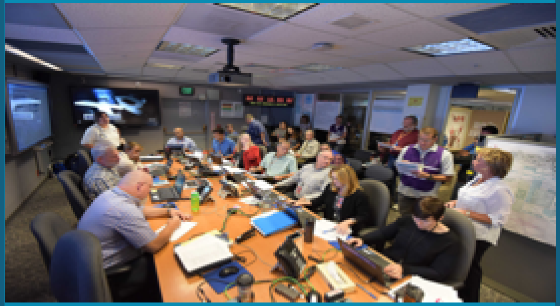
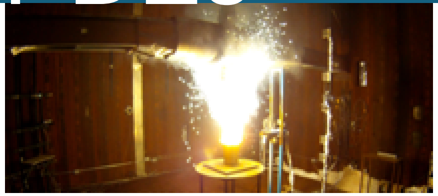
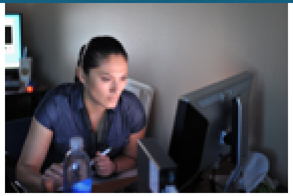
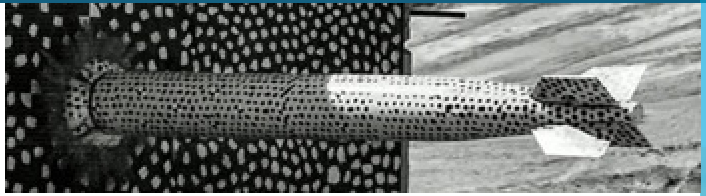
SAND2021-0201PE

Control volume PINNs: a method for solving inverse problems with hyperbolic PDEs









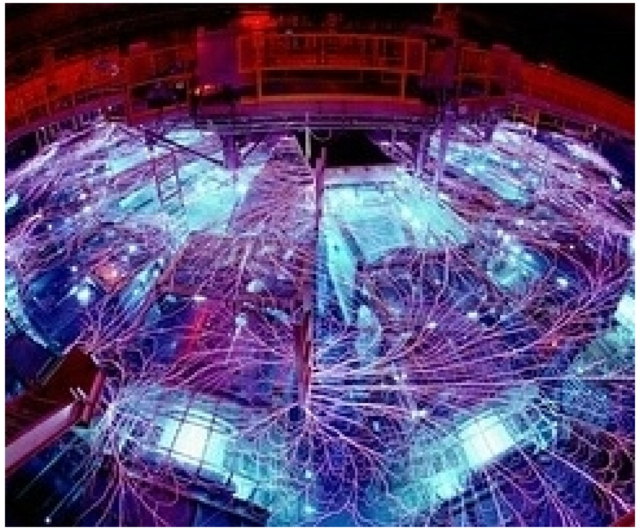
Ravi G. Patel

Center for Computing Research

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.
SAND No:

Materials in extreme environments



Given experimental/high fidelity simulation data from a system,

Find a mathematical model that describes the system

Experiments/simulations generate **noisy, biased, sparse** data

Hyperbolic conservation laws are suitable models for many systems

Typical hyperbolic PDE,

$$\begin{aligned} \partial_t \boldsymbol{u} + \nabla \cdot \boldsymbol{F}(\boldsymbol{u}) &= 0 \qquad && x, t \in \Omega \\ \boldsymbol{u} &= \boldsymbol{u}_0 && t = 0 \\ \boldsymbol{F}(\boldsymbol{u}) \cdot \hat{n} &= g && x \in \Gamma_- \end{aligned}$$

Inverse problems with hyperbolic conservation laws

Typical hyperbolic PDE,

$$\begin{aligned} \partial_t \boldsymbol{u} + \nabla \cdot \boldsymbol{F}(\boldsymbol{u}) = 0 \qquad & x, t \in \Omega \\ \boldsymbol{u} = \boldsymbol{u}_0 \qquad & t = 0 \\ \boldsymbol{F}(\boldsymbol{u}) \cdot \hat{n} = g \qquad & x \in \Gamma_- \end{aligned}$$

Assume we are given the solution,

$$\tilde{\boldsymbol{u}}(\boldsymbol{x}, t)$$

What is $\boldsymbol{F}(\boldsymbol{u})$?

$$\underset{\tilde{\boldsymbol{F}}}{\text{argmin}} \; ||\tilde{\boldsymbol{u}} - \boldsymbol{u}||_\chi$$

$$\text{s.t. PDE}$$

Inverse problems with hyperbolic conservation laws

Ingredients needed:

- A numerical method for the PDE
- A parameterization for $\boldsymbol{F}(\boldsymbol{u})$

Challenges,

- PDE forms discontinuities
- $\boldsymbol{F}(\boldsymbol{u})$ must produce a well-posed IBVP

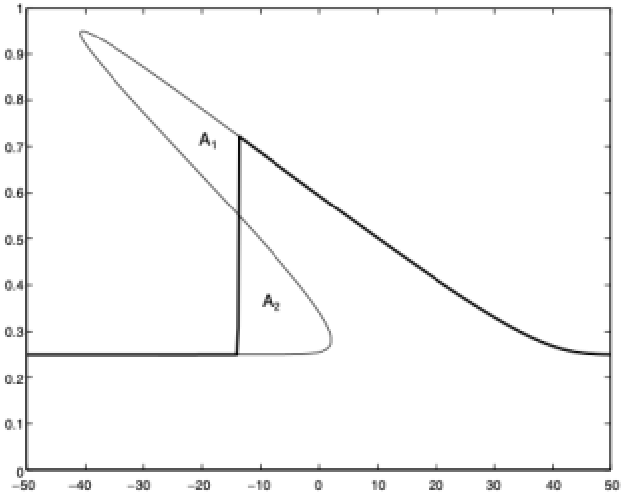
Inverse problems with hyperbolic conservation laws

Ingredients needed:

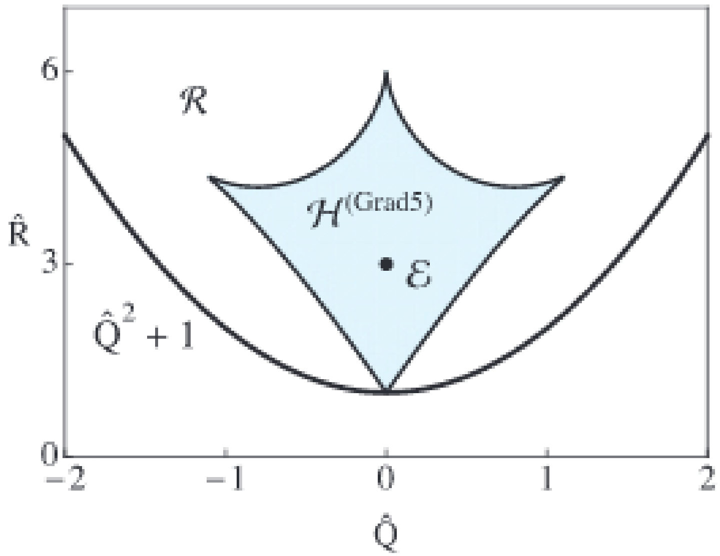
- A numerical method for the PDE
- A parameterization for $\boldsymbol{F}(\boldsymbol{u})$

Challenges,

- PDE forms discontinuities
- $\boldsymbol{F}(\boldsymbol{u})$ must produce a well-posed IBVP



Shock solution to the traffic flow equation[1]



Hyperbolic region for Grad's 5 moment equations[2]

[1] LeVeque, *Finite Volume Methods for Hyperbolic Problems,* 2004

[2] Schaerer and Torrilhon, *Commun. Comput. Phys.,* 2015

Inverse problems with hyperbolic conservation laws: Equation of state discovery

Euler equations,

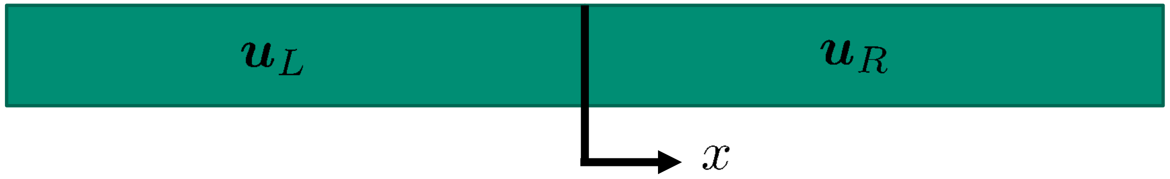
$$\partial_t \rho + \partial_x \rho u = 0$$
$$\partial_t \rho u + \partial_x (\rho u^2 + p) = 0$$
$$\partial_t E + \partial_x u(E + p) = 0$$
$$E = \rho e + \frac{1}{2}\rho u^2$$

Equation of state (EOS) provides closure,

- Entropy: $s = s(\rho, e)$
- Pressure $p = -\rho^2 (\partial_\rho s)(\partial_e s)^{-1}$

We will consider solutions to Riemann problems,

$$\boldsymbol{u}(x,0) = \boldsymbol{u}_L \quad \text{if } x < 0$$
$$\boldsymbol{u}(x,0) = \boldsymbol{u}_R \quad \text{if } x \geq 0$$

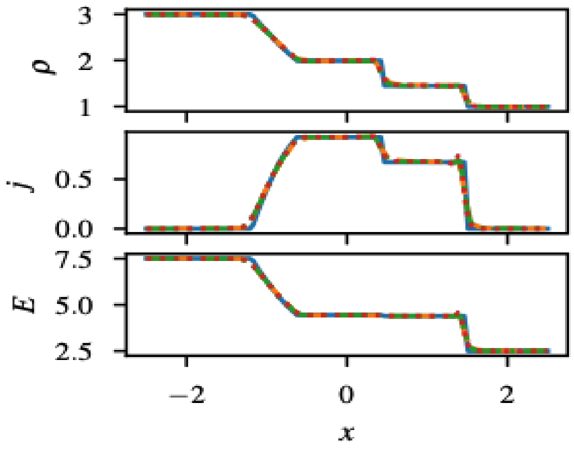


Goal: Find $s(\rho, e)$ given the solution $\rho(x,t), \rho u(x,t), E(x,t)$

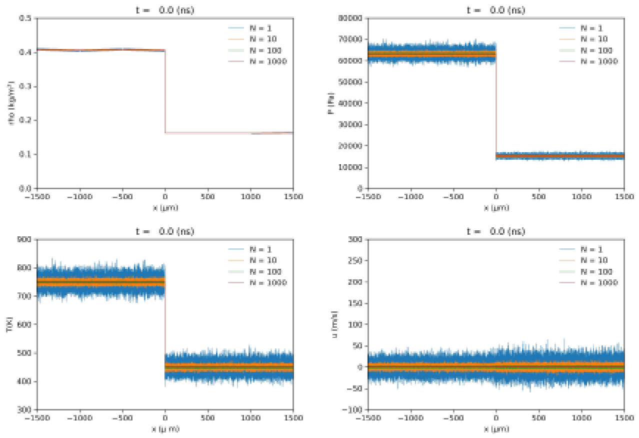
Outline

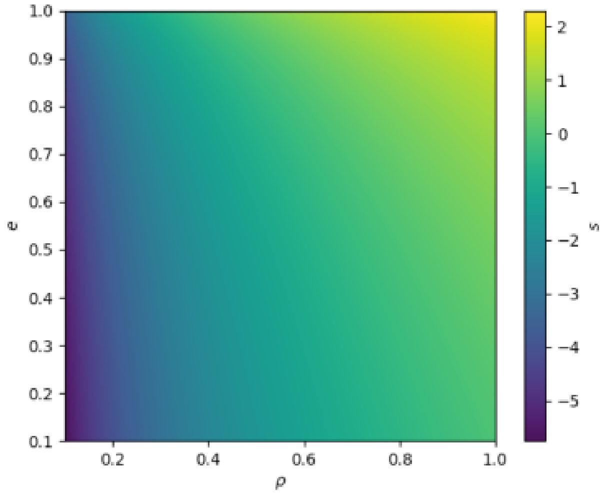
1. CVPINNs as a numerical method for hyperbolic PDEs

$$\partial_t \rho + \partial_x \rho u = 0$$
$$\partial_t \rho u + \partial_x (\rho u^2 + p) = 0$$
$$\partial_t E + \partial_x u(E + p) = 0$$



2. Equation of state discovery with CVPINNs





Physics informed neural networks[1] (PINNs) as a PDE collocation scheme

For the PDE,

$$\begin{aligned} &\partial_t u + \partial_x F(u) = 0 && (x,t) \in interior \\ &\mathcal{B}u = f && (x,t) \in boundary \\ &u = g && t = 0 \end{aligned}$$

Let the solution be defined by a neural network,

$$u = u(x, t; \xi)$$

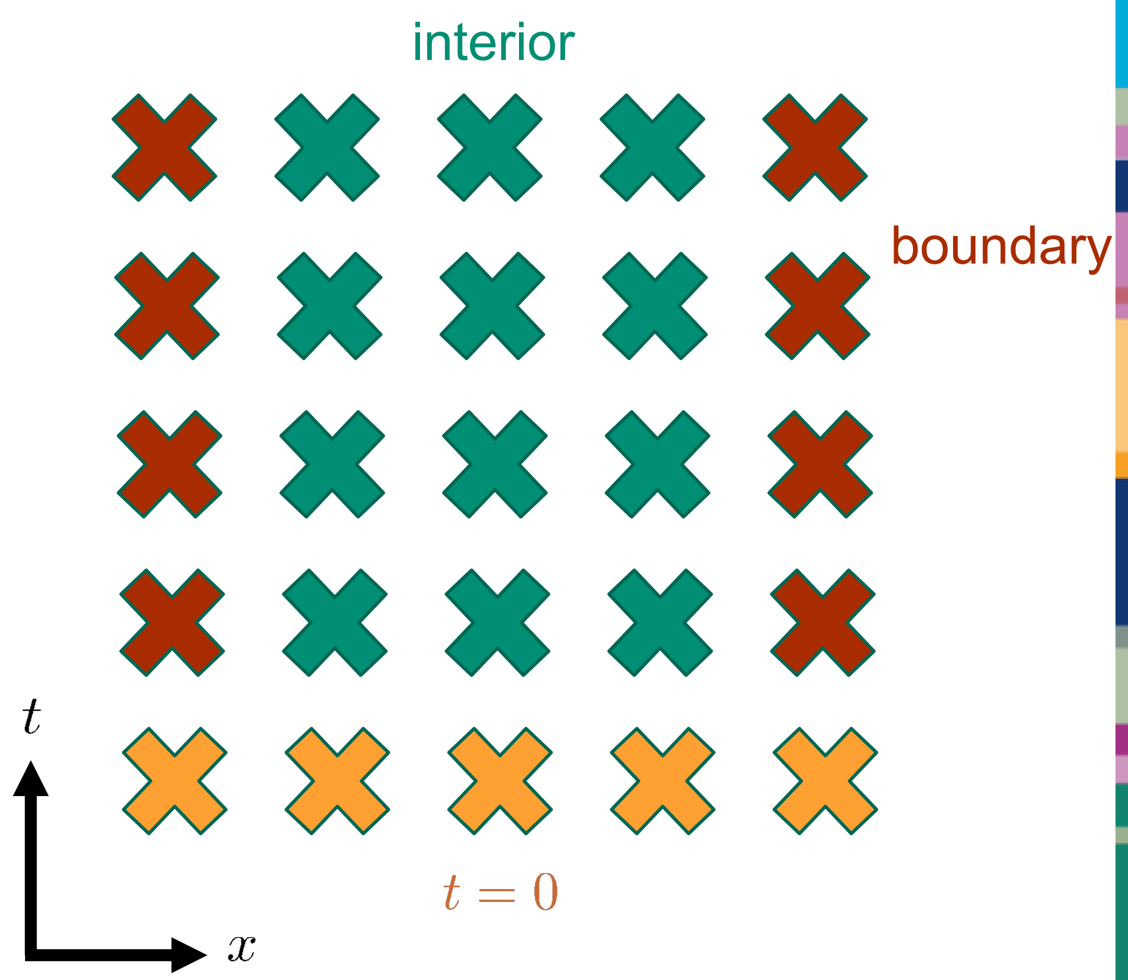
Choose collocation points in space-time

Define a residual,

$$\begin{aligned} R = &||\partial_t u + \partial_x F(u)||^2_{\ell_2(x,t)_{interior}} \\ &+ \lambda_{BC}||\mathcal{B}u - f||^2_{\ell_2(x,t)_{boundary}} \\ &+ \lambda_{IC}||u - g||^2_{\ell_2(x,0)} \end{aligned}$$

Minimize

$$\xi = \underset{\hat{\xi}}{\text{argmin}}\ R(\hat{\xi})$$



[1] Raissi et al., *arXiv:1711.10561*

Control volume PINNs (CVPINNs)

For PDEs of the form,

$$\partial_t u + \partial_x F(u) = 0 \quad (x,t) \in interior$$
$$\mathcal{B}u = f \quad (x,t) \in boundary$$
$$u = g \quad t = 0$$

Let the solution be defined by a neural network,

$$u = u(x,t;\xi)$$

Choose mesh in space-time

Apply divergence theorem to each cell in the mesh

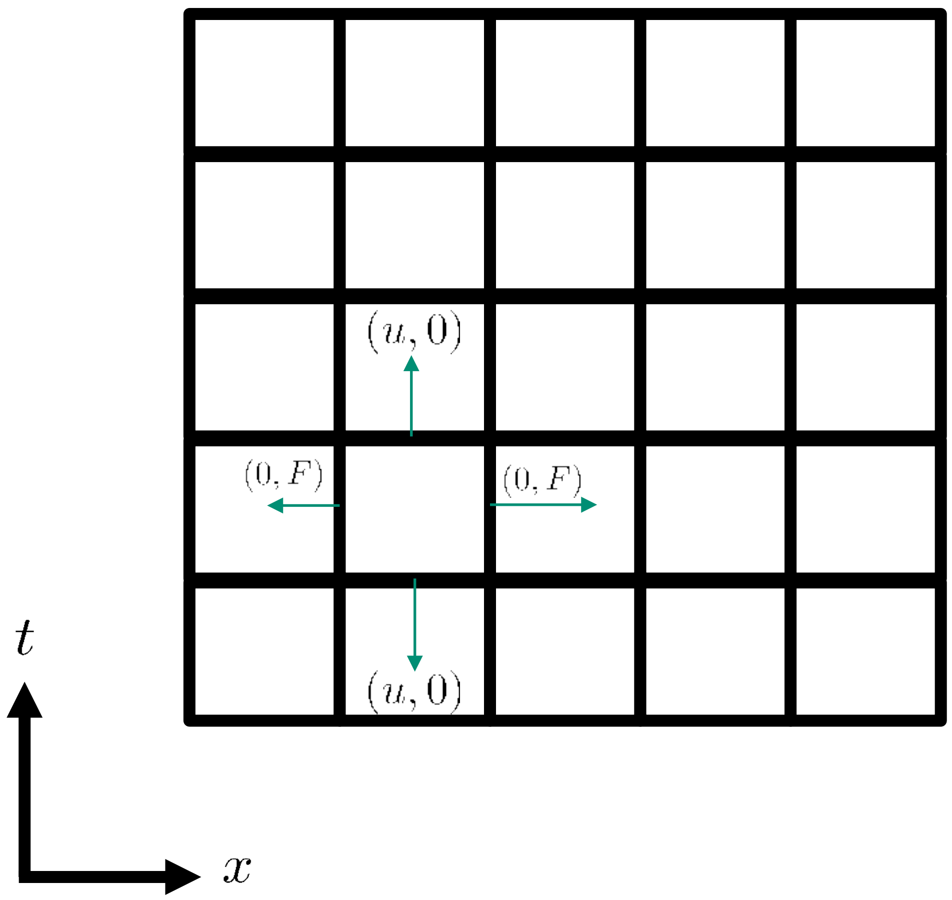
$$R_c = \int_{A_c} \nabla \cdot \begin{pmatrix} u \\ F \end{pmatrix} dA_c = \int_{l_c} \begin{pmatrix} u \\ F \end{pmatrix} \cdot dl_c$$

Approximate integrals with quadrature

Fluxes at boundaries replaced by prescribed values

Minimize residuals

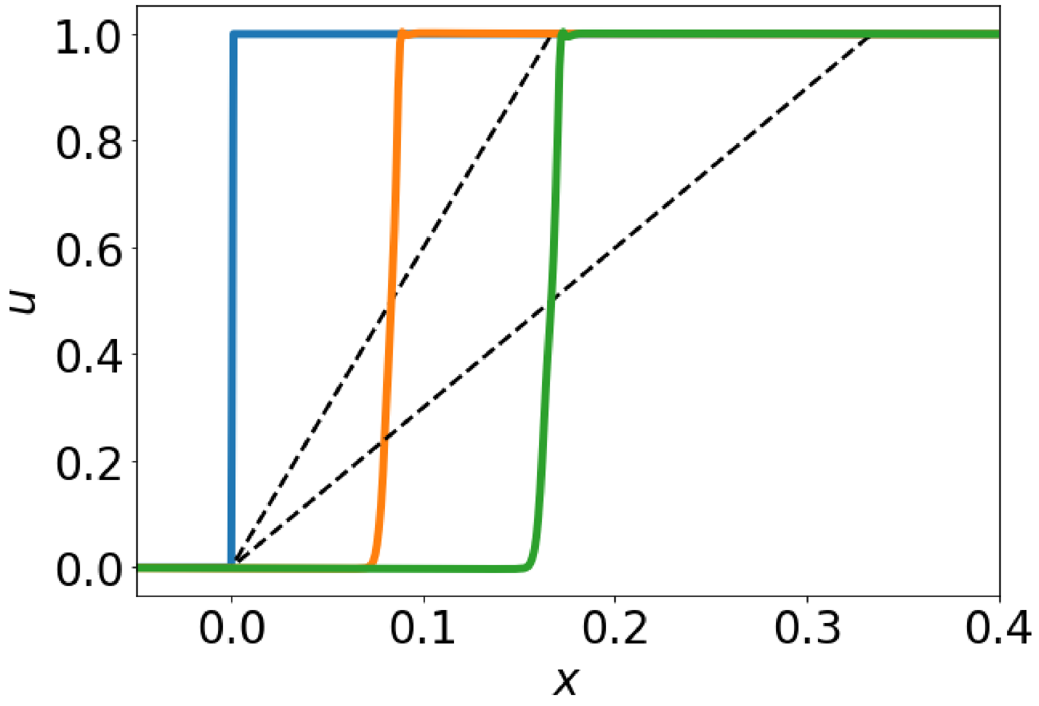
$$\xi = \underset{\hat{\xi}}{\operatorname{argmin}} \sum_c R_c^2$$



Potential issues with CVPINNs

- Oscillatory behavior near the shock
- Solutions to the integral form aren't unique
  - We want the physically meaningful entropy solution,

$$\lim_{\nu \to 0} \partial_t u + \partial_x F(u) = \nu \partial_x^2 u$$



Entropy solution to Burgers equation, $\partial_t u + \frac{1}{2}\partial_x u^2 = 0$

Unphysical solution from unregularized CVPINNs

Viscous regularization

Add von Neumann-Richtmyer viscosity to PDE[1]

$$\partial_t u + \partial_x F = \alpha \partial_x |\partial_x u| \partial_x u$$

Preferential adds viscosity near strong gradients: shocks

Prevents oscillations

Recovers viscosity solution

Potentially overly diffusive

[1] Von Neumann and Richtmyer. *Journal of Applied Physics,* 1950

Entropy regularization

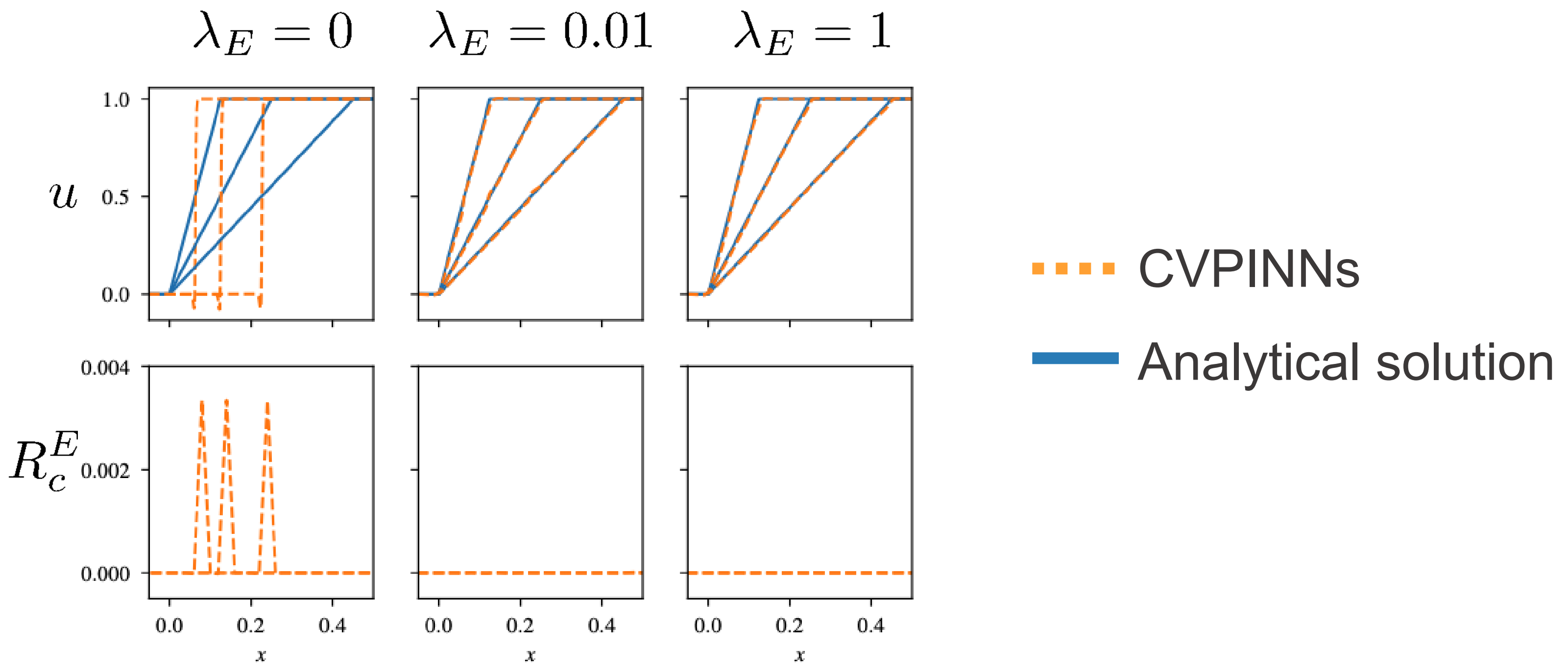
Entropy solution also obeys $\partial_t q + \partial_x \eta \leq 0$ (integral sense) for entropy pair (q, η)

Add another term to the loss,

$$\sum_c R_c^2 + \lambda_E \sum_c \max(0, R_c^E)$$

Where R_c^E is the residual to the above PDIE in integral form for cell c

For Burgers equation, with the neural network initialized to an unphysical solution,



TVD regularization

TVD schemes in standard discretizations prevent oscillations

For $u(x,t)$ at grid values $u_i^n = u(x_i, t_n)$

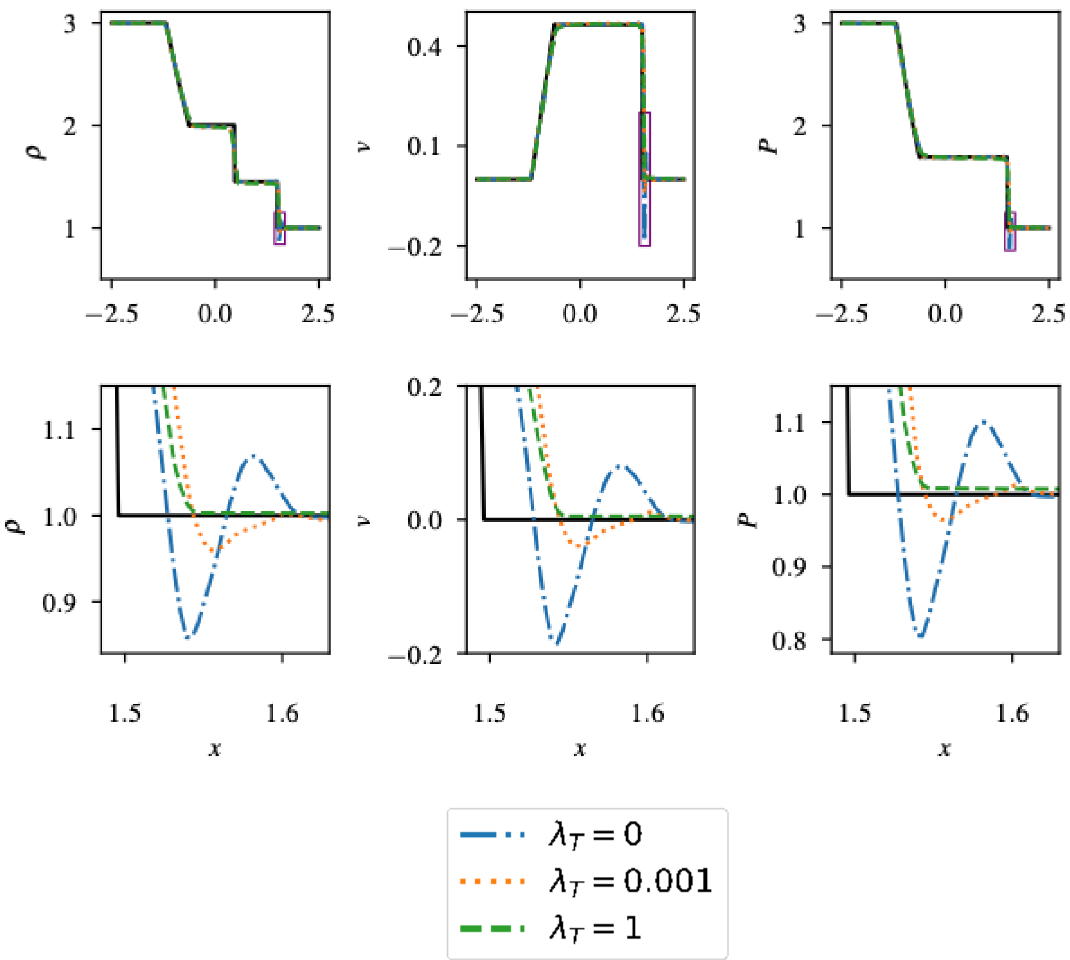
$$TV(u^n) = \sum_i |u_{i+1}^n - u_i^n|$$

$$TV(u^{n+1}) - TV(u^n) \leq 0$$

Define a regular grid on top of the mesh and add another term to the loss:

$$\sum_c R_c^2 + \lambda_E \sum_c \max(0, R_c^E)$$
$$+ \lambda_T \sum_n \max(0, TV(u^{n+1}) - TV(u^n))$$

For Euler equations with gamma law gas,



Summary of CVPINNs regularizations

Viscous

- Prevents oscillations and preferences entropy solution

Entropy

- Preferences entropy solution

TVD

- Prevents oscillations

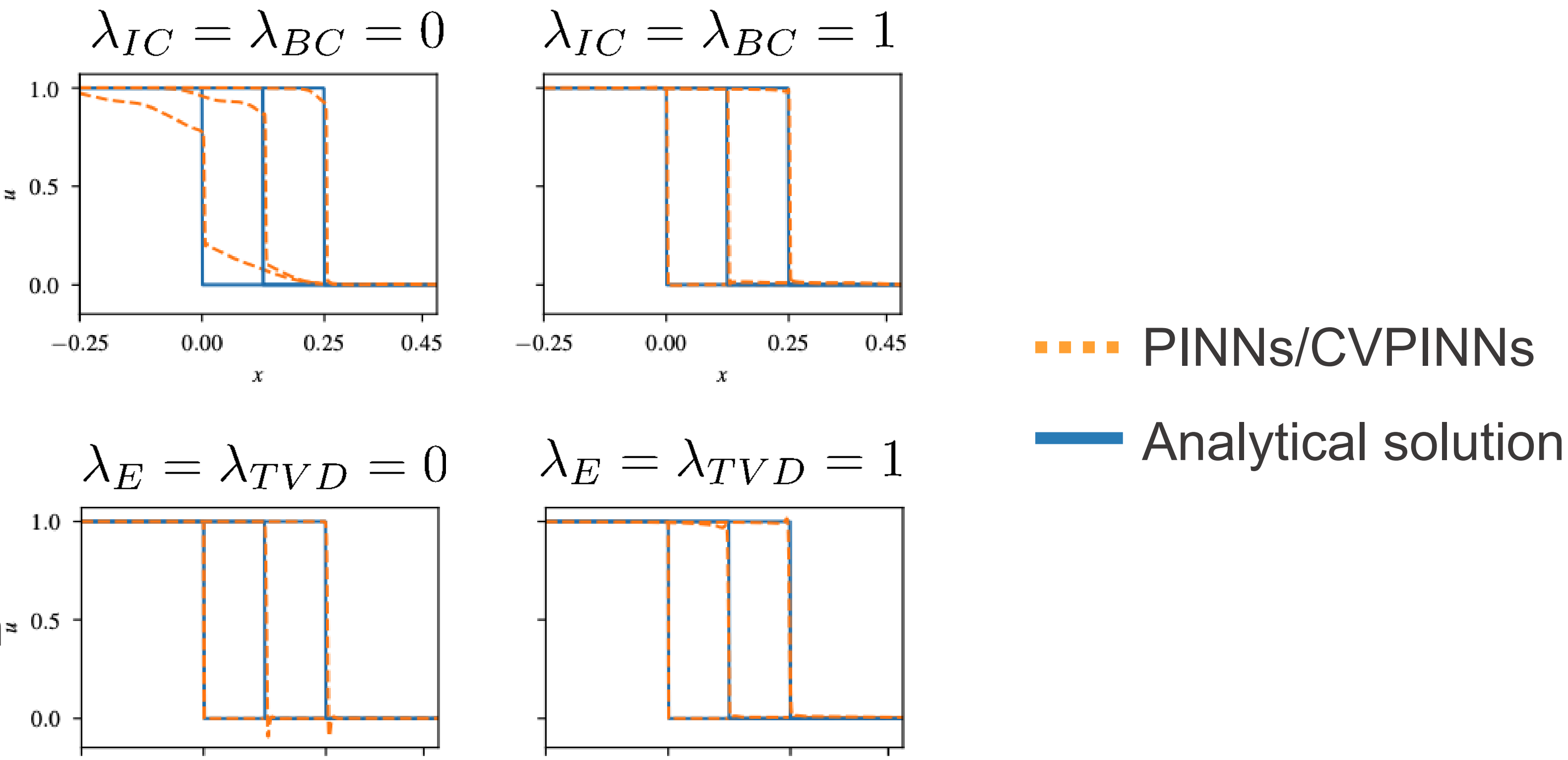
CVPINNs is less sensitive to hyperparameters than PINNs

PINNs has been used successfully for hyperbolic PDEs[1]
- We observed reduced sensitivity to hyperparameters in CVPINNs
  - BC and IC penalties in PINNs
  - Entropy+TVD penalties in CVPINNs

For Burgers equation,

PINNs solution

CVPINNs solution



[1] Mao et al. *CMAME,* 2020

PINNs can fail to find the entropy solution without artificial viscosity

Buckley-Leverett: PDE with nonconvex flux produces solutions with mixed type waves,

$$\partial_t u + \partial_x \left(\frac{u^2}{u^2 + \frac{1}{M}(1-u)^2} \right) = 0$$

Without AV

With AV

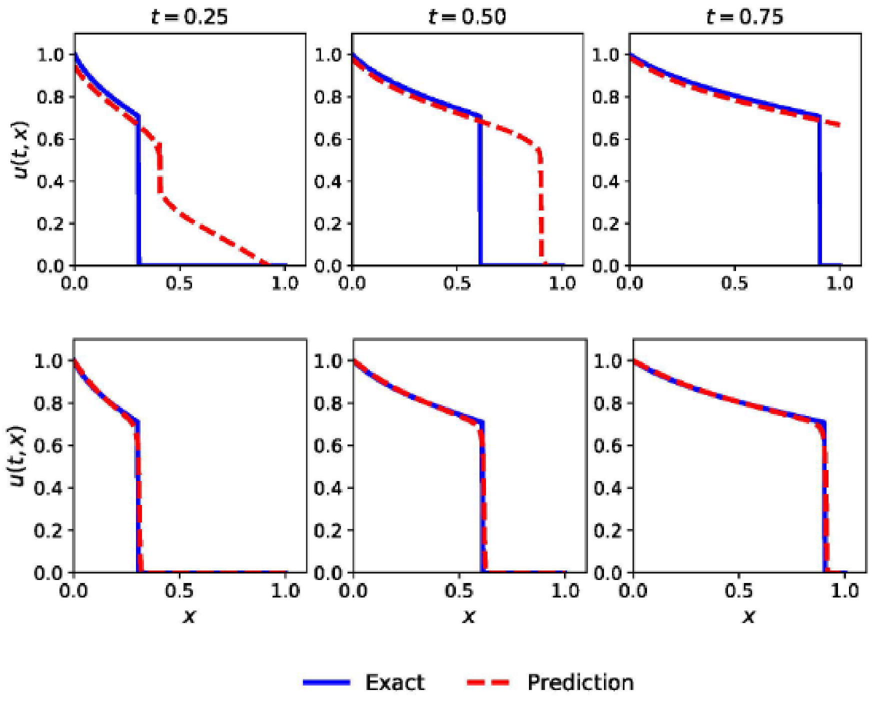
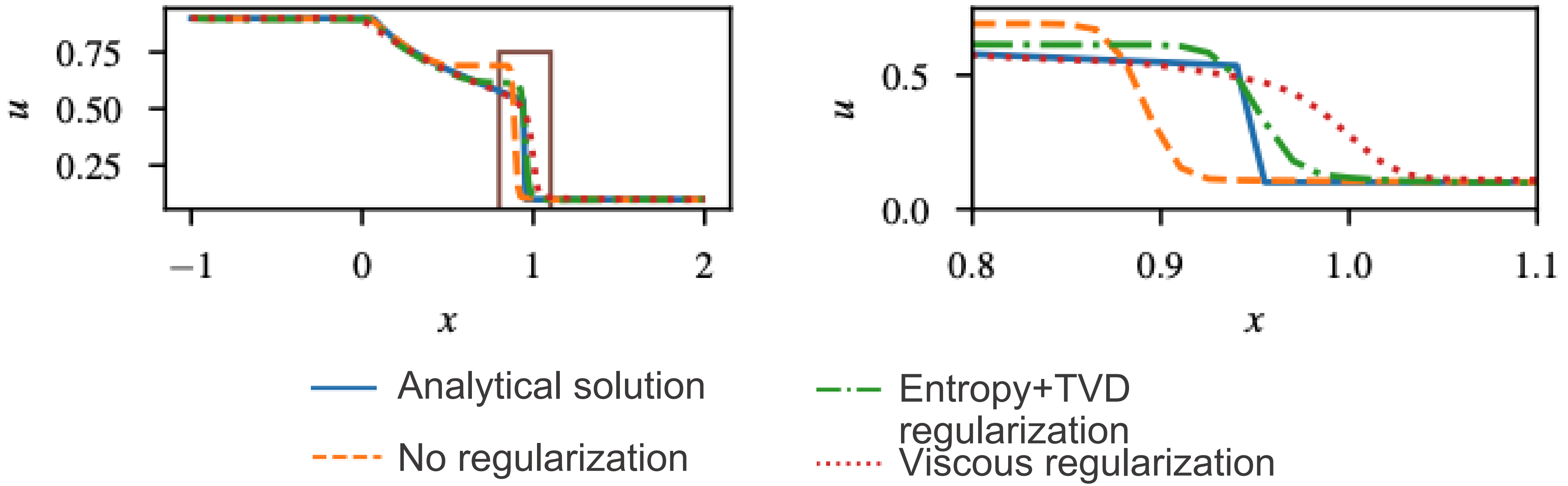


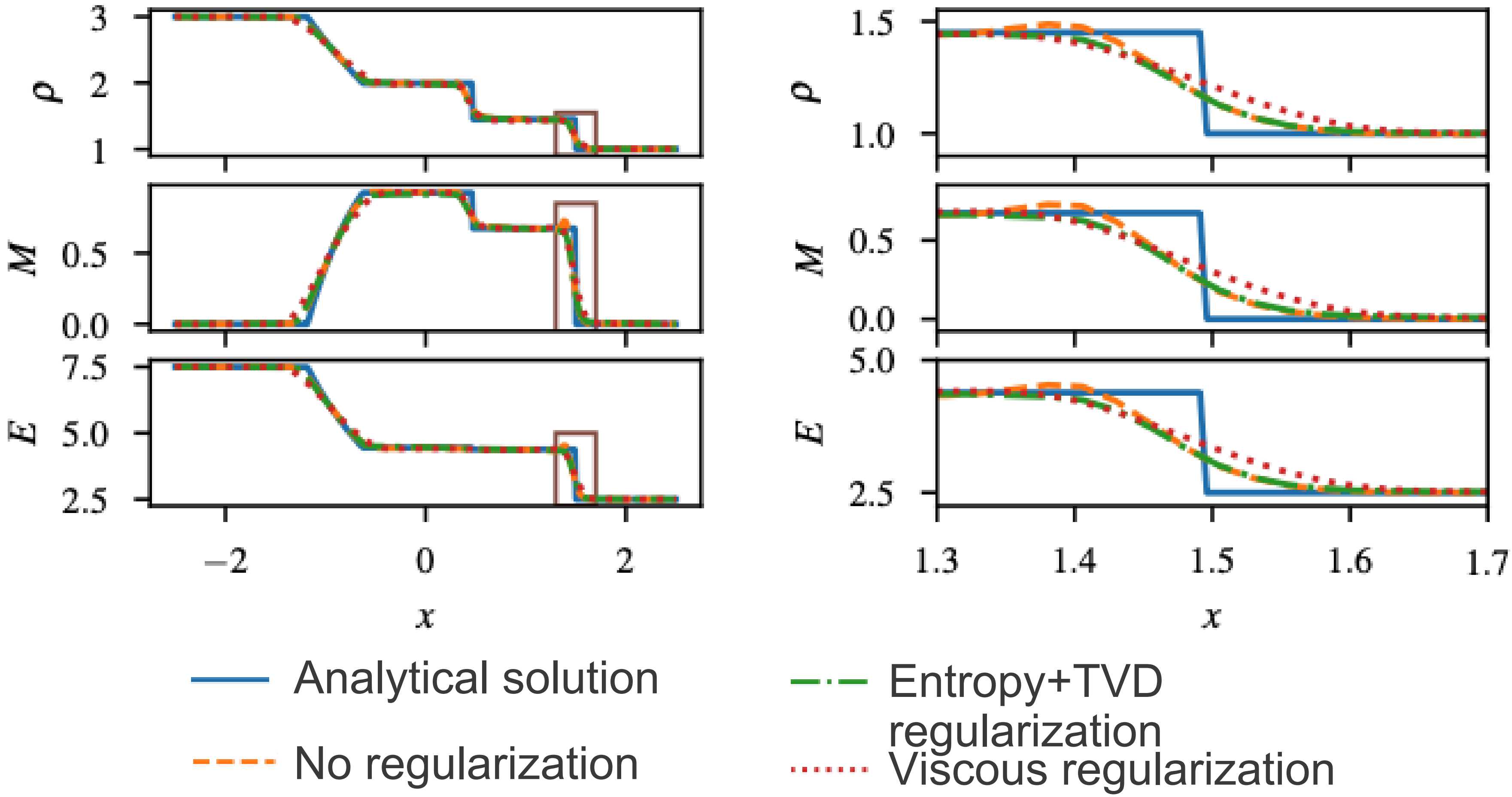
Figure reproduced from [1]

[1] Fuks and Tchelepi, *JMLMC*, 2020

CVPINNs forward solution: Buckley-Leverett equation

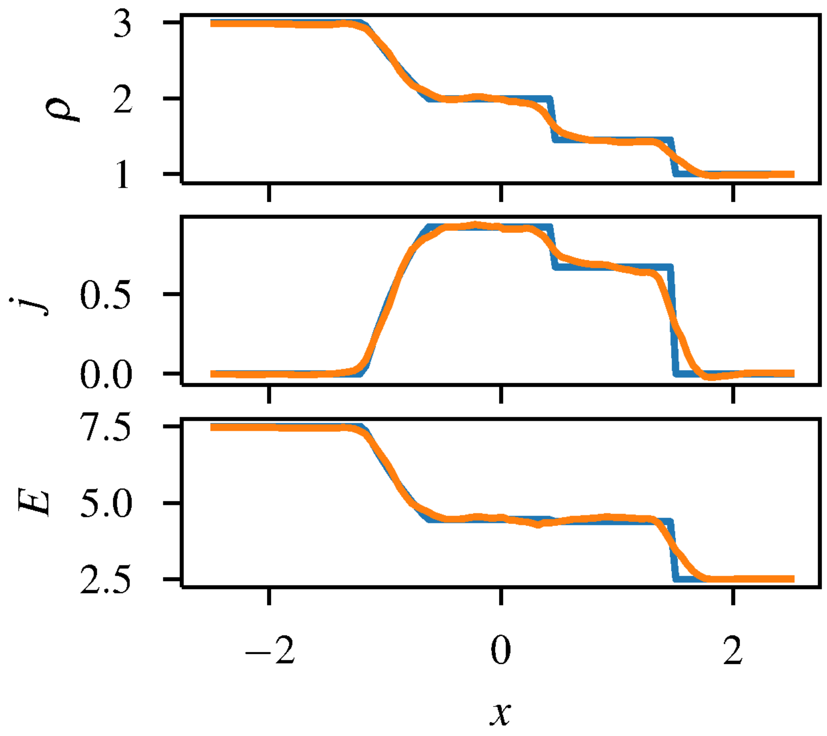


CVPINNs forward solution: Euler equations

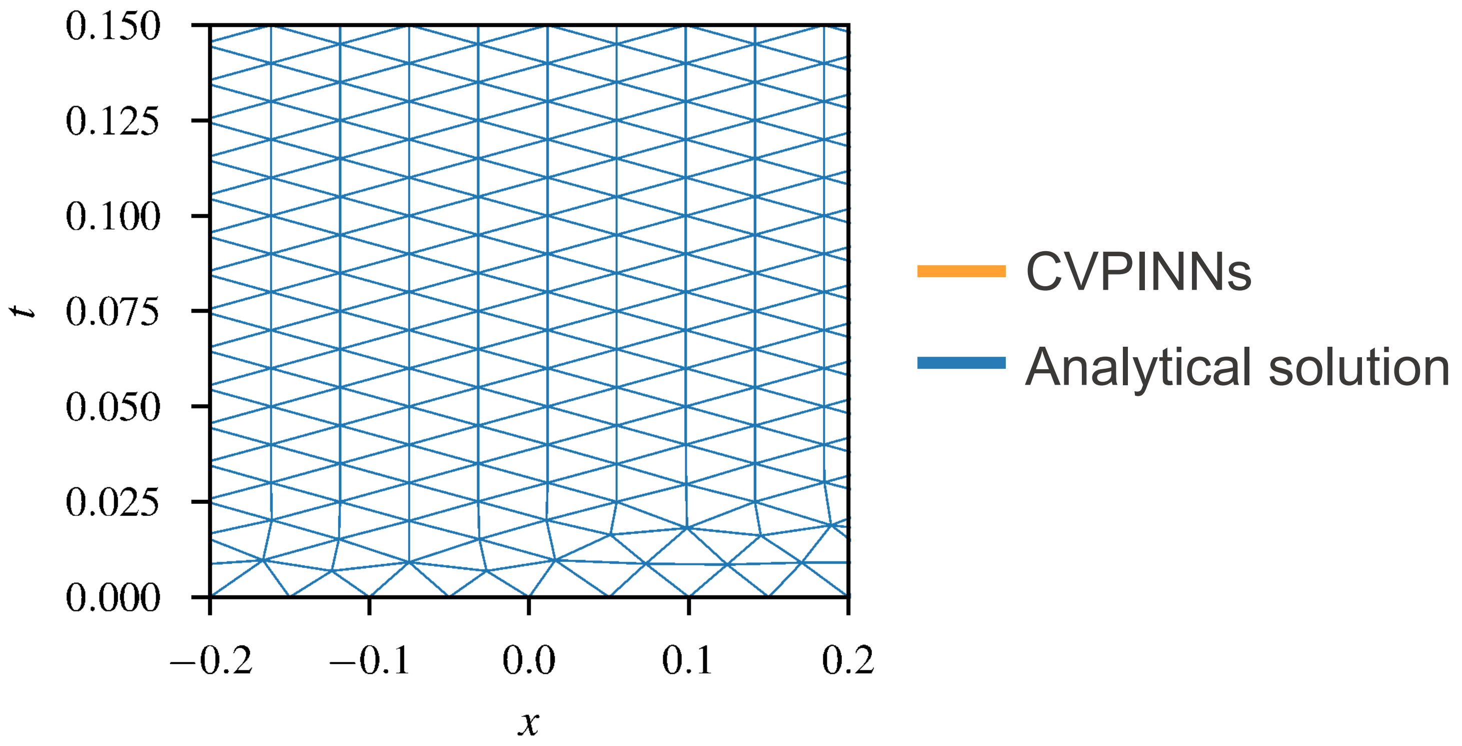


CVPINNs on unstructured meshes

For Euler equations with gamma law gas on triangular mesh[1],



Solution

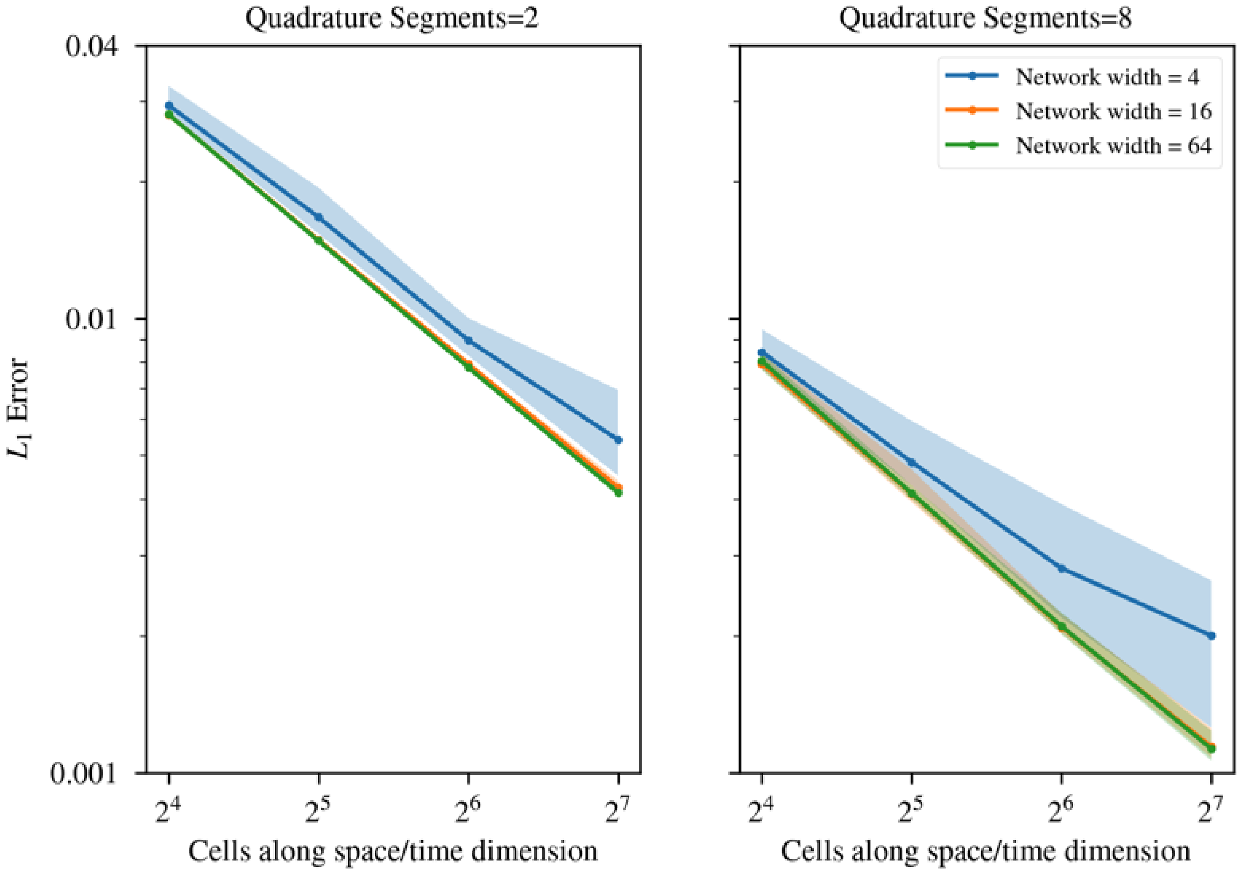


Mesh

[1] pygmsh, https://pypi.org/project/pygmsh/

CVPINNs: L1 error for Burgers rarefaction

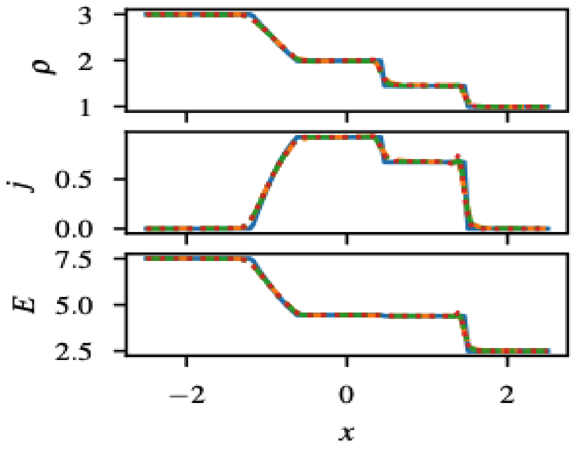
Mean and standard deviation computed over 10 runs



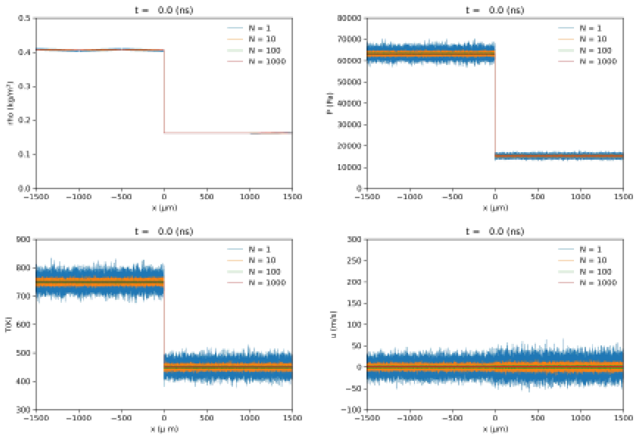
Outline

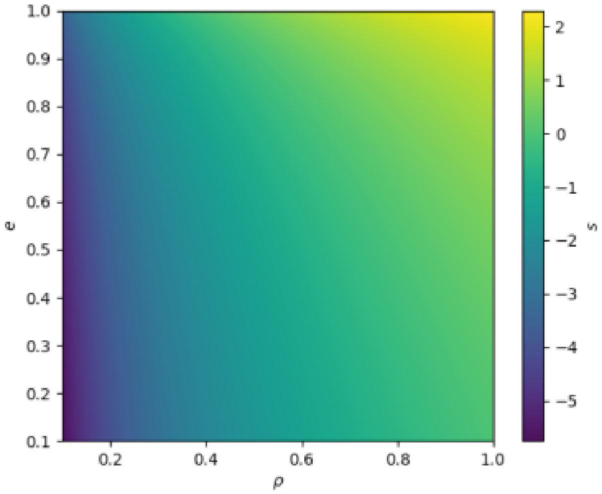
1. CVPINNs as a numerical method for hyperbolic PDEs

$$\partial_t \rho + \partial_x \rho u = 0$$
$$\partial_t \rho u + \partial_x (\rho u^2 + p) = 0$$
$$\partial_t E + \partial_x u(E + p) = 0$$



2. Equation of state discovery with CVPINNs





Equation of state discovery with CVPINNs

Find EOS given solutions to Riemann problems for Euler equations for various $(\boldsymbol{u_L}, \boldsymbol{u_R})$:

Use CVPINNs and add penalty for data:

$$\partial_t \rho + \partial_x \rho u = 0$$
$$\partial_t \rho u + \partial_x (\rho u^2 + p) = 0$$
$$\partial_t E + \partial_x u(E + p) = 0$$
$$s(\rho, e; \xi_{EOS})$$
$$p = -\rho^2 (\partial_\rho s)(\partial_e s)^{-1}$$

$$L = \sum_c R_c^2 + \lambda_E \sum_c \max(0, R_c^E)$$
$$+ \lambda_T \sum_n \max(0, TV(u^{n+1}) - TV(u^n))$$
$$+ \lambda_D ||u - u_{data}||_{\ell_2(x,t)_{data}}$$

Minimize,

$$\xi, \xi_{EOS} = \underset{\hat{\xi}, \hat{\xi}_{EOS}}{\text{argmin}}\ L$$

EOS parameterizations

Parameterized gamma law gas:

$$s(\rho, e) = \log(e^{1/(\gamma-1)}\rho^{-1})$$

Neural network:

$$s(\rho, e) = \mathcal{N}\mathcal{N}(\rho, e; \xi_{EOS})$$

EOS regularization

For a physical reasonable EOS[1],

$$\partial_e s > 0 \qquad \partial_e^2 s \leq 0 \qquad \partial_\rho(\rho^2 \partial_\rho s) < 0$$

First and third condition guarantee hyperbolicity of Euler equations

- Necessary for well-posedness of IBVP

Choose a set of $(\rho, e)_{regularize}$

Add another term to the loss,

$$L = \sum_c R_c^2 + \lambda_E \sum_c \max(0, R_c^E)$$
$$+ \lambda_T \sum_n \max(0, TV(u^{n+1}) - TV(u^n))$$
$$+ \lambda_D ||u - u_{data}||_{\ell_2(x,t)_{data}}$$
$$+ \lambda_R \sum_{(\rho,e)_{regularize}} \left[\max\left(0, -\partial_e s\right) + \max\left(0, \partial_e^2 s\right) + \max\left(0, \partial_\rho(\rho^2 \partial_\rho s)\right)\right]$$

[1] Menikoff and Plohr, *Rev. Modern Phys*, 1989

Model power tradeoff

$$s(\rho, e) = \mathcal{N}\mathcal{N}(\rho, e; \xi_{EOS})$$

$$s(\rho, e) = \log(e^{1/(\gamma-1)}\rho^{-1})$$

Black-box ML

Regularized ML

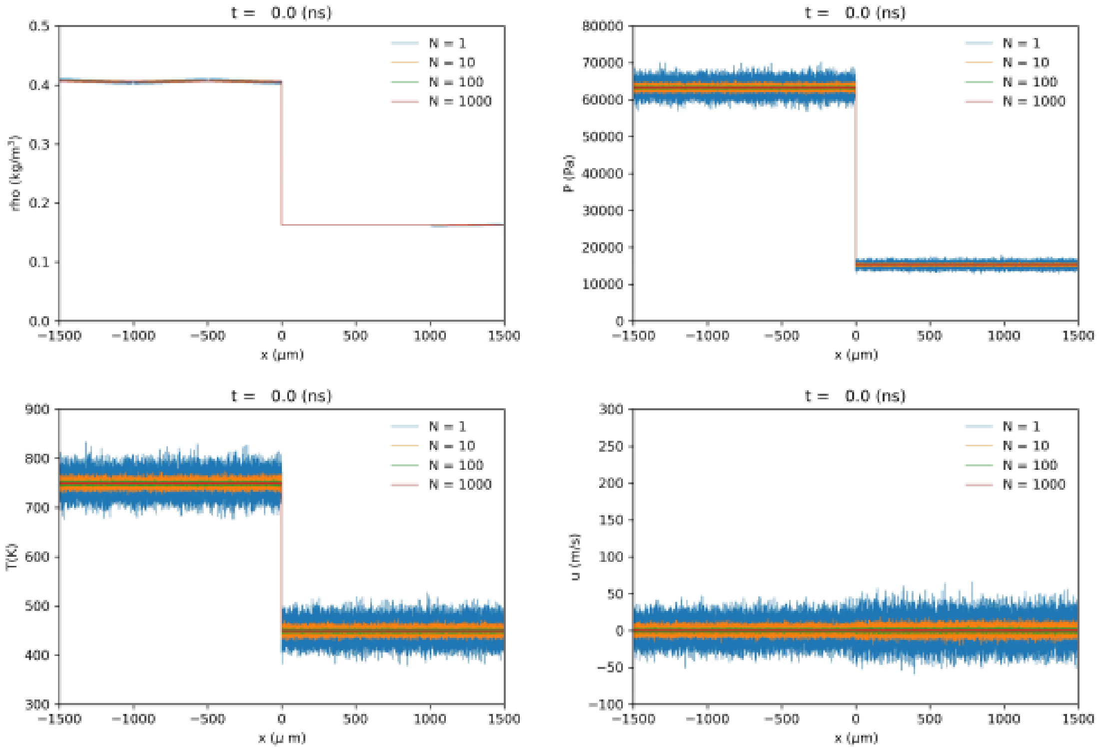
Parameter estimation

Prone to overfitting ⟷ Strong assumptions

Verification with DSMC data

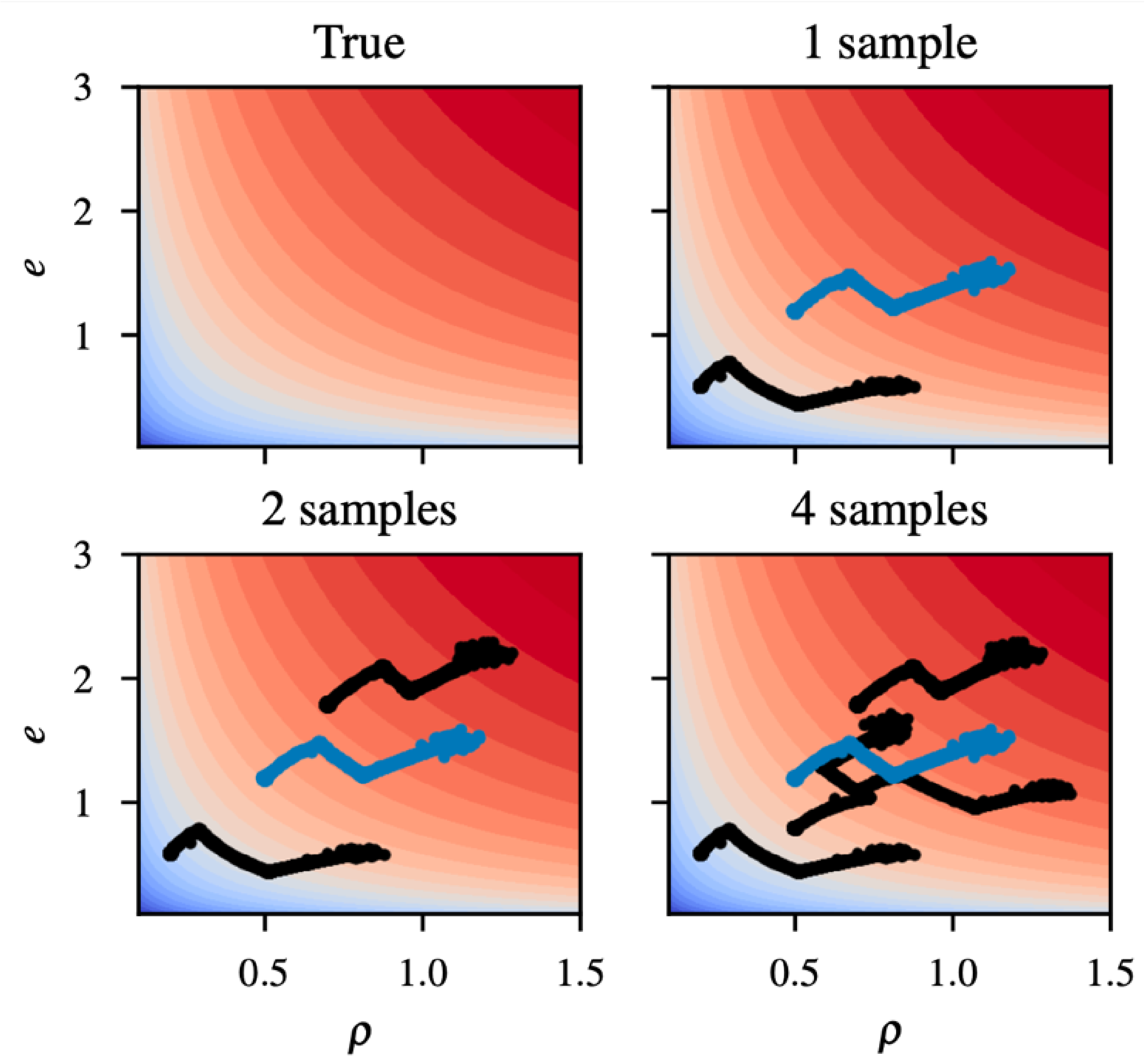
DSMC simulations of Sod shock problems performed in Sparta[1]

- Varying density, pressure jump at t=0
- Hydrodynamic regime
- Argon gas
- Euler with gamma law gas EOS is a good model
- $s(\rho, e) = \log(e^{1/(\gamma-1)}\rho^{-1}) \qquad \gamma = 5/3$
- Can compare fitted EOS with above EOS



[1] Sparta, https://sparta.sandia.gov/

EOS fits: parameterized gamma law gas



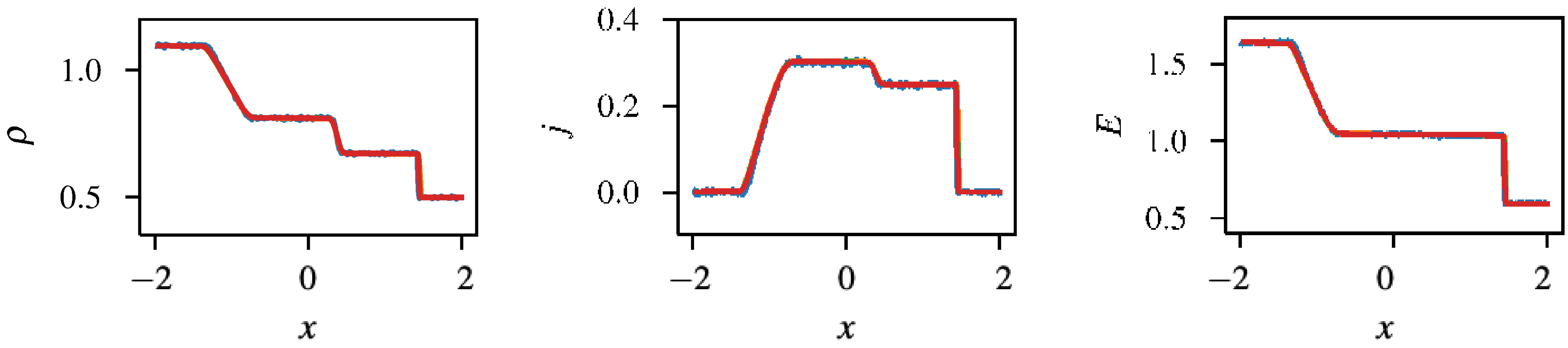
- Training data
- Elliptic regions for $u = 0$
- Test data

Samples	Error in γ
1	1.04%
2	0.74%
4	0.62%

EOS fits test: Parameterized gamma law gas

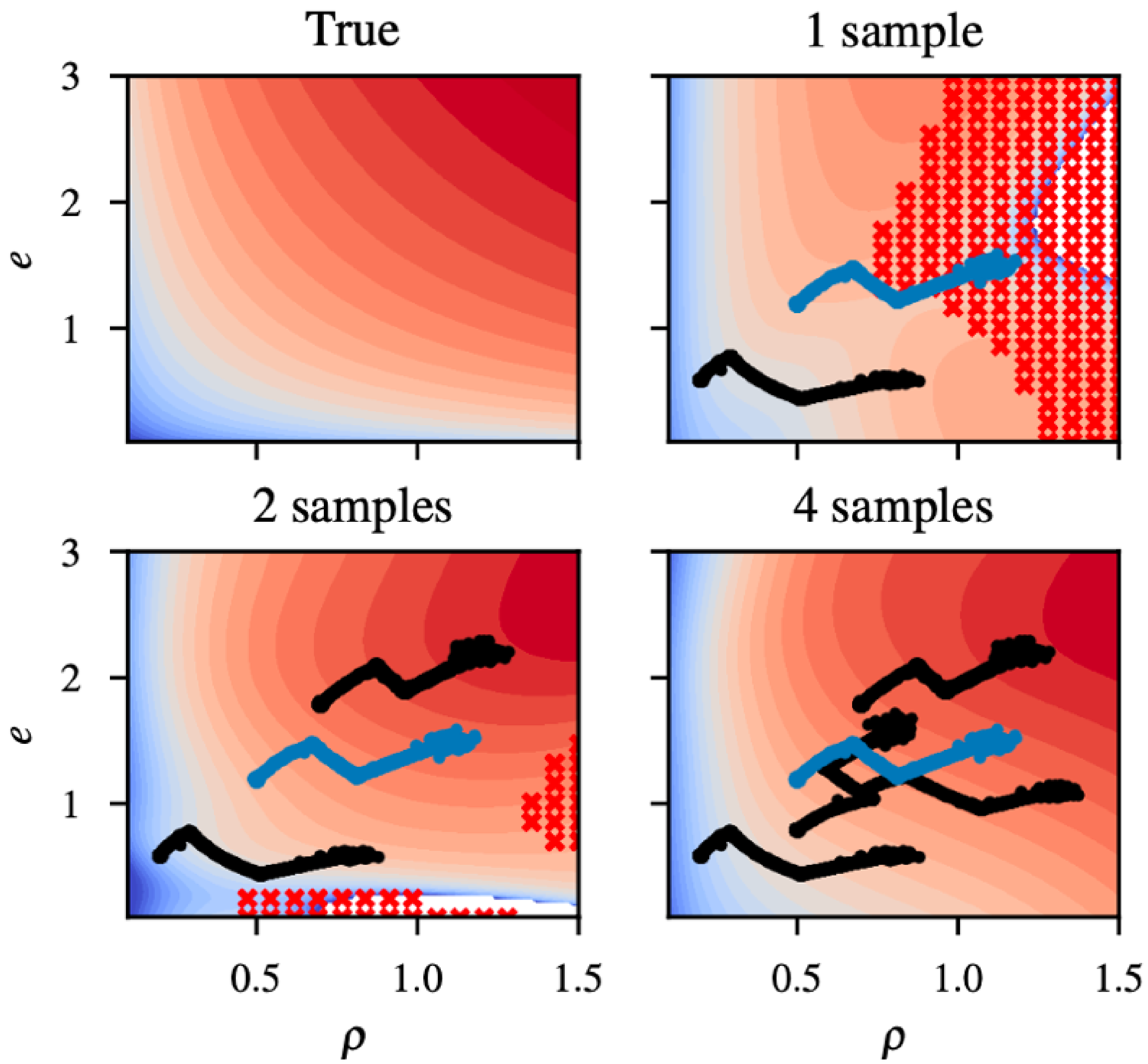
Use fitted EOS's to solve new Riemann problem

- Viscous regularized finite difference (FD) code on a fine mesh



DSMC 1 Sample fit 2 Sample fit 4 Sample fit

EOS fits: Neural network

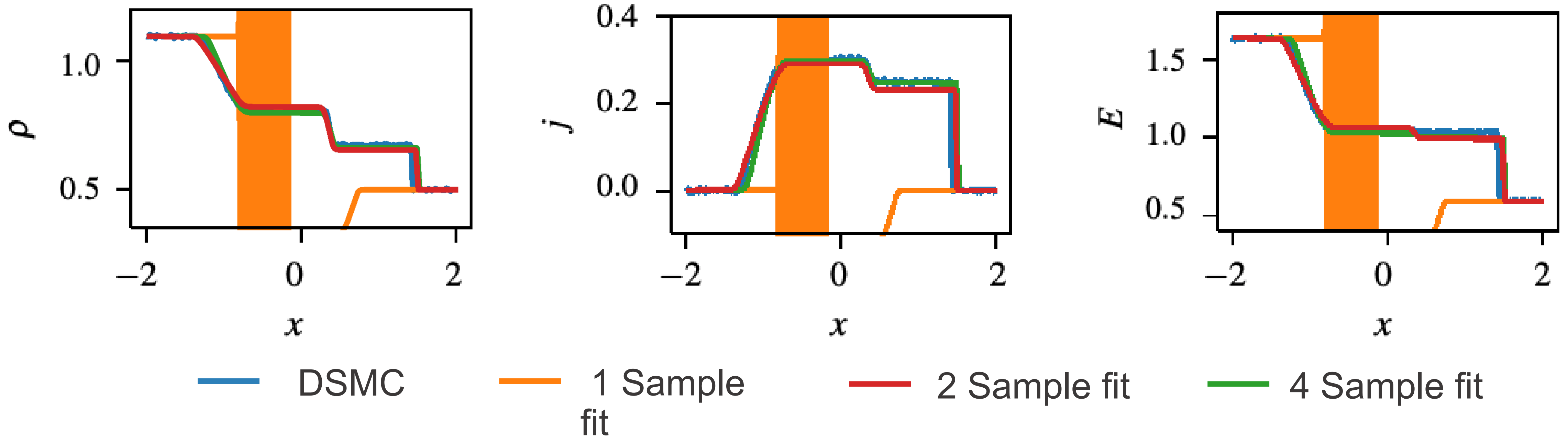


- Training data
- Elliptic regions for $u = 0$
- Test data

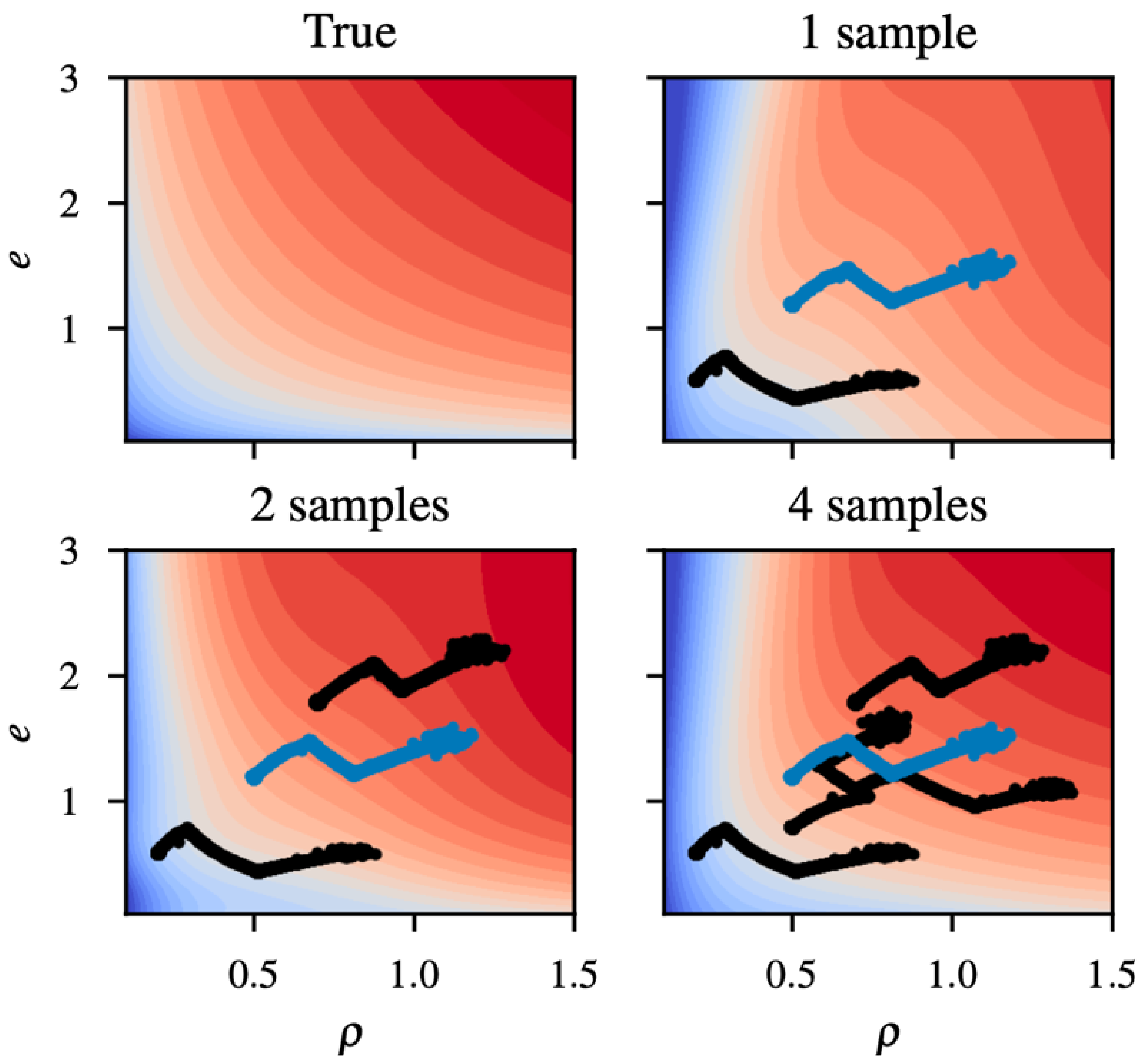
EOS fits test: Neural network

Use fitted EOS's to solve new Riemann problem

- Viscous regularized finite difference (FD) code on a fine mesh



EOS fits: Regularized neural network

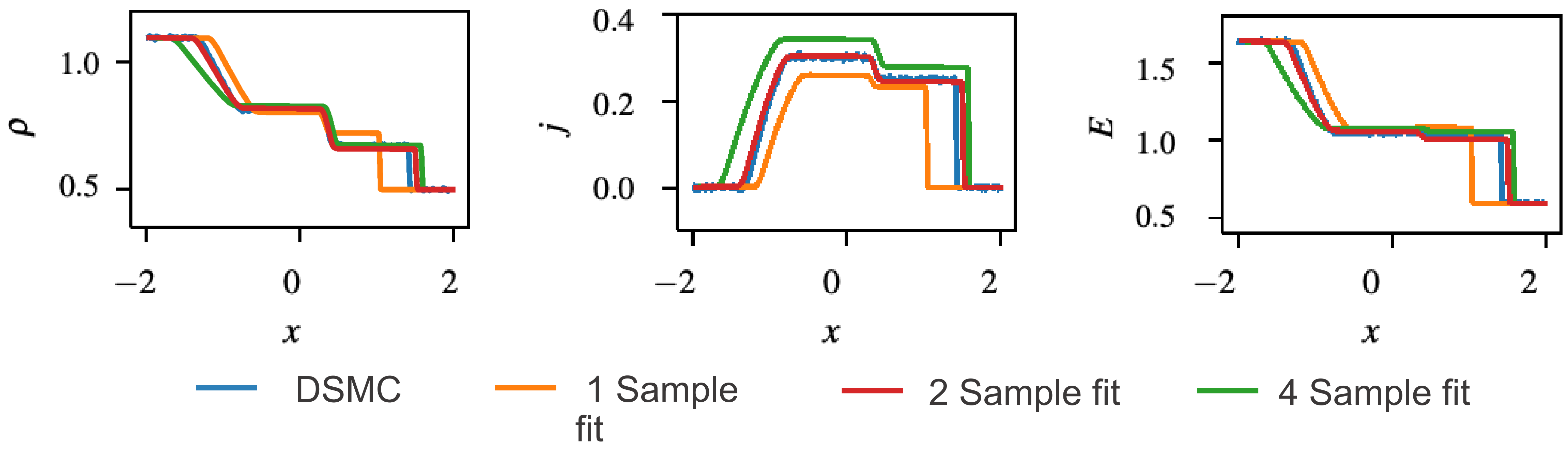


- Training data
- Elliptic regions for $u = 0$
- Test data

EOS fits test: Regularized neural network

Use fitted EOS's to solve new Riemann problem

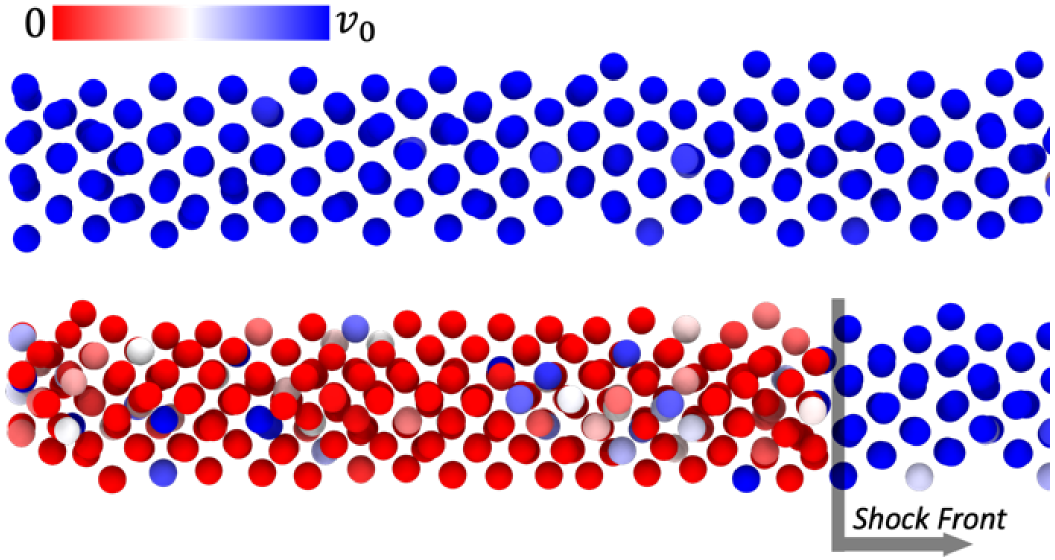
- Viscous regularized finite difference (FD) code on a fine mesh



EOS discovery application: shock hydrodynamics of copper

Perform LAMMPS[1] simulations of the reverse-ballistic impact experiment

- Various impact velocities and initial temperatures



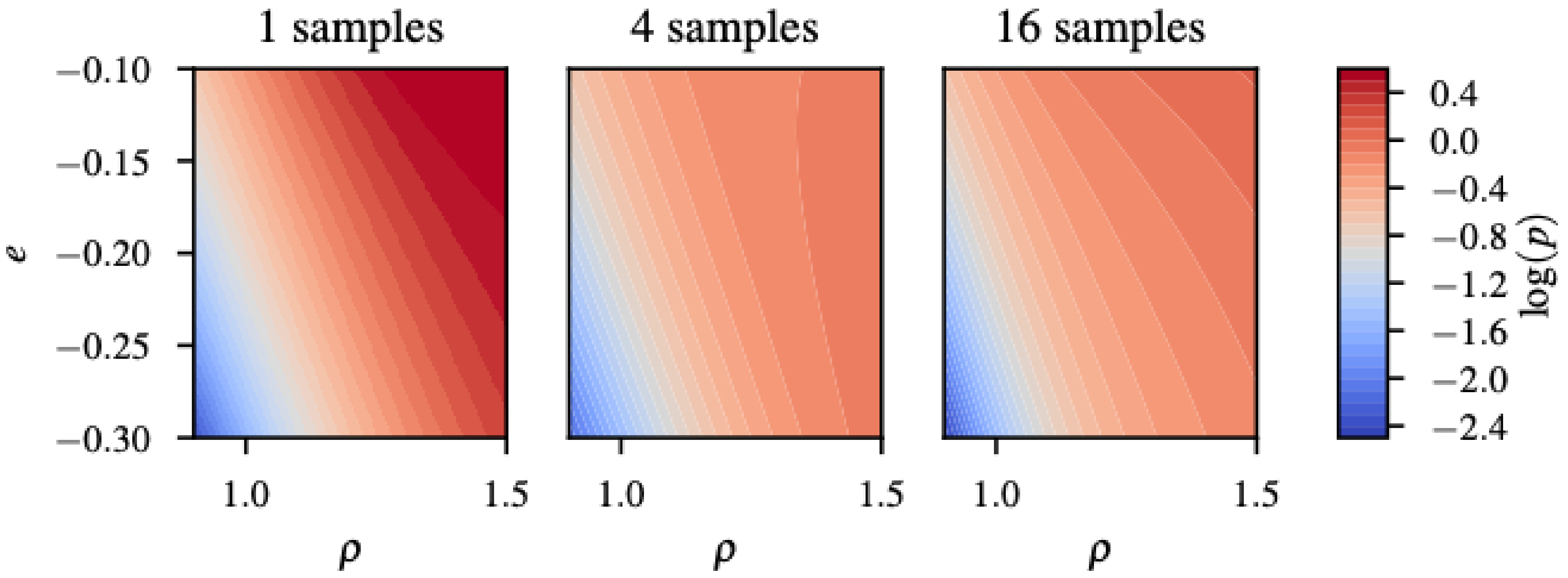
Fit an EOS to the LAMMPS data using CVPINNs

- Regularized neural network parameterization

Use fitted EOS to perform FD simulations of a new impact case and compare to LAMMPS

[1]LAMMPS, https://lammps.sandia.gov

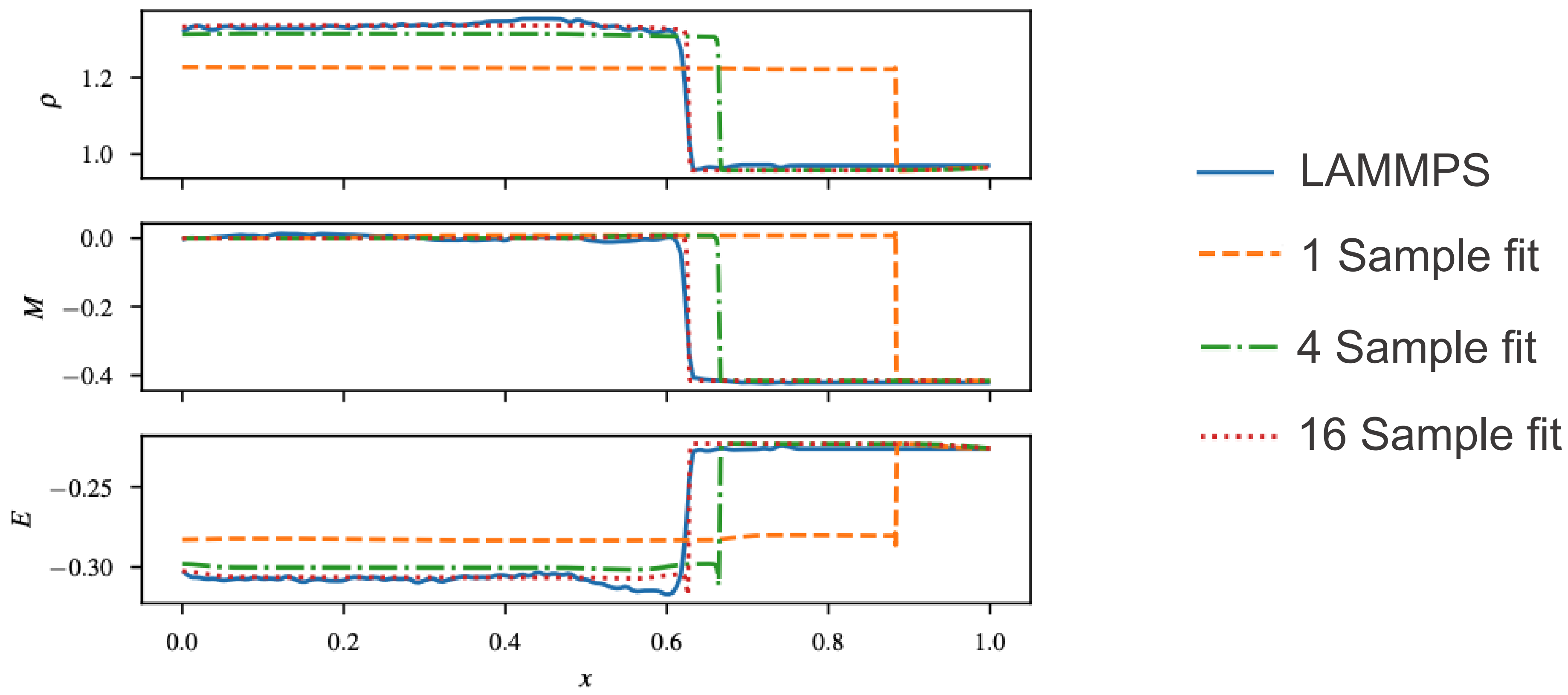
EOS fits for shocked copper



EOS fits test for shocked copper

Use fitted EOS's to solve new impact case

- Viscous regularized finite difference (FD) code on a fine mesh



Future work

Compare shocked cooper EOS to EOS's in literature

- Mie–Grüneisen[1]

Compare to traditional discretizations of hyperbolic PDE's and EOS parameterizations

Extend to multidimensional problems

Extend to other PDEs

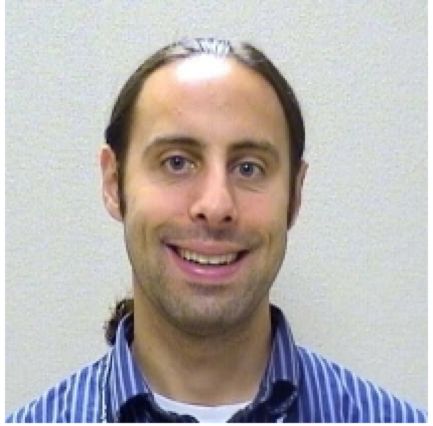
- MHD: Brio-Wu problem[2]
- Reactive flows: Ben-artzi problem[3]

[1]Robinson, *SAND2019-6025*, 2019

[2]Brio and Wu, *JCP*, 1988

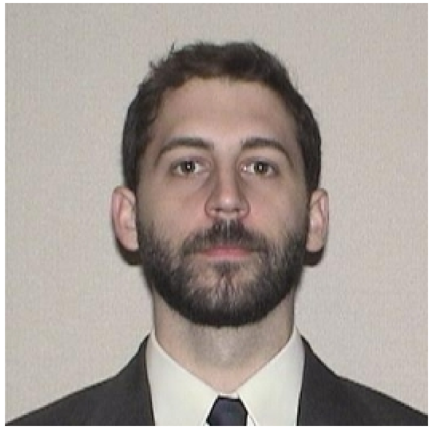
[3]Ben-artzi, *JCP*, 1989

Acknowledgements



Eric C. Cyr

Nat Trask



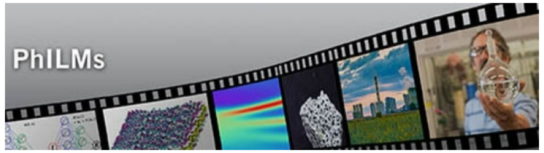
Mitch Wood

U.S. DEPARTMENT OF ENERGY | Office of Science

Ignacio Tomas

Indu Manickam

MK Lee



www.pnnl.gov/computing/philms

Conclusion

CVPINNs provides some benefits over PINNs for solving hyperbolic PDEs
- Less regularization needed
- For some problems, PINNs produces poor answer (Buckley-Leverett) without AV

CVPINNs can be used to solve inverse problems: EOS discovery
- Verified with DSMC simulations of the Sod problem
- Applied to extract an EOS for shocked copper
- Choice of EOS parameterization should be carefully considered
  - Parameterized gamma law gas
  - Neural network
  - Regularized neural network

Preprint:
- Patel et al. *arXiv:2012.05343*