BerkeleyGW IO Optimization

Nicholas Schoeck, Graham Wood, Gabriel Dick

Overview

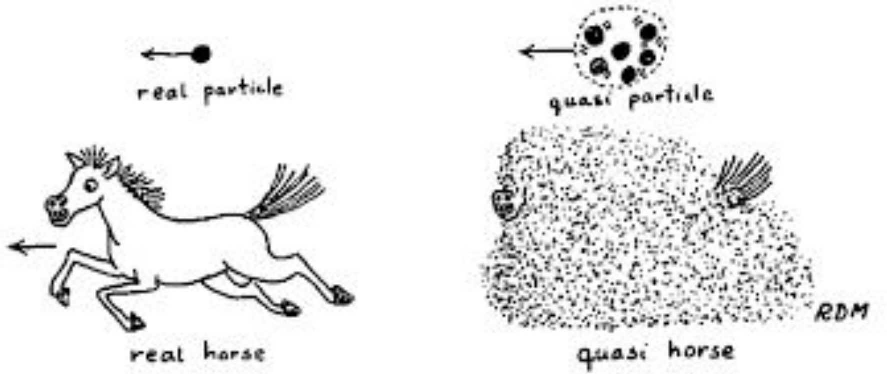
- What is BerkeleyGW
- Overview of HDF5
- Current status of BerkeleyGW
- What related works have done
- Specification of our project
- Optimization and parallelization of IO
- Expected performance increases
- Questions?

BerkeleyGW

- Part of National Renewable Energy Lab in Colorado
- BerkeleyGW is a set of code that calculates the quasiparticle properties and optical responses of a variety of materials.
  - The different materials include bulk semiconductors, insulators, metals, and nano-systems.
- It uses the GW approach, based on approximating the electron *self-energy* as a part of an expansion in a *Coulomb interaction*, to get quasiparticle band gaps and dispersion relations.
  - *Self-energy* is when a particle interacts with a many-body system, creates a cloud, and the cloud disrupts the particle's motion.
  - *Coulomb's Law* is a quantification of the force between two electrically charged particles.

Quasi-Particles

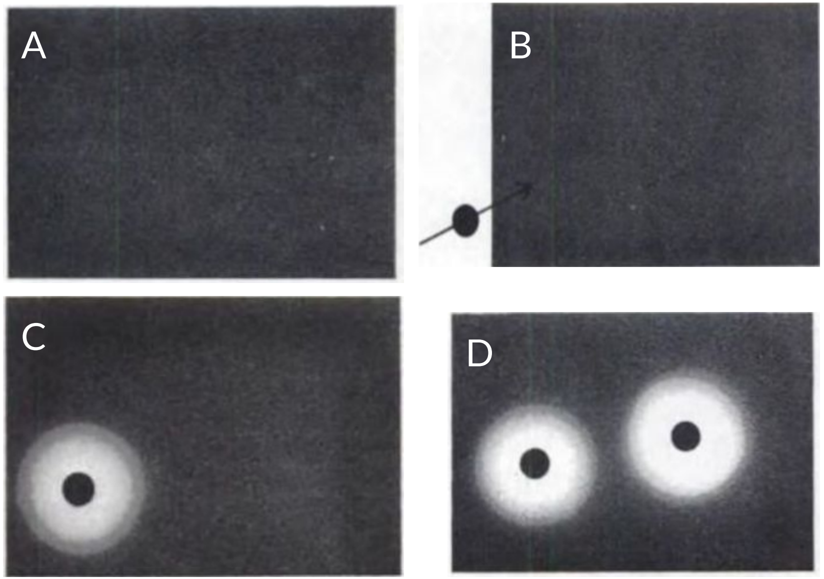
- We can imagine a real particle to be the horse as an object
- Quasiparticle = original individual particle + cloud of disturbed neighbors
  - As the horse runs it kicks up dust and the Horse + Dust is what we consider the Quasi-Particle



Quasi-Particles in Berkeley GW

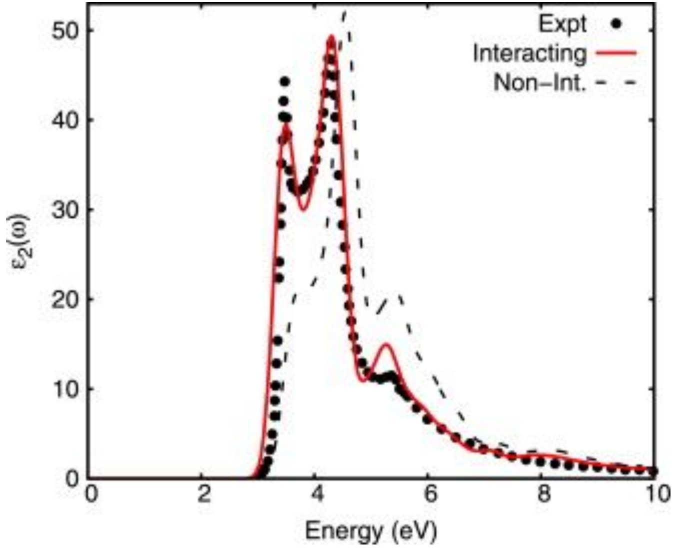
Here is an example of the GW approximation.

A. Many electrons interacting in a uniform, neutral system.
B. Shoot an electron into this system, and the extra electron repels the others.
C. A lack of electrons create a “hole” as other electrons are pushed away
    a. The extra electron + this hole is what the “quasiparticle” is.
D. Two quasiparticles interact through a Coulomb interaction.



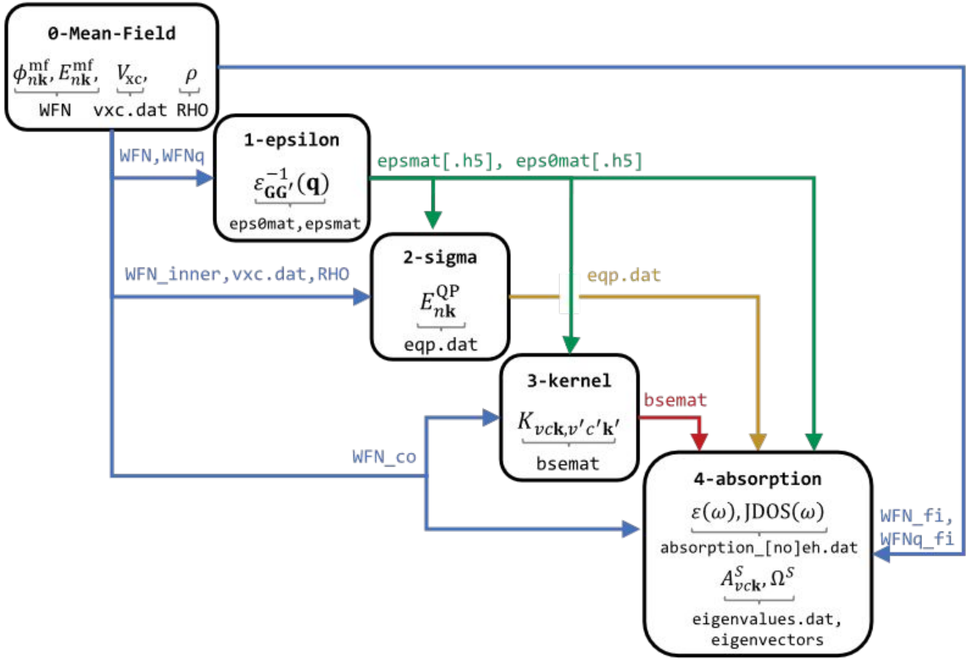
Wave Functions

- Orbitals (wavefunctions) describes how electrons go in a system
- Wave functions are orbitals represented as momentum instead of space
- Wave functions are calculated as a result of the Dyson Equation
- Can be stored in an array



BerkeleyGW's Three Executables

BerkeleyGW uses three main components in the program: Epsilon, Sigma, and BSE.



Epsilon, Sigma, and BSE

Epsilon

- Generates the dielectric matrix and its inverse.

Sigma

- Computes the self-energy corrections to the DFT eigenenergies using the GW approximation.

BSE

- Solves the Bethe-Salpeter equation for correlated electron-hole excitations.

What is HDF5?

HDF stands for “Hierarchical Data Format”, and is a format designed to be a portable data type for any kind of data.
Some of its features include:

- It does not limit the size of files or the number of objects
- Its functionality and its data can be used across virtually any computing platform
    - It can create files on different machines, using different computers, in different languages and can still be effectively used anywhere.
- It can represent a variety of complex objects as well as a variety of metadata
- It uses a virtual file layer to offer flexible storage and transfer capabilities
    - This includes “The parallel I/O driver for HDF5 reduces access times on parallel systems by reading/writing multiple data streams simultaneously.”
    - Allows for simultaneous access by multiple processes(MPI)

Current Status of BerkeleyGW

- Math operations are highly scalable and parallelized
- Has been ran on up to a million cores
- Performance up to 50 GB/s on some machines
- IO of HDF5 documents take 80% of the time (48 minutes out of an hour)
- IO operations are still serial, huge bottleneck
- Can study a system of up to 500 atoms in a single run

Related Works

- The Epsilon portion of the project has parallelized IO
- IO was parallelized over the band dimension

Our task

- Remove IO Absorption code from core project structure
- Create a Kernel for Absorption IO
- Using HDF5 to parallelize HDF5 IO files over K-Points of Wavefunctions
- Removed data dependencies for IO
- Preserve functionality and increase efficiency

Testing

- Remove Kernel from project makes testing easier
- Get running on Beocat
- Testing HDF5 files as input on Beocat over various number of cores
- BerkeleyGW has thorough time and memory tracking implemented

Expected Performance Increase

- Currently, 80% of run time is spent on IO
- Limits size of calculations they can do because of time restraints
- Expected 2x-5x speedups

Impact of Implementation

- Speedups between 2x- 5x
- Allows BerkeleyGW to do larger calculations
- Making it possible to scale it up even more by using time more efficiently and allowing IO to be scalable

Questions?

References

BerkeleyGW. (2017, October 30). About. Retrieved from https://berkeleygw.org/about/.

Deslippe, J., Samsonidze, G., Strubbe, D. A., Jain, M., Cohen, M. L., & Louie, S. G. (2012). 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. Retrieved from https://www.sciencedirect.com/science/article/pii/S0010465511003912?via%3Dihub

The HDF Group. (2014, February 13). HDF5 Technologies. Retrieved from https://support.hdfgroup.org/about/hdf_technologies.html.

Hull, Olivia. (2019) *The GW Appoximation.*

Vigil-Fowler, Derek. (2019) *hdf5, BerkeleyGW.*