

**SOCIEDAD CHILENA DE INGENIERÍA HIDRÁULICA  
XXVI CONGRESO CHILENO DE INGENIERÍA HIDRÁULICA**

**MODELACIÓN DE PROBLEMAS DIRECTOS E INVERSOS EN HIDRODINÁMICA DE  
CANALES MEDIANTE OPERADORES NEURONALES DE FOURIER**

**JONATHAN POBLETE CÁCERES<sup>1</sup>  
YARKO NIÑO CAMPOS<sup>2</sup>  
LUIS ZAMORANO RIQUELME<sup>3</sup>**

**RESUMEN**

Esta investigación evalúa la aplicación de Operadores Neuronales de Fourier (FNO) en problemas directos e inversos de la modelación de flujos en canales. A partir de datos de simulaciones numéricas de Saint Venant 2D para tres geometrías de lecho, se optimizaron arquitecturas FNO para doce escenarios numéricos experimentales. Un hallazgo importante fue la superioridad de un enfoque puramente basado en datos; la inclusión de una función de pérdida informada por la física no mejoró el rendimiento, lo que sugiere que los FNO aprenden implícitamente la física subyacente cuando se dispone de datos suficientes. Los modelos demostraron una capacidad excepcional para resolver problemas inversos dimensionales, reconstruyendo con alta fidelidad la batimetría del lecho y los parámetros hidráulicos ( $R^2 > 0.98$ ) a partir de los campos de velocidad, independientemente de la complejidad geométrica. Sin embargo, se identificaron dos desafíos principales en geometrías complejas: la predicción de componentes de flujo secundarios, como la velocidad transversal, resultó ser difícil y la adimensionalización obstaculizó significativamente la resolución de problemas inversos. El estudio también confirma que la arquitectura FNO óptima es altamente específica para cada tarea, lo que pone énfasis en la necesidad de un ajuste de hiperparámetros automatizado para maximizar el rendimiento.

---

<sup>1</sup> Estudiante de Magíster, Departamento de Ingeniería Civil, Universidad de Chile - email: jonathan.poblete@ing.uchile.cl

<sup>2</sup> Profesor Titular, Departamento de Ingeniería Civil, Advanced Mining Technology Center (AMTC), Universidad de Chile - email: ynino@uchile.cl

<sup>3</sup> Jefe de Ingeniería y Desarrollo, Instituto Nacional de Hidráulica - email:luiszamorano@inh.cl

# 1 INTRODUCCIÓN

La integración del aprendizaje automático (ML) en la dinámica de fluidos computacional (CFD) ofrece un enorme potencial para acelerar simulaciones y resolver problemas complejos, evitando el alto coste computacional de los métodos numéricos tradicionales (Brunton et al. 2019; Vinuesa et al. 2022). Entre las arquitecturas de ML, los Operadores Neuronales de Fourier (FNO) han emergido como una herramienta especialmente prometedora para resolver ecuaciones en derivadas parciales (EDP), demostrando una capacidad superior para capturar dependencias a múltiples escalas y generalizar entre distintas geometrías y resoluciones (Li et al. 2021).

Este estudio investiga la aplicación de los FNO al la modelación numérica Saint Venant 2D de flujos en canales, un problema fundamental en hidrodinámica con aplicaciones en áreas como gestión ambiental y morfología fluvial. Se aborda un conjunto completo de problemas directos (predecir el flujo a partir de condiciones iniciales conocidas) e inversos (inferir parámetros del sistema, como la batimetría, a partir de observaciones del flujo). El objetivo principal es evaluar el rendimiento de los FNO en doce escenarios numéricos experimentales, combinando tres geometrías de lecho de complejidad variable (SLOPEa, NOISEa, BARSa) con formulaciones dimensionales y adimensionales.

Una cuestión central en la aplicación de ML a la física es el equilibrio entre el aprendizaje puramente basado en datos y el aprendizaje informado por la física (PIML), que integra las ecuaciones rectoras en la función de pérdida (Raissi et al. 2019; Karniadakis et al. 2021). Este trabajo evalúa ambos enfoques para determinar las condiciones en las que la regularización física explícita es beneficiosa. Mediante una optimización de hiperparámetros y un análisis comparativo, este estudio busca delinear las fortalezas y debilidades de los FNO, evaluando su robustez en problemas inversos, su capacidad para capturar componentes de flujo complejos y el impacto de la adimensionalización en el rendimiento del modelo.

# 2 MATERIALES Y MÉTODOS

Los datos para este estudio se generaron mediante TELEMAC2D (v8p4), que resuelve las ecuaciones de aguas someras (EAS) promediadas en profundidad o las ecuaciones Saint Venant 2D. El dominio computacional se basa en un canal de laboratorio de 12 m de longitud y 0.3 m de ancho, discretizado con una malla triangular no estructurada con un espaciado de 3 cm.

Las ecuaciones EAS que resuelve TELEMAC2D, y que forman la base de este estudio, son:

$$\frac{\partial h}{\partial t} + \vec{u} \cdot \vec{\nabla}(h) + h \operatorname{div}(\vec{u}) = S_h \quad (1)$$

$$\frac{\partial u}{\partial t} + \vec{u} \cdot \vec{\nabla}(u) = -g \frac{\partial Z}{\partial x} + S_x + \frac{1}{h} \operatorname{div}(h v_t \vec{\nabla} u) \quad (2)$$

$$\frac{\partial v}{\partial t} + \vec{u} \cdot \vec{\nabla}(v) = -g \frac{\partial Z}{\partial y} + S_y + \frac{1}{h} \operatorname{div}(h v_t \vec{\nabla} v) \quad (3)$$

donde  $h$  es la profundidad del agua,  $\vec{u} = (u, v)$  es el vector de velocidad,  $Z$  es la elevación de la superficie libre ( $Z = h + B$ ),  $g$  es la gravedad, y  $\nu_t$  es la viscosidad turbulenta. Los términos  $S_h$ ,  $S_x$ , y  $S_y$  representan las fuentes y sumideros, que para este estudio de flujo en estado estacionario incluyen las fuerzas de fricción del lecho (modeladas con la ley de Manning) y se consideran cero para las fuentes de masa.

Se consideraron tres geometrías de lecho de complejidad creciente: SLOPEa (pendiente lisa), NOISEa (ruido aleatorio) y BARSa (barras alternas). Para generar los datos de este estudio, se muestrearon directamente los parámetros adimensionales (e.g., Froude, Reynolds, etc.) dentro de los rangos de la Tabla 2 utilizando un muestreo de hipercubo latino (LHS), un método estadístico que garantiza una cobertura eficiente del espacio de parámetros (Tang 1993). A partir de este conjunto adimensional, se calcularon los parámetros dimensionales requeridos para las simulaciones. Notablemente, la viscosidad turbulenta ( $\nu_t$ ) para cada caso se derivó directamente del número de Reynolds turbulento ( $Re_t$ ) muestreado. El muestreo se diseñó para generar una mezcla equilibrada de regímenes de flujo tanto subcríticos ( $Fr < 1$ ) como supercríticos ( $Fr > 1$ ), resultando en 12 482 simulaciones únicas para cada geometría. Aunque también se exploró una estrategia de muestreo dimensional (ver Tabla 1), los resultados presentados en este trabajo se basan en los conjuntos de datos generados a partir del muestreo adimensional.

Tabla 1: Límites de los parámetros de simulación para el muestreo dimensional.

Parámetro	Rango	Descripción
Pendiente, $S$	$[3 \times 10^{-6}, 0.1]$	Pendientes fluviales naturales (Cohen et al. 2018).
Coef. de Manning, $n$ [ $s\ m^{-1/3}$ ]	$[0.01, 0.2]$	Valores tabulados para distintos materiales.
Caudal, $Q_0$ [ $m^3\ s^{-1}$ ]	$[0.005, 0.02]$	Condiciones típicas de laboratorio.
Profundidad del agua, $H_0$ [m]	$[0.01, 0.25]$	Consistente con Configuraciones experimentales.

Tabla 2: Rangos de los parámetros adimensionales para el muestreo.

Parámetro	Límite inferior	Límite superior
Relación de aspecto del canal, $Ar$	1.0	40.0
Relación de alturas, $Hr$	0.001	30.0
Número de Froude, $Fr$	0.001	10.0
Parámetro de fricción adimensional, $M$	0.001	500.0
Número de Reynolds turbulento, $Re_t$	100.0	75 000.0

A partir de estos datos, se definieron cuatro tipos de problemas experimentales, Directo Dimensional (DD), Directo Adimensional (DA), Inverso Dimensional (ID) e Inverso Adimensional (IA), cada uno aplicado a las tres geometrías. Los problemas directos consisten en predecir los campos de flujo resultantes ( $H$ ,  $U$ ,  $V$  o sus versiones adimensionales  $H^*$ ,  $U^*$ ,  $V^*$ ) a partir de un conjunto de parámetros de entrada y la geometría del lecho conocida. Por otro lado, los problemas inversos requieren que el modelo infiera un conjunto de parámetros escalares y la geometría del lecho ( $B$  o  $B^*$ ) a partir de los campos de velocidad observados ( $U$ ,  $V$  o  $U^*$ ,  $V^*$ ). Cada tipo de problema se

implementó tanto en su versión dimensional como adimensional, generando así los cuatro escenarios experimentales aplicados a cada una de las tres geometrías de lecho. Las variables de campo adimensionales ( $H^*$ ,  $U^*$ ,  $V^*$ ,  $B^*$ ) se obtienen normalizando sus contrapartes dimensionales por sus escalas características (e.g.,  $H^* = H/h_c$ ), mientras que los parámetros adimensionales inferidos en los problemas inversos incluyen el número de Froude ( $Fr = u_c/\sqrt{gh_c}$ ), el de Reynolds turbulento ( $Re_t = u_c x_c/\nu_t$ ), la relación de aspecto ( $Ar = x_c/y_c$ , que por diseño es igual a la relación de velocidades  $Vr$ ), la relación de alturas ( $Hr = b_c/h_c$ ), y un parámetro de fricción adimensional derivado del término de Manning ( $M = gn^2 x_c/h_c^{4/3}$ ).

El modelo sustituto se basa en una arquitectura de Operador Neuronal de Fourier (FNO) (Li et al. 2021), implementada en PyTorch y esquematizada en la Figura 1. La arquitectura procesa tanto campos espaciales como parámetros escalares. Primero, las entradas se combinan en un único tensor. Luego, un núcleo FNO, compuesto por capas de convolución espectral, procesa el tensor para aprender dependencias a múltiples escalas. Se emplean conexiones residuales (He et al. 2016) para estabilizar el entrenamiento; esta técnica consiste en sumar la entrada de una capa a su salida, mejorando el flujo de gradientes y mitigando el problema de su desvanecimiento, lo que permite entrenar redes más profundas y complejas de manera efectiva. Finalmente, cabezales de salida independientes proyectan la representación interna para generar las predicciones de campo y escalares.

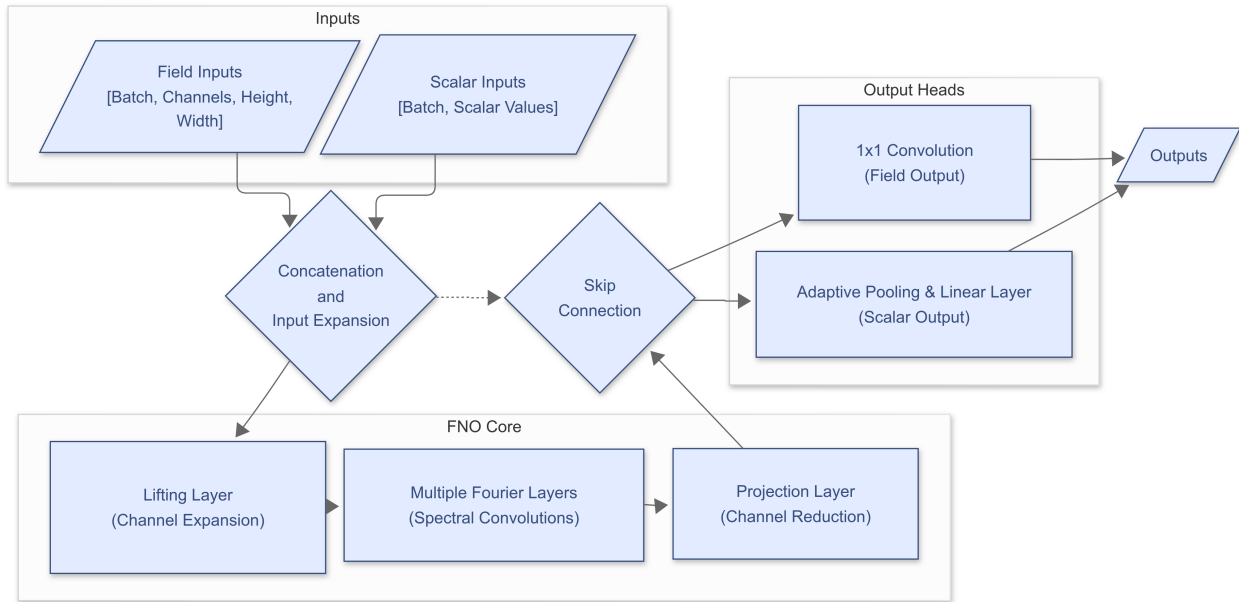


Figura 1: Arquitectura de la red neuronal basada en Operador Neuronal de Fourier (FNO). La red procesa campos espaciales 2D y parámetros escalares a través de cuatro etapas principales: procesamiento de entrada que combina campos espaciales y parámetros escalares expandidos, núcleo FNO que realiza convoluciones espectrales en el dominio de Fourier, conexión residual para mejorar la estabilidad del entrenamiento, y procesamiento de salida mediante cabezales separados para generar predicciones de campos y escalares.

La función de pérdida,  $\mathcal{L}_{total}$ , combina un componente basado en datos ( $\mathcal{L}_{data}$ ) con un componente opcional informado por la física ( $\mathcal{L}_{phys}$ ).  $\mathcal{L}_{data}$  mide la discrepancia con los datos de referencia utilizando la pérdida de Huber, elegida por su robustez frente a valores atípicos. Esta función combina

las propiedades de la pérdida cuadrática (MSE) para errores pequeños y la pérdida lineal (MAE) para errores grandes, lo que reduce su sensibilidad a valores atípicos. Se define como:

$$L_{\delta}(a) = \begin{cases} \frac{1}{2}a^2 & \text{para } |a| \leq \delta, \\ \delta(|a| - \frac{1}{2}\delta) & \text{en otro caso,} \end{cases} \quad (4)$$

donde  $a$  es el residuo (la diferencia entre la predicción y el valor objetivo) y  $\delta$  es un umbral que define la transición entre el régimen cuadrático y el lineal.  $\mathcal{L}_{\text{phys}}$  penaliza los residuos de las ecuaciones de continuidad y momento, dadas por (1), (2) y (3). Cada término de pérdida física ( $\mathcal{L}_{\text{cont}}$ ,  $\mathcal{L}_x$ ,  $\mathcal{L}_y$ ) corresponde a la pérdida de Huber del residuo respectivo con respecto a cero. La pérdida total es:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{data}} \mathcal{L}_{\text{data}} + \mathcal{L}_{\text{phys}} = \lambda_{\text{data}} \mathcal{L}_{\text{data}} + (\lambda_{\text{cont}} \mathcal{L}_{\text{cont}} + \lambda_x \mathcal{L}_x + \lambda_y \mathcal{L}_y). \quad (5)$$

Para equilibrar la contribución de cada componente y evitar que uno domine sobre los demás, las ponderaciones  $\lambda$  se gestionaron dinámicamente con el algoritmo ReLoBRaLo (Bischof et al. 2025), un método que equilibra los diferentes componentes de la pérdida al priorizar dinámicamente los términos que el modelo encuentra más difíciles de minimizar.

El entrenamiento se realizó con el optimizador AdamW (Loshchilov et al. 2019) y un programador de tasa de aprendizaje de un ciclo (OneCycleLR). Se utilizó un algoritmo de detención temprana para evitar el sobreajuste. Los hiperparámetros de cada uno de los doce escenarios se optimizaron individualmente mediante Optuna (Akiba et al. 2019) con un muestreador TPE. El espacio de búsqueda completo se detalla en la Tabla 3. Todo el entrenamiento y la optimización se realizaron en una GPU NVIDIA RTX 4090.

Tabla 3: Espacio de búsqueda de hiperparámetros para la optimización con Optuna.

Hiperparámetro	Tipo	Rango/Opciones
Tasa de aprendizaje	Log-Float	$[1 \times 10^{-6}, 1 \times 10^{-3}]$
Decaimiento de peso	Float	$[0, 0.005]$
Tamaño de lote	Categorico	{8, 16, ..., 1024}
Capas FNO	Entero	[8, 16]
Modos en $x / y$	Entero	$[1, N_x] / [1, N_y]$
Canales (ocultos, proyección)	Entero	[1, 256]
Usar pérdida física	Categorico	{true, false}
Normalizar salida	Categorico	{true, false}

Para evaluar la transferencia de conocimiento entre las distintas geometrías, se realizó un ajuste fino (*fine-tuning*) de modelos ya entrenados. El proceso consistió en tomar un modelo que ya había sido entrenado hasta la convergencia en un conjunto de datos de origen (e.g., SLOPEa) con sus hiperparámetros óptimos. A continuación, los pesos guardados de este modelo base se utilizaron para inicializar un nuevo modelo. Este segundo modelo fue entonces reentrenado sobre un conjunto de datos de destino diferente (e.g., BARSa), permitiendo la actualización de todos sus parámetros. El procedimiento de reentrenamiento (optimizador, tasa de aprendizaje, etc.) se mantuvo idéntico al utilizado para entrenar los modelos desde cero, para así asegurar una comparación justa.

### 3 RESULTADOS

Esta sección detalla el rendimiento de los modelos FNO en los doce escenarios experimentales previamente definidos. El objetivo del estudio es evaluar la capacidad del FNO para actuar como un sustituto del modelo numérico; por lo tanto, todos los valores esperados en las métricas y figuras corresponden a los resultados de las simulaciones de TELEMAC2D. Una conclusión principal y consistente en todos los casos fue que el aprendizaje puramente basado en datos superó sistemáticamente al enfoque informado por la física. La optimización de hiperparámetros nunca seleccionó el componente de pérdida basado en residuos de las EDP, lo que sugiere que, con datos de simulación suficientemente ricos, los FNO aprenden implícitamente las restricciones físicas.

Además, el proceso de optimización reveló que no existe una arquitectura FNO universalmente óptima. Como se muestra en la Tabla 4, la configuración ideal (número de capas, modos de Fourier, etc.) varió según la tarea (directa, inversa, dimensional o adimensional), lo que implica la necesidad de un ajuste de hiperparámetros a medida para maximizar el rendimiento.

Tabla 4: Hiperparámetros de entrenamiento y arquitectura optimizados para los 12 casos de estudio. La tabla muestra la configuración del modelo con mejor rendimiento para cada escenario, destacando la alta especificidad de la tarea. (DD: Directo Dimensional, ID: Inverso Dimensional, DA: Directo Adimensional, IA: Inverso Adimensional. Sufijos B, S, N se refieren a BARSa, SLOPEa, NOISEa).

Parámetro	DD-B	ID-B	DA-B	IA-B	DD-S	ID-S	DA-S	IA-S	DD-N	ID-N	DA-N	IA-N
Tamaño Lote (Entren.)	32	8	64	16	8	16	8	16	8	8	16	16
Tasa Aprend. ( $10^{-4}$ )	2.78	7.38	0.51	5.01	2.37	7.33	2.16	4.99	8.14	4.23	0.39	2.21
Capas FNO	16	1	11	15	9	9	4	5	16	2	4	8
Modos (x/y)	392/4	384/2	180/4	212/4	382/2	384/8	274/4	372/2	38/4	2/10	224/8	156/6
Canales Ocultos	79	111	20	103	57	31	89	149	170	149	92	154
Salida Norm.	true	true	true	false	false	false	true	true	true	true	true	false

En los problemas directos, los modelos FNO predijeron con alta fidelidad las variables primarias, profundidad del agua ( $H$ ) y velocidad longitudinal ( $U$ ), pero tuvieron dificultades con la velocidad transversal ( $V$ ) en geometrías complejas.

Como se detalla en la Tabla 5, los modelos lograron una alta precisión para  $H$  y  $U$  en todas las geometrías. Para BARSa y NOISEa, los valores de  $R^2$  para  $H$  superaron el 0.97 y para  $U$  el 0.88. Sin embargo, la predicción de  $V$  fue deficiente en estos casos complejos ( $R^2 < 0.05$ ). En contraste, para la geometría simple SLOPEa, el modelo predijo  $V$  con una alta precisión ( $R^2 = 0.945$ ). Este patrón, visualizado en la Figura 2 para el caso BARSa, sugiere que la capacidad del FNO para capturar componentes de flujo secundarios es sensible a la complejidad del lecho.

Al emplear entradas y salidas adimensionales, los modelos mostraron tendencias similares. Como se presenta en la Tabla 6, la predicción de  $H^*$  y  $U^*$  fue generalmente acertada en todas las geometrías. Sin embargo, la predicción de  $V^*$  solo fue exitosa para el caso simple SLOPEa ( $R^2 = 0.867$ ), fallando nuevamente en las geometrías más complejas BARSa y NOISEa.

El rendimiento de los FNO en los problemas inversos dimensionales fue alto. Como se detalla en la

Tabla 5: Métricas de rendimiento para los modelos Directos Dimensionales (DD) en los conjuntos de prueba.

Variable	BARSa		SLOPEa		NOISEa	
	MAE	$R^2$	MAE	$R^2$	MAE	$R^2$
H [m]	0.0246	0.972	0.0736	0.979	0.0263	0.974
U [ $\text{m s}^{-1}$ ]	0.0372	0.947	0.0730	0.908	0.0473	0.885
V [ $\text{m s}^{-1}$ ]	0.0233	0.006	0.0740	0.945	0.0065	0.045

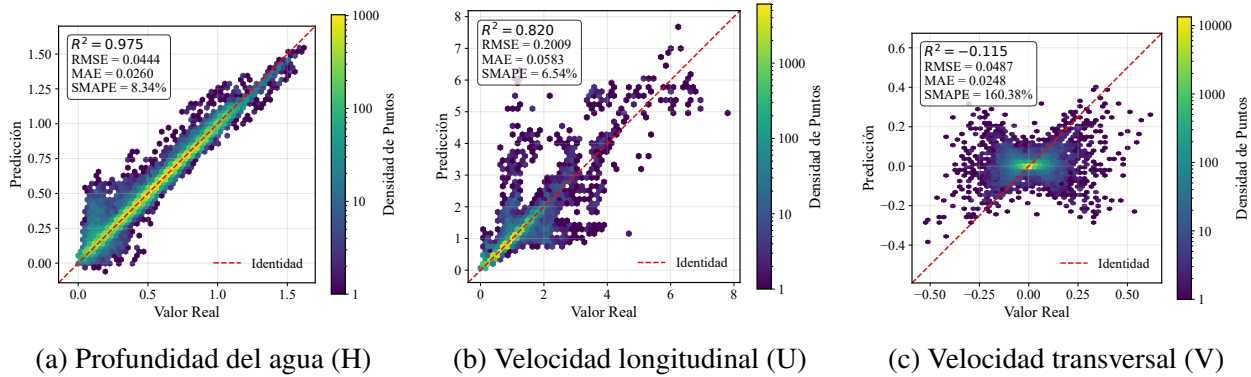


Figura 2: Resultados del modelo Directo Dimensional para la geometría BARSa, comparando las predicciones del FNO con los datos de referencia. (a) Predicción exitosa de la profundidad del agua. (b) Predicción exitosa de la velocidad longitudinal. (c) Predicción deficiente de la velocidad transversal, donde el modelo no logra capturar la estructura del campo de flujo.

Tabla 6: Métricas de rendimiento para los modelos Directos Adimensionales (DA) en los conjuntos de prueba. Se reporta el Error Absoluto Medio (MAE) y el Coeficiente de Determinación ( $R^2$ ).

Variable	BARSa		SLOPEa		NOISEa	
	MAE	$R^2$	MAE	$R^2$	MAE	$R^2$
$H^*$ [-]	0.3392	0.961	0.1548	0.983	0.2843	0.971
$U^*$ [-]	0.0116	0.818	0.0071	0.858	0.0123	0.817
$V^*$ [-]	0.0056	0.010	0.0003	0.867	0.0024	0.134

Tabla 7, los modelos reconstruyeron con una alta fidelidad no solo los parámetros escalares ( $Q_0$ ,  $n$ ,  $\nu_t$ ), sino también los campos espaciales de la batimetría ( $B$ ) y la profundidad del agua ( $H$ ). Es de destacar que este alto rendimiento ( $R^2 > 0.97$  para casi todas las variables) se mantuvo en las tres geometrías, incluidas las más complejas. La Figura 3 muestra la alta calidad de la reconstrucción para el caso BARSa. Estos resultados posicionan a los FNO como una herramienta prometedora para aplicaciones prácticas como la inferencia batimétrica, es decir, la capacidad de estimar la topografía del lecho de un río a partir de mediciones de la velocidad en la superficie.

En contraste con el caso dimensional, la formulación inversa adimensional presentó dificultades significativas. Como se detalla en la Tabla 8, si bien los modelos infirieron correctamente los pará-

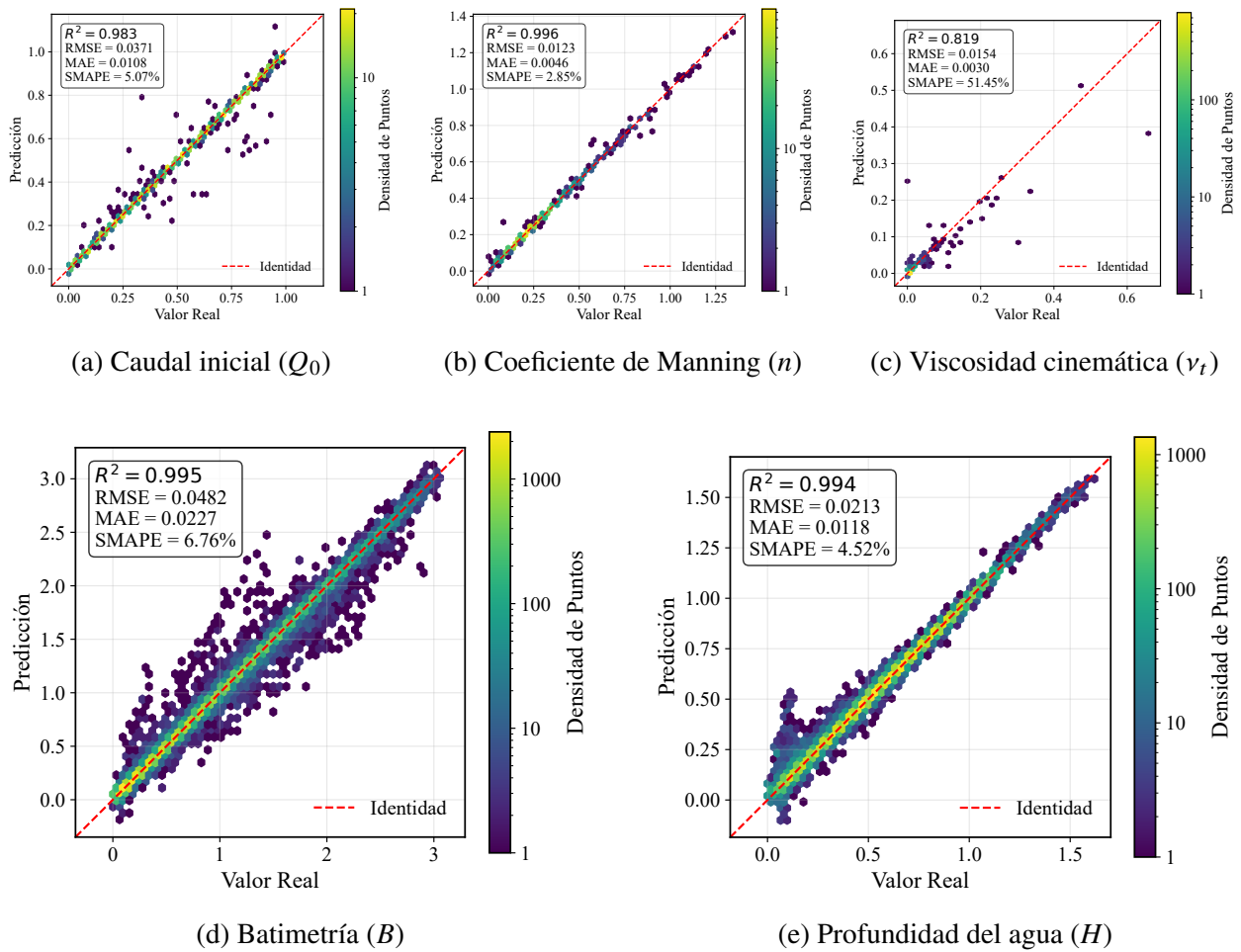


Figura 3: Resultados del modelo Inverso Dimensional para la geometría BARSa. La figura muestra la alta fidelidad con la que se reconstruyen los parámetros escalares desconocidos (a) Caudal inicial  $Q_0$ , (b) Coeficiente de Manning  $n$ , (c) Viscosidad cinemática  $\nu_t$ , y los campos espaciales (d) batimetría  $B$  y (e) profundidad del agua  $H$  a partir de los campos de velocidad de entrada.

Tabla 7: Métricas de rendimiento para los modelos Inversos Dimensionales (ID). Se infieren ( $Q_0$ ,  $n$ ,  $\nu_t$ ,  $B$ ,  $H$ ) a partir de ( $U$ ,  $V$ ). Se reporta el Error Absoluto Medio (MAE) y el Coeficiente de Determinación ( $R^2$ ).

Variable	BARSa		SLOPEa		NOISEa	
	MAE	$R^2$	MAE	$R^2$	MAE	$R^2$
$Q_0$ [ $\text{m}^3 \text{s}^{-1}$ ]	0.0080	0.991	0.0065	0.999	0.0189	0.979
$n$ [ $\text{s m}^{-1/3}$ ]	0.0040	0.996	0.0069	1.000	0.0102	0.987
$\nu_t$ [ $\text{m}^2 \text{s}^{-1}$ ]	0.0021	0.984	0.0069	0.999	0.0020	0.992
$B$ [m]	0.0214	0.993	0.0101	0.994	0.0293	0.990
$H$ [m]	0.0106	1.000	0.0080	1.000	0.0184	1.000

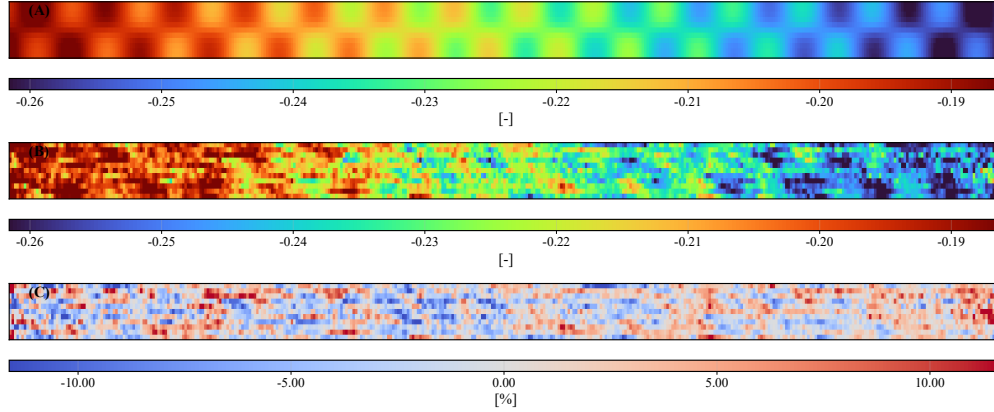
metros escalares adimensionales y la profundidad del agua ( $H^*$ ), la reconstrucción de la batimetría adimensional ( $B^*$ ) fracasó en las geometrías complejas BARSa y NOISEa. Únicamente en el caso simple SLOPEa se logró una reconstrucción de alta fidelidad de  $B^*$  ( $R^2 = 0.988$ ). Esto, visualizado en la Figura 4, sugiere que la adimensionalización puede ocultar información geométrica importante para la inversión en topografías complejas.

Tabla 8: Métricas de rendimiento para los modelos Inversos Adimensionales (IA). Se infieren ( $H^*$ ,  $B^*$ , etc.) a partir de ( $U^*$ ,  $V^*$ ). Se reporta el Error Absoluto Medio (MAE) y el Coeficiente de Determinación ( $R^2$ ).

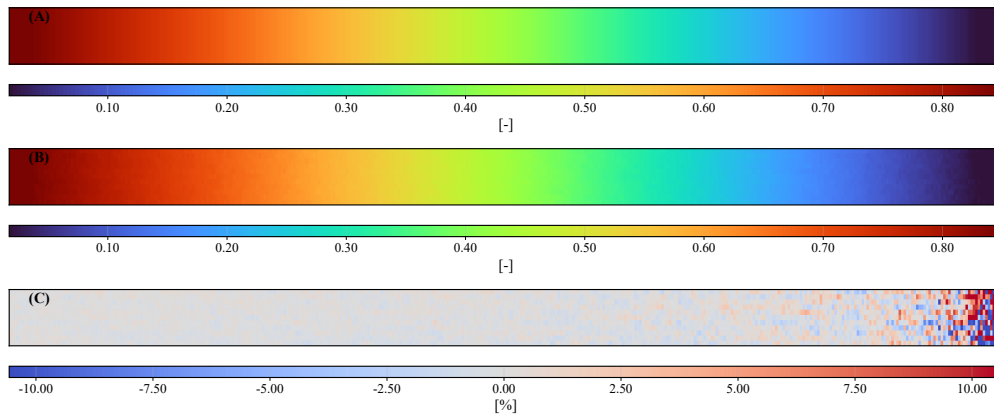
Variable	BARSa		SLOPEa		NOISEa	
	MAE	$R^2$	MAE	$R^2$	MAE	$R^2$
$H^*$ [-]	0.0273	0.997	0.0178	0.999	0.0713	0.981
$B^*$ [-]	0.0539	0.306	0.0076	0.988	0.0517	0.459
$Fr$ [-]	0.0310	0.997	0.0254	0.997	0.0579	0.990
$Hr$ [-]	0.0567	0.991	0.1556	0.994	0.1322	0.948
$Re_t$ [-]	0.0895	0.913	324.29	0.991	0.2073	0.787
$M$ [-]	0.0918	0.969	1.6260	0.996	0.1362	0.958

Se realizaron experimentos de aprendizaje por transferencia para evaluar si el conocimiento de una geometría podía mejorar el entrenamiento en otra. Como se muestra en la Tabla 9, el ajuste fino redujo sistemáticamente el número de épocas necesarias para la convergencia, pero no garantizó una mayor precisión. La transferencia de geometrías complejas a simples (e.g., NOISEa  $\rightarrow$  SLOPEa) a menudo tuvo éxito, mientras que la transferencia de simples a complejas (e.g., SLOPEa  $\rightarrow$  BARSa) resultó con frecuencia en un rendimiento inferior al de un modelo entrenado desde cero. Esto indica que el aprendizaje por transferencia no sustituye a un entrenamiento dedicado para lograr la máxima precisión en tareas complejas.

Finalmente, en términos de eficiencia computacional, los modelos FNO ofrecen una aceleración drástica frente al modelo numérico. Mientras que una simulación en TELEMAC2D requirió en promedio entre 1 y 5 minutos por caso para alcanzar la convergencia en una CPU de 4 núcleos, el modelo FNO entrenado es capaz de realizar una predicción (inferencia) entre 0.3ms y 3ms en una



(a) Geometría compleja (BARSa): Fallo en la reconstrucción de  $B^*$ .



(b) Geometría simple (SLOPEa): Reconstrucción exitosa de  $B^*$ .

Figura 4: Contraste en la reconstrucción de la batimetría adimensional ( $B^*$ ). Mientras que para la geometría compleja BARSa (a) la reconstrucción de  $B^*$  a partir de los campos de velocidad falla, para la geometría simple SLOPEa (b) el error es bajo.

GPU NVIDIA RTX 4090, lo que representa una aceleración de más de cinco órdenes de magnitud. Si bien el entrenamiento de un modelo, incluyendo la optimización de hiperparámetros y la validación cruzada, es un proceso que requiere una inversión computacional inicial significativa (del orden de 4-6 horas), una vez entrenado, el modelo sustituto permite la exploración de escenarios y la resolución de problemas inversos casi en tiempo real.

## 4 DISCUSIÓN

Un hallazgo principal y consistente fue que el componente de pérdida basado en la física ( $\mathcal{L}_{\text{phys}}$ ) nunca fue seleccionado por el optimizador. Este resultado, si bien puede parecer contraintuitivo frente a estudios que demuestran los beneficios de PIML (Raissi et al. 2019; Karniadakis et al. 2021), se alinea con una creciente evidencia que indica que, en regímenes de datos abundantes,

Tabla 9: Rendimiento del aprendizaje por transferencia. Se compara el rendimiento ( $R^2$ ) y la eficiencia (épocas) de los modelos ajustados por transferencia frente a los entrenados desde cero en el conjunto de datos de destino.

Fuente (Tipo y $R^2$ )	Destino	Modelo por Transferencia		Entrenamiento desde Cero	
		$R^2_{val}$	Épocas	$R^2_{val}$	Épocas
<b>Fuente: NOISEa</b>					
IA-NOISEa (0.920)	SLOPEa	0.990	511	<b>0.991</b>	511
IA-NOISEa (0.920)	BARSa	0.956	144	<b>0.973</b>	331
DA-NOISEa (0.971)	SLOPEa	0.978	250	<b>0.982</b>	385
DA-NOISEa (0.971)	BARSa	<b>0.963</b>	180	0.961	511
ID-NOISEa (0.978)	SLOPEa	0.998	502	<b>0.999</b>	488
ID-NOISEa (0.978)	BARSa	0.992	205	0.992	511
DD-NOISEa (0.909)	SLOPEa	0.850	95	<b>0.940</b>	167
DD-NOISEa (0.909)	BARSa	0.787	80	<b>0.946</b>	183
<b>Fuente: SLOPEa</b>					
IA-SLOPEa (0.991)	NOISEa	0.468	292	<b>0.920</b>	222
IA-SLOPEa (0.991)	BARSa	0.589	335	<b>0.973</b>	331
DA-SLOPEa (0.982)	NOISEa	<b>0.973</b>	98	0.971	205
DA-SLOPEa (0.982)	BARSa	<b>0.965</b>	103	0.961	511
ID-SLOPEa (0.999)	NOISEa	0.926	411	<b>0.978</b>	366
ID-SLOPEa (0.999)	BARSa	0.908	295	<b>0.992</b>	511
DD-SLOPEa (0.940)	NOISEa	0.599	120	<b>0.909</b>	148
DD-SLOPEa (0.940)	BARSa	0.598	123	<b>0.946</b>	183
<b>Fuente: BARSa</b>					
IA-BARSa (0.973)	NOISEa	<b>0.958</b>	151	0.920	222
IA-BARSa (0.973)	SLOPEa	0.991	306	0.991	511
DA-BARSa (0.961)	NOISEa	<b>0.974</b>	471	0.971	205
DA-BARSa (0.961)	SLOPEa	0.979	453	<b>0.982</b>	385
ID-BARSa (0.992)	NOISEa	0.969	191	<b>0.978</b>	366
ID-BARSa (0.992)	SLOPEa	0.990	343	<b>0.999</b>	488
DD-BARSa (0.946)	NOISEa	0.892	81	<b>0.909</b>	148
DD-BARSa (0.946)	SLOPEa	0.851	103	<b>0.940</b>	167

Nota: La columna 'Fuente' indica el modelo preentrenado y su  $R^2$  en su conjunto de datos original. El mejor resultado de  $R^2$  entre el modelo por transferencia y el entrenado desde cero se resalta en **negrita**.

los modelos pueden aprender la física subyacente de forma implícita (Brunton et al. 2019; Li et al. 2021). Con un conjunto de datos de más de 12 000 simulaciones por geometría, es plausible que la capacidad de interpolación del FNO fuera suficiente para aproximar eficazmente las soluciones, sin que la regularización explícita aportara una ventaja medible.

Un segundo hallazgo se relaciona con la dificultad del modelo para capturar componentes de flujo de baja magnitud. En los problemas directos dimensionales (DD), la magnitud de la velocidad longitudinal ( $U$ ) es predominantemente mayor que la de la transversal ( $V$ ). Esto introduce un sesgo en la función de pérdida: al minimizar el error agregado, el modelo prioriza la predicción de  $U$ ,

explicando el bajo rendimiento en la predicción de  $V$  para los casos BARSa y NOISEa. Este problema se exagera con la adimensionalización. Por ejemplo, para la geometría BARSa, el MAE para la profundidad del agua ( $H$ ) en el modelo DD fue de 0.0246 m. En contraste, el MAE del modelo DA para  $H^*$  fue de 0.3392; al re-escalar este error utilizando una altura característica representativa del rango de muestreo ( $h_c = 0.1$  m), este equivale a un error dimensional de 0.034 m, que es notablemente superior. Esto sugiere que al escalar  $V$  con una velocidad característica longitudinal ( $u_c$ ) mucho mayor, su señal informativa se degrada, afectando la predicción de otras variables que dependen de ella.

Este mismo efecto explica el drástico contraste observado en los problemas inversos. El éxito del caso dimensional (ID), donde la batimetría se reconstruyó con una alta fidelidad ( $R^2 > 0.98$ ), demuestra que los campos de velocidad dimensionales contienen la información necesaria y que el FNO es capaz de extraerla. En cambio, el fracaso del caso adimensional (IA) en geometrías complejas sugiere que el proceso de adimensionalización, al comprimir la señal de las componentes de flujo secundarias, enmascara las variaciones en los campos de velocidad que son cruciales para inferir una topografía no trivial.

La optimización de hiperparámetros y los experimentos de aprendizaje por transferencia refuerzan la idea de que no existe una arquitectura FNO universalmente óptima. La variabilidad en las configuraciones ideales (Tabla 4) demuestra que cada problema requiere un diseño específico. Esto se conecta directamente con los resultados de la transferencia de aprendizaje: la pérdida de rendimiento observada al pasar de geometrías simples a complejas sugiere que el espacio de características aprendido en un dominio simple es insuficiente para representar las particularidades de un dominio complejo. Aunque la transferencia puede acelerar el entrenamiento, la obtención de la máxima precisión para un problema específico exige un entrenamiento dedicado y una optimización de hiperparámetros a medida.

Este estudio presenta limitaciones que abren nuevas vías de investigación. La dependencia de simulaciones 2D promediadas en profundidad restringe la capacidad de capturar fenómenos tridimensionales, lo que podría ser relevante para la velocidad transversal ( $V$ ). Además, no se incluyeron modelos explícitos de fricción en las paredes del canal. Futuros trabajos podrían explorar el uso de FNO u otras arquitecturas como redes neuronales de grafos (GNNs) (Lino et al. 2022) con datos de simulaciones 3D. Asimismo, el enfoque actual se limita a flujos en estado estacionario; una progresión natural sería extender el marco a regímenes transientes (Wang et al. 2024; Li et al. 2021). Finalmente, para validar su aplicabilidad en el mundo real, sería provechoso evaluar el rendimiento de los FNO en problemas inversos con datos experimentales con más ruido o con datos de batimetrías reales.

## 5 CONCLUSIÓN

Esta investigación ha validado a los Operadores Neuronales de Fourier (FNO) como una herramienta potente para la modelización de flujos en canales, delineando claramente sus fortalezas y debilidades. Se ha demostrado que, con datos suficientes, un enfoque de aprendizaje puramente basado en datos es superior a la regularización física explícita para esta arquitectura.

La contribución más significativa es la demostración de la capacidad de los FNO para resolver problemas inversos dimensionales, reconstruyendo con alta fidelidad la batimetría y los parámetros hidráulicos a partir de campos de velocidad, incluso en geometrías complejas. Sin embargo, el estudio también identifica desafíos claros como la dificultad para predecir componentes de flujo secundarios y el fracaso de la inversión adimensional en escenarios complejos.

Sobre el uso de los FNO en hidrodinámica, son herramientas altamente eficaces para la inferencia de parámetros en su formulación dimensional, pero requieren un diseño cuidadoso y una optimización específica para cada tarea.

## REFERENCIAS

- Akiba, Takuya, Shotaro Sano, Toshihiko Yanase, Takeru Ohta y Masanori Koyama (jul. de 2019). «Optuna: A Next-generation Hyperparameter Optimization Framework». En: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '19: The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. Kdd '19. Anchorage AK USA: ACM, págs. 2623-2631. ISBN: 978-1-4503-6201-6. DOI: 10.1145/3292500.3330701. (Visitado 27-03-2025).
- Bischof, Raphael y Martin A Kraus (2025). «Multi-Objective Loss Balancing for Physics-Informed Deep Learning». En: *Computer Methods in Applied Mechanics and Engineering* 439, pág. 117914. DOI: 10.1016/j.cma.2025.117914.
- Brunton, Steven L, Bernd R Noack y Petros Koumoutsakos (2019). «Machine Learning for Fluid Mechanics». En: DOI: 10.1146/annurev-fluid-010719-060214.
- Cohen, Sagy, Tong Wan, Md Tazmul Islam y J. P. M. Syvitski (2018). «Global River Slope: A New Geospatial Dataset and Global-Scale Analysis». En: *Journal of Hydrology* 563, págs. 1057-1067. DOI: 10.1016/j.jhydrol.2018.06.066.
- He, Kaiming, Xiangyu Zhang, Shaoqing Ren y Jian Sun (2016). «Deep Residual Learning for Image Recognition». En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, págs. 770-778.
- Karniadakis, George Em, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang y Liu Yang (mayo de 2021). «Physics-Informed Machine Learning». En: *Nature Reviews Physics* 3.6, págs. 422-440. ISSN: 2522-5820. DOI: 10.1038/s42254-021-00314-5. (Visitado 04-01-2025).
- Li, Zongyi, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart y Anima Anandkumar (mayo de 2021). *Fourier Neural Operator for Parametric Partial Differential Equations*. DOI: 10.48550/arXiv.2010.08895. arXiv: 2010.08895 [cs]. (Visitado 04-01-2025).
- Lino, Mario, Stathi Fotiadis, Anil A. Bharath y Chris Cantwell (2022). *Towards Fast Simulation of Environmental Fluid Mechanics with Multi-Scale Graph Neural Networks*. arXiv: 2205.02637 [physics.flu-dyn].
- Loshchilov, Ilya y Frank Hutter (ene. de 2019). *Decoupled Weight Decay Regularization*. DOI: 10.48550/arXiv.1711.05101. arXiv: 1711.05101 [cs]. (Visitado 27-03-2025).
- Raissi, M., P. Perdikaris y G.E. Karniadakis (feb. de 2019). «Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations». En: *Journal of Computational Physics* 378, págs. 686-707. ISSN: 00219991. DOI: 10.1016/j.jcp.2018.10.045. (Visitado 04-01-2025).

Tang, Boxin (1993). «Orthogonal Array-Based Latin Hypercubes». En: *Journal of the American statistical Association* 88.424, págs. 1392-1397. doi: 10.1080/01621459.1993.10476423.

Vinuesa, Ricardo y Steven L. Brunton (jun. de 2022). «Enhancing Computational Fluid Dynamics with Machine Learning». En: *Nature Computational Science* 2.6, págs. 358-366. issn: 2662-8457. doi: 10.1038/s43588-022-00264-7. arXiv: 2110.02085 [physics]. (Visitado 04-01-2025).

Wang, Yuning, Alberto Solera-Rico, Carlos Sanmiguel Vila y Ricardo Vinuesa (feb. de 2024). «Towards Optimal  $\beta$ -Variational Autoencoders Combined with Transformers for Reduced-Order Modelling of Turbulent Flows». En: *International Journal of Heat and Fluid Flow* 105, pág. 109254. issn: 0142727X. doi: 10.1016/j.ijheatfluidflow.2023.109254. (Visitado 04-01-2025).