

METHODOLOGY FOR ONLINE ESTIMATION OF RHEOLOGICAL PARAMETERS IN POLYMER MELTS USING DEEP LEARNING AND MICROFLUIDICS

Juan Sandubete-López^{1, 2}, José L. Risco-Martín¹, Alexander H. McMillan², and Eva Besada-Portas¹

¹Dept. of Computer Architecture and Automation, Universidad Complutense de Madrid, Madrid, SPAIN

²Microfluidics Innovation Center, Paris, FRANCE

ABSTRACT

Microfluidic devices are increasingly used in biological and chemical experiments due to their cost-effectiveness for rheological estimation in fluids. However, these devices often face challenges in terms of accuracy, size, and cost. This study presents a methodology, integrating deep learning, modeling and simulation to enhance the design of microfluidic systems, used to develop an innovative approach for viscosity measurement of polymer melts. We use synthetic data generated from the simulations to train a deep learning model, which then identifies rheological parameters of polymer melts from pressure drop and flow rate measurements in a microfluidic circuit, enabling online estimation of fluid properties. By improving the accuracy and flexibility of microfluidic rheological estimation, our methodology accelerates the design and testing of microfluidic devices, reducing reliance on physical prototypes, and offering significant contributions to the field.

1 INTRODUCTION AND RELATED WORK

Rheology is the science that studies how materials flow and deform. Among these rheological properties, viscosity is of special interest as it defines how much the fluid will flow when a force is applied. Viscosity estimation is crucial across various applications, from industrial processes like mold injection and 3D printing to medical diagnostics involving bodily fluids (Shaw 2011; Meek et al. 2014; Zhao et al. 2023). The behavior of non-Newtonian fluids, such as polymer melts or gelled propellants, under different pressure gradient presents significant challenges due to the complex dynamics of their viscosities (Kirchberger et al. 2016).

Traditional methods for measuring viscosity, such as rolling ball, rotational or capillary viscometers, while accurate, are often not suitable for inline measurement and real-time monitoring (Shaw 2011). These standardized methods such as those in (ASTM 2020) typically require discrete liquid samples and are impractical for processes involving continuous or semi-continuous flow of changing fluid compositions. The objective of this study is to develop a methodology that integrates deep learning with microfluidic technology to enable online, real-time estimation of rheological parameters in polymer melts. This approach aims to overcome the limitations of traditional methods by providing a cost-effective, accurate, and flexible solution for continuous monitoring and adjustment of fluid properties.

Recent developments in microfluidic devices are becoming a promising alternative. These devices require minimal fluid volumes and can be directly integrated into larger systems, facilitating real-time rheological estimation (Del Giudice 2022). Various microfluidic designs have already been explored, ranging from devices measuring the deformation of flowing polymers (Lee and Tripathi 2005; Del Giudice 2020) to systems using embedded sensors to measure pressure changes after a particular flow-path (Bie et al. 2019).

Moreover, in recent years deep learning has been successfully used to model many complex processes in fluid dynamics (Vinuesa and Brunton 2021) and in microfluidics (McIntyre et al. 2022), as well as in the intersection of rheology and microfluidics to perform parameters identification of non-Newtonian

fluids (Mustafa et al. 2023; Jarujareet et al. 2023). Nonetheless, estimating the viscosity's parameters from just pressure and flow signals could be convenient for their integration in microfluidic circuits, since these magnitudes are commonly measured. Liu et al. (2022) do so by combining least squares for the fittings. Their approach requires nevertheless of a particular stepped input signal, not being adequate for online estimation. The use of deep learning could help reducing the dependency of this restricted input sequences, allowing for estimating rheological properties, online, from pressure and flow rate signals.

One aspect that makes difficult the development of these systems is the fact that deep learning models requires of large training datasets. Simulating Newtonian and non-Newtonian fluids is often a computationally expensive and time-consuming task, which requires both expertise and resources because it is commonly done by means of Computational Fluid Dynamics (CFD, Galindo-Rosales and Rubio-Hernández 2010). For microfluidics, the simulations might be performed applying a finite-volume method (Keslerová and Kozel 2010; Ferziger et al. 2019) or the Lattice-Boltzmann method (Krüger et al. 2017). Higher levels of abstraction are possible, using the geometry of the systems and their symmetries to approximate the microfluidic system as two-dimensional (Boulais and Gervais 2023) or one-dimensional (Oh et al. 2012) problems. This allows to trade off the accuracy and complexity of the simulations. In this regard, Takken and Wille (2024) have recently proposed a hybrid simulation framework in which high levels of abstraction are used to speed-up the simulations through analytic expressions (where the flow path is abstracted to 1D), using CFD only in cases where this expressions are not available (e.g. junctions of fluidic paths).

Our methodology employs these principles of microfluidics to reduce the complexity of simulating generalized non-Newtonian fluids behaviors approximating the system as one-dimensional. By modeling the microfluidic system as a network of hydraulic resistances and capacitances, we can emulate the flow characteristics of various fluids under different conditions with reduced computational overhead. This model generates synthetic data used to train a deep neural network, an architecture containing Recurrent Neural Networks (RNN), which allows to discover time dependencies on the input data. This deep learning model is trained to predict the fluid's rheological parameters based on observed pressure changes and flow rates.

The main contributions of this work are: a methodology which can be applied to model, simulate and develop microfluidic systems flowing non-Newtonian fluids and integrated with deep learning, and an innovative approach to viscosity estimation for generalized Newtonian fluids, and more concretely, polymer melts, from commonly measured magnitudes such as pressure and flow signals. By providing a robust simulation framework, this work contributes to the broader field of fluid simulation and modeling, offering a practical tool for researchers and engineers working with complex fluid systems.

Examples of potential applications might range from the development of deep neural networks for the estimation of viscosity's parameters (shown in this paper), to chemical process AI-based control in microfluidics, passing through cell-culture anomaly detection within complex fluids.

This paper is organized as follows. Section 2 describes the theoretical microfluidic background underpinning our methodology. Section 3 details our methodology, including the exposition of our microfluidic circuit design, of the physical model used for simulation, and of the architecture of the neural network model used to estimate parameters of the viscosity model. Section 4 presents the conducted experiments and the obtained results. Finally, Section 5 draws the conclusions and presents some future lines of research.

2 BACKGROUND

This section introduces the required background on two areas supporting our work: generalized Newtonian fluid models, and microfluidics systems modeling.

The viscosity of a fluid relates its rate of displacement or shear rate, usually denoted $\dot{\gamma}$, for an imposed shear stress. This shear stress depends of how the material reacts to an applied force under some conditions. For Newtonian fluids, the ratio between these two magnitudes is independent of the applied shear stress, while in non-Newtonian fluids this does not always hold. Many different models have been proposed,

although the most commonly used is the power-law fluid model (Shaw 2011),

$$\eta = \eta_0(\dot{\gamma})^{n-1}, \quad (1)$$

where, η_0 stands for the zero shear rate viscosity, $\dot{\gamma}$ for the aforementioned shear rate, and n for its power index. For $n = 1$, shear rate and viscosity are perfectly independent, which is the behavior of an ideal Newtonian fluid. This model is used by Srivastava and Burns (2006) to semi-analytically approximate the solution to the flow of a non-Newtonian fluid through a hydraulic resistance of rectangular cross-section.

Analytic and semi-analytic expressions have been derived for the streamlined capillary geometries commonly used in microfluidics. These expressions are obtained by considering one-dimensional or two-dimensional approximations to the problem (Mortensen et al. 2004). They are used to speed up the simulations of these systems, accelerating the design process, for example, by simulating the system as analogous electrical problems, i.e., 1D approximation of the microfluidic circuits (Oh et al. 2012).

In these analogous electrical problems, an equivalent Ohm's law (Tipler and Mosca 2004) is defined for Newtonian fluids under laminar flow. The equations of Hagen-Poiseuille (Anna 2013) allows for solving microfluidic circuits in an analogous manner to electrical circuits. In particular, the flow rate, Q_v , through a hydraulic resistance can be estimated as

$$Q_v = \frac{\Delta P}{R_{hyd}}, \quad (2)$$

where R_{hyd} is the *hydraulic resistance* value, and ΔP is the pressure drop over it. Here, the subscript v of Q_v is used to denote that the flow is governed by viscous forces. If the circuit has only one branch with multiple hydraulic resistances interconnected in series, then the total resistance of the circuit can be estimated as $R_{hyd} = R_1 + R_2 + \dots$. The total flow rate passing through the branch can be calculated using equation (2) over the total resistance.

Lastly, and continuing with the analogy, a hydraulic *RC* circuit can be defined by adding a hydraulic capacitance, C , to the circuit. Its transient response will be then governed by a time constant $\tau = R_{hyd}C$. The relation between volume, V , and capacitance C is simply $C = K_{fluid}V$, where K_{fluid} depends on the fluid, being $K_{fluid} = 1.0$ for air.

3 METHODOLOGY

Figure 1 shows the overall work-flow of the designed and implemented methodology, which is explained in this section. The aim of the methodology is to generate data for training a RNN model to estimate the rheological properties of a fluid.

First, a set of values of the viscosity model's parameters is sampled from random uniform distributions within defined ranges. For this set, a random sequence of pressure signals is generated to be used as input to the microfluidic circuit simulator together with the sampled parameters, outputting signals of pressure drop along the resistances, as well as input and output flows to the capacitance. Simulations that do not contain significant information for the corresponding parameters are rejected, requiring the generation of a new random sequence of signals for the same set of parameters. This way, for each simulation, four output signals and one set of parameters of the viscosity model are collected.

Second, the generated dataset is used to perform a supervised training of the RNN model with the purpose of identifying the viscosity's parameters corresponding to each fluid from the signals produced by the simulator. This way, the model is trained for a number of epochs using mini-batches where each mini-batch is composed of sixteen simulations. Each simulation is made up of the four signals (two pressure drop signals, and two flow-rate signals) associated to the set of viscosity model's parameters that have been used to produce them. The RNN model receives these four signals, sample by sample, four values in parallel (one by input signal), outputting one prediction once the whole simulation has passed through.

Last, the simulator is again used to verify the operation of the trained artificial neural network. The input signals from the test dataset are used to test the prediction quality of the RNN model. For each

simulation, the RNN will predict the corresponding set of parameters. These estimated parameters are then simulated and the output four signals of pressure drop and flow rate are compared to the ones produced with the correct set of parameters of viscosity. The difference is evaluated to verify that, for each simulation, if the predicted parameters are different, the output of the simulation is different as well. This indicator is used as guide for the discovery of the right setup to estimate the parameters of the viscosity model.

In the following subsections, the proposed microfluidic circuit design is described along with the corresponding physical model. The simulation process is then introduced, followed by the description of the proposed artificial neural network architecture. Lastly, the used verification process is detailed.

3.1 Design and Modeling of the Microfluidic Circuit

Microfluidic systems development is pursued for many reasons (e.g. increased surface-volume ratio or low-waste applications, Tabeling 2005), however, the most interesting feature of microfluidics related to this work is the reduction of complexity of the system under laminar flow. According to Hagen-Poiseuille's equations (Anna 2013), the laminar flow of a Newtonian liquid passing through a capillary can also be analytically characterized for simple cross-sectional geometries. The expression depends of the geometry and dimensions of the capillary, the dynamic viscosity of the fluid, η , and the applied pressure drop along the capillary (Tabeling 2005). In particular, for a microfluidic resistance of rectangular cross-section of width, w , height, h , and length, L (when $h \ll w$, and $h, w \ll L$) becomes

$$Q_{v,i} = \frac{wh^3 \Delta P_i}{12L\eta} \left(1 - 0.63 \frac{h}{w} \right) = \frac{\Delta P_i}{R_{g,i}\eta}, \quad (3)$$

where $R_{g,i} = 12L/(wh^3(1 - 0.63 \frac{h}{w}))$ stands for the contribution of all the purely geometrical terms to the i -th hydraulic resistance (i.e. $R_i = R_{g,i}\eta$). The subscripts, i , in terms h, w , and L have been omitted for simplicity in equation (3).

Figure 2 shows the equivalence between the designed microfluidic circuit and the corresponding ideal electrical circuit, which can be modeled by one resistance in series connected to another resistance with

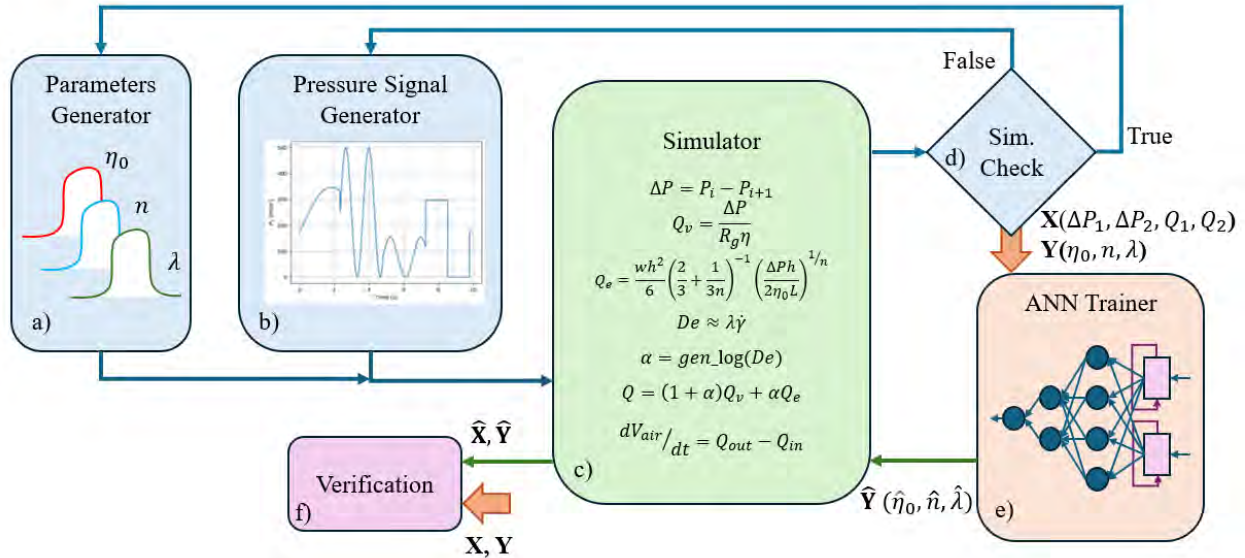


Figure 1: Methodology diagram. The elements in blue, a), b) and d), operate together with the simulator to generate a dataset, X, Y , with which the deep learning model is trained at block e). The simulator uses the predictions of the deep learning model to run simulations, producing an estimation dataset, $\hat{X}, \hat{Y}$, which is compared at f) with the original dataset to verify the methodology.

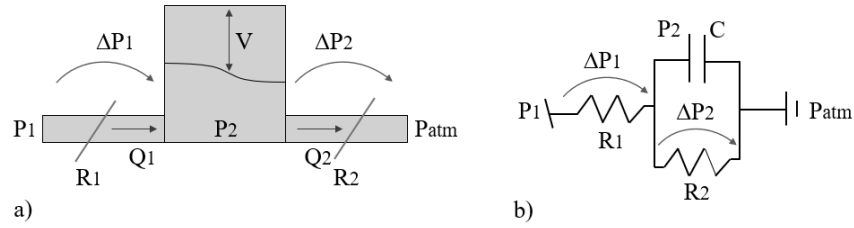


Figure 2: Simulated microfluidic circuit. a) shows the fluidic system diagram composed by one input hydraulic resistance, R_1 , one capacitance characterized by its volume, V , and one output hydraulic resistance, R_2 . b) shows the analogous electrical circuit.

a capacitor in parallel, with $R_1 = R_2$. For Newtonian fluids, the rate at which the capacitance changes its internal pressure is given by a time constant, τ , so resolving by applying the Thevenin's theorem: $\tau = R_{th}C = (R_1 + R_2)C$. In this model, the hydraulic resistance corresponding to the tubing which connects the elements (i.e., from the reservoir to the R_1 , and from R_2 to the waste) and the connectors (hydraulic interfaces between elements) have been neglected since the characteristic dimensions of the cross-section of the capillary are usually around two orders of magnitude above, and the hydraulic resistance decreases with the power of three over these dimension.

In this hydraulic RC circuit in which a Newtonian fluid flows, if the values of R_g and C , are known, the viscosity can straightaway calculated as $\eta = \tau/(R_g C)$ after measuring the circuit time constant τ . The introduced capacitance is thus important because it allows to extract information of the fluid from the transient response.

As it can be observed in Figure 2, an extra resistance (called R_1 hereafter) is included before the capacitance in the hydraulic circuit, or before the RC in the electrical analogy. This input resistance has the purpose of limiting the maximum flow rate to the capacitance. Following with the electrical analogy, it will short-circuit for fast changes in the input signal, leading to high flow-rates that could move the regime out of laminar flow. The value of this new resistance, although arbitrary, is set to be equal to the resistance of the RC part (called R_2 hereafter) for simplicity. Estimates of their values can be calculated from commercially available microfluidic resistance chips with $h = 20\mu m$, $w = 800\mu m$, and $L = 25mm$, which leads to $R_{g1} = R_{g2} = 9.53e16 m^{-3}$.

In polymer melts and other polymeric dissolutions, equation (3) holds while the polymer chains recovery time is smaller than the characteristic time of the flow, because the elastic forces do not have a significant influence over the dynamics of the fluid. The dimensionless Deborah number can be used to estimate when these elastic forces must be taken into account (Anna 2013) and are computed according to

$$De_i \approx \lambda \dot{\gamma}_i = \frac{6\lambda Q_{v,i}}{wh^2}, \quad (4)$$

where λ is the relaxation time of the fluid, and $\dot{\gamma}_i$ the shear rate. Being the rightmost part of the equation the corresponding solution at the wall of the i -th capillary of rectangular cross-section.

For $De > 1/2$, the chains are elongated with the flow and the elasticity of the fluid plays a significant role on the overall observed behavior which cannot be neglected. In this case, semi-analytic equation (5) obtained by Srivastava and Burns (2006) can be used to approximate the flow rate through the given i -th rectangular hydraulic resistance,

$$Q_{e,i} \approx \frac{wh^2}{6} \left(\frac{2}{3} + \frac{1}{3n} \right)^{-1} \left(\frac{h\Delta P_i}{2\eta_0 L} \right)^{1/n}, \quad (5)$$

where η_0 stands for the zero-shear-rate viscosity (which corresponds to η in Newtonian regime) and n for the power-law index. The subscript e in $Q_{e,i}$ is used to denote the flow under influence of elastic forces.

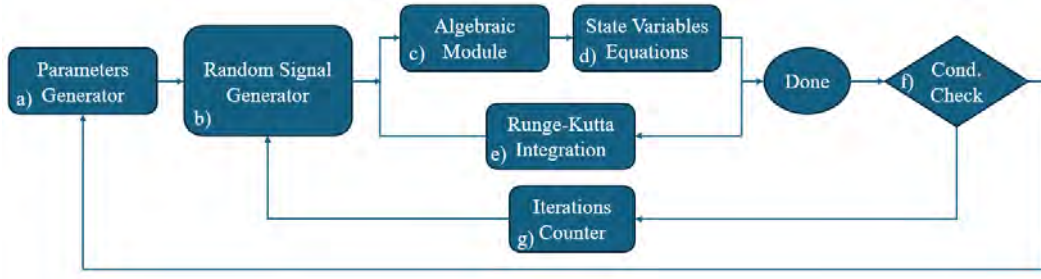


Figure 3: Diagram showing the simulation flow. a) generates set of parameters for which block b) creates a random input pressure signal. In the simulation loop, c) calculates all the algebraic expressions and d) resolves the variation of volume in the capacitance, integrated by e). If the conditions of f) don't match, a new signal is generated and the loop is run again. Otherwise, a new set of parameters is created and the process starts again.

In order to produce a set of equations that allow smooth transitions between the Newtonian and non-Newtonian behavior, the set of expressions is complemented with a generalized logistic function that receives as input the value of De and produces as output the value of α , which is used to weight equation (3) and equation (5) during the transition according to:

$$Q_i = (1 - \alpha)Q_{v,i} + \alpha Q_{e,i} \quad (6)$$

The flow rate, Q_i , through each resistance, R_1 and R_2 , might be different during the transient response, while the capacitance, C , charges or discharges fluid. Lastly, the only state variable considered in our microfluidic simulator is the charging of the capacitance, C . Modeled as a capacitance of the compressed air, the evolution of the volume can be calculated as the balance of fluid flow experienced by the capacitor:

$$\frac{dV}{dt} = Q_2 - Q_1, \quad (7)$$

where Q_1 is the flow going into the capacitance, C , and Q_2 is the flow going out of it. Each of these flows will be calculated according to equation (6).

The rate of change of volume from equation (7), integrated, can be related to the pressure by means of Boyle-Mariotte formula (Tipler and Mosca 2004). For the first hydraulic resistance, the expression will become:

$$\Delta P_1 = P_1 - P_2 = P_1 - P_o \frac{V_{max}}{V}, \quad (8)$$

where V_{max} is the maximum volume of air stored in the capacitance, P_o is its initial internal pressure, and V is the volume of air stored inside, calculated by integrating dV/dt , and P_1, P_2 are the pressures at the input and the output of the first hydraulic resistance, R_1 . Equivalently, the pressure drop along R_2 is calculated as $\Delta P_2 = P_2 - P_3$.

3.2 Simulation

To generate the training and testing datasets for the RNN, simulations are run following the scheme presented in Figure 3. The first block generates a set of random values for the parameters η_0 , n and λ , from their corresponding uniform distributions within the ranges defined in Table 1. These ranges has been selected according to the selected commercially-available hydraulic resistances to be able to observe Newtonian and non-Newtonian behaviors within experiments of 12.5 seconds. The timescale of the tool can be tuned by changing the time duration of the experiments, the values of the hydraulic resistances or the capacitance of the microfluidic circuit in order to study fluids of different degree of viscosity with this setup.

As Table 1 shows, shear-thinning (which happens when $n < 1.0$) and shear-thickening (when $n > 1.0$) are allowed behaviors for the fluid. Nevertheless the range of n remains close to 1.0, because the tubing and connections have been neglected in the current equivalent circuit. This is, high shear-thinning fluids, like xanthan solutions (Mrokowska and Krztoń-Maziopa 2019) or gelled propellants with $n < 0.5$ (Caldas Pinto et al. 2019), will be less accurately simulated under non-Newtonian regime the smaller their behavior exponent is, because higher viscosities will be observed at the tubing, increasing their resistance to flow. And, oppositely, shear-thickening fluids simulations will be more accurate the higher n is.

After the random values of the parameters are generated, a second block generates random sequences of sines, steps and ramps. The reason behind the randomness of the input signals is to increase the variance in the dataset to avoid having to impose patterns in the input signals for estimating the parameters, minimizing the interference with the user's work in a real setup.

Next, in order to generate datasets with valid information, a type of rejection sampling scheme is applied. The process starts using the sampled values of the parameters and of the reference inputs to run the simulations, and continues checking if the simulations output signal fulfill the conditions required by the *conditions check* block.

The simulation is performed such that, at each instant, k , P_1 is imposed from the input pressure signal, then equation (8) is used to calculate the pressure drop over R_1 . This is used in equation (3) to calculate the Newtonian flow-rate, $Q_{v,1}$, and in equation (5), to calculate the non-Newtonian flow rate, $Q_{e,1}$, through R_1 . $Q_{v,1}$, is then used as input to calculate De_1 . From this number, the parameter α_1 is obtained and used to weight $Q_{v,1}$ and $Q_{e,1}$ for the final Q_1 calculation. The process is repeated for R_2 , obtaining the corresponding Q_2 . Finally, the variation rate of the volume at the capacitance is calculated as the difference between Q_2 and Q_1 .

The condition block checks if the output signal contains information of Newtonian and non-Newtonian behavior of the fluid. In particular, if $\bar{\alpha} = \{\alpha_0, \alpha_1, \dots, \alpha_N\}$ is the sequence of values of α that establishes the regime of the flow at equation (6) at different instant $k \in \{0, N\}$, the condition block checks if

$$|\bar{\alpha}_v| = |\{\alpha_k \in \bar{\alpha} \mid \alpha_k < \alpha_{v,th}\}| \geq p_v N, \quad (9)$$

$$|\bar{\alpha}_e| = |\{\alpha_k \in \bar{\alpha} \mid \alpha_k \geq \alpha_{e,th}\}| \geq p_e N \quad (10)$$

where $|\cdot|$ denotes the size of a subset, $\alpha_{v,th}$ and $\alpha_{e,th}$ are thresholds to ensure predominant Newtonian and predominant non-Newtonian regimes for some corresponding percentages, p_v and p_e , of the total samples of the experiment, N . The conditions are nevertheless checked for a maximum number of iterations to ensure that, if the conditions are too difficult to match for a given parameters set, the simulator will not remain blocked in an endless loop.

The input pressure signals are non-overlapping, randomly composed sequence of sine, ramp and step functions of different values generated according to a uniform distribution within the configured values. Each randomly generated pressure signal has a duration of 12.5s with a resolution of 0.05s, or 250 points at 20Hz. This pressure signal, which corresponds with P_1 in Figure 2, remains constant while the dynamics of the rest of the system are calculated. Runge-Kutta (Shampine 1986) is used to integrate the state variable equation (7). The used implementation assumes accuracy of order fourth to control the error, while steps are taken using the order five formulation. The generated dataset is composed of 5500 experiments of 12.5s (250 samples) each one. Figure 5 shows some of the main output signals from an example simulation.

Table 1: Ranges of the uniform distribution for each studied parameter of the fluid.

Parameter	Minimum	Maximum
η_0	$1.0e-3$	$1.6e-3$
n	0.90	1.1
λ	$1.0e-4$	$5.0e-4$

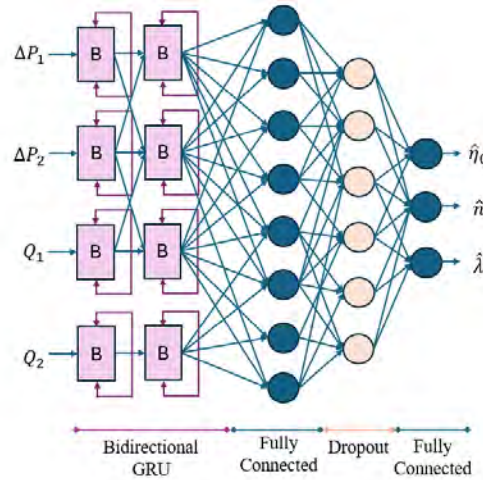


Figure 4: Designed artificial neural network. It receives four inputs signals: ΔP_1 , ΔP_2 , Q_1 , Q_2 , and outputs three viscosity parameters: η_0 , n and λ . The bidirectional GRU layers provides the model with memory capability to deal with temporal signals. The fully connected layers enhance its capacity to model complex functions. The dropout layer helps to regularize the model.

3.3 Artificial Neural Network

The proposed architecture is composed of five layers, has four inputs (namely ΔP_1 , ΔP_2 , Q_1 , Q_2) and three outputs (consisting of the estimated values of parameters η_0 , n and λ).

The two first layers of the model constitute the recurrent part, where different implementations were tested. Namely unidirectional Long Short-Term Memory (LSTM, Hochreiter and Schmidhuber 1997), unidirectional Gated Recurrent Unit (GRU, Cho et al. 2014) and bidirectional GRU (BGRU, Wang et al. 2017). In unidirectional RNN, information can only flow from past instants to the last instant, whereas in bidirectional RNN the signals are analyzed in both (time) directions (Schuster and Paliwal 1997). These BGRU layers were both implemented with twenty hidden units, Rectified Linear Units (RELU) as activation layers, leaky RELU for their recurrent activation, and a recurrent dropout of probability 10%.

After the two RNN layers, two fully connected (i.e. dense) layers are sequentially connected, separated by a dropout layer. The first input layer is implemented with twenty hidden units, followed by a leaky RELU activation layer. The dropout layer is set with probability 10%. The output fully connected layer is composed by three units, one unit for each estimated parameter, with linear activation. The final model version has a total of 11.5k tunable parameters. Its architecture can be observed in Figure 4.

The inputs to the system are two signals of pressure drop and two signal of flow-rate. Each signal is individually normalized in range $(-1.0, 1.0)$, with a length of 250 samples per signal, 1000 samples in total. This configuration allows for its operation in real-time by implementing an input sliding window with the previous setup. The model produces only one estimate of each parameter, i.e., η , n , and λ , after inserting the 250 samples of each signal. The outputs are also in range $(-1.0, 1.0)$ and must be denormalized. The model was trained from zero. Adam optimizer is used with learning rate $2.5e-5$, no weight-decay, for 25 epochs, and Mean Squared Error (MSE) is used as loss function.

3.4 Verification

In order to verify the operation of the simulation, training, and estimation methodology, the following three steps procedure is performed: first, a K-fold cross validation method is applied to ensure the independence between the training and testing dataset. Thirty three percent of the generated dataset is used for the testing, and the remaining sixty seven percent is used for training. Thus, it is chosen $K = 3$, performing a complete

rotation over the dataset after three iterations of the method. Since the generation of the parameters is uniformly performed within the defined range and the dataset is shuffled before the execution of the K-fold cross validation, the performance is expected to be similar for the three iterations. To evaluate it, an accuracy normal distribution curve is fitted of each model.

Afterwards, for each model, simulations are performed using the estimated parameters for each sample of the dataset. The Mean Square Error (MSE) between the curves simulated with the reference parameters and the curves simulated with the estimated parameters is calculated.

Finally, the Pearson correlation coefficients are calculated between the parameters estimation error and the simulated curves error. Correlation coefficients close to the unit imply a high correlation between these two error evaluation. This potentially indicates that the mapping between the parameters and the simulated curves is injective, and thus, the experiments are properly set for the training of the RNN. Otherwise, a correlation coefficient for a particular estimated parameter closer to one half mean that the performed experiments might not be suitable for the RNN to discover the original values.

4 EXPERIMENTS AND RESULTS

In this section we explain how we have used the proposed methodology to explore the optimization of the simulation setup towards the inference of complex parameters, such as the rheological properties of a fluid, from pressure and flow-rate signals that do not have a straightforward linkage.

In order to define the ranges over which the current setup could be useful, the proposed framework can be used to perform a heuristic analysis. For the given resistances, capacitance, and maximum input pressure (arbitrarily set to 600mbar or 60e3 Pa), the objective is to find out what are the ranges for the parameters of the viscosity model for which both Newtonian and non-Newtonian regimes can be observed. The most influential parameter in this regard is the relaxation time, λ , because it regulates the threshold flow-rate at which the fluid will behave in one or the other way. A higher zero-shear viscosity, η_0 , will decrease the observed flow-rate, Q_i , for a given applied pressure, P_i , making it harder to reach the non-Newtonian regime. Figure 5 shows two example simulations with same random input signal. Non-Newtonian regime is only achieved in one of them. The resulting parameters' ranges from this analysis are found in Table 1.

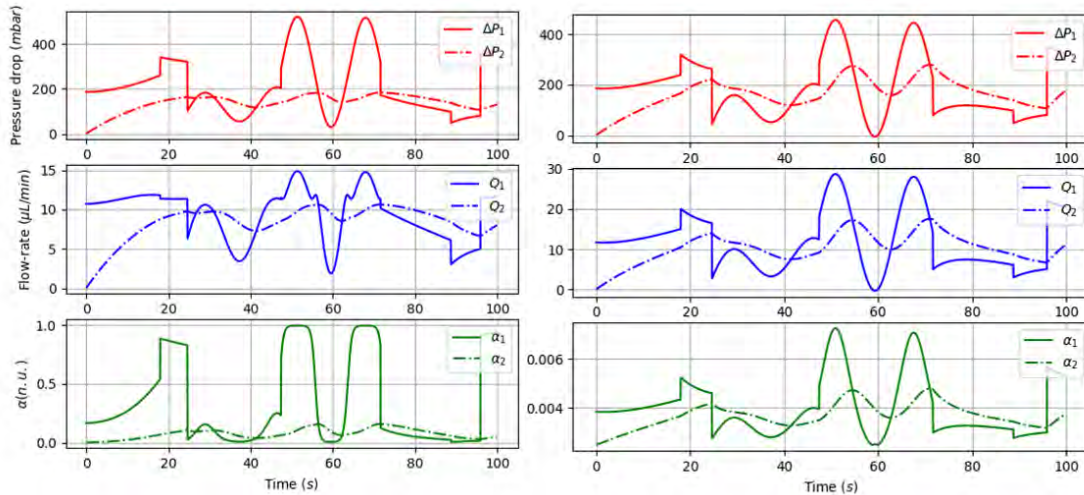


Figure 5: Two simulations performed over 100s with the same random pressure input signal. The parameters correspond to shear-thickening fluid ($n = 1.1$) with, first, $\lambda = 1.0e - 4$ (left), and second, $\lambda = 1.0e - 5$ (right). In the rightmost simulation it can be observed that the fluid doesn't go into non-Newtonian regime ($\alpha \ll 1.0$), thus the viscosity remains low, producing higher-values of flow-rate.

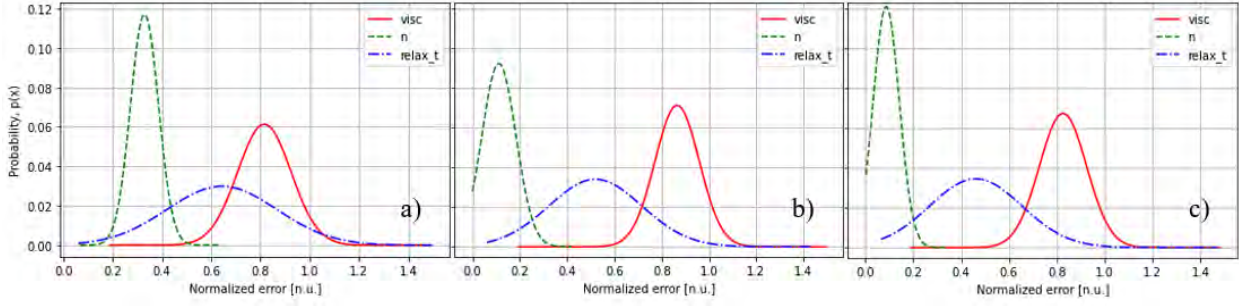


Figure 6: Error distributions for the three estimated parameters (η_0 , n , and λ) training the deep learning model with three different random signal generation setups: a) only step sequences, b) steps and sinusoids sequences, and c) slower steps and sinusoids than in b).

As a first step, an exploration of different RNN-based model designs was performed aiming to reduce the possibilities of having a performance bottleneck at the model architecture. Three RNN implementations were tested: unidirectional LSTM, unidirectional GRU and BGRU. While the benchmark was equivalent for all of them, the unidirectional LSTM often turned unstable, showing more sensitivity to the training hyper-parameters. Unidirectional GRU and BGRU implementations were both stable, being BGRU chosen since it was observed to be faster converging towards the same results for both same and different hyper-parameters.

After that, an exploration of the simulation setup is performed. Among the different parameters that can be configured are the properties of the randomized generation of pressure input signals: ranges for maximum and minimum values, ranges for the random frequency, the duration of each sequence inside the 10s experiment, etc. Figure 6 shows the fitted error distributions for the estimation of each parameter of the viscosity model during the exploration process. Each distribution set of three distributions corresponds to a different deep learning model. The error distribution is normalized to ease their comparison. Each of these models has been trained with a dataset generated with a different configuration. In the first dataset, corresponding to Figure 6.a, the input pressure signal, P_1 , is a sequence made out of 8-10 sinusoidal signals. In the second set, Figure 6.b, P_1 is a 8-10 segments sequence of steps, ramps, and sinusoidal signals. In the third one, Figure 6.c, 3-4 segments of steps, ramps and signals are combined. The reduction of error can be observed, being the behavior exponent, n , the easiest parameter for the models to learn to estimate.

The Pearson correlation coefficients are lastly calculated to be used as an indicator of sensitivity between the used input signals to the deep learning model and the parameters that it is asked to estimate. This is, if the chosen input signals to the model, X , are good candidates to estimate the parameters, Y , simulating the inaccurate values of the parameters, $\hat{Y}$, should produce wrong values of these input signals, $\hat{X}$. The error can be defined as $E(X) = \sqrt{X^2 - \hat{X}^2}$. In Table 2 the correlation coefficients are showed for the model whose results are displayed at Figure 6.c. The correlation shows that it is, indeed, the behavior index, n , the parameter with a higher correlation over the simulated signals, potentially making it the easiest parameter of viscosity model to be estimated.

Table 2: Pearson correlation coefficients between parameter estimation error vectors, ($E(\eta_0)$, $E(n)$, and $E(\lambda)$) and the simulation error vectors ($E(\Delta P_1)$, $E(\Delta P_2)$, $E(Q_1)$, and $E(Q_2)$). The error vectors correspond to the model trained with input signals composed of 3-4 segments of steps, ramps, and signals. Associated error distributions are displayed at Figure 6.c.

	$E(\Delta P_1)$	$E(\Delta P_2)$	$E(Q_1)$	$E(Q_2)$
$E(\eta_0)$	-0.183	0.041	0.101	0.117
$E(n)$	0.044	-0.533	-0.066	-0.164
$E(\lambda)$	0.107	-0.054	0.076	0.095

5 CONCLUSIONS AND FUTURE WORK

In this study, we developed a one-dimensional model for a hydraulic RC circuit to simulate generalized Newtonian fluids. This model was crucial for generating synthetic data to train deep learning models. We used bidirectional GRUs in our methodology, which effectively predicted rheological parameters from the dynamic responses of microfluidic systems. Our findings show that the behavior index n of the fluid correlates strongly with the output signals, making it the most reliably estimated parameter.

For future work, we plan to improve the simulation setup by automating the exploration process to optimize input signal configurations and model parameters systematically. We also aim to build a physical prototype to validate our simulations and refine model assumptions. This step is essential for practical applications in industrial and medical diagnostics. Additionally, we will explore incorporating more complex fluid dynamic models and advanced neural network architectures to handle a wider range of fluids and flow conditions, expanding the applicability of our methodology to various industrial uses.

ACKNOWLEDGMENTS

This work has been supported by the Research Projects IA-GESBLOOM-CM (Y2020/TCS-6420) funded by the Synergic program of the Comunidad Autónoma de Madrid (CAM), SMART-BLOOMS (TED2021-130123B-I00) by MCIN/AEI/10.13039/501100011033 and the European Union by NextGenerationEU/PRTR, and the MSCA-ITN-H2020 Project LasIonDef (GA n.956387).

REFERENCES

- Anna, S. L. 2013. *Non-Newtonian Fluids in Microfluidics*, 1–12. Springer US https://doi.org/10.1007/978-3-642-27758-0_1129-4.
- ASTM 2020. *Annual Book of ASTM Standards*, Volume 06.01, Chapter Standard Test Methods for Rheological Properties of Non-Newtonian Materials by Rotational Viscometer. ASTM International <https://doi.org/10.1520/D2196-20>.
- Bie, Y., X. Guo, P. Song, J. Yang and Z. Li. 2019. “A novel design of flow structure model for online viscosity measurement”. *Insight - Non-Destructive Testing and Condition Monitoring* 61(1):9–14 <https://doi.org/10.1784/insi.2019.61.1.9>.
- Boulais, E. and T. Gervais. 2023. “The 2D microfluidics cookbook – modeling convection and diffusion in plane flow devices”. *Lab on a Chip* 23(8):1967–1980 <https://doi.org/10.1039/D2LC01033J>.
- Caldas Pinto, P., H. Ciezki, K. Naumann, J. Ramsel, P. Kröger, H. Niedermaier *et al.* 2019. “Green gelled propellant highly throttleable rocket motor and gas generator technology: status and application” <https://doi.org/10.1051/eucass/201911091>.
- Cho, K., B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk *et al.* 2014. “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation” <https://doi.org/10.3115/v1/D14-1179>.
- Del Giudice, F. 2020. “Simultaneous measurement of rheological properties in a microfluidic rheometer”. *Physics of Fluids* 32(5) <https://doi.org/10.1063/5.0006060>.
- Del Giudice, F. 2022. “A Review of Microfluidic Devices for Rheological Characterisation”. *Micromachines (Basel)* 13(2):167–186 <https://doi.org/10.3390/mi13020167>.
- Ferziger, J. H., M. Perić, and R. L. Street. 2019. *Computational methods for fluid dynamics*. springer <https://doi.org/10.1007/978-3-642-56026-2>.
- Galindo-Rosales, F. J. and F. J. Rubio-Hernández. 2010. *Numerical Simulations - Examples and Applications in Computational Fluid Dynamics*, Chapter Numerical Simulation in Steady flow of Non-Newtonian Fluids in Pipes with Circular Cross-Section, 3–22. InTech <https://doi.org/10.5772/545>.
- Hochreiter, S. and J. Schmidhuber. 1997. “Long Short-Term Memory”. *Neural Computation* 9(8):1735–1780 <https://doi.org/https://doi.org/10.1162%2Fneco.1997.9.8.1735>.
- Jarujareet, U., K. Wiratchawa, P. Panpisut, and T. Intharrah. 2023. “DeepDDM: A Compact Deep-Learning Assisted Platform for Micro-Rheological Assessment of Micro-Volume Fluids”. *IEEE Access* 11:66467–66477 <https://doi.org/10.1109/ACCESS.2023.3290496>.
- Keslerová, R. and K. Kozel. 2010. “Numerical modelling of incompressible flows for Newtonian and non-Newtonian fluids”. *Mathematics and Computers in Simulation* 80(8):1783–1794 <https://doi.org/https://doi.org/10.1016/j.matcom.2009.12.005>.
- Kirchberger, C., P. Kröger, M. Negri, and H. Ciezki. 2016. “Overview on the Gelled Propellants Activities of DLR Lampoldshausen”. In *52nd AIAA/SAE/ASME Joint Propulsion Conference, 2016* <https://doi.org/10.2514/6.2016-4665>.
- Krüger, T., H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva and E. M. Viggien. 2017. “The lattice Boltzmann method”. *Springer International Publishing* 10(978-3):4–15 <https://doi.org/10.1007/978-3-319-44649-3>.
- Lee, J. and A. Tripathi. 2005. “Intrinsic Viscosity of Polymers and Biopolymers Measured by Microchip”. *Analytical Chemistry* 77(22):7137–7147 <https://doi.org/10.1021/ac050932r>.

- Liu, N., H. Gao, Y. Xu, X. Chai, Y. Hu and L. Duan. 2022. “Design and use of an online drilling fluid pipe viscometer”. *Flow Measurement and Instrumentation* 87:102224 <https://doi.org/https://doi.org/10.1016/j.flowmeasinst.2022.102224>.
- McIntyre, D., A. Lashkaripour, P. Fordyce, and D. Densmore. 2022. “Machine learning for microfluidic design and control”. *Lab on a Chip* 22(16):2925–2937 <https://doi.org/10.1039/d2lc00254j>.
- Meek, G. A., R. Williams, D. Thornton, P. Knapp and S. Cosser. 2014. “F2E - Ultra High Pressure Distributed Pump Common Rail System”. *SAE Mobilus* <https://doi.org/https://doi.org/10.4271/2014-01-1440>.
- Mortensen, N. A., F. Okkels, and H. Bruus. 2004. “Reexamination of Hagen-Poiseuille flow: shape dependence of the hydraulic resistance in microchannels.”. *Physical review. E, Statistical, nonlinear, and soft matter physics* 71 5 Pt 2:057301 <https://doi.org/https://doi.org/10.1103/PhysRevE.71.057301>.
- Mrokowska, M. M. and A. Krztoń-Maziopa. 2019. “Viscoelastic and shear-thinning effects of aqueous exopolymer solution on disk and sphere settling”. *Scientific Reports* 9(1) <https://doi.org/10.1038/s41598-019-44233-z>.
- Mustafa, A., D. Haider, A. Barua, M. Tanyeri, A. Erten and O. Yalcin. 2023. “Machine learning based microfluidic sensing device for viscosity measurements”. *Sensors & Diagnostics* 2(6):1509–1520 <https://doi.org/10.1039/d3sd00099k>.
- Oh, K. W., K. Lee, B. Ahn, and E. P. Furlani. 2012. “Design of pressure-driven microfluidic networks using electric circuit analogy”. *Lab on a Chip* 12(3):515–545 <https://doi.org/10.1039/C2LC20799K>.
- Schuster, M. and K. Paliwal. 1997. “Bidirectional recurrent neural networks”. *IEEE Transactions on Signal Processing* 45(11):2673–2681 <https://doi.org/10.1109/78.650093>.
- Shampine, L. F. 1986. “Some practical Runge-Kutta formulas”. *Mathematics of Computation* 46:135–150 <https://doi.org/10.2307/2008219>.
- Shaw, M. T. 2011. *Introduction to Polymer Rheology*. John Wiley & Sons, Ltd <https://doi.org/https://doi.org/10.1002/9781118170229.ch1>.
- Srivastava, N. and M. A. Burns. 2006. “Analysis of Non-Newtonian Liquids Using a Microfluidic Capillary Viscometer”. *Analytical Chemistry* 78(5):1690–1696 <https://doi.org/10.1021/ac0518046>.
- Tabeling, P. 2005. *Introduction to Microfluidics*. Oxford University Press <https://doi.org/10.1017/S0022112006213855>.
- Takken, M. and R. Wille. 2024. “Accelerated Computational Fluid Dynamics Simulations of Microfluidic Devices by Exploiting Higher Levels of Abstraction”. *Micromachines* 15(1):129 <https://doi.org/10.3390/mi15010129>.
- Tipler, P. and G. Mosca. 2004. *Physics for Scientists and Engineers, Volume 2: Electricity, Magnetism, Light, and Elementary Modern Physics*. W.H. Freeman <https://doi.org/10.1007/978-3-662-67936-4>.
- Vinuesa, R. and S. L. Brunton. 2021. “The potential of machine learning to enhance computational fluid dynamics”. *arXiv preprint arXiv:2110.02085*:1–13.
- Wang, Y., R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly *et al.* 2017. “Tacotron: Towards End-to-End Speech Synthesis”. In *Interspeech* <https://doi.org/10.21437/Interspeech.2017-1452>.
- Zhao, Q., S. Yan, B. Zhang, K. Fan, J. Zhang and W. Li. 2023. “An On-Chip Viscoelasticity Sensor for Biological Fluids”. *Cyborg and Bionic Systems* 4 <https://doi.org/10.34133/cbsystems.0006>.

AUTHOR BIOGRAPHIES

JUAN SANDUBETE-LÓPEZ is an industrial PhD candidate by Universidad Complutense de Madrid, and at Microfluidic Innovation Center, Paris. He holds a masters degree in Systems and Control Engineering from the same university. His research interests include autonomous systems, instrumentation, deep learning, and simulation. His email address is jsandu01@ucm.es.

JOSÉ L. RISCO-MARTÍN received his Ph.D. from UCM, where he currently is Full Professor in the Department of Computer Architecture and Automation. His research interests include systems modeling, simulation, and optimization. His email address is jlrisco@ucm.es.

ALEXANDER MCMILLAN is an R&D project leader at the Microfluidics Innovation Center, where he oversees a team that develops new microfluidic tools through academia-industry collaborations. He earned his PhD from KU Leuven School of Bioscience Engineering, focusing on new polymers for microfluidic devices and their applications in flow chemistry and cell culture. His email address is alexhcmillan@gmail.com.

EVA BESADA-PORTAS is an Associate Professor of Systems Engineering and Automation at UCM. She also holds a PhD in Computer Systems from UCM. Her research interests include uncertainty modeling and simulation, optimal control and planning of unmanned vehicles. Her email address is ebesada@ucm.es.