QuantumWell

Python-based time-independent Schrodinger equation solver using PyQt5 to specify and view a variety of quantum wells, including wells with multiple barriers. The user specifies well parameters or accepts default values. No knowledge of python is necessary. Grinnell College students have used this project since 2016 in their modern physics courses. It was last tested with anaconda python 3.8, September 2022, for Windows 10, macOS 12.5.1, and Linux.

Getting Started

This package uses python3 and a variety of python modules. If you are new to python, it's easiest to install a python platform such as the widely used anaconda individual edition. We are currently using anaconda with python 3.8 installed with the anaconda graphical installer using these installation instructions.

Prerequisites

- macOS, Windows 7/10, or Linux operating system
- python >= 3.8, DO NOT USE PYTHON 2.x
- python modules numpy, scipy, pyqt (version 5), matplotlib, and their prerequisites; all are included in the anaconda platform.

Unless you use additional python platforms, I recommend that users take the anaconda install defaults while paying particular attention to these two options:

1. Select *Install for me only* rather than selecting *Install for all users* to avoid needing admin privileges.
2. For installs on Windows, install Anaconda in the default location to avoid potential startup complications later. I also recommend that, given the opportunity by the installer, you select *Add Anaconda3 to my path environmental variable* even though the installer recommends not using this option. Otherwise, you may have path issues when using the QuantumWell startup scripts.

Installing and Opening the main GUI

- Download the compressed file from Github by double-clicking https://github.com/dukecld/QuantumWell/archive/main.zip.
- Move the downloaded file to any directory; I recommend your Desktop. Extract the contents (double-clicking usually works for zipped files). You should see the QuantumWell-main directory.
- (OR if you are familiar with git software, you may obtain a clone of the git repository. The main directory name will now be QuantumWell. The -main from the zipped file shows that the downloaded file is the contents of the main branch in the repository.)
- From now on, I will use the name, QuantumWell, for the QuantumWell-main directory.
- Open a file browser in the QuantumWell directory Execute the config.py script by double-clicking on:

> - `config_windows.bat` for Windows. If you installed anaconda3 in the default folder, this .bat file will set the path to the anaconda3 default folder making possible the execution of the config.py script.
> - `config_macos.command` for macOS
> - `config_linux.sh` for Linux os.
> - or from a terminal window in the QuantumWell directory, execute `python config.py`. Using an Anaconda_prompt window ensures that all python modules/packages are available.

The `config.py` script first determines if you have all the QuantumWell prerequisites. It then proceeds to create a startup script for the QuantumWell main GUI and records the path to the QuantumWell directory in a config file.

Depending on your operating system, the `config.py` script produces one of the following startup scripts both in your QuantumWell directory and on your Desktop:

- `startQWell.bat` script for Windows machines. This script also if necessary activates a conda environment (adds paths to the anaconda3 folders) to enable its executing the quantumWell.py script.
- `startQWell.command` for macOS machines
- `startQWell.sh` script for Linux systems

You can copy the appropriate `startQWell.xx` to any directory you wish and open the GUI from there.

The `config.py` python script also creates a `qwconfig/config` file off of your home directory which contains the path to your QuantumWell directory and can be used to set the path for importing modules from the `src` directory into e.g. jupyter notebooks. There are more details in the Documentation/UsersGuide. If you are only using the GUIs, you can ignore this paragraph.

Open the QuantumWell GUI by

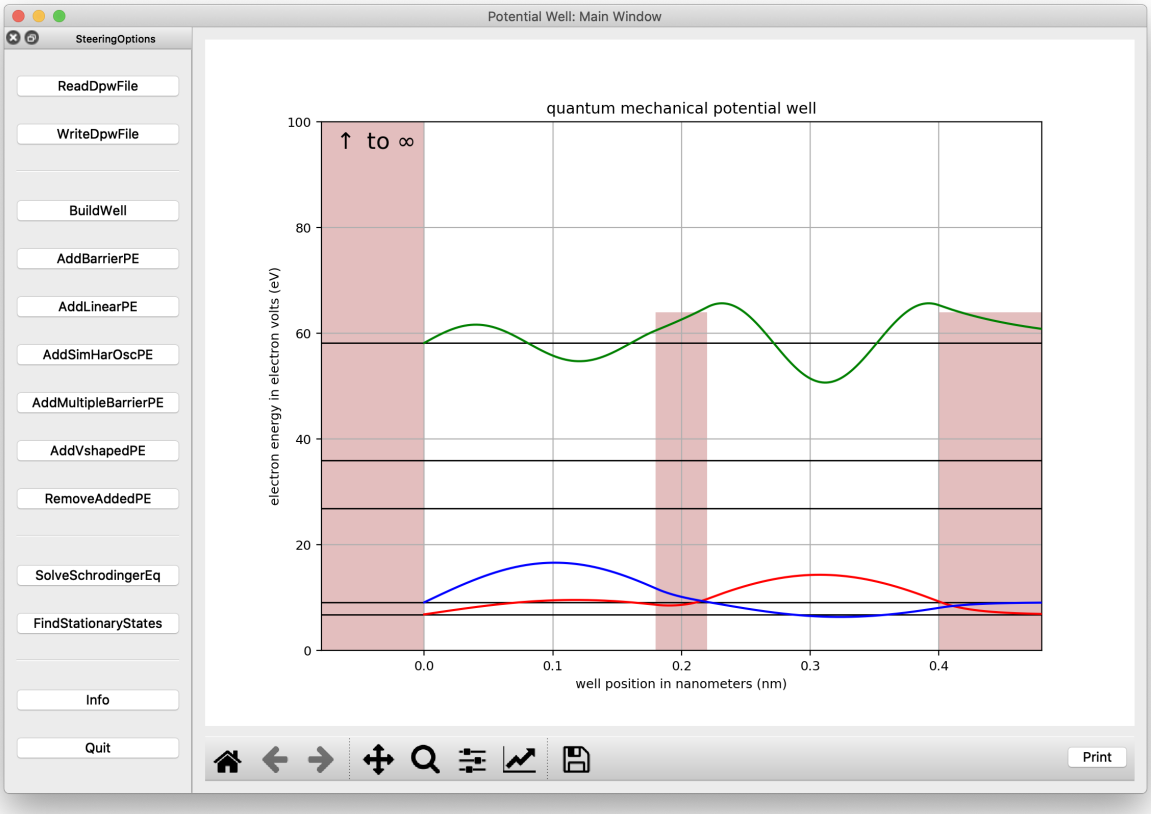
- Double-clicking the `startQWell` script from a file browser set on any directory containing the script
- or executing the script from a terminal whose shell includes paths to the anaconda3 folders.

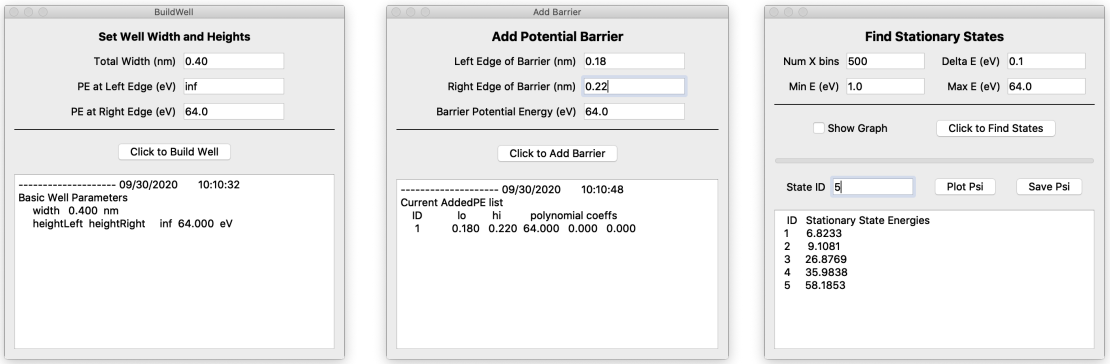
Usage

The QuantumWell GUIs are largely selfexplanatory. Each button on the main GUI opens a side window to set and record well parameters. The *SolveSchrodingerEq* window shows produces the wavefunction for a given energy (which will diverge if the energy is not a stationary-state energy). The *FindStationaryStates* window produces the stationary-state energies and selected state wavefunction plots on the main window. The always present *Message* window shows the history of your session; sending the history to a file and/or to a printer are *Message* window options.

You can find more details in the Documentation/UsersGuide

Here is an image of the main window after using default parameters from the *BuildWell*, *AddBarrierPE*, and *FindStationaryStates* windows. Images of these windows follow the main window.


Potential Well: Main Window
SteeringOptions
ReadDpwFile
WriteDpwFile
BuildWell
AddBarrierPE
AddLinearPE
AddSimHarOscPE
AddMultipleBarrierPE
AddVshapedPE
RemoveAddedPE
SolveSchrodingerEq
FindStationaryStates
Info
Quit
quantum mechanical potential well
↑ to ∞
electron energy in electron volts (eV)
well position in nanometers (nm)
0
20
40
60
80
100
0.0
0.1
0.2
0.3
0.4
Print



Tutorials

The `Tutorials` directory contains jupitor notebook tutorials to provide guides for using the QuantumWell modules in user constructed jupyter notebooks. There is more information about these modules in the Documentation/UsersGuide

Author

Charlie Duke
Professor Emeritus of Physics
Physics Department
Grinnell College
Grinnell, Iowa 50112

License

Acknowledgments

The Grinnell College physics faculty andr second-year modern-physics students have provided many important suggestions leading to a useful and robust code for an introduction to quantum-mechanical potential wells.