Multi-Constitutive Neural Network for Large Deformation Poromechanics Problem

Qi Zhang[1], Yilin Chen[1*], Ziyi Yang[2*], Eric Darve[2,3]*

[1]Department of Civil and Environmental Engineering, Stanford University, [2]Department of Mechanical Engineering, Stanford University

[3]Institute for Computational and Mathematical Engineering, Stanford University

**Equal Contribution*

Motivations

- **PINN:** In many engineering fields, a fair number of problems are closely related to differential equations. Physics-informed neural network (PINN, by Raissi et al 2017) is proposed to solve a given physics problem described by a specific partial differential equation (PDE).
- **Multi-Constitutive Problem**: Multiple PDEs exist for a given problem, corresponding to choosing different constitutive laws. In our problem, we have a universal PDE (mass balance equation):

$$\frac{\partial J}{\partial \hat{t}} - \frac{1}{\varphi_0^3}\frac{\partial}{\partial \hat{X}}\left[\frac{(J-1+\varphi_0)^3}{J^2}\frac{\partial \hat{p}}{\partial \hat{X}}\right] = 0$$

and three constitutive laws of hyper-elasticity:

Law1: $\frac{\partial \hat{p}}{\partial \hat{X}} = \frac{3J^2-1}{2}\frac{\partial J}{\partial \hat{X}}$

Law2: $\frac{\partial \hat{p}}{\partial \hat{X}} = \left[\hat{\gamma}\frac{1-\log J}{J^2} + \hat{\mu}\left(3J^2-1\right)\right]\frac{\partial J}{\partial \hat{X}}$

Law3: $\frac{\partial \hat{p}}{\partial \hat{X}} = \left[\hat{\gamma}\frac{1-\log J}{J^2} + \hat{\mu}\left(1+\frac{1}{J^2}\right)\right]\frac{\partial J}{\partial \hat{X}}$

Our Approach: MCNN

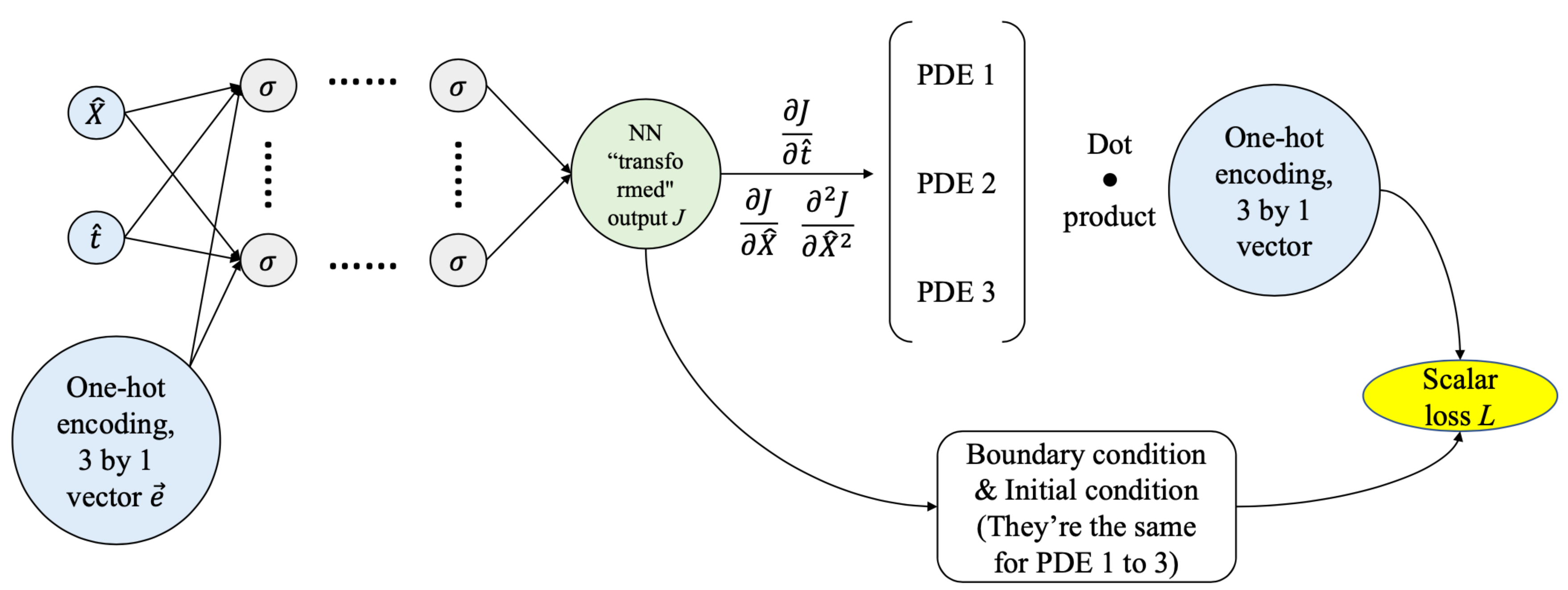

Figure 1: A schematic diagram of our proposed MCNN.

- **MCNN Loss:**

$$\mathcal{L} = \left\{\sum_{i=1}^{3} e_i f_i\left[\frac{\partial}{\partial \hat{t}}J\left(\hat{X},\hat{t},\vec{e}\right), \frac{\partial}{\partial \hat{X}}J\left(\hat{X},\hat{t},\vec{e}\right), \frac{\partial^2}{\partial \hat{X}^2}J\left(\hat{X},\hat{t},\vec{e}\right)\right]\right\}^2 + \mathcal{L}_{BI}$$

$$\mathcal{L}_{BI} = \begin{cases} \left[J\left(0,\hat{t},\vec{e}\right)-\bar{J}\right]^2 & \hat{X}=0, \quad \hat{t}>0 \\ \left[\frac{\partial}{\partial \hat{X}}J\left(\hat{X},\hat{t},\vec{e}\right)\Big|_{\hat{X}=1}\right]^2 & \hat{X}=1, \quad \hat{t}>0 \\ \left[J\left(\hat{X},0,\vec{e}\right)-1\right]^2 & \hat{t}=0, \quad 0<\hat{X}<1 \\ 0 & \hat{t}>0, \quad 0<\hat{X}<1 \end{cases}$$

- **Training Details:**

Table 1: Material properties (same for all three PDEs).

$\hat{\gamma}$	$\hat{\mu}$	φ_0	$\bar{J}$	$\hat{t}$
1/3	1/3	0.3	0.8	$0 \le \hat{t} \le 1$

Table 2: Model parameters of the neural networks. The total size of the training set for MCNN is the same as each independent PINN. All architectures use 5 (hidden layers) $\times$ 50 (neurons) with `tanh` as the activation function. The test sets are always equispaced within the domain. Due to the issue of numerical instability of the Saint-Venant Kirchhoff law [22], we adopt a larger number of epochs for law 1 to make the optimization process more stable. When the number of epochs is fixed, we tune the learning rate and find that the best value is `5e-4`. These learning rates and numbers of epochs are also typical values used in [7, 8, 9].

DNN	Training set	Test set	Optim.	Rate	Epochs
MCNN (ours)	1000 (per law)	10^4 (per law)	Adam [12]	5e-4	10^5
PINN of law 1	3000	10^4	Adam	5e-4	5×10^4
PINN of law 2	3000	10^4	Adam	5e-4	2×10^4
PINN of law 3	3000	10^4	Adam	5e-4	10^4

Results

- Surprisingly MCNN, which is trained for all three laws, achieves higher prediction accuracy than the independent PINN on law 2 and law 3 (see the table below).
- Visualization in Figure 2 and Figure 3 shows that MCNN produces clearly distinct and accurate predictions for all three laws.

Method	Law 1	Law 2	Law 3
MCNN (Ours)	0.3864%	**0.1316%**	**0.1658%**
Independent PINN	**0.2888%**	0.2392%	0.4089%

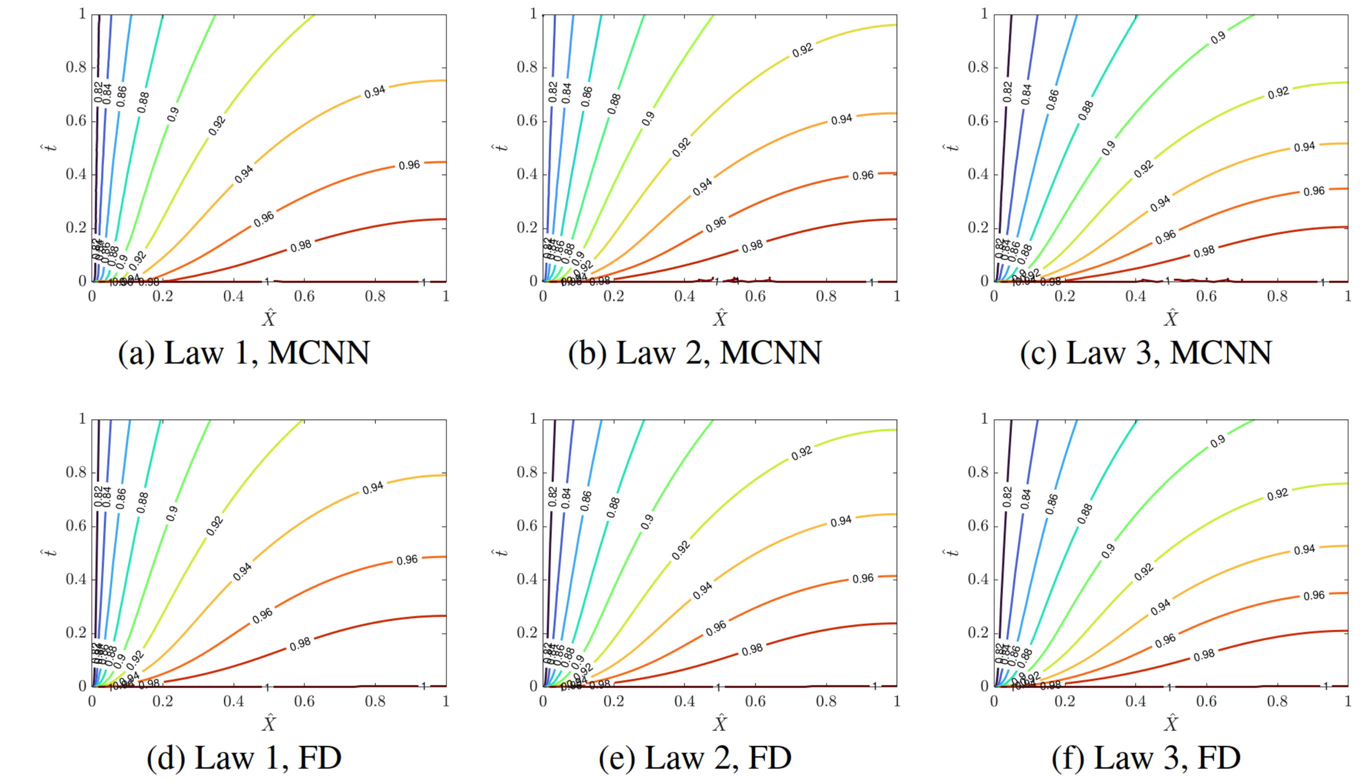

Figure 2: Plots of predicted J with MCNN and FD. MCNN is able to make an accurate inference for each constitutive law.

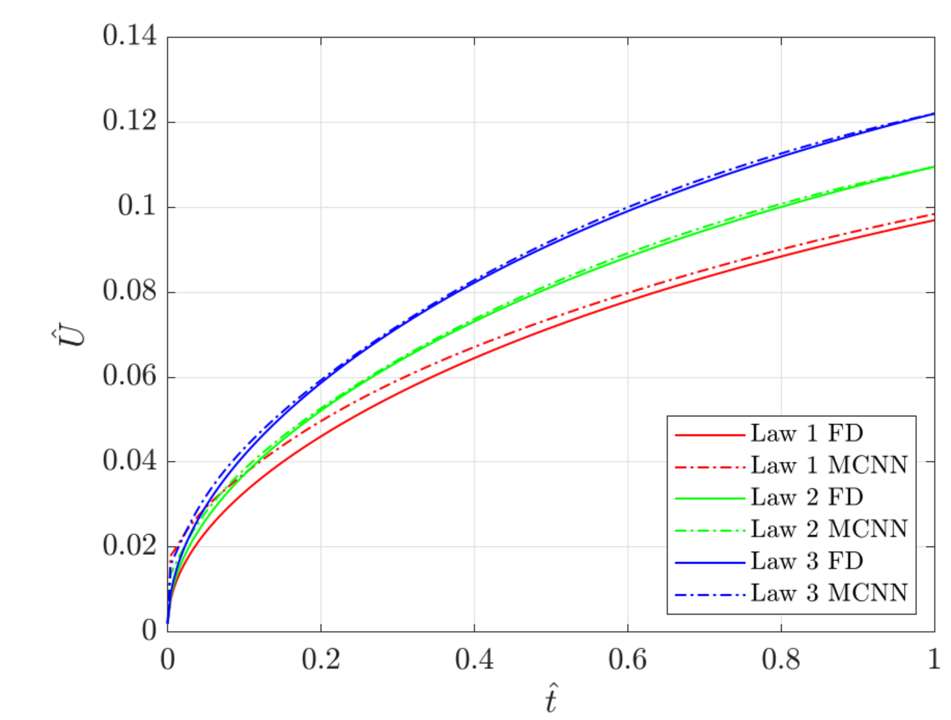

Figure 3: A visualization of MCNN predictions and reference solutions for the dimensionless settlement $\hat{U}$. Predictions from MCNN, even though they are generated by the same model, are very accurate for all three constitutive laws.

Conclusions

- A novel modeling technique called MCNN is proposed to solve the multi-constitutive problems by using a law-encoding vector.
- MCNN could achieve good accuracies for the nonlinear large deformation problem.
- Future work: study encoding vectors of the form (a, b, c) with $a+b+c=1$ and $a, b, c \ge 0$, e.g., (0.2, 0.3, 0.5). This is akin to the idea of fractional derivatives