Learning Unknown Physics of non-Newtonian Fluids

Brandon Reyes and Amanda A. Howard
Pacific Northwest National Laboratory, Richland, WA

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

(Dated: August 18, 2020)

We use physics-informed neural networks (PINNs) to learn viscosity models of two non-Newtonian systems (polymer melts and suspensions of particles) using only velocity measurements. For synthetic velocity data generated with the power-law viscosity model, the PINN-inferred viscosity model agrees with the analytical model for shear rates with large absolute values but deviates for shear rates near zero where the analytical model has an unphysical singularity. Once the viscosity model is learned the PINN method can solve the momentum conservation equation using only the boundary conditions.

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. For example, in non-Newtonian flow experiments one can easily measure velocity, but not stress or viscosity. This makes it impossible to use data-driven ML methods to learn stress as a function of velocity or shear rate. Also, the momentum and mass conservation equations governing non-Newtonian flow are not fully known as one needs to assume a stress-shear-rate relationship (we refer to such relationships as unknown physics) to close the system of these equations. 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. PINNs use the known underlying structure of physical laws governed by PDEs or ODEs to predict unknown functions or functionals from indirect observations. By representing states of the system and hidden physics with neural networks and training using available data subject to the conservation laws, the PINN method can learn unknown physics using sparse and indirect data. In the past, the PINN method was used to learn unknown physics in partially unsaturated flow in porous media [1] and bioreactors [2]. In this work, we extend the PINN method for estimating the non-Newtonian viscosity based solely on velocity data.

PINN METHOD FOR NON-NEWTONIAN FLOW MODELS

Consider a shear flow of a non-Newtonian fluid between two parallel plates satisfying the steady-state momentum conservation equation:

$$\frac{d}{dy}\left[\mu(u_y(y))\frac{du(y)}{dy}\right] = -C \quad \text{for } y \in \Omega = (0, H), \quad (1)$$

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

$$u(0) = 0, \quad u(H) = 0. \quad (2)$$

We consider two cases: no measurements of μ are available and some measurements of μ are present. In both cases we assume that there are N_u measurements of the velocity profile $u(y)$ for $y \in \Omega$: $u^*(y_i)$ for $i = 1, \ldots, N_u$. We approximate the viscosity $\mu(u_y)$ and the velocity $u(y)$ with fully connected feed-forward deep neural networks (DNNs), $u(y) \approx \hat{u}(y;\theta)$ and $\mu(u_y(y)) \approx \hat{\mu}(\hat{u}_y(y;\theta);\gamma)$, where θ and γ are the DNN weights. We train $\hat{u}$ and $\hat{\mu}$ jointly using Eqs. (1) and (2) as constraints. This allows us to train $\hat{\mu}$ even without direct measurements of μ.

We note that the DNNs $\hat{u}$ and $\hat{\mu}$ are known non-linear functions of y and θ and/or γ. Therefore, we can analytically compute the DNN derivatives with respect to y and the weights. The former are needed to impose the physical constraints given by Eq. (1), while the latter are required to update the values for the weights in the process known as backpropagation. Here, we use automatic differentiation to compute the derivatives.

Eq. (1) is enforced in the DNN training by forming an additional "auxiliary" DNN:

$$\hat{f}(y;\theta,\gamma) = \frac{d}{dy}\left[\hat{\mu}(\hat{u}_y(y;\theta);\gamma)\frac{d\hat{u}(y;\theta)}{dy}\right]. \quad (3)$$

We train the DNNs simultaneously by minimizing the

loss function

$$L(\theta,\gamma) = \frac{\omega_1}{N_u}\sum_{i=1}^{N_u}\left[\hat{u}(y_i;\theta) - u^*(y_i)\right]^2 + \frac{\omega_2}{2}\left[\hat{u}(y=0;\theta)^2 + \hat{u}(y=H;\theta)^2\right] + \frac{\omega_3}{N_u}\sum_{i=1}^{N_u}\left[\hat{f}(y_i;\theta,\gamma)+C\right]^2 + \frac{\omega_4}{N_\mu}\sum_{i=1}^{N_\mu}\left[\hat{\mu}(u_{y_i};\gamma) - \mu^*(u_{y_i})\right]^2. \quad (4)$$

In $L(\theta,\gamma)$, the first term forces $\hat{u}(y;\theta)$ to match the velocity measurements, the second term forces $\hat{u}(y;\theta)$ to match the Dirichlet BCs, and the third term forces $\hat{u}$ and $\hat{\mu}$ to satisfy Eq. (1). The last term is present ($\omega_4 \neq 0$) if measurements of μ (i.e., $\mu^*(u_{y_i})$ for $i = 1, ..., N_\mu$) are available and forces $\hat{\mu}$ to match these measurements. The weights $\{\omega_i\}_{i=1,...,4}$ reflect the fidelity level of the data and physics models. For example, u measurements are more accurate than viscosity measurements in general, so $\omega_1 \geq \omega_4$. We note that Eq. (1) is an approximation of the momentum conservation equation because it involves assumptions about the general form of the viscous stress, therefore, $\omega_3 \leq \omega_1$. For some flows the no-slip BCs assumption might not be very accurate, which would affect the relative value of ω_2. The relative values of ω_i can also affect the convergence rate of iterative solutions of the minimization problem $(\theta,\gamma) = \arg\min_{\theta,\gamma} L(\theta,\gamma)$ [3].

To solve this minimization problem, we set the initial values of θ and γ using the Xavier's normal initialization scheme [4]. Next, we run the Adam optimizer [5] for a set number of steps. Finally, we run the quasi-Newton L-BFGS-B optimizer [6] until the desired convergence and tolerance are achieved, because the L-BFGS-B optimizer has a superior rate of convergence and more favorable computational cost for problems with a relatively small number of observed data. We use DNNs with two hidden layers with sixty nodes each and a learning rate of 0.001 for the Adam optimizer. The error $\|e_{\hat{u}}\|_2 = \|\hat{u}(y;\theta) - u^*(y)\|_2 / \|u^*(y)\|_2$ 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. (1).

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. (1) we use the loss function Eq. (4) with $\omega_1 = \omega_4 = 0$ and $\omega_2 = \omega_3 = 1$. We refer to this application of PINNs as the forward PINN method.

n	$\|e_{\hat{u}}\|_2$	$\|f\|_\infty$
0.898	2.345×10^{-4}	7.45×10^{-5}
1.2	2.135×10^{-4}	5.061×10^{-4}

TABLE I. Mean errors computed from 100 inverse PINN solutions for the synthetic data generated from the analytical solution for a power-law fluid with $n = 0.898$ and 1.2.

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 [7], $\mu_{pl}(u_y(y)) = K|u_y(y)|^{n-1}$, where K is the power-law consistency coefficient and n is the power-law index. This model in combination with Eqs. (1) and (2) allows for an analytical solution for $u(y)$ and $du(y)/dy$ [8].

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. (4) 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. The average DNN $\hat{\mu}(\hat{u}_y)$ solution agrees very well the constitutive model for large shear rates. 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 standard deviation in the learned $\mu(u_y)$ is largest at $u_y = 0$ and is several orders of magnitude smaller than the mean value of μ at $u_y = 0$, indicating that the uncertainty of the PINN method due to DNN initialization is relatively small. Fig. 1c depicts the residual $\hat{f}(y;\theta,\gamma)+C$ of Eq. (1) as a function of y. The small values of the residual show that the DNNs $\hat{u}$ and $\hat{\mu}$ approximately satisfy Eq. (1). The $\|e_{\hat{u}}\|_2$ and $\|f\|_\infty$ errors for both values of n are given in Table I. Small errors demonstrate that the inverse PINN method is equally accurate for both shear-thinning and shear-thickening fluids.

Next, we validate the ability of the forward PINN method to solve Eq. (1). We fix the weights of the DNN

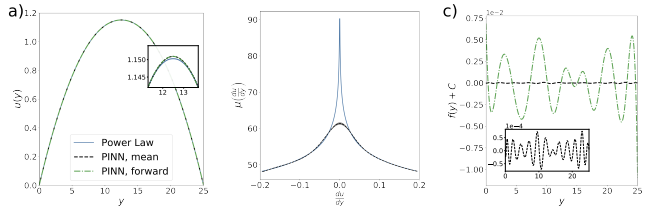

FIG. 1. Inverse and forward PINN solutions for the synthetic data generated from the analytical solution for a power-law fluid with $n = 0.898$. (a) The average velocity profile from 100 runs for the PINN results, and one run for the PINN forward model. (b) The average $\frac{du}{dy}$ of the 100 runs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

$\hat{\mu}$ obtained from the inverse PINN with $n = 0.898$ and train the $\hat{u}(y, \theta)$ DNN by minimizing the loss function Eq. (4) 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 $n = 0.898$. 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. (1). The good agreement between $\hat{u}$ and the reference solution for u and small residuals confirm the accuracy of the forward PINN method for solving non-linear differential equations with constitutive relationships given by a DNN with known weights.

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 [9]. In [9], the DPD results were used to compute $\mu_{DPD}(du(y)/dy)$ using the Irving-Kirkwood relationship [10].

We use the velocity data from [9] and the inverse PINN method with $\omega_1 = \omega_2 = \omega_3 = 1$ and $\omega_4 = 0$ in Eq. (4) to estimate $\mu(u_y)$. To match [9] , $C = 0.75$ and $H = 25$. The relative velocity error and the maximum residual error are given in Table II. For all considered N the relative error in u is less than 0.1% and the maximum residual error is 3 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 $u_{DPD}(y)$. The agreement between $\hat{\mu}(u_y; \theta, \gamma)$ and $\mu_{DPD}(u_y)$ is good but less accurate than the agreement for the velocities. To test whether $u_{DPD}(y)$ and $\mu_{DPD}(u_y)$ satisfy Eq. (1), we train the $\hat{u}(y; \theta)$ and $\hat{\mu}(u_y; \theta, \gamma)$ DNNs conditioned on both u and μ DPD measurements. Figs. 2a and 2b show that conditioning of the DNNs on the DPD measurements of u and the estimates of μ produces DNNs that match well both u and μ data. However, conditioning on the DPD μ estimates also results in the residual errors that are two orders of magnitude larger than the residual errors in the case where no μ estimates are used to train the DNNs, as shown in Fig. 2c.

N	$\|e_{\hat{u}}\|_2$	$\|f\|_\infty$
2	2.568×10^{-4}	1.544×10^{-4}
5	1.921×10^{-4}	1.773×10^{-4}
25	4.396×10^{-4}	1.201×10^{-4}

TABLE II. Errors for $N = 2$, 5, and 25 beads

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 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. 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 $\mu_{DPD}(y)$ data well, but the corresponding residual is very large (on the order of C). 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. (1) subject to the BC Eq. (2). 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. (4) 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 corresponding to $C = 0.75$ well. In addition to this, Fig. 2c demonstrates that the residual of the governing equation is two orders of magnitude smaller than C confirming that $\hat{u}_f(y; \theta)$ approximately solves Eq. (1) subject to Eq. (2).

These results lead to the conclusion that the inverse PINN is capable of estimating the effective viscosity function $\mu(u_y)$, which can be used for solving the momentum conservation equation (1). The predicted viscosity deviates from the viscosity obtained from the DPD simulations for small shear rates with the discrepancy increasing with the number of beads N. Our results show that velocity and viscosity data provided in [9] cannot accurately

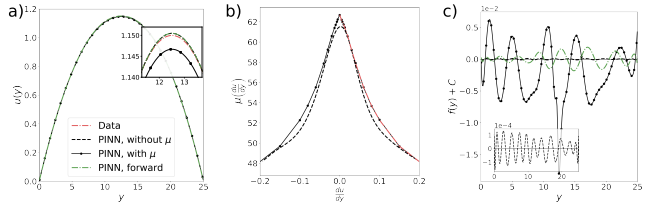

FIG. 2. Inverse and forward model results for $N = 2$. (a) Resulting velocity profiles. (b) Resulting viscosity profile. (c) Error in satisfying the ODE. In the power-law solution we take $n = 0.90$ and $K = 40.79$.

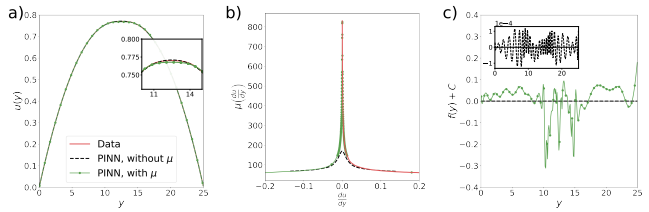

FIG. 3. PINN results for $N = 25$. (a) Resulting velocity profiles. (b) Resulting viscosity profile. (c) Error in satisfying the ODE. In the power-law solution we take $n = 0.79$ and $K = 43.78$.

be described by Eq. (1).

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 [11]. The considered data are obtained from the numerical simulations of suspension flows in a channel using the Force Coupling Method (FCM) [11, 16, 18]. In the considered suspensions, the average particle volume fraction $\phi_a = \frac{\frac{4}{3}\pi a^3 N}{V}$ 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. (1) 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 5a. A key feature of suspensions is irreversible shear-induced migration of particles to areas of low shear rate [12]. Particles in a suspension subjected to a Poiseuille flow will migrate to the channel centerline, greatly increasing the volume fraction at the centerline until it reaches the maximum close-packing limit, as shown in Fig. 5a. This migration also impacts the velocity profile, resulting in a flattened parabola shape 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 (4) 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. (1), 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 are 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 using the Eilers formula $\eta_s(\phi) = \left(1 + \frac{5\phi}{4\left(1-\frac{\phi}{\phi_c}\right)}\right)^2$ [19, 20]. 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 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. (1).

Finally, we employ the inverse PINN method to evaluate η_s as a function of ϕ and in Fig. 5b compare it with the Eilers model and the Krieger model $\eta_s(\phi) = \left(1 - \frac{\phi}{\phi_c}\right)^{-2.5\phi_c}$ [21]. In the PINN method, we compute $\eta_s(\phi)$ using the $\hat{\mu}(\hat{u}_y(y;\theta);\gamma)$ 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. 5b shows that the PINN predicted $\mu(\phi)$ functions agree with the empirical models for small ϕ for all considered ϕ_a. For large ϕ, the PINN estimated $\mu(\phi)$ relationships depend on ϕ_a and deviate from all considered empirical models. There are several reasons for the disagreement between the PINN estimated and empirical viscosity models. At high volume fractions near the close-packing limit, ϕ_c, the particle movements are highly

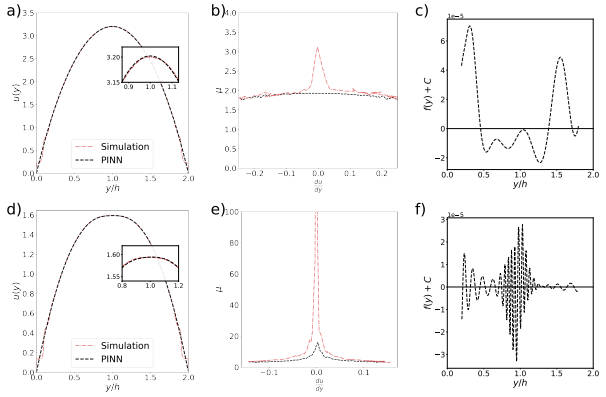

FIG. 4. Inverse PINN results for suspensions with average volume fraction $\phi_a = 0.2$ (a-c) and 0.4 (d-f). The velocity profiles, viscosity profiles, and relative errors in satisfying the ODE are shown. The inverse PINN model is trained with three hidden layers with one hundred nodes each.

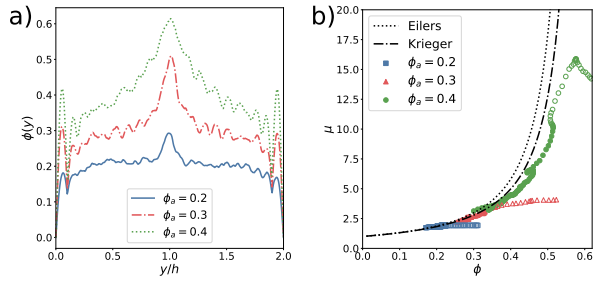

FIG. 5. a) Final suspension local volume fraction profiles $\phi(y)$ in steady-state. b) Suspension viscosities learned from the PINN model as a function of the local volume fraction. Results are compared with the Eilers fit [19, 20] and the Krieger fit [21]. Filled symbols represent points that occur 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 densely packed core of the channel.

correlated leading to non-locality of the particle forces. Therefore, Eq. (1) breaks down for dense suspensions at the centerline. Additionally, Eq. (1) with a constitutive relationship of the Eilers or Krieger analytical forms predicts that the suspension will reach maximum packing with $\phi = \phi_c$ at the centerline that is independent of the average volume fraction ϕ_a [22]. However, FCM simulations [16] and experiments [13] show that the volume at the centerline varies with the initial average volume fraction of the system. It is not surprising that the PINN model (which enforces Eq. (1)) struggles at the centerline, where the dynamics are complex and lubrication forces and particle contacts dominate the particle migration.

In conclusion, we have extended the PINN method for learning unknown physics, including the functional dependence of viscosity on the shear rate and other properties of fluids using indirect measurements such as fluid velocity and volume fraction. We have also demonstrated that once an accurate DNN approximation of the viscosity is available, the PINN method can be used to model non-Newtonian flow without any data except the boundary conditions.

* amt1998@illinois.edu

[1] A. M. Tartakovsky, C. O. Marrero, P. Perdikaris, G. D. Tartakovsky, and D. Barajas-Solano, Water Resources Research **56**, e2019WR026731 (2020).

[2] R. Tipireddy, P. Perdikaris, P. Stinis, and A. Tartakovsky, arXiv preprint arXiv:1904.04058 (2019).

[3] S. Wang, Y. Teng, and P. Perdikaris, arXiv preprint arXiv:2001.04536 (2020).

[4] X. Glorot and Y. Bengio, in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, edited by Y. W. Teh and M. Titterington (PMLR, Chia Laguna Resort, Sardinia, Italy, 2010), vol. 9 of *Proceedings of Machine Learning Research*, pp. 249–256.

[5] D. P. Kingma and J. Ba, arXiv preprint arXiv:1412.6980 (2014).

[6] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, SIAM J. Sci. Comput. **16**, 1190 (1995).

[7] R. Bird, W. Stewart, and E. Lightfoot, *Transport Phenomena*, Wiley International edition (Wiley, 2006), ISBN 9780470115398.

[8] E. Hinch, *Lecture 3: Simple flows*, URL `https://www.whoi.edu/cms/files/lecture03_28327.pdf`.

[9] D. A. Fedosov, G. E. Karniadakis, and B. Caswell, The Journal of Chemical Physics **132**, 144103 (2010).

[10] J. Irving and J. G. Kirkwood, The Journal of chemical physics **18**, 817 (1950).

[11] A. Howard, Ph.D. thesis, Brown University, Providence, RI (2018).

[12] D. Leighton and A. Acrivos, J. Fluid Mech. **181**, 415 (1987).

[13] M. K. Lyon and L. G. Leal, J. Fluid Mech. **363**, 25 (1998).

[14] J. E. Butler, P. D. Majors, and R. T. Bonnecaze, Phys. Fluids **11**, 2865 (1999).

[15] B. Snook, J. E. Butler, and É. Guazzelli, J. Fluid Mech. **786**, 128 (2015).

[16] K. Yeo and M. R. Maxey, J. Fluid Mech. **682**, 491 (2011).

[17] F. Cui, A. Howard, M. Maxey, and A. Tripathi, Phys. Rev. Fluids **2** (2017).

[18] K. Yeo and M. R. Maxey, J. Comput. Phys. **229**, 2401 (2010).

[19] F. Ferrini, D. Ercolani, B. de Cindio, L. Nicodemo, L. Nicolais, and S. Ranaudo, Rheol. Acta **18**, 289 (1979).

[20] J. J. Stickel and R. L. Powell, Annu. Rev. Fluid Mech **37**, 129 (2005).

[21] I. M. Krieger and T. J. Dougherty, Trans. Soc. Rheol. **3**, 137 (1959).

[22] E. Guazzelli and O. Pouliquen, Journal of Fluid Mechanics **852** (2018).

[23] G. Batchelor and J.-T. Green, Journal of Fluid Mechanics **56**, 375 (1972).