



**Extracting forces from noisy dynamics in dusty plasmas**Wentao Yu,<sup>\*</sup> Jonathan Cho, and Justin C. Burton *Department of Physics, Emory University, Atlanta, Georgia 30322, USA* (Received 19 January 2022; revised 5 July 2022; accepted 26 August 2022; published 9 September 2022)

Extracting environmental forces from noisy data is a common yet challenging task in complex physical systems. Machine learning (ML) represents a robust approach to this problem, yet is mostly tested on simulated data with known parameters. Here we use supervised ML to extract the electrostatic, dissipative, and stochastic forces acting on micron-sized charged particles levitated in an argon plasma (dusty plasma). By tracking the subpixel motion of particles in subsequent images, we successfully estimated these forces from their random motion. The experiments contained important sources of non-Gaussian noise, such as drift and pixel locking, representing a data mismatch from methods used to analyze simulated data with purely Gaussian noise. Our model was trained on simulated particle trajectories that included all of these artifacts, and used more than 100 dynamical and statistical features, resulting in a prediction with 50% better accuracy than conventional methods. Finally, in systems with two interacting particles, the model provided noncontact measurements of the particle charge and Debye length in the plasma environment.

DOI: [10.1103/PhysRevE.106.035303](https://doi.org/10.1103/PhysRevE.106.035303)**I. INTRODUCTION**

Huge amounts of experimental data are often collected faster than can be interpreted. In complex physical or biological systems, this data mostly comes in the form of tracked positions of individual agents or particles, yet random noise makes the inference of internal and external forces challenging. Conventional statistical methods often result in systematic error, and require special treatment of error estimation [1,2]. Machine learning (ML) algorithms can infer forces from trajectories without systematic error [3–9], but their reported performance has been restricted to labeled simulated data rather than unlabeled experimental data. Another restriction is data mismatch [10,11]. Data is almost always simulated with Gaussian noise, which is presumed in many inference algorithms, but experimental data may include non-Gaussian noise and other artifacts such as systematic drift. Subsequently, modern inference algorithms should be benchmarked using experimental data where parameter estimates can be verified by alternative, independent methods.

Dusty plasmas, where micron-sized charged particles are suspended in a low-density gaseous plasma, provide opportune experimental data for dynamical inference methods. The particles experience a wide array of forces including electrostatic repulsion, velocity-dependent drag from neutral and charged ions, and stochastic thermal kicks [12]. As a result, dusty plasmas display a wide range of complex, nonequilibrium dynamical phenomena, including superthermal excitations from nonreciprocal forces [13–15], oscillations between “turbulent” and “quiescent” states [16–18], parametric resonance and kinetic heating [19–21], spontaneous oscillations at low pressures [22,23], helical dust “strings” [24,25], and

vortical structure formation at high magnetic fields [26–28]. However, the individual interactions between particles are a subject of active research [29–31], and the external forces acting on a single particle can be complex [22,32–34].

ML has already been applied to a few distinct areas of dusty plasma research. Examples include the interpretation of Langmuir probe and electron density measurements [35–37], and the prediction of particle generation and annihilation in fusion devices [38]. Additionally, ML has been used to identify phase boundaries in dense dust systems [39], and to assist with stereoscopic tracking of many particles in three dimensions [40]. Bayesian analysis and ML have also been applied to investigate the nonlinear dynamics of single dust particles [41]. Importantly, these dynamics provide information about the dust charge, interaction potential, and external fields, essentially acting as a noncontact probe of the system. Both the dust charge ( $Q$ ) and Debye screening length ( $\lambda$ ) between interacting dust particles can be estimated by an analysis of the noisy dynamics [30]; two parameters which are often difficult to accurately measure.

Here we show how the undisturbed, random motion of one and two particles in a dusty plasma can be interpreted using ML to provide accurate information about their interparticle and environmental forces. Crucially, our ML methods are trained with simulations that consider real experimental artifacts such as anisotropic confinement, nonconservative forces, stochastic Levy noise, non-Gaussian tracking error (pixel locking), and experimental drift. These artifacts can be observed in a statistical analysis of the data, yet they rarely included in dynamical inference procedures, leading to data mismatch. In our procedure, features are extracted from the simulated trajectories to train supervised ML models. The models simultaneously predict system-wide parameters with 50% better accuracy than traditional methods such as Fourier

<sup>\*</sup>wentao.yu@emory.edu

spectrum and maximum likelihood estimation in simulated trajectories.

In the experiments, one key feature is that many of the parameters are independently inferred by analyzing the particles' recovery to equilibrium after a perturbation, thus labeling the data and verifying the model's performance on experimental time series. Based on labeling with this alternative method, our prediction on experimental data is evaluated to have the same accuracy as simulated data, alleviating data mismatch. Furthermore, in experiments with two particles, we provide an accurate estimation of  $Q$  and  $\lambda$  solely from the particles' pixel-scale Brownian motion without knowledge of other system-wide parameters, such as Epstein damping. These results will help guide other studies that use ML to quantitatively infer system parameters in real-world, noisy experimental data.

The rest of this paper will be organized as follows. In Sec. II, we detail our experimental setup and three-dimensional (3D) particle imaging and tracking methods. In Sec. III A, we introduce the linearized single-particle model used for our simulations. Section III B explains the dominant source of errors in our experiments. Their mismatch from noncorrelated Gaussian noise that is commonly used in simulations is observed by statistical analysis. We then explain how we handle these errors in our simulations. In Sec. III C, we describe the features extracted from simulated and experimental data. These features are used by our ML models. Section III D explains the different ML models and their corresponding performance on simulated test data compared with conventional methods. Section III E describes the alternative way that we label our experimental data. Finally, Sec. III F demonstrates the performance of our ML models on experimental, single-particle data. Section IV expands our methods to systems of two particles. In Sec. IV A, we introduce the changes to the linearized model for two particles. Sections IV B and IV C explain the simulation details and the features used for the ML models. Lastly, Sec. IV D shows our predictions on experimental two-particle data, including an inference of the particle charge and Debye length.

## II. EXPERIMENTAL METHODS AND PARTICLE TRACKING

Our experiments used melamine-formaldehyde particles with diameters  $9.46 \pm 0.10 \mu\text{m}$  and  $12.8 \pm 0.3 \mu\text{m}$  (microParticles GmbH). The particles were electrostatically levitated in a low-pressure argon plasma above an aluminum electrode with diameter 150 mm [Fig. 1(a)], similar to previous experiments [16,22]. The argon plasma was generated by a 13.56 MHz radio-frequency voltage applied to the electrode, resulting in  $2.9 \pm 0.1 \text{ W}$  of input power and a fixed dc bias voltage of  $-36.3 \pm 1.2 \text{ V}$ . An aluminum ring was placed on the edge of the electrode to provide horizontal confinement. The plasma pressure,  $P$ , was varied between 0.6 and 1.3 Pa. Under these conditions, the typical electron temperature in the plasma was 1.3–1.5 eV [22].

To visualize the particles levitated in the plasma environment, a horizontal laser sheet was generated by focusing with a cylindrical lens, similar to previous experiments [16,17,22]. The scattered light from the particles is captured from above

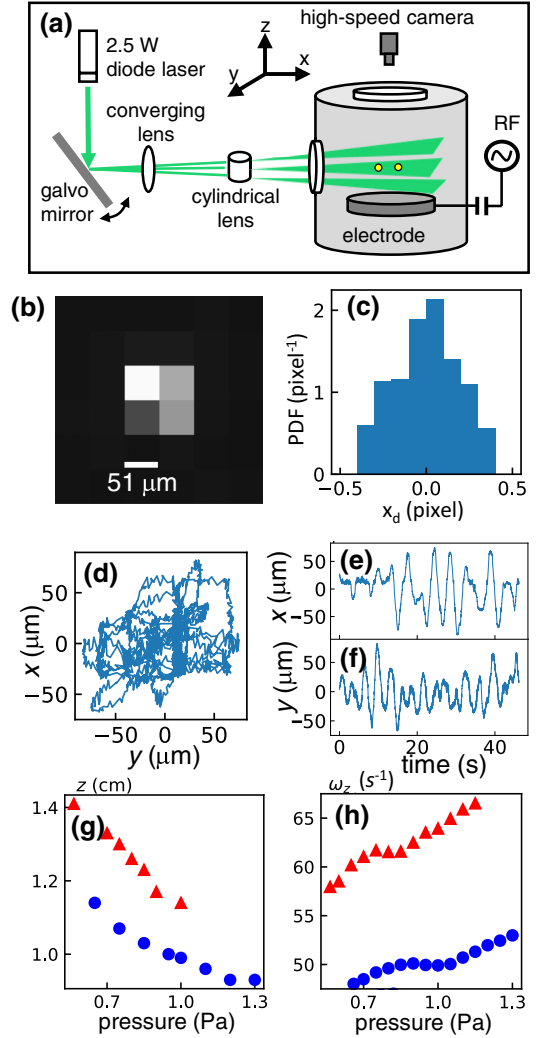


FIG. 1. (a) Experimental setup for the 3D tomographic imaging and particle tracking. The oscillating mirror varies the angle of the incoming laser (wavelength 532 nm). The converging lens focuses the beam in  $z$ , and the cylindrical lens expands the beam in the  $xy$  plane. Particles are imaged and tracked from above as described in the text. (b) Image of the scattered light from a single particle with diameter  $12.8 \mu\text{m}$ . (c) Probability distribution of the decimal part of tracked positions, prior to SPIFF correction. (d) A 45-s trajectory for a single  $12.8\text{-}\mu\text{m}$  particle undergoing stochastic motion. (e),(f) Time series of the  $x$  and  $y$  position corresponding to the same trajectory. (g)  $z$  position as a function of pressure for  $12.8\text{-}\mu\text{m}$  (blue circles) and  $9.46\text{-}\mu\text{m}$  (red triangles) particles. (h) Dominant frequency of motion in the  $z$  direction for both sizes of particles, obtained by Fourier transform.

by a Phantom V7.11 high-speed camera equipped with a macro lens. This allowed tracking of the particle positions in the horizontal,  $xy$  plane. Additionally, we used a mirror attached to a galvo motor to oscillate the laser sheet with a 50-Hz sawtooth wave at an amplitude of a few millimeters. The time base of the camera was synchronized to the function generator driving the galvo, and the camera recorded at 1000 Hz [Fig. 1(a)], resulting in 20 images at different vertical positions per oscillation of the laser sheet. With this tomographic 3D tracking, we simultaneously obtained in-

formation about the vertical and horizontal motion of the particles.

The spatial resolution of our imaging system was  $51 \mu\text{m}$  per pixel in the  $xy$  plane, and  $200 \mu\text{m}$  between image slices in the  $z$  direction. However, by tracking the 3D particle motion using an open source software (TRACKPY [42]), the position of the particles can be located with much better accuracy. The image representing the scattered light from a single particle is shown in Fig. 1(b). The centroid of the particle “blob” is found by calculating the center of mass of the pixels, where the pixel brightness represents the mass contribution of a single pixel [42]. The same centroid procedure is done with image slices in the  $z$  direction. A probability density function of the decimal part of the tracked positions ( $x_d$ ) is shown in Fig. 1(c), showing a strong bias towards integer values. This bias is known as pixel locking [43]. Using the single-pixel interior filling function (SPIFF) algorithm [44,45], these errors can be statistically corrected from the tracked data. Ultimately, our estimated subvoxel resolution in tracking the particles was  $\approx 4 \mu\text{m}$  in the  $xy$  plane, and  $\approx 16 \mu\text{m}$  in  $z$ .

This subpixel error was confirmed using an independent procedure. We created digital movies of bright “particles” moving unidirectionally across a projection screen. The screen was imaged with our camera so that the particles appeared similar in size on the camera sensor when compared to the experiments [i.e., Fig. 1(b)]. Since the trajectory of the particles was predetermined in the movie, we compared the tracked positions to the known values. Despite these procedures, the horizontal resolution was still a significant fraction of the amplitude of the Brownian motion in the experiments. This was evidenced by systematic statistical effects in the analysis of the dynamics, and will be discussed in Sec. III B.

### III. SINGLE-PARTICLE MOTION

#### A. The linearized model

A typical  $xy$  trajectory for a single, isolated particle is shown in Fig. 1(d). The  $x$  and  $y$  time series corresponding to this trajectory is shown in Figs. 1(e) and 1(f). Without any external perturbations, the particle experienced thermally excited motion in three dimensions. The amplitude of the motion was  $\approx 50 \mu\text{m}$  in the  $xy$  plane. A dominant angular frequency of motion ( $\omega \approx 1\text{--}2 \text{ Hz}$ ) is clearly visible in the time series. The amplitude of motion in  $z$  was much smaller; less than our spatial resolution. Nevertheless, we measured the  $z$  position of the particle as a function of gas pressure [Fig. 1(g)], which increased at lower pressure as the electrode’s sheath expanded. Also, by Fourier transforming the time series of the  $z$  position, we estimated the vertical frequency of oscillation [ $\omega_z$ , Fig. 1(h)], which was much larger than the horizontal frequency, indicating strong confinement in the  $z$  direction.

Due to the small amplitude of motion, to lowest order, the particles behaved as stochastic harmonic oscillators. Since the amplitude of motion in  $z$  was much smaller due to the strong confinement, we will ignore motion in the  $z$  direction for our linearized model. Let  $\vec{\mathbf{r}}(t) = x(t)\vec{\mathbf{e}}_x + y(t)\vec{\mathbf{e}}_y$ , denote the two-dimensional (2D) position of a particle at time  $t$ , and dotted

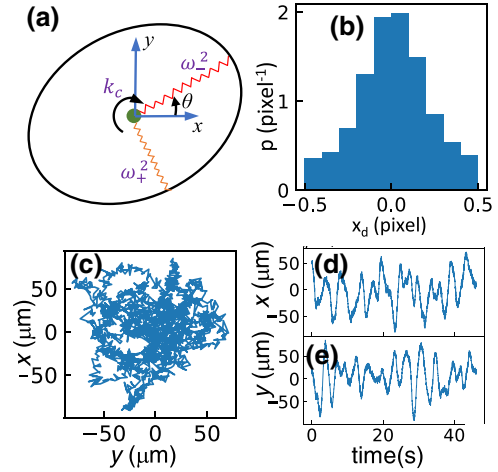


FIG. 2. (a) Linearized external forces on the particle, as described in the text. The black circle represents an equipotential surface of the harmonic confinement. The eccentricity is exaggerated. (b) Probability distribution of the decimal part of positions after simulated pixel locking. (c) A simulated trajectory of length 45 s for a single particle undergoing stochastic motion. (d),(e) Time series of the  $x$  and  $y$  position corresponding to the trajectory shown in (b).

variables refer to time derivatives. The linearized dynamics of one particle reads

$$\ddot{\vec{\mathbf{r}}} = -\vec{\nabla}\phi + \vec{\nabla} \times \vec{\mathbf{A}} - \gamma\dot{\vec{\mathbf{r}}} + \vec{\mathbf{N}}(\alpha), \quad (1)$$

$$\phi = \frac{\omega^2}{2}[(1 - \delta)(x \cos \theta + y \sin \theta)^2 + (1 + \delta)(-x \sin \theta + y \cos \theta)^2], \quad (2)$$

$$\vec{\mathbf{A}} = \frac{k_c}{2}(x^2 + y^2)\vec{\mathbf{e}}_z. \quad (3)$$

This model contains six parameters,  $\gamma$ ,  $\omega$ ,  $\delta$ ,  $\theta$ ,  $k_c$ , and  $\alpha$ . The deterministic confinement force has two components. The conservative potential  $\phi$  resembles a 2D spring characterized by three parameters: the average frequency  $\omega$ , the asymmetry  $\delta$  between two principal axes, and the angle  $\theta$  from the  $x$  axis to the weaker principal axis. Two eigenfrequencies,  $\omega_- = \omega\sqrt{1 - \delta}$  and  $\omega_+ = \omega\sqrt{1 + \delta}$ , and  $\theta$  are displayed in Fig. 2(a). The nonconservative vector potential  $\vec{\mathbf{A}}$ , characterized by a parameter  $k_c$ , represents a rotational force possibly due to drag from the background ion flow [46], particle asymmetries [33], or magnetic fields [47].

Additionally, the particle experiences drag from the background neutral gas. According to Epstein’s law assuming diffuse reflection from neutral gas collisions on the particle surface, the damping coefficient can be expressed as [22,34]

$$\gamma = 1.44 \frac{P}{a_p \rho_p} \sqrt{\frac{2m_{ar}}{\pi k_B T}}. \quad (4)$$

Here,  $P$  is the gas pressure,  $a_p$  is the particle radius,  $\rho_p = 1510 \text{ kg/m}^3$  is the mass density of the particle,  $m_{ar}$  is the mass of an argon atom,  $k_B$  is Boltzmann constant, and  $T = 300 \text{ K}$  is room temperature. An important assumption here is that the particle size is smaller than the mean free path ( $\approx 5 \text{ mm}$  at

TABLE I. The parameters for one-particle simulation.

Name	Description	Range	Drift amp.	Drift time
$\omega$	Confinement freq.	$(1.3, 2.5) \text{ s}^{-1}$	None	None
$\gamma$	Damping coeff.	$(0.4, 1.7) \text{ s}^{-1}$	None	None
$\delta$	Asymmetry	$(0, 0.35)$	None	None
$\theta$	Weak axis	$(-\pi/2, \pi/2)$	None	None
$k_c$	Vortex force coeff.	$(-k_{c,\max}, k_{c,\max}) \text{ s}^{-2}$	None	None
$\alpha$	Noise distribution	$(1.6, 2.0)$	None	None
$x_0$	Equi. position	0	$(0, 0.5)$	$(12, 800) \text{ s}$
$y_0$	Equi. position	0	$(0, 0.5)$	$(12, 800) \text{ s}$
$l_{\text{pixel}}$	Simulated pixel width	$(0.3, 1)$	None	None

$P = 1 \text{ Pa}$ ). For the particles with diameter  $2a_p = 12.8 \text{ }\mu\text{m}$ ,  $\gamma/P = 0.95 \text{ Pa}^{-1} \text{ s}^{-1}$ . For the particles with diameter  $2a_p = 9.46 \text{ }\mu\text{m}$ ,  $\gamma/P = 1.29 \text{ Pa}^{-1} \text{ s}^{-1}$ .

### B. Handling random noise, parameter drift, and measurement error in simulation

The last term in Eq. (1) is a stochastic acceleration,  $N(\alpha)$ , which follows a stable Levy distribution. The parameter  $\alpha$  will be determined by inference. We do not assume *a priori* that the stochastic motion is purely Brownian ( $\alpha = 2$ ), and  $\alpha < 2$  indicates a more heavy-tailed distribution. The Brownian motion of particles in experiments is driven primarily by random impulses from the environment. Thus, in the simulations, temporally independent random noise is added to the *acceleration* of the particles at each time step. The Levy-stable noise was generated by the python function `scipy.stats.levy_stable` with parameter skewness  $\beta = 0$  and center  $\mu = 0$ . The noise scale  $c$  and the parameter for heavy tailness  $\alpha \in (1.6, 2.0)$  was independently chosen for each simulation.

For all simulations, we used a time step  $\Delta t = 0.02 \text{ s}$  to closely follow the experiments. The parameters  $\omega$ ,  $\gamma$ ,  $\delta$ ,  $\theta$ ,  $k_c$ , and  $\alpha$  are randomly chosen from a uniform distribution prior to each simulation. The range of values possible for each parameter are listed in Table I. For the maximum values of  $k_c$ , we chose  $k_{c,\max} = \min(1 \text{ s}^{-2}, 0.9 \times \sqrt{\omega^4 \delta^2 + \omega^2 \gamma^2})$ , which guaranteed that the confinement force was able to provide the necessary centripetal acceleration to keep the particle in a bounded stable orbit. Since none of the parameters have a length scale in their units, the simulated Levy noise scale was arbitrarily set to  $c = \sqrt{\gamma \omega^2 / \Delta t}$ .

Drift was inevitably present in nearly all experiments. This was most noticeable in the drift of the equilibrium position of the particle. The drift was small, less than 1 pixel, but is still comparable to the amplitude of the Brownian motion. We modeled this in simulations as a temporally correlated Gaussian noise added to the *equilibrium positions*. To simulate a time series of temporally correlated noise,  $a_i$ , with standard deviation (STD)  $\sigma$  and characteristic correlation time  $\tau$  much larger than simulation time step,  $\tau \gg \Delta t = 0.02 \text{ s}$ , we used the recursive relation:

$$a_0 = N_0, \\ a_i = \left(1 - \frac{\Delta t}{\tau}\right) a_{i-1} + \frac{\Delta t}{\tau} N_i. \quad (5)$$

Here  $N_i$  is an array of independent and identically distributed Gaussian random numbers with zero mean and unit variance. The final sequence is adjusted by subtracting the mean from each element in the series, and then normalizing the STD to be  $\sigma$ . During simulation, the equilibrium positions ( $x_0$  and  $y_0$ ) both drift with the same timescale  $\tau \in (12, 800) \text{ s}$  and potentially different amplitudes  $\sigma \in (0, 0.5)$ , randomly chosen for each simulation [Eq. (5)].

As discussed in Sec. II, pixel locking was an important source of noise in experimental data. Thus, measurement errors were added to the simulated position time series after all the time steps were *completed*. This was intended to simulate errors associated with tracking the particles in the images. To simulate pixel locking in the position time series, we converted the simulated position to pixels using a length scale  $l_{\text{pixel}}$  and a random offset  $x_{\text{pixel}} \in (-0.5, 0.5)$ . Then we applied a transformation to the decimal portion,  $x_d \in (-0.5, 0.5)$ , of the pixel:

$$x_d = \frac{x + x_{\text{pixel}}}{l_{\text{pixel}}} - \text{round}\left(\frac{x + x_{\text{pixel}}}{l_{\text{pixel}}}\right), \quad (6)$$

$$x_d^* = \text{sgn}(x_d) \frac{|2x_d|^{p_d}}{2} + N_t(\sigma_d). \quad (7)$$

Here  $x_d^*$  is the transformed pixel value,  $p_d$  is an exponent randomly chosen between (1,4) for each time series, and  $N_t$  is a Gaussian noise with zero mean and STD  $\sigma_d \in (0, 0.1)$ . The distribution of the decimal part of simulated ‘‘pixel’’  $x_d$  is plotted in Fig. 2(b). Finally, as in the experiments, we used the SPIFF algorithm [45] on  $x_d^*$  to correct simulated data before training the model. An example of a simulated trajectory and its  $x$  and  $y$  components are shown in Figs. 2(c) and 2(d) and 2(e), respectively.

Though pixel locking was a small source of error, it led to large systematic errors in dynamical quantities such as the 1D velocity distribution. Without further modeling the effects of pixel locking, these errors can be easily mistaken for stochastic noise with  $\alpha < 2$ . In a stochastic underdamped harmonic oscillator simulated with  $\alpha = 2$  and Gaussian measurement error, the 1D velocity distribution was well fit by the form  $\log P(v_x) = A v_x^p$  with  $p = 2$  [Fig. 3(a), red circles]. However, a simulation with either a smaller value of  $\alpha$  (green squares), or pixel locking (yellow triangles) both led to a significantly smaller value of the fitted parameter  $p$ . Experimentally, the fitted  $p$  was usually smaller than 2 [Fig. 3(b)]. It is possible to minimize pixel-locking errors in the velocity distribution

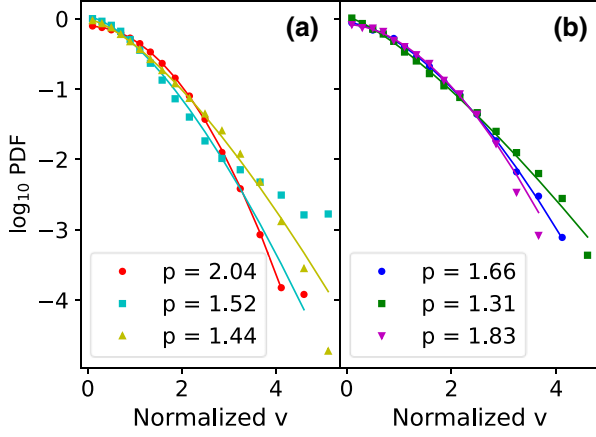


FIG. 3. Probability density function (PDF) of the  $x$  component of the velocity,  $|v_x|$ , normalized by  $\sqrt{\langle v_x^2 \rangle}$ , where the average is over time. (a) The velocity distribution of simulated data. Red circles represent  $\alpha = 2$  (Gaussian noise). Green squares represent  $\alpha = 1.8$  (non-Gaussian noise). Yellow triangles represent  $\alpha = 2$ , but with simulated pixel locking and SPIFF correction. The solid lines are fits to the form  $y = Av^p$  with more weight attached to the left side of the curve (see Sec. III C, part 4). The fitted value of the exponent  $p$  is shown in the inset. (b) Three different velocity distributions for experimental trajectories and the associated fits with exponent  $p$ . All six trajectories in the curve undergo a same low-pass filter with a 4-Hz cutoff.

function by defocusing the camera [30,43]; however, our 3D imaging and tracking methodology required particles with significant brightness due to the low exposure time. Subsequently, it was not possible to determine whether  $\alpha < 2$  or pixel locking leads to non-Gaussian distributions with our current analysis.

### C. Features for ML

The data used to train the ML model consisted of a simulated time series of both the  $x$  and  $y$  motion of the particle. Typically, each time series contained 15 000–100 000 elements, depending on the total length of time of the motion. Although in principle it is possible to use the raw data as input to the ML model, this would drastically increase the computation time. Thus, we choose to extract hundreds of relevant dynamical features of the motion in order to train the model. These ranged from Fourier transforms and autocorrelations, to more sophisticated inference algorithms such as underdamped Langevin inference (ULI) [2].

The confining potential for the particles consists of an asymmetric harmonic trap in  $x$  and  $y$ , as shown in Fig. 2(a). We first extracted a rough estimation of the principal axes, defined by  $\theta$ , for a 2D time series  $[x_t, y_t]$ ,  $t = 0, 1, 2, \dots, T$ .  $T$  is the length of a single time series, and is the first feature. The total time duration of the series is  $T \times \Delta t$ , where  $\Delta t = 0.02$  s. In polar coordinates,  $\phi_t = \arctan \frac{y_t}{x_t}$ . We used 20 bins to form a histogram of  $\phi_t$  between  $(-\pi/2, \pi/2)$  and fit the probability density with

$$p(\phi) = \frac{1 + \delta_{\text{hist}} \cos 2(\phi - \theta_{\text{hist}})}{\pi}. \quad (8)$$

Here  $\delta_{\text{hist}}$  and  $\theta_{\text{hist}}$  are two features.

Let  $\langle p_i, q_i \rangle = \sum_i^T p_i q_i / T$ . The correlation matrix  $\mathbf{C}$  was computed:

$$\mathbf{C} = \begin{bmatrix} \langle x, x \rangle & \langle x, y \rangle \\ \langle x, y \rangle & \langle y, y \rangle \end{bmatrix}. \quad (9)$$

The eigenvalues of the matrix are  $\lambda(1 - \delta_{\text{eig}})$  and  $\lambda(1 + \delta_{\text{eig}})$  and their eigenvectors are  $(\cos \theta_{\text{eig}}, \sin \theta_{\text{eig}})$  and  $(-\sin \theta_{\text{eig}}, \cos \theta_{\text{eig}})$ . Here  $\lambda$ ,  $\delta_{\text{eig}}$ , and  $\theta_{\text{eig}}$  are three features. After calculating the eigenvectors,  $(x_t, y_t)$  are projected onto the (estimated) weaker and stronger principal axes for further analysis:

$$\begin{aligned} w_t &= x_t \cos \theta_{\text{eig}} - y_t \sin \theta_{\text{eig}}, \\ s_t &= x_t \sin \theta_{\text{eig}} + y_t \cos \theta_{\text{eig}}. \end{aligned} \quad (10)$$

$w_t$  and  $s_t$  are then normalized into unit STD, and the following feature extraction algorithms are applied to  $(w_t, s_t)$ :

(1) Fourier spectrum. This is the most commonly used tool to analyze the motion of a 1D harmonic oscillator,  $\xi_t$ . We compute the Fourier spectrum and only analyzed data between  $0.5 \text{ s}^{-1} < \omega < 4 \text{ s}^{-1}$ . This is fitted to analytical prediction for a 1D stochastic harmonic oscillator:

$$\omega I(\omega) = A_{\text{FT}} \left[ \omega^2 \left( 1 - \frac{\omega_{\text{FT}}^2}{\omega^2} \right)^2 + \gamma_{\text{FT}}^2 \right]^{-1/2}, \quad (11)$$

where  $A$ ,  $\omega_{\text{FT}}$ , and  $\gamma_{\text{FT}}$  are fitting parameters. Although  $\theta_{\text{eig}}$  is a good estimate of the principal axes, we performed Fourier analysis on a combination of  $w_t$  and  $s_t$ :  $\xi_t = w_t \cos \phi + s_t \sin \phi$ . The following pseudocode describes the procedure:

for  $\phi = [-\pi/4, 0, \pi/4, \pi/2]$ :  
 $\xi_t = w_t \cos \phi + s_t \sin \phi$ .  
 Conduct 1D Fourier spectrum on  $\xi_t$ .  
 Fit the spectrum using Eq. (11).  
 $A_{\text{FT}, \phi}$ ,  $\omega_{\text{FT}, \phi}$ , and  $\gamma_{\text{FT}, \phi}$  are features.

Altogether 12 features are extracted using the Fourier spectrum.

(2) Autocorrelation is another analysis technique used on 1D time series,  $\xi_t$ , and is defined as

$$A(\tau) = \sum_{t=0}^{T-\tau} \xi_t \xi_{t+\tau} / (T - \tau). \quad (12)$$

$A(\tau)$  was computed for  $\xi_t$  and fitted to the analytic form for a 1D stochastic harmonic oscillator:

$$A(\tau) = \left( 1 + \frac{\gamma_A^2}{\omega_A^2} \right) e^{-\gamma_A \tau} \cos \left( \omega_A \tau - \arctan \frac{\gamma_A}{\omega_A} \right). \quad (13)$$

Similar as the Fourier spectrum, the following loop is performed to extract features.

for  $\phi$  in  $[-\pi/4, 0, \pi/4, \pi/2]$ :  
 $\xi = w \cos \phi + s \sin \phi$ .  
 Normalize  $\xi$  into zero mean and unit variance. Calculate the autocorrelation by Eq. (12).  
 Fit the autocorrelation using Eq. (13).  
 $\omega_{A, \phi}$  and  $\gamma_{A, \phi}$  are features.

Altogether eight features are extracted using autocorrelation.

(3) Percentiles and equipartition law. Let  $P(\xi, p)$  indicate the  $p$  percentile of a 1D time series  $\xi_t$ ; the quantity  $\zeta = \frac{P(\xi, 1) - P(\xi, 99)}{P(\xi, 30) - P(\xi, 70)}$  contains qualitative information about the heavy

tailness of the distribution of the stochastic noise that drives  $\xi$ . Furthermore, according to equipartition, the time-averaged kinetic and potential energies should be equal. As a result,

$$\omega_{\text{ep}}^2 = \frac{\sum_{t=1}^{T-1} \xi_t'^2}{\sum_{t=1}^{T-1} \xi_t^2} \quad (14)$$

is a rough estimation of the eigenfrequency if  $\bar{\mathbf{e}}_\xi$  is a principal axis for the confinement, where  $\xi_t' = \frac{\xi_{t+1} - \xi_{t-1}}{2\Delta t}$  and  $\Delta t = 0.02$  s. To extract features, the following loop is performed.

for  $\phi$  in  $[-2\pi/3, -\pi/3, 0, \pi/3, 2\pi/3, \pi/2]$ :  
 $\xi = w \cos \phi + s \sin \phi$ .

Calculate  $\zeta_\phi$  and  $\omega_{\text{ep},\phi}$  as features.

Altogether 12 features are extracted.

(4) Velocity distribution. For a 1D time series  $\xi_t$ , the central difference velocity is calculated,  $\xi_t'$ . Then we compute the probability distribution  $P(\xi_n')$  of  $\xi_n' = |\xi_t'|/\sqrt{\langle \xi_t'^2 \rangle}$ , as done in Fig. 3. Were the noise purely Gaussian ( $\alpha = 2$ ) with no measurement error, then  $\log[P(\xi_n')] \propto -\xi_n'^2$ . A more heavy-tailed distribution (see Fig. 3) may indicate  $\alpha < 2$  or pixel-locking measurement error. Since the distribution is rather complicated, three different fits are performed to extract features. The first is a fit of  $\log P$  versus  $\xi_n'$ :

$$\log P(\xi_n') = A_0(\xi_n')^{p_0}, \quad (15)$$

where a fitting weight,  $e^{\log P/2}$ , is applied to attach more importance to the beginning of the curve.  $A_0$  and  $p_0$  are fitting parameters. The second fit linearly fits the  $\log P$  vs  $\xi_n'$  curve with  $\xi_n' > 2.5$ . The linear coefficient  $p_1$  is recorded. The third fit linearly fits  $P$  vs  $\xi_n'$  with  $\xi_n' > 2.5$ . The linear coefficient  $p_2$  is recorded. Note that the second and third fits lack physical meaning, but they provide some qualitative information that helps the ML model give quantitative predictions. The following loop was used.

for  $\phi$  in  $[-\pi/12, \pi/4, 7\pi/12]$ :

$\xi = w \cos \phi + s \sin \phi$ .

Plot the histogram of  $P(\xi_n')$ .

Fit histograms to get features  $p_{0,\phi}$ ,  $p_{1,\phi}$ ,  $p_{2,\phi}$ .

Altogether nine features are extracted.

(5) Intermittency analysis. For a 2D time series  $(w_t, s_t)$ , a scalar velocity is defined as

$$\xi_t' = \frac{\sqrt{(w_{t+1} - w_{t-1})^2 + (s_{t+1} - s_{t-1})^2}}{2\Delta t} \quad (16)$$

and its average over time  $\tau$  is defined as

$$\bar{\xi}_i'(\tau) = \sum_{t=i\tau+1}^{(i+1)\tau} \frac{\xi_t'}{\tau}, \quad (17)$$

where  $i = 0, 1, 2, \dots, \lfloor \frac{T-1}{\tau} - 1 \rfloor$ . We introduced this particular measure because at relatively high values of the vortical force amplitude,  $k_c$ , the particle displays intermittent behavior characterized by large orbital excursions from equilibrium, yet below the critical value of  $k_{c,\text{max}}$ . As a result, the  $\xi'$  fluctuates at lower frequencies than other timescales in this system. This is characterized by the STD of  $\bar{\xi}_i'(\tau)$  over  $i$ . Two features are extracted from the 2D trajectory with  $\tau = 500$  and 1300, respectively.

(6) Vorticity estimation. Given a 2D time series  $(w_t, s_t)$ , the 2D velocity vector  $\bar{\mathbf{v}}_t$  is first calculated. Let  $v_{\text{mean}} = \sqrt{\langle v_t^2 \rangle}$

where  $\langle \dots \rangle$  represents averaging over  $t$ . A qualitative estimation of angular velocity is used:

$$\Omega(\tau) = \frac{1}{\Delta t} \left\langle \frac{\bar{\mathbf{v}}_t \times \bar{\mathbf{v}}_{t+\tau}}{v_{\text{mean}} \left( \frac{v_t + v_{t+\tau}}{2} \right) \tau} \right\rangle. \quad (18)$$

This form puts a larger weight on larger velocities, which is necessary since  $\Omega$  is completely dominated by noise for small velocities.  $\Omega(1)$  and  $\Omega(5)$  are two features used in the model.

(7) Linear correlation and mutual information. Built-in python packages [48] are used to compute the linear correlation and mutual information between all pairs of time series  $(w, s)$ ,  $(w, v_w)$ ,  $(w, v_s)$ ,  $(s, v_w)$ , and  $(s, v_s)$ , where  $v_w$  ( $v_s$ ) is the central difference velocity associated with  $w$  ( $s$ ). These measurements are most relevant for large values of  $k_c$ , where circular motion can be detected. Altogether this provides ten features.

(8) ULI [2]. ULI is a maximum-likelihood algorithm based on modified linear regression. The time series  $w, s, v_w$ , and  $v_s$  are used as inputs, along with a linear model of the forces, and the parameters of interest (i.e.,  $\omega, \gamma$ , etc.) are estimated. There are eight coefficients in the linear regression, which are eight features.

(9) The previous eight analyses give 63 features. Then, a bandpass filter is applied to  $w_t$  and  $s_t$  with an upper threshold = 2 Hz and a lower threshold = 0.01 Hz. The previous eight analyses are repeated for 63 more features. This was done to reduce noise in the original data, yet by including analysis on both filtered and unfiltered data, we avoid losing information with little cost of adding features. Counting the 6 features in preprocessing, altogether there are 132 features for the motion of single particles.

#### D. ML methods and performance

Two python-based ML algorithms (gradient boosting, an ensemble of decision trees, and neural network [48]) were trained on 132 extracted features from 400 000 simulated time series (training data set) to predict the six randomly chosen parameters. Within the algorithms, the gradient boosting model has parameters `n_estimators = 250`, `max_depth = 5`, and the dense neural network has five hidden layers, with size (128, 64, 64, 32, 16) and all hyperbolic tangent activations. Before training, both the features and the targets are normalized by the whole training batch to zero mean and unit variance. The performance of each method was benchmarked on 80 000 simulated time series (test data set). Figure 4(f) shows that ML methods are  $\approx 1.5\times$  more accurate at predicting  $\omega$  and  $\gamma$  than simply fitting analytical expressions to the Fourier spectrum of the data along the principal axes of confinement, and two to three times more accurate than ULI [2].

With regard to ULI, we note that the performance was excellent and comparable to the prediction error for ML *when using only Gaussian noise, no pixel locking, and no drift*. These sources of noise seemed to dramatically reduce the performance of ULI, yet these sources of noise are unavoidable in real experimental data. However, despite its lack of parameter estimation power on single, noisy data sets, ULI consistently ranked as one of the most important predictive features in the ML algorithms. Employing ULI in the simulated features

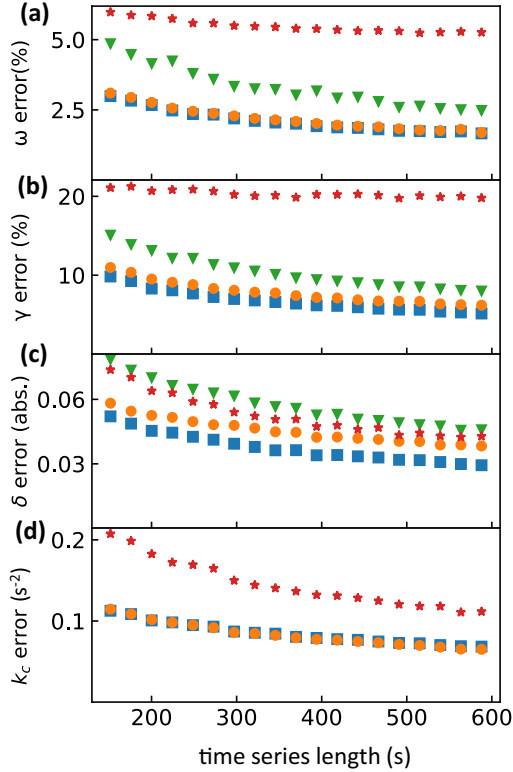


FIG. 4. The prediction error of various models for multiple parameters on one-particle simulated test data. Red stars represent ULI, green triangles represent Fourier spectrum, blue squares represent neural network, and orange circles represent gradient boosting. Note that the Fourier spectrum cannot predict  $k_c$  and must be based on a known  $\theta$ . Since ML is trained on a certain range of all parameters (Table I), unreasonable predictions of Fourier spectrum and ULI are also cropped to that range.

increased the total simulation and feature extraction time by 150%.

### E. Labeling experimental data

It is challenging to verify the accuracy of results when applying ML models to unlabeled experimental data. However, in our experiments, we measured the parameters using an independent, alternative method. By perturbing the particle with a magnet outside the chamber and observing the particle's relaxation to equilibrium, we fit the 2D trajectory and obtained estimates of  $\omega$ ,  $\gamma$ ,  $\delta$ , and  $\theta$ . Initially, we used a “mechanical” method to perturb the particle position by moving a grounded metal rod in close proximity to the single, levitated particle. However, this method would sometimes lead to unwanted particles being deposited in the experiment. By using a small, rare-earth magnet outside of the vacuum chamber, we found nearly identical results without introducing unwanted particles. The magnet was removed in a fraction of a second, while the particle relaxation process took more than 4 s.

Two examples of particle trajectories during relaxation to equilibrium after a perturbation, and the corresponding best fit, are shown in Figs. 5(a) and 5(b). Assuming  $k_c = 0$  and

ignoring the stochastic noise term, Eqs. (1)–(3) can be solved analytically:

$$w(t) = A_w e^{-\gamma t/2} \cos\left(t\sqrt{\omega_-^2 - \frac{\gamma^2}{4}} + \phi_w\right), \quad (19)$$

$$s(t) = A_s e^{-\gamma t/2} \cos\left(t\sqrt{\omega_+^2 - \frac{\gamma^2}{4}} + \phi_s\right), \quad (20)$$

$$x(t) = w(t) \cos \theta - s(t) \sin \theta, \quad (21)$$

$$y(t) = s(t) \cos \theta + w(t) \sin \theta. \quad (22)$$

Here the fitting parameters  $A_w$ ,  $A_s$ ,  $\phi_w$ , and  $\phi_s$  depend on the initial conditions, and  $\gamma$ ,  $\omega_-$ ,  $\omega_+$ , and  $\theta$  are an estimation of the model parameters as described in Sec. III A, assuming  $k_c = 0$ .

### F. Predicting experimental data—Results

We directly compared these measurements with the results from the ML model (the mean of the predictions from neural network and gradient boosting), which measures the parameters *in situ* without perturbations. The difference between the model's predictions and the labels inferred from the aforementioned perturbation method lay within the error bars estimated from the simulated test data in parameters  $\gamma$ ,  $\omega$ ,  $\delta$ , and  $\theta$ . In other words, the model predicts experimental data as accurately as simulated data, so the mismatch between experimental and simulated data was alleviated. In general, ULI was able to predict  $\omega$ ,  $\delta$ , and  $\theta$ , yet with an accuracy that was poor compared to ML, which may be expected since ULI does not require training from multiple data sets.

Both the perturbation method and ML show excellent agreement with the prediction of  $\gamma$  from Epstein's law [Eq. (4)]. The confinement asymmetry,  $\delta$ , could be as large as 0.2 although the experimental configuration was quite symmetric and the illuminating laser only contributed to a 1% asymmetry since a gradient in laser intensity is needed to change the confinement strength. Additionally, the gas flow and pumping rate were low and did not affect  $\delta$ . We speculate that the asymmetry in the confinement may be due to background flows in the plasma environment. ULI produced wildly varying predictions of  $\gamma$ , even sometimes negative values. Thus, we did not include it in Fig. 5(c).

In analyzing the stochastic noise, we found that the prediction of the Levy parameter  $\alpha$  reflected the particle's effective temperature,  $2k_B T \approx m\omega^2(\langle x \rangle^2 + \langle y \rangle^2)$ , where  $k_B$  is Boltzmann's constant [Fig. 5(g), inset]. We have assumed an equipartition between kinetic and potential energy, and expressed the temperature here in terms of the average potential energy to avoid calculating derivatives for the velocity. Importantly, no information about the temperature was passed to the ML model since all time series were normalized. Reported values of  $T$  in dusty plasmas driven by Brownian motion vary from 300 to 1000 K [21,49]. For most experiments, we found  $T = 300$ – $460$  K, with  $1.9 < \alpha < 2$ , indicating nearly Gaussian noise from the room-temperature neutral collisions [Fig. 5(g), inset]. Larger temperatures typically corresponded to smaller values of  $\alpha$ . We speculate that this could be caused

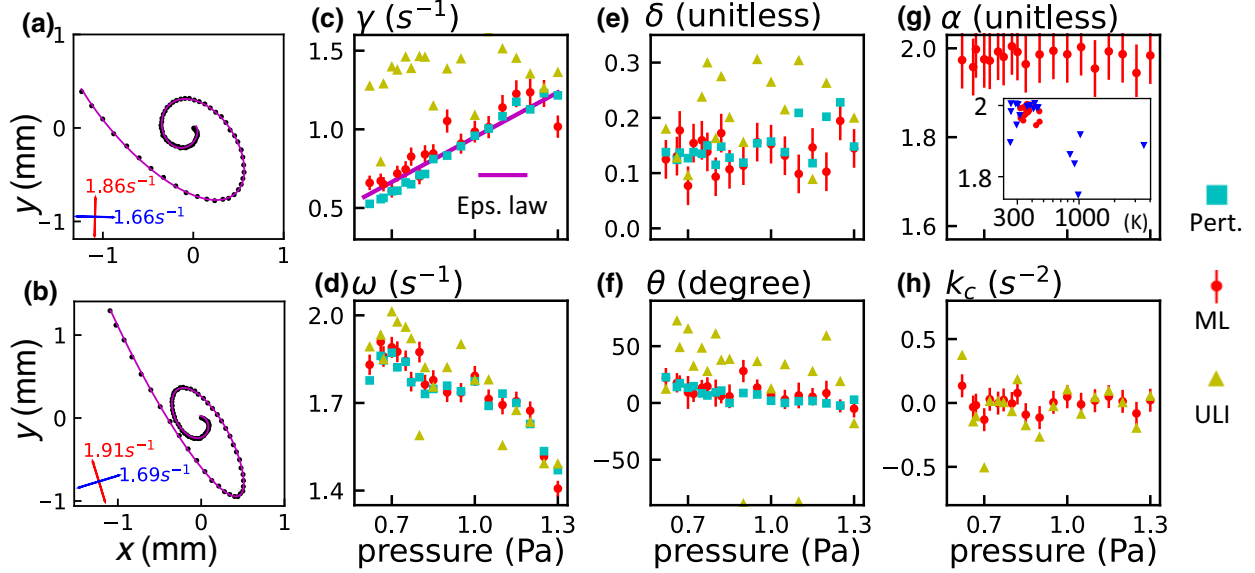


FIG. 5. (a),(b) Two different experiments of the same particle relaxing to equilibrium after a perturbation. The pressure was  $P = 0.80$  Pa. The magenta lines are fits using Eqs. (21) and (22). The red and blue lines indicate  $\omega_+$  and  $\omega_-$  and their orientations, respectively. (c)–(f) The prediction from ML (the mean of the predictions from the neural network model and the gradient boosting model; red circles), ULI (yellow triangles), and reference estimation from the perturbation experiments (Pert.; cyan squares) for  $\gamma$ ,  $\omega$ ,  $\delta$ , and  $\theta$  for particles with diameter  $12.8 \mu\text{m}$ . The purple line in (c) represents the theoretical value of Epstein’s law [Eq. (4)]. (g)  $\alpha$  and (h)  $k_c$  as predicted by ML. These parameters cannot be verified by the perturbation experiments. The inset in (g) shows the prediction of  $\alpha$  correlates with the potential temperature of the particle. Red squares represent  $12.8\text{-}\mu\text{m}$  particles and blue triangles represent  $9.46\text{-}\mu\text{m}$  particles. Error bars were obtained from predictions on the simulated test data set, and errors based on fitting perturbed trajectories are smaller and not shown for clarity.

by contamination with undetectable, small dust particles since the effective temperature was seen to increase over time in some experiments. Often these particles were “dropped” by shutting off the plasma, and a new particle was deposited in its place. In any case, the source of the higher effective temperatures was non-Gaussian, although we could not definitively identify the origin of the noise.

The nonconservative force from Eq. (3) was smaller than the prediction error bars for most experiments [Fig. 5(h)]. Part of the motivation for including  $k_c$  in our linearized model was observations that particles can undergo small elliptical orbits without any apparent input of energy [33]. In Nosenko *et al.* [33], the gravitational leveling of the electrode played a role, presumably due to a feedback between the plate geometry and the background ion flows. Another possibility would be a nonspherical or broken particle, which could then interact with background ion flows.

## IV. TWO-PARTICLE MOTION

### A. Linearized model

When two particles are present, their mutual repulsion,  $m\vec{f}_{ij} = m f_p(r) \vec{e}_{ij}$ , displaces them from the center of the confining potential, as shown in Fig. 6(a). Here,  $f_p$  is the reduced force,  $\vec{e}_{ij}$  is a unit vector from particle  $i$  to  $j$ ,  $r$  is the particle separation, and  $m$  is the mass of a particle. At equilibrium, the particle separation is  $r_0$ , which varies with pressure and the vertical position  $z$  [Figs. 6(b) and 6(c)]. In particular, below  $P \approx 1$  Pa,  $r_0$  sharply increased to a plateau

and the height increased, presumably due to an increase in the plasma Debye length. Here we aim to simultaneously

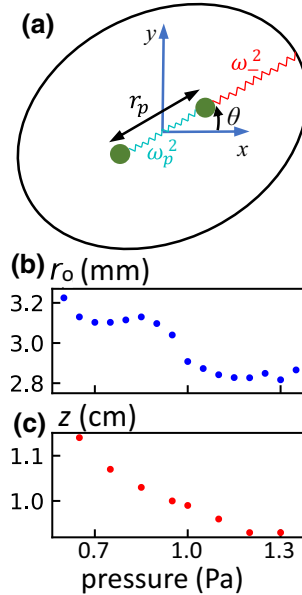


FIG. 6. (a) Two charged particles experience mutual repulsion, characterized by frequency  $\omega_p$  at equilibrium, and external harmonic confinement.  $k_c$  and  $\omega_+$  are omitted for clarity. All subsequent panels show parameter estimates as a function of pressure,  $P$ . (b),(c) The equilibrium particle separation and vertical height above the electrode varied with the gas pressure.



TABLE II. The parameters for two-particle simulation.

Name	Description	Range	Drift amp.	Drift time
$\omega$	Confinement freq.	(1.3,2.5) s <sup>-1</sup>	(0,2%)	(12,800) s
$\omega_p$	Interaction freq.	(1.7,3.3) s <sup>-1</sup>	(0,2%)	(12,800) s
$\gamma$	Damping coeff.	(0.4,1.7) s <sup>-1</sup>	(0,2%)	(12,800) s
$\delta$	Asymmetry	(0,0.35)	(0,0.01)	(12,800) s
$\theta$	Weak dimension	( $-\pi/2, \pi/2$ )	(0,0.1)	(12,800) s
$k_c$	Vortex force coeff.	( $-k_{c,\max}, k_{c,\max}$ ) s <sup>-2</sup>	None	None
$\alpha$	Noise distribution	(1.6,2.0)	None	None
$c_0$	Noise scale	(0.005,0.03)	None	None
$x_0$	Equi. position	0	(0,0.02)	(12,800) s
$y_0$	Equi. position	0	(0,0.02)	(12,800) s
$r_0$	Equi. separation	1	(0,0.02)	(12,800) s
$m_1$	Mass ratio	(0.9,1.1)	None	None
$A$	Second-order coeff.	(-3,10) s <sup>-1</sup>	None	None
$l_{\text{pixel}}$	Simulated pixel width	(0.005,0.02)	None	None

infer

$$\omega_-^2 \frac{r_0}{2} = f_p(r_0), \quad (23)$$

$$\omega_p^2 = -\left. \frac{df_p}{dr} \right|_{r=r_0}, \quad (24)$$

and all other model parameters from Eqs. (1)–(3) with high accuracy using noisy data. We linearize the small-amplitude motion of each particle about their equilibrium position. The equation of motion for particle  $i$  is

$$\ddot{\vec{r}}_i = -\vec{\nabla}\phi + \vec{\nabla} \times \vec{A} - \vec{f}_{ij} - \gamma \dot{\vec{r}}_i + \vec{N}(\alpha), \quad (25)$$

$$\vec{f}_{ij} = f_p \vec{e}_{ij} = \left( -\omega_p^2 (r - r_0) + \omega^2 (1 - \delta) \frac{r_0}{2} \right) \vec{e}_{ij}. \quad (26)$$

The equilibrium force and its differential can be used to solve for two independent parameters in a model for  $f_p$ . The most commonly used model for  $f_p$  assumes a screened, Coulomb interaction:

$$F_D = m f_D = \frac{Q^2}{4\pi\epsilon_0 r} \left( \frac{1}{r} + \frac{1}{\lambda} \right) e^{-r/\lambda}. \quad (27)$$

Here  $\epsilon_0$  is the permittivity of free space. With this assumption, similar linearized models have been used to directly infer  $Q$  and  $\lambda$  from the one-dimensional motion of two particles using Fourier analysis [29,30,50,51]. Here we allow for entanglement between motion in two dimensions, and provide estimates of all model parameters. Analogous to the single-particle model, parameters  $\gamma$ ,  $\omega$ ,  $\delta$ ,  $\theta$ ,  $k_c$ ,  $\alpha$ , and  $\omega_p$  were randomly chosen to simulate time series using Eqs. (25) and (26). The same feature extraction methods as described for a single particle were applied to the center of mass and relative coordinates,  $(\vec{r}_1 + \vec{r}_2)/2$  and  $\vec{r}_1 - \vec{r}_2$ , with some alterations described in Sec. IV C. The neural network and gradient boosting models were trained on 227 extracted features from 400 000 simulated time series to predict the seven parameters, and their performance on simulated test data is shown in Fig. 7.

## B. Simulation details

For simulations involving two particles, the drift of the parameters can change the equilibrium separation of the particles, which is comparable to the amplitude of the Brownian motion. Thus, we allowed for a small drift of many parameters, as listed in Table II. Importantly, the introduction of a repulsive force between the particles leads to a natural length scale,  $r_0$ , which is the equilibrium separation between the

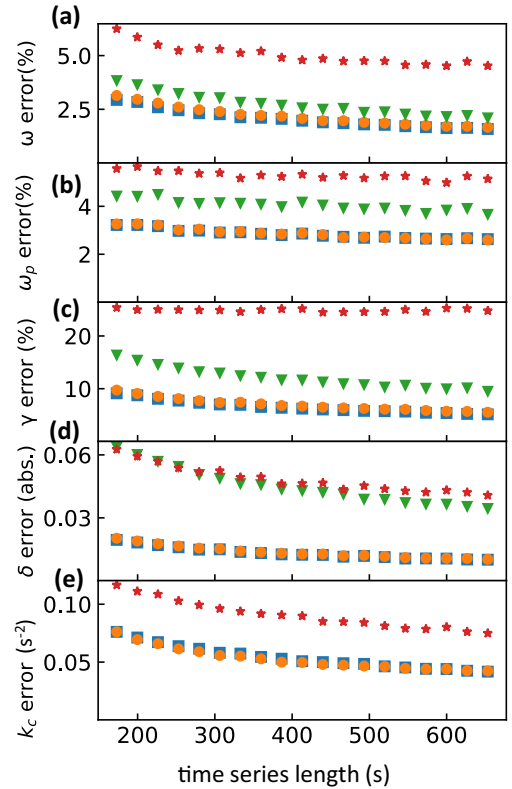


FIG. 7. The prediction error of various models for multiple parameters on two-particle simulated test data. Red stars represent ULI, green triangles represent Fourier spectrum, blue squares represent neural network, and orange circles represent gradient boosting.

particles after balancing external confinement and mutual repulsion. In the simulations,  $r_0$  is set to 1, but is allowed to drift. This fixed length scale means we must choose the amplitude of the noise to match what is observed in the experiments. The noise scale,  $c = c_0\sqrt{\gamma\omega^2/\Delta t}$ , is similar to the simulations for one particle, but here  $c_0 \ll r_0$ . The value of  $c_0$  is randomly chosen in each simulation, and represents the amplitude of Brownian motion in units of length measured by the particle separation,  $r_0$ . Note that  $c$  has units of acceleration since mass is normalized and  $c_0$  has units of length.

Furthermore, to make the model more general, a second-order term with random coefficient  $A$  was added to the reduced particle interaction force,

$$f_p = \left( \frac{A}{r_0}(r - r_0)^2 - \omega_p^2(r - r_0) + \omega^2(1 - \delta)\frac{r_0}{2} \right). \quad (28)$$

This is identical to Eq. (26), albeit with the addition of the second-order term. Note that typically  $r - r_0 \approx c_0 \ll 1$ , so the second-order term is negligible, but was included for generality. Experimentally, the two particles may be slightly different in size, so a mass difference was considered in simulations. We randomly chose the mass of one particle,  $m_1$ , to vary by 10% (Table II). As a reminder, none of the parameters have a mass unit, so we can arbitrarily fix the sum of the masses,  $m_1 + m_2 = 2$ . The acceleration of each particle  $\ddot{\mathbf{r}}_i$  was calculated as

$$\ddot{\mathbf{r}}_i = \bar{\mathbf{f}}_i/m_i, \quad (29)$$

where  $\bar{\mathbf{f}}_i$  is the sum of all the reduced forces exerted on particle  $i$ . After each simulation, pixel-locking noise was added to the trajectory of each particle with  $l_{\text{pixel}}$ , as listed in Table II. For all simulations, we used a second-order, velocity Verlet time-stepping method to integrate the equations of motion.

### C. Features of two-particle motion

For two-particle systems, the rough estimation of  $\theta$  is simply the direction in which the particles align (the weak axis), so we do not perform the preprocessing step. The four-dimensional time series representing the motion of the two particles is projected into the center-of-mass motion and the relative motion. The same steps (1)–(9) as in Sec. III C are applied to the center-of-mass motion (126 features), and similar methods are applied to the relative motion with the following revisions:

(1) The fit of the Fourier spectrum only needs two projections, parallel and perpendicular to the alignment of the two particles, instead of four. Furthermore, an additional fit is conducted to the Fourier spectrum of the mode perpendicular to the alignment, where motion predominantly occurs in the azimuthal direction. If the asymmetry  $\delta = 0$ , this mode would have zero frequency. Thus we used a window  $0.1 \text{ s}^{-1} < \omega < 1 \text{ s}^{-1}$ , and  $\gamma_{\text{FT}}$ ,  $\omega_{\text{FT}}$ , but not  $A_{\text{FT}}$ , are recorded as features. Altogether there are 8 features instead of 12.

(2) The same logic applies for autocorrelation, so there are 6 features instead of 8.

(3) Percentiles and equipartition analysis are performed to two (parallel and perpendicular) projections rather than six. Altogether 4 features instead of 12.

(4) Velocity distribution analysis is performed to two projections rather than three. Altogether six features instead of nine.

The revised eight analyses gives 46 features. After the aforementioned smoothing, the analyses are performed again for another 46 features. Moreover, since the trajectory is normalized by the particle separation so that  $r_0 = 1$ , the STD of each of the four modes before and after smoothing are eight new features. Plus the time series length, altogether we have  $63 \times 2 + 46 \times 2 + 8 + 1 = 227$  features for the motion of two particles.

### D. Predicting experimental data—Results

Figure 8 shows the ML model prediction on experimental data, for particles with diameters 12.8  $\mu\text{m}$  (label 1) and 9.46  $\mu\text{m}$  (label 2). The prediction of  $\delta$  is significantly different between the two sizes of particles, as shown in Fig. 8(a). This is likely due to their difference in vertical equilibrium position in  $z$  [Fig. 1(g)]. Though  $k_c$  seems systemically positive for both particles, its amplitude was usually smaller than the error bar [Fig. 8(b)]. We did not specifically train the model on the parameter  $\theta$  since it is essentially determined by the alignment of the vector  $\mathbf{r}_2 - \mathbf{r}_1$ . Although a finite value of  $k_c$  would lead to an angular displacement away from the weak confinement axis, we found that a numerical value of  $k_c = 0.1 \text{ s}^{-2}$  would lead to a small angular displacement of  $\approx 0.1$  radians.

The damping  $\gamma$  varied linearly with  $P$  and was well described by Epstein's law [Eq. (4)] for both particles, as illustrated in Fig. 8(c). The error bars also decreased considerably when compared to the analysis of a single particle (Fig. 4 vs Fig. 7). Both  $\omega$  and  $\omega_p$  displayed a slight, nonmonotonic variation with pressure, outside the range of the error bars in the prediction [Fig. 8(d)]. We note that the eigenfrequency for the relative coordinate motion is  $\sqrt{\omega_p^2/\mu + \omega^2}$ , where  $\mu = m_1m_2/(m_1 + m_2)$  is the reduced mass. If the masses of the particles are identical, then  $\mu = 1/2$ . However, assuming that the particles' mass differ by less than 10%,  $\mu = 0.495 \pm 0.005$ . Therefore, the coefficient  $\sqrt{1/\mu} \approx \sqrt{2}$  is included in Fig. 8(d) for clarity.

For 12.8- $\mu\text{m}$ -diameter particles, the “kink” for  $\omega$  and  $\omega_p$  lies at the steepest decrease of  $r_0$  in Fig. 6. Although these features may be related, we cannot say for sure since the particle height, charge, and Debye length all vary with pressure, and all contribute to  $\omega$  and  $\omega_p$ . Additionally, we found that the parameter  $\alpha$  could not be well predicted from the center of mass and relative coordinates in the two-particle system. Although we are uncertain why the prediction failed for  $\alpha$ , we speculate that the distribution in errors from pixel locking are nonadditive (unlike the Levy-stable distribution). The motion of the center of mass, for example, is the sum of the particle positions. Thus the total noise distribution for the center of mass and relative coordinates could be quite different from a one-particle system.

Finally, to estimate  $Q$  and  $\lambda$ , we used two models for  $f_p$ . The first is Eq. (27), and the second model is derived from a potential ( $mf_w = -d\phi_w/dr$ ) that incorporates the virtual

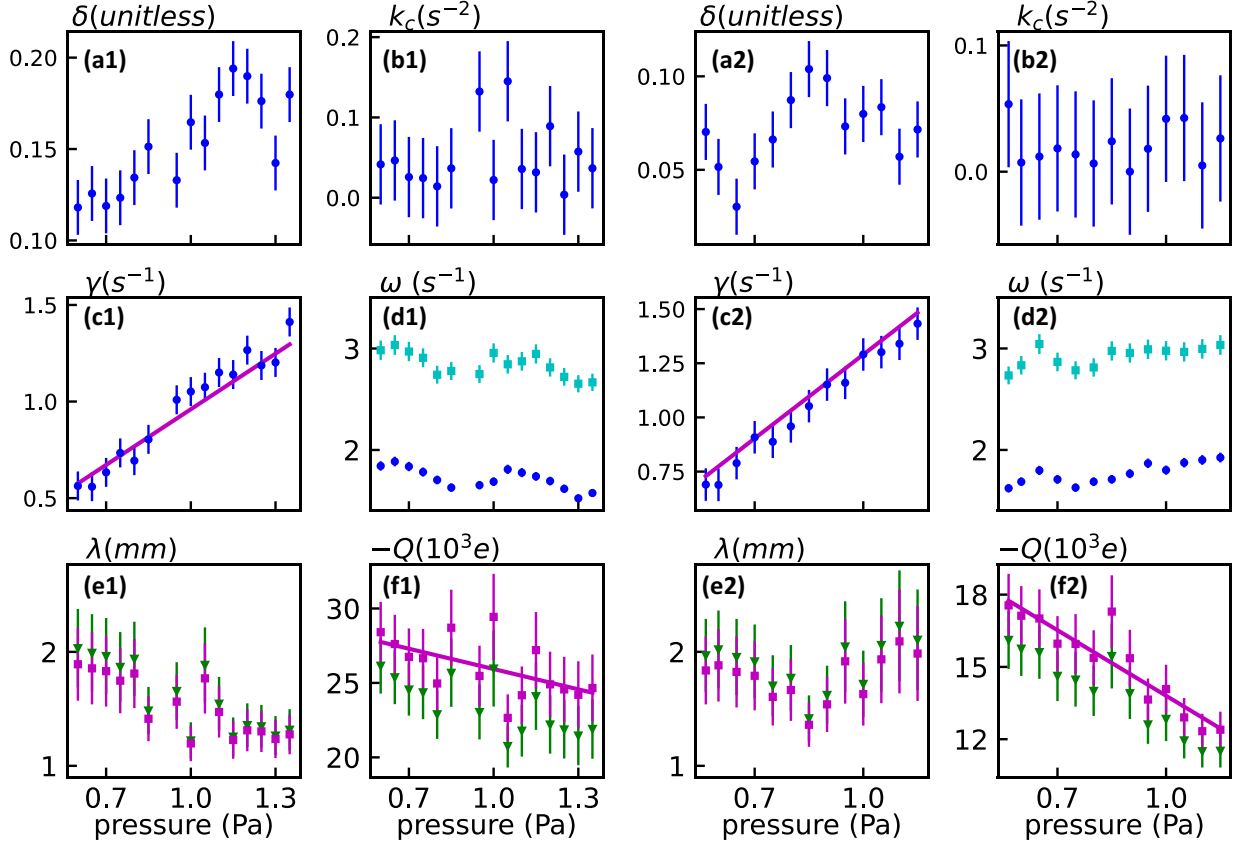


FIG. 8. ML predictions for the two-particle systems. Data is shown for 12.8- $\mu\text{m}$  particles (denoted by 1) and for 9.46- $\mu\text{m}$  particles (denoted by 2). The ML prediction is the average of results from the neural network and gradient boosting methods. (a)–(c) The predictions of  $\delta$ ,  $k_c$ , and  $\gamma$  by ML are represented by blue circles with error bars estimated from Fig. 4. The magenta line in (c) is the theoretical  $\gamma$  from Epstein's law. (d) The confinement frequency  $\omega$  (blue circles) and interaction frequency  $\sqrt{2}\omega_p$  (cyan squares) predicted by ML. (e), (f) The Debye length  $\lambda$  and particle charge  $Q$  calculated from the frequencies using Eqs. (27) (green triangles) and (30) (magenta squares) from the main text. The magenta line is a linear fit to the magenta squares to guide the eye.

charge ( $\tilde{q}$ ) from the ion wake beneath each particle [18,52]:

$$\phi_w = \frac{Q^2}{4\pi\epsilon_0\lambda} \left[ \frac{e^{-r/\lambda}}{r/\lambda} - \tilde{q} \frac{\lambda e^{-r_w/\lambda}}{r_w + b\lambda e^{-r_w/\lambda}} \right]. \quad (30)$$

Here  $r_w = \sqrt{r_0^2 + 0.12\lambda^2}$ ,  $\tilde{q} = 0.3$ , and  $b = 1$ . The parameter  $b$  controls the size of the virtual charge cloud. After inferring  $\omega_p$  and  $\omega_-$ , the interaction force and its derivative at  $r = r_0$  were numerically solved for  $Q$  and  $\lambda$ , following Eqs. (23) and (24). For both models of the interaction force, the predictions of  $Q$  and  $\lambda$  differed by only 10%, as shown in Figs. 8(e) and 8(f), for both sizes of particles. We expect this difference to be much larger if the particles have significant vertical separation, where ion wake interactions lead to nonreciprocal forces [14].

For both particles,  $Q$  decreased with pressure. The Debye screening length  $\lambda$  also decreased with pressure for 12.8- $\mu\text{m}$  particles, but there seemed to be a kink in  $\lambda$  for the 9.46- $\mu\text{m}$  particle data at  $P \approx 0.75$  Pa. This is likely due to the variation in height with pressure, and a similar feature is also visible in  $\delta$  (Fig. 8a2). It is also important to note that the error in estimating  $Q$  and  $\lambda$  can be up to ten times larger than  $\omega$  [Figs. 8(e)

and 8(f)]. This can be illustrated by examining Eqs. (23), (24), and (27). These equations can be solved analytically, resulting in the following expression for  $\lambda$ :

$$\lambda = \frac{r_0}{2} \left( \frac{\omega_p^2 + \omega_-^2}{\sqrt{\omega_p^4 - \omega_-^4}} - 1 \right). \quad (31)$$

When  $\omega_p$  is close to  $\omega_-$ , perhaps with overlapping error bars, then the uncertainty in  $\lambda$  diverges. This emphasizes the importance of accurate estimation of all parameters, independent of the model chosen for  $f_p$ . Although we allowed for a 10% variation in mass between the particles, this variation was not considered when calculating  $Q$ . The reported values of  $Q$  should be taken as an average of the charge on both particles since the mass of each particle may be slightly different.

Finally, we can compare the results shown in Fig. 8(f) to theoretical predictions in our plasma conditions. First, we can estimate the total Debye length in the plasma,

$$\lambda_D = \sqrt{\frac{k_B\epsilon_0 T_e T_i}{e^2 n_p (T_e + T_i)}}, \quad (32)$$

where  $n_p$  is the quasineutral plasma density,  $T_i$  is the ion temperature, and  $e$  is the elementary charge. As stated previously,  $T_e \approx 1.3\text{--}1.5$  eV, and  $T_i \approx 0.026$  eV, so that  $\lambda_D$  is dominated by the ion temperature. For our experimental conditions, the plasma density has been measured previously using a Langmuir probe, and  $n_p \approx (2\text{--}5) \times 10^{13} \text{ m}^{-3}$  [22]. Thus, we expect  $\lambda_D \approx 160\text{--}260 \mu\text{m}$ . However, the screening length between particles ( $\lambda$ ) is known to be 5–15 times larger than  $\lambda_D$  [53]. Thus, our measurement of 1–2 mm shown in Fig. 8(e) is reasonable and approximately ten times larger than  $\lambda_D$ . Although  $\lambda_D$  is expected to increase at low pressures since  $n_p$  decreases, the exact dependence of the screening length  $\lambda$  on the ion Debye length  $\lambda_D$  is not well understood, especially within a plasma sheath. Subsequently, we do not currently have an explanation for the nonmonotonic behavior seen in Fig. 8(e2).

Second, the particle charge  $Q$  can be estimated, to lowest order, by orbital-motion-limited (OML) theory [32,54,55]. In its simplest form, we assume that the electron and ion concentrations are equal (quasineutrality). The electron velocity distribution is Maxwellian, but the ion drift velocity  $u_i$  towards the electrode must be considered in the ion velocity distribution. In this regime, the dust charge is determined by solving the following equation numerically (see Sec. 2.4.3 in [32]):

$$0 = \sqrt{\frac{T_e m_i}{T_i m_e}} e^{eQ/4\pi\epsilon_0 a k_B T_e} \quad (33)$$

$$- \frac{\text{erf}(\xi)[2\pi\epsilon a k_B T_i (2\xi^2 + 1) - qQ]}{8\sqrt{\pi}\epsilon a k_B T_i \xi} - \frac{1}{2} e^{-\xi^2}, \quad (34)$$

where  $\text{erf}(\xi)$  is the error function. The parameter  $\xi$  is the ratio of the drift velocity to the ion thermal velocity,  $\xi = u_i/v_{i,\text{th}}$ , where  $v_{i,\text{th}} = \sqrt{2k_B T_i/m_i}$ ,  $m_i$  is the mass of an argon ion, and  $m_e$  is the electron mass. For a simple estimate, we use the Bohm velocity for the ions because the particles levitate near the edge of the plasma sheath, so that  $u_i = \sqrt{k_B T_e/m_i}$ .

Using the parameters from our experiment,  $\xi \approx 5$ , and OML theory predicts that the 9.46- $\mu\text{m}$  particles are negatively charged with 16 600–19 200 electrons, and the 12.8- $\mu\text{m}$  particles are negatively charged with 22 500–26 000 electrons. This is in excellent agreement with the results shown in Fig. 8(f). Additionally, the dust charge should decrease as pressure increases since ion-neutral collisions become more important as the mean free path decreases ([56,57], Sec. 2.4.4

in [32]). This expectation is also consistent with Fig. 8(f). A more quantitative analysis of the pressure dependence of the particle charge is hampered by small but simultaneous variations of electron temperature, drift velocity, and plasma density as the pressure is changed in the experiment. Additionally, alterations to OML theory to include collisions typically ignore ion drift, and both are present in our experiments. Overall, these results highlight the importance of measuring particle charge *in situ*, as demonstrated here.

## V. CONCLUSION

In conclusion, we have overcome two major restrictions when inferring parameters from experimental data with ML: data labeling and mismatch. In our dusty plasma experiments, we label our data by observing particles' relaxation after perturbation. The mismatch between commonly simulated data (with Gaussian stochastic forces or Gaussian measurement error), and experimental data (with drift, pixel locking, non-Gaussian stochastic force, etc.) is handled by including these artifacts in our simulation. The agreement between the label for experimental data and our models' prediction demonstrates the alleviation of this mismatch. Additionally, this work provides simultaneous and accurate estimates of multiple parameters from noisy data, for example, the model provides an accurate *in situ* determination of Epstein drag, the particle charge, and the Debye length in dusty plasmas.

Our ML model employed supervised machine learning since stochastic forces and trajectories are an inevitable part of the dynamics. Ideally, the forces in dusty plasmas could be inferred from the experimental data itself, without the need for simulations in the training data set. In dynamical systems with many particles, the interaction force cannot be linearized, yet particle motion is dominated by deterministic forces, and noise is secondary. Thus it may be possible to infer interaction forces in more complex and dynamic systems, which remains an open challenge.

## ACKNOWLEDGMENTS

We thank I. Nemenman, E. Abdelaleem, and G. Gogia for stimulating discussions. This material is based upon work supported by the National Science Foundation under Grant No. 2010524, and the U.S. Department of Energy, Office of Science, Office of Fusion Energy Sciences program under Award No. DE-SC0021290.

- 
- [1] B. Lehle and J. Peinke, Analyzing a stochastic time series obeying a second-order differential equation, *Phys. Rev. E* **91**, 062113 (2015).
  - [2] D. B. Brückner, P. Ronceray, and C. P. Broedersz, Inferring the Dynamics of Underdamped Stochastic Systems, *Phys. Rev. Lett.* **125**, 058103 (2020).
  - [3] J. Bongard and H. Lipson, Automated reverse engineering of nonlinear dynamical systems, *Proc. Natl. Acad. Sci. USA* **104**, 9943 (2007).
  - [4] K. Champion, B. Lusch, J. N. Kutz, and S. L. Brunton, Data-driven discovery of coordinates and governing equations, *Proc. Natl. Acad. Sci. USA* **116**, 22445 (2019).
  - [5] S. L. Brunton, J. L. Proctor, and J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, *Proc. Natl. Acad. Sci. USA* **113**, 3932 (2016).
  - [6] B. C. Daniels and I. Nemenman, Automated adaptive inference of phenomenological dynamical models, *Nat. Commun.* **6**, 8133 (2015).
  - [7] B. Lusch, J. N. Kutz, and S. Brunton, Deep learning for universal linear embeddings of nonlinear dynamics, *Nat. Commun.* **9**, 4950 (2018).
  - [8] V. Bapst, T. Keck, A. Grabska-Barwińska, C. Donner, E. D. Cubuk, S. S. Schoenholz, A. Obika, A. W. Nelson, T. Back, D.

- Hassabis *et al.*, Unveiling the predictive power of static structure in glassy systems, *Nat. Phys.* **16**, 448 (2020).
- [9] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach, *Phys. Rev. Lett.* **120**, 024102 (2018).
- [10] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, An analysis of environment, microphone and data simulation mismatches in robust speech recognition, *Comput. Speech Lang.* **46**, 535 (2017).
- [11] J. Wang, G. Gui, and H. Sari, Generalized automatic modulation recognition method based on distributed learning in the presence of data mismatch problem, *Phys. Commun.* **48**, 101428 (2021).
- [12] M. Chaudhuri, A. V. Ivlev, S. A. Khrapak, H. M. Thomas, and G. E. Morfill, Complex plasma—the plasma state of soft matter, *Soft Matter* **7**, 1287 (2011).
- [13] T. Bockwoldt, O. Arp, K. O. Menzel, and A. Piel, On the origin of dust vortices in complex plasmas under microgravity conditions, *Phys. Plasmas* **21**, 103703 (2014).
- [14] A. V. Ivlev, J. Bartnick, M. Heinen, C.-R. Du, V. Nosenko, and H. Löwen, Statistical Mechanics where Newton's Third Law is Broken, *Phys. Rev. X* **5**, 011035 (2015).
- [15] K. Qiao, J. Kong, L. S. Matthews, and T. W. Hyde, Mode couplings and resonance instabilities in finite dust chains, *Phys. Rev. E* **91**, 053101 (2015).
- [16] G. Gogia and J. C. Burton, Emergent Bistability and Switching in a Nonequilibrium Crystal, *Phys. Rev. Lett.* **119**, 178004 (2017).
- [17] G. Gogia, W. Yu, and J. C. Burton, Intermittent turbulence in a many-body system, *Phys. Rev. Research* **2**, 023250 (2020).
- [18] N. P. Kryuchkov, L. A. Mistryukova, A. V. Sapelkin, and S. O. Yurchenko, Strange attractors induced by melting in systems with nonreciprocal effective interactions, *Phys. Rev. E* **101**, 063205 (2020).
- [19] J. D. Williams and E. Thomas, Jr., Measurement of the kinetic dust temperature of a weakly coupled dusty plasma, *Phys. Plasmas* **14**, 063702 (2007).
- [20] G. E. Norman and A. V. Timofeev, Kinetic temperature of dust particle motion in gas-discharge plasma, *Phys. Rev. E* **84**, 056401 (2011).
- [21] J. Kong, K. Qiao, L. S. Matthews, and T. W. Hyde, Temperature measurement of a dust particle in a RF plasma GEC reference cell, *J. Plasma Phys.* **82**, 905820505 (2016).
- [22] J. Méndez Harper, G. Gogia, B. Wu, Z. Laseter, and J. C. Burton, Origin of large-amplitude oscillations of dust particles in a plasma sheath, *Phys. Rev. Research* **2**, 033500 (2020).
- [23] S. Nunomura, T. Misawa, N. Ohno, and S. Takamura, Instability of Dust Particles in a Coulomb Crystal Due to Delayed Charging, *Phys. Rev. Lett.* **83**, 1970 (1999).
- [24] J. Kong, T. W. Hyde, L. Matthews, K. Qiao, Z. Zhang, and A. Douglass, One-dimensional vertical dust strings in a glass box, *Phys. Rev. E* **84**, 016411 (2011).
- [25] T. W. Hyde, J. Kong, and L. S. Matthews, Helical structures in vertically aligned dust particle chains in a complex plasma, *Phys. Rev. E* **87**, 053106 (2013).
- [26] M. Choudhary, R. Bergert, S. Mitic, and M. H. Thomas, Three-dimensional dusty plasma in a strong magnetic field: Observation of rotating dust tori, *Phys. Plasmas* **27**, 063701 (2020).
- [27] M. Schwabe, U. Konopka, P. Bandyopadhyay, and G. E. Morfill, Pattern Formation in a Complex Plasma in High Magnetic Fields, *Phys. Rev. Lett.* **106**, 215004 (2011).
- [28] E. Thomas, Jr., U. Konopka, R. L. Merlino, and M. Rosenberg, Initial measurements of two- and three-dimensional ordering, waves, and plasma filamentation in the magnetized dusty plasma experiment, *Phys. Plasmas* **23**, 055701 (2016).
- [29] T. E. Sheridan, M. R. Katschke, and K. D. Wells, Measurement of electric field and gradient in the plasma sheath using clusters of floating microspheres, *Rev. Sci. Instrum.* **78**, 023502 (2007).
- [30] A. K. Mukhopadhyay and J. Goree, Two-Particle Distribution and Correlation Function for a 1D Dusty Plasma Experiment, *Phys. Rev. Lett.* **109**, 165003 (2012).
- [31] M. Lampe, G. Joyce, and G. Ganguli, Interactions between dust grains in a dusty plasma, *Phys. Plasmas* **7**, 3851 (2000).
- [32] A. Melzer, *Physics of Dusty Plasmas* (Springer Nature, Switzerland AG, 2019).
- [33] V. Nosenko, F. Luoni, A. Kaouk, M. Rubin-Zuzic, and H. Thomas, Active Janus particles in a complex plasma, *Phys. Rev. Research* **2**, 033226 (2020).
- [34] A. Melzer and J. Goree, Fundamentals of dusty plasmas, in *Low Temperature Plasmas Fundamentals, Technologies, and Techniques*, 2nd ed., edited by R. Hippler, H. Kersten, M. Schmidt, and K. H. Schoenbach (Wiley-VCH, Weinheim, 2008), Vol. 1, pp. 157–206.
- [35] Z. Ding, J. Yao, Y. Wang, C. Yuan, Z. Zhou, A. Kudryavtsev, R. Gao, and J. Jia, Machine learning combined with Langmuir probe measurements for diagnosis of dusty plasma of a positive column, *Plasma Sci. Technol.* **23**, 095403 (2021).
- [36] Z. Ding, Q. Guan, C. Yuan, Z. Zhou, and Z. Qu, A method of electron density of positive column diagnosis—combining machine learning and Langmuir probe, *AIP Adv.* **11**, 045028 (2021).
- [37] J. Chalaturnyk and R. Marchand, A first assessment of a regression-based interpretation of Langmuir probe measurements, *Frontiers Phys.* **7**, 63 (2019).
- [38] A. H. Bukhari, M. Sulaiman, M. A. Z. Raja, S. Islam, M. Shoaib, and P. Kumam, Design of a hybrid NAR-RBFs neural network for nonlinear dusty plasma system, *Alexandria Eng. J.* **59**, 3325 (2020).
- [39] H. Huang, M. Schwabe, and C.-R. Du, Identification of the interface in a binary complex plasma using machine learning, *J. Imaging* **5**, 36 (2019).
- [40] Z. Wang, J. Xu, Y. E. Kovach, B. T. Wolfe, E. Thomas, Jr., H. Guo, J. E. Foster, and H.-W. Shen, Microparticle cloud imaging and tracking for data-driven plasma science, *Phys. Plasmas* **27**, 033703 (2020).
- [41] Z. Ding, L. S. Matthews, and T. W. Hyde, A machine learning based Bayesian optimization solution to non-linear responses in dusty plasmas, *Mach. Learn.: Sci. Technol.* **2**, 035017 (2021).
- [42] D. B. Allan, T. Caswell, N. C. Keim, C. M. van der Wel, and R. W. Verweij, soft-matter/trackpy: Trackpy v0.5.0.
- [43] Y. Feng, J. Goree, and B. Liu, Accurate particle position measurement from images, *Rev. Sci. Instrum.* **78**, 053704 (2007).
- [44] S. Burov, P. Figliozzi, B. Lin, S. A. Rice, N. F. Scherer, and A. R. Dinner, Single-pixel interior filling function approach for detecting and correcting errors in particle tracking, *Proc. Natl. Acad. Sci. USA* **114**, 221 (2017).
- [45] Y. Yifat, N. Sule, Y. Lin, and N. F. Scherer, Analysis and correction of errors in nanoscale particle tracking using the

- single-pixel interior filling function (SPIFF) algorithm, *Sci. Rep.* **7**, 16553 (2017).
- [46] K.-B. Chai and P. M. Bellan, Vortex motion of dust particles due to non-conservative ion drag force in a plasma, *Phys. Plasmas* **23**, 023701 (2016).
- [47] U. Konopka, D. Samsonov, A. V. Ivlev, J. Goree, V. Steinberg, and G. E. Morfill, Rigid and differential plasma crystal rotation induced by magnetic fields, *Phys. Rev. E* **61**, 1890 (2000).
- [48] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, Scikit-learn: Machine learning in Python, *J. Mach. Learn. Res.* **12**, 2825 (2011).
- [49] M. Himpel and A. Melzer, Configurational temperature in dusty plasmas, *Phys. Rev. E* **99**, 063203 (2019).
- [50] Z. Ding, K. Qiao, J. Kong, L. S. Matthews and T. W. Hyde, Non-linear response of vertical paired structure in complex plasma, *Plasma Phys. Control. Fusion* **61**, 055004 (2019).
- [51] J. Kong, K. Qiao, L. S. Matthews, and T. W. Hyde, Interaction force in a vertical dust chain inside a glass box, *Phys. Rev. E* **90**, 013107 (2014).
- [52] O. Ishihara and S. V. Vladimirov, Wake potential of a dust grain in a plasma with ion flow, *Phys. Plasmas* **4**, 69 (1997).
- [53] V. S. Nikolaev and A. V. Timofeev, Screening length in dusty plasma crystals, *J. Phys.: Conf. Ser.* **774**, 012172 (2016).
- [54] H. M. Mott-Smith and I. Langmuir, The Theory of Collectors in Gaseous Discharges, *Phys. Rev.* **28**, 727 (1926).
- [55] X.-Z. Tang and G. L. Delzanno, Orbital-motion-limited theory of dust charging and plasma response, *Phys. Plasmas* **21**, 123708 (1994).
- [56] Y. Saitou, A simple method of dust charge estimation using an externally applied oscillating electric field, *Phys. Plasmas* **25**, 073701 (2018).
- [57] S. A. Khrapak and G. E. Morfill, An interpolation formula for the ion flux to a small particle in collisional plasmas, *Phys. Plasmas* **15**, 114503 (2008).