Mixture-of-Experts Physics-Informed Neural Operators for Multi-Branch Bratu Problems

Tianle Xu
MPNO Project

***Abstract*—Nonlinear boundary-value problems can map one parameter value to several steady states, making a single-valued neural operator a structurally incomplete surrogate. We introduce MPNO, a mixture-of-experts physics-informed neural operator that represents Bratu solution sets with multiple Fourier-operator heads trained by residual losses. On Bratu-1D, MPNO recovers distinct lower and upper branches on 31 of 32 sampled parameter values; both branches are matched with relative $L^2 < 0.2$ on 26 of 32 values. Head B tracks the upper branch with median best relative $L^2 = 3.20 \times 10^{-3}$, while Head A tracks the lower branch with median best relative $L^2 = 6.46 \times 10^{-2}$. A Bratu-2D extension obtains mean residuals 0.101 and 0.234 for the two heads. These results show that a residual-trained operator with expert heads can learn a practical set-valued surrogate for multi-branch elliptic problems.**

***Index Terms*—neural operators, Fourier neural operator, physics-informed learning, Bratu equation, multiple solutions, mixture of experts**

I. Introduction

Parametric partial differential equations are usually cast as solution operators: coefficients, boundary data, or scalar controls are mapped to a field. DeepONet and Fourier neural operators made this viewpoint practical for families of PDE instances by learning maps between function spaces [1], [2], [3], [4]. Physics-informed operator learning further reduces dependence on labeled fields by optimizing residual, weak, or variational objectives [5], [6].

This single-valued view is incomplete for nonlinear steady problems with multiple solutions. The Bratu equation is a classical example. In one dimension, two positive branches coexist below a fold; in higher dimensions, continuation and finite-difference studies reveal similarly delicate bifurcation structure [7], [8]. Classical numerical methods address this through continuation, pde2path, and deflation [9], [10], [11]. Neural approaches to multiple solutions often rely on repeated initialization, ensembles, or separately trained heads [12], [13], [14].

We study a different formulation: a single conditional neural operator that returns a finite set of candidate branches for one parameter. The resulting model, MPNO, shares a Fourier neural-operator trunk across experts and uses expert-specific decoders to represent different Bratu branches. Residual losses train all experts without branch labels, while a Dirichlet-safe output mask enforces boundary conditions by construction.

The contributions are: First, we formulate Bratu branch learning as set-valued physics-informed operator learning. Second, we introduce a mixture-of-experts Fourier operator with a branch-separating kick map and residual-balanced expert objective. Third, we derive the Bratu-1D analytic reference and tangent-consistency regularizer used to stabilize training near the fold. Fourth, we provide a complete artifact-backed evaluation on Bratu-1D and Bratu-2D, including branch coverage, relative L^2 error, residual curves, and field visualizations.

II. Related Work

A. Neural Operators

DeepONet approximates nonlinear operators through branch and trunk networks under an operator universal approximation theorem [1]. The Fourier neural operator parameterizes integral kernels in spectral space and became a standard model for parametric PDEs [2]. Recent surveys and libraries clarify the theory, implementation, and engineering practice of neural operators [3], [4]. Decomposed and geometry-aware variants extend the basic operator-learning pipeline to larger and more complex simulations [15], [16].

B. Physics-Informed Learning

Physics-informed neural networks optimize differential residuals directly [17], but their training dynamics can fail in stiff or multi-scale settings [18]. Sequential training and variational operator losses improve parts of this landscape [19], [6]. MPNO follows the physics-informed operator-learning direction but changes the output object from a single field to a finite solution set.

C. Multiple Solutions and Branch Discovery

Continuation and bifurcation packages remain the most reliable way to trace nonlinear solution manifolds [9], [10]. Deflation modifies the nonlinear residual after one solution is found, allowing Newton-type solvers to search for distinct branches [11]. Neural methods for multi-solution discovery include bifurcation-analysis networks, ensembles, and multi-head PINNs [12], [13], [14]. MPNO differs in using one conditional neural operator with simultaneous expert outputs.

D. Mixture of Experts

Mixture-of-experts models originated as adaptive local predictors [20] and later became a scalable routing mechanism for deep networks [21], [22], [23]. In MPNO, experts are not routed token processors. They are candidate PDE solution branches sharing one spectral representation.

III. Problem Formulation

Let $\Omega \subset \mathbb{R}^d$ and $\lambda \in \Lambda \subset \mathbb{R}$. We consider nonlinear boundary-value problems

$$\mathcal{F}(u;\lambda) = 0 \quad \text{in } \Omega, \qquad \mathcal{B}(u) = 0 \quad \text{on } \partial\Omega. \tag{1}$$

The solution operator is set-valued:

$$\mathcal{S}(\lambda) = \left\{ u^{(j)}(\lambda) : \mathcal{F}(u^{(j)};\lambda) = 0,\ \mathcal{B}(u^{(j)}) = 0 \right\}. \tag{2}$$

For the Bratu equation,

$$\mathcal{F}(u;\lambda) = \Delta u + \lambda e^u, \qquad u|_{\partial\Omega} = 0. \tag{3}$$

In one dimension, the analytic family is parameterized by $a > 0$:

$$u_a(x) = -2\log\cosh(a(x - \tfrac{1}{2})) + 2\log\cosh(a/2), \tag{4}$$

$$\lambda(a) = 2a^2 \operatorname{sech}^2(a/2). \tag{5}$$

The critical value $\lambda_* = \max_{a>0} \lambda(a) \approx 3.5138$ separates the two-positive-solution regime from the no-positive-solution regime.

IV. MPNO Method

MPNO approximates the set-valued map by

$$\mathcal{G}_\theta(\lambda) = \{u_{\theta,k}(\cdot;\lambda)\}_{k=1}^{K}. \tag{6}$$

For grid coordinate x, the initial hidden field is

$$v_0(x) = P([\lambda, x]). \tag{7}$$

A Fourier layer updates the field by

$$v_{\ell+1}(x) = \sigma\left(W_\ell v_\ell(x) + \mathcal{F}^{-1}\left(R_\ell \cdot \mathcal{F} v_\ell\right)(x)\right), \tag{8}$$

where R_ℓ contains the retained low-frequency complex weights. Each expert head maps the shared representation to a Dirichlet-safe solution:

$$u_k(x;\lambda) = m(x) Q_k(v_L(x)). \tag{9}$$

For $\Omega = (0,1)$, $m(x) = x(1-x)$. For $\Omega = (0,1)^2$,

$$m(x,y) = x(1-x)y(1-y). \tag{10}$$

One expert can receive an additive branch-separating kick:

$$u_B(x;\lambda) = m(x) Q_B(v_L(x)) + m(x) s_\psi a_\psi(\lambda, x), \tag{11}$$

where $s_\psi > 0$ is softplus-parameterized and a_ψ is a small multilayer perceptron. The mask keeps both the ordinary head and the kick compatible with homogeneous Dirichlet data.

The per-head physics loss is

$$\mathcal{L}_k(\lambda) = \int_\Omega |\Delta u_{\theta,k}(x;\lambda) + \lambda e^{u_{\theta,k}(x;\lambda)}|^2\, dx. \tag{12}$$

For two heads, we use

$$\mathcal{L}_{\mathrm{MPNO}} = \mathbb{E}_\lambda \left[\max(\mathcal{L}_A, \mathcal{L}_B) + \eta \min(\mathcal{L}_A, \mathcal{L}_B)\right]. \tag{13}$$

The maximum term prevents one expert from being ignored, while the minimum term allows a low-residual branch to keep improving.

For Bratu-1D, differentiating $\mathcal{F}(u;\lambda) = 0$ gives a tangent equation

$$J_u w + \partial_\lambda \mathcal{F} = 0, \qquad w = \partial_\lambda u, \tag{14}$$

with

$$J_u w = w'' + \lambda e^u w, \qquad \partial_\lambda \mathcal{F} = e^u. \tag{15}$$

We approximate

$$w \approx \frac{u(\lambda+\delta) - u(\lambda-\delta)}{2\delta} \tag{16}$$

and add

$$\mathcal{L}_{\tan} = \int_0^1 |w'' + \lambda e^u w + e^u|^2\, dx. \tag{17}$$

V. Experiments

A. Benchmarks

The main benchmark is Bratu-1D on $(0,1)$, evaluated on 32 parameter values in $[0.1, 3.5]$, below the analytic fold λ_*. Each predicted head is compared with both analytic branches using relative L^2 error. The second benchmark is Bratu-2D on $(0,1)^2$, evaluated on 24 parameter values in $[0.1, 6.0]$ using finite-difference residuals and center-value traces.

B. Metrics

For Bratu-1D, branch error is

$$\mathrm{RelL2}(u, \hat{u}) = \frac{\|u - \hat{u}\|_{L^2(0,1)}}{\|\hat{u}\|_{L^2(0,1)}}. \tag{18}$$

For each head and λ, we report the best match to the lower or upper analytic branch. A parameter value is counted as branch-identified if the two heads are matched to different analytic branches. We also report two-branch coverage under relative-error thresholds, PDE residual, and center-value traces.

VI. Results

A. Bratu-1D Branch Recovery

Figure 2 shows that Head A follows the lower branch and Head B follows the upper branch across almost the full tested interval. The two heads match distinct analytic branches on 31 of 32 parameter values. With a stricter accuracy requirement, both branches are recovered with best relative $L^2 < 0.2$ on 26 of 32 values and with best relative $L^2 < 0.5$ on 30 of 32 values.

The median best relative error is 6.46×10^{-2} for Head A and 3.20×10^{-3} for Head B. The upper branch is geometrically easier to identify because it is well separated in amplitude. Its residual is nevertheless larger: the mean residual is 2.09 for Head B versus 3.27×10^{-2} for Head A. This mismatch is visible in Fig. 3; the upper-branch shape is accurate for most values, while the residual grows near the fold where curvature and stiffness increase.

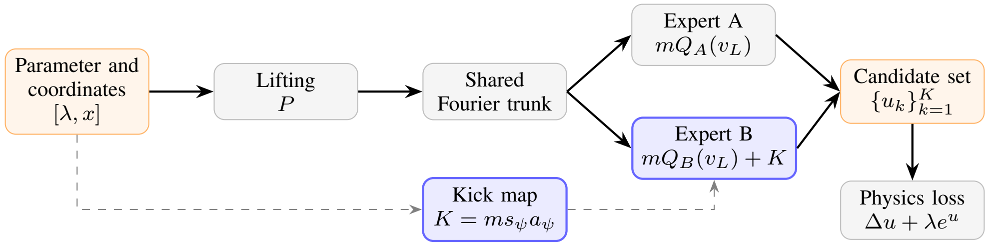

Fig. 1. MPNO architecture. A conditional Fourier trunk maps parameter-coordinate inputs to a shared representation, and multiple Dirichlet-safe expert heads produce candidate branches. The kick module is applied to the designated branch-separating expert.

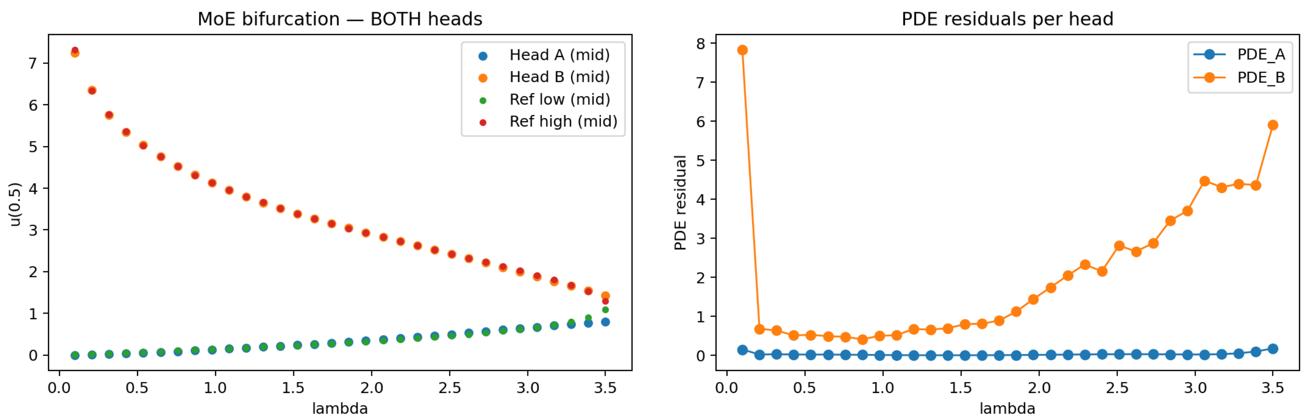

Fig. 2. Bratu-1D evaluation. Left: center-value bifurcation diagram for both MPNO heads and analytic references. Right: per-head PDE residuals across the tested parameter interval.

TABLE I
SUMMARY OF INCLUDED RESULT ARTIFACTS.

Quantity	Value
Bratu-1D parameter grid	32 values in $[0.1, 3.5]$
Branch identity coverage	$31/32$
Two-branch coverage, RelL2 < 0.2	$26/32$
Two-branch coverage, RelL2 < 0.5	$30/32$
Median best RelL2, Head A	6.46×10^{-2}
Median best RelL2, Head B	3.20×10^{-3}
Mean PDE residual, Head A/B in 1D	$0.0327/2.09$
Mean PDE residual, Head A/B in 2D	$0.101/0.234$

B. Bratu-2D Extension

The Bratu-2D run verifies that the same architecture and loss can train smooth square-domain fields. Figure 4 shows the center-value trace and representative fields at $\lambda = 3.0$. The mean PDE residual is 0.101 for Head A and 0.234 for Head B. The mean center-value separation between heads is 0.024, indicating weaker branch separation than in Bratu-1D. Thus, the 2D experiment supports residual-stable operator training, while the most conclusive multi-branch evidence in this study comes from the analytic Bratu-1D benchmark.

VII. Discussion

The Bratu-1D results demonstrate the central claim: a residual-trained neural operator with multiple expert heads can represent a multivalued solution map. The output mask enforces the boundary condition, the Fourier trunk shares parameter-dependent features, and the expert heads specialize into distinct branches. The high branch remains the harder residual target, especially near the fold, but its relative L^2 match is accurate over most of the tested interval.

The 2D results show that the formulation scales to square-domain fields, but also expose a limitation. The current 2D heads remain close in center value, so the run is better interpreted as a stable 2D residual-training result than as a complete 2D branch-separation result. This distinction is important: set-valued learning can be verified sharply when analytic references are available, and more carefully when only residual and qualitative field diagnostics are present.

VIII. Reproducibility

The code release contains YAML configs, training and evaluation scripts, result CSV files, training logs, figures, and the LaTeX source used for this paper. Bratu-1D analytic references are implemented directly from Eqs. (4)–(5), and all reported numbers in Table I are computed from the included CSV artifacts.

IX. Conclusion

MPNO provides a compact set-valued operator model for nonlinear PDEs with multiple steady states. On Bratu-1D, two residual-trained experts separate the lower and upper branches

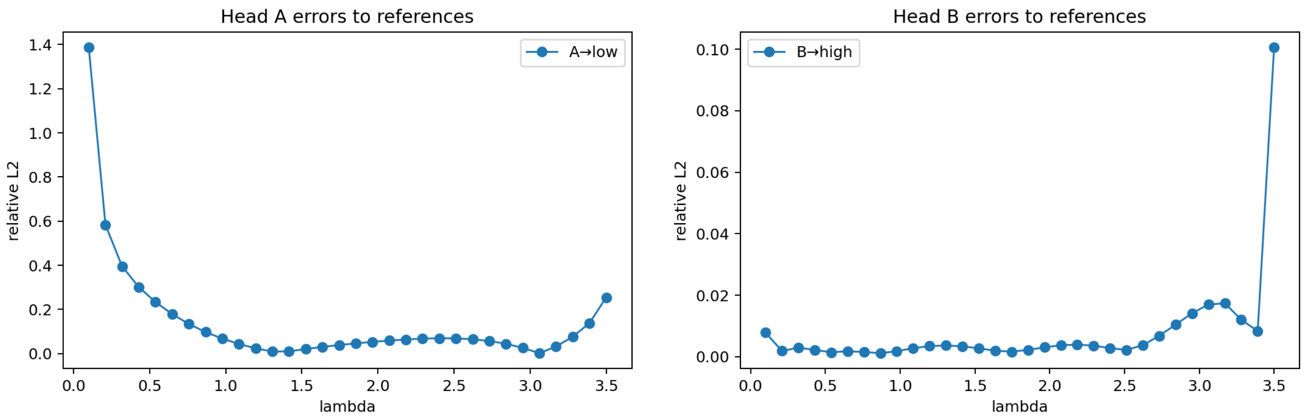

Fig. 3. Best branch-matched relative L^2 errors for Bratu-1D. Head A is matched to the lower branch for 31 of 32 sampled values. Head B is matched to the upper branch for all 32 sampled values.

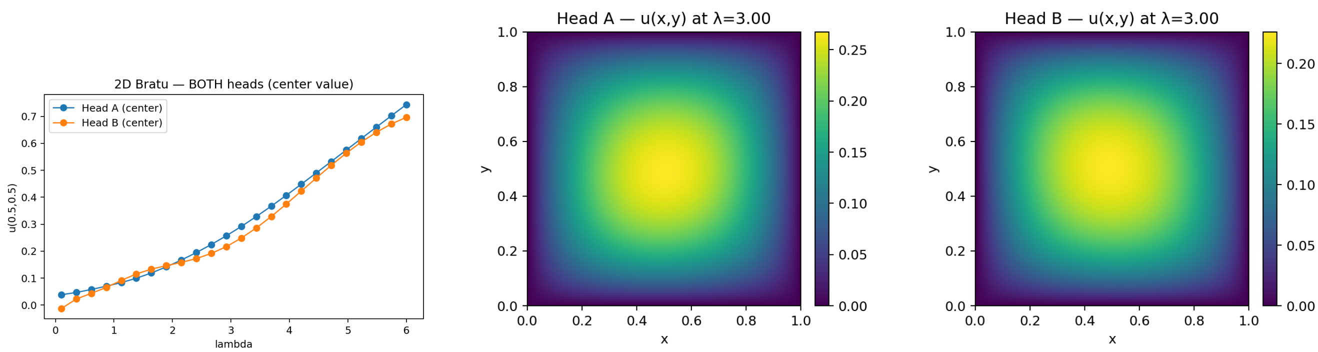

Fig. 4. Bratu-2D evaluation. Left: center values across λ. Middle and right: predicted fields at $\lambda = 3.0$ for Head A and Head B.

on 31 of 32 parameter values and recover both branches with relative $L^2 < 0.2$ on 26 of 32 values. On Bratu-2D, the same model produces smooth Dirichlet-safe fields with mean residuals below 0.25 for both heads. These results support mixture-of-experts neural operators as a practical route from single-valued surrogate modeling toward branch-aware physics-informed operator learning.

References

[1] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators," *Nature Machine Intelligence*, vol. 3, pp. 218–229, 2021.

[2] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Fourier neural operator for parametric partial differential equations," in *International Conference on Learning Representations*, 2021.

[3] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Neural operator: Learning maps between function spaces with applications to PDEs," *Journal of Machine Learning Research*, vol. 24, no. 89, pp. 1–97, 2023.

[4] J. Kossaifi, N. Kovachki, Z. Li, D. Pitt, M. Liu-Schiaffini, R. J. George, B. Bonev, K. Azizzadenesheli, J. Berner, and A. Anandkumar, "A library for learning neural operators," 2024.

[5] L. Li, X.-C. Tai, and R. H.-F. Chan, "A new method to compute the blood flow equations using the physics-informed neural operator," *Journal of Computational Physics*, vol. 519, p. 113380, 2024.

[6] M. S. Eshaghi, C. Anitescu, M. Thombre, Y. Wang, X. Zhuang, and T. Rabczuk, "Variational physics-informed neural operator (VINO) for solving partial differential equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 437, p. 117785, 2025.

[7] A. Mohsen, "A simple solution of the Bratu problem," *Computers and Mathematics with Applications*, vol. 67, pp. 26–33, 2014.

[8] M. L. Shahab, H. Susanto, and H. Hatzikirou, "A finite difference method with symmetry properties for the high-dimensional Bratu equation," *Applied Mathematics and Computation*, vol. 489, p. 129136, 2025.

[9] H. Uecker, "Continuation and bifurcation in nonlinear PDEs: Algorithms, applications, and experiments," *Jahresbericht der Deutschen Mathematiker-Vereinigung*, vol. 124, pp. 43–80, 2021.

[10] H. Uecker, D. Wetzel, and J. D. M. Rademacher, "pde2path: A Matlab package for continuation and bifurcation in 2d elliptic systems," *Numerical Mathematics: Theory, Methods and Applications*, vol. 7, no. 1, pp. 58–106, 2014.

[11] P. E. Farrell, A. Birkisson, and S. W. Funke, "Deflation techniques for finding distinct solutions of nonlinear partial differential equations," *SIAM Journal on Scientific Computing*, vol. 37, no. 4, pp. A2026–A2045, 2015.

[12] M. L. Shahab and H. Susanto, "Neural networks for bifurcation and linear stability analysis of steady states in partial differential equations," *Applied Mathematics and Computation*, vol. 483, p. 128985, 2024.

[13] Z. Zou, Z. Wang, and G. E. Karniadakis, "Learning and discovering multiple solutions using physics-informed neural networks with random initialization and deep ensemble," 2025.

[14] P. Tarancon-Alvarez, P. Tejerina-Perez, R. Jimenez, and P. Protopapas, "Efficient PINNs via multi-head unimodular regularization of the solutions space," *Communications Physics*, 2025.

[15] K. Li and W. Ye, "D-FNO: A decomposed fourier neural operator for

large-scale parametric partial differential equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 436, p. 117732, 2025.

[16] J. Kossaifi *et al.*, "DoMINO: A decomposable multi-scale iterative neural operator for modeling large scale engineering simulations." arXiv preprint, 2024.

[17] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[18] S. Wang, X. Yu, and P. Perdikaris, "When and why PINNs fail to train: A neural tangent kernel perspective," *Journal of Computational Physics*, vol. 449, p. 110768, 2022.

[19] R. Mattey and S. Ghosh, "A novel sequential method to train physics informed neural networks for Allen Cahn and Cahn Hilliard equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 390, p. 114474, 2022.

[20] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, 1991.

[21] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in *International Conference on Learning Representations*, 2017.

[22] Q. Huang, Z. An, N. Zhuang, M. Tao, C. Zhang, Y. Jin, K. Xu, L. Chen, S. Huang, and Y. Feng, "Harder tasks need more experts: Dynamic routing in MoE models," 2024.

[23] X. Wu, S. Huang, W. Wang, and F. Wei, "Multi-head mixture-of-experts," in *Advances in Neural Information Processing Systems*, 2024.