Supporting Information for "Mean Square Displacement Analysis of Single-Particle Trajectories with Localization Error: Brownian Motion in Isotropic Medium" by X. Michalet

Methods

1. Simulations

Brownian diffusion simulations were performed using custom software written in LabVIEW 8.5 (National Instruments, Austin, TX). Simulated trajectories were saved as text files for further processing (see Data Analysis). A trajectory consisted of a succession of positions $\{(x_i, y_i)\}_{1 \le i \le N}$, where N is the number of positions in the trajectory. We used $N = 1,000$ and extracted subtrajectories when smaller number of positions were needed. The software was scriptable, which made it easy to generate large number N_S of simulations for each set of parameters $(\tilde{D}, \Delta t, \sigma)$. We used $N_S = 1,000$ to obtain small statistical errors in the computation of averaged quantities such as the MSD or MSD SDV. For simulations without microsteps, the successive positions of a given trajectory were calculated according to the following algorithm:

$$\begin{cases} \tilde{x}_{i+1} = \tilde{x}_i + \Delta x_i; & x_i = \tilde{x}_i + \chi_i \\ \tilde{y}_{i+1} = \tilde{y}_i + \Delta y_i; & y_i = \tilde{y}_i + \eta_i \end{cases}, \qquad (M.1)$$

where $(\tilde{x}_i, \tilde{y}_i)$ represents a position in the "real" trajectory (without localization uncertainty) and (x_i, y_i) the corresponding position in the "observed" one. In Eq. (M.1), $(\Delta x_i, \Delta y_i) \sim N(0, \sqrt{2\tilde{D}\Delta t})$ and $(\chi_i, \eta_i) \sim N(0, \sigma)$, where $N(0, \alpha)$ indicates a normal distribution with mean zero and standard deviation α and the symbol "~" indicates that the random variables to the left are distributed according to the probability distribution function to the right. We used LabVIEW's default generator of normally distributed random numbers.

For simulation with m microsteps between each position, each "real" trajectory position $(\tilde{\tilde{x}}_i, \tilde{\tilde{y}}_i)$ was calculated following the algorithm described in Appendix C (Eq. (C.5)), where each $(\Delta x_i, \Delta y_i) \sim N(0, \sqrt{2\tilde{D}\delta t})$ and $\Delta t = m\delta t$. The "observed" trajectory $(\bar{x}_i, \bar{y}_i)$ was obtained by adding random variables $(\chi_i, \eta_i) \sim N(0, \sigma)$ to each position as before to account for localization uncertainty:

$$\begin{cases} \tilde{\tilde{x}}_{i+1} = \tilde{\tilde{x}}_i + \sum_{j=1}^{m-1} \frac{m-j}{m} \Delta x_j^{(i+1)}; & \bar{x}_i = \tilde{\tilde{x}}_i + \chi_i \\ \tilde{\tilde{y}}_{i+1} = \tilde{\tilde{y}}_i + \sum_{j=1}^{m-1} \frac{m-j}{m} \Delta y_j^{(i+1)}; & \bar{y}_i = \tilde{\tilde{y}}_i + \eta_i \end{cases}. \qquad (M.2)$$

All the simulations results presented here were performed using no microsteps.

2. Data Analysis

Trajectory MSD curves and MSD fits were computed using custom software written in LabVIEW (AsteriX). The software is used routinely in our lab to track single particle trajectories and analyze the resulting trajectories according to different approaches (including MSD analysis)[1]. The software is scriptable, so that generating the MSD curve

of many trajectories is easy. Similarly, fitting a MSD curve with a variable number of fitting points and exporting the results in a single file can be scripted. Finally, the same software was used to compute the average values of terms in Eq. (D.9) corresponding to many trajectories, or other averaged quantities discussed in the text.

Raw results of these analyses (MSD curve, fit parameters, etc) were exported as text files to be further processed in Origin 8.1 (OriginLab, Northampton, MA) and prepare figures for publication. Comparison of simulation results with theoretical results such as for the MSD SDV (Eq. (D.17)) or error on fitted parameters as a function of number of fitting points (Eq. (F.17), (F.19) and (F.23)), were performed in Origin using the code provided in Appendix G.

Appendix A: Contribution of camera exposure to the localization error

The effect of camera exposure on localization uncertainty is noticeable in the presence of diffusion or drift of either the particle itself or of the microscope setup. Here, we will address the effect of diffusion only, as this is the only regime studied in this article. For the derivation of this effect, we will not concern ourselves with subtleties arising from pixel size or camera noise, as they can easily be incorporated to the final result. Intuitively, the effect of diffusion is to broaden the PSF of the particle[2]. Since the localization uncertainty is typically proportional to the width of the PSF, we expect the uncertainty to increase with diffusion and exposure time t_E. To quantify this effect, we model the localization procedure as a simple measure of the barycenter of the positions of all photons emitted during the exposure time. More sophisticated approaches based on PSF fitting would lead to identical results.

We will suppose that the probe emits p detected photons during the exposure time t_E. Limiting ourselves to a single axis, the barycenter coordinate is given by:

$$\bar{x} = \frac{1}{p}\sum_{i=1}^{p} x_i^{(p)} , \qquad (A.1)$$

where $x_i^{(p)}$ are the photons individual coordinates (the subscript p standing for "photon"). Each photon's measured coordinate depends on two phenomena: diffraction and diffusion. Due to diffraction, a probe located at a precise position $(x(t), y(t))$ will appear as an extended PSF, which therefore represents the probability density function (PDF) of each photon localization. Due to diffusion, a probe initially located at a position $(x(t_0), y(t_0))$ will move around during the integration time t_E. The location of each photon can therefore be considered as the sum of two independent random variables:

$$x_i^{(p)} = x_i^{(m)} + \xi_i^{(p)} , \qquad (A.2)$$

where $x_i^{(m)}$ is the coordinate of the *molecule* at time t_i when the photon was detected (its PDF is governed by diffusion) and $\xi_i^{(p)}$ is a random variable distributed according to the PSF of the microscope and can be viewed as the coordinate of the *photon* in a frame attached to the moving particle. Eq. (A.1) can be rewritten:

$$\bar{x} = \frac{1}{p}\sum_{i=1}^{p} \left(x_i^{(m)} + \xi_i^{(p)} \right) \qquad (A.3)$$

It is well known that the PSF of most microscopes can be satisfactorily approximated by a Gaussian function of standard deviation s_0 given by:

$$s_0 = 0.21\frac{\lambda}{NA}, \tag{A.4}$$

where λ is the emission wavelength and NA the numerical aperture of the objective lens. Therefore:

$$P_\xi\left(\xi_i^{(p)}\right) = \frac{1}{\sqrt{2\pi}s_0}\exp\left(-\frac{\xi_i^{(p)2}}{2s_0^2}\right). \tag{A.5}$$

To compute the average position $\overline{x}$, we need to express Eq. (A.3) in terms of independent random variables with known PDFs. Therefore we rewrite:

$$
\begin{aligned}
x_i^{(m)} &= \left(x_1^{(m)} - x_0^{(m)}\right) + \left(x_2^{(m)} - x_1^{(m)}\right) + \ldots + \left(x_i^{(m)} - x_{i-1}^{(m)}\right) \\
&= \delta x_1^{(m)} + \delta x_2^{(m)} + \ldots + \delta x_i^{(m)}
\end{aligned}
\tag{A.6}
$$

where the $\delta x_i^{(m)}$ are independent Gaussian-distributed random variables with variance:

$$\operatorname{var}\left(\delta x_i^{(m)}\right) = 2\tilde{D}\left(t_i - t_{i-1}\right) = 2\tilde{D}\delta t_i, \tag{A.7}$$

where D is the diffusion coefficient and δt_i the time interval since the previous photon was detected:

$$P_{\delta x}\left(\delta x_i^{(m)}\right) = \frac{1}{\sqrt{4\pi\tilde{D}\delta t_i}}\exp\left(-\frac{\delta x_i^{(m)2}}{2\tilde{D}\delta t_i}\right). \tag{A.8}$$

The sum in Eq. (A.3) thus appears as a sum of independent Gaussian random variables of mean zero (we can assume without loss of generality that the position x_0 of the molecule at time t_0 is the origin, $x_0 = 0$). Therefore, the average position is also Gaussian-distributed with mean zero and variance s equal to the sum of all variances:

$$s^2 = \frac{1}{p}\sum_{i=1}^{p}\left(2\tilde{D}\sum_{j=1}^{i}\delta t_j + s_0^2\right) = s_0^2 + 2\tilde{D}\left(\frac{1}{p}\sum_{i=1}^{p}t_i\right) = s_0^2 + 2\tilde{D}\overline{t}. \tag{A.9}$$

We now need to compute the average time $\overline{t}$. In most cases, photon emission can be assumed to follow a Poisson process characterized by a constant emission rate R (in counts/s or cps), for which the interval between photons is distributed exponentially:

$$p\left(\delta t_i\right) = R.\exp\left(-R\delta t_i\right). \tag{A.10}$$

In this case, the mean time between photon:

$$\overline{t} = \frac{1}{p}\sum_{i=1}^{p}t_i \tag{A.11}$$

is approximately distributed as a Gamma (or Erlang) law of parameters $(p, 2R)$:

$$P\left(\overline{t}\right) \sim \frac{2R}{(p-1)!}\left(2R\overline{t}\right)^{p-1}\exp\left(-2R\overline{t}\right). \tag{A.12}$$

The mean and standard deviation of this distribution are:

$$
\begin{aligned}
\langle\overline{t}\rangle &= \frac{p}{2R} = \frac{t_E}{2} \\
SDV\left(\overline{t}\right) &= \frac{\sqrt{p}}{2R} = \frac{t_E}{2\sqrt{p}}
\end{aligned}
\tag{A.13}
$$

This distribution is approximately Gaussian for $p > 10$, with a width which decreases rapidly for large number of photons, justifying using $\langle \bar{t} \rangle$ instead of $\bar{t}$ in Eq. (A.9)[1]. We obtain that the combined effects of camera exposure t_E and particle diffusion $\tilde{D}$ is to increase the dimension of the PSF according to:

$$s = \sqrt{s_0^2 + \tilde{D}t_E} \ . \tag{A.14}$$

The localization uncertainty (Eq. 4), being proportional to s, increases accordingly. These effects will start to be noticeable when $D.t_E > s_0^2$. In other words, to avoid added localization uncertainty due to diffusion, the exposure needs to be smaller than a critical value. For a particle of diameter much smaller than the diffraction limit, using typical values of $\lambda = 520$ nm and $NA = 1.45$, we have $s_0^2 = 5.7 \ 10^{-3} \ \mu m^2$:

$\tilde{D} = 10^{-2} \ \mu m^2/s \rightarrow t_E < 570$ ms
$\tilde{D} = 10^{-1} \ \mu m^2/s \rightarrow t_E < 57$ ms
$\tilde{D} = 1 \ \mu m^2/s \rightarrow t_E < 5.7$ ms etc…

For slowly diffusing molecules ($\tilde{D} < 10^{-2} \ \mu m^2/s$), this effect is negligible for most experiments (in which usually $t_E < 100$ ms). It is only of concern for fast diffusing molecules ($\tilde{D} > 10^{-1} \ \mu m^2/s$), where the effect can be dominant. Note that in this case, the observed image of the particle will most likely depart significantly from a Gaussian intensity distribution anyway, further increasing the uncertainty on localization using standard fitting approaches.

Until now, we have supposed that the particle's diameter d is much smaller than the diffraction limit (typically $d << 100$ nm), in order to be able to model its PSF as a Gaussian with standard deviation s_0 given by Eq. (A.4). This situation is appropriate for single fluorophore, single quantum dot or other small fluorescent nanoparticle or bead tracking. If this were not the case, our derivation would remain valid as long as the particle's PSF could be modeled as a Gaussian with a standard deviation s_0', with s_0' replacing s_0 in formula (A.5), (A.9) and (A.14) above. In case the particle's PSF cannot be modeled by a Gaussian (which would for instance be the case for a fluorescent probe with dimension > 100 nm), the Gaussian probability distribution of photon location (A.5) would have to be replaced by a more appropriate probability distribution, equal to the convolution of the particle emission profile and the microscope PSF.

The results obtained in this article in the absence of camera exposure effects are simply modified by replacing the PSF parameter s_0 by s (Eq. (A.14)) in the formulas giving the localization uncertainty as a function of s_0 (Eq. 2).

Appendix B: Probability density function of the square displacement
To compute the effect of localization uncertainty on the MSD, we decompose each square displacement corresponding to a time lag of $n\Delta t$ in terms of independent random variables:

---

[1] Note that if we do not assume the photons to be emitted according to a Poisson process, but simply emitted periodically at a frequency R, we still recover the same value for $\langle \bar{t} \rangle$.

$$\left|\vec{r}_{n+i}-\vec{r}_i\right|^2 = \left(x_{n+i}-x_i\right)^2 + \left(y_{n+i}-y_i\right)^2$$
$$= \left(\tilde{x}_{n+i}+\chi_{n+i}-\tilde{x}_i-\chi_i\right)^2 + \left(\tilde{y}_{n+i}+\eta_{n+i}-\tilde{y}_i-\eta_i\right)^2 \tag{B.1}$$

where (x_i, y_i) and (x_{n+i}, y_{n+i}) are the measured coordinates of the particle at time $i\Delta t$ and $(n+i)\Delta t$ and the true coordinates of the particle are indicated by a tilde sign. Localization uncertainties on each coordinate are indicated by Greek letters and are supposed to be Gaussian distributed. For instance:

$$P_\chi\left(\chi_k\right) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\chi_k^2}{2\sigma^2}\right), \tag{B.2}$$

and identically for the η_k's. We can rewrite Eq. (B.1) as:

$$\left|\vec{r}_{n+i}-\vec{r}_i\right|^2 = \left(\tilde{d}\cos\varphi+\chi_{n+i}-\chi_i\right)^2 + \left(\tilde{d}\sin\varphi+\eta_{n+i}-\eta_i\right)^2$$
$$= \tilde{d}^2 + \chi_{n+i}^2 + \chi_i^2 + \eta_{n+i}^2 + \eta_i^2 + C.T. \tag{B.3}$$

where we have defined (see Fig. S5):

$$\begin{cases} \tilde{x}_{n+i}-\tilde{x}_i = \tilde{d}\cos\varphi \\ \tilde{y}_{n+i}-\tilde{y}_i = \tilde{d}\sin\varphi \end{cases} \tag{B.4}$$

and C.T. indicates products of odd powers of uncorrelated random variables. The localization uncertainties are independent random variables, distributed as defined in Eq. (B.2), while the true distance $\tilde{d}=\left|\tilde{\vec{r}}_{n+i}-\tilde{\vec{r}}_i\right|$ is distributed according to:

$$P_{\tilde{d}}(u) = \frac{2}{4\tilde{D}n\Delta t}u\exp\left(-\frac{u^2}{4\tilde{D}n\Delta t}\right) = \frac{2}{n\alpha}u\exp\left(-\frac{u^2}{n\alpha}\right). \tag{B.5}$$

where we have introduced the notation:

$$\alpha = 4\tilde{D}\Delta t \tag{B.6}$$

The angle φ's probability density function (PDF) is uniform in $[0, 2\pi]$. By definition of the kth moments of $\left|\vec{r}_{n+i}-\vec{r}_i\right|^2$:

$$\left\langle\left|\vec{r}_{n+i}-\vec{r}_i\right|^{2k}\right\rangle = \int_{-\infty}^{+\infty}d\chi_i\int_{-\infty}^{+\infty}d\chi_{n+i}\int_{-\infty}^{+\infty}d\eta_i\int_{-\infty}^{+\infty}d\eta_{n+i}\int_0^{+\infty}d\tilde{d}\int_0^{2\pi}d\varphi\ \Psi\left(\tilde{d},\varphi,\chi_i,\chi_{n+i},\eta_i,\eta_{n+i}\right)\left|\vec{r}_{n+i}-\vec{r}_i\right|^{2k} \tag{B.7}$$

where the joint probability of the random variables $\tilde{d},\varphi,\chi_i,\chi_{n+i},\eta_i,\eta_{n+i}$ is given by:

$$\Psi\left(\tilde{d},\varphi,\chi_i,\chi_{n+i},\eta_i,\eta_{n+i}\right) = \frac{2}{n\alpha}\tilde{d}\exp\left(-\frac{\tilde{d}^2}{n\alpha}\right)\frac{1}{2\pi}\frac{1}{4\pi^2\sigma^4}\exp\left(-\frac{\chi_i^2+\chi_{n+i}^2+\eta_i^2+\eta_{n+i}^2}{2\sigma^2}\right) \tag{B.8}$$

For $k = 1$, it is easy to verify that the cross terms of Eq.(B.3) average out to zero, leaving:

$$\left\langle\left|\vec{r}_{n+i}-\vec{r}_i\right|^2\right\rangle = 4\tilde{D}n\Delta t + 4\sigma^2 = n\alpha + \varepsilon, \tag{B.9}$$

where we have introduced the notation:

$$\varepsilon = 4\sigma^2. \tag{B.10}$$

The mean of the MSD for a time lag $n\Delta t$ follows immediately:

$$\left\langle\rho_n\right\rangle = \left\langle\frac{1}{K}\sum_{i=1}^K\left|\vec{r}_{n+i}-\vec{r}_i\right|^2\right\rangle = \frac{1}{K}\sum_{i=1}^K\left\langle\left|\vec{r}_{n+i}-\vec{r}_i\right|^2\right\rangle = \left\langle\left|\vec{r}_{n+i}-\vec{r}_i\right|^2\right\rangle = n\alpha + \varepsilon \tag{B.11}$$

In other words, localization uncertainty introduces a constant offset $\varepsilon = 4\sigma^2$ to the MSD curve. This result has already been established using different methods by several authors (e.g. [3-4]).

Using the same approach and slightly more lengthy algebra, one can verify that:

$$\left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^4 \right\rangle = 2\left(n\alpha + \varepsilon\right)^2 , \tag{B.12}$$

giving the variance of the square displacement (SD):

$$\left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^4 \right\rangle - \left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^2 \right\rangle^2 = \left(n\alpha + \varepsilon\right)^2 . \tag{B.13}$$

Note that the variance of the *mean* square displacement (MSD) is more complex to calculate. This calculation is described in Appendix D.

Eq. (B.9) & (B.12) for the first two moments of the square displacement suggest that the PDF of the SD is actually:

$$P_{SD}\left(\left| \vec{r}_{n+i} - \vec{r}_i \right|^2\right) = \frac{1}{n\alpha + \varepsilon} \exp\left(-\frac{\left| \vec{r}_{n+i} - \vec{r}_i \right|^2}{n\alpha + \varepsilon} \right) \tag{B.14}$$

for which one has:

$$\left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^{2k} \right\rangle = k!\left(n\alpha + \varepsilon\right)^k \tag{B.15}$$

In other words, the PDF of the measured squared displacements is identical to that of the real square displacements (Eq. 10) with the simple addition of the offset $\varepsilon = 4\sigma^2$. This can be easily verified by computing further moments of the SD, although the algebra becomes rapidly tedious. Using wxMaxima[5], we verified Eq. (B.15) up to $k = 4$, strongly supporting this conclusion. Eq. (B.14) is also supported by simulations (data not shown).

As pointed out by an anonymous reviewer, this result can alternatively be established using the properties of sums of normally distributed independent random variables. This is the approach used in ref. [6] to obtain the same result. We have presented this alternative approach as a mean to illustrate the calculations involved in the evaluation of the expressions of Eq. (D.9).

Appendix C: Contribution of exposure time (or number of simulation microsteps) to the MSD curve offset

This appendix is limited to diffusion taking place in one dimension. The results in N dimensions are simply multiplied by N. A similar derivation is sketched out in Goulian & Simon[3] and used in Montiel et al.[6] with a different interpretation. Savin & Doyle obtain the same result using an entirely different approach[4]. We detail it here because it is relevant for two different but practically important situations:

(i) an image is acquired by collecting photons emitted by a single probe during an exposure time t_E, equal or smaller than the frame duration Δt:

$$t_E = \lambda \, \Delta t, \qquad 0 < \lambda \leq 1 \tag{C.1}$$

(ii) a simulation is performed by moving the probe in microsteps of duration δt, the number of microsteps per frame (of duration Δt) being m:

$$\Delta t = m \, \delta t \tag{C.2}$$

To simulate a finite exposure time, the position of the probe within a given frame is computed as the average position of a fraction λ of the m microsteps:

$$\bar{x}_q = \left(x_0 + (x_0 + \Delta x_1) + \ldots + (x_0 + \Delta x_1 + \ldots + \Delta x_{q-1})\right)\Big/q$$

$$= x_0 + \sum_{j=1}^{q-1} \frac{q-j}{q} \Delta x_j \tag{C.3}$$

where the $\{\Delta x_j\}_{j=1\ldots q-1}$ represent the random microsteps performed every δt and $q = \lambda m$. For a pure Brownian motion, the Δx_j are independent and Gaussian distributed:

$$\langle \Delta x_i \rangle = 0$$

$$\langle \Delta x_i \Delta x_j \rangle = 2\tilde{D}\,\delta t\,\delta_{ij} \tag{C.4}$$

where $\tilde{D}$ is the probe's diffusion coefficient and δ_{ij} the Kronecker symbol. Note that for other types of motion, these assumptions may be incorrect.

The two situations are similar in the sense that the average position of the probe measured by the image integrated during the exposure t_E is the limit when $q = \lambda m$ and m tend to infinity of the result obtained for the simulation situation.

We will compute the average square displacement between points separated by a number k of frames, indicating the frame by a subscript in parenthesis. The measured positions in these two frames are:

$$\bar{x}_q^{(1)} = x_1 + \sum_{j=1}^{q-1} \frac{q-j}{q} \Delta x_j^{(1)}$$

$$\bar{x}_q^{(k+1)} = x_{k+1} + \sum_{j=1}^{q-1} \frac{q-j}{q} \Delta x_j^{(k+1)} \tag{C.5}$$

where the position at the beginning of the exposure of frame $k+1$ can be obtained by recurrence by summing over all displacements during the previous frame:

$$x_{k+1} = x_k + \sum_{j=1}^{m} \Delta x_j^{(k)}. \tag{C.6}$$

Therefore

$$x_{k+1} = x_1 + \sum_{i=1}^{k} \sum_{j=1}^{m} \Delta x_j^{(i)}, \tag{C.7}$$

and:

$$\bar{x}_q^{(k+1)} - \bar{x}_q^1 = \sum_{i=1}^{k} \sum_{j=1}^{m} \Delta x_j^{(i)} + \sum_{j=1}^{q-1} \frac{q-j}{q} \Delta x_j^{(k+1)} - \sum_{j=1}^{q-1} \frac{q-j}{q} \Delta x_j^{(1)}$$

$$= \sum_{i=2}^{k} \sum_{j=1}^{m} \Delta x_j^{(i)} + \sum_{j=1}^{q-1} \frac{q-j}{q} \Delta x_j^{(k+1)} + \sum_{j=1}^{q-1} \frac{j}{q} \Delta x_j^{(1)} + \sum_{j=q}^{m} \Delta x_j^{(1)}. \tag{C.8}$$

The average square displacement is thus:

$$\left\langle \left(\bar{x}_q^{(k+1)} - \bar{x}_q^1\right)^2 \right\rangle = \left\langle \left(\sum_{i=2}^{k} \sum_{j=1}^{m} \Delta x_j^{(i)}\right)^2 + \left(\sum_{j=1}^{q-1} \frac{q-j}{q} \Delta x_j^{(k+1)}\right)^2 \right.$$

$$\left. + \left(\sum_{j=1}^{q-1} \frac{j}{q} \Delta x_j^{(1)}\right)^2 + \left(\sum_{j=q}^{m} \Delta x_j^{(1)}\right)^2 + C.T. \right\rangle, \tag{C.9}$$

where the cross terms (*C.T.*) contain products $\Delta x_j^{(i)} \Delta x_k^{(l)}$ with $i \neq l$ or $j \neq k$, whose average value is equal to zero according to Eq. (C.4). We can further simplify Eq. (C.9) by noting that the expansion of the squares results in similar products $\Delta x_j^{(i)} \Delta x_k^{(i)}$, whose average value is equal to zero, unless $j = k$. The final result can be expressed as:

$$\left\langle \left(\overline{x}_q^{(k+1)} - \overline{x}_q^1 \right)^2 \right\rangle = \left(\sum_{i=2}^{k} \sum_{j=1}^{m} 1 + \sum_{j=1}^{q-1} \left(\frac{q-j}{q} \right)^2 + \sum_{j=1}^{q-1} \left(\frac{j}{q} \right)^2 + \sum_{j=q}^{m} 1 \right) \left\langle \Delta x^2 \right\rangle. \qquad (C.10)$$

After some algebra, we obtain:

$$\left\langle \left(\overline{x}_q^{(k+1)} - \overline{x}_q^1 \right)^2 \right\rangle = 2\tilde{D}k\Delta t - \frac{(q-1)(q+1)}{3qm} 2\tilde{D}\Delta t. \qquad (C.11)$$

In other words, the MSD curve can be fitted with the usual linear equation to which a *negative offset* has been added due to averaging of the position during the finite exposure:

$$\rho(t) = 2\tilde{D}t - 2\tilde{D} \left[\frac{(q-1)(q+1)}{3qm} \right] \Delta t, \qquad (C.12)$$

which is the result obtained by Goulian & Simon[3], for $N = 1$ dimensions. Montiel *et al.*[6]. interpret this result as meaning that the apparent diffusion coefficient measured from the first two points of the MSD curve is:

$$D_{app} = \tilde{D} \left(1 - \frac{(q-1)(q+1)}{3qm} \right), \qquad (C.13)$$

whereas we have just seen that this is not the case, but instead means that the MSD curve needs to be fitted with an offset, the expected fitted diffusion constant being the true one. For actual imaging experiments, there is no microstep to speak of and the correct limit is $m \rightarrow \infty$, $q = \lambda m \rightarrow \infty$:

$$\rho(t) = 2\tilde{D}t - \frac{2}{3}\lambda\tilde{D}\Delta t = 2\tilde{D}t - \frac{2}{3}\tilde{D}t_E, \qquad (C.14)$$

which is the result obtain by Savin & Doyle using a different approach.

For simulations which use no microsteps ($q = m = 1$) or use microsteps but only use the final position (or any other microstep position) to localize the particle within a frame ($q = 1, m > 1$), the offset disappears ($q = 1$ in Eq. (C.12)). However, for simulations using a finite number of microsteps and computing the particle position as the average of many (or all) microstep positions ($m \geq q > 1$), Eq. (C.12) applies and

$$\rho(t) = 2N\tilde{D}t - \frac{\lambda}{3} \left(1 - \frac{1}{\lambda^2 m^2} \right) 2N\tilde{D}\Delta t, \qquad (C.15)$$

which, for large enough m, reduces to Eq. (C.14). We have reintroduced the number of spatial dimensions, N, in the final result.

There are two take-home messages coming out of this derivation:

(i) For real measurements, it is important to know the ratio of exposure time to frame duration, $\lambda = t_E/\Delta t$. Whatever this value is, a negative offset will result in the MSD curve given by the last term of Eq. (C.14). Proper fitting to obtain an unbiased diffusion constant requires including this offset (and that due to localization uncertainty discussed in Appendix B). This offset will have a particular importance for large diffusion

coefficients. For small diffusion constants, it might be dwarfed by the localization uncertainty term $2\sigma^2$ (for N = 1 dimension) and omission of this negative offset might not bias the results much. For instance, for a typical uncertainty of 30 nm, a value $\lambda = 1$ and frame duration $\Delta t = 100$ ms:

$$2\sigma^2 = 1.18 \ 10^{-3} \, \mu m^2$$

$$\frac{2}{3}\tilde{D}t_E = 6.7 \ 10^{-2} D \ \mu m^2 \text{ ,}$$

where $\tilde{D}$ is expressed in μm^2/s. For $\tilde{D} << 10^{-2}$ μm^2/s, the effect of the exposure offset will be negligible and both the recovered σ and $\tilde{D}$ values will be negligibly biased by its omission. However, for smaller localization uncertainty or longer exposure time, the effect of neglecting the exposure offset will be felt for much smaller $\tilde{D}$'s.

(ii) For simulations, it appears that it is not equivalent to simulate trajectories using a single Gaussian-distributed step of variance $2\,\tilde{D}\,\Delta t$ or multiple Gaussian-distributed microsteps (*m*) with variance $2\,\tilde{D}\,\Delta t/m$, as far as the MSD curve is concerned, unless one uses a single microstep position to report the position of the particle within a frame. Indeed, in the first case (single microstep), one will not observe any negative offset, whereas in the latter and if the reported location of the particle is obtained by averaging several microstep positions, a negative offset will be observed, given by Eq. (C.15). In other words, simulations that use microsteps to be able to handle boundary conditions should be used carefully. In general, it appears best to compute the location of the particle within a frame as an average of all the microstep locations during that frame.

Appendix D: Standard deviation of the mean square displacement

Qian *et al.* have computed the SDV of the mean square displacement ρ_n in the absence of localization uncertainty[7]. Although this result is a useful starting point, it is insufficient when the uncertainty becomes comparable to the elementary displacement (the average displacement taking place during one frame duration).
Defining[2]:

$$K = N - n \text{ ,} \tag{D.1}$$

and the standard deviation σ_n of the MSD ρ_n at time lag $t_n = n\Delta t$ by:

$$\sigma_n^2 = \left\langle \bar{\rho}_n^2 \right\rangle - \left\langle \bar{\rho}_n \right\rangle^2 . \tag{D.2}$$

Using the standard definition of $\bar{\rho}_n$, Eq. 7:

$$\left\langle \bar{\rho}_n^2 \right\rangle = \left\langle \left(\frac{1}{K} \sum_{i=1}^{K} \left| \vec{r}_{n+i} - \vec{r}_i \right|^2 \right)^2 \right\rangle$$

$$= \frac{1}{K} \left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^4 \right\rangle + \frac{2}{K^2} \sum_{i=1}^{K} \sum_{j=1}^{i-1} \left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^2 \left| \vec{r}_{n+j} - \vec{r}_j \right|^2 \right\rangle \tag{D.3}$$

Computation of the average terms in Eq. (D.3) is subtle because of the localization uncertainty and the correlation between average displacements. We have already calculated the first term in Appendix B (Eq. (B.12)).

---

[2] Note that Qian *et al.* use a different notation (ref. 7). Their trajectory has *n*+1 vertices numbered 0…*n*. We consider a trajectory containing *N* vertices numbered 1…*N*.

There is correlation between square displacements $\left|\vec{r}_{n+i}-\vec{r}_i\right|^2$ and $\left|\vec{r}_{n+j}-\vec{r}_j\right|^2$ ($j < i$) only if the two terms contain common intermediate vertices (case (b) of Fig. S6A), i.e, when $i < j+n$. Noting $\vec{R}_1$ the displacement between the first successive vertices, $\vec{R}_2$ the displacement between the next two successive vertices, etc (see Fig. S6A), we obtain:

$$\left\langle\left|\vec{r}_{n+i}-\vec{r}_i\right|^2\left|\vec{r}_{n+j}-\vec{r}_j\right|^2\right\rangle =$$

$$(a)\, j+n<i \qquad\qquad =\left\langle\vec{R}_1^2\vec{R}_3^2\right\rangle \qquad\qquad . \tag{D.4}$$

$$(b)\, j+n\geq i \qquad\qquad =\left\langle\left(\vec{R}_2+\vec{R}_3\right)^2\left(\vec{R}_1+\vec{R}_2\right)^2\right\rangle$$

In case (a), since $\vec{R}_1$ and $\vec{R}_3$ do not have any vertices in common, their decomposition in terms of random variables as in Eq.(B.3)-(B.4) results in independent terms:

$$\vec{R}_1^2 =\left(\tilde{d}_1\cos\varphi_1+\chi_{j+n}-\chi_j\right)^2+\left(\tilde{d}_1\sin\varphi_1+\eta_{j+n}-\eta_j\right)^2$$
$$\vec{R}_3^2 =\left(\tilde{d}_3\cos\varphi_3+\chi_{i+n}-\chi_i\right)^2+\left(\tilde{d}_3\sin\varphi_3+\eta_{i+n}-\eta_i\right)^2 \tag{D.5}$$

where we have introduced intermediate random variables for $\vec{R}_1$ and $\vec{R}_3$ as described in Fig. S5. As in Appendix B, the φ_k's have uniform PDF in $[0, 2\pi]$ χ_k's, the η_k's have Gaussian PDFs given by Eq. (B.2) and the real displacements $\tilde{d}_k$'s have PDFs given by Eq. (B.5) where α is replaced by:

$$\alpha_j = 4\tilde{D}\Delta t\left(i_e^{(j)}-i_s^{(j)}\right)=\left(i_e^{(j)}-i_s^{(j)}\right)\alpha , \tag{D.6}$$

in which the indices $i_e^{(j)}$, $i_s^{(j)}$ represent the end point and starting point of the vector $\vec{R}_j$.
In case (b) of Fig. S6A for instance, we have:

$$\alpha_1 =(i-j)\alpha = 4(i-j)\tilde{D}\Delta t$$
$$\alpha_2 =(n+j-i)\alpha = 4(n+j-i)\tilde{D}\Delta t \qquad\qquad . \tag{D.7}$$
$$\alpha_3 =(n+i-(n+j))\alpha = 4(i-j)\tilde{D}\Delta t = \alpha_1$$

Case (a) of Eq. (D.4) (Fig. S6A (a)) is the simplest to compute since random variables involved in the expression of $\vec{R}_1$ do not appear in that of $\vec{R}_3$. Therefore, the average can be expressed as the product of two separate averages. After integration over all random variable's PDFs, we obtain:

$$\left\langle\vec{R}_1^2\vec{R}_3^2\right\rangle =\left\langle\vec{R}_1^2\right\rangle\left\langle\vec{R}_3^2\right\rangle=\left(\alpha n+\varepsilon\right)^2 . \tag{D.8}$$

In case (b), expansion of the last expression in Eq. (D.4) results in product of terms involving common random variables, resulting in non separation of the averages, contrary to case (a). The calculations are lengthy but straightforward. Since they will be used for other calculations, we report the values of intermediate quantities:

$$\left\langle \vec{R}_i.\vec{R}_j \right\rangle = -\frac{\varepsilon}{2} \qquad\qquad |i-j|=1$$

$$\left\langle \vec{R}_1.\vec{R}_3 \right\rangle = 0$$

$$\left\langle \vec{R}_i^{\,2} \right\rangle = \alpha_i + \varepsilon$$

$$\left\langle \vec{R}_i^{\,4} \right\rangle = 2\left(\alpha_i + \varepsilon\right)^2$$

$$\left\langle R_i^2 R_j^2 \right\rangle = \left(\alpha_i + \varepsilon\right)\left(\alpha_j + \varepsilon\right) + \frac{\varepsilon^2}{4} \qquad\qquad |i-j|=1$$

$$\left\langle R_1^2 R_3^2 \right\rangle = \left(\alpha_1 + \varepsilon\right)\left(\alpha_3 + \varepsilon\right) + \frac{\varepsilon^2}{4}\delta_{j,i+n}$$

$$\left\langle \vec{R}_i.\vec{R}_j R_k^2 \right\rangle = -\varepsilon\left(\alpha_k + \varepsilon\right) \qquad\qquad i \neq j; k \in \{i,j\}$$

$$= -\frac{\varepsilon}{2}\left(\alpha_k + \varepsilon\right) \qquad\qquad i \neq j; j \neq k; k \neq i$$

$$\left\langle \vec{R}_1.\vec{R}_3 R_2^2 \right\rangle = \frac{\varepsilon^2}{4}$$

$$\left\langle \vec{R}_1.\vec{R}_2 \times \vec{R}_2.\vec{R}_3 \right\rangle = \frac{3}{8}\varepsilon^2 \qquad\qquad\qquad\qquad (D.9)$$

Note that the case $i = j + n$ where the vector $\vec{R}_2 = \vec{0}$, needs a special treatment, since $\vec{R}_1$ and $\vec{R}_3$ have a common vertex, which introduces higher powers of the localization uncertainty random variables.

Using the above results and putting all terms together, we obtain finally:

$$\left\langle |\vec{r}_{n+i} - \vec{r}_i|^2 |\vec{r}_{n+j} - \vec{r}_j|^2 \right\rangle - \left\langle |\vec{r}_{n+i} - \vec{r}_i|^2 \right\rangle\left\langle |\vec{r}_{n+j} - \vec{r}_j|^2 \right\rangle = \alpha_2^2 + \frac{\varepsilon^2}{4}\delta_{j,i+n}$$

$$= \left\langle \vec{R}_2^{\,4} \right\rangle - \left\langle \vec{R}_2^{\,2} \right\rangle^2 + \frac{\varepsilon^2}{4}\delta_{j,i+n}$$

$$= \left\langle |\vec{r}_{n+j} - \vec{r}_i|^4 \right\rangle - \left\langle |\vec{r}_{n+j} - \vec{r}_i|^2 \right\rangle^2 + \frac{\varepsilon^2}{4}\delta_{j,i+n} \qquad (D.10)$$

$$= \alpha^2\left(i - n - j\right)^2 + \frac{\varepsilon^2}{4}\delta_{j,i+n}$$

This result is identical to that used by Qian *et al.* (Hong Qian, private communication), obtained in the absence of localization uncertainty[7], except for the last term in the right hand side of Eq. (D.10). In other words, although none of the individual cross product terms or term differences is trivial (see Eq. (D.9)), the sum of most terms cancels out, leaving only the contribution of the terms for which $j = i + n$, i.e. terms contributed by displacements having a common end vertex.

This result in hand, we can now tackle the computation of the variance of the square displacement:

$$\sigma_n^2 = \frac{1}{K}\left(\left\langle \left|\vec{r}_{n+i}-\vec{r}_i\right|^4\right\rangle - \left\langle \left|\vec{r}_{n+i}-\vec{r}_i\right|^2\right\rangle^2\right) +$$

$$+\frac{2}{K^2}\sum_{i=1}^{K}\sum_{j=1}^{i-1}\left(\left\langle \left|\vec{r}_{n+i}-\vec{r}_i\right|^2\left|\vec{r}_{n+j}-\vec{r}_j\right|^2\right\rangle - \left\langle \left|\vec{r}_{n+i}-\vec{r}_i\right|^2\right\rangle\left\langle \left|\vec{r}_{n+j}-\vec{r}_j\right|^2\right\rangle\right), \quad \text{(D.11)}$$

$$=\frac{1}{K}\left(\alpha n+\varepsilon\right)^2+\frac{2}{K^2}\left(\underbrace{\sum 0}_{(a)}+\underbrace{\sum\left(\alpha^2\left(i-n-j\right)^2+\frac{\varepsilon^2}{4}\delta_{j,i+n}\right)}_{(b)}\right)$$

where we have split the double sum in two parts (a) and (b) corresponding to (a) $n < i \le K$ and (b) $i \le n$ (see Fig. S6B). Obviously, if $n > K$, part (a) does not exist and we are left with part (b). If $n \le K$, the computation of part (b) requires splitting the double sum in three parts (b$_1$), (b$_2$) and their boundary ($j = i - n$), as illustrated in Fig. S6B:

$$\underbrace{\sum}_{(b)} = \sum_{i=1}^{n-1}\sum_{j=1}^{i-1} + \sum_{i=n}^{K}\sum_{j=i-n+1}^{i-1} + \sum_{\substack{i=n\\j=i-n}}^{K} \qquad \text{(D.12)}$$

Finally, our result for the variance of the mean square displacement is:

$$\sigma_n^2 = \frac{n}{6K^2}\left(4n^2K+2K-n^3+n\right)\alpha^2+\frac{1}{K}\left(2n\alpha\varepsilon+\left(1+\frac{1}{2}\left(1-\frac{n}{K}\right)\right)\varepsilon^2\right), \quad n \le K = N-n$$

$$\text{(D.13)}$$

$$=\frac{1}{6K}\left(6n^2K-4nK^2+4n+K^3-K\right)\alpha^2+\frac{1}{K}\left(2n\alpha\varepsilon+\varepsilon^2\right), \qquad n > K = N-n$$

recovering Qian *et al.*'s result when $\varepsilon = 0$ (no localization uncertainty)[3]. For future computations, we rewrite:

$$\sigma_n^2 = \alpha^2/f\left(n,N,x\right), \qquad \text{(D.14)}$$

defining the function *f(n, N, x)*:

$$f\left(n,N,x\right)^{-1} = \frac{n}{6K^2}\left(4n^2K+2K-n^3+n\right)+\frac{1}{K}\left(2nx+\left(1+\frac{1}{2}\left(1-\frac{n}{K}\right)\right)x^2\right), \quad n \le K$$

$$\text{(D.15)}$$

$$=\frac{1}{6K}\left(6n^2K-4nK^2+4n+K^3-K\right)+\frac{1}{K}\left(2nx+x^2\right), \qquad n > K$$

with:

$$K = N - n$$

$$x = \frac{\varepsilon}{\alpha} = \frac{\sigma^2}{\tilde{D}\Delta t}\cdot \qquad \text{(D.16)}$$

We checked this result with simulations. The experimental variance of the mean square displacement ρ_n was calculated for $N_S = 1000$ simulated trajectories according to:

$$\text{var}_{\text{exp}}\left(\bar{\rho}_n\right) = \frac{1}{N_S}\sum_{i=1}^{N_S}\left(\bar{\rho}_n^{(i)}\right)^2-\left(\frac{1}{N_S}\sum_{i=1}^{N_S}\bar{\rho}_n^{(i)}\right)^2 \qquad \text{(D.17)}$$

where $\bar{\rho}_n^{(i)}$ is the MSD of the ith simulation at time lag $n\Delta t$. As shown in Fig. 3A & S1, there is an excellent agreement between Eq. (D.13) and the result of formula (D.17) for different values of ε and α.

---

[3] Note that, as pointed out by an anonymous reviewer, a more concise derivation of Eq. (D.13) can be obtained using the formalism of ref. 8.

The effect of terms in x in Eq. (D.15) is illustrated in Fig. 3A, which represents the ratio σ_n/ρ_n for different values of x (including $x = 0$, the result in the absence of localization uncertainty obtained by Qian *et al.*). Clearly, neglecting x severely underestimates the variance of the MSD, especially at short time lags.

Appendix E: Probability density function of the mean square displacements
With the non-standard definition Eq. 8 of the MSD, the independence of the square displacements allows the PDF of the MSD, ρ_n, to be computed as successive convolutions of the PDF of the individual square displacements, $d_n{}^7$:

$$p_{d^2}(v) = \frac{1}{n\alpha + \varepsilon}\exp\left(-\frac{v}{n\alpha + \varepsilon}\right), \tag{E.1}$$

The convolution of exponential PDFs can be shown to be Gamma distributed:

$$P_{\rho_n}(u) = \frac{K}{n\alpha + \varepsilon}\left(\frac{K}{n\alpha + \varepsilon}u\right)^{K-1}\frac{e^{-\frac{K}{n\alpha+\varepsilon}u}}{(K-1)!} \tag{E.2}$$

where:

$$K = E(N/n), \tag{E.3}$$

is the number of displacements used in the mean square displacement average.
Introducing the *reduced mean square displacement* random variable:

$$z_n = \frac{\rho_n}{\alpha}, \tag{E.4}$$

the PDF of z_n can be rewritten:

$$\gamma_{\lambda,K}(z_n) = \lambda(\lambda z_n)^{K-1}\frac{e^{-\lambda z_n}}{(K-1)!}, \tag{E.5}$$

where we have introduced λ:

$$\lambda = \frac{K}{n+x}. \tag{E.6}$$

The mean and variance of the Gamma distribution are:

$$\mu_{\lambda,K} = \frac{K}{\lambda} = n+x$$

$$\sigma_{\lambda,K}^2 = \frac{K}{\lambda^2} = \frac{(n+x)^2}{K} \tag{E.7}$$

In the case of the standard definition Eq. 7 of the MSD, the correlation between displacements renders the computation of the MSD PDF very cumbersome. We can however try to guess a functional form for this PDF, based on the known properties of the MSD derived in this article. For instance, we know that the mean and variance of the MSD are (Eq. (D.14)):

$$\rho_n = \alpha(n+x)$$

$$\sigma_n^2 = \alpha^2/f(n,N,x) \tag{E.8}$$

Using the reduced variable z_n, the reduced mean and variance read:

$$\langle z_n \rangle = n + x$$

$$\hat{\sigma}_n^2 = \frac{(n+x)^2}{K(n,N,x)} \qquad . \tag{E.9}$$

where we have defined:

$$K(n,N,x) = (n+x)^2 f(n,N,x). \tag{E.10}$$

K(n, N, x) is represented for various values of *x* in Fig. S7 along $K = N - n$, the number of displacements in the standard definition Eq. 7 of $\bar{\rho}_n$.

Comparing Eq. (E.9) and Eq. (E.7), we see that the test function $\gamma_{\lambda,K}(z_n)$ (Eq. (E.5)) in which *K* is defined as in Eq. (E.10) would have the correct mean and standard deviation. It appears that this guess function perfectly describes simulated data for $x = 0$ (no localization error), as illustrated in Fig. S8. In other words, the effect of correlations between displacements in Eq. 7 does not change the functional form of the MSD PDF, which is a Gamma distribution, but reduces the effective number of "independent" displacements *K(n, N, x)*, as shown in Fig. S7. Note that for large *x* and small *n*, *K(n, N, x)* behaves as (n << x):

$$K(n,N,x) \sim \frac{x^2}{1+2nx+\frac{3}{2}x^2} K \sim \frac{2}{3} K . \tag{E.11}$$

In other words, the localization uncertainty is "erasing" the correlation with (or memory of) previous displacements. However, it also reduces the effective number of independent displacement by up to a factor 2/3.

Interestingly, this guess function still does an excellent job of describing the MSD PDF when $x > 0$, as shown in Fig. S9 for $x = 10$.

Note that, for small *n* values, the MSD PDF is very symmetric and can be approximated by a Gaussian distribution of mean and standard deviation given by Eq. (E.9).

In summary, we have shown than the MSD PDF is well described by a Gamma distribution (Eq. (E.5)), whose parameters (λ, *K*) are given by Eq. (E.6) and (E.10). For small time lags, the distribution is very symmetric (quasi Gaussian), while it becomes more asymmetric as *n* increases. This implies that using the least-square fit approach to extract the diffusion parameters is perfectly legitimate as long as the number of points of the MSD used for this purpose is such that the MSD PDF is symmetric. For larger number of MSD points, the least-square fit approach conditions are less well satisfied, although not dramatically so.

Appendix F. Error on parameters of least-square fits of the MSD
Least-square fit to a line (or more generally, to a polynomial) amounts to solving a system of linear equations for the fit parameters[9]. Analytical expressions for the fit parameters can easily be obtained. For a linear fit,

$$\rho(t) = a + bt \tag{F.1}$$

The solutions are:

$$a = \frac{1}{\Delta}\left(\sum_{i=1}^{p}\frac{t_i^2}{\sigma_i^2}\sum_{i=1}^{p}\frac{\rho_i}{\sigma_i^2} - \sum_{i=1}^{p}\frac{t_i}{\sigma_i^2}\sum_{i=1}^{p}\frac{t_i\rho_i}{\sigma_i^2}\right)$$

$$b = \frac{1}{\Delta}\left(\sum_{i=1}^{p}\frac{1}{\sigma_i^2}\sum_{i=1}^{p}\frac{t_i\rho_i}{\sigma_i^2} - \sum_{i=1}^{p}\frac{t_i}{\sigma_i^2}\sum_{i=1}^{p}\frac{\rho_i}{\sigma_i^2}\right) \qquad \text{(F.2)}$$

$$\Delta = \sum_{i=1}^{p}\frac{1}{\sigma_i^2}\sum_{i=1}^{p}\frac{t_i^2}{\sigma_i^2} - \left(\sum_{i=1}^{p}\frac{t_i}{\sigma_i^2}\right)^2$$

where the ($t_i = i.\Delta t$, ρ_i)'s are the coordinates of the experimental MSD data points and the (σ_i)'s are the standard deviations (SDV) for the (ρ_i)'s. p is the number of points of the MSD curve used for the fit ($2 \leq p < N$). Note that Eq. (F.2) corresponds to a weighted fit, in which each data point ρ_i is weighted by the inverse of its variance $1/\sigma_i^2$. We will address the case of an unweighted fit later in this section.

The standard error σ_a on the fitting parameter a (intercept) is given by:

$$\sigma_a^2 \simeq \sum_{i=1}^{p}\sigma_i^2\left(\frac{\partial a}{\partial\rho_i}\right)^2 + 2\sum_{i=1}^{p}\sum_{j<i}\sigma_{ij}^2\left(\frac{\partial a}{\partial\rho_i}\right)\left(\frac{\partial a}{\partial\rho_j}\right) + ..., \qquad \text{(F.3)}$$

with a similar expression for the standard deviation σ_b of the fitting parameter b (slope)[4]:

$$\sigma_b^2 \simeq \sum_{i=1}^{p}\sigma_i^2\left(\frac{\partial b}{\partial\rho_i}\right)^2 + 2\sum_{i=1}^{p}\sum_{j<i}\sigma_{ij}^2\left(\frac{\partial b}{\partial\rho_i}\right)\left(\frac{\partial b}{\partial\rho_j}\right) + ... \qquad \text{(F.4)}$$

For a pure Brownian motion with localization error and finite exposure time:

$$a = \varepsilon$$
$$b = 4D \qquad \text{(F.5)}$$

The errors on a and b are a bit more tedious to compute and require the knowledge of the covariance terms σ_{ij}^2.

Covariance of the mean square displacements
By definition:

$$\sigma_{nm}^2 = \langle\bar{\rho}_n\bar{\rho}_m\rangle - \langle\bar{\rho}_n\rangle\langle\bar{\rho}_m\rangle. \qquad \text{(F.6)}$$

Noting:

$$K = N - n$$
$$P = N - m \qquad \text{(F.7)}$$

and using the results of Appendix D, we have:

---

[4] Note that in the particular case of a linear relationship as in Eq. (F.1), expression (F.3) is an exact Taylor expansion, as higher order partial derivatives of the fit parameters are equal to zero.

$$\left\langle \overline{\rho}_n \right\rangle = \frac{1}{K}\sum_{i=1}^{K}\left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^2 \right\rangle = \alpha n + \varepsilon$$

$$\left\langle \overline{\rho}_m \right\rangle = \frac{1}{P}\sum_{j=1}^{P}\left\langle \left| \vec{r}_{m+j} - \vec{r}_j \right|^2 \right\rangle = \alpha m + \varepsilon \qquad . \qquad (F.8)$$

$$\left\langle \overline{\rho}_n \overline{\rho}_m \right\rangle = \frac{1}{KP}\sum_{i=1}^{K}\sum_{j=1}^{P}\left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^2 \left| \vec{r}_{m+j} - \vec{r}_j \right|^2 \right\rangle$$

As in the calculation of $\left\langle \rho_n^2 \right\rangle$ in Appendix D, we have to consider different cases for the respective order of i, $i + n$, j and $j + m$ when evaluating the bracket expression in the last equation of (F.8). We can assume $m > n$ without loss of generality. We can then distinguish 5 different situations illustrated in Fig. S10, for which the bracket expression will take different forms:

$$\left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^2 \left| \vec{r}_{m+j} - \vec{r}_j \right|^2 \right\rangle =$$

$$(a)\ i\ <\ i+n\ <j\ <\ j+m \qquad \left\langle \vec{R}_1^2 \vec{R}_3^2 \right\rangle$$

$$(b)\ i\ \le j\ \le\ i+n\ <\ j+m \qquad \left\langle \left(\vec{R}_1 + \vec{R}_2\right)^2 \left(\vec{R}_2 + \vec{R}_3\right)^2 \right\rangle$$

$$(c)\ j\ <\ i\ <\ i+n\ <\ j+m \qquad \left\langle \vec{R}_2^2 \left(\vec{R}_1 + \vec{R}_2 + \vec{R}_3\right)^2 \right\rangle \qquad (F.9)$$

$$(d)\ j\ <\ i\ \le j+m\ <\ i+n \qquad \left\langle \left(\vec{R}_2 + \vec{R}_3\right)^2 \left(\vec{R}_1 + \vec{R}_2\right)^2 \right\rangle$$

$$(e)\ j\ <\ j+m\ <\ i\ <\ i+n \qquad \left\langle \vec{R}_3^2 \vec{R}_1^2 \right\rangle$$

which can be computed using the results of (D.9). Note that, as in the calculation of the variance σ_n^2 (Appendix D), one needs to consider separately the cases where $i + n = j$, $i = j$, $i + n = j + m$ or $i = j + m$, in which one of the vectors $\vec{R}_i = \vec{0}$.
We obtain:

$$\left\langle \left| \vec{r}_{n+i} - \vec{r}_i \right|^2 \left| \vec{r}_{m+j} - \vec{r}_j \right|^2 \right\rangle =$$

$$(a) \qquad \left(\alpha n + \varepsilon\right)\left(\alpha m + \varepsilon\right)$$

$$(b) \qquad \left(i+n-j\right)^2 \alpha^2 + \left(\alpha n + \varepsilon\right)\left(\alpha m + \varepsilon\right) + \frac{\varepsilon^2}{4}\delta_{j,i+n}$$

$$(c) \qquad n^2\alpha^2 + \left(\alpha n + \varepsilon\right)\left(\alpha m + \varepsilon\right) + \left(n\alpha + \frac{\varepsilon}{4}\right)\varepsilon\delta_{ij} \qquad (F.10)$$

$$(d) \qquad \left(i-j-m\right)^2 \alpha^2 + \left(\alpha n + \varepsilon\right)\left(\alpha m + \varepsilon\right) + \left(n\alpha + \frac{\varepsilon}{4}\right)\varepsilon\delta_{i+n,j+m}$$

$$(e) \qquad \left(\alpha n + \varepsilon\right)\left(\alpha m + \varepsilon\right) + \frac{\varepsilon^2}{4}\delta_{i,j+m}$$

The term $(\alpha n + \varepsilon)(\alpha m + \varepsilon)$ common to all cases will cancel out with the subtracted term $\left\langle \rho_n \right\rangle\left\langle \rho_m \right\rangle$ of Eq. (F.6). To split the double sum in Eq. (F.8) according to the cases listed

in (F.9), we need to consider two different cases: $m + n \leq N$ and $m + n > N$, illustrated on Fig. S11 and S12 respectively:

$m + n \leq N$:

$$\sum_{j=1}^{P}\sum_{i=1}^{K} = \underset{(a)}{\sum} + \underset{(b_1)}{\sum} + \underset{(b_2)}{\sum} + \underset{(c)}{\sum} + \underset{(d_1)}{\sum} + \underset{(d_2)}{\sum} + \underset{(e)}{\sum}$$

$$= \sum_{j=n+1}^{P}\sum_{i=1}^{j-n-1} + \sum_{j=1}^{n}\sum_{i=1}^{j} + \sum_{j=n+1}^{P}\sum_{i=j-n}^{j} + \sum_{j=1}^{P}\sum_{i=j+1}^{j+m-n} +$$

$$+ \sum_{j=1}^{K-m}\sum_{i=j+m-n+1}^{j+m} + \sum_{j=K-m+1}^{P}\sum_{i=1+m-n+1}^{K} + \sum_{j=1}^{K-m}\sum_{i=j+m+1}^{K} \qquad \text{(F.11)}$$

$m + n > N$:

$$\sum_{j=1}^{P}\sum_{i=1}^{K} = \underset{(b)}{\sum} + \underset{(c)}{\sum} + \underset{(d)}{\sum}$$

$$= \sum_{j=1}^{P}\sum_{i=1}^{j} + \sum_{j=1}^{P}\sum_{i=j+1}^{j+m-n} + \sum_{j=1}^{P}\sum_{i=j+m-n+1}^{K}$$

The calculations were performed using wxMaxima, a free symbolic calculation software[5]. The final result is ($m > n$):

$$\sigma_{nm}^2 = \frac{n\alpha^2}{6KP}\left\{ \begin{array}{c} -N\left(2n^2 - 6mn - 2\right) + \\ -n^3 + 2mn^2 - 6m^2n + n - 2m \end{array} \right\} + \frac{1}{K}\left(2n\alpha\varepsilon + \left(1 - \frac{n}{2P}\right)\frac{\varepsilon^2}{2} \right) \qquad m + n \leq N$$

$$= \frac{\alpha^2}{6K}\left\{ \begin{array}{c} N^2\left(N - 4n - 3m\right) + \\ N\left(6n^2 + 8mn + 3m^2 - 1\right) + \\ -6n^3 - 4m^2n + 4n - m^3 + m \end{array} \right\} + \frac{1}{K}\left(2n\alpha\varepsilon + \frac{\varepsilon^2}{2} \right) \qquad m + n > N \qquad \text{(F.12)}$$

with similar expressions when $m < n$ with n and m exchanged. Note that this result, which was verified with simulations (data not shown), differs from that of Appendix C in Qian *et al.*[7]. Eq. (F.12) can also be rewritten as ($m > n$):

$$\sigma_{nm}^2 = \frac{n}{6KP}\left\{ \begin{array}{c} 4n^2K + 2K - n^3 + n + \\ +(m-n)\left(6nP - 4n^2 - 2\right) \end{array} \right\}\alpha^2 + \frac{1}{K}\left(2n\alpha\varepsilon + \left(1 - \frac{n}{2P}\right)\frac{\varepsilon^2}{2} \right) \qquad m + n \leq N$$

$$= \frac{1}{6K}\left\{ \begin{array}{c} 6n^2K - 4nK^2 + K^3 + 4n - K + \\ +(m-n)\left(\begin{array}{c} (n+m)(2K+P) + \\ +2nP - 3K^2 + 1 \end{array} \right) \end{array} \right\}\alpha^2 + \frac{1}{K}\left(2n\alpha\varepsilon + \frac{\varepsilon^2}{2} \right) \qquad m + n > N \qquad \text{(F.13)}$$

showing that, except for the terms in ε^2 (case $m + n > N$), we recover the result of Eq. (D.15) when $n = m$.

Fig. S2 shows a map of the values σ^2_{nm} for $N = 1000$. Large covariances are obtained for large n and m and small $|m\text{-}n|$, as expected.

Error on the fitted parameters (weighted fit)
To simplify notations, we define the functions $g(i, j, N,x)$ and $h(i, j, N, x)$:

$$g(i, j, N, x) = \sigma^2_{ij} / \alpha^2$$
$$h(i, j, N, x) = g(i, j, N, x) f(i, N, x) f(j, N, x) \text{,}$$

(F.14)

and the intermediate quantities:

$$\Sigma_k = \sum_{i=1}^{p} i^k f(i, N, x)$$

$$\delta = \Sigma_0 \Sigma_2 - \Sigma_1^2$$

$$H_k(i, N, x) = \sum_{j=1}^{i-1} j^k h(i, j, N, x)$$

$$\Sigma_{0,h}(p) = \sum_{i=1}^{p} H_0(i, N, x)$$

(F.15)

$$\Sigma_{1,h}(p) = \sum_{i=1}^{p} \left[i.H_0(i, N, x) + H_1(i, N, x) \right]$$

$$\Sigma_{2,h}(p) = \sum_{i=1}^{p} i.H_1(i, N, x)$$

Equipped with these definitions and the expressions for σ^2_{nm}, we can now compute the variance of the intercept, σ^2_a, given by Eq. (F.3). From Eq. (F.2), we get:

$$\frac{\partial a}{\partial \rho_j} = \frac{1}{\Delta} \left(\frac{1}{\sigma^2_j} \sum_{k=1}^{p} \frac{t_k^2}{\sigma_k^2} - \frac{t_j}{\sigma_j^2} \sum_{k=1}^{p} \frac{t_k}{\sigma_k^2} \right)$$

(F.16)

Using (D.14) and $t_i = i.\Delta t$, we obtain after some algebra:

$$\frac{\sigma^2_a}{\varepsilon^2} = \frac{1}{x^2 \delta} \left\{ \Sigma_2 + \frac{2}{\delta} \left(\Sigma_2^2 \Sigma_{0,h}(p) - \Sigma_1 \Sigma_2 \Sigma_{1,h}(p) + \Sigma_1^2 \Sigma_{2,h}(p) \right) \right\}.$$

(F.17)

Similarly, for the slope b:

$$\frac{\partial b}{\partial \rho_j} = \frac{1}{\Delta} \left(\frac{t_j}{\sigma_j^2} \sum_{k=1}^{p} \frac{1}{\sigma_k^2} - \frac{1}{\sigma_j^2} \sum_{k=1}^{p} \frac{t_k}{\sigma_k^2} \right),$$

(F.18)

and the variance of the slope σ^2_b reads:

$$\frac{\sigma^2_b}{(4D)^2} = \frac{1}{\delta} \left\{ \Sigma_0 + \frac{2}{\delta} \left(\Sigma_1^2 \Sigma_{0,h}(p) - \Sigma_0 \Sigma_1 \Sigma_{1,h}(p) + \Sigma_0^2 \Sigma_{2,h}(p) \right) \right\}.$$

(F.19)

These expressions can be evaluated numerically for different values of the number of fitting points p. Fig. 4 shows examples of the dependence of σ_a/a and σ_b/b on p for a trajectory size of $N = 1000$ and different values of x, while Fig. S13 shows other examples for $N = 100$ and $N = 10$.
The common feature of all these graphs is the existence of a minimum error at a number of fitting points p which depends on $x = \varepsilon/\alpha$. The larger the relative magnitude of

localization uncertainty ε with respect to the elementary square displacement α (i.e. the larger x), the larger the number of MSD points needed to achieve the best parameter accuracy. For $x = 0$ (i.e. no localization uncertainty), the best accuracy for the diffusion coefficient D is always achieved with the first two points of the MSD curve ($p = 2$).

Note that for small numbers of trajectory points ($N = 10$, Fig. S13) and large enough x, the minimum is attained when all points of the MSD curve are used for the fit ($p = N - 1$).

Error on the fitted parameters (unweighted fit)

Obtaining an experimental estimate of the σ_i^2 (Eq. (D.13)) is not trivial. For this reason, the least-square fit to Eq. (F.1) is oftentimes performed without weighting the MSD data points, i.e. setting $\sigma_i^2 = 1$ in Eq. (F.2). In this case, the fit parameters read[9]:

$$a' = \frac{1}{\Delta'}\left(\sum_{i=1}^{p} t_i^2 \sum_{i=1}^{p} \rho_i - \sum_{i=1}^{p} t_i \sum_{i=1}^{p} t_i \rho_i \right)$$

$$b' = \frac{1}{\Delta'}\left(p\sum_{i=1}^{p} t_i \rho_i - \sum_{i=1}^{p} t_i \sum_{i=1}^{p} \rho_i \right) \tag{F.20}$$

$$\Delta' = p\sum_{i=1}^{p} t_i^2 - \left(\sum_{i=1}^{p} t_i \right)^2$$

This has two practical consequences: (i) the fitted parameters will be different from that of a weighted fit (although the theoretical expectations remain the same, i.e. a and b of Eq. (F.1)) and (ii) the standard error on the fitted parameters will also be different.

The standard error on the parameters can still be estimated using the same Eq. (F.3) & (F.4) (in which we need to keep the variance and covariance of the MSD) but replacing (F.16) & (F.18) by the expressions derived from Eq.(F.20):

$$\frac{\partial a'}{\partial \rho_j} = \frac{1}{\Delta'}\left(\sum_{i=1}^{p} t_i^2 - t_j \sum_{i=1}^{p} t_i \right)$$

$$\frac{\partial b'}{\partial \rho_j} = \frac{1}{\Delta'}\left(pt_j - \sum_{i=1}^{p} t_i \right) \tag{F.21}$$

Using $t_i = i.\Delta t$, we obtain:

$$\Delta' = p\sum_{i=1}^{p} t_i^2 - \left(\sum_{i=1}^{p} t_i \right)^2 = \frac{p^2\left(p^2 - 1\right)}{12}(\Delta t)^2$$

$$\frac{\partial a'}{\partial \rho_j} = \frac{6}{p(p-1)}\left(\frac{2p+1}{3} - j \right) \tag{F.22}$$

$$\frac{\partial b'}{\partial \rho_j} = \frac{12}{\Delta t}\frac{1}{p\left(p^2 - 1\right)}\left(j - \frac{p+1}{2} \right)$$

The results for the error on the unweighted fit parameters σ_a^2 and σ_b^2 follow:

$$\frac{\sigma_{a'}^2}{\varepsilon^2} = \frac{1}{x^2}\left(\frac{6}{p(p-1)}\right)^2 \left\{ \begin{array}{l} \sum_{i=1}^{p}\left(\frac{2p+1}{3}-i\right)^2 f(i,N,x)^{-1} + \\ \\ +2\sum_{i=1}^{p}\sum_{j<i}\left(\frac{2p+1}{3}-i\right)\left(\frac{2p+1}{3}-j\right)g(i,j,N,x) \end{array} \right\}$$

$$\frac{\sigma_{b'}^2}{(4D)^2} = \left(\frac{12}{p(p^2-1)}\right)^2 \left\{ \begin{array}{l} \sum_{i=1}^{p}\left(\frac{p+1}{2}-i\right)^2 f(i,N,x)^{-1} + \\ \\ +2\sum_{i=1}^{p}\sum_{j<i}\left(\frac{p+1}{2}-i\right)\left(\frac{p+1}{2}-j\right)g(i,j,N,x) \end{array} \right\}$$

(F.23)

which can easily be computed numerically.

Fig. 6 shows examples of the dependence of σ_a/a and σ_b/b on p for a trajectory size of $N = 1000$ and different values of x, while Fig. S14 shows other examples for $N = 100$ and $N = 10$. As for the weighted fit result, there is a minimum error for a number p of fitting points depending on $x = \varepsilon/\alpha$.

Comparison of the two fitting methods
Since one of the two methods (the unweighted fit) requires less a priori knowledge, it is interesting to compare the expected error on the fitted parameters in both cases. Fig. 7 regroups the results of Fig. 4C-D & 6C-D. It shows that even though the errors are always smaller in the case of a weighted fit, the absolute minima are comparable. In other words, if we know the optimal number of fitting points, there is no real advantage in using the weighted fit method, which requires the additional knowledge of the SDV of the MSD curve.

Appendix G: C-code to compute the main formulas derived in this work

```c
/*-------------------------------------------------------------------*
 * File Name:      MSDFormula.c                                      *
 * Creation:       XM                                                *
 * Purpose:        Supporting Information to:                        *
 * "Mean Square Displacement Analysis of Single-Particle Trajectories *
 *  with Localization Error: Brownian Motion in Isotropic Medium"    *
 *                                                                   *
 * Copyright (c) X. Michalet, 2010                                   *
 *                                                                   *
 *                                                                   *
 * Created: 08/28/2010                                               *
 *-------------------------------------------------------------------*/

/* Relative errors on fit performed with weights */

double NormMSDInterceptError(double x, double P, double N)
{
//      square root of Eq. F18 (sigma_a/a)
//
//      x      : epsilon/alpha= 4.sigma^2/4.D.delta t
//      N      : number of points in the trajectory
//      P      : <N, number of points used for the fit

        int i,j;
        double fi, fj, gij, hij, sum0 = 0, sum1 = 0, sum2 = 0;
        double h0, h1, sumh0 = 0, sumh1 = 0, sumh2 = 0;
        double delta, temp, norm_sigmaa;

        if (P < 2 || P >= N)
              return 0;
        for (i=1; i<= P; i++)
        {
              fi = f(i, N, x);

              sum0 += fi;
              sum1 += i*fi;
              sum2 += i^2*fi;
              h0 = 0; h1 = 0;
              for (j=1; j<i; j++)
              {
                    fj = f(j,N,x);
                    gij = g(i,j,N,x);
                    hij = fi*fj*gij;
                    h0 += hij;
                    h1 += j*hij;
              }
              sumh0 += h0;
              sumh1 += i*h0 + h1;
              sumh2 += i*h1;
        }
        delta = sum0*sum2 - sum1^2;
        temp = (sum2 + 2*(sum2^2*sumh0 - sum1*sum2*sumh1 +
               sum1^2*sumh2)/delta)/delta;
        norm_sigmaa = sqrt(temp)/x;
        return norm_sigmaa;
}
double NormMSDSlopeError(double x, double P, double N)
{
//      square root of Eq. F20 (sigma_b/b)
//
//      x      : epsilon/alpha= 4.sigma^2/4.D.delta t
//      N      : number of points in the trajectory
```

```
//     P      : <N, number of points used for the fit

       int i,j;
       double fi, fj, gij, hij, sum0 = 0, sum1 = 0, sum2 = 0;
       double h0, h1, sumh0 = 0, sumh1 = 0, sumh2 = 0;
       double delta, temp, norm_sigmab;

       if (P < 2 || P >= N)
             return 0;

       for (i=1; i<= P; i++)
       {
             fi = f(i, N, x);

             sum0 += fi;
             sum1 += i*fi;
             sum2 += i^2*fi;
             h0 = 0; h1 = 0;
             for (j=1; j<i; j++)
             {
                    fj = f(j,N,x);
                    gij = g(i,j,N,x);
                    hij = fi*fj*gij;
                    h0 += hij;
                    h1 += j*hij;
             }
             sumh0 += h0;
             sumh1 += i*h0 + h1;
             sumh2 += i*h1;
       }
       delta = sum0*sum2 - sum1^2;
       temp = (sum0 + 2*(sum1^2*sumh0 - sum0*sum1*sumh1 +
              sum0^2*sumh2)/delta)/delta;
       norm_sigmab = sqrt(temp);
       return norm_sigmab;
}

/* Relative errors on fit performed without weights */

double NormMSDInterceptErrorNW(double x, double P, double N)
{
//     square root of Eq. F24 (sigma_a/a)
//
//     x      : epsilon/alpha= 4.sigma^2/4.D.delta t
//     N      : number of points in the trajectory
//     P      : <N, number of points used for the fit

       int i,j;
       double fi, gij;
       double h0, h1, sum0 = 0, sum1 = 0, psum_i;
       double temp0, temp, norm_sigmaa;

       if (P < 2 || P >= N)
             return 0;
       for (i=1; i<= P; i++)
       {
             fi = f(i, N, x);
             sum0 += ((2*P+1.)/3. - i)^2/fi;
             psum_i = 0;
             for (j=1; j<i; j++)
             {
                    gij = g(i,j,N,x);
                    psum_i += ((2*P+1.)/3. - j)*gij;
             }
```

```
                sum1 += psum_i*((2*P+1.)/3. - i);
        }
        temp = (sum0 + 2*sum1);
        norm_sigmaa = sqrt(temp)*6/P/(P-1)/x;
        return norm_sigmaa;
}

double NormMSDSlopeErrorNW(double x, double P, double N)
{
//      square root of Eq. F24 (sigma_b/b)
//
//      x     : epsilon/alpha= 4.sigma^2/4.D.delta t
//      N     : number of points in the trajectory
//      P     : <N, number of points used for the fit

        int i,j;
        double fi, gij;
        double h0, h1, sum0 = 0, sum1 = 0, psum_i;
        double temp, norm_sigmab;

        if (P < 2 || P >= N)
                return 0;
        for (i=1; i<= P; i++)
        {
                fi = f(i, N, x);
                sum0 += ((P+1.)/2. - i)^2/fi;
                psum_i = 0;
                for (j=1; j<i; j++)
                {
                        gij = g(i,j,N,x);
                        psum_i += ((P+1.)/2. - j)*gij;
                }
                sum1 += psum_i*((P+1.)/2. - i);
        }
        temp = (sum0 + 2*sum1);
        norm_sigmab = sqrt(temp)*12/P/(P-1.)/(P+1.);
        return norm_sigmab;
}

double MSDPDF(double z, double n, double N, double x)
{
//      Eq. E5 with the definition Eq. E10 of K
//
//      x     : epsilon/alpha= 4.sigma^2/4.D.delta t
//      N     : number of points in the trajectory
//      n     : MSD point
//      z     : normalized MSD (rho_n/alpha)
//      gammaln: natural logarithm of the gamma function
//              gamma(N) = (N-1)!

        double K, pdf;

        K = (n+x)^2*f(n,N,x);
        pdf = ln(K/(n+x)) + (K-1)*ln(K*z/(n+x)) - K*z/(n+x) - gammaln(K);
        return exp(pdf);
}
double gammadist(double lambda, double k, double x)
{
//      gamma distribution
//      gammaln: natural logarithm of the gamma function
//              gamma(N) = (N-1)!

        double pdf;
```

```
        pdf = ln(lambda)+(k-1)*ln(lambda*x)-lambda*x - gammaln(k);
        return exp(pdf);
}
double f(double n, double N, double x)
{
//      Eq. D15

        double K, f;

        K = N - n;
        if (n <= K)
                f = fminus(n,N,x);
        else
                f = fplus(n,N,x);
        return f;
}
double fminus(double n, double N, double x)
{
        double K, f;

        K = N - n;
        f = n*(4*n^2*K + 2*K - n^3 + n)/6/K^2 + (2*n*x + (1+(1 -
                n/K)/2)*x^2)/K;
        return 1/f;
}
double fplus(double n, double N, double x)
{
        double K, f;

        K = N - n;
        f = (6*n^2*K - 4*n*K^2 + 4*n + K^3 - K)/6/K + (2*n*x + x^2)/K;
        return 1/f;
}
double g(double m, double n, double N, double x)
{
//      Eq. F13

        double g;
        double tmp;

        if (m <= n)
        {
                m = tmp;
                m = n;
                n = tmp;
        }
        if (m+n > N)
                g = (N^2*(N - 4*n - 3*m) + N*(6*n^2 + 8*m*n + 3*m^2 - 1) -
                        6*n^3 - 4*m^2*n + 4*n - m^3 + m)/6/K + (2*n*x + x^2/2)/K;
        else
                g = n*(-N*(2*n^2 - 6*m*n - 2) - n^3 + 2*m*n^2 - 6*m^2*n + n
                        - 2*m)/6/K/P + (2*n*x + (1 - n/2/P)*x^2/2)/K;
        return g;
}
```

References

[1]   F. Pinaud, X. Michalet, G. Iyer, et al., Traffic **10**, 691 (2009).

[2]   J. Schuster, F. Cichos, and C. von Borczyskowski, Journal of Physical Chemistry A **106**, 5403 (2002).

[3]   M. Goulian and S. M. Simon, Biophysical Journal **79**, 2188 (2000).

[4]   T. Savin and P. S. Doyle, Biophysical Journal **88**, 623 (2005).

[5]   Maxima.sourceforge.net, (Souceforge.net, 2009).

[6]   D. Montiel, H. Cang, and H. Yang, Journal of Physical Chemistry B **110**, 19763 (2006).

[7]   H. Qian, M. P. Sheetz, and E. L. Elson, Biophysical Journal **60**, 910 (1991).

[8]   T. Savin and P. S. Doyle, Physical Review E **76**, 021501 (2007).

[9]   P. R. Bevington and E. K. Robinson, *Data Reduction and Error Analysis for the Physical Sciences* (McGraw-Hill, 1992).

Supporting Information Figure Captions

Fig. S1: Comparison between the MSD SDV of simulated data and theory. For each set of parameters (σ, D, Δt), $N_S = 1000$ simulations of $N = 1000$ steps were performed. The MSD and MSD SDV were calculated for each time lag (Eq.(D.17). The corresponding MSD (black) and MSD $\pm$ SDV (dark gray) curves are represented together with the theoretical result (red, Eq. (D.13)) using the corresponding x value. The box above each graph indicates the parameters used for the simulations. MSD units: μm^2/s.

Fig. S2: Map of the covariance terms σ_{nm}^2 for $x = 0$, $N = 1000$ (Eq. (F.13)). The maximum covariance is obtained for large n and m and is symmetric around the axis $n = m$. This behavior is not changed for $x > 0$.

Fig. S3: Relative covariance of the MSD (Eq. (29)). A: For $x = 0$, the relative covariance is approximately equal to n/N (dashed curve), where n is the time lag and N the number of points in the trajectory. Different sections of the relative covariance map ($m = kn$) are shown. B: For $x > 0$, the relative covariance decreases as x increases. Note however, that, as shown in Fig. S2, the covariance itself is unchanged and remains large.

Fig. S4: Schematic explaining the definition of sub-trajectories used in instantaneous diffusion coefficient analysis. A. The case of sub-trajectories of length $N_S = 3$ is illustrated, with two complete sub-trajectories (T_5 and T_8) and one incomplete "end" sub-trajectory (T_1) shown. B. For MSD fit for each sub-trajectory, a series of diffusion coefficient values $D(t)$ is obtained.

Fig. S5: Decomposition of the observed displacement $\vec{d} = \vec{r}_{n+i} - \vec{r}_i$ as a sum of the real displacement $\tilde{\vec{d}} = \tilde{\vec{r}}_{n+i} - \tilde{\vec{r}}_i$ and random localization error in both directions, χ_i, χ_{n+I} and η_i, η_{n+i}.

Fig. S6: A. Two different cases occurring when computing the correlation between two displacements: (a) the two displacements do not have any intermediate vertex in common (they do not "overlap"); (b) the two displacements overlap. By convention, $\vec{R}_1$ represents the displacement between the vertices with the two lowest indices, etc. Note that for this reason, the definition of the $\vec{R}_k$'s are different in case (a) and (b). B. Decomposition of the double sum over i and j in Eq. (D.11) in parts having identical summand (a or b_k) and leading to simple algebraic expressions (b_1 and b_2). This figure is directly inspired from handwritten notes from Prof. Qian.

Fig. S7: Number of effective independent square displacements as a function of time lag n for a trajectory containing $N = 1000$ points (Eq. (E.10)). The different curves correspond to different reduced localization uncertainty parameter x. The dashed curve corresponds to $K = N - n$, the number of displacements corresponding to a time lag of n.

Fig. S8: Reduced mean square displacement distributions for representative values of the time lag n, in the absence of localization uncertainty ($x = 0$). Number of trajectory points, $N = 1000$. Histogram: result of $N_S = 1000$ simulations. Black curve: Gamma distribution with parameters λ, K given by Eq. (E.6) and (E.10).

Fig. S9: Reduced mean square displacement distributions for representative values of the time lag n, in the presence localization uncertainty ($x = 10$). Number of trajectory points, $N = 1000$. Histogram: result of $N_S = 1000$ simulations. Black curve: Gamma distribution with parameters λ, K given by Eq. (E.6) and (E.10).

Fig. S10: Different possibilities for the order of indices i, $i+n$, j and $j+m$ in Eq. (F.9). The vectors $\vec{R}_k$ are defined as represented on the upper part of the figure.

Fig. S11: Decomposition of the double sum over j and i in Eq. (F.8) in parts having identical summand when $m+n \leq N$ (Eq. (F.11)).

Fig. S12: Decomposition of the double sum over j and i in Eq. (F.8) in parts having identical summand when $m+n > N$ (Eq. (F.11)).

Fig. S13: Relative error on fitted parameters (weighted fit, $N = 100$ & $N = 10$ points), Eq. (F.17) and (F.19). Evolution of the relative errors on fitted parameters (A, C: intercept a; B, D: slope b) as a function of the number of MSD points used for the fit. The curves correspond to different values of reduced localization error x.

Fig. S14: Relative error on fitted parameters (unweighted fit, $N = 100$ & $N = 10$ points), Eq. (F.23). Evolution of the relative errors on fitted parameters (A, C: intercept a; B, D: slope b) as a function of the number of MSD points used for the fit. The curves correspond to different values of reduced localization error x.
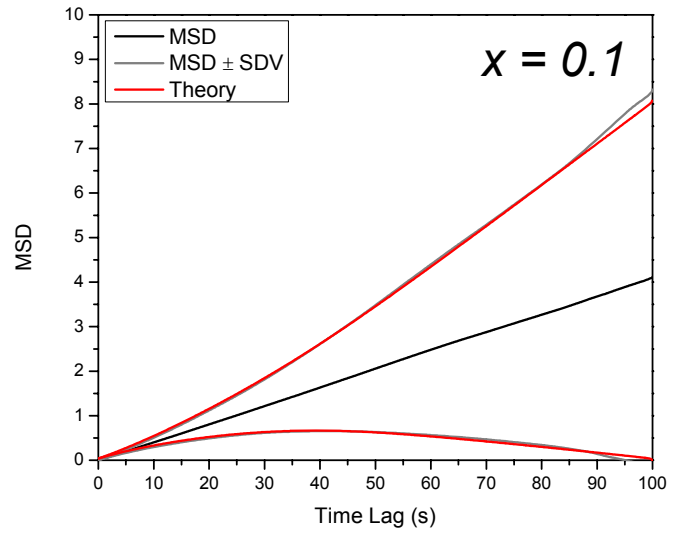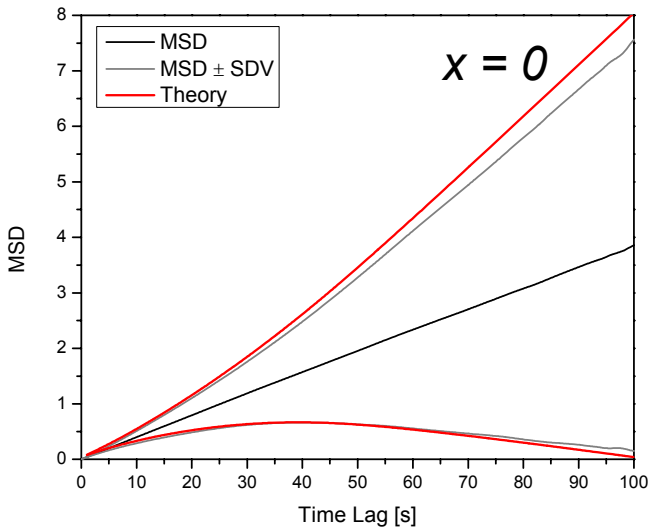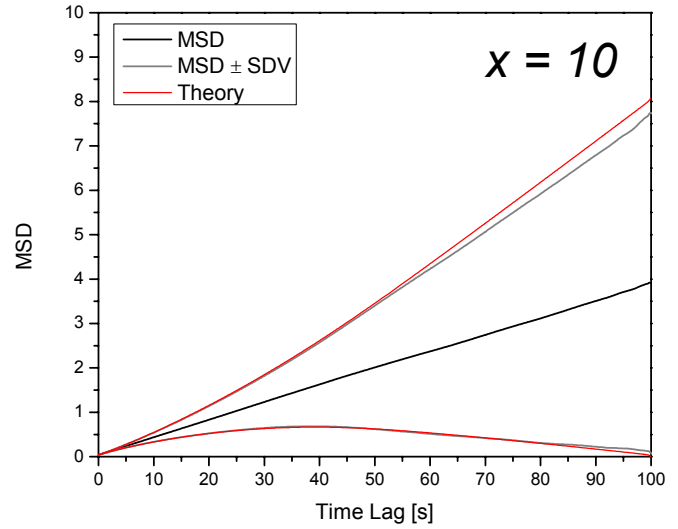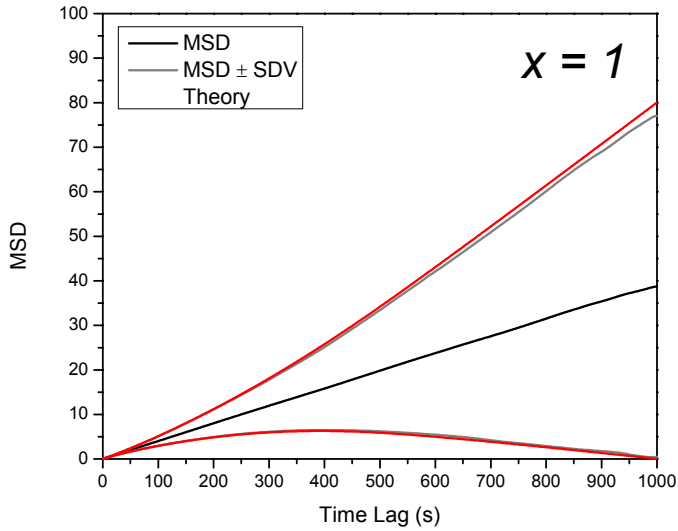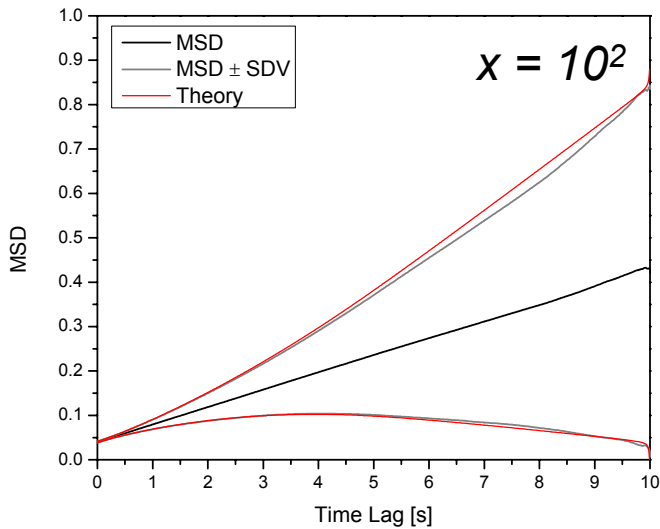
Fig. S1

Fig. S2

Fig. S3

Fig. S4

Fig. S5

A

(a)

R_1 R_2 R_3

| | | | |
| j | j+n | i | i+n |

(b)

R_1 R_2 R_3

| | | | |
| j | i | j+n | i+n |

B

Fig. S6

N = 1000

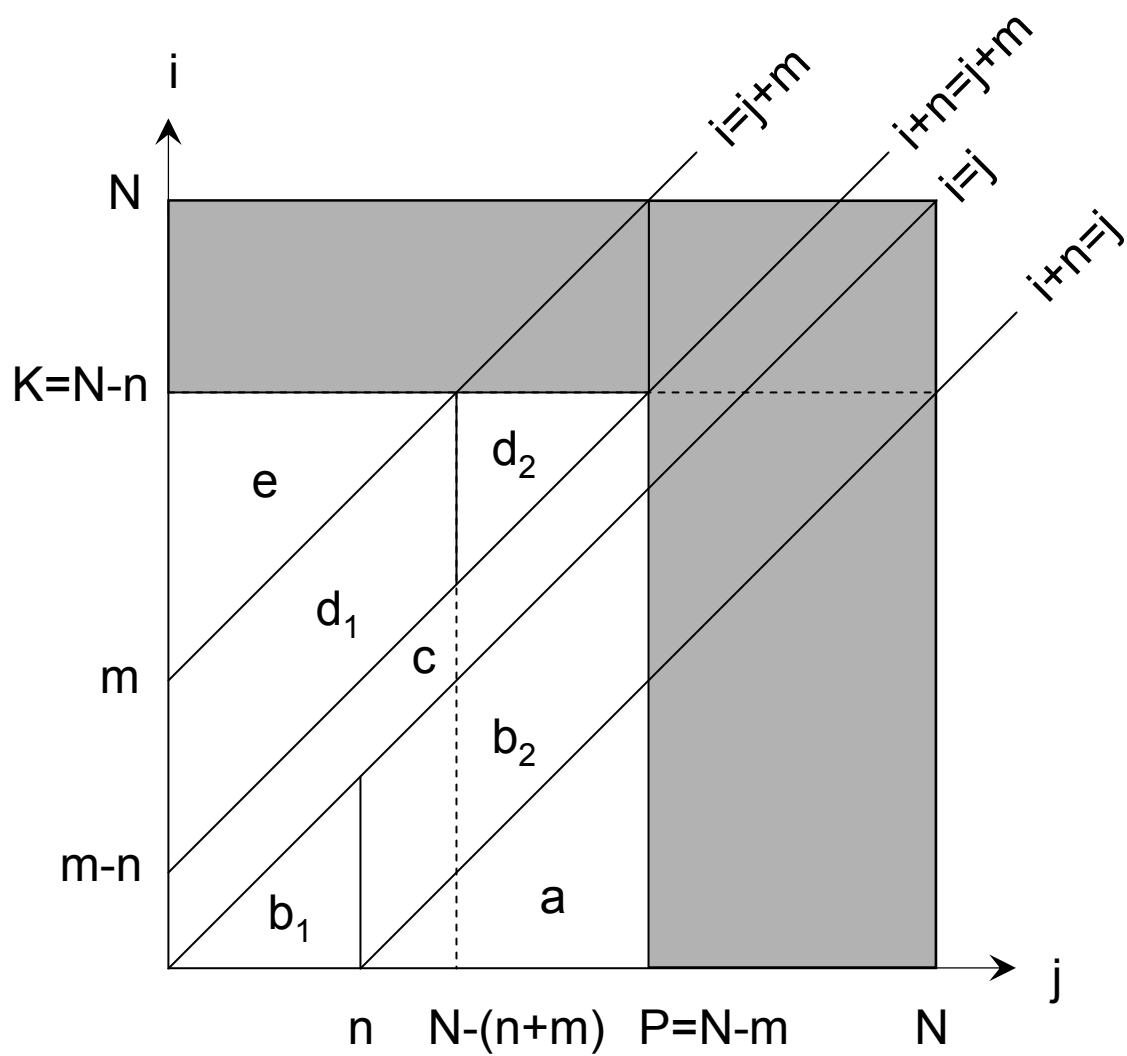K(n,N,x)

- - - uncorr
—— x = 0
—— x = 1
—— x = 10
—— x = 100
—— x = 1000

n
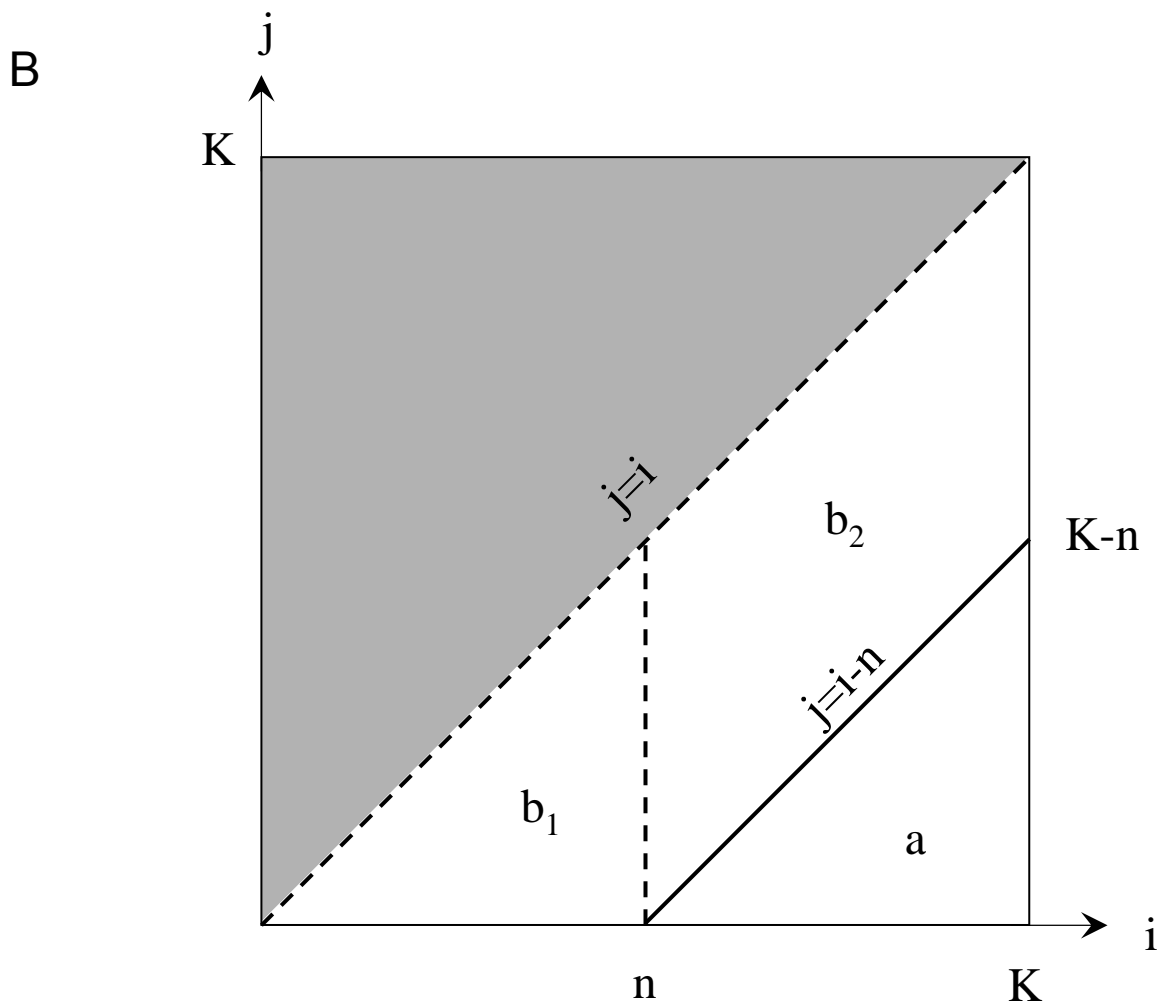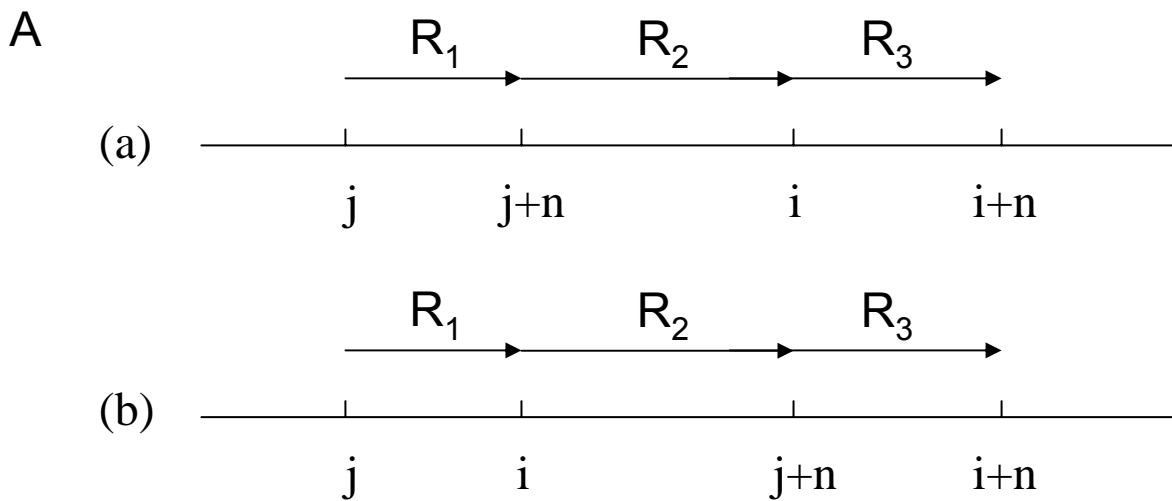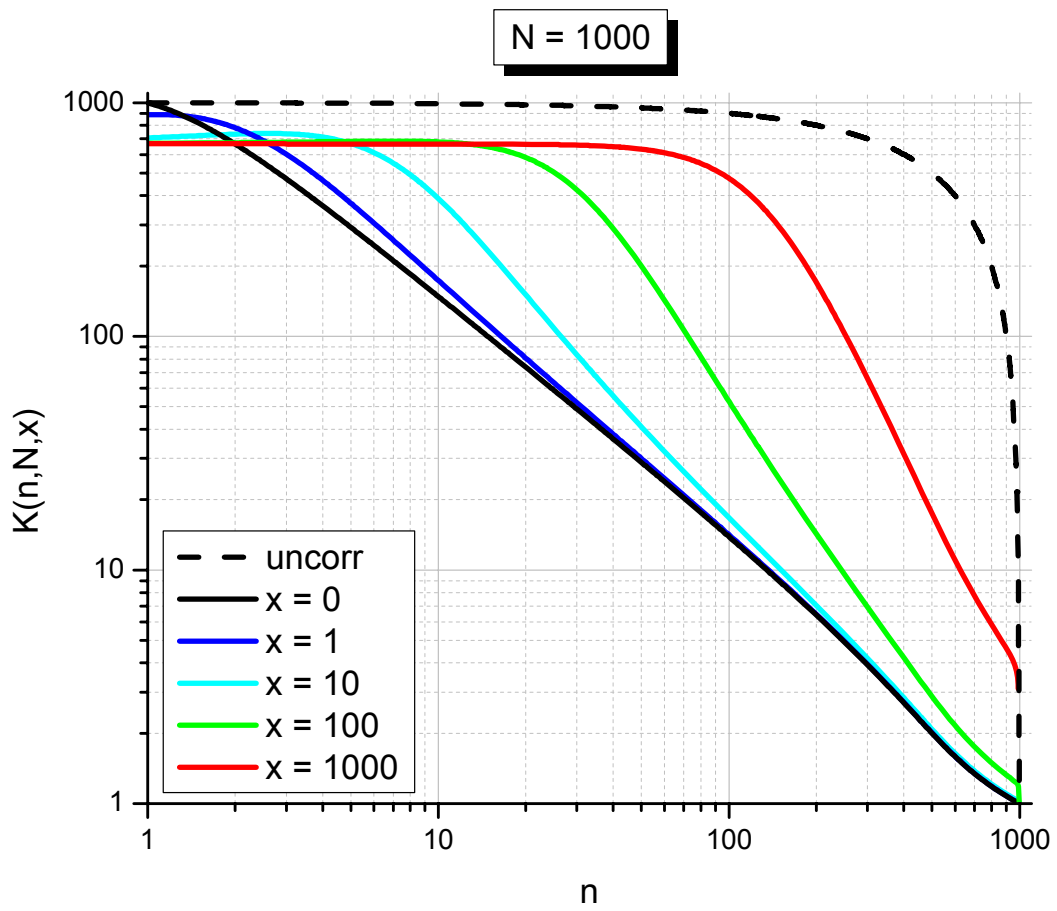
Fig. S7
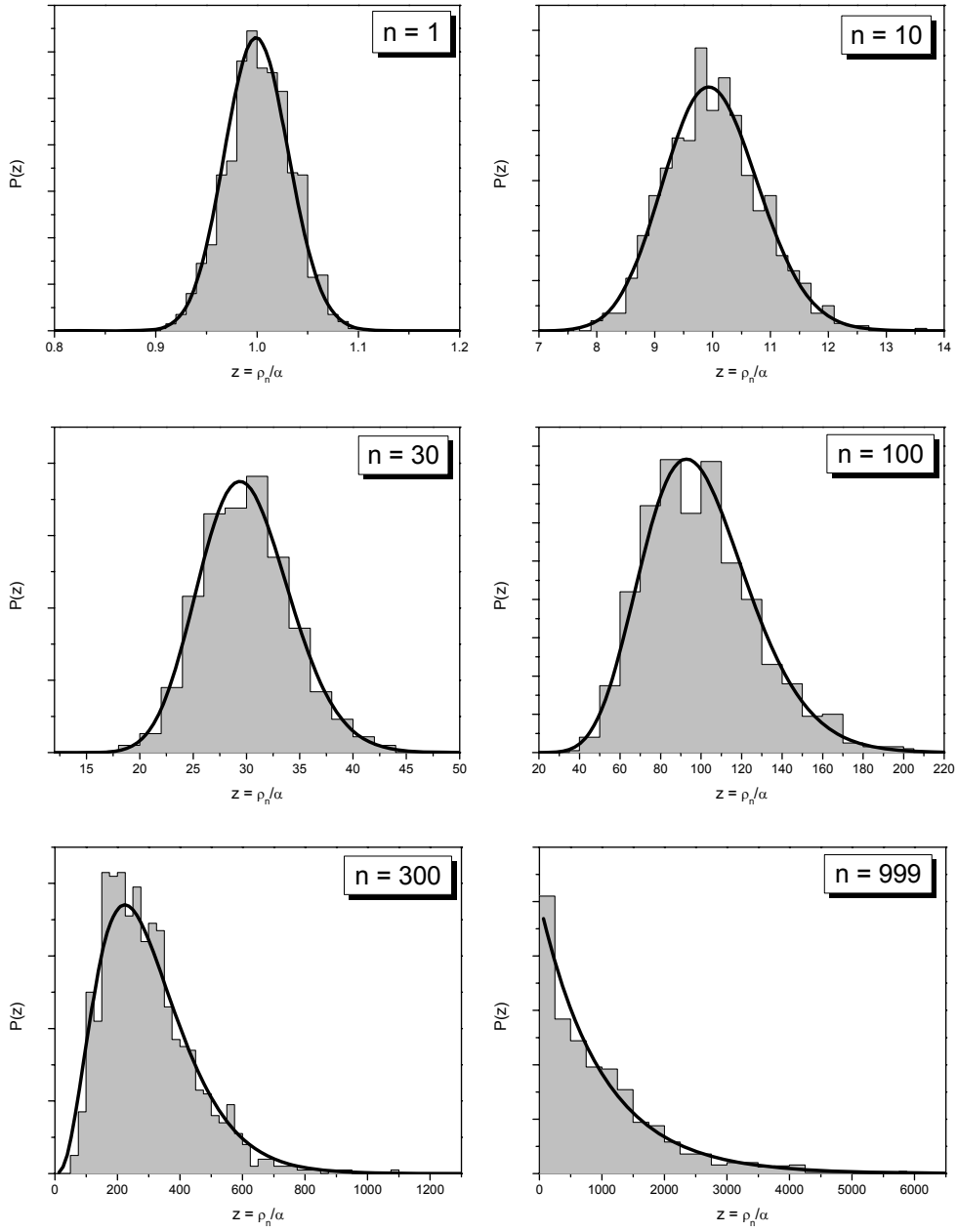
Fig. S8
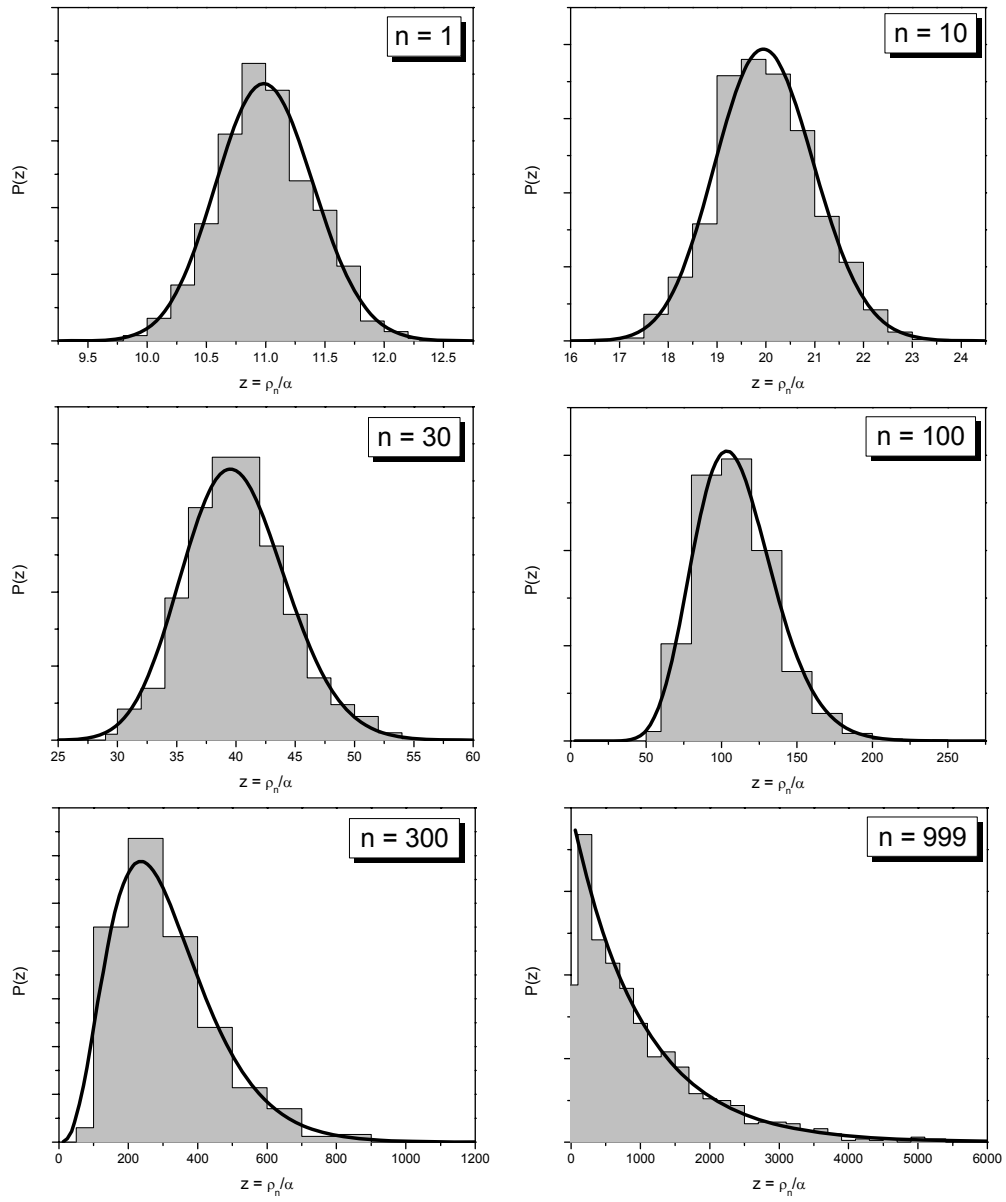
Fig. S9
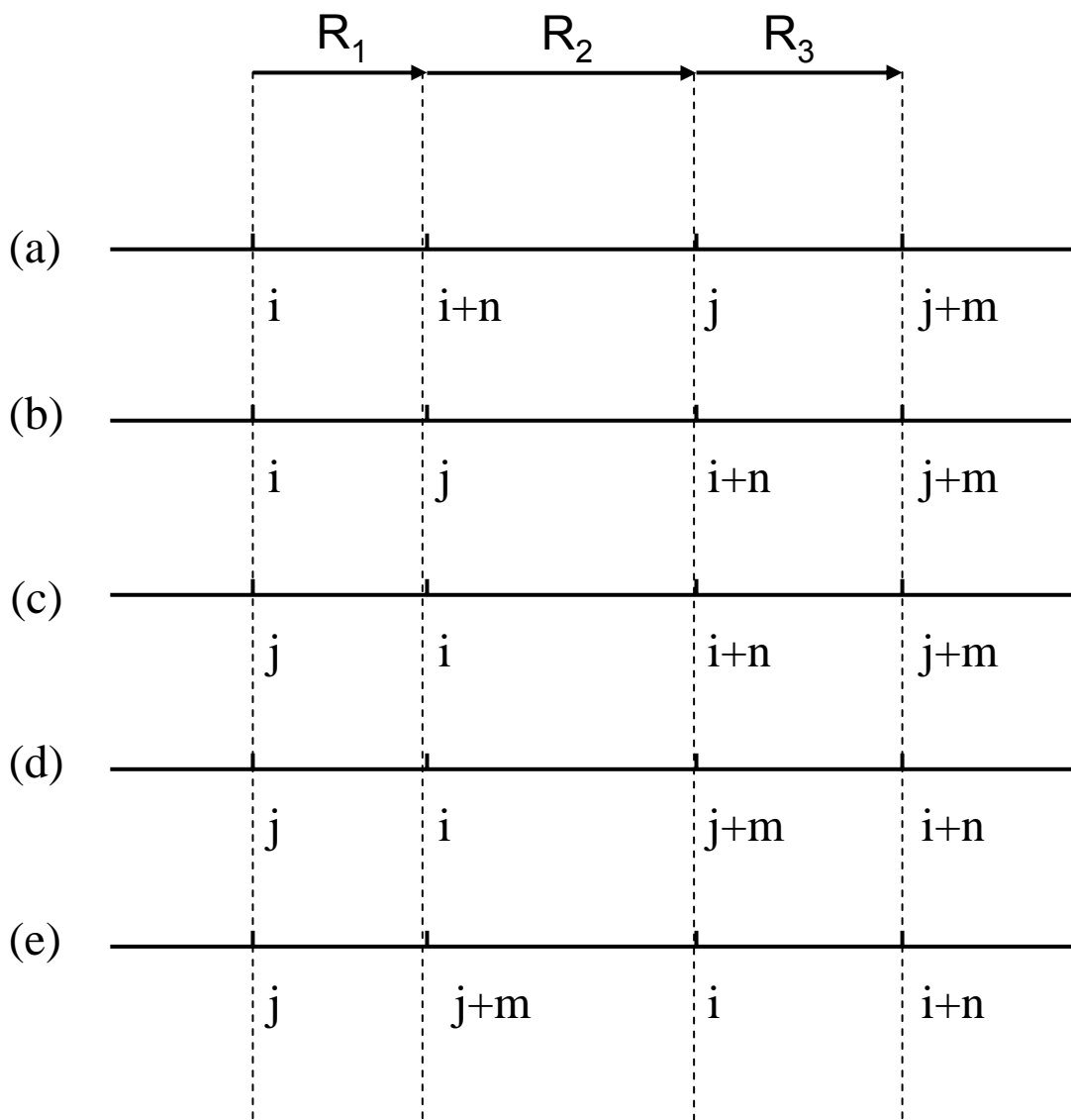
Fig. S10

Fig. S11

m+n > N

Fig. S12

Fig. S13

Fig. S14