MACHINE LEARNING
Science and Technology

PAPER • OPEN ACCESS

Neural force functional for non-equilibrium many-body colloidal systems

To cite this article: Toni Zimmermann *et al* 2024 *Mach. Learn.: Sci. Technol.* **5** 035062

MACHINE
LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
4 June 2024

REVISED
1 August 2024

ACCEPTED FOR PUBLICATION
20 August 2024

PUBLISHED
2 September 2024

PAPER

Neural force functional for non-equilibrium many-body colloidal systems

Toni Zimmermann, Florian Sammüller, Sophie Hermann, Matthias Schmidt and Daniel de las Heras*

Theoretische Physik II, Physikalisches Institut, Universität Bayreuth, D-95447 Bayreuth, Germany

* Author to whom any correspondence should be addressed.

E-mail: delasheras.daniel@gmail.com

Keywords: neural force functional, non-equilibrium colloids, Brownian dynamics

Supplementary material for this article is available online

Abstract

We combine power functional theory and machine learning to study non-equilibrium overdamped many-body systems of colloidal particles at the level of one-body fields. We first sample in steady state the one-body fields relevant for the dynamics from computer simulations of Brownian particles under the influence of randomly generated external fields. A neural network is then trained with this data to represent locally in space the formally exact functional mapping from the one-body density and velocity profiles to the one-body internal force field. The trained network is used to analyse the non-equilibrium superadiabatic force field and the transport coefficients such as shear and bulk viscosities. Due to the local learning approach, the network can be applied to systems much larger than the original simulation box in which the one-body fields are sampled. Complemented with the exact non-equilibrium one-body force balance equation and a continuity equation, the network yields viable predictions of the dynamics in time-dependent situations. Even though training is based on steady states only, the predicted dynamics is in good agreement with simulation results. A neural dynamical density functional theory can be straightforwardly implemented as a limiting case in which the internal force field is that of an equilibrium system. The framework is general and directly applicable to other many-body systems of interacting particles following Brownian dynamics.

1. Introduction

Analyzing how a many-body system reacts to controlled stimuli offers a means to understand its collective behavior emerging from interparticle interactions. Soft matter systems respond to several types of external fields [1–3] such as e.g. electric [4, 5], magnetic [6, 7], gravitational [8–10], optical [11], and mechanical [12, 13] fields. The response of the many-body system to the external perturbation can be highly non-trivial, particularly in non-equilibrium situations. A detailed case-by-case analysis is often required to rationalize the complex dynamics of the system.

Due to the large number of microscopic degrees of freedom in a many-body system, coarse-graining [14] is necessary in order to define a reduced set of relevant variables that encapsulates the physics of the system. Averaging over the many-body probability distribution function yields an exact one-body force balance equation in overdamped Brownian [15] as well as in inertial classical [16] and quantum [17] systems. The one-body fields remain sharply resolved in both space and time [18]. The force balance equation combined with a continuity equation describes the microscopic dynamics of the system at the one-body level.

Depending on the underlying particle dynamics, different contributions appear in the force balance equation [18]. In overdamped Brownian dynamics, the only unknown contribution is the one-body internal force field, $\mathbf{f}_{\text{int}}(\mathbf{r},t)$, originated by an average of the interparticle interactions resolved in position, $\mathbf{r}$, and time, t. The response of the system to an arbitrary external force field can be determined provided that the internal force field is known.

In equilibrium, classical density functional theory (DFT) [19] is the reference framework to build approximations for the one-body internal force field. DFT establishes a mapping from the equilibrium density profile $\rho(\mathbf{r})$ to the internal force field $\mathbf{f}_{\text{int}}(\mathbf{r};[\rho])$ via the one-body direct correlation functional [20] (we indicate with square brackets the functional dependence of the internal force field on the density profile). Several works [21–28] have demonstrated that machine learning is a reliable technique to construct density functionals. An important difference among these works is the dataset used to train the machine learning model. These include pairs of external potentials and density profiles [22, 24, 25] as well as radial distribution functions of homogeneous fluids [28]. Recently, we have shown [26] that a neural network can be efficiently employed to represent the equilibrium functional mapping from the density profile to the direct correlation functional. The mapping can then be used to construct a neural functional theory [26], which for a hard-sphere system outperforms the most sophisticated analytical density functionals [29]. Two- and three-body correlation functions as well as free energy values are accessible via functional calculus implemented with automatic differentiation and functional line integration.

Power functional theory (PFT) demonstrates that fundamental mappings also exist in overdamped and inertial classical and quantum many-body systems in non-equilibrium [18]. For overdamped Brownian systems, there is a formally exact kinematic mapping from both the one-body density $\rho(\mathbf{r},t)$ and velocity $\mathbf{v}(\mathbf{r},t)$ profiles to the internal force field $\mathbf{f}_{\text{int}}(\mathbf{r},t;[\rho,\mathbf{v}])$ [15]. The functional dependence of $\mathbf{f}_{\text{int}}$ is on the whole time-history of both fields, $\rho(\mathbf{r},t)$ and $\mathbf{v}(\mathbf{r},t)$, until the current time t. Previous works have constructed analytical approximations to the kinematic mapping [30], revealing how the non-equilibrium internal force field is responsible for notable phenomena in colloidal systems. These include shear migration [31, 32], the emergence of viscosity and structural forces [33], lane formation [34], the governing mechanisms of the time evolution of the van Hove function [35], and the mobility induced phase separation [36] and freezing [37] in active Brownian particles.

In a recent perspective about dynamical density functional theory [38], we showed that a neural network can accurately represent the functional kinematic mapping from the density and the velocity profiles to the internal force field. We refer here to the trained network as the neural force functional. As a proof of concept, we trained a network using only bulk flows (i.e. $\nabla\cdot\mathbf{v}\neq 0$ and $\nabla\times\mathbf{v}=0$) in which the external force points only along one Cartesian direction. In this type of flow, the non-conservative contribution of the one-body external field is limited to a uniform constant force. Here, we demonstrate that using data augmentation and a local learning approach [26], the mapping can be efficiently learnt in general planar geometry (that is, for flows exhibiting non-vanishing curl and non-vanishing divergence of the velocity field). Furthermore, we showcase the network's predictive capability in several applications. We generate the one-body fields directly from particle-based computer simulations of overdamped isotropic colloidal particles in steady state under the influence of randomly generated external force fields. The particles interact with each other via a Lennard-Jones potential. We then train a neural network to represent the kinematic mapping. The internal force field is learnt locally in space. As a result, the network can be applied to systems of variable size. We show several applications that use the neural force functional combined with the exact force balance equation: (i) splitting and analysing non-equilibrium superadiabatic forces, (ii) performing inverse design by finding the external force field that generates the desired dynamical response of the many body-system (custom flow [39]), and (iii) quantifying non-equilibrium transport coefficients. Finally, we demonstrate that the network is capable to generalize to systems of any length (beyond the size of the simulation box) and to full non-equilibrium situations (beyond steady-states).

Our neural force functional is computationally efficient and delivers results with precision close to simulation data. The capability to process systems of size much larger than that of the training data opens a route to study the dynamics of macroscopic systems with microscopic resolution at near simulation quality.

2. Theory

2.1. Force balance equation

We consider a classical system of N interacting particles suspended in a solvent and following overdamped dynamics. The equation of motion for the ith particle is

$$\gamma\frac{\mathrm{d}\mathbf{r}_i(t)}{\mathrm{d}t}=\boldsymbol{\eta}_i(t)-\nabla_i u\left(\mathbf{r}^N\right)+\mathbf{f}_{\text{ext}}(\mathbf{r}_i,t), \tag{1}$$

where γ is the friction coefficient against the implicit solvent, $\mathbf{r}_i$ is the position of the particle, $\boldsymbol{\eta}_i(t)$ is a Gaussian random force acting at time t on the ith particle, $\mathbf{f}_{\text{ext}}$ is an external force field, ∇_i indicates the derivative with respect to $\mathbf{r}_i$, and

$$u\left(\mathbf{r}^N\right) = \sum_i \sum_{i<j} \phi\left(r_{ij}\right) \tag{2}$$

is the total potential energy of microstate $\mathbf{r}^N = \mathbf{r}_1, \ldots, \mathbf{r}_N$ with $\phi(r_{ij})$ being the interparticle pair potential, r_{ij} being the distance between particles i and j, and the first sum runs over all particles. Although for simplicity only two-body potentials are considered here, PFT [18] is directly applicable to many-body interparticle potentials (see an example in [32]).

One-body fields are obtained as averages of microscopic operators [18]. For example, the one-body density profile is

$$\rho\left(\mathbf{r}, t\right) = \left\langle \sum_i \delta\left(\mathbf{r} - \mathbf{r}_i\right) \right\rangle, \tag{3}$$

with $\delta(\cdot)$ being the Dirac distribution. The angular brackets $\langle\cdot\rangle$ denote an average that in non-equilibrium is performed at each time t over an ensemble of systems. The systems differ in their initial microstate and also in the realization of the random forces, i.e. $\boldsymbol{\eta}_i$ in equation (1). Alternatively, in steady state and in equilibrium, the average can be evaluated over time.

The one-body fields are related via an exact one-body force balance equation [15, 18]

$$\gamma \mathbf{v}\left(\mathbf{r}, t\right) = \mathbf{f}_{\mathrm{id}}\left(\mathbf{r}, t\right) + \mathbf{f}_{\mathrm{int}}\left(\mathbf{r}, t\right) + \mathbf{f}_{\mathrm{ext}}\left(\mathbf{r}, t\right), \tag{4}$$

with $\mathbf{v}(\mathbf{r}, t)$ being the one-body velocity field at position $\mathbf{r}$ and time t. The first contribution on the right hand-side of equation (4) is the thermal diffusive term, which is known exactly:

$$\mathbf{f}_{\mathrm{id}}\left(\mathbf{r}, t\right) = -k_B T \nabla \ln \rho\left(\mathbf{r}, t\right). \tag{5}$$

Here k_B is the Boltzmann constant and T is (absolute) temperature.

The internal force field, $\mathbf{f}_{\mathrm{int}}(\mathbf{r}, t)$, follows from the internal force density field $\mathbf{F}_{\mathrm{int}}(\mathbf{r}, t) = \rho(\mathbf{r}, t)\mathbf{f}_{\mathrm{int}}(\mathbf{r}, t)$, which is simply

$$\mathbf{F}_{\mathrm{int}}\left(\mathbf{r}, t\right) = -\left\langle \sum_i \delta\left(\mathbf{r} - \mathbf{r}_i\right) \nabla_i u\left(\mathbf{r}^N\right) \right\rangle. \tag{6}$$

The last contribution on the right hand-side of equation (4) is the external force field, $\mathbf{f}_{\mathrm{ext}}(\mathbf{r}, t)$.

For overdamped Brownian particles following the equation of motion (1), the force balance equation (4) is exact and valid in general non-equilibrium situations. In non-equilibrium steady state and in equilibrium, the one-body fields become independent of time. Additionally, in equilibrium, the one-body velocity field vanishes everywhere. A detailed derivation of the force balance equation can be found e.g. in [18].

The one-body density profile is linked to the one-body current profile, $\mathbf{J}(\mathbf{r}, t) = \rho(\mathbf{r}, t)\mathbf{v}(\mathbf{r}, t)$, via the continuity equation

$$\dot{\rho}\left(\mathbf{r}, t\right) = -\nabla \cdot \mathbf{J}\left(\mathbf{r}, t\right), \tag{7}$$

where the overdot denotes a partial time derivative. Hence, from a theoretical point of view, the force balance equation (4) can be used in combination with the continuity equation (7) to find the time evolution of the one-body fields $\rho(\mathbf{r}, t)$ and $\mathbf{v}(\mathbf{r}, t)$. However, we first need to approximate the unknown internal force field.

In equilibrium, DFT establishes that the internal force field is a functional of the one-body density profile via

$$\mathbf{f}_{\mathrm{int}}\left(\mathbf{r}; [\rho]\right) = -\nabla \frac{\delta F_{\mathrm{exc}}\left[\rho\right]}{\delta \rho\left(\mathbf{r}\right)}. \tag{8}$$

Here, $F_{\mathrm{exc}}[\rho]$ is the excess (over ideal gas) free energy functional. Hence, knowledge of the equilibrium density profile suffices to determine the internal force field (provided that $F_{\mathrm{exc}}[\rho]$ is known).

Dynamical density functional theory (DDFT) [19, 40] (for a recent and exhaustive review see [41]) approximates the non-equilibrium internal force field by that of an equilibrium system via equation (8). However, there are several concerns that challenge the reliability of this adiabatic approximation [38]. A mayor one follows directly from equation (8): the one-body internal force field in non-equilibrium contains conservative and non-conservative contributions but DDFT predicts always an equilibrium-like force field which is therefore conservative. The non-conservative contribution of the internal force field can be important. For example, colloidal migration in shear fields [33, 42, 43] and lane formation in oppositely driven mixtures [5, 34, 44] are physical phenomena that can be rationalized on the basis of the non-conservative contributions of the non-equilibrium internal force field.

2.2. Power functional theory

To overcome the limitations of DDFT we use PFT [15, 18] which is based on a formally exact variational principle for non-equilibrium many-body systems. In PFT, a functional (with units of power) is minimized with respect to the current profile (or the velocity profile) at fixed density profile and fixed time. The functional is constructed such that the associated Euler-Lagrange equation is the exact force balance equation (4). The functional depends functionally not only on the density (as it is the case in equilibrium DFT) but also on the velocity profile. Formally, the dependence is not instantaneous on time but rather on the complete history of both fields. The original formulation of PFT was done using a functional of ρ and $\mathbf{J}$ [15]. However, working with ρ and $\mathbf{v}$ turns out to be more convenient to construct analytical approximations, see e.g. [30, 33] and section 3.4.

In PFT the internal force field is split into adiabatic and superadiabatic contributions

$$\mathbf{f}_{\text{int}}\left(\mathbf{r},t;[\rho,\mathbf{v}]\right)=\mathbf{f}_{\text{ad}}\left(\mathbf{r},t;[\rho]\right)+\mathbf{f}_{\text{sup}}\left(\mathbf{r},t;[\rho,\mathbf{v}]\right). \tag{9}$$

The adiabatic contribution is defined as the internal force field of a virtual equilibrium system with the same density profile as the non-equilibrium system. Hence, the adiabatic contribution can be formally obtained via equation (8) and it is at each time t a functional of the instantaneous one-body density, $\rho(\mathbf{r},t)$, only. In contrast, the genuine non-equilibrium superadiabatic contribution is given by the functional derivative of the excess power functional $P_t^{\text{exc}}[\rho,\mathbf{v}]$

$$\mathbf{f}_{\text{sup}}\left(\mathbf{r},t;[\rho,\mathbf{v}]\right)=-\frac{1}{\rho\left(\mathbf{r},t\right)}\frac{\delta P_t^{\text{exc}}[\rho,\mathbf{v}]}{\delta\mathbf{v}\left(\mathbf{r},t\right)}. \tag{10}$$

The superadiabatic force is a functional of both ρ and $\mathbf{v}$ since it inherits the functional dependencies of the generating functional. Equation (10) is formally exact but it requires knowledge of the excess power functional. It is possible to construct analytical approximations to the excess power functional using an expansion in powers of the velocity field [30]. Here, we follow a different approach. Instead of finding approximations for $P_t^{\text{exc}}[\rho,\mathbf{v}]$, we machine learn the relation $\{\rho,\mathbf{v}\}\rightarrow\mathbf{f}_{\text{int}}$ that maps both the density and the velocity profile to the internal force field. We learn the complete internal force field, i.e. both superadiabatic and adiabatic contributions, since the splitting can be done *a posteriori* as we show in section 3.2.

The first step in any machine learning application is the generation of the training set that we discuss in the following.

2.3. Simulations and training set

We employ adaptive Brownian dynamics simulations [45] to integrate the many-body equations of motion (1) over time and to generate the data of the training set. All terms contributing to the force balance equation (4) can be obtained from many-body computer simulations. The velocity profile can be sampled either directly via the central difference derivative of the position vector with respect to time or indirectly via the force balance equation. For details see e.g. the appendix of [39]. The training set contains the one-body density and the velocity profiles as input fields, and the one-body internal force as the output field.

We simulate N Lennard-Jones particles in a cubic simulation box with length $L/\sigma=10$ and periodic boundary conditions. Here, σ is the length scale of the Lennard-Jones potential. We use a cutoff distance for the interparticle potential $r_c/\sigma=2.5$. The energy parameter of the Lennard-Jones potential ϵ acts as our energy scale. Our time scale is $\tau=\sigma^2\gamma/\epsilon$ and we work at constant (supercritical) temperature $k_BT/\epsilon=1.5$.

The number of particles N is chosen uniformly in the interval $200\leqslant N\leqslant 800$. Hence, the bulk density in the training set varies within the interval $0.2\leqslant\rho_b\sigma^3\leqslant 0.8$. The particles are randomly initialized in the simulation box and then equilibrated in bulk (without external force) for a total time $t/\tau=10$. We then switch on an external force and wait 100τ for the system to reach a steady state. As shown in the appendix of [38], the initial state has no influence at all on the final steady state (note that we consider ergodic fluids only and stay away from phase transitions). The particle initialization affects only the dynamical path followed by the system towards the steady state (provided that the system is ergodic). Once the system is in steady state, we sample the one-body fields of interest as an average over time during at least $3\cdot 10^3\tau$.

The external force field is generated randomly according to

$$\mathbf{f}_{\text{ext}}\left(z\right)=\sum_{\alpha}f_{\text{ext},\alpha}\left(z\right)\hat{\mathbf{e}}_{\alpha}, \tag{11}$$

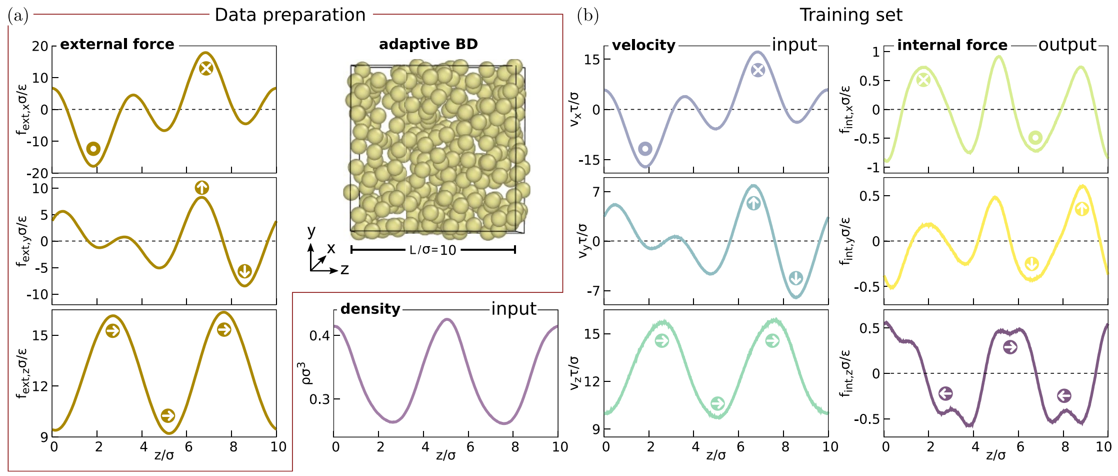

Figure 1. Training set: from particles to fields. (a) Components of a randomly generated external force as a function of the z coordinate and characteristic snapshot of the system in the corresponding steady state. We generate data for the one-body fields from the many-body system by sampling the steady state which results from applying a randomly generated external force field. (b) One-body fields involved in the training set as sampled in the simulations. The fields correspond to the steady state of the external force shown in panel (a). Input fields: density ρ and velocity $\mathbf{v}$ profiles as a function of z. Output field: internal force $\mathbf{f}_{\mathrm{int}}$ as a function of z. Zero baselines in plots of vector components are indicated with horizontal dash-lines. The colored arrows indicate the direction at selected positions of the component of the vector field.

with $f_{\mathrm{ext},\alpha}$ being the Cartesian α-component of the external force field and $\hat{\mathbf{e}}_\alpha$ the unit vector along the α-direction. To make the model as general as possible while avoiding the sampling of multidimensional data, we use external force fields with components along the three spatial directions but allow inhomogeneity to occur only along the z-direction ($\hat{\mathbf{e}}_z$). This planar geometry allows us to work solely with one-dimensional histograms, which eases sampling, data preparation and neural network construction. Each component of the external force field is generated randomly via superposition of Fourier modes

$$f_{\mathrm{ext},\alpha}(z) = a_0^\alpha + \sum_{m=1}^{M_\alpha} a_m^\alpha \sin\left(\frac{2\pi z}{L} k_m^\alpha + \psi_m^\alpha\right). \tag{12}$$

The coefficient a_0^α, which plays a major role determining the average current, is chosen uniformly in the interval $[0, 50\epsilon/\sigma]$. The maximum number of superimposed Fourier modes for component α is M_α which is randomly selected between one and four. The amplitudes a_m^α and the phases ψ_m^α of each Fourier mode are chosen uniformly; the phases within the interval $[0, 2\pi)$ and the amplitudes within the interval $[0, 30\epsilon/\sigma]$ for $\alpha = x, y$ and $[0, 4\epsilon/\sigma]$ for $\alpha = z$. To reduce ergodicity problems during the generation of the training set, the maximum amplitude is smaller along the inhomogeneous direction. The parameters k_m^α are randomly selected integers ranging from one to four.

We then generate several external forces according to equation (12) and run the corresponding simulations. Approximately five percent of the simulations were discarded since they contained regions in which the density was locally very low (smaller than $0.01\sigma^3$) which could be an indication of the system not being ergodic. This can happen if e.g. the external force varies strongly in a small region effectively trapping the particles. We complement the training set with that of [38], which consists of 1000 simulations in which the external force is generated according to equation (12) but where the only non-vanishing external force contribution is the z-component, i.e. $\mathbf{f}_{\mathrm{ext}} = f_{\mathrm{ext},z}(z)\hat{\mathbf{e}}_z$. Roughly ten percent of this subset of simulations corresponds to the relevant case of equilibrium systems for which $\mathbf{f}_{\mathrm{ext}}$ has only a z-component and $a_0^z = 0$. Having equilibrium profiles in the training set is relevant to e.g. split the internal force field into adiabatic and superadiabatic components [46].

We illustrate in figure 1 the data preparation and sampling of the one-body fields with one example of the training set. The one-body fields are sampled in z-direction using histograms with bin size 0.01σ. The final training set contains 2897 individual simulation results which we split as usual into training (1877), validation (868), and test (152) sets. The accuracy of the predictions depends on the amount of data in the training set. The approximate size of the data set was decided based on our experience with previous systems [26, 38]. We have checked that halving the dataset, increases the value of the cost function evaluated on the test data by approximately one order of magnitude. The complete dataset and the trained model are deposited in Zenodo [47]. After training, additional simulation data is obtained to compare with the predictions of the neural network for selected systems (see section 3).

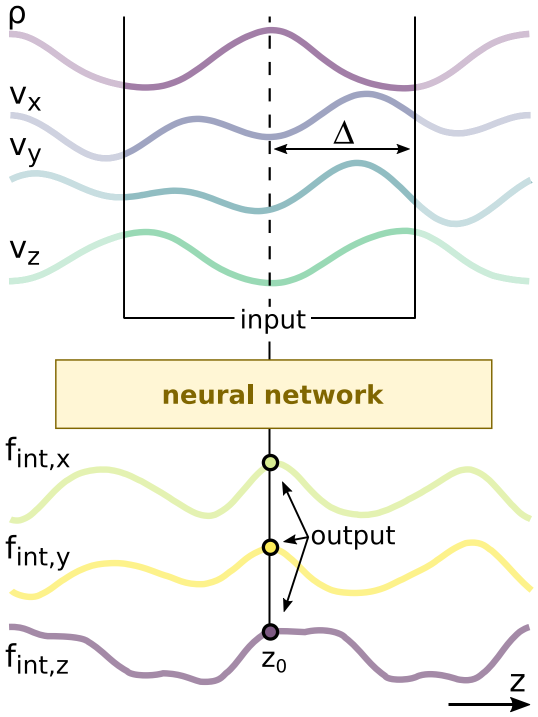

Figure 2. Schematic of the local learning approach. The input fields are the density profile ρ and the three components of the velocity profile $\mathbf{v}$ in an interval centered at z_0 and half-width $\Delta = 2.8\sigma$. The output of the network are the three components of the internal force field at position z_0. The kinematic map is represented with a convolutional neural network.

2.4. Data augmentation

We use coordinate transformations which keep the non-equilibrium physics unchanged to augment the training data. For instance, one can flip the z-axis, i.e. perform the transformation $z \to -z$, and hence duplicate the training data by changing the original data accordingly (which in this case also implies flipping the sign of the z-components of the vector fields $\mathbf{v}$ and $\mathbf{f}_{\text{int}}$). We use eight coordinate systems: the original one, three in which we flip only one axis, other three in which we flip two axes, and one with all axes flipped. Moreover, since the system is inhomogeneous along the z-direction only, we can interchange the x and y-components of $\mathbf{v}$ and $\mathbf{f}_{\text{int}}$. Applying this symmetry to each of the eight coordinate systems results in sixteen possible coordinate transformations to augment our training data.

Data augmentation serves two purposes. It expands the number of input-output pairs available for training (no data augmentation is used in either the validation or the test sets) and consequently the accuracy of the predictions. For example, a neural network trained with half of the data plus data augmentation is approximately as accurate as a network trained with the full training set but without data augmentation. More importantly, data augmentation indirectly imposes the physically relevant symmetries of our system into the neural network. Alternatively, it might be possible to incorporate the symmetries of the system using equivariant neural networks [48] and other physics-informed machine learning techniques [49].

2.5. Local learning

The internal force field at a given position z_0 is fully determined by the density and the velocity profiles in the neighbourhood of that position. This allows us to use a local learning approach, similar to the one used by Sammüller *et al* [26] for equilibrium systems. We use the network to represent the kinematic mapping at position z_0, see figure 2. That is, the output of the neural network is $\mathbf{f}^{\star}_{\text{int}}(z_0)$. We use here and in the following the superscript $\star$ to indicate quantities that have been obtained with the neural force functional. The input fields of the network are ρ and $\mathbf{v}$ in an interval of width 2Δ centered at z_0. Here, we use $\Delta = 2.8\sigma$ which produces optimal results for our training set. Overfitting starts to occur for larger values. On the other hand, the accuracy of the predictions rapidly deteriorates for values of Δ smaller than one particle size, i.e. $\Delta \lesssim \sigma$. The optimal value of Δ might depend on several variables including the interparticle potential, the temperature, and the range of currents in the training set.

Instead of one input-output training sample per simulation, the local learning strategy facilitates to use the data of each histogram bin (10^3 per simulation) individually, thereby increasing data efficiency substantially. Additionally, data augmentation multiplies the number of samples by a factor sixteen, see section 2.4. Hence, from the original 1887 simulations used for training we generate approximately $3 \cdot 10^7$ local input-output pairs.

This local learning approach is crucial since (i) it simplifies the learning process, (ii) it imposes the short-ranged spatial dependence of the underlying functional by construction, and (iii) it allows us to apply the network to systems much larger than the original simulation box. The implementation of local learning eliminates the intrinsic length scale constraint dictated by the size of the simulation box. Note that local learning is possible due to the specific form of the mapping that is represented. Local learning is not possible if one learns e.g. the relation between $\mathbf{f}_{\mathrm{int}}$ and $\mathbf{f}_{\mathrm{ext}}$ since the internal force field at a given position depends on the global shape of the external field.

We use a convolutional neural network with three convolutional layers, see details in appendix. PFT [15, 18] guarantees the existence of the kinematic functional map from the velocity and the density profiles to the internal force field. In time-dependent situations, the map involves the history of the density and velocity fields. Here, we restrict the study to stationary steady-states. Hence, the input data of our network comprises only the required elements to generate the output, without any superfluous or lacking components. We believe this contributes to a successful learning process and makes it possible to efficiently learn a non-trivial problem in statistical physics with a relatively simple network architecture (compared with cutting-edge models).

3. Results

After training, the network acts as a neural force functional representing the kinematic functional mapping from the density and velocity profiles to the internal force field. Combined with the force balance and the continuity equations, the neural force functional allows us to study the many-body dynamics at the one-body level. We show several applications in this section.

3.1. Neural custom flow

Custom flow is a numerical method to find the external force field that generates the desired dynamics (prescribed by the one-body density and velocity profiles). The method has been developed for Brownian [39] and Newtonian [50] many-body systems. We have used custom flow to design flows that facilitate the analysis of the superadiabatic forces [33, 51] and to perform the adiabatic construction [46] (that is, to find the conservative potential that generates in equilibrium the same density profile as that in an out-of-equilibrium system). The adiabatic construction can be used to split the internal force into adiabatic and superadiabatic contributions, see equation (9).

Custom flow, which is an example of the growing field of inverse design in statistical physics [52–54], is computationally expensive. The method requires to run several simulations to iteratively find the precise form of the external force field (up to the imposed numerical tolerance). In contrast, with the network we perform the same task instantly. We fix $\rho(\mathbf{r})$ and $\mathbf{v}(\mathbf{r})$ in steady-state and use the neural network to infer the corresponding $\mathbf{f}^{\star}_{\mathrm{int}}(\mathbf{r};[\rho,\mathbf{v}])$. Then, using the exact force balance equation (4), we solve for the external force field

$$\mathbf{f}^{\star}_{\mathrm{ext}}(\mathbf{r}) = \gamma \mathbf{v}(\mathbf{r}) + k_B T \nabla \ln \rho(\mathbf{r}) - \mathbf{f}^{\star}_{\mathrm{int}}(\mathbf{r}). \tag{13}$$

An example of inverse design using custom flow is shown in figure 3. Generalizing one of the flows considered in [38], we prescribe a density profile with a kink, see figure 3(a), and also include kinks in the x- and y-components of $\mathbf{v}$, see figure 3(b). In planar geometry, the continuity equation (7) imposes that in steady state the z-component of $\mathbf{v}$ is determined up to a multiplicative constant by the density profile

$$v_z(z) = \frac{J_0}{\rho(z)}, \tag{14}$$

with J_0 being the magnitude of the steady current. The corresponding external force is shown in figure 3(c). To test the validity of the method, we next run Brownian dynamics simulations using the predicted external force profile and sample the one-body fields, see the symbols (simulations) and solid-lines (neural force functional) in figure 3. Both sets of profiles agree well, specially considering that this is a demanding test because profiles with kinks were not included during training. Moreover, when comparing the internal force fields obtained with BD and the neural force functional there are two main sources of error. First, the network produces an error calculating the internal force field. Second, due to this error in $\mathbf{f}^{\star}_{\mathrm{int}}$, the external force calculated with the force balance equation (13) is not exactly the one that generates the prescribed kinematic profiles in simulations. Hence, the internal force field sampled in the simulations does not exactly correspond to the internal force field of the prescribed kinematic fields.

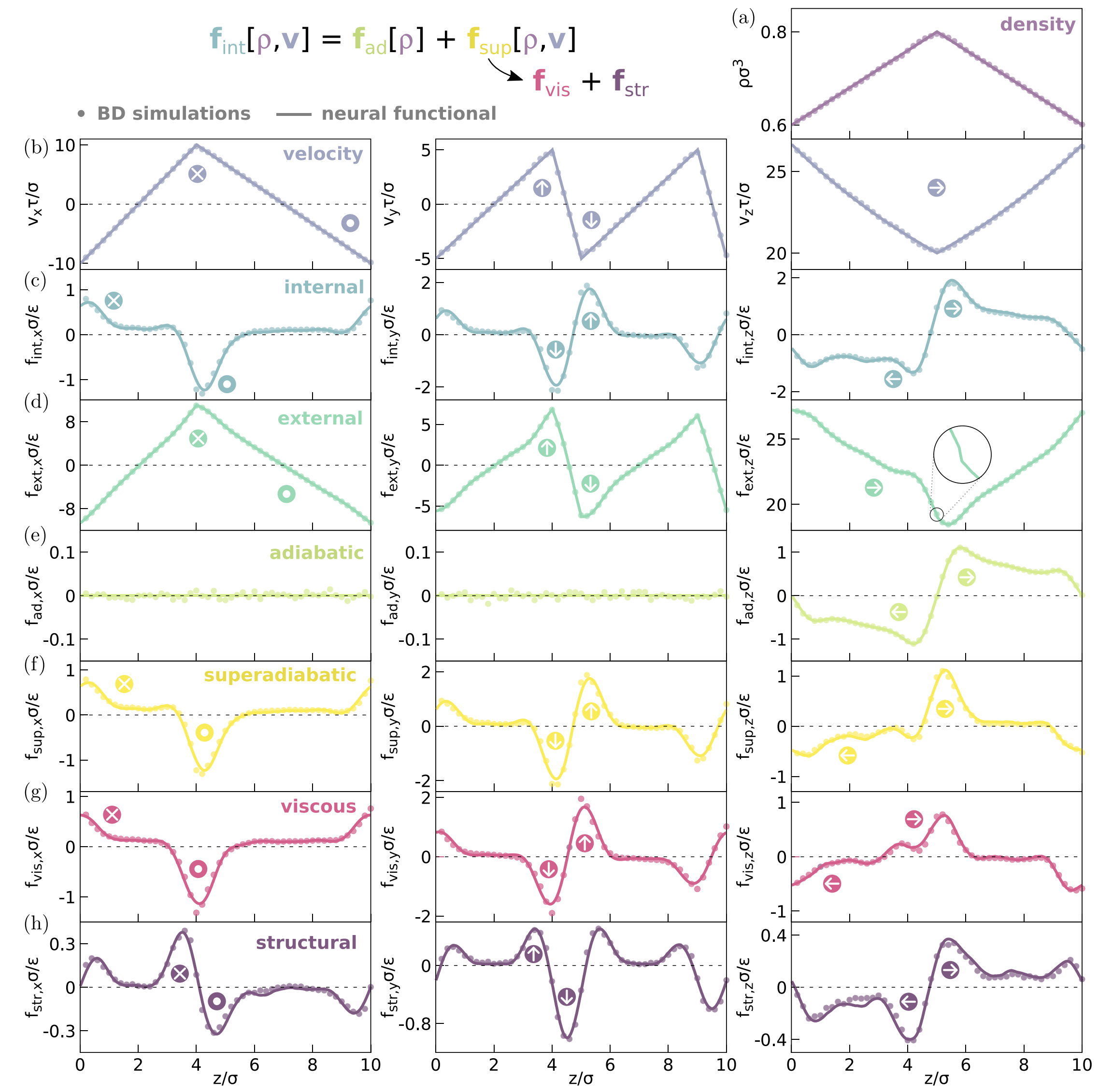

Figure 3. Inverse design via neural custom flow and superadiabatic forces. One-body fields according to the neural force functional (solid lines) and simulations (symbols) as a function of z: (a) density ρ, (b) velocity $\mathbf{v}$, (c) internal force $\mathbf{f}_{\text{int}}$, (d) external force $\mathbf{f}_{\text{ext}}$, (e) adiabatic force $\mathbf{f}_{\text{ad}}$, (f) superadiabatic force $\mathbf{f}_{\text{sup}}$, (g) viscous superadiabatic force $\mathbf{f}_{\text{vis}}$, (h) structural superadiabatic force $\mathbf{f}_{\text{str}}$. The first, second and third column correspond to the x-, y-, and z-components of the respective vector fields. The arrows indicate the direction at selected positions of the component of the vector field. The horizontal dashed lines indicate the zero baseline of the vector components. The inset in (d) is an enhanced view of the region in which $\mathbf{f}_{\text{ext}}$ is discontinuous.

The kink in the density profile creates a discontinuity in the ideal gas diffusive term of the force balance equation, see equation (5). The discontinuity can only be balanced by the external force, see equation (13) and figure 3(d), because both the velocity and the internal force field cannot be discontinuous. This is another advantage of having the internal force as the output field since its properties remain well-behaved even for demanding input fields (besides the benefit of being able to use a local learning approach).

3.2. Superadiabatic forces

To split the internal force field into adiabatic and superadiabatic components, we simply evaluate the neural force functional to yield the internal force field in equilibrium ($\mathbf{v} = 0$) which corresponds to the adiabatic force

$$\mathbf{f}^{\star}_{\text{ad}}\left(\mathbf{r};[\rho]\right) = \mathbf{f}^{\star}_{\text{int}}\left(\mathbf{r};[\rho,\mathbf{v}=0]\right). \tag{15}$$

The superadiabatic contribution is obtained by subtracting the adiabatic part from the total internal force field

$$\mathbf{f}^{\star}_{\text{sup}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) = \mathbf{f}^{\star}_{\text{int}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) - \mathbf{f}^{\star}_{\text{ad}}\left(\mathbf{r};[\rho]\right). \tag{16}$$

In simulations, we sample the adiabatic force field using the external force field obtained from equation (13) with $\mathbf{f}^{\star}_{\text{ad}}$ as the internal force field and $\mathbf{v} = 0$. The resulting force field is conservative since the adiabatic system is in equilibrium. In figures 3(e) and (f) we compare the adiabatic and superadiabatic contributions obtained with the neural force functional and with simulations. The z-component of the adiabatic force field is the only non-vanishing component because the equilibrium internal force field is a gradient field, see equation (8), and our system is homogeneous along the x- and y-directions.

Viscous and structural components

In steady-state, the superadiabatic force field can be split into viscous (or flow), $\mathbf{f}_{\text{vis}}$, and structural, $\mathbf{f}_{\text{str}}$, components

$$\mathbf{f}_{\text{sup}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) = \mathbf{f}_{\text{vis}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) + \mathbf{f}_{\text{str}}\left(\mathbf{r};[\rho,\mathbf{v}]\right). \tag{17}$$

We outline here only the main ideas and refer the reader to [33] for a complete description. The viscous component responds to the direction of the velocity field and it often opposes the flow. The structural component responds to the shape of the velocity field but not to its direction and it can create structure in the fluid. To split $\mathbf{f}_{\text{sup}}$ we create a reverse steady state in which the flow points in opposite direction, $\mathbf{v}_r(\mathbf{r}) = -\mathbf{v}(\mathbf{r})$, but the density profile remains unchanged, $\rho_r(\mathbf{r}) = \rho(\mathbf{r})$. The subscript r refers to quantities in the reverse steady state. Hence, the viscous component flips sign in the reverse system whereas the structural component remains unchanged. The superadiabatic force field in the reverse system is then

$$\mathbf{f}_{\text{sup},r}\left(\mathbf{r};[\rho_r,\mathbf{v}_r]\right) = -\mathbf{f}_{\text{vis}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) + \mathbf{f}_{\text{str}}\left(\mathbf{r};[\rho,\mathbf{v}]\right). \tag{18}$$

Note that if the superadiabatic force field is expanded in powers of the velocity field, then the odd (even) terms in powers of $\mathbf{v}$ generate viscous (structural) force contributions. From equations (17) and (18) it follows that

$$\mathbf{f}_{\text{vis}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) = \frac{\mathbf{f}_{\text{sup}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) - \mathbf{f}_{\text{sup},r}\left(\mathbf{r};[\rho_r,\mathbf{v}_r]\right)}{2}, \tag{19}$$

$$\mathbf{f}_{\text{str}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) = \frac{\mathbf{f}_{\text{sup}}\left(\mathbf{r};[\rho,\mathbf{v}]\right) + \mathbf{f}_{\text{sup},r}\left(\mathbf{r};[\rho_r,\mathbf{v}_r]\right)}{2}. \tag{20}$$

The superadiabatic force field in the reverse system can be obtained using the neural network according to

$$\mathbf{f}^{\star}_{\text{sup},r}\left(\mathbf{r};[\rho_r,\mathbf{v}_r]\right) = \mathbf{f}^{\star}_{\text{int}}\left(\mathbf{r};[\rho,-\mathbf{v}]\right) - \mathbf{f}^{\star}_{\text{ad}}\left(\mathbf{r};[\rho]\right). \tag{21}$$

The viscous and the structural components of the superadiabatic force field (neural force functional and simulations) are shown in figures 3(g) and (h), respectively. To find the external force field that in the simulations generates the reverse system, we use equation (13) with $\mathbf{f}^{\star}_{\text{int}}(\mathbf{r};[\rho,-\mathbf{v}])$ as the internal force field.

3.3. Transport coefficients

We can also use the neural force functional to effortlessly extract transport coefficients such as the shear and bulk viscosities. We illustrate the process with two model shear and bulk flows.

3.3.1. Shear flow

We prescribe the following spatially periodic shear flow (Kolmogorov flow [55])

$$\rho(z) = \rho_b, \tag{22}$$

$$\mathbf{v}(z) = v_0 \sin(2\pi z/L)\,\hat{\mathbf{e}}_x, \tag{23}$$

with ρ_b and v_0 constants. That is, the density profile is uniform and the velocity profile is a sinusoidal wave along the x-direction. The velocity profile is therefore divergence-free $\nabla\cdot\mathbf{v} = 0$ and it has a non-vanishing curl $\nabla\times\mathbf{v}\neq 0$. The one-body fields are shown in figure 4(a).

We have studied this flow in a two-dimensional system in [33] using custom flow to understand the superadiabatic forces. Also, a closely related flow was analysed in [31, 32]. There, a sinusoidal external force drives the system. For sufficiently weak external driving, the velocity profile closely resembles the sinusoidal shape of the driving force and the density is almost uniform. However, for strong enough driving the velocity profile can differ substantially from a sinusoidal wave, and the particles migrate to the regions of low shear rate generating therefore a density modulation [31, 32].

Here, we impose the density profile to be homogeneous, equation (22). Hence, the entire internal force field is superadiabatic. Another advantage of this flow is the natural splitting of the viscous and the structural

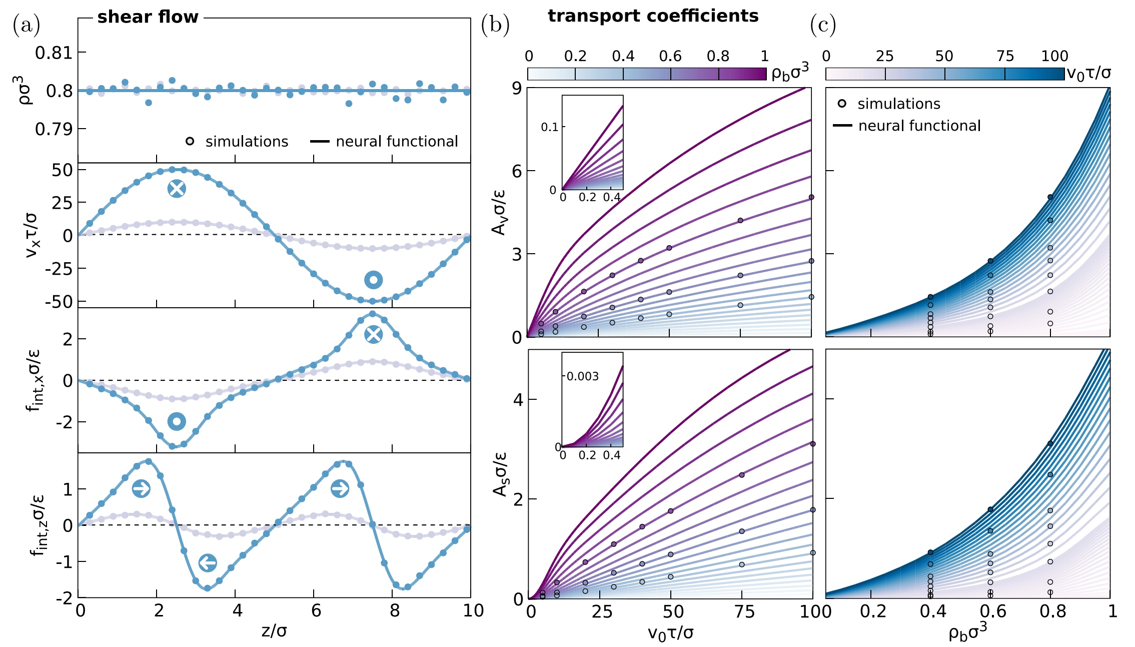

Figure 4. Shear flow. (a) Prescribed density ρ and velocity v_x profiles as well as the predicted viscous ($f_{\mathrm{int,x}}$) and structural ($f_{\mathrm{int,z}}$) components of the internal force field $\mathbf{f}_{\mathrm{int}}$ as a function of z in a model shear flow. Solid lines (symbols) are neural force functional predictions (Brownian dynamics simulations). The average density is set to $\rho_b\sigma^3 = 0.8$ and the imposed flow is a sinusoidal wave of amplitude $v_0\sigma/\tau = 50$ (blue) and 10 (grey). Amplitudes of the viscous (A_v) and the structural (A_s) superadiabatic forces (b) as a function of the strength of the flow v_0 for several values of the average bulk density ρ_b (color bar), and (c) as a function of the bulk density ρ_b for several amplitudes of the flow v_0 (color bar). The amplitudes are obtained as half of the difference between the maximum and the minimum value of the corresponding force. The insets in (b) are a close view of the weak flow regime.

components along different Cartesian directions [33]. The internal force field along the x-direction (parallel to the flow) is viscous and it opposes the flow, see figure 4(a). This component of the force reverses if we reverse the flow. Along the z-direction, perpendicular to the flow, there is additionally a superadiabatic force field which is of structural type. Reversing the flow does not alter this component of the force. If we remove the constraint of uniform density, the structural force field would create a density modulation [31, 32] with peaks around the regions of low shear rate. Note how the structural force field, $\mathbf{f}_{\mathrm{str}}(\mathbf{r}) = f_{\mathrm{int},z}(\mathbf{r})\hat{\mathbf{e}}_z$, tries to move particles from the high ($z/\sigma = 0$ and 5) to the low ($z/\sigma = 2.5$ and 7.5) shear rate regions. An external force field in z-direction balances the structural force such that the density profile remains homogeneous.

To simulate the flow, we use the external force field obtained with neural custom flow (as explained in section 3.1) and then sample the one-body fields. The predictions of the neural force functional agree well with computer simulations, see figure 4(a).

We have studied this flow previously in Brownian [31–33] and Newtonian [51] systems. However, we used either a sinusoidal external force field [31, 32] (in which case the velocity profile is not a perfect sinusoidal wave and there is a density modulation) or custom flow which is computationally demanding [33]. In contrast, using the network it is straightforward to design the flow and to also obtain and analyze the internal force field. For example, we show in figures 4(b) and (c) the amplitudes A_v and A_s of the viscous and structural superadiabatic force fields as a function of the amplitude of the flow v_0 and the bulk density ρ_b, respectively. The amplitude is determined by taking half of the difference between the maximum and minimum values of the superadiabatic force field. These amplitudes are (up to factors of ρ_b) the transport coefficients associated to the viscous and the structural response [31]. In the limit of weak flows $v_0 \to 0$ the viscous force grows linearly with v_0 whereas the structural force grows quadratically, see insets of figure 4(b). This behaviour agrees with the superadiabatic response predicted by an approximated PFT constructed by expanding the excess term in powers of the velocity field [31]. In this geometry, the superadiabatic response generated by the first two terms in the expansion is

$$\rho_b\mathbf{f}_{\mathrm{sup}} = -\eta_s\nabla\times(\nabla\times\mathbf{v}) - \chi\nabla(\nabla\times\mathbf{v})^2\,. \qquad (24)$$

The first term on the right hand side is the viscous component (linear in $\mathbf{v}$) and the second term is the structural component (quadratic in $\mathbf{v}$). The parameters η_s (shear viscosity) and χ are the transport coefficients associated to each superadiabatic force field in the limit of weak flows (recall that we have retained only the first terms of the expansion). Hence, our analysis with the neural force functional confirms

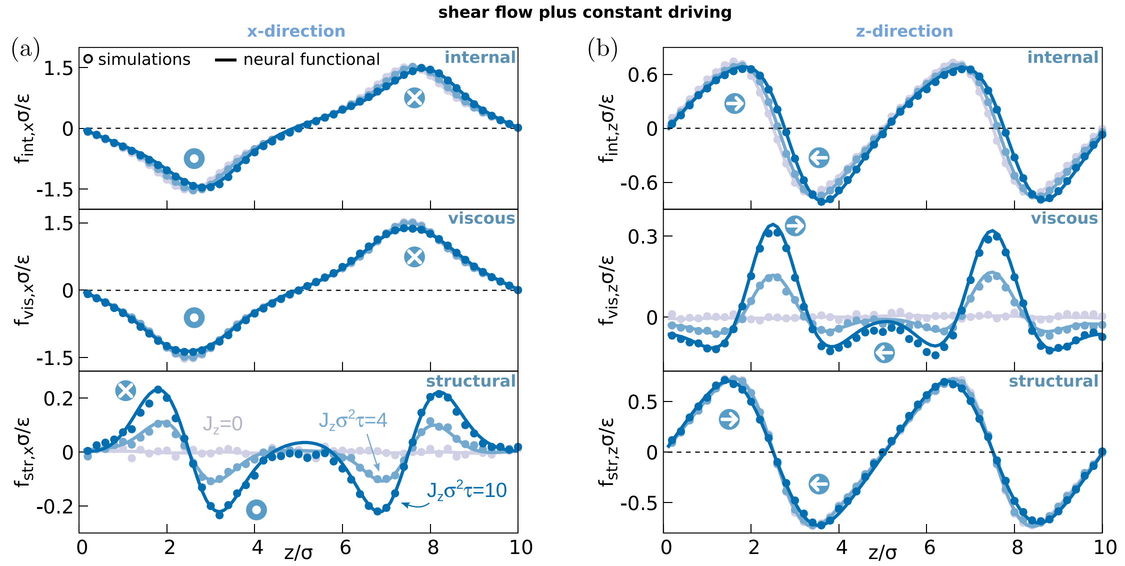

Figure 5. Shear flow plus constant driving. Internal forces in a system with constant density profile, $\rho_b\sigma^3 = 0.7$, following a sinusoidal shear flow along the x-direction, $v_x(z) = v_0 \sin(2\pi z/L)$ with $v_0\tau/\sigma = 25$, and a constant flow along the z-direction $v_z = v_1$ of magnitude $v_1 = 0$ (gray), $J_z\sigma^2\tau = 4$ (light blue), and $J_z\sigma^2\tau = 10$ (dark blue), with $J_z = v_1\rho_b$ the current along the z-direction. The internal force field (top row) is entirely superadiabatic $\mathbf{f}_{\text{int}} = \mathbf{f}_{\text{sup}}$ since ρ is constant. The viscous $\mathbf{f}_{\text{vis}}$ (middle row) and the structural $\mathbf{f}_{\text{str}}$ (bottom row) superadiabatic force fields depend on the value of v_1. The left (right) panels display the x-component (z-component) of the forces.

the suitability of previous analytical approaches. For strong flows the network predicts the saturation of the superadiabatic forces which is also in agreement with previous simulations [56].

3.3.2. Shear plus constant flow

In equilibrium, the neural functional [26] satisfies exact sum rules that follow from Noether invariance [57] even though they have not been imposed during training. In non-equilibrium, the network also complies with symmetries of the underlying physical system that have not been imposed by e.g. data augmentation. For example, we have verified that adding a constant to the x- or y-component of the velocity field in the shear flow shown in figure 4 leaves the internal force field predicted by the network unchanged. This is expected because all one-body fields are homogeneous in both x- and y-directions. Hence, we are simply changing the Galilean reference frame and the internal force field is invariant under such transformation.

The velocity, the internal force field, and the external force field are inhomogeneous along the z-direction. Hence, adding a constant term to the z-component of the velocity is not equivalent to a change between two inertial frames of reference. The network is also able to discern whether adding a constant flow is equivalent to changing to another inertial frame of reference. We show in figure 5 the internal force field and its splitting into viscous and structural components for a shear flow. Like in figure 4, the system is constructed to have constant density and the x-component of the velocity profile is a sinusoidal wave. In addition, there is a constant drift along $\hat{\mathbf{e}}_z$. That is

$$\rho(z) = \rho_b, \tag{25}$$

$$\mathbf{v}(z) = v_0 \sin(2\pi z/L)\,\hat{\mathbf{e}}_x + v_1\hat{\mathbf{e}}_z. \tag{26}$$

Since the density is constant, the entire internal force field is superadiabatic. The presence of the z-component of the velocity clearly alters the internal force field. First, the viscous force along $\hat{\mathbf{e}}_x$ and the structural force along $\hat{\mathbf{e}}_z$ depend on the value of v_1. Additionally, switching on the constant flow along $\hat{\mathbf{e}}_z$ generates a superadiabatic viscous force along $\hat{\mathbf{e}}_z$ and a structural force along $\hat{\mathbf{e}}_x$. Both superadiabatic force fields are not present if v_1 vanishes. We have confirmed with Brownian dynamics simulations that the predictions of the neural force functional closely match the simulation data (compare lines and symbols in figure 5).

A steady state approximation to the excess power functional based on a series expansion in powers of the gradient of the velocity field, $\nabla\mathbf{v}$, can be enough to reproduce the superadiabatic force field accurately [30, 31]. This example shows that there are certain cases in which the mean value of the velocity field is also relevant to determine the superadiabatic forces. For those cases, an approximate analytical excess power functional can contain terms that depend on the velocity field itself, $\mathbf{v}$, such as those included in [33]. For a discussion of exact nonequilibrium sum rules for the non-stationary dynamics, we refer the reader to [57].

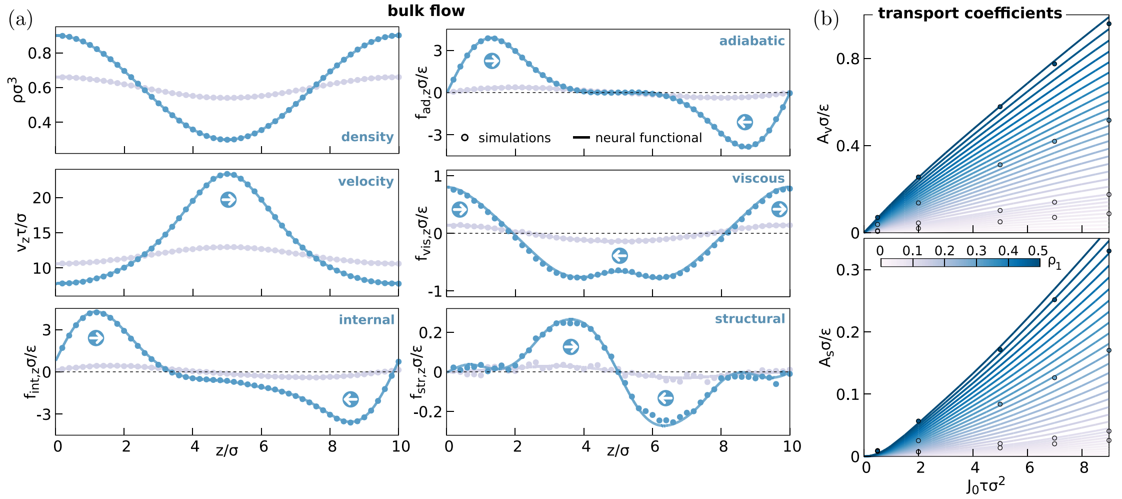

Figure 6. Bulk flow. (a) One-body profiles in a model bulk flow as a function of z for two different density modulations: $\rho_1 = 0.1$ (gray) and $\rho_1 = 0.5$ (blue), see equation (27). In both cases the average density is $\rho_0\sigma^3 = 0.6$ and the current is $J_0\sigma^2\tau = 7$. All the vector fields point along the (gradient) z-direction. Shown are the density ρ, the velocity v_z, the internal force $f_{\mathrm{int},z}$, the adiabatic force $f_{\mathrm{ad},z}$, and the splitting of the superadiabatic force field in viscous $f_{\mathrm{vis},z}$ and structural $f_{\mathrm{str},z}$ components. (b) Transport coefficients of the viscous (A_v) and the structural (A_s) superadiabatic forces as a function of the current J_0 for different values of the density modulation ρ_1 (color bar). Solid lines are neural force functional predictions and symbols are Brownian dynamics simulations.

The Galilean transformation could also be used for data augmentation to potentially improve the quality of the network predictions [58].

3.3.3. Bulk flow

In the shear flow, the homogeneous density profile along with the viscous and structural forces being perpendicular to each other facilitates the analysis of the transport coefficients. Bulk (compressible) flows are more challenging since the flow and viscous components are parallel to each other. Nevertheless, the neural force functional still provides an accurate description of the superadiabatic forces [38] and the transport coefficients for bulk flows. We illustrate this using the following flow

$$\rho(z) = \rho_0\left(1 + \rho_1\cos\left(2\pi z/L\right)\right), \tag{27}$$

$$\mathbf{J}(z) = \rho(z)\,\mathbf{v}(z) = J_0\hat{\mathbf{e}}_z, \tag{28}$$

with constant ρ_0 (average density), ρ_1 (amplitude of density modulation) and J_0 (current). Hence, by construction $\nabla\times\mathbf{v} = 0$ and $\nabla\cdot\mathbf{v} \neq 0$. Since the current is constant, the velocity profile is simply $\mathbf{v}(z) = J_0/\rho(z)\hat{\mathbf{e}}_z$.

The one-body profiles are shown in figure 6(a) and compared to Brownian dynamics simulations. Only the z-component of the internal force field does not vanish. It contains adiabatic and superadiabatic contributions since the density is not homogeneous. Moreover, the superadiabatic contribution incorporates both viscous and structural terms. The complete analysis of the internal force field makes use of the network three times: for the total internal force field, for the splitting into adiabatic and superadiabatic components and to split the superadiabatic force field into viscous and structural components. Each time we obtain an external field via the force balance equation and run a BD simulation to split the forces in simulations. Although each step introduces and accumulates errors, we see in figure 6 that the network predictions remain in good agreement with the simulation results. The neural force functional correctly describes all contributions to $\mathbf{f}_{\mathrm{int}}(\mathbf{r};[\rho,\mathbf{v}])$ even though the overall shape changes non-trivially with e.g. the amplitude of the density modulation, as illustrated in figure 6(a). The transport coefficients associated to the viscous (A_v) and the structural (A_s) components are shown in figure 6(b). They again scale differently in the limit of weak flows ($J_0 \to 0$) signaling a different dependence on powers of the velocity field.

3.4. Beyond the simulation box

The network is trained to reproduce the internal force field locally, at a given space point. Due to this local inference, there is no constraint on the specific choice of the system size. The neural force functional can be used straightforwardly to predict the internal force field in systems that outscale those used during training.

An illustrative example is shown in figure 7. We create a steady state in a system of size 100σ along the z-direction (i.e. ten times larger than the length of the training simulation box). The flow is directed along

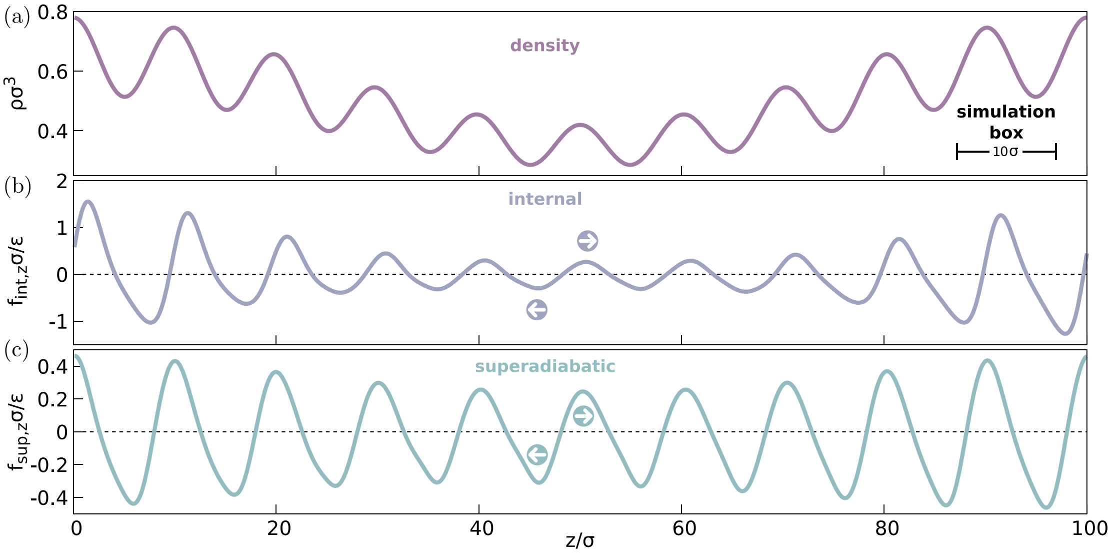

Figure 7. Beyond the simulation box. (a) Density profile ρ and z-components of (b) the internal force field $\mathbf{f}_{\text{int}}$ and (c) the superadiabatic force field $\mathbf{f}_{\text{sup}}$ as a function of z in a system of size 100σ along the z-direction. The neural network is hence used for predictions in a system which is one order of magnitude larger than those used for its training. The system is in steady state with a current of magnitude $J_0\sigma^2\tau = 10$.

the z-axis only and the magnitude of the current is $J_0\sigma^2\tau = 10$. The density profile, and hence the velocity profile, oscillate smoothly at length scales of the order of 10σ. Despite these rather smooth oscillations (as compared to e.g. those in a crystal state) the superadiabatic force, figure 7(c), is a significant contribution to the total internal force, figure 7(b).

This system illustrates the relevance of superadiabatic forces to understand the dynamics in systems much larger than the length scale of the particles. Our local learning approach opens the door to describe the dynamics of macroscopic systems with microscopic resolution.

3.5. Beyond steady states

Although the network has been trained only with steady states, we can use it to investigate full non-equilibrium situations in which the one-body fields depend explicitly on time t. This constitutes an approximation to the real dynamics since the memory effects in full non-equilibrium and in steady state are not identical [59]. Formally, the full time-dependent nonequilibrium mapping as given by PFT requires to include not only instantaneous profiles, but also their history up to the time of interest. Nevertheless, as we demonstrate here, the agreement with simulations is still good since the network provides a reasonable approximation for the superadiabatic force field in full non-equilibrium.

In the general case, a time and space-dependent external force field $\mathbf{f}_{\text{ext}}(\mathbf{r},t)$ drives the dynamics of the density $\rho(\mathbf{r},t)$ and the velocity $\mathbf{v}(\mathbf{r},t)$ profiles.

We discretize time in steps of Δt. Using the continuity equation (7), we evolve the density profile one time step

$$\rho(\mathbf{r},t+\Delta t) = \rho(\mathbf{r},t) - \Delta t\nabla\cdot\mathbf{J}(\mathbf{r},t)\,. \tag{29}$$

Via the exact force balance equation (4), we can calculate the velocity profile at time $t+\Delta t$

$$\gamma\mathbf{v}(\mathbf{r},t+\Delta t) = \mathbf{f}_{\text{int}}(\mathbf{r},t+\Delta t) + \mathbf{f}_{\text{ext}}(\mathbf{r},t+\Delta t) - k_BT\nabla\ln\rho(\mathbf{r},t+\Delta t)\,. \tag{30}$$

Since the internal force at time $t+\Delta t$ is unknown, the solution to this equation can be found using a Picard iteration: We start with a guess for $\mathbf{v}(\mathbf{r},t+\Delta t)$ and use it together with $\rho(\mathbf{r},t+\Delta t)$ from equation (29) and the network to obtain the corresponding internal force field $\mathbf{f}_{\text{int}}(\mathbf{r},t+\Delta t)$. Next, using the right hand side of equation (30) we construct a new $\mathbf{v}(\mathbf{r},t+\Delta t)$ and repeat the procedure until the left and the right hand side of equation (30) coincide up to a given tolerance. In our experience, only a few iterations are needed to achieve convergence.

In practice, if the time step is small enough (here we use $\Delta t/\tau = 5\cdot 10^{-5}$) the Picard iteration to evolve the velocity profile in time is not necessary. The velocity profile at the next time step is well approximated by assuming

$$\gamma\mathbf{v}(\mathbf{r},t+\Delta t) = \mathbf{f}_{\text{int}}(\mathbf{r},t) + \mathbf{f}_{\text{ext}}(\mathbf{r},t+\Delta t) - k_BT\nabla\ln\rho(\mathbf{r},t+\Delta t), \tag{31}$$

with

$$\mathbf{f}_{\text{int}}(\mathbf{r},t) = \mathbf{f}^{\star}_{\text{int}}(\mathbf{r};[\rho(\mathbf{r},t),\mathbf{v}(\mathbf{r},t)]), \tag{32}$$

that is, the network prediction for the internal force field of the previous time step.

Neural dynamical density functional

To draw a comparison we construct here a neural dynamical density functional (nDDFT) which neglects the superadiabatic contributions to the internal force field. The task is straightforward since we simply need to use $\mathbf{f}_{\text{int}}(\mathbf{r},t) = \mathbf{f}^{\star}_{\text{ad}}(\mathbf{r},t;[\rho])$ in equation (31). Recall that the adiabatic force field $\mathbf{f}^{\star}_{\text{ad}}(\mathbf{r},t;[\rho])$ is at each time t that of an equilibrium system ($\mathbf{v}=0$) with the same ρ as the out-of-equilibrium system, see equation (15).

We compare predictions of the neural force functional and nDDFT with computer simulations in figure 8 and in supplementary movie 1. The time-dependent one-body fields in the simulations have been obtained by averaging at the desired time t over an ensemble of $\sim 10^5$ simulations that differ in the initial microstate and in the realization of the noise (Brownian motion). The starting point is a bulk system with constant density $\rho_b\sigma^3 = 0.5$. At $t=0$ we switch on the external force field shown in figure 8(a) which is

$$\mathbf{f}_{\text{ext}}(z,t) = f_0\sin(k_0z)\,\hat{\mathbf{e}}_x + f_1\cos(\omega_1t - k_1z)\,\hat{\mathbf{e}}_z, \tag{33}$$

with parameters $f_0\sigma/\epsilon = 30$, $k_0L = 2\pi$, $f_1\sigma/\epsilon = 5$, $\omega_1\tau = 4\pi$, and $k_1L = 4\pi$. That is, the motion is driven by a static wave in x-direction, and a travelling wave in z-direction.

The system responds to the external driving with a density modulation, figure 8(b), that travels along the z-direction. Dynamical density functional theory is entirely unaware of the internal force field along the x-direction, figure 8(c), which is non-conservative and purely superadiabatic since the density is homogeneous in this direction. Note that the flow in x-direction is given according to the force balance equation (4) by the sum of the external and the internal force fields. Therefore, the absence of superadiabatic forces in nDDFT leads in general to an inaccurate description of such nonequilibrium flows [38], which we have exemplified here via the time-dependent situation depicted in figure 8. In contrast, the internal force field provided by the neural force functional and therefore the flow are in good agreement with simulations, see figure 8(c).

The internal force field along the z-direction, figure 8(d), has both adiabatic and superadiabatic components. The adiabatic component dominates and hence the prediction of nDDFT seems at first glance reasonable as compared to simulations. However, the superadiabatic component, which is correctly reproduced by the neural force functional, is far from being negligible and it is responsible for a clear physical effect. We show in figure 8(d) the amplitude of the density modulation $\Delta\rho$ as a function of time (see also supplementary movie 1). The density modulation grows after switching on the external force and afterwards it varies periodically with time. Dynamical density functional theory completely misses this effect which is due to structural superadiabatic forces generated by the flow in z-direction. Again the neural force functional reproduces the oscillations of the density modulation although it slightly overestimates the amplitude. The differences with simulation results arise, at least partially, due to memory effects being different in steady state and in full non-equilibrium, which is not captured by our neural network due to the steady state training data.

Our neural force functional can simultaneously generalize to systems that are larger than the simulation box and that are in full non-equilibrium. An illustrative example is shown in supplementary movie 2 where a system with size $L/\sigma = 30$ and subject to a complex time-dependent external force is analyzed. The predictions of the neural force functional for the density and the velocity profiles as well as for the internal force field are much closer to the simulation data than those of nDDFT.

4. Conclusions and future work

We have trained a neural network to represent the kinematic functional mapping $\{\rho,\mathbf{v}\}\to\mathbf{f}_{\text{int}}$ described in PFT for particles following overdamped Brownian dynamics. We create the one-body fields directly from particle-based computer simulations of the supercritical Lennard-Jones fluid. After machine learning the mapping, the network can be deployed in several applications including custom flow, the analysis of

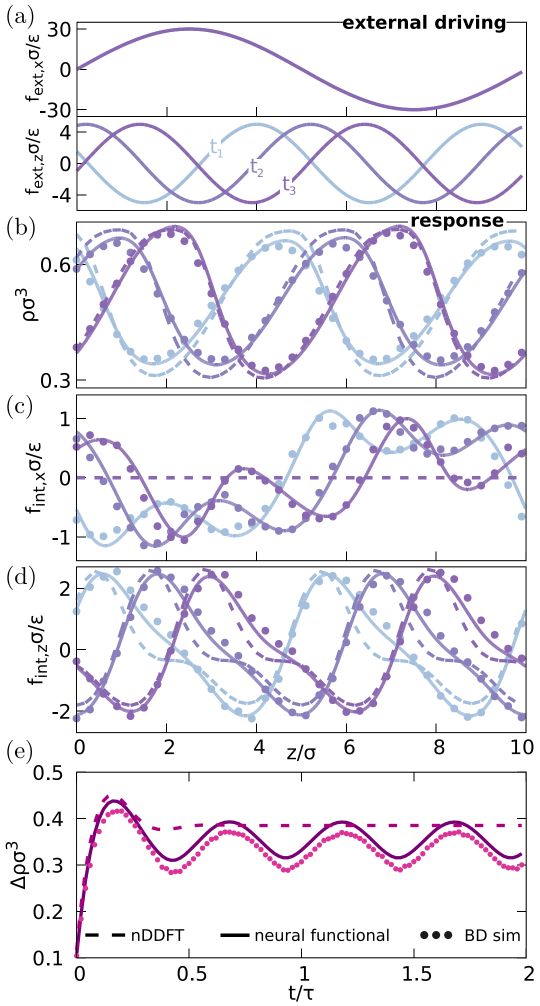

Figure 8. Beyond steady state. Time evolution of selected one-body fields in a full non-equilibrium system according to the neural force functional (solid lines), nDDFT (dashed lines), and BD simulations (symbols). The color indicates the time $t_1/\tau = 0.40$ (blue), $t_2/\tau = 0.64$ (light violet), and $t_3/\tau = 0.72$ (dark violet). Shown are as a function of the z coordinate: the components of the external force field $\mathbf{f}_{\text{ext}}$ in x- and z-direction (a), the density profile ρ (b), and the components of the internal force field in x- (c) and z-direction (d). Panel (e) shows the amplitude of the density modulation $\Delta\rho$ as a function of time t.

superadiabatic forces, and the quantification of transport coefficients. Analogously to neural functional theory in equilibrium [26], the application of the network is not restricted to the original size of the simulation box in which the training data has been created. The representation of the functional kinematic mapping is constructed in a spatially local manner and hence the prediction of the internal force field can be applied straightforwardly to systems of virtually any size. Despite the network being trained with steady-state data, in our tests it gives a reasonable approximation of the internal force field in full non-equilibrium situations. In our comparison with simulations, only relatively small deviations occur due to memory effects. Note, however, that the time scales of memory effects can be small [35] and hence large deviations might appear at sufficiently small time scales. The results agree well with simulations and clearly outperform dynamical density functional theory that can also be straightforwardly implemented as a limiting case of the neural force functional.

The input space of the dataset sets the main limitation of the approach. Since the kinematic mapping is formally exact, the discrepancies between the predictions of the neural force functional and the simulation data are due to the errors associated with the network. Inaccurate predictions can occur if we try to extrapolate far from the input space (for example predicting the internal force of a system with a current much larger than the maximal current included in the training set). The symmetry operations used to augment the data can also be used to asses the validity of the predictions. If the predicted internal force field does not satisfy the symmetry operations, we can be certain that the solution is not accurate. Conversely, if the predicted force field satisfies the symmetry operations, it strongly indicates that the prediction is accurate. For the case of neural dynamical density functional theory, the discrepancies between the predictions and the simulation data are mostly due to the neglect of the superadiabatic forces in nDDFT.

Several applications and extensions of this work merit further consideration. (i) We have trained the network at constant (supercritical) temperature T. A natural extension is to prepare a training set with different values of the temperature and using T as another input network parameter. Complications might arise e.g. in the two-phase regions of the phase diagram. (ii) The method can be directly applied to other interparticle potentials, including hard interactions [60] for which the internal force field can be sampled indirectly via the force balance equation. Of particular interest are the cases of active particles [36] and anisotropic particles. There, the one-body density distribution depends not only on the spatial coordinates but also on the particle orientations. Also, if there are torques acting on the particles, the force balance equation couples to a torque balance equation [61, 62] and hence, the internal torque field also needs to be learnt. (iii) Finite size effects (and those due to truncation of the potential) might be systematically analyzed with a training set containing data from several simulation box sizes (cutoff distances). In particular it might be possible to rationalize the effect of the lateral size of the simulation box [63–65] on the dynamics, which constitutes an open problem that so far has not received much attention. (iv) The network used here is simple in terms of architecture and number of trainable parameters. Therefore, it might be feasible to generalize the geometry further and to train networks with generic two- and three-dimensional flows. Reduced-variance sampling methods [62, 66–69] can improve the sampling efficiency during the generation of the training set. (v) We have used only soft external forces but the training set can be complemented with other profiles such as e.g. particles in the presence of hard walls. (vi) As in equilibrium [26], the neural force functional can be used to perform functional calculus. Using automatic differentiation [70], it is possible to compute the functional derivatives of the output with respect to the inputs of the neural network. This gives direct access to e.g. the second and higher order derivatives of the excess power functional with respect to the velocity. In the limit of weak flows, the second derivative is related to the transport coefficients such as shear and bulk viscosities. Higher order derivatives might provide valuable information about the mathematical structure of the functional. Also, using functional line integration [26, 71] might be a practical route to the determination of the value of the excess power functional for prescribed $\rho(\mathbf{r})$ and $\mathbf{v}(\mathbf{r})$. (vii) Arguably the most promising extension is the application of the method to full non-equilibrium systems for which both, the training set (which then carries an explicit time dependence) and the architecture of the network need to be modified. A neural force functional trained with full non-equilibrium data in combination with automatic differentiation might provide direct access to the memory kernels [72–75] of the many-body system. (viii) Finally, the method is not restricted to overdamped Brownian dynamics. The network can be trained with data from particles following e.g. Newtonian [16], Langevin, and quantum many-body dynamics [17] in full non-equilibrium. There, the internal force field and the kinetic stress tensor carry in general an explicit dependence on the acceleration field [51, 76]. Hence, it might be convenient to also include the acceleration field, $\mathbf{a}$, as an input field of the training set even though it follows from the time derivative of the velocity field ($\mathbf{a} = \dot{\mathbf{v}}$). Moreover, the transport term in the force balance equation is not diffusive and also needs to be learnt.

We have trained the network with simulation data but it might be possible to generate a training set directly from experiments. Arrays of optical tweezers [77], magnetic patterns [78] and micro-fabricated obstacles [79] are examples of experimental setups in which the external force field can be customized. The internal force field could then be measured either directly [80] or indirectly via the force balance equation and also via machine learning the non-equilibrium dynamics from time-lapse microscopy images [81].

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files). The dataset and the trained model are deposited in Zenodo [47].

Appendix. Architecture of the neural network

We use a convolutional neural network, see figure 9(a), implemented in Keras [82]. The size of the input layer is $(561, 4)$. That is, we use 4 channels (one-dimensional arrays) to represent the four one-body profiles (density and three components of the velocity) which are discretized within $[z_0 - \Delta, z_0 + \Delta]$ leading to 561 input values per profile. The input layer is then processed by convolutional layers followed by a fully connected layer.

The first convolutional layer uses 16 filters. Hence, it takes the input layer and outputs a feature map of size $(561, 16)$. Each value of the feature map is calculated by convolving the values of all the channels in the previous layer in a window of size 20 (kernel size).

We use three convolutional layers connected sequentially and double the number of filters in each layer (the kernel size is kept constant). An average pooling layer is connected after each convolutional layer. The

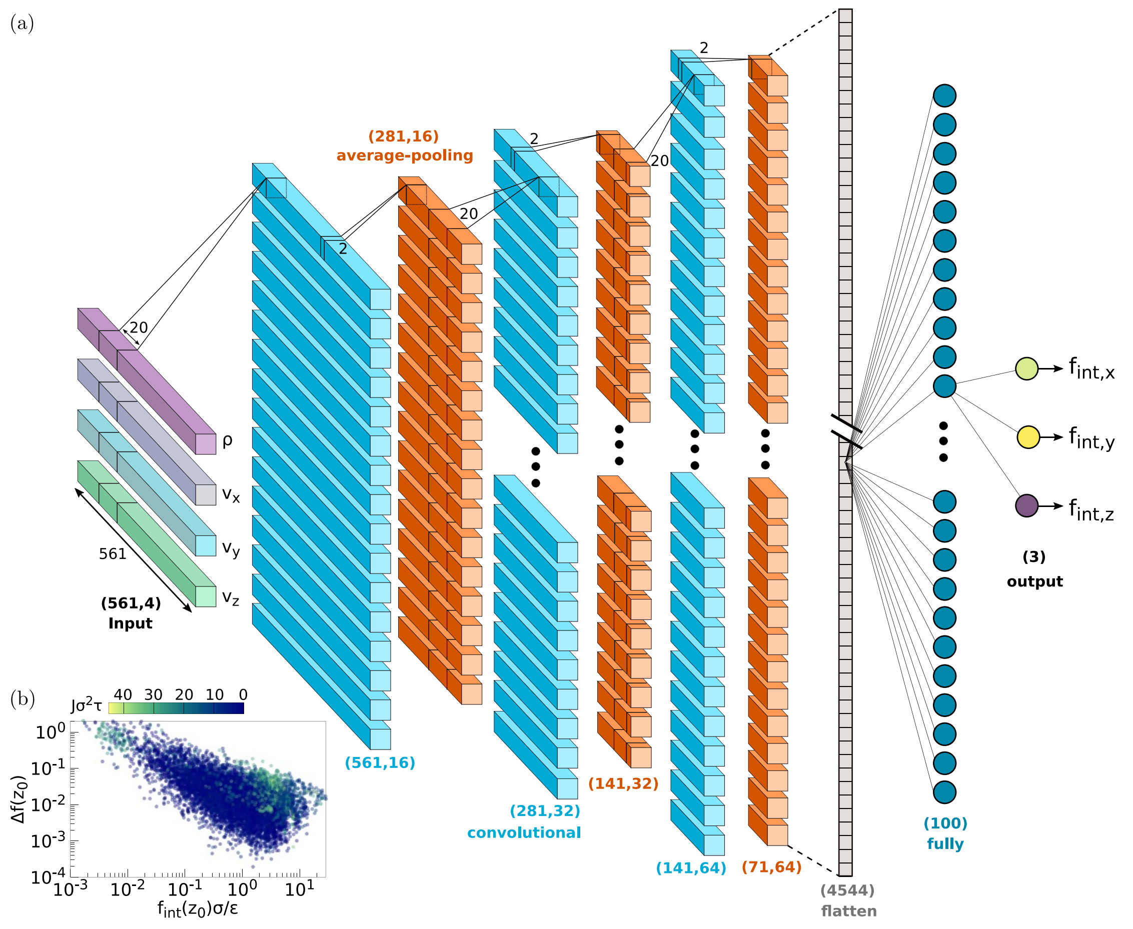

Figure 9. Neural network. (a) Schematic of the convolutional neural network. The input layer contains windows of the density and the three components of the velocity profile. Convolutional (average-pooling) layers are represented in blue (orange). The last average-pooling layer is flattened (grey) and connected to a fully connected layer (blue circles). The output layer consists of three nodes that output the components of $\mathbf{f}_{\text{int}}$ at position z_0. The numbers inside parentheses indicate the elements per channel and the number of channels of each layer. (b) Log-log plot of the relative error $\Delta f(z_0)$ of the neural network as a function of the magnitude of the internal force $f_{\text{int}}(z_0)$ for the samples in the test set. The data points are colored according to the magnitude of the current (color bar).

pooling layers calculate the average of two consecutive values of the feature map in the preceding convolutional layer, halving therefore the size of the feature map.

After the last pooling layer, the two-dimensional data is reshaped to a 1d-array via a flatten layer which is then connected to a fully connected layer containing 100 nodes. This layer connects to the output layer, made of three single nodes that output the three components of the internal force field at position z_0, i.e. $\mathbf{f}^{\star}_{\text{int}}(z_0)$.

Note that we do not normalize either the input fields (density and velocity) or the output field (internal force field) since we are interested in the raw value of the internal force field which depends on the raw values of the density and the velocity profiles. We have not encountered stability problems of the neural network in the range of currents analysed here. Since the internal force field tends to saturate for high values of the current, we do not expect stability problems if the range of currents is increased, but we cannot completely rule out their occurrence.

We use softplus activation functions for the nodes in the network (except the output nodes that use linear activation functions to output the internal force directly). In our experience, the softplus activation functions significantly improve the quality of the output.

The network contains about $5 \cdot 10^5$ parameters that are adjusted with the Adam optimizer to minimize the mean-square-error between network predictions and actual values of the internal force field. The initial learning rate is $2.5 \cdot 10^{-4}$ and it decreases by 1.5% after each epoch. The network is trained for 150 epochs with an initial batch size of 512 which is doubled every 50 epochs. The sequence of input-output pairs in each batch is selected randomly.

We show in figure 9(b) the relative error of the network prediction $\Delta f = |\mathbf{f}^{\star}_{\text{int}}(z_0) - \mathbf{f}_{\text{int}}(z_0)|/|\mathbf{f}_{\text{int}}(z_0)|$ as a function of the magnitude of the internal force field $|\mathbf{f}_{\text{int}}(z_0)|$ for the data in the test set. Recall that quantities without a star are simulation data. Data points are colored according to the magnitude of the current.

Samples with large relative errors typically correspond to cases where the internal force field is small. A large relative Δf is then expected since the statistical noise is comparable to the signal. This likely explains also the general trend of negative slope in figure 9(b). For fixed value of the magnitude of the internal force, the error increases with the magnitude of the current. For a comparable size of the training set, higher precision is achieved in equilibrium using a neural network to represent the functional map from ρ to c_1 [26]. This is not surprising due to the increased mathematical complexity of the non-equilibrium mapping. If higher precision is required, it should be possible to decrease the relative error by increasing the size of the training set.

ORCID iDs

Toni Zimmermann https://orcid.org/0009-0005-1181-1188
Florian Sammüller https://orcid.org/0000-0002-3605-329X
Sophie Hermann https://orcid.org/0000-0002-4012-9170
Matthias Schmidt https://orcid.org/0000-0002-5015-2972
Daniel de las Heras https://orcid.org/0000-0003-3219-0518

References

[1] Löwen H 2001 Colloidal soft matter under external control *J. Phys.: Condens. Matter* **13** R415–32

[2] Erbe A, Zientara M, Baraban L, Kreidler C and Leiderer P 2008 Various driving mechanisms for generating motion of colloidal particles *J. Phys.: Condens. Matter* **20** 404215

[3] Menzel A M 2015 Tuned, driven and active soft matter *Phys. Rep.* **554** 1–45

[4] Velev O D and Bhatt K H 2006 On-chip micromanipulation and assembly of colloidal particles by electric fields *Soft Matter* **2** 738

[5] Vissers T, Wysocki A, Rex M, Löwen H, Royall C P, Imhof A and van Blaaderen A 2011 Lane formation in driven mixtures of oppositely charged colloids *Soft Matter* **7** 2352

[6] Tierno P, Muruganathan R and Fischer T M 2007 Viscoelasticity of dynamically self-assembled paramagnetic colloidal clusters *Phys. Rev. Lett.* **98** 028301

[7] Lips D, Stoop R L, Maass P and Tierno P 2021 Emergent colloidal currents across ordered and disordered landscapes *Commun. Phys.* **4** 224

[8] Sullivan M, Zhao K, Harrison C, Austin R H, Megens M, Hollingsworth A, Russel W B, Cheng Z, Mason T and Chaikin P M 2002 Control of colloids with gravity, temperature gradients and electric fields *J. Phys.: Condens. Matter* **15** S11–S18

[9] Eckert T, Schmidt M and de las Heras D 2021 Gravity-induced phase phenomena in plate-rod colloidal mixtures *Commun. Phys.* **4** 202

[10] Isele M, Hofmann K, Erbe A, Leiderer P and Nielaba P 2023 Lane formation of colloidal particles driven in parallel by gravity *Phys. Rev.* E **108** 034607

[11] Faucheux L P, Bourdieu L S, Kaplan P D and Libchaber A J 1995 Optical thermal ratchet *Phys. Rev. Lett.* **74** 1504

[12] Reichhardt C and Reichhardt C J O 2016 Depinning and nonequilibrium dynamic phases of particle assemblies driven over random and ordered substrates: a review *Rep. Prog. Phys.* **80** 026501

[13] Figueroa-Morales N, Genkin M M, Sokolov A and Aranson I S 2022 Non-symmetric pinning of topological defects in living liquid crystals *Commun. Phys.* **5** 301

[14] Schilling T 2022 Coarse-grained modelling out of equilibrium *Phys. Rep.* **972** 1

[15] Schmidt M and Brader J M 2013 Power functional theory for Brownian dynamics *J. Chem. Phys.* **138** 214101

[16] Schmidt M 2018 Power functional theory for Newtonian many-body dynamics *J. Chem. Phys.* **148** 044502

[17] Schmidt M 2015 Quantum power functional theory for many-body dynamics *J. Chem. Phys.* **143** 174108

[18] Schmidt M 2022 Power functional theory for many-body dynamics *Rev. Mod. Phys.* **94** 015007

[19] Evans R 1979 The nature of the liquid-vapour interface and other topics in the statistical mechanics of non-uniform, classical fluids *Adv. Phys.* **28** 143

[20] The one-body direct correlation functional $c_1(\mathbf{r};[\rho])$ is related to the excess (over ideal gas) free energy functional $F_{\mathrm{exc}}[\rho]$ via $c_1(\mathbf{r};[\rho]) = -\delta\beta F_{\mathrm{exc}}[\rho]/\delta\rho(\mathbf{r})$ with $\beta = 1/k_BT$. The internal force field is then related to the one-body direct correlation function via $\mathbf{f}_{\mathrm{int}}(\mathbf{r};[\rho]) = k_BT\nabla c_1(\mathbf{r};[\rho])$. Here k_B is the Boltzmann constant and T is (absolute) temperature.

[21] Santos-Silva T, Teixeira P I C, Anquetil-Deck C and Cleaver D J 2014 Neural-network approach to modeling liquid crystals in complex confinement *Phys. Rev.* E **89** 053316

[22] Lin S C and Oettel M 2019 A classical density functional from machine learning and a convolutional neural network *SciPost Phys.* **6** 025

[23] Lin S C, Martius G and Oettel M 2020 Analytical classical density functionals from an equation learning network *J. Chem. Phys.* **152** 021102

[24] Cats P, Kuipers S, de Wind S, van Damme R, Coli G M, Dijkstra M and van Roij R 2021 Machine-learning free-energy functionals using density profiles from simulations *APL Mater.* **9** 031109

[25] Malpica-Morales A, Yatsyshin P, Durán-Olivencia M A and Kalliadasis S 2023 Physics-informed Bayesian inference of external potentials in classical density-functional theory *J. Chem. Phys.* **159** 104109

[26] Sammüller F, Hermann S, de las Heras D and Schmidt M 2023 Neural functional theory for inhomogeneous fluids: fundamentals and applications *Proc. Natl Acad. Sci.* **120** e2312484120

[27] Simon A, Weimar J, Martius G and Oettel M 2024 Machine learning of a density functional for anisotropic patchy particles *J. Chem. Theory Comput.* **20** 1062

[28] Dijkman J, Dijkstra M, van Roij R, Welling M, van de Meent J-W and Ensing B 2024 Learning neural free-energy functionals with pair-correlation matching (arXiv:2403.15007)

[29] Hansen-Goos H and Roth R 2006 Density functional theory for hard-sphere mixtures: the White Bear version mark II *J. Phys.: Condens. Matter* **18** 8413–8425

[30] de las Heras D and Schmidt M 2018 Velocity gradient power functional for Brownian dynamics *Phys. Rev. Lett.* **120** 028001

[31] Stuhlmüller N C X, Eckert T, de las Heras D and Schmidt M 2018 Structural nonequilibrium forces in driven colloidal systems *Phys. Rev. Lett.* **121** 098002

[32] Sammüller F, de las Heras D and Schmidt M 2023 Inhomogeneous steady shear dynamics of a three-body colloidal gel former *J. Chem. Phys.* **158** 054908

[33] de las Heras D and Schmidt M 2020 Flow and structure in nonequilibrium Brownian many-body systems *Phys. Rev. Lett.* **125** 018001

[34] Geigenfeind T, de las Heras D and Schmidt M 2020 Superadiabatic demixing in nonequilibrium colloids *Commun. Phys.* **3** 23

[35] Treffenstädt L L and Schmidt M 2021 Universality in driven and equilibrium hard sphere liquid dynamics *Phys. Rev. Lett.* **126** 058002

[36] Hermann S, Krinninger P, de las Heras D and Schmidt M 2019 Phase coexistence of active Brownian particles *Phys. Rev.* E **100** 052604

[37] Hermann S and Schmidt M 2023 Active crystallization from power functional theory *Phys. Rev. E* 109 L022601

[38] de las Heras D, Zimmermann T, Sammüller F, Hermann S and Schmidt M 2023 Perspective: How to overcome dynamical density functional theory *J. Phys.: Condens. Matter* **35** 271501

[39] de las Heras D, Renner J and Schmidt M 2019 Custom flow in overdamped Brownian dynamics *Phys. Rev.* E **99** 023306

[40] Marconi U M B and Tarazona P 1999 Dynamic density functional theory of fluids *J. Chem. Phys.* **110** 8032

[41] te Vrugt M, Löwen H and Wittkowski R 2020 Classical dynamical density functional theory: from fundamentals to applications *Adv. Phys.* **69** 121–247

[42] Frank M, Anderson D, Weeks E R and Morris J F 2003 Particle migration in pressure-driven flow of a Brownian suspension *J. Fluid Mech.* **493** 363–378

[43] Leighton D and Acrivos A 1987 The shear-induced migration of particles in concentrated suspensions *J. Fluid Mech.* **181** 415

[44] Dzubiella J, Hoffmann G P and Löwen H 2002 Lane formation in colloidal mixtures driven by an external field *Phys. Rev.* E **65** 021402

[45] Sammüller F and Schmidt M 2021 Adaptive Brownian dynamics *J. Chem. Phys.* **155** 134107

[46] Fortini A, de las Heras D, Brader J M and Schmidt M 2014 Superadiabatic forces in Brownian many-body dynamics *Phys. Rev. Lett.* **113** 167801

[47] Zimmermann T, Sammüller F, Hermann S, Schmidt M, and de las Heras D 2024 Neural force functional: dataset and model *Zenodo* (https://doi.org/10.5281/zenodo.12731656)

[48] Cohen T and Welling M 2016 Group equivariant convolutional networks *Proc. 33rd Int. Conf. on Machine Learning* vol 48 (PMLR) p 2990

[49] Karniadakis G E, Kevrekidis I G, Lu L, Perdikaris P, Wang S and Yang L 2021 Physics-informed machine learning *Nat. Rev. Phys.* **3** 422–440

[50] Renner J, Schmidt M and de las Heras D 2021 Custom flow in molecular dynamics *Phys. Rev. Res.* **3** 013281

[51] Renner J, Schmidt M and de las Heras D 2022 Shear and bulk acceleration viscosities in simple fluids *Phys. Rev. Lett.* **128** 094502

[52] Miskin M Z, Khaira G, de Pablo J J and Jaeger H M 2015 Turning statistical physics models into materials design engines *Proc. Natl Acad. Sci.* **113** 34

[53] Sherman Z M, Howard M P, Lindquist B A, Jadrich R B and Truskett T M 2020 Inverse methods for design of soft materials *J. Chem. Phys.* **152** 140902

[54] Coli G M, Boattini E, Filion L and Dijkstra M 2022 Inverse design of soft materials via a deep learning–based evolutionary strategy *Sci. Adv.* **8** eabj6731

[55] Obukhov A M 1983 Kolmogorov flow and laboratory simulation of it *Russ. Math. Surv.* **38** 13–26

[56] Jahreis N and Schmidt M 2024 in preparation

[57] Hermann S and Schmidt M 2021 Noether's theorem in statistical mechanics *Commun. Phys.* **4** 176

[58] Ling J, Kurzawski A and Templeton J 2016 Reynolds averaged turbulence modelling using deep neural networks with embedded invariance *J. Fluid Mech.* **807** 155–66

[59] Treffenstädt L L and Schmidt M 2020 Memory-induced motion reversal in Brownian liquids *Soft Matter* **16** 1518–26

[60] Mederos L, Velasco E and Martínez-Ratón Y 2014 Hard-body models of bulk liquid crystals *J. Phys.: Condens. Matter* **26** 463101

[61] Rex M, Wensink H H and Löwen H 2007 Dynamical density functional theory for anisotropic colloidal particles *Phys. Rev.* E **76** 021403

[62] Renner J, Schmidt M and de las Heras D 2023 Reduced-variance orientational distribution functions from torque sampling *J. Phys.: Condens. Matter* **35** 235901

[63] Chacón E, Tarazona P and Alejandre J 2006 The intrinsic structure of the water surface *J. Chem. Phys.* **125** 014709

[64] Duque D, Tarazona P and Chacón E 2008 Diffusion at the liquid-vapor interface *J. Chem. Phys.* **128** 134704

[65] Ogawa K, Oga H, Kusudo H, Yamaguchi Y, Omori T, Merabia S and Joly L 2019 Large effect of lateral box size in molecular dynamics simulations of liquid-solid friction *Phys. Rev.* E **100** 023101

[66] Borgis D, Assaraf R, Rotenberg B and Vuilleumier R 2013 Computation of pair distribution functions and three-dimensional densities with a reduced variance principle *Mol. Phys.* **111** 3486

[67] de las Heras D and Schmidt M 2018 Better than counting: density profiles from force sampling *Phys. Rev. Lett.* **120** 218001

[68] Schultz A J, Moustafa S G, Lin W, Weinstein S J and Kofke D A 2016 Reformulation of ensemble averages via coordinate mapping *J. Chem. Theory Comput.* **12** 1491

[69] Rotenberg B 2020 Use the force! Reduced variance estimators for densities, radial distribution functions and local mobilities in molecular simulations *J. Chem. Phys.* **153** 150902

[70] Margossian C C 2019 A review of automatic differentiation and its efficient implementation *Data Min. Knowl. Discov.* **9** e1305

[71] Brader J M and Schmidt M 2015 Free power dissipation from functional line integration *Mol. Phys.* **113** 2873–80

[72] Lesnicki D, Vuilleumier R, Carof A and Rotenberg B 2016 Molecular hydrodynamics from memory kernels *Phys. Rev. Lett.* **116** 147804

[73] Jung G, Hanke M and Schmid F 2017 Iterative reconstruction of memory kernels *J. Chem. Theory Comput.* **13** 2481

[74] Daldrop J O, Kowalik B G and Netz R R 2017 External potential modifies friction of molecular solutes in water *Phys. Rev.* X **7** 041065

[75] Meyer H, Pelagejcev P and Schilling T 2020 Non-Markovian out-of-equilibrium dynamics: a general numerical procedure to construct time-dependent memory kernels for coarse-grained observables *Europhys. Lett.* **128** 40001

[76] Brütting M, Trepl T, de las Heras D and Schmidt M 2019 Superadiabatic forces via the acceleration gradient in quantum many-body dynamics *Molecules* **24** 3660

[77] Schäffner D, Preuschoff T, Ristok S, Brozio L, Schlosser M, Giessen H and Birkl G 2020 Arrays of individually controllable optical tweezers based on 3D-printed microlens arrays *Opt. Express* **28** 8640

[78] Stuhlmüller N C X, Farrokhzad F, Kuświk P, Stobiecki F, Urbaniak M, Akhundzada S, Ehresmann A, Fischer T M and de las Heras D 2023 Simultaneous and independent topological control of identical microparticles in non-periodic energy landscapes *Nat. Commun.* **14** 7517

[79] Morin A, Desreumaux N, Caussin J-B and Bartolo D 2016 Distortion and destruction of colloidal flocks in disordered environments *Nat. Phys.* **13** 63–67

[80] Dong J, Turci F, Jack R L, Faers M A and Royall C P 2022 Direct imaging of contacts and forces in colloidal gels *J. Chem. Phys.* **156** 214907

[81] Gnesotto F S, Gradziuk G, Ronceray P and Broedersz C P 2020 Learning the non-equilibrium dynamics of Brownian movies *Nat. Commun.* **11** 5378

[82] Chollet F 2021 *Deep Learning With Python* 2nd edn (Manning Publications)