Sobolev Cubatures Strengthen the Approximation Power of Physics Informed Neural Nets

Juan Esteban Suarez – Hecht Lab CASUS

CASUS
CENTER FOR ADVANCED SYSTEMS UNDERSTANDING

www.casus.science

HELMHOLTZ ZENTRUM DRESDEN ROSSENDORF
TECHNISCHE UNIVERSITÄT DRESDEN
CBG Max Planck Institute of Molecular Cell Biology and Genetics
UFZ HELMHOLTZ Centre for Environmental Research
Uniwersytet Wrocławski

SPONSORED BY THE
Federal Ministry of Education and Research
STAATSMINISTERIUM FÜR WISSENSCHAFT KULTUR UND TOURISMUS
Freistaat SACHSEN

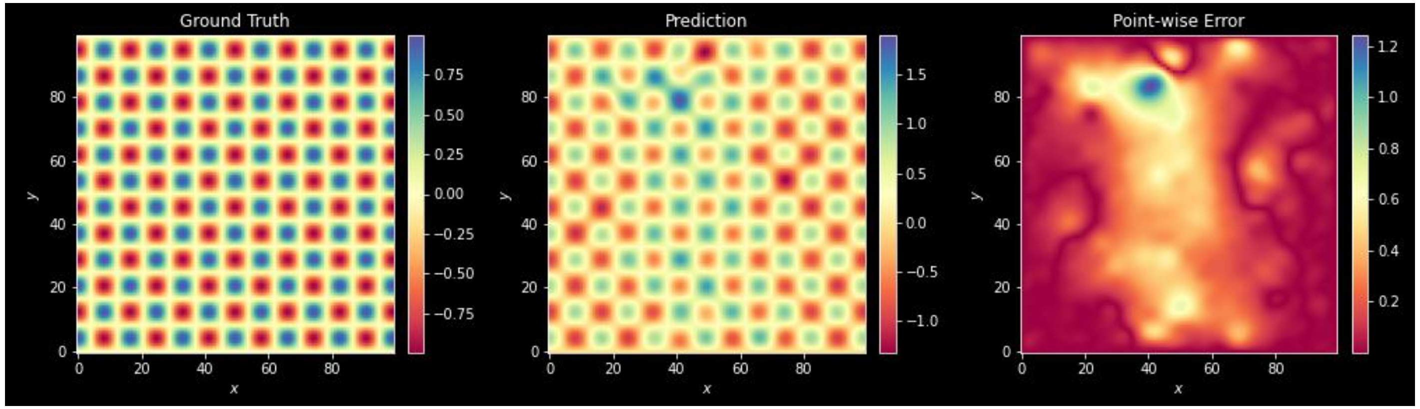
What are PINNs and why not use classic PDE solvers ?

CENTER FOR ADVANCED SYSTEMS UNDERSTANDING

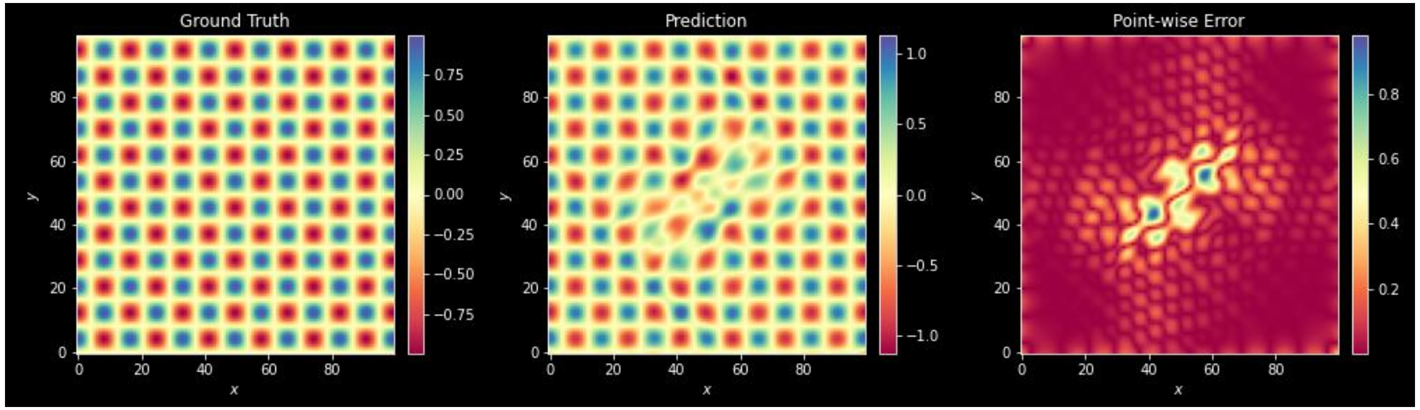
- Neural Networks with physics encoded by a PDE Loss
- Forward problem learning → surrogate models
- Inverse problem / Inferring parameters and governing equations

  → Costly for classic solvers
- Accuracy of PINNs → Challenge we want to tackle

- **2-D Poisson equation forward problem,** $N_{points} = 961, \omega = 6\pi$
- MSE Loss with Inverse Dirichlet [2] and A.D., $\epsilon_{L^\infty} = 1.24$, $t_{cpu} \approx 400$



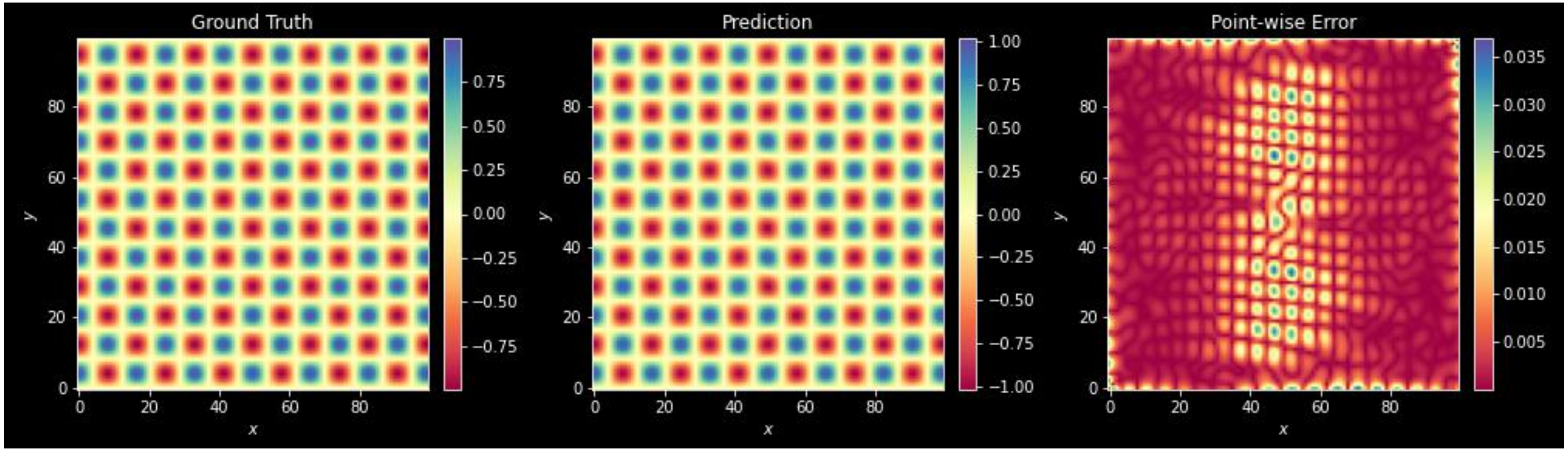
- Sobolev Loss ($\mathcal{L}_r = d_{L^2}, \mathcal{L}_b = d_{H^2}$) with A.D., $\epsilon_{L^\infty} = 9.78 \cdot 10^{-1}$, $t_{cpu} \approx 321$



[2] *Suryanarayana Maddu et al 2021 Mach. Learn.: Sci. Technol. in press https://doi.org/10.1088/2632- 2153/ac3712*

2-D Poisson equation forward problem, $N_{points} = 961$

- Sobolev Loss ($\mathcal{L}_r = d_{L^2}, \mathcal{L}_b = d_{H^2}$) without A.D., $\epsilon_{L^\infty} = 3.6 \cdot 10^{-2}$, $t_{cpu} \approx 190$



- Automatic differentiation is replaced by a discrete differential operator on a polynomial space.

Sobolev Spaces H^k

- $H^0 = L^2$ –Lebesgue space, $\Omega := (-1,1)^m \subset \mathbb{R}^m$

- The k-th Sobolev space $H^k(\Omega;\mathbb{R})$ is defined as:

$$H^k(\Omega;\mathbb{R}) \coloneqq \{f \in L^2(\Omega;\mathbb{R}) \mid \forall\, |\alpha| \le k, D^\alpha f \in L^2(\Omega;\mathbb{R})\}$$

- H^k is a Hilbert space with the norm:

$$||u-v||_{H^k}^2 \coloneqq ||u-v||_{L^2}^2 + \sum_{|\alpha|\le k} ||D^\alpha u - D^\alpha v||_{L^2}^2$$

- Sobolev Embedding Theorem: For $k > \frac{m}{2}, H^k(\Omega;\mathbb{R}) \subseteq C^0(\Omega;\mathbb{R})$

  If $||u-v||_{H^k} = 0 \longrightarrow u(x) = v(x)$, for all $x \in \Omega$

Variational Formulations (Example)

- Poisson equation with Dirichlet boundary conditions:

$$\begin{cases} -\Delta \mathrm{u} - f = 0 \ \text{ for } \mathrm{x} \in \Omega \\ \mathrm{u} = \mathrm{g} \ \text{ for } \mathrm{x} \in \partial\Omega \end{cases}$$

- We denote by **strong variational formulation**:

$$\int_{\Omega} (-\Delta \mathrm{u} - f)\phi \, dx = 0, \qquad \text{for } \mathrm{x} \in \Omega, \forall \phi \in C^{\infty}(\Omega; \mathbb{R})$$

- We denote by **weak variational formulation**:

$$\int_{\Omega} (\nabla \mathrm{u} \cdot \nabla \phi - f\phi) \, dx + \int_{\partial\Omega} (\phi \nabla u) \cdot \boldsymbol{n} \ dx = 0, \qquad \text{for } \mathrm{x} \in \Omega, \forall \phi \in C^{\infty}(\Omega; \mathbb{R})$$

- In practice we solve over a Polynomial space $\Pi_A \subset C^{\infty}(\Omega; \mathbb{R})$,spanned by the Lagrange basis $\{L_\alpha\}_{\alpha \in A}$

Computing Losses

- MSE Loss ⟶ Approximation of the L^2-integral

$$\int_\Omega (u-v)^2 dx \approx \frac{1}{N}\sum_{n\leq \mathrm{N}} \big(u(x_n)-v(x_n)\big)^2$$

- Gauss-Legendre cubature rules ⟶ Integrating polynomials exactly

$$\int_\Omega (u-v)^2 dx \approx \sum_{\alpha\in A} \big(u(P_\alpha)-v(P_\alpha)\big)^2 \omega_\alpha$$

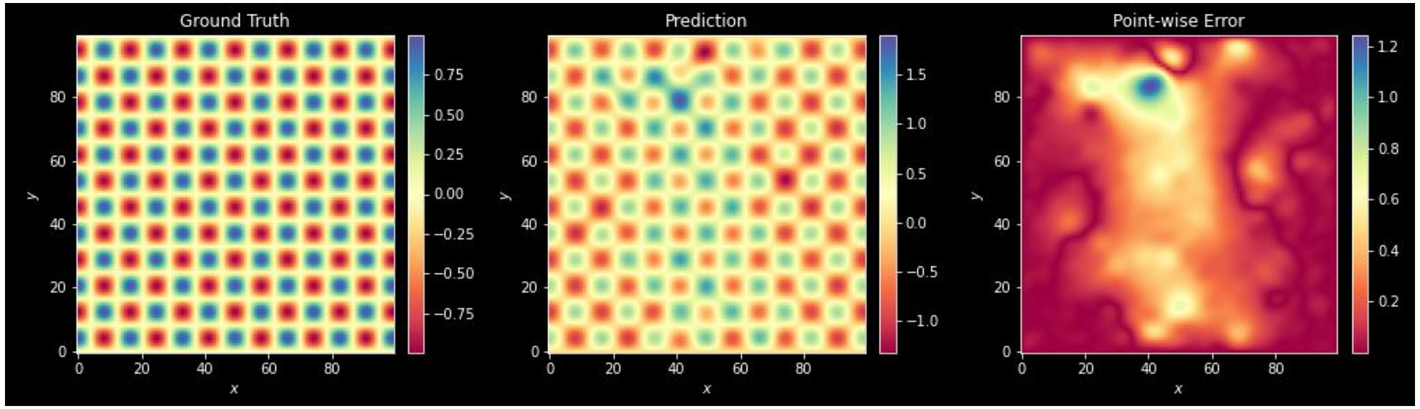
- Generalize the Gauss-Legendere rule to Sobolev-Cubatures, *minterpy* [1]

$$\int_\Omega (D^\alpha u - D^\alpha v)^2 dx \approx \sum_{\alpha\in A} \big(u(P_\alpha)-v(P_\alpha)\big)^2 s_\alpha$$

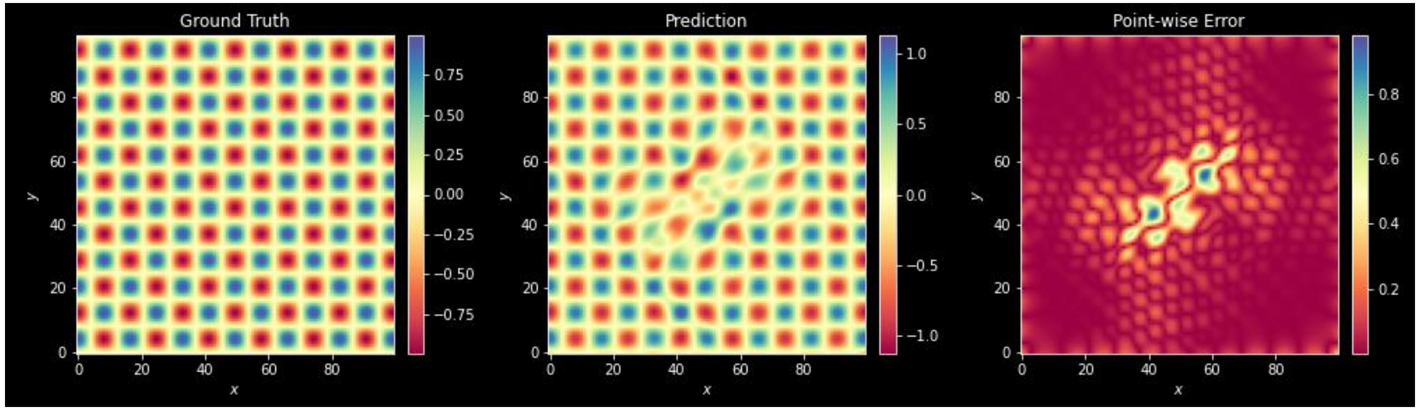
MINTERPY

[1] minterpy Multi dimensional interpolation python 2021 *https://github.com/casus/minterpy/releases*

- **2-D Poisson equation forward problem,** $N_{points} = 961$
- MSE Loss with Inverse Dirichlet [2] and A.D., $\epsilon_{L^\infty} = 1.24$, $t_{cpu} \approx 400$



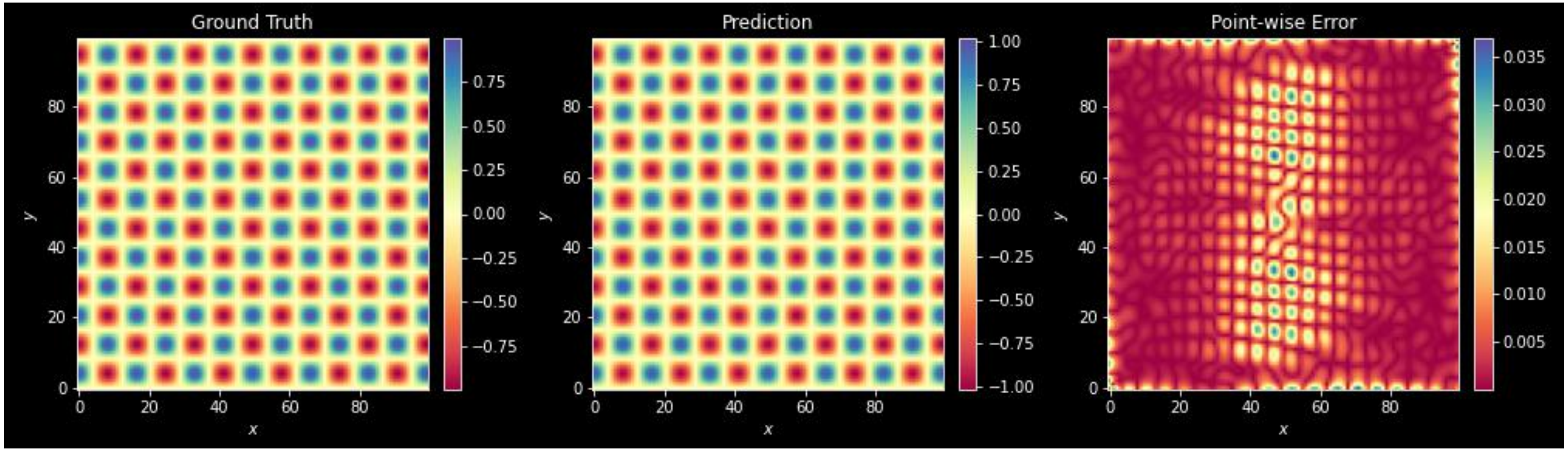
- Strong Sobolev Loss ($\mathcal{L}_r = d_{L^2}, \mathcal{L}_b = d_{H^2}$) with A.D., $\epsilon_{L^\infty} = 9.78 \cdot 10^{-1}$, $t_{cpu} \approx 321$



[2] Suryanarayana Maddu et al 2021 Mach. Learn.: Sci. Technol. in press https://doi.org/10.1088/2632-2153/ac3712

2-D Poisson equation forward problem, $N_{points} = 961$

- Strong Sobolev Loss ($\mathcal{L}_r = d_{L^2}, \mathcal{L}_b = d_{H^2}$) without A.D., $\epsilon_{L^\infty} = 3.6 \cdot 10^{-2}$, $t_{cpu} \approx 190$

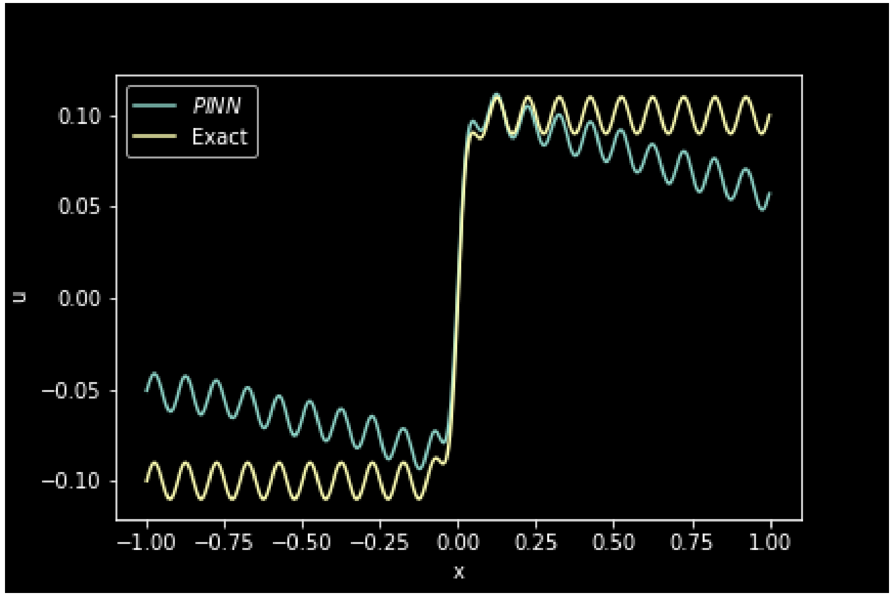


- Automatic differentiation is replaced by a discrete differential operator on a polynomial space.

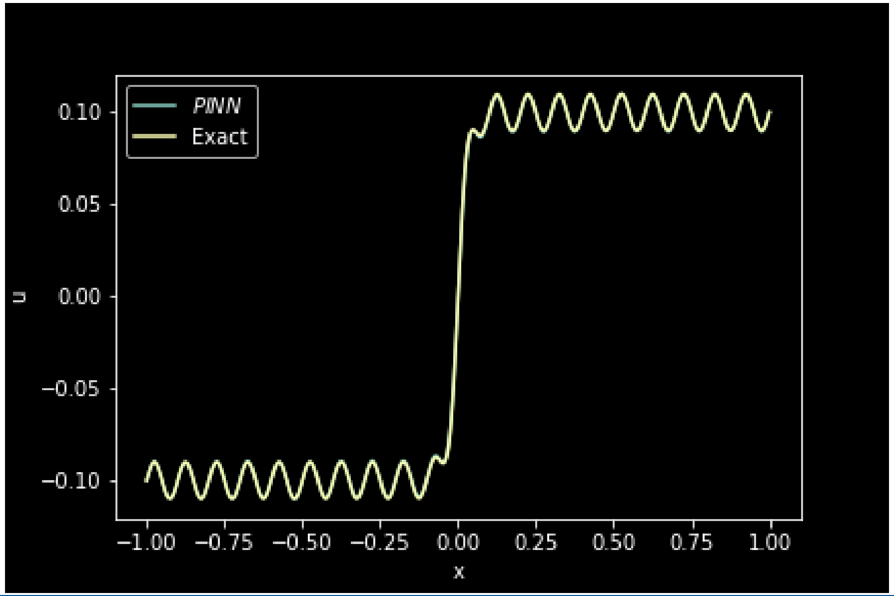
Other Numerical Results

- **1-D Poisson equation forward problem,** $\omega = 15\pi$ **,** $N_{points} = 100$

- Strong Sobolev Loss, $\epsilon_{L^\infty} = 0.049$

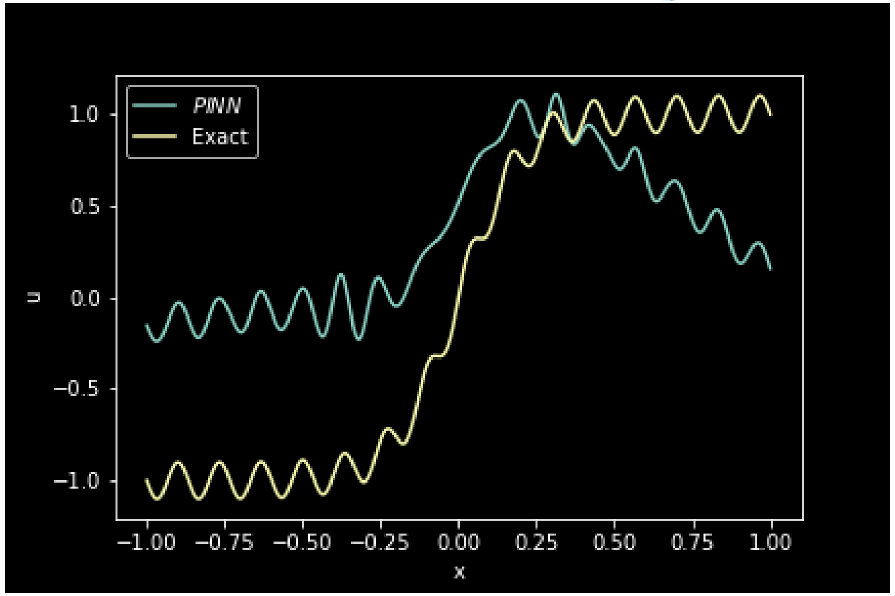


- Strong Variational Loss, $\epsilon_{L^\infty} = 0.002$

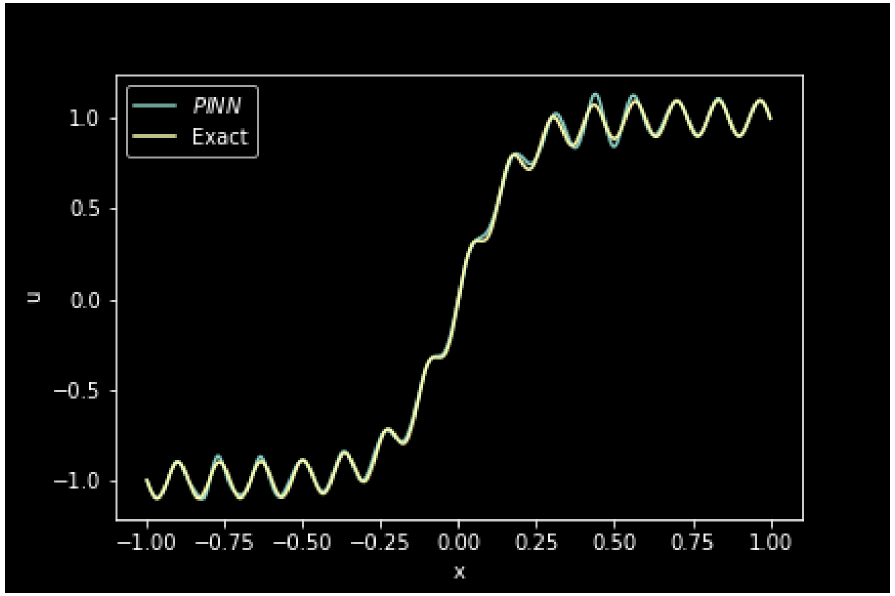


- **1-D Poisson equation forward problem,** $\omega = 15$ **,** $N_{points} = 50$

- Strong Sobolev Loss, $\epsilon_{L^\infty} = 1.002$

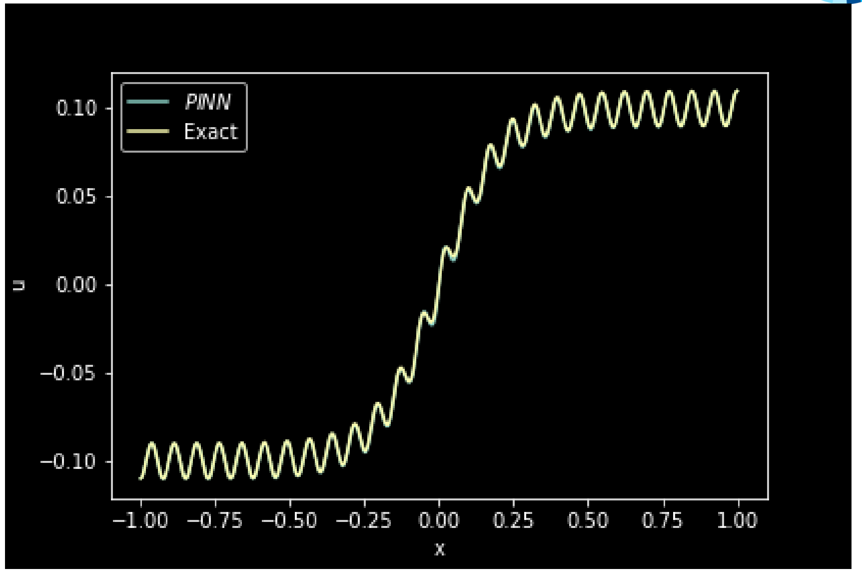


- Variational Loss, $\epsilon_{L^\infty} = 0.063$

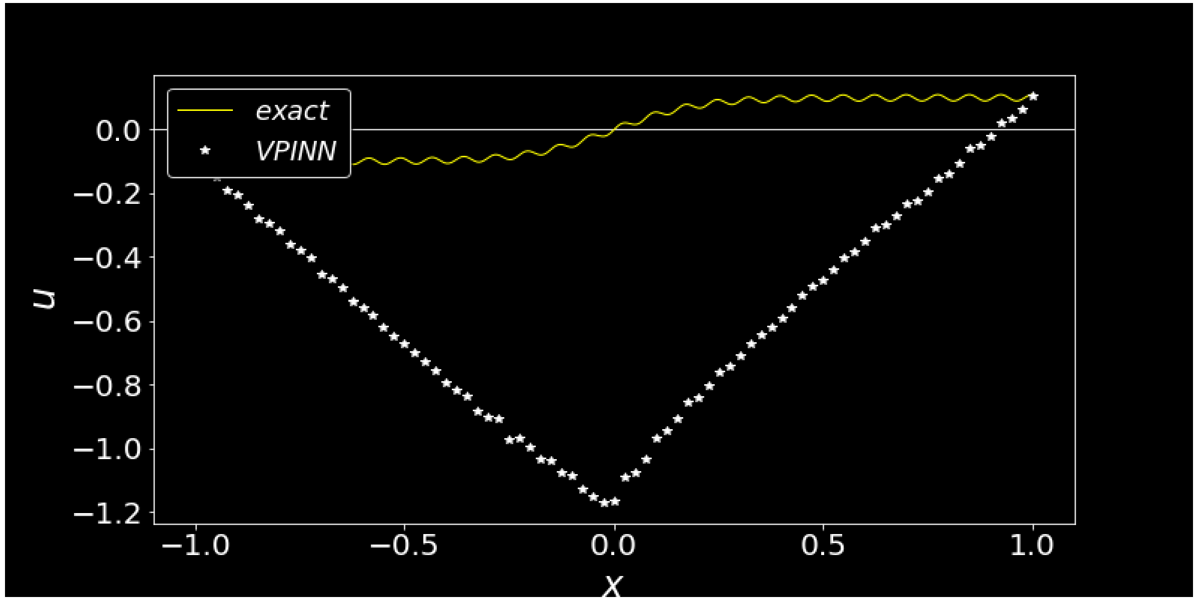


1-D scaled Poisson equation forward problem, $\omega = 26.5\pi$, $N_{points} = 100$

CASUS CENTER FOR ADVANCED SYSTEMS UNDERSTANDING



- Variational Loss, $\epsilon_{L^\infty} = 0.003$, $t_{cpu} \approx 130$

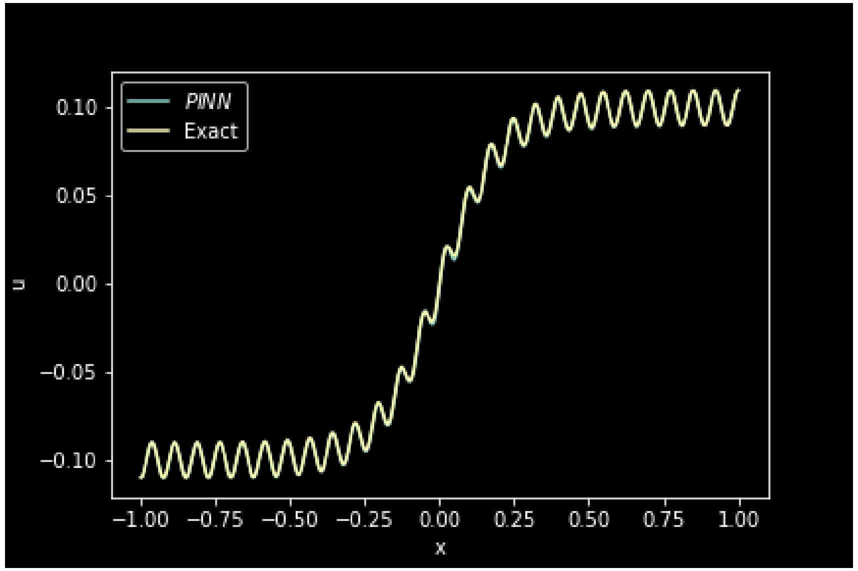


- VPINN [3], $\epsilon_{L^\infty} = 1.17$, $t_{cpu} \approx 144$

[3] Kharazmi E. et al 2021,
Computer Methods in Applied Mechanics and Engineering in press
https://doi.org/10.1016/j.cma.2020.113547

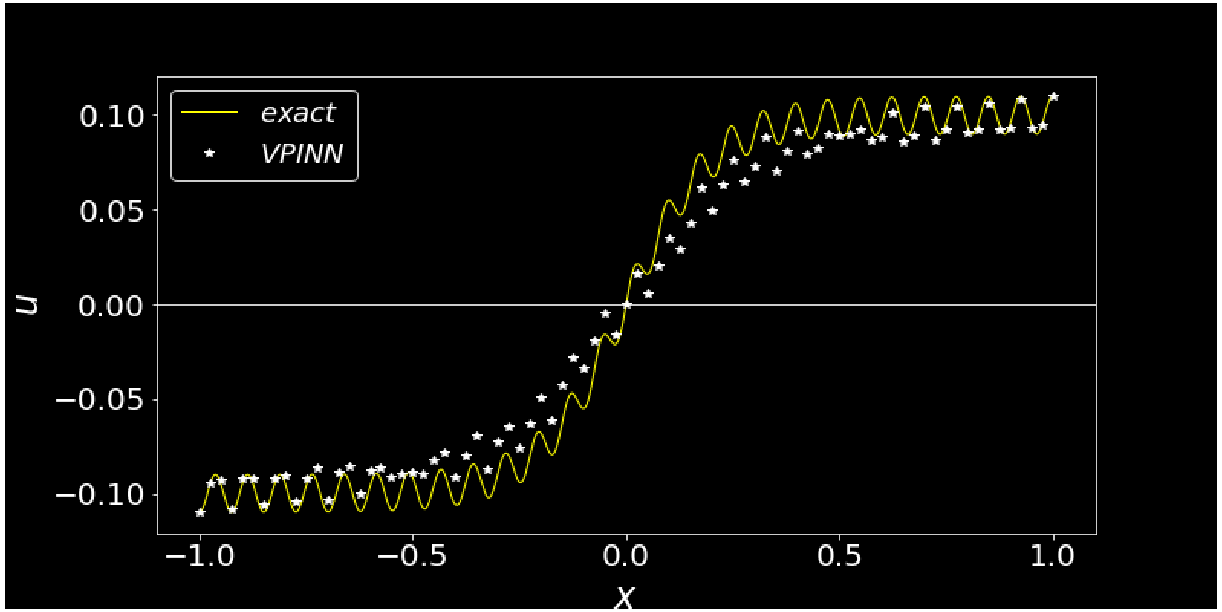
1-D scaled Poisson equation forward problem, $\omega = 26.5$, $N_{points} = 100$

- Variational Loss, $\epsilon_{L^\infty} = 0.003$, $t_{cpu} \approx 130$



- VPINN [3], wit domain decomposition, $N_{dom} = 3$, $\epsilon_{L^\infty} = 0.02$, $t_{cpu} \approx 360$

- MSE with I.D. [2] becomes unstable



Overview

- This are preliminary results ⟶ we are working to scale it up, e.g. higher dimensions

- Sobolev Cubatures are expected to be included as feature on the Beta release of *minterpy*

- We want to extend the approach to complex geometries (Biological Systems)

- We want to apply the method to real problems in cooperation with:
  - Ivo Sbalzarini, MOSAIC group CSBD
  - Attila Cangi, Matter Under Extreme Conditions CASUS
  - Nico Hoffman, ML group HZDR

- Contact us if you are interested ! j.suarez-cardona@hzdr.de

CASUS
CENTER FOR ADVANCED SYSTEMS UNDERSTANDING

Thank you !

References

- [1] *minterpy Multi dimensional interpolation python 2021* *https://github.com/casus/minterpy/releases*

- [2] *Suryanarayana Maddu et al 2021 Mach. Learn.: Sci. Technol. in press* *https://doi.org/10.1088/2632-2153/ac3712*

- [3] *Kharazmi E. et al 2021,* *Computer Methods in Applied Mechanics and Engineering in press* *https://doi.org/10.1016/j.cma.2020.113547*