Experiment 212 **March 11, 2022**

Viscosity of fluids

Beginner's Practica for Physicists II

Juan Provencio Tutor: Malinda de Silva

Contents

1 Goal of the experiment

In this experiment we will analyze the phenomenom of laminar flow by taking a look at the motion of falling spheres in a fluid and the flow of a fluid through a thin tube. For this we will calculate the viscosity of this fluid and Reynold's number for different setups, which will tell us the level of "laminarity" of the respective flow.

2 Foundations

2.1 Friction

For an object to continue moving through a fluid with a constant speed, a force must be applied to it due to the fact that the fluid imposes a friction, force which acts against the movement of this object. This force depends on a special characteristic of the fluid in which we find ourselves. For example, moving a spoon through honey is much more challenging than in a cup of coffee or tea. In a simplified model, we picture layers of material with a surface area A, moving with a velocity v and which are stacked on top of each other at a distance z from a resting surface against which they slide, as depicted in Figure 1

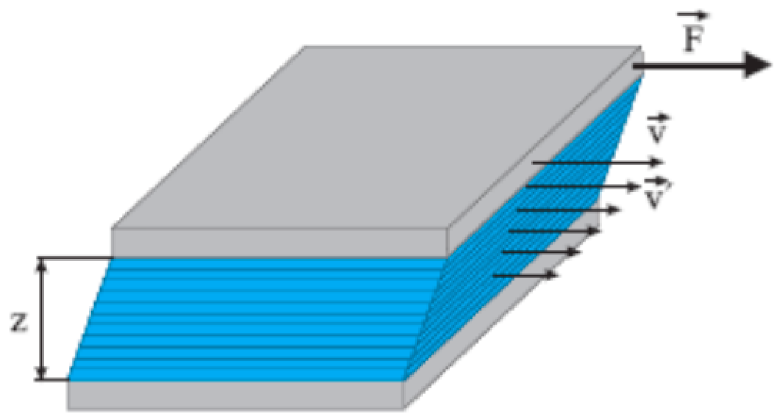


Figure 1: Internal friction between stacked layers of material upon which a force is exerted

On these layers, a force is exerted which is proportional to the velocity, the surface area and inversely proportional to the distance z from the resting anchor point. The aforementioned "characteristic" also comes in play here, as it represents the material dependency of the force and includes a material proportionality constant to the equation.

This quantity is called ”viscosity”.

$$F_r = \eta A \frac{v}{z} \tag{1}$$

This force is better expressed by the velocity gradient

$$= \eta A \frac{\mathrm{d}v}{\mathrm{d}z}. \tag{2}$$

2.2 Laminar and turbulent flow

We can also use this layer model to model fluids. If these layers of fluid encounter a disturbance and remain unmixed afterwards, the fluid is called laminar, as shown in Figure 2 left. If afterwards the fluid mixes it is called turbulent flow. At least in a simplified explanation of this. The takeaway, however, is that the model for friction which we use stops being valid in turbulent flow.

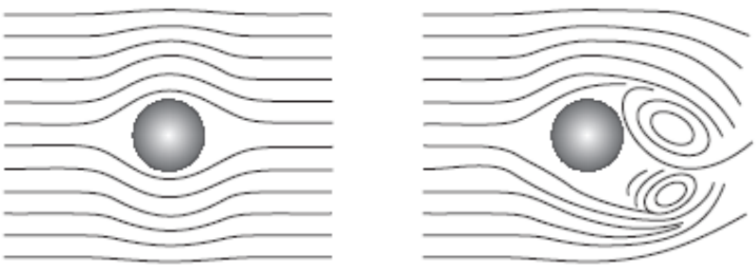


Figure 2: Laminar (left) and turbulent (right) flow

Laminar flow is however a fragile phenomenom, that can be broken with slight disturbances, so it is a good idea to quantisize the level of ”laminarity” of a flow, which can be done using the quasiempirical value of ”Reynold’s number”. This value can be defined as the quotient of twice the kinetic energy and the work done by friction. The larger the number, the more turbulent is the flow. This means that a fast flowing fluid and / or one with a low viscosity are more prone to turbulence. For $E_{\text{kin}} \leq W_{\text{fric}}$, the flow is laminar. With a characteristic length L we can also determine the number as

$$Re = \frac{\rho v L}{\eta} \tag{3}$$

with the mean flow velocity v, the density of the fluid ρ, the diameter L of a pipe (if we’re observing a pipe) and the viscosity η. The critical Reynold’s number Re_{cr} is context specific. For example for fluids flowing through a tube, the value lies around $Re_{\text{cr}} \approx 2300$, while for a ball sinking down a fluid this critical value is around $Re_{\text{cr}} \approx 1$.

2.3 Viscometer after Stokes

For a ball with radius r moving at a constant speed v through a liquid, the force exerted by friction is

$$F_r = 6\pi\eta r v \tag{4}$$

which is called Stokes' Law. This law is only valid for laminar flow, as mentioned above and for infinitely extended liquids. Since it is assumed that the speed is constant, we have to take a look at the moment where the other forces balance out for the velocity to be constant. The forces taken into consideration are of course gravity, buoyancy and friction.

$$F_r + F_g + F_b = 0 \tag{5}$$

out of that we can derive the viscosity

$$\eta = \frac{2}{9} g \frac{\rho_k - \rho_f}{v} r^2 \tag{6}$$

with the density of the balls ρ_k, the fluid ρ_f and the measured velocity v.

2.4 Viscometer after Hagen-Poiseuille

Alternatively, we can take a look at laminar flow through a pipe of length L and radius R, without disturbances. A flow is [] when there is a pressure difference between both ends. We can induce this by a height difference in the fluid. The force due to this pressure difference is

$$F_p = \pi r^2 \Delta p \tag{7}$$

with the pressure difference Δp and the frictional force is

$$F_r = -2\pi r L \eta \frac{\mathrm{d}v}{\mathrm{d}r}. \tag{8}$$

For laminar flow, both forces must cancel out to

$$F_p = F_r \tag{9}$$

which gives us following relationship

$$\frac{\mathrm{d}v}{\mathrm{d}r} = \frac{\Delta p}{2\eta L} r. \tag{10}$$

Since we don't want this dependance on the velocity gradient, we can integrate over r and with the velocity integrate over the cross section of the tube to get the flow rate.

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \frac{\pi \Delta p R^4}{8\eta L}. \tag{11}$$

If we plot the volume over time and fit a linear function to get the slope, we can solve (11) after the viscosity η

$$\eta = \frac{\pi \Delta p R^4}{8 \frac{\mathrm{d}V}{\mathrm{d}t} L}. \tag{12}$$

3 Experimental setup

3.1 Materials and equipment

- Measuring scales with tempered glass, filled with polyethylene glycol. At the lower part of the cylinder there is a precision capillary viscometer (length: 100 mm ± 0.5 mm, capillary diameter 1.5 mm ± 0.01 mm).
- Balls of "Hostaform C" with the following diameters: 2r = 2.0 / 3.0 / 4.0 / 5.0 / 6.0 / 7.144 / 8.0 / 9.0 mm (± 1%). The density of the balls and the density of polyethylene glycol are given in the Appendix.
- Thermometer
- Tweezers, beaker glass
- Scale
- Stopwatches

3.2 Setup

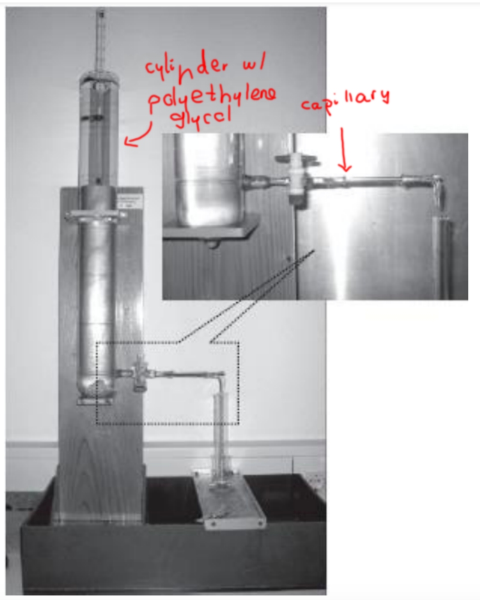
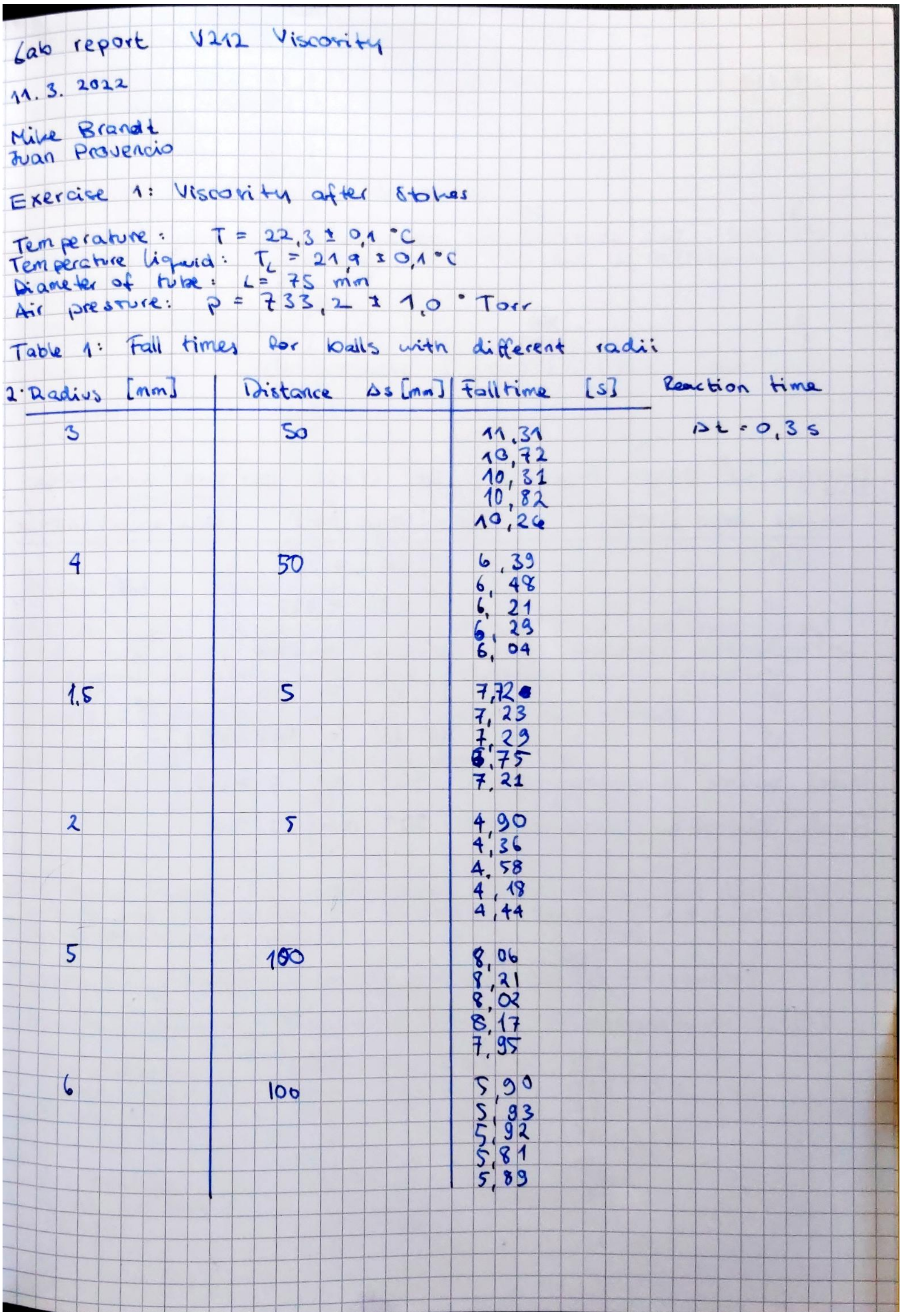


Figure 3: Experimental setup

4 Measurement and evaluation

4.1 Lab report



Lab report V212 Viscosity

11. 3. 2022

Mike Brandt
Juan Provencio

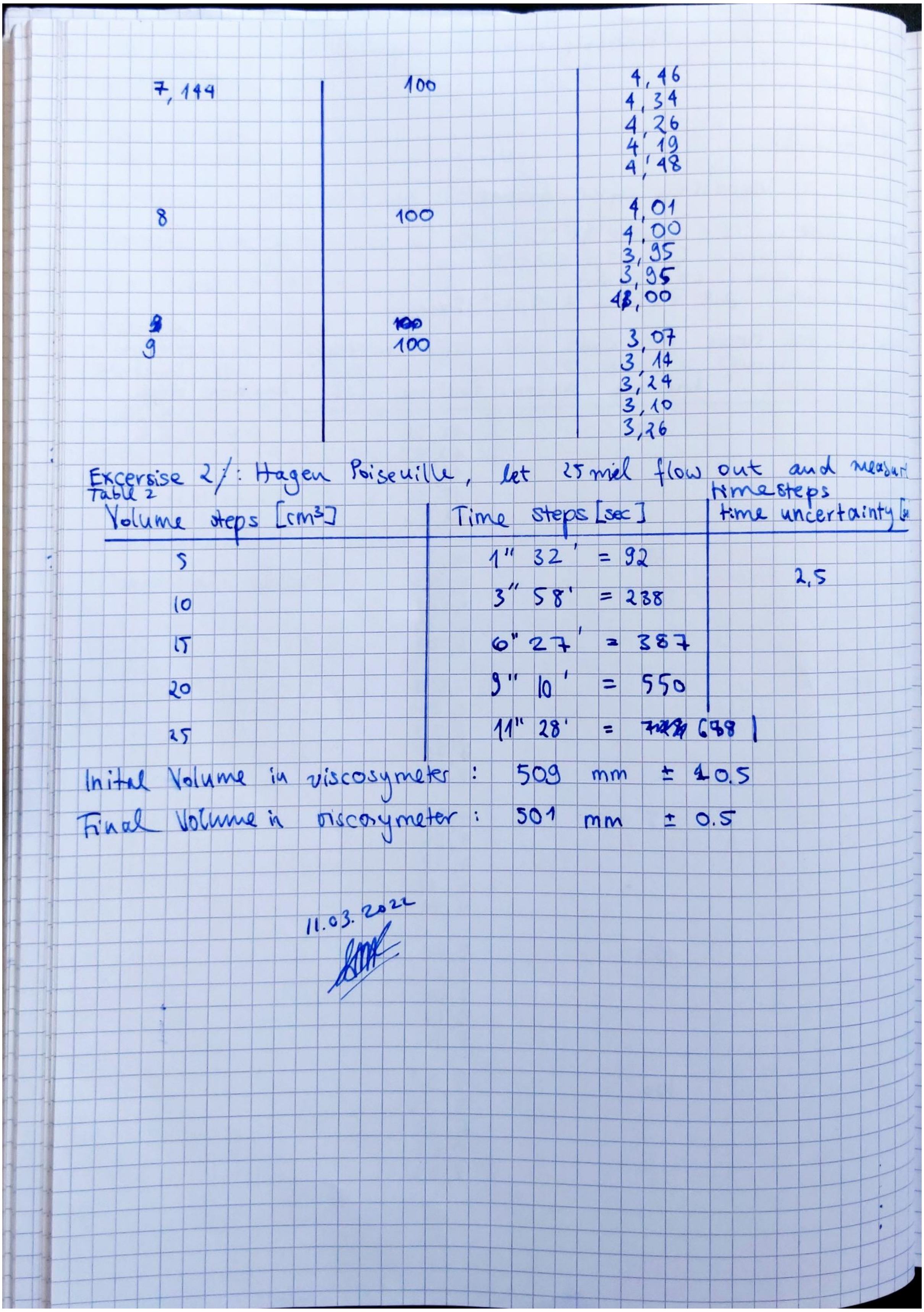
Exercise 1: Viscosity after Stokes

Temperature : $T = 22{,}3 \pm 0{,}1$ °C
Temperature liquid : $T_L = 21{,}9 \pm 0{,}1$ °C
Diameter of tube : $L = 75$ mm
Air pressure : $p = 733{,}2 \pm 1{,}0$ Torr

Table 1: Fall times for balls with different radii

2·Radius [mm]	Distance Δs [mm]	Falltime [s]	Reaction time
3	50	11,31	Δt = 0,3 s
		10,72	
		10,31	
		10,82	
		10,26	
4	50	6,39	
		6,48	
		6,21	
		6,29	
		6,04	
1,5	5	7,72	
		7,23	
		7,29	
		6,75	
		7,21	
2	5	4,90	
		4,36	
		4,58	
		4,18	
		4,44	
5	100	8,06	
		8,21	
		8,02	
		8,17	
		7,95	
6	100	5,90	
		5,93	
		5,92	
		5,81	
		5,89	

Figure 4: Lab report



7,144	100	4,46 4,34 4,26 4,19 4,48
8	100	4,01 4,00 3,95 3,95 4,00
~~9~~ 9	~~100~~ 100	3,07 3,14 3,24 3,10 3,26

Excersise 2/: Hagen Poiseuille, let 25 mil flow out and measure timesteps

Table 2

Volume steps [cm³]	Time steps [sec]	time uncertainty [s]
5	1" 32' = 92	2,5
10	3" 58' = 238	
15	6" 27' = 387	
20	9" 10' = 550	
25	11" 28' = 688	

Initial Volume in viscosymeter : 509 mm ± 0.5

Final Volume in viscosymeter : 501 mm ± 0.5

11.03.2022

Figure 5: Lab report

4.2 Evaluation

In the following we will use gaussian error propagation to calculate the uncertainty of our measurements. This bases on the idea that the errors are relatively small and the factors in a formula are independent of each other. For trivial cases this will not be done explicitly in the evaluation, but the exact formulas can still be consulted in the Annex where the Python code is.

4.2.1 Viscosity after Stokes

To determine the viscosity after Stokes we used a falling sphere viscometer for spheres of the radii given in subsection 3.1. Additionally we conducted the test with the little spheres of radius $r = 1.5/2\,\mathrm{mm}$, but this value has been removed due to irregularities in the evaluation and a lack of density needed for the calculations. The measurement with the sphere of radius $r = 2/2\,\mathrm{mm}$ was also removed due to similar irregularities in the evluation.

First of all, we had to calculate the mean sinking velocity of the spheres after they've reached a constant speed. This was done using the regular statistic functions of the numpy package and the error was given as the error of the mean value. Since the different spheres used had a slightly different density, we adapted the velocity by dividing it with the corresponding density difference to the fluid

$$\frac{\overline{v}}{\rho_k - \rho_f} \equiv \frac{\overline{v}}{\rho} \tag{13}$$

and this value we plotted against the squared radius of the spheres. According to (6), we can use the slope of this fit to determine the viscosity. With the slope m the following is valid

$$\eta = \frac{2}{9}\frac{g}{m}. \tag{14}$$

At the start we made the assumption that the fluid was infinitely stretched, which is clearly not the case. To account for this assumption we need to multiply (4) with a factor λ defined through

$$\lambda = \left(1 + 2.1\frac{r}{R}\right) \tag{15}$$

where R is the radius of the pipe. This means, the velocities must be multiplied by this factor λ. In Diagramm 1 we show both fits and how they differ from each other.

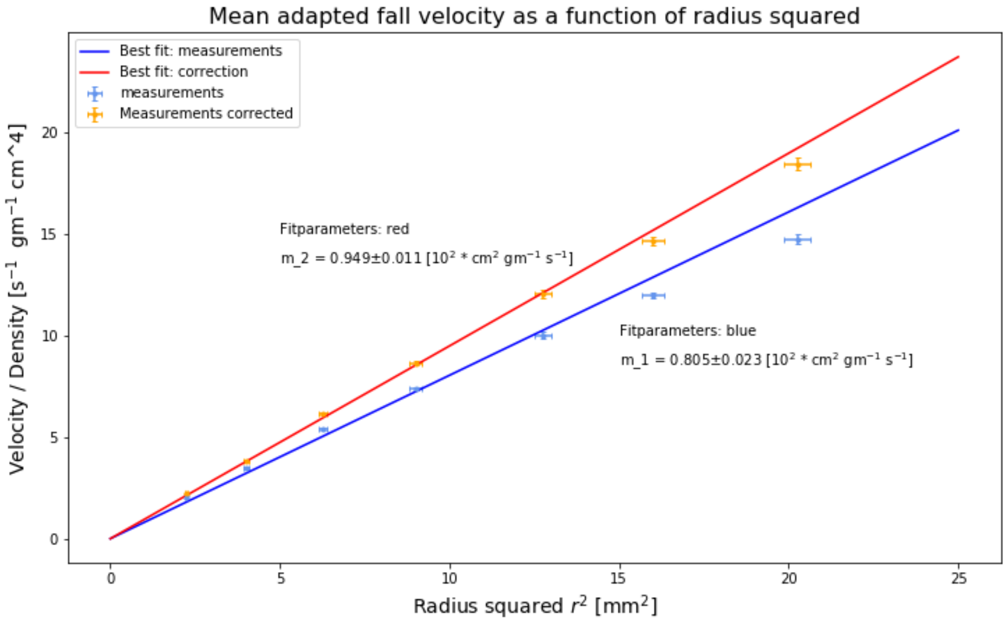


Diagramm 1: Quotient of mean velocity over density difference over squared radius

From both fits we obtain the following results

$$m_1 = 0.805(23) \cdot 10^2\,\mathrm{cm}^2\,\mathrm{g}^{-1}\,\mathrm{s}^{-1} \tag{16}$$
$$m_2 = 0.949(11) \cdot 10^2\,\mathrm{cm}^2\,\mathrm{g}^{-1}\,\mathrm{s}^{-1} \tag{17}$$
$$\tag{18}$$

which provide the following viscosities

$$\eta_1 = 2,71(8)\,\mathrm{g\,cm^{-1}\,s^{-1}} = 0,271(8)\,\mathrm{Pa\,s} \tag{19}$$

$$\eta_2 = 2,296(26)\,\mathrm{g\,cm^{-1}\,s^{-1}} = 0,2296(26)\,\mathrm{Pa\,s} \tag{20}$$

In all following calculations we'll use the corrected value η_2. With this we can determine Reynold's number for all given radii and velocities. We get

Table 1: Reynold's number for given radii and velocities

Radius [mm]	1.5	2	2.5	3	3.572	4	4.5
Velocity [$\mathrm{cm\,s^{-1}}$]	0.469(8)	0.796(10)	1.237(7)	1.698(6)	2.30(3)	2.511(8)	3.16(4)
Re	0.07(2)	0.159(3)	0.309(5)	0.509(8)	0.822(16)	1.004(16)	1.424(28)

The error for Reynold's number was calculated using the relative error of (3) as

$$\Delta Re = Re \cdot \sqrt{\left(\frac{\Delta\rho_f}{\rho_f}\right)^2 + \left(\frac{\Delta\overline{v}}{\overline{v}}\right)^2 + \left(\frac{\Delta r}{r}\right)^2 + \left(\frac{\Delta\eta}{\eta}\right)^2}. \tag{21}$$

Next we calculated the laminar velocity predicted by (6) as

$$v_{\text{lam}} = \frac{2}{9} g \frac{\rho}{\eta} \tag{22}$$

and plotted the quotient of the mean velocity over the laminar velocity against Reynold's number in a logarithmic scale.

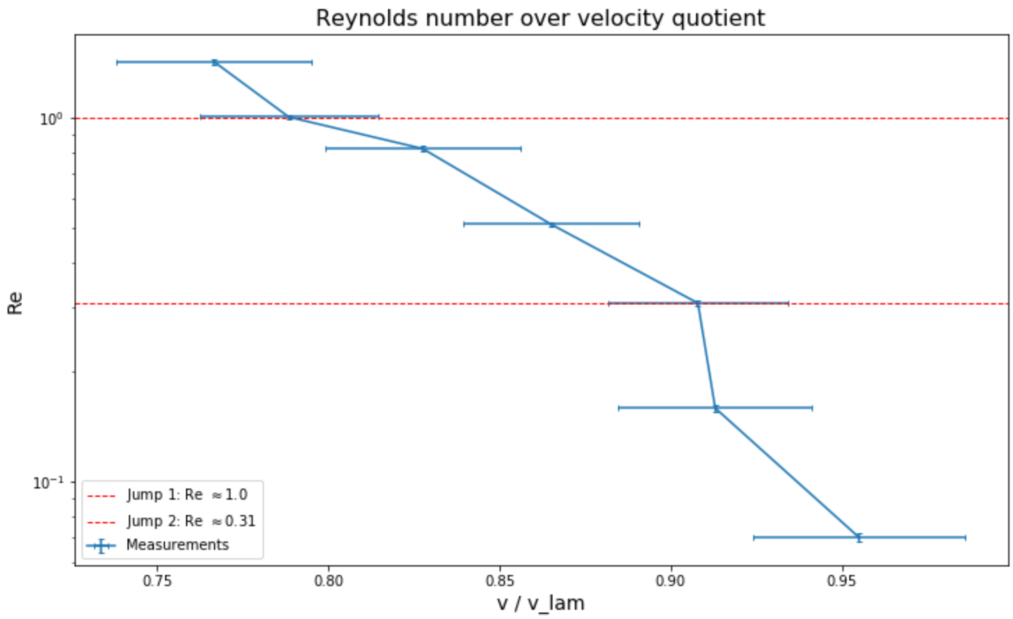


Diagramm 2: Velocity quotient over Reynold's number

In this Diagramm, we were supposed to find the vaguely defined places where the graph "jumps", because at these points we expect Stoke's friction law to stop being valid. This happens twice, once at around $Re \approx 1$ and more drastically at around $Re \approx 0.31$. The first jump, however, may be influenced by our preconceived notion that we'd find a jump at the critical Reynold's number 1, at which the flow stops being laminar. Otherwise we would probably not have included this as a significant observation, since the jump is merely noticeable and can be attributed to a slight measuring error.

4.3 Viscosity after Hagen-Poiseuille

Next we want to determine the viscosity using (11), for which we need the volume flow $\frac{\mathrm{d}V}{\mathrm{d}t}$ which we can do graphically using a linear fit. Similarly to the previous fit, we used a linear approximation anchored at the origin, since we expect the volume that has flowed at the time $t = 0$ to be also zero.

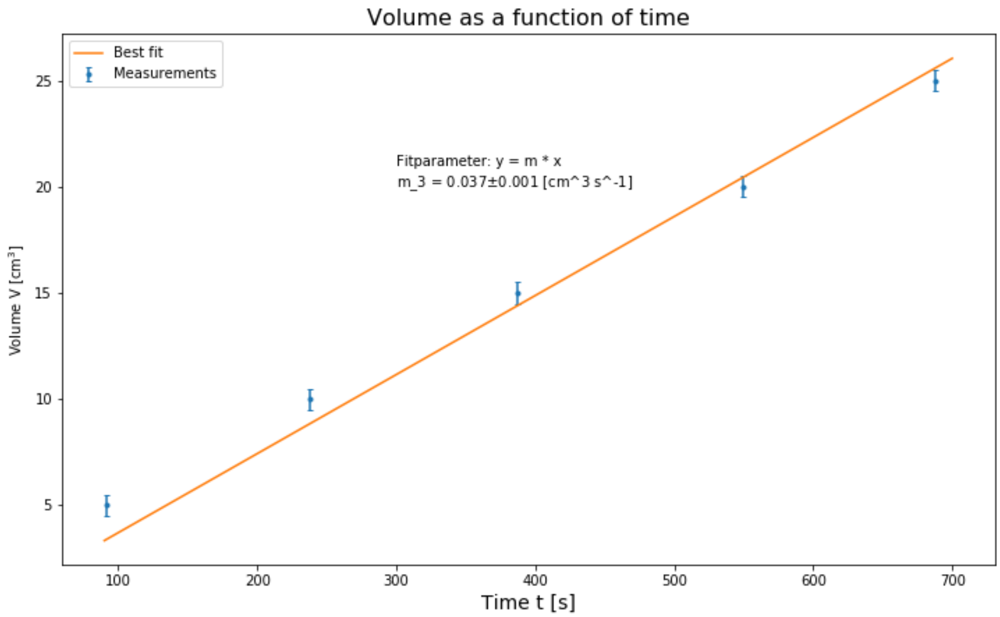


Diagramm 3: Volume over time

We obtain the slope

$$m_3 = \frac{\mathrm{d}V}{\mathrm{d}t} = 0.037(1)\,\mathrm{cm}^3\,\mathrm{s}^{-1} \tag{23}$$

which helps us determine the viscosity given the remaining quantities of the capillary's radius R, the lenght of the capillary L and the pressure difference Δp, which we obtain by converting the mean height of the liquid in the pipe

$$\overline{h} = \frac{h_{\text{initial}} + h_{\text{final}}}{2} \tag{24}$$

$$= 505.0(4)\,\mathrm{mm} \tag{25}$$

and multiplying it by the density of the fluid and Earth's gravity

$$\Delta p = \rho_f g \overline{h} \tag{26}$$

$$= 5686(4)\,\mathrm{Pa}. \tag{27}$$

We finally get

$$\eta_{\text{hp}} = 0.190(8)\,\text{Pa}\,\text{s} \tag{28}$$

For this kind of flow we'd also like to determine Reynold's number. In this case, the critical point is much higher since we're observing a fundamentally different system. We can calculate it using the same equation as before, by calculating the velocity v_A as the quotient of the volume flow through the area

$$v_A = \frac{\mathrm{d}V}{\mathrm{d}t} \cdot \frac{1}{\pi R^2}. \tag{29}$$

The resulting Reynold's number is negligibly small:

$$Re = 0.095(5) \tag{30}$$

so we can be mostly absolute certain that there was laminar flow inside the capillary.

5 Summary and discussion

5.1 Summary

During this experiment, we took a look at two distinct methods of determining the viscosity of a fluid. In one case, we measured the velocity of spheres of different radii falling through the fluid, over which we could approximate a linear fit between this velocity and the squared of their radii. In this case, laminar flow is necessary and thus a Reynold's number that's smaller than 1.

In the second experiment, we took a look at the fluid running through a thin capillary laminarly and measuring the volume change over time. With the volume flow we could similarly determine the viscosity, and here it was much easier to have laminar flow, requiring a Reynold's number of under 2300.

5.2 Discussion

We'll begin by comparing the viscosity of the liquid obtained through both methods. We obtained

Table 2: Comparison between viscosities

	Stokes	$\frac{\Delta\eta}{\eta}$ [%]	Hagen-Poiseuille	$\frac{\Delta\eta_{\text{hp}}}{\eta_{\text{hp}}}$ [%]	σ
η [Pa s]	0.2296(26)	1.15	0.190(8)	4	4.9

The σ-deviation was calculated through

$$\frac{|\eta - \eta_{\text{hp}}|}{\sqrt{(\Delta\eta)^2 + (\Delta\eta_{\text{hp}})^2}} \tag{31}$$

We get a large deviation of nearly 5 σ-ranges, which is unacceptable for the standards held in this course. In the following we will discuss some of the reasons why the experiment may have gone askew. Before delving into the differences in methodology, we'll analyze our results. Using the χ^2-Analysis performed in the Annex in the Python-Code, we can see that none of the fits performed have a particularly positive result. In fact, the probabilty of fit appears to be 0% in all of them. This can lie in the fact that the errors were underestimated or that the data deviates from the theory.

After calculating the Reynold's numbers for each measurement, we can observe that in the first method the critical number is exceeded twice for the radii $r = 4\,\text{mm}$ and $r = 4.5\,\text{mm}$. Since laminarity isn't as strict as establishin a hard border, we can't with certainty dismiss these values as being incompatible with our theory, but it is certainly an explanation for the large discrepancies. Dismissing these values as measuring errors or incompatible with theory yields no significantly better results and the lost information is far too great. A large source for error in the second measuring method could be the fact that there are way fewer data points in total. For the pressure difference for example, we only measured two points, and an inaccurate reading of these values could have falsified the results. To counter this uncertainty a little bit, we used 5 data points to extrapolate the volume flow, instead of just two. Since the flow in this case is much more consistently laminar, we can also expect more reliable results.

Another way to judge the results would be by taking a look at Diagramm 2. According to theory, one would expect turbulent flow to happen when there is a drastic jump in the diagramm. The drastic jump we can observe, however, happens at a much smaller number than we'd expect, so one of the conclusions could be, that in fact most of the measurements were conducted under turbulent flow conditions. Only the previous knowledge of a critical Reynold's number of approximately 1 led us to believe that most measurements had been under laminar flow. Our interpretation of the results contradicts this and could be a further explanation for the large discrepancy. The turbulent flow could have been caused by small air bubbles, which we did not observe during the execution or on an impatient measurement. If the little balls hadn't reached a terminal velocity when we started the stopwatch, the results could have been falsified. To counter this effect, we measured the fall time for the same length at different heights of the pipe. For example between the 200-300, the 300-400 and the 400-500 markings, and we observed similar times in all of these measurements, so we can dismiss this hypothesis.

There's no comparison value and without knowing what kind of polyethylene glycol was used it's impossible to infer out of external sources what the accurate viscosity would have been.

6 Sources

Wagner, J., Universität Heidelberg (2021). Physikalisches Praktikum PAP 2.1 für Studierende der Physik B.Sc..

Experiment Viscosity of fluids

March 24, 2022

7 Annex

[1]:
```
import matplotlib.pyplot as plt
import matplotlib.mlab as mlab
%matplotlib inline
import numpy as np
from numpy import exp, sqrt, log, pi
from scipy.optimize import curve_fit
from scipy.stats import chi2
from scipy import odr
from scipy.integrate import quad
from scipy.signal import find_peaks
from scipy.signal import argrelextrema, argrelmin, argrelmax


def fehler(name, G, sig_G, G_lit, sig_G_lit):
    print(name)
    print('Relativer Fehler: ', sig_G / G * 100)
    print('Rel. Fehler (Vergleich):', sig_G_lit / G_lit * 100)
    print('Absoluter Fehler: ', G - G_lit)
#     print('Verhältnis:', G / G_lit)
    print('Sigma-Abweichung: ', np.abs(G - G_lit) / sqrt(sig_G ** 2
                                                            + sig_G_lit **␣
 ↪2),'\n')


def fehler_small(name, G, sig_G):
```

```
    print(name)
    print('Relativer Fehler: ', sig_G / G * 100)


def ergebnis(name, G, sig_G, komma, einheit):
    print(name + ' =', np.round(G, komma), '+/-', np.round(sig_G,␣
 ↪komma), einheit)


def ergebnis_large(name, G, sig_G, komma, einheit):
    print(name + ' =', np.round(G, komma))
    print('+/-'.rjust(len(name) + 2), np.round(sig_G, komma), einheit)


def fitparameter(name, G, sig_G, komma, einheit):
    return name + ' = ' + str(np.round(G, komma)) + '$\pm$' + str(np.
 ↪round(sig_G, komma)) + einheit
```

Determination of the viscosity after Stokes with a falling sphere viscometer

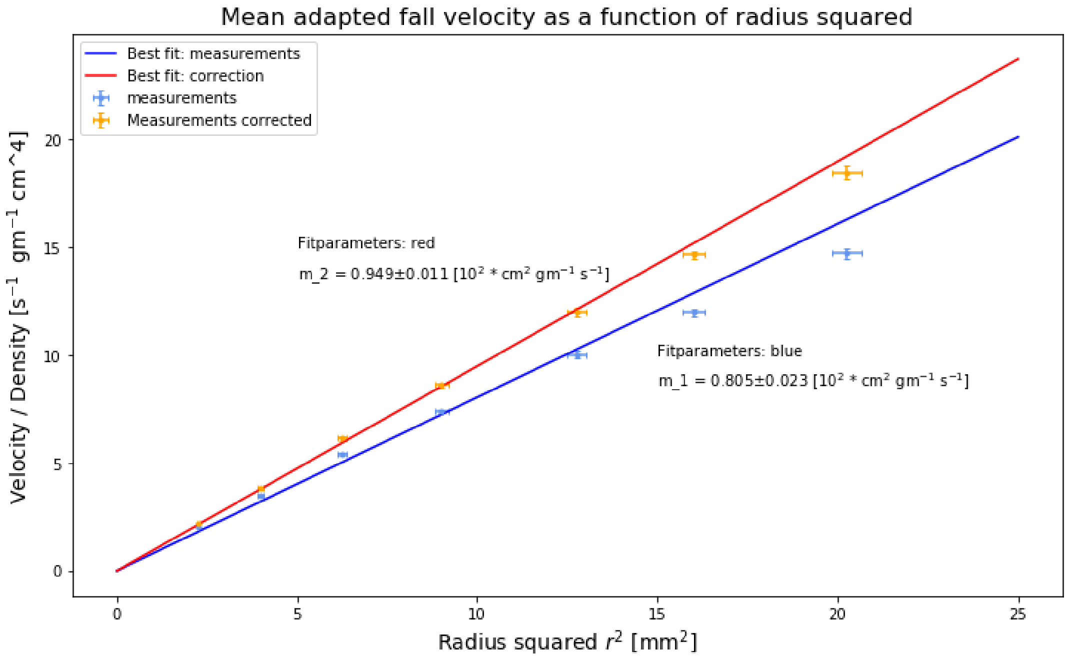
Measurements

[2]:
```
radius = np.array([3, 4, 1.5, 2, 5, 6, 7.144, 8, 9]) / 2 # cm

mask = np.ones(len(radius), dtype = bool) # Remove problematic 1.5, 2␣
 ↪diameter
mask[[2, 3]] = False

radius = radius[mask, ...]

sig_radius = radius * 0.01
radius_sq = radius ** 2
sig_radius_sq = 2 * radius * sig_radius

distance = np.array([50, 50, 5, 5, 100, 100, 100, 100, 100]) / 10 # cm
distance = distance[mask, ...]
```

```
falltime = np.array([np.array([11.31, 10.72, 10.31, 10.82, 10.26]),
                     np.array([6.39, 6.48, 6.21, 6.29, 6.04]),
                     np.array([7.72, 7.23, 7.29, 7.75, 7.21]),
                     np.array([4.90, 4.36, 4.58, 4.18, 4.44]),
                     np.array([8.06, 8.21, 8.02, 8.17, 7.95]),
                     np.array([5.90, 5.93, 5.92, 5.81, 5.89]),
                     np.array([4.46, 4.34, 4.26, 4.19, 4.48]),
                     np.array([4.01, 4.00, 3.95, 3.95, 4.00]),
                     np.array([3.07, 3.14, 3.24, 3.10, 3.26])]) # s
falltime = falltime[mask, ...]

# Density of balls
rho_k = np.array([1.3775, 1.3775, 1.3775, 1.3775, 1.3775, 1.3775, 1.
 ↪3775,
               1.3575, 1.3625]) # g cm^-3
rho_k = rho_k[mask, ...]
sig_rho_k = 0.0025
# Density of polyethylene
rho_pol = 1.1478 # g cm^-3
sig_rho_pol = 0.0002
# Density difference
rho = rho_k - rho_pol
sig_rho = sqrt( (sig_rho_k) ** 2 + (sig_rho_pol) ** 2 )

R = 75 / 2 # mm



# Mean velocity
v_mean = np.array([np.mean(distance[i] / falltime[i])
                   for i in range(len(distance))]) # cm s^-1
sig_v_mean = np.array([1 / sqrt(5 - 1) * np.std(distance[i] /␣
 ↪falltime[i])
                       for i in range(len(distance))])

v_mean_rho = v_mean / rho
```

```
sig_v_mean_rho = v_mean_rho * sqrt( (sig_v_mean / v_mean) ** 2 +␣
 ↪(sig_rho / rho) ** 2 )

# Correction term
lambda_k = (1 + 2.1 * (radius / R))
sig_lambda_k = 2.1 / R * sig_radius
v_mean_r_l = v_mean_rho * lambda_k
sig_v_mean_r_l = sqrt( (sig_v_mean_rho * lambda_k) ** 2
                      + (v_mean_rho * sig_lambda_k) ** 2 )

# Fit
def line(x, m):
    return m * x

popt_1, pcov_1 = curve_fit(line, radius_sq, v_mean_rho, sigma =␣
 ↪sig_v_mean_rho)

m_1 = popt_1[0]
#b_1 = popt_1[1]
sig_m_1 = sqrt(pcov_1[0, 0])
#sig_b_1 = sqrt(pcov_1[1, 1])

popt_2, pcov_2 = curve_fit(line, radius_sq, v_mean_r_l, sigma =␣
 ↪sig_v_mean_r_l)

m_2 = popt_2[0]
#b_2 = popt_2[1]
sig_m_2 = sqrt(pcov_2[0, 0])
#sig_b_2 = sqrt(pcov_2[1, 1])

x = np.linspace(0, 25, 101)

# Plot
plt.figure(figsize = (12, 7))
plt.title('Mean adapted fall velocity as a function of radius squared',
          size = 16)
```

```
# Uncorrected
plt.errorbar(radius_sq, v_mean_rho, xerr = sig_radius_sq, yerr =␣
 ↪sig_v_mean_rho,
             fmt = '.', capsize = 2, label = 'measurements', color =␣
 ↪'cornflowerblue')
plt.plot(x, line(x, *popt_1), label = 'Best fit: measurements', color =␣
 ↪'blue')
plt.text(15, 10, 'Fitparameters: blue')
plt.text(15, 8.5, fitparameter('m_1', m_1, sig_m_1, 3,
                              ' [10$^2$ * cm$^2$ gm$^{-1}$ s$^{-1}$]'))
#plt.text(65, 7, fitparameter('b_1', b_1, sig_b_1, 2,
#                              ' [cm$^4$ gm$^{-1}$ s$^{-1}$]'))

# Corrected
plt.errorbar(radius_sq, v_mean_r_l, xerr = sig_radius_sq, yerr =␣
 ↪sig_v_mean_r_l,
            fmt = '.', capsize = 2, label = 'Measurements corrected',␣
 ↪color = 'orange')
plt.plot(x, line(x, *popt_2), label = 'Best fit: correction', color =␣
 ↪'red')
plt.text(5, 15, 'Fitparameters: red')
plt.text(5, 13.5, fitparameter('m_2', m_2, sig_m_2, 3,
                              ' [10$^2$ * cm$^2$ gm$^{-1}$ s$^{-1}$]'))
#plt.text(10, 12, fitparameter('b_2', b_2, sig_b_2, 2,
#                              ' [cm$^4$ gm$^{-1}$ s$^{-1}$]'))
plt.xlabel('Radius squared $r^2$ [mm$^2$]', size = 14)
plt.ylabel('Velocity / Density [s$^{-1}$  gm$^{-1}$ cm^4]',
          size = 14)
plt.legend(loc = 'upper left')

plt.savefig('images/212/V212Diagramm1.png')
```



[3]:
```
# Good of the fit: blue
chi2_= np.sum((line(radius_sq, *popt_1) - v_mean_rho) ** 2
              / sig_v_mean_rho ** 2)
dof = len(sig_v_mean_rho) - 1 #dof:degrees of freedom, Freiheitsgrad
chi2_red = chi2_/dof

print("chi2 =", chi2_)
print("chi2_red =",chi2_red)

prob = np.round(1 - chi2.cdf(chi2_,dof),2) * 100
print("Probability =", prob, "%")
```

```
chi2 = 159.07396182485218
chi2_red = 26.512326970808697
Probability = 0.0 %
```

[4]:
```
# Good of the fit: red
chi2_= np.sum((line(radius_sq, *popt_2) - v_mean_r_l) ** 2
              / sig_v_mean_r_l ** 2)
dof = len(sig_v_mean_r_l) - 1 #dof:degrees of freedom, Freiheitsgrad
chi2_red = chi2_/dof
```

```
print("chi2 =", chi2_)
print("chi2_red =",chi2_red)

prob = np.round(1 - chi2.cdf(chi2_,dof),2) * 100
print("Probability =", prob, "%")
```

```
chi2 = 26.409693714750645
chi2_red = 4.401615619125107
Probability = 0.0 %
```

Viscosity

[5]:
```
g = 9.81 # m s^-2
eta_1 = 2 * g / (9 * m_1)
eta_2 = 2 * g / (9 * m_2)
sig_eta_1 = eta_1 * sqrt( (sig_m_1 / m_1) ** 2 )
sig_eta_2 = eta_2 * sqrt( (sig_m_2 / m_2) ** 2 )

ergebnis('eta_1', eta_1, sig_eta_1, 2, '[g cm^-1 s^-1]')
ergebnis('eta_2 (corr)', eta_2, sig_eta_2, 3, '[g cm^-1 s^-1]')
```

```
eta_1 = 2.71 +/- 0.08 [g cm^-1 s^-1]
eta_2 (corr) = 2.296 +/- 0.026 [g cm^-1 s^-1]
```

Reynolds number

[6]:
```
Re = rho_pol * v_mean * 2 * radius / eta_2 * 1e-1
sig_Re = Re * sqrt( (sig_rho_pol / rho_pol) ** 2
                   + (sig_v_mean / v_mean) ** 2
                   + (sig_radius / radius) ** 2
                   + (sig_eta_2 / eta_2) ** 2)

print('Correspondnig radius ', radius, '[mm]')
ergebnis_large('Reynolds number: Re', Re, sig_Re, 3, '')
```

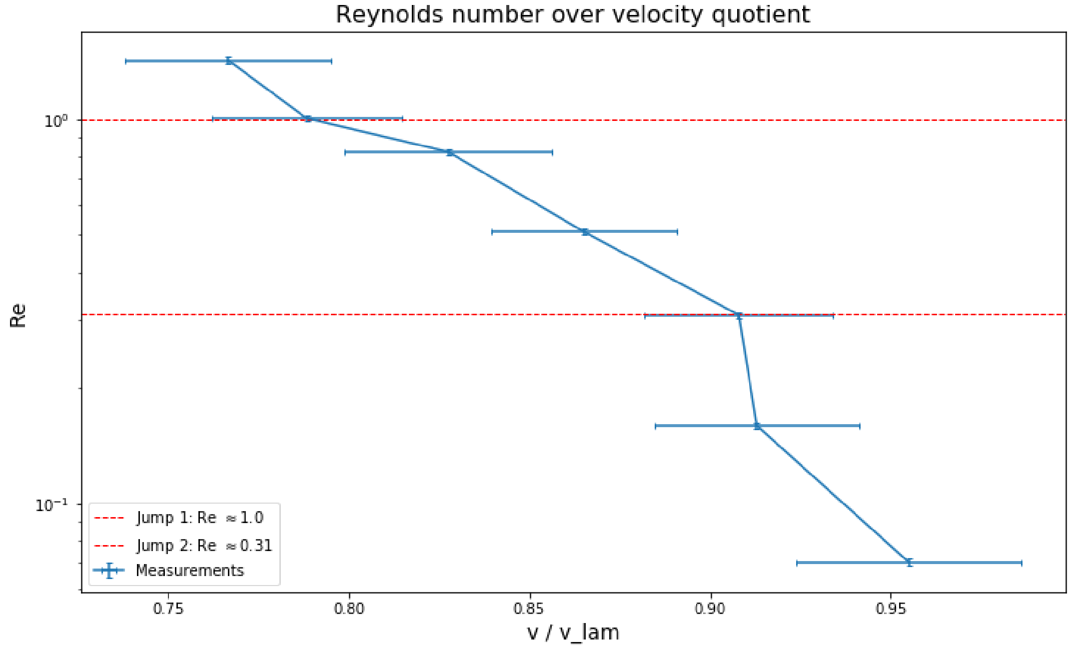
```
Correspondnig radius  [1.5   2.    2.5   3.    3.572 4.    4.5  ] [mm]
Reynolds number: Re = [0.07  0.159 0.309 0.509 0.822 1.004 1.424]
                  +/- [0.002 0.003 0.005 0.008 0.016 0.016 0.028]
```

Laminar velocity

[7]:
```
v_lam = 2 / 9 * g * radius_sq * rho / eta_2
sig_v_lam = v_lam * sqrt( (sig_radius_sq / radius_sq) ** 2
                        + (sig_rho / rho) ** 2
                        + (sig_eta_2 / eta_2) ** 2)
print('radius: ', radius, '[mm]')
ergebnis_large('v_mean', v_mean, sig_v_mean, 3, '[cm s^-1]')
ergebnis_large('v_lam', v_lam, sig_v_lam, 3, '[cm s^-1]')

# Plot
vv_lam = v_mean / v_lam
sig_vv_lam = sqrt( (sig_v_mean / v_mean) ** 2
                 + (sig_v_lam / v_lam) ** 2 )

ergebnis_large('vv_lam', vv_lam, sig_vv_lam, 3, '')

plt.figure(figsize = (12, 7))
plt.errorbar(vv_lam, Re, xerr = sig_vv_lam, yerr = sig_Re, fmt = '-',
             capsize = 2, label = 'Measurements')
plt.yscale('log')
plt.title('Reynolds number over velocity quotient', size = 16)
plt.xlabel('v / v_lam', size = 14)
plt.ylabel('Re', size = 14)
hor2 = 3.1e-1
hor1 = 1e0
plt.axhline(hor1, linestyle = '--', color = 'red', linewidth = 1,
            label = 'Jump 1: Re $\\approx$' + str(hor1))
plt.axhline(hor2, linestyle = '--', color = 'red', linewidth = 1,
            label = 'Jump 2: Re $\\approx$' + str(hor2))
plt.legend(loc = 'best')

plt.savefig('images/212/V212Diagramm2.png')
```

```
radius:  [1.5   2.    2.5   3.    3.572 4.    4.5  ] [mm]
v_mean = [0.469 0.796 1.237 1.698 2.302 2.511 3.164]
     +/- [0.008 0.01  0.007 0.006 0.03  0.008 0.038] [cm s^-1]
v_lam = [0.491 0.872 1.363 1.963 2.782 3.185 4.128]
    +/- [0.013 0.022 0.035 0.05  0.071 0.083 0.107] [cm s^-1]
```

```
vv_lam = [0.955 0.913 0.908 0.865 0.828 0.788 0.767]
     +/- [0.031 0.028 0.026 0.026 0.029 0.026 0.028]
```



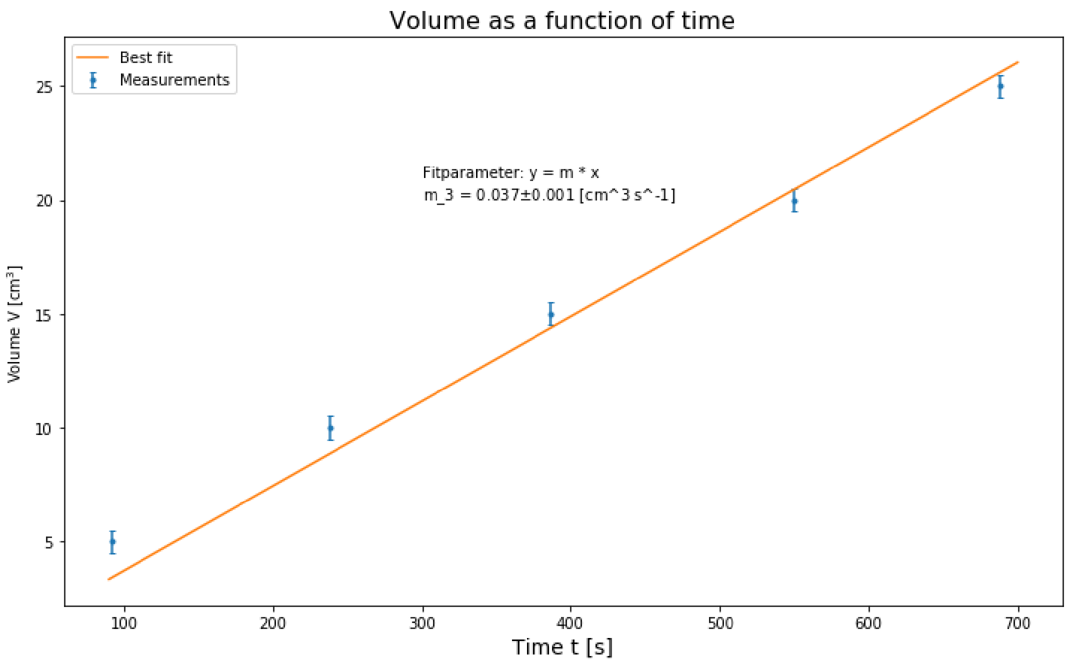
Determination of the viscosity after Hagen-Poiseuille with a capillary viscometer

Measurements

[8]:
```
volume = np.linspace(5, 25, 5) # cm^3
timesteps = np.array([92, 238, 387, 550, 688]) # s
sig_time = 2.5
vol_initial = 509 # mm
vol_final = 501
sig_vol = 0.5
```

[9]:
```
# Fit
popt_3, pcov_3 = curve_fit(line, timesteps, volume)

m_3 = popt_3[0] # cm^3 s^-1
sig_m_3 = sqrt(pcov_3[0, 0])
```

```
t = np.linspace(90, 700, 200)

# Plot
plt.figure(figsize = (12, 7))
plt.errorbar(timesteps, volume, yerr = sig_vol, fmt = '.', capsize = 2,
             label = 'Measurements')
plt.plot(t, line(t, *popt_3), label = 'Best fit')
plt.text(300, 21, 'Fitparameter: y = m * x')
plt.text(300, 20, fitparameter('m_3', m_3, sig_m_3, 3, ' [cm^3 s^-1]'))
plt.title('Volume as a function of time', size = 16)
plt.xlabel('Time t [s]', size = 14)
plt.ylabel('Volume V [cm$^3$]')
plt.legend(loc = 'best')

plt.savefig('images/212/V212Diagramm3.png')
```



[10]:
```
# Good of the fit: orange
chi2_= np.sum((line(timesteps, *popt_3) - volume) ** 2
              / sig_vol ** 2)
dof = len(volume) - 1 #dof:degrees of freedom, Freiheitsgrad
```

```
chi2_red = chi2_/dof

print("chi2 =", chi2_)
print("chi2_red =",chi2_red)

prob = np.round(1 - chi2.cdf(chi2_,dof),2) * 100
print("Probability =", prob, "%")
```

```
chi2 = 18.925005122531953
chi2_red = 4.731251280632988
Probability = 0.0 %
```

Viscosity

[15]:
```
dVdt = m_3 # cm^3 s^-1
sig_dVdt = sig_m_3
hbar = (vol_initial + vol_final) / 2 # mm
sig_hbar = sqrt(2) / 2 * sig_vol

Delta_p = rho_pol * g * hbar # Pa
sig_Delta_p = rho_pol * g * sig_hbar

R = 1.5 / 2 # mm
sig_R = 0.01 / 2
L = 100 # mm
sig_L = 0.5

eta_hp = pi * Delta_p * R ** 4 / (8 * L * dVdt) * 1e-3 # Pa s
sig_eta_hp = eta_hp * sqrt( (sig_Delta_p / Delta_p) ** 2 + (4 * sig_R /
 ↪R) ** 2
                            + (sig_L / L) ** 2 + (sig_dVdt / dVdt) ** 2)

ergebnis('eta_hp', eta_hp, sig_eta_hp, 3, '[Pa s]')
```

```
5686.25859 3.9809821867022346
eta_hp = 0.19 +/- 0.008 [Pa s]
```

Reynold's number

[18]:
```
v_A = dVdt / (pi * R ** 2)
sig_v_A = v_A * sqrt( (sig_dVdt / dVdt) ** 2 + (2 * sig_R / R) ** 2 )
Re_hp = rho_pol * v_A * R / eta_hp
sig_Re_hp = Re_hp * sqrt( (sig_rho_pol / rho_pol) ** 2 + (sig_v_A /␣
↪v_A) ** 2
                        + (sig_L / L) ** 2 + (sig_eta_hp / eta_hp) **␣
↪2)

ergebnis('Re_hp', Re_hp, sig_Re_hp, 3, '')
```

```
Re_hp = 0.095 +/- 0.005
```

Sigmas

[19]:
```
# Viskosität
fehler('eta', 1e-1 * eta_2, 1e-1 * sig_eta_2, eta_hp, sig_eta_hp)
```

```
eta
Relativer Fehler:  1.1507098108390523
Rel. Fehler (Vergleich): 3.9997486817598626
Absoluter Fehler:  0.039647596117293155
Sigma-Abweichung:  4.928246283960424
```

[ ]:
```
```