POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Convex neural networks and Finite Element approximation of non-Newtonian fluids

Laurea Magistrale in Mathematical Engineering - Computational Science and Computational Learning

Author: Julian Vené

Advisor: Prof. M. Verani

Co-advisor: Prof. N. Parolini

Academic year: 2022-2023

1. Introduction

This article is concerned with the integration of neural networks inside a partial differential equation system. Within the framework of PDEs, artificial neural networks have increasingly become integrated into research. Examples of application can be found in [8] where a Physics-Informed Neural Network (PINN) is used to estimate the viscosity of polymer melts; or again in [6] where a neural network is used to model from data the anisotropy of materials with microstructure. Our focus will be devoted to the Stokes equations for non-Newtonian fluids, following the theory and Finite Element approximation results provided in [3]. Along with that, we will merge this theory with Input Convex Neural Networks (also called ICNN or IOCNN in the literature), developing the applications of such architecture firstly proposed in [9] [1].

1.1. Our contribution

In this research, our focus is on a novel approach to model non-Newtonian fluids using neural networks; specifically, we concentrate on modifying the Stokes equation for non-Newtonian fluids by employing an Input Convex Neural Network in place of the conventional viscous term. This modification is noteworthy for two reason: the first, because the ICNN architecture satisfies (as numerically verified later in the article) certain mathematical prerequisites that are crucial for obtaining a unique solution to the Stokes equations; secondly, this approach allows for more adaptable and data-driven modeling of fluid behaviors as opposed to the traditional reliance on mathematical models, such as the Carreau law or power law.
To assess the efficacy of our innovative Stokes-ICNN equation, we undertook an exhaustive analysis that entailed comparing its solutions with the exact solution, derived from the Stokes equation employing the Carreau law as the viscous term.

1.2. Guidelines

In Section 2, the Input Convex Neural Network (ICNN) is introduced and explained, highlighting its unique properties and strengths through theory and experiments.
Section 3 dives into the Stokes equation for non-Newtonian fluids, providing theoretical results

and numerical experiments for the validation of our Finite Element approximation method.
Section 4 incorporates ICNN into the Stokes equation, forming an innovative model for non-Newtonian fluids. The section establishes the ICNN's fulfillment of necessary conditions for the Stokes equation and confirms its effectiveness through comparison with exact solutions.

2. Input Convex Neural Networks

Let us first recall the definition of convex function, which is necessary to understand the proprieties of an Input Convex Neural Network:

Definition 2.1 (Convex function)**.** *A function $f : X \subseteq \mathbb{R}^n \to \mathbb{R}$ is convex if its domain is a convex set and for all $\mathbf{x}, \mathbf{y} \in X$ and all $\lambda \in [0, 1]$, we have:*

$$f(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1-\lambda) f(\mathbf{y})$$

Consider now a feed-forward fully connected neural network $\mathrm{NN} : \mathbb{R}^d \to \mathbb{R}$, where $d \geq 1$ is the dimension of the input, and L hidden layers. We will make use of the following notations:

- $w_{ij}^{(l)} \to$ weight connecting neuron j in layer $l-1$ to neuron i in layer l;
- $b_i^{(l)} \to$ bias of neuron i in layer l;
- $h_i^{(l)} \to$ output of neuron i in layer l, explicitly defined as: $h_i^{(l)} = \sigma_l(\sum_j w_{ij}^{(l)} h_j^{(l-1)} + b_i^{(l)})$.

The first and last layers correspond to the inputs and outputs of the neural network. Indeed, defining y_j as the j^{th} output, we have:

$$h_j^{(0)} = x_j \qquad (j = 1, ..., d)$$
$$h_j^{(L+1)} = y_j.$$

The feed-forward neural network NN turns into a *Input Convex Neural Network* (ICNN) if the following theorem is satisfied:

Theorem 2.1. *Consider a feed-forward fully connected neural network with L hidden layers, and activation function σ_l in layer l. The final output $\mathbf{y}$ is convex with respect to the input $\mathbf{x}$ if the following conditions hold:*

- $w_{ij}^{(2:L+1)} \geq 0$;
- *σ_l is convex and non-decreasing $\forall l \in \{1 : L+1\}$.*

For the proof see [4]. It is important to underline that under this condition the artificial neural network becomes convex in the sense of definition 2.1. In Figure 1 a direct comparison between ICNN and stantard NN is presented, and it is easy to see how the constraint of convexity is preventing the ICNN to overfit the data.

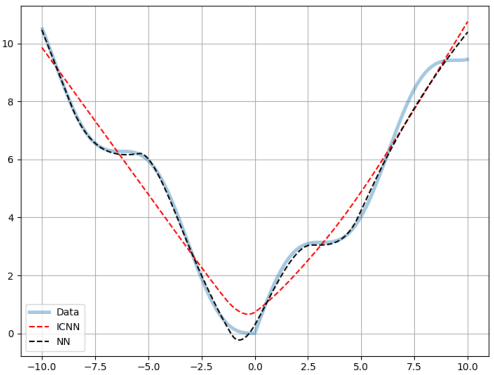


Figure 1: Comparison of prediction between ICNN and standard NN with same architecture, same training procedure.

Let us now consider a function $f : \mathbb{R}^d \to \mathbb{R}$ such that there exist $K > 0$ for which:

$$|f(\mathbf{x}_1) - f(\mathbf{x}_2)| \leq K\|\mathbf{x}_1 - \mathbf{x}_2\| \qquad \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^d$$

where $\|\cdot\|$ is the euclidean norm in $\mathbb{R}^d$. A function f that satifies the above propriety is said to be Lipschitz continous. It follows an important representation theorem for ICNNs:

Theorem 2.2 (Representation power of ICNN)**.** *Let f be a Lipschitz convex function over a compact domain $X \subset \mathbb{R}^d$. For any $\varepsilon > 0$ there exist a Input Convex Neural Network f_{ICNN} with ReLU activation functions such that:*

$$|f(\mathbf{x}) - f_{ICNN}(\mathbf{x})| \leq \varepsilon \qquad \forall \mathbf{x} \in X.$$

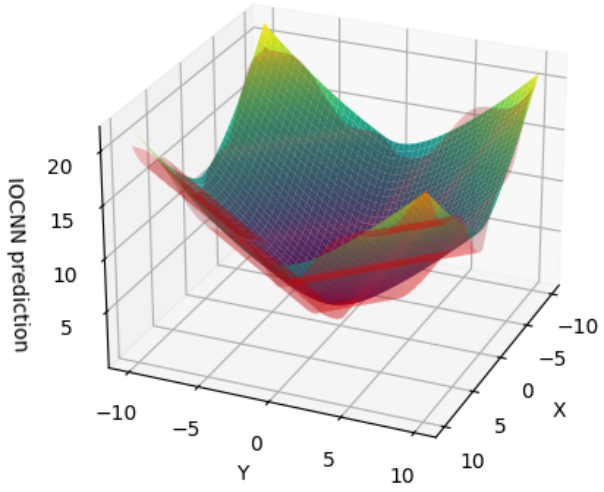
This theorem is stated and proved in [5]. What follows is that, when dealing with systems that can be effectively approximated by convex functions, ICNNs are able to approximate arbitrarily well the provided data.
Considering Theorem 2.1, it becomes important to regulate the sign of each weight $w_{ij}^{(2:L+1)}$ during the neural network training to maintain the

convexity. If a weight $\overline{w} \in \{w_{ij}^{(k)}\}_{i,j,k}$, where $k \in \{1 : L+1\}$, becomes negative during the training process, it must be made positive before the next training iteration. Various algorithms exist, such as nullifying negative weights by setting them to 0. However, the most effective method, as per our findings and [9], is the exponentiation algorithm.

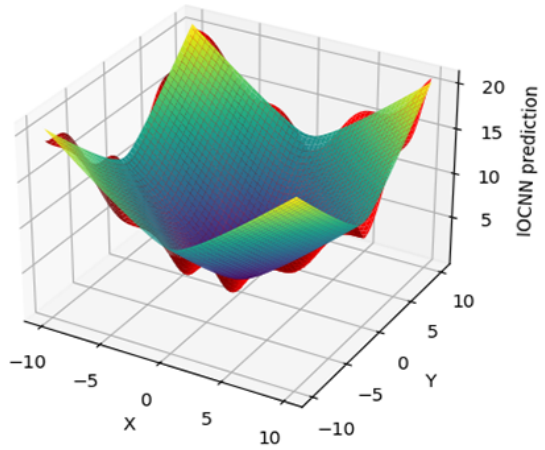
Algorithm 1 Exponentiation

```
constant ϵ
while training in progress do
    do training
    for layer ∈ 2 : L + 1 do
        if w < 0 then
            w ← e^{w−ϵ}
        end if
    end for
end while
```

The parameter ϵ serves the purpose of constraining the updated weights, ensuring their proximity to zero after exponentiation.
In order to assess the robustness of Input Convex Neural Networks (ICNNs), tests were conducted in a two-dimensional (2D) environment. These numerical results aimed to evaluate the ICNN's convex constraint over more complex scenario.



(a) $y(x, y) = |x| + |y| + \sin(x + y)$ — fig. 1



(b) $y(x, y) = |x| + |y| + \sin(x + y)$ — fig. 2

Figure 2: Two perspective of the same test. In red the data surface, in viridis the prediction of the ICNN.

In Figures 2a and 2b, different perspectives of the same dataset are presented, demonstrating that the ICNN effectively captured the function while disregarding the oscillations induced by the $\sin(x+y)$ term, producing this way a convex output.

3. Finite Element approximation of non-Newtonian Stokes flows

Let us consider an incompressible fluid inside a domain $\Omega \subset \mathbb{R}^2$, with Lipschitz boundary $\partial\Omega$. We denote by $\mathbf{u} : \Omega \to \mathbb{R}^2$ its velocity field and by $p : \Omega \to \mathbb{R}$ its pressure. We suppose that $(\mathbf{u}, p)$ satisfies the following equation:

$$\begin{cases} -\nabla \cdot [\tau(x, \varepsilon(\mathbf{u}))] + \nabla p = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \end{cases} \tag{1}$$

coupled with the homogeneous Dirichelet boundary condition for $\mathbf{u}$ and zero mean condition for the pressure p:

$$\mathbf{u} = \mathbf{0} \text{ on } \partial\Omega, \qquad \int_\Omega p = 0.$$

Here τ denotes the stress tensor, which is a suitable function of the strain rate tensor $\varepsilon(\mathbf{u})$ defined as:

$$\varepsilon_{ij}(\mathbf{u}) := \frac{1}{2}(\partial_{x_i} u_j + \partial_{x_j} u_i)$$

and the term $\mathbf{f}$ is a given body force. Regarding the stress tensor, we assume $\tau(x, \varepsilon(\mathbf{u})) =$

$\tau(\varepsilon(\mathbf{u}))$, in particular :

$$\tau(\varepsilon(\mathbf{u})) = k(\|\varepsilon(\mathbf{u})\|)\varepsilon(\mathbf{u}).$$

The norm $\|\cdot\|^2$ represents the squared Frobenius norm, i.e. for $\mathbf{K} \in \mathbb{R}^{n\times n}$ real matrix, $\|\mathbf{K}\|^2 = \|\mathbf{K}\|^2_{FRO} = \sum_{i,j=1}^n K_{ij}^2$.
The function $k(\cdot)$ represents the viscosity of the fluid, which results to be a constant in case of a Newtonian fluid. A popular choice for $k(\cdot)$, which will be considered in the sequel, is the Carreau law:

$$k(t) = k_\infty + (k_0 - k_\infty)(1 + \lambda t^2)^{(r-2)/2} \qquad (2)$$

where $k_0 > k_\infty \geq 0$, $\lambda > 0$ and $r \in (1, 2]$. Before introducing the weak formulation of (1), we first state an important set of assumptions for $k(\cdot)$ which are necessary for the existence and uniqueness of the solution to (1):

Assumptions (A): We assume that $k \in C(0,\infty)$ and that there exist constants $r \in (1,\infty)$, $\alpha \in [0,1]$ and $\varepsilon, C, M > 0$ such that:

$$k(t) \leq C[t^\alpha(1+t)^{1-\alpha}]^{r-2} \qquad \forall t \geq 0$$

$$|k(t)t - k(s)s| \leq C|t-s|[(t+s)^\alpha(1+t+s)^{1-\alpha}]^{r-2} \quad \forall t, s > 0 \text{ such that } |s/t - 1| \leq \varepsilon$$

$$k(t)t - k(s)s \geq M(t-s)[(t+s)^\alpha(1+t+s)^{1-\alpha}]^{r-2} \quad \forall t \geq s \geq 0.$$

These assumptions are satisfied by the Carreau law.

3.1. Weak formulation of the problem

Let us assume that $k(t)$ satisfies the assumptions **(A)**. Let $\mathbf{v} \in [W^{1,r}(\Omega)]^2$, we define:

$$J(\mathbf{v}) := \int_\Omega \Big[\int_0^{\|\varepsilon(\mathbf{v})\|} k(t)t\,dt \Big] - \langle \mathbf{f}, \mathbf{v} \rangle . \qquad (3)$$

We set $\mathbf{X} = [W^{1,r}(\Omega)]^2$, and let $\langle\cdot,\cdot\rangle_{\mathbf{X}^*}$ be the duality pairing between the dual space $\mathbf{X}^*$ and $\mathbf{X}$. It is easy to check that $J(\cdot)$ is Gateaux differentiable on $\mathbf{X}$ with:

$$\big\langle J'(\mathbf{w}_1), \mathbf{w}_2 \big\rangle_{\mathbf{X}^*} = \langle A\mathbf{w}_1, \mathbf{w}_2 \rangle_{\mathbf{X}^*} - \langle \mathbf{f}, \mathbf{w}_2 \rangle \quad \forall \mathbf{w}_1, \mathbf{w}_2 \in \mathbf{X}$$

where $A : \mathbf{X} \to \mathbf{X}^*$ is such that

$$\langle A\mathbf{w}_1, \mathbf{w}_2 \rangle_{\mathbf{X}^*} = \int_\Omega \langle k(\|\varepsilon(\mathbf{w}_1)\|)\varepsilon(\mathbf{w}_1), \varepsilon(\mathbf{w}_2) \rangle .$$

It is proved in [4] that:

$$k(\cdot) \text{ satisfies assumptions } \mathbf{(A)} \implies J(\mathbf{v}) \text{ is strictly convex on } \mathbf{X}.$$

From this result we have the following:

Proposition 3.1. *Let* $\mathbf{V} := \{\mathbf{v} \in \mathbf{X} : \nabla \cdot \mathbf{v} = 0 \text{ in } \Omega\}$. *The problem:*

$$J(\mathbf{u}) \leq J(\mathbf{v}) \qquad \forall \mathbf{v} \in \mathbf{V}$$

or equivalently:

$$\langle A\mathbf{u}, \mathbf{v} \rangle_{\mathbf{X}^*} = \langle \mathbf{f}, \mathbf{v} \rangle \qquad \forall \mathbf{v} \in \mathbf{V} \qquad (4)$$

admits a unique solution.

Let us now define $M = L_0^{r'}(\Omega)$, where r' is the conjugate exponent of r; we proceed to state the weak formulation of (1):
find $(\mathbf{u}, p) \in \mathbf{X} \times M$ such that:

$$\begin{cases} \langle A\mathbf{u}, \mathbf{w} \rangle_{\mathbf{X}^*} - \langle p, \nabla \cdot \mathbf{w} \rangle = \langle \mathbf{f}, \mathbf{w} \rangle & \forall \mathbf{w} \in \mathbf{X} \\ \langle \nabla \cdot \mathbf{u}, q \rangle = 0 & \forall q \in M. \end{cases} \qquad (5)$$

The unique solution $\mathbf{u}$ coincide with the solution of (4) by restricing $\mathbf{w} \in \mathbf{V} \subset \mathbf{X}$ in (5). For the well-posedness of p in (5) we require the Babuska/Brezzi inf-sup condition:

$$\inf_{q\in M} \sup_{\mathbf{w}\in\mathbf{X}} \frac{\langle q, \nabla \cdot \mathbf{w} \rangle}{\|q\|_M \|\mathbf{w}\|_{\mathbf{X}}} \geq \beta > 0 \qquad (6)$$

to hold. Amrouche and Girault (1990) proved in [2] that when taking $\mathbf{X} = [W^{1,r}(\Omega)]^2$ and $M = L_0^{r'}(\Omega)$, with r and r' conjugate exponents, there exist a constant $\beta(r)$ such that:

$$\inf_{q\in M} \sup_{\mathbf{w}\in\mathbf{X}} \frac{\langle q, \nabla \cdot \mathbf{w} \rangle}{\|q\|_M \|\mathbf{w}\|_{\mathbf{X}}} \geq \beta(r) > 0.$$

Hence (6) holds, and therefore the existence of a unique solution $(\mathbf{u}, p)$ in (5) is implied.

3.2. Finite Element approximation

Let $\mathbf{X}_h$ and M_h be two finite dimensional spaces such that

$$\mathbf{X}_h \subset \mathbf{X} \cap [W^{1,\infty}(\Omega)]^2 \quad , \quad M_h \subset M \cap L^\infty(\Omega).$$

The discretized version of (5) becomes:
find $(\mathbf{u}_h, p_h) \in \mathbf{X}_h \times M_h$ such that

$$\begin{cases} \langle A\mathbf{u}_h, \mathbf{w}_h\rangle_{\mathbf{X}^*} - \langle p_h, \nabla \cdot \mathbf{w}_h\rangle = \langle \mathbf{f}, \mathbf{w}_h\rangle \\ \langle \nabla \cdot \mathbf{u}_h, q_h\rangle = 0 \end{cases} \tag{7}$$

holds $\forall \mathbf{w}_h \in \mathbf{X}_h, \forall q_h \in M_h$.
We further introduce two hypothesis which are met when choosing $\mathbf{X}_h$ and M_h as continuous piece-wise polynomials, with a degree of m and $m-1$ respectively:

- **(H1)** Approximation property of $\mathbf{X}_h$.
  There is a continous linear operator $\pi_h : [W_0^{1,r}]^2 \to \mathbf{X}_h$ such that for $j = 0, ..., m$ we have

$$\|\mathbf{w} - \pi_h \mathbf{w}\|_{W^{j+1,r}(\Omega)} \leq Ch^j \|\mathbf{w}\|_{W^{j+1,r}(\Omega)} \quad \forall \mathbf{w} \in [W_0^{1,r} \cap W^{j+1,r}(\Omega)]^2$$

- **(H2)** Approximation property of M_h.
  There is a continous linear operator $\rho_h : L^{r'}(\Omega) \to M_h$ such that for all $j = 0, ..., m$ we have

$$\|q - \rho_h q\|_{L^{r'}(\Omega)} \leq Ch^j \|q\|_{L^{r'}(\Omega)} \quad \forall q \in L^{r'}(\Omega).$$

Under those assumptions we now provide an error bound Theorem which is going to be central for evaluating the numerical results provided later:

Theorem 3.1. *Assume that $k(\cdot)$ satisfies assumptions **(A)**. Let $(\mathbf{u}, p)$ be the unique solution of (5). Then if $r \in (1, 2]$ and **(H1)**, **(H2)** hold, for $j = 1, ..., m$ we have that the unique solution of (7) $(\mathbf{u}_h, p_h) \in \mathbf{X}_h \times M_h$ is such that*

$$\|\mathbf{u} - \mathbf{u}_h\|_{W^{1,r}(\Omega)} + (\|p - p_h\|_{L^{r'}(\Omega)})^{r/[2(r-1)]} \leq C_1 h^{rj/2} \tag{8}$$

where $C_1 = C(\|\mathbf{u}\|_{W^{j+1,r}(\Omega)}, \|p\|_{L^{r'}(\Omega)}, \beta_h(r)^{-1}, \beta_h^{-1})$.

For the proof see [3]. Equation (8) yields the following essential results:

$$\begin{aligned} \|\mathbf{u} - \mathbf{u}_h\|_{W^{1,r}(\Omega)} &\leq Ch^{rj/2}, \\ \|p - p_h\|_{L^{r'}(\Omega)} &\leq Ch^{j(r-1)}. \end{aligned} \tag{9}$$

These two convergence results will serve as the criteria for evaluating the performance of the Finite Element approximation later implemented.

3.3. Numerical results

We now solve numerically the system of equation (7) with Carreau law (2) with parameters $k_0 = 2$, $k_\infty = 0$, $\lambda = 2$ and $r = 1.2, 1.6, 2$, inside the domain $\Omega = (-0.5, 0.5)^2$. The source term $\mathbf{f}$ is manufactured so that the exact solution $(\overline{\mathbf{u}}, \overline{p})$ is given by:

$$\overline{\mathbf{u}}(x, y) = \begin{bmatrix} 5y\sin(x^2+y^2) + 4y\sin(x^2-y^2) \\ -5x\sin(x^2+y^2) + 4x\sin(x^2-y^2) \end{bmatrix}$$
$$\overline{p}(x, y) = \sin(x + y).$$

Dirichlet boundary conditions for velocity given by the exact solution are imposed on the domain boundary: $\mathbf{u}|_{\partial\Omega} = \overline{\mathbf{u}}|_{\partial\Omega}$. Our function spaces will be:

$$\begin{aligned} \mathbf{X}_h := \{\, \mathbf{w}_h \in [C(\Omega)]^2 : \mathbf{w}_h|_\tau \in P_2 \,\forall \tau \in T^h, \\ \text{and } \mathbf{w}_h = \overline{\mathbf{u}} \text{ on } \partial\Omega \,\} \\ M_h := \{\, q_h \in L_0^2(\Omega) : q_h|_\tau \in P_1 \,\forall \tau \in T^h \,\} \end{aligned}$$

where P_m is the space of all polynomials of degree less than or equal to m, and T^h is a collection of disjoint open regular triangles τ such that $\bigcup_{\tau \in T^h} \tau = \Omega$. In order to determine the solution $(\mathbf{u}_h, p_h)$ of (7) the *Line Search Newton Method* [7] was used.
In Table 1 the convergence orders are presented for different values of the parameter r, when choosing polynomial approximator of degree 2 for $\mathbf{u}$ and degree 1 for p.

Conv. ord.	$\|\overline{\mathbf{u}} - \mathbf{u}_h\|_{W^{1,r}(\Omega)}$	$\|\overline{p} - p_h\|_{L^{r'}(\Omega)}$
$r = 1.2$	2.0	2.03
$r = 1.6$	2.0	2.02
$r = 2.0$	2.0	2.03

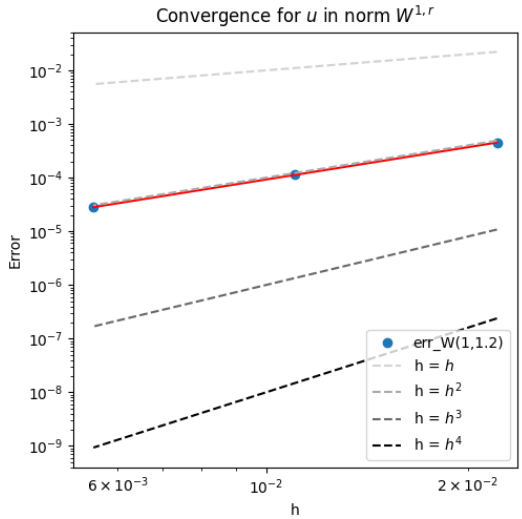
Table 1: Convergence rates for different value of r, with polynomial approximator of degree 2 for $\mathbf{u}$ and degree 1 for p.

Same results were carried out when increasing the polynomial approximators of $\mathbf{u}$ and p by one:

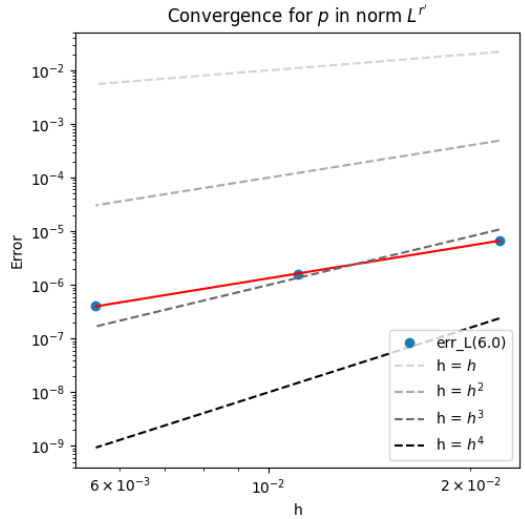
Conv. ord.	$\|\overline{\mathbf{u}} - \mathbf{u}_h\|_{W^{1,r}(\Omega)}$	$\|\overline{p} - p_h\|_{L^{r'}(\Omega)}$
$r = 1.2$	3.01	3.08
$r = 1.6$	3.01	3.24
$r = 2.0$	3.01	3.45

Table 2: Convergence rates for different value of r, with polynomial approximator of degree 3 for $\mathbf{u}$ and degree 2 for p.

To provide an example, in Figure 3 we showcase the convergence of the errors when choosing polynomial approximator of degree 2 for $\mathbf{u}$ and degree 1 for p, in the scenario $r = 1.2$.



(a) Convergence of $\mathbf{u}_h$.



(b) Convergence of p_h.

Figure 3: $r = 1.2$, polynomial approximator of degree 2 for $\mathbf{u}$ and 1 for p.

Data in Tables 1 and 2 demonstrates that the computational results are in line with theoretical predictions reported in Theorem 3.1. Notably, in every scenario, the observed convergence order agree with the theoretical expectations (showing a better order of convergence). The same pattern is evident as the value of r increases. As anticipated by the theory, higher values of r correspond to improved convergence results.

4. Non-Newtonian Stokes equations with neural networks

In this section we replace the Carreau law $k(\cdot)$ inside (5) with an Input Convex Neural Network, $\text{ICNN}_{\boldsymbol{\theta}} : \mathbb{R} \to \mathbb{R}$, obtaining the new description of the stress tensor as:

$$\tau(\varepsilon(\mathbf{u})) = \text{ICNN}_{\boldsymbol{\theta}}(\|\varepsilon(\mathbf{u})\|)\varepsilon(\mathbf{u})$$

where $\boldsymbol{\theta}$ is the set of parameters that uniquely identify our neural network (obtained through the process of training). In particular, we define the ICNN-Stokes equations to be:

$$\begin{cases} \int_\Omega \text{ICNN}_{\boldsymbol{\theta}}(\|\varepsilon(\mathbf{u})\|_{FRO})\varepsilon(\mathbf{u}) : \varepsilon(\mathbf{w}) \\ \qquad - \int_\Omega p\nabla \cdot \mathbf{w} = \int_\Omega \mathbf{f} \cdot \mathbf{w} \\ \int_\Omega q\nabla \cdot \mathbf{u} = 0. \end{cases} \tag{10}$$

In Figure 4, we present the output of the ICNN when trained on data sampled from the Carreau law with the inclusion of a noise source. This test allows us to benchmark the performance of the ICNN in simulating real measurements, where the presence of noise cannot be neglected.

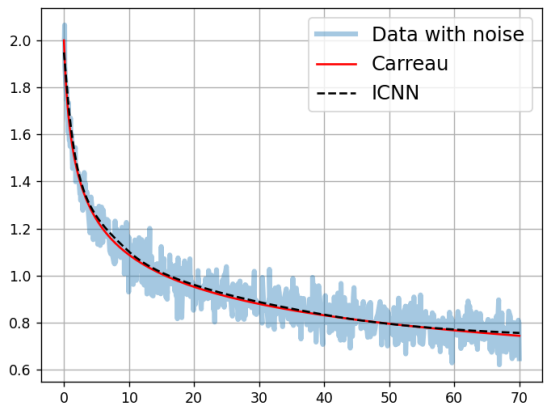


Figure 4: ICNN tested on Carreau law with parameters: $k_\infty = 0, k_0 = 2, \lambda = 2, r = 1.6$.

In order to verify that the trained ICNN satisfies the assumptions **(A)** and hence the existence and uniqueness of the solution of the

ICNN-Stokes equations (10) is guaranteed, the minimization Algorithm 2, coded for this task, was used. Algorithm 2 is not only a validation tool for the assumptions; it serves a dual purpose. Firstly, it rigorously verify whether the trained ICNN is in compliance with the assumptions **(A)**. Secondly, it fine-tunes the values of costants $r \in (1,2)$, $\alpha \in [0,1]$, and $C, M > 0$, in such a way that C and α are minimized, while M is maximized.

Algorithm 2 Assumption verification algorithm

initial guess $C = C_0$, $\alpha = \alpha_0$, $r = r_0$, $M = M_0$; $t_1, ..., t_N \in (0, \infty)$
$k = 0$, max iterations $N_1, N_2 > 0$
while $k \leq N_1$ **do**
  Compute $f = \sum_{j=1}^{N} C_k[t_j^{\alpha_k}(1 + t_j)^{1-\alpha_k}]^{r_k-2} - \text{ICNN}_{\boldsymbol{\theta}}(t_j)$
  minimize $|f|$ with respect to C_k, α_k, r_k imposing the constraint $C_k[t_j^{\alpha_k}(1 + t_j)^{1-\alpha_k}]^{r_k-2} \geq \text{ICNN}_{\boldsymbol{\theta}}(t_j) \quad \forall j = 1, ..., N$
  $k = k + 1$
end while
$k = 0$, $s_1, ..., s_N \in (0, \infty)$
while $k \leq N_2$ **do**
  Compute $f = \sum_{j=1}^{N} \sum_{i=1, s_i \leq t_j}^{N} \text{ICNN}_{\boldsymbol{\theta}}(t_j)t_j - \text{ICNN}_{\boldsymbol{\theta}}(s_i)s_i - M_k(t_j - s_i)[(t_j + s_i)^{\alpha_{N_1}}(1 + t_j + s_i)^{1-\alpha_{N_1}}]^{r_{N_1}-2}$
  Minimize $|f|$ with respect to M_k imposing the constraint $\text{ICNN}_{\boldsymbol{\theta}}(t_j)t_j - \text{ICNN}_{\boldsymbol{\theta}}(s_i)s_i \geq M_k(t_j - s_i)[(t_j + s_i)^{\alpha_{N_1}}(1 + t_j + s_i)^{1-\alpha_{N_1}}]^{r_{N_1}-2} \quad \forall t_j \geq s_i$
  $k = k + 1$
end while

Numerical results were carried on noisy-free Carreau law (2) data samples, with parameters $k_0 = 2, k_\infty = 0, \lambda = 2$ and $r = 1.2, 1.6, 2$. For each value of r, a distinct convex neural network (ICNN) was created and trained using a separate training set. The subsequent table presents the outcomes of the training process conducted on these convex neural networks, which were subsequently tested.

	L^2-error	MSE error
$\text{ICNN}_{r=1.2}$	6.7e-4	3.4e-7
$\text{ICNN}_{r=1.6}$	1.5e-4	7.7e-8
$\text{ICNN}_{r=2.0}$	2.5e-8	1.8e-12

Table 3: L^2 and MSE errors obtained on the validation set.

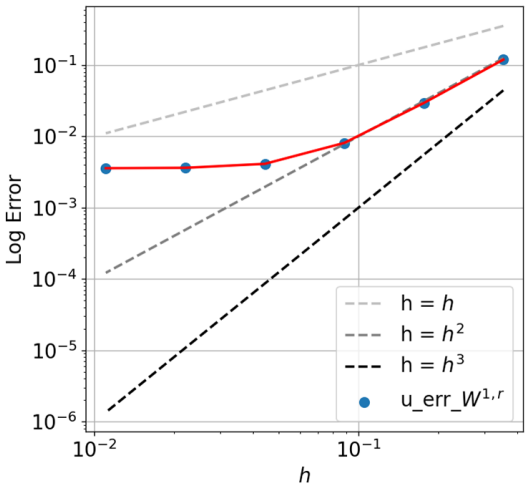
Algorithm 2 was used on each neural network to verify that they satisfies Assumption **(A)**, obtaining the following results:

	$\text{ICNN}_{r=1.2}$	$\text{ICNN}_{r=1.6}$	$\text{ICNN}_{r=2.0}$
C	11.577	4.692	4.742
α	0.100	0.102	0.499
r	1.170	1.590	1.917
M	2.192	3.019	2.021

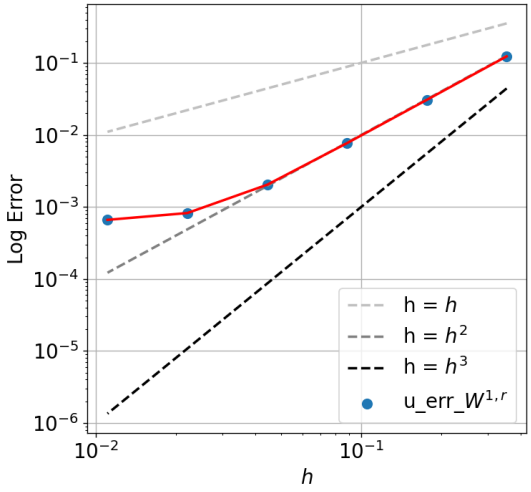
Table 4: Optimized constants that ensure the ICNNs satisfy the assumptions **(A)** and hence the existence and uniqueness of the solution to (10) is guaranteed.

4.1. Convergence results

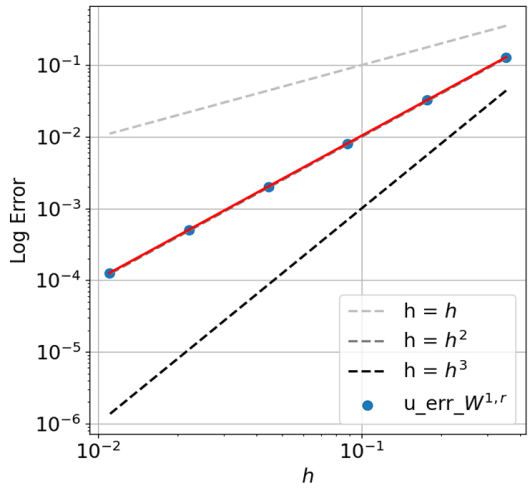
Let $(\mathbf{u}_{h,ICNN}, p_{h,ICNN})$ be the unique solution of the discretized ICNN-Stokes equations (10). In Figures 4 and 5 we show the convergence results of this solution towards the exact solution $(\overline{\mathbf{u}}, \overline{p})$ of (5) as the mesh is refined, for $r = 1.2, 1.6, 2$.



(a) Case $r = 1.2$.

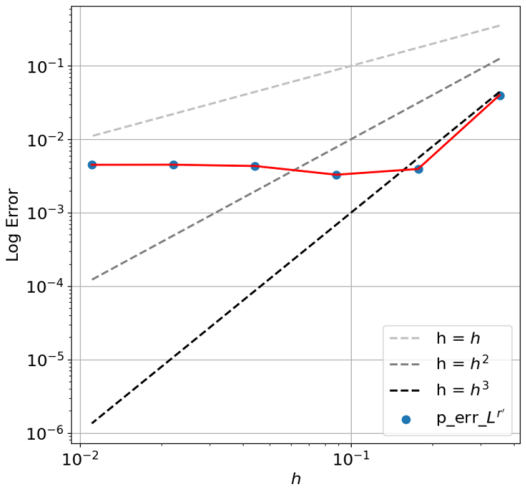


(b) Case $r = 1.6$.

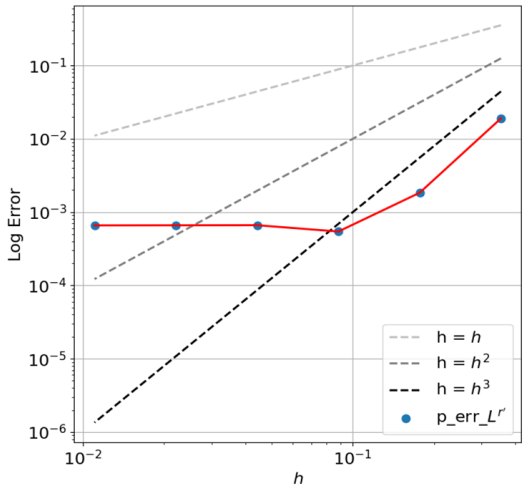


(c) Case $r = 2$.

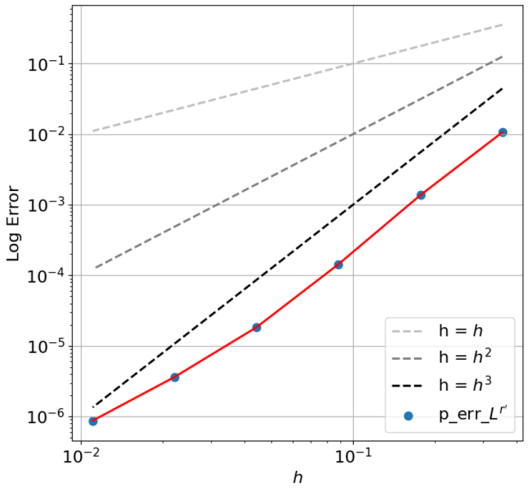
Figure 4: Errors $\|\overline{\mathbf{u}} - \mathbf{u}_{h,ICNN}\|_{W^{1,r}(\Omega)}$ for $r = 1.2, 1.6, 2$.



(d) Case $r = 1.2$.



(e) Case $r = 1.6$.



(f) Case $r = 2$.

Figure 5: Errors $\|\overline{p} - p_{h,ICNN}\|_{W^{1,r}(\Omega)}$ for $r = 1.2, 1.6, 2$.

Examining the plots in Figure 4 and 5 reveals two key findings. First, the errors in $\mathbf{u}_{h,ICNN}$ and $p_{h,ICNN}$ steadily decrease to a minimum in all different scenarios (apart the trivial case $r = 2$ where the function $k(\cdot)$ reduces to a constant). Secondly, the value of the plateau is related to the validation error of ICNN when trained to approximate the Carreau law $k(\cdot)$ (see Table 3). More precisely, denoting $\varepsilon_{\text{ICNN},u}$ and $\varepsilon_{\text{ICNN},p}$ as

$$\lim_{h\to 0} \|\overline{\mathbf{u}} - \mathbf{u}_{h,ICNN}\|_{W^{1,r}(\Omega)} = \varepsilon_{\text{ICNN},u}$$

$$\lim_{h\to 0} \|\overline{p} - p_{h,ICNN}\|_{L^{r}(\Omega)} = \varepsilon_{\text{ICNN},p}$$

we note a correlation between $\varepsilon_{\text{ICNN},u}$, $\varepsilon_{\text{ICNN},p}$ and the L^2 error in Table 3, as it is evident from Figure 6. In particular, we note that an improvement of the accuracy of ICNN (i.e. reduction of

L^2 validation error) corresponds with lower values of $\varepsilon_{\text{ICNN},u}$ and $\varepsilon_{\text{ICNN},p}$.

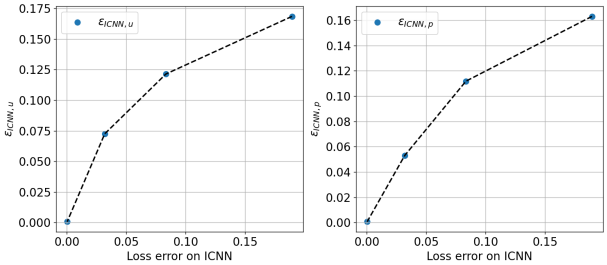


Figure 6: $\varepsilon_{\text{ICNN},u}$ and $\varepsilon_{\text{ICNN},p}$ computed with ICNN increasing performace.

These results stress the importance of validating the ICNN's performance over a validation set and highlight the potential use of ICNN inside the non-Newtonian Stokes problem.

Along with this findings, we propose a new theoretical result through which we are able to bound the error $\varepsilon_{\text{ICNN},u}$:

Proposition 4.1. *Let* $\mathbf{X} = [W^{1,r}(\Omega)]^2$, $M = L^{r'}(\Omega)$. *Let* $A_i : \mathbf{X} \to \mathbf{X}^*$ *be defined as:* $\langle A_i\mathbf{v}, \mathbf{w}\rangle_{\mathbf{X}^*} := \int_\Omega k_i(\|\varepsilon(\mathbf{v})\|)\varepsilon(\mathbf{v}) : \varepsilon(\mathbf{w})$. *Let* $(\mathbf{u}_i, p_i)$ *be the solutions of:*

$$\begin{cases} \langle A_i\mathbf{u}, \mathbf{w}\rangle_{\mathbf{X}^*} - \langle p, \nabla\cdot\mathbf{w}\rangle = \langle \mathbf{f}, \mathbf{w}\rangle & \forall \mathbf{w} \in \mathbf{X} \\ \langle \nabla\cdot\mathbf{u}, q\rangle = 0 & \forall q \in M \end{cases}$$

for $i = 1, 2$. *Then the following inequality hold:*

$$\begin{aligned} &\|\mathbf{u}_1 - \mathbf{u}_2\|_{W^{1,r}(\Omega)} \\ &\leq C\|k_1 - k_2\|_{L^\infty(0,\infty)}\|\varepsilon(\mathbf{u}_2)\|_{L^{r'}(\Omega)} \end{aligned} \tag{11}$$

where $C > 0$ *is a constant.*

Proof. From [3]:

$$\begin{aligned} &\|\mathbf{u}_1 - \mathbf{u}_2\|_{W^{1,r}(\Omega)} \leq \langle A_1\mathbf{u}_1 - A_1\mathbf{u}_2, \mathbf{u}_1 - \mathbf{u}_2\rangle_{\mathbf{X}^*} \\ &= \underbrace{\langle A_1\mathbf{u}_1 - A_2\mathbf{u}_2, \mathbf{u}_1 - \mathbf{u}_2\rangle_{\mathbf{X}^*}}_{(i)} + \\ &+ \underbrace{\langle A_2\mathbf{u}_2 - A_1\mathbf{u}_2, \mathbf{u}_1 - \mathbf{u}_2\rangle_{\mathbf{X}^*}}_{(ii)}. \end{aligned}$$

We than have:

$$\begin{aligned} (i) &= \langle p_2, \nabla\cdot(\mathbf{u}_1 - \mathbf{u}_2)\rangle - \langle p_1, \nabla\cdot(\mathbf{u}_1 - \mathbf{u}_2)\rangle \\ &= 0 \end{aligned}$$

as $\nabla\cdot\mathbf{u}_1 = 0$. Regarding the second expression $(ii) =$

$$\begin{aligned} &\int_\Omega (k_2(\|\varepsilon(\mathbf{u}_2)\|) - k_1(\|\varepsilon(\mathbf{u}_2)\|))\varepsilon(\mathbf{u}_2) : \varepsilon(\mathbf{u}_1 - \mathbf{u}_2) \\ &\leq \|k_1 - k_2\|_{L^\infty(0,\infty)} \int_\Omega \varepsilon(\mathbf{u}_2) : \varepsilon(\mathbf{u}_1 - \mathbf{u}_2). \end{aligned}$$

Using the Hölder inequality we then obtain $(ii) \leq$

$$\begin{aligned} &\|k_1 - k_2\|_{L^\infty(0,\infty)}\|\varepsilon(\mathbf{u}_2)\|_{L^{r'}}\|\varepsilon(\mathbf{u}_1 - \mathbf{u}_2)\|_{L^r} \\ &\leq C\|k_1 - k_2\|_{L^\infty(0,\infty)}\|\varepsilon(\mathbf{u}_2)\|_{L^{r'}}\|\mathbf{u}_1 - \mathbf{u}_2\|_{W^{1,r}}. \end{aligned}$$

where $C > 0$ is a constant. Inserting the results on (i) and (ii) into the first equation we obtain:

$$\|\mathbf{u}_1 - \mathbf{u}_2\|_{W^{1,r}} \leq C\|k_1 - k_2\|_{L^\infty(0,\infty)}\|\varepsilon(\mathbf{u}_2)\|_{L^{r'}}.$$

□

Remark. If we assume $\|\varepsilon(\mathbf{u}_2)\|_{L^{r'}(\Omega)} < \infty$, equation (11) can be further improved to:

$$\|\mathbf{u}_1 - \mathbf{u}_2\|_{W^{1,r}(\Omega)} \leq C\|k_1 - k_2\|_{L^\infty(0,\infty)} \tag{12}$$

where $C = C(\|\varepsilon(\mathbf{u}_2)\|_{L^{r'}(\Omega)})$.

In order to understand the importance of this result, let's first introduce $(\overline{\mathbf{u}}_{ICNN}, \overline{p}_{ICNN})$, formally defined as:

$$\lim_{h\to 0} \mathbf{u}_{h,ICNN} = \overline{\mathbf{u}}_{ICNN}$$

$$\lim_{h\to 0} p_{h,ICNN} = \overline{p}_{ICNN}.$$

Using Proposition 4.1 with $k_1(t) = k(t)$, which is the explicit viscous law of the fluid (in our case the Carreau law) and setting $k_2(t) = \text{ICNN}_{\boldsymbol{\theta}}(t)$, and taking into account that $\|\varepsilon(\overline{\mathbf{u}}_{ICNN})\|_{L^{r'}(\Omega)} < \infty$, we then obtain the bound for $\varepsilon_{\text{ICNN},u}$:

$$\begin{aligned} \varepsilon_{\text{ICNN},u} &= \|\overline{\mathbf{u}} - \overline{\mathbf{u}}_{ICNN}\|_{W^{1,r}(\Omega)} \\ &\leq C\|k - \text{ICNN}_{\boldsymbol{\theta}}\|_{L^\infty(0,\infty)} \end{aligned} \tag{13}$$

where $C = C(\|\varepsilon(\overline{\mathbf{u}}_{ICNN})\|_{L^{r'}(\Omega)})$. This result implies an important fact: the better the ICNN approximate the viscous law (under the L^∞ norm), the better our ICNN-Stokes system is able to reconstruct the expected solution for the velocity $\mathbf{u}$.

Indeed, from Table 5, where we approximated $(\overline{\mathbf{u}}_{ICNN}, \overline{p}_{ICNN})$ with $(\mathbf{u}_{\overline{h},ICNN}, p_{\overline{h},ICNN})$,

where $\overline{h}$ represents an infinitesimal value for h (in particular $\overline{h} = 0.0027$), we are able to see how the L^∞-error between the viscous law and the Input Convex Neural Network is related to $\varepsilon_{\text{ICNN},u}$:

	$\varepsilon_{\text{ICNN},u}$	$\|k - \text{ICNN}_{\boldsymbol{\theta}}\|_{L^\infty(0,\infty)}$
$r = 1.2$	0.001094	0.0202
$r = 1.6$	0.000662	0.0155
$r = 2.0$	0.0000316	0.00001

Table 5: Relationship between $\varepsilon_{\text{ICNN},u}$ and the L^∞-error between Carreau law and ICNN, for different r.

We now turn our attention to demonstrating the convergence of $(\mathbf{u}_{h,ICNN}, p_{h,ICNN})$ towards the solution $(\overline{\mathbf{u}}_{ICNN}, \overline{p}_{ICNN})$, approximated with $(\mathbf{u}_{\overline{h},ICNN}, p_{\overline{h},ICNN})$ previously introduced. The obtained convergence orders are reported in in Tables 6 and 7.

	$\|\overline{\mathbf{u}}_{ICNN} - \mathbf{u}_{h,ICNN}\|_{W^{1,r}(\Omega)}$	$\text{Exp}(\mathbf{u})$
$r = 1.2$	2.54	1.2
$r = 1.6$	2.59	1.6
$r = 2.0$	2.0	2.0

Table 6: Convergence rates of $\mathbf{u}_{h,ICNN}$ towards solution $\overline{\mathbf{u}}_{ICNN}$, compared with expected theoretical results.

	$\|\overline{p}_{ICNN} - p_{h,ICNN}\|_{L^{r'}(\Omega)}$	$\text{Exp}(p)$
$r = 1.2$	2.09	0.4
$r = 1.6$	2.52	1.2
$r = 2.0$	2.2	2.0

Table 7: Convergence rates of $p_{h,ICNN}$ towards solution $\overline{p}_{ICNN}$, compared with expected theoretical results.

In line with the theoretical convergence orders obtained using (9), the data provides evidence that the computational results align with the theoretical predictions; indeed, across all scenarios, the observed convergence order consistently matches or exceeds the expected theoretical results.

In light of these findings, Figure 4 can be further commented. Indeed we have:

$$\begin{aligned}&\|\overline{\mathbf{u}} - \mathbf{u}_{h,ICNN}\|_{W^{1,r}}\\ &\leq \|\overline{\mathbf{u}}_{ICNN} - \mathbf{u}_{h,ICNN}\|_{W^{1,r}} + \|\overline{\mathbf{u}} - \overline{\mathbf{u}}_{ICNN}\|_{W^{1,r}}\\ &\leq h^{jr/2} + \|k - \text{ICNN}_{\boldsymbol{\theta}}\|_{L^\infty(0,\infty)}\end{aligned}$$

where j is the degree of the polynomial approximator of the $\mathbf{u}$. In fact, from Figure 4 we can see how the error $\|\overline{\mathbf{u}} - \mathbf{u}_{h,ICNN}\|_{W^{1,r}}$ decreases with the expected rate, till it reaches a plateau generated by $\|k - \text{ICNN}_{\boldsymbol{\theta}}\|_{L^\infty(0,\infty)}$.

5. Conclusions

Having verified the numerical solver's consistency in Section 3, we integrated the Input Convex Neural Network (ICNN) into the Stokes equation for non-Newtonian fluids, forming a new ICNN-Stokes system of equations.
We proceeded to prove numerically that the Input Convex Neural Networks satisfy Assumptions **(A)** which are a key condition in order for the Stokes equations to admit a unique solution. Section 4 proved that the ICNN-Stokes system accurately recreates the toy problem presented in Section 3, bolstering our belief in using ICNN for representing the fluid's viscous law, and new theoretical results regarding the capability of approximation of the velocity obtained from ICNN-Stokes, were provided.

References

[1] Brandon Amos, Lei Xu, and J. Zico Kolter. Input convex neural networks, 2017. arXiv:1609.07152.

[2] Chérif Amrouche and Vivette Girault. Decomposition of vector spaces and application to the stokes problem in arbitrary dimension. *Czechoslovak Mathematical Journal*, 44(1):109–140, 1994.

[3] John W. Barrett and W.B. Liu. Quasi-norm error bounds for the finite element approximation of a non-newtonian flow. *Numerische Mathematik*, 68(4):437–456, 1994.

[4] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[5] Yize Chen, Yuanyuan Shi, and Baosen Zhang. Optimal control via neural networks: A convex approach, 2019. arXiv:1805.11835.

[6] J.N. Fuhg, N. Bouklas, and R.E. Jones. Learning hyperelastic anisotropy from data via a tensor basis neural network. *Journal of the Mechanics and Physics of Solids*, 168:105022, 2022.

[7] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, 2e edition, 2006.

[8] Brandon Reyes, Amanda A. Howard, Paris Perdikaris, and Alexandre M. Tartakovsky. Learning unknown physics of non-newtonian fluids. *Physical Review Fluids*, 6:073301, Jul 2021.

[9] Sarath Sivaprasad, Ankur Singh, Naresh Manwani, and Vineet Gandhi. The curious case of convex neural networks, 2021. arXiv:2006.05103.