# TRACKING THE DISCONTINUITY: EXPLICIT FRONT TRACKING FOR NEURAL EULER SOLVERS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Physics-informed neural networks smear shocks because of their objective, not their optimiser: across a sharp shock whose two states satisfy the Rankine-Hugoniot conditions, the pointwise strong-form residual scales as the inverse of the width, so its squared norm is minimised by spreading the jump. We develop explicit front tracking for the compressible Euler equations, in which smooth networks are separated by fronts that move by their jump conditions and the residual is never evaluated at a discontinuity. Four components make this work for a system with several interacting wave families. The number, characteristic family and admissibility of the fronts are derived from the exact star state of the Riemann data, with a contact kept only if it carries strength. Two-dimensional fronts are level sets anchored to the initial discontinuity, with the jump conditions imposed along the learned normal at points carried onto the front by Newton projection. The plateaus beside a shock, which lie in the null space of the residual, are anchored through the jump conditions to the exact far-field states rather than to neighbouring learned ones. And the best of several runs is selected by an uncalibrated physics score, without ground truth. On Riemann problems with discontinuities the method is 8 to 14 times more accurate than strong-form and relaxation networks at the same budget on Sod and 386 to 879 times on a problem of two shocks and a contact, anchoring improves the latter a further sixfold, and selection finds the most accurate of three seeds in 11 of 13 settings, against one in three by chance. The method reproduces a planar oblique shock in two dimensions and, on a nonconservative system, realises the jump condition of each prescribed path, where a standard path-conservative scheme responds with the wrong sign. It does not yet handle a curved front, and a classical solver remains three to four orders of magnitude cheaper whenever a rarefaction is present.

## 1 INTRODUCTION

Physics-informed neural networks (PINNs) fit a smooth parametric function to make a differential equation's residual small at sampled points (Raissi et al., 2019). For nonlinear hyperbolic conservation laws this objective is unsound, because the solutions of interest are discontinuous and the residual is not defined at a discontinuity. The failure is documented in practice (Krishnapriyan et al., 2021; Mao et al., 2020; Abbasi et al., 2025) and explained in theory: for a standing Burgers shock smoothed over a width $\varepsilon$, the $L^2$ residual norm grows as $\varepsilon^{-1/2}$ while the $L^2$ error shrinks as $\varepsilon^{1/2}$ (Chaumet & Giesselmann, 2024). The objective and the error move in opposite directions in the limit that matters.

The mechanism is a matter of scaling. A network is smooth, so it represents a shock moving at speed $\sigma$ as a transition of some width $w$, on which the residual $\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = (\mathbf{F}'(\mathbf{U}) - \sigma I)\, \partial_x \mathbf{U}$ is evaluated exactly by automatic differentiation. Its integral across the layer, $[\mathbf{F}(\mathbf{U})] - \sigma\, [\mathbf{U}]$, vanishes for a Rankine-Hugoniot shock, so a weak form sees nothing wrong. Pointwise, however, it vanishes only where $\partial_x \mathbf{U}$ is an eigenvector of $\mathbf{F}'(\mathbf{U})$ with eigenvalue $\sigma$. A smeared contact meets this everywhere and costs the residual nothing; across a shock of a genuinely nonlinear family it cannot hold, so the residual is of order $[\mathbf{U}]\,/w$ and its squared $L^2$ norm grows as $1/w$. We confirmed this for the Euler equations: smearing the exact Sod shock over a width $w$ and scoring it, the strong-form objective falls as $1/w$, from 42.0 at $w = 2 \times 10^{-4}$ to 0.35 at $w = 2 \times 10^{-2}$. **No schedule, architecture or optimiser can fix an objective whose minimiser is not the solution.**



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Existing responses weaken the objective (De Ryck et al., 2024; Chaumet & Giesselmann, 2024), add dissipation (Coutinho et al., 2023), or change the differentiated variable (Zhou & Ma, 2024). A fourth changes the *ansatz*: represent the solution as smooth functions separated by explicitly tracked discontinuities, so that the residual is evaluated only where the solution is smooth and each discontinuity obeys its jump conditions. The idea is classical front tracking and shock fitting (Glimm et al., 1981; Moretti, 2002). Its neural form, the non-diffusive neural network of Lorin & Novruzi (2024), treats one-dimensional laws and systems, including an Euler Riemann problem, and has been extended to several dimensions with graph-surface fronts (Lorin & Novruzi, 2026). We develop this approach for the compressible Euler equations, where several waves of different families interact. The solution is one $\tanh$ network per smooth region, separated by fronts: in one dimension each front is a learned trajectory $s_k(t)$, and in two it is the zero set of a level set $\varphi_k$ anchored to the initial discontinuity, onto which sample points are carried by Newton projection. The strong-form residual is evaluated only away from the fronts; on them, the Rankine-Hugoniot conditions govern the front speed and a one-sided Lax penalty enforces admissibility. Each component below replaced a choice that failed measurably (Appendix B).

**Contributions of this work.**

- **Front structure from the exact star state.** The number of fronts, their characteristic families and their admissibility conditions follow from the Riemann data with no learned component: fans carry no front, and a contact is kept only if it carries strength (Section 3.2).
- **Level-set fronts for the Euler equations in two dimensions**, anchored to the initial discontinuity, with jump and admissibility conditions imposed along the learned normal at Newton-projected points (Section 3.4). They reproduce a planar oblique shock; a curved front remains open (Section 5.4).
- **Anchoring the residual's null space to data.** Constant states satisfy the residual whatever their value and carry much of the remaining error. Fixing the far field to its exact states and anchoring the adjacent plateaus to it through the jump conditions improves the colliding problem sixfold and tightens the seed spread, whereas anchoring them to a neighbouring learned branch removes the gradient that sets the front speed (Section 5.2).
- **Ground-truth-free run selection** by uncalibrated re-scoring of each trained network, which is needed because runs calibrate their own losses; it picks the most accurate of three seeds in 11 of 13 settings (Sections 3.5 and 5.3).
- **Prescribed nonconservative jump conditions.** Where the jump conditions depend on a path in state space, the tracked front imposes the prescribed one, with response slope $1.00$ across a family of paths where a standard path-conservative scheme has slope $-0.26$ (Appendix K).

Neural shock fitting and its extension to several dimensions are prior work (Section 2); the contributions are the components above and their measured effect. Section 5.5 compares the method with a classical solver, which remains far cheaper whenever a rarefaction is present.

## 2 Related work

**Neural solvers for shocks.** Weak and entropy formulations integrate the residual against test functions, which is finite on discontinuous solutions (De Ryck et al., 2024; Chaumet & Giesselmann, 2024). Artificial viscosity, fixed or adaptive, makes the ansatz accurate for a regularised equation whose shock width is set by the viscosity (Mao et al., 2020; Coutinho et al., 2023). Conservative and finite-volume-inspired losses (Jagtap et al., 2020; Patel et al., 2022; Cassia & Kerswell, 2025) and weighted equations (Liu et al., 2024) capture the shock implicitly (Abbasi et al., 2025). Relaxation networks introduce the flux as an independent field so that the singular terms cancel (Zhou & Ma, 2024). Adaptive sampling (Lu et al., 2021), self-adaptive weights (McClenny & Braga-Neto, 2023), causal weighting (Wang et al., 2024), Fourier features (Tancik et al., 2020) and Kolmogorov-Arnold networks (Liu et al., 2025) act on the optimisation path or the spectral bias; we measured all but self-adaptive weights (Appendices H and I). For operator learning, linear reconstructions cannot represent moving discontinuities efficiently (Lanthaler et al., 2023).

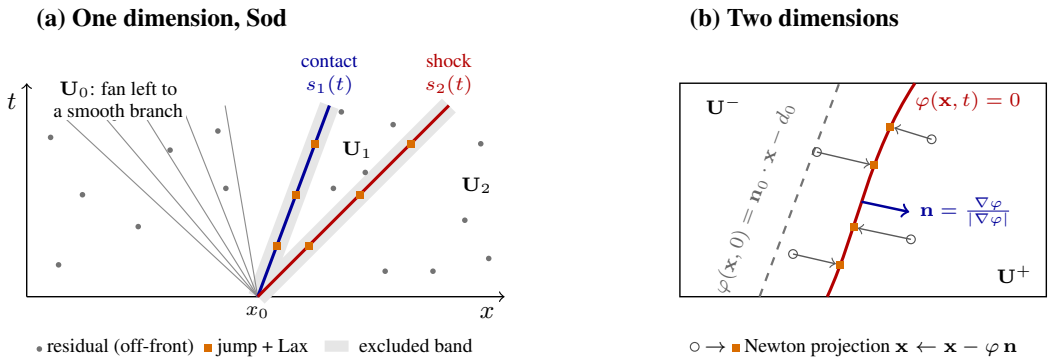



Figure 1: The construction. (a) For Sod, the structure derived from the data gives two fronts (contact, shock) and three smooth branches; the rarefaction has no front. The residual is evaluated away from the fronts, the jump and Lax conditions on them. (b) In two dimensions a front is the zero set of a level set anchored to the initial front; sample points are carried onto it by Newton projection.

**Explicit fronts.** Classical front tracking (Glimm et al., 1981; Chern & Colella, 1987) and shock fitting (Moretti, 2002; Ciallella et al., 2020) represent discontinuities explicitly, and level sets represent moving interfaces as zero sets (Osher & Sethian, 1988), including discontinuities of conservation laws (Aslam, 2001). Neural counterparts represent the smooth regions separately and move each discontinuity by its jump conditions. In one dimension this has been done for scalar laws and systems, including an Euler Riemann problem with a neural initial wave decomposition and Lax conditions (Lorin & Novruzi, 2024), for Burgers' equation (Han et al., 2025) and for two-phase flow (Rehman & Yousuf, 2025). In several dimensions the front has been represented as a graph $x_N = g(x', t)$ over a chosen axis and tested on scalar laws, including shock formation from smooth data, and on shallow water (Lorin & Novruzi, 2026); the method was outlined publicly in 2025 (Pascal et al., 2025). A neural operator for a scalar balance law advances its shocks by the analytical jump law (LeFloch & Xiang, 2026), and lifting to an extra dimension lets one network carry both sides of discontinuities at trainable locations (Sun et al., 2026). An independent reproduction of the one-dimensional method on Burgers' equation did not reproduce shock generation from smooth data (Pascal et al., 2025). We differ in treating the Euler system in two dimensions with a structure derived from the exact star state, in using level sets that fix no axis, and in the evaluation.

**Selection and trivial solutions.** Retraining from several initialisations and selecting by a training-set criterion goes back to supervised learning of flow observables (Lye et al., 2020); for PINNs, keeping the run with the lowest training loss is established practice, justified by stability-based estimates (Mishra & Molinaro, 2023; De Ryck & Mishra, 2024), including for entropy solutions of scalar laws (De Ryck et al., 2024). That constant or trivial states satisfy the residual is known (Leiteritz & Pflüger, 2021; Daw et al., 2023).

## 3 Method

We consider the compressible Euler equations $\partial_t \mathbf{U} + \nabla \cdot \mathbf{F}(\mathbf{U}) = 0$ with $\mathbf{U} = (\rho, \rho\mathbf{u}, E)$ and ideal-gas closure $E = p/(\gamma - 1) + \frac{1}{2}\rho|\mathbf{u}|^2$, $\gamma = 1.4$. A discontinuity with unit normal $\mathbf{n}$ moving with normal speed $\sigma$ between states $\mathbf{U}^-$ and $\mathbf{U}^+$ satisfies the Rankine-Hugoniot (RH) condition

$$\sigma\,[\mathbf{U}] = [\mathbf{F}_\mathbf{n}(\mathbf{U})]\,, \qquad \mathbf{F}_\mathbf{n} = \mathbf{F}\cdot\mathbf{n}, \qquad [q] = q^+ - q^-, \tag{1}$$

and, for a genuinely nonlinear family $k$, the Lax admissibility condition $\lambda_k(\mathbf{U}^-) > \sigma > \lambda_k(\mathbf{U}^+)$ (Lax, 1957). A contact is linearly degenerate: pressure and normal velocity are continuous across it and the density is free to jump.

**Algorithm 1** Wave structure from Riemann data

**Require:** states $\mathbf{U}_L, \mathbf{U}_R$; contact tolerance $\tau$
1: $(p_*, u_*) \leftarrow$ exact star state; $\mathcal{F} \leftarrow [\,]$
2: **if** $p_* > p_L$ **then** append SLOWSHOCK
3: **end if**
4: $\rho_*^L, \rho_*^R \leftarrow$ star densities from the shock or isentropic relation on each side
5: **if** $|\rho_*^L - \rho_*^R| > \tau \max(\rho_*^L, \rho_*^R)$ **then** append CONTACT
6: **end if**
7: **if** $p_* > p_R$ **then** append FASTSHOCK
8: **end if**
9: **return** $\mathcal{F}$ ▷ $|\mathcal{F}|$ fronts, $|\mathcal{F}| + 1$ branches

### 3.1 ANSATZ IN ONE DIMENSION

With $n$ fronts at $s_1(t) < \cdots < s_n(t)$ the solution is

$$\mathbf{U}(x,t) = \textstyle\sum_{k=0}^{n} \chi_k(x,t)\, \mathbf{U}_k(x,t;\theta_k), \qquad \chi_k = \mathbf{1}[s_k(t) < x < s_{k+1}(t)]\,, \tag{2}$$

where each branch $\mathbf{U}_k$ is a multilayer perceptron with softplus outputs for density and pressure. The indicator carries no gradient; fronts enter the loss through their jump conditions. Fronts are written as $s_k(t) = x_0 + t\, v_k(t)$ with $v_1 = w_1(t)$ and $v_k = v_{k-1} + \mathrm{softplus}(w_k(t))$, each $w_k$ a small network, which starts every front at $x_0$ and prevents crossing by construction, and which therefore excludes wave interaction. The front speed $\dot{s}_k$ is obtained exactly by automatic differentiation.

### 3.2 DERIVING THE WAVE STRUCTURE FROM THE INITIAL DATA

For Riemann data the exact star state $(p_*, u_*)$ follows from a Newton iteration on the pressure function (Toro, 2009). A nonlinear wave is a shock, and carries a front, exactly when $p_*$ exceeds the initial pressure on its side; rarefactions are continuous and carry none. The contact carries a front only if the star densities $\rho_*^L, \rho_*^R$ on its two sides differ by more than a relative tolerance $\tau = 10^{-3}$ (Algorithm 1). The star state decides only which fronts exist and their families; its values never enter the loss (Appendix G). For Riemann data it nonetheless determines the exact solution, so the construction is not a general solver (Section 6). The contact test is not a formality: for symmetric two-rarefaction data the contact has zero strength, yet a contact front placed there is unopposed by any loss term, since contact conditions leave density free, and every seed placed a spurious density jump there (Appendix B). Each element replaced a hand-made choice that had failed measurably: without the Lax term one seed in three gave an admissible shock, and a single front with its family hard-coded for Sod placed the shock of a mirrored Sod problem $96\times$ worse than on Sod, against $2.6\times$ with the family derived.

### 3.3 OBJECTIVE

The physics term combines the residual on collocation points outside a band of half-width $0.02$ around every front, the RH defect of Equation 1 at sampled front points, and an admissibility term. For a shock of family $k$ the Lax condition becomes a one-sided penalty with margin $\epsilon = 0.05$,

$$\mathcal{A}_k = \big[\epsilon - \big(\lambda_k(\mathbf{U}^-) - \dot{s}_k\big)\big]_+ + \big[\epsilon - \big(\dot{s}_k - \lambda_k(\mathbf{U}^+)\big)\big]_+\,, \tag{3}$$

which enters the loss as the mean of $\mathcal{A}_k^2$ over sampled front times and is inactive once the condition holds with margin; for a contact we penalise jumps in pressure and velocity and the mismatch between $\dot{s}_k$ and the fluid velocity. Initial and boundary terms complete the loss. The physics, initial and boundary groups are each normalised by their magnitude at initialisation. Normalising the five terms individually instead raises the weight of the residual, the term that rewards smearing, by a factor of $1700$ to $5500$ on Sod and degrades the median Sod error elevenfold (Appendix C).

### 3.4 TWO SPACE DIMENSIONS

In two dimensions front $k$ is the zero set of a level set $\varphi_k(x,y,t)$, giving

$$\mathbf{n} = \frac{\nabla\varphi_k}{|\nabla\varphi_k|}, \qquad \sigma = -\frac{\partial_t\varphi_k}{|\nabla\varphi_k|}. \tag{4}$$

Every condition above carries over with $\mathbf{F}$ replaced by $\mathbf{F_n}$ and velocity by its normal component, while tangential velocity is continuous across a shock. A randomly initialised level set has an arbitrary zero set, so we anchor it,

$$\varphi_k(x, y, t) = (\mathbf{n}_0 \cdot \mathbf{x} - d_0) - t\,\psi_k(x, y, t; \theta), \tag{5}$$

which makes the initial front exact while leaving the normal free to rotate and the front free to curve for $t > 0$; a curved initial front uses its signed distance in place of the first term. Unlike a graph over a fixed axis (Lorin & Novruzi, 2026), the zero set may in principle close or fold. The jump conditions must be imposed on $\{\varphi_k = 0\}$, which cannot be sampled directly, so we carry sampled points onto it with a unit-normal variant, $\mathbf{x} \leftarrow \mathbf{x} - \varphi_k\,\mathbf{n}$, of the generalised Newton iteration (Ben-Israel, 1966) used by Atzmon et al. (2019), without gradient tracking (Algorithm 2, Appendix D). Before any training we verified that the planar RH defect vanishes to $10^{-5}$ for a known shock embedded at five angles and three tangential velocities.

### 3.5 Selecting a run without ground truth

In one dimension the entropy solution of a Riemann problem is unique (Chen et al., 2002), so seeds approximate one function and their spread is a property of the optimiser. Following Mishra & Molinaro (2023), whose estimates bound the error by the training loss under a stability assumption, a lower physics defect should then indicate a better approximation. The final training loss cannot be compared across seeds, however, because each run calibrates its own term scales. We therefore reload each trained network and evaluate an uncalibrated score, averaged over 24 fixed batches shared by every run,

$$S = \mathcal{L}_{\text{res}} + \mathcal{L}_{\text{RH}} + \mathcal{L}_{\text{Lax}} + 10\,\mathcal{L}_{\text{init}} + \mathcal{L}_{\text{bdry}}, \tag{6}$$

whose weights are the training loss's group weights, fixed before any score was computed, and select the run with the lowest $S$. No reference solution enters. Uniqueness is necessary but not sufficient, since ranking imperfect runs also needs a stability estimate we do not prove, and in several dimensions admissible solutions of Euler Riemann problems need not be unique (Klingenberg et al., 2020), so the argument is one-dimensional.

## 4 Experimental protocol

Every neural configuration is run with three seeds, apart from a single-run survey shown for coverage (Appendix I), and we report the median, the best, and the spread (max/min); on the colliding problem the first two seeds differ by $1.2\times$ and the third widens the spread to $5.5\times$. With three seeds per arm no two-sided rank test can reach $p < 0.1$, so we call a comparison resolved only when every seed of one arm beats every seed of the other. Ground truth is the exact Riemann solution on $200\,001$ points, apart from the earlier Sod baselines of Appendices H and I, evaluated on 2001, where a front displacement of $10^{-4}$ is lost in quadrature noise. The two-dimensional network $L^1$ error is a Monte Carlo estimate over $4 \times 10^7$ samples, because grid quadrature of a displaced oblique front is biased. A shock counts as resolved only if its jump lies between $0.25$ and 2 times the exact jump; checked afterwards, none of the nine runs of the construction on the Riemann problems is degenerate, with every predicted density range at least $0.52$ of the reference's. Branches are $\tanh$ perceptrons with $128$ units, five hidden layers in one dimension and four in two; fronts have two hidden layers, of $64$ units in one dimension and $128$ for the level set. Training uses Adam at $10^{-3}$ for $15\,000$ iterations unless stated (Appendix A). The classical reference is a second-order MUSCL finite-volume scheme with HLLC flux, minmod limiting on primitive variables and Heun time stepping at CFL $0.4$, in vectorised NumPy. Networks train on one NVIDIA GB10 GPU; the classical solver runs on its CPU. Every table is generated by script from stored measurements, which accompany the code.

## 5 Results

### 5.1 Accuracy tracks the discontinuous fraction of the solution

Table 1 reports the construction on three Riemann problems spanning the possible structures (Sod, 1978), together with three problems we built to contain exactly one wave each: a single shock whose

Table 1: Density $L^1$ error at the final time, three seeds each. The derived structure is produced by Algorithm 1, not supplied. Spread is max/min over seeds; seconds are per seed.

| Problem | fronts derived | median $L^1(\rho)$ | best | spread | s |
|---|---|---|---|---|---|
| Single shock | 1 shock | $1.58 \times 10^{-4}$ | $2.80 \times 10^{-5}$ | 75.5× | 469 |
| Single contact | 1 contact | $1.35 \times 10^{-3}$ | $3.59 \times 10^{-4}$ | 16.0× | 481 |
| Single rarefaction | none (fan) | $1.94 \times 10^{-2}$ | $1.92 \times 10^{-2}$ | 4.0× | 225 |
| Colliding | shock, contact, shock | $4.92 \times 10^{-4}$ | $1.09 \times 10^{-4}$ | 5.5× | 760 |
| Sod | contact, shock | $5.87 \times 10^{-3}$ | $4.06 \times 10^{-3}$ | 2.1× | 624 |
| Two rarefaction | none | $1.36 \times 10^{-1}$ | $1.05 \times 10^{-1}$ | 2.7× | 292 |

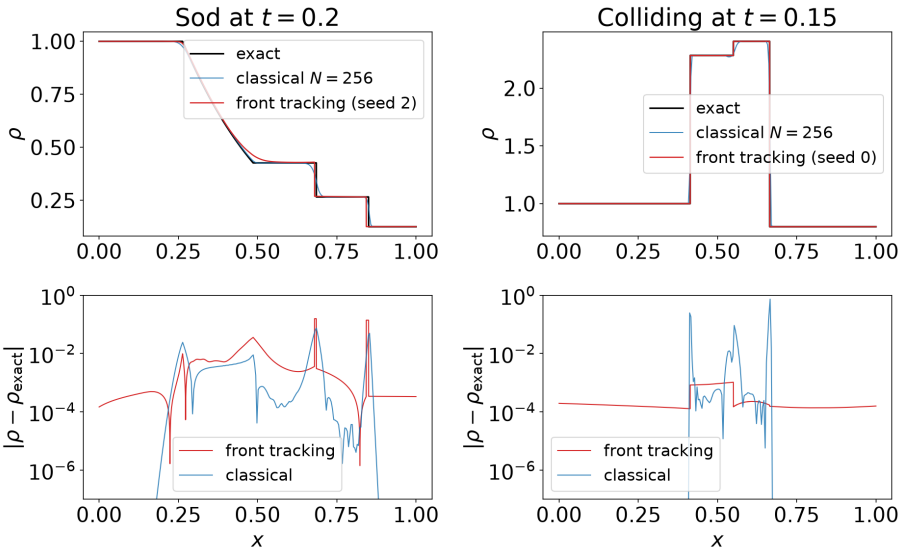



Figure 2: Density at the final time for Sod (left) and the colliding problem (right), median seed, against the exact solution and a classical solver at $N = 256$; lower panels show the pointwise error. Classical error is spread over several cells at each discontinuity. Front-tracking error at a front is confined to the strip between the learned and exact positions, where it equals the jump: visible on Sod, where this seed's fronts are displaced by 5.8 and $7.5 \times 10^{-3}$, and narrower than the plotted sampling on the colliding problem.

two states satisfy the RH conditions, a single contact, and a single left-moving rarefaction for which the derived structure is empty. Figure 2 shows the Riemann solutions.

**Accuracy tracks how much of the solution is discontinuous, not how many waves it has.** A lone shock reaches $1.58 \times 10^{-4}$, a lone contact $1.35 \times 10^{-3}$ and a lone rarefaction $1.94 \times 10^{-2}$, which is worse than the whole Sod problem and 122× worse than the shock. The ordering matches the Riemann problems: the colliding problem is all discontinuities, Sod is two discontinuities and a fan, and the two-rarefaction problem has no front at all, and their errors rise in that order. For a rarefaction the derived structure is empty and the method is an ordinary PINN; a contact fixes its speed but leaves the density jump unconstrained. Against strong-form and relaxation networks (Zhou & Ma, 2024) trained for the same 15 000 iterations (Appendix I), the construction is 8 to 14× more accurate on Sod and 386 to 879× on the colliding problem, and every seed beats every baseline seed on both. On the two-rarefaction problem it is the plain strong form, and a width-matched strong form and a relaxation network are 1.8× more accurate. Because each region has its own branch, parameters grow with the number of fronts, from 67k to 280k, so part of the ordering may be capacity; a structurally wrong ansatz that keeps a zero-strength contact on the two-rarefaction problem is also 6.4× more accurate than the correct empty one. We keep the gate, since an ansatz that invents a discontinuity in a smooth solution is defective whatever its error norm.

### 5.2 WHERE A RESIDUAL OBJECTIVE IS BLIND

**A constant state lies in the null space of the residual operator**: $\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = 0$ holds identically for any constant, so the residual has no opinion about which constant a plateau settles on, and the star states are pinned only by jump conditions at sampled front points. Where the fronts are placed accurately, much of the error goes there. On the colliding problem every front lies within $1.5 \times 10^{-4}$ of its exact position, and the constant states carry 39 to 58 percent of the error (27 to 41 for the star states alone), although the exact solution is constant there. On Sod the farthest front of each seed lies $7 \times 10^{-3}$ to $1.8 \times 10^{-2}$ from its exact position; counting the strip between learned and exact front as front error, the fronts carry 34 to 52 percent, the fan 22 to 36 and the star states 17 to 27 (Appendix E).

**Anchoring to known data can help; anchoring to a learned branch costs reproducibility.** The obvious repair is to construct the plateau beside a shock from the jump conditions instead of learning it. The jump condition on that front is then satisfied identically for every speed, so it carries no gradient and no longer determines the front speed. Across the three variants anchored to a learned branch the median moves between $1.43\times$ better and $5.25\times$ worse, but every one widens the colliding seed spread from $5.5\times$ to between $13\times$ and $122\times$. When the anchor is data instead, namely the far-field states, which are known exactly for Riemann data, anchoring far field and plateau together places all three colliding seeds below all three unanchored ones ($6.09\times$ at the median) and tightens the spread; on Sod the seeds interleave and the $1.24\times$ median difference is not resolved. Anchoring the far field alone is not benign: every Sod seed is worse than every unanchored one ($2.2\times$ at the median), and the colliding spread widens to $23\times$ (Appendix F).

### 5.3 RUN SELECTION

Seed spreads range from $1.4\times$ to $122\times$ across variants, the largest driven by single failed seeds. Scoring every saved network with $S$ (Equation 6), **the lowest-score run is the most accurate of three in 11 of 13 distinct problem and variant settings**, where a random pick succeeds with probability one in three; the settings share two problems and are not independent. Selection improves on the median seed by $1.45\times$ at the median and by up to $15.6\times$ (Appendix G). It selects a seed within a fixed variant; which variant is best was determined with the reference, and pooling all 18 distinct runs of a problem under $S$ instead returns the best run on Sod and, on the colliding problem, a run $1.73\times$ worse than the best, at $15\times$ the training time of the reference-chosen variant; pooling also compares unlike scores, since anchoring the far field makes the initial and boundary terms of $S$ vanish identically on the colliding problem. The two failures bound the claim. With the plateau anchored on the front, on Sod the jump term is flattened on two of three seeds and the initial-condition term then excludes the best one (on the colliding problem the same variant succeeds). With anchored far field and plateau on the colliding problem, the scores of the selected and best seeds differ by 12 percent, and selection returns $8.07 \times 10^{-5}$ where the best seed reached $3.97 \times 10^{-5}$. Causal time weighting (Wang et al., 2024) reduces the single-shock spread from $75.5\times$ to between 1.4 and $2.3\times$, but no causally weighted run matches the best uniform seed (Appendix H).

### 5.4 TWO SPACE DIMENSIONS

A planar shock at an angle is an exact solution of the two-dimensional equations. We embed Sod's right-moving shock at $0^\circ$, $30^\circ$ and $45^\circ$ to the mesh axes and give the network the initial front, the wave family and exact Dirichlet data on all four edges. With both states fixed, the jump condition determines the normal speed, so this test checks the level set, the projection and the optimiser rather than the prediction of an unknown flow. The classical solver at $N = 800^2$ receives the same boundary data (Table 2; Figure 3 in Appendix D).

**On a planar front the construction transfers; on a curved one it fails.** Median front errors of 0.76 to $2.5 \times 10^{-4}$ and median $L^1$ errors of 1.3 to $2.0 \times 10^{-4}$ are comparable to the one-dimensional single shock, and the $45^\circ$ median $L^1$ is $1.5\times$ the $0^\circ$ one, within the seed spread. The classical $L^1$ error is 1.0 to $1.8\times$ the median seed's when its piecewise-constant field is integrated as ours is, but 0.87 to $1.5\times$ when each cell value is compared with the exact cell average, so at $45^\circ$ the classical solver is then the more accurate; at $30^\circ$ and $45^\circ$ the worst seed's error is about three times the classical one, and in two dimensions we have no justified way to exclude it. Whether the construction places

Table 2: Oblique shock at $t = 0.15$ on the unit square. Front-tracking values are medians over three seeds, with the front error the median distance of the projected zero set from the exact plane. The classical front error is the median over interior grid rows of the interpolated density-midpoint crossing, and its $L^1$ integrates the piecewise-constant field against the exact solution. Seconds are per seed (GPU) and per solve (CPU). Appendix D compares estimators.

| | Front tracking (3 seeds, median) | | | Classical, $N = 800^2$ | | |
|---|---|---|---|---|---|---|
| Angle | $L^1(\rho)$ | front err. | s | $L^1(\rho)$ | front err. | s |
| $0°$ | $1.32 \times 10^{-4}$ | $1.35 \times 10^{-4}$ | 782 | $2.35 \times 10^{-4}$ | $5.26 \times 10^{-4}$ | 1038 |
| $30°$ | $1.43 \times 10^{-4}$ | $7.56 \times 10^{-5}$ | 783 | $2.33 \times 10^{-4}$ | $4.30 \times 10^{-4}$ | 1119 |
| $45°$ | $2.04 \times 10^{-4}$ | $2.54 \times 10^{-4}$ | 801 | $2.12 \times 10^{-4}$ | $4.90 \times 10^{-4}$ | 1128 |

Table 3: Serial wall-clock seconds at matched density $L^1$. *The variant whose selected run is most accurate, which is known only from the reference; three seeds. *Pooled*: all distinct runs of all variants, selected by $S$ with no reference. †Extrapolated $1.16\times$ beyond the finest measured resolution. Classical timings are upper bounds (text).

| Problem | selection | runs | our $L^1$ | serial s | classical, equal $L^1$ | verdict |
|---|---|---|---|---|---|---|
| Sod | best variant* | 3 | $2.87 \times 10^{-3}$ | 3546 | $N$=512, 1.0 s | classical 3543× |
| | pooled, by $S$ | 18 | $2.87 \times 10^{-3}$ | 17118 | $N$=512, 1.0 s | classical 17103× |
| Colliding | best variant* | 3 | $6.85 \times 10^{-5}$ | 1614 | $N$=38071, 11810 s† | ours 7.3× |
| | pooled, by $S$ | 18 | $6.85 \times 10^{-5}$ | 24748 | $N$=38071, 11810 s† | classical 2.1× |
| Two rarefaction | best variant* | 3 | $1.05 \times 10^{-1}$ | 835 | $N$=64, 0.04 s | classical 23650× |

the front better also depends on the estimator: the median seed is 1.9 to $5.7\times$ closer than the classical median midpoint crossing, but a per-row mass balance places the classical front $1.1 \times 10^{-5}$ from the exact plane at $0°$, closer than any seed, so we claim no advantage in front placement (Appendix D). A cylindrical blast, with Sod states inside a circle of radius 0.2 and one circular level-set front for the outgoing shock, is the one test in which the front is curved and its normal varies along it; it is not a rotated one-dimensional problem. Against a classical reference at $N = 800^2$, whose threshold sensitivity we declared as a floor of $1.5 \times 10^{-3}$ before running, every seed places the shock 2.4 to $3.3 \times 10^{-2}$ short of the reference radius of 0.446, 5 to 7 percent and 16 to 22 times the floor (Appendix D). The contact and the inward fan are left to the smooth branch, a handicap we also declared beforehand.

### 5.5 Cost at matched accuracy

Table 3 compares the cost of selected runs with that of the classical solver refined to the same accuracy. The classical solver is a capturing scheme; for Riemann data, classical front tracking and the exact solver that Algorithm 1 already calls would be exact at negligible cost, so the comparison measures how efficiently the ansatz represents a known solution, not a use case.

**With a rarefaction present, the classical solver wins by three to four orders of magnitude.** On Sod it reaches our accuracy in about one second, against an hour of training for three seeds. On the two-rarefaction problem a 64-cell grid is more accurate than any seed. Only on the colliding problem, whose every wave is a discontinuity, does a crossover occur. There, without extrapolation, the finest classical run we executed, $N = 32\,768$ cells, reaches $7.76 \times 10^{-5}$ in 8143 s, timed once, while the selected run reaches $6.85 \times 10^{-5}$ with three seeds costing 1614 s serially, a $5.0\times$ advantage. This favours the network in three ways. The variant was chosen with the reference; selecting among all variants without it returns the same run at $3\times$ the cost of that classical run. Our one-dimensional classical runs use the two-dimensional code on an $N \times 4$ strip, measured to be 1.7 to $2.6\times$ slower than a single line of cells at an error within 0.21 percent. And a compiled solver would be faster than vectorised NumPy on a CPU. **The colliding advantage would not survive an optimised classical baseline, and we do not claim it would.** The network is also evaluated at a single training budget (Appendix J).

## 6 DISCUSSION

**Where it is limited.** Where the fronts are accurate, about half of the remaining error sits in the constant states, which the interior residual cannot see and only sampled jump conditions constrain; on Sod, front displacement and the fan carry most of it. Rarefactions receive no help, and reproducibility is poor without selection. The construction needs the exact star state, which for Riemann data already determines the solution: it uses only which fronts exist and their families, but a method that needs Riemann data to set up its ansatz is not a general solver. The one-dimensional fronts cannot interact, the fixed Lax margin $\epsilon$ would bias the weak shocks that form from smooth data, and the residual exclusion band has a fixed width, which on Sod covers the whole right star region until $t \approx 0.05$. Shock formation, which a graph front already handles in two dimensions (Lorin & Novruzi, 2026), is not addressed here; wave interaction and topological change are where a level set should be more natural than a list of fronts or a graph, and the cylindrical result shows that even a single curved front is not yet handled.

**Where the jump conditions are the object of interest.** With nonconservative products the jump conditions depend on a path in state space (Dal Maso et al., 1995), and a scheme consistent with the prescribed path reaches the right limit only under convergence of graphs, which "does hold for the Glimm and front tracking schemes, but usually fails for finite difference schemes" (Castro et al., 2008; Hou & LeFloch, 1994; Abgrall & Karni, 2010). In-cell discontinuous-reconstruction schemes capture isolated shocks at the prescribed path in one dimension (Chalons, 2020; Pimentel-García et al., 2022; 2024), and we do not compare against them; the neural construction is one more member of the class that imposes the jump condition directly. Appendix K measures the slope of computed against exact shock speed across a family of paths. The tracked front is given the path, so a slope near one is expected by construction; the optimiser realises it with slope $0.998$ to $1.004$, and a standard path-conservative scheme of Lax-Friedrichs type has slope $-0.26$ at $6400$ cells, the wrong sign and stronger than at $400$ ($-0.12$). The appendix also records a two-dimensional orientation effect we initially measured and then traced to our own non-rotationally-invariant extension of the system.

**Relation to relaxation.** Relaxation networks (Zhou & Ma, 2024) attack the same defect through the loss and need no wave structure; at our training budget they are more accurate than the construction only where the construction has no front. The two are composable, and we have not tried it.

## 7 CONCLUSION

Changing the ansatz so that the residual never sees a discontinuity removes the defect that makes PINNs smear shocks: on Riemann problems with discontinuities it beats strong-form and relaxation networks by $8$ to $879$ times at the same budget, and it carries over to a planar front in two dimensions. Several limitations remain. Once the fronts are accurate, error shifts to the constant states and fans. Seed spread requires re-scored selection to manage without ground truth, curved fronts remain unsolved, and the computational cost approaches a capturing scheme only where the solution is built entirely from discontinuities and the answer is already known. Ultimately, we expect explicit tracking to be most useful where the jump conditions are themselves the object of interest. For nonconservative systems, we leave the measurement of its cost and comparisons with in-cell reconstruction schemes to future work.

## ETHICS STATEMENT

This work develops numerical methods for fluid dynamics and involves no human subjects, personal data or foreseeable dual-use concerns beyond those of computational fluid dynamics generally.

## REPRODUCIBILITY STATEMENT

Section 4 and Appendix A define the problems, architectures, sampling, loss weights, training budgets and evaluation metrics. And each of the other appendices give every table's source. The supplementary material contains the code that produces every measurement and every derived quantity

in the paper, the measurements themselves as CSV files, and the trained networks for reference. The material's README outlines the commands to reproduce the results from the trained networks or from scratch. Runs that failed a gate are recorded as failures rather than omitted, for completeness.

## AI USE STATEMENT

We declare that search based generative AI tools were used in this work to extensively and exhaustively discover related works. We also declare that claims generated by search integrated AI or AI generated search summaries / text is not made use of in this work. To guard against potential errors such tools make, every returned search result / reference was explicitly reviewed and added only after the confirmation of relevance by authors. The authors declare that generative AI tools are only utilized as aid in exhaustive discovery, and take full responsibility for the content written on this paper.

## REFERENCES

J. Abbasi, A. D. Jagtap, B. Moseley, A. Hiorth, and P. O. Andersen. Challenges and advancements in modeling shock fronts with physics-informed neural networks: a review and benchmarking study. *Neurocomputing*, 657:131440, 2025. doi: 10.1016/j.neucom.2025.131440.

R. Abgrall and S. Karni. A comment on the computation of non-conservative products. *Journal of Computational Physics*, 229(8):2759–2763, 2010. doi: 10.1016/j.jcp.2009.12.015.

T. D. Aslam. A level-set algorithm for tracking discontinuities in hyperbolic conservation laws: I. Scalar equations. *Journal of Computational Physics*, 167(2):413–438, 2001.

M. Atzmon, N. Haim, L. Yariv, O. Israelov, H. Maron, and Y. Lipman. Controlling neural level sets. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Adi Ben-Israel. A Newton-Raphson method for the solution of systems of equations. *Journal of Mathematical Analysis and Applications*, 15(2):243–252, 1966. doi: 10.1016/0022-247X(66)90115-6.

R. G. Cassia and R. R. Kerswell. Godunov loss functions for modelling of hyperbolic conservation laws. *Computer Methods in Applied Mechanics and Engineering*, 2025. doi: 10.1016/j.cma.2025.117782.

Manuel J. Castro, Philippe G. LeFloch, María Luz Muñoz-Ruiz, and Carlos Parés. Why many theories of shock waves are necessary: Convergence error in formally path-consistent schemes. *Journal of Computational Physics*, 227(17):8107–8129, 2008. doi: 10.1016/j.jcp.2008.05.012.

C. Chalons. Path-conservative in-cell discontinuous reconstruction schemes for non conservative hyperbolic systems. *Communications in Mathematical Sciences*, 18(1):1–30, 2020.

Aidan Chaumet and Jan Giesselmann. Improving weak PINNs for hyperbolic conservation laws: Dual norm computation, boundary conditions and systems. *The SMAI Journal of Computational Mathematics*, 10:373–401, 2024. doi: 10.5802/smai-jcm.116.

G.-Q. Chen, H. Frid, and Y. Li. Uniqueness and stability of Riemann solutions with large oscillation in gas dynamics. *Communications in Mathematical Physics*, 228:201–217, 2002.

I.-L. Chern and P. Colella. A conservative front tracking method for hyperbolic conservation laws. LLNL Report UCRL-97200, Lawrence Livermore National Laboratory, July 1987.

Mirco Ciallella, Mario Ricchiuto, Renato Paciorri, and Aldo Bonfiglioli. Extrapolated shock tracking: Bridging shock-fitting and embedded boundary methods. *Journal of Computational Physics*, 412:109440, 2020. doi: 10.1016/j.jcp.2020.109440.

E. J. R. Coutinho, M. Dall'Aqua, L. McClenny, M. Zhong, U. Braga-Neto, and E. Gildin. Physics-informed neural networks with adaptive localized artificial viscosity. *Journal of Computational Physics*, 489:112265, 2023. doi: 10.1016/j.jcp.2023.112265.

Gianni Dal Maso, Philippe G. LeFloch, and François Murat. Definition and weak stability of nonconservative products. *Journal de Mathématiques Pures et Appliquées*, 74(6):483–548, 1995.

A. Daw, J. Bu, S. Wang, P. Perdikaris, and A. Karpatne. Mitigating propagation failures in physics-informed neural networks using retain-resample-release (R3) sampling. In *International Conference on Machine Learning*, 2023.

T. De Ryck and S. Mishra. Numerical analysis of physics-informed neural networks and related models in physics-informed machine learning. *Acta Numerica*, 33:633–713, 2024. doi: 10.1017/S0962492923000089.

Tim De Ryck, Siddhartha Mishra, and Roberto Molinaro. wPINNs: Weak physics informed neural networks for approximating entropy solutions of hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 62(2):811–841, 2024. doi: 10.1137/22M1522504.

J. Glimm, E. Isaacson, D. Marchesin, and O. McBryan. Front tracking for hyperbolic systems. *Advances in Applied Mathematics*, 2(1):91–119, 1981. doi: 10.1016/0196-8858(81)90040-3.

J. Han, J. Chen, F. Hu, and L. Mei. A study on shock capturing in the Burgers' equation based on RH-piecewise PINNs. *Computers & Mathematics with Applications*, 199:309–324, 2025.

Thomas Y. Hou and Philippe G. LeFloch. Why nonconservative schemes converge to wrong solutions: error analysis. *Mathematics of Computation*, 62(206):497–530, 1994. doi: 10.1090/S0025-5718-1994-1201068-0.

Ameya D. Jagtap, Ehsan Kharazmi, and George Em Karniadakis. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 365:113028, 2020. doi: 10.1016/j.cma.2020.113028.

C. Klingenberg, O. Kreml, V. Mácha, and S. Markfelder. Shocks make the Riemann problem for the full Euler system in multiple space dimensions ill-posed. *Nonlinearity*, 33(12):6517–6540, 2020. doi: 10.1088/1361-6544/aba3b2.

Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 26548–26560, 2021.

S. Lanthaler, R. Molinaro, P. Hadorn, and S. Mishra. Nonlinear reconstruction for operator learning of PDEs with discontinuities. In *International Conference on Learning Representations*, 2023.

Peter Lax and Burton Wendroff. Systems of conservation laws. *Communications on Pure and Applied Mathematics*, 13(2):217–237, 1960. doi: 10.1002/cpa.3160130205.

Peter D. Lax. Hyperbolic systems of conservation laws II. *Communications on Pure and Applied Mathematics*, 10(4):537–566, 1957. doi: 10.1002/cpa.3160100406.

P. G. LeFloch and S. Xiang. Physics-informed token transformer methodology for nonlinear balance laws. I. Schwarzschild-Burgers fluid flows. arXiv preprint arXiv:2607.23143, 2026.

R. Leiteritz and D. Pflüger. How to avoid trivial solutions in physics-informed neural networks. arXiv preprint arXiv:2112.05620, 2021.

L. Liu, S. Liu, H. Xie, F. Xiong, T. Yu, M. Xiao, L. Liu, and H. Yong. Discontinuity computing using physics-informed neural networks. *Journal of Scientific Computing*, 98:22, 2024. doi: 10.1007/s10915-023-02412-1.

Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y. Hou, and Max Tegmark. KAN: Kolmogorov–Arnold networks. In *International Conference on Learning Representations*, 2025.

Emmanuel Lorin and Arian Novruzi. Non-diffusive neural network method for hyperbolic conservation laws. *Journal of Computational Physics*, 513:113161, 2024. doi: 10.1016/j.jcp.2024.113161.

Emmanuel Lorin and Arian Novruzi. A diffusion-free multi-layer neural-network method for multidimensional nonlinear hyperbolic equations. arXiv preprint arXiv:2609.28971, 2026.

Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. DeepXDE: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021. doi: 10.1137/19M1274067.

K. O. Lye, S. Mishra, and D. Ray. Deep learning observables in computational fluid dynamics. *Journal of Computational Physics*, 410:109339, 2020. doi: 10.1016/j.jcp.2020.109339.

Zhiping Mao, Ameya D. Jagtap, and George Em Karniadakis. Physics-informed neural networks for high-speed flows. *Computer Methods in Applied Mechanics and Engineering*, 360:112789, 2020. doi: 10.1016/j.cma.2019.112789.

Levi D. McClenny and Ulisses M. Braga-Neto. Self-adaptive physics-informed neural networks. *Journal of Computational Physics*, 474:111722, 2023. doi: 10.1016/j.jcp.2022.111722.

S. Mishra and R. Molinaro. Estimates on the generalization error of physics-informed neural networks for approximating PDEs. *IMA Journal of Numerical Analysis*, 43(1):1–43, 2023. doi: 10.1093/imanum/drab093.

Gino Moretti. Thirty-six years of shock fitting. *Computers & Fluids*, 31(4–7):719–723, 2002. doi: 10.1016/S0045-7930(01)00072-X.

Stanley Osher and James A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988. doi: 10.1016/0021-9991(88)90002-2.

M. Pascal, R. Duvigneau, and L. Monasse. Simulating shock solutions for hyperbolic conservation laws with physics-informed neural networks. Research Report RR-9605, Inria, 2025. hal-05405130.

R. G. Patel, I. Manickam, N. A. Trask, M. A. Wood, M. Lee, I. Tomas, and E. C. Cyr. Thermodynamically consistent physics-informed neural networks for hyperbolic systems. *Journal of Computational Physics*, 449:110754, 2022. doi: 10.1016/j.jcp.2021.110754.

Ernesto Pimentel-García, Manuel J. Castro, Christophe Chalons, Tomás Morales de Luna, and Carlos Parés. In-cell discontinuous reconstruction path-conservative methods for non conservative hyperbolic systems: second-order extension. *Journal of Computational Physics*, 459:111152, 2022. doi: 10.1016/j.jcp.2022.111152.

Ernesto Pimentel-García, Manuel J. Castro, Christophe Chalons, and Carlos Parés. High-order in-cell discontinuous reconstruction path-conservative methods for nonconservative hyperbolic systems: DR.MOOD method. *Numerical Methods for Partial Differential Equations*, 40(6):e23133, 2024. doi: 10.1002/num.23133.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi: 10.1016/j.jcp.2018.10.045.

S. U. Rehman and W. Yousuf. Extended physics informed neural network for hyperbolic two-phase flow in porous media. arXiv preprint arXiv:2511.13734, 2025.

Baifen Ren and Carlos Parés. High-order WENO finite-difference methods for hyperbolic nonconservative systems of partial differential equations. *Journal of Computational Physics*, 535:114047, 2025. doi: 10.1016/j.jcp.2025.114047.

S. Rhebergen, O. Bokhove, and J. J. W. van der Vegt. Discontinuous Galerkin finite element methods for hyperbolic nonconservative partial differential equations. *Journal of Computational Physics*, 227(3):1887–1922, 2008. doi: 10.1016/j.jcp.2007.10.007.

Gary A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27(1):1–31, 1978. doi: 10.1016/0021-9991(78)90023-2.

Q. Sun, Z. Liu, L. Ju, and X. Xu. Lift-and-embed learning method for solving scalar hyperbolic equations with discontinuous solutions. *Journal of Scientific Computing*, 107(3):87, 2026. doi: 10.1007/s10915-026-03268-x.

Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems*, volume 33, pp. 7537–7547, 2020.

Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer, Berlin, Heidelberg, 3rd edition, 2009. doi: 10.1007/b79761.

Sifan Wang, Shyam Sankaran, and Paris Perdikaris. Respecting causality for training physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 421: 116813, 2024. doi: 10.1016/j.cma.2024.116813.

Nan Zhou and Zheng Ma. Capturing shock waves by relaxation neural networks. arXiv preprint arXiv:2404.01163, 2024.

## A Problems and training

Table 4: Problems. States are $(\rho, u, p)$ on $[0, 1]$ with the discontinuity at $x_0 = 0.5$; $T$ is the final time. Parameters count all branches and fronts of the construction.

| Problem | left $(\rho, u, p)$ | right $(\rho, u, p)$ | fronts derived | $T$ | parameters |
|---|---|---|---|---|---|
| Colliding | (1, 1.5, 1) | (0.8, −1.2, 0.6) | shock, contact, shock | 0.15 | 280,335 |
| Sod | (1, 0, 1) | (0.125, 0, 0.1) | contact, shock | 0.2 | 209,163 |
| Two rarefaction | (1, −1, 1) | (1, 1, 1) | none | 0.15 | 66,819 |
| Single shock | (0.266, 0.927, 0.303) | (0.125, 0, 0.1) | 1 shock | 0.2 | 137,991 |
| Single contact | (1, 0.8, 1) | (0.4, 0.8, 1) | 1 contact | 0.3 | 137,991 |
| Single rarefaction | (1, 0, 1) | (0.52, 0.726, 0.4) | none (fan) | 0.25 | 66,819 |

**Training.** Each one-dimensional iteration samples 4000 collocation points, 256 front times, 512 initial and 256 boundary points; the two-dimensional study samples 4000, 512, 1024 and 512. Initial and boundary groups carry weights 10 and 1 relative to the physics group, and each group is normalised by its mean over eight batches at initialisation. Adam runs at $10^{-3}$ for 15 000 iterations unless stated, and the projection of Algorithm 2 takes eight steps. The two-dimensional problems use the unit square and final time 0.15: the oblique problem places Sod's post-shock and right states either side of a planar front along the line $\mathbf{n}_0 \cdot \mathbf{x} = 0.3$, and the cylindrical blast places Sod's left state inside a circle of radius 0.2 centred in the square and its right state outside.

## B From the special case to the general construction

The general construction of Section 3.2 was reached in steps, each forced by a measured failure. All numbers are Sod medians over three seeds unless stated. The single-front rows were evaluated on a 2001-point grid.

**One front, jump conditions only.** Imposing only the RH condition gave a median $L^1$ error of $7.08 \times 10^{-2}$ with a seed spread of 11.7×, and one seed in three produced an admissible shock. The failures had large jumps at misplaced fronts rather than collapsing to a trivial map: two seeds placed the front at $x = 0.61$ and 0.63, 0.24 and 0.22 from the exact shock, with a jump 3.7 and 3.8 times the exact one. The RH conditions alone admit entropy-violating solutions and the optimiser found them.

**One front, with the Lax penalty.** Adding the admissibility penalty moved the median error to $5.57 \times 10^{-3}$, with three of three seeds admissible and the median front distance falling to $4.5 \times 10^{-3}$. The weight matters: at ten times the weight used, the median degrades to $2.55 \times 10^{-2}$, because an over-weighted inequality distorts the solution it should only select (Table 13).

**One front, with the family derived.** The Lax condition needs the characteristic family of the front. With the family hard-coded for Sod, the construction placed the front with median error $4.51 \times 10^{-3}$ on Sod and $4.33 \times 10^{-1}$ on the mirrored problem, whose shock runs left. Deriving the family from the initial pressures gave $1.18 \times 10^{-2}$ on the mirrored problem and $2.1 \times 10^{-3}$ on the Lax shock tube, with every seed admissible. Letting the network choose the family gave medians of $1.73 \times 10^{-2}$ on Sod, where one seed of three chose the wrong family and placed the front $0.256$ from the shock, and $3.20 \times 10^{-2}$ on the mirrored problem.

**Many fronts.** A single front cannot represent two shocks, and the contact in Sod was smeared by whichever branch straddled it. Generalising to an ordered list of fronts required deriving the whole structure (Algorithm 1). With the contact suppressed, the general construction reproduces the single-front implementation bit for bit on all three seeds, from two independently written code paths.

**Contacts of zero strength.** The general form first placed a contact whenever the wave pattern had one. On symmetric two-rarefaction data, where the star density is $0.396$ on both sides of the contact and the exact jump is zero, this licensed a spurious density jump: with the gate removed, every seed places a jump of $0.019$ to $0.229$ across its contact front. Gating on the star-density difference removed it and made the derived structure empty, which is correct. As Section 5.1 reports, the gated construction is nonetheless $6.4\times$ less accurate.

## C Loss calibration

The loss has three groups, physics, initial condition and boundary condition, each normalised by its mean over eight batches at initialisation. Normalising each of the five constituent terms separately, which is the natural choice, amplifies whichever term is already best satisfied. On Sod at initialisation, over three seeds, the residual is $1.8$ to $6.0 \times 10^{-4}$ and the admissibility term $7.0 \times 10^{-2}$ to $6.5 \times 10^{-1}$, so per-term normalisation multiplies the weight of the residual by $1700$ to $5500$ and that of the admissibility term by $1.5$ to $14$. The residual is the term that rewards smearing, and the median Sod error degrades $11\times$, from $5.87 \times 10^{-3}$ to $6.65 \times 10^{-2}$, with every seed worse than every seed of the grouped normalisation. Terms measuring different quantities should be calibrated separately; terms measuring the same quantity at different natural scales must share a scale, because their relative magnitudes carry information that per-term normalisation removes.

## D Two-dimensional details

**Algorithm 2** Projection onto the front

**Require:** points $\mathbf{x}_i$, time $t$, level set $\varphi$, iteration count $m$

**for** $j = 1$ to $m$ **do**
  $\varphi_i, \mathbf{n}_i \leftarrow \varphi(\mathbf{x}_i, t),\ \nabla\varphi/|\nabla\varphi|$ ▷ without gradient tracking
  $\mathbf{x}_i \leftarrow \mathbf{x}_i - \varphi_i\, \mathbf{n}_i$
**end for**
**return** $\mathbf{x}_i$

Algorithm 2 differs from the generalised Newton step $\mathbf{x} \leftarrow \mathbf{x} - \varphi\,\nabla\varphi/|\nabla\varphi|^2$ used by Atzmon et al. (2019) in taking a unit normal; the two coincide when $|\nabla\varphi| = 1$, which the anchored form of Equation 5 satisfies at $t = 0$. Projected points behave as collocation points rather than as quantities carrying their own gradients. The two branches are combined by a hard switch on the sign of $\varphi$, so the level set is trained only through the jump and admissibility defects at the projected points. On a translating plane the projection reaches the zero set to $10^{-6}$ in eight iterations and stays finite on an untrained level set whose gradient is far from unit magnitude. Before training we verified that the planar RH defect vanishes to $10^{-5}$ for Sod's right-moving shock embedded at five angles and three tangential velocities, that the Lax penalty is inactive for the shock's own family and active for the wrong one, and that a uniform state has identically zero residual.

**Front estimators.** The network's front error is the median distance of projected zero-set points from the exact plane. A smeared classical front must first be located, and Table 5 gives three ways: the

single most downstream interpolated midpoint crossing over interior cells (the estimator of Table 7), the median over grid rows of each row's midpoint crossing, and the median over rows of a mass balance, which places each row's front where the exact solution would hold the same mass. The comparison reverses with the estimator: the median seed is closer than the classical median crossing at every angle, but the mass balance places the classical front closer than every seed at $0°$. The classical $L^1$ error depends on the score in the same way. Integrating the piecewise-constant field against the exact solution, as the network's Monte Carlo estimate does, changes it by at most five percent from cell-centre sampling; comparing each cell value with the exact cell average instead lowers it by 17 to 20 percent, because that score ignores the error of a constant cell value across the front.

Table 5: Front error of the oblique shock at $t = 0.15$ under different estimators, and the classical $L^1$ error under three scores.

| | front tracking, zero set | | classical front, $N = 800^2$ | | | classical $L^1(\rho)$, $N = 800^2$ | | |
|---|---|---|---|---|---|---|---|---|
| Angle | median seed | worst seed | extreme crossing | median crossing | median mass | cell centre | cell average | piecewise constant |
| $0°$ | $1.35 \times 10^{-4}$ | $1.79 \times 10^{-4}$ | $5.26 \times 10^{-4}$ | $5.26 \times 10^{-4}$ | $1.09 \times 10^{-5}$ | $2.41 \times 10^{-4}$ | $1.97 \times 10^{-4}$ | $2.35 \times 10^{-4}$ |
| $30°$ | $7.56 \times 10^{-5}$ | $4.39 \times 10^{-4}$ | $8.35 \times 10^{-4}$ | $4.30 \times 10^{-4}$ | $1.72 \times 10^{-4}$ | $2.28 \times 10^{-4}$ | $1.89 \times 10^{-4}$ | $2.33 \times 10^{-4}$ |
| $45°$ | $2.54 \times 10^{-4}$ | $4.05 \times 10^{-4}$ | $6.40 \times 10^{-4}$ | $4.90 \times 10^{-4}$ | $2.70 \times 10^{-4}$ | $2.23 \times 10^{-4}$ | $1.78 \times 10^{-4}$ | $2.12 \times 10^{-4}$ |

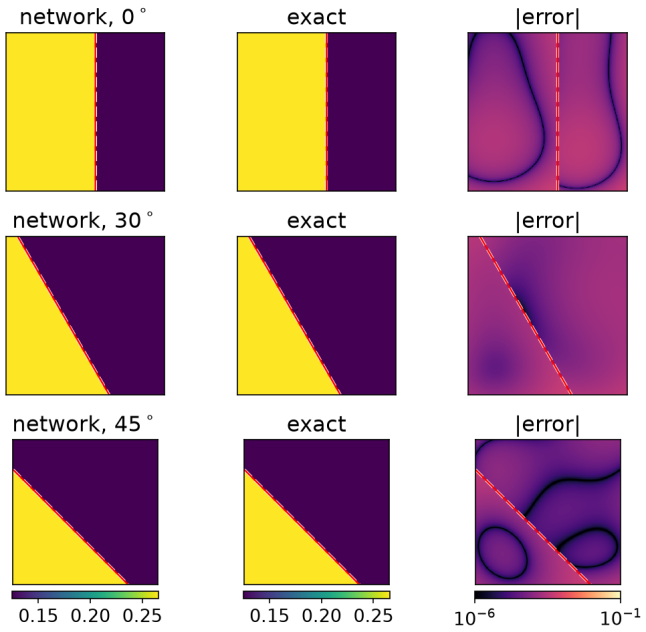



Figure 3: Oblique shock at $0°$, $30°$, $45°$ (rows), showing for each angle the seed with the lowest front error, chosen against the exact solution (medians are in Table 2): network density, exact density, and absolute error on a common logarithmic scale, with the learned zero set (red) over the exact front (white).

Figure 4 shows tangential invariance and front planarity for the seed with the lowest front error at each angle. Across all nine runs the front error ranges from $6.8 \times 10^{-5}$ to $4.4 \times 10^{-4}$, the $L^1$ error from $6.7 \times 10^{-5}$ to $7.1 \times 10^{-4}$, and the peak-to-peak deviation of the front from the exact plane from 1.2 to $4.2 \times 10^{-4}$; every run finished. The network is timed on the GPU and the classical solver on the CPU, and the classical cost grew with an exponent of 3.9 from $N = 200$ to 400 and 3.5 from 400 to 800, above the asymptotic $N^3$; the two finer timings are single runs, so these exponents are indicative.

**Cylindrical blast.** The initial front is the circle of radius 0.2, anchored through its signed distance, and the network carries one front, for the outgoing shock; the contact and the inward rarefaction are

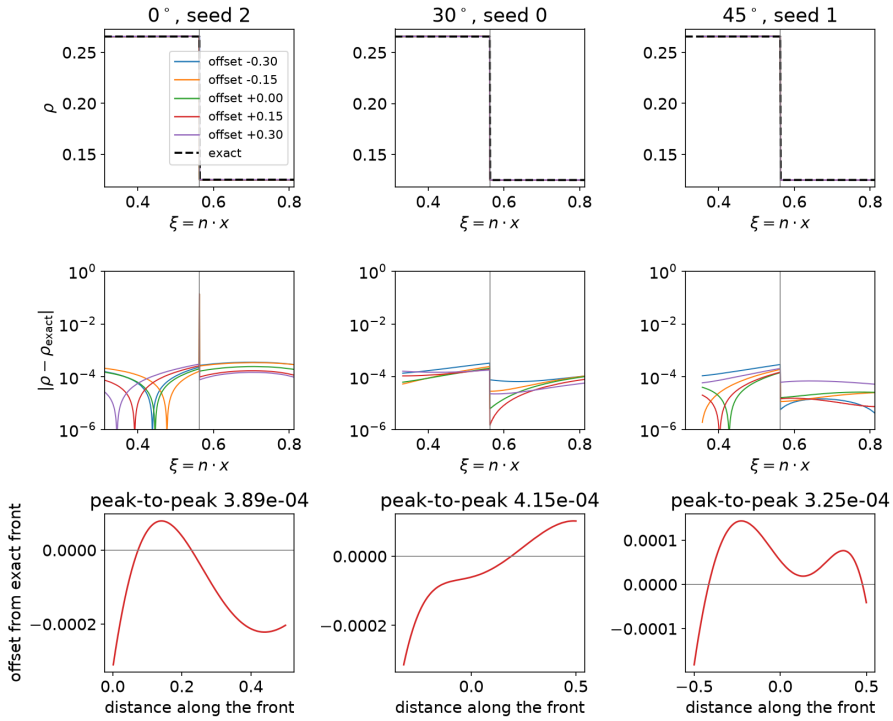



Figure 4: Best seed per angle, chosen by front error. Top: density along the front normal at five tangential offsets, which collapse onto one curve. Middle: the same as absolute error. Bottom: signed distance from the learned front to the exact plane along the front, with peak-to-peak deviation of 3.3 to $4.2 \times 10^{-4}$ on these seeds.

left to the smooth branch behind it. The reference is the classical solver at $N = 800^2$, whose shock radius depends on the density threshold used to locate it: $0.44764$ at $1.2$ times ambient and $0.44617$ at $1.5$ times, so we declared radius errors below $1.5 \times 10^{-3}$ unresolvable before running; between $N = 400^2$ and $800^2$ the interpolated reference radius changes by $3 \times 10^{-4}$. Every seed places the shock inside the reference radius, by 16 to 22 times that floor (Table 6), with an out-of-roundness of $3 \times 10^{-3}$ to $10^{-2}$. We report this as a negative result for curved fronts with this ansatz and budget; we have not isolated whether the unrepresented contact and fan, the anchoring or the optimisation is responsible.

Table 6: Cylindrical blast at $t = 0.15$, three seeds, 15 000 iterations. Ranges are over seeds.

| Seeds | reference radius | learned radius | radius error | out of round | $L^1(\rho)$ |
|---|---|---|---|---|---|
| 3/3 | 0.4462 | 0.4130 to 0.4221 | $2.41 \times 10^{-2}$ to $3.32 \times 10^{-2}$ | $3.03 \times 10^{-3}$ to $1.02 \times 10^{-2}$ | $2.83 \times 10^{-2}$ to $3.96 \times 10^{-2}$ |

**The boundary condition, and a comparison we discarded.** Our first two-dimensional comparison gave the classical solver zero-gradient extrapolation on all four sides while the network received exact Dirichlet data. The classical error then appeared not to converge on oblique shocks, with observed orders of $0.03$ at $30°$ and $0.04$ at $45°$ over $N = 400$ to $800$, against first order for the axis-aligned shock. For a shock crossing a lateral boundary the exact solution varies along that boundary, so zero-gradient extrapolation injects an error that refinement cannot remove. With exact boundary data the axis-aligned error changes by at most 2.2 percent, as it should, and the oblique cases recover convergence (Table 7, Figure 5). The classical error at $N = 800$ is then the same across angles to within 9 percent.

Table 7: The 30° oblique shock under both classical boundary treatments, with the front-tracking median. Both errors are against the exact planar solution. The classical $L^1$ here is sampled at cell centres, unlike the piecewise-constant score of Table 2, and the classical front error is the single most downstream midpoint crossing, not comparable with the network's median; Table 5 compares estimators.

| Method | boundary | $L^1(\rho)$ | front position error | seconds |
|---|---|---|---|---|
| Classical, $N = 50^2$ | zero-gradient | $1.03 \times 10^{-2}$ | $5.34 \times 10^{-3}$ | 0.12 |
| Classical, $N = 100^2$ | zero-gradient | $9.19 \times 10^{-3}$ | $3.10 \times 10^{-3}$ | 0.74 |
| Classical, $N = 200^2$ | zero-gradient | $8.49 \times 10^{-3}$ | $1.64 \times 10^{-3}$ | 6.69 |
| Classical, $N = 400^2$ | zero-gradient | $8.21 \times 10^{-3}$ | $8.32 \times 10^{-4}$ | 104.77 |
| Classical, $N = 800^2$ | zero-gradient | $8.05 \times 10^{-3}$ | $4.21 \times 10^{-4}$ | 1169.73 |
| Classical, $N = 50^2$ | exact Dirichlet | $3.30 \times 10^{-3}$ | $8.72 \times 10^{-3}$ | 0.12 |
| Classical, $N = 100^2$ | exact Dirichlet | $1.71 \times 10^{-3}$ | $6.25 \times 10^{-3}$ | 0.72 |
| Classical, $N = 200^2$ | exact Dirichlet | $8.86 \times 10^{-4}$ | $3.49 \times 10^{-3}$ | 6.46 |
| Classical, $N = 400^2$ | exact Dirichlet | $4.49 \times 10^{-4}$ | $1.51 \times 10^{-3}$ | 98.57 |
| Classical, $N = 800^2$ | exact Dirichlet | $2.28 \times 10^{-4}$ | $8.35 \times 10^{-4}$ | 1119.40 |
| Front tracking, this work | exact Dirichlet | $1.43 \times 10^{-4}$ | $7.56 \times 10^{-5}$ | 783 |

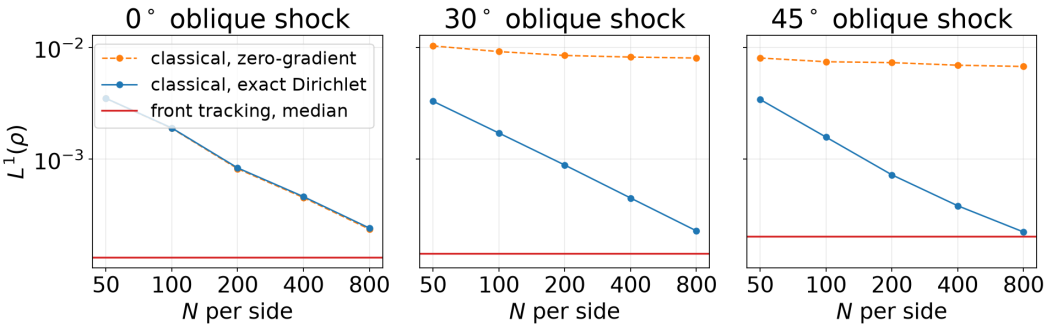



Figure 5: Classical $L^1$ error against resolution at the three angles under both boundary treatments, with the front-tracking median. The treatments coincide at 0°, where zero-gradient extrapolation is exact, and separate at every other angle.

## E WHERE THE ERROR LIES

Table 8 splits each seed's $L^1$ error by region of the exact solution. *Exact band* is the error within $5 \times 10^{-3}$ of an exact front position; *with swept strip* adds the strip between each learned front and its exact position, where the error equals the jump. The other regions exclude the swept strip. On Sod the learned fronts lie far enough from the exact ones that the exact band alone would book the strip to the star states; on the colliding problem the two definitions agree.

Table 8: Percentage of the density $L^1$ error by region, range over three seeds.

| | fronts | | | | | |
|---|---|---|---|---|---|---|
| Problem | exact band | with swept strip | star states | fan | far field | max. front displacement |
| Sod | 17–37 | 34–52 | 17–27 | 22–36 | 3–10 | $7.46 \times 10^{-3}$ to $1.81 \times 10^{-2}$ |
| Colliding | 42–61 | 42–61 | 27–41 | n/a | 13–23 | $2.75 \times 10^{-5}$ to $1.50 \times 10^{-4}$ |

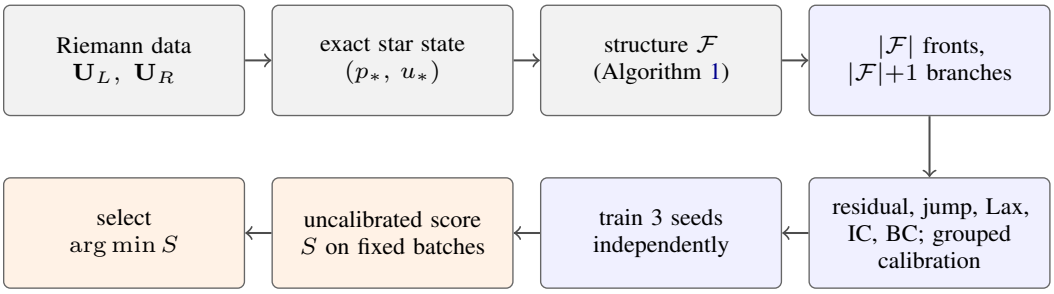
## F PLATEAU ANCHORING

An anchored branch is written $\rho = \rho_{\text{anchor}} e^{wc_\rho}$, $u = u_{\text{anchor}} + wc_u$, $p = p_{\text{anchor}} e^{wc_p}$, where the anchor is obtained by applying the shock relations to the neighbouring branch, or is the exact far-field state, and $w$ localises the learned correction. *Anchored on the front* takes $w = x - s_k$, which vanishes on the front, so the plateau satisfies the jump condition identically there; *anchored in the plateau* takes a Gaussian bump equal to one on the front. On Sod only the right far field can be anchored, because the left far field lies inside the rarefaction branch. Table 9 reports all six variants.

Table 9: Plateau treatments, three seeds each, 15 000 iterations, shared calibration. *Free branches* is the unanchored construction.

| | Sod | | Colliding | |
|---|---|---|---|---|
| Plateau treatment | median $L^1$ | spread | median $L^1$ | spread |
| Free branches | $5.87 \times 10^{-3}$ | 2.1× | $4.92 \times 10^{-4}$ | 5.5× |
| Anchored on the front | $5.20 \times 10^{-3}$ | 2.2× | $2.58 \times 10^{-3}$ | 33.0× |
| Anchored in the plateau | $5.04 \times 10^{-3}$ | 3.6× | $9.05 \times 10^{-4}$ | 12.9× |
| Plateau, adaptive width | $8.29 \times 10^{-3}$ | 4.4× | $3.44 \times 10^{-4}$ | 121.8× |
| Exact far field only | $1.27 \times 10^{-2}$ | 1.6× | $2.94 \times 10^{-4}$ | 22.6× |
| Exact far field + plateau | $4.72 \times 10^{-3}$ | 1.4× | $8.07 \times 10^{-5}$ | 2.1× |

## G RUN SELECTION




*The star state decides only which fronts exist; its values never reach the network, and no reference solution enters training or selection.*

Figure 6: End-to-end procedure. The exact star state of the Riemann data is used only to decide which fronts exist and their families; branch values are learned. Selection uses the uncalibrated score $S$ of Equation 6 on a fixed set of 24 batches shared by all runs.

The *free branches* variant of Table 9 was retrained for the anchoring study and reproduced the free construction's errors exactly on every seed; it is counted once below, which leaves 13 distinct groups in total and 18 distinct runs on each of Sod and the colliding problem. In the Sod variant anchored on the front, the jump defect is below $2 \times 10^{-5}$ on two seeds, and the best seed has the largest initial-condition defect, which excludes it.

## H CAUSAL TIME WEIGHTING

Weighting the residual and jump terms by $e^{-\kappa t/T}$ fits early times before late ones, which suits a problem whose late front position is a consequence of its early one. On the single-shock problem (Table 11) every schedule removes the failed seed that drives the 75.5× spread of uniform weighting, leaving spreads of 1.4 to 2.3×. At $\kappa = 3$ the median is $1.36 \times 10^{-4}$ against $1.58 \times 10^{-4}$ uniform, a difference within the seed spread; steeper schedules leave the final time almost unweighted and the median degrades. Causal weighting raises the floor and lowers the ceiling: uniform weighting's

Table 10: Selecting a run by the ground-truth-free score $S$ of Equation 6. *Selected* is the $L^1$ error of the lowest-score run, *best* the lowest error attained; a ratio of $1.00\times$ (bold) means the score found the best run. *All variants pooled* selects over every distinct run of a problem, and $r$ is the correlation of $\log S$ with $\log L^1$ over those runs. Anchoring variants are undefined on the two-rarefaction problem, which has no fronts.

| Problem | variant | runs | selected | median | best | sel./best | pooled $r$ |
|---|---|---|---|---|---|---|---|
| Sod | Free construction | 3 | $4.06 \times 10^{-3}$ | $5.87 \times 10^{-3}$ | $4.06 \times 10^{-3}$ | **1.00**× | n/a |
| Sod | Anchored on the front | 3 | $5.20 \times 10^{-3}$ | $5.20 \times 10^{-3}$ | $3.08 \times 10^{-3}$ | 1.69× | n/a |
| Sod | Anchored in the plateau | 3 | $3.76 \times 10^{-3}$ | $5.04 \times 10^{-3}$ | $3.76 \times 10^{-3}$ | **1.00**× | n/a |
| Sod | Plateau, adaptive width | 3 | $2.87 \times 10^{-3}$ | $8.29 \times 10^{-3}$ | $2.87 \times 10^{-3}$ | **1.00**× | n/a |
| Sod | Exact far field only | 3 | $1.26 \times 10^{-2}$ | $1.27 \times 10^{-2}$ | $1.26 \times 10^{-2}$ | **1.00**× | n/a |
| Sod | Exact far field + plateau | 3 | $4.23 \times 10^{-3}$ | $4.72 \times 10^{-3}$ | $4.23 \times 10^{-3}$ | **1.00**× | n/a |
| Sod | All variants pooled | 18 | $2.87 \times 10^{-3}$ | $5.98 \times 10^{-3}$ | $2.87 \times 10^{-3}$ | **1.00**× | 0.62 |
| Colliding | Free construction | 3 | $1.09 \times 10^{-4}$ | $4.92 \times 10^{-4}$ | $1.09 \times 10^{-4}$ | **1.00**× | n/a |
| Colliding | Anchored on the front | 3 | $1.65 \times 10^{-4}$ | $2.58 \times 10^{-3}$ | $1.65 \times 10^{-4}$ | **1.00**× | n/a |
| Colliding | Anchored in the plateau | 3 | $1.05 \times 10^{-4}$ | $9.05 \times 10^{-4}$ | $1.05 \times 10^{-4}$ | **1.00**× | n/a |
| Colliding | Plateau, adaptive width | 3 | $1.47 \times 10^{-4}$ | $3.44 \times 10^{-4}$ | $1.47 \times 10^{-4}$ | **1.00**× | n/a |
| Colliding | Exact far field only | 3 | $6.85 \times 10^{-5}$ | $2.94 \times 10^{-4}$ | $6.85 \times 10^{-5}$ | **1.00**× | n/a |
| Colliding | Exact far field + plateau | 3 | $8.07 \times 10^{-5}$ | $8.07 \times 10^{-5}$ | $3.97 \times 10^{-5}$ | 2.04× | n/a |
| Colliding | All variants pooled | 18 | $6.85 \times 10^{-5}$ | $3.19 \times 10^{-4}$ | $3.97 \times 10^{-5}$ | 1.73× | 0.95 |
| Two rarefaction | Free construction | 3 | $1.05 \times 10^{-1}$ | $1.36 \times 10^{-1}$ | $1.05 \times 10^{-1}$ | **1.00**× | n/a |

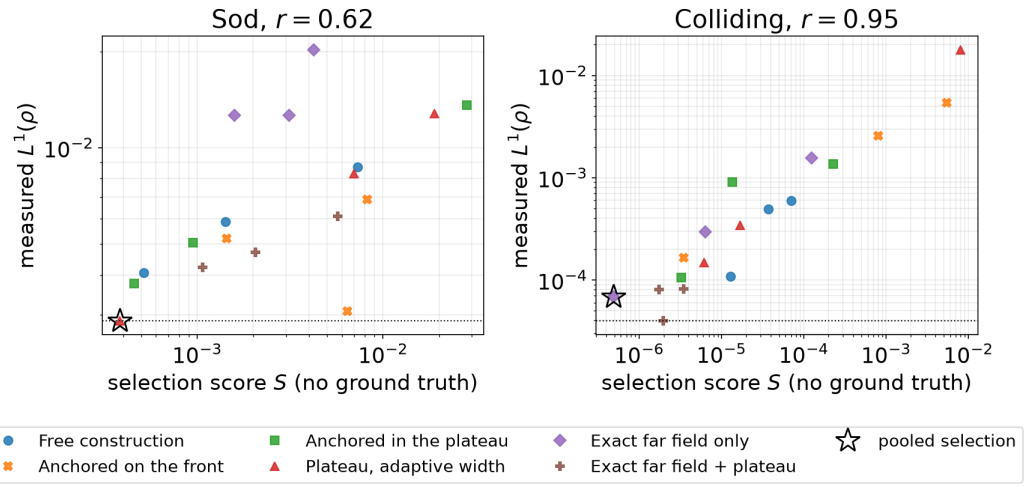



Figure 7: Selection score against measured error for the 18 distinct runs of each problem. On the colliding problem $r = 0.95$ with the error spanning 2.7 decades. On Sod ($r = 0.62$) the far-field-only variant contributes most of the scatter; leaving it out raises $r$ to 0.81. The open star is the pooled selection; the dotted line is the best run.

best run, $2.80 \times 10^{-5}$, is better than any causal run. On the lone contact $\kappa = 6$ improves the median from $1.35 \times 10^{-3}$ to $4.53 \times 10^{-4}$; on the lone rarefaction, which has no front, it changes the median by about ten percent. Applied to the entropy control-volume objective of Table 13, we used the loss-dependent weights of Wang et al. (2024), $w_i = \exp(-\varepsilon \sum_{j<i} \mathcal{L}_j)$, over 16 time bins of 32 control volumes each. With $\varepsilon = 0$, which is uniform weighting on the binned sampling, the median was $4.05 \times 10^{-2}$, against $3.48 \times 10^{-2}$ for the 128 unbinned control volumes; causal weighting at $\varepsilon = 1$ and 10 then moved it to $4.30 \times 10^{-2}$ and $6.65 \times 10^{-2}$.

Table 11: Causal weighting on the single-shock problem, three seeds per row.

| Weighting | median $L^1$ | best | worst | spread |
|---|---|---|---|---|
| uniform | $1.58 \times 10^{-4}$ | $2.80 \times 10^{-5}$ | $2.11 \times 10^{-3}$ | 75.5× |
| $\kappa = 1$ | $1.76 \times 10^{-4}$ | $1.10 \times 10^{-4}$ | $2.47 \times 10^{-4}$ | 2.3× |
| $\kappa = 3$ | $1.36 \times 10^{-4}$ | $9.74 \times 10^{-5}$ | $2.12 \times 10^{-4}$ | 2.2× |
| $\kappa = 6$ | $2.56 \times 10^{-4}$ | $2.08 \times 10^{-4}$ | $2.85 \times 10^{-4}$ | 1.4× |
| $\kappa = 12$ | $3.27 \times 10^{-4}$ | $2.26 \times 10^{-4}$ | $4.46 \times 10^{-4}$ | 2.0× |

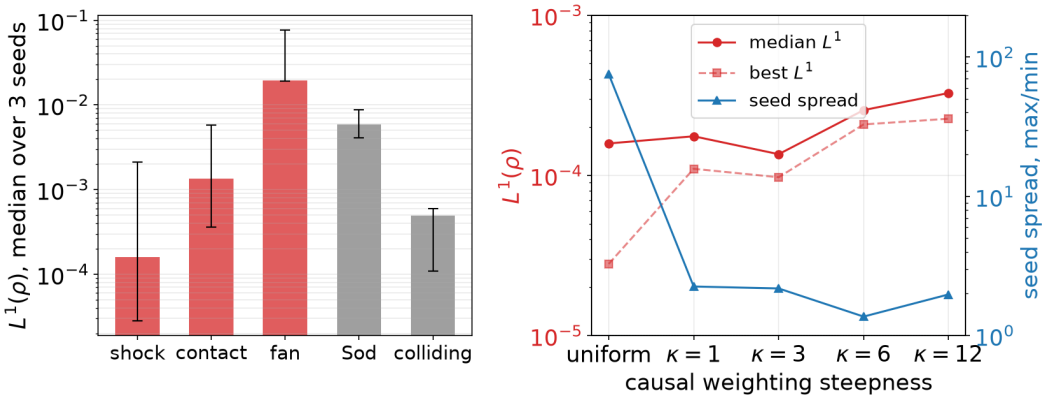



Figure 8: Left: error by wave type against the full Riemann problems. Right: seed spread and error against causal steepness on the single-shock problem.

## I Neural baselines

Table 12 compares the construction on all three Riemann problems with a strong-form PINN, a strong form widened to match the relaxation networks' parameter count, and two relaxation networks, an adaptation of Zhou & Ma (2024) that relaxes the energy flux or all fluxes, trained for the same 15 000 iterations rather than the original schedule. Table 13 places the construction against a wider set of objective, training and architectural alternatives on Sod. The construction with exact far field and plateau is the most accurate neural entry. The general construction ($5.87 \times 10^{-3}$) is on a par with the hand-specified single front with Lax ($5.57 \times 10^{-3}$), which it generalises; the best method without a front, defect correction, is 4.3× worse than the general construction. A separate repetition of the jump-only front, trained in the study of the hybrid representations, gave a median of $7.18 \times 10^{-2}$ (spread 7.1×), consistent with the $7.08 \times 10^{-2}$ in the table. A classical solver at $N = 256$ is more accurate than every neural entry and takes a third of a second. Table 14 covers further formulations from single runs, shown for coverage and not ranking.

## J Classical comparison details

Table 15 gives the classical error and cost against resolution. Exponents are fitted over $N \geq 512$, because the coarsest grids are dominated by fixed overhead: on the colliding problem the pairwise cost exponent rises from 1.31 at $N = 64 \to 128$ to 2.44 at $2048 \to 4096$, then falls to 2.32, 2.17 and 2.12 over the three finest pairs. The fitted exponent, 2.23, is therefore above the local one, and the fitted extrapolation of Table 3 (11 810 s) is about 13 percent above a last-pair extrapolation: it is generous to the network, not conservative. Pairwise convergence orders on the colliding problem range from 0.79 to 1.29. Table 16 gives the equal-accuracy cost for the free construction at its median error and median training time, taken independently over three seeds, a different configuration from Table 3; Table 17 compares the finest classical run executed against the selected result, with no fitting. At a fixed resolution of $N = 256$ the network ranges from 1.65× worse than the classical solver on Sod to 14× better on the colliding problem and 23× worse on the two-rarefaction problem. The

Table 12: Strong-form and relaxation networks against the construction, three seeds each, 15 000 iterations, all on 200 001 evaluation points. Bold marks the lowest median on each problem.

| Problem | method | parameters | median $L^1(\rho)$ | best | spread |
|---|---|---|---|---|---|
| Colliding | Strong-form PINN | 66,819 | $4.20 \times 10^{-1}$ | $3.31 \times 10^{-1}$ | 1.3× |
| Colliding | Strong form, width-matched | 137,267 | $4.32 \times 10^{-1}$ | $2.85 \times 10^{-1}$ | 1.5× |
| Colliding | Relaxation, energy flux | 133,380 | $2.47 \times 10^{-1}$ | $1.34 \times 10^{-1}$ | 2.1× |
| Colliding | Relaxation, all fluxes | 133,638 | $1.90 \times 10^{-1}$ | $1.88 \times 10^{-1}$ | 1.1× |
| Colliding | Front tracking, this work | 280,335 | $\mathbf{4.92 \times 10^{-4}}$ | $1.09 \times 10^{-4}$ | 5.5× |
| Sod | Strong-form PINN | 66,819 | $8.27 \times 10^{-2}$ | $5.83 \times 10^{-2}$ | 2.5× |
| Sod | Strong form, width-matched | 137,267 | $5.43 \times 10^{-2}$ | $4.98 \times 10^{-2}$ | 2.1× |
| Sod | Relaxation, energy flux | 133,380 | $4.67 \times 10^{-2}$ | $2.72 \times 10^{-2}$ | 2.3× |
| Sod | Relaxation, all fluxes | 133,638 | $7.88 \times 10^{-2}$ | $7.80 \times 10^{-2}$ | 1.1× |
| Sod | Front tracking, this work | 209,163 | $\mathbf{5.87 \times 10^{-3}}$ | $4.06 \times 10^{-3}$ | 2.1× |
| Two rarefaction | Strong-form PINN | 66,819 | $1.37 \times 10^{-1}$ | $1.02 \times 10^{-1}$ | 2.4× |
| Two rarefaction | Strong form, width-matched | 137,267 | $7.54 \times 10^{-2}$ | $6.29 \times 10^{-2}$ | 4.8× |
| Two rarefaction | Relaxation, energy flux | 133,380 | $9.96 \times 10^{-2}$ | $9.67 \times 10^{-2}$ | 1.3× |
| Two rarefaction | Relaxation, all fluxes | 133,638 | $\mathbf{7.38 \times 10^{-2}}$ | $6.81 \times 10^{-2}$ | 1.1× |
| Two rarefaction | Front tracking, this work | 66,819 | $1.36 \times 10^{-1}$ | $1.05 \times 10^{-1}$ | 2.7× |

network appears at a single training budget of 15 000 iterations; we did not trace its own cost-accuracy curve on the Euler problems.

**Timing caveats.** One-dimensional classical timings are the minimum of five repeats, except the colliding runs at $N = 16\,384$ and $32\,768$, which were timed once; two-dimensional classical timings are the minimum of three repeats up to $N = 200^2$ and single runs at $400^2$ and $800^2$. The one-dimensional classical runs use the two-dimensional code on an $N \times 4$ strip. On the colliding problem at $N = 1024$, 2048 and 4096, best of three interleaved repeats, the strip takes 1.67, 2.57 and 1.92 times as long as an $N \times 1$ line, with the density $L^1$ error differing by at most 0.21 percent; another job shared the machine, so the factor is approximate, and it was not measured at the finest resolutions. Applied to the finest colliding run, it would reduce the measured serial advantage from 5.0× to between 2.0 and 3.0×. The solver is vectorised NumPy on a CPU; a compiled implementation would be faster still. Both effects favour the network in every ratio we report.

**Concurrency and sequential depth.** Three seeds are independent runs. On the single GPU used, in a separate benchmark of 3000-iteration runs of the free construction measured once per concurrency level, two concurrent runs each took 1.36× their solo time and three took 2.04×, so aggregate throughput saturates at about 1.47×; applied to the selected colliding variant this gives an estimated 1160 s for three concurrent seeds, against a measured 1614 s serially. A classical solve cannot be shortened by independent runs: its step count grows linearly with $N$ under the CFL condition and each step depends on the last. Spatial parallelism reduces the cost per step, not the number of steps. This is a statement about sequential depth, not a wall-clock advantage on one device.

## K NONCONSERVATIVE PRODUCTS

**System and the path family.** We use the two-equation system

$$\partial_t h + \partial_x q = 0, \qquad \partial_t q + \partial_x\big(q^2/h\big) + q\,\partial_x\big(h^2/2\big) = 0, \tag{7}$$

with characteristic speeds $u \pm h\sqrt{u}$, $u = q/h$, studied numerically by Castro et al. (2008). The product $q\,\partial_x(h^2/2)$ is not the derivative of any function of $(h, q)$, so at a discontinuity it is defined only once a path in state space is chosen (Dal Maso et al., 1995), and the jump conditions depend on that choice. For a shock between $(h_L, q_L)$ and $(h_R, q_R)$ with speed $\sigma$ they read

$$\sigma\,[h] = [q]\,, \qquad \sigma\,[q] = \big[q^2/h\big] + \bar{q}\,\big[h^2/2\big]\,, \tag{8}$$

where $\bar{q}$ is the discharge carried along the path. We parameterise a family of paths by a weight $w \in [0, 1]$ with $\bar{q} = w\,q_L + (1 - w)\,q_R$, which depends on the local states and is therefore genuinely

Table 13: Sod at $t = 0.2$. Neural rows are medians over three seeds with spread. The rows above the first rule were evaluated on a 2001-point grid, before the evaluation grid was refined; the others on 200 001 points. *Shock position* is a windowed midpoint crossing for the first eight rows, the median distance of the tracked front from the exact shock for the three front-tracking rows, and the largest front error for ours; it was not measured for the relaxation study (–). *Gated* entries have at least one seed that failed the shock-resolution test: all three for the hard initial condition, one of three for the hard initial condition followed by L-BFGS and for self-similar coordinates, and two of three for front tracking with jump conditions only, judged on the jump across the tracked front. Bold marks the most accurate neural entry; the classical rows at $N = 256$ and 1024 are more accurate still.

| Method | median $L^1(\rho)$ | spread | shock position | seconds |
|---|---|---|---|---|
| Entropy control volumes | $3.48 \times 10^{-2}$ | 1.6× | $2.66 \times 10^{-2}$ | 96 |
| Entropy control volumes, then L-BFGS | $3.84 \times 10^{-2}$ | 1.7× | $2.05 \times 10^{-2}$ | 101 |
| Entropy-guided collocation | $4.27 \times 10^{-2}$ | 2.3× | $1.46 \times 10^{-2}$ | 107 |
| Hard initial condition | $1.21 \times 10^{-1}$ | 1.1× | gated | 100 |
| Hard initial condition, then L-BFGS | $5.08 \times 10^{-2}$ | 2.7× | gated | 106 |
| Self-similar coordinates | $6.44 \times 10^{-2}$ | 1.8× | gated | 165 |
| Hard Rankine-Hugoniot | $1.20 \times 10^{-1}$ | 1.2× | $1.16 \times 10^{-2}$ | 610 |
| Defect correction | $2.51 \times 10^{-2}$ | 1.3× | $4.35 \times 10^{-2}$ | 298 |
| Front tracking, jump conditions only | $7.08 \times 10^{-2}$ | 11.7× | gated | 417 |
| Front tracking, Lax weight 10 | $2.55 \times 10^{-2}$ | 2.7× | $4.06 \times 10^{-2}$ | 508 |
| Front tracking, single front with Lax | $5.57 \times 10^{-3}$ | 2.8× | $4.51 \times 10^{-3}$ | 522 |
| Strong-form PINN | $8.27 \times 10^{-2}$ | 2.5× | – | 175 |
| Strong form, width-matched | $5.43 \times 10^{-2}$ | 2.1× | – | 194 |
| Relaxation, energy flux | $4.67 \times 10^{-2}$ | 2.3× | – | 194 |
| Relaxation, all fluxes | $7.88 \times 10^{-2}$ | 1.1× | – | 167 |
| General construction, this work | $5.87 \times 10^{-3}$ | 2.1× | $7.81 \times 10^{-3}$ | 624 |
| Exact far field + plateau, this work | $\mathbf{4.72 \times 10^{-3}}$ | 1.4× | $8.92 \times 10^{-3}$ | 927 |
| Classical finite volume, $N = 64$ | $1.16 \times 10^{-2}$ | n/a | n/a | 0.05 |
| Classical finite volume, $N = 256$ | $3.55 \times 10^{-3}$ | n/a | n/a | 0.32 |
| Classical finite volume, $N = 1024$ | $1.13 \times 10^{-3}$ | n/a | n/a | 3.41 |

Table 14: Single-run survey on Sod on a 2001-point grid, for coverage only; not comparable to Table 13.

| Method (single run) | $L^1(\rho)$ | shock position | seconds |
|---|---|---|---|
| Strong form | $7.35 \times 10^{-2}$ | gated | 423 |
| Strong form, Fourier features | $2.35 \times 10^{-2}$ | $3.86 \times 10^{-2}$ | 302 |
| Strong form, KAN | $1.89 \times 10^{-2}$ | $3.84 \times 10^{-2}$ | 1219 |
| Finite-volume flux loss (HLLC) | $4.21 \times 10^{-1}$ | gated | 296 |
| Weak form, dual norm | $1.44 \times 10^{-1}$ | gated | 663 |
| Entropy control volumes | $3.22 \times 10^{-2}$ | $2.22 \times 10^{-2}$ | 122 |
| Spectral dual norm $H^{-1}$ | $2.95 \times 10^{-2}$ | $1.99 \times 10^{-2}$ | 1998 |
| Dual norm with entropy | $3.24 \times 10^{-2}$ | $2.74 \times 10^{-3}$ | 1855 |

nonconservative for every $w$. For left state $(1, 1)$ and right depth 0.7 this gives an exact shock speed that rises monotonically from 1.562260 at $w = 0$ to 1.771362 at $w = 1$. The physically selected path is the vanishing-viscosity limit of the regularisation $-\varepsilon D\,\partial_{xx}$. For $D = I$ we solve the travelling-wave equation by shooting along the unstable manifold and obtain speed 1.666939, and the discharge carried by the profile corresponds to $w = 0.495236$; the experiments use 0.495237. The jump condition with this weight returns the same speed to machine precision; this checks the weight parameterisation and its implementation, and is not an independent confirmation of the speed.

**A parameterisation we had to discard.** A first version parameterised the family by a constant carried discharge $\bar{q}$. A constant $\bar{q}$ makes $\bar{q}\,\partial_x(h^2/2)$ an exact derivative, so the system becomes

Table 15: Classical error and wall-clock against resolution; exponents of $L^1 \sim N^{-a}$ and cost $\sim N^b$ fitted over $N \geq 512$. The two finest resolutions were run on the colliding problem only, and timed once.

| | Sod | | Colliding | | Two rarefaction | |
|---|---|---|---|---|---|---|
| $N$ | $L^1(\rho)$ | seconds | $L^1(\rho)$ | seconds | $L^1(\rho)$ | seconds |
| 64 | $1.16 \times 10^{-2}$ | 0.05 | $3.25 \times 10^{-2}$ | 0.04 | $2.13 \times 10^{-2}$ | 0.04 |
| 128 | $6.35 \times 10^{-3}$ | 0.11 | $1.33 \times 10^{-2}$ | 0.11 | $1.15 \times 10^{-2}$ | 0.09 |
| 256 | $3.55 \times 10^{-3}$ | 0.32 | $7.00 \times 10^{-3}$ | 0.30 | $5.95 \times 10^{-3}$ | 0.24 |
| 512 | $1.99 \times 10^{-3}$ | 1.00 | $3.78 \times 10^{-3}$ | 0.92 | $3.03 \times 10^{-3}$ | 0.75 |
| 1024 | $1.13 \times 10^{-3}$ | 3.41 | $2.01 \times 10^{-3}$ | 3.14 | $1.53 \times 10^{-3}$ | 2.56 |
| 2048 | $6.48 \times 10^{-4}$ | 18.77 | $1.07 \times 10^{-3}$ | 15.28 | $7.67 \times 10^{-4}$ | 12.90 |
| 4096 | $3.77 \times 10^{-4}$ | 91.77 | $5.14 \times 10^{-4}$ | 83.16 | $3.84 \times 10^{-4}$ | 66.98 |
| 8192 | $2.21 \times 10^{-4}$ | 452.19 | $2.78 \times 10^{-4}$ | 415.67 | $1.92 \times 10^{-4}$ | 337.69 |
| 16384 | – | – | $1.61 \times 10^{-4}$ | 1874.56 | – | – |
| 32768 | – | – | $7.76 \times 10^{-5}$ | 8142.91 | – | – |
| *fitted exponent* | *0.79* | *2.24* | *0.93* | *2.23* | *0.99* | *2.23* |

Table 16: Equal-accuracy cost for the free construction: median $L^1$ and median training time over three seeds, taken independently.

| Problem | median $L^1$ | seconds | equal-accuracy $N$ | classical seconds | classical faster by |
|---|---|---|---|---|---|
| Sod | $5.87 \times 10^{-3}$ | 624 | 141 | 0.1 | 4731× |
| Colliding | $4.92 \times 10^{-4}$ | 760 | 4303 | 93.3 | 8× |
| Two rarefaction | $1.36 \times 10^{-1}$ | 292 | | beaten by $N = 64$ at 0.04 s | |

conservative and a path-conservative scheme converges, as the Lax-Wendroff theorem guarantees (Lax & Wendroff, 1960). That removed the phenomenon under study; the state-dependent weight is required.

**Path resolvability.** We sweep five weights, compute the exact speed of each, and measure the speed computed by a path-conservative finite-volume scheme of Lax-Friedrichs type at $N = 200$ to 6400 and by the one-dimensional front construction at 6000, 24 000 and 96 000 iterations, three seeds each. The front construction is given the prescribed weight and imposes the corresponding jump condition, so a slope near one tests the optimiser, not whether the method can discover the path. The path response slope is the least-squares slope of computed against exact speed, fitted for the front construction to the median of the three seeds at each weight; a method that resolves the prescribed path has slope 1. The scheme's speed is read from the cell where the depth first crosses its midpoint, which quantises it in steps of 0.025 at $N = 200$ and 0.00078 at $N = 6400$; at $N = 200$ the five weights produce only two distinct speeds, so that slope is below the measurement's resolution. Table 18 and Figure 11 report the results.

The construction has slope 1.004, 0.998 and 0.998, and its median error pooled over weights falls from 0.061 to 0.026 to 0.009 percent as the budget grows 16×; at individual weights the decrease is not always monotone, and at the largest budget the per-run errors range from 0.0003 to 0.11 percent. The scheme's slope is −0.12 at 400 cells and −0.26 at 6400: its answer depends on the prescribed path only weakly and with the wrong sign, and more so on finer grids, while its median error stays near 4 percent over a 16× refinement. Its error is smallest at the viscous path, 0.085 percent at $N = 6400$, but that is about two quantisation steps and only indicative. The two errors are different in kind: one falls with optimisation budget, the other does not fall with resolution. Castro et al. (2008) demonstrated the convergence error on this system and observed that Lax-Friedrichs Hugoniot curves do not respond to a perturbed path on a two-layer system; what we add is a slope across a path family and its behaviour under refinement. Path sensitivity is scheme dependent: Rhebergen et al. (2008) found the effect of the path on discontinuous Galerkin solutions small, Ren & Parés (2025) give WENO finite-difference schemes whose solutions distinguish two path families, and in-cell

Table 17: Finest classical run executed against the selected result, with no extrapolation. Ours is the variant with the most accurate selected run, chosen with the reference, and its seconds are an estimate for three seeds run concurrently (text); run serially they take 1614 s on the colliding problem, a speed ratio of 5.0×. *Accuracy* is classical error over ours and *speed* classical time over ours.

| | finest classical run | | ours, selected | | | |
|---|---|---|---|---|---|---|
| Problem | $L^1(\rho)$ | seconds | $L^1(\rho)$ | seconds | accuracy | speed |
| Sod | $2.21 \times 10^{-4}$ | 452 | $2.87 \times 10^{-3}$ | 2471 | 0.08× | 0.18× |
| Colliding | $7.76 \times 10^{-5}$ | 8143 | $6.85 \times 10^{-5}$ | 1160 | 1.13× | 7.02× |
| Two rarefaction | $1.92 \times 10^{-4}$ | 338 | $1.05 \times 10^{-1}$ | 640 | $1.8 \times 10^{-3}$ | 0.53× |

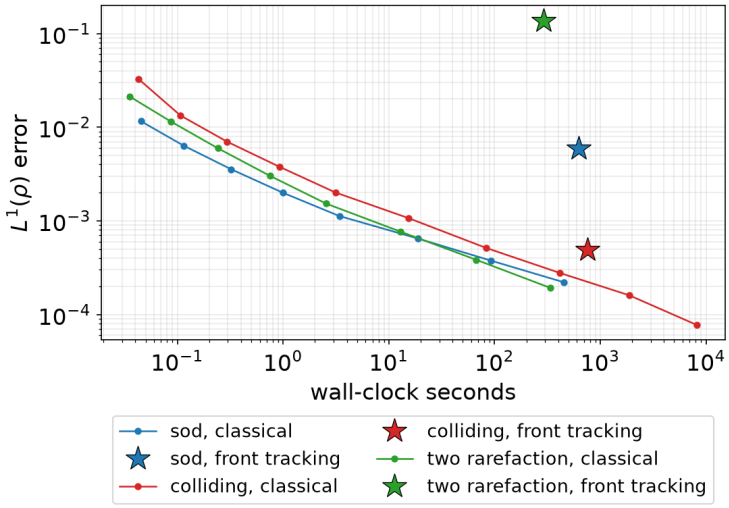
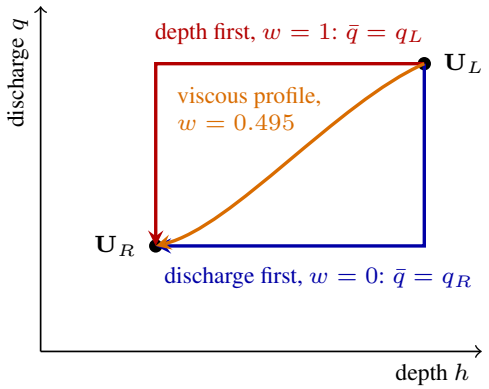



Figure 9: Cost against accuracy on the three one-dimensional problems. Lines are the classical solver from $N = 64$ to $N = 8192$, and to $N = 32\,768$ on the colliding problem; stars are the free construction, median over three seeds.

discontinuous-reconstruction schemes capture isolated shocks at the prescribed path (Chalons, 2020; Pimentel-García et al., 2022; 2024). Our negative slope is specific to the scheme and system tested.

**Magnitude against structure of the viscosity.** Across scalings of the scheme's numerical viscosity from 0.5 to 1.0 of its stability limit, a factor of two, its computed speed at $N = 3200$ varies by 0.19 to 0.35 percent (maximum minus minimum) across the three paths swept, two to four quantisation steps, against an error of 8.5 to 8.8 percent on the discharge-first path. This is consistent with the theory, in which the vanishing-viscosity limit depends on the structure of the regularisation matrix and not on an overall scalar.

**A two-dimensional result we traced to our own extension.** We extended the system to two dimensions and initially measured that a dimensionally split path-conservative scheme selects a shock speed that drifts by about 7 percent with the angle between shock and mesh, without shrinking under refinement. The cause was our extension. Writing the nonconservative term axis by axis, $q_x \partial_x (h^2/2)$ and $q_y \partial_y (h^2/2)$, is not rotationally invariant: projected onto a shock normal $\mathbf{n}$ it carries a factor $n_x^3 + n_y^3$ instead of 1. The front construction had been given the invariant form $|\mathbf{q}| \nabla (h^2/2)$, so the two methods were solving different equations; given that form, the front construction's speed error is 0.12 to 0.27 percent at 0° and at most 0.07 percent at 15° to 45°, over three weights and three seeds. Our control, a constant carried discharge, was both conservative and invariant and so could not separate the explanations. Holding nonconservativity fixed and varying only invariance (Table 19), the orientation spread of the invariant form shrinks with $N$ while that of the axis-wise form does




$$\sigma\,[h] = [q]$$
$$\sigma\,[q] = [q^2/h] + \bar{q}\,[h^2/2]$$
$$\bar{q} = \frac{1}{[h^2/2]} \int_{\text{path}} q \; \mathrm{d}\Big(\tfrac{h^2}{2}\Big)$$

family used here:
$\bar{q} = w\,q_L + (1-w)\,q_R,\ w \in [0,1]$

Figure 10: Paths in state space between the two sides of a shock, drawn schematically with a common right state. The nonconservative product contributes $\bar{q}\,[h^2/2]$ to the momentum jump, where $\bar{q}$ is the discharge carried along the chosen path: moving depth first carries $q_L$, moving discharge first carries $q_R$, and the vanishing-viscosity profile carries a value between. For a fixed right depth each choice gives a different shock speed and right discharge.

Table 18: Path response slope and relative speed error. Front-tracking values are over three seeds per weight; the median error pools all weights. The viscous path is $w = 0.495237$.

| Method | budget | path response slope | median rel. error | error at viscous path |
|---|---|---|---|---|
| Path-conservative FV | 200 cells | $+0.144$ | 3.323% | 1.233% |
| Path-conservative FV | 400 cells | $-0.120$ | 4.849% | 1.608% |
| Path-conservative FV | 800 cells | $-0.191$ | 4.656% | 0.671% |
| Path-conservative FV | 1600 cells | $-0.239$ | 4.366% | 0.202% |
| Path-conservative FV | 3200 cells | $-0.230$ | 4.124% | 0.249% |
| Path-conservative FV | 6400 cells | $-0.256$ | 4.148% | 0.085% |
| Front tracking (ours) | 6000 iterations | $+1.004$ | 0.061% | 0.119% |
| Front tracking (ours) | 24000 iterations | $+0.998$ | 0.026% | 0.146% |
| Front tracking (ours) | 96000 iterations | $+0.998$ | 0.009% | 0.024% |

not. We report this because the error was easy to make and the control that failed to catch it looked adequate.

Table 19: Spread of the computed shock speed across angles $0°$ to $45°$, mean over three path weights, as a percentage of the exact speed, for the split path-conservative scheme.

| 2D carried-discharge form | $N = 200$ | $N = 400$ | $N = 800$ |
|---|---|---|---|
| Axis-wise (not rotationally invariant) | 7.50% | 6.55% | 7.24% |
| Invariant, still nonconservative | 1.97% | 1.49% | 0.66% |

**The construction cannot find the orientation on its own.** Making the shock normal a trainable parameter initialised away from the truth, the two-dimensional construction recovered it in one of six configurations and otherwise converged tens of degrees away, with median speed errors of 19 to 41 percent (Table 20). In the one success, which started $15°$ from the true normal, the median seed's angle error is $0.09°$, so the true normal is at least a local minimum there; a start the same $15°$ from a different true normal failed, and since we did not record the objective at the failed runs we cannot say whether they stopped short of that minimum or found lower ones. The test covers one path weight and one training budget, 12 000 iterations of Adam at $3 \times 10^{-3}$. For Euler, the anchored level set of Equation 5 takes the initial front from the data and learns only its motion.

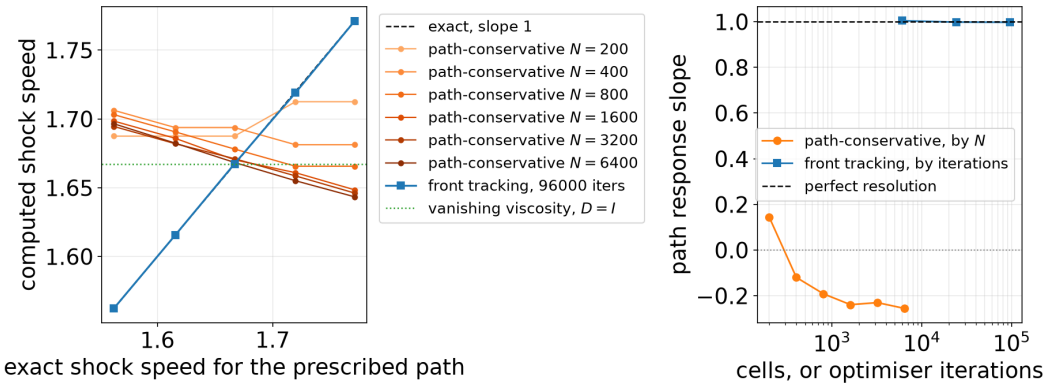



Figure 11: Left: computed against exact shock speed as the prescribed path varies. Right: path response slope against refinement budget, cells for the scheme and iterations for the construction.

Table 20: Two-dimensional nonconservative construction with a trainable normal, three seeds per row, path weight $0.495237$, $12\,000$ iterations.

| True angle | initial angle | median recovered | angle error, median (range) | median speed error |
|---|---|---|---|---|
| $15°$ | $0°$ | $-28.81°$ | $43.81°$ (42.24 to 50.80) | 25.66% |
| $15°$ | $60°$ | $76.71°$ | $61.71°$ (60.74 to 62.28) | 30.04% |
| $30°$ | $0°$ | $-29.07°$ | $59.07°$ (57.78 to 59.78) | 39.80% |
| $30°$ | $60°$ | $84.19°$ | $54.19°$ (52.21 to 54.58) | 19.43% |
| $45°$ | $0°$ | $-22.16°$ | $67.16°$ (62.55 to 68.14) | 40.77% |
| $45°$ | $60°$ | $44.96°$ | $0.09°$ (0.04 to 1.28) | 0.04% |