Microrheology Lab Tutorial

Contents

1 Introduction

Welcome to the microrheology laboratory of 20.410. This document is meant to be a step by step tutorial to make microrheology quantitative measurements using multiple particle tracking. In this technique, fluorescent microspheres are embedded in a sample of the studied material. The spheres undergo Brownian random motion, and their trajectories can be extracted by processing the movies acquired with fluorescence video microscopy. From the trajectories, a statistical quantity called the mean-squared displacement can be calculated, and then related to the mechanical property of the material. This tutorial describes the particle tracking technique and the successive steps to perform such measurements.

2 Experimental Setup - Fluorescence Video Microscopy

2.1 Principle of fluorescence microscopy

The basic function of a fluorescence microscope is to irradiate a specimen with a desired and specific band of wavelengths, and then to separate the much weaker emitted fluorescence from the excitation light. In a properly configured microscope, only the emission light should reach the eye or detector so that the resulting fluorescent signal is superimposed with high contrast against a dark (or black) background. The specimen is illuminated with light of specific wavelength(s) which is absorbed by the fluorophores, causing them to emit longer wavelengths of light (of a different color than the absorbed light). The illumination light is separated from the weaker emitted fluorescence through the use of an emission filter. The typical components of a fluorescence microscope are the light source, the excitation filter, the dichroic mirror, and the emission filter (see figure 1). The filters and the dichroic are chosen to match the spectral excitation and emission characteristics of the fluorophore used in the particles.

2.2 Montage

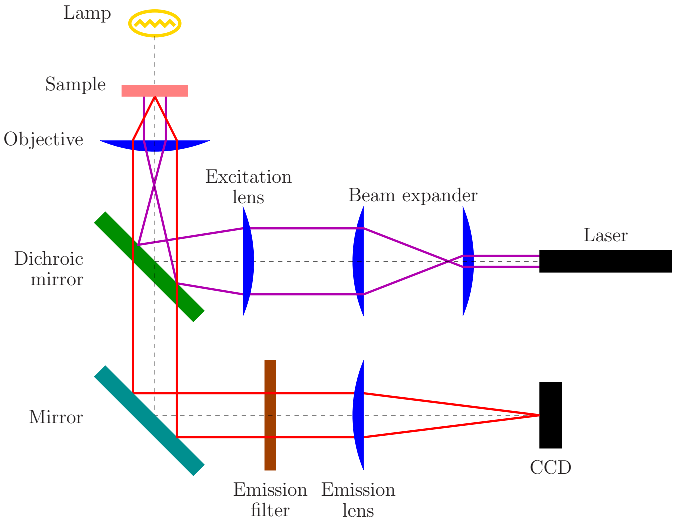


Figure 1: Schematic of the fluorescent microscope used in the microrheology lab.

The microscopes are built in-house in the instrumentation laboratory. Here the source is a $\lambda = 532\,\mathrm{nm}$, $5\,\mathrm{mW}$ green laser pointer. The laser being almost monochromatic, no excitation filter is required. The beam is almost entirely reflected by the dichroic mirror towards the sample. In the observed sample, the fluorescent beads are illuminated by the laser, which wavelength falls in the excitation spectrum of the particles. The latter emit at a maximum wavelength of 605 nm, at which the dichroic mirror is transparent. The signal emitted by the fluorescent particles passes through an emission filter which selects only wavelength above 600 nm. This is used to remove residual fluorescent signal that could pass the dichroic. The filtered light is finally detected by a CCD camera. Figure 1 presents a rough schematic of the microscope used in the lab.

The objective that you will be using has a 40× magnification. Its working distance (distance between the front end of the objective and the sample plane that is in focus) is 0.65 mm. This is the distance you will need to approach the objective from the sample in order to observe it. The sample will be placed on a stage that can be moved in the horizontal plane with fine adjustments, and in the vertical direction with a coarse translation screw. Fine altitude adjustment is possible by using the screw on which the objective is mounted.

To have a more detailed description of how these microscopes were built, you can read the report that was written for the class "20.309: Biological Instrumentation and Measurement Laboratory" (available at http://www.openwetware.org/images/4/48/20.309f06mod3.pdf), where this design was originally developed.

2.3 Samples

We will study 3 different materials during the class: pure water, a 55% volume glycerol solution and living cells. In each of these materials, 1.1 μm diameter fluorescent spheres have been embedded at a concentration of about 5×10^8 beads per mL (refer to appendix B to see how these samples were prepared). The water and glycerol samples have been injected into a custom made sealed chamber (see appendix A to learn how these chambers are built). These chambers are such that drain and evaporation are minimized to avoid unwanted convective motions of the particles.

The cells are NIH 3T3 mouse embryonic fibroblasts. Few days before the experiments, the cells were plated on 35 mm glass-bottom MatTek cell culture dishes. Fluorescent particles were mixed to the growth medium added to the platted cells for a period of a day for bead endocytosis. About ten microspheres should be embedded in each cell. Since the cells are attached to the bottom of the dish, convection motions of the beads is limited to the cell's internal dynamics.

3 Movie Acquisition

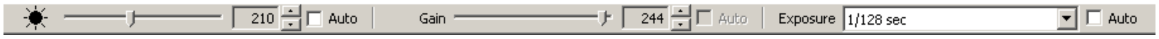
Unlike some microscope you may be used to, the one we built here does not have an eyepiece for direct visual observation. Instead, we observe and capture the movies with a Firewire-enabled CCD camera (DMK 21F04 from The Imaging Source). Its 8-bit monochrome sensor is 640 × 480 pixels, each of which is a square 5.6 μm on a side. The camera software is called IC Capture, and is run from the PC desktop. If the program gives an error and cannot find the connected camera, it may need to have its driver updated. Ask the TA if you encounter this problem.

Find the object of observation using the white light (or bright field) illumination. Change the altitude of the stage using the coarse adjustment. Remember that the working distance of the 40× objective is 0.65 mm. You can refine the altitude using the screw on which the objective is mounted. Switch to the fluorescence (through the laser; use caution because this laser can damage the eyes) only when you have a reasonable image in bright field.

3.1 Acquisition parameters

The image processing step described in the next section uses brightness variations information from the movie to extract accurate particle positions. A typically good, trackable fluorescence image, should consist of white spots (corresponding to the particles) on a black background. The particles' signal should cover the maximum dynamic range of the camera (darkest background and brightest particle's spot) without saturating the signal. When acquiring the movies, it is important to have a "clean" signal that verifies these conditions.

A frame of a movie is a 640 × 480 array of pixels, each pixel contains a value of digitized brightness ranging from 0 to 255 (displayed on a grayscale colortable such as the one used by IC Capture, 0 is usually black and 255 is white; that is $256 = 2^8$ different shades of gray for the 8-bit camera used here). To have the signal of the particles falling in this range, you can adjust the brightness (an offset added to the CCD output signal), the gain (amplification of the output signal) and the exposure time (time during which the camera shutter is open to collect the light when acquiring each frame) using the following status bar:



☀ 210 ☐ Auto Gain 244 ☐ Auto Exposure 1/128 sec ☐ Auto

Experiment for yourself how the image quality is affected when you change these three parameters individually. To avoid a phenomena called "motion blur" occurring when recording movies of mobile objects, you will need to keep the exposure time as small as possible (typically below 1/512 s).

To evaluate the quality of a movie for particle tracking, a useful tool of IC Capture is the histogram: it shows the live brightness distribution of the movie frames (i.e. the horizontal axis is the brightness, from 0 to 255, and the vertical axis is the number of pixels that have a given value of brightness). When having isolated brighter spots on a darker background, a peak should appear on the histogram corresponding to all the pixels having low brightness. The pixels illuminated by the particles are more rare, and their brightness values vary on a wider range. On the histogram, they will thus appear as an almost flat tail on the right hand side of the background peak (see figure 2). This tail is essentially the "useful" signal to perform the particles tracking, and you can almost entirely discriminate other noise signals by varying the acquisition parameters.

Figure 2 shows some typical images and histogram obtained when varying the acquisition parameters. The last figure on the right hand side is an example of what would be a typically good image (and its corresponding histogram) to perform particle tracking. Note however that the parameters are dependent on the microscope you are using, so you will need to adjust them accordingly to obtain an histogram similar to the one seen on this figure.

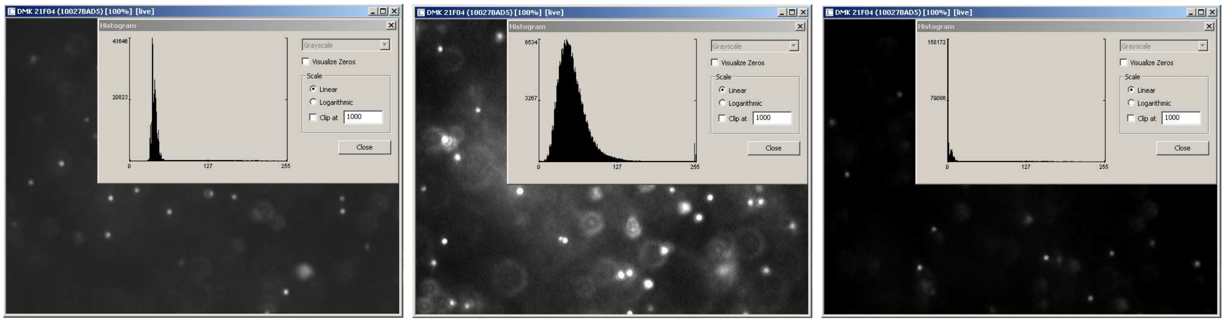
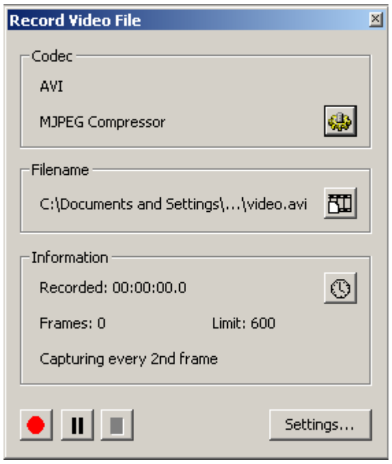


Figure 2: Influence of the acquisition parameters on the quality of the images. The right image shows a typically good image for particle tracking, where the background signal has been discarded and the particles signal covers the entire brightness range, without saturation.

3.2 Recording parameters

The last step consists in recording the movie onto the hard drive for further digital processing. Movie recording parameters can be accessed using the recording settings button. This allows you to change the compression format (Codec), the name and location of the movie file, the duration of the movie in number of frames and the frame rate. The right choice of compression parameters is given by the snapshot on figure 3. Verify that they are correctly set and remember in which directory you save your movie and its filename. Click “Settings...” to get a self-explanatory menu to change any settings you’re not happy with.


Record Video File

Codec
AVI
MJPEG Compressor

Filename
C:\Documents and Settings\...\video.avi

Information
Recorded: 00:00:00.0
Frames: 0 Limit: 600
Capturing every 2nd frame

Figure 3: Recording settings to be used with IC Capture.

4 Movie Processing

The next step in the multiple particle tracking technique is to extract a list of probe trajectories from the movie of particle motions acquired with the video microscope. The image processing algorithms we use in this study were originally developed by Crocker and Grier [1]. These algorithms were initially written in IDL language but they have been recently translated into a Matlab version. The software routines have been made available online by Daniel Blair and Eric Dufresne at http://www.seas.harvard.edu/projects/weitzlab/matlab/ with a brief tutorial (the website http://www.physics.emory.edu/∼weeks/idl/ maintained by Eric Weeks is also a good reference). These procedures have been installed on the PCs, along with some supplementary and essential algorithms written for this laboratory.

4.1 Principles of video processing for particle tracking

We explain in appendix C the image processing principles underlying the tracking and a more detailed description can be found in the original paper [1]. Briefly, there are 3 steps to transform a movie into a list of probe trajectories: removing unwanted noise and background signal in each frame (image restoration), locating the particles in each frame pixel array (features location), and finally connecting successive positions into trajectories (labeling positions).

4.2 Algorithms

Open Matlab, and using the slider on the top of the window, labeled “Current Directory”, find the directory in which you have saved the movie. In order to make the algorithms work, make sure you have the following files at the same location as the movie file you need to analyze:

MATLAB 7 (R14)

- `bpass.m`
- `cntrd.m`
- `getdx.m`
- `pkfnd.m`
- `plot_hist.m`
- `pretrack.m`
- `track.m`

4.2.1 Image restoration and feature location

The procedure `pretrack` loops over all frames of a movie to perform noise filtering and particle localization (using successively on each frame the procedures `bpass`, `pkfnd` and `cntrd`; see appendix C). The following line localize the features in all frames of the movie recorded under the name `movie.avi` (change this name accordingly):

```
>> p=pretrack('movie.avi',11,21);
```

The first number is the filtering parameter called w_1 in the appendix C, and the second number is w_2 the integration parameters used in `cntrd` (see the appendix C for more details). At this point you just need to know that w_1 and w_2 should be about the size in pixel of the spot you need to localize on the frames. The choice $w_1 = 11$ and $w_2 = 21$ is appropriate for the observation of 1.1 μm diameter particles with the 40× objective, where the brightness profile of a particles covers about 15 to 20 pixels. The resulting array variable `p` has 5 columns. The first two columns are the x and y positions of the bright features on each frame. The third and fourth columns are the total brightness and squared radius of gyration of the corresponding feature's brightness profile. These two columns are not used here. The fifth column indicates the index of the frame in which the features were found (i.e. the time index). You can type

```
>> p(1:10,:)
```

to see the first 10 lines of the array.

4.2.2 Building trajectories

The next step is to connect the positions into trajectories. Indeed, so far there is no knowledge on how the successive positions are connected from one frame to the next to belong to the same particle's trajectory. Several parameters are required to perform this calculation. A first set of parameters is stored into the structure called `param` and defined by typing:

```
>> param.good=6; param.mem=0; param.quiet=0; param.dim=2;
```

The field good indicates the minimum trajectory length to be considered. Here trajectories shorter than 6 frames will be discarded. The field mem is the number of steps that a particle can be lost for and then recovered again. Here, if a particle is lost for one frame and reappears the following frame, it will be tracked as a new particle. The field quiet is used to display more information about the tracking process. And dim gives the dimensionality of the tracking. Here we are performing a 2D tracking.

The procedure to build trajectories is then called by:

```
>> t=track(p,7,param);
```

The first input is the pre-tracked array p output from the procedure pretrack. The second input is the important parameter w_3 defined in appendix C. It is the expected maximum displacement, in pixel unit, that a particle would move from one frame to the next. You will need to find this parameter yourself depending on the material you are analyzing (the value $w_3 = 7$ works well for the 1.1 μm particles in pure water, as shown later). We explain in the next paragraphs how to determine w_3 for other systems. The returned tracks array t contains 6 columns: the first two are the (x, y) positions, the next two are brightness and squared radius of the spot (same as before), the fifth column is the time index and the sixth column gives the trajectory index to which the successive positions belong. Again type

```
>> t(1:10,:)
```

to see how this array looks like.

To find a correct value for w_3, you will use the procedure getdx on the tracks array t. This algorithm calculates the list of displacements Δx the particles undergo over the time interval j, called the lag time. Suppose a particle follows the 2D trajectory $\mathbf{r}(t) = (x(t), y(t))$. When recorded on movie, only the positions visited by the particle at the discrete times corresponding to the successive frame captures will be measured. In the track array, the positions $\mathbf{r}_i = (x_i, y_i)$ will be reported at the successive times i corresponding to the frame number i of the movie. Starting from the first frame, the particle is at a position x_1 on the x-axis. Then j frames latter, the same particle is at a position x_{j+1}. A displacement $\Delta x_1 = x_{j+1} - x_1$ can be computed in the x direction for a lag time j. From the same trajectory and starting at later times, other displacements $\Delta x_2 = x_{j+2} - x_2$, $\Delta x_3 = x_{j+3} - x_3$, ... can also be calculated, thus creating a list of displacements for this trajectory (this can be performed in the y direction as well). The procedure getdx returns a two-columns array of the displacements (first column gives the displacement in the x direction, the second in the y direction) calculated for all the trajectories contained in the tracks array t. Its entry is

```
>> dx=getdx(t,1);
```

where the number is the lag time ($j = 1$ in the command above). These displacements are observations of a random variable and will be later used to calculate the important statistical quantity called the mean-squared displacement.

From the definition of w_3, you understand that none of the elements of the array dx calculated above (with a lag time of 1) will be greater than w_3 and less than $-w_3$. The distribution of measured displacement will be cut off above w_3 and below $-w_3$. You need to verify that above w_3 and below $-w_3$, this distribution exhibits only negligible occurrences. You can use

```
>> plot_hist(dx,40);
```

to plot the displacements histogram along with the best Gaussian fit (the red line on the graph). Consider the examples given in figure 4, corresponding to the displacements histogram of 1.1 μm particles in pure water. You can see that $w_3 = 3$ cuts off the distribution at points above which there

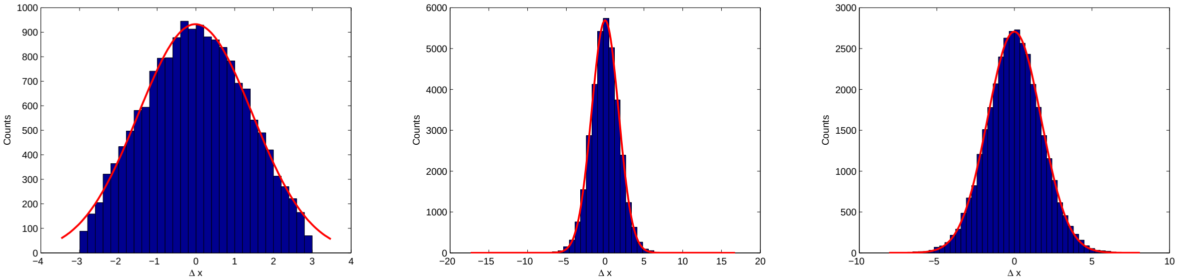


Figure 4: Measured displacements distribution for $w_3 = 3$ (left), $w_3 = 15$ (center) and $w_3 = 7$ (right).

should be more possible events. Using a larger value $w_3 = 15$ lets you see the entire distribution, from which you can evaluate that $w_3 = 7$ is a correct choice that does not significantly affect the shape of the histogram. Note how the distribution of displacements is nicely fitted by a Gaussian distribution in this case.

Once you have determine a correct value for w_3, you are done with the tracking. All further microrheology calculations will be made from the tracked array `t`. You can save this array into an ASCII-delimited file if you feel the need to, by using the following command:

```
>> dlmwrite('t.movie.txt', t, 'delimiter', '\t', 'precision', 6);
```

This type of file should be importable by most applications. Under Matlab, it can be read back to the variable `t` using the following entry:

```
>> t=dlmread('t.movie.txt');
```

5 Microrheology

5.1 Principles (covered in class)

In this section we explain how quantitative studies can be performed from the probes' trajectories extracted by the technique exposed above. Essentially, the amplitude of the particles motion depends on the mechanical properties of the surrounding environment. These properties are often assessed in terms of rheometric measurements. Rheology is the study of the deformation and flow of a material in response to an applied stress. Simple solids store energy and provide an elastic response, whereas simple liquids dissipate energy through viscous flow. For more complex viscoelastic materials, rheological measurements reveal both the solid- and fluid-like responses which generally depend on the time scale at which the sample is excited [2]. Traditionally, rheological measurements are performed on several milliliters of material in a mechanical rheometer (e.g. a cone-and-plate geometry rheometer) by applying a small amplitude oscillatory shear strain $\gamma(t) = \gamma_0 \sin(\omega t)$ and measuring the resultant shear stress $\sigma(t)$. In the viscoelastic linear regime (or regime of small amplitude straining, typically $\gamma_0 \ll 1$), the shear stress is proportional to the amplitude of the applied strain γ_0 and is itself sinusoidally varying in time. It can be represented as:

$$\sigma(t) = \gamma_0 \left[G'(\omega) \sin(\omega t) + G''(\omega) \cos(\omega t) \right] . \qquad (1)$$

In the above equation, the term proportional to $G'(\omega)$ is in phase with the strain, while the term containing $G''(\omega)$ is in phase with the rate of strain $\dot{\gamma}(t) = \mathrm{d}\gamma/\mathrm{d}t$. $G'(\omega)$ is called the storage modulus and represents storage of elastic energy, and $G''(\omega)$ is called the loss modulus and represents the viscous dissipation of that energy. The complex shear modulus $G^*(\omega)$ is defined by

$G^*(\omega) = G'(\omega) + iG''(\omega)$ [2]. Rheology measurements such as these have given valuable insight into structural rearrangements and mechanical response of a wide range of materials. They are particularly valuable in characterizing soft materials or complex fluids. However, conventional mechanical techniques are not always well-suited for all systems. Typically, milliliter sample volumes are required, precluding the study of rare or precious materials, including many biological samples.

To address these issues, a new class of measurement techniques has emerged. These have come to be called *microrheology* methods, and probe the material response on micrometer length scales, using microliter sample volumes [4]. Among the methods currently available to perform such measurements, we focus here on the most popular one: particle tracking microrheology. This methods typically use embedded micron-sized mechanical probes to locally deform the medium, and information is extracted from their motion. Techniques using small particles tracking fall into two classes: those involving *active* manipulation of probe particles within the sample, and those employing *passive* observation of thermal fluctuations of such probe particles. In either case, the probes used are typically spherical beads of between a fraction of micrometer to several micrometers in diameter and measurements are made from their trajectories.

5.2 Quantitative Passive Microrheology (covered in class)

Here we focus on a passive particle tracking microrheology technique. The passive measurements use only the thermal energy to deform the medium, allowing the study of very fragile structures, but consequently, the material must be sufficiently soft to allow detectable motion of particles. Also, the intrinsic stochastic nature of thermal energy requires to collect a large amount of data in order to build a consistent statistic. The dynamics of particle motions are then revealed in the time dependent position correlation function of individual tracer particles [5] through the mean-squared displacement, that is defined as

$$\langle \Delta x^2(\tau) \rangle = \left\langle \left[x(t+\tau) - x(t) \right]^2 \right\rangle \tag{2}$$

Here $x(t)$ is the 1-dimensional particle position at time t, τ is called the lag time and the brackets $\langle \dots \rangle$ indicate an ensemble/time average. When the bead evolves in a continuum, the generalized Stokes-Einstein relation relates the mean-squared displacement to the complex shear modulus [5, 3]. In the Laplace[1] frequency space, it is written:

$$\tilde{G}(s) = \frac{s}{6\pi a} \left[\frac{2k_{\mathrm{B}}T}{s^2 \langle \Delta \tilde{x}^2(s) \rangle} - ms \right] \tag{3}$$

where m and a are the spherical particle's mass and radius, k_{B} is the Boltzmann's constant and T the absolute temperature (see Ref. [3] for a demonstration of this relation). Within a valid frequency range, the inertial effect of the probe particle, ms in the above equation, can be neglected, and we obtain in the Fourier frequency domain:

$$G^*(\omega) = \tilde{G}(i\omega) = \frac{d k_{\mathrm{B}} T}{3\pi a (i\omega) \langle \Delta \tilde{x}^2(i\omega) \rangle} \ . \tag{4}$$

This equation represents an expression of the generalized Stokes-Einstein relation consistent with the convention of standard rheology which generally reports $G^*(\omega)$. For example, we can easily get

[1] The Laplace Transform is defined as:

$$\mathcal{L}_t \left[g(t) \right] = \tilde{g}(s) \equiv \int_0^{+\infty} g(t) e^{-st} \mathrm{d}t \ .$$

the behavior of $\langle \Delta x^2(\tau) \rangle$ in a pure Voigt viscoelastic fluid model, for which $G^*(\omega) = G + i\omega\eta$. We find

$$\langle \Delta x^2(\tau) \rangle = \frac{k_\mathrm{B}T}{3\pi a G}\left(1 - e^{-G\tau/\eta}\right) , \tag{5}$$

from which we can restore the diffusive linear scaling $\langle \Delta x^2(\tau) \rangle = 2D\tau$ in a purely viscous liquid by setting $G = 0$. We introduced here the diffusion coefficient D through the famous (not generalized) Stokes-Einstein relation:

$$D = \frac{k_\mathrm{B}T}{6\pi a \eta} . \tag{6}$$

Unlike in simple fluids, the mean-squared displacement of the tracers in a complex material may scale differently with τ, and we can write $\langle \Delta x^2(\tau) \rangle \sim \tau^{\alpha(\tau)}$. The coefficient α is called the *diffusive exponent*. The particles may exhibit subdiffusive motion ($0 < \alpha < 1$) or become locally constrained ($\alpha = 0$, see for example the elastic limit $\tau \gg \eta/G$ in relation (5)). It is important to note that, as an expression of the second law of thermodynamics, it is expected to have $\alpha \leq 1$ for all passive measurements in any material at equilibrium, independently of any time or length scale involved.

Using this local power-law approximation $\langle \Delta x^2(\tau) \rangle \sim \tau^{\alpha(\tau)}$, it is possible to derive an approximate expression of equation 4 (see Ref. [3]):

$$G^*(\omega) \approx \frac{k_\mathrm{B}T}{3\pi a} \frac{\exp\left[i\pi\alpha\left(1/\omega\right)/2\right]}{\left\langle \Delta x^2\left(1/\omega\right)\right\rangle \Gamma\left[1 + \alpha\left(1/\omega\right)\right]} \tag{7}$$

where Γ designates the gamma function and where the local power law is given by:

$$\alpha\left(1/\omega\right) = \left.\frac{\mathrm{d}\ln\left\langle \Delta x^2\left(\tau\right)\right\rangle}{\mathrm{d}\ln\tau}\right|_{\tau=1/\omega} \tag{8}$$

We use this equation in the procedure `shearmod` explained in the next section, by pointing out that from Eq. 7:

$$\begin{aligned} G'(\omega) &= |G^*(\omega)| \times \cos\left[\pi\alpha(\omega)/2\right] \\ G''(\omega) &= |G^*(\omega)| \times \sin\left[\pi\alpha(\omega)/2\right] \end{aligned} \tag{9}$$

5.3 Microrheology Algorithms

For this part, make sure that the current working directory of Matlab contains the following procedure:

- `msd.m`
- `shearmod.m`

Also, we call as before `t` the tracks array. Verify that you have it loaded in Matlab's workspace.

We have seen that we need to first calculate the mean-squared displacement from the tracks array in order to perform any quantitative measurement. In the tracking algorithms described in section 4.2.2, we used the procedure `getdx` to extract the list of all particles displacements and plot their histogram. From its definition, the mean-squared displacement is simply the variance of this histogram, estimated by:

$$\langle \Delta x^2(\tau) \rangle = \frac{1}{N-1} \sum_{k=1}^{N} \left(\Delta x_k - \frac{1}{N} \sum_{k=1}^{N} \Delta x_k \right)^2 \tag{10}$$

where $\{\Delta x_k\}_{1\leq k\leq N}$ is the list of all displacements extracted from the trajectories (using getdx). Here we will calculate this variance for displacements extracted at various lag times in order to have the entire time dependent quantity $\langle \Delta x^2(\tau)\rangle$. Note that j and τ are just differing by a conversion factor (j is in number of frame time intervals whereas τ is in seconds, see later in the paragraph). The algorithm msd has been written to perform this calculation. Type

```
>> m=msd(t,0.139,1/15);
```

where t is the tracks array previously extracted from the movie. the first number is the conversion factor from pixels to microns (with a 40× magnification and 5.6 μm pixels, we obtain an on-screen magnification of 0.140 μm.pxl^{-1}; the value 0.139 μm.pxl^{-1} is a more precise measurement that takes the microscope's aberration into account). The second number is the time conversion factor from frame index to a duration in seconds (we acquired one every two frame of a video signal that has 30 frames per second). This calculates a 4 columns array m where the first column is the lag time τ in seconds (37 lag times are reported, from 1/15 to 10 s and are log-spaced), the second column is the mean of the displacements in μm, the third column is the mean-squared displacement in μm^2, and the fourth column is the number of observed displacement used to perform these calculations. You can type

```
>> loglog(m(:,1),m(:,3),'s');
```

to create a log-log plot of $\langle \Delta x^2(\tau)\rangle$ versus τ, which already gives a quantitative information on the material.

As pointed out above, the more conventional way to interpret these data is in term of rheological quantities. The routine shearmod converts the mean-squared displacement to rheological moduli using Eqs 7 to 9. Type

```
>> g=shearmod(m,1.1,23);
```

to calculate $G'(\omega)$ (second column of g), $G''(\omega)$ (third column of g) versus the frequency ω (first column of g). The first number is the diameter of the particles in microns, and the second number is the temperature in Celcius (change accordingly if you feel the room temperature is different). Again, to look at the result:

```
>> loglog(g(:,1),g(:,2),'bo',g(:,1),g(:,3),'rs');
```

which plots $G'(\omega)$ versus ω with blue circles and $G''(\omega)$ versus ω with red squares.

For purely viscous fluids such as the water and the glycerol solution samples that we study here, we know that $G^*(\omega) = i\eta\omega$ (that is $G'(\omega) = 0$ and $G''(\omega) = \eta\omega$), where η is the viscosity. Although you might observe a residual non-zero $G'(\omega)$ (mainly coming from noise and statistical uncertainties), $G''(\omega)$ should be consistently much greater over a wide range of frequency. You can use:

```
>> fit(g(1:10,1),g(1:10,3),'poly1')
```

to perform a linear fit of $G''(\omega)$ in the high frequencies region. The command above only uses the 10 highest frequency points (from 1.5 to 15 Hz), where statistical accuracy is the best. It displays the result of the fit, where the coefficient p1 is the slope, that is the value of η in Pa $\cdot$ s. Compare your results with tabulated values.

A Building a Chamber of Observation

The sample is injected into a custom made chamber for observation with video microscopy. The chamber must be a perfectly sealed closed volume, as any leak induces a drain and/or evaporation that can produce unwanted convection motions of the particles. To make the chambers of observation, we put two 1 cm-wide stripes of parafilm on a microscope slide (typically 24 × 60 mm, No. $1\frac{1}{2}$ thickness - that is about 0.18 mm thick), parallel and apart from each other by approximately 5 mm. A square coverslip (18 × 18 mm, No. 1 thickness or about 0.15 mm) is then deposit to bridge both parafilm stripes, as described in the figure 5. The parafilm is briefly melted while moderate pressure is applied on the top coverslip for efficient sealing. We then obtain a channel open at both end, that is 5 mm wide and approximately 150 μm high.

Both end of the cell are sealed using vacuum grease shortly after injection of the sample in order to prevent drying and convective flow. We found that these chambers remain hermetically sealed for at least five days.

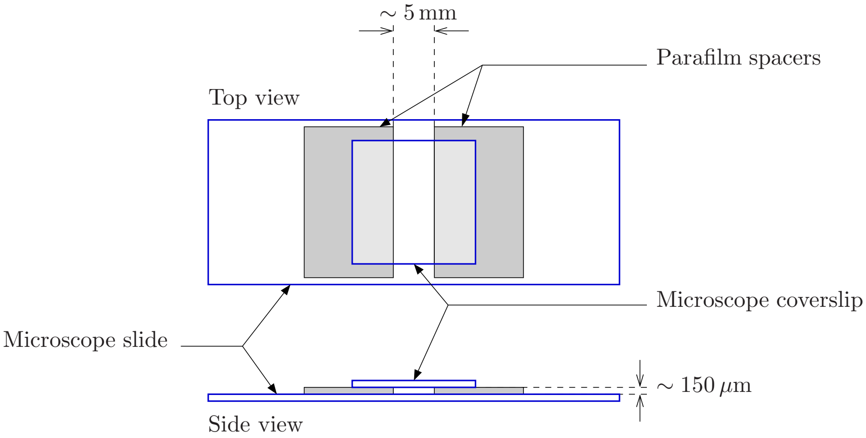


Figure 5: Custom made microscope observation chamber for multiple particle tracking experiments. A standard glass microscope slide holds the chamber made by adding a cover slip separated by two parafilm spacers. The resulting chamber is approximately 5 mm wide and 150 μm high.

B Samples Preparation

The beads were purchased from Molecular Probes (Cat. No. F-8775) and are shipped at 1% solid concentration, that is about 10^{10} microspheres per mL. We mixed thoroughly (vortexing and sonicating) 30 μL of the beads commercial solution into 570 μL of de-ionized filtered water to perform a 20× dilution and reach the desired concentration. About 30 μL was then injected in the chamber previously described.

The same protocol is followed for the glycerol solution, but the 30 μL of beads commercial solution are now mixed with 330 μL of pure glycerol and 240 μL of de-ionized filtered water. We made sure to thoroughly sonicate the solution to ensure well-mixing of the glycerol and the beads in water.

The NIH 3T3 fibroblasts were prepared as follows. Cells were cultured at 37°C in 5% CO_2 in standard 100 mm × 20 mm cell culture dishes (Corning) in a medium referred to as **DMEM++**

that consists of DMEM (Cellgro) supplemented with 10% fetal bovine serum (FBS - from Invitrogen) and 1% penicillin-streptomycin (Invitrogen). The day prior to the microrheology experiments, fibroblasts were plated on 35 mm glass-bottom cell culture dishes (MatTek). On the day of the experiments, the cell confluency should reach about 60%. 1 μm diameter orange fluorescent microspheres (Molecular Probes) were mixed with the growth medium (at a concentration of 5×10^5 beads/mL) and added to the plated cells for a period of 12 to 24 hours for bead endocytosis.

C Details of Multiple Particle Tracking Algorithms

C.1 Images Filtering

The filtering step of the raw image is made thanks to a linear band-pass spatial filter. The typical images obtain in colloidal video microscopy are composed of white circular spots distributed on a black background. The video images are affected by noise from a variety of origins [7]. Subtracting off the background is made by building a background image resulting from the convolution of the original image with a constant kernel of size $2w_1+1$, where w_1 is a larger than the typical radius in pixels of a single particle brightness profile on the array, but smaller than the typical interparticle distance (w_1 is typically a particle image diameter). If the original array is described by $A_{i,j}$ the brightness value for the pixel indexed by (i,j) on the CCD matrix (say i is in the subscript in the x direction and j in the y direction), we obtain the background image by using the convolution:

$$B_{i,j} = (2w_1+1)^{-2} \sum_{|m|\leq w_1} \sum_{|n|\leq w_1} A_{i+m,j+m} \tag{11}$$

which corresponds to keeping the long spatial wavelength (low-pass filter in the spatial frequency space with a cutoff at $(2w_1+1)^{-1}$). Another random noise with small correlation length of about 1 pxl comes from digitization and camera noise, and is filtered by smoothing the original image with a narrow Gaussian kernel, with again a support square mask of size $2w_1+1 \gg 1$. The resulting Gaussian smoothed image is calculated by:

$$G_{i,j} = \left[\sum_{|m|\leq w_1} e^{-m^2/4}\right]^{-2} \sum_{|m|\leq w_1} \sum_{|n|\leq w_1} A_{i+m,j+m} \times e^{-(m^2+n^2)/4} \tag{12}$$

which is again a low-pass filter in the spatial frequency domain, the cutoff being this time 1 pxl^{-1}. The difference between the noise-reduced image G and the background image B is an estimate of the ideal image $\widehat{A}$:

$$\widehat{A} = G - B = A * K \tag{13}$$

where the convolution kernel is given by:

$$K_{m,n} = \left[\sum_{|m|\leq w_1} e^{-m^2/4}\right]^{-2} \times e^{-(m^2+n^2)/4} - (2w_1+1)^{-2} \quad \text{for } |m|, |n| \leq w_1 \tag{14}$$

and $K_{m,n} = 0$ otherwise. The image resulting from this difference is a low intensity background with sharp circular spot corresponding to the original particle images. The choice of a support of size w_1 for the Gaussian filtering allows the calculation of a single convolution step of the image with the kernel K. The result is also usually a higher precision data arrays, if the original 8-bit image has been accordingly converted.

The Matlab code performing this transformation is called `bpass`. From a movie file `movie.avi` located in the same directory than `bpass.m`, you can extract the first frame and store it into the variable array `im` by typing

```
>> im=aviread('movie.avi',1); im=im.cdata(1:464,:,1);
```

The second command in this line removes an advertising stripe (!) automatically inserted by the capturing application IC Capture. Display the image using

```
>> [m,n]=size(im);figure('Units','pixels','Position',[100 100 n m]);
>> colormap('gray');set(gca,'Position',[0 0 1 1]);
>> image(im);
```

where the first two lines are used to display the image so that each element in the data matrix corresponds to a single screen pixel. You can then test the filtering procedure by

```
>> bim = bpass(im,1,11);
>> image(bim);
```

where the first number in the bpass call is the Gaussian kernel correlation length (as explained above, it is usually set to 1 pxl) and the second number is the parameter w_1 described earlier.

C.2 Locating Features

Candidate features are located at the pixel exhibiting the local brightest signal within a distance w_2. In the original IDL codes [1], these local brightness maxima were identified using a non-linear morphological operation called gray-scale dilatation, with a disk of radius w_2 as the structural element. This operation sets the value of an image pixel, say $A_{i,j}$ to the maximum brightness value within a distance w_2 of the pixel coordinate (i, j). A pixel in the original image with same value as in the dilatation-transformed image is then a candidate feature location. The dilatation operation being too slow under Matlab, it has been replaced by a method based on a brightness threshold. This new routine locates pixels which brightness is above a user-input threshold, and remove multiple hits to always keep only one maximum within a distance of minimum separation.

This locally brightest pixel, say at coordinates (i, j), is presumably near the particle image true center. Resolution in locating this center can be gained by using the corresponding local brightness profile. A brightness-weighted centroid is calculated in a circular region centered in (i, j) and of radius w_2 such that

$$x = i + \mu_0^{-1} \textstyle\sum_{m^2+n^2 \leq w_2} m \times A_{i+m,j+n} \quad \text{and} \quad y = j + \mu_0^{-1} \textstyle\sum_{m^2+n^2 \leq w_2} n \times A_{i+m,j+n} \tag{15}$$

are the corrected positions of the brightness center obtained at a subpixel resolution, where $\mu_0 = \sum_{m^2+n^2 \leq w_2} A_{i+m,j+n}$ is the integrated brightness of the corresponding spot. The radius w_2 of the mask is usually chosen slightly larger than the particle image radius to include eventual tails of the profile. For each image, this calculation is repeated for all local maxima found by the previous dilatation transform. Ideally, each of these pre-located particle candidates correspond to a given brightness profile. Moments of each brightness local distribution can also be calculated to characterize its shape. In particular

$$\mu_2 = \sum_{m^2+n^2 \leq w_2} (m^2 + n^2) \times A_{i+m,j+n} \tag{16}$$

is used to evaluate the characteristic squared size of the brightness spot.

On the previous filtered image bim, you can apply the routine pkfnd to find the local maximum brightness (with a resolution of 1 pixel):

```
>> pk = pkfnd(bim,10,31);
```

where the first number is the brightness threshold and the second number is the minimum distance between hits. The list of (x, y) position is stored in the variable pk. The localization is then refine to subpixel resolution by using

```
>> cnt = cntrd(bim,pk,21);
```

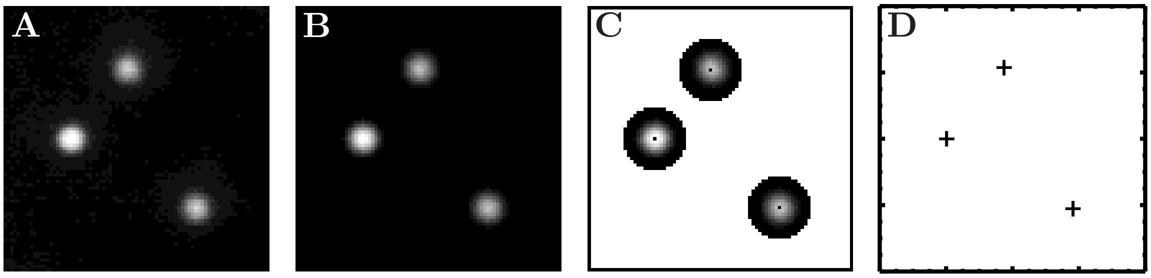


Figure 6: Successive steps of feature location in the tracking algorithms. A) is a typical raw frame from a fluorescent colloidal video microscopy movie. B) is the result of the image restoration from the raw frame. C) illustrates how circular masks are centered on the local brightness maximum and used to clip the individual spot to calculate brightness integrated parameter, such as the center position of the intensity profile or its squared radius of gyration. D) shows the resulting subpixel resolution feature localizations

where the number is the size of the mask w_2 used the calculate the brightness profile moments. This return in the array `cnt` the refined (x, y) estimates, as well as the total brightness μ_0 in the third column and the squared radius of gyration μ_2 in the fourth column.

To see the result of the feature location, you can type

```
>> image(im); hold on;
>> plot(cnt(:,1), cnt(:,2), 'rs');
```

too overlay to detected positions and the original image. You will see that occasionally some spurious features will be detected. In general, either these feature will be discarded by the next step of building the trajectories or will not be statistically relevant. Some more advanced techniques can be used to remove false data. See a detailed description of these methods in Ref. [6].

When performing this feature location in all frames of a movie by using the procedure `pretrack`, a fifth column containing the frame number (i.e. the time) is added to the resulting array (see main text).

C.3 Building Trajectories

Once particles were located in every frame of the movie, found locations must be linked into trajectories. In other words, a given location in a given frame (i.e. at a given time) is assigned a label and the same label must be assigned to the found position corresponding to the same detected particle in the next frame (that is, a priori among all candidate locations found in this frame). In practice, this is done by setting a cutoff w_3 in the maximum possible displacement between successive frame. Hence, the next position is found in a vicinity of a characteristic length scale w_3 around the current position. This is repeated for all located feature in all frame. The histogram of all displacements between successive frames is then plotted to check if w_3 is an appropriate cutoff on the distribution, namely to verify that w_3 does not clip a non-negligible number of event in the histogram. Naturally, the typical distance a particle moves between successive frames must be significantly smaller than the typical inter-particle separation in order to build valid trajectories. This is generally the case at the low volume fraction of probes used in the sample. The resulting output of the tracking routines is a list of x and y positions, the corresponding total brightness and radius of gyration of the particle profile, the time t and particle identities number in the sixth column.

References

[1] J. C. Crocker and D. G. Grier. Methods of digital video microscopy for colloidal studies. *J. Colloid Interface Sci.*, 179:298–310, 1996.

[2] R. G. Larson. *The Structure and Rheology of Complex Fluids.* Oxford University Press, Oxford, UK, 1999.

[3] T. G. Mason. Estimating the viscoelastic moduli of complex fluids using the generalized stokes-einstein equation. *Rheol. Acta*, 39:371–378, 2000.

[4] T. G. Mason, K. Ganesan, J. H. van Zanten, D. Wirtz, and S. C. Kuo. Particle tracking microrheology of complex fluids. *Phys. Rev. Lett.*, 79:3282–3285, 1997.

[5] T. G. Mason and D. A. Weitz. Optical measurements of frequency-dependent linear viscoelastic moduli of complex fluids. *Phys. Rev. Lett*, 74:1250–1253, 1995.

[6] T. Savin. *Multiple particle tracking to assess the microstructure of biological fluids.* PhD thesis, Massachusetts Institute of Technology, 2006.

[7] T. Savin and P. S. Doyle. Static and dynamic errors in particle tracking microrheology. *Biophys. J.*, 88:623–638, 2005.