Strong-Field Laser Physics Modeling

A thesis submitted in partial fulfillment of the requirement
for the degree of Bachelor of Science with Honors in
Physics from the College of William and Mary in Virginia,

by

Adam Holman

Accepted for ________________________
(Honors)

____
Advisor: Dr. Chaloupka

____
Dr. Carone

____
Dr. Sher

____
Dr. Lindberg

Williamsburg, Virginia
May 2005

Abstract

Visual simulations of atomic processes in the presence of a strong laser field are not easily found or available to study. Developing intuitive, interactive, and realistic physical models of the laser-atom interaction is a crucial step in expanding knowledge and understanding. The visualization of electron trajectory and velocity due to the superposition of user-controlled electric, magnetic, and Coulomb fields further enhances both the qualitative and quantitative data obtained.

Acknowledgements

I would first like to thank my advisor, Dr. Jan Chaloupka, for his support and patience with me on this project. He has been a lot more than just a professor to me these last few years and I have learned a great deal from him. Thank you to my family for their love and encouragement and to my friends who have always been there to help when I needed it. Finally, I would like to thank Capital One for their generous employment offer and opportunity next year.

Contents

1 Introduction and Motivation

Atoms in strong laser fields demonstrate unusual phenomena. A fundamental example of this is the ionization of atoms with large work functions using photons of very low energy. In 1902, Philipp Lenard studied how the energy of the emitted photoelectrons varied with the intensity of the light and found that the intensity of the incident light had no effect on the maximum kinetic energy of the emitted electrons [1]. In 1905, Einstein, in an attempt to explain Lenard's interesting results, postulated that light could be thought of as particles (photons), or discrete packets of energy [2]. Each electron must then be ejected by a single photon striking the metal's surface. Quantum mechanically, the energy of the photon is determined by its frequency,

$$E = h\nu \tag{1}$$

where h is Planck's constant ($h = 6.626069 \cdot 10^{-34} J \cdot s$). Figure 1 shows that the kinetic energy of the emitted electron is given by the energy of the photon minus the energy needed to release the electron from the surface (the metal's work function, φ), or

$$KE = h\nu - \varphi \tag{2}$$

This energy thus depends on the frequency of light striking the surface, but not on its intensity. Higher intensity light has more photons, and so will free more electrons. However, if the frequency of the light is such that a single photon is not energetic enough to release an electron from the surface, then none will be ejected no matter how intense the light. However, if the frequency of the light is such that a single photon is not energetic enough to release an electron from the surface, then none will be ejected no matter how intense the light.

This phenomenon could not be understood without the concept of a photon, a discrete amount of light energy for a particular frequency. If light were simply a wave-like phenomenon, as previously thought, then increasing the intensity (and total energy) hitting the surface would eventually provide enough energy to release electrons regardless of the frequency. Additionally, in the classical picture one would expect the energy of the emitted electrons to depend on the intensity of the light, but it does not.

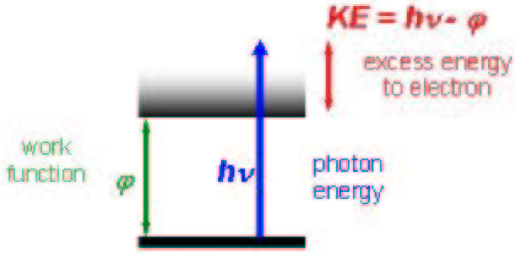


Figure 1: Diagram of Photoelectric effect. Traditionally, an electron is only ionized if the photon energy is greater than the work function, and the difference in energy is given to the kinetic energy of the electron.

With today's powerful and short pulse-width lasers, light again displays some unusual properties. Contrary to the original description of the photoelectric effect, an atom actually can absorb many photons at once through so-called virtual levels as shown in Figure 2 [3]. The absorption of many photons at once is very unlikely, but at very large photon densities it is nonetheless possible. Sunlight reaches Earth with a photon density of $2 \cdot 10^7$ photons per cubic centimeter, while a typical intense laser has a photon density of $4 \cdot 10^{24}$ photons per cubic centimeter [4]. That is a difference of over seventeen orders of magnitude, or one hundred thousand trillion times more photons for a given unit of volume.

An alternate view of strong-field ionization makes use of a classical treatment of the laser field. In this semi-classical picture the atomic Coulomb potential is distorted

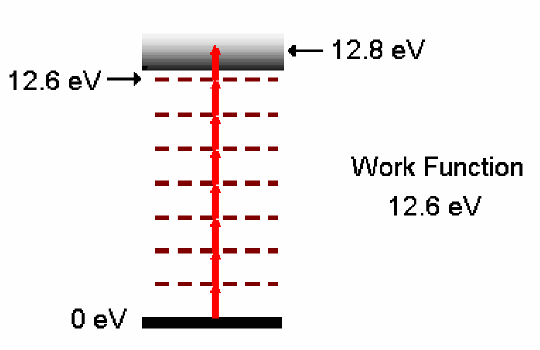


Figure 2: Visualization of Virtual Levels. Eight red photons (780nm) combine to produce the effect of one much larger photon ($8 \cdot 1.6$eV = 12.8 eV), which is greater than the work function of 12.6 eV, allowing for ionization to occur.

by the laser field leading to tunnel ionization [5]. Extensions of this simple single-ionization picture to multiple ionization underestimate the observed yields by many orders of magnitude. It is clear then, that a more complicated process must actually be taking place. Various theoretical models have been proposed to better match calculation with experiment. The current model with the most promise is rescattering. Recent experimental efforts [6] have given strong support to the rescattering model [7]. Rescattering involves tunnel ionization, propagation of the electron in the external electric field, and then collision of the electron back with the parent ion, resulting in impact ionization and liberation of a second electron. This results in a dramatic enhancement in the double ionization yield beyond that predicted by single-active-electron models [8].

Creating interactive Java applets to model strong field laser processes provides a controllable and realistic approach to the visualization of the forces at work. This allows the user to pinpoint the important attributes both intuitively and by trial and error. The output data can then be written to a file for graphical analysis of electron trajectories potentially resulting in new insight and further studies.

2 Background Theory

2.1 Tunnel Ionization

Tunnel ionization is a semi-classical approach for describing atomic ionization in which there is a Coulomb potential in addition to the potential due to an oscillating electric field of a laser pulse. Figure 3 shows the total potential as the superposition of the oscillating laser field potential (at the peak of an optical cycle) and the atomic Coulomb potential. When the electric field of the laser is sufficiently high, ionization can occur through tunneling [5]. It is possible to treat this new potential as a constant for a given amount of time, because the internal time of the electron orbiting is very fast compared to the external time of the oscillation of the laser field. The electron samples that potential many times before it sees any change. This tunneling is very dependent on intensity; as the intensity increases, so does the distortion of the new potential. Since tunneling rates depend very strongly on the width of the barrier this process is highly non-linear with respect to intensity. Observed yields in helium are proportional to the laser peak intensity to the eighth power [8]. The narrower the barrier, the easier and more likely it is for the electron to escape.

2.2 Sequential Ionization

Sequential ionization is described using a single-active-electron model. As the laser pulse evolves, the Coulomb distortion grows due to the increase in the electric field of the laser. Figure 4 shows the electric field as a function of time for a 100-fsec, 800-nm laser pulse. At moderate field strength, the first electron is liberated (point 1),

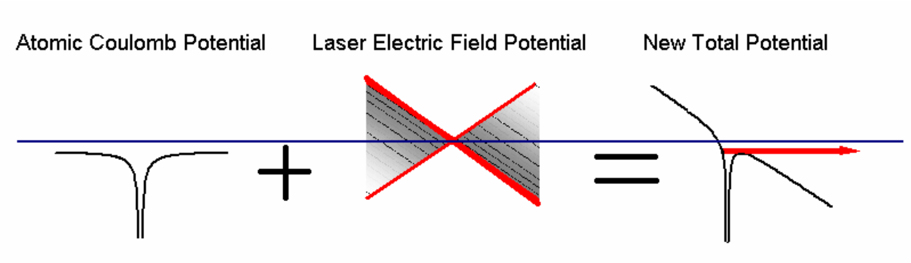


Figure 3: Coulomb potential distortion due to an oscillating potential can result in tunnel ionization. In the above case, the addition of the fields is done at the peak of the optical cycle of the laser field.

then at a later time and higher field strength, the second electron is released through tunnel ionization (point 2). The second electron is more tightly bound following the liberation of the first electron due to a decrease in shielding and thus requires a higher energy to be released. In this description, both ionization processes occur completely independently of one another, or, as viewed from the photon's perspective, it appears that the atom absorbs multiple photons in two separate and independent steps.

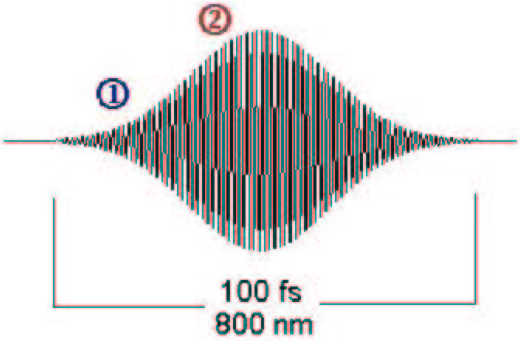


Figure 4: Evolution of the Laser Pulse. In sequential ionization, the first electron is ionized at point 1, and with one electron liberated, the second electron is more tightly bound to the nucleus, only ionizing once the field is at a higher level, at point 2.

2.3 Rescattering

The initial concept of a liberated electron helping to free a bound electron was first suggested by Kuchiev with his atomic antenna model and was cast into the rescattering picture by Corkum in 1993 as a three-step process involving semi-classical theory [5]. First, an electron is liberated through tunnel ionization by methods previously described. The electron then propagates in the combined laser and Coulomb field, with its velocity and trajectory determined primarily by the phase of the laser field at the time of ionization. Finally, depending on that phase, the electron may come back and impact the parent ion, resulting in one of several possibilities. It can be elastically scattered and even further accelerated, or it can be inelastically scattered either liberating an additional electron or exciting the ion to a point where the second electron can be released through field ionization via tunneling. In either case, both electrons end up propagating in the field. Figure 5 provides a visualization of the collision of the electron and the parent ion.

2.4 Observed Yields

Sequential ionization does not predict the observed He^{2+} ion yields. Calculations of single-active-electron ionization greatly underestimate the double-ion yield. Figure 6 shows the dramatic difference between sequential ionization theory (the dark green line) and the actual experimental data (red and blue data points) [8]. The difference between these two lines, labeled 'NS', must be due to a non-sequential ionization process involving some sort of electron-electron correlation.

Even though rescattering works, it has been theoretically calculated almost exclusively using a plane wave. In reality, one does not do experiments with plane waves; experiments are done using highly focused lasers. The typical expression for a focused

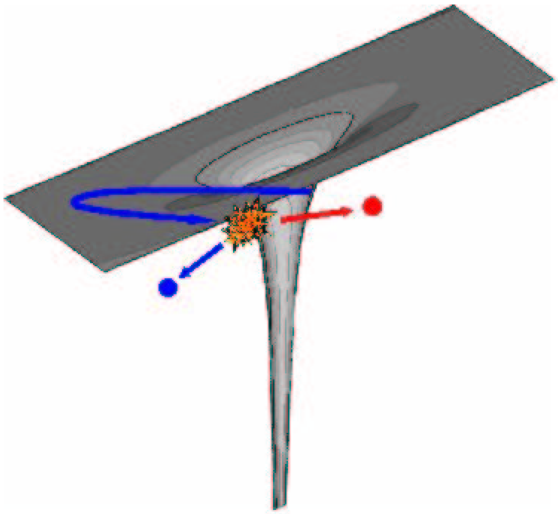


Figure 5: A visualization of rescattering: the blue electron collides back with its parent ion with a high enough kinetic energy that it liberates another electron, leaving two electrons propagating in the field. The phase of the laser field at the time of ionization is the main factor in determining the electron kinetic energy at impact.

laser beam is derived by solving the paraxial wave equation for a Gaussian intensity profile [9]. This solution gives a complete scalar description of the electric field,

$$E(r,t) = A\frac{w_0}{w(z)} \cdot e^{i\varphi(r,z)-\omega t} \cdot e^{-\frac{r^2}{w(z)^2}} \tag{3}$$

where

$$w(z) \equiv w_0\sqrt{1+\frac{z^2}{z_0^2}} \tag{4}$$

is the spot size. The radius of curvature is defined as

$$R(z) \equiv z + \frac{z_0^2}{z} \tag{5}$$

and

$$\varphi(r,z) \equiv kz - \tan^{-1}\left(\frac{z}{z_0}\right) + \frac{kr^2}{2R(z)} \tag{6}$$

is the time dependent phase, where $r^2 = x^2 + y^2$ is the square of the transverse direction, z is the direction of propagation, and $z_0 = \frac{\pi w_0^2}{\lambda}$ is the Rayleigh range. An electric

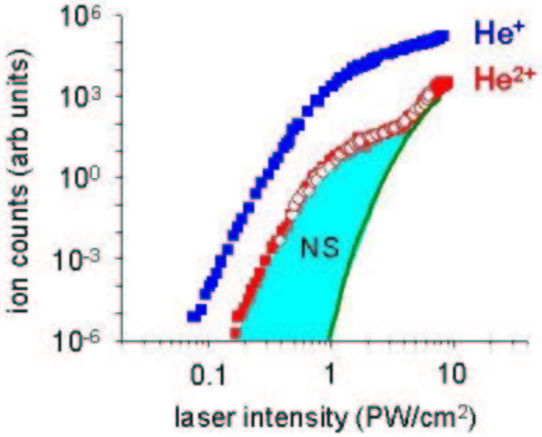


Figure 6: Theoretical yields for sequential ionization are represented by the dark green line. The red line is experimental data, which clearly does not agree with the sequential ionization model. The difference between these two, labeled 'NS', is the non-sequential ionization yield.

field can be approximated as the sum of transverse and longitudinal components such that

$$\vec{E} = \vec{E}_t + \hat{z}\vec{E}_z \tag{7}$$

The requirement from Maxwell's equations that the divergence of the electric field equal zero creates a non-zero longitudinal component to the field [10]. Typically this longitudinal term is assumed to be zero, and the transverse term takes the magnitude of the electric field given in equation A along the polarization direction. This clearly violates Maxwell's equations, since the divergence of the electric field does not equal zero due to the spatial gradient of the magnitude of the electric field (primarily in the transverse direction). If we assume that the change in the field along the z-direction is due mostly to the spatial carrier e^{ikz} (which is already assumed when invoking the paraxial approximation), then setting $\vec{\nabla} \cdot \vec{E} = 0$ gives $E_z = (\frac{i}{k})\vec{\nabla} \cdot \vec{E}_t$ which gives

$$E_z = E(r,t)(-2x)\left[\frac{1}{2R(z)} + \frac{i}{kw^2(z)}\right] \tag{8}$$

for polarization along the x direction. Note that the second term in the brackets is shifted by 180 degrees, or pi, with respect to the main portion of the oscillating electric field.

This out of phase portion of the electric field (E_z), oscillating in the direction of propagation of the beam, can have a significant impact on the classical trajectory of the liberated electron, because the peak of the laser field will correspond to a minimum in this portion of the E_z term (resulting in large drifts demonstrated in the simulations) [5]. Since the rescattering picture relies on the return of the first electron to the ion core, any change to this trajectory should greatly affect the double-ionization yield.

The crucial aspect in rescattering, when considering classical trajectories, is how close the electron gets to the ion core; if the electron misses the ion, how can it help liberate another electron? A path that brings the first electron back to the ion is most likely to yield a rescattering event. In fact, the sister process of rescattering (high harmonic generation) relies on coupling of the electron back to the ground state [11]. Of course, the electron still has to come back to the ion, at which point it recombines with the parent ion to the ground state, and thereby emits a single high energy photon. The excess electron kinetic energy can go into the photon. All of these processes rely on the electron re-encountering the ion; clearly, classical trajectories producing rescattering results should come close to hitting the origin, where the atom is located. By incorporating the longitudinal electric field into the Java simulations, the laser pulse is no longer just a plane wave, giving a more realistic, yet still controllable and intuitive, description of the laser field.

3 Programming

3.1 Introduction

All Java applets were developed and test run with Borland's JBuilder X Foundation, a free web download. Applets were compiled and viewed using the javac and appletviewer programs, respectively, from Sun's Java 2 SDK, SE $v.1.4.2_08$ (also available free online). As with learning any new language, there were problems and obstacles to overcome in order to get the desired results. I was completely unfamiliar with the Java programming language before this project, so, while my learning curve was fairly steep at the beginning, it still took awhile to get the hang of all the little nuances of code required to make the applets even appear on the screen. My first Java experience was running small applications that took input from the keyboard and then outputted some calculation, such as distance between two points, to the screen. I learned that there was a difference between Java applets and applications; applets are Java programs that are intended to be embedded into an HTML document, transported across a network, and executed using a browser [12]. Java applications are stand-alone programs that can be executed using the Java interpreter. As for my applet development, I decided to begin with a few simple applets of nothing more than drawings of non-physics-related objects. Figure 7 is an example from Java Software Solutions, entitled 'Snowman.java', that I used to familiarize myself with the Java applet paint method, in which various colors and shapes are defined and created [12].

In order to begin modeling real physical systems, I developed several applets that obeyed and demonstrated various laws of classical physics. Gravity (both two-body and three-body systems), elastic collisions, and the Lorentz force law have been explored. Java applet 'Movement,' shown in Figure 8, has user controllable velocities in both the horizontal and vertical directions. Collisions with the barriers of the ap-

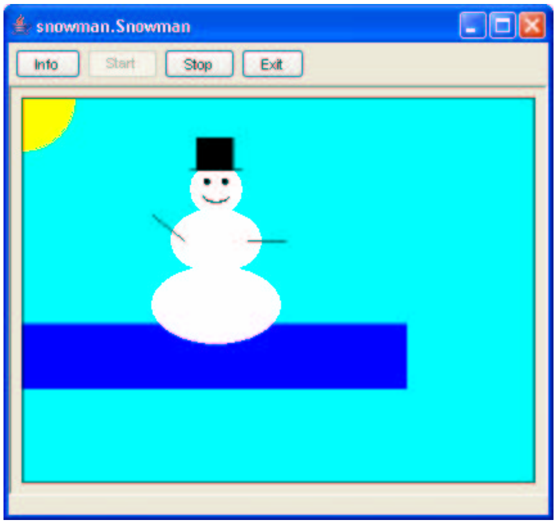


Figure 7: Java applet 'Snowman' required using the Java coordinate system in combination with the paint method to draw a snowman using ovals, rectangles, arcs, lines, and various colors.

plet were made to be perfectly elastic, thus the electron retains all of its pre-collision kinetic energy. In 'Tequalmass,' Newton's universal gravitational law is used to compute the forces on each of the masses. User controlled mass not only increases the calculated mass (and thus force) of the object, but the visual size increases as well, providing a more realistic visualization of the system as seen in Figure 9.

In order to get accurate movement in the applets, it was necessary to calculate the force(s) at work at small time intervals compared to the characteristic time scale of the moving objects. This was done through trial and error; eventually a time interval was found that was a good combination of smooth and accurate motion. Too large of a time step proved to be too inaccurate, while one that was too small resulted in slow movement and long run times, without improving the accuracy of the simulation.

The fundamental constants of nature were left out in these simulations and replaced with arbitrarily selected values in order to get the desired visual results necessary for

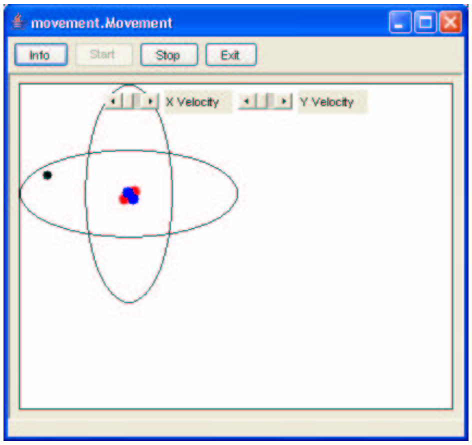


Figure 8: Java applet 'Movement' allows the user to control both the horizontal and vertical velocities via the use of scrollbars at the top of the screen. The electron reacts completely elastically with the walls of the applet. The picture of the atom was drawn by invoking the paint method.

successful applets. For example, the strengths of gravity and the relationship between the strength of the electric and magnetic fields are not in their exact relationship as found in nature, but this does not affect the physics being modeled, only the scale of the visualizations. The laws of physics still hold in terms of their dependence on position or velocity, but the spatial and temporal scales are arbitrary.

Anytime there is a chance for a singularity in the calculations taking place, for example in $\frac{1}{r}$ or $\frac{1}{r^2}$ forces or potentials, it is necessary to avoid that value going to infinity. This is done by adding a small constant, delta, to the denominator, $\frac{1}{(r+\Delta)^2}$, making it impossible to divide by zero at any time. This softening of the potential, as it is called, has a negligible affect on the overall value of the variable being calculated, but is a necessary safety parameter. Not including this term can result in a sudden, non-physical acceleration of the particle.

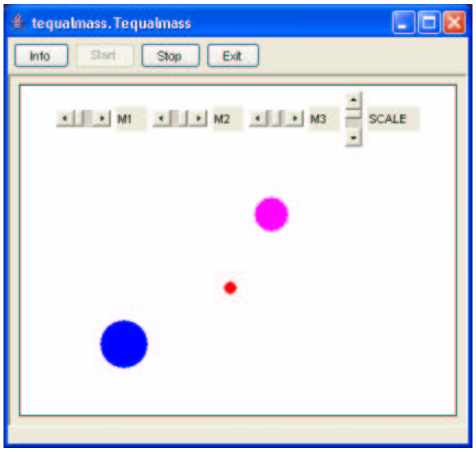


Figure 9: Java applet 'Tequalmass' has three (originally) equal masses react according to the gravitational force between them. The user can control the relative masses of all three objects, and their size (and force) will vary accordingly. The scale scrollbar allows users to zoom in or zoom out if the masses go off the originally visible screen.

3.2 Explanation of 'Simulation.java'

'Simulation.java' is the Java applet (see Figure 10) which superimposes electric, magnetic, Coulomb, and longitudinal electric field forces on an electron in real time, and allows the user to control many parameters, such as the intensity of the electric field, and the strengths of the magnetic field and the longitudinal electric field (both phases). An infinite number of different electron trajectories are possible because of the many parameters adjustable by the user not just before the release of the electron, but at any time.

3.2.1 The *Go* button

Clicking on *Go* releases the electron to be controlled by all the forces that are present, and its position and velocity adjust in real time to any changes that are made by the user via the numerous scrollbars. It allows the user to visualize the oscillating laser field (vertical bar on the right of the applet screen), and then choose when to release the electron. In reality, the phase of release of the electron is determined by the evolving tunneling rates. Here, the user can try various release times to probe the effect of phase on the classical trajectory.

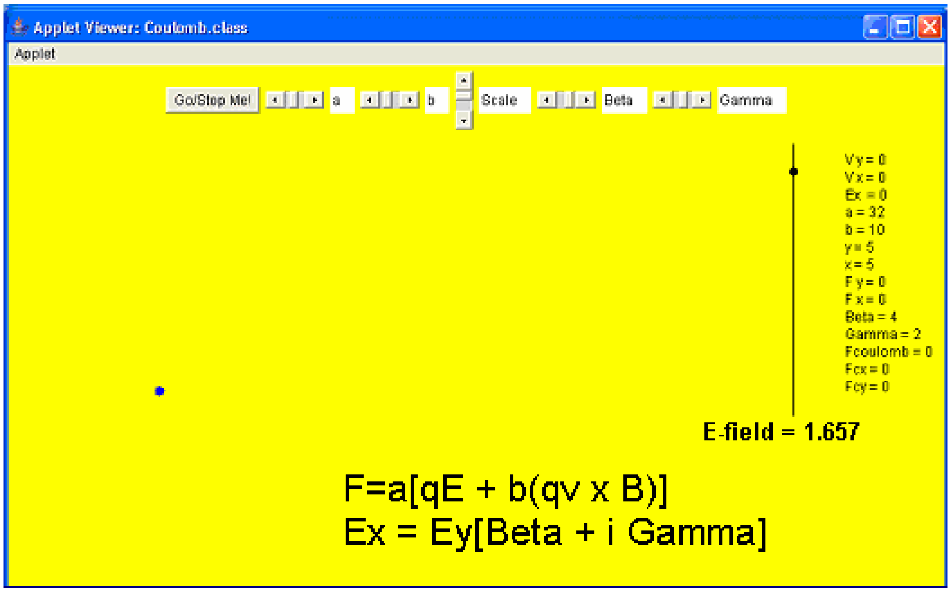
3.2.2 Having two timers

In order to allow the user to select and visualize the phase of the electric field at the time of the start of the simulation, it was necessary to have the electric field oscillating at the start of the applet (and therefore needing a running timer), while at the same time preventing the values of forces and position from being calculated before the *Go* button was clicked. By introducing two timers such that only one is running at the beginning and the other remains at zero, it was possible to get the desired result. The first timer, $timer1$, controls the movement of the oscillating electric field, while $timer2$ (not running at the beginning) controls the movement of the electron and the calculation of all forces. When the *Go* button is clicked, $timer2$ is started with the same value held by $timer1$ at that time, thereby forcing them equal to each other, resulting in a smooth transition from a stationary phase to one of movement.

3.2.3 Scrollbars and Parameters

Figure 10 is a screen shot of Simulation.html, just before the *Go* button was clicked, viewed with appletviewer.exe. The scrollbars at the top control all parameters, while the electric field oscillates up and down as simulated by the black circle on the vertical line. The equations on the bottom of the applet provide the user with a reinforcement of what exactly the parameters are actually controlling.

Scrollbars were the method of choice for parameter control due to their simplistic



```
Vy = 0
Vx = 0
Ex = 0
a = 32
b = 10
y = 5
x = 5
F y = 0
F x = 0
Beta = 4
Gamma = 2
Fcoulomb = 0
Fcx = 0
Fcy = 0
```

E-field = 1.657

F=a[qE + b(qv x B)]
Ex = Ey[Beta + i Gamma]

Figure 10: Screenshot of Coulomb.html just before the *Go* button is clicked. One can see the user adjustable scrollbars at the top of the screen, as well as a printout of all variables and forces at the right. The phase of the electric field is mimicked by the black circle as it goes up and down along its vertical line.

use and visualization of the maximum and minimum values. The parameter a begins at one, but can be varied between zero and ninety, and represents overall intensity of the laser field. The value of b is initially zero and can reach as high as forty, representing the independent strength of the magnetic field. The *Scale* scrollbar is a type of zoom feature. By increasing the value, you are effectively zooming out, allowing the applet to view the electron if it happens to go out of view. By plotting the electron's x and y positions as a function of the variable $SCALE$ namely, $\frac{x_1}{SCALE}$ and $\frac{y_1}{SCALE}$, it is possible to zoom out without affecting any of the values of any parameters or variables. Both *beta* and *gamma* are initially set to zero, but can have integer values between negative and positive ten. This is the control of the so-called longitudinal electric field, E_z.

3.2.4 E_z and B

In order to make the drift (or horizontal motion) of the electron visible, it was necessary to increase the strength of the magnetic field in relation to the electric field. The values chosen for B and E_z do not reflect the true values of the real world, but, as mentioned previously, still model the physics correctly. Since the electric field points vertically on the screen and the propagation (longitudinal) is left-right, the magnetic field then points into and out of the screen.

3.2.5 How force is broken down and turned into position with small time steps

Using small time increments, position and velocity are updated very often in relation to their trajectory, resulting in accurate plots. The following code determined the forces on the electron as well as calculated the trajectory to be plotted. The time step was incremented each time this loop ran.

```
y1 = y1 + (vy1 * dt); //The y-coordinate for the given loop
x1 = x1 + (vx1 * dt); //The x-coordinate for the given loop

B = (E/500); //The magnetic field as a function of electric field
                                        //calculated earlier in the applet

Ex = (Beta/2 * (Math.sin(0.25*time)))  //The in phase portion of the longitudinal electric field
+ (Gamma/2 * (Math.cos(0.25*time))); //The out of phase part of the longitudinal electric field

dist = Math.sqrt(x1 * x1 + y1 * y1; //The distance between the nucleus and the electron

fcoulomb = A / ( (dist + delta) * (dist + delta))  //Calculation of the force due to the
                                                        //Coulomb potential

fcoulombx = -1 * fcoulomb * x1 / dist; // cos(theta) //x-component of the Coulomb force
```

```
fcoulomby = -1 * fcoulomb * y1 / dist; // sin (theta) //y-component of the Coulomb force


forcey = a * ((q * E) - ((b/1) * vx1 * B))
 + fcoulomby;                                //The force on the electron in the vertical
                                              //(polarization) direction due to the Lorentz
                                      //force and the Coulomb potential


dvy1 = (forcey) * dt / M1; //The change in vertical velocity
                                         //experienced by the electron one time step at a time


vy1 = vy1 + dvy1; //The new vertical velocity for this loop


forcex = a * (((b / 1) * (q * (vy1 * B)))) //The force on the electron in the horizontal
+ (q*Ex) + fcoulombx;                            //(propagation) direction due to the Lorentz
                                                 //force, the Coulomb potential, and the
                                                 //longitudinal electric field


dvx1 = (forcex) * dt / M1; //The change in horizontal velocity
                                                 //experienced by the electron one time step
                                                 //at a time


vx1 = vx1 + dvx1; //The new horizontal velocity for this loop
```

3.2.6 Graphics

I used the $fillOval$ command in the graphics to represent the electron. The x and y coordinates must be rounded to the nearest integer for the graphics to work, hence the use of (*int*) which casts an integer value to a floating point or double, allowing the graphics to handle the command. The size of the electron also gets smaller as you zoom out, making the *Scale* feature seem realistic. The constants added to the position are just for the visualization; they start the electron in a position that allows for maximum trajectory viewing under almost all choices of parameters.

```
page.setColor(Color.blue);


page.fillOval( (int) ( (x1 / SCALE) + 120), (int) ( (y1 / SCALE) + 270),
(int) ( (8/SCALE + 1)), (int) ( (8/SCALE + 1)));
```

3.2.7 Writing to a data to a file

The following code would output the desired data (in this case the coordinates of the electron) to a *.csv* file, where it then could be easily incorporated into any spreadsheet or graphing program.

```
FileOutputStream outstream; //defining new variables
PrintStream printstream;

-----------------

public void init() {


try
  {
    outstream = new FileOutputStream("outputfile.csv"); //creates the output file
    printstream = new PrintStream ( outstream ); //allows the data to be sent
  }
  catch (Exception e)
  {
    System.err.println ("Error writing to file");        //just in case there is a problem
                                                         //with writing to the file, this allows
                                                         //the program to continue, just
                                                         //without any output file
  }

-----------------


printstream.println (fmt.format(x1) + "," + fmt.format(y1)); //selecting which variables
     //to send to the output file
```

4 Results

The initial simulations were those of an electron under the force of only a sinusoidal electric field in the vertical direction. In the first test, the intensity of the electric field, corresponding to the value of a, from the force law

$$F = a \cdot qE, \tag{9}$$

was held constant throughout each trial, while the phase of the electric field at the time of release was varied. The applet keeps track of the sinusoidal electric field (which is a sine wave with maximum amplitude two), therefore phi, the release angle, is defined as

$$\varphi = \cos^{-1}\left(\frac{E_{GO}}{2}\right). \tag{10}$$

The peak of the electric field is defined as zero degrees. Figure 11 demonstrates just how important the phase of the electric field is in regards to electron drift. Since the intensity was the same for all runs, the differences in trajectory can be accounted for only by the change of phase angle, *phi*. Comparing trajectories of the electrons after only a half-cycle reveals the importance of phase angle to the rescattering model. In the case where *phi* equals zero, the electron comes back to the original ion every time, but when *phi* is seventy-nine degrees, the electron does not come close to hitting the ion even after a half-cycle. Even with a small *phi*, after a few cycles, the electron no longer returns close enough to the ion for rescattering to take place. Since the rescattering model depends on the electron returning to the ion, any *phi* greater than a few degrees, or after a cycle or more, is very unlikely to help in the rescattering process.

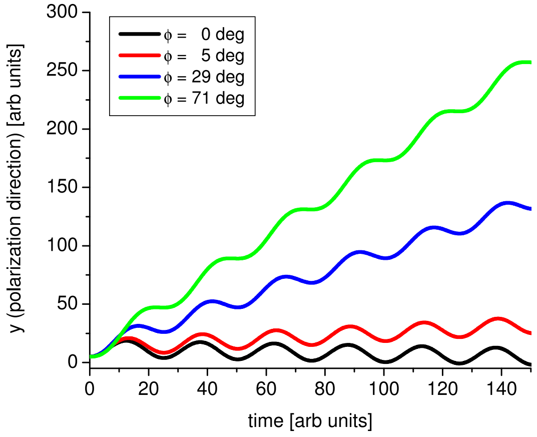


Figure 11: Since intensity was the same for each run, the magnitude of the drift of the electron is clearly dependant upon the phase of the laser field at the time of release, or ionization. The electron remains in the same phase regardless of drift, however.

The next trials varied the parameter a while holding the release constant at the peak of the electric field, meaning $\varphi = 0$. The drift of the electron is determined by *phi*. The excursion (amplitude of the wiggles) is determined by the strength of the electric field. Since *phi* is held constant, Figure 12 shows the effect of increasing electric field strength, or intensity, on electron motion. The greater the intensity, the greater the vertical amplitudes, as expected both intuitively and from actual experiment. There is more kinetic energy and velocity associated with these fields. Again, regardless of amplitude, the electron remains in phase with the other trials.

In the above two examples, there is no drift in the propagation direction. There is no force, thus no movement, in the propagation (horizontal) direction because the electric field is purely in the vertical (polarization) direction and the magnetic field term has been (artificially) turned off.

The electron was then put under the Lorentz force law,

$$F = a\Big[q[E + b(v \times B)]\Big]. \tag{11}$$

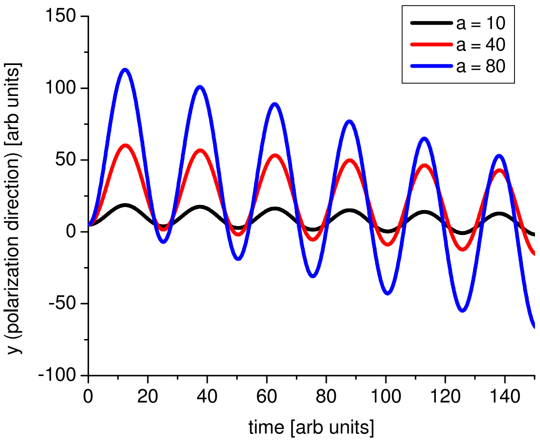


Figure 12: Increasing the intensity, or the value of a in $F = a \cdot qE$, results in larger electron displacement from the nucleus, but in phase regardless of the amplitude, meaning there is more kinetic energy and velocity associated with the higher intensity fields, agreeing with both common sense and physics.

where a and b are user adjustable parameters, and released at the peak of the field three times under varying conditions. Increasing just the magnetic field strength resulted in larger longitudinal drifts, as expected because the $[b(v \times B)]$ term is responsible for the movement in the propagation direction. The velocity is initially in the vertical direction and the magnetic field is into (or out of) the plane of the page, forcing the drift velocity horizontal as the right hand rule, or cross product tells us. A comparison of the blue and black lines in Figure 13 gives a quantitative evaluation of the effects of the magnetic field term. When the intensity was doubled, but the magnetic field strength decreased relative to the overall intensity, the drift of the electron was still very large. In reality, the only way to increase the magnetic field is to increase the intensity, resulting in large transverse *and* longitudinal motion.

Another way to study electron drift and trajectory is to take an inertial frame

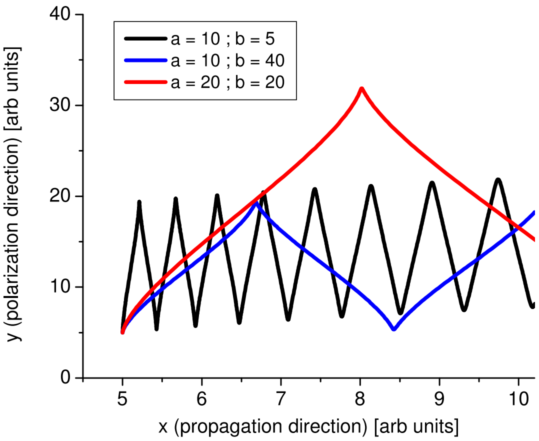


Figure 13: Increasing the strength of the magnetic field (blue line) reduces the chance of rescattering because the electron is pushed farther from the nucleus each cycle. Increasing the overall intensity (red line) results in larger amplitude and thus a more energetic electron, but even at a lower relative magnetic field, the drift is too great to support rescattering.

drifting with the electron as the frame of reference, and then plot x versus y. This is done by subtracting a straight line from all of the x values, which is the same as removing the cycle-averaged motion from the horizontal position. The oscillations are still all there, however the plot now takes on an interesting shape. Figure 14 displays the figure eight motion expected [13], however, a slight drift is seen which is caused by slight rounding and calculation errors in the Java applet. If solved mathematically, the graph would look like just a single 'figure-eight'.

In reality, we do not have independent control of the $v \times B$ term, since we cannot adjust b in the laboratory. It therefore seems impossible for the electron to return to its parent ion with any significant velocity, since it only obtains high velocities at higher intensities, but those higher intensities push the electron too far from the ion for rescattering to take place. However, the presence of the longitudinal electric field as previously described can alter the electrons trajectory drastically enough to produce

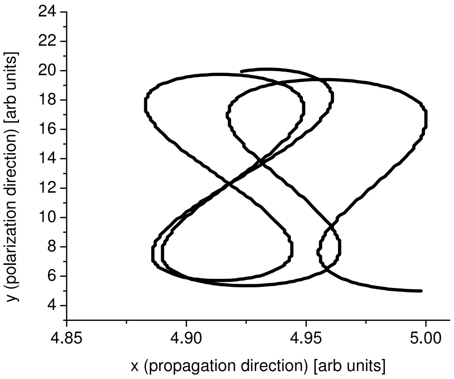


Figure 14: The figure-eight motion of the electron under the Lorentz force when viewed from the rest frame of the electron. There is some slight experimental error in the calculations; otherwise the motion would be seen as a single figure-eight.

results where rescattering is very likely. For the Java simulations, the longitudinal electric field takes the simplified form of equation 8, or

$$E_l = E_t[\beta + i\gamma], \tag{12}$$

where E_t is the transverse sinusoidal electric field, and β and γ are constants resulting in a non-zero longitudinal electric field (E_l), with both in and out-of-phase terms. My java applets had both β and γ ranging from negative ten to positive ten; however I only used positive values in my simulations to emulate more realistic conditions. Again, it is not physically possible to independently control beta and gamma in the real world (as they are governed by the nature of the focusing of the laser beam), but for the sake of simplicity and understanding their effects on electron drift and trajectory, independent variation was allowed. As Figure 15 shows, γ is much more vital to rescattering than β. The electron was released at the peak of the laser field, corresponding to a null in the out- of-phase part (γ) of the longitudinal electric field. As figure 11 showed us, the greater the phi, the greater the drift; therefore a large drift in the propagation (horizontal) direction is associated with gamma. It so happens that this term nearly completely cancels out the horizontal motion associated with the $v \times B$ term of the Lorentz force. In fact, during the third trial (red line), the electron was brought back to very near its original position not just once, but several times before eventually drifting away. This is a perfect example of an opportunity for rescattering. Having the electron return to the ion core with a sizeable amount of energy is the most likely situation for having a rescattering event, and in this case, this happens not once, but three times. This fortuitous balance of forces would in reality depend on the laser intensity, focal characteristics and the position of the atom in the laser focus. Such a combination should result in enhanced rescattering.

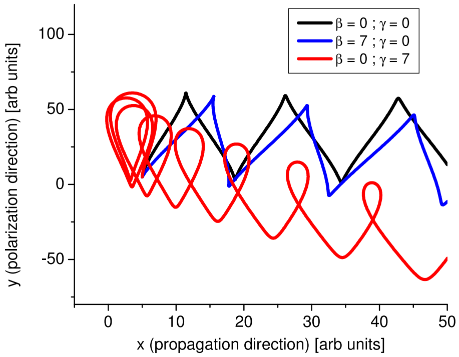


Figure 15: The longitudinal electric field has a dramatic effect on electron trajectory, especially when the out of phase gamma term is increased (red line). Increasing beta only made the drift asymmetrical, not affecting the chances of rescattering significantly (blue line). The black line is the traditional example without any longitudinal electric field as studied previously.

However, the laser-electron situation is actually even more complicated. Including the Coulomb potential from the protons in the nucleus of the ion is necessary to complete a realistic picture of the laser-atom interaction. In the Java simulations, the force associated with this potential takes the form of

$$F_c = \frac{A}{(r+\Delta)^2}, \tag{13}$$

where $A = 800$ is the arbitrary strength of the Coulomb force, $\Delta = 0.1$ is the softening term, and r is the distance between the electron and its parent nucleus. From introductory physics, the real Coulomb force is

$$F_c = \frac{kq_1q_2}{r^2}, \tag{14}$$

where k is Coulomb's constant, q_1 and q_2 are the electric charges of the bodies, and r is the distance between them. Notice that if the charges differ in sign, as they do for a proton and electron, this force will be attractive. Figure 16 shows that at high intensity, this force has little effect on electron trajectory, but at low intensity, the Coulomb potential strongly reacts with the electron, and in this case, causes the electron to wrap around the ion and then travel with a reasonable trajectory and drift expected for the intensity $a = 10$.

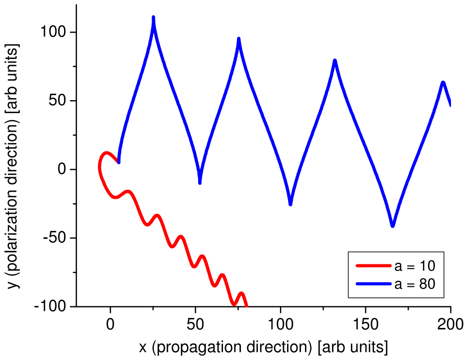


Figure 16: The addition of the Coulomb potential to the Lorentz force greatly affects electron trajectory at low intensities (red line), but practically reduces to just the Lorentz force as the intensity increases (blue line).

Including all the forces mentioned previously into the simulations results in interesting plots. At moderate intensity, it is a combination of all the effects which change electron trajectory, or said differently, there is no one dominant effect. In Figure 17, β was set to three, γ to seven, and intensity and magnetic field strength are shown in the legend.

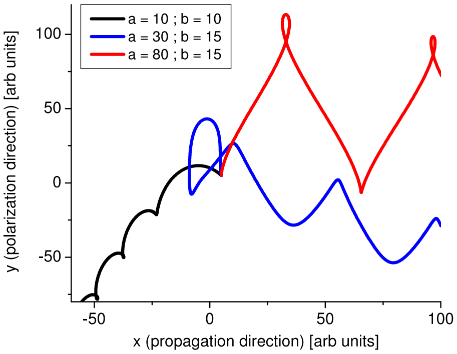


Figure 17: The superposition of the $v \times B$ term, the Coulomb potential, and E_z make for interesting electron trajectories, very dependent upon intensity of the laser field. At low intensities, the Coulomb force can take over and drastically alter the expected path of the electron. At moderate intensities, it is the combination of all forces which contribute to the trajectory of the electron. At high intensities, the electron follows a path changed little by the longitudinal electric field and the Coulomb potential.

If we look more closely at the trajectory of the high intensity electron of Figure 17 using the electron rest frame as our frame of reference, we get a similar figure to that of Figure 14. At high intensities, the affect of the Coulomb potential and E_z are minimal compared to the Lorentz force, so overall motion is similar to that previously studied. The small drift can be accounted for from small errors in the simulation due to rounding and calculating, as well as the forces from the Coulomb potential and E_z. See Figure 18.

Without E_z or the Coulomb potential, it would make sense that as intensity increased, so would the longitudinal drift due to $v \times B$, and the chance of rescattering

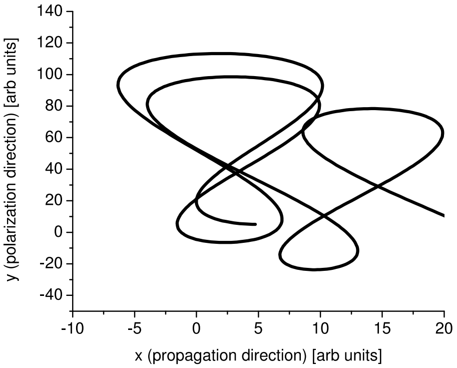


Figure 18: The figure-eight diagram for the trajectory of the electron in a high intensity field. Besides a slight drift due to simulation error in rounding, Coulomb and E_z effects are minimal, but noticeable; otherwise just one figure eight would be seen.

would decrease. Figure 19 is a plot of distance from the nucleus as a function of intensity, or a. The lowest line corresponds to $a = 10$, and for each line above that the a parameter is incremented by five. Again, it appears that at only the lowest intensities does the electron ever come close to impacting the parent ion. But at the low intensity, and thus low kinetic energy, rescattering is unlikely to contribute to ionization.

If now E_z is taken into account, an interesting thing happens. For low intensities, E_z actually dominates the Lorentz force, resulting in a decreased chance of rescattering. It is not until the middle intensities that there is a balance between E_z and Lorentz, thus resulting in trajectories and velocities which seem right for rescattering. Figure 20 plots distance from the ion as a function of intensity for the low intensities ($a = 10$, the red line to $a = 40$, the thick black line). At $a = 40$, rescattering is most likely to occur; any lower intensity results in greater distances from the parent ion.

If we now plot the higher intensities, it is easy to see that any intensity above $a = 40$ results in farther distances from the ion, just like the results seen without the

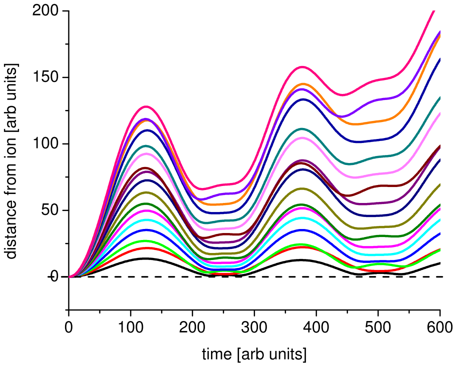


Figure 19: Distance from the ion as a function of intensity ignoring the longitudinal electric field and the Coulomb potential. Each line has an a value five higher than that of the line below it, ranging from $a = 10$ (black line at the bottom) to $a = 90$ (pink line at top). Notice that at only the lowest of intensities does the electron ever come back close to the parent ion (distance equals zero denoted by dashed line) once released.

E_z term. At $a = 40$ therefore, rescattering is most likely to occur; any intensity lower or higher results in greater distances from the parent ion as Figure 21 shows.

If we now superimpose the $a = 40$ trajectory from Figure 19 and Figure 20, we are effectively comparing the effect of E_z, as shown in Figure 22. In this case, the presence of the longitudinal electric field dramatically increases the chance of rescattering, as the electron is brought back over five times closer to the parent ion on the first cycle, and then more than seven times closer on the next.

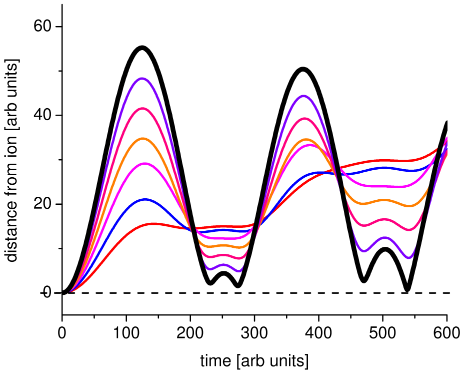


Figure 20: Distance from the ion as a function of intensity ranges $a = 10$ to $a = 40$, including E_z (with $\beta = 3$, $\gamma = 7$). The dark black line corresponds to $a = 40$ and each line below it at $time = 125$ corresponds to a decreasing value of a by five, down to $a = 10$. It is clearly seen that at $a = 40$ rescattering is most likely to happen, since it is the only intensity in which the distance from the ion returns to nearly zero several times.

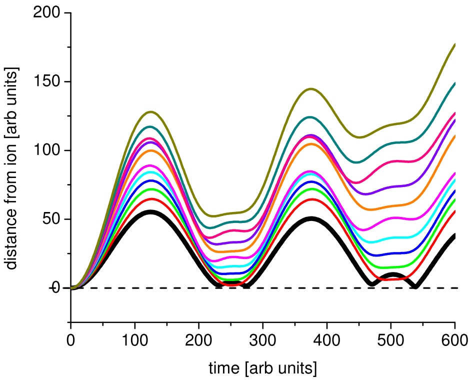


Figure 21: Distance from the ion versus intensity as a function of time. E_z is included, but its effects are minimal at these higher intensities. The dark black line is $a = 40$, and each line successive line has an a value increased by five, up to $a = 90$. It can be seen that at $a = 40$, rescattering is most likely to occur, and any higher intensity results in more electron drift.

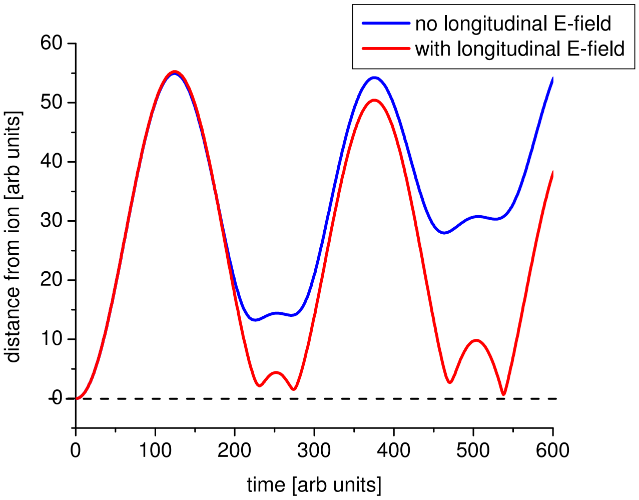


Figure 22: Plot of electron distance from ion versus time ignoring the Coulomb potential. As demonstrated by the graph, the longitudinal electric field (with $\beta = 3$, $\gamma = 7$) greatly increases the chances of rescattering, as the electron comes more than five times closer to the parent ion on the first time cycle and then more than seven times closer the next time.

5 Error Analysis

With the exception of the data for Figure 11, all electrons were released with at the peak of the optical cycle ($E_{GO} = 2$), in an attempt to study and compare the effects of the user-controllable parameters. Even though the electrons were released at the peak of the electric field, slight drift occurred, probably due to the fact that the program started the calculations at the next time step, which would result in a very small, but still measurable phase angle, and thus the drift associated with the electric field. As mentioned previously, the constants used in the calculations were arbitrary, as well as all units of measurement and time. When dealing with high-energy electrons, quantum mechanics should not be ignored, but for simplicity, all quantum effects were left out. Relativistic quantum effects are beyond the scope of this project. Electrons, of course, are not billiard balls, they are wave packets; there is no hitting or striking involved, it is really all about probabilities. This does not make the simulation and data meaningless; to include some quantum effects it would just be necessary to simulate wave packets by launching many similar classical trajectories at once. In fact, a very recent article that ignores quantum effects completely has accounted for all the qualitative features found in strong-field double ionization [14]. This work approximates that laser field as a plane wave.

6 Conclusion

The question of the dynamics of electron correlation remains a fundamental puzzle in quantum mechanics. Its importance goes far beyond the intellectual challenge of the few-body problem; it extends to its wide ranging impact in numerous fields of science and technology. It is the correlated motion of electrons that is responsible for the structure and the evolution of large parts of our macroscopic world. Electron correlation is also responsible for driving chemical reactions, superconductivity, and many other condensed-matter effects [15]. A non-sequential model of ionization must be used if theoretical data is to agree with experiment. The use of the longitudinal electric field in modeling processes of rescattering makes it clear that rescattering should occur for a distinct variety of initial conditions, those mainly depending on intensity and phase of the electric field as well as the gamma term of E_z. Visualization of this, as well as other atomic processes, might prove to be powerful tools in strong field laser physics. My project has laid the groundwork for the continual development of Java applets with realistic (both optical characteristics of the laser focus and quantum effects such as wave-packet spreading) features.

7 Appendix: Java Source Code

Source Code of Simulation.java

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
import javax.swing.*;
import java.text.DecimalFormat;
import java.io.*;

public class Simulation extends Applet implements ActionListener {
private boolean isStandalone = false;

public void actionPerformed(ActionEvent event) {
    //System.out.println("Clicked on the go button");
    //System.out.println("a = " + a + " b = " + b);
        if (Go == false)
        {
                timer2.start();
```

```
              E = Etemp;
              Go = true;
      }
      else
      {
              timer2.stop();
              Go = false;
      }
}
//Get a parameter value
public String getParameter(String key, String def) {
  return isStandalone ? System.getProperty(key, def) :
      (getParameter(key) != null ? getParameter(key) : def);
}
//Construct the applet
public Simulation() {
}

//Initialize the applet

//Component initialization
private void jbInit() throws Exception {
}

//Get Applet information
public String getAppletInfo() {
  return "Applet Information";
}

//Get parameter info
public String[][] getParameterInfo() {
  return null;
}

private Timer timer, timer2;

private Button goButton;
private boolean Go = false;

private Button onButton;

final int APPLET_HEIGHT = 600;
final int APPLET_WIDTH = 800;

private double x1=5;
private double y1=5;

private Scrollbar ascrollbar;
private Label alabel;

private Scrollbar bscrollbar;
private Label blabel;

private Scrollbar SCALEscrollbar;
private Label SCALElabel;

private Scrollbar Betascrollbar;
private Label Betalabel;

private Scrollbar Gammascrollbar;
private Label Gammalabel;

private double a = 1, b = 0, E, Etemp, B, D = 2;

private double time = 0.0;

private int SCALE = 1;

private double forcey, forcex;
private final int q = 2, C = 900, M1 = 90;

private double vx1 = 0, vy1 = 0.0;
private double dvx1=0.0, dvy1 = 0.0, dt = 0.1;

private double Ex, Beta, Gamma;
private int A =800;
private double delta = 0.1; // softening of potential
private double fcoulomb, fcoulombx, fcoulomby, dist;

private Image image;

FileOutputStream outstream;
PrintStream printstream;

Graphics backbuffergc;
Image backbuffer;

public void init() {

image = getImage (getCodeBase(), "background.jpg");

try
{
  //outstream = new FileOutputStream("outputfile445.csv");
  //printstream = new PrintStream ( outstream );

   //printstream.println (x1 + "," + y1);

   //printstream.close();
}
catch (Exception e)
{
  System.err.println ("Error writing to files!!");
}

  addMouseListener(new ReboundMouseListener());

  goButton = new Button("Go");
  goButton.addActionListener(this);
  add(goButton);

  ascrollbar = new Scrollbar(Scrollbar.HORIZONTAL, 1, 10, 0, 100);
  add(ascrollbar);
  ascrollbar.addAdjustmentListener(new ScrollbarAdjustmentListener());

  alabel = new Label("a");
  add(alabel);

  bscrollbar = new Scrollbar(Scrollbar.HORIZONTAL, 0, 10, 0, 50);
  add(bscrollbar);
  bscrollbar.addAdjustmentListener(new ScrollbarAdjustmentListener());

  blabel = new Label("b");
  add(blabel);

  SCALEscrollbar = new Scrollbar(Scrollbar.VERTICAL, 1, 3, 1, 9);
  add(SCALEscrollbar);
  SCALEscrollbar.addAdjustmentListener(new ScrollbarAdjustmentListener());

  SCALElabel = new Label("Scale");
  add(SCALElabel);

  Betascrollbar = new Scrollbar(Scrollbar.HORIZONTAL, 0, 10, -10, 20);
  add(Betascrollbar);
  Betascrollbar.addAdjustmentListener(new ScrollbarAdjustmentListener());

  Betalabel = new Label("Beta");
  add(Betalabel);

  Gammascrollbar = new Scrollbar(Scrollbar.HORIZONTAL, 0, 10, -10, 20);
  add(Gammascrollbar);
  Gammascrollbar.addAdjustmentListener(new ScrollbarAdjustmentListener());

  Gammalabel = new Label("Gamma");
  add(Gammalabel);

  timer = new Timer(10, new ReboundActionListener());
  timer2 = new Timer(10, new ReboundActionListener());
  timer.start();
}
```

```
private class ScrollbarAdjustmentListener
    implements AdjustmentListener {
  public void adjustmentValueChanged(AdjustmentEvent e) {
    a = ascrollbar.getValue();
    b = bscrollbar.getValue();
    SCALE = SCALEscrollbar.getValue();
    Beta = Betascrollbar.getValue();
    Gamma = Gammascrollbar.getValue();
  }
}

// calculating E-field
public double calcE() {
  Etemp = D*Math.sin(0.25 * time);
  time = time + dt;
  if (time > 200000 * (Math.PI)) {
    time = 0.0;
  }

  return Etemp;
}
public void fillvalues() {

  y1 = y1 + (vy1 * dt);
  x1 = x1 + (vx1 * dt);

  B = (E/500);

  Ez = (Beta/2 * (Math.sin(0.25*time))) +  (Gamma/2 * (Math.cos(0.25*time)));

    dist = Math.sqrt(x1 * x1 + y1 * y1);
    fcoulomb = A / ( (dist + delta) * (dist + delta));
    fcoulombx = -1 * fcoulomb * x1 / dist; // cos(theta)
    fcoulomby = -1 * fcoulomb * y1 / dist; // sin (theta)

  forcey = a * ((q * E) - ((b/1) * vx1 * B)) + fcoulomby;
  dvy1 = (forcey) * dt / M1;
  vy1 = vy1 + dvy1;

  forcex = a * (((b / 1) * (q * (vy1 * B)))) + (q*Ez) + fcoulombx;
  dvx1 = (forcex) * dt / M1;
  vx1 = vx1 + dvx1;

  if (x1 < (800 * SCALE)) {
    x1 = x1 + (vx1 * dt);
  }
  else {
    Go = false;
    on = false;
    x1 = 0;
    y1 = 0;
    a = 1;
    b = 0;
    dvy1 = 0;
    dvx1 = 0;
    vy1 = 0;
    vx1 = 0;
    forcex = 0;
    forcey = 0;
    Beta = 0;
    Gamma = 0;

    timer2.stop();
  }

  DecimalFormat fmt = new DecimalFormat ("0.###");

  System.out.println(fmt.format(x1) + "," + fmt.format(y1) + "");

  //printstream.println (fmt.format(x1) + "," + fmt.format(y1));

  //System.out.println("");
}
```

```
public void paint(Graphics page) {

  E = calcE();

  if (Go == true) {
    fillvalues(); }

  setBackground(Color.yellow);
  page.drawString("F = a [q(E + b(v x B))]", (APPLET_WIDTH/2 - 40), APPLET_HEIGHT-70);
  page.drawString("Ex = Ey [Beta + (i * Gamma)]", (APPLET_WIDTH/2 - 45), APPLET_HEIGHT-55);
  final int Xspace = (APPLET_WIDTH-85);
  page.setColor(Color.black);
  DecimalFormat fmt = new DecimalFormat ("0.###");
  page.drawString("V y = " + fmt.format(vy1), Xspace, 85);
  page.drawString("V x = " + fmt.format(vx1), Xspace, 100);
  page.drawString("Ex  = " + fmt.format(Ex), Xspace, 115);
  page.drawString("a = " + fmt.format(a), Xspace, 130);
  page.drawString("b = " + fmt.format(b), Xspace, 145);
  page.drawString("y = " + fmt.format(y1), Xspace, 160);
  page.drawString("x = " + fmt.format(x1), Xspace, 175);
  page.drawString("F y = " + fmt.format(forcey), Xspace, 190);
  page.drawString("F x = " + fmt.format(forcex), Xspace, 205);
  page.drawString("Beta = " + fmt.format(Beta), Xspace, 220);
  page.drawString("Gamma = " + fmt.format(Gamma), Xspace, 235);
  page.drawString("Fcoulomb = " + fmt.format(fcoulomb), Xspace, 250);
  page.drawString("Fcx = " + fmt.format(fcoulombx), Xspace, 265);
  page.drawString("Fcy = " + fmt.format(fcoulomby), Xspace, 280);

  // draw electron
  page.setColor(Color.blue);
  page.fillOval( (int) ( (x1 / SCALE) + 120), (int) ( (y1 / SCALE) + 270),
                (int) ( (8/SCALE + 1)), (int) ( (8/SCALE + 1)));

 page.setColor(Color.black);

 // E field magnitude

 page.fillOval((APPLET_WIDTH-133), ( -3 * (int) ((E/D) * 40) + 180), 8, 8);
```

```
   page.drawLine((APPLET_WIDTH-130), 67, (APPLET_WIDTH-130), (APPLET_HEIGHT-300));

   page.drawString("E field = " + fmt.format(E), (APPLET_WIDTH-160), (APPLET_HEIGHT-280));

  }
  private class ReboundMouseListener
      implements MouseListener {
    public void mouseClicked(MouseEvent event) {
      if (timer.isRunning())
        timer.stop();
      else
        timer.start();
    }
    public void mouseEntered(MouseEvent event) {}

    public void mouseExited(MouseEvent event) {}

    public void mousePressed(MouseEvent event) {}

    public void mouseReleased(MouseEvent event) {}
  }
  private class ReboundActionListener
      implements ActionListener {
    public void actionPerformed(ActionEvent event) {
      repaint();
    }
  }
}
```

References

[1] W. Bruce, *Historical Studies in the Physical Sciences*, Vol. 9, 299-322 (1978).

[2] A. Einstein, *Collected Papers*, 730-743 (2004).

[3] R. L. Smith, Phys. Rev. **128**, 2225 (1962); G. S. Voronov and N. B. Delone, JETP Lett. **1**, 66-68 (1965); P. Agostini, F. Fabre, G. Mainfray, G. Petite, and N. K. Rahman, Phys. Rev. Lett. **42**, 1127-1130 (1979)

[4] J. L. Chaloupka, private communication.

[5] M. Yu Kuchiev, JETP Lett. **45**, 404 (1987); P. B. Corkum, Phys. Rev. **71**, 1994 (1993)

[6] Th. Weber, H. Giessen, M. Weckenbrock, G. Urbasch, A. Staudte, L. Spielberger, O. Jagutzki, V. Mergel, M. Vollmer, and R. Drner, Nature 405, 658-661 (2000); R. Moshammer, B. Feuerstein, W. Schmitt, A. Dorn, C. D. Schrter, J. Ullrich, H. Rottke, C. Trump, M. Wittman, G. Korn, K. Hoffman, and W. Sandner, Phys. Rev. Lett. **84**, 447-450 (2000); Th. Weber, M. Weckenbrock, A. Staudte, L. Spielberger, O.Jagutzki, V. Mergel, F. Afaneh, G. Urbasch, M. Vollmer, H. Giessen, and R. Drner, Phys. Rev. Lett. **84**, 443-446 (2000); B. Witzel, N. A. Papadogiannis, and D. Charalambidis, Phys. Rev. Lett. **85**, 2268-2271 (2000); R. Lafon, J. L. Chaloupka, B. Sheehy, P. M. Paul, P. Agostini, K. C. Kulander and L. F. DiMauro, Phys. Rev. Lett. **86**, 2762-2765 (2001).

[7] J. B. Watson, A. Sanpera, D. G. Lappas, P. L. Knight, and K. Burnett, Phys. Rev. Lett. **78**, 1884-1887 (1997); J. S. Parker, E. S. Smyth, and K. T. Taylor, J. Phys. **B31**, L571-L578 (1998); M. A. Kornberg and P. Lambropoulos, J. Phys. **B32**, L603-L613 (1999); A. Becker and F. H. M. Faisal, Phys. Rev. Lett. **84**, 3546-3549 (2000); R. Panfili, C. Szymanowski, W.-C. Liu, and J. H. Eberly, L.F. DiMauro, R.R. Freeman, K.C. Kulander, eds., (American Institute of Physics, New York, NY 2000); M. Lein, E. K. U. Gross, and V. Engel, Phys. Rev. Lett. **85**, 4707-4710 (2000); G. L. Yudin and M. Ivanov, Phys. Rev. **A**, in press (2001).

[8] B. Walker, B. Sheehy, L. F. DiMauro, P. Agostini, K. J. Schafer, and K. C. Kulander, Phys. Rev. Lett. **73**, 1227-1230 (1994)

[9] P. W. Milonni and J. H. Eberly, *Lasers*, New York: Wiley, 1998, pp. 484-490.

[10] M. Lax, W. H. Louisell, and W. B. Knight, Phys. Rev. **A11**, 1365-1370 (1975).

[11] A. l'Huillier and Ph. Balcou, Phys. Rev. Lett. **70**, 774-777 (1993).

[12] Lewis, John and William Loftus, *Java Software Solutions: Foundations of Program Design*, Boston: Addison Wesley, 2003.

[13] E. S. Sarachik and G. T. Schappert, Phys. Rev. Lett. **D1**, 2738-2753 (1970).

[14] P. J. Ho, R. Panfili, S. L. Haan, and J. H. Eberly, Phys. Rev. Lett. **D1**, 093002 (2005).

[15] Bederson, Benjamin, and Herbert Walther, *Advances in Atomic, Molecular, and Optical Physics*, Amsterdam: Academic Press, 2002.