): integer divide by zero

If this happens after `Calculation parameters:`, it may arise from mistakenly setting `band_index_max` to a value smaller than `band_index_min`. In this case, the number of bands considered will be zero, and thus a divide-by-zero error occurs when `sigma.x` tries to distribute computational loads.

#### 4.5.13 ERROR: screened Coulomb cutoff is bigger than epsilon cutoff

This happens in the `sigma` step. When the cutoff energies of `epsmat.h5` and `eps0mat.h5` are different (this may happen when you want to reuse existing `eps0mat.h5` for a metallic calculation), this may occur.

### 4.6 Trouble shooting in wannier90 and pw2wannier90

#### 4.6.1 w90\_wannier90\_readwrite\_read: mismatch in WTe2.eig

When reading a `.eig` file, `wannier90` doesn't really look at the band index and  $k$ -point index. Rather, it completely relies on `num_bands` and `mp_grid`. Thus, if the number of `nbnd` in the QuantumESPRESSO run – which decides the number of bands in the `.eig` file – isn't `num_bands`, `wannier90` finds bands in the `.eig` file messed up, and the error occurs.

#### 4.6.2 WTe2.amn has not the right number of bands

Sometimes the `.amn` file only contains `num_wann` bands, while the `.eig` file contains `num_bands` bands, and when the two values are different, the error occurs. Redoing the whole procedure with a fixed `prefix.win` file can solve this problem. TODO: what's really going on here

#### 4.6.3 forrtl: severe (174): SIGSEGV, segmentation fault occurred

TODO: occurs when `num_bands > num_wann`

### 4.7 Trouble shooting in kernel and absorption

#### 4.7.1 ERROR: Inconsistent symmetry treatment of the fine and shifted grids with the momentum operator

### 4.8 Checklist for unexpected results

Sometimes the calculation ends successfully, but the result seems strange. Below are some checklists.

#### 4.8.1 Band symmetry higher than the space group shown at the beginning of `bands.out`

- Usually this is because of an approximate symmetry, which is ignored by QuantumESPRESSO because its tolerance is very low.
- Are all steps in the DFT calculation using the same crystal structure?

#### 4.8.2 Band structure looks very far from the literature

- Check the crystal structure: if it comes from relaxation, does it converges?
- For 2D materials, when we change the vacuum distance and use crystal coordinates for atomic positions at the same time, always double check whether we scale the atomic positions correctly. The formula is

$$\text{new } z \text{ coordinate} = \frac{\text{old vacuum distance}}{\text{new vacuum distance}} \times \text{old } z \text{ coordinate.} \quad (4.1)$$

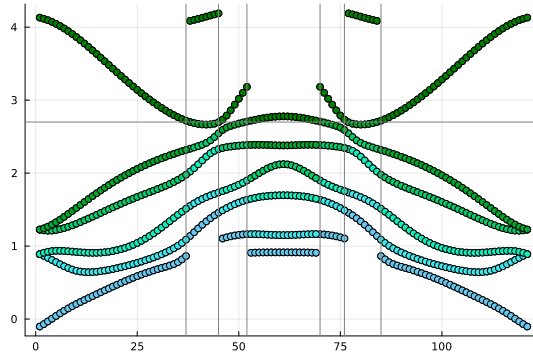


Figure 4.1: Example of how `inteqp` misidentifies DFT-level bands: the data is from the DFT column of `eqp.dat`; the colors of points are about the band index.

- Are all steps in the DFT calculation using the same crystal structure?
- Is the Fermi energy correct? Sometimes we change the band structure but forget to change the Fermi energy used to plot bands.

#### 4.8.3 Band plot is empty

- Are there enough bands? If `nbnd` is not set for an insulator, no conduction band will be considered.
- Is the Fermi energy correct? If the Fermi energy is set too high (which may come from, say, wrong unit), then naturally there is no band in the plot.

#### 4.8.4 Band plot is not continuous

TODO: it's  $k$ -points that are shuffled, or the band indices, or both???

If the system is metallic in the DFT level (regardless of what *GW* says about the material, and the screening model used) and `inteqp` is used, this is expected: somehow, `inteqp` assumes the system is an insulator, and thus states above the Fermi energy (but not too far from it) has to be in one band. Thus, if, say, the 120th band has more than one intersection points with the Fermi energy level, its part below  $\varphi_F$  will be recognized as the 119th band in the eyes of `inteqp`. This can be seen by plotting the DFT column of the `eqp.dat` output of `inteqp` (Figure 4.1).

#### 4.8.5 The size of band gap

DFT is infamous for underestimating the band gap.

- Use hybrid functionals like HSE.
- Let *GW* correct the band structure. TODO: but how? How to avoid the error in Section 4.5.2?

#### 4.8.6 SOC effects are too strong

When SOC effects are much stronger than expected:

- Are you using relativistic pseudopotentials for a non-SOC run? This is *not* correct (and unfortunately QuantumESPRESSO never tells us when this happens).
- TODO:

#### 4.8.7 When we get a semimetal in the DFT step but it should be an insulator

This is similar to Section 4.8.5.



Note that naively feeding the semimetal result into BerkeleyGW while still using the insulator procedure may result in errors in Section 4.5.3.

One way to solve the problem is to manually move the conduction bands and the valence bands away from each other. Naively using the `eqp_correction` option and shifting bands near the Fermi surface away from each other leads to Section 4.5.2, precisely because “QP corrections” (i.e. the energy shift manually added by me) change the character of some states from valence to conduction. The `mf_header/kpoints/occ` and `mf_header/kpoints/ifmax` datasets in the `WFN.h5` file have to be modified accordingly. A procedure to do so can be found [here](#). TODO: should this be done in 2-sigma?

The problem with this method is we don’t have `eqp_correction` in `kernel`. TODO

# Bibliography

- [1] Irene Aguilera, Christoph Friedrich, Gustav Bihlmayer, and Stefan Blügel. G w study of topological insulators bi 2 se 3, bi 2 te 3, and sb 2 te 3: Beyond the perturbative one-shot approach. *Physical Review B*, 88(4):045206, 2013.
- [2] F Aryasetiawan, T Miyake, and R Sakuma. 7 the constrained rpa method for calculating the hubbard u from first-principles. *The LDA+ DMFT approach to strongly correlated materials*, 2011.
- [3] Bradford A Barker, Jack Deslippe, Johannes Lischner, Manish Jain, Oleg V Yazyev, David A Strubbe, and Steven G Louie. Spinor g w/bethe-salpeter calculations in berkeleygw: Implementation, symmetries, benchmarking, and performance. *Physical Review B*, 106(11):115127, 2022.
- [4] J Arjan Berger, Pierre-François Loos, and Pina Romaniello. Potential energy surfaces without unphysical discontinuities: The coulomb hole plus screened exchange approach. *Journal of Chemical Theory and Computation*, 17(1):191–200, 2020.
- [5] Christian Brouder, Gianluca Panati, Matteo Calandra, Christophe Mourougane, and Nicola Marzari. Exponential localization of wannier functions in insulators. *Physical review letters*, 98(4):046402, 2007.
- [6] Mauro Del Ben, H Felipe, Andrew Canning, Nathan Wichmann, Karthik Raman, Ruchira Sasanka, Chao Yang, Steven G Louie, and Jack Deslippe. Large-scale gw calculations on pre-exascale hpc systems. *Computer Physics Communications*, 235:187–195, 2019.
- [7] Jack Deslippe, Georgy Samsonidze, David A. Strubbe, Manish Jain, Marvin L. Cohen, and Steven G. Louie. Berkeleygw: A massively parallel computer package for the calculation of the quasiparticle and optical properties of materials and nanostructures. *Computer Physics Communications*, 183(6):1269–1289, 2012.
- [8] Sergey V Faleev, Mark Van Schilfgaarde, and Takao Kotani. All-electron self-consistent g w approximation: Application to si, mno, and nio. *Physical review letters*, 93(12):126406, 2004.
- [9] Lars Hedin. New method for calculating the one-particle green’s function with application to the electron-gas problem. *Physical Review*, 139(3A):A796, 1965.
- [10] Mark S Hybertsen and Steven G Louie. Electron correlation in semiconductors and insulators: Band gaps and quasiparticle energies. *Physical Review B*, 34(8):5390, 1986.
- [11] Arthur Mar, Stephane Jobic, and James A Ibers. Metal-metal vs tellurium-tellurium bonding in wte2 and its ternary variants tairte4 and nbirte4. *Journal of the American Chemical Society*, 114(23):8963–8971, 1992.
- [12] Nicola Marzari, Arash A Mostofi, Jonathan R Yates, Ivo Souza, and David Vanderbilt. Maximally localized wannier functions: Theory and applications. *Reviews of Modern Physics*, 84(4):1419, 2012.
- [13] John P Perdew, Robert G Parr, Mel Levy, and Jose L Balduz Jr. Density-functional theory for fractional particle number: derivative discontinuities of the energy. *Physical Review Letters*, 49(23):1691, 1982.

- [14] Bi-Ching Shih, Yu Xue, Peihong Zhang, Marvin L Cohen, and Steven G Louie. Quasiparticle band gap of zno: High accuracy from the conventional g 0 w 0 approach. *Physical review letters*, 105(14):146401, 2010.
- [15] M. Shishkin, M. Marsman, and G. Kresse. Accurate quasiparticle spectra from self-consistent gw calculations with vertex corrections. *Phys. Rev. Lett.*, 99:246403, Dec 2007.