Solución de la ecuación de onda con aprendizaje automático basado en la física

Cecilia G. López Aceves, Universidad de Guadalajara
Dra. Emilia Fregoso Becerra, Universidad de Guadalajara
Dr. Miguel Angel Moreles Vázquez, Centro de Investigación en Matemáticas, A.C.

UNIVERSIDAD DE GUADALAJARA
Red Universitaria de Jalisco
CUCEI
CIMAT

RESUMEN

Utilizamos redes neuronales físicamente informadas (PINNs) para la resolución de la ecuación de onda en una dimensión utilizando la biblioteca de funciones DeepXDE, y posteriormente se compararon sus resultados con los obtenidos por el método de diferencias finitas.

INTRODUCCIÓN

Modelar un sistema nos proporciona un mejor entendimiento de su comportamiento. En particular, el modelado de ondas es indispensable en geofísica para estimar estructuras en el subsuelo y comprender la estructura interna del planeta.

Las ondas son perturbaciones autosostenidas de un medio, que se mueven a través del espacio transportando energía y están matemáticamente descritas por la **ecuación de onda**,

$$\frac{1}{c^2}\frac{\partial^2 u}{\partial t^2} = \nabla^2 u.$$

Típicamente, la ecuación de onda es resuelta por métodos numéricos como Elemento Finito y Diferencias Finitas que pueden resolver la ecuación con cierto margen de error. Recientemente se cuenta con una nueva herramienta: *las* ***redes neuronales físicamente informadas*** o PINNs por sus siglas en inglés.

Una red neuronal convencional utiliza pares de datos entrada-salida para su entrenamiento. Por el contrario, las PINNs combinan el aprendizaje basado en datos con el conocimiento físico que tengamos del problema. Esto se logra añadiendo la ecuación diferencial junto con las condiciones iniciales y de frontera que definen a un sistema como términos adicionales a la función de pérdida.

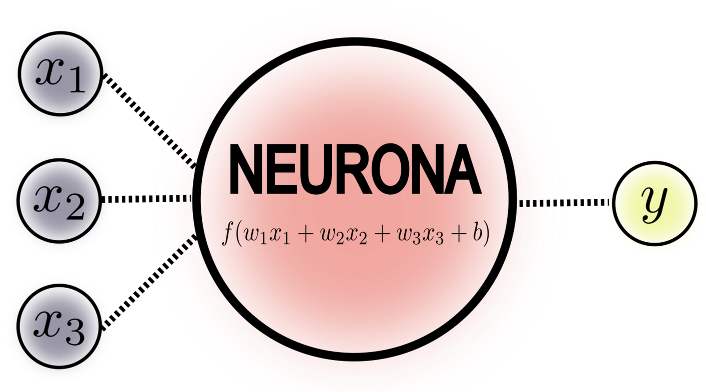


Fig. 1: Función de una neurona

La función de pérdida convencionalmente mide la diferencia entre el valor estimado por la red del valor real dentro de un conjunto de datos de entrenamiento. Como nuestro objetivo es encontrar la solución de la ecuación de onda construyendo una red neuronal que aproxime a esta solución,

$$u(x,t) \approx u_{NN}(x,t;\theta).$$

La **función de pérdida** tiene la siguiente forma,

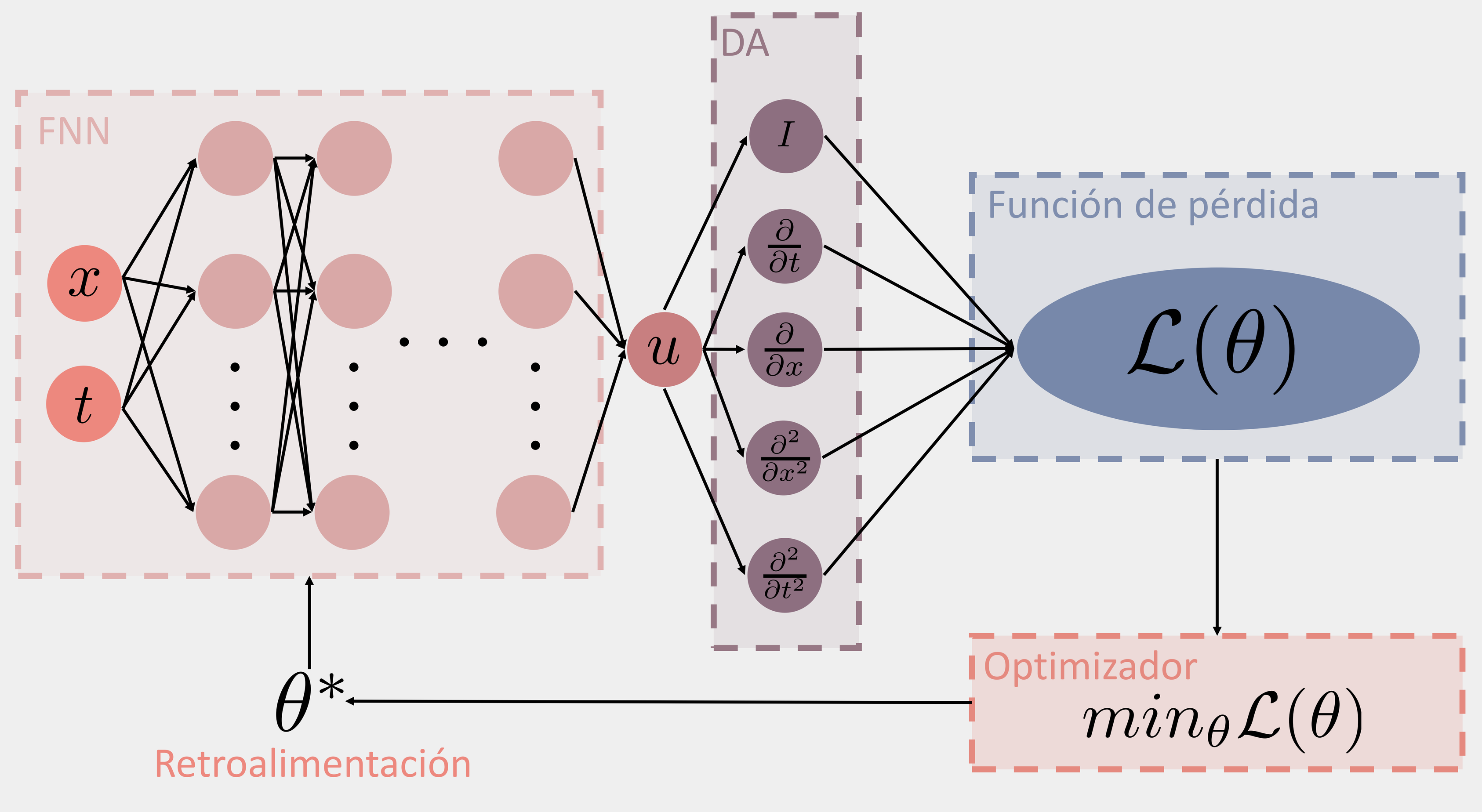
NN

PINN

$$\begin{aligned}\mathcal{L}(\theta) =& \frac{1}{N}\sum_{i=1}^{N} ||u(x_i,t_i) - u_{NN}(x_i,t_i)|| \\ &+ \frac{1}{N_{EDP}} \sum_{\substack{i=1 \\ x_i,t_i \in \Omega}}^{N_{EDP}} ||\nabla^2 u_{NN}(x_i,t_i) - \frac{1}{c^2}\frac{\partial^2 u_{NN}}{\partial t^2}(x_i,t_i)|| \\ &+ \frac{1}{N_{CI}} \sum_{\substack{i=1 \\ x_i \in [0,1]}}^{N_{CI}} \left(||u_{NN}(x_i,0) - u(x_i,0)|| + ||\frac{\partial u_{NN}}{\partial t}(x_i,0) - f_{CI}(u_{NN},x_i,0)|| \right) \\ &+ \frac{1}{N_{CF}} \sum_{\substack{i=1 \\ x_i,t_i \in \partial\Omega}}^{N_{CF}} ||\frac{\partial u_{NN}}{\partial \vec{n}}(x_i,t_i) - f_{CF}(u_{NN},x_i,t_i)||\end{aligned}$$

donde $\theta = \{W^l, b^l\}$ corresponde a los parámetros (pesos y sesgos) de la red. El entrenamiento se basa en estimar estos parámetros para minimizar la función de pérdida.

ESQUEMA DE LA PINN



CONFIGURACIÓN DEL PROBLEMA

El problema consiste en utilizar una red neuronal físicamente informada para solucionar la ecuación de onda no homogénea,

$$\frac{\partial^2 u}{\partial t^2} - c^2\frac{\partial^2 u}{\partial x^2} = g(x,t), \qquad g(x,t) = e^{-3t}\sin x, \qquad t\in[0,1], \qquad x\in[0,1],$$

considerando un pulso gaussiano como condición inicial,

$$u(x,0) = e^{-\left(\frac{x-0.2}{\sigma}\right)^2}, \qquad \sigma = 0.06,$$

con velocidad inicial nula,

$$\frac{\partial u}{\partial t}(x,0) = 0,$$

y condiciones de frontera de Neumann iguales a cero.

Para contar con un conjunto de datos para evaluar las predicciones de la PINN se utilizó el método de diferencias finitas para resolver el problema anterior.

Para escribir el algoritmo de la PINN utilizamos la librería de código abierto **DeepXDE** con el backend de Tensorflow. Se escribió el algoritmo para dos casos:

- **Caso 1**: Con conocimiento de la función $g(x,t)$.
- **Caso 2**: Sin conocimiento de la función $g(x,t)$. La PINN también busca predecir la función $g(x,t)$. Se utilizaron datos de la solución obtenida por el método de diferencias finitas para entrenar a la red.

RESULTADOS

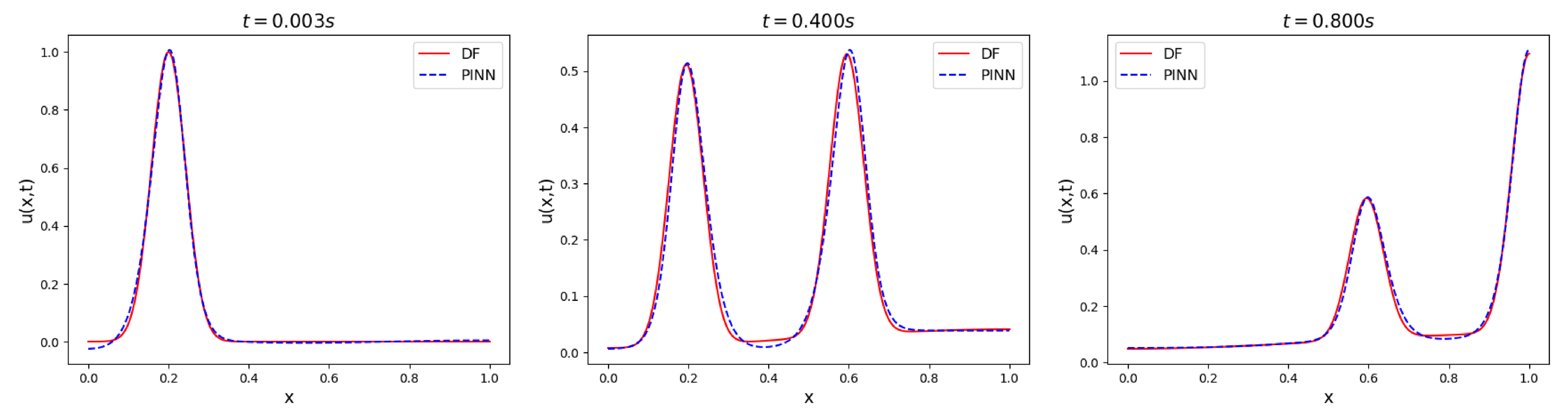
Parámetros de la red:

	N_{EDP}	N_{CI}	N_{CF}	Arquitectura	Función de activación	Inicializador
Caso 1	360	360	360	[2] + [20]*2 + [1]	tanh	Glorot uniform
Caso 2				[2] + [20]*4 + [1]		

En ambos casos después de utilizar el optimizador Adam se usó el optimizador L-BFGS para mejorar la convergencia.

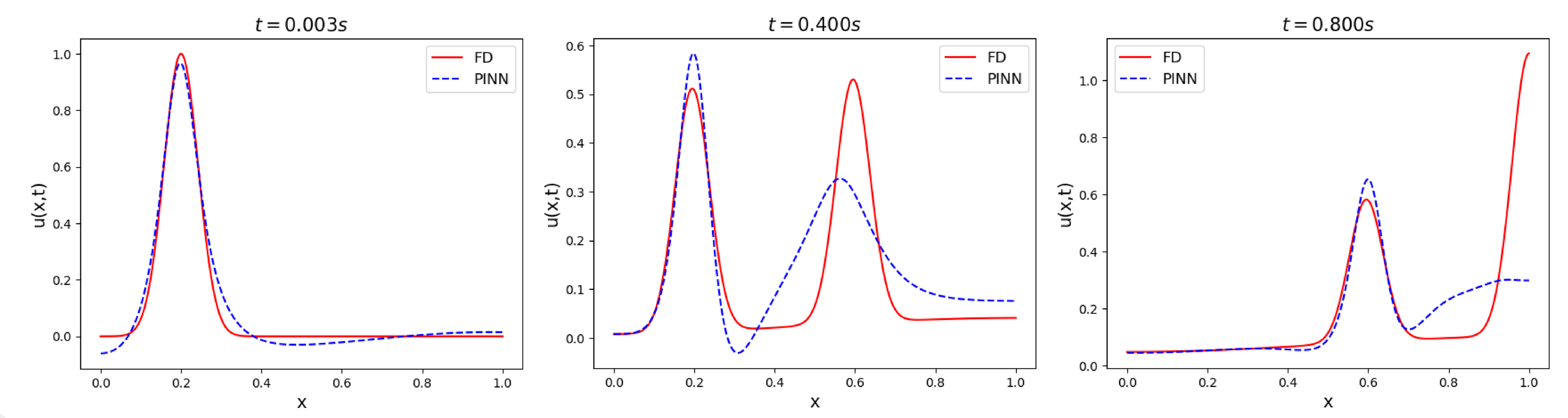
Caso 1

El modelo fue entrenado por 15 000 iteraciones utilizando el optimizador Adam con una taza de aprendizaje establecida en 0.001.



Caso 2

El modelo fue entrenado por 40 000 iteraciones utilizando el optimizador Adam con una taza de aprendizaje establecida en 0.001.



CONCLUSIONES

• Diseñamos el algoritmo de una PINN para resolver la ecuación de onda en una dimensión con resultados cercanos a los obtenidos por métodos numéricos para el caso de un medio homogéneo.

• Esta aproximación no es más eficiente que los métodos numéricos (por el tiempo que toma entrenar la red) pero ofrecen una alternativa libre de mallados y una simplificación de la aplicación de condiciones iniciales y de frontera.

BIBLIOGRAFÍA

[1] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial diffe-rential equations,” Journal of Computational physics, vol. 378, pp. 686–707, 2019.

[2] Lu, L., Meng, X., Mao, Z., & Karniadakis, G. E. (2021). DeepXDE: A deep learning library for solving differential equations. *SIAM review*, *63*(1), 208-228.