Simulation of a random walk around a gold nanorod / Detection of vapor nanobubbles around a continuously heated gold nanosphere

Simulation of a random walk around a gold nanorod / Detection of vapor nanobubbles around a continuously heated gold nanosphere

A.E. Kamp

Huygens-Kamerlingh Onnes Laboratory, Leiden University
P.O. Box 9500, 2300 RA Leiden, The Netherlands

April 28, 2016

Abstract

This thesis consists of two parts, one part describes the heating of a gold nanosphere and the forming of a vapor bubble as a result of this. The first part consist mainly of theory and there are no measurement results to support the theory, the second part describes a Matlab simulation of a random walk of a ATTO 647N molecule close to a gold nanorod that is being placed in an electromagnetic field. The simulation uses the MNPBEM toolbox to calculate the electric field generated by the gold nanorod and a random walk for the movement of the molecule.

Acknowledgement

Contents

Chapter 1

Introduction

1.1 Oldest applications of nanoparticles in history

One of the oldest known applications of metallic nanoparticles by humans is the Lycurgus Cup, it is first described in text in 1845, but the first research was done in 1959. [6] The Lycurgus Cup is a special cup that has a different color when looked through than when looked at, so a different color when it refracted light than when it transmitted light (See figures 1.1 and 1.2). Because of these special optical properties people did not know if it was really glass or a special kind of other material. [6] After a research done using X-ray diffraction it turned out that the Lycurgus cup was made out of glass. [5] There are only a few known examples of ancient glass that show this special optical property and none of the known glasses show such extraordinary color changes as the Lycurgus Cup. [7] After a chemical research it turned out that besides the normal components for ancient roman glass, it also had traces of gold and silver. [5] If it were only the small amount of gold and silver that gave the glass the special properties, there would probably be more glasses known with the same properties, but there were only a few of those known. [2] To find out what was so different about the Lycurgus Cup, the glass of the cup was investigated using a TEM and there appeared to be very small crystals, approximately between 50 nm to 100 nm, that consisted of an alloy of gold and silver. See figure 1.3 [1]

Figure 1.1: *The Lycurgus Cup with reflected light. Source: The Trustees of the British Museum.*

Figure 1.2: *The Lycurgus Cup with transmitted light. Source: The Trustees of the British Museum.*
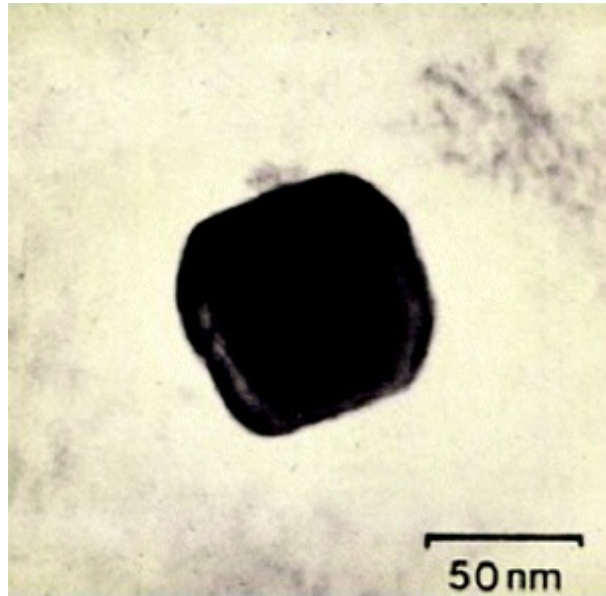
Figure 1.3: *Transmission electron microscopy (TEM) image of one of the particles in the Lycurgus cup. [1] Source: The Trustees of the British Museum.*

1.2   Strange Colors

The Lycurgus Cup is made of glass with metallic nanoparticles, but it is also possible to suspend nanoparticles in a fluid. In the laboratory we have differently sized nanoparticles suspended in water, these particles have similar characteristics as the Lycurgus Cup. When the solutions with nanoparticles are looked at directly they have a different color than when looked through at a light source. It is surprising that a sample made out of water and gold can look red or green while gold and water both have a different color. Why would the properties of a material change when it is in such a small particle? An answer to this question came from Gustav Mie, who used the Maxwell equations to create a theoretical solution for the behavior of the small particles. [19] It appeared that the color of a suspension with nanoparticles was very dependent on the size and shape of the nanoparticle as well as on the properties of the materials it is made of. [11]

1.3   Further studies on nanoparticles

When light goes through the solution containing the metallic nanoparticles, the particles will interact with to the electromagnetic field. When the

3

particle is placed in a electromagnetic field the conduction electrons shift with respect to the immobile ions, generating a restoring force which, combined with the mass of the particle, leads to harmonic oscillations. [19] The beautiful thing about nanoparticles is that they are not molecules, but they are not bulk material either. We could say that they are the bridge between two worlds. [19] The gold nanoparticles can be used to label proteins or cells, the big advantage of using gold nano-particles is that they are stable and chemically inert and therefore they can be monitored over a longer period of time. [18] And because of the behavior in specific wavelengths they can be used to destroy specific cells by heating the nanoparticle attached to, or brought into a cancer cell. [14]

Besides heating, there is also another very interesting application for the nanoparticles. When a metallic nanoparticle is placed in an electromagnetic field, it can behave like a antenna. It will then absorb the electromagnetic energy at a certain wavelength and focus a lot of this energy in certain small areas. This way it is possible to generate a very strong local electric field in an area much smaller than the focal spot of the laser.

I divided my bachelor research in two parts, in the first part we used gold nanoparticles to generate heat in order to create vapor bubbles. In the second part we used gold nanoparticles to generate a very strong local field. The theory and results of those researches will be explained in the following pages.

Part 1

Chapter 2

Theory

2.1 Bubble forming

When nanoparticles are heated, there is a chance that a vapor bubble will be formed. In this research we try to investigate these bubbles. When we consider a bubble formed around a gold nanoparticle to be spherical, we can calculate the surface tension by using the Laplace pressure. The Laplace pressure for a perfectly formed spherical bubble, when we assume a steady state and a liquid-vapor equilibrium, can be derived from the Young-Laplace equation: [4]

$$\Delta p = \frac{2\gamma(T)}{R_b} \tag{2.1}$$

Where:
Δp = the temperature dependent pressure difference across the bubble-fluid interface,
γ = the temperature dependent surface tension,
R_b = is the radius of the bubble.

If we add the atmospheric pressure, we can rewrite equation (2.1) to obtain equation (2.2).

$$\frac{2\gamma(T)}{R_b} + p_0 = p_{vapor}(T) \tag{2.2}$$

Where:
p_0 = atmospheric pressure,
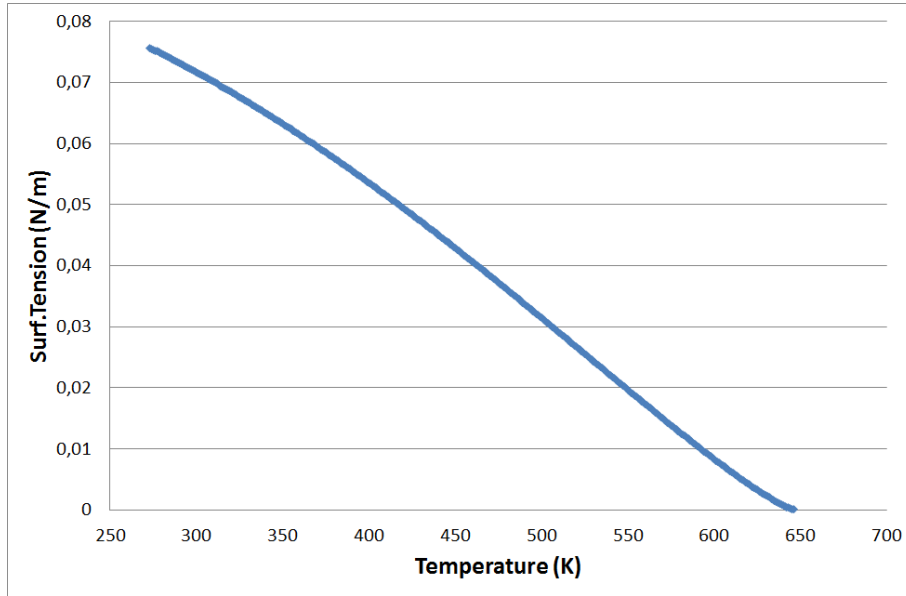p_{vapor} = pressure in the vapor in a steady state.

Figure 2.1: *Surface tension as a function of the temperature for liquid water at saturation. [13]*

When the vapor pressure at the surface is equal to the pressure given in the previous equation when we use the radius of the gold nanosphere as the bubble radius, we find the critical pressure at which a bubble can form. In this thesis we assume a perfect liquid-vapor equilibrium so that when the pressure is equal to or bigger than this value a bubble will be formed and can expand. With this assumption we can rewrite equation (2.2) to obtain equation (2.3).

$$\gamma(T) = \frac{(p_{vapor}(T) - p_0) \times R_p}{2} \tag{2.3}$$

Where:
R_p = radius of the particle.

The pressure in the vapor and the temperature-dependent surface tension can be found in literature and are given in figure 2.1 and 2.2. [13] To find the equilibrium point in a steady state, that is the critical temperature at which a bubble can be formed in this state, we need to find the equilibrium for every radius of the bubble using the values found in the literature and thus the values in figure 2.1 and 2.2. The temperatures when these values are equal to the values given in equation (2.3), are plotted in figure 2.3. For a bubble to form, the water around the particle has to be at least
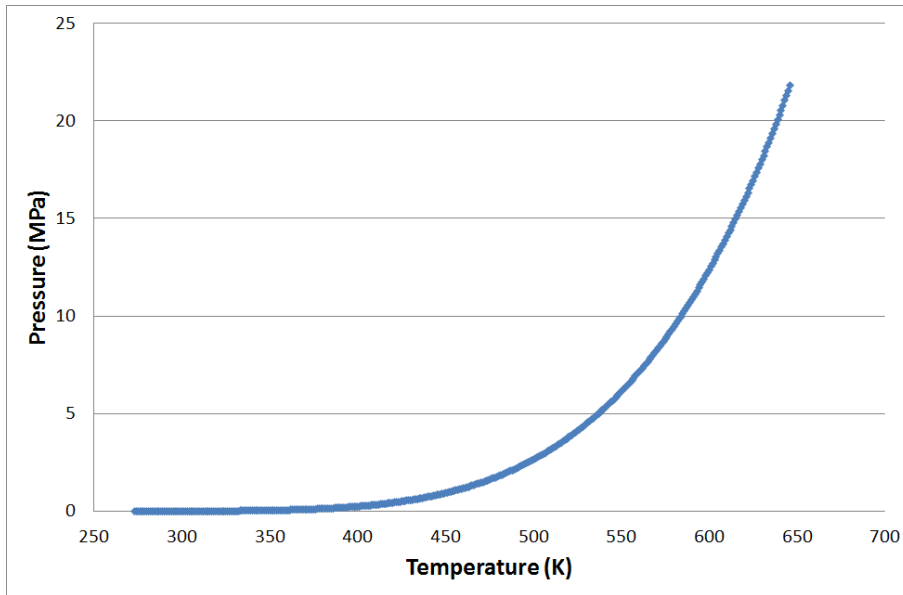
Figure 2.2: *Liquid-gas is in equilibrium with a flat interface. Pressure as a function of the temperature for water. [13]*
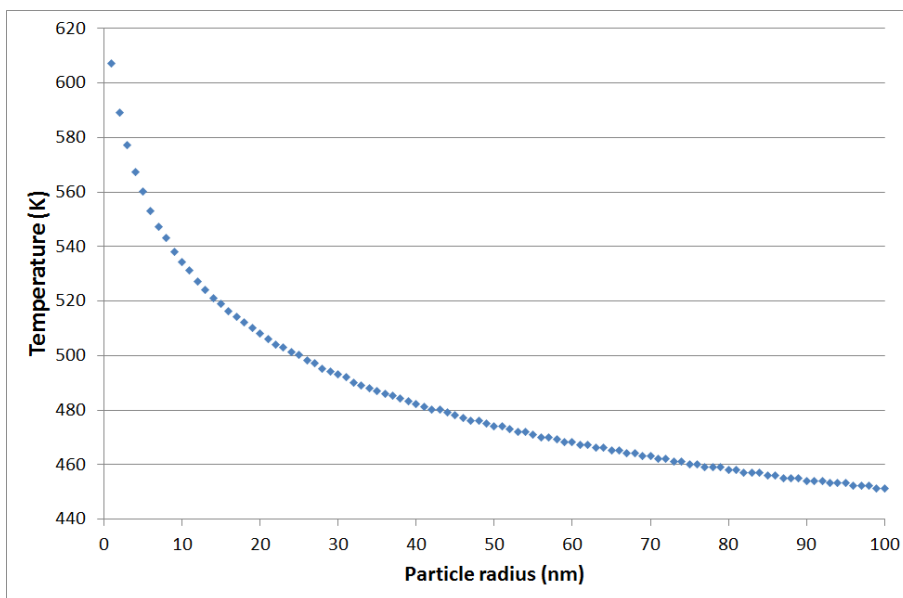

Figure 2.3: *Solutions of equation 2.3 for temperatures between 275K and 650K, this is the minimum temperature at which a stable bubble can be formed.*
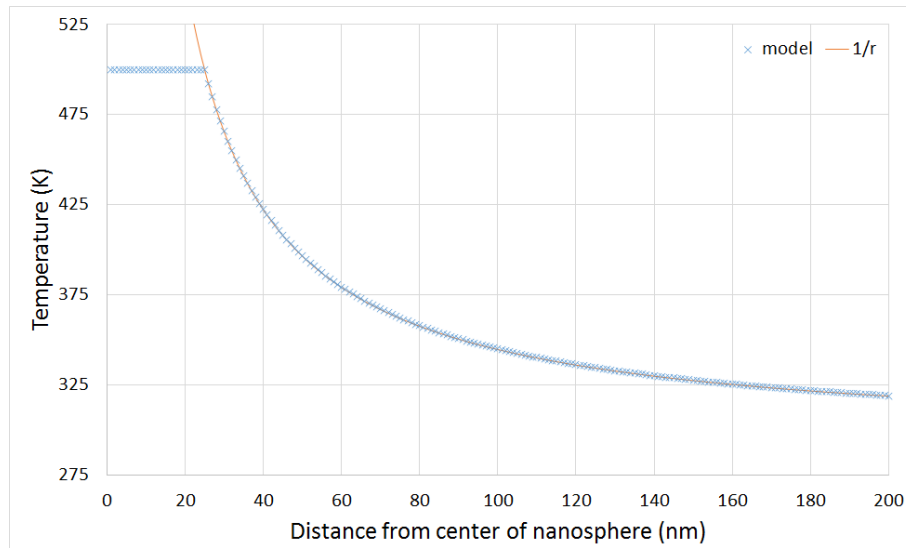
Figure 2.4: *Temperature distribution around a gold nanosphere with a diameter of 50 nm that is placed in an infinitely large water bath kept at 273 K and placed in a laser field with a wavelength of 815 nm, the strength of the electromagnetic field was adjusted so that the center of the nanoparticle would have a steady-state temperature of 500 K. There is a line fit to the graph in the form of $y = a/x + c$ which matches simulation almost perfectly.*

at the critical temperature as can be seen in figure 2.3. As gold is a very good heat conductor, the temperature inside the sphere is nearly uniform under steady state. [13] Because we made the assumption of a steady state and a perfect liquid-vapor equilibrium all of the energy that is put into the nanoparticle will dissipate out of the nanoparticle into the liquid. In a simulation made using the finite method COMSOL it can be seen that the temperature of a heated gold nanosphere is not evenly distributed through the water. In the COMSOL simulation we put a gold nanosphere with a radius of 25 nm in a water bath and heated the particle using a laser with a wavelength of 815 nm. The intensity of the laser was set in such a way that the gold nanoparticle would get a constant temperature of 500 K. So when the particle is in a steady state, the temperature distribution in the water can be seen in figure 2.4.

As can be seen in figure 2.4, the temperature inside the gold nanosphere is distributed evenly. The temperature immediately outside of the gold nanosphere is approximately the same temperature as at the center of the particle. When the temperature of the gold nanoparticle gets higher than the critical temperature found in figure 2.3 a vapor bubble can be formed.
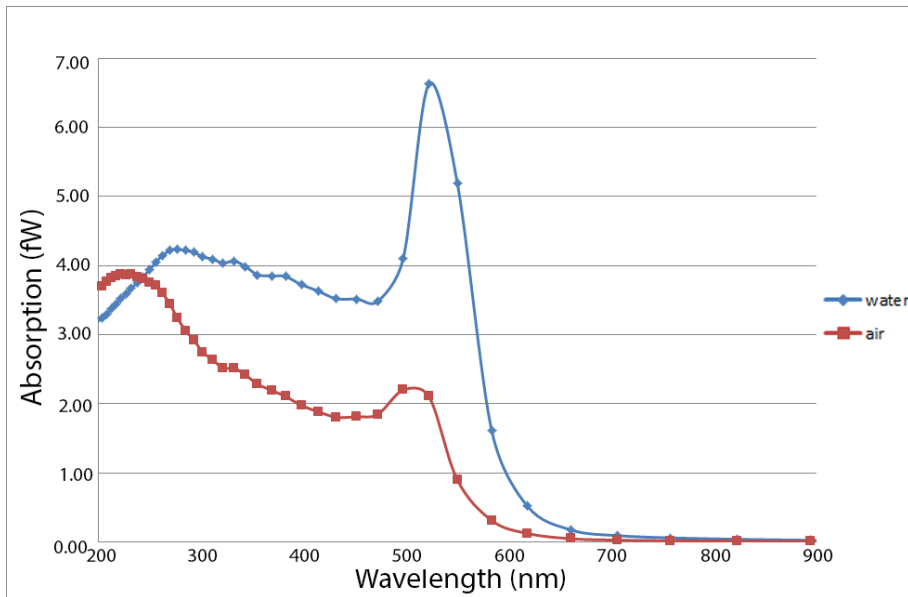
10

Figure 2.5: *When we use the Mie theory for a particle in water and in air we find that the absorption of the gold nanosphere in water is higher than the absorption of air. This calculation is based on a gold nanosphere with a diameter of 50 nm and an electromagnetic field of $1V \cdot m^{-1}$. [16]*

2.2 Spectrum of a gold nanosphere in water

To heat the gold nanosphere we use a laser, because part of the laser light is absorbed by the gold nanoparticle and the particle will be heated up. Not every wavelength is absorbed at the same level and the Mie theory describes the absorption of a particle that interacts with an electromagnetic field. [16] With Matlab it is possible to calculate the behavior of a gold nanosphere in different media. The results of the particle in water and air are plotted in figure 2.5. When the particle is placed in an electromagnetic field the particle will be in water first, but if a bubble forms the particle will become surrounded by vapor. Therefore, it is expected that the absorption of the system will shift from the absorption spectrum of the particle in water, to the absorption spectrum of the particle in air. If we use the Mie theory to calculate the absorption of the nanoparticle in different stages of the bubble growth we find figure 2.6.
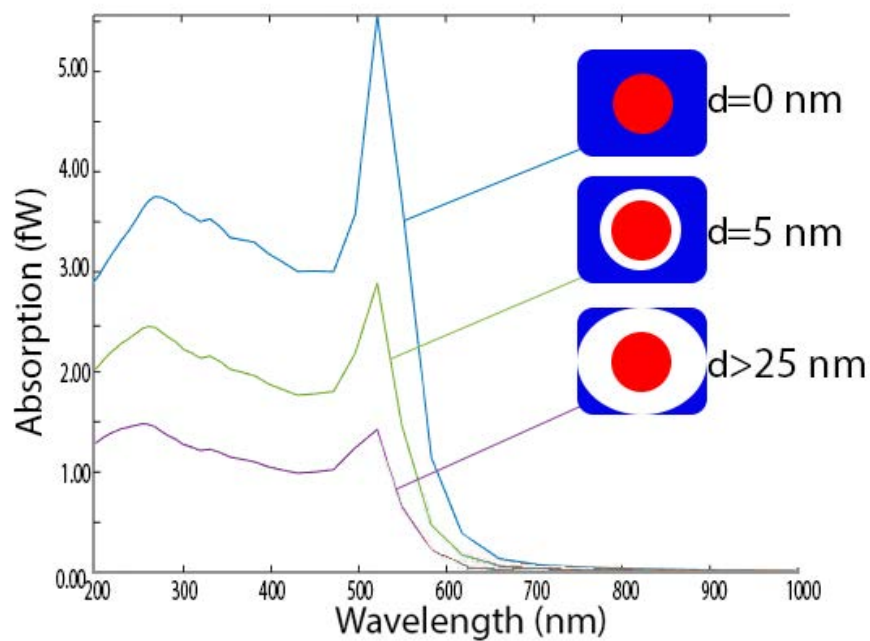
Figure 2.6: *Absorption for the Mie theory for a growing bubble around a gold nanoparticle with a diameter of 50 nm that is placed in an electromagnetic field of $1V \cdot m^{-1}$. The blue line is the absorption of the sphere without a bubble. The green line represents a vapor bubble with 5 nm distance between the edge of the bubble and the surface of the nanoparticle. The purple line represents a thickness of 25 nm and more and closely resembles the absorption spectrum in air. The absorption is highest without a bubble and quickly drops when a vapor bubble is formed. [16] "d" is the thickness of vapor shell.*

Chapter 3

Experimental methods and Results

3.1 Experimental methods

When the critical temperature is reached by the gold nanoparticle, a bubble can be formed around the gold nanoparticle. If a bubble forms the absorption spectrum will change as we saw in figure 2.6. For the measurement of the changes in the absorption spectrum we used a setup with different lasers, as shown in figure 3.1. In the setup laser light with a wavelength of 532 nm is emitted by a laser. This light is separated in two beams by a beam splitter. One beam is pumping a second laser delivering light with a wavelength of 815 nm. The other beam with a wavelength of 532 mn is used to heat the particles in the focus of the beam, the light with a wavelength of 815 nm will be scattered by the gold particles and used to detect the particles. The light passes a lens and is focused on a photodiode which will measure the amount of photons at a set time after the particle is heated. This enables us to observe and detect the bubble forming. We can measure multiple bubbles each at a different time after the heating of the particle started.

3.2 Results and discussion

We did some measurements but the time resolution was not good enough for a good measurement of the bubble forming. With the femtosecond pulse laser, it is possible to increase the time resolution of the experiment. By changing the path length of the detection laser beam, the time difference between the probe and the heating beam is varied. The pump probe technique required different lasers and it took a lot of time to adjust and
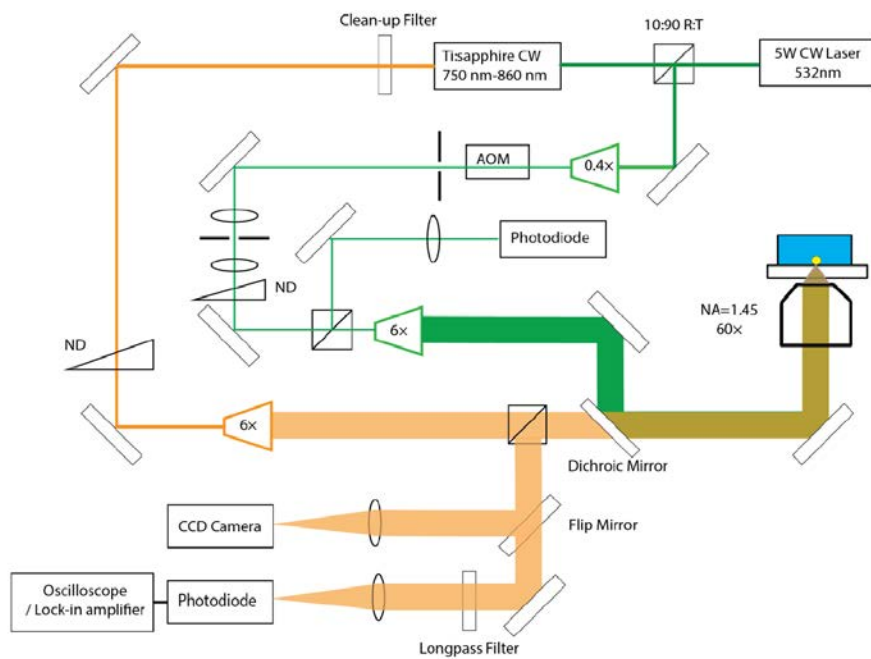
Figure 3.1: *Setup used by me in the experiments described in this part of the thesis. The green beam resembles the laser emitted light with a wavelength of 532 nm and the laser emitted light with a wavelength of 815 nm is shown as salmon orange, there is overlap where both beams overlap this is colored olive green. AOM: acousto-optic modulator, ND: variable neutral-density filter. [12]*

tune the setup. Because of the time limit for the bachelor research I started a second research that consisted of a simulation and is described in part 2 of this thesis.

Part 2

Theory

In the first part we used a spherical gold nanoparticle to generate heat, in this second part we use a rod-shaped nanoparticle to focus the electromagnetic field. The induced electrical-field is changed considerably and can be used for numerous purposes. In this part we use it for single-molecule spectroscopy.

4.1 Single-molecule spectroscopy

In single-molecule spectroscopy the goal is to catch photons that are emitted by a single fluorescent molecule. Each molecule sends out a photon when excited, as can be seen in figure 4.1.

When single-molecule spectroscopy is applied to generate an image, a laser is used to generate the electromagnetic field that is needed to excite the fluorescent molecules. This gives images of single molecules, but when there are a lot of fluorescent molecules in the focal point of the laser then the images will overlap and no single molecule can be identified. This can be solved by using less fluorescent molecules, so that at every moment there is less than one molecule in the focal point of the laser. To get a more detailed image of living cells we need a lot of fluorescent molecules inside of the cell, and we need to reduce the size of the focal point of the laser. This can be done by focusing the electromagnetic field of the laser by using a gold nanoparticle to act as an antenna. [15]

We want to simulate this case, and therefore we want to simulate the electromagnetic field generated by the gold nanoparticle and the movement of the fluorescent molecule around the focal point of the nanoparticle.
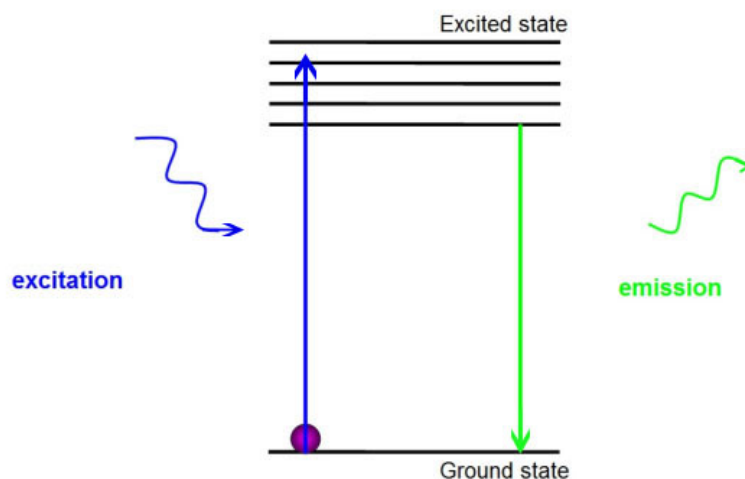
Figure 4.1: *Principle of excitation and emission. When a molecule is placed in an electromagnetic field, there is a chance that it will jump from the ground state to an excited state. Whenever this occurs it can fall back to the ground state, by doing this it can send out a photon. This photon will have a smaller energy thus a longer wavelength than the excitation photon. Source: KHATUA ET AL. ACS NANO, 2014*

4.2 The random walk

When we make the assumption that a fluorescent molecule is moving randomly and the movement is not dependent on the field in which the molecule moves, we can create a random walk. The simplest case of a random walk in Matlab is shown in figure 4.2.

There are some problems with this random walk and our model. One of those problems is the fact that the molecule is totally free to move in any direction, this is the case in real life but in this simulation it would increase the computation time if we were to take in account all the molecule that could enter the near field of the particle. To decrease the simulation time we make the assumption that there is only one molecule in a set space around the particle at a set time. Whenever the molecule leaves the volume boundary, it will bounce back. To implement this, we need to change the way we create the random walk, the way I choose to do this is by using a for-loop. In a for-loop, a certain code is run until a boundary condition is met. This is not as fast as the previous code but it gives us the freedom to control every step of the molecule during the walk instead of just getting the positions of the molecule. In the for loop we constantly check the
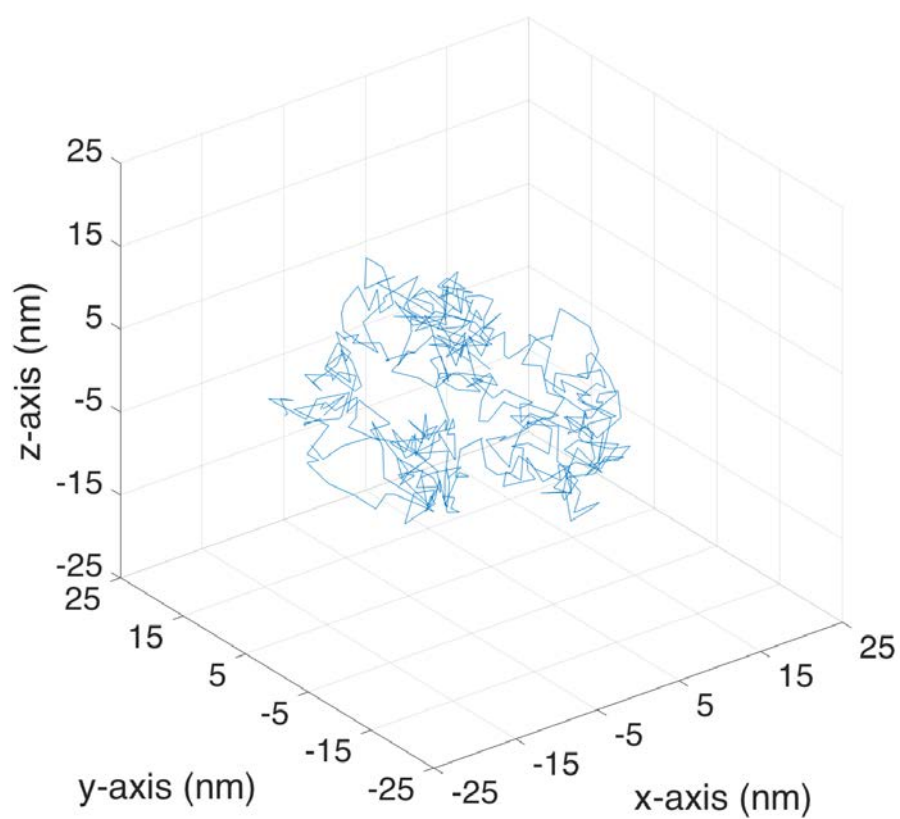
Figure 4.2: *Example of a random walk. In this walk the molecule is released at the point 0,0,0 and is free to move in any direction with the maximum distance of 5 nm from the previous point. The total number of steps taken by the molecule is 500 steps.*
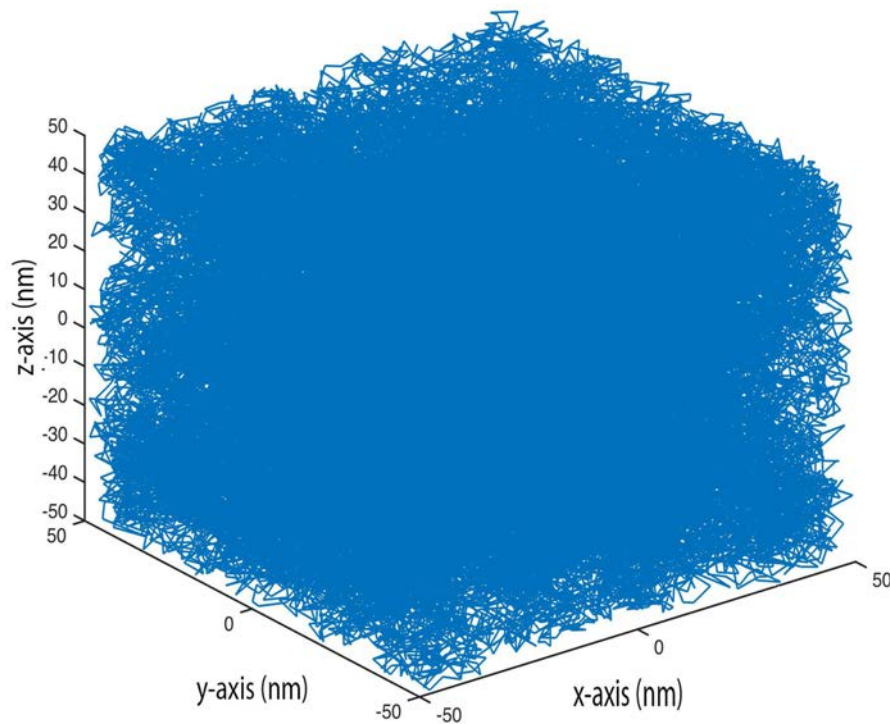
Figure 4.3: *Random walk in which every position the molecule can occupy is within a box with a size of 50 nm and with 10000 steps in 3 dimensions and whereby the distance between the previous point in every direction is between 0 nm and 5 nm.*

position of the molecule and whenever it reaches the boundaries it must retake the step. This means that the molecule can never end a step outside of the box.

This meets our approximation where there is only one molecule at any time within the boundaries of the box. But if we want to introduce a gold nanorod inside our box, we must make sure that our molecule does not enter the gold nanorod. Therefore, we must add another boundary to our script that prevents the molecule from entering the rod. First we must add the boundary condition that the center of the particle is at the center of the box, therefore at position (0,0,0). It follows that we cannot let our molecule start at the middle, otherwise it would start inside of the nano rod. Therefore, we must also generate a starting position, we will do this at random within the set boundaries, therefore the molecule will start at a random position inside the box but outside of the particle. The particle itself will be described by using the symmetry of the rod. If we place the rod along the y-axis then if the total length of the rod is a and the radius of

the rod is b, then the center of the circle on the end of the rod is at position (a/2-b) on the y-axis. It follows that every position that meets one of the following conditions is inside the rod.

$$\left.\begin{array}{c} \sqrt{x^2 + (|y| - (a/2 - b)))^2)} < r, \\ x^2 < r^2 \quad and \quad y^2 < (a/2 - b)^2 \end{array}\right\} \qquad (4.1)$$

If we implement this in our code, we obtain figure 4.4.

The full code file for the random walk can be found in appendix A.

4.3 electric-field map

First we will introduce our molecule into a very simple field. One simple case is the case of a far field, the field of a Gaussian beam. The field of a Gaussian beam is very similar to the field that is generated by a focused laser beam. The goal is to make a time trace of the molecule and the intensity of the electric-field that it has been in. First we will introduce the Gaussian distribution of the electric field. [21]

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \cdot exp(-\frac{(x - \mu)^2}{2\sigma^2}) \qquad (4.2)$$

Where:
σ = variance,
μ = expectation value

This is an analytical equation, and this can easily be used in our model, but when we want to calculate an E-field for our walker we can better use an lattice in which our molecule can walk, this way we only have to check the position instead of calculating the value for every step. If we try to write this into our program it makes most sense to start with the 2 dimensional case and to calculate he E field for every position of the lattice. This means we rewrite x as the distance to the center and we calculate every possible position to find figure 4.5 .

If we want to introduce the random walk into this electric-field, we can use the look-up table for every position of the molecule and find the corresponding electric-field value. If we store this we find figure 4.6

Now we will try to find a good electric-field map for a nanorod. The first model we use is created using COMSOL, this program uses the finite element method to calculate the generated electric-field map.
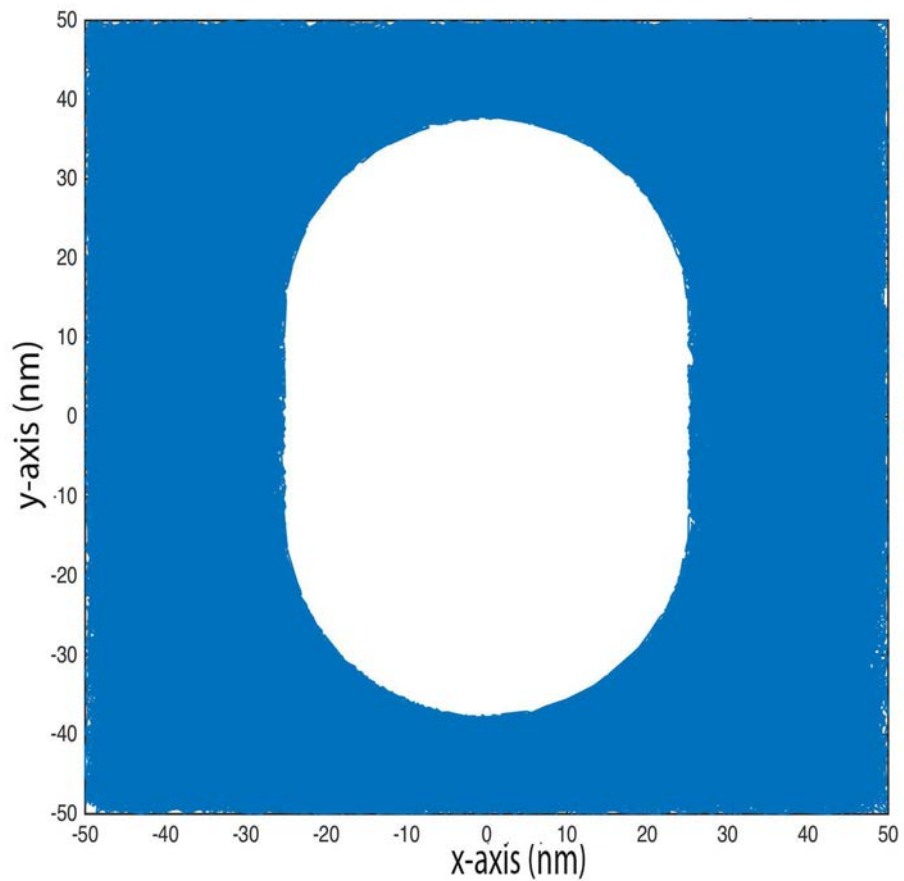
Figure 4.4: *This graph shows the results of the walker within the box with a size 50 nm and with 10000 steps in 2 dimensions and whereby the distance between the previous point in every direction is between 0 nm and 5 nm, also there is a zone which will contain a gold nanorod, which the molecule cannot enter, this zone is determined by the boundaries set in equation (4.1) with a length of 75 nm and a diameter of 50 nm.*

Figure 4.5: *Lattice of the normalized electric-field with a size of 100 nm and in 2 dimensions and whereby the size of every position is 1 nm. The electric field is generated by using the formula (4.2) for every position.*

Figure 4.6: *We used the random walk from figure 4.4 and the field map of figure 4.5 to generate the signal shown above.*

The program of COMSOL generates a mesh which is made up of triangles rather than squares and all the triangles are of a different size, to use this file in Matlab we must change everythi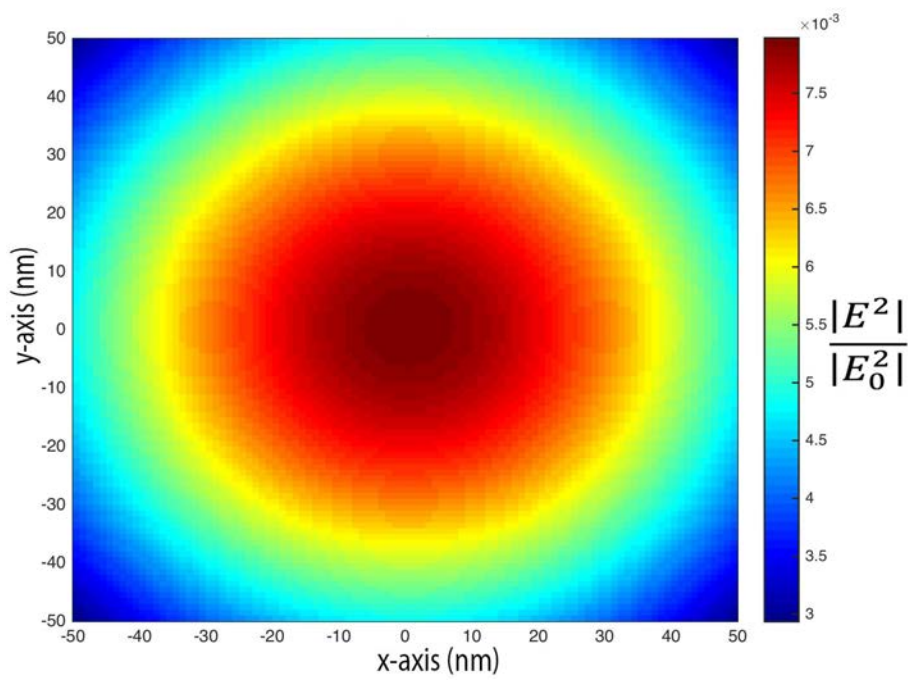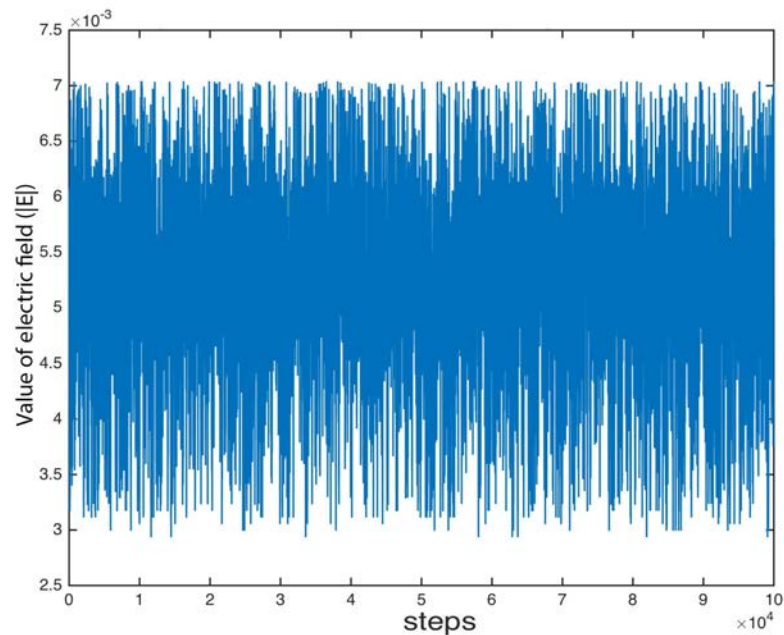ng in a way that we can read the COMSOL file into a lattice made of squares. The transformation gives a lattice that is not of a very good resolution and takes a long time to calculate. In order to get a faster and more accurate electric field, we used the MNPBEM Matlab toolbox. [10] [9] [20] The MNPBEM toolbox can be used for the simulation of metallic nanoparticles, using a boundary element method approach which is developed by F. J. Garcia de Abajo and A. Howie [3]. When we use this method we can generate the electric field faster, in this simulation we choose this method but any electric field can be used for the simulation. By using the MNPBEM Matlab toolbox and the code as found in appendix B, we obtain figure 4.8.

If we compare this to literature values, as can be seen in figure 4.9, we find that the generated field matches the electric-field found in literature. When the 2 dimensional random walk is introduced to the field in figure 4.8 and we store all the values of the field we obtain figure 4.10. [15]

26
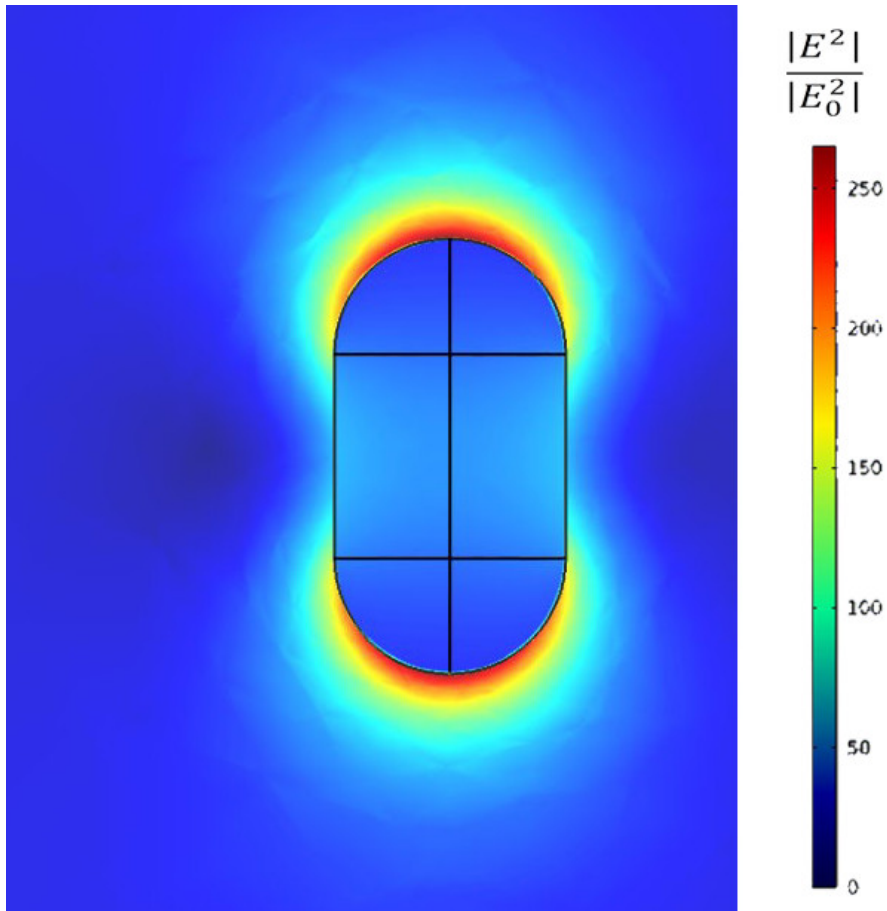
Figure 4.7: *The generated electric-field map generated using the finite element method of COMSOL. The rod has a diameter of 25 nm and a length of 47 nm. The particle is placed in a laser field with the intensity of $1 W \cdot m^{-2}$ that has a wavelength of 633 nm.*
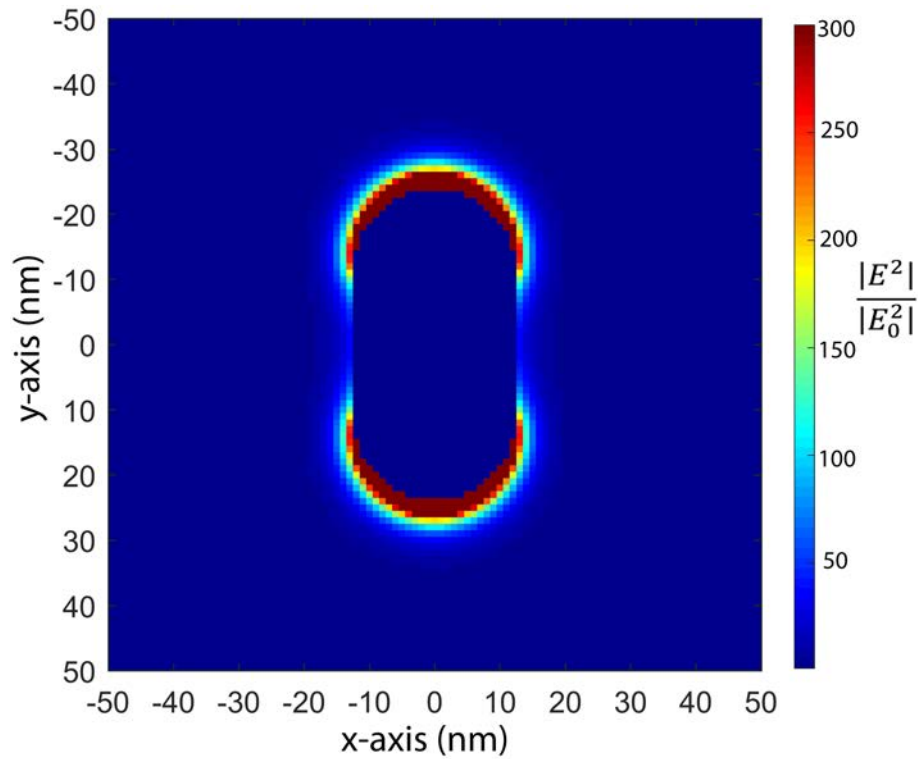
Figure 4.8: *Normalized results of the MNPBEM approximation. The simulation is used on a rod with a diameter of 25 nm and a length of 47 nm. The particle is placed in a laser field that has a wavelength of 633 nm and a polarization along the y-axis, the used code can be found in appendix B.*

Figure 4.9: *Near-field intensity distribution of nanorods under 633 nm excitation with a polarization along the y-axis. The width of all the nanorods is 25 nm and the lengths are shown below the intensity distribution of the nanorod. [15]*

Figure 4.10: *The value of the electric field shown in figure 4.8 for every step in a random walk. The random walk is performed for 10000 steps in 2 dimensions and with a step size of 5 nm and a box size of 50 nm in every direction.*

4.4 Generating photon output and signal

Now we have the path of the molecule and the electric field strength at each point we can describe the output of single photons as a result of that field. When the molecule is at a location with a higher intensity then the chance of a photon being emitted is higher. [15] There is not a constant time between photons that is determined by the intensity but rather an average time between two consecutive photons being released. If we calculate the probability density from the Decay Law we find equation (4.3). [8]

$$p(\tau) = \gamma e^{-\gamma \tau} \tag{4.3}$$

Where:
τ = time between released photon,
γ = constant

If we call the value $u = e^{-\gamma \tau}$, and $du = -\gamma u d\tau$ we can rewrite equation (4.3) to equation (4.4).

$$g(u) = p(\tau)\frac{d\tau}{du} = \gamma u \frac{1}{\gamma u} = 1 \tag{4.4}$$

Combining equation (4.3) and equation (4.4) gives equation (4.5).

$$\tau = \frac{1}{\gamma} |log(u)| \tag{4.5}$$

U is a random number between 0 and 1. If we put this in a Matlab code, we obtain the code as can be found in appendix C, and we will find a photon output graph as can be found in figure 4.11.

By counting all the photons that are released in a given time we can simulate the signal that is generated by an ideal photon counting detector. The counting script can be found in appendix D and generates figure 4.12.

4.5 Further calculations on the photon output

The data that is found previously should, when using the right input data, match the data of the experiments but it is very hard to draw any sensible conclusion when looking at the data in figure 4.12. But if we bin the time that has passed between two consecutive photons we find the results as can be found in figure 4.13.
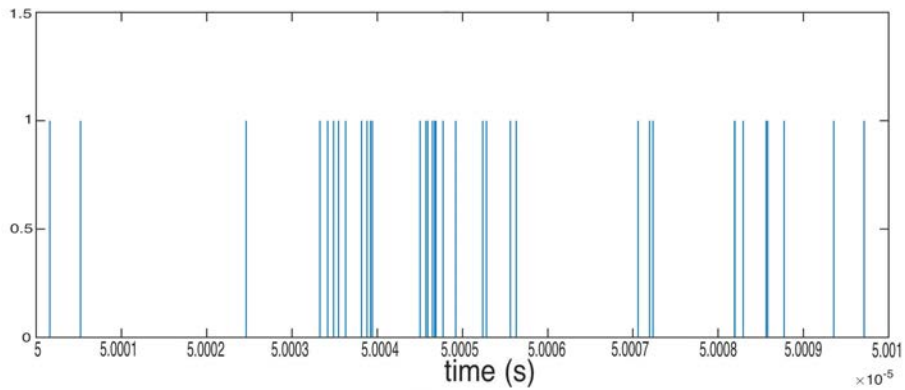
Figure 4.11: *Graph of the photon output for the times between 50000 ns and 50010 ns, every blue line is a photon. This is generated in an electricfield simulated using the MNPBEM approximation, with a rod with a diameter of 25 nm and a length of 47 nm. The particle is placed in a laser field that has a wavelength of 633 n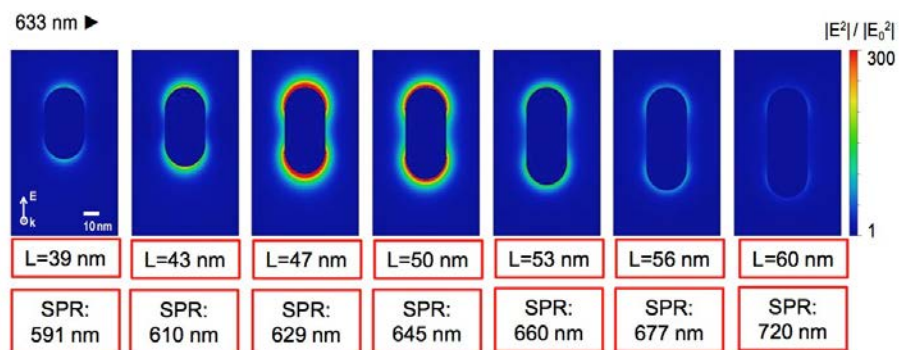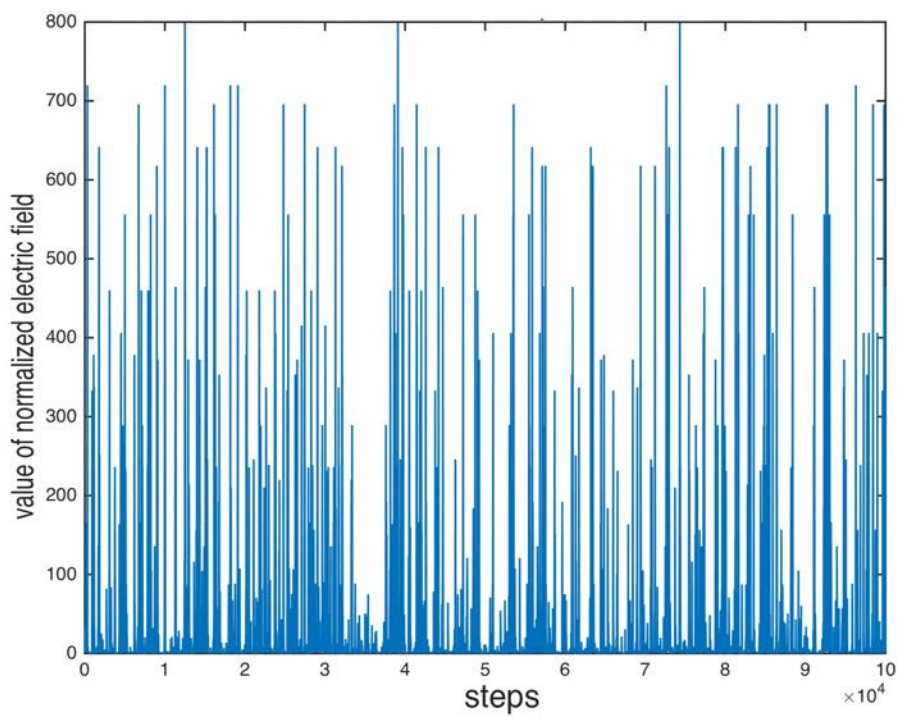m and a polarization along the y-axis. The random walk is performed for 10000 steps in 2 dimensions and with a step size of 5 nm and the time between steps is equal to 1 ns and a box size of 50 nm in every direction.*
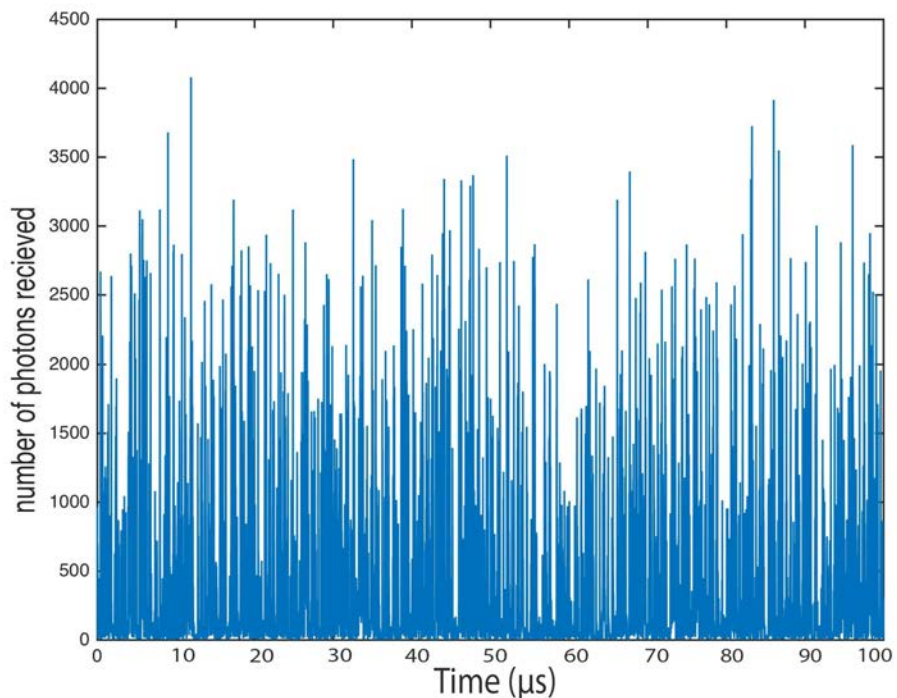
Figure 4.12: *Simulated signal that can be expected to find with a detector with a binning time of 10 ns, with the data as found in figure 4.11.*
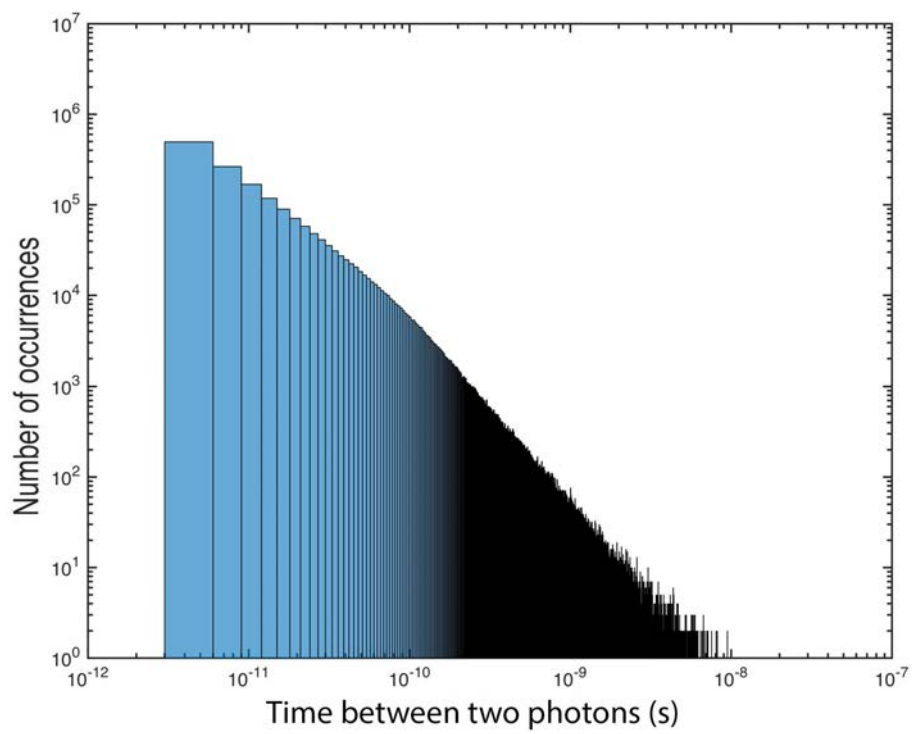
Figure 4.13: *Histogram of delay times between two consecutive photons as found in figure 4.11.*

The model can generate a random walker for a molecule in a box with a rod-shaped particle in the middle, it is able to generate an electrical-field map, it can simulate the photon output and generate a signal as could be expected to be found with a detector and it is able to make a histogram with the time between photons. To validate the simulation we compare the results of our model with actual measurements in the next chapter.

5

Chapter

Experimental methods

5.1 Confocal microscope setup

For the measurements of the fluorescent molecules a homebuilt confocal microscope was used. A 639 nm pulsed laser was passed through a narrowband clean-up filter and was then focused on an immersion objective. The sample was fixed on a stage that was controlled by piezo elements to move the sample around. The emission was collected through the same objective and filtered through an emission filter. The emission was focused on a pinhole and then focused on a single photon counting module. [15]

5.2 Simulation starting values

To compare the simulation results with the actual measurements we must first make an educated guess about the input values. The nanoparticle used in the measurements had a length of 100 nm, a diameter of 45 nm and the laser light that was used to excite the particle had a wavelength of 639 nm. The polarization is along the z-axis which is the same as the symmetry axis of the gold nanorod. The Matlab program on the computer I used could not simulate more than 10^8 steps, this is because of memory limitations of the computer, which will result in a simulation time of approximately 0.1 second. This is too short to generate a good graph since the measurements normally consist of data taken in a timeframe of 200 seconds or more. To be able to compare the simulated values with the actual measurement data we normalized the results so that the area of both graphs are the same.

6

Results and Analysis

6.1 Comparison in Gaussian field

When we look at figure 6.1 we can see that both graphs look the same and that the simulation gives a close representation of the actual photons that are being emitted without the presence of a gold nanorod. The simulation has a slightly higher number of short time counts between photons, this could be due the detector being not able to count every photon.

6.2 comparison with the gold nanorod

When we compare the simulated photon counts with the measurements we find that the simulation does not match the measurement data completely. In the data found by the simulation the time between photons is a lot smaller, and also the number of photons does not reach zero for very short times. This difference can be due to the detector not being able to detect every photon in the measurements separately when they arrive in very short times in between, the simulation does count every individual photon. When looking at figure 6.2 and 6.1 we can see that the simulation gives a close representation of the Gaussian field and a rough representation of the field in the presence of a gold nanorod. The values presented in figure 6.2 and 6.1 are generated after the normalization of the results found in the simulation to match the values found in measurement, it only shows a fraction of the graph in order to focus on the most crucial part.

Figure 6.1: *The red line shows the results of measurements as found by Biswajit Pradhan. The dye is ATTO 647N, manufactured by ATTO-TEC and is placed in a Gaussian laser field with a FWHM of 300 nm that has a wavelength of 639 nm and a polarization along the z-axis. The data is collected for a time span of 200 s. The blue line shows the normalized results of a simulation of the dye in a Gaussian field with a FWHM of 300 nm with a wavelength of 639 nm. The total simulation time is equal to 10 seconds.*
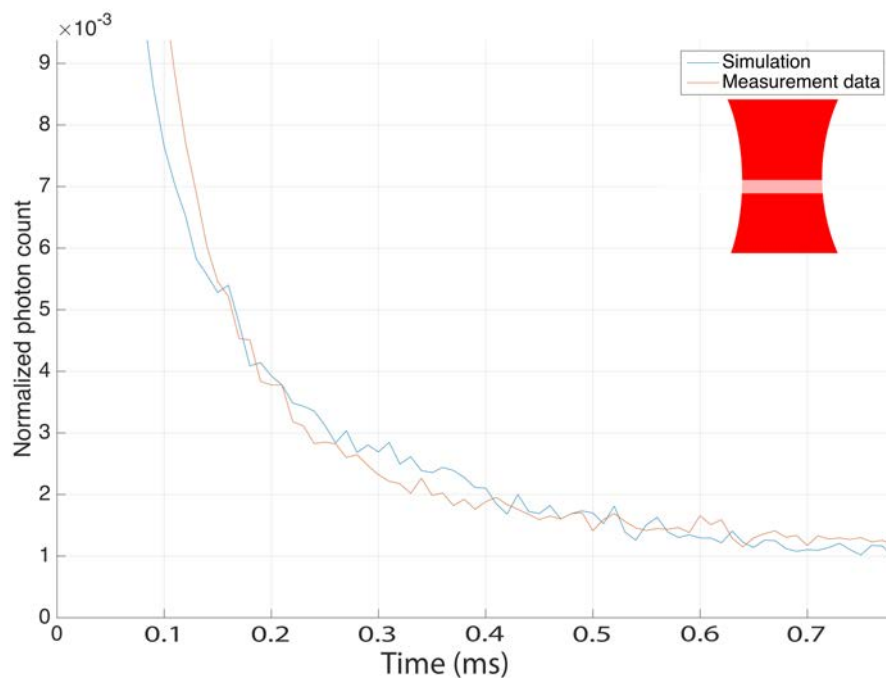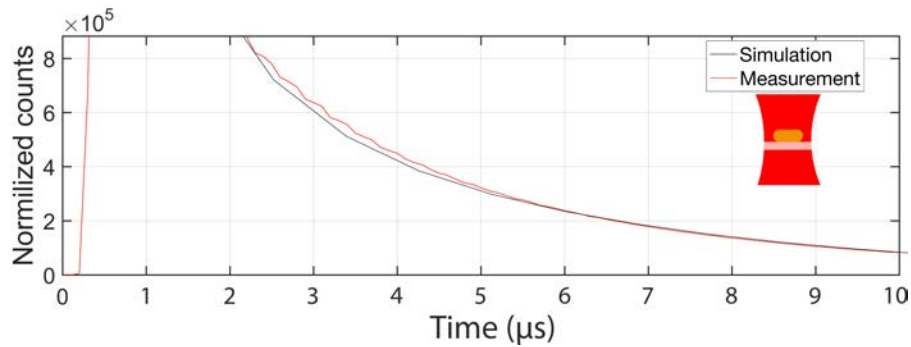
Figure 6.2: *The red line shows the results of measurements as found by Biswajit Pradhan. The dye is ATTO 647N, manufactured by ATTO-TEC and with a rod with a diameter of 45 nm and a length of 100 nm. The particle is placed in a laser field that has a wavelength of 639 nm and a polarization along the z-axis. The data is collected for a time span of 200 s. The blue line shows the normalized results of a simulation of the dye around a gold nanorod. This is generated in an electronic field simulated using the MNPBEM approximation, that simulated a period of 10 seconds.*

6.3   Comparison photon counts

When we compare the photons per time unit for the simulated values and the measured values we find values that are alike. It can be seen that figure 6.3 and 6.4 look similar. But there is a difference, this could be due the delay time that limits the detector. If the detector just received a photon, it is blinded for a set time, this is needed to charge the detector again. This will influence the measurement but will not have an influence on the simulation. Therefore, the simulated number of photons can be expected to be higher than the number of photons found in an actual measurement especially for a higher number of photons per time unit.
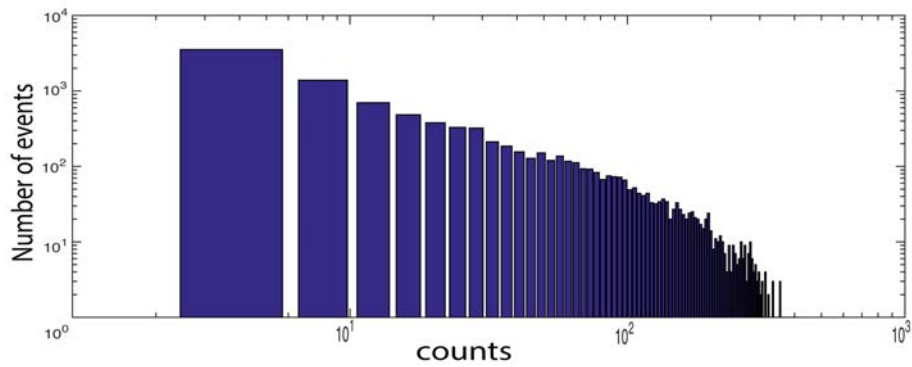
Figure 6.3: *Results of a simulation of the dye around a gold nanorod. This is generated in an electric field simulated using the MNPBEM approximation, with a rod with a diameter of 45 nm and a length of 100 nm. The particle is placed in a laser field that has a wavelength of 639 nm and a polarization along the z-axis. The random walk is performed for 10^{10} steps in 3 dimensions and with a step size of 100 nm and the time between steps is equal to 1 ns and a box size of 100 nm, 30 nm and 75 nm. The total time simulated is equal to 0.1 seconds. The photon count is binned every 10 ns.*
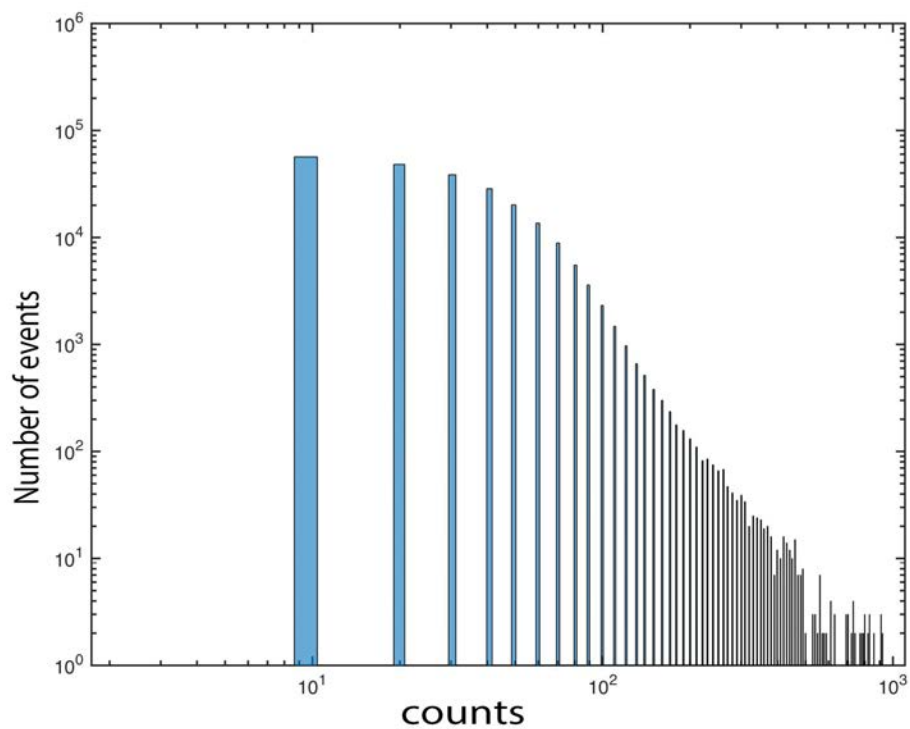
Figure 6.4: *Results of measurements as found by Biswajit Pradhan, represents the binned data from the photon counts. The dye is ATTO 647N, manufactured by ATTO-TEC and with a rod with a diameter of 45 nm and a length of 100 nm. The particle is placed in a laser field that has a wavelength of 639 nm and a polarization along the z-axis.*

Chapter 7

Conclusions and discussion

The results of the simulation around the gold nanorod do not perfectly match all of the measurements. There are some changes needed to improve the simulation, one of the most urgent things to change is the optimization of the calculation speed. I think that an optimization can be made by using the MATLAB parallel computing plugin, when used correctly it is able to use the calculation power of the GPU. The GPU is likely to perform better than the CPU for the random walk since it is not a hard computation but rather a very repetitive one in which the GPU excels. [17]

If the computation time could be decreased, it is also possible to run the simulation in a bigger box and with that include the influence of the Gaussian beam (which influence will increase for a bigger box) and to generate a walk for multiple particles. This can explain why the measurement declines at a different rate than the simulation near the gold nanorod for a short time. When there are multiple particles near the gold nanoparticle multiple photons will be released and are all counted. As a result the time between the photons will be shorter.

Besides this physical limitation of the computer we did not include all the known science in the simulation. The most important scientific phenomenon we did not include is called quenching. This happens when a particle is close to the surface of the gold nanoparticle, there is then another option for the particle to go from the excited state to the ground state, it can transfer its energy to the gold particle without releasing a photon. The gold particle will then dissipate this extra energy in the form of heat. This will decrease the number of photons when the particle is close to the nanoparticle. Besides this it would also improve the simulation if we were to include the Quantum yield and Bleaching and then perform the simulation for multiple particles. Quantum yield is the probability of emission of

a photon by the excited state. Bleaching occurs when the molecule due to photon-induced damage and covalent modification is not fluorescent anymore. This would especially be interesting to see for a multiple particle simulation. The simulation is not totally complete in describing the physical system and its outcome, but it does give a good estimation of what to expect. The reason that there are some things to add to the simulation is mainly due to the relatively short time frame in which we made this simulation. I think it would be a good project to optimize the simulation and to improve it further.

Appendices

code: random walk

```matlab
function [ results ] = randomwalk( vars, results )
    vars.a = waitbar(2/8,'Random walk');
        if vars.dim == 3
        results.A=zeros(vars.S,3);
        h = waitbar(0,'Starting random walk');
        t=0;
        while abs(results.A(1,1))>vars.X_max || abs(results.A(1,2))>vars.Y_max
|| abs(results.A(1,3))>vars.Z_max || results.A(1,2)< 0 ||
(sqrt(results.A(1,1)^2+results.A(1,2)^2+(abs(results.A(1,3))-(((vars.hoogte-
vars.diameter)*1e-9)/2))^2)) < (vars.diameter*1e-9)/2 ||
abs(results.A(1,3))<(vars.hoogte*1e-9)/2 &&
sqrt(results.A(1,1)^2+results.A(1,2)^2)<(vars.diameter*1e-9)/2
            results.A(1,1)=vars.X_max*(rand*2-1);
            results.A(1,2)=vars.Y_max*rand;
            results.A(1,3)=vars.Z_max*(rand*2-1);
        end
        for k=2:vars.S
            results.A(k,1)=results.A(k-1,1)+vars.v*(rand*2-1);
            results.A(k,2)=results.A(k-1,2)+vars.v*(rand*2-1);
            results.A(k,3)=results.A(k-1,3)+vars.v*(rand*2-1);
            while abs(results.A(k,1))>vars.X_max ||
abs(results.A(k,2))>vars.Y_max || abs(results.A(k,3))>vars.Z_max ||
results.A(k,2)< 0 ||
(sqrt(results.A(k,1)^2+results.A(k,2)^2+(abs(results.A(k,3))-(((vars.hoogte-
vars.diameter)*1e-9)/2))^2)) < (vars.diameter*1e-9)/2 ||
abs(results.A(k,3))<(vars.hoogte*1e-9)/2 &&
sqrt(results.A(k,1)^2+results.A(k,2)^2)<(vars.diameter*1e-9)/2
                results.A(k,1)=results.A(k-1,1)+vars.v*(rand*2-1);
                results.A(k,2)=results.A(k-1,2)+vars.v*(rand*2-1);
                results.A(k,3)=results.A(k-1,3)+vars.v*(rand*2-1);
            end
            %waitbar
            t=t+1;
            if t>vars.S/100;
                t=0;
                perc=k/vars.S;
                waitbar(perc,h,sprintf('(%d%%) random
walk',round(100*perc,0)))
            end
        end
        close (h)    %closes waitbar when done with walk
        end
        if vars.dim == 2
        results.A=zeros(vars.S,2);
        h = waitbar(0,'Starting random walk');
        t=0;
        while abs(results.A(1,1))>vars.X_max  ||
abs(results.A(1,2))>vars.Z_max  ||
(sqrt(results.A(1,1)^2+(abs(results.A(1,2))-(((vars.hoogte-vars.diameter)*1e-
9)/2))^2)) < (vars.diameter*1e-9)/2 || abs(results.A(1,2))<((vars.hoogte-
vars.diameter)*1e-9)/2 && abs(results.A(1,1))<(vars.diameter*1e-9)/2
            results.A(1,1)=vars.X_max*(rand*2-1);
            results.A(1,2)=vars.Z_max*(rand*2-1);
        end
        for k=2:vars.S
            results.A(k,1)=results.A(k-1,1)+vars.v*(rand*2-1);
            results.A(k,2)=results.A(k-1,2)+vars.v*(rand*2-1);
            while abs(results.A(k,1))>vars.X_max  ||
```

```matlab
                    abs(results.A(k,2))>vars.Z_max   ||
(sqrt(results.A(k,1)^2+(abs(results.A(k,2))-(((vars.hoogte-vars.diameter)*1e-
9)/2))^2)) < (vars.diameter*1e-9)/2 || abs(results.A(k,2))<((vars.hoogte-
vars.diameter)*1e-9)/2 && abs(results.A(k,1))<(vars.diameter*1e-9)/2
                    results.A(k,1)=results.A(k-1,1)+vars.v*(rand*2-1);
                    results.A(k,2)=results.A(k-1,2)+vars.v*(rand*2-1);
                end
                %waitbar
                t=t+1;
                if t>vars.S/100;
                    t=0;
                    perc=k/vars.S;
                    waitbar(perc,h,sprintf('(%d%%) random
walk',round(100*perc,0)))
                end
            end
        end
        close (h)     %closes waitbar when done with walk
        end
    close(vars.a)
end
```

B

code: electricfield

```matlab
function [ results] = generating_electricfield( vars, results)
    vars.a = waitbar(3/8,'Generating electricfield');
        Z_size=vars.Z_max*1e9;
        if vars.dim == 3
        M_size=sqrt((vars.X_max*1e9)^2+(vars.Y_max*1e9)^2);
        elseif vars.dim == 2
        M_size=vars.X_max*1e9;
        end
        op = bemoptions( 'sim', 'ret' );
        part=trirod( vars.diameter, vars.hoogte, [ 20, 20, 20 ] );
        epstab= {epsconst(1),epstable('gold.dat')};
        results.p = comparticle( epstab, { part }, [ 2, 1 ], 1, op );
        exc = planewave( vars.light_pol, vars.light_dir, op );
        bem = bemsolver( results.p, op );
        results.sig = bem \ exc( results.p, vars.Wavelength );
        [results.m,results.z] = meshgrid(-M_size:vars.M_res:M_size,-
Z_size:vars.M_res:Z_size);
        emesh = meshfield( results.p, 0, results.m, results.z, op, 'mindist',
0.2, 'nmax', 3000, 'waitbar', 1 );
        e = emesh( results.sig );
        E=dot(e,e,3);
        results.E=E/E(1,1);
    close(vars.a)
end
```

C

code: photons output

```matlab
function [ results ] = photonoutput( vars,results)
    vars.a = waitbar(5/8,'Photon output');
        TEMP=NaN((ceil(max(results.Ef)/vars.Fe)*vars.S),1);
        ppp=0;
        t0=results.Ef./vars.Fe;
        h = waitbar(0,'Starting photonout');
        t=0;
        for p=1:vars.S
            pp=0;
            while pp<=vars.steptime
                pp=pp+(vars.steptime./t0(p,1))*abs(log(rand));
                if pp<=vars.steptime
                    ppp=ppp+1;
                    TEMP(ppp)=(p-1)*vars.steptime+pp;
                end
            end
            t=t+1;
            if t>vars.S/100;
                t=0;
                perc=p/vars.S;
                waitbar(perc,h,sprintf('(%d%%) photonout',round(100*perc,0)))
            end
        end
        results.t=NaN(length(TEMP)-sum(isnan(TEMP)),1);
        for p=1:length(results.t)
            results.t(p)=TEMP(p);
        end
        results.dt=diff(results.t);
        close (h)
    close(vars.a)
end
```

Appendix **D**

code: photonbinning

```matlab
function [ results ] = photonbinning( vars, results )
    vars.a = waitbar(6/8,'Photon binning');
    results.Sig=zeros(ceil(max(results.t)/vars.time_int),1);
    TEMP=ceil(results.t./vars.time_int);
    TEMP2=length(results.Sig);
    h = waitbar(0,'Starting photonmeash');
    t=0;
    for p=1:TEMP2
        results.Sig(p)=(sum(TEMP(:) == p));
        t=t+1;
        if t>TEMP2/100;
            t=0;
            perc=p/TEMP2;
            waitbar(perc,h,sprintf('(%d%%) photonmeash',round(100*perc,0)))
        end
    end
    close(h)
    close(vars.a)
end
```

References

[1] D. J. Barbier and Freestone. An investigation of the origin of the colour of the Lycurgus cup by analytical electron microscopy. *Archaeometry*, 32:33–45, 1990.

[2] R.G. Churnside and P.M.G. Proffitt. The Rothschild Lycurgus cup; an analytical investigation. *J. glass studies*, 5:18–23, 1963.

[3] F. J. Garcia de Abajo and A. Howie. Kinetic pathways leading to layer-by-layer growth from hyperthermal atoms: A multibillion time step molecular dynamics study. *Phys. Rev. B*, 65(115418), 2002.

[4] Pierre-Gilles de Gennes, Françoise Brochard-Wyart, and David Quere. *Capillary and Wetting Phenomena—Drops, Bubbles, Pearls, Waves*. Springer, 2002. ISBN 0-387-00592-7.

[5] Ian Freestone, Nigel Meeks, Margaret Sax, and Catherine Higgitt. The Lycurgus cup - a Roman nanotechnology. *Gold Bulletin*, 40/4:270–277, 2007.

[6] D. B. Harden and Jocelyn M. C. Toynbee. VII.—The Rothschild Lycurgus cup. *Archaeologia (Second Series)*, 97:179–212, 1 1959.

[7] Donald B. Harden, Hansgerd Hellenkamper, Kenneth Painter, and David Whitehouse. Glass of the Caesar. *Antiquity*, 62:400–402, 6 1988.

[8] Eugene Hecht. *Optics*. Addison Wesley, 2002. ISBN 978-0-321-18878-0.

[9] U. Hohenester. Simulating electron energy loss spectroscopy with the mnpbem toolbox. *Comp. Phys. Commun.*, 185(1177), 2014.

[10] U. Hohenester and A. Trügler. MNPBEM? A Matlab toolbox for the simulation of plasmonic nanoparticles. *Comp. Phys. Commun.*, 183(370), 2012.

[11] Helmuth Horvath. Gustav Mie and the scattering and absorption of light by particles: Historic developments and basics. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 110:787–799, 2009.

[12] Lei Hou, Mustafa Yorulmaz, Nico R Verhart, and Michel Orrit. Explosive formation and dynamics of vapor nanobubbles around a continuously heated gold nanosphere. *New Journal of Physics*, 17(013050), 2015.

[13] http://webbook.nist.gov/chemistry/. NIST chemistry webbook. Accessed May 04, 2014.

[14] X. Huang, I. H. El-Sayed, W. Qian, and M.A. El-Sayed. Cancer cell imaging and photothermal therapy in the near-infrared region by using gold nanorods. *J. Am. Chem. Soc.*, 128:2115–2120, 2006.

[15] Saumyakanti Khatua, Pedro M. R. Paulo, Haifeng Yuan, Ankur Gupta, Peter Zijlstra, and Michel Orrit. Resonant plasmonic enhancement of single-molecule fluorescence by individual gold nanorods. *ACS NANO*, 2014.

[16] Christian Mätzler. *MATLAB Functions for Mie Scattering and Absorption*. Institut für Angewandte Physik, June 2002. Research Report No. 2002-08.

[17] David A. Patterson and John L. Hennessy. *Computer Organization and Design*. Morgan Kaufmann, 2012. ISBN 978-0-12-374750-1.

[18] G. Rashke, S. Kowarik, T. Franzl, C. Sönnichsen, T.A. Klar, J. Feldmann, A. Nichtl, and K. Kürzinger. Biomolecular recognition based on single gold nanoparticle light scattering. *Nanoletters*, 3:935–938, 2003.

[19] Paul v. Ruijgrok. *Optical manipulation and study of single gold nanoparticles in solution*. PhD thesis, Casimir PhD Series, May 2012. ISBN: 978-90-8593-122-5.

[20] J. Waxenegger, A. Trügler, and U. Hohenester. Plasmonics simulations with the MNPBEM toolbox: Consideration of substrates and layer structures. *Comp. Phys. Commun.*, 193(138), 2015.

[21] Daijiro Yoshioka. *Statistical physics, An Introduction*. Springer, 2007. ISBN 978-3-540-28605-9.