Message Passing Neural PDE Solvers

ICLR 2022

Johannes Brandstetter*[1,2]
Daniel Worrall*[3]
Max Welling[1]

(1) University of Amsterdam
(2) Johannes Kepler University Linz
(3) Qualcomm AI Research

Partial Differential Equations (PDEs) of the form

$$\partial_t \mathbf{u} = F(t, \mathbf{x}, \mathbf{u}, \partial_{\mathbf{x}} \mathbf{u}, \partial_{\mathbf{xx}} \mathbf{u}, ...) \tag{1}$$

are abundant, yet numerical PDE solving is a **splitter field**:

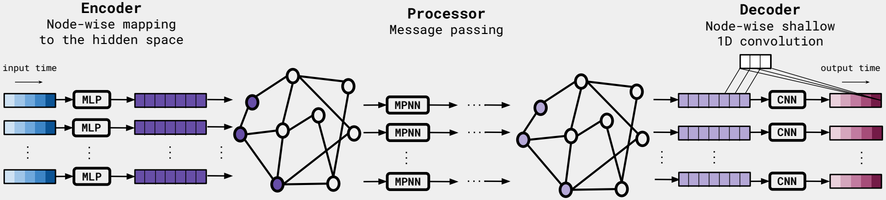
- User requirements, structural requirements, implementation requirements

Goal is to design a **fully numerical PDE solver** which offers flexibility to satisfy as many requirements as possible.

Common way to solve PDEs is to approximate **spatial derivatives** and solve for **temporal derivatives**.

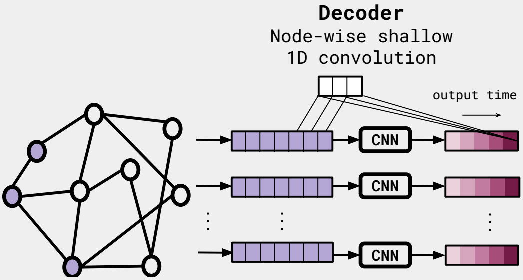
Message passing neural network to update $\mathbf{u}(\mathbf{x}, t) \rightarrow \mathbf{u}'(\mathbf{x}, t')$:

- $\mathbf{u}_i^t$ is a node in the graph with coordinates $\mathbf{x}_i$.
- $\text{Message}_{\text{edge } j \rightarrow i}: \ \mathbf{m}_{ij}^m = \phi\left(\mathbf{f}_i^m, \mathbf{f}_j^m, \mathbf{u}_i^t - \mathbf{u}_j^t, \mathbf{x}_i - \mathbf{x}_j, \theta_{\text{PDE}}\right)$.
- Generalizes **estimation of spatial derivatives**.
- Finite difference, finite volume and WENO scheme are representationally contained (if one, two, or three message passing layers are used).



Decoder is a shallow 1D convolutional network with shared weights across spatial locations.

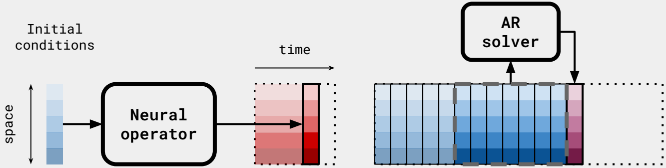
- Smoothes signal over time.
- Reminiscent of linear multistep methods (**temporal update**).



Challenges for autoregressive solvers

Solving PDEs iteratively gives strong physical interpretability, however:

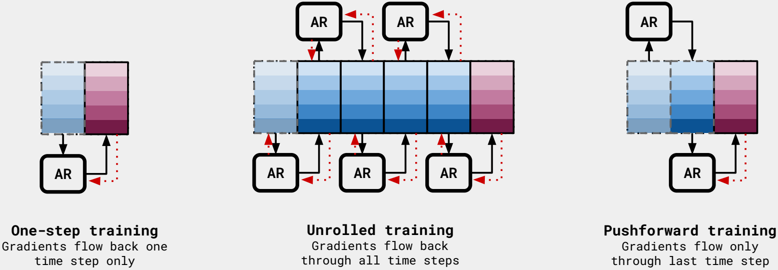
- Hard to train since errors at test time accumulate.
- How to enforce stability? How to simulate error input distribution in training?



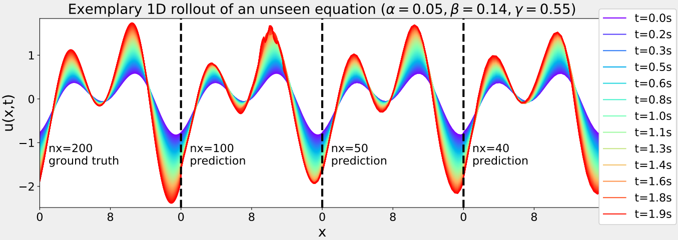
Neural operator
Mapping from initial conditions to output

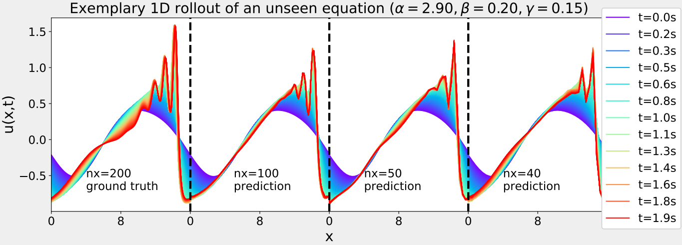
Autoregressive model
Mapping between temporally consecutive time steps

- **Pushforward trick:** mimics distribution shift via adversarial perturbation.
- **Temporal bundling**: synchronous prediction of multiple future timesteps.

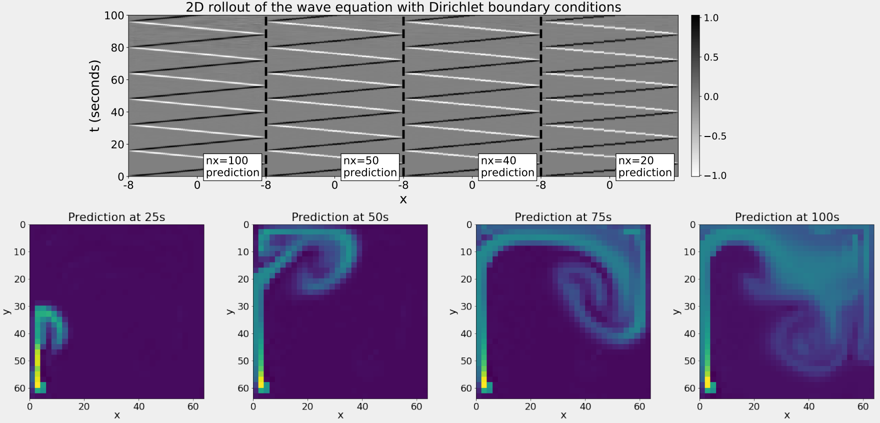


Generalization across different equations, different resolutions





Generalization across boundary conditions, irregular grids, applicability to higher dimensional problems



Poster: 7134

Paper: Message Passing Neural PDE Solvers *arXiv:2202.03376*

Code: *https://github.com/brandstetter-johannes/MP-Neural-PDE-Solvers*