Entropy Stable Conservative Flux Form Neural Nets: Flux Term Recovery for Pulse Propagation in Fibers

Lizuo Liu (Dartmouth College), Anne Gelb (Dartmouth College) and Adityavikram Viswanathan (University of Michigan, Dearborn)

2025 AFOSR EM Annual PI Review, January 7, 2025

Problem of interest

For pulse propagation in fibers, we often focus on the wave equation derived from Maxwell's equations in a dielectric medium:

$$\nabla^2 \mathbf{E} - \frac{1}{c^2}\frac{\partial^2 \mathbf{E}}{\partial t^2} = \mu_0 \frac{\partial^2 \mathbf{P}}{\partial t^2}.$$

- c is the speed of light in vacuum.
- μ_0 is the permeability of free space.
- $\mathbf{P}$ represents the nonlinear polarization for fibers.

Problem of interest

$$
\begin{aligned}
\nabla \cdot \mathbf{D} &= \rho \\
\nabla \cdot \mathbf{B} &= 0 \\
\nabla \times \mathbf{E} &= -\frac{\partial \mathbf{B}}{\partial t} \\
\nabla \times \mathbf{H} &= \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}.
\end{aligned}
$$

- $\mathbf{E}$ is the electric field.
- $\mathbf{D}$ is the electric displacement field ($\mathbf{D} = \epsilon \mathbf{E} + \mathbf{P}$, where $\mathbf{P}$ is the polarization).
- $\mathbf{H}$ is the magnetic field intensity.
- $\mathbf{B}$ is the magnetic flux density.
- ρ is the charge density.
- $\mathbf{J}$ is the current density.

Energy Conservation in Nonlinear Maxwell's Equations

The total electromagnetic energy density in a nonlinear medium is given by

$$u = \frac{1}{2}\left(\epsilon|\mathbf{E}|^2 + \mu|\mathbf{H}|^2\right).$$

- $\epsilon = \epsilon_0 + \epsilon_{\mathrm{NL}}$ is the permittivity, incorporating nonlinear effects.
- μ is the permeability of the medium.

The Poynting vector, representing the energy flux, is

$$\mathbf{S} = \mathbf{E} \times \mathbf{H}.$$

Applying Maxwell's equations yields

$$\nabla \cdot \mathbf{S} = -\frac{\partial u}{\partial t}.$$

Energy conservation is therefore expressed as

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{S} = 0.$$

That is, the rate of change of energy density equals the divergence of the energy flux.

Momentum Conservation in Nonlinear Maxwell's Equations

The momentum density of the electromagnetic field is

$$\mathbf{g} = \frac{\mathbf{E} \times \mathbf{B}}{\mu_0}.$$

The stress tensor for the electromagnetic field is given by

$$\mathbf{T}_{ij} = \epsilon \left(E_i E_j - \frac{1}{2} |\mathbf{E}|^2 \delta_{ij} \right) + \frac{1}{\mu} \left(B_i B_j - \frac{1}{2} |\mathbf{B}|^2 \delta_{ij} \right).$$

The conservation law for momentum is derived as

$$\frac{\partial \mathbf{g}}{\partial t} + \nabla \cdot \mathbf{T} = \mathbf{f}.$$

- **g** is the momentum density.
- **T** is the stress tensor.
- **f** represents external forces, such as those arising from nonlinear polarization.

In a source-free, lossless medium

$$\frac{\partial \mathbf{g}}{\partial t} + \nabla \cdot \mathbf{T} = 0.$$

Ultimate Goal: Given data for **E** and **H** can we discover the conservation law?

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{S} = 0,$$
$$\frac{\partial \mathbf{g}}{\partial t} + \nabla \cdot \mathbf{T} = 0.$$

1. What is the proper representation of the hyperbolic conservation law?
2. What is the algebraic relation from observables **E**, **H** to conserved variables u, $\mathbf{g}$?
3. Automatic inference?
4. Fast and accurate algorithm?

Conservative Flux Neural Network

Using the semidiscrete scheme to learn the conservation laws (Burgers Equation, Shallow water Equation, Euler Equation etc.) from noisy data

$$\frac{d\boldsymbol{u}}{dt} + \frac{d\boldsymbol{f}}{dx} = \boldsymbol{0}, \quad \frac{d\boldsymbol{u}_j(t)}{dt} + \frac{1}{\Delta x}\left[\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}\right] = \boldsymbol{0},$$

with the flux term $\boldsymbol{f}_{j+\frac{1}{2}}$ approximated by a neural network $\boldsymbol{f}_\theta\left(\boldsymbol{u}_{j-p}^n, \ldots, \boldsymbol{u}_{j+q}^n\right)$:

- Solve the DNN-involved system
- Compute the loss between prediction and training data.
- Optimize it.

Zhen Chen, Anne Gelb, and Yoonsang Lee. Designing neural networks for hyperbolic conservation laws, 2022, arxiv:2211.14375

Conservative Flux Neural Network

$$\frac{d\boldsymbol{u}_j(t)}{dt} + \frac{1}{\Delta x}\left[\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}\right] = \mathbf{0},$$

with the flux term $\boldsymbol{f}_{j+\frac{1}{2}}$ approximated by a neural network $\boldsymbol{f}_\theta\left(\boldsymbol{u}^n_{j-p}, \ldots, \boldsymbol{u}^n_{j+q}\right)$

Pros:

- Effective in extracting the conservation laws from noisy data
- Easy to implement

Cons:

- Might introduce spurious oscillation
- No entropy stability guarantee

Target of this work:

- Control the spurious oscillations
- Try to make scheme entropy stable

Methods

$$\frac{d\boldsymbol{u}_j(t)}{dt} + \frac{1}{\Delta x}\left[\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}\right] = \boldsymbol{0},$$

Target of this work:

- Control the spurious oscillations: Slope Limiter, Flux Limiter, etc.
- Try to make scheme entropy stable: Entropy Stable Scheme

Eitan Tadmor. Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems. Acta Numerica, 12:451–512, 2003

Kurganov Tadmor Scheme

$$\frac{d}{dt}u_j(t) = -\frac{H_{j+1/2}(t) - H_{j-1/2}(t)}{\Delta x}$$

Local Lax Friedrichs Flux

$$H_{j+1/2}(t) := \frac{f\left(u^+_{j+1/2}(t)\right) + f\left(u^-_{j+1/2}(t)\right)}{2} - \frac{a_{j+1/2}(t)}{2}\left[u^+_{j+1/2}(t) - u^-_{j+1/2}(t)\right]$$

$$a_{j+1/2}(t) := \max\left\{\rho\left(\frac{\partial f}{\partial u}\left(u^+_{j+1/2}(t)\right)\right), \rho\left(\frac{\partial f}{\partial u}\left(u^-_{j+1/2}(t)\right)\right)\right\}$$

The intermediate values $u^{\pm}_{j+1/2}$ are given by

$$u^+_{j+1/2} := u_{j+1}(t) - \frac{\Delta x}{2}\left(u_x\right)_{j+1}(t), \quad u^-_{j+1/2} := u_j(t) + \frac{\Delta x}{2}\left(u_x\right)_j(t)$$

Slope Limiter: the term $(u_x)_j$ is approximated by

$$(u_x)_j \approx \text{minmod}\left(\frac{u_{j+1} - u_j}{\Delta x}, \frac{u_j - u_{j-1}}{\Delta x}\right)$$

Alexander Kurganov, Eitan Tadmor, New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection–Diffusion Equations, Journal of Computational Physics, Volume 160, Issue 1, 2000

Kurganov Tadmor Scheme Enhanced Conservative Flux Net

Central idea: Replace $\boldsymbol{f}(\boldsymbol{u})$ by neural network

$$\cdots$$

with the numerical flux

$$\boldsymbol{H}_{j+1/2}(t) := \frac{\boldsymbol{f}_\theta\left(\boldsymbol{u}^+_{j+1/2}(t)\right) + \boldsymbol{f}_\theta\left(\boldsymbol{u}^-_{j+1/2}(t)\right)}{2} - \frac{a_{j+1/2}(t)}{2}\left[\boldsymbol{u}^+_{j+1/2}(t) - \boldsymbol{u}^-_{j+1/2}(t)\right],$$

where

$$a_{j+1/2}(t) := \max\left\{\rho_W\left(\frac{\partial \boldsymbol{f}_\theta}{\partial \boldsymbol{u}}\left(\boldsymbol{u}^+_{j+1/2}(t)\right)\right), \rho_W\left(\frac{\partial \boldsymbol{f}_\theta}{\partial \boldsymbol{u}}\left(\boldsymbol{u}^-_{j+1/2}(t)\right)\right)\right\}$$

is the maximum wave speed.

$$\cdots$$

where $\rho_W(J)$ is another neural network.

Test Problem: 1D Euler's equations

$$\rho_t + (\rho u)_x = 0,$$
$$(\rho u)_t + \left(\rho u^2 + p\right)_x = 0,$$
$$(E)_t + (u(E+p))_x = 0.$$

The spatial domain is $(-5, 5)$. We assume Dirichlet boundary conditions. Initial conditions given by

$$\rho(x,0) = \begin{cases} \rho_l, & \text{if } x \le x_0, \\ 1 + \varepsilon \sin(5x), & \text{if } x_0 < x \le x_1, \\ 1 + \varepsilon \sin(5x) e^{-(x-x_1)^4} & \text{otherwise,} \end{cases} \quad u(x,0) = \begin{cases} u_l, & \text{if } x \le x_0, \\ 0, & \text{otherwise,} \end{cases}$$

$$p(x,0) = \begin{cases} p_l, & \text{if } x \le x_0, \\ p_r, & \text{otherwise,} \end{cases} \quad E(x,0) = \frac{p_0}{\gamma - 1} + \frac{1}{2}\rho(x,0)u(x,0)^2.$$

- 300 training trajectories generated; time step $\Delta t = 0.005s$; spatial grid$\Delta x = \frac{10}{512}$.
- Gaussian noise added to the training data

$$\begin{bmatrix} \tilde{\rho}\,(x_i, t_l) \\ \tilde{\rho u}\,(x_i, t_l) \\ \tilde{E}\,(x_i, t_l) \end{bmatrix} = \begin{bmatrix} \rho\,(x_i, t_l) \\ \rho u\,(x_i, t_l) \\ E\,(x_i, t_l) \end{bmatrix} + \eta \bar{\boldsymbol{u}} \xi_{i,l}, \quad \xi_{i,l} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right),$$

$\bar{\boldsymbol{u}}$ is the mean absolute value of the training data $\boldsymbol{u} = [\rho, \rho u, E]^T$ over the entire dataset up to time $t = t_L$. The noise intensity coefficient $\eta \in [0, 1]$, is chosen as 0, $.25$, $.5$, and 1.

Metrics

- Discrete conserved quantity remainder

$$C(\mathbf{u}(t_l)) := \left| \sum_{j=1}^{n-1} \left(\bar{u}_j(t_l) - \bar{u}_j(t_0) \right) \Delta x - \sum_{s=1}^{l} \left(F_a^{s-1} - F_b^{s-1} \right) \Delta t \right| ,$$

where $\mathbf{u}(t) = (\bar{u}_0(t), \ldots, \bar{u}_n(t))^T \in \mathbb{R}^n$ is the prediction at time t, and flux terms F_a^{s-1} and F_b^{s-1} are the calculated flux operators at each respective boundary defined by

$$F_a^{s-1} = \frac{1}{\Delta t} \int_{t_{s-1}}^{t_s} f(u(a,t))dt, \quad F_b^{s-1} = \frac{1}{\Delta t} \int_{t_{s-1}}^{t_s} f(u(b,t))dt.$$

- Discrete entropy remainder

$$\mathcal{J}(\mathbf{u}(t_l)) := \left(\sum_{j=1}^{n-1} \left(\bar{U}_j(t_l) - \bar{U}_j(t_0) \right) \Delta x - \sum_{s=1}^{l} \left(\bar{F}_a^{s-1} - \bar{F}_b^{s-1} \right) \Delta t \right) ,$$

We say we have obtained an *entropy-stable* network operator if $\mathcal{J}(\mathbf{u}(t_l)) \leq 0$.

- Relative ℓ_2 prediction error

$$\mathcal{R}(\mathbf{u}(t_l), \mathbf{u}_{true}(t_l)) := \frac{\|\mathbf{u}(t_l) - \mathbf{u}_{true}(t_l)\|_2}{\|\mathbf{u}_{true}(t_l)\|_2} .$$

Results

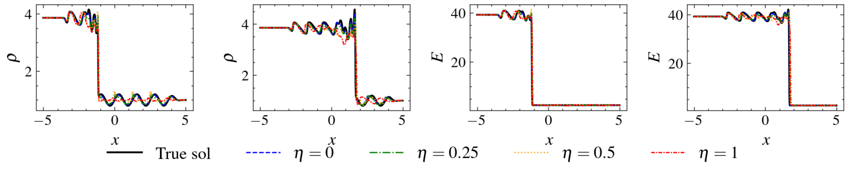

Figure: Comparison of the reference solution (black solid line) of density ρ and energy E in Euler's equation with the KT-enhanced CFN predictions with $\eta = 0, .25, .5, 1$: (left) $t = .8$ of ρ, (middle-left) $t = 1.6$ of ρ, (middle-right) $t = .8$ of E, (right) $t = 1.6$ of E.

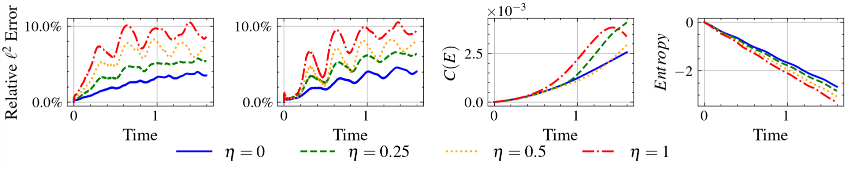

Figure: Relative ℓ_2 prediction error: (left) ρ (middle-left) E, and (middle-right) discrete conserved quantity remainder $C(E)$, and (right) discrete entropy remainder $\mathcal{J}([\rho, \rho u, E]^T)$ for Euler's equation with $\eta = 0, .25, .5, 1$.

Results

$$u_t + \frac{1}{2}\left(u^2\right)_x + \frac{1}{2}\left(u^2\right)_y = 0$$

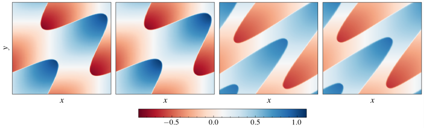

Figure: Comparison of the reference solution of u in 2D Burgers' equation with KT-enhanced CFN predictions for noise coefficient $\eta = 1$: (left) reference solution at t = .4 (800 time steps), (middle-left) predictions at t = .4 (800 time steps), (middle-right) reference solution at t = .8 (1600 time steps), (right) predictions at t = .8 (1600 time steps) with training data for the first 20 time steps only.

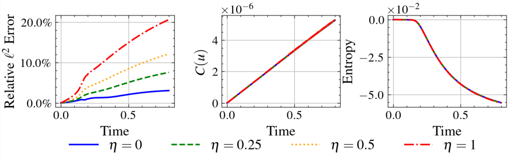

Figure: (left) Relative ℓ^2 prediction error, (middle) Discrete conserved quantity remainder $C(u)$, (right) discrete entropy remainder $\mathcal{J}(u)$ for 2D Burgers' equation, $t \in [0, .8]$, and $\eta = 0, 0.25, 0.5, 1$.

Conclusions

Ultimate Goal: Given data for **E** and **H** can we discover the conservation law?

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{S} = 0,$$
$$\frac{\partial \mathbf{g}}{\partial t} + \nabla \cdot \mathbf{T} = 0.$$

1. What is the proper representation of the hyperbolic conservation law?
2. What is the algebraic relation from observables **E**, **H** to conserved variables u, $\mathbf{g}$?
3. Automatic inference?
4. Fast and accurate algorithm?

- Entropy Stable CFN: High order information stablize the prediction. Real Data, Combination with Discountinuous Galerkin,. . .
- Applications for EM Waves: learning relations of **P** and **E** and the underlining conservation law in fibers.

Reference

- Lizuo Liu, Tongtong Li, Anne Gelb, and Yoonsang Lee. *Entropy Stable Conservative Flux Form Neural Networks*. 2024. arXiv, https://arxiv.org/abs/2411.01746.

Lizuo.Liu@dartmouth.edu