

Physics 408

Optics Laboratory

Department of Physics & Astronomy
UBC



2023/2024 Winter Term 2
(Last edited December 22, 2023 by V. Milner)

Chapter 1

Rules and Resources

1.1 Safety Rules – READ THIS IN FULL!

Please do not be apprehensive of these labs. If you are careful, the danger involved in working with laser beams used here is extremely minimal. However, if you fail to heed the following warnings, bad things may happen. Because of the importance of these safety rules, we will be penalizing (subtracting marks!) those who don't follow them.

Do Not enter the lab without permission

Even if you have card access, you are not allowed in the Optics Lab without one of the lab supervisors, i.e. one of the TAs, Dr. Van Dongen or Dr. Milner being present! No supervisor - no entry, no exceptions!

Do Not enter a curtained section without permission

If you see that the curtain around a particular section is closed, do not poke your head in! The occupants might be either doing some sensitive data collection, which requires darkness, or they may be aligning their laser beam and send it inadvertently towards the curtain. Either way, you will ruin their data or your eyes, so always ask from outside before entering.

Do Not touch the high voltage electrodes in the HeNe lab

There is a plastic casing surrounding the laser tube, there is no reason for you to remove this casing or to place your hands within the confines of it. People have died by mishandling laser power supplies. You won't, but there is no reason to test this theory.

Do Not stare into the laser beam

In this lab we use class 3R (1-5 mW) and 3B (5-500 mW) HeNe lasers. If for some reason the beam does impact near the area of your eyes your “blink reflex” should be enough to protect you for a very short exposure. That being said, you should never place your head/eye in the path of the beam to see where it is going or for any reason at all. This might also happen by accident if,

for example, you bend down to the beam level or an optic is being moved or removed with the laser not being blocked. A small index card (provided) is a much better means of observing the path of the beam. It is also important to watch for stray reflections off of mirrors or other reflective surfaces. This means that any **watches, rings, or bracelets should be removed** before beginning this experiment.

If you do get a laser beam or scatter in your eye please warn your partner and do not try to repeat the process to figure out what happened. If possible block the beam or turn off the laser and report the incident to lab staff. If you want to check for laser scatter one method is to put your BACK to the experiment. With the room lights off place a white sheet of paper in FRONT of you where you would like to check if the laser light is hitting the paper.

Laser goggles are provided and should be used when you are aligning the optical components in your setup. If you feel uncomfortable with the light intensity or anything else laser related, please seek out assistance. Curtains and doors are provided to protect others from your laser equipment. Please be mindful to appropriately block out access to random bystanders so they do not have direct line of sight to your optical equipment.

Do Not touch any optical surface

It does not take many impurities on the optical surfaces (e.g. scratches, finger-prints, and even dust) to prevent lasing action from occurring in the HeNe lab, to ruin the cavity mode in the Cavity lab, to mess up the interference pattern in the Michelson lab, or to destroy the Fourier image in the Fourier lab. Either hold optical elements by their mounts or on their edges as applicable.

Do Not eat or drink in the lab rooms

If you must eat, do so in the hallway.

Do Not move optics mounts or other hardware fixed to the bench with screws painted in red

These are elements that are already in their proper place and moving them will make doing the lab impossible. If you don't have a tool to loosen a screw, you probably shouldn't be trying to move that component. If you do move an optical element held with red screws or suspect that such an element is not in its proper place, please immediately contact the lab TA or professor to help you reset this element - obtaining good data and completing the lab may depend on it.

Do Not place foreign objects close to the optical setup

It feels very convenient to place your laptop, or your paper notebook, right by your setup and type/write your report as you work. Do not do it! A computer monitor or a pen can easily scatter light into your eyes and cause serious injuries. Plastic covers, e.g. in the HeNe lab, were not designed or tested to hold extra weight and will easily break if used as desk surfaces.

1.2 Logistics and Resources

How will the labs be marked?

All questions in the lab manual labeled with red upper-case letters, e.g. **(A)**, **(B)**, **(C)**, etc., are for marks. Those which are labeled with blue lower-case roman numerals, e.g. **(i)**, **(ii)**, **(iii)**, etc., are for bonus points. Although the weights of each point may be slightly different, you may expect that all of them will be pretty similar.

Working with lab desktop computers

Although you are welcome to bring and use your own laptop computer, you will need lab desktop computers for data collection. After you login to the lab computers with your CWL account, you will be able to store your files and data under `C:\Users\yourUsername\Desktop\yourFolders`. Please be mindful of the (relatively) limited space on the local hard drive and clean after yourself when you are done with a particular lab.

On every lab desktop, you will also see the following folders with read-only files:

- `C:\Users\public\Desktop\labName`
- `C:\Users\public\Desktop\hardware`
- `C:\Users\public\Desktop\software`

The `labName` directories, with `labName=Cavity, HeNe, Michelson or Fourier`, contain useful materials pertaining to each particular lab, including this manual and videos explaining the relevant optical alignment procedures.

The `hardware` folder contains subdirectories with technical spec sheets and manuals of all electronic, optical and opto-mechanical components. These are useful for various calibration procedures, which you are asked to do in these labs. Computer communication protocols can also be found in these folders.

For several experiments in this course you will use the Raspberry Pi CCD cameras to capture images or movies – please read Appendix A for details on how to operate these cameras in real time.

Finally, the `software` directory contains a number of examples of MATLAB scripts, demonstrating the way you can communicate with the hardware components used in all four labs, e.g. `raspiCamera.m` for capturing images and movies with a Raspberry Pi camera or `thorlabsStage.m` for controlling the position of a Thorlabs translation stage.

Feel free to copy these files (especially MATLAB scripts) to your own folders and modify them for your own purposes.

Pacing yourself

If you find yourself spending more than 10 or 15 minutes trying (unsuccessfully) to get the proper alignment of any element in any part of any experiment, you should seek the advice of a TA or lab instructor (or a friend!) to get you past this hurdle. The point of the lab is both to learn lab techniques (such as optical

alignment) and to perform a certain set of experiments. Make sure you don't spend too much time on any given task since you are expected to complete each experiment (typically, up to six separate experimental tasks).

File sharing platform and feedback

As you work on each experiment within your lab, you will collect experimental data. To share the data with your lab partner, as well as with your TA, who may provide you with important feedback, we will use the Microsoft Teams platform. Look on Canvas for the link to your lab Team, appropriately called 'PHYS 408 L2X 2012W2 Optics', where X=A,B,C or D. Inside your Team, you will find multiple channels. The main public channel called 'General' will be used for general announcements, notifications and all labs related discussions within your section. On top of that, you and your lab partner will have a private channel with a name similar to 'Lab1_Cavity_Optics3' or 'Lab2_Fourier_Optics5', where the first part 'LabX' (X=1,2 or 3) says whether this is your first, second or third lab in this course; the second part is the type of the lab; and the third part 'OpticsYY' (YY=1,...,12) is the name of the desktop which serves the specific setup you are working on. These private channels will be pre-set by your TA and used for storing and sharing data within your group (the files are stored on your UBC OneDrive and can be accessed from anywhere), as well as for seeking feedback from your TA by showing your intermediate results and asking questions pertaining to your specific lab and group.

Lab report format and submission

As you work on each experiment in your lab, it is critical to keep a detailed and organized *electronic* lab notebook. Since the provided examples of communicating with experimental hardware and acquiring data are written in MATLAB (see `C:\Users\public\Desktop\software` on your lab desktop), and because MATLAB is an extremely powerful tool for data processing available to every UBC student, you are encouraged to use MATLAB "Live Scripts" as a means of keeping your electronic notes¹. In the course of each lab, keep your electronic notebook in your private MS Teams channel, where it will be accessible to your TA and used for giving you feedback on your progress, providing help and guidance. At the end, you will submit your final lab report as a pdf file on Canvas. It is your responsibility to keep track of the deadlines, outlined on the main Canvas page.

1.3 Collaborations and academic integrity

Although you and your partner will be working in the lab together, and although we do encourage scientific collaboration among students working on the same project, it goes without saying that everybody is expected to complete their work independently. Most data acquisitions are relatively quick; hence, you and your partner should be using distinct data sets (probably, taken one after another) in your individual lab reports. If for some reason you want to share data with your partner (e.g. due to the very long data collection time), please get

¹Python scripts and Jupiter notebooks will be accepted as well

an approval from your TA first. Close similarities between any two lab reports will be considered as plagiarism and will be treated with utmost seriousness following an official UBC policy on academic misconduct.

Chapter 2

The Michelson Interferometer

2.1 Objective

In an optical interferometer, the incident light beam is divided into two parts by means of a beam-splitter. The divided beams propagate through different paths and then are recombined. Depending on the difference in optical path lengths of the two beams, the recombined beams may be in phase and produce constructive interference, or 180° out of phase and produce destructive interference. Observation of the resulting interference fringes as a function of various physical parameters is used in numerous studies and applications. In this lab you will explore and use one of the most popular interferometers – the Michelson interferometer, while recording interference patterns produced by light from three different sources (a HeNe laser, a sodium lamp and a white light source). Specific objectives are:

1. To investigate the shape of interference patterns.
2. To understand the properties and the calibration procedures of the Michelson interferometer.
3. To use the interferometer for exploring spectral characteristics of various sources of light.
4. To learn the technique and understand the principles of Fourier Transform spectroscopy.

2.2 Pre-lab Study

Before you start this lab, familiarize yourself with the following material from the course textbook, “Fundamentals of Photonics” by Saleh & Teich, which provides essential background for completing the required experimental tasks. Note that all this material has been covered (or will be covered - depending on your lab schedule), in class before (or after) your completion of this lab. In the latter case, you do not need to understand every single detail in the text

(this will be thoroughly discussed in lectures), but you should form the general conceptual picture and become familiar with the mathematical tools used in the relevant topics. The following list refers to the **Third Edition (2019)** of the textbook.

- **Experiments 1,2 and 3:** Chapter 2.2B on the spherical waves and Chapter 2.5A on the interference of two waves; Example 2.5-1 on Michelson interferometer (as used in LIGO).
- **Experiments 4,5 and 6:** Chapter 12.1B on the temporal coherence and spectrum; Chapter 12.2A and B on the interference of partially coherent waves and Fourier-Transform spectroscopy, as well as on the Effect of spectral width on interference.
- Appendices A, B, C.

2.3 Experiments

Figure 2.1 illustrates the physical arrangement of your experimental setup. The light beam, originating from the light source **S** (e.g. a HeNe laser), gets routed towards the interferometer with mirrors **M**_a and **M**_b. In the case of the HeNe beam, it is expanded with lens **L** (not needed for the sodium lamp or the white light source) before passing through the beamsplitter cube **BSC**. The beam-splitter reflects half of the beam towards mirror **M**₂, and transmits the other half of the beam towards mirror **M**₁. After reflection from these plane mirrors, the two beams are recombined on the beam splitter and exit the interferometer through the ground glass plate **GG** and towards the **CCD** camera. ‘Folding’ mirror **M**_f allows to increase the distance between the ground glass plate and the camera as required for proper imaging. Mirror **M**₂ is mounted on a Thorlabs translation stage which can be controlled either by hand or from the computer (see `thorlabsStage.m`). Vacuum cell **V** at the top of the picture can be lowered into one of the arms of the interferometer for the experiment on the refractive index of air. Not shown in the picture is a neutral-density (ND) filter installed after the HeNe laser to cut down the laser power if desired.

2.3.1 Interference pattern

Start your experiments using a HeNe laser as a light source (as in Fig. 2.1). For the initial alignment, remove the expanding lens **L**. **CAUTION:** as always, block the laser beam before removing or inserting optical components into the beam path! To help you find the fringes, place the ground glass plate **GG** after the cube. Verify that the Raspberry Pi camera is focused on the ground glass using the focusing aid provided and adjusting the camera’s objective lens. Please do not over-tighten the black adjustment ring on the camera against the housing body! Turn on the laser and adjust mirrors **M**₁ and **M**₂ so that the two spots, one from each arm, that you see on the output face of the cube (or the ground glass plate) overlap with one another.

Note that although it is common to draw a Michelson interferometer with the light source at precisely 45° with respect to the beam splitter, and the arm-defining mirrors set exactly perpendicular to the beams from the beam splitter,

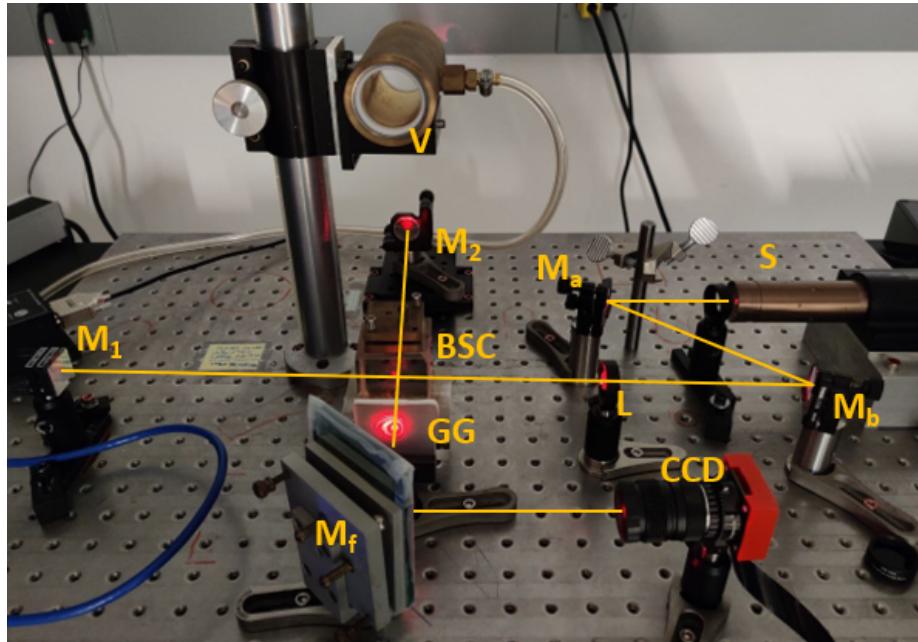


Figure 2.1: An aligned interferometer with the interference fringes visible on the ground glass plate (**GG**). The laser path is shown with orange lines for reference.

this does not have to be the case. In reality, the laser, the beamsplitter and mirror **M**₁ may be placed casually on an optical table, with no more care than is needed to assure that the laser beam strikes the mirror and its reflection returns to the beamsplitter. If mirror **M**₂ is angled so that the reflection from it goes to the same spot on the beam splitter, the instrument will be symmetric and will produce an interference pattern with high contrast fringes.

You may start seeing the beam flickering as soon as the dots overlap – a result of the interference you are looking for. To enlarge the beams and observe the interference pattern, insert lens **L** in front of the beamsplitter cube, as shown in Figure 2.1.

- (A) When the interferometer is aligned properly, you should have a nice expanded interference pattern which looks similar to the one pictured in the upper corner of Figure 2.2. Record the image and explain the circular symmetry of the fringes. What happens to the interference pattern when you tilt one of the mirrors? Record the corresponding image and explain whether the circular symmetry disappears and why? *Hint: If you are having difficulty recognizing circular symmetry, try moving the stage to another position. Do not get stuck on the search for the perfectly round fringes. In practice, the fringes may be distorted due to imperfections in the flatness of the mirrors. In fact, Michelson and other interferometers are used in industrial fabrication plants exactly for this purpose - to verify the flatness of mirrors.*

- (i) What determines the spacing between the concentric fringes? To answer

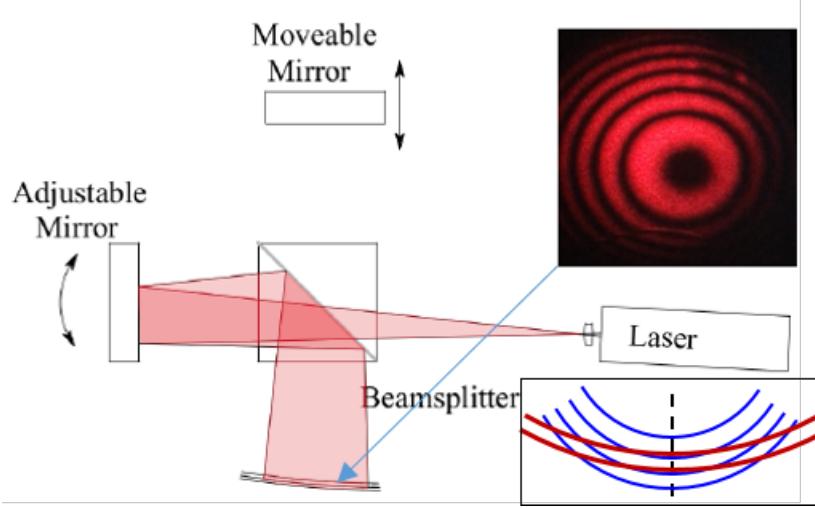


Figure 2.2: If a small lens is placed in front of the laser, the beam expands. This does not alter the alignment of the interferometer, but merely increases the output wavefronts (only one arm is shown for clarity). Circular fringe patterns from a well aligned interferometer is shown in the upper right corner. Adjusting the alignment of mirrors M_1 and M_2 will center this circular pattern. Adjusting the position of the moveable mirror to alter the optical path difference will make the fringes grow or shrink. This could be explained by sketching the two overlapping spherical wavefronts, which traveled different optical paths (bottom right corner).

this question, add the missing trace for the second arm in Fig. 2.2 and analyze the overlap of the two wavefronts at the output. For simplicity, model the two interfering fields as spherical waves which traveled slightly unequal optical paths between their origin and the interference plane, and therefore have different radii of curvature. Your sketch of the two wavefronts should look similar to the one in the bottom right corner of Fig. 2.2. How does the fringe spacing depend on the optical path difference? *Hint: identify on the sketch the regions of constructive interference; what happens to them when the path lengths are very close to one another?* Move mirror M_2 by jogging the translation stage and record a few fringe patterns at different positions of the mirror. Do your observations agree with the theoretical predictions? Can you tell whether you are moving towards smaller or bigger difference between the two arm lengths?

2.3.2 Index of refraction of air

The interferometer can be used to characterize extremely small changes in the optical path length differences between the two arms. You can use this sensitivity to even measure the phase shift induced by the propagation of light through air and, by extension, the index of refraction of air. Imagine removing air from one arm of the interferometer and then putting it back in. The phase difference

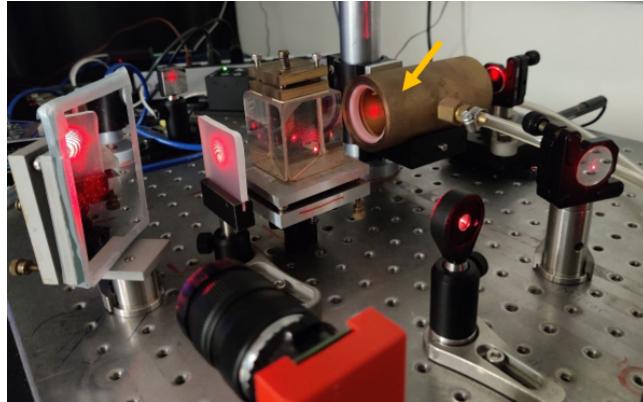


Figure 2.3: Vacuum chamber (marked with an arrow) inserted into the arm of the interferometer.

due to the difference in the optical path length between the two cases (with and without air) is:

$$\Delta\phi = kd(n_{\text{air}} - 1), \quad (2.1)$$

where k is the wave vector, d is twice the length of the evacuated part, and n_{air} is the refractive index of air at the wavelength of the light (here, 632.8 nm). Since every time the phase difference changes by 2π , the interference pattern shifts by exactly one fringe, one can determine $\Delta\phi$ by counting the number of fringes moving through a small region in the interference pattern as you are introducing air back into one of the interferometer arms. This is exactly the technique you will be using in this experiment, keeping the HeNe laser as your light source. Prepare your data acquisition MATLAB script which will allow you to record the brightness of the interference pattern within a small region of interest (smaller than the size of an average fringe on the camera) as a function of time. *Hint: the best way to do it is by recording a video with a known frame rate – refer to raspiCamera.m to see how it can be done. Aim to collect a video at 30 frames per second (the highest frame rate available) for the entire time that the air is leaking into the cell.*

Insert the glass vacuum cell by lowering it between the beam-splitter and mirror M_2 , as shown in Fig. 2.3. Be careful to do it without hitting the mirrors. Please support the air cell from the bottom while lowering or raising it in case it falls down unexpectedly. To prepare a vacuum in the cell, close the “leak valve” and open the “pump valve” (see Fig. 2.4) and turn on the vacuum pump. **CAUTION:** remember to shield your eyes from the HeNe laser when bending down to turn on the pump. After the pressure in the cell (as read by the meter) has dropped to the minimum on the scale, close the “pump valve” and turn off the pump. You are now ready for the actual experiment!

- (A) How many times does the beam traverse the cell in the interferometer? Compute the total optical path length difference for the HeNe light at 632.8 nm with and without air in a vacuum cell 10 cm in length. What is the expected number of fringes that will pass as air is leaked into a cell of this length initially under a perfect vacuum?

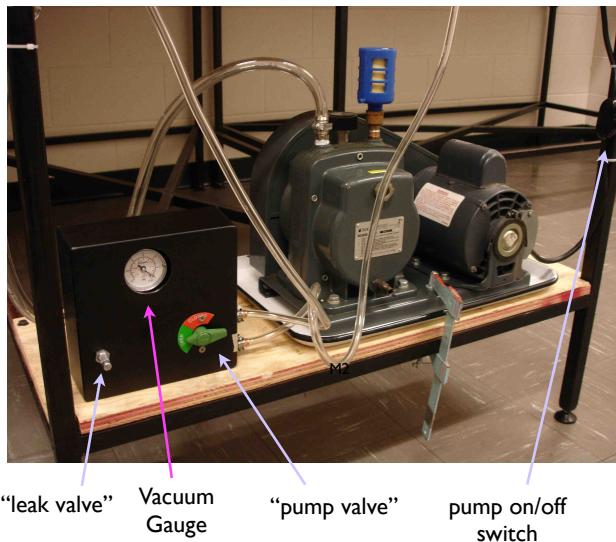


Figure 2.4: Vacuum pump, valves, and vacuum gauge for the vacuum cell.
NOTE: the gauge reads inches of mercury below atmospheric pressure, so the better the vacuum the higher the number!

- (B) Start by taking a single image and use it for defining the region of interest for the following scans. Now start the data acquisition and slowly open the “leak valve” to let in air while you record the fringes. You may need some practice introducing the air fast enough that the pressure rises to atmospheric pressure during the time you have set while not rising too quickly so that you miss the passage of fringes due to the finite sampling time of the video capture. At the end, process the video file and plot the measured fringe brightness in the region of interest as a function of time.
Hint: the time should be extracted from the known frame rate and frame number.
- (C) Would you have seen the same number of fringes if you had used the Na lamp instead of the HeNe laser for this measurement? Why or why not?
- (D) What is your value for the index of refraction of air at $\lambda = 632.8 \text{ nm}$? Does this agree with the “accepted” value? What are your sources of error?
 - (i) Could you use this apparatus as a barometer or thermometer? If yes, how? If not, why not?

2.3.3 Calibrating the interferometer

In many applications of the interferometer, one scans the path length difference between the two arms. In the next set of experiments, you will be doing this by moving the translation stage over a set distance. However, the stage may not move exactly the distance, or at the speed, you expected it to move (due to

mechanical backlashes and inaccuracies of its built-in calibration). It is therefore critical for many interferometric scans to precisely calibrate the moving translation stage, i.e. to find a numerical “calibration factor” that converts the set distance into the actual distance traveled. Keep using the HeNe laser in this experiment.

To do this, you will again write your own Matlab script, which will control both the Thorlabs translation stage and the Raspberry Pi camera. As before, take a look at the two scripts `thorlabsStage.m` and `raspiCamera.m` in the **Software** folder, and then put together your own program using the example commands from those scripts. At the end, your code should start scanning the stage and, while it is moving, record a movie with the camera. Similarly to the previous experiment on the refractive index of air, you will then extract the brightness level in the small region of interest, but this time – as a function of the stage position. **NOTE:** When controlling the Thorlabs stage, it is perfectly normal to get a couple of warnings from MATLAB saying ”Warning: ACTX-CONTROL will be removed in a future release”. Please do NOT try to disable that warning or alter ANY files belonging to the MATLAB software!

When you capture data from the camera, the program records the intensity of a point on the CCD at a rate of 30 frames per second. This means that the distance the stage will have moved per frame is dependent on the speed you set. For example, for a speed of $1 \mu\text{m}/\text{s}$ the stage should (ideally) move

$$\frac{1 \mu\text{m}/\text{s}}{30 \text{ frames}/\text{s}} = \frac{33 \text{ nm}}{\text{frame}}. \quad (2.2)$$

- (A) What distance does the mirror need to move such that one fringe passes a reference point on the CCD camera?
- (B) Calibrate the interferometer at three different speeds: $1,3$ and $5 \mu\text{m}/\text{s}$. Repeat your calibration a few times for each speed setting to determine your calibration accuracy.

2.3.4 Average wavelength of the sodium lamp

In the previous section, you used the known wavelength of the light source (HeNe) to determine the exact displacement of the mirror (and thus, to calibrate the translation stage). Here, you will do the opposite: using the calibrated mirror displacement, you will measure the wavelength of light from a different source – the sodium lamp. Excited sodium atoms primarily emit light at two wavelengths, λ_1 and λ_2 , also known as the D1 and D2 lines. The separation between the D lines ($\Delta\lambda$) is relatively small, i.e. $\Delta\lambda \ll \lambda_1, \lambda_2$. Here, you will first neglect this splitting and measure the average wavelength $\bar{\lambda} \approx \lambda_1 \approx \lambda_2$ of the sodium source.

Lift the vacuum cell, removing it from the light path. Install the sodium lamp into an already existing post in front of the HeNe laser. **NOTE:** DO NOT MOVE THE HeNe LASER! As a general rule, please keep the lens, ground glass plate, beam diffuser, and white light filter in the mounts provided. Do not swap mounts or leave components lying on the table. If you need to remove an optical element, unscrew the clamp holding the mount instead of pulling the optic or the post from the mount. Remove the beam expanding lens and add a plastic light diffuser (not the ground glass!) close to the entrance face of the

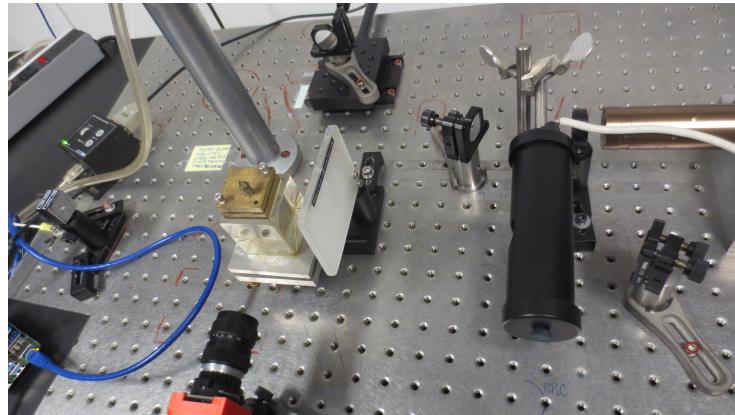


Figure 2.5: Sodium lamp in front of the HeNe laser (which remains in its original post), and the plastic diffuser in front of the beamsplitter cube.

beamsplitter cube (see Figure 2.5). Caution should be given not to put the diffuser too close to the lamp as this may melt the plastic (ask your TA to show you how it's done). Note that the lamp will take a few minutes to warm up. The large ‘folding’ mirror should also be removed and the Raspberry Pi camera should be moved closer to the beamsplitter cube so that it is facing the exit face of the cube. Leave the objective lens on the camera and re-adjust its focus and aperture for optimal fringe viewing (when fringes are found).

Move mirror M_2 to the approximate middle point of the range of Thorlabs translation stage. To do that, open up the `APTUser` program on the desktop. Using the *Settings* button on the GUI make sure the max velocity is 0.5 mm/s for both the move and jog. Next, push the *Home/Zero* button to zero the stage. The stage will move to the end of its travel range and the computer will note this as the ‘home’ or zero position. Position the stage around 10 to 12 mm. If the interferometer was properly aligned with the HeNe source in the previous steps (i.e. the two laser spots were nicely overlapped on the output face of the beamsplitter cube), fringes should now appear on the camera (if not, go back to the HeNe source and re-align). You might also have coincidentally placed your stage at one of the low contrast positions – so if you are sure about the proper beam alignment, jog M_2 in either direction and check for fringes again. You will need to make sure the CCD camera is not saturated by adjusting the aperture on the lens assembly. As with the HeNe source, fringes of different shapes may be produced by appropriate adjustment of mirrors M_1 and M_2 . You are now ready to measure the wavelength of the sodium lamp.

- (A) What speed and span would be reasonable to use to measure the wavelength and why?
- (B) Scan the path length difference in the same way you did it while calibrating the translation stage, while recording the fringe intensity at one location on the CCD camera. Count the fringes (as you did in the calibration task) and calculate the average wavelength of the two sodium lines.

- (i) Process your data using a complementary (and very powerful) method of “Fourier Transform (FT) spectroscopy”. To execute the latter, make sure to read Appendix C, including the sub-section on the numerical “FFT analysis with MATLAB”. Carry out your own Fourier transform of the collected data, and plot the calculated light spectrum ($G(k)$) in Appendix C). Find the average wavelength and compare with your findings in task (B) above.
- (ii) In comparison to the more “primitive” method of fringe counting, what extra information is provided by the method of FT spectroscopy?

2.3.5 Line splitting of the sodium lamp

As mentioned in the previous section, the emission peak from sodium atoms is split into two lines with wavelengths λ_1 and λ_2 , known as D1 and D2 lines. This “fine structure” splitting occurs due to the electronic spin-orbit coupling. Each spectral line generates its own fringe pattern, and the two patterns overlap with one another. Consider a position of the translation stage, for which the bright fringes from the D1 line overlap with the bright D2 fringes. Due to the very small wavelength difference, the dark D1 and D2 fringes will also overlap with one another, and the resulting interference pattern will exhibit high bright-to-dark contrast (as if from a single-wavelength monochromatic source) – the fringes are said to have “high visibility”. Now imagine changing the arm length d of the interferometer in such a way as to satisfy:

$$2d_{\text{high} \rightarrow \text{low}} = (N + 1/2)\lambda_1 = N\lambda_2 \quad (\lambda_1 < \lambda_2), \quad (2.3)$$

where N is an integer. Since we moved a *half-integer* number of λ_1 , the D1 line is now producing a dark fringe (the two arms are interfering destructively), whereas the D2 line is still interfering constructively (moved by an *integer* number of λ_2). Now the *dark* fringes from the D1 line will overlap with the *bright* D2 fringes, and vice versa, leading to a much lower contrast of the overall interference pattern – the fringes are said to attain “low visibility”. Hence, the end result of the line splitting is the modulation of the fringe visibility, which becomes apparent if the scan is extended to much longer length scales than what you used in the previous section, i.e. well beyond a few tens of microns. An example of such modulation is shown in Figure 2.6.

One can extract the wavelength difference $\Delta\lambda$ from the modulated fringe pattern using Equation 2.3:

$$\Delta\lambda \equiv \lambda_2 - \lambda_1 = 2d_{\text{high} \rightarrow \text{low}} \left(\frac{1}{N} - \frac{1}{N + 1/2} \right) \approx \frac{d_{\text{high} \rightarrow \text{low}}}{N^2} \approx \frac{\bar{\lambda}^2}{4d_{\text{high} \rightarrow \text{low}}}, \quad (2.4)$$

where $\bar{\lambda} \approx \lambda_1 \approx \lambda_2$ is the average wavelength found in the previous section, and we used the fact that $N \gg 1$ (convince yourself that this assumption is correct).

- (A) Notice that in this task, you are seeking to measure the modulation period of the *fringe contrast* rather than the *fringe intensity*, which you recorded in your previous measurement. How would you calculate the fringe contrast from the recorded CCD images? What speed and travel range of the translation stage would it be reasonable to use in this case? *Hint,*

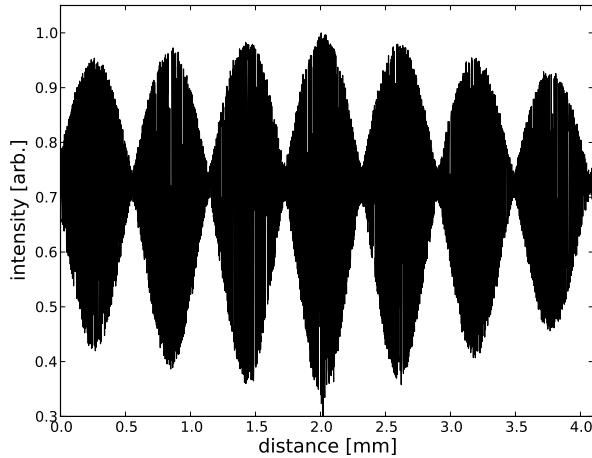


Figure 2.6: Fringe pattern for the Na lamp over a relatively large distance. The beat pattern is the result of the two different wavelength components in the emitted light. If you zoomed in on this pattern, you would see individual fringes appearing periodically as the motor moves. Note that it took longer than 45 minutes to record this trace, which is *not* what we recommend you to do in this section of the lab.

using the same speed of the translation stage as you used in collecting the fringe intensity data but over a much longer travel range may result in a painfully long scan (e.g. it took longer than 45 minutes to collect the data in Fig. 2.6!). How can you avoid this?

- (B) Modify your MATLAB script to record the contrast of the fringe pattern instead of the intensity of a single fringe. Execute the scan and process the data by counting the number of modulation periods. From that, determine the splitting of the two sodium D lines. Specify the uncertainty of your result and compare it with the known values.
 - (i) As in the experiment on the average wavelength, perform Fourier transform of the observed beat signal. Plot the calculated spectrum and use it to determine the D1–D2 line splitting. Compare with your result in task (B) above.

2.3.6 Coherence length of white light

Coherence length is the distance over which a wave maintains a well-defined phase (that is, the phase does not change in a random fashion). Imagine taking a carbon copy of a wave (any wave) and translating it with respect to the original by some distance. The distance over which you can translate the copied wave and have it look the same as the original one is the coherence length. For a perfect sinusoid, this length is infinite. However, for the electric field from a real light source, it will always be finite owing to multiple reasons, whose analysis is well beyond the scope of this study. The coherence length of a simple HeNe

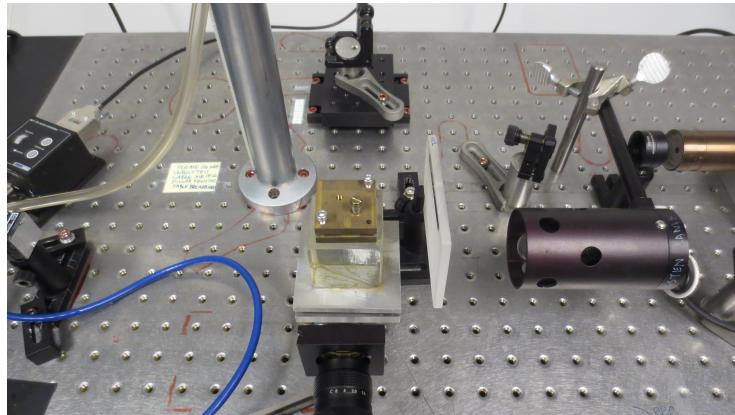


Figure 2.7: White light source instead of the sodium lamp.

laser (multi-mode, as in this lab) is on the order of 20 cm. For each D line of a sodium lamp, it is only slightly shorter around 7 cm. This is exactly the reason for the interference fringes, which you observed so far, to persist during the scan of the translation stage on the scale of a few mm.

The situation is drastically different for a white light source, whose coherence length is only about $3 \mu\text{m}$. In this section, you will measure the coherence length of white light by measuring the distance, over which the translation stage can move without losing the interference fringe pattern.

- (A) Since the white-light fringes are expected to appear when the path lengths of the two arms of the interferometer differ by less than the very short white-light coherence length, i.e. almost identical, the first task is to find the position of the translation stage corresponding to the “zero path length” difference (ZPL).

To find the ZPL, use your setup with the sodium lamp. Do your best to align the interferometer to observe circular fringes of high-contrast. *Hint: if needed, go back to the HeNe source; then continue with the sodium lamp.* Now run your scan as you did in point (B) of the previous section and record the fringe contrast as a function of distance. If you completed this scan last week, do not rely on that data since the ZPL sometimes changes slightly from week to week (e.g. if your fellow students abuse the translation stage position controller or if the table gets bumped). The ZPL is the position of the mirror corresponding to the strongest fringe contrast. Find that position. You may end up with two (or maybe even three) potential candidate positions for ZPL. Scan around each of those positions, while carefully watching the fringe pattern on the camera web interface. The sign that you are near the zero path length difference is that the radius of curvature of the circular fringes is the largest. Why is that?

- (B) Now replace the sodium lamp with the white light source, as shown in Figure 2.7. Note that the sodium lamp will be hot, so grab it only by the post. At first, you probably won’t see any fringes, despite being close to

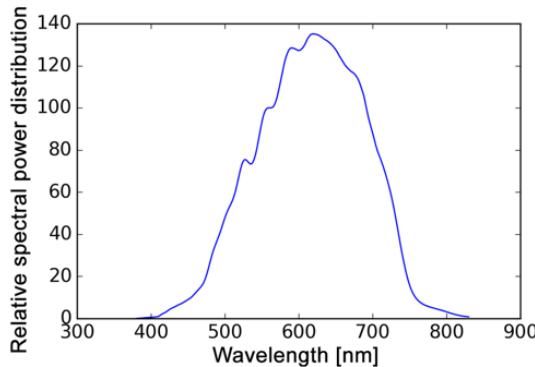


Figure 2.8: Example spectrum from a tungsten halogen lamp [Retzlaff, M.-G. et al. Journal of Sensors and Sensor Systems. 6. 171-184 (2017)], similar to the one used in this lab.

the ZPL. Don't get disappointed – there is one more trick to find them, based on your work in part (A). Place a narrow band filter either in front of the camera or after the white light lamp. Putting the filter in front of the camera is better because the non filtered scattered white light will not pollute your images. If you do put the filter right after the white light source, please keep it at least a few cm away from the bulb. Similarly, the beam diffuser should not be placed too close to the white light source or it will melt!

Slowly (at a speed of $10 \mu\text{m/s}$) scan (using the APT GUI) the region around what you believe is the ZPL position until you see the fringes appear on the camera web interface (again, you might want to check a few candidates for the ZPL). Be ready to quickly press the stop button on the APT user interface when you see the fringes appear. With the filter in place, the range over which fringes should be visible is about 50 microns. If you are spending more than 15 minutes without finding the fringes, ask a TA to check that you are doing everything correctly.

Record the fringe intensity as a function of distance. Explain why does the filter help find the fringes. *Hint: when you find the fringes, write down the stage position for future reference and note the direction you approached the zero-path length difference. The latter is important for mechanical backlash in the micrometer screw. Since you will be using a MATLAB script to take data, make sure that your code moves the stage forward (i.e. to higher values on the indexed stage) always approaching your position of zero-path length difference from below. When you repeat your scan, first move backwards beyond the initial starting point, then forward again, so as to reach the starting point also from below, thus avoiding the backlash.*

- (C) Finally, remove the filter and find the interference fringes from the white light source by scanning the translation stage in the same narrow travel range, where you just saw the fringes with the filter installed. This time, the fringes will appear and disappear within $2 \mu\text{m}$, so you will probably have to adjust the max velocity and step distance of the stage: don't go

too fast or you will miss the fringes. Also, use the actuator toggle with care since rapid changes in motion of the stage might also misalign the interferometer. From the observed scan, determine the coherence length of unfiltered white light. Does it agree with your expectations? How does it compare with the previous scan, when the filter was on? Explain.

- (i) After recording the white-light fringe pattern, you can once again use the powerful method of Fourier Transform spectroscopy to compute the spectrum of the white light source used in this lab (both with and without the filter). The latter is a standard tungsten-halogen lamp, whose typical spectrum is shown in Fig. 2.8. Keep in mind that the measured spectrum is ultimately limited by the wavelength dependent transmission of the optical elements in the Michelson interferometer and by the sensitivity of the CCD detector.

Appendix A

Raspberry Pi Camera Operation

For several experiments in this course you will use the Raspberry Pi CCD cameras to capture images or movies. While you will be taking your final data using Matlab (or Python) scripts (see an example in the `Software` directory), it is often convenient to look at the camera image in real time, e.g. when you search for various cavity modes or optimize the interference pattern. For a real time view of the raspberry pi camera it is useful to use the Raspberry Pi Camera Web Interface. This interface simply needs a web browser and to see the interface type in the web url: <http://142.103.238.21/html/> where the IP address is different for different desktop stations and can be found on the corresponding monitor.

If you want to view the camera on your laptop, then you need to be connected to UBC VPN. The web interface has many configurable options but for us there are only a few to be concerned with. When you first open the page, it will look similar to the one shown in Fig. A.1.

On this page the items of interest are:

1. The stop camera/ start camera button. You must press the stop camera button to free up the camera when you want to take images with MATLAB instead. Press the start camera button when you want to use the web interface again.
2. Record image or record video start. These will record an image or video saved on the raspberry pi.
3. Download Videos and Images will allow you to see images or videos you have taken and you can download them to the computer you are using the web interface on. You can also use this to delete picture files saved on the raspberry pi. **Please clean up after yourself as the RasPi memory is limited!**
4. The Camera Settings menu provides many options which we will refer to later in this document.
5. The System menu has options. **Please do not touch any of those except the Reset Settings button!** The reset settings button can

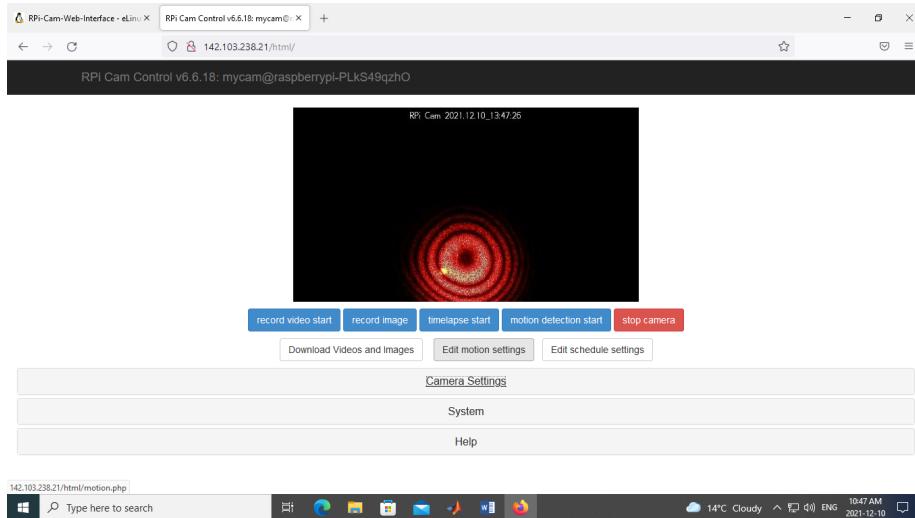


Figure A.1: An example of the web interface for operating Raspberry Pi cameras in real time.

help if the web interface is left in an odd state or if you want to return to the default settings. The web interface saves the last used settings even if the web browser is closed.

Please note that more than one person can have a web interface open at the same time to see the camera but it is not advisable to be tweaking settings on different computers at the same time.

Camera Settings Menu. There are many many options in the camera settings. You do not need to and should not play with most of them. The main settings of use are:

1. Exposure mode to ‘off’ or ‘auto’. ‘Off’ allows you to set the shutter speed yourself.
2. Shutter speed (in micro seconds) can be set when the exposure mode is set to off. Note that a shutter speed of 0 is auto exposure regardless of the exposure mode setting.
3. White Balance to ‘off’ or ‘auto’. This is useful when you want to correctly collect the red pixel values for data analysis. For the most part we are using HeNe lasers whose wavelength is red (633 nm) so it makes sense to set the white balance to ‘off’, for example, for taking cross sections of images where the red pixel value only is needed.
4. Other settings such as ISO may be useful. Sharpness, contrast, brightness, saturation can be played with but typically the shutter speed is the most valuable.
5. Image quality changes the amount of image compression and is mildly useful.

6. If you want truly uncompressed data for analysis, for example, for cross section analysis then the raw layer can be set to ‘On’. The raw Bayer information is appended to the end of the jpeg image file and must be extracted. This option makes the file size large so only use if needed for analysis.
7. Annotation and size can be changed if you want to have a descriptor written on your image.

We do not recommend altering other settings unless you have a very good reason to. If you find that the camera view is strange, first try resetting the settings. If buttons are not working at all or the preview is updating only sporadically, the web interface will need to be reinstalled on the raspberry pi. **Please ask for assistance if that is the case and do not do that on your own.**

Resources for more information: <https://elinux.org/RPi-Cam-Web-Interface>, which is based on the picamera module url<https://picamera.readthedocs.io/en/release-1.13/>.

Raspberry Pi High Q camera image resolution and effective pixel size. It is important to realize that the effective pixel size of the RasPi camera changes depending on the image resolution specified. The specifications for the Raspberry Pi High Q camera are given as:

Sony IMX477R stacked, back-illuminated sensor, 12.3 megapixels, 7.9 mm sensor diagonal, $1.55 \mu\text{m} \times 1.55 \mu\text{m}$ pixel size.

The maximum resolution of this camera is 4056×3040 pixels. Just to make sense of these numbers it means that there are more pixels along one direction than another which you can tell from looking at the rectangular shape of the CCD chip. If we multiply (4056×3040) we get 12330240 total pixels which agrees with the 12.3 megapixel specification. Each physical pixel is $1.55 \mu\text{m} \times 1.55 \mu\text{m}$ in size which gives us an approximate 7.9 mm diagonal, as shown in Fig. A.2.

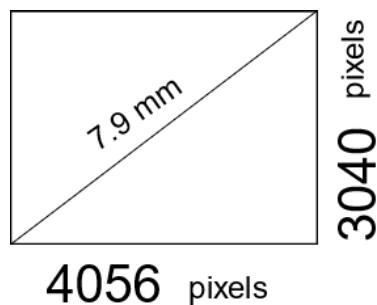


Figure A.2: The CCD chip of the raspberry pi high Q camera has 4056×3040 pixels with a 7.9 mm diagonal dimension.

Note that the image resolution can be specified by the user. For example, the web interface we use to collect raspberry pi images has a default image resolution of (2592×1944) . This means that images collected have 2592×1944 elements

so that the effective pixel size has increased in comparison to the $1.55 \mu\text{m} \times 1.55 \mu\text{m}$ size of the physical pixel elements. In that case the effective pixel size is instead $2.42 \mu\text{m} \times 2.42 \mu\text{m}$ as calculated by:

$$\begin{aligned}(4056/2592) \times 1.55 \mu\text{m} &= 2.42 \mu\text{m}, \\ (3040/1944) \times 1.55 \mu\text{m} &= 2.42 \mu\text{m}.\end{aligned}$$

This is important for the HeNe lab where you are asked to measure the beam diameter with a camera. In that case it is important to know the separation distance between the elements in your image array. You need to take into account your image resolution used when saving your images to calculate the correct beam diameter.

Appendix B

Thorlabs Advanced Positioning Technology (APT) Software

In Cavity and Michelson labs, we have translation stages whose position can be controlled by software. The motorized actuators which change their length are either Z825B or Z625B from Thorlabs. The controllers that are used to interface from a computer to the actuators are either the TDC001 or the OptoDC Driver (ODC001) from Thorlabs. These controllers have a USB connection to the lab computer that controls it.

There are two ways to control the motor length, and therefore translation stage position. First, there is a software GUI provided by the Thorlabs company called the APT (for Advanced Positioning Technology) User software. The second means of communications is by programming using, for example, MATLAB to send custom Active X Control commands to do things such as set the desired speed or position of the motor actuator. This Appendix will explain how to use the APT user software and will briefly discuss the programming through MATLAB.

APT GUI

To open the APT user software click on the APT User icon on the desktop or go to the Start Menu and to the Thorlabs option and click on APT user. You should see a GUI show up as shown in Figure B.1 below.

On this GUI the position of the motor is shown with a value between 0 and 25 mm. Pressing the Home button will make the motor go to its home position which is 0 mm. If you click inside the box where the position is shown, in our case 11.0000 mm, then a box will appear where you can type in the position you would like the motor to go to. If, while you are moving, you want the motor to stop - press the Stop button.

There are two ways the motor can move, either by being told as mentioned above to go to a certain position, or by performing a ‘jog’. A jog means the motor moves by a certain pre-set amount from its current position. The jog button up and down arrow will increase or decrease the motor position by an



Figure B.1: The APT User Software GUI.

amount set by the ‘Jog Step Distance’. That Step Distance can be set by clicking on the Settings tab on the bottom right and a window will appear as shown in Figure B.2.

The most useful setting in the Motor Driver Settings window is the ‘Max Vel’ input box at the top left under ‘Moves - Velocity Profile’. This is the speed at which the motor will move when you input a specific position to go to as described earlier by clicking in the position indicator box and inputting the desired position. The other settings in this Motor Driver Settings window are of less importance and should not necessarily be played with. In addition, the ‘Stage/Axis’ and ‘Advanced’ tabs are not recommended to be altered. To re-iterate, the main usage of this Motor Driver Settings window is to set the maximum velocity the motor moves at when a new position is asked of it.

Troubleshooting of the APT User GUI: If the APT user software stops working it may be needed to unplug the power supply of the controller from the wall, wait for about five seconds and then plug it back in. PLEASE DO NOT UNPLUG THE POWER OF THE CONTROLLER BY DISCONNECTING THE POWER CONNECTOR PLUGGED INTO THE CONTROLLER FROM THE CONTROLLER. The controller can be damaged by doing that. Leave the power connector plugged into the controller and unplug the power supply from the wall instead.

When the controller has been power cycled, it will read a zero position when the APT user software is reopened. This is an incorrect reading since the motor position is still at wherever it was when the power was turned off. To fix this press the Home button on the APT User GUI and watch the position just before it goes to home and resets itself to zero. It may say something like negative 11.5 mm which means that it had to travel back approximately 11.5 mm to get home. Then, to get back to the approximate position where you started, type in for example 11.5 mm into the position window and it will go back to the last position before the power was cycled. If all of this confuses you, please ask one

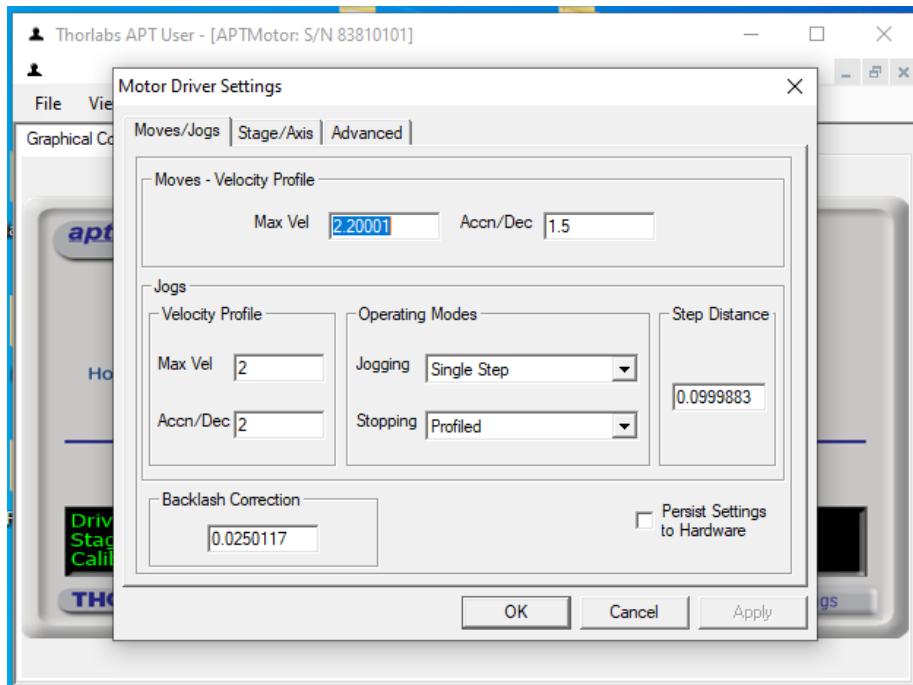


Figure B.2: The Motor Driver Settings window in which the velocity at which the motor moves to a new position can be set as well as other settings.

of the instructors for help before you destroy something!

Note: Please, properly close the APT User software when you are finished using the computer and properly sign out of the computer. This will help the next user to not have troubles due to existing unclosed connections.

MATLAB control of the motor

The second way to move the motor is by sending commands to the controller using MATLAB. The possible commands are many and complex and are fully documented in the `thorlabsStage_ProgrammingManual.pdf` posted in the `Hardware/Thorlabs Translation Stage` folder. The commands that you will need are given in the `thorlabsStage.m` MATLAB script as an example. This script gives example of how to initiate a connection to the controller, how to set the speed and position of the motor actuator, and finally how to cleanly close the connection to the controller and the motor. In the optical cavity lab, you are asked to combine commands from the `thorlabsStage.m` script with commands that record data from an oscilloscope in order to perform the “knife edge” scan. In the Michelson interferometer lab, you are asked to combine commands from the `thorlabsStage.m` script with commands found in `raspiCamera.m` script to collect data from the Raspberry Pi camera at the same time as the motor is being controlled.

Note 1: If you run the `thorlabsStage.m` script (or any script that you create) that controls the motor, please first close any APT User GUI window

that you have open. If you desire to have a Figure B.1 window appear as When you control the motor by command in MATLAB, a window looking similar to Figure B.1 will appear as a MATLAB figure. Please be aware that it is NOT the APT User GUI window, so please do not click in it to control the motor. That window shows up just so you can see what is happening with the motor position as your script is running and that window should be closed when you are done running your script.

Note 2: Please, be aware that the serial number found on the controller must be correctly input by hand into either the `thorlabsStage.m` script or any script you write yourself. In the `thorlabsStage.m` script, this is the line that looks like this:

```
% Motor serial number (specific to each station)  
>> motorSN=83810101;
```

Appendix C

Fourier Transform Spectroscopy

So far you have investigated the behavior of the interference fringe pattern as a function of the optical path difference for different optical sources. In general the fringe pattern intensity versus the optical path difference (or equivalently versus the time delay between the two interfering beams) is related to the *power spectrum* of the light entering the amplitude division interferometer by a Fourier transform. Therefore one can easily measure the optical power spectrum using the measurement of the intensity variation as a function of stage position.

Let's see how this works¹. The electric field incident on the camera is the sum of the electric fields \mathbf{E}_1 and \mathbf{E}_2 arriving from mirrors M1 and M2 respectively after having passed through the beamsplitter. Assuming for the moment that the input source is a purely *monochromatic* wave, we can characterize these electric fields with a vector amplitude (encoding the strength and polarization of the wave) and a spatial and time dependent phase:

$$\mathbf{E}_1 = \mathbf{A}_1 e^{i(\mathbf{k}_1 \cdot \mathbf{r} - \omega t + \phi_1)} \quad (\text{C.1})$$

$$\mathbf{E}_2 = \mathbf{A}_2 e^{i(\mathbf{k}_2 \cdot \mathbf{r} - \omega t + \phi_2)}. \quad (\text{C.2})$$

The phase includes terms ($\phi_1 = |\mathbf{k}_1| l_1$ and $\phi_2 = |\mathbf{k}_2| l_2$) which depend on the optical path lengths from the beamsplitter along path l_1 (encountering M1) and path l_2 (encountering M2). The intensity on the camera is simply the square of the total electric field:

$$I = |\mathbf{E}|^2 = \mathbf{E} \cdot \mathbf{E}^* = (\mathbf{E}_1 + \mathbf{E}_2) \cdot (\mathbf{E}_1^* + \mathbf{E}_2^*) \quad (\text{C.3})$$

$$= |\mathbf{A}_1|^2 + |\mathbf{A}_2|^2 + 2\mathbf{A}_1 \cdot \mathbf{A}_2 \cos \theta \quad (\text{C.4})$$

$$= I_1 + I_2 + 2\sqrt{I_1 I_2} \cos \theta \quad (\text{C.5})$$

where we have assumed in the last step that the waves have the same polarization and that $\theta = (\mathbf{k}_1 - \mathbf{k}_2) \cdot \mathbf{r} + \phi_1 - \phi_2$. You might be wondering about a white light source which is probably completely unpolarized. Is this assumption still valid in that case? This assumption is justified since the two fields \mathbf{E}_1 and \mathbf{E}_2 are copies of the incident field generated by the beam splitter. As long as the

¹This derivation follows that in "Introduction to Modern Optics" by Grant R. Fowles

beam splitter and the optics which follow preserve the polarization of the light, these two fields will arrive at the detector with exactly the same polarization (whatever it was at the input of the interferometer).

If the interferometer is well aligned so that the waves are co-linear ($\mathbf{k}_1 = \mathbf{k}_2$) and the 50/50 beamsplitter generates two waves of the same intensity ($I_1 = I_2 = I/2$), then the interference pattern is simply

$$I(x) = I(1 + \cos kx) \quad (\text{C.6})$$

where $x = l_1 - l_2$ is the path length difference and $k = |\mathbf{k}_1| = 2\pi/\lambda$ is the magnitude of the wavevector.

Okay, now in general, the light illuminating the interferometer will be composed of *many* different frequencies. So, let's model the power spectrum of the incident light as a continuous distribution, $G(\omega)$. Then the total optical power emitted into the interferometer in the frequency range from ω_0 to $\omega_0 + d\omega$ (where $d\omega$ is an infinitesimally small frequency band) is simply $G(\omega_0)d\omega$. For convenience, we can instead use the distribution over wavelength $G(k)$ (where $\omega = ck$). Then the optical power, P , hitting the detector from illuminating the interferometer with this polychromatic source will be the sum of the interference patterns from each monochromatic wave composing $G(k)$. Assuming the interferometer is ideal and has no losses, we have then:

$$P(x) = \int_0^\infty G(k) (1 + \cos kx) dk \quad (\text{C.7})$$

$$= \int_0^\infty G(k) dk + \int_0^\infty G(k) \frac{e^{ikx} + e^{-ikx}}{2} dk \quad (\text{C.8})$$

$$= \frac{1}{2}P(0) + \frac{1}{2} \int_{-\infty}^\infty G(k)e^{ikx} dk \quad (\text{C.9})$$

where $P(0)$ is simply the power measured at zero path length difference. We can rearrange this expression and define the power function $W(x)$

$$W(x) = 2P(x) - P(0) = \int_{-\infty}^\infty G(k)e^{ikx} dk. \quad (\text{C.10})$$

We see then that the power function $W(x)$ is the *inverse* Fourier transform of the power spectral density $G(k)$. Inverting this expression we have that the power spectrum is the Fourier transform of the power function

$$G(k) = \frac{1}{2\pi} \int_{-\infty}^\infty W(x)e^{-ikx} dx, \quad (\text{C.11})$$

or equivalently

$$G(\nu) = \frac{1}{2\pi} \int_{-\infty}^\infty W(x)e^{-i2\pi\nu x/c} dx. \quad (\text{C.12})$$

FFT analysis with MATLAB

In order to compute the power spectra from your fringe pattern data, you will need to do some data processing. First, you will need to properly truncate your data because it may include points when the stage was moving backwards

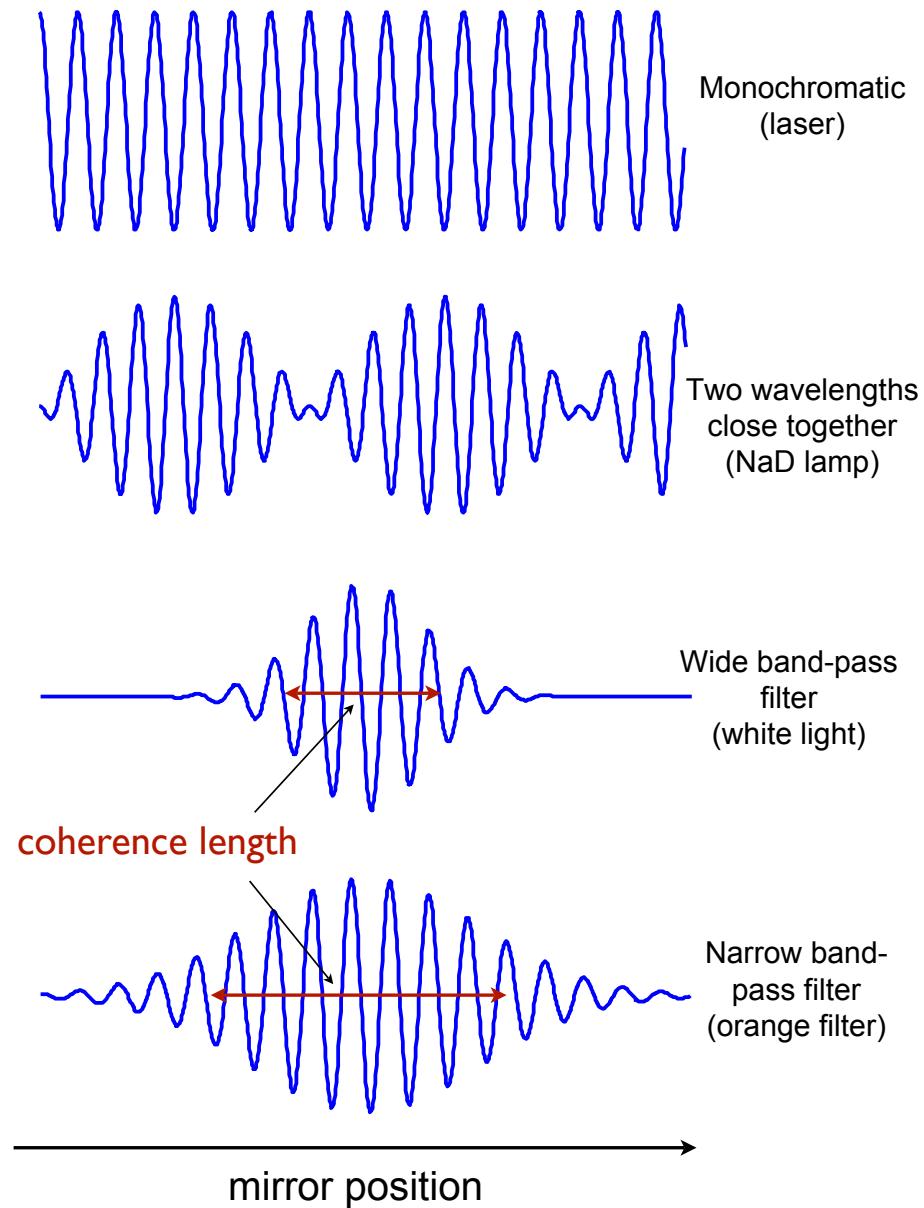


Figure C.1: Fringe patterns plotted as a function of the mirror position. Notice how the fringe visibility collapses and then revives as a function of the mirror position when the spectrum is discrete. Also, note that if the mirror is moved a distance d , the optical path length between the two arms in the interferometer change by $2d$.

and then sitting stationary before and after the slow sweep through the ZPL position.

Assuming that your calibrated relative position of the Thorlabs stage is stored in vector \mathbf{x} , and the retrieved fringe brightness is stored in vector \mathbf{Yx} ,

start by plotting the raw data:

```
>> plot(x,Yx);
```

Find the indices *i1* and *i2*, between which the recorded brightness data look reasonable and truncate both vectors:

```
>>x=x(i1:i2);
>>Yx=Yx(i1:i2);
>>plot(x,Yx);
```

Condition your data by subtracting the mean value. Doing so will remove the (usually dominating, but not very useful) dc component from the Fourier Transform:

```
>>Yx0=Yx-mean(Yx);
```

Now take the Fourier Transform using the built-in **fft** function:

```
>>Yk0=fft(Yx0);
>>Yk=fftshift(Yk0);
```

The second line above re-arranges the Fourier spectrum from the lowest negative to the highest positive spatial frequency. To create the vector of spatial frequencies, use:

```
>>dx=x(2)-x(1);
>>k=linspace(-pi/dx,pi/dx,length(x));
```

Finally, plot the Fourier power spectrum, using the absolute values of the calculated vector *Yk* (remember that FT returns complex values):

```
>>plot(k,abs(Yk));
```

Finding the main peak on the plotted Fourier spectrum will now allow you to determine the (spatial) period in the recorded data, and from it – the wavelength of light. In doing these final calculations, remember that *dx* in the example above is the distance that the stage moved between the two consecutive images, so the amount of the total optical path length varied between consecutive frames is twice that distance!