Learning WENO for entropy stable schemes to solve conservation laws

Deep Ray
Department of Mathematics
University of Maryland, College Park

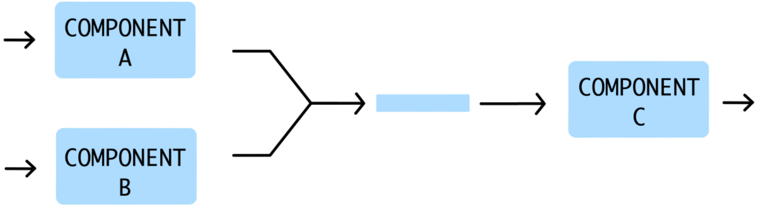
Email: deepray@umd.edu
Website: deepray.github.io

Joint work with Philip Charles

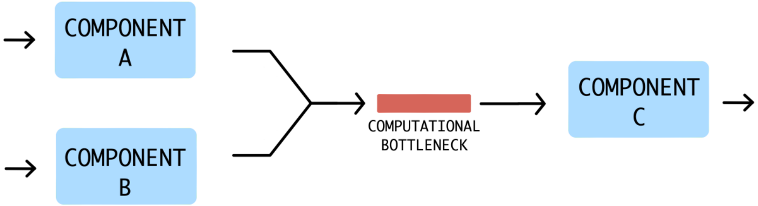
BRIN Workshop on SciML
February 23, 2024

UNIVERSITY OF MARYLAND

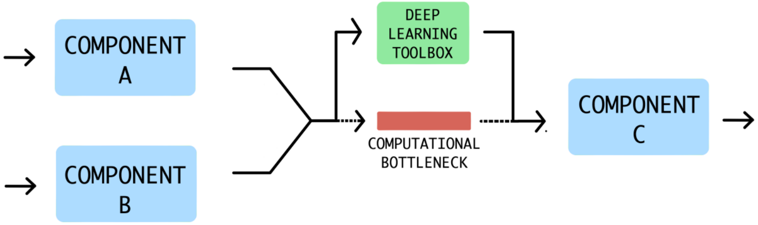
- Efficient, mathematically sound numerical algorithms for solving problems.



- Efficient, mathematically sound numerical algorithms for solving problems.
- Often these methods contain computational bottlenecks.



- Efficient, mathematically sound numerical algorithms for solving problems.
- Often these methods contain computational bottlenecks.
- **Idea:** Replace bottleneck by deep learning toolbox.



Deep learning-based numerical enhancements: a synergistic approach!

Outline

- Conservation laws and entropy conditions
- TeCNO & the sign property
- SP-WENO reconstruction and issues
- A deep learning-based SP-WENO
- Numerical results
- Conclusion

Consider the PDE (in 1D for simplicity)

$$\frac{\partial \boldsymbol{u}(x,t)}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u}(x,t))}{\partial x} = 0$$
$$\boldsymbol{u}(x,0) = \boldsymbol{u}_0(x)$$

- Shallow water equations
- Euler equations
- MHD equations

Non-linearity of $\boldsymbol{f}$

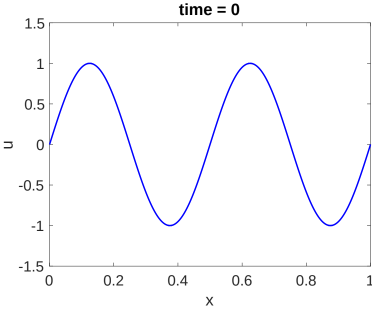
$\Longrightarrow$

Discontinuities in finite time

$\Longrightarrow$

Consider weak (distributional) solutions

Weak solutions need not be unique!



Consider the PDE (in 1D for simplicity)

$$\frac{\partial \boldsymbol{u}(x,t)}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u}(x,t))}{\partial x} = 0$$
$$\boldsymbol{u}(x,0) = \boldsymbol{u}_0(x)$$

- Shallow water equations
- Euler equations
- MHD equations

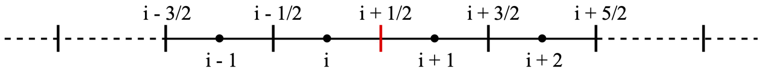
Assume PDE is equipped with entropy-entropy flux pair $\big(\eta(\boldsymbol{u}), q(\boldsymbol{u})\big)$ satisfying

$$\partial_{\boldsymbol{u}} q(\boldsymbol{u}) = \boldsymbol{v}^\top \partial_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{u})$$
$$\boldsymbol{v}(\boldsymbol{u}) = \partial_{\boldsymbol{u}} \eta(\boldsymbol{u}) \quad \rightarrow \quad \text{(entropy variables)}$$

Entropy condition: To pick a physically relevant weak solution

$$\boxed{\frac{\partial \eta(\boldsymbol{u})}{\partial t} + \frac{\partial \boldsymbol{q}(\boldsymbol{u})}{\partial x} \leq 0}$$

Existence, uniqueness of solutions for scalar conservation laws [Kruzkov, 1970].



Consider the one-dimensional setting ($d = 1$). Discretize (uniformly) the domain $\Omega = \bigcup_i I_i$, where

$$I_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] = [x_i - \frac{h}{2}, x_i + \frac{h}{2}]$$

Consider the semi-discrete finite difference scheme

$$\frac{\mathrm{d}\boldsymbol{u}_i(t)}{\mathrm{d}t} + \frac{1}{h}\left(\boldsymbol{f}_{i+\frac{1}{2}} - \boldsymbol{f}_{i-\frac{1}{2}}\right)$$

where:

$\boldsymbol{u}_i \rightarrow$ approximation of $\boldsymbol{u}(x_i, t)$

$\boldsymbol{f}_{i+\frac{1}{2}} \rightarrow$ consistent, conservative numerical flux at $x_{i+\frac{1}{2}}$

Interested in schemes satisfying a discrete entropy condition.

TeCNO schemes

A special class of arbitrary high-order entropy stable schemes [Fjordholm et al., 2012] with the flux

$$\boldsymbol{f}_{i+\frac{1}{2}} = \boldsymbol{f}^{*,2p}_{i+\frac{1}{2}} - \frac{1}{2}\boldsymbol{R}_{i+\frac{1}{2}}\Lambda_{i+\frac{1}{2}}[\![\boldsymbol{z}]\!]_{i+\frac{1}{2}}$$

satisfying the discrete entropy condition

$$\frac{\mathrm{d}\eta(\boldsymbol{u}_i)}{\mathrm{d}t} + \frac{1}{h}\left(q_{i+\frac{1}{2}} - q_{i-\frac{1}{2}}\right) \leq 0$$

where $q_{i+\frac{1}{2}}$ is a consistent numerical entropy flux.

A special class of arbitrary high-order entropy stable schemes [Fjordholm et al., 2012] with the flux

$$\boldsymbol{f}_{i+\frac{1}{2}} = \boldsymbol{f}^{*,2p}_{i+\frac{1}{2}} - \frac{1}{2}\boldsymbol{R}_{i+\frac{1}{2}}\Lambda_{i+\frac{1}{2}}[\![\boldsymbol{z}]\!]_{i+\frac{1}{2}}$$

$\boldsymbol{f}^{*,2p}_{i+\frac{1}{2}}$ is a $2p$-th order entropy conservative central flux [Lefloch et al., 2002] that leads to a satisfaction of a discrete entropy equality

$$\frac{\mathrm{d}\eta(\boldsymbol{u}_i)}{\mathrm{d}t} + \frac{1}{h}\left(q^*_{i+\frac{1}{2}} - q^*_{i-\frac{1}{2}}\right) = 0$$

Can be constructed provided a second-order entropy conservative flux is available.

TeCNO schemes

A special class of arbitrary high-order entropy stable schemes [Fjordholm et al., 2012] with the flux

$$\boldsymbol{f}_{i+\frac{1}{2}} = \boldsymbol{f}^{*,2p}_{i+\frac{1}{2}} - \frac{1}{2} \boldsymbol{R}_{i+\frac{1}{2}} \Lambda_{i+\frac{1}{2}} [\![\boldsymbol{z}]\!]_{i+\frac{1}{2}}$$

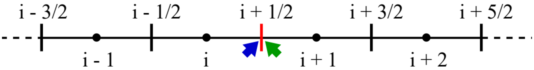
$\boldsymbol{R}_{i+\frac{1}{2}} \rightarrow$ matrix of right eigenvectors of $\partial_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{u})$

$\boldsymbol{\Lambda}_{i+\frac{1}{2}} \rightarrow$ non-negative diagonal matrix of depending on eigenvalues of $\partial_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{u})$

These are evaluated at some averaged solution state at $x_{i+\frac{1}{2}}$.

A special class of arbitrary high-order entropy stable schemes [Fjordholm et al., 2012] with the flux

$$\boldsymbol{f}_{i+\frac{1}{2}} = \boldsymbol{f}^{*,2p}_{i+\frac{1}{2}} - \frac{1}{2}\boldsymbol{R}_{i+\frac{1}{2}}\Lambda_{i+\frac{1}{2}}[\![\boldsymbol{z}]\!]_{i+\frac{1}{2}}$$



At each $x_{i+\frac{1}{2}}$:

- Define the (locally) scaled entropy variables $\boldsymbol{z}_j = \boldsymbol{R}^{\top}_{i+\frac{1}{2}}\boldsymbol{v}_j$.
- Use cell values of $\boldsymbol{z}$ to reconstruct (component-wise) polynomials $\boldsymbol{z}_i(x), \boldsymbol{z}_{i+1}(x)$ in the cells I_i, I_{i+1} respectively.
- Evaluate the interface values and jump at $x_{i+\frac{1}{2}}$:

$$\boldsymbol{z}^{-}_{i+\frac{1}{2}} = \boldsymbol{z}_i(x_{i+\frac{1}{2}}), \qquad \boldsymbol{z}^{+}_{i+\frac{1}{2}} = \boldsymbol{z}_{i+1}(x_{i+\frac{1}{2}}), \qquad [\![\boldsymbol{z}]\!]_{i+\frac{1}{2}} = \boldsymbol{z}^{+}_{i+\frac{1}{2}} - \boldsymbol{z}^{-}_{i+\frac{1}{2}}$$

- In smooth regions

$$[\![\boldsymbol{z}]\!]_{i+\frac{1}{2}} \sim \mathcal{O}(h^k)$$

A special class of arbitrary high-order entropy stable schemes [Fjordholm et al., 2012] with the flux

$$\boldsymbol{f}_{i+\frac{1}{2}} = \boldsymbol{f}^{*,2p}_{i+\frac{1}{2}} - \frac{1}{2}\boldsymbol{R}_{i+\frac{1}{2}}\Lambda_{i+\frac{1}{2}}[\![\boldsymbol{z}]\!]_{i+\frac{1}{2}}$$

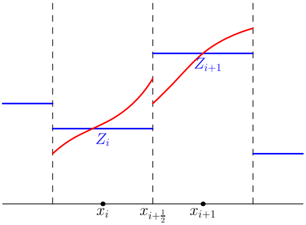
Most importantly, the reconstruction needs to satisfy the sign property.

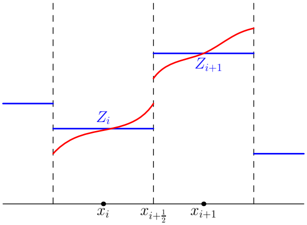
Sign property [Fjordholm et al., 2012]

A reconstruction algorithm used is said to satisfy the sign property if the following condition holds (component-wise) at $x_{i+\frac{1}{2}}$

$$\mathrm{sign}([\![z]\!]_{i+\frac{1}{2}}) = \mathrm{sign}(\Delta z_{i+\frac{1}{2}})$$

where $\Delta z_{i+\frac{1}{2}} = z_{i+1} - z_i$.





Only a handful reconstruction are known to have this property!

Second-order reconstruction with minmod limiter

$$
\begin{aligned}
z^-_{i+\frac{1}{2}} &= v_i + \frac{1}{2}\text{minmod}(\Delta z_{i+\frac{1}{2}}, \Delta z_{i-\frac{1}{2}}) \\
z^+_{i+\frac{1}{2}} &= v_{i+1} - \frac{1}{2}\text{minmod}(\Delta z_{i+\frac{3}{2}}, \Delta z_{i+\frac{1}{2}})
\end{aligned}
$$

where

$$
\text{minmod}(a,b) = \begin{cases} \text{sign}(a)\min\left(|a|,|b|\right), & \text{if } \text{sign}(a) = \text{sign}(b) \\ 0, & \text{otherwise} \end{cases}
$$

ENO interpolation [Fjordholm et al., 2013]
Construct k-th degree polynomial by adaptively choosing the stencil.

Third-order weighted ENO (WENO) interpolation
SP-WENO [Fjordholm and R., 2016], SP-WENOc [R., 2018].

Third-order reconstruction [Cheng and Nie, 2016]
Quadratic polynomial + limiter.

ENO idea: To construct a polynomial $z_i(x)$ in cell I_i, consider k stencils (each with k cells) and pick the stencil where the polynomial would be the smoothest, i.e., away from discontinuities.

Disadvantages:

- Consider $2k-1$ cells but finally only use k cells.
- Accuracy issues due to linear instabilities [Rogerson and Meiburg, 1990].

ENO vs WENO

ENO idea: To construct a polynomial $z_i(x)$ in cell I_i, consider k stencils (each with k cells) and pick the stencil where the polynomial would be the smoothest, i.e., away from discontinuities.

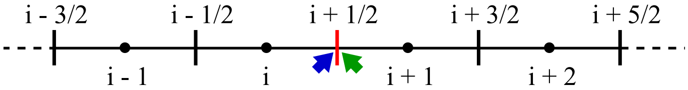
Disadvantages:

- Consider $2k-1$ cells but finally only use k cells.
- Accuracy issues due to linear instabilities [Rogerson and Meiburg, 1990].

WENO idea: Weighted combination of all k-th order candidate polynomials in ENO to achieve $(2k-1)$-th order accuracy.

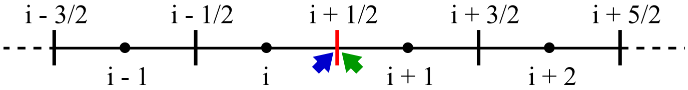
Goals: Weights of WENO should:

- Give $(2k-1)$-th order accuracy in smooth regions.
- Adapt near discontinuities.
- Ensure the sign property is satisfied.



Reconstruction from the left: Using candidate stencils $\{x_i, x_{i+1}\}$ and $\{x_{i-1}, x_i\}$

$$z^-_{i+\frac{1}{2}} = w_0 z_i^{(0)}(x_{i+\frac{1}{2}}) + w_1 z_i^{(1)}(x_{i+\frac{1}{2}})$$



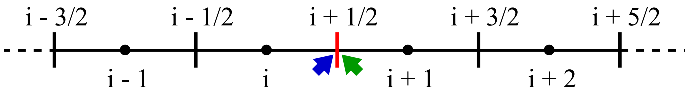
Reconstruction from the left: Using candidate stencils $\{x_i, x_{i+1}\}$ and $\{x_{i-1}, x_i\}$

$$z^-_{i+\frac{1}{2}} = w_0 z_i^{(0)}(x_{i+\frac{1}{2}}) + w_1 z_i^{(1)}(x_{i+\frac{1}{2}})$$

Reconstruction from the right: Using candidate stencils $\{x_i, x_{i+1}\}$ and $\{x_{i+1}, x_{i+2}\}$

$$z^+_{i+\frac{1}{2}} = \tilde{w}_0 z_{i+1}^{(0)}(x_{i+\frac{1}{2}}) + \tilde{w}_1 z_{i+1}^{(1)}(x_{i+\frac{1}{2}})$$

Third-order SP-WENO



Need to pick weights $w_0, w_1, \tilde{w}_0, \tilde{w}_1$ such that:

- **Consistency:**

$$w_0 + w_1 = 1, \quad \tilde{w}_0 + \tilde{w}_1 = 1, \quad w_0, w_1, \tilde{w}_0, \tilde{w}_1 \geq 0.$$

- Satisfy the **sign property**, i.e.,

$$\boxed{\tilde{w}_0(1 - \theta^-) + w_1(1 - \theta^+) \geq 0}$$

where the jump ratios are

$$\theta^- = \frac{\Delta z_{i+\frac{3}{2}}}{\Delta z_{i+\frac{1}{2}}} = \frac{z_{i+2} - z_{i+1}}{z_{i+1} - z_i}, \quad \theta^+ = \frac{\Delta z_{i-\frac{1}{2}}}{\Delta z_{i+\frac{1}{2}}} = \frac{z_{i+2} - z_{i+1}}{z_{i+1} - z_i}$$

- **Negation symmetry:** The weights remain unchanged if $z \mapsto -z$.

Two variants of SP-WENO were hand-crafted [Fjordholm and R., 2016; R., 2018] satisfying:

- All the above properties
- Third-order accuracy for smooth solutions

However when used with TeCNO schemes, lead to spurious Gibbs oscillations near discontinuities.

Reason: Reconstructions can result in a jump $[\![\boldsymbol{z}]\!]_{i+\frac{1}{2}} \approx 0$. Thus

$$\boldsymbol{f}^{*,2p}_{i+\frac{1}{2}} - \frac{1}{2}\boldsymbol{R}_{i+\frac{1}{2}}\Lambda_{i+\frac{1}{2}}[\![\boldsymbol{z}]\!]_{i+\frac{1}{2}} \xrightarrow{\text{reduces to}} \boldsymbol{f}^{*,2p}_{i+\frac{1}{2}} \quad \text{(central flux)}$$

i.e., no dissipation!

This is okay in smooth regions, but not near discontinuities.

Strategy: Use a deep neural network to predict the WENO weights such that

- All important properties are strongly imposed.
- Learn from data how to correctly reconstruct near smooth regions and discontinuities.
- Achieve this a model agnostic manner.

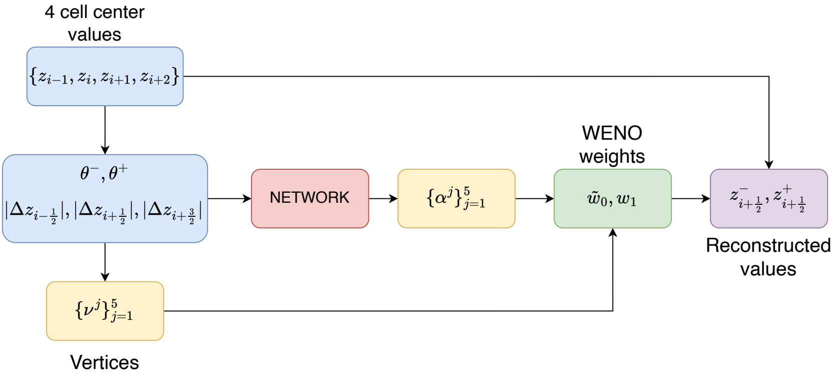
Key ingredients:

- Enough to determine $\tilde{w}_0$ and w_1
- Feasible region $\mathcal{R} \in [0,1]^2$: if $(\tilde{w}_0, w_1) \in \mathcal{R}$ then
  - Sign property holds
  - Third-order accuracy in smooth regions.
- Values of θ^+ and θ^- create 6 cases
- In each case, $\mathcal{R}$ is a convex polygon with 5 vertices $\{\nu^j\}_{j=1}^5$ in $[0,1]^2$.

Goal: Design a network that estimated the convex weights $\{\alpha^j\}_{j=1}^5$ for the vertices such that

$$\tilde{w}_0 = \sum_{j=1}^{5} \alpha^j \nu_1^j \quad w_1 = \sum_{j=1}^{5} \alpha^j \nu_2^j$$

leads to accurate WENO reconstructions for both smooth and discontinuous functions.



Network trained by minimizing the (left and right) interface mismatch error

$$\left|z^-_{i+\frac{1}{2}} - z(x^-_{i+\frac{1}{2}})\right| + \left|z^+_{i+\frac{1}{2}} - z(x^+_{i+\frac{1}{2}})\right|$$

averaged over all training samples.

Network architecture: Feedforward network with 3 hidden layers of with 5 each.

Training dataset: Samples generated using the following functions

No.	Type	$z(x)$	Parameters
1	Smooth	$ax^3 + bx^2 + cx + d$	$a, b, c, d \in \mathbb{U}[-10, 10]$
2	Smooth	$(x-a)(x-b)(x-c)+d$	$a, b, c, d \in \mathbb{U}[-2, 2]$
3	Smooth	$\sin(a\pi x + b)$	$a, b \in \mathbb{U}[-2, 2]$
4	Discontinuous	$\begin{cases} ax + b \text{ if } x \leq 0.5 \\ cx + d \text{ if } x > 0.5 \end{cases}$	$a, b, c, d \in \mathbb{U}[-5, 5]$

50,000 smooth samples and 50,000 discontinuous samples created.

No solutions to conservation laws used!

- TeCNO scheme with fourth-order entropy conservative flux $\boldsymbol{f}^{*,4}_{i+\frac{1}{2}}$.
- Third-order SSP-RK3 [Gottlieb et al, 2001] to integrate semi-discrete scheme.
- Reconstructions considered:
  - Second-order ENO-2 (4 cells per $x_{i+\frac{1}{2}}$)
  - Third-order ENO-3 (6 cells per $x_{i+\frac{1}{2}}$)
  - Third-order SP-WENO (4 cells per $x_{i+\frac{1}{2}}$)
  - Third-order SP-WENOc (4 cells per $x_{i+\frac{1}{2}}$)
  - New third-order SP-WENO-DL (4 cells per $x_{i+\frac{1}{2}}$)

Linear advection: Smooth data

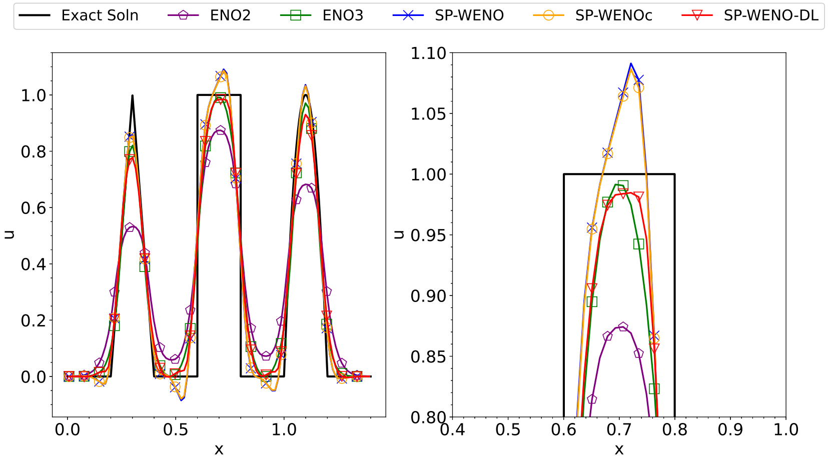
Solving $\partial_t u + \partial_x u = 0$ on $[-\pi, \pi]$ till $T = 0.5$ with periodic BC and
Test 1: $u_0 = \sin(x)$
Test 2: $u_0 = \sin^4(x)$

	N	ENO3 L^1_h Error	ENO3 Rate	SP-WENO L^1_h Error	SP-WENO Rate	SP-WENOc L^1_h Error	SP-WENOc Rate	SP-WENO-DL L^1_h Error	SP-WENO-DL Rate
Test 1	100	3.23e-5	-	6.90e-5	-	6.80e-5	-	1.66e-4	-
	200	4.04e-6	3.00	7.65e-6	3.17	7.48e-6	3.18	3.58e-5	2.21
	400	5.05e-7	3.00	8.29e-7	3.20	8.17e-7	3.20	4.57e-6	2.97
	600	1.50e-7	3.00	2.26e-7	3.20	2.23e-7	3.20	1.35e-6	3.02
	800	6.31e-8	3.00	8.72e-8	3.31	8.60e-8	3.31	5.72e-7	2.97
	1000	3.23e-8	3.00	4.21e-8	3.27	4.15e-8	3.26	2.95e-7	2.97
Test 2	100	1.48e-3	-	1.52e-3	-	1.46e-3	-	1.87e-3	-
	200	1.98e-4	2.91	1.68e-4	3.18	1.68e-4	3.12	2.61e-3	2.84
	400	2.58e-5	2.94	1.79e-5	3.23	1.78e-5	3.23	3.35e-5	2.96
	600	8.25e-6	2.81	4.69e-6	3.31	4.70e-6	3.29	9.59e-6	3.08
	800	4.64e-6	2.00	1.81e-6	3.31	1.80e-6	3.33	3.93e-6	3.10
	1000	3.46e-6	1.31	8.64e-7	3.32	8.61e-7	3.31	2.03e-6	2.96

Note:

- Deterioration of accuracy with ENO3 in Test 2
- SP-WENO and SP-WENOc have accuracy > 3 – jump vanishing
- SP-WENO-DL more dissipative but third-order

Solved on $[0, 1.4]$ till $T = 1.4$ with periodic BC



Under/overshoots with SP-WENO and SP-WENOc

The full 3D model:

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho \\ \rho\mathbf{u} \\ E\end{bmatrix} + \nabla\cdot\begin{bmatrix}\rho\mathbf{u} \\ p\mathbf{I} + \rho(\mathbf{u}\otimes\mathbf{u}) \\ (E+p)\mathbf{u}\end{bmatrix} = \mathbf{0},$$

Total energy

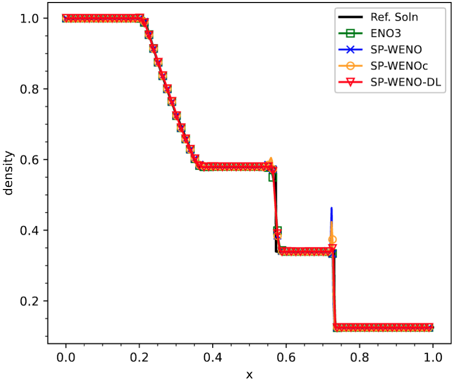
$$E = \rho\left(\frac{|\mathbf{u}|^2}{2} + e\right).$$

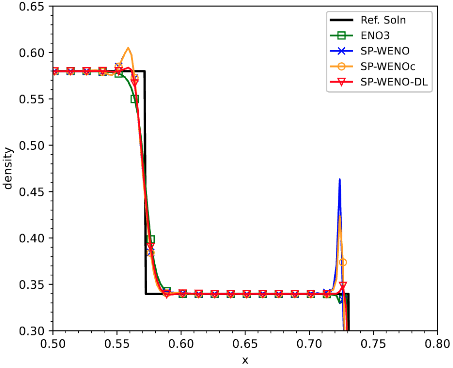
with the internal energy e given by EOS

$$e = \frac{p}{(\gamma-1)\rho}$$

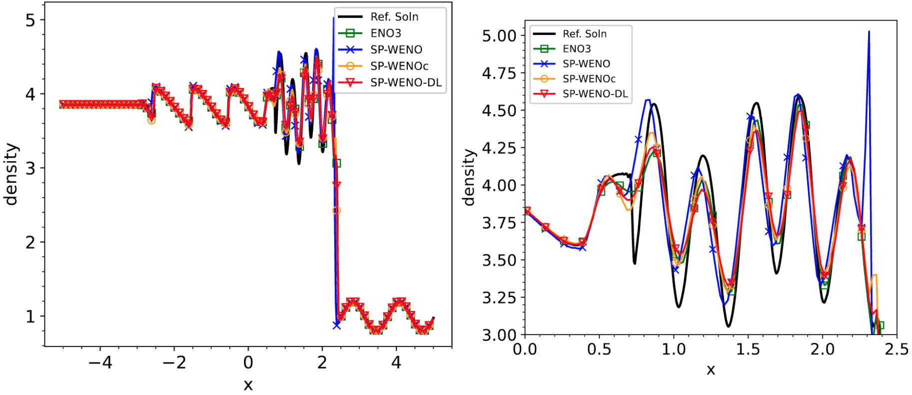
where $\gamma = 1.4$ is the ratio of specific heats.

Solved on $[0, 1]$ till $T = 0.2$ with Neumann BC. $N = 400$

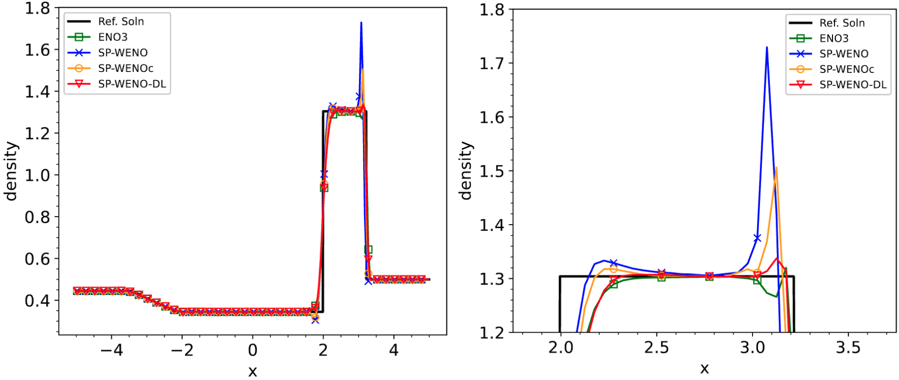




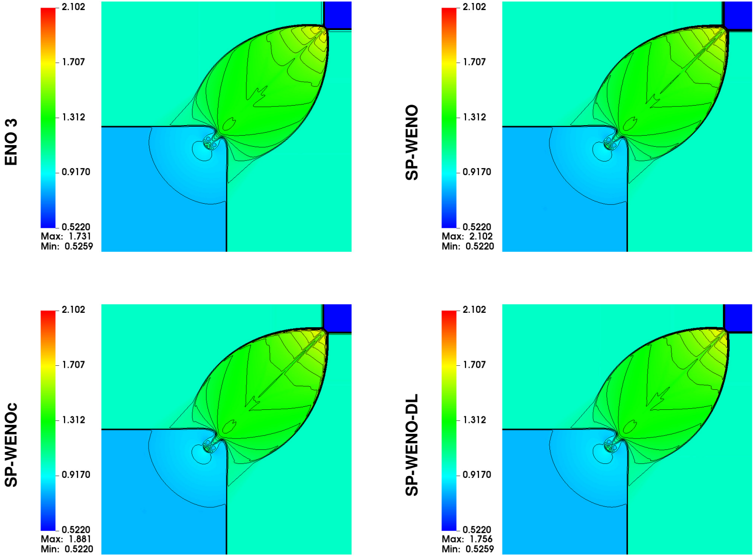
Solved on $[-5, 5]$ till $T = 1.8$ with Neumann BC. $N = 400$



Solved on $[-5, 5]$ till $T = 1.3$ with Neumann BC. $N = 200$



Solved on $[0,1] \times [0,1]$ till $T = 0.25$ with Neumann BC using 200×200 cells



Conclusion

- Demonstrated a data-driven approach to learn reconstruction algorithms.
- Trained a network to learn WENO weights.
- Strong embedding of structural properties, such as the sign property.
- Network agnostic of any specific conservation model – single network for all models.
- Performs better than existing SP-WENO variants.
- Next steps: higher-order SP-WENO-DL, reconstruction on unstructured grids, hybrid schemes.

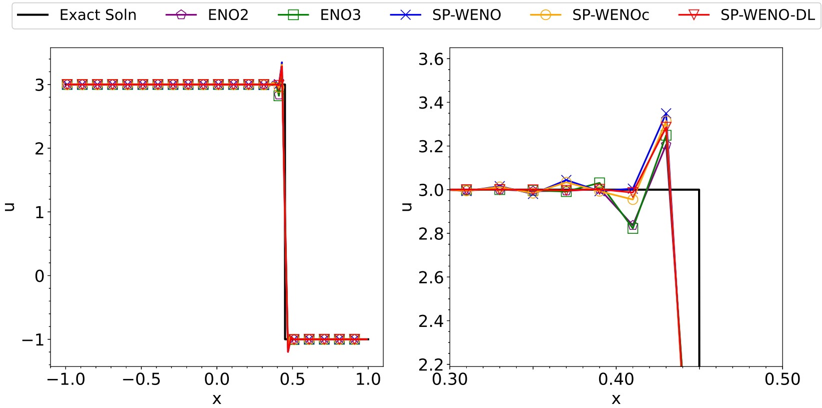
Conclusion

- Demonstrated a data-driven approach to learn reconstruction algorithms.
- Trained a network to learn WENO weights.
- Strong embedding of structural properties, such as the sign property.
- Network agnostic of any specific conservation model – single network for all models.
- Performs better than existing SP-WENO variants.
- Next steps: higher-order SP-WENO-DL, reconstruction on unstructured grids, hybrid schemes.

Questions?

Burgers equation: isolated shock wave

Solving $\partial_t u + \partial_x u^2/2 = 0$ on $[-1, 1]$ till $T = 0.5$ with Neumann BC. $N = 100$



Under/overshoots with all methods! But better profile with SP-WENO-DL prior to the shock.