Experiment 19 - The Index of Refraction of Gasses

Index of refraction for each configuration

Green laser in air

We calculate the mean number of fringe counts for the green laser in air, then use that value to calculate the index of refraction of the green laser (543.52 nm) at the room temperature and pressure. We then use this refractive index value to find the refractive index at standard temperature and pressure, making sure to propagate uncertainties throughout.

In[1]:=
```
SetDirectory[NotebookDirectory[]];
airGreen = Flatten[Import["air_green.csv"]];
```

In[3]:=
```
(* Define a function to calculate mean fringe
 count and standard error in fringe count, resp. *)
meanCount[data_] := {Mean[data], StandardDeviation[data]/Sqrt[Length[data]]}
```

In[4]:=
```
(* Calculate mean fringe count and standard error in fringe count, resp. *)
meanAirGreen = meanCount[airGreen]
```

Out[4]= {181.873, 0.176888}

In[5]:=
```
(* Define a function to calculate n-
 1 at room conditions and corresponding error, resp. *)
lengthCell = 0.1867662 (* m *);
index[data_, λVac_] := λVac/(2 * lengthCell) meanCount[data]
```

In[7]:=
```
(* Calculate n-1 at room conditions and corresponding error, resp. *)
greenWavelength = 543.52 * 10^-9 (* nm *);
airGreenIndex = index[airGreen, greenWavelength]
```

Out[8]= $\{0.00026464, 2.57386 \times 10^{-7}\}$

We now use this refractive index value to find the refractive index at standard temperature and pressure, making sure to propagate uncertainties throughout. We keep in mind that $N \propto \frac{P}{T}$ and $(n - 1) \propto N$ as found in the pre-lab. (Note: We use as the room conditions the average of the temperature resp. pressure at the beginning and at the end of the experiment.)

In[9]:=
```
temp = Mean[{22.8, 23.8}] + 273.15 (* K *);
tempError = 0.5 (* Estimate of the error in temperature *);
pressure = Mean[{746, 744}] * 133.322 (* Pa *);
pressureError = 133.322 (* Estimate of the error in pressure *);
stdTemp = 273.15 (* K *);
stdPressure = 760 * 133.322 (* Pa *);
```

In[15]:=
```
(* Define a function to calculate n-
 1 at standard conditions using the index at room conditions *)
```
$$\text{stdIndex}[\text{data\_}, \lambda\text{Vac\_}] := \frac{\text{stdPressure} * \text{temp}}{\text{pressure} * \text{stdTemp}}\, \text{index}[\text{data}, \lambda\text{Vac}][[1]]$$
$$\left\{1, \text{Sqrt}\left[\left(\frac{\text{index}[\text{data}, \lambda\text{Vac}][[2]]}{\text{index}[\text{data}, \lambda\text{Vac}][[1]]}\right)^2 + \left(\frac{\text{tempError}}{\text{temp}}\right)^2 + \left(\frac{\text{pressureError}}{\text{pressure}}\right)^2\right]\right\}$$

In[16]:=
```
(* Calculate n-1 at standard conditions and corresponding error, resp. *)
stdAirGreenIndex = stdIndex[airGreen, greenWavelength]
```

Out[16]= $\{0.000292996, 6.92883 \times 10^{-7}\}$

Red laser in air

We carry out the same process as above for the red laser (632.99 nm) in air.

In[17]:=
```
airRed = Flatten[Import["air_red.csv"]];
```

In[18]:=
```
(* Calculate mean fringe count and standard error in fringe count, resp. *)
meanAirRed = meanCount[airRed]
```

Out[18]= $\{154.65, 0.0600925\}$

In[19]:=
```
(* Calculate n-1 at room conditions and corresponding error, resp. *)
redWavelength = 632.99 * 10^-9 (* nm *);
airRedIndex = index[airRed, redWavelength]
```

Out[20]= $\{0.000262071, 1.01833 \times 10^{-7}\}$

In[21]:=
```
(* Calculate n-1 at standard conditions and corresponding error, resp. *)
stdAirRedIndex = stdIndex[airRed, redWavelength]
```

Out[21]= $\{0.000290152, 6.35521 \times 10^{-7}\}$

Green laser in CO_2

We carry out the same process as above for the green laser in CO_2.

In[22]:=
```
co2Green = Flatten[Import["co2_green.csv"]];
```

In[23]:=
```
(* Calculate mean fringe count and standard error in fringe count, resp. *)
meanCo2Green = meanCount[co2Green]
```
Out[23]= {276.2, 0.159861}

In[24]:=
```
(* Calculate n-1 at room conditions and corresponding error, resp. *)
co2GreenIndex = index[co2Green, greenWavelength]
```
Out[24]= $\{0.000401893, 2.32611 \times 10^{-7}\}$

In[25]:=
```
(* Calculate n-1 at standard conditions and corresponding error, resp. *)
stdCo2GreenIndex = stdIndex[co2Green, greenWavelength]
```
Out[25]= $\{0.000444957, 9.93105 \times 10^{-7}\}$

Red laser in CO_2

We carry out the same process as above for the red laser in CO_2.

In[26]:=
```
co2Red = Flatten[Import["co2_red.csv"]];
```

In[27]:=
```
(* Calculate mean fringe count and standard error in fringe count, resp. *)
meanCo2Red = meanCount[co2Red]
```
Out[27]= {235.45, 0.15864}

In[28]:=
```
(* Calculate n-1 at room conditions and corresponding error, resp. *)
co2RedIndex = index[co2Red, redWavelength]
```
Out[28]= $\{0.000398995, 2.68832 \times 10^{-7}\}$

In[29]:=
```
(* Calculate n-1 at standard conditions and corresponding error, resp. *)
stdCo2RedIndex = stdIndex[co2Red, redWavelength]
```
Out[29]= $\{0.000441748, 9.97647 \times 10^{-7}\}$

Green laser in helium

We carry out the same process as above for the green laser in helium.

In[30]:=
```
heliumGreen = Flatten[Import["helium_green.csv"]];
```

In[31]:=
```
(* Calculate mean fringe count and standard error in fringe count, resp. *)
meanHeliumGreen = meanCount[heliumGreen]
```
Out[31]= {23.02, 0.0916515}

In[32]:=
```
(* Calculate n-1 at room conditions and corresponding error, resp. *)
heliumGreenIndex = index[heliumGreen, greenWavelength]
```
Out[32]= $\{0.000033496, 1.3336 \times 10^{-7}\}$

In[33]:=
```
(* Calculate n-1 at standard conditions and corresponding error, resp. *)
stdHeliumGreenIndex = stdIndex[heliumGreen, greenWavelength]
```
Out[33]= $\{0.0000370852, 1.67901\times 10^{-7}\}$

Red laser in helium

We carry out the same process as above for the red laser in CO_2.

In[34]:=
```
heliumRed = Flatten[Import["helium_red.csv"]];
```

In[35]:=
```
(* Calculate mean fringe count and standard error in fringe count, resp. *)
meanHeliumRed = meanCount[heliumRed]
```
Out[35]= {19.55, 0.0307318}

In[36]:=
```
(* Calculate n-1 at room conditions and corresponding error, resp. *)
heliumRedIndex = index[heliumRed, redWavelength]
```
Out[36]= $\{0.0000331295, 5.20783\times 10^{-8}\}$

In[37]:=
```
(* Calculate n-1 at standard conditions and corresponding error, resp. *)
stdHeliumRedIndex = stdIndex[heliumRed, redWavelength]
```
Out[37]= $\{0.0000366795, 9.78557\times 10^{-8}\}$

Summary of results

In[38]:=
```
summary = {{"Laser", "Gas", "(n-1) - Room Conditions", "σ(n-1) - Room Conditions",
    "(n-1) - Standard Conditions", "σ(n-1) - Standard Conditions"},
   {"Green", "Air", airGreenIndex[[1]], NumberForm[airGreenIndex[[2]], 3],
    stdAirGreenIndex[[1]], NumberForm[stdAirGreenIndex[[2]], 3]},
   {"Red", "Air", airRedIndex[[1]], NumberForm[airRedIndex[[2]], 3],
    stdAirRedIndex[[1]], NumberForm[stdAirRedIndex[[2]], 3]},
   {"Green", "CO2", co2GreenIndex[[1]], NumberForm[co2GreenIndex[[2]], 3],
    stdCo2GreenIndex[[1]], NumberForm[stdCo2GreenIndex[[2]], 3]},
   {"Red", "CO2", co2RedIndex[[1]], NumberForm[co2RedIndex[[2]], 3],
    stdCo2RedIndex[[1]], NumberForm[stdCo2RedIndex[[2]], 3]},
   {"Green", "Helium", heliumGreenIndex[[1]], NumberForm[heliumGreenIndex[[2]], 3],
    stdHeliumGreenIndex[[1]], NumberForm[stdHeliumGreenIndex[[2]], 3]},
   {"Red", "Helium", heliumRedIndex[[1]], NumberForm[heliumRedIndex[[2]], 3],
    stdHeliumRedIndex[[1]], NumberForm[stdHeliumRedIndex[[2]], 3]}};
```

In[69]:=
```
Grid[summary, Alignment → Left, Spacings → {1.5, 1.2},
  Frame → All, ItemStyle → Directive[FontSize → 12, "Text"],
  Background → {None, {Gray, LightGray, LightGray, None, None, LightGray, LightGray},
    {{2, 1} → LightGreen, {4, 1} → LightGreen, {6, 1} → LightGreen,
     {3, 1} → LightRed, {5, 1} → LightRed, {7, 1} → LightRed}}] // ScientificForm
```

Out[69]//ScientificForm=

Laser	Gas	(n–1) – Room Conditions	$\sigma_{(n-1)}$ – Room Conditions	(n–1) – Standard Conditions	$\sigma_{(n-1)}$ – Standard Conditions
Green	Air	2.6464×10^{-4}	2.57×10^{-7}	2.92996×10^{-4}	6.93×10^{-7}
Red	Air	2.62071×10^{-4}	1.02×10^{-7}	2.90152×10^{-4}	6.36×10^{-7}
Green	CO_2	4.01893×10^{-4}	2.33×10^{-7}	4.44957×10^{-4}	9.93×10^{-7}
Red	CO_2	3.98995×10^{-4}	2.69×10^{-7}	4.41748×10^{-4}	9.98×10^{-7}
Green	Helium	3.3496×10^{-5}	1.33×10^{-7}	3.70852×10^{-5}	1.68×10^{-7}
Red	Helium	3.31295×10^{-5}	5.21×10^{-8}	3.66795×10^{-5}	9.79×10^{-8}

Using the results to calculate n_e and λ_0 for each gas

We recall from problem 3 of the pre-lab that equation (19.9) describing the refractive index of this system has solutions as given by the following functions:

In[40]:=
```
kb = 1.38064852 * 10^-23 (* JK^-1 *);
nDensity = stdPressure / (kb * stdTemp);
re = 2.82 * 10^-15 (* m *);
a[index_] := (2 π)/(nDensity * re) index
c[λVac_] := 1/λVac^2
(* Define a function that evaluates the solutions to (19.9)
 to find λ0 and ne respectively, with respective uncertainties *)
soln[greenIndex_, redIndex_] := {{1/
    Sqrt[(a[greenIndex][[1]] c[greenWavelength] - a[redIndex][[1]] c[redWavelength]) /
      (a[greenIndex][[1]] - a[redIndex][[1]])],
   1/(2 Sqrt[
        (a[greenIndex][[1]] c[greenWavelength] - a[redIndex][[1]] c[redWavelength]) /
        (a[greenIndex][[1]] - a[redIndex][[1]])])
    * Sqrt[((a[greenIndex][[2]] c[greenWavelength])^2 +
        (a[redIndex][[2]] c[redWavelength])^2) /
       (a[greenIndex][[1]] c[greenWavelength] - a[redIndex][[1]] c[redWavelength])^2 +
      (a[greenIndex][[2]]^2 + a[redIndex][[2]]^2) /
       (a[greenIndex][[1]] - a[redIndex][[1]])^2]},
  {(a[greenIndex][[1]] a[redIndex][[1]] (c[greenWavelength] - c[redWavelength])) /
    (a[greenIndex][[1]] - a[redIndex][[1]]),
   (a[greenIndex][[1]] a[redIndex][[1]] (c[greenWavelength] - c[redWavelength])) /
     (a[greenIndex][[1]] - a[redIndex][[1]]) *
    Sqrt[(a[greenIndex][[2]] / a[greenIndex][[1]])^2 + (a[redIndex][[2]] / a[redIndex][[1]])^2 +
      (a[greenIndex][[2]]^2 + a[redIndex][[2]]^2) /
       (a[greenIndex][[1]] - a[redIndex][[1]])^2]}}
```

■ For air

In[46]:=
```
{λ0Air (* m *), neAir} = soln[stdAirGreenIndex, stdAirRedIndex]
```

Out[46]=
```
{{1.03082 × 10^-7, 1.70468 × 10^-8}, {2.20437, 0.72874}}
```

For CO_2

In[47]:=
```
{λ0Co2 (* m *), neCo2} = soln[stdCo2GreenIndex, stdCo2RedIndex]
```

Out[47]= $\{\{8.91589\times10^{-8}, 1.95596\times10^{-8}\}, \{4.51692, 1.98134\}\}$

For helium

In[48]:=
```
{λ0Helium (* m *), neHelium} = soln[stdHeliumGreenIndex, stdHeliumRedIndex]
```

Out[48]= $\{\{1.09248\times10^{-7}, 2.61844\times10^{-8}\}, \{0.247266, 0.118451\}\}$

In[75]:=
```
pl = 6.62607004 * 10^-34 (* Js *);
c = 299 792 458 (* ms^-1 *);
e0[λ0_] := (pl * c)/(λ0[[1]]) * 6.242 * 10^18 {1, λ0[[2]]/λ0[[1]]} (* eV *)
summaryValues =
  {{"Gas", "λ0 (m)", "σλ0 (m)", "E0 (eV)", "σE0 (eV)", "I1 (eV)", "ne", "σne", "# of e-
per particle", "# of e- in
valence shell/
HOMO"}, {"Air", λ0Air[[1]],
    NumberForm[λ0Air[[2]], 2], e0[λ0Air][[1]], e0[λ0Air][[2]], "N2: 15.6;
O2: 12.1", neAir[[1]], NumberForm[neAir[[2]], 2], "N2: 14;
O2: 16", "N2: 2;
O2: 2"}, {"CO2", λ0Co2[[1]], NumberForm[λ0Co2[[2]], 2], e0[λ0Co2][[1]],
    e0[λ0Co2][[2]], "13.8", neCo2[[1]], NumberForm[neCo2[[2]], 3], "22", "4"},
   {"Helium", λ0Helium[[1]], NumberForm[λ0Helium[[2]], 2],
    e0[λ0Helium][[1]], e0[λ0Helium][[2]], "24.6",
    neHelium[[1]], NumberForm[neHelium[[2]], 2], "2", "2"}};
Grid[summaryValues, Alignment → Left, Spacings → {1, 1}, Frame → All,
 ItemStyle → Directive[FontSize -> 10, "Text"],
 Background → {{Gray, None}, {LightGray, None}}]
```

Out[79]=

Gas	λ_0 (m)	σ_{λ_0} (m)	E_0 (eV)	σ_{E_0} (eV)	I_1 (eV)	n_e	σ_{n_e}	# of e⁻ per particle	# of e⁻ in valence shell/ HOMO
Air	1.03082×10^{-7}	1.7×10^{-8}	12.0286	1.98919	N_2: 15.6; O_2: 12.1	2.20437	0.73	N_2: 14; O_2: 16	N_2: 2; O_2: 2
CO_2	8.91589×10^{-8}	$2.\times10^{-8}$	13.9071	3.05092	13.8	4.51692	1.98	22	4
Helium	1.09248×10^{-7}	2.6×10^{-8}	11.3498	2.72031	24.6	0.247266	0.12	2	2

Conclusion

We observe that n_e appears to have some dependence on the number of electrons per particle, but upon closer observation a clearer trend is observed whereby n_e corresponds closely to the number of electrons in the highest occupied molecular orbital (HOMO) for air (mixture of N_2 and O_2) and CO_2. The trend appears to be broken by the discrepancy between n_e for helium and the number of electrons in its valence shell. This could be due to the fact that these electrons are in a more stable state than those in the non-bonding HOMO of N_2, O_2 and the antibonding MO of CO_2 (i.e. the relationship does not hold for very stable electronic states).

We observe that E_0 appears to correspond roughly to the ionization energy (I_1) of the molecules. Again this trend is broken for Helium. This could also be due to the stability of the electronic state of electrons in helium as described above.

In both cases, we have too few data points and too large uncertainties to conclude anything about the above trends and further investigation is needed.