

Lab 4: Physical Optics II

Daniel Geisz & Heidi Hu

October 16, 2019

1 Lab 4 - Physical Optics II

1.0.1 Experiment B3: The Index of Refraction of Air

Note: The lab manual said we only need to include an explanation of what we did and what results we got, so this will be a much shorter lab report.

Objective:

- Find the index of refraction of air using a Michelson interferometer

Method and Procedure: To do this experiment, we first set up the Michelson interferometer as described and shown in Setup B of the lab manual. We had already shown in the previous experiment that the index of refraction of air is dependent on the air's pressure, and we noticed that as we increased the pressure of the air in the pressure chamber, the fringes of the interference pattern produced by the apparatus expanded from the center of the pattern as the pattern underwent fringe cycles. In order to determine the index of refraction of air n_{air} , we started with a pressure P_o in the pressure chamber, we increase the pressure by a total amount of ΔP and measured the number of fringe cycles Δm , and finally we used the solution to Prelab Question 2e which states:

$$n_{air} = 1 + \Delta m \frac{P_o}{\Delta P} \frac{\lambda_o}{2t}$$

Where λ_o is the wavelength of the laser used in this experiment, and t is the thickness of the pressure chamber. The corresponding equation for error propagation is given by:

$$\delta n_{air} = \Delta m \frac{P_o}{\Delta P} \frac{\lambda_o}{2t} \sqrt{\left(\frac{\delta P_o}{P_o}\right)^2 + \left(\frac{\delta \Delta P}{\Delta P}\right)^2 + \left(\frac{\delta t}{t}\right)^2}$$

Data & Analysis: We used a laser of wavelength $\lambda_o = 6.346 \cdot 10^{-5}$ cm.

Using a barometer app, we measured the air pressure of the room to be $P_o = 100.124$ kPa.

We measured the thickness of the vacuum chamber to be $t = 1.2 \pm 0.05$ cm.

The following table shows the data we took for Δm and ΔP . Here, “dm” represents Δm , “dp” represents ΔP and is measured in kPa, and “delta dp” is the uncertainty in our measurement of ΔP .

[9]: data

```
[9]:   dm  dp  delta dp
0    1  14         1
1    2  26         1
2    3  38         1
3    4  52         1
```

By putting this data through the equations in the methods and procedures section, we obtain the following data table. Here “n” represents n_{air} , and “dn” represents the uncertainty in that value. The other values are the same as in the previous table.

[12]: data2

```
[12]:   dm  dp  delta dp         n         dn
0    1  14         1  1.000272  0.000023
1    2  26         1  1.000293  0.000017
2    3  38         1  1.000301  0.000015
3    4  52         1  1.000293  0.000013
```

The following script calculates the average value of n_{air} .

```
[26]: print("Average n: ", data2.n.mean())
      print("Uncertainty of Mean: ", data2.dn.mean())
```

Average n: 1.0002899481013627

Uncertainty of Mean: 1.6859218039747714e-05

So we see that our data yields an average value of n_{air} of:

$$n_{air} = 1.000290 \pm 0.000017$$

If we choose to only use the data point with the lowest uncertainty, namely the last data point, then our data yields:

$$n_{air} = 1.000293 \pm 0.000013$$

Both these values are very similar, and a quick Google search informs us that n_{air} is about 1.0003 on average, so we see our results make sense.

1.0.2 Summary and Conclusions:

In this lab, we were exploring physical optics, namely how to use a Michelson Interferometer. Due to the power outage, we were actually only able to explore the interferometer. Using the interferometer, we were able to witness a circular diffraction pattern, and by counting fringe cycles, we were able to very accurately determine the index of refraction of the air in the lab.

1.0.3 Contributions:

We each did 50% of the work.

1.0.4 References

We used Python and the open source library Pandas to create the data tables shown above. The following scripts were used to create tables and perform necessary calculations for this report.

```
[3]: import pandas as pd
```

```
[5]: def n(m, p):  
    return 1 + m * (100.124/p) * (6.346e-5/2*1.2)  
  
def dn(n, p, dp):  
    return (n - 1) * ((dp/p)**2+(0.05/1.2)**2)**0.5
```

```
[11]: data = pd.DataFrame({'dm': [1, 2, 3, 4], "dp": [14, 26, 38, 52], 'delta dp': [1, 1, 1, 1]})  
data2 = data  
data2['n'] = n(data.dm, data.dp)  
data2['dn'] = dn(data2.n, data.dp, data['delta dp'])
```