



CHORUS

This is the accepted manuscript made available via CHORUS. The article has been published as:

Learning unknown physics of non-Newtonian fluids

Brandon Reyes, Amanda A. Howard, Paris Perdikaris, and Alexandre M. Tartakovsky

Phys. Rev. Fluids **6**, 073301 — Published 9 July 2021

DOI: [10.1103/PhysRevFluids.6.073301](https://doi.org/10.1103/PhysRevFluids.6.073301)

Learning Unknown Physics of non-Newtonian Fluids

Brandon Reyes

Colorado School of Mines, Golden, CO

Amanda A. Howard

Pacific Northwest National Laboratory, Richland, WA

Paris Perdikaris

University of Pennsylvania, Philadelphia, PA

Alexandre M. Tartakovsky*

*Pacific Northwest National Laboratory, Richland,
WA; Department of Civil and Environmental Engineering,
University of Illinois Urbana-Champaign, Urbana, IL*

(Dated: June 9, 2021)

We present a formulation of the physics-informed neural network (PINN) method for learning effective viscosity of the generalized Newtonian fluid from measurements of velocity and pressure in time-dependent three dimensional flows and apply it to estimate viscosity models of two non-Newtonian systems (polymer melts and suspensions of particles) in shear flow between two parallel plates using only velocity measurements from numerical simulations. The PINN-inferred viscosity models agree with empirical models for shear rates with large absolute values but deviate for shear rates near zero where analytical models have an unphysical singularity. We show that once the unknown physics is learned the PINN method can be used to solve the momentum conservation equation governing flow of non-Newtonian fluids.

I. INTRODUCTION

In many applications data is scarce and indirect and the governing physics is not fully known, which limits the utility of standard machine learning (ML) and physics-based methods. On one hand, the conservation laws do not provide a closed system of equations. For example, the momentum and mass conservation equations governing fluid flow and solid body deformation require a stress-shear-rate relationship (we refer to such relationships as unknown physics) to close the system of these equations. On the other hand, experiments usually provide measurements of state variables and not stresses. For example, in non-Newtonian flow experiments one can easily measure velocity, but not stress or viscosity. Other examples include non-linear heat and mass transport, where one usually measures temperature and concentrations and not heat and mass fluxes or temperature-dependent thermal conductivity and concentration-dependent diffusion coefficient. This makes it impossible to use data-driven ML methods to learn unknown physics (stresses and fluxes) as functions or functionals of state variables. It is important to note that standard parameter estimation methods cannot be used for learning unknown physics because the function space is infinite-dimensional. It is this issue that the physics-informed neural network (PINN) method attempts to solve. PINN uses the known underlying structure of physical laws governed by PDEs or

ODEs to predict unknown functions or functionals from indirect observations. In the past, the PINN method was used to solve Navier-Stokes and Darcy equations [1, 2] and learn parameter fields and constitutive relationships in the models of flow in porous media [3, 4]. In this work, we extend the PINN method for estimating the non-Newtonian viscosity based solely on velocity data, where the (unknown) effective viscosity is a symmetric function of the shear rate. This makes the PINN method applicable for studying a wide range of complex physical processes with partially known physics.

II. PINN METHOD FOR NON-NEWTONIAN FLOW MODELS

Consider flow of a non-Newtonian incompressible fluid satisfying the momentum conservation equation:

$$\rho \frac{\partial \mathbf{v}}{\partial t} + \rho \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla P + \nabla \cdot \boldsymbol{\tau} + \mathbf{F}_b, \quad (\mathbf{x}, t) \in \Omega \times (0, T] \quad (1)$$

and the continuity equation

$$\nabla \cdot \mathbf{v} = 0, \quad (\mathbf{x}, t) \in \Omega \times (0, T] \quad (2)$$

subject to the appropriate boundary conditions. Here, $\mathbf{x} = [x, y, z]^T$ is the coordinate vector, t is time, $\mathbf{v}(\mathbf{x}, t) = [u(\mathbf{x}, t), v(\mathbf{x}, t), w(\mathbf{x}, t)]^T$ is the fluid velocity vector (u , v , and w are the x , y , and z components of \mathbf{v}), P is pressure, ρ is a known constant density, \mathbf{F}_b is the prescribed body force per unit volume, and $\boldsymbol{\tau}$ is a shear stress tensor. In the derivations below and numerical examples,

* amt1998@illinois.edu

we assume a generalized Newtonian approximation with the stress tensor of the form

$$\boldsymbol{\tau} = \mu(\dot{\boldsymbol{\epsilon}})\dot{\boldsymbol{\epsilon}}, \quad (3)$$

where $\dot{\boldsymbol{\epsilon}} = (\nabla \mathbf{v} + \nabla \mathbf{v}^T)$ is the strain rate, $\dot{\epsilon} = \sqrt{\frac{1}{2}\dot{\boldsymbol{\epsilon}} : \dot{\boldsymbol{\epsilon}}}$ is the effective strain rate, and $\mu(\dot{\epsilon})$ is the unknown strain-rate-dependent viscosity. Later in this section, we discuss how to extend the proposed PINN method to non-Newtonian fluids that are not purely viscous, i.e., cannot be accurately approximated with the generalized Newtonian stress model (3).

We consider two cases: (1) no measurements of μ are available and (2) N_μ measurements of μ are given for $\dot{\epsilon}_i$ values of $\dot{\epsilon}$, $\{\mu^*(\dot{\epsilon}_i)\}_{i=1}^{N_\mu}$. In both cases we assume that there are N_v measurements of the velocity \mathbf{v} , $\mathbf{v}^*(\mathbf{x}_i^v, t_i^v)$ at \mathbf{x}_i^v - t_i^v space-time coordinates and N_P measurements of P , $P^*(\mathbf{x}_i^P, t_i^P)$ at \mathbf{x}_i^P - t_i^P space-time coordinates. We approximate the viscosity $\mu(\dot{\epsilon})$ and the velocity $\mathbf{v}(\mathbf{x}, t)$ and pressure $P(\mathbf{x}, t)$ with fully connected feed-forward deep neural networks (DNNs). The velocity vector field is approximated as $\mathbf{v}(\mathbf{x}, t) \approx \hat{\mathbf{v}}(\mathbf{x}, t; \theta)$, where in general three-dimensional case, the $\hat{\mathbf{v}}$ DNN has four dimensional input layer with components (x, y, z, t) , three-dimensional output layer with components $(\hat{u}, \hat{v}, \hat{z})$, and the parameters (weights and biases) are denoted by θ . Details of DNNs architectures can be found elsewhere (e.g., [3]). Here, we just mention that each output of $\hat{\mathbf{v}}$ is a known function of \mathbf{x} and t and parameters θ . Using the differentiation chain rule or the automatic differentiation (AD) [5] packages available in most ML libraries, one can exactly compute the time and spatial derivatives of $\hat{u}(\mathbf{x}, t; \theta)$, $\hat{v}(\mathbf{x}, t; \theta)$ and $\hat{z}(\mathbf{x}, t; \theta)$ and functions of these derivatives, including the strain rate $\hat{\boldsymbol{\epsilon}}(\mathbf{x}, t; \theta) = (\nabla \hat{\mathbf{v}}(\mathbf{x}, t; \theta) + \nabla \hat{\mathbf{v}}(\mathbf{x}, t; \theta)^T)$ and effective strain rate $\hat{\epsilon}(\mathbf{x}, t; \theta) = \sqrt{\frac{1}{2}\hat{\boldsymbol{\epsilon}}(\mathbf{x}, t; \theta) : \hat{\boldsymbol{\epsilon}}(\mathbf{x}, t; \theta)}$. With this approximation of the shear rate at hand, we can approximate effective viscosity with a DNN that is function of $\hat{\epsilon}$ as $\mu(\hat{\epsilon}(\mathbf{x}, t; \theta)) \approx \hat{\mu}(\hat{\epsilon}(\mathbf{x}, t; \theta); \gamma)$, where γ is the assembly of weights and biases of the $\hat{\mu}$ DNN. The $\hat{\mu}$ DNN has five-dimensional input layer $(x, y, z, t, \hat{\epsilon})$ and one-dimensional output layer. The pressure field is approximated as $P(\mathbf{x}, t) = \hat{P}(\mathbf{x}, t, \beta)$, where β is an assembly of the parameters of the \hat{P} DNN, which has four-dimensional input layer and one-dimensional output layer. We train \hat{u} , $\hat{\mu}$, and \hat{P} (i.e., evaluate θ , γ , and β) jointly using Eqs. (1) and (2) and the appropriate boundary conditions as constraints. This allows us to train $\hat{\mu}$ even without direct measurements of μ .

In the PINN method, the PDE constraints are imposed in a soft form, i.e., by minimizing the residuals of Eqs. (1) and (2). We remind that the DNN derivatives with respect to x , t , and the parameters θ and γ can be computed analytically. The derivatives with respect to x and t are needed to compute the PDE residuals, while the derivatives with respect to parameters are required to estimate the parameters' values in the process known as backpropagation [6]. Here, we use AD to compute deriva-

tives.

Substituting \hat{u} and $\hat{\mu}$ in the governing equations forms the ‘‘auxiliary’’ DNNs:

$$\hat{\mathbf{f}}(\mathbf{x}, t; \theta, \gamma, \beta) = \rho \frac{\partial \hat{\mathbf{v}}}{\partial t} + \rho \hat{\mathbf{v}} \cdot \nabla \hat{\mathbf{v}} + \nabla \hat{P} - \nabla(\hat{\mu}(\hat{\epsilon})\hat{\epsilon}) \quad (4)$$

and

$$\hat{g}(\mathbf{x}, t; \theta) = \nabla \cdot \hat{\mathbf{v}}. \quad (5)$$

We train the DNNs simultaneously by minimizing the loss function

$$\begin{aligned} L(\theta, \gamma, \beta) = & \frac{\omega_v}{N_v} \sum_{i=1}^{N_v} [\hat{\mathbf{v}}(\mathbf{x}_i^v, t_i^v; \theta) - \mathbf{v}^*(\mathbf{x}_i^v, t_i^v)]^2 \quad (6) \\ & + \frac{\omega_P}{N_P} \sum_{i=1}^{N_P} [\hat{P}(\mathbf{x}_i^P, t_i^P; \beta) - P^*(\mathbf{x}_i^P, t_i^P)]^2 \\ & + \frac{\omega_\mu}{N_\mu} \sum_{i=1}^{N_\mu} [\hat{\mu}(\hat{\epsilon}_i; \gamma) - \mu^*(\dot{\epsilon}_i)]^2 \\ & + \frac{\omega_f}{N_f} \sum_{i=1}^{N_f} [\hat{\mathbf{f}}(\mathbf{x}_i^f, t_i^f; \theta, \gamma, \beta) - \mathbf{F}_b]^2 \\ & + \frac{\omega_g}{N_g} \sum_{i=1}^{N_g} [\hat{g}(\mathbf{x}_i^g, t_i^g; \theta)]^2 \end{aligned}$$

as

$$(\theta, \gamma, \beta) = \arg \min_{\theta^*, \gamma^*, \beta^*} L(\theta^*, \gamma^*, \beta^*) \quad (7)$$

In $L(\theta, \gamma, \beta)$, the first three terms force $\hat{\mathbf{v}}(\mathbf{x}_i^v, t_i^v; \theta)$, $\hat{P}(\mathbf{x}_i^P, t_i^P; \beta)$, and $\hat{\mu}(\hat{\epsilon}_i; \gamma)$ to match the measurements of these variables and the last two terms force the DNNs to satisfy the governing equations. Dirichlet boundary conditions for \mathbf{v} and P are imposed by treating them as the measurements of \mathbf{v} and P on the corresponding boundaries and including them in the first two terms in $L(\theta, \gamma, \beta)$. Neumann boundary conditions are imposed by penalizing the normal derivatives at the corresponding boundaries in the loss function [3]. The weights $\{\omega_i\}_{i=v, P, \mu, f, g}$ reflect the fidelity level of the data and physics models. For example, in general, u and P measurements are more accurate than viscosity measurements, so $\omega_v \geq \omega_\mu$ and $\omega_P \geq \omega_\mu$. We note that Eq. (3) is an approximation relying on the generalized Newtonian model of the viscous stress, therefore, $\omega_f \leq \omega_u$ and ω_P . The relative values of ω_i can also affect the convergence rate of iterative solutions of the minimization problem (7) [7].

For non-Newtonian fluids that are not purely viscous, Eq. (3) can be generalized as [8, 9]

$$\boldsymbol{\tau} = \boldsymbol{\tau}^E - \frac{1}{3} \text{tr}(\boldsymbol{\tau}^E) \mathbf{I}, \quad (8)$$

where \mathbf{I} is the unit tensor and $\boldsymbol{\tau}^E$ is the so-called extra stress defined as

$$\boldsymbol{\tau}^E = A\mathbf{I} + B\dot{\boldsymbol{\epsilon}} + C\dot{\boldsymbol{\epsilon}}^2. \quad (9)$$

Here, A , B , and C are functions of the three scalar invariants of $\dot{\epsilon}$: $= \text{tr}(\dot{\epsilon})$, $II_e = \frac{1}{2}[\text{tr}(\dot{\epsilon}^2) - I_e^2]$, and $III_e = \det \dot{\epsilon}$. The PINN method can be extended to the stress model (8), by approximating A , B , and C with DNNs that are functions of I_e , II_e , and III_e . For viscoelastic fluids, $\boldsymbol{\tau}^E$ is given by integral equations or differential equations [10]. The extension of the PINN method for such fluids is less straight forward and is outside of the scope of this paper. A starting point in such analysis could be a recent work on learning viscoelastic constitutive relationships with physics-constraint DNNs in finite-element models of material deformation [11].

In the following, we apply the PINN method to learn μ in a shear flow between two parallel plates satisfying the steady-state momentum conservation equation:

$$\frac{d}{dy} \left[\mu(u_y(y)) \frac{du(y)}{dy} \right] = -C \quad \text{for } y \in \Omega = (0, H), \quad (10)$$

where the velocity vector is given by $\mathbf{u} = (u(y), 0, 0)^T$, $u_y \equiv du/dy$ is the shear rate, H is the channel width, and C is a force per unit volume. Eq (10) assumes that the viscous stress has the form $\mu(u_y)u_y(y)$, where $\mu(u_y)$ is the unknown shear-rate-dependent viscosity and that flow is laminar. The fluid velocity u is subject to the no-slip boundary conditions (BCs):

$$u(0) = 0, \quad u(H) = 0. \quad (11)$$

In this example, the loss function takes the form

$$\begin{aligned} L(\theta, \gamma) = & \frac{\omega_1}{N_u} \sum_{i=1}^{N_u} [\hat{u}(y_i; \theta) - u^*(y_i)]^2 \\ & + \frac{\omega_2}{2} [\hat{u}(y=0; \theta)^2 + \hat{u}(y=H; \theta)^2] \\ & + \frac{\omega_3}{N_f} \sum_{i=1}^{N_f} [\hat{f}(y_i; \theta, \gamma) + C]^2 \\ & + \frac{\omega_4}{N_\mu} \sum_{i=1}^{N_\mu} [\hat{\mu}(u_{y_i}; \gamma) - \mu^*(u_{y_i})]^2, \end{aligned} \quad (12)$$

where

$$\hat{f}(y; \theta, \gamma) = \frac{d}{dy} \left[\hat{\mu}(\hat{u}_y(y; \theta); \gamma) \frac{d\hat{u}(y; \theta)}{dy} \right]. \quad (13)$$

To solve this minimization problem we set the initial values of θ and γ using the Xavier's normal initialization scheme [12]. Next, we run the Adam optimizer [13] for a set number of steps. Finally, we run the quasi-Newton L-BFGS-B optimizer [14] until the desired convergence and tolerance are achieved. We find that for the considered here problems, this combination of the optimizers increases the convergence rate and reduces the computational cost as compared to using either optimizer alone. We use DNNs with two hidden layers with sixty nodes each and a learning rate of 0.001 for the Adam optimizer unless otherwise noted. The error $\|\hat{e}_{\hat{u}}\|_2 =$

$\|\hat{u}(y; \theta) - u^*(y)\|_2 / \|u^*(y)\|_2$ estimates the accuracy of the DNN approximations of u relative to the u measurements and the error $\|f\|_\infty = \max_{1 \leq i \leq N_u} |f(y_i; \theta, \gamma) + C|$ is a measure of how well the DNN approximations of u and μ satisfy Eq. (10).

We refer to the PINN method that is used to evaluate the unknown viscosity function given the measurements u (or u and μ) as the inverse PINN. Once $\hat{\mu}$ is trained, the PINN method can also be used to solve the momentum conservation equation without observations of u (and/or μ) if the shear rate does not exceed the maximum shear rate in the experiment used to train $\hat{\mu}$. To train \hat{u} as an approximate solution of Eq. (10) we use the loss function Eq. (12) with $\omega_1 = \omega_4 = 0$ and $\omega_2 = \omega_3 = 1$. We refer to this application of PINNs as the forward PINN method. More details about DNN approximation and training in the PINN framework can be found in [1, 3].

III. VALIDATION OF THE INVERSE AND FORWARD PINN METHODS

We first validate the ability of the inverse PINN method to learn the unknown shear-dependent viscosity using velocity data generated with the Ostwald-de Waele power-law effective viscosity model [15], $\mu_{pl}(u_y(y)) = K |u_y(y)|^{n-1}$, where K is the power-law consistency coefficient and n is the power-law index. This model in combination with Eqs. (10) and (11) allows for an analytical solution for $u(y)$ and $du(y)/dy$ [16].

We generate two data sets by selecting $N_u = 501$ uniformly distributed measurements of u from the analytical solution for u using both $n = 0.898$ (shear-thinning fluid) and $n = 1.2$ (shear-thickening fluid) with $C = 0.75$, $H = 25$, and $K = 40.788$. For both values of n we train the $\hat{u}(y; \theta)$ and $\hat{\mu}(\hat{u}_y(y; \theta); \gamma)$ DNNs by minimizing the loss function Eq. (12) with $\omega_1 = \omega_2 = \omega_3 = 1$ and $\omega_4 = 0$. We note that the minimization problem is not convex and its solution (θ, γ) can depend on the initial values of θ and γ . To demonstrate how different initial values for the weights affect the PINN solution, we solve the minimization problem with 100 different initializations of θ and γ and then average the resulting DNNs $\hat{u}(y; \theta)$ and $\hat{\mu}(\hat{u}_y(y; \theta); \gamma)$ to obtain the solutions for $u(y)$ and $\mu(u_y)$, respectively. For $n = 0.898$ the average solutions are compared with the analytical solutions in Figs. 1a and 1b. The average DNN $\hat{u}(y)$ solution agrees well with the analytical $u(y)$ solution with the maximum error of 0.1% at $y = H/2$. The average DNN $\hat{\mu}(\hat{u}_y)$ solution agrees with the analytical model for large shear rates ($|u_y| > 0.02$) within 1.44%. For small shear rates, the DNN solution deviates from the analytical solution and for zero shear rate has a finite value while the analytical solution has a nonphysical singularity. Fig. 1b also shows that the maximum standard deviation in the learned $\mu(u_y)$ is at $u_y = 0$ and is two orders of magnitude smaller than the mean value of μ at $u_y = 0$, indicating that the uncertainty of the PINN method due to

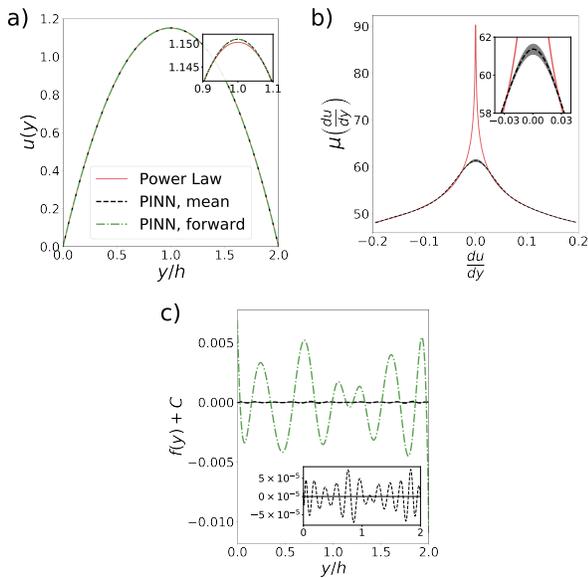


FIG. 1. Inverse and forward PINN solutions for the synthetic data generated from the analytical solution for a power-law fluid with $n = 0.898$. (a) The average velocity profile from 100 runs for the PINN results, and one run for the PINN forward model. (b) The average $\frac{du}{dy}$ of the 100 runs versus the average μ . The width of the gray area corresponds to four standard deviation of μ . (c) Average error in satisfying the ODE.

DNN initialization is relatively small. Fig. 1c depicts the residual $\hat{f}(y; \theta, \gamma) + C$ of Eq. (10) as a function of y . The small values of the residual ($\|f\|_\infty = 7.45 \times 10^{-5}$ and 5.061×10^{-4} for $n = 0.898$ and 1.2, respectively) and $\|e_{\hat{u}}\|_2$ ($\|e_{\hat{u}}\|_2 = 2.345 \times 10^{-4}$ and 2.135×10^{-4} for $n = 0.898$ and 1.2, respectively) show that the DNNs \hat{u} and $\hat{\mu}$ both approximate the data and satisfy Eq. (10) for shear-thinning and shear-thickening fluids.

Next, we validate the ability of the forward PINN method to solve Eq. (10). We fix the weights of the DNN $\hat{\mu}$ obtained from the inverse PINN with $n = 0.898$ and train the $\hat{u}(y, \theta)$ DNN by minimizing the loss function Eq. (12) with $\omega_1 = \omega_4 = 0$, $\omega_2 = \omega_3 = 1$. Fig. 1a shows that the trained $\hat{u}(y, \theta)$ closely agrees with the analytical solution for the power-law fluid with $n = 0.898$ with the maximum error of 0.36%. Fig. 1c shows the maximum residual corresponding to this DNN is two orders of magnitude smaller than C , indicating that \hat{u} approximately satisfies Eq. (10).

IV. MONODISPERSE POLYMER MELTS

We consider a synthetic Dissipative Particle Dynamic (DPD) fluid consisting of chains of N equal-size beads connected by springs to model polymer melts. Two-dimensional DPD simulations of such fluids between two parallel plates with chains made of $N = 2, 5$, and 25 beads are presented in [17]. In [17], the DPD results were used to compute $\mu_{DPD}(du(y)/dy)$ using the Irving-

Kirkwood relationship [18].

We use the velocity data from [17] and the inverse PINN method with $\omega_1 = \omega_2 = \omega_3 = 1$ and $\omega_4 = 0$ in Eq. (12) to estimate $\mu(u_y)$. To match [17], $C = 0.75$ and $H = 25$. The relative velocity error and the maximum residual error are $\|e_{\hat{u}}\|_2 = 2.6 \times 10^{-4}$, 1.9×10^{-4} , and 4.4×10^{-4} and $\|f\|_\infty = 1.5 \times 10^{-4}$, 1.8×10^{-4} , and 1.2×10^{-4} for $N=2, 5$, and 25, respectively. For all considered N , the relative error in u is less than 0.1% and the maximum residual error is three orders of magnitude smaller than the driving force C , indicating that the DNN \hat{u} accurately approximates data and the DNNs \hat{u} and $\hat{\mu}$ satisfy the governing equations. Figs. 2a and 2b compare the velocity profiles and viscosities estimated from the DPD simulation, μ_{DPD} and from the PINN method for $N = 2$. The DNN velocity profile $\hat{u}(y; \theta)$ closely matches the DPD velocity profile $u_{DPD}(y)$ with the maximum error of 5.52%. The agreement between $\hat{\mu}(u_y; \theta, \gamma)$ and $\mu_{DPD}(u_y)$ is less accurate with the maximum error of 3%. To test whether $u_{DPD}(y)$ and $\mu_{DPD}(u_y)$ satisfy Eq. (10), we train the $\hat{u}(y; \theta)$ and $\hat{\mu}(u_y; \theta, \gamma)$ DNNs conditioned on both u_{DPD} and μ_{DPD} measurements. Figs. 2a and 2b show that conditioning of the DNNs on the DPD measurements of u_{DPD} and the estimates of μ_{DPD} produces DNNs that match well both u_{DPD} and μ_{DPD} data (within 3.30% error for \hat{u} .) However, conditioning on the μ_{DPD} estimates also results in the residual errors that are two orders of magnitude larger than the residual errors in the case where no μ_{DPD} estimates are used to train the DNNs, as shown in Fig. 2c.

Next, we use the PINN method to evaluate the viscosity of the polymer melt with 25-bead chains. As for the melt with $N = 2$, we first train the \hat{u} and $\hat{\mu}$ DNNs using only $u_{DPD}(y)$ measurements. Fig. 3 shows that the \hat{u} DNN agrees well with the $u_{DPD}(y)$ measurements (the maximum error is less than 5.43%) and the resulting residual point errors are nearly zero (more than four orders of magnitude smaller than C). We also see that $\hat{\mu}$ significantly deviates from the $\mu_{DPD}(u_y)$ values estimated from the DPD simulations near a shear rate of zero (more than 80%). Then, we train the \hat{u} and $\hat{\mu}$ DNNs using both $u_{DPD}(y)$ and $\mu_{DPD}(y)$ data. Fig. 3 demonstrates that the resulting DNNs fit the $u_{DPD}(y)$ and $\mu_{DPD}(u_y)$ data well (within 7.18% for \hat{u}), but the corresponding residual is very large (on the order of C), demonstrating that these data cannot accurately be described by Eq. (10). We obtain similar results for the polymer melt with $N = 5$.

Finally, we demonstrate that once $\hat{\mu}(u_y; \theta, \gamma)$ is trained the forward PINN method can be used to solve Eq. (10) subject to the BC (11). We use the weights γ in the $\hat{\mu}(u_y; \theta, \gamma)$ DNN obtained above from the inverse PINN and train the forward solution, $\hat{u}_f(y; \theta)$, DNN by minimizing the loss function Eq. (12) with $\omega_1 = \omega_4 = 0$ and $\omega_2 = \omega_3 = 1$ for $C = 0.75$. For $N = 2$, Fig. 2a shows that the $\hat{u}_f(y; \theta)$ DNN matches the experimental data well with a maximum error of 5.53%. In addition, Fig. 2c demonstrates that the residual of the govern-

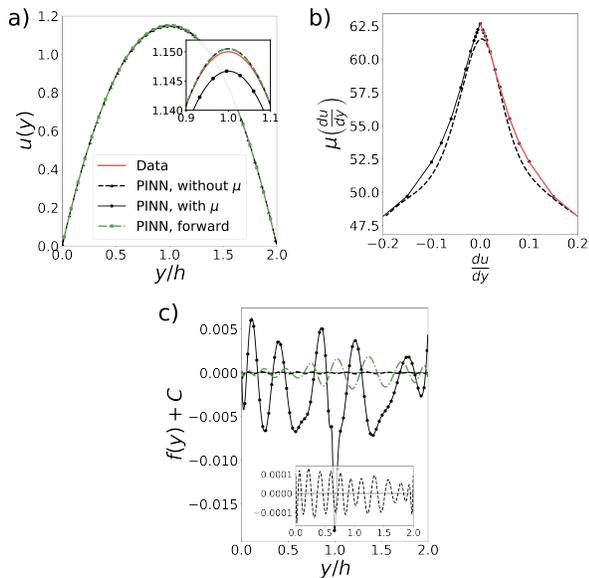


FIG. 2. Inverse and forward PINN model results for polymer chains of length $N = 2$. (a) Resulting velocity profiles. (b) Resulting viscosity profile. (c) Error in satisfying the ODE. Simulation data is from [17].

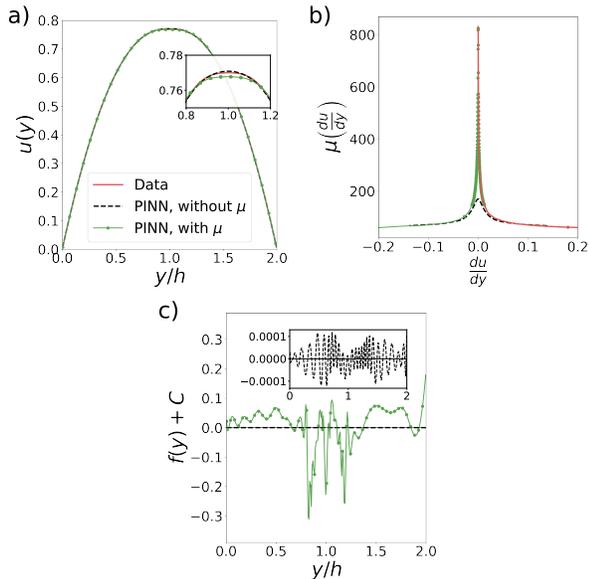


FIG. 3. Inverse PINN results for polymer chains of length $N = 25$. (a) Resulting velocity profiles. (b) Resulting viscosity profile. (c) Error in satisfying the ODE. Simulation data is from [17].

ing equation is two orders of magnitude smaller than C confirming that $\hat{u}_f(y; \theta)$ approximately solves Eq. (10) subject to Eq. (11).

V. DENSE SUSPENSIONS OF SPHERICAL PARTICLES

In this section, we employ the inverse PINN method to learn the shear-rate-dependent viscosity of densely packed spherical particles suspended in a Newtonian fluid using the velocity measurements presented in [19]. The considered data are obtained from the numerical simulations of suspension flows in a channel using the force coupling method (FCM) [19–21]. In the considered suspensions, the average particle volume fraction $\phi_a = \frac{4}{3}\pi a^3 N/V$ ranges from 0.2 to 0.4, where a is the particle radius, N is the number of particles, and V is the volume of the domain. In the FCM simulations, the particle radius was set to $a = 1$, the channel length to $L_x = 80$, the height to $H = 40$, and the width to $L_z = 30$. The channel walls were located at $y = 0$ and 40 , constant Dirichlet BCs for pressure were prescribed at the $x = 0$ and 80 boundaries with the pressure drop over the length of the channel $\Delta P/L_x = 0.029$, and periodic conditions were used in the z direction. At the continuum level, the considered suspension behaves as a non-Newtonian fluid and can be described by Eq. (10) with $C = \Delta P/L_x$.

The velocity profiles for the suspension flows with $\phi_a = 0.2$ and 0.4 are shown in Figs. 4a and d, respectively, and the local volume fractions $\phi(y)$ are depicted in Fig. 6a. A key feature of suspensions is irreversible shear-induced migration of particles to areas of low shear rate [22]. Particles in a suspension subjected to a Poiseuille flow will migrate to the channel centerline, increasing the volume fraction at the centerline. For high average volume fractions ϕ_a , the volume fraction at the centerline reaches the maximum close-packing ϕ_c , as shown in Fig. 6a. This migration also impacts the viscosity, resulting in a flattened parabola shape of the velocity profile that is observed in Figs. 4a and d.

As in the analysis of polymer melts above, we use the inverse PINN to find the viscosity $\mu(u_y(y))$ by approximating u and μ with $\hat{u}(y; \theta)$ and $\hat{\mu}(\hat{u}_y(y; \theta); \gamma)$ DNNs trained by minimizing the loss function (12) with $\omega_1 = \omega_2 = \omega_3 = 1$ and $\omega_4 = 0$. We use $N_u = 401$ measurements of the velocity profile $u(y)$ from the FCM simulations. Because the velocity profiles from the simulations (see Fig. 4a and d) deviate from the flattened-parabola shape near the walls due to particle layering, a phenomenon that cannot be described by Eq. (10), we train the PINN with velocity data in the range $y \in [0.25y/h, 1.75y/h]$, but still impose the zero Dirichlet BCs for u at $y = 0, H$.

Figs. 4a and 4d compare the velocity profiles of the suspension flow observed in the numerical simulations and approximated with the $\hat{u}(y; \theta)$ DNN for $\phi_a = 0.2$ and 0.4 , respectively. The $\hat{\mu}(\hat{u}_y(y; \theta); \gamma)$ DNN and the viscosity estimated from the numerical experiments are plotted in Figs. 4b and 4e. The viscosity $\mu(u_y)$ for the FCM simulations is found by computing $u_y(y)$ and $\phi(y)$ from the simulation data, assuming that $\mu(u_y) = \eta_s(\phi(u_y))\eta_f$ and

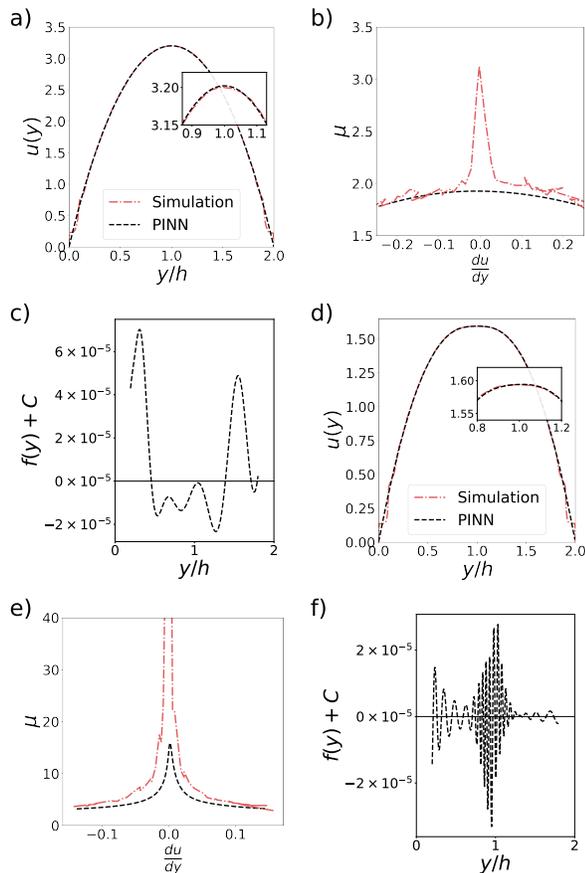


FIG. 4. Inverse PINN results for suspensions with average volume fraction $\phi_a = 0.2$ (a-c) and 0.4 (d-f). Panels (c) and (f) show the residuals of Eq (10) for the two ϕ_a values after the PINN-estimated velocity and viscosity are substituted into it. The inverse PINN model is trained with three hidden layers with one hundred nodes each. In the direct suspension simulations, $\mu(u_y)$ is estimated using the Eilers model and $u(y)$ and $\phi(y)$ measurements.

using the Eilers formula [23, 24]

$$\eta_s(\phi) = \left(1 + \frac{5\phi}{4 \left(1 - \frac{\phi}{\phi_c} \right)} \right)^2. \quad (14)$$

Here, η_f is the fluid viscosity (which was set to unity in the FCM simulations) and ϕ_c is the maximum volume fraction of a suspension ($\phi_c = 0.62$ in the FCM simulations.) We observe that the PINN method is able to accurately learn the velocity profile with the maximum error of 1.27% for $\phi_a = 0.2$ and 4.73% for $\phi_a = 0.4$ and captures the increase in viscosity at the channel centerline. Figs. 4c and 4f demonstrate that the residuals are three orders smaller than $C = 0.0288$, indicating that the $\hat{u}(y; \theta)$ and $\hat{\mu}(\hat{u}_y(y; \theta); \gamma)$ DNNs satisfy Eq. (10).

Finally, we employ the inverse PINN method to evaluate η_s as a function of ϕ that is compared with the Eilers, Krieger [25], and Boyer et al [26] models in Fig. 6b. In the PINN method, we compute $\eta_s(\phi)$ using the

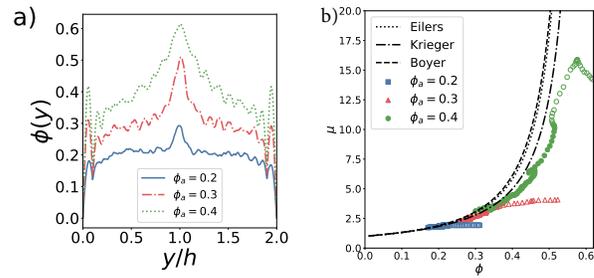


FIG. 5. a) Final suspension local volume fraction profiles $\phi(y)$ in steady-state. b) Suspension viscosities learned from the PINN model as a function of the local volume fraction. Results are compared with the Eilers [23, 24], Krieger [25], and Boyer et al [26] models fitted to the data. Filled symbols represent points that occur in the range $0h \leq y \leq 0.85h$, and empty symbols are in the range $0.85h \leq y \leq h$, to denote the deviations that occur from the theoretical values in the middle of the channel.

$\hat{\mu}(\hat{u}_y(y; \theta); \gamma)$ and $\hat{u}(y; \theta)$ DNN models of viscosity and velocity and $\phi(y)$ observed in the FCM simulations. The considered empirical models predict similar μ values for $\phi < 0.35$ away from the channel centerline. The empirical models assume that $\mu(\phi)$ is independent of ϕ_a . Fig. 6b shows that the PINN-predicted $\hat{\mu}(\phi)$ functions agree with the empirical models for small ϕ for all considered ϕ_a . For large ϕ , the PINN-estimated $\hat{\mu}(\phi)$ relationships depend on ϕ_a and deviate from all considered empirical models.

VI. DISCUSSION AND CONCLUSIONS

We formulated the physics-informed neural network (PINN) method for learning effective viscosity of the generalized Newtonian fluid from measurements of velocity and pressure in time-dependent three dimensional flows and used it to estimate viscosity models of two non-Newtonian systems (polymer melts and suspensions of particles) in shear flow between two parallel plates using only velocity measurements from numerical simulations. We observed the deviation of the PINN-estimated viscosity from viscosity given by the power-law model or directly estimated from numerical experiments for relatively small values of $|u_y|$. We note that for the considered shear flows, the problem of estimating $\mu(u_y)$ is ill-posed. For the shear flow described by Eq. (10), it follows that $u_y \mu = -C(y - H/2)$ and, at $y = H/2$, one has $0 \cdot \mu = 0$. For small but non-zero $|u_y|$ evaluation of viscosity using this relationship as

$$\mu = -C \frac{(y - \frac{H}{2})}{u_y} \quad (15)$$

is subject to numerical instabilities, i.e., small changes in the estimates of u_y from data (e.g., due to the measurement noise) can produce large changes in μ values. To

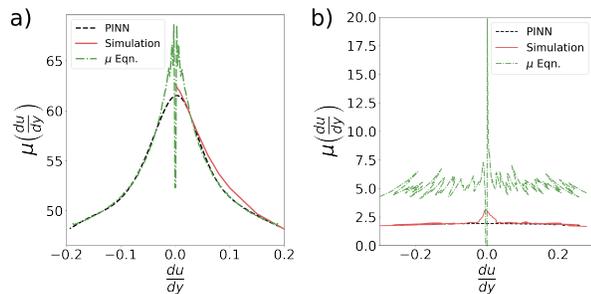


FIG. 6. Viscosity of (left) the polymer melt with $N = 2$ and (right) suspension with $\phi = 0.2$ obtained from the PINN method, direct numerical simulations (denoted as “Simulation”), and Eq (15) (denoted as “ μ Eqn.”).

demonstrate this point, in Figure 6a we compare the viscosities of the polymer melt with $N = 2$ computed with the PINN method, found in [17] from DPD simulations, and estimated from Eq. (15) where the derivatives u_y were computed from the measurements of u as a function of y using a fourth order centered finite difference scheme. The velocity data from DPD polymer melt simulations has relatively little noise and, for $|du/dy| > 0.4$, Eq. (15) predicts μ values that are in a close agreement with the PINN-estimate of μ . For $|u_y| < 0.4$, Eq. (15) produces very noisy estimates of $\mu(u_y)$ that explodes at $|u_y| = 0$. Noise in data makes the estimation of μ for small u_y even more difficult. For example, for the suspension with $\phi_a = 0.2$ (that has more noise in the velocity data than the polymer melts data), the μ estimates from

Eq. (15) are very noisy even for relatively large values of $|du/dy|$ and explode as $|du/dy| \rightarrow 0$, as can be seen in Figure 6b. Smoothing of the velocity data and/or fitting local polynomials to data would reduce the noise, but not completely eliminate it.

The PINN method regularizes this inverse problem by (1) using a global DNN approximation of μ and u , and (2) requiring μ and u to satisfy Eq. (10) at N_f points that could be significantly larger than the number of velocity measurements. Because of this, the PINN-estimate of μ is a monotonic function of $|u_y|$ even when the u measurements are noisy (as is the case with the considered suspension data.)

Finally, we note that a successful application of the PINN method to three-dimensional flow problems depends on the availability of data. As experimental and numerical techniques get more advanced, more data containing three-dimensional measurements of velocity and/or pressure fields become available. For example, large-scale high-fidelity numerical simulations of complex fluids provide full information about the three-dimensional velocity and pressure fields. Modern velocimetry methods, including particle image velocimetry, can provide three-dimensional velocity measurements in flow experiments.

ACKNOWLEDGMENTS

This research was partially supported by the U.S. Department of Energy (DOE) Advanced Scientific Computing (ASCR) program. Pacific Northwest National Laboratory is operated by Battelle for the DOE under Contract DE-AC05-76RL01830.

-
- [1] M. Raissi, A. Yazdani, and G. E. Karniadakis, Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations, *Science* **367**, 1026 (2020).
 - [2] Q. He and A. M. Tartakovsky, Physics-informed neural network method for forward and backward advection-dispersion equations, arXiv preprint arXiv:2012.11658 (2020).
 - [3] A. M. Tartakovsky, C. O. Marrero, P. Perdikaris, G. D. Tartakovsky, and D. Barajas-Solano, Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems, *Water Resources Research* **56**, e2019WR026731 (2020).
 - [4] Q. He, D. Barajas-Solano, G. Tartakovsky, and A. M. Tartakovsky, Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport, *Advances in Water Resources*, 103610 (2020).
 - [5] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, Automatic differentiation in machine learning: a survey, *The Journal of Machine Learning Research* **18**, 5595 (2017).
 - [6] Y. Hirose, K. Yamashita, and S. Hijiya, Back-propagation algorithm which varies the number of hidden units, *Neural networks* **4**, 61 (1991).
 - [7] S. Wang, X. Yu, and P. Perdikaris, When and why pinns fail to train: A neural tangent kernel perspective, arXiv preprint arXiv:2007.14527 (2020).
 - [8] D. C. Leigh, *Nonlinear continuum mechanics: an introduction to the continuum physics and mathematical theory of the nonlinear mechanical behavior of materials* (McGraw-Hill, 1968).
 - [9] S. Richardson, *Fluid mechanics*, new york, hemisphere pub (1989).
 - [10] M. Alves, P. Oliveira, and F. Pinho, Numerical methods for viscoelastic fluid flows, *Annual Review of Fluid Mechanics* **53**, 509 (2021).
 - [11] K. Xu, A. M. Tartakovsky, J. Burghardt, and E. Darve, Inverse modeling of viscoelasticity materials using physics constrained learning (2020), arXiv:2005.04384 [math.NA].
 - [12] X. Glorot and Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research, Vol. 9, edited by Y. W. Teh and M. Titterton (PMLR, Chia Laguna Resort, Sardinia, Italy, 2010) pp. 249–256.

- [13] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).
- [14] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, A limited memory algorithm for bound constrained optimization, *SIAM J. Sci. Comput.* **16**, 1190 (1995).
- [15] R. Bird, W. Stewart, and E. Lightfoot, *Transport Phenomena*, Wiley International edition (Wiley, 2006).
- [16] E. Hinch, Lecture 3: Simple flows.
- [17] D. A. Fedosov, G. E. Karniadakis, and B. Caswell, Steady shear rheometry of dissipative particle dynamics models of polymer fluids in reverse poiseuille flow, *The Journal of Chemical Physics* **132**, 144103 (2010).
- [18] J. Irving and J. G. Kirkwood, The statistical mechanical theory of transport processes. iv. the equations of hydrodynamics, *The Journal of chemical physics* **18**, 817 (1950).
- [19] A. Howard, *Numerical Simulations to Investigate Particle Dispersion in Non-Homogeneous Suspension Flows*, Ph.D. thesis, Brown University, Providence, RI (2018).
- [20] K. Yeo and M. R. Maxey, Simulation of concentrated suspensions using the force-coupling method, *J. Comput. Phys.* **229**, 2401 (2010).
- [21] K. Yeo and M. R. Maxey, Numerical simulations of concentrated suspensions of monodisperse particles in a Poiseuille flow, *J. Fluid Mech.* **682**, 491 (2011).
- [22] D. Leighton and A. Acrivos, The Shear-Induced Migration of Particles in Concentrated Suspensions, *J. Fluid Mech.* **181**, 415 (1987).
- [23] F. Ferrini, D. Ercolani, B. de Cindio, L. Nicodemo, L. Nicolais, and S. Ranaudo, Shear viscosity of settling suspensions, *Rheol. Acta* **18**, 289 (1979).
- [24] J. J. Stickel and R. L. Powell, Fluid Mechanics and Rheology of Dense Suspensions, *Annu. Rev. Fluid Mech* **37**, 129 (2005).
- [25] I. M. Krieger and T. J. Dougherty, A Mechanism for Non-Newtonian Flow in Suspensions of Rigid Spheres, *Trans. Soc. Rheol.* **3**, 137 (1959).
- [26] F. Boyer, É. Guazzelli, and O. Pouliquen, Unifying suspension and granular rheology, *Physical Review Letters* **107**, 188301 (2011).