I am Lili Rassouli, a second-year Ph.D. student in chemical engineering from the university at buffalo. I am working in Dr. Dupuis's lab on the modeling of electron transfer in photoelectrochemical devices. I will discuss in this report how I can take advantage of the computational chemical packages discussed in this workshop.

As I have mentioned, my project is about modeling the electron transfer in photoelectrochemical cells, specifically on hematite semiconductors. Different parameters affect the efficiency of photoelectrochemical cells. For instance, carrier lifetime and mobility are two critical characteristics of PECs. Knowing more detailed knowledge about the charge carrier structure and its dynamics could lead to the discovery of better PEC materials. So I need to define the excited state in a solid state.

Among all different packages, I focused on Libra and quantum espresso as they are applicable for solid states and periodic systems. I concentrated on one of the tutorials in Libra; that you can find it in Github (https://github.com/compchem-cybertraining/Tutorials_Libra/tree/master/6_dynamics/2_nbra_workflows). This tutorial was discussed on the fourth day of the workshop. You can find its details on the workshop website under section 8. NBRA workflows with Libra.

The approach is used in this nonadiabatic molecular dynamics tutorial is based on trajectory surface hopping (TSH). Two main approximations are Neglect-of-Back- Reaction Approximation (NBRA) and Single-particle approximation (SP). In NBRA, after photoexcitation and change of electronic states, nuclear evolutions are neglected. In SP, 1-electron molecular orbitals are considered to build Kohn–Sham (KS) orbitals or Slater determinants (SDs).

I ran this tutorial for hematite data. It is in 4 different steps that are given in the workshop GitHub under course_work/lilirass/Workshop_project.

In the first step, the molecular dynamics trajectory was calculated using the Quantum Espresso software package. PWscf (Plane-Wave Self-Consistent Field) sub-package is specifically used in this code. So for calculating the molecular dynamics, md calculation is used in the first step.

I chose plane-wave energy cut-off equal to 100 Ry and the charge density cut-off equal to 500 Ry to have a better convergence. One hundred fs of dynamics was calculated with 100 steps and a time-step of 1 fs. The Verlet algorithm was used to integrate Newton's equation.

X0.md.out would be the most critical output that shows 100 fs molecular dynamics trajectory of Hematite using the Quantum Espresso software package.

In the second step, the vibronic Hamiltonian was calculated. According to the output of step1, wavefunctions for each timestep were calculated. The SCF calculation was used at each molecular geometry. Then the vibronic Hamiltonian was constructed using the calculated wavefunctions.

Some important parameters in step 2 are: nbnd, minband, maxband, and num_alpha_ks_orbs.

Number of electrons = 136 according to the x0.md.out from step 1

Number of orbitals = 136/2 = 68

nbnd = number of orbitals + arbitrary number of excited orbitals

nbnd = 68 + 12 = 80

minband and maxband are maximum and minimum orbital numbers that are close to the HOMO and LUMO orbitals. I chose them according to this picture:
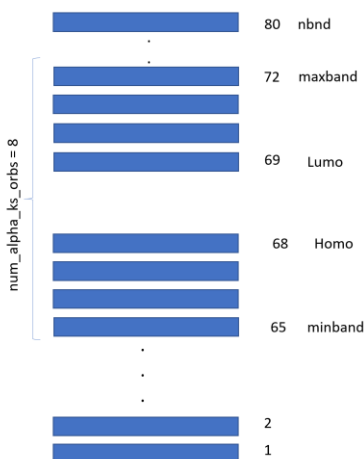
The outputs of the second step are time-dependent Kohn Sham energies and time-averaged nonadiabatic couplings plots as shown in the following pictures:
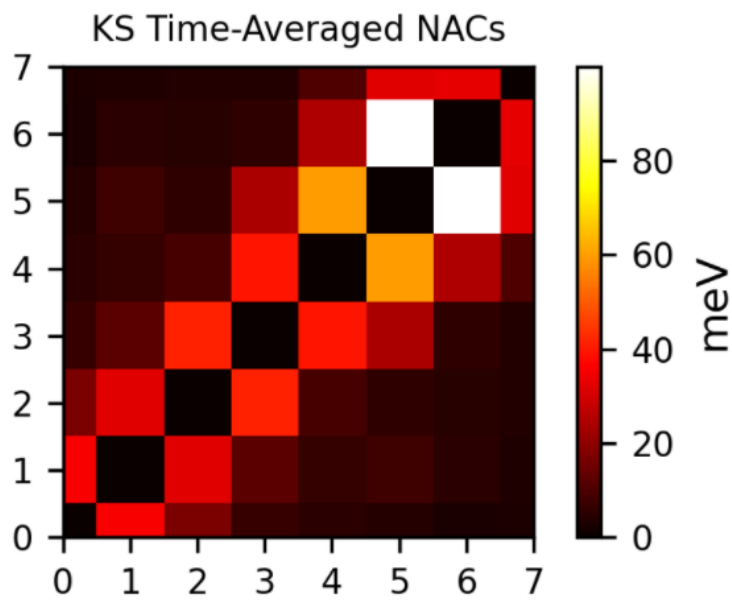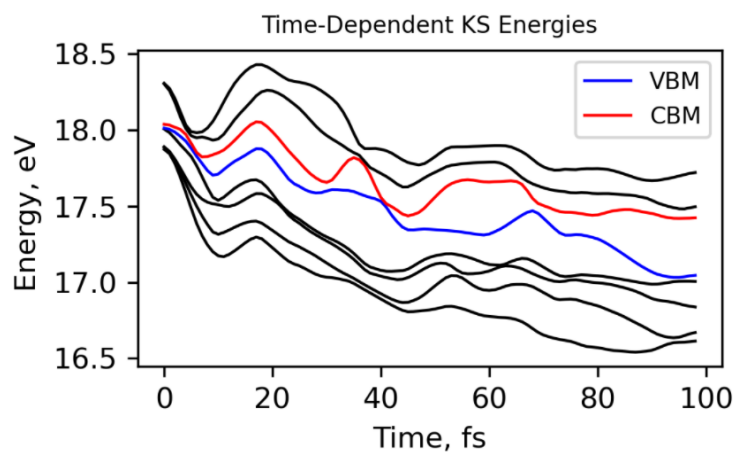
Other important outputs are in the res folder:

S: Overlap matrix for time-step "N" in the Kohn-Sham orbital basis

St: Time-derivative Overlap matrix for time-step "N" and "N+1" in the Kohn-Sham orbital basis

Hvib: Vibronic Hamiltonian matrix for the time-step "N" and "N+1" in the Kohn-Sham orbital basis

In step 3, I computed the energies and nonadiabatic couplings in the Slater determinant basis instead of KS orbitals. The vibronic Hamiltonian calculated in step 2 was used as an input file in this step. I used Libra to auto-generate a Slater determinant basis instead of writing that manually. For this purpose, only alpha excitation was defined. The number of alpha and beta conduction bands were defined according to the following picture for Hematite. So cbm_alpha_index = 5 and cbm_beta_index = 13
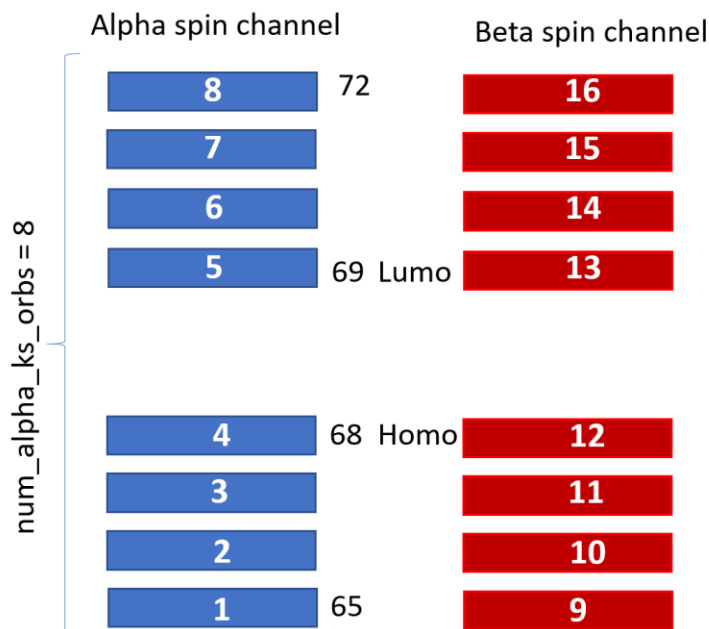

I calculated S, St, and Hvib from step 2 to define vibronic Hamiltonian in the Slater Determinant basis. Different parameters such as orthogonalization to orthogonalize the Kohn-Sham orbital basis can be selected in parameters of step3.run function.

I discuss one example for constructing a 3 Slater determinant basis according to KS orbitals. The first one is the ground state determinant, and the next two are the excitations:

[4, -12], [5, -12], [6, -12]

Finally, the outputs from step 3 are time-dependent slater determinant energy and slater determinant time average nonadiabatic couplings plots. A folder with Vibronic Hamiltonian matrix for slater determinant was also generated.
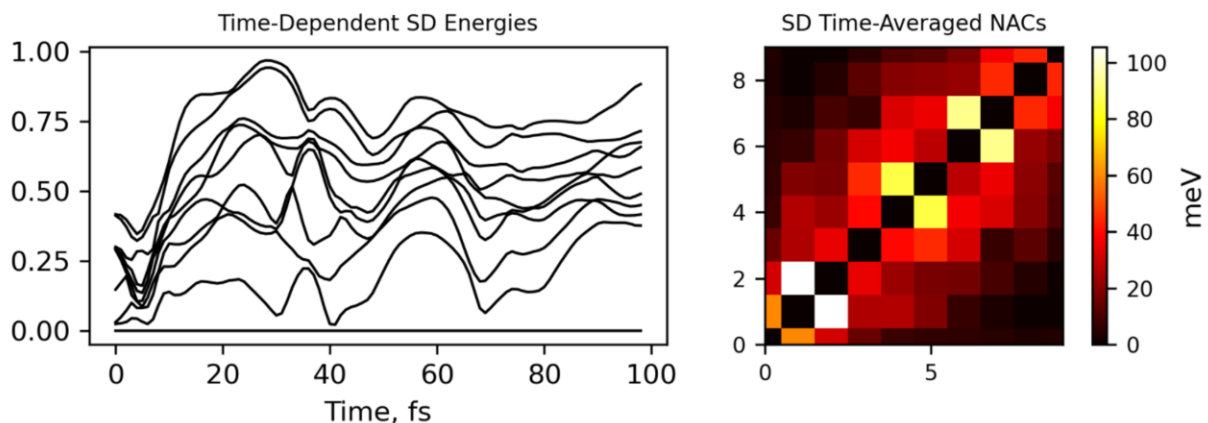
In the last step, NBRA NAMD was used to compute average decoherence times, to compute the energy gaps along the trajectory, to compute average energy gaps, and to plot maps of dephasing times for all pairs of states.

Particular active space should be defined to be considered in the dynamics:

I have used eight alpha and eight beta orbitals in the previous steps. At this step, the index of the CBM and VBM are from 0, so homo would be three and Lumo would be four in this case. It is started from 0 here, because the computer indexes the first row and column matrix elements from zero.

In addition, a subset of 100 dynamics should be defined with the list of init_time and n_steps. For init_time=40 and n_steps = 50, The simulation will run from initial time 40 to 90.

One another important parameter is nstates in step4.get.Hvib function. I used Libra to auto-generate the Slater determinant basis. The number of Slater determinants generated in step 3 defines nstates. That is 10 in my case.

The outputs of the fourth step are dephasing time for Kohn-sham orbitals and Slater determinant. In addition, the plots of electron population in Lumo orbitals with different decoherence_method are sketched.

I defined the next steps for myself:
- I will change the QE input file to consider spin-polarization and magnetization
- I wil run the code with CP2K instead of QE
- I will run the code for many-body treatment of excited states instead of single particle
- I will calculate the exciton lifetime