Real-time calibration of a feedback trap

Momčilo Gavrilov, Yonggun Jun,[a)] and John Bechhoefer[b)]

Department of Physics, Simon Fraser University, Burnaby, British Columbia V5A 1S6, Canada

Feedback traps use closed-loop control to trap or manipulate small particles and molecules in solution. They have been applied to the measurement of physical and chemical properties of particles and to explore fundamental questions in the non-equilibrium statistical mechanics of small systems. These applications have been hampered by drifts in the electric forces used to manipulate the particles. Although the drifts are small for measurements on the order of seconds, they dominate on time scales of minutes or slower. Here, we show that a recursive maximum likelihood (RML) algorithm can allow real-time measurement and control of electric and stochastic forces over time scales of hours. Simulations show that the RML algorithm recovers known parameters accurately. Experimental estimates of diffusion coefficients are also consistent with expected physical properties. © 2014 AIP Publishing LLC. [http://dx.doi.org/10.1063/1.4894383]

I. INTRODUCTION

The feedback trap is a new experimental technique developed in 2005 by Cohen and Moerner, who termed it the ABEL (Anti-Brownian ELectrokinetic) trap.[1] The technique allows trapping of small particles and molecules in solution by creating a feedback loop[2] where one images the object to be trapped and then applies an electric force to the charged object that moves it in the desired direction. Because direct electric forces are stronger than the dipolar forces used in optical and magnetic tweezers, feedback traps can trap molecules and submicrometer particles that are otherwise impossible to confine. It is even possible to trap a single fluorescent dye molecule in water.[3,4]

The feedback trap has had two major types of application. First, it has been used to probe the physical or chemical properties of the trapped object. These include the diffusion constant and electric-field mobility of single particles,[5] their fluctuations,[6] and elastic and dissipation parameters that characterize the internal degrees of freedom of more complex objects.[7,8] Feedback traps can also be used to estimate the chemical properties of single molecules, including photodynamic and enzymatic properties of biomolecules[9] and the interplay between fluorescence spectroscopy and conformation at the single-molecule level.[10,11]

The second kind of application has been to the study fundamental questions in statistical mechanics. The key feature of such traps is the ability to impose arbitrary, time-dependent virtual potentials on particles. For example, Cohen studied the motion of a particle in a virtual double-well potential and in radial potentials of the form $U(r) \sim r^n$.[12]

In preliminary work,[13] we used time-dependent virtual potentials to study Landauer's principle,[14,15] which relates information erasure to thermodynamic work. Our initial attempts were frustrated by small but persistent drifts that, over

hours or days, led to significant systematic errors in work measurements. Extensive investigation showed that the drifts were mostly caused by potential offsets that are linked to chemical reactions at the electrodes used to impose the electric field on the particle. A secondary cause was temperature-dependent offsets of the voltage amplifier used to impose potential differences across the electrodes. We conclude that such drifts have been present, not only in our work, but in previous experiments, as well, and have an importance that grows with the duration of measurements. Improving temperature and potential control can reduce somewhat the magnitude of the drifts but cannot make them small enough to be neglected in long experiments. A recent discussion of feedback-trap calibration has successfully demonstrated accurate parameter measurements over short time scales (up to 1 min).[4] The focus here is on time scales of hours, or even days. Such long times are important in tests of stochastic thermodynamics, which depend on high-precision statistics for particle trajectories.

Here, rather than try to eliminate drifts, we continuously measure and correct for them in real time as the experiment runs. We will see that such techniques can successfully remove the effects of drifts from experimental data that are collected over days. We also correct for biases that arise because of subtle correlations in the noise that enter because of the structure of the feedback loop. In related work, we have used the techniques described here to make the highest precision measurements of Landauer's principle achieved to date.[16]

Below, we describe in detail the methods we used to carry out the real-time calibration of the feedback trap. We begin, in Sec. II A by briefly recalling some of the details of our trap, in particular as they relate to the timing of data acquisition. In Sec. III, we review the equations of motion that describe rigid particles in a feedback trap and show how to cast the equations in a more convenient and general form, which is required to implement the calibration. In Sec. IV, we discuss the principles of ordinary recursive least-squares (RLS) fit and recursive maximum likelihood (RML) algorithm. The

---

a)Present address: Department of Developmental and Cell Biology, University of California, Irvine, California 92697-2300, USA.
b)Email: johnb@sfu.ca

basic idea is to recast least-squares fits to allow old fit parameters to be updated each time a new data point is taken, rather than redoing the whole fit. The resulting speedup is important for real-time operation. In Sec. V, we generalize the previous discussion from one to two spatial dimensions. The former is easier to follow, but the latter is what we actually use. In Appendix B, we describe our control software. Because the methods we use are real time, they must be integrated with the rest of the experiment, leading to a rather complex control program, whose details are crucial in achieving a successful calibration. In Sec. VI, we show that our analysis methods work on simulated data. That is, we show that if we simulate data with known parameters, then our analysis routines recover the known parameters. We will see that the task is complicated by correlations in the noise term that bias the inference if not accounted for. Finally, in Sec. VII, we present experimental results. We see convergence similar to that observed in simulations and, in the case of the diffusion coefficient measurement, argue that the absolute values are in the expected range, as well. We conclude that the advances described here show sufficient mastery of the experiment to obtain reliable thermodynamic measurements.

II. FEEDBACK TRAP

The operation of a feedback trap is illustrated in Fig. 1. The trap first acquires an image of an object and uses image-analysis software to estimate its position [Figs. 1(a) and 1(b)]. Based on the observed position ($\bar{x}_n$) and the chosen virtual potential, the program calculates the force to be applied as the negative gradient of the imposed potential [Fig. 1(c)]. This force is applied as an electrical force by applying voltage over the set of two horizontal electrodes given in Fig. 1(d). At the end of the cycle, a particle has been displaced relative to its previous position because of feedback and thermal (diffusion) forces. We use a cycle time of 10 ms.
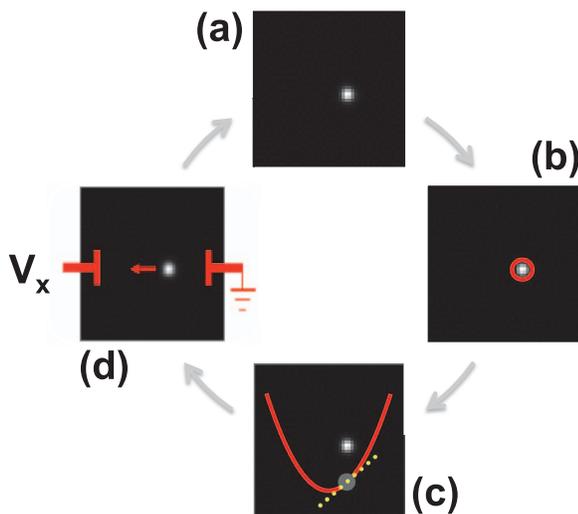
The scheme in Fig. 1 is easily generalized to the two-dimensional case by calculating the position along the other axis and inserting an additional set of "vertical" electrodes in Fig. 1(d). We thus apply forces along two directions independently.

A. Experimental setup

Our version of the feedback trap, described more completely in Ref. 17, uses 210-nm polystyrene beads in water, confined in the vertical dimension in an 800-nm-thick cell and controlled in the two lateral dimensions by two sets of electrodes. Particle images are recorded using an EM-CCD camera mounted to a home-built epifluorescence microscope. The digital image is loaded into a LabVIEW program that uses a centroid algorithm to determine the particle position, calculate the required forces, and output voltages to two data acquisition devices (DAQs). The analog output of the first DAQ controls the intensity of the excitation laser, regulating it so that the detected fluorescence intensity stays constant, even as the particle bleaches, thus keeping the measurement noise constant.

The analog outputs of the other DAQ are used to apply forces on the particle. They are sent to a home-built voltage amplifier with a gain of 15 and then applied to two pairs of electrodes. To limit current flows through the cell due to capacitative charging, we place a 10 kΩ resistor in parallel with each pair of electrodes and 1 kΩ resistor in series. Since the electrical resistance of the flow cell is ≈10 MΩ, the voltage drop due to the series resistor is negligible.

The update time of the feedback trap is set to 10 ms, while the camera exposure time is 5 ms. The delay between the midpoint of the camera exposure and the application of a feedback voltage response is set to 10 ms. The timing diagram for the data acquisition is shown in Fig. 2.

III. PARTICLE DYNAMICS IN A FEEDBACK TRAP

Previously,[18] we derived the equations of motion for a particle in a one-dimensional virtual harmonic potential of the form $U_{\text{harm}}(x) = \frac{1}{2}\kappa x^2$, where κ is the force constant and x is the displacement from equilibrium. Here, we generalize to the case of motion in an arbitrary, time-dependent virtual potential $U(x, t)$ and then derive an alternate form of the equations
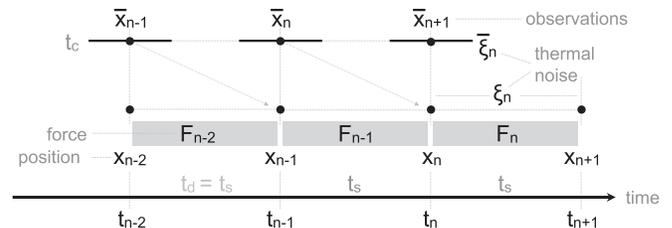

FIG. 1. One time step (one cycle) of a feedback trap: (a) Acquire an image; (b) estimate position ($\bar{x}_n$) from the image; (c) calculate feedback force as a gradient of potential $F_n = -\partial_x U(\bar{x}_n, t)$; (d) apply feedback force as an electric force via set of electrodes. At the end of the cycle, deterministic and stochastic forces will have changed the particle position.

FIG. 2. Timing diagram for the feedback trap. The particle's position is averaged over an exposure time t_c, and forces are updated after t_s. The observed position $\bar{x}_n$ is used for calculating feedback force F_n. The force is applied with a time delay $t_d = t_s$, measured from the middle of the exposure. Both the deterministic force F_n and the thermal force ξ_n affect the unknown real position x_{n+1}.

of motion that is more convenient for numerical simulations and for inferring parameter values.

A. Coupled one-dimensional equations of motion

Let F_n be the force due to the virtual potential, held constant over the time interval t_s, defined by $[t_n, t_{n+1}) \equiv [nt_s, (n+1)t_s)$. An important point is that the virtual potential is based on the observed position $\bar{x}_n$ rather than the unknown true position x_n. The observed position $\bar{x}_n$ is inferred from a camera image that averages the motion over a time $t_c < t_s$. The response to each camera exposure is to update the desired force and necessarily involves a finite time delay, t_d, which must be at least $\frac{1}{2}t_c$. In our experiments, for simplicity, we set $t_d = t_s$, as illustrated in Fig. 2.

With that choice of delay, the dynamics are determined by two coupled, discrete stochastic equations for x_n and $\bar{x}_n$. In the overdamped limit, which applies in all cases we study, the equations are[18]

$$x_{n+1} = x_n + \frac{1}{\gamma} F_n t_s + \xi_n, \tag{1a}$$

$$\bar{x}_{n+1} = x_n + \frac{1}{8\gamma}(F_n - F_{n-1})t_c - \xi_n^{(0)} + \bar{\xi}_n + \sigma_n. \tag{1b}$$

In Eq. (1a), the true position x_{n+1} depends on the deterministic force $F_n = -\partial_x U(\bar{x}_n, t_n)$ applied during time t_s and on the thermal noise ξ_n. The particle drag coefficient γ gives the response to the feedback force and is related to the diffusion coefficient by Einstein's law, $D = kT/\gamma$. As we discuss below in Sec. VII B, the confined vertical geometry implies that the drag coefficient γ is significantly larger than the Stokes-Einstein value for an isolated sphere. The effects of thermal fluctuations are approximated, as usual, by a Gaussian random variable ξ_n, which satisfies $\langle \xi_n \rangle = 0$ and $\langle \xi_n \xi_m \rangle = 2Dt_s \delta_{mn}$, where δ_{mn} is the Kronecker delta symbol.

In Eq. (1b), the observed position $\bar{x}_{n+1}$ depends on the true position x_n but also on the feedback forces during previous time steps, via the term $\frac{1}{8\gamma}(F_n - F_{n-1})t_c$. This term results from the finite camera exposure: the timing of the feedback trap is set so that the force is updated, from F_{n-1} to F_n, at precisely the midpoint of the camera exposure. A change in force at the update then biases the position, accounting for the $\frac{1}{8\gamma}(F_n - F_{n-1})t_c$ term. The noise terms include the effects of thermal noise as averaged over the camera exposure t_c, and the observation noise σ_n. A detailed explanation and derivation of Eq. (1) is given in Ref. 18.

B. Reduction to a single equation of motion

It is convenient to eliminate the unobservable true position x_n from Eq. (1), reducing the two coupled equations to a single relation that is more suitable for simulation and further analysis. To accomplish this reduction, we convert absolute positions into intervals, defining $\Delta x_n \equiv x_{n+1} - x_n$ for true positions and $\overline{\Delta x}_n \equiv \bar{x}_{n+1} - \bar{x}_n$ for observed positions. Then,

$$\overline{\Delta x}_n = \Delta x_{n-1} + \frac{1}{8\gamma}(F_n - 2F_{n-1} + F_{n-2})t_c - \xi_n^{(0)}$$
$$+ \xi_{n-1}^{(0)} + \bar{\xi}_n - \bar{\xi}_{n-1} + \sigma_n - \sigma_{n-1}. \tag{2}$$

Substituting $\Delta x_{n-1} = \frac{1}{\gamma} F_{n-1}t_s + \xi_{n-1}$ from Eq. (1a) gives

$$\overline{\Delta x}_n = \frac{t_s}{\gamma} \underbrace{\left[F_{n-1} + \frac{1}{8}\frac{t_c}{t_s}(F_n - 2F_{n-1} + F_{n-2}) \right]}_{\text{deterministic forces} \equiv \bar{F}_{n-1}}$$

$$+ \underbrace{\xi_{n-1} - \xi_n^{(0)} + \xi_{n-1}^{(0)} + \bar{\xi}_n - \bar{\xi}_{n-1} + \sigma_n - \sigma_{n-1}}_{\text{stochastic terms} \equiv \zeta_n}$$

$$\equiv \frac{t_s}{\gamma} \bar{F}_{n-1} + \zeta_n, \tag{3}$$

where terms proportional to t_s/γ are deterministic and are collected into a single effective force, $\bar{F}_{n-1}$. The camera-exposure corrections are small (typically $\lesssim 6\%$) so that $\bar{F}_{n-1} \approx F_{n-1}$. See Appendix A for a discussion.

The terms forming ζ_n in the second grouping are stochastic and result from thermal fluctuations and observation noise. The camera integrates Brownian motion of the particle over the exposure time t_c, while the observation noise σ_n results from the finite resolution of the optical microscope and the finite number of photons collected during the camera exposure.

Since ζ_n is a linear combination of Gaussian random variables with zero mean, it, too, has zero mean and is characterized solely by its covariance. The effective noise ζ_n is not white but has correlations[18–21]

$$\langle \zeta_n^2 \rangle = 2Dt_s - \frac{2}{3}Dt_c + 2\sigma^2,$$

$$\langle \zeta_n \zeta_{n-1} \rangle = \frac{1}{3}Dt_c - \sigma^2, \tag{4}$$

$$\langle \zeta_n \zeta_{n-p} \rangle = 0, \qquad p \geq 2.$$

The cross-correlation $\langle \zeta_n \zeta_{n-1} \rangle$ arises because neighboring intervals, $\overline{\Delta x}_n = \bar{x}_{n+1} - \bar{x}_n$ and $\overline{\Delta x}_{n-1} = \bar{x}_n - \bar{x}_{n-1}$ share the noise that is present in measurement $\bar{x}_n$. Equation (4) shows that nearest-neighbor correlations can be positive or negative, or even zero with a judicious choice of t_c. Note that the observation noise variance σ^2 is shot-noise limited and thus $\sim t_c^{-1}$.

For simulations, it is convenient to write the correlated noise term ζ_n in terms of uncorrelated Gaussian random variables ψ_n, with $\langle \psi_n \rangle = 0$ and $\langle \psi_n \psi_m \rangle = \delta_{nm}$. More succinctly, $\psi_n \sim \mathcal{N}(0, 1)$, and

$$\zeta_n \equiv c_+ \psi_n + c_- \psi_{n-1},$$

$$c_\pm = \frac{1}{2}\left(\sqrt{2Dt_s} \pm \sqrt{2Dt_s - \frac{4}{3}Dt_c + 4\sigma^2} \right), \tag{5}$$

where the constants $c_\pm$ are chosen to satisfy Eq. (4).

C. Generating feedback voltages

Equation (3) does not fully specify the equations of motion, since we still need to relate the forces F_n to the applied voltages V_n and to the desired virtual potential $U(x, t)$. We first relate the desired forces to voltages placed across the electrodes in the feedback trap. For the range of applied voltages (generally, $|V_n| < 10$ V), the response is linear, so that we expect $F_n = \frac{\mu}{\gamma} V_n$, where V_n is the voltage applied across

the electrodes, γ is the particle drag, and μ is the mobility. Empirically, however, we find that $F_n = \frac{\mu}{\gamma}(V_n - V_0)$, where the voltage V_0 leads to a drift and reflects amplifier and electrochemical offsets in the applied voltage. The drift is important on time scales of minutes and longer. Substituting this relation into the expression for $\bar{F}_{n-1}$ in Eq. (3) shows that $\bar{F}_{n-1} = \frac{\mu}{\gamma}(\bar{V}_{n-1} - V_0)$, where

$$\bar{V}_{n-1} = V_{n-1} + \frac{1}{8}\frac{t_c}{t_s}(V_n - 2V_{n-1} + V_{n-2}). \quad (6)$$

The equation of motion then becomes

$$\overline{\Delta x}_n = t_s \mu (\bar{V}_{n-1} - V_0) + \zeta_n. \quad (7)$$

We note that "mobility" is not quite the correct terminology, as the standard definition relates the particle drift velocity to the local field; however, since determining fields is difficult, it is common practice to state a mobility in terms of the potential drop across the electrodes.[4]

We next need to relate forces to the desired virtual potential $U(x, t)$. At time $t_{n-1} = (n-1)t_s$, we output the voltage V_{n-1}, which is based on the measurement $\bar{x}_{n-1}$ (whose midpoint is at t_{n-2}, as shown in Fig. 2). We choose this voltage so that

$$\frac{\mu}{\gamma}(V_{n-1} - V_0) = -\partial_x U(\bar{x}_{n-1}, t_{n-1}). \quad (8)$$

In principle, we should use $\bar{V}_{n-1}$ in Eq. (8); however, that choice leads to a numerically unstable algorithm because it involves taking the ratio of two small numbers, $\frac{t_c}{8t_s}$ and $[\partial_x U(x_{n-1}, t_{n-1}) - \frac{\mu}{\gamma}V_{n-1}]$. Thus, we take advantage of the fact that $\bar{V}_n \approx V_n$ in writing Eq. (8). (See Appendix A.) Explicitly,

$$V_{n-1} = -\mu^{-1}\partial_x U(x_{n-1}, t_{n-1})\gamma + V_0. \quad (9)$$

A final simplification is to redefine the drag γ in terms of diffusion using Einstein's relation, $\gamma = (kT)/D$ and to write the potential in units of kT. Then, collecting all the equations together, we have

$$\overline{\Delta x}_n = t_s \mu (\bar{V}_{n-1} - V_0) + \zeta_n,$$

$$\bar{V}_{n-1} \equiv V_{n-1} + \left(\frac{1}{8}\right)\left(\frac{t_c}{t_s}\right)(V_n - 2V_{n-1} + V_{n-2}),$$

$$V_n = -\mu^{-1}\partial_x U(x_n, t_n)/D + V_0,$$

$$\zeta_n = c_+ \psi_n + c_- \psi_{n-1}, \quad (10)$$

$$c_\pm = \frac{1}{2}\left(\sqrt{2Dt_s} \pm \sqrt{2Dt_s - \frac{4}{3}Dt_c + 4\sigma}\right)^2,$$

$$\psi_n \sim \mathcal{N}(0, 1).$$

Finally, we note that Eqs. (10) assume that, over the timescale t_s, the equivalent continuous potential does not change significantly. Such changes can occur in two ways: by the motion of the particle in a fixed potential and by the time-dependence of the potential itself. For the former, we ask that $\alpha \equiv t_s/t_r \ll 1$, where t_r is the relaxation time for motion in a potential. In a harmonic potential with force constant κ, the relaxation time would be $t_r = \sqrt{\gamma/\kappa}$. Here, with a general $U(x, t)$, the

force constant generalizes to $\kappa = -\partial_{xx}U(x, t)$, which is approximately the curvature of the potential function. Conservatively, $\kappa(x, t)$ should be evaluated at the point of maximum curvature. Note that Eq. (8) is for a single force F_{n-1} and must be generalized to $\bar{F}_{n-1}$ using the definition in Eq. (3).

IV. ONLINE PARAMETER ESTIMATION

Equation (10) contains four undetermined parameters: the mobility μ, drift V_0, and noise terms $c_\pm$. From $c_\pm$, we can deduce D and σ, assuming that t_s and t_c are known. (The latter are known, either because the hardware gives deterministic control over timing or because we measure t_s and t_c independently, as described in Ref. 17.) As discussed in the Introduction, we need to measure the parameters experimentally in real time, while the experiment is running, as opposed to off-line analysis of recorded data. Real-time parameter values are required in order to impose correctly the proper virtual potential. In particular,

- μ relates displacements to voltages;
- V_0 allows drift compensation;
- $c_\pm$ sets the scale of the potential relative to kT.

Since most of our experiments last several days and since photobleaching limits the particle lifetime in a feedback trap to a few hours, we need to acquire data for several different particles. Each particle has its own radius and charge, which translates to an individual diffusion coefficient D and mobility μ, which must be estimated. The mobility also varies significantly with location in the cell. The voltage offset $|V_0|$ is typically ≈ 200 mV and the mobility ≈ 10 μm/s/V. Together, these lead to drift velocities $v_0 = \mu V_0$ that are typically 2 μm/s. The drift terms become comparable to diffusion on time scales of roughly a second, with $v_0 t^* \sim \sqrt{Dt^*} \Rightarrow t^* \sim D/v_0^2 \approx 1$ s. (These are worst-case estimates; often $t^* \approx 30$ s.) We thus need an algorithm that can calculate a running average of the parameter estimates over many time steps while still being fast enough to update at each time step, since at each time step we need to output the correct force, based on the current calibration. The RLS algorithm described in Sec. IV A can fulfill both requirements.

A. Recursive least squares

We begin by formulating the parameter-estimation problem as a linear, least-squares curve-fit. We first rewrite Eq. (10) in a vector form

$$\overline{\Delta x}_n = \boldsymbol{\varphi}_n^T \boldsymbol{\theta} + \zeta_n, \quad (11)$$

where $\boldsymbol{\varphi}_n^T = (\bar{V}_{n-1} \ 1)$ and $\boldsymbol{\theta} = t_s \mu \left(\begin{smallmatrix} 1 \\ -V_0 \end{smallmatrix}\right)$. If we neglect, for now, correlations in the noise term ζ_n, the optimal estimate $\hat{\boldsymbol{\theta}}$ of the parameters is determined by minimizing the function

$$\chi^2(\boldsymbol{\theta}) = \sum_{n=0}^{N}\left(\overline{\Delta x}_n - \boldsymbol{\varphi}_n^T \boldsymbol{\theta}\right)^2. \quad (12)$$

Because parameter values drift, we will need to estimate them at every time step. Although in principle one could

minimize $\chi^2(\boldsymbol{\theta})$ at each time step, it is well known that the least-squares problem can be formulated *recursively*, with updated estimates of the parameters inferred from old estimates and new data [see, e.g., Ref. 22, Chap. 2]. If we again assume decorrelated noise, the recursive algorithm is given by iterating the three RLS equations

$$\hat{\boldsymbol{\theta}}_{n+1} = \hat{\boldsymbol{\theta}}_n + \boldsymbol{L}_{n+1}\varepsilon_n,$$

$$\boldsymbol{L}_{n+1} = \frac{\boldsymbol{P}_n\boldsymbol{\varphi}_{n+1}}{1 + \boldsymbol{\varphi}_{n+1}^T\boldsymbol{P}_n\boldsymbol{\varphi}_{n+1}}, \qquad (13)$$

$$\boldsymbol{P}_{n+1} = \left(\boldsymbol{I} - \boldsymbol{L}_{n+1}\boldsymbol{\varphi}_{n+1}^T\right)\boldsymbol{P}_n,$$

where $\boldsymbol{P}_{n+1}$ is the parameter covariance matrix and where $\varepsilon_n = \overline{\Delta x}_{n+1} - \boldsymbol{\varphi}_{n+1}^T\hat{\boldsymbol{\theta}}_n$ defines the *innovations*, the difference between observed and predicted displacements. The RLS algorithm is a simplified version of the *Kalman filter*,[2] and the vector $\boldsymbol{L}_n$ is known as the *Kalman gain*: it gives the relative weight of the old parameter estimates and the new information contained in ε_n. The estimate $\hat{\boldsymbol{\theta}}_n$ includes all data collected up to timestep n.

B. Decorrelating the noise

If one does not account for correlations in the noise term ζ_n, the parameter estimates will be biased. We can avoid bias by transforming to new coordinates where the noise terms are independent.[22,23] We change variables in Eq. (3) by first applying the Z transform, or generating function, which is a discrete version of the Laplace transform. For the sequence ζ_n, the Z transform is

$$\mathcal{Z}[\zeta_n] = \zeta(z) \equiv \sum_{n=0}^{\infty}\zeta_n z^{-n}. \qquad (14)$$

Since $\zeta_n = c_+\psi_n + c_-\psi_{n-1}$ and $\mathcal{Z}[\psi_{n-1}] = z^{-1}\psi(z)$, Eq. (11) implies that

$$\overline{\Delta x}(z) = [\boldsymbol{\varphi}(z)]^T\boldsymbol{\theta} + (c_+ + c_-z^{-1})\psi(z), \qquad (15)$$

where $\boldsymbol{\varphi}^T(z) = \left(\bar{V}(z)\ \frac{1}{1-z^{-1}}\right)$. Dividing Eq. (15) by $(c_+ + c_-z^{-1})$ then gives

$$\overline{\Delta x}^{(f)}(z) = \left[\boldsymbol{\varphi}^{(f)}(z)\right]^T\boldsymbol{\theta} + \psi(z), \qquad (16)$$

where the *filtered* versions of $\overline{\Delta x}(z)$ and $\boldsymbol{\varphi}(z)$ are

$$\overline{\Delta x}^{(f)}(z) = \frac{\overline{\Delta x}(z)}{c_+ + c_-z^{-1}},$$

$$\boldsymbol{\varphi}^{(f)}(z) = \left(\frac{\bar{V}(z)}{c_+ + c_-z^{-1}}\ \left(\frac{1}{1-z^{-1}}\right)\left(\frac{1}{c_+ + c_-z^{-1}}\right)\right). \qquad (17)$$

Multiplying Eq. (17) by $(c_+ + c_-z^{-1})$ and inverting the Z transform gives recursive formulas for the filtered observed position $\Delta\bar{x}_n^{(f)}$ and voltage term $\boldsymbol{\varphi}_n^{(f)}$

$$\Delta\bar{x}_n = c_+\Delta\bar{x}_n^{(f)} + c_-\Delta\bar{x}_{n-1}^{(f)}$$

$$\Rightarrow \quad \Delta\bar{x}_n^{(f)} = \frac{1}{c_+}\left(\Delta\bar{x}_n - c_-\Delta\bar{x}_{n-1}^{(f)}\right),$$

$$\boldsymbol{\varphi}_n = c_+\boldsymbol{\varphi}_n^{(f)} + c_-\boldsymbol{\varphi}_{n-1}^{(f)} \qquad (18)$$

$$\Rightarrow \quad \boldsymbol{\varphi}_n^{(f)} = \frac{1}{c_+}\left(\boldsymbol{\varphi}_n - c_-\boldsymbol{\varphi}_{n-1}^{(f)}\right).$$

In components, the form of the filtered input in Eq. (18) is $(\boldsymbol{\varphi}_n^{(f)})^T = (\bar{V}_{n-1}^{(f)}\ c_0^{-1})$, where $c_0 = c_+ + c_- = \sqrt{2Dt_s}$. In terms of the filtered variables, the relation between displacement and voltages becomes

$$\Delta\bar{x}_n^{(f)} = \left(\boldsymbol{\varphi}_n^{(f)}\right)^T\boldsymbol{\theta} + \psi_n, \qquad (19)$$

Thus, we first recursively filter $\bar{x}_n$ and $\boldsymbol{\varphi}_n$ and then use the resulting $\bar{x}_n^{(f)}$ and $\varphi_n^{(f)}$ in an ordinary RLS algorithm to estimate $\hat{\boldsymbol{\theta}}_n$.

C. Time-dependent parameters

The algorithms for determining μ_n and V_{0n} given in Sec. IV B implicitly assume that the underlying parameter values are constant. Empirically, they drift. To account for the drift, we can reformulate a running-average version of RLS that weights recent observations more than ones taken in the past.[22] For N measurements, we write the exponentially weighted $\chi^2(\boldsymbol{\theta})$ function as

$$\chi^2(\boldsymbol{\theta}) = \sum_{n=0}^{N}\lambda^{N-n}\left\{\overline{\Delta x}_n^{(f)} - \left[\boldsymbol{\varphi}_n^{(f)}\right]^T\boldsymbol{\theta}\right\}^2, \qquad (20)$$

which leads to a slightly altered version of the RLS equations (Eqs. (13)),

$$\hat{\boldsymbol{\theta}}_{n+1} = \hat{\boldsymbol{\theta}}_n + \boldsymbol{L}_{n+1}\varepsilon_n,$$

$$\boldsymbol{L}_{n+1} = \frac{\boldsymbol{P}_n\boldsymbol{\varphi}_{n+1}^{(f)}}{\lambda + \left[\boldsymbol{\varphi}_{n+1}^{(f)}\right]^T\boldsymbol{P}_n\boldsymbol{\varphi}_{n+1}^{(f)}}, \qquad (21)$$

$$\boldsymbol{P}_{n+1} = \frac{1}{\lambda}\left(\boldsymbol{I} - \boldsymbol{L}_{n+1}\left[\boldsymbol{\varphi}_{n+1}^{(f)}\right]^T\right)\boldsymbol{P}_n,$$

In Eq. (21), the *forgetting factor* $\lambda \in (0, 1)$, with $\lambda = 1$ implying that all measurements are equally weighted. The forgetting factor is conveniently expressed in terms of a timescale as $\lambda = 1 - 1/\tau$, since $\lambda^n = (1 - 1/\tau)^n \approx e^{-n/\tau}$, with n an integer and τ in units of the time step t_s. The forgetting time τ should be chosen shorter than the drift, to track parameter variations.

Finally, we estimate the particle diffusion coefficient D and the observation noise variance σ^2. The equations for $c_{\pm}$ relate the noise correlations of ζ_n to the diffusion constant and observation noise. From Eq. (10),

$$\zeta_n = \overline{\Delta x}_n - \mu_n t_s(\bar{V}_{n-1} - (V_0)_{n-1}). \qquad (22)$$

After obtaining ζ_n, we calculate running averages of the variance and correlation functions

$$\langle \zeta^2 \rangle_n = \lambda \langle \zeta^2 \rangle_{n-1} + (1-\lambda)\zeta_n^2,$$

$$\langle \zeta\zeta_- \rangle_n = \lambda \langle \zeta\zeta_- \rangle_{n-1} + (1-\lambda)(\zeta\zeta_-)_n, \tag{23}$$

where λ again sets the filtering time, $\langle \zeta^2 \rangle_n$ and $\langle \zeta\zeta_- \rangle_n$ are estimates of the variance and unit-lag covariance, respectively, and $(\zeta\zeta_-)_n$ is the new lag-one covariance at timestep $n-1$. The diffusion and the observation noise are then

$$D_n = \frac{1}{2t_s}(\langle \zeta^2 \rangle_n + 2\langle \zeta\zeta_- \rangle_n),$$

$$\sigma_n^2 = \frac{1}{3} D_n t_c - \langle \zeta\zeta_- \rangle_n. \tag{24}$$

The estimators in Eq. (24) are nearly optimal for short single-particle trajectories.[24] Here, "optimal" means reaching the Cramér-Rao lower bound for the variance of an unbiased estimator. The estimators in Eq. (24) are optimal for large signal-to-noise ratio, $\mathrm{SNR} = \sqrt{Dt_s}/\sigma \gg 1$. Here, $\mathrm{SNR} \approx 1$, and the estimator has a slightly larger variance than the best possible estimator. (See Fig. 4(d) of Ref. 24.) The simplicity and short calculation time of Eqs. (23) and (24) make the estimator a good choice for the present problem.

D. Recursive maximum likelihood

Notice that estimating D and σ (equivalently, $c_\pm$) requires estimates of μ and V_0 (see Eq. (22)), while estimates of μ and V_0 depend on the filtering operation to decorrelate the noise and require estimates of $c_\pm$ (see Eq. (17)). We can determine all four parameters self-consistently, a procedure that is known as the RML algorithm.[22,23] Here, the algorithm consists of solving Eqs. (18) and (21)–(24) together.

Although the set of material and noise parameters that minimizes the χ^2 criterion in Eq. (20) is a solution to the RML algorithm, the estimates can diverge, as the coupling between noise parameters and the estimates of the material parameters makes the equations nonlinear. To make the algorithm converge, we use nominal $c_\pm$ values initially to estimate μ and V_0. Then, after the initial estimates for μ and V_0 have stabilized, we use those values to refine $c_\pm$ and again estimate μ and V_0. To check that the RML algorithm converges to the correct values, we have performed two tests:

- We simulated time series and confirmed that the inferred parameter values for all four parameters were consistent with the simulation values. (See Sec. VI.)
- We independently measured the observation noise directly, using the variance in the apparent position of an immobilized bead on a glass surface.[17] The "stuck bead" values of σ^2 agreed with those found for diffusing particles using the RML algorithm.

For more details on the RLM algorithm, see Sec. 6.1.3 of Ref. 23.

V. TWO-DIMENSIONAL FEEDBACK TRAP

For simplicity, the above discussion was for one dimension (1D), while the actual experiment explores two-dimensional (2D) motion, with the particle confined in the z direction by using a thin cell. The lateral x-y coordinate system is defined in terms of the camera's pixel array. Most of the previous discussion then directly generalizes to two dimensions. Since the applied electric fields are not along the camera coordinate axes, the x-y equations of motion are coupled and must be unscrambled.

Two sets of electrodes, Pairs 1 and 2, impose a 2D virtual potential. The schematic diagram of the setup is given in Ref. 17. Empirically, the fields from Pairs 1 and 2 differ by up to 60% in magnitude and deviate from the x and y axes by up to 45°. We account for these effects by introducing a mobility matrix $\boldsymbol{\mu}$,

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_{x1} & \mu_{x2} \\ \mu_{y1} & \mu_{y2} \end{pmatrix}, \tag{25}$$

where the subscripts indicate the transformation between V_1 and V_2 to x and y displacements. Although $\boldsymbol{\mu}$ varies with position inside the cell (which measures 2 mm square), we find it to be constant over the scale of virtual potentials (several microns) at a fixed position within the cell. We note that $\boldsymbol{\mu}$ is proportional to the mobility $\times t_s$, with a geometrical factor relating applied potentials at the electrodes to fields at the particle that must be calibrated empirically.

The 2D version of Eq. (10) then is

$$\bar{\boldsymbol{x}}_{n+1} = \bar{\boldsymbol{x}}_n + t_s\boldsymbol{\mu}(\bar{\boldsymbol{V}}_{n-1} - \boldsymbol{V}_0) + \boldsymbol{\zeta}_n, \tag{26}$$

where all bold quantities other than $\boldsymbol{\mu}$ are 2D vectors. The rest of the 1D analysis carries forward exactly as before, leading to filtered displacement equations of the form

$$\Delta\bar{\boldsymbol{x}}_n^{(f)} = \big[\boldsymbol{\varphi}_n^{(f)}\big]^T \boldsymbol{\theta} + \boldsymbol{\psi}_n, \tag{27}$$

where $\boldsymbol{\theta}$ now has 6 elements (4 from the matrix $\boldsymbol{\mu}$ and 2 from $\boldsymbol{V_0}$). The voltages are collected into a 2×6 matrix, which has only 2 independent components, made from the voltages from electrode pairs 1 and 2. The noise $\boldsymbol{\psi}_n \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{1})$, where $\boldsymbol{0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ and $\boldsymbol{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

Rather than writing out Eq. (27) in components, it is more convenient to split it into two uncoupled equations with two individual RLS updates that are each based on three parameters, $\boldsymbol{\theta}_x^T = t_s\big(\mu_{x1} \ \ \mu_{x2} \ \ \mu_{x1}V_0^{(1)} + \mu_{x2}V_0^{(2)}\big)$ and $\boldsymbol{\theta}_y^T = t_s\big(\mu_{y1} \ \ \mu_{y2} \ \ \mu_{y1}V_0^{(1)} + \mu_{y2}V_0^{(2)}\big)$. These equations are

$$\overline{\Delta x_n}^{(f)} = t_s\big(\bar{V}_n^{(1)} \ \ \bar{V}_n^{(2)} \ \ 1\big)^{(f)} \begin{pmatrix} \mu_{x1} \\ \mu_{x2} \\ \mu_{x1}V_0^{(1)} + \mu_{x2}V_0^{(2)} \end{pmatrix} + \psi_n^{(x)},$$

$$\tag{28}$$

$$\Delta\bar{y}_n^{(f)} = t_s\big(\bar{V}_n^{(1)} \ \ \bar{V}_n^{(2)} \ \ 1\big)^{(f)} \begin{pmatrix} \mu_{y1} \\ \mu_{y2} \\ \mu_{y1}V_0^{(1)} + \mu_{y2}V_0^{(2)} \end{pmatrix} + \psi_n^{(y)}.$$

From the RLS fit algorithm,[22] we calculate at each time step two vectors of length 3 that contain time-varying estimates of each fit parameter. We also calculate two 3×3 covariance matrices that give the uncertainties of the best parameter estimates. We update the best estimate and covariance matrix at each time step. Since the 3×3 covariance matrix depends only on the inputs, it is the same for both equations and thus calculated only once per time step. The forgetting algorithm is used to estimate $\boldsymbol{\mu}_n$ and $\boldsymbol{V}_{0n}$, as in the 1D case. The running averages of the diagonal elements of the matrices $\langle \boldsymbol{\zeta}_n \boldsymbol{\zeta}_n^T \rangle$ and $\langle \boldsymbol{\zeta}_n \boldsymbol{\zeta}_{n-1}^T \rangle$ are used to estimate the vectors $\boldsymbol{D}_n$ and $\boldsymbol{\sigma}_n$ via Eq. (24). (Physically, of course, we expect D and σ to be the same along x and y. Calculating the two components independently checks that we have correctly decoupled the dynamics.)

From the imposed 2D potential $U(x, y, t)$, we generate forces by taking the negative gradient of the potential, $\boldsymbol{F}_n \equiv -\nabla U(\bar{\boldsymbol{x}}_n, \bar{\boldsymbol{y}}_n, t_n)$. As in Sec. III C, we update the voltages at each time step as

$$\boldsymbol{V}_n = -\boldsymbol{\mu}^{-1} [\boldsymbol{D}^{-1} \nabla U(\bar{\boldsymbol{x}}_n, t_n)] + \boldsymbol{V}_0, \qquad (29)$$

where $\boldsymbol{D} = \begin{pmatrix} D_x & 0 \\ 0 & D_y \end{pmatrix}$. One small effect that we do not model is that the noise components along x and y have a small cross-correlation arising from the fact that the number of photons detected is the same for both axes. We do not observe any effects traceable to this small correlation.

VI. SIMULATIONS

Simulations of the particle dynamics are useful in showing that the rather complicated RLM algorithm we propose here actually works. That is, we will show that we can simulate a data set with known parameters ($\boldsymbol{\mu}$, $\boldsymbol{V_0}$, $\boldsymbol{D}$, and $\boldsymbol{\sigma}$) and recover their values accurately.

A. RLS estimate

In Sec. IV, we introduced the recursive least squares (RLS) algorithm to determine material parameters in real time, with its extension to filtering to decorrelate the noise. Here, we use simulations to test the basic RLS algorithm and show that, once noise has been decorrelated, it converges to the correct values of material parameters in a feedback trap. The method proposed in Sec. IV A does not depend on the shape of potential or on how voltages are applied; rather, the only requirements are that the voltages must vary sufficiently, so that they are sufficiently *persistent*, in the language of adaptive control.[22] The greater the voltage range explored, the faster the algorithm converges.

In our experiments, we initially trap a particle in a virtual harmonic potential, to let the estimates of its material parameters (mobility and diffusion constant) converge sufficiently. To check the procedure, we simulate a particle in the potential $U = \frac{1}{2}\kappa x^2$ and estimate its properties. In Fig. 3, we show both filtered and naive estimates of the unknown mobility. The naive RLS estimate, which neglects nearest-neighbor noise correlations, is biased down by $\approx 9\%$. In Sec. VII A (Fig. 5), we show the simulation and experimental results.
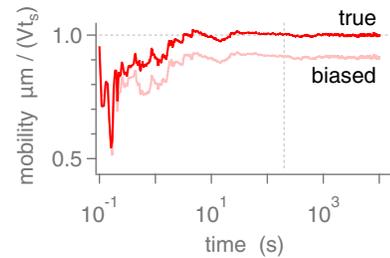

FIG. 3. Simulation of RML mobility estimate without (gray curve) and with (black curve) proper noise decorrelation. Dashed line marks the mobility value used in the simulation, $\mu = 1 \ \mu\text{m}/(Vt_s)$. Vertical dashed line shows time used for calibration, 200 s.

We checked parameter convergence in two stages. First, we assumed that we knew in advance both the diffusion constant D and the observation noise σ. The latter can be pre-calibrated accurately, but the former varies from particle to particle—by a large amount, if the particle is an aggregate and by a small amount if an "elementary" size. But the first stage of parameter estimation requires a nominal value of D. In a second set of simulations, we checked whether a bad guess affected parameter convergence and saw no difference, even when the initial estimate for D was off by a factor of ten.

VII. EXPERIMENTAL DATA

In Secs. II–VI, we developed theoretical methods for measuring particle properties in a feedback trap. The framework handles properly (decorrelates) the different sources of noise in the feedback trap and was tested using simulations in Sec. VI A. Here, we show typical experimental data confirming that the various parameters do converge in practical settings. Furthermore, we use values of those parameters to run the simulation, and test the self-consistency between theory, simulation, and experiment.

We discussed a preliminary version of our experimental setup in Ref. 17 and gave first measurements of power spectrum and variance, as inferred from time series of position measurements in a virtual harmonic potential. In that work, we were not able to measure particle properties while trapping. The imposed feedback gain (equivalent to mobility) did not, in fact, match that measured in post-experiment analysis. Moreover, the feedback gain drifted in time and was not constant. As a result, we had to treat both the average feedback gain and diffusion coefficient as free parameters in a curve fit. With the development of the RML method presented here, we no longer need to fit parameters to the power spectrum. We show that we can use the RML algorithm to estimate the parameters and then simply plot the power spectrum based on those parameters. Figure 4 shows the remarkable agreement that we can now achieve. In particular, we note that the solid line, calculated according to the theory in Ref. 18, is not a fit but rather a plot, based on parameters taken from the RML formalism presented here. This agreement justifies the rather complicated RML analysis of the parameters, the results of which we now describe in detail.
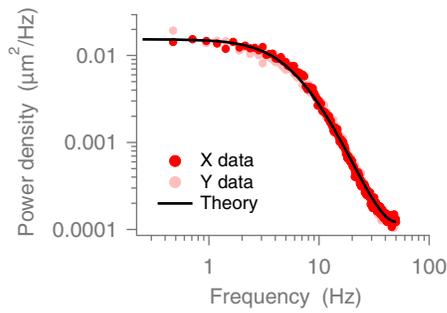
FIG. 4. Power spectra for a particle in a feedback trap. Dark (light) red markers indicate position measurements along the x- (y)-axis. Solid black line shows the prediction from Ref. 18, plotted (not fit) using the independently inferred parameters.

A. Recursive maximum likelihood algorithm

The experimental test of our implementation of the RML algorithm (Fig. 4) uses a harmonic virtual potential with feedback gain $\alpha' = 0.2$. At each time step, we apply a force proportional to the observed position, where α' is the proportionality coefficient, $F_n = -\alpha' \frac{1}{Dt_s} \bar{x}_n$. For small feedback gains, $\alpha' \approx \alpha = t_r/t_s$, which was defined above. For the special case of a virtual harmonic potential, Eq. (29) implies that output voltages are calculated as $V_n = -\alpha' \mu^{-1} \bar{x}_n/t_s + V_0$. In particular, they do not explicitly depend on a particle's diffusion constant. Nevertheless, the RML estimation algorithm for μ^{-1} and V_0 uses the diffusion constant as an input, and the constant also affects the power spectrum density calculated from the particle's position measurements.

As discussed in Appendix B, we adjust the forgetting parameter λ in several stages (in order to get the RML algorithm to converge). Figure 5 shows an example of particle parameters recorded during an experiment run (a) and the simulation result (b). We estimate ten parameters: four for the mobility μ, two for the drift V_0, two for the diffusion D, and two for the observational noise σ. The time series in Fig. 5 shows several stages in the convergence of three parameters: the mobility component μ_{x1}, the drift voltage along one pair of electrodes $V_{0(1)}$, and the diffusion D along x-axis. The full convergence occurs over five stages, denoted (i)–(v). Stage (i) occurs after a new particle is detected. During this time, the illumination is adjusted so that the light intensity detected from the trapped bead matches the setpoint value. Feedback voltages are generated using initial guesses for inverse mobility and drift. In Stage (ii), the RML estimate is turned on, with a short time constant $\tau = 10$, which is increased to $\tau = 100$ in Stage (iii). (The time constants are given in units of $t_s = 10$ ms.) The initial guesses are replaced by their RML estimates in Stage (iv). Finally, in Stage (v), the time constant is set to $\tau = 1000$, and we record data for further analysis. Returning to Fig. 4, we see that the good agreement between power-spectrum data and the corresponding theory from[18] requires accurate estimates of the timing parameters $t_s = 10$ ms, $t_d = 10$ ms (delay), and $t_c = 5$ ms, of the observation noise $\sigma = 40$ nm, and of the diffusion constant $D = 1.57 \pm 0.06\ \mu m^2/s$. The diffusion constant is calculated from the mean value of the RML estimate for D shown in Fig. 5, and its uncertainty is a systematic error due to the length calibration. As discussed below, the statistical error is negligible.

Figure 4 also shows that the experimental data along both axes are similar, as expected physically (both directions are equivalent). The result has stronger implications, as it also means that we have properly estimated the off-diagonal elements of the mobility matrix. Incorrect values would lead to differences in the power spectrum, as well as cross
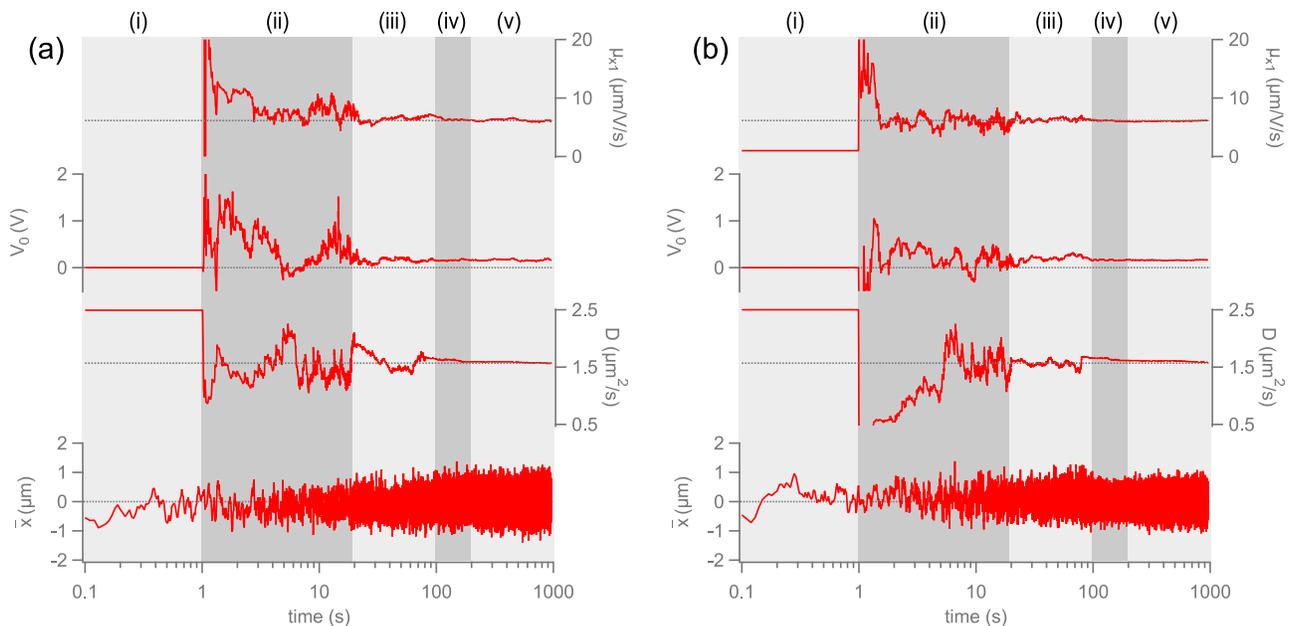

FIG. 5. Estimates of the mobility component μ_{x1}, offset voltage along one pair of electrodes V_0, diffusion coefficient D, and observed position $\bar{x}$ along the x-axis from experiment (a) and simulation (b). Estimates of unknown parameters converge in 5 stages as the convergence time τ is adjusted, as discussed in Appendix B 2: (i) No RLS estimate. (ii) and (iii) Initial convergence with $\tau = 10$ and $\tau = 100$, respectively. (iv) Initial guesses for particle's properties are replaced by RLS estimates. (v) Full convergence, with time constant set to $\tau = 1000$. We obtain the simulation parameters from the mean value of experimental data in Stage (v). $\mu_{x1} = 6.08\ \mu m^2/s$, $V_0 = 0.16$ V, $D = 1.57\ \mu m^2/s$.

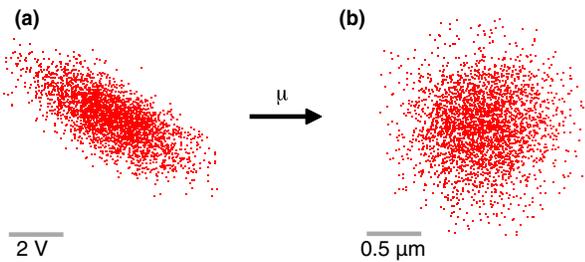FIG. 6. Experimental scatter plot showing decorrelation transformation (the matrix $\boldsymbol{\mu}$ defined in Eq. (25)) between voltages and positions. (a) Voltage applied to Electrode 2 vs. voltage applied to Electrode 1. (b) Position measurements, y vs. x.

correlations (arising because a voltage that is supposedly aligned along one camera axis has components along the other axis). In Fig. 6, we show how the applied voltages affect the observed position. Although the applied voltages are highly correlated [Fig. 6(a)], due to mobility matrix $\boldsymbol{\mu}$, they create an independent and uncorrelated position measurements [Fig. 6(b)]. We use Fig. 6(b), together with the power spectrum analysis in Fig. 4, to test whether particle dynamics follow the imposed virtual potential.

B. Estimating the diffusion constant

The RML estimation algorithm successfully converges to the various parameter values used in numerical simulations and experimental runs. But are the values that we deduce in the experiment correct? To answer this question, we must first estimate independently the expected values of these parameters. For the mobility, the charge on each particle is unknown, and the complexity of the electrode and cell geometry means that we do not know the electric field at the particle—only the voltage at the electrodes. Mobility measurements are thus relative.

We can do better estimating the lateral diffusion constant. For an isolated sphere of radius r in a fluid of viscosity η of infinite extent, the Stokes-Einstein relation gives $D_\infty = kT/\gamma$, with the fluid drag coefficient $\gamma = 6\pi r\eta$. However, r, T, and η must be estimated. In addition, the particle is not in an infinite fluid medium but is confined between two parallel plates, an effect that increases the drag γ and reduces the measured D.

1. Diffusion in an infinite medium

We first estimate D_∞. The radius is known to $\pm 3\%$.[17] The temperature varies considerably day to day in the laboratory but can be measured to $\pm 1\,^\circ$C by placing a thermistor near the sample during the experiment. The uncertainty in absolute temperature is small (0.3%), but that of water viscosity due to temperature uncertainty is more significant, $\pm 3\%$.[25] Together, these imply a $\pm 5\%$ uncertainty in D_∞. For $r = 100$ nm and $T = 26\,^\circ$C, the nominal value is 2.4 ± 0.1 μm^2/s.

2. Diffusion in a confined medium

The dominant uncertainty in the estimation of γ and D is due to the uncertainty in the absolute sample thickness.

Because the particle diameter is ≈ 210 nm and the nominal plate spacing is ≈ 800 nm, confinement effects are strong. Qualitatively, the extra shear between fluid that pinned to the moving sphere boundary and fluid at the stationary cell plate boundary increase γ and reduce D. Below, we argue that the electrostatic repulsion from the walls implies that the particle is usually near the center of the cell, where the diffusion coefficient D_{mid} is well approximated by the fifth-order expansion,[26]

$$\frac{D_{\text{mid}}}{D_\infty} \approx 1 - 1.004\beta + 0.418\beta^3 + 0.21\beta^4 - 0.169\beta^5, \quad (30)$$

where $\beta = r/d$ is the ratio between the particle radius r and the cell thickness d. Unfortunately, while we use a profilometer to measure cell thickness prior to assembly, we do not have a direct way of measuring d, the thickness of the assembled, filled cell. We estimate the uncertainty in d to be about ± 200 nm ($\pm 25\%$) from the optical images of particles that move in and out of focus in the feedback trap. The depth of focus is ≈ 500 nm, and we can compare particle images to standard out-of-focus images, leading to a range of 600–1000 nm in particular samples and a similar $\pm 25\%$ uncertainty in D.

3. Electrostatic repulsion from the walls

We have assumed that the particle is mostly near the cell's midplane. How good is this approximation? To avoid having particles stick to the cell walls, we ensure that both the glass walls and latex spheres are negatively charged.[17] Electrostatic forces in water are screened by counterions, over a length scale quantified by the Debye length (λ_D). The Debye length in pure, deionized water is 960 nm, but CO_2 in the air and impurities from the sample cell reduce the screening. Behrens and Grier[27] reported the Debye length in "real" water to be 275 nm. Although much reduced, such a length is still significant on the scale of the 800 nm cell and implies that the particle will spend most of its time near the midplane of the cell.

Figure 7 combines our analysis of hydrodynamic drag and electrostatic repulsion in a confined geometry. The gray shaded area shows the screening (Debye) length, from which the particle is effectively excluded. The particle is nearly
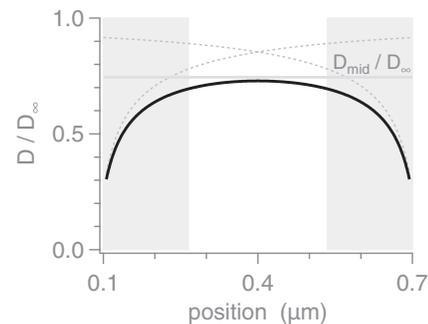

FIG. 7. Effects on the diffusion in confined geometry due to hydrodynamic and electrostatic effect for a cell of thickness $d = 800$ nm. The solid line is based on the superposition of two independent one-wall corrections (dotted gray lines). The solid gray line is the midpoint diffusion approximation. Shaded area is the Debye length λ_D.

always in the central area, where $D \approx D_{\mathrm{mid}}$, with at most a 5% overestimate.

4. Test of diffusion measurements

As a quantitative test of these ideas, we measured D in a cell with nominal parameters $d = 800$ nm, $T = 26\,°$C, $r = 105$ nm. Using 400 s of data in a harmonic trap, we found $D = 1.54 \pm 0.06\ \mu\mathrm{m}^2/\mathrm{s}$, where the uncertainty is dominated by the precision of the length calibration between camera pixels and absolute length. (The statistical error from the fit, $4 \times 10^{-5}\ \mu\mathrm{m}^2/\mathrm{s}$, is negligible.) The experimental measurement is consistent with the mid-plane estimate of $D_{\mathrm{mid}} = 1.8 \pm 0.5\ \mu\mathrm{m}^2/\mathrm{s}$. The uncertainty in the latter estimate is dominated by the systematic error due to the uncertainty in cell thickness. Using a smaller particle or thicker cell would reduce that source of systematic error. Such tests are not possible in the current setup, as smaller particles require faster update times t_s, and thicker cells require either tracking the vertical direction or an imaging system with large depth of focus. Nonetheless, the important point is that the values that we measure for the diffusion constant are consistent with expectations, given the experimental geometry.

VIII. CONCLUSION

We have shown that a RML algorithm allows one to reconstruct accurately the mobility and diffusion constants of particles in a feedback trap. In further work, we have used the procedures developed here to study Landauer's principle, which gives a lower bound to the amount of work needed to erase a bit of information.

It is interesting to compare the present study with the recent work of Wang and Moerner.[4] In the latter work, camera images are replaced by a continuously scanning laser and a single-pixel detector that detects individual photon counts.[3,28] Each individual count leads to an updated estimate of particle position, using a modified Kalman filter. The advantage of such a setup is vastly increased speed and a simplicity that comes from having instantaneous estimates of particle position (no camera exposure effects). The limitations are that interpreting extended objects become difficult. In addition, the presence of background photons leads to non-Gaussian statistics and the need for more sophisticated algorithms (expectation-maximization in Ref. 4 and assumed density filter in a related work[29]). Both of those algorithms are significantly more complicated than the RML algorithm here.

The algorithm given here will simplify even more if we can shorten the camera exposure to make the camera-correction terms not just small but completely negligible. To keep the observation noise at the same level, we would then have to compensate for the shorter exposure by increasing the illumination intensity. Higher intensities can lead to accelerated photobleaching of fluorescent molecules, but non-fluorescent imaging that depends on detecting scattered light would not suffer from such problems. Gold nanoparticles, which show strong plasmonic scattering at wavelengths that depend on their size, are attractive candidates for such studies.

APPENDIX A: EFFECT OF CAMERA EXPOSURE

To see that $\overline{F}_{n-1} \approx F_{n-1}$, we define the deviation $\delta F_{n-1} = \overline{F}_{n-1} - F_{n-1}$ and consider

$$\frac{\langle (\delta F_{n-1})^2 \rangle}{\langle F^2 \rangle} = \left(\frac{t_c}{8 t_s} \right)^2 \frac{\langle (F_n - 2F_{n-1} + F_{n-2})^2 \rangle}{\langle F^2 \rangle}$$

$$= \left(\frac{t_c}{8 t_s} \right)^2 \frac{(6\langle F^2 \rangle - 8\langle F\, F_{-1} \rangle + 2\langle F\, F_{-2} \rangle)}{\langle F^2 \rangle}, \tag{A1}$$

where $\langle F^2 \rangle \equiv \langle F_n^2 \rangle$. We drop the n index because of the homogeneity in time. The $\langle F\, F_{-1} \rangle$ term similarly contains contributions from both $\langle F_n F_{n-1} \rangle$ and $\langle F_{n-1} F_{n-2} \rangle$. Next, we recognize that the forces F_n are correlated on a relaxation time scale t_r that is given by $t_r = \kappa/\gamma$, where γ is the fluid drag and where κ is the local curvature of the potential. That is, near the position x_n, the potential is locally $U(x) \approx \frac{1}{2}\kappa(x - x_n)^2$. If we define, as above, $\alpha = t_s/t_r$, then the overdamped-dynamics correlations are given by

$$\langle F\, F_{-p} \rangle \approx \langle F^2 \rangle e^{-|p|\alpha}. \tag{A2}$$

The expression in Eq. (A1) is then

$$6\langle F^2 \rangle - 8\langle F\, F_{-1} \rangle + 2\langle F\, F_{-2} \rangle$$

$$\approx 6\langle F^2 \rangle - 8\langle F^2 \rangle e^{-\alpha} + 2\langle F^2 \rangle e^{-2\alpha}$$

$$= 4\alpha\langle F^2 \rangle + \mathcal{O}(\alpha^2). \tag{A3}$$

Substituting back into Eq. (A1) then gives

$$\sqrt{\frac{\langle (\delta F_{n-1})^2 \rangle}{\langle F^2 \rangle}} = \left(\frac{\sqrt{\alpha}\, t_c}{4 t_s} \right) \approx 0.06, \tag{A4}$$

for $\alpha = 0.2$ and $t_c/t_s = 0.5$. We note that $\bar{V}_n \approx V_n$, by the same argument.

APPENDIX B: CONTROL PROGRAM

As noted in the main text, the RLM algorithm can diverge, and careful attention to the initial iterations is crucial for convergence. These initial iterations inevitably are strongly coupled to the actual functioning of the control program. The experimental protocol is complicated, because particles may bleach and need to be discarded, because new unwanted particles may diffuse into the field of view, etc. Here, we first give a simplified overview of trap operation and then provide a more detailed account.

1. Overview

Figure 8 shows a simplified flowchart of the control software. The basic structure is that of a *state machine*: at each time step, an image is acquired, the program determines the
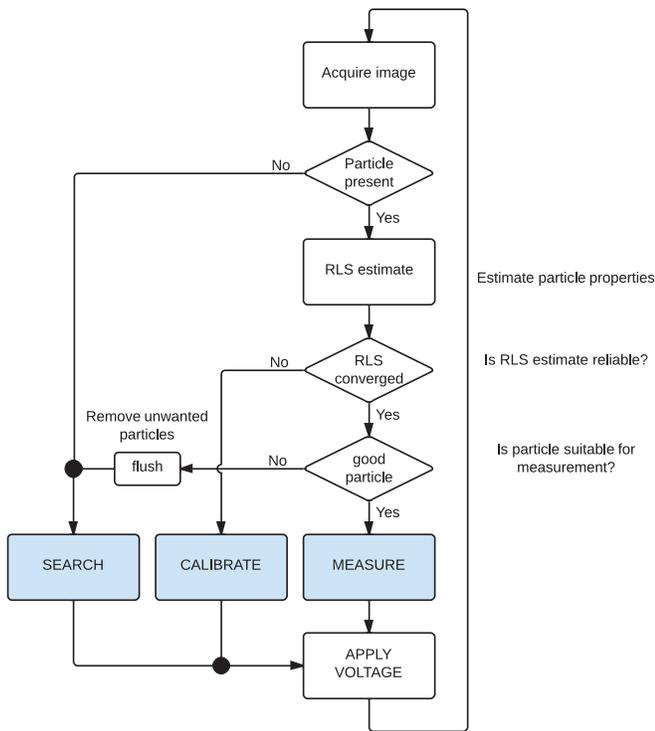
FIG. 8. Simplified flowchart of the control program. The three basic modes of operation are indicated by the shaded blue boxes.

trap state and then responds by selecting a given operational mode. In the simplified version given in Fig. 8, there are three modes:

- *Search*: No particle is present and we apply a large voltage to sweep new particles into the field of view.
- *Calibrate*: A new particle is present and its properties are being calibrated, a process that takes 400 s.
- *Measure*: A calibrated particle is present, and we apply the appropriate virtual potential, updating the calibration to account for drifts. The update uses the RML algorithm described above.

In the simplified Fig. 8, the trap operates mostly in the *Measure* state and follows a loop that starts from "Acquire image" and proceeds downwards to "Apply voltage," before looping back to acquire another image. If there are problems—no good particle is detected (i.e., because it has bleached or because an unwanted particle has diffused into the field of view)—then the system switches to search mode, where it flushes the existing particle and searches for a new one by imposing a large, constant voltage. Once a new particle has been found, the trap enters a calibration mode, to let parameter estimates stabilize, before new measurements can proceed.

2. Details

The description in Appendix B 1 leaves out many details. Figure 9 shows the complete flowchart for trap operation. There are eight possible particle states, and the program responds by operating in one of four modes of operation. The states depend on the number of particles in the field of view

(0 or 1 or ≥ 2) and are further classified into substates, as follows:

- *Single particle*: good | unknown | partially known | bleached | aggregate
- *Two or more particles*: comparable intensities | one is significantly brighter
- *No particle present*.

The modes of operation are *Search, Calibrate, Measure, Flush*.

We also introduce two different timers. The first is denoted by t and measures the time since a new particle was detected. This time is also used for defining the protocols of time-dependent potentials. The second time is denoted by t_{fl} and counts the time since an unwanted particle was detected. It is reset to zero when an unwanted particle is detected and when unwanted particles are flushed. Flushing is accomplished by setting a large voltage (6 V). The mode operates until $t_{fl} = 2$ s. Note that images are acquired every $t_s = 10$ ms, while the trap operates in flush mode, but their content is ignored until the flush is complete.

After the cell is flushed, we maintain the same large DC voltage. Now, however, its purpose is to search for a new particle. We then analyze each acquired image. Based on the summed light intensity in the image, we determine whether a particle is present,[17] which we infer if the intensity is higher than a threshold value ($I > I_{th}$) during two consecutive time steps. We input the observed position of a new particle and the associated applied voltages into the RLS algorithm to estimate the material parameters. We occasionally encounter instability when the initial parameter guess is too far off, especially when they are over estimated. In such cases, the program simply flushes and then reverts to search mode.

When a new particle is detected, we reset the covariance matrix P in Eq. (21) by multiplying all elements by 10^4. We also set the forgetting factor to $\lambda = 0.99$ (or $\tau = 100$). Then, after 20 s, we increase τ to 1000 and continue to update the RLS algorithm. For $100 < t < 200$ s, we use the ordinary RLS algorithm (filtered to decorrelate noise) with fixed $c_{\pm}$ to estimate μ and V_0. During this time, the RLS algorithm converges to a constrained steady state. For $t > 200$ s, we let the noise parameters $c_{\pm}$ vary in the full RLM algorithm. This elaborate initialization procedure keeps the RLM algorithm from diverging.

After the initial RLS convergence ($t = 200$ s), we increase the forgetting factor lifetime to $\tau = 10\,000$ and start the full RLM algorithm. We also start to estimate D and σ, using Eq. (24). The RLM convergence is not sensitive to the value of D; however, the shape of imposed potential is. It takes an additional 200 s for the RLM algorithm and D estimate to fully converge ($t < 400$ s). If the estimated diffusion coefficient is smaller than a threshold value D_{th}, we conclude that the particle is an aggregate and flush it.

After the RLM algorithm has fully converged, we impose the virtual potential and perform the work measurements. At each time step, we check whether a particle has bleached, by examining the laser output power. The laser operates in a proportional-integral feedback loop that attempts to keep the fluorescence intensity constant by altering the input laser
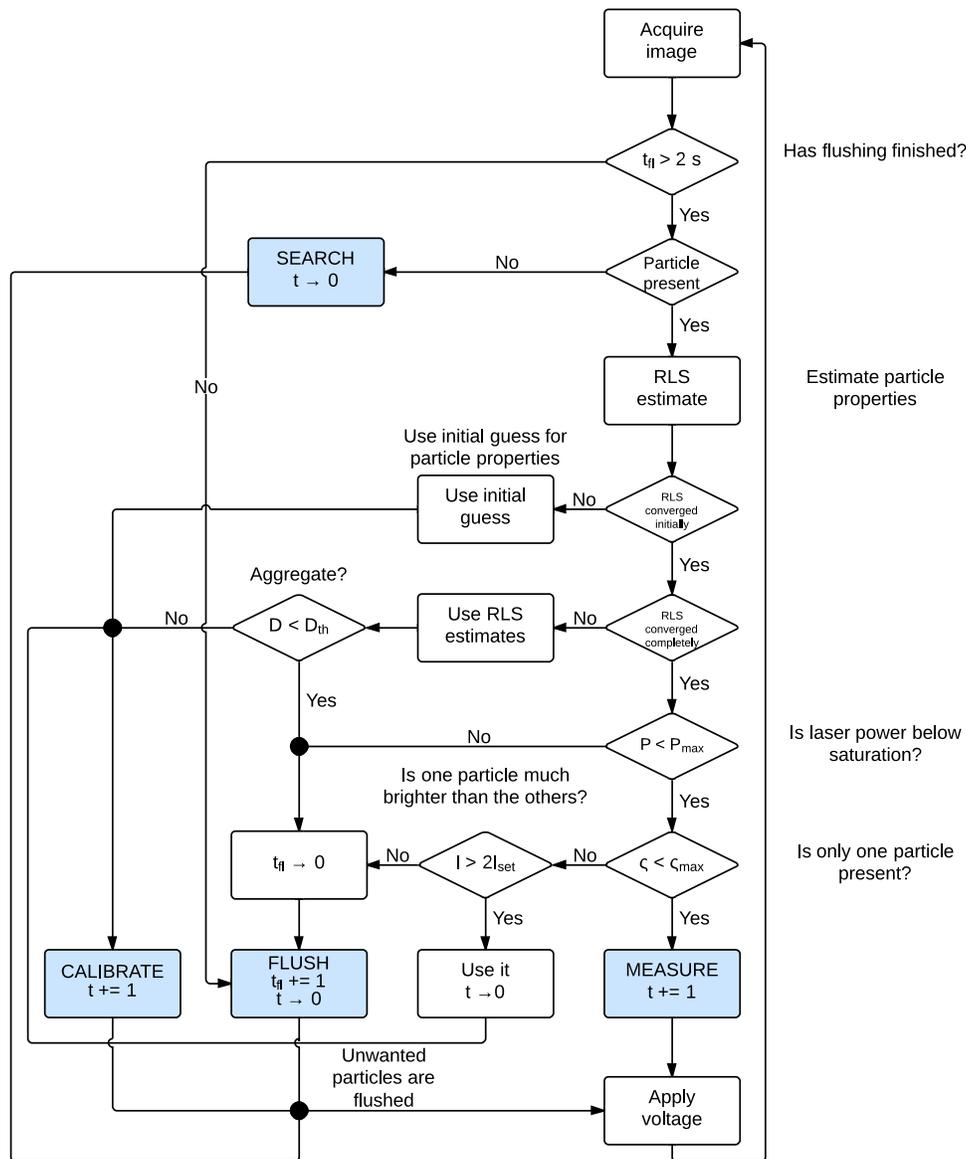
FIG. 9. Full flow chart with automated event identification and detailed event handling.

power, which saturates at a high value when the particle is too dim.[17] If the particle has bleached, we flush it. We also test the noise term ζ_n at each time step. From Eq. (22), this noise term can be interpreted as the difference between the measured displacement $\overline{\Delta x}_n$ and the displacement $t_s \mu(\bar{V}_{n-1} - V_0)$ imposed by the feedback trap. If the difference between observed and imposed displacements is too big ($\zeta_n > \zeta_{max}$), then two or more particles may be present in the observation area. This happens when a new, unwanted particle diffuses into the observation area and then is mis-tracked by image analysis algorithm. The inferred position of such a particle is usually located in between the actual positions of old and new particles, leading to a sudden, large "displacement" ζ_n. In this case, we flush both particles. Sometimes, one particle is significantly brighter than the other. In this case, we keep it and use it to calibrate the system. The dimmer particle is not trapped and quickly leaves the field of view. It also becomes immediately less visible after the AOD control algorithm reduces the laser intensity.

Each of the four possible modes of operation generates two voltages at each time step, which are sent to the two electrode pairs to create the desired displacements.

[1]A. E. Cohen and W. E. Moerner, App. Phys. Lett. **86**, 093109 (2005).

[2]J. Bechhoefer, Rev. Mod. Phys. **77**, 783 (2005).

[3]A. P. Fields and A. E. Cohen, Proc. Natl. Acad. Sci. U.S.A. **108**, 8937 (2011).

[4]Q. Wang and W. E. Moerner, Nat. Meth. **11**, 555 (2014).

[5]A. E. Cohen and W. E. Moerner, Proc. Natl. Acad. Sci. U.S.A. **103**, 4362 (2006).

[6]A. E. Cohen and W. E. Moerner, Phys. Rev. Lett. **98**, 116001 (2007).

[7]A. E. Cohen and W. E. Moerner, Proc. Natl. Acad. Sci. U.S.A. **104**, 12622 (2007).

[8]R. H. Goldsmith and W. E. Moerner, Nat. Chem. **2**, 179 (2010).

[9]Q. Wang, R. H. Goldsmith, Y. Jiang, S. D. Bockenhauer, and W. Moerner, Acc. Chem. Res. **45**, 1955 (2012).

[10]Q. Wang and W. E. Moerner, J. Phys. Chem. B **117**, 4641 (2013).

[11]G. S. Schlau-Cohen, Q. Wang, J. Southall, R. J. Cogdell, and W. E. Moerner, Proc. Natl. Acad. Sci. U.S.A. **110**, 10899 (2013).

[12]A. E. Cohen, Phys. Rev. Lett. **94**, 118102 (2005).

[13]A. Cho, Science **332**, 171 (2011).

[14]R. Landauer, IBM J. Res. Develop. **5**, 183 (1961).

[15]A. Bérut, A. Arakelyan, A. Petrosyan, S. Ciliberto, R. Dillenschneider, and E. Lutz, Nature (London) **483**, 187 (2012).

[16]Y. Jun, M. Gavrilov, and J. Bechhoefer, "High-precision test of Landauer's principle in a feedback trap," e-print arXiv:1408.5089.

[17]M. Gavrilov, Y. Jun, and J. Bechhoefer, Proc. SPIE **8810**, 881012 (2013).

[18]Y. Jun and J. Bechhoefer, Phys. Rev. E **86**, 061106 (2012).

[19]M. Goulian and S. M. Simon, Biophys. J. **79**, 2188 (2000).

[20]T. Savin and P. S. Doyle, Biophys. J. **88**, 623 (2005).

[21]A. E. Cohen, "Trapping and manipulating single molecules in solution," Ph.D. thesis (Stanford University, 2006).

[22]K. J. Åström and B. Wittenmark, *Adaptive Control*, 2nd ed. (Dover, 2008).

[23]I. D. Landau and G. Zito, *Digital Control Systems: Design, Identification and Implementation* (Springer, 2006).

[24]C. L. Vestergaard, P. C. Blainey, and H. Flyvbjerg, Phys. Rev. E **89**, 022726 (2014).

[25]W. Haynes, *CRC Handbook of Chemistry and Physics*, 93rd ed. (Taylor & Francis, 2012).

[26]J. Happel and H. Brenner, *Low Reynolds Number Hydrodynamics: With Special Applications to Particulate Media* (Springer Netherlands, 1983).

[27]S. H. Behrens and D. G. Grier, J. Chem. Phys. **115**, 6716 (2001).

[28]Y. Jian, Q. Wang, A. E. Cohen, N. Douglas, J. Frydman, and W. E. Moerner, Proc. SPIE **7038**, 703807 (2008).

[29]A. P. Fields and A. E. Cohen, Opt. Exp. **20**, 22585 (2012).