# Optical Trapping and Inspection of Nanoparticles with Double-Nanohole Optical Traps

By

Skyler J. Wheaton

B.Sc., University of Washington, 2012

A Thesis Submitted in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Physics and Astronomy

# Supervisory Committee

Optical Trapping and Inspection of Nanoparticles with Double-Nanohole Optical Traps

By

Skyler J. Wheaton

B.Sc., University of Washington, 2012

**Supervisory Committee**

**Dr. Reuven Gorodon, Supervisor**

(Department of Electrical and Computer Engineering)

**Dr. Geoffrey Steeves, Departmental Member**

(Department of Physics and Astronomy)

# Abstract

**Supervisory Committee**

**Dr. Reuven Gorodon, Supervisor**

(Department of Electrical and Computer Engineering)

**Dr. Geoffrey Steeves, Departmental Member**

(Department of Physics and Astronomy)

This thesis presents the optical trapping of various nanometric particles (both biological and non-biological) and methods that can be used to extract information about the trapped particle from the signal transmitted through a nanoaperture trap. These methods are used to detect the excitation of vibrational modes in trapped particles due to the presence of a beat signal between two tunable trapping lasers and the molecular weight of the particle by examining the transmitted signal.

Optical trapping has long been used to trap ever smaller particles in gentle non-destructive ways. In its infancy, only the optical trapping of micron sized particles was feasible. Due to various limitations, changes to the optical trapping scheme were needed to push its limits into the nanometric regime. Nanoaperture assisted optical trapping has allowed for the optical trapping of particles as small as 5 nm in diameter. By making use of specially chosen nanoapertures in gold films higher trapping strengths with lower incident laser powers have become possible. While this is an accomplishment in and of itself there are several issues associated with working with such small systems. Most notably, the ability to observe such

systems is very limited. Traditional optical trapping of micron sized particles could make easy use of optical inspection, however in the nanometric regime this is not possible. It has since become a focus of the trapping community to find sophisticated ways to use the limited data available to probe these systems and their trapped targets.

Once a particle is trapped the only information available about the particle is contained in the signal transmitted through the nanoaperture. The first main area of research in this thesis covers using this information to extract the molecular weight of the trapped particles for identification. In the same vein, Raman has been a tool widely used in the past to identify and probe systems of large ensembles of particles. While this is incredibly effective in some situations, it is not effective at the single particle limit. To form an analog that can be used within an optical trapping setup a new method of exciting Raman active vibrational modes with twin trapping lasers is presented. The low wavenumber vibrational spectra are presented for several different particles as well as a wide array of globular proteins.

# Contents

# List of Figures

# Acknowledgments

I would like to profusely thank my supervisor Dr. Reuven Gordon for his unyielding support, insight, drive, and patience. He has been the primary driving force behind my growth and development as a scientist during my time in his lab. His insight was the foundation of the primary experiments on which this thesis is based. His guidance and style have served as a model for me to work towards to improve myself and become the best researcher I can. I am truly grateful. In addition to my supervisor I would like to thank my supervisory committee comprised of Dr. Geoffery Steeves and Dr. Stephanie Willerth, for their advice and support during my time here at the University of Victoria.

I would like to thank the fellow members of my lab for all of their continued support and advice. Ahmed Al Balushi, Abhay Kotnala, Ryan M. Gelfand, and Yashaswini Rajashekara. Their kindness and willingness to share their insight has been paramount in my understanding and the development of my work.

Lastly I would like to thank my family for all the sacrifices they have made to give me the opportunity to be the person I am today, I will always be grateful for their unyielding support and love.

# Chapter 1

# Introduction

The overarching topic of this thesis will be on the analysis and extraction of information from the signal collected from an aperture based trapping system. This chapter will cover the basic concepts of optical trapping as they are applied to the research presented later. The basics of the dynamics of a particle in an aperture trap will be introduced along with how they relate to the properties of the trapped particle.

## 1.1 Basic Concepts of Optical Trapping

Optical trapping was initially used to control and reliably contain micron sized particles by exploiting the gradient force of a highly focused beam of light[1]. This technique has continued to grow since its inception and the limits of its abilities have been reached, and subsequently broken, multiple times. It has been used for the placement and three dimensional manipulation of micron sized objects[1], the gentle trapping of large viruses, and recently the trapping of nanometric particles[2].

The key difficulty with optical trapping is the careful balance needed between using enough incident power to properly contain the particle while remaining in a range of incident power that will not be damaging to the particle. Considering the trapping power needed to

contain a particle scales as the radius cubed[1], this becomes a particularly imminent problem as the radius of target particle approaches the sub 100 micron range. In order to account for these limitations as well as to beat the diffraction limit of light, plasmonic trapping schemes have been used. Initially 100 micron particles were trapped by exciting localized surface plasmons in gold disks deposited on glass[3]. Moving forward gold nano antennas were used to make highly focused hot spots that were used to trap sub-micron particles[3]. Lastly we approach the trapping scheme that will be covered extensively in this thesis, nanoaperture assisted optical trapping. Using this approach, nanoapertures have been used to trap particles with a radius as small as 2 nm. This trapping deemed Double Nano-Hole (DNH) optical trapping is the basis of the trapping used in my experiments which seek to extract more useful information out of these trapping events.

## 1.2 Advanced DNH Trapping

These past works have focused mainly on achieving proper and sustained trapping of ever smaller particles. However there has been a lack of ability to probe the particles once they are trapped. It is of particular difficulty to inspect such a small environment in any form. In a standard DNH trapping experiment the only accessible data is the transmission through the nanoaperture. Hidden in the transmission is a rich set of information on the particle and its motion inside the trap. By carefully analyzing this data this small amount of information can be used for powerful analysis of the trapped particles.

By advancing on the typical DNH trapping scheme to use two trapping lasers as opposed to a single laser a whole new set of information can be extracted from the transmission signal. By tuning two trapping laser sources a beat frequency between them can be used to excite low

frequency acoustic vibrational modes of trapped particles. This information is particularly potent because it can directly identify trapped particles swiftly and easily with no cost on the strength or reliability of trapping.

## 1.3 Organization of This Thesis

This thesis will first cover the underlying theory of optical trapping that is employed throughout this thesis. Chapter 2 will cover the expansion of the basic optical trapping scheme to nanoaperture assisted trapping and the theory involved.

After the theory has been covered the cornerstone experiments this thesis is based on will be covered in heavy detail with emphasis on the data collected and its relevance to the physics of the system. Chapter 4 will conclude the thesis with potential future research opportunities.

## 1.4 Author's Contributions

This thesis is based off of a collection of projects that have been a result of the author's personal work. All of which have either been published or submitted to peer reviewed journals for assessment. The contributions of the author are provided below:

### 1.4.1 Double Nanohole Acoustic Raman [4]

The previous experiments conducted by my group have been primarily based on the successful trapping and the qualification of different nanoparticles within a DNH trap. Until

recently actually identifying trapped particles has not been possible. The ability to identify trapped particles reliably is paramount for the ability to work in inhomogeneous solutions or to confirm flow channel results.

Here we study the ability to excite Raman active modes of trapped nanoparticles that are contained within a DNH optical trap. The first run of the experiment shows early data obtained from trapped 20 nm polystyrene particles. The resonance peaks are compared to those predicted by Lamb's theory, also known as the elastic continuum model. Following this, proteins of varying size and shape also have their Raman spectra swept and shown. Currently there is little data to compare these spectra to while there has been little work conducted in the acoustic vibration region of proteins. The experiment was designed by Skyler Wheaton, Ryan Gelfand and Reuven Gordon. The experiments and data processing were carried out by Skyler Wheaton. The manuscript was written by Skyler Wheaton and Reuven Gordon.

## 1.4.2 Molecular Weight Measurement of Proteins Trapped in DNH Optical Trap [5]

This work focused on trying to find ways that the traditional DNH trapping setup could be used to extract more specific information about the trapped particles, with little to no modification of the original setup. By careful examination of the minute features in the transmission signal such as the root mean square fluctuation and the autocorrelation time constant it was found that these quantities had a clear dependence on the molecular weight of the trapped particle. The basis of these findings were compared to a basic hydrodynamic model of the DNH trap, showing good agreement. The experiment was designed by Skyler Wheaton and

Reuven Gordon. The experiments and data processing were carried out by Skyler Wheaton. The manuscript was written by Skyler Wheaton and Reuven Gordon.

### 1.4.3 Optical Trapping Review [6]

This work was a review piece written for analyst on the basic background of nanoaperture optical trapping applied to sensing and discovery. It contains a thorough history of nanoaperture trapping and showcases both its strengths and weaknesses. The manuscript was written primarily by Ahmed Al Balushi, with contributions from Skyler Wheaton, Abhay Kotnala, Ryan Gelfand, and Yashaswini Rajashekara.

# Chapter 2

# Theory

This chapter will be used to cover the overarching theory behind optical tweezers and subsequently optical nano-tweezers. The theory of optical trapping will be covered in two specific regimes, one where the particle size is significantly greater than the wavelength of light, another where the particle's size is much smaller than the wavelength of light being used to trap the particle. Moving towards nano scale trapping Bethe's nano-aperture theory is covered and then applied to nano scale trapping.

## 2.1 Optical Trapping Fundamentals

The key premise of optical trapping is to exploit the interaction between the electric field gradient due to a highly focused beam and a dielectric particle. The goal is to create a small potential well that can gently hold a dielectric particle in place with small forces, usually on the scale of pN-fN [7]. In a highly focused beam there will be varying beam widths at different points along the direction of propagation. The region with the smallest width is known as the beam's waist. It is this waist that the particle will be drawn to. This net force in the direction of the waist is a known result of a particle in a gradient electric field[1].This gradient force scales with the polarizability of the particle and the gradient of the field as follows:

$$F_{grad} = -\frac{n_b}{2}\,\alpha \nabla E^2$$

where $\alpha = n_b^2 r^3 \left(\frac{m^2-1}{m^2-2}\right)$, $n_b$ is the refractive index of the surrounding medium, and *m* is the effective refractive index of the particle. The particle now submerged in this gradient field will be drawn to the point of highest electric field intensity, or in this case, the central focus point of the beam. In fact, in this setup there are multiple forces acting on the particle, not simply just the gradient force. The key to obtain trapping is to balance the scattering force, the force due to the momentum transfer of photons striking the particle, and the gradient force. The scattering force which is equal to:

$$F_{scat} = \frac{I_0}{c}\frac{128\pi^5 r^6}{3\lambda^4}\left(\frac{m^2-1}{m^2+2}\right)^2 n_b$$

where c is the speed of light, $I_0$ is the intensity of the incident light, and $\lambda$ is the wavelength of the incident light, will create a net motion of the particle in the direction of propagation, while the gradient force will continue to attract the particle to the beam waist. With these properties in mind one can consider releasing a particle into the trap slightly off center, the motion one would expect is in fact simple harmonic motion.

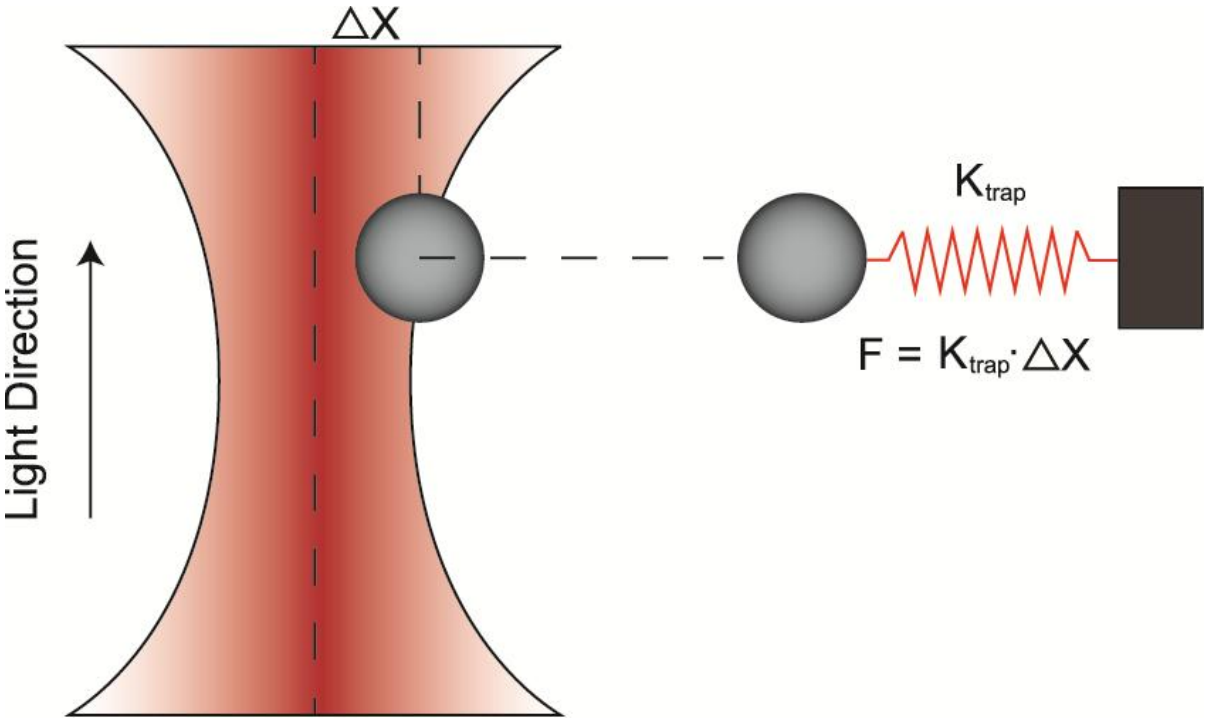


**Figure 1 | Trapping kinematics.** A schematic of the trapping fundamentals is shown. The particle's offset from the beam's center and waist is used in the spring approximation. The force on the particle is estimated using Hooke's law.

The basic equations of this form of optical trapping follow very closely with a traditional ball and spring system. This is depicted in Figure 1. We can think of the optical trap as a 3 dimensional simple harmonic oscillator with 3 spring constant (k) values, one for x, y, z. the Kx and Ky values will be based simply off of the gradient force, while Kz will take into account the scattering and gradient force. Following this analogy further it is expected that the force applied to the particle by the trap will scale linearly with the particle's distance from the point of maximum field intensity.

## 2.2 Optical Trapping Analysis

To analyze the dynamics of optical trapping behavior, one starts with consideration of the size of the target particle and the wavelength of light to be used. In an instance where the size of the particle is much greater than the wavelength of light being used to trap it, simple ray optics modeling of the particle can sufficiently describe the trapping behavior. When the particle's size is smaller than the wavelength of light being used to trap, Maxwell's equations in their time independent form with proper boundary conditions can be used.

## 2.3 Ray Optics Regime Modeling

When the particle that is being trapped is much larger than the wavelength of the light being used to create the trapping potential a ray optics approach can be used to explain the trapping dynamics. As rays of light enter and exit the dielectric bead they are refracted, and such the direction of the ray that exits the bead is not in most cases the same as the direction of the incident ray. This change in direction indicates a change of momentum in the system, by Newton's third law we can assert that an equal and opposite change in the particle's momentum must have occurred. This change in momentum will draw the particle to the region of highest intensity, in the case of an optical trap the beam will be focused so that the center of the beam at the beams waist will be the point of highest intensity and thus the point where the particle is attracted/confined to. When the particle is directly at the center of the beam and located near the waist the change in momentum is symmetric around either side of the particle and will thus contain no lateral component. At this point only an axial force remains, this force is balanced out by the scattering force of the light on the particle at a point just below the waist opposite the

direction of propagation. These mechanics of the optical trapping system are outlined in Figure 2.

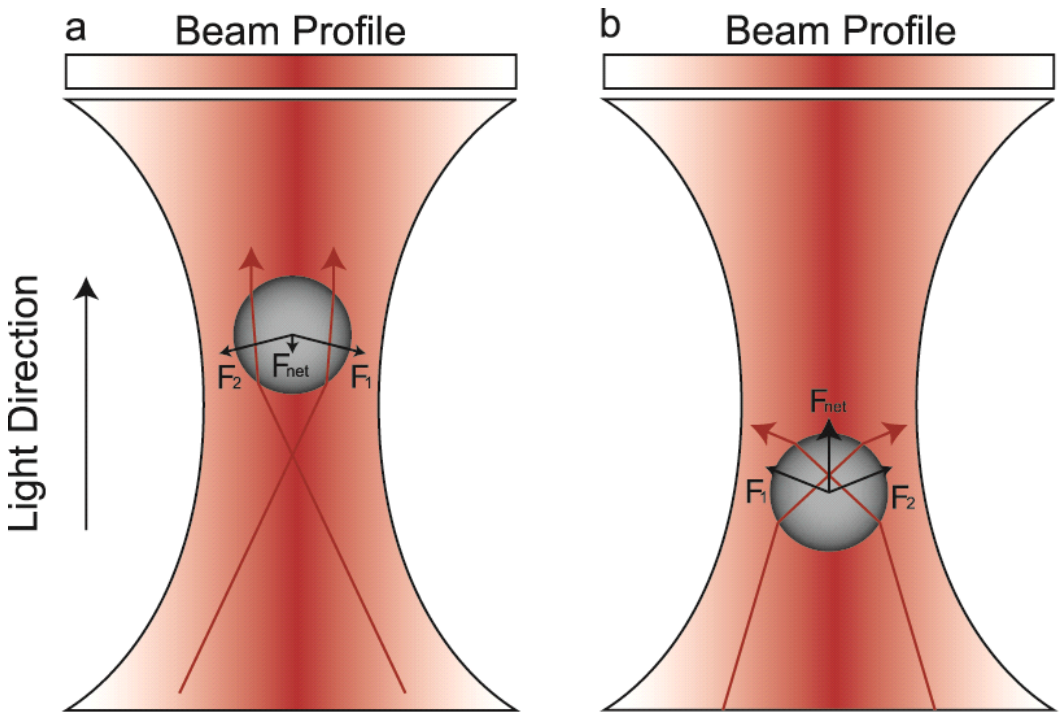


**Figure 2 | Diagram of trapping fundamentals.** a) Diagram of particle situated at the center of the beam above the waist, net force is weak towards the center. b) Diagram of the particle below the waist, net force pulls the particle strongly up to the center.

## 2.4 Bethe's Theory

When the object to be trapped is much smaller than the wavelength of light being used to create the trapping potential, the time dependence of Maxwell's equations can be neglected. In this regime Maxwell's equations can be solved by using electrostatic boundary matching conditions. This approach can be applied to the situation of light attempting to propagate through

a sub wavelength size aperture. In this situation the light will be cutoff in the aperture and will be unable to pass through. Instead, the light will be diffracted at the edge of the aperture[8].

The theory of diffraction for small apertures was first derived by Hans Bethe in 1944. Bethe explored the problem of light diffraction in sub-wavelength apertures in an infinite plane. Bethe's results are most easily reproduced by starting with the quasi-static approximation of Maxwell's equations. In this approximation the system is a plane wave incident on a circular aperture with a diameter much shorter than the wavelength of the incident light on an infinite perfect electric conductor (PEC). We consider the plane wave incident normally to the PEC plane and the electric and magnetic field parallel to the PEC plane. As mentioned this system, given the size of the aperture being much smaller than the wavelength of the incident light, can be solved using electrostatic boundary condition matching. The light that will be diffracted by the aperture can be approximated as the emission from a magnetic dipole centered at the aperture. The effective magnetic dipole moment of the aperture in response to the incident plane wave can be calculated by solving the boundary conditions of the magnetic potential at both sides of the PEC plane[9]. The power emitted by this dipole is a good model for the transmitted power through the hole. In free-space, the optical transmission is expressed as:

$$T = \frac{4Z_0\pi^3 n^4}{6\lambda_0^4}\left(\frac{8r^3}{3}H_0\right)^2 \propto \frac{r^6}{\lambda^4}$$

Where $Z_0$ is the free-space impedance, $\lambda_0$ is the wavelength in free-space, r is the aperture's radius, $H_0$ is the magnetic field of the incident wave and $n$ is the refractive index of the surrounding medium. After normalizing to the area of the hole, an important conclusion of Bethe's theory can be obtained: the optical transmission through a sub-wavelength aperture is

inversely proportional to the fourth power of the light wavelength, that is $T \propto \left(\frac{r}{\lambda}\right)^4$. In the situation where the aperture is filled with/surrounded by a dielectric medium with refractive index $n_1 > n$ we see that the transmission through the aperture will increase by a factor of $(n_1 - n)^4$.

Similarly to the ray optics modeling of focused beam optical trapping an analysis of the increase in transmission can be associated with a change in the net momentum of the photons through the aperture. As the particle attempts to leave the aperture, the optical size of the aperture will decrease, lowering the transmission. As the transmission is lowered the net photon momentum through the aperture is decreased. Considering Newton's third law, it is expected that an equal and opposite change in momentum should result on the particle effectively pulling it back into the trap. This phenomenon has been coined self-induced back action trapping, or SIBA[10]. DNH optical trapping is simply SIBA with a highly efficient aperture designed to create a localized trapping region. The dynamics of this effect are schematically shown in Figure 3.

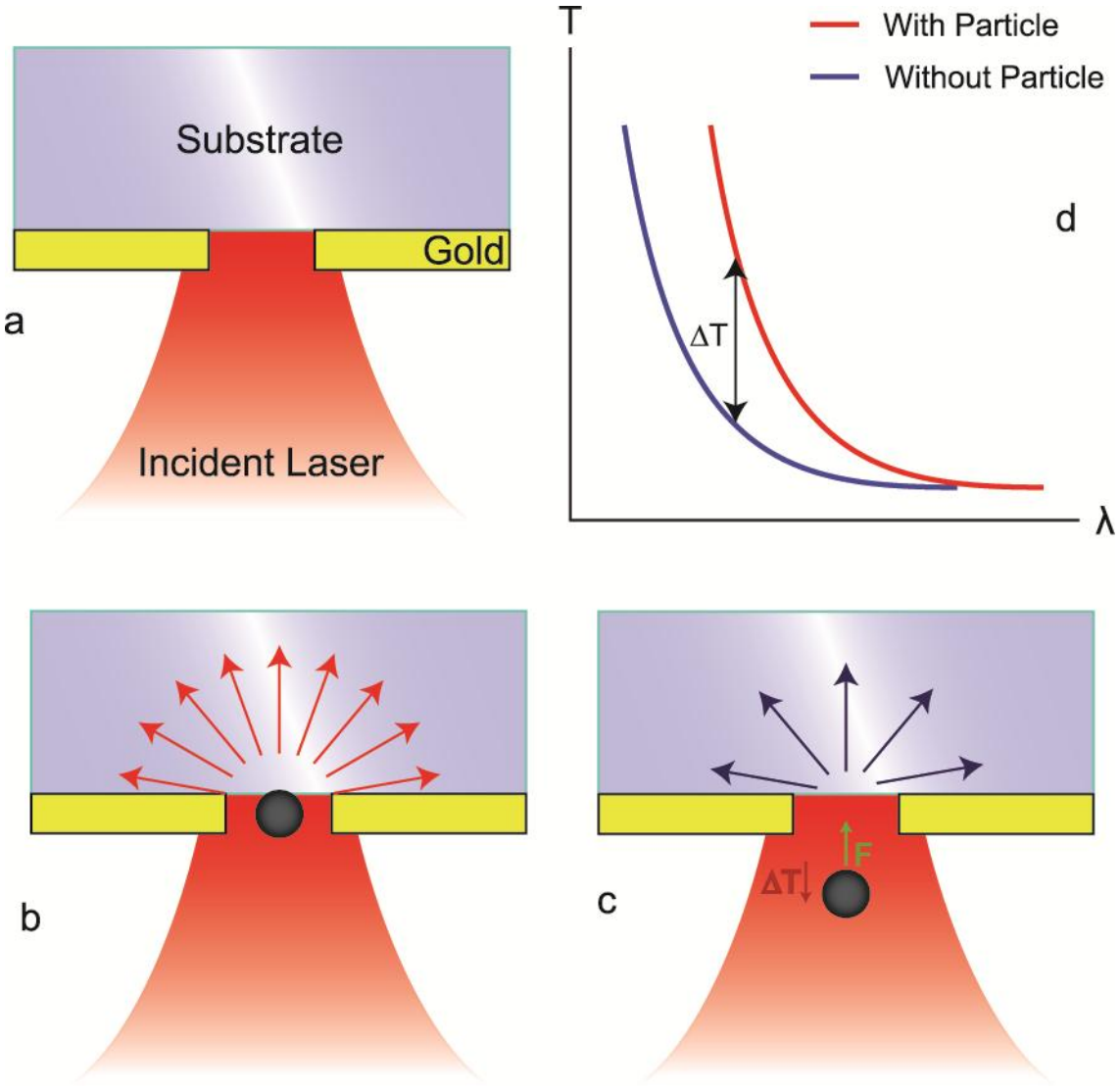


**Figure 3 | Diagram of aperture and nanoparticle transmission.** a) Basic setup showing an incident laser source on an aperture in gold. b) System with particle introduced to aperture, large transmission shown in red. c) As the particle moves away, the transmission is reduced, force towards the aperture is increased. d) Example of transmission curves of the aperture with and without a nanoparticle present.

# Chapter 3

# Methods and Experiments

This chapter will cover the basics of the methods used in all of the aforementioned experiments. The fabrication of the DNHs will be covered. Then the standard DNH tweezer setup will be covered along with the sample preparation process. The experiment to measure the molecular weight of proteins from the transmission signal in a DNH tweezer is discussed along with the results. Following that the DNH-EAR setup will be discussed along with a discussion of the data.

## 3.1 Double Nanohole Fabrication

The DNHs were fabricated using a focused ion beam mill (FIB). The machine used in the fabrication process is the Hitach FB-2100 Focused-Ion Beam system. Each pattern is prepared as a 2000 by 2000 bitmap binary image. Pixels labeled as 0 are exposed to the ion beam while pixels with a value of 1 are left untouched. An example of a fabrication pattern is shown in Figure 4 alongside the resulting fabricated trap.

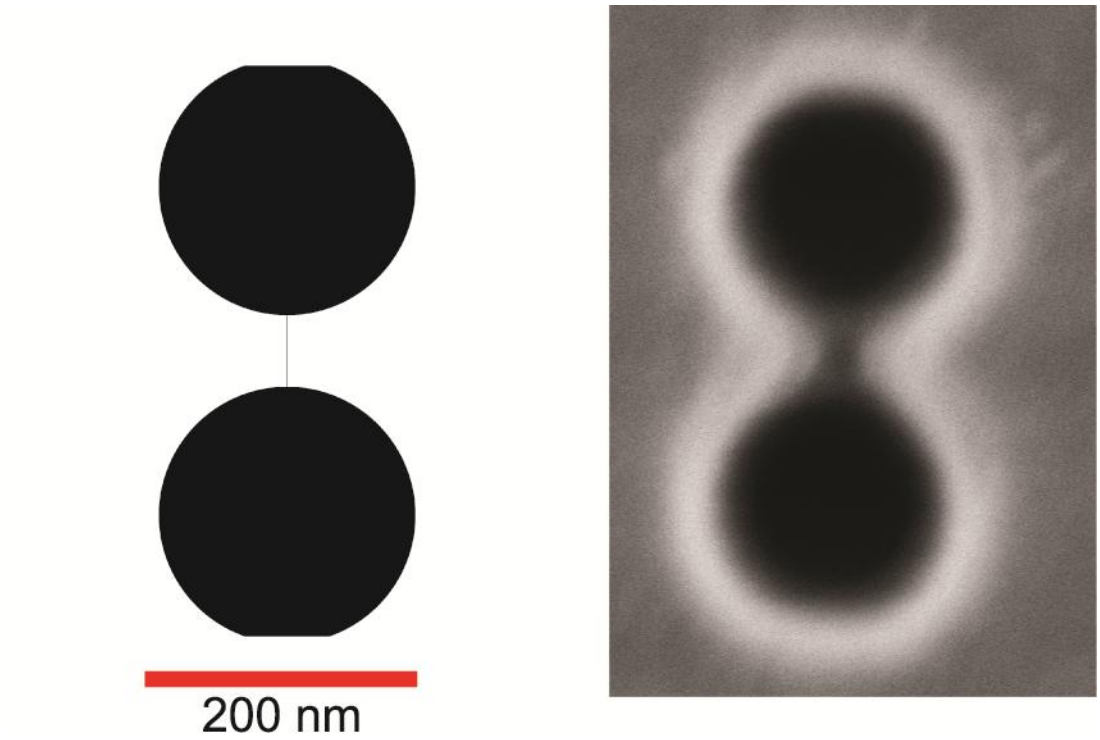


**Figure 4 | DNH pattern and fibbed sample through an electron microscope at the same scale.**

The bitmap seen in Figure 4 was milled into a 100 nm gold slide at a magnification of 80k with an accelerating voltage of 40 kV, and a beam limiting aperture of 5 µm in diameter. The number of passes used varied based on needs, but was generally in the range of 30 to 35 passes.

The structure of the bitmap is two solid spheres with a single pixel line in between them. Ideally the two circles would be just nearly touching, however to account for the finite beam width in the FIB they are separated by 25 nm, the same logic applies to the cut line. Although it is 1 pixel in width the resulting separation between the tips of the DNH will be that of the beams FWHM, approximately 15 – 20 nm.

## 3.2 Double Nanohole Optical Trapping Setup

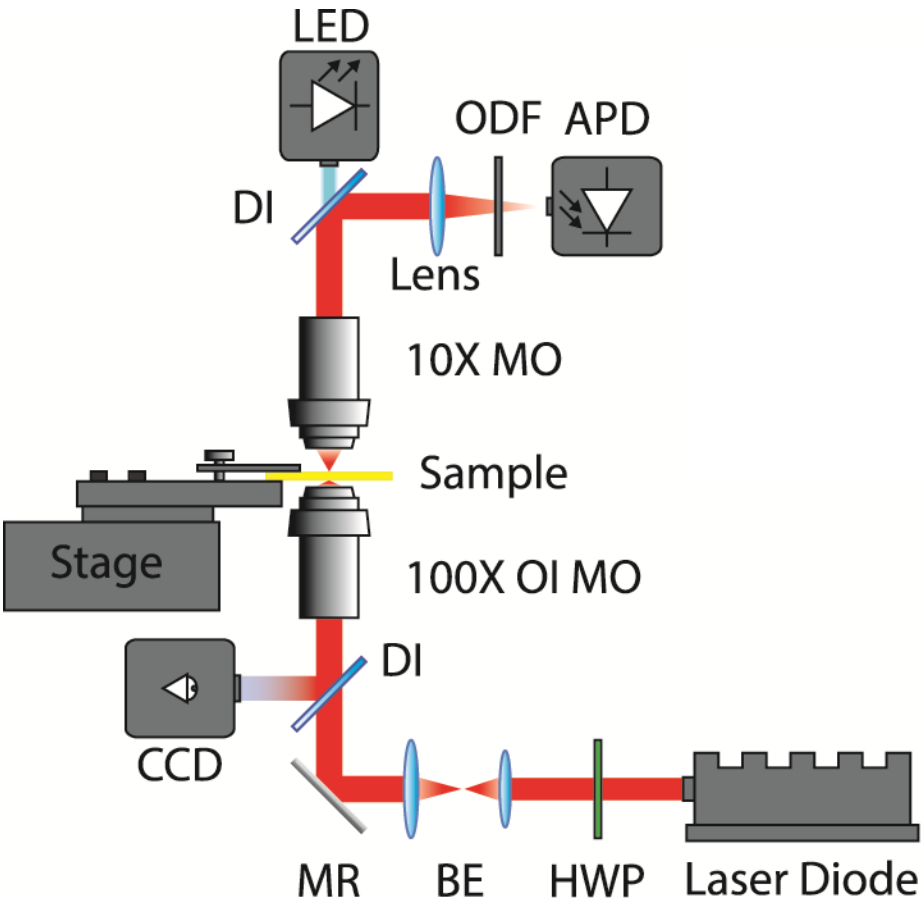


**Figure 5 | DNH Trapping Setup.** Figure 5 shows a schematic of the DNH trapping setup containing: half wave plate (HWP); beam expander (BE); mirror (MR); dichroic reflector (DI); CCD camera (CCD); optical density filter (ODF); avalanche photodiode (APD).

Figure 5 shows the basic DNH trapping setup based on the Thorlabs optical tweezer kit OTKB. This setup is the basis of all of the experiments presented in this work. There are two

variations of the DNH setup, one which makes use of a free space laser centered at 830 nm, while another uses fiber coupled lasers at 853 nm. The setup is designed to run the incoming laser through a collimator and then subsequently a beam expander to ensure proper filling of a 100x oil immersion objective (1.25 numerical aperture) that is used to focus the beam on the DNH milled into a 100 nm thick gold sample. The DNH structure is highly polarization dependent, like similar structures[11], and thus the beam is also passed through a $\lambda/2$ wave plate to align the polarization of the beam such that the electric field is aligned with the two cusps of the DNH structure. This configuration produces the highest local field enhancement between the two cusps. Once the laser is incident on the gold sample a piezo triple axis control stage is used to adjust the focus and the position of the spot on the sample. Once the DNH is found the spot is centered and focused on the aperture until the transmission is maximized.

The transmission signal through the aperture is collected by a 10x condenser (objective) which then collimates the light up into a 45 degree dichroic mirror which redirects that signal onto a path incident on an avalanche photodiode (APD) which is used to record the relatively weak transmission. To focus on the small active region of the APD (~1 $mm^2$) a small adjustable convex lens is used on a cage mount so its focus point can be shifted to fit the needs of the user. The APD will output a voltage signal. This voltage signal is read at various rates (depending on the needs of the experiment) usually in the range of 10 kHz to 1MHz.

## 3.3 Double Nanohole Sample Preparation

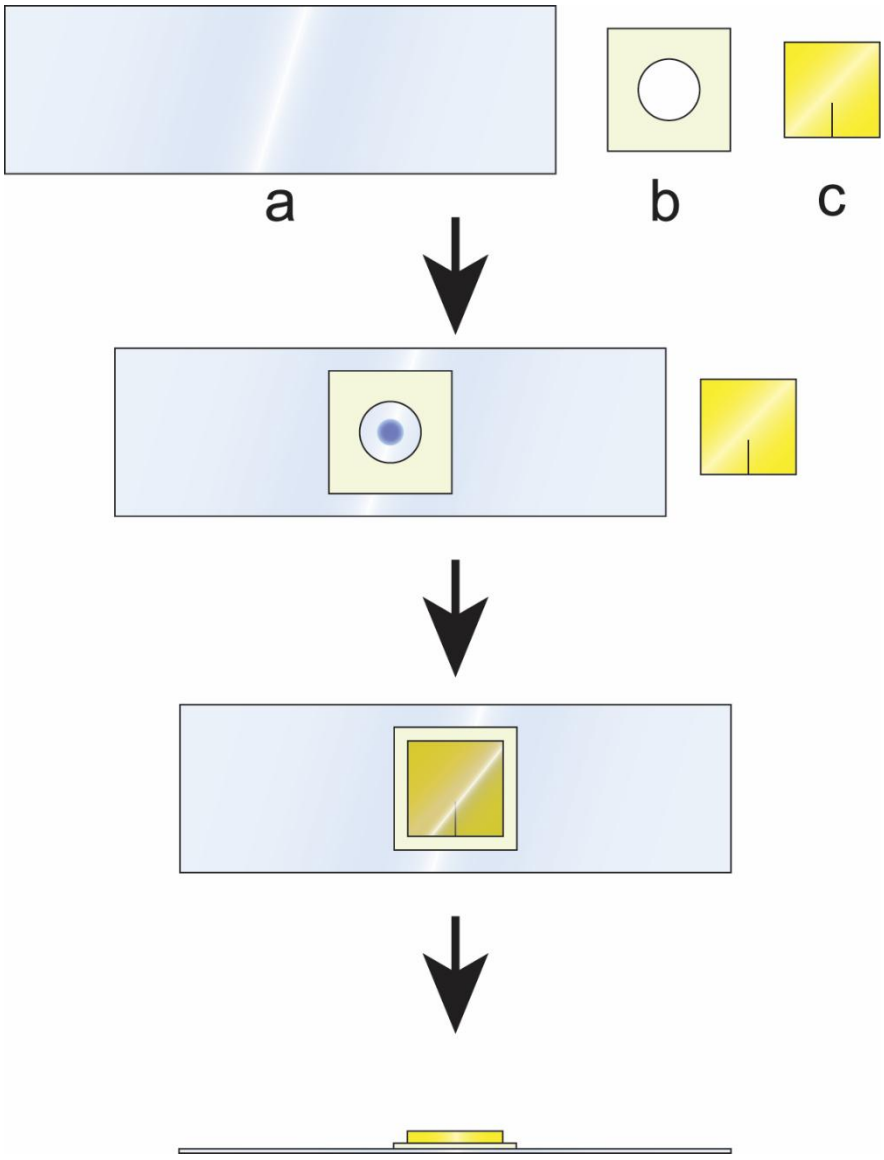


**Figure 6 | Diagram of sample preparation.** a) Glass side, b) double sided adhesive spacer, c) gold sample with DNHs.

Samples are prepared for trapping in a three step process. A number zero glass slide is removed from its packaging and cleaned using ethanol to reduce the chances of contamination of

the solution which will contain the trapping target. Once the glass slide is cleaned a double sided adhesive microscope spacer (grace bio labs) has one side of its adhesive exposed and is affixed to the center of the glass slide. Once there is a reasonable seal between the spacer and the glass slide a micro pipette is used to place a 10 microliter spot of a solution containing the desired target with various different concentrations (commonly a concentration of .1 percent is used) at the center of the circular recession. The gold sample containing the DNHs is now lightly from end to end pressed down on the top side of the exposed adhesive on the microscope slide to make a small cavity containing the fluid.

Now that the sample is assembled a small amount of microscope immersion oil is placed on the lower side of the slide. The sample is now installed into the setup by clamping it down on a small slide holder that is affixed to the piezo control stage. From here the DNHs are found using the imaging CCD attached to the setup, and the laser is aligned on the desired DNH. The transmission is maximized by changing the position gently and adjusting the polarization. Once the transmission is maximized the setup can be left alone until trapping is achieved.

## 3.4 Molecular Weight Characterization of Single Globular Proteins using Optical Nanotweezers

**Abstract:**

We trap a set of molecular weight standard globular proteins using a double nanohole optical trap. The root mean squared variation of the trapping laser transmission intensity gives a linear dependence with the molecular weight, showing the potential for analysis of globular proteins. The characteristic time of the autocorrelation of the trapping laser intensity variations

scales with a -2/3 power dependence with the volume of the particle. A hydrodynamic laser tweezer model is used to explain these dependencies. Since this is a single particle technique that operates in solution and can be used to isolate an individual particle, we believe that it provides an interesting alternative to existing analysis methods and shows promise to expand the capabilities of protein related studies to the single particle level.

## 3.4.1 Introduction

The three most cited journal papers are on techniques for protein analysis[12-14]. It is clearly of great interest to the scientific community to have ways to study these basic elements of life. Here we present a method that can be used to obtain information about single proteins that are gently trapped in an optical tweezer. We have shown previously the capabilities of our double nanohole (DNH) optical nanotweezers to trap single proteins[15], observe folding and unfolding of trapped proteins, probe protein small molecule interactions[16, 17]and to probe their protein-DNA interactions[18].

Moving forward, the goal is to use the DNH tweezer approach to not only probe interactions, but to probe the physical properties of trapped targets directly. Given the importance of understanding proteins and their interactions there already exist several methods that attempt to extract this information like mass spectrosocopy[19], amino acid analysis[20] gel electrophoresis[21], and gas chromatography[22]. The approach we present here is unique in that it is inherently a single protein technique. The proteins are contained in liquid solution. We do not require tagging or agents to be introduced to the system. Lastly, this approach does not induce nor does it require denaturing of the protein allowing us to identify a protein and then probe

subsequent interactions with the protein after it has been identified, giving the option of working in heterogeneous mixtures dynamically.

In this work we target globular proteins because they are approximately spheroidal[23]. Membrane proteins are not considered because we have not developed the approach to trap such targets within a membrane. We also trapped collagen, a fibrous protein, but the events were too brief to perform the required analysis. For each protein that was trapped two measures of the transmission signal were used for sizing, the root mean square value of the transmission signal and the autocorrelation of the transmission. The root mean square value of the transmission was calculated by scanning the raw data with a moving window of 100 points, in each window the standard deviation of the points is calculated and stored. After the entire 2 minutes of transmission data has been processed the mean of the standard deviations is calculated and used as a measure of the signal fluctuations in the raw data. The autocorrelation is found by calculating the cross-correlation of the data against itself at different points in time by shifting the data by a set number of point in integer steps, this offset is also denoted as the number of lags. For each lag the cross correlation of the data is calculated and stored. The final result, also known as the autocorrelation function, is visualized by plotting the cross-correlation at each point with respect to the number of lags present between the input data and the lagged data for that point.

For each of the two measurements made on each protein, the results are linked back to the properties and dynamics of the particles within the trap by considering a Langevin analysis of our laser tweezer system[24]. Also known as a power spectrum analysis, this technique analyzes the laser tweezer system as a simple spherical particle, subject to Brownian motion, contained

within a harmonic trapping potential. The motion of a particle in this system would be described by the following equation[24]:

$$m\ddot{x}(t) + \gamma\dot{x}(t) + \kappa x(t) = (2k_bT\gamma_0)^{\frac{1}{2}}\,\zeta(t)$$

where $x(t)$ is the trajectory of the Brownian particle, *m* is the particle's mass, $\gamma$ is the Stokes drag on the particle, $-\kappa x(t)$ is the restoring force of the potential, and $(2k_bT\gamma_0)^{\frac{1}{2}}\,\zeta(t)$ accounts for a random Gaussian process meant to simulate Brownian forces on the particle at a set temperature *T*. By solving for $x(t)$ and Fourier transforming, two key values that define the dynamics of the laser tweezer, the corner frequency $f_c$ and the diffusion constant *D*, can be found by fitting lorentzians to the power spectrum of $x(t)$. These values are used to calculate the stiffness of the laser tweezer trap $\kappa$ which can then be used in conjunction with the known stokes drag $\gamma$ to get a measure of the autocorrelation time constant. The Stokes drag itself is calculated as $\gamma = 6\pi\eta r$ where $\eta$ is the surrounding fluid's shear viscosity and r is the particle's radius, this value describes the friction force experienced by the particle as it attempts to move through a fluid. Lastly a correction to the Stokes drag is presented using Faxén's law to account for the close proximity between the particle and the walls of the trapping aperture[25]. This correction, which is heavily based on the position of the particle within the trap presents a correction to the Stokes drag of the form:

$$\gamma = \frac{6\pi\eta r}{1 - \frac{9}{16}\left(\frac{r}{h}\right) + \frac{1}{8}\left(\frac{r}{h}\right)^3 - \frac{45}{256}\left(\frac{r}{h}\right)^4 - \frac{1}{16}\left(\frac{r}{h}\right)^5}$$

where r is the radius of the particle and h is the distance from the particle to the wall of the trapping site.

## 3.4.2 Methods

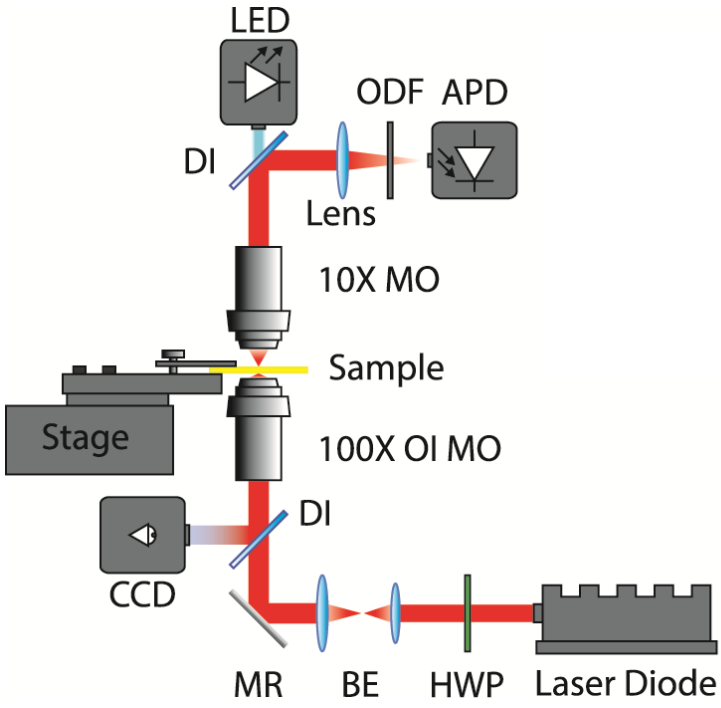


**Figure 7 | DNH Trapping Setup.** Figure 7 shows a schematic of the DNH trapping setup containing: half wave plate (HWP); beam expander (BE); mirror (MR); dichroic reflector (DI); CCD camera (CCD); optical density filter (ODF); avalanche photodiode (APD)

| Protein Name | $M_r$ (KDa) | Volume ($nm^3$) | Radius (nm) |
|---|---|---|---|
| **Aprotinin** | 6.5 | 7.88 | 1.234 |
| **Ribonuclease** | 13.7 | 16.6 | 1.582 |
| **Trypsin** | 23.3 | 28.3 | 1.733 |
| **Carbonic Anhydrase** | 29 | 35.1 | 2.031 |
| **Ovalbumin** | 43 | 52.1 | 2.317 |

| | | | |
|---|---|---|---|
| **Conalbumin** | 75 | 90.9 | 2.789 |

**Table 1 |** Calculated protein properties from molecular weight assuming a specific volume of 0.73 $cm^3/g$

To study the different behavior of the proteins in the trap, samples were prepared separately for each of the five proteins from the GE LMW filtration calibration kit (GE28-4038-41, Sigma Aldrich), including aprotinin, ribonuclease a, carbonic anhydrase, ovalbumin, and conalbumin along with two additional samples prepared with trypsin, and a complex of trypsin and ovalbumin (properties of each protein are listed in Table 1). The DNH traps are made by focused ion beam milling into 100 nm thick gold slides (EMF corp). These slides are adhered to a glass slide (EMS 032414-9) containing a solution of our desired protein diluted in phosphate buffer solution in a nano-well made by an adhesive microscope spacer (Grace Bio-labs GBL654002). The completed sample is mounted onto our setup, a schematic of the trapping setup can be seen in Figure 7. The complete sample preparation, setup and alignment process is outlined in our previous publications[15-18].

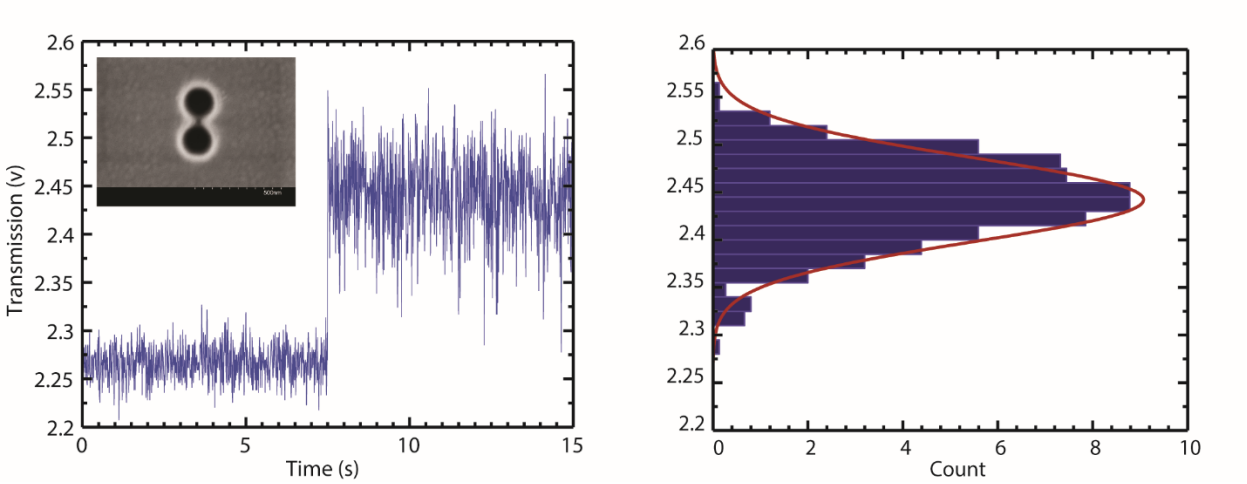


**Figure 8 | Trapping event with RMS measurement of trapped signal.** Figure 8 shows a trapping event of carbonic anhydrase. The histogram is of the last 5 seconds of trapped signal shown. Inset shows a DNH milled into 100 nm of gold.

Solutions were prepared for each protein with a concentration of 0.1 percent weight per volume. Each was trapped on the same DNH with the same trapping intensity of 5 mW/μm$^2$. Time to trap was dependent on the concentration of the target in the fluid medium, but typically around 5 minutes. Trapping was observed as a sudden jump in transmission (Figure 8) due to the dielectric loading of the particle upon entering the trap. These jumps are typically 15 – 25 percent of the untrapped signal for a 10 nm radius dielectric particle like polystyrene[10], and 5 – 15 percent for a typical protein[15]. Once trapping was achieved, 180 seconds of transmission data was collected at a rate of 1 MHz. Each measurement was repeated multiple times on the same DNH and on a different DNH to ensure reproducibility. Figure 8 shows a trapping event for carbonic anhydrase along with a histogram of the trapped signal. The full width at half max (FWHM) of this Gaussian distribution also gives the root mean square (RMS) variation of the signal.

## 3.4.3 Results and Discussion

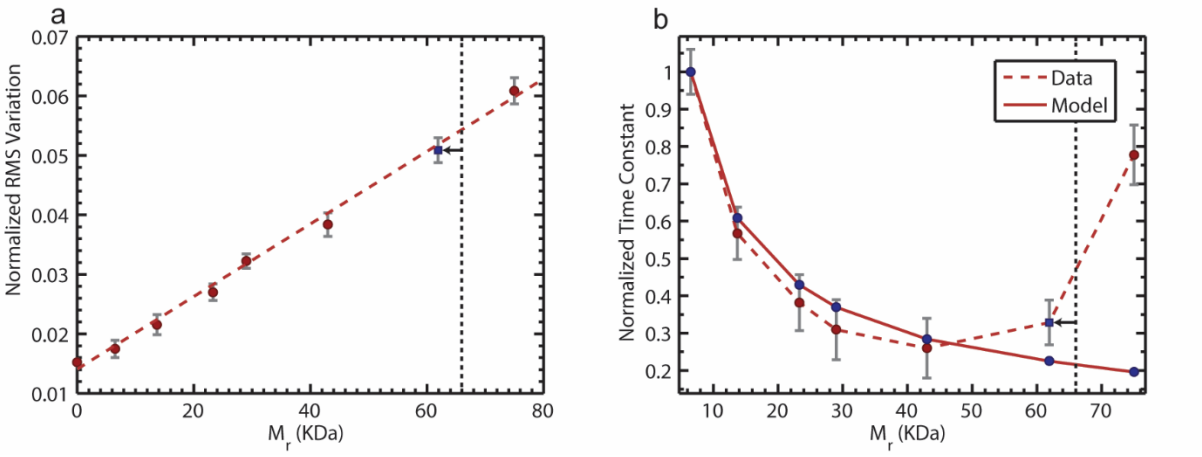


**Figure 9 | RMS Variation of Trapped Particles, Autocorrelation of Trapped Particles.** a) RMS of the trapped particles with respect to their molecular weight, the first point is of an empty trap surrounded by water. b) Autocorrelation relaxation time for the trapped particles with respect to their molecular weight. Conalbumin does not follow a -2/3 power dependence, as described in the text. Complex of trypsin and ovalbumin shown as blue square data point.

Figure 9 shows the data collected from the trapping signal of each trapped protein. Figure 9a shows the dependence on the RMS fluctuations of the signal with respect to the molecular weight of the protein that is trapped. From this data we see a linear dependence on the RMS of the trapping signal with the molecular weight of the particle. The RMS was measured multiple times for each particle showing a mean standard deviation of .0018. This scaling between the molecular weight, which can be associated with the approximate size of the particle, and the

RMS in the transmission signal is an expected result based on theory which will be discussed in the following section. The data point with trypsin (23.3 kDa) complexed with ovalbumin (43 kDa) is shown as the square blue data point. It should be noted that the complex's molecular weight is given as 4.4 kDa lower than the sum of trypsin and ovalbumin, as noted by the shift shown with arrows in Fig. 3. This lower value is the result of tryptic cleaving of ovalbumin[26]. Obviously more can be done to explore this; however, the present work is aimed at establishing the technique, and the many various applications are beyond our present scope. Figure 9b shows the time constant associated with each protein's transmission signal autocorrelation function. Red circles show the data obtained from the experiment, while blue circles show the scaling based on the hydrodynamic radius to the -2/3 power. The error associated in each auto correlation measurement was assessed by comparing the change in the time constant across different windows in a single 180 second trapping event, as well as across 2 different trapping events.

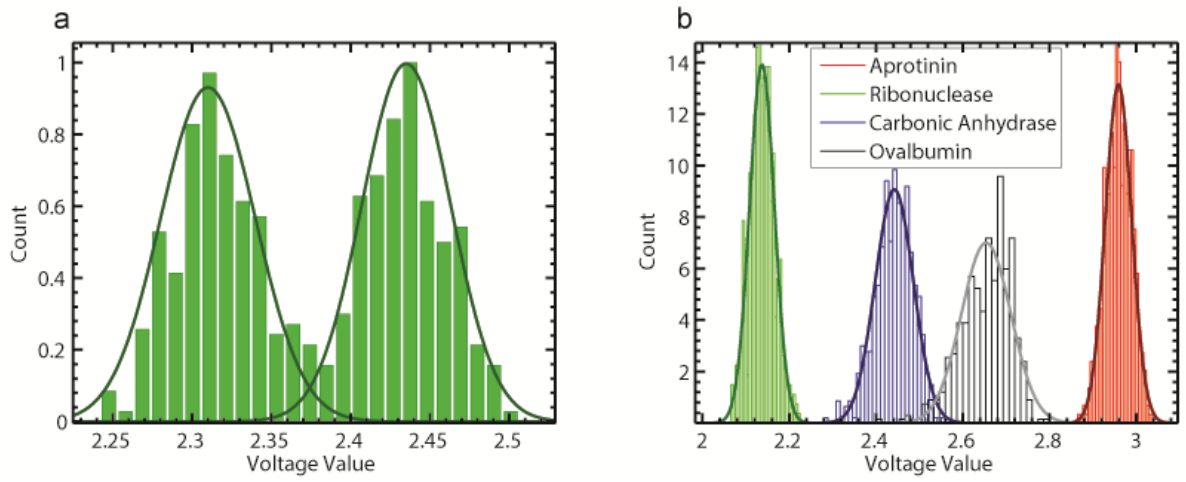


**Figure 10 | Histograms of trapping data.** a) Bimodal distribution of voltage values for conalbumin. b) Distribution of the remaining four GE LMW calibration kit proteins.

We note that the unusually large timescale of conalbumin (last point of Figure 9b) may be the result of conformational fluctuations of the protein itself influencing the autocorrelation. Evidence of these fluctuations can also be detected in the distribution of the trapping data of the protein as seen in Figure 10a. Rather than a single Gaussian peak, we see a bimodal distribution for conalbuimn. We also note that the distribution for the closely related protein ovalbumin, shown in Figure 10b, shows some hint of bimodal behavior, albeit considerably less so. We have noted similar conformal changes, as opposed to translational changes, for other proteins in the past, and used this to probe protein-small molecule interactions[26]. Conalbumin has two loosely connected sections so it is easy to see where fluctuations could arise[27]. This shows a limitation of the autocorrelation approach to determine the MW of some proteins. At the same time, it provides some indication that the protein may have large scale motion that is not translation and so there is some inherent value to this additional information. Further investigation is required to best make use of these features which are beyond the scope of this work.

## 3.4.4 Theoretical Analysis

Here we consider the potential energy of the particle in the trap and laser tweezer/hydrodynamic theory to explain the observed linear and -2/3 exponent dependencies observed in Figure 9. The potential energy of a Rayleigh particle can be written in the dipole limit as:

$$U = \frac{1}{2}\alpha|E|^2 \qquad (1)$$

where $\alpha$ is the polarizability of the particle and $E$ is the electric field. $\alpha$ scales as the particle radius to the third power, i.e. like the volume (assuming a spherical shape). It should be noted that this energy increases as the particle is trapped (like a dielectric in a capacitor increases its potential energy), yet the total energy of the system goes down (like a dielectric that is sucked into a capacitor[28]). The proteins are not perfectly spherical and so the approach is approximate; however similar formulations are used for dielectrophoresis and have been shown to work in practice[29]. The intensity is proportional to the potential energy, similarly their RMS values will also be proportional:

$$\sqrt{\langle I^2 \rangle} \propto \sqrt{\langle u^2 \rangle} \tag{2}$$

By plugging Equation 1 in place of our potential energy in Equation 2 and making note that $\alpha$ is a constant gives:

$$\sqrt{\langle I^2 \rangle} \propto \alpha \sqrt{\langle \frac{|E|^4}{4} \rangle} \tag{3}$$

Here we see from Equation 3 that the RMS of our transmission signals intensity will scale with the polarizability, which as noted earlier scales with the particle volume and hence it should show an approximately linearly dependence with molecular weight, provided the mass density is the same. This trend is seen in the data presented in Figure 9a, however it is worth noting that the data does not trend to an RMS value of zero for an empty trap since there is a base line RMS due to instrumentation noise.

Using a Langevin analysis of laser tweezers, the time constant of the autocorrelation is given by[24]:

$$\tau = \frac{\gamma}{k} \tag{4}$$

Where the Stokes' drag $\gamma$ scales as the radius of the particle, *r*, while *k* scales as the radius cubed. With this our final result for our time constant $\tau$ is $\frac{1}{r^2}$ or in terms of volume:

$$\tau \sim V^{-\frac{2}{3}} \tag{5}$$

When our protein is trapped in our nano-scale aperture surface effects come into play. Faxén's law gives a correction the Stokes drag[25]:

$$\gamma = \frac{6\pi\eta r}{1 - \frac{9}{16}\left(\frac{r}{h}\right) + \frac{1}{8}\left(\frac{r}{h}\right)^3 - \frac{45}{256}\left(\frac{r}{h}\right)^4 - \frac{1}{16}\left(\frac{r}{h}\right)^5} \tag{6}$$

Where $\eta$ is the viscosity of the medium, *r* is the radius of the particle, and *h* is the distance from the center of the protein to the wall. This can increase the drag by a factor of three close to the wall where the particle is trapped[23], so that $h \approx r$. Eq. 6 will still give a linear dependence with *r* to first order and so the analysis presented above holds so long as the variation from the surface is not too large.

To test these two models against our acquired data (fits shown in Figure 9) each of the proteins was approximated as a sphere of the smallest radius that could contain the given mass of the protein, time constant values were then calculated using the estimated protein's volume with Equation 5. These calculations were done assuming a specific volume of the proteins of

0.73 $cm^3/g$ (Ref. 30). These values for the volume and radius of the proteins are shown in Table 1.

## 3.4.5 Discussion

Upon trapping, a sudden increase is seen in the transmission through the DNH. As discussed previously this is due to the dielectric loading of the particle on the tips of the DNH structure. Additionally, in the rare event of trapping multiple targets, secondary and tertiary jumps in signal are seen. Such events are easily detected and have been excluded from this analysis.

Our approach is promising for studying specific protein interactions in heterogeneous solutions. Given the ability to gently trap and maintain trapping of a single protein for extended periods, our approach has the advantage of working in heterogeneous mixtures with no additional work needed when compared to a homogeneous mixture. We can readily trap proteins from the local population, identify their size based on their RMS and autocorrelation properties and proceed to either release them in favor of another target, or interact with them within the trap. Past works have shown that changes to proteins via small molecule interactions can drastically change the autocorrelation of the signal[16] while affecting the total size and weight of the particle only minutely. We have also shown the possibility of using the DNH trap on the end of a fiber[31]. This development is promising for multiplexing to parallel and hence speed up the analysis.

In its current state, our method does show variation in the nanofabricated traps. For DNHs that are milled using a focused ion beam, there can be variation in the shape and overall

size of the trap. For traps milled in the same run, the variation is minimal and has shown only ~1-2 percent shifts in RMS while showing almost no variation in the autocorrelation time constant. However, DNHs milled weeks or months apart will be subject to the changes in the focus ion beam alignment and can yield much greater variation. One possible solution to this is the use of a lift off process from a master template that we are presently developing[32].

We note that plasmonic interactions are often associated with heating; however, it has been shown that gold films effectively remove temperature gradients due to high thermal conduction[33]. For example, the temperature increase of a nanorod was calculated to be a 1000 times more than a similarly resonant nanoaperture[34]. We have not seen any evidence of strong thermal influence in the trapping of proteins such as strong power-dependence in the RMS fluctuations or denaturing (which would occur at temperature elevations of tens of degrees[35]).

## 3.4.6 Conclusions

In conclusion we have demonstrated a new technique that can be used to gently probe the size of single proteins in a way that is both reproducible and scalable. Our approach has the advantages of working with the proteins in a liquid environment, it is non-destructive to the protein and it does not require denaturing of the protein. We have demonstrated that our method can discern between different proteins based on their size using both the root mean square variation value of the aperture transmission signal and the autocorrelation of the transmission signal. We have presented models that match with our experimental results, we note that the autocorrelation analysis is not accurate for proteins that have large conformational fluctuations, such as conalbumin. Given our approach's ability to gently trap and release proteins on

command, it has the potential to streamline working with heterogeneous mixtures to isolate choice proteins from the surrounding solution or to quickly make assessments on the relative concentrations of each protein in the mixture. Additionally many protein-protein or protein-molecule binding events can result in changes in the overall size and conformation of the protein that can be inspected using our method.

## 3.5 Probing the Raman-Active Acoustic Vibrations of Nanoparticles with Extraordinary Spectral Resolution

### 3.5.1 Introduction

The interaction of light with mechanical vibrations has had diverse impact in two disparate regimes typified by sub-gigahertz cavity optomechanics of micron-sized structures[36] and super-terahertz Raman spectroscopy of molecular vibrations[37]. Between these frequencies, there is a gap where new technologies are desired. This gap is crucial to nanotechnology, since it contains the acoustic regime of nanoparticles, with valuable information about their specific size, shape and material properties. A diverse range of nanoparticles have acoustic vibrations in this frequency window, including colloids[38], quantum dots[39], proteins[40], DNA[41], and virions. To identify the properties of these nanoparticles, it is desirable to have an ultra-sensitive, low-frequency approach to Raman with high spectral resolution.

Spectral identification of nanoparticles using conventional Raman, the detection of vibrational modes in nanoparticle ensembles by monitoring shifts in photon energy after an incident laser source is inelastically scattered off of the ensemble[42], has been unfeasible due to

limited spectral resolution, which is typically at 1 $cm^{-1}$ for high-resolution systems. Furthermore, elastic scattering from the excitation laser source should be filtered out, which typically limits the minimum frequency of vibrations to above 10 $cm^{-1}$. For the most part, only nanoparticles smaller than 10 nm can have their acoustic vibrations probed with conventional Raman[43]. Yet many particles of interest, such as virions and colloidal particles, are larger than 10 nm. Also, smaller particles that are not as stiff, particularly biomaterials such as proteins and DNA oligomers, have large-scale vibration modes around 100 GHz[44], as will be shown here. It is particularly this region that we will address in this paper: particles of the order of 10 nm, with resonances in the range of 0.7 $cm^{-1}$ to 10 $cm^{-1}$; however, the technique may be extended to a broader range of particle sizes and frequencies in a straightforward way. While past Raman experiments have looked at large numbers of nanoparticles simultaneously[45]; here we observe the Raman-active modes in a solution of single nanoparticles trapped by a laser tweezer. We note that this is not the first demonstration of single particle sensitivity with Raman[46].

Ultrafast techniques represent another approach to access Raman-active acoustic vibrations, for example optical Kerr effect spectroscopy, for which the underlying physical mechanism is electrostriction[47]. These ultra-fast approaches have been applied to resonant metal nanoparticles[48, 49]. With such resonant scatterers, a strong Kerr effect response can be achieved, allowing for single nanoparticle probing[50, 51]. The application of such an approach to non-resonant nanoparticles has been lacking, which limits the field of use substantially. When applying such techniques to proteins or other large molecules, very broad peaks are observed[51] due to viscous damping and the well-known effect of vibrational energy relaxation – the

redistribution of energy among the various vibrational modes on the picosecond timescale.[52]

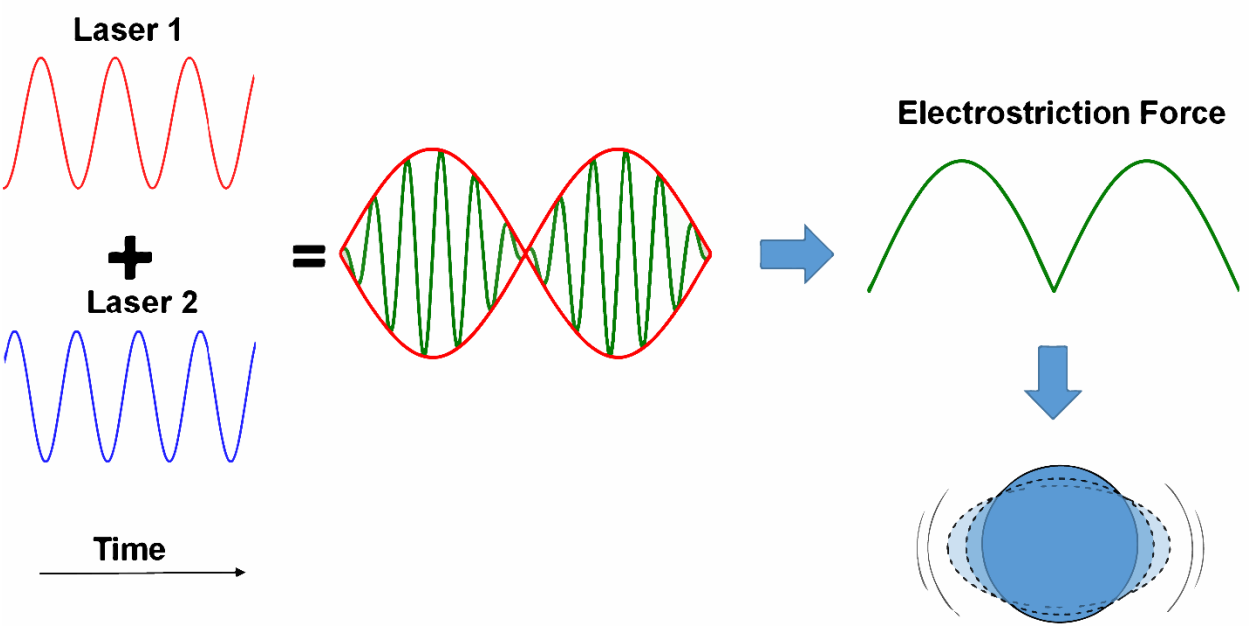


**Figure 11 | Working principle of DNH-EAR experiment.** Red and blue sources represent the two lasers. Their interaction will produce a beat signal shown in green and enveloped in red. This beat signal will modulate the electrostriction force at the trapping site causing vibrations in the particle. The solid blue circle shows the equilibrium position, dashed lines show example deformations.

## 3.5.2 Results

Here we use a different approach that has features in common with past Raman and ultra-fast techniques but is fundamentally different, allowing for extraordinary spectral resolution (especially for proteins) in the Raman-active vibrational spectra for individual nanoparticles. A simplified schematic of the approach is shown in Figure 11. The beat frequency between two lasers of slightly different wavelength is used to resonantly excite Raman-active vibration modes of the nanoparticles. Like the optical Kerr effect, this is actuated through electrostriction. Electrostriction is an effect felt by all dielectrics when they are introduced to an external electric field. Upon being introduced to an external electric field the ions in the crystal lattice that make up the material will be displaced slightly, with positive ions being pushed in the direction of the

field and negative ions being pushed against the direction of the electric field. This displacement will propagate through the entire material resulting in a net strain on the material in the direction of the electric field[53]. The beating lasers lead to a modulation in the electrostriction force at the beat frequency. When the beat frequency is tuned to an acoustic resonance, the particle is excited to oscillate resonantly. Unlike other Raman approaches, we do not use a spectrometer to measure the inelastically scattered light; rather, we measure directly the intensity fluctuations from the motion of the nanoparticle. This makes EAR naturally insensitive to elastic Rayleigh scattering, so that no filtering is required to remove the zero-loss line.

Our approach is fundamentally different from coherent ultra-fast techniques since we measure the low-frequency incoherent thermally driven motion of the nanoparticle in the laser tweezer system[28], which is well-studied for laser tweezers[54], and not the high-frequency coherently scattered light. In Ref. 28, we have shown that the typical harmonic potential with viscous drag and Langevin fluctuations[54] sufficiently explains the dynamics of the DNH trap by comparing dynamic trapping measurements with noise autocorrelation analysis. Both methods are in good agreement and confirm that the time constant is the spring constant divided by Stokes' drag. Therefore the noise fluctuations are driving by local heating, and so we do not require a strongly scattering object, like a resonant metal nanoparticle, to measure changes in local temperature. For proteins, we can measure extraordinarily sharp resonances by pumping resonantly with electrostriction, while not being susceptible to vibrational energy relaxation or have our signal washed out be viscous damping because the signal is in the heating of the object at the vibrational resonance. Considering the thermal diffusion coefficient of water ($1.43 \times 10^{-7}$ $m^2/s$), there is a thermal bottleneck of approximately a nanosecond for heat to leave the nanoparticle for a length scale of 10 nm, so that it may have elevated temperature and greater

thermal motion when exciting by electrostriction at tens of gigahertz. The intramolecular redistribution of energy among the various vibrational modes does not adversely affect our measurement since we are measuring the thermally driven motion from the particle heating up and not the coherent scattering.

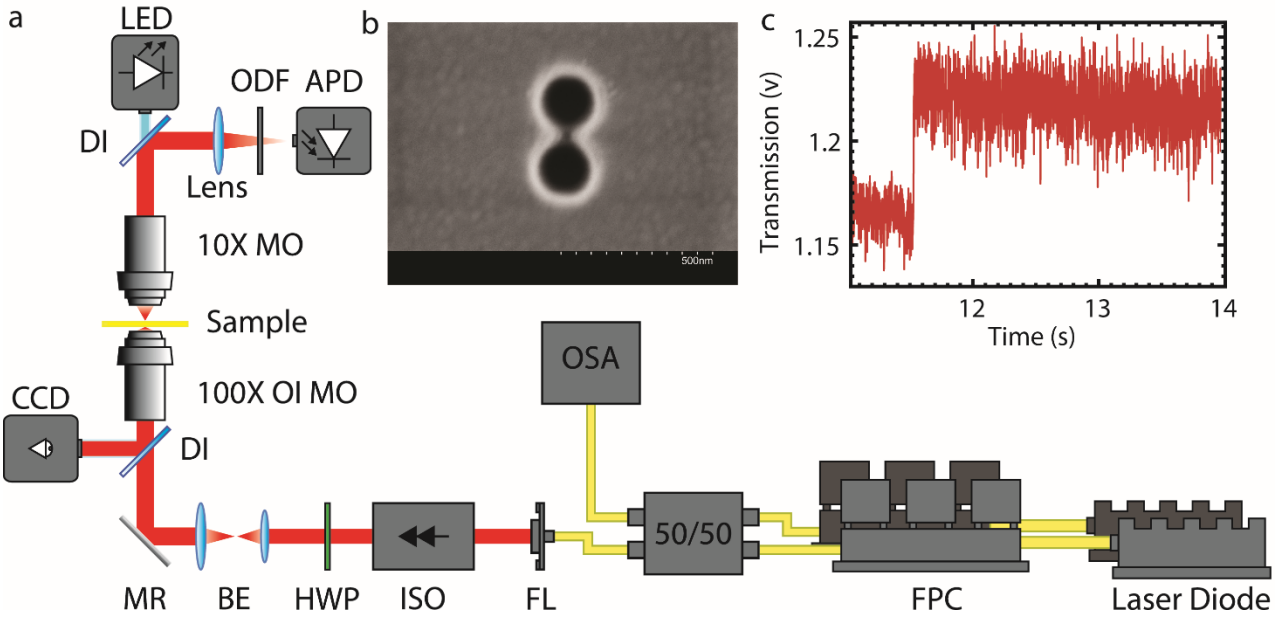


**Figure 12 | Experimental setup.** (a) Schematic of the dual-laser DNH tweezer setup. Containing: Optical spectrum analyzer (OSA), fibre coupler (50/50); fibre polarization controller (FPC); fibre launcher (FL); optical isolator (ISO); half wave plate (HWP); mirror (MR); dichroic reflector (DI); avalanche photodiode (APD); CCD camera (CCD). (b) SEM image of DNH in Au layer. (c) A sample trapping event seen as step in DNH transmission.

To isolate individual nanoparticles, we use a double-nanohole (DNH) laser tweezer, for which we have demonstrated the ability to trap, manipulate and interact individual proteins in past works[28, 55-58]. A schematic is shown in Fig. 12(a). This is an inverted microscope laser tweezer system that uses the DNH aperture as the trapping site[57]. The DNH in a metal film, shown in Fig. 12(b), is required to localize the laser field in a small gap where the two nanoholes overlap, and thereby enable the trapping of nanoparticles in the single digit nanometer range[10]. Changes in transmission through the DNH are measured as voltage changes on the avalanche

photodiode (APD) and these show the trapping of nanoparticles, as well as their motion once trapped. These changes are actuated by the nanoparticle trapped in the DNH, since the traps are sensitive to dielectric loading[10]. A trapping event is observed as a discrete step in the transmission through the DNH, as shown in Fig. 12(c). After this initial step from trapping, the transmission through the DNH shows increased fluctuations, due to the motion of the particle in the trap. Variations in these fluctuations are the signal that is detected in EAR. The typical step in the DNH transmission after the trapping of a 20 nm polystyrene nanoparticle is 10%-20%, whereas the step for trapping proteins is typically 2%-7%. This is expected since the proteins are smaller in size and their polarizability is smaller, but this is partially balanced by viscous damping, as will be discussed below. Since the EAR signal is given by the motion of the particle in the trap, it depends not only on the polarizability of the particle, but also on viscous damping of the acoustic vibrations, thermal diffusion and the electromagnetic properties of the DNH.

Unlike past works that used resonant metal nanoparticles to excite vibrations through heating, we use a nanoaperture that has 3-4 orders of magnitude less heating due to the thermal conduction of the surrounding film[58]. Heating of a nanoparticle in an isotropic medium will only excite the $l$=0 acoustic mode of a sphere, unless the symmetry is broken by a boundary[49]. Stimulated Raman, for which the underlying physical mechanism is electrostriction, can resonantly excite Raman active vibrations in a nanoparticle.

The main innovation of this work with respect to our past laser tweezer experiments is the use of two trapping lasers instead of one. By temperature tuning, we can obtain a wide range of beat frequencies between 10 GHz and 0.3 THz. It is easy to extend the range to >10 THz by many approaches, such as replacing one of the lasers with a widely-tunable external cavity laser. In principle, it is possible to measure the Raman scattered light from this oscillation, but this

introduces issues associated with filtering out the zero loss line and achieving sufficient coherent signal. Instead, we note that the particle motion increases during resonant excitation and we detect this increased motion. Experimentally, there is an increase in the light intensity fluctuations measured on the APD when the beat frequency matches a vibrational resonance of the nanoparticle.

***Raman Spectra of Dielectric Nanoparticles***

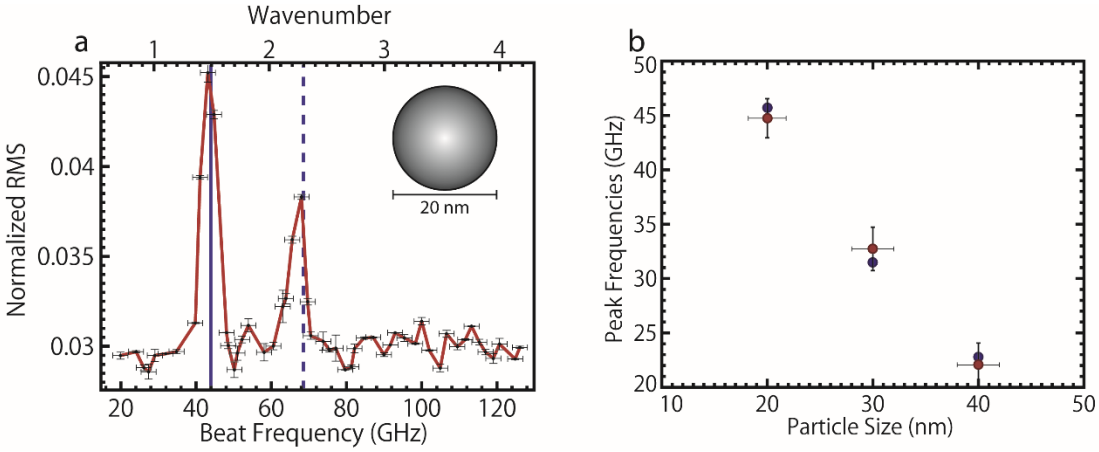


**Figure 13 | Raman spectrum of a 20 nm polystyrene nano sphere and experimentally confirmed resonant peaks.** (a) Root mean squared (RMS) fluctuation in the scattering signal for a 20 nm diameter polystyrene sphere plotted against the beat frequency of the trapping lasers. The solid blue line shows the expected frequency peak at 44 GHz for the $l$ = 2 peak. The dashed blue line shows the expected peak for the $l$ = 0 mode. (b) The experimentally confirmed position of the $l$ = 2 peaks is plotted against their theoretical values for 3 different sizes.

Figure 13(a) shows the EAR spectrum: the root mean squared (RMS) variation in the transmitted intensity through the DNH as a function of the beat frequency between the lasers for a polystyrene particle of 20 nm diameter. Peaks in the RMS are seen for beat frequencies of 44 GHz and 68 GHz, corresponding to 1.47 $cm^{-1}$ and 2.28 $cm^{-1}$. Using values for the sound velocity in polystyrene, Lamb's theory predicts the peak at 44 GHz[59], which is shown as a vertical solid blue line in the Figure. This peak is the quadrupolar accordion mode ($l$=2, $n$=0), the second peak is attributed to the $l$ = 0, $n$ = 0 spherical breathing mode (dashed blue line shows expected value).

The ratio between the peak positions between these $l = 0$ and $l = 2$ mode is 1.56, similar to past experiments and theory[60-62]. Figure 13(b) shows the experimentally confirmed $l = 2$ peak frequencies with respect to the sizes of different polystyrene nanospheres in the trap. The error bars on the particle size correspond to the manufacturer's specifications.

While the physical mechanism of electrostriction is consistent with the observed inversion-symmetry modes ($l$=0 and $l$=2), and the increased Brownian motion is consistent with heating, many features of the mechanisms underlying this technique are not entirely clear. For example, we note that the $l$=0 acoustic mode increases in RMS amplitude with increase in particle size as approximately the square of the radius, while the $l$=2 mode shows almost no size variation in the amplitude of the RMS. A cubed dependence is expected in the driving force of electrostriction[63]. Viscous damping is monotonic for the $l$=0 mode; however, it is much more complicated for the $l$=2 mode in the nanometric regime, especially for softer materials like polystyrene and PMMA[64]. We do not believe translational Stokes' drag plays a role in the size dependence because it does not influence the amplitude of particle motion in a laser tweezer; as shown from the equipartition theorem[52]. To understand these features, further theoretical investigation of the underlying physical mechanisms is required.

We did attempt to test the sensitivity to viscosity by adding 10% glycerol to the solution, increasing the viscosity by 30%. We found that the $l$=0 mode did not change significantly in amplitude (8% decrease), but the $l$=2 mode showed a substantial reduction (31%). We note that also the thermal conductivity and Claussius-Mossitti both also change by ~4% when adding this amount of glycerol. This will change the heating and light scattering properties to reduce the baseline RMS; however, there should be no distinction between the $l$=0 and $l$=2 modes with

these additional factors. The viscous damping in this regime for the *l*=2 mode displays non-monotonic behavior and so we do not suggest a simple scaling dependence.

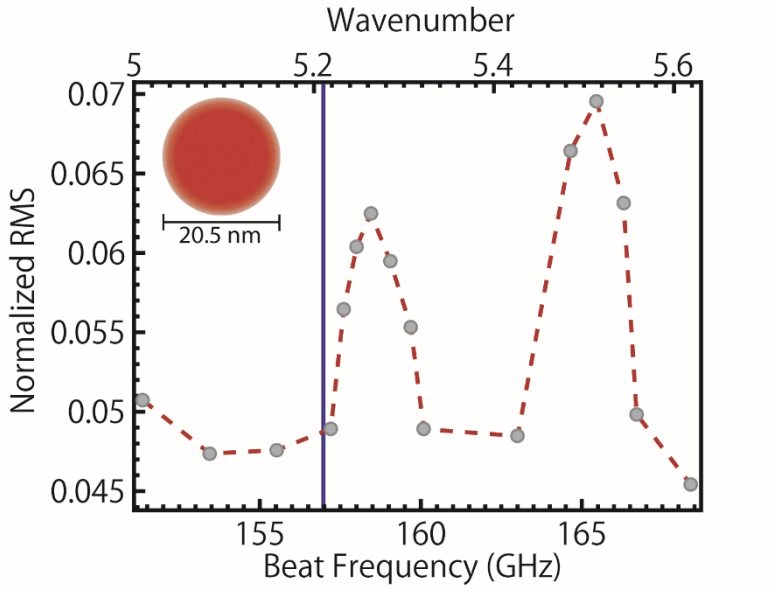


**Figure 14 | Short range Raman spectrum of a 20.5 nm titania nanosphere.** These two peaks show fine splitting of the $l$ = 2 mode due to the anisotropic nature of the particle.

To probe material anisotropy, we performed similar measurements on titanium dioxide nanospheres of diameter 20.5 nm. Figure 14 shows a narrow range of the spectrum for the region around the calculated $l$ = 2 peak, but at a higher beat frequency, which is due to the material properties of titanium dioxide. (Note error bars are omitted from this and subsequent graphs in the main text, but they are quantitatively similar to those shown in Fig. 3a). It is clear from this measurement that we can resolve a splitting in the Raman-active acoustic peak, which we attribute to the anisotropy of these nanospheres in the anatase phase[66, 67]. The difference in the $l$ = 2 resonance predicted from Lamb's theory for the sound velocities along the different orientations of anatase titanium dioxide is 7.0 GHz peak; whereas the observed splitting was 7.2 GHz. The high resolution of EAR, that was not previously accessible to Raman, is critical for

the measurement of important material properties of the nanoparticle. We note that the resolution obtained here is likely a limitation of the nanoparticles studied since the lasers we use have a 2 MHz specified linewidth.

***Raman Spectra of Proteins***

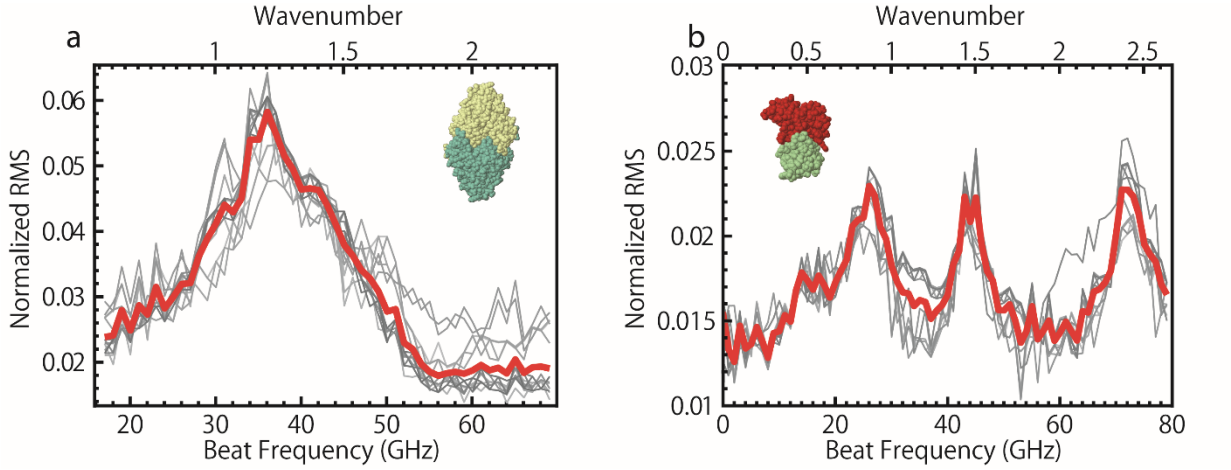


**Figure 15 | Raman spectra of two globular proteins.** (a) 22 different sweeps across 11 trapping events of carbonic anhydrase showing a singular broad peak centered around 38 GHz. (b) 20 different sweeps across 10 trapping events of conalbumin showing 2 distinct peaks and a single finely split peak. Red curves show the average of all sweeps

Example spectra for two proteins are shown in Figure 15. These proteins show distinct spectra. Carbonic anhydrase (Fig. 15a) is torus shaped with a hollow center and its spectrum shows a broad structure with a central peak at 38 GHz. Conalbumin (Fig. 15b) is a larger globular protein with two coupled lobes and it shows three sharper peaks in the scanning range. As represented in Fig. 5, the spectra for these proteins have each been reproduced 20 or more times, on 10 or more distinct trapping events, on 3 or more different DNHs, showing the same peaks.

We do not observe denaturing of the protein, which would change the transmission through the aperture substantially. This suggests that the heating of the protein is only moderate

and is consistent with our past works on proteins[57]. Double-step time series indicating the trapping of multiple proteins in a single DNH were excluded from this study.

### 3.5.3 Discussion

All of the above measurements were taken at the single particle level. This is remarkable when compared with past Raman measurements that typically probed thousands of nanoparticles[36, 37, 41, 44, 45] It also opens exciting possibilities of measuring nanoparticles prior to manipulation. Others have already shown that aperture tweezers may be used to translate nanoparticles[40]. Combined with this approach to Raman, we have shown that it is possible to measure the specific properties of the trapped nanoparticle prior to manipulation to ensure that it has the desired characteristics. Two example applications where this would be helpful: 1) we could identify and isolate a single quantum dot with the desired size, shape and material properties before placing it in a nano-device, or 2) we could identify a particular virion from a virus population before translating it to another location where infection properties can be studied.

In conclusion, we have demonstrated EAR, an approach to probing the Raman-active acoustic modes of individual nanoparticles with extraordinary spectral resolution, almost three orders of magnitude finer than conventional Raman. This approach is well suited for the 1 $cm^{-1}$ to 10 $cm^{-1}$ range that is challenging for conventional Raman, but particularly relevant to the study of nanoparticles. Due to the unprecedented resolution, this approach offers unique capabilities in measuring the size, shape and material properties of nanoparticles, including anisotropy. We anticipate broader adoption of the method, with applications in a wide range of fields including nanotechnology (e.g., to select and place quantum dots in a nanocircuit[68]),

biology (e.g., to probe individual virions in a heterogeneous population[69]), and biochemistry (e.g., to probe deformation and mechanical changes within allostery[70]).

### 3.5.4 Methods

The sample is prepared by sandwiching the desired solution between a glass slide (EMS 032414-9), an adhesive slide spacer (Grace Bio-labs GBL654002) and a 100 nm gold slide with DNH structures milled into it using a focused ion beam mill. The backside of the sample is covered with a small amount of immersion oil to work with the oil immersion lens that focuses the laser sources on the DNHs.

The sample is put in place and the fiber polarization controller is used to minimize the loss through the isolator. The beam is maximized in power and is aligned onto a DNH by using a small CCD camera to visually see the spot and the structures. The spot is fully aligned with the DNH and the polarization of the beam is controlled using the half wave plate. The polarization is varied until the signal through the DNH is maximized. Trapping is seen as a considerable jump in the APD signal, usually 20 to 30 percent of the untrapped signal. The transmission signal in our setup is large and a regular photo detector could be used while in its current configuration we need to insert an optical density filter to avoid saturation of the APD.

While the target is trapped the difference in the two sources' frequency is important, so this is monitored with a fiber-coupled optical spectrum analyzer (OSA). The temperature of one of the DFB lasers is varied to change the wavelength (from the nominal operating value of 853 nm for both DFBs). Each laser has a change of 0.06 nm per degree Celsius. After the changes are made the new wavelength difference is found and a 10 – 30 second window of data is recorded from the APD for that wavelength difference value.

The tuning is repeated until the desired range is covered. For each wavelength difference the data associated with that value is run through a windowed root mean square filter. From the output of this filter the mean RMS is collected and plotted against its wavelength difference value, which is converted into the beat frequency.

### 3.5.5 Conclusions

In this experiment and the following publication we designed and executed a new experimental process to access the low frequency acoustic vibrational modes of nanoparticles. This information has otherwise been unobtainable with the exception of brillouin scatter which has one magnitude less resolution than EAR. This research has also been applied to studying the structure and potentially sequencing single stranded DNA and the internal structure of small viruses in with diameters in the 30 nm range. This new experiment has shown promise to have a substantial use in the bio-science and drug discovery.

# Chapter 4

# Conclusion and Future Works

The main achievement of this work was finding new and innovative ways to extract information about optically trapped particles from the limited data available. I have shown through my work that the molecular weight of trapped particles can be probed by examining the fluctuations they shown in the transmission signal of the optical trap. Additionally this approach shows promise for examining small changes in trapped particles when new stimuli are introduced. For example, complexing two proteins and examining their change in fluctuation and autocorrelation in real time. To further probe and collect more information from trapped targets the DNH-EAR approach was conceived. By using two tunable trapping lasers as opposed to one, low wavenumber acoustic modes of particles were shown to be excited by the modulated electrostriction force within the trapping region. The spectra produced by these experiments were shown to be unique across different types of particles showing the power of this approach to be used in single particle identification.

These two advancements of the DNH optical trapping show extreme promise to allow the DNH tweezers to be used in several different high impact areas, most notably in drug discovery. However, this is in the long term. On a more short term scale there is a variety of different directions to take this technique that can both help different fields of study and advance the field of optical trapping and manipulation directly. In the following section I will outline some of these future developments.

## 4.1 DNA Sequencing

By pushing the DNH-EAR setup into the near to mid terahertz regime we can begin to excite intramolecular modes as opposed to lower frequency acoustic modes. With this in mind, it is possible to use the DNH-EAR setup to sequence and identify different strands of DNA completely in solution.

## 4.2 Fiber Multiplexing

By using a fiber with a DNH specially fabricated on the end the DNH tweezer system can effective be reduced down significantly in both size and cost. By combining several of these fibers into a plate it would be possible to have ten to a hundred DNH traps running on a solution at one time. This gives the possibility of highly efficient concentration analysis in fluid, and is also of special interest for the development of the DNH system as a drug discovery technique.

# Bibliography


1. A. Ashkin, J. Dziedzic, J. Bjorkholm, S. Chu, Opt. Lett., 1986, 11, 288-290.
2. Y. Pang, R. Gordon, Nano Lett., 2011, 11(9), 3763–3767.
3. M. Juan, M. Righini, R. Quidant, Nature Phot., 2011, 5, 349-356.
4. S. Wheaton, R. M. Gelfand, R. Gordon, Nature Phot., 2015, 9, 68-72.
5. S. Wheaton, R. Gordon, Analyst, 2015.
6. A. A. Al Balushi, et al., Analyst, 2015.
7. A. Kotnala, R. Gordon, Nano Lett., 2014, 14(2), 853-856.
8. J. Weiner, RPP., 2009, 6(72).
9. J. D. Jackson, "Classical Electrodynamics", Wiley, New York, 1998, pg 130–133.
10. M. L. Juan, R. Gordon, Y. Pang, F. Eftekhari, R. Quidant, Nature Phys., 2009, 5, 915-919
11. J. Berthelot, S. S. Aćimović, M. L. Juan, M. P. Kreuzer, J. Renger, R. Quidant, Nature Nanotech., 2014, 9, 295-299
12. L. Farr, R. Randall, O. Lowry, N. Rosebrough, A.J. Biol. Chem., 1951, 193, 265-275
13. U.K. Laemmli, Nature, 1970, 227, 680-685.
14. M. Bradford, Ana. Biochem., 1976, 72(1), 248-254.
15. Y. Pang, R. Gordon, Nano Letters, 2012, 12(1), 402-406.
16. A. Al Balushi, A. Zehtabi-Oskuie, R. Gordon, Biomedical Optics Express, 2013, 4(9), 1504-1511.
17. A. Al Balushi, R. Gordon, ACS Photonics, 2014, 1(5), 389-393.
18. A. Kotnala, R. Gordon, Biomedical Optics Express, 2014, 5(6), 1886-1894.
19. A. Shevchenko, M. Wilm, O. Vorm, M. Mann, Anal. Chem., 1996, 68(5), 850-858.
20. F. Kaiser, C. Gehrke, R. Zumwalt, K. Kuo, Journal of Chromatography A, 1974, 94, 113-133.
21. H. Schägger, G. Jagow, Anal. Biochem., 1987, 166, 368-379.
22. E.S. Erwin, G.J. Marco, E.M. Emery, Journal of Dairy Sci., 1961, 44(1), 1768-1771.

23. K. Dill, Biochemistry, 1985, 24(6), 1501-1509.

24. A. Kotnala, R. Gordon, Nano Letters, 2014, 14(2), 853-856.

25. J. Happel, H. Brenner, "Low Reynolds Number Hydrodynamics"; Prentice-Hall, Inc.: Upper Saddle River, NJ, 1965.

26. D. Bonenfant et al., Proc. Natl. Acad. Sci., 2003, 100(3), 880-885.

27. A. Bezkorovainy, R. Zschocke, D. Grohlich, BBA – Protein Structure, 1969, 181(1), 295-304.

28. D. Griffiths, "Introduction to Electrodynamics 3rd edition", 1998, 194-198.

29. H. Morgan, M. Hughes, N. Green, Biophysical Journal, 1999, 77, 516-525.

30. F. M. Richards, Journal of Molecular Biology, 1974, 82(1), 1-14.

31. R. M. Gelfand, S. Wheaton, R. Gordon, Optics Letters, 2014, 39(22), 6415-6417.

32. A. Zehtabi-Oskuie, A. Zinck, R. M. Gelfand, R. Gordon, Nanotechnology, 2014, 25, 495301.

33. K. Wang, K. B. Crozier, ChemPhysChem, 2012, 13, 2639-2648.

34. P. Melentiev et al., Optics Express, 2013, 21(12), 13896-13905.

35. Charles Tanford, Advances in Protein Chemistry, 1968, 23, 121-282.

36. D. Van Thourhout, J. Roels, 2010, Nature Photon. 4, 211–217.

37. H. K. Yadav, et al., 2006, Phys. Rev. Lett. 97, 085502.

38. M. Fujii, S. Hayashi, K. Yamamoto, 1990, Appl. Phys. Lett. 57, 2692–2694.

39. G. J. Thomas, P. Murphy, 1975, Science 188, 1205–1207.

40. H. Urabe, Y. Tominaga, 1981, J. Phys. Soc. Jpn 50, 3543–3544.

41. A. Campion, P. Kambhampati, 1998, Chem. Soc. Rev. 27, 241–250.

42. E. V. Efremov, F. Ariese, C. Gooijer, 2007, Anal. Chem., 606(2), 119-134.

43. A. Nicolaï, P. Delarue, P. Senet, "Computational Methods to Study the Structure and Dynamics of Biomolecules and Biomolecular Processes Vol. 1" (ed. Liwo, A.) Ch. 15, 483–524.

44. L. Saviot, B. Champagnon, E. Duval, A. I. Ekimov, 1998, Phys. Rev. B 57, 341–346.

45. S. Nie, S. R. Emory, 1997, Science 275, 1102–1106.

46. Q. Zhong, J. Fourkas, 2008, J. Phys. Chem. B 49, 15529–15539.

47. G. V. Hartland, 2006, Annu. Rev. Phys. Chem. 57, 403–430.

48. M. Pelton, et al., 2009, Nature Nanotech. 4, 492–495.

49. M. A. Van Dijk, M. Lippitz, M. Orrit, 2005, Phys. Rev. Lett. 95, 267406.

50. K. Yu, P. Zijlstra, J. E. Sader, Q. H. Xu, M. Orrit, 2013, Nano Lett. 13, 2710–2716.

51. D. Turton, et al., 2014, Nature Commun. 5, 3999.

52. H. Fujisaki, J. E. Straub, 2005, Proc. Natl. Acad. Sci. USA 102, 6726–6731.

53. R. E. Pelrine, R. D. Kornbluh, J. P. Joseph, Sensor and Actuators A, 1998, 64, 77-85

54. K. C. Neuman, S. M. Block, 2004, Rev. Sci. Instrum. 75, 2787–2809.

55. Y. Pang, R. Gordon, 2011, Nano Lett. 11, 3763–3767.

56. A. Kotnala, D. DePaoli, R. Gordon, 2013, Lab on a Chip 13, 4142–4146.

57. Y. Pang, R. Gordon, 2011, Nano Lett. 12, 402–406.

58. P. Melentiev, A. Afanasiev, A. Kuzin, A. Baturin, V. Balykin, 2013, Opt. Express 21, 13896–13905.

59. N. N. Ovsyuk, V. N. Novikov, 1996, Phys. Rev. B 53, 3113–3118.

60. M. Ivanda, et al., 2006, J. Raman Spectrosc. 37, 161–165.

61. C. Pighini, D. Aymes, N. Millot, L. Saviot, 2007, J. Nanopart. Res. 9, 309–315.

62. T. Still, M. Mattarelli, D. Kiefer, G. Fytas, M. Montagna, 2010, J. Phys. Chem. Lett. 1, 2440–2444.

63. L. D. Landau, E. M. Lifshitz, “Electrodynamics of Continuous Media”, 55–58 (Pergamon Press, 1960).

64. D. B. Murray, L. Saviot, 2007, J. Phys. Conf. Ser. 92, 71–78.

65. Y. Ding, B. Xiao, 2014, Comput. Mater. Sci. 82, 202–218.

66. C. H. Wang, J. McHale, J. Chem. Phys. 72, 4039–4044 (1980).

67. J. Berthelot, et al., 2014, Nature Nanotech. 9, 295–299.

68. A. G. Curto, et al., 2010, Science 329, 930–933.

69. Y. Pang, H. Song, J. H. Kim, X. Hou, W. Cheng, 2014, Nature Nanotech. 9, 624–630.

70. H. N. Motlagh, J. O. Wrabl, J. Li, V. J. Hilser, 2014, Nature 508, 331–339.

# Appendix A: Matlab EAR data processing code

Reads the data files output by the code displayed in appendix C.

For each file with a specific prefix, the file is opened and the data is scanned with a preset window size. In each window the RMS fluctuations of the signal is recorded and stored in an array. Once the file is completely scanned the mean value of all of the RMS window outputs is stored into a global array. Additionally for each file loaded this way a special file containing the readout temperature of the laser diode is read in and the beat frequency between the two lasers is calculated based off of the known scaling of the laser's center wavelength with temperature. Once all of the files with a set prefix have been scanned by the script, the temperature data and RMS data are plotted against each other to produce an EAR spectrum.

```
%AutoBeatRMS.

%Parameters for windowed RMS
window = 40;
overlap = 0;
padding = 0;


filePrefix = 'fileprefix'
rmsData = [];
beatData = [];

%What file number to start and stop at
lim = 60
start = 0

for i = start:lim
    rmsData = [rmsData
meanrmsfile(strcat(filePrefix,num2str(i),'.txt'),window,overlap,padding)];
    beatData = [beatData getBeatFreqFromTemp(852.944,
strcat(filePrefix,num2str(i)))];

end


%Setup Figure
figure('Units', 'pixels', ...
    'Position', [100 100 500 375]);
hold on;
```

```
%Fancy line with dots
plot(beatData, rmsData,'--','LineWidth',2, 'Color',[.7 .2 .2] ,
'MarkerEdgeColor',[.4 .4 .4],'MarkerFaceColor',[.6 .6 .6],'MarkerSize',6)
plot(beatData, rmsData,'o','LineWidth',1, 'Color',[.7 .7 .7] ,
'MarkerEdgeColor',[.4 .4 .4],'MarkerFaceColor',[.6 .6 .6],'MarkerSize',6)


hXLabel = xlabel('Beat Frequency (GHz)');
hYLabel = ylabel('Normalized RMS');


set( gca                       , ...
    'FontName'   , 'Helvetica' );

set([hXLabel, hYLabel]  , ...
    'FontSize'   , 15          );

set(gca,'fontsize',15)


set(gca, ...
  'Box'         , 'off'     , ...
  'TickDir'     , 'out'     , ...
  'TickLength'  , [.02 .02] , ...
  'XMinorTick'  , 'on'      , ...
  'YMinorTick'  , 'on'      , ...
  'LineWidth'   , 1         );

set(gcf, 'PaperPositionMode', 'auto');
```

--Get beat frequency from temperature --

```
%GetBeatFreqFromTemp -- Takes a wavelength and a file with temperatures to
%find the beat freqeuncy between two temperature tuned lasers
function y = getBeatFreqFromTemp(firstWL, filename)
%Given the stationary wavelength of one laser, and the temperature of the
%tuned laser, find the beat freqeuncy basd ont he measured temperature
%scaling.
firstWL = 852.944;
temp = importdata(strcat(filename,'-temp.txt'), ',');
temp = temp(1);

secondWL = 0.04858*temp + 851.97;


Cnms = 2.9979e17;
w1 = firstWL;
w2 = secondWL;
freq =  abs(Cnms / w1 - Cnms / w2);
freq = freq * 1e-9;
```

```
y = freq
```

--Windowed RMS measurement --

```
function y = rms(signal, windowlength, overlap, zeropad)

delta = windowlength - overlap;

%% CALCULATE RMS

indices = 1:delta:length(signal);

if length(signal) - indices(end) + 1 < windowlength
    if zeropad
        signal(end+1:indices(end)+windowlength-1) = 0;
    else
        indices = indices(1:find(indices+windowlength-1 <= length(signal), 1,
'last'));
    end
end

y = zeros(1, length(indices));
signal = signal.^2;

index = 0;
for i = indices
    index = index+1;
    y(index) = std(signal(i:i+windowlength-1))/(mean(signal(i:i+windowlength-
1)));
end
```

# Appendix B: Automatic data logging and trapping detection code

C# Form and program that will read out a DAQ card and watch the data for events that appear to be trapping.

At each step the program records a preset number of points at a given rate from a DAQ card. After the set of data is taken the mean is calculated and plotted in real time. After each window the mean value of the window is compared to the mean value of the data currently in the plot window. If there is an increase in the mean value greater than a set threshold the program flags the point as a trapping event. Once trapping is detected an email is sent to the user and an additional minute of data is taken and saved to a text file.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;

using System.Windows.Forms;
using System.Diagnostics;
using AnalogIO;
using MccDaq;
using System.Collections;
using System.Windows.Forms.DataVisualization.Charting;

using System.Net.Mail;

namespace ULAI01
{
    public partial class DAQWATCHER : Form
    {

        MccDaq.MccBoard DaqBoard = new MccDaq.MccBoard(0);
        public Timer tmrBuffer;
        public Timer tmrConvert;
        public Timer tmrWrite;
        public Timer tmrMs;
        private int HighChan = 3, NumAIChans = 4;
        private int ADResolution;
        string[] seriesArray = { "Voltage"};
        Series series;
        bool clearBuffer = false;
        int convertCount = 0;
        double meanValue = 0;
        List<float> dataList = new List<float>();
        List<float> tempDataList = new List<float>();
        bool first = true;
        bool hasTrapped = false;
        bool firstTrapAction = true;
```

```
        MccDaq.Range Range;



        private IntPtr MemHandle = IntPtr.Zero;
        private bool HighResAD = false;
        private MccDaq.ScanOptions Options;


        static int numPoints = 5000 * 10;
        const int FirstPoint = 0;
        private ushort[] ADData = new ushort[numPoints]; // array to hold the input
values
        private uint[] ADData32 = new uint[numPoints]; //  dimension an array to hold the
high resolution input values
        public double[] tempVals = new double[numPoints];



        AnalogIO.clsAnalogIO AIOProps = new AnalogIO.clsAnalogIO();


        private void frmDataDisplay_Load(object eventSender, System.EventArgs eventArgs)
        {

           // int LowChan;
           // MccDaq.TriggerType DefaultTrig;

            InitUL();

            // determine the number of analog channels and their capabilities
           // int ChannelType = clsAnalogIO.ANALOGINPUT;
           // NumAIChans = AIOProps.FindAnalogChansOfType(DaqBoard, ChannelType,out
ADResolution, out Range, out LowChan, out DefaultTrig);
        }

        private void tmrWrite_Tick(object eventSender, System.EventArgs eventArgs)
        {
            writeAllToFile();
            resetAll();
        }

        public void restAll()
        {

            dataList.Clear();
            clearBuffer = false;
            convertCount = 0;
            meanValue = 0;
            first = true;
            hasTrapped = false;
            firstTrapAction = true;
            bufferBar.ForeColor = Color.Blue;
            bufferBar.Value = 0;


            tmrWrite.Enabled = false;
```

```
            tmrBuffer.Enabled = true;
            bufferBar.ForeColor = Color.Blue;

        }


        public void resetAll()
        {

            dataList.Clear();
            clearBuffer = false;
            convertCount = 0;
            meanValue = 0;
            first = true;
            hasTrapped = false;
            firstTrapAction = true;
            bufferBar.ForeColor = Color.Blue;
            bufferBar.Value = 0;


            if (emailBox.Checked == true)
            {

                MailMessage objeto_mail = new MailMessage();
                SmtpClient client = new SmtpClient();
                client.Port = 587;
                client.Host = "smtp.gmail.com";
                client.Timeout = 10000;
                client.EnableSsl = true;
                client.DeliveryMethod = SmtpDeliveryMethod.Network;
                client.UseDefaultCredentials = false;
                client.Credentials = new
System.Net.NetworkCredential("trappingsetupalpha", "allyourbases");
                objeto_mail.From = new MailAddress("trappingsetupalpha@gmail.com");
                objeto_mail.To.Add(new MailAddress(emailField.Text));
                objeto_mail.Subject = "Trapping update!!!";
                objeto_mail.Body = "Congratulations, your particle has just trapped!";
                client.Send(objeto_mail);

            }

        }

        private void tmrMs_Tick(object eventSender, System.EventArgs eventArgs)
        {









        }
        private void tmrBuffer_Tick(object eventSender, System.EventArgs eventArgs)
        {
            convertCount += 100;
```

```
            if (convertCount >= int.Parse(bufferSizeComboBox.SelectedItem.ToString()) *
1000)
            {
                clearBuffer = true;
            }
            bufferBar.Value = convertCount;

        }


        public double computeMean()
        {
            double sum = 0;
            foreach (double i in dataList)
            {
                sum += i;
            }

            return sum/((double)dataList.Count);
        }


        private void tmrConvert_Tick(object eventSender, System.EventArgs eventArgs)
        {



            tmrConvert.Stop();

            double HighResEngUnits;
            MccDaq.ErrorInfo ULStat;
            int Chan;
            String chanString = "1";
            bool ValidChan = int.TryParse(chanString, out Chan);



            int Rate = 50000; //  per channel sampling rate ((samples per second) per
channel)
            int Count = 100; //  total number of data points to collect

            ADData = new ushort[Count]; // array to hold the input values
            ADData32 = new uint[Count]; //  dimension an array to hold the high
resolution input values
            tempVals = new double[Count];


            MccDaq.Range Range = MccDaq.Range.Bip5Volts; // set the range
            ULStat = DaqBoard.AInScan(Chan, Chan, Count, ref Rate, Range, MemHandle,
Options);

            if (ULStat.Value == MccDaq.ErrorInfo.ErrorCode.FreeRunning)
            { // Turn off library error handling for subsequent calls
                ULStat = MccDaq.MccService.ErrHandling(MccDaq.ErrorReporting.DontPrint,
MccDaq.ErrorHandling.StopAll);
            }
```

```
            if (HighResAD)
            {
                ULStat = MccDaq.MccService.WinBufToArray32(MemHandle, ADData32,
FirstPoint, Count);

                for (int i = 0; i < ADData32.Length; ++i)
                {

                    ULStat = DaqBoard.ToEngUnits32(Range, ADData32[i], out
HighResEngUnits);
                    double val = double.Parse(HighResEngUnits.ToString("F5"));//  print
the counts

                    if (val != 0)
                    {

                        tempDataList.Add((float)val);
                        //Console.WriteLine(val);
                    }
                }

            }


            else
            {
                ULStat = MccDaq.MccService.WinBufToArray(MemHandle, ADData, FirstPoint,
Count);

                for (int i = 0; i < ADData.Length; ++i)
                {
                    ULStat = DaqBoard.ToEngUnits32(Range, ADData[i], out
HighResEngUnits);
                    double val = double.Parse(HighResEngUnits.ToString("F5"));//  print
the counts

                    if (val != 0)
                    {

                        tempDataList.Add((float)val);
                        // Console.WriteLine(val);
                    }


                }
            }

            float meanVal = getMean(tempDataList);

            dataChart.Series["BufferData"].Points.AddY(meanVal);
            valbox.Text = meanVal+"";
            rmsBox.Text = calculateMeanRMS(50)+"";
            dataList.AddRange(tempDataList);

            tempDataList.Clear();

                if (first)
```

```
        {
            meanValue = meanVal;
            first = false;
        }


        if (hasTrapped == false && trappingBox.Checked)
        {
            hasTrapped = checkForTrap(meanVal);
            meanValue = computeMean();

        }

        if (hasTrapped && firstTrapAction == true)
        {
            tmrWrite.Enabled = true;
            tmrBuffer.Enabled = false;
            firstTrapAction = false;
            bufferBar.Value = bufferBar.Maximum;

            bufferBar.ForeColor = Color.Red;
        }

        if (clearBuffer && hasTrapped == false)
        {
            dataChart.Series["BufferData"].Points.Clear();
            dataList.Clear();
            clearBuffer = false;
            convertCount = 0;
        }


    tmrConvert.Start();

}


public float sumOfSquares(List<float> data)
{
    float sum = 0;
    for (int i = 0; i < data.Count; i++)
    {

        sum += (float)data[i] * (float)data[i];

    }

    return sum;

}


public float STD(List<float> data)
{
    float sumSquare = sumOfSquares(data);
    float mean = getMean(data);
    return (float)Math.Sqrt((sumSquare / data.Count) - mean * mean);
}
```

```
public float calcRMS(List<float> data)
{
    float RMSval = 0;
    RMSval = STD(data);
    return RMSval;
}


public float getMean(List<float> data)
{
    float sum = 0;
    for (int i = 0; i < data.Count; i++)
    {

        sum += (float)data[i];

    }
    return sum / ((float)data.Count);
}


public float calculateMeanRMS(int window)
{
    List<float> RMSVals = new List<float>();
    List<float> tempList = new List<float>();
    for (int i = 0; i < dataList.Count - window; i += window)
    {

        for (int j = i; j < i + window; j++)
        {
            tempList.Add((float)dataList[j]);
        }
        //normalizeToMax(tempList);
        RMSVals.Add(calcRMS(tempList));
        tempList.Clear();
    }

    return getMean(RMSVals);
}

public bool checkForTrap(float val)
{
    double perc = double.Parse(percentComboBox.SelectedItem.ToString());
    if (val >= meanValue + (meanValue * (perc / 100.0)))
    {
        return true;
    }

    return false;
}


private void InitUL()
{

    AIOProps.ReportError = MccDaq.ErrorReporting.PrintAll;
```

```
        AIOProps.HandleError = MccDaq.ErrorHandling.StopAll;
        MccDaq.ErrorInfo ULStat = MccDaq.MccService.ErrHandling
            (AIOProps.ReportError, AIOProps.HandleError);


        // Get the resolution of A/D
        int ADRes;
        DaqBoard.BoardConfig.GetAdResolution(out ADRes);

        // check If the resolution of A/D is higher than 16 bit.
        // If it is, then the A/D is high resolution.
        if (ADRes > 16)
            HighResAD = true;

        //  set aside memory to hold data
        if (HighResAD)
            MemHandle = MccDaq.MccService.WinBufAlloc32Ex(numPoints);
        else
            MemHandle = MccDaq.MccService.WinBufAllocEx(numPoints);

        Options = MccDaq.ScanOptions.ConvertData;

    }

    public DAQWATCHER()
    {
       InitializeComponent();
       series = this.dataChart.Series.Add(seriesArray[0]);
    }



    [STAThread]
          static void Main()
          {
                Application.Run(new DAQWATCHER());
          }

    private void label1_Click(object sender, EventArgs e)
    {

    }

    private void textBox1_TextChanged(object sender, EventArgs e)
    {

    }

    public void writeAllToFile()
    {
        tmrMs.Enabled = false;
        tmrBuffer.Enabled = false;
        tmrConvert.Enabled = false;
        tmrWrite.Enabled = false;

        using (System.IO.StreamWriter file = new
System.IO.StreamWriter(@"AutoTrapCapture.txt"))
        {
```

```
            foreach (double value in dataList)
            {
                    file.WriteLine(value+"");
            }
        }
    }

    private void startButton_Click(object sender, EventArgs e)
    {
        dataChart.Series["BufferData"].Points.Clear();
        bufferBar.Maximum = int.Parse(bufferSizeComboBox.SelectedItem.ToString()) *
1000;

        if (tmrMs.Enabled)
        {
            tmrMs.Enabled = false;
        }
        else
        {
            tmrMs.Enabled = true;
        }

        if (tmrBuffer.Enabled)
        {
            tmrBuffer.Enabled = false;
        }
        else
        {
            tmrBuffer.Enabled = true;
        }


        if (tmrConvert.Enabled)
        {
            tmrConvert.Enabled = false;
        }
        else
        {
            tmrConvert.Enabled = true;
        }
    }

    private void label5_Click(object sender, EventArgs e)
    {

    }

    private void label3_Click(object sender, EventArgs e)
    {

    }

    private void textBox1_TextChanged_1(object sender, EventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
```

```
        {
            hasTrapped = true;
        }

        private void button2_Click(object sender, EventArgs e)
        {
            restAll();
        }


    }
}
```

# Appendix C: Automatic temperature sweeping and data logging code

Thread able program that can run multiple GPIB devices iteratively while taking data from a DAQ card.

Written in C#, the program iteratively tunes the temperature of a laser diode controller to adjust the center wavelength of the diode. Once the temperature is within .1 percent of the desired value, a 5 second window of data is taken and recorded into a text file. Additionally the temperature of the laser diode is also recorded and stored in a different text file. After the data is saved, the process continues until the desired ending temperature is reached.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Diagnostics;
using System.Threading.Tasks;
using System.Windows.Forms;
using Ivi.Visa.Interop;
using Automation.BDaq;
using Thorlabs.TL4000;
using System.Diagnostics;
using System.Collections;
using System.Threading;
using System.Runtime.InteropServices;
namespace WindowsFormsApplication1
{


    public class SweepWorker
    {
        private bool driverInitialized = false;
        private Thorlabs.TL4000.TL4000 driver;
        private List<TL4000Device> tl4000Devices;
        private const int BUFFER_SIZE = 256;
        List<float> dataList = new List<float>();
        public List<float> rmsList = new List<float>();
        public List<float> tempList = new List<float>();
        double startingTemp = 0;
        double endingTemp = 0;
```

```
        double currentTemp = 0;
        double stepSize = 0;
        String filePrefix = "";
        int fileCount = 0;
        Boolean firstRun = true;
        Form1 form;
        MccDaq.MccBoard DaqBoard = new MccDaq.MccBoard(0);
        MccDaq.Range Range;
        private int HighChan = 3, NumAIChans = 4;
        private int ADResolution;

        private IntPtr MemHandle = IntPtr.Zero;
        private bool HighResAD = false;
        private MccDaq.ScanOptions Options;


        static int numPoints = 5000*10;
        const int FirstPoint = 0;
        private ushort[] ADData = new ushort[numPoints]; // array to hold the input
values
        private uint[] ADData32 = new uint[numPoints]; //  dimension an array to hold the
high resolution input values
        public double[] tempVals = new double[numPoints];

        AnalogIO.clsAnalogIO AIOProps = new AnalogIO.clsAnalogIO();


        public SweepWorker(String dir, String name, double startTemp, double endTemp,
double step, Form1 form)
        {
            tl4000Devices = new List<TL4000Device>();
            tl4000Devices.Add(new TL4000Device(TL4000_Devices.CLD1000, "0"));
            startingTemp = startTemp;
            endingTemp = endTemp;
            stepSize = step;
            filePrefix = dir+name;
            this.form = form;
            TL4000_ConnectDevice();
            InitUL();
        }


        private void InitUL()
        {

            //  Initiate error handling
            //   activating error handling will trap errors like
            //   bad channel numbers and non-configured conditions.
            //   Parameters:
            //     MccDaq.ErrorReporting.PrintAll :all warnings and errors encountered
will be printed
            //     MccDaq.ErrorHandling.StopAll   :if an error is encountered, the
program will stop

            AIOProps.ReportError = MccDaq.ErrorReporting.PrintAll;
            AIOProps.HandleError = MccDaq.ErrorHandling.StopAll;
            MccDaq.ErrorInfo ULStat = MccDaq.MccService.ErrHandling
```

```
            (AIOProps.ReportError, AIOProps.HandleError);


        // Get the resolution of A/D
        int ADRes;
        DaqBoard.BoardConfig.GetAdResolution(out ADRes);

        // check If the resolution of A/D is higher than 16 bit.
        // If it is, then the A/D is high resolution.
        if (ADRes > 16)
            HighResAD = true;

        //  set aside memory to hold data
        if (HighResAD)
            MemHandle = MccDaq.MccService.WinBufAlloc32Ex(numPoints);
        else
            MemHandle = MccDaq.MccService.WinBufAllocEx(numPoints);

        Options = MccDaq.ScanOptions.ConvertData;

    }




    private void takeDataStream()
    {

        float EngUnits;
        double HighResEngUnits;
        MccDaq.ErrorInfo ULStat;
        System.UInt16 DataValue;
        System.UInt32 DataValue32;
        int Chan;
         int dataCount = 0;
         String chanString = "1";
         bool ValidChan = int.TryParse(chanString, out Chan);



        int Rate = 5000; //  per channel sampling rate ((samples per second) per
channel)
        int Count = Rate*5; //  total number of data points to collect

        ADData = new ushort[Count]; // array to hold the input values
        ADData32 = new uint[Count]; //  dimension an array to hold the high
resolution input values
        tempVals = new double[Count];


        MccDaq.Range Range = MccDaq.Range.Bip5Volts; // set the range
        ULStat = DaqBoard.AInScan(Chan, Chan, Count, ref Rate, Range, MemHandle,
Options);

        if (ULStat.Value == MccDaq.ErrorInfo.ErrorCode.FreeRunning)
        { // Turn off library error handling for subsequent calls
```

```
                ULStat = MccDaq.MccService.ErrHandling(MccDaq.ErrorReporting.DontPrint,
MccDaq.ErrorHandling.StopAll);
            }


            if (HighResAD)
            {
                ULStat = MccDaq.MccService.WinBufToArray32(MemHandle, ADData32,
FirstPoint, Count);

                for (int i = 0; i < ADData32.Length; ++i)
                {

                    ULStat = DaqBoard.ToEngUnits32(Range, ADData32[i], out
HighResEngUnits);
                    double val = double.Parse(HighResEngUnits.ToString("F5"));//  print
the counts

                    if (val != 0)
                    {
                        dataCount++;
                        dataList.Add((float)val);
                        //Console.WriteLine(val);
                    }
                }

            }


            else
            {
                ULStat = MccDaq.MccService.WinBufToArray(MemHandle, ADData, FirstPoint,
Count);

                for (int i = 0; i < ADData.Length; ++i)
                {
                    ULStat = DaqBoard.ToEngUnits32(Range, ADData[i], out
HighResEngUnits);
                    double val = double.Parse(HighResEngUnits.ToString("F5"));//  print
the counts

                    if (val != 0)
                    {
                        dataCount++;
                        dataList.Add((float)val);
                       // Console.WriteLine(val);
                    }


                }
            }


            using (System.IO.StreamWriter file = new System.IO.StreamWriter(@filePrefix +
fileCount + ".txt"))
            {
                foreach (double value in dataList)
                {
```

```
                if (value != 0)
                {
                    file.WriteLine(value + "");
                }
                else
                {
                    //  return;
                }
            }
        }


        float rmsMean = calculateMeanRMS(40);
        rmsList.Add(rmsMean);
        tempList.Add(getBeatFromTemp((float)currentTemp));
     //   Console.WriteLine("In thread count : " + rmsList.Count);
        dataList.Clear();

    }



    public float getBeatFromTemp(float temp)
    {
        float firstWL = float.Parse(this.form.wlTextBox.Text);

        // float secondWL = 0.048137F * temp + 851.99F;
        //float secondWL = .05275F * temp + 852117.7F;
        // float secondWL = .051991F * temp + 851.76F;
        float secondWL = .041F * temp + 973.793F;  // ahmed Dec18, 2014
        float Cnms = 2.9979e17F;
        float w1 = firstWL;
        float w2 = secondWL;
        float freq = Math.Abs(Cnms / w1 - Cnms / w2);
        freq = freq * 1e-9F;
        return freq;
    }



    private void TL4000_ConnectDevice()
    {
        if (driverInitialized)
            this.driver.Dispose();
        tl4000Devices[0].SerialNumber = "M00299556";

        try
        {
            this.driver = new Thorlabs.TL4000.TL4000(tl4000Devices[0].ResourceString,
true, true);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
            return;
        }

        StringBuilder manufacturer = new StringBuilder(BUFFER_SIZE);
```

```
            StringBuilder deviceName = new StringBuilder(BUFFER_SIZE);
            StringBuilder firmware = new StringBuilder(BUFFER_SIZE);

            int result = this.driver.identificationQuery(manufacturer, deviceName, null,
firmware);
            this.driver.switchTecOutput(true);
            this.driver.switchLdOutput(true);

        }

        public void setTemp(float newTemp)
        {
            this.driver.setTempSetpoint(newTemp);
        }

       public void setWaveLength(float newWL)
       {

       }


        public void DoWork()
        {

            while (!_shouldStop)
            {

                while (_shouldPause)
                {

                }
                ResourceManager rm;
                IMessage session = null;
                double temp = startingTemp + stepSize * fileCount;
                currentTemp = temp;
                if (temp >= endingTemp)
                {
                    _shouldStop = true;
                }

                try
                {

                 //rm = new ResourceManager();
                 //  session = (IMessage)rm.Open("GPIB0::5",
Ivi.Visa.Interop.AccessMode.NO_LOCK, 0, "");
                  // Debug.WriteLine("Sopping");
                   //session.WriteString(":WAVELENGTH " + wavelength);
                    setTemp((float)temp);
                    using (System.IO.StreamWriter file = new
System.IO.StreamWriter(@filePrefix + fileCount + "-wavelength.txt"))
                    {
                        file.WriteLine(temp);
                    }


                }
                finally
```

```
                {
                 //   session.Close();
                }

                System.Threading.Thread.Sleep(1000);
//Logging for taking data from the OSA if calibration for new lasers is needed
                if (false)
                {
                    try
                    {

                        rm = new ResourceManager();

                        session = (IMessage)rm.Open("GPIB::5",
Ivi.Visa.Interop.AccessMode.NO_LOCK, 0, "");

                        session.WriteString("syst:comm:gpib:buff on");
                        session.WriteString("form ascii");
                        session.WriteString("sens:swe:poin 1000");
                        session.WriteString("init:imm;*opc?");
                        session.WriteString("trac:data:y? tra");
                        String Data = session.ReadString(25600);
                        // Debug.WriteLine(Data);

                        String[] dataArray = Data.Split(',');
                        double[] sweepValues = new double[dataArray.Length];
                        double[] wavelengthValues = new double[dataArray.Length];
                        session.WriteString("sens:wav:star?");
                        double Startw = Double.Parse(session.ReadString(256));
                        session.WriteString("sens:wav:stop?");
                        double Endw = Double.Parse(session.ReadString(256));

                        session.WriteString("sens:swe:poin?");
                        double sweepLength = Double.Parse(session.ReadString(256));

                        double Bucket = (Endw - Startw) / (sweepLength - 1);
                        // OSACHART.Series["dataSeries"].Points.Clear();
                        for (int i = 0; i < dataArray.Length; i++)
                        {
                            wavelengthValues[i] = (Startw + (Bucket * (i))) * 1000000000;
                            //Debug.WriteLine(wavelengthValues[i]);
                            sweepValues[i] = Double.Parse(dataArray[i]) * -1.0;
                        }

                        using (System.IO.StreamWriter file = new
System.IO.StreamWriter(@filePrefix + fileCount + "-amp.txt"))
                        {
                            foreach (double value in sweepValues)
                            {
                                file.WriteLine(value + "");
                            }
                        }


                        using (System.IO.StreamWriter file = new
System.IO.StreamWriter(@filePrefix + fileCount + "-wl.txt"))
                        {
                            foreach (double value in wavelengthValues)
```

```
                    {
                        file.WriteLine(value + "");
                    }
                }
            }
            finally
            {
                session.Close();
            }
        }

        takeDataStream();
        fileCount++;
    }


}

public float sumOfSquares(List<float> data)
{
    float sum = 0;
    for (int i = 0; i < data.Count; i++)
    {

        sum += (float)data[i] * (float)data[i];

    }

    return sum;

}


public float STD(List<float> data)
{
    float sumSquare = sumOfSquares(data);
    float mean = getMean(data);
    return (float)Math.Sqrt((sumSquare / data.Count) - mean * mean);
}


public float calcRMS(List<float> data)
{
    float RMSval = 0;
    RMSval = STD(data);
    return RMSval;
}


public float getMean(List<float> data)
{
    float sum = 0;
    for (int i = 0; i < data.Count; i++)
    {

        sum += (float)data[i];

    }
```

```
            return sum / ((float)data.Count);
        }


        public float calculateMeanRMS(int window)
        {
            List<float> RMSVals = new List<float>();
            List<float> tempList = new List<float>();
            for (int i = 0; i < dataList.Count - window; i += window)
            {

                for (int j = i; j < i + window; j++)
                {
                    tempList.Add(dataList[j]);
                }
                //normalizeToMax(tempList);
                RMSVals.Add(calcRMS(tempList));
                tempList.Clear();
            }

            return getMean(RMSVals);
        }

        public void normalizeToMax(List<float> data)
        {
            for (int i = 0; i < data.Count; i++)
            {
                data[i] = data[i] / ((float)data.Max());

            }
        }

        public double getProgress()
        {
            return fileCount / ((endingTemp - startingTemp) / ((double)stepSize));

        }
        public void RequestStop()
        {
            _shouldStop = true;
        }
        public void RequestPause()
        {
            if (_shouldPause == true)
            {
                _shouldPause = false;
            }
            else
            {
                _shouldPause = true;
            }

        }

        private volatile bool _shouldPause = false;
        private volatile bool _shouldStop = false;
    }
}
```