Construction of Wannier Functions from Local Orbitals

Robert Van Wesep

July 25, 2013

1 Site-Centered Case

I'll begin by defining our pre-orthogonalized Wannier functions using a projection of the Bloch states associated with bands within some energy interval:

$$\tilde{W}_n(\mathbf{r}-\mathbf{R}) = \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{R}} \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle \Psi^{\sigma j\mathbf{k}} | \phi_n \rangle \Psi^{\sigma j\mathbf{k}}(\mathbf{r}) \tag{1}$$

Here n is a composite index that includes α, σ, l and m. α labels the atomic sphere, σ labels spin, l is a true angular momentum quantum number and m labels real harmonic functions in a local coordinate system. $\mathbf{R}$ is a lattice vector. The bands included in the sum depend on $\mathbf{k}$ because a band may be inside the desired energy interval for some wavevectors and not for others. The prime in the sum over $\mathbf{k}$ indicates that this is a weighted sum over the first Brillion zone (an approximate integral). ϕ_n is a local orbital with a radial part determined by the Schrödinger equation with the boundary condition that the function must go to zero at the muffin tin radius and an angular part equal to a real harmonic in a some local coordinate system.

$$\phi_n(\mathbf{r}) = \sum_{m_1=-l}^{l} \phi_l^{\alpha}(r) R_{lmm_1} Y_{lm_1}(\hat{\mathbf{r}})$$

R transforms spherical harmonics with a global z-axis into real harmonics in a local coordinate system.

In order to find the expression that contains the quantities that are used in the code, we must expand Ψ inside the muffin tin:

$$\Psi^{\sigma j\mathbf{k}}(\mathbf{r}) = \sum_{lm\lambda} A^{\sigma j\mathbf{k}\alpha}_{lm\lambda} f^{\alpha}_{l\lambda}(r) Y_{lm}(\hat{\mathbf{r}})$$

The α index in Ψ is "hidden" by the fact that $\mathbf{r} \in$ muffin tin α. λ labels both APW functions and local orbitals. Using this expansion we can find the inner product in (1):

$$\begin{aligned}
\langle \Psi^{\sigma j\mathbf{k}} | \phi_n \rangle &= \int \mathrm{d}^3 r \sum_{l'm'\lambda} A^{*\sigma j\mathbf{k}\alpha}_{l'm'\lambda} f^{\alpha}_{l'\lambda}(r) Y^*_{l'm'}(\hat{\mathbf{r}}) \sum_{m_1=-l}^{l} \phi^{\alpha}_l(r) R_{lmm_1} Y_{lm_1}(\hat{\mathbf{r}}) \\
&= \sum_{l'm'\lambda} \sum_{m_1=-l}^{l} A^{*\sigma j\mathbf{k}\alpha}_{l'm'\lambda} \int_0^{r_{mt}} \mathrm{d}r\, r^2 f^{\alpha}_{l'\lambda}(r) \phi^{\alpha}_l(r) R_{lmm_1} \int \mathrm{d}\Omega\, Y^*_{l'm'}(\hat{\mathbf{r}}) Y_{lm_1}(\hat{\mathbf{r}}) \\
&= \sum_{\lambda} \sum_{m_1=-l}^{l} A^{*\sigma j\mathbf{k}\alpha}_{lm_1\lambda} \int_0^{r_{mt}} \mathrm{d}r\, r^2 f^{\alpha}_{l'\lambda}(r) \phi^{\alpha}_l(r) R_{lmm_1}
\end{aligned}$$

The code performs this projection by calling the subroutine wan_genprjlo in the subroutine wan_gencsv contained in the file mod_wannier.f90:

```
call wan_genprjlo(ilo,prj_to_ylm,ias,lm,ispn,lmmax,wfsvmt(1,1,1,1,j),&
wanc(n,j))
```

In mod_wannier.f90, subroutine wan_genprjlo:

```
do m1=-l,l
  lm1=idxlm(l,m1)
  do io1=1,nufr(l,is)
! project to local orbital
    if (wannier_prjao.eq.0) then
      prjlo=prjlo+dconjg(wfsvmt(lm1,io1,ias,ispn))*&
        &ufrp(l,io1,apword(l,is)+ilo,ic)*rylm_lps(lm,lm1,ias)
```

In this code `wfsvmt` is A, `ufrp` is $\int \mathrm{d}r\, r^2 f^{\alpha}_{l'\lambda}(r) \phi^{\alpha}_l(r)$ and `rylm_lps` is R.

Now we must orthogonalize these Wannier functions which means we must examine the quantity

$$\langle \tilde{W}_n(\mathbf{R})|\tilde{W}_{n'}(\mathbf{R}')\rangle = \int \mathrm{d}^3 r\, \tilde{W}_n^*(\mathbf{r}-\mathbf{R})\tilde{W}_{n'}(\mathbf{r}-\mathbf{R}')$$

$$= \int \mathrm{d}^3 r \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle\phi_n|\Psi^{\sigma j\mathbf{k}}\rangle \Psi^{*\sigma j\mathbf{k}}(\mathbf{r}) \sum_{j'=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle\Psi^{\sigma' j'\mathbf{k}}|\phi_{n'}\rangle \Psi^{\sigma' j'\mathbf{k}}(\mathbf{r})$$

$$= \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle\phi_n|\Psi^{\sigma j\mathbf{k}}\rangle \sum_{j'=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle\Psi^{\sigma' j'\mathbf{k}}|\phi_{n'}\rangle \int \mathrm{d}^3 r\, \Psi^{*\sigma j\mathbf{k}}(\mathbf{r})\Psi^{\sigma' j'\mathbf{k}}(\mathbf{r})$$

$$= \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle\phi_n|\Psi^{\sigma j\mathbf{k}}\rangle\langle\Psi^{\sigma j\mathbf{k}}|\phi_{n'}\rangle$$

Let's define an overlap matrix by

$$O_{nn'}(\mathbf{k}) \equiv \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle\phi_n|\Psi^{\sigma j\mathbf{k}}\rangle\langle\Psi^{\sigma j\mathbf{k}}|\phi_{n'}\rangle \tag{2}$$

Then we can define an inverse square root matrix

$$S_{nn'}(\mathbf{k}) \equiv O_{nn'}^{-1/2}(\mathbf{k})$$

and define new Wannier functions as

$$W_n(\mathbf{r}-\mathbf{R}) = \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{R}} \sum_{n'} S_{n'n}(\mathbf{k}) \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle\Psi^{\sigma j\mathbf{k}}|\phi_{n'}\rangle \Psi^{\sigma j\mathbf{k}}(\mathbf{r}) \tag{3}$$

In mod_wannier.f90, subroutine wan_ort_k this procedure is performed on the expansion coefficients $\langle\Psi^{\sigma j\mathbf{k}}|\phi_n\rangle$, which are represented by `wanc`:

```
! compute ovelap matrix
s=zzero
do m1=1,nwantot
  do m2=1,nwantot
    do j=1,nstsv
```

```
      s(m1,m2)=s(m1,m2)+dconjg(wanc(m1,j))*wanc(m2,j)
    enddo
  enddo
enddo
! compute S^{-1/2}
call isqrtzhe(nwantot,s,ierr)
ierr_=ierr
! compute Wannier function expansion coefficients
wanc_ort=zzero
if (ierr.eq.0) then
  do m1=1,nwantot
    do m2=1,nwantot
      wanc_ort(m1,:)=wanc_ort(m1,:)+wanc(m2,:)*s(m2,m1)
    enddo
  enddo
  wanc=wanc_ort
endif
```

Let's check to make sure that our new Wannier functions give the desired result.

$$
\begin{aligned}
\langle W_n(\mathbf{R})|W_{n'}(\mathbf{R}')\rangle &= \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \sum_{n''} S_{n''n}(\mathbf{k}) \langle \phi_{n''}|\Psi^{\sigma j\mathbf{k}}\rangle \sum_{n'''} S_{n'''n'}(\mathbf{k}) \langle \Psi^{\sigma j\mathbf{k}}|\phi_{n'''}\rangle \\
&= \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} \sum_{n''n'''} S_{nn''}(\mathbf{k}) S_{n'''n'}(\mathbf{k}) \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle \phi_{n''}|\Psi^{\sigma j\mathbf{k}}\rangle \langle \Psi^{\sigma j\mathbf{k}}|\phi_{n'''}\rangle \\
&= \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} \sum_{n''n'''} S_{nn''}(\mathbf{k}) O_{n''n'''}(\mathbf{k}) S_{n'''n'}(\mathbf{k}) \\
&= \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} [\mathbf{SOS}]_{nn'}(\mathbf{k}) = \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} [\mathbf{SS}^{-1}\mathbf{S}^{-1}\mathbf{S}]_{nn'}(\mathbf{k}) \\
&= \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}'-\mathbf{R})} \delta_{nn'} = \delta_{\mathbf{R}\mathbf{R}'}\delta_{nn'}
\end{aligned}
$$

The last task is to compute the inverse square root matrix. This can be done by diagonalizing the overlap matrix and taking the inverse square

root of the eigenvalues:

$$\mathbf{S} = \mathbf{O}^{-1/2} \text{ means } \left(\mathbf{S}^{-1}\right)^2 = \left(\mathbf{S}^2\right)^{-1} = \mathbf{O}$$

$$\text{Let } \mathbf{O} = \mathbf{U}\mathbf{D}\mathbf{U}^\dagger, \text{ then } [\mathbf{S}, \mathbf{O}] = 0 \implies \mathbf{S} = \mathbf{U}\mathbf{D}'\mathbf{U}^\dagger$$

$$\begin{aligned}
\mathbf{U}\mathbf{D}'\mathbf{U}^\dagger = \mathbf{S} = \mathbf{O}^{-1/2} &= \left[\mathbf{U}\mathbf{D}\mathbf{U}^\dagger\right]^{-1/2} \\
\left[\mathbf{U}\mathbf{D}'\mathbf{U}^\dagger\mathbf{U}\mathbf{D}'\mathbf{U}^\dagger\right]^{-1} &= \mathbf{U}\mathbf{D}\mathbf{U}^\dagger \\
\left[\mathbf{U}\mathbf{D}'^2\mathbf{U}^\dagger\right]^{-1} &= \\
\mathbf{U}\left(\mathbf{D}'^2\right)^{-1}\mathbf{U}^\dagger = \mathbf{U}\mathbf{D}\mathbf{U}^\dagger &\implies \mathbf{D}' = \mathbf{D}^{-1/2} \\
\therefore \mathbf{S} = \mathbf{U}\mathbf{D}^{-1/2}\mathbf{U}^\dagger &
\end{aligned}$$

2 Linear Combinations

Rather than be restricted to projecting onto local orbitals such that each Wannier function is associated with one particular orbital, we may instead like to project onto a linear combination of local orbitals. This can be important in compounds for which an atomic-sphere-centric, ionic picture is not natural and instead the substructures appropriate for understanding the physics are molecular in nature. We would like any such linear combination to preserve the orthonormality of the set of Wannier functions that we have contructed thus far. Hence, we seek a unitary transformation:

$$W_n^{\mathrm{LC}}(\mathbf{r} - \mathbf{R}) = \sum_{n'} U_{nn'} W_{n'}(\mathbf{r} - \mathbf{R}) \tag{4}$$

This is simple enough, but what we are considering is the projection onto a linear combination of local orbitals in order to produce our set of Wannier functions rather than taking a linear combination after the projection has already been performed. I will show below that these alternative ways of thinking are in fact two sides of the same coin. Let's start by plugging (3) into (4):

$$W_n^{\mathrm{LC}}(\mathbf{r} - \mathbf{R}) = \sum_{n'} U_{nn'} \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{R}} \sum_{n''} S_{n''n'}(\mathbf{k}) \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle \Psi^{\sigma j\mathbf{k}} | \phi_{n''} \rangle \Psi^{\sigma j\mathbf{k}}(\mathbf{r}) \tag{5}$$

It seems logical to define our linear combination of local orbitals in the same way as we defined the linear combination of Wannier functions, so we will do just that and see if we get consistant results.

$$|\phi_n^{\mathrm{LC}}\rangle = \sum_{n'} U_{nn'} |\phi_{n'}\rangle \tag{6}$$

Inverting (6) is simple because the transformation is unitary. Plugging the inverted relation into (5):

$$W_n^{\mathrm{LC}}(\mathbf{r}-\mathbf{R}) = \sum_{n'n''} U_{nn'} \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{R}} S_{n''n'}(\mathbf{k}) \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle \Psi^{\sigma j\mathbf{k}} | \sum_{n'''} U^*_{n''n'''} |\phi^{\mathrm{LC}}_{n'''}\rangle \Psi^{\sigma j\mathbf{k}}(\mathbf{r}) \tag{7}$$

With this result in mind, let's perform the same procedure that we did in order to construct the atomic-like Wannier functions. We begin by writing an expression analogous to (1):

$$\tilde{W}_n^{\mathrm{LC}}(\mathbf{r}-\mathbf{R}) = \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{R}} \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle \Psi^{\sigma j\mathbf{k}} | \phi_n^{\mathrm{LC}}\rangle \Psi^{\sigma j\mathbf{k}}(\mathbf{r}) \tag{8}$$

We can define an overlap matrix in the same way as (2) and write it in terms of that previous matrix:

$$O_{nn'}^{\mathrm{LC}}(\mathbf{k}) \equiv \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle \phi_n^{\mathrm{LC}} | \Psi^{\sigma j\mathbf{k}}\rangle \langle \Psi^{\sigma j\mathbf{k}} | \phi_{n'}^{\mathrm{LC}}\rangle = \sum_{n''n'''} U^*_{n''n} O_{n''n'''}(\mathbf{k}) U_{n'n'''} \tag{9}$$

The inverse square root matrix can be used in the same way as before to orthogonalize (8):

$$W_n^{\mathrm{LC}}(\mathbf{r}-\mathbf{R}) = \sum_{\mathbf{k}}{}' \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{R}} \sum_{n'} S_{n'n}^{\mathrm{LC}}(\mathbf{k}) \sum_{j=N_1^{\mathbf{k}}}^{N_2^{\mathbf{k}}} \langle \Psi^{\sigma j\mathbf{k}} | \phi_{n'}^{\mathrm{LC}}\rangle \Psi^{\sigma j\mathbf{k}}(\mathbf{r}) \tag{10}$$

In a manner similar to how the inverse square root matrix was computed in the previous section, it is easy to show from (9) that $S_{n'n}^{\mathrm{LC}}(\mathbf{k})$ can be written as

$$S_{nn'}^{\mathrm{LC}}(\mathbf{k}) = \sum_{n''n'''} U^*_{n''n} S_{n''n'''}(\mathbf{k}) U_{n'n'''}$$

By inspecting (7) with this in mind, we can see that (5) and (10) are consistant with each other. Thus, it does not matter whether we first project onto a linear combination of local orbitals and then orthogonalize the resulting Wannier functions or we produce orthogonalized Wannier functions from the local orbitals and then form linear combinations from those. As long as we use the same transformation in both cases both methods are valid.

3 Automatic Linear Combinations

In the previous section I implicitly assumed the following:

1. For a real system an orthonormal set of Wannier functions could be constructed that spans the Hilbert space defined by the Bloch states on to which we are interested in projecting.

2. Any linear combination would be produced via a unitary transformation, which requires that the number of orbitals produced via the linear combination is equal to the number of orbitals from which the combinations are constructed.

A simple example can illustrate where both of these assumptions produce a scheme that is inadequate for a real system. To be concrete, suppose we are interested in projecting onto three bands, which means that we can project at most three local orbitals. To see this it is useful to think of the $\langle \Psi^{\sigma j \mathbf{k}} | \phi_n \rangle$ from (1) as components of a vector in Bloch space (with j indexing the components) defined at each point in the Brillion zone. If we are projecting onto three bands than these vectors will be three-dimensional. Now, eventually these vectors are orthonormalized and for that to succeed there can be at most three vectors and so there can be at most three $|\phi_n\rangle$. With this limited number of options it becomes quite possible that no set of three orbitals will span the three-dimensional Bloch space at every point in the Brillion zone. Actually it is even worse than that, because even if we are happy with only projecting one or two orbitals it may still be possible to fail. For example, if one of these orbitals is null at some point in the Brillion zone there is no hope of orthonormalizing the set as it will always be linearly dependent (even if that set contains only one vector!). There are some solutions to this conundrum.

First, we can use the framework of Section 2 to form a unitary transformation from a set of local orbitals that is larger than three and then pick out only three (or less) from the result that will hopefully span our Bloch space. If we do this we can, in practice, only go one way on the two way street outlined in that section. That is to say, we must perform the unitary tranformation on the local orbitals and then project rather than projecting first and then performing the transformation. In reality the projection of the transformed local orbitals would only be performed on the three (or less) that we have picked. Now, this can and has been done, but it has so far proceeded by way of trial and error as there is no good way to know with certainty whether or not the set of linear combinations of local orbitals used for projection will span our three-dimentional Bloch space. This proves to be quite labor intensive for systems for which there are a large number of orbitals that could potentially come into play and the orbital character of the bands changes dramatically across the Brillion zone. So this solution, while expedient, is not nessesarily ideal.

Second, we can use all of the orbitals we need from the start regardless of the number of bands and use linear algebra to produce from those a set that is guaranteed to work. This, in effect, automatically defines the linear combination outlined in the previous paragraph, hence the title of this section. The bit of linear algebra that is useful is the singular value decomposition. To begin, let us construct a matrix $\mathbf{A}$ whose columns are given by N vectors $\mathbf{v}_n$ defined by

$$\mathbf{v}_n \equiv \begin{bmatrix} \langle \Psi^1 | \phi_n \rangle \\ \langle \Psi^2 | \phi_n \rangle \\ \langle \Psi^3 | \phi_n \rangle \end{bmatrix} \tag{11}$$

I have supressed the σ and $\mathbf{k}$ indices, but keep in mind that this procedure will be performed at every point in the Brillion zone as this will become important later.

Now, let us look at the singular value decomposition of $\mathbf{A}$:

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\dagger \tag{12}$$

$\mathbf{\Sigma}$ is a real, diagonal matrix whose entries are the "singular values" of $\mathbf{A}$, the columns of $\mathbf{U}$ and $\mathbf{V}$ are the "left-" and "right-singular vectors" of $\mathbf{A}$, respectively. Now, it is the set of left-singular vectors that we are interested in. The subset of left-singular vectors that correspond to finite singular

values span the range of $\mathbf{A}$. This is precisely what we want. If we have enough vectors (each corresponding to a local orbital), then the range of $\mathbf{A}$ should be the same as the space defined by our three bands and the left-singular vectors should contain Wannier coefficients that can be safely orthonormalized.

Futhermore, this effectively produces a linear combination of local orbitals. I think this is conceptually obvious, but it is mathematically subtle. To see this let us denote our left-singular vectors as $\mathbf{w}_i$ and write these vectors in terms of the original $\mathbf{v}_n$. Now this is a bit complicated, because in order to do this we need to find a subset of three $\mathbf{v}_n$ for each $\mathbf{w}_i$ that is linearly independent. In other words we need a basis. Though I cannot recall a theorem, I think this is always possible for the particular case we have here, so let us move forward assuming that we have found such a subset (that could, in principle, be different for each $\mathbf{w}_i$) and the index $m(i)$ runs over this subset:

$$\mathbf{w}_i = \sum_{m(i)} \mathbf{v}^\dagger_{m(i)} \mathbf{w}_i \mathbf{v}_{m(i)} = \sum_{n=1}^{N} \alpha_{ni} \mathbf{v}_n \tag{13}$$

where I have constructed the α_{ni} in such a way that the ones for $n \neq m(i)$ are zero. I've done this in order to make a conceptual point. Namely that these α_{ni} are akin to the $U_{nn'}$ described in Section 2, but with a caveat. That is the matrix formed by the α_{ni} is, in general, rectangular and, thus, not unitary, which is why I am using different notation for it. Now here it is useful to write these as $\alpha^{\mathbf{k}}_{ni}$ as a reflection of the reality that these coefficients will be different for each point in the Brillion zone. In this way the $\alpha^{\mathbf{k}}_{ni}$ are again different from the $U_{nn'}$ as the latter were constant with $\mathbf{k}$. Constant coefficients conform more to the notion of forming linear combinations of orbitals in order to describe the local chemistry of a compound. The fact that the linear combination here varies over the Brillion zone is a possible conceptual shortcoming of the method. However, one could imagine taking some kind of average of the $\alpha^{\mathbf{k}}_{ni}$ over the Brillion zone as a compromise that would return us to more comfortable conceptual territory.

A shrewd read may say at this point, "Wait, if we have found a subset of the $\mathbf{v}_n$ that spans the same space as the $\mathbf{w}_i$, why not use those? What's the point of all this anyway?" Here there are a couple more subtle points. First, we "found" the subset of the $\mathbf{v}_n$ only in the sense that we said that they existed and we could, in principle, find them. In the real world, we do

not know a priori what these particular $\mathbf{v}_n$ are, which gets at the point of this method. Also, the suitible subset may differ at different points in the Brillion zone, so it would be impossible to pick the three $\mathbf{v}_n$ that we need from the beginning. Of course, I cannot dismiss the question completely. It touches on a potential, practical difficulty in interpreting the $\mathbf{w}_i$ we have found in terms of the local orbitals with which we started. The inner products $\mathbf{v}_n^\dagger \mathbf{w}_i$ may still be useful for such an interpretation, but I have not worked out how yet.

Moving from the concrete to the general, there is, of course, nothing special about the number three other than it is in a sense small enough to potentially give us trouble. That is not to say that if the number of bands are 20 we would be on safe ground; that is certainly not the case in general. We also could imagine $\mathbf{A}$ being a square matrix. In this example we may be attempting to project orbitals onto an equal number of bands. This method could still be useful as a way to produce some number of orbitals less than the number of bands if we find the straight forward projection does not work. We could further imagine that the number of bands is greater than the number of orbitals. In this case the method should still produce a set of vectors that can be orthonormalized. Though, I have trouble seeing the practical utility of the method in this last case.