

Versuch 222 Viskosimeter

Leonardo Karl Reiter

March 21, 2024

Contents

1 Zielsetzung	2
2 Grundlagen	2
2.1 Reibungskraft	2
2.2 Reynoldszahl Re	2
2.3 Kugelfallviskosimeter	3
2.4 Kapillarviskosimeter	4
3 Durchführung	5
3.1 Bestimmung der Viskosität nach Stokes	5
3.2 Bestimmung der Viskosität nach Hagen-Poiseuille	6
4 Auswertung	7
4.1 Teil 1: Kugelfallviskosimeter	7
4.1.1 Wichtige Größen	7
4.1.2 Messgrößen	8
4.1.3 Berechnung der Terminalgeschwindigkeiten v_{te}	8
4.1.4 Ladenburgsche Korrektur von v_{te}	9
4.1.5 Berechnung der dynamischen Viskosität η nach Stokes und Ladenburg	10
4.1.6 Reynoldszahl Re	12
4.1.7 Bestimmung der Kritischen Raynoldszahl Re_{kr}	13
4.1.8 Dynamische Viskosität η nach Stokes ohne Turbulenzen.	16
4.2 Teil 2: Kapillarviskosimeter	17
4.2.1 Berechnen der Durchflussmenge Q	18
4.2.2 Berechnung der dynamischen Viskosität η nach Hagen-Poiseuille	19
4.2.3 Bestimmung der Reynoldszahl Re_{Ka} in der Kapillare	20
5 Diskussion	21
5.1 Raynoldszahlen beim Kugelfallviskosimeter	21
5.2 Fitgüten beim Kugelviskosimeter	21
5.3 Fazit	22

1 Zielsetzung

Ziel des Versuchs ist es die Viskosität von Polyethylenglykol auf zwei verschiedene Weisen zu bestimmen. Zuerst wird ein sogenanntes Kugelfallviskosimeter verwendet, hier werden die Fallgeschwindigkeiten, verschieden großer Kugeln gemessen um so über ein Kräftegleichgewicht die Viskosität zu bestimmen. Die zweite Methode verwendet das Hagen-Poiseuille-Gesetz um die Viskosität mittels eines Kapillarviskosimeters zu bestimmen. Anschließend werden beide Werte miteinander verglichen um zu sehen ob diese miteinander übereinstimmen.

2 Grundlagen

Wird ein Objekt in ein Fluid bewegt so wird es die, wird das Fluid Einfluss auf dessen Bewegung haben, das Fluid wird eine Kraft auf das Objekt ausüben welche von der Geschwindigkeit v , der Viskosität η und der Beschaffenheit des Objekts abhängen wird. Diese Reibungskraft wird von intermolekularen Wechselwirkungen innerhalb des Fluids und in der Grenzschicht zwischen FLuid und Objekt verursacht. Die Viskosität η ist somit ein Maß dafür wie stark diese Wechselwirkungen und die daraus resultierenden Reibungskräfte sind. Oder anders gesagt charakterisiert die Viskosität die "Zähigkeit" von Fluiden.

2.1 Reibungskraft

Allgemein ist die Reibung zwischen einer glatten Oberfläche A , mit einer relativen Geschwindigkeit v zum Fluid, in einem Abstand z zur Oberfläche wirkt, nach Newton gegeben durch:

$$F_R = \eta \cdot A \cdot \frac{v}{z} \quad (1)$$

$$= \eta \cdot A \cdot \frac{\partial v}{\partial z} \quad (2)$$

Hier ist η eine fluidspezifische Größe mit den SI-Einheiten: $[\eta] = \text{Pa} \cdot \text{s}$, angegeben wird und als **dynamische Viskosität** bezeichnet. Dieses Newtonsches Reibungsgesetz gilt allerdings nur in **Laminarströmen**, bei denen die Fluidschichten linear voneinandergleiten ohne sich zu vermischen. Bei hohen Geschwindigkeiten (fluidabhängig) und gewissen Körpergeometrien ist bricht der laminare Strom zusammen und es kommt zu Vermischungen der Fluidschichten, man spricht dann von **Turbulenzströmung**. In solchen turbulenten Strömungen ist der Strömungswiderstand deutlich größer und das Newtonsche Reibungsgesetz (2) ist nichtmehr gültig.

2.2 Reynoldszahl Re

Um abzuschätzen ob eine Strömung laminar, oder turbulent ist definiert man die Reynoldszahl Re . Ergibt sich aus dem Vergleich der Kinetischen Energie E_{kin} und den Reibungsverlusten W_{Reibung} wie folgt:

$$Re = \frac{2E_{\text{kin}}}{W_{\text{Reibung}}} \quad (3)$$

Bei großen Reynoldszahlen ist wird die Strömung zunehmend instabil.

Ab einer **kritischen Reynoldszahl** Re_{kr} bricht die Laminarströmung zusammen und wird vollständig turbulent. Je größer die kinetische Energie des Fluids desto instabiler wird die Strömung. Hohe Reibungsverluste können dies wieder ausgleichen sodass die Strömung wieder laminar wird.

Die Reynoldszahl lässt sich ebenso über die geometrischen Eigenschaften des Strömungssystem charakterisierten. Dabei definiert man eine **charakteristische Länge** L . Diese entspricht n einem Rohrsystem

beispielsweise dem Rohrdurchmesser d_{Rohr} , bewegt man hingegen ein Kugel durch eine Flüssigkeit wie wir es beim Kugelfallviskosimeter vorhaben, beschreibt L dem Durchmesser der Kugel d_{Kugel} .

$$Re = \frac{\rho v}{\eta} \cdot L \quad (4)$$

$$= \frac{\rho v}{\eta} \cdot d_{\text{Rohr}} \quad (5)$$

$$= \frac{\rho v}{\eta} \cdot d_{\text{Kugel}} \quad (6)$$

Die kritische Reynoldszahl Re_{kr} ist systemabhängig und wird meist experimentell bestimmt. Beispielsweise zeigt sich, dass der Fluss einer Flüssigkeit durch ein Rohr, ab einer Reynoldszahl $Re_{kr} = 2300$ turbulent wird.

Bei einer Kugel die durch eine Flüssigkeit bewegt wird ist die Reynoldszahl viel kleiner und die Strömung wird bei $Re_{kr} \approx 1$ schon turbulent.

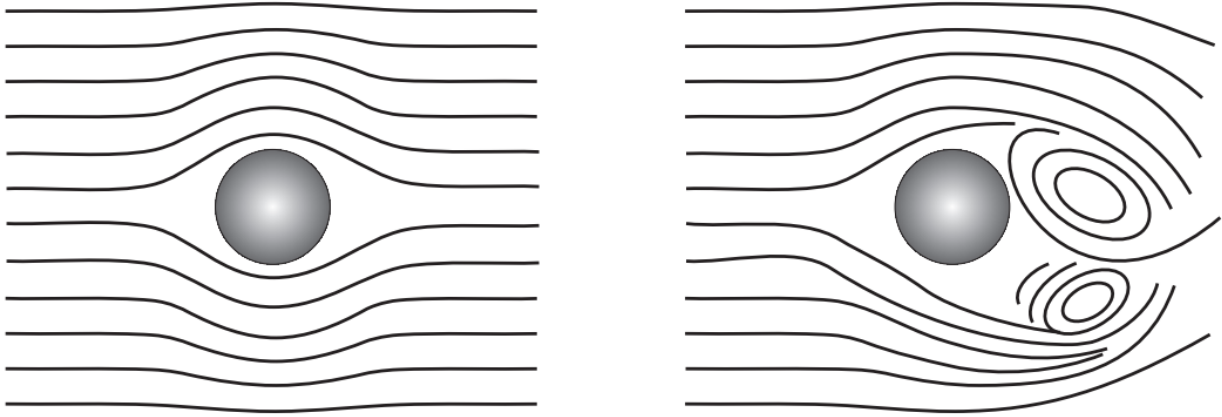


Figure 1: Bewegung einer Kugel durch eine Flüssigkeit. Links: Laminare Strömung bei der die Flüssigkeit den Körper symmetrisch umfließt. Die einzelnen Schichten gleiten aneinander ab ohne sich zu vermischen. Rechts: Turbulente Strömung bei hohen Geschwindigkeiten. In Folge der Wirbelbildung kommt es zu einer Vermischung der Flüssigkeit.

2.3 Kugelfallviskosimeter

Bewegt sich ein Kugel des Radius r mit konstanter Geschwindigkeit v durch eine Flüssigkeit mit dynamischer Viskosität η so wirkt auf sie die Reibungskraft F_R :

$$F_R = 6\pi\eta rv \quad (7)$$

Dies ist die Stokessche Reibung und ist nach wie vor nur in laminaren Strömungen, und unendlich ausgedehnten Flüssigkeiten gültig. Lassen wir eine Kugel in das Viskosimeter fallen so Wirken drei Kräfte. Die Schwerkraft $F_g = \rho_K V_K g$, die Auftriebskraft $F_A = -\rho_f V_K g$ und die Stokessche Reibung (7). Zwischen diesen soll ein Gleichgewicht herrschen sodass die Kugel mit konstanter Geschwindigkeit, ihrer **terminalgeschwindigkeit** v_{te} fällt.

$$F_g + F_A + F_R = 0 \quad (8)$$

$$\Rightarrow v = v_{te} = \text{const.} \quad (9)$$

$$\Rightarrow \eta = \frac{2}{9} g \frac{(\rho_K - \rho_f)}{v_{te}} r^2 \quad (10)$$

Durch die Messung der Fallgeschwindigkeit v_{te} lässt sich die dynamische Viskosität η des Mediums abschätzen.

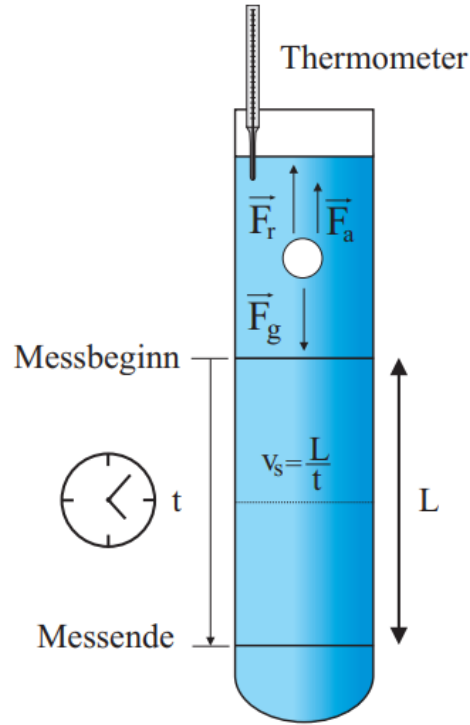


Figure 2: Bestimmung der Viskosität einer Flüssigkeit mit einem Kugelfallviskosimeter. Bewegt sich die Kugel mit konstanter Geschwindigkeit, heben sich alle angreifenden Kräfte auf.

2.4 Kapillarviskosimeter

Die andere Methode die wir untersuchen wollen, ist die des Kapillarviskosimeters, dabei wird der Volumenstrom einer laminaren Rohrströmung gemessen.

Fließt eine Flüssigkeit durch ein Rohr der Länge L Radius R und Druckdifferenz $\Delta p = p_1 - p_2$ zwischen Ein- und Ausgang so wirkt, aufgrund der Druckdifferenz Δp , auf jeden der coaxialen Zylindermäntel mit Radius $r < R$ (laminare Strömung), die Kraft F_p :

$$F_p = \pi r^2 (p_1 - p_2) \quad (11)$$

Andererseits wirkt auch die Newtonsche Reibungskraft F_R :

$$F_R = -2\pi r L \eta \frac{dv}{dr} \quad (12)$$

Bei einer Stationären Strömung bei der sich die einzelnen Schichten mit konstanter Geschwindigkeit bewegen, muss wieder ein Kräftegleichgewicht herrschen:

$$F_p = F_R \quad (13)$$

$$-2\pi r L \eta \frac{dv}{dr} = \pi r^2 (p_1 - p_2) \quad (14)$$

Daraus folgt für den Geschwindigkeitsgradienten dv/dr :

$$\frac{dv}{dr} = \frac{(p_1 - p_2)}{2\eta L} r \quad (15)$$

$$\Rightarrow v(r) = \frac{(p_1 - p_2)}{4\eta L} (R^2 - r^2) \quad (16)$$

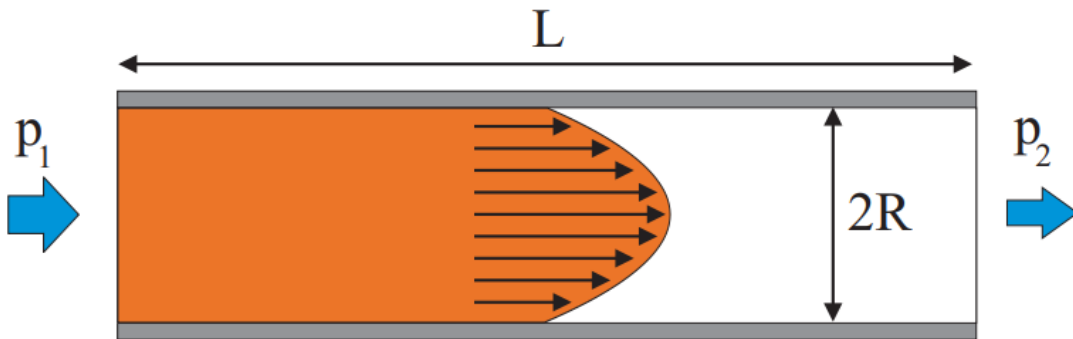


Figure 3: Laminare Rohrströmung. Unter dem Einfluss der Druckdifferenz $\Delta p = p_1 - p_2$ strömt die Flüssigkeit in einem zylindrischen Rohr mit einem parabelförmigen Geschwindigkeitsprofil.

Den Volumenstrom durch die Querschnittsfläche des Rohres bestimmt man über Integration des Geschwindigkeitsprofils $v(r)$ über die gesamte Querschnittsfläche $A_Q = \pi R^2$.

$$\frac{dV}{dt} = \int_0^R 2\pi r \cdot v(r) dr = \frac{\pi(p_1 - p_2)R^4}{8\eta L} \quad (17)$$

Sind Radius R und Länge L des Rohres, sowie Druckdifferenz $\Delta = (p_1 - p_2)$ bekannt. So lässt sich über die das Hagen-Poiseuille-Gesetz die Viskosität bestimmen.

3 Durchführung

3.1 Bestimmung der Viskosität nach Stokes

Im ersten Teil des Versuchs werden mehrere Kugeln verschiedener Radien in das Kugelfallviskosimeter gelegt und die Zeit festgehalten in der sie die Strecke zwischen zwei Markierungen zurücklegen. Von jedem Durchmesser werden 5 Kugeln untersucht um so den Mittelwert zu bilden um den menschlichen Fehler zu reduzieren. Wichtig dabei ist es zu verhindern dass Luftbläschen an der Kugel haften bleiben da diese einfluss auf Volumen und Auftriebskraft hätten.

3.2 Bestimmung der Viskosität nach Hagen-Poiseuille

An der unteren rechten Seite der Anordnung befindet sich das Kapillarviskosimeter, der Hahn wird vorsichtig geöffnet und es wird der Abfluss in einer gewissen Zeit festgehalten. Wichtig dabei ist es zu warten dass sich die Strömungsverhältnisse stabilisiert haben, da sonst die Messung verfälscht ist.

4 Auswertung

```
[1]: #Benötigte Pakete
%matplotlib inline
import numpy as np
import io
from decimal import Decimal
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from scipy.stats import chi2
plt.rcParams.update({'font.size': 20})
plt.rcParams["figure.figsize"] = 16, 9

[2]: def sig(val1,err1,val2,err2):
    sig = np.abs(val2-val1)**2/np.abs(np.abs(err2)+np.abs(err1))**2
    return round(sig,3)

def format_e(n):
    a = '%e' % Decimal(n)
    return a.split('e')[0].rstrip('0').rstrip('.')+'e'+a.split('e')[1]

def form(val,err):
    a = "{value:4.3f} ± {error:4.3f}".format(value=val,error=err)
    return a

def fitgüte(x,y,delta_x,delta_y,popt,fit_funktion):
    chisquare=np.sum((fit_funktion(popt, x)-y)**2/
                    (delta_y**2+((fit_func(popt, x+delta_x)-fit_func(popt,
→x-delta_x))/2)**2))

    #Freiheitsgrade
    dof=dof = x.size-popt.size
    if dof == 0:
        chisquare_red = 'inf'
        prob = 100
    else:
        chisquare_red = chisquare/dof
        prob = round(1-chi2.cdf(chisquare,dof),2)*100 #Fitwahrscheinlichkeit

    print('chi_squared= ' + str(chisquare))#format_e(chisquare))
    print('chi_squared_red= ' + str(chisquare_red))#format_e(chisquare_red))
    print('Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: ' +
          str(prob) + ' %')
```

4.1 Teil 1: Kugelfallviskosimeter

4.1.1 Wichtige Größen

```
[3]: g=9.81 # m/s**2
p0=1000.3 # hPa
dp0=0.2 # hPa
T = 23 # °C
dt = 0.5 # °C
```



```
D_rohr = 75*1e-3 # m
dD_rohr = 1*1e-3 # m
rho_Fl=1.1468e3 #kg/m**3
drho_Fl=0.0006e3 #kg/m**3
```

4.1.2 Messgrößen

```
[4]: # Kugeln
d = np.array([1.5,2,3,4,5,6,7.144,8,9])*1e-3 # m
d_err = d*0.01 # Fehler 1%

rho_K = np.array([1.390,1.375,1.375,1.375,1.375,1.375,1.375,1.355,1.360])*1e3 #kg/m^3
drho_K = np.ones(rho_K.size)*0.005 #kg/m^3

# Messwerte für Geschwindigkeiten
T = np.array([np.array([31.56,29.58,31.70,31.81,31.54]), # sec
              np.array([41.37,40.31,40.37,38.32,37.90]),
              np.array([10.14, 9.92,10.23, 9.67, 9.87]),
              np.array([12.24,11.67,11.90,12.32,11.54]),
              np.array([15.18,15.54,16.17,15.95,15.46]),
              np.array([11.65,11.50,11.50,11.07,11.07]),
              np.array([ 8.39, 8.31, 8.36, 8.20 ,8.70]),
              np.array([ 7.67, 7.50, 7.70, 7.43, 8.04]),
              np.array([ 5.89, 5.92, 6.00, 5.93, 5.95]),])

t = np.ones(len(T))
dt_sys = 0.3 # sec
dt = np.ones(len(T))

for i in range(0,len(T),1):
    t[i] = np.mean(T[i])
    dt[i] = np.std(T[i]) + dt_sys # stat. Fehler + syst. Fehler

s = np.array([50,100,50,100,200,200,200,200,200])*1e-3 # m
ds = 0.002 # m
```

4.1.3 Berechnung der Terminalgeschwindigkeiten v_{te}

Die Terminalgeschwindigkeit v_{te} ergibt sich aus:

$$v_{te} = \frac{s}{t} \quad (18)$$

$$\Delta v_{te} = \sqrt{\left(\frac{1}{t} \Delta s\right)^2 + \left(\frac{s}{t^2} \Delta t\right)^2} \quad (19)$$

```
[5]: def v_te(s,ds,t,dt):
    v = s/t
    dv = np.sqrt(((1)/(t)*ds)**2 + ((s)/(t**2)*dt)**2)
    return v, dv

v, dv = v_te(s,ds,t,dt)
```

```

print('\nTerminalgeschwindigkeit v_te:\n')
for i in range(0,len(v),1):
    print('v_te_'+str(i+1)+' =', form(v[i]*1e3,dv[i]*1e3), ' [mm/s]')
print('')

```

Terminalgeschwindigkeit v_te:

```

v_te_1 = 1.601 ± 0.086 [mm/s]
v_te_2 = 2.522 ± 0.115 [mm/s]
v_te_3 = 5.017 ± 0.322 [mm/s]
v_te_4 = 8.379 ± 0.457 [mm/s]
v_te_5 = 12.771 ± 0.549 [mm/s]
v_te_6 = 17.609 ± 0.858 [mm/s]
v_te_7 = 23.832 ± 1.348 [mm/s]
v_te_8 = 26.082 ± 1.760 [mm/s]
v_te_9 = 33.681 ± 1.938 [mm/s]

```

4.1.4 Ladenburgsche Korrektur von v_{te}

Wir gehen davon aus dass die Terminalgeschwindigkeit linear von dem Quadrat des Kugelradius r^2 abhängt

$$v_{te} = \frac{2}{9}g \frac{(\rho_K - \rho_f)}{\eta} \cdot r^2 \quad (20)$$

$$\Rightarrow \frac{v_{te}}{(\rho_K - \rho_f)} = \frac{2g}{9\eta} \cdot r^2 \quad (21)$$

Diese gilt aber nur für unendlich ausgedehnte Flüssigkeiten, Unsere ist in einem Zylinder mit Durchmesser $D = 75 \pm 1$ [mm]. Dazu wird die sogenannte Ladenburgsche Korrektur eingeführt:

$$v_{korr} = \lambda \cdot v_{te} \quad \text{mit } \lambda = \left(1 + 2.1 \frac{d}{D}\right) \quad (22)$$

$$\Delta v_{korr} = \sqrt{(\Delta \lambda v_{te})^2 + (\lambda \Delta v_{te})^2} \quad \text{mit } \Delta \lambda = \sqrt{\left(\frac{1}{D} \Delta d\right)^2 + \left(\frac{d}{D^2} \Delta D\right)^2} \quad (23)$$

```

[6]: def v_korr(v,dv):
    lam = (1 + 2.1*d/D_rohr)
    dlam = np.sqrt(((1)/(D_rohr)*d_err)**2 + ((d)/(D_rohr**2)*dD_rohr)**2)
    v_korr = v*lam
    dv_korr = np.sqrt((dlam*v)**2 + (dv*lam)**2)
    return v_korr, dv_korr

v_korr, dv_korr = v_korr(v,dv)

print('\nLadenburg Korrektur v_korr:\n')
for i in range(0,len(v_korr),1):
    print('v_korr_'+str(i+1)+' =', form(v_korr[i]*1e3,dv_korr[i]*1e3), ' [mm/s]')
print('')

```

Ladenburg Korrektur v_{korr} :

```
v_korr_1 = 1.668 ± 0.090 [mm/s]
v_korr_2 = 2.663 ± 0.121 [mm/s]
v_korr_3 = 5.438 ± 0.349 [mm/s]
v_korr_4 = 9.318 ± 0.509 [mm/s]
v_korr_5 = 14.559 ± 0.626 [mm/s]
v_korr_6 = 20.567 ± 1.002 [mm/s]
v_korr_7 = 28.599 ± 1.618 [mm/s]
v_korr_8 = 31.925 ± 2.155 [mm/s]
v_korr_9 = 42.169 ± 2.428 [mm/s]
```

4.1.5 Berechnung der dynamischen Viskosität η nach Stokes und Ladenburg

```
[33]: x, dx = (d/2)**2, 2*d/4 *d_err # [mm^2]
y, dy = v/(rho_K - rho_Fl), dv/(rho_K - rho_Fl)
y_korr, dy_korr = v_korr/(rho_K - rho_Fl), dv_korr/(rho_K - rho_Fl)

def fit_func(p, x):
    (s) = p
    return s*x

def fit(x,dx,y,dy,c,name,para0,fit_func,plt_range):

    from scipy import odr
    model = odr.Model(fit_func)

    x = x
    y = y
    delta_x = dx
    delta_y = dy

    #Startparameter
    para0 = para0

    data = odr.RealData(x, y, sx=delta_x, sy=delta_y)
    odr = odr.ODR(data, model, beta0=para0)
    out = odr.run()

    #1-Sigma
    popt = out.beta
    perr = out.sd_beta

    #Sigma-Umgebung
    nstd = 1 # um n-Sigma-Umgebung im Diagramm zu zeichnen

    popt_top = popt+nstd*perr
    popt_bot = popt-nstd*perr

    #Plot-Umgebung
    x_fit = np.linspace(min(x)/plt_range, max(x)*plt_range, 1000)
```

```

fit = fit_func(popt, x_fit)
fit_top = fit_func(popt_top, x_fit)
fit_bot = fit_func(popt_bot, x_fit)

plt.errorbar(x, y, xerr=delta_x, yerr=delta_y, lw=1, ecolor='k', mec=c, mfc=c,
fmt='.', capsize=2,
label='Messdaten '+name)
plt.plot(x_fit, fit, color=c, lw=1, label='Fit '+name)
plt.fill_between(x_fit, fit_top, fit_bot, color=c, alpha=.1,
label=str(nstd)+r'$\sigma$'+'-Umgebung')
print('\nFitgüte '+ name+ ':\n')
fitgüte(x,y,delta_x,delta_y,popt,fit_func) #[: -3]
return popt, perr

popt, perr = fit(x,dx,y,dy,
'r','(ohne Korrektur)', [1], fit_func, 1.1)
popt_korr, perr_korr = fit(x,dx,y_korr,dy_korr,
'g','(mit Korrektur)', [1], fit_func, 1.1)

#Auswertung
def Eta(popt,perr):
    S = popt[0]
    dS = perr[0]
    eta = 2*g/(9*S)
    deta = 2*g/(9*S**2)* dS
    return eta, deta

eta, deta = Eta(popt,perr)
eta_korr, deta_korr = Eta(popt_korr, perr_korr)
print('')
plt.ticklabel_format(axis='both', style='sci', scilimits=(0,5), useMathText=True)
plt.title('Dynamische Viskosität $\eta$ nach Stokes')
plt.xlabel('Quadrat des Kugelradius $r^2$ [m$^2$]')
plt.ylabel('Sinkgeschwindigkeit $\frac{v}{\rho_K - \rho_f}$ [m/s]')
plt.grid(ls=':')
plt.legend(loc='best', fontsize=14)
plt.show()

print('\nDynamische Viskosität:\n')
print('(ohne Korrektur): eta = ', form(eta,deta))
print('(mit Korrektur): eta = ', form(eta_korr,deta_korr))
print('sigma = ', sig(eta,deta,eta_korr,deta_korr))

```

Fitgüte (ohne Korrektur):

```

chi_squared= 47.1186244178148
chi_squared_red= 5.88982805222685
Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: 0.0 %

```

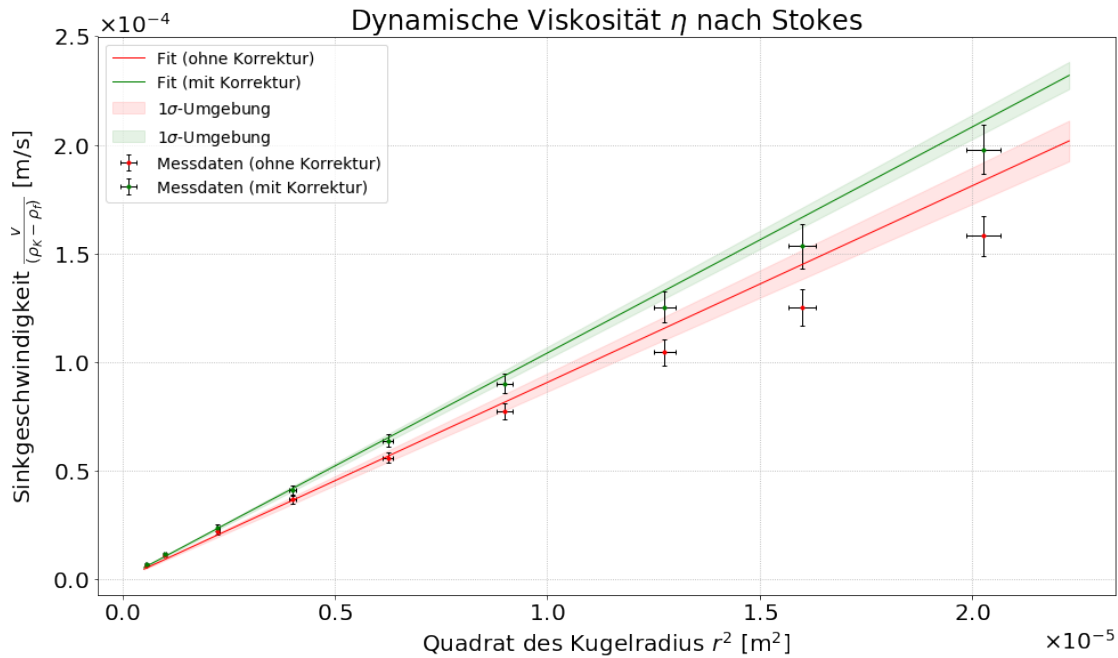
Fitgüte (mit Korrektur):

```

chi_squared= 16.039841031780888
chi_squared_red= 2.004980128972611

```

Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: 4.0 %



Dynamische Viskosität:

(ohne Korrektur): $\eta = 0.241 \pm 0.011$

(mit Korrektur): $\eta = 0.209 \pm 0.006$

$\sigma = 3.476$

$$\eta_{\text{Stokes}} = 0.241 \pm 0.011 \text{ [Pas]} \quad (24)$$

$$\eta_{\text{Stokes}}^{\text{korr}} = 0.209 \pm 0.006 \text{ [Pas]} \quad (25)$$

$$\Rightarrow 3.48\sigma \quad (26)$$

Die zwei Werte für die dynamische Viskosität η sind signifikant voneinander verschieden. Daraus lässt sich schließen, dass die Ladenburg-Korrektur sicher relevant für die spätere Analyse sein wird. Außerdem ist die Geringe Fitgüte auffällig, nach der das verwendete lineare Modell sich nicht wirklich anwenden lässt. Wir vermuten das dies mit der turbulenz der Strömung um die Kugel zu tun hat, wir wollen daher im folgenden Abschnitt die Reynoldszahl berechnen und sehen ob diese an irgendeinem Punkt den theoretischen kritischen Wert von $Re_{\text{kr}} \approx 1$ überschreitet.

4.1.6 Reynoldszahl Re

Die Reynoldszahl ist Systemabhängig und wird deshalb für jede Kugel einzeln berechnet nach:

$$Re = \frac{\rho_f \bar{v} d}{\eta} \quad (27)$$

$$\Delta Re = \sqrt{\left(\frac{\bar{v} d}{\eta} \cdot \Delta \rho_f\right)^2 + \left(\frac{\rho_f d}{\eta} \cdot \Delta \bar{v}\right)^2 + \left(\frac{\rho_f \bar{v}}{\eta} \cdot \Delta d\right)^2 + \left(\frac{\rho_f \bar{v} d}{\eta^2} \cdot \Delta \eta\right)^2} \quad (28)$$

```
[8]: def Reynolds(v,dv,eta,deta):
    Re = rho_Fl*v*d/eta
    dRe = np.sqrt(((v*d/eta)*drho_Fl)**2
                  + ((rho_Fl*d/eta)*dv)**2
                  + ((rho_Fl*v/eta)*d_err)**2
                  + ((rho_Fl*v*d/eta**2)*deta)**2)

    return Re, dRe

Re, dRe = Reynolds(v,dv,eta,deta)
print('\nReynoldszahl:\n')
for i in range(0,len(Re),1):
    print('Re_'+str(i+1)+' = ', format_e(Re[i]), '±', format_e(dRe[i]))
print('')
```

Reynoldszahl:

```
Re_1 = 1.144006e-02 ± 8.224682e-04
Re_2 = 2.403218e-02 ± 1.580096e-03
Re_3 = 7.171675e-02 ± 5.719676e-03
Re_4 = 1.597071e-01 ± 1.154876e-02
Re_5 = 3.042695e-01 ± 1.947504e-02
Re_6 = 5.03419e-01 ± 3.42233e-02
Re_7 = 8.112528e-01 ± 5.986809e-02
Re_8 = 9.942328e-01 ± 8.200918e-02
Re_9 = 1.444384e+00 ± 1.077141e-01
```

Man sieht dass an zwei Stellen der Wert der Reynoldszahl dem theoretisch kritischen Wert sehr nahe kommen bzw. gar überschreiten. An dieser Stelle muss davon ausgegangen werden dass hier die Umströmung der Kugel von laminar in turbulent übergeht. Somit unser Modell (Stokes) für die Berechnung der dynamischen Viskosität an Gültigkeit verliert.

4.1.7 Bestimmung der Kritischen Reynoldszahl Re_{kr}

Um die Sache genauer zu untersuchen wollen wir den genauen Punkt wissen an dem die Strömung in unserem Fall von laminar in turbulent übergeht. Wir suchen dazu einen Punkt, an dem die theoretische Laminargeschwindigkeit v_{lam} verglichen mit der von uns indirekt gemessenen, korrigierten (Ladenburg) Geschwindigkeit v_{kor} kleiner ist. Das bedeutet nämlich dass die Kugel langsamer fällt als sie sollte, es treten dann höhere Reibungskräfte auf, als vom Laminarstrommodell vorhergesagt, es muss also Turbulenzen geben, welche die Kugel zusätzlich verlangsamen.

```
[9]: def v_lam(d,d_err,rho_K,drho_K):
    r, dr = d/2, d_err/2
    v_lam = 2/9*g*(rho_K - rho_Fl)/eta *r**2
    dv_lam = np.sqrt(((2/9*g/eta *r**2)*drho_Fl)**2
                    + ((2/9*g/eta *r**2)*drho_K)**2)
```

```

        +((2/9*g*(rho_K - rho_Fl)/eta**2 *r**2)*deta)**2
        +((4/9*g*(rho_K - rho_Fl)/eta *r)*dr)**2)
    return v_lam, dv_lam
v_lam, dv_lam = v_lam(d,d_err,rho_K,drho_K)

v_v_lam = v_korr/v_lam
dv_v_lam = np.sqrt((1/v_lam *dv_korr)**2
                    +(v_korr/v_lam**2 * dv_lam)**2)

def fit_func(p, x):
    (s,a) = p
    return s*np.log(x)+a
breakpoint = 7

popt1, perr1 = fit(Re[0:breakpoint+1],dRe[0:breakpoint+1],v_v_lam[0:
    ↪breakpoint+1],dv_v_lam[0:breakpoint+1],
    'g','laminare Strömung',[1,1],fit_func,1.2)
popt2, perr2 = fit(Re[breakpoint:],dRe[breakpoint:],v_v_lam[breakpoint:
    ↪],dv_v_lam[breakpoint:],
    'r','turbulente Strömung',[1,1],fit_func,1.2)

def intersect(m1,m2,b1,b2,fm1,fm2,fb1,fb2):
    xi = np.exp((b1-b2) / (m2-m1))
    dxi = np.sqrt((xi/(m2-m1)*fb1)**2+(xi/(m2-m1)*fb2)**2
                  +((b1-b2)*xi/((m2-m1)**2)*fm2)**2
                  +((b2-b1)*xi/((m2-m1)**2)*fm1)**2)

    yi = m1 * xi + b1
    return xi, dxi

Re_kr, dRe_kr = intersect(popt1[0],popt2[0],popt1[1],popt2[1],
                          perr1[0],perr2[0],perr1[1],perr2[1])

print()
plt.title('Bestimmung der kritischen Reynoldszahl ${Re}_{kr}$')
plt.axvline(Re_kr,ls=':',label='${Re}_{kr}$ ='+form(Re_kr,dRe_kr))
plt.xscale('log')
plt.grid(which='major')
plt.grid(which='minor', linestyle=':')
plt.xlabel('${Re}$')
plt.ylabel('${v}/{v}_{lam}$')
plt.legend(loc='best',fontsize=14)
plt.show()

print('\nKritische Reynoldszahl:\n')
print('Re_kr =',format_e(Re_kr), '±', format_e(dRe_kr))
print('Re_kr_theo =', form(1,0))
print('sigma', sig(Re_kr,dRe_kr,1,0))
print('')

```

Fitgüte laminare Strömung:

```

chi_squared= 0.5300799228166498
chi_squared_red= 0.08834665380277497

```

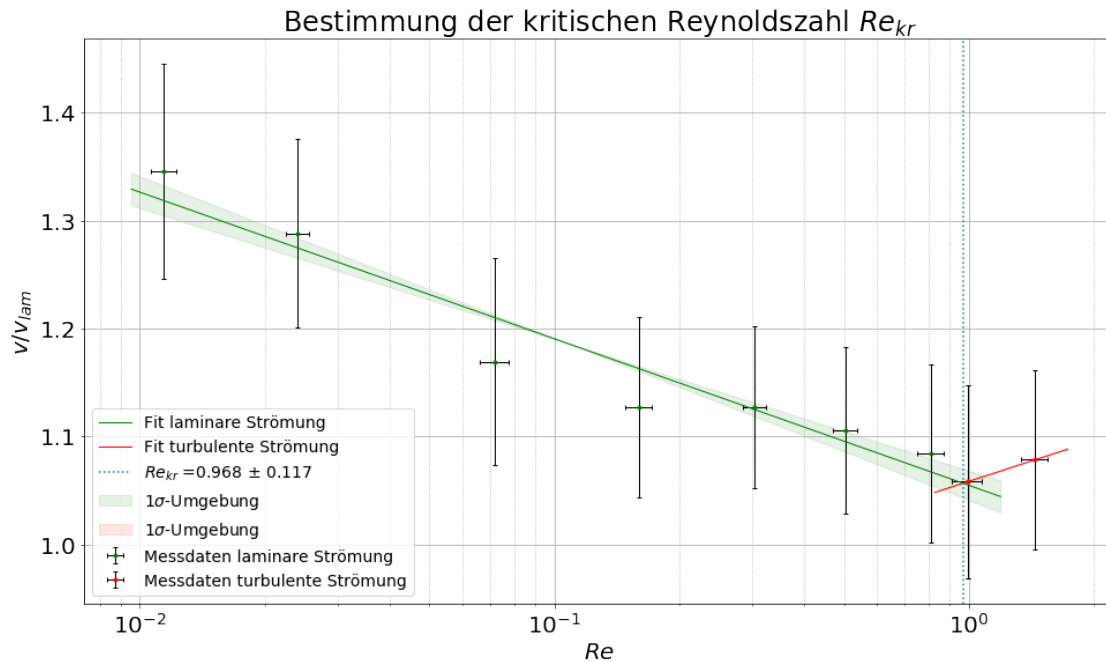
Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: 100.0 %

Fitgüte turbulente Strömung:

chi_squared= 6.1752411766232385e-30

chi_squared_red= inf

Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: 100 %



Kritische Reynoldszahl:

$Re_{kr} = 9.679896e-01 \pm 1.17052e-01$

$Re_{kr_theo} = 1.000 \pm 0.000$

sigma 0.075

$$Re_{kr} = 0.968 \pm 0.117 \quad (29)$$

$$Re_{kr}^{theo} = 1.000 \pm 0.000 \quad (30)$$

$$\Rightarrow 0.075\sigma \quad (31)$$

Zu erkennen ist dass die Kugeln mit Radius $R = 9$ und 7 [mm] außerhalb des laminaren Bereiches sind. Dies hat allerdings zur Folge dass die in Teil 4.1.5 Berechnete Viskosität an zwei Werten verfälscht worden ist. Wir wollen diese also noch einmal untersuchen um zu sehen ob diese Verfälschung aufgrund der turbulenten Strömung signifikant ist oder nicht.

4.1.8 Dynamische Viskosität η nach Stokes ohne Turbulenzen.

```
[52]: x_re, dx_re = x[:breakpoint], dx[:breakpoint]
      y_re, dy_re = y_korr[:breakpoint], dy_korr[:breakpoint]

      def fit_func(p, x):
          (s) = p
          return s*x

      popt, perr = fit(x,dx,y,dy,
                      'r','(ohne Korrektur)', [1],fit_func,1.1)
      popt_korr, perr_korr = fit(x,dx,y_korr,dy_korr,
                                'g','(mit Korrektur)', [1],fit_func,1.1)
      popt_re, perr_re = fit(x_re,dx_re,y_re,dy_re,
                             'royalblue','(ohne Turbulenz)', [1],fit_func,1.1)

      eta_re, deta_re = Eta(popt_re, perr_re)
      print('')
      plt.ticklabel_format(axis='both', style='sci', scilimits=(0,5), useMathText=True)
      plt.title('Dynamische Viskosität  $\eta$  nach Stokes')
      plt.xlabel('Quadrat des Kugelradius  $r^2$  [m $^2$ ])')
      plt.ylabel('Sinkgeschwindigkeit  $\frac{v}{(\rho_K - \rho_f)}$  [m/s]')
      plt.grid(ls=':')
      plt.legend(loc='best',fontsize=14)
      plt.show()

      print('\nDynamische Viskosität:\n')
      print('(mit Korrektur):   eta = ', form(eta_korr,deta_korr))
      print('(ohne Turbulenz):   eta = ', form(eta_re,deta_re))
      print('sigma =', sig(eta_re,deta_re,eta_korr,deta_korr))
```

Fitgüte (ohne Korrektur):

chi_squared= 47.1186244178148

chi_squared_red= 5.88982805222685

Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: 0.0 %

Fitgüte (mit Korrektur):

chi_squared= 16.039841031780888

chi_squared_red= 2.004980128972611

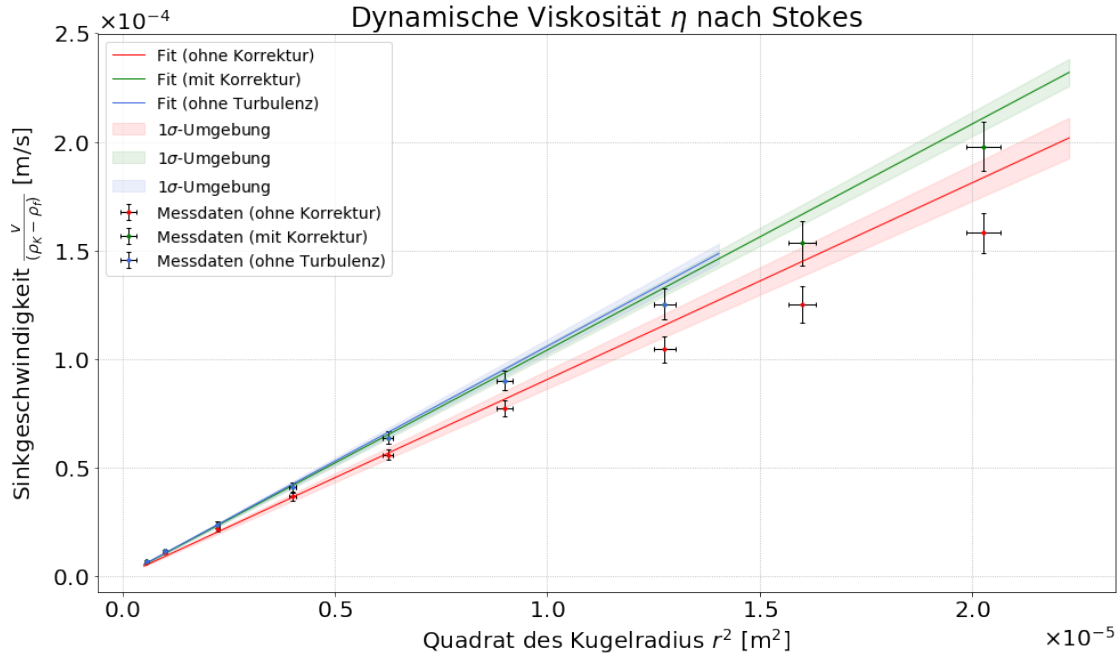
Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: 4.0 %

Fitgüte (ohne Turbulenz):

chi_squared= 12.782844655015982

chi_squared_red= 2.1304741091693304

Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: 5.0 %



Dynamische Viskosität:

(mit Korrektur): $\eta = 0.209 \pm 0.006$

(ohne Turbulenz): $\eta = 0.206 \pm 0.006$

$\sigma = 0.083$

Die Fitgüte hat sich nur minimal verbessert von 4.0% auf 5.0%. es scheint also weitere Parameter geben die unser Modell unvollständig machen, darauf soll in der Diskussion noch konkreter eingegangen werden. Der Wert für die erhaltene Viskosität hat sich nicht großartig verändert.

4.2 Teil 2: Kapillarviskosimeter

Nach dem Gesetz von Hagen-Poiseuille ergibt sich für die Durchflussmenge Q durch ein Rohr:

$$\frac{dV}{dt} = Q = \int_0^R 2\pi r \cdot v(r) dr \quad (32)$$

$$= \frac{\pi(p_1 - p_2)R^4}{8\eta L} \quad (33)$$

$$= \frac{\pi\rho_f g(h_1 - h_2)R^4}{8\eta L} \quad (34)$$

$$\Rightarrow \eta = \frac{\pi\rho_f g(h_1 - h_2)R^4}{8QL} \quad (35)$$

4.2.1 Berechnen der Durchflussmenge Q

```
[63]: V = np.array([6,10,15,20,25,30]) *1e-6 # abgetropftes Volumen [m^3]
dV = np.full(V.size,1) *1e-6

t2 = np.array([182.82,289.40,433.36,621.56,731.22,921.68]) # Zeit [sec]
dt2 = np.full(t2.size,0.5)

R = 1.5 *1e-3 /2 # Kapillarradius [m]
dR = 0.01 *1e-3 /2

L = 100 *1e-3 # Kapillarlänge [m]
dL = 0.5 *1e-3

h1 = 490 *1e-3 #Anfangshöhe [m]
dh1 = 2 *1e-3

h2 = 48 *1e-3 # Endhöhe [m]
dh2 = 2 *1e-3

h_mean = (h1 + h2)/2
dh_mean = np.sqrt((dh2)**2+(dh1)**2)

def fit_func(p, x):
    (s,a) = p
    return s*x+a

popt, perr = fit(t2,dt2,V,dV,'r','Durchflussmenge',[1,1],fit_func,1.1)
Q, dQ = popt[0], perr[0]
print()
plt.ticklabel_format(axis='both', style='sci', scilimits=(0,3), useMathText=True)
plt.grid(ls=':')
plt.title('Viskosität nach Hagen-Poiseuille')
plt.xlabel('Zeit $t$ [sec]')
plt.ylabel('$V$ [$m^3$]')
plt.legend(loc='best')
plt.show()

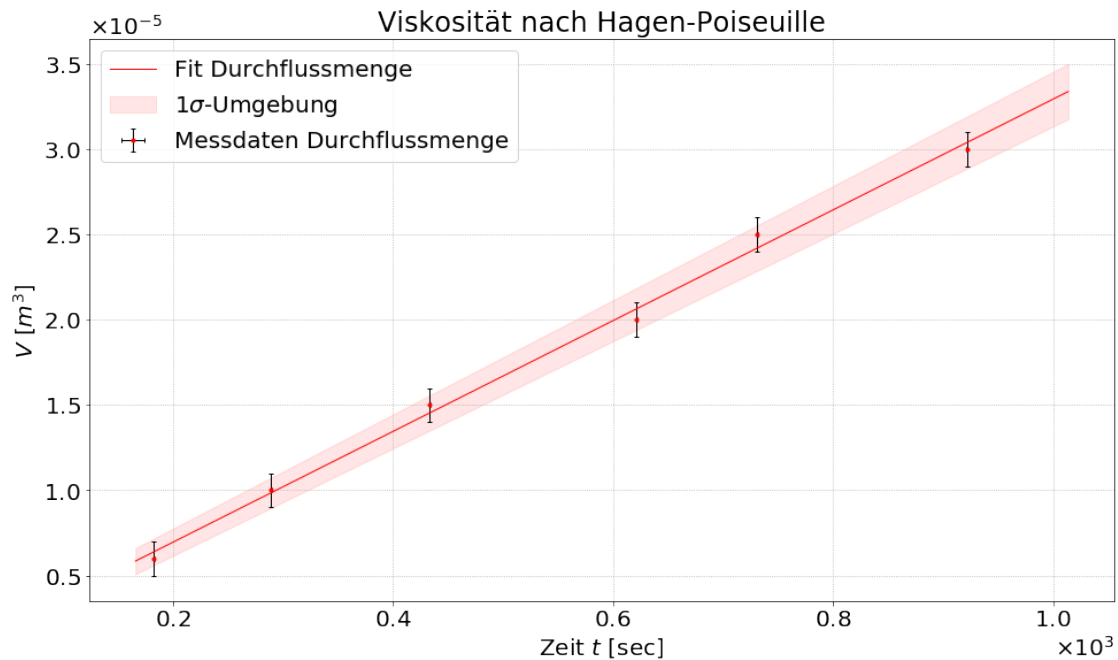
eta2 = np.pi*(rho_Fl*h_mean*g)*(R**4)/(8*L*Q)
deta2 = np.sqrt((((np.pi*(h_mean*g)*(R**4)/(8*L*Q))*drho_Fl)**2)
            +((np.pi*(rho_Fl*g)*(R**4)/(8*L*Q))*dh_mean)**2
            +((4*np.pi*(rho_Fl*h_mean*g)*(R**3)/(8*L*Q))*dR)**2
            +((np.pi*(rho_Fl*h_mean*g)*(R**4)/(8*L**2*Q))*dL)**2
            +((np.pi*(rho_Fl*h_mean*g)*(R**4)/(8*L*Q**2))*dQ)**2)

print('\nDurchflussmenge:\n')
print('Q =', format_e(eta2), '+', format_e(deta2), ' [m^3/sec]')
print('')
```

Fitgüte Durchflussmenge:

```
chi_squared= 1.594228902355326
chi_squared_red= 0.3985572255888315
```

Wahrscheinlichkeit ein größeres oder gleiches Chi-Quadrat zu erhalten: 81.0 %



Durchflussmenge:

$Q = 2.095261e-01 \pm 8.735877e-03 \text{ [m}^3\text{/sec]}$

4.2.2 Berechnung der dynamischen Viskosität η nach Hagen-Poiseuille

```
[64]: print('\nViskosität nach Hagen-Poiseuille:\n')
      print('eta_HP =', form(eta2,deta2))
      print('eta_Stokes =', form(eta_re,deta_re))
      print('sigma', sig(eta2,deta2,eta_re,deta_re))
      print('')
```

Viskosität nach Hagen-Poiseuille:

```
eta_HP = 0.210 ± 0.009
eta_Stokes = 0.206 ± 0.006
sigma 0.057
```

$$\eta_{\text{HP}} = 0.210 \pm 0.009 \text{ [Pa s]} \quad (36)$$

$$\eta_{\text{Stokes}} = 0.206 \pm 0.006 \text{ [Pa s]} \quad (37)$$

$$\Rightarrow 0.057\sigma \quad (38)$$

Die Ergebnisse stimmen sehr gut überein und sind mit einer σ -Abweichung von $0.27 < 3$ nicht signifikant voneinander verschieden.

4.2.3 Bestimmung der Reynoldszahl Re_{Ka} in der Kapillare

```
[57]: Re2 = 2*rho_Fl*Q/(np.pi*R*eta2)
      dRe2 = np.sqrt(((2*Q/(np.pi*R*eta2))*drho_Fl)**2
                    + ((2*rho_Fl/(np.pi*R*eta2))*dQ)**2
                    + ((2*rho_Fl*Q/(np.pi*R**2*eta2))*dR)**2
                    + ((2*rho_Fl*Q/(np.pi*R*eta2**2))*deta2)**2)

      print('\nReynoldszahl Kapillare:\n')
      print('Re_Ka = ', form(Re2, dRe2))
      print('Re_Ka_kr = ', form(2300, 0))
      print('sigma = ', sig(Re2, dRe2, 2300, 0))
      print()
```

Reynoldszahl Kapillare:

```
Re_Ka = 0.148 ± 0.008
Re_Ka_kr = 2300.000 ± 0.000
sigma = 87740955615.637
```

Die von uns gemessene Reynoldszahl für das verwendete Kapillarsystem beträgt $Re_{Ka} \approx 0.15 \ll Re_{kr} \approx 2300$. Die Strömung verhält sich somit mit hoher Sicherheit laminar.

5 Diskussion

In diesem Versuch wurden zwei verschiedenen Messmethoden zur Bestimmung der dynamischen Viskosität η , untersucht. Dabei wurden zwei Fluidmechanische Systeme realisiert, einmal der Fluss um eine Kugel, und zum anderen, der Fluss durch eine Rohr, in beiden Fällen war zu überprüfen ob der Fluss laminar oder turbulent gewesen ist. Und diese dann mit den kritischen Reynoldszahlen des jeweiligen Systems zu vergleichen.

5.1 Reynoldszahlen beim Kugelfallviskosimeter

Bei dem Kugelfallviskosimeter sind die Reynoldszahlen bis auf in zwei Fällen kleiner als die kritische Reynoldszahl. Dies wurde in Teil 4.1.7 genauer besprochen. Die kritische Reynoldszahl die wir erhalten haben ist:

$$\begin{aligned} Re_{kr} &= 0.968 \pm 0.117 \\ Re_{kr}^{\text{theo}} &= 1.000 \pm 0.000 \\ &\Rightarrow 0.075\sigma \end{aligned}$$

Somit ist in 7 von 9 Fällen der Fluss größtenteils laminar. Die zuvor erhaltene Viskosität (Teil 4.1.5) musste somit korrigiert werden. Diese Korrektur wurde in Teil 4.1.8 durchgeführt.

Wir erhalten Folgende Werte:

$$\begin{aligned} \eta_{\text{Stokes}} &= 0.241 \pm 0.011 \text{ [Pas]} && (\text{Stokes}) \\ \eta_{\text{Stokes}}^{\text{korr}} &= 0.209 \pm 0.006 \text{ [Pas]} && (\text{Ladenburg}) \\ &\Rightarrow 3.48\sigma \end{aligned}$$

Die Korrektur der Turbulenz erbrachte stattdessen folgenden Wert:

$$\begin{aligned} \eta_{\text{Stokes}}^{\text{korr}} &= 0.209 \pm 0.006 \text{ [Pas]} && (\text{Ladenburg}) \\ \eta_{\text{Stokes}}^{\text{turb}} &= 0.206 \pm 0.006 \text{ [Pas]} && (\text{ohne Turbulenz}) \\ &\Rightarrow 0.08\sigma \end{aligned}$$

Die Verfälschung der dynamischen Viskosität durch mögliche Turbulenzen ist nicht signifikant.

5.2 Fitgüten beim Kugelfallviskosimeter

Ziel war es unter anderem durch die Betrachtung der Turbulenzen die Fitgüte des Linearen Modells aus Teil 4.1.5 zu verbessern, dies ist nur Teils gelungen. Sie ist von 4.0% auf 5.0% gestiegen. Daraus lässt sich folgern dass das Modell immer noch nicht akkurat den Verlauf der Daten erklären kann. Eine sehr einschränkende Bedingung des Modells ist dabei, dass die Fitgerade durch den Ursprung gehen muss (um physikalisch zu bleiben). Eventuell verhalten sich die kleineren Kugeln aber auch verschieden, da kann es zum Beispiel sein, dass immernoch winzige Luftblasen an der Oberfläche haften bleiben, die für zusätzlichen Auftrieb sorgen. Ignoriert man diese Bedingung lassen sich Fitgüten im Bereich von 50 – 99% erhalten. Daraus lässt sich schließen dass das von uns verwendete Modell zu einfach ist um die reale Bewegung der Kugeln in der Flüssigkeit akkurat zu beschreiben.

5.3 Fazit

Die geringen Fitgüten trotz der vielen Korrekturen beim Kugelfallviskosimeter, machen es als Messmethode eher inakurat. Das Modell von Hagen und Poiseuille, ist dabei deutlich sicherer von unseren Messdaten bestätigt worden. Es kann somit mit höherer Sicherheit die Viskosität berechnet werden. Um dies zu untermauern sind weitere Untersuchungen, vor allem am Kugelfallviskosimeter notwendig, dabei sollte man vor allem ein größeres Gefäß verwenden um weniger Turbulenzen zu bekommen und mehr, vor allem größere Kugelradien zu analysieren.

Abschließen kann man noch einmal zusammenfassen dass die beiden Messmethoden im Rahmen ihrer Ungenauigkeiten den selben Wert hervorgebracht haben und somit durchaus davon ausgegangen werden kann dass die tatsächliche dynamische Viskosität von Polyethylenglykol η_{PEG} irgendwo zwischen den beiden Werten, nach dem Stokes'schem- bzw. Hagen-Poiseuille'schem Prinzip liegt wobei der Wert nach Hagen-Poiseuille, etwas aussagekräftiger ist Aufgrund der sicher nicht turbulenten Strömung und der nicht notwendigen Korrekturen.