

WS19/20, PAP2.1, Versuch 223:  
Brown'sche Bewegung

Praktikanten:  
Gerasimov, V. & Reiter, L.

Betreuer:  
Schmidt-Kaler, F.

Versuchsdurchführung:  
12. November, 2019

## Inhaltsverzeichnis

	Seite
<b>1 Einführung</b>	<b>2</b>
<b>2 Versuchsaufbau, Literaturwerte &amp; Vorbereitung</b>	<b>2</b>
<b>3 Durchführung</b>	<b>3</b>
3.1 Probenpräparation . . . . .	3
3.2 Aufnahme einer Bildfolge . . . . .	3
3.3 Temperatur . . . . .	4
3.4 Eichung des Abbildungsmaßstabs . . . . .	4
<b>4 Kurvenanpassung mit Python</b>	<b>5</b>
4.1 Source Code & Input . . . . .	5
4.2 Output . . . . .	8
<b>5 Auswertung</b>	<b>8</b>
<b>6 Fazit</b>	<b>9</b>

## 1 Einführung

In diesem Versuch<sup>1</sup> werden wir die Brown'sche Bewegung von Partikeln suspendiert in Wasser mit einem Mikroskop beobachten und deren statistische Bewegung untersuchen. Durch Vermessen der Teilchenbahn und der Berechnung der pro Zeiteinheit auftretenden mittleren Verschiebung, können wir die Boltzmannkonstante  $k_B$  bestimmen. Eine genaue Bestimmung der Boltzmannkonstante mit Hilfe der Brownschen Bewegung ist nur bei der Beobachtung sehr vieler Einzelschritte möglich und daher im Praktikum aus Zeitgründen nicht möglich. Bei einer sorgfältigen Durchführung ist aber eine Genauigkeit von besser als 10 % möglich.

## 2 Versuchsaufbau, Literaturwerte & Vorbereitung

- Durchlichtmikroskop Motic B1 mit CCD-Kamera
- Kugelförmige Partikel suspendiert in Wasser
- Thermometer
- Objektmikrometer
- Boltzmann-Konstante<sup>2</sup>  $k_B = 1.380649 \times 10^{-23} \text{ J K}^{-1}$  ist als fundamentale Konstante exakt festgelegt und hat besitzt deswegen Unsicherheit.

---

<sup>1</sup>Dr. J.Wagner - Physikalisches Anfängerpraktikum - V. 1.1 Stand 1/2018, Versuch 223

<sup>2</sup><https://physics.nist.gov/cgi-bin/cuu/Value?k>, [Stand 17.1.2019]

### 3 Durchführung<sup>3</sup>

#### 3.1 Probenpräparation

Wir wollen die Brown'sche Bewegung suspendierter Latex-Partikel in Wasser untersuchen. Um diese mit dem Mikroskop beobachten zu können, benötigen wir eine Probenfassung, die einerseits dick genug ist, so dass sich die suspendierten Partikel darin frei bewegen können, andererseits muss diese auch dünn genug sein, damit eine Fokussierung mit dem Mikroskop möglich ist. Um dies zu gewährleisten, werden wir zunächst eine Probenfassung gemäß Abbildung 2 anfertigen: Auf einen Objektträger wird ein doppelseitiges Klebeband aufgebracht, in dessen Mitte zuvor ein Loch ausgestanzt wurde. In diese Öffnung wird die Probenflüssigkeit eingefüllt und mit einem Deckglas verschlossen. Das doppelseitige Klebeband erfüllt dabei zwei Aufgaben: Zum einen vergrößert dieses das Probenvolumen, so dass sich die suspendierten Partikel frei bewegen können, zum anderen dient es zur Abdichtung der Flüssigkeit, wodurch ungewollte Strömungen durch Verdunstungsprozesse unterdrückt werden.

Wir fertigen vor Versuchsbeginn stets eine neue Probe an, schneiden dazu ein Stück doppelseitiges Klebeband passend auf die Größe des Deckglases (**24 mm × 32 mm**) zurecht und stanzen danach mit dem Lodeisen zentrisch ein Loch in das Klebeband (mit Holzunterlage). Anschließend kleben wir das Klebeband mittig auf den Objektträger und entfernen die Abdeckfolie. Wir schütteln die Flasche mit der Probenflüssigkeit gut durch und pipettieren **250 µl** der Probenflüssigkeit in die ausgestanzte Öffnung des Klebebands. Wir werfen die Pipettenspitze nach Gebrauch sofort in den Abfall. Der Durchmesser der Partikel ist auf der Flasche angegeben. Wir notieren diesen Wert, legen das Deckglas auf das doppelseitige Klebeband und drücken es mit einem Papiertuch vorsichtig an. Dabei darf ruhig etwas von der Flüssigkeit herausfließen. Allerdings dürfen sich keine (größeren) Luftblasen in der Flüssigkeit bilden! Danach trocknen die Probe mit einem Papiertuch ab und geben auf die Mitte des Deckglases einen Tropfen Immersionsöl. Wir spannen nun die Probe auf den Mikroskoptisch (Abbildung 1) ein und wählen am Objektivkranz des Mikroskops das Objektiv mit 100-facher Vergrößerung und einer Numerischen Apertur  $NA=1.25$  aus ( $100/1.25$ ).

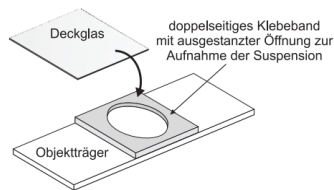


Abbildung 2: Skizze der Probenfassung. Die ausgestanzte Öffnung des doppelseitigen Klebebands wird mit der zu untersuchenden Suspension befüllt und anschließend mit dem Deckglas verschlossen.

#### 3.2 Aufnahme einer Bildfolge

Wir starten vom Desktop aus das Programm **Kamera.exe**. Dieses Programm nimmt in einem festen Zeitabstand ein Bild auf und speichert dieses auf dem Computer. Wir tragen im Messprogramm für den Zeitabstand **1 s** ein und schalten die

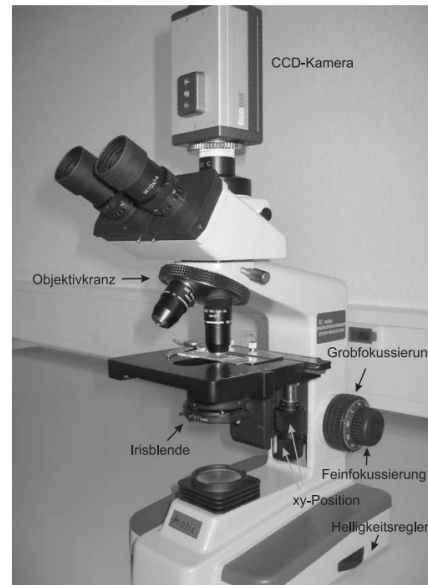


Abbildung 1: Mikroskop

Option „Bilder speichern“ zunächst ab und suchen uns ein Partikel aus, in dessen unmittelbarer Umgebung sich keine anderen Partikel befinden und stellen die xy-Position des Mikroskoptisches so ein, dass sich das ausgewählte Partikel im Zentrum des Mikroskopbildes befindet. Zur Verbesserung des Kontrastes sollten wir die Irisblende am Kondensor auf die Position MIN stellen. Da die Brown'sche Bewegung nicht nur in der xy-Bildebene, sondern auch in z-Richtung erfolgt, wird es passieren, dass das zu beobachtende Partikel aus dem Fokus läuft und somit nicht mehr sichtbar wird. Um dem entgegenzuwirken, muss man die Fokussierung des Mikroskops mit dem Feinregler dauernd nachjustieren. Dies erfordert einiges an Feingefühl und besonders Konzentration. Wir führen zunächst eine Probemessung durch: Damit sich die Probe durch die Mikroskopbeleuchtung nicht zusätzlich aufheizt, drehen die Helligkeit auf das Minimum zurück. Die Kamera ist auch bei dieser Minimalbeleuchtung empfindlich genug, kontrastreiche Bilder zu liefern. Das Programm wird durch Anklicken des Pfeils in der linken oberen Ecke gestartet und wir versuchen der Bewegung eines Partikels über mehrere Minuten auf dem Monitor zu folgen. Sobald das Partikel auch nur leicht unscharf zu erkennen ist, müssen wir sofort mit dem Feintrieb des Mikroskops den Fokus vorsichtig nachstellen. Das Partikel muss während der ganzen Zeit eindeutig erkennbar sein! Wenn wir nun genug Übung im Nachfokussieren erlangt haben und die zuvor genannten Punkte bezüglich der systematischen Fehler berücksichtigt haben, können wir mit der eigentlichen Messung beginnen. Wir stoppen das Messprogramm. Wir schalten die Option „Bilder speichern“ ein und starten wir erneut das Programm. Insgesamt ist jede Sekunde und mindestens **150 mal**, ein Bild aufzunehmen. Nach der Messung entfernen wir die Probe und werfen diese in den Abfall.

#### 3.3 Temperatur

Wir notieren die Zimmertemperatur **T** im Versuchsraum mit Hilfe des Thermometers. Messung wurden dem Versuchsprotokoll (12. Novemberr, 2019) entnommen und nach hier übertragen:

$$T = 22.6(7) ^\circ C$$

#### 3.4 Eichung des Abbildungsmaßstabs

Um später die Position des Partikels ausmessen zu können, müssen wir den Abbildungsmaßstab des Mikroskops bestimmen und benutzen dazu das ausliegende Objektmikrometer. Wir geben einen Tropfen Immersionsöl auf das Objektmikrometer und legen dieses auf den Mikroskoptisch. Wir stellen vorsichtig den Fokus ein und positionieren den Mikroskoptisch so, dass man den Maßstab erkennen kann. Wir beenden das Messprogramm und starte das Programm **Eichung.exe**, optimieren nochmals die Bildschärfe und speichern dann das Eichbild (Abb.3) ab. Wir reinigen das Objektmikrometer mit einem Papiertuch und legen es zurück in die Aufbewahrungsbox. Messung wurden dem Versuchsprotokoll (12. Novemberr, 2019) entnommen und nach hier übertragen:

$$\text{Anzahl der Pixel pro } 20 \mu\text{m} : 222 \pm 3$$

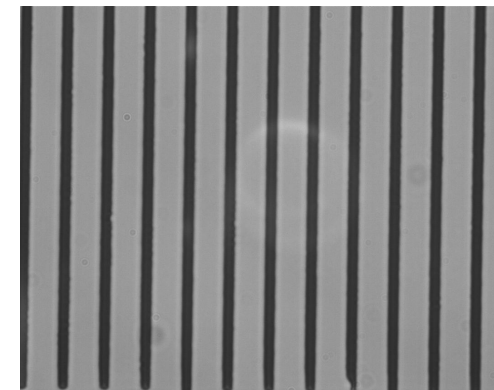


Abbildung 3: Eichbild

<sup>3</sup>Dr. J.Wagner - Physikalisches Anfängerpraktikum - V. 1.1 Stand 1/2018, Versuch 223

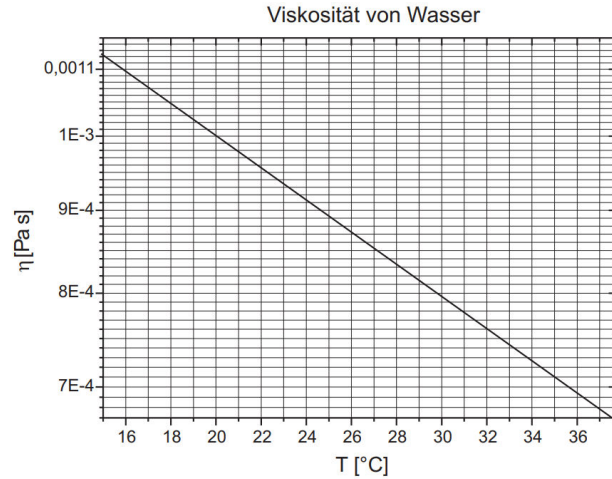


Abbildung 4

## 4 Kurvenanpassung mit Python

### 4.1 Source Code & Input

Für eine der folgenden Rechnungen gehen davon aus, dass die Kumulative Verschiebung linear mit der Zeit zunimmt. Daher ist unser funktionales Modell für die Ausgleichsrechnung wie folgt:

$$\langle r^2 \rangle = \langle \Delta t \rangle S + b \quad (1)$$

So sieht unsere Python-Implementierung aus:

Header:

```
1 %matplotlib inline
2 import matplotlib.pyplot as plt
3 import numpy as np
4 from scipy.stats import norm
5 from decimal import Decimal
6
7 def format_e(n):
8     a = '%e' % Decimal(n)
9     return a.split('e')[0].rstrip('0').rstrip('.') + 'e' + a.split('e')[1]
10
11 def comma_to_float(valstr):
12     return float(valstr.decode('utf-8').replace(',', '.'))
```

Messwerte aus DAT-Datei in den Einheiten [s],[ $\mu m$ ] und [ $\mu m$ ]:

```
1 t, x, y = np.loadtxt('data\Messung2.dat', skiprows=1, usecols=(1, 2, 3), converters= {1:
2     comma_to_float, 2:comma_to_float, 3:comma_to_float}, unpack=True)
```

Diagramm (Abb.5) wird erstellt:

```
1 fig, ax = plt.subplots(1, figsize=[6.4, 4.8])
2 plt.plot(x, y, lw=1, color='C3', marker='s', alpha=.50, label='Messdaten')
3 plt.title('Brownsche Bewegung')
4 plt.xlabel('Position ' + r'$x$' + ' ' + r'$\mu m$')
5 plt.ylabel('Position ' + r'$y$' + ' ' + r'$\mu m$')
6 plt.legend(loc='best')
7
8 fig.savefig('figures/223_Fig1.pdf', format='pdf', bbox_inches='tight')
```

Abgeschätzter Kugelradius  $a$  des betrachteten Teilchens wurde durch nachträgliche Betrachtung der aufgenommenen Bilder in Pixeln bestimmt und danach mit Hilfe der Eichungsdaten

$$a = (8 \pm 2)px = (8 \pm 2)px \frac{20 \mu m}{222 px}$$

auf  $\mu m$  umgerechnet:

```
1 a = 8 * (20/222)
2 a_std = 2 * (20/222)
```

Zimmertemperatur  $T$  wie zuvor gemessen:

```
1 T = 22.6 + 273.15
2 T_std = 0.7
```

Viskosität  $\eta = 9.4(2) \times 10^{-4} Pa s$  von Wasser in uns gegebener Abbildung 4 für  $T = 22.6(7) ^\circ C$  abgelesen:

```
1 eta = 9.4e-4
2 eta_std = 0.2e-4
3
4 dt = np.array([])
5 dx = np.array([])
6 dy = np.array([])
7 i = 0
8 while i < t.size-1:
9     dt = np.append(dt, t[i+1]-t[i])
10    dx = np.append(dx, x[i+1]-x[i])
11    dy = np.append(dy, y[i+1]-y[i])
12    i = i+1
13    r_squared = dx**2+dy**2
14
15    r_squared_mean = np.mean(r_squared)
16    r_squared_mean_std = np.std(r_squared)/np.sqrt(r_squared.size)
17    dt_mean = np.mean(dt)
18    dt_mean_std = np.std(dt)/np.sqrt(dt.size)
19
20    all_data = np.append(dx,dy)
21
22    counts = all_data.size
23    mu = np.mean(all_data)
24    sigma = np.std(all_data)
```

Plot-Umgebung wird angegeben:

```
1 x_fit = np.linspace(-4,4, 1000)
2 binwidth = sigma/2
3 x_bin = np.linspace((min(all_data))-binwidth/2, max(all_data)+binwidth/2, int((max(all_data)-min(
4     all_data))/binwidth+1))
5 gauss = norm.pdf(x_fit, mu, sigma)*counts*binwidth
```

Diagramm (Abb.6) wird erstellt:

```
1 fig, ax = plt.subplots(1, figsize=[6.4 *1.5, 4.8])
2 plt.title('Histogramm der Verschiebungen')
3 plt.xlabel('Bewegung entlang 1 Dimension ' + r'$\Delta x$' + ' bzw. ' + r'$\Delta y$' + ' ' + r'$\mu m$')
4 plt.ylabel('counts')
5 plt.hist(all_data, bins=x_bin, color='C0', label='Histogram der Messdaten')
6 plt.plot(x_fit, gauss, 'C3-', lw=2, label='Gauß-Verteilung:\n' + r'$\mu$' + ' = ' + str(format_plt(mu))
7     + ',\n' + r'$\sigma$' + ' = ' + str(format_plt(sigma)))
8 plt.grid(True)
9 plt.legend(loc='best')
10
11 fig.savefig('figures/223_Fig2.pdf', format='pdf', bbox_inches='tight')
12
13
14 r_kumm = np.cumsum(r_squared)
```

darzustellende Daten werden übergeben:

```
1 x = t[:-1]
2 y = r_kumm
```

Fitfunktion (1) wird deklariert:

```

1 from scipy.optimize import curve_fit
2
3 def fit_func(x,a,b):
4     return a*x+b
5
6 popt, pcov = curve_fit(fit_func, x, y)

```

Angabe welche Sigma-Umgebung der Fitfunktion im Diagramm dargestellt werden soll:

```

1 nstd = 8
2 popt_top = popt+nstd*np.diag(pcov)
3 popt_bot = popt-nstd*np.diag(pcov)

```

Plot-Umgebung wird angegeben:

```

1 x_fit = np.linspace(min(x)-(max(x)-min(x))/10, max(x)+(max(x)-min(x))/10, 1000)
2 fit = fit_func(x_fit, *popt)
3 fit_top = fit_func(x_fit, *popt_top)
4 fit_bot = fit_func(x_fit, *popt_bot)

```

Diagramm (Abb.7) wird erstellt:

```

1 fig, ax = plt.subplots(1, figsize=[6.4*2, 4.8*2])
2 plt.title('Kumulative Verschiebung')
3 plt.grid(True)
4 plt.plot(t[:-1], r_kumm, color='C3', marker='.', linewidth=0, label='Messdaten')
5 plt.plot(x_fit, fit, 'k', lw=1, label='Fit')
6 ax.fill_between(x_fit, fit_top, fit_bot, color='C3', alpha=.25, label=str(nstd)+r'$\sigma$'+
7 'Umgebung')
8 plt.xlabel('Zeit t [s]')
9 plt.ylabel('Summe $\Sigma (r_i^2)$ [$\mu m^2$]')
10 plt.legend(loc='best')
11 fig.savefig('figures/223_Fig3.pdf', format='pdf', bbox_inches='tight')

```

Auslesen der Messergebnisse und Umwandlung der Abhängigkeit von Zeit zur Abhängigkeit vom Weg:

```

1 s = popt[0]
2 s_std = pcov[0][0]

```

Wir wollen die Boltzmann-Konstante  $k_B$  auf zwei verschiedene Weisen berechnen. Zuerst durch die Annahme, dass die verstrichene Zeit und das durchschnittliche Quadrat der Verschiebung proportional zusammenhängen, also:

$$k_{B,1} = \frac{6\pi \langle r^2 \rangle a \eta}{4T \langle \Delta t \rangle} \quad (2)$$

$$\Delta k_{B,1} = k_{B,1} \sqrt{\left(\frac{\Delta \langle r^2 \rangle}{\langle r^2 \rangle}\right)^2 + \left(\frac{\Delta a}{a}\right)^2 + \left(\frac{\Delta T}{T}\right)^2 + \left(\frac{\Delta \langle \Delta t \rangle}{\langle \Delta t \rangle}\right)^2} \quad (3)$$

Berechnung von  $k_{B,1}$  und  $\Delta k_{B,1}$  nach (2) bzw. (3):

```

1 k_1 = 6*r_squared_mean*np.pi*a/(4*T*dt_mean) * 1e-18 * eta
2 k_1_std = k_1*np.sqrt((r_squared_mean_std/r_squared_mean)**2+(a_std/a)**2+(T_std/T)**2+(dt_mean_std/
dt_mean)**2+(eta_std/eta)**2)

```

Und danach durch Bestimmung der Steigung  $S$  einer Regressionslinie die die verstrichene Zeit auf kumulative Verschiebung abbildet (Abb.7):

$$k_{B,2} = S \frac{6\pi a \eta}{4T} \quad (4)$$

$$\Delta k_{B,2} = k_{B,2} \sqrt{\left(\frac{\Delta S}{S}\right)^2 + \left(\frac{\Delta a}{a}\right)^2 + \left(\frac{\Delta T}{T}\right)^2} \quad (5)$$

Berechnung von  $k_{B,2}$  und  $\Delta k_{B,2}$  nach (4) bzw. (5):

```

1 k_2 = 6*s*np.pi*a/(4*T) * 1e-18 * 9.4e-4
2 k_2_std = k_2*np.sqrt((s_std/s)**2+(a_std/a)**2+(T_std/T)**2+(eta_std/eta)**2)

```

Ausgabe der Messergebnisse wird erstellt:

```

1 print('Anzahl der Messungen = ', counts)
2 print('r_squared_mean [(\mu m)^2] = ', format_e(r_squared_mean), ' +- ', format_e(r_squared_mean_std
))
3 print('dt_mean [s] = ', format_e(dt_mean))
4 print('Steigung:')
5 print('S [(\mu m)^2 / s] = ', format_e(s), ' +- ', format_e(s_std))
6 print('Boltzmann-Konstante')
7 print('k_1 [J/K]: '+format_e(k_1)+' +- '+format_e(k_1_std))
8 print('k_2 [J/K]: '+format_e(k_2)+' +- '+format_e(k_2_std))
9

```

## 4.2 Output

```

1 Anzahl der Messungen = 292
2 r_squared_mean [(\mu m)^2] = 1.574166e+00 +- 1.087616e-01
3 dt_mean [s] = 1.000473e+00
4
5 Steigung:
6 S [(\mu m)^2 / s] = 1.559976e+00 +- 3.188454e-05
7
8 Boltzmann-Konstante
9 k_1 [J/K]: 1.698465e-23 +- 4.420316e-24
10 k_2 [J/K]: 1.68395e-23 +- 4.225282e-24

```

Wir erfahren also dass,

Anzahl der Messungen 1-dimensionaler Verschiebungen : 292

$$\langle r^2 \rangle = 1.57(11) \mu m^2$$

$$\langle \Delta t \rangle = 1.000473 s$$

$$S = 1.559976(32) \mu m^2 s^{-1}$$

$$k_{B,1} = 1.70(44) \times 10^{-23} J K^{-1}$$

$$k_{B,2} = 1.69(42) \times 10^{-23} J K^{-1}$$

- Abbildung 5 zeigt uns die Bewegung des beobachteten Teilchens während der gesamten Zeitspanne an.
- Abbildung 6 zeigt ein Histogramm der einzelnen Verschiebungen des Teilchens entlang der x- und y-Achse.
- Abbildung 7 zeigt uns kumulative Verschiebung des Teilchens an, also die genäherte Pfadlänge als Funktion der verstrichenen Zeit an.

## 5 Auswertung

Wir vergleichen die folgenden Messwerte und die definierte Naturkonstante selbst:

$$k_{B,1} = 1.70(44) \times 10^{-23} J K^{-1}$$

$$k_{B,2} = 1.69(42) \times 10^{-23} J K^{-1}$$

$$k_B = 1.380649 \times 10^{-23} J K^{-1}$$

zuerst ist festzustellen, dass unsere Methoden die Boltzmann-Konstante zu berechnen im Prinzip genau das gleiche Problem beschreiben. Sowohl die Länge entlang des Pfades (Kumulierte Verschiebung) als auch die Erwartete Entfernung zwischen Start- und Endposition skalieren linear, sogar proportional mit der Zeit. Egal für welche Rechnung man sich entscheidet wird, das Ergebnis vom diesem Versuch ist deutlich im 1-Sigma-Bereich des Literaturwerts. Das liegt hauptsächlich an dem relativ großen Fehlern:

$$\frac{\Delta k_{B,1}}{k_{B,1}} \approx 39\%$$

$$\frac{\Delta k_{B,2}}{k_{B,2}} \approx 40\%$$

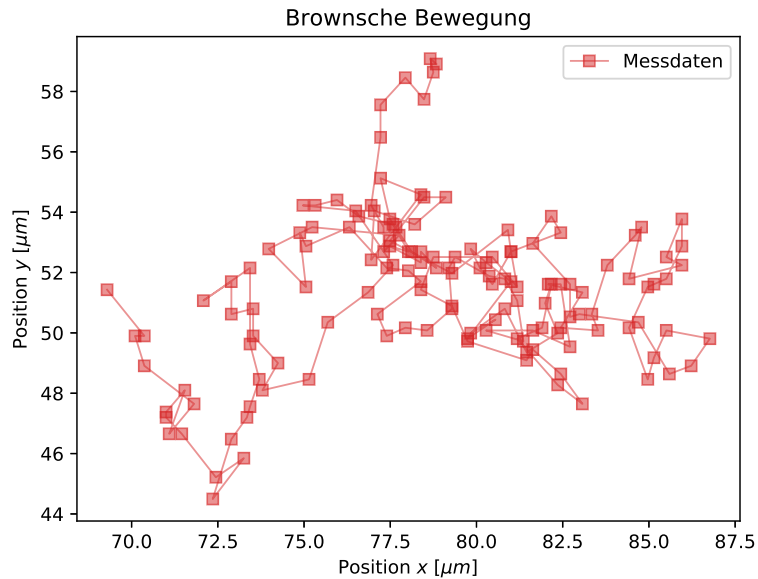


Abbildung 5

Den signifikantesten Anteil am Fehler hat der Teilchenradius  $a$  mit

$$\frac{\Delta a}{a} \approx 25\%$$

D.h. um die Messung deutlich zu verfeinern müssen entweder größere Teilchen betrachtet werden und/oder eine feinere Messung von  $a$  gefunden werden (z.B. Kamera mit besserem Auflösungsvermögen). Unter anderem werden größere Teilchen sich aber langsamer bewegen, was andere Probleme bei der Versuchsdurchführung hat.

## 6 Fazit

Die Wahl verschiedener Parameter bei der Versuchsdurchführung (in diesem Fall z.B. Radius des Teilchens) ist sehr wichtig und kann, das Ergebnis und die Genauigkeit des Versuches stark beeinflussen. Statistische Fehler bleiben immer erhalten. Zusätzliche Parameter bei der Anpassung von Messdaten an ein theoretische Modell, können systematische Fehler korrigieren, wenn sie Begründet sind. Sie können aber auch die Ergebnissenauigkeit fälschlicherweise reduzieren, wenn sie keinem experimentellen oder theoretischen Grund dienen. Trotzdem haben wir mit diesem Versuch die Boltzmann-Konstante  $k_B$  nur auf bis zu den Faktor 1.39 genau messen können und keiner der Messwerte weicht mehr als um 1-Sigma von einander oder dem wahren Wert ab.

$$k_{B,1} = 1.70(44) \times 10^{-23} \text{ J K}^{-1}$$

$$k_{B,2} = 1.69(42) \times 10^{-23} \text{ J K}^{-1}$$

$$k_B = 1.380649 \times 10^{-23} \text{ J K}^{-1}$$

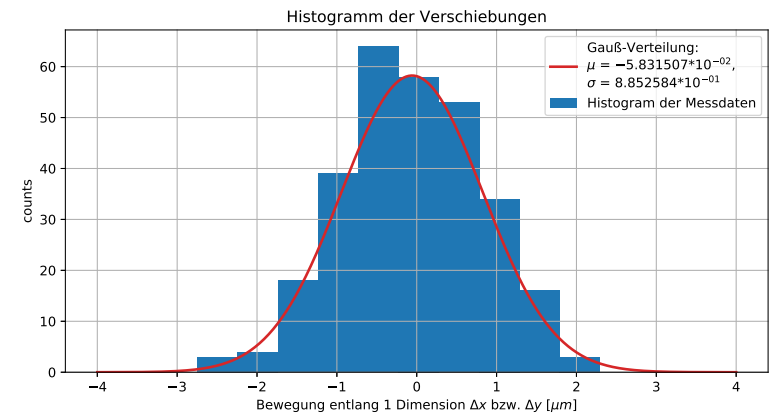


Abbildung 6

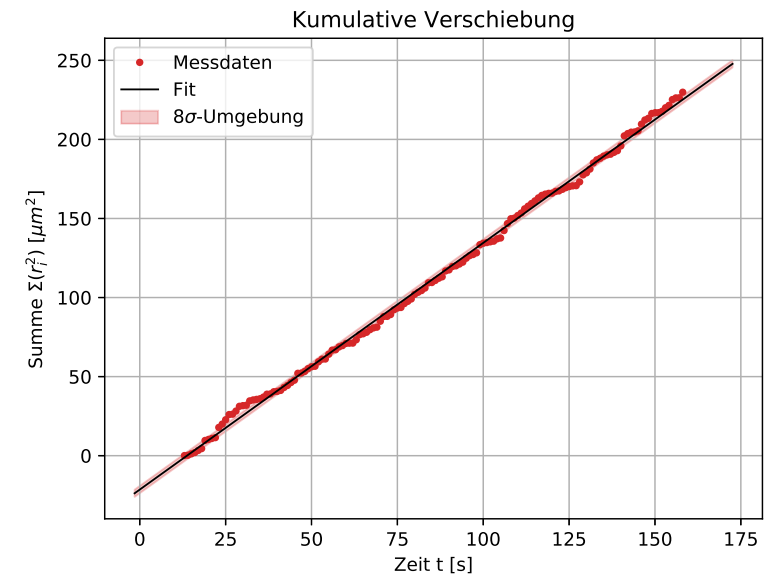


Abbildung 7