2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Differentiable Fluid Physics Parameter Identification By Stirring and For Stirring

Wenqiang Xu*[1], Dongzhe Zheng*[2], Yutong Li[1], Jieji Ren[1], Cewu Lu[1]

***Abstract*— Fluid interactions are crucial in daily tasks, with properties like density and viscosity being key parameters. The property states can be used as control signals for robot operation. While density estimation is simple, assessing viscosity, especially for different fluid types, is complex. This study introduces a novel differentiable fitting framework, DiffStir, tailored to identify key physics parameters through stirring. Then, given the estimated physics parameters, we can generate commands to guide the robotic stirring. Comprehensive experiments were conducted to validate the efficacy of DiffStir, showcasing its precision in parameter estimation when benchmarked against reported values in the literature. More experiments and videos can be found in the supplementary materials and on the website: `https://diffstir.robotflow.ai`.**

I. INTRODUCTION

Humans interact with fluid every day, from mixing up coffee powder into water to squeezing toothpaste. Understanding fluid is the key to fluid-related manipulation tasks. In domestic scenarios, density and viscosity are two important physics properties to identify. For example, in the kitchen, adjusting density usually relates to adjusting the extent of a certain flavor (e.g., sweetness and saltness). Viscosity can indicate the solidity of the pastry, which is usually adopted by human to decide when to accelerate or stop the stirring operation. Thus, a robot system that can estimate the fluid physics parameters and generate the commands based on the parameters will benefit domestic robots in kitchen scenarios.

Though density can be easily estimated using Archimedes' principle, viscosity is more complex. In hydraulic mechanics, the dynamics behaviors of fluids are described by different constitutive models [1], [2], [3], which means they can be characterized by a few parameters given the predefined stress-strain relationship. Based on the stress-strain relationship, fluids can be categorized in *Newtonian* (Fig. 1-a) and *non-Newtonian* (Fig. 1-b) types. Newtonian fluids are characterized by a linear stress-strain relationship. Meanwhile, non-Newtonian fluids have varied viscosity based on the rate of deformation caused by shear stress. These fluids are typically described by constitutive models such as the Carreau [4], Cross [5] and Herschel-Bulkley [6] models.

In this work, we present a novel differentiable fitting framework named **DiffStir** to identify the key physics parameters through a simple action, stirring, a common operation in the kitchen. It utilizes a robot arm to grasp a rod and conducts the robotic stirring operation on a cup of certain fluid with recorded trajectories and forces. After synchronizing the action and force between the real-world configuration and a differentiable Material Point Method (diffMPM) [7]-based simulator, we can estimate the fluid parameters of the pre-set constitutive models by matching the observations from both worlds. Different non-Newtonian constitutive models have their own preferences for fluid. For example, the Carreau model is suitable for shear-thinning fluids that can approach Newtonian behaviors at high shear rates; the Cross model is good for shear-thinning and shear-thickening fluids across a wide shear rate range; and the Herschel-Bulkley model is best for fluids with a yield stress, such as muds and some toothpaste. Based on it, we design an online strategy to adaptively select the most proper constitutive models given the real-world recordings. Finally, the viscosity of complex fluids can change during stirring due to various physicochemical transformations. Considering the process of stirring the dough, the viscosity changes with stirring, and different extents of viscosity have different practical meanings in cooking, thus it can be used as command signals for robot operation in robot-assisted cooking applications.

Fig. 1: (a) Newtonian fluids: distilled water, red wine, black coffee, and cooking oil. Despite their varied molecular structures, fluids in (a) consistently obey Newtonian principles. (b) Non-Newtonian fluids and their properties: toothpaste (Bingham plasticity), Oobleck (shear-thickening, impact resistance), chocolate sauce (thixotropy), and Bulgarian yogurt (viscoelasticity).

To evaluate the proposed DiffStir, we conduct extensive experiments from three aspects to demonstrate the accuracy of the estimated parameters with reference values from the literature [8], [9], [10]: density estimation, apparent viscosity estimation, and mixture ratio inference. These tasks are crucial in many fluid-related scenarios such as kitchen and chemistry laboratory.

We summarize our contributions as follows:

*Equal contribution

[1]{vinjohn, davidliyutong, jiejiren, lucewu}@sjtu.edu.cn. Cewu Lu is the corresponding author, a member of Qing Yuan Research Institute and MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, China.

[2]dz1011@wildcats.unh.edu. Work is done when Dongzhe is an intern at SJTU.

- We propose to identify the key physics parameters, density, and viscosity, of fluid in household scenarios through a common operation, stirring. The identification process bridges the real world and the differentiable simulator and can achieve fluid understanding during robotic operation.
- We conduct extensive experiments on different Newtonian and non-Newtonian fluids to validate the usability and accuracy of our system. We hope our system can serve as a foundational tool for liquid-related manipulation tasks.

II. Related Works

A. Fluid Constitutive Model

Fluid dynamics research has provided insights into complex phenomena, such as viscoelasticity, viscoplasticity, and elastoplasticity. Viscoelastic models like Maxwell [11] and Oldroyd-B [12] are fundamental in understanding biological and polymeric fluids. Elastoplastic models, such as Prandtl-Reuss [13] and Desai [14], capture behaviors seen in foams and slurries. Viscoplastic models, including Casson [15], Sisko [16], and Bingham [17], emphasize yield stress in substances like blood, toothpaste, and mayonnaise. In our work, we integrate the Carreau, Cross, Herschel-Bulkley [18], and Kelvin-Voigt models [19] to represent diverse fluid behaviors, capturing shear behaviors, yield stress, and viscoelastic properties.

B. Fluid Identification Method

Fluid parameter identification uses instrument-based and computation-based methods. Instrument-based methods, such as falling ball viscometers, oscillatory shear tests, and rotational viscometry, have limitations in daily applications [20], [21], [22], [23]. Our approach integrates differentiable physics simulation and robotic feedback to augment these techniques.

Computational methods include CFD [24], [25], [26], which can identify parameters but may be computationally demanding. Our method follows conservation laws and Navier-Stokes equations, maintains CFD constitutive relations, and simplifies computation using MLS-MPM for handling large deformations.

Data-driven approaches, like machine learning and neural networks [27], [28], [29], [30], can predict parameters but may lack physical validation. Our technique emphasizes a robust physical framework for accuracy and stability.

C. Fluid Simulation for Robot

Many robotic simulation platforms exist, but simulating non-Newtonian fluids remains challenging. PyBullet [31] and MuJoCo [32] lack fluid simulation capabilities. FleX [33], SoftGym [34], SAPIEN [35], and TDW [36] focus on Newtonian fluid dynamics, with limitations in closed source, scalability, and physics interaction. These platforms are restricted to Newtonian fluids, neglecting non-Newtonian fluid intricacies prevalent in real robotic scenarios.

Differentiable platforms like SPNets [37] and PHIFLOW [38] have been proposed. SPNets is limited in single-phase non-Newtonian fluids, while PHIFLOW faces challenges integrating solid interactions.

Our method, compared to FluidLab [39], employs the Kelvin-Voigt viscoelastic model to capture non-Newtonian fluid behaviors, covering shear thinning to thickening. FluidLab omits essential fluid parameters like viscosity and complex rheology phenomena.

III. Preliminary on Fluid Constitutive Models

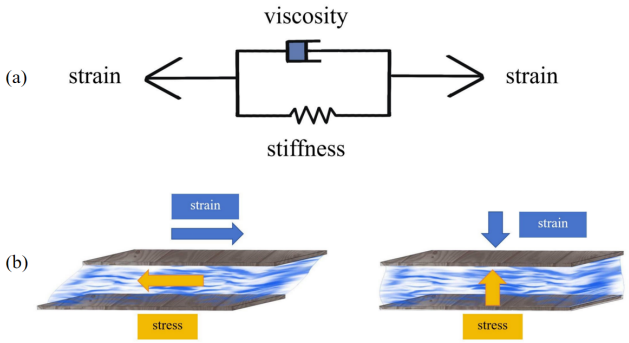

Fig. 2: (a) Kelvin-Voigt model considers fluid viscosity and elasticity. (b) Two sub-figures representing fluid dynamics: relative sliding between plates (dynamic viscosity and shear resistance) and a downward force on a plate (bulk modulus and compressibility).

1) Kelvin-Voigt Fluid Model Integrating Volume-Related Energy Density:

In this work, we use the Kelvin-Voigt model as the constitutive model [19] to describe the fluid's properties and behavior. The Kelvin-Voigt model combines a spring (elasticity) and a dashpot (viscosity) in parallel (Fig. 2(a)) to represent materials likes polymeric solutions, foams and gels [40]. The Kelvin-Voigt model uses a bulk modulus κ for elastic response to volume change. For non-linear elastic deformation, the Neo-Hookean model describes the reaction through the energy density function W_v, related to the volume ratio J:

$$W_v(J) = \frac{\kappa}{2}\left(\frac{1}{2}(J^2-1) - \ln J\right) \tag{1}$$

Differentiating W_v, we obtain stress τ_v, and incorporating it into the Kelvin-Voigt model, then the total stress, σ, is calculated as:

$$\sigma = E\varepsilon + \eta\dot{\varepsilon} = \frac{\kappa}{2}(J - \frac{1}{J})I + \eta\dot{\epsilon} \tag{2}$$

Here, ε is the strain, E is the elastic modulus, $\dot{\varepsilon}$ is the strain rate, η is the viscosity parameter, and the identity tensor I, reflecting its isotropic effect.

Common household fluids are hard to significantly change the volume under external force. In simulation, volume-related stress is applied to maintain volume. Without this term, the simulated volume will shrink.

2) Extending to Non-Newtonian Fluids:

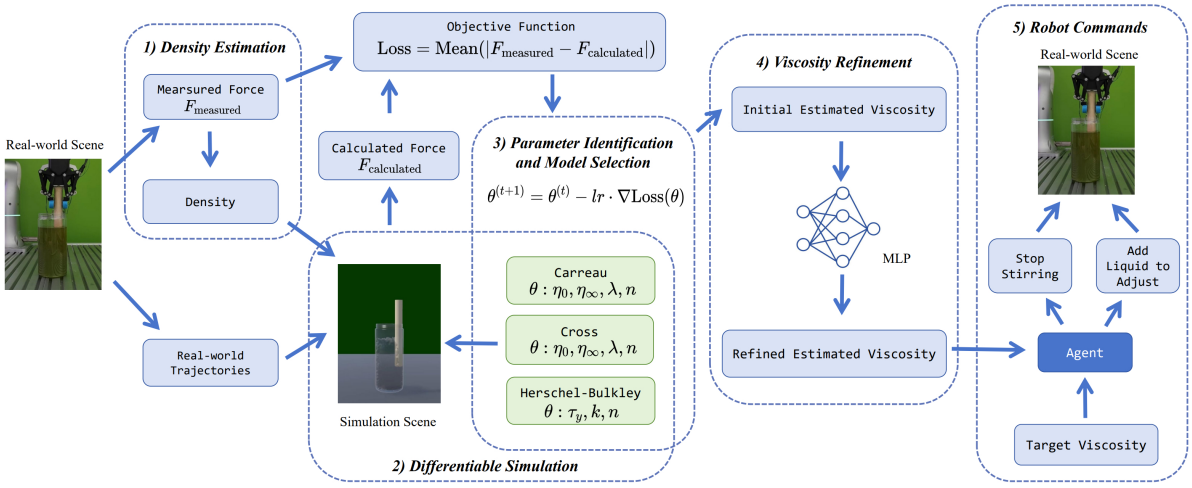

Fig. 3: Pipeline of **DiffStir**. 1) It takes the recorded real-world trajectories and the forces of a robot grasping a rod and stirring in a container. With the measured force, it can estimate the fluid density. 2) The estimated density along with the trajectories can help to replicate the robotic stirring scene in a differentiable simulator. 3) By comparing the force calculated from the simulator and the real-world measurements, it can adaptively find the most proper parameters and models. 4) It adopts a neural network to refine the estimated parameters. 5) Finally, in the application of mixture ratio inference, by comparing the recognized refined viscosity with the target viscosity, the agent makes decisions and outputs operational commands to adjust the ratio, helping to obtain liquid mixtures that meet the target viscosity in robotics operations.

To fully grasp fluid behavior, we combine continuum mechanics with non-Newtonian fluid models, focusing on Carreau, Cross, and Herschel-Bulkley [41]. These models explain the changing relationship between viscosity η and shear rate $\dot{\gamma}$, using the **apparent viscosity** function $\eta(\dot{\gamma})$. Take the Carreau model as an example, the apparent viscosity is:

$$\eta(\dot{\gamma}) = \eta_\infty + (\eta_0 - \eta_\infty)(1 + (\lambda\dot{\gamma})^2)^{(n-1)/2}, \quad (3)$$

The fluid's stress, σ, is a function of the parameters $\theta = (\eta_0, \eta_\infty, \lambda, n)$:

$$\sigma(\theta) = \tau_v I + \left(\eta_\infty + (\eta_0 - \eta_\infty)(1 + (\lambda\dot{\gamma})^2)^{(n-1)/2}\right)\dot{\epsilon}. \quad (4)$$

More details can be found in [41] and our supplementary materials. As shown in Fig. 3, the parameters θ for these models are: (a) Carreau Model: $(\eta_0, \eta_\infty, \lambda, n)$, (b) Cross Model: $(\eta_0, \eta_\infty, \lambda, n)$, (c) Herschel-Bulkley Model: (τ_y, n, k), where η_0 is the zero shear rate viscosity, η_∞ is the infinite shear rate viscosity, λ is a multiplier constant, n is the power-law index, τ_y is the yield stress, and k is a consistency index. The Herschel-Bulkley viscosity can be derived from the estimated yield (see supplementary materials).

IV. DiffStir Method

In this section, we describe the proposed DiffStir system. Density and viscosity are estimated individually and sequentially. The overall pipeline is illustrated in Fig. 3.

1) Density Estimation: By integrating a force sensor at a robotic arm's end-effector, our method estimates density via force measurement on a submerged stirring rod. The density ρ is calculated with Archimedes' principle as:

$$\rho = \frac{G - R}{Vg}, \quad (5)$$

where G is the gravitational force of the stirring rod, R is the equilibrium force measured by the force sensor, V is the volume of the immersed part of the rod, and the g is the gravitational acceleration. G, R, V can all be measured from the real world, the experimental setup of measurement is described in Sec. V-C.1.

2) Differentiable Simulation: To further estimate the viscosity, we build a bridge between the real and simulated observation via a differentiable simulator. We first replicate the setup of the robotic stirring in the real world to the simulator. The density estimated from the last step is used to set up the fluid in the simulated world. Then, we conduct the stirring operation in the real world, and transfer the recorded trajectories (e.g. rod positions, velocities) and the force applied to the rod to the simulated environment. The physics engine MLS-MPM [42] in the simulator can produce the interaction force between the fluid and the rod, $F_{\text{calculated}}$. The fluid is modeled by multiple particles, and the particles are moved by the stirring operation. The particle dynamics is governed by a constitutive model. Thus, the same stirring operation can cause different particle movements due to different constitutive models. If the constitutive model contains the parameter of viscosity, the particle movement can exhibit the effect of drag and horizontal shear. In this sense, the reactive $F_{\text{calculated}}$ is parametric.

In our simulation, we represent the fluid as a collection of particles. Each particle's stress, σ, is governed by a constitutive relation. The force exerted on a single particle due to this stress can be computed as $\mathbf{F}_{\text{particle}}(\theta) = \int \sigma(\theta) \cdot \mathbf{n}\, dA$, where $\mathbf{n}$ is the outward normal of the particle's surface and A is the area over which the stress acts. The cumulative force exerted on the stirring rod, resulting from all particles in contact with it, is given by $F_{\text{calculated}}(\theta) = \sum_{i=1}^{N} \mathbf{F}_{\text{particle}_i}(\theta)$.

Here, N means the number of particles interacting with the rod.

3) Parameter Identification and Model Selection: We attach a thin-film pressure sensor to the gripper finger and measure the force from the rod, F_{measured}. By assuming no contact between the rod and the liquid container, the force applied to the pressure sensor is the same as the force applied to the rod from the fluid. In this way, we can compare the measured force F_{measured} in the real world, and the calculated force $F_{\text{calculated}}$ in the simulator, and construct a loss term to optimize the parameters θ associated with $F_{\text{calculated}}$:

$$\text{Loss}(\theta) = \text{Mean}\left(|F_{\text{measured}} - F_{\text{calculated}}(\theta)|\right). \quad (6)$$

By leveraging the power of differentiable simulation, optimization of θ can be simply achieved using the gradient descent algorithm, expressed as:

$$\theta^{(t+1)} = \theta^{(t)} - lr \cdot \nabla \text{Loss}(\theta) \quad (7)$$

lr denotes the learning rate, t is the iteration, and $\nabla \text{Loss}(\theta)$ is the gradient of the loss function concerning θ.

Different constitutive models prefer specific fluids; thus, our system uses the Carreau, Cross, and Herschel-Bulkley models to suit different fluids. We employ an adaptive method to select the most suitable model and parameters, basing our choice on the lowest error from Eq. 6. Notably, while these models predominantly describe non-Newtonian fluids, they can capture Newtonian behaviors under specific conditions, e.g., the Cross model when the power-law index n nears zero.

4) Viscosity Refinement: Given the numerical differences and measurement noises between simulations and real-world setups, our system uses a multi-layer perceptron (MLP) network to correct these systematic errors. We generate 2500 estimated/reference viscosity pairs as the training samples. We leave the details of the training set construction in supplementary materials.

Our MLP rectifier has an architecture that consists of two input neurons (representing average stirring rate and preliminary viscosity), a hidden layer with 8 neurons, and a single output neuron for refined viscosity. For training, we set a learning rate of 0.01, achieving a balance between speed and stability in convergence. The model underwent 50 epochs of training, allowing for sufficient learning and weight adjustments.

5) Robot Commands: The estimated parameters can be used as command signals for robot operations. Here we give two examples of robots. First, the robot can automatically stop stirring when the mixture reaches the target viscosity. Second, the robot can provide prompts to add extra liquid. These two operations are common in daily household tasks. We conduct these operations in real robot experiments of mixture ratio inference.

V. Experimental Setup

A. Real-World Setup

To ensure rigorous experimentation for fluid dynamics during stirring, we design a precise setup, emphasizing alignment between simulation and real-world conditions (Fig. 3).

1) Setup Specifications: We utilize a cylindrical PET container (diameter: 8.5cm, height: 16.5cm) and fill it with 738ml of fluid. A cylindrical wooden stirring rod (length: 20cm, diameter: 2.5cm) is submerged 8cm into the fluid.

2) Force Sensing: We sense the reactive force with an RP-C18.3-ST resistive thin-film pressure sensor with a diameter of 18.3mm on the robotic gripper, DH AG95, detecting force variations as minute as 0.005N.

3) Calibration and Data Processing: Before experimentation, a stationary "dry run" establishes a baseline force profile, accounting for non-fluidic forces. This baseline, when subtracted from stirring force data, reveals fluidic resistance.

4) Tasks & Fluid Selection: We undertake three tasks: density estimation, apparent viscosity estimation, and mixture ratio inference. Fluids range from Newtonian to non-Newtonian and span various viscosities and densities:

Density Estimation: It is to validate the density estimation functionality alone. We employ high fructose syrup (F60), diluted syrup solution, water, and two starch solutions, which are prepared with ample starch to achieve high density.

Apparent Viscosity Estimation: Since fluids with large variations in density may not necessarily have large variations in viscosity, we select a different set of fluids to process the viscosity estimations: corn starch solutions at 24 Baumé (CS 24) and 25 Baumé (CS 25), castor oil (Oil), propylene glycol (PG), ketchup, and water.

Mixture Ratio Influence: It is to validate the sensitivity of our method to detect subtle viscosity change during the mixing of two fluids and command the robot to change its operation. The reason we adopt viscosity to indicate the mixture ratio instead of density is that the measurement of density should be conducted when the fluid mixture is diffused completely and will not be separated again during static placement. Such a requirement is not met in many fluid mixture cases. On the other hand, viscosity indicates the dynamic behavior of the fluid, which can be estimated during dynamic operation. We employ high fructose syrup (F60) and water.

B. Simulation Setup

In the MLS-MPM simulation, we set the time step to 10^{-3} seconds and the MPM grid width to 0.01m to balance computational efficiency and simulation precision. We use 6500 particles to balance fluid detail and computational speed within 24GB memory of Nvidia GeForce RTX 4090 GPU. Each simulation period runs for 8,000 frames, or 8 seconds, with a full gradient computation.

C. Task Setup

1) Density Estimation: We measure the gravitational force G of the stirring rod using the end-effector's force sensor in the robot arm during "dry run" while keeping it vertical; the immersion depth h is measured with a 0.1mm-precision ruler; and the volume V is found by using $V = Ah$, where A is the cross-sectional area. When equilibrium is reached, the net force R is read directly from the end-effector's force sensor.

2) Apparent Viscosity Estimation: As shown in Fig. 4, we validate our system's viscosity estimations, η, at a zero shear rate, $\eta(\dot{\gamma} = 0)$, because many zero-shear results of different fluids are reported so that we can compare [8], [9], [10]. If the apparent viscosity is consistent with the previous reports, we can validate the accuracy and usability of the estimated underlying physics parameters like η_0, η_∞, λ, τ_y, k, and n.

3) Mixture Ratio Inference: We conduct a solution mixing experiment, which blends F60 syrup and water to meet the viscosity of given samples. We give three different target sample solutions with ratios of 1:2, 1:3, and 1:10 respectively. These ratios are unknown to the DiffStir system. The robot is given an initial mixing ratio of 1:20 and tasked with stirring the mixture until homogeneous. It then employs robotic commands to adjust the mixture's viscosity to match that of the target sample. Throughout the process, we record the volume of each liquid added. To evaluate the robot's performance, we compare the final actual mixing ratio with the target sample's ratio.

VI. Results

A. Density Estimation

For density, we simply compare the DiffStir estimates ("ρ_{est}") with reference standards ("ρ_{ref}")(calculated by mass/volume), as detailed in Table I.

TABLE I: Benchmarking DiffStir's density estimation functionality against reference standards (kg/m^3).

Fluid	ρ_{est}	ρ_{ref}	Bias (%)
F60 Syrup	1333.7	1290	3.39
Diluted Syrup	1065.8	1139.3	-6.45
Water	915.1	1000	-8.49
Starch A	1641.6	1563.7	4.98
Starch B	1814.8	1845.5	-1.66

We calculate the bias by $\rho_{\text{est}}/\rho_{\text{ref}} - 1$. As shown in Table I, DiffStir system can provide accurate estimates (absolute bias $< 5\%$) for most liquid types. The slightly higher bias observed for diluted syrup and Water can be attributed to their overall lower densities, making them more sensitive to measurement inaccuracies.

B. Apparent Viscosity Estimation

a) Parameters for Apparent Viscosity: In Table II, we report all the parameters estimated for all the constitutive models: the Carreau Model, Cross Model, and Herschel-Bulkley Model. Parameters for these models are reported with units: τ_y in mPa, k in mPa·s^n, η_0 and η_∞ in mPa·s, n as dimensionless, and λ ensuring dimensional consistency. These parameters are required to calculate the apparent viscosity in the respective formulas.

b) Adaptive Selection Strategy: According to our adaptive selection strategy, we will automatically select the most proper model to characterize the fluids, as shown in Table III. It delineates average force deviations (N) for different fluids over 8s force curves. The model manifesting the smallest deviation, marked in bold, is selected as the most proper for providing accurate viscosity estimations. The system generally shows small force deviations, with the most precise measurement reaching as low as 0.006 N for PG.

TABLE II: Estimated parameters for the Carreau, Cross, and Herschel-Bulkley models for various fluids.

Fluid	ρ_{est}	Carreau $\eta_0, \eta_\infty, n, \lambda$	Cross $\eta_0, \eta_\infty, n, \lambda$	Herschel-Bulkley τ_y, k, n
CS 24	1146.82	224, 756, 1.1, 0.023	255, 812, 0.35, 0.1	341, 214, 1.3
CS 25	1175.38	903, 1347, 1.1, 0.002	957, 1342, 0.31, 0.04	335, 858, 1.3
Oil	866.88	1794, 341, 1.0, 16	1667, 1977, 0.07, 1231	23, 1868, 0.8
PG	936.27	16, 66, 1.1, 0.026	24, 76, 0.00, 93	11, 33, 0.8
Ketchup	1156.1	3989, 2567, 0.7, 5992	6135, 5568, 0.70, 1.97	2663, 3442, 0.5
Water	915.1	3, 5, 1.1, 0.001	4, 53, 0.01, 9.68	17, 0.01, 1.1

TABLE III: Average force deviations (N) for various fluids using different models.

Fluid	Carreau	Cross	Herschel-Bulkley
CS 24	**0.010**	0.016	0.255
CS 25	**0.017**	0.023	0.347
Oil	**0.048**	0.063	0.175
PG	0.008	**0.006**	0.151
Ketchup	0.475	0.513	**0.397**
Water	0.011	**0.009**	0.137

c) Comparing Apparent Viscosity Estimation with Baseline Method: We justify our choice of the constitutive models against the baseline model, the corotated model, which is adopted in FluidLab [39], on reactive force comparison and viscosity estimation. Table IV presents a comparison between viscosity estimated with our model, those with the Corotated model, and the reference values from established scientific principles and empirical findings [8], [9], [10]. Fig. 4 illustrates the congruence between simulation-calculated forces and real-world measurements, underscoring DiffStir's capability to accurately reflect real-world dynamics based on its adapted fluid constitutive parameters.

TABLE IV: Comparison of apparent viscosity values from our system: using our constitutive model and the baseline corotated model, against reference values.

Fluid	Our / Corotated	Reference	Bias (%)
CS24	133 / 87	155.55	-14.50 / -44.07
CS25	338 / 215	366.25	-7.71 / -41.30
Oil	615 / 743	650	-5.38 / 14.31
PG	18.7 / 11.7	16.2	15.43 / -27.78
Ketchup	2755.3 / 2145	3000	-8.16/ -28.50
Water	0.73 / 0.58	0.89	-17.98 / -34.83

d) Effectiveness of Viscosity Refinement Network: We validate the effectiveness of the viscosity refinement. Table V contrasts the initial and refined estimations with the reference

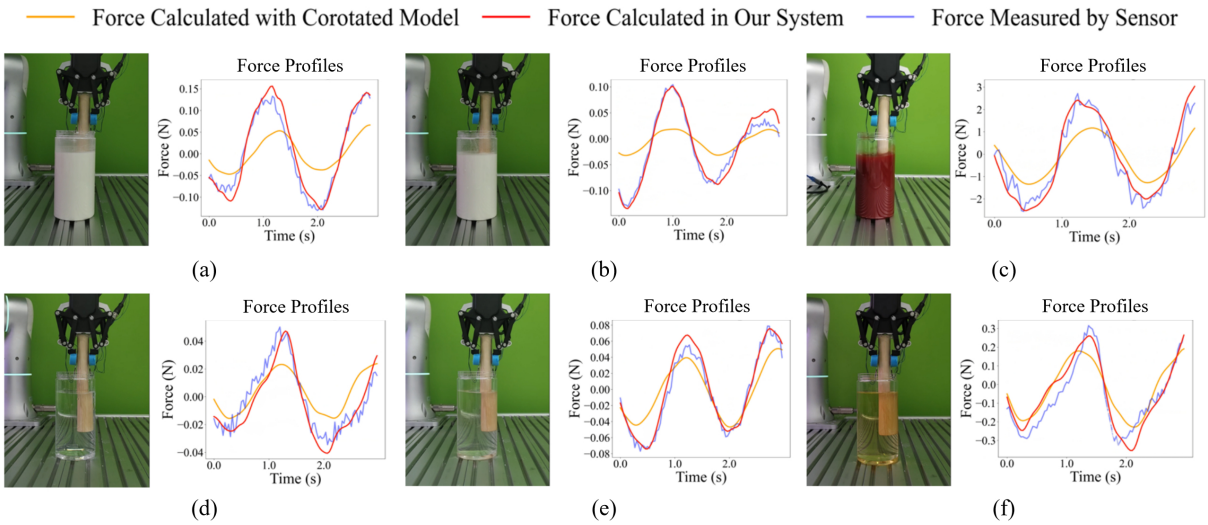

Fig. 4: Force Comparison During Stirring: From panels (a) to (f), the fluids represented are corn Starch solution (24 Baumé), corn starch solution (25 Baumé), ketchup, water, propylene glycol, and castor oil respectively. In each panel, the left side illustrates the real-world stirring scenario, and the right side showcases force comparisons in a fixed horizontal direction on the stirring rod. The red line depicts forces from simulations after fitting fluid parameters using our system, and the blue line, refined through adaptive Gaussian filtering, signifies experimental forces. The overlaps demonstrate DiffStir's precision.

values. We evaluate the refinement by comparing the "closeness" of the initial and refined viscosity to the reference values. We calculate the bias between the estimated and reference viscosity respectively, and then we calculate the difference in the absolute value of the bias. The results show that incorporating the MLP rectifier significantly improves the estimation accuracy by 21.7% over these six fluids.

TABLE V: Comparative viscosity estimations.

Fluid	Initial	Refined	Reference
CS24	123	133	155.55
CS25	302	338	366.25
Oil	793	615	650
PG	23	18.7	16.2
Ketchup	2663	2755.3	3000
Water	1.7	0.73	0.89

C. Mixture Ratio Inference

In the above experiments, we verify the efficacy of the DiffStir system in identifying density and viscosity. Based on these results, we further utilize the estimated parameters to guide the robot operations in the real-world task of mixture ratio inference. The robot aims to achieve precise preparation of liquid mixtures in specific mixture ratios by identifying viscosity dynamically through two operations: automatic stopping and providing instructions for adding liquids.

The robot stirs different liquids to form an initial mixture and utilizes **DiffStir** to online monitor the changing viscosity of the mixture. Once the viscosity of the mixture is detected to match the predetermined target sample, the robot immediately stops stirring.

When the viscosity of the mixture is stable but has not reached the target viscosity, the system recommends adding the higher-viscosity liquid or the lower-viscosity liquid based on the actual viscosity value. Following the system's recommendations, volunteers gradually introduce the corresponding liquid in 5ml increments, repeating this process until the mixture viscosity matches the target sample. This process can also be automatically conducted by a robot arm, however, this will complicate the operation and go beyond the scope of this work.

We experiment on three target samples which are the mixture of syrup and water, with ratios of 1:2, 1:3, and 1:10 respectively. In the end, our system estimates the mixture ratio to 1:1.7, 1:2.7, and 1:11 respectively.

To note, this process does not only estimate the mixture ratio but also replicates the fluid mixture with similar viscosity properties. This could be useful for many applications in the kitchen or chemistry laboratory, when the system should identify and replicate an unknown mixture.

VII. Conclusion

In this work, we introduced DiffStir, a novel differentiable fitting framework, adept at identifying the key physics parameters through a commonly practiced action, stirring. By using a robotic arm and synchronizing real-world actions with a diffMPM-based simulator, the density and viscosity of various fluids are estimated with considerable accuracy. We've showcased that not only can the method adapt to different fluid dynamics behaviors but also has the potential to bridge the gap between physical simulations and real-world observations. Through comprehensive evaluations, DiffStir has exhibited its efficiency and precision in parameter estimation, marking a notable advancement in the domain of fluid dynamics understanding in everyday scenarios.

REFERENCES

[1] M. J. Harris, "Fast fluid dynamics simulation on the gpu." *SIGGRAPH Courses*, vol. 220, no. 10.1145, pp. 1 198 555–1 198 790, 2005.

[2] J. D. Anderson and J. Wendt, *Computational fluid dynamics*. Springer, 1995, vol. 206.

[3] J. H. Park, G. P. Dalwankar, A. Bartsch, A. George, and A. B. Farimani, "Fluid property prediction leveraging ai and robotics," *arXiv preprint arXiv:2308.02715*, 2023.

[4] P. J. Carreau, "Rheological equations from molecular network theories," *Transactions of the Society of Rheology*, vol. 16, no. 1, pp. 99–127, 1972.

[5] M. M. Cross, "Rheology of non-newtonian fluids: a new flow equation for pseudoplastic systems," *Journal of colloid science*, vol. 20, no. 5, pp. 417–437, 1965.

[6] W. H. Herschel and R. Bulkley, "Konsistenzmessungen von gummi-benzollösungen," *Kolloid-Zeitschrift*, vol. 39, pp. 291–300, 1926.

[7] Y. Hu, L. Anderson, T.-M. Li, Q. Sun, N. Carr, J. Ragan-Kelley, and F. Durand, "Difftaichi: Differentiable programming for physical simulation," *arXiv preprint arXiv:1910.00935*, 2019.

[8] (2008) Liquids - dynamic viscosities. The Engineering ToolBox. Accessed: 13 September 2023. [Online]. Available: https://www.engineeringtoolbox.com/absolute-viscosity-liquids-d_1259.html

[9] (2003) Viscosity - absolute (dynamic) vs. kinematic. The Engineering ToolBox. Accessed: 13 September 2023. [Online]. Available: https://www.engineeringtoolbox.com/dynamic-absolute-kinematic-viscosity-d_412.html

[10] (2003) Liquids - kinematic viscosities. The Engineering ToolBox. Accessed: 13 September 2023. [Online]. Available: https://www.engineeringtoolbox.com/kinematic-viscosity-d_397.html

[11] F. Renaud, J.-L. Dion, G. Chevallier, I. Tawfiq, and R. Lemaire, "A new identification method of viscoelastic behavior: Application to the generalized maxwell model," *Mechanical Systems and Signal Processing*, vol. 25, no. 3, pp. 991–1010, 2011.

[12] J. G. Oldroyd, "On the formulation of rheological equations of state," *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, vol. 200, no. 1063, pp. 523–541, 1950.

[13] H.-K. Hong and C.-S. Liu, "Prandtl-reuss elastoplasticity: on-off switch and superposition formulae," *International Journal of Solids and Structures*, vol. 34, no. 33-34, pp. 4281–4304, 1997.

[14] C. S. Desai, "Unified dsc constitutive model for pavement materials with numerical implementation," *International Journal of Geomechanics*, vol. 7, no. 2, pp. 83–101, 2007.

[15] E. Merrill, E. Gilliland, G. Cokelet, H. Shin, A. Britten, and R. Wells, "Rheology of human blood, near and at zero flow: effects of temperature and hematocrit level," *Biophysical Journal*, vol. 3, no. 3, pp. 199–213, 1963.

[16] A. Sisko, "The flow of lubricating greases," *Industrial & Engineering Chemistry*, vol. 50, no. 12, pp. 1789–1792, 1958.

[17] E. C. Bingham, *An investigation of the laws of plastic flow*. US Government Printing Office, 1917, no. 278.

[18] F. Chen and H. Yan, "Constitutive model for solid-like, liquid-like, and gas-like phases of granular media and their numerical implementation," *Powder Technology*, vol. 390, pp. 369–386, 2021.

[19] N. E. Dowling, S. L. Kampe, and M. V. Kral, "Mechanical behavior of materials: engineering methods for deformation, fracture, and fatigue," *(No Title)*, 1999.

[20] M. Brizard, M. Megharfi, E. Mahe, and C. Verdier, "Design of a high precision falling-ball viscometer," *Review of scientific instruments*, vol. 76, no. 2, 2005.

[21] T. Labuza, A. Kaanane, and J. Chen, "Effect of temperature on the moisture sorption isotherms and water activity shift of two dehydrated foods," *Journal of Food science*, vol. 50, no. 2, pp. 385–392, 1985.

[22] S. Gunasekaran and M. M. Ak, "Dynamic oscillatory shear testing of foods—selected applications," *Trends in Food Science & Technology*, vol. 11, no. 3, pp. 115–127, 2000.

[23] R. L. Powell, "Rotational viscometry," *Rheological measurement*, pp. 260–298, 1998.

[24] A. K. Soares, D. I. Covas, and L. F. Reis, "Analysis of pvc pipe-wall viscoelasticity during water hammer," *Journal of Hydraulic Engineering*, vol. 134, no. 9, pp. 1389–1394, 2008.

[25] R. C. Carvalho, I. L. Herzog, H. R. B. Orlande, M. J. Colaço, I. M. Madeira, and N. Chakraborti, "Parameter estimation with the markov chain monte carlo method aided by evolutionary neural networks in a water hammer model," *Computational and Applied Mathematics*, vol. 42, no. 1, p. 35, 2023.

[26] K. Urbanowicz, A. Bergant, R. Grzejda, and M. Stosiak, "About inverse laplace transform of a dynamic viscosity function," *Materials*, vol. 15, no. 12, p. 4364, 2022.

[27] K. Fukami, K. Hasegawa, T. Nakamura, M. Morimoto, and K. Fukagata, "Model order reduction with neural networks: Application to laminar and turbulent flows," *SN Computer Science*, vol. 2, pp. 1–16, 2021.

[28] M. Morimoto, K. Fukami, K. Zhang, A. G. Nair, and K. Fukagata, "Convolutional neural networks for fluid flow analysis: toward effective metamodeling and low dimensionalization," *Theoretical and Computational Fluid Dynamics*, vol. 35, no. 5, pp. 633–658, 2021.

[29] S. B. Reddy, A. R. Magee, R. K. Jaiman, J. Liu, W. Xu, A. Choudhary, and A. Hussain, "Reduced order model for unsteady fluid flows via recurrent neural networks," in *International Conference on Offshore Mechanics and Arctic Engineering*, vol. 58776. American Society of Mechanical Engineers, 2019, p. V002T08A007.

[30] F. D. A. Belbute-Peres, T. Economon, and Z. Kolter, "Combining differentiable pde solvers and graph neural networks for fluid flow prediction," in *international conference on machine learning*. PMLR, 2020, pp. 2402–2411.

[31] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," 2016.

[32] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2012, pp. 5026–5033.

[33] M. Macklin, M. Müller, N. Chentanez, and T.-Y. Kim, "Unified particle physics for real-time applications," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, pp. 1–12, 2014.

[34] X. Lin, Y. Wang, J. Olkin, and D. Held, "Softgym: Benchmarking deep reinforcement learning for deformable object manipulation," in *Conference on Robot Learning*. PMLR, 2021, pp. 432–448.

[35] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, *et al.*, "Sapien: A simulated part-based interactive environment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 097–11 107.

[36] C. Gan, J. Schwartz, S. Alter, D. Mrowca, M. Schrimpf, J. Traer, J. De Freitas, J. Kubilius, A. Bhandwaldar, N. Haber, *et al.*, "Threedworld: A platform for interactive multi-modal physical simulation," *arXiv preprint arXiv:2007.04954*, 2020.

[37] C. Schenck and D. Fox, "Spnets: Differentiable fluid dynamics for deep neural networks," in *Conference on Robot Learning*. PMLR, 2018, pp. 317–335.

[38] P. Holl, V. Koltun, and N. Thuerey, "Learning to control pdes with differentiable physics," *arXiv preprint arXiv:2001.07457*, 2020.

[39] Z. Xian, B. Zhu, Z. Xu, H.-Y. Tung, A. Torralba, K. Fragkiadaki, and C. Gan, "Fluidlab: A differentiable environment for benchmarking complex fluid manipulation," *arXiv preprint arXiv:2303.02346*, 2023.

[40] M. A. Meyers and K. K. Chawla, *Mechanical behavior of materials*. Cambridge university press, 2008.

[41] M. Escudier, P. Oliveira, and F. Pinho, "Fully developed laminar flow of purely viscous non-newtonian liquids through annuli, including the effects of eccentricity and inner-cylinder rotation," *International journal of heat and fluid flow*, vol. 23, no. 1, pp. 52–73, 2002.

[42] Y. Hu, Y. Fang, Z. Ge, Z. Qu, Y. Zhu, A. Pradhana, and C. Jiang, "A moving least squares material point method with displacement discontinuity and two-way rigid body coupling," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.