UNIVERSITY OF OXFORD

Efficient Error Certification for Physics-Informed Neural Networks

Francisco Girbal Eiras

Torr Vision Group · EPSRC CDT Autonomous Intelligent Machines and Systems (AIMS)

ICML 2024

Efficient Error Certification for Physics-Informed Neural Networks

UNIVERSITY OF OXFORD

Francisco Eiras
University of Oxford
eiras@robots.ox.ac.uk

Adel Bibi
University of Oxford
adel.bibi@eng.ox.ac.uk

Rudy Bunel
Google DeepMind
rbunel@google.com

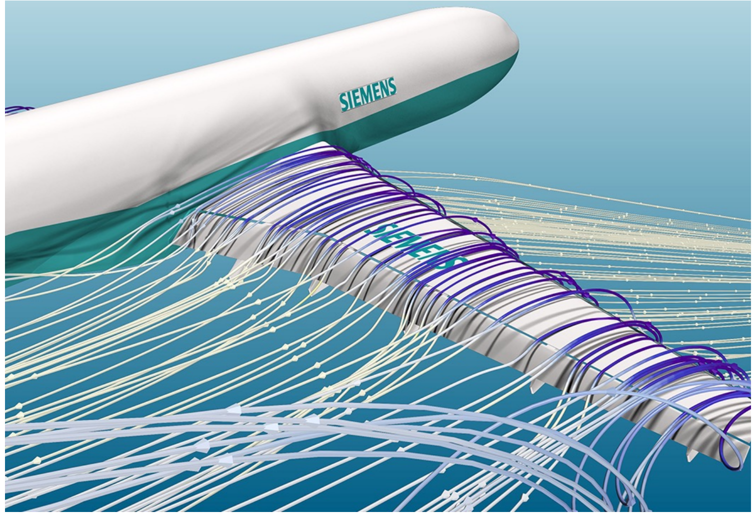
Krishnamurthy Dj Dvijotham
Google DeepMind
dvij@google.com

Philip H.S. Torr
University of Oxford
philip.torr@eng.ox.ac.uk

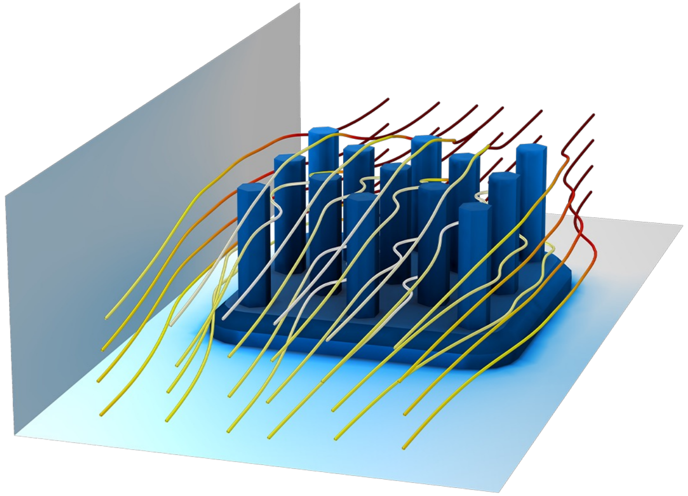
M. Pawan Kumar
Google DeepMind
mpawan@google.com

arXiv
https://arxiv.org/abs/2305.10157

Physical PDEs and Where to Find Them



Aerodynamics (e.g., Euler's equation)



Thermodynamics (e.g., Heat equation)

Earth and Space Sciences (e.g., turbulence in 2-D Navier-Stokes)

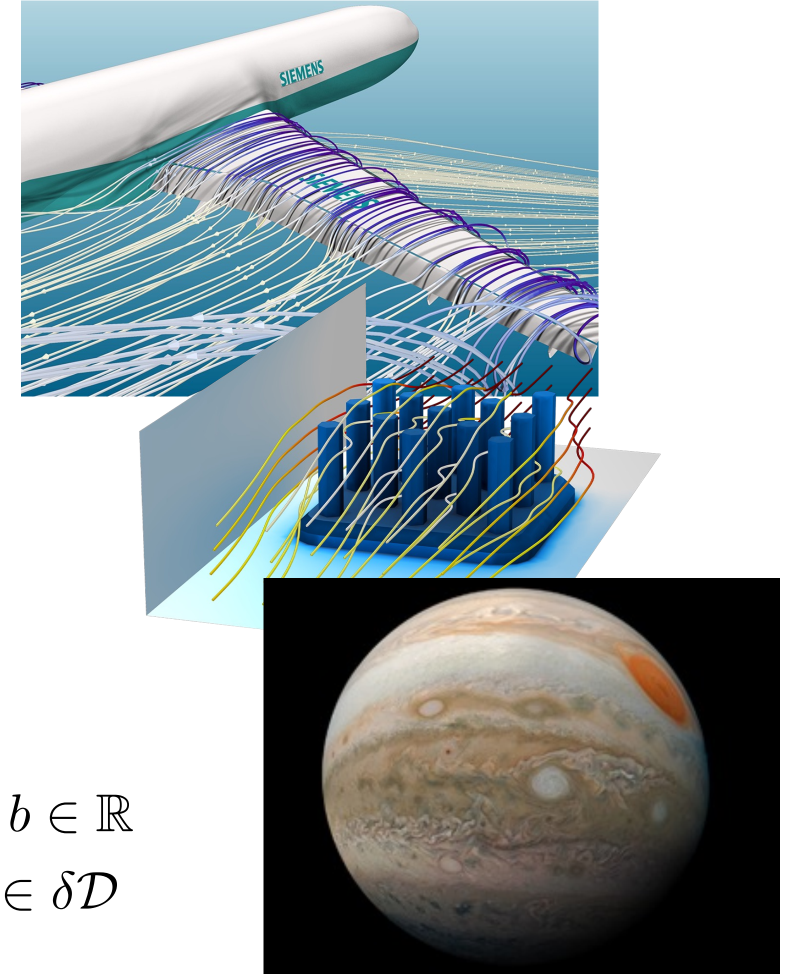
Among many others...

Nonlinear (Physical) Partial Differential Equations

solution — nonlinear spatial differential operator

$$\underbrace{\partial_t u(t,x) + \mathcal{N}[u](t,x)}_{\text{residual}} = 0, \qquad x \in \mathcal{D}, t \in [0,T]$$

domain



s.t.

1. **Initial condition**: $u(0,x) = u_0(x)$
2. **Robin boundary conditions**: $au(t,x) + b\partial_{\mathbf{n}} u(t,x) = u_b(t,x) \qquad a,b \in \mathbb{R}$

$$x \in \delta\mathcal{D}$$

PDE example: Diffusion-Sorption Equation

UNIVERSITY OF OXFORD

- Applications in groundwater contaminant transportation
- **1D Diffusion-Sorption**:

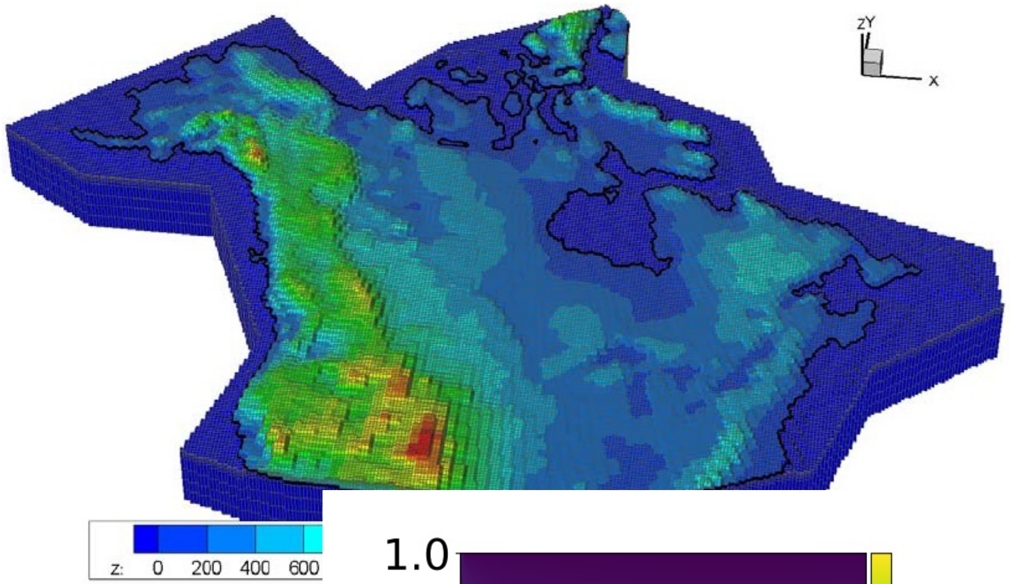
$$\partial_t u(t,x) - D/R(u(t,x))\partial_{x^2} u(t,x) = 0$$

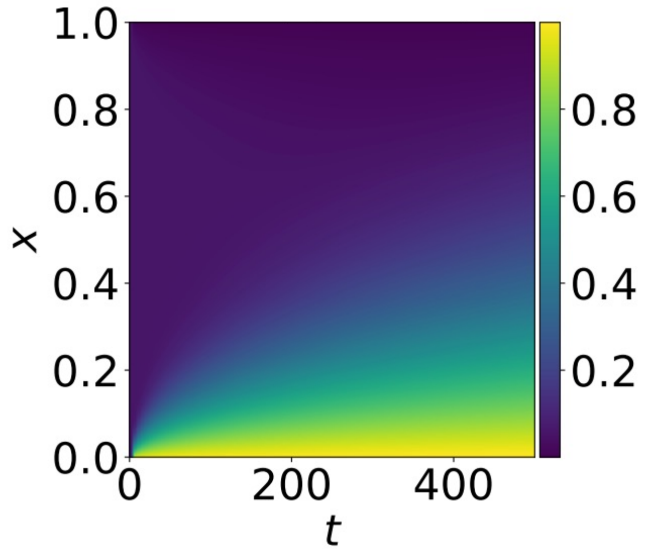
where

$$R(u(t,x)) = 1 + \frac{(1-\phi)}{(\phi)}\rho_s k n_f u^{n_f - 1}(t,x)$$

for physical constants D, ϕ, ρ_s, k, n_f, and:

$$u(0,x) = u(t,0) = 0, \quad u(t,1) = D\partial_x u(t,1)$$





1D Diffusion-Sorption

Takamoto, Makoto, et al. "PDEBench: An extensive benchmark for scientific machine learning." Advances in Neural Information Processing Systems 35 (2022): 1596-1611.
Lemieux, JM, et al. "Coupling continental glaciations with groundwater flow models–Surface/subsurface interactions over the Canadian landscape during the Wisconsinian glaciation."

UNIVERSITY OF OXFORD

Issue: Solving for $u(t, x)$ is computationally expensive

Solution: Use NNs to approximate it

Physics-Informed Neural Networks (PINNs)

UNIVERSITY OF OXFORD

- Approximate solution using a neural network, $u_\theta(t,x) \simeq u(t,x)$
- Take the residual evaluated for u_θ as the network $f_\theta(t,x) = \partial_t u_\theta(t,x) + \mathcal{N}[u_\theta](t,x)$

PINN

- Train *both networks* jointly using a loss evaluated at collocation points $\mathbb{P}$ (i.e. points in the domain):

$$\mathcal{L} = \underbrace{\sum_{x \in \mathbb{P}_0} |u(0,x) - u_\theta(0,x)|^2}_{\text{initial conditions}} + \underbrace{\sum_{(t,x) \in \mathbb{P}_b} |u(t,x) - u_\theta(t,x)|^2}_{\text{boundary conditions}} + \underbrace{\sum_{(t,x) \in \mathbb{P}_f} |f_\theta(t,x)|^2}_{\text{residual}}$$

- **Evaluate empirically by comparing $u_\theta(t,x)$ to the solution obtained by a numerical solver**

Raissi, Maziar, Paris Perdikaris, and George E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." Journal of Computational physics 378 (2019): 686-707.

PINN: 1D Diffusion-Sorption Equation

UNIVERSITY OF OXFORD

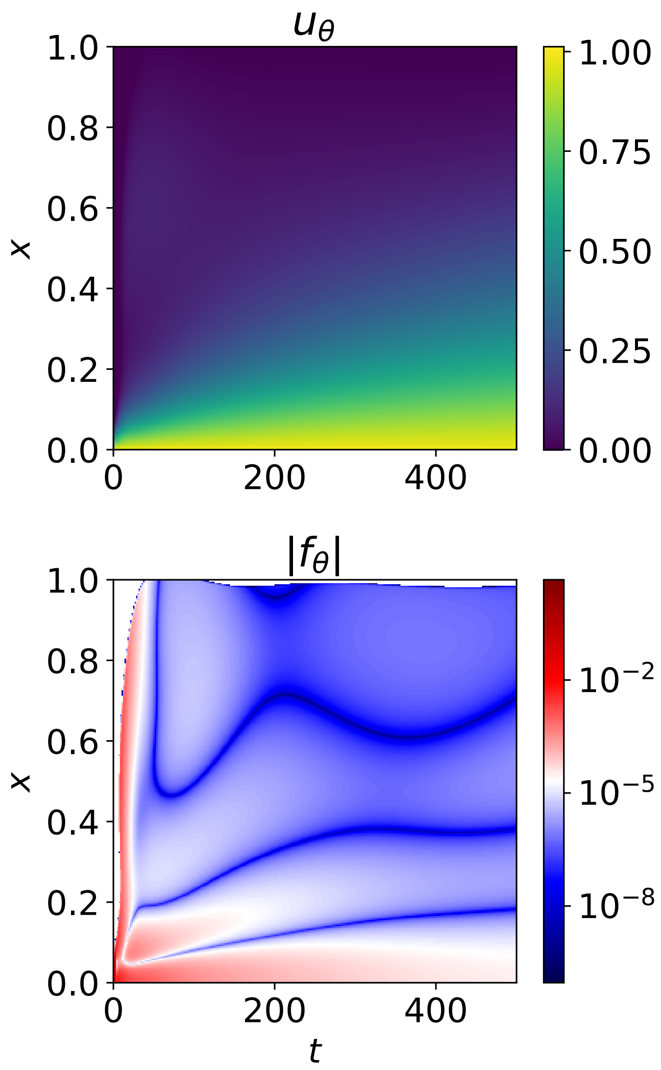
- Inference times on 2 core CPU + 1 GPU (NVIDIA V100):

Numerical Solver	PINN [Takamoto et al. 2022]
59.83s	2.7 x 10^{-3}

- PINN average ℓ_2 solution error is 9.9 x 10^{-2} compared to numerical solver

- Valid PDE solution **must** satisfy $f_\theta(t, x) = 0$
- *Is it satisfied across the domain?*

	10^4 Uniform Samples	10^6 Uniform Samples
$\max \lvert f_\theta \rvert^2$	1.1 x 10^{-3}	21.09



Takamoto, Makoto, et al. "PDEBench: An extensive benchmark for scientific machine learning." Advances in Neural Information Processing Systems 35 (2022): 1596-1611.

UNIVERSITY OF OXFORD

How can we be confident errors are small enough across the *entire* domain?

Defining Correctness Conditions for PINNs

UNIVERSITY OF OXFORD

- Intuitively, a PINN is a correct approximation of the underlying PDE if:
  1. *The solution satisfies the initial conditions* to a reasonable degree
  2. *The solution satisfies the boundary conditions* to a reasonable degree
  3. The *norm of the PINN output is small* enough

- Formally, for a D dimensional spatial input $\hat{\mathbf{x}} \in \mathcal{D}$, and solution/PINN input $\mathbf{x} = (t, \hat{\mathbf{x}})$:

Definition 1 (Correctness Conditions for PINNs). *$u_\theta : [0,T] \times \mathcal{D} \to \mathbb{R}$ is a $\delta_0, \delta_b, \varepsilon$-globally correct approximation of the exact solution $u : [0,T] \times \mathcal{D} \to \mathbb{R}$ if:*

① $\max_{\hat{\mathbf{x}} \in \mathcal{D}} |u_\theta(0, \hat{\mathbf{x}}) - u_0(\hat{\mathbf{x}})|^2 \leq \delta_0,$

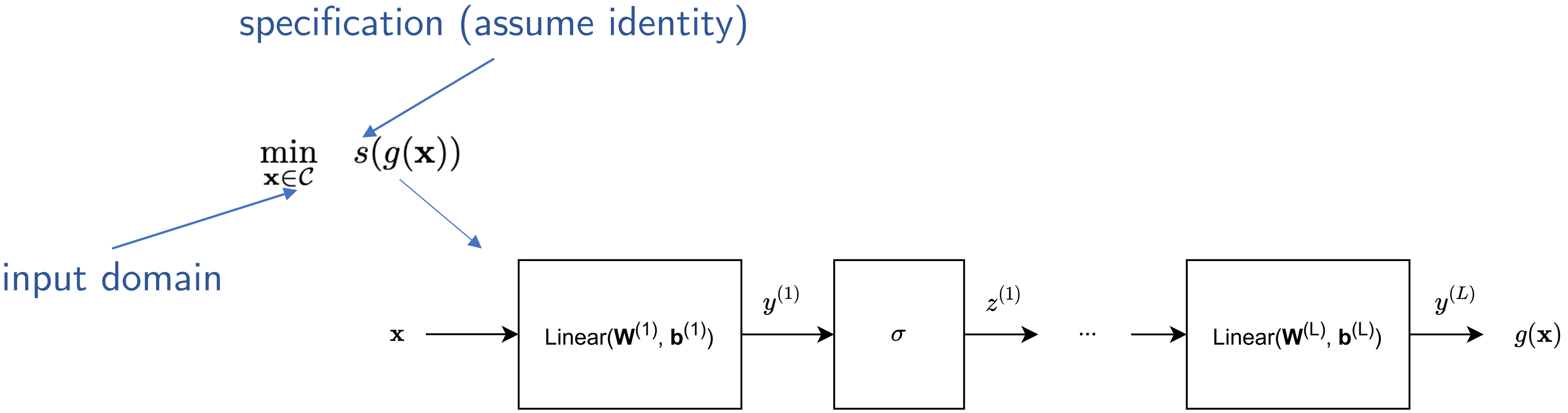
② $\max_{t \in [0,T], \hat{\mathbf{x}} \in \delta\mathcal{D}} |a u_\theta(t, \hat{\mathbf{x}}) + b\partial_{\mathbf{n}} u_\theta(t, \hat{\mathbf{x}}) - u_b(t, \hat{\mathbf{x}})|^2 \leq \delta_b,$

③ $\max_{\mathbf{x} \in \mathcal{C}} |f_\theta(\mathbf{x})|^2 \leq \varepsilon.$

Verification of Neural Networks

UNIVERSITY OF OXFORD

- CROWN/α-CROWN [Zhang et. al 2018, Xu et. al 2020]



  - Relax network to a linear program → solve it in closed form: $\min_{\mathbf{x}\in\mathcal{C}} g(\mathbf{x}) \geq u_b = \min_{\mathbf{x}\in\mathcal{C}} \mathbf{A}^L\mathbf{x} + \mathbf{a}^L$

relaxed problem

- Apply it to u_θ and f_θ using the previous specifications (?)

Zhang, Huan, et al. "Efficient neural network robustness certification with general activation functions." Advances in neural information processing systems 31 (2018).

Xu, Kaidi, et al. "Fast and complete: Enabling complete neural network verification with rapid and massively parallel incomplete verifiers." arXiv preprint arXiv:2011.13824 (2020).

Correctness Certification of PINNs (2)

UNIVERSITY OF OXFORD

Definition 1 (Correctness Conditions for PINNs). $u_\theta : [0,T] \times \mathcal{D} \to \mathbb{R}$ *is a* $\delta_0, \delta_b, \varepsilon$*-globally correct approximation of the exact solution* $u : [0,T] \times \mathcal{D} \to \mathbb{R}$ *if:*

✅ $$\max_{\hat{\mathbf{x}} \in \mathcal{D}} |u_\theta(0, \hat{\mathbf{x}}) - u_0(\hat{\mathbf{x}})|^2 \leq \delta_0,$$

❌ $$\max_{t \in [0,T], \hat{\mathbf{x}} \in \delta\mathcal{D}} |a u_\theta(t, \hat{\mathbf{x}}) + b \partial_{\mathbf{n}} u_\theta(t, \hat{\mathbf{x}}) - u_b(t, \hat{\mathbf{x}})|^2 \leq \delta_b,$$

❌ $$\max_{\mathbf{x} \in \mathcal{C}} |f_\theta(\mathbf{x})|^2 \leq \varepsilon.$$

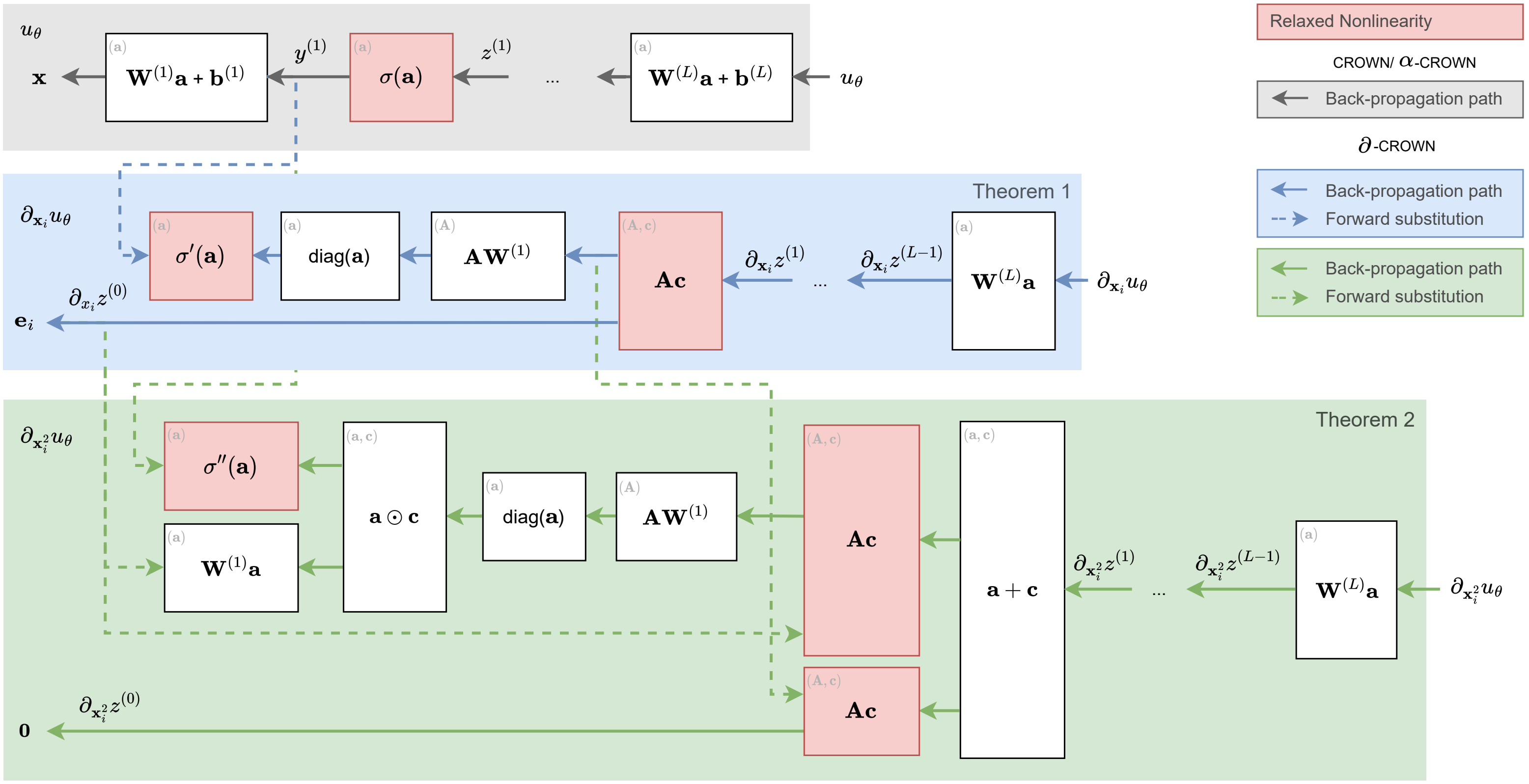
- *Applying CROWN/α-CROWN to boundary/residual conditions*:

❌ Architecture is completely different - f_θ is a nonlinear function of partial derivatives of u_θ

❌ Regression problem – bounds might be too loose to be informative

∂-CROWN: Bounding Partial Derivatives of u_θ and f_θ

- u_θ is bound using CROWN [Zhang et al. 2018]; partial derivatives require purpose-built efficient solution



Zhang, Huan, et al. "Efficient neural network robustness certification with general activation functions." Advances in neural information processing systems 31 (2018).

Xu, Kaidi, et al. "Fast and complete: Enabling complete neural network verification with rapid and massively parallel incomplete verifiers." arXiv preprint arXiv:2011.13824 (2020).

∂-CROWN: Bounding Partial Derivatives of u_θ and f_θ

UNIVERSITY OF OXFORD

- u_θ is bound using CROWN [Zhang et al. 2018]; partial derivatives require purpose-built efficient solution

1st (Theorem 1) and 2nd (Theorem 2) linear bounding of partial derivatives

Hybrid scheme with complexity $\mathcal{O}(L)$ instead of $\mathcal{O}(L^2)$ from [Xu et al. 2020]

Global bounds computed in close-form (similarly to [Zhang et al. 2018])

Theorem 1 (∂-CROWN: Linear Bounding $\partial_{\mathbf{x}_i} u_\theta$). *There exist two linear functions $\partial_{\mathbf{x}_i} u_\theta^U$ and $\partial_{\mathbf{x}_i} u_\theta^L$ s.t. it holds $\forall \mathbf{x} \in \mathcal{C}$: $\partial_{\mathbf{x}_i} u_\theta^L \leq \partial_{\mathbf{x}_i} u_\theta \leq \partial_{\mathbf{x}_i} u_\theta^U$, with the linear coefficients computed recursively in closed-form in $\mathcal{O}(L)$ time.*

Theorem 2 (∂-CROWN: Linear Bounding $\partial_{\mathbf{x}_i^2} u_\theta$). *Assume that through a previous bounding of $\partial_{\mathbf{x}_i} u_\theta$, we have linear lower and upper bounds on $\partial_{\mathbf{x}_i} z^{(k-1)}$ and $\partial_{z^{(k-1)}} z^{(k)}$. There exist two linear functions $\partial_{\mathbf{x}_i^2} u_\theta^U$ and $\partial_{\mathbf{x}_i^2} u_\theta^L$ s.t. it holds $\forall \mathbf{x} \in \mathcal{C}$: $\partial_{\mathbf{x}_i^2} u_\theta^L \leq \partial_{\mathbf{x}_i^2} u_\theta \leq \partial_{\mathbf{x}_i^2} u_\theta^U$, with the coefficients computed recursively in closed-form in $\mathcal{O}(L)$ time.*

- f_θ is linearly bounded using McCormick envelopes; global bounds computed in close-form

Zhang, Huan, et al. "Efficient neural network robustness certification with general activation functions." Advances in neural information processing systems 31 (2018).

Xu, Kaidi, et al. "Fast and complete: Enabling complete neural network verification with rapid and massively parallel incomplete verifiers." arXiv preprint arXiv:2011.13824 (2020).

Correctness Certification of PINNs (3)

Definition 1 (Correctness Conditions for PINNs). $u_\theta : [0,T] \times \mathcal{D} \to \mathbb{R}$ *is a* $\delta_0, \delta_b, \varepsilon$*-globally correct approximation of the exact solution* $u : [0,T] \times \mathcal{D} \to \mathbb{R}$ *if:*

✅ $$\max_{\hat{\mathbf{x}} \in \mathcal{D}} |u_\theta(0, \hat{\mathbf{x}}) - u_0(\hat{\mathbf{x}})|^2 \leq \delta_0,$$

❌ $$\max_{t \in [0,T], \hat{\mathbf{x}} \in \delta\mathcal{D}} |a u_\theta(t, \hat{\mathbf{x}}) + b \partial_{\mathbf{n}} u_\theta(t, \hat{\mathbf{x}}) - u_b(t, \hat{\mathbf{x}})|^2 \leq \delta_b,$$

❌ $$\max_{\mathbf{x} \in \mathcal{C}} |f_\theta(\mathbf{x})|^2 \leq \varepsilon.$$

- *Applying CROWN/α-CROWN to boundary/residual conditions*:

✅ Architecture is completely different - f_θ is a nonlinear function of partial derivatives of u_θ

❌ Regression problem – bounds might be too loose to be informative

↑ Greedy Input Branching

Correctness Certification of PINNs (4)

UNIVERSITY OF OXFORD

Definition 1 (Correctness Conditions for PINNs). $u_\theta : [0,T] \times \mathcal{D} \to \mathbb{R}$ *is a* $\delta_0, \delta_b, \varepsilon$*-globally correct approximation of the exact solution* $u : [0,T] \times \mathcal{D} \to \mathbb{R}$ *if:*

✅ $$\max_{\hat{\mathbf{x}} \in \mathcal{D}} |u_\theta(0, \hat{\mathbf{x}}) - u_0(\hat{\mathbf{x}})|^2 \leq \delta_0,$$

✅ $$\max_{t \in [0,T], \hat{\mathbf{x}} \in \delta\mathcal{D}} |a u_\theta(t, \hat{\mathbf{x}}) + b \partial_{\mathbf{n}} u_\theta(t, \hat{\mathbf{x}}) - u_b(t, \hat{\mathbf{x}})|^2 \leq \delta_b,$$

✅ $$\max_{\mathbf{x} \in \mathcal{C}} |f_\theta(\mathbf{x})|^2 \leq \varepsilon.$$

- *Applying CROWN/α-CROWN to boundary/residual conditions*:
  - ✅ Architecture is completely different - f_θ is a nonlinear function of partial derivatives of u_θ
  - ✅ Regression problem – bounds might be too loose to be informative

Experiments: Certifying with ∂-CROWN

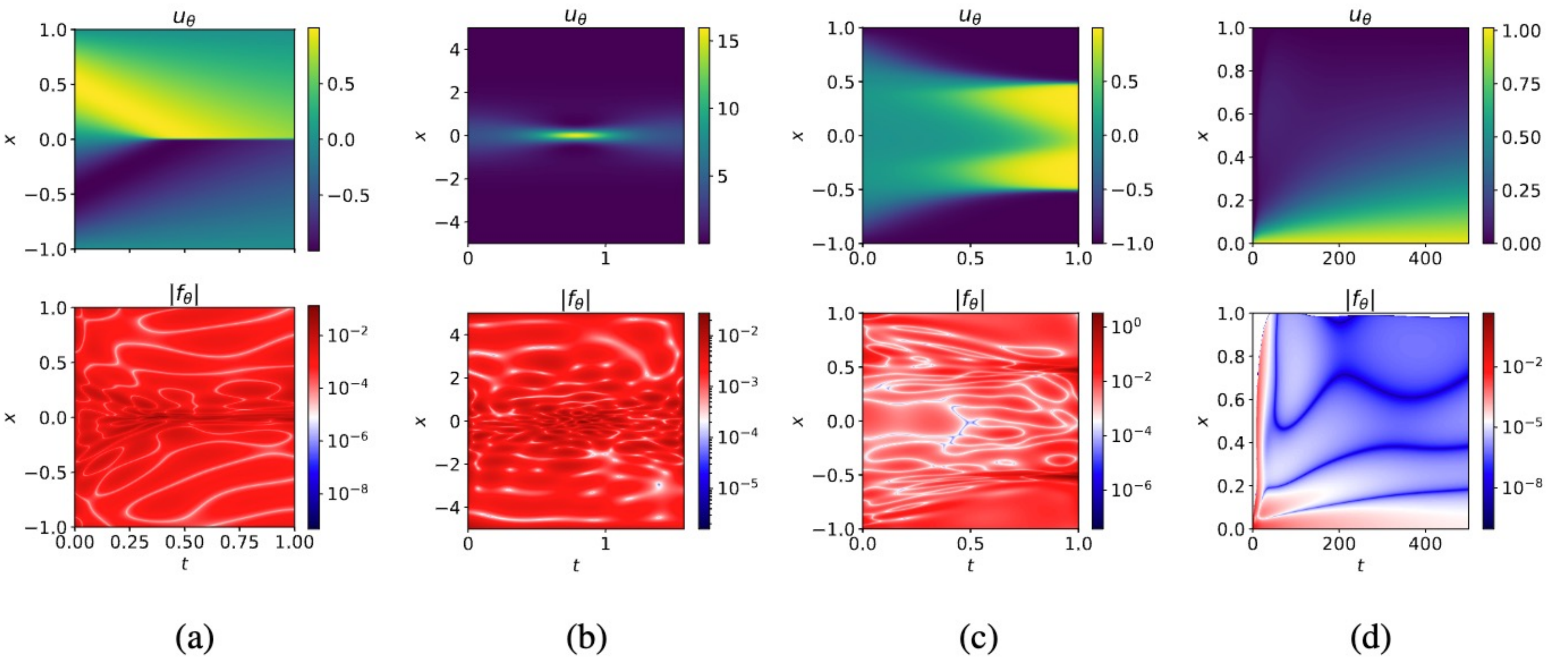


Figure 2: **Certifying with ∂-CROWN**: visualization of the time evolution of u_θ, and the residual errors as a function of the spatial temporal domain (log-scale), $|f_\theta|$, for **(a)** Burgers' equation [Raissi et al., 2019b], **(b)** Schrödinger's equation [Raissi et al., 2019b], **(c)** Allan-Cahn's equation [Monaco and Apiletti, 2023], and **(d)** the Diffusion-Sorption equation [Takamoto et al., 2022].

Raissi, Maziar, Paris Perdikaris, and George E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." Journal of Computational physics 378 (2019): 686-707.

Monaco, Simone, and Daniele Apiletti. "Training physics-informed neural networks: One learning to rule them all?." Results in Engineering 18 (2023): 101023.

Takamoto, Makoto, et al. "PDEBench: An extensive benchmark for scientific machine learning." Advances in Neural Information Processing Systems 35 (2022): 1596-1611.

Experiments: Certifying with ∂-CROWN

UNIVERSITY OF OXFORD

			Empirical l_b		Certified u_b
			MC max (10^4)	MC max (10^6)	∂-CROWN u_b (time [s])
(a) Burgers	①	$\lvert u_\theta(0,x) - u_0(x)\rvert^2$	1.59×10^{-6}	1.59×10^{-6}	2.63×10^{-6} (116.5)
	②	$\lvert u_\theta(t,-1)\rvert^2$	8.08×10^{-8}	8.08×10^{-8}	6.63×10^{-7} (86.7)
		$\lvert u_\theta(t,1)\rvert^2$	6.54×10^{-8}	6.54×10^{-8}	9.39×10^{-7} (89.8)
	③	$\lvert f_\theta(\mathbf{x})\rvert^2$	1.23×10^{-3}	1.80×10^{-2}	1.03×10^{-1} (2.8×10^5)
(b) Schrödinger	①	$\lvert u_\theta(0,x) - u_0(x)\rvert^2$	7.06×10^{-5}	7.06×10^{-5}	8.35×10^{-5} (305.2)
	②	$\lvert u_\theta(t,5) - u_\theta(t,-5)\rvert^2$	7.38×10^{-7}	7.38×10^{-7}	5.73×10^{-6} (545.4)
		$\lvert \partial_x u_\theta(t,5) - \partial_x u_\theta(t,-5)\rvert^2$	1.14×10^{-5}	1.14×10^{-5}	5.31×10^{-5} (2.4×10^3)
	③	$\lvert f_\theta(\mathbf{x})\rvert^2$	7.28×10^{-4}	7.67×10^{-4}	5.55×10^{-3} (1.2×10^6)
(c) Allen-Cahn	①	$\lvert u_\theta(0,x) - u_0(x)\rvert^2$	1.60×10^{-3}	1.60×10^{-3}	1.61×10^{-3} (52.7)
	②	$\lvert u_\theta(t,-1) - u_\theta(t,1)\rvert^2$	5.66×10^{-6}	5.66×10^{-6}	5.66×10^{-6} (95.4)
	③	$\lvert f_\theta(\mathbf{x})\rvert^2$	10.74	10.76	10.84 (6.7×10^5)
(d) Diffusion-Sorption	①	$\lvert u_\theta(0,x)\rvert^2$	0.0	0.0	0.0 (0.2)
	②	$\lvert u_\theta(t,0) - 1\rvert^2$	4.22×10^{-4}	4.39×10^{-4}	1.09×10^{-3} (72.5)
		$\lvert u_\theta(t,1) - D\partial_x u_\theta(t,1)\rvert^2$	2.30×10^{-5}	2.34×10^{-5}	2.37×10^{-5} (226.4)
	③	$\lvert f_\theta(\mathbf{x})\rvert^2$	1.10×10^{-3}	21.09	21.34 (2.4×10^6)

Efficiency of ∂-CROWN and Solution-Residual Errors

Table 2: **Efficiency of ∂-CROWN**: comparison of ∂-CROWN (Ours), Interval Bound Propagation (IBP) and LiRPA upper bounds obtained with greedy input branching (for N_b branches) in Burgers' equation for fixed runtime limits (150s, 100s, or 10^4s). Lower is better.

	Ours (N_b)	IBP (N_b)	LiRPA (N_b)
$\lvert u_\theta(0,x)\rvert^2$ (150s)	2.63×10^{-6} (10^4)	4.12×10^{-3} (10^5)	$\mathbf{2.23 \times 10^{-6}}$ (10^4)
$\lvert u_\theta(t,-1)\rvert^2$ (100s)	6.63×10^{-7} (10^4)	1.23×10^{-5} (10^5)	$\mathbf{6.34 \times 10^{-7}}$ (10^4)
$\lvert u_\theta(t,1)\rvert^2$ (100s)	9.39×10^{-7} (10^4)	5.69×10^{-5} (10^5)	$\mathbf{9.12 \times 10^{-7}}$ (10^4)
$\lvert f_\theta(x,t)\rvert^2$ (10^4s)	$\mathbf{1.30 \times 10^{1}}$ (1.3×10^5)	2.78×10^3 (5×10^6)	1.78×10^2 (1.9×10^4)

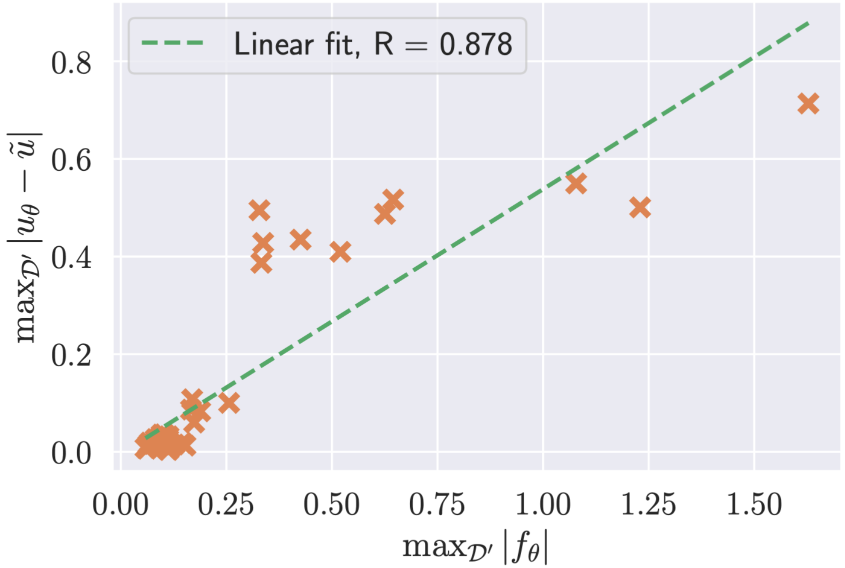


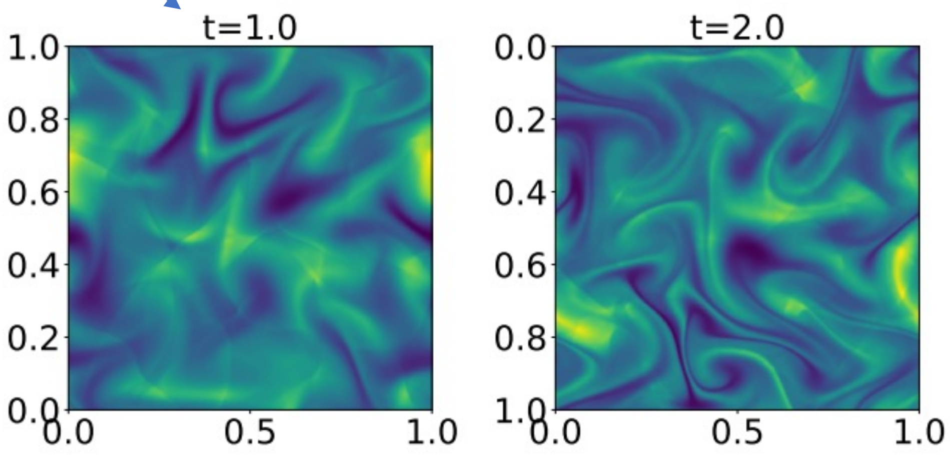
Figure 2: **Residual and solution errors**: connection of the maximum residual error ($\max_{\mathcal{S}'} |f_\theta|$) and the maximum solution error, $\max_{\mathcal{S}'} |u_\theta - \tilde{u}|$, for networks at different epochs of the training process (in orange).

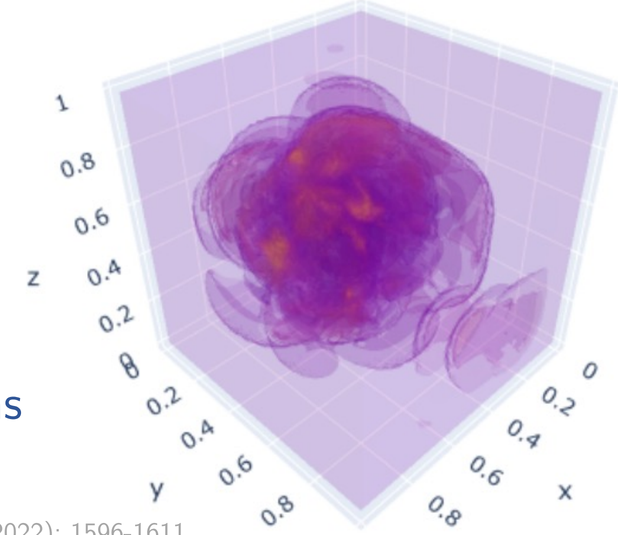
Future work

UNIVERSITY OF OXFORD

- Improve scalability of ∂-CROWN to apply it to larger PINNs, e.g., from [Takamoto et al. 2022]
- Extend framework to other physics-informed neural operators [Li et al. 2021]
- ∂-CROWN provides **post-training certification**:
  - Investigate further training methods to improve empirical/certified PINN accuracy

2D Compressible Navier-Stokes equations





3D Turbulent Flow equations

arXiv
https://arxiv.org/abs/2305.10157

Takamoto, Makoto, et al. "PDEBench: An extensive benchmark for scientific machine learning." Advances in Neural Information Processing Systems 35 (2022): 1596-1611.
Li, Zongyi, et al. "Physics-informed neural operator for learning partial differential equations." arXiv preprint arXiv:2111.03794 (2021).