Simulation of a Brownian Particle in an Optical Trap

Introduction

Nature displays many examples of what can accurately be considered as multiple, indeed many particle systems with highly complicated dynamics between their elements: atoms in gases, liquids and superfluids; electrons and ions in conductors, semiconductors and plasmas; nucleons in neutron stars and electron-degenerate matter in white dwarfs; even shoals of fish, flocks of birds, traffic, financial markets and human organisations (neglecting their social behavioural aspects)[1],[2]. From a practical point of view, it might appear that modelling such systems is near-impossible, if not very difficult and computationally-expensive at the least. However, from the macroscopic point of view, the cumulative effects of such intricate interactions can be considered in terms of macroscopic variables, simplifying the problem greatly. The details of the motion of individual particles becomes averaged from this perspective, and mean characteristics provide a sufficient description of the dynamics of the system. Furthermore, the deviations of an individual particle from the mean behaviour can often be approximated by some random fluctuation providing a non-deterministic, stochastic evolution [2].

This approach underlies the basic principle explored in this project. Simple stochastic system are quite accessible in terms of their modelling and provide an easy route to exploring stochastic phenomena. One such system is the Brownian particle, a microscopic particle in a fluid. Its movement is described by the phenomenon of Brownian motion, first observed systematically by Robert Brown in 1827 [3]. one of the simplest stochastic processes. The motion of a single particle arises from thermal agitation due to collisions with other particles in the fluid and is related to the temperature and viscosity of the medium. Taking the simulation a step further, a Brownian particle will be considered in an optical trap, adding an optical force to the model. Such a situation is very applicable, with ‘optical tweezers’, a form of optical trap used to manipulate microparticles, being common in experimental physics and having great potential in biophotonics for the study of biological systems. The interplay between the probabilistic and determined forces is central to understanding of Brownian motion in nature, where some determined force, be it gravity or electrostatic interaction, is often present.

The motion is modelled by a Langevin equation [4]: a stochastic differential equation which incorporates a random term to an ordinary differential equation (ODE) [5], in this case, describing an overdamped harmonic oscillator. Thus the equation equates the force on the particle (first lefthand term) to the sum of: a friction force (third righthand term), defined as a friction coefficient multiplied by the particle velocity; an oscillator restoring force (second righthand term), defined as a trap stiffness constant multiplied by the particle position and acting in the opposite direction to the friction force; and a fluctuating white noise component (first righthand term) [2]:

$$m \frac{\partial^2 x(t)}{\partial t^2} = W(t) \sqrt{2 \gamma T k_B} + k x(t) - \gamma \frac{\partial x(t)}{\partial t} \qquad (1)$$

Where m is mass, x(t) is particle position (dependent on time), $\frac{\partial x(t)}{\partial t}$ is particle velocity, $\frac{\partial^2 x(t)}{\partial t^2}$ is particle acceleration, γ is the friction coefficient, k is the trap stiffness constant, t is time, T is temperature, k_B is Boltzmann's constant and W(t) is the random coefficient.

The friction arises from the viscous drag of the fluid in which the particle is suspended, and depends on the viscosity of the medium, which must be large enough that the system is overdamped. The restoring force is the optical force which induces the harmonic oscillation, comparable to Hooke's Law. The white noise term depends on the temperature of the system.

Numerical solutions of such stochastic differential equations is not an easy task as the white noise term causes the solution to be discontinuous almost everywhere. It is achieved in this project by use of a finite difference algorithm: the continuous-time solution x(t) of an ODE is approximated by a discrete-time sequence x_i calculated by the evaluation of the finite difference equation at regular intervals of sufficiently small time steps Δt [2]. The finite difference equation is aquired by substituting into the ODE:

x_i for x(t),

$\frac{x_i - x_{i-1}}{\Delta t}$ for $\frac{\partial x(t)}{\partial t}$ (cf. for velocity v, $v = \frac{\Delta x}{\Delta t} = \frac{x_i - x_{i-1}}{\Delta t}$), (2)

and $\frac{\frac{x_i - x_{i-1}}{\Delta t} - \frac{x_{i-1} - x_{i-2}}{\Delta t}}{\Delta t} = \frac{x_i - 2 x_{i-1} + x_{i-2}}{\Delta t^2}$ for $\frac{\partial^2 x(t)}{\partial t^2}$ (cf. for acceleration a, $a = \frac{\Delta v}{\Delta t} = \frac{v_i - v_{i-1}}{\Delta t}$). (3)

The physical system explored in this project is a silica microparticle in water. The silica microparticle has radius R=1μg and mass m=11pg, and the water has viscosity $\eta = 10^{-3}\,\mathrm{Ns}/m^2$ and temperature T=300K.

The value of the friction coefficient γ will be taken as $6\,\pi\,\eta\,R$, by Stoke's Law (assuming the silica microparticle is a sphere) [6].

Code

```
kb = 1.3806488 × 10^-23;
T = 300;
R = 10^-6;
η = 10^-3;
γ = 6 π η R;
m = 11 × 10^-15;
τ = m/γ;
dif = kb T/γ;
kp = 10^-6;
w := RandomVariate[NormalDistribution[0, 1]];
ϕ[k_] := Print["ϕ=", SetPrecision[1000 * γ/k, 2], " ms"];
Figure1 = Module[{n, k = 100, W, x, sd},
    sim[dt_] := {
      n = 30/dt;
      W := w/√dt ;
      ListPlot[ParallelTable[W, {n}],
       PlotRange → {-8, 8}, DataRange → {0, 30}, AxesLabel → {"t", "W_i"},
```

```
      Epilog → Text[Row[{"Δt=", If[IntegerQ[dt], dt, N[dt] ]}], {4, -7}]],
     x = ParallelTable[NestList[(# + Sqrt[dt] * w) &, 0, n], {k}];
     sd = ParallelTable[StandardDeviation[Table[x[[i, j]], {i, k}]], {j, n}];
     ListPlot[{x[[1]], -sd, sd}, Joined → True, PlotRange → {-8, 8},
      DataRange → {0, 30}, PlotStyle → {Blue, {Gray, Thin}, {Gray, Thin}},
      Filling → {2 → {3}}, AxesLabel → {"t", "x_i"},
      Epilog → Text[Row[{"Δt=", If[IntegerQ[dt], dt, N[dt] ]}], {4, -7}]]};
   GraphicsGrid[Transpose[Table[sim[i], {i, {1, 1 / 2, 1 / 10}}]]],
    ImageSize → Large]];
Figure2 = Module[{dt = 10^-8, nd = 12 000},
   wi = ParallelTable[w, {nd + 1}];
   inertial[rand_] := Quiet[
     {x[i] == x[i - 1] (2 + dt γ/m)/(1 + dt γ/m) - x[i - 2]/(1 + dt γ/m) + rand dt^(3/2) Sqrt[2 kb T γ]/(m (1 + dt * γ/m)), x[-1] == 0, x[0] == 0}];
   noninertial[rand_] := Quiet[{x[i] == x[i - 1] + Sqrt[2 dt dif] rand, x[0] == 0}];
   calibrate[factor_, list_, a_, b_] :=
    Partition[Riffle[Range[a dt, b dt, dt] / τ, factor * list], 2];
   fig2ab[n_, lim_, label_] :=
    ListPlot[{calibrate[10^9,
       Quiet[RecurrenceTable[noninertial[wi[[i]]], x, {i, 0, n}]], 0, n],
      calibrate[10^9, Quiet[RecurrenceTable[inertial[wi[[i]]], x, {i, n}]], 0, n]},
     Joined → True, AxesLabel → {"t/τ", "x [nm]"}, PlotStyle → {{Thin, Black}, Red},
     PlotRange → {-lim, lim}, PlotLabel → label];
   fig2c[rec_, n_, k_] := Module[{X, VACFM, mirror},
     X = ParallelTable[Quiet[RecurrenceTable[rec, x, {i, 0, n}]], {j, k}];
     VACFM = Mean[ParallelTable[AbsoluteCorrelationFunction[Table[Table[
           (X[[j, i + 1]] - X[[j, i]]) / dt, {i, n}], {j, k}][[j]], {n - 1}], {j, k}]];
     mirror = Join[Reverse[VACFM], VACFM];
     calibrate[1, mirror / Max[mirror], -n, n]];
   fig2d[rec_] := Module[{xi, predata},
     xi = RecurrenceTable[rec, x, {i, nd}];
     predata =
      Mean[ParallelTable[Table[(xi[[i + k]] - xi[[i]])^2, {k, nd / 2}], {i, nd / 2}]];
     calibrate[10^18, predata, 0, nd / 2]];
   GraphicsGrid[{{fig2ab[60, 1, "(a)"], fig2ab[6000, 10, "(b)"]}, {
      Module[{n = 540, k = 13, w2},
       w2 = ParallelTable[Table[w, {n + 1}], {k + 1}];
       ListPlot[Quiet[{fig2c[inertial[w2[[j, i]]], n, k], fig2c[noninertial[
            w2[[j, i]]], n, k]}], Joined → True, PlotRange → All, Epilog →
         Inset[Grid[{{Red, Text["Inertial"]}, {Black, Text["Non-inertial"]}}],
          {-5, 0.3}], PlotStyle → {Red, {Black, Thin}}, AxesLabel →
         {"t/τ", Tooltip["C_v(t) [a.u.]", "Arbitrary Units"]}, PlotLabel → "(c)"]],
      ListLogLogPlot[Quiet[{fig2d[inertial[wi[[i]]]], fig2d[noninertial[
           wi[[i]]]]}], Joined → True, PlotRange → All, PlotStyle → {Red, Black},
```

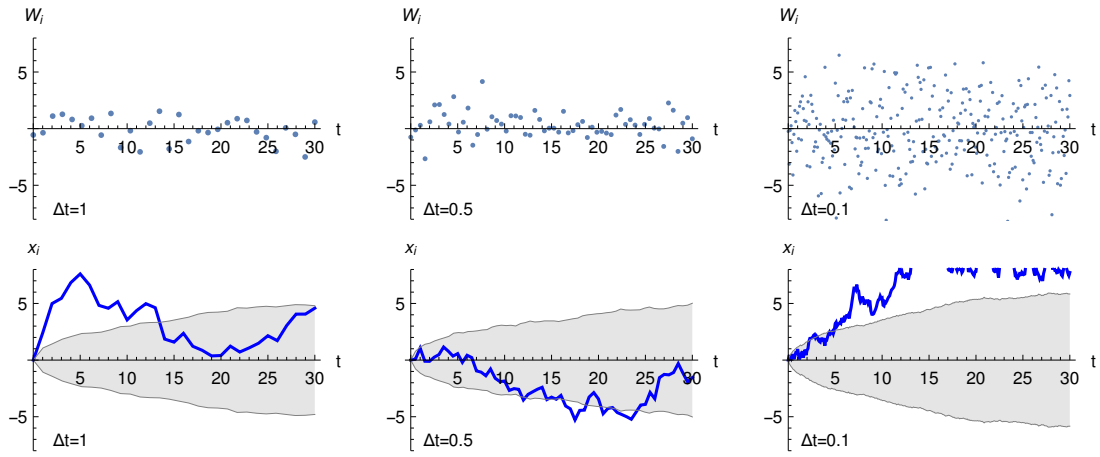
```
        AxesLabel → {"t/τ", "<x(t)^2>[nm^2]"}, PlotLabel → "(d)"]}], ImageSize → 600]];
Module[{n0 = 20 000, n1 = 3000, n2 = 6000, n3 = 15 000, ki = 1,
   p = 15, l = 200, l2 = 250},
  dt2 := γ/(100 Max[vk]);
  vw := Table[{w, w, w}, {Max[n0, n1, n3]}];
  vwi = ParallelTable[vw, {Max[ki, p]}];
  vk = {kp, kp, kp/5};
  vr[0] = {0, 0, 0};
  vr[i_] := {x[i], y[i], z[i]};
  equ[rand_, k_] := vr[i - 1] - 1/γ dt2 k vr[i - 1] + Sqrt[2 dif dt2] rand;
  walk1[n_, rand_, k_] := Table[vr[i] = equ[rand, k], {i, n}];
  walk =
   Mean[ParallelTable[Quiet[walk1[Max[n0, n3], vwi[[j, i]], vk]], {j, ki}]];
  walki[num_, n_] := ParallelTable[walk[[i, num]], {i, n}];
  ProbDist[a_, s_] :=
   SmoothDensityHistogram[10^9 Partition[Riffle[walki[1, n0], walki[a, n0]], 2],
    PlotRange → {{-l, l}, {-s l, s l}}, AspectRatio → s];
  Figure3 = Quiet[Grid[{{
       Graphics3D[{{Opacity[0.1], Ellipsoid[{0, 0, 0}, {200, 200, 400}]},
         {Blue, Line[10^9 walk1[n1, vw[[i]], vk]]}},
        Axes → True, AxesLabel → {"x [nm]", "y [nm]", "z [nm]"}],
       Grid[{{ProbDist[3, 2], "z [nm]"}, {ProbDist[2, 1], "y [nm]"},
         {"x [nm]", Null}}]}}]];
  Var[k_] := Module[{},
    vr[0] = 0;
    vwi1 = ParallelTable[Table[w, {n0}], {Max[{p, ki}]}];
    awalk = ParallelTable[Table[vr[i] = equ[vwi1[[j, i]], k], {i, n2}], {j, p}];
    Mean[ParallelTable[Variance[awalk[[i]]], {i, p}]]];
  stiff[k_, label_, kj_] := Module[{walki2, walk2, vk2},
    vk2 = {k, k, 2 * 10^-7};
    vr[0] = {0, 0, 0};
    walk2 = Mean[ParallelTable[Table[vr[i] = vr[i - 1] -
          dt2 1/γ vk2 vr[i - 1] + Sqrt[2 dif dt2] vwi[[j, i]], {i, n3}], {j, kj}]];
    walki2[num_] := ParallelTable[walk2[[i, num]], {i, n3}];
    SmoothDensityHistogram[10^9 * Partition[Riffle[walki2[1], walki2[2]], 2],
     PlotRange → {{-l2, l2}, {-l2, l2}}, PlotLabel → label]];
  range = Join[Range[2, 8, 2] 10^-7, Range[1, 9, 1] 10^-6];
  data = Table[Var[i], {i, range}];
  simulation = Partition[Riffle[10^6 range, 10^14 data], 2];
  theory = α x^-δ + β;
  sol = FindFit[simulation, theory, {α, β, δ}, x];
  curve = Plot[theory /. sol, {x, 0, 10}, PlotStyle → Blue];
  fig4a = ListPlot[simulation, AxesLabel → {"k_x [fN/nm]", "σ^2_x x10^-4 [nm^2]"},
    PlotStyle → Black, PlotRange → {{0, 10}, {0, 3}},
    Epilog → Inset[Grid[{{Blue, Text["Fitted Line"]},
       {Black, Text["Simulations"]}}], {8.5, 2.7}]];
```

```
  Figure4a = Show[fig4a, curve];
  Figure4b[kj_] := GraphicsGrid[{{stiff[0.2 kp, "(i)", kj],
      stiff[kp, "(ii)", kj], stiff[5 kp, "(iii)", kj]}}, ImageSize → Large];
  equation = SetPrecision[theory /. sol, 3];];
Module[{nd = 3000, k = 10},
  calib[factor_, list_, a_, b_] :=
   Partition[Riffle[factor * Range[-a * dt3, b * dt3, dt3], list], 2];
  fig5a[ki_] := Quiet[Module[{X, PACFM, mirror, n},
     dt3 = γ/(100 ki);
     n = Ceiling[0.5 * 100 * ki/γ];
     wi = Table[w, {n + 1}];
     rec = Quiet[{x[i] ==
         x[i - 1] - (1/γ) ki x[i - 1] dt3 + Sqrt[2 * dt3 * dif] * wi[[i]], x[0] == 0}];
     X = ParallelTable[Quiet[RecurrenceTable[rec, x, {i, 0, n}]], {j, k}];
     PACFM = Mean[ParallelTable[AbsoluteCorrelationFunction[
         Table[Table[X[[j, i]], {i, n}], {j, k}][[j]], {n - 1}], {j, k}]];
     mirror = Join[Reverse[PACFM], PACFM];
     calib[1000, mirror/Max[mirror], n, n]]];
  fig5b[ki_] := Module[{}, Quiet[Mean[Table[
       dt3 = γ/(100 ki);
       wi = Table[w, {nd + 1}];
       xi = RecurrenceTable[{x[i] ==
           x[i - 1] - (1/γ) ki x[i - 1] dt3 + √(2 dt3 dif) * wi[[i]], x[0] == 0}, x, {i, nd}];
       predata = 10^18 * Mean[ParallelTable[Table[(xi[[i + k]] - xi[[i]])^2,
           {k, nd/2}], {i, nd/2}]];
       calib[1000, predata, 0, nd/2], {k}]]];
  Figure5 = GraphicsRow[{
     Quiet[ListPlot[
       {fig5a[0.2 kp], fig5a[kp], fig5a[5 kp]}, Joined → True,
       PlotRange → {{-500, 500}, {-0.4, 1}}, PlotStyle → {Orange, Red, Black},
       AxesLabel → {"t [ms]", "C[x](t) [n.u.]"}]],
     Quiet[ListLogLogPlot[{fig5b[0.2 kp], fig5b[kp], fig5b[5 kp]},
       Joined → True, AxesLabel → {"t [ms]", "<x(t)>^2 [nm^2]"},
       PlotStyle → {Orange, Red, Black}, PlotRange → {{0.5, 100}, {1, 10^5}}]],
     Grid[{{Orange, Text["k=0.2 fN/nm"]}, {Red, Text["k=1 fN/nm"]},
       {Black, "k=5 fN/nm"}}, Alignment → Left]},
    ImageSize → 900, Alignment → Left];];
```

Code Validation and Application

Simulation of White Noise

Figure1



The only undefined term in the Langevin equation (1) is the white noise term W(t). Thus, the first step is to simulate the white noise.

W(t) must be almost everywhere discontinuous, have infinite varition, have zero mean, <W(t)>=0, for all t, $< W(t)^2 > =1$ for each t, and independent $W(t_1)$ and $W(t_2)$ for $t_1 \neq t_2$[1],[2]. One way to achieve this is through the use of a sequence of random Gaussian numbers with zero mean and variance $\frac{1}{\Delta t}$ [2], which were generated using *Mathematica*'s inbuilt functions. If w_i is the sequence of generated random Gaussian numbers with zero mean and unit variance, then rescaling provides W_i:

$$W_i = \frac{w_i}{\sqrt{\Delta t}} \qquad (4)$$

with variance $\frac{1}{\Delta t}$.

This was implemented for three different time steps, Δt= 1, $\frac{1}{2}$ and $\frac{1}{10}$ seconds, and plots graphed for each case, shown as the top three plots in Figure 1 above. They demonstrate that the values of W_i increase (generally) as the time step Δt gets smaller and the ensemble diverges, sa shown in [2].

The simplest version of a free diffusion equation is one of the form: $\frac{\partial x(t)}{\partial t}$=W(t) and has a 'random walk' solution. The finite difference equation corresponding to this, following the process described in the introduction, is:

$$x_i = x_{i-1} + \sqrt{\Delta t}\ w_i \qquad (5)$$

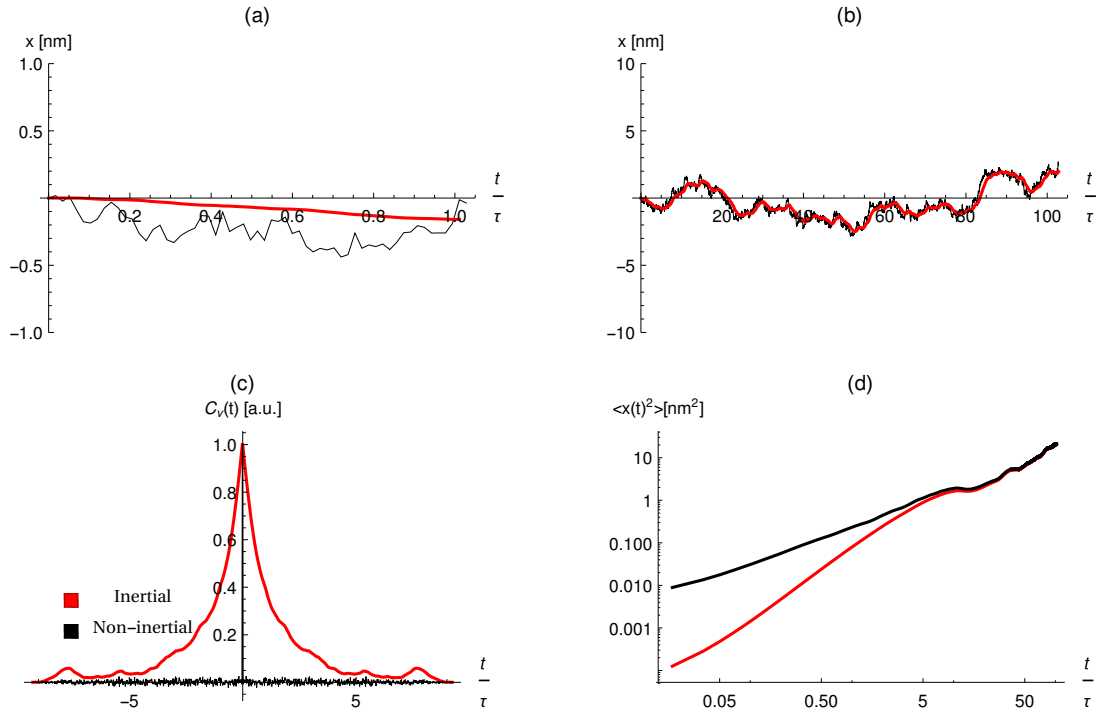
using w_i as previously defined. Using again the same three values for the time step, the free diffusion trajectories x_i are demonstrated in the bottom three plots in Figure 1 above. The x_i trajectories become more jagged as the time step decreases, as shown in [2].

Although these three plots are each different, since they are individual solutions of a random process, if many solutions are averaged over, it is observed that the statistical properties do not change. The shaded areas in the graphs demonstrate this as they are roughly the same: they show the standard deviation about the mean position at each horizontal position by analysing 100 trajectories. So, looking at position n, all the x_n are taken from each calculated trajectory and the standard deviation of the data found. This point is plotted (with its mirror reflection about the x-axis) and the process is repeated for x_{n+1} and so on. The slight differences in each graph are present because only a finite number (100) of trajectories where present in the averaging. They would be the same exactly in the infinite limit of number of trajectories averaged over. The results agree with those obtained in [2], whose graphs are slightly more correlated with each other as they calculte using 10,000 trajectories. In this project, 100 trajectories were considered sufficient to demonstrate the

concept, so in the interest of time (ie. time waiting for the calculation to evaluate) no more were averaged over.

The Regime Transition

Figure2



Before introducing the optical trap, the Brownian motion of an untrapped particle will be investigated.

Brownain motion, the apparently perpetual and random movement of particles suspended in a fluid, can be described as diffusive behaviour [7]. Diffusion occurs as a result of the collisions of the silica microparticle with the water molecules, each collision changing the microparticle velocity, which then moves in the resultant direction until the next collision. Following many such interactions, the motion of the particle becomes effectively randomised. However, this is only applicable at long time scales. [4] At shorter time scales, the dynamics of the Brownian particle is dominated by its inerita, rather than diffusive effects.

Thus, the motion of the particle displays different behaviours depending on the time scale of the observation. For short times, ballistic motion is observed, while at longer time scales, diffusive motion is observed. The ratio τ, the momentum relaxation time [6], is equal to $\frac{m}{\gamma}$ and provides the time scale of the transition from ballistic behaviour to diffusive behaviour [1], so times shorter than or comparable to τ display the ballistic regime, and longer times show Brownian motion characteristic of the diffusive regime. The time τ thus identifies the transition from the ballistic to the diffusive regime.

Equation is manipulated to provide the equations required to model the particle in these regimes.Omitting the optical force term from equation (1) gives the appropriate Langevin equation for ballistic Brownian motion:

$$m \frac{\partial^2 x(t)}{\partial t^2} = W(t) \sqrt{2 \gamma T k_B} - \gamma \frac{\partial x(t)}{\partial t} \qquad (6)$$

By making the appropriate substitutions (2) and (3) into (6) to obtain the finite difference equation and then rearranging for x_i, the appropriate recurrence relation is found for the trajectory (presented in the form given by a FullSimplify):

$$x_i = \frac{\sqrt{2}\,\Delta t^{3/2}\sqrt{T\gamma k_B}\,w_i}{m\left(1+\frac{dt\,\gamma}{m}\right)} - \frac{x_{-2+i}}{1+\frac{\gamma\Delta t}{m}} + \frac{\left(2+\frac{\gamma\Delta t}{m}\right)x_{-1+i}}{1+\frac{\gamma\Delta t}{m}} \qquad (7)$$

This is shown as the red line in Figure 2a and 2b. All trajectories in Figures 2a and 2b were calculated with the same set of random numbers for the white noise term to provide comparable solutions, and a time step in Δt=10ns.

The trajectory calculated from this will depend on the mass of the particle, ie. is inertial, so it is expected that this equation will be accurate at short time scales (time >> τ), ie. the ballistic regime. To simulate the Brownian motion at longer time scales, the dynamics are dominated by diffusion, so it should be possible to drop the inertial term (by setting mass m=0) in equation (6) to obtain, on rearrangement:

$$\frac{\partial x(t)}{\partial t} = \sqrt{\frac{2\,T\,k_B}{\gamma}}\,W(t) \qquad (8)$$

The Einstein relation (9) [7] relates a diffusion coefficient D to the friction coefficient γ by :

$$D = \frac{k_B T}{\gamma} \qquad (9)$$

Thus the noninertial stochastic equation for the Brownian particle becomes:

$$\frac{\partial x(t)}{\partial t} = \sqrt{2D}\;W(t) \qquad (10)$$

Converting this to a finite difference equation yields:

$$x_i = x_{i-1} + \sqrt{2\,D\,\Delta t}\;\;w_i \qquad (11)$$

This trajectory should provide a good approximation to Brownian motion at long time scales (time ≤ τ), ie. the diffusive regime. This is shown as the black lines in Figures 2a and 2b.

Figure 2a compares the noninertial to the inertial trajectory for short time scales. Agreeing with the results observed in [2], the inertial trajectory is smooth while the noninertial trajectory is discontinuous, with a change in direction at every step forming a series of broken line segments. This indicates that the velocity of the inertial solution is well-defined and also changes smoothly, while the noninertial case has not a well-defined velocity. For the short time scales less than the momentum relaxation time τ in Figure 2a, the red line (using inertial equation (7)) shows the most accurate model of the motion of the Brownian particle in the ballistic regime, with inertially-dominated dynamics. As shown, the noninertial trajectory (black line) is not very accurate at this scale.

Figure 2b repeats the comparison for a longer time scale. Both trajectories show characteristic jagged Brownian motion behaviour as microscopic details are not resolvable at this scale as they were in the previous case. The plot demonstrates that the noninertial trajectory (black line) provides a good approximation to Brownin motion, which is more accurately described by the full inertial equation (7) shown by the black line. Thus it is a suitable approximation to drop inertial terms when considering Brownian motion of a particle at time scales longer than the momentum relaxation time. These results agree with [2], and the simulation provides similar results to those obtained in [6].

Figure 2c shows the velocity autocorrelation function (VAF) for the inertial (red) and noninertial cases (black). This statistical quantity relates to the time that it takes for the particle to ‘forget its initial velocity (see explanation of diffusion above) and is described by:

$$C_v(t) = \overline{v(t'+t)\,v(t')} \qquad (12)$$

It was calculated using the built-in Mathematica function AbsoluteCorrelationFunction to perform the discrete finite-step analogue of equation (12):

$C_{v,n} = \overline{v_{i+n}\, v_i}$ (13)

with the overbar representing a time average. In more general terms, the calculation is a measure of the similarity of a function with itself as a function of the time lag between measurements of the function, thus it indicates correlation between the values of the trajectory at different times.

For the inertial trajectory (red line), the function decays to zero in a time scale comparable to the momentum relaxation time τ, showing that the particle velocity becomes uncorrelated with its initial value over this time scale, agreeing with [2].This shows the transition from the ballistic regime to the diffusive regime, where steps have become uncorrelated with each other, occuring at an approximate time of a few τ.

For the noninertial case (black), the function drops to zero immediately, indicating that the velocity is uncorrelated and therefore does not have a charcteristic time scale, again agreeing with [2]. This is not a problem in the diffusive regime (long time scale) as diffusion causes the velocity of the particle to become randomised, but indicates that the noninertial approximation does not hold at small time scales.

The plot was produced by calculating the VAF (for 540 points from 0 to $t/\tau \sim 9$) 13 times and taking the average of the 13 calculations. This was observed to give a result more similar to the one obtained in [2], and could be further improved (getting rid of the oscillation in the tails) by averaging over a greater number of calculations. This was not performed in the interest of computational time. Other methods to improve the VAF are presented in the PCF discussion.

The general behaviour of the VAF graph is also confirmed in [1].

Figure d shows a log-log plot of the mean-square displacement (MSD), quantifying how a particles moves from its initial position. It is defined by:

$\langle x(t)^2 \rangle = \overline{[x(t'+t)\, x(t')]^2}$ (14)

which gives the discrete finite-step form:

$\langle x_n^2 \rangle = \overline{(x_{i+n} - x_i)^2}$ (15)

Taking a 12000 element list of the trajectory, the MSD for the first 6000 points was calculated by subtracting the i^{th} trajectory element from the $(i+j)^{\text{th}}$ element and squaring the result, starting at i=1 and running through the j's (from 1 to 6000), then repeating for i=2, i=3, ..., i=6000 and then taking the average of all the j=1 results, all the j=2 results, etc.

Einstein's description of diffusive Brownian motion predicts a MSD proportional to t [3], while for ballistic motion, the MSD is expected to vary with t^2 [2].

The noninertial calculation, shown by the black line in Figure 4d demonstrates linear behaviour for all times. Thus, this approximation holds for long time scles, but is inaccurate at shorter time scales.

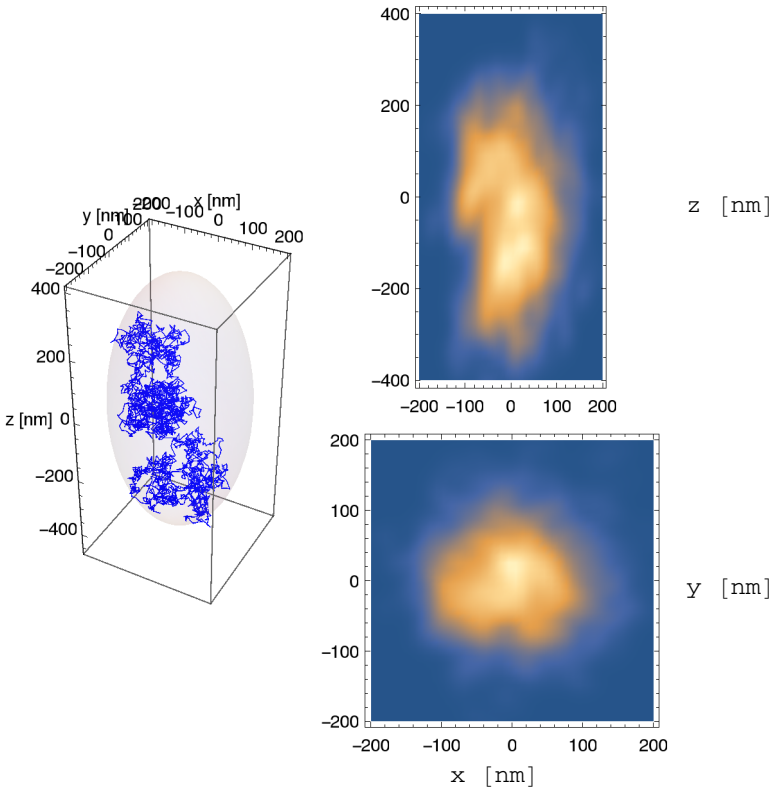
The inertial trajectory (red line, Figure 2d) transitions from quadratic behaviour at short times to linear behaviour at long times, demonstrating the transition from ballistic motion to diffusive motion. The inertial calculation thus holds for all considered time scales. Additionally, the MSD graph demonstrates the transition from quadratic to linear behaviour (for the inertial case), ie. ballistic motion to diffusive motion, occuring at approximately time τ, as expected.

These results agree with those obtained in [2]. The calculation could be done several times and the results averaged to improve the graph: no averaging was performed as it took an unpractically long time to evaluate for this project, so the results lines sometimes wiggle around a bit nearing a time

scale of 100 t/τ. However, this wiggle would likely be eliminated by averaging many calculations to give a more accurate plot like that shown in [2].

Optical Traps

Figure3



Now an optical trap will be introduced to the simulation so that an additional deterministic force acts on the silica bead. The optical trap operates by attracting a transparent particle in a medium of lower refractive index to the position of maximal laser intensity, confining it to the focal spot of a laser by the process of momentum transfer from photons to the particle [8]. A particle outside of the centre of the beam will refract more light in the direction opposing the centre of the beam.The emitted photons exert an equal but opposite force by Newton' s Third Law on the particle, directed towards the middle of the beam. Thus, the particle moves in this direction and is trapped at the point of maximum laser intensity. Furthermore, the focused laser beam(s) can be used to produce three independent harmonic traps along each Cartesian axis, trapping the particle in three dimensions [9].
There exists a dynamic equilibrium between the thermal noise driving the trapped Brownian particle out of the trap and the optical forces confining it. The restoring force acts on the time scale given by the ratio [2]:

$$\phi = \frac{\gamma}{k} \qquad (16)$$

which is generally much greater than r. Thus, a time step At significantly less than 0 (to allow study of the restoring harmonic trap force) can still be sufficiently greater than r such that the noninertial approximation may be employed. Therefore, the particle motion can be described by a set of three independent Langevin equations, each of which is obtained by adding back in the restoring force term to equation (10). This yields:

$\frac{\partial x(t)}{\partial t} = \sqrt{2D}\ W_x(t) - \frac{k_x}{\gamma} x(t)$ (17)

where k_x is the trap stiffness in the x-direction and W_x is a random coefficient for the x-direction, and similarly two more equations in y and z, which can be combined to give the relevant three dimensional Langevin equation:

$\frac{\partial \underline{r}(t)}{\partial t} = \sqrt{2D}\ \underline{W}(t) - \frac{k}{\gamma}\underline{r}(t)$ (18)

where $\underline{W}$ is the random white noise coefficient vector $\begin{pmatrix} W_x \\ W_y \\ W_z \end{pmatrix}$, $\underline{k}$ is the trap stiffness vector $\begin{pmatrix} k_x \\ k_y \\ k_z \end{pmatrix}$ and $\underline{r}$ is the position vector $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ (The first and last being variables of time).

Following the usual technique, this provides the finite difference equation:

$\underline{r}_i = \underline{r}_{i-1} + \sqrt{2D\Delta t}\ \underline{w}_i - \frac{\Delta t}{\gamma}\underline{k}\ \underline{r}_{i-1}$ (19)

Where $\underline{r}_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}$ gives particle position at time t_i, $\underline{w}_i = \begin{pmatrix} w_{i,x} \\ w_{i,y} \\ w_{i,z} \end{pmatrix}$ is the vector of zero mean, unit variance Gaussian random numbers and the $\underline{k}$ and $\underline{r}_{i-1}$ vectors multipliy together element-wise to produce the three dimensional column vector $\begin{pmatrix} k_x \cdot x_{i-1} \\ k_y \cdot y_{i-1} \\ k_z \cdot z_{i-1} \end{pmatrix}$.

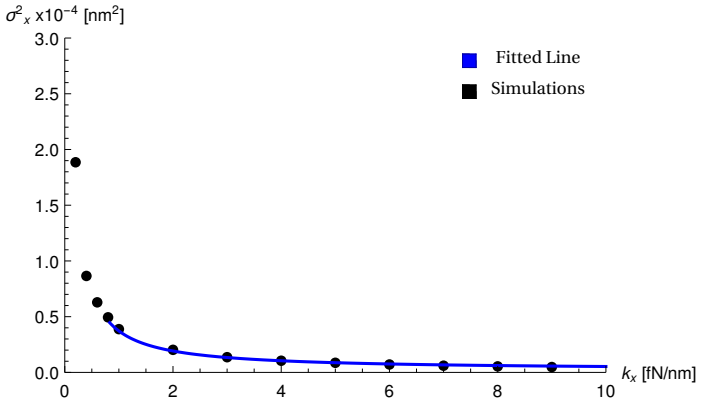
The three-dimensional plot in Figure 3 shows the simulated trajectory of the silica microparticle suspended in water and trapped by an optical trap with stiffness $k_x = k_y = 1$ fN/nm and $k_z = 0.2$ fN/nm for 3000 steps. The shaded ellipsoid traces an equiprobability surface and demonstrates that the decreased trap stiffness in the z-direction allows the particle more freedom of movement in this direction, resulting in a stretching of the equiprobability surface that would be obtained in the isotropic case (ie. a sphere). This is further demonstrated by the probability distributions of finding the particle in the x,y- and x,z-planes as shown on the right of Figure 3.

These distributions appear to follow a two-dimensional Gaussian distribution around the trap centre, agreeing with the results of [2]. The simulated path generated also appears very similar to that obtained in [2]. The probablity distributions are generated using a trajectory which is 20,000 steps long.They could be improved by increasing the length of the trajectory, although the current results sufficiently demonstrate the concept.

On generating averaged probablity distributions using the average for each step i of n trajectories, it was found that the plot became more centralised as n was increased. It was spectulated that the limit of the graph as n tended to infinity would be a dirac-delta function centred at the centre of the trap, demonstrating that the particle is indeed confined about the centre point of the trap. This is explored again below.

Adjusting Trap Stiffness

Figure4a



One of the great advantages of an optical trap is that the stiffness of the trap can easily be adjusted by changing the optical power of the laser(s) [2]. Measuring the variance σ^2_m (m referring to x, y or z) of the particle position about the trap centre in the m-plane provides a method of quantifying the trap stiffness. Holding k_z constant at 0.2 fN/nm and k_y constant at 1 fN/nm, k_x was varied over a range of values between 0.2 and 9 fN/nm, calculating the variance σ^2_x for a 6000 step trajectory 10 times and taking the average of the variations for each trap stiffness value k_x. The calculation was only performed in one dimension as changing trap stiffness in one direction will not affect motion in orthogonal directions. The results are plotted in Figure 4a and demonstrate that the particle becomes more strongly confined (so position varies from centre of trap less) as the trap stiffness is increased. The graph shows that the confinement becomes very sensitive at very low trap stiffnesses.

These results correlate in shape (although not in scale: these results are 10^8times smaller) with the results obtained in [2], where σ^2_{xy} was shown as a function of either k_x or k_{xy}. It is unclear what they are actually dealing with as the text refers first to dealing with k_x ,then k_{xy} for the same thing, and it does not seem to make sense to have a σ^2_{xy} measurement “in the y-plane” - surely this would then be σ^2_y? In addition, assuming k_{xy} refers to both k_x and k_y,a change in k_x does not affect motion in the y-direction, so why bother considering it? Regardless of this misunderstanding and of the scaling discrepancy, the graph in [2] and in this project are of similar shape, so this aspect was investigated in more detail.

(It can only be assumed that in [2], y-plane refers to x,y-plane and that k_{xy} refers to both k_x and k_y. Continuing the assumption, [2] varied both k_x and k_y together and measured the variance in the x,y-plane. This is different from the one-dimensional analysis presented in this project, but both techniques should give similar and equally valid indications of the bahaviour of the system. This also explains the scaling error).

The blue line in Figure 4a shows a best line fitted to the data. The fit assumes that the data forms a function of the form $k_x^{-\delta}\,\alpha+\beta$, where α,β and δ are arbitrary constants. [2] suggests that σ^2_x varies inversely with k_x, which would suggest δ=1. The equation of the line fitted to the simulated data is:

```
Print[equation, "                (20)"]
```

$$0.0203+\frac{0.351}{x^{1.03}}\qquad(20)$$

Where x represents k_x. Thus in this run of the notebook, δ is found to be:

```
result = SetPrecision[δ /. sol, 3]
```

```
1.03
```

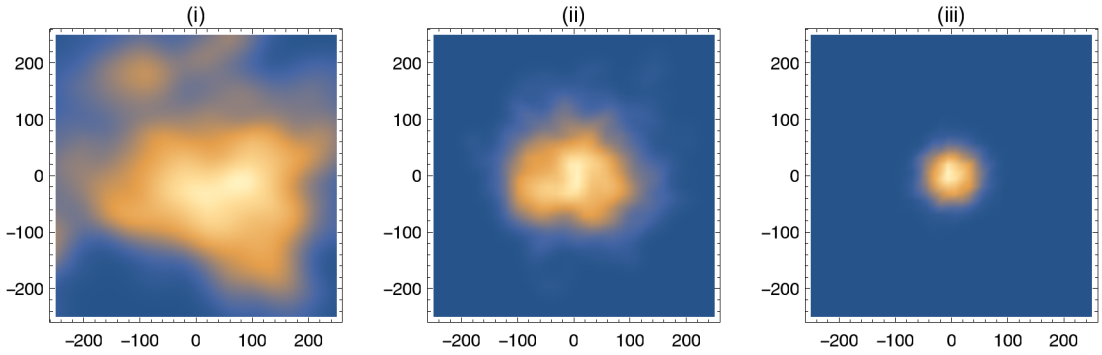
Which, as an approximation, is close to 1 (a different result is obtained each time the notebook is evaluated due to the random term, generally lying between 0.75 and 1, not uncommonly giving 1 as the solution). This δ result as a percentage of 1 is:

```
Print[result * 100, " %"]
```

```
103. %
```

This percentage would likely increase if the proposed steps to increase the accuracy of the simulation were implemented, and the δ value calculated several times and averaged.
This suggests the inversely proportional relationship between variance and trap stiffness (in one direction) holds.
The accurcay of this result could be improved by both increasing the number of steps in each trajectory and increasing the number of trajectories averaged over for each plotted point. However, it would most be improved by calculating and plotting more points on the graph, especially in the region near the origin where it increases most quickly. Also, there is the consideration that some mistake has been made in this project due to a misunderstanding of [2].
The best line does not actually fit every time the notebook is evaluated, only most of the time. If the fit is retrained to a region $1 < k_x < 9$, then the fit is much more successful, but it was decided to leave in the full range fit to try and improve the accuracy of the results.
Considering an x,y-plane cross-section of the three-dimensional system at the origin, the probability distribtutions were calculated to be as shown in Figure 4b, with trap stiffnesses of $k_z = 0.2\,\text{fN/nm}$ and (i) $k_y = k_x = 0.2\,\text{fN/nm}$, (ii) $k_y = k_x = 1\,\text{fN/nm}$ and (iii) $k_y = k_x = 5\,\text{fN/nm}$. These demonstrate that the position distribution shrinks as the trap stiffness is increased, in concordence with the results of [2]. These distributions gain appear to be two-dimensional Gaussian distributions.

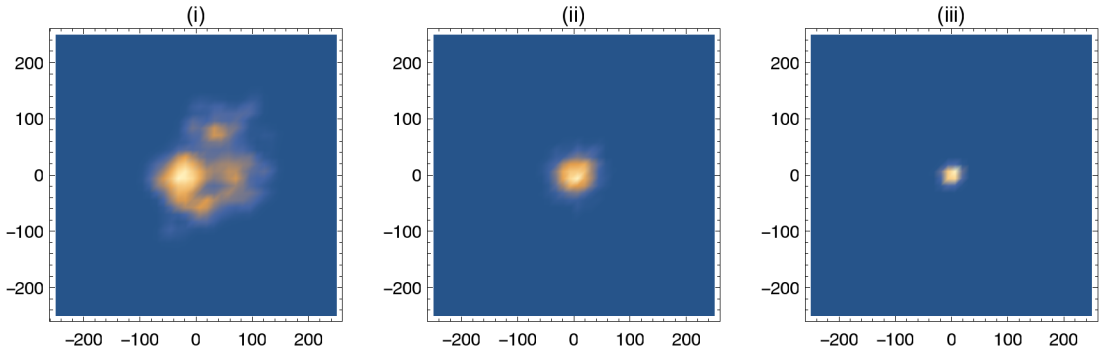
```
Figure4b[1]
```



Considering an x,y-plane cross-section of the three-dimensional system at the origin, the probability distribtutions were calculated to be as shown in Figure 4b, with trap stiffnesses of $k_z = 0.2\,\text{fN/nm}$ and (i) $k_y = k_x = 0.2\,\text{fN/nm}$, (ii) $k_y = k_x = 1\,\text{fN/nm}$ and (iii) $k_y = k_x = 5\,\text{fN/nm}$. These demonstrate that the position distribution shrinks as the trap stiffness is increased, in concordance with the results of [2]. These distributions again appear to be two-dimensional Gaussian distributions, or at least, they would be in the limit of an infinite number of steps in the calculated trajectory. These graphs are calculated from a trajectory with 15,000 steps; the number of steps is the limiting

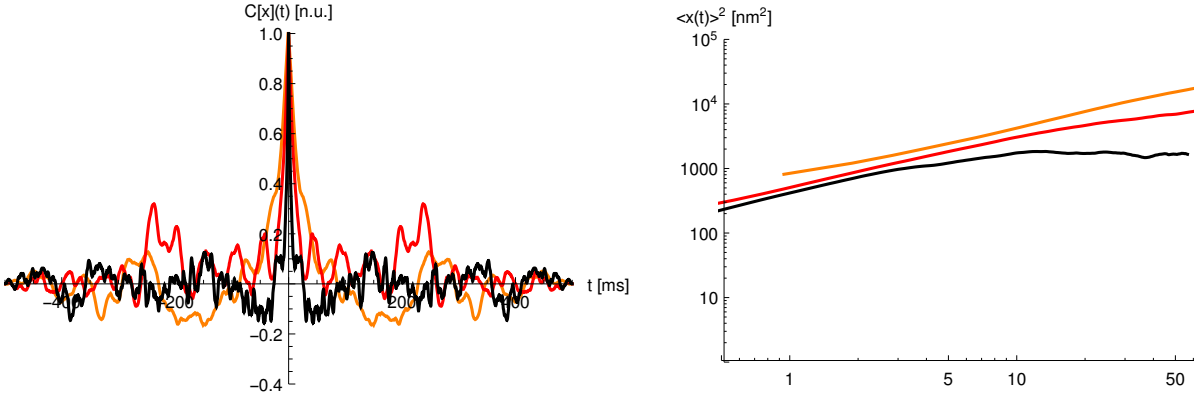
factor that causes Figure 4b(i) to often not evaluate a particularly symmetric graph, as expected. A longer trajectory was not evaluated in the interest of computational time.
Alternatively, the calculation can be repeated and the results averaged. This is demonstrated below for averaging 10 sets of results:

```
Figure4b[10]
```



This, as can be seen, amplifies the areas of large and small probability, so low probability areas becomes extremely low probability areas and high probability areas become extremely high probability areas. Thus, this effect means that this technique is not very good for realistically demonstrating the particle probability distribution. However, it does demonstrate the point that the particle is indeed confined exactly about the centre point of the trap, approaching the case where there is no thermal noise and the particle is stationary. It would be interesting to see how this effect compared to a decrease in the temperature of the system towards absolute zero [9]. It also demonstrates that the randomness in the system averages to zero, as it should since a zero mean Gaussian random number generator is used, and indeed, this was a characteristic by which the white noise W(t) was outlined [2].

```
Figure5
```



The position autocorrelation function (PAF) is calculated similarly to the VAF, but with positional values rather than velocity. Thus, similar to (12), it if defined as:

$C_x(t) = \overline{x(t'+t)\,x(t')}$ (21)

Which gives the finite-difference relation:

$C_{x,n} = \overline{x_{i+n}\, x_i}$ (22)

It is shown in the leftmost graph of Figure 5, with the PAF calculated for three different trap stiffnesses, as defined in the legend.

As the VAF showed the effect of the time scale τ, the PCF shows the effect of the time scale ϕ, relating to how quickly the trap pulls the particle back towards the central position after thermal noise drives it out. As the stiffness increases, the particle undergoes a stronger restoring force and the correlation time decreases: the graph shows the rate of decay of the PCF increasing with increasing trap stiffness, or in other words, the characteristic decay time of the PCF ϕ (roughly where the curve reaches zero) decreases. This demonstrates that the particle is brought back to the central position more rapidly by the restoring force at greater trap stiffnesses. The values of ϕ for each trap stiffness approximate the expected values provided by equation (16).

These conclusions and the data supporting them agree with [2] closely. The time step Δt used to calculate the PCF had to be smaller than ϕ, but still greater than τ in order for the noninertial approximation to be accurate. Δt was defined to be ϕ/100, which gave a time step Δt about 1000 times greater than τ. To improve the accuracy of the PCF results, it could be suggested to decrease the size of the time step (this applies to all calculations, except where Δt cannot be made lower). However, in this case, Δt must be kept significantly greater than τ. Thus, performing the calculation starting with the inertial finite-difference equation (rather than the noninertial version) and using a smaller time step Δt (which would then be acceptable) would be a good way to improve the accuracy of the PCF results. The average over a greater number of calculations could also be taken.

The right graph of Figure 5 shows the MSD for the trapped Brownian particle, for different trap stiffnesses. Whilst the free particle had a MSD that increased indefinitely, as far as was investigated (certainly into the t/τ=1000s), the trapped particle's MSD reaches a plateau which increases with decreasing trap stiffness. This suggests that the plateau is related to the confinement imposed by the trap. This makes physical sense as a harmonic restoring force imposes a boundary beyond which the particle has insufficient energy to reach, so the distance the particle can reach from the centre point has some maximum value, and thus the MSD will reach a maximum in a long time scale. Also, a greater trap stiffness (and thus greater restoring force) will result in the particle being more confined, ie. generally reaching a lower distance from the central position, and thus the MSD maximum will be lowered. The transition from the linear growth, characteristic of any noninertial trajection (corresponding generally to the free diffusive behaviour), to the plateau imposed by the optical trapping occurs, as expected, at approximately ϕ. For example, in the case that k=5 fN/nm, equation (16) gives the characteristic delay time:

```
ϕ[5 kp]
```

ϕ=3.8 ms

Which is where the transition of the k = 5 fN/nm line (black) occurs in the MSD graph on the fight of Figure 5. Also notice that this correlates well to where the k = 5 fN/nm line (black) in the PCF graph on the left drops to zero, as it should. The results for the k=5 fN/nm case:

```
ϕ[kp]
```

ϕ=19. ms

similarly correlates well with the graphs. Finally, the k=0.2 fN/nm case also approximately matches with the graphs:

```
ϕ[0.2 kp]
```

```
ϕ=94. ms
```

Again, the results correlate closely with those produced in [2] to give the same conclusions.

Conclusions

The Brownian motion of a particle suspended in a fluid was investigated. The physical case of a silica microparticle suspended in water was used to provide criteria for the computational modelling of this motion. The motion was explored in the free ballistic and diffusive regimes, and then under the confinement of optical traps of various trap stiffness.
The simulation used Gaussian random numbers to generate white noise for the random term in the stochastic Langevin equations used. The general approach was to then produce a finite-difference equation relating to the Langevin equation, from which the trajectory of the particle could be calculated.
It was demonstrated that the ballistic regime behaviour was dominated by particle inertia and could be accurately modelled using an inertial finite-difference equation. This transition to the diffusive regime at time scales greater than τ, in which velocity became randomised and thus a noninertial finite-difference equation provided a good approximation of the motion. An analysis of the VFC and the MSD confirmed these conclusions.
Introducing a harmonic restoring force to the stochastic equation modelled the effects of an optical trap confining the Brownian particle. Increasing the trap stiffness further confined the motion of the particle, shrinking the probability distribution of the particle towards the central position. The decay time ϕ also decreased with increasing trap stiffness, demonstrating the more rapid response of the restoring force in drawing the particle back to the central position. Further analysis showed how the trap limited the maximum distance the particle could reach from the central position, causing the MSD to reach a plateau on long time scales. This plateau was reached sooner for stronger traps.

There is great scope for further investigation of the topic. Extensions include:
Exploring the effects of different viscosities of the medium on the system.
Exploring the action of more complex forces on the particle, like an alternative optical potential (eg. double-well potential, asymmetric potential well or time-varying potential).
Investigating the effect of using a different random number generator (eg. Box-Muller algorithm or the Marsaglia polar algorithm).
Considering limiting cases, for example, running the simulation with a time step Δt that is not significantly smaller than ϕ. Would the numerical solution not converge, as suggested by [2]? What kind of behaviour might be observed (eg. unphysical oscillatory behaviour or divergence)?
Exploring stochastic resonnt damping in systems with time-dependent potentials.
Setting up a double-well potential and exploring the Kramers transition phenomenum.
Applying an oscillatory driving force to the double-well potential and exploring stochastic resonance [10], or
Finding the optimal temperture for synchronisation of the Kramers jump frequency with the oscillating force frequency.
Replacing the passive silica microbead with an active Brownian particle, or microswimmer [11].

References

[1] R. Mahnke, J. Kaupuzs & I. Lubashevsky, *Physics of Stochastic Processes: How Randomness Acts in Time*, 1st ed. Weinheim, Germany: Wiley-VCH, 2009.

[2] G. Volpe & G. Volpe, 'Simulation of a Brownian particle in an optical trap', *American Journal of Physics*, vol. 81, no. 3, 2013.

[3] R. Brown, 'A Brief Account of Microscopical Observations Made in the Months of June, July, and August 1827, on the Particles contained in the Pollen of Plants', *Philosophical Magazine*, vol. 161, no. 4, 1828.

[4] 'Chapter IX: The Langevin Approach', in *Stochastic Processes in Physics and Chemistry*, Amsterdam; North-Holland, 2001, pp. 219-243.

[5] W. Coffey, Y. Kalmykov & J. Waldron, The Langevin Equation (Series in Contemporary Chemical Physics #14): With Applications to Stochastic Problems in Physics, Chemistry and Electrical Engineering, Vol. 14, 2nd ed. 5 Tok Tuck Link, Singapore: World Scientific Publishing Company, Incorporated, 2005.

[6] T. Li & M. Raizen, 'Brownian motion at short time scales', *Annalen der Physik*, vol. 525, no. 4, 2013.

[7] A. Einstein, 'The theory of the Brownian Motion', *Annalen Der Pkysik*, vol. 19, no. 2, pp. 371-381, Feb. 1906.

[8] A. Ashkin, J. Dziedzic, J. Bjorkholm & S. Chu, 'Observation of a single-beam gradient force optical trap for dielectric particles', *Optical Angular Momentum*, vol. 11, no. 5, May 1986.

[9] T. Li, S Kheifets, D. Medellin & M.G. Raizen, Measurement of the instantaneous velocity of a Brownian particle, *Science* 328, 5986, 1673-1675 (2010).

[10] A. Allison & D. Abbott, 'Stochastic Resonance, Brownian Ratchets and the FokkerPlanck Equation', *AIP Publishing*, vol. 665, pp. 74–83, 2003.

[11] G. Volpe, S. Gigan & G. Volpe, 'Simulation of the active Brownian motion of a microswimmer', American Journal of Physics, vol. 82, no. 7, 2014.

In[5]:=
```
Export[FileNameJoin[{NotebookDirectory[], "project.pdf"}],
  EvaluationNotebook[]];
```