CRPA EQS

Overall, the most 3 important quantities in cRPA calculation are : The complex density zrhokq $M_G^{n,n'}(k,q)$, the wannier projectors $P_{Ln}(k)$ and the potential kernel W_r(U), W(full interaction), V(bare Coul interaction)

The first quantities : **zrhokq** is calculated within the subroutine genzrhokq, which relys on original elk subroutines: **genzrho.f90** and **zftzf.f90**. See these routines and **genvchi0.f90** and **gwsefmk.f90** for the usuage

For the calculation steps: The wavefunction of $\mathbf{k}$ and $\mathbf{k}+\mathbf{q}$ is firstly calculated via subroutine **genwfsvp**,

The **zrhokq** is calculated by first calc complex dens in MT and IR reagion then transform to$\mathbf{G}+\mathbf{q}$ space

```
1  call genzrho(.true.,.false.,ngtc,wfmtkq(:,:,ispn,ist1),wfirkq(:,&
2  ispn,ist1), wfmtk(:,:,jspn,ist2),wfirk(:,jspn,ist2),zrhomt(:,:,&
3  ist2),zrhoir(:,ist2))
4  call zftzf(ngrf,jlgqr,ylmgq,ngvc,sfacgq,zrhomt(:,:,ist2)
5  &,zrhoir(:,ist2), zfgq)
```

Basically follow this equation:(similar to amadon paper A2)

$$M_G^{n,n'}(\mathbf{q},\mathbf{k}) = \langle \psi_{k+q,n} | \, e^{-i\cdot(\mathbf{q}+\mathbf{G})\cdot\mathbf{r}} \, | \psi_{k,n'} \rangle \tag{0.1}$$

In programe, the complex density $M_G^{n,n'}(k,q)$ (**zrhokq**) is calculated for all non-reduced k points(**nkptnr**) and reduced-q points. which means the number of total loops is $nkptnr \times nqpt$. And one more thing need to noticed is that the complex density is generated for all states as you can see from **line 23-30** below:

```
subroutine genzrhokq(iqp,vkpl,vqpl,zzrhokq)
! some initialization code here ....
call findqpt(vqpl,isym,iq)
!generate G+q related quantities
call gengqf(ngrf,vqc(:,iq),vgqc,gqc,jlgqr,ylmgq,sfacgq)
! k+q-vector in lattice coordinates
call findkpt(vkpl,isym,jk)
vkql(:)=vkpl(:)+vqpl(:)
call findkpt(vkql,isym,jkq)
! equivalent reduced k-points for k and k+q/not necessary,
! just for check
ik=ivkik(ivk(1,jk),ivk(2,jk),ivk(3,jk))
ikq=ivkik(ivk(1,jkq),ivk(2,jkq),ivk(3,jkq))
! index to all states/maybe set it to corrlated states
nst=nstsv
do ist1=1,nstsv
  idx(ist1)=ist1
end do
call genwfsvp(.false.,.false.,nst,idx,ngdgc,igfc,vkpl,ngpk,igpigk,wfmtk,ngtc, &
 wfirk)
! calculate the wavefunctions for all states of the input k+q-point
call genwfsvp(.false.,.false.,nst,idx,ngdgc,igfc,vkql,ngpkq,igpigkq,wfmtkq,ngtc, &
 wfirkq)
! loop over all states
zzrhokq(:,:,:,:)=0.d0
do ist1=1,nstsv
! calculate the matrix of <psi_{k+q,n,spin}|exp(-i(q+G)r))|psi_{k,n',spin'}>
! parralazation applied here.

  allocate(zrhomt(npcmtmax,natmtot,nstsv),zrhoir(ngtc,nstsv))
  allocate(zfgq(ngrf))
```

```
  do ist2=1,nstsv
    do ispn=1,nspinor
      do jspn=1,nspinor
        call genzrho(.true.,.false.,ngtc,wfmtkq(:,:,ispn,ist1),wfirkq(:,ispn,ist1), &
        wfmtk(:,:,jspn,ist2),wfirk(:,jspn,ist2),zrhomt(:,:,ist2),zrhoir(:,ist2))
        !matrix of <psi_{k+q,n,spin}|exp(i(q+G)r))|psi_{k,n',spin'}>,
        ! so take the conjg
        ! equation for M_G,n,n'
        call zftzf(ngrf,jlgqr,ylmgq,ngvc,sfacgq,zrhomt(:,:,ist2),zrhoir(:,ist2), &
        zfgq)
        zzrhokq(ist1,ist2,jspn,ispn,:)=conjg(zfgq(:))
      end do
    end do
  end do
!para end here
  deallocate(zrhomt,zrhoir,zfgq)
end do
```

Then projection procedure of the complex densities with the wannier projectors(similar to amadon equation A1-A2. the quantity of M_G^{mm}) :

$$M_G^{m,m'}(\mathbf{q}) = \frac{1}{N_k} \sum_{\substack{k\in nrk \\ n,n'\in corr}} M_G^{n,n'}(\mathbf{k},\mathbf{q}) P^{\dagger}_{m,n}(k+q) P_{m'n'}(k) \tag{0.2}$$

In subroutine *genwanrhokq*, it calculates the projection of the densities $k+q$ and k. The wannier projector is generated previously and read in when peroforming the projection.

The wannier projector is stored in $WANPRJ\_ELK.OUT$ as a binary file. (2022/04/27: now wanprj is generated on the fly)

The *zrhokq* generated previously for $k+q$ and k is directly passed into sub-

routine *genwanrhokq*, the wannier projector is generated for each k and $k+q$ vecs via:

```
subroutine genwanrhokq(ikp,iqp,vkpl,vqpl,zrhokq,mmm)
!init here and some vars omitted
nkst=projstwanprj(ik)
nkqst=projstwanprj(ikq)
idxk(1:nkst)=idxwanprj(1:nkst,ik)
idxkq(1:nkqst)=idxwanprj(1:nkqst,ikq)
call wanprojkpl(vkpl,idxk(1:nkst),ldwanprj,nprojwanprj,maxnstwanprj,nkst,&
  subulmwanprj,sublmwanprj,symlmwanprj,wanprjk)
call wanprojkpl(vkql,idxkq(1:nkqst),ldwanprj,nprojwanprj,maxnstwanprj,nkqst,&
  subulmwanprj,sublmwanprj,symlmwanprj,wanprjkq)
```

The the projection procedure in equation 0.2:

```
mmm=0.d0
! see initwanprj_globalvars.f90 and modwanprj for the definitions
! briefly: nkst,nkqst is the number states of k/k+q in correlated
!            subspace
!          idxk/idxkq is the index of these states
nkst=projstwanprj(ik)
! the eigenvalues should be exactly same as the equivalent reduced-k
nkqst=projstwanprj(ikq)
idxk(1:nkst)=idxwanprj(1:nkst,ik)
idxkq(1:nkqst)=idxwanprj(1:nkqst,ikq)
do ig=1,ngrf
  do iorb=1,norb
    is=orb(iorb,1)
    l=orb(iorb,2)
    lmmax=2*l+1
    i=l**2+1
    j=(l+1)**2
```

```
18      do ia=1,natoms(is)! for atom specifcation here.
19        ias=idxas(ia,is)
20        do ispn=1,nspinor
21          do jspn=1,nspinor
22            do ld1=1,lmmax
23              do ld2=1,lmmax
24                do ist1=1,nkqst
25                  istkq=idxkq(ist1)
26                  do ist2=1,nkst
27                    istk=idxk(ist2)
28                    mmm(ld1,ld2,ispn,jspn,ig)=mmm(ld1,ld2,ispn,jspn,ig)+&
29                    zrhokq(istkq,istk,ispn,jspn,ig)*&
30                    conjg(wanprjkq(ld1,ist1,ispn,iorb,ia))*&
31                    wanprjk(ld2,ist2,jspn,iorb,ia)
32                  end do!over ist2/nkst
33                end do!over ist1/nkqst
34              end do!over ld2/ldk
35            end do!over ld1/ldkq
36          end do!over jspn/spink
37        end do!over ispn/spinkq
38      end do!over ia
39    end do!over iorb
40  end do!over ig
```

The projected complex density $M_G^{m,m'}(q,k)$ then summed over non-reduced kpoints as described in eq 0.2

Then, we introduce the potential(v,W or U) into the projected densities $M_G^{m,m'}(q)$. The potential kernel V_{bare} or W or W_r is previously calculated(controled by block *potmode*). V_{bare}(potmode=0) and W(potmode=2) are original tasks in Elk, so

should be right. For V_{bare}, the potential is treated as diagnal matrix of G:

$$V_{bare}(G) = \frac{4\pi}{|q+G|^2} \tag{0.3a}$$

$$V(G,G^{'}) = \delta_{G,G^{'}} V_{bare}(G) \tag{0.3b}$$

While for full interaction W, the square root average of the bare intercation v is used :

$$v(G) = \frac{\sqrt{4\pi}}{|q+G|} \tag{0.4a}$$

$$\epsilon^{-1}(G,G^{'}) = \frac{1}{1 - v(G)P(G,G^{'})v(G^{'})} \tag{0.4b}$$

$$W(G,G^{'}) = v(G)\epsilon^{-1}(G,G^{'})v(G^{'}) \tag{0.4c}$$

And the potential will then been projected onto wannier space via the quantity of MMM generated in subroutine *genwanrhokq*: For V:

$$V_{m_1,m_2,m_3,m_4} = \sum_{q,G,G^{'}} V(G,G^{'}) M_G^{m_3,m_1\dagger}(\mathbf{q}) M_{G^{'}}^{m_2,m_4}(\mathbf{q}) \mathcal{W}(q) \tag{0.5a}$$

$$V_{m_1,m_2,m_3,m_4} = \sum_{q,G} M_G^{m_3,m_1\dagger}(\mathbf{q}) M_G^{m_2,m_4}(\mathbf{q}) \frac{4\pi}{|q+G|^2} \mathcal{W}(q) \tag{0.5b}$$

For full interaction W:

$$W_{m_1,m_2,m_3,m_4} = \sum_{q,G,G^{'}} W(G,G^{'}) M_G^{m_3,m_1\dagger}(\mathbf{q}) M_G^{m_2,m_4}(\mathbf{q}) \mathcal{W}(q) \tag{0.6a}$$

$$W_{m_1,m_2,m_3,m_4} = \sum_{q,G,G^{'}} M_G^{m_3,m_1\dagger}(\mathbf{q}) M_{G^{'}}^{m_2,m_4}(\mathbf{q}) \frac{4\pi\epsilon^{-1}_{G,G^{'}}}{|q+G||q+G^{'}|} \mathcal{W}(q) \tag{0.6b}$$

Where $\mathcal{W}(q)$ is the weight of the q-point. This is done by:

```
1 subroutine genumatq(iqp,vqpl,mmmq,umatq)
2 !init/ vars omitted
```

```
do ig=1,ngrf
z1=sqrt(gclgq(ig))
do jg=1,ngrf
  z2=sqrt(gclgq(jg))*z1
  do ld1=1,ldwanprj
    do ld2=1,ldwanprj
      do ld3=1,ldwanprj
        do ld4=1,ldwanprj
          do ispn1=1,nspinor
            ild1=ld1+(ispn1-1)*(ldwanprj)
            do ispn2=1,nspinor
              ild2=ld2+(ispn2-1)*(ldwanprj)
              do ispn3=1,nspinor
                ild3=ld3+(ispn3-1)*(ldwanprj)
                do ispn4=1,nspinor
                  ild4=ld4+(ispn4-1)*(ldwanprj)
                  ! conjg(M_m1m3,',G(q)) * (M_m2m4,',G'(q)) * 1/|q+G||q+G'| *
                  ! epsiv_r(G,G',q,w=1:nw) = u_m1m2m3m4,'(q,w=1:nw)
                  z=&
                  conjg(mmmq(ld3,ld1,ispn3,ispn1,ig))*&
                  mmmq(ld2,ld4,ispn2,ispn4,jg)*&
                  epsi(ig,jg,1:nwrf)*z2
                  umatq(ild1,ild2,ild3,ild4,1:nwrf)=&
                  z+umatq(ild1,ild2,ild3,ild4,1:nwrf)
                end do
              end do
            end do
          end do!over ispn
        end do!overld4
      end do!over ld3
    end do!over ld2
  end do!overld1
end do!over jg
```

```
36 end do!over ig
```

And for spherical expression, the U_{full} and J_{full} is(ignore the spin) :

$$U_{m_1,m_2} = \frac{1}{(2l+1)^2} \sum_{m_1,m_2} W_{m_1,m_2,m_1,m_2} \tag{0.7a}$$

$$J_{m_1,m_2} = \frac{1}{(2l+1)(2l)} \sum_{m_1,m_2 \neq m_1} W_{m_1,m_2,m_2,m_1} \tag{0.7b}$$

$$\tag{0.7c}$$

$$\tag{0.8}$$

The Update(2022/04/27): Found that if the wannier projector is generated for each k+q vecs, then the summation of the off digaonal G seems to be right(at least at same magnitude). That posibly means that the phase factor of the wavefuncs may play an interesting role when calculating the off digaonal quantities.(see wanprjkpl.f90 and initwanprj_globalvars.f90) So the wannier projector is now generated on the fly when calculating the projected complex densities. This change has been updated(2022/04/27). What to do next(2022/04/27).: The zrhokq is generated for all states. And the projection is done by picking the correlated states(see **genwanrhokq** and **genzrhokq**). Thus, it is obviously cheaper if just generating the complex density in correlated subspace :

$$M_d^{\mathbf{q},\mathbf{k},G}(n \in d, n' \in d) = \langle \psi_{k+q,n\in d} | \, e^{-i\cdot(\mathbf{q}+\mathbf{G})\cdot\mathbf{r}} \, | \psi_{k,n'\in d} \rangle \tag{0.9}$$

or for all states:

$$M_{all}^{\mathbf{q},\mathbf{k},G}(n \in all, n' \in all) = \langle \psi_{k+q,n\in all} | \, e^{-i\cdot(\mathbf{q}+\mathbf{G})\cdot\mathbf{r}} \, | \psi_{k,n'\in all} \rangle \tag{0.10}$$

However, the value between 2 are different, very confused:

$$M_{all}^{\mathbf{q},\mathbf{k},G}(n=d_1,n'=d_2) \neq M_{d}^{\mathbf{k},\mathbf{q},G}(n=d_1,n'=d_2) \tag{0.11}$$

So I was wondering why they are not equal. I'm going to build a simplest repeatable subroutines to see if the projected densities will also be different. Possibly try just use M_d

Correct: After test, they are equal:(2022/04/27)

$$\Psi_{MT/IR,k,allst}(ist \in d) = \Psi_{MT/IR,k,dstate}(ist \in d) \tag{0.12}$$

Update:(2022/04/29) update Jmm expression.

Problem:(2022/04/29) 1. for Ni, V is stable $\sim$ 24 with k,q and gmaxrfwhile for J_bare, is too low $\sim$ $0.63 - 0.65$ (0.8-0.9 in the paper) May be something in wanprjkpl of the wannier projector is wrong