We describe the visualization of 3D optical lattices based on Sisyphus cooling implemented with open source software. As a main framework to build the simulation software, we utilized Python and its open source libraries, such as Numpy, Scipy, TkInter, and QuTiP. Along with the computational and simulational part, the software comes up with a graphical user interface (GUI). Thus, we expect researchers without programming background can easily access the software and utilize it for guiding their experimental implementations of optical lattices. The software has been published to a Github repository (https://github.com/ohpyupi/3d-optical-lattice). Utilizing the software, we plot the adiabatic light shift potentials found by diagonalizing the effective Hamiltonian for the light shift operator. Our program incorporates a variety of atomic ground state configurations with total angular momentum ranging from $J=\frac{1}{2}$ to $J = 4$ and a variety of laser beam configurations including the two-beam lin⊥lin configuration, the four-beam umbrella configuration, and four beams propagating in two orthogonal planes. In addition to visualizing the lattice, the program also evaluates lattice parameters such as the vibrational frequency for atoms trapped deep in the wells.
VISUALIZATION OF 3D OPTICAL LATTICES AND GRAPHICAL USER INTERFACE
SOFTWARE DEVELOPMENT

Master’s Thesis

Submitted to the

Faculty of Miami University

in partial fulfilment of

the requirements for the degree of

Master of Science

by

Hoseong Asher Lee

Miami University

Oxford, Ohio

2016

Advisor: Dr. James Clemens

Reader: Dr. Samir Bali

Reader: Dr. Perry Rice

Reader: Dr. Khalid Eid

©2016 Hoseong Asher Lee
This Master’s Thesis titled

VISUALIZATION OF 3D OPTICAL LATTICES AND GRAPHICAL USER INTERFACE SIMULATION SOFTWARE DEVELOPMENT

by

Hoseong Asher Lee

has been approved for publication by

College of Arts & Science

and

Department of Physics

______________________________
Dr. James Clemens

______________________________
Dr. Samir Bali

______________________________
Dr. Perry Rice

______________________________
Dr. Khalid Eid
### Table of Contents

1. **INTRODUCTION** .......................................................................................................................... 1  
   1. A. DOPPLER COOLING ............................................................................................................... 2  
   1. B. SISYPHUS COOLING AND OPTICAL PUMPING MECHANISMS .......................... 3  
   1. C. APPLICATION OF OPTICAL LATTICE: QUANTUM COMPUTING .................. 5  
      1. C. I. WHAT IS QUANTUM COMPUTING? ................................................................. 5  
      1. C. II. THE DRAWBACK OF REALIZATION OF QUANTUM COMPUTING  
      AND HOW TO OVERCOME WITH OPTICAL LATTICES. ..................................... 6  
   1. D. APPLICATION OF OPTICAL LATTICE: BROWNIAN RATCHET ....................... 7  
      1. D. I. WHAT IS BROWNIAN RATCHET? ................................................................. 7  
      1. D. II. BROWNIAN RATCHET IN OPTICAL LATTICES ........................................ 8  

2. **CONSTRUCTION OF OPTICAL LATTICES** ................................................................................. 10  
   2. A. ATOMIC TRANSITIONS ...................................................................................................... 10  
   2. B. 3D LASER CONFIGURATIONS ....................................................................................... 10  
      2. B. I. STANDARD TETRAHEDRON CONFIGURATION ........................................... 10  
      2. B. II. UMBRELLA-LIKE CONFIGURATION ............................................................ 12  
   2. C. EFFECTIVE HAMILTONIAN AND DIPOLE MOMENT: WIGNER-ECKART  
   THEOREM ................................................................................................................................... 14  

3. **GRAPHICAL USER INTERFACE SOFTWARE** ........................................................................... 16  
   3. A. STRUCTURE OF SOFTWARE ............................................................................................ 17  
   3. B. DIRECTIONS TO THE SOFTWARE ............................................................................... 18  
      3. B. I. DIRECTIONS TO SETTING UP PARAMETERS ............................................... 19  
      3. B. I. 1. DIRECTIONS TO SETTING UP PARAMETERS:  
      STANDARD TETRAHEDRON WINDOW ................................................................. 19  
      3. B. I. 2. DIRECTIONS TO SETTING UP PARAMETERS:  
      UMBRELLA-LIKE TAB ................................................................................................. 21  
   3. C. THE FUNCTIONS OF THE SOFTWARE AND THE RESULTS ........................................ 22  
      3. C. I. SINGLE PLOTS ........................................................................................................ 22  
      3. C. II. DOUBLE PLOTS .................................................................................................... 24  
      3. C. III. SLIDE IMAGES .................................................................................................... 26  
      3. C. IV. DIPOLE MOMENTS .............................................................................................. 27  
      3. C. V. POLARIZATION CONFIGURATIONS AND WAVE VECTORS .................. 28  
      3. C. VI. VIBRATIONAL FREQUENCIES ......................................................................... 29  

4. **CONCLUSION** ............................................................................................................................. 29  

5. **REFERENCE** ............................................................................................................................. 32
1. INTRODUCTION

It’s been around a hundred years since people started to recognize quantum mechanics as the governing theory of our world instead of Newtonian classical mechanics. Nowadays, quantum mechanics is not a new theory any more. If a person gets interested in quantum mechanics, he or she can go to a library or bookstore and can find many books telling about the theory and its astonishing applications. One of the most innovative and interesting applications of quantum mechanics, of course, is quantum computing. The concept of quantum computing is first proposed by Feynman in 1982 [1]. It’s been long time after the concept proposal, but we still cannot see any significant outcomes in our daily life. One of the major difficulties of building such systems is that it is challenging to hold a quantum system in a pure state for a significantly long time. In the quantum regime particles do not have a well defined physical realization, but instead they have a delocalized wave nature as well. In other words, even though a researcher might place an atom in a specific place with extreme accuracy, he or she cannot guarantee its position at a later time. The particle can tunnel around due to quantum effects. For this reason, the pure quantum applications or systems have seemed to be in the realm of science fiction, and it has not yet impacted the everyday life of the general population.

In spite of such difficulties, in 1975 research groups published a paper, introducing a way of cooling atoms [2]. Since it used the concept of the Doppler effect, the method is now called Doppler Cooling. After Doppler cooling, researchers could have atomic and molecular system trapped and finally utilize the systems for their further research. When describing a quantum scale system, the temperature is an appropriate quantity to quantify the spatial fluctuations of the system. The lowest temperature of the system by Doppler cooling is called the Doppler limit, $T_D$. However, few years later another method to cool atoms below the Doppler limit was proposed in 1989 [3]. In the Doppler cooling, the atomic systems were assumed to be two-level systems, but for a system with more than two levels, the interference between sublevels can play a role to achieve the lower temperature. This new method is called Sisyphus cooling with its limit temperature, the recoil temperature, $T_R$.

The mechanisms of Sisyphus cooling can be represented as a potential energy curve of an atomic system, which is called the optical lattice potential. Thus, the atom trapped by the cooling methods can be thought of as captured in the wells of the periodic optical lattice potential. In our study, the main focus is to understand various configurations and setup of the atomic system and laser system. Secondly, we will study how optical lattices can be formulated knowing the configurations. Finally, we will present our theoretical prediction of the optical lattices corresponding to its system configuration and the graphical user interface simulation software we invented and implemented for our prediction.
1. A. DOPPLER COOLING

Doppler cooling is one of the laser cooling mechanisms to achieve a low temperature. An atomic system would be placed within a counter-propagating laser field system, and its frequencies will be detuned to the red from an atomic resonance. If an atom stays still without any relative motion to the laser beams there will be no net interaction between atom and the lasers. However, if the atom moves toward one of the laser beams, the frequency of the laser beam which the atom sees will be shifted to the blue. Due to this Doppler shift, the atom will absorb a photon, losing the velocity along the direction where the photon came. Then, the atom will emit a photon in a random direction. According to momentum conservation the velocity will increase in the random direction. Thus, there will be two independent processes in Doppler shift. One is the damping or cooling process, and the other is random or heating process. Since the damping force the atom experiences will be proportional to the velocity of atom \( v \), the cooling rate Error can be predicted to be proportional to \( v^2 \). This leads us to the relationship between the kinetic energy of the atom and the cooling rate. However, at low velocity, the heating process proportional to the scattering rate would be independent of the kinetic energy of the atom. Knowing that the temperature limit would be achieved where the cooling effect and the heating process will be balanced [4], the equilibrium temperature of such atomic system in low intensity laser beams with the red detuning will be deduced as [5]

\[
k_B T_E = \frac{h \gamma}{4} \left( 1 + \frac{(2 \Delta \gamma)^2}{2 |\Delta/\gamma|} \right)
\]  (1.1)

where \( T_E \) is the equilibrium temperature and, \( \gamma \) is the natural width, and \( \Delta < 0 \) is the detuning in the two-level atomic system. The Doppler limit is the minimum value of the equilibrium temperature where \( \Delta = -\gamma/2 \). Thus the Doppler limit is

\[
k_B T_D = \frac{\hbar \gamma}{2}
\]  (1.2)

where \( T_D \) is the Doppler limit temperature.
1. B. SISYPHUS COOLING AND OPTICAL PUMPING MECHANISMS

For Sisyphus cooling mechanism, the polarization of the laser fields comes into play to decrease the temperature below the Doppler limit [6]. For example, the counter-propagating laser fields with linear polarization would form the polarization gradient (see Fig. 1). The total polarization of the field will vary according to the position. This will, eventually, give rise to the position-dependent potential energy shift due to the coupling between the atom’s dipole moment and the polarization of the laser field. For an atomic system, the potential energy shift would be determined by the structure of the relevant energy levels of the atomic system. As a simple example, we would employ an atomic system with $J_g = \text{ground state}$ and $J_e = \frac{3}{2}$ excited state. The sublevels of the ground and excited states will be coupled according to the polarization of the laser field.

![Diagram of polarization gradient](image)

Figure 1. The polarization gradient of the 1D linearly polarized laser field system. Along the propagation direction of z-axis, the polarization of the total field varies according to the position z [3].
The potential energy shift due to the atomic laser field coupling of the example is described in Fig. 2. While an atom climbs the potential energy hill, it loses some of its kinetic energy. As soon as it reaches the top of it, the polarization gradient becomes the circular polarization. Due to the unsymmetrical coupling (see Fig. 3), the state of the atom will collapse to another sublevel of the ground states. Losing its kinetic energy, it climbs the hill again and again. The loss of the kinetic energy of the atom will be scaled by the amplitude of the potential energy Error. Thus, for each cycle, the amount of the energy loss by Sisyphus cooling would be

$$\Delta(k_B T_{sis}) \sim U_0$$

(1. 3)

Figure. 2 The potential energy shift of a system with $J_g = 1/2$ to $J_e = 3/2$ transition. While the atom climbs the hill, it is more probable for the state to remain the same. However, when it hits the top, it becomes highly probable to collapse into another sublevel of the ground states [3].
As the energy of the atom disappears due to Sisyphus cooling, the atom’s energy would not be enough to climb the hills any more. Thus, being captured in the well of the potential energy, the atom will experience random spontaneous photon emissions. Due to the momentum conservation law, the atomic energy will be strictly related to the energy gained by the spontaneous emission. This energy is called recoil energy and will be the limit of Sisyphus cooling,

\[ k_B T_R = \frac{\hbar^2 k^2}{2m}. \]  \hspace{1cm} (1.4)

Figure 3 The sublevels of the ground and the excited states \((J_g = 1/2 \text{ and } J_e = 3/2)\) will be coupled by the polarization of the laser field. And the coupling strengths are given with Clebsch-Gordan coefficients [3].

1. C. APPLICATION OF OPTICAL LATTICE: QUANTUM COMPUTING

1. C. I. WHAT IS QUANTUM COMPUTING?

To differentiate the traditional computing from quantum computing, the term classical computing is used frequently. As its name conveys, the classical computing is powered by the laws of classical mechanics, which are decisive and definite. However, in the world of quantum computing, the computing power benefits from the features of superposition and entanglement allowed by quantum mechanics. When distinguishing such two different computing systems from one another, it is helpful to consider the properties of their smallest data units. For classical
computing, a bit is used for its smallest data unit. A bit is a binary digit that is composed of the values of 0 and 1. For example, to represent lowercase alphabet “a”, a bit string 1100001 is used in ASCII (American Standard Code Information Interchange) system. Similarly, a quantum bit is the smallest data unit in quantum computing, which is called frequently qubit. To construct a qubit, instead of sequential combination of 0 and 1, quantum states of $|0\rangle$ and $|1\rangle$ are to be combined, following superposition principle (see Fig. 4). Taking advantage of the superposition principle from quantum mechanics, for a certain computational process, the performance of quantum computing can be boosted exponentially. In other words, if we prepare a state of N qubits, we can compute a calculation $2^N$ faster than with a classical computer.

![Classical Bit vs Qubit](image)

**Figure. 4** Classical bits are either 0 or 1. However, the qubit is a superposition of the quantum state $|0\rangle$ and $|1\rangle$. Bloch spheres are commonly used to represent the superposition of the states of qubits.

**1. C. II. THE DRAWBACK OF REALIZATION OF QUANTUM COMPUTING AND HOW TO OVERCOME WITH OPTICAL LATTICES.**

In spite of such a tremendous advantage of quantum computing, there are still realistic problems that have limited the further development of quantum computing. As mentioned earlier, there are many issues with building an appropriate quantum system in an actual lab environment. The physical quantity “spin” is one of the powerful candidates of qubits. Then, utilizing spins of quantum systems as qubits, each qubit has to be entangled with the others to build huge quantum states. At the same time, the states have to be entangled to the external environment because at the end to determine the states we have to observe it. However,
interaction between the quantum states and the external environment can cause decoherence, which results in the loss of quantum information. Thus, while maintaining quantum entanglement of the qubits, the entanglement between the states and the environment has to be minimized for the stability of the states. In such conditions, optical lattices are a very promising way of building quantum systems for qubits. When the quantum scale atoms are localized in the wells of the optical lattices, the atoms are very stable, providing the long coherence time for the entanglement. In addition to it, the compactness of the optical lattices allow 1000 qubits to reside in a small volume of $(10^2)^3$ where $\lambda$ is the optical wavelength [7].

1. D. APPLICATION OF OPTICAL LATTICE: BROWNIAN RATCHET

1. D. I. WHAT IS BROWNIAN RATCHET?

The history of a Brownian motor dates back to 1962. A physicist, Dr. Richard Feynman, suggested a ratchet, normally called Feynman's ratchet, on a microscopic scale. In his thought experiment, he employed a ratchet connected to a vane with an axle and a pawl that causes a rectified motion from random atomic collisions in a single heat bath (see Fig. 5). His suggestion started from a question about whether such a system can be physically possible or not. At that time, the concept of a ratchet was closer to a perpetual motion machine. The ratchet could not produce an overall net work due to the violation of the second law of thermodynamics [8] in his final analysis. According to the law, no heat engine is allowed to work in a single heat bath. To make the ratchet work, the pawl and the vane are supposed to be in different thermal baths. The modification to Feynman's ratchet became just a miniature version of a classical heat engine whose net work is extracted from the gradient of the thermal field.

Although Feynman's ratchet is proved to be unrealistic, there followed several types of microscopic motors or ratchets of similar design. One of them is the Brownian motor. Brownian motion is the primary source that makes a Brownian motor work. However, in a sense that Brownian motion is an example of a random walk processes, it is difficult to believe that it can produce a rectified motion or dc current via a fluctuation or ac current. Nevertheless, without violating the conservation of energy and the second law of thermodynamics, a Brownian motor could be realized. A protein motor is one example of Brownian motor in nature [9]. One reason the Brownian motor could successfully produce a net rectified motion is that it is in a nonequilibrium state, while Feynman's ratchet was in a thermal equilibrium state. Thus, by applying external fluctuation, we can break the system's equilibrium and it could be possible to make the system produce a directed motion even from a single bath [10]. The big advantage of Brownian motors is that it is not interrupted by environmental thermal noise or fluctuation, but it takes advantage of it to produce a directed motion. Moreover, the power of thermal noise is great
enough to operate molecular motors. On a microscopic scale, the strength of thermal noise power is thought to be 8-9 orders of magnitude greater than the power required to operate molecular motors [11]. Thus, the Brownian motor is one of the powerful candidates for microscopic molecular motor and is applicable to microscopic engineering field, such as nanotechnology, biophysics, and bioengineering fields.

![Diagram of a ratchet and vane system with different thermal baths](image)

**Figure. 5** To avoid the violation of the second law of thermodynamics, the ratchet and the vane have to be separated in different thermal baths. Then, the net work will be extracted from the gradient of temperature field.

1. **D. II. BROWNIAN RATCHET IN OPTICAL LATTICES**

Optical lattices are a great resource that can be used to study atomic or molecular dynamics. One of the reasons is that optical lattices are easily tuned by external parameters [12]. Along with that, since optical lattices provide the symmetrical potential energy with thermal noise fields, the lattices can be a great place to study and realize the Brownian motor systems. Furthermore, it was found that at a certain height of the optical lattices, the Brownian motors can exhibit another type of random motion process, the Levy walk [13].

During the 1920s, a mathematician Paul Levy studied a new random process, the Levy flight, which follows a Cauchy distribution, one type of heavy-tailed distributions. In contrast to a Gaussian distribution, the Cauchy distribution exhibits an infinite second moment. The characteristic of infinite moments is that the process is scale-invariant. In other words, in a perspective of geometry, Levy flight can exhibit fractal property in its trajectories. However, due
to the infinite moments, the Levy flight had a difficulty in being applied to real science. Then, Paul Levy revised a modification of the Levy flight called a Levy walk. In a Levy walk, we consider a displacement after time $t$, while distance from $N$ steps are taken into account in Levy flight [14]. This space-time coupling method could successfully remove the divergent behavior of moments. Then, its second moment is to be revised as

$$<x(t)^2> \sim t^{\mu}, \text{where } 1 < \mu \leq 2$$

In addition to it, another strange feature of Levy walk is that the displacement at time $t$ is dominated by a single step of the walk [15]. It is thought that Levy flight is not a globally happening event in a Brownian motor system, but it happens locally with specific circumstances or parameters. By changing the process from Brownian motion to Levy walk, the system's flight distance is highly increased as expected (See Fig 6). The understanding and manipulation of the Levy walk in a Brownian motor will give us a significant improvement in transportation on the microscopic scale.

Figure. 6 Brownian motion and Levy flight. Basically Brownian motion shows shorter steps when the system is traveling. However, the steps taken by Levy flight is much longer than that of Brownian motion system. If a Brownian motor exhibits the Levy walk, its performance can be highly boosted.
2. CONSTRUCTION OF OPTICAL LATTICES

2. A. ATOMIC TRANSITIONS

In the previous chapter, we have discussed the history and the concepts of atomic cooling by looking at Doppler cooling and Sisyphus cooling procedures. Now, we will mention more detail of the construction of optical lattices from certain atomic species and external laser configurations. To achieve the Sisyphus cooling procedure, as mentioned earlier, the atoms are required to have more than two quantum internal states. In our study, we implement atomic species with ground state total angular momentum \( J_g = \frac{1}{2}, 1, 2, 3, \) and \( 4 \) (excited state total angular momentum \( J_e \) is assumed as \( J_g + 1 \)). In case of \( J_g = 1 \), the total number of ground states will be eight. For the ground state, there will be three sublevels, such as \(|g\,1\,-1>, |g\,1\,0>, \) and \(|g\,1\,-1>\), while for the excited state, five states, such as \(|e\,2\,-2>, |e\,2\,-1>, |e\,2\,0>, |e\,2\,1>, \) and \(|e\,2\,2>\). In general, it is expected to have \((4I+4)\) states for \( J_g = I \) and \( J_e = I+1 \). In an experimental setup, \(^{85}\text{Rb}\) and \(^{87}\text{Rb}\) are mostly used as atoms representing atomic transition \( J_g = 3 \) and \( J_e = 4 \) and \( J_g = 2 \) and \( J_e = 3 \), respectively [16].

It is important to figure out atomic transitions or sublevel structure of atom species because the sublevels eventually will be coupled to the certain polarization of the external laser beams, giving rise to the dipole moment operators.

2. B. 3D LASER CONFIGURATIONS

To finally build the Hamiltonian operator of optical lattices, we are still left with external laser configurations. In our study, the configurations we will cover are two, which are standard and mostly studied in actual lab environment. One is ‘standard tetrahedron’ and another is ‘umbrella-like’.

2. B. I. STANDARD TETRAHEDRON CONFIGURATION

Standard tetrahedron is also referred to as the symmetric beam-splitting configuration. In 3D laser configuration, the minimal number of laser beams is four [6]. Assuming we have 1D counter-propagating laser beam configuration, one beam coming from \(-z\) direction will be split into other two beams in the plane of \(XZ\) with the angle of \(2\theta_x\), while another beam coming from \(+z\) direction will be split into other two beams in the plane of \(YZ\) with the angle of \(2\theta_y\) (see Fig. [16]).
Total electric field, the wave vectors and the polarizations of the laser beam configuration is described below.

\[ E(r) \sim \sum_{j} \hat{e}_j \exp(i(\hat{k} \cdot r)) \]  

(2.1)

\[ \hat{k}_1 = k \sin \theta_x \hat{x} + k \cos \theta_x \hat{z} \]  

(2.2)

\[ \hat{k}_2 = -k \sin \theta_x \hat{x} + k \cos \theta_x \hat{z} \]  

(2.3)

\[ \hat{k}_3 = k \sin \theta_y \hat{y} - k \cos \theta_y \hat{z} \]  

(2.4)

\[ \hat{k}_4 = -k \sin \theta_y \hat{y} - k \cos \theta_y \hat{z} \]  

(2.5)

Figure. 7 The standard tetrahedron configuration is considered as one of the most commonly used in a real lab environment. The configuration can be achieved by dividing each beam in the 1D configuration into
other two beams. One set of beams travels in the plane of XZ and another set of beams travels in the plane of YZ with angle of $2\theta_x$ and $2\theta_y$, respectively.

When describing the polarization of the fields, usually we take care of two cases, the in-plane and the out-of-plane cases. For the in-plane cases, the polarizations of the beams lie in the plane of incidence. However, for the out-of-plane cases, the polarizations of the beams are orthogonal to the plane of incidence.

First, we will describe the polarizations of the out-of-plane case,

$$\hat{e}_1 = \hat{e}_2 = \hat{y}$$ \hspace{1cm} (2.6)

$$\hat{e}_3 = \hat{e}_4 = \hat{x}$$ \hspace{1cm} (2.7)

Secondly, the polarizations of the in-plane case will be as follows.

$$\hat{e}_1 = \cos \theta_x \hat{x} - \sin \theta_x \hat{z}$$ \hspace{1cm} (2.8)

$$\hat{e}_2 = \cos \theta_x \hat{x} + \sin \theta_x \hat{z}$$ \hspace{1cm} (2.9)

$$\hat{e}_3 = \cos \theta_y \hat{y} + \sin \theta_y \hat{z}$$ \hspace{1cm} (2.10)

$$\hat{e}_4 = \cos \theta_y \hat{y} - \sin \theta_y \hat{z}$$ \hspace{1cm} (2.11)

**2. B. II. UMBRELLA-LIKE CONFIGURATION**

Unlike the standard tetrahedron configuration, only one beam is divided into three beams in the umbrella-like configuration. The three beams make the angle $\nu$ with the z axis, while they are far apart to each other with the angle of $2\pi/3$ (see Fig. 8). The example wave vectors and polarization configurations chosen for our study are given below.

$$\hat{k}_1 = -k \sin \nu \hat{x} + k \cos \nu \hat{z}$$ \hspace{1cm} (2.12)

$$\hat{k}_2 = k \sin \nu \sin \frac{\pi}{6} \hat{x} - k \sin \nu \cos \frac{\pi}{6} \hat{y} + k \cos \nu \hat{z}$$ \hspace{1cm} (2.13)

$$\hat{k}_3 = k \sin \nu \sin \frac{\pi}{6} \hat{x} + k \sin \nu \cos \frac{\pi}{6} \hat{y} + k \cos \nu \hat{z}$$ \hspace{1cm} (2.14)
When choosing the polarization of the umbrella-like cases, there is more freedom than the standard tetrahedron cases. Thus, in our case, we assumed that one beam that would be split into three had the polarization of $\hat{y}$, while another one has the polarization of $\hat{x}$. Then, the polarization of the configuration will be as follows.

$$\hat{e}_1 = \hat{y}$$  \hspace{1cm} (2.16)

$$\hat{e}_2 = \cos \frac{\pi}{6} \hat{x} + \sin \frac{\pi}{6} \hat{y}$$  \hspace{1cm} (2.17)

$$\hat{e}_3 = -\sin \frac{\pi}{6} \hat{y} + \cos \frac{\pi}{6} \hat{z}$$  \hspace{1cm} (2.18)

$$\hat{e}_4 = \hat{x}$$  \hspace{1cm} (2.19)

Figure 8 As another configuration than standard tetrahedron, the umbrella-like configuration is also studied in many lab environments. In the umbrella-like case, only a beam is divided into other three beams. The three beams show the angle of $\nu$ to the z axis and they are far apart to each other with the angle of $2\pi/3$. 
2. C. EFFECTIVE HAMILTONIAN AND DIPOLE MOMENT: WIGNER-ECKART THEOREM

In our derivation, we have assumed that our laser system is under the condition of low intensity and large detuning. Then we can safely assume that the saturation of the atoms will be low and the excited state populations will be considerably less than the ground states. In accordance with such condition, we can divide time regime into two. One is spontaneous emission time and another is optical pumping time. Since in our case the saturation is low, the optical pumping time will be comparably longer than the spontaneous emission time. Then, the excited states populations will follow ground state sublevels adiabatically because due to the low saturation, the excited states populations and the coherence between ground states and excited states will relax quickly. Thus, the variables depending on the excited states can be adiabatically eliminated. Along with the adiabatic elimination, the effective Hamiltonian will depend only on the atomic external coordinates and the internal ground sublevels [17].

\[
\hat{H}_{\text{eff}} = U_o \left( E(r) \cdot \hat{d} \right) \left( E(r) \cdot \hat{d} \right)^\dagger 
\]

\[
\hat{d} = \sum_{m_g,q=-1,0,1} C_{m_g}^{m_{g}+q} |e; J_e, m_g + q g; J_g, m_g \rangle \langle \epsilon_q |. 
\]

\(E(r)\) is the electric field from laser beams at position vector \(r\). And \(\hat{d}\) is the laser-induced dipole moment, while \(C_{m_g}^{m_{g}+q}\) represents the Clebsch-Gordan coefficients. The parameter \(q\) represents the index of circular polarization of external beam fields such that 0 represents linear polarization, -1 left-circular polarization, and 1, right-circular polarization. Then, \(\epsilon_q\) will be the polarization of the external laser fields with the index of \(q\) such that \(\epsilon_0 = \hat{z}\), \(\epsilon_{-1} = \frac{\hat{x} - i \hat{y}}{\sqrt{2}}\), and \(\epsilon_{+1} = \frac{\hat{x} + i \hat{y}}{\sqrt{2}}\).

One might wonder why the effective Hamiltonian is in the form of Eqn. (2.20) instead of \(\hat{H}_{\text{eff}} = -E(r) \cdot \hat{d}\), a general form of effective Hamiltonian due to interaction between dipole moment and external electric fields. This results can be explained according to perturbation theory in quantum mechanics. Using the theory, one can obtain the energy from the effective Hamiltonian by a series of energy corrections as described below.

\[
E_{\text{eff}} = E_{n1} + E_{n2} + E_{n3} + \ldots 
\]
The first order correction to n-th energy of the original Hamiltonian, $E_{n1}$, will be omitted because the diagonal components of the dipole moments are zeros. Then, we can assume that $E_{eff} \approx E_{n2}$. According to quantum mechanics, the second order correction is as below.

$$E_{n2} \sim \sum_{n \neq k} |\langle \psi_n | \hat{H}_{eff} | \psi_k \rangle|^2 \quad (2.23)$$

$\psi_n$ is n-th eigenstate of the original Hamiltonian of the system. For simplicity of the derivation, we will assume two level systems in this case. Then, the Eqn. (2.23) will be narrowed down into

$$E_{g2} \sim |\langle \psi_g | \hat{H}_{eff} | \psi_e \rangle|^2 \quad (2.24)$$

$$= \langle \psi_g | \hat{H}_{eff} | \psi_e \rangle \langle \psi_g | \hat{H}_{eff} | \psi_e \rangle^* \quad (2.25)$$

$$= \langle \psi_g | \hat{H}_{eff} | \psi_e \rangle \langle \psi_e | \psi_g \rangle + \langle \psi_g | \hat{H}_{eff} | \psi_g \rangle \langle \psi_g | \hat{H}_{eff}^\dagger | \psi_g \rangle \quad (2.26)$$

$$= \langle \psi_g | \hat{H}_{eff} \star \hat{H}_{eff}^\dagger | \psi_g \rangle \quad (2.27)$$

On the other hand, without perturbation theory, a ground energy from the effective Hamiltonian in two level system in general

$$E_{eff, g} = \langle \psi_g | \hat{H}_{eff} | \psi_g \rangle \quad (2.28)$$

Utilizing Eqn. (2.20), Eqn. (2.27), and Eqn. (2.28), one can deduce

$$\hat{H}_{eff} \sim (E(r) \cdot \hat{d})^\dagger (E(r) \cdot \hat{d}) \quad (2.29)$$

In addition to it, it is reasonable to raise a question because of the presence of the Clebsch-Gordan coefficients in the dipole moment. Basically, the Clebsch-Gordan coefficients come from the mathematical connection between the coefficients and the strength of coupling between ground sublevels and excited sublevels due to the polarization of the external laser beams. This mathematical result is famous in quantum mechanics and called the Wigner-Eckart theorem. The theorem says,

$$\langle \alpha', j', m' | \hat{T}_q^{(k)} | \alpha, j, m \rangle = \langle j, m; k, q | j', m' \rangle \frac{\langle a' | \hat{T}_q^{(k)} | a \rangle}{\sqrt{2j'+1}} \quad (2.30)$$

Here, $k$ is the rank of the tensor operator $\hat{T}_q^{(k)}$ and $q = -k, -k+1, \ldots, k-1, k$. Additionally, $m' = m + q$ and $| j - k | \leq j' \leq | j + k |$. Thus, the coefficients $C_{m_q}^{m_q+q}$ at Eqn. (2.21) describes the
Clebsch-Gordan coefficients for the addition of $|j, m>$ and $|1, q=-1, 0, 1>$. Because the dipole moment operator is the vector operator, the tensor rank for this case is one and $k = 1$. Utilizing Wigner-Eckart theorem, one can easily figure out the coupling strengths between sublevels. Usually, the Clebsch-Gordan coefficients are provided in a diagram for convenience. Following such tradition, the example of the coupling strengths of the atomic transition $J_g = 2$ to $J_e = 3$ is depicted in Fig. 9.

Figure 9 The squared Clebsch-Gordan coefficients for the atomic system with $J_g = 2$ to $J_e = 3$ transition. The bold solid line represents the right-circular polarization, while the dotted line linear and plain solid line left-circular polarization.

3. GRAPHICAL USER INTERFACE SOFTWARE

From the beginning of Chapter 2, we have covered how the optical lattices can be formulated and how to construct the Hamiltonian operator from the information of atomic systems and external laser configurations. In the perspective of experimentalists, it is very important to predict results before they begin their experiments in actual lab environment. In real labs, any slight changes or mistakes on experiment set-up could cause the huge difference in the result. Thus, we have built software to guide researchers who work on realization of optical lattices in their labs. In addition to it, we have added graphical user interface (GUI) part to the software, utilizing TkInter package as a main framework of GUI part. We wish the GUI part of the software provides a great convenience to the experimentalists, letting them only focus on their experiments not on studying programming languages or skills.

There are many languages for scientific purpose such as matlab, mathematica or Julia. However, the main reasons that made us to choose Python for main language for the software development are its simplicity to learn, abundance of scientific computing packages, and its amazing popularity.
Python is famous and known as its simplicity and briefness of syntax. As mentioned earlier, there are many science and engineering programs that provides Python classes to their students in their curriculum. Thus, Python became one of the most familiar scientific languages to college students. This trends falls into our philosophy in developing our software and made us to choose Python as main language without any doubts. Secondly, there are a number of scientific packages provided in Python. This not only save our effort and time for development but also a user of the software can easily modify and understand the software.

3. A. STRUCTURE OF SOFTWARE

```
-- (1) optlattice.py       -- (5) figures
-- (2) module             ---- (5-a) etc
  ---- (2-a) classes.py
  ---- (2-b) configurations.py ---- (5-b) tetrahedron
  ---- (2-c) functions.py      ---- (5-b-i) xy
  ---- (2-d) gui_functions.py ---- (5-b-ii) xz
  ---- (2-e) gui_tetra.py     ---- (5-c) umbrella
  ---- (2-f) gui_umbrella.py  ---- (5-c-i) xy
  ---- (2-g) mathematics.py   ---- (5-c-ii) xz
-- (3) ref                ---- (5-c-iii) yz
-- (4) README.md
```

Figure. 10 This figure illustrates the directory structure of the software when downloaded via Github repository. It is recommended to understand the structure and if a user wants more advanced usage of the software, such as custom polarization configuration setting for umbrella-like case. For standard tetrahedron configuration, we have implemented a function which let a user can specify his or her own custom polarization configuration to some degree. However, since umbrella-like configuration contains a great deal of freedom, we omitted a custom polarization configuration function. Thus, to set up customized polarization of umbrella-like case, it is highly recommended to modify the software source code.
Even though the software contains the GUI part for convenient usage, still it is necessary for a user to understand the structure of the software for more advanced usage of the software (see Fig. 10). Thus, in this section, I will briefly talk about the software’s directory structure and a brief introduction to the directories. If a user downloads the software from Github repository, a user will have a directory named ‘3d-optical-lattice’. Then, inside the directory, all the files are included. The structure is drawn in Fig. 11. First of all, a user can find an executable python file named optlattice.py. By typing the command “python optlattice.py” in a terminal, a user can run the software. Secondly, for efficient and maintainable coding, most of functionalities and mathematics formula are put in separate python script files under the directory module. Thus, if a user wants to directly change or add some functions into the software, module directory must be the right choice to start. Thirdly, there is a directory named ref. We have included all of our reference papers that we looked up during our research. Fourthly, README.md is a markdown text file that works as an instructional page in the github repository. I personally added the detailed installation guide into the file for users. Fifthly, there is a directory called figures. One of the function of the software is to save slide images and user can see the slices of the 3d optical lattices. Then all the slide images will be saved in the directories of xy, xz, and yz under the figures directory.

3. B. DIRECTIONS TO THE SOFTWARE

Figure. 11 When a user starts the software, a basic window will open. And a user can choose one of three options, Standard Tetrahedron, Umbrella, and Quit. The buttons of Standard Tetrahedron and Umbrella will let a user have another window that provides a simulation environment for both standard tetrahedron configuration and umbrella-like configuration, respectively. The button of Quit will let the program be terminated.
Once the software begins, the users can see three buttons with different functions (see Fig. 11). First, if a user hits the button of Standard Tetrahedron, another window will be present. In the window, the user can set up an optical lattice with a variety of atomic system choices in standard tetrahedron laser configurations. Next, the button of Umbrella will pop up another window to set up optical lattices of a variety of atomic systems in Umbrella-like laser configurations. And the button of Quit will close the window and terminate the software.

3. B. I. DIRECTIONS TO SETTING UP PARAMETERS

Assuming that a user hits one of the buttons between Standard Tetrahedron and Umbrella, as mentioned just above, another window will be open. In the tab, mainly there are input parameter sections in which users have to specify their parameters, such as atomic transition, polarization configurations, and etc. For most of part, Standard Tetrahedron and Umbrella tabs share some parameters. However, there are also few different things in setting up the parameters. In this section, we will cover how to set up parameters, starting with standard tetrahedron case.

3. B. I. 1. DIRECTIONS TO SETTING UP PARAMETERS: STANDARD TETRAHEDRON WINDOW

![Image of standard tetrahedron window with annotations]

Figure. 12 In the standard tetrahedron window, mainly users are allowed to set up parameters ranging from atomic transitions to plot resolution. All parameters that we will cover are labeled with red-colored numbers in circles.
In Fig. 12, parameters that we will talk about are labeled with red-colored numbers in circles. Thus, to avoid confusion I will use its number to address corresponding parameters.

(1) One of the parameters that users will need to choose first will be atomic transition. In the most left and top of the standard tetrahedron tab, there is an input box for atomic transition, which is next to the label ‘Transition’.

(2) Below the atomic transition part, there are two input boxes labeled with ‘Theta_x’ and ‘Theta_y’. These two parameters refer to the angles between split beams in XZ plane and YZ plane, respectively (see Fig. 7).

(3) At the middle and the left part of the tab, there is a parameter input box for choosing laser configuration. Though it has drop down box, the only option a user can choose is one, standard tetrahedron case. We left this input box for showing which configuration a user is working on.

(4) Next, the user can choose the polarization configuration parameter. When plotting optical lattices, this parameter is one of the important factors that change the shape of the lattices. At first, the parameter is set to out-of-plane. This means that the polarization of the laser field are not in the plane of incidence, but perpendicular to it. The configuration is the same as Eqn. (2.6) and Eqn. (2.7). For other options, there are two, in-plane and custom. If a user choose in-plane for his or her option, the polarization configuration would then now lie in the plane of incidence and will be exactly the same as Eqn. (2.8), (2.9), (2.10), and (2.11). And if the option custom is selected, a user can choose their customized polarization configurations. Basically, even though Eqn. (2.6) and Eqn (2.7) describe the polarization that are out of the plane of incidence, there are still other freedom to choose other configuration that are still perpendicular to the plane. This option is added to satisfy researchers who want to plot optical lattices with their own taste of polarization configurations. The custom polarization can be set up separately and we will cover it later.

(5) Below the laser configuration part, a user can find an input box for scale. The scale here is not a physical quantity and does not directly affect to the shape of optical lattices. However, it sets the size of plots for optical lattices. For convenience, when plotting all optical lattices, the wavelengths of external laser fields are all normalized to one. Then, if a user sets the scale to one, each axis of plots will be from negative one to positive one.

(6) At the bottom of the scale parameter, there is a dropbox for resolution. Just like the scale parameter, the resolution parameter is not a physical quantity but is a parameter for
plotting optical lattices. There are three options for the resolution, low, medium, and high. If a user selects the low option and plot optical lattices, the resolution of the optical lattices will be low or poor. However, if the high option is selected, the resolution of the lattices will be much higher and more clear to see. The reason for this parameter is that drawing optical lattices takes time. Even though the low option gives us low resolution of optical lattices, we can get the lattices within around 20 seconds. However, for the high option, the optical lattices will be plotted in around 30 seconds to 1 minute.

(7) As we have covered earlier, a user can choose customized polarization configurations. On the right part of the tab, there are four vectors named $e_1$, $e_2$, $e_3$, and $e_4$ (The subscript number indicates the laser beams labeled with the same number in Fig. 7). And next to the vectors, there is a part that a user can specify its x, y, and z components.

(8) As we will talk about later, the software has a function to plot slide images of optical lattices. The parameters, ‘Number of slices’, ‘Range’, and ‘to’, are the parameters that are related to the slide function. First, if a user set ‘Number of slices’ parameter to 10, the user will eventually see 10 slices at the end. Secondly, ‘Range’ and ‘to’ parameters set the starting point and the end point for slide plots. For example, assuming a user set ‘Number of slices’, ‘Range’, and ‘to’ to 10, 0, and 1, respectively, if a user want XY slide plots, there will be 10 optical lattice plots in XY with the Z value ranging from 0 to 1.

3. B. I. 2 DIRECTIONS TO SETTING UP PARAMETERS:
UMBRELLA-LIKE TAB

Most of parameters are shared between the standard tetrahedron window and the umbrella-like tab. However, there are some differences in choosing parameters on the umbrella-like tab.

(1) First of all, since the umbrella-like configuration is achieved by asymmetric beam-splitting, the angle parameter that will choose the shape of the optical lattices (see Fig. 8) is different. The angle parameter is now called ‘mu’.

(2) There is an input part named Laser Configurations. It has only one option, but it works to show which configuration a user is now using on.

(3) Lastly, Unlike standard tetrahedron case, in this window, we have only one option called ‘standard’ for polarization configuration parameter. Since in umbrella-like case, there is a great deal of freedom, we didn’t make the GUI part for choosing the polarization configuration. However, if a user wants to specify custom polarization,
he/she still can do it by setting the custom polarization in the file `configurations.py` under the `module` directory.

![Figure 13 Umbrella-like tab. There are some different parameters that a user will have to specify before running the software. The parameters that we will explain are labeled with red-colored numbers in circles.]

3. C. THE FUNCTIONS OF THE SOFTWARE AND THE RESULTS

Once all the parameters are set up properly, a user can now run the software to plot optical lattices. In this section, we will provide example results of each function, ranging from single plots to vibrational frequencies.

3. C. I. SINGLE PLOTS

It is important to predict the minimum wells of an optical lattice because the wells will be the spots where atoms or molecules will be stored. When it comes to 3D optical lattices rather than 1D or 2D, finding the minimum wells is not an easy task because there the data size increases exponentially by the order of dimension size. To overcome such difficulty, we have adopted an optimization algorithm from Scipy. By using the algorithm, we could comprehensively decrease the simulation time to detect the global minimum points from a huge set of 3D data points of optical lattices. For single plots, there are three options, Plot XY, Plot, YZ, and Plot XZ. Each option will draw optical lattices on the corresponding plane. For example, if a use hit the button Plot XY, the result of an optical lattice will be provided in XY plane. The example results for single plot function is presented in Fig. 14 and 15.
Figure 14 The optical lattice represents the optical lattice with $J_g = \frac{1}{2}$ to $J_e = 3/2$ transition and $\theta_x = \theta_y = 60.00^\circ$ in standard tetrahedron configuration with out-of-plane polarization. The 3D geometry of the optical lattice can be interpreted by referring the colors of the contour plot. The region with the bluer color represents the region with the lower value of optical lattice potential energy, and vice versa. On the right-side of the plot, there is a bar that describes the quantitative meaning of the colors. In this software, for convenience, we have normalized some physical quantities, such as Hamiltonian, wavelengths.
Figure 15 The optical lattice represents the optical lattice with $J_g = \frac{1}{2}$ to $J_e = 3/2$ transition and $\mu = 60.00^\circ$ in umbrella-like configuration. The minimum wells of the plot matches also the global minimum wells of the entire 3D optical lattices.

### 3. C. II. DOUBLE PLOTS

Single plot can be a good function to investigate the shape of an optical lattice. However, since it shows only a cross-sectional plot of an entire optical lattice, it is somewhat difficult to understand the geometry of optical lattice entirely. For better understanding of the 3D geometry of optical lattices, we have implemented double plot function. Utilizing the function, a user will be provided with double cross-sectional plots of an optical lattices. There are three options for
the function, Plot XY&XZ, Plot XY&YZ, and Plot YZ&XZ. Thus, if a user hit the button Plot XY&XZ, the result of an optical lattice will be provided with cross-sectional images on both XY plane and XZ plane. The example results are given in Figure 16 and 17.

Figure. 16 The optical lattice represents the optical lattice with $J_g = 2$ to $J_e = 3$ transition and $\theta_x = \theta_y = 30.00^\circ$ in standard tetrahedron configuration with in-plane polarization. Double cross-sectional plots help users understand the 3D shape of the optical lattice.

Figure. 17 The optical lattice represents the optical lattice with $J_g = 4$ to $J_e = 5$ transition and $\mu = 23.00^\circ$ in umbrella-like configuration. Double cross-sectional plots help users understand the 3D shape of the optical lattice.
3. C. III. SLIDE IMAGES

Figure. 18. The slide images represent the cross-sectional images of an optical lattice on XY planes with Z ranging from 0.0 to 0.9. The optical lattice composed of atomic system with $J_g = 2$ to $J_e = 3$ transition and $\theta_x = \theta_y = 54.70^\circ$ in standard tetrahedron configuration with out-of-plane polarization.

Though single plot and double plot functions are enough to generalize the shape of the 3D optical lattices and enough to predict the global minimum wells, it may still be useful to obtain slide images of 3D optical lattices in certain range. In accordance with such necessity, we have added a function of slide images. When a user sets up the relevant parameters for the slide image function, the function will save the slide images on a designated directory. In Fig. 18, the example of slide image results is presented.
3. C. IV. DIPOLE MOMENTS

Figure. 19. Dipole moment operator of an optical lattices with atomic transition \( J_g = 1/2 \) to \( J_e = 3/2 \) transition and \( \theta_x = \theta_y = 60.00^\circ \) in standard tetrahedron configuration with out-of-plane polarization.

Our philosophy when we were building the software was to make all steps to obtain optical lattices to be calculated by the software, decreasing the calculation part in which a user should be involved. Thus, the software also provides dipole moment operators for each optical lattices, which is one of the important factors when constructing Hamiltonian operators.
Fig. 19 represents one example result obtained by using dipole moment operator function. The function implemented Eqn. (2.21) to calculate the dipole moment operators of the lattices. The size of matrices is the same as the number of possible sublevels of the system. Thus, in this case we have obtained 6x6 matrices. I set the order of rows and columns of the matrices according to the descending order of excited sublevels and ground sublevels. In other words, each cell of rows and columns of the matrices in this case represents the states \(|e\ 3/2\ 3/2>, |e\ 3/2\ 1/2>, |e\ 3/2\ -1/2>, |e\ 3/2\ -3/2>, |g\ 1/2\ 1/2>, and |g\ 1/2\ -1/2>.\]

3. C. V. POLARIZATION CONFIGURATIONS AND WAVE VECTORS

In addition to dipole moment operators, we have built functions that provide users with wave vectors and polarization vectors on a corresponding optical lattice. As mentioned earlier, one should consider unit of the vectors in such a way that a wavelength of laser beam fields has been normalized. The examples of polarization vectors and wave vectors are given in Fig. 20 and 21, respectively.

![Figure. 20 The polarization vectors of an optical lattice with atomic transition of \(J_g = 1/2\) to \(J_e = 3/2\) transition and \(\theta_x = \theta_y = 60.00^\circ\) in standard tetrahedron configuration with in-plane polarization.](image)

![Figure. 21 The wave vectors of an optical lattice with atomic transition of \(J_g = 1/2\) to \(J_e = 3/2\) transition and \(\theta_x = \theta_y = 60.00^\circ\) in standard tetrahedron configuration with in-plane polarization.](image)
3. C. VI. VIBRATIONAL FREQUENCIES

Figure. 22. The vibrational frequencies of an optical lattice with atomic transition of $J_i = 1/2$ to $J_e = 3/2$ transition and $\theta_x = \theta_y = 60.00^\circ$ in standard tetrahedron configuration with in-plane polarization. The frequencies are obtained by calculating the second derivatives of the optical lattice potential energy at the global minima points. In finding global minima, we have implemented Scipy’s optimize() function and for solving second derivatives, Numpy’s gradient() function has been employed.

The physical quantity vibrational frequency is one of the important physical quantities in characterizing optical lattices. In our software, we have built a function that solving the vibration frequencies by obtaining the second derivatives of the global minima of the optical lattices. We have implemented Scipy’s optimize() function and Numpy’s gradient() function. The former let the software determining the global minima and the latter the second derivatives at the minima.

4. CONCLUSION

Optical lattice research is definitely one of the promising research topics in the field of physics. I, personally, came up with optical lattice while I studied on Brownian motors and Levy flights. Obviously, optical lattice is formulated by quantum mechanical interaction between an atom’s internal structure and external beam fields. However, after then it is far more closer to semiclassical regime than pure quantum regime. Only the quantum effect that drives the atoms under optical lattice random process is quantum tunnelling. One very interesting fact is that Levy flights, which is considered to be nonlinear or chaotic behaviors happens by the pure random process, quantum tunnelling. While I was taking a nonlinear dynamics class, I studied that though chaotic behavior resembles random or stochastic behavior, they cannot be treated as the same, but strictly different effect in nature. In that perspective, I believe along with other researches in optical lattice it will be one of the most important and promising topics to reason why and how the nonlinear behavior comes from the pure quantum or random effect.
Unfortunately, during my master’s research I wasn’t able to reach the answer for that question. Rather than that, I decided to build a simulation software for 3D visualization of optical lattices. The optical lattice research actually was proposed by a faculty member at Miami University, Dr. Samir Bali. He was at the same time my research collaborator and thesis committee. When I was talking to him about the research, he implied that he needs a simulation tool to compare their experimental results with theoretical results after he finishes the preparation for experimental set-up for optical lattices. In addition to it, he wanted the software to contain GUI part to let experimental physics students and researchers easily manipulate it. According to his suggestions, I built the GUI software with a philosophy of user-friendliness.

For the main language of the software, I employed Python. One of the reasons is that Python language is very popular in the field of science and a number of students have an experience with Python during their college time. Even though, I implemented GUI to the software, I strongly believe there will be a high chance for a user to go through the source file and modify for his or her own purpose. For example, though I have included GUI part where a user can specify custom polarization to some degree, the freedom in choosing polarization goes beyond the capability of the current software. Thus, for more delicate choice of polarization, one is recommended to go through and touch source files and modify them for his or her own purpose. In terms of that, Python language would be the most appropriate choice for the software’s framework because of its familiarity to science students and easiness to learn.

Another advantage of using Python is that there are many scientific libraries written in Python, such as Numpy, Scipy, or QuTiP. It will be difficult for one to make all the algorithms for quantum mechanical calculation or mathematical optimization. However, by using Python I was able to save my efforts and time for implementing the algorithms already written and packaged into a library. For example, at the end of the optical lattice calculation, the software returns a great size of matrix, for example 100x100. And even worse, since we deals with 3D optical lattices, there becomes a way more calculations to handle. However, implementing Scipy’s optimize() function, we could easily find a minimal point in 3D and increase the performance of the software.

Along with a philosophy of user-friendliness, I tried to build a software with a philosophy of object-oriented programming (OOP). Rather than counting on functions when we need to do some specific calculations, I defined many objects such as Hamiltonians and plots. Thus, a few lines are just enough for plotting optical lattices once all parameters are passed correctly. For an advanced user who has good programming background, it will be much more easy and efficient to plot lattices from coding based on modules included in the software. As mentioned above, all the objects and functions are packaged and stored in module directory of the software. I hope the OOP of my software make a user can strip out the codes he or she wants and use anywhere else for his or her own purpose.

In addition to it, I published the software to Github repository so that anybody can reach to the software and contribute to version-up or upgrade to the software. Now, one weak aspect of the
software is that in umbrella-like configuration, the plots shows us strange shapes of optical lattice, while standard tetrahedron configuration shows well-shaped and well-like optical lattices. We assume that for umbrella-like configurations, the rotation of axis might be required for us to obtain better resolution and better-looking shape of the optical lattices. I did not implement the function to rotate the axis of the optical lattices. However, in any future not only I but also any other users over the world can contribute to the addition of such function to the software.
5. REFERENCE